Отчёт по лабораторной работе 3

Анализ трафика в Wireshark

Гафоров Нурмухаммад

Содержание

Список иллюстраций

Список таблиц

1 Введение

1.1 Цель работы

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

2 Ход выполнения

На рабочем компьютере был установлен и запущен анализатор сетевого трафика **Wireshark**. Для начала захвата данных выбран активный сетевой интерфейс Ethernet. После запуска фиксации пакетов в консоли Windows с помощью команды `ipconfig` были определены сетевые параметры устройства: IP-адрес **192.168.1.180**, маска подсети **255.255.255.0** и основной шлюз **192.168.1.1**.

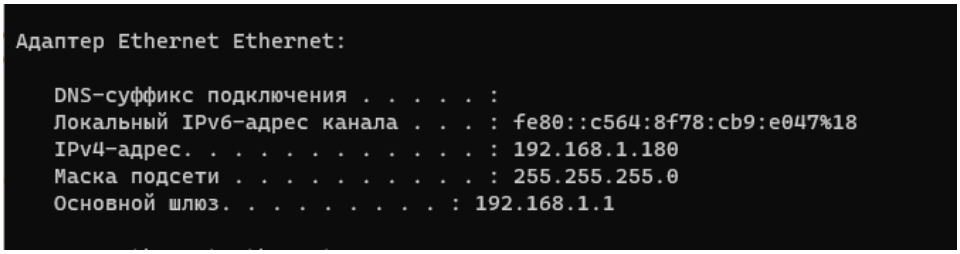


Рис. 2.1: Определение IP-адреса и шлюза по умолчанию

Далее был выполнен тест связи с шлюзом по умолчанию командой `ping 192.168.1.1`. Все пакеты успешно доставлены, потери отсутствовали, что подтвердило корректность сетевого подключения.

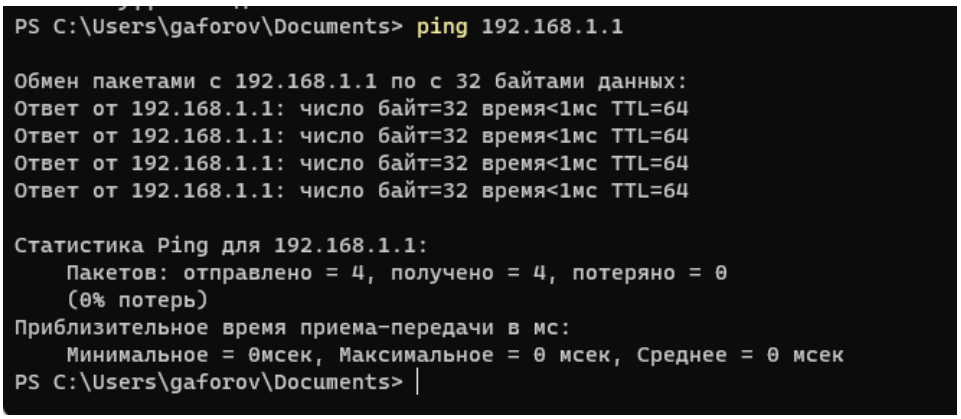


Рис. 2.2: Проверка доступности шлюза

В процессе захвата трафика в **Wireshark** применён фильтр `arp or icmp`. В списке пакетов отобразились ICMP-запросы и ответы, соответствующие выполненной команде `ping`. В кадре ICMP-запроса длина составила 74 байта, тип кадра — **Ethernet II**. В поле источника указан MAC-адрес устройства **ASUSTeKCOMPu_7B:B6:F2**, а в поле назначения — MAC-адрес шлюза **HuaweiTechno_8F:73:82**.

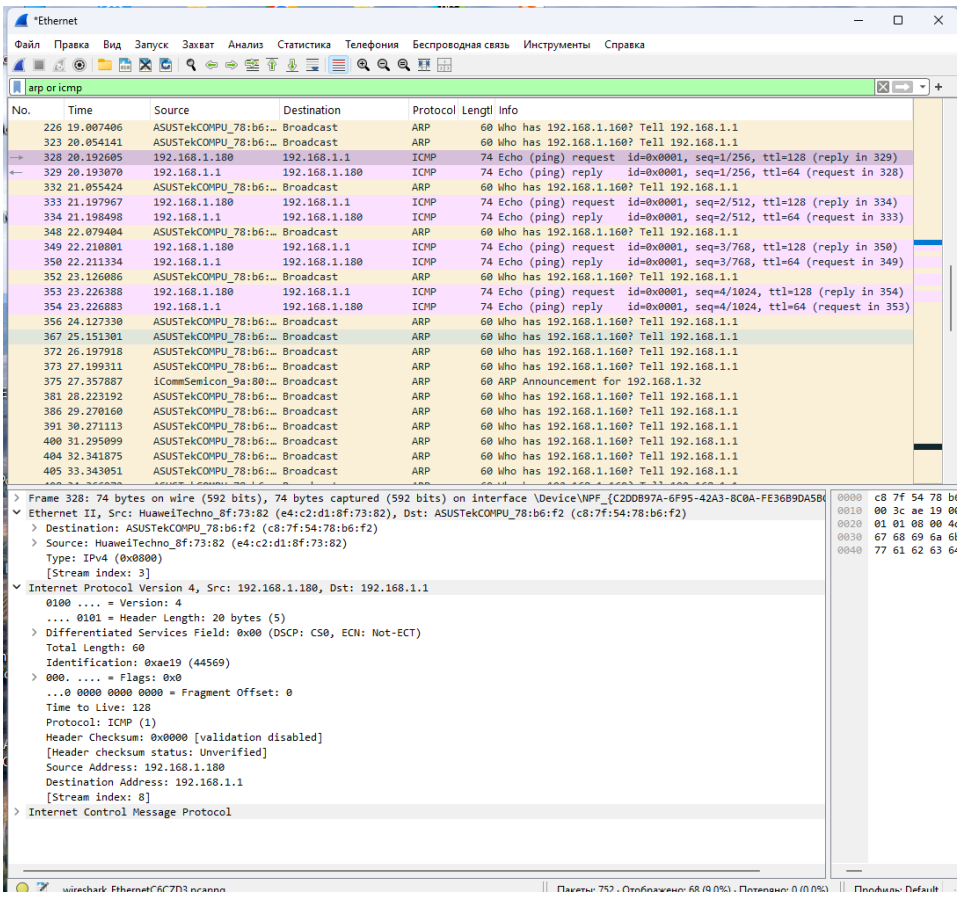


Рис. 2.3: ICMP Echo Request к шлюзу

В ICMP-ответе наблюдается обратная ситуация: источником выступает MAC-адрес шлюза, а получателем — MAC-адрес компьютера. Это подтверждает двусторонний обмен эхо-запросами между узлами.

