

# **Лабораторная работа №3**

**Дисциплина: Сетевые технологии**

Комягин Андрей Николаевич

# Содержание

<b>1</b>	<b>Цель</b>	<b>5</b>
1.1	Цель работы . . . . .	5
<b>2</b>	<b>Ход работы</b>	<b>6</b>
2.1	MAC-адресация . . . . .	6
2.2	Анализ кадров канального уровня в Wireshark . . . . .	7
2.2.1	Анализ ICMP-трафика . . . . .	7
2.2.2	Анализ ARP-трафика . . . . .	8
2.3	Анализ протоколов транспортного уровня . . . . .	10
2.3.1	Анализ HTTP . . . . .	10
2.3.2	Анализ DNS . . . . .	12
2.3.3	Анализ QUIC . . . . .	13
2.4	Анализ handshake протокола TCP . . . . .	14
2.4.1	handshake . . . . .	14
2.4.2	График Потока . . . . .	17
<b>3</b>	<b>Выводы</b>	<b>18</b>
	<b>Список литературы</b>	<b>19</b>

# Список иллюстраций

2.1	ipconfig . . . . .	6
2.2	ICMP 2356 . . . . .	7
2.3	ICMP 2357 . . . . .	8
2.4	ARP 81 . . . . .	9
2.5	APR 82 . . . . .	10
2.6	HTTP GET Request (152) . . . . .	11
2.7	HTTP 200 OK Response (155) . . . . .	11
2.8	DNS Standard Query (63) . . . . .	12
2.9	DNS Standard Query Response (64) . . . . .	13
2.10	QUIC . . . . .	14
2.11	QUIC . . . . .	14
2.12	syn . . . . .	15
2.13	syn, ack . . . . .	16
2.14	ack . . . . .	17
2.15	График Потока . . . . .	17

## **Список таблиц**

# **1 Цель**

## **1.1 Цель работы**

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

## 2 Ход работы

### 2.1 MAC-адресация

**С помощью команды `ipconfig` определить основные параметры сетевого соединения.**

Для определения текущих сетевых настроек была использована команда `ipconfig` в консоли Windows.

```
Адаптер беспроводной локальной сети Беспроводная сеть:  
  
DNS-суффикс подключения . . . . . :  
Локальный IPv6-адрес канала . . . : fe80::28f2:9ef4:cafe:7d81%6  
IPv4-адрес. . . . . : 192.168.0.138  
Маска подсети . . . . . : 255.255.255.0  
Основной шлюз. . . . . : 192.168.0.1
```

Рис. 2.1: `ipconfig`

Из полученных данных были определены ключевые параметры адаптера беспроводной сети:

- IPv4-адрес: 192.168.0.138 — текущий IP-адрес моего устройства в локальной сети.
- Маска подсети: 255.255.255.0 — определяет, какая часть IP-адреса относится к сети, а какая — к узлу.
- Основной шлюз: 192.168.0.1 — IP-адрес маршрутизатора (роутера), через который осуществляется выход в другие сети, включая Интернет.

MAC-адрес моего устройства (2c:6d:c1:60:d8:d0) был определен в ходе последующего анализа трафика в Wireshark.

## 2.2 Анализ кадров канального уровня в Wireshark

**Захватить и проанализировать пакеты ARP и ICMP в части кадров канального уровня (Ethernet II).**

Для генерации трафика была выполнена команда `ping 192.168.0.1` (ping основного шлюза). В Wireshark был применен фильтр **arp or icmp**.

### 2.2.1 Анализ ICMP-трафика

На скриншоте ниже виден обмен ICMP-пакетами (эхо-запросы и эхо-ответы).

#### 1. ICMP Echo (ping) Request (пакет №2356):

Описание: Мое устройство (192.168.0.138) отправляет эхо-запрос на основной шлюз (192.168.0.1).

Заголовок Ethernet II:

Source MAC: Intel\_60:d8:d0 (2c:6d:c1:60:d8:d0) (мой ПК)

Destination MAC: TpLinkTechno\_59:88:0b (28:ee:52:59:88:0b) (мой роутер)

Type: IPv4 (0x0800)

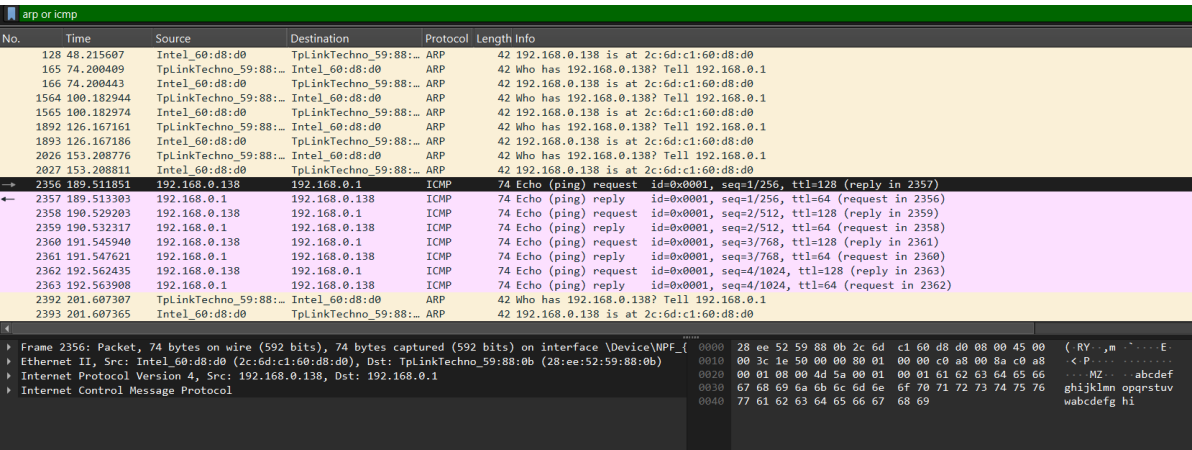


Рис. 2.2: ICMP 2356

## 2. ICMP Echo (ping) Reply (пакет №2357):

Описание: Основной шлюз (192.168.0.1) отвечает на запрос, подтверждая свою доступность.

