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: P33102

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

Санкт-Петербург, 2024 г.

Содержание

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 → tin → 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.

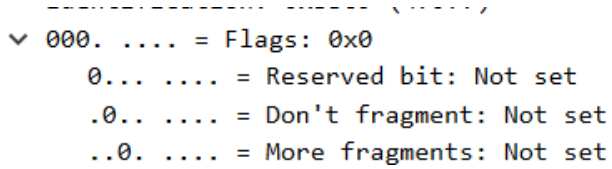


Рис. 1.3: ping MF

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.

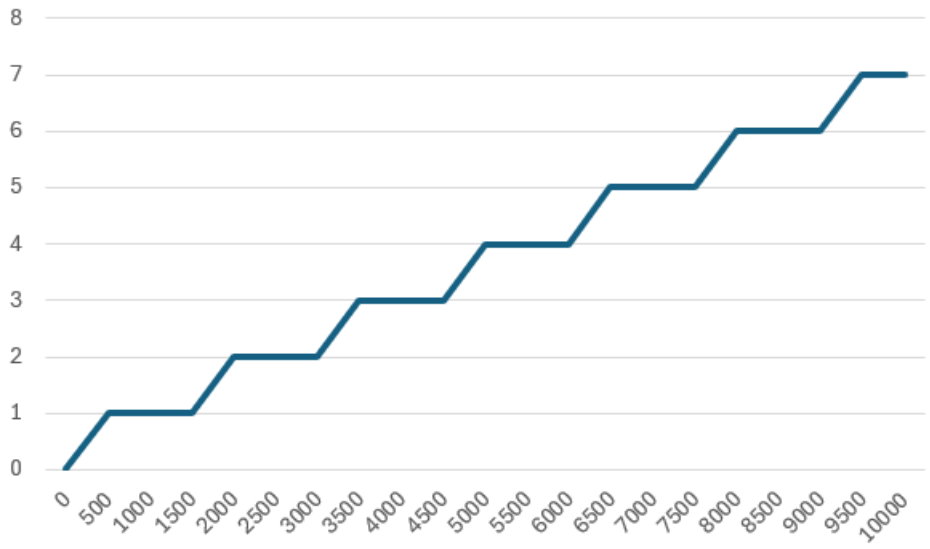


Рис. 1.4: Фрагментация ping-пакетов

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

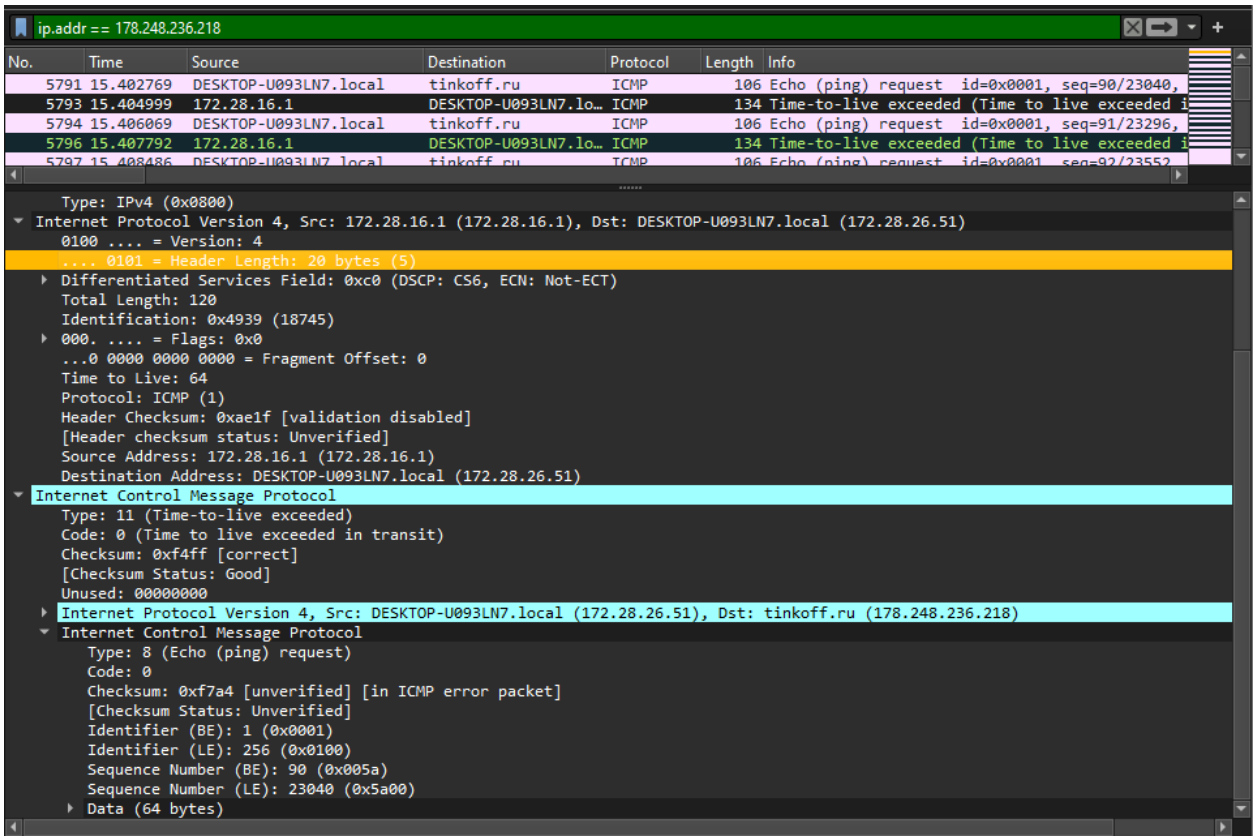


Рис. 1.7: Tracert data size

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 хопах.

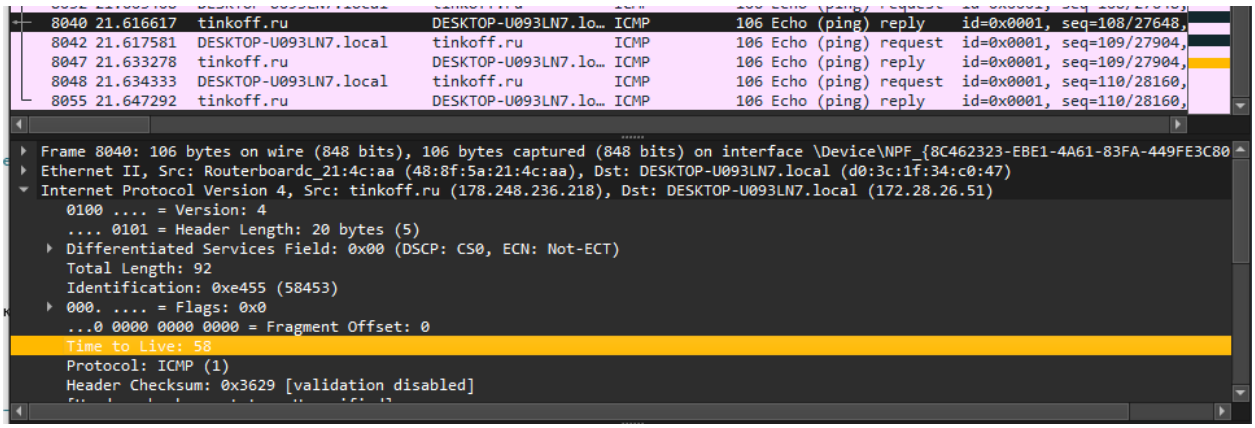


Рис. 1.8: Tracert TTL

3. Чем отличаются ICMP-пакеты, генерируемые утилитой tracert, от ICMP пакетов, генерируемых утилитой ping (см. предыдущее задание). — В tracert происходит увеличение TTL

