

**РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ**

**Факультет физико-математических и естественных наук**

дисциплина: Сетевые технологии

Студент: Талебу Тенке франк Устон

Группа: НФИбд-02-23

**МОСКВА**

2025 г.

## Цель работы

Изучить посредством Wireshark кадры Ethernet, проанализировать PDU протоколы транспортного и прикладного уровней стека TCP/IP.

## Задание

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

## **Выполнение лабораторной работы**

№1

С помощью команды ipconfig для ОС типа Windows выведем информацию о текущем сетевом соединении. маску подсети (используется для определения сетевой и узловой частей IPv4-адреса) и шлюз.

*Рисунок 1. Команда ipconfig*

```

Адаптер беспроводной локальной сети Беспроводная сеть:

DNS-суффикс подключения . . . . . : 
Описание. . . . . : Realtek 8822CE Wireless LAN 802.11ac PCI-E NIC
Физический адрес. . . . . : E0-75-26-2E-93-5C
DHCP включен. . . . . : Да
Автонастройка включена. . . . . : Да
Локальный IPv6-адрес канала . . . . . : fe80::f505:9056:79e9:894%20(Основной)
IPv4-адрес. . . . . : 172.16.74.82(Основной)
Маска подсети . . . . . : 255.255.254.0
Аренда получена. . . . . : 11 октября 2025 г. 18:52:39
Срок аренды истекает. . . . . : 11 октября 2025 г. 19:52:39
Основной шлюз. . . . . : 172.16.74.1
DHCP-сервер. . . . . : 192.168.80.59
IAID DHCPv6 . . . . . : 199259430
DUID клиента DHCPv6 . . . . . : 00-01-00-01-2A-FF-04-6A-54-EF-92-C5-46-04
DNS-серверы. . . . . : 37.18.92.5
                                         193.232.218.194
NetBIOS через TCP/IP. . . . . : Включен

Адаптер Ethernet Ethernet:

Состояние среды. . . . . : Среда передачи недоступна.
DNS-суффикс подключения . . . . . :
Описание. . . . . : Realtek PCIe GbE Family Controller
Физический адрес. . . . . : 54-EF-92-C5-46-04
DHCP включен. . . . . : Да
Автонастройка включена. . . . . : Да
PS C:\Users\Taleubou tenkeu>

```

Используем также опцию /all для вывода более подробной информации.

Определим МАС-адреса сетевых интерфейсов на моем компьютере.

У меня есть помимо основной беспроводной сети WI-FI еще две локальные сети (я так понимаю они виртуальные). МАС-адрес для первого виртуального адаптера. МАС-адрес состоит из 6 октетов: первые 3 октета идентифицируют производителя, последние 3 октета идентифицируют сетевой интерфейс. Нас интересуют последние два бита (нулевой и первый биты). У меня оба нули => мой адрес индивидуальный и глобально администрируемый.

```

Адаптер беспроводной локальной сети Подключение по локальной сети* 1:

Состояние среды. . . . . : Среда передачи недоступна.
DNS-суффикс подключения . . . . . :
Описание. . . . . : Microsoft Wi-Fi Direct Virtual
Физический адрес. . . . . : E2-75-26-2E-93-5C
DHCP включен. . . . . : Да
Автонастройка включена. . . . . : Да

```

Рисунок 3. MAC-адрес в случае виртуального адаптера 1

исунок 2. Команды ipconfig /all

```
Адаптер беспроводной локальной сети Подключение по локальной сети* 2:  
Состояние среды . . . . . : Среда передачи недоступна.  
DNS-суффикс подключения . . . . . :  
Описание . . . . . : Microsoft Wi-Fi Direct Virtual  
Физический адрес . . . . . : F2-75-26-2E-93-5C
```

MAC-адрес для второго виртуального адаптера Этот адрес является индивидуальным и локально администрируемым (поэтому по нему нельзя узнать производителя).

Рисунок 4. MAC-адрес в случае виртуального адаптера 2

```
Адаптер беспроводной локальной сети Беспроводная сеть:  
DNS-суффикс подключения . . . . . :  
Описание . . . . . : Realtek 8822CE Wireless LAN 802.11ac PCI-E NIC  
Физический адрес . . . . . : E0-75-26-2E-93-5C  
DHCP включен . . . . . : Да  
Автонастройка включена . . . . . : Да  
Локальный IPv6-адрес канала . . . . . : fe80::f505:9056:79e9:894%20(Основной)  
IPv4-адрес . . . . . : 172.16.74.82(Основной)
```

MAC-адрес для беспроводной сети WI-FI Этот адрес является индивидуальным и глобально администрируемым

Рисунок 5. MAC-адрес в случае WI-FI

## №2

Из предыдущего задания мы узнали адрес основного шлюза

```
PS C:\Users\Taleubou tenkeu> ping 172.16.74.82  
Обмен пакетами с 172.16.74.82 по с 32 байтами данных:  
Ответ от 172.16.74.82: число байт=32 время<1мс TTL=128  
  
Статистика Ping для 172.16.74.82:  
Пакетов: отправлено = 4, получено = 4, потеряно = 0  
(0% потеря)  
Приблизительное время приема-передачи в мс:
```

Теперь пропингуем его, предварительно запустив Wireshark и включив захват трафика. Посылаются 4 пакета, 4 пакета получено назад.

