

Анализ трафика в Wireshark

Лабораторная работа №3

Гафоров Нурмухаммад

9 октября 2025

Российский университет дружбы народов, Москва, Россия

Цели и задачи работы

Цель лабораторной работы

Изучить принципы работы сетевых протоколов ARP, ICMP, TCP, UDP и QUIC посредством анализа трафика в программе **Wireshark**.

Ход выполнения

Определение сетевых параметров

На локальном компьютере с помощью команды `ipconfig` были определены IP-адрес, маска подсети и шлюз по умолчанию.

```
Адаптер Ethernet Ethernet:  
  
DNS-суффикс подключения . . . . . :  
Локальный IPv6-адрес канала . . . . . : fe80::c564:8f78:cb9:e047%18  
IPv4-адрес . . . . . : 192.168.1.180  
Маска подсети . . . . . : 255.255.255.0  
Основной шлюз. . . . . : 192.168.1.1
```

Рис. 1: Определение IP-адреса и шлюза по умолчанию

Проверка доступности шлюза

Был выполнен тест связи командой ping 192.168.1.1. Пакеты получены без потерь, что подтвердило корректную работу сети.

```
PS C:\Users\gaforov\Documents> ping 192.168.1.1

Обмен пакетами с 192.168.1.1 по с 32 байтами данных:
Ответ от 192.168.1.1: число байт=32 время<1мс TTL=64

Статистика Ping для 192.168.1.1:
    Пакетов: отправлено = 4, получено = 4, потеряно = 0
              (0% потеря)
Приблизительное время приема-передачи в мс:
    Минимальное = 0мсек, Максимальное = 0 мсек, Среднее = 0 мсек
PS C:\Users\gaforov\Documents> |
```

Рис. 2: Проверка доступности шлюза

Анализ ICMP-трафика

В Wireshark применён фильтр arp or icmp. Зафиксированы пакеты ICMP Echo Request и Echo Reply между устройством и шлюзом.

The screenshot shows a Wireshark capture window titled "arp or icmp". The packet list pane displays several ICMP Echo Request and Echo Reply messages exchanged between a source IP of 192.168.1.180 and a destination IP of 192.168.1.1. The details pane shows the ICMP payload and the corresponding ARP requests and replies. The bottom pane shows the raw hex and ASCII data for selected packets.

