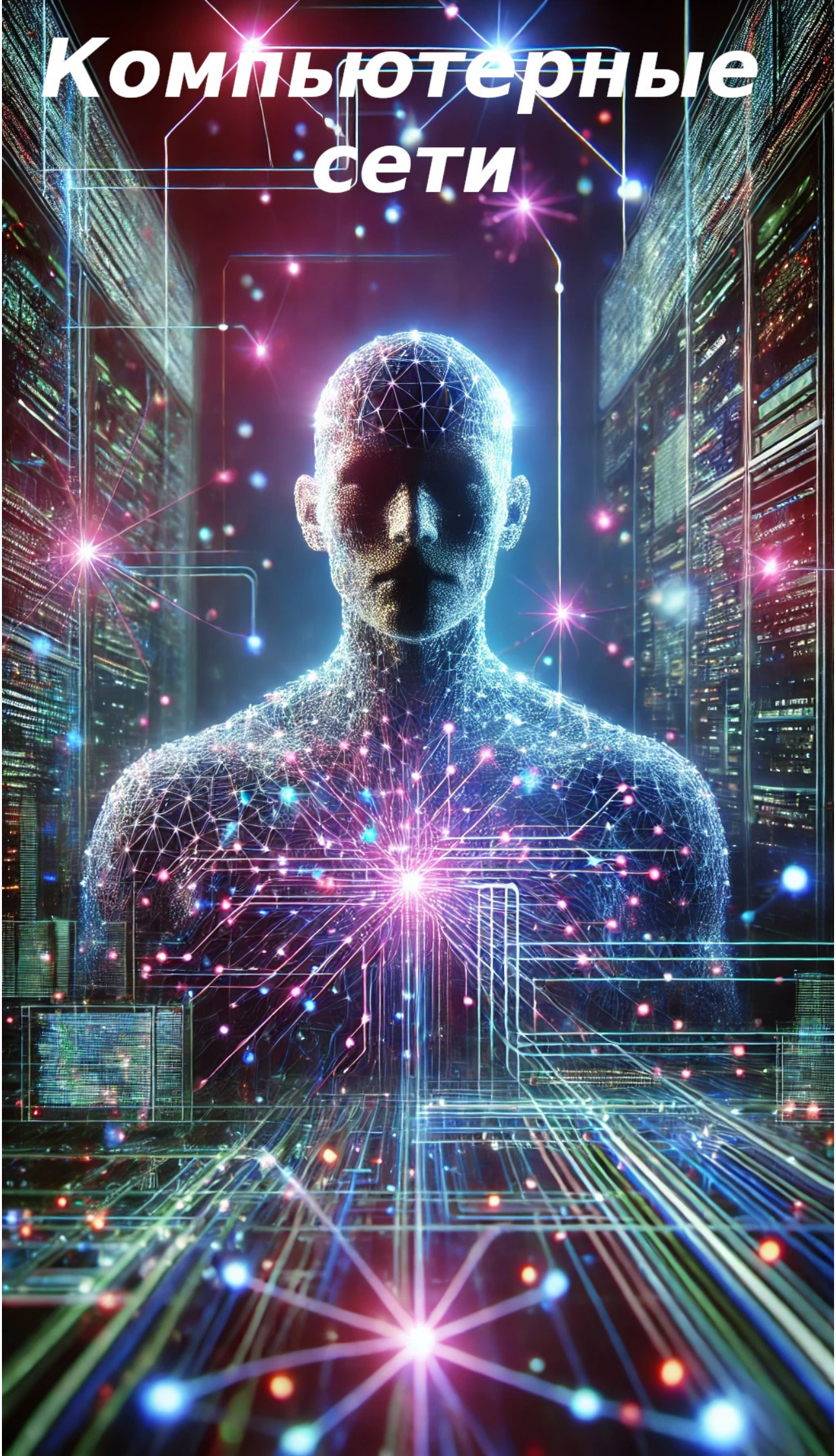
Компьютерные сети



By Aleksei Mihailovich A.
and ChatGPT.

Первое издание.
ZФЛГ Лето ҒƉ CMZX

Оглавление.

- **Ошибки и контроль целостности:**
  Методы обнаружения и коррекции ошибок на канальном уровне.

6. Сетевой уровень: маршрутизация и IP-адресация

- **IP-адресация:**
  Структура IPv4 и введение в IPv6, понятия подсетей и масок.
- **Маршрутизация:**
  Основные принципы выбора маршрута, понятие маршрутизаторов и алгоритмов маршрутизации.
- **Протоколы маршрутизации:**
  Обзор таких протоколов, как RIP, OSPF, BGP – их задачи и особенности.

7. Транспортный уровень: TCP и UDP

- **TCP (Transmission Control Protocol):**
  Как устанавливается соединение, принципы контроля ошибок, управление потоком и надежная доставка данных.
- **UDP (User Datagram Protocol):**
  Отличия от TCP, ситуации, в которых применяется UDP (например, трансляция видео, онлайн-игры).
- **Сравнение TCP и UDP:**
  Примеры, где выбор того или иного протокола имеет значение для работы приложений.

8. Прикладной уровень: протоколы и сервисы

- **Основные прикладные протоколы:**
  HTTP/HTTPS для веб-трафика, FTP для передачи файлов, SMTP и POP3/IMAP для электронной почты.
- **DNS (Domain Name System):**
  Принцип работы системы доменных имен и ее роль в упрощении доступа к ресурсам сети.
- **Другие сервисы:**
  Протоколы для обмена сообщениями, стриминга и других современных задач.

9. Основы безопасности компьютерных сетей

- **Угрозы и риски:**
  Обзор основных кибератак (DDoS, MITM, фишинг) и их влияния на сети.
- **Методы защиты:**
  Шифрование, VPN, файрволы – что это такое и как они работают.
- **Аутентификация и управление доступом:**
  Способы идентификации пользователей и защита информации в сети.

10. Мониторинг, управление и диагностика сети

- **Инструменты для диагностики:**
  Программы и утилиты (ping, traceroute, Wireshark) для анализа работы сети.
- **Протоколы мониторинга:**
  SNMP, NetFlow – как администраторы отслеживают состояние сети.
- **Практические случаи:**
  Разбор типичных проблем и способов их решения.

11. Современные тенденции и перспективы развития сетей

- **Интернет вещей (IoT):**
  Как подключение бытовых устройств меняет ландшафт сетевых технологий.
- **Программно-определяемые сети (SDN):**
  Концепция управления сетью с помощью программного обеспечения.
- **Будущее мобильных сетей:**
  Роль 5G и перспективы дальнейшего развития технологий связи.

12. Заключение и рекомендации для дальнейшего изучения

- **Подведение итогов:**
  Основные выводы и ключевые понятия, которые необходимо усвоить.
- **Рекомендации для практики:**
  Советы по самостоятельному экспериментированию, настройке небольших сетей, участию в форумах и сообществах.
- **Ресурсы и литература:**
  Список книг, онлайн-курсов, статей и видеоуроков для углубленного изучения темы.

От автора.

Данная книга является моим продолжением изучения компьютеров. После того, как я изучил с вами в общих чертах устройство ПК(персонального компьютера), не плохо было бы узнать как они взаимодействуют между собой? Многие люди даже не представляют себя без интернета, что является взаимодействием множества устройств в одной сети.

Свои замечания и пожелания присылайте мне на почту(books_feedback@acypyc.info), буду стараться ответить всем. Надеюсь что данное путешествие будет таким же полезным и интересным для вас как и для меня.

Желаю приятного прочтения!
Быть Добру!
Алексей Михайлович А.

PS

Как и в прошлой книге(Устройство и работа ПК), исторические вырезки приведены на основе общепринятых, т.е. официальных данных. Их вы можете найти в учебниках и энциклопедиях. Это техническая литература, вся информация обработана и собрана с помощью ИИ. Данная книга принесёт вам пользу если вы сфокусируетесь на компьютерных сетях и как они работают.

1. Введение в компьютерные сети

- **Что такое компьютерная сеть:**
  Общее представление, почему сети появились и зачем они нужны.

Компьютерная сеть – это группа устройств (компьютеров, смартфонов, планшетов и серверов), объединённых для обмена информацией. Представьте себе группу людей, которые обмениваются сообщениями, делятся рецептами или важными новостями. Так же и устройства в сети передают данные друг другу — от простых текстовых сообщений до сложных видеозвонков.

Почему появились компьютерные сети и зачем они нужны?

1. **Обмен информацией:**
   Раньше компьютеры работали изолированно. С появлением сетей стало возможно мгновенно передавать файлы, документы, фото и видео, что значительно ускорило работу в бизнесе, науке и повседневной жизни.
2. **Доступ к ресурсам:**
   В сети можно делиться ресурсами, такими как принтеры или серверы для хранения данных. Это позволяет нескольким пользователям использовать одно устройство, экономя время и средства.
3. **Совместная работа:**
   Сети дают возможность людям работать вместе над проектами, даже если они находятся в разных частях мира. Благодаря этому можно проводить видеоконференции, совместно редактировать документы и обмениваться идеями.
4. **Упрощение коммуникации:**
   Электронная почта, мессенджеры и другие средства связи позволяют быстро обмениваться информацией, что помогает оперативно принимать решения и делиться новостями.

Жизненная аналогия – Город:
Представьте себе город, где дома – это компьютеры или другие устройства, а улицы и дороги – каналы связи (кабели, беспроводные сети). Как в городе, где люди перемещаются по улицам, чтобы встретиться с друзьями или попасть на работу, данные в компьютерной сети перемещаются по «улицам» от одного устройства к другому. И как в городе существуют правила дорожного движения, которые помогают организовать движение, в сети существуют протоколы, обеспечивающие упорядоченную и надежную передачу информации.

Таким образом, компьютерные сети позволяют устройствам общаться и работать вместе, как жители города, которые используют дороги и правила для безопасного и эффективного передвижения.

- **История развития сетей:**
  Краткий экскурс от ARPANET до современных технологий.

История компьютерных сетей начинается с экспериментов в 1960-х годах и проходит путь от первых небольших соединений до глобальной сети, которой является Интернет сегодня.

1. ARPANET – зарождение идеи
В 1969 году в США была создана ARPANET – первая сеть, объединявшая компьютеры в нескольких университетах и исследовательских центрах. Ее цель была проста: обмениваться данными и информацией между разными учреждениями, чтобы ученые могли работать сообща и делиться результатами исследований. Можно представить ARPANET как первые «телефонные линии» для компьютеров, которые позволяли им «разговаривать» друг с другом.

2. Протокол TCP/IP(Transmission Control Protocol/Internet Protocol - Протокол управления передачей/Интернет-протокол) – универсальный язык общения
В 1970-х годах появились протоколы TCP/IP, разработанные Винтом Серфом и Бобом Каном. Эти протоколы стали своего рода универсальным языком, позволяющим разным устройствам обмениваться информацией, независимо от их производителя или модели. Благодаря TCP/IP сети стали «понимать» друг друга, и это стало фундаментом для объединения множества небольших сетей в одну большую – Интернет.

3. Появление Интернета и World Wide Web(Всемирная паутина)
К 1980-м годам ARPANET постепенно превратилась в Интернет – глобальную сеть, которая объединяла не только научные учреждения, но и коммерческие организации. В 1989 году Тим Бернерс-Ли предложил идею World Wide Web (WWW), создав систему гипертекстовых ссылок и графический интерфейс, что сделало работу в сети намного удобнее для обычных пользователей. Благодаря этому Интернет стал доступен широким массам.

4. Современные технологии и беспроводные сети
Со временем сети стали стремительно развиваться. Появились беспроводные технологии (Wi-Fi(wireless fidelity - беспроводная точность), мобильный интернет), которые освободили нас от привязки к кабелям. Развиваются облачные сервисы, позволяющие хранить данные и работать с ними через Интернет, а также появляются новые технологии, такие как Интернет вещей (IoT), где даже бытовые приборы могут обмениваться данными.

Итог:
История развития сетей – это путь от первых экспериментов с ARPANET до современной глобальной сети, которая соединяет миллиарды устройств по всему миру. Этот путь сопровождался появлением новых технологий и стандартов, позволяющих сделать обмен информацией быстрым, удобным и доступным для всех.

- **Роль компьютерных сетей в повседневной жизни:**
  Примеры применения в работе, образовании, развлечениях и т.д.

Компьютерные сети играют огромную роль в нашей повседневной жизни, позволяя людям общаться, работать, учиться и отдыхать. Давайте разберёмся, как они это делают:

1. **Работа:**
   Представьте офис, где сотрудники могут обмениваться документами, участвовать в видеоконференциях и работать вместе, даже если находятся в разных городах или странах. Благодаря сетям, вы можете отправлять электронные письма, использовать программы для совместной работы и получать доступ к рабочим ресурсам из любого места.
2. **Образование:**
   Благодаря компьютерным сетям, образование стало доступным для всех. Студенты могут посещать онлайн-курсы, смотреть видеоуроки, участвовать в вебинарах и общаться с преподавателями через интернет. Это похоже на огромную библиотеку и класс в одном: знания доступны, когда вам это нужно.
3. **Развлечения:**
   Сети дают возможность смотреть фильмы, слушать музыку, играть в онлайн-игры и общаться в социальных сетях. Представьте себе кинотеатр или игровую площадку, куда можно зайти в любое время – вот что нам предоставляют современные технологии.
4. **Повседневная жизнь:**
   Интернет помогает нам решать множество ежедневных задач: делать покупки онлайн, пользоваться банковскими услугами, заказывать еду или такси. Это как иметь целый город сервисов и магазинов под рукой, где можно всё найти и купить, не выходя из дома.

Таким образом, компьютерные сети объединяют нас в едином цифровом пространстве, делая жизнь удобнее, интереснее и продуктивнее.

2. Основные понятия и термины

- **Узлы и устройства:**
  Определения терминов «хост», «клиент», «сервер», «компьютер», «устройство».

Давайте представим компьютерную сеть как небольшой город, где каждое здание – это устройство, а улицы – это каналы связи.

- **Узел / Хост:**
  Узел или хост – это любое здание в городе, которое имеет свой адрес и может общаться с другими зданиями. В компьютерной сети узлом может быть компьютер, смартфон, сервер или любой другой прибор, подключённый к сети.
- **Клиент:**
  Клиент – это здание, откуда жители (данные) отправляют запросы или приходят за услугами. Например, когда вы заходите в кафе, вы приходите за кофе – ваше здание выступает в роли клиента, запрашивая услугу.
- **Сервер:**
  Сервер – это здание, которое предоставляет услуги или хранит информацию, к которой обращаются другие здания. Вернёмся к кафе: это кафе (сервер) готовит и подаёт кофе, когда его просят. В сети сервер отвечает на запросы клиентов, предоставляя им нужные данные или ресурсы.
- **Компьютер и устройство:**
  Компьютер – это один из видов устройств, как, например, офисное здание в городе, где происходят различные расчёты, хранится информация и выполняется работа. Однако под устройством понимается любое электронное оборудование, которое может подключаться к сети: это и компьютеры, и смартфоны, и планшеты, и даже принтеры или умные телевизоры.

Таким образом, в нашем городском образе:

- Каждый дом или здание – это **хост** или **узел**.
- Если вы приходите в кафе, чтобы заказать кофе, вы – **клиент**.
- А кафе, где готовят кофе, – это **сервер**.
- И все эти здания – это разные **устройства**, выполняющие свои функции и объединённые сетью, как улицы города.

- **Данные и их передача:**
  Что такое пакеты, кадры, биты и байты; понятие скорости передачи, задержки, пропускной способности.

Давайте представим, что вы хотите отправить длинное письмо другу, но вместо одного огромного конверта вы разбиваете письмо на несколько небольших посылок. Именно так данные перемещаются по сети!

