

Модель OSI

Лекция 1

Artem Beresnev



t.me/ITSMDao

t.me/ITSMDaoChat

Содержание:

- ▶ Проблемы сетевой коммуникации
- ▶ Модель ISO - OSI. Зачем нужно в ней разбираться?
- ▶ Передача данных по сетевому стеку. Инкапсуляция-деинкапсуляция.
- ▶ Назначение уровней стека.
- ▶ Зачем нужна многоуровневая адресация?

лит-ра:

- Комп-е сен , 5 лнд.
(Олифера)
- Комп-е сен
(Бланшильда)

сайт:

book.itep.ru

Проблемы сетевой коммуникации

- ▶ Необходимость обеспечить взаимодействие разнообразных систем
- ▶ Необходимость работать через разное оборудование во время одного сеанса связи
- ▶ Необходимость организационно разграничивать управление в крупных сетях

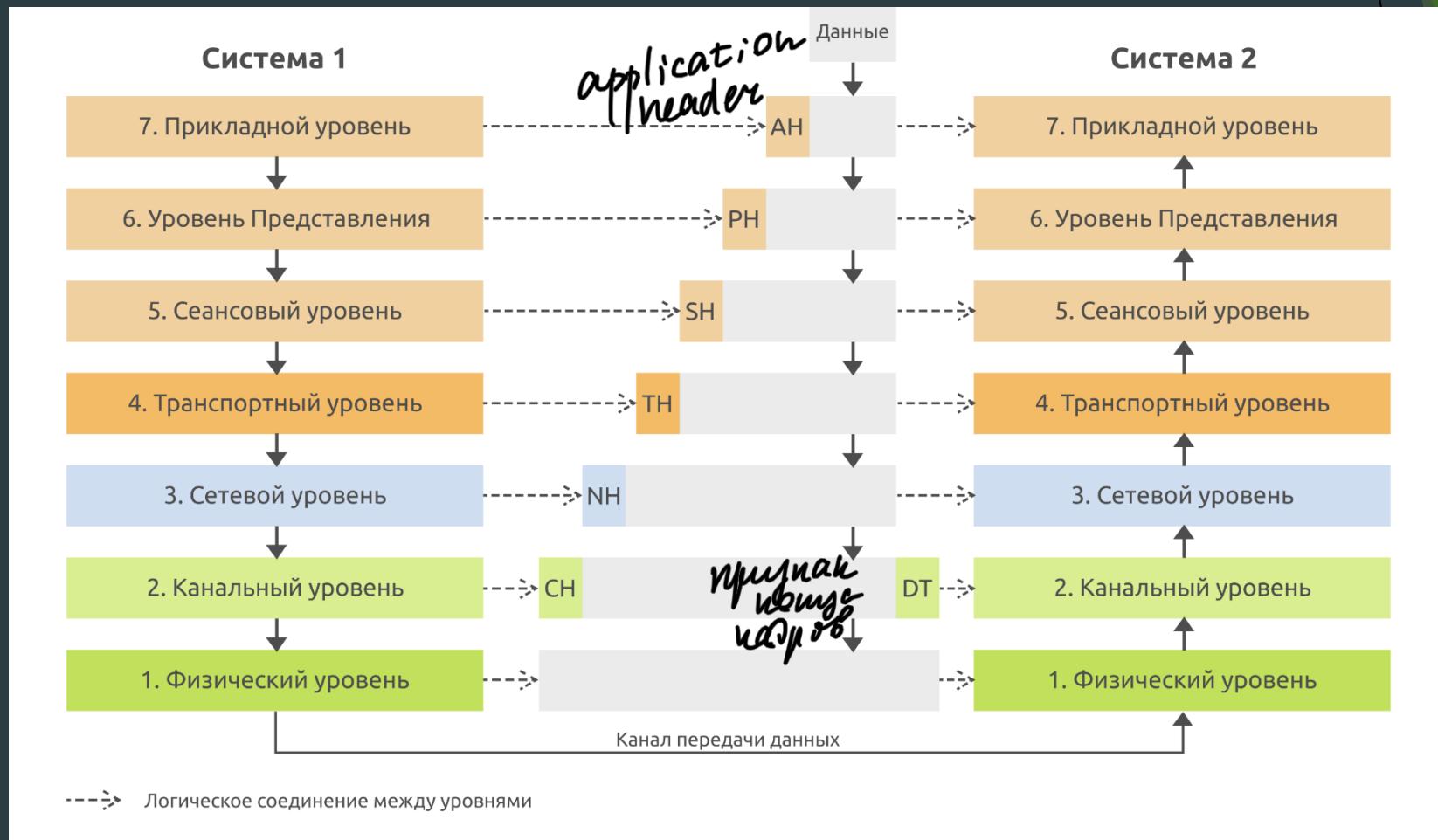
аппарат. и программ. платформ, различ. коммун-
(вз-е различ. оборуд-е
в рамках одной органи.)