Рисунок 6. Пингование шлюза

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	172.16.74.82	172.16.74.82	ICMP	64	Echo (ping) request id=0x0001, seq=24/6144, ttl=128
2	0.000290	172.16.74.82	172.16.74.82	ICMP	64	Echo (ping) reply id=0x0001, seq=24/6144, ttl=128
3	4.173118	172.16.74.82	172.16.74.82	ICMP	64	Echo (ping) request id=0x0001, seq=25/6400, ttl=128
4	4.173211	172.16.74.82	172.16.74.82	ICMP	64	Echo (ping) reply id=0x0001, seq=25/6400, ttl=128
5	5.187267	172.16.74.82	172.16.74.82	ICMP	64	Echo (ping) request id=0x0001, seq=26/6656, ttl=128
6	5.187460	172.16.74.82	172.16.74.82	ICMP	64	Echo (ping) reply id=0x0001, seq=26/6656, ttl=128
7	6.203204	172.16.74.82	172.16.74.82	ICMP	64	Echo (ping) request id=0x0001, seq=27/6912, ttl=128
8	6.203550	172.16.74.82	172.16.74.82	ICMP	64	Echo (ping) reply id=0x0001, seq=27/6912, ttl=128
9	7.218028	172.16.74.82	172.16.74.82	ICMP	64	Echo (ping) request id=0x0001, seq=28/7168, ttl=128
10	7.218162	172.16.74.82	172.16.74.82	ICMP	64	Echo (ping) reply id=0x0001, seq=28/7168, ttl=128

В строке фильтра пропишем фильтр icmp. Убедимся, что в списке пакетов отобразятся только пакеты ICMP, в частности пакеты, которые были

Рисунок 7. Пакеты ICMP

сгенерированы с помощью команды ping, отправленной с моего устройства на шлюз по умолчанию.

Изучим эхо-запрос и эхо-ответ ICMP в программе Wireshark:

```

Frame 1: Packet, 64 bytes on wire (512 bits), 64 bytes captured (512 bits) on interface \Device\NPF_Loopback, id 0
  Section number: 1
  > Interface id: 0 (\Device\NPF_Loopback)
    Encapsulation type: NULL/Loopback (15)
    Arrival Time: Oct 11, 2025 20:23:48.675056000 RTZ 2 (зима)
    UTC Arrival Time: Oct 11, 2025 17:23:48.675056000 UTC
    Epoch Arrival Time: 1760203428.675056000
    [Time shift for this packet: 0.000000000 seconds]
    [Time since reference or first frame: 0.000000000 seconds]
  Frame Number: 1
  Frame Length: 64 bytes (512 bits)
  Capture Length: 64 bytes (512 bits)
  [Frame is marked: False]
  [Frame is ignored: False]
  [Protocols in frame: null:ip:icmp:data]
  Character encoding: ASCII (0)
  [Coloring Rule Name: ICMP]
  [Coloring Rule String: icmp || icmpv6]
  > Null/Loopback

```

На панели списка пакетов (верхний раздел) выберем первый указанный кадр ICMP — эхо-запрос. Изучим информацию на панели сведений о пакете в средней части экрана. На вкладке физического уровня можно найти длину кадра (64 бита), тип Ethernet – Ethernet (1).

Рисунок 8. Кадр физического уровня

Чтобы узнать MAC-адрес источника и шлюза перейдем на канальный

уровень. Адрес источника (Source, откуда запрос отправлен) – это адрес моего устройства Адрес шлюза (destination, то куда отправлен запрос) Тип адреса тут указан (показаны нулевые и первые биты MAC-адресов). Что адрес источника, что адрес шлюза индивидуальные и глобально администрируемые.

```

    ✓ Ethernet II, Src: Intel_6b:6a:cf (b0:60:88:6b:6a:cf), Dst: IPv4mcast_fb (01:00:5e:00:00:fb)
      ✓ Destination: IPv4mcast_fb (01:00:5e:00:00:fb)
        .... ..0. .... .... .... = LG bit: Globally unique address (factory default)
        .... ..1. .... .... .... = IG bit: Group address (multicast/broadcast)
      ✓ Source: Intel_6b:6a:cf (b0:60:88:6b:6a:cf)
        .... ..0. .... .... .... = LG bit: Globally unique address (factory default)
        .... ..0. .... .... .... = IG bit: Individual address (unicast)
      Type: IPv4 (0x0800)
      [Stream index: 0]
    > Internet Protocol Version 4, Src: 172.16.74.155, Dst: 224.0.0.251
  
```

Рисунок 9. Кадр канального уровня

Далее посмотрим на полученный ответ. Тут все почти то же самое, что и в запросе (длина кадра 64 бита). Только теперь MAC-адрес источника – MAC-адрес шлюза а адрес назначения – адрес моего устройства

