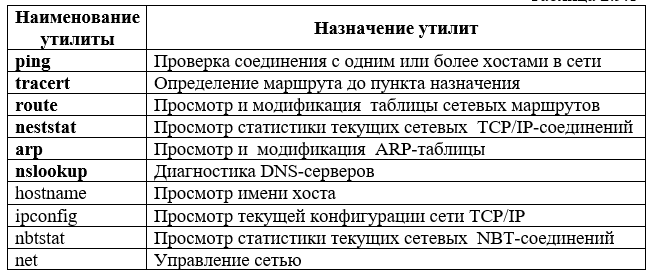
1. Перечислить все ***сетевые утилиты***.



1. Перечислить все ***уровни модели OSI/ISO***. Описать назначение каждого уровня.

Уровни модели OSI/ISO:

**Прикладной** – определение способа взаимодействия пользователей с системой (определение интерфейса). (HTTP, Telnet, FTP, SMTP)

**Представительский** – представление данных в едином формате и виде; кодирование, шифрование, сжатие.

**Сеансовый** – установка, поддержание и разрыв сеанса. Это точка взаимодействия программ и комп. сети (TCP, UPD)

**Транспортный** – контроль передачи данных, разбиение данных на блоки, индексация блоков данных.

**Сетевой** – адресация и маршрутизация. Результат – пакет, которой обрабатывается транспортным уровнем. (IP, ICMP, ARP)

**Канальный** – 2 подуровня: управления доступом к среде передачи данных (определяет методы совместного использования сетевыми адаптерами среды передачи данных) и управления логическим каналом (канал между 2 сетевыми адаптерами, способы обнаружения и исправления ошибок передачи данных). Назначение: подготовить блок данных (кадр) для следующего сетевого уровня.

**Физический** – определяет свойства среды передачи данных (коаксиальный кабель, витая пара) и способы ее соединения с сетевыми адаптерами. (Ethernet)

1. Поясните понятие ***сетевой протокол***.

**Сетевой протокол** — набор правил, позволяющий осуществлять соединение и обмен данными между несколькими устройствами.

1. Указать где в OSI/ISO проходит ***граница между аппаратным и программным обеспечением***.

Канальный уровень

1. Определить понятие ***CSMA/CD***.

Множественный доступ с прослушиванием несущей и обнаружением коллизий

1. Как называется ***программное обеспечение реализующий подуровень LLC канального уровня***.

Драйвер

1. Что определяет ***спецификация NDIS***?

Спецификация интерфейса сетевого драйвера

1. Свойства ***ненадежных протоколов***. Примеры ненадежных и ***надежных*** протоколов

1) отсутствие механизмов обеспечения надежности (пакеты не упорядочиваются, прием не подтверждается);

2) отсутствие гарантий доставки;

3) отсутствие обработки соединений (нет установления, управления и завершения соединения между отправителем и получателем);

4) отсутствие буферизации (оперирует только одним пакетом);

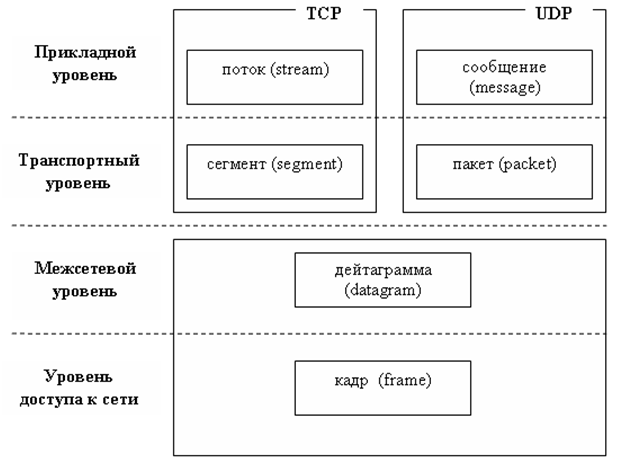
5) отсутствие фрагментации (не разбивает сообщение на несколько пакетов);

6) отсутствие проверки соответствия контрольной суммы.

Примеры ненадежных протоколов: UDP, ICMP.

Примеры надежных протоколов: TCP, SMTP.

1. Перечислить все уровни ***модели TCP/IP***. Описать назначение каждого уровня. Привести примеры протоколов каждого уровня.

