

# Курс: Сети и телекоммуникации (СиТ)

Темы: 1. Адресация в сетях  
2. Технологии ETHERNET  
3. Сети сотовой связи  
4. Internet of Things  
5. История СиТ

Широков Владимир Леонидович  
каф. ВМСС, WhatsApp: 89852846055  
[ShirokofVL@mail.ru](mailto:ShirokofVL@mail.ru)

# **Исторические вехи СиТ**

**DARPA – 1969 (29.10.69 –  
неофициальный ДР Internet);**

**Сеть ALOHA – 1970; Ethernet – 1973;**

**Модель OSI – 1981-1982;**

**Стек TCP/IP – 1982 (ARPANET-01.01.83 –  
официальный ДР Internet);**

**Сотовая связь 1G –1982, 2G –1992;**

**Wi-Fi – 1997; WMAN (WiMAX) – 2001**

# Классы сетей

- **Персональные PAN** (Personal Area Networks) – вокруг человека ~10м
- **Локальные LAN** (Local Area Networks) – дом, квартира, коттедж и участок ~100м
- **Городские MAN** (Metropolitan Area Networks) – муниципалитет (район), корпоративная сеть (фирма, завод)
- **Глобальные WAN** (Wide Area Networks) – города, регионы, страны

# Модель OSI (МВОС) и стек TCP/IP



## Отличия TCP/IP от OSI (МВОС):

- Канальный и физический реализуются машиной СПД
- Сетевой уровень называется межсетевым
- Уровни сессий и представления реализуются приложениями

# Уровни адресации

- 1. MAC-адрес** – 2-й уровень (канальный OSI и машина СПД стека TCP/IP) – DLL (LLC, MAC) – Ethernet, PPP
- 2. IP-адрес** (IPv4 и IPv6) – 3-й уровень (сетевой OSI, межсетевой стека TCP/IP) – пакеты
- 3. Адресация уровня 4+** (транспортный и приложений модели OSI и стека TCP/IP) – порты, номера протоколов и сегменты TCP, UDP, сокеты и DNS

# Назначение адресации

- Преодоление сложности
- Идентификация узлов, каналов, интерфейсов, транспортных протоколов (номера, порты)
- Функционирование уровней и сети в целом
- Передача данных через уровни сетевой иерархии (согласно моделей OSI и TCP/IP)
- Обеспечение однозначного соответствия между адресами разных уровней (MAC, IP, порты, сокеты, DNS)

# 1. MAC-адресация (2-й уровень OSI)

- Адресация канального уровня (идентификаторы интерфейсов и активного оборудования) – 3 типа:
  - MAC-48 – сетевое оборудование 48 бит (6 байт)
  - EUI-48 (Extended Unique Identifier, расширенный ID)
  - EUI-64 (шина FireWire Apple) – 64 бита (8 байт) IPv6
- MAC (Media Access Control) – глобально уникальная адресация канального уровня для управления доступом к среде

