

САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
УНИВЕРСИТЕТ
ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, МЕХАНИКИ И ОПТИКИ
ФАКУЛЬТЕТ ИНФОКОММУНИКАЦИОННЫХ ТЕХНОЛОГИЙ

Отчет по лабораторной работе №5
по курсу «Компьютерные сети»

Тема: Изучение работы протоколов стека TCP/IP с помощью Wireshark

Выполнил:
Закоурцев Андрей
К3220

Проверил:
Харитонов А.Ю.

Санкт-Петербург
2025 г.

СОДЕРЖАНИЕ

Стр.

ВВЕДЕНИЕ	3
1 Начало работы с Wireshark.....	4
1.0.1 Захват трафика	4
1.0.2 Статистика	4
1.0.3 Фильтры	9
2 Сбор и анализ данных протокола ICMP	15
2.0.1 Сбор и анализ данных протокола ICMP по локальным узлам	15
2.0.2 Сбор и анализ данных протокола ICMP по удаленным узлам	16
3 Анализ полей TCP	18
ЗАКЛЮЧЕНИЕ	24

ВВЕДЕНИЕ

Современные компьютерные сети основаны на стеке протоколов TCP/IP, который обеспечивает передачу данных между устройствами. Понимание принципов работы этого стека, а также умение анализировать сетевой трафик являются важными навыками.

Целью данной работы является изучение стека TCP/IP путем анализа сетевых пакетов, передаваемых и принимаемых с его использованием. Для сбора и анализа трафика будет использоваться программа Wireshark. В ходе работы предстоит освоить методы фильтрации трафика, выявления установленных соединений и их последующего анализа.

Wireshark · Endpoints · Беспроводная сеть

Endpoint Settings

- ☐ Разрешение имён
- ☒ Ограничить по фильтру от
- Копировать
- Карта

Ethernet · 12		IPv4 · 60		IPv6 · 7		TCP · 149		UDP · 115	
Адрес	Пакеты	Байты	Пакетов отправлено	Байтов отправлено	Пакетов получено	Байтов получено	Пакетов отправлено	Байтов отправлено	Пакетов получено
188.166.145.130	2 866	3 МБ	2 320	3 МБ	546	3 МБ	2 320	3 МБ	546
192.168.0.4	5 526	5 МБ	1 841	486 кБ	3 685	486 кБ	1 841	486 кБ	3 685
162.159.137.232	471	564 кБ	347	476 кБ	124	476 кБ	347	476 кБ	124
162.159.130.234	140	129 кБ	98	120 кБ	42	120 кБ	98	120 кБ	42
149.154.167.223	83	50 кБ	48	46 кБ	35	46 кБ	48	46 кБ	35
8.8.8.8	201	50 кБ	108	36 кБ	93	36 кБ	108	36 кБ	93
23.22.252.240	187	74 кБ	82	34 кБ	105	34 кБ	82	34 кБ	105
20.7.251.164	73	39 кБ	39	27 кБ	34	27 кБ	39	27 кБ	34
74.125.205.194	113	32 кБ	57	17 кБ	56	17 кБ	57	17 кБ	56
184.50.201.213	33	18 кБ	18	15 кБ	15	15 кБ	18	15 кБ	15
169.254.69.241	95	14 кБ	95	14 кБ	0	14 кБ	95	14 кБ	0
34.120.52.64	75	19 кБ	47	14 кБ	28	14 кБ	47	14 кБ	28
143.198.234.31	39	18 кБ	22	14 кБ	17	14 кБ	22	14 кБ	17
149.154.167.41	94	20 кБ	46	12 кБ	48	12 кБ	46	12 кБ	48
8.8.4.4	39	15 кБ	21	11 кБ	18	11 кБ	21	11 кБ	18
192.168.0.1	115	14 кБ	58	10 кБ	57	10 кБ	58	10 кБ	57
4.207.247.139	33	11 кБ	20	8 кБ	13	8 кБ	20	8 кБ	13
64.233.162.101	22	9 кБ	12	8 кБ	10	8 кБ	12	8 кБ	10
2.16.56.86	18	8 кБ	10	7 кБ	8	7 кБ	10	7 кБ	8
74.125.205.188	17	8 кБ	10	7 кБ	7	7 кБ	10	7 кБ	7
93.186.225.205	27	10 кБ	13	6 кБ	14	6 кБ	13	6 кБ	14
87.240.132.67	31	10 кБ	15	6 кБ	16	6 кБ	15	6 кБ	16
209.85.233.95	20	7 кБ	11	6 кБ	9	6 кБ	11	6 кБ	9

Протокол

- ☐ Bluetooth
- ☐ BPv7
- ☐ DCCP
- ☒ Ethernet
- ☐ FC
- ☐ FDDI
- ☐ IEEE 802.11

Рисунок 1.2 — Узел с наибольшим трафиком

2. Узел, осуществивший наибольшее количество широковещательных рассылок: выставляем фильтр на широковещательный трафик - `eth.dst == ff:ff:ff:ff:ff:ff || ip.dst == 255.255.255.255`, получаем узел 169.254.69.241 (Рисунок 1.3).

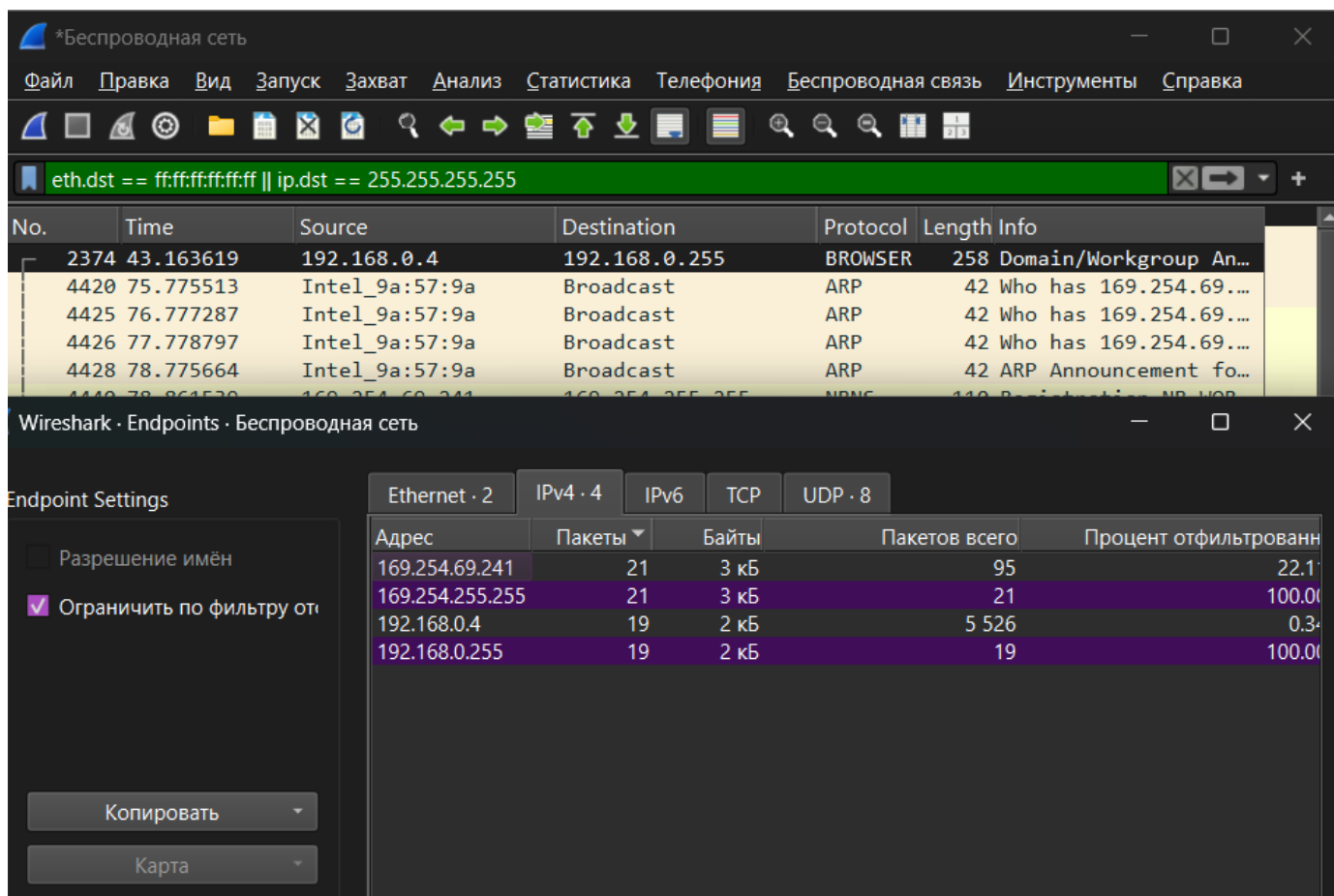


Рисунок 1.3 — Узел с наибольшим широковещательным трафиком

3. Самый активный TCP-порт на хосте (по количеству переданных пакетов): для этого переходим на вкладку TCP в статистике и сортируем по количеству пакетов. Получилось, что самый активный порт - 22222 с адреса 188.166.145.130 (Рисунок 1.4).