No.	Time	Source	Destination	Protocol	Length	Info
9327	574.927034	172.16.74.82	224.0.0.251	MDNS	93	Standard query 0x0000 ANY DESKTOP-99UCHU9._dosvc._tcp.local, "QM" question
9328	574.935874	fe80::f505:9056:79e.. ff02::fb		MDNS	113	Standard query 0x0000 ANY DESKTOP-99UCHU9._dosvc._tcp.local, "QM" question
9329	574.963917	172.16.74.155	172.16.74.82	MS-DO	129	Handshake Message (Reply)
9330	574.964398	172.16.74.82	172.16.74.155	MS-DO	69	BitField Message (has 15 of 80 pieces)
9331	574.964650	172.16.74.155	172.16.74.82	TCP	60	7680 → 51801 [FIN, ACK] Seq=1 Ack=76 Win=65280 Len=0
9332	574.964700	172.16.74.82	172.16.74.155	TCP	54	51801 → 7680 [ACK] Seq=76 Ack=2 Win=65280 Len=0
9333	574.964830	172.16.74.82	172.16.74.155	TCP	54	51801 → 7680 [FIN, ACK] Seq=76 Ack=2 Win=65280 Len=0
9334	575.005407	172.16.74.155	172.16.74.82	TCP	60	7680 → 51801 [ACK] Seq=2 Ack=77 Win=65280 Len=0
9335	575.005407	172.16.74.155	172.16.74.82	MS-DO	69	BitField Message (has 14 of 80 pieces)
9336	575.005407	172.16.74.155	172.16.74.82	TCP	60	7680 → 51800 [FIN, ACK] Seq=91 Ack=91 Win=65280 Len=0
9337	575.005508	172.16.74.82	172.16.74.155	TCP	54	51800 → 7680 [ACK] Seq=91 Ack=92 Win=65280 Len=0
9338	575.006003	172.16.74.82	172.16.74.155	TCP	54	51800 → 7680 [FIN, ACK] Seq=91 Ack=92 Win=65280 Len=0
9339	575.015045	172.16.74.155	172.16.74.82	TCP	60	7680 → 51800 [ACK] Seq=92 Ack=92 Win=65280 Len=0
9340	575.048923	fe80::1c05:bd94:51e.. ff02::16		ICMPv6	90	Multicast Listener Report Message v2
9341	575.112904	172.16.74.82	173.194.220.188	TCP	55	[TCP Keep-Alive] 51915 → 5228 [ACK] Seq=1 Ack=1 Win=255 Len=1
9342	575.130427	173.194.220.188	172.16.74.82	TCP	66	[TCP Keep-Alive ACK] 5228 → 51915 [ACK] Seq=1 Ack=2 Win=1047 Len=0 SLE=1 SRE=2

```

    ✓ Ethernet II, Src: Intel_6b:6a:cf (b0:60:88:6b:6a:cf), Dst: IPv4mcast_fb (01:00:5e:00:00:fb)
      ✓ Destination: IPv4mcast_fb (01:00:5e:00:00:fb)
        .... ..0. .... .... .... = LG bit: Globally unique address (factory default)
        .... ..1. .... .... .... = IG bit: Group address (multicast/broadcast)
      ✓ Source: Intel_6b:6a:cf (b0:60:88:6b:6a:cf)
        .... ..0. .... .... .... = LG bit: Globally unique address (factory default)
        .... ..0. .... .... .... = IG bit: Individual address (unicast)
      Type: IPv4 (0x0800)
      [Stream index: 0]
    > Internet Protocol Version 4, Src: 172.16.74.155, Dst: 224.0.0.251
  
```

Беспроводная сеть: <live capture in progress>

Рисунок 10. Эхо-ответ

Изучим кадры данных протокола ARP. Hardware type – это адрес канального уровня (Ethernet (1)), Protocol type – сетевой уровень (протокол IPv4), далее указаны размеры MAC-адреса (6 байт) и размер IPv4-адреса (4

байта). Код запроса – 1.

```
PS C:\Users\Taleubou tenkeu> ping www.yandex.ru

Обмен пакетами с www.YANDEX.ru [5.255.255.77] с 32 байтами данных:
Ответ от 5.255.255.77: число байт=32 время=26мс TTL=248
Ответ от 5.255.255.77: число байт=32 время=8мс TTL=248
Ответ от 5.255.255.77: число байт=32 время=5мс TTL=248
Ответ от 5.255.255.77: число байт=32 время=5мс TTL=248

Статистика Ping для 5.255.255.77:
    Пакетов: отправлено = 4, получено = 4, потеряно = 0
              (0% потеря)
Приблизительное время приема-передачи в мс:
    Минимальное = 5мсек, Максимальное = 26 мсек, Среднее = 11 мсек
PS C:\Users\Taleubou tenkeu>
```

Изучим данные в полях заголовка Ethernet II.

Здесь указаны MAC-адреса источника и получателя. Получатель в нашем случае – широковещательный адрес (групповой и локально администрируемый). Источник – адрес нашего шлюза (индивидуальный и глобально администрируемый).

Рисунок 11. Протокол ARP

```
PS C:\Users\Taleubou tenkeu> ping www.yandex.ru

Обмен пакетами с www.YANDEX.ru [5.255.255.77] с 32 байтами данных:
Ответ от 5.255.255.77: число байт=32 время=5мс TTL=248
Ответ от 5.255.255.77: число байт=32 время=6мс TTL=248
Ответ от 5.255.255.77: число байт=32 время=5мс TTL=248
Ответ от 5.255.255.77: число байт=32 время=5мс TTL=248

Статистика Ping для 5.255.255.77:
    Пакетов: отправлено = 4, получено = 4, потеряно = 0
              (0% потеря)
Приблизительное время приема-передачи в мс:
    Минимальное = 5мсек, Максимальное = 6 мсек, Среднее = 5 мсек
PS C:\Users\Taleubou tenkeu>
```

Начнем новый процесс захвата трафика в Wireshark. Пропингуем сайт яндекса.

Изучим запрос протокола ICMP. Адрес источника (Source, откуда запрос отправлен) – это адрес моего устройства Адрес получателя (destination, то куда

Рисунок 12. Пингование сайта www.yandex.ru

отправлен запрос) Что адрес источника, что адрес шлюза индивидуальные и глобально администрируемые.

