

НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО
ФАКУЛЬТЕТ ПРОГРАММНОЙ ИНЖЕНЕРИИ И КОМПЬЮТЕРНОЙ ТЕХНИКИ
НАПРАВЛЕНИЕ ПРОГРАММНАЯ ИНЖЕНЕРИЯ
ОБРАЗОВАТЕЛЬНАЯ ПРОГРАММА СИСТЕМНОЕ И ПРИКЛАДНОЕ
ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ
СПЕЦИАЛИЗАЦИЯ СИСТЕМНОЕ ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ

ОТЧЕТ ПО ДОМАШНЕЙ РАБОТЕ № 4
курса «Компьютерные сети»
по теме: «Работа с инструментом Wireshark
и анализ сетевого трафика»

Выполнил студент:
Тюрин Иван Николаевич
группа: Р33102

Преподаватель:
Авксентьева Е. Ю.,
Алиев Т. И.

Санкт-Петербург, 2024 г.

Содержание

Лабораторная работа № 4. Работа с инструментом Wireshark и анализ сетевого трафика	2
1. Цель работы	2
2. Выполнение задания	4
1. Анализ трафика утилиты ping	4
2. Анализ трафика утилиты tracert (traceroute)	5
3. Анализ HTTP-трафика	6
4. Анализ DNS-трафика	7
5. Анализ ARP-трафика	8
6. Анализ трафика утилиты nslookup	9
7. Анализ FTP-трафика	9
8. Анализ DHCP-трафика	9
9. Анализ Telegram-трафика	10
3. Вывод	11

Лабораторная работа № 4

Работа с инструментом Wireshark

и анализ сетевого трафика

1. Цель работы

Цель работы – изучить структуру протокольных блоков данных, анализируя реальный трафик на компьютере студента с помощью бесплатно распространяемой утилиты Wireshark.

В процессе выполнения домашнего задания выполняются наблюдения за передаваемым трафиком с компьютера пользователя в Интернет и в обратном направлении. Применение специализированной утилиты Wireshark позволяет наблюдать структуру передаваемых кадров, пакетов и сегментов данных различных сетевых протоколов. При выполнении УИР рекомендуется выполнить анализ последовательности команд и определить назначение служебных данных, используемых для организации обмена данными в протоколах: ARP, DNS, FTP, HTTP, DHCP.

Задание

1. Запустить Wireshark (иногда для этого требуются права Администратора). В появившемся окне выбрать интерфейс, для которого необходимо осуществлять анализ проходящих через него пакетов. В качестве интерфейса, используемого для захвата трафика, выбрать физический адаптер, через который компьютер подключён к Интернету (обычно этот адаптер называется Local или «Подключение по локальной сети»). Если меню для выбора адаптера не появляется при запуске Wireshark, нужно запустить из «Меню» команду «Capture > Options». После выбора адаптера, нужно запустить процесс захвата трафика (кнопка Start).
2. Инициировать процесс передачи трафика по сети (например, в браузере открыть сайт, заданный по варианту, или запустить соответствующую сетевую утилиту – см. ниже).

3. Установить значение «Фильтра», чтобы из всего множества перехватываемых пакетов Wireshark отобразил только те, которые имеют отношение к выполняемому заданию. Для корректного создания фильтра следует пользоваться всплывающими подсказками Wireshark, которые активизируются при наборе фильтра. В качестве альтернативного способа можно использовать интерактивный конструктор фильтра, нажав на кнопку «Expression» в правой части элемента «Фильтр».
4. Дождаться появления данных в списке захваченных пакетов и убедиться, что количество пакетов достаточно для выполнения задания.
5. Сохранить захваченный трафик в файл-трассу (pcap). Указанный файл нужно предъявить по первому требованию преподавателя во время защиты, если в этом возникнет необходимость.
6. Описать в отчёте структуру наблюдаемых PDU (кадров, пакетов, сегментов) как для запросов, так и ответов. Указать название и назначение всех заголовков всех уровней OSI-модели в пакетах с учётом порядка инкапсуляции (для этого нужно раскрывать соответствующие значки «+» в поле с детальной информацией о выбранном пакете).
7. Написать в отчёте ответы на вопросы задания (для этого может потребоваться самостоятельно изучить назначение соответствующей заданию сетевой утилиты, использованной для создания трафика).
8. Поместить в отчёт скриншоты окна Wireshark, иллюстрирующие ответы из вышеуказанных п.6 и п.7.

Адрес сайта, в котором по очереди встречаются инициалы (ФИО) студента в латинской транскрипции:

Тюрин Иван Николаевич \rightarrow tin \rightarrow tinkoff.ru

Найдем нужный IP-адрес с помощью команды `nslookup tinkoff.ru`.
Получаем IPv4 адрес `178.248.236.218`.
Для Wireshark используем маску `ip.addr == 178.248.236.218`.

2. Выполнение задания

2. 1. Анализ трафика утилиты ping

Команды для выполнения: `ping -l 100 tinkoff.ru .`

Shell с командой ping: [1.1](#); результат захвата трафика в Wireshark: [1.2](#).

1. Имеет ли место фрагментация исходного пакета, какое поле на это указывает? — Да, при больших пакетах, появились фрагменты Fragmented IP protocol
2. Какая информация указывает, является ли фрагмент пакета последним или промежуточным? — Флаг MF = 0.

```
-----
▼ 000. .... = Flags: 0x0
    0... .... = Reserved bit: Not set
    .0.. .... = Don't fragment: Not set
    ..0. .... = More fragments: Not set
```

Рис. 1.3: ping MF

3. Чему равно количество фрагментов при передаче ping-пакетов? —
3) Количество фрагментов при передаче ping-пакетов может изменяться в зависимости от размера пакета и максимального размера передаваемых данных
4. Построить график, в котором на оси абсцисс находится размер_пакета, а по оси ординат — количество фрагментов, на которое был разделён каждый ping-пакет.

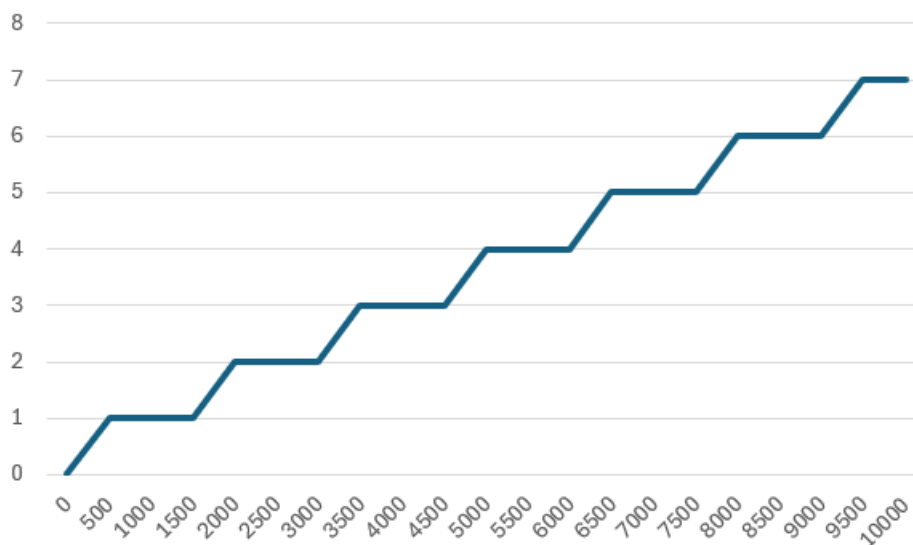


Рис. 1.4: Фрагментация ping-пакетов

5. Как изменить поле TTL с помощью утилиты ping? — Добавить ключ `-i`
6. Что содержится в поле данных ping-пакета? — Символы английского алфавита

2. 2. Анализ трафика утилиты tracert (traceroute)

Команда для выполнения: `tracert -d tinkoff.ru`.

