Отчёт по лабораторной работе 3

Анализ трафика в Wireshark

Метвалли Ахмед Фарг Набеех

Содержание

Список иллюстраций

Список таблиц

1 Цель работы

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

2 Выполнение

3 Выполнение

3.1 Анализ кадров канального уровня в Wireshark

1. На устройство установлен и запущен **Wireshark**. Для анализа был выбран активный беспроводной сетевой интерфейс, начат захват трафика.

2. С помощью команды **ipconfig** в консоли Windows определён IP-адрес устройства и шлюз по умолчанию.
   Устройство имеет адрес `192.168.212.42`, а шлюз по умолчанию — `192.168.212.16`.

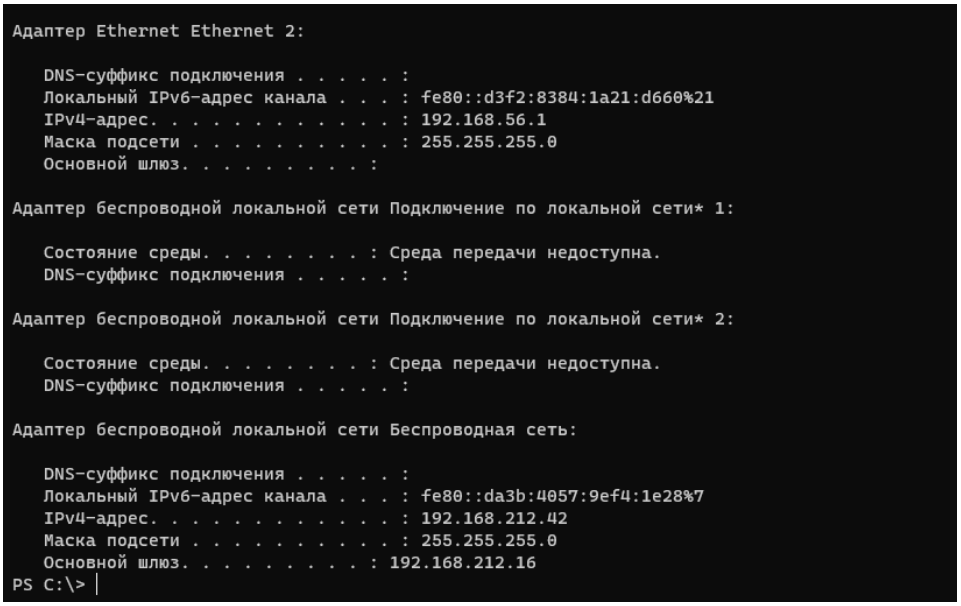


Рис. 3.1: Результат ipconfig

3. Выполнена команда **ping** шлюза по умолчанию (`192.168.212.16`). На экране

отобразились ответы от шлюза.

4. В Wireshark был применён фильтр `arp or icmp`. В списке пакетов отобразились только пакеты ARP и ICMP.

