Отчет по лабораторной работе №3

Анализ трафика в Wireshark

Галацан Николай, НПИбд-01-22

Содержание

Список иллюстраций

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).

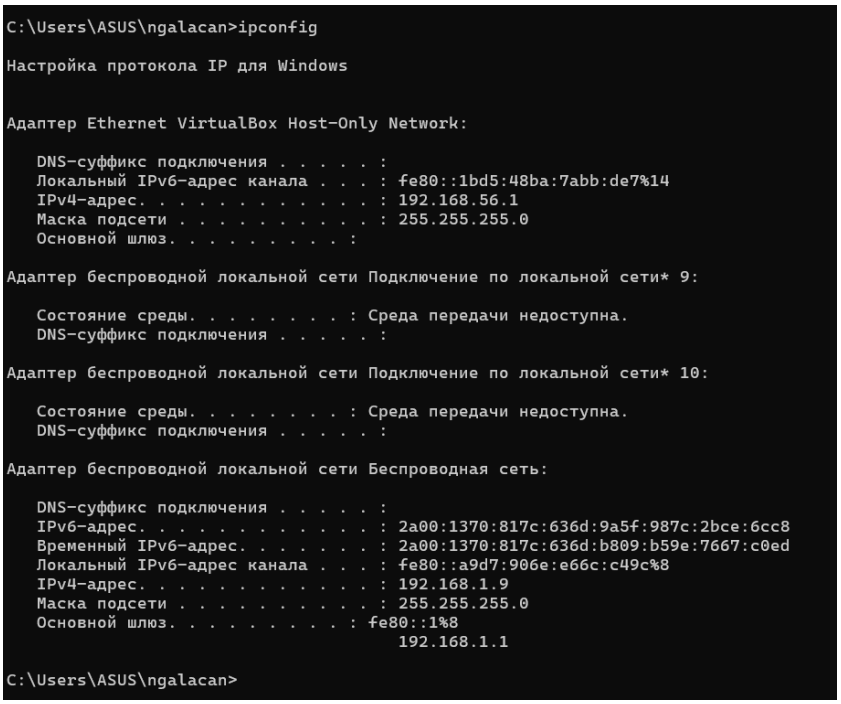

Рис. 2.1: Команда ipconfig

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).

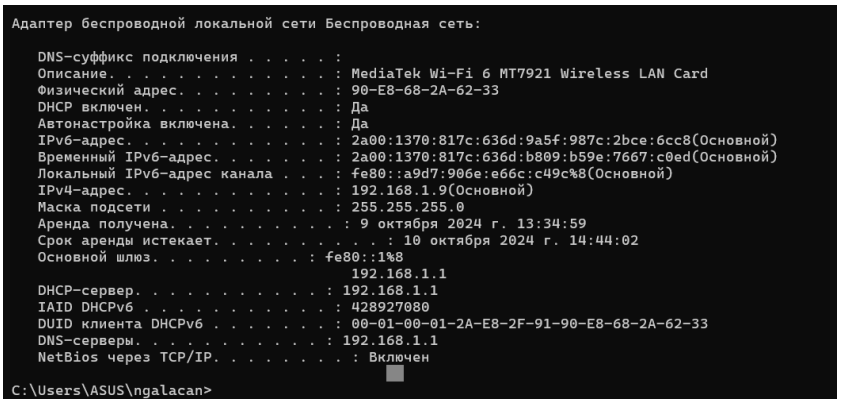

Рис. 2.2: Команда `ipconfig /all`

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).

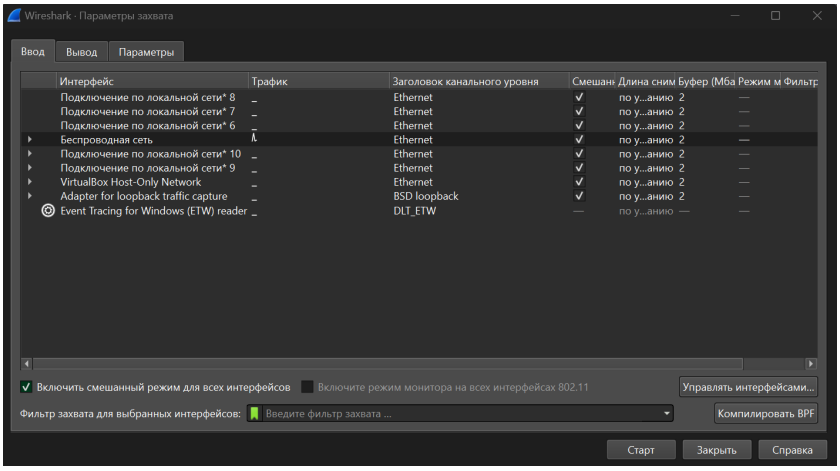

Рис. 2.3: Запуск захвата трафика

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 с 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)

