Сетевые технологии

Лабораторная работа №3

Тойчубекова Асель Нурлановна

Содержание

Список иллюстраций

Список таблиц

1 Цель работы

В этой лабораторной работе я изучу с помощью Wireshark кадры Ethernet и проанализирую PDU протоколов транспортного и прикладного уровней стека TCP/IP.

2 Теоретическое введение

Wireshark — это анализатор сетевого трафика, позволяющий исследовать работу сетевых протоколов и отслеживать обмен данными между устройствами. Он используется для диагностики сетевых неполадок, анализа безопасности и изучения структуры протоколов.

Одним из ключевых протоколов транспортного уровня является TCP (Transmission Control Protocol) — протокол, обеспечивающий надёжную передачу данных между клиентом и сервером. Установление соединения в TCP происходит в три этапа, называемых трёхступенчатым handshake:

Клиент отправляет запрос на соединение (SYN).

Сервер подтверждает запрос и предлагает свои параметры (SYN, ACK).

Клиент подтверждает установку соединения (ACK).

Другим современным транспортным протоколом является QUIC (Quick UDP Internet Connections), разработанный Google. Он работает поверх UDP, но обеспечивает надёжность и безопасность, аналогичную TCP и TLS. QUIC ускоряет установление соединения, объединяя процесс обмена криптографическими и транспортными параметрами, а также поддерживает шифрование каждого пакета и возможность смены сетевого пути без потери соединения.

3 Задание

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

4 Выполнение лабораторной работы

4.1 MAC-адресация

С помощью команды ipconfig выведим информацию о текущем сетевом соединении. В выводе видно, что активным является адаптер беспроводной сети (Wi-Fi), так как только у него указаны реальные IP-адреса.

IPv4-адрес: 172.16.48.177 — это адрес устройства в локальной сети.

Маска подсети: 255.255.254.0 — показывает диапазон доступных адресов в сети.

Основной шлюз: 172.16.48.1 — это адрес маршрутизатора, через который осуществляется выход в интернет.

Остальные адаптеры (например, OpenVPN, Ethernet, Bluetooth) находятся в состоянии «Среда передачи недоступна», то есть неактивны и не участвуют в сетевом обмене. (рис. 4.1 и рис. 4.2).

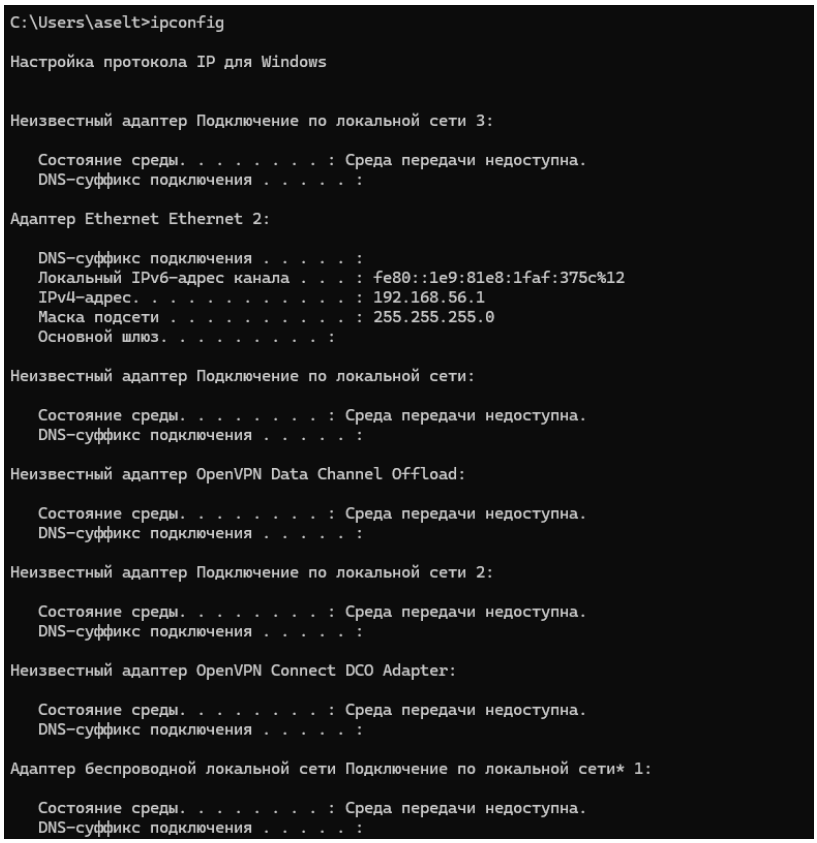

Рисунок 4.1: Сетевые настройки устройства

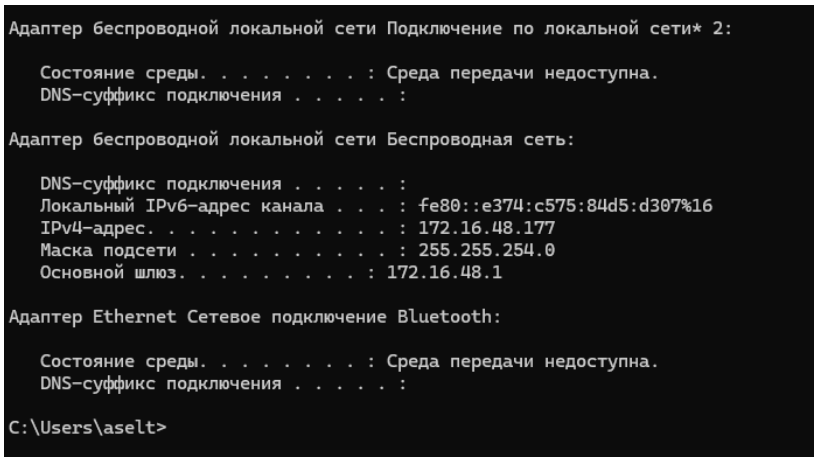

Рисунок 4.2: Сетевые настройки устройства

Далее с помощью команды ipconfig / all просмотрим более подробную информацию обо всех сетевых интерфейсов компьютера. Мы видим, что активный сетевой

адаптер (Wi-Fi): F4-6A-DD-79-EC-4D Также есть неактивные виртуальные адаптеры (VirtualBox, OpenVPN, Bluetooth и др.).

Структура MAC-адреса

MAC-адрес состоит из 6 байт (12 шестнадцатеричных цифр): F4-6A-DD | 79-EC-4D

F4-6A-DD — идентификатор производителя (OUI), принадлежит Realtek Semiconductor Corp.

79-EC-4D — уникальный номер сетевого интерфейса, присвоенный устройству производителем.

Определение типа адреса

Чтобы определить тип MAC-адреса, нужно посмотреть первый байт (в примере — F4):

В двоичном виде F4 = 11110100

Последний бит (справа налево второй) показывает, индивидуальный или групповой:

0 - индивидуальный (уникальный для устройства)

1 - групповой (многоадресный, multicast)

Второй бит справа налево показывает, глобально или локально администрируемый:

0 - глобально администрируемый (назначен производителем)

1 - локально администрируемый (изменён пользователем или программно)

Для F4 оба бита равны 0 ⟶ адрес индивидуальный и глобально администрируемый. (рис. 4.3, рис. 4.4, рис. 4.5).