Wireshark · Endpoints · comp_network_lab5.pcapng

Endpoint Settings

☐ Разрешение имён

☒ Ограничить по фильтру от

Копировать

Карта

Протокол

- ☐ Bluetooth
- ☐ BPv7
- ☐ DCCP
- ☒ Ethernet
- ☐ FC
- ☐ FDDI
- ☐ IEEE 802.11
- ☐ IEEE 802.15.4
- ☒ IPv4
- ☒ IPv6
- ☐ IPX
- ☐ JXTA
- ☐ LTP
- ☐ MPTCP
- ☐ NCP
- ☐ SCTP

Ethernet · 12		IPv4 · 60		IPv6 · 7		TCP · 149		UDP · 115	
Адрес	Порт	Пакеты	Байты	Пакетов отправлено	Байтов отправлено	Пакеты	Байты	Пакетов отправлено	Байтов отправлено
188.166.145.130	22222	2 810	3 МБ	2 294	3 000				
192.168.0.4	5377	2 728	3 МБ	474	4 000				
162.159.137.232	443	409	486 кБ	343	470				
192.168.0.4	5373	409	486 кБ	66	1 000				
23.22.252.240	443	187	74 кБ	82	300				
162.159.130.234	443	140	129 кБ	98	120				
192.168.0.4	5370	140	129 кБ	42	500				
192.168.0.4	5363	96	40 кБ	54	200				
8.8.8.8	443	88	25 кБ	48	100				
34.120.52.64	443	75	19 кБ	47	100				
20.7.251.164	443	73	39 кБ	39	200				
149.154.167.41	443	70	17 кБ	37	100				
149.154.167.223	443	67	48 кБ	42	400				
91.105.192.100	80	64	7 кБ	24	100				
91.105.192.100	443	64	9 кБ	24	100				
192.168.0.4	6451	59	47 кБ	20	100				
188.166.145.130	22223	56	27 кБ	26	100				
192.168.0.4	5344	56	27 кБ	30	200				
192.168.0.4	5380	55	16 кБ	18	100				
192.168.0.4	5361	45	9 кБ	22	100				
142.250.150.113	443	44	17 кБ	6	396 600				
192.168.0.4	6448	42	11 кБ	18	100				
143.198.234.31	22222	39	18 кБ	22	100				
192.168.0.4	6460	39	18 кБ	17	100				
162.159.138.232	443	37	9 кБ	9	100				
192.168.0.4	5336	37	5 кБ	20	100				
192.168.0.4	5372	37	9 кБ	28	100				
192.168.0.4	6491	37	20 кБ	17	100				
162.159.135.233	443	36	9 кБ	8	100				
192.168.0.4	5374	36	9 кБ	28	100				
192.168.0.4	6462	36	20 кБ	17	100				
162.159.138.234	443	35	9 кБ	8	100				

Рисунок 1.4 — Самый активный TCP порт

4. Графики интенсивности TCP и UDP трафика (пункт Io Graphs)

Чтобы построить график заходим в графики ввода/вывода и добавляем данные с помощью фильтра. Синий график - TCP, красный - UDP. При этом на моем графике заметно, что в период от 100 до 300 секунд пакетов нет - я перезапускал настройку сети и поэтому не было соединения с беспроводной сетью. (Рисунок 1.5).

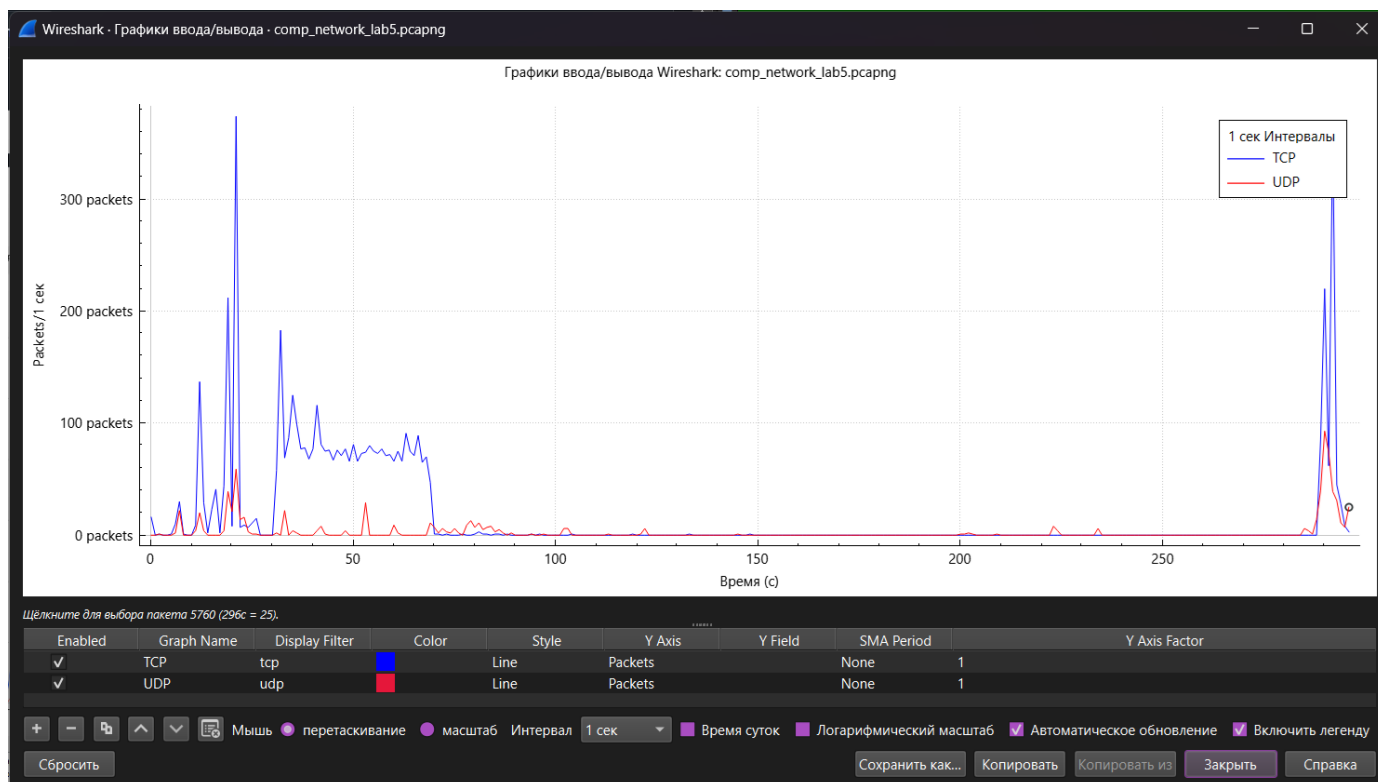


Рисунок 1.5 — График TCP и UDP пакетов

5. Диаграмма связей только для пакетов, содержащих сообщения протокола HTTPS

Для начала прописываем фильтр для выборки только https пакетов на 443 порту:

```
tcp.port==443
```

После заходим в статистику в график потока, тип потока выбираем TCP Flows. Результат на Рисунке 1.6.