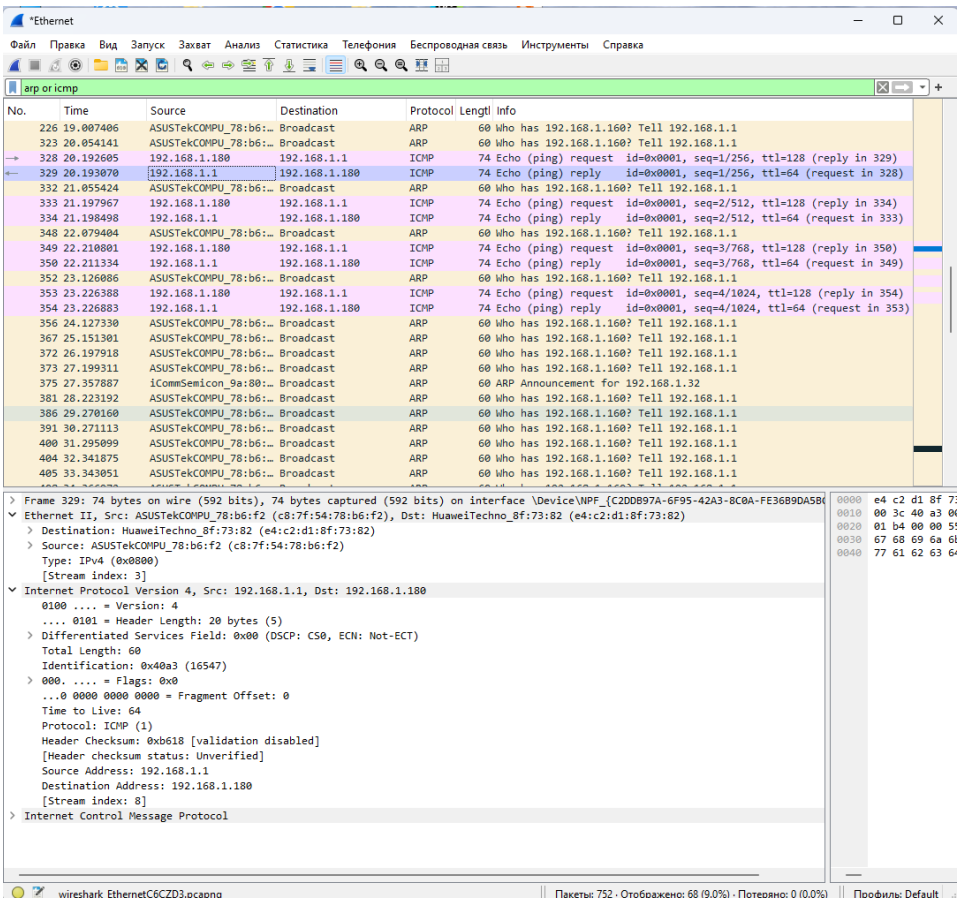


Рис. 2.4: ICMP Echo Reply от шлюза

При анализе кадров ARP установлено, что перед началом ICMP-обмена выполняется разрешение адреса шлюза. В кадрах ARP-запроса устройство отправляет широковещательное сообщение с запросом MAC-адреса для IP **192.168.1.1**, а в ARP-ответе шлюз возвращает свой физический адрес.

