**1. IP это?**

IP (Internet Protocol) - это протокол сетевого уровня, который обеспечивает передачу данных между устройствами в сети. IP-адрес - это уникальный идентификатор устройства в сети, который состоит из четырех чисел, разделенных точками. Например, 192.168.0.1. Символ \*\. не имеет отношения к IP-адресу и может использоваться для поиска файлов по шаблону.

**2. Port это?**

Port - это логический канал для передачи данных между устройствами или программами. Он может быть физическим (например, USB-порт) или виртуальным (например, порт TCP/IP). Каждый порт имеет свой уникальный номер, который используется для идентификации канала связи. В операционной системе порты используются для обмена данными между приложениями и устройствами, а также для установления соединений между компьютерами в сети.

**3. Что такое TCP порт и чем он отличается от IP?**

TCP порт - это номер, который используется для идентификации конкретного приложения или сервиса, работающего на устройстве в сети TCP/IP. Он является частью адреса назначения в TCP/IP пакете. Протокол TCP использует порты для обеспечения надежной передачи данных между приложениями на разных устройствах в сети.

IP (Internet Protocol) - это протокол, который определяет способ передачи данных между устройствами в сети. Он не использует порты для идентификации конкретных приложений или сервисов, а только для идентификации устройств в сети. IP адреса используются для маршрутизации пакетов данных в сети.

Таким образом, TCP порт и IP адрес - это два разных понятия, которые используются в протоколе TCP/IP для обеспечения передачи данных между приложениями на разных устройствах в сети.

**4. Какие виды транспортного протокола сущетсвуют?**

Я знаю два вида транспортного протокола: TCP (Transmission Control Protocol) и UDP (User Datagram Protocol). TCP обеспечивает надежную передачу данных с гарантией доставки и контролем ошибок, а UDP обеспечивает быструю передачу данных без гарантии доставки и контроля ошибок.

TCP (Transmission Control Protocol) - это протокол транспортного уровня, который обеспечивает надежную передачу данных в сети. Он гарантирует доставку данных без потерь, дублирования или искажения. Для этого TCP использует механизмы контроля ошибок, а также управление потоком и управление сеансом.

TCP работает на основе установления соединения между двумя устройствами. Это означает, что передача данных начинается только после того, как обе стороны подтвердят готовность к передаче. Кроме того, TCP разбивает данные на пакеты и отправляет их по отдельности. После отправки каждого пакета TCP ожидает подтверждения от получателя, что пакет был успешно доставлен. Если подтверждение не получено, TCP повторно отправляет пакет.

TCP также обеспечивает управление потоком данных, чтобы избежать перегрузки сети. Для этого он использует механизмы управления окнами, которые позволяют отправителю контролировать количество передаваемых данных.

UDP (User Datagram Protocol) - это протокол транспортного уровня, который обеспечивает быструю передачу данных без гарантии доставки и контроля ошибок. UDP не устанавливает соединение между устройствами и не обеспечивает никаких механизмов контроля ошибок.

UDP просто отправляет данные в виде датаграмм без какой-либо гарантии их доставки. Это означает, что данные могут быть потеряны, дублированы или искажены в процессе передачи. Однако UDP очень быстрый и эффективный протокол, который часто используется для передачи потокового видео, аудио и других приложений, где скорость более важна, чем точность.

**5. Различия между стеком tcp/ip и моделью OSI?**

Стек TCP/IP и модель OSI - это две различные концепции, которые используются для описания сетевых протоколов и их взаимодействия. Основное отличие между ними заключается в том, что стек TCP/IP является более реалистичной и практичной моделью, используемой в реальных сетях, в то время как модель OSI является более теоретической и используется в учебных целях.

Стек TCP/IP состоит из четырех уровней: уровень сетевого доступа, уровень интернета, транспортный уровень и прикладной уровень. Каждый уровень выполняет свою функцию в передаче данных в сети, начиная от физической передачи до обработки приложений.