Терминал с командой `tracert`: [1.5](#); результат захвата трафика в Wireshark: [1.6](#).

1. Сколько байт содержится в заголовке IP? Сколько байт содержится в поле данных? — $20 + 64$ байт

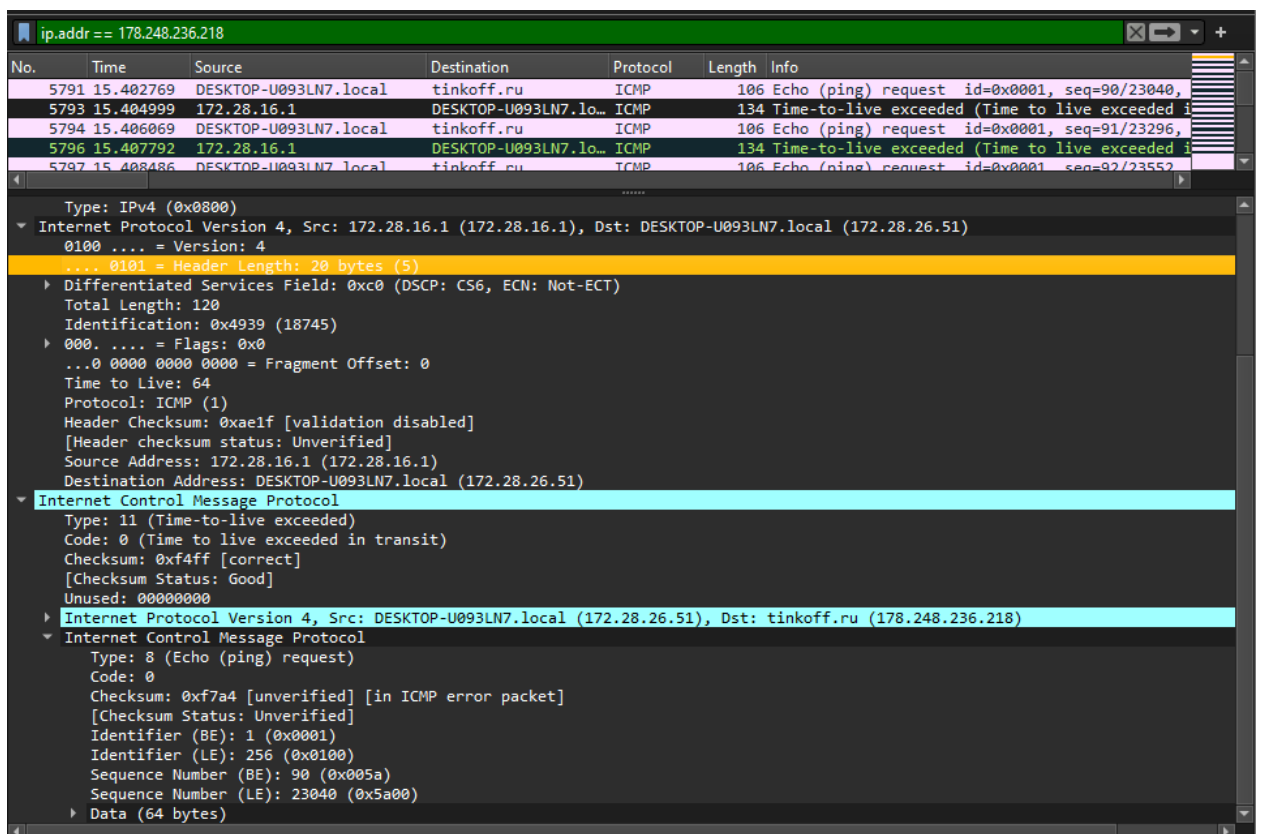


Рис. 1.7: Tracert data size

2. Как и почему изменяется поле TTL в следующих друг за другом ICMP пакетах tracert? Для ответа на этот вопрос нужно проследить изменение TTL при передаче по маршруту, состоящему из более чем двух хопов. — Увеличивается на 1 каждый 3 пакет, для выявления расстояния в хопх.

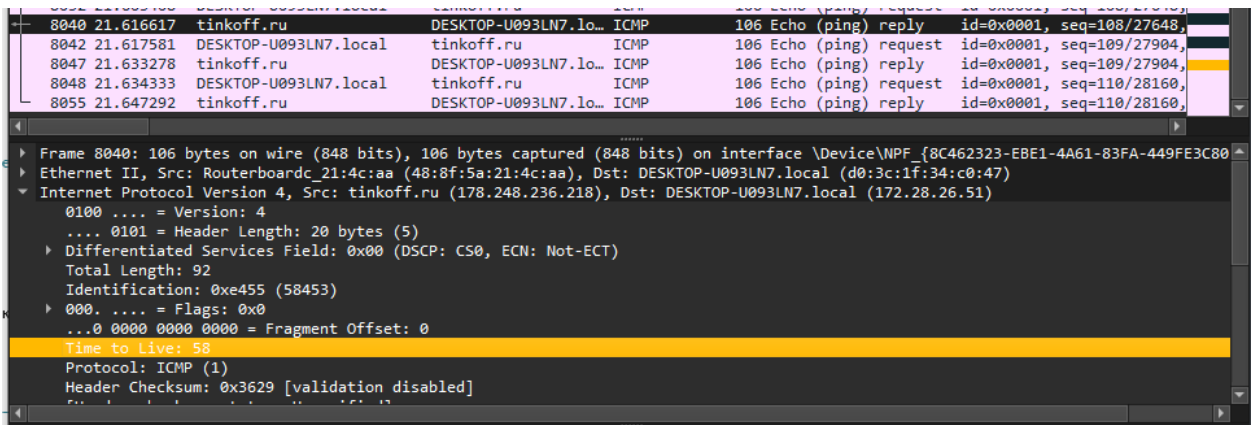


Рис. 1.8: Tracert TTL

3. Чем отличаются ICMP-пакеты, генерируемые утилитой tracert, от ICMP пакетов, генерируемых утилитой ping (см. предыдущее задание). — В tracert происходит увеличение TTL
4. Чем отличаются полученные пакеты «ICMP reply» от «ICMP error» и зачем нужны оба этих типа ответов? — Различные значения в поле TYPE. ICMP reply: получение ответного сообщения; ICMP error: ошибка.
5. Что изменится в работе tracert, если убрать ключ «-d»? Какой дополнительный трафик при этом будет генерироваться? — ключ -d используется для того, чтобы предотвратить преобразование IP-адресов в имена хостов, если убрать ключ -d при использовании tracert, то утилита будет пытаться разрешать DNS-имена для каждого узла на пути

2. 3. Анализ HTTP-трафика

Результат перехвата трафика в Wireshark: 1.9. Как можно видеть, HTTP-трафик скрывается за TLS, так что для просмотра заголовков GET-запроса нужно отслеживать сайт без TLS или каким-то образом добавить свои «debug»-сертификаты в Wireshark.