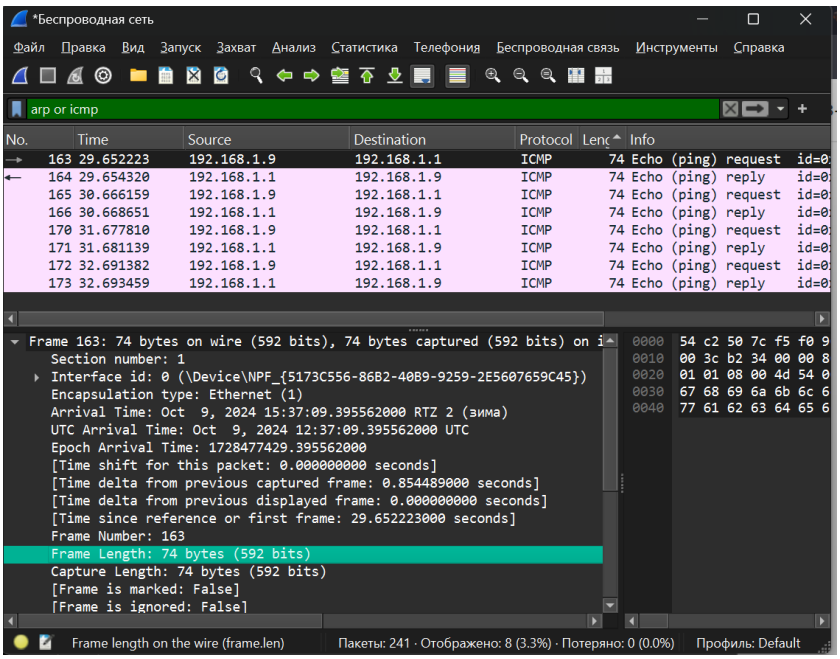

Рис. 2.5: Кадр ICMP - эхо-запрос: информация о длине кадра

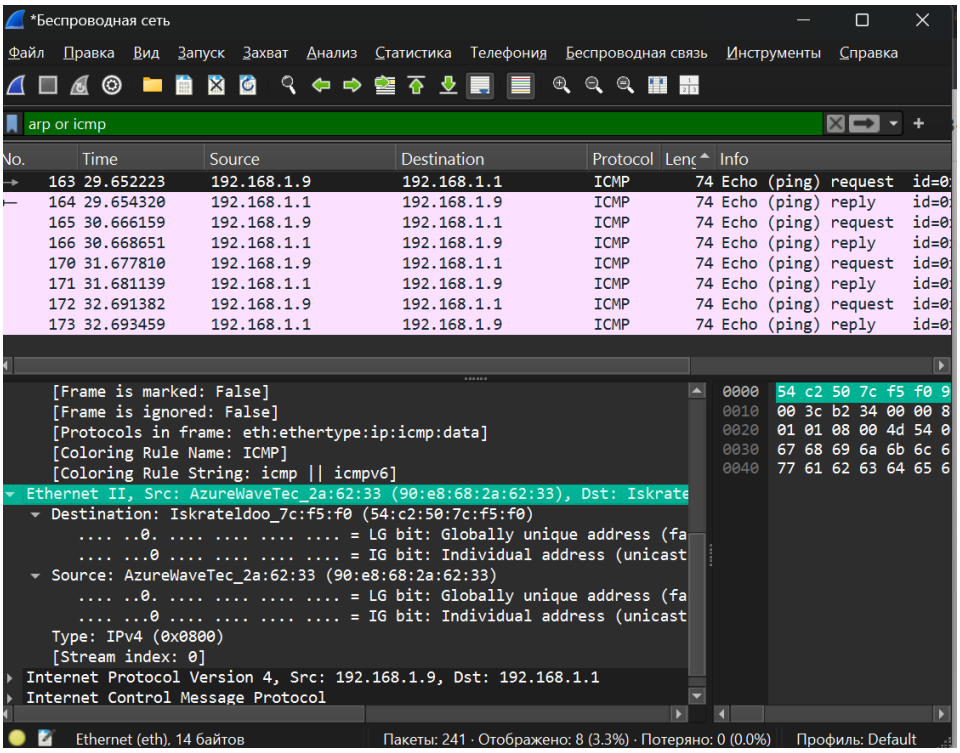

Рис. 2.6: Кадр ICMP - эхо-запрос: информация о типе Ethernet и MAC-адресах

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).

