

# Презентация лабораторной работы №3

Бакулин Никита 1032201747

# Цель работы

- Изучение посредством Wireshark кадров Ethernet, анализ PDU протоколов транспортного и прикладного уровней стека TCP/IP.

# Задачи

- Изучение возможностей команды `ipconfig` для ОС типа Windows (`ifconfig` для систем типа Linux). Определение MAC-адреса устройства и его типа.
- Установить на домашнем устройстве Wireshark. С помощью Wireshark захватить и проанализировать пакеты ARP и ICMP в части кадров канального уровня.
- С помощью Wireshark захватить и проанализировать пакеты HTTP, DNS в части заголовков и информации протоколов TCP, UDP, QUIC.
- С помощью Wireshark проанализировать handshake протокола TCP.

# Результаты выполнения

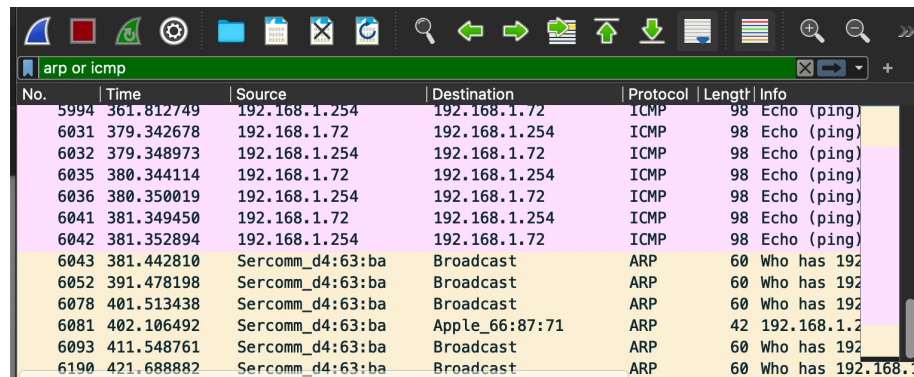
- Изучение возможностей команды `ipconfig` для ОС типа Windows (`ifconfig` для систем типа Linux). Определение MAC-адреса устройства и его типа.

```
nbakulin@mbp-nbakulin ~ % ifconfig
lo0: flags=8049<UP,LOOPBACK,RUNNING,MULTICAST> mtu 16384
    options=1203<RXCSUM,TXCSUM,TXSTATUS,SW_TIMESTAMP>
    inet 127.0.0.1 netmask 0xff000000
    inet6 ::1 prefixlen 128
    inet6 fe80::1%lo0 prefixlen 64 scopeid 0x1
    nd6 options=201<PERFORMNUD,DAD>
gif0: flags=8010<POINTOPOINT,MULTICAST> mtu 1280
stf0: flags=0<> mtu 1280
anpi0: flags=8863<UP,BROADCAST,SMART,RUNNING,SIMPLEX,MULTICAST> mtu 1500
    options=400<CHANNEL_IO>
    ether 76:07:e2:76:64:b2
    inet6 fe80::7407:e2ff:fe76:64b2%anpi0 prefixlen 64 scopeid 0x4
    nd6 options=201<PERFORMNUD,DAD>
    media: none
    status: inactive
anpi1: flags=8863<UP,BROADCAST,SMART,RUNNING,SIMPLEX,MULTICAST> mtu 1500
    options=400<CHANNEL_IO>
    ether 76:07:e2:76:64:b3
    inet6 fe80::7407:e2ff:fe76:64b3%anpi1 prefixlen 64 scopeid 0x5
    nd6 options=201<PERFORMNUD,DAD>
    media: none
    status: inactive
```

Рис. 1

# Результаты выполнения

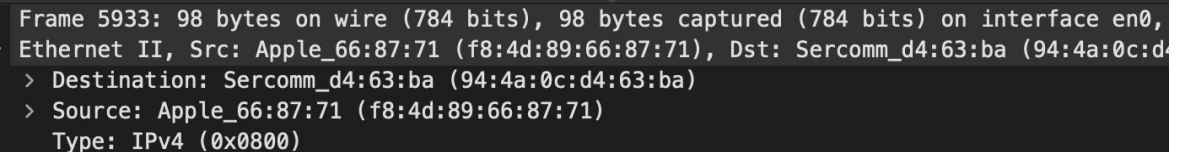
- Установить на домашнем устройстве Wireshark. С помощью Wireshark захватить и проанализировать пакеты ARP и ICMP в части кадров канального уровня.