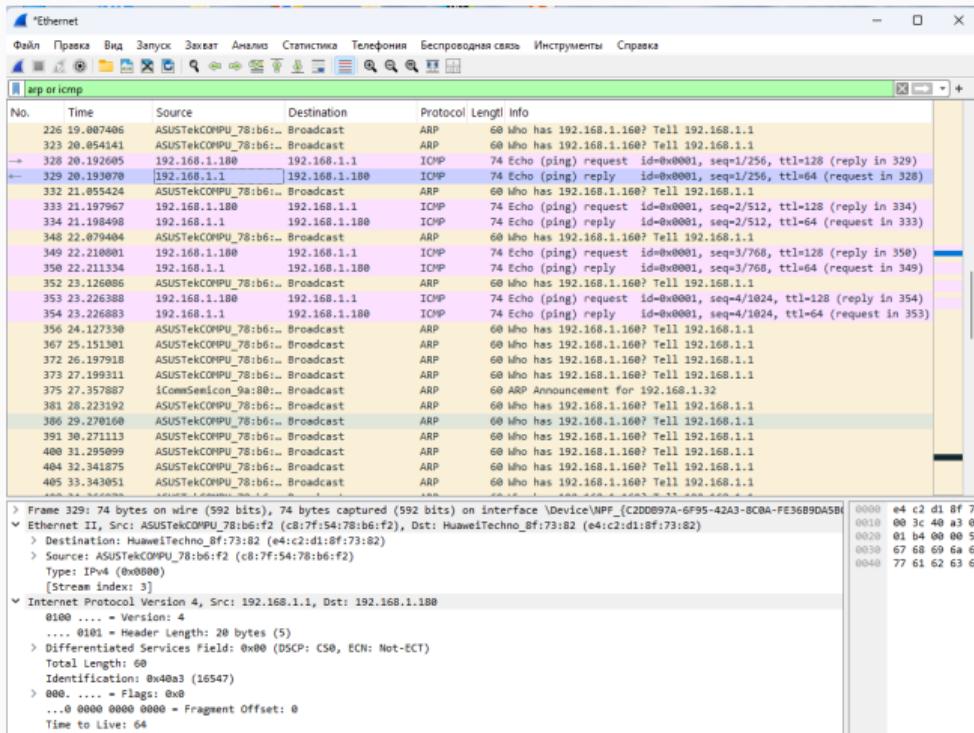
No.	Time	Source	Destination	Protocol	Length	Info
226	19.007406	ASUSTekCOMPU_78:b6:..	Broadcast	ARP	60	Who has 192.168.1.160? Tell 192.168.1.1
323	20.054141	ASUSTekCOMPU_78:b6:..	Broadcast	ARP	60	Who has 192.168.1.160? Tell 192.168.1.1
→ 328	20.192695	192.168.1.180	192.168.1.1	ICMP	74	Echo (ping) request id=0x0001, seq=1/256, ttl=128 (reply in 328)
← 329	20.193078	192.168.1.1	192.168.1.180	ICMP	74	Echo (ping) reply id=0x0001, seq=1/256, ttl=64 (request in 328)
332	21.055424	ASUSTekCOMPU_78:b6:..	Broadcast	ARP	60	Who has 192.168.1.160? Tell 192.168.1.1
333	21.197967	192.168.1.180	192.168.1.1	ICMP	74	Echo (ping) request id=0x0001, seq=2/512, ttl=128 (reply in 334)
334	21.198498	192.168.1.1	192.168.1.180	ICMP	74	Echo (ping) reply id=0x0001, seq=2/512, ttl=64 (request in 333)
348	22.079404	ASUSTekCOMPU_78:b6:..	Broadcast	ARP	60	Who has 192.168.1.160? Tell 192.168.1.1
349	22.210801	192.168.1.180	192.168.1.1	ICMP	74	Echo (ping) request id=0x0001, seq=3/768, ttl=128 (reply in 350)
350	22.211334	192.168.1.1	192.168.1.180	ICMP	74	Echo (ping) reply id=0x0001, seq=3/768, ttl=64 (request in 349)
352	23.126086	ASUSTekCOMPU_78:b6:..	Broadcast	ARP	60	Who has 192.168.1.160? Tell 192.168.1.1
353	23.226388	192.168.1.180	192.168.1.1	ICMP	74	Echo (ping) request id=0x0001, seq=4/1024, ttl=128 (reply in 354)
354	23.226883	192.168.1.1	192.168.1.180	ICMP	74	Echo (ping) reply id=0x0001, seq=4/1024, ttl=64 (request in 353)
356	24.127338	ASUSTekCOMPU_78:b6:..	Broadcast	ARP	60	Who has 192.168.1.160? Tell 192.168.1.1
367	25.151301	ASUSTekCOMPU_78:b6:..	Broadcast	ARP	60	Who has 192.168.1.160? Tell 192.168.1.1
372	26.197918	ASUSTekCOMPU_78:b6:..	Broadcast	ARP	60	Who has 192.168.1.160? Tell 192.168.1.1
373	27.199311	ASUSTekCOMPU_78:b6:..	Broadcast	ARP	60	Who has 192.168.1.160? Tell 192.168.1.1
375	27.357887	iComSemicon_9a:00:..	Broadcast	ARP	60	ARP Announcement for 192.168.1.32
381	28.223192	ASUSTekCOMPU_78:b6:f2	Broadcast	ARP	60	Who has 192.168.1.160? Tell 192.168.1.1
386	29.270168	ASUSTekCOMPU_78:b6:f2	Broadcast	ARP	60	Who has 192.168.1.160? Tell 192.168.1.1
391	30.271113	ASUSTekCOMPU_78:b6:f2	Broadcast	ARP	60	Who has 192.168.1.160? Tell 192.168.1.1
406	31.295099	ASUSTekCOMPU_78:b6:f2	Broadcast	ARP	60	Who has 192.168.1.160? Tell 192.168.1.1
408	32.341875	ASUSTekCOMPU_78:b6:f2	Broadcast	ARP	60	Who has 192.168.1.160? Tell 192.168.1.1
405	33.343051	ASUSTekCOMPU_78:b6:f2	Broadcast	ARP	60	Who has 192.168.1.160? Tell 192.168.1.1