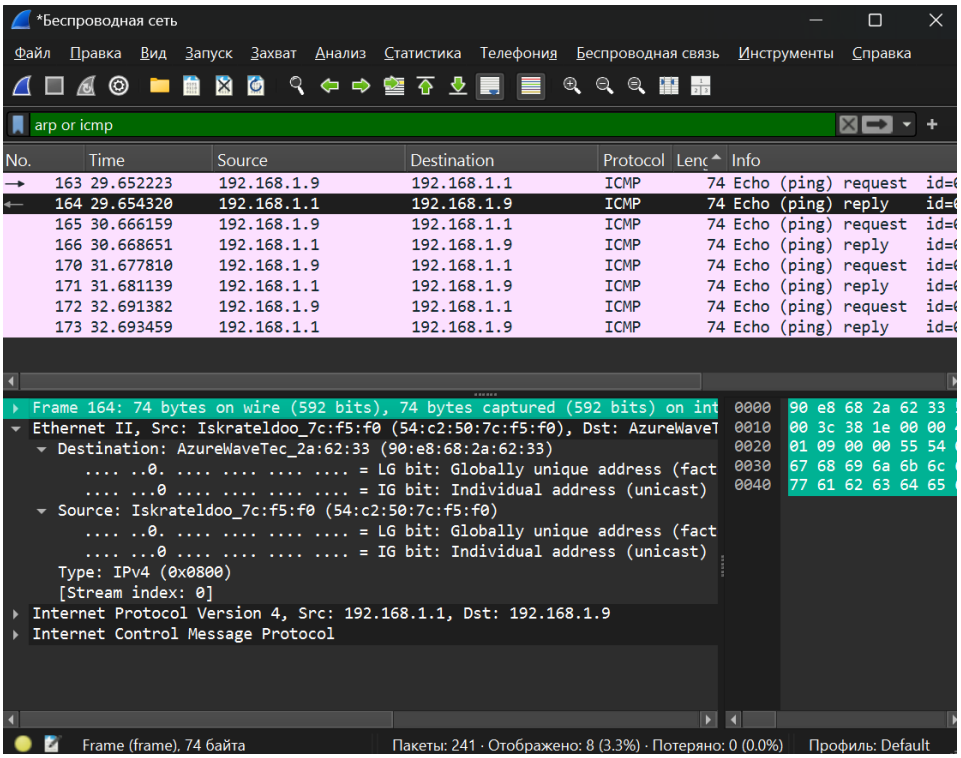

Рис. 2.7: Кадр ICMP - эхо-ответ: информация о длине кадра, типе Ethernet, MAC-адресах

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).

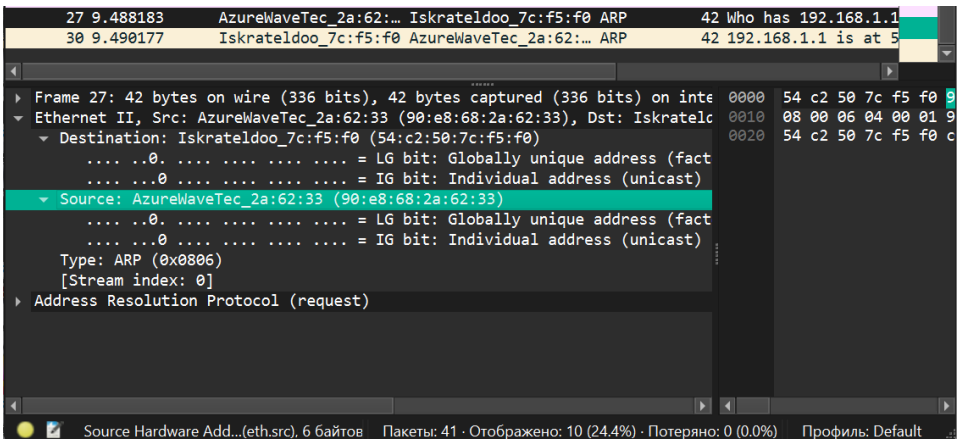

Рис. 2.8: Кадр ARP

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).