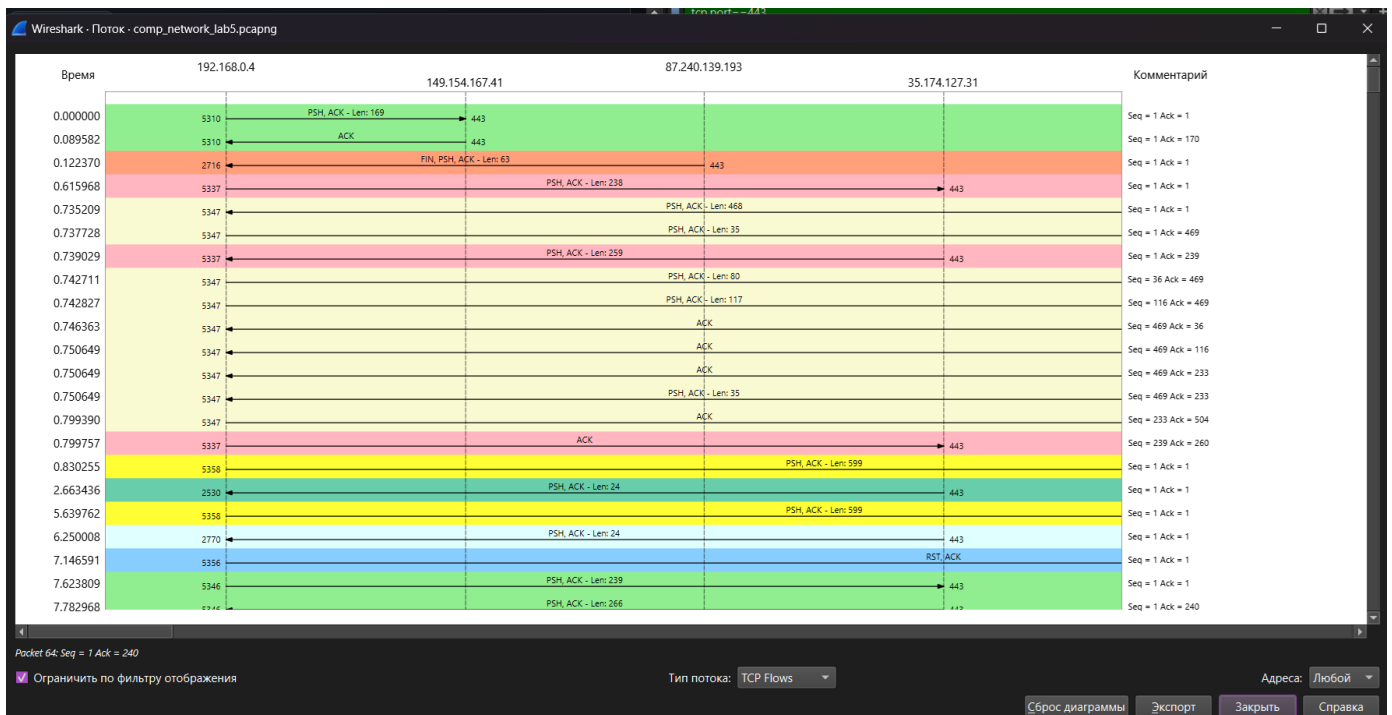


Рисунок 1.6 — График потока для https трафика

1.0.3 Фильтры

1. Необходимо отфильтровать так, чтобы не показывало http трафик веб-сервера хоста на 80 порту. Для этого нужно исключить трафик, исходящий с 80 порта источника, и трафик, идущий на 80 порт источника. Применяем фильтр (Рисунок 1.7):

```
http && !(ip.src == 192.168.0.4 && tcp.srcport == 80)
&& !(ip.dst == 192.168.0.4 && tcp.dstport == 80)
```

The screenshot shows the Wireshark interface with a packet capture filter applied: `http && !(ip.src == 192.168.0.4 && tcp.srcport == 80) && !(ip.dst == 192.168.0.4 && tcp.dstport == 80)`. The packet list shows 20 filtered packets, all of which are HTTP requests and responses.

No.	Time	Source	Destination	Protocol	Length	Info
398	18.934089	192.168.0.4	192.168.0.1	HTTP	332	GET /wpad.dat HTTP/1.1
1211	31.188418	192.168.0.4	2.16.21.66	HTTP	208	GET /connecttest.txt HTTP/1.1
1212	31.188553	192.168.0.4	2.16.21.66	HTTP	208	GET /connecttest.txt HTTP/1.1
1215	31.196543	2.16.21.66	192.168.0.4	HTTP	241	HTTP/1.1 200 OK (text/plain)
1216	31.196543	2.16.21.66	192.168.0.4	HTTP	241	HTTP/1.1 200 OK (text/plain)
2317	42.583352	192.168.0.4	192.168.0.1	HTTP	180	GET /wpad.dat HTTP/1.1
2325	42.602212	192.168.0.4	192.168.0.1	HTTP	180	GET /wpad.dat HTTP/1.1
4803	290.185488	192.168.0.4	2.16.21.40	HTTP	165	GET /connecttest.txt HTTP/1.1
4805	290.192143	2.16.21.40	192.168.0.4	HTTP	241	HTTP/1.1 200 OK (text/plain)
4833	290.616767	192.168.0.4	149.154.167.223	HTTP	154	POST /api HTTP/1.1 (application/json)
4907	290.804358	192.168.0.4	149.154.175.55	HTTP	106	POST /api HTTP/1.1 (application/json)
4909	290.804954	192.168.0.4	149.154.175.51	HTTP	98	POST /api HTTP/1.1 (application/json)
5361	292.474109	192.168.0.4	91.105.192.100	HTTP	254	POST /api HTTP/1.1 (application/json)
5364	292.481734	192.168.0.4	91.105.192.100	HTTP	126	POST /api HTTP/1.1 (application/json)
5373	292.488584	192.168.0.4	91.105.192.100	HTTP	270	POST /api HTTP/1.1 (application/json)
5382	292.491089	192.168.0.4	91.105.192.100	HTTP	230	POST /api HTTP/1.1 (application/json)
5384	292.493133	192.168.0.4	91.105.192.100	HTTP	342	POST /api HTTP/1.1 (application/json)
5389	292.497351	192.168.0.4	91.105.192.100	HTTP	258	POST /api HTTP/1.1 (application/json)
5390	292.497436	192.168.0.4	91.105.192.100	HTTP	162	POST /api HTTP/1.1 (application/json)
5397	292.502204	192.168.0.4	91.105.192.100	HTTP	126	POST /api HTTP/1.1 (application/json)
5404	292.515509	192.168.0.4	149.154.167.41	HTTP	150	POST /api HTTP/1.1 (application/json)
5422	292.525057	192.168.0.4	149.154.167.41	HTTP	98	POST /api HTTP/1.1 (application/json)
5446	292.538510	192.168.0.4	149.154.167.41	HTTP	182	POST /api HTTP/1.1 (application/json)
5450	292.541227	192.168.0.4	149.154.167.223	HTTP	342	POST /api HTTP/1.1 (application/json)

Рисунок 1.7 — Фильтр на http трафик без трафика веб-сервера

2. Фильтр для кадров Ethernet, отправленных с сетевого интерфейса хоста: для начала узнаем MAC-адрес сетевого адаптера через `ipconfig`, потом прописываем фильтр `eth.src == 30-05-05-9A-57-9A` (Рисунок 1.8).

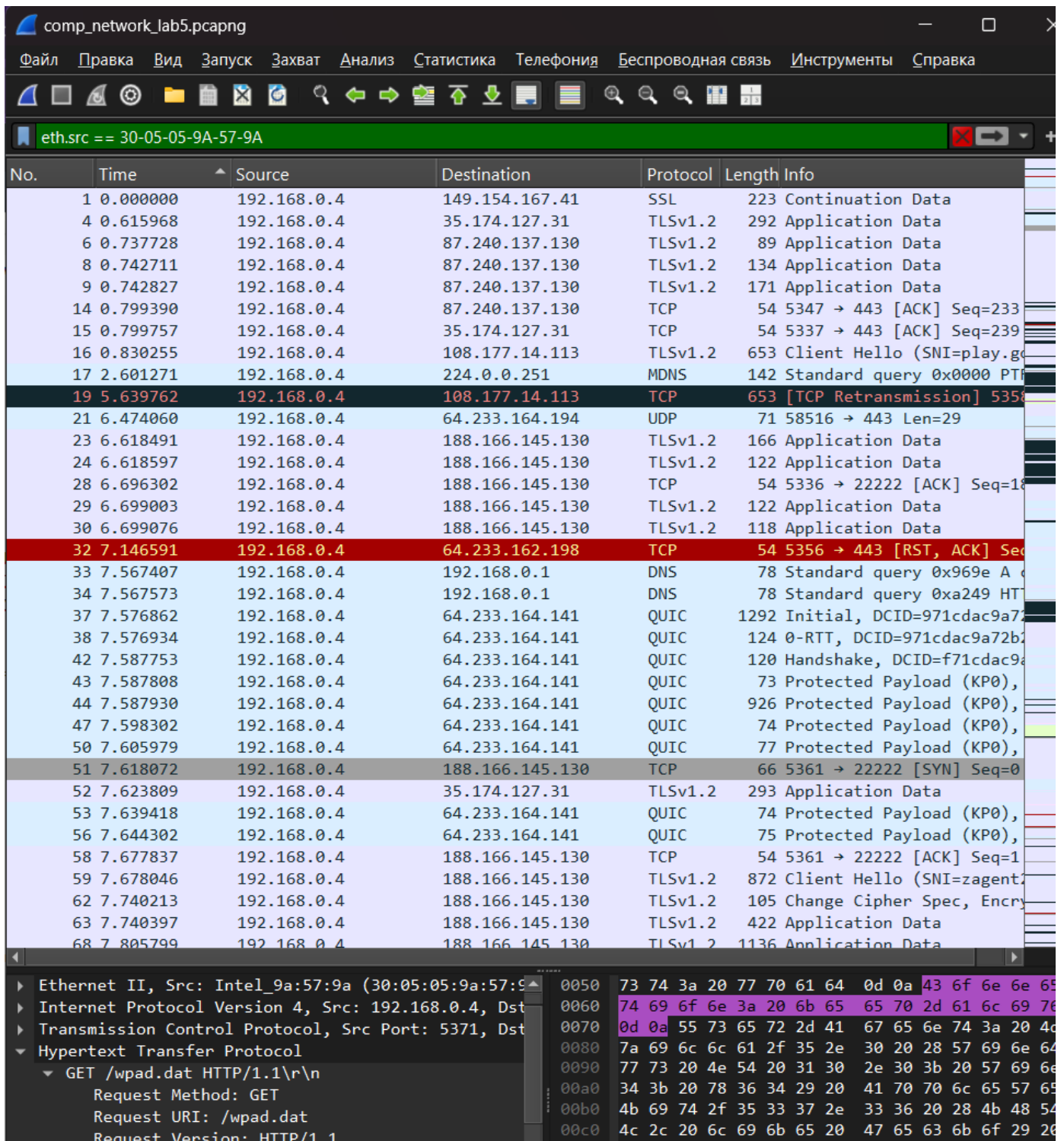
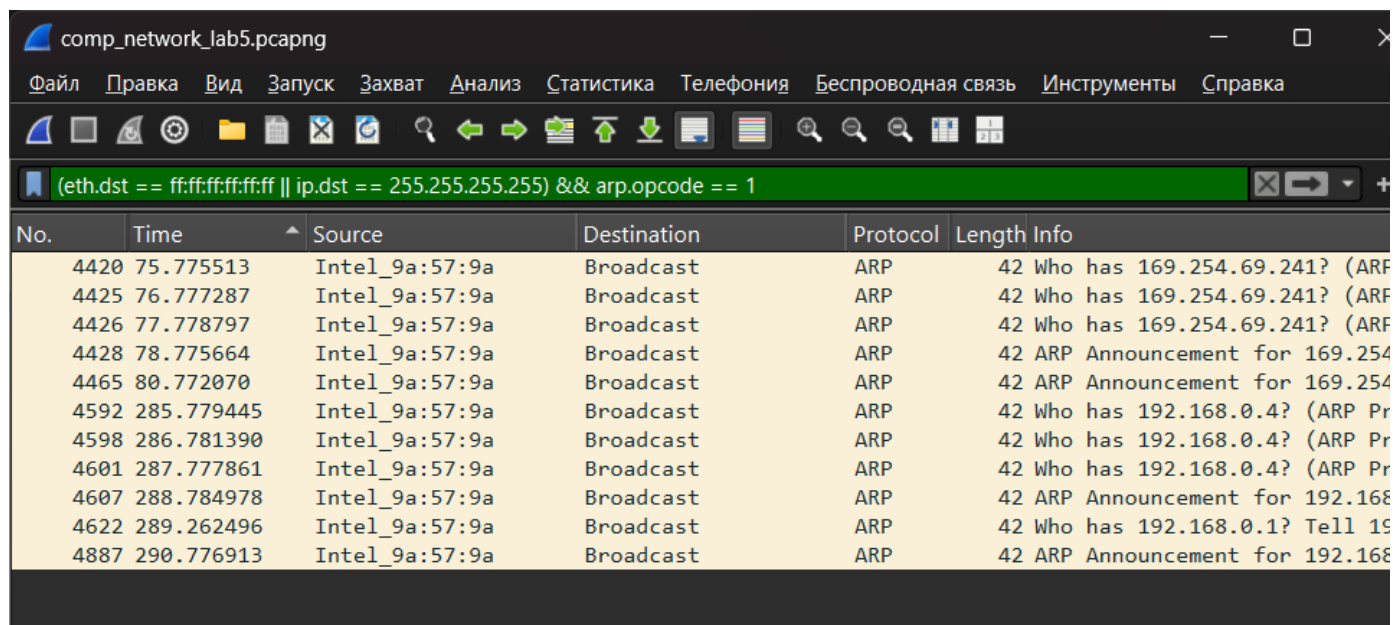


