

Операционные системы

лекции 20, 21

- **Файловая система**
- **Основы сетевого взаимодействия**

10.11.2010



Файловая система

Задачи:

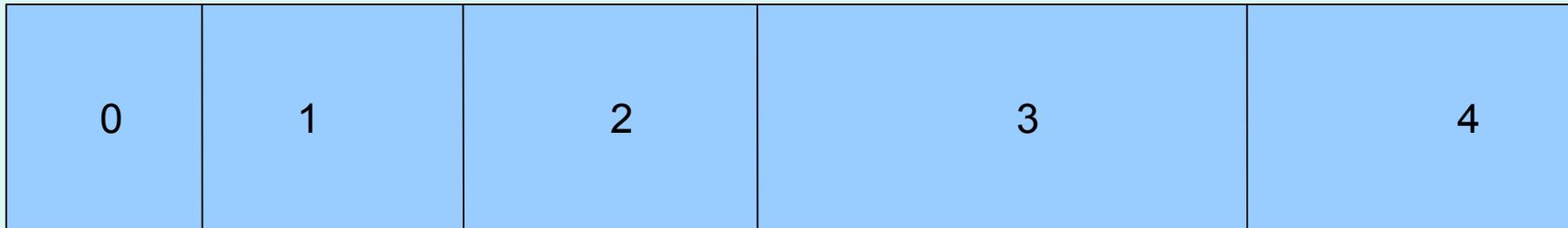
1. Хранение информации
2. Организация доступа к информации
3. Защита

Файловая система ОС UNIX иерархическая. Именованние файлов — многоуровневое.



Системная организация файловой системы

ВЗУ разбивается на блоки



N

- 0 блок начальной загрузки
- 1 Суперблок
- 2 Индексные дескрипторы
- 3 Блоки данных
- 4 Область выгрузки





Индексный дескриптор

Основная структура, описывающая атрибуты файла

- тип файла, права доступа
- количество ссылок из каталогов к ИД
- привилегии доступа к файлу (пользователь, группа пользователя)
- длина файла
- время посл. обращения к файлу, создания, модификации
- информация об адресации блоков на диске

Дополнительные свойства:

04000 s-бит владельца: при выполнении процесс получит права владельца файла (/etc/passwd)