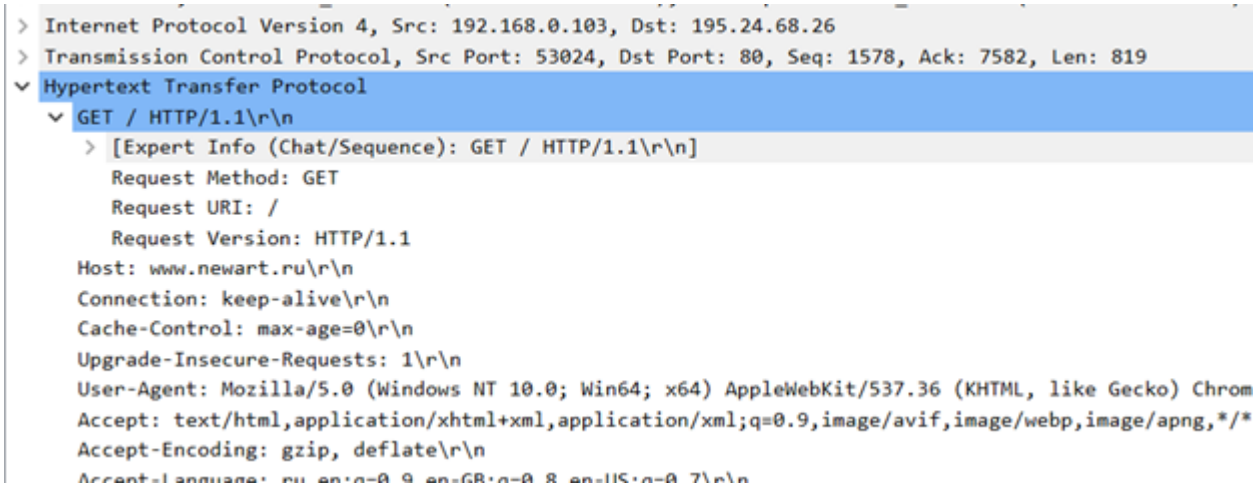
4. Чем отличаются полученные пакеты «ICMP reply» от «ICMP error» и зачем нужны оба этих типа ответов? — Различные значения в поле TYPE. ICMP reply: получение ответного сообщения; ICMP error: ошибка.

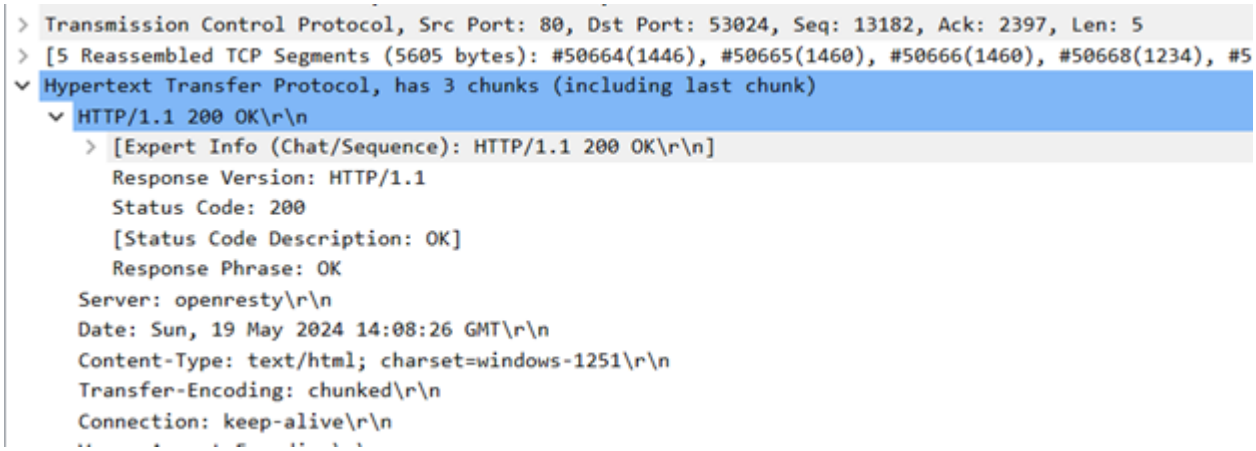
5. Что изменится в работе tracert, если убрать ключ «-d»? Какой дополнительный трафик при этом будет генерироваться? — ключ -d используется для того, чтобы предотвратить преобразование IP-адресов в имена хостов, если убрать ключ -d при использовании tracert, то утилита будет пытаться разрешать DNS-имена 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:





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.

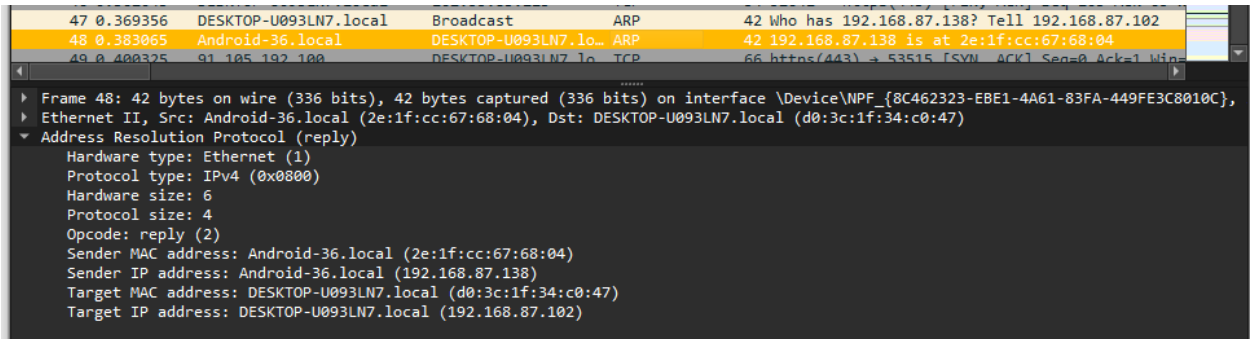
- *Обратный запрос* посылает IP и просит вернуть доменное имя.

3. В какой ситуации нужно выполнять независимые DNS-запросы для получения содержащихся на сайте изображений? — Когда картинки лежат на другом доменном имени.