Основные понятия:

- **Бит и байт:**
  - **Бит** – самая маленькая единица информации, которая может быть либо 0, либо 1.
  - **Байт** – группа из 8 бит. Можно думать о байте, как о маленьком слове или кусочке информации, который складывается из восьми «кирпичиков» (битов).
- **Пакеты и кадры:**
  - **Пакет** – это как конверт, в который вы кладёте часть вашего письма (данные). Когда письмо разбито на части, каждая часть отправляется отдельно, а потом на месте назначения все конверты собираются в исходное письмо.
  - **Кадр** – это упаковка на более низком уровне, которая используется внутри одной сети. Представьте, что кадр – это упаковка для передачи пакета по определённой улице, где добавлена информация о том, откуда и куда направляется эта посылка внутри города.

Понятия скорости, задержки и пропускной способности:

- **Скорость передачи:**
  Это как скорость почтовой службы. Она определяет, насколько быстро данные (ваши посылки) могут двигаться по сети от отправителя к получателю. Чем выше скорость, тем быстрее придёт сообщение.
- **Задержка:**
  Это время, которое проходит между отправкой посылки и её получением. Представьте, что задержка – это время ожидания в почтовом отделении, когда посылка находится в пути. Если задержка мала, посылка быстро доставляется; если велика – доставка затягивается.
- **Пропускная способность:**
  Это как вместимость почтового центра или количество дорог в городе. Пропускная способность определяет, сколько данных можно передать одновременно за определённый промежуток времени. Если пропускная способность высокая, можно одновременно отправлять много посылок, и они не будут задерживаться.

Жизненная аналогия – почтовая система:

1. **Ваше письмо:**
   Представляет данные, которые вы хотите передать.
2. **Разбиение письма на посылки:**
   Письмо делится на несколько частей – это как разделение данных на пакеты. Каждая посылка содержит часть информации.

3. **Упаковка каждой посылки:**
   Каждая посылка укладывается в специальный конверт (пакет) с адресом получателя, а внутри этого конверта может быть ещё дополнительная упаковка (кадр) для транспортировки по конкретной части сети.
4. **Доставка посылок:**
   Скорость, с которой почтальоны доставляют посылки, – это скорость передачи данных. Если на дороге возникают пробки, это задержка, а количество посылок, которые можно отправить одновременно – это пропускная способность.

Таким образом, компьютерная сеть передаёт данные так же, как почтовая система доставляет письма: данные разбиваются на маленькие части (пакеты и кадры), передаются с определённой скоростью, могут возникать задержки, и вся система зависит от того, сколько данных можно отправить одновременно.

- **Протоколы и стандарты:**
  Роль протоколов в обеспечении согласованной работы сети.

Представьте, что компьютерная сеть – это большой международный фестиваль, где собраны участники из разных стран. Чтобы все могли общаться, танцевать и веселиться, необходимо, чтобы у всех был общий язык и правила. Вот здесь на помощь приходят протоколы и стандарты.

Что такое протоколы и стандарты?

- **Протоколы** – это набор правил и инструкций, по которым устройства (как люди на фестивале) обмениваются информацией. Они определяют, как правильно «говорить», чтобы собеседник всё понял: какие данные отправлять, в каком формате, и как их обрабатывать.
- **Стандарты** – это общепринятые договорённости, которые делают так, что все устройства, даже если они произведены разными компаниями и в разных странах, работают по одним и тем же правилам. Благодаря стандартам, все участники фестиваля знают, что, например, музыка должна играть на определённой громкости, а танцы – в определённом ритме.

Жизненный пример – фестиваль с участниками разных стран:

1. **Общий язык общения:**
   Представьте, что на фестивале люди говорят на разных языках. Если никто не договорится, как общаться, никто не поймёт друг друга. Протоколы в сети работают как общий язык, например, TCP/IP, который позволяет всем устройствам обмениваться данными, даже если они созданы разными производителями.
2. **Правила поведения:**
   На фестивале установлены правила, чтобы не возникало хаоса: где танцевать, как приветствовать друг друга, где можно купить еду. Точно так же протоколы и стандарты задают порядок в сети. Они помогают устройствам «договариваться», как передавать информацию, чтобы данные не терялись и доходили вовремя.
3. **Согласованная работа:**
   Если все участники фестиваля будут действовать по одним и тем же правилам, мероприятие пройдет успешно. Так и в компьютерной сети: благодаря протоколам и стандартам устройства могут работать вместе, независимо от их типа или производителя. Это обеспечивает надежную и согласованную работу сети.

Таким образом, протоколы и стандарты – это те «правила игры» и «общий язык», которые делают возможным эффективное взаимодействие всех устройств в сети, как на международном фестивале, где благодаря договорённостям все участники могут общаться, сотрудничать и веселиться вместе.

3. Архитектура и классификация сетей

- **Типы сетей:**
  LAN (локальные сети), WAN (глобальные сети), MAN, PAN – особенности и применение.

Давайте представим, что разные типы сетей – это как различные уровни связи в нашем повседневном пространстве, например, в городе.

LAN (Local Area Network - Локальная сеть)

Что это такое:
LAN охватывает небольшую территорию, например, дом, офис или школу. Все устройства в такой сети находятся близко друг к другу и могут быстро обмениваться информацией.

Жизненный пример:
Представьте дом, в котором все комнаты связаны проводами или Wi-Fi. Ваш компьютер, смартфон и принтер подключены к одной сети – это и есть LAN. Это как если бы все комнаты в доме общались между собой напрямую.

WAN (Wide Area Network - Глобальная сеть)

Что это такое:
WAN охватывает большие расстояния – города, страны и даже весь мир. Это сеть, которая соединяет между собой множество локальных сетей.

Жизненный пример:
Подумайте о дорогах и трассах, которые связывают разные города и страны. Интернет – самый большой WAN, где данные перемещаются между различными городами, континентами и даже странами.

MAN (Metropolitan Area Network - Метрополитенская сеть)

Что это такое:
MAN располагается в пределах одного города или крупного района. Это сеть, которая охватывает всю городскую территорию и может соединять, например, все административные учреждения или компании одного города.

Жизненный пример:
Представьте городскую транспортную систему: автобусы, трамваи и метро, которые перемещают людей внутри города. MAN действует подобно этим транспортным средствам, соединяя разные районы одного большого города.

PAN (Personal Area Network - Персональная сеть)

Что это такое:
PAN охватывает очень маленькую территорию – обычно вокруг одного человека. Это сеть, которая соединяет личные устройства, находящиеся рядом.

Жизненный пример:
Подумайте о вашем личном пространстве, где вы носите с собой смартфон, наушники, умные

часы и планшет. Они могут общаться между собой по Bluetooth или Wi-Fi, образуя небольшую сеть вокруг вас – это и есть PAN.

Итоговая аналогия:

- **LAN** – как дом, где все комнаты (устройства) связаны проводами или Wi-Fi.
- **MAN** – как транспортная система города, объединяющая разные районы.
- **WAN** – как сеть дорог, соединяющая разные города и страны.
- **PAN** – как ваше личное пространство, где ваши гаджеты «общаются» между собой.

Таким образом, разные типы сетей созданы для удобства и эффективности обмена информацией на разных уровнях – от домашней атмосферы до глобального масштаба.

- **Топологии сетей:**
  Звезда, шина, кольцо, меш – их преимущества и недостатки.

***Топологией** (компоновкой, конфигурацией, структурой) компьютерной сети называют физическое расположение компьютеров сети друг относительно друга и способ соединения их линиями связи.**

Давайте представим, что у нас есть группа друзей, и мы решаем, как лучше общаться друг с другом, чтобы информация передавалась быстро и без сбоев. Разные способы организации общения между друзьями можно сравнить с топологиями сетей.

---

1. Топология «Звезда»

Как устроена:
Каждый друг напрямую связан с одним центральным человеком (например, организатором встречи), который передаёт сообщения всем остальным.

Преимущества:

- Если один друг теряет связь с организатором, остальные продолжают общаться.
- Легко управлять: все сообщения проходят через одного центрального участника.

Недостатки:

- Если организатор (центральный узел) отпадает, связь между всеми друзьями прекращается.

Жизненный пример:
Представьте офис, где все сотрудники звонят на ресепшн, а оттуда звонят дальше. Если ресепшн не работает, никто не может связаться.

---

2. Топология «Шина»

Как устроена:
Все друзья используют одну общую линию связи – как один длинный телефонный кабель, к которому все подключены.

Преимущества:

- Простота установки и меньшая затратность кабеля.
- Легко добавить нового участника, просто подключив его к этой линии.

Недостатки:

- Если основной кабель ломается, вся сеть оказывается нарушена.
- Чем больше участников, тем медленнее может идти передача сообщений, так как все используют одну линию.

Жизненный пример:
Это как если бы все друзья сидели вдоль одной длинной скамейки и передавали записки по цепочке. Если скамейка сломается, передача остановится.

---

3. Топология «Кольцо»

Как устроена:
Друзья садятся за стол по кругу, и сообщение передаётся от одного к другому по кругу, пока не достигнет адресата.

Преимущества:

- Сообщения передаются последовательно, что может упорядочить процесс.
- Каждое соединение имеет своё направление, что упрощает организацию обмена информацией.

Недостатки:

- Если один участник или его связь сломается, кольцо нарушается, и сообщение не дойдет до конца.
- Для восстановления связи часто требуется специальное оборудование или процедуры.

Жизненный пример:
Представьте круглый стол, где каждый передаёт слово следующему. Если один человек не передаёт дальше слово, весь круг может замолчать.

---

4. Топология «Меш»

Как устроена:
Каждый друг напрямую связан со всеми остальными – у каждого есть своя прямая линия связи с каждым участником.

Преимущества:

- Высокая надёжность: если связь с одним участником временно пропадает, можно связаться по другой линии.
- Нет единой точки отказа: сбой в одном соединении не влияет на другие.

Недостатки:

- Очень много связей, что делает такую систему сложной и дорогой в реализации, особенно при большом количестве участников.

Жизненный пример:
Это как если бы каждый друг имел свой телефон и мог позвонить любому другому напрямую. Если кто-то не отвечает, всегда можно позвонить с другого телефона, но поддерживать такую сеть очень затратно и сложно, если друзей много.

Таким образом, выбор топологии зависит от задачи:

- «Звезда» удобна и проста, но центральный узел должен быть надёжным.
- «Шина» экономична, но чувствительна к поломке основной линии.
- «Кольцо» обеспечивает упорядоченную передачу, но уязвима к сбоям в любом звене.
- «Меш» обеспечивает максимальную надёжность за счёт дублирования связей, но требует много ресурсов.

Эти жизненные примеры помогают понять, как разные способы организации общения (топологии) влияют на надёжность и эффективность передачи информации в компьютерных сетях.

- **Модели взаимодействия:**
  Введение в модель OSI и модель TCP/IP – зачем нужны уровни и как они работают.

Модели взаимодействия в компьютерных сетях помогают разделить процесс передачи данных на отдельные этапы или уровни, каждый из которых выполняет свою задачу. Это можно сравнить с процессом отправки письма по почте.

Почему нужны уровни?

Представьте, что вы хотите отправить письмо другу. Чтобы оно дошло до адресата, необходимо выполнить несколько шагов:

1. **Подготовка письма:** Вы пишете текст письма.
2. **Упаковка:** Письмо складывают и помещают в конверт.
3. **Маркировка:** На конверт пишут адрес получателя и отправителя.
4. **Доставка:** Почтовая служба передаёт письмо по разным этапам (сортировка, транспортировка) до пункта назначения.
5. **Получение:** Друг получает письмо и читает его.

Каждый из этих этапов выполняет свою роль, и если один из них не сработает, письмо может не дойти до адресата. Точно так же в компьютерной сети данные проходят через разные уровни, чтобы гарантировать, что информация передается правильно и надёжно.

Модель OSI

Модель OSI (Open Systems Interconnection - Взаимосвязь открытых систем) делит процесс передачи данных на **7 уровней**:

1. **Физический уровень:**
   Отвечает за передачу «сырых» битов (нулей и единиц) по физическим каналам, таким как кабели или радиоволны. Его можно сравнить с транспортной системой: представьте, что конверты с письмами перевозятся по дорогам. Физический уровень обеспечивает «дороги» для передачи информации между устройствами, позволяя данным «двигаться» от отправителя к получателю.
2. **Канальный уровень:**
   Обеспечивает установление связи между устройствами, проверяет и исправляет ошибки передачи. Это как качественная упаковка и защита конверта, чтобы письмо не повредилось.
3. **Сетевой уровень:**
   Определяет маршрут, по которому данные будут двигаться по сети. Сравните это с выбором почтового маршрута, чтобы письмо дошло до нужного города.
4. **Транспортный уровень:**
   Гарантирует, что все части письма (если оно разделено) дойдут до адресата и будут правильно собраны. Это как служба доставки, следящая, чтобы письмо пришло целым и невредимым.
5. **Сеансовый уровень:**
   Устанавливает, управляет и завершает «разговор» между устройствами. Сравните это с координацией встречи между отправителем и получателем для обмена письмами.
6. **Уровень представления:**
   Отвечает за преобразование данных в формат, понятный обоим сторонам. Это как переводчик, который помогает понять, что написано в письме, если язык отличается.

7. **Прикладной уровень:**
   Это уровень, на котором работают приложения (например, почтовые клиенты, веб-браузеры). Он отвечает за доступ к данным и взаимодействие с пользователем, как конечное чтение и ответ на письмо.

Модель TCP/IP

Модель TCP/IP, которая лежит в основе Интернета, несколько проще и состоит из **4 уровней**:

1. **Сетевой интерфейс (Link):**
   Объединяет функции физического и канального уровней. Это подготовка и упаковка письма.
2. **Интернет-уровень:**
   Отвечает за маршрутизацию данных по сети, как выбор оптимального почтового маршрута.
3. **Транспортный уровень:**
   Гарантирует надёжную передачу данных, следит за доставкой каждого «конверта». Например, протокол TCP обеспечивает, чтобы письмо не потерялось.
4. **Прикладной уровень:**
   Работает с конечными приложениями, позволяя пользователю отправлять и получать данные – это как сам процесс чтения и отправки писем.

Жизненный пример – отправка письма

1. **Подготовка письма:**
   Вы пишете письмо (данные, которые нужно передать).
2. **Упаковка и маркировка:**
   Письмо складывается в конверт с адресом (на физическом и канальном уровнях, подготовка данных).
3. **Маршрут доставки:**
   Почтовая служба выбирает маршрут, по которому письмо отправится (сетевой уровень — маршрутизация).
4. **Надёжная доставка:**
   Служба доставки следит, чтобы письмо не потерялось и пришло в целости (транспортный уровень — надежность передачи).
5. **Получение и понимание:**
   Друг получает письмо, открывает его и читает (сеансовый, представления и прикладной уровни — установка соединения, преобразование и взаимодействие с приложением).

Итог

Модели OSI и TCP/IP помогают разделить сложный процесс передачи данных на понятные и независимые этапы (уровни). Это как пошаговый процесс отправки письма: от написания и упаковки до доставки и прочтения. Каждый уровень выполняет свою задачу, обеспечивая,

что данные передаются корректно, надёжно и в нужном формате, даже если устройства, участвующие в передаче, различны по своему устройству или расположению.

4. Физический уровень и оборудование

- **Средства передачи данных:**
  Обзор кабельных технологий (витая пара, коаксиальный кабель, оптоволокно) и беспроводных технологий (Wi-Fi, Bluetooth, мобильные сети).

Средства передачи данных — это разные способы, которыми устройства обмениваются информацией. Давайте рассмотрим их на простых примерах, как если бы мы выбирали, каким способом передать сообщение или груз.