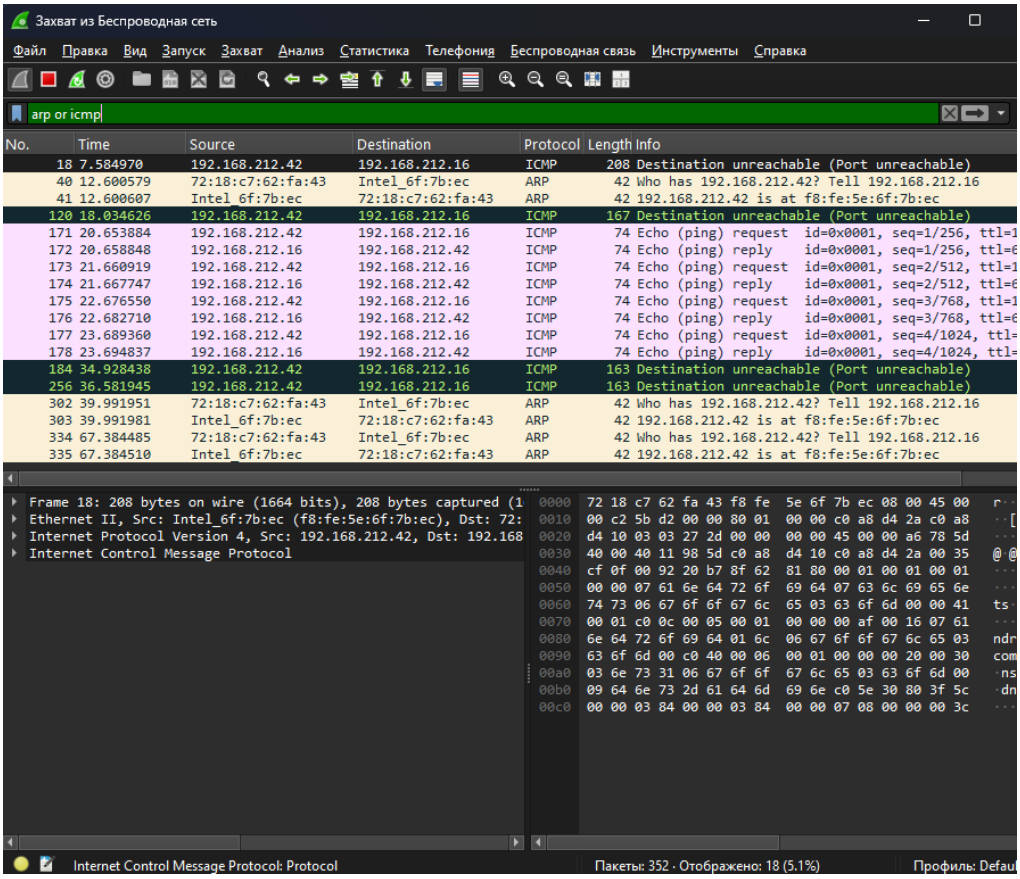


Рис. 3.2: Фильтр arp or icmp

5. Анализ ICMP эхо-запроса:

   - Длина кадра — 74 байта.
   - Тип кадра Ethernet II.
   - MAC-адрес источника: `Intel_6f:7b:ec (f8:fe:5e:6f:7b:ec)`.

- MAC-адрес шлюза: `72:18:c7:62:fa:43`.

- Тип MAC-адресов — индивидуальные (unicast).

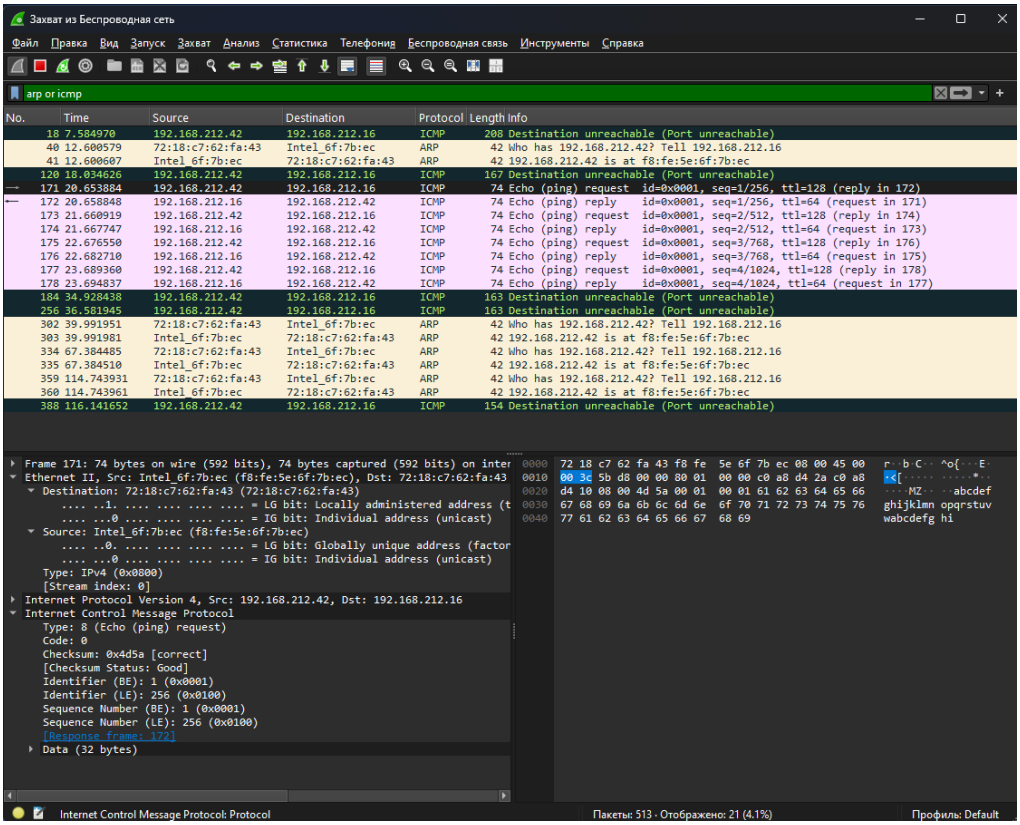


Рис. 3.3: ICMP Echo Request

6. Анализ ICMP эхо-ответа:

- Длина кадра — 74 байта.

- Тип кадра Ethernet II.

- MAC-адрес источника: `72:18:c7:62:fa:43`.

- MAC-адрес получателя: `f8:fe:5e:6f:7b:ec`.

- Тип MAC-адресов — индивидуальные (unicast).