The image shows the Wireshark interface with a packet capture on the 'arp or icmp' filter. The packet list table is as follows:

No.	Time	Source	Destination	Protocol	Length	Info
5994	361.812749	192.168.1.254	192.168.1.72	ICMP	98	Echo (ping)
6031	379.342678	192.168.1.72	192.168.1.254	ICMP	98	Echo (ping)
6032	379.348973	192.168.1.254	192.168.1.72	ICMP	98	Echo (ping)
6035	380.344114	192.168.1.72	192.168.1.254	ICMP	98	Echo (ping)
6036	380.350019	192.168.1.254	192.168.1.72	ICMP	98	Echo (ping)
6041	381.349450	192.168.1.72	192.168.1.254	ICMP	98	Echo (ping)
6042	381.352894	192.168.1.254	192.168.1.72	ICMP	98	Echo (ping)
6043	381.442810	Sercomm_d4:63:ba	Broadcast	ARP	60	Who has 192.168.1.254
6052	391.478198	Sercomm_d4:63:ba	Broadcast	ARP	60	Who has 192.168.1.254
6078	401.513438	Sercomm_d4:63:ba	Broadcast	ARP	60	Who has 192.168.1.254
6081	402.106492	Sercomm_d4:63:ba	Apple_66:87:71	ARP	42	192.168.1.254
6093	411.548761	Sercomm_d4:63:ba	Broadcast	ARP	60	Who has 192.168.1.254
6190	421.688882	Sercomm_d4:63:ba	Broadcast	ARP	60	Who has 192.168.1.254

Рис. 2



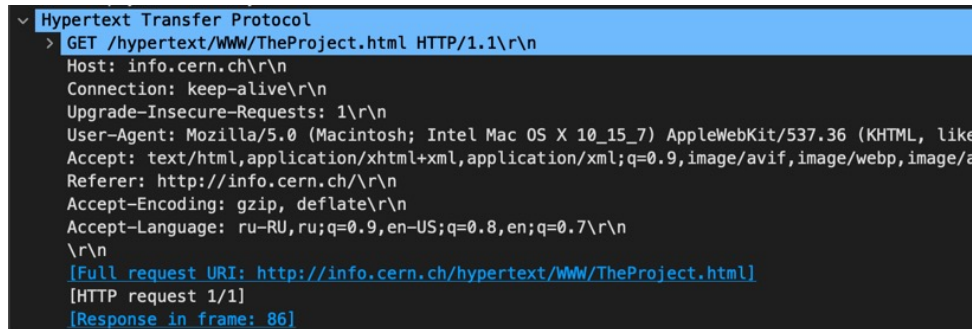
The image shows the packet details pane for frame 5933. The text is as follows:

```
Frame 5933: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface en0,
Ethernet II, Src: Apple_66:87:71 (f8:4d:89:66:87:71), Dst: Sercomm_d4:63:ba (94:4a:0c:d4:63:ba)
> Destination: Sercomm_d4:63:ba (94:4a:0c:d4:63:ba)
> Source: Apple_66:87:71 (f8:4d:89:66:87:71)
Type: IPv4 (0x0800)
```

Рис. 3

# Результаты выполнения

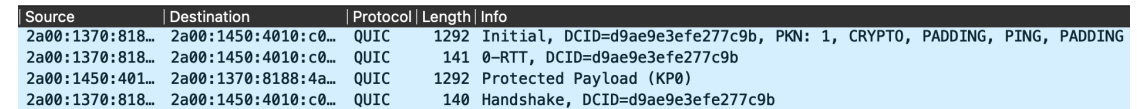
- С помощью Wireshark захватить и проанализировать пакеты HTTP, DNS в части заголовков и информации протоколов TCP, UDP, QUIC.