Кабельные технологии

1. **Витая пара:**
   Это кабель, состоящий из двух медных проводников, скрученных вместе. Скручивание помогает уменьшить помехи, подобно тому, как если бы вы обматывали два шнура вместе, чтобы они не путались и не мешали друг другу.
   Жизненный пример:
   Представьте, что вы отправляете маленькие послания по внутренней почте в офисе, используя две тонкие, но аккуратно скрученные веревки. Это помогает посланиям идти правильно, без «пересечений» и потерь.
2. **Коаксиальный кабель:**
   Коаксиальный кабель имеет центральный проводник, окружённый слоем изоляции, оплеткой и внешней оболочкой. Он хорошо защищает сигнал от внешних помех.
   Жизненный пример:
   Это как толстая, защищённая труба для воды, по которой вода (или сигнал) проходит, не теряя напора и не загрязняясь от внешних факторов. Такой кабель часто используется для передачи телевизионного сигнала или подключения к интернету.
3. **Оптоволокно:**
   В оптоволоконном кабеле данные передаются с помощью света, проходящего по тонким стеклянным или пластиковым нитям. Этот способ обеспечивает очень высокую скорость передачи и почти не подвержен помехам.
   Жизненный пример:
   Представьте, что вы отправляете сообщение с помощью лазерного луча по прозрачному стеклянному каналу — это очень быстро и точно, как если бы луч света передавал информацию с одного конца комнаты на другой.

Беспроводные технологии

1. **Wi-Fi(Wireless Fidelity - беспроводная точность):**
   Позволяет устройствам подключаться к сети без проводов, используя радиоволны.
   Жизненный пример:
   Это как если бы в вашем доме была невидимая почтовая система: вы можете сидеть в любой комнате, а сообщения (данные) все равно доставляются к вам через «воздушные каналы».
2. **Bluetooth:**
   Предназначен для передачи данных на коротких расстояниях между устройствами, например, для подключения наушников к смартфону.
   Жизненный пример:
   Это как разговор между соседями через приглушённый голос: они общаются, находясь

рядом, и передают друг другу информацию без проводов.

3. **Мобильные сети:**
   Сети типа 3G, 4G, 5G обеспечивают связь через сотовые вышки, позволяя получать интернет и совершать звонки практически в любом месте.
   Жизненный пример:
   Подумайте о мобильной сети как о системе почтовых станций, которые находятся по всему городу: они ловят ваш сигнал и передают его, даже когда вы находитесь вне дома.

Итог:

- **Кабельные технологии** (витая пара, коаксиальный кабель, оптоволокно) похожи на различные виды труб или проводов, которые физически соединяют устройства и надежно передают данные.
- **Беспроводные технологии** (Wi-Fi, Bluetooth, мобильные сети) позволяют обмениваться данными без проводов, как невидимые мосты или каналы, обеспечивая мобильность и удобство.

Каждый способ передачи данных выбирается в зависимости от нужд: для стабильной и быстрой передачи часто используют кабели, а для мобильности и свободы передвижения — беспроводные технологии.

- **Сетевое оборудование:**
  Функции модемов, маршрутизаторов, коммутаторов, концентраторов и точек доступа.

Сетевое оборудование помогает устройствам обмениваться данными и организует работу сети. Представьте, что сеть — это большой город, где каждое устройство — это житель, а оборудование помогает им общаться и передвигаться. Рассмотрим основные виды оборудования и их функции на простых примерах:

Модем

Функция:
Модем преобразует сигналы, чтобы можно было передавать данные через телефонные линии, кабели или другие носители. Он переводит цифровую информацию из вашего компьютера в формат, подходящий для передачи по линии, и обратно.

Жизненный пример:
Представьте, что вы пишете письмо на родном языке, а ваш друг говорит на другом. Модем — это как переводчик, который переводит ваше письмо так, чтобы почтовая служба могла его доставить, а затем переводит ответ обратно на ваш язык.

Маршрутизатор

Функция:
Маршрутизатор выбирает оптимальный путь для передачи данных между различными сетями, направляя пакеты информации туда, где они нужны. Он работает как «трафик-менеджер» для данных.

Жизненный пример:
Представьте, что вы отправляете посылку. Маршрутизатор — это как сотрудник сортировочного центра, который определяет, какой путь выбрать, чтобы посылка добралась до адресата как можно быстрее и без потерь.

Коммутатор

Функция:
Коммутатор работает внутри локальной сети (LAN). Он принимает данные и отправляет их только тому устройству, для которого они предназначены, избегая лишнего «шумного» вещания.

Жизненный пример:
Это как офисный менеджер, который, получив звонок или сообщение, передаёт его непосредственно тому сотруднику, которому оно адресовано, вместо того чтобы сообщать всем подряд.

Концентратор

Функция:
Концентратор (хаб) принимает входящие данные и передаёт их всем устройствам в сети. Он

не умеет выбирать, кому адресовать информацию, поэтому посылает одно и то же сообщение всем.

Жизненный пример:
Представьте, что вы объявляете что-то через мегафон на школьном собрании. Все слышат одно и то же объявление, даже если оно предназначено только для определённой группы. Такой способ менее эффективен, потому что лишняя информация рассеивается по всему залу.

Точка доступа

Функция:
Точка доступа обеспечивает беспроводное подключение к сети. Она позволяет устройствам, таким как смартфоны, планшеты и ноутбуки, выходить в сеть без проводов.

Жизненный пример:
Это как дверь в кафе с бесплатным Wi-Fi, через которую клиенты заходят и получают доступ к интернету. Точка доступа создаёт «радиомост», соединяя беспроводные устройства с проводной сетью.

Ниже представлена краткая схема типичной домашней сети с описанием того, как соединяются устройства и какая у них роль:

Схема подключения:

1. **Интернет-провайдер**
   Провайдер предоставляет физическую инфраструктуру и соединение с глобальной сетью (интернетом). Он обеспечивает «дороги» для передачи данных к вашему дому.
2. **Модем**
   Сигнал от провайдера поступает в модем. Модем преобразует полученный сигнал в цифровой формат, понятный вашим устройствам.
   Жизненный пример: Это как почтовая станция, где поступают письма, но их нужно перевести с иностранного языка на понятный для вас.
3. **Маршрутизатор (Wi-Fi роутер)**
   Модем подключается к маршрутизатору. Маршрутизатор (часто в виде Wi-Fi роутера) распределяет интернет-сигнал по домашней сети: он обеспечивает проводное и беспроводное (Wi-Fi) подключение устройств.
   Жизненный пример: Представьте диспетчера, который направляет посылки и письма в нужное отделение внутри вашего дома.
4. **Коммутатор (при необходимости)**
   Если вам нужно больше проводных подключений, к маршрутизатору можно подключить коммутатор, который расширяет количество портов.
   Жизненный пример: Это как дополнительный распределительный центр, который помогает доставить письма в каждый уголок большого офиса или дома.
5. **Конечные устройства**
   К маршрутизатору (или к коммутатору, если он используется) подключаются компьютеры, ноутбуки, смартфоны, умные телевизоры, принтеры и другие устройства через проводное или беспроводное соединение.
   Жизненный пример: Это конечные получатели почты – жители дома или офиса,

которые получают и используют информацию.

Итоговая схема подключения(данная схема работает и в обратном направлении, например когда вы посылаете данные(письма, чат и т.д)**):**

Интернет-провайдер → Модем → Маршрутизатор (Wi-Fi роутер) → *(при необходимости Коммутатор)* → **Конечные устройства**

Таким образом, роль каждого элемента выглядит так:

- **Провайдер** обеспечивает внешнее соединение с интернетом.
- **Модем** переводит сигнал провайдера в цифровой формат.
- **Маршрутизатор (Wi-Fi роутер)** распределяет интернет-сигнал внутри вашего дома.
- **Коммутатор** (если нужен) расширяет число проводных подключений.
- **Конечные устройства** – это компьютеры, смартфоны и другие гаджеты, которые пользуются интернетом.

Эта схема позволяет обеспечить стабильное и удобное подключение к интернету в домашней сети.

5. Канальный уровень: основы передачи данных

- **Ethernet:**
  Принципы работы, понятие MAC-адресов, методы доступа к среде передачи.

Ethernet – это технология, позволяющая устройствам в локальной сети обмениваться данными через кабели, чаще всего используя витую пару. Давайте разберём, как она работает, что такое MAC-адреса и как осуществляется доступ к линии передачи, на простом примере.

Принципы работы Ethernet

Представьте, что Ethernet – это система почтовой службы в небольшом районе, где все дома (устройства) соединены дорогами (кабелями). Когда вы отправляете посылку (данные), она упаковывается в конверт (кадр). Этот конверт содержит всю информацию: откуда и куда отправляется посылка, а также саму полезную нагрузку.

Понятие MAC-адресов

Каждый дом в нашем районе имеет свой уникальный почтовый адрес. В мире Ethernet у каждого устройства есть **MAC-адрес** (Media Access Control address - Адрес управления доступом к среде) – уникальный идентификатор, записанный в сетевой карте.

- **Жизненный пример:** Если вы хотите отправить посылку своему соседу, вы обязательно укажете его уникальный адрес, чтобы почтальон знал, куда доставить письмо. Точно так же, когда устройство посылает данные, в кадре указывается MAC-адрес получателя, и только это устройство «забирает» свою посылку.

Методы доступа к среде передачи

Чтобы все посылки (данные) доставлялись правильно, нужно, чтобы дома не пытались отправлять посылки одновременно по одной и той же узкой дороге. В Ethernet для этого используется метод, называемый **CSMA/CD** (Carrier Sense Multiple Access with Collision Detection - Контроль носителя множественного доступа с обнаружением столкновений):

1. **Слушаем дорогу:** Прежде чем отправить посылку, устройство «слушает» кабель (среду передачи), чтобы убедиться, что по нему сейчас никто не отправляет данные.
2. **Передача:** Если кабель свободен, устройство отправляет свои данные.
3. **Обнаружение столкновений:** Если вдруг два устройства одновременно решат отправить данные, их посылки могут столкнуться. В этом случае каждое устройство замечает столкновение и останавливает передачу.
4. **Повторная отправка:** После короткой случайной паузы устройства повторяют попытку отправки, чтобы избежать повторного столкновения.

- **Жизненный пример:**
  Представьте, что два соседа одновременно выходят на узкую улочку, чтобы отправить посылки. Если они столкнутся, оба временно останавливаются, обсуждают ситуацию и спустя несколько минут снова пытаются отправить посылки, но уже по очереди.

Замечание:

В современных сетях часто используются коммутаторы (switches), которые создают для каждого устройства отдельный «проход», что устраняет проблему столкновений. Это как если бы у каждого дома была своя отдельная дорога к почтовому центру, и посылки доставлялись без риска столкновения.

Итог

- **Ethernet** делит данные на кадры, которые отправляются по кабелям.
- Каждый кадр содержит MAC-адреса отправителя и получателя, как уникальные почтовые адреса домов.
- Метод **CSMA/CD** помогает устройствам «слушать» кабель и избегать одновременной передачи данных, предотвращая столкновения.
- В современных сетях благодаря коммутаторам каждый канал становится индивидуальным, что повышает скорость и надежность передачи.

Таким образом, Ethernet обеспечивает эффективную и организованную передачу данных в локальной сети, как хорошо налаженная почтовая служба в небольшом районе.

- **Коммутация:**
  Как данные передаются по сегментам сети, роль коммутаторов в разделении трафика.

Коммутация — это процесс, при котором данные перемещаются по сегментам сети с помощью специальных устройств, называемых коммутаторами. Давайте разберем, как это работает и зачем нужны коммутаторы, на простом жизненном примере.

Как данные передаются по сегментам сети

В локальной сети устройства обмениваются информацией, передавая данные в виде кадров. Когда устройство отправляет данные, эти кадры направляются в коммутатор, который анализирует их, читая адрес получателя (MAC-адрес). Коммутатор затем отправляет эти данные только в тот сегмент сети, где находится устройство-получатель. Таким образом, трафик разделяется по сегментам, и ненужная рассылка данных по всем портам не происходит.

Роль коммутаторов в разделении трафика

Коммутатор выполняет функцию «умного распределителя»:

- **Уменьшение перегрузки:** Вместо того чтобы рассылать данные всем устройствам, коммутатор направляет их напрямую к нужному получателю.
- **Повышение эффективности:** Это снижает вероятность столкновений данных и уменьшает нагрузку на сеть, что делает передачу информации быстрее и надежнее.

Жизненный пример

Представьте большое офисное здание с множеством квартир. Если почтальон доставляет посылку, он не разносит ее во все квартиры, а точно знает, в какую квартиру доставить посылку, используя адрес.

- **Коммутатор** — это как почтальон, который получает письмо (данные) и направляет его только в ту квартиру (сегмент сети), для которой оно предназначено.
- **Сегменты сети** — это отдельные квартиры или офисы в здании. Каждый сегмент имеет свою «дверь», через которую коммутатор может доставить информацию напрямую.

Благодаря такой системе, сообщения не теряются и не создается ненужный «шум», когда данные отправляются только там, где они действительно нужны.

Таким образом, **коммутация** с использованием коммутаторов позволяет эффективно распределять трафик по сегментам сети, как почтальон, который точно доставляет письма по адресам в большом здании, обеспечивая быструю и надежную передачу информации.

*****Что бы не путать понятия и не путаться, дополнительные пояснение*****
Интернет-провайдер:
Провайдер обеспечивает физическое подключение к Интернету – это как городские дороги, по которым движется почта.

1. **Модем:**
   Модем получает сигнал от провайдера и преобразует его в цифровой формат. Это как почтовая станция, где письмо готовится к дальнейшей доставке.
2. **Маршрутизатор (Wi-Fi роутер):**
   Маршрутизатор подключается к модему и распределяет интернет-сигнал по вашей домашней сети (LAN). Он работает на уровне IP-адресов и обеспечивает связь между вашим домом и внешним интернетом – как диспетчер, который направляет посылки по нужным маршрутам. Благодаря встроенной точке доступа, он часто называется Wi-Fi роутером, позволяющим подключаться к сети беспроводным способом.
3. **Локальная сеть (LAN) и сегменты:**
   Ваша LAN – это область, где все устройства (компьютеры, смартфоны, принтеры и т.д.) связаны между собой и могут обмениваться данными. Каждое устройство – это узел, а сама LAN считается сегментом сети, поскольку объединяет все узлы в одном пространстве.
4. **Коммутатор (Switch):**
   Если вам нужно больше проводных подключений или вы хотите оптимизировать передачу данных внутри LAN, можно подключить коммутатор. Он работает на уровне MAC-адресов, получая данные и направляя их только к тому устройству, для которого они предназначены – как умный почтальон, который доставляет письмо только в нужную квартиру. Коммутатор действует внутри LAN, а не между глобальными сетями.

Общая схема подключения:

Интернет-провайдер → Модем → Маршрутизатор (Wi-Fi роутер) → *(при необходимости Коммутатор)* **→ Конечные устройства (узлы в LAN)**

Краткое различие:

- **Маршрутизатор (Wi-Fi роутер):** Соединяет вашу домашнюю сеть с интернетом, работает с IP-адресами, направляя данные между глобальными сетями и вашим домом.
- **Коммутатор:** Расширяет и оптимизирует проводное подключение внутри LAN, распределяя данные между устройствами по MAC-адресам.

Таким образом, в вашей домашней сети провайдер обеспечивает внешнее соединение, модем преобразует сигнал, маршрутизатор (Wi-Fi роутер) распределяет его по LAN, а коммутатор помогает эффективно управлять проводными соединениями внутри этой сети.

Распространённое заблуждение, что устройства в сети должны подключаться строго в «порядке уровней» OSI, то есть сначала устройство физического уровня, потом канального, затем сетевого. На самом деле, OSI-модель — это абстрактная, концептуальная модель, которая описывает, какие функции выполняются на каждом уровне, а не указывает физическую последовательность подключения.

Вот как это работает на практике:

- **Модем** выполняет функции физического уровня, преобразуя сигналы от провайдера в цифровой формат.