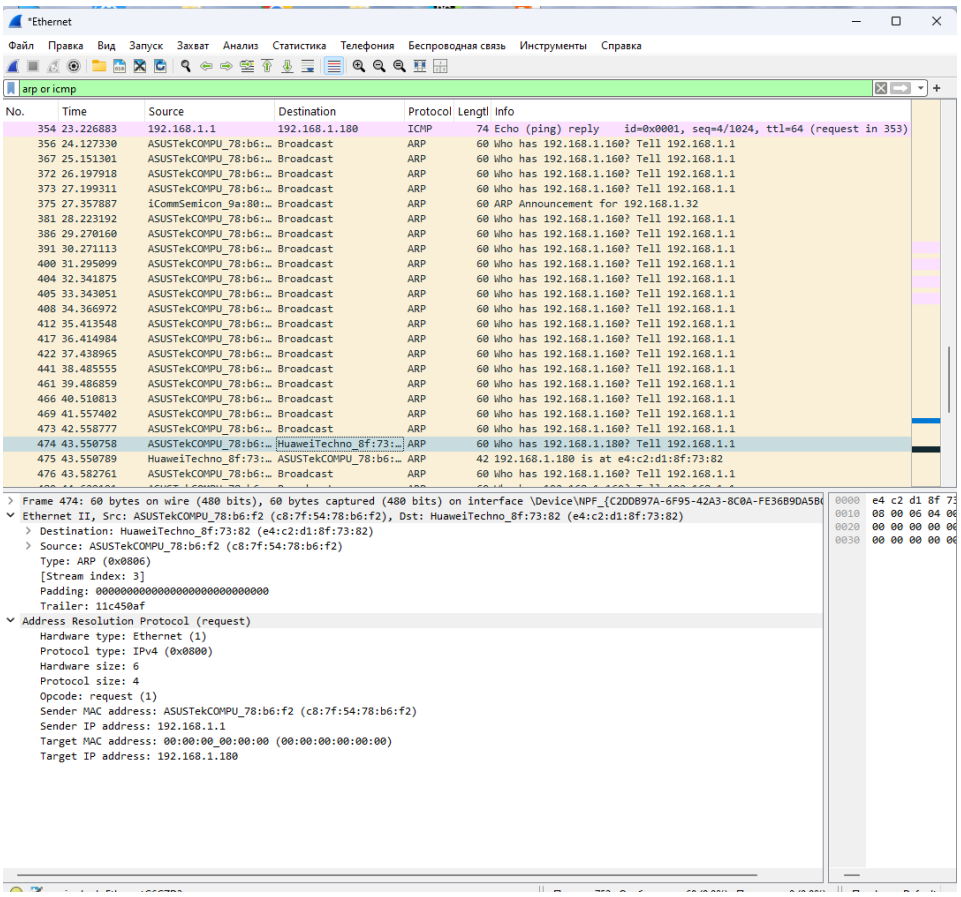


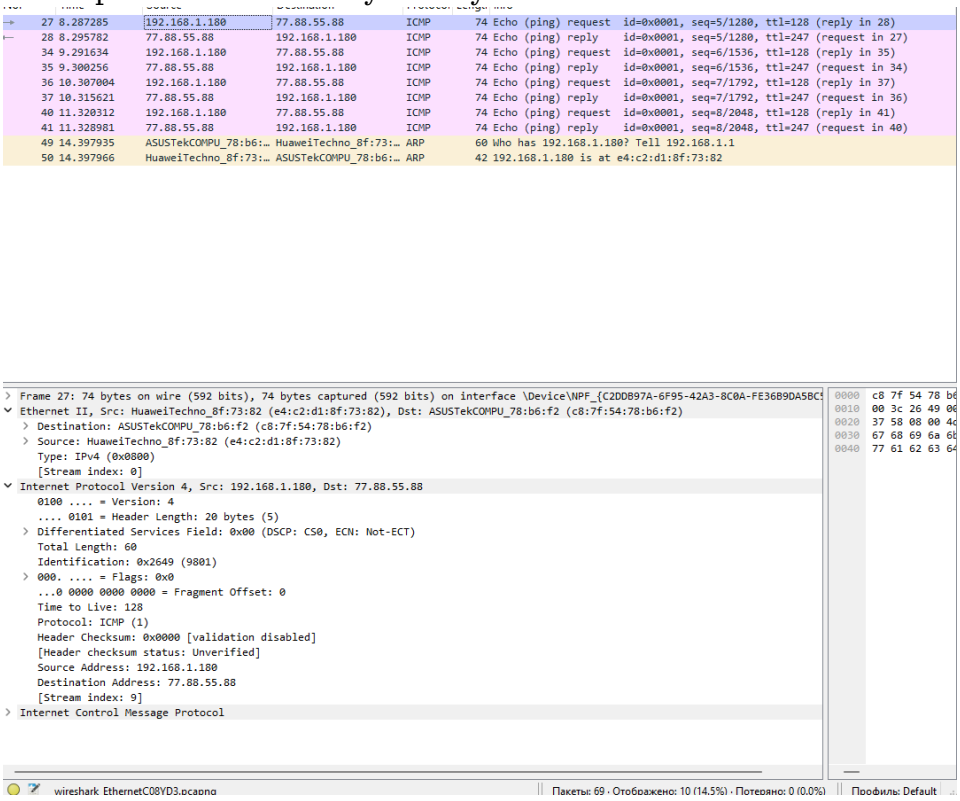
Рис. 2.5: Анализ ARP-запроса

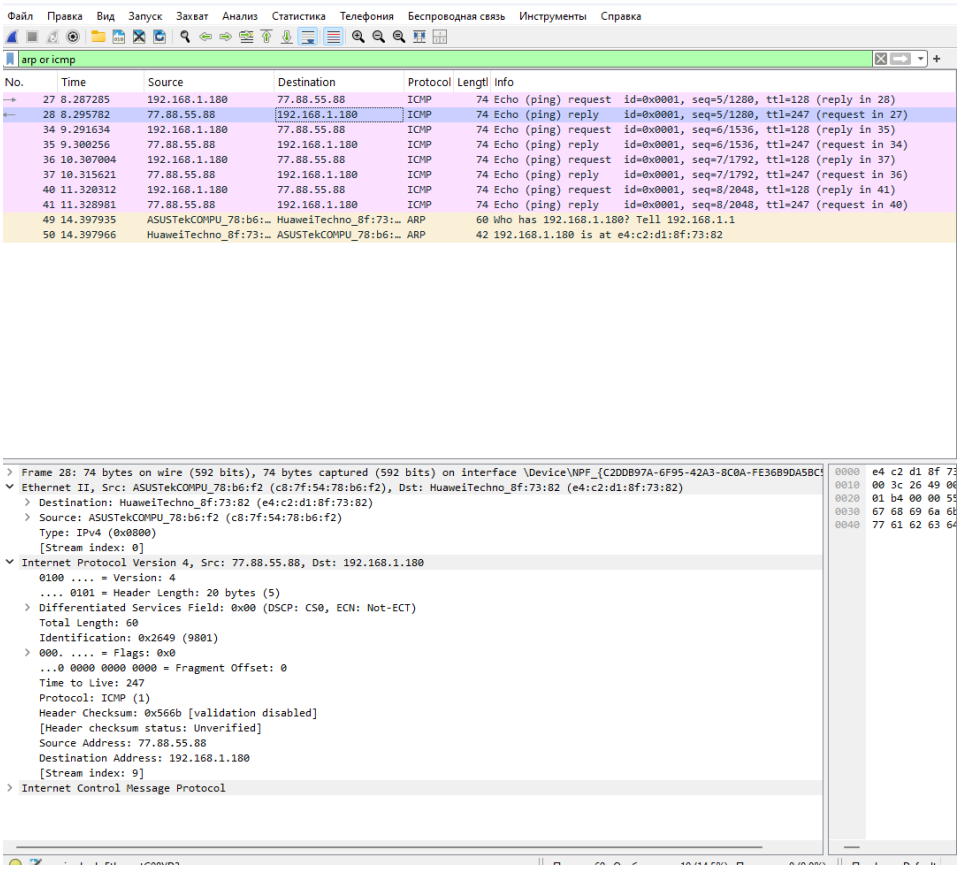
Затем начат новый процесс захвата трафика. В консоли выполнена команда `ping yandex.ru`, что позволило исследовать прохождение пакетов к удалённому узлу в сети Интернет. Все запросы успешно завершились с временем отклика около **8 мс**.

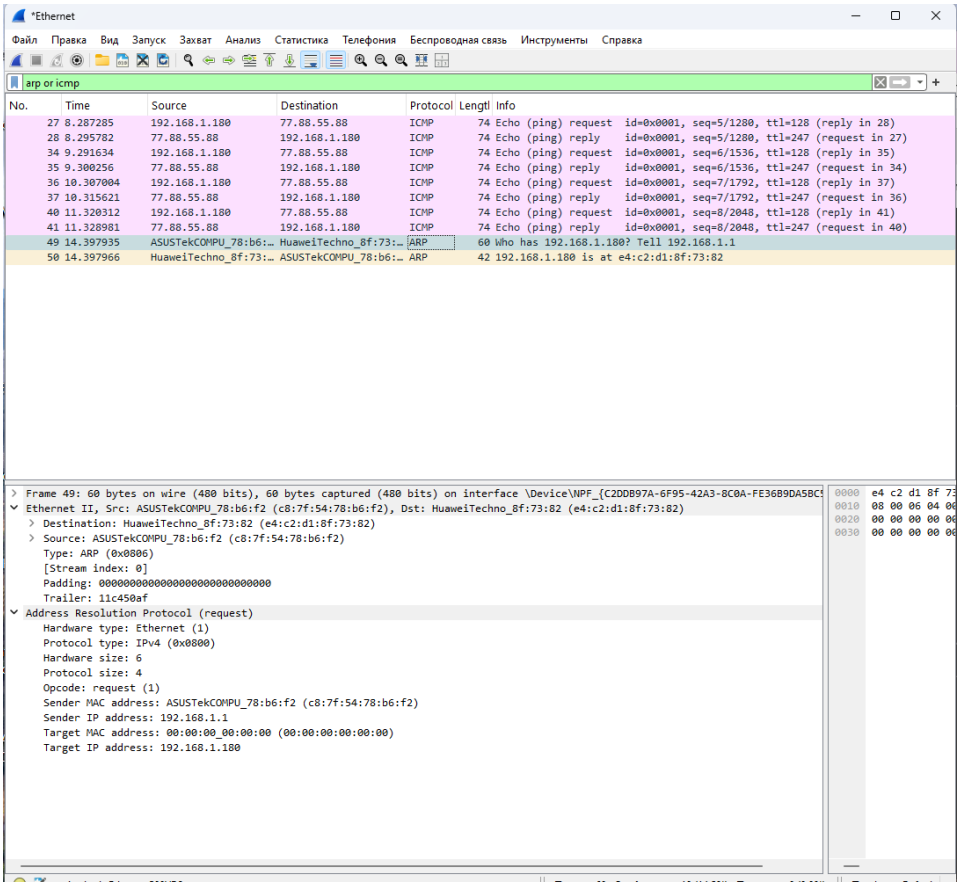
```
PS C:\Users\gaforov\Documents>
PS C:\Users\gaforov\Documents> ping yandex.ru

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
Ответ от 77.88.55.88: число байт=32 время=8мс TTL=247
Ответ от 77.88.55.88: число байт=32 время=8мс TTL=247
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% потерь)
Приблизительное время приема-передачи в мс:
    Минимальное = 8мсек, Максимальное = 8 мсек, Среднее = 8 мсек
PS C:\Users\gaforov\Documents>
```

Рис. 2.6: Проверка доступности внешнего ресурса

Применение фильтра `arp or icmp` показало обмен ICMP Echo Request и Echo Reply между локальным адресом **192.168.1.180** и удалённым сервером **77.88.55.88** (yandex.ru). На этапе разрешения адресов также фиксировались ARP-запросы к локальному шлюзу.




Файл Правка Вид Запуск Захват Анализ Статистика Телефония Беспроводная связь Инструменты Справка
arp or icmp
No. Time Source Destination Protocol Lengtl Info
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 28: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface \Device\NPF_{C2DDB97A-6F95-42A3-8C0A-FE36B9DA5BC
Ethernet II, Src: ASUSTekCOMPU_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: ASUSTekCOMPU_78:b6:f2 (c8:7f:54:78:b6:f2)
Type: IPv4 (0x0800)
[Stream index: 0]
Internet Protocol Version 4, Src: 77.88.55.88, Dst: 192.168.1.180
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: 247
Protocol: ICMP (1)
Header Checksum: 0x566b [validation disabled]
[Header checksum status: Unverified]
Source Address: 77.88.55.88
Destination Address: 192.168.1.180
[Stream index: 9]
Internet Control Message Protocol
0000 e4 c2 d1 8f 73
0010 00 3c 26 49 00
0020 01 b4 00 00 55
0030 67 68 69 6a 6b
0040 77 61 62 63 64



Для анализа транспортных протоколов был запущен анализатор **Wireshark**. В качестве активного сетевого интерфейса выбран Ethernet. После начала захвата трафика в браузере был открыт сайт **CERN** по адресу `http://info.cern.ch/`.