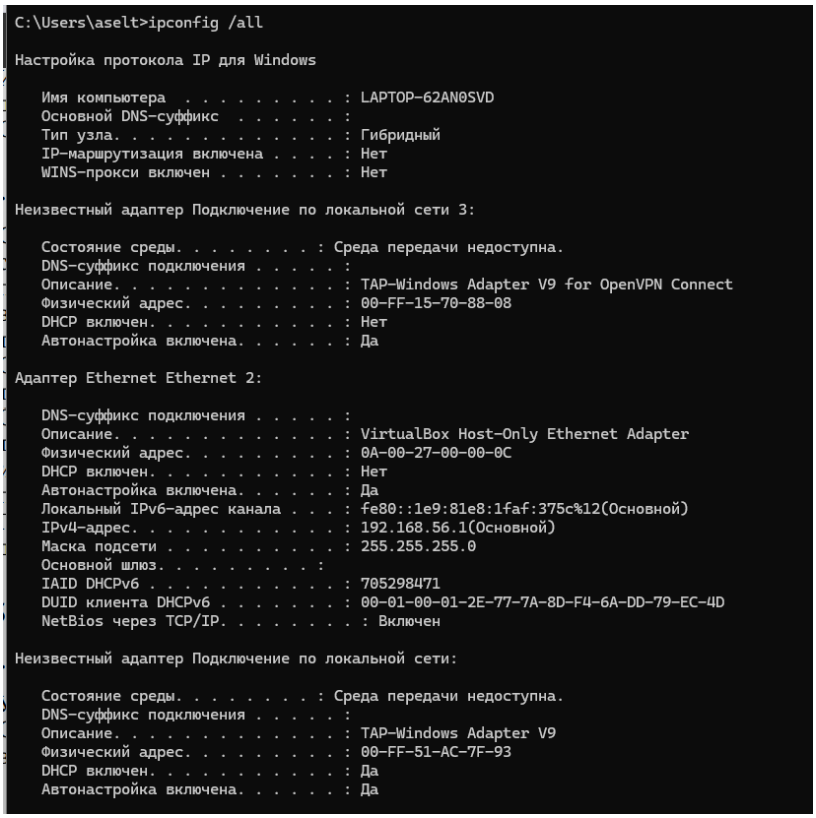

Рисунок 4.3: Информация о сетевых интерфейсов компьютера

Рисунок 4.4: Информация о сетевых интерфейсов компьютера

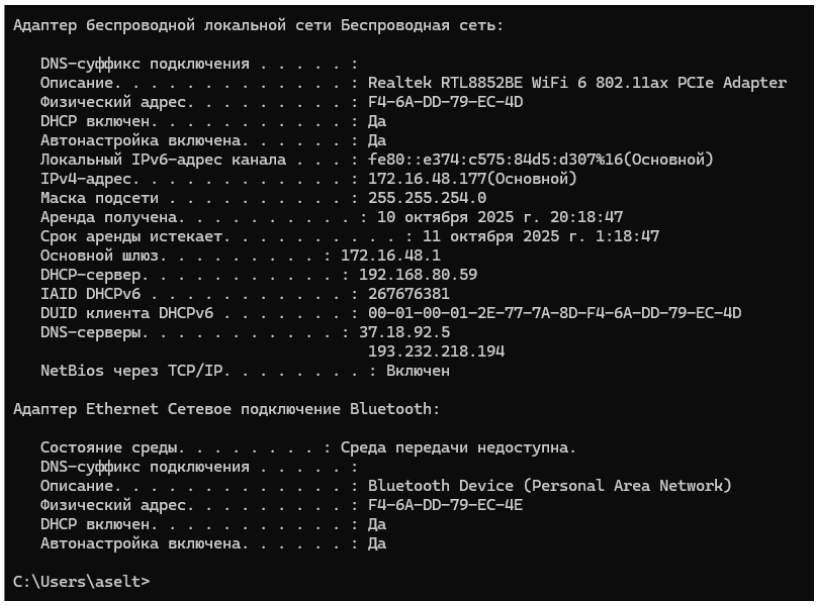

Рисунок 4.5: Информация о сетевых интерфейсов компьютера

4.2 Анализ кадров канального уровня в Wireshark

Установим на нашем компьютере Wireshark. (рис. 4.6).

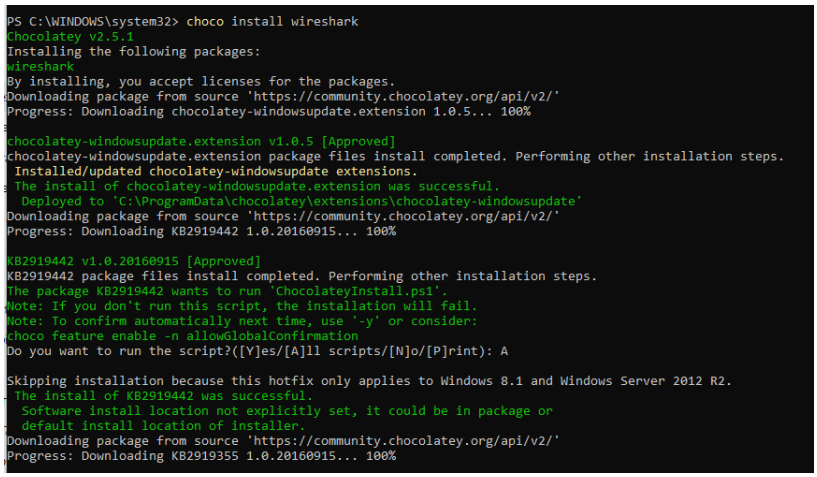

Рисунок 4.6: Установка Wireshark

Запустите Wireshark. Выберим активный на устройстве сетевой интерфейс. Убедимся, что начался процесс захвата трафика. (рис. 4.7).

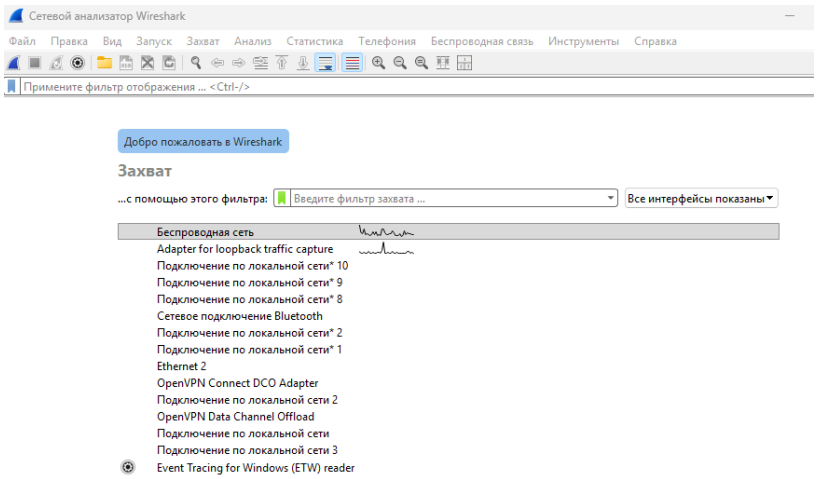

Рисунок 4.7: Установка Wireshark

На вашем устройстве в консоли определим с помощью команды ipconfig, IP-адрес нашего устройства и шлюз по умолчанию (default gateway). Мы видим, что наш IP-адрес= 172.16.48.177, а основной шлюз=172.16.48.1. (рис. 4.8).