- **Маршрутизатор (Wi-Fi роутер)** работает на сетевом уровне, принимая цифровой сигнал от модема и распределяя его между сетями (локальной и внешней) на основе IP-адресов.
- **Коммутатор** действует в пределах локальной сети на канальном уровне, направляя данные между устройствами по их MAC-адресам.

При этом устройства подключаются так, чтобы обеспечить корректную работу всей сети, независимо от того, на каком уровне OSI они работают. Например, в домашней сети модем подключается к маршрутизатору, а маршрутизатор может быть непосредственно связан с конечными устройствами или с коммутатором для расширения числа проводных подключений.

Таким образом, последовательность подключения определяется их функциональной ролью и практическими потребностями сети, а не порядком уровней OSI. Это как если бы вы строили кухню: плита, холодильник и стол могут находиться в разном порядке, но каждый из них выполняет свою задачу, и вместе они создают рабочее пространство.

- **Ошибки и контроль целостности:**
  Методы обнаружения и коррекции ошибок на канальном уровне.

При передаче данных по сети могут возникать ошибки – отдельные биты могут измениться или потеряться. Для их обнаружения и коррекции на канальном уровне применяются специальные методы, позволяющие проверить, что данные не были повреждены в процессе передачи, и, если были, попытаться их исправить.

Как это работает:

1. **Обнаружение ошибок:**
   Перед отправкой данных к ним добавляется специальная контрольная информация (например, контрольная сумма или CRC(Cyclic Redundancy Check - Циклическая проверка избыточности) – циклический избыточный код).
   Пример из жизни:
   Представьте, что вы отправляете рецепт по почте и прикрепляете к нему список ингредиентов, который подсчитывает общее количество букв. Получатель, получив рецепт, снова подсчитывает буквы. Если число не совпадает с указанным, значит, где-то произошла ошибка.
2. **Коррекция ошибок:**
   Если ошибка обнаружена, устройство может запросить повторную передачу данных (метод ARQ(Automatic Repeat reQuest) – Автоматический повторный запрос) или, в некоторых случаях, исправить ошибку самостоятельно, если применяются специальные коды коррекции ошибок.
   Пример из жизни:
   Если рецепт пришёл с опечатками, вы можете позвонить отправителю и попросить прислать новый, исправленный вариант. Либо, если опечатка всего в одной букве и о ней можно догадаться, вы можете сами догадаться, что имелось в виду.

Основные методы:

- **Проверка четности:**
  Добавляется дополнительный бит (бит четности), который помогает обнаружить, изменилось ли нечётное или чётное число бит в данных.
- **Контрольная сумма и CRC:**
  Более сложные алгоритмы, которые вычисляют определённое значение по всему блоку данных. Получатель пересчитывает это значение и сравнивает с полученным – если они не совпадают, данные повреждены.
- **ARQ (автоматический повторный запрос):**
  Если обнаружена ошибка, отправитель автоматически передаёт данные повторно, чтобы получатель получил корректную информацию.

Итог:

Контроль целостности и методы обнаружения/коррекции ошибок на канальном уровне – это как система проверки и перепроверки отправляемых писем. Перед отправкой вы добавляете контрольные отметки, а получатель сверяет их, чтобы убедиться, что письмо дошло без

повреждений. Если что-то пошло не так, письмо отправляется повторно или исправляется, чтобы информация осталась точной и полной.

6. Сетевой уровень: маршрутизация и IP-адресация

- **IP-адресация:**
  Структура IPv4 и введение в IPv6, понятия подсетей и масок.

IP-адресация(Internet Protocol address – Интернет протокол адресации) — это способ присвоения уникальных адресов каждому устройству в сети, чтобы они могли находить друг друга и обмениваться данными. Представьте, что каждый компьютер или смартфон — это дом в большом городе, и для доставки почты (данных) каждому нужен свой адрес.

IPv4

- **Структура IPv4(**Internet Protocol version 4 – Интернет Протокол версии 4**)**:
  IPv4-адрес состоит из 32 бит, которые обычно записываются в виде четырех чисел от 0 до 255, разделенных точками, например: **192.168.1.1**.
  Жизненный пример:
  Это как если бы каждый дом имел адрес в виде четырех чисел: улица, номер дома, квартира и дополнительный код.
- **Ограничения IPv4:**
  Всего существует около 4 миллиардов возможных адресов, что сегодня становится недостаточным для всех подключенных устройств.

IPv6

- **Структура Ipv6(**Internet Protocol version 6 – Интернет Протокол версии 6**):**
  IPv6 — новая версия IP-адресации, которая использует 128 бит, что позволяет иметь гораздо больше адресов. Адреса записываются в виде восьми групп четырех шестнадцатеричных цифр, разделенных двоеточиями, например:
  2001:0db8:85a3:0000:0000:8a2e:0370:7334.
  Жизненный пример:
  Если IPv4 — это адрес, состоящий из четырех чисел, то IPv6 — это как длинный адрес, включающий много деталей, что позволяет точно определить каждый дом даже в огромном городе с миллиардами жителей.

Подсети и маски

- **Подсети:**
  Подсеть — это часть большой сети, разделенная на более мелкие группы, чтобы упростить управление и повысить безопасность.
  Жизненный пример:
  Представьте, что весь город разделен на районы. Каждый район — это подсеть, где дома (устройства) имеют общую часть адреса, указывающую на этот район.
- **Маски:**
  Маска подсети определяет, какая часть IP-адреса относится к сети (например, к району), а какая — к конкретному устройству (дому). Например, маска **255.255.255.0** означает, что первые три числа адреса (например, 192.168.1) определяют сеть, а последнее число — конкретное устройство в этой сети.

Жизненный пример:
Если представлять адрес как "Город-Район-Дом", то маска помогает понять, какая часть адреса говорит о городе и районе, а какая — о номере дома. Это как если бы все дома на одной улице имели одинаковый префикс, а индивидуальные номера определяли конкретный дом.

Итог

- **IPv4** — это система с 32-битными адресами, записываемыми как четыре числа (например, 192.168.1.1).
- **IPv6** — современная система с 128-битными адресами, дающая практически неограниченное количество уникальных адресов, записываемых в шестнадцатеричном формате с двоеточиями.
- **Подсети и маски** помогают разделить большую сеть на более мелкие управляемые группы, где маска определяет, какая часть адреса относится к сети, а какая — к конкретному устройству.

Таким образом, IP-адресация позволяет устройствам в сети иметь уникальные «домашние адреса», а подсети и маски помогают организовать эти адреса так, чтобы управление сетью было удобным и эффективным.

Давайте разберёмся в различиях между IP-адресом, MAC-адресом и маской подсети.

IP-адрес
IP-адрес (Internet Protocol address) — это логический адрес, который назначается устройству для идентификации и маршрутизации в сети. Он может быть представлен в виде IPv4 (например, 192.168.1.10) или IPv6 (например, 2001:0db8:85a3:0000:0000:8a2e:0370:7334). Этот адрес позволяет передавать данные между устройствами как в локальной сети, так и в интернете.

MAC-адрес
MAC-адрес (Media Access Control address) — это уникальный физический адрес, присваиваемый сетевому интерфейсу устройства на этапе производства. Он записывается в виде шести групп шестнадцатеричных чисел (например, 00:1A:2B:3C:4D:5E). MAC-адрес используется на уровне канального уровня (Layer 2) для идентификации устройств внутри одной локальной сети и управления передачей данных между ними.

Маска подсети
Маска подсети — это инструмент, позволяющий разделить IP-адрес на две части: часть, идентифицирующую сеть, и часть, идентифицирующую конкретное устройство (хост) внутри этой сети. Например, маска 255.255.255.0 указывает, что первые три октета IP-адреса обозначают сеть, а последний октет — конкретное устройство. Это помогает определить, находится ли устройство в той же сети, что и отправитель, или данные нужно направлять через маршрутизатор для дальнейшей передачи.

Кратко:

- **IP-адрес** — логический идентификатор для связи и маршрутизации в сети.
- **MAC-адрес** — физический, уникальный адрес сетевой карты для взаимодействия внутри локальной сети.
- **Маска подсети** — инструмент для разделения IP-адреса на сетевую и хостовую части, определяющий границы сети.

- **Маршрутизация:**
  Основные принципы выбора маршрута, понятие маршрутизаторов и алгоритмов маршрутизации.

Маршрутизация — это процесс выбора оптимального пути для передачи данных (пакетов) от отправителя к получателю через сеть. Представьте себе ситуацию с отправкой посылки по почте в большом городе: чтобы посылка добралась до адресата, её нужно правильно направить через несколько почтовых отделений, которые подбирают наилучший маршрут с учётом дорожной ситуации.

Основные понятия:

1. **Маршрутизатор (роутер):**
   Это устройство, которое работает на сетевом уровне (Layer 3 — Третий слой(уровень)) модели OSI. Оно принимает пакеты данных, анализирует их IP-адреса и определяет, по какому маршруту передать их дальше, чтобы они достигли конечного пункта назначения.
   Жизненный пример:
   Представьте диспетчера в почтовом центре, который, получив посылку, смотрит на адрес и решает, по каким дорогам и через какие офисы её отправить, чтобы доставка прошла быстро и без проблем.
2. **Алгоритмы маршрутизации:**
   Это набор правил и методов, которые маршрутизаторы используют для выбора наилучшего маршрута. Например, алгоритмы могут учитывать количество узлов (хопов), задержки, пропускную способность и даже текущую загруженность сетевых путей.
   Примеры алгоритмов:
   - **RIP (Routing Information Protocol - Протокол маршрутной информации):** выбирает маршрут с наименьшим количеством переходов.
   - **OSPF (Open Shortest Path First - Сначала откройте кратчайший путь):** ищет кратчайший путь по различным параметрам.
   - **BGP (Border Gateway Protocol - Протокол пограничного шлюза):** используется для маршрутизации между автономными системами в глобальном интернете.

Жизненный пример:

Представьте, что вы планируете автомобильное путешествие из одного города в другой.

- **Навигатор** (как маршрутизатор) анализирует разные маршруты: один может быть короче, но с пробками, а другой — длиннее, но с гладким движением.
- **Алгоритм маршрутизации** (например, используемый навигатором) сравнивает варианты по времени в пути, состоянию дорог и загруженности, чтобы выбрать наилучший маршрут.
- Вы отправляете свой автомобиль по выбранному маршруту, и он проходит через несколько перекрёстков и развязок, пока не достигнет пункта назначения.

Точно так же, когда вы отправляете данные через интернет, ваш запрос проходит через множество маршрутизаторов, которые по очереди выбирают оптимальные пути, чтобы данные доставились быстро и без потерь.

Итог:

- **Маршрутизация** — это как система «умного» навигатора для данных, которая определяет лучший путь от отправителя к получателю.
- **Маршрутизатор** (роутер) — это устройство, которое выполняет эту задачу, анализируя IP-адреса и используя таблицы маршрутизации.
- **Алгоритмы маршрутизации** помогают выбирать оптимальный путь, учитывая разные параметры сети, подобно тому как навигатор выбирает лучший маршрут для автомобиля.

Таким образом, маршрутизация обеспечивает эффективное и надежное перемещение данных через сложную сеть, позволяя каждому пакету найти свой путь, как посылка, которая проходит через несколько почтовых отделений до своего адресата.

-

*****Что бы не путать понятия и не путаться, дополнительные пояснение*****

В передаче данных участвуют и оборудование провайдера, и устройства пользователей, и оба типа оборудования играют важную роль в маршрутизации информации от отправителя к получателю. Т.е. при посылке данных от одного пользователя другому, маршрут состоит из устройств провайдеров и двух пользователей между которыми должна быть связь.

Как это работает:

1. **Ваше оборудование:**
   Ваш компьютер или смартфон сначала отправляет данные через ваш домашний маршрутизатор (Wi-Fi роутер). Этот маршрутизатор обеспечивает связь внутри вашей локальной сети (LAN) и передаёт данные дальше.
2. **Оборудование провайдера:**
   Далее данные попадают в инфраструктуру вашего интернет-провайдера. Здесь используются различные устройства, такие как:
   - **Модем:** преобразует сигнал от провайдера в цифровой формат.
   - **Маршрутизаторы и коммутаторы провайдера:** направляют данные по своим сетям, обеспечивая передачу информации между различными сетями и регионами.
   - **Магистральные линии и сервера:** передают данные на большие расстояния между разными провайдерами и регионами.
3. **Промежуточное оборудование:**
   Если данные направляются, например, в соседний город или даже в другую страну, они проходят через несколько маршрутизаторов и коммутаторов, принадлежащих различным провайдерам, прежде чем достичь конечного пункта назначения.

Жизненный пример:

Представьте, что вы отправляете письмо:

- **Ваш почтовый ящик (домашнее устройство)** – вы кладёте письмо.
- **Ваш местный почтовный отдел (ваш домашний маршрутизатор)** – забирает письмо из вашего ящика и отправляет его дальше.

- **Почтовые центры провайдера** – письмо проходит через местный, затем региональный, а потом национальный почтовый центр, где оно сортируется и направляется дальше, возможно, через несколько почтовых отделений.
- **Почтовый отдел получателя** – после прохождения через всю систему, письмо попадает в местное отделение, откуда уже доставляют его по конкретному адресу.

Таким образом, как и в почтовой системе, в интернете данные проходят через множество звеньев. Ваше личное оборудование отправляет данные, а затем мощное оборудование провайдера (маршрутизаторы, коммутаторы, магистральные сети и т.д.) обеспечивает их дальнейшую передачу до конечного адресата.

- **Протоколы маршрутизации:**
  Обзор таких протоколов, как RIP, OSPF, BGP – их задачи и особенности.

Протоколы маршрутизации – это набор правил, по которым маршрутизаторы выбирают лучший путь для передачи данных от отправителя к получателю. Проще говоря, они помогают «навигации» данных по интернету, чтобы они шли по оптимальному маршруту. Давайте рассмотрим несколько популярных протоколов:

---

1. RIP (Routing Information Protocol)

- **Как работает:**
  RIP – это простой протокол, который выбирает маршрут по количеству «прыжков» (хопов) между узлами сети. Чем меньше хопов, тем предпочтительнее маршрут.
- **Особенности:**
  - Легкость настройки и использования.
  - Ограничение – максимальное число хопов (обычно 15), что делает его неэффективным для очень больших сетей.
- **Жизненный пример:**
  Представьте, что вам нужно доставить посылку, и вы выбираете маршрут, руководствуясь количеством пересечений улиц: чем меньше перекрестков, тем лучше. Если маршрут проходит через 5 перекрестков, а другой – через 10, вы выберете первый вариант, не учитывая качество дорог или пробки.

---

2. OSPF (Open Shortest Path First)

- **Как работает:**
  OSPF – более сложный и точный протокол, который собирает подробную информацию о состоянии каждого сегмента сети (так называемый «link-state»). Он использует алгоритм Дейкстры для расчёта кратчайшего пути, учитывая не только количество узлов, но и такие параметры, как скорость каналов, задержки и загруженность.
- **Особенности:**
  - Подходит для крупных корпоративных сетей.
  - Быстро реагирует на изменения в сети (например, при отказе узла).
- **Жизненный пример:**
  Если вернуться к примеру с доставкой посылки, OSPF – это как навигатор, который анализирует не только количество перекрестков, но и состояние дорог: учитывает пробки, ремонтные работы и дорожные условия, чтобы выбрать самый быстрый и надежный маршрут.

---

3. BGP (Border Gateway Protocol)

- **Как работает:**
  BGP используется для маршрутизации между автономными системами – это те части сети, которыми управляют разные организации или интернет-провайдеры. Он не только ищет кратчайший путь, но и учитывает множество других факторов, таких как политика маршрутизации, надежность и коммерческие соглашения между

организациями.

- **Особенности:**
    - Ключевой протокол для глобального интернета.
    - Решает задачи маршрутизации между разными провайдерами и большими сетевыми сегментами.
- **Жизненный пример:**
  Представьте, что вам нужно доставить посылку из одной страны в другую. BGP – это как международная логистическая компания, которая выбирает маршрут с учетом не только расстояния, но и таможенных правил, торговых соглашений и политической обстановки. Здесь выбор маршрута может зависеть от множества факторов, а не только от количества перекрестков.