Для сайта без TLS результат будет примерно такой:

```

> Internet Protocol Version 4, Src: 192.168.0.103, Dst: 195.24.68.26
> Transmission Control Protocol, Src Port: 53024, Dst Port: 80, Seq: 1578, Ack: 7582, Len: 819
▼ Hypertext Transfer Protocol
  ▼ GET / HTTP/1.1\r\n
    > [Expert Info (Chat/Sequence): GET / HTTP/1.1\r\n]
      Request Method: GET
      Request URI: /
      Request Version: HTTP/1.1
    Host: www.newart.ru\r\n
    Connection: keep-alive\r\n
    Cache-Control: max-age=0\r\n
    Upgrade-Insecure-Requests: 1\r\n
    User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrom
    Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*
    Accept-Encoding: gzip, deflate\r\n
    Accept-Language: ru,en;q=0.9,en-GB;q=0.8,en-US;q=0.7\r\n

> Transmission Control Protocol, Src Port: 80, Dst Port: 53024, Seq: 13182, Ack: 2397, Len: 5
> [5 Reassembled TCP Segments (5605 bytes): #50664(1446), #50665(1460), #50666(1460), #50668(1234), #5
▼ Hypertext Transfer Protocol, has 3 chunks (including last chunk)
  ▼ HTTP/1.1 200 OK\r\n
    > [Expert Info (Chat/Sequence): HTTP/1.1 200 OK\r\n]
      Response Version: HTTP/1.1
      Status Code: 200
      [Status Code Description: OK]
      Response Phrase: OK
    Server: openresty\r\n
    Date: Sun, 19 May 2024 14:08:26 GMT\r\n
    Content-Type: text/html; charset=windows-1251\r\n
    Transfer-Encoding: chunked\r\n
    Connection: keep-alive\r\n
    ..

```

2. 4. Анализ DNS-трафика

Воспользуемся командой для сброса DNS-настроек в системе: `ipconfig /flushdns` (рис. 1.10).

Результат перехвата трафика в Wireshrk в принципе совпадает с результатом предыдущего задания.

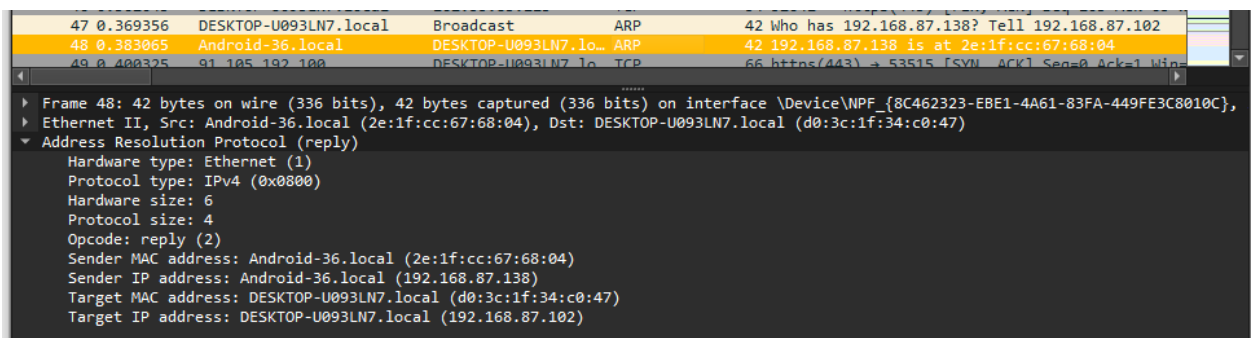
1. Почему адрес, на который отправлен DNS-запрос, не совпадает с адресом посещаемого сайта? — Адрес отправки соответствует шлюзу по умолчанию, так как очищен кэш и нужно получить с DNS сервера адрес запрашиваемого сайта.
2. Какие бывают типы DNS-запросов? —
 - *Итеративный запрос* посылает доменное имя DNS-серверу и просит вернуть либо IP-адрес этого домена, либо имя DNS-сервера, авторитативного для этого домена. При этом сервер DNS не опрашивает другие серверы для получения ответа.
 - *Рекурсивный запрос* посылает DNS-серверу доменное имя и просит возвратить IP-адрес запрошенного домена. При этом сервер может обращаться к другим DNS-серверам.

- *Обратный запрос* посылает IP и просит вернуть доменное имя.
3. В какой ситуации нужно выполнять независимые DNS-запросы для получения содержащихся на сайте изображений? — Когда картинки лежат на другом доменном имени.

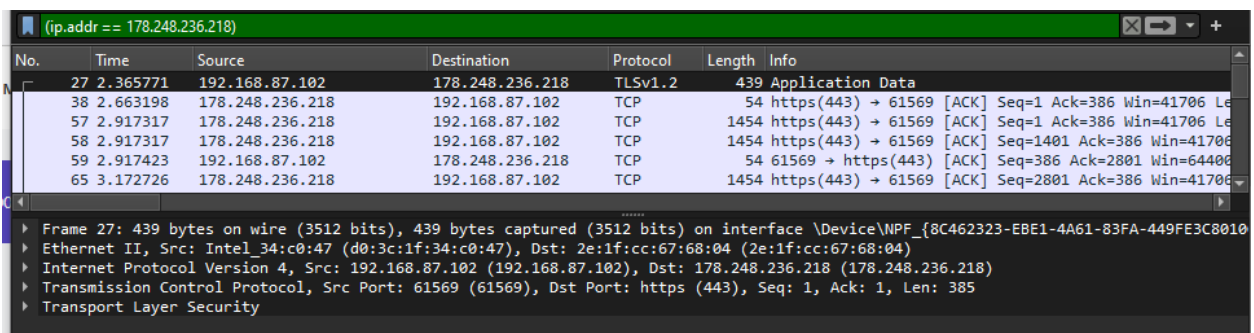
2. 5. Анализ ARP-трафика

Очистим таблицу ARP с помощью команды: `netsh interface ip delete arpcache` а проверим результат с помощью `arp -a`. Результат очистки кажется неоднозначным, т.к. почти сразу там появляются новые записи, см. рис. 1.12.

1. Какие MAC-адреса присутствуют в захваченных пакетах ARP протокола? Что означают эти адреса? Какие устройства они идентифицируют? — хостовой компьютер и мобильную точку доступа в телефоне.



2. Какие MAC-адреса присутствуют в захваченных HTTP-пакетах и что означают эти адреса? Что означают эти адреса? Какие устройства они идентифицируют? — адрес отправляемого устройства, адрес принимающего устройства



3. Для чего ARP-запрос содержит IP-адрес источника? — Что бы добавить информацию о узле в ARP таблицу

2. 6. Анализ трафика утилиты nslookup

Выполним следующие команды: `nslookup tinkoff.ru` и `nslookup -type=NS tinkoff.ru`. Результат выполнения смотри на рис. 1.13.