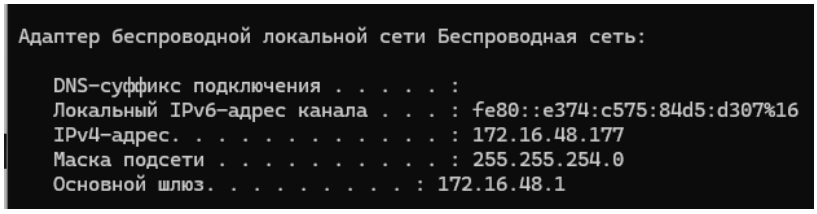

Рисунок 4.8: Сетевые настройки устройства

На нашем устройстве в консоли с помощью команды ping адрес_шлюза пропингуем шлюз по умолчанию. (рис. 4.9).

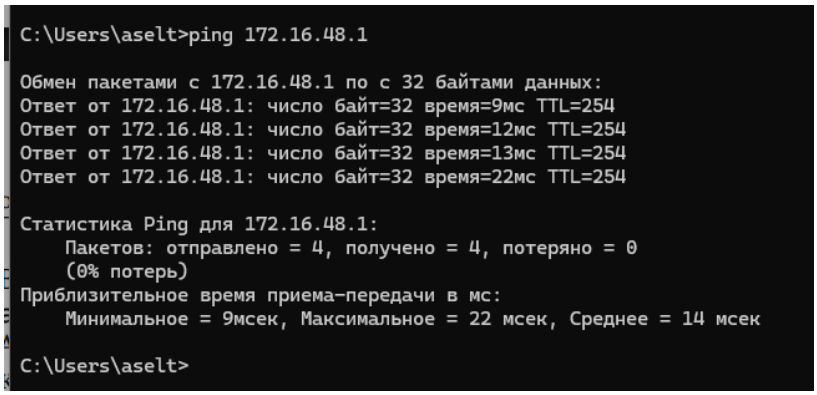

Рисунок 4.9: Проверка связи с Wi-fi

В Wireshark остановим захват трафика. В строке фильтра пропишем фильтр arp or icmp. Убедимся, что в списке пакетов отобразились только пакеты ARP или ICMP, в частности пакеты, которые были сгенерированы с помощью команды ping, отправленной с устройства на шлюз по умолчанию. (рис. 4.10).

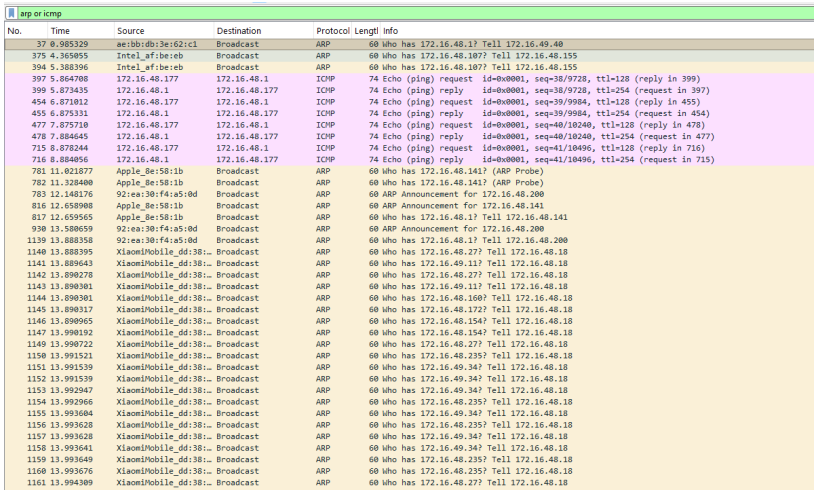

Рисунок 4.10: Просмотр пакетов ARP и ICMP

Просмотрим эхо-запрос ICMP в программе Wireshark.

Данные пакета (из панели сведений)

Длина кадра (Frame Length): 74 bytes (на проводе и захвачено).

Тип Ethernet: Ethernet II, Type: IPv4 (0x0800).

IP-адреса: источник 172.16.48.177, назначение (шлюз) 172.16.48.1.

MAC-адреса

MAC источника (ваш компьютер): f4:6a:dd:79:ec:4d (отмечено как LiteonTechno_79:ec:4d).

MAC назначения (шлюз/маршрутизатор): 70:18:07:60:9c:f8 (в Wireshark показан как устройство Cisco).

Определение типа MAC-адресов (коротко и как проверять)

Берём первый байт каждого MAC и смотрим его два младших бита (в двоичной записи):

Источник f4 - F4 hex = $1111\ 0100_2$

LSB (бит 0) = 0 - индивидуальный (unicast)

бит 1 (U/L) = 0 - глобально администрируемый (назначен производителем)- f4:6a:dd:79:ec:4d — индивидуальный, глобально администрируемый.

Назначение 70 - 70 hex = $0111\ 0000_2$

LSB = 0 - индивидуальный (unicast)

бит 1 = 0 - глобально администрируемый - 70:18:07:60:9c:f8 — индивидуальный, глобально администрируемый. (рис. 4.11).

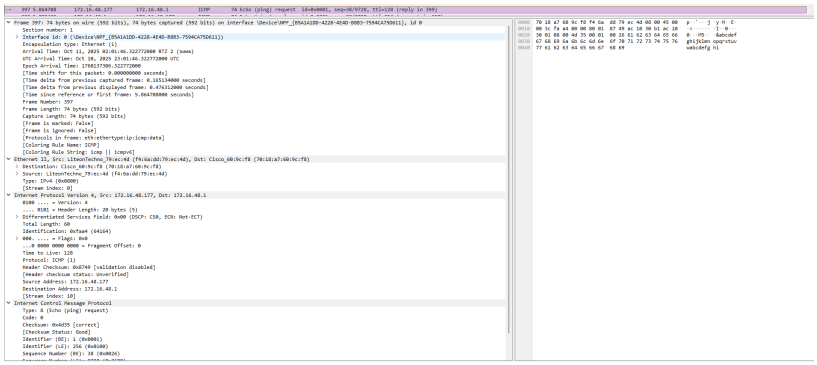

Рисунок 4.11: ICMP-запрос

Теперь просмотрим ICMP-ответ.