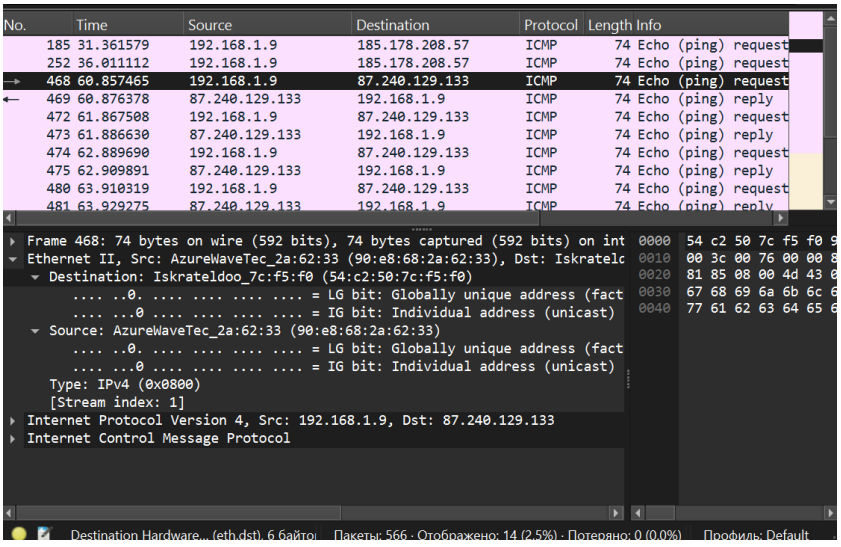

Рис. 2.9: Пинг `vk.com`: запрос

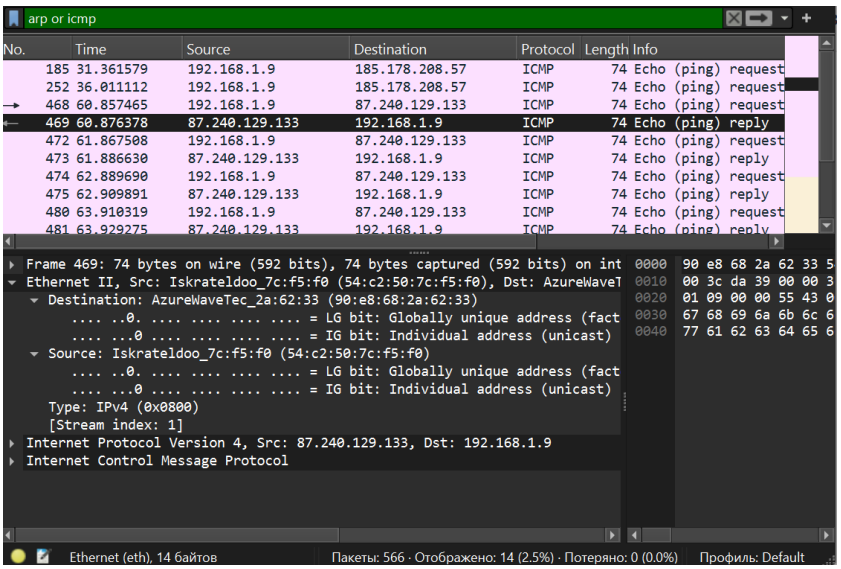

Рис. 2.10: Пинг vk.com: ответ

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).