Итог

- **RIP** выбирает маршрут по числу узлов – как если бы вы выбирали маршрут с минимальным количеством перекрестков.
- **OSPF** анализирует состояние каждой дороги, чтобы найти самый быстрый и надежный путь – как современный навигатор, учитывающий множество параметров.
- **BGP** управляет международными и межпровайдерскими маршрутами, учитывая сложные политические и коммерческие условия – как глобальная логистическая система доставки.

Каждый из этих протоколов решает задачу маршрутизации в зависимости от масштаба и сложности сети, помогая данным находить оптимальный путь от отправителя к получателю.

7. Транспортный уровень: TCP и UDP

- **TCP (Transmission Control Protocol – Протокол контроля передачи):**
  Как устанавливается соединение, принципы контроля ошибок, управление потоком и надежная доставка данных.

TCP (Transmission Control Protocol) — это протокол, который обеспечивает надёжную, последовательную и корректную передачу данных между двумя устройствами в сети. Он работает по принципу установления соединения, контроля ошибок и управления потоком данных. Давайте рассмотрим, как это происходит, и приведём жизненный пример.

---

1. Установление соединения (Three-Way Handshake - Трехстороннее рукопожатие)

Перед тем как начать передачу данных, устройства должны договориться об установлении связи. TCP использует так называемый «трёхсторонний рукопожатие»:

- **Шаг 1:** Отправитель посылает запрос на установление соединения (SYN - synchronize - синхронизировать).
- **Шаг 2:** Получатель отвечает подтверждением запроса (SYN-ACK – SYNchronize-ACKnowledgement – синхронизировать - подтверждение).
- **Шаг 3:** Отправитель отправляет финальное подтверждение (ACK - acknowledgement - подтверждение), после чего соединение считается установленным.

Жизненный пример:
Представьте, что вы хотите позвонить другу. Вы набираете номер (SYN), друг слышит звонок и отвечает: «Привет, я на линии, подтверждаю» (SYN-ACK), а вы говорите: «Отлично, связь установлена» (ACK). Теперь можно начать разговор.

---

2. Контроль ошибок

TCP следит за тем, чтобы данные доходили без ошибок. Для этого используются номера последовательности и подтверждения (ACK):

- **Номера последовательности:** Каждый сегмент данных получает свой номер, чтобы получатель мог определить порядок и целостность данных.
- **Подтверждения (ACK):** Получатель отправляет подтверждения о получении данных. Если подтверждение не получено в течение определённого времени, отправитель повторно передаёт соответствующий сегмент.

Жизненный пример:
Это похоже на отправку пачки писем с номерами на конвертах. Получатель, получив каждое письмо, отмечает, что оно пришло. Если какое-то письмо не пришло, отправитель повторно отправляет его, чтобы всё было получено в нужном порядке.

---

3. Управление потоком

TCP регулирует, сколько данных может быть отправлено, чтобы получатель успевал их обрабатывать, и чтобы не возникало перегрузки сети. Это называется механизмом «скользящего окна» (sliding window):

- **Скользящее окно:** Отправитель может отправлять данные до тех пор, пока не получит подтверждение о получении предыдущих сегментов. Размер окна определяется возможностями получателя и текущей нагрузкой в сети.

Жизненный пример:
Представьте, что вы отправляете серию посылок, но перед тем как отправить следующую, ждёте подтверждения, что предыдущая доставлена. Это помогает избежать ситуации, когда получатель перегружается большим количеством посылок одновременно.

---

4. Надёжная доставка данных

Благодаря установлению соединения, контролю ошибок и управлению потоком TCP обеспечивает надежную доставку данных:

- **Повторная передача:** Если данные теряются или повреждаются, TCP автоматически запрашивает их повторную отправку.
- **Соблюдение порядка:** Получатель собирает данные в правильном порядке, даже если они пришли не последовательно.

Жизненный пример:
Это как если бы вы отправляли важное письмо, и если почтальон не смог доставить его, вы снова отправляете его, чтобы письмо точно дошло. Кроме того, если письмо разбито на несколько частей, получатель собирает их в правильном порядке, как пазл, чтобы восстановить исходное сообщение.

Итог

TCP устанавливает соединение (тройное рукопожатие), следит за целостностью данных (через номера последовательности и подтверждения), управляет потоком (механизм скользящего окна) и обеспечивает надежную доставку, повторно отправляя потерянные данные. Это делает TCP «надежным почтовым сервисом», который гарантирует, что ваше сообщение будет доставлено в полном объеме и в правильном порядке.

- **UDP (User Datagram Protocol - Протокол пользовательских датаграмм):**
  Отличия от TCP, ситуации, в которых применяется UDP (например, трансляция видео, онлайн-игры).

UDP (User Datagram Protocol) — это протокол передачи данных, который работает иначе, чем TCP. Он обеспечивает быструю, но менее надежную передачу информации. Давайте разберем его основные отличия и ситуации применения на простом примере.

Основные отличия UDP от TCP:

1. **Отсутствие установления соединения:**
   - **TCP:** Перед отправкой данных устанавливается соединение (тройное рукопожатие), чтобы гарантировать, что обе стороны готовы к передаче.
   - **UDP:** Данные отправляются сразу, без предварительного установления соединения. Это называется «без установления соединения» (connectionless).
2. **Надежность и контроль ошибок:**
   - **TCP:** Гарантирует, что все данные доставлены корректно, благодаря подтверждениям, повторной передаче потерянных пакетов и проверке последовательности.
   - **UDP:** Не обеспечивает контроля доставки. Если пакет теряется, нет автоматического повторного запроса. Это снижает задержки, но может привести к потере части информации.
3. **Управление потоком:**
   - **TCP:** Регулирует скорость передачи данных, чтобы не перегружать сеть и получателя.
   - **UDP:** Отправляет данные максимально быстро, не заботясь о скорости их обработки на стороне получателя.

Когда применяется UDP?

UDP выбирают в тех случаях, когда важнее скорость передачи, а небольшая потеря данных допустима. Примеры таких ситуаций:

- **Трансляция видео и аудио:**
  При потоковой передаче видео или аудио задержки могут негативно влиять на качество. Если небольшой фрагмент данных теряется, это может вызвать незначительные артефакты(под "артефактами" в контексте потоковой передачи видео или аудио понимаются небольшие визуальные или звуковые искажения, которые появляются, когда часть данных не дошла до получателя), но поток продолжается без задержек.
- **Онлайн-игры:**
  В играх скорость реакции критична. Если некоторые пакеты данных теряются, игра может немного «подергиваться», но задержки в передаче могут привести к плохому игровому опыту.
- **VoIP (Voice over Internet Protocol(Голосовая связь по интернет-протоколу)голосовая связь через интернет):**
  Для телефонных звонков через интернет небольшие потери данных допустимы, чтобы

обеспечить минимальную задержку, что важно для нормального разговора.

Жизненный пример:

Представьте, что вы организуете массовое рассылку коротких сообщений для группы друзей:

- **TCP-подход:**
  Вы сначала звоните каждому другу, убеждаетесь, что он готов принять сообщение, затем отправляете его, и если друг не подтверждает, звоните еще раз. Это надежно, но занимает время.
- **UDP-подход:**
  Вы просто отправляете SMS всем друзьям одновременно, не ожидая подтверждения. Если кто-то не получает одно сообщение, это не критично, потому что основное сообщение все равно дошло и разговор продолжается.

Таким образом, UDP позволяет быстро передавать данные без лишних задержек, даже если не гарантируется, что каждый маленький фрагмент информации будет получен. Это делает его идеальным для приложений, где скорость важнее абсолютной надежности.

- **Сравнение TCP и UDP:**
  Примеры, где выбор того или иного протокола имеет значение для работы приложений.

TCP и UDP – это два основных протокола для передачи данных в сети, и выбор между ними зависит от того, что важнее: надежность или скорость.

TCP (Transmission Control Protocol)

- **Надежность:**
  TCP устанавливает соединение перед передачей данных (трёхстороннее рукопожатие). Он гарантирует, что все данные будут получены в правильном порядке. Если какой-то пакет теряется, TCP отправляет его повторно.
- **Контроль ошибок и управление потоком:**
  Получатель подтверждает получение каждого пакета. Если подтверждение не приходит, отправитель повторно передаёт недостающие данные. Это гарантирует целостность информации, но может приводить к задержкам.
- **Применение:**
  TCP идеально подходит для приложений, где важна полная и правильная передача данных, например:
  - **Передача файлов, загрузка веб-страниц, электронная почта, онлайн-банкинг.**
- **Жизненный пример:**
  Представьте, что вы отправляете важное письмо с документами через курьерскую службу, которая гарантирует доставку с подтверждением получения. Если документ теряется, курьер незамедлительно отправит его снова. Так же работает TCP – каждый пакет данных подтверждается и при необходимости повторно отправляется, чтобы вы точно получили всё, что отправлено.

UDP (User Datagram Protocol)

- **Скорость:**
  UDP отправляет данные без предварительного установления соединения и без подтверждения получения. Это делает передачу максимально быстрой, но без гарантий доставки или порядка.
- **Отсутствие контроля ошибок:**
  Если некоторые данные теряются, протокол не пытается их восстановить. Это может привести к небольшим ошибкам, но уменьшает задержки.
- **Применение:**
  UDP используется там, где важнее скорость, а небольшая потеря данных допустима:
  - **Потоковое видео и аудио (например, онлайн-трансляции, видеоконференции).**
  - **Онлайн-игры, где задержки могут повлиять на игровой процесс.**
  - **Голосовые звонки по интернету (VoIP).**
- **Жизненный пример:**
  Представьте, что вы отправляете массовое SMS-сообщение друзьям. Вы отправляете

сообщение всем сразу, не дожидаясь подтверждения, что каждый его получил. Если кому-то сообщение не дошло, это не критично – большинство получат его мгновенно, и разговор продолжится. Так работает UDP – данные идут быстро, и если потерялась часть информации, это не приводит к значительным проблемам, особенно если важна скорость.

Итоговое сравнение:

- **TCP:**
  Гарантирует, что данные будут доставлены полностью и в правильном порядке. Подходит для критически важных задач, где важна надежность (например, загрузка файлов или работа с банковскими сайтами).
- **UDP:**
  Обеспечивает быструю передачу данных, не заботясь о гарантии доставки каждого пакета. Идеален для приложений, где небольшие потери данных не критичны, а важна минимальная задержка (например, потоковое видео или онлайн-игры).

Выбор между TCP и UDP зависит от конкретной задачи: если вам нужно быть уверенным в том, что все данные дошли правильно, выбирайте TCP; если важна скорость и допустимы незначительные потери данных, используйте UDP.

8. Прикладной уровень: протоколы и сервисы

- **Основные прикладные протоколы:**
  HTTP/HTTPS для веб-трафика, FTP для передачи файлов, SMTP и POP3/IMAP для электронной почты.

Основные прикладные протоколы – это набор правил, с помощью которых приложения обмениваются данными через Интернет. Они отвечают за передачу информации в зависимости от типа задачи. Давайте рассмотрим их на простых примерах:

HTTP/HTTPS(Hypertext transfer protocol (secure) - Протокол передачи гипертекста (безопасный)) для веб-трафика

- **HTTP (HyperText Transfer Protocol):**
  Этот протокол используется для передачи веб-страниц. Когда вы вводите адрес сайта в браузере, ваш компьютер запрашивает страницу через HTTP, и сервер отправляет её вам.
- **HTTPS (HTTP Secure):**
  Это безопасная версия HTTP, которая шифрует данные для защиты информации от перехвата.

Жизненный пример:
Представьте, что вы приходите в библиотеку (сайт). HTTP – это обычный запрос, когда библиотекарь просто отдаёт вам книгу. А HTTPS – это когда библиотекарь передаёт книгу в запечатанном конверте, чтобы никто не смог её подглядеть или украсть, если вы берёте её домой.

FTP для передачи файлов

- **FTP (File Transfer Protocol - Протокол передачи файлов):**
  Этот протокол предназначен для передачи больших файлов между компьютерами. Он позволяет загружать файлы на сервер и скачивать их обратно.

Жизненный пример:
Это как отправка посылки через курьерскую службу. Вы упаковываете файл (письмо или посылку), сдаёте его курьеру (FTP-клиенту), а курьер доставляет его в нужное место (FTP-сервер). Когда получатель получает посылку, он её распаковывает и использует.

Давайте уточним аналогию для FTP и разберём, кто в ней какую роль играет

В FTP есть два основных участника: FTP-клиент и FTP-сервер.

- **FTP-клиент** – это программа на вашем компьютере, которую вы используете для загрузки (upload) или скачивания (download) файлов. Вы, как пользователь, управляете FTP-клиентом.
- **FTP-сервер** – это удалённый компьютер или устройство, на котором хранятся файлы и к которому вы подключаетесь с помощью FTP-клиента.

Чтобы представить это на примере курьерской службы, можно сказать так:

1. **Отправка файла (upload):**
   Представьте, что вы хотите отправить посылку. Вы берёте посылку (файл) и отдаёте её сотруднику курьерской службы. В этой аналогии:
   - **Вы с вашим FTP-клиентом** – это как отправитель, который передаёт посылку.
   - **FTP-сервер** – это как склад или пункт приёма курьерской службы, куда доставляется посылка.
   - **Курьерская служба** – это механизм передачи данных по сети (сам протокол FTP), который перемещает посылку от вас к складу.
2. **Получение файла (download):**
   Теперь представьте, что вы хотите получить посылку, которая уже находится на складе. Вы обращаетесь к курьерской службе, и они доставляют посылку вам.
   - **FTP-сервер** в этом случае хранит посылку (файл).
   - **Вы через FTP-клиент** запрашиваете эту посылку.
   - **Механизм передачи (протокол FTP)** доставляет посылку с сервера к вам.

Таким образом, ни FTP-клиент, ни FTP-сервер не являются «курьером» в буквальном смысле. Вместо этого:

- **FTP-клиент** – это инструмент, с помощью которого вы отправляете или получаете файлы (как вы отдаёте или забираете посылку).
- **FTP-сервер** – это место, куда файл передаётся для хранения или откуда он доставляется (как склад или почтовое отделение).
- **Сам процесс передачи данных по сети** (то, что происходит между клиентом и сервером) можно сравнить с работой курьера, который перемещает посылку от отправителя к получателю.

Так что, правильнее будет сказать, что вы, используя FTP-клиент, передаёте файл на FTP-сервер, а механизм передачи данных по сети выполняет роль курьера, перемещающего посылку. Это помогает понять, что клиент и сервер – это две стороны взаимодействия, а «курьер» – это процесс, обеспечиваемый протоколом для доставки данных.

SMTP(Simple Mail Transfer Protocol - Простой протокол передачи почты) и POP3/IMAP(Post Office Protocol 3- Почтовый протокол 3/Internet Message Access Protocol - Протокол доступа к сообщениям в Интернете) для электронной почты

- **SMTP (Simple Mail Transfer Protocol):**
  Протокол, который используется для отправки электронной почты с вашего почтового клиента на сервер. Он как «отправитель», который берёт ваше письмо и отправляет его дальше.
- **POP3 (Post Office Protocol) и IMAP (Internet Message Access Protocol):**
  Эти протоколы используются для получения и управления электронной почтой.
  - **POP3** скачивает письмо с сервера на ваше устройство и, как правило, удаляет его с сервера.
  - **IMAP** позволяет работать с письмами прямо на сервере, синхронизируя почту на нескольких устройствах.

Жизненный пример:
Представьте, что вы отправляете письмо по почте. SMTP – это способ, которым ваше письмо отправляется почтовой службой. Когда письмо приходит к получателю, он может либо сразу

забрать его (POP3), либо оставлять его в почтовом отделении и получать доступ к нему с разных устройств (IMAP), как если бы вы могли проверять свой почтовый ящик онлайн.

