ДЕПАРТАМЕНТ ОБРАЗОВАНИЯ И НАУКИ БРЯНСКОЙ ОБЛАСТИ

ЖУКОВСКИЙ ФИЛИАЛ
ГОСУДАРСТВЕННОГО БЮДЖЕТНОГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАТЕЛЬНОГО УЧРЕЖДЕНИЯ
«БРЯНСКИЙ АГРАРНЫЙ ТЕХНИКУМ ИМЕНИ ГЕРОЯ РОССИИ А.С. ЗАЙЦЕВА»

Пособие для студентов

ПМ «Настройка сетевой инфраструктуры»

специальности 09.02.06 «Сетевое и системное администрирование»

Разработчик:
преподаватель высшей
квалификационной категории
И.А. Козенцев
(в тексте использовались фрагменты
сгенерированные нейросетями)

г. Жуковка
2025 г.

Оглавление

ТЕМА-1. Компьютерные сети

Введение

Представьте себе мир без компьютерных сетей: каждый компьютер — это изолированный остров, информация не может путешествовать, общение между людьми ограничено физической передачей дискет или флешек. Компьютерная сеть — это как система дорог, связывающая эти острова в единый материк, позволяя данным, ресурсам и услугам свободно перемещаться. Сегодня сети — это кровеносная система современного общества, от работы банкомата до просмотра видео в интернете. Данный материал закладывает фундамент для понимания, как устроены эти "цифровые дороги", как они классифицируются, из чего состоят и как обеспечивают нашу связь с миром.

1. Виды компьютерных сетей. Глобальные и локальные сети

Базовое определение: Компьютерная сеть — это совокупность компьютеров и других устройств, соединенных каналами связи для обмена информацией и совместного использования ресурсов (принтеров, файлов, интернет-канала).

Ключевая идея здесь — **совместное использование**. Аналогия: в семье есть один принтер, к которому могут печатать все члены семьи со своих ноутбуков и телефонов — это уже простейшая сеть. Или общая папка на сервере компании, куда все сотрудники складывают отчеты, вместо того чтобы постоянно пересылать их по почте.

Сети классифицируют в первую очередь по **географическому признаку** — по размеру покрываемой территории.

1.1. Локальная вычислительная сеть (ЛВС, LAN — Local Area Network)

Определение: LAN — это сеть, ограниченная небольшой географической зоной: одним зданием, офисом, этажом, квартирой или даже одной комнатой.

Смысл и примеры: Главная задача LAN — обеспечить быструю и эффективную связь между устройствами внутри одной организации или дома. Представьте себе офис небольшой фирмы. Все компьютеры сотрудников соединены между собой и с общим сервером, где лежат базы данных и документы. Сотрудники могут:

- Мгновенно открывать общие файлы.

- Печатать на сетевых принтерах.
- Проводить внутренние видеоконференции.
- Играть по сети в одной комнате.

Характерные черты LAN:

- **Высокая скорость передачи данных:** Современные LAN (например, на основе технологии Ethernet) обеспечивают скорости 100 Мбит/с, 1 Гбит/с (Гигабит) и даже 10 Гбит/с. Это как широкий многополосный скоростной автобан внутри предприятия.
- **Низкая стоимость передачи данных:** Так как сеть своя, собственная, то за передачу каждого мегабайта внутри нее не нужно платить провайдеру.
- **Ограниченная территория:** Управление и обслуживание сети осуществляется одной организацией или человеком.

Аналогия: LAN — это как внутренняя телефонная связь в большом здании (набрав короткий номер, вы мгновенно соединяетесь с коллегой в соседнем кабинете). Или как система трубопроводов и электросетей внутри вашего дома — они ваши, быстрые и локальные.

1.2. Глобальная вычислительная сеть (ГВС, WAN — Wide Area Network)

Определение: WAN — это сеть, охватывающая огромные географические пространства: города, страны, континенты и всю планету. Самый известный пример WAN — это Интернет.

Смысл и примеры: Если LAN связывает устройства "внутри", то WAN связывает "внешние" объекты. Компания имеет головной офис в Москве и филиалы в Санкт-Петербурге и Владивостоке. Чтобы их сети стали единым целым, они арендуют каналы связи у телекоммуникационных провайдеров. Эти каналы (спутниковые, оптоволоконные, радиоканалы) и образуют WAN-соединение между локальными сетями филиалов.

Характерные черты WAN:

- **Огромное расстояние:** Может соединять континенты.
- **Более низкая скорость по сравнению с LAN:** Хотя магистральные каналы очень быстры, задержки (латентность) из-за огромных расстояний всегда присутствуют. Данным нужно время, чтобы "долететь" из одной точки планеты в другую. Если LAN — это автобан, то WAN — это система межконтинентальных авиаперелетов: в целом быстро, но есть время на взлет, посадку и сам перелет.
- **Использование инфраструктуры публичных операторов:** Обычно WAN строится на арендованных каналах связи (например, у операторов "Ростелеком", "Транстелеком").
- **Более высокая стоимость:** Аренда каналов и доступ в интернет — платные услуги.

Аналогия: WAN — это как система почтовой или курьерской службы между городами и странами. Вы можете отправить письмо (пакет данных) в любую точку мира, используя общедоступную инфраструктуру (почтовые отделения, аэропорты), но это требует времени и оплаты услуг перевозчика.

Важное уточнение: Часто можно встретить промежуточные понятия, такие как:

- **Городская сеть (MAN — Metropolitan Area Network):** Сеть в пределах города. Например, сеть кабельного телевидения или Wi-Fi покрытие в центре мегаполиса.
- **Персональная сеть (PAN — Personal Area Network):** Самая маленькая сеть вокруг одного человека. Например, соединение смартфона с беспроводными наушниками по Bluetooth или с умными часами.

2. Виды сетевых архитектур

Сетевая архитектура определяет, **как организовано взаимодействие между компьютерами в сети**, как они общаются друг с другом и кто чем управляет. Это логическая, а не физическая структура.

2.1. Одноранговая сеть (Peer-to-Peer, P2P)

Определение: В одноранговой сети все компьютеры (узлы) равноправны. Каждый компьютер может выступать и как клиент (запрашивать ресурсы), и как сервер (предоставлять ресурсы). Централизованного управления нет.

Смысл и примеры: Идеально подходит для небольших сетей (до 10-15 компьютеров), где не требуется высокий уровень безопасности или централизованное управление.

- **Пример 1:** Домашняя сеть. Несколько ноутбуков и смартфонов соединены через Wi-Fi-роутер. Вы можете расшарить папку с фильмами на одном ноутбуке и смотреть эти фильмы с другого.
- **Пример 2:** Файлообменные сети (торренты). Каждый пользователь, скачивая файл, одновременно начинает раздавать (предоставлять) уже скачанные части другим пользователям. Нет одного главного сервера со всем содержимым.

Преимущества:

- Простота организации и низкая стоимость (не нужен дорогой сервер).
- Отказоустойчивость: выход из строя одного компьютера не парализует всю сеть.

Недостатки:

- Низкая безопасность: управление доступом к ресурсам размазано по всем компьютерам.
- Сложность администрирования и резервного копирования в больших сетях.
- Производительность падает с ростом числа узлов.

Аналогия: Одноранговая сеть — это как круглый стол переговоров, где все участники равны. Каждый может задать вопрос и каждый может на него ответить. Или как бартерная экономика в небольшой деревне, где каждый что-то производит и обменивается с соседями напрямую.

2.2. Сеть на основе сервера (Клиент-Серверная архитектура)

Определение: В такой сети есть четкое разделение ролей. Выделяются специальные, более мощные и надежные компьютеры — **серверы**, которые предоставляют ресурсы (файлы, приложения, доступ в интернет, почтовые услуги). Остальные компьютеры — **клиенты** (рабочие станции) — используют эти ресурсы.

Смысл и примеры: Это стандарт для средних и крупных организаций (бизнес, образование, государственные учреждения).

- **Пример 1:** Файловый сервер в компании. Все важные документы хранятся централизованно на защищенном сервере с регулярным резервным копированием. Сотрудники со своих рабочих станций (клиентов) обращаются к нему для работы.
- **Пример 2:** Веб-сервисы. Когда вы открываете сайт (например, vk.com), ваш браузер (клиент) отправляет запрос на веб-сервер ВКонтакте. Сервер обрабатывает запрос и отправляет вам обратно код страницы, который отображается в браузере.
- **Пример 3:** Онлайн-игры (MMO). Игровой клиент на вашем ПК соединяется с игровым сервером компании-разработчика, который хранит состояние всего игрового мира и синхронизирует действия всех игроков.

Преимущества:

- **Высокая безопасность:** Управление доступом централизовано, можно гибко настраивать права для каждого пользователя.
- **Централизованное управление и резервное копирование:** Администратор управляет всеми ресурсами из одной точки.
- **Масштабируемость и производительность:** Мощные серверы справляются с нагрузкой от сотен и тысяч клиентов.

Недостатки:

- **Высокая стоимость:** Необходимо приобретать специализированное серверное оборудование и ПО, оплачивать работу системного администратора.
- **Наличие единой точки отказа:** Выход из строя критически важного сервера (например, контроллера домена) может парализовать работу всей сети.

Аналогия: Клиент-серверная архитектура — это как ресторан. Вы (клиент) делаете заказ официанту. Официант передает заказ на кухню (серверу). Повара (сервер) готовят блюдо, используя централизованные запасы продуктов и оборудование, а затем через официанта предоставляют его вам. Кухня — мощный, специализированный центр, обслуживающий множество клиентов.

3. Основные компоненты сетей, сетевая среда и сетевые устройства

Любая сеть состоит из "активных" устройств, "пассивной" среды и конечных точек.

3.1. Сетевая среда передачи данных (Среда связи)

Это физическая "дорога", по которой бегут данные.

1. **Проводные среды:**
   - **Витая пара (UTP/STP):** Самый распространенный тип кабеля в LAN. Представляет собой несколько пар изолированных медных проводов, скрученных вместе. Скручивание снижает электромагнитные помехи. Бывает разной категории (Cat 5e, Cat 6, Cat 7), определяющей максимальную скорость. **Аналогия:** Телефонный кабель, но с большим количеством жил и лучше защищенный.
   - **Оптоволоконный кабель:** Использует световые импульсы для передачи данных через тонкие стеклянные или пластиковые нити. Обеспечивает огромные скорости (десятки и сотни Гбит/с) и нечувствителен к электромагнитным помехам. Используется для магистральных каналов WAN и в высокоскоростных LAN. **Аналогия:** Если витая пара — это медная труба для воды, то оптоволокно — это прозрачный световод, по которому пускают "зайчики" от фонарика с очень быстрым морганием.
2. **Беспроводные среды:**
   - **Радиоволны (Wi-Fi, Bluetooth, сотовые сети):** Данные передаются с помощью модулированных радиосигналов определенных частот. **Аналогия:** Как радиостанция, только вместо музыки передаются нули и единицы.

3.2. Конечные устройства (Узлы)

Это устройства, которые являются источником или получателем информации в сети.

- **Персональные компьютеры, ноутбуки, смартфоны, планшеты, серверы, сетевые принтеры, IP-камеры.** У всех у них есть **сетевой адаптер (сетевая карта, NIC)**, который позволяет подключаться к сетевой среде (разъем для кабеля или Wi-Fi-антенна).

3.3. Сетевые устройства (Активное оборудование)

Это "интеллектуальные регуляторы движения" на сетевых дорогах.

1. **Сетевой концентратор (Hub):** Примитивное устройство. Получает сигнал на один порт и просто копирует его на **все остальные порты**. Не понимает адресов. **Проблема:** Создает излишний трафик и коллизии (когда два устройства начинают передачу одновременно). **Аналогия:** Громкоговоритель в комнате: кто что скажет — слышат все, и если говорят двое, возникает каша.

2. **Коммутатор (Switch):** "Умный" наследник хаба. Он **изучает** MAC-адреса (физические адреса сетевых карт) устройств, подключенных к его портам, и строит таблицу коммутации. При получении кадра данных он отправляет его **только на тот порт, к которому подключен адресат**. Это резко уменьшает нагрузку на сеть. **Аналогия:** Телефонный оператор на старой АТС: он знает, какой шнур к какому абоненту ведет, и соединяет звонящего напрямую с нужным собеседником, не подключая всех остальных.
3. **Маршрутизатор (Router):** Ключевое устройство для соединения **разных сетей**. Его основная задача — **маршрутизация**: определение оптимального пути для пакета данных из одной сети в другую (например, из вашей домашней LAN в глобальную сеть Интернет). Работает на основе IP-адресов. **Аналогия:** Почтовое отделение на границе города. Оно получает письма (пакеты) от жителей своего города (LAN), смотрит адрес назначения и решает, отправить ли письмо в другой район этого же города (другая LAN) или загрузить в почтовую машину, следующую в другой город (WAN/Интернет).
4. **Точка доступа беспроводной сети (Access Point, AP):** Устройство, создающее беспроводную сеть (Wi-Fi) и подключающее беспроводных клиентов к проводной сети (часто через коммутатор). **Аналогия:** Беспроводной удлинитель сигнала. Подключается к "проводному миру" и транслирует его в "эфир" в виде радиосигнала Wi-Fi.
5. **Маршрутизатор Wi-Fi (домашний роутер):** В быту чаще всего используется комбинированное устройство, которое включает в себя функции маршрутизатора, коммутатора (4-8 портов), точки доступа Wi-Fi и часто межсетевого экрана. Это "шлюз" между вашей домашней LAN и сетью провайдера.

4. Технологии подключения к Интернет

Интернет — это глобальная WAN. Чтобы подключить к ней локальную сеть или отдельный компьютер, используются различные технологии "последней мили" — от провайдера до пользователя.

4.1. Проводные технологии

1. **xDSL (ADSL, VDSL):** Технология, использующая обычную телефонную медную пару. Позволяет одновременно говорить по телефону и пользоваться интернетом, разделяя частоты. **ADSL** — асимметричный: скорость скачивания выше, чем скорость отдачи. **Аналогия:** По одной и той же грунтовой дороге (телефонная линия) могут ехать две разные машины (голос и данные) по разным полосам (частотам).
2. **Docsis (коаксиальный кабель):** Технология, используемая кабельными телевизионными операторами. Передача данных идет по тому же коаксиальному кабелю, что и телесигнал. **Аналогия:** По одному широкому трубопроводу (кабель) текут два разных потока: цветной (ТВ) и прозрачный (интернет).
3. **Оптоволокно до... (FTTx):** Самый современный и быстрый способ.

- **FTTH (Fiber To The Home):** Оптоволокно заведено прямо в квартиру. **Аналогия:** К вашему дому подведен прямой высокоскоростной магистральный газопровод (данных).

- **FTTB (Fiber To The Building):** Оптоволокно подведено к дому (в подъезд), а дальше по квартирам разводится витой парой. **Аналогия:** Магистраль подведена к дому, а по стоякам разводятся трубы поменьше.

4. **Ethernet-подключение:** Провайдер заводит в помещение обычный Ethernet-кабель (витая пара) и назначает вам IP-адрес. По сути, ваша сеть становится частью большой сети провайдера.

4.2. Беспроводные технологии

1. **Спутниковый интернет:** Сигнал идет через спутник на геостационарной орбите. Используется там, где нет проводной инфраструктуры. Главный недостаток — большая задержка (ping), так как сигналу нужно пройти ~36 000 км до спутника и обратно. **Аналогия:** Общение через переводчика, который находится очень далеко. Фраза и ответ идут долго.
2. **Мобильный интернет (3G/4G LTE/5G):** Использует инфраструктуру сотовой связи. Данные передаются через базовые станции (вышки) оператора. Позволяет быть онлайн почти везде, где есть покрытие. **Аналогия:** Голосовая связь по мобильному телефону, но вместо голоса передаются данные.

5. Качество и надежность сетей

Сеть должна не просто работать, а работать **хорошо**. Ключевые параметры:

1. **Пропускная способность (Bandwidth):** Максимальное количество данных, которые можно передать по каналу связи за единицу времени. Измеряется в битах в секунду (бит/с, Кбит/с, Мбит/с, Гбит/с). **Аналогия:** Ширина трубы или количество полос на шоссе. Чем шире, тем больше машин (данных) может проехать одновременно.
2. **Задержка (Latency, Ping):** Время, за которое пакет данных доходит от источника до получателя и возвращается обратно. Измеряется в миллисекундах (мс). Критически важно для онлайн-игр, видеозвонков. **Аналогия:** Время, которое проходит между вопросом, заданным по телефону, и получением ответа. Даже если голос четкий (высокая пропускная способность), большая задержка сделает диалог неудобным.
3. **Джиттер (Jitter):** Нестабильность задержки. Если пакеты приходят с разной скоростью (один за 10 мс, следующий за 50 мс), это вызывает проблемы, например, "рваный" звук в VoIP. **Аналогия:** Вы ждете автобусы, которые должны ходить каждые 10 минут. Но один приходит через 5 минут, следующий — через 15. Это нестабильность — джиттер.
4. **Надежность и доступность (Uptime):** Процент времени, когда сеть работает исправно. Измеряется в "девятках". Например, доступность 99.9% означает, что сеть может простаивать не более 8.76 часов в год. Обеспечивается резервированием оборудования и каналов связи.
5. **Безопасность:** Часть качества сети — это защищенность данных от угроз. Перетекает в следующий раздел.

6. Основные понятия сетевой безопасности

Сетевая безопасность — это комплекс мер, направленных на защиту данных, ресурсов и инфраструктуры сети от несанкционированного доступа, атак, повреждения или уничтожения.

Угрозы:

- **Вирусы, черви, трояны:** Вредоносное ПО, которое может уничтожить данные или использовать ваш компьютер для атак на других.
- **Атаки типа "Отказ в обслуживании" (DoS/DDoS):** Злоумышленники направляют на сервер или сеть огромный поток ложных запросов, перегружая его и делая недоступным для легитимных пользователей. **Аналогия:** Толпа людей, которая одновременно звонит в одну и ту же службу доставки пиццы с фальшивыми заказами, так что настоящие клиенты не могут дозвониться.
- **Взлом и несанкционированный доступ:** Попытки получить доступ к данным или системам, на которые у злоумышленника нет прав.
- **Фишинг:** Обман пользователей с целью заставить их раскрыть конфиденциальную информацию (логины, пароли, данные карт) через поддельные сайты или письма. **Аналогия:** Мошенник, переодетый в полицейского, чтобы вы добровольно отдали ему кошелек.

2. **Средства защиты:**

- **Антивирусное ПО:** "Иммунная система" компьютера.
- **Межсетевой экран (Firewall):** "Пограничный контрольно-пропускной пункт" между сетями (например, между вашей LAN и Интернетом). Анализирует входящий и исходящий трафик по заданным правилам и блокирует потенциально опасный. **Аналогия:** Охранник на входе в здание, который проверяет пропуска и не пускает посторонних.
- **Шифрование:** Преобразование данных в нечитаемый вид с помощью специального ключа. Даже если данные перехватят, их нельзя будет прочесть без ключа. Используется в HTTPS, VPN. **Аналогия:** Отправка письма не текстом, а секретным шифром, который знаете только вы и ваш адресат.
- **Виртуальные частные сети (VPN):** Создают зашифрованный "туннель" через общедоступную сеть (Интернет). Для внешнего наблюдателя ваш трафик выглядит как бессмысленный шифропоток, идущий на сервер VPN, а ваше местоположение маскируется. **Аналогия:** Вы строите закрытый, охраняемый мост (туннель) через бурную реку (Интернет), по которому ваши ценные грузы (данные) перевозятся в полной безопасности.
- **Системы обнаружения и предотвращения вторжений (IDS/IPS):** "Сигнализация и автоматическая защита", которые не только обнаруживают атаку по аномальной активности, но и могут автоматически на нее отреагировать (например, заблокировать атакующий IP-адрес).

7. Тенденции развития сетей

Сети постоянно эволюционируют. Вот ключевые направления:

1. **Повсеместное внедрение 5G:** Новое поколение мобильной связи обедает не просто высокую скорость, а **сверхнизкие задержки** (менее 1 мс) и возможность подключения огромного количества устройств на небольшой площади. Это основа для Интернета вещей (IoT) и беспилотного транспорта. **Аналогия:** Переход от обычных дорог к сети "умных" автобанов, где машины общаются друг с другом и инфраструктурой в реальном времени.
2. **Программно-определяемые сети (SDN — Software-Defined Networking):** Отделение "мозга" сети (управляющей плоскости) от "мускулов" (устройств, пересылающих данные). Управление сетью становится централизованным, гибким и программируемым с помощью ПО, а не настройкой каждого устройства в отдельности. **Аналогия:** Раньше каждый светофор на перекрестке (сетевое устройство) работал по своей программе. SDN — это единый центр управления движением, который в реальном времени видит всю картину и гибко переключает все светофоры для оптимизации потока.
3. **Массовое развитие Интернета вещей (IoT):** В сеть включаются не компьютеры и смартфоны, а любые устройства: датчики, лампочки, холодильники, станки, автомобили. Сети должны адаптироваться к работе с миллиардами маломощных устройств, передающих небольшие объемы данных.
4. **Повышение роли искусственного интеллекта (ИИ) в сетях:** ИИ используется для прогнозирования нагрузки, автоматического обнаружения и устранения аномалий и атак, оптимизации маршрутизации. **Аналогия:** Автопилот для управления сложной сетевой инфраструктурой.
5. **Повышение значимости безопасности (Security by Design):** Безопасность перестает быть "дополнением" и становится неотъемлемой частью проектирования сетей и устройств с самого начала.

Заключение

Мы рассмотрели фундаментальные основы компьютерных сетей: от их классификации и архитектур до ключевых компонентов, технологий подключения, критериев качества и основ безопасности. Понимание этих концепций является критически важным первым шагом для будущего сетевого и системного администратора. Это тот язык, на котором говорят специалисты, и та карта, которая позволяет ориентироваться в сложном и постоянно меняющемся мире цифровых коммуникаций. Дальнейшее обучение будет углублять и детализировать каждую из этих тем, наполняя их конкретными технологиями, протоколами и практиками настройки.

Контрольные вопросы к ТЕМЕ-1

1. По какому основному признаку классифицируют компьютерные сети на локальные (LAN) и глобальные (WAN)? Кратко охарактеризуйте каждую из них.
2. Приведите пример локальной (LAN) и пример глобальной (WAN) сети из повседневной жизни.
3. В чём основное различие между одноранговой (P2P) сетью и сетью на основе сервера (клиент-сервер)?
4. Где логично использовать одноранговую архитектуру, а где — клиент-серверную? Обоснуйте свой ответ.
5. Перечислите основные компоненты компьютерной сети (среду, устройства, узлы).
6. Какая основная функция у сетевого коммутатора (switch) и чем он принципиально лучше концентратора (hub)?
7. Для чего предназначен маршрутизатор (router) и в чём его ключевое отличие от коммутатора?
8. Назовите три различные технологии проводного подключения к Интернету.
9. Объясните простыми словами разницу между двумя параметрами качества сети: «пропускная способность» (bandwidth) и «задержка» (latency).
10. Что такое DDoS-атака и как она нарушает нормальную работу сети или сервера?
11. С помощью какого базового средства защиты можно создать зашифрованный канал для передачи данных через общедоступную сеть Интернет?
12. Какую ключевую возможность, важную для развития Интернета вещей (IoT) и беспилотных систем, предоставляет технология связи 5G по сравнению с предыдущими поколениями?

Практические задания по ТЕМЕ-1

Задание 1. Выбор архитектуры для нового офиса

Вы начинающий сетевой администратор. Небольшой стартап по разработке мобильных приложений (8 человек) арендует open-space помещение в бизнес-центре Казани и просит вас организовать для них компьютерную сеть. У каждого сотрудника есть ноутбук, есть один мощный компьютер для сборки проектов, три сетевых принтера и потребность в общем файловом хранилище.

- **Вопрос:** Какую сетевую архитектуру (одноранговую или клиент-сервер) вы порекомендуете в данной ситуации и почему? Аргументируйте свой ответ, опираясь на преимущества и недостатки каждой из архитектур.

Задание 2. Диагностика проблемы с сетью в школе

В городской школе г. Тюмени в компьютерном классе из 12 ПК, соединенных в сеть, резко упала общая скорость обмена файлами между компьютерами. При этом доступ в Интернет через маршрутизатор работает нормально. Вы обнаруживаете, что вместо сетевого коммутатора в классе много лет использовался старый сетевой концентратор (hub).

- **Вопрос:** Объясните, почему замена концентратора на коммутатор, скорее всего, решит проблему со скоростью в локальной сети класса. В своем ответе опишите принцип работы обоих устройств.

Задание 3. Выбор технологии подключения для филиала

Компания из Екатеринбурга открывает небольшой удаленный филиал (офис на 5 человек) в посёлке городского типа в Свердловской области. В посёлке нет оптоволоконной инфраструктуры крупных провайдеров. Доступны следующие варианты подключения к Интернету для работы с корпоративной почтой и 1С: 1) ADSL через местную телефонную станцию, 2) Спутниковый интернет.

- **Вопрос:** Какой из двух вариантов вы посоветуете выбрать для обеспечения стабильной работы с бухгалтерской программой и видеозвонками? Дайте развернутый ответ, сравнив технологии по ключевым параметрам из теоретического материала.

Задание 4. Анализ инцидента безопасности

В приемную главного врача больницы в г. Воронеже поступил тревожный звонок из регистратуры. Сообщили, что на всех компьютерах в регистратуре появилось всплывающее окно с требованием заплатить деньги для разблокировки доступа к медицинским картам пациентов. Предварительный анализ показал, что в отделении отсутствовало обновленное антивирусное ПО, а один из сотрудников ранее открыл вложение из подозрительного письма, якобы от «Роспотребнадзора».

- **Вопрос:** Опираясь на материал, назовите не менее двух видов сетевых угроз, с которыми, вероятно, столкнулась больница. Какие базовые средства защиты могли бы предотвратить или смягчить такой инцидент?

Задание 5. Планирование сети для «умного дома»

Клиент в новостройке г. Сочи хочет организовать в своей квартире «умный дом». Он планирует установить 5 IP-камер видеонаблюдения, 10 умных лампочек, датчики температуры и протечки воды, умные розетки, а также обеспечить бесперебойный Wi-Fi для 4K-видео на нескольких телевизорах. Существующий домашний Wi-Fi-роутер его не устраивает.

- **Вопрос:** Исходя из современных тенденций развития сетей, какую ключевую характеристику (помимо высокой скорости) должен обеспечивать новый маршрутизатор или сетевое решение для стабильной и одновременной работы всех перечисленных устройств «умного дома»? Ответ поясните.

ТЕМА-2. Сетевые протоколы и коммуникации

Введение

Представьте себе огромный город с миллионами жителей, говорящих на разных языках, использующих разные виды транспорта и следующие разным правилам. Чтобы город функционировал, нужны единые правила: дорожное движение, языки общения, стандарты для строительства и коммуникаций. Компьютерная сеть – это такой же цифровой "город", где устройства (компьютеры, серверы, телефоны, принтеры) должны понимать друг друга. Изучаемая тема – это и есть "правила дорожного движения" и "языки общения" в мире сетей. Мы разберем, как информация превращается в сигналы, как устройства договариваются о передаче, какие организации устанавливают эти правила и как данные путешествуют от вашей клавиатуры до удаленного сервера и обратно.

1. Кодирование и параметры сообщения

Базовая идея: Любое сообщение (текст, изображение, звук) для передачи по сети должно быть преобразовано в универсальный, понятный всем устройствам цифровой вид – последовательность битов (0 и 1). Этот процесс называется кодированием.

Смысл и аналогия: Представьте, что вы хотите отправить другу письмо. Вы думаете идею (сообщение в уме), затем записываете ее словами на бумаге (кодирование в текст). Но если ваш друг не знает ваш язык, письмо будет бесполезным. Поэтому вы договариваетесь об общем языке (стандарте кодирования). В цифровом мире таким универсальным "алфавитом" являются биты.

Примеры и пояснения:

- **Текст:** Буква "A" в кодировке ASCII представляется как число 65, которое в двоичной системе – 01000001. Каждая буква, цифра, знак препинания имеют свой уникальный двоичный код.
- **Изображение:** Картинка разбивается на крошечные точки – пиксели. Каждый пиксель кодируется набором битов, описывающих его цвет. Например, в системе RGB цвет представляется тремя числами (от 0 до 255) для красного, зеленого и синего каналов. Красный цвет (255, 0, 0) в двоичном виде – это три группы по 8 бит.
- **Параметры сообщения:** Это его характеристики, которые важны для передачи.
  - **Размер (объем данных):** Измеряется в битах, байтах, килобайтах. Как вес посылки на почте. Большой файл (тяжелая посылка) может требовать разбивки на части.

- **Формат (тип данных):** Определяется расширением файла (.txt, .jpg, .mp3) или специальными заголовками внутри данных. Как отметка на конверте "Документы" или "Хрупкое". Это помогает принимающей стороне понять, как интерпретировать биты.
- **Важность/Приоритет:** Некоторые данные критичны (например, сигнал управления в VoIP-звонке), другие могут подождать (загрузка обновления). Как срочное письмо с отметкой "Express" vs еженедельная рассылка.

Итог: Кодирование – это перевод информации на "машинный язык" (биты). Параметры сообщения – это мета-информация, описывающая эти биты, необходимая для их корректной обработки и доставки.

2. Сетевые протоколы. Взаимодействие протоколов

Базовая идея: Сетевой протокол – это строго определенный набор правил и соглашений, по которым происходит обмен данными между устройствами в сети.

Смысл и аналогия: Протокол – это не программа, а именно правила. Это как сценарий или регламент дипломатической встречи. Он определяет: 1) **Формат данных** (как выглядит "фраза"), 2) **Последовательность действий** (кто говорит первым, как отвечают), 3) **Способ обработки ошибок** (что делать, если сообщение не расслышали).

Примеры и пояснения:

- **Протокол HTTP (HyperText Transfer Protocol):** Правила для обмена веб-страницами.
- **Правило:** Клиент (браузер) отправляет запрос `GET /index.html HTTP/1.1`. Сервер, получив эту точно сформулированную фразу, понимает, что от него хотят, и отвечает: `HTTP/1.1 200 OK`, а затем отправляет содержимое файла.
- **Взаимодействие протоколов:** Сложные задачи решаются не одним, а набором (стеком) протоколов, работающих совместно, как отделы в компании.
- **Аналогия с отправкой посылки:** Вы (пользователь) хотите отправить книгу другу.

1. Вы упаковываете книгу в коробку, пишете адрес **получателя** и **отправителя** (работает **протокол прикладного уровня**, например, почтовое приложение).
2. Вы относите коробку на почту. Почта проверяет правила отправки (вес, размер, запрещенные предметы) и наклеивает свою транспортную накладную с штрих-кодом (**протокол транспортного уровня**, например, TCP).
3. Почта решает, как везти посылку: самолетом, поездом, авто. Определяет маршрут и пункты сортировки (**протокол сетевого уровня**, например, IP).
4. В каждом пункте сортировки посылку грузят на конкретный грузовик, который следует по конкретному шоссе (**протокол канального уровня**, например, Ethernet).

- Каждый протокол решает свою задачу, используя услуги протокола уровня ниже и предоставляя услуги протоколу уровня выше. Они взаимодействуют через стандартные интерфейсы.

Итог: Протоколы – это "правила игры" в сети. Их взаимодействие строится по принципу разделения ответственности, что делает сетевую коммуникацию гибкой и надежной.

3. Набор протоколов TCP/IP и процесс обмена данными

Базовая идея: TCP/IP – это не один протокол, а целое семейство (стек) протоколов, которое является фундаментом современного интернета. Он реализует практический подход к передаче данных.

Смысл и аналогия: Если представить связь как процесс отправки письма, то:

- **IP (Internet Protocol)** – отвечает за **адресацию и маршрутизацию**. Это как конверт с IP-адресами отправителя и получателя. IP "не заботится" о содержимом, его задача – доставить пакет до нужного сетевого интерфейса в глобальной сети. Он **ненадежный** (не гарантирует доставку) и **не устанавливает соединение** (дейтаграммный режим).
- **TCP (Transmission Control Protocol)** – отвечает за **надежную доставку потока данных**. Это как служба курьерской доставки с подтверждением. TCP устанавливает виртуальное соединение (рукопожатие), разбивает большой объем данных на сегменты, нумерует их, ждет подтверждения о получении и в случае потери – пересылает заново.

Процесс обмена данными по TCP/IP (упрощенный пример – загрузка страницы):

1. Вы вводите yandex.ru в браузере.
2. Приложение (браузер) использует протокол **DNS** (часть стека TCP/IP), чтобы узнать IP-адрес сервера Яндекса (например, 77.88.55.55).
3. Браузер формирует HTTP-запрос. Этот запрос передается **транспортному уровню**.
4. **Транспортный уровень (TCP)** разбивает запрос на сегменты, добавляет служебную информацию (номера портов источника и назначения, например, порт браузера 54321 и порт веб-сервера 80). **Порт** – это идентификатор приложения на устройстве, аналог номера квартиры в доме (где IP-адрес – это адрес дома).
5. **Сетевой уровень (IP)** берет TCP-сегмент, упаковывает его в IP-пакет, добавляя IP-адреса источника (ваш) и назначения (77.88.55.55).
6. **Канальный уровень (например, Ethernet)** упаковывает IP-пакет в **кадр (frame)**, добавляя MAC-адреса источника и следующего узла (например, вашего роутера). Кадр передается в физическую среду (кабель, Wi-Fi).
7. Пакет путешествует по сети через маршрутизаторы (которые работают на сетевом уровне, анализируя IP-адреса), пока не достигнет сервера Яндекса.

8. На сервере происходит обратный процесс (**деинкапсуляция**): канальный уровень снимает заголовок кадра, сетевой – IP-заголовок, транспортный (TCP) собирает сегменты в исходный HTTP-запрос и передает его прикладному уровню (веб-серверу).
9. Веб-сервер обрабатывает запрос и отправляет ответ, повторяя весь путь в обратном направлении.

Итог: TCP/IP – это рабочая модель интернета. IP обеспечивает глобальную адресацию и доставку "из конца в конец", а TCP – надежность этой доставки для приложений.

4. Организации по стандартизации

Базовая идея: Чтобы сетевые устройства разных производителей по всему миру могли работать вместе, нужны единые, открытые стандарты. Их разрабатывают и поддерживают международные организации.

Смысл и пояснение:

- **ISO (International Organization for Standardization):** Разрабатывает стандарты в самых разных областях. В мире сетей ее ключевое достижение – **модель OSI** (Open Systems Interconnection) – эталонная семиуровневая модель для описания сетевых взаимодействий. Это как "идеальная карта мира" для сетевых инженеров.
- **IEEE (Institute of Electrical and Electronics Engineers):** Разрабатывает стандарты в области электротехники, радиоэлектроники и, что критично для нас, **локальных сетей (LAN)**. Самый известный стандарт – **IEEE 802.3 (Ethernet)** и **IEEE 802.11 (Wi-Fi)**. Они определяют, как устройства физически подключаются и общаются "на прямой видимости".
- **ISOC (Internet Society):** Некоммерческая организация, продвигающая открытое развитие и использование интернета.
- **IAB (Internet Architecture Board):** Архитектурный совет интернета, входящий в ISOC. Отвечает за общее видение архитектуры и долгосрочное развитие протоколов интернета.
- **IETF (Internet Engineering Task Force):** Это **практически самая важная для TCP/IP организация**. Инженерная рабочая группа, которая занимается разработкой и совершенствованием конкретных протоколов интернета (TCP, IP, HTTP, DNS и тысячи других). Стандарты IETF публикуются в виде документов RFC (Request for Comments). Это открытый процесс, в котором может участвовать любой специалист.

Аналогия: Представьте строительство железных дорог по всему миру.

- **ISO** создала идеальный план (OSI), как должны выглядеть пути, станции, расписания.
- **IEEE** установила стандарт на ширину колеи и конструкцию рельсов (Ethernet, Wi-Fi) для участков внутри одной страны (локальная сеть).
- **ISOC/IAB/IETF** – это международный консорциум, который договорился, как соединить все эти железные дороги между странами (интернет), создав правила перехода с одной колеи на

другую, единые билеты (IP) и систему надежной доставки грузов (TCP). IETF – это инженеры на местах, которые постоянно улучшают эти правила.

Итог: Стандартизация – краеугольный камень современных сетей. Благодаря работе этих организаций ваш ноутбук от Apple может по Wi-Fi (IEEE) подключиться к роутеру TP-Link и через интернет (по правилам IETF) получить данные с сервера на Linux.

5. Многоуровневые модели OSI и TCP/IP. Инкапсуляция данных. Протокольные блоки данных (PDU)

Базовая идея: Для упрощения проектирования и понимания сложного процесса сетевого взаимодействия используют многоуровневые (слоеные) модели. Каждый уровень решает свою конкретную задачу, предоставляя услуги уровню выше.

Модель OSI (7 уровней) – эталонная:

1. **Прикладной (Application):** Интерфейс для пользователя и приложений (HTTP, FTP, SMTP). *Пример: Почтовый клиент Outlook.*
2. **Представительский (Presentation):** Преобразование данных (кодирование, шифрование, сжатие). *Пример: Перекодировка текста из UTF-8 в ASCII.*
3. **Сеансовый (Session):** Управление сеансом связи (установка, поддержание, завершение диалога). *Пример: Восстановление прерванной закачки.*
4. **Транспортный (Transport):** Обеспечение надежной или ненадежной доставки "от конца к концу" (TCP, UDP). *Определяет, как разбивать данные.*
5. **Сетевой (Network):** Определение маршрута, логическая адресация (IP-адреса). *Работа маршрутизаторов.*
6. **Канальный (Data Link):** Организация доступа к среде, физическая адресация (MAC-адреса), обнаружение ошибок. *Работа коммутаторов.*
7. **Физический (Physical):** Передача битов (0 и 1) через физическую среду (кабель, радиоволны). *Уровни напряжения, разъемы.*

Модель TCP/IP (4 уровня) – практическая:

- **Прикладной (Application):** Объединяет уровни 5, 6, 7 модели OSI. Здесь работают HTTP, FTP, DNS, SMTP.
- **Транспортный (Transport):** Аналог 4-го уровня OSI (TCP, UDP).
- **Межсетевой (Internet):** Аналог 3-го уровня OSI (IP, ICMP).
- **Уровень сетевого доступа (Network Access):** Объединяет уровни 1 и 2 модели OSI (Ethernet, Wi-Fi, PPP).

Инкапсуляция данных – ключевой процесс:
Смысл: Это процесс упаковки данных при движении **вниз** по уровням модели от приложения к физической среде. Каждый уровень добавляет к данным, полученным от

уровня выше, свою служебную информацию (**заголовок**, а иногда и "конец" – трейлер). Это как матрешка или отправка ценного предмета в нескольких коробках.

Протокольный блок данных (PDU) – единица информации на каждом уровне:

- **Данные (Data)** – PDU прикладного уровня.
- **Сегмент (Segment – TCP) или Дейтаграмма (Datagram – UDP)** – PDU транспортного уровня (данные + TCP/UDP заголовок).
- **Пакет (Packet)** – PDU сетевого уровня (сегмент + IP-заголовок). *Часто термины "пакет" и "дейтаграмма" используют как синонимы.*
- **Кадр (Frame)** – PDU канального уровня (пакет + заголовок и трейлер канального уровня, например, Ethernet).
- **Биты (Bits)** – PDU физического уровня (кадр, преобразованный в сигнал).

Пример инкапсуляции (отправка email):

1. Вы пишете текст письма (**Данные**).
2. Транспортный уровень (TCP) добавляет свой заголовок (с портами), получается **TCP-сегмент**.
3. Сетевой уровень (IP) добавляет свой заголовок (с IP-адресами), получается **IP-пакет**.
4. Канальный уровень (Ethernet) добавляет заголовок (с MAC-адресами) и трейлер (для проверки ошибок), получается **Ethernet-кадр**.
5. Физический уровень превращает этот кадр в последовательность **битов** (электрические импульсы в кабеле).

На принимающей стороне происходит обратный процесс – **деинкапсуляция**: каждый уровень снимает свой заголовок и передает "внутренность" уровню выше, пока не дойдет до исходного сообщения.

Итог: Модели OSI и TCP/IP дают нам карту для понимания сетей. Инкапсуляция – это механизм, по которому данные, обрастая служебной информацией, путешествуют по этой карте от отправителя к получателю.

6. Доступ к локальным ресурсам. Сетевая адресация. MAC- и IP- адреса

Доступ к локальным ресурсам
Базовая идея: Это обмен данными между устройствами, находящимися в одной локальной сети (LAN) без участия маршрутизаторов. Устройства "видят" друг друга напрямую на канальном уровне.

Смысл и пример: Печать документа на сетевом принтере в одном офисе, передача файла между двумя компьютерами по общей папке, просмотр видео с NAS (сетевого хранилища) дома. Для этого используется **физическая (MAC) адресация**.

Сетевая адресация
Для однозначной идентификации устройств в сети используется система из двух основных типов адресов:

1. **MAC-адрес (Media Access Control) – Физический адрес, адрес канального уровня.**

- **Формат:** 48-битное число, обычно записывается в шестнадцатеричной системе, например, 00-1A-2B-3C-4D-5E. Производитель "прошивает" его в сетевую карту устройства при изготовлении. Он **глобально уникален** (в идеале).
- **Назначение:** Идентификация устройства **в пределах одного сегмента локальной сети**. Это "паспорт" сетевого интерфейса.
- **Аналогия:** Это серийный номер двигателя вашего автомобиля или уникальный отпечаток пальца. Он не меняется и используется для идентификации "железа" в непосредственном окружении.
- **Как работает доступ по MAC-адресу:** Когда компьютер А хочет отправить данные компьютеру Б в одной сети, он использует протокол **ARP (Address Resolution Protocol)**, чтобы выяснить MAC-адрес компьютера Б, зная его IP-адрес. Затем он создает кадр Ethernet, где в поле "Получатель" указывает MAC-адрес компьютера Б, и отправляет его "в эфир". Все устройства в сети "видят" этот кадр, но только устройство с совпадающим MAC-адресом примет и обработает его.

2. **IP-адрес (Internet Protocol) – Логический адрес, адрес сетевого уровня.**

- **Формат:** Для IPv4 – это 32-битное число, записываемое как четыре десятичных числа от 0 до 255, разделенных точками, например, 192.168.1.10. Назначается **вручную администратором или автоматически** (по DHCP) и зависит от сети, в которой находится устройство.
- **Назначение:** Идентификация устройства и его сетевого расположения в **глобальной составной сети (интернете или большой корпоративной сети)**. Это "почтовый адрес" устройства в сетевом городе.
- **Аналогия:** Это ваш домашний адрес (город, улица, дом, квартира). Если вы переедете в другой город (другую сеть), ваш адрес изменится. Адрес позволяет найти ваш дом (сеть) в мире, а номер квартиры (порт) – конкретного жильца (приложение) внутри.
- **Структура:** Состоит из двух частей: **Номер сети** (идентифицирует группу устройств) и **Номер узла** (идентифицирует конкретное устройство в этой сети). Маска подсети определяет границу между этими частями.

Сравнение MAC и IP:

- **MAC:** Физический, постоянный, работает на уровне 2, для локальной доставки.
- **IP:** Логический, изменяемый, работает на уровне 3, для глобальной маршрутировки.

Итог: Доступ к локальным ресурсам происходит по MAC-адресам. MAC – это "имя" устройства для соседей, IP – это "координаты" устройства в мире. Для связи в одной сети нужно знать MAC-адрес получателя, который находится через его IP-адрес с помощью ARP.

7. Доступ к удалённым ресурсам. Шлюз по умолчанию

Доступ к удалённым ресурсам
Базовая идея: Это обмен данными с устройствами, находящимися **в других сетях** (например, доступ к веб-сайту в интернете). Для этого необходима **маршрутизация** – процесс определения пути для пакетов через промежуточные сети.

Смысл: Ваш компьютер в домашней сети (192.168.1.10) хочет связаться с веб-сервером (77.88.55.55). Сравнивая свою сетевую часть IP-адреса с сетевой частью адреса сервера, компьютер понимает, что сервер находится в **другой сети**.

Шлюз по умолчанию (Default Gateway)
Базовая идея: Это IP-адрес маршрутизатора в локальной сети, который является **"выходными дверями"** из этой сети во внешний мир. Если устройству нужно отправить пакет в другую сеть, оно отправляет его не напрямую получателю (так как не знает пути), а на шлюз по умолчанию, доверяя ему дальнейшую маршрутизацию.

Смысл и аналогия: Представьте небольшой городок (ваша локальная сеть). Все жители знают друг друга в лицо (общение по MAC). Но чтобы отправить посылку в другой город (удаленная сеть), все несут ее на местную почту (шлюз по умолчанию). Почта знает, как связаться с почтами других городов и доставить посылку до адресата.

Как это работает (пошагово на примере):

1. Компьютер (192.168.1.10) хочет зайти на сайт yandex.ru (77.88.55.55).
2. С помощью DNS он узнает IP-адрес сервера.
3. Компьютер сравнивает свою сеть (допустим, 192.168.1.0/24) с сетью назначения. Они разные.
4. **Решение:** Отправить пакет не напрямую (неизвестно как), а на шлюз по умолчанию, который прописан в его настройках (например, 192.168.1.1 – адрес домашнего роутера).
5. Но чтобы отправить пакет на шлюз, нужен его **MAC-адрес** (общение в локальной сети идет по MAC)! Компьютер использует протокол **ARP**, чтобы узнать MAC-адрес для IP 192.168.1.1.
6. Компьютер инкапсулирует данные:
   - **IP-заголовок:** *Источник:* 192.168.1.10, *Назначение:* 77.88.55.55.
   - **Ethernet-заголовок:** *Источник:* MAC компьютера, *Назначение:* **MAC-адрес шлюза** (192.168.1.1).
7. Кадр отправляется на шлюз. Роутер (192.168.1.1) принимает его (MAC совпадает), деинкапсулирует до IP-пакета.
8. Роутер видит, что конечный IP-адрес (77.88.55.55) не из его локальной сети. Он смотрит в свою таблицу маршрутизации и находит следующий маршрутизатор у интернет-провайдера, которому нужно переслать пакет.
9. Роутер заново инкапсулирует IP-пакет в **новый кадр**:
   - **IP-заголовок остается прежним** (от 192.168.1.10 к 77.88.55.55) – это сквозная адресация "от конца к концу".
   - **Новый Ethernet-заголовок (или PPPoE и т.д.):** *Источник:* MAC интерфейса роутера к провайдеру, *Назначение:* MAC следующего маршрутизатора провайдера.

10. Пакет уходит в интернет, проходя через цепочку маршрутизаторов, пока не достигнет сети Яндекса и, наконец, целевого сервера.
11. Ответ от сервера проделывает обратный путь, также проходя через шлюз по умолчанию в вашей сети, чтобы попасть на ваш компьютер.

Итог: Шлюз по умолчанию – это критически важный элемент конфигурации любого сетевого устройства. Это "посредник", который позволяет устройствам в локальной сети выходить за ее пределы, обеспечивая связь с удаленными ресурсами по всему миру через механизмы маршрутизации на основе IP-адресов.

Заключение

Мы рассмотрели фундаментальные основы сетевых протоколов и коммуникаций. От преобразования информации в биты (кодирование) до сложного, многоуровневого процесса ее доставки через локальные и глобальные сети. Ключевые выводы:

1. Сетевые коммуникации строятся на **протоколах** – строгих правилах, которые обеспечивают взаимопонимание устройств.
2. Модели **OSI и TCP/IP** предоставляют нам карту для структурированного понимания этих процессов.
3. Принцип **инкапсуляции/деинкапсуляции** и **иерархии PDU** – это механизм "путешествия" данных по уровням этой модели.
4. Адресация двухуровневая: **MAC-адреса** для точной доставки "соседу" в одной сети, **IP-адреса** для глобальной маршрутизации "через сети".
5. **Доступ к локальным ресурсам** осуществляется напрямую по MAC-адресам.
6. **Доступ к удаленным ресурсам** требует наличия **шлюза по умолчанию** (маршрутизатора), который, используя IP-адресацию, перенаправляет трафик во внешние сети.

Этот теоретический фундамент является отправной точкой для дальнейшего углубленного изучения сетевых технологий, настройки оборудования и диагностики сетевых проблем.

Контрольные вопросы к ТЕМЕ-2

1. Что такое процесс кодирования сообщения в контексте компьютерных сетей, и какой универсальный вид принимает информация в результате этого процесса?
2. Какие три основных элемента определяет сетевой протокол? Приведите пример протокола и его действие.
3. Объясните ключевые различия в задачах протоколов IP и TCP в стеке TCP/IP, используя аналогию с отправкой письма.
4. Какая организация разработала эталонную модель OSI, а какая — практически значимые стандарты для локальных сетей (Ethernet, Wi-Fi)?

5. Какую практическую роль играет IETF (Internet Engineering Task Force) в развитии интернета?
6. Назовите и кратко охарактеризуйте четыре уровня модели TCP/IP.
7. Что такое инкапсуляция данных? Опишите, как меняется название единицы данных (PDU) при движении от прикладного уровня к физическому.
8. Для доступа к какому типу ресурсов (локальному или удаленному) в первую очередь используется MAC-адрес?
9. В чем состоит принципиальное отличие MAC-адреса от IP-адреса по способу назначения и области действия?
10. Что должен сделать компьютер в локальной сети, чтобы отправить данные устройству в другой сети, и почему он не может сделать это напрямую?
11. Что такое «шлюз по умолчанию» (Default Gateway) и какова его роль в сетевых коммуникациях?
12. Какой протокол используется для определения MAC-адреса устройства, когда известен его IP-адрес в пределах одной локальной сети?

Практические задания по ТЕМЕ-2

Задание 1. Проблема с печатью в локальном офисе

Вы начинающий системный администратор в небольшой российской компании «Северные Технологии» в Архангельске. В бухгалтерии стоит сетевой принтер. Сотрудница Мария Ивановна жалуется, что не может отправить документ на печать со своего компьютера. При этом её коллега за соседним столом печатает без проблем. Вы проверяете — сетевая папка с документами у Марии Ивановны открывается. Основываясь на изученном материале, предположите, на каком уровне сетевого взаимодействия (используя модель TCP/IP) скорее всего возникает проблема, и какой тип адресации (MAC или IP) в данный момент не используется корректно? Объясните свою логику.

Задание 2. Анализ сетевой конфигурации компьютера

В учебном компьютерном классе колледжа в Казани студент случайно изменил настройки сети. Теперь его компьютер не заходит на сайты в интернете, но при этом видит другие компьютеры в локальной сети колледжа и может передавать им файлы. Вы, как администратор, вызываете команду ipconfig /all и видите, что компьютер имеет IP-адрес 192.168.10.25, корректную маску подсети и MAC-адрес. Однако в строке «Основной шлюз» стоит значение 0.0.0.0. Используя теоретический материал, объясните, почему при такой конфигурации доступ в интернет отсутствует, а локальная связь работает. Что означает адрес 0.0.0.0 в данном контексте?

Задание 3. Маршрутизация «посылки» данных

Представьте, что вы объясняете принцип работы интернета новому сотруднику на аналогии с почтовой службой. Вам нужно описать путь электронного письма от сотрудника в московском офисе компании «Ромашка» (IP: 10.1.1.5) до партнёра во Владивостоке в компании «Лотос» (IP: 172.16.20.10).
Создайте пошаговое описание этого пути, используя следующие понятия: **инкапсуляция, IP-адрес (как адрес дома), MAC-адрес (как идентификатор автомобиля для перевозки), шлюз по умолчанию (местное почтовое отделение), маршрутизатор (сортировочный центр), деинкапсуляция**. Укажите, какой адрес (MAC или IP) меняется, а какой остаётся неизменным на всём пути.

Задание 4. Выбор стандарта для новой сети

Директор техникума в Екатеринбурге поручил вам спроектировать новую беспроводную сеть для библиотеки и приобрёл оборудование разных производителей: точки доступа от «Eltex», сетевые адаптеры для компьютеров от «D-Link» и «TP-Link». Он беспокоится, совместимо ли это оборудование между собой. Опираясь на изученный материал об организациях по стандартизации, назовите конкретную организацию и стандарт, которые гарантируют, что оборудование разных производителей сможет работать вместе в одной локальной беспроводной сети. Объясните, почему это возможно.

Задание 5. Диагностика по PDU

К вам, как к сетевому администратору, обратился пользователь из отдела кадров. Он говорит: «У меня не открывается наш корпоративный портал!». Вы начинаете диагностику. Представьте мысленный алгоритм ваших рассуждений, идя **сверху вниз по уровням модели TCP/IP**. Для каждого уровня сформулируйте один простой проверочный вопрос или действие, которое поможет локализовать проблему (например, на Прикладном уровне: «А открывается ли обычный сайт, например, yandex.ru?»). Используйте в рассуждениях термины: **прикладной уровень, транспортный уровень, сетевой уровень, уровень сетевого доступа**.

ТЕМА-3. Сетевые технологии Ethernet

Раздел 1: Введение в Ethernet. Основополагающие принципы

1.1. Семейство сетевых технологий Ethernet: эволюция и единство

Ethernet – это не единый стандарт, а целое семейство технологий для построения локальных вычислительных сетей (ЛВС). Его ключевая особенность – единство на фундаментальном уровне. Несмотря на то, что Ethernet эволюционировал от коаксиального кабеля со скоростью 10 Мбит/с до оптических линий со скоростью 400 Гбит/с и выше, его сердцевина осталась неизменной: **формат кадра (обрамление данных) и базовый метод доступа к среде (CSMA/CD, а затем его наследник в виде полнодуплексной работы)**.

- **Аналогия:** Представьте себе автомобиль. За столетие изменилось всё: бензиновый двигатель сменился электрическим, механические системы управления стали электронными, появились навигаторы и автопилоты. Но основные принципы – четыре колеса, руль, педали тормоза и газа, необходимость ездить по дорогам (среде) – остались. Так и Ethernet. Независимо от скорости и типа кабеля, устройства «понимают» друг друга благодаря единому «языку» – формату кадра.

1.2. Принцип работы Ethernet: Кадры, Пакеты и Адресация

Основная задача Ethernet – доставить блок данных (пакет от вышестоящего протокола, например, IP) от одного устройства в сети к другому. Ethernet делает это, упаковывая эти данные в свой собственный «конверт» – **кадр Ethernet**. Этот кадр содержит всю служебную информацию, необходимую для доставки в пределах одной локальной сети (сегмента).

- **Пример:** Вы хотите отправить письмо другу в другом городе. Вы пишете текст (полезные данные – пакет IP). Затем кладете его в конверт, на котором указываете **полный адрес получателя (MAC-адрес)** и **обратный адрес отправителя (ваш MAC-адрес)**. Это и есть формирование кадра. Почтовое отделение (сетевое оборудование) смотрит на адрес на конверте и решает, как его доставить.

1.3. Взаимодействие на подуровнях LLC и MAC

Для удобства проектирования сетей уровень работы с данными (Уровень 2 модели OSI) разделен на два подуровня:

- **LLC (Logical Link Control – Управление логической связью):** Это «интеллектуальный» подуровень. Его задача – определить, какому протоколу верхнего уровня (например, IP или

IPX) принадлежат данные внутри кадра. Он добавляет в кадр специальный идентификатор. Это позволяет одному сетевому интерфейсу работать с несколькими сетевыми протоколами одновременно.

- **MAC (Media Access Control – Управление доступом к среде):** Это «рабочий» подуровень. Он отвечает за физическую доставку кадра. В его ведении находятся:

- **MAC-адресация** (добавление адресов отправителя и получателя).
- **Формирование кадра** (обрамление данных служебными полями).
- **Управление доступом к общей среде** (кабелю) – алгоритм CSMA/CD для полудуплексных сред или контроль за полнодуплексной передачей.
- **Обнаружение ошибок** (создание и проверка контрольной суммы кадра).

- **Аналогия:** Представьте себе грузовую компанию (Уровень 2). В ней есть **диспетчер (LLC)**, который принимает заказ, определяет тип груза (хрупкий, скоропортящийся – аналог типа протокола) и передает задание водителю. **Водитель-грузчик (MAC)** берет задание, загружает груз в стандартный контейнер своего размера (кадр), наклеивает на него адресные бирки (MAC-адреса), проверяет крепления (контрольная сумма) и везет его по дорогам, соблюдая правила движения (CSMA/CD).

Раздел 2: Управление доступом и адресация в Ethernet

2.1. Управление доступом к среде передачи данных (CSMA/CD)

Исторически Ethernet использовал общую среду (один длинный коаксиальный кабель, к которому подключались все компьютеры). Чтобы устройства не передавали данные одновременно и не «перекрикивали» друг друга, был разработан метод **CSMA/CD (Carrier Sense Multiple Access with Collision Detection – множественный доступ с контролем несущей и обнаружением коллизий)**.

- **Принцип работы по шагам:**

1. **Прослушивание (Carrier Sense):** Устройство, желающее отправить данные, сначала «прислушивается» к кабелю. Если слышит, что кто-то уже передает (есть «несущая»), оно ждет.
2. **Передача:** Если в кабеле тишина, устройство начинает передачу своего кадра.
3. **Обнаружение столкновения (Collision Detection):** Во время передачи устройство продолжает «слушать» кабель. Если обнаруживается, что другой узел тоже начал передачу (сигналы наложились – произошла **коллизия**), оба устройства немедленно прекращают передачу.
4. **Задержка и повтор:** Каждое устройство выжидает случайный промежуток времени (чтобы снова не начать одновременно), после чего возвращается к шагу 1.

- **Аналогия:** Групповое обсуждение в комнате без ведущего (общая среда). Прежде чем говорить, человек слушает, не говорит ли кто-то еще (прослушивание). Если тихо, он начинает говорить. Если два человека начинают говорить одновременно, они слышат друг

друга (обнаружение коллизии), оба замолкают, выжидают паузу (случайную, чтобы не начать снова синхронно), и затем один из них пытается снова.

Важно: В современных сетях на витой паре и с использованием коммутаторов (switch) каждый порт – это отдельная среда, работающая в **полнодуплексном** режиме (передача и прием одновременно). Здесь CSMA/CD не нужен, так как коллизии исключены на физическом уровне.

2.2. MAC-адрес: идентификация Ethernet

MAC-адрес (Media Access Control address) – это уникальный физический адрес сетевого интерфейса (сетевой карты, Wi-модуля, порта коммутатора). Он «прошит» производителем оборудования и состоит из **48 бит (6 байтов)**, обычно записываемых в шестнадцатеричном формате, например: 00-1A-2B-3C-4D-5E или 00:1A:2B:3C:4D:5E.

- **Структура:**
  - Первые 3 байта (например, 00-1A-2B) – это **OUI (Organizationally Unique Identifier)**, идентификатор организации-производителя.
  - Последние 3 байта (3C-4D-5E) – серийный номер, назначаемый производителем.
- **Пример:** Это как уникальный серийный номер (VIN) автомобиля. Он присваивается на заводе и однозначно идентифицирует именно эту машину (сетевую карту) среди всех других в мире.

2.3. Атрибуты кадра Ethernet (формат кадра)

Кадр Ethernet – это стандартизированная структура. Основные поля (атрибуты):

1. **Преамбула и SFD:** Специальная битовая последовательность, которая «подготавливает» приемник, синхронизируя его тактовые сигналы. Как слово «Внимание!» перед объявлением.
2. **MAC-адрес получателя (Destination MAC):** Куда доставить кадр.
3. **MAC-адрес отправителя (Source MAC):** Откуда пришел кадр.
4. **Тип/Длина (EtherType/Length):** Указывает, какой протокол верхнего уровня (например, IPv4 – 0x0800) инкапсулирован в данных кадра.
5. **Данные (Payload):** Собственно передаваемая информация (пакет IP). Имеет минимальный и максимальный размер (46-1500 байт для классического Ethernet).
6. **Контрольная последовательность кадра (FCS):** Циклическая контрольная сумма (CRC), вычисленная на основе всего кадра. Принимающая сторона вычисляет свою сумму и сравнивает с этой. Если не совпадает – кадр поврежден и отбрасывается. Это как опечатанный конверт с защитой от вскрытия.

2.4. Типы адресации (рассылки) в Ethernet

У MAC-адреса получателя в кадре может быть три формы:

- **Одноадресная (Unicast):** Кадр предназначен **одному конкретному** устройству. MAC-адрес получателя – это уникальный физический адрес целевой сетевой карты. Пример: 00-1A-2B-3C-4D-5E.
- **Групповая (Multicast):** Кадр предназначен **группе** устройств, подписанных на определенный сервис (например, видеопоток или обновление от сервера). MAC-адрес начинается со специального байта, например, 01-00-5E-xx-xx-xx для IP-мультикаста. Это как адрес на конверте: «Всем жильцам подъезда №3».
- **Широковещательная (Broadcast):** Кадр предназначен **всем без исключения** устройствам в локальном сегменте сети. MAC-адрес получателя – все единицы: FF-FF-FF-FF-FF-FF. Это как громкоговоритель, объявляющий: «Внимание, всем!».

2.5. Сквозное подключение, MAC- и IP-адреса

В процессе передачи данных через Интернет или крупную сеть работают **две системы адресации**:

- **IP-адрес (Уровень 3):** Логический, иерархический адрес. Он определяет **место устройства в глобальной сети** и может меняться. Это как почтовый адрес: «г. Москва, ул. Тверская, д. 7, кв. 10».
- **MAC-адрес (Уровень 2):** Физический, плоский адрес. Он определяет **конкретное устройство в локальном сегменте сети** (в пределах одного коммутатора/маршрутизатора). Это как фамилия и имя человека, живущего в этой квартире.
- **Пример сквозного пути (End-to-End):** Ваш компьютер (IP: 192.168.1.10, MAC: AA-AA-AA-AA-AA-AA) хочет отправить данные на сервер в Интернете (IP: 8.8.8.8). На каждом **локальном этапе** (хопе) используется MAC-адресация.

1. Ваш компьютер определяет, что сервер 8.8.8.8 находится в другой сети. Поэтому он отправляет кадр не напрямую, а **шлюзу по умолчанию** (вашему домашнему роутеру, IP: 192.168.1.1).
2. Он упаковывает IP-пакет в кадр Ethernet, где:

- **MAC-адрес получателя:** MAC-адрес роутера (например, BB-BB-BB-BB-BB-BB).
- **MAC-адрес отправителя:** свой (AA-AA-AA-AA-AA-AA).

3. Роутер принимает кадр, отбрасывает оболочку Ethernet (Уровень 2), смотрит на IP-адрес (8.8.8.8) и решает, куда его переслать дальше по своему маршруту. На следующем интерфейсе он снова упакует пакет в новый кадр, но уже с **другими MAC-адресами** (своего порта и следующего маршрутизатора).

Вывод: IP-адреса работают от отправителя до получателя **сквозным образом**, а MAC-адреса – только на **каждом отдельном канальном сегменте**.

Раздел 3: Протокол ARP – связующее звено между уровнями

3.1. Принципы работы ARP (Address Resolution Protocol)

Возникает ключевой вопрос: как устройство узнает **MAC-адрес** того самого шлюза или соседа по сети, если ему известен только его **IP-адрес**? Для этого и существует **ARP**.

ARP – это протокол, который разрешает (соотносит) известный IP-адрес в неизвестный MAC-адрес в пределах одной локальной сети.

- **Алгоритм работы:**

1. Устройство А (IP: 192.168.1.10) хочет отправить данные устройству Б (IP: 192.168.1.20), но не знает его MAC-адрес.
2. Устройство А создает и отправляет **широковещательный ARP-запрос** (ARP Request) на адрес FF-FF-FF-FF-FF-FF. В этом запросе содержится вопрос: «У кого IP-адрес 192.168.1.20? Сообщите свой MAC-адрес устройству 192.168.1.10!». Этот запрос получают **все** устройства в локальной сети.
3. Устройство Б, увидев свой IP-адрес в запросе, отправляет **одноадресный ARP-ответ** (ARP Reply) напрямую устройству А. В ответе указано: «IP-адрес 192.168.1.20 принадлежит MAC-адресу CC-CC-CC-CC-CC-CC».
4. Устройство А получает ответ и заносит пару **IP->MAC** в свою **ARP-таблицу (ARP cache)**. Теперь оно знает, как доставить кадр.

- **Аналогия:** Вы пришли в большой офис (локальная сеть) и вам нужно найти сотрудника Иванова (IP-адрес). Вы не знаете, в каком он кабинете (MAC-адрес). Вы выходите в коридор и громко спрашиваете: «Иванов здесь? Отзовитесь!» (широковещательный ARP-запрос). Все слышат, но только Иванов откликается: «Я здесь, в кабинете 305!» (ARP-ответ). Вы запоминаете: Иванов -> кабинет 305, и идете туда.

3.2. Таблицы ARP на сетевых устройствах

Каждое устройство с TCP/IP стеком (компьютер, сервер, маршрутизатор) ведет собственную **ARP-таблицу**. Это временная база данных, хранящая соответствия IP- и MAC-адресов. Записи в ней имеют время жизни (обычно 2-20 минут), после чего удаляются, чтобы обеспечить актуальность при смене оборудования.

- **Пример таблицы ARP на ПК (команда arp -a в Windows/Linux):**

```
Интернет-адрес      Физический адрес       Тип
192.168.1.1         bb-bb-bb-bb-bb-bb      динамический
192.168.1.20        cc-cc-cc-cc-cc-cc      динамический
```

3.3. Основные недостатки протокола ARP

1. **Нагрузка на среду передачи данных:** Каждый ARP-запрос – это широковещательный кадр. В больших и активных сетях множество устройств могут одновременно инициировать ARP-запросы, создавая излишний широковещательный трафик (так называемый «broadcast storm» – широковещательная буря), который загружает все порты коммутаторов и процессоры всех узлов.
2. **Проблемы безопасности – ARP-спуфинг (ARP Poisoning):** Протокол ARP не имеет механизмов аутентификации. Устройство безоговорочно доверяет полученному ARP-ответу. Злоумышленник может отправлять поддельные (spoofed) ARP-ответы, заявляя: «IP-адрес шлюза (192.168.1.1) – это мой MAC-адрес DD-DD-DD-DD-DD-DD». Жертва обновит свою ARP-таблицу и начнет отправлять весь свой интернет-трафик злоумышленнику, который сможет его перехватывать (атака «человек посередине» – Man-in-the-Middle). Борьба с этим требует дополнительных мер (статические ARP-записи, защитные механизмы на коммутаторах).

Раздел 4: Коммутаторы Ethernet – основа современной ЛВС

4.1. Основная информация о портах коммутатора. Функция Auto-MDIX

Коммутатор (switch) – это интеллектуальное устройство Уровня 2, которое принимает кадры Ethernet и **пересылает их только на тот порт, к которому подключен получатель**, а не на все, как это делал концентратор (hub).

- **Порты коммутатора** характеризуются:
  - **Скоростью** (100 Мбит/с, 1 Гбит/с, 10 Гбит/с).
  - **Режимом дуплекса** (полудуплекс – устарел, полнодуплекс – одновременная приём/передача).
  - **Автосогласованием (Auto-negotiation):** Порт автоматически договаривается с подключенным устройством о максимально возможной скорости и режиме дуплекса.
  - **Auto-MDIX (Automatic Medium-Dependent Interface Crossover):** Умная функция, которая **автоматически определяет тип кабеля** (прямой или кроссоверный) и настраивает передающую и приемную пары соответствующим образом. Благодаря этому для подключения любого устройства (ПК к коммутатору, коммутатора к коммутатору) можно использовать **прямой кабель**, что исключает ошибки.

4.2. Таблица MAC-адресов коммутатора. Процесс коммутации

«Интеллект» коммутатора основан на **таблице MAC-адресов (CAM-таблица – Content Addressable Memory)**. В этой таблице хранятся соответствия: **MAC-адрес устройства -> номер порта коммутатора**.

- **Принцип самообучения коммутатора:**

1. **Изучение (Learning):** Когда коммутатор получает кадр на каком-либо порту, он смотрит на **исходный MAC-адрес (Source MAC)** этого кадра и записывает в таблицу: «MAC-адрес AA-AA-AA-AA-AA-AA находится на порту 1». Так он постепенно изучает расположение всех устройств в сети.
2. **Пересылка (Forwarding):** Когда коммутатор получает кадр, он смотрит на **MAC-адрес получателя (Destination MAC)**.

- Если адрес найден в таблице, кадр пересылается **только на указанный порт** (одноадресная передача).
- Если адрес не найден (неизвестный одноадресный) или это широковещательный/мультикаст-адрес, кадр **заливается (flooding)** на все порты, кроме того, с которого он пришел.

3. **Фильтрация (Filtering):** Если адрес получателя находится на том же порту, откуда пришел кадр (отправитель и получатель в одном сегменте), коммутатор **отбрасывает (фильтрует)** этот кадр, не пересылая его дальше. Это изолирует ненужный трафик.

- **Пример:** В офисе три ПК: А (порт 1), Б (порт 2), В (порт 3).

1. ПК А отправляет кадр ПК Б. Коммутатор запоминает MAC-А -> порт 1. Так как MAC-В ему неизвестен, он заливает кадр на порты 2 и 3.
2. ПК Б отвечает ПК А. Коммутатор запоминает MAC-В -> порт 2. Теперь он знает, где находится MAC-А (порт 1), и отправляет кадр **только на порт 1**.
3. В дальнейшем весь трафик между А и Б будет идти напрямую через коммутатор, не затрагивая порт ПК В. Это повышает безопасность и производительность.

4.3. Конфигурации коммутаторов: фиксированная и модульная

- **Фиксированная конфигурация:** Коммутатор представляет собой моноблок с заранее заданным неизменным количеством и типом портов (например, 24 гигабитных медных порта и 4 SFP-порта для оптики). Это наиболее распространенный и экономичный вариант для доступа к конечным пользователям (компьютерам, IP-телефонам).
- **Модульная конфигурация (шасси):** Состоит из шасси (корпуса) с управляющим модулем и слотами для установки различных линейных карт (модулей). Позволяет гибко наращивать количество и тип портов (медные, оптические, разных скоростей) под нужды сети. Используется в ядре и распределении крупных сетей.

4.4. Сравнение коммутации уровня 2 и уровня 3

- **Коммутатор Уровня 2 (L2 Switch):** Принимает решения **исключительно на основе MAC-адресов**. Он не «понимает» IP-адреса. Его задача – эффективная доставка кадров внутри одной IP-подсети (одного широковещательного домена). Он создает множество изолированных доменов коллизий, но все устройства находятся в одном широковещательном домене.
- **Маршрутизатор (Router, Устройство Уровня 3):** Принимает решения **на основе IP-адресов**. Он соединяет **разные IP-подсети (разные широковещательные домены)**, фильтруя широковещательный трафик и определяя маршруты между сетями.
- **Коммутатор Уровня 3 (L3 Switch):** Это гибрид. В его основе – высокопроизводительный L2-коммутатор, в который добавлен **аппаратный модуль маршрутизации**. Он может:

- Маршрутизировать трафик между VLAN (разными подсетями) **на скорости коммутации**, что гораздо быстрее, чем у традиционных маршрутизаторов.
- Осуществлять обычную L2-коммутацию внутри одной VLAN.
- Это устройство для ядра корпоративной сети, где требуется высокая скорость передачи между множеством подсетей.

4.5. Виртуальный интерфейс коммутатора (SVI), Маршрутируемый порт, EtherChannel уровня 3

- **SVI (Switched Virtual Interface – Виртуальный интерфейс коммутатора):** Это не физический, а **логический интерфейс**, связанный с конкретной VLAN. На L3-коммутаторе SVI выступает в роли **шлюза по умолчанию** для всех устройств в этой VLAN. Например, создав VLAN 10 и назначив SVI VLAN 10 IP-адрес 192.168.10.1, вы дадите всем ПК в этой VLAN точку входа в другие сети.
- **Аналогия:** SVI – это «виртуальный диспетчерский пункт» на этаже (VLAN). Все сотрудники этажа (устройства VLAN) знают, что чтобы выйти за его пределы, нужно обратиться к диспетчеру (SVI).
- **Маршрутизируемый порт (Routed Port):** Физический порт L3-коммутатора, который **настроен как порт маршрутизатора**. Он не принадлежит ни одной VLAN, не работает с тегами 802.1Q. На него назначается IP-адрес напрямую. Используется для подключения к другим маршрутизаторам или L3-устройствам (например, порт для связи с провайдером или между двумя маршрутизаторами в ядре сети).
- **Конфигурация маршрутизируемого порта (на примере CLI Cisco):**

```
interface GigabitEthernet0/1
  no switchport   // переводит порт в режим маршрутизации (L3)
  ip address 10.0.0.1 255.255.255.252 // Назначение IP-адреса
  no shutdown
```

- **EtherChannel уровня 3:** EtherChannel – это технология агрегации нескольких **физических** каналов Ethernet в один **логический** для увеличения пропускной способности и обеспечения отказоустойчивости. Обычно это L2-технология. Но L3-коммутатор может создать **EtherChannel уровня 3**, где логический порт (Port-channel) будет являться маршрутизируемым. На этот порт-channel назначается IP-адрес, и трафик между сетями распределяется по всем физическим линкам в канале с балансировкой нагрузки.

Заключение

Изученный материал охватывает фундаментальные основы семейства технологий Ethernet, которые являются краеугольным камнем современных локальных сетей. От понимания принципа работы CSMA/CD и формата кадра, через осознание роли MAC-адресов и протокола ARP как «переводчика» между сетевыми уровнями, к пониманию работы интеллектуального ядра сети – коммутаторов, их эволюции от L2 к L3-функционалу. Эти знания позволяют не только конфигурировать сетевое оборудование, но и анализировать

процессы передачи данных, выявлять и устранять базовые неисправности в сети, а также проектировать эффективные и безопасные сетевые сегменты.

Контрольные вопросы к ТЕМЕ-3

1. Что является ключевым неизменным элементом всех технологий семейства Ethernet, несмотря на эволюцию скоростей и сред передачи?
2. Для чего нужен кадр Ethernet и какие два основных типа адресов он содержит?
3. Какие задачи решают подуровни LLC и MAC канального уровня модели OSI?
4. Опишите своими словами принцип работы метода доступа CSMA/CD. В каких условиях он применяется в современных сетях?
5. Что такое MAC-адрес, какова его структура и в чем его принципиальное отличие от IP-адреса?
6. Назовите три основных типа адресации (рассылки) в Ethernet по MAC-адресу получателя. Как выглядит широковещательный MAC-адрес?
7. Объясните на примере, почему при передаче данных через сеть IP-адрес считается сквозным (end-to-end), а MAC-адрес – локальным (hop-to-hop).
8. Какую основную задачу решает протокол ARP и как он это делает (опишите шаги)?
9. Каковы два основных недостатка протокола ARP?
10. Как коммутатор Ethernet узнает, на какой его порт подключено устройство с определенным MAC-адресом (опишите процесс самообучения)?
11. Что такое функция Auto-MDIX на порту коммутатора и какую проблему пользователя она решает?
12. В чем главное функциональное отличие коммутатора уровня 2 (L2) от коммутатора уровня 3 (L3)?

Практические задания по ТЕМЕ-3

Задание 1: Анализ локального трафика

Вы начинающий сетевой администратор в колледже в г. Казани. Преподаватель жалуется, что при запуске видеолекции с локального медиасервера в его аудитории (все ПК в одной сети) видео у всех студентов подгружается с задержкой, хотя интернет-канал свободен. Вы запускаете анализатор трафика (например, Wireshark) на своем ноутбуке и видите огромное количество кадров с MAC-адресом получателя FF:FF:FF:FF:FF:FF, в данных которых постоянно повторяется вопрос об одном и том же IP-адресе.

- **Вопрос:** Какой сетевой процесс вызывает такую проблему? Объясните, почему эти кадры загружают сеть и мешают передаче видео. Как можно оперативно решить проблему (предложите одно простое действие на целевом устройстве)?

Задание 2: Неисправность межподсетного обмена

В филиале компании в Новосибирске настроена сеть с двумя отделами: Бухгалтерия (VLAN 10, адреса 192.168.10.0/24) и Отдел продаж (VLAN 20, адреса 192.168.20.0/24). Их шлюзом

по умолчанию является коммутатор L3, у которого созданы SVI с адресами 192.168.10.1 и 192.168.20.1. Компьютер бухгалтера (192.168.10.5) не может «пропинговать» компьютер менеджера (192.168.20.15). При этом внутри своих VLAN все пингуются. Вы проверяете ARP-таблицу на компьютере бухгалтера командой `arp -a` и не видите там записи для IP-адреса 192.168.20.15.

- **Вопрос:** Почему в ARP-таблице компьютера бухгалтера не может появиться запись для IP-адреса 192.168.20.15, даже если они активно пытаются обмениваться данными? Чей MAC-адрес должен быть в этой таблице для успешной отправки пакета в другую подсеть и почему?

Задание 3: Выбор оборудования для проекта

Вы помогаете модернизировать сеть небольшой гостиницы в Сочи. Владелец хочет, чтобы Wi-Fi для гостей был полностью изолирован от сети администрации и системы управления номерами. Вам нужно выбрать тип сетевого устройства для ядра этой сети, к которому будут подключены: 1) маршрутизатор провайдера, 2) коммутатор гостевого Wi-Fi, 3) коммутатор служебных помещений. Предложенный поставщиком обычный L2-коммутатор не подходит.

- **Вопрос:** Устройство какого типа (L2-коммутатор, маршрутизатор или L3-коммутатор) необходимо и почему именно оно требуется для решения задачи изоляции и маршрутизации между сетями гостей и администрации? Объясните кратко принцип.

Задание 4: Диагностика по таблице MAC-адресов

К вам обратился коллега из техникума в Екатеринбурге. Он смотрит на таблицу MAC-адреров своего коммутатора и видит странную запись: один и тот же MAC-адрес студентского ноутбука одновременно «висит» на порту 3 и порту 7. При этом студент физически находится в одном кабинете. Коллега уверен, что это признак атаки или неисправности.

- **Вопрос:** Предложите самое простое и вероятное объяснение этой ситуации, не связанное со злым умыслом. На что, скорее всего, не обратил внимания ваш коллега при первоначальной настройке сети? (Используйте знания о функциях современных коммутаторов).

Задание 5: Конфигурация порта для подключения к провайдеру

Вы настраиваете новый L3-коммутатор в офисе компании в Калининграде. От интернет-провайдера вам пришел инженер и протянул кабель, сказав, что для подключения выделен адрес 195.10.20.33/30, а шлюз провайдера — 195.10.20.34. Вам нужно настроить 24-й порт коммутатора для этого подключения.

- **Вопрос:** Какую принципиально важную команду (в стиле CLI Cisco) вы должны ввести в конфигурации этого порта *в первую очередь*, чтобы он стал маршрутизируемым, а затем назначить ему IP-адрес? Объясните, почему нельзя просто назначить IP-адрес на порт коммутатора без этой команды (чем такой порт по умолчанию отличается от порта маршрутизатора)?

ТЕМА-4. Сетевой уровень

Введение

Представьте себе мировую почтовую систему. Вы пишете письмо в одном городе, а получатель живёт в другом, возможно, на другом континенте. Чтобы письмо дошло, на конверте вы указываете конкретный адрес: страну, город, улицу, дом и квартиру. Само письмо может путешествовать разными путями: сначала его забирает почтальон, затем оно едет на сортировочный узел, потом – в аэропорт, летит на самолёте, снова проходит сортировку в стране назначения и, наконец, доставляется местным почтальоном до нужной квартиры. При этом само содержимое письма (текст) остаётся неизменным и защищённым конвертом.

Компьютерные сети работают по схожему принципу. **Сетевой уровень** в этой аналогии – это та самая система логической адресации и маршрутизации, которая обеспечивает доставку «конверта с данными» (пакета) от отправителя к получателю через множество промежуточных пунктов (маршрутизаторов), даже если они находятся в совершенно разных сетях. Это уровень, который отвечает на вопрос: «Как найти путь от одной сети к другой?».

Если канальный уровень (уровень 2 модели OSI) работает в пределах одной «деревни» или «улицы» (одной локальной сети), используя физические MAC-адреса, то сетевой уровень (уровень 3) работает в масштабах «страны» или «мира» (интернета), используя логические IP-адреса. Именно он позволяет объединить множество отдельных сетей в единое целое – интернет.

1. Сетевой уровень в процессе передачи данных

Основная идея: Сетевой уровень отвечает за логическую адресацию и сквозную доставку пакетов между устройствами в разных сетях, определяя оптимальный путь их следования.

Пояснение смысла: Представьте, что вам нужно отправить посылку из Москвы во Владивосток. Вы не знаете точного маршрута, да вам это и не нужно. Вы просто отдаёте посылку в транспортную компанию, указав адрес назначения. Далее на каждом логистическом узле (в Москве, потом, например, в Новосибирске, затем в Хабаровске) специалисты смотрят на конечный адрес и решают, куда отправить посылку дальше, чтобы она приблизилась к цели. Каждый такой узел принимает решение локально, основываясь на своей карте маршрутов (таблице маршрутизации). Сетевой уровень в компьютерных сетях выполняет ту же функцию: каждый маршрутизатор – это логистический узел, который, получив пакет, смотрит на его IP-адрес назначения и решает, в какой «следующий город» (на какой свой интерфейс или соседний маршрутизатор) его отправить.

Процесс передачи данных на сетевом уровне:

1. **Инкапсуляция:** Когда приложение (например, браузер) хочет отправить данные на удалённый сервер, транспортный уровень (например, TCP) делит их на сегменты. Сетевой уровень берёт такой сегмент, помещает его в свой «конверт» – **IP-пакет**, добавляя служебную информацию, главными полями которой являются **IP-адрес источника** (ваш адрес) и **IP-адрес назначения** (адрес сервера). Этот процесс упаковки называется инкапсуляцией.
2. **Маршрутизация:** Сформированный пакет поступает на первый маршрутизатор (часто это ваш домашний роутер). Маршрутизатор снимает информацию канального уровня (фрейм), анализирует IP-адрес назначения. Если адрес находится не в directly connected сети (не в «своём районе»), маршрутизатор ищет в своей таблице маршрутизации, куда дальше отправить пакет. Он выбирает «следующий прыжок» (next-hop) – адрес соседнего маршрутизатора, который, как считается, знает путь к цели лучше.
3. **Транзит:** Процесс повторяется на каждом следующем маршрутизаторе. Каждый «перевалочный пункт» принимает независимое решение о дальнейшем пути. Это децентрализованная система.
4. **Доставка:** Когда пакет, наконец, прибывает в сеть назначения, последний маршрутизатор определяет, что устройство с нужным IP находится в его непосредственно подключённой сети, и отправляет пакет ему напрямую, используя уже канальный уровень (MAC-адрес).
5. **Деинкапсуляция:** Получатель на своём сетевом уровне получает IP-пакет, проверяет его целостность, снимает служебную информацию сетевого уровня (IP-заголовок) и передаёт «внутренность» (транспортный сегмент) на вышестоящий уровень для дальнейшей обработки.

Ключевая аналогия: Почта vs. Курьерская служба.

- **Почта (соединение без установления соединения, datagram):** Вы бросаете открытку в почтовый ящик. Каждое почтовое отделение на пути обрабатывает её независимо. Открытки одного письма могут пойти разными путями и прийти в разное время. Это аналогия протокола **IP** – он не гарантирует доставку, порядок или целостность.
- **Курьерская служба (логическое соединение):** Вы звоните, договариваетесь, курьер приезжает, забирает посылку и несёт ответственность за её доставку по определённому маршруту. Это задача **транспортного уровня (TCP)**, который работает поверх IP и добавляет гарантии. Сетевой уровень (IP) – это сама почтовая инфраструктура, по которой едут и курьеры, и открытки.

2. Протоколы сетевого уровня

Основная идея: Протокол сетевого уровня – это набор правил и форматов данных, которые определяют, как устройства должны адресовать пакеты и каким образом они должны маршрутизироваться по составной сети.

Пояснение смысла: Чтобы все логистические узлы (маршрутизаторы) в мире понимали друг друга, они должны использовать единый «язык» и стандарты для адресов и обработки накладных. Таким «языком» интернета является семейство протоколов **TCP/IP**. Непосредственно на сетевом уровне ключевыми протоколами этого семейства являются:

1. **IP (Internet Protocol) – Интернет Протокол.** Это фундаментальный, базовый протокол. Его задача – доставить пакет от источника к получателю «по возможности» (best-effort delivery). Он не устанавливает соединение заранее и не гарантирует, что пакет дойдёт, дойдёт целым или в порядке отправки. Его можно сравнить с почтовой службой для открыток. Существует две основные версии: **IPv4** и **IPv6**.
2. **ICMP (Internet Control Message Protocol) – Протокол межсетевых управляющих сообщений.** Это «сигнальный» протокол, «помощник» IP. Он используется для отправки служебных сообщений о ошибках и операционной информации. Например, если маршрутизатор не может доставить IP-пакет (сеть недостижима), он отправляет обратно источнику пакет **ICMP с сообщением «Destination Unreachable»**. Утилита ping использует ICMP-пакеты «Echo Request» и «Echo Reply» для проверки доступности узла.
3. **ARP (Address Resolution Protocol) – Протокол разрешения адресов.** Хотя технически он работает на стыке сетевого и канального уровней, его функция критически важна для сетевого уровня. ARP преобразует **IP-адрес** (логический, уровень 3) в **MAC-адрес** (физический, уровень 2) в пределах одной локальной сети. Это как узнать, какой конкретно почтовый ящик (MAC) стоит по указанному адресу дома (IP) на вашей улице (сети).
4. **Протоколы маршрутизации (RIP, OSPF, EIGRP, BGP).** Это «диспетчерские» протоколы, которые работают на маршрутизаторах. Их задача – не передавать пользовательские данные, а **обмениваться информацией между маршрутизаторами**, чтобы каждый из них мог построить свою **таблицу маршрутизации** – карту путей по сети. RIP и OSPF常用于内部них сетей предприятия (IGP), а BGP – для обмена маршрутами между разными интернет-провайдерами (EGP).

3. Основные характеристики IP-протокола

Основная идея: IP – это простой, но мощный протокол, основанный на идее дейтаграммной, не гарантирующей доставки передачи данных с глобальной логической адресацией.

Пояснение смысла и характеристики:

1. **Дейтаграммный режим (Connectionless):** Каждый IP-пакет путешествует по сети независимо от других, даже если они часть одного большого сообщения. Маршрутизаторы обрабатывают каждый пакет отдельно, на основе текущей информации в таблице маршрутизации. Нет предварительного «звонка» для установления пути. **Аналогия:** Каждая открытка в серии отправляется отдельно; почта не резервирует для вас цельный путь.
2. **Не гарантирует доставку (Unreliable):** IP не предусматривает подтверждений получения пакета. Он делает всё возможное для доставки (best effort), но если на маршрутизаторе

переполнены буферы, если истёк срок жизни пакета (TTL), если произошёл сбой, пакет будет просто отброшен без уведомления отправителя (хотя для уведомления может быть отправлен ICMP-пакет). Гарантии обеспечивают вышележащие уровни (например, TCP).

