

Отчет по лабораторной работе №3

Анализ трафика в Wireshark

Галацан Николай, НПИбд-01-22

Содержание

1	Цель работы	4
2	Выполнение лабораторной работы	5
2.1	MAC-адресация	5
2.2	Анализ кадров канального уровня в Wireshark	6
2.3	Анализ протоколов транспортного уровня в Wireshark	12
2.4	Анализ handshake протокола TCP в Wireshark	15
3	Выводы	18

Список иллюстраций

2.1	Команда <code>ipconfig</code>	5
2.2	Команда <code>ipconfig /all</code>	6
2.3	Запуск захвата трафика	7
2.4	Пинг шлюза по умолчанию	7
2.5	Кадр ICMP - эхо-запрос: информация о длине кадра	8
2.6	Кадр ICMP - эхо-запрос: информация о типе Ethernet и MAC-адресах	8
2.7	Кадр ICMP - эхо-ответ: информация о длине кадра, типе Ethernet, MAC-адресах	9
2.8	Кадр ARP	10
2.9	Пинг <code>vk.com</code> : запрос	10
2.10	Пинг <code>vk.com</code> : ответ	11
2.11	Кадр ARP - эхо-ответ	11
2.12	Кадр http - запрос	12
2.13	Кадр http - ответ	13
2.14	Кадр dns - запрос	13
2.15	Кадр dns - ответ	14
2.16	Кадр quic - ответ	14
2.17	Первая ступень handshake TCP	15
2.18	Вторая ступень handshake TCP	16
2.19	Третья ступень handshake TCP	17
2.20	График потока	17

1 Цель работы

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

2 Выполнение лабораторной работы

2.1 MAC-адресация

С помощью команды `ipconfig` для ОС типа Windows вывожу информацию о текущем сетевом соединении. Просматриваю информацию о сетевых адаптерах и конкретно о беспроводном соединении. Отсюда можно узнать IPv6-адрес, IPv4-адрес (уникальный IPv4-адрес узла), маску подсети (используется для определения сетевой и узловой частей IPv4-адреса) и шлюз (рис. 2.1).

```
C:\Users\ASUS\ngalacan>ipconfig

Настройка протокола IP для Windows

Адаптер Ethernet VirtualBox Host-Only Network:

    DNS-суффикс подключения . . . . . : 
    Локальный IPv6-адрес канала . . . . : fe80::1bd5:48ba:7abb:de7%14
    IPv4-адрес. . . . . : 192.168.56.1
    Маска подсети . . . . . : 255.255.255.0
    Основной шлюз. . . . . : 

Адаптер беспроводной локальной сети Подключение по локальной сети* 9:

    Состояние среды. . . . . : Среда передачи недоступна.
    DNS-суффикс подключения . . . . . : 

Адаптер беспроводной локальной сети Подключение по локальной сети* 10:

    Состояние среды. . . . . : Среда передачи недоступна.
    DNS-суффикс подключения . . . . . : 

Адаптер беспроводной локальной сети Беспроводная сеть:

    DNS-суффикс подключения . . . . . : 
    IPv6-адрес. . . . . : 2a00:1370:817c:636d:9a5f:987c:2bce:6cc8
    Временный IPv6-адрес. . . . . : 2a00:1370:817c:636d:b809:b59e:7667:c0ed
    Локальный IPv6-адрес канала . . . . : fe80::a9d7:906e:e66c:c49c%8
    IPv4-адрес. . . . . : 192.168.1.9
    Маска подсети . . . . . : 255.255.255.0
    Основной шлюз. . . . . : fe80::1%8
                           192.168.1.1

C:\Users\ASUS\ngalacan>
```

Рис. 2.1: Команда `ipconfig`

Ввожу `ipconfig /all` для вывода более подробной информации. Просматри-

ваю данные о беспроводном соединении. Вижу описание устройства (производитель MediaTek, MAC-адрес - 90-E8-68-2A-62-33). MAC-адрес состоит из 6 октетов: первые 3 октета идентифицируют производителя, последние 3 октета идентифицируют сетевой интерфейс (рис. 2.2).

```
Адаптер беспроводной локальной сети Беспроводная сеть:
DNS-суффикс подключения . . . . . :
Описание. . . . . : MediaTek Wi-Fi 6 MT7921 Wireless LAN Card
Физический адрес. . . . . : 90-E8-68-2A-62-33
DHCP включен. . . . . : Да
Автонастройка включена. . . . . : Да
IPv6-адрес. . . . . : 2a00:1370:817c:636d:9a5f:987c:2bce:6cc8(Основной)
Временный IPv6-адрес. . . . . : 2a00:1370:817c:636d:b809:b59e:7667:c8ed(Основной)
Локальный IPv6-адрес канала. . . . : fe80::a9d7:906e:e66c:c49c%8(Основной)
IPv4-адрес. . . . . : 192.168.1.9(Основной)
Маска подсети. . . . . : 255.255.255.0
Аренда получена. . . . . : 9 октября 2024 г. 13:34:59
Срок аренды истекает. . . . . : 10 октября 2024 г. 14:44:02
Основной шлюз. . . . . : fe80::1%8
                  192.168.1.1
DHCP-сервер. . . . . : 192.168.1.1
IAID DHCPv6 . . . . . : 428927080
DUID клиента DHCPv6 . . . . . : 00-01-00-01-2A-E8-2F-91-90-E8-68-2A-62-33
DNS-серверы. . . . . : 192.168.1.1
NetBios через TCP/IP. . . . . : Включен
C:\Users\ASUS\ngalacan>
```

Рис. 2.2: Команда ipconfig /all

Проверив на специальном сайте производителя устройства по первым 3 октетам я выяснил, что устройство выпущено компанией AzureWave Technology Inc., располагающейся в Тайвани. Взяв первый байт (90) и переведя в двоичную систему счисления, получаю 10010000. Так как последний бит = 0, адрес является индивидуальным. Предпоследний бит = 0, следовательно, адрес глобально администрируемый.