# Использование MAC-48

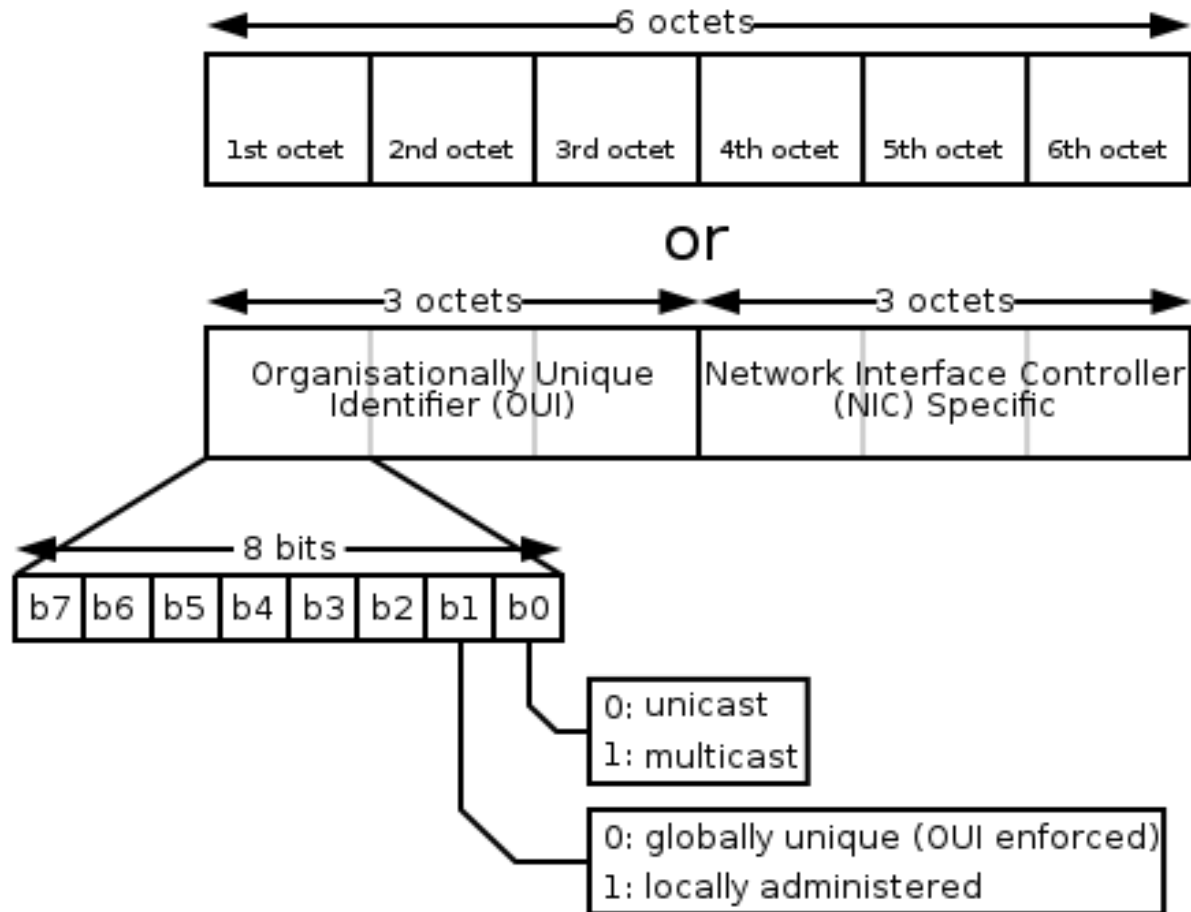
Используется в сетевом оборудовании:

- Локальные сети (Ethernet, ARCNET, Token Ring)
- FDDI сети (Fiber Distributed Data Interface) – распределенный оптический интерфейс данных
- WiMAX (Worldwide interoperability for Microwave Access) – интероперабельная всемирная микроволновая сеть доступа, наряду с LTE (4G – Fourth Generation, 4-е поколение)



# Общая структура MAC

- По стандарту Ethernet IEEE 802.3 MAC – это 6-байт (октетов), в которых выделяются следующие части:



# Структура MAC-адреса

- MAC – это 6 пар шестнадцатеричных цифр, записанных через дефис или двоеточие;

**Пример:** 7C-D6-F5-E5-7A-B9 или 7C:D6:F5:E5:7A:B9, где A=10 ( $1010_2$ ), B=11 ( $1011_2$ ), C=12 ( $1100_2$ ), D=13 ( $1101_2$ ), E=14 ( $1110_2$ ), F=15 ( $1111_2$ )

- MAC идентифицирует отправителей и получателей кадров Ethernet канального (аппаратного) уровня;
- MAC кодирует до  $2^{48}$  (281 474 976 710 656) адресов (должно хватить до 2100 года)

# Элементы MAC-адреса

MAC-адрес имеет 4 части:

- 2 основные (каждая по 3 байта, или октета)
  - старшие 3 байта – OUI (Organizationally Unique Identifier) – уникальный код производителя
  - младшие 3 байта – NIC (Network Interface Controller) – код контроллера сетевого интерфейса (исключение – SNA IBM)
- 2 специальные (2 младших бита в первом октете OUI)
  - 0-й бит указывает для какого адресата предназначен кадр: одиночного (0) или группового (1)
  - 1-й бит указывает, является адрес администрируемым глобально (0) или локально (1).