Данные пакета

Frame length: 74 bytes (на проводе и захвачено).

Тип Ethernet: Ethernet II, Type: IPv4 (0x0800).

IP-адреса:

Источник (шлюз): 172.16.48.1

Назначение (ваш компьютер): 172.16.48.177

MAC-адреса

MAC источника (шлюз): 70:18:a7:60:9c:f8 (Cisco) - 70 (hex → 0111 0000_2) - индивидуальный,глобально администрируемый

MAC назначения (ваш компьютер): f4:6a:dd:79:ec:4d (LiteonTechno) - f4 (hex → 1111 0100_2) -индивидуальный,глобально администрируемый. (рис. 4.12).

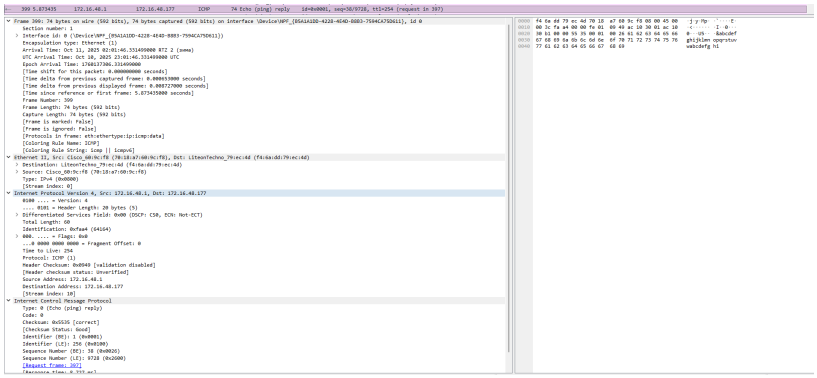

Рисунок 4.12: ICMP-ответ

Изучим кадры данных протокола ARP. Изучим данные в полях заголовка Ethernet II.

Кадр ARP (Frame 375):

Длина кадра: 60 байт (480 бит)

Тип Ethernet: Ethernet II

MAC-адрес источника: f4:7b:09:af:be:eb (сетевой интерфейс отправителя)

MAC-адрес назначения: ff:ff:ff:ff:ff:ff (широковещательный — Broadcast)

Протокол в кадре: ARP (0x0806)

IP-адрес источника: 172.16.48.155

IP-адрес назначения: 172.16.48.107

Тип MAC-адресов: индивидуальные (у источника), широковещательный (у назначения)

Назначение кадра: запрос ARP, цель — определить MAC-адрес устройства с IP 172.16.48.107

Пояснение:

В ARP-запросе источник знает только IP назначения, поэтому отправляет кадр на Broadcast.

MAC-адрес источника уникален и глобально администрируемый (первые три байта f4:7b:09 идентифицируют производителя Intel).

MAC-адрес назначения — широковещательный, потому что ARP-запрос адресован всем устройствам в локальной сети. (рис. 4.13).

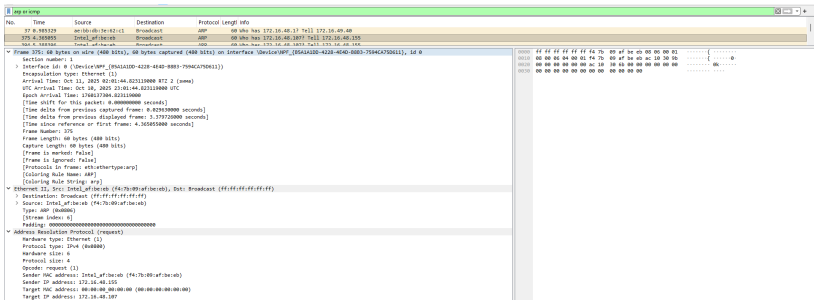

Рисунок 4.13: Протокол ARP

Начнем новый процесс захвата трафика в Wireshark. На устройстве в консоли пропингуем по имени какой-нибудь известный вам адрес, habr.com. (рис. 4.14).

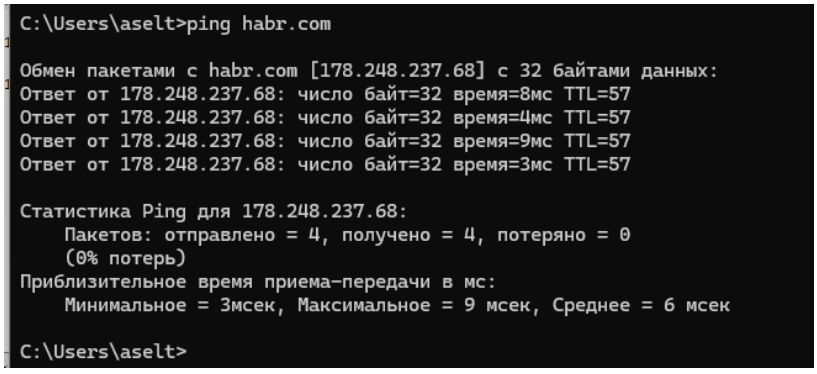

Рисунок 4.14: Установка связи с habr.com

В Wireshark остановим захват трафика. Изучим протокол ARP.

Данные кадра

Номер кадра: 148

Длина кадра (Frame Length): 42 bytes (336 bits)

Тип Ethernet: Ethernet II, Type: ARP (0x0806)

IP-адрес отправителя: 172.16.48.51

IP-адрес цели (запрашиваемый): 172.16.48.146

MAC-адреса

MAC источника (Sender MAC): 20:04:84:63:46:95 (Wireshark показывает как Apple_63:46:95)-индивидуальный, глобально администрируемый.

MAC назначения (Destination): ff:ff:ff:ff:ff:ff — Broadcast (широковещательный) - широковещательный (broadcast)

Кадр 148 — это ARP-запрос: устройство с MAC 20:04:84:63:46:95 и IP 172.16.48.51 посылает широковещательный запрос Who has 172.16.48.146? Tell 172.16.48.51, чтобы узнать MAC-адрес хоста с IP 172.16.48.146. MAC отправителя — индивидуальный и глобально администрируемый; MAC назначения — широковещательный (групповой). (рис. 4.15).

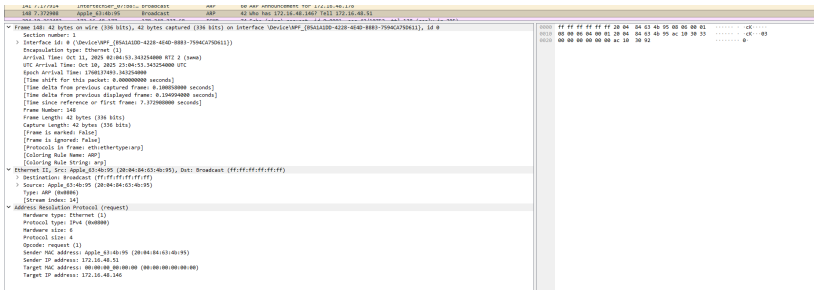

Рисунок 4.15: Протокол ARP

Изучим запросы протокола ICMP.