2.2 Анализ кадров канального уровня в Wireshark

Запускаю Wireshark и выбираю беспроводное соединение. Запускаю захват трафика (рис. 2.3).

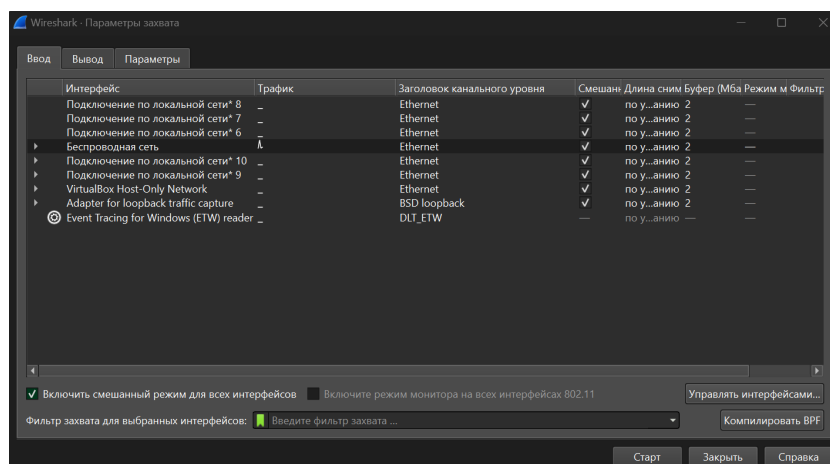


Рис. 2.3: Запуск захвата трафика

Шлюз по умолчанию для моего устройства - 192.168.1.1 (было определено в предыдущем задании). С помощью команды `ping 192.168.1.1` пингую шлюз по умолчанию (рис. 2.4).

```
C:\Users\ASUS\ngalacan>ping 192.168.1.1

Обмен пакетами с 192.168.1.1 по 32 байтами данных:
Ответ от 192.168.1.1: число байт=32 время=2мс TTL=64
Ответ от 192.168.1.1: число байт=32 время=2мс TTL=64
Ответ от 192.168.1.1: число байт=32 время=3мс TTL=64
Ответ от 192.168.1.1: число байт=32 время=2мс TTL=64

Статистика Ping для 192.168.1.1:
    Пакетов: отправлено = 4, получено = 4, потеряно = 0
    (0% потерь)
Приблизительное время приема-передачи в мс:
    Минимальное = 2мсек, Максимальное = 3 мсек, Среднее = 2 мсек

C:\Users\ASUS\ngalacan>
```

Рис. 2.4: Пинг шлюза по умолчанию

Останавливаю захват трафика. В строке фильтра указываю `arp or icmp`. Вижу 4 пакета-запроса и 4 пакета-ответа. Выбираю запрос и просматриваю в нижней части экрана информацию о нем. Длина кадра - 74 байта (592 бита)(рис. 2.5), относится к типу Ethernet II, MAC-адрес источника - адрес моего устройства, MAC-адрес получателя - 54:C2:50:7C:F5:F0. Оба адреса индивидуальные и глобально администрируемые (последние биты в двоичном виде равны 0) (рис. 2.6)