1. Чем различается трасса трафика в п.2 и п.4, указанных выше? — При запуске в п.2 утилита ищет IP-адрес хоста (запись типа A (IPv4) или AAAA (IPv6)). При запуске в п.4 утилита ищет Name Server для запрашиваемого хоста.
2. Что содержится в поле «Answers» DNS-ответа? — Данные запрашиваемого типа DNS-записи: для A - IPv4-адрес, для NS - список authoritative Name Server
3. Каковы имена серверов, возвращающих авторитативный (authoritative) отклик? — Авторитативный отклик возвращают серверы, которые являются ответственными за зону, в которой описана информация, необходимая DNS-клиенту

2. 7. Анализ FTP-трафика

Установим в Wireshark фильтр `ftp || ftp-data`.

1. Сколько байт данных содержится в пакете FTP-DATA? — Размер может быть любой, но не больше MTU.
2. Как выбирается порт транспортного уровня, который используется для передачи FTP-пакетов? — Для потока управления на сервере используется порт 21. Для передачи данных используется порт 20, если передача идет в активном режиме, либо с любого порта клиента к любому порту сервера в пассивном режиме.
3. Чем отличаются пакеты FTP от FTP-DATA? — FTP используется для выполнения команд (request/response), а FTP-DATA работает с файлами.

2. 8. Анализ DHCP-трафика

Установим в Wireshark фильтр `bootp`.

Выполним команды `ipconfig /release` для сброса IP и `ipconfig /renew` для обновления IP; результат см. на рис. 1.14 и рис. 1.15. Результат перехвата трафика в Wireshark можно видеть на рис. 1.16.

1. Чем различаются пакеты «DHCP Discover» и «DHCP Request»? — Оба запроса выполняются клиентом, DHCP Discover ищет DHCP-сервер в своей канальной среде, а DHCP Request принимает предлагаемый адрес и уведомляет DHCP-сервер об этом.
2. Как и почему менялись MAC- и IP-адреса источника и назначения в переданных DHCP-пакетах. — В качестве MAC-адреса источника клиент изначально подставил свой MAC-адрес, а MAC-адрес сервера он не знает, поэтому использует широковещательный MAC адрес. Соответственно, в заголовке IP-пакета в качестве адреса источника клиент использовал 0.0.0.0. При отправке Offer или ACK пакетов, адреса источника соответствуют адресам DHCP-сервера, адреса назначения широковещательные.
3. Каков IP-адрес DHCP-сервера? — 172.128.16.1

No.	Time	Source	Destination	Protocol	Length	Info
3915	16.878449	0c489a2d-0b5b-4c9f-a7e7...	255.255.255.255	DHCP	342	DHCP Discover - Transaction ID 0x19f543e2
4021	17.358263	0c489a2d-0b5b-4c9f-a7e7...	255.255.255.255	DHCP	342	DHCP Request - Transaction ID 0x843b9302
4338	18.591081	0c489a2d-0b5b-4c9f-a7e7...	255.255.255.255	DHCP	354	DHCP Request - Transaction ID 0x19f543e2
4544	25.242054	0c489a2d-0b5b-4c9f-a7e7...	255.255.255.255	DHCP	344	DHCP Discover - Transaction ID 0x3ebf83b1
4794	28.078405	0c489a2d-0b5b-4c9f-a7e7...	255.255.255.255	DHCP	342	DHCP Discover - Transaction ID 0x1147db96
4803	28.211432	172.28.16.1	DESKTOP-U093LN7.lo...	DHCP	342	DHCP Offer - Transaction ID 0x3ebf83b1
4805	28.212309	0c489a2d-0b5b-4c9f-a7e7...	255.255.255.255	DHCP	370	DHCP Request - Transaction ID 0x3ebf83b1
4806	28.216031	172.28.16.1	DESKTOP-U093LN7.lo...	DHCP	342	DHCP ACK - Transaction ID 0x3ebf83b1
4864	28.453407	0c489a2d-0b5b-4c9f-a7e7...	255.255.255.255	DHCP	364	DHCP Request - Transaction ID 0xb8df9f7
4866	28.475535	172.28.16.1	DESKTOP-U093LN7.lo...	DHCP	342	DHCP ACK - Transaction ID 0xb8df9f7
4909	28.662098	0c489a2d-0b5b-4c9f-a7e7...	255.255.255.255	DHCP	342	DHCP Request - Transaction ID 0x1fa2f0c3
5929	32.202860	0c489a2d-0b5b-4c9f-a7e7...	255.255.255.255	DHCP	332	DHCP Request - Transaction ID 0xb01f160b
6637	34.506122	0c489a2d-0b5b-4c9f-a7e7...	255.255.255.255	DHCP	338	DHCP Request - Transaction ID 0x71c66cb7
7433	39.930378	0c489a2d-0b5b-4c9f-a7e7...	255.255.255.255	DHCP	342	DHCP Request - Transaction ID 0x843b9305
8080	43.915855	0c489a2d-0b5b-4c9f-a7e7...	255.255.255.255	DHCP	354	DHCP Request - Transaction ID 0xe80c90f
8251	44.969426	0c489a2d-0b5b-4c9f-a7e7...	255.255.255.255	DHCP	342	DHCP Request - Transaction ID 0xafd9d82
8477	46.902504	0c489a2d-0b5b-4c9f-a7e7...	255.255.255.255	DHCP	342	DHCP Request - Transaction ID 0xafd9d82

4. Что произойдёт, если очистить использованный фильтр «bootp»? — Будут отображаться все пакеты.

2. 9. Анализ Telegram-трафика

- Чем различаются пакеты разных видов Telegram-трафика (текст, аудио, видео)? — Текстовые данные передаются с помощью TSL. При загрузке аудио и видео используются TCP и SSLv2. Во время аудио-звонков используется TCP с SSL.