Итог

- **HTTP/HTTPS** обеспечивают передачу веб-страниц: HTTP для обычного доступа, HTTPS – для безопасного.
- **FTP** используется для отправки и получения файлов, как отправка посылок.
- **SMTP** отвечает за отправку электронных писем, а **POP3/IMAP** – за их получение и управление ими.

Эти протоколы помогают разным приложениям правильно и эффективно обмениваться информацией, каждый в своей области, как различные службы в почтовой системе, обеспечивающие доставку и получение писем или посылок.

- **DNS (Domain Name System):**
  Принцип работы системы доменных имен и ее роль в упрощении доступа к ресурсам сети.

DNS (Domain Name System - Система доменных имен) – это система, которая переводит понятные человеку доменные имена (например, *example.com*) в числовые IP-адреса, необходимые для установления связи между устройствами в интернете.

Как это работает:

1. **Ввод доменного имени:**
   Когда вы вводите адрес сайта в браузере (например, *google.com*), вы обращаетесь к DNS-системе, а не напрямую к IP-адресу.
2. **Запрос к DNS-серверу:**
   Ваш компьютер отправляет запрос на DNS-сервер, который выступает в роли «телефонной книги». Он ищет, какой IP-адрес соответствует введенному доменному имени.
3. **Получение IP-адреса:**
   DNS-сервер находит нужную запись и отправляет IP-адрес обратно вашему компьютеру.
4. **Установление соединения:**
   После получения IP-адреса ваш компьютер использует его для подключения к серверу, где размещен сайт.

Жизненный пример:

Представьте, что вы приезжаете в незнакомый город и хотите посетить определённое кафе, но вы не знаете его точный адрес. Вместо того чтобы искать его на карте с долгим набором цифр, вы звоните в справочную службу или консультируетесь у местного жителя, говоря: «Где находится кафе *«Солнечный берег»*?». Вам отвечают: «Оно находится на улице Ленина, дом 15». Теперь вы уже знаете точный адрес и можете дойти до кафе.

- **Доменные имена** – это как название кафе (*«Солнечный берег»*).
- **IP-адрес** – это как точный адрес (улица Ленина, дом 15).
- **DNS-сервер** – это как справочная служба, которая знает все адреса и подсказывает, куда идти.

Роль DNS в упрощении доступа к ресурсам сети:

- **Удобство:**
  Вам не нужно запоминать сложные числовые IP-адреса – достаточно помнить доменное имя.
- **Динамичность:**
  Если сервер перемещается или меняется его IP-адрес, обновление происходит на стороне DNS-сервера, а пользователи продолжают использовать одно и то же доменное имя.

- **Масштабируемость:**
  DNS позволяет эффективно управлять огромным числом записей для всех сайтов в интернете, делая навигацию простой и быстрой.

Таким образом, DNS играет ключевую роль, делая доступ к ресурсам сети понятным и удобным, как если бы у вас всегда была под рукой актуальная «телефонная книга» или справочник адресов.

- **Другие сервисы:**
  Протоколы для обмена сообщениями, стриминга и других современных задач.

Другие сервисы в интернете используют специальные протоколы, которые обеспечивают обмен сообщениями, передачу потокового видео и аудио, а также выполнение других современных задач. Эти протоколы оптимизированы под конкретные требования, где важна скорость, реальное время или возможность взаимодействия множества участников.

Обмен сообщениями

- **Протоколы, например, XMPP (Extensible Messaging and Presence Protocol - Расширяемый протокол обмена сообщениями и присутствия):**
  Позволяют мгновенно отправлять текстовые сообщения, уведомления о присутствии (онлайн/офлайн статус) и даже передавать файлы.
  Жизненный пример:
  Представьте, что вы общаетесь с друзьями в мессенджере, как в разговоре по телефону, только текстом. Протокол XMPP обеспечивает мгновенную доставку сообщений, так что вы сразу узнаете, если друг начал печатать ответ.

Стриминг (передача видео и аудио)

- **Протоколы, например, RTP (Real-time Transport Protocol - Протокол передачи данных в реальном времени), RTSP (Real Time Streaming Protocol - Протокол потоковой передачи в реальном времени), HLS (HTTP Live Streaming — HTTP трансляция в реальном времени) и DASH (Dynamic Adaptive Streaming over HTTP - Динамическая адаптивная потоковая передача по HTTP):**
  Эти протоколы используются для передачи мультимедийного контента (видео и аудио) в режиме реального времени. Они обеспечивают адаптивное качество, подстраивающееся под скорость интернета, и минимизируют задержки, чтобы воспроизведение происходило почти мгновенно.
  Жизненный пример:
  Представьте, что вы смотрите прямую трансляцию спортивного матча. Даже если в сети случаются незначительные перебои, протоколы стриминга позволяют видео продолжать показываться, а качество автоматически подстраивается под текущую скорость подключения, чтобы вы не пропустили важный момент.

Голосовая связь и видеоконференции

- **Протоколы, например, SIP (Session Initiation Protocol - Протокол инициирования сеанса) и WebRTC (Web Real-Time Communication - Веб-коммуникация в реальном времени):**
  Эти протоколы позволяют организовать голосовые и видеозвонки через интернет. Они минимизируют задержки, обеспечивая качественную связь в реальном времени, что особенно важно для онлайн-конференций, удалённой работы и общения.
  Жизненный пример:
  Представьте видеоконференцию, где участники находятся в разных городах, но общаются, как будто сидят в одной комнате. WebRTC позволяет передавать видео и аудио через браузер без установки дополнительного ПО, обеспечивая плавное и

непрерывное общение.

Другие современные задачи

- **Протоколы обмена данными в реальном времени, например, для игр или совместной работы:**
  Здесь могут использоваться специальные UDP-базированные решения, где приоритет отдается минимальной задержке, даже если допустима небольшая потеря данных.
  Жизненный пример:
  В онлайн-играх каждое мгновение имеет значение. Протоколы, оптимизированные для игр, позволяют быстро передавать данные о действиях игроков, обеспечивая плавный игровой процесс, даже если некоторые пакеты теряются без критичных последствий.

Итог

- **Протоколы обмена сообщениями (например, XMPP)** позволяют мгновенно передавать текст и файлы, как если бы вы писали SMS или использовали чат в мессенджере.
- **Протоколы стриминга (например, RTP, HLS)** обеспечивают передачу видео и аудио в режиме реального времени, как просмотр прямой трансляции, где качество автоматически адаптируется под условия сети.
- **Протоколы голосовой связи и видеоконференций (например, SIP, WebRTC)** делают возможными звонки и конференции, обеспечивая общение без значительных задержек.
- **Протоколы для других задач (например, для онлайн-игр)** оптимизированы для минимальной задержки, что критично для интерактивного взаимодействия.

Таким образом, различные прикладные протоколы выполняют специфические задачи в интернете, обеспечивая удобство, скорость и эффективность обмена информацией в разных ситуациях, как специализированные службы в городе, каждая из которых делает свою работу наилучшим образом.

9. Основы безопасности компьютерных сетей

- **Угрозы и риски:**
  Обзор основных кибератак (DDoS, MITM, фишинг) и их влияния на сети.

Угрозы и риски в сети – это те опасности, которые могут нарушить работу сети или украсть данные. Рассмотрим несколько основных видов кибератак и их влияние на сети с жизненными примерами.

---

1. DDoS (Distributed Denial of Service - Распределенный отказ в обслуживании)

- **Что это:**
  DDoS-атака представляет собой массовую отправку запросов от множества заражённых компьютеров (ботнет), что приводит к перегрузке сервера или сети.
- **Влияние:**
  Из-за огромного количества запросов легитимные пользователи не могут получить доступ к сервису – сайт может «упасть», и работа нарушается.
- **Жизненный пример:**
  Представьте, что к входу в магазин одновременно подходят сотни людей(подставные лица, они не будут ничего покупать, но отвлекут на себя ресурсы магазина), а в магазине есть ограниченное количество сотрудников, способных обслуживать клиентов, но в меньшем количестве. В результате нормальные покупатели не могут войти, а магазин не может обслужить всех – это аналог перегрузки сервера.

---

2. MITM (Man-In-The-Middle, атака «человек посередине»)

- **Что это:**
  При MITM-атаке злоумышленник тайно перехватывает и, возможно, изменяет данные, передаваемые между двумя сторонами.
- **Влияние:**
  Атакующий может украсть конфиденциальную информацию (например, пароли или банковские данные) или изменить передаваемые данные, нарушая доверие между пользователем и сервисом.
- **Жизненный пример:**
  Представьте, что вы отправляете письмо своему другу через почту. Но на полпути ваше письмо перехватывает кто-то, меняет его содержание или читает его, а затем пересылает другу. Вы и ваш друг уже не уверены, что получили оригинальное сообщение – вот так работает MITM.

---

3. Фишинг

- **Что это:**
  Фишинг – это мошенническая атака, при которой злоумышленники рассылают поддельные сообщения или создают копии сайтов, чтобы обманным путём заставить вас ввести личные данные, такие как пароли, номера карт и логины.
- **Влияние:**
  После получения ваших данных мошенники могут использовать их для кражи денег,

доступа к вашим аккаунтам или другой незаконной деятельности.

- **Жизненный пример:**
  Представьте, что вы получаете письмо, якобы от вашего банка, с просьбой подтвердить ваши данные для «безопасности». Письмо выглядит очень убедительно, и вы вводите свои данные, думая, что помогаете защитить свой счёт. На самом деле, это ловушка, и мошенники теперь получают доступ к вашему счёту.

Итог

- **DDoS-атака** перегружает сеть, как если бы к магазину подступило слишком много людей, и никто не смог попасть внутрь.
- **MITM-атака** позволяет злоумышленнику подслушивать и изменять сообщения между сторонами, как если бы кто-то подменял письма в почтовой службе.
- **Фишинг** обманывает пользователей, заставляя их раскрывать личную информацию, как если бы вам прислали поддельное письмо от банка с просьбой подтвердить данные.

Эти угрозы могут серьёзно нарушить работу сетей и поставить под угрозу безопасность ваших данных, поэтому важно использовать антивирусы, обновлять программное обеспечение и быть внимательным к подозрительным сообщениям и сайтам.

- **Методы защиты:**
  Шифрование, VPN, файрволы – что это такое и как они работают.

Методы защиты помогают охранять ваши данные от несанкционированного доступа, подслушивания и других угроз. Рассмотрим три основных метода защиты: шифрование, VPN и файрволы, с простыми примерами.

---

1. Шифрование

- **Что это такое:**
  Шифрование превращает ваши данные в набор символов, который можно прочитать только при наличии специального ключа. Это как если бы вы писали письмо на секретном языке, понятном только вам и получателю.
- **Как работает:**
  Когда вы отправляете сообщение, оно преобразуется в «код», который нельзя прочитать без ключа. Только тот, кто знает ключ, может его расшифровать и понять, что именно было отправлено.
- **Жизненный пример:**
  Представьте, что вы отправляете секретное письмо другу, написанное на выдуманном языке. Даже если кто-то перехватит это письмо, он не сможет понять, о чём оно, потому что не знает секретного языка. Только ваш друг, который знает этот язык, сможет прочитать письмо.

---

2. VPN (Vertual Private Network - Виртуальная Частная Сеть)

- **Что это такое:**
  VPN создаёт защищённое соединение (или «туннель») между вашим устройством и сервером VPN. Через этот туннель все ваши данные шифруются, и ваш реальный IP-адрес скрывается.
- **Как работает:**
  Ваши данные сначала проходят через зашифрованный туннель до VPN-сервера, а затем уже попадают в интернет. Это помогает защитить вашу личную информацию, особенно при использовании общественных Wi-Fi сетей.
- **Жизненный пример:**
  Представьте, что вам нужно доехать до важного места, но вы не хотите, чтобы посторонние видели, куда именно вы направляетесь. Вместо обычного автомобиля вы садитесь в лимузин с тонированными стеклами, в котором никто не может разглядеть, кто внутри и куда вы едете. Таким образом, ваш маршрут и цель остаются скрытыми от посторонних глаз.

VPN делает примерно то же самое для ваших данных:

- **Защищённый канал:** Ваш интернет-трафик проходит через «черный лимузин» — зашифрованный туннель, благодаря чему никто не может увидеть, какие сайты вы посещаете или откуда подключаетесь.
- **Скрытие информации:** Ваш реальный IP-адрес скрывается, как если бы ваш автомобиль не выдавал номерной знак, что делает вашу активность в сети анонимной.

Таким образом, VPN позволяет вашим данным перемещаться по сети скрытно и безопасно,

как если бы вы ехали в лимузине с тонированными стеклами, не раскрывая свою личность и маршрут.

---

3. Файрвол (Firewall – Огненная стена. Брандмауэр)

- **Что это такое:**
  Файрвол – это программа или устройство, которое фильтрует входящий и исходящий сетевой трафик. Он блокирует подозрительные или нежелательные подключения и разрешает только безопасные.
- **Как работает:**
  Файрвол проверяет все запросы, которые поступают в вашу сеть, и решает, какие из них допустимы, а какие – нет, основываясь на заранее заданных правилах.
- **Жизненный пример:**
  Представьте здание с охранником у входа. Охранник проверяет документы каждого, кто пытается войти, и пускает только тех, кто имеет разрешение. Если кто-то пытается проникнуть без нужного пропуска, охранник его не пускает. Так файрвол защищает вашу сеть, блокируя несанкционированные подключения.

Итог

- **Шифрование** превращает данные в непонятный код, который могут расшифровать только те, кто знает секретный ключ (как секретный язык).
- **VPN** создаёт защищённый туннель для ваших данных и скрывает ваш реальный IP, как будто ваш разговор проходит через специальное защищённое пространство.
- **Файрволы** фильтруют трафик, разрешая только безопасные подключения, как охранник на входе в здание.

Эти методы вместе помогают защитить вашу личную информацию и обеспечить безопасность работы в интернете.

- **Аутентификация и управление доступом:**
  Способы идентификации пользователей и защита информации в сети.

Аутентификация и управление доступом – это два взаимосвязанных механизма, которые помогают защитить информацию в сети, удостоверяясь, кто вы, и определяя, что вам разрешено делать.

Аутентификация

Что это такое:
Аутентификация — это процесс проверки, что вы действительно тот, за кого себя выдаёте. Обычно это происходит с помощью ввода пароля, использования отпечатка пальца, распознавания лица, смс-кода или пропуска.

Жизненный пример:
Представьте, что вы приходите в офисное здание. Прежде чем войти, на входе установлен турникет или охрана, которые требуют предъявить вашу пропускную карточку или удостоверение личности. Если вы показываете правильный пропуск, система подтверждает вашу личность и позволяет войти. Это и есть аутентификация.

Управление доступом

Что это такое:
После того как система удостоверилась, кто вы, управление доступом определяет, какие ресурсы и данные вам разрешено использовать. Это набор правил, который ограничивает или предоставляет доступ к определённым функциям или информации.

Жизненный пример:
Когда вы уже вошли в офисное здание, не все двери открыты для каждого сотрудника. Например, дверь в вашу рабочую зону открыта, но в серверную комнату могут входить только IT-специалисты. Даже если кто-то прошёл проверку на входе (аутентификация), доступ к определённым помещениям ограничен в зависимости от его должности и прав. Это и есть управление доступом.

Общая картина

- **Аутентификация** — это как проверка вашего пропуска на входе в здание, которая удостоверяет, что вы действительно сотрудник.
- **Управление доступом** — это система, которая после входа определяет, какие комнаты или ресурсы доступны именно вам. Например, вы можете попасть в ваш офис, но не сможете войти в серверную, доступ к которой ограничен.