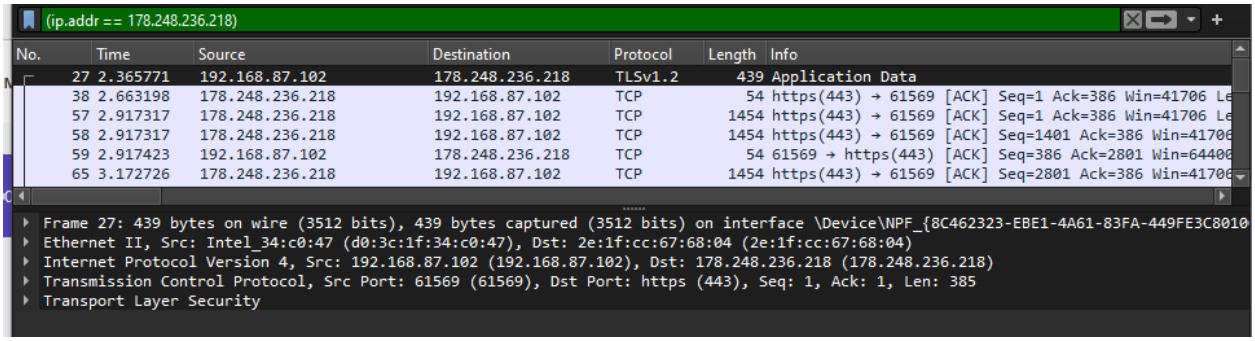
2. 5. Анализ ARP-трафика

Очистим таблицу ARP с помощью команды: `netsh interface ip delete arpc` 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 Wireshrk фильтр `ftp || ftp-data` .

1. Сколько байт данных содержится в пакете FTP-DATA? — Размер может быть любы, 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

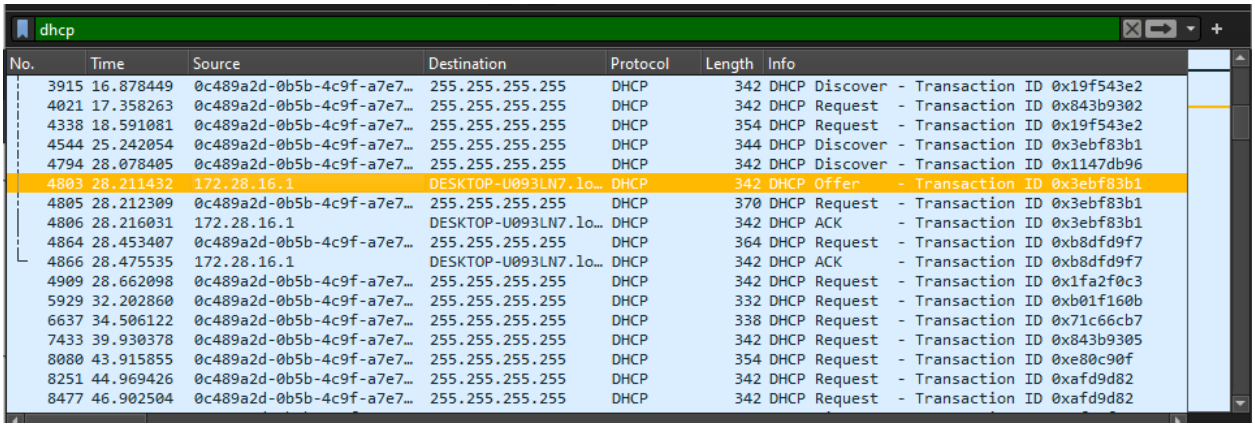
Установим в Wireshark фильтр `bootp`.

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.

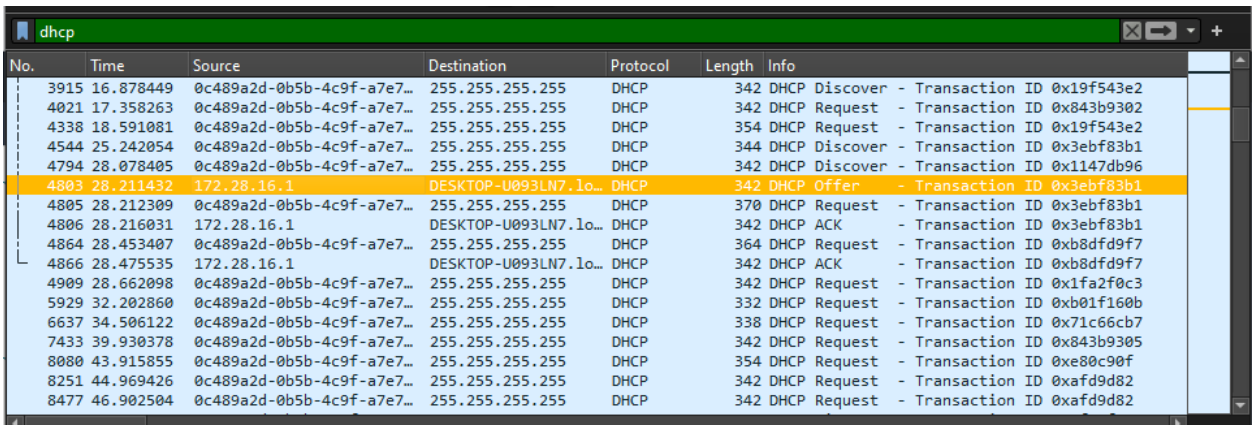
3. Каков IP-адрес DHCP-сервера? — 172.128.16.1



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

2. 9. Анализ Telegram-трафика

- Чем различаются пакета 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.



- 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 `tsl`, `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