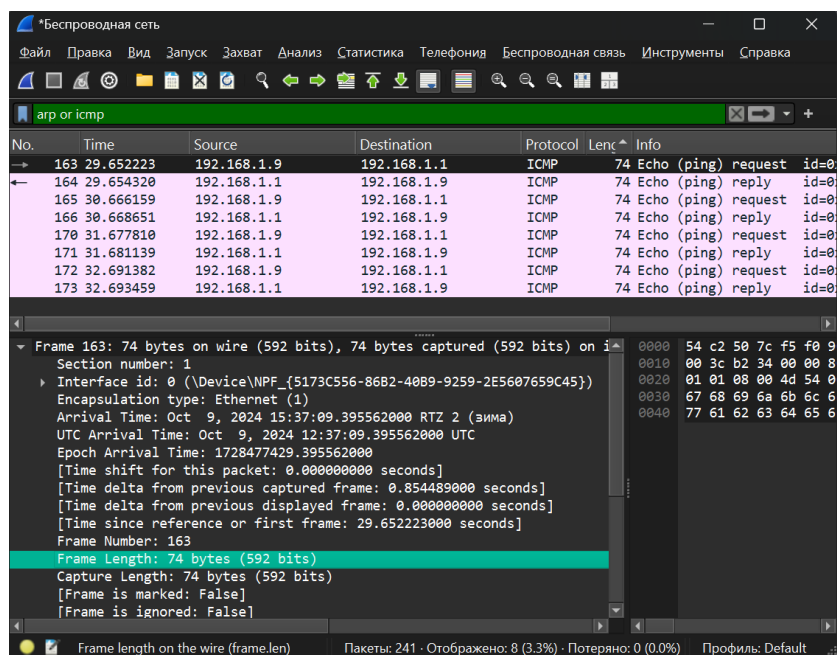


Рис. 2.5: Кадр ICMP - эхо-запрос: информация о длине кадра

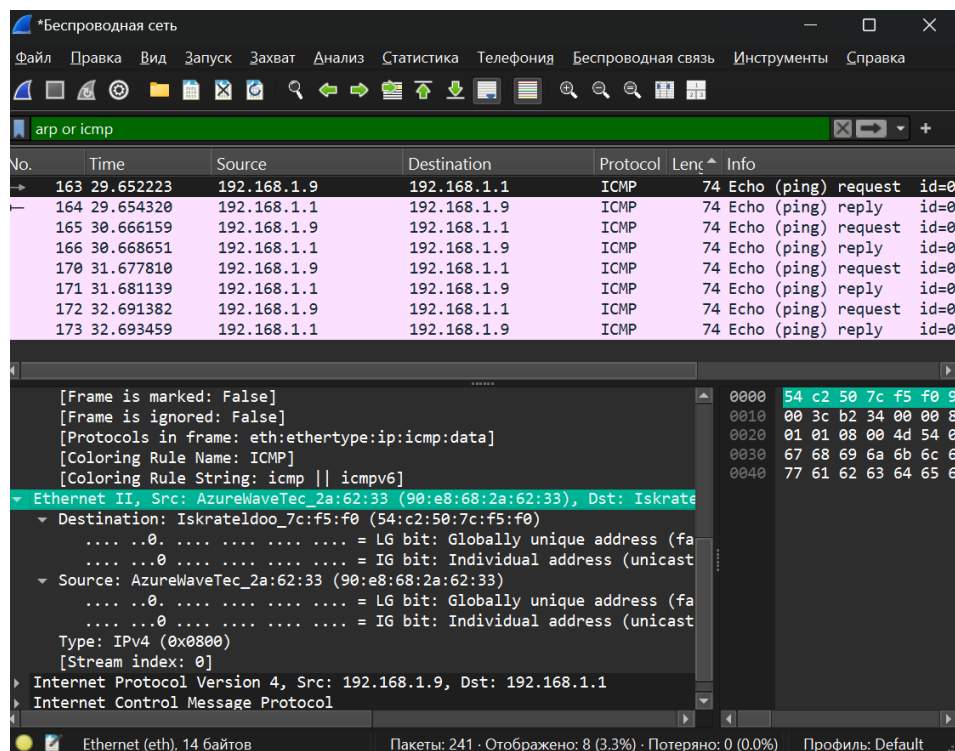


Рис. 2.6: Кадр ICMP - эхо-запрос: информация о типе Ethernet и MAC-адресах

Выбираю эхо-ответ и просматриваю информацию. Длина кадра - 74 байта

(592 бита) (рис. 2.5), относится к типу Ethernet II, MAC-адрес источника - 54:C2:50:7C:F5:F0, MAC-адрес получателя - адрес моего устройства. Оба адреса индивидуальные и глобально администрируемые (последние биты в двоичном виде равны 0) (рис. 2.7).

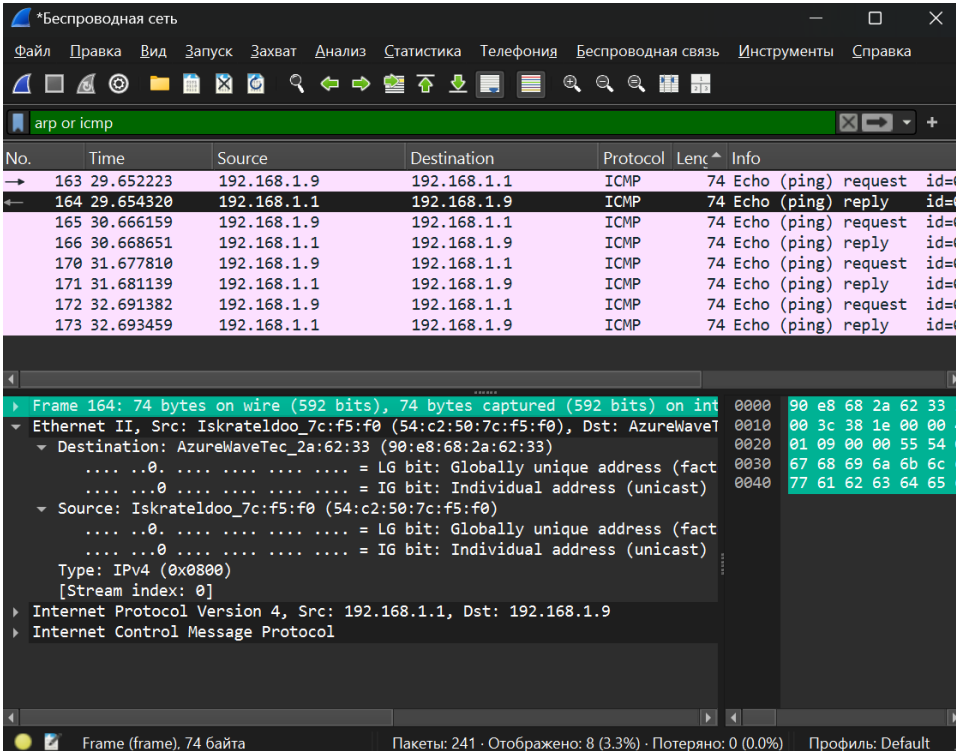


Рис. 2.7: Кадр ICMP - эхо-ответ: информация о длине кадра, типе Ethernet, MAC-адресах