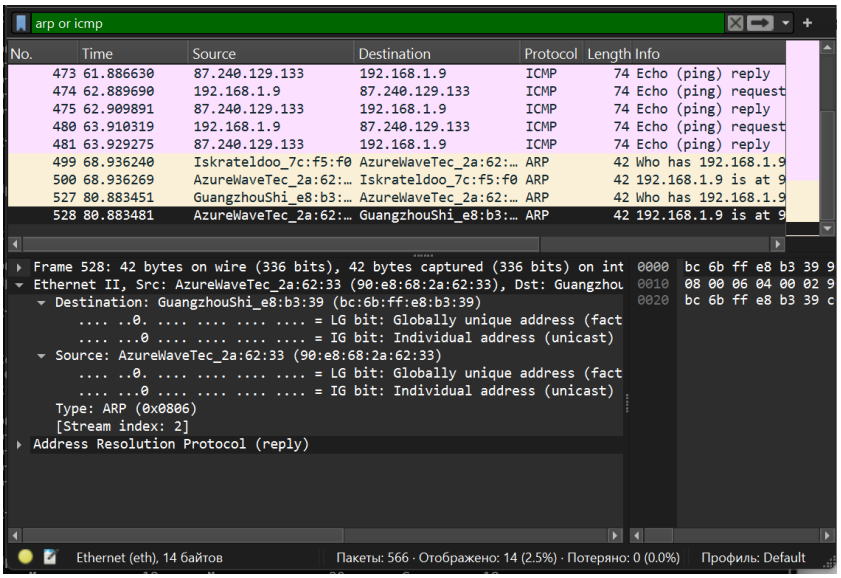

Рис. 2.11: Кадр ARP - эхо-ответ

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)

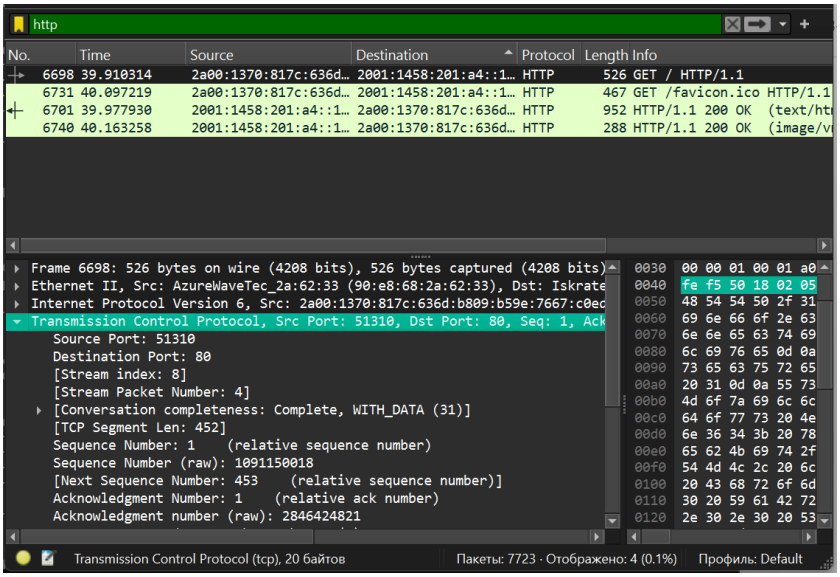

Рис. 2.12: Кадр http - запрос

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

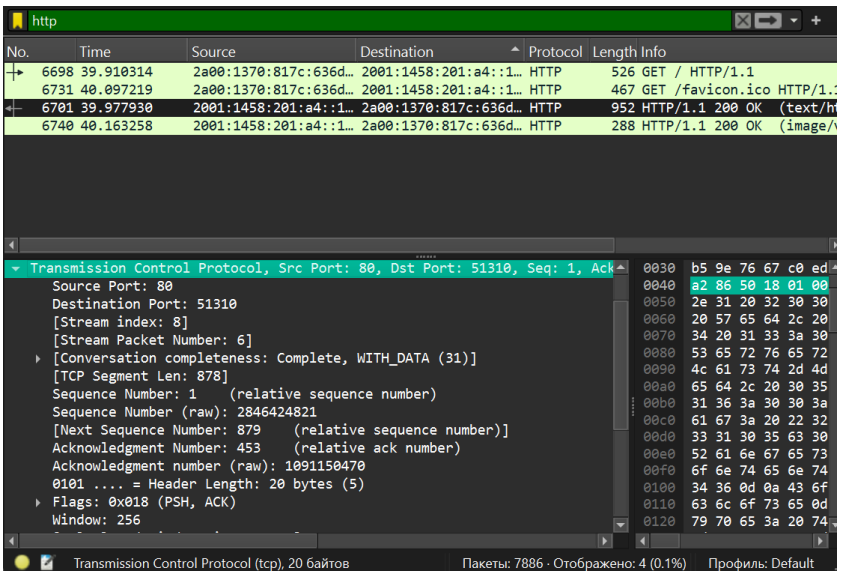

Рис. 2.13: Кадр http - ответ

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).