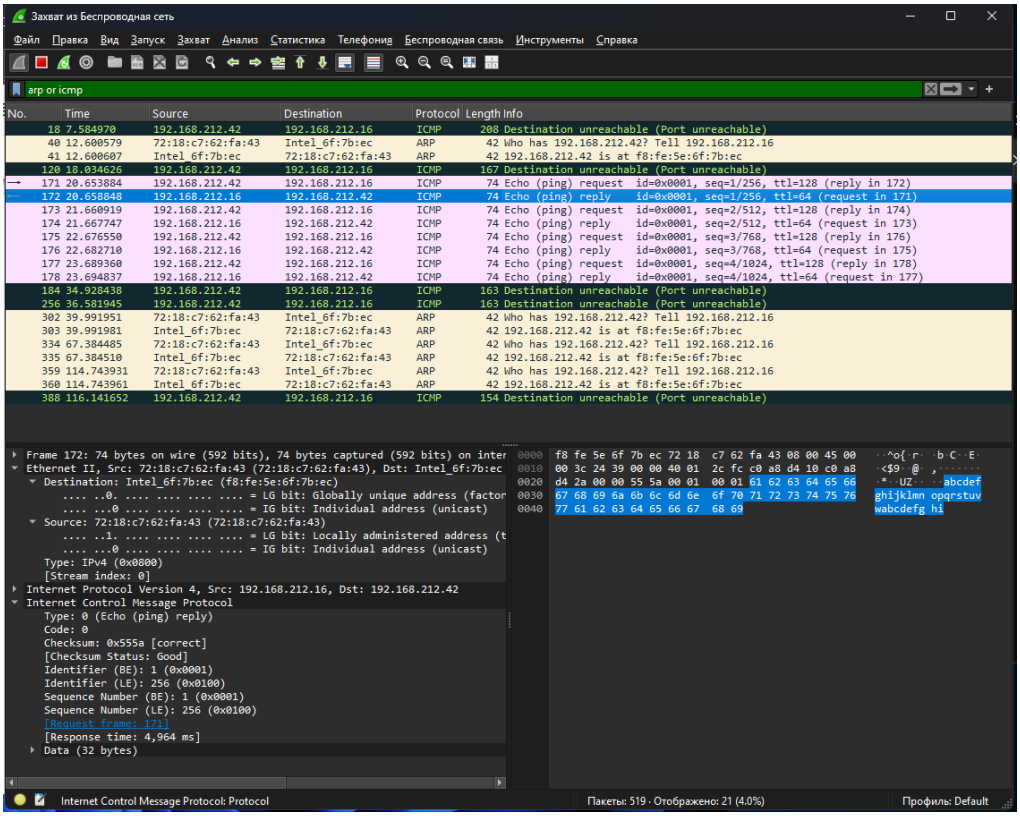


Рис. 3.4: ICMP Echo Reply

7. Анализ ARP-запроса:

   - Длина кадра — 42 байта.

   - Отправитель: `192.168.212.42`.

   - Запрос: «Кто имеет IP 192.168.212.16?»

   - Тип MAC-адресов — индивидуальные.

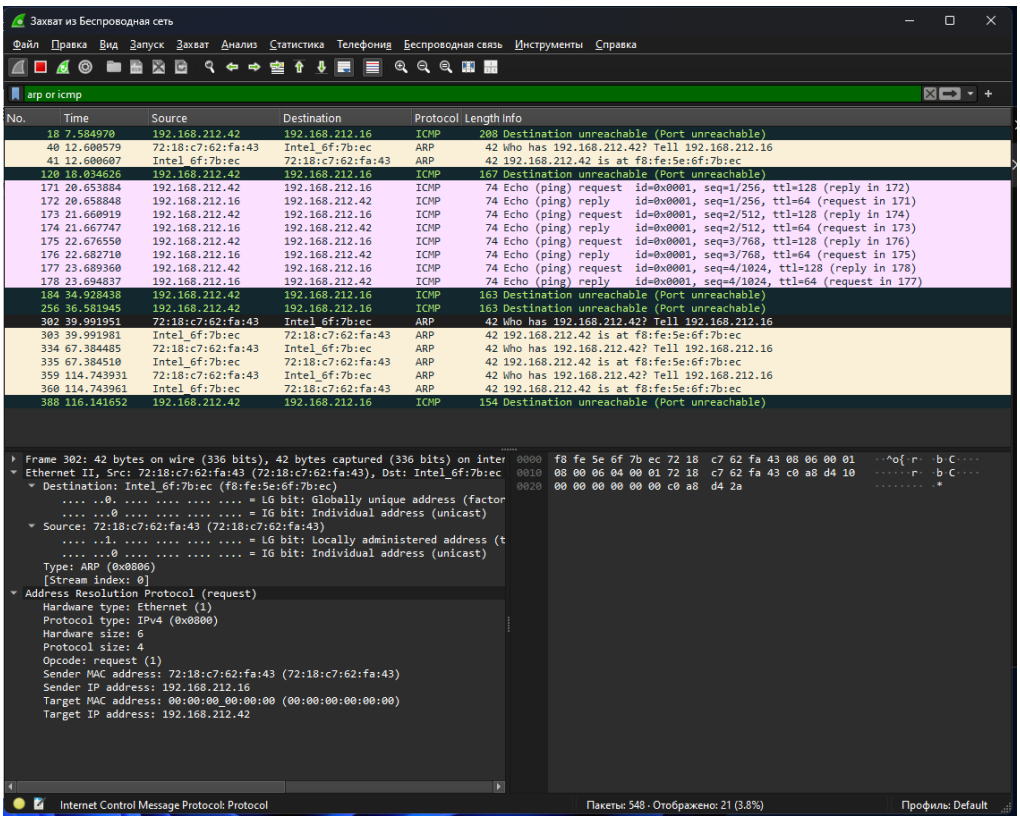


Рис. 3.5: ARP Request

8. Далее был выполнен ping внешнего узла `ya.ru`. Ответы пришли успешно, что подтверждает работоспособность соединения.