Таким образом, аутентификация и управление доступом работают вместе, чтобы обеспечить безопасность: сначала система узнаёт, кто вы, а затем предоставляет вам доступ только к тем ресурсам, к которым у вас есть разрешение. Это помогает защитить важную информацию от несанкционированного доступа.

10. Мониторинг, управление и диагностика сети

- **Инструменты для диагностики:**
  Программы и утилиты (ping, traceroute, Wireshark) для анализа работы сети.

Инструменты для диагностики сети помогают понять, как работает ваша сеть, обнаружить проблемы и узнать, где именно может быть сбой. Вот несколько популярных программ и утилит:

---

1. Ping

- **Что это такое:**
  Команда «ping» отправляет небольшой пакет данных на другой компьютер или сервер и измеряет, сколько времени требуется для ответа.
- **Как помогает:**
  Если вы получаете ответ быстро, значит связь хорошая. Если нет – может быть проблема в соединении.
- **Жизненный пример:**
  Представьте, что вы стучите в дверь соседа, чтобы узнать, дома ли он. Если сосед открывает дверь сразу, связь отличная. Если долго ждёте или ответа нет – возможно, сосед отсутствует или дверь закрыта.

Как пользоваться командой "ping":

1. **Открыть командную строку или терминал:**
   - На Windows: нажмите **Win + R**, введите `cmd` и нажмите Enter.
   - На macOS или Linux: откройте приложение «Терминал».
2. **Введите команду с нужным адресом:**
   Введите команду в следующем формате:

   ```
   ping [адрес]
   ```

   где `[адрес]` — это IP-адрес или доменное имя сервера.
   Например, чтобы проверить соединение с Google, введите:

   ```
   ping google.com
   ```

3. **Анализ результатов:**
   После ввода команды ваш компьютер отправит несколько небольших пакетов данных (запросов) на указанный адрес.
   Вы увидите строки с информацией, примерно так:

   ```
   Ответ от 142.250.190.78: число байт=32 время=14мс TTL=115
   ```

   Это означает, что запрос достиг сервера, сервер ответил, и время отклика составило 14 миллисекунд.

Жизненный пример:

Представьте, что вы хотите узнать, открыт ли офис вашего друга. Вы идете к его двери (это

как отправка запроса с помощью команды ping) и стучите (это как отправка пакета данных). Если он отвечает, значит, он дома (получен ответ). Команда ping делает нечто похожее: она «стучится» по сети и ждет ответа, чтобы убедиться, что связь установлена и работает.

Таким образом, команда ping помогает быстро проверить, доступен ли удалённый компьютер или сервер и как быстро происходит обмен данными между вашим устройством и целевым ресурсом.

---

2. Traceroute (или tracert)

- **Что это такое:**
  Эта утилита показывает маршрут, который проходит пакет данных от вашего компьютера до конечного пункта назначения. Она показывает все промежуточные узлы (маршрутизаторы), через которые проходят данные.
- **Как помогает:**
  Вы можете увидеть, где возникают задержки или потери данных. Это помогает понять, на каком участке маршрута происходит сбой.
- **Жизненный пример:**
  Представьте, что вы отправляете посылку и хотите узнать, через какие почтовые отделения она проходит. Traceroute покажет вам последовательность отделений, и если посылка задерживается на каком-то конкретном этапе, вы сразу заметите, где возникла проблема.

Утилиты `traceroute` для Linux и `tracert` для Windows являются стандартными инструментами, предустановленными в соответствующих операционных системах.

Для Windows: Утилита `tracert` уже включена в систему. Чтобы использовать её, откройте командную строку и введите `tracert`, за которым следует адрес назначения.

Руководство: https://learn.microsoft.com/ru-ru/windows-server/administration/windows-commands/tracert

Для Linux: Утилита `traceroute` обычно предустановлена. Если её нет, вы можете установить её через менеджер пакетов вашей дистрибуции. Например, для Debian/Ubuntu используйте команду `sudo apt-get install traceroute`.

Руководство: https://packages.debian.org/ru/sid/traceroute

Для iOS: На устройствах с iOS (iPhone, iPad) утилита `traceroute` не предустановлена. Однако вы можете воспользоваться сторонними приложениями из App Store, такими как "Network Analyzer" или "iNetTools", которые предоставляют функциональность `traceroute`.

Если вы ищете дополнительные возможности, вы можете рассмотреть использование `MTR` (My Traceroute) — инструмента, объединяющего функции `traceroute` и `ping`. Он доступен для Linux и Windows. Для Linux его можно установить через менеджер пакетов, например, командой `sudo apt-get install mtr`. Для Windows существует версия с графическим интерфейсом под названием `WinMTR`, доступная для загрузки по ссылке:
https://winmtr.ru/
https://winmtr.net/

Обратите внимание, что для macOS (операционной системы для компьютеров Mac) утилита `traceroute` также предустановлена и доступна через терминал.

Вы можете найти полезную информацию в обсуждениях на официальном форуме поддержки Apple: https://discussions.apple.com/thread/7116347?sortBy=rank

---

3. Wireshark

- **Что это такое:**
  Wireshark – это мощный анализатор сетевого трафика, который позволяет «перехватывать» и изучать пакеты данных, проходящие через вашу сеть.
- **Как помогает:**
  С его помощью можно увидеть, какие данные передаются, выявить подозрительную активность или ошибки, а также проанализировать работу протоколов.
- **Жизненный пример:**
  Представьте, что вы смотрите запись с камер наблюдения, чтобы понять, кто проходил через вход в здание и когда. Wireshark позволяет вам «просмотреть» все данные, которые передаются по сети, и выявить, если что-то идёт не так, как должно.

Руководство: https://habr.com/ru/articles/735866/

Скачать: https://www.wireshark.org/

Итог

- **Ping** проверяет, доступен ли другой компьютер и насколько быстро он отвечает – как стук в дверь.
- **Traceroute** показывает, какой путь проходит информация от вас до цели, помогая обнаружить, где возникают задержки – как отслеживание маршрута посылки.
- **Wireshark** анализирует сами данные, позволяя увидеть, что именно передается по сети – как просмотр записи с камер, чтобы понять, кто и что делает.

Эти инструменты позволяют администраторам и специалистам по безопасности оперативно выявлять и устранять проблемы в работе сети, делая её более стабильной и безопасной.

- **Протоколы мониторинга:**
  SNMP, NetFlow – как администраторы отслеживают состояние сети.

Протоколы мониторинга помогают администраторам следить за состоянием сети, обнаруживать проблемы и анализировать её работу. Два популярных протокола в этой области – SNMP и NetFlow. Давайте разберёмся, как они работают, на простых примерах.

SNMP (Simple Network Management Protocol - Простой протокол сетевого управления)

- **Что это такое:**
  SNMP – это протокол, который позволяет администраторам собирать информацию о состоянии устройств в сети (маршрутизаторов, коммутаторов, серверов и т.д.). Он помогает получать данные о нагрузке, ошибках, использовании ресурсов и прочем.
- **Как работает:**
  Администратор настраивает систему управления, которая периодически опрашивает устройства в сети. Эти устройства, в свою очередь, отправляют информацию о своей работе, используя SNMP-сообщения. Если что-то идет не так, система может сгенерировать тревогу.
- **Жизненный пример:**
  Представьте, что у вас есть умный дом с датчиками температуры, движения и освещения. Специальное приложение регулярно получает данные с этих датчиков и сообщает вам, если, например, в комнате слишком жарко или если кто-то вошёл в дом. SNMP работает по похожему принципу, собирая «показания» от сетевых устройств, чтобы администратор всегда знал, что происходит в сети.

NetFlow(сетевой поток)

- **Что это такое:**
  NetFlow – это протокол, который собирает информацию о потоках данных в сети. Он фиксирует, откуда и куда идут данные, сколько трафика проходит через конкретные узлы и как долго длится соединение.
- **Как работает:**
  Устройства в сети, поддерживающие NetFlow, записывают сведения о каждом «потоке» данных – например, IP-адрес отправителя, получателя, объем переданных данных и время соединения. Эти данные отправляются на центральный сервер, где администратор может анализировать, как используется сеть, обнаруживать аномалии и оптимизировать её работу.
- **Жизненный пример:**
  Представьте, что на въезде в город установлены счётчики, которые фиксируют, сколько автомобилей проезжает по каждому маршруту, откуда они приехали и куда направляются. Анализируя эту информацию, городские службы могут определить, где возникают пробки или аварийные ситуации, и принять меры. Точно так же NetFlow собирает информацию о «трафике» в сети, позволяя выявлять, например, участки с высокой нагрузкой или подозрительную активность.

Итог

- **SNMP** работает как система мониторинга умного дома: она опрашивает устройства, получает от них данные и предупреждает администратора о возможных проблемах.
- **NetFlow** напоминает дорожные счётчики: он фиксирует потоки данных, анализирует их и помогает понять, как и где используется сеть.

Эти инструменты позволяют администраторам оперативно выявлять и устранять проблемы, поддерживать стабильную работу сети и улучшать её производительность.

- **Практические случаи:**
  Разбор типичных проблем и способов их решения.

Давайте рассмотрим несколько типичных проблем в сети и способы их решения на понятных примерах.

---

1. Нет доступа к интернету

- **Проблема:**
  Вы открываете браузер, но никакой сайт не загружается.
- **Возможные причины:**
  - Неправильное подключение кабелей или неисправность оборудования.
  - Проблемы на стороне интернет-провайдера.
- **Диагностика и решение:**
  - **Шаг 1:** Используйте команду **ping** для проверки доступности какого-либо известного сайта (например, `ping google.com`).
    - Если ответ не получен, возможно, дело в подключении.
  - **Шаг 2:** Проверьте, правильно ли подключены модем и роутер, и попробуйте их перезагрузить.
  - **Шаг 3:** Если проблема сохраняется, свяжитесь с технической поддержкой вашего провайдера.
- **Жизненный пример:**
  Это как если бы в вашем доме отключили электричество. Вы проверяете, работает ли счетчик (используете ping), затем проверяете, не выбило ли предохранитель (перезагружаете оборудование). Если ничего не помогает, звоните электрику (технической поддержке).

---

2. Слабый или нестабильный сигнал Wi-Fi

- **Проблема:**
  Ваш смартфон или ноутбук теряет связь, или интернет работает медленно в одной части дома.
- **Возможные причины:**
  - Роутер находится слишком далеко от устройства.
  - Помехи от других сетей или бытовой техники.
- **Диагностика и решение:**
  - **Шаг 1:** Проверьте уровень сигнала на устройстве – попробуйте переместиться ближе к роутеру.
  - **Шаг 2:** Измените расположение роутера в доме, поставьте его в центральное место, чтобы сигнал равномерно покрывал всю территорию.
  - **Шаг 3:** Попробуйте сменить канал Wi-Fi, чтобы уменьшить помехи от соседних сетей.
- **Жизненный пример:**
  Это как если бы вы пытались поговорить по мобильному телефону в подвале – сигнал слабый. Перейдя на верхний этаж или ближе к окну, вы начнете слышать собеседника

четче.

---

3. Конфликт IP-адресов

- **Проблема:**
  Некоторые устройства в сети не могут подключиться к интернету, потому что у них совпадают IP-адреса.
- **Возможные причины:**
  - Два устройства получают один и тот же IP-адрес при статической настройке или сбое DHCP(Dynamic Host Configuration Protocol - Протокол динамической конфигурации хоста).
- **Диагностика и решение:**
  - Проверьте настройки сети на каждом устройстве.
  - Если конфликт обнаружен, измените настройки так, чтобы каждому устройству выдавался уникальный IP-адрес (например, используйте автоматическую настройку через DHCP).
- **Жизненный пример:**
  Это как если два человека пытаются зарегистрироваться по одному и тому же адресу – почтовая служба не знает, кому доставлять письма. Решением будет уточнение адресов, чтобы у каждого был свой уникальный адрес.

---

4. Проблемы с DNS (Domain Name System)

- **Проблема:**
  Вы вводите адрес сайта, но он не открывается, хотя подключение к интернету есть.
- **Возможные причины:**
  - DNS-сервер, который вы используете, не отвечает или работает с ошибками.
- **Диагностика и решение:**
  - Попробуйте открыть сайт по IP-адресу. Если сайт откроется, проблема в DNS.
  - Измените настройки DNS в вашем роутере или устройстве, используя, например, общедоступные серверы Google (8.8.8.8) или Cloudflare (1.1.1.1).
- **Жизненный пример:**
  Это как если вы спрашиваете у местного жителя дорогу, но он дает неверный адрес. Переключившись на другого человека (другой DNS-сервер), вы получаете правильную информацию и находите нужное место.

Итог

Практические случаи в сетях похожи на повседневные ситуации:

- **Нет доступа к интернету** – как отсутствие электричества в доме: проверьте оборудование и обратитесь к специалистам, если ничего не помогает.
- **Слабый Wi-Fi сигнал** – как плохой мобильный сигнал: переместитесь ближе к источнику или настройте оборудование.

- **Конфликт IP-адресов** – как два человека с одинаковым адресом: каждому устройству нужен уникальный адрес.
- **Проблемы с DNS** – как получение неверного адреса от местного жителя: смените источник информации.

Эти примеры показывают, что диагностика и решение сетевых проблем включают проверку соединения, корректную настройку оборудования и обращение за помощью, если самостоятельно разобраться не удается.

11. Современные тенденции и перспективы развития сетей

- **Интернет вещей (IoT):**
  Как подключение бытовых устройств меняет ландшафт сетевых технологий.

Интернет вещей (IoT - Internet of Things) – это концепция, согласно которой повседневные устройства становятся «умными» благодаря подключению к интернету. Такие устройства, как умные холодильники, термостаты, лампочки, камеры и даже бытовая техника, могут обмениваться информацией, работать вместе и управляться удалённо. Это меняет ландшафт сетевых технологий, расширяя количество подключённых устройств, создавая новые возможности для автоматизации и требуя более продвинутых мер безопасности.

Как это работает:

- **Подключение устройств:**
  Благодаря встроенным датчикам и микропроцессорам бытовые приборы получают возможность выходить в интернет. Они могут отправлять данные о своей работе (например, температуру, состояние, уровень энергии) и получать команды от пользователя или других устройств.
- **Обмен информацией:**
  Устройства обмениваются данными через сеть. Например, умный термостат может получать данные о погоде, а система освещения – информацию о присутствии людей в комнате, чтобы включаться или выключаться автоматически.
- **Централизованное управление:**
  С помощью специальных приложений или облачных платформ пользователи могут контролировать и настраивать работу всех подключённых устройств из одного места. Это повышает удобство и эффективность управления домом, офисом или производственными процессами.

Жизненный пример: Умный дом

Представьте себе обычный дом, где все приборы подключены к интернету:

- **Умный термостат** автоматически регулирует температуру в доме, основываясь на данных о погоде и вашем расписании, экономя энергию и повышая комфорт.
- **Умные лампочки** включаются и выключаются по вашему голосовому или мобильному управлению, а также могут реагировать на изменение освещённости в комнате.
- **Умный холодильник** отслеживает запасы продуктов и даже может предложить рецепты, основываясь на имеющихся ингредиентах.
- **Система безопасности** с камерами и датчиками движения мгновенно отправляет уведомления на смартфон, если обнаруживает подозрительную активность.

Все эти устройства общаются между собой и с вами через интернет, создавая единую систему, которая делает жизнь удобнее и эффективнее.

Итог

Интернет вещей (IoT) преобразует обычные бытовые устройства в интеллектуальные помощники, которые:

- Подключаются к сети и обмениваются данными.
- Автоматически реагируют на изменения условий.
- Управляются централизованно для повышения удобства и эффективности.

Так, благодаря IoT, наш дом, офис и даже целые города становятся «умнее», что открывает новые возможности для комфорта, экономии ресурсов и безопасности.

- **Программно-определяемые сети (SDN):**
  Концепция управления сетью с помощью программного обеспечения.