Нахожу кадры данных протокола ARP. Длина кадра равняется 42 байт, заголовок Ethernet занимает первые 14 байт кадра, кадр относится к типу Ethernet II. MAC-адрес пункта назначения – это первые 6 байт заголовка Ethernet, а MAC-адрес источника - следующие 6 байт заголовка Ethernet, оба MAC-адреса являются индивидуальными и глобально администрируемыми. Также в заголовке Ethernet последние два байта обозначают вложенным пакет типа ARP (рис. 2.8).

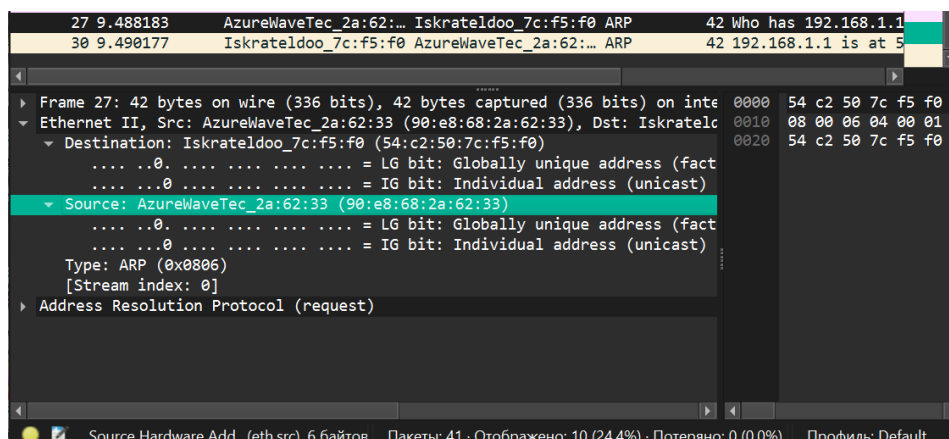


Рис. 2.8: Кадр ARP

Снова начинаю захват трафика. Пингую адрес vk.com. Остановив захват трафика, просматриваю кадры. Длина кадра - 74 байта (592 бита). Для запроса источником является мой сетевой интерфейс, получателем - сам сайт, для ответа наоборот. MAC-адрес точки назначения – это первые 6 байт заголовка Ethernet, а MAC-адрес источника – следующие 6 байт заголовка Ethernet, оба MAC-адреса являются индивидуальными и глобально администрируемыми (рис. 2.9, рис. 2.10).

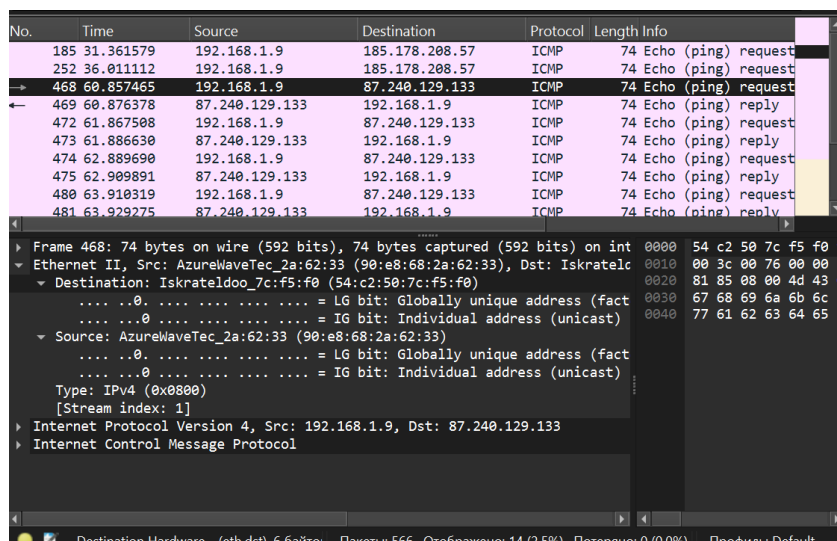


Рис. 2.9: Пинг vk.com: запрос

No.	Time	Source	Destination	Protocol	Length	Info
185	31.361579	192.168.1.9	185.178.208.57	ICMP	74	Echo (ping) request
252	36.011112	192.168.1.9	185.178.208.57	ICMP	74	Echo (ping) request
468	60.857465	192.168.1.9	87.240.129.133	ICMP	74	Echo (ping) request
469	60.876378	87.240.129.133	192.168.1.9	ICMP	74	Echo (ping) reply
472	61.867508	192.168.1.9	87.240.129.133	ICMP	74	Echo (ping) request
473	61.886630	87.240.129.133	192.168.1.9	ICMP	74	Echo (ping) reply
474	62.889690	192.168.1.9	87.240.129.133	ICMP	74	Echo (ping) request
475	62.909891	87.240.129.133	192.168.1.9	ICMP	74	Echo (ping) reply
480	63.910319	192.168.1.9	87.240.129.133	ICMP	74	Echo (ping) request
481	63.929275	87.240.129.133	192.168.1.9	ICMP	74	Echo (ping) reply