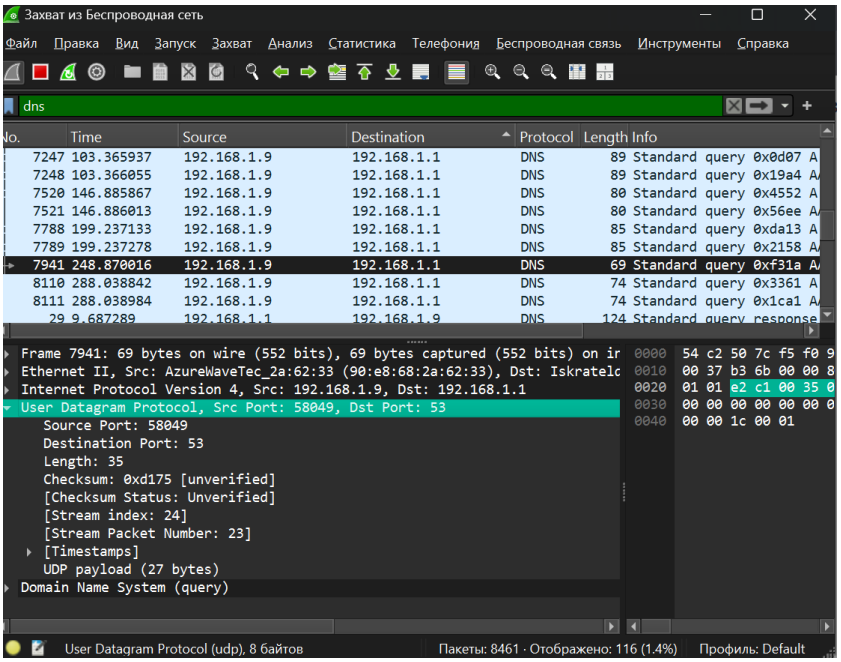

Рис. 2.14: Кадр dns - запрос

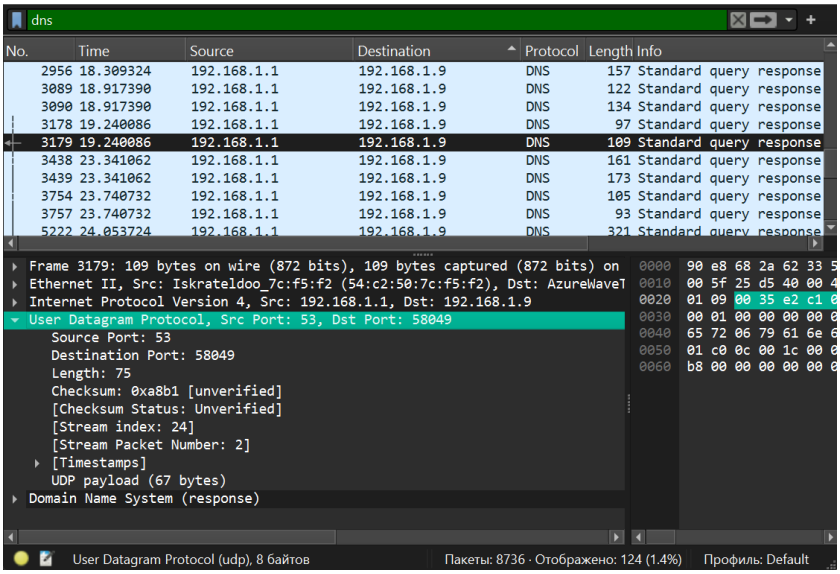

Рис. 2.15: Кадр dns - ответ

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).

