ГУАП

КАФЕДРА № 42

ОТЧЕТ   
ЗАЩИЩЕН С ОЦЕНКОЙ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

ПРЕПОДАВАТЕЛЬ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ассистент |  |  |  | Д.Д. Савельева |
| должность, уч. степень, звание |  | подпись, дата |  | инициалы, фамилия |

|  |
| --- |
| ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ №2 |
| АНАЛИЗ ТРАФИКА КОМПЬЮТЕРНЫХ СЕТЕЙ УТИЛИТОЙ WIRESHARK |
| по курсу: Инфокоммуникационные системы и сети |
|  |

РАБОТУ ВЫПОЛНИЛ

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| СТУДЕНТ ГР. № | 4329 |  |  |  | Д.С. Шаповалова |
|  |  |  | подпись, дата |  | инициалы, фамилия |

Санкт-Петербург 2025

# **1. Цель работы:**

Изучить структуру протокольных блоков данных, анализируя реальный трафик с помощью утилиты Wireshark.

# **2. Ход работы:**

*1. Анализ трафика утилиты ping*

Был отслежен и проанализирован трафик утилитой Wireshark, создаваемый утилитой ping, при запуске из командной строки. В качестве “размера\_пакета” были поочерёдно использованы значения от 100 до 10000.

Результат отслеживания представлен на рисунках 1.1-1.3:

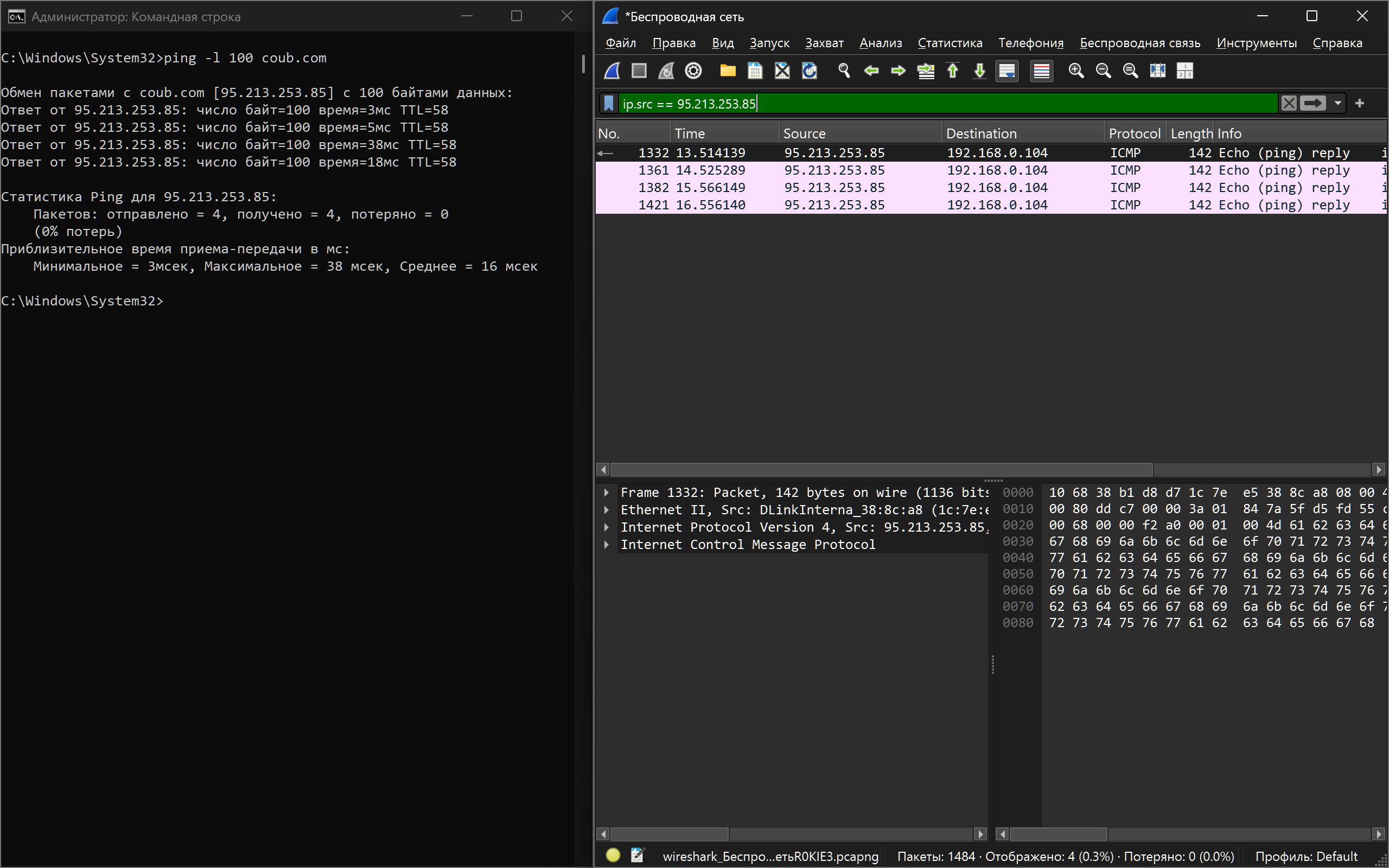


Рисунок 1.1 – ping -l 100 coub.com

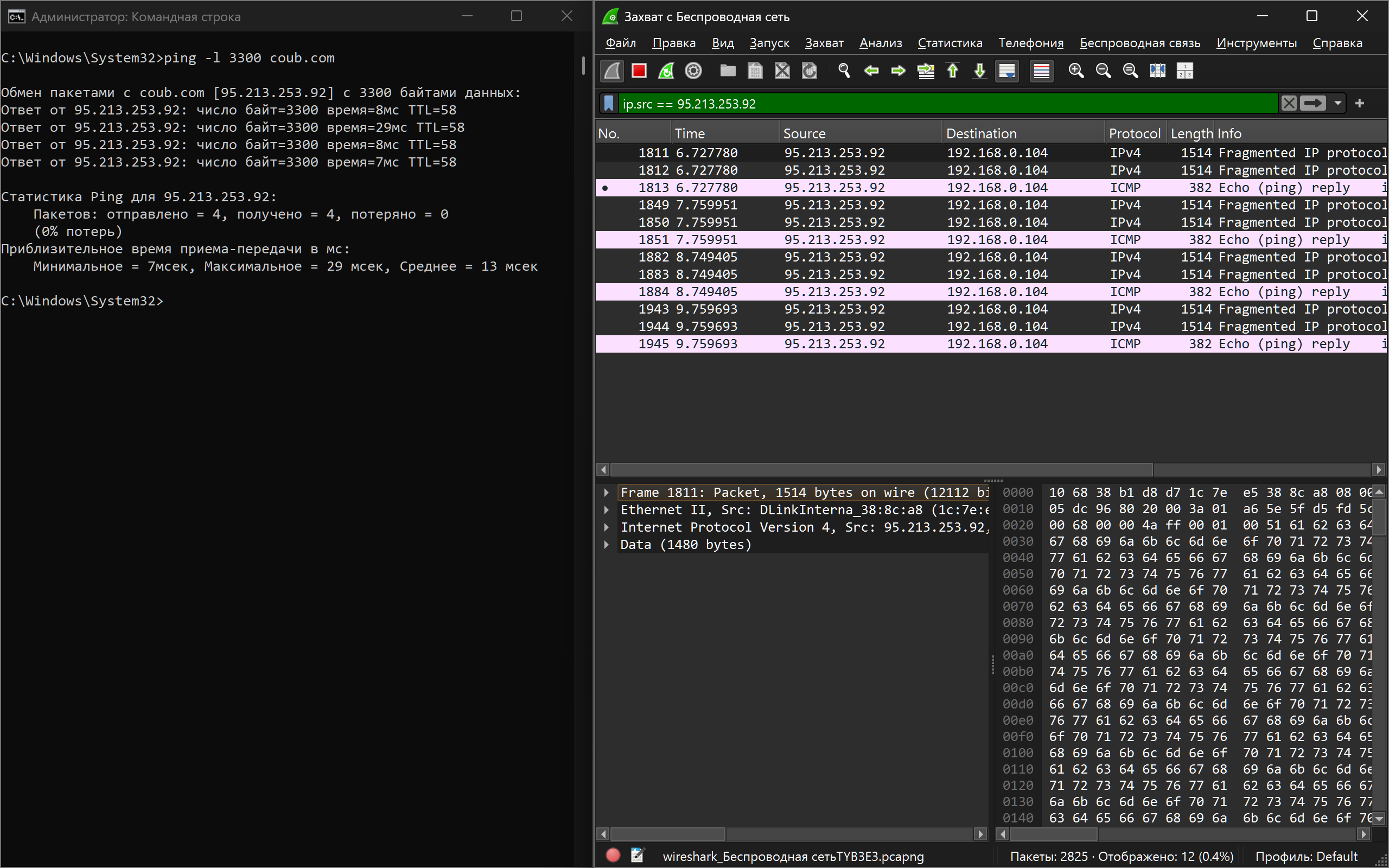


Рисунок 1.2.1 – ping -l 3300 coub.com

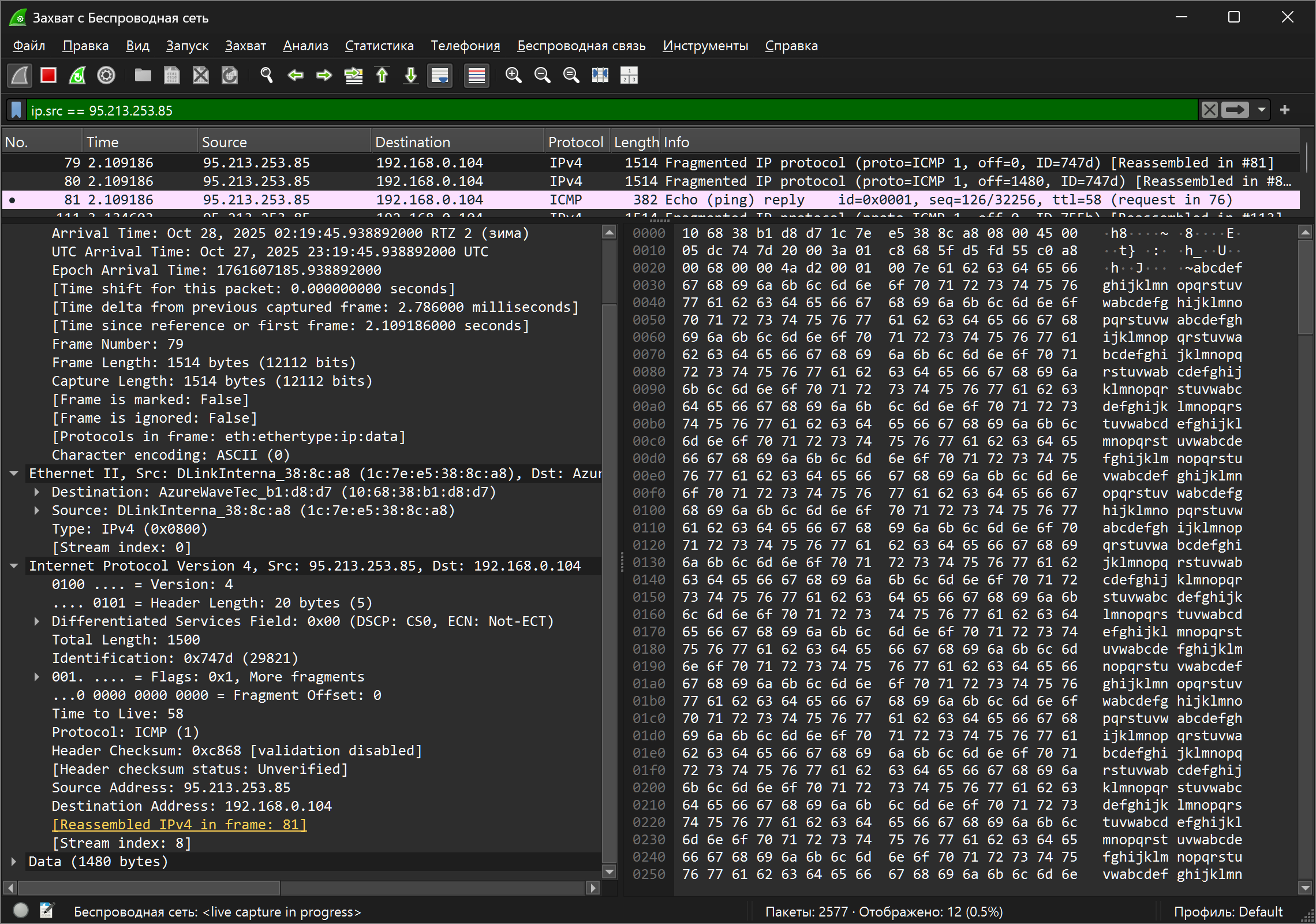


Рисунок 1.2.2 – ping -1 3300 coub.com, подробнее

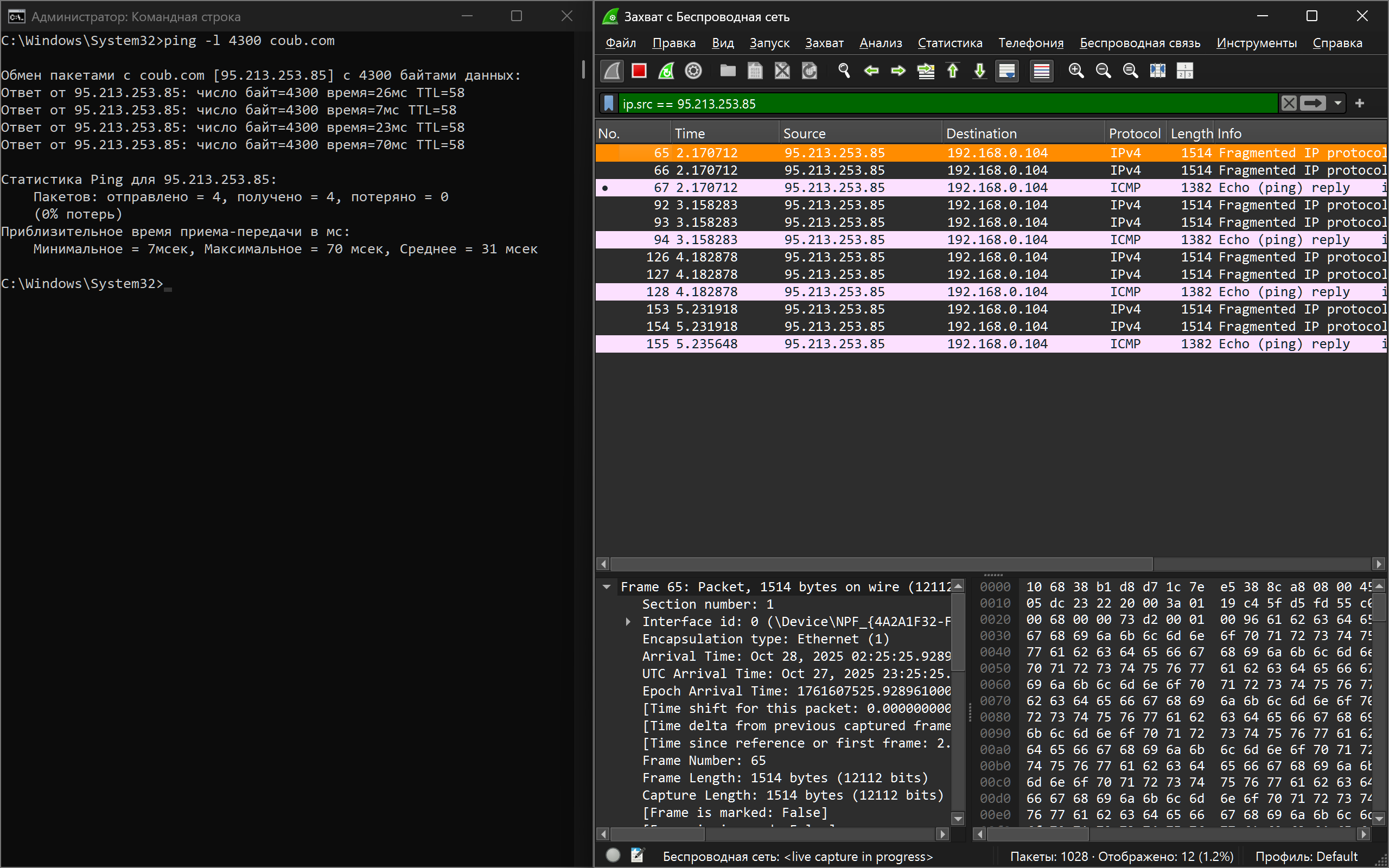


Рисунок 1.3.1 – ping -1 4000 coub.com

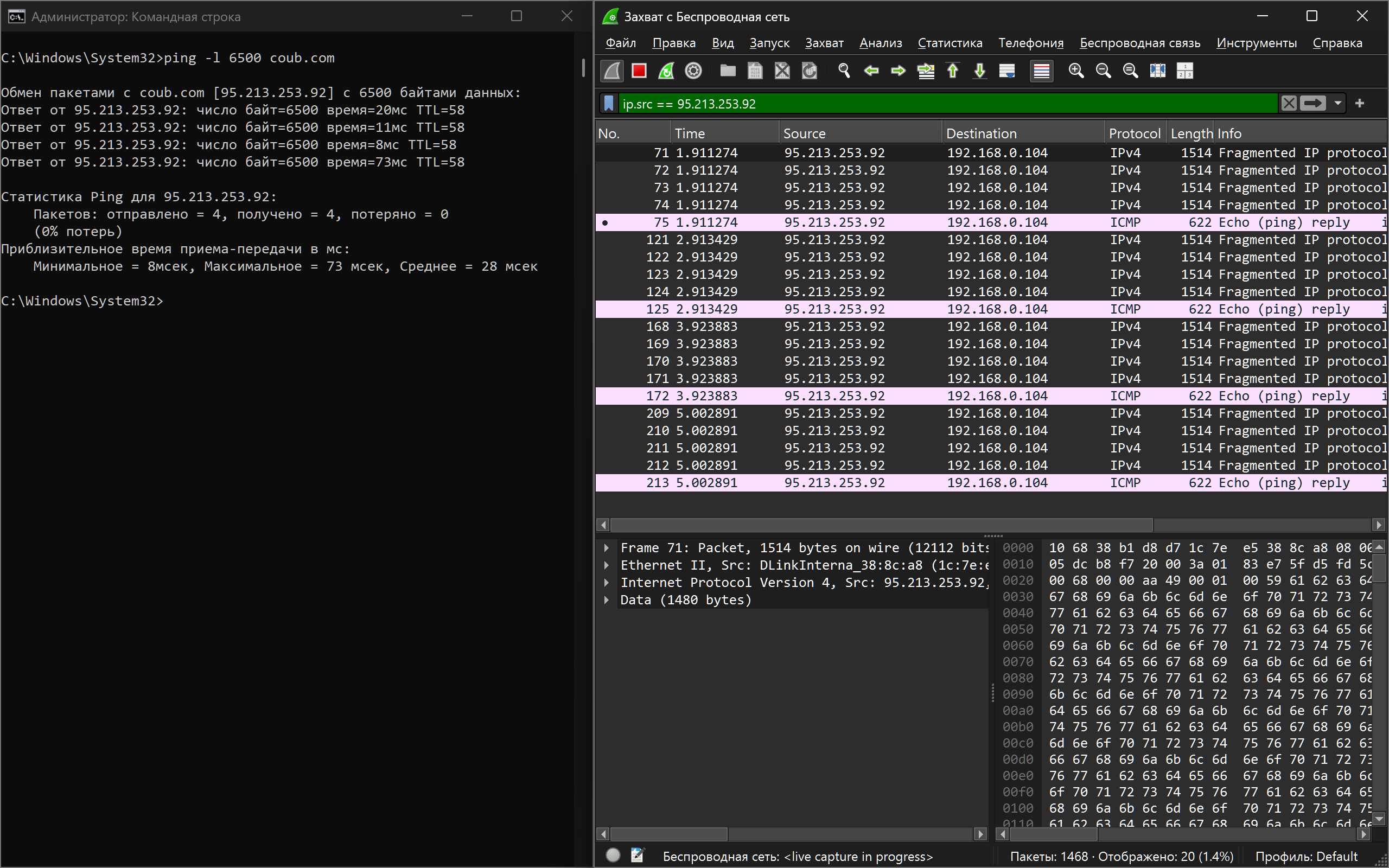


Рисунок 1.3.2 – ping -l 6500 coub.com

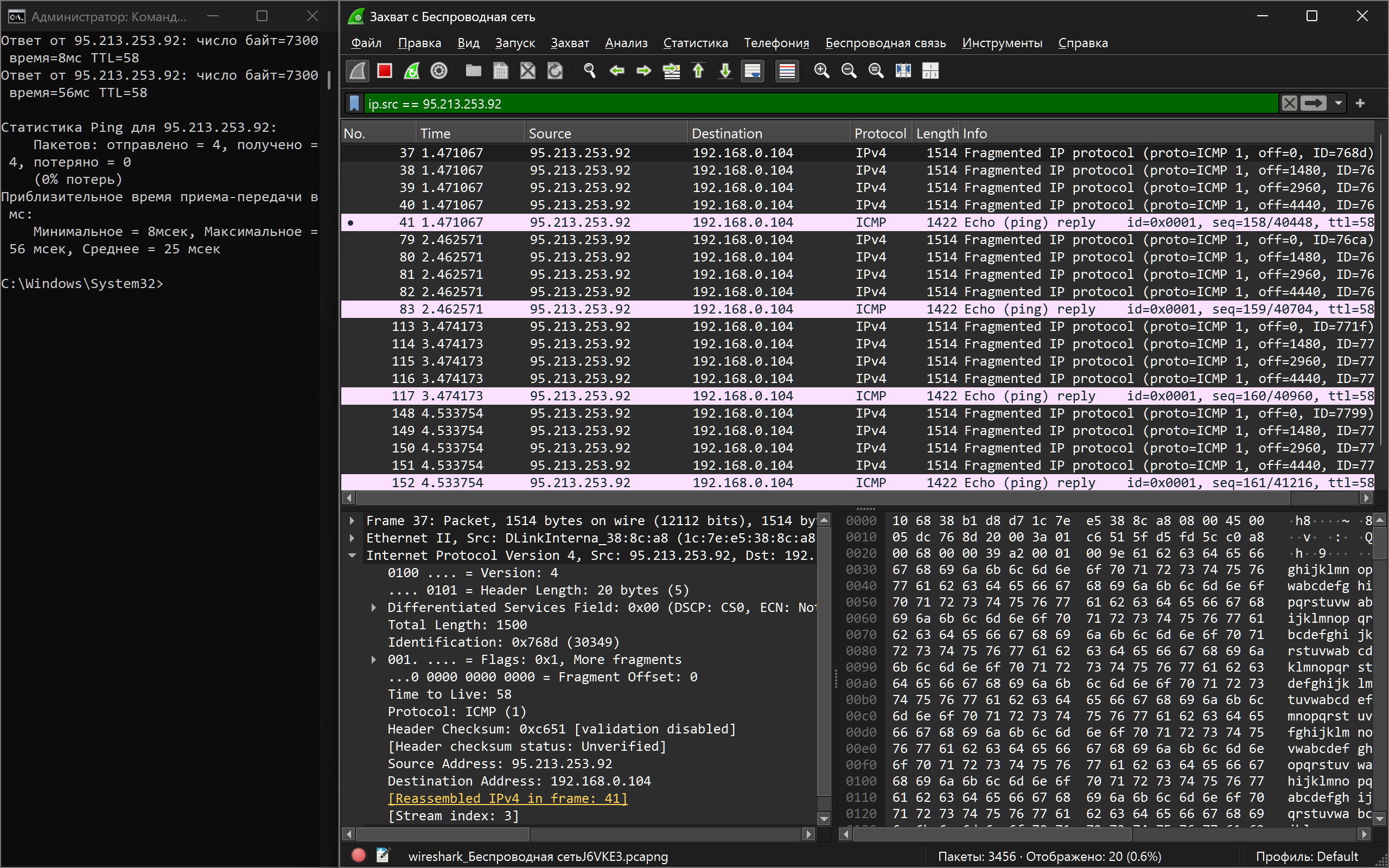


Рисунок 1.3.3 – ping -1 7300 coub.com

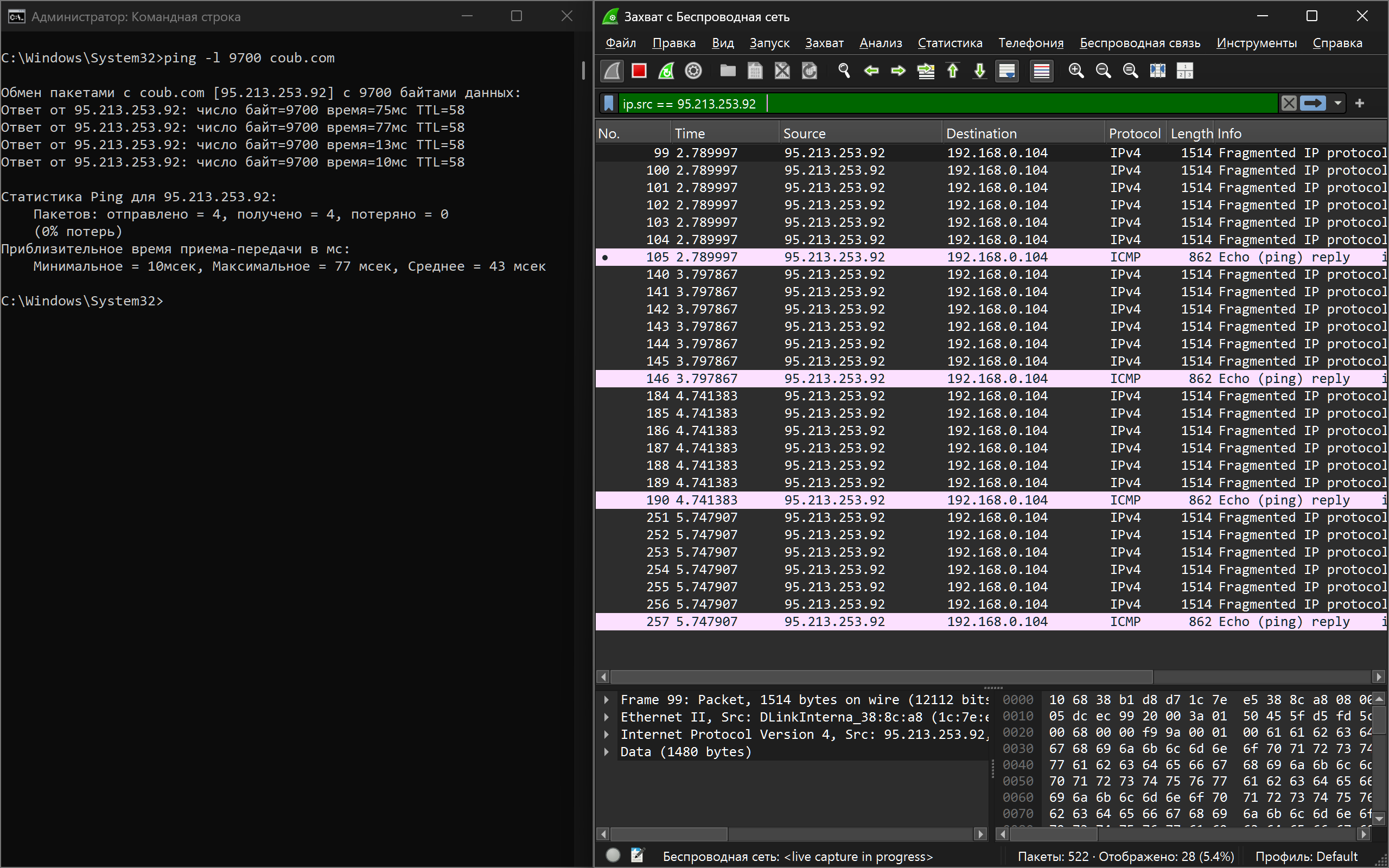


Рисунок 1.4 - ping -l 9700 coub.com