Заголовок Ethernet II:

Source MAC: TpLinkTechno\_59:88:0b (мой роутер)

Destination MAC: Intel\_60:d8:d0 (мой ПК)

Type: IPv4 (0x0800)

2356	189.511851	192.168.0.138	192.168.0.1	ICMP	74 Echo (ping) request	id=0x0001, seq=1/256, ttl=128 (reply in 2357)
2357	189.513303	192.168.0.1	192.168.0.138	ICMP	74 Echo (ping) reply	id=0x0001, seq=1/256, ttl=64 (request in 2356)
2358	190.529203	192.168.0.138	192.168.0.1	ICMP	74 Echo (ping) request	id=0x0001, seq=2/512, ttl=128 (reply in 2359)
2359	190.532317	192.168.0.1	192.168.0.138	ICMP	74 Echo (ping) reply	id=0x0001, seq=2/512, ttl=64 (request in 2358)
2360	191.545940	192.168.0.138	192.168.0.1	ICMP	74 Echo (ping) request	id=0x0001, seq=3/768, ttl=128 (reply in 2361)
2361	191.547621	192.168.0.1	192.168.0.138	ICMP	74 Echo (ping) reply	id=0x0001, seq=3/768, ttl=64 (request in 2360)
2362	192.562435	192.168.0.138	192.168.0.1	ICMP	74 Echo (ping) request	id=0x0001, seq=4/1024, ttl=128 (reply in 2363)
2363	192.563908	192.168.0.1	192.168.0.138	ICMP	74 Echo (ping) reply	id=0x0001, seq=4/1024, ttl=64 (request in 2362)
2392	201.607307	TpLinkTechno_59:88:0b	Intel_60:d8:d0	ARP	42 Who has 192.168.0.138? Tell 192.168.0.1	
2393	201.607365	Intel_60:d8:d0	TpLinkTechno_59:88:0b	ARP	42 192.168.0.138 is at 2c:6d:c1:60:d8:d0	

Frame 2357: Packet, 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface \Device\NPF\_{...} [Ethernet II, Src: TpLinkTechno\_59:88:0b (28:ee:52:59:88:0b), Dst: Intel\_60:d8:d0 (2c:6d:c1:60:d8:d0)]  
Internet Protocol Version 4, Src: 192.168.0.1, Dst: 192.168.0.138  
Internet Control Message Protocol

Рис. 2.3: ICMP 2357

## 2.2.2 Анализ ARP-трафика

На скриншотах ниже представлен детальный анализ ARP-запроса и ответа.

### 1. ARP Request (пакет №81):

Описание: Роутер (TpLinkTechno) выполняет широковещательный запрос с целью узнать MAC-адрес устройства с IP 192.168.0.138. Запрос звучит как: “Кто имеет IP 192.168.0.138? Сообщите 192.168.0.1”.

Заголовок Ethernet II:

Destination MAC: Broadcast (ff:ff:ff:ff:ff:ff) (отправка всем устройствам в сети)

Source MAC: TpLinkTechno\_59:88:0b

Type: ARP (0x0806)



arp or icmp						
Список пакетов		Фильтр отображения		Введите фильтр отображения ...		
Опции: Обычные и многобайтовые		Чувствительность к регистру		Назад Множественные случаи		
No.	Time	Source	Destination	Protocol	Length Info	
81	20.294288	TpLinkTechno_59:88:0b	Intel_60:d8:d0	ARP	42 Who has 192.168.0.138? Tell 192.168.0.1	
82	20.294315	Intel_60:d8:d0	TpLinkTechno_59:88:0b	ARP	42 192.168.0.138 is at 2c:6d:c1:60:d8:d0	
127	48.215575	TpLinkTechno_59:88:0b	Intel_60:d8:d0	ARP	42 Who has 192.168.0.138? Tell 192.168.0.1	
128	48.215607	Intel_60:d8:d0	TpLinkTechno_59:88:0b	ARP	42 192.168.0.138 is at 2c:6d:c1:60:d8:d0	
165	74.200409	TpLinkTechno_59:88:0b	Intel_60:d8:d0	ARP	42 Who has 192.168.0.138? Tell 192.168.0.1	
166	74.200443	Intel_60:d8:d0	TpLinkTechno_59:88:0b	ARP	42 192.168.0.138 is at 2c:6d:c1:60:d8:d0	
1564	100.182944	TpLinkTechno_59:88:0b	Intel_60:d8:d0	ARP	42 Who has 192.168.0.138? Tell 192.168.0.1	
1565	100.182974	Intel_60:d8:d0	TpLinkTechno_59:88:0b	ARP	42 192.168.0.138 is at 2c:6d:c1:60:d8:d0	
1892	126.167161	TpLinkTechno_59:88:0b	Intel_60:d8:d0	ARP	42 Who has 192.168.0.138? Tell 192.168.0.1	
1893	126.167186	Intel_60:d8:d0	TpLinkTechno_59:88:0b	ARP	42 192.168.0.138 is at 2c:6d:c1:60:d8:d0	
2026	153.208776	TpLinkTechno_59:88:0b	Intel_60:d8:d0	ARP	42 Who has 192.168.0.138? Tell 192.168.0.1	
2027	153.208811	Intel_60:d8:d0	TpLinkTechno_59:88:0b	ARP	42 192.168.0.138 is at 2c:6d:c1:60:d8:d0	
2356	189.511851	192.168.0.138	192.168.0.1	ICMP	74 Echo (ping) request id=0x0001, seq=1/256, ttl=128 (reply in 2357)	
2357	189.513303	192.168.0.1	192.168.0.138	ICMP	74 Echo (ping) reply id=0x0001, seq=1/256, ttl=64 (request in 2356)	
2358	190.529203	192.168.0.138	192.168.0.1	ICMP	74 Echo (ping) request id=0x0001, seq=2/512, ttl=128 (reply in 2359)	
2359	190.532317	192.168.0.1	192.168.0.138	ICMP	74 Echo (ping) reply id=0x0001, seq=2/512, ttl=64 (request in 2358)	
2360	191.545940	192.168.0.138	192.168.0.1	ICMP	74 Echo (ping) request id=0x0001, seq=3/768, ttl=128 (reply in 2361)	
2361	191.547621	192.168.0.1	192.168.0.138	ICMP	74 Echo (ping) reply id=0x0001, seq=3/768, ttl=64 (request in 2360)	
2362	192.562435	192.168.0.138	192.168.0.1	ICMP	74 Echo (ping) request id=0x0001, seq=4/1024, ttl=128 (reply in 2363)	
2363	192.563908	192.168.0.1	192.168.0.138	ICMP	74 Echo (ping) reply id=0x0001, seq=4/1024, ttl=64 (request in 2362)	
2392	201.607307	TpLinkTechno_59:88:0b	Intel_60:d8:d0	ARP	42 Who has 192.168.0.138? Tell 192.168.0.1	
2393	201.607365	Intel_60:d8:d0	TpLinkTechno_59:88:0b	ARP	42 192.168.0.138 is at 2c:6d:c1:60:d8:d0	