Frame 328: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface '\Device\NPF_{C200097A-6F95-42A3-8C0A-FE30B9045040}'
Ethernet II, Src: HuaweiTechno_Bf:73:82 (e4:c2:d1:8f:73:82), Dst: ASUSTekCOMPU_78:b6:f2 (c8:7f:54:78:b6:f2)
> Destination: ASUSTekCOMPU_78:b6:f2 (c8:7f:54:78:b6:f2)
> Source: HuaweiTechno_Bf:73:82 (e4:c2:d1:8f:73:82)
> Type: IPv4 (0x0800)
> [Stream index: 3]
> Internet Protocol Version 4, Src: 192.168.1.180, Dst: 192.168.1.1
> 0100 - Version: 4
> 0101 - Header Length: 20 bytes (5)
> Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
> Total Length: 60
> Identification: 0xae19 (44569)
> 0000 - Flags: 0x0
> ..0 0000 0000 0000 - Fragment Offset: 0
> Time to Live: 128

0000 c8 7f 54 78 b1
0010 00 3c ae 19 01
0020 01 01 08 00 4d
0030 67 68 69 6a 61
0040 77 61 62 63 64

Анализ ICMP-трафика

В Wireshark применён фильтр `arp or icmp`. Зафиксированы пакеты ICMP Echo Request и Echo Reply между устройством и шлюзом.



Анализ ARP-протокола

Перед ICMP-обменом выполняется разрешение MAC-адреса шлюза через ARP-запрос и ответ.