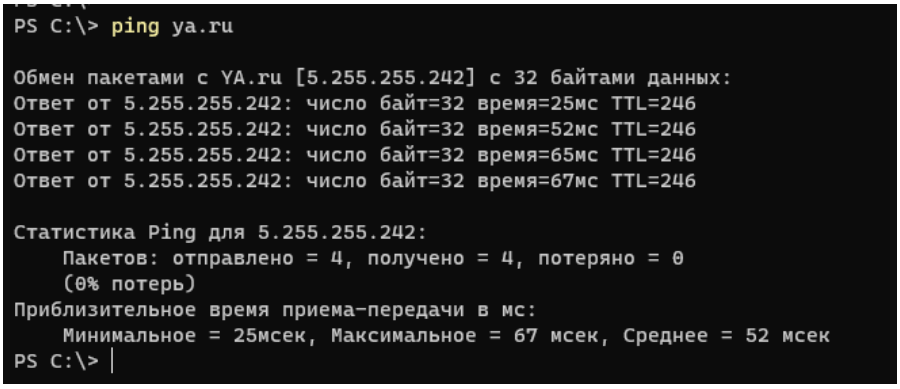
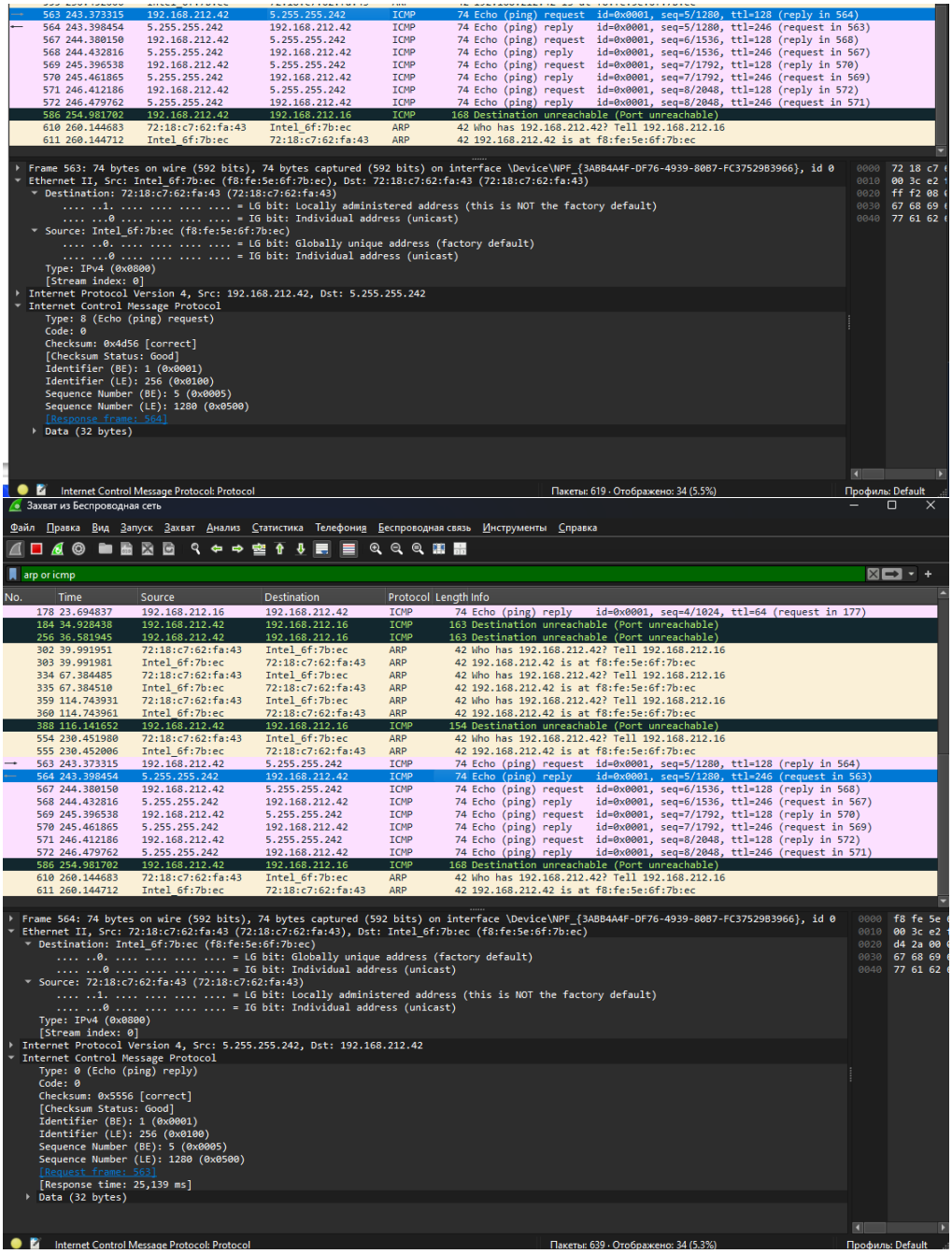


Рис. 3.6: Ping ya.ru

9. В Wireshark зафиксированы пакеты ICMP при обмене с внешним сервером (`5.255.255.242`). Отображены эхо-запросы и ответы, а также промежуточ-

ные ARP-запросы.