<p>Frame 81: Packet, 42 bytes on wire (336 bits), 42 bytes captured (336 bits) on interface \Device\NPF_{1F000000-0000-0000-0000-000000000000}</p> <p>Ethernet II, Src: TpLinkTechno_59:88:0b (28:ee:52:59:88:0b), Dst: Intel_60:d8:d0 (2c:6d:c1:60:d8:d0)</p> <p>Destination: Intel_60:d8:d0 (2c:6d:c1:60:d8:d0)</p> <p>.....0..... = LG bit: Globally unique address (factory default)</p> <p>.....0..... = IG bit: Individual address (unicast)</p> <p>Source: TpLinkTechno_59:88:0b (28:ee:52:59:88:0b)</p> <p>.....0..... = LG bit: Globally unique address (factory default)</p> <p>.....0..... = IG bit: Individual address (unicast)</p> <p>Type: ARP (0x0806)</p> <p>[Stream index: 0]</p> <p>Address Resolution Protocol (request)</p>	<p>0000 2c 6d c1 60 d8 d0 28 ee 52 59 88 0b 08 06 00 01 ,м ( RV .....</p> <p>0010 08 00 06 04 00 01 28 ee 52 59 88 0b c0 a8 00 01 ..... ( RV .....</p> <p>0020 00 00 00 00 00 c0 a8 00 8a ..... ..</p>
--	--

Рис. 2.4: ARP 81

## 2. ARP Reply (пакет №82):

Описание: Мое устройство отвечает на ARP-запрос, сообщая свой MAC-адрес.  
 Ответ звучит как: “192.168.0.138 находится по MAC-адресу 2c:6d:c1:60:d8:d0”.

Заголовок Ethernet II: Destination MAC: TpLinkTechno\_59:88:0b (адресный ответ, не широковещательный)

Source MAC: Intel\_60:d8:d0

Type: ARP (0x0806)

arp or icmp

Список пакетов

Фильтр отображения

Введите фильтр отображения ...

Опции: Обычные и многобайтовые

Чувствительность к регистру

Назад

Множественные случаи

No.	Time	Source	Destination	Protocol	Length Info
81	20.294288	TpLinkTechno_59:88:0b	Intel_60:d8:d0	ARP	42 Who has 192.168.0.138? Tell 192.168.0.1
82	20.294315	Intel_60:d8:d0	TpLinkTechno_59:88:0b	ARP	42 192.168.0.138 is at 2c:6d:c1:60:d8:d0
127	48.215575	TpLinkTechno_59:88:0b	Intel_60:d8:d0	ARP	42 Who has 192.168.0.138? Tell 192.168.0.1
128	48.215607	Intel_60:d8:d0	TpLinkTechno_59:88:0b	ARP	42 192.168.0.138 is at 2c:6d:c1:60:d8:d0
165	74.208409	TpLinkTechno_59:88:0b	Intel_60:d8:d0	ARP	42 Who has 192.168.0.138? Tell 192.168.0.1
166	74.208443	Intel_60:d8:d0	TpLinkTechno_59:88:0b	ARP	42 192.168.0.138 is at 2c:6d:c1:60:d8:d0
1564	100.182944	TpLinkTechno_59:88:0b	Intel_60:d8:d0	ARP	42 Who has 192.168.0.138? Tell 192.168.0.1
1565	100.182974	Intel_60:d8:d0	TpLinkTechno_59:88:0b	ARP	42 192.168.0.138 is at 2c:6d:c1:60:d8:d0
1892	126.167161	TpLinkTechno_59:88:0b	Intel_60:d8:d0	ARP	42 Who has 192.168.0.138? Tell 192.168.0.1
1893	126.167186	Intel_60:d8:d0	TpLinkTechno_59:88:0b	ARP	42 192.168.0.138 is at 2c:6d:c1:60:d8:d0
2026	153.208776	TpLinkTechno_59:88:0b	Intel_60:d8:d0	ARP	42 Who has 192.168.0.138? Tell 192.168.0.1
2027	153.208811	Intel_60:d8:d0	TpLinkTechno_59:88:0b	ARP	42 192.168.0.138 is at 2c:6d:c1:60:d8:d0
2356	189.511851	192.168.0.138	192.168.0.1	ICMP	74 Echo (ping) request id=0x0001, seq=1/256, ttl=128 (reply in 2357)
2357	189.513303	192.168.0.1	192.168.0.138	ICMP	74 Echo (ping) reply id=0x0001, seq=1/256, ttl=64 (request in 2356)
2358	190.529203	192.168.0.138	192.168.0.1	ICMP	74 Echo (ping) request id=0x0001, seq=2/512, ttl=128 (reply in 2359)
2359	190.532317	192.168.0.1	192.168.0.138	ICMP	74 Echo (ping) reply id=0x0001, seq=2/512, ttl=64 (request in 2358)
2360	191.545940	192.168.0.138	192.168.0.1	ICMP	74 Echo (ping) request id=0x0001, seq=3/768, ttl=128 (reply in 2361)
2361	191.547621	192.168.0.1	192.168.0.138	ICMP	74 Echo (ping) reply id=0x0001, seq=3/768, ttl=64 (request in 2360)
2362	192.562435	192.168.0.138	192.168.0.1	ICMP	74 Echo (ping) request id=0x0001, seq=4/1024, ttl=128 (reply in 2363)
2363	192.563908	192.168.0.1	192.168.0.138	ICMP	74 Echo (ping) reply id=0x0001, seq=4/1024, ttl=64 (request in 2362)
2392	201.607307	TpLinkTechno_59:88:0b	Intel_60:d8:d0	ARP	42 Who has 192.168.0.138? Tell 192.168.0.1
2393	201.607365	Intel_60:d8:d0	TpLinkTechno_59:88:0b	ARP	42 192.168.0.138 is at 2c:6d:c1:60:d8:d0