3. **Отсутствие контроля порядка (No Sequencing):** Пакеты могут прибывать к получателю не в том порядке, в котором были отправлены, так как они могли пойти разными маршрутами. **Пример:** Вы отправляете три пакета: A, B, C. Из-за изменения маршрутов они могут прийти как C, A, B. Сборку в правильном порядке обеспечивает транспортный уровень.
4. **Глобальная логическая адресация:** IP вводит универсальную систему адресов, не зависящую от физического оборудования (в отличие от MAC-адресов). Каждое сетевое устройство, участвующее в IP-коммуникации, должно иметь уникальный (в пределах своей сети) IP-адрес. Это как наличие почтового адреса у каждого дома в мире, который понимают все почтовые службы.
5. **Фрагментация и повторная сборка:** Разные сети на пути следования пакета могут иметь разные ограничения на максимальный размер передаваемого блока данных (MTU). Если IP-пакет больше, чем MTU следующей сети, маршрутизатор делит его на несколько меньших фрагментов. Эти фрагменты путешествуют самостоятельно и собираются в исходный пакет только на конечном получателе. **Аналогия:** Большую картину нельзя пронести через узкую дверь, поэтому её разбирают на несколько частей, каждую проносят отдельно, а собирают уже в комнате.

4. Структура пакетов IPv4 и IPv6

4.1. Структура пакета IPv4

Основная идея: Заголовок IPv4 – это стандартизированный «конверт» размером минимум 20 байт, содержащий всю необходимую информацию для доставки и обработки пакета.

Пояснение смысла и поля (с примерами):

Представьте бумажный бланк для отправки посылки. В нём есть обязательные графы.

- **Версия (Version, 4 бита):** Указывает версию IP. Для IPv4 здесь всегда стоит число 4. Это как пометка «Бланк формы №4».
- **Длина заголовка (IHL, 4 бита):** Указывает размер самого заголовка в 32-битных словах. Минимальное значение – 5 (5 * 4 байта = 20 байт). Нужно, чтобы принимающая сторона знала, где заканчивается служебная информация и начинаются данные.
- **Тип обслуживания (ToS/DSCP, 8 бит):** Поле для указания приоритета пакета. Например, пакеты голосовой связи (VoIP) можно пометить как более приоритетные, чем пакеты загрузки файла, чтобы маршрутизаторы в случае перегрузки обрабатывали их в первую очередь. Это как пометка «СРОЧНО» или «ОСТОРОЖНО, ХРУПКОЕ» на коробке.
- **Общая длина (Total Length, 16 бит):** Длина всего пакета (заголовок + данные) в байтах. Максимум – 65535 байт. По этому полю устройство понимает, где заканчивается пакет.

- **Идентификатор (Identification, 16 бит):** Уникальный номер, присваиваемый отправителем. Все фрагменты одного исходного пакета будут иметь одинаковый идентификатор, чтобы получатель мог понять, что эти куски принадлежат вместе. Это как номер накладной, который ставится на все коробки одной большой посылки.
- **Флаги (Flags, 3 бита):** Управляют фрагментацией. Например, один из флагов (MF – More Fragments) говорит: «1 – за этим фрагментом последуют другие, 0 – это последний фрагмент». Другой флаг (DF – Don't Fragment) запрещает маршрутизаторам фрагментировать пакет: если он не проходит, его отбрасывают и отправляют ICMP-ошибку.
- **Смещение фрагмента (Fragment Offset, 13 бит):** Указывает позицию данного фрагмента в исходном неделённом пакете. Измеряется в блоках по 8 байт. Позволяет собрать фрагменты в правильном порядке, даже если они пришли не по очереди.
- **Время жизни (TTL, Time to Live, 8 бит): Критически важное поле.** Изначально задаётся отправителем (часто 64, 128, 255). Каждый маршрутизатор на пути уменьшает это значение на 1. Если TTL достигнет нуля, пакет уничтожается, и отправителю может прийти ICMP-сообщение «Time Exceeded». Это защищает сеть от вечной циркуляции «заблудившихся» пакетов. **Аналогия:** У вас есть 100 условных единиц «топлива» на поездку. Каждый логистический центр забирает 1 единицу. Если топливо закончилось, посылка дальше не идёт.
- **Протокол (Protocol, 8 бит):** Код, указывающий, данные какого протокола вышележащего уровня находятся «внутри» пакета. Например: 6 – TCP, 17 – UDP, 1 – ICMP. Это как пометка «Внутри: документы» или «Внутри: образцы». Получатель, увидев 6, передаст распакованные данные модулю TCP.
- **Контрольная сумма заголовка (Header Checksum, 16 бит):** Проверяет целостность только заголовка (не данных!). Рассчитывается отправителем, пересчитывается каждым маршрутизатором (так как меняется TTL). Если сумма не совпадает, пакет считается повреждённым и отбрасывается.
- **IP-адрес источника (Source IP Address, 32 бита):** Логический адрес отправителя.
- **IP-адрес назначения (Destination IP Address, 32 бита):** Логический адрес получателя.
- **Опции (Options, переменная длина):** Дополнительные, редко используемые параметры. Из-за переменной длины заголовка усложняется обработка.

4.2. Структура пакета IPv6

Основная идея: Заголовок IPv6 радикально упрощён (фиксированный размер 40 байт) для ускорения обработки маршрутизаторами, а также включает встроенные современные функции.

Пояснение смысла и ключевые отличия от IPv4:

Заголовок IPv6 – это как новый, оптимизированный электронный бланк с продуманными полями.

- **Фиксированный размер 40 байт:** Нет поля «длина заголовка», нет опций в основном заголовке. Это позволяет маршрутизаторам обрабатывать пакеты быстрее.

- **Убраны ненужные или редко используемые поля:** Нет **фрагментации на промежуточных маршрутизаторах**, нет **контрольной суммы заголовка** (целостность обеспечивается на уровнях 2 и 4). Это упрощает и ускоряет работу маршрутизаторов.
- **Новое поле «Класс трафика» (Traffic Class, 8 бит):** Аналог DSCP в IPv4, для управления качеством обслуживания (QoS).
- **Новое поле «Метка потока» (Flow Label, 20 бит):** Позволяет маркировать пакеты, принадлежащие одному потоку данных (например, видеотрансляции), чтобы маршрутизаторы могли обрабатывать их единообразно (по одному и тому же пути, с одним приоритетом).
- **Длина полезной нагрузки (Payload Length, 16 бит):** Указывает длину данных, следующих **после** IP-заголовка (включая возможные дополнительные заголовки расширения).
- **Следующий заголовок (Next Header, 8 бит):** Аналог поля «Протокол» в IPv4, но с расширенной функциональностью. Может указывать либо на транспортный протокол (TCP/UDP), либо на наличие следующего **заголовка расширения** IPv6. Заголовки расширения выстраиваются в цепочку, обеспечивая гибкость.
- **Предел прыжков (Hop Limit, 8 бит):** Прямой аналог TTL в IPv4. Счётчик переходов через маршрутизаторы.
- **Адрес источника (Source Address, 128 бит):** Увеличенный до 128 бит IP-адрес отправителя.
- **Адрес назначения (Destination Address, 128 бит):** Увеличенный до 128 бит IP-адрес получателя.

Заголовки расширения IPv6: Это отдельные блоки служебной информации, которые помещаются между основным заголовком и транспортным протоколом. Они используются по необходимости. Например:

- **Заголовок маршрутирования:** Задаёт список маршрутизаторов, которые должен посетить пакет.
- **Заголовок фрагментации:** Если фрагментация всё же необходима, она выполняется только отправителем, а не промежуточными маршрутизаторами. Информация о фрагментах помещается в этот отдельный заголовок.
- **Заголовок IPsec (AH/ESP):** Позволяет легко внедрять шифрование и аутентификацию на сетевом уровне.

5. Особенности и преимущества протокола IPv6

Основная идея: IPv6 создан не просто для расширения адресного пространства, а для решения фундаментальных проблем IPv4, предлагая более простую, эффективную, безопасную и масштабируемую архитектуру.

Пояснение смысла и преимущества:

1. **Огромное адресное пространство (Главное преимущество):** 128-битный адрес против 32-битного в IPv4. Это примерно 3.4*10^38 уникальных адресов. **Аналогия:** Если IPv4 – это

количество песчинок на небольшом пляже, то IPv6 – это количество песчинок на всей планете Земля. Это позволяет присвоить уникальный адрес каждому устройству, датчику (IoT), автомобилю, смартфону, что снимает проблему нехватки адресов и необходимость в сложных технологиях вроде NAT.

2. **Упрощённый заголовок и ускоренная маршрутизация:** Фиксированный размер, отсутствие контрольной суммы и фрагментации на маршрутизаторах позволяют аппаратным средствам обрабатывать пакеты намного быстрее.
3. **Иерархическая и эффективная адресация:** Адреса IPv6 выделяются интернет-регистратурами (RIR) провайдерам большими блоками, что позволяет агрегировать маршруты и значительно уменьшать размер таблиц маршрутизации в магистральных сетях интернета. Это улучшает масштабируемость всей сети.
4. **Встроенная (нативная) безопасность (IPsec):** Поддержка IPsec (набор протоколов для шифрования и аутентификации) является обязательной частью стандарта IPv6. Это означает, что любая IPv6-коммуникация по умолчанию может быть защищена. В IPv4 IPsec – это дополнительная опция. **Аналогия:** В IPv6 «защищённый канал» – это стандартная комплектация автомобиля, а в IPv4 – дорогая опция, которую не все заказывают.
5. **Автоконфигурация адресов (Stateless Address Autoconfiguration – SLAAC):** Устройство может самостоятельно сгенерировать себе глобальный IPv6-адрес, используя свой MAC-адрес и префикс сети, полученный от маршрутизатора через широковещательные сообщения (Router Advertisement). Это упрощает настройку крупных сетей (например, IoT). Есть и Stateful конфигурация через DHCPv6.
6. **Улучшенная поддержка мобильности (Mobile IPv6):** Протокол позволяет устройству сохранять свой постоянный IPv6-адрес, перемещаясь между сетями, обеспечивая непрерывность соединений.
7. **Устранение NAT «как необходимости»:** Хотя NAT (трансляция сетевых адресов) может использоваться в IPv6 для других целей (например, конфиденциальности), он больше не требуется для экономии адресов. Это возвращает архитектуру интернета к изначальной идее end-to-end connectivity, где любое устройство может напрямую связаться с любым другим, что упрощает разработку приложений (особенно пиринговых, VoIP, игр).

6. Методы маршрутизации узлов

Основная идея: Узлу (хосту – компьютеру, серверу) необходимо принять решение: отправить пакет непосредственно получателю в своей локальной сети или же передать его шлюзу по умолчанию для дальнейшей маршрутизации во «внешний мир».

Пояснение смысла и процесс принятия решения:

Каждый узел с IP-стеком имеет простую **логику маршрутизации**, основанную на его собственной небольшой **таблице маршрутизации**.

Шаг 1: Проверка адреса назначения.
Узел сравнивает IP-адрес назначения в пакете со своей собственной конфигурацией.

- **Если адрес назначения – это собственный адрес узла** (или широковещательный адрес 255.255.255.255/локальный 127.0.0.1), пакет не отправляется в сеть, а передаётся выше по стеку протоколов на этом же устройстве.
- **Если нет** – переходим к следующему шагу.

Шаг 2: Проверка локальной сети (Directly Connected Network).
Узел использует свою **маску подсети**, чтобы определить, принадлежит ли адрес назначения к той же IP-сети, в которой находится он сам.

- **Пример:** Узел имеет IP 192.168.1.10 с маской 255.255.255.0 (/24). Это означает, что его сеть – это 192.168.1.0/24. Адреса от 192.168.1.1 до 192.168.1.254 находятся в этой сети.

- Если адрес назначения – 192.168.1.50, узел понимает: «Получатель в моей локальной сети».
- Если адрес назначения – 8.8.8.8, узел понимает: «Получатель в другой сети».

Шаг 3: Выбор метода доставки.

- **Если получатель в локальной сети (прямая доставка):** Узел использует протокол **ARP**, чтобы найти **MAC-адрес** устройства с нужным IP-адресом. Затем он инкапсулирует IP-пакет в кадр канального уровня (Ethernet) с MAC-адресом назначения и отправляет его прямо в сеть через свой сетевой интерфейс. Маршрутизатор здесь не нужен.

- **Аналогия:** Вы хотите передать записку соседу по подъезду. Вы просто идёте к его двери и кладёте записку в почтовый ящик.
- **Если получатель в другой сети (непрямая доставка):** Узел не знает, как добраться до удалённой сети. У него есть конфигурационный параметр – **Шлюз по умолчанию (Default Gateway)**. Обычно это IP-адрес маршрутизатора в локальной сети (например, 192.168.1.1). Узел отправляет пакет **на MAC-адрес шлюза по умолчанию**, но при этом **IP-адрес назначения в пакете остаётся неизменным** (адрес удалённого сервера). Узел как бы говорит: «Эй, маршрутизатор, я не знаю, как доставить это по адресу 8.8.8.8. Ты отвечаешь за связь с внешним миром, разберись».

- **Аналогия:** Вы хотите отправить письмо в другой город. Вы опускаете его в почтовый ящик у себя во дворе (шлюз). Местное почтовое отделение (маршрутизатор) забирает письмо и дальше отвечает за его маршрутизацию по стране. Вы доверяете эту работу шлюзу.

Итог: Метод маршрутизации узла – это простой бинарный выбор на основе маски подсети: «Локальный или нет?». Вся сложность маршрутизации между сетями перекладывается на маршрутизаторы.

7. Таблица маршрутизации узлов и маршрутизатора для протоколов IPv4 и IPv6

Основная идея: Таблица маршрутизации – это база данных, хранящаяся в памяти сетевого устройства (хоста или маршрутизатора), которая содержит «карту» путей к различным сетевым назначениям.

7.1. Таблица маршрутизации узла (хоста)

Пояснение смысла: У обычного компьютера или сервера таблица маршрутизации очень простая. Она создаётся автоматически при настройке IP-адреса и шлюза.

Типичные записи (можно посмотреть командой route print в Windows или ip route в Linux):

1. **Маршрут к loopback-интерфейсу (127.0.0.0/8):** Для внутренней коммуникации приложений на самом устройстве.
2. **Маршрут к локальной подсети (Directly Connected Network):** Например, 192.168.1.0/24 через интерфейс Ethernet0 с метрикой (стоимостью) 10. Это означает: «Все адреса в этой сети доступны напрямую через мой сетевой адаптер».
3. **Шлюз по умолчанию (Default Route):** Самая важная запись. Например, 0.0.0.0/0 через шлюз 192.168.1.1. Запись 0.0.0.0/0 означает **любой адрес назначения**. Если пакет не подпадает под более специфичный маршрут (например, под локальную сеть), он будет отправлен по этому маршруту – на шлюз 192.168.1.1. Это «маршрут последнего выбора».
4. **Широковещательные и multicast-маршруты.**

Аналогия для хоста: Ваш личный адресник. В нём есть:

- Запись о вашем собственном доме (локальная сеть).
- Запись «Для всех остальных адресов – обращайся в главное почтовое отделение района» (шлюз по умолчанию).

7.2. Таблица маршрутизации маршрутизатора

Пояснение смысла: Это сложная, динамически обновляемая база данных, которая является «мозгом» маршрутизатора. Решения о пересылке пакетов принимаются на её основе.

Ключевые столбцы таблицы маршрутизации (например, вывод show ip route на Cisco):

- **Код источника (код протокола):** Показывает, откуда получен маршрут.
  - C – Connected (непосредственно подключённая сеть).
  - S – Static (статический, введённый администратором вручную).
  - O – OSPF (узнан через протокол OSPF).
  - R – RIP (узнан через RIP).
  - D – EIGRP.

- * – кандидат на шлюз по умолчанию.
- **Сеть назначения и маска:** Например, 192.168.5.0/24 или 10.0.0.0/8.
- **Административная дистанция (AD):** Числовое значение (от 0 до 255), определяющее **достоверность источника** маршрута. Чем меньше, тем лучше. Например, Connected (0) или Static (1) более надёжны, чем RIP (120). Если к одной сети ведут два маршрута из разных источников, выбирается тот, у которого AD меньше.
- **Метрика (Metric):** Числовое значение, определяющее **предпочтительность пути** в рамках одного протокола. Может учитывать количество прыжков (hop count), пропускную способность, задержку, нагрузку канала. Чем меньше метрика, тем лучше путь.
- **Следующий прыжок (Next-Hop):** IP-адрес соседнего маршрутизатора, на который нужно отправить пакет, чтобы он двинулся к цели.
- **Выходной интерфейс (Interface):** Физический или логический порт маршрутизатора, через который нужно отправить пакет, чтобы достичь next-hop.

Процесс поиска в таблице (Longest Prefix Match):
Это **ключевой принцип**. Маршрутизатор ищет не просто первое совпадение, а **самое точное совпадение префикса сети**.

- **Пример:** В таблице есть два маршрута:

1. 10.0.0.0/8 через Next-Hop A (более общий маршрут, охватывает 16 млн адресов).
2. 10.1.1.0/24 через Next-Hop B (более специфичный маршрут, охватывает 256 адресов).

- Приходит пакет на адрес 10.1.1.50.
- Маршрутизатор выбирает **второй маршрут** (10.1.1.0/24), потому что его маска длиннее (/24 > /8), а значит, он более точно описывает сеть назначения. Это как поиск адреса: сначала ищется точный адрес дома, и только если его нет – общее направление на район.

Для IPv6 (show ipv6 route): Принципы абсолютно те же. Изменяются только форматы адресов (шестнадцатеричные) и коды протоколов (например, S для статического, O для OSPFv3, C для connected, L для локального адреса интерфейса).

8. Устройство маршрутизатора – Процессор, память, операционная система

Основная идея: Маршрутизатор – это специализированный компьютер, чья аппаратная и программная архитектура оптимизирована для одной задачи – эффективной маршрутизации пакетов.

Пояснение смысла и компоненты:

1. **Центральный процессор (CPU / Route Processor):**

- **Роль:** «Мозг» маршрутизатора. Выполняет общие задачи: запуск операционной системы (IOS, Junos), обработка протоколов маршрутизации (расчёт таблиц, обмен сообщениями с соседями), управление интерфейсами, выполнение команд администратора.

- **Аналогия:** Диспетчер в логистическом центре, который не разгружает каждый грузовик лично, но управляет графиками, общается с другими центрами и решает стратегические задачи.
- **Важно:** В современных высокопроизводительных маршрутизаторах **пересылкой пакетов (data plane)** часто занимаются отдельные специализированные микросхемы (ASIC), чтобы разгрузить CPU. CPU же сосредоточен на **control plane** (управлении).

2. **Память (Memory):** В маршрутизаторе несколько типов памяти, как в ПК, но с чётким разделением функций.

- **RAM (Random Access Memory, оперативная память):**

- **Рабочая память.** Хранит **текущую конфигурацию** (running-config), **таблицу маршрутизации (Routing Table)**, **таблицу ARP**, кэши, буферы пакетов. Всё содержимое RAM стирается при выключении питания.

- **NVRAM (Non-Volatile RAM, энергонезависимая RAM):**

- Хранит **сохранённую конфигурацию (startup-config)**. Это конфигурация, которая будет загружена при следующем включении. Не стирается при отключении питания.

- **Flash-память (Flash Memory):**

- **«Жёсткий диск» маршрутизатора.** Хранит **образ операционной системы (IOS image)**, файлы конфигураций, резервные копии,日志文件. Энергонезависима.

- **ROM (Read-Only Memory, постоянная память):**

- Хранит **код POST (Power-On Self-Test)** и **миниатюрную, неизменяемую версию IOS** для восстановления (RxBoot). Обновлению не подлежит.

3. **Операционная система (например, Cisco IOS, Juniper Junos, MikroTik RouterOS):**

- **Роль:** Специализированная ОС, обеспечивающая весь функционал маршрутизатора: интерфейс командной строки (CLI), реализацию сетевых протоколов, управление интерфейсами, безопасность, инструменты диагностики.
- **Особенности:** Часто не имеет графического интерфейса (работа через CLI), модульная, устойчивая. Её команды (show, configure, ping, traceroute) – это основной инструмент сетевого администратора.

9. Подключение к маршрутизатору через различные порты

Основная идея: Для первоначальной настройки и управления маршрутизатором необходим физический доступ к нему через специальные служебные интерфейсы.

Пояснение смысла и типы подключений:

1. **Консольный порт (Console Port):**

- **Назначение: Первичный, самый низкоуровневый способ доступа.** Используется для первоначальной настройки, восстановления паролей, отладки при проблемах с сетевым

подключением. Работает даже если маршрутизатор не имеет IP-адреса или его интерфейсы выключены.

- **Физика:** Обычно это порт RJ-45 или mini-USB.
- **Подключение:** С помощью кабеля «консольный кабель» (Cisco Rollover Cable) или USB-кабеля к COM- или USB-порту компьютера. На ПК используется программа-эмулятор терминала (Putty, Tera Term, SecureCRT) с параметрами: скорость 9600 бод, 8 бит данных, 1 стоп-бит, без контроля чётности (8N1).
- **Аналогия:** Это как подключить клавиатуру и монитор непосредственно к системному блоку сервера в серверной комнате (KVM). Полный контроль «изнутри».

2. **Вспомогательный порт (Auxiliary Port, AUX):**

- **Назначение:** Удалённый доступ через модем. В современных реалиях используется редко. Позволяет подключить внешний модем для доступа к консоли по телефонной линии, если сеть недоступна.
- **Подключение:** Аналогично консоли, но через модем.

3. **Подключение через сетевые интерфейсы (по Telnet / SSH / HTTP/HTTPS):**

- **Назначение: Удалённое управление** после того, как маршрутизатору назначены IP-адреса и подняты сетевые интерфейсы. Это основной способ ежедневной работы.
- **Telnet (Порт 23):** Протокол удалённого терминала **без шифрования**. Все данные (включая логины и пароли) передаются в открытом виде. **Не рекомендуется** для использования в производственных сетях.
- **SSH (Secure Shell, Порт 22): Основной и безопасный** протокол. Обеспечивает шифрование всего сеанса. Должен использоваться всегда, когда возможен доступ из ненадёжных сетей.
- **HTTP/HTTPS (Порты 80/443):** Веб-интерфейс (GUI). Предоставляет графический интерфейс для настройки. Чаще встречается на маршрутизаторах SOHO-уровня (домашних/малых офисных). На корпоративных устройствах может быть отключён в целях безопасности. **HTTPS** предпочтительнее из-за шифрования.
- **Для доступа:** На ПК в эмуляторе терминала указывается IP-адрес одного из интерфейсов маршрутизатора (часто VLAN 1 или выделенный management-интерфейс).

10. Настройка исходных параметров, интерфейсов, шлюза по умолчанию и других характеристик маршрутизатора

Основная идея: Настройка маршрутизатора – это процесс задания ему параметров, которые превращают его из пассивного устройства в активный элемент сети, способный принимать решения о маршрутизации.

Пояснение смысла и базовые шаги (на примере CLI Cisco IOS):

Шаг 0: Подключение и вход.
Подключаемся через консольный порт, включаем маршрутизатор. Видим приглашение Router>. Это **пользовательский режим (User EXEC Mode)**, где доступны только команды просмотра (show).

Шаг 1: Переход в привилегированный режим.

```
Router> enable
Router#
```

Приглашение меняется на #. Это **привилегированный режим (Privileged EXEC Mode)**. Здесь доступны все команды просмотра и диагностики, но ещё не изменения конфигурации.

Шаг 2: Переход в режим глобальной конфигурации.

```
Router# configure terminal
Router(config)#
```

Теперь мы в **режиме глобальной конфигурации**. Команды, введённые здесь, влияют на работу устройства в целом.

Шаг 3: Настройка имени устройства (hostname).

```
Router(config)# hostname BRANCH-ROUTER
BRANCH-ROUTER(config)#
```

Имя помогает идентифицировать устройство в сети. Приглашение меняется сразу.

Шаг 4: Настройка паролей (базовая безопасность).

- **Пароль для входа в привилегированный режим:**

```
BRANCH-ROUTER(config)# enable secret myStrongPassword
```

Пароль myStrongPassword будет храниться в зашифрованном виде.

- **Пароль на консольный порт:**

```
BRANCH-ROUTER(config)# line console 0
BRANCH-ROUTER(config-line)# password consolePass
BRANCH-ROUTER(config-line)# login
BRANCH-ROUTER(config-line)# exit
```

- **Пароль на доступ по Telnet/SSH (на виртуальные терминалы VTY):**

```
BRANCH-ROUTER(config)# line vty 0 4
BRANCH-ROUTER(config-line)# password vtyPass
BRANCH-ROUTER(config-line)# login
BRANCH-ROUTER(config-line)# transport input ssh  # Разрешить только SSH
(лучшая практика)
BRANCH-ROUTER(config-line)# exit
```

Шаг 5: Настройка IP-адреса на интерфейсе (например, GigabitEthernet 0/0).
Интерфейс – это «дверь» маршрутизатора в сеть.

```
BRANCH-ROUTER(config)# interface gigabitEthernet 0/0
BRANCH-ROUTER(config-if)# description LINK-TO-LAN  # Описание для себя
BRANCH-ROUTER(config-if)# ip address 192.168.1.1 255.255.255.0
BRANCH-ROUTER(config-if)# no shutdown  # АКТИВИРОВАТЬ интерфейс (по умолчанию они выключены)
BRANCH-ROUTER(config-if)# exit
```

Теперь маршрутизатор знает, что в сети 192.168.1.0/24 он находится по адресу 192.168.1.1. Он автоматически добавит маршрут к этой сети в таблицу (Connected).

Шаг 6: Настройка шлюза по умолчанию для самого маршрутизатора.
Зачем он маршрутизатору? Чтобы маршрутизатор мог отправлять пакеты в сети, о которых у него нет конкретных маршрутов (например, для отправки логов на внешний сервер, для доступа к NTP, или для трафика, исходящего от него самого).

text

```
BRANCH-ROUTER(config)# ip route 0.0.0.0 0.0.0.0 10.0.0.1
```

Эта команда гласит: «Все пакеты для любых сетей (0.0.0.0/0), для которых нет более точного маршрута, отправляй на next-hop 10.0.0.1». Адрес 10.0.0.1 должен быть достижим через один из интерфейсов маршрутизатора (например, через другой настроенный интерфейс, смотрящий в сеть провайдера).

Аналогия для шагов 5 и 6: Вы настраиваете логистический узел:

1. Даёте адрес своей приёмной площадке для местных грузов (ip address на интерфейсе).
2. Указываете адрес главного распределительного центра страны, куда отправлять все грузы, пункт назначения которых вам неизвестен (ip route 0.0.0.0 0.0.0.0 ...).

Шаг 7: Настройка для IPv6 (аналогичные шаги).

```
BRANCH-ROUTER(config)# ipv6 unicast-routing  // Включаем маршрутизацию IPv6
BRANCH-ROUTER(config)# interface gigabitEthernet 0/0
BRANCH-ROUTER(config-if)# ipv6 address 2001:db8:acad:1::1/64  // Назначаем статический адрес
BRANCH-ROUTER(config-if)# ipv6 address fe80::1 link-local  // Можно задать link-local адрес
BRANCH-ROUTER(config-if)# exit
BRANCH-ROUTER(config)# ipv6 route ::/0 2001:db8:acad:1::100  // Шлюз по умолчанию для IPv6
```

Шаг 8: Сохранение конфигурации.
Вся конфигурация, сделанная выше, пока что находится только в RAM (running-config). При перезагрузке она исчезнет.

```
BRANCH-ROUTER# copy running-config startup-config
```

или

```
BRANCH-ROUTER# write memory
```

Теперь конфигурация сохранена в NVRAM (startup-config) и будет загружаться при каждом включении.

Заключение: Настройка маршрутизатора – это процесс «оживления» его интерфейсов, задания им адресов (как почтовых адресов для сетей) и указания путей (шлюзов) для движения данных. Понимание принципов сетевого уровня, структуры IP-пакетов и логики работы таблицы маршрутизации позволяет осознанно выполнять эту настройку, превращая набор аппаратных компонентов в ключевой элемент инфраструктуры передачи данных.

Контрольные вопросы к ТЕМЕ-4

1. Какая основная задача сетевого уровня в модели передачи данных? Приведите простую аналогию.
2. Назовите основные характеристики протокола IP (Internet Protocol). В чём заключается его принцип "best-effort delivery"?
3. Чем отличается дейтаграммный (connectionless) режим передачи IP от режима с установлением соединения?
4. Для чего в заголовке IPv4-пакета служит поле TTL (Time to Live)? Опишите принцип его работы.
5. Перечислите не менее трёх ключевых преимуществ протокола IPv6 перед IPv4.
6. Как узел (хост) определяет, находится ли компьютер-получатель в его локальной сети или в удалённой? Какой параметр конфигурации он для этого использует?
7. Что такое шлюз по умолчанию (Default Gateway) и для чего он нужен конечному узлу?
8. Какая запись в таблице маршрутизации указывает на шлюз по умолчанию для IPv4?
9. По какому правилу (принципу) маршрутизатор выбирает маршрут из таблицы, если есть несколько подходящих записей для одной сети назначения?
10. Для первоначальной настройки маршрутизатора чаще всего используется специальный порт. Как он называется и почему именно он?
11. Какая команда в Cisco IOS используется для перехода в режим, позволяющий вносить изменения в конфигурацию устройства (режим глобальной конфигурации)?
12. Что происходит с конфигурацией маршрутизатора, введённой в командной строке, после его перезагрузки, если не выполнить команду сохранения? В какой памяти хранится текущая и сохранённая конфигурация?

Практические задания по ТЕМЕ-4

Задание 1. Диагностика локальной сети в малом офисе

Ситуация: Вы работаете сетевым администратором в небольшой компании в Нижнем Новгороде. Сотрудник из отдела бухгалтерии (компьютер с IP-адресом 192.168.15.25, маска подсети 255.255.255.0) жалуется, что не может получить доступ к сетевому принтеру, расположенному в том же офисе (IP-адрес принтера 192.168.15.100). При этом доступ в интернет у сотрудника работает исправно. Основываясь на теоретическом материале о методах маршрутизации узлов, предположите, в чём наиболее вероятная причина проблемы? Какой простой шаг нужно предпринять для её проверки? Обоснуйте свой ответ.

Задание 2. Анализ проблемы с доставкой пакетов

Ситуация: Вы настраиваете сеть в учебном центре ДОСААФ. С рабочей станции успешно выполняются команды ping до локальных серверов, но при попытке обратиться к удалённому учебному порталу (например, education.ru) пакеты не доходят. Вы запускаете команду tracert и видите, что пакеты доходят только до первого же маршрутизатора вашей сети, а далее след обрывается. Основываясь на знаниях о структуре IP-пакетов и процессе маршрутизации, какое поле в IP-заголовке могло стать причиной такого поведения на первом же внешнем маршрутизаторе провайдера? Что, скорее всего, не было настроено на вашем маршрутизаторе, ведущем во внешнюю сеть?

Задание 3. Выбор между IPv4 и IPv6 для нового проекта

Ситуация: Вы участвуете в планировании IT-инфраструктуры для нового умного склада в логистическом парке «Москва». Технический директор предлагает использовать для всех систем учёта, датчиков контроля климата и роботизированных тележек новейший протокол IPv6, аргументируя это его современностью. Главный бухгалтер настаивает на проверенном IPv4, чтобы не было проблем с совместимостью. Вам поручено подготовить краткий сравнительный анализ для принятия решения. Основываясь на материале об особенностях протоколов, приведите один технический аргумент **в пользу** IPv6 именно для данного проекта (склад с множеством датчиков и устройств) и один аргумент об **ограничении** IPv6, который может склонить решение в сторону IPv4 в российских реалиях.

Задание 4. Восстановление доступа к маршрутизатору после сбоя

Ситуация: После внезапного отключения электроэнергии в филиале компании в Казани офисный маршрутизатор перезагрузился. Сетевые подключения внутри офиса восстановились, но доступ в интернет пропал. Вы, находясь в головном офисе, не можете подключиться к устройству по SSH для диагностики. Коллега на месте подтверждает, что индикаторы на маршрутизаторе горят, кабели подключены. Какая наиболее вероятная причина потери удалённого доступа? Какой аппаратный компонент маршрутизатора,

вероятно, не сохранил критически важные настройки? Что должен был сделать администратор после первоначальной настройки устройства, чтобы избежать этой ситуации?

Задание 5. Объяснение принципа маршрутизации на бытовом примере

Ситуация: Вам нужно объяснить принцип работы таблицы маршрутизации и шлюза по умолчанию новому стажёру, не использующую сложную терминологию. Придумайте аналогию из **российской почтовой системы** (не курьерской службы), которая бы наглядно показала:

1. Как почтовое отделение (маршрутизатор) решает, отправить письмо (пакет) внутри своего города (локальной сети) или в другой город.
2. Что такое «шлюз по умолчанию» в этой аналогии.
3. Что произойдёт, если в почтовом отделении не будет маршрута для города назначения и не будет указан шлюз по умолчанию.
   Опишите эту аналогию кратко, в 3-4 предложения.

ТЕМА-5. Транспортный уровень

1. Введение в транспортный уровень модели OSI/TCP-IP

Представьте себе большую почтовую службу, которая доставляет посылки по всему миру. Сетевой уровень (например, протокол IP) – это служба, которая маркирует посылки адресами отправителя и получателя (IP-адресами) и прокладывает маршруты между городами и странами. Но что происходит, когда посылка прибывает в нужный город и конкретный адрес? В большом офисном здании (которое представляет собой один компьютер с одним IP-адресом) могут работать десятки разных компаний и служб. Как почтальон поймёт, кому именно из сотрудников здания предназначается конверт? И как обеспечить, чтобы важный юридический документ дошёл без ошибок, в то время как рекламный флаер не так критичен к надежности?

Именно эти задачи решает **транспортный уровень**. Он является посредником между приложениями, работающими на компьютере (браузером, мессенджером, игрой), и сетью. Его основная роль – обеспечить **доставку данных от конкретного приложения на одном компьютере к конкретному приложению на другом компьютере**, причем с заданным качеством обслуживания.

Фундаментальные задачи транспортного уровня:

1. **Мультиплексирование и демультиплексирование:** Разделение общего сетевого потока данных между множеством приложений (как почтальон в нашем примере распределяет письма по разным офисам в здании).
2. **Надежная или ненадежная доставка:** Предоставление приложениям выбора. Нужна ли гарантия, что каждый байт данных дойдет целым и в правильном порядке (как при отправке документа), или важнее скорость, а потеря части данных допустима (как в видеозвонке)?
3. **Управление потоком:** Регулировка скорости передачи данных, чтобы быстрый отправитель не "захлестнул" медленного получателя.
4. **Управление перегрузкой:** Регулировка скорости передачи, чтобы не перегрузить саму сеть (дороги, по которым едут посылки).

Ключевая аналогия: Если сеть – это система дорог (IP), то транспортный уровень – это службы доставки (TCP – как надежная курьерская служба с отслеживанием и подтверждением, UDP – как обычная почта, которая просто бросает письмо в ящик).

2. Мультиплексирование сеансов связи. Адресация портов

Вернёмся к нашему офисному зданию (компьютеру с IP-адресом 192.168.1.10). В нём одновременно работают:

- Веб-браузер, загружающий страницу

- Мессенджер, принимающий сообщения
- Онлайн-игра, передающая данные о положении игрока
- FTP-клиент, скачивающий файл

Все эти программы (процессы) используют одну сетевую карту и один IP-адрес для общения с внешним миром. Как сеть понимает, какой пакет данных какому приложению предназначен?

Для этого транспортный уровень вводит понятие **порта**. **Порт** – это числовой идентификатор (от 0 до 65535), который указывает на конкретное приложение или службу внутри компьютера. Вместе IP-адрес и порт образуют уникальный адрес для конечной точки общения – **сокет** (socket).

- **Пример:** Сервер с IP-адресом 93.184.216.34 предоставляет веб-сайт. Веб-сервер "слушает" (ожидает подключений) на **порту 80**. Полный адрес для подключения к веб-странице будет 93.184.216.34:80. Ваш браузер, отправляя запрос, сам выбирает для себя свободный порт на вашем компьютере, например, 192.168.1.10:54321. Таким образом, соединение уникально идентифицируется двумя сокетами: 93.184.216.34:80 и 192.168.1.10:54321.

Мультиплексирование – это процесс на стороне отправителя, когда множество потоков данных от разных приложений (с разных портов) "упаковываются" в исходящий сетевой трафик.
Демультиплексирование – это обратный процесс на стороне получателя, когда входящий поток данных "распаковывается" и распределяется по соответствующим приложениям на основе номеров портов в заголовках транспортных протоколов.

Аналогия с почтой: IP-адрес – это адрес здания. Номер порта – это номер квартиры или офиса внутри этого здания. Почтальон (сетевой уровень) приносит письмо в здание, а консьерж (транспортный уровень) смотрит на номер квартиры (порт) и опускает письмо в нужный почтовый ящик (буфер приложения).

Классификация портов:

- **Известные порты (Well-Known Ports, 0-1023):** Закреплены за общесистемными службами (HTTP – 80, HTTPS – 443, DNS – 53, SSH – 22).
- **Зарегистрированные порты (Registered Ports, 1024-49151):** Могут быть зарегистрированы для менее распространенных служб.
- **Динамические/частные порты (Dynamic/Private Ports, 49152-65535):** Используются клиентами временно для исходящих соединений.

3. Протокол UDP (User Datagram Protocol)

UDP – это простой, минималистичный протокол транспортного уровня. Его главный принцип – "отправил и забыл". Он предоставляет приложениям самый базовый сервис:

возможность отправить блок данных (датаграмму) с указанием порта назначения, без каких-либо гарантий.

Характеристики UDP:

- **Ненадежный:** Нет подтверждений о доставке. Отправитель не узнает, дошла ли датаграмма до получателя, была ли она потеряна или повреждена в пути.
- **Без установления соединения:** Нет процедуры "рукопожатия". Каждая датаграмма отправляется независимо, даже если адресат не готов её принять.
- **Не гарантирует порядок:** Датаграммы могут приходить в порядке, отличном от отправленного.
- **Отсутствие контроля потока и перегрузки:** Отправитель может посылать данные с любой скоростью, рискуя перегрузить получателя или сеть.

Структура UDP-датаграммы (очень простая):
Заголовок всего 8 байт и содержит всего 4 поля: порт источника, порт назначения, длина и контрольная сумма. Основную часть составляет полезные данные от приложения.

Процессы UDP-сервера и клиента:

1. **Сервер:** Программа-сервер привязывается к конкретному порту (например, DNS-сервер к порту 53) и переходит в режим ожидания (recvfrom()).
2. **Клиент:** Программа-клиент создает датаграмму, указывает в ней IP-адрес и порт сервера, свой порт-источник и отправляет её (sendto()).
3. **Обмен:** Сервер получает датаграмму, обрабатывает её и, если требуется, отправляет ответную датаграмму на адрес и порт клиента, указанные в полученном пакете.

Где используется UDP? (Область применения)

- **Приложения, чувствительные к задержкам, где потеря части данных предпочтительнее ожидания:** Онлайн-видео (Zoom, Skype), онлайн-игры (где важна актуальность состояния, а не каждый кадр), голосовая связь (VoIP).
- **Запрос-ответные протоколы с малым размером данных:** DNS-запросы (один короткий запрос – один короткий ответ, при потере запрос просто повторяется).
- **Широковещательная и многоадресная рассылка:** Трансляция видео или аудио на множество клиентов (например, IP-телевидение).
- **Протоколы, реализующие свою собственную логику надежности поверх UDP:** Например, QUIC (новый протокол для HTTP/3), который ускоряет веб-соединения.

Аналогия: UDP – как отправка открытки обычной почтой. Вы написали текст, указали адрес, бросили в почтовый ящик. Вы не знаете, дойдет ли она, не помнется ли, придет ли первой или последней в серии открыток. Но это быстро и дешево. Для поздравительной открытки это приемлемо.

4. Протокол TCP (Transmission Control Protocol)

TCP – это сложный, надежный протокол с установлением соединения. Его цель – создать между двумя приложениями виртуальный **надёжный канал связи**, по которому поток байтов будет передан без ошибок, в правильном порядке и без потерь.

Ключевые характеристики TCP:

- **Надёжность:** Гарантирует доставку всех данных. Использует механизмы подтверждения (ACK) и повторной передачи (Retransmission) потерянных сегментов.
- **Установление соединения:** Перед обменом данными проводится процедура "трёхстороннего рукопожатия" для согласования параметров связи.
- **Гарантия порядка:** Данные нумеруются (последовательные номера), и на стороне получателя собираются в исходном порядке, даже если пакеты пришли вразнобой.
- **Контроль потока (Flow Control):** Механизм "скользящего окна" позволяет получателю диктовать отправителю комфортную скорость передачи, исходя из своей готовности принимать данные.
- **Управление перегрузкой (Congestion Control):** TCP динамически оценивает загрузку сети и регулирует скорость отправки, чтобы не вызывать коллапс в перегруженных узлах.

4.1. Установление TCP-соединения: "Трёхстороннее рукопожатие"

Прежде чем начать передачу данных, клиент и сервер должны "поздороваться" и договориться о начале общения. Этот процесс состоит из трёх шагов:

1. **SYN (Синхронизация):** Клиент отправляет серверу специальный сегмент с флагом SYN=1 и случайным начальным номером последовательности (Sequence Number, Seq=X). Это означает: "Привет, сервер! Я хочу установить соединение, и мой начальный номер для подсчёта байтов – X".
2. **SYN-ACK (Синхронизация и Подтверждение):** Сервер, получив SYN, отвечает сегментом с двумя флагами: SYN=1 и ACK=1. В этом сегменте он указывает: свой начальный номер последовательности (Seq=Y) и номер подтверждения (Ack=X+1). Ack=X+1 означает: "Я получил твой SYN с номером X и ожидаю следующий байт с номером X+1". Этим сервер подтверждает получение SYN клиента и отправляет свой SYN.
3. **ACK (Подтверждение):** Клиент отправляет серверу сегмент с флагом ACK=1. Он указывает номер подтверждения Ack=Y+1, что значит: "Я получил твой SYN с номером Y, всё в порядке, соединение установлено". После этого может начаться передача данных.

Аналогия с телефонным звонком:

1. **(SYN)** Вы набираете номер и слышите гудки. ("Привет, я хочу поговорить").
2. **(SYN-ACK)** На другом конце поднимают трубку и говорят "Алло?". ("Я здесь, готов к разговору, ты меня слышишь?").

3. **(ACK)** Вы говорите "Привет, это Петя!". ("Да, я тебя слышу, соединение установлено, начинаем разговор").

   После этого виртуальный "провод" между приложениями проложен, и по нему можно передавать данные с гарантией.

4.2. Надёжность TCP: Подтверждения, номера последовательностей и повторная передача

Как TCP обеспечивает надежность?

1. **Сегментация и нумерация:** Приложение отправляет через TCP поток байтов. TCP разбивает этот поток на части – **сегменты**. Каждому байту в потоке присваивается порядковый номер. В заголовок каждого сегмента записывается номер первого байта в этом сегменте.
2. **Подтверждение (ACK):** Получатель, успешно приняв сегмент, отправляет отправителю специальное **подтверждение (ACK)**. В нем содержится номер следующего ожидаемого байта. Например, если получен сегмент с байтами 1000-1499, то отправится ACK с номером 1500 ("Я успешно получил всё вплоть до байта 1499 и жду байт номер 1500").
3. **Таймеры и повторная передача:** На стороне отправителя для каждого отправленного, но не подтверждённого сегмента заводится таймер. Если в течение заданного времени (RTT – Round Trip Time) подтверждение не пришло, сегмент считается утерянным и передаётся заново.

Аналогия с курьером и инвентаризацией:
Вы отправляете ценные ящики с книгами (байты данных) со склада А на склад Б. Каждый ящик имеет уникальный номер (номер последовательности). На складе Б принимают ящики, проверяют их целостность и отправляют на склад А квитанцию (ACK) с номером последнего принятого ящика. Например, квитанция "Получены ящики вплоть до №15". Если на складе А не получили квитанцию за ящик №10 в течение разумного времени, они считают его потерянным и отправляют его дубликат.

4.3. Управление потоком в TCP: Механизм "Скользящего окна"

Проблема: что, если отправитель (мощный сервер) отправляет данные быстрее, чем получатель (маломощный смартфон) может их обработать и принять в свой буфер? Данные будут теряться.

Решение: **Механизм скользящего окна (Sliding Window).**

- Получатель в каждом подтверждении (ACK) сообщает отправителю свой **размер доступного окна приёма (Window Size)**. Это количество байт, которое получатель готов принять прямо сейчас.
- Отправитель имеет право отправить без ожидания подтверждения количество данных, не превышающее это окно.
- По мере того как получатель обрабатывает данные и освобождает буфер, он увеличивает размер окна в следующих ACK-сообщениях, и отправитель может ускориться.
- Если буфер получателя заполняется, он уменьшает размер окна до нуля, временно останавливая передачу. Когда буфер освободится, он отправит ACK с новым размером окна, и передача возобновится.

Аналогия с конвейером: Получатель – это рабочий на конвейере. У него есть стол ограниченного размера (буфер), куда кладут детали. В каждом подтверждении он сообщает отправителю: "На моём столе свободно место для 10 деталей". Отправитель может отправить сразу 10 деталей, но не больше. Как только рабочий обработает 5 деталей, он крикнет: "Теперь свободно место для 15!" (новое окно = старое окно + 5). Таким образом, скорость работы конвейера автоматически подстраивается под скорость рабочего.

4.4. Завершение TCP-соединения

Соединение должно быть так же вежливо закрыто, как и установлено. Обычно это **четырёхэтапный процесс** (хотя иногда он сводится к трём шагам).

1. **FIN (Завершение от одной стороны):** Сторона, которая первой решила завершить передачу (например, клиент), отправляет сегмент с флагом FIN=1. Это означает: "Я закончил передачу данных, но ещё готов получать твои данные, если ты их отправляешь".
2. **ACK (Подтверждение FIN):** Вторая сторона (сервер) отправляет подтверждение (ACK) на полученный FIN.
3. **FIN (Завершение от второй стороны):** Когда и сервер готов завершить передачу, он отправляет свой сегмент с флагом FIN=1.
4. **ACK (Финальное подтверждение):** Клиент отправляет подтверждение (ACK) на FIN от сервера. После этого все ресурсы соединения освобождаются.

Аналогия с окончанием разговора:

1. **(FIN)** Вы говорите: "Ну всё, я закончил, мне больше нечего сказать".
2. **(ACK)** Собеседник отвечает: "Хорошо, я понял, что ты закончил".
3. **(FIN)** Собеседник добавляет: "И я тоже всё сказал".
4. **(ACK)** Вы отвечаете: "Ладно, тогда пока!" и кладёте трубку.

5. Сравнение TCP и UDP: итоговая таблица и области применения

Критерий	TCP	UDP
Надёжность	**Надёжный.** Гарантирует доставку, порядок, целостность.	**Ненадёжный.** Доставка, порядок, отсутствие дубликатов не гарантируются.
Соединение	**С установлением соединения.** Требует "рукопожатия".	**Без установления соединения.** Данные отправляются сразу.
Скорость/Накладные расходы	**Медленнее, выше накладные расходы.** Из-за установления соединения, подтверждений, контроля потока.	**Быстрее, меньше накладных расходов.** Минимальный заголовок, нет служебной логики.
Контроль потока и перегрузки	**Присутствует.** Автоматически регулирует скорость под получателя и сеть.	**Отсутствует.** Отправитель решает, как быстро слать данные.
Модель доставки	**Поток байтов.** Границы сообщений не сохраняются.	**Датаграммы.** Сохраняет границы сообщений, отправленных приложением.
Примеры приложений	Веб-браузеры (HTTP/HTTPS), электронная почта (SMTP, IMAP), передача файлов (FTP), удалённый доступ (SSH).	Видеоконференции, потоковое видео, онлайн-игры, DNS-запросы, широковещательные рассылки (DHCP).

Простое правило выбора для разработчика:

- **Используйте TCP, когда важна каждая крупица информации и её порядок.** Например, веб-страница, электронное письмо, файл, банковская транзакция. Вы не хотите, чтобы половина текста письма пропала или абзацы перепутались.
- **Используйте UDP, когда важна скорость и актуальность, а потери допустимы.** Например, в видеозвонке лучше потерять несколько кадров (небольшие искажения), чем "зависнуть" в ожидании их повторной отправки. В игре важнее получить

самое свежее положение противника с небольшой задержкой, чем абсолютно точное, но уже устаревшее положение.

6. Заключение

Транспортный уровень является сердцем коммуникаций в сетях TCP/IP, предоставляя приложениям необходимый сервис: от простой и быстрой отправки сообщений (UDP) до создания надёжных, управляемых виртуальных каналов (TCP). Понимание различий между этими протоколами, принципов их работы (порты, рукопожатие, подтверждения, окно) является фундаментальным для любого специалиста в области сетевого и системного администрирования. Это знание позволяет не только правильно настраивать сетевое оборудование (например, настраивать правила файрвола для конкретных портов), но и диагностировать проблемы ("почему тормозит передача файлов?" – возможно, проблемы с TCP-окном или перегрузкой сети), а также понимать логику работы практически любого сетевого приложения, с которым приходится сталкиваться в профессиональной деятельности.

Контрольные вопросы к ТЕМЕ-5

1. Назовите две основные задачи транспортного уровня, которые отличают его от сетевого уровня (IP).
2. Что такое порт в контексте транспортного уровня и для чего он нужен? Приведите пример сокета (адреса конечной точки соединения).
3. Объясните процесс мультиплексирования и демультиплексирования на транспортном уровне, используя аналогию с почтовой службой.
4. Перечислите четыре ключевые характеристики протокола UDP (кратко опишите каждую одной фразой).
5. В каких типах приложений и почему чаще всего используется протокол UDP? Приведите три примера.
6. Опишите три ключевые характеристики протокола TCP, обеспечивающие его надежность.
7. Как называется и из скольких шагов состоит процесс установления соединения в TCP? Назовите эти шаги и соответствующие им флаги.
8. Объясните смысл «трёхстороннего рукопожатия» TCP, используя аналогию с телефонным звонком.
9. Как TCP обеспечивает надежность при потере данных в сети? Какой механизм для этого используется?
10. Для решения какой проблемы в TCP используется механизм «скользящего окна»? Объясните его основную идею.
11. Какой протокол (TCP или UDP) является соединённым, а какой — нет? Что это означает на практике?

12. Используя простое правило выбора для разработчика, определите, какой протокол (TCP или UDP) следует использовать для отправки электронной почты, а какой — для онлайн-видеозвонка. Обоснуйте свой выбор.

Практические задания по ТЕМЕ-5

Задание 1. «Диагностика работы сервиса»

Вы – администратор в небольшом колледже в Екатеринбурге. Преподаватель жалуется, что не может загрузить учебные материалы с внутреннего файлового сервера files.edu-uralsk.ru. При этом доступ к обычным веб-сайтам в Интернете у него работает. Вы проверяете сетевую доступность: команда ping до адреса сервера показывает, что пакеты успешно доходят и возвращаются.

- **Вопрос:** Основываясь на знании функций транспортного уровня, между какими двумя компонентами (назовите их конкретно) предположительно возникла проблема, если IP-связность есть? Какой простой, но информативный шаг вы можете предпринять в командной строке Windows (netstat или telnet) или Linux (nc), чтобы проверить свою догадку, не запуская браузер? (Подсказка: подумайте о портах и состоянии «прослушивания»).

Задание 2. «Выбор протокола для корпоративного приложения»

Ваша компания в Казани разрабатывает мобильное приложение для курьерской службы. Одна функция приложения — передавать координаты местоположения курьера на карте в режиме реального времени на сервер для отслеживания. Вторая функция — формировать и отправлять на сервер детальный электронный отчёт о доставке (чек, подпись клиента, фото) в конце рабочего дня.

- **Вопрос:** Какой транспортный протокол (TCP или UDP) вы порекомендуете использовать для каждой из этих функций? Дайте краткое обоснование для каждого выбора, исходя из требований к надёжности и скорости.

Задание 3. «Анализ сетевого трафика»

При рассмотрении дампа сетевого трафика (например, в Wireshark) молодой администратор в Твери увидел следующую последовательность TCP-сегментов между компьютером 192.168.1.50 и сервером 93.184.216.34:

1. [SYN] от 192.168.1.50:49152 к 93.184.216.34:443
2. [SYN, ACK] от 93.184.216.34:443 к 192.168.1.50:49152
3. [ACK] от 192.168.1.50:49152 к 93.184.216.34:443

- **Вопрос 1:** Какому реальному действию пользователя, скорее всего, соответствует этот обмен сегментами?

- **Вопрос 2:** Если бы вместо третьего сегмента [ACK] администратор увидел множество повторных [SYN] от клиента, о какой типичной сетевой проблеме это могло бы свидетельствовать? (Используйте аналогию из теоретического материала).

Задание 4. «Настройка межсетевого экрана»

В ИТ-отделе больницы в Новосибирске настраивают межсетевой экран (firewall) для защиты медицинской информационной системы. Система состоит из сервера баз данных (СУБД) и клиентских рабочих станций врачей. Необходимо разрешить только легитимный обмен данными.

- **Вопрос:** Исходя из знания о характеристиках TCP и UDP, какой из этих протоколов, вероятно, используется для связи между клиентами и сервером СУБД? Аргументируйте. Как этот вывод должен повлиять на правило файрвола: стоит ли просто разрешить весь TCP-трафик между этими узлами или необходимы более тонкие настройки? Если да, то настройка какого параметра транспортного уровня (указанного в заголовках пакетов) будет ключевой для создания безопасного правила?

Задание 5. «Объяснение принципа коллеге»

Ваш коллега, начинающий системный администратор в Краснодаре, не может понять, почему при скачивании большого файла из Интернета скорость сначала резко растёт, а потом колеблется вокруг некоторого значения, даже если канал, казалось бы, не загружен полностью. Вы знаете, что это связано с одним из механизмов TCP.

- **Вопрос:** Как называется этот механизм? Объясните коллеге на простом примере с дорожным движением (придумайте аналогию), почему TCP ведёт себя именно так, вместо того чтобы постоянно передавать данные с максимально возможной скоростью.

ТЕМА-6. Уровень приложений

Введение в верхние уровни модели OSI/TCP IP

Прежде чем погрузиться в детали уровня приложений, важно понять его место в общей картине сетевого взаимодействия. Вспомним двух известных «архитекторов» сетей: эталонную модель **OSI** (7 уровней) и практическую модель **TCP/IP** (4 уровня). Уровень приложений в TCP/IP — это как бы «слияние» трех верхних уровней модели OSI: прикладного (Application), представления (Presentation) и сеансового (Session). Представьте себе отправку письма:

- **Уровень приложений (OSI & TCP/IP):** Вы решаете, что хотите написать письмо другу (цель/сервис). Это уровень ваших мыслей и намерений.
- **Уровень представления (OSI):** Вы облачаете свои мысли в слова и предложения на понятном вам и другу языке, возможно, шифруете часть текста или используете особый стиль (кодирование, шифрование, сжатие). Это уровень «формата» данных.
- **Сеансовый уровень (OSI):** Вы начинаете диалог: «Привет!», ждете ответа «Привет!», затем излагаете суть, задаете вопросы, поддерживаете связь, пока не решите, что разговор окончен (установление, управление и завершение сеанса).

В реальном мире Интернета (модель TCP/IP) эти три функции тесно переплетены и реализуются непосредственно в приложениях и их протоколах, которые мы и будем изучать.

1. Уровень приложений: Сущность и назначение

Базовое определение: Уровень приложений — это набор протоколов и интерфейсов, с помощью которых пользовательские приложения (браузер, почтовая программа, мессенджер) получают доступ к сетевым услугам. Это «лицо» сети для конечного пользователя.

Главный смысл: Этот уровень **не создает сами приложения** (например, Photoshop или игру), а предоставляет им стандартизированные «правила общения» по сети. Его задача — определить, как приложения разных производителей на разных компьютерах могут понимать друг друга.

Аналогия: Язык общения в профессии. Допустим, вы сантехник. Вы приезжаете к клиенту, и чтобы устранить проблему, вы используете:

1. **Профессиональный жаргон** («нужно заменить кран-буксу в смесителе»).
2. **Стандартные инструменты** (разводной ключ, фум-лента).
3. **Четкую последовательность действий** (перекрыть воду, открутить, установить новое, проверить).

Это и есть ваш «протокол уровня приложений» в мире сантехники. Благодаря ему другой сантехник, приехав после вас, поймет, что было сделано. В компьютерных сетях роль такого «профессионального языка» играют протоколы: HTTP для веб-страниц, SMTP для отправки почты и т.д.

Примеры распространенных приложений, использующих сеть:

- **Веб-браузеры** (Google Chrome, Firefox): используют HTTP/HTTPS.
- **Почтовые клиенты** (Outlook, Thunderbird) и веб-почта (Gmail): используют SMTP (отправка), POP3/IMAP (получение).
- **Программы обмена файлами** (BitTorrent, uTorrent): используют P2P-протоколы.
- **Мессенджеры** (Telegram, WhatsApp): используют свои прикладные протоколы (часто на основе HTTP/WebSocket).
- **Удаленный доступ** (TeamViewer, RDP): используют протоколы для передачи экрана и управления.

2. Архитектура сетевых приложений: Клиент-Сервер и P2P

Как приложения организуют свое общение? Есть две фундаментальные философии.

2.1. Модель «Клиент-Сервер»

Базовое определение: Это архитектура, в которой есть постоянные, мощные узлы — **серверы**, предоставляющие услуги (хранение сайтов, почты, файлов), и многочисленные, периодически подключающиеся узлы — **клиенты**, которые эти услуги запрашивают.

Суть и аналогия: Ресторан. Сервер — это кухня и штат поваров. Клиент — это посетитель за столиком. Клиент (посетитель) инициирует взаимодействие: изучает меню (запрос), делает заказ. Сервер (кухня) обрабатывает запрос, готовит блюдо и предоставляет результат. Сервер всегда «на связи» и готов к запросам, клиент обращается тогда, когда ему нужно. Общение идет по строгим правилам (официант принимает заказ по определенной форме).

Ключевые признаки:

- **Централизация:** Сервер — единая точка управления и хранения данных.
- **Постоянная доступность сервера:** Он должен работать 24/7.

- **Четкие роли:** Один предоставляет услугу (сервер), многие пользуются (клиенты).
- **Масштабируемость:** Можно увеличивать мощность сервера (больше процессоров, памяти) или добавлять новые серверы для обслуживания растущего числа клиентов.

Примеры: Веб-сайты (браузер-клиент запрашивает страницы с веб-сервера), электронная почта (почтовый клиент отправляет письмо через SMTP-сервер), онлайн-банкинг.

2.2. Одноранговые сети (P2P — Peer-to-Peer)

Базовое определение: Это децентрализованная архитектура, в которой участники сети (пиры, peers) равноправны. Каждый узел может выступать как в роли клиента (запрашивать данные), так и в роли сервера (предоставлять данные).

Суть и аналогия: Соседская барахолка или совместное чтение книги. Допустим, у группы друзей есть одна большая книга, которой нет в магазинах. Они договариваются: каждый фотографирует и распечатывает по одной главе, а потом все обмениваются главами. В итоге у каждого собирается полная книга. Каждый друг здесь — и клиент (получает главы от других), и сервер (раздает свою главу). Нет центрального «книжного магазина».

Ключевые признаки:

- **Децентрализация:** Нет выделенных серверов, все узлы равны.
- **Самоорганизация:** Сеть функционирует за счет прямых связей между участниками.
- **Использование ресурсов всех участников:** Пропускная способность и дисковое пространство каждого пира работают на общее дело.
- **Устойчивость:** Отключение даже многих узлов необязательно «положит» всю сеть, так как данные могут храниться в разных местах.

Примеры:

- **Файлообменные сети:** BitTorrent. Чтобы скачать фильм, вы подключаетесь к «рою» (swarm) таких же пользователей. Части файла (фрагменты) скачиваются одновременно с десятков других компьютеров, у которых эти части уже есть. При этом вы тоже начинаете раздавать уже скачанные фрагменты.
- **Некоторые мессенджеры и VoIP-сервисы** (например, Skype в ранних версиях) использовали P2P для прямого соединения абонентов.
- **Блокчейн-сети** (криптовалюты): каждый участник хранит копию всей цепочки транзакций.

Сравнение:

- **Клиент-сервер:** Удобен для управления, безопасности, централизованных обновлений. Риск — «единая точка отказа» (сервер).
- **P2P:** Эффективно использует ресурсы, масштабируется за счет роста числа пользователей, отказоустойчив. Риск — сложность управления, вопросы безопасности и авторского права.

3. Протоколы уровня приложений: Языки общения сервисов

Протокол — это строго определенный набор правил и форматов сообщений для обмена данными между сетевыми устройствами.

3.1. HTTP и HTTPS — Протоколы Всемирной паутины

HTTP (HyperText Transfer Protocol — Протокол передачи гипертекста):

- **Смысл:** Это «язык», на котором ваш браузер (клиент) разговаривает с веб-сервером, чтобы получить веб-страницу. Он работает по принципу «запрос-ответ».
- **Аналогия: Покупка в магазине самообслуживания.**

1. **Запрос (Request):** Вы (клиент) берете с полки товар (URL) и несете его к кассе. Ваш запрос: «Я хочу это!» (GET-запрос).
2. **Ответ (Response):** Кассир (сервер) пробивает товар и отдает его вам в пакете. Пакет — это ответ, содержащий сам товар (HTML-страницу, картинку) и чек (заголовки HTTP с информацией: тип содержимого, размер, дата).

- **Особенности:** HTTP **не сохраняет состояние** (stateless). Каждая покупка (запрос) для кассира — как первая. Он не помнит вас. Чтобы «помнить» (например, корзину покупок на сайте), используются дополнительные механизмы (куки, сессии).
- **Пример формата простого HTTP-запроса:**

```
GET /index.html HTTP/1.1
Host: www.example.com
User-Agent: Mozilla/5.0
```

(Клиент просит у сервера www.example.com файл index.html по протоколу HTTP версии 1.1).

HTTPS (HTTP Secure):

- **Смысл:** Это тот же HTTP, но работающий внутри защищенного «туннеля». Все данные между браузером и сервером шифруются.
- **Аналогия: Пересылка важного письма.** HTTP — это как открытка: текст видит любой почтальон (провайдер, хакер в сети). HTTPS — это как отправка письма в запечатанном сейф-пакете, который может открыть только адресат, имея специальный ключ (шифрование с помощью SSL/TLS).
- **Как распознать:** Адрес сайта начинается с https://, и в адресной строке браузера есть значок замка.

3.2. SMTP, POP3, IMAP — Протоколы электронной почты

Электронная почта — сложный сервис, в котором используются разные протоколы для разных этапов.

- **SMTP (Simple Mail Transfer Protocol):** Протокол для **отправки** почты и ее **пересылки** между почтовыми серверами.
  - **Аналогия: Почтовое отделение для отправки.** Вы пишете письмо, кладете в конверт, пишете адрес (адресата и обратный) и несете на почту (SMTP-сервер). Почта принимает письмо и отвечает за его дальнейшую маршрутизацию до почтового отделения адресата.
  - **Пример:** Ваша почтовая программа (Outlook) использует SMTP, чтобы отправить ваше письмо на SMTP-сервер вашего провайдера (например, smtp.gmail.com).
- **POP3 (Post Office Protocol v3):** Протокол для **загрузки** почты с сервера на ваш локальный компьютер (обычно с удалением с сервера).
  - **Аналогия: Получение почты из ящика домой.** Вы приходите в свое почтовое отделение, забираете все письма из своего абонентского ящика и уносите их домой. В ящике на почте писем не остается.
  - **Особенность:** Почта хранится на одном устройстве (ПК), доступ с других устройств затруднен, так как писем на сервере уже нет.
- **IMAP (Internet Message Access Protocol):** Протокол для **управления** почтой прямо на сервере.
  - **Аналогия: Работа с документами в банковской ячейке.** Вы приходите в банк (подключаетесь к серверу), открываете свою ячейку (почтовый ящик), просматриваете документы (письма), можете отсортировать их по папкам, удалить некоторые, но все оригиналы остаются в ячейке. Вы можете прийти в банк снова с другого входа (другого устройства) и увидеть всё в том же порядке.
  - **Преимущество:** Синхронизация состояния (прочитано/не прочитано, папки) между всеми вашими устройствами (телефон, ноутбук, планшет).

3.3. DNS (Domain Name System) — Система доменных имён

Базовое определение: Это «телефонная книга» Интернета. Она преобразует понятные человеку имена (например, google.com) в понятные компьютеру числовые IP-адреса (например, 142.250.185.78).

Главный смысл: Мы не помним номера телефонов (IP-адреса) всех сайтов. Мы помним имена (домены). DNS автоматически находит нужный номер.

Аналогия: Навигатор в автомобиле. Вы хотите попасть в кафе «У Васи». Вы не знаете его географических координат (широта/долгота — аналог IP-адреса). Вы вводите в навигатор название. Навигатор (DNS-система) обращается к своей базе данных и находит координаты этого кафе, после чего строит маршрут. Домен (vasya-cafe.ru) -> IP-адрес -> маршрутизация по сети.

Формат сообщений и иерархия DNS:

- **Иерархия (дерево справа налево):** www.subdomain.example.com.

1. **Корневая зона (.)** — невидимая точка в конце. Управляется корневыми серверами.
2. **Домен первого (верхнего) уровня (TLD):** .com, .ru, .org, .net.
3. **Домен второго уровня:** example.
4. **Субдомен (поддомен):** subdomain.
5. **Имя хоста:** www.

- **Процесс разрешения имени (рекурсивный запрос):**

1. Вы в браузере вводите example.com.
2. Ваш компьютер спрашивает у **локального DNS-резолвера** (обычно у вашего провайдера): «Где example.com?»
3. Резолвер, если не знает, идет спрашивать у **корневых серверов**: «Где .com?»
4. Корневой сервер отвечает: «Спроси у серверов .com, вот их адреса».
5. Резолвер спрашивает у **сервера .com**: «Где example.com?»
6. Сервер .com отвечает: «Вот IP-адрес авторитативного сервера для example.com».
7. Резолвер спрашивает у **авторитативного сервера example.com**: «Какой IP у example.com?»
8. Авторитативный сервер дает конечный ответ: 93.184.216.34.
9. Резолвер кэширует этот ответ и передает его вашему компьютеру.
10. Браузер устанавливает соединение с IP-адресом 93.184.216.34.

Утилита nslookup: Это командная строка для «ручного» опроса DNS. Позволяет диагностировать проблемы.

- **Пример:** Открываете командную строку (cmd), пишете nslookup google.com. Утилита покажет, какой IP-адрес она получила от DNS для этого домена, и какой сервер ей ответил.

3.4. DHCP (Dynamic Host Configuration Protocol)

Базовое определение: Это протокол, который **автоматически** назначает компьютеру (хосту) все необходимые сетевые настройки для работы в IP-сети.

Главный смысл: Избавить администратора и пользователя от ручного ввода IP-адреса, маски, шлюза и DNS-серверов на каждом устройстве.

Аналогия: Автоматическая выдача пропусков на предприятии. Вы новичок, заходите в здание (подключаетесь к сети Wi-Fi/Ethernet). Охранник (DHCP-сервер) автоматически:

1. Регистрирует вас (видит MAC-адрес вашего устройства).
2. Выдает вам уникальный пропуск (динамический IP-адрес из пула).
3. Сообщает правила: куда можно ходить (шлюз по умолчанию), как читать указатели (адреса DNS-серверов).

4. Указывает, на сколько выдан пропуск (время аренды адреса — lease time). Чтобы его продлить, нужно просто показаться охраннику снова (обновить аренду).

Процесс из 4 шагов (DORA):

1. **Discover** (Поиск): Компьютер кричит в сеть: «Есть тут DHCP-сервер? Мне нужны настройки!»
2. **Offer** (Предложение): DHCP-сервер откликается: «Я сервер! Вот тебе свободный IP-адрес, держи предварительное предложение».
3. **Request** (Запрос): Компьютер: «Отлично! Я хочу этот адрес, давай его мне!»
4. **Acknowledge** (Подтверждение): Сервер: «Хорошо, адрес 192.168.1.10 закреплен за тобой на 24 часа. Вот еще маска, шлюз и DNS».

3.5. FTP и SMB — Протоколы для работы с файлами

- **FTP (File Transfer Protocol):** Стандартный протокол для передачи файлов между клиентом и сервером по сети.
  - **Особенность:** Часто использует два соединения: **управляющее** (для команд, порт 21) и **данных** (для передачи самих файлов, порт 20 или динамический).
  - **Аналогия: Работа с грузчиками на складе.** Вы (клиент) находитесь на проходной (порт 21) и отдаете распоряжения кладовщику (серверу): «Принеси ящик №5!» (управляющее соединение). Кладовщик идет на склад, находит ящик и передает его вам через отдельную дверь для грузов (соединение данных, порт 20).
  - **Режимы:** Активный (сервер сам устанавливает соединение данных с клиента) и пассивный (клиент устанавливает оба соединения к серверу, что полезно за фаерволом).
- **SMB (Server Message Block) / CIFS:** Протокол для общего доступа к файлам, принтерам и другим ресурсам в локальных сетях, преимущественно под Windows.
  - **Смысл:** Обеспечивает прозрачный доступ к сетевым дискам, как к локальным.
  - **Аналогия: Общая полка в офисе.** На сервере есть папка \\Server\Projects. Вы на своем компьютере просто подключаете ее как сетевой диск Z:. Теперь все файлы на этой «полке» доступны всем, у кого есть права доступа, так, будто они лежат прямо на вашем компьютере.
  - **Применение:** «Сетевое окружение» в Windows, общие папки.

4. Современные концепции

4.1. Концепция «Всеобъемлющий Интернет» (Internet of Everything, IoE) и BYOD (Bring Your Own Device)

- **BYOD:** Политика, разрешающая сотрудникам использовать свои личные устройства (смартфоны, планшеты, ноутбуки) для работы с корпоративными данными и приложениями.

- **Плюсы:** Удобство для сотрудника, повышение мобильности и удовлетворенности.
- **Проблемы для сисадмина:** Безопасность (личное устройство может быть заражено), управление (как контролировать то, что вам не принадлежит?), разделение данных (личные и рабочие).
- **Аналогия:** Раньше компания выдавала всем одинаковые служебные автомобили (корпоративные ноутбуки). Теперь сотрудники могут приезжать на работу на своей машине (личный гаджет). Компании нужно обеспечить им пропуск на парковку (доступ к Wi-Fi) и, возможно, установить на их машины логотип фирмы (корпоративную почту и приложения), но при этом не иметь права обыскивать багажник (личные данные).

- **Всеобъемлющий Интернет (IoE):** Расширение идеи Интернета вещей (IoT). Это концепция, где в сеть объединены не только датчики и устройства, но и **люди, процессы и данные**. Цель — принимать более эффективные решения за счет сбора и анализа информации со всех этих источников.

- **Пример «умного города»:** Датчики на улицах (IoT) показывают плотность трафика (данные). Эта информация автоматически анализируется (процесс) и поступает в приложение для водителей (люди), которое предлагает альтернативный маршрут. Одновременно система регулирует работу светофоров (процесс), чтобы разгрузить пробку. Здесь взаимодействуют все четыре компонента IoE.

4.2. Доставка данных по конвергентным сетям

Базовое определение: Конвергентная сеть — это инфраструктура, которая передает по единой физической среде (чаще всего по IP-сети) различные типы трафика: данные, голос (телефония, VoIP), видео (веб-камеры, видеоконференции), который раньше требовал отдельных выделенных сетей.

Главный смысл: Экономия, упрощение управления и гибкость. Вместо трех «труб» (кабельной сети для интернета, телефонной линии и телевизионного кабеля) используется **одна «труба»** (одна IP-сеть), по которой идут все виды информации в виде пакетов.

Аналогия: Универсальная транспортная система. Раньше были отдельные пути для поездов (данные), канатные дороги для грузов (телефония) и воздушные трассы для самолетов (видео). Теперь построили единую высокоскоростную автомагистраль (IP-сеть). По ней едут все: и грузовики с контейнерами (пакеты данных), и автобусы с пассажирами (оцифрованные голосовые пакеты), и трейлеры с кинооборудованием (видеопоток). Для приоритетного трафика (например, экстренной связи) есть выделенные полосы (технологии QoS — Quality of Service, гарантия качества обслуживания).

Примеры конвергенции:

- **VoIP (Voice over IP):** Обычный телефонный разговор оцифровывается и передается как поток IP-пакетов через интернет.
- **IPTV:** Телевизионный сигнал доставляется абоненту по протоколу IP.

- **Единая кабельная система в офисе:** По одному Ethernet-кабелю к рабочему месту подводится и интернет, и IP-телефония.

Заключение

Уровень приложений — это вершина айсберга, с которой взаимодействует пользователь. Понимание его протоколов и архитектур (клиент-сервер, P2P) является фундаментом для системного администратора. Это знание позволяет не только настраивать сервисы (DNS, DHCP, веб, почту), но и грамотно диагностировать проблемы («почему не открывается сайт?», «почему не приходит почта?»), планировать сетевую инфраструктуру и адаптироваться к современным тенденциям, таким как конвергенция и массовое использование мобильных персональных устройств в корпоративной среде. Каждый протокол — это инструмент в арсенале сисадмина, и эффективность работы всей сети во многом зависит от того, насколько правильно и к месту эти инструменты применяются.

Контрольные вопросы к ТЕМЕ-6

1. Какие три уровня модели OSI объединяет в себе уровень приложений модели TCP/IP?
2. В чем главное отличие архитектуры «клиент-сервер» от одноранговой (P2P) сети? Приведите по одному примеру для каждой архитектуры.
3. Какую основную функцию выполняет протокол DNS? Приведите бытовую аналогию для объяснения его работы.
4. В чем ключевая разница между протоколами POP3 и IMAP для доступа к электронной почте?
5. Что означает аббревиатура HTTPS и чем этот протокол принципиально отличается от HTTP?
6. Опишите кратко процесс DORA, который выполняет протокол DHCP при подключении устройства к сети.
7. Какую основную задачу решает концепция BYOD (Bring Your Own Device) и какую основную проблему для администратора она создает?
8. Для чего используется протокол SMTP в системе электронной почты?
9. Что позволяет определить или проверить с помощью утилиты командной строки nslookup?
10. Какой протокол чаще всего ассоциируется с организацией общих папок и сетевых принтеров в среде Windows? (SMB/FTP)
11. Что общего между протоколами FTP и SMB, и в чем их основное практическое различие для пользователя?
12. Что означает термин «конвергентная сеть»? Приведите один пример конвергенции в современной сети.

Практические задания по ТЕМЕ-6

Задание 1. Анализ архитектуры сервиса

Ситуация: В небольшом российском интернет-магазине «Уютный дом» (uyutniydom.ru) принимают решение о способе реализации чата для оперативной поддержки покупателей на сайте. Один разработчик предлагает использовать готовое P2P-решение, аргументируя это отказоустойчивостью и экономией на серверных мощностях. Другой настаивает на классической клиент-серверной модели.

- Вопрос: Какой аргумент, основанный на ключевом различии архитектур, должен привести сторонник клиент-серверной модели, чтобы его позиция выглядела убедительнее для задачи онлайн-поддержки? (Подсказка: подумайте, кто должен «видеть» всю историю переписки и управлять ею).

Задание 2. Диагностика проблемы с электронной почтой

Ситуация: Сотрудник отдела кадров компании «СеверСталь» в Мурманске жалуется системному администратору: «Настроил почту в Outlook на рабочем ПК, все письма скачал. Теперь захожу с корпоративного телефона — ящик пустой, и на телефоне нет тех писем, что есть на компьютере». Администратор сразу понимает вероятную причину.

- Вопрос: Какой протокол для получения почты (POP3 или IMAP), скорее всего, использован в настройках Outlook на ПК? Дайте краткое объяснение, почему это привело к описанной проблеме.

Задание 3. Поиск причины недоступности сайта

Ситуация: Пользователь из Казани звонит в службу поддержки провайдера «ВолгаТелеком» и сообщает, что у него не открывается популярный сайт знакомств love.ru, при этом другие сайты работают. Специалист поддержки со своего компьютера в сети компании сайт открывает. Он решает проверить цепочку доставки информации.

- Вопрос: Какую простую утилиту командной строки должен посоветовать специалист пользователю запустить на своём компьютере, чтобы проверить, преобразуется ли доменное имя love.ru в IP-адрес? Какой результат этой проверки укажет на проблему именно с преобразованием имени, а не с самой сетью?

Задание 4. Выбор протокола для корпоративного файлообмена

Ситуация: В проектной организации «МостСтрой» в Санкт-Петербурге рабочие чертежи и сметы хранятся на центральном файловом сервере. Инженерам требуется постоянный доступ к этим файлам с возможностью их редактирования прямо с сетевой папки. Администратор рассматривает два протокола: FTP и SMB.

- Вопрос: Какой протокол (FTP или SMB) лучше подходит для описанной задачи и почему? (Подсказка: вспомните аналогию с «грузчиками на складе» и «общей полкой в офисе»).

Задание 5. Планирование сети для нового офиса

Ситуация: Системному администратору поручено спланировать IT-инфраструктуру для нового филиала банка в городе Тюмень. Будет проведена структурированная кабельная

система (СКС). Помимо компьютеров, необходимо подключить IP-телефоны для внутренней связи и камеры видеонаблюдения, которые также передают данные по сети.

- Вопрос: Как называется подход, при котором голос (телефония), видео (камеры) и данные (работа компьютеров) передаются по одной физической IP-сети? Какой ключевой механизм (сокращение из трёх букв) необходимо будет настроить на сетевом оборудовании, чтобы голосовая связь имела наивысший приоритет и не прерывалась из-за загрузки канала файловыми передачами?

ТЕМА-7. IP-адресация

Введение

В мире компьютерных сетей, где миллиарды устройств обмениваются информацией, необходима точная и однозначная система адресации, подобная почтовой. IP-адресация — это фундамент такой системы. Она позволяет уникально идентифицировать каждое сетевое устройство (узел) и обеспечивает логическую маршрутизацию данных через сложную паутину сетей. Данный материал посвящён четвертой версии Интернет-протокола (IPv4), которая, несмотря на появление IPv6, остается широко используемой и является ключевой для понимания основ сетевого взаимодействия. Мы детально разберём структуру адресов, их классификацию, методы назначения и вспомогательные сервисы, без которых функционирование современных сетей было бы невозможным.

1. Структура IPv4-адресов. Сетевая и узловая часть IP-адреса

Фундаментальная идея: IPv4-адрес — это уникальный 32-битный (четырёхбайтный) логический идентификатор, назначаемый устройству для его участия в сетевом обмене. Он состоит из двух иерархических частей: **сетевой** и **узловой (хостовой)**. Это разделение аналогично структуре почтового адреса: сначала указывается город и улица (сетевая часть — определяет «район»), а затем номер дома и квартиры (узловая часть — определяет конкретное «здание» внутри района).

Пояснение смысла: Зачем нужно такое разделение? Представьте, что маршрутизатор (сетевой «почтамт») получает пакет данных. Если бы адреса были плоскими (просто последовательность чисел без структуры), маршрутизатору пришлось бы хранить гигантскую таблицу с информацией о пути ко всем возможным адресам в Интернете, что нереально. Благодаря сетевой части, маршрутизатор смотрит только на «сетевой префикс» адреса назначения. Он определяет: «Ага, этот пакет предназначен для сети 192.168.1.0. Я знаю путь к этой сети (она прямо подключена ко мне или путь к ней лежит через соседний маршрутизатор)». Достигнув целевой сети (например, вашей домашней Wi-Fi сети), последнее устройство (коммутатор или маршрутизатор) смотрит на **узловую часть**, чтобы определить, какому конкретному устройству (ноутбуку, телефону, принтеру) доставить пакет.

Пример и аналогия:

- **Полный IPv4-адрес:** 192.168.1.10
- **Аналогия с почтовым адресом:** «Россия, Москва, ул. Тверская, д. 10».
  - Россия, Москва, ул. Тверская — это **сетевая часть**. Эта информация используется для доставки письма в нужный город и на нужную улицу.

- д. 10 — это **узловая часть**. Эта информация используется почтальоном на улице Тверской, чтобы опустить письмо в конкретный почтовый ящик.
- Внутри сети 192.168.1.0 могут находиться устройства с адресами 192.168.1.1, 192.168.1.2, ..., 192.168.1.254. Их сетевая часть (192.168.1) одинакова, а узловая (.1, .2, ...) различается. Для внешнего мира (Интернета) все эти устройства представляются одной сетью 192.168.1.0.

Граница между сетевой и узловой частями **не фиксирована** и определяется **маской подсети**.

2. Преобразование адресов между двоичным и десятеричным представлением

Фундаментальная идея: Компьютеры и сетевое оборудование работают с двоичными числами (битами, 0 и 1). IPv4-адрес — это изначально 32 бита. Для удобства восприятия человеком эти 32 бита разбиваются на 4 группы по 8 бит (октеты), и каждый октет преобразуется в десятичное число от 0 до 255. Запись вида 192.168.1.10 называется **точечно-десятичной нотацией (dotted-decimal)**.

Пояснение смысла: Понимание двоичного представления критически важно для работы с масками подсетей, расчета сетевых адресов и диагностики проблем. Это язык, на котором «думают» сетевые устройства.

Метод преобразования:

1. **Из десятичного в двоичное:** Каждое десятичное число октета (0-255) преобразуется в 8-битное двоичное число. Самый простой способ — деление на 2 с записью остатков в обратном порядке.

- Пример для октета 192:

- 192 / 2 = 96, остаток **0**
- 96 / 2 = 48, остаток **0**
- 48 / 2 = 24, остаток **0**
- 24 / 2 = 12, остаток **0**
- 12 / 2 = 6, остаток **0**
- 6 / 2 = 3, остаток **0**
- 3 / 2 = 1, остаток **1**
- 1 / 2 = 0, остаток **1**

- Читаем остатки **снизу вверх**: 11000000. Это 8 бит, 192 в двоичном виде.

2. **Из двоичного в десятичное:** Каждая позиция в 8-битном октете имеет вес: слева направо 128, 64, 32, 16, 8, 4, 2, 1. Складываем веса тех позиций, где стоят единицы.

- Пример для октета 11000000:

- Позиции: 128(1), 64(1), 32(0), 16(0), 8(0), 4(0), 2(0), 1(0).
- Сумма: 128 + 64 = **192**.

Развернутый пример для всего адреса:

- IPv4-адрес: 192.168.1.10
- Двоичное представление (по октетам):
  - 192 -> 11000000
  - 168 -> 10101000
  - 1 -> 00000001
  - 10 -> 00001010
- Полная 32-битная строка: **11000000.10101000.00000001.00001010**.

Аналогия: Представьте, что у вас есть 32 ячейки (бита), каждая может быть либо темной (0), либо светлой (1). Чтобы не записывать длинную строку из 0 и 1, мы группируем ячейки по 8 и для каждой группы вычисляем «суммарную яркость» (десятичное число от 0 до 255). Так получается компактная и понятная человеку запись.

3. Маска подсети IPv4

Фундаментальная идея: Маска подсети — это 32-битное число, которое **однозначно определяет границу** между сетевой и узловой частями IP-адреса. Она «накладывается» на IP-адрес, как трафарет: биты маски, равные 1, указывают на биты IP-адреса, принадлежащие сетевой части; биты маски, равные 0, указывают на биты узловой части.

Пояснение смысла: Без маски адрес 192.168.1.10 был бы просто числом. Маска придает ему смысл, указывая, сколько бит слева отвечает за идентификацию сети. Она позволяет делить большие сетевые пространства на меньшие подсети (субнетинг) и агрегировать маршруты (суммаризация).

Представление:

- Записывается в том же точечно-десятичном формате, что и IP-адрес.
- Всегда представляет собой последовательность идущих подряд 1, а затем подряд 0.
- **Пример стандартной маски для «класса С»:** 255.255.255.0.
  - Двоичный вид: 11111111.11111111.11111111.00000000
  - Это означает, что первые 24 бита IP-адреса — сетевая часть, последние 8 бит — узловая.

Совместное использование с IP-адресом (аналогия с трафаретом):

- IP-адрес: 192.168.1.10 -> 11000000.10101000.00000001.00001010
- Маска: 255.255.255.0 -> 11111111.11111111.11111111.00000000
- Применяем логическую операцию **И (AND)** между соответствующими битами адреса и маски:
  - 1 AND 1 = 1
  - 1 AND 0 = 0
  - 0 AND 1 = 0
  - 0 AND 0 = 0
- Результат (сетевой адрес): 11000000.10101000.00000001.00000000 -> 192.168.1.0

- Биты, где в маске были 1, «проглянули» биты IP-адреса.
- Биты, где в маске были 0, «закрыли» биты IP-адреса, обнулив их.

Сокр. запись (CIDR): Вместо 192.168.1.10 255.255.255.0 часто пишут 192.168.1.10/24. Число /24 — это длина префикса, количество единичных бит в маске.

4. Сетевой адрес, адрес узла и широковещательный адрес сети IPv4

В каждой подсети есть три специальных адреса, которые **не могут быть назначены конкретному устройству (хосту)**.

1. Сетевой адрес (Network Address) или адрес подсети:

- **Определение:** Первый адрес в подсети. Все биты узловой части равны 0.
- **Назначение:** Служит «именем» или идентификатором всей подсети. Это адрес «улицы», а не конкретного дома.
- **Пример для 192.168.1.0/24:** 192.168.1.0 (узловая часть .0 в двоичном виде 00000000).

2. Широковещательный адрес (Broadcast Address):

- **Определение:** Последний адрес в подсети. Все биты узловой части равны 1.
- **Назначение:** Пакет, отправленный на этот адрес, будет **доставлен всем узлам** в данной подсети. Это аналог громкоговорителя на улице, который слышат все жители.
- **Пример для 192.168.1.0/24:** 192.168.1.255 (узловая часть .255 в двоичном виде 11111111).

3. Адреса узлов (Host Addresses):

- **Определение:** Все адреса между сетевым и широковещательным. Это адреса, которые присваиваются конечным устройствам.
- **Пример для 192.168.1.0/24:**
  - Диапазон адресов узлов: 192.168.1.1 — 192.168.1.254.
  - Всего доступных адресов для устройств: 254 (из 256 возможных комбинаций 8 бит вычитаем 2 служебных адреса).

Расчет на примере другой подсети: Допустим, сеть 10.0.8.0/22 (маска 255.255.252.0).

- Маска /22 означает, что под узловую часть отведено 32-22=10 бит.
- **Сетевой адрес:** Обнуляем все 10 бит узловой части. 10.0.8.0.
- **Широковещательный адрес:** Устанавливаем все 10 бит узловой части в 1. После расчета: 10.0.11.255.
- **Диапазон адресов узлов:** 10.0.8.1 — 10.0.11.254.

Важная аналогия: Представьте гостиницу «Сетевая» с 256 номерами (от 0 до 255).

- **Номер 0 (192.168.1.0)** — это не гостиничный номер, а вывеска на здании, его официальное название (сетевой адрес).
- **Номера 1-254 (192.168.1.1-254)** — это обычные номера для постояльцев (адреса узлов).
- **Номер 255 (192.168.1.255)** — это система оповещения по всем номерам. Если отправить сообщение в этот «номер», его услышат во всех реальных номерах (широковещательный адрес).