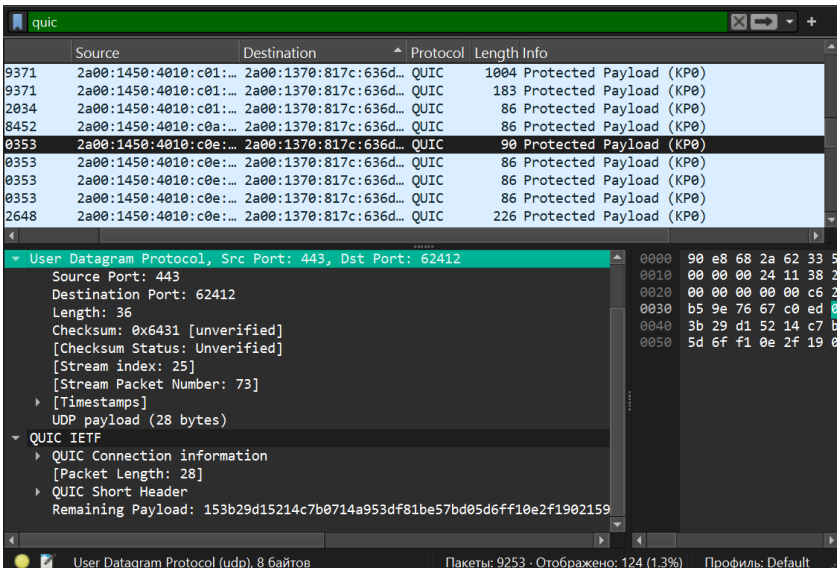

Рис. 2.16: Кадр quic - ответ

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).

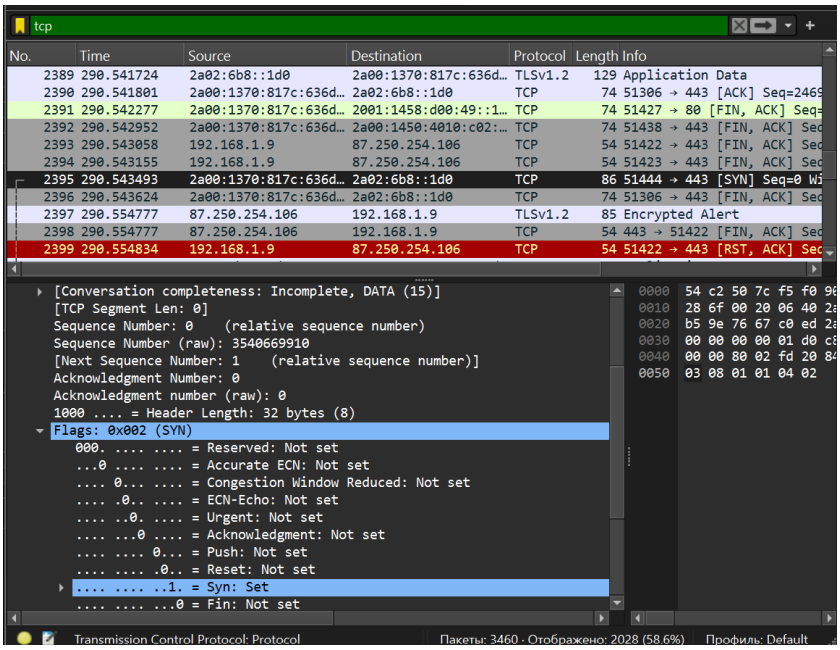

Рис. 2.17: Первая ступень handshake TCP

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 B устанавливается начальное значение счётчика — ISSb; поле Номер подтверждения (Acknowledgment Number) содержит значение ISSa, полученное в первом пакете от хоста A 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).