4

Frame 82: Packet, 42 bytes on wire (336 bits), 42 bytes captured (336 bits) on interface \Device\NPF\_{1F000000-0000-0000-0000-000000000000}

Ethernet II, Src: Intel\_60:d8:d0 (2c:6d:c1:60:d8:d0), Dst: TpLinkTechno\_59:88:0b (28:ee:52:59:88:0b)

Destination: TpLinkTechno\_59:88:0b (28:ee:52:59:88:0b)

....0. .... = LG bit: Globally unique address (factory default)

....0. .... = IG bit: Individual address (unicast)

Source: Intel\_60:d8:d0 (2c:6d:c1:60:d8:d0)

....0. .... = LG bit: Globally unique address (factory default)

....0. .... = IG bit: Individual address (unicast)

Type: ARP (0x0806)

[Stream index: 0]

Address Resolution Protocol (reply)

28 ee 52 59 88 0b 2c 6d c1 60 d8 d0 08 06 00 01 ( RY . . . . . )

0010 08 00 06 04 00 02 2c 6d c1 60 d8 d0 c0 a8 00 8a ( . . . . . )

0020 28 ee 52 59 88 0b c0 a8 00 01 ( RY . . . . . )

Рис. 2.5: APR 82

## 2.3 Анализ протоколов транспортного уровня

### 2.3.1 Анализ HTTP

**Проанализировать информацию по протоколу TCP в случае HTTP-запросов и ответов.**

Был осуществлен переход на сайт <http://info.cern.ch/>, после чего трафик был отфильтрован по http.

#### 1. HTTP GET Request (пакет №152):

Описание: Клиент (мой ПК) запрашивает у сервера 188.184.67.127 корневую страницу /hypertext/WWW/TheProject.html.

Протокол TCP: Запрос инкапсулирован в TCP-сегмент. Source Port - динамический (49153), Destination Port - 80 (стандартный для HTTP).

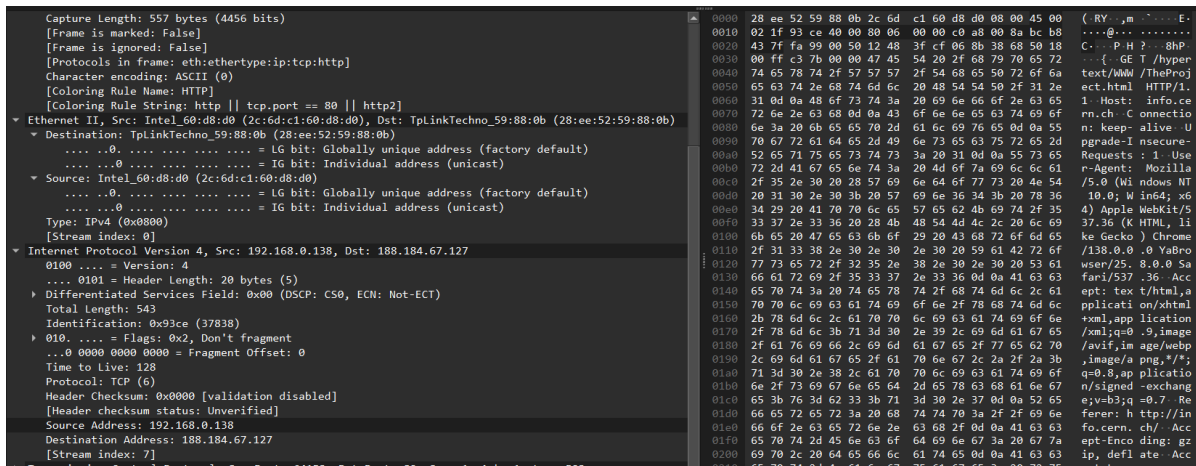


Рис. 2.6: HTTP GET Request (152)

## 2. HTTP 200 OK Response (пакет №191):

Описание: Сервер успешно отвечает на запрос, отправляя содержимое HTML-страницы.

Протокол TCP: Ответ также передается по TCP. Source Port - 80, Destination Port - 49153.