Frame 469: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface 0	
Ethernet II, Src: Iskrateldoo_7c:f5:f0 (54:c2:50:7c:f5:f0), Dst: AzureWaveTec_2a:62:33 (90:e8:68:2a:62:33)	0000 90 e8 68 2a 62 33 5
Destination: AzureWaveTec_2a:62:33 (90:e8:68:2a:62:33)	0010 00 3c da 39 00 00 3
Source: Iskrateldoo_7c:f5:f0 (54:c2:50:7c:f5:f0)	0020 01 09 00 00 55 43 0
Type: IPv4 (0x0800)	0030 67 68 69 6a 6b 6c 6
[Stream index: 1]	0040 77 61 62 63 64 65 6
Internet Protocol Version 4, Src: 87.240.129.133, Dst: 192.168.1.9	
Internet Control Message Protocol	

Рис. 2.10: Пинг vk.com: ответ

Изучаю запросы и ответы ARP. MAC-адрес точки назначения – это первые 6 байт заголовка Ethernet, а MAC-адрес источника – следующие 6 байт заголовка Ethernet, оба MAC-адреса являются индивидуальными и глобально администрируемыми (рис. 2.11).

No.	Time	Source	Destination	Protocol	Length	Info
473	61.886630	87.240.129.133	192.168.1.9	ICMP	74	Echo (ping) reply
474	62.889690	192.168.1.9	87.240.129.133	ICMP	74	Echo (ping) request
475	62.909891	87.240.129.133	192.168.1.9	ICMP	74	Echo (ping) reply
480	63.910319	192.168.1.9	87.240.129.133	ICMP	74	Echo (ping) request
481	63.929275	87.240.129.133	192.168.1.9	ICMP	74	Echo (ping) reply
499	68.936240	Iskrateldoo_7c:f5:f0	AzureWaveTec_2a:62:33	ARP	42	Who has 192.168.1.9
500	68.936269	AzureWaveTec_2a:62:33	Iskrateldoo_7c:f5:f0	ARP	42	192.168.1.9 is at 9
527	80.883451	GuangzhouShi_e8:b3:39	AzureWaveTec_2a:62:33	ARP	42	Who has 192.168.1.9
528	80.883481	AzureWaveTec_2a:62:33	GuangzhouShi_e8:b3:39	ARP	42	192.168.1.9 is at 9

Frame 528: 42 bytes on wire (336 bits), 42 bytes captured (336 bits) on interface 0	
Ethernet II, Src: AzureWaveTec_2a:62:33 (90:e8:68:2a:62:33), Dst: GuangzhouShi_e8:b3:39 (bc:6b:ff:e8:b3:39)	0000 bc 6b ff e8 b3 39 9
Destination: GuangzhouShi_e8:b3:39 (bc:6b:ff:e8:b3:39)	0010 08 00 06 04 00 02 9
Source: AzureWaveTec_2a:62:33 (90:e8:68:2a:62:33)	0020 bc 6b ff e8 b3 39 c
Type: ARP (0x0806)	
[Stream index: 2]	
Address Resolution Protocol (reply)	

Рис. 2.11: Кадр ARP - эхо-ответ

2.3 Анализ протоколов транспортного уровня в Wireshark

Запустив Wireshark, начинаю захват трафика. Открываю в браузере сайт, работающий по протоколу HTTP (<http://info.cern.ch/>). Перемещаюсь по страницам. В строке фильтра указываю http и просматриваю информацию по протоколу TCP о запросе. Порт источника задан случайно и равен 51310, порт назначения равен 80 - это стандартный порт HTTP (рис. 2.12)

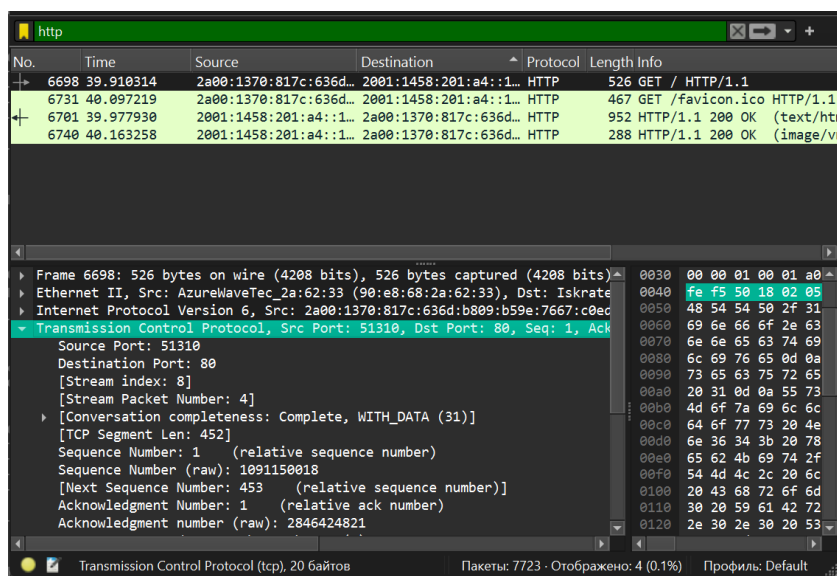


Рис. 2.12: Кадр http - запрос

В случае ответа порты заданы наоборот, то есть источник - 80 порт, назначение - 51310 (рис. 2.13)