1. Уровень доступа к сети (Network Access Layer): - для создания локальных сетей и соединение с глобальными сетями. Примеры: LAN, WAN, Ethernet, SLIP, PPP)
2. Межсетевой уровень (Internet Layer): маршрутизация пакетов между сетями. Примеры протоколов: IP, ICMP, ARP, RARP
3. Транспортный уровень (Transport Layer): передача данных по сети между процессами. Примеры протоколов: TCP, UDP.
4. Прикладной уровень (Application Layer): взаимодействие между приложениями и пользователями. Примеры протоколов: HTTP, FTP, SMTP, DNS, DHCP, POP3
5. Поясните понятия ***хост***, ***адрес хоста***, ***имя хоста***.

Хост - любое [устройство](http://ru.wikipedia.org/wiki/%D0%A3%D1%81%D1%82%D1%80%D0%BE%D0%B9%D1%81%D1%82%D0%B2%D0%BE), имеющее сетевой адаптер

Адрес хоста – ip. Имя хоста – DNS имя.

1. Какая ***организация поддерживает сетевые*** протоколы Internet. Как называются ***документы, описывающие эти протоколы***.

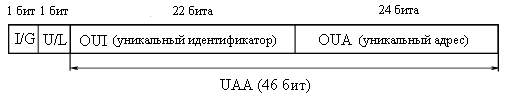
IETF (Internet Engineering Task Force).

RFC (Request for Comments).

1. Что такое ***МАС-адрес***? Структура ***Ethernet МАС-адреса***.

МАС-адрес (Media Access Control address) – это уникальный идентификатор, присваиваемый хосту.

MAC-адрес состоит из 48 бит, представленных в шестнадцатиричном формате.



1 бит – групповой 1, индивидуальный 0

1 бит обычно в 0. Установка бита U/L в 1 означает, что адрес задан не производителем сетевого адаптера, а организацией.

22 бита – уникальный идентификатор (код производителя)

24 бита – уникальный адрес (выбираются изготовителем для каждого устройства)

1. Как ***посмотреть MAC-адрес сетевой карты*** на компьютере?

ipconfig /all и getmac

1. Основное назначение ***межсетевого уровня***.

Доставка дейтаграмм

1. Структура ***IP-адреса***.

IP-адрес – последовательность из 32 бит, разделенное на группы по 8 бит, называемых ***октетами***. 2 части: *номера подсети* (IDподсети) и *номера узла* (ID хоста) в этой подсети. При передаче пакета из одной подсети в другую используется ID подсети. Когда пакет попал в подсеть назначения, ID хоста указывает на конкретный узел в рамках этой подсети.

Существует 2^32 уникальных ip адресов

1. Типы ***IP-адресации***. ***Классы адресов Internet***.

**Тип: динамическая и статическая.**

**Бесклассовая** - вычисление адреса сети выполняется с помощью операции конъюнкции между IP-адресом и маской подсети.

**Классовая** длина маски строго фиксирована.

|  |  |  |
| --- | --- | --- |
| **Класс** | **Диапазон адресов** | **Диапазон частных адресов** |
| А | 0.0.0.0 – 127.255.255.255 | 10.0.0.0 – 10.255.255.255 |
| B | 128.0.0.0 – 191.255.255.255 | 172.16.0.0 – 172.31.255.255 |
| C | 192.0.0.0 – 223.255.255.255 | 192.168.0.0 – 192.168.255.255 |
| D | 224.0.0.0 – 239.255.255.255 | не предусмотрен |
| E | 240.0.0.0 – 247.255.255.255 | не предусмотрен |



1. Поясните понятия ***публичный IP-адрес*** и ***частный IP-адрес***.

**Публичный -** присваивается провайдером при подключении к интернету(он уникален)

**Частный -** присваивается администратором локальной сети компьютерам, объединенным в одну сеть

1. Как посмотреть ***IP-адрес компьютера***.

Ipconfig /all

1. Как протестировать ***IP-соединение в локальной сети*** ?

ping

1. Как получить ***перечень сетевых узлов*** между двумя хостами?

Tracert, pathping

1. Перечислите ***параметры настройки TCP/IP***.

IP,DNS,DHCP, маска, шлюз

1. Поясните понятие ***маска подсети***.

**Маска подсети -**  битовая маска для определения по IP-адресу адреса подсети и адреса узла (хоста, компьютера, устройства) этой подсети.

1. Основные ***отличия между IPv4 и IPv6***.

В IPv6 длина адреса расширена до 128 бит (против 32 в IPv4), что позволяет обеспечить больше уровней иерархии адресации, увеличить число адресуемых узлов, упростить авто-конфигурацию. В ipv6 реализована повышенная система безопасности и не использует классовую адресацию.

1. Поясните понятие ***сетевой порт***. На каком уровне модели TCP/IP это понятие определено.