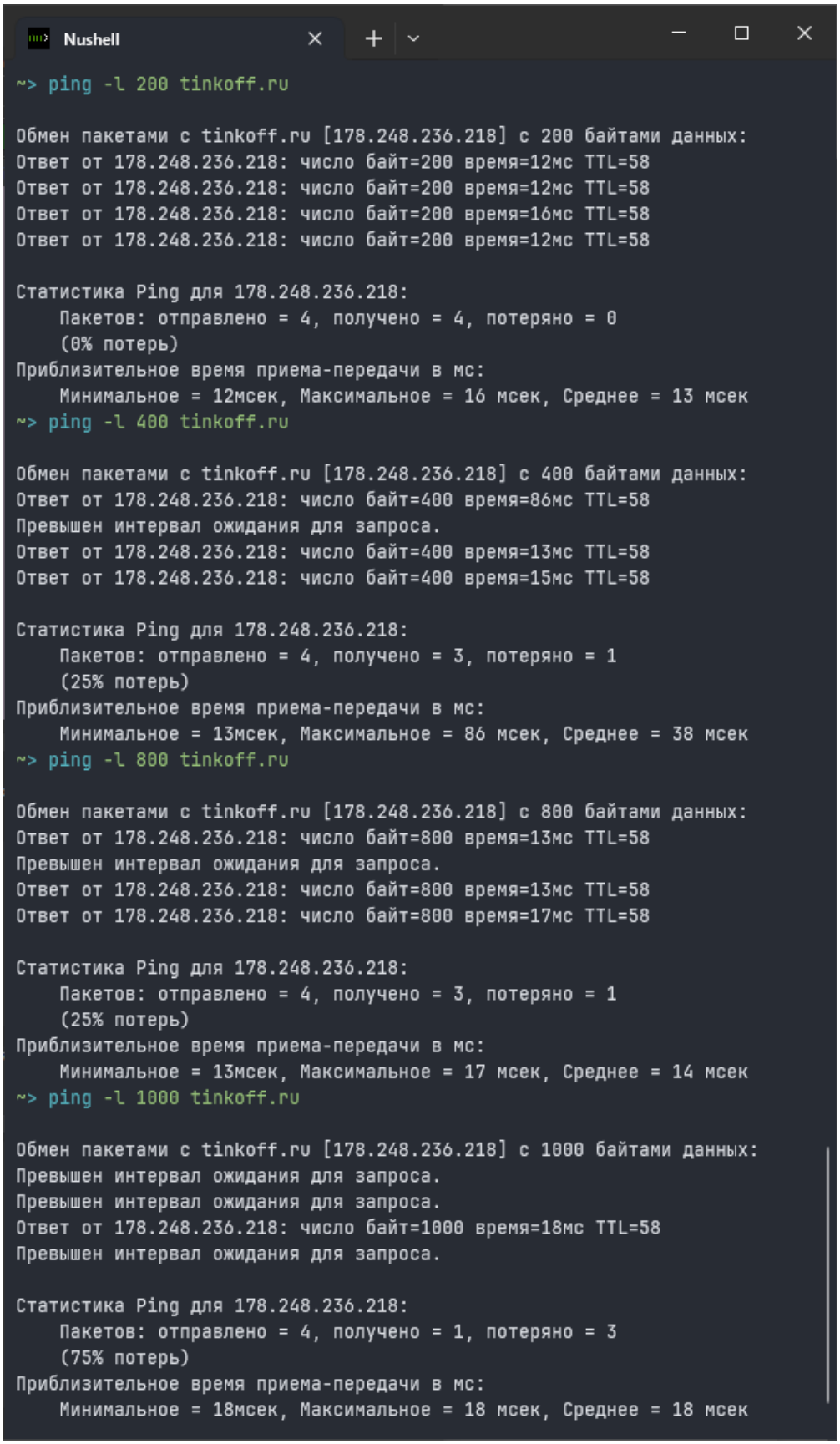


Рис. 1.1: Enter Caption

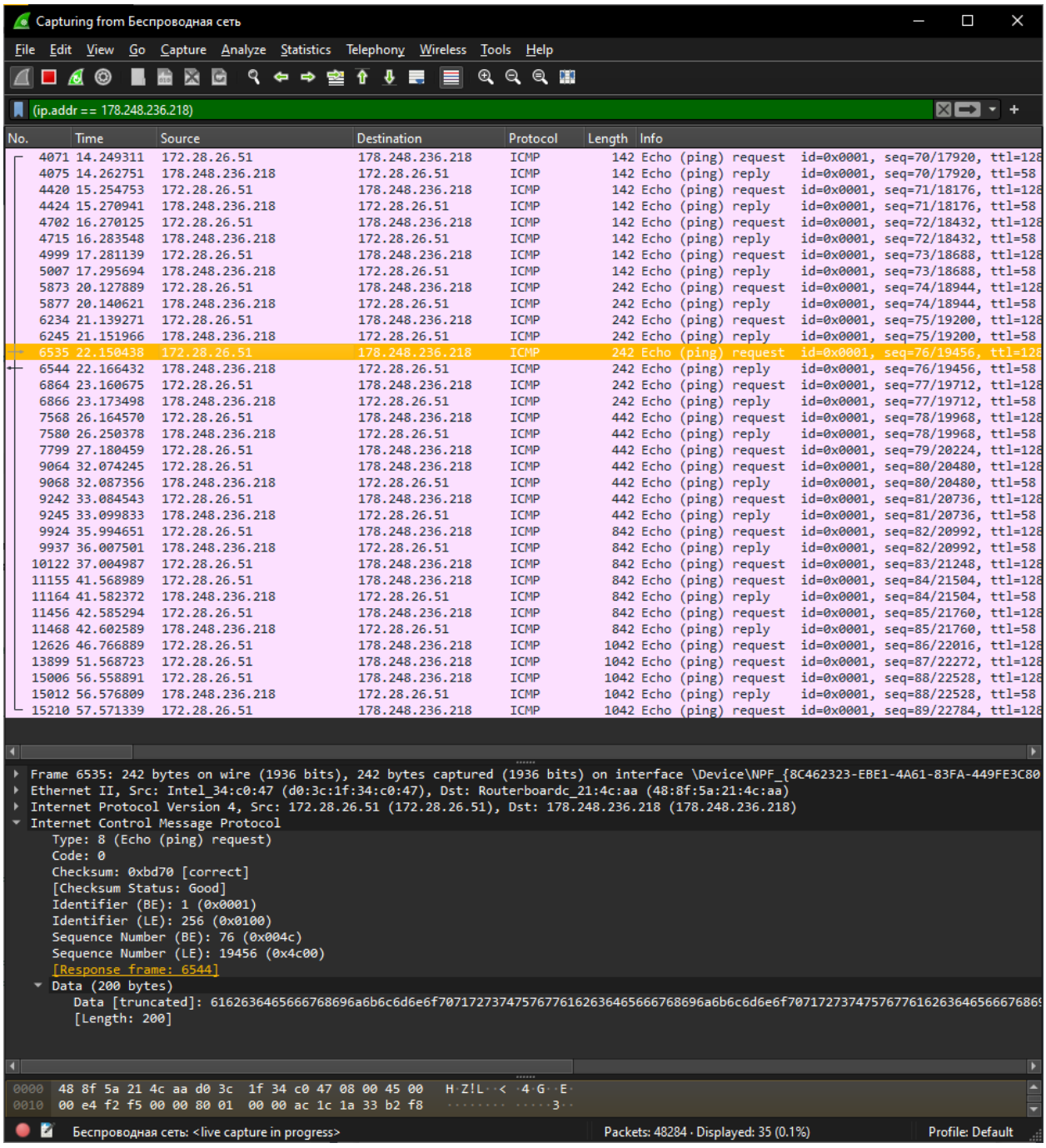


Рис. 1.2: Ping в Wireshark

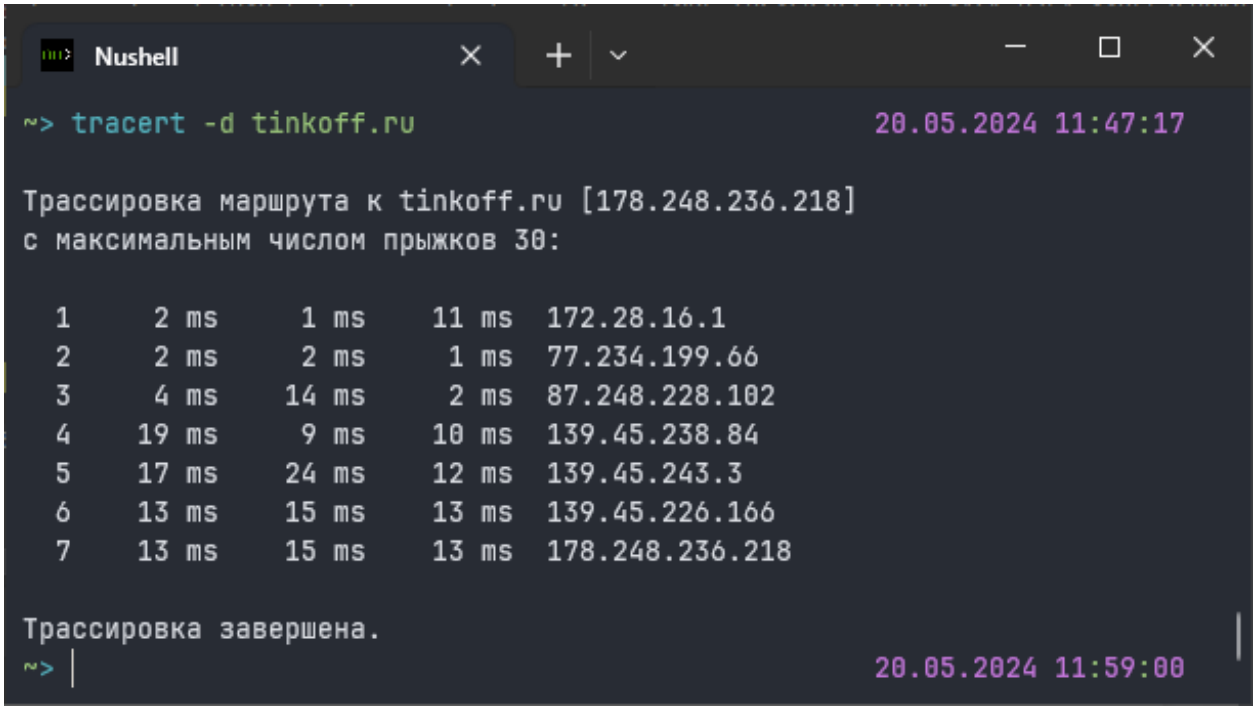