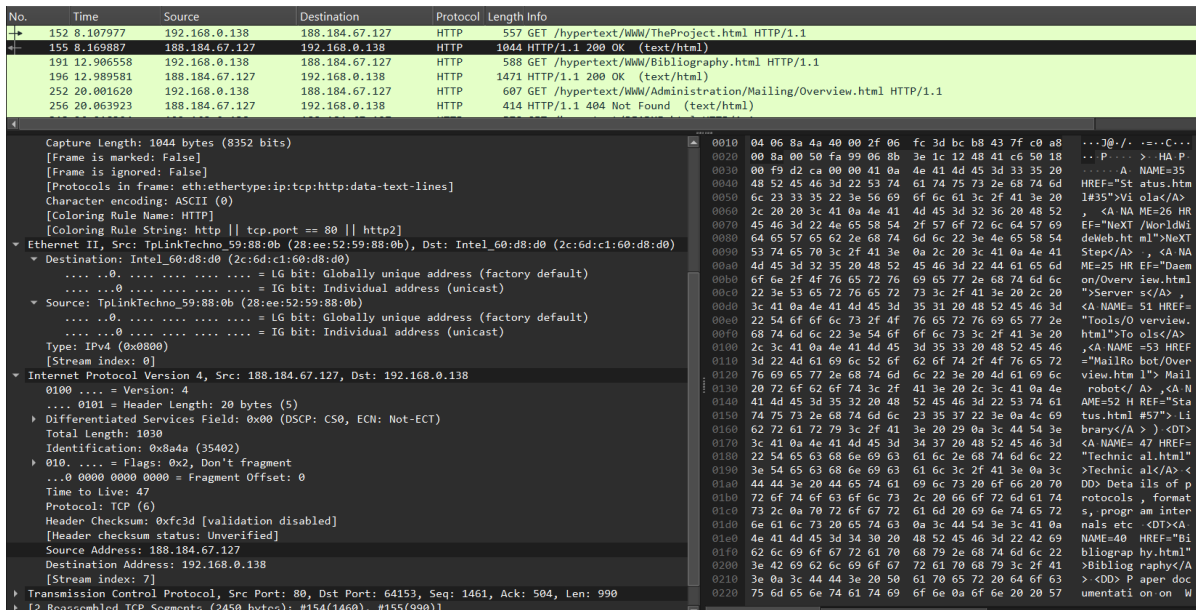


Рис. 2.7: HTTP 200 OK Response (155)

### 2.3.2 Анализ DNS

**Проанализировать информацию по протоколу UDP в случае DNS-запросов и ответов.**

Был захвачен трафик во время работы в браузере и отфильтрован по dns.

1. DNS Standard Query (пакет №63):

Описание: Клиент запрашивает у DNS-сервера (192.168.0.1) IP-адрес для домена на api.browser.yandex.ru.

Протокол UDP: Запрос инкапсулирован в UDP-дейтаграмму. Source Port - динамический (58384), Destination Port - 53 (стандартный для DNS).

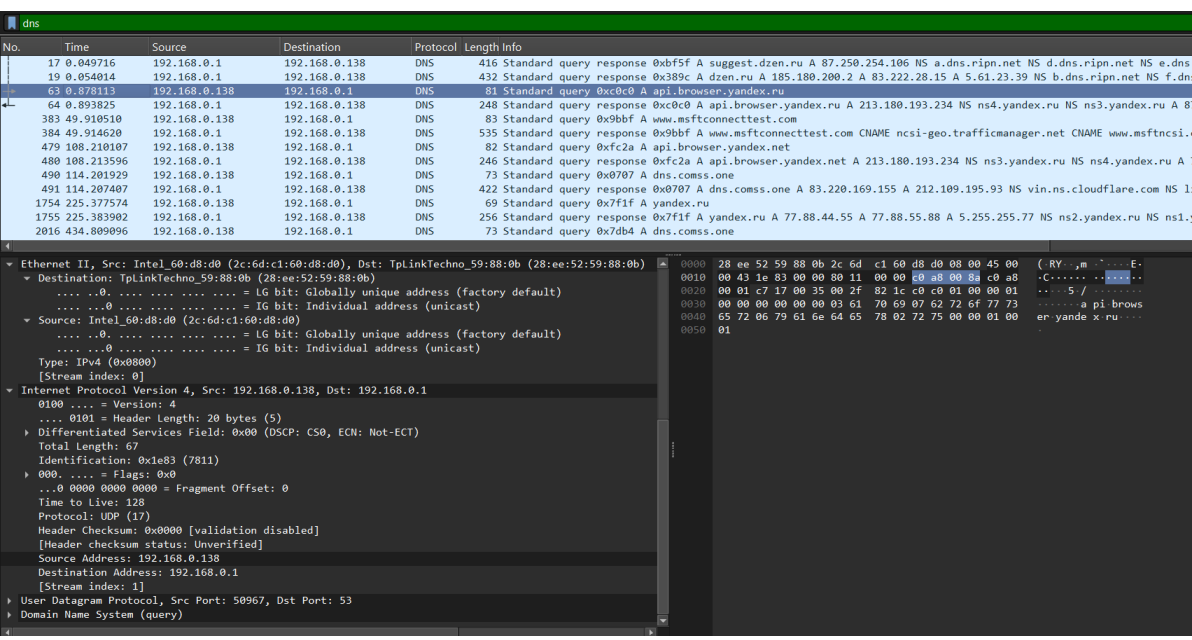


Рис. 2.8: DNS Standard Query (63)

2. DNS Standard Query Response (пакет №64):

Описание: DNS-сервер отвечает, предоставляя IP-адрес для запрошенного домена.

Протокол UDP: Source Port - 53, Destination Port - 58384.