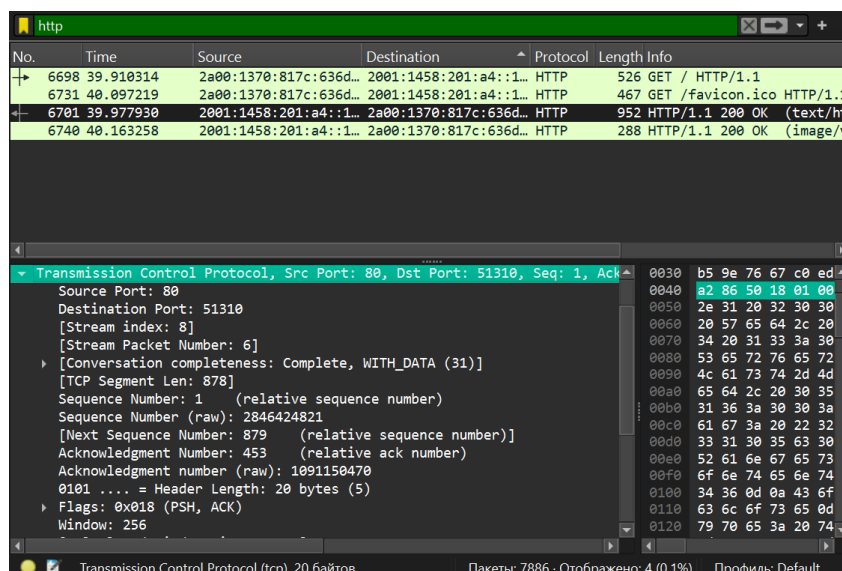


Рис. 2.13: Кадр http - ответ

В Wireshark в строке фильтра ввожу dns и просматриваю информацию по протоколу UDP в случае запроса. Порт источника задан случайно и равен 58049, порт назначения равен 53 (порт DNS по умолчанию) (рис. 2.14). В случае ответа порты заданы наоборот рис. (2.15).

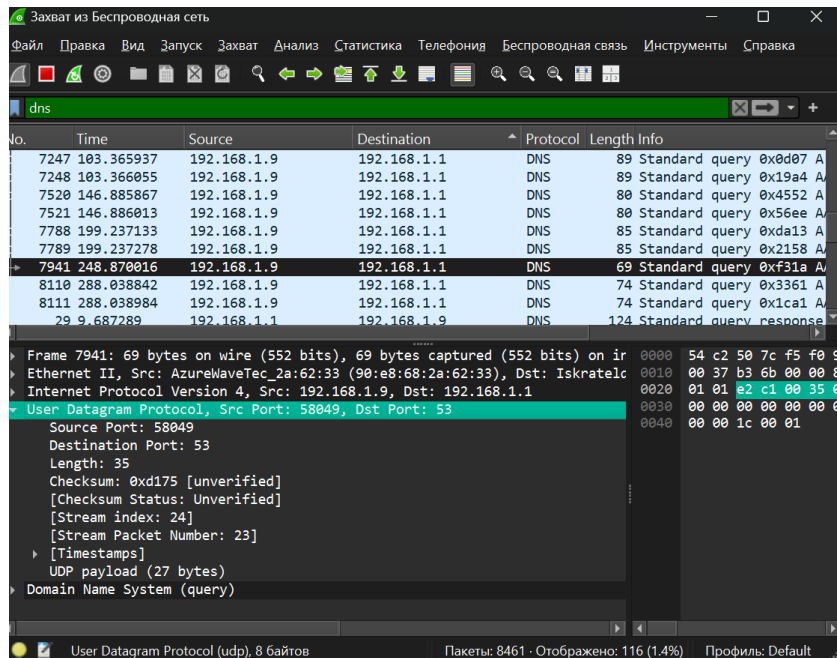


Рис. 2.14: Кадр dns - запрос

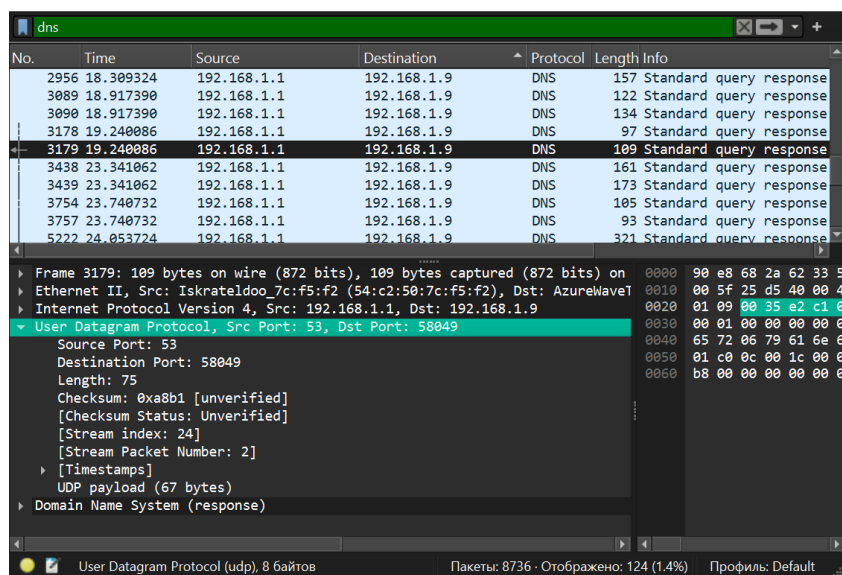


Рис. 2.15: Кадр dns - ответ

В строке фильтра указываю quic. Порт источника задан случайно, порт назначения равен 443 - это стандартный порт HTTPS, следовательно, quic сразу шифруется. В случае ответа порты заданы наоборот (рис. 2.16).