1. Имеет ли место фрагментация исходного пакета, какое поле на это указывает?

Фрагментация возникает, если размер ICMP-пакета (Echo Request/Reply) превышает максимально допустимое значение MTU для данного канала (обычно 1500 байт для Ethernet).

На фрагментацию указывает поле "Flags" (Флаги) в заголовке IP-пакета.

Если установлен флаг "More fragments", это означает, что данный пакет является промежуточным фрагментом, а не последним.

На скриншотах (Рисунок 1.1) для маленького пакета (100 байт) флаг "More fragments" всего не установлен, что говорит об отсутствии фрагментации.

На скриншотах для больших пакетов (Рисунок 1.2, 1.3, 1.4) можно увидеть несколько ICMP-пакетов с установленным флагом "More fragments" для всех, кроме последнего.

2. Какая информация указывает, является ли фрагмент пакета последним или промежуточным?

На наличие фрагментации указывают поля заголовка IP-пакета:

• Flags (бит MF — More Fragments)

• Fragment Offset (смещение фрагмента в общем пакете).

Если пакет не фрагментирован, MF=0 и Offset=0. Если наблюдается фрагментация, то хотя бы один пакет имеет MF=1 или ненулевой Offset.

3. Чему равно количество фрагментов при передаче ping-пакетов?