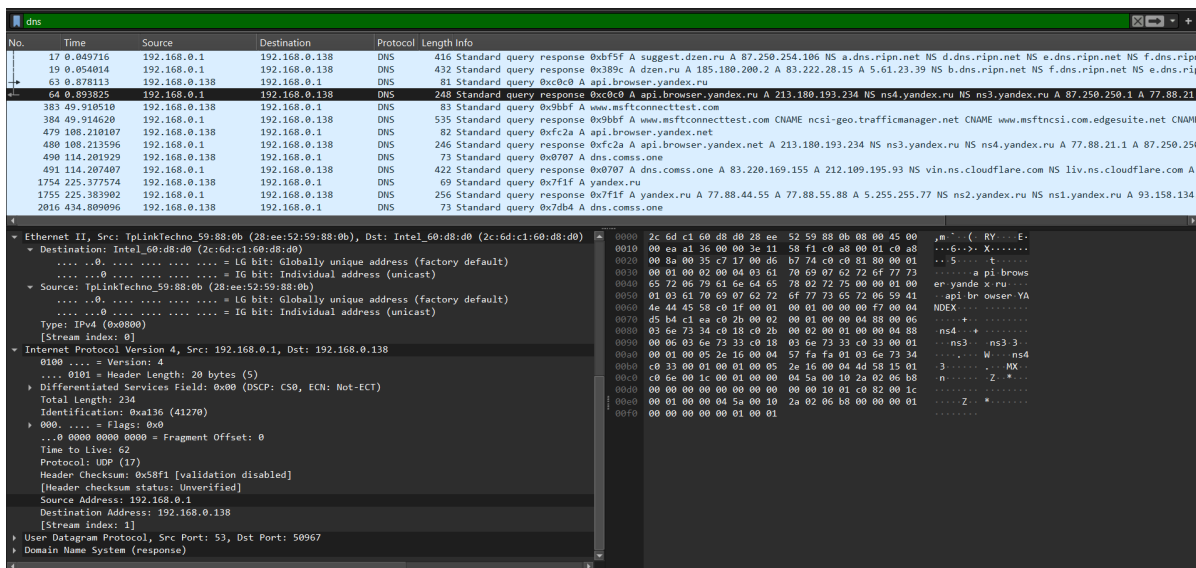


Рис. 2.9: DNS Standard Query Response (64)

## 2.3.3 Анализ QUIC

### Проанализировать информацию по протоколу QUIC.

Был захвачен трафик к современному веб-ресурсу, использующему протокол QUIC.

Описание: QUIC (Quick UDP Internet Connections) — это транспортный протокол, работающий поверх UDP. Он обеспечивает шифрование по умолчанию и более быстрое установление соединения.

На скриншотах виден обмен пакетами Initial и Handshake, которые служат для установления защищенного соединения между клиентом 192.168.0.138 и сервером 142.250.74.131.

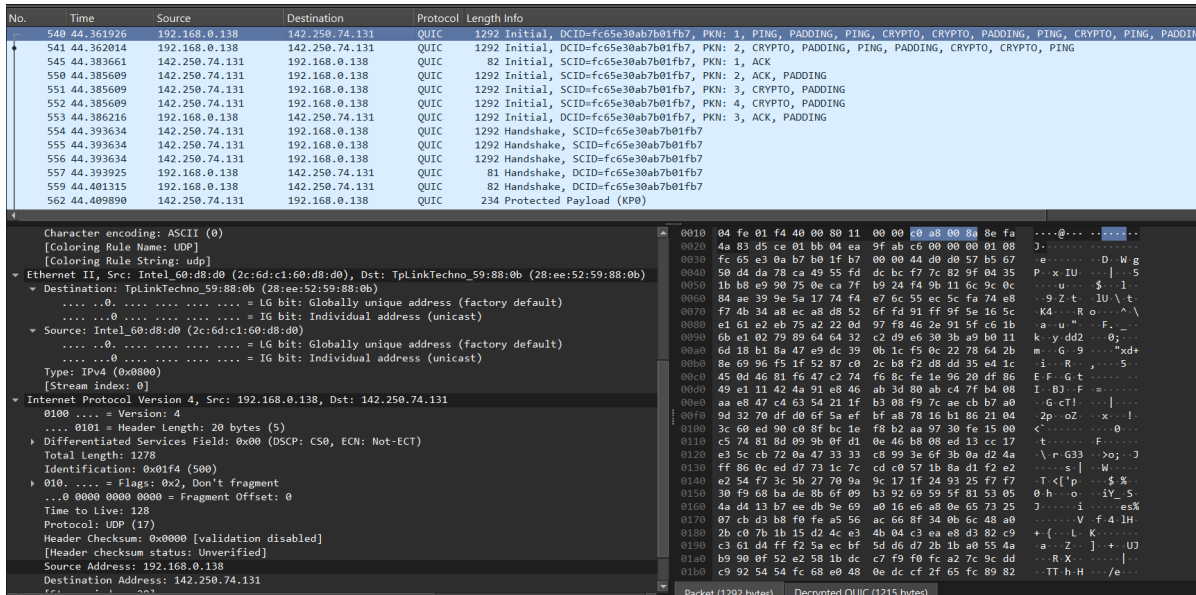


Рис. 2.10: QUIC

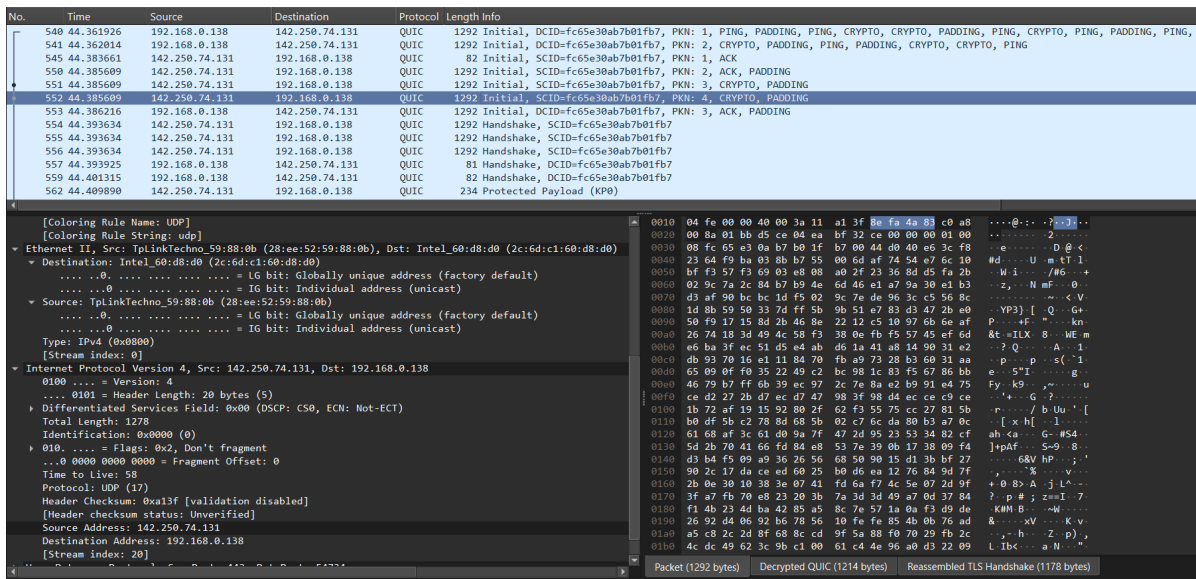


Рис. 2.11: QUIC

## 2.4 Анализ handshake протокола TCP

### 2.4.1 handshake

С помощью Wireshark проанализировать handshake протокола TCP.