dhcp							
No.	Time	Source	Destination	Protocol	Length	Info	
3915	16.878449	0c489a2d-0b5b-4c9f-a7e7...	255.255.255.255	DHCP	342	DHCP Discover	- Transaction ID 0x19f543e2
4021	17.358263	0c489a2d-0b5b-4c9f-a7e7...	255.255.255.255	DHCP	342	DHCP Request	- Transaction ID 0x843b9302
4338	18.591081	0c489a2d-0b5b-4c9f-a7e7...	255.255.255.255	DHCP	354	DHCP Request	- Transaction ID 0x19f543e2
4544	25.242054	0c489a2d-0b5b-4c9f-a7e7...	255.255.255.255	DHCP	344	DHCP Discover	- Transaction ID 0x3ebf83b1
4794	28.078405	0c489a2d-0b5b-4c9f-a7e7...	255.255.255.255	DHCP	342	DHCP Discover	- Transaction ID 0x1147db96
4803	28.211432	172.28.16.1	DESKTOP-U093LN7.lo...	DHCP	342	DHCP Offer	- Transaction ID 0x3ebf83b1
4805	28.212309	0c489a2d-0b5b-4c9f-a7e7...	255.255.255.255	DHCP	370	DHCP Request	- Transaction ID 0x3ebf83b1
4806	28.216031	172.28.16.1	DESKTOP-U093LN7.lo...	DHCP	342	DHCP ACK	- Transaction ID 0x3ebf83b1
4864	28.453407	0c489a2d-0b5b-4c9f-a7e7...	255.255.255.255	DHCP	364	DHCP Request	- Transaction ID 0xb8dfd9f7
4866	28.475535	172.28.16.1	DESKTOP-U093LN7.lo...	DHCP	342	DHCP ACK	- Transaction ID 0xb8dfd9f7
4909	28.662098	0c489a2d-0b5b-4c9f-a7e7...	255.255.255.255	DHCP	342	DHCP Request	- Transaction ID 0x1fa2f0c3
5929	32.202860	0c489a2d-0b5b-4c9f-a7e7...	255.255.255.255	DHCP	332	DHCP Request	- Transaction ID 0xb01f160b
6637	34.506122	0c489a2d-0b5b-4c9f-a7e7...	255.255.255.255	DHCP	338	DHCP Request	- Transaction ID 0x71c66cb7
7433	39.930378	0c489a2d-0b5b-4c9f-a7e7...	255.255.255.255	DHCP	342	DHCP Request	- Transaction ID 0x843b9305
8080	43.915855	0c489a2d-0b5b-4c9f-a7e7...	255.255.255.255	DHCP	354	DHCP Request	- Transaction ID 0xe80c90f
8251	44.969426	0c489a2d-0b5b-4c9f-a7e7...	255.255.255.255	DHCP	342	DHCP Request	- Transaction ID 0xafd9d82
8477	46.902504	0c489a2d-0b5b-4c9f-a7e7...	255.255.255.255	DHCP	342	DHCP Request	- Transaction ID 0xafd9d82

- Какой Wireshark-фильтр следует использовать для независимой идентификации Telegram-трафика разных видов (текст, аудио, видео)? — В Wireshark можно установить фильтр по протоколу, соответственно нужно установить `ts1`, `tcp`, `ssl`. При этом можно обнаружить, что весь трафик направляется на один (или несколько) IP адресов, поэтому можно еще фильтровать по нему, как я и сделал.

3. Вывод

В ходе лабораторной работы был проанализирован сетевой трафик с помощью программы Wireshark. Изучили, какие пакеты передаются при работе утилит `ping`, и `tracert` и какую информацию они содержат. Также был проведен анализ трафика HTTP запросов и влияние на него кэширования данных. Кэширование также влияет на работу DNS, во время выполнения работы нам необходимо было очистить кэши и посмотреть на работу DNS-запросов. Далее был рассмотрен трафик при выполнении `arp`-запросов, для этого нужно было очистить `arp`-таблицу. Кроме того был проанализирован трафик при работе с FTP. И последним был рассмотрен трафик Telegram, мы узнали, какие протоколы используются в нем для передачи разных типов данных

```
Nushell
~> ping -l 200 tinkoff.ru

Обмен пакетами с tinkoff.ru [178.248.236.218] с 200 байтами данных:
Ответ от 178.248.236.218: число байт=200 время=12мс TTL=58
Ответ от 178.248.236.218: число байт=200 время=12мс TTL=58
Ответ от 178.248.236.218: число байт=200 время=16мс TTL=58
Ответ от 178.248.236.218: число байт=200 время=12мс TTL=58

Статистика Ping для 178.248.236.218:
    Пакетов: отправлено = 4, получено = 4, потеряно = 0
    (0% потерь)
Приблизительное время приема-передачи в мс:
    Минимальное = 12мсек, Максимальное = 16 мсек, Среднее = 13 мсек
~> ping -l 400 tinkoff.ru

Обмен пакетами с tinkoff.ru [178.248.236.218] с 400 байтами данных:
Ответ от 178.248.236.218: число байт=400 время=86мс TTL=58
Превышен интервал ожидания для запроса.
Ответ от 178.248.236.218: число байт=400 время=13мс TTL=58
Ответ от 178.248.236.218: число байт=400 время=15мс TTL=58

Статистика Ping для 178.248.236.218:
    Пакетов: отправлено = 4, получено = 3, потеряно = 1
    (25% потерь)
Приблизительное время приема-передачи в мс:
    Минимальное = 13мсек, Максимальное = 86 мсек, Среднее = 38 мсек
~> ping -l 800 tinkoff.ru

Обмен пакетами с tinkoff.ru [178.248.236.218] с 800 байтами данных:
Ответ от 178.248.236.218: число байт=800 время=13мс TTL=58
Превышен интервал ожидания для запроса.
Ответ от 178.248.236.218: число байт=800 время=13мс TTL=58
Ответ от 178.248.236.218: число байт=800 время=17мс TTL=58

Статистика Ping для 178.248.236.218:
    Пакетов: отправлено = 4, получено = 3, потеряно = 1
    (25% потерь)
Приблизительное время приема-передачи в мс:
    Минимальное = 13мсек, Максимальное = 17 мсек, Среднее = 14 мсек
~> ping -l 1000 tinkoff.ru

Обмен пакетами с tinkoff.ru [178.248.236.218] с 1000 байтами данных:
Превышен интервал ожидания для запроса.
Превышен интервал ожидания для запроса.
Ответ от 178.248.236.218: число байт=1000 время=18мс TTL=58
Превышен интервал ожидания для запроса.

Статистика Ping для 178.248.236.218:
    Пакетов: отправлено = 4, получено = 1, потеряно = 3
    (75% потерь)
Приблизительное время приема-передачи в мс:
    Минимальное = 18мсек, Максимальное = 18 мсек, Среднее = 18 мсек
```

Рис. 1.1: Enter Caption


```
Nushell
~> tracert -d tinkoff.ru
20.05.2024 11:47:17

Трассировка маршрута к tinkoff.ru [178.248.236.218]
с максимальным числом прыжков 30:

 1      2 ms      1 ms      11 ms    172.28.16.1
 2      2 ms      2 ms       1 ms    77.234.199.66
 3      4 ms     14 ms       2 ms    87.248.228.102
 4     19 ms      9 ms     10 ms    139.45.238.84
 5     17 ms     24 ms     12 ms    139.45.243.3
 6     13 ms     15 ms     13 ms    139.45.226.166
 7     13 ms     15 ms     13 ms    178.248.236.218

Трассировка завершена.
~> |
20.05.2024 11:59:00
```