No.	Time	Source	Destination	Protocol	Length	Info
354	23.226883	192.168.1.1	192.168.1.180	ICMP	74	Echo (ping) reply id=0x0001, seq=4/1024, ttl=64 (request in 353)
356	24.127330	ASUSTekCOMP_78:b6:f2	Broadcast	ARP	60	Who has 192.168.1.180? Tell 192.168.1.1
367	25.151301	ASUSTekCOMP_78:b6:f2	Broadcast	ARP	60	Who has 192.168.1.180? Tell 192.168.1.1
372	26.197918	ASUSTekCOMP_78:b6:f2	Broadcast	ARP	60	Who has 192.168.1.180? Tell 192.168.1.1
373	27.199311	ASUSTekCOMP_78:b6:f2	Broadcast	ARP	60	Who has 192.168.1.180? Tell 192.168.1.1
375	27.357887	iCommSilicon_9:a:00	Broadcast	ARP	60	ARP Announcement for 192.168.1.32
381	28.2232192	ASUSTekCOMP_78:b6:f2	Broadcast	ARP	60	Who has 192.168.1.180? Tell 192.168.1.1
386	29.270161	ASUSTekCOMP_78:b6:f2	Broadcast	ARP	60	Who has 192.168.1.180? Tell 192.168.1.1
391	30.271113	ASUSTekCOMP_78:b6:f2	Broadcast	ARP	60	Who has 192.168.1.180? Tell 192.168.1.1
400	31.295899	ASUSTekCOMP_78:b6:f2	Broadcast	ARP	60	Who has 192.168.1.180? Tell 192.168.1.1
404	32.341875	ASUSTekCOMP_78:b6:f2	Broadcast	ARP	60	Who has 192.168.1.180? Tell 192.168.1.1
405	33.343851	ASUSTekCOMP_78:b6:f2	Broadcast	ARP	60	Who has 192.168.1.180? Tell 192.168.1.1
408	34.366972	ASUSTekCOMP_78:b6:f2	Broadcast	ARP	60	Who has 192.168.1.180? Tell 192.168.1.1
412	35.419544	ASUSTekCOMP_78:b6:f2	Broadcast	ARP	60	Who has 192.168.1.180? Tell 192.168.1.1
417	36.414984	ASUSTekCOMP_78:b6:f2	Broadcast	ARP	60	Who has 192.168.1.180? Tell 192.168.1.1
422	37.438965	ASUSTekCOMP_78:b6:f2	Broadcast	ARP	60	Who has 192.168.1.180? Tell 192.168.1.1
441	38.485555	ASUSTekCOMP_78:b6:f2	Broadcast	ARP	60	Who has 192.168.1.180? Tell 192.168.1.1
461	39.486859	ASUSTekCOMP_78:b6:f2	Broadcast	ARP	60	Who has 192.168.1.180? Tell 192.168.1.1
466	40.518813	ASUSTekCOMP_78:b6:f2	Broadcast	ARP	60	Who has 192.168.1.180? Tell 192.168.1.1
469	41.557482	ASUSTekCOMP_78:b6:f2	Broadcast	ARP	60	Who has 192.168.1.180? Tell 192.168.1.1
473	42.558777	ASUSTekCOMP_78:b6:f2	Broadcast	ARP	60	Who has 192.168.1.180? Tell 192.168.1.1
474	43.598758	ASUSTekCOMP_78:b6:f2	HuaweiTechno_8f:73:82	ARP	60	Who has 192.168.1.180? Tell 192.168.1.1
475	43.598789	HuaweiTechno_8f:73:82	ASUSTekCOMP_78:b6:f2	ARP	42	192.168.1.180 is at e4:c2:d1:8f:73:82
476	43.582761	ASUSTekCOMP_78:b6:f2	Broadcast	ARP	60	Who has 192.168.1.180? Tell 192.168.1.1

Frame 474: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface \Device\NPF_{C2D0B97A-6F95-42A3-8C0A-FE36B90A5B
Ethernet II, Src: ASUSTekCOMP_78:b6:f2 (c8:7f:54:78:b6:f2), Dst: HuaweiTechno_8f:73:82 (e4:c2:d1:8f:73:82)
> Destination: HuaweiTechno_8f:73:82 (e4:c2:d1:8f:73:82)
> Source: ASUSTekCOMP_78:b6:f2 (c8:7f:54:78:b6:f2)
> Type: ARP (0x0806)
[Stream index: 3]
Padding: 00000000000000000000000000000000
Trailer: 11c450bf
▼ Address Resolution Protocol (request)
Hardware type: Ethernet (1)
Protocol type: IPv4 (0x0800)
Hardware size: 6
Protocol size: 4
Opcode: request (1)
Sender MAC address: ASUSTekCOMP_78:b6:f2 (c8:7f:54:78:b6:f2)
Sender IP address: 192.168.1.1
Target MAC address: 00:00:00:00:00:00 (00:00:00:00:00:00)
Target IP address: 192.168.1.180

Проверка связи с внешним ресурсом

Командой ping yandex.ru выполнен тест соединения с удалённым узлом. Среднее время отклика составило 8 мс.

```
PS C:\Users\gaforov\Documents>
PS C:\Users\gaforov\Documents> ping yandex.ru

Обмен пакетами с yandex.ru [77.88.55.88] с 32 байтами данных:
Ответ от 77.88.55.88: число байт=32 время=8мс TTL=247

Статистика Ping для 77.88.55.88:
    Пакетов: отправлено = 4, получено = 4, потеряно = 0
                (0% потеря)
Приблизительное время приема-передачи в мс:
    Минимальное = 8мсек, Максимальное = 8 мсек, Среднее = 8 мсек
PS C:\Users\gaforov\Documents> |
```

Рис. 6: Проверка доступности внешнего ресурса

Анализ ICMP при обращении к внешнему серверу

Фильтр arp or icmp показал обмен ICMP-запросами с адресом 77.88.55.88 (yandex.ru).