02000 s-бит группы владельца

01000 T-бит после окончания работы программа остается в области swapping





Типы файлов и их свойства

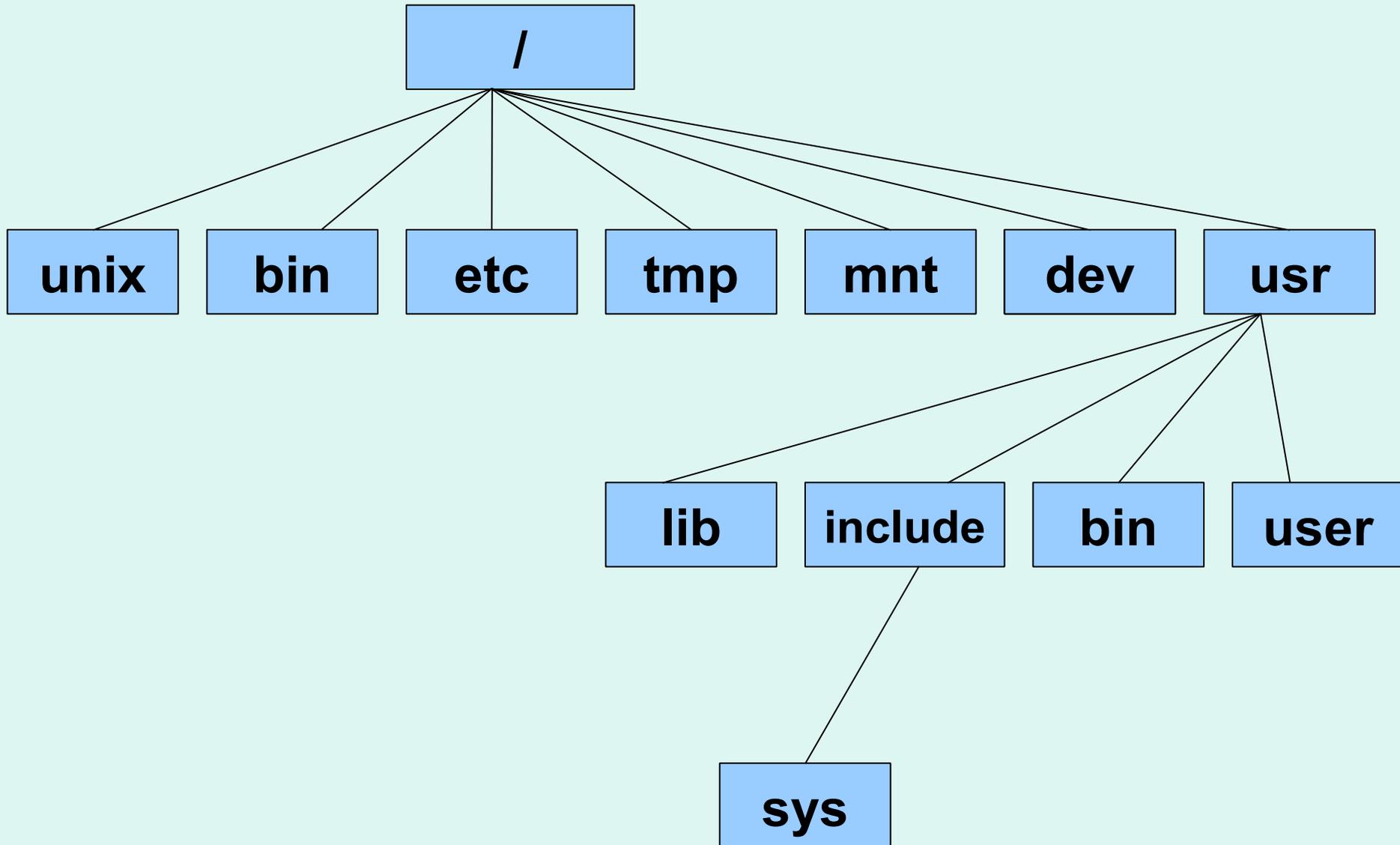
1) Регулярные файлы

2) Каталоги

3) Файлы устройств
/dev



Логическая структура файловой системы





Основы сетевого взаимодействия

Протокол

Протокол — набор правил, позволяющий осуществлять соединение и обмен данными между несколькими сторонами

Стороны: физические устройства, логические устройства, процессы

Протокол состоит из:

- Описания формата передачи данных
- Описания возможных состояний работы сторон
- Описания правил перехода из состояния в состояние





Пример взаимодействия

Покупка товара по каталогу в европейском магазине с доставкой почтой.

Стороны: магазин и клиент

1. Заполнение формы заказа

Форма определяет формат обмена данными, понятный магазину и клиенту

Прикладной уровень (application layer)

уровень взаимодействия прикладных программ





Пример взаимодействия

2. Магазин функционирует на международном уровне, необходима договоренность о языке, на котором заполняется форма.

Например, язык — французский

Перевод формы на фр. язык прикладываем к заказу.

Представительский уровень (presentation layer)

Уровень взаимодействия, на котором определяется представление данных (язык, кодировка)





Пример взаимодействия

3. Магазин может предоставлять скидки покупателям с историей, дарить подарки, Начислять бонусы... или еще как-либо поддерживать историю взаимодействия

К переводу формы заказа добавляем номер бонусной карты покупателя

Сеансовый уровень (session layer) - уровень, который обеспечивает длительное взаимодействие





Пример взаимодействия

4. Чтобы почта могла доставить заказ, необходимо указать адрес доставки письма

Форму, перевод, приклеенный номер бонусной карты вкладываем в конверт с адресом, индексом и отдаем конверт на почту (транспортный агент)

Транспортный уровень (transport layer) — это уровень, на котором выполняется доставка сообщений (писем).





Пример взаимодействия

5.Почта, как правило, не доставляет письмо от отправителя до получателя непосредственно: письма могут проходить через несколько промежуточных пунктов, на каждом из которых пересортировываться. Письма, отправляемые на один сортировочный пункт, упаковываются в мешки, на которых пишется адрес сортировочного пункта

Сетевой уровень (network layer) - уровень, на котором информация пересылается между узловыми пунктами.





Пример взаимодействия

6. Для доставки мешков писем между узловыми пунктами, например, на самолете, необходимо упаковать мешки в спец. контейнер, написать на контейнере сопроводительную документацию.

Канальный уровень (data-link layer) — это уровень взаимодействия, на котором информация готовится к пересылке физическим носителем.