Рисунок 13. Запрос протокола ICMP

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	172.16.74.82	172.16.74.82	ICMP	64	Echo (ping) request id=0x0001, seq=24/6144, ttl=128
2	0.000290	172.16.74.82	172.16.74.82	ICMP	64	Echo (ping) reply id=0x0001, seq=24/6144, ttl=128
3	4.173118	172.16.74.82	172.16.74.82	ICMP	64	Echo (ping) request id=0x0001, seq=25/6400, ttl=128
4	4.173211	172.16.74.82	172.16.74.82	ICMP	64	Echo (ping) reply id=0x0001, seq=25/6400, ttl=128
5	5.187267	172.16.74.82	172.16.74.82	ICMP	64	Echo (ping) request id=0x0001, seq=26/6656, ttl=128
6	5.187460	172.16.74.82	172.16.74.82	ICMP	64	Echo (ping) reply id=0x0001, seq=26/6656, ttl=128
7	6.203204	172.16.74.82	172.16.74.82	ICMP	64	Echo (ping) request id=0x0001, seq=27/6912, ttl=128
8	6.203550	172.16.74.82	172.16.74.82	ICMP	64	Echo (ping) reply id=0x0001, seq=27/6912, ttl=128
9	7.218028	172.16.74.82	172.16.74.82	ICMP	64	Echo (ping) request id=0x0001, seq=28/7168, ttl=128
10	7.218162	172.16.74.82	172.16.74.82	ICMP	64	Echo (ping) reply id=0x0001, seq=28/7168, ttl=128

Изучим запрос протокола ICMP. Адрес источника (Source, то куда откуда

Рисунок 14. Ответ протокола ICMP

запрос отправлен) - Адрес получателя (Destination, то куда отправлен запрос) – это адрес моего устройства Что адрес источника, что адрес шлюза индивидуальные и глобально администрируемые.

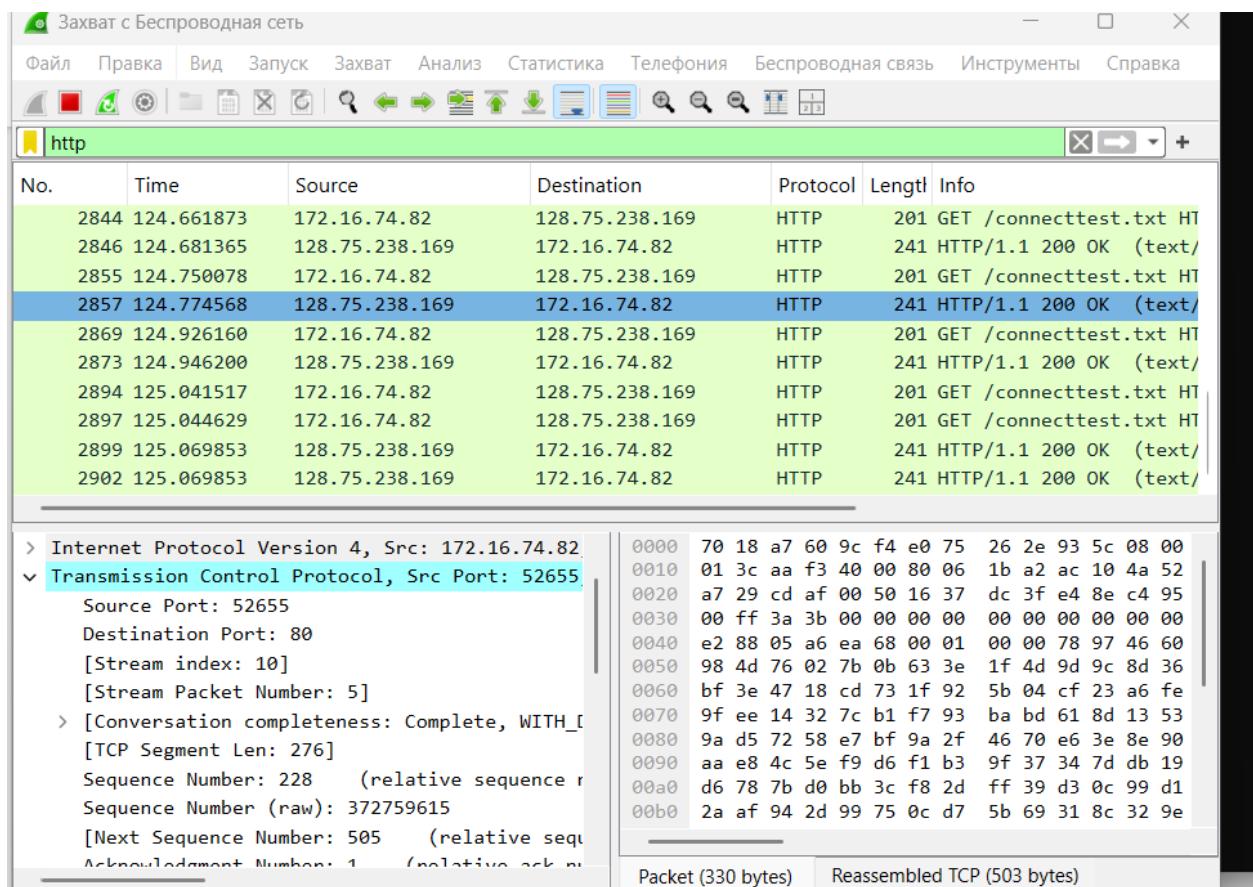
### №3

В браузере перейдем на сайт, работающий по протоколу HTTP (например, на сайт CERN <http://info.cern.ch/>). Для получения большей информации для Wireshark поперемещались по ссылкам или разделам сайта в браузере.