Рис. 1.5: Tracert в консоли

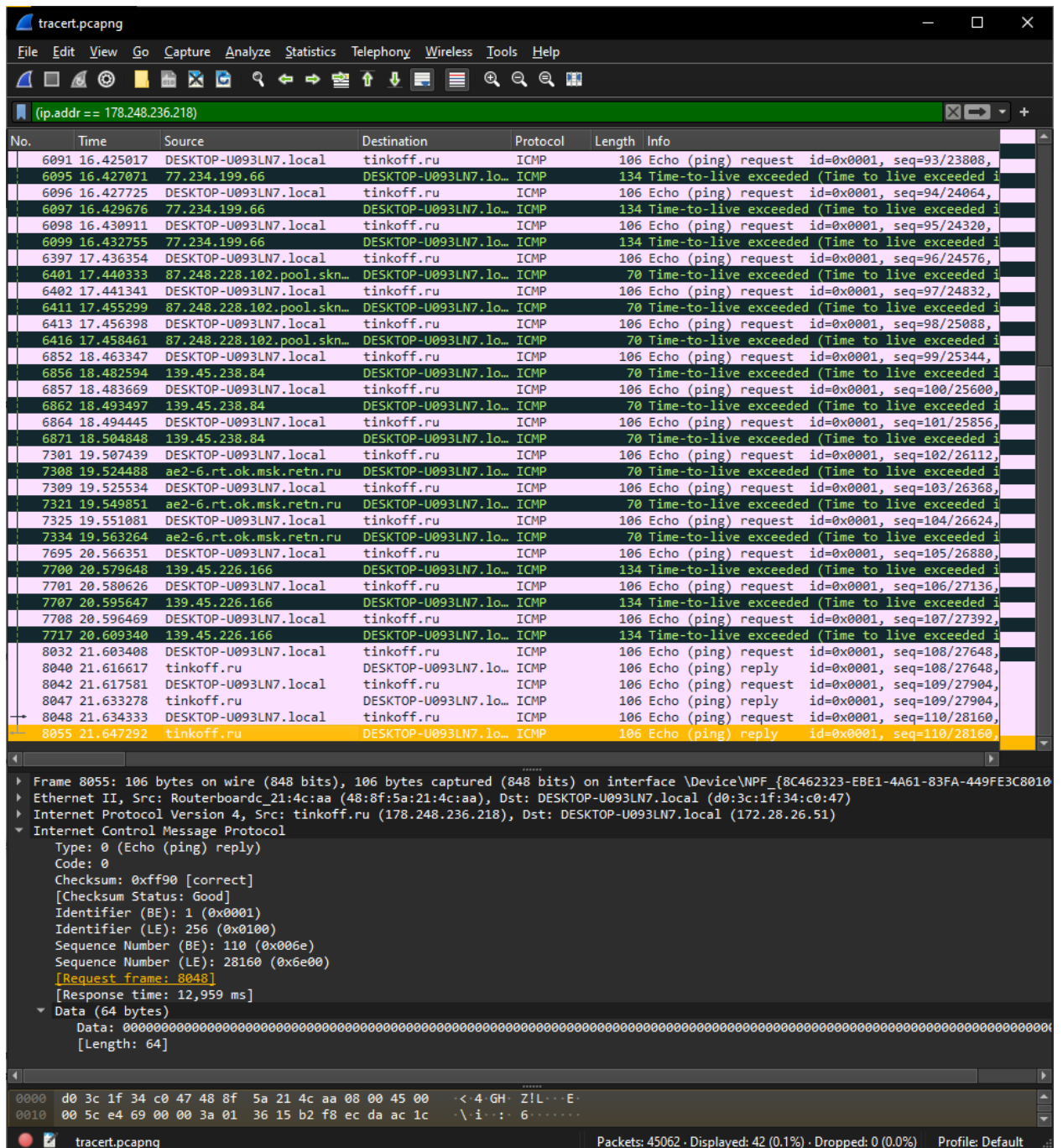


Рис. 1.6: Tracert в Wireshark

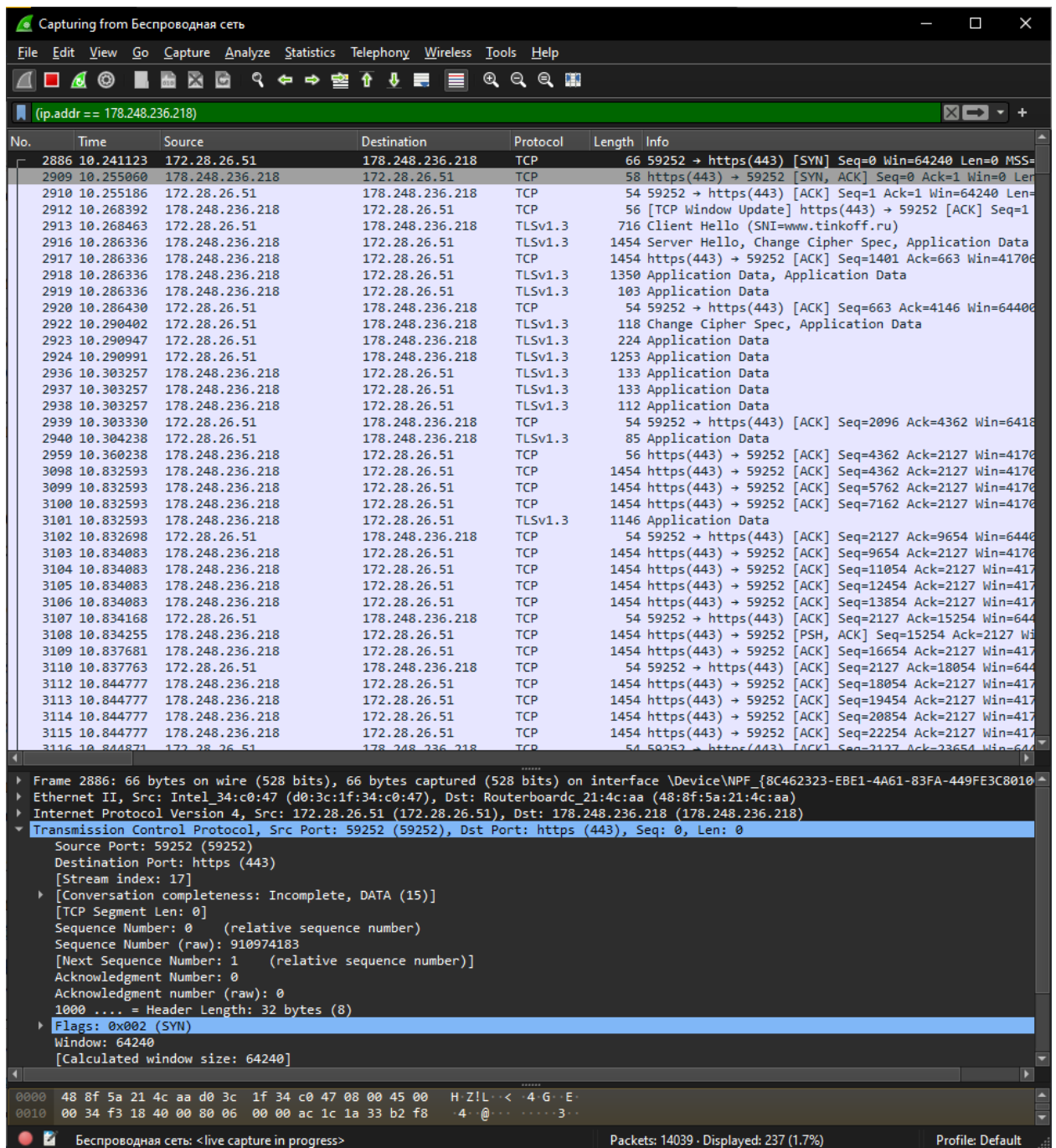


Рис. 1.9: Http в Wireshark



Рис. 1.10: Сброс настроек DNS

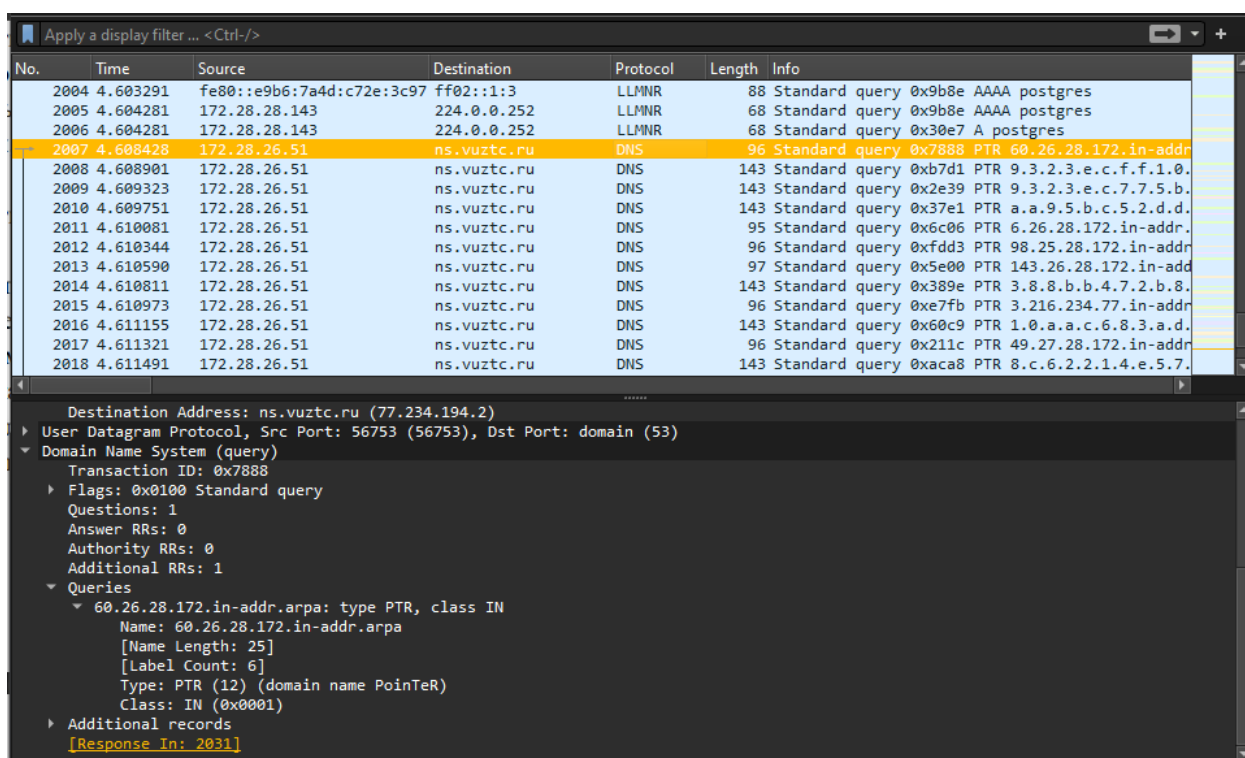


Рис. 1.11: Перехват трафика DNS в Wireshark без фильтрации

```
Nushell
~> netsh interface ip delete arpccache; arp -a
OK.

Интерфейс: 192.168.192.1 --- 0x8
  адрес в Интернете    Физический адрес    Тип
  224.0.0.22           01-00-5e-00-00-16    статический

Интерфейс: 172.28.26.51 --- 0x12
  адрес в Интернете    Физический адрес    Тип
  224.0.0.22           01-00-5e-00-00-16    статический

Интерфейс: 172.22.0.1 --- 0x15
  адрес в Интернете    Физический адрес    Тип
  224.0.0.22           01-00-5e-00-00-16    статический

Интерфейс: 172.18.112.1 --- 0x18
  адрес в Интернете    Физический адрес    Тип
  224.0.0.22           01-00-5e-00-00-16    статический

Интерфейс: 192.168.245.1 --- 0x1b
  адрес в Интернете    Физический адрес    Тип
  224.0.0.22           01-00-5e-00-00-16    статический

Интерфейс: 192.168.96.1 --- 0x24
  адрес в Интернете    Физический адрес    Тип
  224.0.0.22           01-00-5e-00-00-16    статический

Интерфейс: 172.21.128.1 --- 0x32
  адрес в Интернете    Физический адрес    Тип