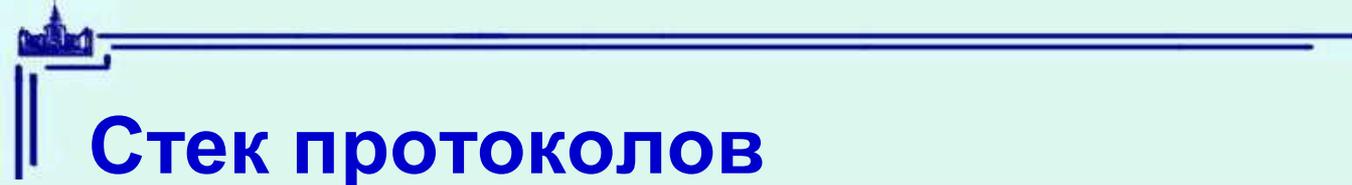
Пример взаимодействия

7. Мешки грузятся в самолет, самолет летит по определенному маршруту, на определенной высоте в соответствии с планом полета

Физический уровень (physical layer) — это уровень взаимодействия, на котором происходит реальная (физическая) доставка данных

Поступление в точку назначения: заказ проходит все ступени обработки снизу вверх. На каждом уровне удаляется контейнер, конверт или служебная информация соответствующего уровня и данные передаются на уровень вверх.