Количество фрагментов зависит от размера полезной нагрузки (-l), который вы указали, и от MTU (Maximum Transmission Unit) вашей сети. MTU — это максимальный размер кадра/cегмента, который может быть передан без фрагментации (обычно 1500 байт для Ethernet).

* Для 100: Размер пакета меньше MTU. Фрагментации нет, количество фрагментов = 1.
* Для 3300-4300: (Размер IP-заголовка + ICMP-заголовок + 3300 байт данных) > MTU. Требуется фрагментация. Количество фрагментов = 3.
* Для 6500-7300: Количество фрагментов больше – 5.
* Для 9700**:** Количество фрагментов ещё больше – 7.

4. Построить график, в котором на оси абсцисс находится размер\_пакета, а по оси ординат – количество фрагментов, на которое был разделён каждый ping-пакет.

Рисунок 1.5 – График зависимости числа фрагментов от размера пакета

5. Как изменить поле TTL с помощью утилиты ping?

Поле TTL (Time to Live) в заголовке IP-пакета можно изменить с помощью ключа -i в ОС Windows.

Синтаксис: ping -i <TTL> -l <размер> <адрес>

Пример: ping -i **50** -l 100 coub.com

6. Что содержится в поле данных ping-пакета?

Поле данных ICMP-пакета (Echo Request/Reply), генерируемого утилитой ping, по умолчанию содержит набор повторяющихся буквенно-цифровых символов. Обычно это последовательность из символов латинского алфавита (abcdefghijklmnopqrstuvwabc...).

Назначение этого заполнения — достичь запрошенного размера пакета (-l). Если установить -l 100, в данных будет 100 байт этого служебного заполнения. По шестнадцатеричному представлению пакета в Wireshark, можно увидеть данную повторяющуюся последовательность.

*Структура наблюдаемых PDU для ICMP Echo Request и Echo Reply*

Структура PDU ICMP Echo Reply:

Уровень 2 (Канальный): Ethernet-кадр с MAC-адресами источника (устройства на стороне сервера) и назначения (вашего компьютера).

Уровень 3 (Сетевой): IP-пакет, содержащий IP-адреса отправителя (сервер) и получателя (ваш компьютер). Поле TTL указывает оставшееся время жизни пакета.

Уровень 4 (Транспортный/Служебный): В IP-пакете размещено ICMP-сообщение типа Echo Reply. Его заголовок содержит значения: Type = 0 (Echo Reply), Code = 0, контрольную сумму. Идентификатор и порядковый номер совпадают с теми, что были в соответствующем Echo Request, что подтверждает принадлежность ответа именно этому запросу. Полезная нагрузка возвращается в неизменном виде.

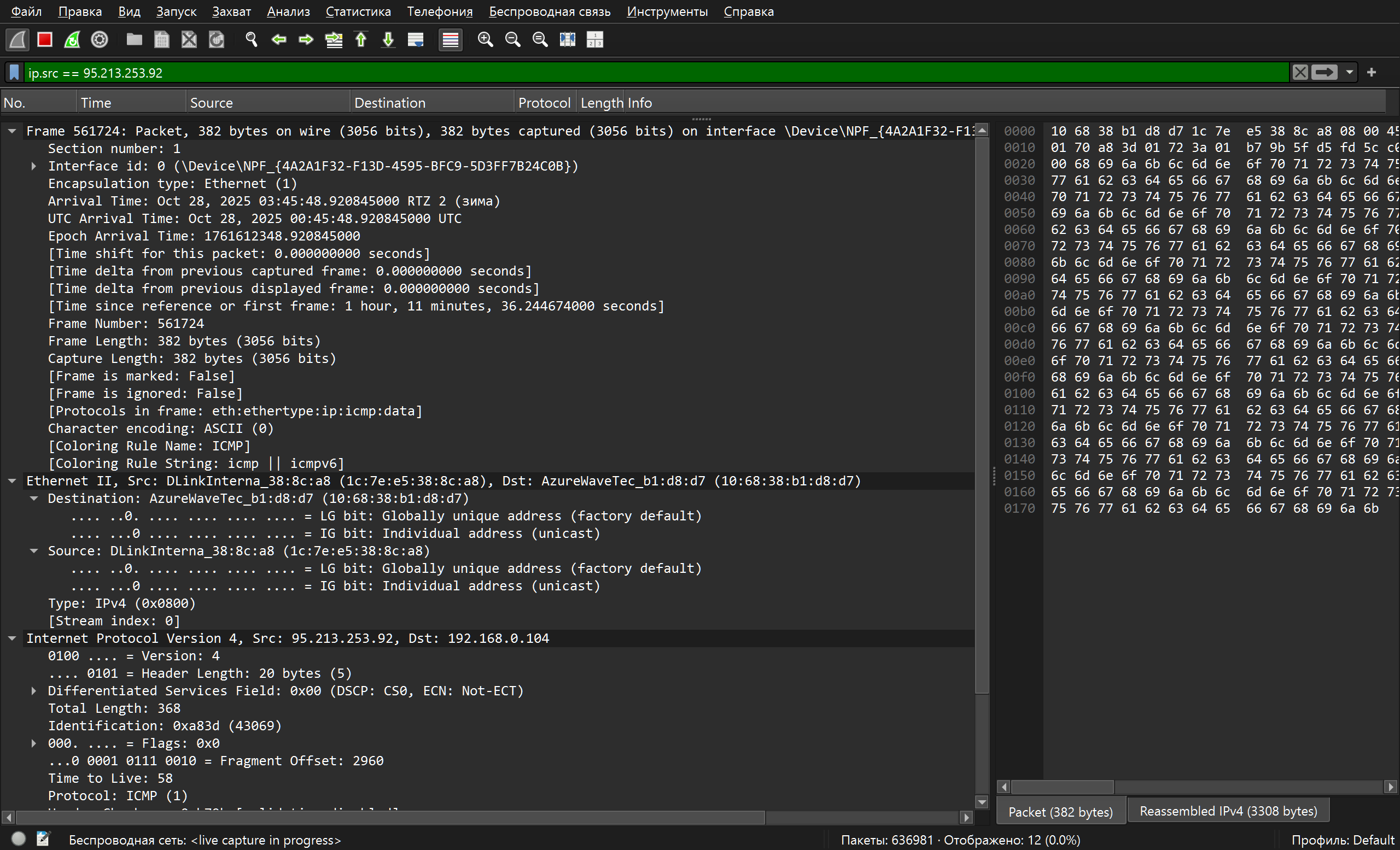


Рисунок 2.1 – ICMP для Echo Reply

Структура PDU при фрагментации IP-пакета

Уровень 2 (Канальный): Каждый фрагмент большого пакета передаётся отдельным Ethernet-кадром. Все они содержат MAC-адреса источника и назначения, а также поле EtherType = IPv4.

Уровень 3 (Сетевой): Каждый фрагмент формируется как отдельный IP-пакет, имеющий одинаковое значение поля Identification. В поле Fragment offset задаётся смещение данных текущего фрагмента относительно начала исходного пакета. Флаг MF (More Fragments) указывает, является ли фрагмент промежуточным (MF=1) или последним (MF=0).

Уровень 4 (Транспортный/Служебный): Полный заголовок ICMP (Echo Request или Reply) содержится только в первом фрагменте. Последующие фрагменты включают лишь часть данных (payload) из исходного ICMP-пакета.