5. Присвоение узлу статического и динамического IPv4-адреса

Фундаментальная идея: IP-адрес должен быть корректно настроен на сетевом интерфейсе устройства. Это можно сделать вручную (статически) или автоматически (динамически) с помощью специального протокола.

1. Статическая адресация (Static Assignment):

- **Суть:** Адрес, маска, шлюз по умолчанию и DNS-серверы вводятся администратором вручную в настройки сетевой карты устройства.
- **Преимущества:** Постоянство, предсказуемость, полный контроль. Идеально для серверов, сетевых принтеров, критичного оборудования.
- **Недостатки:** Трудоемкость администрирования в больших сетях, высокая вероятность человеческой ошибки (конфликта адресов).
- **Пример:** Вы заходите в «Центр управления сетями» Windows, выбираете «Свойства» подключения, затем «Протокол IPv4», выбираете «Использовать следующий IP-адрес» и вводите: Адрес: 192.168.1.100, Маска: 255.255.255.0, Шлюз: 192.168.1.1.

2. Динамическая адресация (Dynamic Assignment):

- **Суть:** Адрес автоматически назначается устройству на ограниченное время (аренда) специальным сервером.
- **Основной протокол: DHCP (Dynamic Host Configuration Protocol)**. Это «арендное бюро» для IP-адресов.
- **Процесс (упрощенно):**

1. **Discover:** Устройство (клиент), подключившись к сети, кричит: «Есть тут DHCP-сервер? Мне нужен адрес!» (широковещательный пакет).
2. **Offer:** DHCP-сервер отвечает: «Я сервер. Предлагаю тебе адрес 192.168.1.50 на 24 часа» (широковещательный или unicast пакет).
3. **Request:** Клиент: «Отлично, я принимаю адрес 192.168.1.50!»
4. **Acknowledge:** Сервер: «Подтверждаю аренду адреса 192.168.1.50. Вот еще маска, шлюз и DNS».

- **Преимущества:** Автоматизация, экономия времени, отсутствие конфликтов адресов, эффективное использование пула адресов.

- **Недостатки:** Зависимость от доступности DHCP-сервера, непостоянство адреса (может меняться).
- **Аналогия:** Статический адрес — как купленная в собственность квартира (адрес постоянный). Динамический адрес (DHCP) — как номер в гостинице: вам его выдают при заселении на определенный срок, а при выезде он освобождается для другого гостя.

6. Многоадресная передача (Multicast)

Фундаментальная идея: Multicast — это метод рассылки данных **определенной группе** получателей в IP-сети. Это компромисс между Unicast (один-к-одному, нагрузка на источник) и Broadcast (один-ко-всем, нагрузка на всех, даже незаинтересованных).

Пояснение смысла: Неэффективно, чтобы видеосервер потокового вещания отправлял отдельную копию видеопотока тысячам зрителей (Unicast). Также нельзя использовать Broadcast, так как тогда трафик получили бы все устройства в сети, включая датчики и принтеры, и это привело бы к коллапсу. Multicast решает эту проблему: сервер отправляет **единственный поток** на специальный групповой адрес, а сетевые маршрутизаторы и коммутаторы **копируют и доставляют** этот поток только тем устройствам, которые явно «подписались» на эту группу.

Диапазон адресов Multicast (IPv4): 224.0.0.0 — 239.255.255.255.

- **Примеры известных групп:**
  - 224.0.0.1 — все узлы в локальной сети (аналог ограниченного широковещательного запроса).
  - 224.0.0.2 — все маршрутизаторы в локальной сети.
  - 224.0.0.9 — протокол маршрутизации RIPv2.
  - 239.255.255.250 — протокол обнаружения служб SSDP (используется UPnP).

Аналогия: Радиостанция. Она передает сигнал на одной частоте (multicast-адресе). Только те радиоприемники (сетевые устройства), которые настроены на эту частоту (присоединились к multicast-группе), будут принимать и обрабатывать передачу. Все остальные устройства её игнорируют.

7. Публичные и частные IPv4-адреса. IPv4-адреса специального назначения. Присвоение IP-адресов.

1. Публичные (глобальные) адреса (Public Addresses):

- **Определение:** Адреса, которые используются для маршрутизации в глобальной сети Интернет. Они должны быть **уникальными** во всем мире.
- **Назначение:** Выделяются интернет-провайдерам (ISP) региональными интернет-регистраторами (RIR). Провайдер, в свою очередь, выдает их своим клиентам.

- **Аналогия:** Официальный почтовый адрес здания, который зарегистрирован в государственном реестре и по которому можно доставить письмо из любой точки мира.

2. Частные (приватные) адреса (Private Addresses):

- **Определение:** Адреса, зарезервированные для использования в **закрытых (локальных) сетях**. Они **не маршрутизируются** в Интернете. Это означает, что маршрутизаторы в Интернете должны отбрасывать пакеты с такими адресами в поле назначения.
- **Диапазоны (определены в RFC 1918):**
  - 10.0.0.0 — 10.255.255.255 (маска /8, 1 сеть класса A, 16+ млн адресов)
  - 172.16.0.0 — 172.31.255.255 (маска /12, 16 сетей класса B, 1+ млн адресов)
  - 192.168.0.0 — 192.168.255.255 (маска /16, 256 сетей класса C, 65 тыс. адресов)
- **Назначение:** Используются в домашних сетях, офисах, предприятиях. Позволяют создавать внутреннюю адресацию без необходимости получать уникальные публичные адреса для каждого устройства.
- **Выход в Интернет:** Осуществляется через технологию **NAT (Network Address Translation)**, которую выполняет маршрутизатор (роутер). Он «подменяет» частный адрес устройства на свой публичный адрес при отправке пакета в Интернет и выполняет обратную подмену при получении ответа.
- **Аналогия:** Внутренняя нумерация кабинетов в большом бизнес-центре. У каждого кабинета есть внутренний номер (например, каб. 305 — частный адрес 192.168.1.5). Для внешней почты используется единый адрес здания (публичный IP роутера). Секретарь на reception (NAT-маршрутизатор) принимает всю почту, смотрит, для какого кабинета она, и доставляет внутрь.

3. Адреса специального назначения:

- **Адрес текущего узла (Loopback Address):** 127.0.0.1 (сеть 127.0.0.0/8). Используется для тестирования сетевого стека самого устройства. Пакет, отправленный на этот адрес, никогда не покидает компьютер. Аналог разговора с самим собой по рации для проверки её работы.
- **Сетевой адрес 0.0.0.0:** В конфигурациях обозначает «любой адрес» или адрес по умолчанию. Например, маршрут 0.0.0.0/0 — маршрут по умолчанию (шлюз «последней надежды»).
- **Ограниченный широковещательный адрес (Limited Broadcast):** 255.255.255.255. Широковещание **только в пределах своей локальной сети** (broadcast domain). Не пересылается маршрутизаторами.
- **Адреса для локальной автономной конфигурации (APIPA/Link-Local):** 169.254.0.0/16. Назначаются автоматически, если устройство не смогло получить адрес от DHCP-сервера. Позволяет организовать связь только в пределах локального сегмента сети.

Присвоение IP-адресов (общий принцип):

1. **Планирование:** Определяются размеры сетей, количество узлов, необходимость в публичных адресах.
2. **Выбор диапазона:** Для внутренней сети выбирается подходящий диапазон частных адресов (обычно 192.168.x.0/24 для небольших сетей).

3. **Определение маски:** Рассчитывается маска подсети, исходя из планируемого количества узлов в каждом сегменте.
4. **Резервирование адресов:** В пуле выделяются статические адреса для серверов, принтеров, сетевого оборудования.
5. **Настройка сервисов:** Настраивается DHCP-сервер для выдачи динамических адресов из оставшегося пула.
6. **Настройка NAT:** На граничном маршрутизаторе настраивается трансляция частных адресов в публичный(ые).

8. ICMP-сервисы. Отличия для протоколов IPv4. Сообщения ICMPv4 «Запрос к маршрутизатору», «Объявление от маршрутизатора», «Запрос соседнего узла» и «Объявление соседнего узла».

Фундаментальная идея: ICMP (Internet Control Message Protocol) — это вспомогательный протокол сетевого уровня, предназначенный для **обмена служебными и диагностическими сообщениями** между сетевыми устройствами. Он не переносит данные приложений, а используется для сообщения об ошибках (например, хост недоступен) и выполнения диагностических операций (ping, traceroute). ICMPv4 работает поверх IPv4.

Ключевые сообщения ICMPv4 и их аналогии:

- **Echo Request / Echo Reply (Эхо-запрос/Эхо-ответ):** Основа команды ping. «Ты жив?» — «Да, жив!».
- **Destination Unreachable (Адресат недостижим):** Маршрутизатор или хост уведомляет отправителя, что пакет не может быть доставлен (нет маршрута, порт закрыт и т.д.).
- **Time Exceeded (Время истекло):** Генерируется, когда TTL (Time To Live) пакета достигает нуля. Критично для работы tracert.
- **Source Quench (Подавление источника):** Устаревшее сообщение для контроля перегрузки. «Пожалуйста, отправляй помедленнее».
- **Redirect (Перенаправление):** Маршрутизатор сообщает хосту: «Для этого адреса есть лучший путь, отправляй пакеты не мне, а вот на тот маршрутизатор».

Сообщения для обнаружения соседей (важны для IPv4, но критичны для IPv6):
Эти сообщения помогают устройствам автоматически обнаруживать друг друга в локальной сети без использования протокола ARP (который есть только в IPv4).

1. **Router Solicitation (RS) — «Запрос к маршрутизатору»:** Когда узел включается в сеть, он отправляет широковещательное сообщение RS (224.0.0.2), спрашивая: «Есть ли в этой сети маршрутизаторы? Пожалуйста, отзовитесь!».
2. **Router Advertisement (RA) — «Объявление от маршрутизатора»:** Маршрутизатор периодически или в ответ на RS отправляет широковещательное сообщение RA (224.0.0.1). В нём содержится информация о себе, о сетевом префиксе, настройках автоконфигурации и шлюзе по умолчанию. Это ключевой механизм для **бесстатистической автоматической настройки адресов (SLAAC)** в IPv6, а в IPv4 может использоваться для дополнения DHCP.
3. **Neighbor Solicitation (NS) — «Запрос соседнего узла»:** Аналог ARP-запроса в IPv4, но более универсальный. Узел отправляет запрос на multicast-адрес, производный от IPv4-

адреса целевого устройства, чтобы узнать его MAC-адрес. «Устройство с адресом 192.168.1.5, какой у тебя MAC-адрес?».

4. **Neighbor Advertisement (NA) — «Объявление соседнего узла»:** Ответ на NS. Узел сообщает свой MAC-адрес. «Я 192.168.1.5, мой MAC-адрес AA:BB:CC:11:22:33».

Аналогия: Представьте новый жилой комплекс.

- **RS/RA:** Новый житель (узел) выходит на балкон и кричит: «Есть ли тут управляющий (маршрутизатор)?». Управляющий выходит и объявляет всем: «Я управляющий, адрес комплекса такой-то, мой офис тут-тут, правила такие-то».
- **NS/NA:** Житель кв.101 хочет передать посылку жителю кв.205, но не знает его в лицо (MAC-адрес). Он идет к двери 205 и стучит (NS). Житель 205 открывает и представляется (NA).

9. Тестирование сети с помощью эхо-запросов. Трассировка маршрута. Время прохождения сигнала в прямом и обратном направлениях (RTT). Время жизни (TTL) IPv4 и предел переходов IPv4.

1. Ping (Эхо-запросы):

- **Суть:** Утилита, использующая пару сообщений ICMP **Echo Request** и **Echo Reply**.
- **Цель:** Проверка **достижимости** узла и **качества** соединения (задержки, потери пакетов).
- **Ключевой параметр — RTT (Round-Trip Time):** Время от момента отправки Echo Request до момента получения Echo Reply. Измеряется в миллисекундах (мс). Показывает общую задержку «туда и обратно».

- **Пример вывода ping 8.8.8.8:**

```
Ответ от 8.8.8.8: число байт=32 время=45мс TTL=54
Ответ от 8.8.8.8: число байт=32 время=47мс TTL=54
Здесь время=45мс — это RTT.
```

- **Аналогия:** Эхолот. Вы посылаете звуковой импульс (Echo Request) и замеряете время, через которое вернется эхо (Echo Reply). По времени судите о расстоянии до дна (задержке до узла).

2. Traceroute / Tracert (Трассировка маршрута):

- **Суть:** Утилита для определения пути, который проходят пакеты от источника до назначения.
- **Принцип работы (гениальный и простой):** Основан на полях **TTL (Time To Live)** в заголовке IPv4 и сообщении ICMP **Time Exceeded**.

- **TTL (Предел переходов):** Это счетчик «скачков» (hop). Каждый маршрутизатор, обрабатывающий пакет, уменьшает TTL на 1. Если TTL становится равен 0, маршрутизатор **отбрасывает пакет** и отправляет обратно отправителю сообщение ICMP **Time Exceeded** с указанием своего адреса.
- **Алгоритм tracert:**

1. Отправляется первый пакет (часто UDP или ICMP Echo) с **TTL=1**. Первый же маршрутизатор уменьшает TTL до 0, отбрасывает пакет и шлет обратно Time Exceeded. Мы узнаём адрес первого маршрутизатора.
2. Отправляется второй пакет с **TTL=2**. Он проходит первый маршрутизатор (TTL=1), достигает второго, там TTL становится 0, и второй маршрутизатор шлет Time Exceeded.
3. Процесс повторяется, пока пакет не достигнет целевого узла. Целевой узел отвечает сообщением **Destination Unreachable (Port Unreachable)** для UDP или **Echo Reply** для ICMP.

- **Вывод трассировки:** Показывает цепочку маршрутизаторов с их IP-адресами и, часто, с тремя замерами RTT для каждого «скачка».

```
tracert ya.ru

 1     2 ms     2 ms     1 ms  router.home [192.168.1.1]
 2    10 ms     9 ms    10 ms  10.100.10.1
 3    12 ms    11 ms    12 ms  core-router.city.isp.net [195.14.10.254]
 ... и так далее до цели.
```

- **Аналогия:** Игра в «глухой телефон» с ограниченным числом передач. Вы отправляете сообщение и говорите: «Передай не более чем 1 человеку» (TTL=1). Оно доходит до первого человека (маршрутизатора 1) и «умирает», а этот человек сообщает вам своё имя. Затем вы отправляете то же сообщение с лимитом «2 человека» (TTL=2), узнаете второго участника цепочки, и так далее, пока сообщение не дойдет до конечного адресата.

Важность TTL: Помимо использования в трассировке, TTL предотвращает бесконечную циркуляцию пакетов по сети (петли маршрутизации) – это «срок годности» пакета. Типичные начальные значения: 64 (Linux), 128 (Windows), 255 (сетевые устройства).

Заключение

IP-адресация IPv4 — это стройная и логичная система, являющаяся краеугольным камнем современных компьютерных сетей. Понимание структуры адреса (сетевая/узловая часть), умение работать с масками и вычислять параметры подсети, знание различий между типами адресов (публичные, частные, специальные) и методов их назначения – обязательные компетенции специалиста по системному и сетевому администрированию. Сервисы ICMP, в свою очередь, предоставляют незаменимый инструментарий для диагностики и поддержания работоспособности сети, позволяя проверять доступность узлов, определять пути следования трафика и обнаруживать проблемы на маршруте. Освоив этот материал, вы заложите прочный фундамент для дальнейшего изучения более сложных тем, таких как динамическая маршрутизация, IPv6 и безопасность сетей.

Контрольные вопросы к ТЕМЕ-7

1. Что такое IPv4-адрес и из скольких битов он состоит?
2. На какие две логические части делится IP-адрес и какова их функция?
3. Что такое маска подсети и для чего она нужна?
4. Какой адрес в подсети называется сетевым, а какой — широковещательным? Можно ли их назначать устройствам?
5. В чём заключается основное отличие статического и динамического (DHCP) способов назначения IP-адреса?
6. Какой диапазон IPv4-адресов зарезервирован для использования в частных (локальных) сетях?
7. С какой целью используется IP-адрес 127.0.0.1 (адрес петли обратной связи)?
8. Что такое многоадресная рассылка (multicast) и какой диапазон адресов для неё используется?
9. Для чего предназначен протокол ICMP? Приведите примеры двух основных типов его сообщений.
10. Как работает утилита ping и что показывает параметр RTT в её результатах?
11. Какой параметр в заголовке IPv4-пакета используется утилитой tracert для определения пути и зачем он нужен?
12. Что происходит с пакетом, когда значение его поля TTL (время жизни) достигает нуля?

Практические задания по ТЕМЕ-7

Задание 1. Диагностика домашней сети (Частные адреса, DHCP, маска)

Ситуация: Вы помогаете соседу разобраться с его домашней сетью в новостройке в г. Краснодаре. Он жалуется, что ноутбук подключается к Wi-Fi роутера, но «Интернет не работает». Вы заходите в настройки сетевого подключения ноутбука и видите следующую конфигурацию IPv4:

- IP-адрес: 169.254.12.45
- Маска подсети: 255.255.0.0
- Основной шлюз: не указан.
- DNS-серверы: не указаны.

Вопросы:

1. Объясните, что означает адрес 169.254.12.45? Как он появился на ноутбуке?
2. Почему при такой конфигурации Интернет не работает, даже если кабель от провайдера «Ростелеком» подключен к роутеру и на нем горят все лампочки?
3. Какие два простейших действия (одно на ноутбуке, одно с оборудованием) вы предложите выполнить для решения проблемы, исходя из информации в теоретическом материале?

Задание 2. Планирование адресного пространства небольшого офиса (Сетевая/узловая часть, маска, диапазоны)

Ситуация: В небольшой московской компании «ВегаСтрой» открывается новый офис на 8 сотрудников. Им нужно организовать локальную сеть. Имеется: 8 компьютеров, 1 сетевой принтер, 1 IP-телефон, 1 точка доступа Wi-Fi и маршрутизатор (роутер). Все устройства должны быть в одной сети. Администратор решил использовать частный адресный диапазон.

Вопросы:

1. Какой из частных диапазонов (10.x.x.x, 172.16.x.x или 192.168.x.x) наиболее логично и просто использовать в этой ситуации? Объясните почему.
2. Выбран диапазон 192.168.15.0. Какая стандартная маска подсети (например, /24, /25 и т.д.) подойдет для размещения всех 11 устройств с большим запасом? Укажите её в формате /XX.
3. Приняв выбранную вами маску, назовите:

- Сетевой адрес этой подсети.
- Широковещательный адрес.
- Диапазон адресов, которые можно назначить устройствам.

Задание 3. Анализ сетевого пакета (Двоичный/десятичный вид, сетевой адрес)

Ситуация: Вы изучаете логи сетевого анализатора (sniffer) в учебном классе колледжа. В логе видна попытка подключения с адресом 11000000.10101000.00000001.01100101 на адрес 192.168.1.255.

Вопросы:

1. Переведите исходный IP-адрес (бинарный) в привычный десятичный формат.
2. К какому типу передачи (unicast, multicast, broadcast) относится пакет, отправленный на адрес 192.168.1.255? Объясните, что это значит.
3. Предположим, маска сети в этом сегмере стандартная для таких адресов (/24). Определите, в одной ли сети находятся источник (11000000.10101000.00000001.01100101) и получатель (192.168.1.255) пакета? Ответ обоснуйте, рассчитав сетевые адреса для обоих.

Задание 4. Диагностика связи с удаленным сервером (Ping, Tracert, TTL)

Ситуация: Сотрудник филиала компании в Екатеринбурге жалуется на нестабильную связь с корпоративным сервером в Москве. Вы как администратор начали диагностику.

Шаг 1. Вы выполнили команду ping msk-srv.vega.ru. Было отправлено 4 пакета. Результат: 3 пакета вернулись с временем ~95 мс, 1 пакет был потерян.
Шаг 2. Затем вы выполнили tracert msk-srv.vega.ru. На 5-м «прыжке» (hop) вы увидели три звездочки * * * (таймаут), а на 6-м и далее — снова нормальные ответы с возрастающим временем.

Вопросы:

1. Что показывает команда ping в данном случае? Как интерпретировать потерю 1 пакета из 4 и значение времени ~95 мс?

2. Какой принцип работы tracert позволил вам обнаружить конкретный проблемный участок пути (5-й хоп)?
3. Объясните, почему после «зависшего» 5-го хопа трассировка продолжилась и достигла цели? Что это может сказать о проблеме (полный отказ устройства или просто его настройки)?

Задание 5. Выбор схемы адресации для учебного класса (Статика/DHCP, публичные/частные)

Ситуация: В кабинете информатики «Лицея №1» г. Казани находятся 20 стационарных компьютеров, объединенных в сеть. Кабинет имеет выделенный канал связи с Интернетом через школьный сервер. Необходимо выбрать схему адресации.

Вариант А: На каждом компьютере вручную прописать статический IP-адрес из диапазона 192.168.10.1 - 192.168.10.20, маску 255.255.255.0, шлюз 192.168.10.254.
Вариант Б: Настроить на школьном сервере службу DHCP для автоматической выдачи адресов из того же диапазона (192.168.10.1 - 192.168.10.20), а на компьютерах поставить «Получить IP-адрес автоматически».

Вопросы:

1. Какой ключевой недостаток **Варианта А** (ручная настройка) проявится, если учителю потребуется временно подключить к сети свой ноутбук для демонстрации?
2. Какой потенциальный риск **Варианта Б** (DHCP) можно предвидеть, если в настройках DHCP-сервера по ошибке указан диапазон 192.168.10.100 - 192.168.10.120, а на компьютерах по-прежнему настроена автоматическая адресация?
3. Исходя из материала, можно ли использовать в этой локальной сети компьютеров публичные IP-адреса, например, выданные провайдером «Эр-Телеком»? Почему да или почему нет?

ТЕМА-8. Разделение IP-сетей на подсети

Введение

Современные компьютерные сети редко представляют собой одну огромную, плоскую структуру, где все устройства соединены в общем пространстве. Представьте себе большой город, который не разделен на районы, улицы и дома, а представляет собой хаотичное скопление зданий. Найти нужный адрес в таком городе было бы невероятно сложно. Точно так же и в сетях: по мере роста количества устройств управление ими, обеспечение безопасности и эффективная маршрутизация трафика становятся невозможными без логической организации. Именно эту задачу – организацию и структурирование IP-сети – решает разделение на подсети (subnetting). Это фундаментальный навык для сетевого администратора, сравнимый с умением архитектора разбивать территорию на кварталы с четкой планировкой. В рамках данного материала мы последовательно разберем, зачем нужно делить сети, как это делается технически, и как планировать адресное пространство для создания эффективной, безопасной и масштабируемой сетевой инфраструктуры.

1. Сегментация IP-сетей: базовые идеи и необходимость

1.1. Что такое сегментация и зачем она нужна?

Базовая идея: Сегментация сети – это процесс логического разделения одной большой IP-сети на несколько меньших частей, называемых подсетями (subnets). Это аналогично разделению одного большого офисного здания (исходная сеть) на отдельные этажи или отделы (подсети) с собственными правилами доступа и своим внутренним порядком.

Смысл и назначение:

1. **Уменьшение широковещательного трафика:** В пределах одной IP-сети (одного широковещательного домена) широковещательные пакеты (например, ARP-запросы «кто имеет IP-адрес X?») рассылаются всем устройствам. Представьте, что в большом зале (исходной сети) 500 человек, и каждый, кому нужно что-то сказать всем, кричит на весь зал. Скоро возникнет хаос и «шум». Разделение на подсети – это возведение перегородок, создание отдельных комнат. Теперь крик (широковещательный пакет) слышен только в одной комнате (подсети) на 20 человек. Это резко снижает бесполезную нагрузку на сетевые интерфейсы всех устройств и повышает общую производительность.
2. **Повышение безопасности:** Подсети изолируют группы устройств. Атака или сбой в одной подсети (например, в сети гостевого Wi-Fi) с меньшей вероятностью затронут критически важные устройства в другой подсети (например, серверы бухгалтерии). Это как установка противопожарных дверей между отделами.

3. **Логическая организация:** Подсети позволяют группировать устройства по функциональному признаку: отдельная подсеть для серверов, для пользователей бухгалтерии, для инженерного отдела, для сетевой инфраструктуры (маршрутизаторов, коммутаторов), для точек беспроводного доступа. Это упрощает управление, мониторинг и применение политик (например, правил фильтрации).
4. **Оптимизация маршрутизации:** Для внешних маршрутизаторов (например, провайдера) вся ваша внутренняя сеть может быть представлена как одна или несколько агрегированных записей. Внутри же организации маршрутизаторы знают точную структуру подсетей, что позволяет эффективно направлять трафик.

Пример: Исходная сеть компании: 192.168.1.0/24 (256 возможных адресов). Все устройства – серверы, принтеры, компьютеры сотрудников – в одной «куче». Любой широковещательный запрос идет на 255 устройств. После сегментации мы создаем:

- Подсеть 192.168.1.0/26 (62 узла) – для серверов.
- Подсеть 192.168.1.64/26 (62 узла) – для отдела продаж.
- Подсеть 192.168.1.128/26 (62 узла) – для отдела разработки.
- Подсеть 192.168.1.192/26 (62 узла) – для сетевой инфраструктуры.

Теперь широковещательный трафик из отдела продаж не мешает работе серверов.

1.2. Компоненты IP-адресации: адрес сети, адрес узла и маска

Чтобы понять, как работает разделение, нужно четко представлять структуру IP-адреса (IPv4).

Базовая идея: Любой IPv4-адрес состоит из 32 бит. В традиционной, несегментированной сети (сетевой класс A, B, C) эта последовательность бит жестко делится на две части:

- **Сетевой префикс (Network Prefix):** Левая часть адреса, идентифицирующая конкретную сеть в интернете или корпоративной инфраструктуре. Как почтовый индекс, который определяет город и район.
- **Идентификатор узла (Host ID):** Правая часть адреса, идентифицирующая конкретное устройство (хост) внутри этой сети. Как номер дома и квартиры в пределах района.

Маска подсети (Subnet Mask) – это 32-битное число, которое является ключом к пониманию границы между сетевым префиксом и идентификатором узла. Ее назначение – «наложить» на IP-адрес и показать, какие биты относятся к сети, а какие – к узлу.

- **Биты, установленные в 1 (единицы)** в маске, соответствуют **сетевой части** IP-адреса.
- **Биты, установленные в 0 (нули)** в маске, соответствуют **части узла** IP-адреса.

Пример (аналогия): Допустим, у нас адрес 192.168.1.10 в сети 192.168.1.0/24.

- **IP-адрес в двоичном виде:** 11000000.10101000.00000001.00001010

- **Маска /24 (или 255.255.255.0) в двоичном виде:** 11111111.11111111.11111111.00000000
- Применяем логическую операцию **И (AND)** между адресом и маской:

```
Адрес:  11000000.10101000.00000001.00001010
Маска:  11111111.11111111.11111111.00000000
Результат (Адрес сети): 11000000.10101000.00000001.00000000 ->
192.168.1.0
```

- Часть, где в маске нули (00001010), – это идентификатор узла (10).

Вывод: Маска однозначно определяет размер сети. Маска /24 оставляет для узлов 8 бит (последний октет), что дает 2^8 - 2 = 254 адреса для устройств (два адреса зарезервированы: адрес сети .0 и широковещательный .255).

2. Обмен данными между подсетями: роль маршрутизатора

Базовая идея: Подсети являются изолированными широковещательными доменами. Прямой обмен данными на канальном уровне (уровень L2, по MAC-адресам) возможен **только** между устройствами, находящимися в одной подсети. Для связи между разными подсетями **обязательно** требуется устройство третьего (сетевого) уровня модели OSI – **маршрутизатор (роутер)**.

Смысл и аналогия: Представьте две комнаты в здании (две подсети). Люди внутри одной комнаты могут разговаривать друг с другом напрямую (коммутация на L2). Но чтобы передать сообщение из одной комнаты в другую, нужно выйти в коридор (магистральную сеть), найти дверь (интерфейс маршрутизатора) и отдать сообщение дежурному (маршрутизатору). Дежурный, зная план здания (таблицу маршрутизации), отнесет сообщение к двери нужной комнаты (другому интерфейсу) и передаст его внутрь.

Технический процесс на примере:
Устройство A (192.168.1.10/24, шлюз 192.168.1.1) хочет отправить данные устройству B (192.168.2.20/24).

1. **Принятие решения:** Устройство A сравнивает свою подсеть (192.168.1.0/24) и подсеть назначения (192.168.2.0/24). Они разные.
2. **Обращение к шлюзу:** Понимая, что цель в другой сети, устройство A не пытается искать MAC-адрес устройства B. Вместо этого оно отправляет пакет на MAC-адрес своего **шлюза по умолчанию** – интерфейса маршрутизатора (192.168.1.1), находящегося в его же подсети.
3. **Работа маршрутизатора:** Маршрутизатор получает пакет на интерфейс 192.168.1.1. Он «снима» заголовок канального уровня (с MAC-адресами) и анализирует IP-адрес назначения (192.168.2.20).

4. **Поиск маршрута:** Маршрутизатор смотрит в свою таблицу маршрутизации и находит запись: «Сеть 192.168.2.0/24 доступна через интерфейс 192.168.2.1» (другой свой интерфейс).
5. **Переупаковка и отправка:** Маршрутизатор создает **новый** кадр канального уровня. В качестве IP-адресов источника и назначения остаются оригинальные (192.168.1.10 и 192.168.2.20). Но MAC-адрес источника теперь – это MAC интерфейса 192.168.2.1, а MAC-адрес назначения – это MAC устройства B (который маршрутизатор узнал через ARP-запрос в подсети 192.168.2.0/24). Пакет отправляется в подсеть назначения.

Ключевой вывод: Маршрутизатор – это «почтовое отделение» между подсетями. У каждого хоста в подсети должен быть корректно прописан IP-адрес шлюза по умолчанию (интерфейс маршрутизатора в этой подсети), иначе связь с внешним миром (другими подсетями и интернетом) будет невозможна.

3. Планирование адресации и расчетные формулы для сегментации

3.1. Разбиение на подсети на основе требований

Перед тем как делить сеть, необходимо провести **планирование**.

1. **Определить количество необходимых подсетей.** Сколько у нас будет логически изолированных групп? (Серверы, отделы, филиалы, пользователи Wi-Fi).
2. **Определить максимальное количество узлов (хостов) в каждой подсети.** Сколько устройств будет в самом большом отделе? Сколько серверов? Это определяет **минимальный размер** каждой подсети.

Базовые формулы:

- **Количество подсетей:** 2^n, где n – количество бит, «заимствованных» у исходной хостовой части для создания префикса подсети.
- **Количество узлов в подсети:** 2^m - 2, где m – количество бит, оставшихся для идентификатора хоста после заимствования. Минус 2 – это зарезервированные адреса: адрес самой подсети (все биты хостовой части = 0) и широковещательный адрес (все биты хостовой части = 1).

Пример пошагового планирования (Classful подход):
Дан адрес сети 192.168.1.0/24. Требуется создать 5 подсетей, в самой крупной из которых будет до 30 устройств.

1. **Шаг 1: Определяем необходимое количество бит для подсетей.** Нужно минимум 5 подсетей. 2^2 = 4 (мало), 2^3 = 8 (достаточно). Значит, нужно заимствовать n = 3 бита из хостовой части.
2. **Шаг 2: Определяем новую длину маски.** Исходная маска /24 (24 бита на сеть). Добавляем 3 заимствованных бита. Новая маска: /27 (или 255.255.255.224 в десятичной форме).
3. **Шаг 3: Проверяем, хватает ли места для хостов.** При маске /27 для хостов остается 32 - 27 = 5 бит (m = 5). Количество узлов: 2^5 - 2 = 32 - 2 = 30. Это **ровно** удовлетворяет требованию (до 30 узлов). Если бы требовалось больше, пришлось бы пересматривать план или использовать другую исходную сеть.
4. **Шаг 4: Определяем сами подсети.** Три заимствованных бита в четвертом октете дают 2^3 = 8 комбинаций, т.е. 8 подсетей (используем 5 из них). Приращение (шаг) между подсетями равен значению младшего значащего бита в заимствованной части. В нашем случае заимствовано 3 бита в последнем октете. Вес младшего из этих битов – 32 (позиция 2^5). Значит, шаг = 32.

- Подсеть 0: 192.168.1.0/27 (адреса узлов: .1 – .30, широковещательный: .31)
- Подсеть 1: 192.168.1.32/27 (адреса узлов: .33 – .62, широковещательный: .63)
- Подсеть 2: 192.168.1.64/27
- Подсеть 3: 192.168.1.96/27
- Подсеть 4: 192.168.1.128/27
- Подсеть 5: 192.168.1.160/27
- Подсеть 6: 192.168.1.192/27
- Подсеть 7: 192.168.1.224/27

Важно: Некоторые руководства рекомендуют избегать использования первой (.0) и последней (.224) подсетей (подсети нуля и «all-ones»), хотя современное оборудование (начиная с 1995 года) поддерживает их в рамках протокола **CIDR**.

3.2. Определение параметров подсети по IP-адресу и маске

Частая задача: дан IP-адрес хоста и маска, определить адрес сети, широковещательный адрес, диапазон допустимых адресов узлов.

Алгоритм (на примере 10.50.45.10/19):

1. **Записать маску.** /19 = 255.255.224.0 (19 единиц, затем 13 нулей).
2. **Перевести в двоичный вид интересующий октет, где проходит граница.** Граница проходит внутри 3-го октета, т.к. маска 224 = 11100000.
3. **Применить операцию AND к IP-адресу.** Третий октет адреса: 45 = 00101101. AND с 224 (11100000):
   00101101 AND 11100000 = 00100000 = 32 в десятичной системе.
4. **Адрес сети:** 10.50.32.0.
5. **Широковещательный адрес:** В хостовой части (последние 13 бит) все единицы. Третий октет: зафиксирована сетевая часть 001 (первые 3 бита из 8), хостовую часть (последние 5 бит

в этом октете) ставим в 1: 00111111 = 63. Четвертый октет (полностью хостовая часть): все 1 = 255. Итого: 10.50.63.255.

6. **Диапазон адресов узлов:** От 10.50.32.1 до 10.50.63.254.

4. Разбиение на подсети с использованием маски переменной длины (VLSM)

4.1. Проблема классического подхода (FLSM) и введение VLSM

Проблема: В рассмотренном выше примере (Classful или Fixed-Length Subnet Mask – FLSM) мы использовали для всех подсетей **одинаковую** маску (/27). Это привело к тому, что каждая подсеть получила по 30 адресов, даже если в одной из них нужно всего 5 узлов (например, для сетевого оборудования), а в другой – 30. Это неэффективное использование адресного пространства, ведущее к его быстрому истощению.

Решение: VLSM (Variable Length Subnet Mask – маска подсети переменной длины). Это техника, позволяющая делить исходную сеть на подсети **разного размера**, в зависимости от реальных потребностей в количестве узлов.

Аналогия: У вас есть большой земельный участок (исходная сеть /24). По FLSM вы делите его на 8 одинаковых маленьких участков по 1/8 каждый. Но вам нужен один большой дом (подсеть на 100 узлов), три средних коттеджа (по 30 узлов) и четыре места для гаражей (по 2 узла). Одинаковые участки не подходят. VLSM позволяет сначала отрезать большой кусок под дом, потом из оставшегося – средние под коттеджи, и наконец, из остатков – маленькие под гаражи. Минимизируем отходы.

4.2. Базовая модель и назначение блоков адресов с использованием VLSM

Алгоритм планирования VLSM:

1. **Составить список всех требуемых подсетей, отсортированный по убыванию количества необходимых узлов.**
2. **Выделять адресное пространство последовательно, начиная с самой большой подсети**, используя минимально возможную маску (т.е. оставляя достаточно бит для хостов).
3. **Следующую по списку подсеть выделять из оставшегося свободного пространства.**

Подробный пример:
Исходная сеть: 192.168.10.0/24. Требования:

- Сеть А: 100 узлов.

- Сеть B: 50 узлов.
- Сеть C: 20 узлов.
- Сеть D: 10 узлов.
- Сеть E: 5 узлов (связи «точка-точка» между маршрутизаторами).

Шаг 1: Сортируем по убыванию: A(100) → B(50) → C(20) → D(10) → E(5).

Шаг 2: Выделяем подсеть A. Нужно 100 узлов. Ближайшая степень двойки: 128. Значит, для хостов нужно 7 бит (2^7 - 2 = 126 узлов, хватит). Маска: 32 - 7 = 25. Берем первую подсеть /25 из 192.168.10.0/24:

- **Подсеть A:** 192.168.10.0/25. Диапазон: .1 – .126, широковещательный: .127.

Шаг 3: Выделяем подсеть B из оставшегося. Свободна вторая половина: 192.168.10.128/25. Нужно 50 узлов. Ближайшая степень двойки: 64. Нужно 6 бит для хостов (2^6 - 2 = 62). Маска: 32 - 6 = 26. Делим блок 128/25 пополам (первая подсеть /26 из него):

- **Подсеть B:** 192.168.10.128/26. Диапазон: .129 – .190, широковещательный: .191.

Шаг 4: Выделяем подсеть C. Свободен блок 192.168.10.192/26 (вторая половина от 128/25). Нужно 20 узлов. Ближайшая степень двойки: 32. Нужно 5 бит для хостов (2^5 - 2 = 30). Маска: 32 - 5 = 27. Делим блок 192/26 пополам:

- **Подсеть C:** 192.168.10.192/27. Диапазон: .193 – .222, широковещательный: .223.

Шаг 5: Выделяем подсеть D. Свободен блок 192.168.10.224/27. Нужно 10 узлов. Ближайшая степень двойки: 16. Нужно 4 бита для хостов (2^4 - 2 = 14). Маска: 32 - 4 = 28. Делим блок 224/27 на две подсети /28:

- **Подсеть D:** 192.168.10.224/28. Диапазон: .225 – .238, широковещательный: .239.

Шаг 6: Выделяем подсеть E (Point-to-Point). Свободен блок 192.168.10.240/28. Для связи двух маршрутизаторов нужно всего 2 адреса. Ближайшая степень двойки: 4. Нужно 2 бита для хостов (2^2 - 2 = 2). Маска: 32 - 2 = 30. Делим блок 240/28 на подсети /30:

- **Подсеть E:** 192.168.10.240/30. Диапазон: .241 и .242, широковещательный: .243.
- Остальные блоки (244/30, 248/30, 252/30) остаются свободными для будущего расширения.

Преимущество VLSM: Максимально экономное использование адресного пространства.

5. Особенности проектирования IPv6-сети и разбиение на подсети

5.1. Основные отличия IPv6

IPv6 был создан, в первую очередь, для решения проблемы исчерпания адресов IPv4. Его ключевые особенности:

- **Гигантское адресное пространство:** 128 бит против 32 бит в IPv4. Адрес записывается в шестнадцатеричном виде, группами по 16 бит, разделенными двоеточиями (например, 2001:0db8:85a3:0000:0000:8a2e:0370:7334).
- **Упрощенная структура адреса:** В IPv6 четко разделены части:
  - **Префикс сети (Network Prefix):** Как правило, первые 64 бита. Выделяется провайдером (глобальный префикс) или используется для локальных сетей (ULA, Link-Local).
  - **Идентификатор интерфейса (Interface ID):** Последние 64 бита. Часто формируется автоматически из MAC-адреса устройства (EUI-64) или случайным образом для конфиденциальности.
- **Отсутствие широковещательных адресов (Broadcast):** Их роль заменяют многоадресные рассылки (multicast) на определенные группы.
- **Упрощение работы администратора:** Благодаря обилию адресов отпадает необходимость в сложных механизмах экономии (например, NAT в его основном виде). Основная задача – логическая организация, а не борьба за каждый адрес.

5.2. Разбиение на подсети в IPv6

Базовая идея: Поскольку стандартом де-факто является использование первых 64 бит под сетевой префикс, а последних 64 бит – под идентификатор интерфейса, разбиение на подсети происходит **внутри этих 64 бит сетевого префикса**.

Схема деления:

1. **Глобальный префикс провайдера (Global Routing Prefix):** Обычно /48 или /32. Выдается вашей организации.
2. **Идентификатор подсети (Subnet ID):** Биты, которые администратор заимствует из части, следующей за глобальным префиксом, и до 64-го бита. Именно эти биты используются для создания внутренних подсетей.
3. **Идентификатор интерфейса (Interface ID):** Последние 64 бита, неизменные при создании подсетей.

Пример: Провайдер выдал организации блок 2001:0db8:acad::/48.

- Это дает 65536 (2^{16}) возможных подсетей внутри организации, так как для Subnet ID остаются биты с 49-го по 64-й (16 бит).
- Создание подсетей становится тривиальным: мы просто увеличиваем длину префикса с /48 до, например, /64 и присваиваем уникальное значение Subnet ID.

Практический пример создания подсетей:

- **Подсеть для серверов:** 2001:0db8:acad:0001::/64 (можно сократить до 2001:db8:acad:1::/64)
- **Подсеть для пользователей:** 2001:0db8:acad:0002::/64
- **Подсеть для гостей Wi-Fi:** 2001:0db8:acad:0003::/64
- **Подсеть для управляющих устройств:** 2001:0db8:acad:0004::/64

Каждая из этих подсетей имеет **ровно 2^{64} (18 квинтиллионов)** возможных адресов для устройств, что исключает любые проблемы с нехваткой адресов внутри сегмента.

Роль идентификатора интерфейса: Он является аналогом идентификатора узла в IPv4, но значительно большего размера. Его ключевая особенность – уникальность в пределах канала связи (подсети). Он не изменяется при разбиении на подсети. Администратор работает только с левой, сетевой частью (первые 64 бита), назначая разные значения Subnet ID для разных логических сегментов сети.

Вывод по IPv6: Планирование IPv6-сети фокусируется на **логической иерархии и простоте маршрутизации**. Администратор, получив от провайдера префикс (например, /48), получает в свое распоряжение огромное, практически неисчерпаемое пространство для создания подсетей фиксированного удобного размера (/64). Задача сводится к созданию понятного плана нумерации этих подсетей (использованию осмысленных значений для Subnet ID в шестнадцатеричной системе), что облегчает дальнейшее обслуживание и диагностику сети.

Заключение

Разделение IP-сетей на подсети – это краеугольный камень проектирования любой современной сетевой инфраструктуры, будь то небольшая офисная сеть или корпоративная распределенная система. От простого деления на основе классов (FLSM) индустрия перешла к гибкому и экономичному методу VLSM, который позволяет оптимально использовать дефицитное адресное пространство IPv4. Принцип сегментации не только решает проблему широковещательного шторма, но и закладывает основы безопасности, производительности и управляемости сети.

С появлением IPv6 сама проблема нехватки адресов ушла в прошлое, однако логическая организация с помощью подсетей осталась критически важной для структурирования, маршрутизации и применения политик безопасности. Техника разделения в IPv6 стала даже проще, сместив фокус с битовых вычислений на продуманное планирование иерархии.

Таким образом, мастерское владение концепциями подсетей, маски, VLSM и понимание различий между IPv4 и IPv6 являются обязательными компетенциями для специалиста в

области сетевого и системного администрирования, позволяя ему строить надежные, эффективные и готовые к будущему развитию сети.

Контрольные вопросы к ТЕМЕ-8

1. Что такое сегментация (разделение на подсети) IP-сети и какую основную проблему она помогает решить в первую очередь?
2. Какое сетевое устройство является обязательным для организации обмена данными между узлами, находящимися в разных подсетях?
3. Что такое маска подсети и как по ней определить границу между адресом сети и адресом узла?
4. Какую операцию необходимо выполнить с IP-адресом и маской подсети, чтобы получить адрес сети?
5. Как рассчитывается количество доступных адресов для узлов (хостов) в подсети, зная количество бит в хостовой части?
6. Для чего при планировании разбиения сети с использованием VLSM подсети нужно располагать в порядке убывания требуемого количества узлов?
7. В чем состоит ключевое преимущество метода VLSM по сравнению с классическим (FLSM) подходом к разбиению на подсети?
8. Какая часть IPv6-адреса (сколько бит) обычно отводится под идентификатор интерфейса (хоста)?
9. Какую часть IPv6-адреса администратор изменяет для создания различных подсетей внутри выданного организации блока (префикса)?
10. Что такое широковещательный адрес подсети и как он формируется?
11. Как называется адрес интерфейса маршрутизатора в подсети, который должен быть прописан в настройках всех локальных узлов для связи с другими сетями?
12. Какова основная причина, по которой в IPv6-сетях разбиение на подсети считается более простым и не вызывает проблем с нехваткой адресов?

Практические задания по ТЕМЕ-8

Задание 1. Диагностика локальной сети в школьном компьютерном классе.

Ситуация: Вам, как сетевому администратору небольшой городской школы, поступила жалоба от учителя информатики. В компьютерном классе 15 компьютеров, соединённых в сеть. Учитель говорит: «Раньше всё работало, но после замены старого маршрутизатора на новый из настройки «Мастера быстрой настройки» на нём, компьютеры в классе видят друг друга в сетевом окружении и могут печатать на общем принтере, но **не могут зайти в электронный дневник через интернет**. На всех компьютерах высвечивается ошибка «Сеть без доступа к интернету», хотя кабель провайдера подключён, и на самом маршрутизаторе индикатор интернета горит».

Используя знания о сегментации сетей и обмене данными между подсетями, сформулируйте наиболее вероятную причину проблемы в **одном предложении**. Какой **один параметр** в настройках сетевого подключения компьютеров, скорее всего, необходимо проверить и, при необходимости, исправить?

Задание 2. Планирование сети для нового офиса ИП Сидорова.

Индивидуальный предприниматель Сидоров открывает новый офис в бизнес-центре. Он обратился к вам для настройки сети. В офисе будет:

- Отдел продаж: 6 стационарных компьютеров.
- Бухгалтерия: 3 компьютера и 1 сетевой принтер.
- Приёмная: 2 компьютера.
- Гостевой Wi-Fi для посетителей (предполагается до 10 одновременных подключений).
- 3 IP-камеры видеонаблюдения.
- 1 сетевой накопитель (NAS).

Провайдер выделил для офиса статический адрес, но вы решили использовать для внутренней сети частный диапазон 192.168.15.0/24. **Экономно** спланируйте адресацию, разбив эту сеть на подсети так, чтобы логически отделить ключевые группы устройств. Не используя VLSM (для простоты), предложите разбиение с **одинаковой маской** для всех подсетей, но так, чтобы в каждой было **достаточно адресов** для указанного оборудования и небольшого запаса (минимум +2 адреса). Определите и запишите:

- Какую маску (в формате /XX и в виде 255.255.255.XXX) вы выберете для всех подсетей?
- Сколько всего подсетей и сколько полезных адресов в каждой вы получите?
- Назовите, для каких групп устройств вы создадите первые три подсети (назначьте им, например, адреса сетей 192.168.15.0, .32, .64 с выбранной вами маской).

Задание 3. Анализ чужой конфигурации на производственном предприятии.

Вы пришли на завод в качестве нового администратора. Предыдущий специалист оставил вам схему с записанными настройками для одного из сегментов сети цеха. На схеме указано: «Подсеть станков с ЧПУ: 172.16.5.160/27, шлюз: 172.16.5.161». Вам нужно добавить в эту подсеть новый станок. Ваша задача – определить **диапазон IP-адресов**, которые можно назначить новому станку, не конфликтуя с уже существующими устройствами. Рассчитайте и запишите:

- Адрес сети.
- Широковещательный адрес.
- **Конкретный диапазон** адресов, доступных для назначения хостам (станкам). Учтите, что один адрес (172.16.5.161) уже занят шлюзом.

Задание 4. Экономия адресов в филиале банка с помощью VLSM.

В небольшом дополнительном офисе банка «Российский кредит» в городе N необходимо организовать сеть из следующих сегментов:

- Сеть кассовой зоны и операционистов: требуется 25 адресов.
- Сеть внутренних служебных ПК (бухгалтерия, руководство): требуется 10 адресов.
- Сеть для системы видеонаблюдения: требуется 8 адресов.

- Две служебные сети для соединения маршрутизатора офиса с маршрутизатором головного офиса и с локальным межсетевым экраном (каждая «точка-точка» требует 2 адреса). Головной офис выделил для этого филиала блок 10.23.8.0/24. Продемонстрируйте принцип VLSM, **указав порядок (очередность)** выделения подсетей для максимальной экономии адресов. Перечислите названия сетей в том порядке, в котором вы будете «отрезать» для них подсети от общего блока 10.23.8.0/24. Объясните в одном предложении, почему вы выбрали именно такой порядок.

Задание 5. Подготовка к внедрению IPv6 в вузе.

Государственный технический университет получает от провайдера для пилотного проекта IPv6-сети префикс 2001:DB8:ACAD:1::/48. Вам поручено подготовить концепцию логического разбиения. Необходимо спроектировать адресацию для:

- Сети главного учебного корпуса.
- Сети общежития №1.
- Сети административного блока.
- Сети библиотечного комплекса.

Используя стандартную для IPv6 практику, запишите **первые четыре подсети** (/64), которые вы выделите из полученного префикса, в **полной и сокращённой** шестнадцатеричной форме. По аналогии с примером из теории, используйте последовательные числа для идентификатора подсети (Subnet ID). Например, для главного корпуса: 2001:0DB8:ACAD:1:0000::/64 (полная) и 2001:DB8:ACAD:1::/64 (сокращённая).

ТЕМА-9. Создание и настройка небольшой компьютерной сети

Введение

Представьте, что вам нужно построить небольшой посёлок, где дома (компьютеры и другие устройства) должны общаться друг с другом: обмениваться письмами, звонить, передавать посылки. Просто проложить дороги между домами недостаточно. Нужно спланировать, как эти дороги пройдут (топология), установить перекрёстки и знаки (сетевые устройства и протоколы), дать каждому дому уникальный адрес (система адресации), поставить охрану на въезде (безопасность) и обучить жителей правилам общения. Создание небольшой компьютерной сети — это аналогичный комплексный процесс проектирования, монтажа и настройки, обеспечивающий связь между устройствами для обмена данными и совместного использования ресурсов. Этот материал посвящён ключевым шагам и принципам этого процесса.

Раздел 1: Планирование и создание небольшой компьютерной сети

1.1. Определение ключевых факторов

Прежде чем покупать оборудование и тянуть кабели, необходимо ответить на ряд фундаментальных вопросов, определяющих весь облик будущей сети.

- **Цель сети (Зачем?):** Что будет делать сеть? Будет ли это сеть для обучения в компьютерном классе, для небольшого офиса из 10 человек или для домашнего использования с доступом в интернет? Пример: Сеть в кабинете информатики предназначена для доступа учащихся к общим учебным материалам на сервере и выхода в интернет под контролем преподавателя. Цель определяет всё последующие решения.
- **Масштаб и рост (Сколько и насколько вырастет?):** Сколько устройств (компьютеры, принтеры, IP-камеры) будет подключено изначально и сколько может добавиться в ближайшие 1-2 года? Пример: Если в офисе 5 компьютеров, но планируется расширение до 15, нужно выбирать коммутатор не с 8, а минимум с 16 или 24 портами, чтобы не покупать новый через полгода.

- **Бюджет (За сколько?):** Финансовые ограничения — это реальность. Они заставляют искать баланс между стоимостью и функциональностью. Пример: Можно купить недорогой домашний Wi-Fi роутер для офиса, но он "упадёт" при подключении 20 устройств. Для офиса нужна более надёжная и производительная, но и более дорогая модель.
- **Физическая среда (Где?):** Где будут расположены устройства? Это одно помещение, несколько комнат на одном этаже или разные этажи? Это определяет длину кабелей и необходимость в дополнительной инфраструктуре (кабельные каналы, розетки). Пример: В историческом здании прокладка витой пары может быть затруднена, и придётся больше полагаться на беспроводные технологии, что повлияет на безопасность и скорость.
- **Требования к производительности и надёжности (Как быстро и без перебоев?):** Какая нужна скорость передачи данных? Допустимы ли кратковременные перерывы в работе? Пример: Для бухгалтерии, которая передаёт большие файлы отчётов, важна высокая скорость и стабильность проводного соединения. Для гостевого Wi-Fi в кафе достаточно "посредственной" скорости, но важно покрытие.

1.2. Выбор топологии сети

Топология — это схема, карта, способ соединения устройств в сети. Это "чертёж дорог" нашего посёлка.

- **Шинная топология:** Все устройства подключены к одному общему кабелю (шине). **Аналогия:** Один длинный телефонный провод, к которому параллельно подключены все телефонные аппараты. Если кто-то говорит, слышат все. **Плюс:** Простота и дешевизна монтажа. **Минус:** Обрыв основного кабеля выводит из строя всю сеть. Низкая производительность и безопасность при большом числе устройств. Сейчас почти не используется в чистом виде.
- **Кольцевая топология:** Каждое устройство соединено с двумя соседними, образуя кольцо. Данные ходят по кругу. **Аналогия:** Круг стола, где записка передаётся от одного человека к следующему, пока не найдёт адресата. **Плюс:** Предсказуемое время доставки данных. **Минус:** Выход из строя любого устройства или кабеля разрывает кольцо и останавливает всю сеть (если нет резервного кольца).
- **Звездообразная топология:** Самая распространённая. Все устройства подключены к центральному узлу — коммутатору (switch). **Аналогия:** Центральная телефонная станция (АТС). Все телефоны подключены к станции, которая обеспечивает связь между любыми двумя абонентами. **Плюсы:** Высокая надёжность. Поломка одного кабеля или устройства не влияет на остальных. Легко добавлять новые устройства. Простота диагностики. **Минус:** Выход из строя центрального коммутатора парализует всю сеть. Поэтому для критически важных сетей коммутатор должен быть надёжным, а иногда их ставят два (резервирование).
- **Смешанная топология:** На практике в сетях, даже небольших, часто комбинируют топологии. **Пример:** Внутри каждого отдела офиса устройства соединены "звездой" вокруг своего коммутатора. А сами эти коммутаторы соединены между собой в "звезду" вокруг

основного, более мощного коммутатора (ядра сети) или в кольцо для повышения отказоустойчивости.

Вывод для небольшой сети: В подавляющем большинстве случаев для небольшой сети (до 20-50 устройств) оптимальна и достаточна **звездообразная топология** с одним или несколькими коммутаторами.

1.3. Выбор сетевых устройств

Это "инфраструктура" нашего посёлка: перекрёстки, почтовые отделения, шлагбаумы.

- **Сетевой адаптер (NIC - Network Interface Card):** Устройство, которое вставляется в компьютер (или уже встроено в него) и позволяет ему подключаться к сети. Это "почтовый ящик" и "дверь" устройства в сетевой мир. У каждого адаптера есть уникальный физический адрес — MAC-адрес.
- **Коммутатор (Switch): Ключевое устройство для локальной сети (LAN).** Его задача — принимать данные от одного устройства и грамотно пересылать их именно тому устройству, которому они предназначены, а не всем подряд. **Аналогия:** Умный почтальон в маленьком посёлке. Он получает письмо (кадр данных) с адресом получателя (MAC-адрес). Он знает, в каком доме (порту) живёт каждый житель. Поэтому он несёт письмо прямо к нужной двери (в нужный порт). **Пример:** Компьютер А отправляет файл на компьютер Б. Коммутатор получает этот файл, смотрит на адрес назначения и отправляет его только на тот порт, к которому подключён компьютер Б. Компьютеры В и Г этот файл не увидят. Это повышает и безопасность, и общую скорость работы сети.
- **Маршрутизатор (Router):** Устройство, которое соединяет разные сети между собой. Его основная задача — **маршрутизация**: определение оптимального пути для передачи данных между сетями. **Аналогия:** Главный почтамт города, который связывает ваш маленький посёлок (ваша домашняя/офисная сеть) с огромной мировой почтовой системой (Интернет). Он получает письмо (пакет данных), смотрит на индекс и адрес (IP-адрес), и решает, через какую "магистраль" (канал связи) его отправить дальше. **В небольшой сети** это чаще всего устройство, которое подключает вашу локальную сеть к интернету от провайдера. Современные домашние/офисные маршрутизаторы часто объединяют в себе: сам маршрутизатор, Wi-Fi точку доступа, коммутатор (4-8 портов) и простой межсетевой экран.
- **Точка беспроводного доступа (Wireless Access Point, WAP):** Устройство, создающее беспроводную сеть (Wi-Fi), к которой могут подключаться клиенты (ноутбуки, телефоны). **Аналогия:** Радиобашня, которая транслирует сигнал, чтобы жители могли общаться по радиосвязи без проводов. Часто встроена в маршрутизатор.
- **Кабели и разъёмы:** "Дороги" сети. Для проводных соединений в локальной сети стандартом является **витая пара (UTP/FTP)** категории 5e или 6 с разъёмами **RJ-45**.

Итог по выбору: Для типичной небольшой сети нужно: маршрутизатор (шлюз в интернет + Wi-Fi), один или несколько коммутаторов (если портов на маршрутизаторе не хватает), сетевые кабели и, конечно, сами компьютеры с сетевыми адаптерами.

1.4. Выбор и настройка протоколов. Система адресации

Протоколы — это **правила и язык** общения в сети. Без единого языка устройства не поймут друг друга.

- **Стек протоколов TCP/IP — это фундамент современного интернета и большинства сетей.**
  - **IP (Internet Protocol):** Протокол межсетевого взаимодействия. Его главная задача — **адресация** и **маршрутизация** пакетов данных. Он отвечает на вопросы: "Куда идти?" и "Как добраться?". IP-адрес — это уникальный логический адрес устройства в сети. **Аналогия:** Почтовый индекс и улица с домом. IP-адрес (например, 192.168.1.10) позволяет доставить данные в нужную сеть и конкретное устройство в ней.
  - **TCP (Transmission Control Protocol) и UDP (User Datagram Protocol):** Протоколы транспортного уровня. Они отвечают за **доставку данных между приложениями** на устройствах. **Аналогия с почтой:**
    - **TCP — это заказное письмо с уведомлением.** Он устанавливает соединение, гарантирует доставку, проверяет целостность данных, запрашивает повторную отправку, если что-то потерялось. Медленнее, но надёжнее. Используется для веб-страниц (HTTP/HTTPS), электронной почты, передачи файлов (FTP).
    - **UDP — это открытка.** Он просто отправляет данные без установки соединения и гарантий доставки. Быстрее, но ненадёжно. Используется для видеостримов, онлайн-игр, голосовой связи (VoIP), где скорость важнее, а потерю нескольких пакетов можно пережить.
- **Система IP-адресации для небольшой сети:**
  - **Публичные (белые) IP-адреса:** Выдаются провайдером, уникальны в интернете. Их мало и они дороги.
  - **Приватные (серые) IP-адреса:** Используются **внутри** локальных сетей. Они не маршрутизируются в интернете. Это ваши "внутренние номера" в посёлке. Стандартные диапазоны: 10.0.0.0 – 10.255.255.255, 172.16.0.0 – 172.31.255.255, **192.168.0.0 – 192.168.255.255** (самый популярный для домашних/малых сетей).
  - **Маска подсети:** Определяет, какая часть IP-адреса относится к номеру сети, а какая — к номеру устройства в этой сети. **Пример:** В сети 192.168.1.0 с маской 255.255.255.0 все устройства имеют адреса вида 192.168.1.X, где X — число от 1 до 254. Первые три числа (192.168.1) — это идентификатор сети, последнее (X) — идентификатор хоста.
  - **Основной шлюз (Default Gateway):** IP-адрес маршрутизатора в вашей локальной сети. Это "выходная дверь" из вашей сети в интернет. Все запросы "наружу" компьютер отправляет на этот адрес.
  - **DNS-сервер:** "Телефонная книга интернета". Преобразует понятные человеку имена сайтов (www.example.com) в понятные машине IP-адреса (93.184.216.34). Обычно это адрес, предоставленный провайдером, или публичные DNS-серверы (например, Google: 8.8.8.8).
- **Настройка протоколов:** В небольшой сети чаще всего используется **DHCP (Dynamic Host Configuration Protocol)**. Это служба (обычно работающая на маршрутизаторе), которая **автоматически** выдаёт каждому новому устройству, подключённому к сети, IP-

адрес, маску, шлюз и адрес DNS. Это избавляет от ручного ввода адресов на каждом компьютере. **Аналогия:** Администрация нового посёлка автоматически присваивает вновь прибывшим жителям свободный адрес и выдаёт карту с указанием, где находится почтамт (шлюз) и справочное бюро (DNS).

Раздел 2: Меры по обеспечению безопасности сети

Безопасность сети — это не роскошь, а необходимость. Вашу сеть, как и ваш дом, нужно защищать от злоумышленников.

2.1. Уязвимости и сетевые атаки

- **Уязвимость:** Слабое место в системе, которое можно использовать для нарушения её безопасности. **Примеры:** Старое программное обеспечение с известными ошибками, простые пароли, неправильно настроенные права доступа, открытые порты на маршрутизаторе.
- **Сетевые атаки:** Действия, направленные на использование уязвимостей. Их можно классифицировать:

- **Разведывательные атаки (Reconnaissance):** Цель — собрать информацию о сети, не нанося прямого вреда. **Аналогия:** Вор "прощупывает" район: какие дома, какие замки на дверях, когда жильцы уходят на работу. **Примеры атак:**

  - **Сканирование портов:** Злоумышленник проверяет, какие "двери" (сетевые порты) на вашем устройстве или маршрутизаторе открыты и какие службы за ними работают (веб-сервер, файловый обмен и т.д.).
  - **Сканирование ping (эхо-запрос):** Отправка ICMP-пакетов для выявления "живых" устройств в сети.
  - **Сбор информации из публичных источников (OSINT):** Поиск информации о компании/сети в интернете, соцсетях.
- **Атаки доступа (Access Attacks):** Цель — получить несанкционированный доступ к системе или данным. **Аналогия:** Взлом замка или проникновение в дом через незапертое окно. **Примеры атак:**

  - **Подбор пароля (Brute Force):** Автоматизированный перебор множества комбинаций пароля.
  - **Использование уязвимостей (Exploits):** Запуск специального кода, использующего "дыру" в программе, чтобы получить контроль над системой.
  - **Фишинг:** Обман пользователя с целью заставить его ввести свои учётные данные на поддельном сайте.
- **Отказ в обслуживании (DoS-атаки):** Цель — не получить доступ, а нарушить доступ к ресурсу для законных пользователей, перегрузив его запросами. **Аналогия:** Толпа

злоумышленников непрерывно звонит в одну и ту же службу доставки пиццы, занимая все линии. Настоящие клиенты не могут дозвониться. **Примеры:**

- **Обычная DoS-атака:** Атака с одного источника.
- **Распределённая DoS-атака (DDoS):** Атака с тысяч заражённых компьютеров (ботнета) одновременно. Справиться с ней намного сложнее.

2.2. Базовые меры защиты

- **Резервное копирование (Backup):** Самая важная и часто недооценённая мера! Атака может уничтожить или зашифровать ваши данные. Резервная копия — это ваша "спасательная шлюпка". **Правило 3-2-1:** Храните **3** копии данных, на **2** разных типах носителей (например, внешний диск и облако), при этом **1** копия должна храниться в другом физическом месте (оффсайт).
- **Обновление и установка исправлений (Patching):** Производители постоянно выпускают обновления, которые закрывают обнаруженные уязвимости. **Пример:** Windows Update, обновления роутера, обновления антивируса. Регулярное обновление — это "починка заборов и замков" в вашей сети.
- **Межсетевые экраны (Firewalls):** Это "пограничный контрольно-пропускной пункт" между сетями (например, между вашей локальной сетью и интернетом). Он анализирует весь входящий и исходящий трафик по заданным правилам и блокирует потенциально опасный. **Аналогия:** Охранник на въезде в посёлок, который проверяет машины по списку. **Как работает:** Правила обычно основаны на IP-адресах, портах и типах протоколов. Например, правило может запрещать любой входящий трафик из интернета, кроме ответов на запросы изнутри сети или подключений к веб-серверу (порт 80/443).
- **Аутентификация, авторизация и учёт (AAA):**
  - **Аутентификация (Authentication): "Кто ты?"** Процесс проверки личности пользователя или устройства. Чаще всего по логину и паролю, по отпечатку пальца, по смарт-карте.
  - **Авторизация (Authorization): "Что тебе можно?"** Процесс определения, к каким ресурсам и что именно разрешено делать прошедшему аутентификацию пользователю. **Пример:** Пользователь Иван прошел аутентификацию (ввёл пароль). Авторизация определяет, что ему можно читать файлы в папке "Бухгалтерия", но нельзя их удалять, и он не имеет доступа к папке "Дирекция" вовсе.
  - **Учёт (Accounting): "Что ты делал?"** Процесс логирования (записи в журнал) действий пользователя: когда зашёл, что открывал, сколько времени работал. Это для контроля и расследования инцидентов.
- **Включение протокола SSH (Secure Shell):** При управлении сетевым оборудованием (коммутаторами, маршрутизаторами) по сети вместо старого и небезопасного протокола **Telnet** (который передаёт всё, включая пароли, в открытом тексте) **обязательно** нужно использовать **SSH**. SSH шифрует весь сеанс связи, включая пароли и команды. **Аналогия:** Разница между отправкой важного приказа открытой радиосвязью (Telnet), которую могут услышать все, и отправкой его по зашифрованному каналу связи (SSH).

Раздел 3: Управление сетевым оборудованием

3.1. Файловые системы маршрутизаторов и коммутаторов

Сетевое оборудование (Cisco, MikroTik и др.) — это специализированные компьютеры. У них есть своя операционная система (например, Cisco IOS) и своя файловая система для хранения необходимых файлов.

- **Что хранится:**
  - **Операционная система устройства (OS Image):** "Прошивка", которая делает маршрутизатор маршрутизатором.
  - **Файл конфигурации запущенной системы (Running Config):** Текущие настройки, которые действуют прямо сейчас. Хранится в оперативной памяти (RAM). При выключении питания они теряются!
  - **Файл конфигурации для загрузки (Startup Config):** Настройки, сохранённые в постоянной памяти (NVRAM). При включении устройства именно эти настройки загружаются в Running Config.
  - **Другие файлы:** Файлы журналов (log), дампы памяти.

Аналогия: У вас есть компьютер. На жёстком диске (постоянная память) установлена Windows (OS Image) и сохранён документ "МоиНастройки.txt" (Startup Config). Когда вы включаете компьютер, Windows загружается в оперативную память, а настройки из документа применяются к системе — это текущее состояние (Running Config). Если вы вносите изменения в настройки системы, но не сохраняете их в документ, то после перезагрузки всё вернётся к старому.

3.2. Резервное копирование и восстановление конфигурации

Потеря конфигурации устройства из-за сбоя или ошибки администратора может парализовать сеть. Поэтому её нужно регулярно резервировать.

- **Резервное копирование с помощью текстовых файлов:** Самый простой способ. Открыть сессию управления устройством (через консоль или SSH), отобразить текущую конфигурацию командой (например, `show running-config`), скопировать весь текст из терминала и вставить в текстовый файл на своём компьютере (например, `backup_router_20241001.txt`). **Восстановление:** В случае сбоя нужно заново подключиться к устройству (возможно, через консольный порт с настройками по умолчанию) и вручную ввести все команды из этого текстового файла.
- **Резервное копирование с помощью протокола TFTP:** TFTP (Trivial File Transfer Protocol) — простой протокол для передачи файлов по сети. На компьютер администратора устанавливается TFTP-сервер (программа). С маршрутизатора/коммутатора даётся команда

сохранить конфигурацию или образ IOS прямо на этот TFTP-сервер. Это удобнее для автоматизации и работы с бинарными файлами (образами OS). **Важно:** TFTP небезопасен (нет шифрования), используется только внутри доверенной сети.

- **Резервное копирование на USB-накопитель:** Многие современные устройства имеют USB-порт. Конфигурацию или образ системы можно скопировать прямо на флешку, подключённую к устройству. Это быстрый и автономный способ, не требующий сетевых служб.

3.3. Встроенные службы маршрутизации

Даже в небольшом маршрутизаторе для дома/офиса есть базовые службы маршрутизации.

- **Статическая маршрутизация:** Администратор вручную прописывает в конфигурации маршрутизатора пути до конкретных сетей. **Пример:** У вас есть две сети: основная (192.168.1.0/24) и сеть для гостей (192.168.2.0/24). Чтобы они "видели" друг друга через маршрутизатор, на нём нужно прописать статический маршрут. Подходит для очень простых сетей с 1-2 маршрутизаторами.
- **Динамическая маршрутизация (в простейших формах):** В более продвинутых настройках малых офисных маршрутизаторов могут быть включены простые протоколы динамической маршрутизации (например, RIP), которые позволяют маршрутизаторам автоматически обмениваться информацией о доступных сетях и строить таблицы маршрутизации без вмешательства человека. В типичном же домашнем/малом маршрутизаторе его главная и часто единственная "динамическая" задача — это маршрут по умолчанию (default route) в интернет, который он сам получает от провайдера по DHCP или PPPoE.

3.4. Поддержка беспроводных подключений и настройка встроенного маршрутизатора

- **Беспроводные подключения:** Современный маршрутизатор включает в себя точку беспроводного доступа (WAP). Ключевые настройки для безопасности:
  - **Имя сети (SSID):** Видимое имя вашей Wi-Fi сети.
  - **Режим безопасности и шифрования: Категорически избегать WEP и WPA (устарели, взламываются за минуты). Использовать только WPA2-Personal (AES) или, если поддерживается, WPA3.** Это устанавливает сложный "замок" на вашу беспроводную сеть.
  - **Ключ/Пароль сети (Pre-Shared Key, PSK):** Должен быть сложным (длинная фраза из случайных слов или набор символов).
  - **Фильтрация по MAC-адресам (опционально, но ненадёжно):** Можно разрешить подключение только устройствам с известными вам MAC-адресами. Однако MAC-адрес легко подделать (спуфить), поэтому это лишь дополнительный, но не основной барьер.
- **Настройка встроенного маршрутизатора:** Обычно выполняется через **веб-интерфейс**. Нужно подключиться к сети маршрутизатора, ввести в браузере его IP-адрес (часто

192.168.0.1 или 192.168.1.1), ввести логин/пароль (часто admin/admin, которые нужно сразу сменить!). В интерфейсе настраивается:

1. **Подключение к интернету (WAN):** Тип подключения (DHCP, PPPoE, статический IP — данные от провайдера).
2. **Локальная сеть (LAN):** Диапазон IP-адресов (например, 192.168.1.100-200), включение DHCP-сервера.
3. **Беспроводная сеть (Wi-Fi):** Установка SSID, пароля, стандарта безопасности (WPA2/WPA3).
4. **Безопасность:** Включение встроенного межсетевого экрана (Firewall), настройка правил, отключение удалённого администрирования из интернета.
5. **Переадресация портов (Port Forwarding):** Если нужно из интернета получить доступ к внутреннему серверу (например, к видеокамере или игровому серверу).

Заключение

Создание и настройка небольшой компьютерной сети — это последовательный процесс, основанный на чётком планировании, понимании базовых принципов взаимодействия устройств и протоколов, а также на неукоснительном соблюдении мер безопасности. От выбора топологии "звезда" и правильных устройств (коммутатор, маршрутизатор) до настройки IP-адресации через DHCP, установки сложных паролей, включения межсетевого экрана и шифрования Wi-Fi с помощью WPA2/WPA3 — каждый шаг важен. Управление оборудованием, включая резервное копирование конфигураций и знание основ файловых систем, позволяет поддерживать сеть в работоспособном состоянии и быстро восстанавливать её после сбоев.

Помните, что сетевая безопасность — это не разовое действие, а постоянный процесс: обновление ПО, анализ журналов, контроль доступа. Грамотно спроектированная и защищённая небольшая сеть станет надёжным фундаментом для работы и развития любой организации или учебного заведения.

Контрольные вопросы к ТЕМЕ-9

1. Назовите три ключевых фактора, которые необходимо определить при планировании компьютерной сети.
2. Какая сетевая топология является наиболее распространённой для построения небольших сетей и почему?
3. Чем принципиально отличается работа коммутатора (switch) от работы маршрутизатора (router)?
4. Объясните разницу между протоколами TCP и UDP. Приведите по одному примеру использования каждого из них.

5. Что такое приватный (серый) IP-адрес? Приведите пример диапазона таких адресов.
6. Для чего в сети используется DHCP-сервер?
7. Что такое межсетевой экран (firewall) и какова его основная задача?
8. Объясните разницу между аутентификацией и авторизацией.
9. Почему для удалённого управления сетевым оборудованием предпочтительнее использовать протокол SSH, а не Telnet?
10. Где хранится файл конфигурации маршрутизатора, который загружается при его включении (startup-config)?
11. Какова основная цель атаки типа "Отказ в обслуживании" (DoS-атака)?
12. Какой стандарт безопасности и шифрования необходимо использовать при настройке беспроводной сети Wi-Fi для обеспечения базовой защиты?

Практические задания по ТЕМЕ-9

Задание 1. Планирование сети для нового офиса

В г. Перми открывается новый офис транспортной компании «Быстрая доставка». В нём будут работать: директор, бухгалтер, два менеджера по логистике и секретарь-администратор. Им необходимо создать компьютерную сеть для совместного доступа к 1С-Бухгалтерии, общим документам, а также для выхода в интернет. Планируется установить 5 стационарных компьютеров, 1 многофункциональное устройство (принтер/сканер/копир) с сетевым интерфейсом и обеспечить Wi-Fi для личных смартфонов сотрудников и ноутбуков гостей.

- **Вопрос:** Исходя исключительно из информации теоретического материала, предложите минимально необходимый набор сетевого оборудования для создания этой сети. Объясните кратко роль каждого устройства в данной ситуации. (Подсказка: подумайте, какие устройства из раздела 1.3 будут выполнять ключевые функции для объединения ПК, подключения принтера, выхода в интернет и раздачи Wi-Fi).

Задание 2. Диагностика проблем с подключением

Сотрудник филиала компании в г. Воронеже жалуется, что не может зайти на корпоративный портал portal.vashafirma.ru. При этом у него открываются сайты yandex.ru и vk.com. Вы, как администратор, дистанционно проверяете настройки его компьютера и видите в свойствах сетевого подключения: IP-адрес – 192.168.15.45, Маска подсети – 255.255.255.0, Основной шлюз – 192.168.15.1, DNS-сервер – 192.168.15.1.

- **Вопрос:** Опираясь на теорию (раздел 1.4), предложите наиболее вероятную причину проблемы и простое действие для её проверки. (Подсказка: вспомните, какую функцию выполняет DNS-сервер и где он находится в данной конфигурации).

Задание 3. Выбор мер безопасности для кабинета информатики

В колледже г. Твери в компьютерном классе (15 ПК) обновляется оборудование и настраивается новая сеть. Студенты активно пользуются флеш-накопителями для переноса работ, а также выходят в интернет для поиска информации. Преподаватель опасается вирусов и несанкционированного доступа студентов к настройкам сетевого оборудования.

- **Вопрос:** Используя знания из раздела 2, порекомендуйте три конкретные и несложные меры безопасности, которые можно реализовать в этой ситуации. Объясните для каждой меры, от какой конкретной угрозы (уязвимости или типа атаки) она защищает. (Подсказка: подумайте о контроле доступа к управлению, защите беспроводной сети и базовом правиле защиты данных).

Задание 4. Восстановление работоспособности сети после сбоя

В небольшом офисе интернет-магазина в г. Сочи после внезапного отключения электричества перестал работать интернет. При осмотре вы видите, что индикаторы на маршрутизаторе горят, локальная сеть между компьютерами работает (они видят друг друга), но сайты не открываются. Вы подозреваете, что слетели настройки маршрутизатора.

- **Вопрос:** Опишите два возможных плана ваших действий по восстановлению доступа в интернет, основанных на информации из раздела 3. Первый план – на случай, если у вас есть резервная копия конфигурации этого маршрутизатора. Второй план – если резервной копии нет. (Подсказка: вспомните, где хранятся текущие и сохранённые настройки, а также как можно получить доступ к настройкам маршрутизатора для их просмотра и исправления).

Задание 5. Анализ инцидента и предложение решений

В сеть небольшого проектного бюро в г. Екатеринбурге проник вирус-шифровальщик, который зашифровал важные файлы с чертежами на нескольких компьютерах. Расследование показало, что один из сотрудников получил фишинговое письмо, якобы от банка, и запустил вложенный файл. Также выяснилось, что пароль для входа в веб-интерфейс офисного маршрутизатора был установлен по умолчанию admin/admin, а Wi-Fi сеть была защищена устаревшим стандартом WPA.

- **Вопрос:** Укажите три уязвимости (неправильных действия или упущения), которые привели или могли бы привести к инциденту, опираясь на материал разделов 2 и 3. Для каждой уязвимости предложите простое корректирующее действие, которое необходимо предпринять для предотвращения подобных ситуаций в будущем. (Подсказка: проанализируйте этапы атаки: проникновение (как?), слабая защита периметра (что?), последствия (чего можно было избежать?)).

ТЕМА-10. Введение в масштабирование сетей

1. Введение. Почему сети нужно масштабировать?

Представьте себе небольшой офис на 10 человек, где все компьютеры подключены к одному коммутатору, который, в свою очередь, подключен к интернету. Это простая и эффективная система, похожая на однокомнатную квартиру — все находятся в одном пространстве, общаются напрямую, управление простое. Но что произойдет, если бизнес будет успешным, и через год в офисе будет работать уже 100 человек? Если мы просто добавим больше компьютеров к тому же старому коммутатору, возникнут серьезные проблемы: перегрузка каналов связи, хаотичное распространение служебного трафика, невозможность локализовать неполадки, угрозы безопасности.

Масштабирование сети — это не просто «добавление большего количества оборудования». Это **планомерный процесс проектирования и развития сетевой инфраструктуры**, который позволяет ей расти, сохраняя или улучшая производительность, управляемость, безопасность и надежность. Основная идея здесь — **предусмотреть рост заранее**, чтобы расширение не превращалось в аварийную переделку всего и вся.

Аналогия с городом: Маленькая деревня может обходиться одной грунтовой дорогой. Но когда она разрастается до города, требуется планировка: главные магистрали (проспекты), второстепенные улицы, переулки, развязки, светофоры (регулировщики трафика), районы (жилые, промышленные). Без этой иерархии в городе начнутся вечные пробки и хаос. Точно так же и с сетью.

2. Реализация проекта сети. От идеи к работающей инфраструктуре

Реализация проекта сети — это последовательный переход от теоретического плана (проекта) к физически работающей системе. Это процесс, состоящий из четких этапов, пропуск любого из которых ведет к проблемам.

Базовый цикл реализации включает:

1. **Анализ требований:** Что нужно сети? Количество пользователей, типы приложений (видеоконференции, передача файлов, VoIP), требования к безопасности, планы на будущий рост.
   - **Пример:** Строительной фирме нужно передавать большие файлы чертежей (требуется высокая пропускная способность), а кол-центру — обеспечить стабильный голосовой трафик (требуется низкая задержка и джиттер).

2. **Проектирование:** Создание детального плана на основе требований. Выбор технологий, протоколов, топологии, составление спецификации оборудования.
3. **Закупка и подготовка:** Приобретение оборудования, кабелей, подготовка помещений (стойки, кабельные каналы).
4. **Физическая установка и настройка:** Монтаж оборудования, прокладка кабелей, базовая настройка устройств (прошивка, IP-адресация).
5. **Тестирование и ввод в эксплуатацию:** Проверка всех соединений, производительности, отказоустойчивости. Подписание акта и передача сети администраторам.
6. **Документирование:** Фиксация всей конфигурации, схемы соединений, паролей. Без этого этапа дальнейшее управление и масштабирование подобны ремонту в квартире без плана проводки.

Аналогия со строительством дома: Вы не начинаете строить, просто купив кирпичи. Сначала вы определяетесь, сколько комнат нужно, сколько этажей (анализ). Затем заказываете архитектурный проект (проектирование). Покупаете материалы по спецификации (закупка). Строители возводят стены, кладут коммуникации (установка). Вы проверяете, течет ли крыша, работает ли электричество (тестирование). И, наконец, получаете папку с планами дома (документирование).

3. Проект иерархической сети. Трехуровневая модель как основа масштабируемости

Это ключевая концепция для понимания масштабирования. Вместо плоской, однородной сети, где все устройства равны, используется **иерархическая (слоистая) модель**. Она делит сеть на четкие уровни, каждый из которых выполняет свою функцию. Классической является **трехуровневая модель (Core — Distribution — Access)**.

Уровень 1: Уровень доступа (Access Layer)

- **Функция: Точка входа** для конечных устройств (компьютеров, принтеров, IP-телефонов, точек доступа Wi-Fi) в сеть. Основная задача — обеспечить подключение и выполнить первоначальную политику безопасности (например, проверку по MAC-адресу, выделение VLAN).
- **Оборудование:** Неуправляемые или, что чаще, **управляемые коммутаторы** с большим количеством портов.
- **Аналогия:** Это **подъезды или этажи в многоквартирном доме**. Коммутатор доступа — это этаж, к которому подключены квартиры (устройства пользователей). Он собирает их трафик и отправляет дальше, в лифтовую шахту или на лестничную клетку.

Уровень 2: Уровень распределения (Distribution Layer, или Aggregation Layer)

- **Функция: Агрегация трафика** с нескольких коммутаторов доступа. Это «интеллектуальный» уровень, где применяются основные политики сети: маршрутизация между VLAN, фильтрация трафика (ACL — списки контроля доступа), контроль качества

обслуживания (QoS). Он изолирует проблемы на уровне доступа, не позволяя им распространяться на всю сеть.

- **Оборудование:** Мощные **управляемые коммутаторы уровня 3 (маршрутизирующие коммутаторы)**, способные выполнять функции маршрутизации.
- **Аналогия:** Это **диспетчерские или распределительные щиты в районе**. Они собирают потоки из многих домов (уровень доступа), сортируют их (электричество — в одну сеть, воду — в другую, аналогия с VLAN), применяют правила (ограничение мощности — аналогия ACL) и направляют дальше к магистралям.

Уровень 3: Уровень ядра (Core Layer)

- **Функция: Высокоскоростная магистраль** для передачи трафика между уровнями распределения, а также к центральным ресурсам (серверы, интернет-шлюз). Главные требования к ядру — скорость и надежность. Здесь **не выполняется** фильтрация трафика или сложная маршрутизация, чтобы не замедлять передачу. Задача — «прокачать» гигантские объемы данных как можно быстрее.
- **Оборудование:** Самые производительные **коммутаторы или маршрутизаторы** с высокой плотностью скоростных портов (10 Гбит/с, 40 Гбит/с и выше). Обычно используется резервирование (дублирование устройств и каналов).
- **Аналогия:** Это **скоростные кольцевые автодороги или магистральные линии метро** в большом городе. Их цель — быстро доставить пассажиров (пакеты данных) из одного района (уровень распределения) в другой, минуя все внутренние перекрестки. На них нет светофоров (фильтрации), только скоростное движение.

Преимущества иерархической модели для масштабирования:

- **Простота расширения:** Чтобы добавить новое подразделение из 50 человек, вы просто добавляете коммутатор доступа, подключаете его к уровню распределения и настраиваете политики в одном месте. Не нужно переделывать всю сеть.
- **Локализация проблем:** Сбой на одном коммутаторе доступа затронет только его пользователей, а не всю компанию.
- **Упрощение проектирования:** Каждый уровень проектируется под свою задачу, что упрощает выбор оборудования.
- **Улучшенная безопасность:** Политики безопасности можно применять централизованно на уровне распределения.

4. Расширение сети. Методы и принципы

Расширение — это практическое применение масштабирования. Есть два основных пути:

Вертикальное масштабирование (Scaling Up)

- **Суть:** Увеличение мощности **существующих** устройств.

- **Действия:** Добавление более мощных процессоров (модулей) в коммутатор, увеличение объема оперативной памяти, установка более скоростных интерфейсных модулей (например, замена гигабитных портов на 10-гигабитные).
- **Когда применять:** Когда «узким местом» является производительность конкретного устройства, но общая структура сети еще не исчерпала себя.
- **Пример/аналогия:** Вы живете в квартире и решаете сделать перепланировку, объединив комнаты и купив более вместительную мебель, чтобы разместить больше вещей, не переезжая. Или замена двигателя в автомобиле на более мощный.

Горизонтальное масштабирование (Scaling Out)

- **Суть:** Добавление **новых** устройств в сеть и распределение нагрузки между ними.
- **Действия:** Добавление новых коммутаторов доступа, внедрение дополнительных устройств уровня распределения или ядра.
- **Когда применять:** Когда нужно увеличить количество портов, повысить отказоустойчивость или географически расширить сеть. Это более гибкий и надежный путь.
- **Пример/аналогия:** Вместо перепланировки квартиры вы покупаете соседнюю квартиру и делаете между ними проход, создавая дуплекс. Или вместо одного мощного сервера вы используете кластер из нескольких менее мощных, распределяя между ними нагрузку.

Важнейший принцип при расширении — модульность. Сеть должна состоять из стандартных, легко заменяемых и повторяемых блоков (модулей). Например, типовой проект подключения этажа в здании (1 коммутатор распределения + N коммутаторов доступа). При строительстве нового корпуса этот модуль просто копируется и подключается к существующему ядру.

5. Выбор сетевых устройств. Ключевые критерии

Выбор оборудования — это не вопрос личных предпочтений, а строгое соответствие требованиям проекта и месту устройства в иерархии.

1. Определение роли в иерархии:

- **Для уровня доступа:** Важны **количество и тип портов** (медные RJ-45, оптические SFP), поддержка Power over Ethernet (PoE) для телефонов и точек доступа, возможность сегментации на VLAN.
- **Для уровня распределения:** Важна **производительность маршрутизации** (скорость обработки пакетов — pps), количество VLAN, поддержка списков контроля доступа (ACL), пропускная способность внутренней шины (backplane).
- **Для уровня ядра:** Критически важны **пропускная способность** и **надежность**. Высокая плотность скоростных портов, резервирование блоков питания, возможность «горячей» замены модулей.

2. Управляемость:

- **Неуправляемые коммутаторы:** Работают «из коробки», без возможности настройки. Подходят только для самых маленьких сетей или на самом периферийном уровне.
- **Управляемые коммутаторы:** Имеют интерфейс настройки (веб, CLI), позволяют настраивать VLAN, QoS, агрегацию каналов (Link Aggregation), STP. Это **основной инструмент** для построения масштабируемой сети.

3. Плотность портов и скорость:

- Сколько устройств нужно подключить сейчас? Сколько планируется через 3-5 лет? Обычно берут с запасом 20-30%. Скорость портов: 1 Гбит/с — стандарт для доступа, 10 Гбит/с и выше — для агрегации и ядра.

4. Поддержка сетевых протоколов и функций:

- Нужны ли функции маршрутизации (Level 3)? Поддержка стека коммутаторов (Stacking) для объединения нескольких физических устройств в одно логическое? Поддержка протоколов резервирования (например, EtherChannel, STP/RSTP/MSTP)?

Аналогия с транспортом: Выбирая автомобиль, вы смотрите на его назначение. Для перевозки семьи — вместительный минивэн (коммутатор доступа с множеством портов). Для строительных работ — мощный пикап (коммутатор уровня распределения с высокой производительностью). Для перевозки грузов между городами — скоростной фура с двумя спальнями (ядро сети с резервированием). Нельзя эффективно использовать спортивный автомобиль для перевозки мебели.

6. Коммутационное оборудование. Основа локальной сети

Коммутатор (Switch) — это главное устройство для соединения устройств внутри локальной сети (LAN). Он работает на **канальном уровне (уровень 2)** модели OSI/ TCP/IP, оперируя **MAC-адресами**.