№	Слой	Источник	Назначение	Протокол	Действие
27	8.287285	192.168.1.180	77.88.55.88	ICMP	74 Echo (ping) request id=0x0001, seq=5/1280, ttl=128 (reply in 28)
28	8.295782	77.88.55.88	192.168.1.180	ICMP	74 Echo (ping) reply id=0x0001, seq=5/1280, ttl=247 (request in 27)
34	9.291634	192.168.1.180	77.88.55.88	ICMP	74 Echo (ping) request id=0x0001, seq=6/1536, ttl=128 (reply in 35)
35	9.300256	77.88.55.88	192.168.1.180	ICMP	74 Echo (ping) reply id=0x0001, seq=6/1536, ttl=247 (request in 34)
36	10.307004	192.168.1.180	77.88.55.88	ICMP	74 Echo (ping) request id=0x0001, seq=7/1792, ttl=128 (reply in 37)
37	10.315621	77.88.55.88	192.168.1.180	ICMP	74 Echo (ping) reply id=0x0001, seq=7/1792, ttl=247 (request in 36)
40	11.320312	192.168.1.180	77.88.55.88	ICMP	74 Echo (ping) request id=0x0001, seq=8/2048, ttl=128 (reply in 41)
41	11.328981	77.88.55.88	192.168.1.180	ICMP	74 Echo (ping) reply id=0x0001, seq=8/2048, ttl=247 (request in 40)
49	14.397935	ASUSTekCOMPU_78:b6:..	HuaweiTechno_8f:73:..	ARP	60 Who has 192.168.1.180? Tell 192.168.1.1
50	14.397966	HuaweiTechno_8f:73:..	ASUSTekCOMPU_78:b6:..	ARP	42 192.168.1.180 is at e4:c2:d1:8f:73:82

> Frame 27: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface \Device\NPF_{C2DDB97A-6F95-42A3-8C0A-FE36B90A5BC:
✓ Ethernet II, Src: HuaweiTechno_8f:73:82 (e4:c2:d1:8f:73:82), Dst: ASUSTekCOMPU_78:b6:f2 (c8:7f:54:78:b6:f2)
> Source: HuaweiTechno_8f:73:82 (e4:c2:d1:8f:73:82)
Type: IPv4 (0x0800)
[Stream index: 0]
✓ Internet Protocol Version 4, Src: 192.168.1.180, Dst: 77.88.55.88
0100 = Version: 4
.... 0101 = Header Length: 20 bytes (5)
> Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
Total Length: 60
Identification: 0x2649 (9801)
> 000. = Flags: 0x0
...0 0000 0000 0000 = Fragment Offset: 0
Time to Live: 128
Protocol: ICMP (1)
Header Checksum: 0x0000 [validation disabled]
[Header checksum status: Unverified]
Source Address: 192.168.1.180
Destination Address: 77.88.55.88
[src, dst, type] = [192.168.1.180, 77.88.55.88, 1]
0000 c8 7f 54 78 b6
0010 00 3c 26 49 06
0020 37 58 08 00 4c
0030 67 68 69 68 68
0040 77 61 62 63 64

Анализ HTTP и TCP

При фильтрации по `http` зафиксированы запросы `GET` и ответы `HTTP/1.1 304 Not Modified`.
Данные передаются через `TCP` с портом `80`.

The screenshot shows a NetworkMiner capture session titled "Ethernet". The main pane lists network traffic with columns: No., Time, Source, Destination, Protocol, Length!, Info. Several entries are highlighted in green, indicating they are related to the selected frame. The selected frame (Frame 28) is detailed in the bottom panes:

- Frame 28:** 250 bytes on wire (2000 bits), 250 bytes captured (2000 bits) on interface \Device\NPF_{C20D897A-8F95-42A3-8C0A-FE368
- Ethernet II, Src: ASUSTekCPU_78:6f:f2 (c8:7f:54:78:6f:f2), Dst: HuaweiTechno_8f:73:82 (e4:c2:d1:8f:73:82)**
- Internet Protocol Version 4, Src: 188.184.67.127, Dst: 192.168.1.180**
- Transmission Control Protocol, Src Port: 80, Dst Port: 50781, Seq: 1, Ack: 556, Len: 196**