3.2 Анализ протоколов транспортного уровня в Wireshark

1. На устройстве был запущен **Wireshark**, выбран активный сетевой интерфейс и начат захват трафика.

2. В браузере был открыт сайт, работающий по протоколу **HTTP**. Для получения достаточного количества пакетов осуществлён переход по разделам сайта.

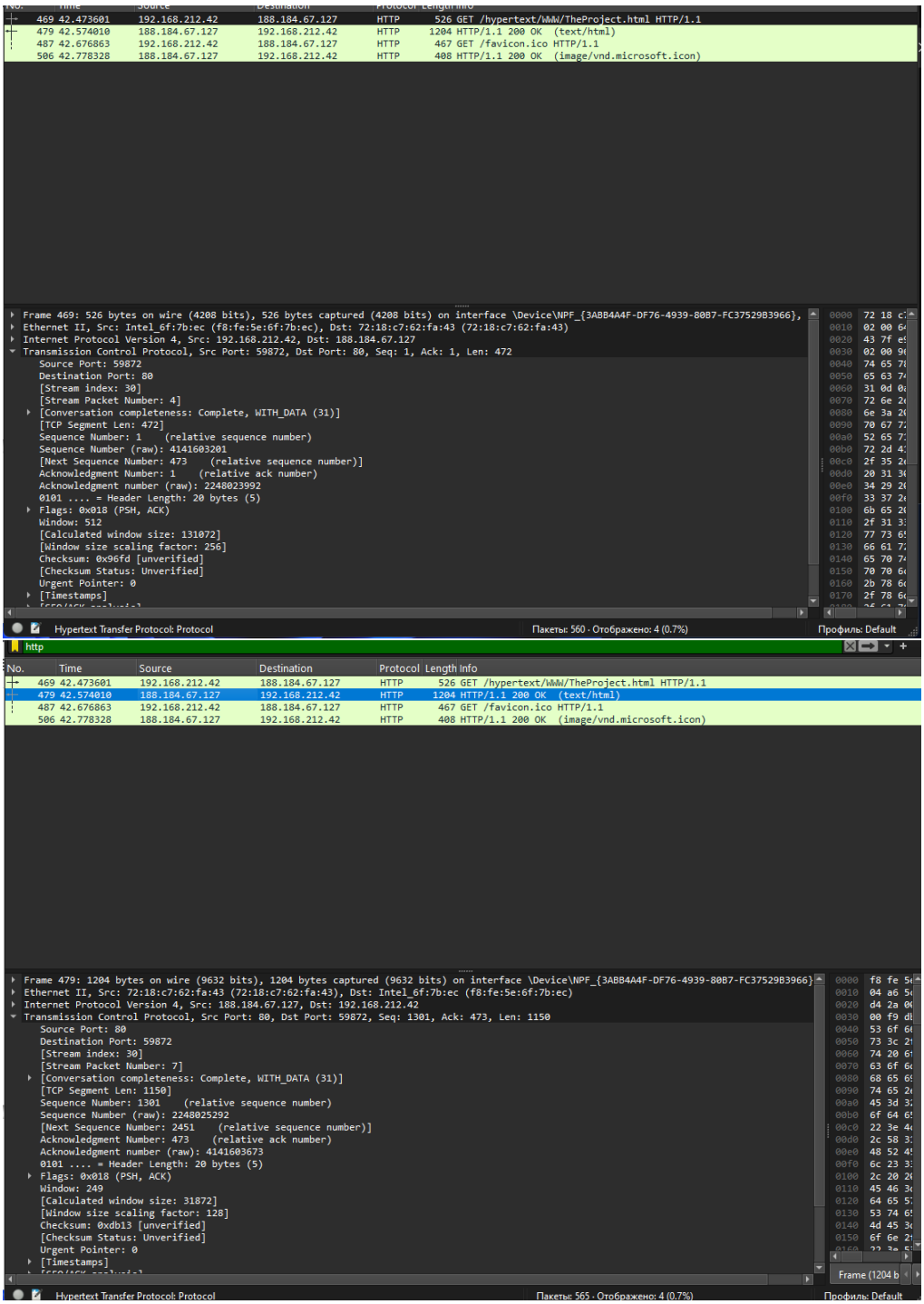
3. В Wireshark применён фильтр `http`. На экране отобразились HTTP-запросы и ответы, передаваемые по протоколу **TCP**:

    - Зафиксирован запрос `GET /hypertext/WWW/TheProject.html`.

    - Сервер ответил кодом `200 OK` и передал HTML-документ.