Базовая функция: Принимает кадр (frame) данных на один из своих портов, смотрит на MAC-адрес назначения в заголовке кадра и **интеллектуально перенаправляет его только на тот порт, к которому подключен устройство-получатель**. Это фундаментальное отличие от старого хаба (концентратора), который просто дублировал трафик на все порты, создавая «домен коллизий».

Ключевые концепции для масштабирования:

1. Таблица MAC-адресов: Это «память» коммутатора, где он хранит соответствие «MAC-адрес -> номер порта». Коммутатор изучает ее автоматически: увидев кадр от устройства А на порту 1, он записывает «MAC_A -> порт 1». Для масштабируемой сети важна **размерность этой таблики** (CAM-table). В крупной сети с тысячами устройств коммутатор с таблицей на 256 записей быстро заполнит ее и начнет работать неэффективно.

2. Виртуальные LAN (VLAN): Это, пожалуй, **важнейшая технология для масштабирования и безопасности**. VLAN позволяет логически разделить один физический коммутатор (и даже сеть из нескольких коммутаторов) на **несколько изолированных широковещательных доменов**.

- **Пример:** В одном коммутаторе можно создать VLAN 10 для отдела бухгалтерии, VLAN 20 для отдела продаж и VLAN 30 для гостевого Wi-Fi. Компьютеры из разных VLAN не «увидят» друг друга на канальном уровне, даже если подключены к одному и тому же железу.
- **Преимущество для масштабирования:** Мы можем создавать логические сети по функциональному или территориальному признаку, не прокладывая новые кабели. Это резко снижает широковещательный трафик (broadcast) в каждой VLAN, что повышает общую производительность.
- **Аналогия:** VLAN — это создание **нескольких независимых виртуальных этажей** в одном физическом здании. Люди с 1-го этажа (VLAN 10) не могут просто так зайти на 5-й этаж (VLAN 20) — для этого нужен лифт с пропуском (маршрутизатор).

3. Агрегация каналов (Link Aggregation, EtherChannel, LACP): Технология объединения нескольких физических каналов (например, 2x1 Гбит/с) в один логический (2 Гбит/с). Решает две задачи:
* **Увеличение пропускной способности** между устройствами (например, между коммутатором доступа и распределения).
* **Повышение отказоустойчивости:** Если один физический канал выйдет из строя, трафик автоматически пойдет по оставшимся.
* **Аналогия:** Это как собрать из нескольких узких дорожек одну широкую многополосную трассу. Если одну полосу перекроют, движение продолжится по другим.

7. Маршрутизаторы. Связь между сетями

Если коммутатор работает внутри сети, то маршрутизатор (Router) работает **между сетями**. Он функционирует на **сетевом уровне (уровень 3)**, оперируя **IP-адресами**.

Базовая функция: Определение оптимального пути для передачи пакета данных из одной IP-сети (например, 192.168.1.0/24) в другую (например, 10.0.0.0/8 или в интернет). Для этого он использует **таблицы маршрутизации** и протоколы маршрутизации (RIP, OSPF, BGP).

Роль в масштабировании сети:

1. Сегментация на подсети (IP-сети): В большой организации одна гигантская IP-сеть (например, на 10 000 адресов) неэффективна. Маршрутизатор (или маршрутизирующий коммутатор на уровне распределения) позволяет разбить ее на множество подсетей (subnets), соответствующих VLAN.
* **Пример:** VLAN 10 (Бухгалтерия) -> подсеть 192.168.10.0/24. VLAN 20 (Продажи) -> подсеть 192.168.20.0/24.
* **Преимущество:** Локализация широковещательного трафика, упрощение управления

безопасностью (можно применять политики на стыке сетей), более эффективное использование адресного пространства.

2. Связь с внешним миром: Маршрутизатор выступает в роли **шлюза по умолчанию** для внутренних сетей, направляя трафик в интернет или в сеть другого филиала.

3. Реализация политик безопасности: На маршрутизаторе (или на специализированном межсетевом экране) настраиваются списки контроля доступа (ACL), которые разрешают или запрещают трафик **между сетями** на основе IP-адресов, портов и протоколов.
* **Пример:** ACL может запретить компьютерам из гостевой сети (VLAN 30) обращаться к серверам в сети бухгалтерии (VLAN 10), но разрешить им выход в интернет.

Аналогия: Если коммутаторы — это разветвленная система дорог внутри одного города (сети), то маршрутизатор — это **пограничный пункт или кольцевая развязка на выезде в другой город (другую сеть)**. Он решает, по какой трассе (маршруту) отправить машину (пакет) в нужный пункт назначения, и проверяет документы (ACL).

8. Управляющие устройства и системы. Центр управления сетью

В масштабируемой сети из десятков или сотен коммутаторов и маршрутизаторов ручная настройка каждого устройства по отдельности становится кошмаром. Здесь на помощь приходят **системы централизованного управления**.

1. Консольный доступ (CLI - Command Line Interface):

- Это основной способ тонкой настройки профессионального оборудования (через консольный кабель, SSH, Telnet).
- **Для масштабирования важно:** Умение создавать типовые скрипты конфигурации, которые можно применять к множеству однотипных устройств, что экономит время и снижает риск ошибок.

2. Протоколы управления сетью (SNMP - Simple Network Management Protocol):

- Позволяет системе управления (NMS — Network Management System) **собирать информацию** с сетевых устройств (загрузка CPU, интерфейсов, ошибки, температура) и получать оповещения о проблемах (trap).
- **Пример:** Система Zabbix, PRTG получают от коммутатора данные о том, что порт загружен на 95%, и отправляют предупреждение администратору.

3. Системы централизованной аутентификации (RADIUS, TACACS+):

- Вместо хранения паролей администраторов на каждом устройстве, они хранятся на центральном сервере. Администратор входит одним логином/паролем на любое сетевое устройство.

- **Для масштабирования:** Резко повышает безопасность и упрощает управление учетными записями (например, при увольнении сотрудника доступ отключается в одном месте).

4. Программно-конфигурируемые сети (SDN - Software-Defined Networking):

- Это более современная и продвинутая концепция. Ее суть — **отделение плоскости управления** (решения о том, куда отправлять трафик) от **плоскости данных** (непосредственная пересылка трафик).
- В классической сети каждое устройство (коммутатор, маршрутизатор) само принимает решения на основе своих локальных таблиц. В SDN все «интеллектуальные» решения принимает **центральный контроллер**. А сетевые устройства становятся простыми «передатчиками», которые выполняют команды контроллера.
- **Аналогия:** Классическая сеть — это **автомобили с водителями-людьми** на дорогах. Каждый водитель сам решает, куда ехать. SDN — это **беспилотные автомобили, управляемые из единого центра**. Контроллер видит всю картину пробок (состояние сети) в реальном времени и может мгновенно перенаправить потоки машин (трафика) по оптимальным маршрутам.

Общая аналогия для системы управления: Представьте, что вам нужно контролировать работу светофоров в городе из 10 перекрестков. Вы можете ездить к каждому и настраивать его отдельно (ручное управление). А можете создать **единый центр управления дорожным движением**, куда стекается информация со всех камер и датчиков (SNMP), и откуда можно дистанционно менять режимы работы всех светофоров сразу, реагируя на пробки или аварии. Второй подход — это и есть использование управляющих систем в масштабируемой сети.

Заключение

Масштабирование сети — это комплексный, итеративный процесс, основанный на предвидении и планировании. Его фундаментом служит **иерархическая модель проектирования**, которая создает структуру, подобную каркасу растущего здания. **Вертикальное и горизонтальное масштабирование** — это инструменты для усиления этого каркаса. Правильный **выбор устройств** (коммутаторов и маршрутизаторов) в соответствии с их ролью в иерархии обеспечивает необходимую производительность и функциональность. Ключевые технологии, такие как **VLAN, агрегация каналов и маршрутизация**, являются «цементом», который связывает элементы в единое, но гибкое целое. И, наконец, **системы централизованного управления** становятся «нервной системой» этой сложной конструкции, позволяя эффективно контролировать, диагностировать и адаптировать сеть к постоянно меняющимся требованиям бизнеса.

Создавая сеть с учетом этих принципов, вы не просто решаете сегодняшние задачи, а закладываете прочную основу для ее безболезненного и контролируемого роста в будущем.

Контрольные вопросы к ТЕМЕ-10

1. Что такое масштабирование сети и какую основную проблему оно решает?
2. Назовите ключевые этапы реализации проекта сети.
3. Опишите основную функцию каждого из трех уровней иерархической сетевой модели (доступ, распределение, ядро).
4. В чем разница между вертикальным (Scaling Up) и горизонтальным (Scaling Out) масштабированием? Приведите пример каждого подхода.
5. Какой главный критерий выбора сетевого устройства для уровня доступа по сравнению с уровнем ядра?
6. Объясните базовый принцип работы коммутатора на канальном уровне модели OSI/TCP/IP.
7. Что такое VLAN и как эта технология помогает в масштабировании и обеспечении безопасности сети?
8. Какую основную задачу решает технология агрегации каналов (Link Aggregation)?
9. В чем заключается основная функция маршрутизатора и на каком уровне модели OSI/TCP/IP он работает?
10. Какую роль в масштабировании сети играет сегментация на подсети (IP-сети)?
11. Для чего используется протокол SNMP в системах управления сетью?
12. Какова основная идея концепции программно-конфигурируемых сетей (SDN)?

Практические задания к ТЕМЕ-10

Задание 1. Анализ неудачного проекта

В небольшой компании «Северный ветер» (г. Архангельск), занимающейся лесопилкой, сеть росла стихийно. Сначала был один коммутатор в цеху, потом добавили второй в офисе, соединили их. Со временем начались жалобы: при передаче больших файлов чертежей с компьютера главного инженера (в цеху) на компьютер бухгалтера (в офисе) скорость падает до нуля, параллельно «падает» голосовая связь по IP-телефонии. Администратор, выпускник колледжа, сказал директору, что проблема в «плохом железе» и нужно купить один самый мощный и дорогой коммутатор на 48 портов, заменить им всё, и проблемы уйдут.
Проанализируйте предложение администратора, используя принципы иерархического проектирования. Объясните, почему замена на один мощный коммутатор, скорее всего, **не решит** системно проблем компании, и предложите базовую логику исправления ситуации.

Задание 2. Планирование расширения филиала

Сеть головного офиса транспортной компании «Восточный экспресс» во Владивостоке построена по трехуровневой модели. В новом филиале в г. Находка открывается склад и офис продаж на 30 рабочих мест. Руководство поручило вам спроектировать сетевое подключение филиала. Известно, что в филиале необходимо создать две изолированные группы: сеть сотрудников склада/офиса и гостевой Wi-Fi для клиентов.
Опишите, из каких типовых сетевых устройств (по их роли в иерархии) должна состоять локальная сеть филиала. Объясните, как будет организована связь между этими двумя

группами (сотрудники/гости) внутри филиала и как филиал в целом будет подключен к сети головного офиса для доступа к корпоративным серверам.

Задание 3. Выбор оборудования для учебного центра

Техникуму в Ростове-на-Дону выделили grant на модернизацию сетевой инфраструктуры учебного центра. Есть три лаборатории, в каждой по 20 компьютеров для студентов. Требования: 1) изолировать трафик каждой лаборатории друг от друга; 2) обеспечить высокоскоростной доступ из всех лабораторий к общему серверу с учебными материалами; 3) предоставить доступ в интернет.

В спецификации закупки значатся: неуправляемые коммутаторы на 24 порта, управляемые коммутаторы 2-го уровня на 24 порта, управляемые маршрутизирующие коммутаторы (уровня 3) на 48 портов, маршрутизатор для доступа в интернет.

Распределите типы оборудования по ролям в предполагаемой иерархической сети учебного центра. Обоснуйте свой выбор для каждого пункта (лаборатория, агрегация, ядро/выход в интернет).

Задание 4. Решение проблемы перегрузки канала

В администрации небольшого городка в Подмосковье после установки системы видеонаблюдения (20 IP-камер) начались проблемы в работе основной сети. Камеры подключили к свободным портам существующих коммутаторов в разных корпусах. Теперь при активной записи с камер сотрудники жалуются на «подвисания» в 1С и медленную работу с документами.

Используя концепции из теоретического материала, предложите два технических решения (одно на основе вертикального, другое — на основе горизонтального масштабирования), которые могли бы решить эту проблему. Объясните логику каждого решения.

Задание 5. Авария и восстановление

В сетевой инфраструктуре колл-центра «Диалог» в Казани, построенной по иерархическому принципу, произошел сбой. Перестал работать один из двух коммутаторов уровня распределения, который агрегировал трафик с 4-х этажей здания. В результате 100 операторов (2 этажа) остались без связи. Операторы на двух других этажах работают нормально.

Объясните, почему авария на одном устройстве уровня распределения привела к отказу только части сети, а не всей? Какой принцип иерархического проектирования здесь проявился? Предложите, какая технология (упомянутая в материале) могла бы предотвратить такую аварию, и как она работает в данной ситуации.

ТЕМА-11. Избыточность LAN

1. Введение - проблема избыточности в локальных сетях

Представьте себе городскую дорожную развязку с множеством мостов. Если один мост внезапно закрывается на ремонт, движение не останавливается полностью — автомобили могут использовать другие мосты, чтобы добраться до нужного пункта.
Эта **избыточность** (наличие запасных путей) обеспечивает надежность транспортной системы. Точно так же в локальных сетях (LAN) избыточность связей между коммутаторами критически важна для обеспечения бесперебойной работы: если один кабель выйдет из строя или один коммутатор откажет, сеть должна продолжать функционировать за счет альтернативных путей.

Однако здесь возникает фундаментальная проблема, которой не существует в дорожном движении. В Ethernet-сетях данные передаются в виде **кадров** (frames). Когда коммутатор получает кадр на одном из своих портов, он изучает MAC-адрес отправителя и запоминает, с какого порта пришел этот адрес. Затем он проверяет MAC-адрес получателя и, если знает, на какой порт его отправить, пересылает кадр только на этот порт. Если адрес неизвестен, коммутатор отправляет кадр на **все порты**, кроме того, с которого он пришел. Это называется **затопление** (flooding).

Теперь представьте, что в сети есть несколько путей между двумя устройствами — образуются **петли** (loops). Кадр с неизвестным адресом начинает бесконечно циркулировать по петле, каждый раз копируясь коммутаторами. Это приводит к **широковещательному шторму** (broadcast storm): сеть переполняется дублирующимися кадрами, загрузка процессоров коммутаторов достигает 100%, легитимный трафик не может пройти, и сеть полностью "ложится". Петля даже из двух избыточных связей способна парализовать всю сеть за секунды.

Таким образом, перед нами стоит противоречивая задача: **необходимо физически создать избыточные связи для надежности, но логически оставить только один активный путь, чтобы избежать петель**. Именно эту задачу и решает семейство протоколов **Spanning Tree Protocol (STP)** — "Протокол покрывающего дерева".

2. Основная идея и аналогия протокола Spanning Tree

Название протокола — "Покрывающее дерево" — происходит из теории графов. Граф — это совокупность вершин (коммутаторов) и ребер (связей между ними). **Дерево** — это связный граф без циклов (петель). **Покрывающее дерево** — это дерево, которое включает в себя все вершины исходного графа, но только часть его ребер, ровно столько, чтобы связать все вершины без петель.

Ключевая аналогия: планирование авиамаршрутов.
Представьте, что вы — диспетчер новой авиакомпании, которая открыла рейсы между шестью городами: Москва, Санкт-Петербург, Екатеринбург, Новосибирск, Сочи и Калининград. Первоначально из энтузиазма вы проложили прямые рейсы между всеми городами (полносвязная сеть). Но скоро вы понимаете, что это экономически невыгодно и создает хаос в логистике. Вам нужно оставить минимальную сеть маршрутов, чтобы из любого города можно было добраться в любой другой, но **без лишних дублирующих путей**. Вы рисуете все города и все возможные рейсы, а затем начинаете "вычеркивать" лишние линии, следя, чтобы не образовалось кольцевых маршрутов (например, Москва-СПб-Екатеринбург-Москва). В итоге вы получаете дерево маршрутов. Если один из оставшихся рейсов отменяется (например, Москва-Новосибирск), вам придется временно "включить" один из запасных маршрутов (например, Москва-Екатеринбург-Новосибирск), чтобы сохранить связность сети, и снова убедиться, что не образуется петель.

STP работает по тому же принципу:

1. Он позволяет физически проложить все кабели (создать все возможные "рейсы").
2. Алгоритм автоматически анализирует топологию и логически **блокирует** (переводит в резервное состояние) некоторые порты на коммутаторах, разрывая потенциальные петли.
3. В результате активными остаются только те связи, которые образуют древовидную (беспетлевую) структуру, охватывающую все коммутаторы — **активное покрывающее дерево**.
4. Если активная связь выходит из строя, протокол пересчитывает дерево и **разблокирует** один из резервных путей, восстанавливая связность.

3. Принцип работы STP: выбор корня, роли портов и состояний

Классический STP (IEEE 802.1D) работает по строгому алгоритму, который можно разбить на четкие шаги. Для его понимания введем несколько фундаментальных понятий.

3.1. Идентификатор моста (Bridge ID) и выбор корневого моста

Вся логика STP строится вокруг выбора **Корневого моста** (Root Bridge). Это главный коммутатор в сети, "корень дерева". Все пути в активной топологии строятся как кратчайшие пути к этому корню. Корень выбирается на основе **Идентификатора моста** (Bridge ID).

Bridge ID — это 8-байтовое значение, состоящее из двух частей:

1. **Приоритет моста** (2 байта, по умолчанию 32768).
2. **MAC-адрес коммутатора** (6 байтов).

Правило выбора: Корневым становится коммутатор с **наименьшим** Bridge ID. Сравнение идет сначала по приоритету. Если приоритеты равны, сравниваются MAC-адреса. MAC-

адрес уникален, поэтому гарантированно найдется один коммутатор с наименьшим значением.

Пример: В сети три коммутатора: SW1 (MAC: AA-AA-AA-AA-AA-AA), SW2 (MAC: BB-BB-BB-BB-BB-BB), SW3 (MAC: CC-CC-CC-CC-CC-CC). У всех приоритет по умолчанию 32768. Сравниваем Bridge ID: (32768, AA-AA-...) < (32768, BB-BB-...) < (32768, CC-CC-...). Корневым будет SW1, так как его MAC-адрес условно "меньше".

Аналогия: Выбор капитана команды. Представьте, что три группы туристов (коммутаторы) должны объединиться в один поход. У каждого есть имя (MAC) и уровень опыта (приоритет). Сначала смотрят на уровень опыта: кто опытнее, тот и ведущий. Если опыт одинаковый, смотрят на имя в алфавитном порядке. Человек с именем "Алексей" (меньшее значение) станет капитаном.

3.2. Определение ролей портов: корневой, назначенный, блокированный

После выбора корневого моста каждый коммутатор определяет роль каждого своего порта, участвующего в STP.

1. **Корневой порт (Root Port, RP):** Это порт на **каждом НЕкорневом** коммутаторе, который имеет **лучший путь к корневому мосту**. У каждого некорневого коммутатора может быть только один корневой порт. Корневой мост корневых портов не имеет.
2. **Назначенный порт (Designated Port, DP):** На каждом **сегменте сети** (на каждой физической связи между двумя коммутаторами или между коммутатором и компьютером) должен быть выбран один назначенный порт. Это порт, который будет **передавать трафик в сторону от корневого моста** в данном сегменте. Назначенный порт всегда находится на том коммутаторе, который имеет лучший путь к корневому мосту для этого сегмента.
3. **Альтернативный/Резервный порт (Alternate/Blocking Port):** Все остальные порты, которые не стали ни корневыми, ни назначенными, переводятся в состояние **блокировки** (Blocking). Они не передают и не принимают пользовательские данные, но **продолжают "слушать" служебные сообщения STP** (BPDU). Это и есть те самые "выключенные" связи, которые разрывают петли и создают избыточность. Если активный путь выйдет из строя, один из блокированных портов может быть переведен в активное состояние.

Аналогия: Система водоснабжения. Корневой мост — это главный резервуар с водой. Корневой порт на каждом узле — это та труба, по которой вода **приходит** к этому узлу с наилучшим напором (кратчайшим путем от резервуара). Назначенный порт — это труба, по которой вода **уходит** от этого узла к следующему потребителю, если этот узел находится ближе к резервуару, чем сосед. Запасные (блокированные) трубы — это перекрытые задвижки. Если основная труба лопнет, одну из задвижек можно открыть.

3.3. Состояния портов в STP

Порт не может мгновенно перейти из состояния блокировки в состояние пересылки трафика. Это предотвращает временные петли во время переконфигурации. Порты проходят несколько состояний:

1. **Блокирование (Blocking):** Порт заблокирован, не пересылает данные, слушает BPDU. **(Начальное состояние для всех портов при включении).**
2. **Прослушивание (Listening):** Порт определен как корневой или назначенный. Он начинает обрабатывать BPDU, но еще не пересылает пользовательские данные. На этом этапе происходит изучение топологии.
3. **Обучение (Learning):** Порт начинает изучать MAC-адреса из заголовков проходящих кадров, заполняя свою таблицу MAC-адресов, но все еще не пересылает данные.
4. **Пересылка (Forwarding):** Порт активно пересылает пользовательские данные.

Аналогия: Прием на работу нового сотрудника. 1) **Блокирование** — кандидат только прислал резюме, мы его рассматриваем (слушаем). 2) **Прослушивание** — мы пригласили его на собеседование, он рассказывает о себе (отправляет BPDU), но к работе не приступает. 3) **Обучение** — сотрудник на испытательном сроке, изучает процессы компании (учит MAC-адреса), но самостоятельной работы еще не ведет. 4) **Пересылка** — сотрудник прошел испытательный срок и приступил к полноценной работе.

Время перехода из блокирования в пересылку в классическом STP занимает **30-50 секунд** (20 сек Blocking→Listening, 15 сек Listening→Learning, 15 сек Learning→Forwarding). Это долго для современных сетей, что привело к появлению более быстрых версий протокола.

4. Типы протоколов семейства STP

Классический STP (802.1D) был первым, но не последним. Для устранения его недостатков (главным образом, скорости сходимости) были разработаны новые стандарты.

1. **STP (IEEE 802.1D) — Классический:** Описан выше. Одно общее дерево для всей сети. Медленная сходимость (~50 сек).
2. **RSTP (Rapid STP, IEEE 802.1w) — Быстрый:** Кардинально переработанный протокол. Основные улучшения:

- **Скорость:** Время сходимости сокращено до **1-2 секунд**. Это достигнуто за счет новых механизмов Proposal/Agreement (Предложение/Согласие), которые позволяют портам договариваться о переходе в состояние Forwarding напрямую, минуя промежуточные таймеры.
- **Новые роли портов:** Добавлены роли **Alternate** (альтернативный, резервный путь к корню) и **Backup** (резервный назначенный порт).
- **Новые состояния:** Упрощены до трех: **Discarding** (отбрасывание, объединяет Blocking, Listening), **Learning**, **Forwarding**.

3. **PVST+ (Per-VLAN Spanning Tree Plus) и Rapid PVST+:** Проприетарные протоколы Cisco. Это не отдельные стандарты, а **метод работы** STP или RSTP.

- **Ключевая идея:** Вместо одного spanning-tree на всю сеть, **для каждой VLAN создается отдельный экземпляр** дерева. Это позволяет:

- **Балансировка нагрузки:** Разные VLAN можно заставить использовать разные физические пути. Например, трафик VLAN 10 идет по одной избыточной связи, а трафик VLAN 20 — по другой. Обе связи активны, но для разных VLAN, что повышает общую пропускную способность.
- **Изоляция:** Проблема в дереве одной VLAN не затрагивает другие VLAN.
  - **PVST+** использует "медленный" алгоритм STP (802.1D) для каждого VLAN.
  - **Rapid PVST+** использует "быстрый" алгоритм RSTP (802.1w) для каждого VLAN. **Это рекомендуемый протокол в сетях Cisco.**
4. **MSTP (Multiple STP, IEEE 802.1s):** Стандартный аналог PVST+. Позволяет группировать несколько VLAN в один экземпляр spanning-tree (MST Instance), что экономит ресурсы коммутаторов по сравнению с PVST+.

Аналогия: Эволюция светофоров.

- **Классический STP** — это перекресток, где светофор переключается по жесткому таймеру, даже если машин нет. Долгое ожидание (50 сек).
- **RSTP** — это "умный" светофор с датчиком движения. Как только подъезжает машина с боковой дороги, он быстро переключается (1-2 сек).
- **PVST+** — это многоуровневая развязка. Каждый уровень (VLAN) обслуживает свой поток машин, и они могут двигаться параллельно, не мешая друг другу, увеличивая общую пропускную способность.
- **MSTP** — это та же развязка, но с группировкой: несколько похожих потоков (VLAN) пускают по одному уровню для упрощения управления.

5. Настройка протоколов STP: базовые принципы

Хотя настройка зависит от конкретного производителя (рассмотрим на примере синтаксиса Cisco IOS), общие принципы универсальны.

5.1. Общая последовательность настройки STP

1. **Выбор версии протокола:** Указать коммутатору, какой тип STP использовать.
2. **Назначение приоритета для выбора корневого моста:** Вмешаться в автоматический выбор, чтобы назначить корневым самый мощный и центрально расположенный коммутатор.
3. **Настройка стоимости портов:** Влиять на выбор активных путей, изменяя стоимость портов (параметр, определяющий "лучший" путь).
4. **Защита STP (PortFast, BPDU Guard, Root Guard):** Настройка дополнительных механизмов безопасности и ускорения.

5.2. Настройка PVST+

PVST+ включен на коммутаторах Cisco по умолчанию. Основные команды:

```
bash
Switch(config)# spanning-tree mode pvst       ! Режим PVST+ (обычно стоит
по умолчанию)
Switch(config)# spanning-tree vlan 10 root primary  ! Сделать этот
коммутатор корневым для VLAN 10
                                                ! (автоматически
понижает приоритет до 24576 или меньше)
Switch(config)# spanning-tree vlan 20 root secondary ! Сделать резервным
корневым для VLAN 20
Switch(config)# interface GigabitEthernet0/1
Switch(config-if)# spanning-tree vlan 10 cost 10    ! Вручную задать
стоимость для VLAN 10
Switch(config-if)# spanning-tree portfast           ! Включить PortFast
(только для портов, ведущих к ПК/серверам!)
! PortFast немедленно переводит порт в состояние Forwarding, минуя
Listening/Learning.
```

Пример балансировки с PVST+:

- На коммутаторе SW1: spanning-tree vlan 10 priority 24576, spanning-tree vlan 20 priority 28672.
- На коммутаторе SW2: spanning-tree vlan 10 priority 28672, spanning-tree vlan 20 priority 24576.
- **Результат:** Для VLAN 10 корнем будет SW1, и активный путь пойдет через него. Для VLAN 20 корнем будет SW2, и активный путь пойдет через него. Трафик распределен.

5.3. Настройка Rapid PVST+

Настройка аналогична, но режим меняется на rapid-pvst. Это **рекомендуемый** вариант.

```
bash
Switch(config)# spanning-tree mode rapid-pvst  ! Включить Rapid PVST+
Switch(config)# spanning-tree vlan 10 root primary
Switch(config)# interface range GigabitEthernet0/1 - 24
Switch(config-if-range)# spanning-tree portfast edge  ! PortFast для всех
портов, ведущих к конечным устройствам
Switch(config-if-range)# spanning-tree bpduguard enable  ! BPDU Guard
защитит порт: если на порту с PortFast придет BPDU (значит, подключили
коммутатор), порт будет отключен (err-disable).
```

Ключевое отличие в настройке: После включения rapid-pvst протокол сам начинает использовать быстрые механизмы сходимости RSTP. Многие дополнительные настройки (как стоимости) выполняются аналогично PVST+.

6. Типичные проблемы и рекомендации по настройке STP

1. **Неназначенный корневой мост:** Самая частая проблема. Если все коммутаторы оставить с приоритетом по умолчанию, корнем станет коммутатор с наименьшим MAC-адресом, который может быть старым или маломощным и расположенным на периферии сети. **Решение:** Вручную назначить корневым и резервным корневым самые мощные и центральные коммутаторы командой `spanning-tree vlan X root primary/secondary`.
2. **Слишком большое время восстановления (в классическом STP):** 50 секунд простоя для современного приложения недопустимо. **Решение:** Перейти на **Rapid PVST+** или **MSTP**.
3. **Отсутствие защиты PortFast/BPDU Guard на портах доступа:** Если к порту с включенным PortFast (который сразу переходит в Forwarding) подключить другой коммутатор, это может мгновенно создать петлю, так как порт не будет слушать BPDU. **Решение:** Всегда включать **BPDU Guard** на портах с PortFast. Если на такой порт придет BPDU, он будет аварийно отключен.
4. **Несогласованные настройки на двух концах линка:** Если на одном коммутаторе порт настроен как PortFast, а на другом — нет, или используются разные версии STP, это может привести к нестабильной работе. **Решение:** Следить за единообразием конфигурации.
5. **Чрезмерная нагрузка на CPU при использовании PVST+ в больших сетях:** Каждый экземпляр STP для каждой VLAN потребляет ресурсы. Если VLAN десятки или сотни, это может нагрузить процессор. **Решение:** Перейти на **MSTP**, который группирует VLAN.

Золотое правило: Всегда проектировать и настраивать spanning-tree. Никогда не полагаться на работу по умолчанию. Основная последовательность:

1. Выбрать и настроить корневой и резервный корневой коммутаторы.
2. Включить Rapid PVST+.
3. Настроить PortFast и BPDU Guard на всех портах, ведущих к конечным устройствам (ПК, серверам, IP-камерам).
4. При необходимости выполнить тонкую настройку стоимостей для оптимального пути.
5. Документировать топологию и настройки STP.

7. Заключение

Протокол Spanning Tree — это фундаментальный механизм, который делает избыточные сети LAN жизнеспособными. Он элегантно решает противоречие между необходимостью физического резервирования и недопустимостью логических петель. Понимание его принципов — выбор корня, определение ролей портов, состояний — является основой для администрирования любой сложной сетевой инфраструктуры.

Эволюция от медленного классического STP к Rapid PVST+ и MSTP показывает путь развития сетевых технологий в сторону скорости, гибкости и эффективности использования ресурсов. Грамотная настройка STP, с явным назначением корневых коммутаторов и включением защитных механизмов, превращает его из "необходимого зла" в надежный и предсказуемый инструмент построения отказоустойчивых сетей.

Запомните главную аналогию: STP — это умный диспетчер, который из паутины всех возможных дорог (физических связей) строит четкое иерархическое дерево маршрутов (логическую активную топологию), имея наготове план по быстрому включению запасных путей в случае аварии. Без этого диспетчера любая избыточность привела бы к мгновенному коллапсу сети.

Контрольные вопросы к ТЕМЕ-11

1. Какова основная цель реализации избыточных связей в локальной сети (LAN)?
2. Какая основная проблема возникает при наличии физических петель (нескольких путей) в сети Ethernet, и к чему она приводит?
3. В чем заключается ключевая задача протокола Spanning Tree (STP)?
4. Что такое Корневой мост (Root Bridge) в STP и на основе какого параметра он выбирается?
5. Перечислите три основные роли портов коммутатора в протоколе STP (корневой, назначенный, альтернативный). Дайте краткое определение каждой.
6. Почему порт в STP не может мгновенно перейти из состояния блокировки в состояние пересылки трафика? Назовите промежуточные состояния.
7. Чем принципиально отличается Rapid STP (RSTP) от классического STP с точки зрения основного преимущества?
8. В чем заключается основная идея протоколов PVST+ и Rapid PVST+ по сравнению с одним общим STP для всей сети?
9. Какая практическая польза от использования PVST+/Rapid PVST+, позволяющая повысить эффективность использования каналов связи?
10. Какой одной командой в конфигурации коммутатора Cisco можно сделать его основным (корневым) для определенной VLAN в Rapid PVST+?
11. Для чего на портах коммутатора, подключенных к компьютерам или серверам, включают функцию PortFast?
12. Почему вместе с PortFast рекомендуется всегда включать защитный механизм BPDU Guard?

Практические задания к ТЕМЕ-11

Задание 1. Анализ аварии

В небольшом филиале компании в Твери администратор для надёжности соединил два офисных коммутатора двумя патч-кордами. Через несколько минут после подключения второго кабеля пользователи начали массово жаловаться на полную недоступность сети и интернета. Индикаторы портов на коммутаторах бешено мигают. Администратор, недавно окончивший колледж, вспомнил про изученную тему и сразу предположил причину.

- **Вопрос:** Как называется возникшее в сети явление и какой базовый механизм сети Ethernet его вызывает? Какой протокол необходимо было предварительно настроить на коммутаторах, чтобы эта авария не произошла?

Задание 2. Выбор корневого коммутатора

В сетевой инфраструктуре головного офиса компании в Москве используются три коммутатора Cisco одного модели, купленные в разное время.

- SW-ОСНОВНОЙ (MAC: aaaa.aaaa.aaaa)
- SW-ЗАПАС (MAC: bbbb.bbbb.bbbb)
- SW-СТАРЫЙ (MAC: cccc.cccc.cccc)
  Все они работают с настройками STP по умолчанию. В документации указано, что корневым мостом должен быть самый новый и производительный коммутатор SW-ОСНОВНОЙ.
- **Вопрос:** Станет ли SW-ОСНОВНОЙ корневым мостом в текущей конфигурации? Обоснуйте свой ответ, объяснив логику выбора. Если нет, то какую одну простую команду нужно ввести на коммутаторе SW-ОСНОВНОЙ, чтобы он гарантированно стал корневым для VLAN 1?

Задание 3. Планирование балансировки трафика

В дата-центре компании в Екатеринбурге между двумя основными коммутаторами агрегации (АГР-1 и АГР-2) проложены два гигабитных канала для избыточности. Весь трафик разделён на две основные группы: VLAN 10 (сервисы для сотрудников) и VLAN 20 (сервисы для гостей). Администратор хочет использовать оба физических канала активно, чтобы повысить общую пропускную способность, но при этом избежать петель.

- **Вопрос:** Какой тип протокола spanning-tree из рассмотренных в материале позволяет реализовать эту задачу? Опишите кратко, как будет выглядеть логика: для какого VLAN какой коммутатор нужно сделать корневым, чтобы трафик разных VLAN пошёл разными путями.

Задание 4. Настройка портов доступа

Администратор сети в кампусе учебного заведения в Санкт-Петербурге подключает новый компьютер в аудитории к коммутатору. Он знает, что порт, ведущий к конечному устройству, можно настроить для быстрого выхода в рабочее состояние. Он вводит команду spanning-tree portfast на интерфейсе.

- **Вопрос:** Какое состояние порта (Blocking, Listening, Learning, Forwarding) будет пропущено благодаря этой команде, и сколько времени (примерно) это сэкономит по сравнению с классическим STP? Какой **обязательной** командой защиты нужно дополнить эту конфигурацию и почему (опишите возможный риск без неё)?

Задание 5. Диагностика медленного подключения

Сотрудник офиса в Новосибирске пожаловался, что после переподключения его ноутбука к другой свободной розетке в переговорной (которая ведёт на свободный порт коммутатора) доступ в сеть появляется только через 30-40 секунд. При этом на основном рабочем месте в

его кабинете сеть доступна сразу. Администратор проверил кабель — исправен. Тип коммутатора и его базовые настройки STP одинаковы в обоих коммутационных шкафах.

- **Вопрос:** Исходя из теоретического материала, какая наиболее вероятная настройка протокола spanning-tree отличается на порту в кабинете сотрудника и на порту в переговорной? Как называется функция, ускоряющая выход на порту в кабинете в состояние пересылки трафика?

ТЕМА-12. Агрегирование каналов

Введение

Представьте себе широкую многополосную автомагистраль. Поток машин движется быстро и беспрепятственно, потому что есть несколько полос для движения. Если одна полоса временно перекрыта, движение не останавливается, машины просто перестраиваются на соседние полосы. Теперь представьте узкий сельский мост, где может проехать только одна машина за раз. Это создает очереди, заторы и становится «узким местом» для всего трафика. В компьютерных сетях таким «мостом» часто является канал (link) между двумя сетевыми устройствами, например, коммутатором и сервером или двумя коммутаторами. По мере роста нагрузки одиночный канал, даже высокоскоростной, может стать ограничивающим фактором. Технология **агрегирования каналов (link aggregation)** позволяет объединить несколько физических каналов в один логический, создавая ту самую «многополосную магистраль» для данных. Это одна из ключевых технологий для повышения надежности и производительности современной сетевой инфраструктуры.

1. Основные понятия агрегирования каналов

Агрегирование каналов – это метод объединения двух или более физических сетевых соединений (каналов, портов) в одно логическое соединение с целью увеличения общей пропускной способности, обеспечения отказоустойчивости и балансировки нагрузки между этими каналами.

Давайте разберем ключевые понятия через аналогии:

- **Физический канал (Physical Link):** Это реальный кабель (медный или оптоволоконный), подключенный к конкретному физическому порту на сетевом устройстве. Это как одна отдельная труба для воды.
- **Логический канал (Logical Link / Port-Channel / Aggregated Link):** Это виртуальный интерфейс, создаваемый на основе группы физических каналов. Для операционной системы устройства и для протоколов верхних уровней это выглядит как один-единственный интерфейс с суммарной пропускной способностью. Это как сборный коллектор, в который входят несколько труб. Вода (данные) течет по коллектору, но распределяется по отдельным трубам.
- **Пропускная способность (Bandwidth):** Объем данных, который может быть передан по каналу за единицу времени. Если мы объединяем два канала по 1 Гбит/с, логический канал будет иметь пропускную способность 2 Гбит/с. Важно: один отдельный сетевой поток (например, передача одного большого файла) не станет автоматически в два раза быстрее, так как он, вероятно, будет идти только по одному из физических каналов. Но два разных потока могут идти параллельно по разным каналам, что увеличивает общую производительность для множества пользователей.

- **Отказоустойчивость (Fault Tolerance):** Способность системы продолжать работу при отказе одного из ее компонентов. Если один физический канал в агрегированной группе выходит из строя (перегрызли кабель, сгорел порт), трафик автоматически и почти мгновенно перенаправляется на оставшиеся исправные каналы. Для сетевых протоколов и пользователей это событие часто остается незамеченным, так как логическое соединение остается активным. Это похоже на переход движения с закрытой полосы на соседние.
- **Балансировка нагрузки (Load Balancing):** Алгоритм, определяющий, по какому конкретно физическому каналу внутри логической группы будет отправлен тот или иной пакет данных. Цель – максимально равномерно распределить трафик, избегая ситуации, когда один канал перегружен, а другие простаивают. Это подобно умной системе светофоров, которая направляет машины с разных улиц на разные полосы магистрали, чтобы не создавать пробок.

Пример для закрепления:
У вас есть сервер, на котором работают три виртуальные машины: 1) файловый сервер для бухгалтерии, 2) веб-сервер для корпоративного сайта, 3) сервер баз данных для CRM-системы. Если сервер подключен к коммутатору одним кабелем на 1 Гбит/с, все запросы от всех пользователей к этим трем службам будут «сталкиваться» в очереди в этом единственном канале. Установив второй кабель и настроив агрегирование, вы создаете логический канал 2 Гбит/с. Теперь трафик к файловому серверу может идти по первому кабелю, а к веб-серверу – по второму, эффективно устраняя «узкое место».

2. Принцип работы EtherChannel

EtherChannel – это конкретная фирменная технология агрегирования каналов от компании Cisco Systems, ставшая отраслевым стандартом де-факто. Это частный случай реализации общей идеи агрегирования. Принципы, лежащие в основе EtherChannel, являются фундаментальными для понимания технологии в целом.

Ключевой принцип: Согласованность (Consistency)

Все физические порты, объединяемые в EtherChannel, должны иметь абсолютно идентичные настройки на обоих концах соединения. Это **самое важное правило**.

Почему? Представьте наш сборный коллектор из труб. Если одна труба имеет диаметр 10 см, а другая – 5 см, то общая пропускная способность коллектора будет ограничена, и распределение потока станет неравномерным и сложным для управления. В сети параметры «диаметра труб» – это настройки портов.

Что должно быть идентичным:

- **Скорость (Speed):** Все порты должны работать на одной скорости (например, все на 1 Гбит/с или все на 10 Гбит/с).
- **Дуплекс (Duplex):** Все порты должны работать в одном режиме дуплекса (полудуплекс или, что чаще всего, полный дуплекс).
- **VLAN-ы (Virtual LAN):** Если порты работают в режиме access (доступ), они должны принадлежать одному и тому же VLAN. Если в режиме trunk (магистраль), они должны

пропускать один и тот же набор VLAN с одинаковыми настройками (например, одинаковый native VLAN).

- **Другие параметры:** Настройки STP (Spanning Tree Protocol), QoS (Quality of Service) и т.д.

Принцип распределения трафика: Алгоритм хэширования

EtherChannel не делит один сетевой поток (сессию) на части для параллельной передачи по разным каналам. Вместо этого он решает, по какому *целому* физическому каналу отправить *целый* кадр. Решение принимается на основе определенных полей заголовков кадра.

Алгоритм берет информацию из кадра (например, IP-адреса отправителя и получателя, MAC-адреса, номера TCP/UDP портов) и пропускает ее через математическую функцию – хэш-функцию. Результатом этой функции является число, которое определяет номер выбранного физического канала внутри группы.

Пример с почтовыми отправлениями:
Представьте, что у логического канала EtherChannel есть 4 физических «ящика» (порта) для отправки писем (кадров). Алгоритм решает, в какой ящик положить письмо, исходя из адреса отправителя и получателя. Все письма от Ивана к Петру всегда будут класться в ящик №1. Все письма от Ольги к Марии – всегда в ящик №2. Это гарантирует, что все письма в рамках одной переписки (одного сетевого соединения) придут в правильном порядке, так как они идут одним и тем же маршрутом. При этом переписки разных людей распределяются по разным ящикам, эффективно используя все каналы.

Конфигурационные протоколы EtherChannel

Для автоматического согласования и создания канала используются протоколы. Они позволяют устройствам «договориться» о возможности объединения портов.

- **PAgP (Port Aggregation Protocol):** Закрытый протокол Cisco. Работает в активном (active – инициирует переговоры) и пассивном (desirable – отвечает на переговоры) режимах.
- **LACP (Link Aggregation Control Protocol):** Открытый стандартизированный протокол (IEEE 802.3ad, позже 802.1AX). Работает в активном (active) и пассивном (passive) режимах, аналогично PAgP. LACP является отраслевым стандартом и предпочтительнее для использования в гетерогенных (разнородных) сетях, где оборудование может быть от разных производителей.

Аналогия с протоколами: Это как два способа договориться о строительстве общего моста. PAgP – это внутренний регламент одной компании. LACP – это общегосударственный стандарт, по которому могут работать разные компании. Оба приводят к одному результату – созданию моста (EtherChannel), но говорить они должны на одном языке.

3. Настройка агрегирования каналов / Настройка EtherChannel

Настройка всегда выполняется на обоих устройствах, между которыми создается канал. Рассмотрим общую логику настройки на примере оборудования Cisco (CLI – командная строка).

Шаг 1. Подготовка физических портов.
Перед объединением необходимо убедиться, что порты физически подключены и на них установлены базовые корректные настройки (speed, duplex, режим доступа или транка).

Шаг 2. Создание логического интерфейса Port-channel.
Это виртуальный интерфейс. Ему присваивается номер (например, 1).
Switch(config)# interface port-channel 1
На этом интерфейсе настраиваются все параметры, которые должны быть общими для группы: описание, режим (access/trunk), VLAN и т.д. **Важно: настройки на физических портах, которые позже будут добавлены в группу, должны быть удалены или будут перезаписаны настройками с Port-channel.**

Шаг 3. Назначение физических портов в созданную группу.
Заходим в конфигурацию каждого физического порта и указываем, что он принадлежит конкретному Port-channel, а также выбираем протокол агрегации.

```
Switch(config)# interface range gigabitethernet 0/1 - 2
Switch(config-if-range)# channel-group 1 mode active
```

В этой команде:

- channel-group 1 – указывает на логический интерфейс port-channel 1.
- mode active – задает режим работы протокола LACP (активный). Возможные варианты:
  - active – LACP активный (рекомендуется).
  - passive – LACP пассивный.
  - on – режим принудительного включения EtherChannel **без использования протоколов** (PAgP/LACP). Опасен, так как при неполном совпадении настроек на другом конце может привести к сетевой петле. Используется только если оборудование не поддерживает LACP/PAgP.

Шаг 4. Проверка конфигурации.
После назначения портов логический интерфейс port-channel 1 автоматически становится активным, если переговоры прошли успешно.

Полный упрощенный пример настройки LACP между двумя коммутаторами:
Цель: Создать магистральный EtherChannel из портов Gi0/1 и Gi0/2 между Switch1 и Switch2.

На Switch1:

```
Switch1(config)# interface port-channel 1
Switch1(config-if)# switchport mode trunk
Switch1(config-if)# switchport trunk native vlan 99
Switch1(config-if)# exit
Switch1(config)# interface range gigabitethernet 0/1 - 2
Switch1(config-if-range)# switchport mode trunk
Switch1(config-if-range)# switchport trunk native vlan 99
Switch1(config-if-range)# channel-group 1 mode active
Switch1(config-if-range)# no shutdown
```

На Switch2 (конфигурация зеркальна):

```
Switch2(config)# interface port-channel 1
Switch2(config-if)# switchport mode trunk
Switch2(config-if)# switchport trunk native vlan 99
Switch2(config-if)# exit
Switch2(config)# interface range gigabitethernet 0/1 - 2
Switch2(config-if-range)# switchport mode trunk
Switch2(config-if-range)# switchport trunk native vlan 99
Switch2(config-if-range)# channel-group 1 mode active
Switch2(config-if-range)# no shutdown
```

4. Проверка, поиск и устранение неполадок в работе EtherChannel

После настройки необходимо проверить состояние канала.

Основные команды проверки:

1. **show etherchannel summary** – **Самая важная команда.** Выводит сводную таблицу всех EtherChannel-групп.

```
Switch# show etherchannel summary
Flags:  D - down        P - bundled in port-channel
        I - stand-alone s - suspended
        H - Hot-standby (LACP only)
        R - Layer3      S - Layer2
        U - in use      f - failed to allocate aggregator

        M - not in use, minimum links not met
```

```
Group  Port-channel  Protocol    Ports
------+-------------+-----------+--------------------
1      Po1(SU)         LACP      Gi0/1(P)   Gi0/2(P)
```

Ключевые моменты:

- Po1(SU): S – Layer2 EtherChannel, U – in use (используется). Это хорошо.
- Protocol: LACP.
- Порты Gi0/1(P) и Gi0/2(P): P – bundled in port-channel (успешно объединены в port-channel). Это отличный результат.

2. **show etherchannel port-channel** – Показывает детальную информацию о конкретном логическом интерфейсе, включая список активных портов.
3. **show interfaces port-channel 1** – Показывает статус и статистику логического интерфейса так же, как для любого физического интерфейса.
4. **show interfaces trunk** – Покажет Po1 в списке транковых интерфейсов, если он настроен как trunk.

Типичные проблемы и их решение:

- **Проблема:** В выводе show etherchannel summary у портов стоит статус I (stand-alone) или s (suspended), а у Po1 – статус D (down) или M (minimum links not met).
- **Причина 1: Несовпадение настроек портов.** Самая частая причина. Проверьте командами show run interface gigabitethernet 0/1 и show interfaces gigabitethernet 0/1 switchport:
  - Совпадает ли speed/duplex? (должны быть auto или одинаковые фиксированные значения).
  - Совпадает ли режим (access/trunk)?
  - Совпадает ли Native VLAN для trunk-портов?
  - Совпадает ли список разрешенных VLAN для trunk-портов?
  - Принадлежат ли access-порты одному VLAN?
- **Решение:** Приведите настройки на обоих концах к полному соответствию. Помните: окончательные настройки должны быть на интерфейсе port-channel, а на физических портах оставить только команду channel-group.
- **Проблема:** EtherChannel не поднимается в режиме on (принудительно).
- **Причина:** Режим on не использует протоколы для переговоров. Необходимо **точной совпадение** настроек на обоих концах, и оба конца должны быть в режиме on. Частая ошибка – один конец в режиме on, а другой в active.
- **Решение:** Убедиться, что на обоих устройствах выбран режим on, или, что лучше, перевести оба конца на mode active (LACP).
- **Проблема:** Неравномерное распределение трафика (load-balancing).
- **Причина:** Алгоритм хэширования может плохо работать для конкретного типа трафика. Например, если алгоритм основан только на IP-адресах, а весь трафик идет между двумя серверами, то все потоки будут использовать один физический канал.
- **Решение:** Изучить текущий алгоритм командой show etherchannel load-balance. Изменить алгоритм глобально (например, на src-dst ip and port) командой port-channel load-balance src-dst-ip-port. Это позволит учитывать больше параметров для более равномерного распределения.

- **Проблема:** Один из портов в группе down, но трафик не перераспределился.
  - **Причина:** Возможно, не сработал механизм отказоустойчивости. Проверьте статус порта и кабеля.
  - **Решение:** Используйте команды show interfaces status для проверки состояния порта. Убедитесь, что на оставшихся портах статус P (bundled). Трафик должен был перераспределиться автоматически.

Процедура поиска неисправностей (упрощенная):

1. Физический уровень: Горят ли индикаторы на портах? Целы ли кабели?
2. Проверка show etherchannel summary: Какой статус у группы и портов?
3. Проверка согласованности настроек портов на обоих устройствах.
4. Проверка используемого протокола и режима (LACP active/passive, PAgP, on).
5. Проверка логического интерфейса port-channel (настроен ли он, включен ли).

Заключение

Агрегирование каналов, в частности технология EtherChannel, – это мощный и относительно простой инструмент сетевого администратора. Его фундаментальная идея – создание широкого, надежного логического пути из нескольких узких физических. Понимание принципа согласованности настроек, механизма распределения трафика через хэширование и умение работать с протоколами LACP/PAgP являются ключевыми для успешного внедрения этой технологии. Это не просто увеличение «цифры» пропускной способности, а создание отказоустойчивой, предсказуемой и масштабируемой сетевой инфраструктуры, которая способна незаметно для пользователей пережить отказ отдельных компонентов и эффективно обслуживать растущие потоки данных. Освоив базовые концепции, представленные в этом материале, вы сделали важный шаг к управлению современными высоконагруженными сетями.

Контрольные вопросы к ТЕМЕ-12

1. Дайте простое определение агрегированию каналов своими словами. Какую основную проблему сети оно решает?
2. Что такое физический канал и логический канал в контексте агрегирования? Приведите бытовую аналогию для их различия.
3. Объясните, как агрегирование каналов обеспечивает отказоустойчивость. Что происходит, если один из физических каналов выходит из строя?
4. Почему агрегирование каналов увеличивает общую пропускную способность, но не обязательно ускоряет передачу одного большого файла?
5. Что такое EtherChannel и как эта технология связана с общим понятием агрегирования каналов?

6. Назовите самое важное правило, которое должно выполняться для всех портов, объединяемых в EtherChannel. Почему оно так критично?
7. Какие два параметра настройки порта (кроме протокола агрегации) обязательно должны совпадать на всех портах группы EtherChannel? Приведите пример.
8. Какой принцип лежит в основе распределения трафика (балансировки нагрузки) в EtherChannel? Разделяет ли он один сетевой поток на несколько каналов?
9. Чем принципиально отличается режим настройки EtherChannel «on» от режимов «active» и «passive»?
10. Какой протокол агрегации (PAgP или LACP) является отраслевым стандартом и почему его предпочтительнее использовать?
11. Какой ключевой командой можно быстро проверить статус всех EtherChannel-групп на коммутаторе Cisco? Что означает статус порта «P» в её выводе?
12. Если в выводе команды проверки порты EtherChannel имеют статус «I» (stand-alone), какая наиболее вероятная причина неисправности?

Практические задания к ТЕМЕ-12

Задание 1. Планирование модернизации в учебном центре.

В учебном центре «Квант» в городе Томск два коммутатора Cisco в серверной комнате соединены одним кабелем (1 Гбит/с). К одному из коммутаторов подключен сервер с обучающими видеоресурсами, к которому одновременно обращаются пользователи 30 компьютерных классов. Администратор заметил, что в часы пик связь с сервером замедляется, а индикаторы порта на коммутаторе и сетевой карте сервера постоянно горят. Руководство выделило средства на покупку ещё одной сетевой карты для сервера и дополнительного модуля SFP для коммутатора.

- Какое простое и эффективное решение должен предложить администратор, используя новое оборудование, чтобы решить проблему «узкого места», не меняя основную сетевую архитектуру?
- Объясните кратко, какие два ключевых преимущества даст это решение в описанной ситуации.

Задание 2. Анализ неудачной настройки.

Молодой специалист Иван в ИТ-отделе заводоуправления в Нижнем Новгороде получил задание увеличить пропускную способность канала между двумя коммутаторами. Он подключил два новых кабеля между устройствами и на одном коммутаторе сконфигурировал интерфейсы следующим образом:

```
interface GigabitEthernet0/1
 switchport mode trunk
 switchport trunk native vlan 10
 channel-group 10 mode active
!
```

```
interface GigabitEthernet0/2
 switchport mode access
 switchport access vlan 20
 channel-group 10 mode active
```

На другом коммутаторе он настроил оба порта Gi0/1 и Gi0/2 идентично первому (как trunk с native vlan 10). Однако EtherChannel не образовался, и коммутатор показывает ошибки.

- В чём заключается фундаментальная ошибка в конфигурации Ивана? Сформулируйте правило, которое он нарушил.
- Как нужно исправить конфигурацию на первом коммутаторе, чтобы EtherChannel мог быть установлен? (Укажите только необходимые изменения).

Задание 3. Выбор протокола для смешанной среды.

В сетевой инфраструктуре регионального филиала банка в Казани используются коммутаторы Cisco (основная сеть) и несколько коммутаторов другого производителя для системы видеонаблюдения. Возникла необходимость создать отказоустойчивое и производительное соединение с повышенной пропускной способностью между коммутатором Cisco и одним из «сторонних» коммутаторов.

- Какой протокол агрегации каналов (PAgP или LACP) должен выбрать администратор для этой задачи и почему?
- В каком режиме (active или passive) нужно настроить протокол на коммутаторе Cisco, если на коммутаторе видеонаблюдения поддерживается только пассивный режим LACP?

Задание 4. Диагностика проблемы с распределением трафика.

Администратор Светлана в call-центре в Екатеринбурге настроила EtherChannel из 4 каналов между коммутаторами, обслуживающими рабочие места операторов. После настройки она заметила, что индикаторы нагрузки на двух портах горят постоянно, а на двух других почти неактивны. При этом операторы жалуются на периодические подтормаживания. Светлана проверила команду show etherchannel summary, и статус всех портов отображается как «P» (bundled), а Po1(SU) – «in use».

- Исходя из теоретического материала, какая наиболее вероятная причина неравномерной загрузки физических каналов, если сам EtherChannel работает?
- Какое действие (какую команду или настройку) может предпринять Светлана, чтобы попытаться улучшить распределение трафика между всеми четырьмя каналами?

Задание 5. Обоснование решения для резервирования.

В небольшой гостинице в Сочи критически важным является доступ в интернет для системы онлайн-бронирования, которая работает на локальном сервере. Сервер подключен к основному коммутатору одним кабелем. Владелец опасается, что обрыв этого кабеля или поломка порта коммутатора парализует работу с бронированиями.

- Предложите наиболее экономичное и надёжное решение на базе технологии агрегирования каналов, которое устранит эту «точку отказа». Что нужно добавить к текущей конфигурации (минимально)?
- Объясните, как будет вести себя сеть в случае, если один из физических каналов (например, основной кабель) будет физически повреждён.

ТЕМА-13. Беспроводные локальные сети

1. Концепции беспроводной связи. Введение в беспроводную связь

Беспроводная связь — это способ передачи информации между устройствами без использования физических проводников (кабелей). Основная идея аналогична передаче звука с помощью голоса: вы говорите (передаёте радиоволны), а кто-то в пределах слышимости слушает (принимает их). Вместо голоса используются электромагнитные волны определённых частот, чаще всего в радиодиапазоне.

Фундаментальная идея: Данные (тексты, изображения, видео) преобразуются в цифровой сигнал (последовательность единиц и нулей), который затем "накладывается" на радиоволну-переносчик. Этот процесс называется **модуляцией**. На принимающей стороне происходит обратный процесс — **демодуляция**, извлечение цифровых данных из радиосигнала.

Аналогия: Представьте, что вам нужно передать другу через реку корзину с яблоками (данные). Вы не можете перебросить её просто так. Вы ставите корзину в лодку (радиоволна) и переправляете. Лодка — это переносчик, а сам факт помещения корзины в лодку — модуляция. На том берегу друг забирает корзину (демодуляция). Разные типы беспроводной связи (Wi-Fi, Bluetooth, сотовая связь) — это разные "типы лодок" (частоты, протоколы), предназначенные для разных "рек" (дистанций и задач).

Ключевые преимущества:

- **Мобильность:** Устройства не привязаны к розетке кабелем. Это как переход от стационарного телефона к мобильному.
- **Простота развертывания:** Не нужно тянуть кабели через стены и потолки, что особенно важно в исторических зданиях или на временных объектах.
- **Масштабируемость:** Легко добавить нового пользователя — достаточно знать пароль от сети, а не прокладывать к его рабочему месту новый кабель.

Основной недостаток: Средой передачи является открытое пространство, что делает связь подверженной помехам (от других сетей, бытовых приборов) и потенциально менее безопасной, чем проводная, где сигнал физически заключён в кабель.

2. Компоненты сетей WLAN

WLAN (Wireless Local Area Network) — беспроводная локальная сеть, стандартизированная семейством протоколов IEEE 802.11, которое в быту известно как **Wi-Fi**.

Базовые компоненты:

1. **Точка доступа (Access Point, AP):** Это центральное устройство, которое создаёт беспроводную сеть. Её можно сравнить с **беспроводным коммутатором**. AP выполняет роль "переводчика" между беспроводной средой и проводной сетью (обычно Ethernet). Она транслирует свой идентификатор — **SSID (Service Set Identifier)** — это имя сети, которое вы видите в списке доступных подключений на своём телефоне или ноутбуке.

   Пример: Точка доступа — это как рация диспетчера в таксопарке. Она постоянно находится на одном канале, транслируя свою "позывную" (SSID). Все таксисты (клиентские устройства) знают эту позывную и могут установить с ней связь для получения заказов (данных).

2. **Беспроводной клиент (станция, STA):** Любое устройство, оснащённое беспроводным сетевым адаптером (Wi-Fi модулем): ноутбук, смартфон, планшет, принтер, телевизор.
3. **Беспроводной маршрутизатор (Wi-Fi Router):** Универсальное устройство, которое объединяет в себе несколько функций: **точка доступа + маршрутизатор + сетевой коммутатор**. Он не только создаёт беспроводную сеть, но и соединяет её с другой сетью (чаще всего с глобальной сетью Интернет), распределяя IP-адреса между клиентами (функция DHCP-сервера).

   Аналогия: Беспроводной маршрутизатор — это "штаб-квартира" вашей домашней сети. Он имеет выход во внешний мир (Интернет через WAN-порт), внутренний коммутатор для подключения проводных устройств (ПК, телевизора) и "громкоговоритель" (точку доступа) для объявления своей беспроводной сети всем домашним устройствам.

4. **Контроллер беспроводной сети (WLC):** Используется в крупных корпоративных сетях для централизованного управления десятками или сотнями точек доступа. Контроллер упрощает настройку, мониторинг и обеспечение безопасности всей беспроводной инфраструктуры. Это "главный диспетчер" для всех AP в здании или кампусе.

3. Топологии сетей WLAN 802.11

Топология определяет способ организации взаимодействия между устройствами.

1. **Режим "Инфраструктура" (Infrastructure Mode):** Самый распространённый режим. Все беспроводные клиенты подключаются к сети не напрямую друг к другу, а через точку доступа (AP). AP выступает в роли центрального концентратора и моста в проводную сеть.

- **BSS (Basic Service Set):** Базовая зона обслуживания. Это один "пузырь" сети, создаваемый одной точкой доступа. Все клиенты в этом "пузыре" знают SSID и подключаются к одной AP.
- **ESS (Extended Service Set):** Расширенная зона обслуживания. Несколько BSS (точек доступа), объединённых одной проводной сетью (обычно LAN) и транслирующих **один и тот же SSID**. Это позволяет клиенту перемещаться по зданию (роуминг) — его ноутбук автоматически переключится с AP на первом этаже на более мощный сигнал AP на втором, не разрывая соединение. Это как сеть сотовой связи внутри здания: много вышек (AP), но вы говорите по телефону, перемещаясь между ними без обрывов.

Пример: В университете в каждой аудитории своя точка доступа, но имя сети везде одно, например, "University_Wi-Fi". Студент, выходя из одной аудитории и заходя в другую, остаётся в одной логической сети.

2. **Режим "Ad-hoc" (Independent Basic Service Set, IBSS):** Прямое соединение "точка-точка" между клиентами без использования точки доступа. Устройства общаются напрямую. Эта топология используется реже (например, для прямой передачи файлов между ноутбуками или в multiplayer-играх).

   Аналогия: Режим инфраструктуры — это общение в офисе через секретаря (AP). Все вопросы и документы вы передаёте секретарю, а он — адресату. Режим Ad-hoc — это когда два сотрудника в коридоре напрямую договариваются о чём-то друг с другом, минуя секретаря.

4. Принципы работы беспроводной локальной сети. Структура кадра 802.11. Функционирование беспроводной связи

Основной принцип — разделение среды (CSMA/CA).
В проводных сетях Ethernet используется алгоритм CSMA/CD (обнаружение коллизий). В беспроводной среде "услышать" коллизию сложно, так как устройство не может одновременно передавать и слушать эфир на одной частоте. Поэтому применяется **CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance)** — множественный доступ с контролем несущей и **предотвращением** коллизий.

Как это работает (упрощённо):

1. **Ожидание тишины:** Устройство "слушает" эфир. Если эфир свободен (нет передач), оно ждёт случайный короткий промежуток времени (DIFS).
2. **Готовность:** После DIFS, если эфир всё ещё свободен, устройство посылает короткий контрольный сигнал **RTS (Request To Send)** точке доступа, сообщая о намерении передать данные и о том, сколько времени для этого потребуется.
3. **Разрешение:** Точка доступа, получив RTS, отвечает сигналом **CTS (Clear To Send)**, который слышат все клиенты в зоне. CTS, по сути, говорит: "Всем молчать, я разрешил этому устройству передавать в течение следующего времени".
4. **Передача данных:** Клиент передаёт данные.
5. **Подтверждение:** Точка доступа, получив данные без ошибок, отправляет клиенту подтверждение **ACK (Acknowledgment)**. Если ACK не пришло, клиент повторяет передачу.

Аналогия: Представьте комнату, где много людей (устройства) и один ведущий (точка доступа). Чтобы задать вопрос, вы не просто кричите, когда захотите. Вы:

1. Ждёте тишины.
2. Поднимаете руку (RTS).

3. Ведущий кивает вам (CTS), давая слово.
4. Вы задаёте вопрос (передача данных).
5. Ведущий подтверждает: "Понял ваш вопрос" (ACK). Только после этого кто-то другой может попытаться "поднять руку".

Структура кадра 802.11
Кадр — это "конверт", в который упаковываются данные для передачи. Основные части кадра:

- **Заголовок (Header):** Содержит служебную информацию: MAC-адреса источника и назначения, MAC-адрес точки доступа (для управления), контроль последовательности (чтобы не перепутать порядок кадров) и информацию о продолжительности передачи (чтобы другие устройства знали, как долго эфир будет занят).
- **Тело (Frame Body):** Полезная нагрузка — собственно те данные, которые нужно передать (фрагмент веб-страницы, часть файла).
- **Контрольная сумма (FCS):** Число, рассчитанное на основе данных в кадре. Приёмник выполняет такой же расчёт. Если результаты не совпали, значит, данные исказились в эфире, и кадр нужно передать заново.

5. Управление каналами

Радиоспектр — ограниченный ресурс. Стандарты Wi-Fi (802.11b/g/n) работают в диапазоне 2.4 ГГц, который разделён на несколько частично перекрывающихся каналов. В России их 13. Канал — это конкретная "полоса" частот внутри диапазона.

Проблема: Если две соседние точки доступа (например, у вас и у соседа) работают на одном или близких (перекрывающихся) каналах, они создают друг другу помехи. Это как два человека, пытающихся поговорить с вами одновременно на одной теме — вы с трудом разбираете их речь.

Решение:

1. **Выбор непересекающихся каналов:** В диапазоне 2.4 ГГц непересекающимися являются каналы 1, 6 и 13. Грамотная настройка сети подразумевает, что соседние AP должны быть настроены на эти разные каналы для минимизации помех.
2. **Диапазон 5 ГГц (802.11a/n/ac/ax):** Этот диапазон предлагает больше каналов, и они практически не пересекаются. Кроме того, в нём меньше помех от бытовых приборов. Однако сигнал 5 ГГц хуже проникает через стены, чем сигнал 2.4 ГГц.
3. **Автоматический выбор канала:** Большинство современных маршрутизаторов умеют при запуске "прослушивать" эфир и автоматически выбирать наименее загруженный канал.

Аналогия: Диапазон 2.4 ГГц — это узкая улица с малоэтажными домами (малая помехозащищённость), где все жильцы (устройства) вынуждены общаться. Диапазон 5 ГГц — это широкий проспект с многоэтажными домами (лучшая изоляция каналов), где больше места для одновременных разговоров.

6. Безопасность беспроводных локальных сетей. Угрозы для сетей WLAN. Обеспечение безопасности WLAN

Угрозы:

1. **Неавторизованный доступ (взлом):** Злоумышленник подключается к вашей сети без разрешения. Он может использовать ваш интернет-канал, а что хуже — получить доступ к вашим сетевым ресурсам (файлам, камерам, принтерам).
2. **Перехват данных (сниффинг):** Так как радиосигнал распространяется в эфире, его можно перехватить специальным программным обеспечением. Если данные передаются в открытом виде, злоумышленник может прочитать ваши логины, пароли, переписку.
3. **Создание ложной точки доступа (Rogue AP / Evil Twin):** Злоумышленник разворачивает свою точку доступа с SSID, идентичным вашей легитимной сети (например, "Free_Wi-Fi" в кафе). Ваше устройство может автоматически подключиться к этой фальшивке, и весь ваш трафик пойдёт через компьютер злоумышленника.
4. **Глушение сигнала (DoS-атака):** Специальным устройством генерируется мощный помеховый сигнал на частоте Wi-Fi, что делает сеть полностью неработоспособной.

Методы обеспечения безопасности:

1. **Аутентификация:** Процесс проверки легитимности клиента.

- **Открытая сеть (Open):** Аутентификации нет. Самый небезопасный вариант.
- **WEP (Wired Equivalent Privacy):** Устаревший и крайне ненадёжный метод. Взламывается за несколько минут. **Не использовать!**
- **WPA (Wi-Fi Protected Access):** Пришёл на смену WEP. Использует улучшенный протокол шифрования TKIP.
- **WPA2 (самый распространённый на сегодня):** Обязательное использование стойкого алгоритма шифрования **AES (Advanced Encryption Standard)**. Аутентификация может осуществляться по:

  - **Общему ключу (Pre-Shared Key, PSK):** Все пользователи знают один пароль (домашний вариант, "WPA2-Personal").
  - **Серверу RADIUS:** Каждому пользователю выдаются уникальные учётные данные (логин/пароль или сертификат). Используется в корпоративных сетях ("WPA2-Enterprise").
- **WPA3:** Новейший стандарт, предлагающий ещё более стойкую защиту, в том числе от офлайн-подбора паролей.

  Аналогия: Аутентификация — это как пропускной режим на предприятие. WEP — это бумажный пропуск, который легко подделать. WPA2-Personal — это единый код от турникета для всех сотрудников (если код узнает посторонний, он пройдёт). WPA2-Enterprise — это выдача каждому сотруднику индивидуальной электронной карты доступа (пропажа одной карты не ставит под угрозу всю систему).

2. **Шифрование:** Процесс преобразования открытых данных (plain text) в зашифрованные (cipher text) с помощью ключа. Даже если данные перехвачены, без ключа их невозможно прочитать.

- **AES:** Современный и очень надёжный симметричный алгоритм шифрования. Один и тот же ключ используется и для шифрования, и для расшифровки.

3. **Скрытие SSID:** Точка доступа не транслирует своё имя в эфир. Сеть становится "невидимой" для обычного сканирования. Однако это слабая защита — специализированные программы легко обнаружат такую сеть по служебному трафику.
4. **Фильтрация по MAC-адресам:** На точке доступа создаётся "белый список" MAC-адресов разрешённых устройств. Это похоже на список госномеров машин, которым разрешён въезд на территорию. Однако MAC-адрес легко подделать (спуфить), поэтому это также не является надёжной защитой, а лишь дополняет основные методы.

Итоговый совет для базовой безопасности: Всегда используйте **WPA2/WPA3 с длинным сложным паролем (не менее 12-15 символов, буквы, цифры, знаки)**. Это фундамент безопасности домашней или малой офисной сети.

7. Настройка беспроводных локальных сетей. Настройка беспроводного маршрутизатора. Настройка беспроводных клиентов

Настройка беспроводного маршрутизатора (на примере типовых шагов):

1. **Физическое подключение:** Кабель от интернет-провайдера подключается в **WAN-порт** маршрутизатора. Компьютер для первоначальной настройки подключается патч-кордом в **LAN-порт**.
2. **Вход в веб-интерфейс:** В браузере компьютера вводится IP-адрес маршрутизатора (обычно 192.168.0.1 или 192.168.1.1). Логин и пароль указаны на наклейке на корпусе устройства (часто admin/admin).
3. **Настройка подключения к Интернету (WAN):** Выбирается тип подключения, предоставленный провайдером (чаще PPPoE, Dynamic IP или Static IP). Вводятся необходимые учётные данные (логин/пароль для PPPoE).
4. **Настройка беспроводной сети (Wi-Fi):**

- **Включить беспроводное вещание.**
- **Задать имя сети (SSID):** Например, "Home_Network".
- **Выбрать режим работы (802.11):** Рекомендуется смешанный режим (b/g/n или a/n/ac/ax) для совместимости со всеми устройствами.
- **Выбрать канал:** Предпочтительно "Авто" (Auto), чтобы маршрутизатор сам выбрал наименее загруженный.
- **Выбрать метод безопасности: WPA2-Personal (AES)** или **WPA2/WPA3-Personal**.
- **Задать надёжный пароль (Pre-Shared Key, PSK):** Ключевая фраза для подключения клиентов.

5. **Настройка локальной сети (LAN):** Обычно оставляются настройки по умолчанию, включая работу DHCP-сервера, который автоматически раздаёт IP-адреса всем устройствам в сети.

6. **Смена пароля администратора: Крайне важный шаг!** Стандартный пароль admin необходимо сменить на сложный, чтобы защитить настройки маршрутизатора от несанкционированного доступа.

Настройка беспроводных клиентов:
Этот процесс максимально упрощён на современных устройствах.

1. На клиенте (ноутбук, телефон) включается Wi-Fi адаптер.
2. В списке доступных сетей выбирается SSID вашей сети (например, "Home_Network").
3. При запросе вводится пароль (PSK), который был установлен на маршрутизаторе.
4. Если пароль верен, клиент получает от маршрутизатора IP-адрес (по DHCP) и подключается к сети и Интернету.

8. Поиск и устранение неполадок в работе сетей WLAN

Типичные проблемы и их решение:

1. **"Нет подключения" или "Ограниченный доступ":**

- **Проверка физического уровня:** Перезагрузите маршрутизатор и клиентское устройство. Проверьте, не отключён ли Wi-Fi адаптер кнопкой или в настройках ОС.
- **Проверка подключения на стороне клиента:** Убедитесь, что устройство подключено к правильному SSID. Попробуйте "забыть" эту сеть в настройках и подключиться заново, введя пароль.
- **Проверка IP-адресации:** Убедитесь, что клиент получил IP-адрес (например, в диапазоне 192.168.1.x). В командной строке (cmd) выполните ipconfig. Если адрес вида 169.254.x.x — DHCP не сработал. Попробуйте назначить IP-адрес вручную в настройках сетевого адаптера (если вы уверены в параметрах своей сети) или перезапустите службу DHCP на маршрутизаторе.

2. **Низкая скорость или нестабильное соединение:**

- **Помехи и выбор канала:** Самый частый источник проблем. Используйте мобильное приложение для анализа Wi-Fi сетей (например, Wi-Fi Analyzer). Посмотрите, на каком канале работает ваша сеть и нет ли на нём множества соседних сетей. Вручную переключите маршрутизатор на самый свободный канал (1, 6, 11 в диапазоне 2.4 ГГц).
- **Физические препятствия и расстояние:** Стены, особенно железобетонные, и металлические конструкции сильно ослабляют сигнал. Попробуйте переместить маршрутизатор в более открытое, центральное место, подальше от микроволновых печей, радионянь, беспроводных колонок.
- **Устаревший режим работы:** Если ваш маршрутизатор работает в устаревшем режиме 802.11g, а клиент поддерживает 802.11n/ac, переведите маршрутизатор в смешанный режим (например, 802.11b/g/n).

3. **Не удается подключиться, хотя пароль введён верно:**

- **Проверка типа безопасности:** Убедитесь, что на клиенте и на маршрутизаторе выбран одинаковый метод безопасности (например, WPA2-Personal). Иногда старые устройства не

поддерживают WPA2, а новые по умолчанию могут не поддерживать WPA. Установите на маршрутизаторе смешанный режим WPA/WPA2, если есть такая проблема.

- **Сброс настроек маршрутизатора:** В крайнем случае можно выполнить аппаратный сброс маршрутизатора к заводским настройкам (кнопкой Reset) и провести настройку заново.

Общий подход к диагностике: Всегда двигайтесь от простого к сложному: **перезагрузка -> проверка физического соединения и расстояния -> проверка настроек безопасности и каналов -> анализ с помощью специализированных инструментов.** Понимание базовых принципов, изложенных выше, даёт основу для осмысленного поиска и устранения практически любых неполадок в беспроводной сети.

Контрольные вопросы к ТЕМЕ-13

1. Назовите основное преимущество беспроводной связи по сравнению с проводной.
2. Какой процесс преобразования цифровых данных в радиосигнал используется в беспроводной связи?
3. Перечислите три основных типа устройств, являющихся компонентами беспроводной локальной сети (WLAN).
4. Чем отличается точка доступа (Access Point) от беспроводного маршрутизатора (Wi-Fi Router)?
5. Как называется стандартный идентификатор (имя) беспроводной сети, которое видит пользователь?
6. Какой режим топологии WLAN (инфраструктуры или Ad-hoc) используется для организации роуминга в пределах большого здания?
7. Как называется основной алгоритм доступа к среде в беспроводных сетях, предназначенный для предотвращения коллизий?
8. Для чего в структуре кадра 802.11 используется поле FCS (контрольная последовательность кадра)?
9. Какие три непересекающихся канала рекомендуются для использования в диапазоне 2.4 ГГц для минимизации взаимных помех?
10. Какой современный и наиболее распространённый стандарт безопасности для домашних Wi-Fi сетей с общим паролем (Pre-Shared Key) следует использовать?
11. Назовите один из самых частых источников низкой скорости и нестабильности в беспроводной сети.
12. Какой порт на беспроводном маршрутизаторе предназначен для подключения кабеля от интернет-провайдера?

Практические задания к ТЕМЕ-13

Задание 1. Диагностика помех в многоквартирном доме.

Студент-стажёр в небольшой IT-фирме города Подольска получил задание улучшить работу

Wi-Fi в офисе. Он обнаружил, что в списке доступных сетей рядом с офисным SSID «OFIS-2024» видно более 15 соседних сетей. Большинство из них имеют стандартные названия вроде «TP-Link_1234», «Beeline_AB1C» и «MGTS_Fiber». Скорость в офисе нестабильная, особенно в часы пик. Опираясь на материал, какой самый вероятный источник проблемы и какое простое действие (не требующее покупки нового оборудования) вы порекомендуете предпринять стажёру в первую очередь? Дайте развёрнутый ответ с пояснением.

Задание 2. Выбор оборудования для точки общественного доступа.

Администрации колледжа в Екатеринбурге необходимо организовать открытую гостевую Wi-Fi сеть для посетителей в фойе главного корпуса. Основные требования: простота подключения для гостей и невозможность доступа из этой сети к внутренним ресурсам колледжа (серверу документов, камерам наблюдения). Из оборудования есть обычные бытовые Wi-Fi маршрутизаторы. Можно ли использовать имеющееся оборудование для этой задачи? Если да, то опишите минимально необходимые настройки безопасности на маршрутизаторе, которые удовлетворят требованиям. Если нет, объясните, почему, и какой тип сетевого компонента больше подходит для этой роли.

Задание 3. Анализ неудачного подключения.

К вам, как сетевому администратору офиса в Казани, обратился сотрудник, который не может подключить свой новый ноутбук к корпоративной защищённой сети «KORP-WPA2». Он уверен, что вводит правильный пароль, но получает постоянную ошибку «Не удалось подключиться к сети». Его старый ноутбук и смартфон подключаются без проблем. Изучив характеристики нового ноутбука, вы понимаете, что он поддерживает только самые современные стандарты Wi-Fi. Какая наиболее вероятная причина несовместимости, связанная с настройкой точки доступа, и как её можно устранить, не снижая общий уровень безопасности сети для остальных устройств?

Задание 4. Планирование сети для длинного помещения.

Вам поручено обеспечить стабильным Wi-Fi покрытием длинный и узкий выставочный зал в музее Томска. Вы устанавливаете один мощный маршрутизатор у одного из торцов зала. При тестировании вы обнаруживаете, что на противоположном конце зала связь есть, но скорость очень низкая и соединение постоянно обрывается. Объясните, с чем это может быть связано, используя концепции из теоретического материала. Предложите решение с использованием двух точек доступа, опишите, как их нужно настроить относительно друг друга (режим работы, SSID, каналы), чтобы обеспечить бесшовное перемещение посетителей по залу.

Задание 5. Реагирование на инцидент безопасности.

В беспроводной сети небольшого гостиничного комплекса в Сочи «Hotel_Guest» используется устаревший метод безопасности. Вы, как приглашённый специалист, за 10 минут с помощью общедоступного ПО на своём ноутбуке получаете пароль от этой сети. Какой метод безопасности, скорее всего, используется в этой сети? Какой единственный верный шаг вы порекомендуете владельцу для немедленного исправления ситуации? Составьте краткий план ваших первоочередных действий по переводу сети на безопасный режим, учитывая, что все гости будут временно отключены.

ТЕМА-14. Настройка и устранение неполадок в работе OSPF для одной области

Введение

Протокол OSPF (Open Shortest Path First) является одним из наиболее распространенных протоколов динамической маршрутизации, используемых в современных компьютерных сетях. Его основная задача – эффективно и быстро обмениваться информацией о топологии сети между маршрутизаторами, чтобы каждый из них мог построить оптимальную карту сети и определить кратчайшие пути до всех известных сетей. В рамках изучения работы OSPF для одной области (single area OSPF) мы сосредоточимся на базовых принципах его функционирования, расширенных параметрах настройки, а также методах диагностики и устранения неисправностей. Представьте, что OSPF – это система дорожных знаков и карт, которую устанавливают и поддерживают сами водители (маршрутизаторы). Они постоянно общаются друг с другом по радио, сообщая о новых дорогах, заторах или авариях, чтобы каждый мог построить самый быстрый и надежный маршрут.

1. Основы OSPF для одной области

1.1. Концепция области (Area) и её важность

Базовая идея: OSPF использует разделение большой сети на логические сегменты, называемые областями (Areas), для уменьшения служебного трафика и упрощения управления. Работа в рамках одной области (часто Area 0, или магистральная область) – это фундаментальный и самый простой сценарий развертывания OSPF.

Пояснение смысла: Представьте крупный город без районов. Каждому таксисту (маршрутизатору) пришлось бы знать в деталях каждую улицу, каждый переулок во всем городе. Это создало бы огромную нагрузку: карта была бы непомерно большой, а обновления о ремонте дорог пришлось бы рассылать всем таксистам без исключения. OSPF решает эту проблему, разбивая "город-сеть" на "районы-области". В рамках одного района таксисты знают свою территорию идеально (полная карта внутри области), а о других районах знают только основную информацию о том, как до них добраться (суммарные маршруты). В нашем случае мы рассматриваем ситуацию, когда весь "город" – это один район (одна область). Это подходит для сетей среднего размера, где нет необходимости в сложной иерархии.

Пример: Средний колледж или предприятие с 20-30 маршрутизаторами. Вся сеть может быть объявлена одной областью OSPF (например, Area 0). Каждый маршрутизатор будет знать полную топологию сети всего кампуса.

1.2. Ключевые состояния и отношения смежности (Adjacency)

Базовая идея: Для обмена информацией OSPF маршрутизаторы формируют специальные логические связи, называемые смежностями (adjacencies). Не все соседние маршрутизаторы становятся "смежными". Процесс установления смежности проходит через несколько четких состояний.

Пояснение смысла: Представьте, что вы новичок в офисе. Сначала вы видите коллег (состояние *Down/Init*). Вы посылаете приветствие ("Привет, я здесь!") – это *Hello-пакеты*. Коллега, получивший ваше приветствие, добавляет вас в список знакомых (состояние *2-Way*). Но для совместной работы вам нужно обменяться не только именами, но и полными данными о текущих проектах (состояния *ExStart/Exchange/Loading*). После полного синхронизированного обмена базами данных (Link-State Database, LSDB) вы становитесь полноценными рабочими партнерами (состояние *Full*). Только с такими партнерами (смежными маршрутизаторами) происходит обновление информации о сети. Это предотвращает хаотичный обмен данными и гарантирует согласованность информации у всех участников.

Пример: Маршрутизаторы R1 и R2 подключены к одной Ethernet-сети. Они начинают рассылать многоадресные Hello-пакеты. Обнаружив друг друга, они проверяют параметры (таймеры, номер области, аутентификацию). Если всё совпало, они начинают процесс выбора главного (DR/BDR) и обмена полными базами данных LSDB. После успешного обмена их отношение переходит в состояние *Full*, и они начинают регулярно обмениваться только инкрементальными обновлениями.

1.3. Типы сетей и выбор DR/BDR

Базовая идея: В сетях с множественным доступом (Multi-Access), таких как Ethernet, где к одному сегменту подключено много маршрутизаторов, OSPF оптимизирует процесс обмена, выбирая специальные устройства: DR (Designated Router) и BDR (Backup Designated Router).

Пояснение смысла: В большой совещательной комнате (Ethernet-сегмент), где сидят 20 человек (маршрутизаторов), если каждый начнет одновременно выкрикивать новости об изменениях в своих отделах, возникнет хаос и "шторм" объявлений. Чтобы этого избежать, выбирается "докладчик" (DR) и его заместитель (BDR). Все остальные (DROTHER) устанавливают смежность и обмениваются информацией только с DR и BDR. DR затем централизованно распространяет информацию по всей комнате. Это резко сокращает количество необходимых логических соединений: с $N*(N-1)/2$ до $2*(N-2) + 1$.

Пример: В VLAN с IP-сетью 192.168.1.0/24 подключены 5 маршрутизаторов: R1, R2, R3, R4, R5. Они начинают выбор DR/BDR на основе приоритета и Router ID. Допустим, R1 стал DR (Router ID 1.1.1.1, приоритет 100), R2 – BDR (Router ID 2.2.2.2, приоритет 50). Тогда R3, R4, R5 установят состояние *Full* только с R1 и R2. Если R3 обнаружит изменение в подключенной к нему сети, он отправит обновление только на адреса DR (224.0.0.6). DR (R1), в свою очередь, оповестит всех остальных, отправив обновление на адрес 224.0.0.5.

2. Расширенные параметры настройки для одной области

2.1. Маршрутизация на уровнях распределения и ядра

Базовая идея: В хорошо структурированной сети существует иерархия: ядро (core), распределение (distribution) и доступ (access). OSPF позволяет оптимизировать работу на каждом уровне за счет настройки приоритетов интерфейсов, стоимостей маршрутов и типов сетей.

Пояснение смысла: Представьте транспортную систему города. **Уровень доступа (access)** – это подъездные пути к домам (коммутаторы доступа, конечные пользователи). **Уровень распределения (distribution)** – это крупные улицы и развязки, собирающие трафик с множества подъездных путей (распределительные маршрутизаторы или коммутаторы L3). **Уровень ядра (core)** – это скоростные магистрали и кольцевые дороги, обеспечивающие максимально быструю переброску трафика между районами (высокопроизводительные core-маршрутизаторы).

В OSPF для этой иерархии используются:

- **Cost (стоимость):** Пропускная способность канала. На уровне ядра используются каналы с высокой пропускной способностью (10 Гбит/с, 40 Гбит/с), которым автоматически назначается низкая стоимость (например, cost для 10 Гбит/с = 1). На уровне доступа каналы уже (1 Гбит/с, 100 Мбит/с), и их стоимость выше. Это гарантирует, что трафик будет идти по наиболее быстрым магистральным путям.
- **Приоритет интерфейса для выбора DR/BDR:** На уровне распределения в Ethernet-сегментах логично назначать более высокие приоритеты надежным и производительным маршрутизаторам, чтобы они стали DR, взяв на себя служебную нагрузку.

Пример: Сеть предприятия. Уровень ядра: связь между Core1 и Core2 через интерфейс 10 GigabitEthernet. OSPF cost = 100 / 10 000 = 1. Уровень распределения: связь Distribution1 с Access1 через интерфейс 1 GigabitEthernet. OSPF cost = 100 / 1 000 = 10. Если Access1 хочет отправить пакет в сеть за Core2, он выберет путь через Distribution1 и Core1, так как суммарная стоимость этого пути (10 + 1) будет меньше, чем любой альтернативный путь с более медленными каналами.

2.2. Распространение маршрута по умолчанию

Базовая идея: Маршрут по умолчанию (0.0.0.0/0) указывает маршрутизаторам "выход во внешний мир" (например, в интернет). В OSPF его можно централизованно распространить по всей области с помощью специальной команды.

Пояснение смысла: В здании есть главный выход на улицу. Охраннику на этом выходу (граничному маршрутизатору, подключенному к провайдеру) проще всего сообщить всем сотрудникам внутри здания (всем маршрутизаторам области OSPF): "Если вам нужно попасть в любое место за пределами нашего здания, идите ко мне". Это и есть маршрут по

умолчанию. Вместо того чтобы вручную прописывать этот маршрут на каждом устройстве, OSPF позволяет граничному маршрутизатору "прорекламировать" себя как шлюз по умолчанию для всей области.

Пример: Маршрутизатор Perimeter_Router подключен к интернету через интерфейс Serial0/0/0. На нём настроен статический маршрут по умолчанию ip route 0.0.0.0 0.0.0.0 100.64.1.1. Чтобы раздать этот маршрут всем OSPF-маршрутизаторам внутри Area 0, на Perimeter_Router в конфигурации OSPF вводится команда default-information originate. После этого все внутренние маршрутизаторы получат в своей таблице маршрутизации OSPF-маршрут к сети 0.0.0.0/0 с адресом следующего прыжка на Perimeter_Router.