Идентификационный номер процесса, получающего или отправляющего данные с помощью транспортного уровня. (TCP/UDP)

1. Как ***классифицируются сетевые порты***.

|  |  |
| --- | --- |
| Хорошо известные номера портов (системные службы) | 0 – 1023 |
| Зарегистрированные номера портов (промышленные приложения и процессы) | 1024 – 49151 |
| Динамические номера портов (процессам спец.службой ос) | 49152– 65535 |

1. Как посмотреть какими программами заняты сетевые порты на компьютере?

netstat

1. Поясните понятие архитектура ***клиент/сервер***.

Если все процессы приложения можно условно разбить на две группы: клиент и сервер. Обмен данными осуществляется только между клиентом и сервером, инициатором обмена является клиент.

1. Что такое ***сетевая служба***. Приведите примеры сетевых служб.

Сетевая служба – программная реализация протоколов сетевого уровня TCP/IP. Примеры: DHCP, DNS, NBT, Telnet, FTP, WWW (протокол HTTP) и служба электронной почты (протоколы SMTP, POP3, IMAP4).

1. Поясните понятие ***интерфейс внутренней петли***.

интерфейс, который позволяет клиенту и серверу, находящимся на одном хосте, обмениваться данными посредством протокола TCP/IP, не покидающими пределы этого хоста.

127.0.0.1

1. Назначение сетевых служб ***DSN*** и ***DHCP***.

**DNS(Domain Name System)**: преобразует символические доменные имена в IP-адреса

**DHCP** (Dynamic Host Configuration Protocol): позволяет автоматически получать IP-адрес (маска и шлюз)

1. Организация, ведающая распределением ***IP-адресов***, поддержкой ***сетевых доменов Internet верхнего уровня***, ***регистрацией портов***.

**ICANN** (Internet Corporation for Assigned Names and Numbers).

1. Поясните понятие ***сетевой сокет***.

**Сетевой сокет -** совокупность ip-адреса и номера порта.

1. Назначение ***стандарта POSIX***.

**Стандарт POXIS**(Portable Operating System Interface for Unix) - это набор документов, описывающих интерфейсы между прикладной программой и ОС

1. ***Структура TCP/IP TCP-сервера***. Все функции и все параметры функций.

* для инициализации библиотеки WS2\_32.DLL.
* создание сокета с помощью функции **socket()**; связывание сокета с IP-адресом и портом с помощью функции **bind()**;
* - Функция listen переводит сокет, ориентированный на поток, в состояния прослушивания. - Функция accept переводит процесс сервера в состояние ожидания, до момента пока программа клиента не выполнит функцию connect (создание сокета клиента cS и параметры сокета clnt)
* Четвертый и пятый блоки программы сервера предназначены для обмена данными по созданному соединению. Следует обратить внимание, что, во-первых, используются функции send и recv, а во-вторых, в качестве параметра эти функции используют сокет, созданный командой accept.
* Closesocket,wsacleanup.

1. ***Структура TCP/IP TCP-клиента***. Все функции и все параметры функций.
2. инициализации библиотеки WS2\_32.DLL.
3. **socket()**: функция для создания нового сокета клиента.
   * Параметры функции:
     + **domain** - домен сокета, который должен использоваться (обычно **AF\_INET** для IPv4 или **AF\_INET6** для IPv6).
     + **type** - тип сокета, который должен использоваться (обычно **SOCK\_STREAM** для TCP или **SOCK\_DGRAM** для UDP).
     + **protocol** - протокол, который должен использоваться (обычно **0** для автоматического выбора соответствующего протокола).
4. **connect()**: функция для установления соединения с удаленным сервером.
   * Параметры функции:
     + **sockfd** - идентификатор сокета клиента, который должен быть использован для установления соединения.
     + **addr** - структура, содержащая IP-адрес и порт сервера, с которым нужно установить соединение.
5. **send()**: функция для отправки данных на удаленный сервер.
   * Параметры функции:
     + **sockfd** - идентификатор сокета клиента, который должен быть использован для отправки данных.
     + **buf** - буфер с данными, которые должны быть отправлены.
     + **len** - размер буфера данных.
6. **recv()**: функция для приема данных от удаленного сервера.
   * Параметры функции:
     + **sockfd** - идентификатор сокета клиента, который должен быть использован для приема данных.
     + **buf** - буфер, в который будут записаны принятые данные.
     + **len** - максимальный размер данных, которые могут быть приняты.
7. **close()**: функция для завершения соединения с удаленным сервером и освобождения ресурсов, связанных с сокетом.
   * Параметры функции:
     + **sockfd** - идентификатор сокета клиента, который должен быть закрыт.