Модель ISO - OSI

- ▶ Предпосылки появления модели
- ▶ Основные принципы: - *без приложений от приложений*
 - ▶ Выделение логически изолированных этапов коммуникации
 - ▶ Реализация поддержки этапов в виде программных или аппаратных модулей - «черных ящиков»
 - ▶ Регламентация интерфейсов
 - ▶ Описание функций и задач модулей без описания алгоритмов

Модель ISO - OSI

- находит уровень альтернатив от основных (голосу - я люблю)
- 1 уровень == 1 протокол в OSI



на канальном уровне проек спадающие свойства заголовков

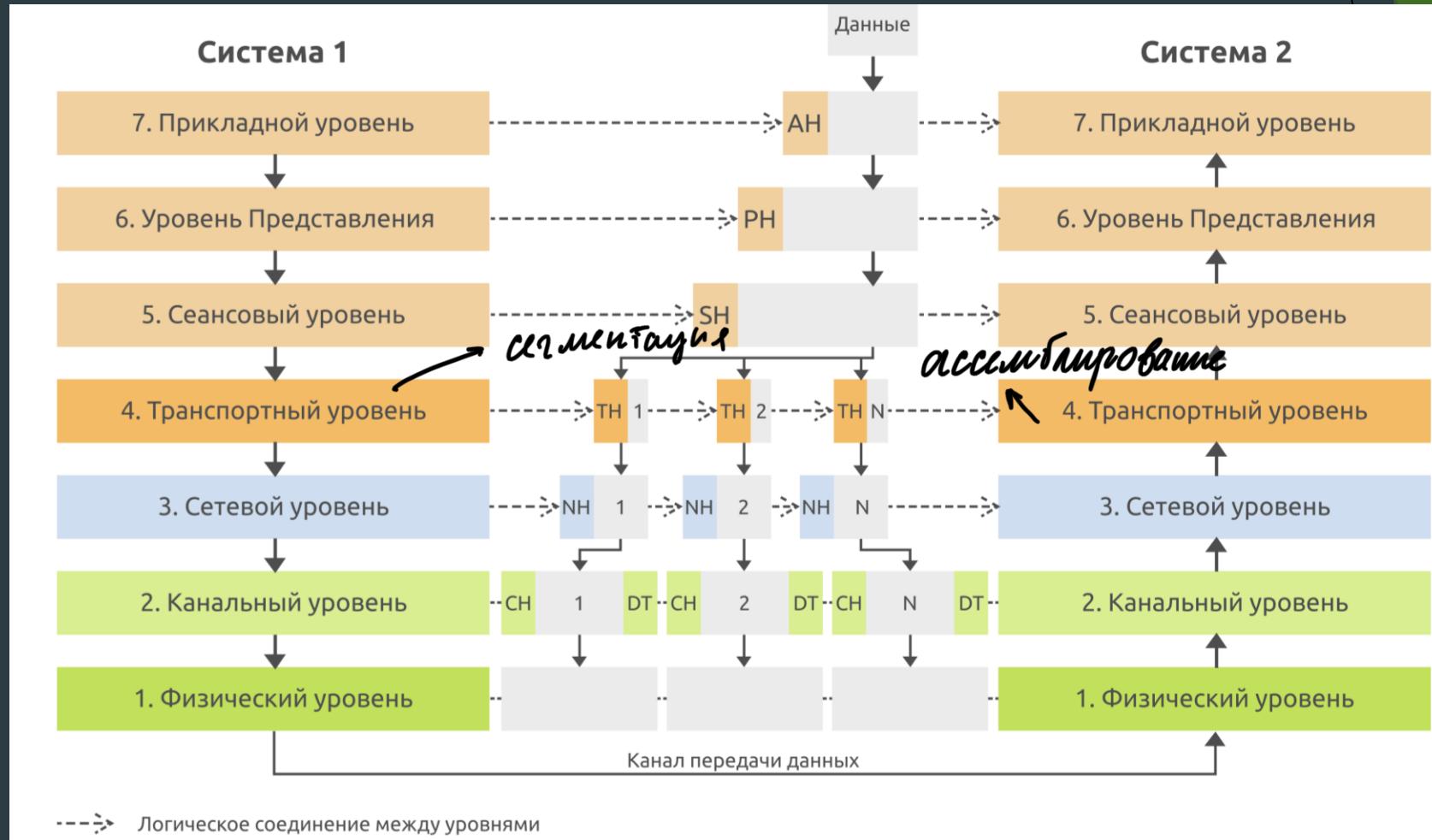
Модель ISO - OSI

сегментации:

1. начало
распред. по
брюмели
(фрагм. фрагм.
сетевых
пакетов)



2. форматы
амибши на др-х,
изобрет. данных



Прикладной уровень (application layer)

Основные функции: – *интерфейс бу-1*

- ▶ Передача служебной информации приложений;
- ▶ Предоставляет приложениям информацию об ошибках.

Примеры протоколов: *TCP/IP* :

FTP (File Transfer Protocol), **Telnet** (TERminaL NETwork),
HTTP (HyperText Transfer Protocol), **POP3** (Post Office
Protocol Version 3), **SMTP** (Simple Mail Transfer
Protocol).

Уровень представления данных (presentation layer)

Основные функции:

- ▶ Сжатие данных;
- ▶ Шифрование данных;
- ▶ Перекодировка данных.

Примеры протоколов:

SSL (Secure Socket Layer), **RDP** – Remote Desktop Protocol

Сеансовый уровень (session layer)

Основные функции:

- ▶ Обеспечивает установление, поддержание и завершение сеанса связи, позволяя приложениям взаимодействовать между собой длительное время.

Примеры протоколов:

L2TP (Layer 2 Tunneling Protocol), **NetBIOS** (Network Basic Input Output System), **PAP** (Password Authentication Protocol), **PPTP** (Point-to-Point Tunneling Protocol), **RPC** (Remote Procedure Call Protocol)

Транспортный уровень (transport layer)

Основные функции:

- ▶ Обеспечивает надежную доставку данных, подтверждение приема и сегментацию потока, получаемого от сеансового уровня.

Примеры протоколов:

TCP (Transmission Control Protocol), UDP (User Datagram Protocol)

Сетевой уровень (network layer)

Основные функции:

- ▶ Решает задачу доставки данных по составной сети, межсетевую адресацию, трансляцию физических адресов в сетевые.

в составн. сети

Примеры протоколов:

IP/IPv4/IPv6 (Internet Protocol), **IPX** (Internetwork Packet Exchange), **IPsec** (Internet Protocol Security), **ICMP** (Internet Control Message Protocol), **RIP** (Routing Information Protocol), **OSPF** (Open Shortest Path First), **ARP** (Address Resolution Protocol).