    - Дополнительно передавались файлы: `favicon.ico` и графические данные.

    - Для каждого пакета видны исходные и целевые порты TCP, порядковые и подтверждающие номера сегментов.

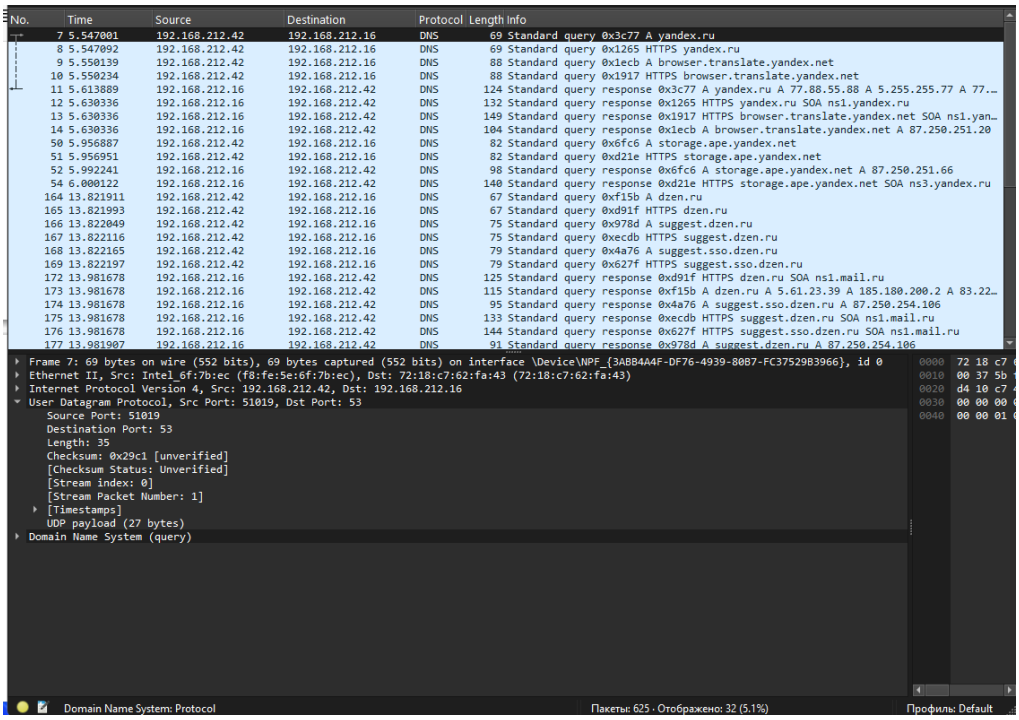


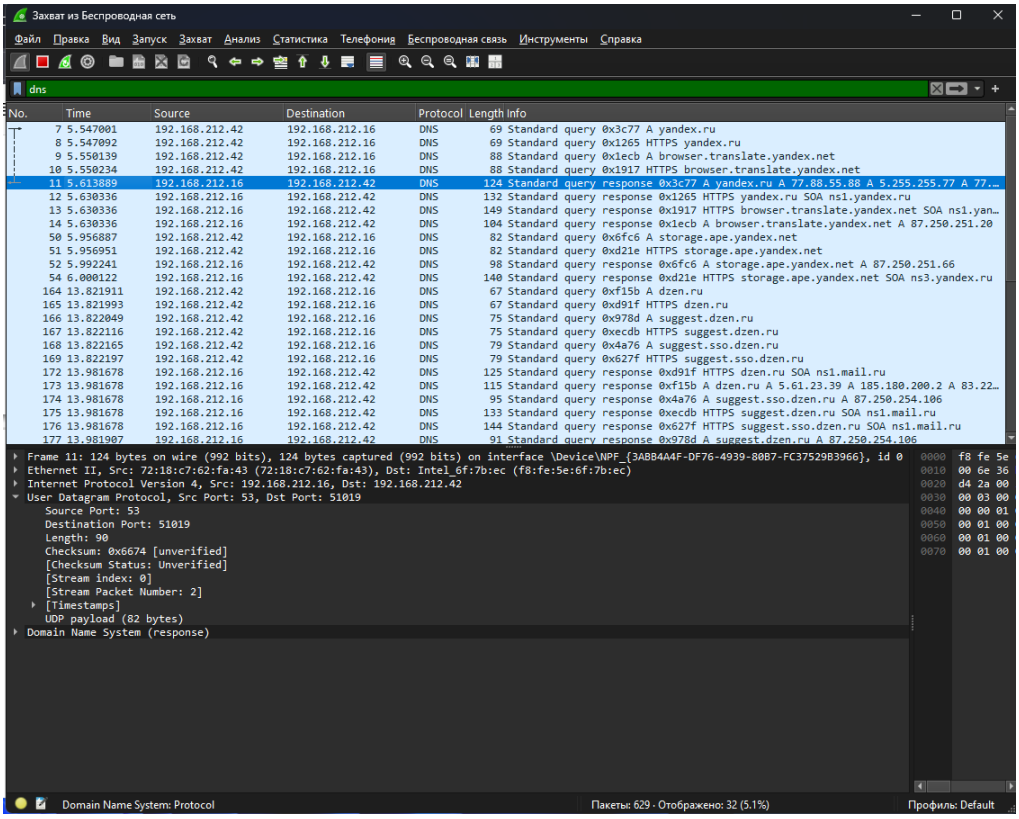
4. Далее был применён фильтр dns. В захвате зафиксированы **UDP-пакеты** с DNS-запросами и ответами:

    - Запросы на разрешение имени `yandex.ru` и других доменов.