# Определение MAC-адреса

- Windows – getmac /v или ipconfig /all
- HP-UX – /usr/sbin/lanscan
- Linux – ip link show
- Mac OS X – ifconfig
- Cisco IOS, JUNOS – show interfaces
- OpenBSD – ifconfig | grep lladdr
- FreeBSD – grep ether
- QNX6 – ifconfig или nicinfo
- QNX4 – netinfo -l

## Пример:

```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows [Version 10.0.22000.1455]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Admin>getmac /v

Connection Name Network Adapter Physical Address Transport Name
=====
Ethernet 2 Cisco AnyConnec N/A Hardware not present
Ethernet Realtek PCIe Gb B4-B6-86-D0-13-D5 Media disconnected
Беспроводная се Intel(R) Dual B 7C-76-35-45-7A-09 \Device\Tcpip_{C99F5AB9-0132-49A8-B19C-D31A81448BCC}

C:\Users\Admin>
```

# Смена MAC-адреса

Возможна следующими способами:

1) через веб-интерфейс

Маршрутизаторы могут дублировать MAC-адрес сетевой карты подключённого к ним компьютера

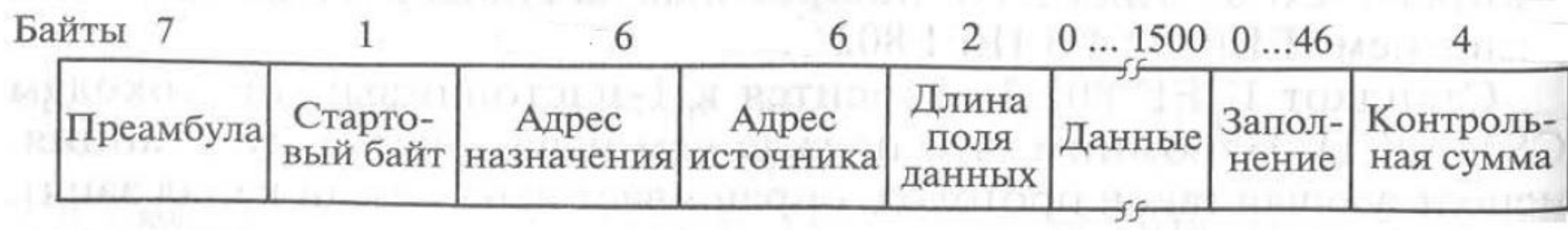
2) через программу-драйвер

Э оборудование, в котором это нельзя сделать, например, в STB (Set-Top-Box) IP-TV

3) программатором

# MAC уровня Ethernet

## Структура кадров (стандарт IEEE 802.3):



Байты 1-7 преамбулы 10101010.  $\Sigma$  задержка = 5,6 мкс. Стартовый 10101011 – начало передачи. Адреса – MAC. Мин. кадр – 64 байта.

*Заполнение (расширение носителя)* – компенсация длины – до 46 байт.

Контрольная  $\Sigma$  – циклический избыточный код CRC (Cycle Redundancy Code).

$t_{\min}$  обнаружения коллизии  $\rightarrow$  64 байт на 2,5 км (4 репитера) – 51,2 мкс.

При больших скоростях мин. длина кадра FE – 640, GbE – 6400 байт.

## **2. Логическая адресация (3-й уровень)**

- На 3-м уровне (сетевой OSI, межсетевой TCP/IP) используются адреса IPv4 или IPv6
- IPv4 состоит из IP-адреса и маски
- IP-адрес занимает 32 бита, т.е. 4 байта (октета)
- IPv4 записывается как 4 числа 0÷255 в 10сс, разделённые точками, например, 192.168.0.0
- Маска – сплошная последовательность «1», записывается 4 числами, разделёнными точками, отделяющими № сети от № хоста

# Перевод чисел из 10сс в 2сс

7	6	5	4	3	2	1	0
$2^7=128$	$2^6=64$	$2^5=32$	$2^4=16$	$2^3=8$	$2^2=4$	$2^1=2$	$2^0=1$