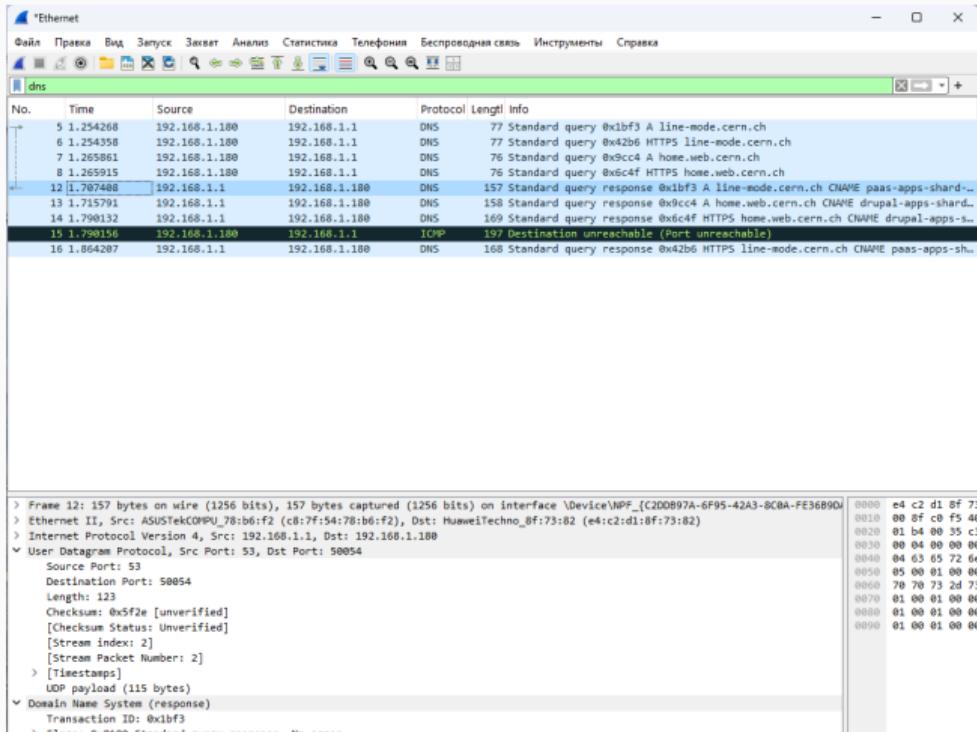
The details pane shows the structure of the selected frame:

- Source Port: 80
- Destination Port: 50781
- [Stream index: 2]
- [Stream Packet Number: 6]
- [Conversation completeness: Complete, WITH_DATA (31)]
- [TCP Segment Len: 196]
- Sequence Number: 1 (relative sequence number)
- Sequence Number (raw): 545287938
- [Next Sequence Number: 197 (relative sequence number)]
- Acknowledgment Number: 556 (relative ack number)
- Acknowledgment number (raw): 3284902225
- 0101 a Header Length: 20 bytes (5)

The bytes pane shows the raw hex and ASCII data of the captured frame.

Анализ DNS и UDP

С помощью фильтра `dns` зафиксированы DNS-запросы и ответы, использующие порты 53 (сервер) и 50854 (клиент).



Анализ протокола QUIC

Фильтр quic выявил пакеты **Initial**, **Handshake**, **PING** и **CRYPTO**, передаваемые по UDP.
Протокол обеспечивает быструю и защищённую передачу данных.