2.3. Точная настройка интерфейсов OSPF

Базовая идея: Поведение OSPF на каждом интерфейсе можно гибко управлять, изменяя такие параметры, как стоимость (cost), таймеры Hello/Dead, тип сети, приоритет DR/BDR и аутентификация.

Пояснение смысла: Это аналогично тонкой настройке двигателя автомобиля под разные условия: для города, для трассы, для бездорожья. Настройка интерфейса OSPF позволяет адаптировать протокол к конкретной сетевой среде.

Основные параметры и примеры их использования:

1. **Ручное задание стоимости (Cost):** ip ospf cost 50. Используется, когда автоматический расчет по полосе пропускания не отражает реальных предпочтений администратора. Например, основной канал 1 Гбит/с имеет cost=10, а резервный канал 100 Мбит/с – cost=100. Чтобы заставить трафик идти по резервному каналу только при аварии основного, можно искусственно увеличить cost основного канала до 500, сделав резервный более предпочтительным.
2. **Изменение таймеров Hello/Dead:** ip ospf hello-interval 5, ip ospf dead-interval 20. Уменьшение этих таймеров (например, до 1 и 4 секунд) ускоряет обнаружение сбоев на канале, но увеличивает служебный трафик и нагрузку на ЦП. Используется на высоконадежных и критически важных каналах (например, между core-маршрутизаторами). **Важно:** Таймеры должны быть одинаковыми на обоих концах связи, иначе смежность не установится.
3. **Изменение типа сети OSPF:** ip ospf network point-to-point. На VLAN-интерфейсе (SVI) по умолчанию OSPF считает сеть broadcast (широковещательной) и пытается выбрать DR/BDR, даже если в VLAN только два маршрутизатора. Принудительное указание типа "точка-точка" упрощает процесс, убирает выбор DR и сразу устанавливает смежность.
4. **Задание приоритета для выбора DR/BDR:** ip ospf priority 0. Приоритет (0-255). Если установить приоритет = 0, маршрутизатор никогда не станет DR/BDR на этом интерфейсе. Это полезно для маршрутизаторов уровня доступа с ограниченными ресурсами. Самый мощный маршрутизатор в сегменте получает самый высокий приоритет (например, 100).

2.4. Защита OSPF

Базовая идея: OSPF обменивается критически важной информацией о топологии сети. Эту информацию необходимо защитить от подмены, прослушивания или несанкционированного участия в процессе маршрутизации.

Пояснение смысла: Это система пропусков и паролей на вашем рабочем месте. Без защиты любой человек (устройство) мог бы войти в сеть, объявить себя "главным выходом" (DR) или разослать ложную информацию о маршрутах, что привело бы к сбою или перехвату трафика.

Основные методы защиты:

1. **Аутентификация на основе пароля (Plain Text):** Самый простой метод. На всех интерфейсах, участвующих в OSPF, задается одинаковый пароль. Пароль передается в открытом виде, поэтому защищает только от случайного подключения, но не от злоумышленника, который прослушивает трафик.
   - **Аналогия:** Секретное слово, которое выкрикивается вслух при входе в клуб. Его могут подслушать.
   - **Пример команды:** ip ospf authentication-key MY_PASSWORD и ip ospf authentication.
2. **Аутентификация MD5 (Хэширование):** Надежный метод. Пароль не передается по сети. Вместо этого передается хэш (отпечаток) сообщения, вычисленный с использованием секретного ключа. Получатель, зная тот же ключ, может проверить целостность и подлинность сообщения.
   - **Аналогия:** У вас и у охранника есть один и тот же секретный шифр. Вы передаете не пароль, а зашифрованную фразу. Охранник, расшифровав её своим шифром, понимает, что вы свой.
   - **Пример команды:** ip ospf message-digest-key 1 md5 SECRET_KEY_123 и ip ospf authentication message-digest.
3. **Фильтрация с помощью пассивных интерфейсов (Passive-interface):** Команда passive-interface предотвращает отправку OSPF Hello-пакетов на указанном интерфейсе, но при этом сеть, подключенная к этому интерфейсу, всё еще анонсируется в OSPF.
   - **Пояснение смысла:** Вы не хотите устанавливать соседские отношения OSPF в сегменте, где находятся только конечные пользователи (нет других маршрутизаторов). Это бессмысленно и создает ненужный служебный трафик.
   - **Пример:** Интерфейс GigabitEthernet0/1 маршрутизатора подключен к сети пользователей 10.10.10.0/24. В конфигурации OSPF добавляется passive-interface GigabitEthernet0/1. Маршрутизатор будет объявлять сеть 10.10.10.0/24 своим OSPF-соседям, но сам не будет пытаться найти OSPF-соседей в этой сети.

3. Устранение неполадок реализации OSPF для одной области

3.1. Составляющие процедуры поиска и устранения неполадок

Базовая идея: Устранение неполадок – это системный процесс, следующий от общего к частному. Для OSPF он включает несколько четких этапов, основанных на модели OSI.

Методология:

1. **Определение проблемы:** Получить от пользователя или системы мониторинга информацию: "Что не работает? Для кого не работает? Когда началось?" (Например: "Сервер в VLAN 10 недоступен с рабочего места в VLAN 20").
2. **Сбор информации:** Использовать команды для получения данных о текущем состоянии.
3. **Анализ и выдвижение гипотез:** Сравнить собранные данные с ожидаемым нормальным состоянием. Выдвинуть предположение о причине.
4. **Проверка гипотез и изоляция неисправности:** Последовательно проверять гипотезы, сужая круг возможных причин.
5. **Решение проблемы и документирование:** Применить исправление и убедиться в его эффективности. Зафиксировать действия.

Применение к OSPF: В рамках OSPF мы проверяем связность последовательно на разных уровнях:

- **Уровень 1 (Физический и канальный):** Работает ли интерфейс? (show interfaces, show ip interface brief). Есть ли уровень 2 связность? (MAC-адрес соседа виден?).
- **Уровень 3 (Сетевой):** Назначены ли IP-адреса? Находятся ли интерфейсы в одной подсети? (show ip interface).
- **Уровень 3 (Протокол OSPF):** Видят ли маршрутизаторы друг друга как соседи? (show ip ospf neighbor). Совпадают ли базовые параметры OSPF? Полна ли база данных LSDB? (show ip ospf database). Есть ли нужные маршруты в таблице маршрутизации? (show ip route ospf).

3.2. Поиск и устранение неполадок в маршрутизации OSPFv2 для одной области

Типичные проблемы и их решение:

1. **Проблема:** Смежность не устанавливается (сосед не появляется в выводе show ip ospf neighbor).
   - **Причины и проверки:**
     - **Разные номера областей (Area ID):** Проверить show ip ospf interface. Интерфейсы на двух концах связи должны быть в одной области.
     - **Разные таймеры Hello/Dead:** Проверить show ip ospf interface. Убедиться, что Hello Interval и Dead Interval совпадают.
     - **Ошибка аутентификации:** Проверить тип и ключ аутентификации на обоих интерфейсах (show ip ospf interface).
     - **Разные маски подсети в broadcast-сетях:** В сетях типа "broadcast" и "non-broadcast" для установления смежности маски должны совпадать. Проверить show ip interface.
     - **Фильтрация многоадресных адресов:** Заблокированы ли пакеты на адреса 224.0.0.5 (All OSPF Routers) и 224.0.0.6 (All DR Routers)? Проверить ACL или настройки межсетевого экрана на интерфейсе.

- **Интерфейс в пассивном режиме:** Проверить show ip ospf interface. Если в статусе интерфейса указано Passive, OSPF не отправляет Hello-пакеты.

2. **Проблема:** Смежность установлена (состояние *Full*), но маршруты отсутствуют в таблице маршрутизации.

- **Причины и проверки:**

- **Несовпадение MTU:** Если на одном интерфейсе MTU 1500, а на другом 1400, смежность может установиться, но обмен базами данных завершится с ошибкой. Проверить show ip ospf neighbor. В подробном выводе может быть указано несоответствие MTU. Команда для проверки: show interfaces.
- **Фильтрация сетей при анонсировании:** Используется ли команда distribute-list или prefix-list, которая запрещает анонсировать определенные сети? Проверить конфигурацию OSPF.
- **Интерфейс-источник (Router ID) дублируется:** Если в одной области есть два маршрутизатора с одинаковым Router ID, возникнет конфликт, и их LSA (Link-State Advertisement) будут игнорироваться. Проверить show ip ospf.

3.3. Поиск и устранение неполадок в OSPFv3 для одной области

Базовая идея: OSPFv3 – это версия OSPF для IPv6. Принципы работы, состояния, типы сетей идентичны OSPFv2. Основные отличия заключаются в адресации (используются IPv6-адреса link-local для установления смежности) и структуре пакетов.

Ключевые отличия при устранении неполадок:

1. **Адресация:** В OSPFv3 маршрутизаторы используют link-local IPv6-адреса (начинаются с FE80::/10) для установления соседских отношений и отправки OSPF-пакетов. Глобальные unicast-адреса используются только для вычисления маршрутов к ним. Поэтому при проблемах со смежностью нужно в первую очередь проверять, есть ли link-local-адрес на интерфейсе (show ipv6 interface) и могут ли маршрутизаторы "достучаться" друг до друга на link-local уровне (проверить ping на link-local-адрес соседа).
2. **Команды:** Все команды OSPFv2 начинаются с ip ospf ..., а команды OSPFv3 – с ipv6 ospf ....

- show ipv6 ospf neighbor – таблица соседей OSPFv3.
- show ipv6 ospf interface – параметры OSPFv3 на интерфейсах.
- show ipv6 route ospf – таблица маршрутизации OSPFv3.

3. **Типичная проблема OSPFv3:** Неверно указанный идентификатор экземпляра (instance ID). В OSPFv3 на одном интерфейсе может работать несколько независимых экземпляров OSPF. Они должны совпадать у соседей. Проверить в show ipv6 ospf interface, что параметр Instance ID совпадает.

Пример сценария неполадки OSPFv3:
Маршрутизаторы R1 и R2 подключены друг к другу. OSPFv3 смежность не устанавливается.

- **Шаг 1:** show ipv6 interface brief – проверяем, что интерфейсы включены и имеют link-local-адреса.

- **Шаг 2:** ping (link-local-адрес соседа)%(интерфейс) – проверяем связность на уровне link-local. Например, ping FE80::A8BB:CCFF:FE00:200%GigabitEthernet0/0.
- **Шаг 3:** show ipv6 ospf interface – проверяем, что интерфейсы добавлены в нужную область OSPFv3, совпадают ли таймеры, приоритеты.
- **Шаг 4:** debug ipv6 ospf hello – (использовать с осторожностью) смотрим, отправляются и принимаются ли Hello-пакеты, и что в них содержится.

Заключение

Настройка и устранение неполадок OSPF для одной области требуют глубокого понимания фундаментальных принципов работы протокола: установления смежности, выбора DR/BDR, обмена и синхронизации баз данных состояния каналов, вычисления кратчайших путей по алгоритму Дейкстры. Расширенные параметры, такие как стоимость, таймеры, аутентификация и распространение маршрута по умолчанию, являются инструментами тонкой настройки, позволяющими адаптировать OSPF к конкретным требованиям сети.

Методология устранения неполадок должна быть системной, начиная с проверки физической связности и заканчивая анализом таблиц маршрутизации. Для OSPFv2 и OSPFv3 подход одинаков, но необходимо учитывать особенности адресации IPv6 в OSPFv3. Успешное владение этими знаниями и навыками позволяет обеспечивать стабильную и эффективную работу динамической маршрутизации в корпоративных сетях и сетях среднего размера.

Контрольные вопросы к ТЕМЕ-14

1. Что такое область (Area) в OSPF и какова основная цель её использования?
2. Какие два ключевых устройства выбираются в сетях с множественным доступом (Multi-Access) для оптимизации OSPF, и какова их роль?
3. Объясните простыми словами, что такое "стоимость" (Cost) маршрута в OSPF и от чего она зависит.
4. Какой командой на граничном маршрутизаторе можно распространить маршрут по умолчанию (0.0.0.0/0) для всех маршрутизаторов в области OSPF?
5. Что произойдёт, если на двух соседних интерфейсах маршрутизаторов, между которыми должна установиться смежность OSPF, будут заданы разные таймеры Hello?
6. Назовите два основных типа аутентификации в OSPF. Какой из них является более безопасным и почему?
7. Для чего используется команда passive-interface в конфигурации OSPF?
8. Какой командой можно посмотреть таблицу установленных соседних отношений (смежностей) OSPF на маршрутизаторе?
9. Что является самой частой причиной проблемы, когда OSPF-смежность установлена (состояние Full), но нужные маршруты отсутствуют в таблице маршрутизации?
10. Какое основное назначение Link-State Database (LSDB) в OSPF?

11. Чем принципиально отличается установление соседских отношений в OSPFv3 от OSPFv2 с точки зрения используемых адресов?
12. Какой командой в OSPFv2 можно вручную задать стоимость (Cost) интерфейса, отличную от автоматически рассчитанной?

Практические задания к ТЕМЕ-14

Задание 1. Анализ проблемного сегмента в локальной сети колледжа

Вы — администратор сети филиала колледжа в г. Казань. После подключения нового маршрутизатора учебной лаборатории к существующей сети OSPF области 0 вы обнаружили, что этот маршрутизатор не устанавливает смежность с соседним маршрутизатором, расположенным в серверной. При этом кабель проложен корректно, индикаторы портов горят зелёным. В настройках обоих интерфейсов указана одинаковая IP-сеть 172.16.10.0/24 и номер области 0. Какие **три** параметра настройки OSPF на интерфейсах вы проверите в первую очередь, чтобы выявить причину? Объясните, почему каждый из этих параметров критичен.

Задание 2. Оптимизация OSPF в сети малого офиса

В сети офиса российской IT-компании «Вега» в Санкт-Петербурге используется один маршрутизатор Mikrotik (R1) в качестве шлюза в интернет и три маршрутизатора Cisco (R2, R3, R4) для внутренней маршрутизации между отделами. Все они работают в одной области OSPF. Администратор, который настраивал сеть ранее, на всех интерфейсах внутренних маршрутизаторов прописал статический маршрут по умолчанию на R1. Этот подход признан неудобным для масштабирования. Какой **одной** командой на маршрутизаторе R1 можно исправить ситуацию, чтобы все внутренние маршрутизаторы автоматически получали маршрут по умолчанию через OSPF? Опишите принцип работы этой команды в контексте данной ситуации.

Задание 3. Выбор DR в сети учебного центра

В учебном центре «АйТиКласс» в Москве в одном коммутируемом VLAN (сеть 10.20.30.0/24) находятся четыре маршрутизатора: мощный CoreSwitch (Router-ID 10.0.0.1), два идентичных маршрутизатора учебных стендов Lab-R1 и Lab-R2 (Router-ID 10.0.0.2 и 10.0.0.3), а также старый резервный маршрутизатор OldBackup (Router-ID 10.0.0.4). После перезагрузки всех устройств DR неожиданно стал OldBackup, что привело к повышенной нагрузке на него и замедлению обновлений маршрутов. Исходя из логики работы OSPF, предложите **два** технических решения (разных по сути), которые гарантированно обеспечат выбор в качестве DR устройства CoreSwitch. Объясните, как каждое решение повлияет на процесс выбора.

Задание 4. Диагностика потери маршрута

Вы получаете сообщение от сотрудников отдела разработки компании «РосИнтег» в Новосибирске о том, что они потеряли доступ к серверу в сети 192.168.105.0/24, расположенному в другом здании. Оба здания соединены маршрутизаторами, работающими в одной области OSPF. Вы заходите на пограничный маршрутизатор отдела разработки и видите, что смежность OSPF с маршрутизатором из соседнего здания установлена (состояние Full). Однако маршрута к сети 192.168.105.0/24 в таблице маршрутизации нет. Основываясь на теоретическом материале, сформулируйте **две** наиболее вероятные причины такой ситуации. Какие команды вы выполните для проверки каждой из этих гипотез?

Задание 5. Миграция на IPv6 в кампусе вуза

В сети Екатеринбургского технического университета завершён первый этап перехода на dual-stack (одновременная поддержка IPv4 и IPv6). Настроен OSPFv2 для IPv4, и он работает стабильно. Вы приступаете к настройке OSPFv3 для IPv6 в той же одной области. После базовой настройки на двух соседних маршрутизаторах смежность OSPFv3 не устанавливается, хотя ping на глобальных IPv6-адресах проходит успешно. Вспомните ключевое отличие OSPFv3 от OSPFv2 при установлении соседства. Какой адрес и какую команду вы используете для первичной проверки базовой сетевой связности, необходимой для работы OSPFv3?

ТЕМА-15. OSPF для нескольких областей

Введение и базовые предпосылки

Представьте себе небольшой город. Все его улицы хорошо известны каждому таксисту, информация о дорожной ситуации быстро распространяется среди всех водителей. Это аналогия OSPF в одной области (single-area OSPF), которая прекрасно работает для сетей ограниченного размера. Однако что происходит, когда город превращается в мегаполис? Если каждый таксист будет знать о каждой улице, каждом переулке и каждом временном ремонте во всем гигантском городе, его карта станет невероятно сложной, а обновления о пробках будут огромными и будут постоянно загружать радиоэфир. Система станет неэффективной и неустойчивой.

Точно так же и в компьютерных сетях. Протокол OSPF, когда работает в рамках одной области (Area 0 – магистральная область), при росте сети сталкивается с проблемами:

1. **Огромные таблицы маршрутизации и топологии:** Каждый маршрутизатор хранит информацию о каждой сети в автономной системе.
2. **Высокая нагрузка от служебного трафика:** Маршрутизаторы постоянно пересчитывают свои базы данных при любом изменении в сети, даже самом незначительном.
3. **Медленная сходимость:** Процесс восстановления после сбоя затягивается из-за большого объема данных для обработки.

Решение — разделение большой автономной системы OSPF на несколько областей (Multi-Area OSPF). Это позволяет создать иерархическую структуру, аналогичную административному делению мегаполиса на районы, округа или префектуры. Мэр города (маршрутизатор уровня магистрали) не знает о каждой лавочке в каждом дворе, но он точно знает, как связаны между собой районы и как проехать в любой из них. Глава же района знает все детали внутри своего района, но для поездки в другой район обращается к централизованной информации.

1. Принцип работы OSPF для нескольких областей. Назначение OSPF для нескольких областей

Принцип работы основан на строгой иерархии и сокрытии деталей внутри областей.

1. **Иерархическая структура:**

- **Area 0 (Магистральная область / Backbone Area):** Это обязательный центральный элемент. Все остальные области **обязаны** подключаться к Area 0. Она служит магистралью для передачи маршрутной информации между областями. Аналогия: кольцевая линия метро или центральный автовокзал, связывающий все районы города.

- **Немагистральные области (Non-Backbone Areas):** Области с другими идентификаторами (Area 1, Area 10, Area 100 и т.д.). Они подключаются к Area 0 напрямую или через виртуальные линки (редкий случай, рассматривается в продвинутых курсах). Каждая такая область — это отдельный "район" сети.

2. **Назначение Multi-Area OSPF:**

- **Сокращение служебного трафика:** Изменения в одной области (например, добавление сети в Area 1) не вызывают полного пересчета базы данных OSPF в маршрутизаторах Area 2. Детали изменения остаются внутри Area 1.
- **Уменьшение размера таблиц топологии и маршрутизации:** Маршрутизатор в Area 2 не знает полной топологии Area 1. Он знает только **суммарные маршруты**, ведущие в эту область, и путь к ней через Area 0.
- **Повышение производительности и масштабируемости:** За счет снижения нагрузки на CPU и память маршрутизаторов сеть может расти практически неограниченно.
- **Изоляция нестабильности:** Проблемы с линками или маршрутизаторами в одной области оказывают минимальное влияние на стабильность работы других областей.

Пример аналогии: Федеральная почтовая служба.

- **Area 0** — это федеральный сортировочный центр в Москве.
- **Area 1** — это городской почтамт Санкт-Петербурга со всеми своими отделениями.
- **Area 2** — это почтамт Екатеринбурга.
- Работник в отделении связи в Екатеринбурге (маршрутизатор внутри Area 2) не знает всех адресов всех домов в Санкт-Петербурге. Он знает только, что письмо для Санкт-Петербурга нужно отправить в свой городской почтамт (граничный маршрутизатор Area 2), а тот отправит его в федеральный центр Москвы (Area 0), который, в свою очередь, переправит его в почтамт Санкт-Петербурга (Area 1). А уже внутри Санкт-Петербурга письмо найдет свой точный адрес.

2. Принцип работы пакетов LSA в OSPF для нескольких областей

LSA (Link-State Advertisement) — это не пакеты, а **объявления о состоянии каналов**. Это "сообщения", которые OSPF-маршрутизаторы рассылают для описания своей части сети. Они хранятся в базе данных состояния каналов (LSDB). Разные типы LSA играют ключевую роль в реализации иерархии.

Важнейшие типы LSA для Multi-Area OSPF:

1. **Тип 1 (Router LSA):**

- **Что описывает?** Свои активные интерфейсы OSPF и соседей по области. Генерируется **каждым** маршрутизатором.
- **Где распространяется? Только внутри своей области.** Это фундаментальное правило для сокрытия деталей. Маршрутизатор в Area 0 никогда не увидит Type 1 LSA из Area 1.

- **Аналогия:** Внутрирайонная газета, которая публикует новости только о своем районе (какие магазины открылись, где ремонт дорог). В другие районы эта газета не поступает.

2. **Тип 2 (Network LSA):**

- **Что описывает?** Сегменты сети с множественным доступом (Ethernet), на которых выбран DR (Designated Router). Генерируется DR.
- **Где распространяется? Только внутри своей области**, как и Type 1.

3. **Тип 3 (Summary LSA):**

- **Это "сердце" иерархии Multi-Area OSPF.**
- **Что описывает?** Маршруты к сетям, которые находятся **в других областях**. Не содержит информации о топологии (какие маршрутизаторы и как связаны), а только сам факт: "Сеть 192.168.1.0/24 находится в Area 1, и добраться до нее можно через меня".
- **Кто генерирует? ABR (Area Border Router)** — граничный маршрутизатор области. Это маршрутизатор, у которого интерфейсы находятся как минимум в двух разных областях OSPF (обычно одна из них — Area 0).
- **Где распространяется?** ABR принимает Type 1 и Type 2 LSA из своей немагистральной области (например, Area 1), "суммирует" информацию о содержащихся в них сетях, создает Type 3 LSA и ретранслирует их **в Area 0**. Другой ABR, подключенный к Area 0 и, например, к Area 2, получит эти Type 3 LSA из Area 0 и, в свою очередь, ретранслирует их (как свои) **в Area 2**.
- **Аналогия:** Глава района (ABR) составляет для мэрии (Area 0) сводный отчет: "В моем районе проживает 50 000 человек". Он не присылает список всех жителей с адресами (Type 1 LSA), а только обобщенную цифру (Type 3 LSA). Мэрия рассылает эти сводки по другим районам, чтобы их главы знали общую демографическую картину города.

4. **Тип 4 (ASBR Summary LSA):**

- **Что описывает?** Местоположение ASBR (Autonomous System Boundary Router). ASBR — это маршрутизатор, который соединяет область OSPF с внешним миром (другой автономной системой, например, использующей протокол EIGRP или статические маршруты).
- **Кто генерирует?** ABR, который знает путь к ASBR.
- **Назначение:** Сообщить маршрутизаторам в других областях, как добраться до "шлюза во внешний мир".

5. **Тип 5 (AS External LSA):**

- **Что описывает?** Маршруты к сетям, находящимся **за пределами автономной системы OSPF** (внешние маршруты).
- **Кто генерирует?** ASBR.
- **Где распространяется? По всей автономной системе OSPF** (во все области), за исключением особых типов областей (stub, totally stubby, NSSA).

Итог процесса: Маршрутизатор внутри Area 2 строит свою полную картину мира следующим образом:

- **Для своей Area 2:** У него есть полная база данных из Type 1 и Type 2 LSA от всех маршрутизаторов Area 2. Он знает **полную топологию** своего "района".

- **Для других областей (Area 0, Area 1):** У него есть только Type 3 LSA от своего ABR, которые говорят: "чтобы попасть в сети Area 1, иди на ABR". Детальной топологии Area 1 у него нет.
- **Для внешних сетей:** У него есть Type 5 LSA (и, возможно, Type 4), которые говорят, как выйти во внешний мир.

3. Таблица маршрутизации и типы маршрутов OSPF

Алгоритм SPF (Dijkstra), обрабатывая базу данных LSA, строит таблицу маршрутизации. В Multi-Area OSPF появляются новые типы маршрутов, отображаемые в таблице.

Обозначения в таблице маршрутизации Cisco IOS:

1. **O (Intra-Area):**

- **Значение:** Маршрут к сети, находящейся **в той же области**, что и данный маршрутизатор.
- **Как появился:** Рассчитан алгоритмом SPF на основе **Type 1 и Type 2 LSA**.
- **Административное расстояние:** 110.
- **Пример:** O 10.1.1.0/24 [110/50] via 192.168.1.1, FastEthernet0/0
- **Аналогия:** Маршрут внутри своего района, рассчитанный по детальной карте района.

2. **O IA (Inter-Area):**

- **Значение:** Маршрут к сети, находящейся **в другой области OSPF**.
- **Как появился:** Получен через **Type 3 LSA** от ABR. **Не** является результатом расчета SPF для чужой области.
- **Пример:** O IA 172.16.1.0/24 [110/200] via 10.0.0.1, Serial0/0/0
- **Аналогия:** Маршрут в другой район. Вы знаете не детальную карту того района, а только указание: "чтобы попасть в район X, езжай до центральной площади (ABR)".

3. **O E1 / O E2 (External Type 1 / Type 2):**

- **Значение:** Маршрут к сети **вне автономной системы OSPF** (внешний маршрут).
- **Как появился:** Получен через **Type 5 LSA** от ASBR.
- **Разница между E1 и E2:**

- **O E2 (по умолчанию):** Метрика равна только внешней метрике, назначенной ASBR. Не учитывает стоимость пути внутри OSPF-домена до самого ASBR. Выбор маршрута основывается только на этой внешней метрике.
- **O E1:** Метрика рассчитывается как **внешняя метрика + стоимость пути внутри OSPF до ASBR**. Более информативна для выбора оптимального пути к внешней сети, если к ней ведут несколько ASBR.
- **Аналогия:** Маршрут в другой город.

- **E2:** Билет на поезд стоит 1000 рублей. Цена фиксированная, независимо от того, из какого района города вы едете на вокзал.
- **E1:** Общая стоимость поездки: 1000 рублей (билет) + 50 рублей (проезд на метро до вокзала из вашего района). Если из другого района метро стоит 100 рублей, то общая стоимость будет 1100 рублей. Маршрутизатор выберет путь через ближайший к нему ASBR.

4. Настройка OSPF для нескольких областей

Ключевые шаги на маршрутизаторах Cisco:

1. **Включение OSPF и задание Router-ID:**

```
cisco
Router> enable
Router# configure terminal
Router(config)# router ospf 1        ! 1 - номер процесса (локальная
значимость)
Router(config-router)# router-id 1.1.1.1 ! Явное задание стабильного
Router-ID
```

2. **Объявление сетей на интерфейсах с указанием области:**

- Команда network <адрес_сети> <обратная_маска> area <номер_области> говорит OSPF, на каких интерфейсах "запустить" протокол и в какую область их включить.
- **Обратная маска (wildcard mask):** Определяет, какие биты IP-адреса проверять. 0 — бит должен совпадать, 1 — бит может быть любым.

- Сеть 192.168.1.0/24 имеет обратную маску 0.0.0.255.
- Один интерфейс с IP 10.0.0.1/30 можно объявить как network 10.0.0.0 0.0.0.3 или, точнее, network 10.0.0.1 0.0.0.0 area 0.

Пример конфигурации для топологии с тремя областями:

- **Маршрутизатор ABR1** (интерфейс в Area 0 и интерфейс в Area 1):

```
cisco
ABR1(config)# router ospf 1
ABR1(config-router)# router-id 1.1.1.1
ABR1(config-router)# network 10.0.0.0 0.0.0.3 area 0 ! Интерфейс в Area 0
ABR1(config-router)# network 192.168.1.0 0.0.0.255 area 1 ! Интерфейс в
Area 1
```

- **Внутренний маршрутизатор в Area 1 (Internal Router in Area 1):**

```
cisco
IR-Area1(config)# router ospf 1
IR-Area1(config-router)# router-id 2.2.2.2
IR-Area1(config-router)# network 192.168.1.0 0.0.0.255 area 1 ! Все
интерфейсы в Area 1
IR-Area1(config-router)# network 172.16.1.0 0.0.0.255 area 1
```

- **Маршрутизатор ABR2** (интерфейс в Area 0 и интерфейс в Area 2):

```
cisco
ABR2(config)# router ospf 1
ABR2(config-router)# router-id 3.3.3.3
ABR2(config-router)# network 10.0.0.4 0.0.0.3 area 0 ! Другой линк в Area 0
ABR2(config-router)# network 192.168.2.0 0.0.0.255 area 2 ! Интерфейс в Area 2
```

- **ASBR** (маршрутизатор в Area 2, подключенный к внешней сети):

```
cisco
ASBR(config)# router ospf 1
ASBR(config-router)# router-id 4.4.4.4
ASBR(config-router)# network 192.168.2.0 0.0.0.255 area 2
ASBR(config-router)# redistribute static subnets ! Импорт статических маршрутов в OSPF как внешних (Type 5)
ASBR(config)# ip route 0.0.0.0 0.0.0.0 203.0.113.1 ! Статический маршрут "в Интернет"
```

Команда redistribute на ASBR отвечает за распространение внешних маршрутов (в данном случае статических) в OSPF-домен.

5. Объединение маршрутов OSPF (Route Summarization)

Объединение (суммаризация) — это ключевая техника для дальнейшего уменьшения таблиц маршрутизации и повышения стабильности. Оно выполняется **на ABR** (для межзональных маршрутов) и **на ASBR** (для внешних маршрутов).

- **Принцип:** Вместо рассылки множества Type 3 LSA о каждой конкретной сети внутри области (например, 172.16.1.0/24, 172.16.2.0/24, 172.16.3.0/24), ABR может создать и разослать одну объединенную LSA об одной крупной сети (например, 172.16.0.0/16), которая покрывает все эти подсети.
- **Преимущества:**

1. Еще меньше записей в таблице маршрутизации у маршрутизаторов в других областях.
2. **Стабильность:** Если внутри Area 1 "погасла" одна подсеть 172.16.3.0/24, это изменение не будет анонсировано за пределы области, так как объединенный маршрут 172.16.0.0/16 по-прежнему корректен. Это подавляет ненужные обновления.

- **Настройка на ABR (для маршрутов из Area 1):**

```
cisco
ABR1(config)# router ospf 1
ABR1(config-router)# area 1 range 172.16.0.0 255.255.0.0
```

Эта команда говорит ABR1: "Если у тебя есть сети в Area 1, попадающие в диапазон 172.16.0.0/16, то ретранслируй в другие области только один объединенный маршрут 172.16.0.0/16, скрывая более конкретные".

- **Аналогия:** Глава района (ABR) вместо отчета "у меня живут Ивановы, Петровы, Сидоровы..." отправляет в мэрию общий отчет: "в моем районе проживают семьи на буквы И, П, С". Если семья Сидоровых переедет, общий отчет менять не нужно.

6. Проверка OSPF для нескольких областей

После настройки необходимо проверить работоспособность. Основные команды:

1. **show ip ospf neighbor**
   - Показывает состояние соседства по OSPF.
   - **Критически важно:** Убедиться, что на интерфейсах, ведущих к соседям, установилось состояние **FULL** (для DR/BDR может быть 2WAY). Это значит, что базы данных синхронизированы.
2. **show ip route ospf**
   - Показывает только OSPF-маршруты в таблице маршрутизации.
   - **Что ищем?** Наличие маршрутов с метками O (внутри области), O IA (межзональные), O E1/E2 (внешние). Это подтвердит, что иерархия работает.
3. **show ip ospf database**
   - **Самая информативная команда.** Показывает базу данных состояния каналов (LSDB).
   - **Что проверять на внутреннем маршрутизаторе Area 2?**
     - Он должен видеть **полный набор Type 1 и Type 2 LSA** для своей Area 2.
     - Он должен видеть **Type 3 LSA** (Summary Net Link States), созданные ABR для сетей из других областей (Area 0, Area 1).
     - Он может видеть **Type 5 LSA** (Type-5 AS External Link States), если в системе есть ASBR.
   - **Что проверять на ABR?** ABR будет иметь **разные LSDB для каждой из своих областей**. Нужно смотреть с указанием области: show ip ospf database database-summary или по интерфейсам.
4. **show ip protocols**
   - Дает сводную информацию о настроенных протоколах маршрутизации. Показывает параметры OSPF, идентификатор процесса, Router-ID и, что важно, **список областей, к которым подключен данный маршрутизатор**.

Типичная последовательность проверки:

1. Убедились, что соседство (FULL) установлено.
2. Проверили, что маршруты в таблице есть и имеют ожидаемые типы (O, O IA).
3. Убедились, что база данных содержит ожидаемые типы LSA.
4. Проверили связность командой ping с IP-адресов интерфейсов в разных областях.

Заключение

Использование OSPF с несколькими областями превращает протокол из решения для сетей среднего масштаба в мощный, масштабируемый инструмент для построения крупных корпоративных и операторских сетей. Основная философия заключается в иерархии, сокрытии деталей и контроле над распространением служебной информации. Ключевыми "архитекторами" этой иерархии являются граничные маршрутизаторы областей (ABR), которые фильтруют информацию с помощью Summary LSA (Type 3), и граничные маршрутизаторы автономной системы (ASBR), которые обеспечивают связь с внешними маршрутами. Понимание ролей маршрутизаторов (Internal, ABR, ASBR), типов LSA и соответствующих им записей в таблице маршрутизации (O, O IA, O E1/E2) является фундаментальным для проектирования, настройки и поиска неисправностей в сложных сетях OSPF.

Контрольные вопросы к ТЕМЕ-15

1. Какие три основные проблемы возникают в сети OSPF при увеличении её размера, и как решает эти проблемы использование нескольких областей (Multi-Area)?
2. Какую роль выполняет область Area 0 (магистральная область) в иерархической структуре Multi-Area OSPF?
3. Дайте определения следующим типам маршрутизаторов OSPF: внутренний маршрутизатор (Internal Router), граничный маршрутизатор области (ABR), граничный маршрутизатор автономной системы (ASBR).
4. Где (внутри какой части сети) распространяются LSA Типа 1 (Router LSA) и LSA Типа 2 (Network LSA)? Почему это важно?
5. Какой тип LSA является основным для создания иерархии Multi-Area OSPF? Кто его генерирует и какую информацию он содержит?
6. Чем отличается информация, которую несёт LSA Типа 3 (Summary LSA), от информации в LSA Типа 1? Приведите аналогию.
7. Как обозначаются в таблице маршрутизации Cisco маршрут к сети внутри своей области (Intra-Area) и маршрут к сети в другой области (Inter-Area)?
8. Маршрутизатор в области Area 2 видит в своей таблице маршрутизации запись «O IA 172.16.1.0/24». Объясните, что означает каждый элемент этой записи.
9. По какой команде на маршрутизаторах Cisco производится включение интерфейса в определённую область OSPF (например, в Area 1)?
10. На каком типе маршрутизатора (Internal, ABR, ASBR) настраивается объединение (суммаризация) маршрутов для уменьшения количества LSA, отправляемых в другие области?
11. Какой командой проверяется состояние соседства OSPF, и какое состояние (STATE) должно быть у работающего соседа для обмена маршрутной информацией?
12. Какой командой можно просмотреть полную базу данных состояния каналов (LSDB) маршрутизатора и увидеть все полученные LSA разных типов?

Практические задания к ТЕМЕ-15

Задание 1. Планирование структуры сети филиала

Вы — сетевой администратор в компании «РосТехСеть», которая расширяется. Главный офис в Москве уже работает на OSPF. В Санкт-Петербурге открывается новый филиал, в котором будет своя локальная сеть из 4 отделов (подсетей). Руководство требует, чтобы сбой в сети питерского филиала (например, отключение коммутатора в одном отделе) не вызывал пересчет маршрутов и служебную нагрузку на оборудование в московском головном офисе.

- **Вопрос:** Как вы спроектируете подключение филиала к сети компании с использованием Multi-Area OSPF? Назовите роль, которую будет играть маршрутизатор филиала, подключенный к каналу связи с Москвой. Какую область OSPF вы назначите сетям внутри филиала, а какую — каналу связи с головным офисом? Поясните свой выбор, исходя из принципов иерархии.

Задание 2. Анализ таблицы маршрутизации

После настройки Multi-Area OSPF в сети колледжа вы на маршрутизаторе в учебном корпусе (Area 10) выполняете команду show ip route ospf и видите, среди прочих, следующие записи:

```
O    192.168.10.0/24 [110/50] via 10.0.1.1, FastEthernet0/1
O IA 172.16.100.0/24 [110/200] via 10.0.1.1, Serial0/0/0
O E2 8.8.8.0/24 [110/20] via 192.168.10.2, FastEthernet0/2
```

- **Вопрос:** Дайте развернутое пояснение по каждой строке. В какой области находится сеть 192.168.10.0/24 относительно вашего маршрутизатора? Какой тип маршрутизатора (ABR или ASBR) и в какой области, скорее всего, находится устройство с IP 10.0.1.1? Куда ведет маршрут 8.8.8.0/24 и почему его метрика не учитывает стоимость пути внутри сети колледжа?

Задание 3. Диагностика проблемы с LSA

Администратор воронежского филиала компании «АгроСвязь» пожаловался, что после добавления в его сеть (Area 30) новой тестовой подсети 10.30.99.0/24, эта сеть не появилась в таблицах маршрутизации сотрудников в белгородском филиале (Area 40). При этом все старые сети из Area 30 в Area 40 видны. Вы, как ответственный за магистраль (Area 0), проверили связность: канал между ABR области 30 и Area 0 работает.

- **Вопрос:** Опираясь на принципы работы LSA, предположите наиболее вероятную причину проблемы. Какой тип LSA отвечает за распространение информации о сетях между областями? На каком конкретном устройстве (укажите его тип) могла быть допущена ошибка в конфигурации, из-за которой информация о новой сети не покинула Area 30?

Задание 4. Выбор типа внешнего маршрута (E1/E2)

В сети университета настроен Multi-Area OSPF. Имеется два канала выхода в интернет: быстрый и дорогой через маршрутизатор-ASBR в главном здании (Area 0) и медленный резервный через ASBR в общежитии (Area 5). Необходимо настроить маршрутизацию так,

чтобы трафик из компьютерного класса в учебном корпусе (Area 10) в интернет в первую очередь шел через быстрый канал, даже если путь до ASBR в главном здании длиннее, а из самой Area 5 — через свой локальный выход.

- **Вопрос:** Какой тип внешних маршрутов (E1 или E2) необходимо анонсировать с ASBR, чтобы реализовать такое требование? Дайте развернутое объяснение, как в этом случае будет рассчитываться метрика для маршрутизатора в Area 10 и для маршрутизатора внутри Area 5.

Задание 5. Применение объединения маршрутов

В сети крупного банка используется Multi-Area OSPF. В области «Дополнительные офисы» (Area 200) постепенно добавляются подсети для новых отделений по схеме: 10.200.1.0/24, 10.200.2.0/24, ..., 10.200.50.0/24. Администратор области заметил, что при каждом открытии нового офиса в других областях происходит незначительное, но заметное увеличение загрузки процессоров на ABR.

- **Вопрос:** Какую операцию можно выполнить на ABR области 200, чтобы добавление 51-го, 52-го и последующих офисов больше не вызывало рассылку служебной информации (LSA) в магистральную область и другие области? Сформулируйте команду для настройки, предполагая, что больше 100 офисов не планируется. Объясните, почему это решение снижает нагрузку.

ТЕМА-16. Подключение к глобальной сети

Введение

Представьте себе, что вы хотите отправить письмо другу, который живёт в другом городе или даже стране. Вы не можете просто протянуть руку и передать ему конверт – вам нужна система доставки: почтовое отделение, транспорт, маршруты, адреса. Компьютерные сети внутри одного здания (локальные сети) – это как общение в пределах одной комнаты. Но когда необходимо связать компьютеры, находящиеся в разных городах, странах и на континентах, мы вступаем в область **глобальных сетей (Wide Area Network, WAN)**. Это и есть тема нашего изучения: как организовано это «мировое почтовое сообщение» для данных, какие технологии и принципы лежат в его основе, и как мы можем к нему подключиться.

1. Обзор технологий глобальной сети

Базовая идея: Глобальная сеть (WAN) – это совокупность телекоммуникационных каналов, устройств и протоколов, которые обеспечивают обмен данными между географически удалёнными сетями (например, между локальной сетью филиала в Москве и головным офисом в Новосибирске) или доступ к ресурсам, расположенным на большом расстоянии (например, к сайту в интернете).

Пояснение смысла: Ключевое отличие WAN от локальной сети (LAN) – это расстояние и, как следствие, необходимость использования каналов связи, которые не принадлежат вам. Вы не можете проложить собственный кабель между городами (это невероятно дорого и сложно). Поэтому вы арендуете «информационные магистрали» у специализированных компаний – **провайдеров услуг связи (операторов)**.

Примеры и аналогии:

- **Дороги и магистрали:** LAN – это дороги внутри вашего частного владения или микрорайона. WAN – это федеральные трассы, железные дороги, воздушные коридоры, которые принадлежат государству или крупным компаниям. Чтобы доставить груз (данные) в другой город, вы должны выехать на эту общую магистраль, соблюдая общие правила движения (протоколы).
- **Телефонная сеть:** Классическая телефонная связь – прообраз глобальной сети. Чтобы позвонить в другой город, вы набираете код города (префикс), а затем ваш голос проходит через множество коммутаторов и каналов, принадлежащих оператору.
- **Конкретные технологии:**
  - **Выделенные линии (leased line):** Постоянное прямое соединение «точка-точка». Аналогия: вы арендовали отдельный железнодорожный путь между двумя вашими складами. Он всегда свободен только для ваших грузов, надёжен, но дорог.

- **Цифровая сеть с интеграцией услуг (ISDN):** «Цифровая телефонная линия». Позволяет одновременно передавать голос и данные. Аналогия: улучшенная телефонная линия, по которой можно и говорить, и отправлять факсы/данные без модема.
- **Frame Relay и X.25:** Технологии коммутации пакетов с гарантированной пропускной способностью (CIR). Аналогия: вы арендуете не целый путь, а гарантированную полосу на общей железной дороге. Ваши грузы смешиваются с другими, но вам обещают пропускать не менее, скажем, 10 вагонов в час.
- **Asynchronous Transfer Mode (ATM):** Технология, которая делит все данные на маленькие, фиксированные «ячейки» (cell). Очень быстрая и предсказуемая, использовалась для передачи голоса, видео и данных. Аналогия: все грузы, независимо от размера (контейнер, коробка, письмо), разбиваются на стандартные одинаковые ящики, что ускоряет их сортировку и пересылку на гигантских автоматизированных сортировочных станциях.
- **Ethernet over SDH/SONET и MPLS:** Современные высокоскоростные технологии. MPLS (Multiprotocol Label Switching) – это «экспресс-доставка». Данным на входе в сеть оператора наклеивается метка (label), и все коммутаторы внутри сети оператора быстро передают пакет дальше, просто глядя на эту метку, без глубокого анализа. Аналогия: на вашем грузе наклеивают ярлык с конечным адресом. На каждой сортировочной станции работники не читают весь адрес, а просто смотрят на ярлык и быстро перекидывают груз в нужный выход.
- **Оптоволоконные технологии (DWDM):** Позволяют по одному волокну передавать множество независимых световых потоков (каналов) с разной длиной волны. Аналогия: по одной трубе можно пустить множество разноцветных шариков (каждый цвет – свой канал), они не смешаются.

2. Цель создания глобальных сетей

Базовая идея: Главная цель – преодоление географических ограничений для обеспечения обмена информацией и совместного использования ресурсов.

Пояснение смысла: Бизнес, наука, образование и обычная жизнь не ограничиваются стенами одного здания. Необходимость в оперативном взаимодействии на расстоянии и двигала эволюцию WAN.

Примеры и аналогии:

- **Консолидация бизнеса:** Крупная компания с филиалами в 50 городах. Цель WAN – создать единое информационное пространство: чтобы база данных клиентов в центральном офисе была доступна менеджерам во всех филиалах, чтобы бухгалтерские отчёты сводились автоматически, чтобы проводились общие видеоконференции. Без WAN пришлось бы возить жесткие диски курьером.
- **Доступ к уникальным ресурсам:** Учёный в региональном университете нуждается в мощности суперкомпьютера, расположенного в научном центре за тысячи километров. WAN даёт ему удалённый доступ для расчётов.
- **Резервирование и аварийное восстановление:** Компания хранит резервные копии своих данных не в соседней комнате (где они могут погибнуть от того же пожара), а в специальном дата-центре в другом регионе. Ежедневно по WAN эти копии обновляются.

- **Общедоступные сервисы (Интернет):** Самая масштабная цель – предоставление всемирного доступа к информации, общению, развлечениям, торговле. Глобальная сеть здесь выступает как универсальная «информационная среда обитания».

3. Принцип работы глобальной сети

Базовая идея: Работа WAN строится на принципах **коммутации** (как установить путь между абонентами) и использования **протоколов** (как общаться по установленному пути).

Пояснение смысла: Данные не летают по сети «просто так». Сеть должна знать, куда их направить, и отправитель с получателем должны понимать формат этих данных.

Примеры и аналогии:

- **Коммутация каналов (телефонная сеть):** Перед началом передачи устанавливается физический сквозной канал от отправителя к получателю. Этот канал занят всё время сеанса связи, даже если вы молчите. Аналогия: вы звоните другу. АТС создаёт временный «электрический мост» между вашими трубками. Линия занята до тех пор, пока вы не положите трубку.
- **Коммутация пакетов (основа Интернета и современных WAN):** Данные разбиваются на небольшие блоки – **пакеты**. Каждый пакет снабжается заголовком с адресом назначения и номером (как порядковым номером страницы в книге). Пакеты независимо путешествуют по сети, могут идти разными путями. На принимающей стороне они собираются в исходное сообщение по номерам. Аналогия: Вы хотите отправить другу многотомную энциклопедию. Вы разбираете её на отдельные тома (пакеты) и отправляете каждый том отдельным письмом по почте. Письма могут ехать разными поездами, приходить в разной последовательности. Друг, получив все тома, расставляет их по номерам и читает.
- **Роль маршрутизаторов:** Это ключевые устройства WAN. Они находятся на границе между сетями (например, между вашей LAN и сетью провайдера). Маршрутизатор смотрит на адрес назначения в заголовке пакета, заглядывает в свою **таблицу маршрутизации** (дорожную карту сети) и решает, в какой «следующий перекрёсток» (следующий маршрутизатор) отправить пакет. Аналогия: Таможенный пункт на границе между странами. Офицер смотрит на конечный адрес на посылке (пакете), сверяется с картой маршрутов (таблицей) и отправляет её дальше по нужной дороге в соседней стране.
- **Протоколы как правила дипломатии:** Чтобы маршрутизаторы разных производителей и сети разных операторов понимали друг друга, нужны единые правила – протоколы. **IP (Internet Protocol)** – главный протокол, определяющий систему адресации (IP-адрес) и формат пакета. **TCP (Transmission Control Protocol)** – протокол, отвечающий за надёжную доставку: он разбивает данные на пакеты, нумерует их, ждёт подтверждения о получении и в случае потери отправляет заново. Аналогия: Представьте конференцию с делегатами из 200 стран. Чтобы они могли договориться, им нужен: 1) единый язык общения (например, английский – это аналог IP), и 2) регламент встречи: кто говорит, когда, как задавать вопросы, как подтвердить, что тебя поняли (это аналог TCP).

4. Выбор технологии глобальной сети

Базовая идея: Выбор зависит от трёх ключевых факторов: **требований к производительности, уровня необходимой надёжности и бюджета**.

Пояснение смысла: Не существует «лучшей» технологии для всех. Нужно найти баланс между скоростью, задержкой, безопасностью и стоимостью.

Примеры и аналогии:

- **Сценарий 1: Банк, соединяющий центральный офис с отделением.** Требования: максимальная надёжность, безопасность, постоянная доступность, предсказуемая задержка (для транзакций). Выбор: **Выделенная линия (например, оптическая) или MPLS/VPN.** Аналогия: Для перевозки золотого запаса вы выберете бронированный инкассаторский автомобиль с охраной по специально оговоренному маршруту (выделенная линия). Это дорого, но безопасно и надёжно.
- **Сценарий 2: Средняя компания, которой нужен доступ в интернет и связь с удалёнными сотрудниками.** Требования: хорошая скорость, приемлемая надёжность, низкая стоимость. Выбор: **Широкополосный доступ (Ethernet, xDSL) через VPN.** Аналогия: Вы пользуетесь общественным транспортом (метро, электричка) для ежедневных поездок. Это дёшево и эффективно для большинства нужд, хотя иногда бывают задержки.
- **Сценарий 3: Резервный канал связи на случай аварии основного.** Требования: независимость от основного канала, может быть меньшая скорость, низкая стоимость простоя. Выбор: **3G/4G/5G модем или спутниковый канал.** Аналогия: В машине у вас есть запасное колесо. Оно не такое удобное, как основное, и на нём не стоит ездить постоянно, но в критической ситуации оно спасёт.
- **Критерии выбора в виде вопросов:**

1. **Сколько данных и с какой скоростью нужно передавать? (Пропускная способность, Мбит/с).**
2. **Насколько критичны задержки? (Пинг, мс).** Для голоса и видео задержки критичны, для почты – нет.
3. **Требуется ли гарантированная полоса? (CIR).** Нужно ли, чтобы скорость никогда не падала ниже определённого порога?
4. **Насколько важна безопасность?** Нужно ли шифрование трафика?
5. **Какова география подключений?** Одна точка или многие?
6. **Каков бюджет?**

5. Сервисы глобальной сети

Базовая идея: Сервисы – это услуги, которые предоставляются пользователям или компаниям поверх базовой транспортной инфраструктуры WAN.

Пояснение смысла: Технологии (как MPLS или оптоволокно) – это «трубы». Сервисы – это то, что по этим трубам течёт и как это организовано для конечного потребителя.

Примеры и аналогии:

- **Доступ в Интернет:** Базовый сервис. Провайдер даёт вам «выход» в мировую сеть.
- **Виртуальная частная сеть (VPN):** Сервис, создающий защищённый «туннель» через общедоступную сеть (например, Интернет). Ваши данные шифруются на входе в туннель и расшифровываются на выходе. Аналогия: Вы отправляете ценный груз через общую почтовую систему, но упаковываете его в сейф, который может открыть только ваш партнёр. Даже если сейф перехватят, они не получат доступа к содержимому.
  - **Remote Access VPN:** Удалённый сотрудник подключается к офисной сети из дома или кафе.
  - **Site-to-Site VPN:** Постоянно соединяет две локальные сети разных офисов через Интернет.
- **Телефония (IP-телефония, VoIP):** Сервис передачи голосовых звонков в виде пакетов данных по IP-сети. Аналогия: Раньше голос шёл по отдельным медным проводам (традиционная телефония). Теперь ваш разговор оцифровывается, разбивается на пакеты и идёт вместе со всеми другими данными по одной «широкой трубе».
- **Видеоконференцсвязь:** Сервис для проведения онлайн-совещаний с видео и звуком.
- **Облачные сервисы (IaaS, PaaS, SaaS):** Современный тренд. Вместо того чтобы покупать серверы и ПО, компания арендует вычислительные мощности, платформы или приложения у провайдера (например, Яндекс.Облако, Amazon AWS). WAN обеспечивает доступ к этим «облакам». Аналогия: Раньше в каждом доме была своя дровяная печь для отопления и готовки (свой сервер). Теперь вы подключились к централизованной системе отопления и газоснабжения (облако) и платите только за то, что потребили.

6. Инфраструктура частных глобальных сетей

Базовая идея: Это физические и логические компоненты, которые организация разворачивает или арендует для создания своей собственной, закрытой от публичного доступа WAN.

Пояснение смысла: Цель – построить безопасную, управляемую и предсказуемую сеть между своими объектами. Часто строится на арендованных каналах у операторов.

Примеры и аналогии:

- **Компоненты инфраструктуры:**
  - **Маршрутизаторы (Cisco, Juniper, Huawei):** Устанавливаются в каждом офисе. Это «шлюзы» в глобальную сеть.
  - **WAN-линки:** Арендованные каналы (выделенные линии, MPLS-каналы).
  - **Модемы/CSU/DSU:** Устройства, которые подключают маршрутизатор к физической линии провайдера (например, к медной паре или оптике).
  - **Фаерволы и системы шифрования:** Обеспечивают безопасность на границах сети.

- **Центр управления сетью (NOC):** Специальная комната или служба, где на мониторах отображается состояние всех каналов и устройств, и где инженеры следят за работой сети.
- **Аналогия: Частная железная дорога корпорации.** У компании есть заводы в нескольких городах. Она арендует у государства землю (право прохода) и рельсы (физическую среду), но ставит свои поезда (данные), свои станции (маршрутизаторы), нанимает своих машинистов и диспетчеров (администраторов), строит свои склады и логистические центры (серверы). Движение по этой дороге закрыто для посторонних.

7. Инфраструктура общедоступной глобальной сети (Интернет)

Базовая идея: Это гигантская, децентрализованная, многоуровневая сеть сетей, построенная на сотрудничестве тысяч независимых провайдеров.

Пояснение смысла: Никто не владеет Интернетом в целом. Он работает благодаря тому, что провайдеры разных уровней договариваются между собой о соединении своих сетей и обмене трафиком.

Примеры и аналогии:

- **Уровни провайдеров:**
  - **Провайдеры первого уровня (Tier-1):** «Владельцы магистралей». У них есть глобальная опорная сеть, покрывающая целые континенты. Они обмениваются трафиком друг с другом бесплатно (пиринг). Примеры: Lumen (CenturyLink), Deutsche Telekom, AT&T. Аналогия: Владельцы трансокеанских морских маршрутов или межконтинентальных авиационных коридоров.
  - **Провайдеры второго и третьего уровня (Tier-2, Tier-3):** Региональные и местные провайдеры. У них нет глобальной сети, поэтому они **покупают** транзит трафика у Tier-1 или у других Tier-2, чтобы их клиенты могли связаться с любой точкой Интернета. Аналогия: Региональные автобусные компании. Они возят пассажиров внутри области, но чтобы отправить человека в другую страну, они покупают билет у федеральной железной дороги или авиакомпании (Tier-1).
- **Точки обмена трафиком (IXP – Internet Exchange Point):** Физические места, где провайдеры напрямую соединяют свои сети для обмена трафиком местных пользователей, минуя дорогой транзит через Tier-1. Аналогия: Большой центральный автовокзал в столице. Автобусные компании из разных регионов приезжают туда, и пассажиры могут пересесть с автобуса одной компании на автобус другой, не ехая для этого в какой-то далёкий транспортный хаб.
- **Система доменных имён (DNS):** «Телефонная книга Интернета». Преобразует удобные для человека имена (www.yandex.ru) в машинные IP-адреса (77.88.55.60). Это критически важный сервис инфраструктуры. Аналогия: Вы не помните номер телефона такси, но помните название службы «Яндекс.Такси». Вы открываете приложение-справочник (DNS), находите название, и оно даёт вам номер для связи.

8. Выбор сервисов глобальной сети

Базовая идея: Выбор сервисов делается на основе анализа **бизнес-задач, требований пользователей и экономической эффективности**, а не только технических возможностей.

Пояснение смысла: Сначала нужно понять, *что* нужно делать (задачи), а потом подбирать *как* это сделать (сервисы).

Примеры и аналогии:

- **Задача 1: Обеспечить удалённую работу 50 менеджерам по продажам.**
  - **Потребности:** Безопасный доступ к внутренней CRM-системе и почте из любого места.
  - **Выбор сервиса: Remote Access VPN + двухфакторная аутентификация.** Каждый менеджер устанавливает на ноутбук VPN-клиент, подключается к офису через зашифрованный канал и работает, как будто сидит в офисе.
- **Задача 2: Наладить ежедневный обмен большими объёмами данных (архивами видеозаписей) между двумя студиями.**
  - **Потребности:** Высокая скорость, гарантированная полоса пропускания, прогнозируемое время передачи.
  - **Выбор сервиса: Выделенный канал (Ethernet) между студиями или аренда защищённого FTP-сервера в облаке с быстрым каналом.** Первое решение – для прямого и полного контроля, второе – более гибкое.
- **Задача 3: Заменить устаревшую офисную АТС и сократить расходы на междугороднюю связь.**
  - **Потребности:** Внутренняя телефония, звонки между филиалами, выход на городские/мобильные номера.
  - **Выбор сервиса: Внедрение IP-АТС (например, Asterisk) с использованием SIP-транка от оператора.** Все звонки между филиалами становятся бесплатными (идут по корпоративной WAN), а исходящие звонки тарифицируются по более низким, оптовым тарифам.
- **Критерии выбора сервисов:**

1. **Функциональность:** Что именно должен уметь делать сервис?
2. **Безопасность:** Соответствует ли он политикам информационной безопасности компании?
3. **Масштабируемость:** Можно ли его легко расширить при росте компании?
4. **Стоимость владения (TCO):** Включает не только абонплату, но и затраты на оборудование, администрирование, обучение.
5. **Поддержка и SLA (Service Level Agreement):** Есть ли у провайдера гарантии на время устранения аварий и доступность сервиса (например, 99.9% времени)?

Заключение

Подключение к глобальной сети – это не просто «провести интернет». Это комплексное решение, которое начинается с понимания целей (зачем?), продолжается анализом

технологий и принципов (как это работает?) и заканчивается осознанным выбором инфраструктуры и сервисов (что и у кого арендовать?). Современный специалист по сетевым технологиям должен видеть картину целиком: от физического кабеля до бизнес-требований пользователя. Умение проектировать, подключать и обслуживать WAN-сегменты является одной из ключевых компетенций в области системного и сетевого администрирования, поскольку именно глобальные сети делают локальные ресурсы доступными миру, а мировые ресурсы – доступными локально.

Контрольные вопросы к ТЕМЕ-16

1. Что такое глобальная сеть (WAN) и каково её ключевое отличие от локальной сети (LAN)?
2. Назовите одну из основных целей создания глобальных сетей для бизнеса.
3. Какой принцип коммутации лежит в основе работы современного Интернета и большинства WAN?
4. Что такое пакет в сетях с коммутацией пакетов и для чего он нужен?
5. Какое ключевое устройство находится на границе сетей и отвечает за определение пути для пакетов в WAN?
6. Что такое протокол в контексте компьютерных сетей? Приведите пример базового протокола.
7. Назовите три основных фактора, влияющих на выбор технологии глобальной сети.
8. Что такое VPN и какую основную проблему он решает?
9. Что является основным компонентом инфраструктуры частной глобальной сети, устанавливаемым в каждом офисе компании для подключения к WAN?
10. Кто такие провайдеры первого уровня (Tier-1) в структуре общедоступной сети Интернет?
11. Какую функцию выполняет система DNS в интернете?
12. Что такое SLA в контексте выбора сервиса глобальной сети?

Практические задания к ТЕМЕ-16

Задание 1. Анализ требований для выбора технологии.

Ситуация: Сеть филиала банка в Рязани, где работает 15 сотрудников, необходимо подключить к головному офису в Москве. Требуется ежедневно передавать финансовые отчёты общим объёмом до 2 ГБ, а также обеспечить стабильную голосовую связь через IP-телефонию между руководителями. Бюджет ограничен.
Вопрос: Какая из рассмотренных технологий глобальной сети (выделенная линия, MPLS/VPN, широкополосный доступ с VPN) будет наименее подходящей для данного сценария и почему? Обоснуйте свой ответ, опираясь на критерии выбора из теоретического материала.

Задание 2. Выбор инфраструктуры для расширения бизнеса.

Ситуация: Российская компания по разработке ПО, имеющая офис в Новосибирске,

открывает небольшой филиал в Казани (5 сотрудников). Основная задача филиала — доступ к внутреннему порталу, системе управления задачами (Jira) и участие в видеоконференциях.
Вопрос: Предложите тип инфраструктуры (частная WAN или использование общедоступной сети с определённым сервисом), который будет наиболее экономически целесообразен на начальном этапе. Объясните свой выбор, сравнив ключевые характеристики этих подходов.

Задание 3. Диагностика проблемы на основе принципов работы сети.

Ситуация: Сотрудник в удалённом офисе в Твери жалуется, что при попытке открыть корпоративный сайт, расположенный на сервере в Санкт-Петербурге, страница иногда загружается очень медленно или не загружается вовсе, хотя файлы с локального сервера в Твери копируются быстро.
Вопрос: Опираясь на принцип коммутации пакетов и роль маршрутизаторов, предложите наиболее вероятную причину проблемы не в локальной сети Твери, а именно в глобальном сегменте. Какой компонент глобальной сети может быть «узким местом»?

Задание 4. Подбор сервиса под конкретную бизнес-задачу.

Ситуация: Строительная компания из Краснодара выиграла тендер на возведение объекта в Сочи. На площадке будет работать инженер-сметчик с ноутбуком, которому требуется несколько раз в день загружать в центральную систему обновлённые чертежи и сметы (файлы большого объёма).
Вопрос: Какой конкретный сервис глобальной сети, рассмотренный в материале, логично предложить для решения этой задачи, учитывая мобильность сотрудника и необходимость безопасной передачи данных? Дайте название сервиса и кратко поясните, как он будет применяться в этой ситуации.

Задание 5. Построение аналогии для объяснения клиенту.

Ситуация: Вы, как сетевой администратор, должны объяснить руководителю отдела, не разбирающемуся в технических деталях, почему для нового офиса в Екатеринбурге вы рекомендуете не просто «интернет», а услугу «MPLS-VPN» от оператора.
Вопрос: Используя аналогию из теоретического материала (про дороги, магистрали или железнодорожные пути), составьте краткое и понятное объяснение (2-3 предложения), в чём ключевое преимущество MPLS-VPN перед обычным интернет-каналом для связи с московским офисом.

ТЕМА-17. Соединение «точка-точка»

Введение

В мире компьютерных сетей соединения между устройствами можно организовать по-разному. Одним из фундаментальных и исторически важных способов является прямое соединение двух устройств, так называемое соединение «точка-точка» (Point-to-Point, P2P). Представьте себе обычный телефонный разговор между двумя людьми: существует прямая, выделенная линия связи (пусть даже виртуальная), по которой информация передается от одного абонента к другому и наоборот. Никто третий в этот момент не может подключиться к этому разговору. Именно по такому принципу и работают многие магистральные и удаленные (WAN) соединения, например, связь между маршрутизатором провайдера и вашим домашним или офисным маршрутизатором.

Данная тема посвящена изучению технологий, которые позволяют организовать такое прямое соединение. Мы разберем, как данные передаются по последовательным каналам, познакомимся с принципами работы ключевых протоколов, таких как HDLC и PPP, и научимся понимать, как настраивать и проверять подобные соединения.

1. Обзор последовательного соединения «точка-точка». Связь по последовательному каналу.

Базовая идея: Последовательное соединение «точка-точка» — это физический или логический канал связи, который соединяет исключительно два устройства. Данные передаются бит за битом (последовательно) по одному физическому пути.

Пояснение смысла и аналогии:
Представьте, что вам нужно передать книгу из пункта А в пункт Б.

- **Параллельная передача (для контраста):** Вы могли бы разорвать книгу на листы, раздать каждую страницу отдельному курьеру, и все они побегут по разным дорожкам (параллельным путям) одновременно. Это быстро, но требует много ресурсов (курьеров, дорожек), и сложно гарантировать, что все страницы прибудут одновременно и в правильном порядке, особенно если дорожки разного качества.
- **Последовательная передача (наш случай):** Вы поручаете книгу одному курьеру. Он будет перелистывать страницы и передавать их по одной через единственную узкую тропинку. Страницы идут строго одна за другой. Это может быть немного медленнее в теории, но требует лишь одного канала, надежно и предсказуемо. Порядок страниц никогда не собьется.

Именно так долгое время работали выделенные линии связи (например, арендованная медная пара или оптоволоконный кабель между двумя зданиями). На физическом уровне это часто был **последовательный канал**, такой как RS-232 (COM-порт) или V.35.

Ключевая формулировка: В сетевых технологиях под **последовательным соединением** часто подразумевают не только физическую передачу битов, но и весь стек технологий, включая канальный уровень, который управляет этим соединением. На физическом уровне — это просто провод (или его аналог), по которому «текут» биты. А вот чтобы эти биты превратились в осмысленные сетевые пакеты (например, IP-пакеты), нужен специальный протокол канального уровня, который «упаковывает» данные.

2. Инкапсуляция HDLC (High-Level Data Link Control)

Базовая идея: HDLC — это один из простейших и самых старых протоколов канального уровня для соединений «точка-точка». Его основная задача — организовать сырой поток битов в четко оформленные **кадры** (frames), которые имеют начало, конец, адрес, управляющую информацию и проверку на ошибки.

Пояснение смысла и аналогии:
Вернемся к нашему курьеру с книгой. Представьте, что вместе с книгой вам нужно отправить несколько других документов, а также быть уверенным, что все дошло целым.

- **Поток битов без HDLC:** Курьер просто сбрасывает в пункт Б стопку несброшюрованных листов из разных книг. Получатель не понимает, где начинается один документ, а где кончается другой.
- **Инкапсуляция HDLC:** Перед отправкой каждый документ вы кладете в специальный **конверт** (кадр HDLC). У этого конверта есть:

1. **Флаговое поле (Flag, 1 байт: 01111110)** — Яркая цветная полоска на краю конверта. Она говорит: «Внимание! Здесь начинается новый документ!». Такая же полоска ставится в конце, чтобы показать, где документ заканчивается.
2. **Адресное поле (Address, 1-2 байта)** — Поле «Кому». В соединении «точка-точка» оно часто имеет стандартное значение, так как получатель всегда один.
3. **Управляющее поле (Control, 1-2 байта)** — Служебная пометка. Например, «это данные», «это команда управления соединением», «это подтверждение о получении предыдущего конверта».
4. **Поле данных (Data, переменная длина)** — Сам документ (сетевой пакет, например, IP), который мы хотим передать.
5. **Поле контрольной суммы (FCS, 2-4 байта)** — Специальный цифровой код, вычисленный на основе содержимого конверта (адреса, управления, данных). Получатель вычисляет такую же сумму. Если цифры не совпали — значит, конверт был поврежден в пути (были помехи на линии).
6. **Замыкающее флаговое поле (Flag)** — Та же яркая полоска, означающая конец конверта.

Важный нюанс: Поскольку флаг (01111110) — это специальная последовательность, она не должна случайно появиться внутри данных. HDLC использует технику **битового стаффинга (bit stuffing)**: если в данных встречается пять единиц подряд, передатчик автоматически вставляет ноль. Приемник, увидев пять единиц и ноль, этот ноль удаляет. Это гарантирует, что флаг уникален.

Пример: Данные 01111100111110 будут переданы как 011111**0**0011111**0**0 (вставленные биты выделены). Приемник уберет нули после пяти единиц.

Недостаток HDLC: Это проприетарный протокол. Разные производители (Cisco, Juniper и др.) реализовали его по-своему, добавили свои поля. Поэтому оборудование Cisco «не поймет» кадр HDLC от оборудования другого вендора. Для создания универсального соединения потребовался открытый стандарт — PPP.

3. Принцип работы протокола PPP (Point-to-Point Protocol)

Базовая идея: PPP — это более совершенный, гибкий и открытый протокол канального уровня, пришедший на смену HDLC. Он решает не только задачу инкапсуляции, но и обеспечивает установление связи, проверку качества линии, согласование параметров и поддержку多种сетевых протоколов.

Пояснение смысла и аналогии:
PPP — это не просто конверт, а целый **процесс отправки заказного письма с дополнительными услугами**.

1. **Установление физического соединения:** Сначала нужно подключить кабель (поднять трубку телефона). На физическом уровне появляется несущая (carrier detect).
2. **Установление канала (Link Establishment):** Далее PPP отправляет специальные **кадры управления (LCP)**, чтобы «поздороваться» и договориться о базовых параметрах соединения. Это аналог того, как вы, взяв трубку, говорите «Алло?» и слышите в ответ «Да, я на линии, слышу вас». LCP проверяет, работает ли линия, договаривается о размере кадра, необходимости аутентификации.
3. **Аутентификация (опционально, но часто используется):** По вашему желанию, почтовый сервис (PPP) может потребовать паспорт (логин и пароль), чтобы убедиться, что отправитель — это действительно вы. Делается это с помощью специальных протоколов (PAP, CHAP) в рамках всё тех же LCP-кадров.
4. **Согласование сетевого уровня (Network Layer Negotiation):** После проверки линии и личности, нужно понять, *какие именно документы* вы будете отправлять. Будет ли это IP-пакет, пакет IPX или что-то еще? Для этого PPP использует **кадры NCP (Network Control Protocol)**. Для каждого сетевого протокола есть свой NCP. Например, **IPCP (Internet Protocol Control Protocol)** «договаривается» о том, что мы будем передавать IP-трафик, и может даже автоматически назначить IP-адреса обоим концам соединения.

5. **Фаза передачи данных:** Только после успешного прохождения всех предыдущих этапов начинается обмен «полезными» данными — IP-пакетами, которые инкапсулируются в кадры PPP.
6. **Завершение соединения:** Когда связь больше не нужна, отправляются LCP-кадры для аккуратного разрыва соединения (как вежливое «До свидания» перед тем, как повесить трубку).

Структура кадра PPP проще, чем у HDLC:

- **Флаги:** Как и в HDLC, обозначают начало и конец кадра (значение 0x7E).
- **Адрес:** Всегда имеет стандартное значение 0xFF (широковещательный адрес), так как получатель на линии всего один. Это поле часто называют «остатком от HDLC».
- **Управление:** Также стандартное значение 0x03.
- **Протокол (2 байта): Самое важное поле PPP!** Оно указывает, что находится в поле данных. Например:
  - `0xC021` — значит, в данных находится **кадр LCP** (управление каналом).
  - `0xC023` — значит, в данных находится **кадр аутентификации PAP**.
  - `0x8021` — значит, в данных находится **кадр NCP для IP (IPCP)**.
  - `0x0021` — значит, в данных находится обычный **IP-пакет** (наши полезные данные).
- **Данные:** Полезная нагрузка (LCP-пакет, NCP-пакет или IP-пакет).
- **Контрольная сумма (FCS):** Для обнаружения ошибок.

4. Преимущества протокола PPP

1. **Открытый стандарт и совместимость:** В отличие от HDLC, PPP является стандартом IETF (RFC 1661, 1662). Это гарантирует, что маршрутизатор Cisco сможет установить PPP-соединение с маршрутизатором Huawei, MikroTik или любым другим.
2. **Многопротокольность:** PPP может переносить трафик разных сетевых протоколов (IP, IPX, AppleTalk) одновременно, просто указывая тип в поле «Протокол».
3. **Встроенная аутентификация:** Поддержка PAP и CHAP повышает безопасность соединения, предотвращая несанкционированный доступ к каналу.
4. **Согласование параметров и контроль качества:** Механизмы LCP позволяют автоматически согласовать размер кадра, обнаружить петлю на линии (loopback) и проверить качество связи.
5. **Возможность назначения адресов:** С помощью NCP (например, IPCP) PPP может автоматически назначить IP-адреса конечным точкам, что критически важно для dial-up-подключений (когда пользователь звонит по модему).
6. **Гибкость:** PPP может работать поверх самых разных физических сред: последовательный кабель (RS-232), телефонная линия (модем), сотовая связь, оптоволокно (SONET/SDH), Ethernet (PPPoE — что используется в большинстве домашних подключений к интернету).

5. LCP и NCP – «мозг» протокола PPP

LCP (Link Control Protocol) – Протокол управления каналом.

- **Роль:** «Строитель и смотритель моста». Отвечает за создание, поддержку и разрыв канального соединения.
- **Что делает LCP:**
  - **Проверка связи (Echo-Request / Echo-Reply):** Посылает «эхо-запросы», чтобы убедиться, что удаленная сторона жива и линия работает.
  - **Согласование параметров:** Обсуждает с удаленной стороной максимальный размер кадра (MRU), необходимость использования сжатия заголовков или специальных управляющих символов.
  - **Обнаружение петли:** Помогает выявить, не подключен ли кабель обратно на то же устройство (ошибка при настройке).
  - **Выбор метода аутентификации:** Договаривается, будет ли использоваться PAP, CHAP или аутентификация не требуется.

NCP (Network Control Protocol) – Протокол управления сетевым уровнем.

- **Роль:** «Таможенный инспектор и логист». Работает **после того, как LCP построил мост**. Отвечает за «пропуск» конкретных типов сетевого трафика через этот мост.
- **Что делает NCP (на примере самого важного — IPCP):**
  - **Активирует протокол:** Сообщает удаленной стороне: «Теперь я буду отправлять IP-пакеты».
  - **Назначает IP-адреса:** Может запросить у удаленной стороны IP-адрес для себя (как это было в dial-up) или подтвердить статический адрес.
  - **Согласует адреса DNS-серверов:** Может передать информацию о DNS.
  - Для других протоколов существуют свои NCP: IPXCP, ATCP и т.д.

Аналогия: Представьте, что вы открываете представительство компании в другой стране (устанавливаете WAN-соединение).

- **LCP** — это юристы и строители. Они арендуют офисное здание (устанавливают физическую и канальную связь), проводят в него коммуникации, устанавливают систему безопасности (аутентификация).
- **NCP** — это менеджеры по конкретным направлениям. Только после сдачи офиса в эксплуатацию приходит, например, **менеджер по IP-перевозкам (IPCP)**. Он нанимает курьеров (настраивает маршрутизацию IP), получает для них местные пропуска (IP-адреса) и начинает организовывать пересылку посылок (IP-пакетов).

6. Сеансы PPP

Базовая идея: Сеанс PPP — это полный цикл жизни соединения, от установления до разрыва. Он проходит через четко определенные фазы, что делает процесс надежным и предсказуемым.

Этапы сеанса PPP (наглядная последовательность):

1. **Фаза неактивности (мертвая фаза):** Соединение разорвано или физической связи нет.
2. **Фаза установления канала (LCP Negotiation):** Появилась физическая несущая. Устройства обмениваются LCP-пакетами Configure-Request, Configure-Ack, Configure-Nak, Configure-Reject, чтобы договориться о базовых параметрах. **Если согласование не удалось, соединение сбрасывается.**
3. **Фаза аутентификации (Authentication):** Если она была согласована на предыдущем этапе, теперь происходит проверка подлинности с помощью PAP или CHAP. **Если аутентификация не пройдена, соединение разрывается.**
4. **Фаза сетевого уровня (NCP Negotiation):** LCP-соединение установлено и аутентифицировано. Теперь для каждого необходимого сетевого протокола запускается свой NCP. Чаще всего это IPCP. После успешного завершения этой фазы начинается передача пользовательских данных.
5. **Фаза открытого соединения (Open):** Активная фаза передачи данных. В это время LCP продолжает работать в фоновом режиме, периодически проверяя качество линии (Echo).
6. **Фаза завершения (Termination):** Когда связь больше не нужна, отправляются LCP-пакеты Terminate-Request и Terminate-Ack для корректного разрыва канала. Устройства возвращаются в фазу неактивности.

Этот строгий порядок гарантирует, что передача данных начнется только тогда, когда канал полностью подготовлен и проверен.

7. Настройка протокола PPP (практический аспект)

Базовая идея: Настройка PPP на сетевом оборудовании (например, маршрутизаторе Cisco) включает в себя выбор PPP в качестве протокола инкапсуляции на последовательном интерфейсе и, при необходимости, настройку параметров аутентификации.

Пошаговая логика настройки (обобщенный вид):

1. **Переход в режим настройки интерфейса:** Выбирается физический последовательный интерфейс (например, Serial 0/0/0), который подключен к линии «точка-точка».
2. **Назначение IP-адреса:** Интерфейсу назначается IP-адрес из подсети, общей для двух устройств на концах канала.
3. **Установка инкапсуляции PPP:** Командой encapsulation ppp указывается, что на данном интерфейсе будет использоваться протокол PPP, а не HDLC или что-то еще.
4. **(Опционально) Настройка аутентификации:**

- Создается список имен пользователей и паролей (username … password …).
- На интерфейсе выбирается метод аутентификации: ppp authentication pap или ppp authentication chap.
- При использовании PAP, возможно, потребуется отправить свои учетные данные удаленной стороне командой ppp pap sent-username ….

Пример аналогии для настройки: Представьте, что вы настраиваете двустороннюю рацию.

1. Вы берете в руки конкретную рацию (заходите в режим настройки интерфейса Serial 0/0/0).
2. Присваиваете ей позывной (назначаете IP-адрес).
3. **Ключевой шаг:** Выбираете открытый, понятный всем протокол общения, например, русский язык (encapsulation ppp), а не секретный жаргон (HDLC).
4. Если нужно, устанавливаете кодовое слово для идентификации своих (настраиваете CHAP).

8. Аутентификация PPP: PAP и CHAP

Базовая идея: Это механизмы проверки подлинности удаленной стороны перед установлением соединения. Они предотвращают несанкционированный доступ к каналу связи.

PAP (Password Authentication Protocol)

- **Принцип:** Очень простой, но небезопасный. Пароль передается **в открытом виде (clear-text)**.
- **Процесс:** Клиент посылает запрос на соединение с именем пользователя и паролем. Сервер проверяет их по своей базе. Если данные верны — соединение разрешается.
- **Аналогия:** Вы подходите к секретному объекту и называете охраннику свою фамилию и кодовое слово вслух. Любой, кто стоит рядом, может их подслушать и использовать позже.
- **Недостаток:** Уязвим к атаке «прослушивания» (sniffing).

CHAP (Challenge-Handshake Authentication Protocol)

- **Принцип:** Более безопасный. Пароль никогда не передается по сети. Используется механизм «рукопожатия с запросом-ответом» и хеширования (обычно MD5).
- **Процесс:**

1. После установления LCP-соединения сервер (тот, кто проверяет) отправляет клиенту **случайное число (challenge, «вызов»)**.
2. Клиент берет это случайное число и свой секретный пароль, пропускает их через хеш-функцию (MD5) и получает некий результат (хеш).
3. Клиент отправляет серверу этот хеш и свое имя.
4. Сервер, зная имя клиента, находит в своей базе его пароль. Он проделывает ту же операцию: берет отправленное им же случайное число и пароль клиента из своей базы, вычисляет хеш.
5. Если вычисленный сервером хеш совпадает с хешом, полученным от клиента, значит, клиент знает правильный пароль. Аутентификация успешна.

- **Аналогия:** Охранник (сервер) дает вам (клиенту) случайный лист бумаги с числом (challenge). У вас и у охранника есть один и тот же секретный код (пароль). Вы записываете это число на своем секретном коде, получаете оттиск (хеш) и отдаете оттиск охраннику. Охранник делает то же самое со своим экземпляром кода. Если оттиски совпали — вы свой человек. При этом сам секретный код (шифр) никогда не показывался.
- **Преимущества CHAP:** Безопаснее, т.к. нет передачи пароля. Кроме того, CHAP может выполнять **периодическую повторную аутентификацию** в процессе сеанса, что повышает безопасность.

9. Отладка соединений WAN. Отладка PPP.

Базовая идея: После настройки соединение может не подняться. Для диагностики проблем используются специальные команды, которые показывают состояние интерфейса, протоколов и статистику.

Ключевые методы отладки:

1. **Проверка состояния интерфейса:** show interfaces serial X/X/X
   - **Что смотрим:** Статус интерфейса Serial... is up означает, что физический уровень работает (есть несущая). Статус протокола line protocol is up означает, что канальный уровень (PPP) работает. Если протокол down, а физика up — проблема на уровне PPP (неверная настройка, ошибка аутентификации).
   - **Аналогия:** Проверка, включен ли телефон (физика) и есть ли гудок в трубке (протокол).
2. **Просмотр инкапсуляции:** show interfaces serial X/X/X (строка Encapsulation PPP). Убедиться, что на обоих концах стоит PPP.
3. **Отладка PPP-переговоров (в реальном времени):** debug ppp negotiation
   - **Что показывает:** В режиме реального времени выводит все этапы сеанса PPP: обмен LCP-пакетами (Conf-Req, Conf-Ack, Conf-Nak), процесс аутентификации (PAP/CHAP), обмен NCP (IPCP). Это основной инструмент для поиска неисправностей.
   - **Пример вывода проблемы:** Вы можете увидеть, что удаленная сторона отвергает (Conf-Nak) ваш запрос на аутентификацию CHAP, или что IPCP не согласуется из-за конфликта IP-адресов.
   - **Важно:** Эта команда создает высокую нагрузку на устройство, ее используют временно и отключают (undebug all).
4. **Отладка аутентификации:** debug ppp authentication
   - **Что показывает:** Детали только по процессу аутентификации PAP или CHAP. Показывает успешные и неуспешные попытки.
   - **Пример:** Можно увидеть сообщение CHAP: O FAILURE, означающее сбой аутентификации CHAP (неверный пароль или имя пользователя).
5. **Проверка журналов (logs):** Сообщения об ошибках PPP (например, %PPP-4-AUTHFAIL) обычно пишутся в системный лог устройства, который можно просмотреть командой show log.

Типичные проблемы и их признаки:

- **Несоответствие инкапсуляции:** На одном конце PPP, на другом HDLC. Статус протокола будет down, в debug видно, что LCP-пакеты не принимаются или отвергаются.
- **Ошибка аутентификации:** Неверный пароль или имя пользователя в конфигурации. Debug покажет FAILURE. Интерфейс может циклически переходить из up в down.
- **Несогласованные параметры LCP:** Например, разные значения MRU. Debug покажет Conf-Nak.
- **Нет IP-адреса или конфликт адресов:** IPCP не может завершить переговоры. Debug ppp negotiation покажет, что IPCP «завис» на фазе запроса/подтверждения.

Заключение

Соединение «точка-точка» и протокол PPP являются краеугольным камнем в построении территориальных сетей (WAN). Понимание принципов последовательной передачи, этапов инкапсуляции, строгой последовательности сеанса PPP (LCP -> Authentication -> NCP -> Data) и методов аутентификации является обязательным для любого системного и сетевого администратора. Несмотря на то, что сегодня многие выделенные линии эмулируются поверх Ethernet (используются технологии вроде Metro Ethernet), логика протоколов канального уровня, такая как установление соединения, аутентификация и инкапсуляция, остается актуальной и находит свое воплощение в более современных реализациях, таких как PPPoE, который обеспечивает ваше домашнее подключение к интернету. Умение настраивать и, что более важно, грамотно диагностировать проблемы PPP-соединений — ключевой практический навык в данной области.

Контрольные вопросы к ТЕМЕ-17

1. Что такое соединение «точка-точка» на физическом уровне?
2. Чем принципиально отличается последовательная передача данных от параллельной (на основе аналогии из материала)?
3. Какова основная задача протокола канального уровня HDLC?
4. Для чего в кадре HDLC (и PPP) используются флаговые поля (Flag) и поле контрольной суммы (FCS)?
5. Назовите главный недостаток протокола HDLC, который решил протокол PPP.
6. Перечислите основные преимущества протокола PPP перед базовым HDLC (достаточно трёх).
7. Что обозначает поле «Протокол» в заголовке кадра PPP и почему оно является ключевым?
8. Какую функцию выполняет LCP (Link Control Protocol) в рамках сеанса PPP?
9. Какую функцию выполняет NCP (Network Control Protocol), например IPCP, в рамках сеанса PPP?

10. Опишите основное различие между методами аутентификации PAP и CHAP в протоколе PPP.
11. Если при проверке команда показывает, что состояние интерфейса «Serial … is up», а «line protocol is down», на каком уровне (физическом или канальном) наиболее вероятна проблема?
12. Какая команда отладки является основной для пошагового наблюдения за процессом установления PPP-соединения (переговорами LCP, аутентификацией, согласованием NCP)?

Практические задания к ТЕМЕ-17

Задание 1. Диагностика нерабочего канала между офисами

Сетевой администратор в Перми настроил выделенную линию связи между главным офисом и складом. Линия арендована у местного провайдера «УралСвязь». Физическая связь есть (индикаторы на портах горят), но IP-связности между маршрутизаторами нет. При проверке на маршрутизаторе главного офиса команда show interfaces serial 0/0/0 показывает: Serial0/0/0 is up, line protocol is down. Инкапсуляция на интерфейсе настроена как ppp.

- **Вопрос:** На каком именно этапе установления соединения возник сбой? Перечислите все возможные причины (не менее трёх), связанные с работой протокола PPP, которые могли привести к такой ситуации, основываясь на изученной последовательности сеанса PPP.

Задание 2. Выбор технологии для нового филиала

Компания «СеверСталь» в Череповце открывает небольшой филиал в Вологде. Для связи с главным офисом необходим надёжный и безопасный выделенный канал. Поставщик услуг предлагает два варианта настройки инкапсуляции на своих маршрутизаторах Cisco: HDLC или PPP. Администратор компании хочет использовать оборудование разных вендоров (например, на стороне филиала поставить более дешёвый маршрутизатор MikroTik).

- **Вопрос:** Какой протокол инкапсуляции должен выбрать администратор и почему? Приведите два ключевых аргумента в пользу вашего выбора, исходя из фундаментальных различий между протоколами.

Задание 3. Анализ проблемы с удалённым сотрудником

Удалённый сотрудник в Твери подключается к корпоративной сети компании «РосТех» в Москве через выделенную линию с использованием PPP. Внезапно его соединение начало постоянно обрываться и устанавливаться заново. Коллега-администратор, анализируя логи центрального маршрутизатора, обнаружил повторяющуюся ошибку: %PPP-4-AUTHFAIL: Interface Serial0/1/0, CHAP authentication failed.

- **Вопрос:** Объясните, что означает эта ошибка и на каком этапе сеанса PPP она происходит. Предложите две наиболее вероятные практические причины, которые могли привести к сбою аутентификации CHAP в данной ситуации.

Задание 4. Расшифровка процесса установления связи

Молодой администратор в Казани, настраивая PPP-соединение, запустил команду отладки debug ppp negotiation и получил следующий упрощённый лог:

```
1. LCP: State is Listen
2. LCP: CONFREQ received
3. LCP: CONFACK sent
4. CHAP: O CHALLENGE id 15
5. CHAP: I RESPONSE id 15
6. CHAP: O SUCCESS id 15
7. IPCP: CONFREQ received
8. IPCP: CONFACK sent
```

- **Вопрос:** Восстановите по логу последовательность этапов установления PPP-сеанса. Определите, на каком устройстве (локальном или удалённом) была запущена отладка? Обоснуйте свой ответ, используя обозначения из лога (O = Outgoing, I = Incoming).

Задание 5. Оптимизация безопасности соединения

Администратор сети небольшого банка в Ярославле обнаружил, что соединение PPP между отделением и операционным центром настроено с использованием аутентификации PAP. Он понимает, что это несёт риски для безопасности финансовых данных.