## Алгоритм:

- 1) найти макс. число степени  $2 \leq$  исходному
- 2) найти остаток и макс. число степени  $2 \leq$  остатку
- 3) повторить п.2 до исчерпания остатка

Пример.  $202 = 128 + 64 + 8 + 2$

7	6	5	4	3	2	1	0
1	1	0	0	1	0	1	0



# Перевод больших чисел в 2сс

7	6	5	4	3	2	1	0
$2^7=128$	$2^6=64$	$2^5=32$	$2^4=16$	$2^3=8$	$2^2=4$	$2^1=2$	$2^0=1$

## Алгоритм:

- 1) вычесть число (247) из максимального (255)
- 2) найти макс. число степени  $2 \leq$  остатку ( $8=2^3$ )

Пример.  $247=128+64+32+16+4+2+1$ ,  $255-247=8=2^3$

7	6	5	4	3	2	1	0
1	1	1	1	0	1	1	1

# Структура IPv4

- IP: № сети и № хоста
- Маска: отделяет № сети от № хоста (4 числа в 10сс, разделённые точками)



# Классы и маски IPv4

- А (1.0.0.0 – 127.255.255.255, маска 255.0.0.0):
  - 126 сетей (7 бит,  $2^7-1$ ), 16 777 215 хостов (24 бита,  $2^{24}-1$ ).
- В (128.0.0.0 – 191.255.255.255, маска 255.255.0.0):
  - 16 382 сетей (14 бит,  $2^{14}-1$ ), 65 535 хостов (16 бит,  $2^{16}-1$ ).
- С (192.0.0.0 – 223.255.255.255, маска 255.255.255.0):
  - 2 млн сетей (19 бит,  $2^{19}-1$ ), 255 хостов (8 бит,  $2^8-1$ ).
- D (224.0.0.0 – 239.255.255.255) – групповая (Multicast) передача, маска 255.255.255.255
- E (240.0.0.0 – 247.255.255.255) – резерв

# Специальные IP\*

- Все нули – хост, где находится ПО, используемое при загрузке обратившегося хоста
- № сети – все нули и № хоста – хоста данной сети
- Все единицы в IP – вещание на все хосты данной сети
- № сети и все единицы – вещание на удалённую сеть
- Сеть 127 – интерфейс хоста разворачивает пакет с этим адресом на вход себе (тестирование сетевого ПО)

00	Данный хост
00 ... 00 Хост	Хост данной сети
11	Вещание на локальную сеть
Сеть 1111 ... 1111	Вещание на удаленную сеть
127 (Любые значения)	Кольцевая проверка линии

# Протокол DHCP

Dynamic Host Configuration Protocol – динамической настройки хостов:

- прикладной протокол, работающий по модели «клиент-сервер»
- Автоматически назначает хостам IP-адреса и другие параметры, необходимые для работы в сети
- необходим для автоматического подключения хостов к сети

# Протокол ARP

- Address Resolution Protocol – разрешения адресов, рассылает запросы, собирает ответы, определяя MAC хоста по его IP, создаёт ARP-таблицу соответствия IP<->MAC

223.1.2.1	08:00:39:00:2F:C3
223.1.2.3	08:00:5A:21:A7:22
223.1.2.4	08:00:10:99:AC:54

# Протокол RARP

- Reverse ARP – обратный ARP, определяющий IP по MAC хоста, т.е. получает логические адреса отправителей

### **3. Адресация уровня 4+**

❖ Порт (port, 16 бит, 1÷65535, через «:» после IP/DNS, модель клиент-сервер) – связь с IP:

- передача файлов (FTP) – tcp/21
- сопоставление DNS и IP – udp/53
- гипертекст (HTTP) – tcp/80

❖ Протокол (8 бит, 1÷255, TCP, UDP, SCTP, DCCP):

- #tcp=6 (Transmission Control Protocol) – управления передачей
- #udp=17 (User Datagram Protocol) – датаграмм пользователей
- #sctp=6&17 (Stream Control Transmission Protocol) – управления потоками, #dccp=33 (Datagram Congestion Control Protocol) – датаграммный с отслеживанием перегрузок