В Wireshark в строке фильтра укажем http и проанализируем информацию по протоколу TCP в случае запросов и ответов.

Открываем раздел протокола TCP в случае запроса. Видим, что порт получателя (это стандартный порт для http). Порт источника(он определяется случайным образом из незанятых и непrivилегированных портов). Также тут есть поле Порядковый номер (Sequence Number) и поле Номер подтверждения (Acknowledgment Number).

*Рисунок 15. Протокол TCP (случай запроса)*



Далее рассмотрим ответ. Здесь у нас поменялись местами порты источника и получателя.

Рисунок 16. Протокол TCP (случай ответа)

Теперь порт источника – порт сайта (80). А порт получателя (выбранный случайным образом).

В Wireshark в строке фильтра укажем dns и проанализируем информацию по протоколу UDP в случае запросов и ответов.

В случае запроса: порт источника (выбранный случайным образом из незанятых и непrivилегированных портов). Порт получателя

Wireshark screenshot showing a DNS request (No. 576) from 103.86.96.100 to 172.16.74.82. The packet details pane displays the DNS header and payload.

No.	Time	Source	Destination	Protocol	Length	Info
576	48.115650	103.86.96.100	172.16.74.82	DNS	236	Standard query response
577	48.116851	103.86.99.100	172.16.74.82	DNS	236	Standard query response
580	48.117796	103.86.99.100	172.16.74.82	DNS	236	Standard query response
583	48.141793	103.86.96.100	172.16.74.82	DNS	236	Standard query response
833	60.101840	172.16.74.82	37.18.92.5	DNS	79	Standard query 0x6ccb A
834	60.104481	37.18.92.5	172.16.74.82	DNS	367	Standard query response
892	63.916382	172.16.74.82	37.18.92.5	DNS	87	Standard query 0xc1aa A
893	63.920362	37.18.92.5	172.16.74.82	DNS	544	Standard query response
1954	118.819934	172.16.74.82	37.18.92.5	DNS	80	Standard query 0x417f A
1955	118.822082	37.18.92.5	172.16.74.82	DNS	365	Standard query response

[Stream index: 0]  
 > Internet Protocol Version 4, Src: 172.16.74.82.  
 < User Datagram Protocol, Src Port: 52641, Dst Port: 53  
 Source Port: 52641  
 Destination Port: 53  
 Length: 37  
 Checksum: 0xcecb [unverified]  
 [Checksum Status: Unverified]  
 [Stream index: 13]  
 [Stream Packet Number: 1]  
 > [Timestamps]  
 UDP payload (29 bytes)

Hex dump:  
 0000 70 18 a7 60 9c f4 e0 75 26 2e 93 5c 08 00 45  
 0010 00 39 e5 60 00 00 80 11 dd d9 ac 10 4a 52 25  
 0020 5c 05 cd a1 00 35 00 25 ce cb 21 16 01 00 00  
 0030 00 00 00 00 00 00 07 63 68 61 74 67 70 74 03  
 0040 6f 6d 00 00 01 00 01

Рисунок 17. Протокол UDP (случай запроса)

Wireshark screenshot showing a DNS response (No. 834) from 172.16.74.82 to 37.18.92.5. The packet details pane displays the DNS header and payload.

No.	Time	Source	Destination	Protocol	Length	Info
833	60.101840	172.16.74.82	37.18.92.5	DNS	79	Standard query 0x6ccb A
834	60.104481	37.18.92.5	172.16.74.82	DNS	367	Standard query response
892	63.916382	172.16.74.82	37.18.92.5	DNS	87	Standard query 0xc1aa A
893	63.920362	37.18.92.5	172.16.74.82	DNS	544	Standard query response
1954	118.819934	172.16.74.82	37.18.92.5	DNS	80	Standard query 0x417f A
1955	118.822082	37.18.92.5	172.16.74.82	DNS	365	Standard query response
2230	144.820710	172.16.74.82	37.18.92.5	DNS	86	Standard query 0x86e7 A
2231	144.823625	37.18.92.5	172.16.74.82	DNS	350	Standard query response
2335	148.846659	172.16.74.82	37.18.92.5	DNS	71	Standard query 0x1aa5 A
2336	148.851751	37.18.92.5	172.16.74.82	DNS	424	Standard query response

[Stream index: 0]  
 > Internet Protocol Version 4, Src: 172.16.74.82.  
 < User Datagram Protocol, Src Port: 52641, Dst Port: 53  
 Source Port: 52641  
 Destination Port: 53  
 Length: 37  
 Checksum: 0xcecb [unverified]  
 [Checksum Status: Unverified]  
 [Stream index: 13]  
 [Stream Packet Number: 1]  
 > [Timestamps]  
 UDP payload (29 bytes)

Hex dump:  
 0000 70 18 a7 60 9c f4 e0 75 26 2e 93 5c 08 00 45  
 0010 00 39 e5 60 00 00 80 11 dd d9 ac 10 4a 52 25  
 0020 5c 05 cd a1 00 35 00 25 ce cb 21 16 01 00 00  
 0030 00 00 00 00 00 00 07 63 68 61 74 67 70 74 03  
 0040 6f 6d 00 00 01 00 01

User Datagram Protocol (udp), 8 byte(s)

Рисунок 18. Протокол UDP (случай ответа)

В случае ответа порт источника а порт получателя.