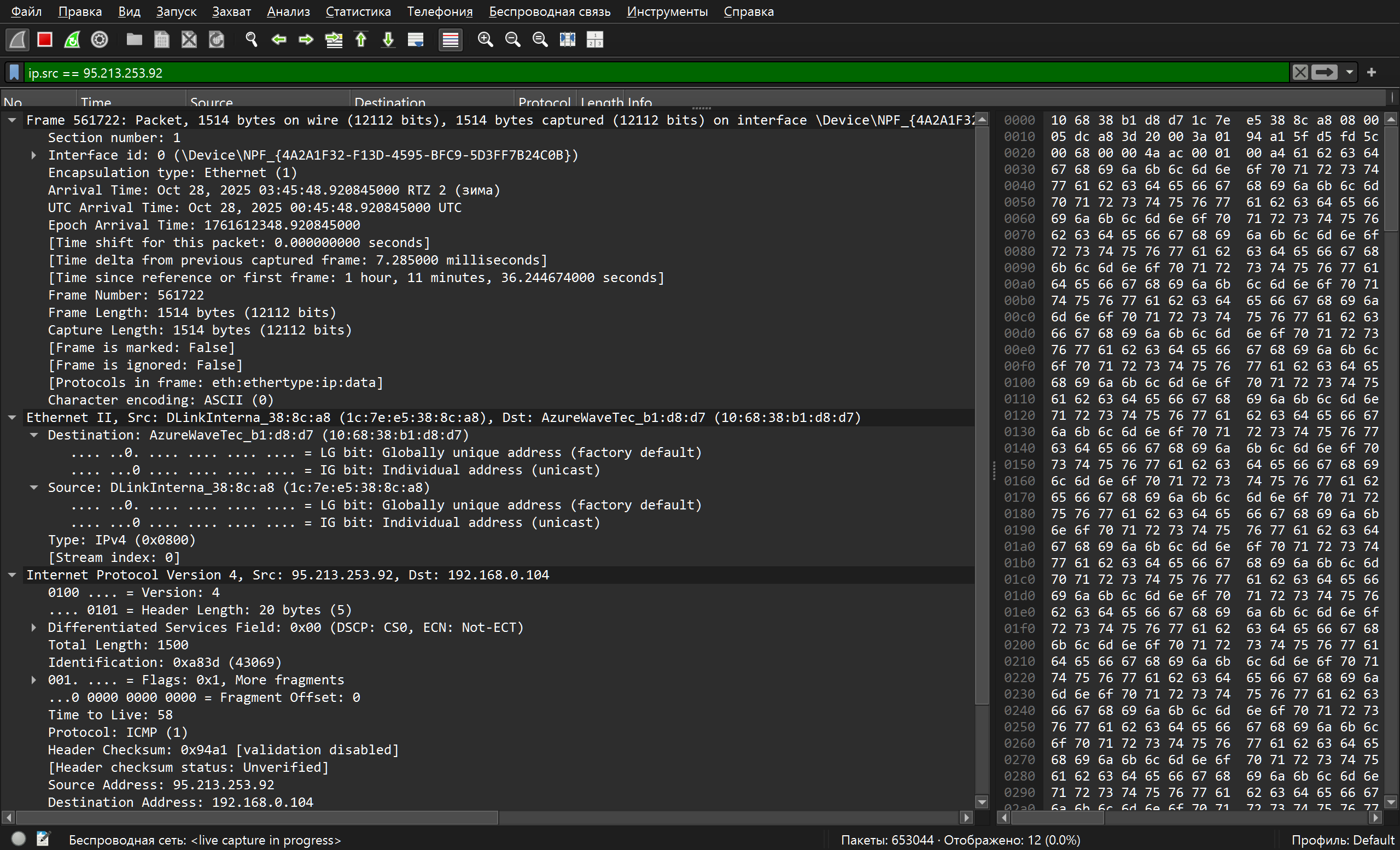


Рисунок 2.2 – Fragmented IP protocol

*2. Анализ трафика утилиты tracert (traceroute)*

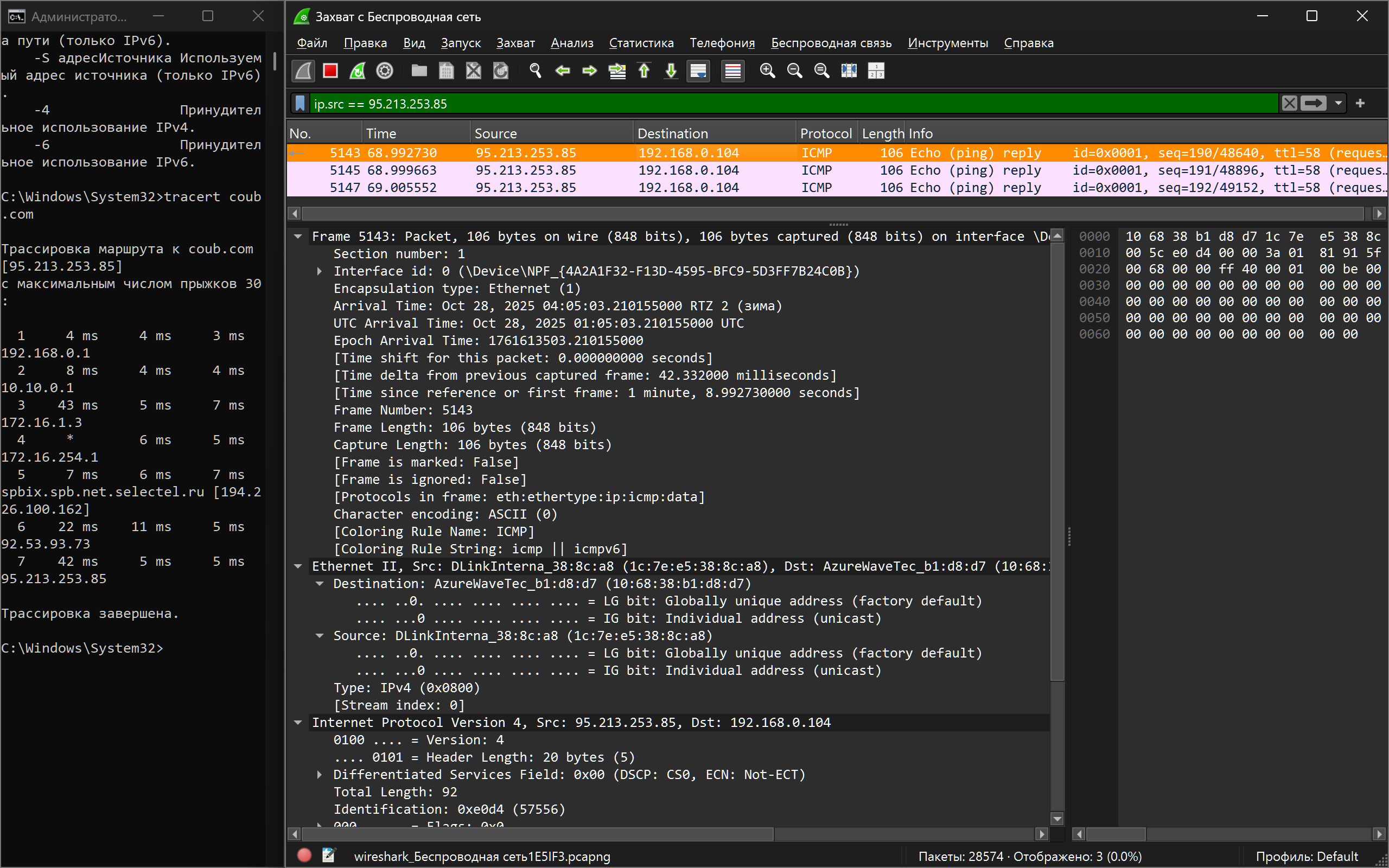
Был отслежен и проанализирован трафик утилитой Wireshark, создаваемый утилитой traceroute, при запуске из командной строки. 