The screenshot shows a Wireshark capture of QUIC traffic on the 'Ethernet' interface. The packet list pane displays 176 QUIC frames, primarily from source 192.168.1.188 to destination 192.250.74.131. Key frame types include:

- Frame 1154: 1292 Initial, DCID=0x4d5163ef9a29bf, PKN: 1, PING, PING, PADDING, PING...
- Frame 1155: 1292 Initial, DCID=0x4d5163ef9a29bf, PKN: 2, CRYPTO, PING, CRYPTO, CR...
- Frame 1156: 124 0-RTT, DCID=0x4d5163ef9a290f
- Frame 1157: 1292 Initial, DCID=0x4d5163ef9a290f, PKN: 1, PADDING, PING, CRYPTO, CR...
- Frame 1158: 1292 Initial, DCID=0x4d5163ef9a290f, PKN: 2, CRYPTO
- Frame 1159: 1292 Initial, DCID=0x4d5163ef9a290f, PKN: 3, PING, PADDING, PING, PING...
- Frame 1160: 124 0-RTT, DCID=0x4d5163ef9a292f1
- Frame 1162: 82 Initial, SCID=0x4d5163ef9a29bf, PKN: 1, ACK
- Frame 1163: 1292 Initial, SCID=0x4d5163ef9a29bf, PKN: 2, ACK, PADDING
- Frame 1164: 82 Initial, SCID=fbea2bb6c8fe92f1, PKN: 1, ACK
- Frame 1165: 82 Initial, SCID=fbea2bb6c8fe92f1, PKN: 2, ACK
- Frame 1166: 1292 Initial, SCID=fbea2bb6c8fe92f1, PKN: 3, ACK, PADDING
- Frame 1167: 1292 Initial, SCID=fbea2bb6c8fe92f1, PKN: 4, ACK, PADDING
- Frame 1168: 1292 Initial, SCID=0x4d5163ef9a29bf, PKN: 3, CRYPTO, PADDING
- Frame 1169: 349 Protected Payload (XPN)
- Frame 1170: 985 Protected Payload (XPN)
- Frame 1171: 76 Protected Payload (XPN)
- Frame 1172: 120 Handshake, DCID=0x4d5163ef9a29bf
- Frame 1173: 73 Protected Payload (XPN), DCID=0x4d5163ef9a29bf
- Frame 1174: 1292 Initial, SCID=fbea2bb6c8fe92f1, PKN: 5, CRYPTO, PADDING
- Frame 1175: 342 Protected Payload (XPN)
- Frame 1176: 120 Handshake, DCID=fbea2bb6c8fe92f1
- Frame 1177: 998 Protected Payload (XPN)
- Frame 1178: 73 Protected Payload (XPN), DCID=fbea2bb6c8fe92f1

The details and bytes panes are visible at the bottom of the interface.

Анализ handshake TCP

Исследован процесс установления TCP-соединения (Three-Way Handshake):