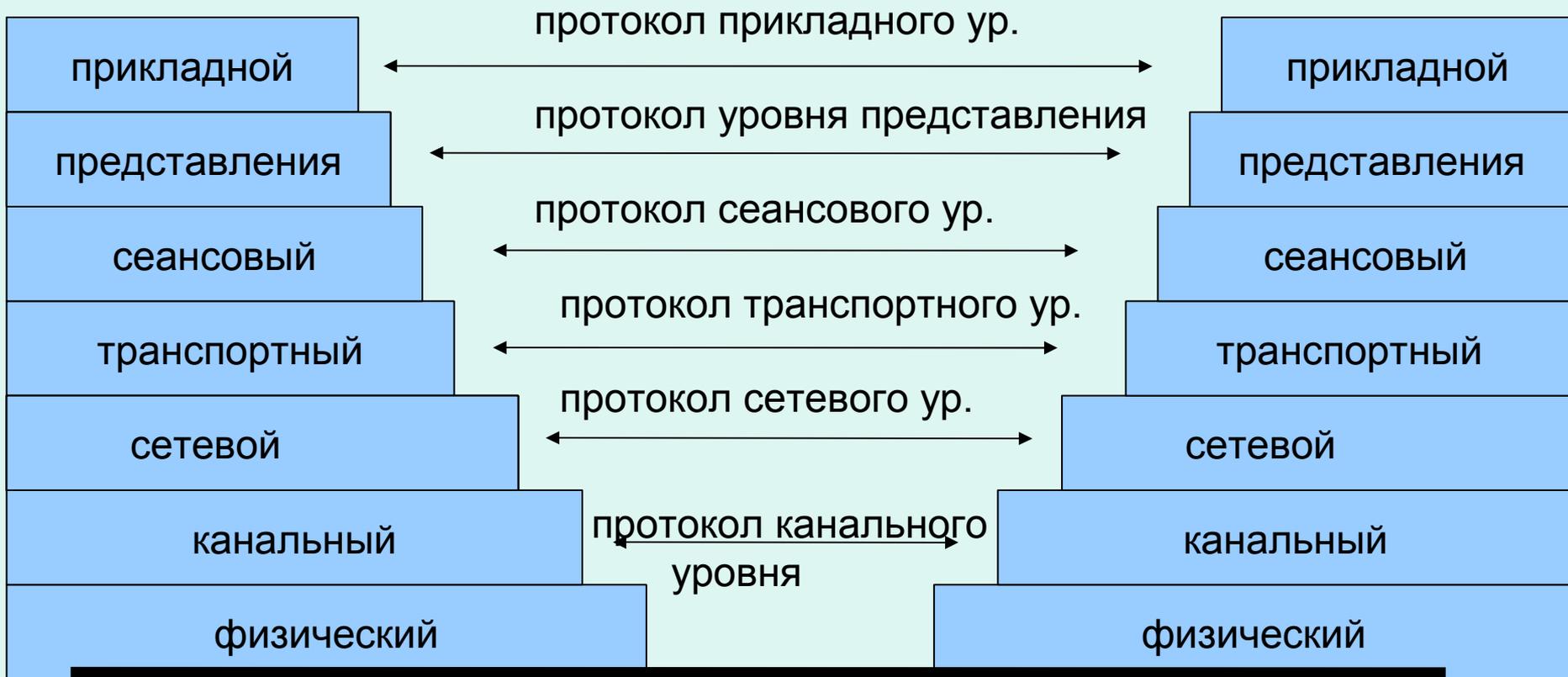


Стек протоколов

Стек протоколов — это такое разделение взаимодействия на уровни, при котором

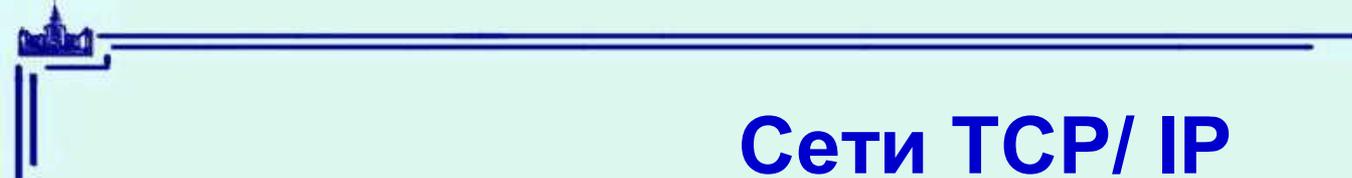
- для протокола верхнего уровня не важно, каким образом данные будут доставлены до получателя и какая вспомогательная информация при этом еще потребуется
 - Для протокола нижнего уровня не важно, какие именно данные поступили от верхних уровней, доставляются любые данные
 - Протокол каждого уровня описывает взаимодействие сторон на этом же уровне
- 

Эталонная модель ISO/OSI



Физическая среда передачи данных





Сети TCP/ IP

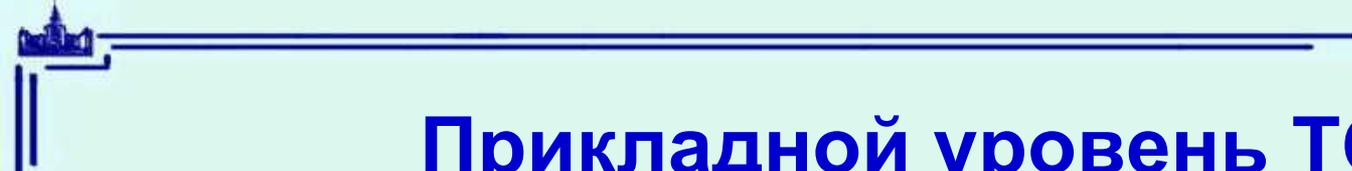
Протокол TCP/IP - Transmission Control Protocol/Internet Protocol (Протокол управления передачей/Протокол Internet)

Стек протоколов TCP/IP включает в себя протоколы 4 уровней:

- прикладного (application)
- транспортного (transport)
- сетевого (internet)
- уровня доступа к среде (network access)

Работают поверх различных протоколов канального и физического уровня, обеспечивающих пакетную пересылку данных. Протоколы канального и физического уровня не специфицируются. На практике используются: Ethernet (протоколы IEEE 802.3), WiFi (IEEE 802.11), GPRS/EDGE, и т. д.





Прикладной уровень ТСР/ІР

НТТР

Основой НТТР является технология «клиент-сервер»: клиент — инициирует соединение и посылают запрос сервер (поставщик информации) — ожидает соединения, обрабатывает запрос, возвращают обратно сообщение с результатом.

Получение информации с Web-сайтов.

FTP

Передача файлов.

Протокол позволяет подключаться к серверам FTP - серверам, просматривать содержимое каталогов и загружать файлы с сервера или на сервер.





Транспортный уровень ТСП/IP

TCP (Transmission Control Protocol) — протокол надежной передачи данных (пакетов), с подтверждением. Передается поток данных, с предварительной установкой соединения.

Осуществляет повторный запрос данных в случае потери данных.

Устраняет дублирование при получении двух копий одного пакета

Гарантирует, что приложение получит данные точно в такой же последовательности, в какой они были отправлены, и без потерь.

UDP (User Datagram Protocol) — пользовательский протокол данных) — это транспортный протокол для передачи данных без установления соединения.

UDP не гарантирует доставку дейтаграммы. В случаях влияния внешних факторов, приводящих к сбоям, протокол UDP не предусматривает стандартного механизма повторения передачи потерянных пакетов.

Быстро и эффективно доставляет данные для приложений, которым требуется большая пропускная способность линий связи, либо требуется малое время доставки данных.





Сетевой уровень TCP/IP

протокол IP

IPv4.