Рис. 1.5: Tracert в консоли

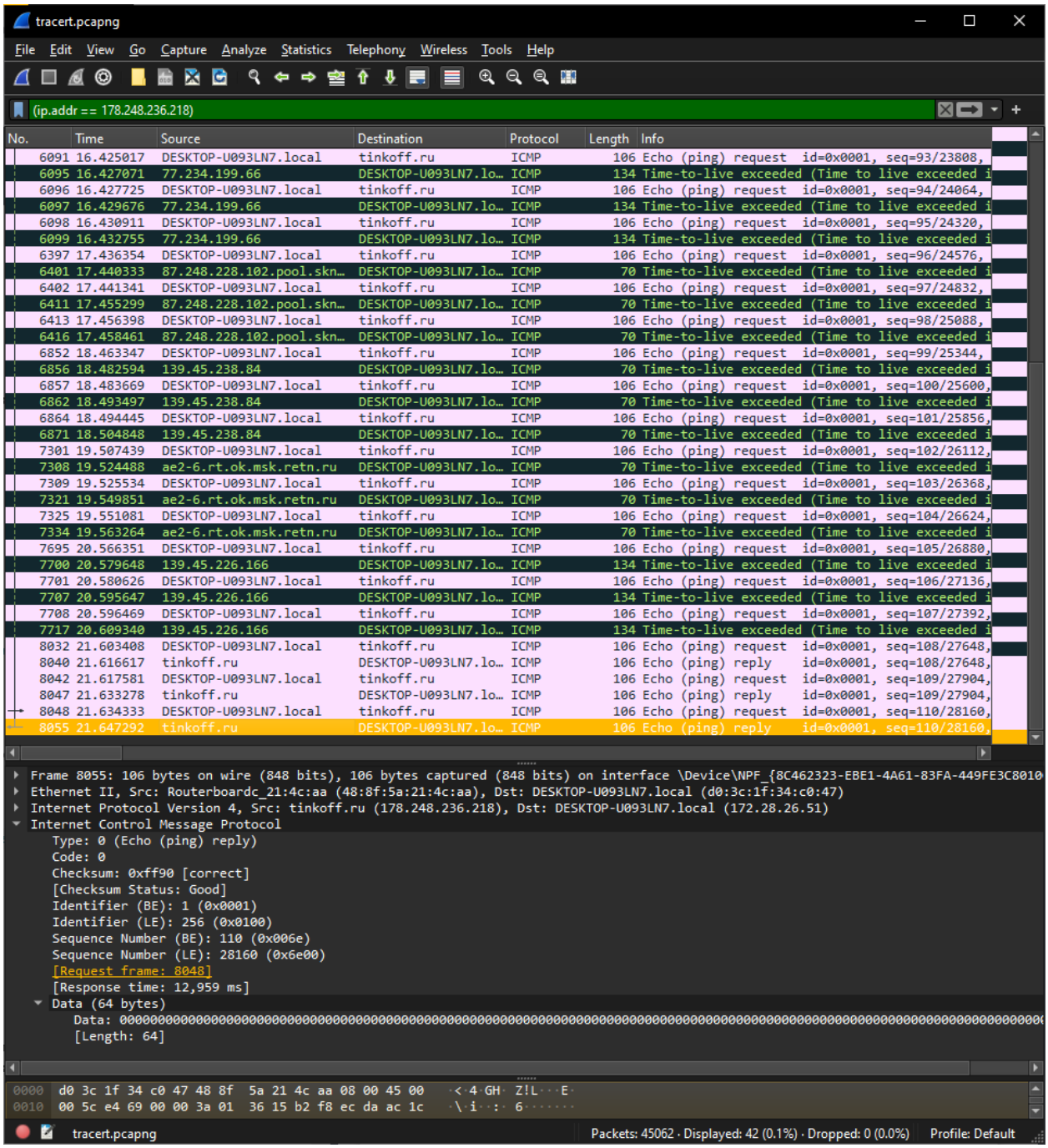


Рис. 1.6: Tracert в Wireshark

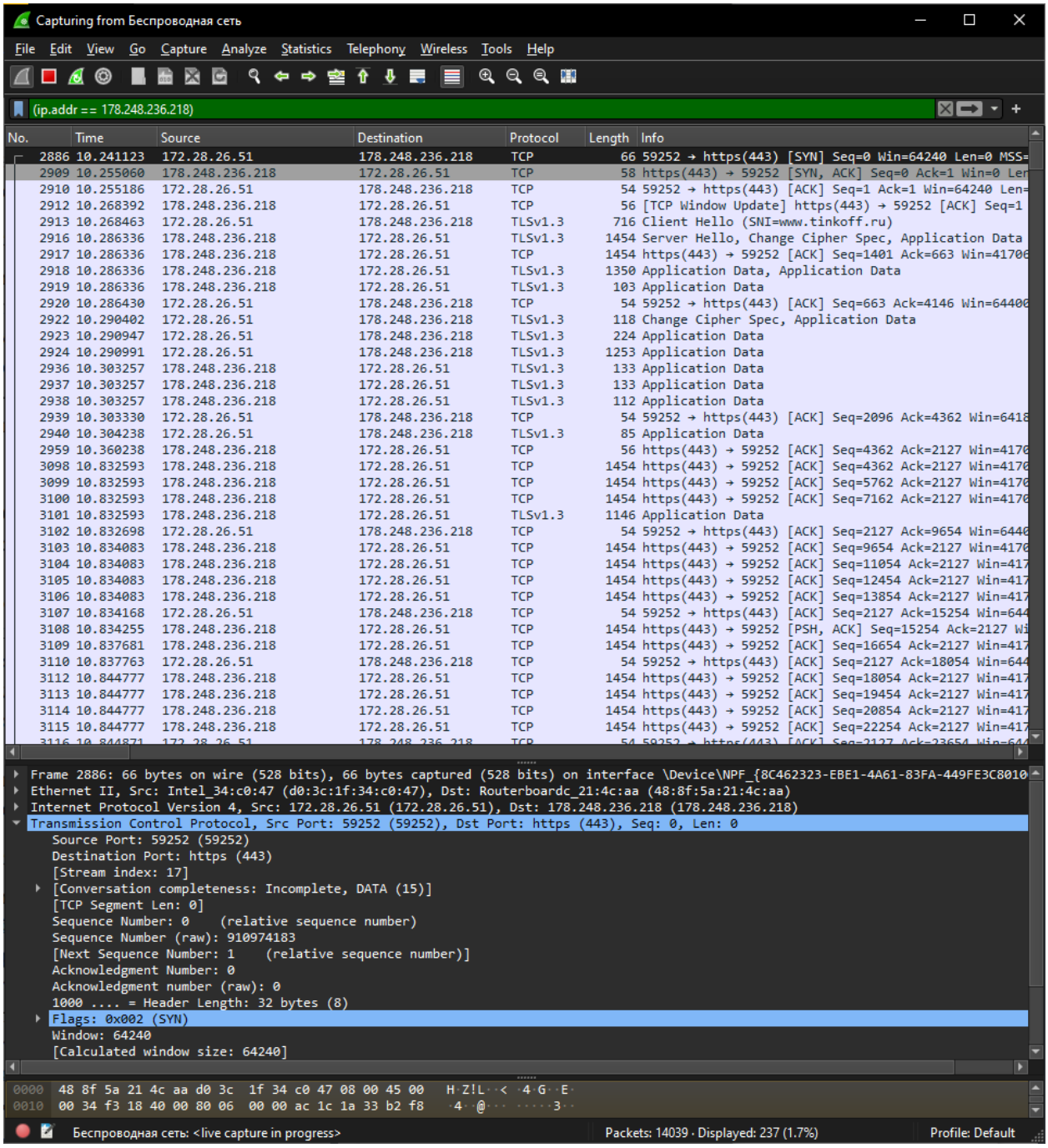