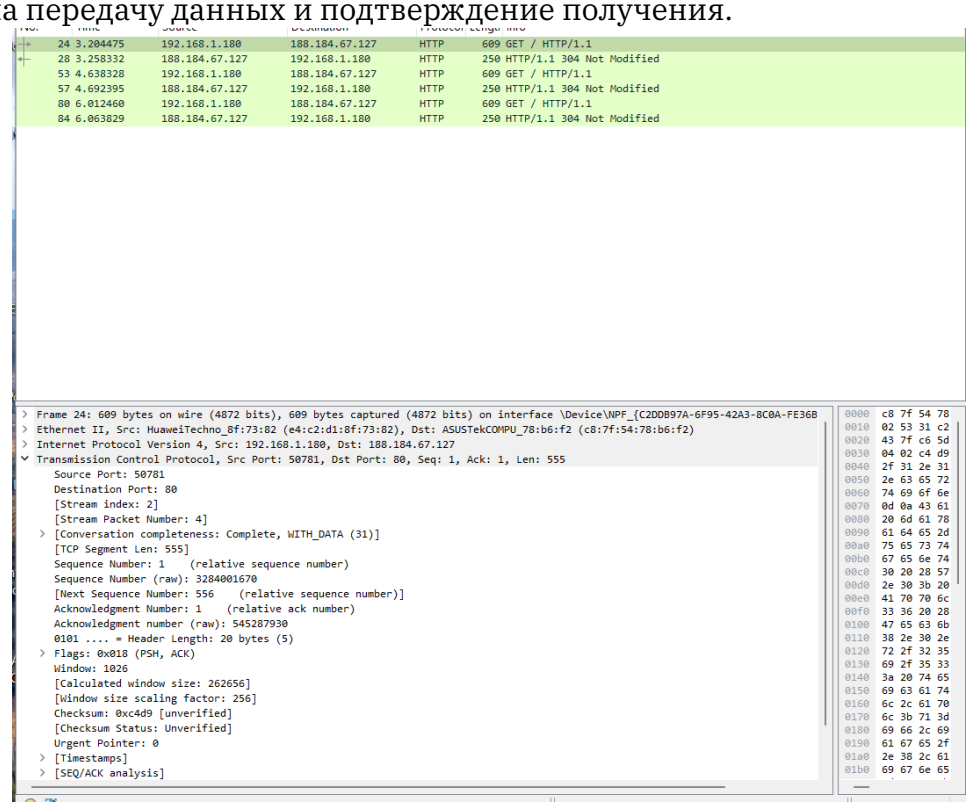
В ходе анализа пакетов использовались фильтры `http`, `dns` и `quic`, что позволило последовательно изучить работу протоколов **TCP**, **UDP** и **QUIC** на транспортном уровне.

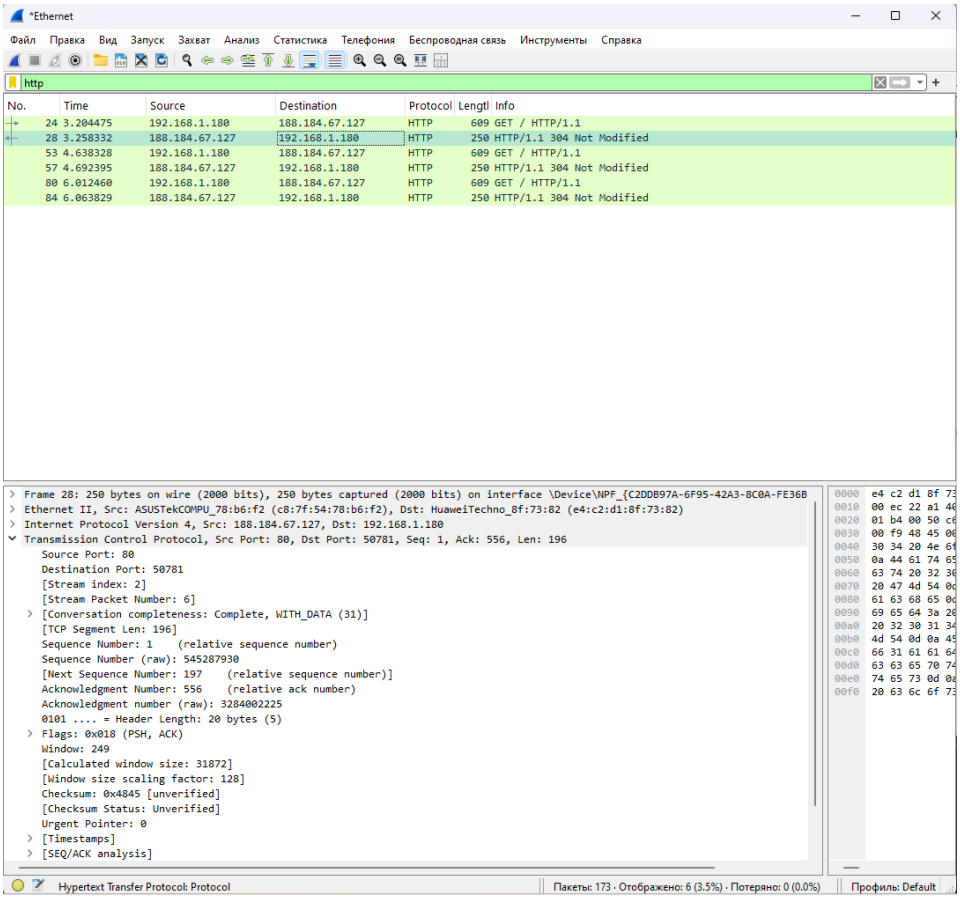
2.1 Анализ HTTP и протокола TCP

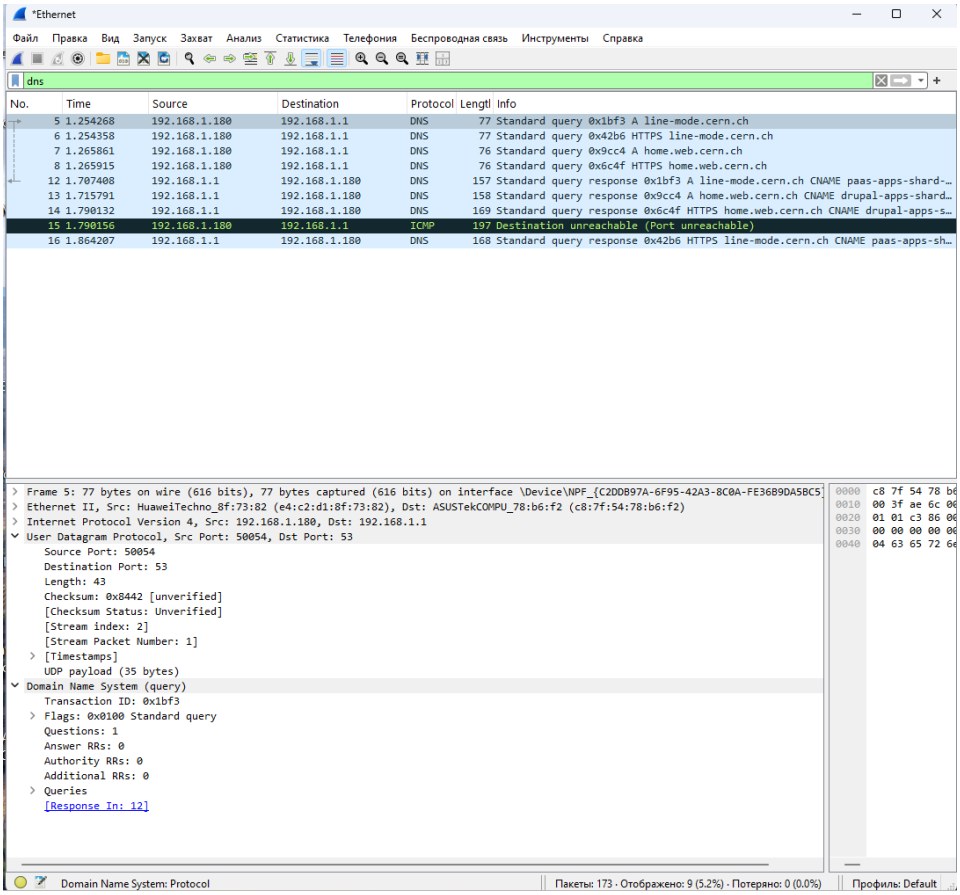
При фильтрации трафика по протоколу `http` в Wireshark были зафиксированы запросы **GET** и ответы **HTTP/1.1 304 Not Modified**. Каждый HTTP-пакет передаётся в рамках соединения **TCP**, обеспечивающего надёжную доставку данных.