- **Вопрос:** Чем именно протокол PAP небезопасен в данном контексте? Какой механизм работы CHAP делает его более предпочтительным для защиты от перехвата учетных данных? Ответ дайте, используя аналогию с «охранником и секретным кодом», приведённую в материале.

ТЕМА-18. Решения широкополосного доступа

Введение в контекст - цифровая трансформация труда

Современный мир переживает революцию в организации труда, которая напрямую связана с развитием компьютерных сетей и технологий широкополосного доступа. Представьте себе офис, который может находиться где угодно: на кухне домашней квартиры, в загородном доме, в коворкинге в другом городе или даже в кафе на берегу моря. Это стало возможным благодаря технологиям, которые обеспечивают высокоскоростной, надежный и постоянный доступ к корпоративным ресурсам через глобальную сеть Интернет. Изучаемая тема лежит на стыке организационных изменений в бизнесе и технических решений, которые эти изменения делают возможными. Мы будем рассматривать не просто технологии подключения, а целостную экосистему, где бизнес-требования определяют выбор технических средств, а возможности технологий открывают новые формы организации бизнес-процессов.

Глава 1. Удалённая работа как новый стандарт

1.1. Сущность и определение удалённой работы

Удалённая работа (Remote Work) — это форма организации трудовой деятельности, при которой сотрудник выполняет свои профессиональные обязанности вне стационарного офиса работодателя, используя информационно-коммуникационные технологии для связи и доступа к необходимым ресурсам.

Пояснение смысла: Ключевая идея здесь — **разделение физического места нахождения работника и места расположения ресурсов, необходимых для работы**. Аналогия: представьте библиотеку. Раньше, чтобы прочитать книгу, вам нужно было физически прийти в здание библиотеки (офис). Теперь у вас есть электронный читательский билет (учётные данные), и вы можете получить доступ к оцифрованным книгам (рабочим файлам, программам) из любого места, где есть интернет. Сами «книги» хранятся на библиотечном сервере (корпоративном сервере или в облаке), а вы лишь «запрашиваете» их для чтения и работы с ними. Удалённая работа превратила рабочий компьютер из конкретного устройства в офисе в «виртуальный портал», доступ к которому можно получить с различных устройств из любой точки мира.

1.2. Преимущества удалённой работы

Преимущества рассматриваются с разных точек зрения: для сотрудника, для работодателя и для общества в целом.

Для сотрудника:

- **Гибкость графика и автономия.** Сотрудник получает больший контроль над своим рабочим днём. Пример: родитель может начать работу раньше, чтобы освободить время на встречу ребёнка из школы, а затем завершить задачи вечером. Это похоже на переход от жёсткого расписания автобусных рейсов к личному автомобилю — вы сами выбираете маршрут и время поездки, соблюдая общие правила дорожного движения (дедлайны).
- **Экономия времени и ресурсов.** Исключаются ежедневные поездки в офис (коммутирование). Сэкономленные часы и деньги на транспорте превращаются в дополнительный ресурс для отдыха, семьи или профессионального развития.
- **Повышение качества жизни и возможность жить в любом месте.** Специалист из небольшого города может работать на столичную или международную компанию без необходимости переезда. Это стирает географические границы рынка труда.

Для работодателя:

- **Снижение операционных издержек.** Требуется меньше офисных площадей, меньше расходов на аренду, коммунальные услуги, содержание рабочих мест. Аналогия: вместо строительства одного огромного универсама в центре города компания создаёт сеть небольших складов-логистических центров на окраинах, откуда товары (результаты труда) доставляются прямо клиентам, экономя на содержании крупного объекта.
- **Расширение пула талантов.** Поиск сотрудников больше не ограничен территорией одного города. Компания может нанимать лучших специалистов, независимо от их места проживания.
- **Повышение продуктивности и снижение текучки кадров.** Исследования часто показывают рост продуктивности удалённых сотрудников за счёт меньшего количества отвлечений и более комфортных условий. Удовлетворённость режимом работы повышает лояльность сотрудников.

Для общества:

- Снижение транспортной нагрузки и экологической нагрузки от сокращения числа ежедневных поездок.
- Развитие регионов, так как люди могут оставаться жить в родных городах, не теряя в качестве работы и уровне дохода.

1.3. Бизнес-требования для удалённых работников

Это ключевой связующий элемент между организационной теорией и технической реализацией. Бизнес-требования — это набор условий, которые необходимо выполнить, чтобы удалённая работа была эффективной и безопасной.

1. **Надёжный и безопасный доступ к корпоративной сети.** Это базовое требование. Сотрудник должен иметь возможность так же просто и безопасно подключиться к внутренним системам компании (файловым серверам, базам данных, внутренним порталам), как если бы он сидел в офисе. **Пример:** Бухгалтер должен работать с 1С, которая

установлена на сервере в офисе. Техническое решение (например, VPN) создаёт «защищённый туннель» через интернет, который для программы 1С выглядит как обычная сетевая карта, подключённая к офисной локальной сети.

2. **Обеспечение информационной безопасности (ИБ).** Это критическое требование. Работа извне несёт риски утечки данных.

- **Аутентификация и авторизация:** Усиленная проверка личности сотрудника (двухфакторная аутентификация — пароль + код из SMS). **Аналогия:** Недостаточно просто назвать свою фамилию на проходной (пароль). Нужно также предъявить пропуск со своей фотографией (уникальный код).
- **Шифрование данных:** Всё, что передаётся между компьютером сотрудника и офисом, должно быть зашифровано, даже если передаётся через публичные сети. Представьте, что вы отправляете коллеге секретный документ не открыткой (открытый текст), а в сейфе, закрытом на код (шифрование).
- **Защита конечных точек:** Требования к антивирусной защите, регулярным обновлениям, запрету использования непроверенного ПО на рабочих ноутбуках.

3. **Средства коммуникации и collaboration (совместной работы).** Удалённая команда должна иметь инструменты для эффективного общения и совместной работы: корпоративная почта, видеоконференции (Zoom, Teams), мессенджеры (Slack, Telegram для бизнеса), облачные документы (Google Docs, Office 365), системы управления проектами (Jira, Trello).
4. **Техническая поддержка и стандартизация оборудования.** Компания должна определить, какое оборудование (ноутбуки, маршрутизаторы) она предоставляет сотруднику, а какое он может использовать своё (BYOD — Bring Your Own Device). Должна быть организована удалённая техническая поддержка, способная помочь сотруднику решить проблему с подключением или ПО без физического присутствия.
5. **Чёткие организационные правила и KPI (ключевые показатели эффективности).** Управление по результатам становится важнее контроля за временем, проведённым на рабочем месте. Должны быть ясные задачи, сроки и метрики успеха.

Глава 2. Сравнение решений широкополосного доступа

Широкополосный доступ (Broadband Access) — это высокоскоростной доступ в Интернет, обеспечивающий передачу данных на скоростях, значительно превышающих возможности старых технологий (таких как dial-up модем). Минимальная скорость для признания доступа широкополосным варьируется, но обычно это десятки и сотни Мбит/с.

Основная задача удалённого работника — установить надёжный «мост» между своим домом и корпоративной сетью. Этот «мост» и есть широкополосное соединение. Существует три основных типа материалов, из которых можно построить этот мост: **медные/оптические кабели, телефонные линии (медь) и радиоволны.** Выбор зависит от «ландшафта» (доступности технологии), «нагрузки» (необходимой скорости и стабильности) и «бюджета».

2.1. Кабельное подключение (DOCSIS)

Суть технологии: Используется существующая инфраструктура кабельного телевидения (коаксиальный кабель). Данные передаются по тому же кабелю, что и телевизионный сигнал, но на разных частотах.

Аналогия: Представьте широкую многополосную автомобильную дорогу (кабель). Одна или две полосы выделены для автобусов регулярного сообщения — это аналоговые TV-каналы. Остальные полосы отданы под общий поток данных (интернет), который организуется с помощью специальных правил (стандарт DOCSIS). Провайдер устанавливает «развязку» — модем у вас дома, который отделяет интернет-трафик от TV-сигнала.

Преимущества:

- **Высокая скорость (до 1 Гбит/с и более).** Широкий «кабель» позволяет пропустить много «машин» (данных).
- **Относительная простота подключения:** если в дом уже заведён кабель ТВ.
- **Стабильность:** на качество связи меньше влияют погодные условия по сравнению с беспроводными технологиями.

Недостатки:

- **Разделение пропускной способности (Shared Medium).** Все абоненты в вашем доме/районе используют один «кабель» до оборудования провайдера. В часы пик, когда многие смотрят видео или играют, скорость может проседать. **Пример:** Это как водопроводная труба на многоквартирный дом. Если вечером все одновременно открывают краны, напор воды (скорость) у каждого падает.
- **Зависимость от инфраструктуры конкретного провайдера (кабельного оператора).**

2.2. DSL (Digital Subscriber Line — цифровая абонентская линия)

Суть технологии: Использует уже существующую инфраструктуру обычных телефонных линий (медную пару). Ключевая особенность — интернет-сигнал передаётся на частотах, отличных от голосового, что позволяет одновременно говорить по телефону и пользоваться интернетом.

Аналогия: Представьте узкую горную тропинку (телефонная линия). Раньше по ней мог идти только один человек (голосовой звонок). Технология DSL научилась отправлять по этой же тропинке одновременно и пешехода (голос), и велосипедиста (данные на низкой частоте), и даже мотоциклиста (данные на высокой частоте), разделив для них разные «уровни» тропы (частоты). Специальный прибор — сплиттер — играет роль регулировщика на входе в дом, который разделяет поток.

Основные виды xDSL (где x — обозначение типа):

- **ADSL (Asymmetric DSL)** — **асимметричный** доступ. Скорость загрузки (Download) значительно выше скорости отдачи (Upload). Идеально для потребителя, который больше

скачивает (смотрит фильмы, читает). Для удалённой работы часто недостаточен из-за низкой скорости отдачи, необходимой для видеозвонков и загрузки файлов на сервер.

- **VDSL (Very high bitrate DSL)** — более продвинутая версия, обеспечивающая более высокие скорости на коротких расстояниях (до 1-2 км от узла провайдера). Более сбалансирован.
- **SDSL (Symmetric DSL)** — **симметричный** доступ, где скорости приёма и передачи равны. Редкий и дорогой вариант, критически важный для некоторых бизнес-задач (например, для размещения сервера).

Преимущества DSL:

- **Широкая доступность.** Телефонные линии есть почти везде.
- **Выделенная линия.** В отличие от кабельного, ваше соединение не делится с соседями напрямую на последнем участке (от узла до дома). Это как лифт в вашей квартире, а не общий на этаж.

Недостатки:

- **Сильная зависимость качества и скорости от расстояния до узла провайдера (АТС).** Чем дальше, тем хуже сигнал и ниже скорость. «Тропинка» (линия) длиннее — больше потерь.
- **Чувствительность к качеству медной пары.** Старые, окисленные провода резко снижают возможности.
- **Ограниченная максимальная скорость,** особенно по сравнению с оптикой.

2.3. Беспроводные широкополосные сети

Суть технологии: Использование радиоволн различного диапазона для создания подключения «последней мили» (от вышки до дома/устройства).

Основные виды:

1. **Спутниковый интернет.** Связь идёт через спутник на геостационарной орбите.
   - **Плюсы:** Доступность в любой точке, где есть вид на южное небо (для удалённых посёлков, морей).
   - **Минусы: Высокая задержка (ping).** Сигналу нужно пройти ~36 000 км до спутника и обратно. Это делает онлайн-игры и видеоконференции с низкой задержкой практически невозможными. Стоимость оборудования и трафика высока.
   - **Аналогия:** Общение через переводчика, который стоит на другом конце огромного зала. Вы кричите вопрос, он его слышит, переводит, кричит ответ — на всё уходит много времени.
2. **Радиоинтернет (WiMAX, LTE/4G/5G от фиксированных провайдеров).** Провайдер устанавливает базовую станцию (вышку), абонент — направленную антенну на крыше.
   - **Плюсы:** Быстрое развёртывание, хорошая скорость (особенно 4G/5G), мобильность (для LTE-модемов).

- **Минусы:** Зависимость от прямой видимости и погоды (сильный дождь, снег могут ухудшить сигнал). Пропускная способность базовой станции делится между всеми подключёнными абонентами.

3. **Мобильный интернет (3G/4G/5G через USB-модемы или смартфоны).**

- **Плюсы:** Высокая мобильность, повсеместное покрытие в городах, простота настройки.
- **Минусы:** Нестабильность скорости (зависит от нагрузки на вышку и вашего перемещения), частое ограничение трафика, переменная задержка. Для критически важных бизнес-задач обычно используется как резервный канал.

Глава 3. Выбор решений широкополосного доступа для удалённой работы

Критерии выбора вытекают из бизнес-требований, рассмотренных ранее.

1. **Скорость и, что критично, симметрия канала.**

- **Скорость загрузки (Download):** важна для получения файлов, загрузки обновлений, просмотра видеоинструкций.
- **Скорость отдачи (Upload): КРИТИЧНО ВАЖНЫЙ ПАРАМЕТР** для удалённого работника. От неё зависит качество видеозвонков (вы передаёте своё видео), скорость загрузки файлов на корпоративный сервер, отзывчивость удалённого рабочего стола. Минимум для комфортной работы — 10-20 Мбит/с на отдачу. **ADSL с его 1-2 Мбит/с здесь часто проваливается.** Приоритет: **Оптика (PON/GPON) -> Кабель (DOCSIS) -> VDSL -> Радиоканал -> ADSL -> Спутник.**

2. **Стабильность (низкий пинг и джиттер) и надёжность.**

- **Пинг (Ping, задержка):** Время отклика сервера. Измеряется в миллисекундах (мс). Низкий пинг (5-30 мс для проводных технологий) жизненно необходим для видеоконференций (чтобы не было «разрывов» речи), для работы в терминальных сессиях (RDP, Citrix), где каждое ваше движение мыши должно мгновенно отображаться на экране.
- **Джиттер (Jitter):** Нестабильность задержки. Если пинг скачет: 20 мс, потом 150 мс, потом 40 мс — это плохо. Голос в Zoom будет рваным, звук пропадать. **Аналогия:** Вы ведёте машину с постоянной скоростью 60 км/ч (низкий джиттер). Потом резко давите на газ до 120, затем на тормоз до 20. Пассажиров (пакеты данных) укачивает, поездка (связь) становится некомфортной и неэффективной.
- **Надёжность (Uptime):** Процент времени, когда соединение доступно. Проводные технологии (оптика, кабель) обычно надёжнее беспроводных.

3. **Лимиты трафика.** Многие тарифы, особенно беспроводные (LTE, спутник) и устаревшие DSL, имеют ограничение на объём скачанных/отправленных данных в месяц. Для удалённой работы, которая может включать ежедневные видеовстречи и передачу больших файлов, необходим **безлимитный тариф**.
4. **Техническая поддержка провайдера.** Важно, чтобы в случае обрыва провайдер мог оперативно восстановить связь. Корпоративные тарифы часто включают приоритетное обслуживание.

Итоговая рекомендация для удалённого работника (в порядке приоритета):

1. **Оптоволоконный интернет (FTTH/PON):** Идеальный вариант. Высокая симметричная скорость (100/100 Мбит/с и выше), низкий пинг, высочайшая стабильность.
2. **Кабельный интернет (DOCSIS 3.1):** Отличная скорость, но нужно мониторить скорость отдачи и возможные проседания в пиковые часы.
3. **VDSL2:** Хороший вариант, если нет опции 1 и 2, а телефонная линия современная и расстояние до узла мало.
4. **Радиоинтернет/Fixed Wireless Access (4G/5G):** Как основной канал — рискованно из-за зависимости от погоды и нагрузки на вышку. Часто используется как **резервный (failover) канал** на случай проблем с основным проводным.
5. **Мобильный 4G/5G (через модем):** Решение для высокой мобильности или как временный/резервный вариант.
6. **ADSL и спутниковый интернет: НЕ РЕКОМЕНДУЮТСЯ** для постоянной полноценной удалённой работы из-за низкой скорости отдачи (ADSL) или очень высокой задержки (спутник).

Глава 4. Практические аспекты настройки: xDSL и PPPoE

4.1. Общая логика настройки подключения xDSL

Шаг 1: Аппаратное обеспечение.

- **DSL-модем или маршрутизатор с DSL-портом.** Это специализированное устройство, которое понимает, как работать с сигналом на телефонной линии.
- **Сплиттер (Splitter) или микрофильтры (Filters).** Сплиттер разделяет частоты: низкие (голос) идут на телефонный аппарат, высокие (данные) — на модем. Если телефон один, сплиттер ставится на входе в квартиру. Если розеток несколько, на каждую телефонную розетку, куда включён телефон или факс, ставится микрофильтр, а модем включается в розетку без него.

Шаг 2: Физическое подключение.

1. Телефонная линия от розетки подключается к входу сплиттера (LINE).
2. От выхода сплиттера (PHONE) кабель идёт к телефону.
3. От выхода сплиттера (MODEM/DSL) кабель идёт к порту DSL или LINE на модеме.
4. От порта LAN модема кабель идёт к компьютеру или к порту WAN домашнего Wi-Fi роутера (если модем без функции роутера).

Шаг 3: Настройка в веб-интерфейсе модема.

- Зайти в настройки модема (обычно через браузер, адрес 192.168.1.1 или 192.168.0.1).
- Выбрать тип соединения. Для большинства провайдеров в России это **PPPoE** или **PPPoE over ATM** (для старых линий).

- Ввести логин и пароль, выданные провайдером.
- Настроить параметры VPI/VCI (виртуальный путь/виртуальный канал). Это цифровые идентификаторы, которые говорят оборудованию провайдера, куда направлять ваш трафик. Например, 0/35 или 8/35. Эти значения уникальны для каждого провайдера.
- Применить настройки. Модем установит соединение.

4.2. Обзор и настройка PPPoE

PPPoE (Point-to-Point Protocol over Ethernet) — это протокол туннелирования, который инкапсулирует PPP-кадры внутри Ethernet-кадров.

Разбор смысла по аналогии:
Представьте, что вы отправляете письмо другу.

- **Ethernet** — это стандартная почтовая система с конвертами определённого размера и формата (кадры Ethernet). Она отлично работает внутри одной сети (например, в вашем доме или в офисе провайдера).
- Но провайдеру нужно **идентифицировать вас** среди миллионов абонентов, чтобы начислить плату и предоставить доступ. Простого адреса Ethernet (MAC-адреса) для этого недостаточно.
- **PPP (Point-to-Point Protocol)** — это протокол, созданный для установления **персональной, защищённой сессии** между двумя точками. Он умеет проводить аутентификацию (проверку имени и пароля) и назначать сетевые настройки (IP-адрес).
- **PPPoE** — это способ «упаковать» персональное PPP-сообщение (ваш логин/пароль и данные) в стандартный почтовый конверт Ethernet, чтобы отправить его по сети провайдера. На стороне провайдера PPP-сообщение «распаковывается», проверяется логин/пароль, и если всё верно, вам выделяется IP-адрес и открывается доступ в интернет.

Преимущества PPPoE для провайдера:

- Простая и знакомая система аутентификации (логин/пароль).
- Гибкость: пользователь может получить динамический или статический IP.
- Возможность учёта трафика и контроля сессии.

Настройка PPPoE на устройстве пользователя (маршрутизаторе или в Windows):

Вариант А: Настройка в маршрутизаторе (рекомендуется).

1. Подключитесь к веб-интерфейсу роутера.
2. Найдите раздел «Сеть» (Network) или «Интернет» (WAN).
3. Выберите тип подключения **PPPoE**.
4. В поля «Имя пользователя» (Username) и «Пароль» (Password) введите данные от провайдера.
5. Часто требуется задать имя службы (Service Name), его можно оставить пустым или ввести, если указано провайдером.
6. Установите режим подключения «Подключать автоматически» (Always On).

7. Сохраните настройки. Роутер установит PPPoE-соединение и раздаст интернет на все устройства в вашей домашней сети.

Вариант Б: Настройка PPPo-соединения в Windows (если компьютер подключён напрямую к модему).

1. Откройте «Панель управления» -> «Центр управления сетями и общим доступом» -> «Создание и настройка нового подключения или сети».
2. Выберите «Подключение к Интернету» -> «Высокоскоростное (с PPPoE)».
3. Введите логин и пароль, выданные провайдером. Можете задать имя подключения, например, «Домашний интернет».
4. Нажмите «Подключить». Windows создаст ярлык в списке сетевых подключений, через который можно устанавливать и разрывать это соединение.

Важное замечание: При использовании PPPoE на роутере **не нужно** создавать отдельное подключение на каждом компьютере. Роутер сам устанавливает сессию с провайдером, а для ваших домашних устройств работает как обычный шлюз (gateway) с технологией NAT.

Заключение

Тема широкополосного доступа для удалённой работы является ярким примером симбиоза бизнес-логики и сетевых технологий. Эффективный удалённый сотрудник — это не только дисциплинированный специалист, но и грамотный «сетевой инженер» своего домашнего сегмента. Понимание преимуществ и недостатков различных технологий (кабель, xDSL, беспроводные сети) позволяет сделать осознанный выбор, который напрямую влияет на продуктивность и качество связи. Умение настроить подключение, особенно такое распространённое, как PPPoE, является базовым практическим навыком. Освоив эти фундаментальные идеи, вы сможете не только обеспечить себя стабильным рабочим местом вне офиса, но и в будущем участвовать в проектировании и поддержке корпоративных систем удалённого доступа для целых организаций. Технологии широкополосного доступа — это тот самый цифровой мост, который соединяет талантливых людей с возможностями глобальной экономики, и понимание принципов его построения крайне важно для современного системного администратора.

Контрольные вопросы к ТЕМЕ-18

1. Что такое удалённая работа с точки зрения использования информационно-коммуникационных технологий?
2. Назовите три ключевых преимущества удалённой работы для сотрудника.
3. Какое бизнес-требование является самым критичным для безопасности при организации удалённой работы и почему?
4. Дайте определение широкополосного доступа в Интернет. Чем он принципиально отличается от устаревшего dial-up доступа?

5. В чём состоит главный недостаток кабельного интернета (DOCSIS) с точки зрения разделения пропускной способности?
6. Как основная технология DSL позволяет одновременно использовать телефонную линию для разговора и передачи данных?
7. Чем принципиально отличается асимметричный доступ (ADSL) от симметричного (SDSL)? Какой из них более предпочтителен для удалённой работы и почему?
8. Почему спутниковый интернет часто не подходит для видеоконференций и онлайн-игр, несмотря на возможную высокую скорость?
9. Какой параметр подключения, кроме скорости, является критически важным для качества видеозвонков и отзывчивости удалённого рабочего стола? Дайте его определение.
10. Перечислите технологии широкополосного доступа в порядке убывания предпочтительности для организации стационарного рабочего места удалённого сотрудника (от лучшей к худшей).
11. Какую функцию выполняет сплиттер (разделитель) в схеме подключения xDSL?
12. Какую основную задачу решает протокол PPPoE для интернет-провайдера при предоставлении доступа абоненту?

Практические задания к ТЕМЕ-18

Задание 1. Анализ и рекомендация для офиса на «удалёнке»

Ситуация: Небольшая IT-компания из Москвы решила перевести часть сотрудников на постоянную удалённую работу. Среди них — два ведущих программиста, дизайнер и менеджер проектов. Программистам необходимо часто загружать большие сборки кода на корпоративный Git-сервер и участвовать в длительных митингах с демонстрацией экрана. Дизайнеру передаёт макеты весом до 2-3 ГБ. Менеджер в основном работает в мессенджерах и проводит короткие планерки.

Задача: Основываясь на материале, проанализируйте и объясните, какой параметр широкополосного доступа будет **ключевым** при выборе тарифа для каждого из этих специалистов. Для кого из них не подойдет самый дешёвый тариф ADSL с конфигурацией 20 Мбит/с на скачивание и 1 Мбит/с на отдачу? Свой ответ обоснуйте.

Задание 2. «Пропавший» голос в телефонной линии

Ситуация: В офис небольшой фирмы в старом фонде города Тулы провели интернет по технологии ADSL. После установки и настройки модема сотрудники пожаловались, что интернет работает, но на городской телефонный номер, подключенный в этой же линии, теперь невозможно дозвониться — абоненты слышат только тишину или помехи.

Задача: Используя знания о принципах работы xDSL, предположите наиболее вероятную техническую ошибку, которую мог допустить монтажник. Что, скорее всего, он забыл установить или подключил неправильно? Как это исправить?

Задание 3. Выбор технологии для удалённого посёлка

Ситуация: Сетевому администратору поручено организовать стабильное удалённое рабочее место для бухгалтера, который переехал в сельский посёлок в Ленинградской области. Из

доступных вариантов от местных провайдеров есть только три: 1) Спутниковый интернет (нового поколения, с высокой скоростью загрузки). 2) ADSL по старой, но работающей телефонной линии. 3) 4G/LTE интернет от сотового оператора с хорошим уровнем сигнала.
Задача: Проанализируйте каждый вариант, учитывая специфику работы бухгалтера (работа в 1С через защищённое VPN-подключение, ежедневная отправка отчётных файлов, участие в аудиоконференциях). Какой вариант вы бы не рекомендовали категорически и почему? Какой вариант, вероятно, будет наилучшим компромиссным решением и требует дополнительной проверки?

Задание 4. Диагностика проблемы с видеосвязью

Ситуация: Сотрудник, работающий из дома в Новосибирске, жалуется на некачественную связь во время видеоконференций в Zoom. Изображение от него коллегам часто «зависает», звук пропадает и слышится рваным. При этом файлы из интернета он скачивает с нормальной высокой скоростью. Администратор попросил его измерить скорость на сайте speedtest.net. Результат: Download — 85 Мбит/с, Upload — 5 Мбит/с, Ping — 25 мс.
Задача: Основываясь на результатах теста и материале лекции, определите, какой параметр подключения, скорее всего, является причиной проблем. Является ли сам показатель скорости загрузки (Download) в 85 Мбит/с гарантией качества видеосвязи? Какой совет по изменению тарифа или технологии можно дать сотруднику?

Задание 5. Настройка резервного канала

Ситуация: Руководство компании в Екатеринбурге, где большинство сотрудников работает удалённо, требует от IT-отдела обеспечить бесперебойность связи. Критически важно, чтобы при обрыве основного проводного интернета у сотрудника связь восстанавливалась автоматически в течение минуты.
Задача: Предложите схему построения домашней сети для удалённого сотрудника, которая удовлетворяет этому требованию. Какое дополнительное оборудование и какой тип второго подключения необходимо использовать? Опишите принцип работы такой схемы, используя понятия «основной канал» и «резервный (failover) канал».

ТЕМА-19. Защита межфилиальной связи

Введение. Почему нужна защищённая связь между филиалами?

Представьте, что ваша компания — это государство со столицей (головной офис) и несколькими удалёнными городами (филиалы). Между ними постоянно курсируют важные грузы: финансовые отчёты, внутренние документы, конфиденциальные переговоры, доступ к базам данных. Если отправлять эти грузы обычной открытой почтой (как данные через публичный Интернет), их может перехватить, прочитать или подменить любой злоумышленник. Это как везти золотые слитки на открытой телеге по дороге, полной разбойников.

Для решения этой проблемы были созданы технологии **VPN (Virtual Private Network — Виртуальная Частная Сеть)**. Их ключевая идея — построить поверх ненадёжной публичной сети (Интернет) безопасный, защищённый «тоннель», по которому данные будут передаваться так, как будто все офисы соединены одной внутренней, частной сетью. Это похоже на строительство охраняемого подземного метро между городами поверх обычных дорог. Снаружи видно только входы и выходы, а что происходит внутри туннеля — скрыто.

1. Сети VPN: Основная идея и базовые формулировки

VPN (Virtual Private Network) — это обобщённое название технологии, которая создаёт защищённое логическое (виртуальное) сетевое соединение поверх другой сети (чаще всего Интернета). Ключевые слова здесь: **виртуальная** и **частная**.

- **Виртуальная**: Физически устройства не соединены прямыми выделенными кабелями. Они используют уже существующую инфраструктуру Интернета. Соединение существует не как физический объект, а как набор правил и параметров на устройствах.
    - **Аналогия**: Звонок по видеосвязи (Zoom, Skype). Физически вы и ваш собеседник находитесь в разных комнатах, но технология создаёт виртуальное пространство встречи, где вы можете общаться лицом к лицу.
- **Частная**: Данные внутри VPN изолированы от остального трафика Интернета. Только участники VPN могут получить к ним доступ, для всех остальных они выглядят как непонятный «шум».
    - **Аналогия**: Шифрованная рация. Два человека настраивают свои рации на одну секретную частоту и используют кодовый язык. Даже если кто-то будет слушать эфир, он услышит лишь шипение и неразборчивые звуки.

Основная цель VPN — обеспечить **конфиденциальность** (данные не могут прочитать посторонние), **целостность** (данные не могут быть незаметно изменены при передаче) и **доступность** (законные пользователи могут установить соединение) передаваемой информации.

2. Типы сетей VPN: Кто с кем соединяется?

VPN классифицируются по их архитектуре и целям использования. Два основных типа, которые нам необходимы:

2.1. VPN типа «Сеть-Сеть» (Site-to-Site VPN)

Это как построить защищённый мост между двумя зданиями. Постоянный туннель создаётся между двумя сетевыми устройствами (маршрутизаторами, межсетевыми экранами), расположенными в разных филиалах.

- **Как это работает**: Маршрутизатор в головном офице и маршрутизатор в филиале настраиваются как «концы» туннеля. Когда компьютер в филиале пытается отправить пакет в сеть головного офиса, локальный маршрутизатор «понимает», что путь лежит через туннель. Он шифрует этот пакет, упаковывает его в новый «конверт» с адресами маршрутизаторов и отправляет через Интернет. Получающий маршрутизатор в головном офисе распаковывает внешний «конверт», расшифровывает содержимое и передаёт исходный пакет нужному компьютеру в своей локальной сети.
- **Пример**: Главный офис в Москве и склад в Ростове. Любой компьютер в московской сети «видит» компьютеры в ростовской сети как соседние, локальные. Сотрудникам не нужно устанавливать специальное ПО, всю работу по организации туннеля берут на себя маршрутизаторы.

2.2. VPN удалённого доступа (Remote Access VPN)

Это как выдать доверенным сотрудникам особый ключ от служебного входа в здание. Временный туннель создаётся между отдельным устройством пользователя (ноутбук, смартфон) и специальным VPN-шлюзом в корпоративной сети.

- **Как это работает**: Сотрудник, находясь дома или в кафе, запускает на своём ноутбуке VPN-клиент (специальную программу). Клиент устанавливает защищённое соединение с VPN-шлюзом компании. После успешной проверки логина и пароля (часто ещё и с помощью одноразового кода) ноутбук сотрудника как бы «включается» в корпоративную сеть. Он получает внутренний IP-адрес и может работать с файловыми серверами, внутренними сайтами и базами данных, как если бы он физически сидел в офисе.
- **Пример**: Бухгалтер работает из дома. Она подключается через VPN к серверу «1С» в офисе. Все данные шифруются и передаются по туннелю, поэтому даже если её домашний Wi-Fi взломан, злоумышленник не сможет перехватить финансовую информацию.

3. Туннели GRE: «Голый» каркас туннеля

GRE (Generic Routing Encapsulation) — это простой протокол туннелирования. Его ключевая задача — **инкапсуляция**, то есть упаковка одного типа сетевых пакетов внутрь другого для передачи по чужой сети.

- **Основная идея GRE**: Взять оригинальный пакет (например, с приватными адресами) и «завернуть» его в новый IP-заголовок с публичными адресами маршрутизаторов. Сам GRE не обеспечивает шифрование или защиту данных! Он лишь создаёт виртуальную «трубу» между точками.
- **Аналогия**: Отправка секретного письма. Вы пишете письмо на родном языке (оригинальный пакет с приватным адресом), кладёте его в стандартный почтовый конверт с адресом и индексом (новый заголовок GRE с публичными IP), а затем — во внешний почтовый мешок для международной отправки (кадр для передачи по Интернету). Конверт GRE позволяет доставить письмо, но его содержимое не защищено — конверт можно вскрыть и прочесть.

Настройка туннеля GRE (принципиальная схема):

1. На маршрутизаторах двух филиалов определяются их «внешние» публичные IP-адреса в Интернете (например, 195.10.10.1 для Москвы и 200.20.20.1 для Ростова).
2. На каждом маршрутизаторе создаётся виртуальный **интерфейс туннеля**. Ему присваивается IP-адрес из новой, специальной подсети, например, 10.0.0.1 для Москвы и 10.0.0.2 для Ростова.
3. В конфигурации туннеля указывается:
   - **Источник туннеля** — публичный IP своего маршрутизатора.
   - **Назначение туннеля** — публичный IP удалённого маршрутизатора.
4. В таблицу маршрутизации добавляется правило: «Чтобы отправить данные в сеть филиала, отправляй их на IP-адрес виртуального интерфейса туннеля».

Недостаток GRE: Отсутствие безопасности. Поэтому GRE часто используется в паре с IPsec, который выполняет роль «бронирования» этой трубы.

4. IPsec: Броня для данных

IPsec (Internet Protocol Security) — это не один протокол, а целый комплекс протоколов и стандартов, созданный для защиты IP-трафика. Если GRE строит туннель, то IPsec его укрепляет, запирает на замок и ставит охрану.

Основные задачи IPsec:

1. **Конфиденциальность**: Шифрование данных. Достигается с помощью протокола **ESP (Encapsulating Security Payload)**.
2. **Целостность и аутентификация**: Проверка, что данные не были изменены в пути, и подтверждение подлинности отправителя. Используются протоколы **AH (Authentication Header)** и механизмы в **ESP**.

3. **Защита от повторов**: Предотвращение ситуации, когда злоумышленник перехватывает и повторно отправляет легитимный пакет (например, команду «перевести деньги»).

Структура и принцип работы IPsec (базовые идеи):

- **SA (Security Association — Ассоциация безопасности)**: Это фундаментальное понятие. SA — это «договорённость» между двумя устройствами о том, **как** защищать трафик. В нём хранится вся информация, необходимая для защиты данных: выбранные алгоритмы шифрования (AES, 3DES), алгоритмы проверки целостности (SHA, MD5), секретные ключи, сроки жизни и т.д. Для двусторонней связи нужно как минимум две SA (входящая и исходящая).
  - **Аналогия**: SA — это подробный протокол совместной работы двух спецагентов. В нём прописано: будем общаться на языке эсперанто (алгоритм AES), каждое сообщение заверяем отпечатком пальца (хеш SHA-256), ключ от шифра храним в сейфе №5, и менять всё это будем каждые 24 часа.
- **IKE (Internet Key Exchange — Обмен ключами по Интернету)**: Это протокол, который автоматически устанавливает эти самые SA. Его главная задача — безопасно, по незащищённой сети, договориться о секретных ключах шифрования.
  - **Как это происходит (упрощённо)**:

1. **Фаза 1**: Два устройства доказывают друг другу свою подлинность (по предварительному ключу, цифровым сертификатам) и создают **безопасный канал (ISAKMP SA)**. Этот канал нужен для защиты переговоров следующей фазы. Используется алгоритм Диффи-Хеллмана для генерации общего секретного ключа, даже если все переговоры перехватываются.
2. **Фаза 2**: Уже по защищённому каналу, созданному на Фазе 1, устройства договариваются о конкретных параметрах для защиты пользовательских данных. Создаются пары **IPsec SA** для туннеля.

- **Два режима работы IPsec**:

1. **Транспортный режим**: Защищается только **полезная нагрузка** (данные) IP-пакета, исходный IP-заголовок остаётся открытым. Используется чаще для связи «хост-хост» (например, между двумя серверами).
2. **Туннельный режим**: Защищается **весь исходный IP-пакет** (заголовок и данные). Весь этот блок шифруется и становится полезной нагрузкой нового пакета с новым заголовком. Именно этот режим используется для VPN «Сеть-Сеть». Он полностью скрывает структуру внутренней сети.

Связка GRE over IPsec — это классическое решение: GRE создаёт гибкий туннель, способный передавать разные типы трафика (мультикаст, например, для маршрутизации), а IPsec в туннельном режиме обеспечивает его шифрование и защиту.

5. Решения VPN для удалённого доступа: как подключить мобильного сотрудника

Для удалённого доступа чаще всего используются два подхода:

5.1. IPsec для удалённого доступа
Принцип похож на «Сеть-Сеть», но в роли одного из «концов» туннеля выступает не маршрутизатор филиала, а ПО на компьютере пользователя (IPsec-клиент). Клиент выполняет те же этапы IKE, устанавливает IPsec SA и подключается к корпоративному шлюзу. Это надёжный и безопасный метод.

5.2. SSL/TLS VPN (более современный и распространённый для удалённого доступа)
Здесь используется не IPsec, а протоколы безопасности, знакомые всем по веб-сайтам (HTTPS).

- **Как это работает**: Пользователь заходит обычным браузером на специальный адрес VPN-шлюза компании (например, vpn.company.com). Устанавливается стандартное защищённое HTTPS-соединение (то самое, с «замочком»). Далее, внутри этого защищённого сеанса, пользователь аутентифицируется, и шлюз предоставляет ему доступ к внутренним ресурсам. Часто используется тонкий клиент или Java-апплет, который перенаправляет трафик приложений.
- **Преимущества**:
  - **Универсальность**: Не требует установки специального ПО, достаточно браузера.
  - **Проходимость**: Трафик SSL выглядит как обычный веб-трафик на 443-м порту, что редко блокируется в общественных сетях (в отличие от специфичных портов IPsec).
  - **Гибкость доступа**: Можно настроить доступ не ко всей сети, а только к определённым веб-приложениям (например, только к веб-интерфейсу почты).

Заключение. Единая картина защиты межфилиальной связи

Подводя итог, защита связи между филиалами строится на комбинации рассмотренных технологий:

1. Для постоянной связи между офисами используется **VPN типа «Сеть-Сеть»**.
2. В её основе может лежать **GRE туннель**, обеспечивающий гибкую передачу трафика.
3. Для безопасности этот туннель обязательно защищается комплексом **IPsec**, который обеспечивает шифрование, аутентификацию и целостность данных, используя **IKE** для безопасного установления соединения.
4. Для мобильных сотрудников и работы из дома чаще всего применяются **VPN удалённого доступа** на основе **SSL/TLS**, как более удобные, или на основе **IPsec**, как более традиционные и глубоко интегрируемые в сеть.

Таким образом, современная корпоративная сеть превращается из набора изолированных островов в единое, безопасное и защищённое пространство, где географическое расстояние не является препятствием для безопасного обмена информацией. Изучение этих базовых

принципов — первый и ключевой шаг к пониманию практической организации таких защищённых сетей.

Контрольные вопросы к ТЕМЕ-19

1. Как расшифровывается аббревиатура VPN и какова основная цель этой технологии?
2. В чём заключается ключевое отличие VPN типа «Сеть-Сеть» от VPN удалённого доступа?
3. Приведите простую аналогию, поясняющую принцип работы VPN.
4. Что такое протокол GRE и какова его основная функция? Обеспечивает ли он безопасность данных?
5. Почему туннель GRE часто используют в комбинации с IPsec?
6. Перечислите три основные задачи (цели) протокола IPsec.
7. Что такое SA (Security Association) в контексте IPsec и для чего она нужна?
8. Какую функцию выполняет протокол IKE в процессе установления IPsec-соединения?
9. Назовите два основных режима работы IPsec и поясните, какой из них используется для организации VPN между филиалами.
10. Каков основной принцип работы SSL/TLS VPN для удалённого доступа и в чём его преимущество перед IPsec для мобильных пользователей?
11. Какой элемент VPN-архитектуры (маршрутизатор или ПО пользователя) отвечает за создание туннеля в схеме «Сеть-Сеть» и в схеме удалённого доступа?
12. Представьте, что сотрудник в кафе подключается к корпоративному файловому серверу. Объясните, какую роль в этой ситуации играет VPN.

Практические задания к ТЕМЕ-19

Задание 1. Анализ проекта для малого бизнеса

В туристической фирме «Алтай-Тур» в Барнауле есть головной офис и маленький филиал в Горно-Алтайске. Для связи используется обычный интернет, и директор жалуется на случайные утечки данных о клиентах при передаче отчётов. Вам предложили два бюджетных решения: 1) Настроить простой GRE-туннель между офисами. 2) Купить подписку на корпоративный мессенджер с шифрованием для пересылки файлов. Объясните, почему ни одно из этих решений полностью не подходит для защиты межфилиальной связи, и сформулируйте краткую рекомендацию на основе изученного материала.

Задание 2. Выбор типа VPN для сотрудников

Сеть магазинов «Домашний уют» в Ростовской области имеет центральный склад и 10 торговых точек. Администрации нужно предоставить удалённый доступ к системе складского учёта (1С) двум категориям сотрудников: 1) Статичным бухгалтерам в каждом магазине, которые работают со стационарных компьютеров. 2) Мобильному менеджеру по снабжению, который часто работает с ноутбука из гостиниц и кафе. Какой тип VPN (Сеть-

Сеть или удалённого доступа) вы порекомендуете для каждой категории и почему? Аргументируйте, исходя из логики работы этих технологий.

Задание 3. Поиск уязвимости в конфигурации

Молодой администратор в компании «Северсталь-Сервис» в Череповце настроил между центральным офисом и цехом туннель GRE. Он рапортует руководству, что связь между сетями работает, а данные теперь защищены. Его более опытный коллега, взглянув на конфигурацию, сказал, что защита не обеспечена, и это создаёт риски. Объясните, в чём прав опытный коллега, и какой ключевой элемент защиты забыл добавить молодой администратор, судя по изученному материалу.

Задание 4. Объяснение принципа работы для нетехнического руководства

Директору строительной компании «Фундамент» в Новосибирске вы предлагаете внедрить IPsec VPN для связи с удалённой проектной бригадой в Томске. Директор, не вдаваясь в технические детали, спрашивает: «Объясните мне на пальцах, как эта штука защитит наши чертежи? Что конкретно она делает с данными, когда их отправляют?». Сформулируйте максимально простой и наглядный ответ, используя не более трёх коротких предложений и обязательно употребив понятия «шифрование», «проверка целостности» и «виртуальный туннель».

Задание 5. Решение проблемы подключения удалённого работника

Сотрудник консалтинговой фирмы «Эксперт-Москва», находясь в командировке во Владивостоке в гостинице, не может подключиться к корпоративному IPsec VPN через клиент на своём ноутбуке. При этом он свободно заходит на публичный сайт своей компании. Вы предполагаете, что проблема может быть на стороне гостиничной сети. Основываясь на знании технологий VPN, выдвините две наиболее вероятные гипотезы, почему IPsec-подключение может блокироваться, и предложите, какое альтернативное решение для удалённого доступа (упомянутое в теории) с большей вероятностью сработает в этой ситуации и почему.

ТЕМА-20. Фундаментальные принципы безопасной сети

Введение: Почему безопасность сети — это фундамент, а не дополнение

Представьте, что вы строите дом. Вы можете выбрать красивые обои, современную технику, удобную планировку. Но всё это не будет иметь значения, если вы забудете залить фундамент или сделаете его из непрочных материалов. При первом же сильном дожде или небольшом землетрясении дом покосится, стены потрескаются, а жизнь в нём станет опасной. Сетевая безопасность в ИТ-инфраструктуре — это и есть тот самый фундамент. Это не «дополнительная опция», которую можно докупить позже, если останутся деньги. Это базовый принцип проектирования, установки и эксплуатации любой сети, от маленького офиса до глобальной корпоративной инфраструктуры.

Работа системного администратора, особенно в области сетей, часто сравнивается с работой архитектора и смотрителя крепости. Вы не просто подключаете провода и настраиваете маршрутизаторы. Вы проектируете цифровое пространство, которое должно быть одновременно удобным для своих обитателей (пользователей и служб) и неприступным для непрошеных гостей (злоумышленников). Игнорирование безопасности на этапе проектирования — это как строить крепость с широкими воротами, без стен и с картой всех потайных ходов, развешанной на ближайшем перекрёстке. Переделывать и укреплять уже работающую сеть всегда сложнее, дороже и менее эффективно, чем изначально заложить в неё принципы безопасности.

Цель этого материала — не сделать вас экспертами по кибербезопасности за один урок. Эта цель — заложить прочный концептуальный фундамент. Мы разберём, кто и зачем атакует сети, каким «инструментарием» они пользуются (вирусы, черви, трояны), и по каким основным «протоколам» действуют (методы атак). Понимание этих основ позволит вам в будущем грамотно подходить к настройке оборудования, разработке политик и реагированию на инциденты, видя за отдельными настройками их глубинный смысл и место в общей системе обороны.

Часть 1: Современный ландшафт угроз — Кто, кому и зачем?

Прежде чем изучать методы защиты, необходимо понять противника. Киберугрозы — это не абстрактное «зло», а целенаправленные действия конкретных субъектов с определёнными мотивами. Представьте, что вы охраняете склад. Ваши действия будут кардинально различаться в зависимости от того, кого вы ждёте: голодного бездомного, ищущего еду; вора-одиночку, нацеленного на конкретный товар; или хорошо обученный вооружённый отряд, планирующий захватить весь склад. В цифровом мире всё аналогично.

1.1. Субъекты угроз (Акторы)

Это «кто» в уравнении кибербезопасности. Их можно классифицировать по уровню подготовки, ресурсам и мотивации.

1. **Любопытствующие новички («Скрипткидди»):**

- **Кто:** Чаще всего подростки или начинающие энтузиасты с минимальными техническими навыками.
- **Инструменты:** Они не пишут свои программы, а используют готовые, скачанные из интернета инструменты для взлома (скрипты, эксплойты). Отсюда и название — «дети, использующие скрипты».
- **Мотив:** Не деньги или данные, а любопытство, желание потешить самолюбие, «повыпендриваться» перед друзьями, почувствовать свою силу. Их атака похожа на то, как подросток трясёт забор на стадионе, чтобы проверить, упадёт ли он. Они не планируют угнать стадион, им просто интересно.
- **Пример:** Студент колледжа, используя простую программу для сканирования портов, находит незащищённую веб-камеру в общежитии и начинает за ней наблюдать, чтобы потом похвастаться в чате.
- **Опасность:** Несмотря на низкую квалификацию, они опасны массовостью и непредсказуемостью. Их небрежные действия могут случайно нанести серьёзный ущерб, как ребёнок, случайно включивший пожарную сигнализацию и вызвавший панику.

2. **Киберпреступники:**

- **Кто:** Цифровые бандиты. Часто это организованные группы, работающие как бизнес.
- **Инструменты:** Профессиональный арсенал: собственное вредоносное ПО, фишинговые платформы, сети заражённых компьютеров (ботнеты).
- **Мотив:** Прямая финансовая выгода. Это их работа. Они атакуют, чтобы украсть деньги, данные для продажи, заблокировать доступ к информации за выкуп (шифровальщики).
- **Пример:** Финансовый директор компании получает письмо, якобы от генерального директора: «Срочно! Переведи 100 000 евро на этот счёт по контракту». Письмо идеально стилизовано, ссылка ведёт на поддельный корпоративный портал. Если директор введёт там свои данные, деньги уйдут преступникам. Это целевая фишинговая атака.
- **Опасность:** Высокая и прицельная. Они тратят время на разведку, выбирают самые уязвимые цели и наносят прямой финансовый ущерб. Это главный враг большинства компаний.

3. **Хактивисты:**

- **Кто:** Группы или индивидуумы, движимые политическими, идеологическими или социальными убеждениями.

- **Инструменты:** От дефейса (изменения главной страницы сайта) до DDoS-атак и утечек данных.
- **Мотив:** Не деньги, а послание. Они хотят привлечь внимание к проблеме, навредить репутации организации, которую считают виновной, «наказать» оппонента.
- **Пример:** Группа хактивистов, выступающая за защиту животных, организует DDoS-атаку на сайты крупных меховых фабрик, делая их недоступными для клиентов на несколько дней. Или взламывает сайт правительственного органа и размещает там политический манифест.
- **Опасность:** Репутационный и операционный ущерб. Атака может привести к простою сервисов и потере доверия клиентов.

4. **Государственные структуры (APT — Advanced Persistent Threat):**

- **Кто:** Группы, спонсируемые государствами. Элита киберпространства.
- **Инструменты:** Экспертные знания, нулевые дни (уязвимости, неизвестные публике), неограниченные ресурсы, физический доступ к инфраструктуре.
- **Мотив:** Шпионаж (промышленный, политический, военный), саботаж, дестабилизация. Их задача — долго и незаметно находиться в сети жертвы, собирая информацию или ожидая команды на действие.
- **Пример:** Группа APT внедряется в сеть энергетической компании через цепочку взломов: сначала взламывается компьютер инженера-подрядчика, через него получают доступ к сети подрядчика, а уже оттуда, используя доверительные отношения между сетями, проникают в закрытую сеть энергокомпании. Всё это происходит месяцами, без шума. Цель — изучить систему управления сетями, чтобы в будущем, в случае конфликта, иметь возможность отключить свет в целом регионе.
- **Опасность:** Критически высокая. Такие группы способны наносить ущерб национального масштаба. Защита от них — задача национальных киберкоманд и требует экстраординарных мер.

5. **Внутренние нарушители (Инсайдеры):**

- **Кто:** Самый коварный враг, потому что он уже внутри «крепости». Это может быть недовольный сотрудник, конкурент, завербованный штатный специалист.
- **Инструменты:** Их легитимный доступ к системам и данным. Им не нужно взламывать брандмауэр — у них есть ключ от двери.
- **Мотив:** Месть, корысть, шпионаж, неосторожность.
- **Пример:** Системный администратор, которого увольняют по сокращению, перед уходом создаёт скрытую учётную запись с правами администратора или устанавливает логическую бомбу — программу, которая через месяц после его ухода удалит ключевые конфигурационные файлы. Или сотрудник, случайно отправивший файл с персональными данными клиентов не тому адресату.
- **Опасность:** Высокая, так как их действия сложно отследить, и они уже находятся в зоне доверия. Принцип «разделения обязанностей» и «минимальных привилегий» создан именно для минимизации этого риска.

Понимание мотивации и возможностей этих акторов — первый шаг к построению адекватной защиты. Вы не будете строить противоатомный бункер от скрипткидди, но и не станете защищать государственные секреты паролем «123456».

1.2. Объекты угроз (Атаки на что?)

Атакуют не просто «компьютеры», а конкретные ценности, которые в цифровом мире называются **активами**.

1. **Конфиденциальность:** Принцип, гарантирующий, что информация доступна только тем, кто имеет на неё право. Атака на конфиденциальность — это **кража информации**. Пример: перехват пароля, кража базы данных клиентов, подслушивание сетевого трафика. Аналогия: кто-то вскрыл ваш почтовый ящик и прочитал все письма, или подслушал ваш телефонный разговор.
2. **Целостность:** Принцип, гарантирующий точность и полноту информации, защиту от несанкционированного изменения. Атака на целостность — это **подмена или искажение данных**. Пример: злоумышленник изменяет сумму перевода в банковском платеже или правит оценку в электронной зачётной книжке. Аналогия: кто-то проник в аптеку и поменял этикетки на пузырьках с лекарствами. Данные (этикетки) есть, но их содержание теперь ложно и опасно.
3. **Доступность:** Принцип, гарантирующий, что авторизованные пользователи имеют доступ к информации и связанным с ней активам, когда это необходимо. Атака на доступность — это **блокирование доступа к ресурсу**. Пример: DDoS-атака, которая перегружает сайт запросами, делая его недоступным для настоящих клиентов; или шифровальщик, который блокирует доступ к файлам на сервере. Аналогия: кто-то заблокировал вход в здание или спрятал ключи от сейфа, не взяв ничего внутри, но лишив вас возможности работать.

Эта триада — **C.I.A. (Confidentiality, Integrity, Availability)** — краеугольный камень информационной безопасности. Любая угроза, любой метод атаки направлен на нарушение одного, двух или сразу всех этих принципов. Ваша задача как администратора — строить защиту, обеспечивающую все три компонента.

Часть 2: Классический арсенал: Вирусы, черви и троянские кони

Это три классических типа вредоносного программного обеспечения (malware), которые часто путают. Понимание разницы между ними критически важно, так как от этого зависят методы распространения и защиты.

2.1. Компьютерные вирусы: Цифровые паразиты

- **Фундаментальная идея:** Вирус — это фрагмент вредоносного кода, который **для своего распространения и активации требует действия пользователя и наличия «хозяина» — другой программы или файла.** Он паразитирует на чём-то другом.
- **Аналогия:** Биологический вирус. Сам по себе он инертен. Чтобы размножаться и причинять вред, ему нужна живая клетка-хозяин. Человек (пользователь) должен «активировать» его, например, вдохнув (открыв заражённый файл).

- **Как работает:**

1. **Внедрение:** Вирус «вшивает» свой код в код другой, легитимной программы (например, в calc.exe — калькулятор) или в документ с макросами (Word, Excel).
2. **Активация:** Вирус ничего не делает, пока пользователь не запустит заражённую программу или не откроет заражённый документ с разрешением на выполнение макросов.
3. **Выполнение:** После запуска хозяина, первым делом выполняется код вируса. Он может:

- **Размножаться:** Найти на диске другие подходящие файлы и заразить их.
- **Нанести вред:** Удалить файлы, повредить данные, вывести на экран назойливое сообщение.
- **Скрыться:** Попытаться «спрятаться» от антивируса.

- **Пример:** Вирус ILOVEYOU (2000 год). Он рассылался по email как файл LOVE-LETTER-FOR-YOU.TXT.vbs. Пользователь, привлечённый темой любви, открывал вложение (действие пользователя!). Скрипт (VBS) активировался, рассылал себя всем контактам в адресной книге и перезаписывал файлы на компьютере. Он паразитировал на почтовом клиенте (Outlook) и действии пользователя.
- **Ключевой признак:** Не может распространяться самостоятельно.
  Требует **хозяина** и **действия пользователя**.

2.2. Сетевые черви: Самостоятельные агрессоры

- **Фундаментальная идея:** Червь — это **самостоятельная вредоносная программа, которая для своего распространения использует уязвимости в сетевых службах и операционных системах. Ей не нужна программа-хозяин и, зачастую, не нужно действие пользователя.**
- **Аналогия:** Настоящий червь, который самостоятельно ползает по саду (сети), проникая в новые растения (компьютеры) через естественные щели (уязвимости).
- **Как работает:**

1. **Сканирование:** Червь, попав на один компьютер, начинает автоматически сканировать сеть (или интернет) в поисках других машин.
2. **Эксплуатация:** Он ищет конкретную «дыру» — уязвимость в сетевой службе (например, в службе удалённого доступа, в операционной системе).
3. **Заражение:** Найдя уязвимую машину, червь **самостоятельно** передаёт на неё свой код и запускает его, без какого-либо ведома пользователя той машины.
4. **Цикл повторяется:** Новоинфицированная машина начинает сканировать сеть и заражать другие.

- **Пример:** Червь WannaCry (2017), который на самом деле был червём-шифровальщиком. Он использовал уязвимость в сетевом протоколе Windows (EternalBlue). Компьютеры, на которых не было установлено критическое обновление безопасности от Microsoft, были уязвимы. Червь, попав в сеть, автоматически находил такие компьютеры, заражал их, шифровал файлы и пытался заразить следующие. Скорость распространения была колоссальной именно из-за его самостоятельности.
- **Ключевой признак: Самостоятельное распространение** по сети через эксплуатацию уязвимостей. Действие пользователя не требуется.

2.3. Троянские кони (Трояны): Волк в овечьей шкуре

- **Фундаментальная идея:** Троян — это вредоносная программа, которая **маскируется под легитимное, полезное программное обеспечение, чтобы обмануть пользователя и побудить его к установке.** Её цель — не столько размножаться, сколько выполнить скрытую зловредную функцию.
- **Аналогия:** Легендарный Троянский конь. Жители Трои сами завезли в свой город огромного деревянного коня, думая, что это дар (полезная программа). Ночью из коня вышли воины (вредоносная нагрузка) и открыли ворота врагу.
- **Как работает:**

1. **Маскировка:** Троян выглядит как что-то нужное: crack для программы, бесплатный генератор ключей, новая игра, «важная» обновление Flash Player, якобы пришедшая по почте.
2. **Обман пользователя:** Пользователь **сам** скачивает и запускает трояна, думая, что получает пользу.
3. **Выполнение скрытой функции:** После установки троян начинает делать то, для чего был создан, скрывая свои действия:

- **Бэкдор (задняя дверь):** Открывает на компьютере скрытый сетевой порт, через который злоумышленник может получить полный удалённый контроль. Компьютер становится «зомби» в составе ботнета.
- **Кейлоггер:** Записывает все нажатия клавиш и отправляет злоумышленнику, похищая пароли, номера карт, переписку.
- **Прокси-сервер:** Превращает компьютер в перевалочный пункт для незаконного трафика, скрывая следы преступников.

- **Пример:** Пользователь ищет в интернете бесплатную (пиратскую) версию дорогого графического редактора. Находит сайт, качает файл Photoshop_Crack.exe. Запускает его. Программа, возможно, даже активирует редактор, но параллельно тихо устанавливает кейлоггер, который начинает перехватывать пароли от банковских сайтов.
- **Ключевой признак: Обман пользователя** для внедрения. Не распространяется самостоятельно и не является паразитом для других файлов. Его сила — в социальной инженерии.

Важное резюме:

- **Вирус:** «Зарази меня, запустив свою программу».
- **Червь:** «Я сам найду дыру в твоей системе и проникну в неё».
- **Троян:** «Я полезный подарок, открой меня, и ты всё поймёшь... позже».

В современном мире вредоносное ПО редко встречается в «чистом» виде. Чаще это **гибриды**. Например, почтовый червь, который рассылает себя как троянское вложение, а внутри содержит вирусный код для заражения файлов. Но понимание базовых моделей поведения позволяет предсказать векторы атаки и выбрать правильные средства защиты: антивирус (для файлов), межсетевые экраны и обновления (от червей), обучение пользователей (от троянов).

Часть 3: Методы атак — Как ломают фундамент

Зная противника и его инструменты, перейдём к тактике. Методы атак — это конкретные техники, используемые для нарушения конфиденциальности, целостности или доступности.

3.1. Социальная инженерия: Взлом самого слабого звена — человека

Это не технический, а психологический метод. Его цель — обманом заставить человека совершить действие, нарушающее безопасность, или раскрыть информацию.

- **Фундаментальная идея:** Легче обмануть человека, чем взломать сложную криптосистему. Доверие, страх, жадность, любопытство — вот уязвимости, которые эксплуатирует социальный инженер.
- **Основные виды:**

1. **Фишинг (Наживка):** Массовая рассылка писем или сообщений, имитирующих легитимного отправителя (банк, соцсеть, коллега). Цель — заставить перейти по ссылке на поддельный сайт и ввести учётные данные.

- **Пример:** «Ваш аккаунт Яндекс.Денег заблокирован! Срочно перейдите по ссылке для разблокировки». Ссылка ведёт на yandex-dengi.secure-login[.]com (поддельный домен), где ждут ваш логин и пароль.

2. **Целевой фишинг (Спишинг):** Точечная, тщательно подготовленная атака на конкретного человека или отдел (например, бухгалтерию). Письмо персонифицировано, содержит реальные имена, ссылки на внутренние события.
3. **Претекстинг:** Создание вымышленного, но правдоподобного сценария (претекста) для получения информации. Злоумышленник представляется техподдержкой, сотрудником охраны, аудитором.

- **Пример:** Звонок в отдел: «Здравствуйте, это Иван из техподдержки. У нас сбой в системе учёта паролей. Для восстановления вашего дохода, подтвердите, пожалуйста, ваш текущий пароль — мы его сверим и перевыпустим ключ».

4. **Кви про кво (Что-то за что-то):** Злоумышленник предлагает «помощь» или «вознаграждение» в обмен на информацию или действие.

- **Пример:** «Я новый стажёр в IT, мне нужен доступ к общей папке для обучения, дайте ваш пароль, я посмотрю и сразу выйду». Или раздача на улице «бесплатных» флешек с логотипом компании, на которых записан троян.

Защита: Обучение, бдительность, проверка подлинности запросов через альтернативные каналы (не по телефону, с которого позвонили, а по официальному номеру компании).

3.2. Атаки на доступность: DDoS-атаки

- **Фундаментальная идея:** Перегрузить целевой ресурс (сервер, канал связи) огромным количеством бесполезных запросов, чтобы исчерпать его вычислительную мощность, память или пропускную способность. Настоящие пользователи не могут получить доступ — доступность нарушена.

- **Аналогия:** Тысячи людей (ботов) одновременно начинают звонить на телефонную линию популярной пиццерии. Линия постоянно занята. Настоящие клиенты, желающие заказать пиццу, не могут дозвониться. Пиццерия теряет заказы.
- **Как работает:**

1. **Формирование ботнета:** Злоумышленник заражает множество компьютеров по всему миру троянами, создавая сеть «зомби» (ботнет).
2. **Команда на атаку:** По сигналу все компьютеры в ботнете начинают одновременно отправлять запросы на один целевой сервер.
3. **Перегрузка:** Сервер тратит все ресурсы на обработку этих фальшивых запросов и «ложится» или начинает крайне медленно отвечать.

- **Пример:** Атака на серверы онлайн-игры перед крупным турниром. Конкуренты или хактивисты нанимают ботнет, чтобы лишить игроков возможности тренироваться и сорвать мероприятие, нанеся ущерб репутации компании-организатора.

Защита: Специализированные сервисы очистки трафика (DDoS-протекция), фильтрация на уровне сети, резервирование каналов связи.

3.3. Атаки на целостность и конфиденциальность: Man-in-the-Middle (MITM)

- **Фундаментальная идея:** Злоумышленник тайно внедряется в канал связи между двумя сторонами (например, между вашим ноутбуком и точкой Wi-Fi в кафе) и получает возможность перехватывать, читать и даже модифицировать передаваемые данные.
- **Аналогия:** Вы отправляете другу письмо через почту. Злоумышленник перехватывает его на почтамте, читает, может что-то дописать или заменить, а затем отправляет дальше. Друг получает письмо, даже не подозревая, что оно было прочитано и изменено.
- **Как работает (упрощённо):**

1. **Перехват сессии:** Злоумышленник в той же сети заставляет трафик жертвы проходить через свой компьютер (например, подделывая ответы ARP-протокола — ARP-spoofing).
2. **Расшифровка (если возможно):** Если соединение не защищено (HTTP) или используется слабое шифрование, атакующий читает всё: пароли, переписку, номера карт.
3. **Подмена данных:** Может, например, подменить номер счёта в банковском переводе, который вы только что отправили.

- **Пример:** Вы в кафе подключаетесь к открытой Wi-Fi сети Free_Cafe_WiFi. Рядом сидящий злоумышленник со своим ноутбуком с помощью специальной программы организует MITM-атаку. Когда вы заходите на сайт без HTTPS (с простым HTTP), он видит все данные, которые вы вводите.

Защита: Всегда использовать защищённые протоколы (HTTPS, VPN), не доверять открытым публичным сетям без дополнительного шифрования, проверять SSL-сертификаты сайтов.

3.4. Эксплуатация уязвимостей программного обеспечения

- **Фундаментальная идея:** Любая сложная программа содержит ошибки (баги). Некоторые из этих ошибок создают бреши в системе безопасности — **уязвимости**. Злоумышленник

находит такую уязвимость и создаёт для неё **эксплойт** — специальный код, который эту уязвимость использует для получения контроля над системой.

- **Аналогия:** В идеально спроектированной крепости есть потайная дверь, которую забыли запереть (уязвимость). Враг узнаёт об этой двери (находит уязвимость), делает по её чертежам отмычку (эксплойт) и проникает внутрь.
- **Типы уязвимостей:**
  - **Переполнение буфера:** Самая известная класс уязвимостей. Если программа не проверяет длину вводимых данных, злоумышленник может отправить такой большой объём данных, который «перельётся» через отведённую область памяти (буфер) и затопит соседнюю область, содержащую код. Этим кодом можно подменить свой, заставив процессор выполнить команды злоумышленника.
  - **Межсайтовый скриптинг (XSS):** Уязвимость веб-приложений, позволяющая внедрить вредоносный скрипт на страницу, которую видят другие пользователи. Скрипт может украсть их куки (сессии) или перенаправить на фишинговый сайт.
  - **SQL-инъекция:** Уязвимость, позволяющая внедрить произвольный SQL-код в запрос к базе данных. Можно прочитать, изменить или удалить данные из БД, обойдя систему авторизации.
- **Пример «Нулевого дня»:** Уязвимость, о которой неизвестно разработчику. Пока она не обнаружена и не закрыта патчем, эксплойт для неё практически неуязвим для традиционных средств защиты. Именно такие уязвимости часто используют APT-группы и создатели сетевых червей.

Защита: Своевременное обновление всего ПО (ОС, приложения, прошивки устройств) — это самый важный и простой способ закрыть известные уязвимости. Использование систем обнаружения/предотвращения вторжений (IDS/IPS).

Заключение: Фундамент в действии — Принципы построения безопасной сети

Изучив угрозы и методы атак, можно сформулировать основные фундаментальные принципы, которые должны лечь в основу вашей работы как сетевого администратора.

1. **Принцип минимальных привилегий:** Каждый пользователь, каждая программа, каждое системное过程 должны обладать **ровно тем набором прав и доступов, который необходим для выполнения их прямых задач, и не более того.** Не давайте бухгалтеру права администратора на сервер, не запускайте веб-сервер от имени суперпользователя root.
2. **Принцип разделения обязанностей:** Критически важные операции (например, выпуск платежа, изменение конфигурации ядра сервера) должны требовать участия нескольких лиц или систем. Это защищает от ошибок одного человека и от действий инсайдера-одиночки.
3. **Принцип глубокой эшелонированной обороны (Defense in Depth):** Нельзя полагаться на один рубеж обороны (например, только на антивирус или только на брандмауэр). Защита должна быть многоуровневой: обученные пользователи (соц. инженерия) -> антивирус на рабочих станциях (вирусы, трояны) -> межсетевой экран и IPS на границе сети (черви, атаки извне) -> разделение сети на сегменты (VLAN) -> строгий контроль доступа внутри сети ->

шифрование критичных данных -> своевременное обновление (уязвимости) -> системы мониторинга и анализа журналов (обнаружение аномалий). Если злоумышленник преодолеет один уровень, он должен встретить следующий.

4. **Принцип «открытого дизайна» (Керкгоффса):** Безопасность системы должна зависеть не от секретности её устройства или алгоритмов, а от секретности относительно небольшого, легко сменяемого элемента — **ключа**. Пароли, сертификаты, токены должны быть хорошо защищены, а сама архитектура сети может быть известна. Нельзя полагаться на «безопасность через неясность».
5. **Принцип постоянного мониторинга и улучшения:** Безопасность — не состояние, а процесс. Новые угрозы появляются каждый день. Необходимо постоянно отслеживать события в сети (анализ логов), изучать новости об уязвимостях, тестировать свою защиту (пентесты), обучать сотрудников и адаптировать политики безопасности.

Понимание этих принципов, в сочетании с практическими навыками настройки сетевого оборудования, позволит вам не просто выполнять инструкции, а **осмысленно проектировать и поддерживать сетевую инфраструктуру, где безопасность будет неотъемлемой частью её фундамента.** Вы будете знать, зачем ставите то или иное правило в брандмауэре, почему сегментируете сеть и почему так важно требовать от коллег регулярно менять пароли. Это и есть основа профессионализма в области сетевого и системного администрирования.

Контрольные вопросы к ТЕМЕ-20

1. Что такое триада CIA в информационной безопасности и какие три принципа она включает?
2. Чем отличается компьютерный вирус от сетевого червя по способу распространения?
3. Какой главный признак троянской программы (трояна)?
4. Назовите два основных мотива действий киберпреступников (в отличие от хактивистов).
5. Что такое APT-группы и чем они характеризуются?
6. Какая угроза исходит от внутреннего нарушителя (инсайдера)?
7. Какой метод атаки направлен в первую очередь на нарушение доступности (Availability) ресурса?
8. Что такое фишинг и в чём его основная цель?
9. Какая базовая защитная мера является наиболее эффективной против эксплуатации известных уязвимостей программного обеспечения?
10. В чём заключается суть атаки "Человек посередине" (Man-in-the-Middle)?
11. Что такое принцип минимальных привилегий в информационной безопасности?
12. Как называется стратегия защиты, предполагающая создание нескольких уровней (эшелонов) безопасности?

Практические задания к ТЕМЕ-20

Задание 1. Анализ инцидента в городской поликлинике

В компьютерной сети городской поликлиники в Иваново произошёл инцидент. На компьютере в регистратуре появилось окно с требованием заплатить 30 000 рублей биткоинами для расшифровки файлов с расписанием врачей и данными пациентов. Технический специалист установил, что всё началось с письма, пришедшего на внутреннюю почту с темой «Срочное обновление программы «Электронная регистратура» от Минздрава». Сотрудник регистратуры открыл вложение и запустил файл Update_MedPortal.exe.

К какому основному типу вредоносного ПО (вирус, червь, троян) с большой вероятностью относится эта угроза, исходя из способа проникновения? Объясните свой ответ, опираясь на признаки из учебного материала. На нарушение какого именно принципа триады CIA была направлена эта атака?

Задание 2. Рекомендации для интернет-магазина «Уральские самоцветы»

Владелец небольшого интернет-магазина сувениров и украшений «Уральские самоцветы» из Екатеринбурга обратился к вам за консультацией. Он жалуется, что в последний месяц несколько раз падала скорость работы сайта, а однажды он был полностью недоступен около часа. При этом аналитики хостинга показали, что в эти моменты на сайт приходили сотни тысяч запросов с разных адресов, большинство из которых вели на одну и ту же страницу каталога, но никаких реальных заказов не оформляли.

Какой метод атаки, описанный в материале, вероятнее всего, использовался против сайта магазина? Кого (какого субъекта угроз) вы могли бы заподозрить в первую очередь, если бы атака носила не массовый, а целенаправленный характер против именно этого бизнеса? Дайте краткое обоснование для каждого из двух ваших предположений.

Задание 3. Расследование утечки в проектной организации

В проектном институте «СибГидроПроект» в Новосибирске произошла утечка конфиденциальных чертежей нового объекта. Расследование показало, что файлы были скопированы с сервера на личную флешку одним из инженеров за неделю до его увольнения по собственному желанию. При этом у инженера был стандартный доступ к этому сетевому ресурсу для выполнения своих обязанностей.

Нарушение какого фундаментального принципа безопасности стало ключевой причиной успеха этого инцидента? Предложите одну конкретную и простую меру, которую администратор мог бы реализовать на основе другого принципа безопасности, чтобы минимизировать риск подобных событий в будущем.

Задание 4. Планирование защиты для пункта выдачи заказов

Вы помогаете настроить сеть для нового пункта выдачи заказов крупного маркетплейса в Казани. В небольшом помещении есть:

1. Рабочее место менеджера с ПК для работы с 1С и почтой.
2. Открытая бесплатная Wi-Fi сеть для клиентов, ожидающих заказы.
3. Принтер для печати этикеток, подключенный к сети.

Используя принцип глубокой эшелонированной обороны (Defense in Depth), предложите два несложных технических или организационных мероприятия разного типа (например, не оба технических), которые вы порекомендуете внедрить для повышения безопасности. Объясните, от какой примерно угрозы или метода атаки помогает каждое из ваших предложений.

Задание 5. Объяснение для бухгалтера «ВостокСтрой»

Бухгалтер ООО «ВостокСтрой» из Хабаровска получила по рабочей почте письмо, якобы от генерального директора. В письме был текст: «Наталья, срочно необходимо оплатить аванс по новому контракту. Все детали и реквизиты здесь: [ссылка]. Жду подтверждения». Ссылка вела на сайт vostok-stroy.office365-authorize[.]com. Бухгалтер, заподозрив неладное, обратилась к вам.

Используя знания из теоретического материала, объясните ей максимально просто и понятно:
а) Как называется этот метод атаки?
б) Какой субъект угроз (актор), скорее всего, стоит за такой атакой и какова его основная цель?
в) Какое самое простое действие она должна предпринять, чтобы проверить подлинность запроса, прежде чем что-либо делать?

ТЕМА-21. Безопасность Сетевых устройств

Введение

Современная сеть — это не просто набор кабелей, коммутаторов и маршрутизаторов, соединённых между собой. Это сложный цифровой организм, жизненно важный для функционирования любой организации. Представьте, что сетевое устройство (например, маршрутизатор или коммутатор) — это крепость, которая охраняет информационные потоки внутри вашей компании. Если враг захватит эту крепость, он получит контроль над всеми коммуникациями: сможет перехватывать конфиденциальные данные, останавливать работу отделов или даже парализовать всю организацию. Поэтому безопасность сетевых устройств — это не дополнительная опция, а фундаментальная необходимость, основа, на которой строится вся защита информационной инфраструктуры.

Безопасность сетевых устройств охватывает четыре ключевых аспекта, которые мы подробно разберём:

1. **Безопасный доступ к устройствам** – Кто и как может попасть в "крепость"? Как убедиться, что это именно тот, за кого он себя выдает?
2. **Назначение административных ролей** – Что конкретно может делать каждый, кто попал внутрь? Можно ли всем доверять всё?
3. **Мониторинг и управление устройствами** – Как постоянно следить за состоянием "крепости", вовремя замечать угрозы и эффективно управлять ею?
4. **Использование функции автоматизированной настройки безопасности** – Как можно поручить рутинную работу по защите "умным" системам, чтобы сосредоточиться на главном?

Каждый из этих аспектов — это слои защиты, словно кольца крепостной стены. Чем больше слоёв и чем они надёжнее, тем сложнее злоумышленнику достичь своей цели. Давайте начнём изучение с первого и самого критичного слоя — обеспечения безопасного доступа.

1. Безопасный доступ к устройствам

Безопасный доступ — это основа основ. Если мы не можем контролировать, кто подключается к нашему сетевому устройству для управления им, то все остальные меры безопасности теряют смысл. Это как установить самый сложный замок на дверь, но оставить ключ под ковриком.

Фундаментальная идея: Безопасный доступ — это комплекс методов и технологий, предназначенных для строгой аутентификации (проверки личности), авторизации

(определения прав) и учёта (фиксации действий) всех лиц и систем, которые пытаются получить управляющий доступ к конфигурации сетевого устройства.

Смысл и аналогия: Представьте, что вы — директор завода. У вас есть кабинет с секретными чертежами. Доступ в кабинет должны иметь только определённые сотрудники (аутентификация). Но даже среди них главный инженер может брать чертеги на стол для изучения (читать конфигурацию), а его помощник — только смотреть на них через стеклянную витрину (просматривать статусы интерфейсов). Уборщице же вход в кабинет запрещён вовсе (авторизация). Кроме того, в журнале охраны фиксируется, кто и когда заходил в кабинет и что выносил (учёт). Сетевые устройства требуют такого же, а часто и более строгого, подхода.

Рассмотрим ключевые механизмы безопасного доступа.

1.1. Методы аутентификации: "Кто ты?"

Аутентификация отвечает на вопрос: "Действительно ли пользователь тот, за кого себя выдаёт?".

1.1.1. Локальные учётные записи

Это самый простой метод. Имя пользователя и пароль хранятся непосредственно в памяти самого сетевого устройства (например, маршрутизатора Cisco в конфигурационном файле).

- **Пример:** `username admin secret MyStrongP@ssw0rd`. Эта команда создаёт на устройстве пользователя `admin` с паролем `MyStrongP@ssw0rd`.
- **Аналогия:** Это как список доверенных лиц, записанный в блокноте, который хранится внутри самой охраняемой крепости. Чтобы войти, нужно назвать имя из списка и сказать кодовое слово (пароль).
- **Плюсы:** Простота настройки, работает всегда, даже если внешние системы недоступны.
- **Минусы и риски:**
  - **Слабость паролей:** Если пароль простой (`12345`, `admin`), его легко угадать или подобрать (атака brute-force).
  - **Сложность управления:** Если в сети 100 устройств, и уволился один администратор, пароль нужно сменить на всех 100 устройствах вручную.
  - **Отсутствие детализации:** Часто все администраторы используют одну учётную запись `admin`, и в журналах невозможно понять, кто конкретно выполнил те или иные действия.

1.1.2. Аутентификация через внешний сервер (AAA-сервер)

Это профессиональный и рекомендуемый подход. AAA означает Authentication, Authorization, Accounting (Аутентификация, Авторизация, Учёт). Устройство не хранит пароли само, а пересылает запрос на проверку на специальный защищённый сервер, чаще всего использующий протоколы RADIUS или TACACS+.

- **Пример:** Администратор подключается к коммутатору и вводит свой логин и пароль. Коммутатор не проверяет их сам, а отправляет эти данные на сервер TACACS+. Сервер сверяет их со своей базой и отвечает коммутатору: "Да, пользователь ivanov существует, и пароль верный. Пропусти его.".
- **Аналогия:** Вместо блокнота внутри крепости есть телефонная связь с центральным штабом охраны (сервером). Когда к воротам подходит человек, часовой звонит в штаб, называет имя пришедшего и его кодовое слово. Штаб проверяет их по своей центральной, всегда актуальной базе и отдаёт приказ: "Впустить" или "Задержать!".
- **Плюсы:**
  - **Централизованное управление:** Учётные записи хранятся в одном месте. Смена пароля или блокировка пользователя происходит мгновенно для всех устройств в сети.
  - **Сильная аутентификация:** Поддержка двухфакторной аутентификации (например, пароль + код из приложения на телефоне).
  - **Подробный учёт действий:** Сервер TACACS+ может записывать каждую введённую команду, что критично для расследования инцидентов.
- **Минус:** Появляется единая точка отказа. Если сервер AAA недоступен, администраторы могут не попасть на устройства. Для избежания этого настраивают несколько резервных серверов и локальную учётную запись для экстренных случаев.

1.1.3. Двухфакторная аутентификация (2FA)

Это дополнение к предыдущим методам, которое радикально повышает безопасность. Пользователь должен предоставить два доказательства своей личности (фактора):
1. **Что-то, что он знает** (пароль, PIN-код).
2. **Что-то, что у него есть** (телефон с генератором кодов, USB-токен, смарт-карта) или **что-то, что является частью его** (отпечаток пальца, скачок радужной оболочки).

- **Пример:** Администратор хочет подключиться к межсетевому экрану. Сначала он вводит имя пользователя и сложный пароль (первый фактор). Затем система просит ввести временный шестизначный код, который появляется в мобильном приложении Google Authenticator на его телефоне (второй фактор). Только после ввода верного кода доступ разрешается.
- **Аналогия:** Чтобы попасть в сверхсекретный объект, недостаточно назвать кодовое слово (пароль). Нужно также приложить специальную пропускную карту к считывателю (второй фактор). Даже если злоумышленник подслушал кодовое слово, без карты он останется снаружи.

1.2. Защита каналов управления: "Как ты передаёшь пароль?"

Не менее важно, *каким путём* передаются учётные данные. Передача пароля в открытом виде — это катастрофа.

1.2.1. Небезопасные протоколы (Telnet, HTTP)

Протокол Telnet и обычный HTTP передают все данные, включая логины и пароли, в незашифрованном, текстовом виде.

- **Пример:** Администратор использует программу PuTTY для подключения к маршрутизатору по Telnet. Пакеты идут по сети. Злоумышленник, подключившийся к тому же коммутатору, запускает программу-анализатор трафика (сниффер) и видит в потоке данных строку: `username=admin&password=MyPass`. Пароль скомпрометирован.
- **Аналогия:** Вы диктуете секретный пароль своему помощнику по рации, которая транслирует ваш голос на частоте, доступной всем окружающим. Любой с такой же рацией может его услышать.

1.2.2. Безопасные протоколы (SSH, HTTPS)
Протоколы SSH (для командной строки) и HTTPS (для веб-интерфейса) используют криптографию. Перед началом передачи данных между клиентом и устройством устанавливается зашифрованный "туннель". Всё, что проходит через него, выглядит как бессмысленный набор символов для перехватчика.

- **Пример:** Тот же администратор подключается к маршрутизатору, но использует протокол SSH. Сниффер видит лишь непонятные данные: `3e8a...1f4c...a9b2...`. Расшифровать их без секретного ключа практически невозможно.
- **Аналогия:** Вы и ваш помощник договорились о секретном шифре (ключе). Теперь, говоря по открытой рации, вы передаёте не сам пароль, а зашифрованный набор слов. Для посторонних это просто шум.
- **Рекомендация:** Использование Telnet и HTTP для управления должно быть **категорически запрещено** в производственных сетях. Допустимы только SSH (порт 22) и HTTPS (порт 443).

1.3. Ограничение источников доступа

Даже с сильной аутентификацией и шифрованием нужно контролировать, *откуда* можно подключаться.

Фундаментальная идея: Не все компьютеры в сети должны иметь право управлять сетевыми устройствами. Доступ должен быть разрешён только с доверенных, специально выделенных для этого узлов.

- **Механизм:** Использование списков контроля доступа (ACL), которые настраиваются на виртуальных интерфейсах управления (VTY lines для SSH/Telnet) или на интерфейсе устройства.
- **Пример:** На маршрутизаторе настраивается правило: "Разрешить подключения по SSH только с IP-адреса 10.0.10.5 (рабочая станция администратора) и 10.0.20.10 (сервер управления). Все остальные подключения отвергать."
- **Аналогия:** В крепости есть несколько потайных входов. Но охрана знает, что легальные посыльные приходят только с двух определённых дорог (доверенные IP-адреса). Любой, кто подходит с другой стороны, считается врагом, даже если он правильно назвал пароль. Возможно, пароль был перехвачен, и теперь им пытается воспользоваться злоумышленник с другого места.

Итог по безопасному доступу: Мы должны гарантировать, что: 1) пользователь — это именно тот, за кого он себя выдаёт (сильная аутентификация, в идеале 2FA), 2) его учётные данные передаются по защищённому каналу (SSH/HTTPS), и 3) он подключается только с разрешённого, доверенного места (ограничение по IP). Только после выполнения этих условий можно переходить к определению того, что именно этот пользователь может делать внутри системы — к назначению административных ролей.

2. Назначение административных ролей

Представьте, что после строгой проверки мы впустили человека в нашу цифровую "крепость" — сетевое устройство. Теперь возникает новый вопрос: а что ему здесь *можно* делать? Можно ли доверить новому стажёру изменение маршрутизации всей компании? Или стоит ли главному инженеру тратить время на перезагрузку портов? Ответ на эти вопросы даёт система административных ролей и уровней привилегий.

Фундаментальная идея: Принцип наименьших привилегий (Principle of Least Privilege, PoLP) — пользователь должен обладать только теми правами доступа, которые абсолютно необходимы для выполнения его прямых служебных обязанностей, и ничем более.

Смысл и аналогия: В больнице есть главный врач, хирург, медсестра и санитар. У всех есть доступ в здание (аутентификация), но их права внутри сильно различаются.

- **Главный врач** может зайти в любой кабинет, просмотреть любые истории болезни и отдать распоряжение любому сотруднику (полный доступ privilege level 15).
- **Хирург** имеет полный доступ в операционную и к инструментам, но не может увольнять медсестёр (ограниченные права на управление кадрами).
- **Медсестра** может вводить лекарства по назначению врача и вести графики, но не может назначать лечение (права на чтение конфигурации и выполнение ограниченного набора команд).
- **Санитар** может убирать помещения, но не имеет доступа к аптечке и медицинским картам (только просмотр статуса устройств, без доступа к конфигурации).

Если дать санитарю ключи от аптеки, это создаст ненужный риск. Точно так же в сетевом устройстве.

2.1. Уровни привилегий (Privilege Levels)

Многие сетевые устройства (особенно от Cisco) используют иерархическую систему уровней привилегий от 0 до 15.

- **Уровень 0 (user EXEC mode):** Минимальные права. Только несколько команд для просмотра (show, ping). Пользователь видит лишь общую информацию, но не может что-либо изменить.

Аналогия: гость на заводе, который может посмотреть на работу конвейера из специальной галереи, но не имеет доступа к цеху.

- **Уровень 1-14:** Промежуточные, настраиваемые уровни. Администратор сам решает, какие команды доступны на каждом уровне. Например, на уровне 7 можно разрешить команды для сброса интерфейсов (shutdown/no shutdown), но запретить изменение паролей.
- **Уровень 15 (privileged EXEC mode или enable mode):** Полный, неограниченный доступ ко всем командам устройства. Аналогия: главный инженер завода с мастер-ключом от всех помещений.

Пример настройки (Cisco):

```
! Создание пользователя с ролью "Оператор" (только просмотр)
username operator privilege 5 secret OperPass123
! На уровень привилегий 5 назначаем конкретные команды
privilege exec level 5 show running-config
privilege exec level 5 show interfaces status
! Этому пользователю запрещены команды configure terminal, reload, copy run start
```

2.2. Ролевая модель доступа (Role-Based Access Control, RBAC)

RBAC — это более гибкая и современная система, чем фиксированные уровни привилегий. Вместо привязки к числу (уровню 5) создаются **именованные роли (views)**, которым явно назначается набор разрешённых команд.

- **Пример ролей:**
  - **ROLE_MONITOR:** Просмотр всех параметров устройства (show), ping, traceroute. Для сотрудников техподдержки 1-й линии.
  - **ROLE_OPERATOR:** Все права ROLE_MONITOR + сброс интерфейсов, управление портами доступа. Для сетевых техников.
  - **ROLE_ADMIN:** Все права ROLE_OPERATOR + настройка маршрутизации, списков доступа, системных параметров. Для ведущих инженеров.
  - **ROLE_SUPERADMIN:** Полный доступ, включая управление другими пользователями и ролями. Для архитекторов сети.
- **Преимущества RBAC:**

1. **Наглядность:** Имя роли (ROLE_OPERATOR) понятнее, чем абстрактный "уровень 7".
2. **Гибкость:** Легко добавить новую команду в существующую роль или создать новую роль под специфичную задачу (например, ROLE_FIREWALL_ADMIN только для правил безопасности).
3. **Безопасность:** Минимизирует риск ошибки или злонамеренных действий. Стажёр с ролью ROLE_MONITOR физически не сможет случайно стереть конфигурацию, так как у него нет соответствующих команд.

2.3. Командование и разделение обязанностей (SoD)

Это расширение принципа RBAC. Для выполнения критически важных операций требуется согласие двух или более пользователей с разными ролями.

- **Смысл:** Предотвращение злоупотреблений и ошибок одним администратором.
- **Пример:** Чтобы изменить конфигурацию межсетевого экрана, разрешающую доступ из интернета во внутреннюю сеть, требуется:

1. **Инженер безопасности** (роль FW_CHANGE) создаёт и проверяет правило.
2. **Ответственный руководитель** (роль FW_APPROVE) должен отдельной командой утвердить и применить это правило.

- **Аналогия:** Для запуска ядерного реактора один оператор вводит код, а второй должен одновременно повернуть свой ключ. Один человек не может сделать это в одиночку.
- **Техническая реализация:** Часто требует интеграции с внешними системами управления (см. ниже) или скриптов, проверяющих multiple approvals.

Итог по назначению ролей: Мы должны уйти от ситуации, когда все администраторы используют одну учётку с полными правами. Вместо этого создаются чёткие, именованные роли, соответствующие должностным обязанностям. Это снижает риски, упрощает аудит и делает сеть более управляемой. После того как мы определили, кто имеет доступ и что он может делать, мы должны постоянно следить за тем, что происходит на устройствах — переходим к мониторингу и управлению.