Основные данные пакета

Номер кадра: 204

Длина кадра (Frame Length): 74 bytes (592 bits)

Тип Ethernet: Ethernet II, Type: IPv4 (0x0800)

IP-адреса: источник 172.16.48.177, назначение 178.248.237.68 (пинг наружному хосту)

MAC-адреса (Ethernet II)

MAC источника (ваш компьютер): f4:6a:dd:79:ec:4d (Wireshark показывает как LiteonTechno_79:ec:4d)- — индивидуальный, глобально администрируемый.

MAC назначения (шлюз/маршрутизатор): 70:18:27:60:9c:f8 (помечен как Cisco_60:9c:f8) — индивидуальный, глобально администрируемый. (рис. 4.16).

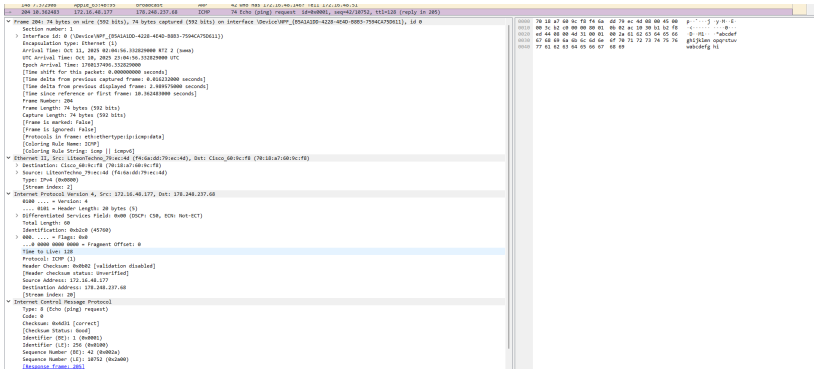

Рисунок 4.16: ICMP-запрос

Изучим ответы протокола ICMP.

Основные данные

Длина кадра (Frame Length): 74 bytes.

Тип Ethernet: Ethernet II, Type: IPv4 (0x0800).

IP-адреса: источник 178.248.237.68, назначение 172.16.48.177.

MAC-адреса (Ethernet II)

MAC источника: 70:18:37:60:9c:f8 (Wireshark показывает как Cisco_60:9c:f8) — это MAC шлюза/маршрутизатора/удалённого устройства, от которого пришёл ответ. - индивидуальные (unicast) и глобально администрируемые.

MAC назначения: f4:6a:dd:79:ec:4d (Wireshark показывает как LiteonTechno_79:ec:4d) — это MAC вашего компьютера. - индивидуальные (unicast) и глобально администрируемые. (рис. 4.17).

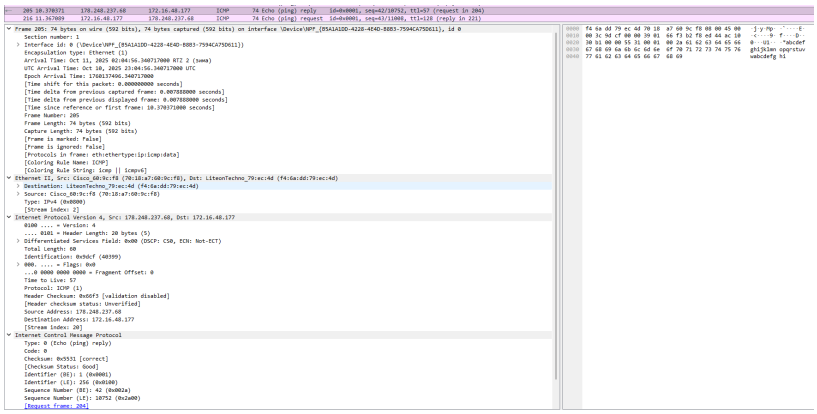

Рисунок 4.17: ICMP-ответ

4.3 Анализ протоколов транспортного уровня в Wireshark

Выберим активный на устройстве сетевой интерфейс. Убедимся, что начался процесс захвата трафика. На устройстве в браузере перейдем на сайт, работающий по протоколу HTTP, http://info.cern.ch/. По перемещаемся по ссылкам или разделам сайта в браузере. (рис. 4.18).

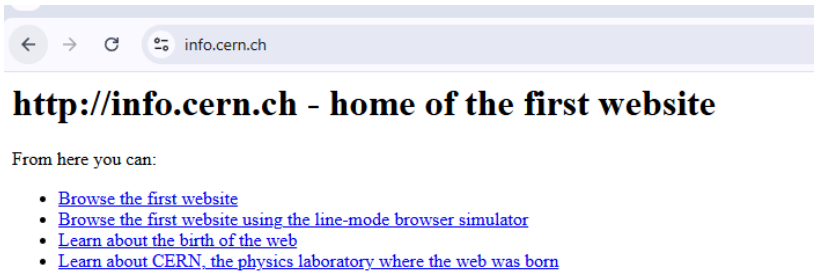

Рисунок 4.18: Сайт http://info.cern.ch/

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

Кадр 751 (HTTP-запрос по протоколу TCP)

Длина кадра: 652 байта

Тип Ethernet: Ethernet II (IPv4 – 0x0800)

MAC-адрес источника: f4:6a:dd:79:ec:4d (LiteonTechno_79:ec:4d)

MAC-адрес получателя: 70:18:a7:60:9c:f8 (Cisco_60:9c:f8)

Тип MAC-адресов: индивидуальные, глобально администрируемые

Информация по TCP:

Протокол уровня транспорта: TCP (6)

Порт источника: 62398

Порт назначения: 80 (HTTP)

Флаги: PSH, ACK — данные передаются и подтверждаются

Номер последовательности (Seq): 1

Номер подтверждения (Ack): 1

Размер окна: 65280

Длина TCP-сегмента: 598 байт

Тип данных: HTTP-запрос GET /hypertext/DataSources/byOrganisation/Overview.html HTTP/1.1

Кадр представляет собой TCP-сегмент, содержащий HTTP-запрос от клиента (IP: 172.16.48.177) к серверу (IP: 188.184.67.127) на порт 80. Клиент инициировал передачу данных с подтверждением получения предыдущих пакетов (ACK). Использование флага PSH указывает на немедленную передачу данных приложения (HTTP-запроса). (рис. 4.19).