В заголовках пакетов TCP отображались порты источника и назначения (например, **50781 → 80**), что подтверждает стандартное использование порта 80

для HTTP. Также присутствовали управляющие флаги PSH и ACK, указывающие на передачу данных и подтверждение получения.



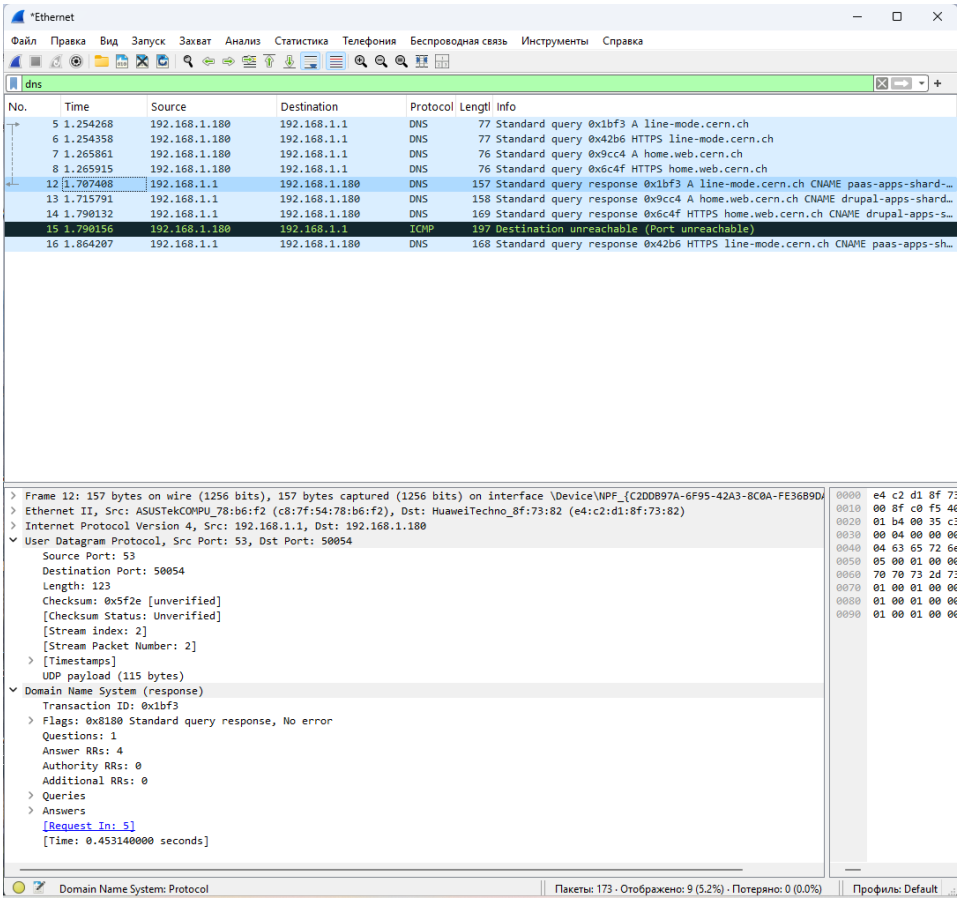

*Ethernet
Файл Правка Вид Запуск Захват Анализ Статистика Телефония Беспроводная связь Инструменты Справка
http
No. Time Source Destination Protocol Lengtl Info
24 3.204475 192.168.1.180 188.184.67.127 HTTP 609 GET / HTTP/1.1
28 3.258332 188.184.67.127 192.168.1.180 HTTP 250 HTTP/1.1 304 Not Modified
53 4.638328 192.168.1.180 188.184.67.127 HTTP 609 GET / HTTP/1.1
57 4.692395 188.184.67.127 192.168.1.180 HTTP 250 HTTP/1.1 304 Not Modified
80 6.012460 192.168.1.180 188.184.67.127 HTTP 609 GET / HTTP/1.1
84 6.063829 188.184.67.127 192.168.1.180 HTTP 250 HTTP/1.1 304 Not Modified
Frame 28: 250 bytes on wire (2000 bits), 250 bytes captured (2000 bits) on interface \Device\NPF_{C2DDB97A-6F95-42A3-8C0A-FE36B
Ethernet II, Src: ASUSTekCOMPU_78:b6:f2 (c8:7f:54:78:b6: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
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): 545287930
[Next Sequence Number: 197 (relative sequence number)]
Acknowledgment Number: 556 (relative ack number)
Acknowledgment number (raw): 3284002225
0101 .... = Header Length: 20 bytes (5)
Flags: 0x018 (PSH, ACK)
Window: 249
[Calculated window size: 31872]
[Window size scaling factor: 128]
Checksum: 0x4845 [unverified]
[Checksum Status: Unverified]
Urgent Pointer: 0
[Timestamps]
[SEQ/ACK analysis]
0000 e4 c2 d1 8f 73
0010 00 ec 22 a1 40
0020 01 b4 00 50 c6
0030 00 f9 48 45 00
0040 30 34 20 4e 6f
0050 0a 44 61 74 65
0060 63 74 20 32 30
0070 20 47 4d 54 0d
0080 61 63 68 65 0d
0090 69 65 64 3a 20
00a0 20 32 30 31 34
00b0 4d 54 0d 0a 45
00c0 66 31 61 61 64
00d0 63 63 65 70 74
00e0 74 65 73 0d 0a
00f0 20 63 6c 6f 73
Hypertext Transfer Protocol: Protocol
Пакеты: 173 · Отображено: 6 (3.5%) · Потеряно: 0 (0.0%)
Профиль: Default