## Адресация уровня 4+ (2)

- ❖ Датаграмма (дейтаграмма) – блок данных без соединения и без орг. вирт. канала (Ethernet, IP, UDP)
- ❖ В пакете IP указывается порт источника (обратный) и порт назначения (обратный порт в UDP не нужен, => 0)
- ❖ Сóкет (socket – конечная точка соединения) – программный интерфейс обмена между процессами (клиентские и серверные) – пара IP адрес и порт
- ❖ DNS (Domain Name System) – система доменных имён (символьное представление IP, уровень приложений)



# Классы портов

- *Известные* порты 1÷1023 (описаны RFC, одобрены IESG):
  - RFC (Request for Comments) – технические требования
  - IESG (Internet Engineering Steering Group) – группа инженерного регламента Internet
  - IANA (Internet Assigned Numbers Authority) – администрация адресов пространства Internet, RIPE – IANA в Европе
- *Другие* порты:
  - после регистрации в IANA (RIPE) – 1024÷49151
  - без регистрации – 49152÷65535
- Приложения используют порты как клиенты, начинают исходящие соединения, а серверы обслуживают порты

# Использование портов

- Обмен данными ведётся по модели клиент-сервер и по протоколу
- Для установки соединения используются:
  - ✓ номер протокола
  - ✓ два IP-адреса для построения маршрута между хостами отправителя и получателя
  - ✓ два порта: порт протокола получателя и порт отправителя/источника (обратный)
- Клиенты инициируют исходящие соединения или слушают порты
- Серверы «слушают» и коммутируют порты клиентов

## Использование портов (2)

Известные порты:

- HTTP (гипертекст) – tcp/80
- FTP (передача файлов) – tcp/21
- Сопоставление DNS имён и IP-адресов – udp/53
- SMTP-сервер e-mail «слушает» tcp/25
- POP-клиент e-mail «слушает» tcp/110

Узнать номера активных портов, выданных транспортным протоколам, можно в Windows и Unix-подобных ОС при помощи утилит netstat и nmap (UNIX и Linux)

# Состояние портов

При создании socket процесс обмена запрашивает у ОС порт с определённым номером, которая предоставит порт

- либо откажет, если он отдан другому процессу
- любой свободный порт с другим номером

Состояние порта <sup>[1]</sup>	Описание состояния
Открыт или прослушивается ( <i>open</i> или <i>listen</i> )	Сервер «слушает» порт, т.е. готов принимать подключения клиентов
Фильтруется ( <i>filtered</i> )	Не определяется, открыт или закрыт, например, из-за работы файервола
Закрыт ( <i>closed</i> )	ОС ещё не выдала порт процессу, но в любой момент может это сделать

# Процесс обмена

Поскольку обмен между процессами ведётся по протоколам, для соединения необходимы:

- номер используемого протокола
- два IP-адреса отправителя и получателя для маршрутизации
- два номера портов отправителя и получателя для соединения

*Примечание.* Номер протокола указывается в поле «протокол» заголовка IP-пакета (1 байт, например, #icmp=1, #igmp=2)

# Использование портов в URL и сокетов

- Номера портов указываются в адресах веб-сайтов (URL) через «:»
- По умолчанию в протоколе HTTP используется порт 80
- По умолчанию в протоколе HTTPS используется порт 443

*Пример.* URL вида `http://www.example.com:8080/path/` указывает, что веб-ресурс обслуживается веб-сервером на порту 8080

## 5. Бесклассовая междоменная маршрутизация

**CIDR** (Classless Inter-Domain Routing) – IP/m, где

m = количество единиц в маске

**Табл.** Полезный ряд 2-х и 10-х чисел:

Двоичные	Десятичные	Двоичные	Десятичные
11111111	255	01000000	$64 = 2^6$
11111110	254	00110000	$48 = 2^5 + 2^4$
11111100	252	00100000	$32 = 2^5$
11111000	248	00011000	$24 = 2^4$
11110000	240	00010000	$16 = 2^4$
11100000	224	00001000	$8 = 2^3$
11000000	192	00000100	$4 = 2^2$
10101000	$168 = 2^7 + 2^5 + 2^3$	00000011	$3 = 2 + 1$
10000000	$128 = 2^7$	00000010	$2 = 2^1$
01100000	$96 = 2^6 + 2^5$	00000001	$1 = 2^0$