Модель OSI состоит из семи уровней: физический уровень, канальный уровень, сетевой уровень, транспортный уровень, сеансовый уровень, уровень представления и прикладной уровень. Каждый уровень также выполняет свою функцию в передаче данных в сети, но модель OSI более абстрактна и менее применима к реальным сетям.

Одним из основных отличий между стеком TCP/IP и моделью OSI является то, что стек TCP/IP не имеет отдельного уровня для управления сеансами и уровня представления, в то время как модель OSI включает эти уровни. Кроме того, стек TCP/IP использует более простой подход к управлению сетью, который позволяет более эффективно работать в реальных условиях.

В целом, стек TCP/IP является более практичной и реалистичной моделью, которая используется в реальных сетях, в то время как модель OSI является более теоретической и используется в учебных целях.

**6. Из чего обычно состоит Web-сервис? (3 самых главных "модуля")**

Web-сервис обычно состоит из трех главных модулей:

1. Протокол передачи данных - это протокол, который определяет, как данные будут передаваться между клиентом и сервером. Наиболее распространенными протоколами являются HTTP (HyperText Transfer Protocol) и HTTPS (HTTP Secure).

2. Серверное приложение - это программа, которая обрабатывает запросы от клиентов и возвращает им ответы. Это может быть написано на разных языках программирования, таких как Java, Python, Ruby и другие.

3. База данных - это хранилище данных, которое используется для хранения информации, которая будет использоваться серверным приложением. Наиболее распространенными базами данных являются MySQL, PostgreSQL и MongoDB.

**7. Через какой язык происходит взаимодействие с командной строкой Linux?**

Взаимодействие с командной строкой Linux происходит через язык командного интерпретатора (shell), наиболее распространенным из которых является bash.

**8. Через какой язык происходит взаимодействие с командной строкой Linux?**

bash

SSH (Secure SHell - защищенная оболочка) — сетевой протокол прикладного уровня, предназначеный для безопасного удаленного доступа к UNIX-системам. Данный протокол эффективен тем, что шифрует всю передаваемую информацию по сети. По умолчанию, используется 22-й порт. В основном он нужен для удаленного управления данными пользователя на сервере, запуска служебных команд, работы в консольном режиме с базами данных.

Внутренним (локальным). Так называют адрес, используемый только в пределах одной локальной сети. Внешним (реальным, глобальным). Это уникальный адрес, под которым компьютер виден прочим устройствам в глобальной сети.

Гипервизор - это программное обеспечение или слой аппаратуры, которое позволяет запускать и управлять виртуальными машинами (ВМ) на физическом сервере. Он работает непосредственно на аппаратном обеспечении и предоставляет виртуализацию ресурсов, таких как процессоры, память, хранилище и сеть, для различных ВМ, которые могут выполнять различные операционные системы и приложения.

Гипервизор разделяет физические ресурсы сервера между ВМ, позволяя им работать независимо друг от друга. Он обеспечивает изоляцию между ВМ, так что сбои или проблемы в одной ВМ не влияют на другие. Гипервизор также может управлять ресурсами, назначать им приоритеты, масштабировать или перемещать ВМ между физическими серверами.

Существуют два основных типа гипервизоров:

1. Тип 1 (нативный или беспосредственный): работает непосредственно на аппаратном обеспечении сервера и запускает ВМ поверх него. Примеры включают VMware ESXi, Microsoft Hyper-V и Xen.

2. Тип 2 (хостовый): запускается на операционной системе хоста и предоставляет гипервизорные функции на этом уровне. Примеры включают VMware Workstation, Oracle VirtualBox и Microsoft Virtual PC.

Гипервизоры широко используются в виртуализации серверов и облачных вычислений, где они позволяют эффективно использовать физические ресурсы серверов, упрощают управление и развертывание ВМ, а также обеспечивают высокую отказоустойчивость и масштабируемость систем.