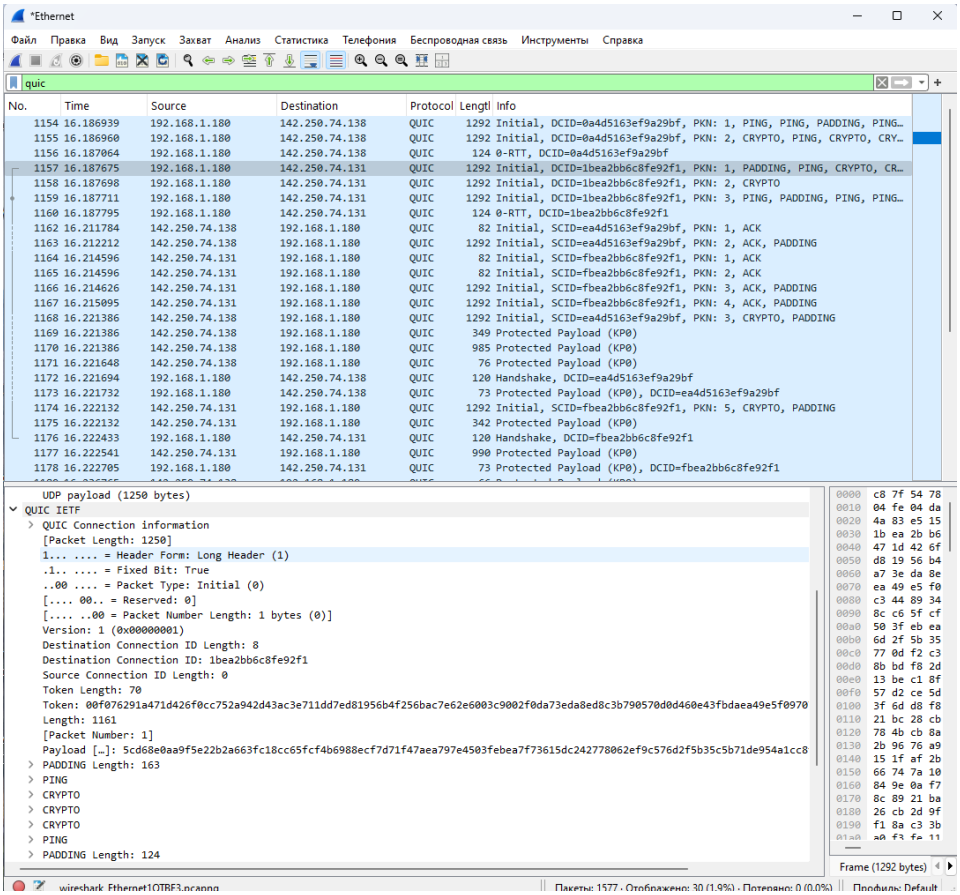


2.2 Анализ DNS и протокола UDP

Для анализа разрешения имён применён фильтр `dns`. В Wireshark зафиксированы стандартные **DNS-запросы** к адресу шлюза **192.168.1.1** и **DNS-ответы**, содержащие сведения о найденных доменах.

Передача данных происходила по протоколу **UDP**, что подтверждается указанием портов **53** (сервер DNS) и **50854** (клиент). В пакете запроса содержалось поле `Standard query`, а в ответе — `Standard query response, No error`, что свидетельствует об успешном разрешении имени.

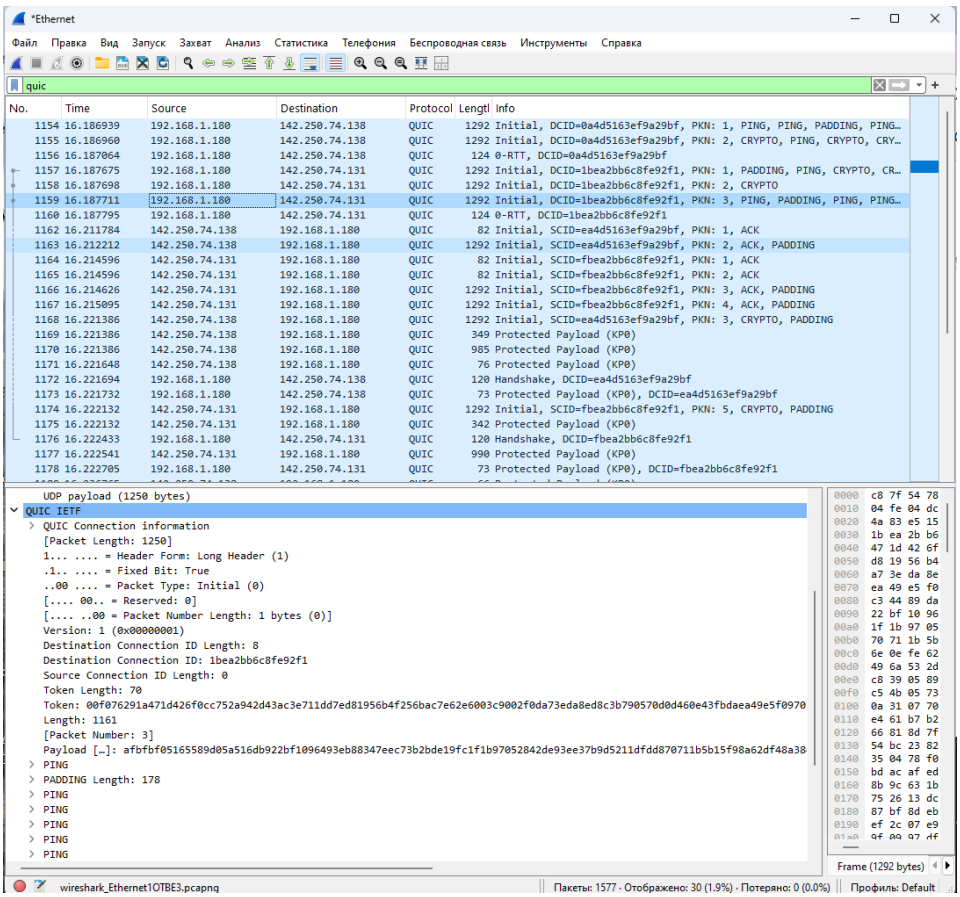

*Ethernet
Файл Правка Вид Запуск Захват Анализ Статистика Телефония Беспроводная связь Инструменты Справка
dns
No. Time Source Destination Protocol Lengtl Info
5 1.254268 192.168.1.180 192.168.1.1 DNS 77 Standard query 0x1bf3 A line-mode.cern.ch
6 1.254358 192.168.1.180 192.168.1.1 DNS 77 Standard query 0x42b6 HTTPS line-mode.cern.ch
7 1.265861 192.168.1.180 192.168.1.1 DNS 76 Standard query 0x9cc4 A home.web.cern.ch
8 1.265915 192.168.1.180 192.168.1.1 DNS 76 Standard query 0x6c4f HTTPS home.web.cern.ch
12 1.707408 192.168.1.1 192.168.1.180 DNS 157 Standard query response 0x1bf3 A line-mode.cern.ch CNAME paas-apps-shard-…
13 1.715791 192.168.1.1 192.168.1.180 DNS 158 Standard query response 0x9cc4 A home.web.cern.ch CNAME drupal-apps-shard…
14 1.790132 192.168.1.1 192.168.1.180 DNS 169 Standard query response 0x6c4f HTTPS home.web.cern.ch CNAME drupal-apps-s…
15 1.790156 192.168.1.180 192.168.1.1 ICMP 197 Destination unreachable (Port unreachable)
16 1.864207 192.168.1.1 192.168.1.180 DNS 168 Standard query response 0x42b6 HTTPS line-mode.cern.ch CNAME paas-apps-sh…
Frame 12: 157 bytes on wire (1256 bits), 157 bytes captured (1256 bits) on interface \Device\NPF_{C2DDB97A-6F95-42A3-8C0A-FE36B9D
Ethernet II, Src: ASUSTekCOMPU_78:b6:f2 (c8:7f:54:78:b6:f2), Dst: HuaweiTechno_8f:73:82 (e4:c2:d1:8f:73:82)
Internet Protocol Version 4, Src: 192.168.1.1, Dst: 192.168.1.180
User Datagram Protocol, Src Port: 53, Dst Port: 50054
Source Port: 53
Destination Port: 50054
Length: 123
Checksum: 0x5f2e [unverified]
[Checksum Status: Unverified]
[Stream index: 2]
[Stream Packet Number: 2]
[Timestamps]
UDP payload (115 bytes)
Domain Name System (response)
Transaction ID: 0x1bf3
Flags: 0x8180 Standard query response, No error
Questions: 1
Answer RRs: 4
Authority RRs: 0
Additional RRs: 0
Queries
Answers
[Request In: 5]
[Time: 0.453140000 seconds]
0000 e4 c2 d1 8f 73
0010 00 8f c0 f5 40
0020 01 b4 00 35 c3
0030 00 04 00 00 00
0040 04 63 65 72 6e
0050 05 00 01 00 00
0060 70 70 73 2d 73
0070 01 00 01 00 00
0080 01 00 01 00 00
0090 01 00 01 00 00
Domain Name System: Protocol
Пакеты: 173 · Отображено: 9 (5.2%) · Потеряно: 0 (0.0%)
Профиль: Default