Рисунок 1.8 — Фильтр для кадров Ethernet, отправленных с сетевого интерфейса хоста

3. Напишите фильтр, отбирающий только широковещательные сообщения. Определите назначение 3-х широковещательных рассылок разных протоколов (или тех, которые удалось обнаружить).

Широковещательные сообщения мы уже находили фильтром `eth.dst == ff:ff:ff:ff:ff:ff || ip.dst == 255.255.255.255`, теперь добавим к нему фильтры для поиска разных типов рассылок:

"`arp.opcode == 1` поиск MAC-адреса по известному IP (Рисунок 1.9



The screenshot shows the Wireshark interface with a packet capture filter applied: `(eth.dst == ff:ff:ff:ff:ff:ff || ip.dst == 255.255.255.255) && arp.opcode == 1`. The packet list shows several ARP broadcast packets from source Intel_9a:57:9a to destination Broadcast.

No.	Time	Source	Destination	Protocol	Length	Info
4420	75.775513	Intel_9a:57:9a	Broadcast	ARP	42	Who has 169.254.69.241? (ARP Request)
4425	76.777287	Intel_9a:57:9a	Broadcast	ARP	42	Who has 169.254.69.241? (ARP Request)
4426	77.778797	Intel_9a:57:9a	Broadcast	ARP	42	Who has 169.254.69.241? (ARP Request)
4428	78.775664	Intel_9a:57:9a	Broadcast	ARP	42	ARP Announcement for 169.254.69.241
4465	80.772070	Intel_9a:57:9a	Broadcast	ARP	42	ARP Announcement for 169.254.69.241
4592	285.779445	Intel_9a:57:9a	Broadcast	ARP	42	Who has 192.168.0.4? (ARP Request)
4598	286.781390	Intel_9a:57:9a	Broadcast	ARP	42	Who has 192.168.0.4? (ARP Request)
4601	287.777861	Intel_9a:57:9a	Broadcast	ARP	42	Who has 192.168.0.4? (ARP Request)
4607	288.784978	Intel_9a:57:9a	Broadcast	ARP	42	ARP Announcement for 192.168.0.4
4622	289.262496	Intel_9a:57:9a	Broadcast	ARP	42	Who has 192.168.0.1? Tell 192.168.0.1
4887	290.776913	Intel_9a:57:9a	Broadcast	ARP	42	ARP Announcement for 192.168.0.1

Рисунок 1.9 — поиск MAC-адреса по известному IP

"`nbns` разрешение имен в локальной сети (Рисунок 1.10).

No.	Time	Source	Destination	Protocol	Length	Info
4440	78.861539	169.254.69.241	169.254.255.255	NBNS	110	Registration NB WORKGROUP<00
4441	78.861658	169.254.69.241	169.254.255.255	NBNS	110	Registration NB LAPTOP-C320C
4442	78.861791	169.254.69.241	169.254.255.255	NBNS	110	Registration NB LAPTOP-C320C
4446	79.613914	169.254.69.241	169.254.255.255	NBNS	110	Registration NB LAPTOP-C320C
4447	79.614028	169.254.69.241	169.254.255.255	NBNS	110	Registration NB LAPTOP-C320C
4448	79.614067	169.254.69.241	169.254.255.255	NBNS	110	Registration NB WORKGROUP<00
4462	80.376957	169.254.69.241	169.254.255.255	NBNS	110	Registration NB WORKGROUP<00
4463	80.377140	169.254.69.241	169.254.255.255	NBNS	110	Registration NB LAPTOP-C320C
4464	80.377213	169.254.69.241	169.254.255.255	NBNS	110	Registration NB LAPTOP-C320C
4469	81.129399	169.254.69.241	169.254.255.255	NBNS	110	Registration NB LAPTOP-C320C
4470	81.129507	169.254.69.241	169.254.255.255	NBNS	110	Registration NB LAPTOP-C320C
4471	81.129544	169.254.69.241	169.254.255.255	NBNS	110	Registration NB WORKGROUP<00
4479	81.899981	169.254.69.241	169.254.255.255	NBNS	110	Registration NB WORKGROUP<1e
4486	82.659920	169.254.69.241	169.254.255.255	NBNS	110	Registration NB WORKGROUP<1e
4494	83.414667	169.254.69.241	169.254.255.255	NBNS	110	Registration NB WORKGROUP<1e
4495	84.170258	169.254.69.241	169.254.255.255	NBNS	110	Registration NB WORKGROUP<1e
4619	288.866664	192.168.0.4	192.168.0.255	NBNS	110	Registration NB LAPTOP-C320C
4620	288.866929	192.168.0.4	192.168.0.255	NBNS	110	Registration NB LAPTOP-C320C
4621	288.867139	192.168.0.4	192.168.0.255	NBNS	110	Registration NB WORKGROUP<00
4645	289.624986	192.168.0.4	192.168.0.255	NBNS	110	Registration NB WORKGROUP<00
4646	289.625111	192.168.0.4	192.168.0.255	NBNS	110	Registration NB LAPTOP-C320C
4647	289.625164	192.168.0.4	192.168.0.255	NBNS	110	Registration NB LAPTOP-C320C

Рисунок 1.10 — Фильтр nbns

"browser весь трафик Browser: служебный протокол Windows для обнаружения сетевых ресурсов (Рисунок 1.11).

No.	Time	Source	Destination	Protocol	Length	Info
2374	43.163619	192.168.0.4	192.168.0.255	BROWSER	258	Domain/Workgroup Announcement
4501	84.930521	169.254.69.241	169.254.255.255	BROWSER	228	Request Announcement LAPTOP-
4502	84.940065	169.254.69.241	169.254.255.255	BROWSER	243	Host Announcement LAPTOP-C32
4512	86.444009	169.254.69.241	169.254.255.255	BROWSER	228	Request Announcement LAPTOP-
4514	87.946088	169.254.69.241	169.254.255.255	BROWSER	228	Request Announcement LAPTOP-
4517	89.460844	169.254.69.241	169.254.255.255	BROWSER	228	Request Announcement LAPTOP-
5714	294.926968	192.168.0.4	192.168.0.255	BROWSER	228	Request Announcement LAPTOP-
5715	294.936101	192.168.0.4	192.168.0.255	BROWSER	243	Host Announcement LAPTOP-C32

Рисунок 1.11 — Трафик browser

4. ip и mac адреса при это будут указаны в поле destination. Для broadcast это mac-адрес 255.255.255.255 или ff:ff:ff:ff:ff:ff

5. Фильтры уже были указаны в 3-м пункте.