В Wireshark в строке фильтра укажем `quic` и проанализируем информацию по протоколу `quic` в случае запросов и ответов.

Как и в случае `dns` можем посмотреть информацию транспортного уровня по протоколу `UDP`. Порт источника задан случайно, выбором из непривилегированных и незанятых портов, и равен, порт получателя равен - это стандартный порт `HTTPS`, то есть `quic` сразу криптуется.

Для создания альтернативы `TCP` поверх `UDP` строятся протоколы прикладного уровня `QUIC` IETF, которые управляют трафиком, управляют качеством обслуживания

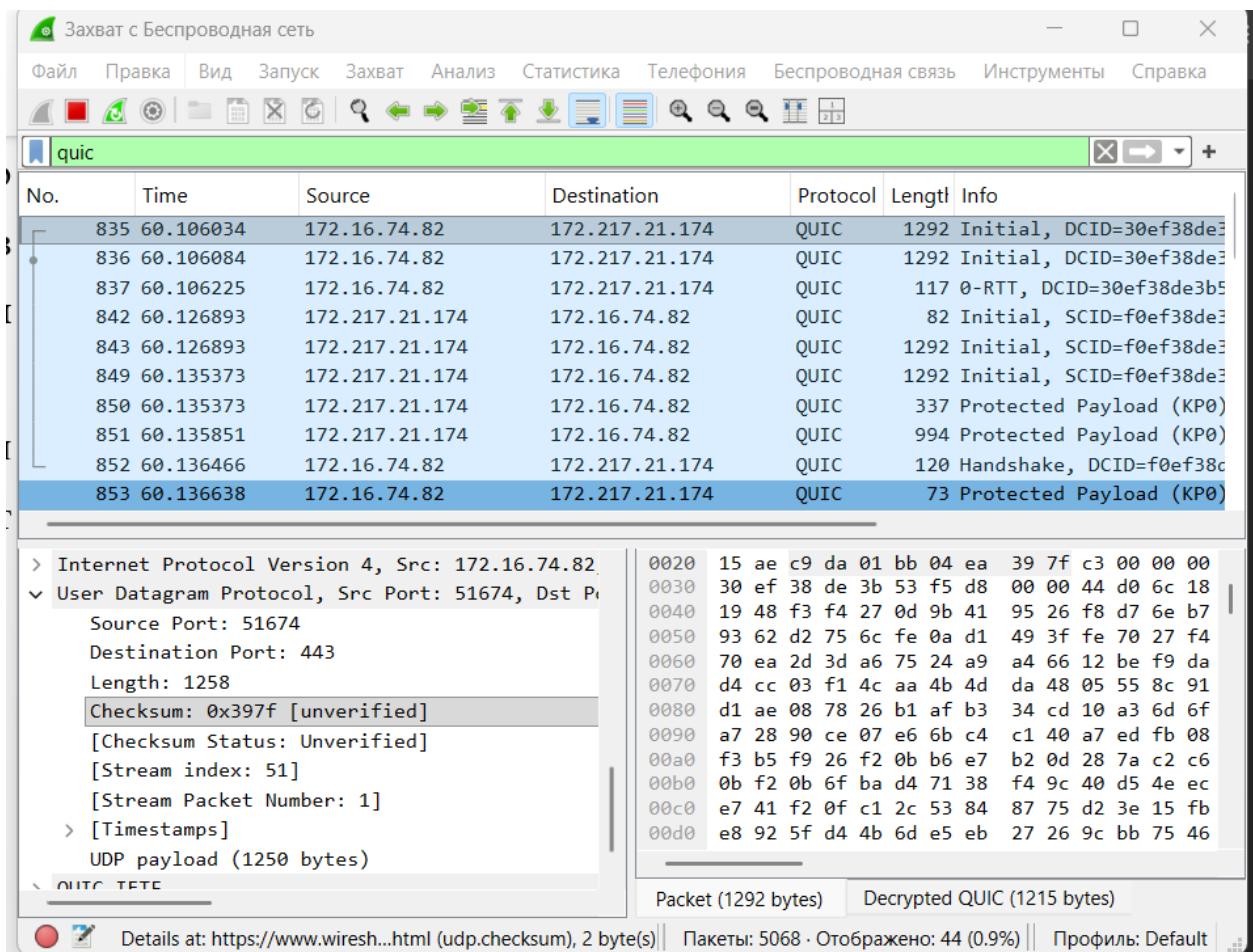


Рисунок 19. Запрос quic

quic

No.	Time	Source	Destination	Protocol	Length	Info
835	60.106034	172.16.74.82	172.217.21.174	QUIC	1292	Initial, DCID=30ef38de3b53f
836	60.106084	172.16.74.82	172.217.21.174	QUIC	1292	Initial, DCID=30ef38de3b53f
837	60.106225	172.16.74.82	172.217.21.174	QUIC	117	0-RTT, DCID=30ef38de3b53f5d
842	60.126893	172.217.21.174	172.16.74.82	QUIC	82	Initial, SCID=f0ef38de3b53f
843	60.126893	172.217.21.174	172.16.74.82	QUIC	1292	Initial, SCID=f0ef38de3b53f
849	60.135373	172.217.21.174	172.16.74.82	QUIC	1292	Initial, SCID=f0ef38de3b53f
850	60.135373	172.217.21.174	172.16.74.82	QUIC	337	Protected Payload (KP0)
851	60.135851	172.217.21.174	172.16.74.82	QUIC	994	Protected Payload (KP0)
852	60.136466	172.16.74.82	172.217.21.174	QUIC	120	Handshake, DCID=f0ef38de3b5
853	60.136638	172.16.74.82	172.217.21.174	QUIC	73	Protected Payload (KP0), DC
854	60.151787	172.217.21.174	172.16.74.82	QUIC	67	Protected Payload (KP0)
855	60.156722	172.217.21.174	172.16.74.82	QUIC	162	Protected Payload (KP0)
863	60.190810	172.16.74.82	172.217.21.174	QUIC	74	Protected Payload (KP0), DC
2804	186.332985	172.16.74.82	142.250.74.132	QUIC	1292	Initial, DCID=93fa7dff2df1a
2805	186.333074	172.16.74.82	142.250.74.132	QUIC	1292	Initial, DCID=93fa7dff2df1a
2807	186.353403	142.250.74.132	172.16.74.82	QUIC	82	Initial, SCID=f3fa7dff2df1a
2809	186.351860	142.250.74.132	172.16.74.82	QUIC	1292	Initial, SCID=f3fa7dff2df1a