  224.0.0.22           01-00-5e-00-00-16    статический

Интерфейс: 172.18.176.1 --- 0x48
  адрес в Интернете    Физический адрес    Тип
  224.0.0.22           01-00-5e-00-00-16    статический
~>
```

Рис. 1.12: Результат очистки ARP-таблицы

```
Nushell
~> nslookup tinkoff.ru
20.05.2024 12:26:04
тхЕтхЕ: UnKnown
Address: 172.28.16.1

Не заслуживающий доверия ответ:
тх : tinkoff.ru
Address: 178.248.236.218

~> nslookup -type=NS tinkoff.ru
тхЕтхЕ: UnKnown
Address: 172.28.16.1

Не заслуживающий доверия ответ:
tinkoff.ru      nameserver = ns2.tinkoff.ru
tinkoff.ru      nameserver = ns8-l2.nic.ru
tinkoff.ru      nameserver = ns4-l2.nic.ru
tinkoff.ru      nameserver = ns1.tinkoff.ru

tinkoff.ru      nameserver = ns8-l2.nic.ru
tinkoff.ru      nameserver = ns4-l2.nic.ru
tinkoff.ru      nameserver = ns1.tinkoff.ru
tinkoff.ru      nameserver = ns2.tinkoff.ru
ns2.tinkoff.ru  internet address = 185.169.154.98
ns8-l2.nic.ru   internet address = 91.217.21.20
ns4-l2.nic.ru   internet address = 91.217.20.20
ns1.tinkoff.ru  internet address = 178.248.239.11
ns8-l2.nic.ru   internet address = 91.217.21.20
ns4-l2.nic.ru   internet address = 91.217.20.20
ns1.tinkoff.ru  internet address = 178.248.239.11
ns2.tinkoff.ru  internet address = 185.169.154.98
~> |
20.05.2024 12:26:48
```