1. SYN – инициация соединения клиентом.
2. SYN/ACK – подтверждение сервером.
3. ACK – завершение установления.

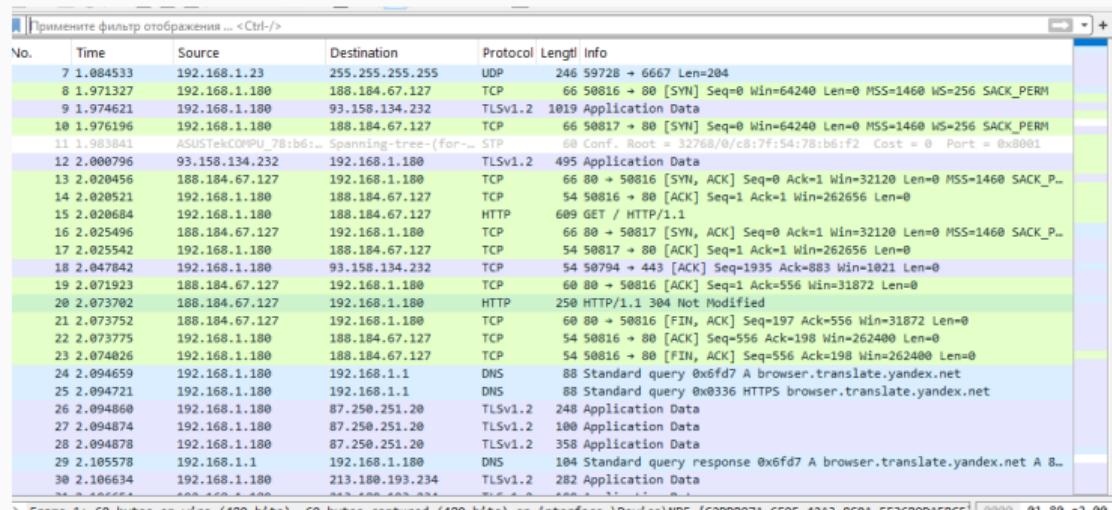
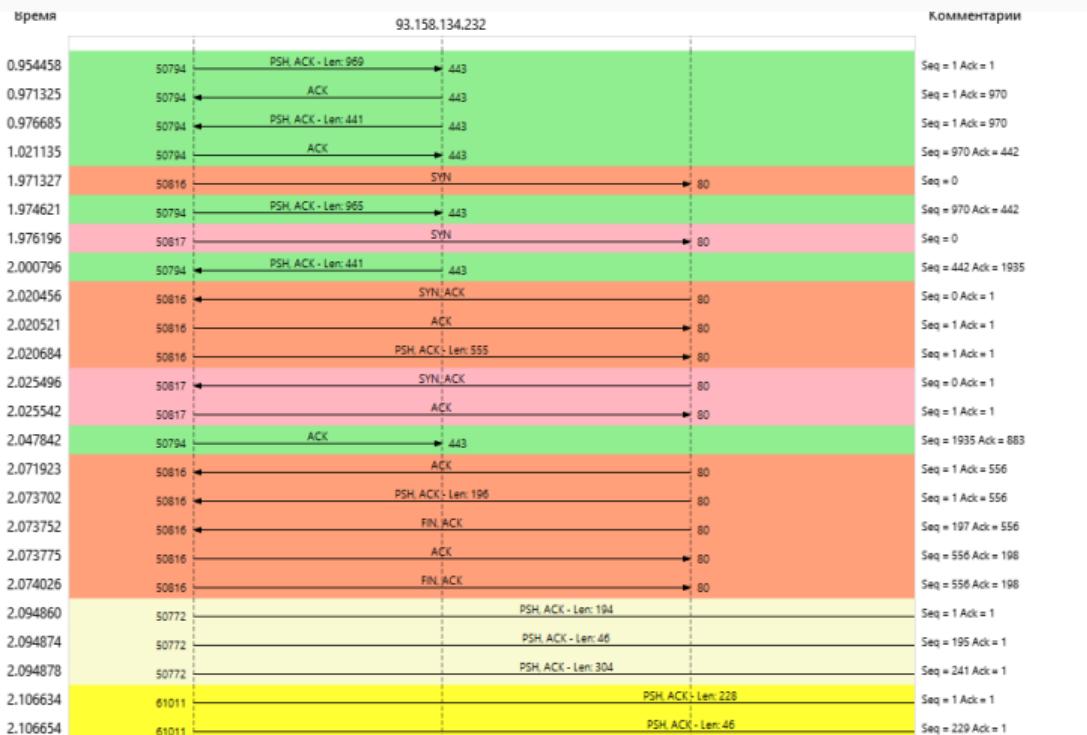


Рис. 11: Захват TCP handshake в Wireshark

График потока TCP

На графике потока отображается последовательность обмена сегментами SYN → SYN/ACK → ACK, подтверждающая корректное установление соединения.



Выводы

Итог лабораторной работы

В результате работы были исследованы протоколы ARP, ICMP, TCP, UDP и QUIC с помощью анализатора **Wireshark**.

Получены практические навыки по анализу сетевого трафика и пониманию принципов работы транспортного уровня стека TCP/IP.

Процесс трёхэтапного установления соединения (**TCP Handshake**) был подробно изучен, что подтвердило надёжность и согласованность протокола TCP.