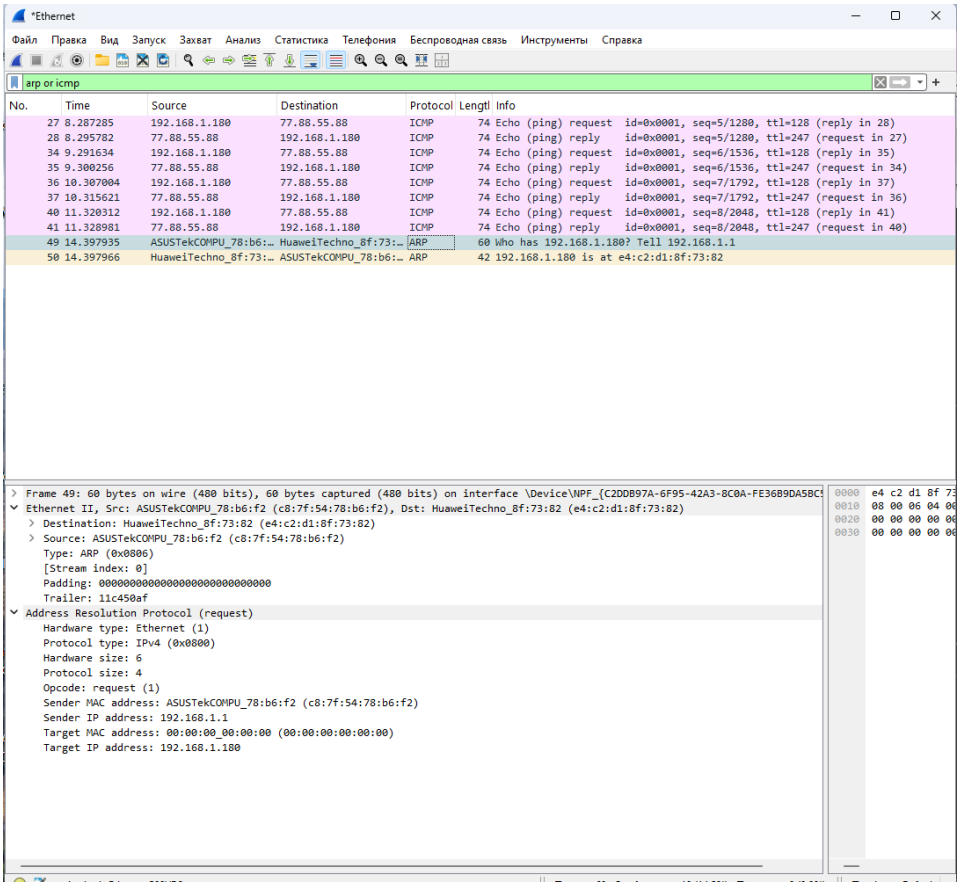


2.3 Анализ протокола QUIC

Для анализа современного транспортного протокола **QUIC** использовался фильтр `quic`. В захваченном трафике были зафиксированы пакеты типа **Initial**, **Handshake**, **PING** и **CRYPTO**, передаваемые по **UDP**.

Каждый пакет содержал заголовок **QUIC IETF**, в котором указаны параметры соединения: `Packet Length`, `Connection ID`, `Packet Type` и `Token`. В кадрах фиксировались элементы обмена ключами шифрования и подтверждения соединения, характерные для работы QUIC поверх UDP.


*Ethernet
Файл Правка Вид Запуск Захват Анализ Статистика Телефония Беспроводная связь Инструменты Справка
quic
No. Time Source Destination Protocol Lengtl Info
1154 16.186939 192.168.1.180 142.250.74.138 QUIC 1292 Initial, DCID=0a4d5163ef9a29bf, PKN: 1, PING, PING, PADDING, PING…
1155 16.186960 192.168.1.180 142.250.74.138 QUIC 1292 Initial, DCID=0a4d5163ef9a29bf, PKN: 2, CRYPTO, PING, CRYPTO, CRY…
1156 16.187064 192.168.1.180 142.250.74.138 QUIC 124 0-RTT, DCID=0a4d5163ef9a29bf
1157 16.187675 192.168.1.180 142.250.74.131 QUIC 1292 Initial, DCID=1bea2bb6c8fe92f1, PKN: 1, PADDING, PING, CRYPTO, CR…
1158 16.187698 192.168.1.180 142.250.74.131 QUIC 1292 Initial, DCID=1bea2bb6c8fe92f1, PKN: 2, CRYPTO
1159 16.187711 192.168.1.180 142.250.74.131 QUIC 1292 Initial, DCID=1bea2bb6c8fe92f1, PKN: 3, PING, PADDING, PING, PING…
1160 16.187795 192.168.1.180 142.250.74.131 QUIC 124 0-RTT, DCID=1bea2bb6c8fe92f1
1162 16.211784 142.250.74.138 192.168.1.180 QUIC 82 Initial, SCID=ea4d5163ef9a29bf, PKN: 1, ACK
1163 16.212212 142.250.74.138 192.168.1.180 QUIC 1292 Initial, SCID=ea4d5163ef9a29bf, PKN: 2, ACK, PADDING
1164 16.214596 142.250.74.131 192.168.1.180 QUIC 82 Initial, SCID=fbea2bb6c8fe92f1, PKN: 1, ACK
1165 16.214596 142.250.74.131 192.168.1.180 QUIC 82 Initial, SCID=fbea2bb6c8fe92f1, PKN: 2, ACK
1166 16.214626 142.250.74.131 192.168.1.180 QUIC 1292 Initial, SCID=fbea2bb6c8fe92f1, PKN: 3, ACK, PADDING
1167 16.215095 142.250.74.131 192.168.1.180 QUIC 1292 Initial, SCID=fbea2bb6c8fe92f1, PKN: 4, ACK, PADDING
1168 16.221386 142.250.74.138 192.168.1.180 QUIC 1292 Initial, SCID=ea4d5163ef9a29bf, PKN: 3, CRYPTO, PADDING
1169 16.221386 142.250.74.138 192.168.1.180 QUIC 349 Protected Payload (KP0)
1170 16.221386 142.250.74.138 192.168.1.180 QUIC 985 Protected Payload (KP0)
1171 16.221648 142.250.74.138 192.168.1.180 QUIC 76 Protected Payload (KP0)
1172 16.221694 192.168.1.180 142.250.74.138 QUIC 120 Handshake, DCID=ea4d5163ef9a29bf
1173 16.221732 192.168.1.180 142.250.74.138 QUIC 73 Protected Payload (KP0), DCID=ea4d5163ef9a29bf
1174 16.222132 142.250.74.131 192.168.1.180 QUIC 1292 Initial, SCID=fbea2bb6c8fe92f1, PKN: 5, CRYPTO, PADDING
1175 16.222132 142.250.74.131 192.168.1.180 QUIC 342 Protected Payload (KP0)
1176 16.222433 192.168.1.180 142.250.74.131 QUIC 120 Handshake, DCID=fbea2bb6c8fe92f1
1177 16.222541 142.250.74.131 192.168.1.180 QUIC 990 Protected Payload (KP0)
1178 16.222705 192.168.1.180 142.250.74.131 QUIC 73 Protected Payload (KP0), DCID=fbea2bb6c8fe92f1
UDP payload (1250 bytes)
QUIC IETF
QUIC Connection information
[Packet Length: 1250]
1... .... = Header Form: Long Header (1)
.1.. .... = Fixed Bit: True
..00 .... = Packet Type: Initial (0)
[.... 00.. = Reserved: 0]
[.... ..00 = Packet Number Length: 1 bytes (0)]
Version: 1 (0x00000001)
Destination Connection ID Length: 8
Destination Connection ID: 1bea2bb6c8fe92f1
Source Connection ID Length: 0
Token Length: 70
Token: 00f076291a471d426f0cc752a942d43ac3e711dd7ed81956b4f256bac7e62e6003c9002f0da73eda8ed8c3b790570d0d460e43fbdaea49e5f0970
Length: 1161
[Packet Number: 3]
Payload […]: afbfbf05165589d05a516db922bf1096493eb88347eec73b2bde19fc1f1b97052842de93ee37b9d5211dfdd870711b5b15f98a62df48a38
PING
PADDING Length: 178
PING
PING
PING
PING
PING
0000 c8 7f 54 78
0010 04 fe 04 dc
0020 4a 83 e5 15
0030 1b ea 2b b6
0040 47 1d 42 6f
0050 d8 19 56 b4
0060 a7 3e da 8e
0070 ea 49 e5 f0
0080 c3 44 89 da
0090 22 bf 10 96
00a0 1f 1b 97 05
00b0 70 71 1b 5b
00c0 6e 0e fe 62
00d0 49 6a 53 2d
00e0 c8 39 05 89
00f0 c5 4b 05 73
0100 0a 31 07 70
0110 e4 61 b7 b2
0120 66 81 8d 7f
0130 54 bc 23 82
0140 35 04 78 f0
0150 bd ac af ed
0160 8b 9c 63 1b
0170 75 26 13 dc
0180 87 bf 8d eb
0190 ef 2c 07 e9
Frame (1292 bytes)
wireshark_Ethernet1OTBE3.pcapng
Пакеты: 1577 · Отображено: 30 (1.9%) · Потеряно: 0 (0.0%)
Профиль: Default