Программно-определяемые сети (SDN - Software-Defined Network) – это современный подход к управлению компьютерными сетями, при котором весь контроль и настройка сети осуществляются централизованно с помощью программного обеспечения, а не на каждом устройстве отдельно.

Как это работает:

- **Централизованный контроль:**
  В традиционных сетях администратор настраивает каждый маршрутизатор, коммутатор и другое оборудование индивидуально. В SDN все эти устройства получают свои инструкции от одного центрального контроллера, который «говорит», как нужно вести себя всей сети.
- **Программное управление:**
  Центральный контроллер использует специальное программное обеспечение для мониторинга, анализа и управления потоками данных в сети. Это позволяет динамично менять настройки, оптимизировать маршруты, быстро реагировать на проблемы и даже автоматически исправлять сбои.
- **Разделение функций:**
  Физическое оборудование в SDN становится «простыми устройствами», а вся логика управления перемещается в программное обеспечение. Это делает сеть гибче, легче в управлении и обновлении.

Жизненный пример:

Представьте себе умный город:

- **Традиционный подход:**
  Каждый светофор в городе работает автономно и настроен вручную. Если на одной улице возникает пробка, изменение настроек каждого светофора занимает много времени и требует вмешательства специалистов.
- **SDN-подход:**
  Теперь представьте, что все светофоры города подключены к единой центральной системе управления. Эта система постоянно мониторит дорожную ситуацию: если в одном районе начинает скапливаться трафик, центральная система автоматически изменяет режим работы светофоров, чтобы перенаправить поток автомобилей и снять затор.
  Точно так же, в SDN центральный контроллер анализирует состояние сети и динамически перенастраивает маршруты, чтобы данные шли максимально быстро и эффективно.

Итог:

Программно-определяемые сети (SDN) делают управление сетью проще и гибче, перемещая всю логику управления в центральное программное обеспечение. Это похоже на умный

город, где все светофоры управляются централизованно для оптимизации дорожного движения, а не каждый по отдельности. Такой подход позволяет быстрее реагировать на изменения, повышать эффективность сети и легче масштабировать её при необходимости.

- **Будущее мобильных сетей:**
  Роль 5G и перспективы дальнейшего развития технологий связи.

Будущее мобильных сетей во многом связано с внедрением технологии 5G(5th Generation – Пятое Поколение), которая меняет представление о скорости и возможностях связи. Вот основные моменты, объяснённые простыми словами с жизненным примером:

Что такое 5G и чем оно отличается?

- **Более высокая скорость:**
  5G обещает значительно быстрее передавать данные по сравнению с предыдущими поколениями (например, 4G). Это значит, что загрузка видео, игр или файлов происходит почти мгновенно.
- **Низкая задержка:**
  Задержка – это время, которое требуется для передачи данных от отправителя к получателю. 5G снижает задержку до минимального уровня, что особенно важно для таких приложений, как онлайн-игры, видеоконференции и управление транспортными средствами в режиме реального времени.
- **Большая ёмкость:**
  5G позволяет подключать гораздо больше устройств одновременно. Это необходимо для развития Интернета вещей (IoT), где миллионы умных устройств – от датчиков в умном доме до автономных автомобилей – будут обмениваться данными.

Жизненный пример:

Представьте, что раньше вы ездили на автомобиле по обычной дороге с пробками (это была сеть 4G). Вас часто задерживали светофоры и заторы, а скорость движения зависела от дорожной ситуации. Теперь представьте, что у вас есть специальная автострада для вашего автомобиля, по которой можно ехать со скоростью, близкой к максимально возможной, без остановок и пробок – это и есть сеть 5G.

В такой сети:

- **Загрузка и стриминг:**
  Вы можете смотреть фильмы или играть в онлайн-игры без каких-либо задержек и зависаний.
- **Умные устройства:**
  Ваш умный дом, автомобиль, носимые гаджеты и даже системы управления городским транспортом будут работать как слаженный механизм, обмениваясь данными моментально.
- **Новые технологии:**
  Возможны новые приложения, такие как виртуальная и дополненная реальность, которые требуют сверхбыстрой и стабильной связи для реалистичного взаимодействия.

Перспективы дальнейшего развития:

Помимо 5G, ученые и инженеры уже работают над следующими шагами в эволюции

мобильных сетей:

- **6G и дальше:**
  Эти технологии обещают ещё большую скорость и ещё более низкую задержку, что может открыть двери для новых видов сервисов и приложений, о которых мы сегодня даже не мечтаем.
- **Интеграция с IoT:**
  Будущие сети будут ещё лучше интегрироваться с умными устройствами, делая города и дома по-настоящему «умными».
- **Развитие сетевой инфраструктуры:**
  Улучшение качества обслуживания, повышение безопасности и устойчивости сетей – всё это поможет обеспечить бесперебойное соединение даже в условиях высокой нагрузки.

Таким образом, 5G становится ключевым шагом в развитии мобильных сетей, предлагая невероятную скорость, минимальную задержку и возможность подключения огромного количества устройств. Это, как бы, ваша личная супер-автострада, которая делает все поездки быстрыми и комфортными, а в будущем технологии будут развиваться дальше, открывая новые горизонты для инновационных решений и улучшения качества жизни.

12. Заключение и рекомендации для дальнейшего изучения

- **Подведение итогов:**
  Основные выводы и ключевые понятия, которые необходимо усвоить.

Подведем итоги, выделив основные выводы и ключевые понятия, которые помогут вам понять, как работают компьютерные сети.

Основные идеи:

1. **Компьютерная сеть – это система для обмена информацией:**
   Представьте город, где каждое здание имеет свой уникальный адрес. Так же устройства в сети (компьютеры, смартфоны, серверы) используют уникальные адреса (IP, MAC) для связи между собой. Это позволяет им отправлять «письма» (данные) друг другу.
2. **Многоуровневая структура сети (модель OSI):**
   Сеть делится на уровни, каждый из которых выполняет свою задачу:
   - **Физический уровень:** отвечает за передачу «сырого» сигнала по кабелям или через радио.
   - **Канальный уровень:** отвечает за формирование кадров и проверку целостности данных (как упаковка и маркировка посылок).
   - **Сетевой уровень:** выбирает маршруты для данных (маршрутизатор работает на этом уровне, как диспетчер, выбирающий оптимальный путь для посылки).
   - **Транспортный уровень:** обеспечивает надежную доставку данных (например, TCP гарантирует, что все «письма» доставлены в нужном порядке, а UDP – быстро, хоть и не гарантированно).
   - **Прикладной уровень:** здесь работают программы, использующие сети для обмена информацией (например, браузеры, почтовые клиенты).
3. **Протоколы – это набор правил для обмена данными:**
   Например, протоколы HTTP/HTTPS используются для просмотра веб-страниц, FTP – для передачи файлов, SMTP и POP3/IMAP – для работы с электронной почтой, а DNS переводит удобные доменные имена в IP-адреса. Эти протоколы делают работу сети понятной и удобной для пользователей.
4. **Инструменты для диагностики и мониторинга:**
   Программы типа ping, traceroute и Wireshark помогают определить, насколько хорошо работает сеть, обнаружить и устранить проблемы. Администраторы используют протоколы SNMP и NetFlow для постоянного контроля за состоянием сети.
5. **Защита сети:**
   Методы защиты (шифрование, VPN, файрволы) работают как охранники и секретные каналы, защищая вашу информацию от несанкционированного доступа и подслушивания.
6. **Будущее технологий – IoT, SDN и 5G:**
   - **Интернет вещей (IoT):** превращает обычные бытовые устройства в «умные», которые могут общаться и работать вместе, как сотрудники в хорошо организованном офисе.
   - **Программно-определяемые сети (SDN):** позволяют централизованно управлять сетью с помощью программного обеспечения, делая её гибкой и быстрой, как умный город с централизованной системой управления.

- **5G и будущее мобильных сетей:** обещают еще большую скорость, минимальную задержку и возможность подключения огромного количества устройств, что откроет новые возможности в области онлайн-игр, видеоконференций и IoT.

Жизненный пример: Город и почтовая служба

Представьте себе большой город, где каждое здание имеет свой уникальный адрес.

- **Почтовая служба** (сеть) доставляет письма между зданиями (устройствами).
- **Маршрутизатор** – это как диспетчер, который направляет посылки по оптимальному маршруту через город.
- **Специализированные службы** (DNS, инструменты диагностики, системы безопасности) обеспечивают, чтобы письма не потерялись, доставлялись вовремя и оставались защищенными.

Итог

Чтобы работать с современными сетями, важно понять, что они – это сложная, многоуровневая система обмена данными, где каждая часть (от физического подключения до прикладных протоколов) играет свою роль. Знание основных понятий, таких как IP-адресация, маршрутизация, протоколы передачи (TCP, UDP), инструменты мониторинга и методы защиты, помогает обеспечить надежное и безопасное общение устройств в сети. В будущем такие технологии, как IoT, SDN и 5G, продолжат расширять возможности сетей, делая их еще быстрее, умнее и более интегрированными в нашу повседневную жизнь.

Таким образом, основное, что нужно усвоить – это понимание того, как информация перемещается в сети, как обеспечивается её надежность и безопасность, и какие новые технологии будут влиять на будущее связи.

- **Рекомендации для практики:**
  Советы по самостоятельному экспериментированию, настройке небольших сетей, участию в форумах и сообществах.

Рекомендации для практики – это советы, которые помогут вам углубить знания в области компьютерных сетей через самостоятельное экспериментирование и общение с экспертами.

Что можно сделать:

1. **Экспериментируйте с домашней сетью:**
   Попробуйте настроить собственную сеть, даже если она небольшая. Подключите компьютер, смартфон, принтер и другие устройства к Wi-Fi роутеру. Поэкспериментируйте с настройками IP-адресов, сменой каналов Wi-Fi, настройкой гостевой сети. Это поможет понять, как работают устройства в реальном времени.
2. **Используйте симуляторы и виртуализацию:**
   Программы вроде Cisco Packet Tracer, GNS3 или VirtualBox позволяют создать виртуальную сеть. Вы можете настроить маршрутизаторы, коммутаторы и серверы без необходимости покупать дорогое оборудование. Это отличный способ практиковаться, тестировать новые идеи и отрабатывать навыки.
3. **Изучайте инструменты диагностики:**
   Освойте базовые утилиты, такие как **ping**, **traceroute** и **Wireshark**. Они помогут вам видеть, как данные перемещаются по сети, и помогут выявить проблемы. Практическое использование этих инструментов укрепит ваши знания о работе сетей.
4. **Участвуйте в сообществах и форумах:**
   Присоединяйтесь к онлайн-сообществам и форумам, где обсуждают вопросы сетевых технологий – например, на Habr, Stack Overflow, специализированных группах в Telegram или форумах кибербезопасности. Общение с единомышленниками и опытными специалистами позволит получить ответы на вопросы и обменяться опытом.
5. **Читайте книги и смотрите обучающие видео:**
   Найдите литературу по сетям (например, по моделям OSI, протоколам маршрутизации и безопасности) и обучающие курсы. Видео на YouTube или специализированные онлайн-курсы помогут вам лучше понять сложные концепции через визуальные примеры.

Жизненный пример:

Представьте, что вы решили построить свой маленький город из игрушечных домиков:

- **Домики** – это ваши устройства (компьютеры, принтеры, смартфоны).
- **Улицы** – это сетевые кабели или Wi-Fi, по которым перемещаются данные.
- **Светофоры и дорожные знаки** – это настройки маршрутизаторов и коммутаторов, которые помогают организовать движение данных.
- **Почтовая служба** – это система передачи данных, где используются инструменты диагностики для отслеживания «посылок» (данных).
- **Общение с соседями** – это участие в форумах и сообществах, где вы обмениваетесь опытом по управлению вашим «городом».

Когда вы экспериментируете с разными способами организации вашего маленького города,

вы учитесь управлять сетью, понимаете, как решать проблемы и совершенствуете свои навыки.

Итог:

- **Самостоятельное экспериментирование:** Настройте свою домашнюю сеть или виртуальную лабораторию, чтобы применять теорию на практике.
- **Изучение инструментов:** Используйте утилиты диагностики для анализа работы сети.
- **Общение с экспертами:** Участвуйте в сообществах, где можно задать вопросы и получить советы.
- **Непрерывное обучение:** Читайте книги и смотрите обучающие материалы для углубления знаний.

Эти рекомендации помогут вам не только лучше понять, как работают компьютерные сети, но и приобрести практические навыки, необходимые для успешной работы в этой области.

- **Ресурсы и литература:**
  Список книг, онлайн-курсов, статей и видеоуроков для углубленного изучения темы.

Для углубленного изучения компьютерных сетей очень важно не только теоретическое чтение, но и практические примеры, видеоуроки и общение с единомышленниками. Вот несколько русскоязычных ресурсов, которые помогут вам освоить тему:

Книги

- **«Компьютерные сети» Эндрю Таненбаума**
  Классический учебник, переведённый на русский язык. Он подробно описывает все уровни модели OSI, протоколы, маршрутизацию и многое другое.
  Жизненный пример: Если вы хотите разобраться, как устроен автомобиль, вы сначала изучаете схему двигателя – так же книга Таненбаума помогает понять «двигатель» сети.
- **«Сетевые технологии. Принципы, технологии, протоколы»**
  Ещё один учебник, доступный на русском языке, который охватывает современные технологии передачи данных и сетевую безопасность.

Онлайн-курсы

- **SkillFactory**
  Российская образовательная платформа, где есть курсы по IT и сетевым технологиям. Здесь вы найдёте практические занятия и проекты, позволяющие применить теорию на практике.
  Жизненный пример: Представьте, что вы проходите кулинарный мастер-класс – тут вам не только показывают рецепт, но и дают возможность самому приготовить блюдо.
- **GeekBrains**
  Еще одна популярная платформа, предлагающая курсы по компьютерным сетям, кибербезопасности и другим IT-темам. Курсы часто включают реальные кейсы и проекты.
- **Stepik**
  Бесплатные курсы на русском языке, где можно найти основы компьютерных сетей, изучить протоколы и даже поработать с виртуальными лабораториями.
- **Coursera (русскоязычные курсы)**
  Некоторые университеты предлагают курсы по сетям с русскими субтитрами или переводом, что позволяет получать знания от ведущих мировых специалистов с удобной локализацией.

Статьи и видеоуроки

- **Habr (habr.com)**
  Огромное сообщество IT-специалистов, где регулярно публикуются статьи, гайды и обзоры по компьютерным сетям, кибербезопасности и современным технологиям.
  Жизненный пример: Это как читать отзывы о новой модели смартфона – здесь эксперты делятся практическими советами и разбирают реальные кейсы.

- **YouTube-каналы**
  Каналы вроде «ITProger», «LoftBlog» и другие регулярно выкладывают видеоуроки и вебинары по темам, связанным с сетями. Видео позволяют увидеть на практике, как настроить оборудование, проводить диагностику и устранять неполадки.

Общение и практический опыт

- **Форумы и сообщества**
  Помимо чтения и просмотра уроков, полезно общаться на форумах, таких как разделы на Habr, специализированные группы в Telegram или ВКонтакте, где можно задать вопросы и поделиться опытом.
- **Практические лаборатории**
  Используйте симуляторы (например, Cisco Packet Tracer или GNS3) для моделирования сетевых инфраструктур. Это как собирать конструктор, где на практике можно увидеть, как работают разные компоненты сети.

Итог

Чтобы глубже понять, как устроены компьютерные сети, изучайте теорию с помощью классических книг (например, Таненбаума), проходите онлайн-курсы на таких платформах, как SkillFactory и GeekBrains, и регулярно следите за свежими материалами на Habr и YouTube. Такое сочетание теории и практики позволит вам не только усвоить основные понятия, но и применять знания в реальных условиях, как если бы вы учились управлять настоящим автомобилем, начиная с теории устройства двигателя и заканчивая практическим вождением.

Перейти в оглавление