Рисунок 3.1 – tracert coub.com

1. Сколько байт содержится в заголовке IP? Сколько байт содержится в поле данных?

* Заголовок IP: Стандартный заголовок IPv4 без опций имеет фиксированный размер 20 байт. Это видно в строке "Header Length: 20 bytes (5)" в Wireshark.
* Поле данных (Payload): Размер поля данных зависит от типа пакета.
  + В ICMP-пакетах, генерируемых tracert (как UDP-зонды, так и ответы), поле данных включает в себя:
    1. ICMP-заголовок (8 байт для сообщений об ошибках, таких как "Time Exceeded").
    2. Вложенный IP-заголовок и первые 8 байт исходного пакета, который вызвал ошибку (ещё 28 байт: 20 байт IP + 8 байт данных).
  + Таким образом, общий размер IP-пакета (заголовок + данные) для ICMP Time Exceeded обычно составляет 20 + 36 = 56 байт, что можно увидеть в поле "Total Length" заголовка IP.

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.

Как изменяется: Утилита tracert отправляет серию пакетов, в которых значение поля TTL последовательно увеличивается на 1, начиная с 1. Первый пакет имеет TTL=1, второй — TTL=2, третий — TTL=3 и так далее.

Почему это происходит: Это является основным алгоритмом работы утилиты.

Пакет с TTL=1 достигает первого маршрутизатора. Тот уменьшает TTL на 1, получает 0, отбрасывает пакет и отправляет обратно ICMP "Time-to-live exceeded". Так мы узнаём первый хоп.

Пакет с TTL=2 проходит первый маршрутизатор (TTL уменьшается до 1), но достигает второго маршрутизатора. Тот уменьшает TTL до 0, отбрасывает пакет и отправляет ошибку. Так мы узнаём второй хоп.

Этот процесс повторяется, пока пакет с достаточно большим TTL не достигнет целевого хоста. Ответ целевого хоста (например, "Port Unreachable") сигнализирует об окончании построения маршрута.

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

ICMP-пакеты, генерируемые утилитами tracert и ping, имеют фундаментальные различия в своей сути, несмотря на то что обе используют протокол ICMP для диагностики.

Утилита ping работает по принципу эхо-запроса. Она генерирует ICMP Echo Request (Тип 8) и ожидает в ответ ICMP Echo Reply (Тип 0) от целевого узла. Пакеты ping отправляются с большим значением TTL (обычно 64 или 128), чтобы гарантировать, что они достигнут конечной цели. Главная задача ping — подтвердить доступность узла и измерить время двусторонней задержки (RTT).

Утилита tracert (в ее реализации для Windows) работает по иному принципу. Она сама не генерирует ICMP Echo Request. Вместо этого она отправляет серию UDP-пакетов на заведомо неиспользуемый порт целевого узла. Ключевое отличие заключается в манипуляции полем TTL. tracert намеренно отправляет первые пакеты с TTL=1, затем с TTL=2 и так далее, чтобы каждый из них был отброшен на определенном промежуточном маршрутизаторе.

В ответ на эти отброшенные UDP-пакеты tracert получает не эхо-ответы, а сообщения об ошибках — ICMP Time-to-live exceeded (Тип 11) от каждого маршрутизатора на пути. Когда же UDP-пакет наконец достигает цели, утилита получает финальное сообщение об ошибке — ICMP Destination Unreachable (Port Unreachable) (Тип 3, Код 3). Таким образом, tracert использует ICMP-пакеты с ошибками как инструмент для построения карты маршрута, в то время как ping использует штатные ICMP-запросы и ответы для проверки связи.

4. Чем отличаются полученные пакеты «ICMP reply» от «ICMP error» и зачем нужны оба этих типа ответов?

ICMP Reply (Ответ):

Пример: ICMP Echo Reply (Тип 0).

Назначение: Это штатный, ожидаемый ответ на успешно доставленный запрос. Он означает: "Я получил твой пакет и корректно его обработал".

В tracert: Такой пакет приходит только от целевого хоста в конце маршрута (в виде ICMP Destination Unreachable, что для tracert является аналогом "порт недоступен, но хост достигнут").

ICMP Error (Ошибка):

Примеры: ICMP Time Exceeded (Тип 11), ICMP Destination Unreachable (Тип 3).

Назначение: Это сообщение об аномальной ситуации. Оно означает: "Я не смог доставить твой пакет по назначению, и вот причина".

В tracert: Эти пакеты приходят от промежуточных маршрутизаторов (Time Exceeded) и являются основным источником информации для построения маршрута.

Оба типа сообщений необходимы для диагностики сети. Reply подтверждает конечную доступность, а Error указывает на проблему на пути (например, обрыв на определённом участке, что без tracert и ICMP Errors было бы сложно локализовать).

5. Что изменится в работе tracert, если убрать ключ “-d”? Какой дополнительный трафик при этом будет генерироваться?

Ключ -d в Windows tracert отключает reverse DNS-запросы, то есть преобразование IP-адресов маршрутизаторов в их доменные имена.

Что изменится без -d: В выводе tracert вместо голых IP-адресов (например, 192.168.1.1) появятся попытки отобразить доменные имена маршрутизаторов (например, router1.isp.my-city.net).

Какой трафик сгенерируется: Для этого tracert будет инициировать обратные DNS-запросы (PTR-запросы) для каждого уникального IP-адреса, полученного в ходе работы. Это создаст дополнительный DNS-трафик (UDP-пакеты на порт 53), что может заметно увеличить время выполнения всей операции tracert, особенно если DNS-серверы отвечают медленно.

*3. Анализ HTTP-трафика*

Был отслежен и проанализирован HTTP-трафик, создаваемый браузером при посещении Интернет-сайта www.coub.com. В списке захваченных пакетов были проанализированы следующие пары HTTP-сообщений (запрос-ответ):

● GET-сообщение от клиента (браузера);

● ответ сервера.

При выполнении данного пункта возникли проблемы с условными GET-запросами. Они не приходили от браузера к веб-сайту. Это могло произойти по причине, того что сайт не поддерживает условные GET-запросы.