# Адреса CIDR (1-2)

1) Найти адрес сети 192.168.19.254/25

$$m = 11111111.11111111.11111111.10000000$$

$$(255.255.255.128) \ \& \ \underline{11000000.10101000.00010011.11111110} = \\ 11000000.10101000.00010011.10000000 \Rightarrow$$

Адрес сети = 192.168.19.128

2) Найти адрес сети 192.168.19.254/23

$$m = 11111111.11111111.11111110.00000000$$

$$(255.255.254.0) \ \& \ \underline{11000000.10101000.00010011.11111110} = \\ 11000000.10101000.00010010.00000000 \Rightarrow$$

Адрес сети = 192.168.18.0



## Адреса CIDR (3-4)

3) Может ли существовать узел с указанным IP-адресом?

192.168.10.144/28, т.е. маска

11111111.11111111.11111111.11110000 => 255.255.255.240

192.168.10.XX

144 = 128 + 16 => 10010000, т.е. 192.168.10.144 – адрес сети!

4) Может ли существовать узел с указанным IP-адресом?

192.168.10.143/28

143 = 128 + 15 => 10001111, т.е.

192.168.10.143/28 - широковещательный адрес!

## Адреса CIDR (5-7)

5) Сколько узлов м.б. в сети с маской /28 ?

$$32 - 28 = 4, 2^4 - 2 = 14 \text{ (от 0001 до 1110)}$$

Не забывать вычесть 2, т.к. все 0 – это адрес сети, а все 1, т.е. 1111 – это широковещательный адрес!

6) Сколько узлов м.б. в сети с маской /23 ?

$$32 - 23 = 9 \text{ бит, } 2^9 - 2 = 510$$

7) Обратная задача: найти маску для адресации 127 узлов

$$2^7 - 2 = 126, \text{ т.е. 7 бит не хватит, нужно 8 бит, т.е.}$$

маска m /24 и значит = 255.255.255.0

# Подсети CIDR

- 1) упорядочить подсети по убыванию числа узлов
- 2) определить максимальное число бит и общую маску сети
- 3) определить для каждой подсети число бит для адресации узлов
- 4) занести в таблицу и выделить цветом биты адресации узлов
- 5) сформировать последние биты масок подсетей

Пример. IP/m = 192.168.64.0/23, N1=100, N2=10, N3=25, N4=10,  
N5=50, N6=200

Решение:

- 1) N6, N1, N5, N3, N2, N4=10
- 2) максимальное число  $32 - 23 = 9$  бит для адресации узлов подсетей, m = 255.255.254.0
- 3) число бит адресации узлов в каждой подсети:  
8, 7, 6, 5, 4 и 4 бита соответственно

# Подсети CIDR (2)

ni	8	7	6	5	4	3	2	1	0
200	0								
100	1	0							
50	1	1	0						
25	1	1	1	0					
10	1	1	1	1	0				
10	1	1	1	1	1				
Подсеть	Кол-во узлов, ni	IP/m подсети Ni (i=1,6)		IP первого узла подсети Ni		IP последнего узла подсети Ni		Широковещательный IP	
N6	200	192.168.64.0/24		192.168.64.1		192.168.64.200		192.168.64.255	
N1	100	192.168.65.0/25		192.168.65.1		192.168.65.100		192.168.65.127	
N5	50	192.168.65.128/26		192.168.65.129		192.168.65.178		192.168.65.191	
N3	25	192.168.65.192/27		192.168.65.193		192.168.65.217		192.168.65.223	
N2	10	192.168.65.224/28		192.168.65.225		192.168.65.234		192.168.65.239	
N4	10	192.168.65.240/28		192.168.65.241		192.168.65.250		192.168.65.255	