▼ QUIC IETF

- ▼ QUIC Connection information
  - [Connection Number: 0]
  - [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: 30ef38de3b53f5d8
  - Source Connection ID Length: 0
  - Token Length: 0
  - Length: 1232
  - [Packet Number: 1]
  - Payload [...]: 188c561948f3f4270d9b419526f8d76el
- > PING

Packet (1292 bytes) Decrypted QUIC (1215 bytes)

Specifies if this is an individual (unicast) address (eth.dst.ig), 1 bit(s) Пакеты: 10126 · Отображено: 44 (0.4%) || Профиль: Default

Рисунок 20. Ответ quic

В случае ответа порты заданы наоборот, то есть источник порт, получатель.

#### №4

в Wireshark пакетов TCP.

1	1703 94.989011	172.16.74.82	35.190.80.1	TCP	66 [TCP Retransmission]	62463
1	1706 95.238136	172.16.74.82	35.190.80.1	TCP	66 [TCP Retransmission]	63095
e	1707 95.298002	172.16.74.82	35.190.80.1	TCP	66 [TCP Retransmission]	52535
e	1753 98.994972	172.16.74.82	35.190.80.1	TCP	66 [TCP Retransmission]	62463
e	1754 99.244971	172.16.74.82	35.190.80.1	TCP	66 [TCP Retransmission]	63095
	1755 99.307061	172.16.74.82	35.190.80.1	TCP	66 [TCP Retransmission]	52535

Проанализируем handshake протокола TCP.

Рисунок 21. Handshake TCP

Режим активного доступа (Active Open). Клиент посыпает сообщение SYN, ISSa, т.е. в передаваемом сообщении установлен бит SYN (Synchronize Sequence Number), а в поле Порядковый номер (Sequence Number) — начальное 32-битное значение ISSa (Initial Sequence Number).

Значение Sequence Number равно (ISSa), значение Acknowledgment Number равно 0. Также видим, что установлен флаг SYN.

Рисунок 22. Протокол TCP для первой ступени handshake

```
✗ [Conversation completeness: Incomplete (28)]
    ..0. .... = RST: Absent
    ...1 .... = FIN: Present
    .... 1... = Data: Present
    .... .1.. = ACK: Present
    .... ..0. = SYN-ACK: Absent
    .... ...0 = SYN: Absent
    [Completeness Flags: ·FDA··]
[TCP Segment Len: 1335]
Sequence Number: 1575      (relative sequence number)
Sequence Number (raw): 3589699135
[Next Sequence Number: 2910      (relative sequence number)]
Acknowledgment Number: 1      (relative ack number)
Acknowledgment number (raw): 2088089850
0101 .... = Header Length: 20 bytes (5)
✗ Flags: 0x018 (PSH, ACK)
    000. .... .... = Reserved: Not set
    ...0 .... .... = Accurate ECN: Not set
    .... 0... .... = Congestion Window Reduced: Not set
    .... .0.. .... = ECN-Echo: Not set
    .... ..0. .... = Urgent: Not set
    .... ...1 .... = Acknowledgment: Set
    .... .... 1.... = Push: Set
    .... .... .0.. = Reset: Not set
    .... .... ..0. = Syn: Not set
    .... .... ...0 = Fin: Not set
    [TCP Flags: .....AP...]
Window: 255
[Calculated window size: 255]
[Window size scaling factor: -1 (unknown)]
Checksum: 0xa327 [unverified]
[Checksum Status: Unverified]
Urgent Pointer: 0
> [Timestamps]
> [SEQ/ACK analysis]
[Client Contiguous Streams: 1]
[Server Contiguous Streams: 1]
```

Режим пассивного доступа (Passive Open). Сервер откликается, посылая сообщение SYN, ACK, ISSb, ACK(ISSa+1), т.е. установлены биты SYN и ACK; в поле Порядковый номер (Sequence Number) хостом В устанавливается начальное значение счётчика — ISSb; поле Номер подтверждения (Acknowledgment Number) содержит значение ISSa, полученное в первом пакете от хоста А и увеличенное на единицу.

Sequence Number равен 20880889850 (начальное значение счётчика — ISSb). Установлены флаги SYN, ACK.