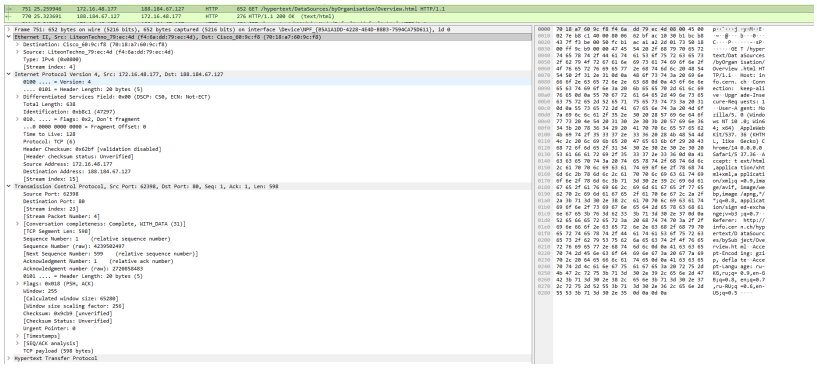

Рисунок 4.19: HTTP-запрос

Теперь проанализируем информацию по протоколу TCP в случае ответов.

Кадр 770 (HTTP-ответ по протоколу TCP)

Длина кадра: 276 байт

Тип Ethernet: Ethernet II (IPv4 – 0x0800)

MAC-адрес источника: 70:18:a7:60:9c:f8 (Cisco_60:9c:f8)

MAC-адрес получателя: f4:6a:dd:79:ec:4d (LiteonTechno_79:ec:4d)

Тип MAC-адресов: индивидуальные, глобально администрируемые

Информация по TCP:

Протокол уровня транспорта: TCP (6)

Порт источника: 80 (HTTP)

Порт назначения: 62398

Флаги: PSH, ACK — данные передаются и подтверждаются

Номер последовательности (Seq): 2921

Номер подтверждения (Ack): 599

Размер окна: 31872

Длина TCP-сегмента: 222 байта

Тип данных: HTTP-ответ HTTP/1.1 200 OK (text/html)

Кадр представляет собой TCP-сегмент с HTTP-ответом от сервера (IP: 188.184.67.127) клиенту (IP: 172.16.48.177). Использование флага PSH указывает на немедленную передачу данных приложения (тело ответа HTTP). ACK подтверждает получение предыдущих сегментов TCP. Этот сегмент является частью установленного TCP-соединения для передачи HTTP-данных. (рис. 4.20).

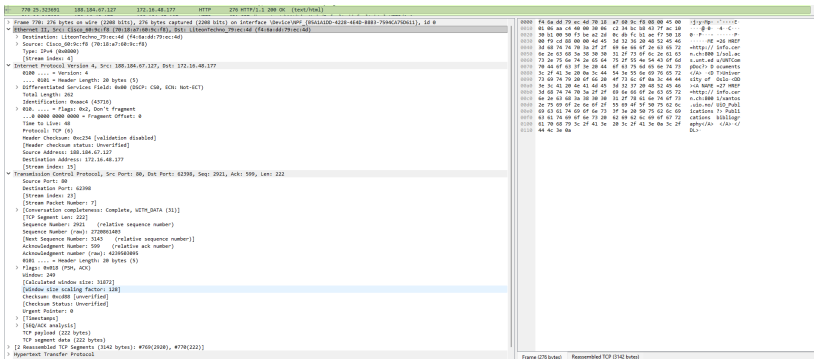

Рисунок 4.20: HTTP-ответ

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

Кадр 419 (DNS-запрос по протоколу UDP)

Длина кадра: 79 байт

Тип Ethernet: Ethernet II (IPv4 – 0x0800)

MAC-адрес источника: f4:6a:dd:79:ec:4d (LiteonTechno_79:ec:4d)

MAC-адрес получателя: 70:18:a7:60:9c:f8 (Cisco_60:9c:f8)

Тип MAC-адресов: индивидуальные, глобально администрируемые

Информация по UDP:

Протокол уровня транспорта: UDP (17)

Порт источника: 64394

Порт назначения: 53 (DNS)

Длина UDP-пакета: 45 байт

Информация по DNS:

Тип запроса: стандартный DNS-запрос

Запрашиваемое доменное имя: accounts.google.com

Тип записи: A (IP-адрес)

Назначение запроса: разрешение доменного имени в IP-адрес

Кадр представляет собой DNS-запрос от клиента (IP: 172.16.48.177) к DNS-серверу (IP: 37.18.92.5) по UDP. Клиент запрашивает IP-адрес для домена accounts.google.com.

Использование UDP позволяет быстро передавать небольшие запросы без установки соединения, что характерно для протокола DNS. (рис. 4.21).

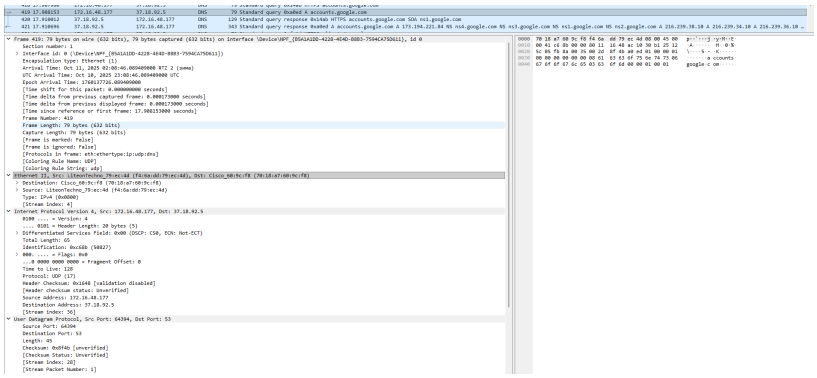

Рисунок 4.21: DNS-запрос

Проанализируем информацию по протоколу UDP в случае ответов.

Кадр 420 (DNS-ответ по протоколу UDP)

Длина кадра: 129 байт

Тип Ethernet: Ethernet II (IPv4 – 0x0800)

MAC-адрес источника: 70:18:37:60:9c:f8 (Cisco_60:9c:f8)

MAC-адрес получателя: f4:6a:dd:79:ec:4d (LiteonTechno_79:ec:4d)

Тип MAC-адресов: индивидуальные, глобально администрируемые

Информация по UDP:

Протокол уровня транспорта: UDP (17)

Порт источника: 53 (DNS)

Порт назначения: 51234

Длина UDP-пакета: 95 байт

Информация по DNS:

Тип ответа: стандартный DNS-ответ

Ответ на запрос домена: accounts.google.com

Типы записей:

A-запись: 173.194.221.84 (IP-адрес)

NS-запись: ns4.google.com

Цель ответа: разрешение доменного имени в IP-адрес для клиента

Кадр представляет собой DNS-ответ от сервера (IP: 37.18.92.5) клиенту (IP: 172.16.48.177). Сервер возвращает IP-адрес и данные NS для запрошенного домена accounts.google.com. Использование UDP позволяет передавать быстрые ответы без установления соединения. (рис. 4.22).