The screenshot shows a network packet capture interface with a tree view on the left and a packet details pane on the right. The tree view shows a collapsed 'Hypertext Transfer Protocol' entry. The packet details pane is expanded, showing the following information:

```
> GET /hypertext/WWW/TheProject.html HTTP/1.1\r\n
Host: info.cern.ch\r\n
Connection: keep-alive\r\n
Upgrade-Insecure-Requests: 1\r\n
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/a
Referer: http://info.cern.ch/\r\n
Accept-Encoding: gzip, deflate\r\n
Accept-Language: ru-RU,ru;q=0.9,en-US;q=0.8,en;q=0.7\r\n
\r\n
[Full request URI: http://info.cern.ch/hypertext/WWW/TheProject.html]
[HTTP request 1/1]
[Response in frame: 86]
```

Рис. 4



The screenshot shows a network packet capture interface with a table of captured packets. The table has columns for Source, Destination, Protocol, Length, and Info. The packets are QUIC packets.

Source	Destination	Protocol	Length	Info
2a00:1370:818...	2a00:1450:4010:c0...	QUIC	1292	Initial, DCID=d9ae9e3efe277c9b, PKN: 1, CRYPTO, PADDING, PING, PADDING
2a00:1370:818...	2a00:1450:4010:c0...	QUIC	141	0-RTT, DCID=d9ae9e3efe277c9b
2a00:1450:401...	2a00:1370:8188:4a...	QUIC	1292	Protected Payload (KP0)
2a00:1370:818...	2a00:1450:4010:c0...	QUIC	140	Handshake, DCID=d9ae9e3efe277c9b

Рис. 5

# Результаты выполнения

- С помощью Wireshark проанализировать handshake протокола TCP.

Source	Destination	Protocol	Length	Info
192.168.1.72	185.189.255.78	TCP	78	55897 → 22 [SYN] Seq=0 Win=65535
185.189.255.78	192.168.1.72	TCP	74	22 → 55897 [SYN, ACK] Seq=0 Ack=1
192.168.1.72	185.189.255.78	TCP	66	55897 → 22 [ACK] Seq=1 Ack=1 Win=

Рис. 6

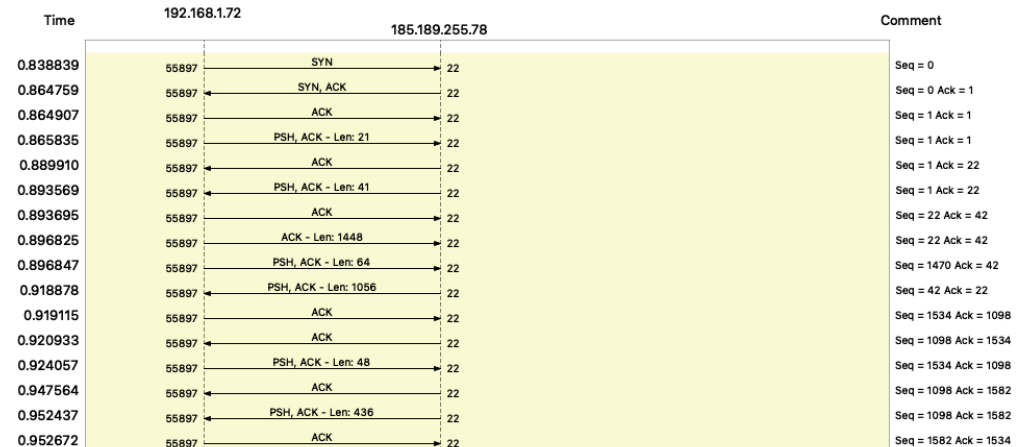


Рис. 7