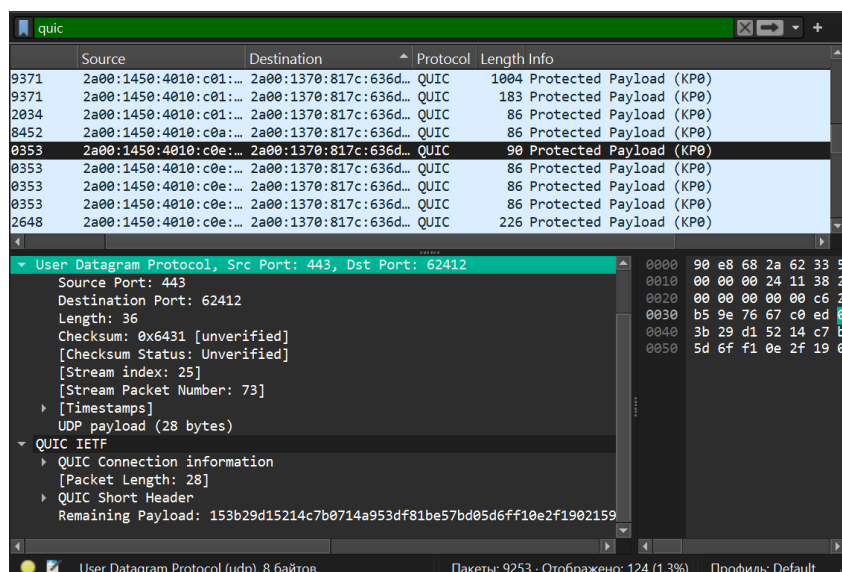


Рис. 2.16: Кадр quic - ответ

2.4 Анализ handshake протокола TCP в Wireshark

Начав захват трафика, запускаю в браузере сайт, работающий по протоколу HTTP (<http://info.cern.ch/>). Установление связи клиент-сервер в TCP осуществляется в три этапа (трёхступенчатый handshake).

1. Режим активного доступа (Active Open). Клиент посылает 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)

Нахожу кадр с флагом SYN. Sequence Number = 0 (рис. 2.17).

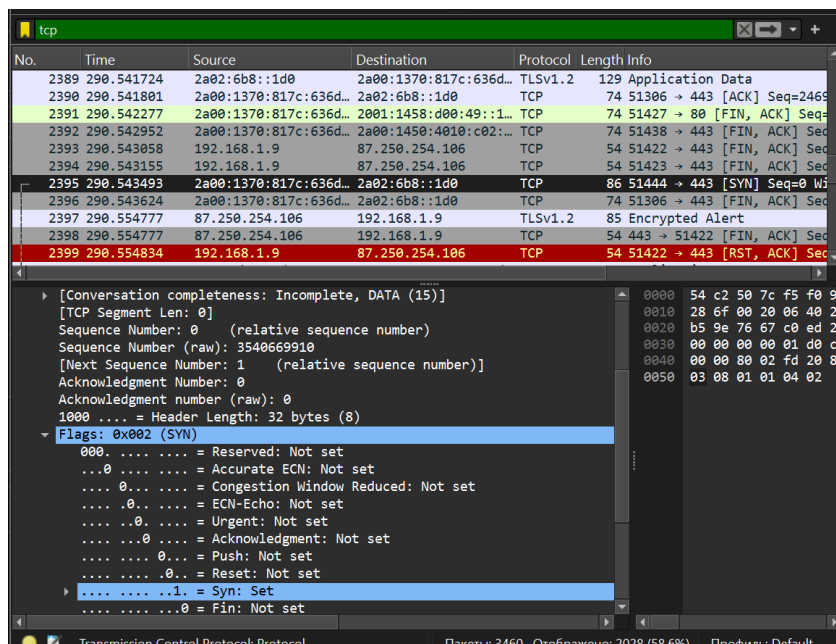


Рис. 2.17: Первая ступень handshake TCP

2. 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.

Кадр с флагами SYN и ACK, где ACK равен Sequence Number из предыдущего шага, увеличенный на 1 ($0 + 1 = 1$) (рис. 2.18).