6. Для определения к какому типу устройства подключен хост, проанализируем пакеты. Можно заметить, что 95 процентов трафика приходится на мой ip адрес 192.168.0.4, что исключает возможность хаба, а также видны arp и dhcp запросы, что исключает коммутатор, в таком случае я подключен к маршрутизатору.

2 Сбор и анализ данных протокола ICMP

2.0.1 Сбор и анализ данных протокола ICMP по локальным узлам

У меня дома два маршрутизатора, поэтому я подключился к тому, к которому подключены компьютеры. Чтобы узнать локальные ip адреса я посмотрел вывод ipconfig. Компьютер - 192.168.5.15, ноутбук с которого делается лабораторная - 192.168.5.10

После я создал правило межсетевого экрана, чтобы разрешить icmp запросы и пинганул устройства по очереди (Рисунок 2.1).

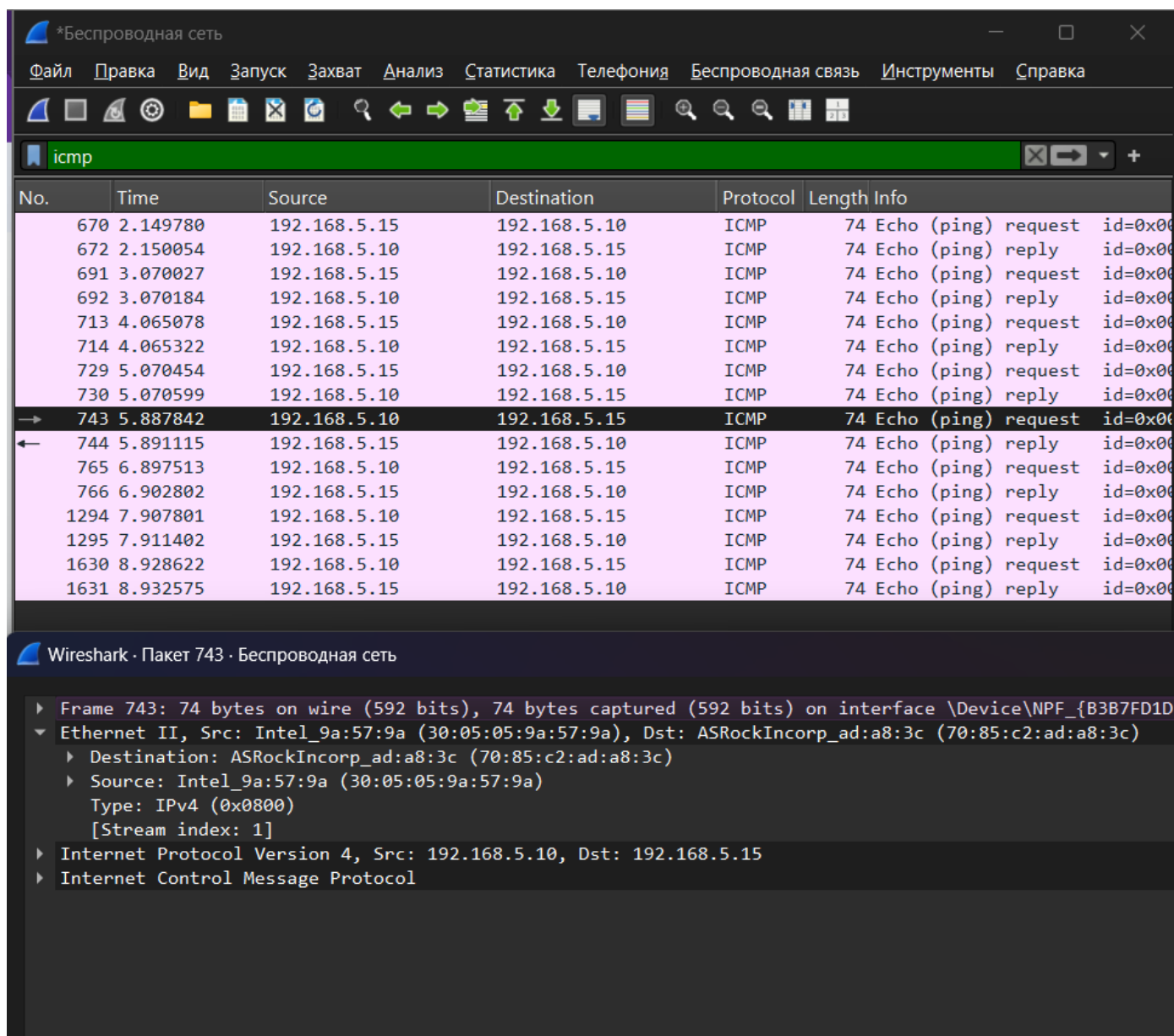


Рисунок 2.1 — ICMP запросы

Можно видеть, что было отправлено по 4 пакета друг к другу и получены ответы на них. Вопросы:

1. Совпадает ли MAC-адрес источника с интерфейсом компьютера? Да, адрес совпал.
2. Совпадает ли MAC-адрес назначения в программе Wireshark с MAC-адресом источника? Да, адрес также совпал
3. Как ваш ПК определил MAC-адрес другого устройства, с которого был отправлен эхо-запрос с помощью команды ping? Через ARP запрос с одного устройства на другое. (У кого такой адрес?)

2.0.2 Сбор и анализ данных протокола ICMP по удаленным узлам

Попробуем пингануть сайты зарубежных сми - The Economist (Великобритания), FowNews (США) и Spiegel (Германия). (Рисунок 2.2)

The image shows a Windows command prompt window on the left and a Wireshark network analyzer window on the right. The command prompt displays the results of three ping commands: one to www.economist.com, one to www.foxnews.com, and one to www.spiegel.de. Each command shows the exchange of four packets and the resulting statistics. The Wireshark window shows a packet capture of the first ping to www.economist.com. The packet list pane shows four ICMP Echo (ping) request and reply packets. The packet details pane shows the structure of the first packet, including the Ethernet II header, Internet Protocol Version 4 header, and Internet Control Message Protocol header. The packet bytes pane shows the raw data of the packet.

Command Prompt Output:

```
C:\Users\andrz>ping www.economist.com

Обмен пакетами с www.economist.com [104.18.42.19] с 32 байтами данных:
Ответ от 104.18.42.19: число байт=32 время=8мс TTL=59
Ответ от 104.18.42.19: число байт=32 время=10мс TTL=59
Ответ от 104.18.42.19: число байт=32 время=7мс TTL=59
Ответ от 104.18.42.19: число байт=32 время=13мс TTL=59

Статистика Ping для 104.18.42.19:
Пакетов: отправлено = 4, получено = 4, потеряно = 0
(0% потеря)
Приблизительное время приема-передачи в мс:
Минимальное = 7мсек, Максимальное = 13 мсек, Среднее = 9 мсек

C:\Users\andrz>ping www.foxnews.com

Обмен пакетами с j.sni.global.fastly.net [146.75.118.132] с 32 байтами данн
х:
Ответ от 146.75.118.132: число байт=32 время=41мс TTL=59
Ответ от 146.75.118.132: число байт=32 время=42мс TTL=59
Ответ от 146.75.118.132: число байт=32 время=41мс TTL=59
Ответ от 146.75.118.132: число байт=32 время=45мс TTL=59

Статистика Ping для 146.75.118.132:
Пакетов: отправлено = 4, получено = 4, потеряно = 0
(0% потеря)
Приблизительное время приема-передачи в мс:
Минимальное = 41мсек, Максимальное = 45 мсек, Среднее = 42 мсек

C:\Users\andrz>ping www.spiegel.de

Обмен пакетами с aacfb9d106f4.link11.de [128.65.210.183] с 32 байтами данн
х:
Ответ от 128.65.210.183: число байт=32 время=47мс TTL=54
Ответ от 128.65.210.183: число байт=32 время=46мс TTL=54
Ответ от 128.65.210.183: число байт=32 время=47мс TTL=54
Ответ от 128.65.210.183: число байт=32 время=45мс TTL=54

Статистика Ping для 128.65.210.183:
Пакетов: отправлено = 4, получено = 4, потеряно = 0
(0% потеря)
Приблизительное время приема-передачи в мс:
Минимальное = 45мсек, Максимальное = 47 мсек, Среднее = 46 мсек

C:\Users\andrz>
```

Wireshark Packet Capture Details:

No.	Time	Source	Destination	Protocol	Length	Info
427	3.449701	192.168.5.10	104.18.42.19	ICMP	74	Echo (ping) request id=0x00
428	3.457716	104.18.42.19	192.168.5.10	ICMP	74	Echo (ping) reply id=0x00
431	4.459320	192.168.5.10	104.18.42.19	ICMP	74	Echo (ping) request id=0x00
432	4.469168	104.18.42.19	192.168.5.10	ICMP	74	Echo (ping) reply id=0x00
439	5.473906	192.168.5.10	104.18.42.19	ICMP	74	Echo (ping) request id=0x00
440	5.480933	104.18.42.19	192.168.5.10	ICMP	74	Echo (ping) reply id=0x00
445	6.497287	192.168.5.10	104.18.42.19	ICMP	74	Echo (ping) request id=0x00
446	6.510317	104.18.42.19	192.168.5.10	ICMP	74	Echo (ping) reply id=0x00
1671	13.504893	192.168.5.10	146.75.118.132	ICMP	74	Echo (ping) request id=0x00
1673	13.546235	146.75.118.132	192.168.5.10	ICMP	74	Echo (ping) reply id=0x00
1689	14.522493	192.168.5.10	146.75.118.132	ICMP	74	Echo (ping) request id=0x00
1690	14.564604	146.75.118.132	192.168.5.10	ICMP	74	Echo (ping) reply id=0x00
1707	15.536781	192.168.5.10	146.75.118.132	ICMP	74	Echo (ping) request id=0x00
1708	15.578299	146.75.118.132	192.168.5.10	ICMP	74	Echo (ping) reply id=0x00
1713	16.542274	192.168.5.10	146.75.118.132	ICMP	74	Echo (ping) request id=0x00
1714	16.587600	146.75.118.132	192.168.5.10	ICMP	74	Echo (ping) reply id=0x00
1749	19.759722	192.168.5.10	128.65.210.183	ICMP	74	Echo (ping) request id=0x00
1750	19.806749	128.65.210.183	192.168.5.10	ICMP	74	Echo (ping) reply id=0x00
1757	20.783209	192.168.5.10	128.65.210.183	ICMP	74	Echo (ping) request id=0x00
1758	20.829665	128.65.210.183	192.168.5.10	ICMP	74	Echo (ping) reply id=0x00
1771	21.798560	192.168.5.10	128.65.210.183	ICMP	74	Echo (ping) request id=0x00
1779	21.845422	128.65.210.183	192.168.5.10	ICMP	74	Echo (ping) reply id=0x00
1785	22.812226	192.168.5.10	128.65.210.183	ICMP	74	Echo (ping) request id=0x00
1789	22.857704	128.65.210.183	192.168.5.10	ICMP	74	Echo (ping) reply id=0x00

Рисунок 2.2 — Ping сайтов зарубежных сми

При этом URL преобразовывались в ip адреса при помощи DNS, тогда посмотрим ip и mac адреса для данных сайтов:

1. www.economist.com

ip - 104.18.42.19

mac - 8c:68:c8:d7:45:26

2. www.foxnews.com

ip - 146.75.118.132

mac - 8c:68:c8:d7:45:26

3. www.spiegel.de

ip - 128.65.210.183

mac - 8c:68:c8:d7:45:26

Почему программа Wireshark показывает фактические MAC-адреса локальных узлов, но не показывает фактические MAC-адреса удаленных узлов?

MAC-адреса используются только для коммутации в локальной сети, при маршрутизации mac-адреса подменяются на mac-адрес маршрутизатора, через который проходят пакеты.

3 Анализ полей ТСП

Нам необходимо подключиться к FTP-серверу и проанализировать установление ТСП соединения. Для этого выбран учебный сервер с адресом 109.167.241.255 (Рисунок 3.1).

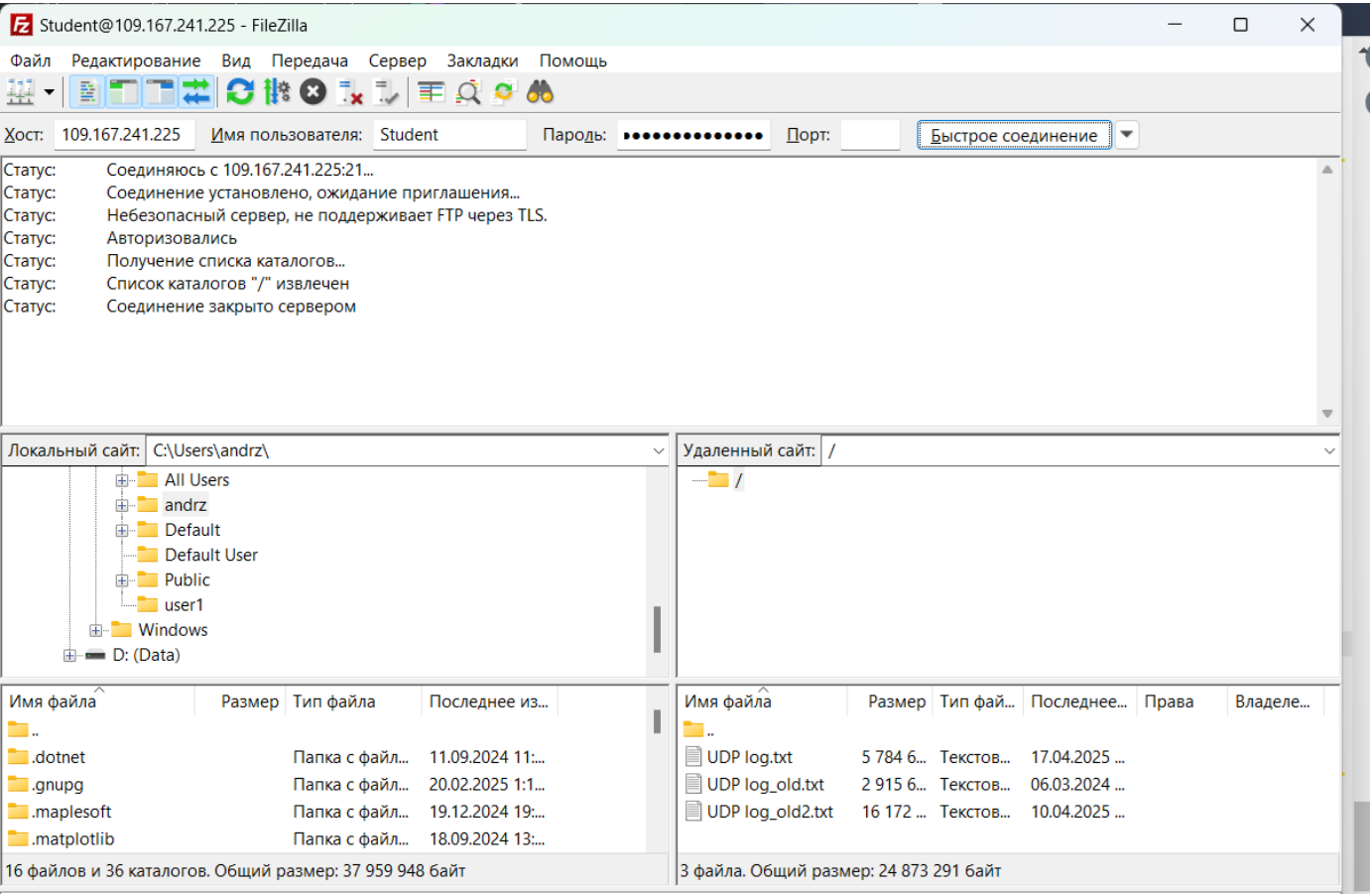


Рисунок 3.1 — FTP-сервер

Я отфильтровал пакеты, чтобы первыми были пакеты трехстороннего квинтирования. (Рисунок 3.2).

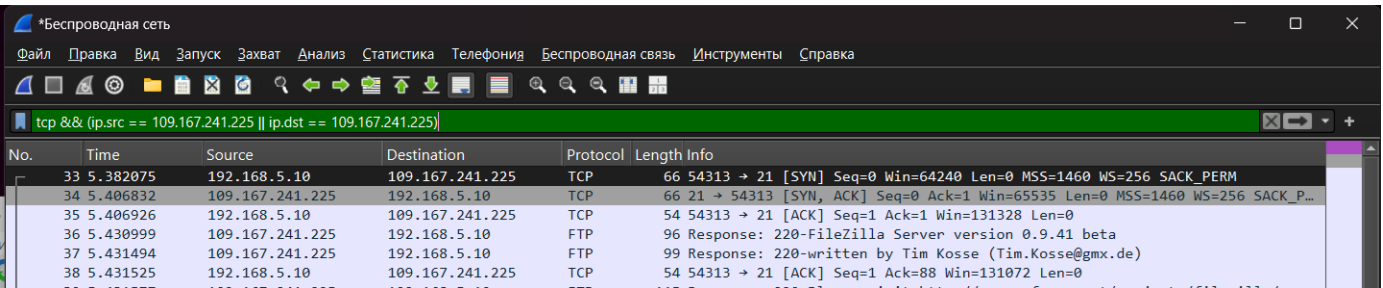


Рисунок 3.2 — Трехстороннее квинтирование

Теперь посмотрим подробнее на каждый из трех TCP пакетов, которые устанавливают соединение.