3. Мониторинг и управление устройствами

Безопасность — это не разовое действие, а непрерывный процесс. Даже самая совершенная начальная настройка может устареть или быть скомпрометирована. Поэтому сетевые устройства нуждаются в постоянном наблюдении (мониторинге) и инструментах для оперативного реагирования (управления).

Фундаментальная идея: Мониторинг — это система сбора, анализа и визуализации данных о состоянии и работе сетевых устройств в реальном времени и в исторической ретроспективе, предназначенная для оперативного выявления аномалий, проблем производительности и признаков атак.

Смысл и аналогия: Представьте диспетчерский центр аэропорта. Диспетчеры не просто запустили самолёты в рейс и забыли о них. Они постоянно наблюдают за ними на радарах (мониторинг в реальном времени), видят их высоту, скорость, курс. Если самолёт отклоняется от маршрута, теряет высоту или пропадает с экрана, срабатывает сигнализация. У диспетчеров есть радиосвязь, чтобы дать команду экипажу скорректировать курс (управление). А вся история полётов записывается в журналы для последующего разбора (логи).

3.1. Что нужно мониторить?

1. **Доступность (Availability):** Живо ли устройство? Отвечает ли на запросы?
   - **Механизм:** Протокол ICMP ping, SNMP-опросы статуса.
   - **Пример:** Система мониторинга каждые 60 секунды отправляет ping на IP-адрес маршрутизатора. Если три ping подряд не получили ответа, система генерирует alert: "Маршрутизатор Core-R1 недоступен!".
2. **Нагрузка и производительность (Performance):** Не перегружено ли устройство?
   - **Параметры:** Загрузка CPU, использование оперативной памяти, заполненность таблиц (ARP, MAC-адресов), трафик на интерфейсах.
   - **Пример:** График в системе мониторинга показывает, что загрузка CPU коммутатора неожиданно выросла с 20% до 95% и держится. Это может быть признаком DDoS-атаки или сбоя в работе какого-то ПО.
3. **Конфигурация и её изменения:** Не изменилась ли конфигурация несанкционированно?
   - **Механизм:** Сравнение текущей конфигурации (running-config) с эталонной (golden config), сохранённой на безопасном сервере. Использование протокола Syslog для сбора сообщений об изменениях.
   - **Пример:** На сервер Syslog с маршрутизатора приходит сообщение: %SYS-5-CONFIG_I: Configured from console by admin on vty0 (10.0.10.5). Это нормально. Но если приходит сообщение с другого IP-адреса или в нерабочее время — это повод для тревоги.
4. **События безопасности (Security Events):**
   - **Неудачные попытки входа:** %SEC_LOGIN-4-LOGIN_FAILED: Login failed [user: hacker] [Source: 192.168.1.100].
   - **Нарушения политик доступа:** Попытки подключиться к закрытым портам (сканирование).
   - **Изменения правил безопасности:** Добавление или удаление ACL.

3.2. Протоколы и системы мониторинга

3.2.1. Syslog

Протокол для отправки текстовых сообщений о событиях с устройства на центральный сервер-коллектор.

- **Аналогия:** Это "чёрный ящик" или бортовой журнал самолёта. Все важные события (включение, выключение, ошибки, действия пользователей) записываются в хронологическом порядке.
- **Важность:** Syslog — главный источник информации для расследования инцидентов. Кто, когда и что сделал? Ответы часто лежат в логах.
- **Пример настройки на устройстве:**

```
! Направляем логи уровня "informational" и выше (0-6) на сервер Syslog
```

```
logging host 10.0.50.100
logging trap informational
```

3.2.2. SNMP (Simple Network Management Protocol)

Протокол для опроса устройств и получения структурированных данных (числовых значений) об их состоянии.

- **Аналогия:** Если Syslog — это рассказ пилота о полёте, то SNMP — это приборная доска самолёта, которая передаёт точные цифры: высота 10000 м, скорость 850 км/ч, температура двигателя 300°C.
- **Как работает:** На устройстве работает **SNMP-агент**. Система управления (NMS — Network Management System) периодически опрашивает агент, запрашивая значения конкретных переменных (OID — Object Identifier). Например, OID для загрузки CPU имеет свой уникальный номер.
- **Версии: SNMPv1/v2c** небезопасны (пароль-community string передаётся в открытом виде). **SNMPv3** поддерживает шифрование и аутентификацию — только он должен использоваться для безопасного мониторинга.
- **Пример использования:** Система Zabbix по графику опрашивает через SNMPv3 коммутатор, получает значение счётчика входящих ошибок на интерфейсе и строит график. При резком всплеске ошибок Zabbix посылает уведомление в Telegram администратору.

3.2.3. NetFlow / sFlow / IPFIX

Протоколы для сбора и анализа информации о сетевом трафике (метаданных). Они говорят не "как работает устройство", а "какой трафик через него проходит".

- **Что передают:** Источник и получатель (IP, порт), тип протокола, объём данных, время.
- **Пример безопасности:** NetFlow может показать, что рабочий компьютер (10.0.1.15) неожиданно начал отправлять гигабайты данных на внешний сервер (185.1.2.3) по порту 4444 (часто используется троянами). Это явный признак заражения.

3.3. Системы управления (NMS)

Это "диспетчерские центры" сети. Они объединяют в себе сбор данных (через SNMP, Syslog, NetFlow), их хранение, визуализацию (дашборды, карты сети, графики) и генерацию оповещений.

- **Популярные системы:** LibreNMS, Zabbix, PRTG, Nagios, коммерческие решения от крупных вендоров.
- **Ключевые функции:**

1. **Единое окно:** Видеть состояние всей сети из одной веб-консоли.
2. **Автоматическое обнаружение:** Система сама находит новые устройства в сети.
3. **Графики и тренды:** Показывают, как параметры менялись за час, день, месяц. Помогают планировать апгрейд (например, если память коммутатора постоянно заполняется на 90%).

4. **Автоматизация реакции:** Например, если интерфейс "упал", система может автоматически создать заявку в тикет-систему (Jira, ServiceNow).

Итог по мониторингу и управлению: Мы должны не только настроить устройство и раздать права, но и непрерывно "слушать" его пульс, отслеживать его "здоровье" и иметь инструменты для быстрого вмешательства. Это превращает сеть из набора "чёрных ящиков" в прозрачную, управляемую и предсказуемую среду. Однако ручное управление десятками и сотнями устройств становится неэффективным. Здесь на помощь приходят технологии автоматизации.

4. Использование функции автоматизированной настройки безопасности

Ручная настройка каждого устройства по отдельности — устаревший, медленный и подверженный ошибкам подход. Человек может опечататься, забыть применить политику на одном устройстве или потратить часы на однотипные действия. В современных сетях, особенно больших и сложных, безопасность обеспечивается не вручную, а с помощью политик и автоматизации.

Фундаментальная идея: Автоматизация настройки безопасности — это подход, при котором конфигурации устройств управляются не прямым вводом команд, а через централизованно определённые политики (желаемое состояние), которые затем автоматически, точно и согласованно применяются ко всем соответствующим устройствам с помощью программных средств.

Смысл и аналогия: Представьте, что вам нужно покрасить забор длиной в километр. Вы можете делать это вручную кисточкой — это долго, утомительно, и качество будет неровным (ручная настройка). А можете использовать краскопульт, который равномерно наносит краску по заранее заданным параметрам (автоматизация). Более того, вы создаёте единый стандарт: "Все заборы должны быть синего цвета, краска марки N, два слоя" (политика). Если через год часть забора поцарапали, система сама заметит отклонение от стандарта и автоматически подкрасит это место (исправление дрейфа конфигурации).

4.1. Концепция "Infrastructure as Code" (IaC)

Это основа современной автоматизации. Конфигурация сети описывается в виде файлов на декларативном языке (YAML, JSON, собственный DSL), которые являются "исходным кодом" инфраструктуры.

- **Декларативный подход:** Вы описываете **ЧТО** хотите получить (например, "на интерфейсах Eth0/1-24 должен быть включен порт безопасности"), а не **КАК** это сделать (последовательность команд `interface range ...`, `switchport port-security`).

- **Пример (упрощённо, в стиле Ansible):**

```yaml
- hosts: access_switches
  tasks:
    - name: Enable port-security on access ports
      cisco.ios.ios_config:
        lines:
          - switchport port-security
          - switchport port-security maximum 2
          - switchport port-security violation restrict
        parents: "interface range Ethernet0/1-24"
```

Этот файл, называемый плейбуком, можно применить ко всем коммутаторам доступа в сети. Система (Ansible) сама подключится к каждому устройству по SSH и выполнит нужные команды.

Преимущества IaC для безопасности:

1. **Повторяемость и согласованность:** На всех устройствах конфигурация будет идентичной, без "человеческого фактора".
2. **Версионность:** Файлы конфигурации хранятся в системе контроля версий (Git). Можно увидеть, кто, когда и что изменил, и при необходимости откатиться к предыдущей, рабочей версии.
3. **Тестирование:** Конфигурацию можно протестировать на стенде перед внедрением в реальную сеть.
4. **Скорость развёртывания:** Новая политика безопасности (например, блокировка уязвимого порта) применяется ко всей сети за минуты, а не за дни.

4.2. Инструменты автоматизации

4.2.1. Ansible

Популярный, простой в освоении инструмент, использующий агент-лес архитектуру (не требует установки агентов на устройства). Работает по SSH. Идеален для автоматизации конфигурации, сбора информации, разовых задач.

- **Аналогия:** Универсальный пульт дистанционного управления, который может одновременно отправить одну и ту же команду на все ваши устройства (телевизор, кондиционер, свет). Вы пишете сценарий (плейбук) — "включи свет на кухне и установи в гостиной температуру 22°C" — и Ansible выполняет его.

4.2.2. Терраформ (Terraform)

Инструмент для оркестрации "облачной" и программно-определяемой инфраструктуры.

Может управлять не только сетевыми устройствами, но и виртуальными машинами, облачными сервисами.

- **Аналогия:** Если Ansible — это пульт управления, то Terraform — это архитектор и прораб, который по чертежам (конфигурационным файлам) строит целое здание (инфраструктуру) с нуля, обеспечивая правильные связи между всеми компонентами.

4.3. Функции автоматизированной настройки безопасности на самих устройствах

Современные устройства также имеют встроенные средства для автоматизации.

4.3.1. AutoSecure (в устройствах Cisco)

Интерактивный скрипт, который "задаёт вопросы" администратору и затем автоматически применяет набор рекомендованных практик безопасности: отключает неиспользуемые службы, настраивает безопасный доступ (SSH, ACL), включает защиту от спуфинга.

- **Пример:** Команда `auto secure` в режиме enable. Это быстрый способ привести новое устройство к базовому безопасному состоянию.

4.3.2. Конфигурационные шаблоны (Configuration Templates)

В системах управления (вроде Cisco DNA Center) можно создать шаблон конфигурации с переменными.

- **Как работает:** Создаётся шаблон для коммутатора доступа. В нём есть переменные, например, `{ip_address}`, `{default_gateway}`, `{vlan_number}`. При добавлении нового коммутатора в систему нужно лишь заполнить эти переменные для конкретного устройства, и система сама сгенерирует и применит полную конфигурацию.
- **Преимущество:** Гарантирует, что все новые устройства попадают в сеть уже безопасно настроенными, в соответствии с корпоративным стандартом.

4.3.3. Zero-Touch Provisioning (ZTP)

Технология, позволяющая новому устройству автоматически получить базовую конфигурацию и адрес управления при первом включении в сети, без вмешательства инженера.

- **Процесс:** Устройство загружается, запрашивает по DHCP не только IP-адрес, но и адрес скрипта или сервера автоматизации. Затем оно подключается к этому серверу, который идентифицирует устройство (по серийному номеру) и передаёт ему предназначенную для него конфигурацию.
- **Смысл для безопасности:** Исключает этап незащищённой ручной начальной настройки через консоль. Устройство с самого первого момента работает в безопасном режиме под управлением централизованной системы.

4.4. Исправление дрейфа конфигурации (Configuration Drift Remediation)

Дрейф конфигурации — это ситуация, когда реальная конфигурация устройства отклоняется от эталонной, утверждённой политикой. Это может произойти из-за экстренного ручного исправления, ошибки или атаки.

- **Как с этим борется автоматизация:** Системы автоматизации (например, Ansible с определённой периодичностью) могут выполнять **проверку на соответствие**. Они считывают текущую конфигурацию с устройства, сравнивают её с эталонной из репозитория и либо сообщают о различиях, либо **автоматически приводят устройство в соответствие с эталоном**.
- **Пример:** Кто-то вручную открыл на маршрутизаторе порт Telnet для "быстрого решения проблемы". Ночью задача Ansible, запускаемая по расписанию, обнаруживает это отклонение от политики "только SSH" и автоматически отключает Telnet, возвращая устройство в безопасное состояние.

Итог по автоматизации: Автоматизация — это не просто "удобно". Это необходимый этап эволюции для обеспечения безопасности в масштабе. Она позволяет внедрять безопасность быстро, единообразно, документировано и с возможностью лёгкого контроля и исправления. Без неё управление безопасностью в большой сети превращается в сизифов труд.

Заключение

Безопасность сетевых устройств — это многослойная, взаимосвязанная конструкция, подобная матрёшке или кольцам обороны крепости.

1. **Безопасный доступ (Внешний периметр)** — это фундамент. Мы тщательно проверяем личность каждого, кто стучится в наши ворота (аутентификация), разрешаем ему говорить только по секретному каналу (шифрование) и пускаем только с доверенных направлений (ACL).
2. **Административные роли (Внутренние правила)** — попав внутрь, каждый получает ровно столько власти, сколько нужно для его работы, и ни капли больше (принцип наименьших привилегий, RBAC). Это минимизирует ущерб от ошибок или злого умысла.
3. **Мониторинг и управление (Система наблюдения и реагирования)** — мы никогда не оставляем крепость без присмотра. Мы постоянно слушаем её "пульс" (SNMP), читаем её "дневники" (Syslog) и анализируем "гостей" (NetFlow), чтобы мгновенно заметить любую угрозу и отреагировать на неё.
4. **Автоматизация (Центральный штаб и роботы-исполнители)** — мы уходим от кустарного ручного труда. Мы создаём единые, проверенные стандарты безопасности (политики, шаблоны, код) и поручаем их точное, безошибочное исполнение машинам. Это позволяет нам управлять безопасностью целой сети как единым целым, быстро, согласованно и надёжно.

Только комплексное применение всех этих принципов превращает набор сетевого оборудования в защищённую, устойчивую и управляемую инфраструктуру, способную противостоять современным угрозам. Изучение и практическое освоение этих инструментов является ключевой задачей для будущего специалиста по сетевому и системному администрированию.

Контрольные вопросы к ТЕМЕ-21

1. Что такое аутентификация в контексте безопасного доступа к сетевым устройствам?
2. Назовите два фактора, используемые в двухфакторной аутентификации (2FA).
3. Почему протокол SSH является более безопасным для удалённого управления, чем Telnet?
4. В чём заключается основная идея принципа наименьших привилегий (Principle of Least Privilege)?
5. Какие основные функции выполняет система AAA (произносится как "три А")?
6. Для чего используется протокол Syslog на сетевых устройствах?
7. Какую основную задачу решает ролевая модель доступа RBAC?
8. Что такое "дрейф конфигурации" (configuration drift) и почему это плохо для безопасности?
9. Какой протокол (SNMP или NetFlow) используется для сбора информации о том, какой именно трафик проходит через устройство?
10. Назовите ключевое преимущество использования централизованного сервера (RADIUS/TACACS+) для аутентификации по сравнению с локальными учётными записями на каждом устройстве.
11. Что означает подход "Infrastructure as Code" (IaC) при настройке безопасности?
12. Как с помощью списков контроля доступа (ACL) можно повысить безопасность доступа к устройству?

Практические задания к ТЕМЕ-21

Задание 1. Анализ ошибки стажёра

В небольшой компании «Ромашка» в г. Тюмени сетевой администратор ушёл в отпуск, оставив стажёру задачу: подключиться к маршрутизатору в филиале и проверить его статус. Стажёр с рабочего компьютера (IP-адрес 192.168.5.10) использовал программу PuTTY и успешно ввёл логин и пароль. Однако через несколько дней выяснилось, что пароль учётной записи администратора стал известен неизвестным лицам, которые попытались изменить настройки.

Изучив журналы, вы обнаружили, что стажёр использовал тип подключения Telnet, а пароль был «qwerty123».

Объясните, какие **три ключевые ошибки** с точки зрения безопасного доступа были допущены в этой ситуации. Ваш ответ должен опираться на теоретический материал.

Задание 2. Разработка ролевой политики для колл-центра

В компании «МегаФон-Сибирь» есть сеть колл-центра в Новосибирске, состоящая из 50 коммутаторов. К управлению сетью имеют отношение три категории сотрудников:

1. **Сотрудники техподдержки 1-й линии:** Их задача — оперативно проверять, «живы» ли порты сотрудников колл-центра, и перезагружать их по телефонной заявке.
2. **Старшие сетевые инженеры:** Они настраивают VLAN'ы, основные параметры безопасности и следят за стабильностью.
3. **Главный архитектор сети:** Полный контроль и ответственность за всю инфраструктуру.

Предложите **названия ролей (RBAC)** для каждой категории и кратко опишите, какие типовые действия (команды) должны быть разрешены каждой роли, исходя из принципа наименьших привилегий.

Задание 3. Выбор протокола для мониторинга

Вы — администратор в школе №207 г. Казани. Вам поручили организовать мониторинг загрузки центрального коммутатора. Вам нужно, чтобы система Zabbix могла автоматически снимать показатели использования процессора (CPU) и памяти, а также получать важные сообщения о состоянии устройства (например, о перезагрузке или ошибках на портах).
Вам предлагают два варианта настройки: 1) использовать только Syslog, 2) использовать только SNMPv2c с community-строкой «public».
Проанализируйте оба варианта. Объясните, почему **ни один из этих вариантов не является удовлетворительным** с точки зрения безопасности и полноты мониторинга? Какой подход был бы правильным?

Задание 4. Планирование ответа на инцидент

В ИТ-отделе завода «Уралмаш» в Екатеринбурге сработала система мониторинга. Она показала, что с IP-адреса (195.144.1.33), не входящего в доверенную сеть предприятия, происходит множество попыток подключиться к интерфейсу управления маршрутизатором ядра сети по порту 22 (SSH). Попытки неуспешны.
Опишите **последовательность ваших действий** как администратора, основываясь на изученных принципах. Какие разделы теоретического материала (безопасный доступ, мониторинг, роли) вы задействуете на каждом шаге для анализа и нейтрализации этой потенциальной угрозы?

Задание 5. Аргументация для внедрения автоматизации

Вы — молодой специалист, которого взяли в ИТ-отдел «Сбербанка» в г. Калининграде. Вы заметили, что настройка новых точек доступа Wi-Fi для офисов выполняется вручную: инженер подключается к каждому устройству по консоли и вводит один и тот же набор из 20 команд. Вы хотите предложить руководителю внедрить инструмент автоматизации (например, Ansible).
Сформулируйте **три конкретных аргумента** из области безопасности и эффективности, которые вы приведёте на совещании, чтобы обосновать необходимость этого перехода. Ваши аргументы должны быть связаны с идеями, изложенными в теоретическом материале.

ТЕМА-22. Авторизация, аутентификация и учет доступа (AAA)

Введение

В современном мире, где информация является ключевым активом, а сетевые технологии пронизывают все сферы деятельности, критически важным становится контроль за тем, кто, что и как может делать в информационной системе. Представьте себе офисное здание. Не каждый может просто войти в него. Сначала вам нужно предъявить пропуск (доказать, что вы имеете право войти), затем, возможно, пройти к определенной двери, которая открывается только вашим ключом или отпечатком пальца, и, наконец, ваши перемещения внутри здания могут фиксироваться журналами охраны. Эта многоуровневая система безопасности в цифровом мире называется **AAA** (Authentication, Authorization, Accounting – Аутентификация, Авторизация и Учет).

Целью данного материала является формирование четкого понимания трех фундаментальных столпов безопасности доступа: **Аутентификации** (кто ты?), **Авторизации** (что тебе разрешено?) и **Учета** (что ты делал?). Мы разберем, как эти принципы реализуются в простых локальных системах и в более сложных централизованных решениях на основе сервера.

1. Свойства AAA: Три кита информационной безопасности

Концепция AAA – это не просто набор функций, а целостная философия управления доступом. Каждый компонент решает свою задачу, но только вместе они создают надежную и контролируемую систему.

1.1. Аутентификация (Authentication) – «Кто ты?»

Базовая идея: Аутентификация – это процесс проверки и подтверждения заявленной пользователем (устройством, системой) идентичности (личности). Это ответ на вопрос «Ты действительно тот, за кого себя выдаешь?».

Пояснение смысла: Представьте, что вы звоните в банк и говорите: «Здравствуйте, это Иванов Иван Иванович, хочу узнать баланс счета». Первое, что сделает оператор, – попросит вас подтвердить личность. Это и есть аутентификация. В цифровом мире пользователь

заявляет о себе с помощью **идентификатора** (логина, имени пользователя), но этого недостаточно, так как логин может знать кто угодно. Поэтому необходимо предъявить **секрет**, известный только настоящему пользователю и системе.

Формулировка: Аутентификация – это процедура проверки подлинности субъекта (пользователя, устройства, процесса) путем сравнения предъявленных им учетных данных с эталонными данными, хранящимися в системе.

Примеры и аналогии:

1. **Пароль:** Самый распространенный пример. Логин – это ваше имя, пароль – ваш секретный ключ. Как ключ от квартиры: если ключ подошел, дверь открывается (система верит, что вы – хозяин).
2. **Биометрия:** Отпечаток пальца, сканирование сетчатки глаза, распознавание лица в смартфоне. Система сравнивает ваш биометрический шаблон с эталонным.
3. **Смарт-карта или токен:** Физический предмет, который нужно вставить в считыватель или использовать для генерации одноразового кода. Аналогично пропуску на предприятие или банковской карте с чипом.
4. **Многофакторная аутентификация (MFA):** Комбинация двух или более факторов из разных категорий: «что-то, что вы знаете» (пароль), «что-то, что у вас есть» (смартфон с кодом), «что-то, что вы есть» (отпечаток). Как в банке: чтобы совершить крупный платеж, нужен и пароль от интернет-банка (знание), и код из СМС (владение устройством).

Итог: Успешная аутентификация не дает пользователю никаких прав, кроме одного – система теперь **знает**, с кем имеет дело.

1.2. Авторизация (Authorization) – «Что тебе разрешено?»

Базовая идея: Авторизация – это процесс определения прав и разрешений, которыми обладает уже аутентифицированный пользователь, на доступ к определенным ресурсам и выполнение конкретных операций. Это ответ на вопрос «Что ты можешь делать здесь?».

Пояснение смысла: Вы успешно прошли охрану в офисном здании (аутентифицировались). Но это не значит, что вы можете зайти в любой кабинет. В кабинет финансового отдела пропустят только бухгалтеров, в серверную – только системных администраторов, а в архив генерального директора – только его самого. Эти правила доступа и есть авторизация.

Формулировка: Авторизация – это процесс предоставления аутентифицированному субъекту определенных прав на доступ к ресурсам и выполнение операций в системе на основе предварительно заданных политик безопасности.

Примеры и аналогии:

1. **Файловая система NTFS/APFS:** После входа в Windows или macOS (аутентификация) вы можете открыть одни папки, но не можете даже просмотреть другие. Вы можете

редактировать одни документы, но только читать другие. Эти права (Read, Write, Execute, Modify) назначаются пользователям и группам.

2. **Сетевые устройства (маршрутизатор, коммутатор):** Пользователь «admin» может настраивать все интерфейсы и правила, а пользователь «operator» – только просматривать состояние портов.
3. **Веб-сайт:** Гость (неаутентифицированный пользователь) видит только главную страницу. Зарегистрированный пользователь видит свой профиль и может комментировать. Модератор может редактировать комментарии других. Администратор может все.
4. **Офисный блокнот:** Вы можете прочитать любую страницу (чтение), писать на чистой странице (запись), но не можете вырвать страницу, написанную кем-то другим (удаление). Если блокнот общий, на его обложке может быть список, кто что может с ним делать.

Итог: Авторизация всегда следует **после** успешной аутентификации и определяет **границы дозволенного** для известного системе субъекта.

1.3. Учет (Accounting) – «Что ты делал?»

Базовая идея: Учет (или аудит) – это процесс регистрации, учета и анализа действий аутентифицированного пользователя, а также потребленных им сетевых ресурсов. Это ответ на вопросы «Когда, что и как долго ты это делал?».

Пояснение смысла: Вы не только прошли в здание и в свой кабинет, но ваши перемещения фиксируются камерами, а вход и выход – электронной системой. Это позволяет установить, кто, когда и где находился. В IT учет позволяет анализировать действия пользователей, расследовать инциденты, биллинг (списание средств за использованный трафик или время) и планировать нагрузку на ресурсы.

Формулировка: Учет (Accounting) – это процесс сбора и регистрации информации о деятельности и потреблении ресурсов аутентифицированными пользователями в течение определенного периода времени для целей анализа, биллинга, аудита и обеспечения безопасности.

Примеры и аналогии:

1. **Журналы событий (Logs):** Любой сервер, сетевое устройство или программа ведет логи. Запись «Пользователь admin вошел в систему с IP-адреса 192.168.1.5 в 14:30, изменил конфигурацию интерфейса Eth0 в 14:35, вышел в 15:00» – это типичный пример учета.
2. **Биллинг провайдера:** Интернет-провайдер учитывает объем трафика, который вы передали и получили, или время вашего соединения, чтобы выставить счет. Аутентификация (логин/пароль на PPPoE/L2TP) открывает доступ, авторизация определяет вашу скорость (тариф), а учет – считает, сколько гигабайт вы использовали.
3. **Турникет в метро:** Вы приложили карту (аутентификация), система проверила, что на карте есть поездки (авторизация), списала одну поездку и зафиксировала время и станцию входа (учет).

4. **Детализация звонков у мобильного оператора:** В личном кабинете вы можете увидеть таблицу: кому, когда и как долго вы звонили. Это результат работы системы учета.

   Итог: Учет создает «цифровой след» активности, обеспечивая подотчетность пользователей и предоставляя данные для анализа и оптимизации работы системы.

   Взаимосвязь ААА: Эти три процесса строго последовательны и неразрывны.

1. **Сначала** система должна понять, **кто вы** (Аутентификация).
2. **Затем,** зная, кто вы, система решает, **что вам можно** (Авторизация).
3. **И параллельно,** предоставляя вам доступ, система **записывает, что и как вы делаете** (Учет).

   Попытка авторизовать или учесть действия неаутентифицированного пользователя бессмысленна – система не знает, кому назначать права и чьи действия записывать.

2. Локальная ААА аутентификация

Базовая идея: Локальная ААА – это модель, в которой все три компонента (данные для проверки подлинности, правила доступа и журналы событий) хранятся и обрабатываются непосредственно на том самом устройстве (коммутаторе, маршрутизаторе, сервере, компьютере), к которому пользователь пытается получить доступ.

Пояснение смысла: Это похоже на дом с замком, ключ от которого хранится только внутри этого дома. Чтобы открыть дверь, вам нужен именно тот ключ, который подходит к этому конкретному замку. Хозяин (администратор) отдельно для каждого дома (устройства) создает набор ключей (учетные записи) и список, кто в какую комнату может зайти (правила авторизации). Все это управляется независимо.

Формулировка: Локальная аутентификация и авторизация – это метод контроля доступа, при котором учетные данные пользователей (например, логины и пароли), а также политики прав доступа хранятся в собственной, внутренней базе данных устройства, и процесс проверки выполняется этим же устройством без обращения к внешним службам.

2.1. Принцип работы и примеры

- **На сетевом устройстве (Cisco IOS):** Администратор вручную создает пользователей и пароли непосредственно в конфигурации маршрутизатора или коммутатора с помощью команд username ... password .... Когда пользователь пытается подключиться по SSH или через консоль, устройство само сверяет введенные данные со своим внутренним списком.

- **Аналогия:** У сторожа на проходной есть свой бумажный журнал с фамилиями и печатями сотрудников, которым разрешен проход. Он сам проверяет пропуск и сверяется со своим списком.
- **На персональном компьютере (Windows, Linux):** Учетные записи пользователей хранятся локально на жестком диске компьютера (файл SAM в Windows, /etc/passwd и /etc/shadow в Linux). Вход в систему проверяется самой операционной системой этого компьютера.
- **На изолированном веб-приложении:** Простая программа может иметь свою собственную таблицу в базе данных с логинами и хэшами паролей.

2.2. Преимущества локальной AAA

1. **Простота и скорость:** Не нужно настраивать сложную инфраструктуру серверов. Проверка происходит мгновенно, так как данные «под рукой».
2. **Независимость от сети:** Устройство может аутентифицировать пользователей, даже если у него нет связи с другими серверами или интернетом. Это критично для аварийного доступа (консольный порт).
3. **Простота настройки для малых систем:** Для одного-двух устройств или изолированной системы это самый быстрый и логичный способ.

2.3. Недостатки и ограничения локальной AAA

1. **Проблема масштабируемости:** Представьте, что в компании 100 сетевых устройств. Администратору придется вручную создавать и обновлять одни и те же учетные записи на каждом из 100 устройств. При смене пароля – вносить изменения 100 раз. Это неэффективно и чревато ошибками.
2. **Слабый централизованный контроль:** Невозможно единообразно применить политику безопасности ко всем устройствам. На одном устройстве пароль может быть простым, на другом – сложным.
3. **Проблемы с учетом и аудитом:** Журналы событий разбросаны по всем устройствам. Чтобы понять, что делал пользователь в сети, нужно собирать логи с десятков источников.
4. **Уязвимость к отказам:** Проблема здесь не в отказе службы, а в сложности управления. Потеря пароля администратора к конкретному устройству может потребовать его полного сброса.

Вывод: Локальная AAA идеальна для **аварийного доступа** (учетная запись для консоли всегда должна быть локальной на случай, если сеть недоступна), для **изолированных устройств** или **очень маленьких сетей**. Однако для корпоративной среды с множеством устройств и пользователей она не подходит.

3. Server-based AAA (AAA на основе сервера)

Базовая идея: Server-based AAA – это модель, в которой функции аутентификации, авторизации и учета вынесены на выделенный центральный сервер (или группу серверов). Сетевое устройство (клиент AAA), к которому подключается пользователь, не проверяет учетные данные сам, а пересылает их на специализированный сервер для проверки и получения инструкций.

Пояснение смысла: Это как система контроля доступа в большом бизнес-центре. На входе стоит турникет, но он **сам по себе не принимает решения**. Вы прикладываете пропуск, турникет отправляет номер этого пропуска на центральный сервер охраны. Сервер в своей единой базе данных проверяет: действителен ли пропуск, кому он принадлежит, имеет ли этот человек право войти в эту конкретную дверь в данное время суток. Затем сервер отправляет команду турникету: «Открыть» или «Запретить». Все журналы проходов хранятся централизованно на сервере.

Формулировка: Server-based AAA – это архитектура контроля доступа, при которой сетевые устройства (клиенты) делегируют функции аутентификации, авторизации и учета выделенному внешнему серверу, использующему единую централизованную базу данных пользователей и политик.

3.1. Ключевые компоненты архитектуры

1. **Клиент (Supplicant) / Пользователь:** Тот, кто запрашивает доступ (пользователь, компьютер).
2. **NAS (Network Access Server) / AAA-клиент:** Сетевое устройство (маршрутизатор, коммутатор, точка доступа Wi-Fi, VPN-шлюз), которое контролирует точку доступа. Оно принимает запрос от пользователя и переправляет его на AAA-сервер. **NAS – это «посредник»**.
3. **AAA-сервер:** Центральный сервер, который хранит единую базу данных учетных записей, политики доступа и занимается учетом. Он принимает запрос от NAS, обрабатывает его и возвращает ответ (разрешить/запретить + параметры авторизации).
4. **Протокол взаимодействия:** Язык, на котором говорят NAS и AAA-сервер. Самые распространенные – **RADIUS** и **TACACS+**.

3.2. Детальный пример работы (сценарий доступа сотрудника к корпоративной Wi-Fi сети)

1. **Запрос:** Сотрудник на ноутбуке выбирает корпоративную сеть Wi-Fi «CORP_OFFICE» и вводит свой логин (iivanov) и пароль.
2. **Передача запроса:** Точка доступа Wi-Fi (она же NAS) получает эти данные. Она не проверяет их сама. Вместо этого она упаковывает их в специальный **запрос на аутентификацию** протокола RADIUS и отправляет этот пакет на предварительно настроенный AAA-сервер (например, на сервер под управлением FreeRADIUS).

3. **Обработка на сервере:** AAA-сервер получает запрос. Он извлекает логин iivanov и хэш пароля. Затем обращается к своей центральной базе данных (это может быть Active Directory, LD-каталог или локальная база).
4. **Аутентификация:** Сервер сравнивает полученный хэш пароля с эталонным хэшем из базы. Если они совпадают – аутентификация **успешна**. Сервер теперь знает, что это действительно Иванов.
5. **Авторизация (определение прав):** Сервер проверяет, какие политики назначены пользователю iivanov или группе «Сотрудники». В политике может быть написано: «После успешной аутентификации в Wi-Fi назначить VLAN 10 (гостевая сеть), ограничить скорость до 10 Мбит/с, запретить доступ к сети серверов». Сервер формирует **пакет ответа**, куда включает не просто «ОК», а все эти параметры (VLAN, скорость, фильтры доступа).
6. **Исполнение:** Точка доступа получает от сервера пакет «Access-Accept» с параметрами. Она выполняет команды сервера: переводит порт ноутбука Иванова в VLAN 10, применяет ограничение полосы пропускания и правила фильтрации. **Доступ предоставлен согласно централизованной политике.**
7. **Учет (Accounting Start):** Точка доступа сразу отправляет на AAA-сервер сообщение «Сессия для iivanov началась», а когда сотрудник отключается – «Сессия для iivanov завершилась, переданный трафик: 150 МБ».

3.3. Протоколы Server-based AAA: RADIUS vs TACACS+

Это два основных «языка» общения между NAS и AAA-сервером.

Характеристика	**RADIUS (Remote Authentication Dial-In User Service)**	**TACACS+ (Terminal Access Controller Access-Control System Plus)**
Транспортный протокол	**UDP** (порты 1812/1813). Быстрее, но менее надежен.	**TCP** (порт 49). Надежнее, с установкой соединения.
Шифрование	Шифрует **только пароль** в запросе на доступ. Остальная информация (логин, параметры авторизации) передается открыто.	Шифрует **весь** пакет передачи данных между клиентом и сервером. Более безопасен.
Модульность AAA	Объединяет аутентификацию и авторизацию в один процесс.	Четко **разделяет** процессы AAA. Можно использовать разные серверы для каждой функции.

Характеристика	RADIUS (Remote Authentication Dial-In User Service)	TACACS+ (Terminal Access Controller Access-Control System Plus)
Гибкость авторизации	Хорошо подходит для контроля доступа к сети (VLAN, ACL, QoS).	Обеспечивает более детальный контроль на уровне **команд** (например, настройка устройств).
Типичное применение	**Контроль доступа пользователей:** Wi-Fi, VPN, порталы авторизации, коммутируемый доступ.	**Контроль доступа администраторов:** Управление сетевым оборудованием (коммутаторы, маршрутизаторы).

Аналогия: Представьте, что вам нужно получить доступ к чему-либо.

- **RADIUS** – как пропуск на территорию завода. Вы предъявляете его на КПП (пароль шифруется), вас впускают и сразу говорят: «Идите в цех №3, но не заходите в красные зоны». Все в одном действии.
- **TACACS+** – как доступ в диспетчерскую с множеством переключателей. Сначала вы доказываете личность охраннику (аутентификация). Затем начальник смены сверяется со списком и говорит: «Вам разрешено пользоваться только этими тремя переключателями» (авторизация). И все ваши манипуляции с ними записываются в журнал (учет). Процессы разделены.

3.4. Преимущества Server-based AAA

1. **Масштабируемость:** Добавление нового пользователя или устройства требует изменений только в центральной базе данных, а не на каждом устройстве.
2. **Централизованное управление:** Единая точка для настройки политик безопасности, смены паролей, блокировки учетных записей.
3. **Усиленная безопасность:** Использование надежных протоколов (TACACS+), возможность легкого внедрения MFA на уровне сервера.
4. **Централизованный учет и аудит:** Все журналы доступа со всех устройств стекаются в одно место, что упрощает мониторинг, анализ и расследование инцидентов.
5. **Гибкость и избыточность:** Можно настроить несколько AAA-серверов для отказоустойчивости.

3.5. Практическое применение Server-based AAA

- **Управление доступом администраторов:** Все инженеры входят на коммутаторы и маршрутизаторы не с локальными паролями, а используя свои учетные записи AD/LDAP

через TACACS+. Сервер решает, кому из них разрешено выполнять команды configure terminal, а кому – только show.

- **Корпоративные Wi-Fi и VPN:** Сотрудники аутентифицируются в беспроводной сети или при подключении из дома с помощью единой учетной записи домена через RADIUS.
- **Гостевой доступ:** Создание временных учетных записей для посетителей с ограничением по времени и доступным ресурсам.

Вывод: Server-based AAA – это **стандарт де-факто** для построения безопасной и управляемой сетевой инфраструктуры в организациях любого среднего и крупного размера. Он заменяет хаос разрозненных локальных паролей на порядок, контроль и безопасность централизованной системы.

Заключение

Концепция AAA представляет собой системный и многоуровневый подход к информационной безопасности. Ее понимание является фундаментальным для любого специалиста в области сетевого и системного администрирования.

- **Аутентификация** устанавливает личность, **авторизация** определяет границы, а **учет** обеспечивает подотчетность.
- **Локальная AAA** проста и надежна в изоляции, но совершенно непригодна для управления крупными распределенными системами из-за проблем с масштабированием и контролем.
- **Server-based AAA** решает эти проблемы, вынося логику принятия решений на центральный сервер. Использование протоколов **RADIUS** (для доступа пользователей) и **TACACS+** (для доступа администраторов) позволяет построить гибкую, безопасную и легко управляемую инфраструктуру.

На практике в современной сети эти подходы часто комбинируются: для аварийного консольного доступа к критическому оборудованию **всегда** настраивается надежная локальная учетная запись (последняя линия обороны), в то время как весь штатный доступ (через SSH, Wi-Fi, VPN) осуществляется через центральный AAA-сервер. Это создает отказоустойчивую и в то же время удобную в администрировании среду.

Таким образом, освоение принципов AAA – это не просто изучение аббревиатуры, а приобретение ключевого навыка проектирования и поддержки безопасного доступа в любой информационной системе.

Контрольные вопросы к ТЕМЕ-22

1. Какие три основных процесса объединяет в себе концепция AAA? Дайте их краткие определения.

2. В какой последовательности всегда выполняются процессы AAA? Почему эта последовательность является обязательной?
3. Приведите пример из обычной жизни, который иллюстрирует разницу между аутентификацией и авторизацией.
4. Какой компонент AAA отвечает за сбор данных о действиях пользователя и потреблении ресурсов для целей анализа и биллинга?
5. Что такое локальная AAA аутентификация? Назовите основной ее недостаток в корпоративной среде.
6. Где хранятся учетные данные пользователей при использовании локальной AAA на сетевом устройстве?
7. Что такое Server-based AAA? Какую основную проблему масштабируемости она решает?
8. Назовите три ключевых компонента архитектуры Server-based AAA.
9. Какую роль играет устройство NAS (Network Access Server) в модели Server-based AAA?
10. Назовите два основных протокола, используемых в Server-based AAA. Какой из них чаще применяется для контроля доступа пользователей к сети (Wi-Fi, VPN)?
11. Какой протокол — RADIUS или TACACS+ — разделяет процессы аутентификации, авторизации и учета и шифрует весь пакет передачи данных?
12. Какой метод AAA (локальный или на основе сервера) рекомендуется использовать для организации аварийного доступа к консоли сетевого устройства и почему?

Практические задания к ТЕМЕ-22

Задание 1. Анализ инцидента безопасности в небольшом филиале

Ситуация: В небольшом региональном филиале компании «Ромашка» в г. Тула произошел инцидент. Бывший сотрудник отдела продаж, уволенный две недели назад, смог удаленно подключиться к корпоративной Wi-Fi сети и скопировать коммерческое предложение с файлового сервера. Расследование показало, что для доступа к Wi-Fi использовались его старые логин и пароль. На всех точках доступа в филиале была настроена локальная AAA.

- **Вопрос:** Какой из трех процессов AAA (аутентификация, авторизация, учет) явно не был задействован или сконфигурирован должным образом в этой ситуации? Объясните, какая именно процедура не была выполнена и как это позволило произойти инциденту.

Задание 2. Выбор архитектуры для растущей компании

Ситуация: Компания «Вектор» из Перми, предоставляющая IT-аутсорсинг, стремительно росла. За два года количество инженеров выросло с 3 до 15, а количество обслуживаемых клиентских сетевых устройств (коммутаторов, маршрутизаторов) — с 20 до более 200. Старший администратор жалуется, что он тратит до 30% рабочего времени только на то, чтобы добавить или сменить пароли для доступа к оборудованию новых инженеров на всех этих устройствах, и часто что-то упускает.

- **Вопрос:** Какую архитектуру AAA вы порекомендуете внедрить компании «Вектор» для решения этой проблемы? Кратко опишите, как будет работать процесс предоставления

доступа новому инженеру в этой новой архитектуре, и какой протокол (RADIUS или TACACS+) для этого подойдет лучше и почему.

Задание 3. Планирование системы контроля для стройплощадки

Ситуация: Крупная строительная компания «УралСтрой» в г. Екатеринбург начинает новый проект. На временной площадке развернута локальная сеть с точкой доступа Wi-Fi для прорабов, геодезистов и бухгалтерии. Необходимо обеспечить доступ в сеть и интернет только для сотрудников компании, при этом для прорабов должен быть открыт доступ к внутреннему серверу с чертежами, а для всех остальных — только доступ в интернет. Связи с центральным офисом на время монтажа нет.

- **Вопрос:** Исходя из ограничений, какую модель AAA (локальную или на основе сервера) возможно реализовать в данной ситуации? Предложите, как можно организовать разный уровень доступа (к серверу чертежей и только в интернет) в рамках выбранной вами модели, используя ее базовые принципы.

Задание 4. Определение типа атаки по логам

Ситуация: Системный администратор колл-центра в Казани при анализе логов обнаружил подозрительную активность. Было зафиксировано множество попыток входа на VPN-шлюз с разных логинов (сотрудников), но с использованием небольшого набора часто встречающихся паролей (например, qwerty123, password, 1q2w3e). Все попытки были заблокированы.

- **Вопрос:** На преодоление какого именно из процессов AAA была направлена эта атака? Дайте название такому типу атак. Объясните свой ответ, опираясь на определения из теоретического материала.

Задание 5. Рекомендации по расследованию

Ситуация: В IT-отделе банка «Восточный» в Хабаровске заподозрили, что один из администраторов в нерабочее время вносил несанкционированные изменения в конфигурацию межсетевого экрана. Все администраторы используют для доступа к оборудованию единые учетные записи домена. Руководитель отдела поручил вам выяснить, кто именно это мог сделать.

- **Вопрос:** Какой компонент AAA должен быть обязательно корректно настроен в данной инфраструктуре, чтобы можно было однозначно идентифицировать виновного? Какие конкретные данные из журналов этого процесса потребуются для расследования?

ТЕМА-23. Реализация технологий брандмауэра

Введение

Безопасность компьютерных сетей сегодня является одним из краеугольных камней информационных технологий. Представьте себе средневековый замок. У него есть высокие стены, ров с водой, подъёмный мост и стража у ворот. Эти элементы защищают жителей замка от внешних угроз, контролируют, кто может войти или выйти, и следят за тем, что происходит на границах владений. В мире сетевых технологий роль таких защитных сооружений выполняют брандмауэры (firewalls). Брандмауэр — это программный или аппаратный барьер, который стоит между вашей внутренней сетью (например, сетью колледжа или офиса) и внешними сетями, чаще всего Интернетом. Его основная задача — анализировать весь сетевой трафик, проходящий через него, и на основе заранее определённых правил решать: пропустить этот трафик или заблокировать.

В рамках данной темы мы рассмотрим три фундаментальных технологии, которые лежат в основе построения современных защитных рубежей: **ACL (Access Control Lists — списки контроля доступа)**, **технологию брандмауэра** как целостную систему, **контекстный контроль доступа (CBAC — Context-Based Access Control)** и подход к организации защиты на основе **политик и зон безопасности**.

Чтобы понять логику изучения, проведём аналогию с эволюцией охраны объекта. Сначала была просто стена со списком разрешённых и запрещённых лиц у ворот (ACL). Потом охрану научили не только сверяться со списком, но и проверять, соответствует ли цель визита заявленной, и следить за тем, что гость делает внутри (брандмауэр с CBAC). Наконец, весь объект разделили на зоны с разным уровнем безопасности: закрытая зона для секретных разработок, общая зона для сотрудников и гостевой периметр (политики на основе зон). Это и есть путь от простых фильтров к интеллектуальным системам защиты.

Глава 1. Списки контроля доступа (ACL) — фундаментальный кирпич защиты

1.1 Суть и назначение ACL

Список контроля доступа (ACL) — это, в самой простой форме, упорядоченный набор правил (инструкций), которые говорят сетевому устройству (маршрутизатору или коммутатору): какой трафик **разрешить (permit)**, а какой **запретить (deny)**.

Ключевая идея: ACL работает как примитивный сторож на проходной. У него в руках есть список (часто бумажный), в котором построчно написано: «Пропустить Иванова», «Не пускать Сидорова», «Всех остальных не пускать». Он не анализирует, *зачем* пришёл человек, что у него в сумке или куда он идёт дальше. Он просто сравнивает входящего человека (точнее, его пропуск или фамилию) с записями в списке, сверяясь сверху вниз. Первое совпадение определяет судьбу посетителя.

Пример из жизни: Представьте вход в общежитие вашего колледжа. На вахте лежит список:

1. Студенты 1-го корпуса — проход разрешён.
2. Студенты 3-го корпуса — проход запрещён (у них своё общежитие).
3. Все посторонние — проход запрещён.

Это и есть ACL. Если подходит студент 1-го корпуса, вахтёр, проверив его по первому пункту, пропускает его. Студент 3-го корпуса будет остановлен на втором пункте. Любой другой человек — на третьем.

1.2 Как устроено правило ACL? Критерии фильтрации

Каждое правило ACL состоит из двух частей: **критерий отбора (условие)** и **действие**.

В сетевом мире критериями чаще всего являются:

- **IP-адрес источника** (откуда пришёл пакет). Аналог: «Откуда вы приехали?»
- **IP-адрес назначения** (куда идёт пакет). Аналог: «В какую комнату вы направляетесь?»
- **Номер протокола** (например, TCP, UDP, ICMP). Аналог: «На каком языке вы говорите?»
- **Номер порта** (если протокол TCP/UDP). Это критически важный критерий. Порты указывают на конкретное сетевое приложение или службу. Например:
  - Порт 80 — HTTP (веб-трафик).
  - Порт 443 — HTTPS (защищённый веб-трафик).
  - Порт 25 — SMTP (электронная почта, отправка).
  - Порт 3389 — RDP (удалённый рабочий стол).

Действие всегда одно из двух: **permit** (разрешить) или **deny** (запретить).

Пример сетевого ACL:
Допустим, мы хотим разрешить из внутренней сети (192.168.1.0/24) доступ в Интернет только для веб-серфинга (HTTP/HTTPS) и запретить всё остальное.

Упрощённо правила могут выглядеть так:

1. **Критерий:** Источник 192.168.1.0/24, Назначение любое, Протокол TCP, Порт назначения 80 или 443. **Действие:** PERMIT.
2. **Критерий:** Любой источник, Любое назначение, Любой протокол. **Действие:** DENY.

Важнейшая особенность: ACL обрабатывает правила **последовательно, сверху вниз**. Как только пакет совпал с условием правила, применяется указанное действие, и проверка остальных правил прекращается. Поэтому порядок правил жизненно важен. Правило «запретить всё» (DENY ANY) должно почти всегда стоять **последним**, иначе оно заблокирует весь трафик, и правила ниже него никогда не сработают.

1.3 Типы ACL: Стандартные и расширенные

- **Стандартные ACL (Standard ACL):** Самые простые. Они идентифицируют трафик **только по IP-адресу источника**. Их аналог — сторож, который смотрит только на пропуск и не интересуется, куда вы идёте. Их часто размещают как можно ближе к месту назначения трафика, чтобы не блокировать лишнего.
  - **Пример правила:** permit host 192.168.1.5 — разрешить трафик от компьютера с адресом 192.168.1.5.
- **Расширенные ACL (Extended ACL):** Более гибкие и мощные. Они могут фильтровать трафик по множеству параметров: **IP-адрес источника и назначения, тип протокола (TCP, UDP, ICMP) и номера портов**. Их аналог — сторож, который проверяет и пропуск, и назначение, и даже цель визита. Их рекомендуется размещать как можно ближе к источнику трафика, чтобы «мусорный» трафик не загружал сеть.
  - **Пример правила:** deny tcp 192.168.1.0 0.0.0.255 any eq 23 — запретить всем компьютерам из сети 192.168.1.0/24 (telnet использует порт 23) устанавливать Telnet-соединения куда бы то ни было.

1.4 Сильные и слабые стороны ACL. Почему их недостаточно?

Сильные стороны:

- **Простота и предсказуемость:** Правила чёткие, их поведение легко понять.
- **Высокая скорость обработки:** На аппаратном уровне сравнение адресов и портов выполняется очень быстро.
- **Фундамент:** ACL является базовым строительным блоком для более сложных технологий.

Слабые стороны (ключевой момент для понимания перехода к брандмауэрам):

- **Статичность и бесконтекстность:** ACL видит каждый сетевой пакет **изолированно**. Он не понимает связи между пакетами. Для ACL пакет «из точки А в точку Б на порт 80» и пакет «из точки Б в точку А с произвольного порта» — это два абсолютно независимых события.
- **Проблема с возвратным трафиком:** Это главный недостаток. Представим, что внутренний компьютер (192.168.1.10) запрашивает веб-страницу у внешнего сервера (8.8.8.8:80). Внешний сервер должен отправить ответ. Но с точки зрения ACL на границе сети, этот ответ — это входящий пакет **от 8.8.8.8 на 192.168.1.10 с какого-то случайного порта (не 80!)**. Чтобы разрешить такой возвратный трафик, при использовании только ACL пришлось бы открывать входящие соединения со всего Интернета на все высокие порты внутренних

компьютеров, что смертельно опасно. ACL не помнит, что соединение было инициировано изнутри.

- **Не анализирует содержимое:** ACL не заглядывает «внутрь» пакета. Он не может определить, является ли HTTP-запрос легитимным или это попытка эксплуатации уязвимости веб-сервера.

Вывод по ACL: ACL — это эффективный инструмент для простой фильтрации на основе адресов и портов, но он **не является полноценным брандмауэром**. Он не умеет отслеживать состояние соединений, что критически важно для безопасного взаимодействия с динамическими средами, такими как Интернет. Эта проблема и привела к созданию технологий **брандмауэров с отслеживанием состояния (stateful firewalls)**, первым шагом к которым стал CBAC.

Глава 2. Технология брандмауэра: от простого фильтра к интеллектуальному защитнику

2.1 Эволюция: от ACL к Stateful Firewall

Чтобы преодолеть ограничения ACL, была разработана технология **брандмауэра с отслеживанием состояния (Stateful Firewall)**. Его основное отличие — наличие **таблицы состояний (State Table или Session Table)**.

Ключевая идея: Stateful Firewall — это уже не просто сторож со списком, а **внимательный инспектор с отличной памятью**. Он не только проверяет документы у входа, но и запоминает, кто, куда и зачем вошёл. Когда этот человек (пакет) возвращается назад, инспектор уже ожидает его и легко пропускает, не сверяясь с жёстким списком правил на каждый чих.

Процесс работы на примере:

1. **Инициирование изнутри:** ПК (192.168.1.10) хочет зайти на сайт google.com. Он отправляет пакет: **от 192.168.1.10:1500 (случайный высокий порт) на 172.217.16.206:80**.
2. **Проверка на выход:** Брандмауэр проверяет это исходящее соединение по своим правилам (которые могут быть очень строгими, например, «разрешить исходящий HTTP/HTTPS»). Если правило разрешает, брандмауэр **не просто пропускает пакет, а создаёт запись в своей таблице состояний**. В этой записи хранится информация о сессии: IP и порт источника, IP и порт назначения, протокол, таймеры.
3. **Возврат трафика:** Сервер Google отправляет ответ: **от 172.217.16.206:80 на 192.168.1.10:1500**.
4. **Проверка на вход:** Входящий пакет попадает на брандмауэр. Вместо того чтобы искать подходящее разрешающее правило во входном ACL (которого нет и быть не должно для случайных портов), брандмауэр проверяет свою **таблицу состояний**. Он видит: «Ага, у меня

есть активная сессия для этой пары адресов и портов, инициированная изнутри. Значит, этот пакет — законный ответ». И пропускает его.

5. **Завершение:** Когда сессия завершается (обмен пакетами FIN в TCP или по таймеру неактивности для UDP), запись удаляется из таблицы.

Аналогия: Вы звоните другу. Вы (инициатор) набираете его номер (порт). Ваш оператор связи (брандмауэр) фиксирует этот вызов. Когда друг вам перезванивает с того же номера, оператор видит, что это ответ на ваш вызов, и соединяет вас. Если же вам позвонит незнакомец (входящий вызов «из ниоткуда»), оператор его заблокирует или отправит на автоответчик (если нет соответствующего правила).

2.2 Компоненты и функции современного брандмауэра

Современный брандмауэр — это не одна функция, а комплекс модулей:

1. **Модуль фильтрации пакетов (Packet Filter):** Тот самый базовый механизм, похожий на ACL, для первичной грубой фильтрации.
2. **Модуль отслеживания состояния (Stateful Inspection):** Сердце брандмауэра. Отвечает за таблицу состояний и проверку входящего трафика на соответствие активным сессиям.
3. **Модуль трансляции сетевых адресов (NAT):** Часто встроен в брандмауэр. Позволяет множеству внутренних компьютеров с приватными адресами (например, 192.168.1.0/24) выходить в Интернет через один публичный IP-адрес, что также повышает безопасность, скрывая внутреннюю структуру сети.
4. **Модуль контроля на уровне приложений (Application Layer Gateway/Proxy):** Самый «умный» модуль. Он понимает протоколы конкретных приложений (HTTP, FTP, SIP). Он может анализировать содержимое пакетов: проверять URL, блокировать вредоносные скрипты в теле веб-страницы, контролировать команды в FTP-сессии. Это уже следующий эволюционный шаг — **межсетевые экраны следующего поколения (NGFW)**.

Таким образом, технология брандмауэра — это комплексный подход, который начинается с простой фильтрации (ACL), но обязательно включает в себя отслеживание состояния (stateful inspection) для безопасной работы с двусторонними сетевыми протоколами.

Глава 3. Контекстный контроль доступа (CBAC) — реализация stateful-инспекции в мире Cisco

3.1 Что такое CBAC и зачем он нужен?

Контекстный контроль доступа (CBAC — Context-Based Access Control) — это конкретная технология, реализованная в оборудовании Cisco, которая обеспечивает функциональность **брандмауэра с отслеживанием состояния (stateful firewall)**. Если ACL

видит пакеты как отдельные фотографии, то CBAC видит целый фильм — последовательность пакетов, связанных одним сетевым соединением (сессией).

Основная задача CBAC — решить ту самую проблему возвратного трафика, которую не мог решить обычный ACL, **без необходимости открывать в брандмауэре широкие и опасные «дыры»**.

Ключевая идея: CBAC действует как **умный и временный швейцар**. Когда резидент (внутренний хост) говорит: «Я жду гостя из такого-то ресторана (внешний сервер)», швейцар (CBAC) записывает это в свою книгу. Когда гость прибывает, швейцар сверяет его с записью и, если всё совпадает, временно открывает дверь именно для этого гостя. Как только визит завершён (сессия закрыта), запись в книге стирается, и дверь для этого гостя снова закрывается.

3.2 Принцип работы CBAC на детальном примере

Рассмотрим классический пример с протоколом FTP, который использует два соединения: управляющее (control, порт 21) и данных (data, порт 20 или случайный высокий порт). Без CBAC настроить безопасный доступ к FTP-серверу извне было крайне сложно.

Сценарий: Внутренний FTP-сервер (192.168.1.100). Внешний клиент (10.0.0.5) хочет подключиться.

Шаг 1: Настройка инспектирования.
Администратор настраивает CBAC на внешнем интерфейсе брандмауэра командой, подобной ip inspect name MYFW ftp. Это означает: «Следи за всеми FTP-сессиями, которые будут проходить через этот интерфейс, и запоминай их контекст».

Шаг 2: Инициирование соединения извне (пассивный режим FTP для простоты).

1. Внешний клиент (10.0.0.5:2000) отправляет запрос на внутренний сервер (192.168.1.100:21): «Хочу подключиться».
2. Этот пакет попадает на внешний интерфейс брандмауэра. На нём есть **входящий ACL (для трафика извне внутрь)**, который, в целях безопасности, скорее всего, запрещает почти всё. Но пусть для нашего примера есть правило: permit tcp any host 192.168.1.100 eq 21. Оно разрешает внешним клиентам стучаться только на 21-й порт сервера. Пакет пропускается.
3. CBAC, видя, что это начало FTP-сессии (порт 21), **создаёт в таблице состояний временную запись** об этом соединении.

Шаг 3: Динамическое открытие порта для данных — «магия» CBAC.
4. В процессе работы FTP-клиент и сервер договариваются о порте для передачи данных. Допустим, сервер говорит: «Подключайся для данных на мой порт 5000».
5. Когда сервер (192.168.1.100) готов отправить или принять данные, он, фактически, ожидает соединения от клиента (10.0.0.5) на порт 5000. Но наш входящий ACL не знает об этом и будет блокировать такие попытки!
6. **Вот здесь срабатывает CBAC.** Он, отслеживая управляющее соединение на порту 21,

«слышит» (анализирует) команды протокола FTP и понимает, что сервер ожидает данных на порту 5000 от клиента 10.0.0.5.
7. **CBAC автоматически и динамически создаёт временное разрешающее правило во входящем ACL** (или в отдельной таблице состояний) именно для этой пары: «разрешить пакеты от 10.0.0.5 на 192.168.1.100 порт 5000». Это правило существует **только на время сессии данных**.

Шаг 4: Передача данных и закрытие.
8. Клиент подключается для передачи данных, и брандмауэр пропускает эти пакеты, так как для них есть временное правило.
9. Как только передача данных завершена и управляющее соединение закрывается, CBAC **удаляет временную запись из таблицы состояний и стирает то самое временное правило** из ACL. Порт 5000 на сервере снова становится «невидимым» и защищённым для внешнего мира.

Итог: Благодаря CBAC нам не пришлось вручную и на постоянной основе открывать в ACL порт 5000 (который к тому же каждый раз разный) для всего Интернета. Брандмауэр сам, интеллектуально и на короткое время, открывал доступ ровно для того, кто в этом нуждался в рамках легитимной сессии.

3.3 Что инспектирует CBAC?

CBAC может отслеживать состояние не только FTP, но и множества других протоколов:

- **TCP-протоколы:** HTTP, SMTP, Telnet и др. Он отслеживает установление (SYN, SYN-ACK, ACK), поддержание и корректное завершение (FIN) TCP-сессий.
- **UDP-протоколы:** DNS, TFTP, VoIP (RTP). Для UDP, который не имеет состояния, CBAC эмулирует состояние, отслеживая запрос-ответ и используя таймеры неактивности.
- **Сложные протоколы:** Как в примере с FTP, CBAC может «заглядывать» внутрь протокола прикладного уровня, чтобы понять, какие динамические порты нужно открыть (это называется инспектирование протоколов уровня приложений — Application Layer Protocol Inspection).

Вывод по CBAC: CBAC — это мощный механизм, который превращает обычный маршрутизатор Cisco в полноценный stateful-брандмауэр. Он обеспечивает безопасность, отслеживая контекст соединений и динамически управляя доступом, что кардинально превосходит статическую фильтрацию ACL.

Глава 4. Политики брандмауэра, основанные на зонах (Zone-Based Policy Firewall — ZBF)

4.1 Проблема классического подхода и философия ZBF

Традиционная настройка брандмауэра (включая CBAC) часто была привязана к **сетевым интерфейсам**. Вы настраивали ACL и инспектирование на каждом интерфейсе отдельно: «на интерфейсе Gig0/0 (внешний) для входящего трафика делаем то-то, на интерфейсе Gig0/1 (внутренний) — другое». При росте сети это становилось сложно и неструктурированно.

Политики, основанные на зонах (Zone-Based Policy Firewall, ZBF или ZPF) — это современная, более гибкая и интуитивно понятная модель безопасности от Cisco. Она полностью заменяет старый метод с ACL на интерфейсах и CBAC.

Ключевая идея: Представьте, что вы делите свою сеть не на провода и порты, а на **зоны с одинаковым уровнем доверия**.

- **Зона Inside (Внутренняя):** Ваши доверенные пользователи, рабочие станции, принтеры.
- **Зона Outside (Внешняя):** Дикий и опасный Интернет.
- **Зона DMZ (Демилитаризованная зона):** Серверы, которые должны быть частично доступны извне (веб-сервер, почтовый сервер). Это буферная зона между доверенной и недоверенной.

Главное правило ZBF: Трафик может свободно течь *внутри* одной зоны без ограничений. Но весь трафик, который пытается пройти *между* разными зонами, **по умолчанию блокируется**. Чтобы его пропустить, необходимо явно создать и применить **политику безопасности (security policy)**.

Аналогия: Офисное здание. У вас есть:

- **Зона «Рабочие кабинеты» (Inside):** Сотрудники могут свободно ходить друг к другу.
- **Зона «Приёмная/Клиентская» (DMZ):** Клиенты могут заходить сюда, общаться с секретарём.
- **Зона «Улица» (Outside):** Совсем внешний мир.

Правила:

- Сотрудник из кабинета хочет выйти на улицу (Inside -> Outside) — нужен пропуск (политика).
- Клиент с улицы хочет зайти в приёмную (Outside -> DMZ) — нужно разрешение (политика).
- Клиент из приёмной хочет пройти в рабочий кабинет (DMZ -> Inside) — **строжайше запрещено** по умолчанию. Чтобы это разрешить, нужна очень серьёзная проверка и особая политика.
- Сотрудник из кабинета хочет зайти в приёмную (Inside -> DMZ) — нужна политика, но менее строгая.

4.2 Компоненты Zone-Based Policy Firewall

1. **Зоны (Zones):** Логические контейнеры для сетевых сегментов. Интерфейсы устройства присоединяются к зонам.
2. **Пары зон (Zone Pairs):** Направление, для которого создаётся политика. Например, пара «from Inside to Outside» описывает трафик из внутренней сети в Интернет.

3. **Классы трафика (Class-Maps):** Определяют, *какой именно* трафик мы хотим контролировать. Здесь как раз используются знания об ACL, протоколах и портах. Например, класс «ВЕБ-ТРАФИК» может отбирать весь TCP-трафик на порты 80 и 443.
4. **Политики (Policy-Maps):** Определяют, *что делать* с трафиком, попавшим в класс. Это сердце ZBF. Действия могут быть:

- **Inspect (Аналог CBAC):** Разрешить трафик и отслеживать его состояние (stateful inspection). Это самое распространённое действие для исходящего трафика.
- **Pass:** Разрешить трафик, но **не отслеживать** его состояние (статика, как ACL). Используется редко, для специфичных протоколов.
- **Drop:** Молча отбросить трафик (запретить).
- **Log:** Записывать информацию о трафике в лог.

5. **Применение политик (Service-Policy):** Политика (Policy-Map) применяется к конкретной паре зон (Zone-Pair).

4.3 Полный пример настройки и работы ZBF

Задача: Разрешить внутренним пользователям (Inside) выходить в Интернет (Outside) по HTTP/HTTPS и использовать DNS. Всё остальное запретить. Разрешить из Интернета доступ только к нашему веб-серверу в DMZ на порт 80.

Шаг 1: Создание зон и назначение интерфейсов.

- Создаём зоны: INSIDE, OUTSIDE, DMZ.
- Интерфейс Gig0/1 (подключен к локальной сети) назначаем зоне INSIDE.
- Интерфейс Gig0/0 (подключен к провайдеру) назначаем зоне OUTSIDE.
- Интерфейс Gig0/2 (подключен к серверам) назначаем зоне DMZ.
- **Важно:** Трафик между интерфейсами, принадлежащими одной зоне, не контролируется брандмауэром.

Шаг 2: Создание политики для доступа изнутри наружу (INSIDE -> OUTSIDE).

1. **Класс трафика «ВНУТРЕННИЙ-ВЕБ»:** Отбираем трафик TCP на порты 80, 443.
2. **Класс трафика «ВНУТРЕННИЙ-DNS»:** Отбираем трафик UDP на порт 53.
3. **Политика «ПОЛИТИКА-ВНУТРИ-НАРУЖУ»:**

- Для трафика из класса «ВНУТРЕННИЙ-ВЕБ» задаём действие **inspect** (разрешить и отслеживать состояние).
- Для трафика из класса «ВНУТРЕННИЙ-DNS» задаём действие **inspect.**
- Весь остальной трафик (не попавший в классы) по умолчанию будет отброшен (неявное правило deny all в конце политики).

4. **Создаём пару зон** from INSIDE to OUTSIDE и применяем к ней нашу политику «ПОЛИТИКА-ВНУТРИ-НАРУЖУ».

Результат: Пользователи могут открывать сайты и использовать DNS. Возвратный трафик разрешается динамически благодаря действию inspect (stateful inspection). Они не могут использовать ICMP, Telnet, FTP (если только эти протоколы не добавлены в политику).

Шаг 3: Создание политики для доступа извне в DMZ (OUTSIDE -> DMZ).

1. **Класс трафика «ВНЕШНИЙ-К-ВЕБСЕРВЕРУ»:** Отбираем трафик TCP на порт 80 с любым источником на IP-адрес нашего веб-сервера в DMZ.
2. **Политика «ПОЛИТИКА-СНАРУЖИ-В-DMZ»:** Для трафика из этого класса задаём действие **inspect**.
3. **Создаём пару зон** from OUTSIDE to DMZ и применяем политику.

Результат: Любой пользователь из Интернета может подключиться к порту 80 нашего веб-сервера. Попытки подключиться к другим портам этого сервера или к другим серверам в DMZ будут заблокированы (так как не попали в класс и подпадают под неявный deny).

Шаг 4: Что запрещено по умолчанию?

- **OUTSIDE -> INSIDE:** Полностью запрещено. Нет созданной пары зон и политики для этого направления. Никто из Интернета не может инициировать соединение во внутреннюю сеть. Это максимально безопасная конфигурация.
- **DMZ -> INSIDE:** Полностью запрещено. Даже если злоумышленник скомпрометирует веб-сервер в DMZ, он не сможет с него атаковать внутреннюю сеть. Это принцип сегментации.

4.4 Преимущества Zone-Based Policy Firewall

1. **Более ясная логика:** Конфигурация отражает политику безопасности («что и кому разрешено»), а не технические детали интерфейсов.
2. **Повышенная безопасность:** Чёткое разделение на зоны и правило «запрещено по умолчанию» исключает случайные ошибки конфигурации, оставляющие «дыры».
3. **Гибкость и масштабируемость:** Легко добавлять новые интерфейсы в существующие зоны или создавать новые зоны (например, «Зона для гостей», «Зона для IoT-устройств») с собственными политиками.
4. **Унификация:** Один механизм (ZBF) заменяет собой разрозненные ACL, CBAC и NAT, управляя всем через единую понятную структуру.

Заключение

В рамках изучения темы «Реализация технологий брандмауэра» мы прошли путь от простейших статических фильтров до сложных интеллектуальных систем управления безопасностью на основе зон.

1. **ACL** — это базовый, статичный инструмент для фильтрации по адресам и портам, неспособный самостоятельно справиться с динамическим характером сетевых протоколов, особенно с возвратным трафиком.
2. **Технология брандмауэра** решает эту проблему за счёт **отслеживания состояния (stateful inspection)**. Брандмауэр запоминает инициированные из доверенной зоны соединения и

автоматически разрешает ответы на них, не требуя опасных статических правил «на всё про всё».

3. **CBAC** — это конкретная реализация stateful-брандмауэра в экосистеме Cisco, которая не только отслеживает состояния TCP/UDP сессий, но и может анализировать протоколы прикладного уровня (как FTP) для динамического открытия портов, необходимых для работы сложных протоколов.
4. **Политики, основанные на зонах (ZBF)** — это современный, структурированный подход к проектированию безопасности. Он заменяет привязку правил к интерфейсам на логическую модель зон с разным уровнем доверия. В этой модели весь межзональный трафик запрещён по умолчанию, а разрешается только явно описанными политиками, что кардинально повышает безопасность и удобство управления.

Эти технологии образуют иерархическую пирамиду, где каждый следующий уровень использует идеи предыдущего, добавляя к ним новые возможности интеллектуального анализа и управления. Понимание этой эволюции — от простого списка (ACL) через контекстное запоминание (CBAC) к стратегическому разделению на зоны ответственности (ZBF) — является фундаментом для грамотной настройки и администрирования сетевой безопасности в любой современной организации.

Контрольные вопросы к ТЕМЕ-23

1. Что такое ACL и какова его основная функция в сетевой безопасности?
2. Назовите ключевое отличие стандартных (Standard) и расширенных (Extended) списков контроля доступа (ACL).
3. Почему порядок правил в ACL имеет критическое значение? Приведите простой пример.
4. В чём заключается главный недостаток ACL, который привёл к появлению брандмауэров с отслеживанием состояния?
5. Какую основную проблему сетевой безопасности решает технология брандмауэра с отслеживанием состояния (Stateful Firewall)?
6. Что такое «таблица состояний» (State Table) и какую роль она играет в работе stateful брандмауэра?
7. Объясните, для чего предназначен контекстный контроль доступа (CBAC) и в чём его основная «магия».
8. Приведите пример протокола, для безопасной работы с которым особенно необходим механизм, подобный CBAC, и объясните почему.
9. В чём заключается основная философия подхода к безопасности на основе зон (Zone-Based Policy)?
10. Каково главное правило движения трафика между разными зонами безопасности в модели Zone-Based Policy Firewall?
11. Какое действие политики (policy action) в ZBF является аналогом работы CBAC и обеспечивает отслеживание состояния соединения?
12. Назовите одно ключевое преимущество использования модели безопасности на основе зон (ZBF) перед классической настройкой ACL на интерфейсах.

Практические задания к ТЕМЕ-23

Задание 1. Решение проблемы обратного трафика в «ТехноБанке»

Системный администратор филиала «ТехноБанка» в Казани настроил на маршрутизаторе расширенный ACL для защиты внутренней сети (192.168.10.0/24). Он разрешил только исходящие HTTP- и HTTPS-запросы во внешнюю сеть (Интернет). Однако сотрудники жалуются, что веб-страницы в браузере не открываются, хотя сам запрос, кажется, проходит. Администратор проверил ACL: правила `permit tcp 192.168.10.0 0.0.0.255 any eq 80` и `permit tcp 192.168.10.0 0.0.0.255 any eq 443` присутствуют и применяются на правильном интерфейсе. Объясните, в чём заключается фундаментальная причина возникшей проблемы, исходя из ограничений технологии ACL, и какую базовую технологию брандмауэра необходимо задействовать для её корректного решения.

Задание 2. Конфигурация FTP-доступа в «Городской больнице №1»

В «Городской больнице №1» Екатеринбурга развёрнут внутренний FTP-сервер (172.16.50.10) для обмена документами между отделениями. Администратору необходимо настроить доступ к этому серверу для партнёрской лаборатории из внешней сети, используя технологию брандмауэра. Лаборатория имеет статический IP-адрес 85.93.10.15. Стандартный ACL, разрешающий входящие подключения с этого адреса к серверу, настроен, но передача файлов не работает. Какой ключевой аспект работы протокола FTP (в его активном режиме) не учитывает статический ACL, и какая функция брандмауэра (конкретно, какого типа инспектирование) требуется для автоматического и безопасного решения этой проблемы?

Задание 3. Проектирование зон безопасности для «Умного кампуса»

При проектировании сети «Умного кампуса» нового техникума в Томске администратор решил внедрить модель безопасности на основе зон (ZBF). Он выделил следующие логические сегменты сети: 1) компьютеры администрации и бухгалтерии, 2) студенческие компьютеры в лабораториях, 3) общедоступные Wi-Fi точки для гостей, 4) серверная с внутренними сервисами (1С, Moodle), 5) серверная с веб-сайтом техникума. Помогите администратору, классифицировав эти сегменты по зонам безопасности (например, Inside, DMZ, Guest), исходя из принципа одинакового уровня доверия. Обоснуйте, почему вы поместили каждую группу именно в эту зону, особенно уделив внимание двум серверным сегментам.

Задание 4. Анализ сбоя видеоконференции в «Дальневосточной логистике»

В компании «Дальневосточная логистика» (Владивосток) после настройки политик межсетевого экрана на основе зон перестала работать система видеоконференцсвязи между филиалами. Система использует для передачи голоса и видео протоколы UDP (RTP) с динамическими номерами портов. Администратор создал политику для пары зон `from INSIDE to OTHER_BRANCH` с действием `pass` для UDP-трафика на нужный адрес. Соединение устанавливается, но звука и видео нет. Почему действие `pass` в ZBF оказалось неприменимым для данной ситуации, и какое действие следовало применить, чтобы обеспечить работу динамических UDP-потоков безопасным образом?

Задание 5. Выбор стратегии для точки доступа в «Парке культуры»

В «Центральном парке культуры и отдыха» Нижнего Новгорода организована бесплатная Wi-Fi сеть для посетителей. Администратору нужно обеспечить гостям доступ только в Интернет (HTTP/HTTPS), но полностью изолировать их друг от друга и от внутренней сети администрации парка. У него есть два варианта: 1) Настроить на точке доступа подробные расширенные ACL, запрещающие трафик между клиентами и во внутреннюю сеть. 2) Реализовать простую модель ZBF с двумя зонами: Guest (гостевой Wi-Fi) и Inside (сеть администрации). Какая из этих стратегий будет логичнее, проще в поддержке и соответствует философии «запрещено по умолчанию»? Объясните, как будет работать правило безопасности между зонами в выбранной вами стратегии.

ТЕМА-24. Реализация технологий предотвращения вторжения

Введение: От охраны периметра к круглосуточному телохранителю

Представьте, что ваша компьютерная сеть — это банк. У банка есть входная дверь. В роли такой двери в мире сетевой безопасности долгое время выступал **межсетевой экран (Firewall)**. Его задача — проверять, кто входит и выходит, сверяясь со списком правил (например, "сотрудник с пропуском №5 может войти в хранилище с 9 до 18"). Он смотрит на базовую информацию: откуда пришел пакет данных (IP-адрес), куда он направляется (порт) и по какому протоколу.

Но что, если злоумышленник каким-то образом получил легитимный пропуск (украл учётные данные сотрудника) или проник внутрь через незаметный лаз (уязвимость в программном обеспечении)? Он уже внутри банка. Теперь он может спокойно, никого не беспокоя, начать вскрывать сейфы, устанавливать подслушивающие устройства или пробивать тоннель в соседний магазин. Межсетевой экран на это уже не среагирует — для него этот человек "свой".

Здесь на сцену выходит **технология предотвращения вторжений (Intrusion Prevention System — IPS)**. Если Firewall — это охрана на входе, то IPS — это **круглосуточный, бдительный телохранитель, который следует за каждым человеком внутри банка**, анализирует его действия в реальном времени и мгновенно пресекает любые подозрительные или опасные активности. Он не просто наблюдает (как это делает его "младший брат" — система обнаружения вторжений, IDS), а активно вмешивается, блокируя атаку на месте.

Цель данного материала — дать фундаментальное понимание сути IPS: как она работает (технологии), на что она опирается (сигнатуры), как её внедрить в сеть и как убедиться, что она выполняет свою работу.

Глава 1. IPS технологии: Механизмы бдительного телохранителя

1.1. Сущность и отличие от IDS

Первый и ключевой концепт, который необходимо усвоить — это **разница между обнаружением (Detection) и предотвращением (Prevention)**.

- **IDS (Intrusion Detection System — Система обнаружения вторжений)** — это **система видеонаблюдения и сигнализации**. Она пассивно наблюдает за трафиком (делает его копию), анализирует его, и если видит что-то подозрительное, посылает тревожный сигнал (alert) администратору. "ВНИМАНИЕ! Камера №12 зафиксировала человека, который пытается открыть сейф!" Запись есть, сигнал есть, но кто остановит злоумышленника? Это должен сделать администратор, получив сигнал и приняв меры. Между обнаружением и реакцией проходит время, за которое атака может завершиться.
- **IPS (Intrusion Prevention System — Система предотвращения вторжений)** — это **сигнализация, интегрированная с автоматической системой нейтрализации**. Это тот же самый бдительный "телохранитель", который не только видит попытку открыть сейф, но и **мгновенно хватает злоумышленника за руку, обездвиживает его и блокирует ему дальнейшие действия**. IPS стоит **"в разрыв"** сетевого трафика (in-line), пропуская через себя каждый пакет данных в реальном времени. Решение — блокировать или пропустить — она принимает на лету, без участия человека.

Аналогия:
Допустим, к вам в почтовый ящик приходят письма.

- **Firewall** решает, может ли почтальон из конкретного города (IP) вообще опускать письма в ящики на вашей улице (сети).
- **IDS** — это клерк, который копирует все пришедшие письма, вскрывает копии и читает их. Если в копии письма он находит фразу "выиграл миллион" (сигнатура мошенничества), он бежит к вам и кричит: "Вам пришло мошенническое письмо!".
- **IPS** — это клерк, который сидит прямо перед вашим ящиком. Он вскрывает **каждое** входящее письмо до того, как оно упадёт в ящик. Читает его. Если письмо нормальное — он заклеивает его обратно и кладёт в ящик. Если же он видит фразу "выиграл миллион" — он **немедленно рвёт это письмо в клочья и выбрасывает его в мусор**, и оно никогда не доходит до вас. При этом он также может позвонить отправителю (источнику атаки) и сказать: "Больше писем с этого адреса не принимаются".

1.2. Основные методы анализа трафика в IPS

Как же IPS понимает, что перед ней — атака, а не легитимный трафик? Она использует три фундаментальных метода, часто комбинируя их.

1. Обнаружение на основе сигнатур (Signature-based Detection).
Это самый распространённый и точный метод для известных угроз. **Сигнатура** — это уникальный шаблон или "отпечаток пальца" атаки. IPS содержит огромную базу таких сигнатур, которая постоянно обновляется производителем.

- **Пример:** Вирус "Чернобыль" (CIH) заражал исполняемые файлы, добавляя в них определённую последовательность байтов 0xEA 0x75 0x00 0x00 0x00. Сигнатурой для него была именно эта последовательность. Если IPS, анализируя трафик или файл, видит эту цепочку байтов — она блокирует пакет и останавливает передачу файла. Это похоже на работу антивируса.
- **Более сложный пример (для сетевой атаки):** Атака "Ping of Death". Её суть — отправка ICMP-пакета (ping) размером больше максимально допустимого (65 535 байт), что могло вызвать крах системы. Сигнатура для такой атаки: ICMP-пакет, у которого в заголовке указан размер > 65535. IPS, видя такой пакет, блокирует его.

2. Обнаружение аномалий (Anomaly-based Detection).
Этот метод пытается выявить неизвестные атаки или сложные целевые вторжения, для которых ещё нет сигнатуры. Он основан на создании **модели нормального поведения** (baseline) для сети, пользователя или приложения. Всё, что сильно отклоняется от этой модели, считается подозрительным.

- **Аналогия:** Вы знаете, что ваш коллега Иван обычно заходит в офис в 9 утра, работает за компьютером, в обед ходит в столовую и уходит в 18:00. Это его "базовая линия поведения". Если однажды система контроля доступа показывает, что Иван вошёл в офис в 3 часа ночи и попытался открыть дверь в серверную, IPS на основе аномалии зафиксирует: "Нарушено нормальное поведение пользователя 'Иван' — несанкционированный доступ в нерабочее время к критичному ресурсу". Это событие будет заблокировано и поднимет тревогу.
- **Технический пример:** Веб-сервер в вашей сети в обычном режиме обрабатывает не более 100 запросов в секунду от одного IP-адреса. Базовая линия зафиксировала это. Если вдруг с одного адреса начинает поступать 10 000 запросов в секунду (атака типа HTTP-flood, часть DDoS), IPS распознает это как **аномалию** (отклонение от baseline) и начнёт отбрасывать эти запросы, защищая сервер.

3. Обнаружение на основе политик (Policy-based Detection).
Этот метод наиболее прост и основан на строгих, заранее заданных администратором правилах. Он не ищет конкретные сигнатуры или аномалии, а просто следит за соблюдением **политики безопасности**.

- **Пример:** Политика безопасности компании гласит: "На рабочих станциях запрещено использование протокола Telnet из-за его небезопасности (данные передаются в открытом виде)". Администратор создаёт в IPS правило: **"Блокировать весь трафик по протоколу TCP к порту 23 (стандартный порт Telnet) и от него"**. Неважно, является ли сессия Telnet атакой или легитимным действием сотрудника, — она будет заблокирована, потому что нарушает политику. Это жёсткий, но очень эффективный метод контроля.

Сводная аналогия трёх методов:
Представьте охрану на входе в НИИ.

- **Сигнатурный метод:** У охраны есть папка с фотографиями известных шпионов. Каждого входящего он сравнивает с этими фото. Совпадение — задержание.

- **Метод аномалий:** Охрана изучает обычный график и поведение сотрудников: кто во сколько приходит, куда ходит. Если сотрудник из бухгалтерии внезапно пытается пройти в закрытую лабораторию ядерной физики в 2 ночи — это аномалия, доступ блокируется для расследования.
- **Метод политик:** На входе висит правило: "Проход с животными запрещён". Охрана не анализирует, злая собака или милый хомячок. Видит животное — не пускает. Правило есть правило.

Глава 2. IPS сигнатуры: "Отпечатки пальцев" цифровых угроз

Сигнатуры — это "мозг" и "память" сигнатурного IPS. Понимание их структуры и типов критически важно.

2.1. Что такое сигнатура?

Сигнатура IPS — это формализованное правило или шаблон, который описывает уникальные характеристики известной сетевой атаки, вредоносного программного обеспечения или уязвимости. Это не просто строка текста, а сложная конструкция, которая может анализировать различные уровни сетевого стека OSI.

Аналогия с медициной: Сигнатуры — это симптомы болезней, собранные в медицинском справочнике. Высокая температура + кашель + боль в горле = симптомокомплекс (сигнатура) ангины. Врач (IPS), видя эти симптомы у пациента (пакет данных), ставит диагноз "ангина" (обнаруживает атаку) и назначает лечение (блокирует трафик).

2.2. Типы сигнатур

1. Сигнатуры контекста (Context Signatures).
Анализируют **заголовки** сетевых пакетов (IP, TCP, UDP, ICMP). Они не смотрят внутрь "тела" пакета (полезную нагрузку — payload).

- **Пример:** Сигнатура для обнаружения сканирования портов может выглядеть так: Если с одного IP-адреса за 5 секунд поступили TCP-пакеты с флагом SYN (запрос на соединение) на более чем 20 различных портов одного целевого хоста. Это не нарушает правила сети само по себе, но является явным признаком разведки, и IPS может начать блокировать такие запросы или хотя бы занести источник в "чёрный список" на время.

2. Сигнатуры содержимого (Content Signatures).
Анализируют **полезную нагрузку (payload)** пакета — те самые данные, которые передаются внутри (например, тело HTTP-запроса, вложение письма, команда в протоколе).

- **Пример (очень упрощённо):** Попытка SQL-инъекции. Злоумышленник пытается через форму ввода на сайте передать команду базе данных. Вместо имени пользователя он вводит что-то вроде: `' OR '1'='1`. Сигнатура может искать в HTTP-запросах POST или GET последовательность символов `' OR` или `--` (комментарий в SQL), что является типичным признаком инъекции. Обнаружив это в теле пакета, IPS заблокирует этот HTTP-запрос до того, как он дойдёт до веб-сервера.
- **Пример для вируса:** Сигнатура может искать в передаваемых файлах шестнадцатеричную последовательность, соответствующую коду вируса.

3. Комплексные (гибридные) сигнатуры.
Самые продвинутые. Они комбинируют анализ контекста и содержимого, а также учитывают состояние (state) соединения.

- **Пример:** Атака, эксплуатирующая уязвимость в веб-сервере. Сигнатура может быть такой:

1. **Контекст:** Пакет адресован на порт 80 (HTTP) нашего веб-сервера.
2. **Содержимое:** В теле HTTP-запроса содержится очень длинная строка, превышающая допустимый буфер веб-приложения (например, 5000 символов вместо обычных 100).
3. **Состояние:** Это первый пакет в новом TCP-соединении (флаг SYN).
   Только одновременное выполнение всех трёх условий (контекст + содержимое + состояние) приведёт к срабатыванию сигнатуры и блокировке пакета. Это снижает количество ложных срабатываний.

2.3. Жизненный цикл сигнатур

1. **Исследование:** Производители IPS, специализированные лаборатории (например, Cisco Talos, Palo Alto Unit 42) и сообщество исследователей безопасности анализируют новую вредоносную программу или метод атаки.
2. **Создание:** Выделяются уникальные характеристики атаки и на их основе создаётся сигнатура (правило).
3. **Тестирование:** Сигнатура тестируется, чтобы убедиться, что она корректно обнаруживает угрозу и не даёт ложных срабатываний на легитимный трафик.
4. **Публикация (обновление):** Сигнатура добавляется в базу данных и распространяется среди всех клиентов в виде **обновления сигнатур (Signature Update)**. Это критически важный процесс, аналогичный обновлению антивирусных баз.
5. **Развёртывание:** Администратор сети применяет обновление на своих IPS.
6. **Действие:** IPS начинает использовать новую сигнатуру для проверки трафика.
7. **Устаревание:** Со временем атака может перестать использоваться, или сигнатура вызывает слишком много ложных срабатываний. Тогда её могут отключить или удалить из базы.

Глава 3. Реализация IPS: Размещение телохранителя в структуре сети

Реализация IPS — это не просто установка программы. Это процесс выбора типа системы, её физического или виртуального размещения и тонкой настройки.

3.1. Типы IPS

1. Сетевые IPS (NIPS — Network-based IPS).
Это специализированное аппаратное устройство или программное решение, которое устанавливается в **ключевых точках сети** для мониторинга всего трафика, проходящего через эти точки. Это "телохранитель на перекрёстке всех коридоров".

- **Типичное место установки:** На границе сети, между внешним маршрутизатором и внутренним firewall, для защиты от внешних угроз. Или между критически важными сегментами внутренней сети (например, между сетью бухгалтерии и сетью отдела разработки).
- **Аналогия:** Стационарный пост охраны в главном вестибюле бизнес-центра, который проверяет всех, кто проходит через турникеты.