Рис. 1.9: Http в Wireshark

Рис. 1.10: Сброс настроек DNS

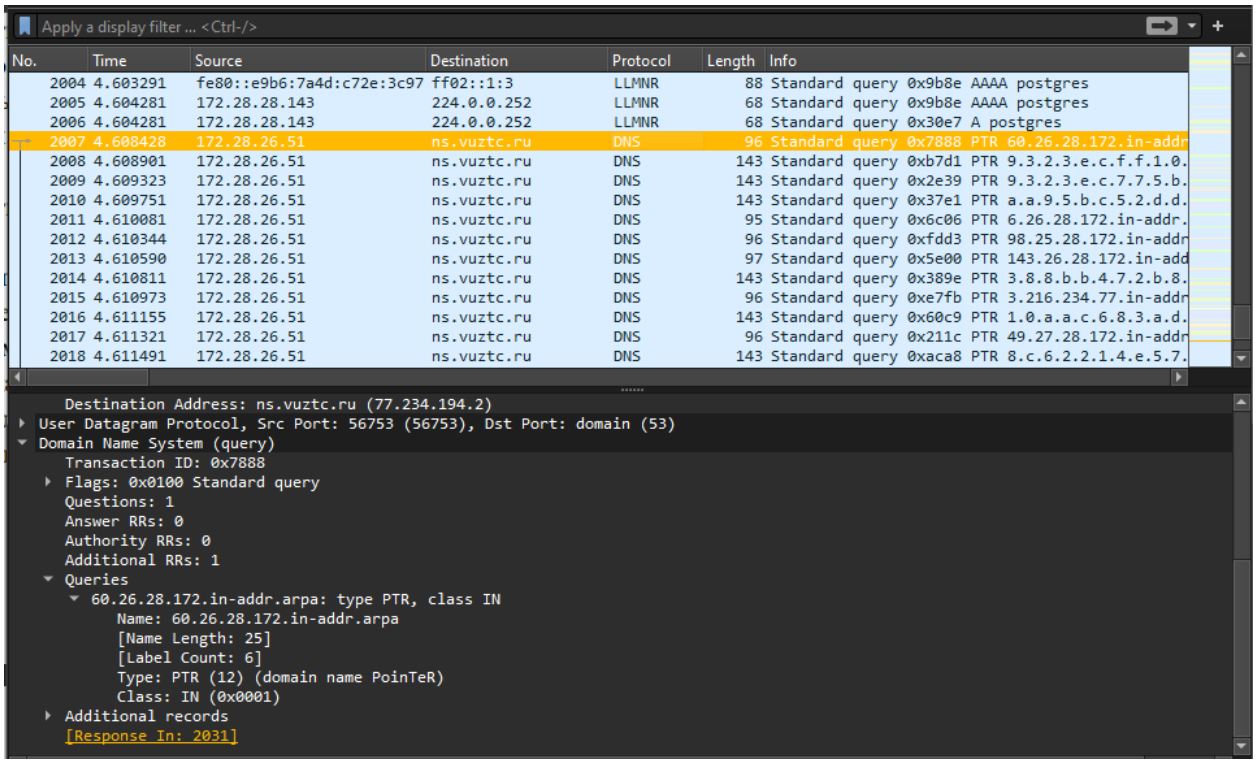


Рис. 1.11: Перехват трафика DNS в Wireshark без фильтрации