Первый TCP пакет (Рисунок 3.3):

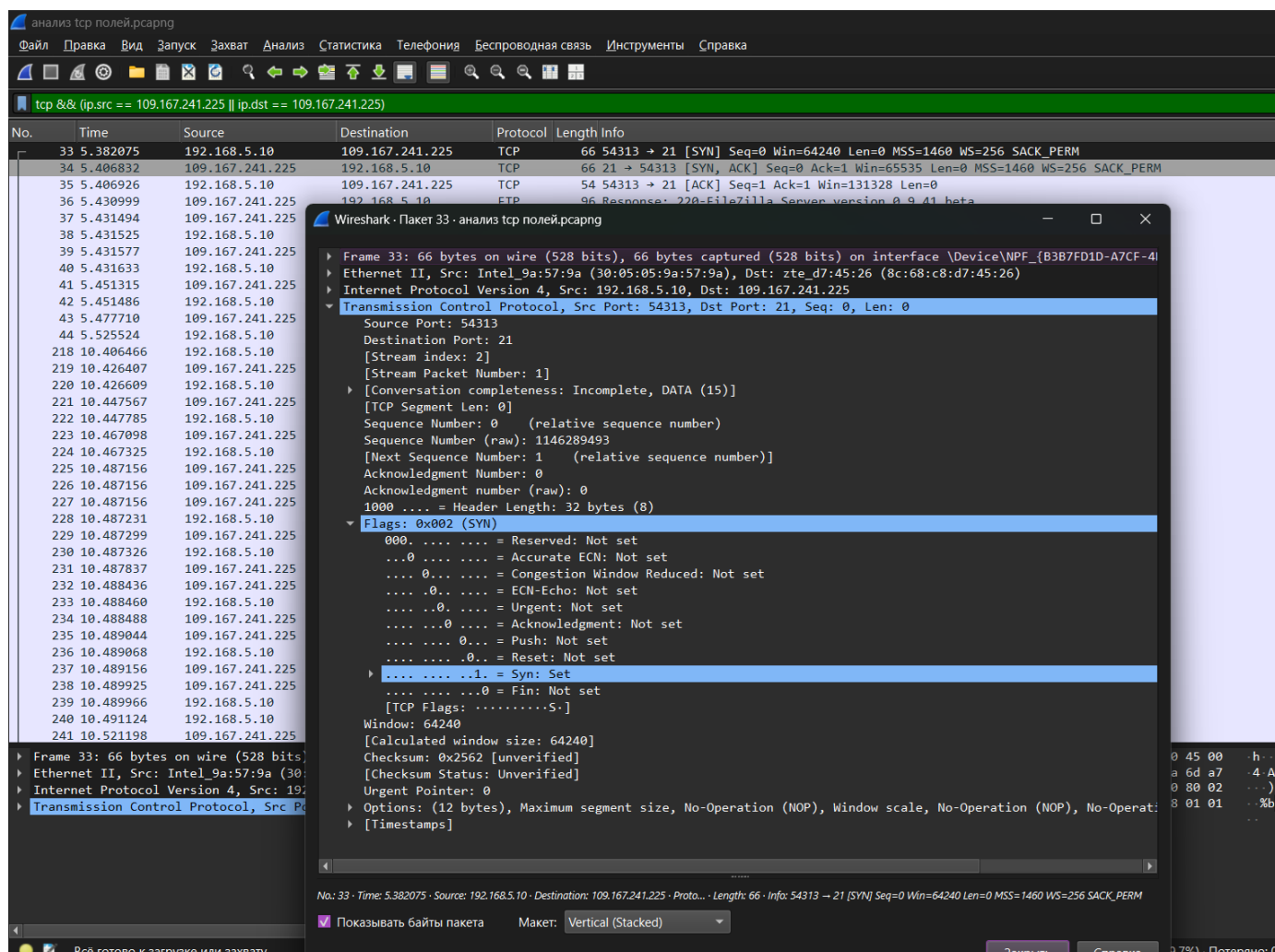


Рисунок 3.3 — Первый TCP пакет

Название поля	Значение поля
IP-адрес источника	192.168.5.10
IP-адрес назначения	109.167.241.225
Номер порта источника	54313
Номер порта назначения	21
Порядковый номер	0
Номер подтверждения	0
Длина заголовка	32
Размер окна	64240

Второй TCP пакет (Рисунок 3.4):

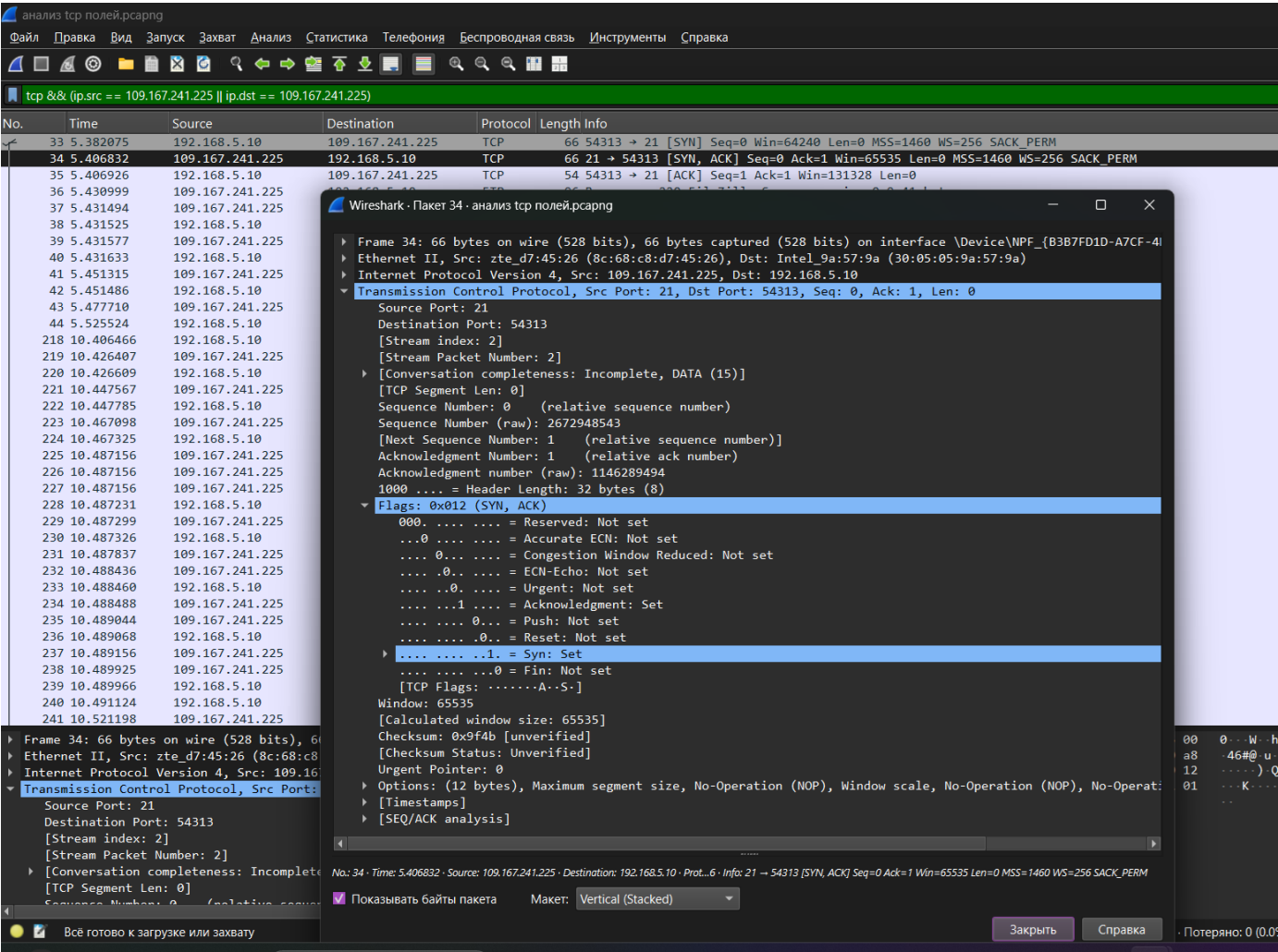


Рисунок 3.4 — Второй TCP пакет

Название поля	Значение поля
IP-адрес источника	109.167.241.225
IP-адрес назначения	192.168.5.10
Номер порта источника	21
Номер порта назначения	54313
Порядковый номер	0
Номер подтверждения	1
Длина заголовка	32
Размер окна	65535

Третий TCP пакет (Рисунок 3.5):