Канальный уровень (data link layer)

Основные функции: - *управление помехами. оборуд.*

- ▶ Обеспечивает формирование фреймов (frames) – кадров;
- ▶ Обеспечивает контроль ошибок и управление потоком данных (data flow control);
- ▶ Логическое кодирование данных.

Примеры протоколов:

в локальн. сети

ATM, Ethernet, EAPS (Ethernet Automatic Protection Switching), FDDI (Fiber Distributed Data Interface), MPLS (Multiprotocol Label Switching), PPP (Point-to-Point Protocol), SLIP (Serial Line Internet Protocol)

Физический уровень (physical layer)

Основные функции:

- ▶ Обеспечивает физическое кодирование бит кадра в электрические (оптические) сигналы и передачу их по линиям связи;
- ▶ Определяет тип кабелей и разъемов, назначение контактов и формат физических сигналов.

Примеры протоколов:

IEEE 802.15 (Bluetooth), IRDA, EIA-RS-232, EIA-422, Ethernet, DSL, ISDN, IEEE 802.11.

Зачем нужна OSI?

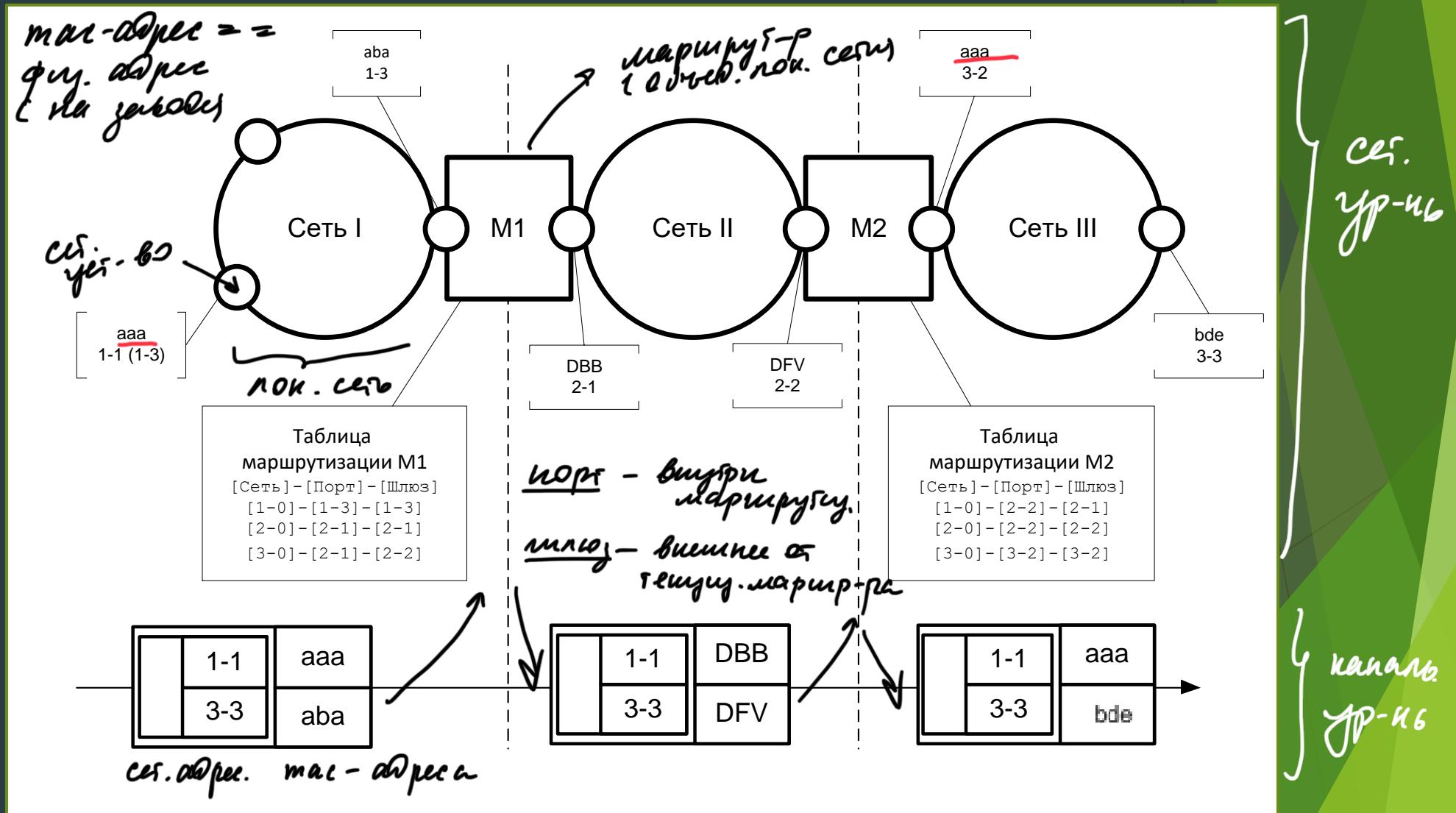
- ▶ Терминология
- ▶ Идеология

использована на практике
- модель функционирования
- цепь из пеп. независимых
пресеков на одном
уровне

Взаимодействие сетевого и канального уровней

software !=
hardware

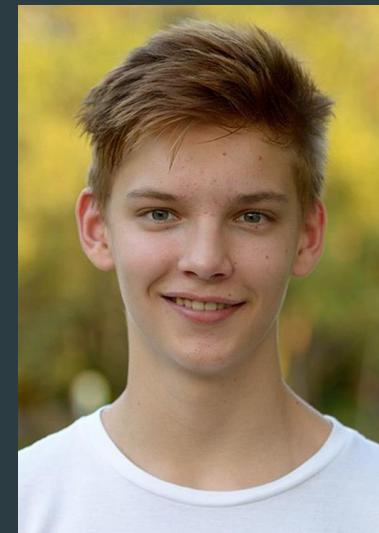
сет. адрес не
использ. т.к.
того, что не канал.
ур-е могут исп.
разные прохожд.



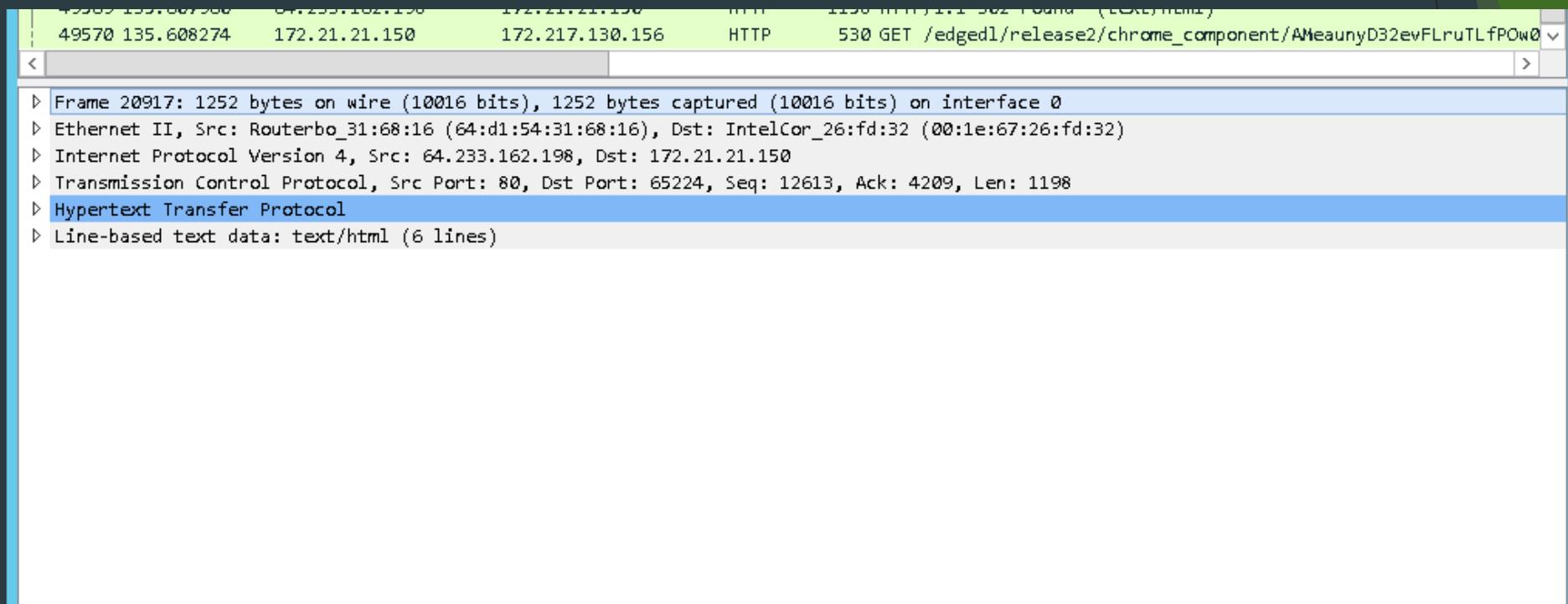
Обратите внимание:

- ▶ В сетях 1 и 3 есть узлы с одинаковыми адресами канального уровня. Это возможно, так как область действия адресации канального уровня - локальная сеть;
- ▶ В составной сети адреса сетевого уровня из одной локальной сети должны иметь одинаковую сетевую часть. Это нужно для решения задачи маршрутизации;
- ▶ В составной сети адреса сетевого уровня должны быть уникальными;
- ▶ За счет процедуры инкапсуляции межсетевое взаимодействие не зависит от природы канальных протоколов в локальных сетях.

Взаимодействие сетевого и канального уровней



Пример реальной инкапсуляции:



Wireshark screenshot showing network traffic details. The top row displays the following information:

Frame	Source IP	Destination IP	Protocol	Port	Request/Response	URL
49569	199.86.98.6	64.233.162.198	HTTP	80	HTTP request	/edgedl/release2/chrome_component/AMeaunyD32evFLruTLfPOw0
49570	135.608274	172.21.21.150	HTTP	530	HTTP response	HTTP/1.1 502 Bad Gateway (TLS/SSL)

The bottom pane shows the detailed structure of the selected frame (Frame 49570), expanded to show the Ethernet II header, Internet Protocol Version 4 header, Transmission Control Protocol header, and Hypertext Transfer Protocol header.

Пример реальной инкапсуляции:

```
Frame 20917: 1252 bytes on wire (10016 bits), 1252 bytes captured (10016 bits) on interface 0
Ethernet II, Src: Routerbo_31:68:16 (64:d1:54:31:68:16), Dst: IntelCor_26:fd:32 (00:1e:67:26:fd:32)
    Destination: IntelCor_26:fd:32 (00:1e:67:26:fd:32)
    Source: Routerbo_31:68:16 (64:d1:54:31:68:16)
    Type: IPv4 (0x0800)
Internet Protocol Version 4, Src: 64.233.162.198, Dst: 172.21.21.150
Transmission Control Protocol, Src Port: 80, Dst Port: 65224, Seq: 12613, Ack: 4209, Len: 1198
Hypertext Transfer Protocol
Line-based text data: text/html (6 lines)
```

Пример реальной инкапсуляции:

```
Frame 20917: 1252 bytes on wire (10016 bits), 1252 bytes captured (10016 bits) on interface 0
Ethernet II, Src: Routerbo_31:68:16 (64:d1:54:31:68:16), Dst: IntelCor_26:fd:32 (00:1e:67:26:fd:32)
Internet Protocol Version 4, Src: 64.233.162.198, Dst: 172.21.21.150
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
    Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
        Total Length: 1238
        Identification: 0x8e1c (36380)
    Flags: 0x0000
        ...0 0000 0000 0000 = Fragment offset: 0
        Time to live: 122
        Protocol: TCP (6)
        Header checksum: 0x08ab [validation disabled]
        [Header checksum status: Unverified]
        Source: 64.233.162.198
        Destination: 172.21.21.150
Transmission Control Protocol, Src Port: 80, Dst Port: 65224, Seq: 12613, Ack: 4209, Len: 1198
Hypertext Transfer Protocol
Line-based text data: text/html (6 lines)
```