Было инициировано соединение с веб-сервером, трафик был отфильтрован по `tcp.port == 80`. Были проанализированы первые три пакета, составляющие трёхступенчатое рукопожатие.

Шаг 1: SYN (пакет №143)

Клиент (192.168.0.138) отправляет серверу (188.184.67.127) сегмент с установленным флагом SYN (Synchronize). Это запрос на установку соединения.

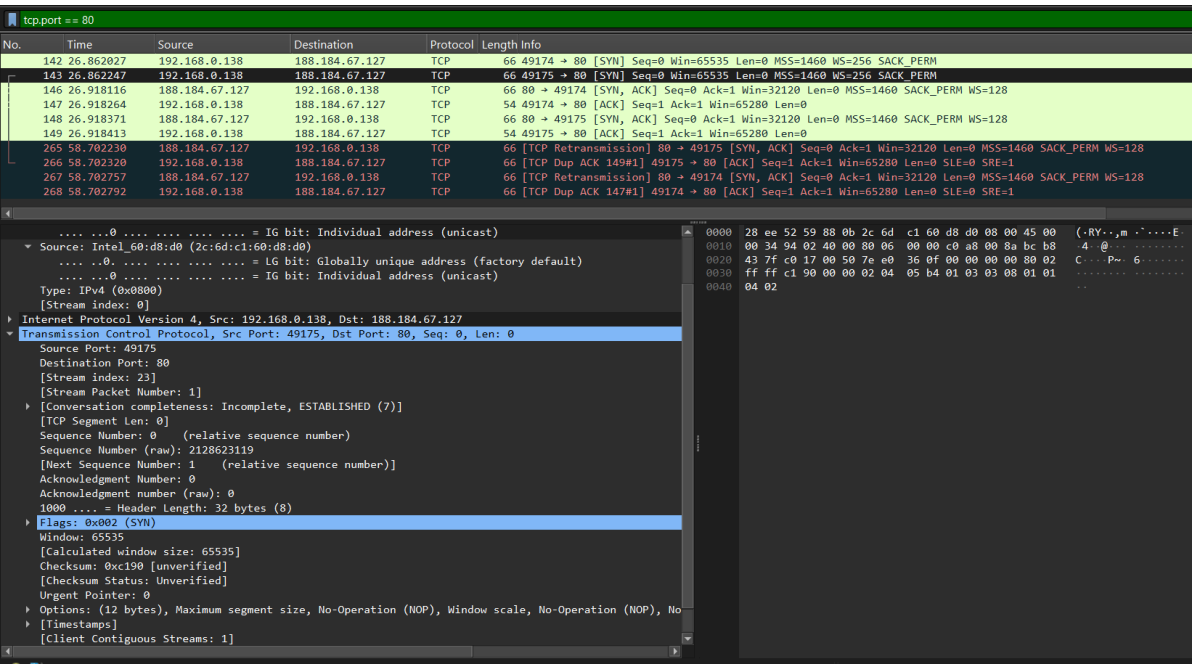


Рис. 2.12: syn

Шаг 2: SYN, ACK (пакет №146)

Сервер отвечает сегментом с двумя флагами: SYN (он также предлагает синхронизировать номер последовательности) и ACK (Acknowledgment - подтверждает получение первого пакета от клиента).



No.	Time	Source	Destination	Protocol	Length	Info
142	26.862027	192.168.0.138	188.184.67.127	TCP	66	49174 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM
143	26.862247	192.168.0.138	188.184.67.127	TCP	66	49175 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM
146	26.918116	188.184.67.127	192.168.0.138	TCP	66	80 → 49174 [SYN, ACK] Seq=0 Ack=1 Win=32120 Len=0 MSS=1460 SACK_PERM WS=128
147	26.918264	192.168.0.138	188.184.67.127	TCP	54	49174 → 80 [ACK] Seq=1 Ack=1 Win=65280 Len=0
148	26.918371	188.184.67.127	192.168.0.138	TCP	66	80 → 49175 [SYN, ACK] Seq=0 Ack=1 Win=32120 Len=0 MSS=1460 SACK_PERM WS=128
149	26.918413	192.168.0.138	188.184.67.127	TCP	54	49175 → 80 [ACK] Seq=1 Ack=1 Win=65280 Len=0
265	58.702230	188.184.67.127	192.168.0.138	TCP	66	[TCP Retransmission] 80 → 49175 [SYN, ACK] Seq=0 Ack=1 Win=32120 Len=0 MSS=1460 SACK_PERM WS=128
266	58.702320	192.168.0.138	188.184.67.127	TCP	66	[TCP Dup ACK 149#1] 49175 → 80 [ACK] Seq=1 Ack=1 Win=65280 Len=0 SLE=0 SRE=1
267	58.702757	188.184.67.127	192.168.0.138	TCP	66	[TCP Retransmission] 80 → 49174 [SYN, ACK] Seq=0 Ack=1 Win=32120 Len=0 MSS=1460 SACK_PERM WS=128
268	58.702792	192.168.0.138	188.184.67.127	TCP	66	[TCP Dup ACK 147#1] 49174 → 80 [ACK] Seq=1 Ack=1 Win=65280 Len=0 SLE=0 SRE=1

