1. Протокол TCP (Transmission Control Protocol) является одним из основных протоколов интернета, предназначенным для обеспечения надежной доставки данных между узлами в сети. Основные свойства протокола TCP включают:

* Надежность: TCP обеспечивает надежную доставку данных путем использования подтверждений и повторной передачи данных в случае потери или повреждения пакетов.
* Управление потоком: TCP контролирует поток данных между отправителем и получателем, чтобы предотвратить перегрузку получателя.
* Управление нагрузкой: TCP использует механизмы для управления пропускной способностью сети и адаптации к изменяющимся условиям сети.
* Установление соединения: TCP использует трехэтапное установление соединения (handshake) между отправителем и получателем перед началом передачи данных.
* Гарантированная доставка: TCP гарантирует, что данные будут доставлены в правильном порядке и без потерь.
* Отправка потока данных: TCP упорядочивает и передает данные в виде потока, разбивая их на пакеты и восстанавливая исходный поток на получающей стороне.

1. Процедура установки TCP-соединения, называемая трехэтапным рукопожатием (three-way handshake), состоит из следующих шагов:

* Шаг 1: Отправитель (клиент) отправляет пакет с установкой соединения (SYN) на получателя (сервер).
* Шаг 2: Получатель получает пакет SYN, генерирует свой собственный пакет SYN и ACK (подтверждение), и отправляет его обратно отправителю.
* Шаг 3: Отправитель получает пакет SYN-ACK, отправляет подтверждение ACK получателю, и установка соединения завершается.

Процедура закрытия TCP-соединения состоит из шагов:

* Шаг 1: Одна из сторон отправляет пакет с флагом FIN (завершение) другой стороне для инициирования закрытия соединения.
* Шаг 2: Получатель получает пакет FIN и отправляет подтверждение ACK.
* Шаг 3: Получатель также отправляет свой собственный пакет FIN для закрытия соединения со своей стороны.
* Шаг 4: Отправитель получает пакет FIN и отправляет подтверждение ACK.
* Шаг 5: Закрытие соединения завершается, когда обе стороны подтверждают получение флага FIN и ACK.

1. В компьютерных сетях порт - это числовой идентификатор, используемый для адресации конкретного процесса или службы, выполняющегося на узле (компьютере). Порт представляет собой 16-битное число и может принимать значения от 0 до 65535.

Порты делятся на две категории: зарегистрированные порты и эфемерные (или частные) порты. Зарегистрированные порты (от 0 до 1023) зарезервированы для известных служб и протоколов, таких как HTTP (порт 80) или FTP (порт 21). Эфемерные порты (от 1024 до 65535) используются динамически и могут быть назначены процессам и службам при их запуске.

Использование портов позволяет множеству процессов на одном узле обмениваться данными по сети, поскольку каждый процесс может быть идентифицирован по своему уникальному порту.

1. В контексте компьютерных сетей сокет (socket) - это программный интерфейс, который позволяет процессу взаимодействовать с сетевыми протоколами (например, TCP или UDP). Сокет обеспечивает абстракцию для отправки и получения данных через сетевое соединение.

Сокеты представляются в виде дескрипторов (числовых идентификаторов), которые привязываются к конкретному адресу и порту на узле. Процесс может создавать сокеты, подключаться к удаленным узлам или принимать входящие соединения через сокеты.

Сокеты могут быть использованы для реализации различных сетевых приложений, таких как веб-серверы, клиенты электронной почты, чат-приложения и другие приложения, которые требуют обмена данными через сеть.

1. Полудуплексный канал связи - это канал передачи данных, который позволяет передавать информацию в обоих направлениях, но только в одном направлении за раз. В полудуплексном режиме передачи данные могут перемещаться только в одном направлении за один промежуток времени, и необходимо переключаться между направлениями.

Примером полудуплексного канала связи может быть рационированная радиолиния, где передача данных осуществляется поочередно в одном направлении, а затем в другом направлении. Пользователь может либо передавать данные, либо принимать их, но не одновременно.

1. Дуплексный канал связи - это канал передачи данных, который позволяет одновременную двустороннюю передачу информации. В дуплексном режиме передачи данные могут перемещаться в обоих направлениях одновременно.

Примером дуплексного канала связи является полно-дуплексное соединение по протоколу TCP. В полно-дуплексном режиме как отправитель, так и получатель могут передавать данные независимо друг от друга, что позволяет одновременную двустороннюю коммуникацию.

1. Протоколы UDP (User Datagram Protocol) и TCP являются двумя основными протоколами транспортного уровня интернета. Они отличаются в следующих аспектах:

* Надежность доставки: TCP обеспечивает надежную доставку данных, гарантируя, что данные будут доставлены в правильном порядке и без потерь. UDP, с другой стороны, является ненадежным протоколом, который не гарантирует надежную доставку и может потерять пакеты или доставить их в неправильном порядке.

Установление соединения: TCP использует трехэтапное установление соединения (handshake) между отправителем и получателем перед началом передачи данных, в то время как UDP не требует установления соединения перед передачей данных. UDP является протоколом без установления соединения.

* Контроль потока и управление нагрузкой: TCP обеспечивает механизмы контроля потока и управления нагрузкой, чтобы предотвратить перегрузку получателя и адаптироваться к изменяющимся условиям сети. UDP не предоставляет встроенных механизмов для контроля потока или управления нагрузкой.
* Оверхед: Из-за своей надежности и механизмов управления TCP обладает большим объемом заголовков и дополнительной информации, что приводит к большему оверхеду по сравнению с UDP. UDP имеет более компактные заголовки, что делает его более эффективным в передаче небольших пакетов данных.
* Применение: TCP часто используется для приложений, где надежная доставка данных и контроль потока являются критическими, таких как веб-страницы, файловые передачи, электронная почта и другие приложения, где целостность данных имеет высокое значение. UDP обычно используется в приложениях, где небольшая задержка более важна, чем надежная доставка данных, таких как потоковое видео, голосовая связь и игры в реальном времени.