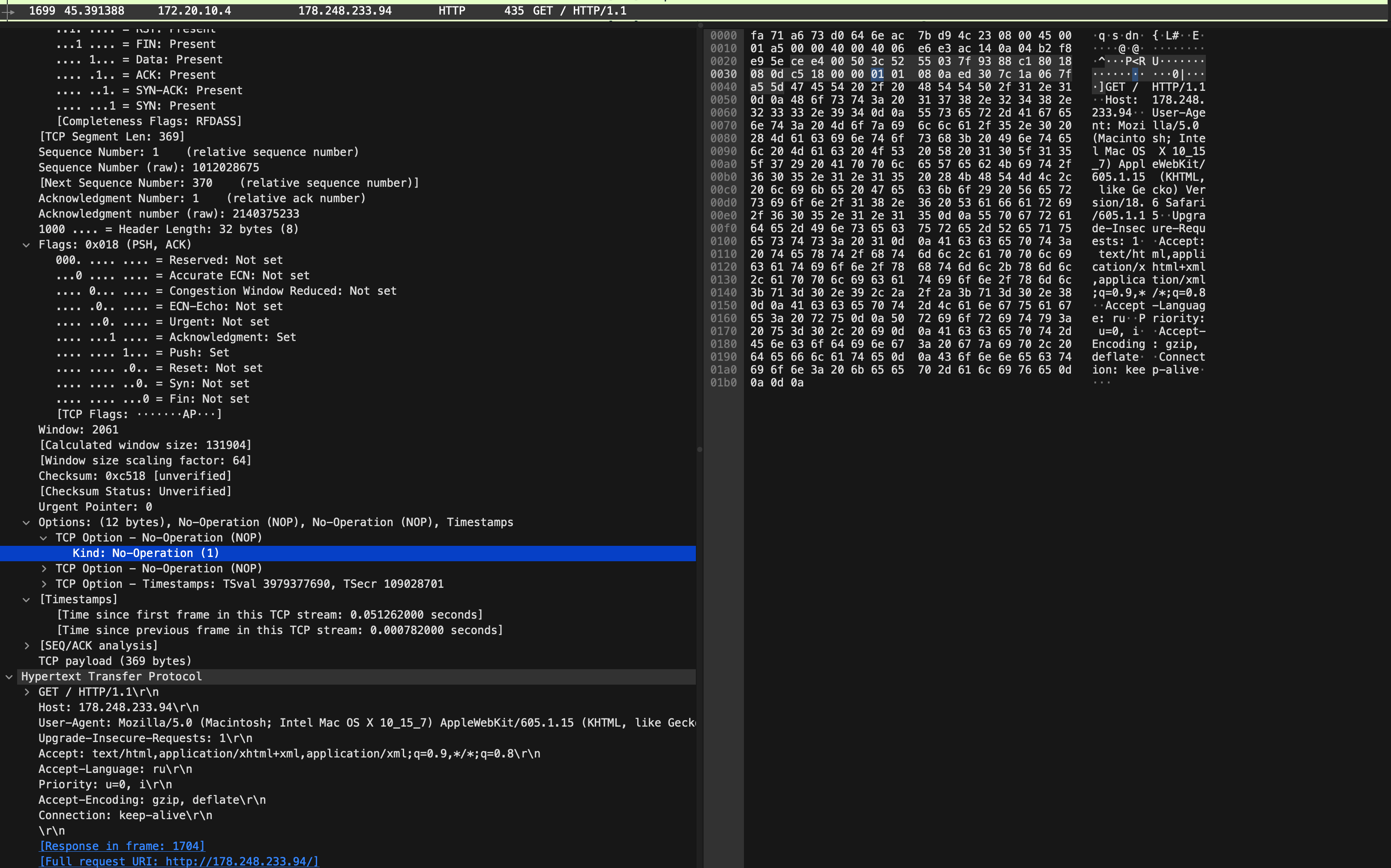


Рисунок 4.1 – get запрос

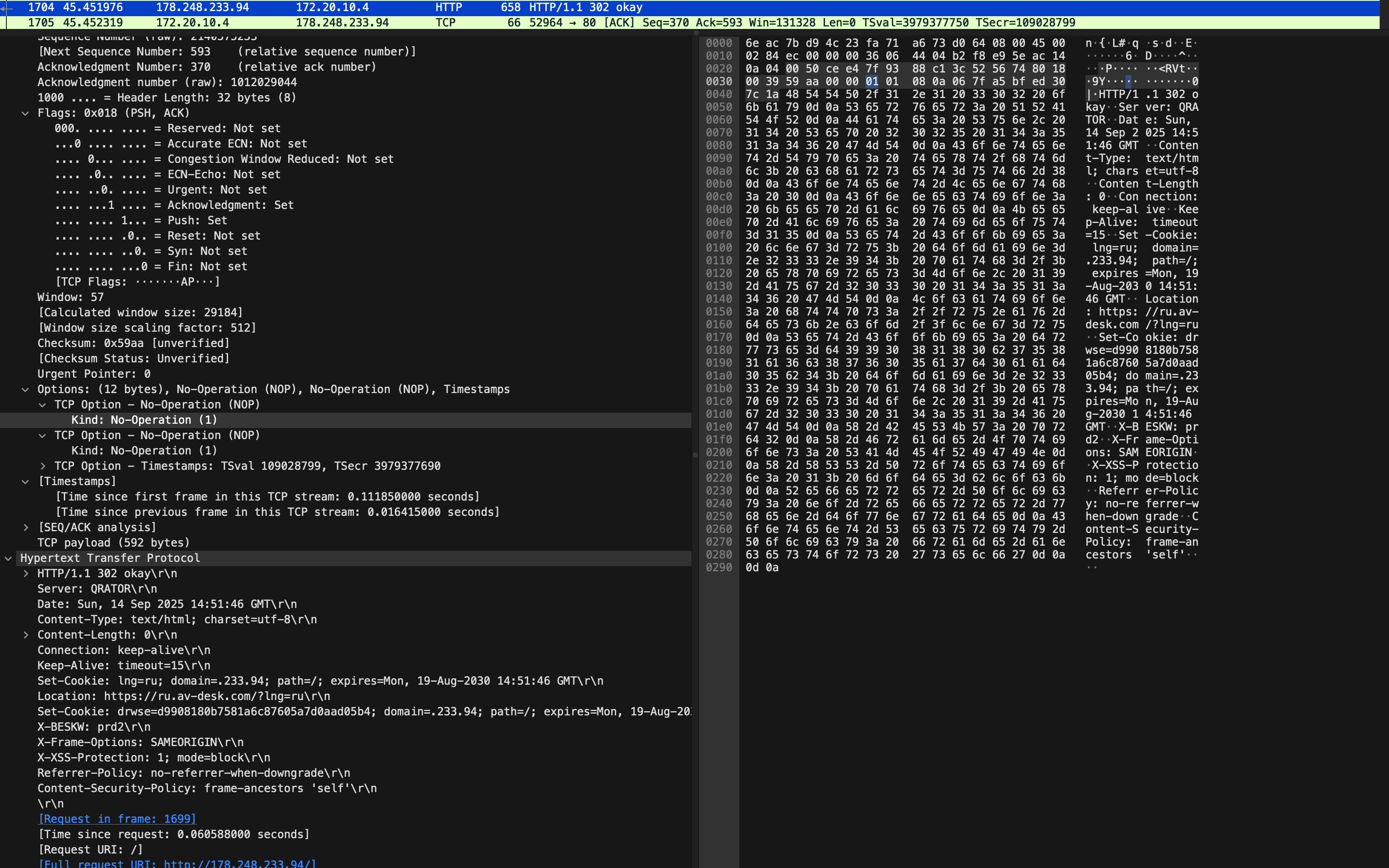


Рисунок 4.2 – Ответ сервера с переадресацией(код 302)

Структура PDU HTTP GET-запроса (первичное обращение)

* Уровень 2 (Канальный): Ethernet II кадр, содержащий MAC-адреса отправителя (клиента) и получателя (шлюза).
* Уровень 3 (Сетевой): Internet Protocol Version 4 пакет с IP-адресами источника (клиент) и назначения (сервер coub.com).
* Уровень 4 (Транспортный): Transmission Control Protocol сегмент. Устанавливается соединение между клиентом и сервером (порт клиента – случайный высокий, порт сервера – 80). Заголовок TCP содержит номера последовательностей и подтверждений, флаги (PSH, ACK), обеспечивающие надежную доставку.
* Уровень 7 (Прикладной): Hypertext Transfer Protocol – непосредственно HTTP-запрос.
  + Request Method: GET
  + Request URI: /
  + HTTP-Version: HTTP/1.1
  + Заголовки (Headers): Содержат метаинформацию: Host (адрес сервера), User-Agent (информация о браузере и ОС), Accept (типы контента, которые понимает клиент), Accept-Encoding (поддерживаемые методы сжатия).

# **3. Вывод:**