.....0..... = IG bit: Individual address (unicast)	0000 26 6d c1 60 d8 d0 28 ee 52 59 88 0b 08 00 45 00	4 0 / Z C ...
Source: TplinkTechno_59:88:0b (28:ee:52:59:88:0b)	0010 00 34 00 00 40 00 2f 06 8a 5a bc b8 43 7f c0 a8	..P To F ...
.....0..... = IG bit: Globally unique address (factory default)	0020 00 8a 00 50 c0 16 54 6f fe c5 46 60 b0 96 80 12	}x% .....
.....0..... = IG bit: Individual address (unicast)	0030 7d 78 25 8c 00 00 02 04 05 b4 01 01 04 02 01 03	..
Type: IPv4 (0x0800)	0040 03 07	
[Stream index: 0]		
Internet Protocol Version 4, Src: 188.184.67.127, Dst: 192.168.0.138		
Transmission Control Protocol, Src Port: 80, Dst Port: 49174, Seq: 0, Ack: 1, Len: 0		
Source Port: 80		
Destination Port: 49174		
[Stream index: 22]		
[Stream Packet Number: 2]		
[Conversation completeness: Incomplete, ESTABLISHED (7)]		
[TCP Segment Len: 0]		
Sequence Number: 0 (relative sequence number)		
Sequence Number (raw): 1416625861		
[Next Sequence Number: 1 (relative sequence number)]		
Acknowledgment Number: 1 (relative ack number)		
Acknowledgment number (raw): 1180741782		
1000 .... = Header Length: 32 bytes (8)		
Flags: 0x012 (SYN, ACK)		
Window: 32120		
[Calculated window size: 32120]		
Checksum: 0x258c [unverified]		
[Checksum Status: Unverified]		
Urgent Pointer: 0		
Options: (12 bytes), Maximum segment size, No-Operation (NOP), No-Operation (NOP), SACK permitted,		
[Timestamps]		
[SEQ/ACK analysis]		

Рис. 2.13: syn, ack

### Шаг 3: ACK (пакет №147)

Клиент отправляет серверу сегмент с флагом ACK, подтверждая получение пакета SYN, ACK от сервера. На этом рукопожатие завершается, и соединение считается установленным.



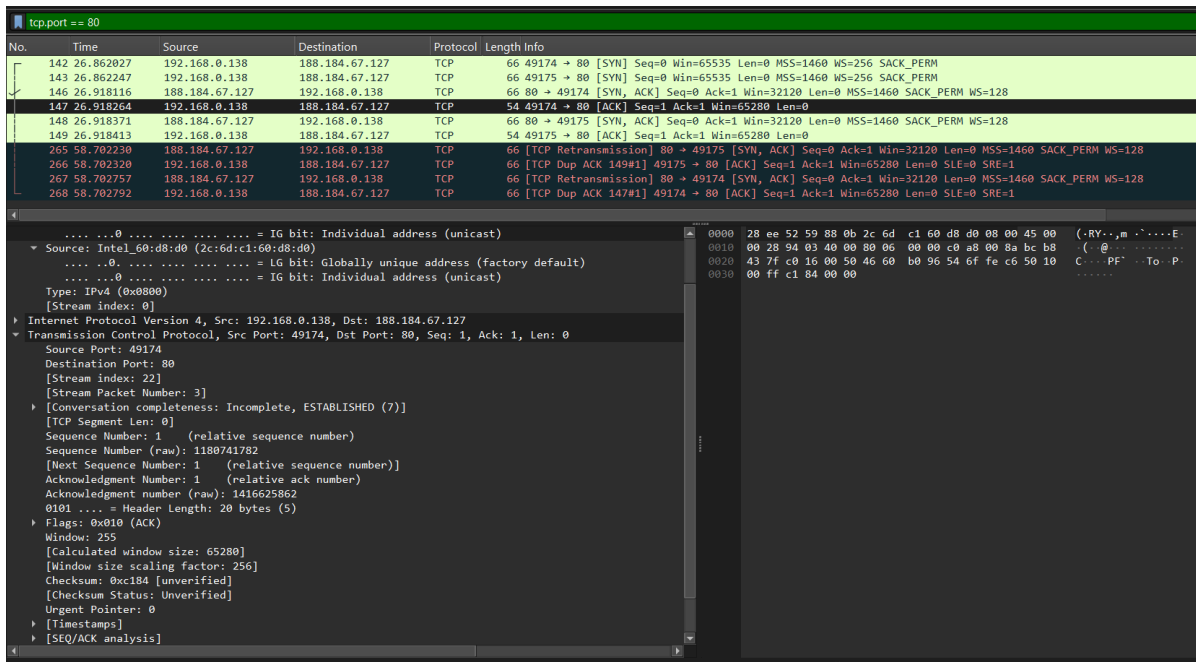


Рис. 2.14: ack

## 2.4.2 График Потока

Для визуализации обмена был построен график потока, на котором наглядно представлено всё TCP-соединение, включая начальное трёхступенчатое рукопожатие и последующие повторные передачи (Retransmissions).

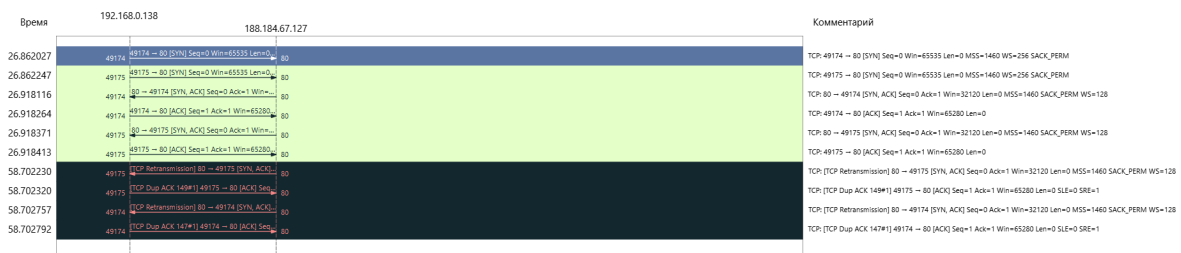


Рис. 2.15: График Потока

## **3 Выводы**

В ходе работы изучение посредством Wireshark кадров Ethernet, анализ PDU протоколов транспортного и прикладного уровней стека TCP/IP прошли успешно.

## **Список литературы**

(ТУИС)[<https://esystem.rudn.ru/course/view.php?id=9060>]