В протоколе IP этой версии каждому узлу сети ставится в соответствие IP-адрес длиной 4 байта.

При этом компьютеры в подсетях объединяются общими начальными битами адреса.

IPv6 — вводится в эксплуатацию. Под адрес отводится 128 бит.

В середине 2010 года в Интернете присутствовало более 3000 сетей, работающих по протоколу IPv6.
более 320 000 сетей в адресном пространстве IPv4





Адресация компьютеров

- У протокола IP v4 размер адреса — 32 бита, адрес делится на 4 байта, каждый байт записывается в десятичном виде, байты разделяются точкой:

212.192.248.182

Специальные адреса IPv4

127.0.0.1 — localhost — «встроенная» сеть, которая состоит из одного данного компьютера, присутствует на каждом компьютере, даже когда он не подключен к сети

192.168.X.Y — адреса, выделенные для локального использования.

Выход в Интернет возможен через прокси-сервер

10.X.Y.Z — адреса, выделенные для локального использования

Адрес: адрес сети . номер компьютера в сети

- Адрес сети: номер компьютера равен 0, то есть 192.168.10.0
- Широковещательный адрес: в номере компьютера все биты установлены в 1, например, 192.168.10.255





Domain Name System (DNS)

- Сервис отображения символических имен в IP-адреса
- Для отображения имени в IP-адрес процесс запрашивает DNS-сервер, который в ответ возвращает IP-адрес
- DNS-сервер может располагаться на другом компьютере

Пример: www.msu.kz

kz — домен верхнего уровня

msu — домен второго уровня

www — имя хоста в домене

- Для получения адресов серверов домена kz необходимо обратиться к корневым серверам
 - Для получения адресов серверов msu.kz необходимо обратиться к серверам зоны kz
 - Для получения адреса www.msu.kz необходимо обратиться к серверу зоны msu.kz
- 



Сетевой уровень TCP/IP

Протокол IP

пакеты доставляются от хоста-отправителя до хоста-получателя

- Доставка пакета не гарантируется: если по каким-либо причинам очередной маршрутизатор не может переслать пакет дальше, пакет уничтожается
- Корректность доставки не гарантируется — содержимое может оказаться испорченным

Параметры IP

- TTL (time to live) — количество «прыжков» от хоста к хосту, после которого пакет уничтожается
- MTU (maximal transfer unit) — максимальный размер фрейма (то есть пакета на канальном/физическом уровне), который допустим в данной среде передачи данных.

Например, MTU Ethernet — 1500 байтов

Если размер пакета превышает MTU, пакет может быть разбит на пакеты, либо уничтожен





Транспортный уровень

Доставка данных: от процесса-отправителя до процесса-получателя

- На одном хосте могут работать разные процессы, предоставляющие сетевые сервисы
- Один процесс может предоставлять несколько сервисов
- Дополнительная идентификация соединения — номер порта

Порт — число от 1 до 65535 (16 бит)

Порты с номерами до 1024 зарезервированы для использования процессами с правами администратора

- Некоторые номера портов приписаны (обычно используются) стандартными сервисами
- Каждому исходящему запросу или датаграмме присваивается номер порта

Идентификация соединения:

IP-адрес, порт отправителя, IP-адрес, порт получателя.





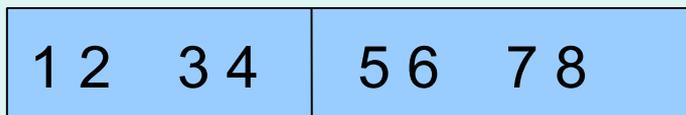
Кодирование данных

На разных хостах сети могут использоваться разные способы хранения целых и вещественных чисел, разные требования к выравниванию полей структур, разные кодировки символов...

- Преобразование данных в формат передачи данных по сети — сериализация (serialization)
 - Обратное преобразование — десериализация
- 

Представление целых чисел

0x12345678 Big-endian



Little-endian



0x1234



Sparc, Motorola 68000



x86

В качестве сетевого формата представления целых чисел принят Big-endian формат.





Перекодировка целых чисел

Перекодирование целых

Network byte order — порядок байт в сети

Host byte order — локальный порядок байт

```
#include <arpa/inet.h>
```

```
uint32_t htonl (uint32_t hostlong) ;
```

```
uint16_t htons (uint16_t hostshort) ;
```

```
uint32_t ntohl (uint32_t netlong) ;
```

```
uint16_t ntohs (uint16_t netshort) ;
```