- Ответы содержат несколько IP-адресов для одного доменного имени.

- Применялись стандартные порты: источник — динамический порт клиента, назначение — `53/UDP`.





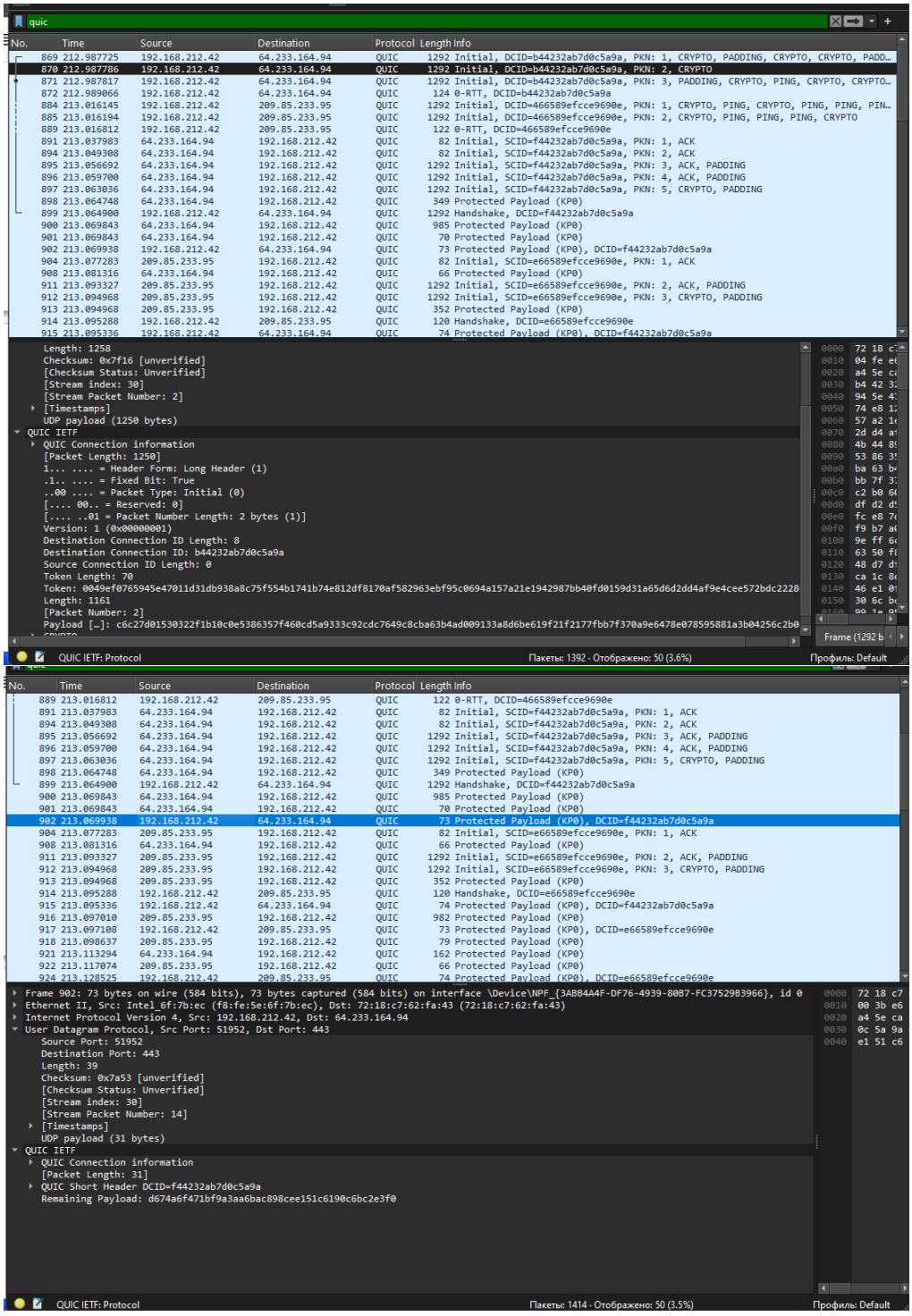
5. После этого был применён фильтр `quic`. В списке пакетов отобразились сессии по протоколу **QUIC** (поверх UDP):

    - Видны начальные пакеты соединения `Initial` с идентификаторами сессий.

    - Присутствуют зашифрованные сегменты `Protected Payload`.

    - Передача данных осуществляется через порт `443/UDP`.

    - QUIC обеспечивает функции транспортного уровня, аналогичные TCP, но с более высокой производительностью.



6. Захват трафика был завершён.