2.4 Анализ handshake протокола TCP в Wireshark

Для анализа механизма установления соединения **TCP (handshake)** был запущен анализатор трафика **Wireshark**. В качестве активного сетевого интерфейса выбран Ethernet, после чего начат захват сетевых пакетов. Для генерации TCP-трафика использовалось подключение по протоколу **HTTP** к сайту **188.184.67.127 (info.cern.ch)**.

В захваченном трафике в Wireshark применён фильтр `tcp`, что позволило выделить пакеты, относящиеся к установлению соединения. На экране отчётливо видна последовательность **трёхэтапного рукопожатия (Three-way handshake)**:

1. **SYN** — клиент (192.168.1.180) инициирует соединение, отправляя сегмент с установленным флагом `SYN` и указанием начального номера последова-

тельности (`Seq=0`, `Win=64240`).

2. **SYN, ACK** — сервер (188.184.67.127) подтверждает запрос, устанавливая флаги `SYN` и `ACK`, а также возвращая свой номер последовательности (`Seq=0`, `Ack=1`).

3. **ACK** — клиент отправляет подтверждение (`Ack=1`), завершая установление соединения.

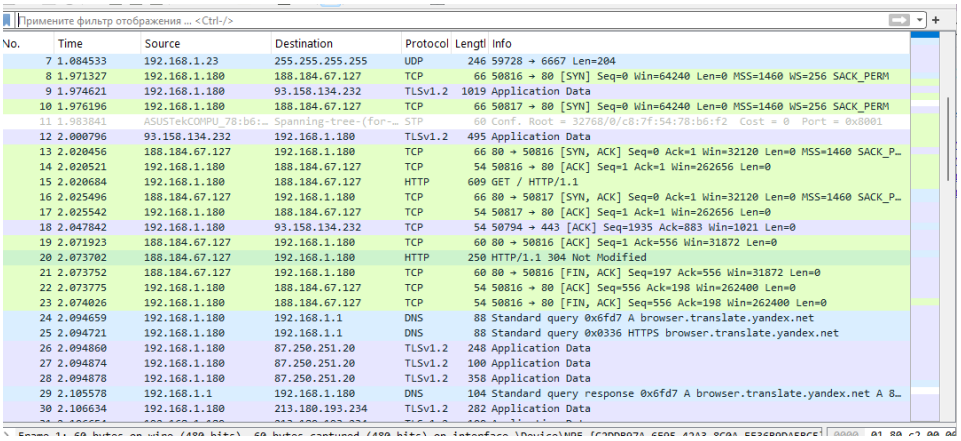


Рис. 2.7: Захват TCP handshake в Wireshark

После завершения рукопожатия начинается передача данных по протоколу **HTTP**. На кадрах видно, что клиент отправляет запрос `GET / HTTP/1.1`, а сервер отвечает сообщением `HTTP/1.1 304 Not Modified`. Это подтверждает, что соединение установлено корректно и используется для передачи прикладных данных.

Дальнейший анализ статистики TCP, выполненный через меню **«Статистика → График потока (Flow Graph)»**, демонстрирует временную диаграмму с обменом сегментами:

- первые три пакета (SYN → SYN/ACK → ACK) образуют фазу установления соединения,
- далее следуют сегменты передачи данных и завершения соединения (флаги `FIN, ACK`).

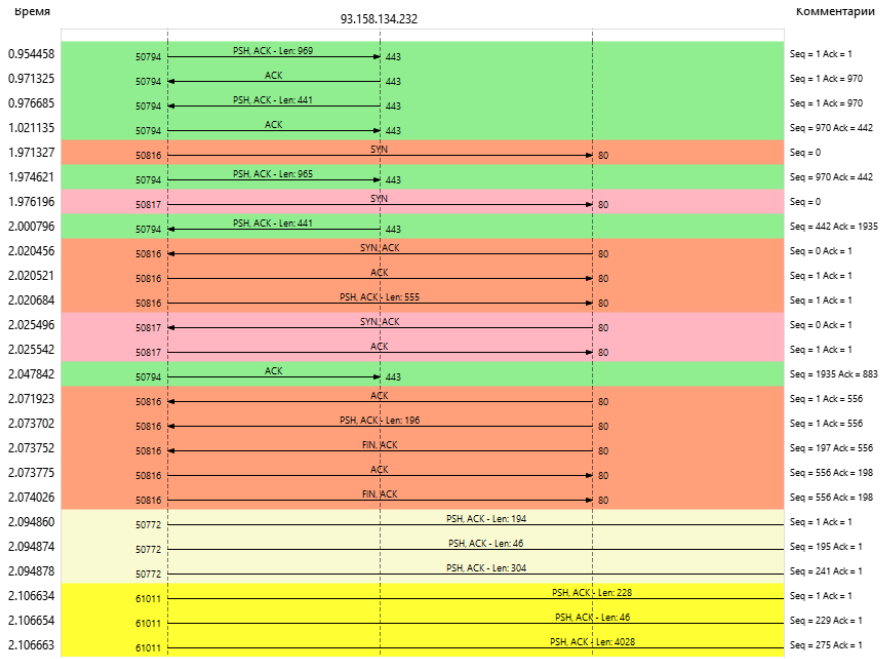


Рис. 2.8: График потока

3 Вывод

В ходе работы был проанализирован процесс установления соединения по протоколу **TCP** с использованием анализатора трафика **Wireshark**. Исследование показало, что трёхэтапное рукопожатие (**Three-Way Handshake**) обеспечивает надёжное установление соединения между клиентом и сервером перед началом передачи данных. В результате анализа зафиксированы стадии обмена сегментами **SYN**, **SYN/ACK** и **ACK**, подтверждающие успешную синхронизацию номеров последовательности и готовность сторон к коммуникации. Полученные данные подтверждают корректную реализацию транспортного уровня в модели OSI и демонстрируют принцип работы TCP при передаче HTTP-запросов.