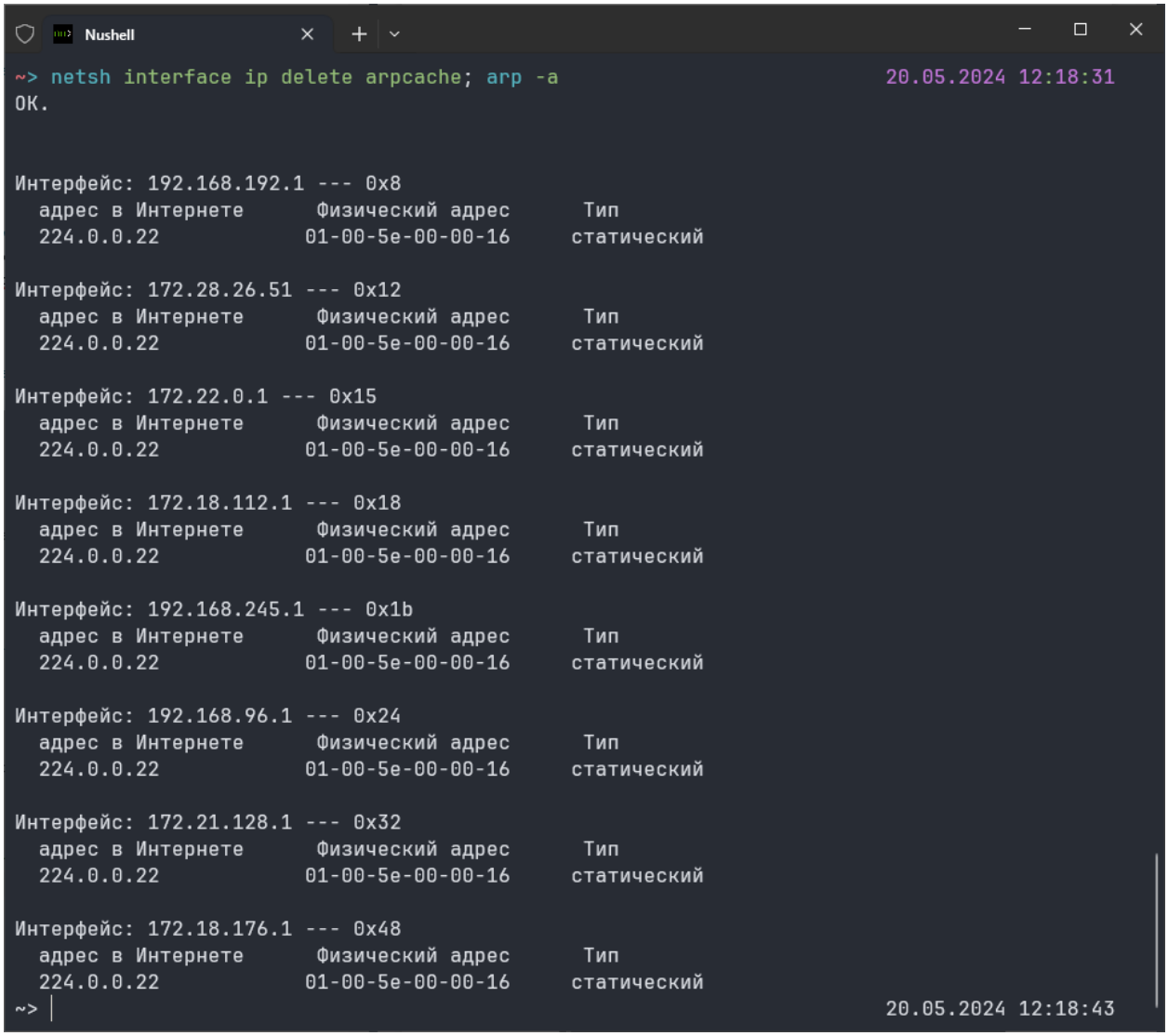


Рис. 1.12: Результат очистки ARP-таблицы

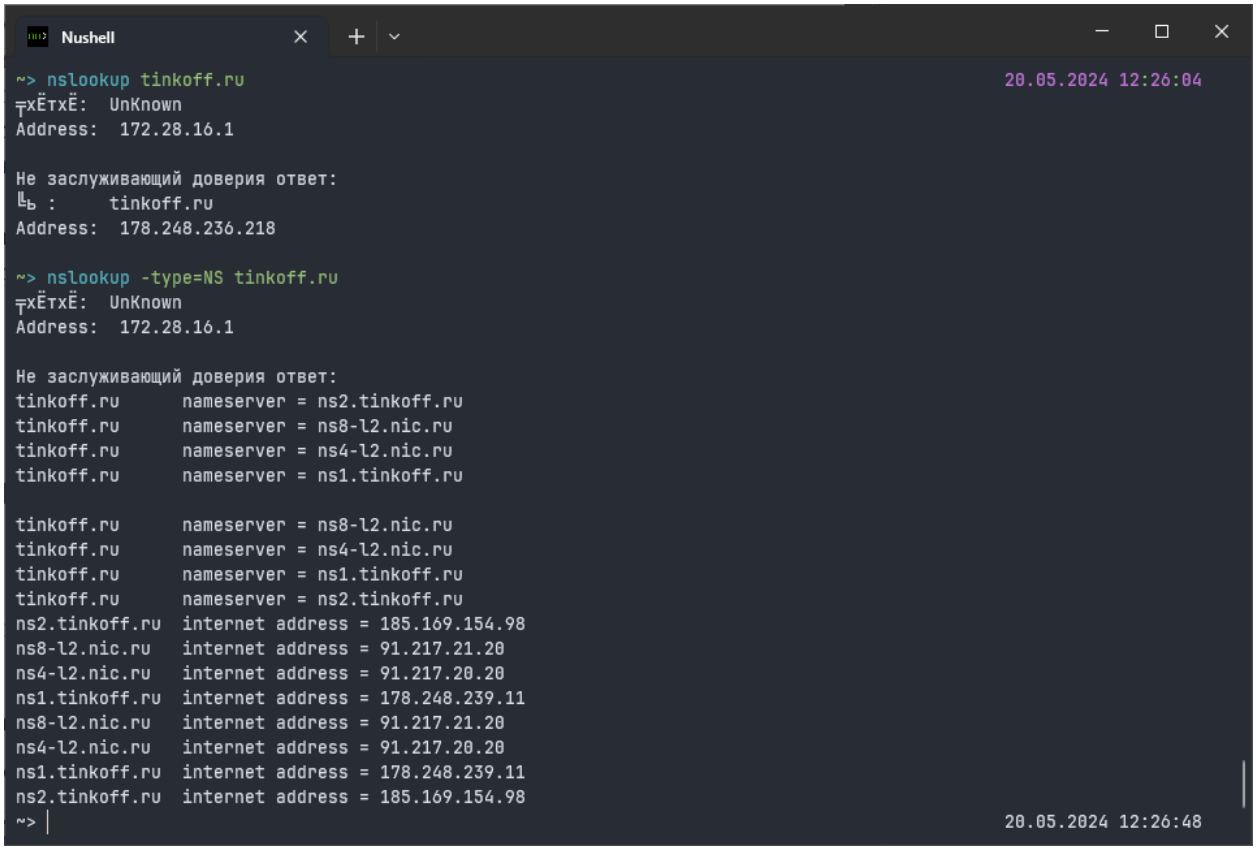


Рис. 1.13: Результат выполнения Nslookup.