Рис. 1.13: Результат выполнения Nslookup.

```
Nushell
~> ipconfig /release

Настройка протокола IP для Windows

Невозможно выполнять операции над Ethernet, пока отключена сеть.
Невозможно выполнять операции над OpenVPN Connect DCO Adapter, пока отключена сеть.
Невозможно выполнять операции над Подключение по локальной сети* 1, пока отключена сеть.
Невозможно выполнять операции над Сетевое подключение Bluetooth, пока отключена сеть.

Неизвестный адаптер Подключение по локальной сети:

    Состояние среды. . . . . : Среда передачи недоступна.
    DNS-суффикс подключения . . . . . :

Адаптер Ethernet Ethernet:

    Состояние среды. . . . . : Среда передачи недоступна.
    DNS-суффикс подключения . . . . . :

Адаптер Ethernet vEthernet (Default Switch):

    DNS-суффикс подключения . . . . . :
    Локальный IPv6-адрес канала . . . : fe80::6fee:809d:9363:f537%21
    IPv4-адрес. . . . . : 172.22.0.1
    Маска подсети . . . . . : 255.255.240.0
    Основной шлюз. . . . . :

Адаптер Ethernet vEthernet (Беспроводная се):

    DNS-суффикс подключения . . . . . :
    Локальный IPv6-адрес канала . . . : fe80::8ae6:39b7:d456:7f73%50
    IPv4-адрес. . . . . : 172.21.128.1
    Маска подсети . . . . . : 255.255.240.0
    Основной шлюз. . . . . :

Неизвестный адаптер OpenVPN Connect DCO Adapter:

    Состояние среды. . . . . : Среда передачи недоступна.
    DNS-суффикс подключения . . . . . :

Адаптер беспроводной локальной сети Подключение по локальной сети* 1:

    Состояние среды. . . . . : Среда передачи недоступна.
    DNS-суффикс подключения . . . . . :
```

Рис. 1.14: Результат сброса IP

```
Nushell
~> ipconfig /renew 20.05.2024 12:35:44

Настройка протокола IP для Windows

Невозможно выполнять операции над Подключение по локальной сети, пока отключена сеть.
Невозможно выполнять операции над Ethernet, пока отключена сеть.
Невозможно выполнять операции над OpenVPN Connect DCO Adapter, пока отключена сеть.
Невозможно выполнять операции над Подключение по локальной сети* 1, пока отключена сеть.
Невозможно выполнять операции над Подключение по локальной сети* 2, пока отключена сеть.
Невозможно выполнять операции над Сетевое подключение Bluetooth, пока отключена сеть.

Неизвестный адаптер Подключение по локальной сети:

    Состояние среды. . . . . : Среда передачи недоступна.
    DNS-суффикс подключения . . . . . :

Адаптер Ethernet Ethernet:

    Состояние среды. . . . . : Среда передачи недоступна.
    DNS-суффикс подключения . . . . . :

Адаптер Ethernet vEthernet (VMware Network ) 2:

    DNS-суффикс подключения . . . . . :
    Локальный IPv6-адрес канала . . . : fe80::d105:9130:b742:84a4%72
    IPv4-адрес. . . . . : 172.18.176.1
    Маска подсети . . . . . : 255.255.240.0
    Основной шлюз. . . . . :

Адаптер Ethernet vEthernet (Беспроводная се):

    DNS-суффикс подключения . . . . . :
    Локальный IPv6-адрес канала . . . : fe80::8ae6:39b7:d456:7f73%50
    IPv4-адрес. . . . . : 172.21.128.1
    Маска подсети . . . . . : 255.255.240.0
    Основной шлюз. . . . . :

Неизвестный адаптер OpenVPN Connect DCO Adapter:

    Состояние среды. . . . . : Среда передачи недоступна.
    DNS-суффикс подключения . . . . . :

Адаптер беспроводной локальной сети Подключение по локальной сети* 1:

    Состояние среды. . . . . : Среда передачи недоступна.
```

Рис. 1.15: Результат обновления IP

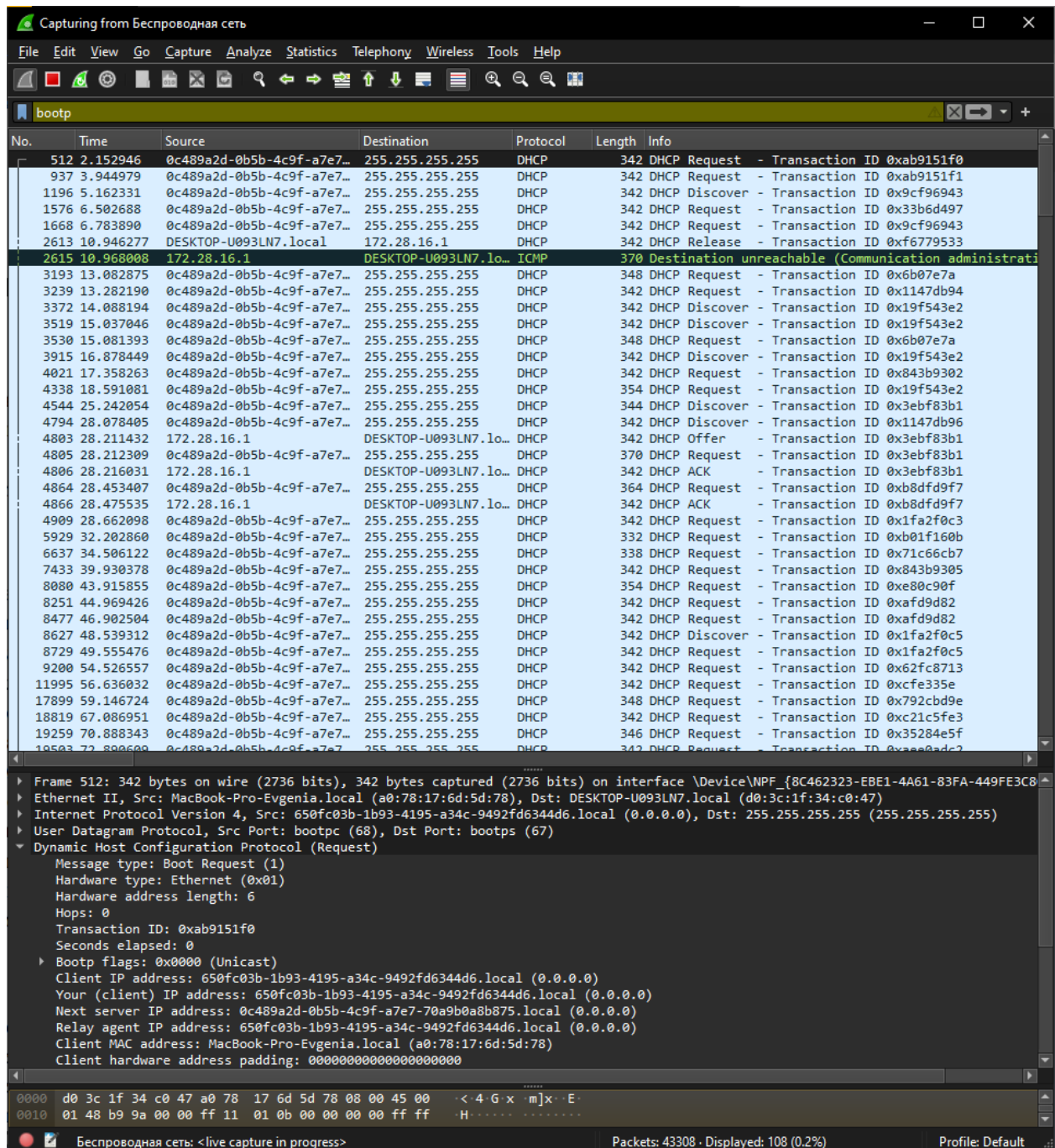


Рис. 1.16: Результат перехвата трафика в Wireshark при работе с DHCP