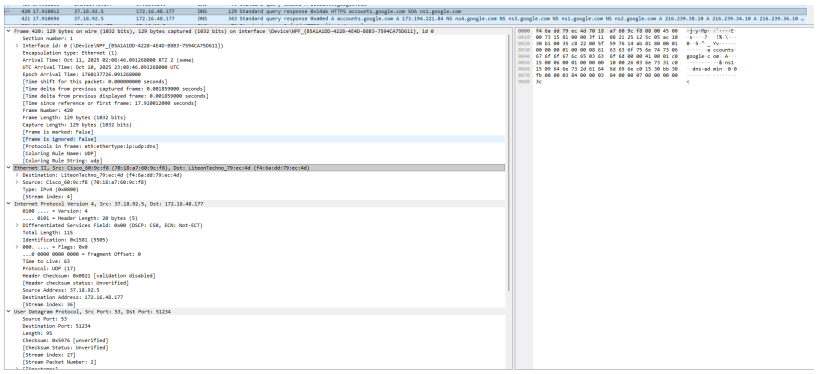

Рисунок 4.22: DNS-ответ

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

Кадр 617 — QUIC-запрос:

Длина кадра: 1292 байта

MAC-адрес источника: f4:6a:dd:79:ec:4d (LiteonTechno_79:ec:4d)

MAC-адрес получателя: 70:18:27:60:9c:f8 (Cisco_60:9c:f8)

Тип MAC-адресов: индивидуальные, глобально администрируемые

IP-адрес источника: 172.16.48.177

IP-адрес получателя: 142.250.74.142

Протокол уровня транспорта: UDP (17)

Порт источника: 54703

Порт назначения: 443 (HTTPS/QUIC)

Тип пакета QUIC: Initial

Содержимое: CRYPTO, PING, PADDING, DCID, PIN — инициирование защищённого соединения, обмен криптографическими данными, подтверждение доступности.

(рис. 4.23).

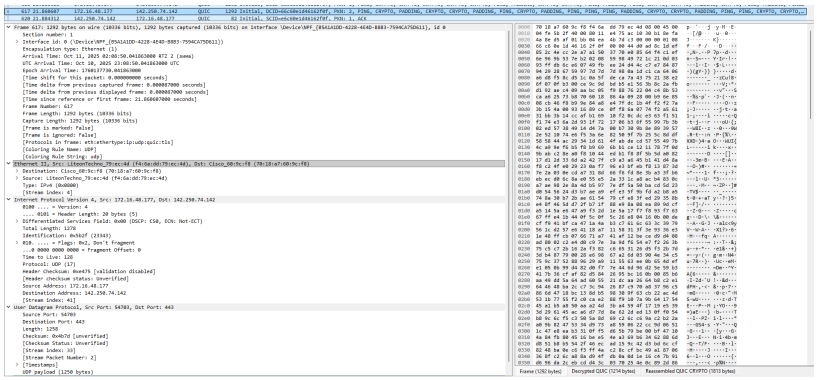

Рисунок 4.23: QUIC -запрос

Проанализируем информацию по протоколу quic в случае ответов.

Кадр 620 — QUIC-ответ (Initial, ACK):

Длина кадра: 82 байта

MAC-адрес источника: 70:18:a7:60:9c:f8 (Cisco_60:9c:f8)

MAC-адрес получателя: f4:6a:dd:79:ec:4d (LiteonTechno_79:ec:4d)

IP-адрес источника: 142.250.74.142

IP-адрес получателя: 172.16.48.177

Протокол транспорта: UDP (17)

Порт источника: 443

Порт назначения: 54703

Содержимое QUIC: Initial пакет с ACK, содержит SCID (Server Connection ID) и PKN=1 — подтверждение получения первого пакета от клиента, начало обмена ключами.

Кадр 621 — QUIC-ответ (Initial, ACK, PADDING):

Длина кадра: 1292 байта

MAC-адрес источника: 70:18:a7:60:9c:f8 (Cisco_60:9c:f8)

MAC-адрес получателя: f4:6a:dd:79:ec:4d (LiteonTechno_79:ec:4d)

IP-адрес источника: 142.250.74.142

IP-адрес получателя: 172.16.48.177

Протокол транспорта: UDP (17)

Порт источника: 443

Порт назначения: 54703

Содержимое QUIC: Initial пакет с ACK и PADDING, PKN=2 — подтверждение второго шага и выравнивание пакета для защиты от анализа трафика, продолжение криптографического обмена.

QUIC использует UDP для передачи зашифрованного трафика между клиентом и сервером HTTPS.

Кадры 620 и 621 — ответы сервера на запрос клиента, включающие подтверждения приёма и продолжение безопасного обмена данными. (рис. 4.24).

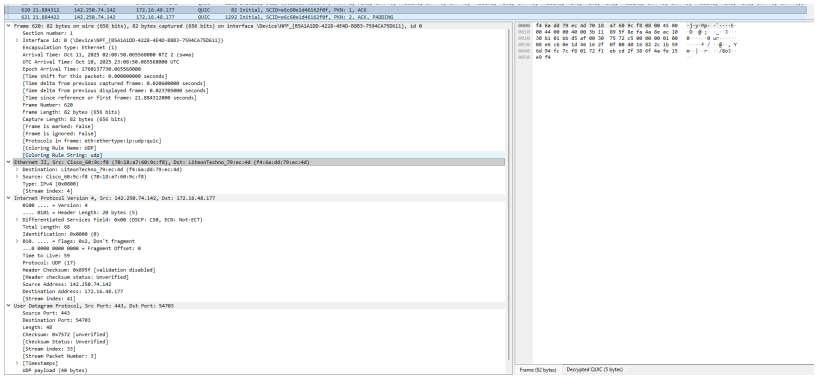

Рисунок 4.24: QUIC -ответ

4.4 Анализ handshake протокола TCP в Wireshark

Выберем активный на устройстве сетевой интерфейс. Убедимся, что начался процесс захвата трафика.

На нашем устройстве в браузере перейдем вновь на сайт CERN http://info.cern.ch/, работающий по протоколу http,ля захвата в Wireshark пакетов TCP. (рис. 4.25).

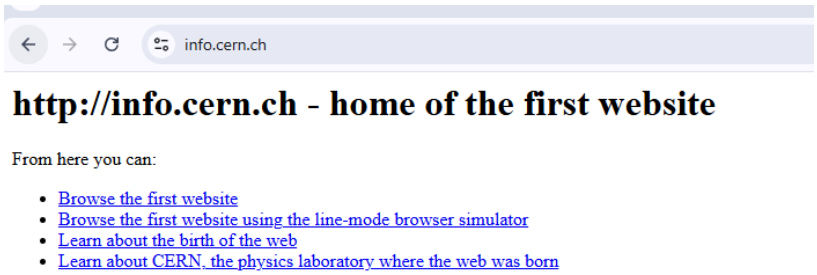

Рисунок 4.25: Сайт Cern

В Wireshark проанализируем handshake протокола TCP.

Кадр 307 — SYN (инициализация соединения от клиента)

Источник: 172.16.48.177

Назначение: 188.184.67.127

Порт источника: 56113

Порт назначения: 80

Флаги TCP: SYN (0x002)

Последовательный номер (Seq): 0

Размер окна (Window): 65535

Опции TCP:

MSS=1468 — максимальный размер сегмента

WS=256 — масштаб окна

SACK permitted — разрешение SACK

Смысл: Клиент инициирует соединение, сообщая серверу свой начальный Seq номер. (рис. 4.26).