3.3 Анализ handshake протокола TCP в Wireshark

1. На устройстве был запущен **Wireshark**, выбран активный сетевой интерфейс и начат захват пакетов.

2. Для анализа установленного соединения было выполнено обращение к сайту по протоколу **HTTP**. В процессе обмена зафиксированы пакеты TCP, включая этап установки соединения (handshake).

3. Анализ TCP handshake:

   - Первое сообщение (SYN) инициирует соединение, устанавливается начальный порядковый номер (Sequence Number).

   - Второе сообщение (SYN, ACK) подтверждает приём SYN от клиента и устанавливает собственный порядковый номер.

   - Третье сообщение (ACK) завершает процесс трёхстороннего рукопожатия, соединение считается установленным.

   На скриншоте ниже виден процесс установления соединения по TCP, включая `SYN`, `SYN-ACK` и `ACK`:

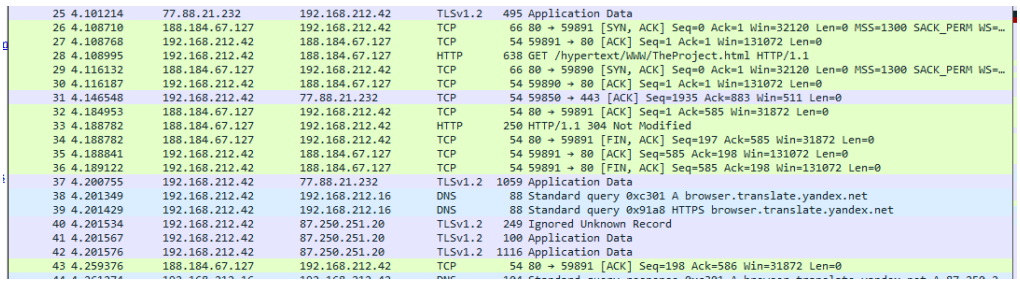


Рис. 3.7: TCP Handshake

4. В ходе обмена были также зафиксированы HTTP-запросы и ответы поверх установленного TCP-соединения. Например:

- `GET /hypertext/WWW/TheProject.html HTTP/1.1`

- Ответ сервера: `HTTP/1.1 304 Not Modified`

5. Для наглядности в меню **Статистика → График Потока** был построен график TCP-сессии. На нём отображается трёхстороннее рукопожатие:

   - Первый сегмент `SYN` от клиента.

   - Ответный сегмент `SYN, ACK` от сервера.

   - Заключительный сегмент `ACK` от клиента.

     После этого происходит передача HTTP-запроса и ответа.

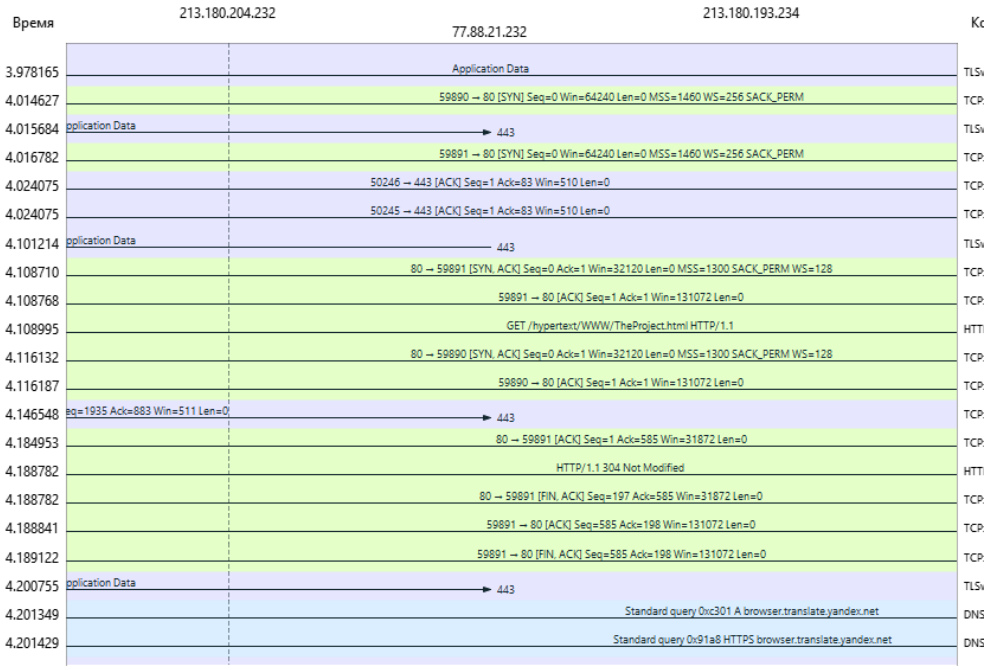


Рис. 3.8: График потока TCP

6. Захват трафика был остановлен.

4 Заключение

В ходе работы был проанализирован процесс установления соединения по протоколу TCP.
С помощью Wireshark зафиксированы пакеты трёхстороннего рукопожатия (SYN, SYN-ACK, ACK), а также последующая передача данных HTTP.