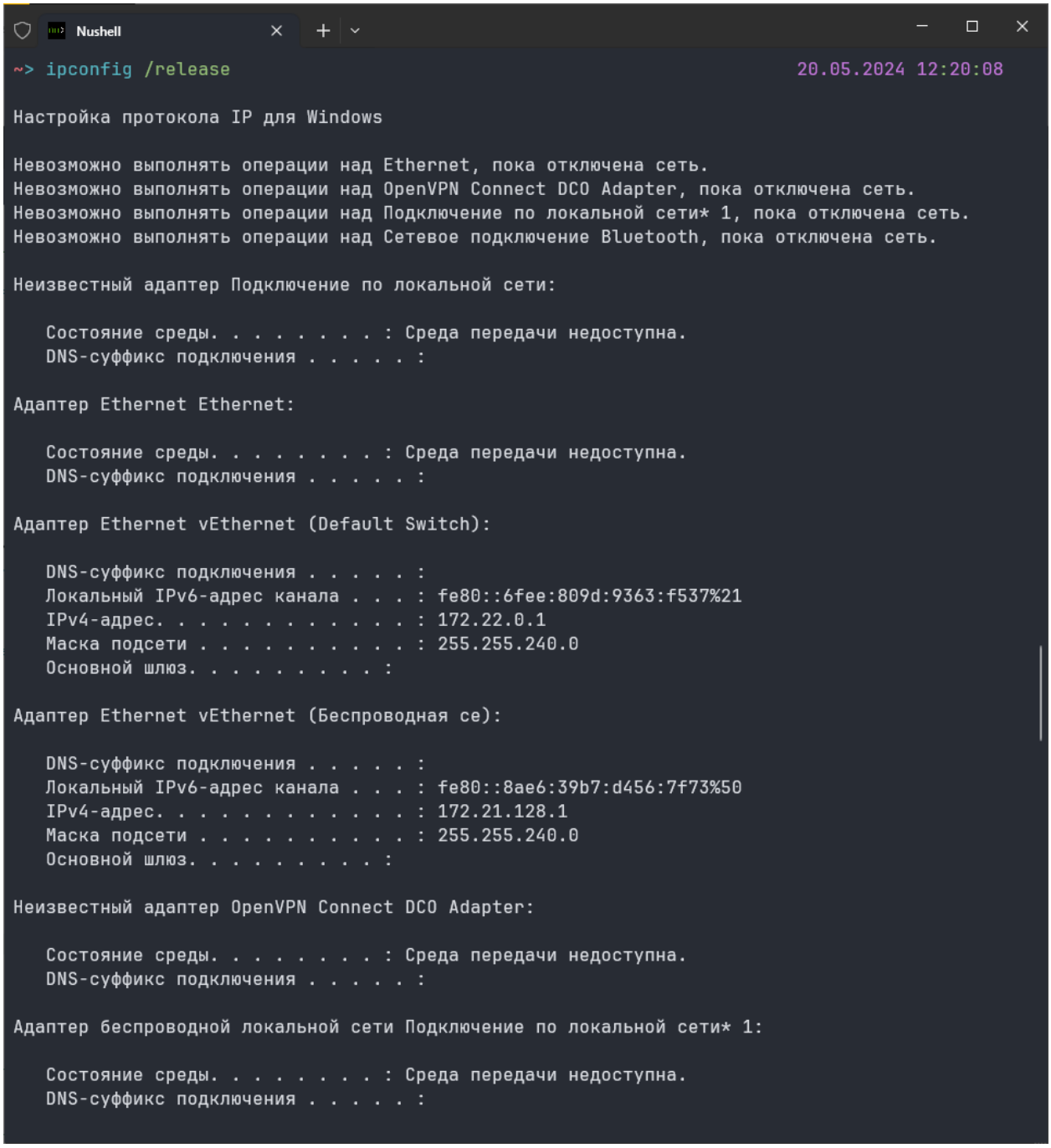


Рис. 1.14: Результат сброса IP

Рис. 1.15: Результат обновления IP

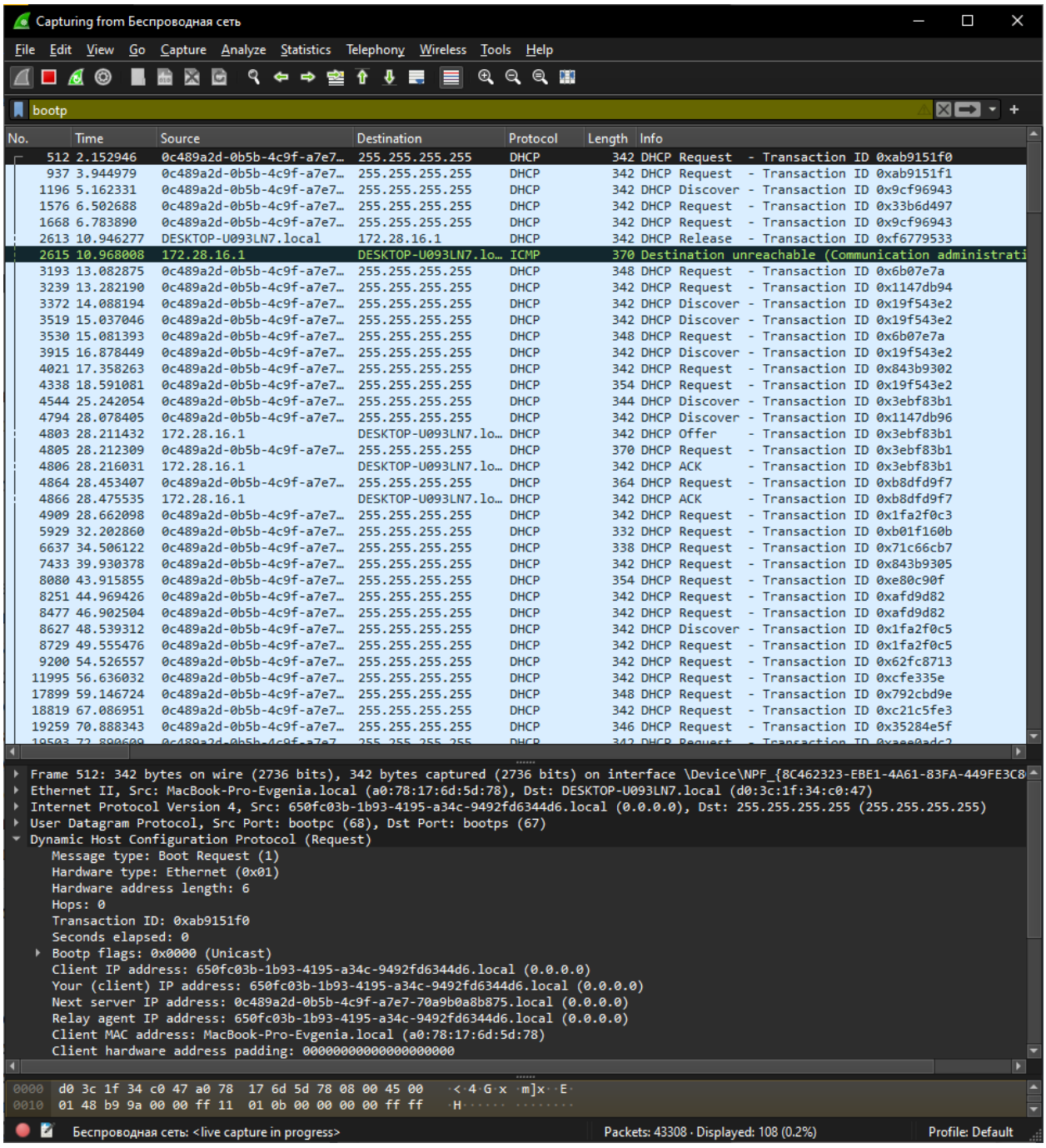


Рис. 1.16: Результат перехвата трафика в Wireshark при работе с DHCP