2. Узловые IPS (HIPS — Host-based IPS).
Это программный агент, который устанавливается **непосредственно на защищаемый сервер или рабочую станцию**. Он контролирует трафик и активность только на этом конкретном хосте: системные вызовы, изменения в реестре, запуск процессов, сетевую активность этого хоста.

- **Преимущество:** Видит зашифрованный трафик после его расшифровки на хосте (в то время как сетевая IPS слепа перед трафиком, зашифрованным по SSL/TLS).
- **Аналогия:** Персональный телохранитель, приставленный к конкретному важному лицу (CEO компании). Он следует за ним везде, знает все его привычки и защищает именно его, независимо от местоположения.
- **Пример:** HIPS на веб-сервере может заблокировать попытку скрипта изменить системный файл /etc/passwd или запустить подозрительный процесс с правами root.

3. Беспроводные IPS (WIPS — Wireless IPS).
Специализированные системы для защиты беспроводных сетей (Wi-Fi). Они обнаруживают и блокируют "злонамеренные" точки доступа (Rogue AP), атаки типа "человек посередине" (Evil Twin) и несанкционированные клиентские устройства.

3.2. Ключевые режимы работы и размещения

При внедрении NIPS существует два фундаментальных подхода к её интеграции в сеть:

1. Режим "In-line" (Сквозной или активный режим).
Это основной режим работы полноценной IPS. Устройство физически или

логически **встроено в канал передачи данных**. Все пакеты проходят через него. Это позволяет не только анализировать, но и **отбрасывать, модифицировать или задерживать пакеты в реальном времени**.

- **Как это выглядит:** Сетевой кабель от маршрутизатора подключается не напрямую к внутреннему коммутатору, а **сначала в порт IPS, а затем из другого порта IPS — на коммутатор**. IPS становится "невидимым" узлом на пути следования трафика.
- **Аналогия:** Пропускной пункт на мосту. Чтобы попасть в город, **вся** машина должна проехать через контрольно-пропускной пункт (IPS). Если машина в "чёрном списке" (совпадает с сигнатурой), шлагбаум не откроется.

2. Режим "Tap" или "Span" (Пассивный мониторинг).
В этом режиме IPS **не находится в прямом трафик-потоке**. К ней подводится копия трафика с помощью специального аппаратного ответвителя (Network TAP) или функции зеркалирования портов (SPAN) на коммутаторе. В этом режиме IPS работает как **IDS** — только обнаруживает угрозы и генерирует алерты, но не может их блокировать. Этот режим часто используется на этапе внедрения для "обучения" системы и анализа трафика без риска блокировки легитимного трафика.

- **Аналогия:** Наблюдатель на вышке рядом с мостом. Он видит все машины, фотографирует номера подозрительных и докладывает по рации на пост у въезда в город. Но остановить машину он физически не может.

3.3. Базовые шаги реализации (внедрения)

1. **Определение целей:** Что мы защищаем? Веб-серверы? Базу данных? Всю внутреннюю сеть? От каких угроз (внешних, внутренних)?
2. **Выбор типа и модели IPS:** Исходя из целей, пропускной способности сети и бюджета.
3. **Определение точек размещения:** Где в топологии сети будет стоять устройство? Обычно сразу за внешним firewall на границе сети.
4. **Физическое подключение (для NIPS):** Установка в стойку, подключение в разрыв канала связи (in-line) или на SPAN-порт коммутатора (tap).
5. **Базовая сетевая настройка:** Назначение IP-адресов, настройка маршрутов.
6. **Настройка политик безопасности (самый важный этап):**

- **Создание "белых списков" (Whitelist):** Определение абсолютно доверенного трафика (например, внутренние IP-адреса, трафик от системы обновления Windows), который IPS не должна проверять. Это снижает нагрузку.
- **Настройка групп сигнатур:** Включение только тех категорий сигнатур, которые актуальны для вашей сети. Например, если у вас нет серверов на базе Apache, можно отключить сигнатуры для атак специфично на Apache. Это снижает риск ложных срабатываний.
- **Определение действий (Actions):** Что делать при срабатывании сигнатуры? **Блокировать источник? Блокировать соединение? Только предупредить?** Чаще всего для критических сигнатур настраивается Блокировать источник на X минут.

- **Настройка исключений (Exceptions):** Если легитимное приложение или действие в вашей сети ложно срабатывает на какую-то сигнатуру, для неё создаётся исключение, чтобы не блокировать бизнес-процесс.

7. **Тестирование в пассивном режиме:** Первоначальный запуск в режиме мониторинга (без блокировки) для оценки количества и характера угроз, а также для выявления ложных срабатываний.
8. **Перевод в активный режим:** После тонкой настройки политик и исключений система переводится в активный режим (in-line) с функцией блокирования.

Глава 4. Проверка и мониторинг IPS: Оценка работы телохранителя

Установленная и работающая IPS — это не "поставил и забыл". За ней нужен постоянный контроль, как за любым сложным охранным механизмом.

4.1. Ключевые показатели и логи

1. Журналы событий (Logs).
Это хроника всех действий IPS: какие пакеты были проанализированы, какие сигнатуры сработали, какие действия были предприняты, какие исключения применены.

- **Что смотреть:** Администратор должен регулярно просматривать логи, особенно **сработавшие сигнатуры с высоким уровнем критичности (Severity: Critical, High)**. Это прямой индикатор попыток атаки.
- **Пример записи в логе:** [Время: 2023-10-27 14:05:01] [Уровень: Высокий] [Сигнатура: "WEB-IIS CodeRed v2 root.exe access"] [Действие: Блокировано] [Источник: 94.102.49.100:54321] [Получатель: 192.168.1.10:80] [Протокол: TCP]. Эта запись говорит, что IPS заблокировала известную атаку на веб-сервер IIS от конкретного злоумышленника.

2. Производительность и состояние системы.
IPS, работающая в режиме in-line, — это потенциальное "узкое место". Необходимо следить за:
* **Загрузка ЦП/Памяти:** Высокая загрузка может привести к потере пакетов и задержкам в сети.
* **Пропускная способность (throughput):** Сравнивается ли с заявленной производителем? Не достигнут ли предел?
* **Количество проверенных пакетов/сессий в секунду.**
* **Задержка (latency):** Время, которое IPS добавляет к передаче пакета. Для критичных приложений (голос, видео) это важно.

3. Эффективность обнаружения.
Оценивается по двум ключевым метрикам, взятым из теории обнаружения сигналов:

- **Ложные срабатывания (False Positives — FP):** Это ситуация, когда IPS **ошибочно идентифицирует легитимный трафик как атаку** и блокирует его. **Высокий уровень FP — главная проблема плохо настроенной IPS.** Это ведёт к нарушению работоспособности сервисов и недоверию к системе.
  - **Пример:** Сигнатура, настроенная на блокировку передачи архивов .ZIP, может заблокировать сотрудника, который законно отправляет проект архивом партнёру.
- **Ложные пропуски (False Negatives — FN):** Это ситуация, когда **реальная атака осталась незамеченной и не заблокированной** IPS. Это более опасная ситуация, так как создаёт ложное чувство безопасности.
  - **Пример:** Появилась новая, неизвестная (zero-day) атака, для которой ещё не создана сигнатура. Метод аномалий может её не обнаружить, если поведение атакующего имитирует нормальную активность.

Идеальная IPS должна минимизировать **оба** этих показателя. На практике настраивается баланс: чуть больше FP, но почти нулевые FN для критичных угроз.

4.2. Процессы мониторинга и проверки

1. Регулярный просмотр отчётов и дашбордов.
Современные IPS имеют веб-интерфейсы с графическими панелями (dashboard), которые визуализируют информацию: топ источников атак, топ сигнатур, статистику по протоколам, карту угроз в реальном времени.

2. Настройка и проверка оповещений (Alerts).
Критически важные события (например, срабатывание сигнатуры с уровнем Critical или множественные атаки с одного IP) должны дублироваться не только в лог, но и приходить администратору по **email, SMS или в SIEM-систему** (System Information and Event Management — централизованная система управления событиями безопасности).

- **Пример настройки:** "Если в течение 1 минуты с одного IP-адреса срабатывает 5 различных сигнатур уровня High — отправить SMS администратору и автоматически добавить этот IP в чёрный список на 24 часа".

3. Тестирование IPS.
Чтобы убедиться, что IPS не "проспала" и корректно блокирует угрозы, проводят контролируемые тесты.

- **Использование тестовых сигнатур:** Многие производители предоставляют "пробные" безвредные сигнатуры, имитирующие атаку. Отправив такой тестовый пакет, можно убедиться, что IPS его обнаружила и выполнила заданное действие.
- **Важность:** Это как учебная тревога для пожарной охраны. Проверяет, работает ли система оповещения и реагирования в принципе.

4. Анализ ложных срабатываний и настройка исключений.
Это рутинная, но vitalная работа. Каждое ложное срабатывание должно быть расследовано:
1. Почему легитимный трафик попал под сигнатуру?
2. Можно ли изменить сигнатуру или её параметры (например, порог срабатывания)?
3. Если нет — создать **точечное исключение (exception)** для этого конкретного правила и для этого конкретного источника/получателя трафика, если он доверенный.

5. Обновление!
Это нельзя забывать. Регулярное обновление **баз сигнатур** и **программного обеспечения** IPS — это залог её способности противостоять новейшим угрозам. Процесс должен быть автоматизированным и проверяемым.

Заключение

Технология предотвращения вторжений (IPS) — это активный, интеллектуальный рубеж обороны современной сети, перекрывающий те угрозы, которые обходят классические межсетевые экраны. Её ядро — это комбинация методов: точного распознавания "отпечатков пальцев" известных атак (сигнатуры), выявления отклонений от нормы (аномалии) и жёсткого следования правилам (политики). Успешная реализация IPS требует не только грамотного размещения в топологии сети (in-line для предотвращения, tap для мониторинга), но и кропотливой настройки политик, минимизации ложных срабатываний и непрерывного мониторинга её работы, состояния и эффективности.

Понимание этих фундаментальных идей — сигнатур, методов анализа, режимов работы и принципов мониторинга — создаёт прочную основу для освоения практических навыков настройки и управления системами предотвращения вторжений, что является ключевой компетенцией специалиста по сетевому и системному администрированию в условиях современных киберугроз.

Контрольные вопросы к ТЕМЕ-24

1. В чём заключается ключевое различие между системами обнаружения (IDS) и предотвращения (IPS) вторжений?
2. Какой метод обнаружения атак в IPS основан на поиске уникального «отпечатка пальца» известной угрозы?
3. Как называется режим работы IPS, при котором она физически или логически встроена в канал передачи данных и может блокировать трафик?
4. Приведите пример технической сигнатуры для сетевой атаки, основанной на анализе содержимого (payload) пакета данных.
5. Какой тип IPS устанавливается непосредственно на защищаемый сервер или рабочую станцию для контроля её активности?

6. Что такое «ложное срабатывание» (False Positive) системы IPS и к каким последствиям оно может привести?
7. Как называется процесс создания модели нормального поведения сети или хоста, используемый IPS для выявления аномалий?
8. Для какой основной цели при первоначальном внедрении IPS её часто временно запускают в пассивном режиме мониторинга (режим TAP/SPAN)?
9. Что необходимо регулярно загружать в IPS для обеспечения её способности обнаруживать новые, ранее неизвестные угрозы?
10. Назовите одну из ключевых метрик, за которой должен следить администратор для оценки нагрузки и корректной работы IPS в сети.
11. Какой метод обнаружения в IPS просто запрещает трафик, нарушающий строго заданные администратором правила, например, использование определённого протокола?
12. Что такое «узловая IPS» (HIPS) и какое у неё есть преимущество перед сетевой IPS (NIPS) в контексте зашифрованного трафика?

Практические задания к ТЕМЕ-24

Задание 1. Анализ режима работы

Сетевой администратор филиала компании «Ромашка» в Твери установил новую IPS. После внедрения он заметил, что система успешно выявляет множество подозрительных событий, однако трафик от заблокированных IP-адресов продолжает достигать серверов, вызывая перегрузку. Какой ключевой режим работы IPS, вероятно, был выбран администратором ошибочно? Объясните, в чём заключается его ошибка и какой режим необходимо активировать для решения проблемы.

Задание 2. Выбор метода защиты

В небольшом интернет-магазине из Новосибирска, торгующем сувенирами, обновили веб-платформу. Через несколько дней IPS начала массово блокировать легитимных покупателей, которые пытались оставить отзыв с длинным текстом в соответствующей форме. Анализ показал, что срабатывает сигнатура «WEB-CLIENT Wordpress long comment attempt». Какие два основных метода обнаружения (помимо сигнатурного) имеются в арсенале IPS, и какой из них мог бы помочь в данной ситуации, чтобы не отключать защиту полностью? Предложите общий принцип настройки.

Задание 3. Локализация ложного срабатывания

IPS в сети колледжа в Екатеринбурге блокирует доступ к образовательному порталу «Российская электронная школа» для компьютеров в одном из кабинетов. В логах указана сигнатура «WEB-MISC /etc/passwd access attempt». Администратор выяснил, что в этом кабинете студенты на уроках информатики легитимно изучают работу с Linux через веб-терминал, встроенный в портал. Какое действие является наиболее корректным для быстрого восстановления доступа, без снижения общего уровня безопасности? Дайте развернутый ответ, используя правильную терминологию из материала.

Задание 4. Тактика реагирования на атаку

В логи корпоративной IPS компании «ВостокСтрой» (Владивосток) поступило 15 записей за 2 минуты о срабатывании различных сигнатур (от сканирования портов до попыток SQL-инъекции), и все — с одного внешнего IP-адреса 203.0.113.5. Стандартное действие для каждой из этих сигнатур в IPS настроено как «Тревога (Alert)». Несмотря на это, атаки продолжаются. Какую настройку в политике безопасности IPS должен изменить администратор, чтобы автоматически прекратить эту атаку? Опишите, как должно быть сконфигурировано правило или действие.

Задание 5. Выбор типа IPS

Главный офис банка в Казани переходит на модель работы с удаленными сотрудниками. Критически важные приложения (например, «Клиент-Банк») теперь доступны с личных ноутбуков сотрудников через VPN. Руководство службы безопасности понимает, что сетевой IPS (NIPS) в дата-центре не может анализировать зашифрованный трафик между ноутбуком и VPN-сервером. Какой тип IPS следует дополнительно рассмотреть для защиты самих конечных устройств (ноутбуков) сотрудников? Объясните, в чём заключается ключевое преимущество этого типа IPS в описанном сценарии.

ТЕМА-25. Безопасность локальной сети

Введение

Представьте себе жилой многоквартирный дом. Его безопасность обеспечивается на нескольких уровнях. Сначала есть общая дверь с домофоном (уровень периметра), затем — дверь в вашу конкретную квартиру, а внутри — сейф для самых ценных вещей. Локальная сеть (Local Area Network, LAN) — это и есть такой «цифровой дом», где работают пользователи, хранятся данные и функционируют сервисы. Безопасность этого «дома» нельзя обеспечить только крепкими стенами (сетевым экраном на границе с интернетом). Злоумышленник, оказавшись внутри (физически подключившись к розетке или по Wi-Fi), может свободно перемещаться по подъездам и квартирам, если не приняты дополнительные меры внутри самого здания. Данный материал посвящён именно внутренней безопасности локальной сети, фокусируясь на защите компьютеров пользователей, безопасности на уровне канального доступа (Layer-2), настройке соответствующих механизмов и особенностях защиты беспроводных и голосовых (VoIP) сетей.

1. Обеспечение безопасности пользовательских компьютеров

Пользовательский компьютер (рабочая станция) — это конечная точка сети, «квартира» в нашем цифровом доме. Чаще всего именно она становится целью атак, так как является самым слабым звеном из-за человеческого фактора.

1.1. Базовые принципы защиты конечных точек

Безопасность рабочей станции строится на нескольких фундаментальных принципах, аналогичных правилам личной гигиены и безопасности в реальной жизни.

- **Принцип наименьших привилегий:** Пользователь и программы на его компьютере должны иметь ровно столько прав, сколько необходимо для выполнения рабочих задач, и не более. **Смысл:** Это предотвращает случайное или намеренное изменение критически важных системных файлов, установку нежелательного программного обеспечения или отключение средств защиты. **Пример/аналогия:** В офисе рядовому сотруднику выдают ключ только от своего кабинета и, возможно, от туалета. У него нет ключа от серверной, кабинета директора или сейфа с документацией. Если его ключ (учётная запись) будет скомпрометирован, ущерб

будет локализован. Работа из-под учётной записи администратора для повседневных задач — это как ходить по офису с брелком, на котором висят все ключи от здания.

- **Регулярное обновление программного обеспечения:** Установка актуальных обновлений для операционной системы, прикладных программ и, особенно, антивирусных баз. **Смысл:** Подавляющее большинство атак использует уже известные уязвимости («дыры»), для которых производители выпустили «заплатки» (обновления). **Пример/аналогия:** Ваша входная дверь имеет известный производителю дефект замка, который позволяет вскрыть его отмычкой особой формы. Производитель высылает вам новый, улучшенный сердечник замка (обновление). Если вы его не установите, любой, знающий об этой уязвимости, сможет открыть вашу дверь. «Чернобыль» (WannaCry) 2017 года заразил сотни тысяч компьютеров именно через известную уязвимость в Windows, для которой обновление было выпущено за два месяца до начала эпидемии.
- **Использование антивирусного программного обеспечения (АВПО) и персональных сетевых экранов (брандмауэров): Смысл:** АВПО предназначено для обнаружения, блокировки и удаления вредоносного кода (вирусы, трояны, шпионское ПО). Персональный брандмауэр контролирует сетевую активность приложений, предотвращая несанкционированные соединения «изнутри наружу» (утечка данных) и «снаружи внутрь». **Пример/аналогия:** Антивирус — это иммунная система организма и личный врач, который проверяет всю еду (файлы) и предметы (программы), попадающие в организм, на наличие инфекций. Персональный брандмауэр — это личный охранник у двери вашей квартиры, который проверяет, кому и что вы выносите из дома, и кто стучится к вам в дверь, сверяясь со списком ожидаемых гостей (правилами).
- **Шифрование дисков:** Использование технологий (например, BitLocker в Windows, FileVault в macOS, LUKS в Linux) для шифрования всего содержимого жёсткого диска. **Смысл:** При физической краже или утере устройства (ноутбук, диск) данные останутся недоступны без знания специального ключа (пароля). **Пример/аналогия:** Это превращает все ваши вещи в квартире в нечитаемый набор символов. Даже если вор вынесет весь ваш сейф (жёсткий диск) и вскроет его физически, он найдёт там только бессмысленный набор металлических пластин (зашифрованные данные), а не драгоценности и документы.
- **Осознанность пользователя (кибергигиена):** Самый важный и нетехнический элемент. Включает в себя умение распознавать фишинговые письма, не использовать непроверенные USB-носители, создавать сложные пароли. **Смысл:** Технические средства бессильны, если пользователь сам, обманутый, откроет дверь злоумышленнику. **Пример/аналогия:** Самый совершенный домофон и замок не помогут, если жилец поверит в легенду «я сантехник от управляющей компании» и нажмёт кнопку открытия двери подъезда. Фишинговое письмо, маскирующееся под уведомление от банка с просьбой «срочно подтвердить данные карты» — это и есть цифровой «лжесантехник».

2. Соображения по безопасности второго уровня (Layer-2)

Чтобы понять угрозы на втором уровне, необходимо вспомнить основы модели OSI. Канальный уровень (Layer-2, L2) отвечает за передачу данных между непосредственно соединёнными сетевыми устройствами в рамках одной локальной сети. Его основная адресация — это MAC-адреса (физические, «железные» адреса сетевых карт), а ключевое

устройство — коммутатор (switch). Если на сетевом уровне (L3) мы думаем об IP-адресах и маршрутизаторах, то на L2 мы думаем о «коммутационной таблице» и портах коммутатора. Безопасность L2 часто упускается из виду, что делает её «ахиллесовой пятой» многих сетей.

2.1. Почему безопасность L2 критически важна? Аналогия с почтовой службой.

Представьте, что коммутатор — это сортировочный центр почтовой службы в маленьком городе. У каждого жителя (устройства) есть уникальный физический адрес (MAC-адрес). Центр (коммутатор) изучает, с какого участка (порта) приходят письма от каждого адреса, и ведёт журнал: «Иванов И.И. (MAC-A) — участок №1 (порт 1), Петров П.П. (MAC-B) — участок №2 (порт 2)».
Когда приходит письмо для Иванова, центр смотрит в журнал и отправляет его только на участок №1. Это эффективно и конфиденциально. Вот что происходит при атаках на L2:

- **Угроза 1: Переполнение коммутационной таблицы (MAC Flooding).** Злоумышленник (на участке №3) начинает отправлять в центр тысячи открыток с фальшивыми обратными адресами (поддельные MAC-адреса). Журнал центра переполняется, и он перестаёт понимать, где кто находится. В панике, чтобы письма не потерялись, центр превращается в простой «ретранслятор»: он начинает рассылать все приходящие письма **на все участки сразу**. Теперь письмо для Иванова получает не только он, но и Петров, и злоумышленник. Конфиденциальность потеряна. В сетевых терминах коммутатор деградирует до режима концентратора (hub), и злоумышленник может «прослушивать» (sniff) весь трафик.
- **Угроза 2: Подмена MAC-адреса (MAC Spoofing).** Злоумышленник отправляет в центр письмо с обратным адресом «Иванов И.И.» (поддельный MAC-адрес), но со своего участка №3. Центр обновляет журнал: «Иванов И.И. — теперь участок №3». Всё почту для Иванова теперь получает злоумышленник. Это атака «отравления» ARP-таблицы (ARP poisoning/spoofing). ARP — это протокол, который как раз переводит IP-адреса в MAC-адреса. Отравив ARP-таблицы, атакующий может встать «между» жертвой и маршрутизатором, перехватывая и даже изменяя весь её трафик (атака «человек посередине», Man-in-the-Middle, MitM).
- **Угроза 3: Подмена порта коммутатора (Switch Port Stealing).** Используется в сочетании с MAC Spoofing. Атакующий быстро чередует фреймы с реальным MAC-адресом жертвы и своим поддельным, заставляя коммутатор постоянно переключать порт назначения. В моменты, когда коммутатор связывает MAC жертвы с портом атакующего, трафик идёт к нему.
- **Угроза 4: Атаки на spanning-tree protocol (STP).** STP предотвращает образование петель в сети с избыточными связями между коммутаторами. Злоумышленник может подключить свой коммутатор и объявить себя «корневым мостом» с наивысшим приоритетом. Вся сеть перенастроит свои пути через него, отдав атакующему весь трафик для прослушивания.

Смысл защиты L2: Если не контролировать канальный уровень, все вышележащие защиты (сложные пароли, шифрование на уровне приложений) могут быть обойдены. Атакующий,

получивший доступ к L2, становится «невидимым» с точки зрения IP-адресов и может действовать изнутри сети с минимальными препятствиями.

3. Конфигурация безопасности второго уровня

К счастью, современные управляемые коммутаторы предоставляют ряд инструментов для противодействия описанным угрозам. Их настройка — прямая обязанность системного администратора.

3.1. Порт безопасности (Port Security)

Это базовый и мощный механизм. Он позволяет зафиксировать, какой MAC-адрес (или адреса) могут работать на конкретном физическом порту коммутатора.

- **Как работает:** Администратор настраивает порт, например, в режиме «sticky» — коммутатор автоматически запоминает первый подключившийся MAC-адрес и привязывает его к этому порту. Или вручную задаёт разрешённый MAC-адрес.
- **Действия при нарушении (Violation Mode):**
  - **Protect:** Фреймы с неразрешённых MAC-адресов тихо отбрасываются, без уведомлений.
  - **Restrict:** Фреймы отбрасываются, и генерируется сообщение-ловушка (SNMP trap) для администратора.
  - **Shutdown (рекомендуемый по умолчанию):** Порт полностью отключается (переходит в состояние «err-disabled»). Это самое строгое действие, которое немедленно останавливает атаку, но требует ручного вмешательства для включения порта.
- **Пример/аналогия:** Это система контроля доступа на предприятии. Каждому сотруднику (устройству) выдают пропуск (MAC-адрес), который работает только на одном определённом турникете (порт коммутатора). Если кто-то попытается пройти с этим пропуском через другой турникет или подделать пропуск, турникет заблокируется и вызовет охрану (режим shutdown). Это предотвращает подключение несанкционированных ноутбуков в розетки в переговорных комнатах.

3.2. DHCP Snooping

DHCP — это протокол автоматической раздачи IP-адресов. Злоумышленник может развернуть свой «злой» DHCP-сервер в сети и раздавать клиентам свой IP-адрес в качестве шлюза по умолчанию, перенаправляя весь их трафик через себя.

- **Как работает:** На коммутаторе включается функция DHCP Snooping. Все порты классифицируются как **доверенные (trusted)** (те, к которым подключены легитимные DHCP-серверы) и **недоверенные (untrusted)** (порты пользователей). Коммутатор «прослушивает» DHCP-трафик, строит базу данных о соответствии IP, MAC, порта и времени аренды. Сообщения от DHCP-сервера, пришедшие на недоверенный порт (т.е. от поддельного сервера), немедленно блокируются.

- **Пример/аналогия:** В городе есть официальная служба выдачи адресов (DHCP-сервер). Мошенник открывает свою контору и начинает раздавать фальшивые адреса, указывая свой дом как «главную площадь города» (шлюз). Городская стража (DHCP Snooping) знает, где находится официальная контора (доверенный порт). Все объявления из других мест (недоверенные порты) о выдаче адресов признаются незаконными и уничтожаются.

3.3. Dynamic ARP Inspection (DAI)

Эта функция защищает от ARP-спуфинга (подмены MAC-адресов), но для своей работы **полностью зависит от базы данных, созданной DHCP Snooping** (или статически настроенных записей).

- **Как работает:** Коммутатор проверяет все ARP-запросы и ответы, проходящие через недоверенные порты. Он сверяет пару «IP-адрес — MAC-адрес» из ARP-пакета с доверенной базой данных от DHCP Snooping. Если пакет содержит несоответствующую информацию (например, кто-то заявляет, что IP-адрес шлюза теперь имеет его MAC), такой пакет отбрасывается.
- **Пример/аналогия:** Продолжая аналогию с городом: злоумышленник (ARP-спуфер) ходит по улицам и кричит: «Всем слушать! Теперь я — мэр! Все официальные обращения несите ко мне в дом (поддельный MAC)!». Но городская стража (DAI) сверяется с официальным реестром жителей (база DHCP Snooping), где записан настоящий адрес мэра. Услышав ложное объявление, стража арестовывает (отбрасывает) мошенника.

3.4. Защита протокола spanning-tree (STP Guard)

- **Root Guard:** Настраивается на портах, которые **не должны** становиться корневыми. Если на такой порт приходит предложение (BPDU) с более высоким приоритетом (претензия на корневой мост), порт блокируется для протокола STP, но остаётся работающим для пользовательского трафика.
- **BPDU Guard:** Чаще используется на портах, ведущих к конечным пользователям, где BPDU-фреймы не должны появляться в принципе. Если на порт с включённым BPDU Guard приходит любой BPDU-фрейм, порт немедленно отключается (err-disabled). Это мгновенно реагирует на попытку подключения несанкционированного коммутатора или атаки на STP.
- **Пример/аналогия:** В корпоративной иерархии есть официальный генеральный директор (корневой мост). Root Guard — это инструкция для всех вице-президентов (критические uplink-порты): «Если к вам придет человек, заявляющий, что он новый гендир с лучшими полномочиями, не верьте ему и изолируйте его, но продолжайте работать». BPDU Guard — это правило для рядовых сотрудников (порты пользователей): «Если к вам на стол вдруг ляжет приказ, подписанный «генеральный директор», немедленно вызывайте охрану и блокируйте кабинет (отключайте порт)».

4. Безопасность беспроводных сетей, VoIP

4.1. Безопасность беспроводных сетей (Wi-Fi)

Беспроводная сеть радикально расширяет периметр защиты. Сигнал доступен за физическими стенами здания. Ключевые концепции:

- **Аутентификация:** Процесс проверки легитимности пользователя/устройства, желающего подключиться.
  - **WPA2/WPA3 — Personal (PSK):** Используется общий ключ (парольная фраза) для всей сети. **Уязвимость:** Пароль знают многие, его можно подобрать или перехватить при установке соединения. Сравнимо с одним ключом от подъезда для всех жителей — если ключ украден, безопасность всех под угрозой.
  - **WPA2/WPA3 — Enterprise:** Используется отдельный сервер аутентификации (обычно RADIUS). Каждый пользователь подключается со своим уникальным логином и паролем (часто учётной записью домена). **Смысл и преимущество:** Максимальная безопасность. Учётные записи можно централизованно отзывать. Каждому пользователю генерируется уникальный сеансовый ключ шифрования. Если устройство сотрудника утеряно, достаточно заблокировать его учётную запись в AD, и доступ будет прекращён для этого устройства, но не для всей сети. **Аналогия:** Это полноценный КПП с охранником, проверяющим у каждого служебное удостоверение (логин/пароль или сертификат) по базе данных. У каждого сотрудника после проверки — свой уникальный пропуск на сегодня (сеансовый ключ).
- **Шифрование:** Процесс кодирования передаваемых данных. WPA2 использует стандарт AES-CCMP, который считается криптографически стойким. Старые WEP и TKIP — взломаны и небезопасны. WPA3 вносит дальнейшие улучшения, защищая даже от офлайн-подбора паролей.
- **Скрытие SSID, фильтрация по MAC-адресу:** Это **неэффективные** меры безопасности. SSID передаётся в открытом виде при подключении, его легко обнаружить. MAC-адрес легко подделать (MAC spoofing). Эти методы создают лишь видимость защиты и не должны рассматриваться как серьёзные.
- **Сегментация сети:** Беспроводные гости должны быть помещены в отдельную VLAN, полностью изолированную от корпоративной сети. Это предотвращает доступ гостей к внутренним ресурсам.

4.2. Безопасность VoIP (Voice over IP)

IP-телефония совмещает в себе уязвимости голосовой связи и IP-сетей. Основные угрозы:

- **Перехват разговоров (сниффинг):** Если голосовой трафик передаётся в открытом виде (без шифрования), его можно перехватить и прослушать, как и любой другой сетевой трафик.
- **Подмена номера (Caller ID Spoofing):** Атакующий может подделать номер звонящего, чтобы выдать себя за доверенное лицо (техподдержку, начальника).
- **Атаки на доступность (DoS/DDoS):** Наводнение VoIP-сервера или IP-телефонов фальшивыми запросами, приводящее к отказу в обслуживании — невозможности совершать/принимать звонки.

- **Vishing (голосовой фишинг):** Социальная инженерия по телефону с использованием подмены номера.

Меры безопасности для VoIP:

- **Сегментация (VLAN для VoIP):** Выделение голосового трафика в отдельную VLAN. Это не только улучшает качество связи (QoS), но и изолирует его от данных, затрудняя перехват.
- **Шифрование сигнального (SIP) и медиапотока (RTP):** Использование протоколов вроде SRTP (Secure RTP) и TLS для SIP. **Смысл:** Даже если трафик будет перехвачен, он будет представлять собой зашифрованную белиберду.
- **Аутентификация телефонов и пользователей:** IP-телефоны должны аутентифицироваться на сервере (по логину/паролю или сертификату), чтобы исключить подключение несанкционированных устройств.
- **Настройка брандмауэров для VoIP:** Грамотная конфигурация правил, разрешающих только необходимый для работы VoIP трафик и блокирующую аномальную активность.

Заключение

Безопасность локальной сети — это многослойная оборона, где нельзя пренебрегать ни одним уровнем. Защита пользовательских компьютеров создаёт прочный «иммунитет» конечных точек. Понимание и нейтрализация угроз на канальном уровне (Layer-2) с помощью портов безопасности, DHCP Snooping, DAI и защиты STP лишает атакующего возможности скрытно перемещаться и перехватывать данные внутри самой сети. Наконец, специализированные подходы к безопасности беспроводных и VoIP-сетей учитывают их уникальные характеристики, расширяя и усиливая общую систему защиты. Только комплексное применение всех этих мер позволяет построить отказоустойчивую и безопасную локальную сеть, способную противостоять современным угрозам. Администратор, понимающий эти фундаментальные идеи, может грамотно настроить сетевое оборудование и разработать политики, превращающие сеть из уязвимой «открытой площади» в защищённую «крепость» с контролируемыми точками доступа на всех этажах и во всех комнатах.

Контрольные вопросы к ТЕМЕ-25

1. Назовите основной принцип безопасности, который предписывает предоставлять пользователям и программам ровно столько прав, сколько необходимо для работы, и не более.
2. Почему регулярное обновление операционной системы и программ считается одной из самых важных мер защиты пользовательского компьютера?
3. Какую основную функцию выполняет персональный сетевой экран (брандмауэр) на рабочей станции?

4. Что такое фишинг и почему защита от него зависит в первую очередь от осознанности пользователя?
5. Как называется сетевая атака на втором уровне, при которой злоумышленник отправляет коммутатору огромное количество фреймов с поддельными MAC-адресами, заставляя его работать как концентратор?
6. В чём суть атаки ARP Spoofing (отравления ARP-таблицы) и к чему она может привести?
7. Какие три основных действия (violation mode) может выполнять функция Port Security на коммутаторе при обнаружении неразрешённого MAC-адреса на порту?
8. Для чего используется функция DHCP Snooping и какие два типа портов она предполагает (trusted и untrusted)?
9. Какой механизм безопасности второго уровня защищает сеть от ARP-spoofing, используя для проверки базу данных, созданную DHCP Snooping?
10. В чём ключевое отличие методов аутентификации WPA2-Personal (PSK) и WPA2-Enterprise для беспроводных сетей?
11. Почему такие меры, как скрытие SSID Wi-Fi сети или фильтрация по MAC-адресу, считаются неэффективными для обеспечения реальной безопасности?
12. Какую основную меру безопасности рекомендуется применять для изоляции и защиты трафика IP-телефонии (VoIP) в локальной сети?

Практические задания к ТЕМЕ-25

Задание 1. Анализ инцидента в малом офисе (г. Воронеж)

В небольшой компании «Вектор-Строй» после подключения нового ноутбука подрядчика в свободную розетку в переговорной комнате несколько сотрудников начали жаловаться на перебои с доступом в интернет и странное поведение банковских сайтов. Старший администратор, проверив логи центрального коммутатора, обнаружил, что на одном из портов постоянно меняются привязанные MAC-адреса, а таблица коммутации заполнена до предела. Объясните, какую атаку, вероятно, совершил злоумышленник, используя ноутбук подрядчика? К каким двум последствиям (описанным в материале) это привело? Какую самую строгую меру Port Security (режим violation) нужно было заранее настроить на портах в переговорной, чтобы автоматически предотвратить подобный инцидент?

Задание 2. Планирование обновления Wi-Fi сети в колледже (г. Казань)

В IT-лицее принято решение модернизировать устаревшую беспроводную сеть. Старая сеть использовала стандарт WPA2 с единым паролем, который был написан на доске объявлений и известен всем, включая учащихся и гостей. Администрация хочет обеспечить безопасный доступ для преподавателей (с доступом к внутренним сетевым ресурсам, например, журналу) и отдельный доступ для учащихся и гостей в интернет. Опираясь на изученный материал, предложите два ключевых технических решения, которые необходимо реализовать. Первое — по типу аутентификации, второе — по логической структуре сети. Объясните одним предложением преимущество вашего решения для аутентификации преподавателей по сравнению со старым.

Задание 3. Расследование странных звонков (г. Екатеринбург)

В офисе компании «УралТрансЛогистика» сотрудники отдела бухгалтерии начали получать телефонные звонки (по корпоративной IP-телефонии) от якобы «специалиста технической поддержки». Звонивший убедительно просил сообщить логины и пароли для «проведения срочных работ». При проверке логов VoIP-сервера выяснилось, что вызовы поступали с внутренних номеров, которые на тот момент были не зарегистрированы в системе. Какую базовую уязвимость в настройке безопасности VoIP это выявило? Какую технологию второго уровня, изучаемую в рамках безопасности локальной сети, можно было бы применить к портам, к которым подключаются IP-телефоны, чтобы аппаратная подделка номера (имитация телефона) стала практически невозможной?

Задание 4. Выбор стратегии для филиала банка (г. Сочи)

Администратору филиала «Южного Банка» в Сочи поставлена задача максимально обезопасить внутреннюю сеть операционных касс от несанкционированного прослушивания трафика. Риском считается как внешний злоумышленник, так и недобросовестный сотрудник. Физический доступ к сетевым розеткам в зале для клиентов сведён к минимуму. Основываясь на материале о безопасности второго уровня, предложите два взаимосвязанных механизма (функции), которые нужно настроить на банковских коммутаторах. Эти механизмы должны в комплексе предотвратить две основные угрозы: 1) установку неавторизованного устройства для прослушивания, 2) перехват или подмену сетевых адресов между легитимными устройствами (например, между рабочей станцией кассира и сервером). Назовите эти два механизма и кратко (одной фразой) поясните, что каждый из них блокирует.

Задание 5. Локализация проблемы с доступом в интернет (г. Нижний Новгород)

В call-центре компании «ВолгаСвязь» после плановой замены коммутатора часть сотрудников не смогла получить IP-адрес по DHCP и вышла в интернет, используя статические настройки (APIPA). Другая часть работает нормально. Подозрение пало на неисправность нового оборудования. Однако, более опытный коллега, вспомнив про изученную теорию, предположил, что проблема может быть в конфигурации безопасности. Он проверил настройки и обнаружил, что на всех портах, где возникла проблема, включена функция безопасности, которая сверяет ARP-ответы с доверенной базой, а на порту, куда подключен DHCP-сервер, не включена критически важная для работы этой базы опция. Какую функцию безопасности, блокирующую ARP-ответы, скорее всего, включили на проблемных портах? И какую другую функцию, необходимую для её корректной работы, забыли активировать на порту DHCP-сервера?

ТЕМА-26. Криптографические системы

Введение

Представьте, что вы хотите отправить другу секретное письмо, но знаете, что по дороге его могут перехватить и прочитать. Как сделать так, чтобы даже если письмо попадёт в чужие руки, его содержимое осталось непонятным? Или представьте, что вы получаете сообщение от банка с просьбой перевести деньги — как убедиться, что это действительно банк, а не мошенник? Эти и многие другие проблемы современного цифрового мира решает криптография.

Криптография (от греческих слов "kryptós" — скрытый и "gráphein" — писать) — это наука о методах обеспечения конфиденциальности, целостности и аутентичности информации. Сегодня мы живём в мире цифровых коммуникаций: онлайн-банкинг, электронная почта, мессенджеры, интернет-магазины. Во всех этих системах наша личная и финансовая информация постоянно передаётся по сетям, которые по своей природе не являются безопасными. Криптография создаёт "цифровые сейфы" и "электронные печати", которые защищают информацию в этом небезопасном пространстве.

В рамках этого модуля мы рассмотрим основные криптографические сервисы: обеспечение базовой целостности и аутентичности, конфиденциальность информации, а также познакомимся с революционной технологией — криптографией открытых ключей, которая легла в основу безопасного интернета.

1. Базовые концепции и терминология криптографии

1.1. Основные понятия

Прежде чем углубиться в конкретные методы, давайте определим ключевые термины, которые будут использоваться на протяжении всего материала.

Открытый текст (plaintext) — это исходная, незашифрованная информация. Например, текст вашего сообщения другу: "Встречаемся в 18:00 у фонтана".

Шифротекст (ciphertext) — это результат шифрования, преобразованный текст, который выглядит как бессмысленный набор символов. Например, то же сообщение после шифрования может выглядеть так: "X5gH!9qPmL*3tR@vZ#2".

Шифрование (encryption) — процесс преобразования открытого текста в шифротекст с использованием специального алгоритма и ключа.

Дешифрование (decryption) — обратный процесс преобразования шифротекста обратно в открытый текст.

Ключ (key) — секретная информация, используемая алгоритмом шифрования и дешифрования. Если сравнить шифрование с замком, то ключ — это уникальная комбинация, которая этот замок открывает и закрывает. Важно понимать: безопасность современной криптографии основывается не на секретности алгоритма (большинство алгоритмов общедоступны), а на секретности ключа.

Алгоритм шифрования (cipher) — математическая функция или набор правил, которые определяют, как именно выполняется преобразование текста.

Криптоанализ — наука о методах взлома шифров, то есть о способах дешифрования информации без знания ключа. Состязание между криптографами (создателями шифров) и криптоаналитиками (взломщиками) — это движущая сила развития криптографии на протяжении всей её истории.

1.2. Историческая аналогия: от Цезаря к цифровой эре

Чтобы понять эволюцию идей, давайте совершим краткий экскурс в историю.

Шифр Цезаря (пример симметричного шифрования) — один из древнейших известных шифров. Юлий Цезарь использовал его для переписки со своими генералами. Алгоритм прост: каждая буква в сообщении заменяется на букву, находящуюся на фиксированное число позиций дальше в алфавите. Например, при сдвиге на 3 позиции:

- A → D
- B → E
- C → F
- ...
- "ПРИВЕТ" → "ТУЛЗИХ" (в русском алфавите)

Здесь ключом является число 3 (сдвиг). Отправитель и получатель заранее договариваются об этом ключе. Чтобы прочитать сообщение, получатель выполняет обратную операцию — сдвигает буквы на 3 позиции назад.

Проблема: Как безопасно передать ключ? Если вы можете безопасно передать ключ генералу, почему бы не передать сразу всё сообщение? Эта дилемма — **проблема распределения ключей** — тысячелетиями ограничивала применение криптографии только теми случаями, когда стороны могли встретиться лично.

Ситуация кардинально изменилась только в 1970-х годах с появлением **криптографии с открытым ключом**, которая позволила решить проблему распределения ключей. Об этом революционном прорыве мы подробно поговорим в разделе 4.

2. Криптографические сервисы: цели защиты информации

Криптография предоставляет не одну, а несколько взаимосвязанных услуг (сервисов) по защите информации. Давайте рассмотрим каждую из них, используя понятные аналогии из реальной жизни.

2.1. Конфиденциальность (Confidentiality)

Определение: Обеспечение доступности информации только тем субъектам, которые имеют на это право. Проще говоря, конфиденциальность — это **секретность**. Никто посторонний не должен прочитать ваше сообщение.

Аналогия: Закрытое письмо в сейфе. Вы пишете письмо, кладёте его в сейф (шифрование) и отправляете получателю. По дороге сейф могут украсть, вскрыть, потрясти, но без ключа (который есть только у получателя) прочитать письмо невозможно. Даже если злоумышленник перехватит сейф, информация останется защищённой.

Пример из IT: Когда вы вводите пароль на сайте, он должен передаться на сервер. Если соединение не защищено, пароль может "подсмотреть" кто-то в вашей Wi-Fi сети. Протокол HTTPS (где S означает Secure) использует шифрование, чтобы превратить ваш пароль в шифротекст перед отправкой. Даже если данные перехватят, увидеть пароль не удастся.

Техническая суть: Конфиденциальность достигается путём **шифрования** данных с использованием криптографического ключа. Только тот, кто обладает правильным ключом для **дешифрования**, может восстановить исходную информацию.

2.2. Целостность (Integrity)

Определение: Гарантия того, что информация не была изменена несанкционированным образом. Важно не только скрыть содержание, но и быть уверенным, что его не подменили по дороге.

Аналогия: Пломба на контейнере. Вы отправляете ценный груз в опечатанном контейнере. Получатель, прежде чем вскрывать, проверяет пломбу. Если пломба цела — груз не вскрывали и не подменяли. Если пломба повреждена — значит, в пути что-то пошло не так, и содержимому доверять нельзя. Целостность здесь подтверждается состоянием пломбы, а не замка (конфиденциальности). Контейнер может быть и прозрачным — главное, чтобы его не вскрывали.

Пример из IT: Вы скачиваете с официального сайта драйвер для видеокарты. Как убедиться, что хакер не подменил файл на своём сервере вирусом? Разработчик вычисляет для оригинального файла специальную криптографическую "контрольную сумму" (хеш) и публикует её на своём сайте. После скачивания вы самостоятельно вычисляете хеш для

полученного файла и сравниваете с официальным. Если они совпадают — файл цел и невредим.

Техническая суть: Целостность обеспечивается с помощью **криптографических хеш-функций** и **кодов аутентификации сообщений (MAC)**, которые создают для данных уникальный "цифровой отпечаток". Любое минимальное изменение данных приводит к совершенно другому "отпечатку".

2.3. Аутентичность (Authenticity)

Определение: Подтверждение подлинности источника информации. Это ответ на вопросы: "Кто это отправил?", "Действительно ли это сообщение пришло от заявленного отправителя?".

Аналогия: Подпись и печать на документе. Вы получаете приказ о повышении зарплаты. Бумага формата А4, текст напечатан. Как убедиться, что это не шутка коллеги? Вы смотрите на подпись директора и фирменную печать. Вы доверяете документу, потому что доверяете этим атрибутам, подтверждающим его подлинность. Аутентичность — это про установление авторства.

Пример из IT: Вы получаете email от банка с ссылкой для "восстановления пароля". Сообщение идеально имитирует фирменный стиль банка. Как понять, фишинг это или нет? Современные системы электронной почты (при правильной настройке) используют криптографические методы (например, DKIM, SPF), которые позволяют почтовому серверу проверить, действительно ли письмо пришло с серверов указанного банка, а не с поддельного адреса.

Техническая суть: Аутентичность обеспечивается с помощью механизмов **электронной подписи** и **кодов аутентификации**, которые неразрывно связывают сообщение с секретным ключом отправителя. Проверить подлинность может любой, у кого есть соответствующий открытый ключ отправителя.

Важное замечание: Часто целостность и аутентичность обеспечиваются вместе. Если вы можете проверить, что сообщение пришло от конкретного человека (аутентичность), и оно не менялось (целостность), это даёт гарантию **аутентичности источника данных**.

3. Базовая целостность и аутентичность: криптографические хеш-функции

3.1. Что такое хеш-функция?

Криптографическая хеш-функция — это математический алгоритм, который принимает на вход данные любого размера (файл, текст, видео) и возвращает короткую, фиксированной длины строку байтов, называемую **хешем**, **дайджестом** или "цифровым отпечатком".

Представьте себе гигантский мясорубку. Вы загружаете в неё целую книгу (входные данные любого размера). На выходе получается небольшая, однородная фаршевая котлетка фиксированного размера (хеш). Попытка загрузить даже немного другую книгу приведёт к получению совершенно другой котлетки.

Свойства идеальной криптографической хеш-функции:

1. **Детерминированность:** Одни и те же входные данные всегда дают одинаковый хеш.
   - *Пример:* Файл report.pdf всегда будет иметь хеш a1b2c3d4..., независимо от того, кто и на каком компьютере его вычисляет.
2. **Высокая скорость вычисления:** Получить хеш от данных должно быть computationally легко и быстро.
3. **Эффект лавины:** Минимальное изменение входных данных (даже один бит) должно радикально менять хеш.
   - *Пример:*
     - Хеш от фразы "Я сдам экзамен на 5": 5f4dcc3b5aa765d61d8327deb882cf99
     - Хеш от фразы "Я сдам экзамен на 4": 7c6a180b36896a0a8c02787eeafb0e4c
     - Изменилась одна цифра (и соответствующий ей байт в кодировке), а хеши стали абсолютно непохожими.
4. **Необратимость (свойство односторонней функции):** По имеющемуся хешу **невозможно** восстановить исходные данные. Из котлетки нельзя восстановить оригинальную книгу.
   - Это защищает, например, пароли. Системы хранят не пароли пользователей, а их хеши. При вводе пароля система вычисляет его хеш и сравнивает с хранимым. Даже если злоумышленник украдёт базу хешей, он не сможет напрямую узнать пароли.
5. **Устойчивость к коллизиям:** Практически невозможно найти два разных набора входных данных, которые дадут одинаковый хеш.
   - *Аналогия:* Уникальность отпечатков пальцев. Теоретически у двух разных людей могут быть одинаковые отпечатки, но на практике вероятность этого исчезающе мала. Так же и с хешами.

3.2. Практическое применение хешей для проверки целостности

Сценарий: Скачивание дистрибутива операционной системы (например, Ubuntu).

1. **На стороне разработчика:** После сборки образа ubuntu-24.04.iso вычисляется его хеш с помощью алгоритма SHA-256. Полученная строка, например a1b2...c3d4, публикуется на официальном сайте.

2. **На стороне пользователя:** Вы скачали файл ubuntu-24.04.iso. Чтобы проверить, не был ли файл повреждён при загрузке или подменён злоумышленником, вы вычисляете его хеш той же функцией SHA-256.
3. **Сравнение:** Вы сравниваете вычисленный хеш с тем, что опубликован на сайте.

- **Совпали** — файл идеально цел, это точная копия того, что выложили разработчики.
- **Не совпали** — файл повреждён или изменён. **Категорически нельзя** его использовать, так как он может содержать вирусы или бэкдоры.

Этот процесс гарантирует **целостность данных**. Однако он пока не гарантирует **аутентичность**, потому что злоумышленник мог подменить и файл, *и хеш* на сайте-двойнике. Для решения этой проблемы нужна электронная подпись.

3.3. От целостности к аутентичности: HMAC

Для одновременной проверки целостности и аутентичности источника данных (когда отправитель и получатель имеют общий секретный ключ) используется схема **HMAC (Hash-based Message Authentication Code — код аутентификации сообщения на основе хеша)**.

Как это работает (упрощённо):

1. Отправитель и получатель заранее разделяют секретный ключ K.
2. Отправитель, чтобы отправить сообщение M, вычисляет не просто хеш H(M), а HMAC = H(K + M) (на самом деле процесс сложнее, но суть в объединении ключа и сообщения перед хешированием).
3. Отправитель отправляет пару: (M, HMAC).
4. Получатель, получив сообщение, самостоятельно вычисляет HMAC' от полученного M, используя тот же секретный ключ K.
5. Если вычисленный HMAC' совпадает с полученным HMAC, это доказывает две вещи:

- **Целостность:** M не изменилось (иначе хеш бы не сошёлся).
- **Аутентичность:** Сообщение пришло от того, кто знает секретный ключ K (посторонний, не знающий K, не смог бы сгенерировать правильный HMAC).

Аналогия: Представьте, что вы с коллегой делите секретную фразу "синий слон". Вы отправляете документ и на последней странице пишете не просто свою подпись, а свою подпись + "синий слон", и всё это запечатываете в конверт с лаком. Коллега, получив документ, видит подпись и фразу. Он знает, что только вы двое знаете фразу, значит, документ точно от вас и не менялся. Это HMAC.

Ограничение: HMAC требует, чтобы у сторон **уже был общий секретный ключ**. Для установления связи с незнакомой стороной (например, при подключении к сайту банка) используется более мощный инструмент — **криптография с открытым ключом и электронная подпись**.

4. Конфиденциальность: симметричное шифрование

Конфиденциальность — это классическая задача криптографии: спрятать смысл сообщения от посторонних. Основной механизм её обеспечения — **шифрование**. Мы начнём с более древней и интуитивно понятной модели — **симметричного шифрования**.

4.1. Принцип симметричного шифрования

В симметричных криптосистемах для **шифрования** и **дешифрования** используется **один и тот же секретный ключ** (или ключи, которые легко выводятся один из другого). Отсюда и название — "симметричное".

Схема работы:

1. Отправитель (Алиса) и получатель (Боб) **заранее и безопасным способом** договариваются об общем секретном ключе K.
2. Алиса шифрует своё открытое сообщение M с помощью ключа K и алгоритма шифрования E. Получается шифротекст C = E(K, M).
3. Алиса отправляет шифротекст C по незащищённому каналу (например, интернету).
4. Боб получает C и применяет операцию дешифрования D с тем же ключом K, чтобы восстановить исходное сообщение: M = D(K, C).
5. Противник (Ева), перехвативший C, не сможет восстановить M, не зная ключа K.

Аналогия: Настоящий замок с ключом. Один и тот же ключ (K) закрывает (шифрует) и открывает (дешифрует) дверь сейфа (сообщение M). Чтобы передать что-то ценное, вы должны сначала лично встретиться и обменяться дубликатами ключа.

4.2. Современные алгоритмы симметричного шифрования: AES

Исторические шифры (вроде шифра Цезаря) легко взломать. Современные стандарты — это сложные математические конструкции, проверенные мировым криптографическим сообществом.

AES (Advanced Encryption Standard — усовершенствованный стандарт шифрования) — самый распространённый и надёжный на сегодня симметричный алгоритм. Принят в качестве стандарта правительством США в 2001 году после открытого конкурса.

- **Ключи:** AES поддерживает ключи длиной 128, 192 или 256 бит. Чем длиннее ключ, тем выше стойкость.
  - *Пример масштаба:* Ключ в 128 бит — это строка из 16 символов (байт). Количество возможных комбинаций: 2^128 ≈ 3.4 * 10^38. Если представить, что каждый атом во Вселенной (их ~10^80) может проверять миллиард ключей в секунду, на перебор всех комбинаций уйдёт время, многократно превышающее возраст Вселенной. Безопасность AES

основана не на секретности алгоритма (он открыт), а на нереальности полного перебора (brute-force).

- **Блочный шифр:** AES работает с блоками данных фиксированного размера (128 бит = 16 байт). Если сообщение больше, оно разбивается на блоки и шифруется по частям с использованием специальных **режимов шифрования** (CBC, GCM и др.), которые добавляют случайность и защищают от шаблонов.
- **Применение:** AES используется повсеместно: в протоколе HTTPS (TLS/SSL), для шифрования архивов (ZIP, RAR), виртуальных дисков (BitLocker, FileVault), в беспроводных сетях (WPA2) и многих других системах.

4.3. Проблема симметричной криптографии

Главный недостаток симметричных систем — **проблема безопасного распределения ключей**. Как Алисе и Бобу, находящимся в разных странах, впервые договориться об общем секретном ключе K, не подвергая его риску перехвата?

Представьте, что у вас нет безопасного канала. Если вы отправите ключ по email, его перехватят. Если позвоните и продиктуете — прослушают. Эта фундаментальная проблема сдерживала развитие криптографии до 1970-х годов. Решением стала **криптография с открытым ключом**.

5. Криптография открытых ключей (асимметричная криптография)

Этот раздел описывает революцию в криптографии. В 1976 году Уитфилд Диффи и Мартин Хеллман предложили принципиально новую концепцию, которая позволила решить проблему распределения ключей.

5.1. Основная идея: два ключа вместо одного

В асимметричных криптосистемах у каждого участника есть не один, а **пара математически связанных ключей**:

1. **Открытый ключ (Public Key):** Этот ключ **не является секретным**. Его можно свободно публиковать, рассылать, размещать на сайте — как номер телефона в телефонной книге. Его функция — **для шифрования** информации, предназначенной владельцу.
2. **Закрытый ключ (Private Key):** Этот ключ хранится в строжайшем секрете его владельцем. Его функция — **для дешифрования** информации, зашифрованной с помощью соответствующего открытого ключа.

Ключевой принцип: То, что зашифровано открытым ключом, может быть расшифровано **только** соответствующим закрытым ключом. И наоборот, в контексте

подписи (см. ниже), данные, подписанные закрытым ключом, могут быть проверены соответствующим открытым ключом.

5.2. Аналогия для понимания

Представьте себе специальный **замок с защёлкой**, который захлопывается, но его нельзя открыть без ключа.

1. **Генерация пары:** Вы создаёте для себя такой замок. У вас есть сам замок (открытый ключ) и единственный ключ от него (закрытый ключ).
2. **Распространение открытого ключа:** Вы производите тысячи копий своего замка (открытого ключа) и раздаёте всем желающим: друзьям, коллегам, размещаете на сайте.
3. **Шифрование (отправка секретного сообщения):** Кто-то (Алиса) хочет отправить вам секретное послание. Она кладёт его в коробку, захлопывает коробку **вашим** замком (вашим открытым ключом) и отправляет вам. После щелчка замка открыть коробку **невозможно** без ключа. Даже Алиса, которая только что закрыла её, не может открыть.
4. **Дешифрование (получение сообщения):** Коробка доходит до вас. Только у вас есть уникальный ключ (ваш закрытый ключ) от всех таких замков. Вы открываете коробку ключом и читаете сообщение.

Самое главное: Алисе **не нужно было заранее договариваться с вами о секрете!** Ей достаточно было знать ваш открытый (публичный) ключ-замок. Безопасный канал для предварительного обмена ключами больше не требуется.

5.3. Математическая основа (очень упрощённо)

Магия двух ключей основана на **односторонних функциях с потайным ходом** (trapdoor functions). Это операции, которые легко выполнить в одну сторону, но чрезвычайно сложно — в обратную, если не знать некий секрет ("потайной ход").

Классическая аналогия — смешивание красок:

- **Легко (прямое преобразование):** Смешать синюю и жёлтую краску, чтобы получить зелёную.
- **Очень сложно (обратное преобразование):** Имея только зелёную краску, разделить её на исходные синюю и жёлтую компоненты.

В криптографии с открытым ключом (например, в алгоритме RSA) используются задачи из теории чисел:

- **Легко:** Перемножить два больших простых числа p и q (получить N = p * q). Простые числа — это числа, делящиеся только на себя и на 1 (например, 17, 53, 101).
- **Чрезвычайно сложно (без знания p и q):** Имея только огромное число N (которое может иметь 600 и более цифр), разложить его на исходные простые множители p и q.

Открытый и закрытый ключи генерируются на основе этих чисел p, q и N.
Знание p и q (закрытый ключ) позволяет легко выполнять операции дешифрования, в то время как злоумышленник, знающий только N (открытый ключ), столкнётся с нерешаемой на практике задачей факторизации.

5.4. Решение проблемы распределения ключей: протокол Диффи-Хеллмана

Первый и красивейший пример использования асимметричной идеи — это не шифрование, а **безопасный обмен ключами**. Протокол Диффи-Хеллмана позволяет двум сторонам, общающимся по незащищённому каналу, создать общий секретный ключ, который никто из подслушивающих вычислить не сможет.

Аналогия с красками (классическая):

1. Алиса и Боб договариваются об **общем публичном цвете** (например, жёлтый). Это не секрет.
2. **Секретные цвета:** Алиса выбирает свой секретный цвет (скажем, красный). Боб выбирает свой секретный цвет (скажем, синий). Они никому их не говорят.
3. **Публичный обмен:** Алиса смешивает общий жёлтый цвет со своим секретным красным и получает оранжевую смесь. Отправляет её Бобу. Боб смешивает общий жёлтый со своим секретным синим и получает зелёную смесь. Отправляет её Алисе. Перехватчик Ева видит только оранжевую и зелёную краски.
4. **Формирование общего секрета:** Алиса получает зелёную смесь от Боба и добавляет в неё **свой секретный красный** цвет. Получается **коричневая** краска. Боб получает оранжевую смесь от Алисы и добавляет в неё **свой секретный синий** цвет. **Он тоже получает коричневую краску!**
5. **Результат:** Алиса и Боб получили одинаковый цвет (коричневый), который и будет их **общим секретным ключом**. Ева же видела только оранжевый и зелёный. Чтобы получить коричневый, ей нужно было бы "разделить" одну из этих смесей, что аналогично нерешённой математической задаче (дискретному логарифмированию).

Таким образом, не передавая по сети ни одного общего секрета, Алиса и Боб смогли создать его. Этот общий секрет теперь можно использовать как ключ для **быстрого симметричного шифрования** (например, AES).

5.5. Обеспечение аутентичности и целостности: электронная цифровая подпись (ЭЦП)

Асимметричная криптография решает не только проблему конфиденциальности, но и проблему аутентичности/целостности в открытой среде.

Принцип: Если данные можно зашифровать открытым ключом, а расшифровать закрытым, то возможна и обратная операция: **зашифровать (подписать) закрытым ключом, а расшифровать (проверить) открытым**. Поскольку закрытый ключ известен только

владельцу, успешная проверка открытым ключом доказывает, что данные были подписаны именно им.

Процесс создания и проверки подписи:

1. **Подписание (отправитель):**
   - Отправитель (Алиса) вычисляет хеш H от своего сообщения M.
   - Алиса **шифрует** этот хеш H своим **закрытым ключом**. Получается **цифровая подпись** S.
   - Алиса отправляет получателю пару: (M, S).
2. **Проверка (получатель):**
   - Получатель (Боб) получает (M, S).
   - Боб вычисляет хеш H' от полученного сообщения M той же хеш-функцией.
   - Боб **расшифровывает** подпись S с помощью **открытого ключа Алисы** (который у всех есть). Если подпись верна, расшифровка даст исходный хеш H.
   - Боб сравнивает H (из подписи) и H' (вычисленный им). Если они совпадают, это доказывает:

- **Аутентичность:** Подпись S мог создать только тот, у кого есть закрытый ключ Алисы (то есть сама Алиса).
- **Целостность:** Хеши совпали, значит, сообщение M не изменилось с момента подписания.

Аналогия: Вы ставите на документ уникальную печать, которую можно проверить под специальным публично доступным увеличительным стеклом. Любой, у кого есть это стекло (открытый ключ), может убедиться, что печать подлинная (аутентичность) и документ под ней не менялся (целостность), но не может создать такую же печать без секретного штампа (закрытого ключа).

ЭЦП — основа доверия в цифровом мире: подписание программного обеспечения, юридически значимых документов, транзакций с криптовалютами и т.д.

6. Комбинирование технологий: реальные протоколы (на примере HTTPS/SSL/TLS)

На практике все три рассмотренных сервиса (конфиденциальность, целостность, аутентичность) и оба типа криптографии (симметричная и асимметричная) работают вместе. Ярчайший пример — протокол **HTTPS**, защищающий ваше общение с сайтами.

Когда вы в адресной строке видите **https://** и значок замка, происходит примерно следующее (сильно упрощённо):

1. "Рукопожатие" (Handshake) с использованием асимметричной криптографии:

- Ваш браузер подключается к серверу, например, банка.
- Сервер отправляет браузеру свой **SSL-сертификат**, который содержит его **открытый ключ** и информацию о домене, подписанную доверенным **Удостоверяющим Центром**

(ЦА). Браузер проверяет эту подпись с помощью открытого ключа ЦА (встроенного в браузер), убеждаясь, что сервер действительно принадлежит банку, а не мошенникам (**аутентификация сервера**).

- Браузер генерирует случайный **сеансовый ключ** (session key) для симметричного шифрования.
- Браузер **шифрует** этот сеансовый ключ **открытым ключом сервера** и отправляет его на сервер.
- Только сервер, обладая своим **закрытым ключом**, может **расшифровать** и получить сеансовый ключ.

2. Установление безопасного канала с симметричным шифрованием:

- Теперь и у браузера, и у сервера есть общий секретный сеансовый ключ, безопасно переданный с помощью асимметричного шифрования.
- **Все дальнейшее общение** (передача паролей, номеров карт, личных сообщений) шифруется **симметричным алгоритмом** (обычно AES) с использованием этого быстрого сеансового ключа. Это обеспечивает **конфиденциальность

Контрольные вопросы к ТЕМЕ-26

1. Что такое конфиденциальность в контексте криптографических систем и какой бытовой пример её иллюстрирует?
2. Чем отличается целостность данных от их аутентичности? Приведите пример ситуации, где важно именно обеспечение целостности.
3. Что такое криптографическая хеш-функция и какое её ключевое свойство предотвращает восстановление исходных данных из хеша?
4. Как с помощью хеш-функции можно убедиться, что скачанный с интернета файл не был повреждён или изменён?
5. Объясните принцип работы симметричного шифрования. В чём заключается его главная практическая проблема?
6. Почему алгоритм AES считается надёжным, если его принципы работы общедоступны?
7. В чём состоит фундаментальное отличие криптографии с открытым ключом от симметричной криптографии?
8. Опишите своими словами аналогию с замком и ключом, объясняющую принцип работы шифрования с открытым ключом.
9. Какую проблему, неразрешимую для симметричных систем, удалось решить с появлением криптографии открытых ключей?
10. Что такое электронная цифровая подпись и какие два свойства информации она позволяет одновременно проверить?
11. Почему в реальных системах (как HTTPS) используется комбинация симметричного и асимметричного шифрования, а не только один тип?
12. Как в протоколе HTTPS решается задача безопасной передачи сеансового ключа для симметричного шифрования?

Практические задания к ТЕМЕ-26

Задание 1. Анализ инцидента с ПО

Вы — системный администратор в колледже г. Тюмени. Преподаватель информатики пожаловался, что после установки драйвера для интерактивной доски, скачанного с сайта board-driver.ru, на его компьютере появились признаки вирусной активности. На официальном сайте производителя доски (китайской компании) для этого драйвера указан хеш SHA-256: 8f3a2b...c9d1e5. Вы скачали тот же файл с board-driver.ru и вычислили его хеш: a4b5c6...d7e8f9.

- **Вопрос:** О чём свидетельствует несовпадение хешей? Какие два криптографических сервиса (целостность, конфиденциальность, аутентичность) были нарушены в этой ситуации и почему? Что бы вы рекомендовали делать в таких случаях?

Задание 2. Планирование безопасного обмена

Администрации небольшого филиала банка в г. Казани необходимо наладить ежедневный безопасный обмен закрытыми отчетными файлами между менеджером филиала и головным офисом в Москве. Объём файлов большой (сотни мегабайт). Руководство рассматривает два варианта:

1. Купить специализированную программу, которая использует шифрование AES.
2. Использовать бесплатное ПО с поддержкой PGP (технология, основанная на шифровании открытым ключом).

- **Вопрос:** Какой из вариантов, вероятно, будет работать быстрее для ежедневной передачи больших файлов и почему? С какой практической проблемой столкнутся сотрудники при первом подключении, если выберут вариант 2 (PGP), и как её можно решить, не встречаясь лично?

Задание 3. Имитация атаки «злоумышленник посередине»

Вы проводите инструктаж для сотрудников техподдержки на примере упрощённой схемы. Представьте ситуацию: клиент (Алиса) хочет безопасно получить от сервера (Боб) секретный номер сессии, используя идею асимметричного шифрования.

- **Шаг 1:** Боб должен прислать Алисе свой открытый ключ.
- **Шаг 2:** Алиса шифрует этим ключом номер сессии и отправляет его Бобу.
- Вы выступаете в роли злоумышленника (Ева) и можете перехватывать и подменывать сообщения.
- **Вопрос:** Какое единственное сообщение вам нужно перехватить и подменить, чтобы в итоге и Алиса, и Боб думали, что общаются друг с другом, но секретный номер сессии стал известен вам? Поясните схему ваших действий.

Задание 4. Выбор механизма защиты для служебной переписки

В IT-отделе российской компании «Ромашка» решили защитить внутреннюю переписку по электронной почте от подделки. Требуется гарантировать, что если сотрудник получил

письмо от начальника отдела с инструкцией, то он может быть уверен: 1) письмо точно отправил начальник, 2) текст письма не менялся после отправки.

- **Вопрос:** Какой криптографический механизм (хеширование, HMAC, электронная подпись на основе открытого ключа) лучше всего подходит для этой задачи? Объясните свой выбор, указав, почему два других механизма не идеальны в данном контексте (опирайтесь на их свойства из теоретического материала).

Задание 5. Диагностика проблемы с веб-сайтом

Пользователь из бухгалтерии жалуется, что не может зайти на сайт nalog.gov.ru (ФНС России) через браузер. Браузер выдаёт ошибку «Небезопасное соединение» или «Сертификат недействителен». При этом другие сайты, включая gosuslugi.ru, открываются нормально.

- **Вопрос:** Какой этап установления защищённого HTTPS-соединения, скорее всего, не проходит? На обеспечение какого из трёх криптографических сервисов (конфиденциальность, целостность, аутентичность) напрямую влияет эта ошибка? Что, исходя из теоретического материала, является «паспортом» сервера в этом процессе и кто его «подписывает»?

ТЕМА-27. Управление безопасной сетью

Введение

Представьте себе, что вы архитектор, но строите вы не дома, а цифровые города – компьютерные сети. Сначала вы проектируете планировку улиц (топологию), затем проводите коммуникации (кабели, Wi-Fi), устанавливаете правила движения (протоколы) и, наконец, заселяете жителей (пользователей и данные). Но любой город, даже самый современный, беззащитен перед угрозами: грабителями (хакерами), стихийными бедствиями (аппаратными сбоями) или внутренними беспорядками (ошибками сотрудников). **Управление безопасной сетью** – это искусство и наука проектирования, построения и постоянной защиты такого «цифрового города», чтобы он функционировал стабильно, а его «жители» и их «имущество» (данные) были в безопасности. Это не просто установка антивируса или фаервола. Это целостная система взглядов, процессов и правил, которая начинается с идеи на чертеже и не заканчивается никогда, сопровождая сеть на протяжении всей её жизни.

---

1. Принципы безопасности сетевого дизайна

Прежде чем прокладывать первую витую пару, необходимо заложить в проект фундаментальные правила безопасности. Эти принципы – аксиомы, которые нельзя нарушать.

- **Принцип наименьших привилегий:** Это самый важный принцип. Каждому пользователю, программе или устройству должны предоставляться **ровно те права и доступ, которые необходимы для выполнения их конкретных задач, и не больше.** Это как выдача ключей в офисе: уборщице нужен ключ от кладовки с инвентарём, но не от сейфа в бухгалтерии; бухгалтеру – доступ к финансовой программе, но не к настройкам межсетевого экрана. **Пример:** Пользователю из отдела маркетинга не нужен доступ к серверу с зарплатными ведомостями. Сетевая служба, отвечающая за печать, не должна иметь права удалять файлы на файловом сервере.
- **Глубокоэшелонированная оборона (Defense in Depth):** Нельзя полагаться на один рубеж защиты (например, только фаервол). Безопасность должна быть многослойной, как матрёшка или крепостные стены. Если злоумышленник преодолеет один барьер, его должен встретить следующий. **Аналогия:** Защита замка. Это и ров с водой (сетевой экран), и высокие стены (система обнаружения вторжений, IDS), и внутренняя цитадель (шифрование данных), и стража, проверяющая всех внутри (антивирус на рабочих станциях, политики паролей). Даже если враг переплывёт ров, у него ещё много препятствий впереди.

- **Разделение обязанностей и сегментация:** Большую плоскую сеть, где все устройства «видят» друг друга, очень опасно. Её нужно делить на логические сегменты (подсети, VLAN), изолируя друг от друга разные отделы, типы трафика или уровни доверия. **Пример:** В больнице сеть аппаратов МРТ и систем контроля лекарств должна быть изолирована от сети для гостевого Wi-Fi посетителей. В компании бухгалтерия, отдел кадров и склад должны находиться в разных сетевых сегментах. Это как водонепроницаемые отсеки на корабле – если один отсек (сегмент) будет затоплен (скомпрометирован), вода не хлынет во все остальные, и корабль (вся сеть) останется на плаву.
- **Простота и управляемость:** Сложная система безопасности, в которой администратор сам путается, опаснее, чем её отсутствие. Конфигурации должны быть максимально простыми, стандартизированными и документированными. Если для решения одной задачи есть 10 сложных правил вместо одного понятного, возрастает риск ошибки при настройке и при последующем обслуживании. **Аналогия:** Замок с тысячью уникальных, хитроумных ключей для каждой двери в здании будет казаться надёжным, но потеря одного главного мастер-ключа или уход одного слесаря, который знал все хитрости, приведёт к катастрофе. Лучше иметь меньше, но стандартных и надёжных замков.
- **Отказоустойчивость и избыточность:** Система безопасности не должна быть «единой точкой отказа». Если какой-то защитный компонент выйдет из строя, это не должно парализовать всю сеть или полностью обнулить защиту. **Пример:** Использование двух фаерволов в режиме кластера: если один «упадёт», второй мгновенно и незаметно для пользователей возьмёт на себя всю нагрузку. Или наличие «холодного» запасного сервера аутентификации на случай сбоя основного.

2. Безопасная архитектура

Это практическое воплощение принципов дизайна в конкретные схемы, модели и схемы размещения устройств. Это «генеральный план» вашего безопасного цифрового города.

- **Модель «Доверенная/Недоверенная зона»:** Классическая и фундаментальная модель. Вся сеть делится на зоны по уровню доверия.
  - **Внутренняя (доверенная) сеть:** Сеть компании, где находятся сотрудники, серверы. Защищается периметровыми средствами.
  - **Демилитаризованная зона (DMZ):** Полудоверенная буферная зона. Сюда выносятся сервисы, которые должны быть доступны извне: веб-сервер, почтовый шлюз. Из интернета можно попасть только в DMZ, а оттуда во внутреннюю сеть – крайне ограниченно и под строгим контролем. Это как контрольно-пропускной пункт на границе. Грузы (данные) сначала проверяются на таможне (в DMZ), и только потом, если они разрешены, могут попасть внутрь страны (во внутреннюю сеть).
  - **Внешняя (недоверенная) сеть:** Интернет. Источник большинства угроз.
- **Zero Trust (Нулевое доверие):** Современная эволюция подхода. Девиз: «Никому не верь, проверяй всегда». В модели Zero Trust нет изначально доверенной внутренней сети. **Каждая попытка доступа к любому ресурсу (файлу, приложению, серверу) проверяется, независимо от того, откуда она исходит – из интернета или изнутри офиса.** Доступ

предоставляется на основе строгой аутентификации (например, многофакторной), проверки устройства (соответствует ли оно политикам безопасности) и контекста (в какое время, с какого места пользователь пытается подключиться). **Пример:** Сотрудник в офисе хочет открыть общий файл. В классической модели ему бы сразу разрешили доступ, раз он «внутри». В Zero Trust его запрос будет проверен: его личность подтверждена по отпечатку и SMS-коду, его ноутбук проверен на наличие актуальных обновлений и антивируса, и только после этого доступ к файлу будет открыт.

- **Схемы размещения защитных устройств:** Куда именно ставить фаервол, IPS/IDS, прокси-сервер? Это вопрос архитектуры.
  - **Периметровая защита:** Фаервол стоит на границе между интернетом и сетью компании (или DMZ). Как главные ворота города.
  - **Распределённая защита:** Помимо главного фаервола, ставятся дополнительные на границах между критически важными сегментами внутренней сети (например, между сетью отдела разработки и финансов). Это внутренние КПП в городе.
  - **Защита на уровне хоста:** Антивирус, персональный фаервол на каждом компьютере и сервере. Это как личная охрана и крепкие двери у каждого жителя города.

---

3. Управление процессами и безопасность

Безопасность – это не только технологии, но и **процессы**, то есть чётко описанные, повторяемые последовательности действий. Самый надёжный фаервол можно обесточить, если кто-то запустит вредонос с флешки, найденной на парковке.

- **Управление учётными записями:** Процесс создания, выдачи, изменения и удаления учётных записей. Уволен сотрудник – его учётная запись должна быть немедленно заблокирована, а не висеть месяцами. Это как сбор пропуска при увольнении.
- **Управление изменениями (Change Management):** Любое изменение в сети (установка обновления, новая настройка маршрутизатора) должно проходить через стандартный процесс: заявка -> утверждение -> тестирование в изолированной среде -> запланированное внедрение -> документирование. Это предотвратит ситуации, когда «что-то сломалось» из-за спонтанных действий администратора.
- **Управление инцидентами:** Что делать, если безопасность всё же нарушена? Должен быть план: обнаружение -> анализ -> локализация (отключение заражённого сегмента) -> ликвидация угрозы -> восстановление -> анализ причин и извлечение уроков. Без такого процесса паника приведёт к большим потерям.
- **Управление правами доступа:** Регулярный аудит – кто, к чему и когда имеет доступ. Это как ежегодная инвентаризация ключей и пропусков в организации.

4. Тестирование сети на уязвимости

Вы построили защиту, но как проверить, что она работает? Для этого нужны регулярные проверки, как учебные тревоги или учения МЧС.

- **Аудит безопасности (Security Audit):** Комплексная проверка на соответствие стандартам и политикам. Аудиторы смотрят конфигурации, интервьюируют сотрудников, проверяют документацию.
- **Сканирование уязвимостей (Vulnerability Scanning):** Автоматизированная проверка всех устройств в сети на наличие известных «дыр» (уязвимостей). Сканер, как робот с чек-листом, обходит сеть и сверяет версии ПО, настройки с базой известных проблем (CVE). **Пример:** Сканер обнаруживает, что на одном из серверов стоит устаревшая версия веб-сервера Apache с критической уязвимостью, через которую можно удалённо выполнить код. Администратор получает отчёт и знает, что этот сервер нужно срочно обновить.
- **Тестирование на проникновение (Penetration Testing, Pentest):** Это уже не автоматическое сканирование, а моделирование реальной атаки этичными хакерами (белыми шляпами). Их цель – не просто найти дыру, а **использовать её**, чтобы проникнуть в сеть и получить доступ к критичным данным, доказав тем самым реальную опасность. **Аналогия:** Сканирование уязвимостей – это как инспектор, который ходит по дому и трясёт все двери, проверяя, хорошо ли они закрыты. Тестирование на проникновение – это как тот же инспектор, который, найдя слабый замок на окне, вскрывает его, заходит в дом, открывает сейф и оставляет записку: «Ваши драгоценности могли быть у меня. Замените замок на окне и усильте защиту сейфа».

---

5. Непрерывность бизнеса, планирование восстановления аварийных ситуаций

Безопасность – это не только защита от злоумышленников, но и обеспечение работоспособности бизнеса в любых условиях. Сеть – это кровеносная система компании. Если она откажет, бизнес умрёт.

- **План обеспечения непрерывности бизнеса (Business Continuity Plan, BCP):** Всеобъемлющий план, как компании продолжать работать в условиях серьёзного сбоя: кибератаки, пожара, наводнения. Он охватывает не только IT, но и связь, персонал, поставки. **Пример:** При масштабной DDoS-атаке на основной сайт BCP предписывает переключить трафик на резервную площадку, активировать коммуникацию с клиентами через социальные сети и горячую линию.
- **План аварийного восстановления (Disaster Recovery Plan, DRP):** Конкретная, техническая часть BCP, касающаяся восстановления IT-инфраструктуры и данных после катастрофы. Основные понятия:
  - **RPO (Recovery Point Objective):** Допустимая точка возврата. Сколько данных мы можем позволить себе потерять? Если RPO = 4 часа, значит, резервное копирование должно

выполняться не реже чем каждые 4 часа, и в случае аварии мы потеряем данные, созданные за последние 4 часа.

- **RTO (Recovery Time Objective):** Допустимое время восстановления. Как быстро система должна быть снова доступна? Если RTO = 2 часа, значит, у нас есть 2 часа с момента аварии до момента, когда критичные сервисы снова заработают.
- **Аналогия:** Ваш дом (серверная) сгорел. RPO определяет, как часто вы делали копии семейного фотоархива (данные) и оставили их в банковской ячейке. Если копия была неделю назад (RPO=7 дней), вы потеряете все фото за последнюю неделю. RTO определяет, через какое время после пожара вы сможете жить и работать: если вы сняли квартиру на следующий день (RTO=1 день), бизнес (ваша жизнь) почти не пострадает; если искали жильё месяц (RTO=30 дней), последствия катастрофичны.

---

6. Жизненный цикл сети и планирование

Сеть – это не статичный объект, а живой организм, который рождается, развивается, стареет и требует замены. Управление безопасностью должно сопровождать её на всех этапах.

- **Этапы жизненного цикла:**

1. **Планирование и проектирование:** Определение потребностей, разработка безопасной архитектуры (самый важный этап для закладки фундамента безопасности).
2. **Внедрение и развёртывание:** Закупка, установка и настройка оборудования и ПО с соблюдением политик безопасности.
3. **Эксплуатация и обслуживание:** Повседневная работа: мониторинг, установка обновлений, управление пользователями, реагирование на инциденты. Самый долгий этап.
4. **Модернизация и развитие:** Плановое обновление устаревающих компонентов, масштабирование.
5. **Вывод из эксплуатации:** Безопасное списание оборудования. Важно: простое удаление данных недостаточно. Жёсткие диски должны быть физически уничтожены или стёрты специальным ПО, чтобы данные нельзя было восстановить. Старый маршрутизатор с сохранёнными конфигурациями, выброшенный на свалку, – это готовый план проникновения для злоумышленника.

- **Планирование мощностей:** Постоянный мониторинг загрузки каналов связи, процессоров серверов, места на дисках, чтобы заранее спланировать и закупить ресурсы до того, как их нехватка приведёт к сбоям. **Пример:** Если график нагрузки интернет-канала показывает, что через 3 месяца он будет постоянно заполнен на 95%, нужно уже сейчас планировать его расширение, чтобы не ждать, когда скорость упадёт и бизнес начнёт терять деньги.

7. Разработка регламентов компании и политик безопасности

Это «конституция» и «уголовный кодекс» вашего цифрового города. Технологии без правил бесполезны, так как люди – самое слабое звено.

- **Политика информационной безопасности (ПИБ):** Верховный документ, утверждаемый руководством. Он задаёт цели, распределяет ответственность, декларирует серьёзное отношение компании к безопасности.
- **Регламенты и процедуры:** Конкретные инструкции, которые объясняют **как** выполнять требования политики.
  - **Регламент использования паролей:** Минимальная длина (12 символов), сложность (буквы верхнего/нижнего регистра, цифры, спецсимволы), срок действия (смена каждые 90 дней), запрет на повторное использование.
  - **Регламент работы с персональными данными:** Кто, как и при каких условиях имеет право обрабатывать данные клиентов и сотрудников.
  - **Регламент использования интернета и электронной почты:** Какие сайты запрещены, можно ли открывать вложения от неизвестных отправителей, правила пересылки конфиденциальной информации.
  - **Регламент удалённой работы:** Требования к домашнему компьютеру для подключения к корпоративной сети (установленный антивирус, включённый фаервол), обязательное использование VPN.
- **Политика допустимого использования (AUP):** Документ, который подписывает каждый сотрудник при приёме на работу. Он простым языком объясняет, что можно и чего нельзя делать с корпоративными IT-ресурсами. Нарушение AUP может быть основанием для дисциплинарного взыскания. **Пример пункта AUP:** «Запрещается устанавливать на рабочий компьютер непроверенное программное обеспечение, скачанное из интернета. Все программы должны быть согласованы с IT-отделом».

Заключение

Управление безопасной сетью – это непрерывный циклический процесс: **Спроектировал -> Построил -> Защитил (настроил политики) -> Проверил (протестировал) -> Проанализировал (оценил риски и инциденты) -> Улучшил (внёс изменения) -> И снова спроектировал…** Это комплексная дисциплина, которая стоит на трёх китах: **Люди** (обучение, регламенты), **Процессы** (управление изменениями, инцидентами) и **Технологии** (фаерволы, шифрование). Игнорирование любого из этих компонентов делает всю систему уязвимой. Современный сетевой администратор – это не просто «настройщик роутеров», а инженер-архитектор, юрист-регламентатор и аналитик-стратег в одном лице, от компетенции которого напрямую зависит устойчивость и успех всего бизнеса в цифровом мире.

Контрольные вопросы к ТЕМЕ-27

1. Назовите и кратко поясните принцип, который гласит, что пользователь или программа должны получать ровно тот доступ, который необходим для выполнения задачи, и не больше.
2. В чём заключается суть стратегии «глубокоэшелонированной обороны» (Defense in Depth) при построении безопасной сети? Приведите простую аналогию.
3. Что такое сегментация сети и для чего она применяется? Приведите пример из теоретического материала.
4. Как называется буферная зона сети, куда обычно размещают серверы, доступные из интернета (например, веб-сервер)?
5. В чём основное отличие современной модели безопасности «Zero Trust» (Нулевое доверие) от классической модели с «доверенной внутренней сетью»?
6. Как называется процесс автоматизированной проверки сети на наличие известных программных «дыр» и устаревших версий ПО?
7. Чем отличается **сканирование уязвимостей** от **тестирования на проникновение (Pentest)** по своей основной цели?
8. Что означает аббревиатура RTO в контексте планирования аварийного восстановления? Дайте простое определение.
9. Что означает аббревиатура RPO в контексте планирования аварийного восстановления? Дайте простое определение.
10. На каком этапе жизненного цикла сети закладывается её фундаментальная безопасная архитектура?
11. Какой верховный документ в компании задаёт общие цели и распределяет ответственность в области защиты информации?
12. Как называется конкретный документ (правило), который подписывает сотрудник и который регламентирует, что можно и нельзя делать с корпоративным компьютером и интернетом?

Практические задания к ТЕМЕ-27

1. Ситуация в филиале «Восток-Лес» (г. Красноярск).

Администратор Сергей настроил в небольшой сети (25 компьютеров) общий доступ к папке «Бухгалтерия» для всех пользователей, чтобы «было удобнее». Вскоре произошёл инцидент: случайно удалились важные файлы с финансовой отчётностью, а установить, кто именно это сделал, невозможно. Объясните Сергею, какой **фундаментальный принцип безопасности сетевого дизайна** он нарушил. Как, следуя этому принципу, нужно было правильно настроить доступ к папке? Опишите логику настройки прав для бухгалтера Анны и менеджера по продажам Ивана.

2. Планирование в компании «УралТранс» (г. Екатеринбург).

Директор хочет понять необходимость покупки второго, резервного межсетевого экрана. Он говорит: «У нас уже есть один хороший фаервол, зачем тратиться на второй?». Подготовьте для директора краткое обоснование, используя понятие **«единой точки отказа»** и принцип **«глубокоэшелонированной обороны»**. Приведите простую аналогию из сферы транспорта или логистики, понятную для него как руководителя транспортной компании (например, про запасные мосты или пути).

3. Инцидент в клинике «Здоровье+» (г. Казань).

После визита пациента, который воспользовался гостевым Wi-Fi, в локальной сети клиники начал распространяться вирус. Вирус попал на компьютер администратора регистратуры, но, к счастью, не затронул сервер с медицинскими картами и лабораторным оборудованием. Объясните, благодаря какому **архитектурному решению** (принципу) удалось избежать катастрофы и изоляции каких именно сетевых зон не произошло. Нарисуйте схематично (словесно опишите), как должна быть устроена сеть клиники, чтобы гостевой Wi-Fi, компьютеры администрации и медицинское оборудование были разделены.

4. Дилемма в IT-отделе «ЖилКомСервис» (г. Волгоград).

Администратор Мария предлагает ежеквартально проводить автоматическое сканирование сети на уязвимости. Её коллега Алексей считает, что этого мало, и настаивает на ежегодном полноценном тестировании на проникновение силами сторонних специалистов, что дороже. Объясните, почему оба специалиста по-своему правы. В чём **ключевое отличие** между двумя предлагаемыми ими типами проверок? Какой проверкой можно обнаружить, что пароль «Qwerty123» на одном из роутеров позволяет получить из интернета полный контроль над ним?

5. Вопрос от руководства фабрики «Берёзка» (г. Кострома).

После случая сшибания электропитания и простоя производства на 8 часов, генеральный директор поручил IT-специалисту разработать план восстановления. Он задал два прямых вопроса: «Сколько данных мы максимально можем потерять?» и «За какое время критичные системы (конвейер, складской учёт) должны снова заработать?». Какие **две ключевые метрики (показатели)** должен определить и обосновать IT-специалист в ответе на эти вопросы? Дайте их точные названия и расшифруйте аббревиатуры. Приведите пример значений этих метрик для системы складского учёта, если резервное копирование данных проводится каждую ночь в 23:00, а договорённость с цехом такова, что простой системы более чем на 4 рабочих часа недопустим.