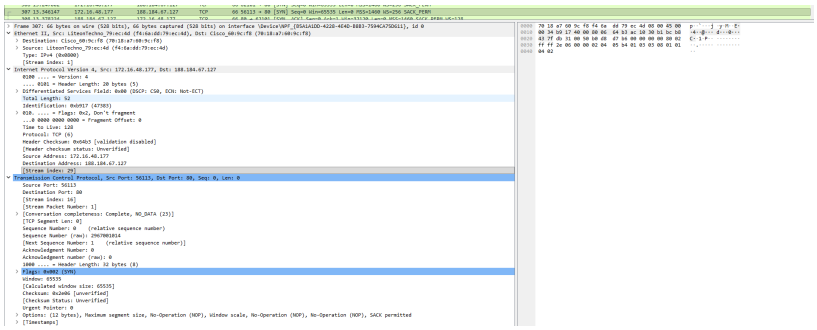

Рисунок 4.26: SYN

Кадр 308 — SYN+ACK (ответ сервера)

Источник: 188.184.67.127

Назначение: 172.16.48.177

Порт источника: 80

Порт назначения: 62101 (соответствует клиентскому порту + NAT/локальный)

Флаги TCP: SYN, ACK (0x012)

Seq сервера: 0

Ack: 1 — подтверждение Seq клиента + 1

Размер окна: 32120

Опции TCP: MSS=1460, WS=128, SACK permitted

Смысл: Сервер принимает запрос клиента и подтверждает его Seq, одновременно отправляя свой SYN для установления двустороннего соединения. (рис. 4.27).

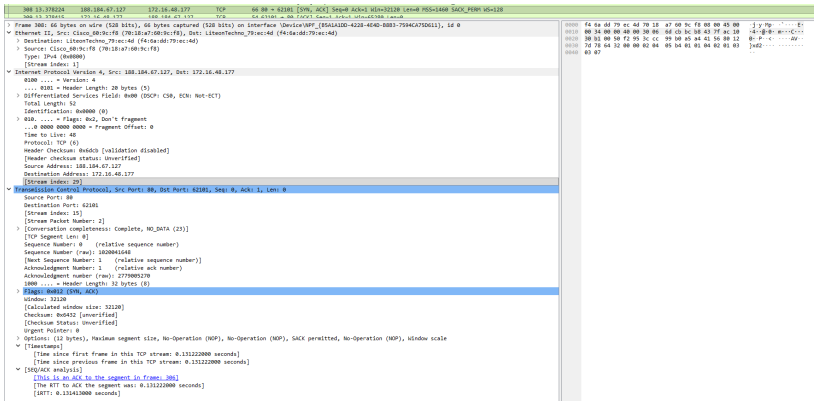

Рисунок 4.27: SYN+ACK

Кадр 309 — ACK (подтверждение клиентом)

Источник: 172.16.48.177

Назначение: 188.184.67.127

Порт источника: 62101

Порт назначения: 80

Флаги TCP: ACK (0x010)

Seq: 1 — следующий Seq клиента

Ack: 1 — подтверждение Seq сервера + 1

Смысл: Клиент подтверждает получение SYN+ACK от сервера. (рис. 4.28).

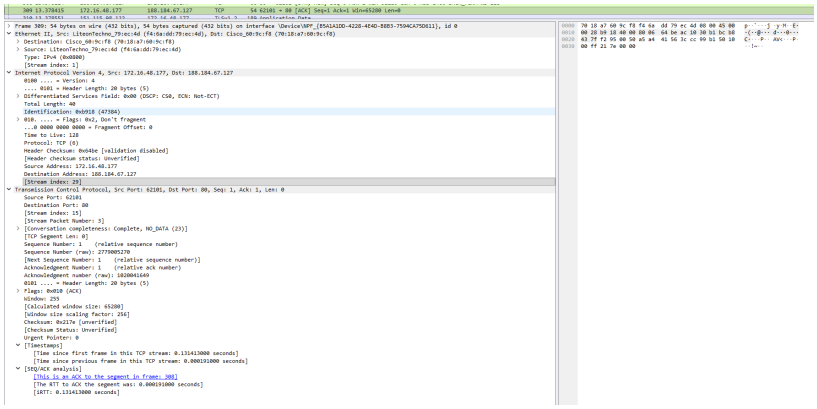

Рисунок 4.28: ACK

В Wireshark в меню «Статистика» выберем «График Потока». Рассмотрим процесс установлении соединения по TCP.

Seq и Ack показывают согласование номеров последовательностей, обеспечивая надежную доставку данных.

Флаги SYN/ACK изменяются от SYN → SYN+ACK → ACK, что является стандартной трехсторонней рукопожатой процедурой TCP.

Размер окна (Window) указывает на емкость буфера приема каждого участника и может масштабироваться (WS) для высокопроизводительных соединений.

SYN (клиент → сервер)

Сообщение: TCP 56448 → 80 [SYN] Seq=0 Win=65535 Len=0

ACK: отсутствует, так как это первый сегмент.

Win: 65535 — размер окна, сколько байт клиент готов принять.

SYN-ACK (сервер ⟶ клиент)

Сообщение: TCP 80 ⟶ 56448 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0

ACK: 1 — подтверждает получение SYN от клиента.

Win: 8192 — сколько байт сервер готов принять от клиента.

ACK (клиент ⟶ сервер)

Сообщение: TCP 56448 ⟶ 80 [ACK] Seq=1 Ack=1 Win=65280 Len=0

ACK: 1 — подтверждает получение SYN сервера.

Win: 65280 — обновленный размер окна, сколько клиент готов принять. (рис. 4.29).

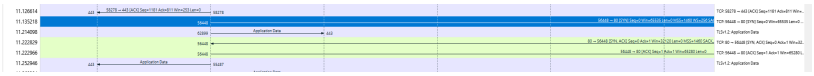

Рисунок 4.29: График Потока

5 Выводы

В ходе выполнения лабораторной работы №3 я изучила с помощью Wireshark кадры Ethernet и проанализировала PDU протоколов транспортного и прикладного уровней стека TCP/IP.

Список литературы

1. Barr D. Common DNS Operational and Configuration Errors: RFC / RFC Editor. —02/1996. — DOI: 10.17487/rfc1912.
2. Security-Enhanced Linux. Linux с улучшенной безопасностью: руководство пользователя / M. McAllister, S. Radvan, D. Walsh, D. Grift, E. Paris, J. Morris. — URL: https://docs-old.fedoraproject.org/ru-RU/Fedora/13/html/Security-Enhanced_Linux/index.html.
3. Systemd. — 2015. — URL: https:/ /wiki .archlinux .org /index .php /Systemd.
4. Костромин В. А. Утилита lsof — инструмент администратора. — URL: http : / / rus linux.net/kos.php?name=/papers/lsof/lsof.html