Пример реальной инкапсуляции:

```
▷ Frame 20917: 1252 bytes on wire (10016 bits), 1252 bytes captured (10016 bits) on interface 0
▷ Ethernet II, Src: Routerbo_31:68:16 (64:d1:54:31:68:16), Dst: IntelCor_26:fd:32 (00:1e:67:26:fd:32)
▷ Internet Protocol Version 4, Src: 64.233.162.198, Dst: 172.21.21.150
▷ Transmission Control Protocol, Src Port: 80, Dst Port: 65224, Seq: 12613, Ack: 4209, Len: 1198
    Source Port: 80
    Destination Port: 65224
    [Stream index: 76]
    [TCP Segment Len: 1198]
    Sequence number: 12613      (relative sequence number)
    [Next sequence number: 13811      (relative sequence number)]
    Acknowledgment number: 4209      (relative ack number)
    0101 .... = Header Length: 20 bytes (5)
    ▷ Flags: 0x018 (PSH, ACK)
        Window size value: 297
        [Calculated window size: 76032]
        [Window size scaling factor: 256]
        Checksum: 0xe06b [unverified]
        [Checksum Status: Unverified]
        Urgent pointer: 0
    ▷ [SEQ/ACK analysis]
    ▷ [Timestamps]
    TCP payload (1198 bytes)
    ▷ Hypertext Transfer Protocol
    ▷ Line-based text data: text/html (6 lines)
```

Пример реальной инкапсуляции:

```
► Frame 20917: 1252 bytes on wire (10016 bits), 1252 bytes captured (10016 bits) on interface 0
► Ethernet II, Src: Routerbo_31:68:16 (64:d1:54:31:68:16), Dst: IntelCor_26:fd:32 (00:1e:67:26:fd:32)
► Internet Protocol Version 4, Src: 64.233.162.198, Dst: 172.21.21.150
► Transmission Control Protocol, Src Port: 80, Dst Port: 65224, Seq: 12613, Ack: 4209, Len: 1198
└ Hypertext Transfer Protocol
  ┌ HTTP/1.1 302 Found\r\n
  │ Date: Wed, 08 Jul 2020 05:32:14 GMT\r\n
  │ Pragma: no-cache\r\n
  │ Expires: Fri, 01 Jan 1990 00:00:00 GMT\r\n
  │ Cache-Control: no-cache, must-revalidate\r\n
  │ [truncated]Location: http://r15---sn-axq7sn7z.gvt1.com/edged1/chromewebstore/L2Nocm9tZV9leHRlbnNpb24vYmxvYnNvYjFkQUFWdm1aXy12MHFUTGhWQUN
  │ Content-Type: text/html; charset=UTF-8\r\n
  │ Server: ClientMapServer\r\n
  └ Content-Length: 566\r\n
    X-XSS-Protection: 0\r\n
    X-Frame-Options: SAMEORIGIN\r\n
    \r\n
    [HTTP response 12/16]
    [Time since request: 0.286006000 seconds]
    [Prev request in frame: 20572]
    [Prev response in frame: 20575]
    [Request in frame: 20896]
    [Next request in frame: 21593]
    [Next response in frame: 21596]
    [Request URI: http://redirector.gvt1.com/edged1/chromewebstore/L2Nocm9tZV9leHRlbnNpb24vYmxvYnNvYjFkQUFWdm1aXy12MHFUTGhWQUNiMUV1UQ/0.57.44]
    File Data: 566 bytes
  ┌ Line-based text data: text/html (6 lines)
```

Выводы

- ▶ Стек
- ▶ Уровни
- ▶ Интерфейсы