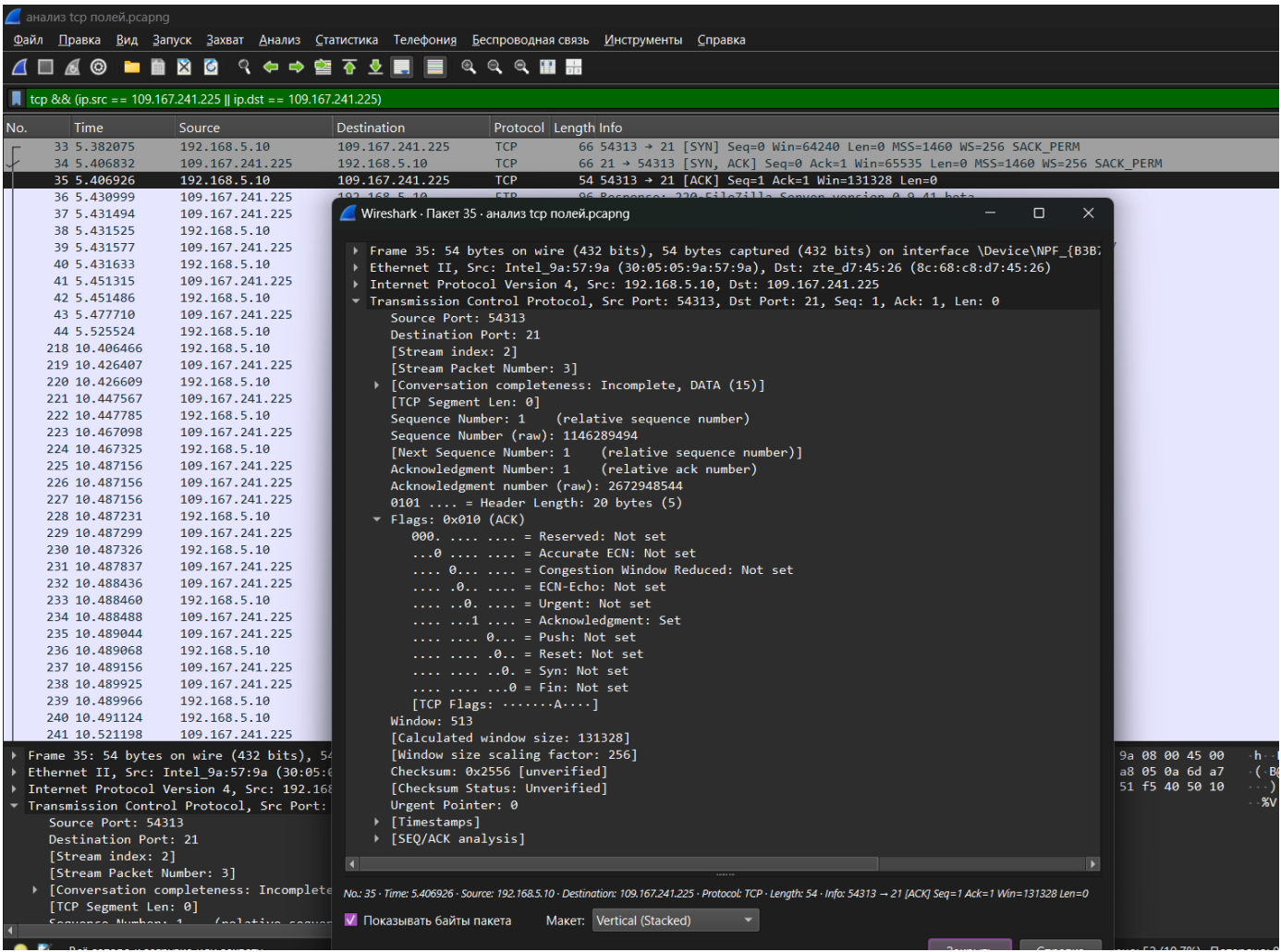


Рисунок 3.5 — Третий TCP пакет

Название поля	Значение поля
IP-адрес источника	192.168.5.10
IP-адрес назначения	109.167.241.225
Номер порта источника	54313
Номер порта назначения	21
Порядковый номер	1
Номер подтверждения	1
Длина заголовка	20
Размер окна	513

После установления TCP соединения FTP-сервер отправляет сообщение response:220 и отправляется подтверждение сеансу TCP на сервере. (Рисунок 3.6)

5.382075	192.168.5.10	109.167.241.225	TCP	66 54313 → 21 [SYN] Seq=0 Win=64...
5.406832	109.167.241.225	192.168.5.10	TCP	66 21 → 54313 [SYN, ACK] Seq=0 A...
5.406926	192.168.5.10	109.167.241.225	TCP	54 54313 → 21 [ACK] Seq=1 Ack=1 ...
5.430999	109.167.241.225	192.168.5.10	FTP	96 Response: 220-FileZilla Serve...
5.431494	109.167.241.225	192.168.5.10	FTP	99 Response: 220-written by Tim ...
5.431525	192.168.5.10	109.167.241.225	TCP	54 54313 → 21 [ACK] Seq=1 Ack=88...
5.431577	109.167.241.225	192.168.5.10	FTP	115 Response: 220 Please visit ht...
5.431633	192.168.5.10	109.167.241.225	FTP	64 Request: AUTH TLS
5.451315	109.167.241.225	192.168.5.10	FTP	94 Response: 502 SSL/TLS authent...
5.451486	192.168.5.10	109.167.241.225	FTP	64 Request: AUTH SSL
5.477710	109.167.241.225	192.168.5.10	FTP	94 Response: 502 SSL/TLS authent...
5.525524	192.168.5.10	109.167.241.225	TCP	54 54313 → 21 [ACK] Seq=21 Ack=2...

Рисунок 3.6 — Response 220 и подтверждение

После передается FTP трафик (Рисунок 3.7).

No.	Time	Source	Destination	Protocol	Length	Info
36	5.430999	109.167.241.225	192.168.5.10	FTP	96	Response: 220-FileZilla Server version 0.9.41 beta
37	5.431494	109.167.241.225	192.168.5.10	FTP	99	Response: 220-written by Tim Kosse (Tim.Kosse@gmx...
39	5.431577	109.167.241.225	192.168.5.10	FTP	115	Response: 220 Please visit http://sourceforge.net/...
40	5.431633	192.168.5.10	109.167.241.225	FTP	64	Request: AUTH TLS
41	5.451315	109.167.241.225	192.168.5.10	FTP	94	Response: 502 SSL/TLS authentication not allowed
42	5.451486	192.168.5.10	109.167.241.225	FTP	64	Request: AUTH SSL
43	5.477710	109.167.241.225	192.168.5.10	FTP	94	Response: 502 SSL/TLS authentication not allowed
218	10.406466	192.168.5.10	109.167.241.225	FTP	68	Request: USER Student
219	10.426407	109.167.241.225	192.168.5.10	FTP	89	Response: 331 Password required for student
220	10.426609	192.168.5.10	109.167.241.225	FTP	77	Request: PASS FksG5\$%^rgtdSDFH
221	10.447567	109.167.241.225	192.168.5.10	FTP	69	Response: 230 Logged on
222	10.447785	192.168.5.10	109.167.241.225	FTP	60	Request: SYST
223	10.467098	109.167.241.225	192.168.5.10	FTP	86	Response: 215 UNIX emulated by FileZilla
224	10.467325	192.168.5.10	109.167.241.225	FTP	60	Request: FEAT
225	10.487156	109.167.241.225	192.168.5.10	FTP	69	Response: 211-Features:
226	10.487156	109.167.241.225	192.168.5.10	FTP	61	Response: MDTM
227	10.487156	109.167.241.225	192.168.5.10	FTP	68	Response: REST STREAM
229	10.487299	109.167.241.225	192.168.5.10	FTP	61	Response: SIZE
231	10.487837	109.167.241.225	192.168.5.10	FTP	82	Response: MLST type*;size*;modify*;
232	10.488436	109.167.241.225	192.168.5.10	FTP	61	Response: MLSD
234	10.488488	109.167.241.225	192.168.5.10	FTP	61	Response: UTF8
235	10.489044	109.167.241.225	192.168.5.10	FTP	61	Response: CLNT
237	10.489156	109.167.241.225	192.168.5.10	FTP	61	Response: MFMT
238	10.489925	109.167.241.225	192.168.5.10	FTP	63	Response: 211 End
240	10.491124	192.168.5.10	109.167.241.225	FTP	59	Request: PWD
241	10.521198	109.167.241.225	192.168.5.10	FTP	85	Response: 257 "/" is current directory.
242	10.521982	192.168.5.10	109.167.241.225	FTP	62	Request: TYPE I
243	10.550308	109.167.241.225	192.168.5.10	FTP	73	Response: 200 Type set to I
244	10.550636	192.168.5.10	109.167.241.225	FTP	60	Request: PASV
245	10.578263	109.167.241.225	192.168.5.10	FTP	105	Response: 227 Entering Passive Mode (109,167,241,2...
246	10.578994	192.168.5.10	109.167.241.225	FTP	60	Request: MLSD
250	10.626828	109.167.241.225	192.168.5.10	FTP	79	Response: 150 Connection accepted
252	10.629894	109.167.241.225	192.168.5.10	FTP	71	Response: 226 Transfer OK

Рисунок 3.7 — FTP трафик

В конце когда требуется завершить сеанс FTP сервер отправляет сегмент с флагом FIN, потом клиент отвечает с флагом подтверждения ACK, после отправляет сегмент с флагом FiN для завершения соединения и сервер отвечает с флагом ACK подтверждения. После сеанс завершается. (Рисунок 3.8).

Рисунок 3.8 — Завершение ТСР соединения

ЗАКЛЮЧЕНИЕ

В ходе выполнения данной лабораторной работы были освоены основные принципы работы стека TCP/IP, проведён анализ передаваемых и принимаемых пакетов, получены навыки сбора сетевого трафика с использованием программы Wireshark, а также приобретён опыт фильтрации захваченного трафика, поиска и анализа сетевых соединений.