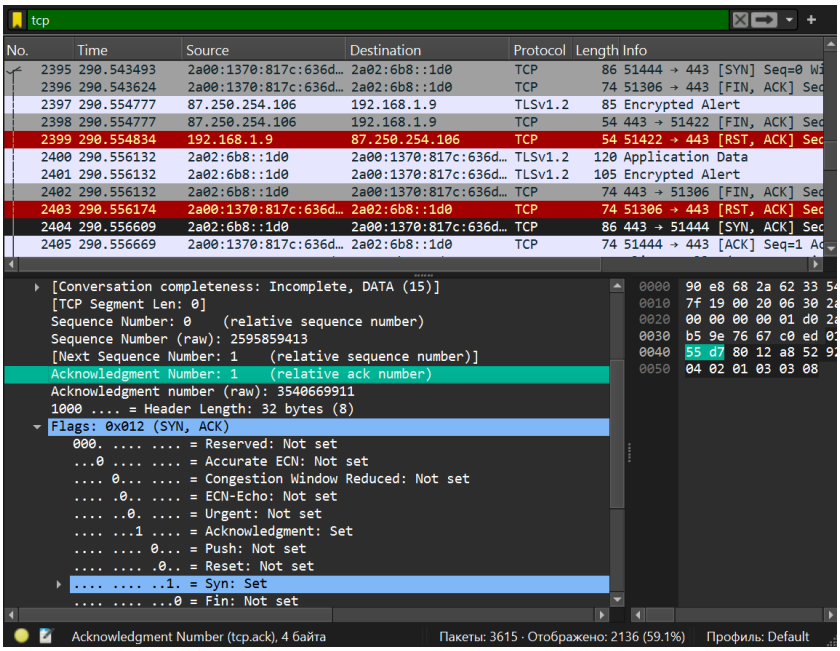

Рис. 2.18: Вторая ступень handshake TCP

3. Завершение рукопожатия. Клиент отправляет подтверждение получения SYNсегмента от сервера с идентификатором, равным ISN (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).

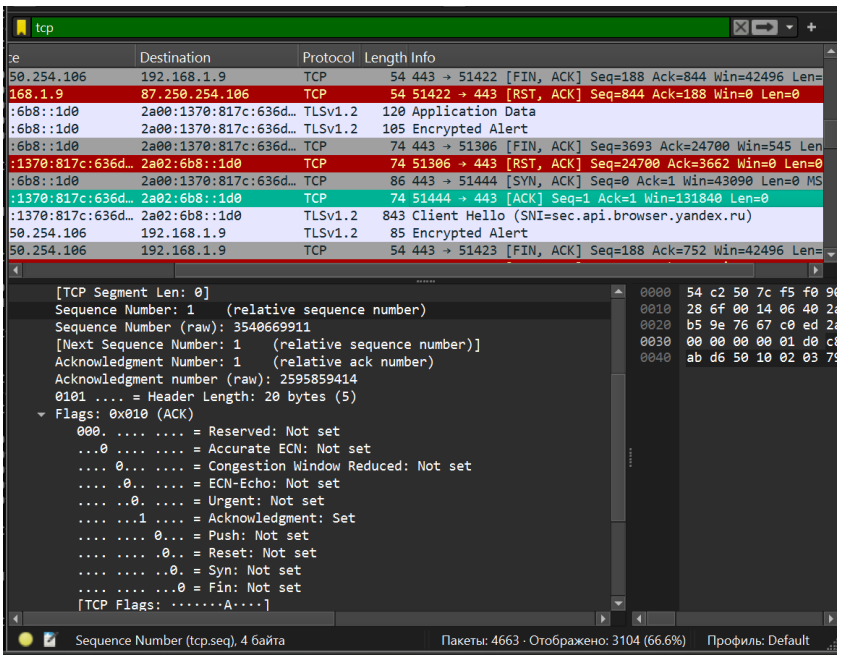

Рис. 2.19: Третья ступень handshake TCP

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 оправил подтверждение получение SYN-сегмента, оба значения Syn и Ack равны 1 (рис. 2.20).

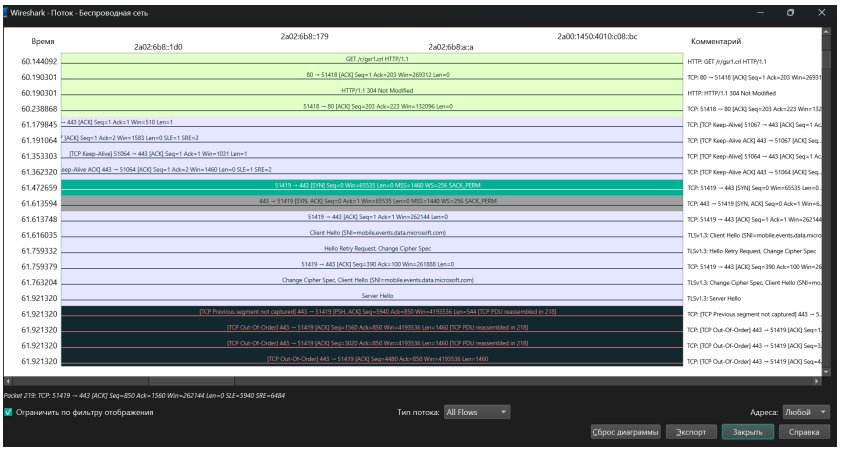

Рис. 2.20: График потока

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.