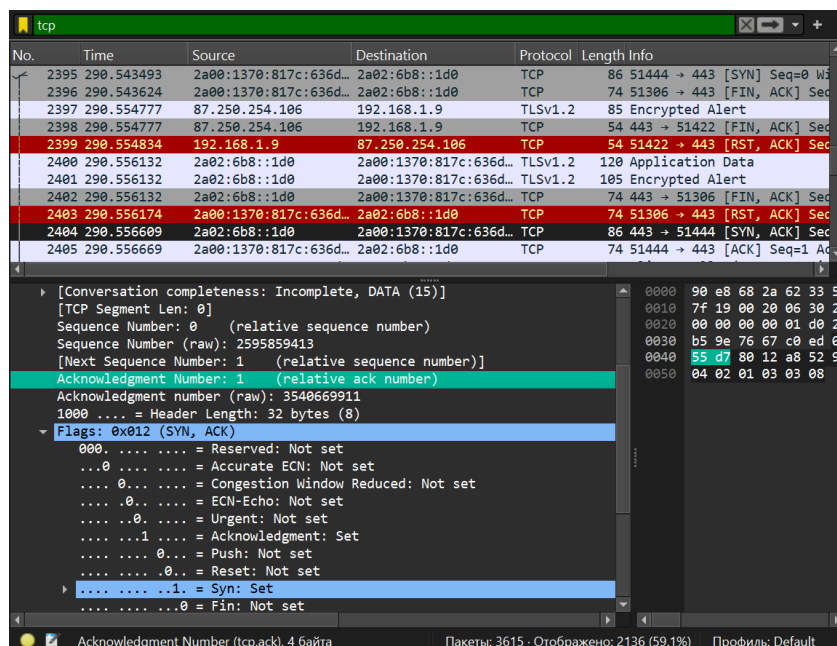


Рис. 2.18: Вторая ступень handshake TCP

3. Завершение рукопожатия. Клиент отправляет подтверждение получения SYNсегмента от сервера с идентификатором, равным $ISN(\text{сервера})+1$: ACK, $ISSa+1$, $ACK(ISSb+1)$. В этом пакете установлен бит ACK, 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.

Теперь клиент может посылать пакеты с данными на сервер по только что созданному виртуальному TCP-каналу: ACK, $ISSa+1$, $ACK(ISSb+1)$; DATA.

Кадр с флагом ACK, где Sequence Number равен 1, Acknowledgment Number равен 1 (рис. 2.19).

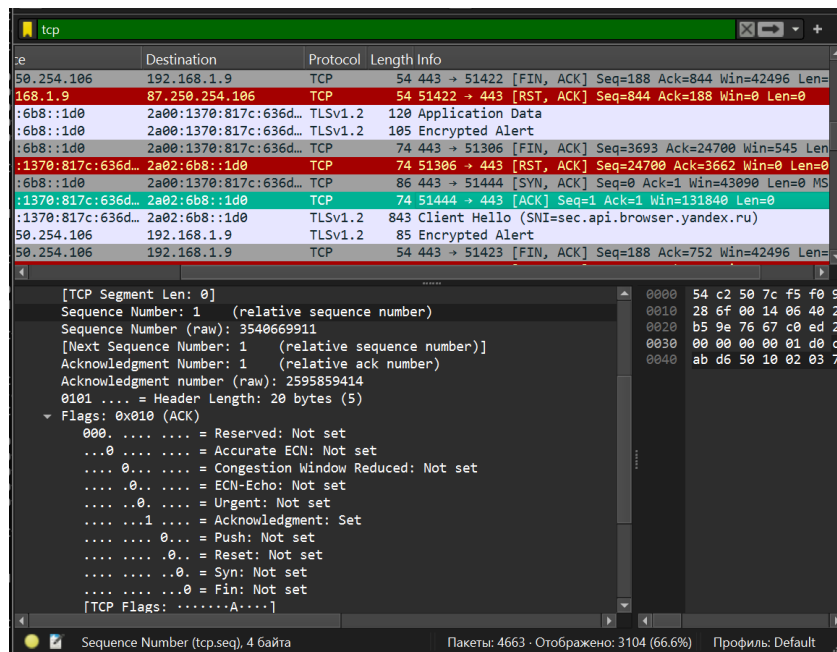


Рис. 2.19: Третья ступень handshake TCP

В Wireshark в меню «Статистика» выбираю «График Потока». На графике видно, что сначала клиент послал сообщение на сервер, значение Seq = 0. Затем сервер откликнулся, значение Seq = 0, а значение Sck = 1. И в третьем пакете клиент отправил подтверждение получения SYN-сегмента, оба значения Syn и Ack равны 1 (рис. 2.20).

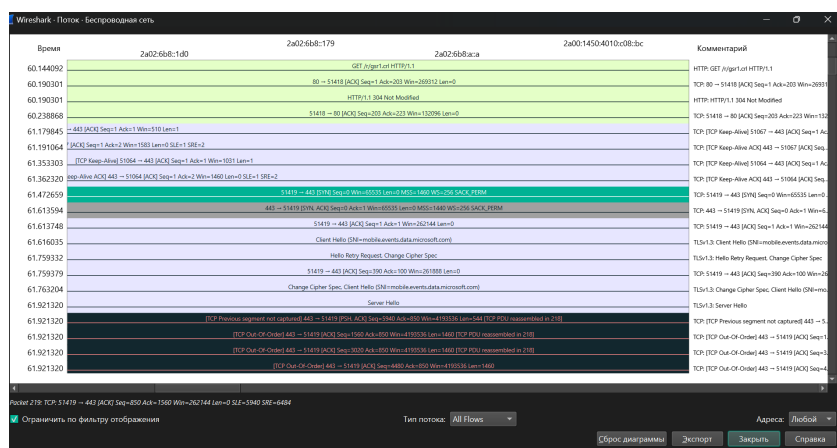


Рис. 2.20: График потока

В Wireshark останавливаю захват трафика.

3 Выводы

В результате выполнения работы были изучены посредством Wireshark кадры Ethernet, произведен анализ PDU протоколов транспортного и прикладного уровней стека TCP/IP.