Рисунок 23. Протокол TCP для второй ступени handshake

```
[TCP Segment Len: 1335]
Sequence Number: 1575      (relative sequence number)
Sequence Number (raw): 3589699135
[Next Sequence Number: 2910      (relative sequence number)]
Acknowledgment Number: 1      (relative ack number)
Acknowledgment number (raw): 2088089850
0101 .... = Header Length: 20 bytes (5)
▼ Flags: 0x018 (PSH, ACK)
    000. .... .... = Reserved: Not set
    ...0 .... .... = Accurate ECN: Not set
    .... 0... .... = Congestion Window Reduced: Not set
    .... .0.. .... = ECN-Echo: Not set
    .... ..0. .... = Urgent: Not set
    .... ...1 .... = Acknowledgment: Set
    .... .... 1.... = Push: Set
    .... ..... 0... = Reset: Not set
    .... ..... 0.. = Syn: Not set
    .... ..... 0 = Fin: Not set
```

Завершение рукопожатия. Клиент отправляет подтверждение получения SYN

сегмента от сервера с идентификатором, равным ISN (сервера)+1: ACK, ISSa+1, ACK(ISSb+1). В этом пакете установлен битACK, поле Порядковый номер (Sequence Number) содержит ISSa+1, поле Номер подтверждения (Acknowledgment Number) содержит значение ISSb+1. Посылкой этого пакета заканчивается трёхступенчатый handshake, и TCP-соединение считается установленным.

```
✓ Transmission Control Protocol, Src Port: 55653, Dst Port: 443, Seq:  
    Source Port: 55653  
    Destination Port: 443  
    [Stream index: 3]  
    [Stream Packet Number: 8]  
    ✓ [Conversation completeness: Incomplete (28)]  
        ..0. .... = RST: Absent  
        ...1 .... = FIN: Present  
        .... 1... = Data: Present  
        .... .1.. = ACK: Present  
        .... ..0. = SYN-ACK: Absent  
        .... ...0 = SYN: Absent  
        [Completeness Flags: .FDA..]  
    [TCP Segment Len: 1335]  
    Sequence Number: 1575      (relative sequence number)  
    Sequence Number (raw): 3589699135  
    [Next Sequence Number: 2910      (relative sequence number)]  
    Acknowledgment Number: 1      (relative ack number)  
    Acknowledgment number (raw): 2088089850  
    0101 .... = Header Length: 20 bytes (5)
```

Рисунок 24. Протокол TCP для третьей ступени handshake

Далее посмотрим график потока. Здесь в принципе все то же самое, что мы уже разобрали, только на графике. Клиент посыпает запрос серверу (установлен бит SYN), Seq = 0. Далее сервер отвечает клиенту (установлены биты SYN, ACK), Seq = 0, Ack =1 (это относительные значения). Ну звршение рукопожатия: клиент отправляет серверу подтверждение получения SYN сегмента, Seq = 1, Ack = 1.

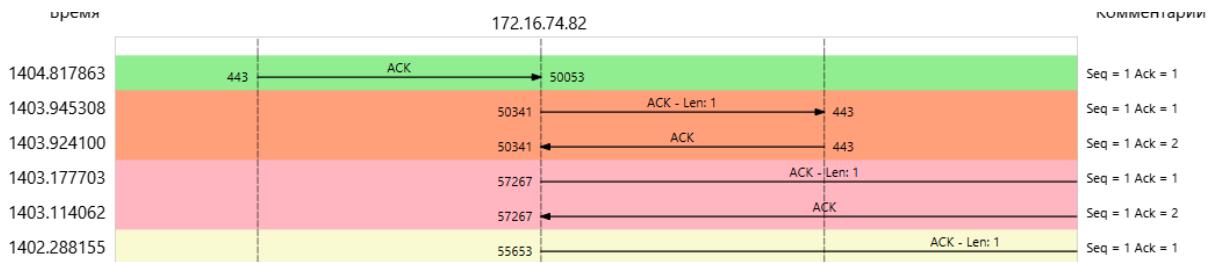
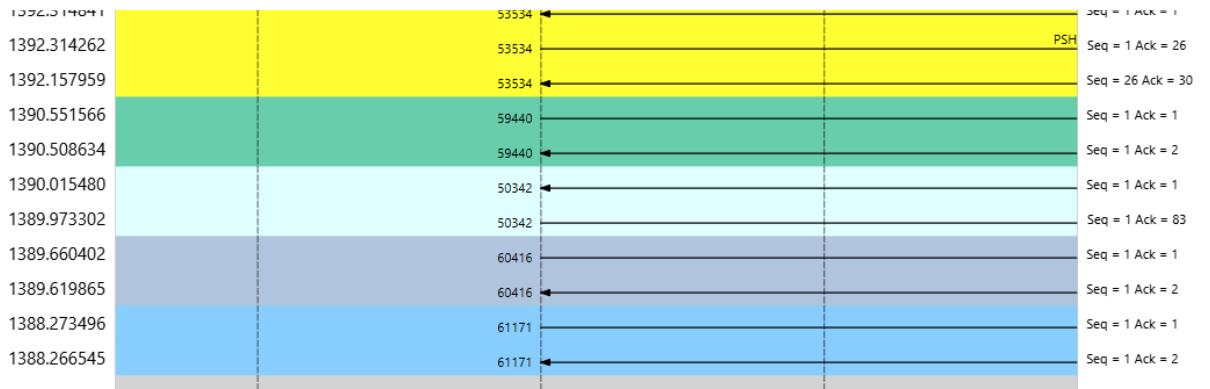


Рисунок 25. График потока



## Выводы

В процессе выполнения данной лабораторной работы я изучил посредством Wireshark кадры Ethernet, проанализировал PDU протоколы транспортного и прикладного уровней стека TCP/IP.