МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное автономное образовательное учреждение
высшего образования «Санкт-Петербургский политехнический
университет Петра Великого»

Институт компьютерных наук и кибербезопасности

Высшая школа технологий искусственного интеллекта

Направление: 02.03.01 Математика и компьютерные науки

Отчет по дисциплине «Сети ЭВМ и телекоммуникации компьютерных сетей» Лабораторная работа №2 «Анализ пакетного трафика»

Студент,
группа 5130201/20102 _____ Шклярова К. А.

Преподаватель _____ Мулюха В.А.

«_____»_____ 2025 г.

Санкт-Петербург, 2025

1 Постановка задачи

Необходимо установить и запустить программу windump (с соответствующими библиотеками), windump лучше запускать с опциями -XX -s 128 -S -e. Вывод windump можно перенаправить в файл.

Не останавливая windump выполнить последовательно:

1. Команду ping того адреса, который написан;
2. Команду tracert того адреса, который указан
3. Войти через браузер на третий адрес

Остановить windump

В отчёте привести трафик, соответствующий проводимым действиям и уметь ответить на вопросы какая строчка вывода соответствует какому действию и что вообще делает. Где какое поле и т.п.

Если между действиями в windump попал другой трафик, то часть вывода можно сократить, приведя только нужный.

Вариант	Ping	tracert/traceroute	web-доступ
35	github.com	biobiochile.cl	newzealand.com

2 Предварительная настройка

- Скачиваем Windump.exe, WinPCap.exe в `C:\Program Files`
- Скачиваем WireShark, чтобы открывать файл с захваченным трафиком (с расширением pcap)

3 Алгоритм действий

1. Открываем командную строку от имени администратора:

   ```
   cd C:\Program Files
   ```

2. Выводим список доступных сетевых интерфейсов для WinDump:

   ```
   windump -D
   1.\Device\NPF_{C18AAF6E-FDDD-4B63-A9EC-8536A386771C} (Microsoft)
   2.\Device\NPF_{E76D7E5B-C086-48E0-9F4B-65304A86D5D4} (Microsoft)
   3.\Device\NPF_{F1146FB5-D291-44D6-B27E-EAAE0F6DE8FF} (Microsoft)
   ```

   - `\Device\NPF_{...}` — уникальный идентификатор (GUID) сетевого интерфейса в Windows.
   - NPF — NetGroup Packet Filter (драйвер WinPcap/Npcap для захвата пакетов).
   - (Microsoft) — описание интерфейса.

После определения нужного интерфейса (в данном случае №3), можно запускать WinDump с параметром -i 3 (interface).

3.1 Ping

1. Запускаем windump с определенными параметрами:

   ```
   windump -XX -s 128 -S -e -i 3 -w D:\Seti\ping.pcap icmp
   ```

   - -XX — выводит данные каждого пакета в шестнадцатеричном формате и ASCII.
   - -s 128 — захватывает первых 128 байт каждого пакета.
   - -S — отображает абсолютные номера последовательности TCP.
   - -e — выводит MAC-адреса и тип кадра.
   - -i 3 — захватывает трафик 3-го интерфейса (физический сетевой).

- -w D:\Seti\ping.pcap — сохраняет трафик в файл.
- icmp — Internet Control Message Protocol.

2. Открываем 2-ю консоль cmd.
3. Выполняем команду:

```
ping github.com
```

4. Останавливаем windump в 1-ой консоли: Ctrl+C.

3.2 Tracert

1. Запуск windump с определенными параметрами:

```
windump -XX -s 128 -S -e -i 3 -v -w D:\Seti\tracert.pcap icmp
```

- -v — для вывода дополнительной информации.

2. Открываем 2-ю консоль cmd.
3. Выполняем команду:

```
tracert biobiochile.cl
```

4. Останавливаем windump в 1-ой консоли: Ctrl+C.

3.3 Web-доступ

1. Запуск windump с определенными параметрами:

```
windump -XX -s 128 -S -e -i 3 -v -w D:\Seti\web.pcap host newzealand.com
```

2. Открываем 2-ю консоль cmd.
3. Выполняем команду:

```
curl -v http://newzealand.com
```

4. Останавливаем windump в 1-ой консоли: Ctrl+C.

4 Аналитика результата windump

4.1 PING

Результат выполнения представлен на Рис. 1.

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.0.104	140.82.121.4	ICMP	74	Echo (ping) request id=0x0001, seq=167/42752, ttl=128 (reply in 2)
2	0.058526	140.82.121.4	192.168.0.104	ICMP	74	Echo (ping) reply id=0x0001, seq=167/42752, ttl=53 (request in 1)
3	1.015365	192.168.0.104	140.82.121.4	ICMP	74	Echo (ping) request id=0x0001, seq=168/43008, ttl=128 (reply in 4)
4	1.079001	140.82.121.4	192.168.0.104	ICMP	74	Echo (ping) reply id=0x0001, seq=168/43008, ttl=53 (request in 3)
5	2.025730	192.168.0.104	140.82.121.4	ICMP	74	Echo (ping) request id=0x0001, seq=169/43264, ttl=128 (reply in 6)
6	2.089806	140.82.121.4	192.168.0.104	ICMP	74	Echo (ping) reply id=0x0001, seq=169/43264, ttl=53 (request in 5)
7	3.035562	192.168.0.104	140.82.121.4	ICMP	74	Echo (ping) request id=0x0001, seq=170/43520, ttl=128 (reply in 8)
8	3.095569	140.82.121.4	192.168.0.104	ICMP	74	Echo (ping) reply id=0x0001, seq=170/43520, ttl=53 (request in 7)

Рис. 1. Ping

4.1.1 Echo request – эхо запрос

- 192.168.0.104 — локальный IP компьютера.
- 140.82.121.4 — IP сервера github.com.
- ICMP — протокол.
- 74 байта — Размер пакета (заголовок + данные).
- Echo request — Тип пакета (запрос ping).
- id=0x0001 — Уникальный идентификатор, позволяющий сопоставить запрос и ответ.
- seq=167/42752 — Номер последовательности, отслеживает порядок запросов (167 — номер в десятичной системе, 42752 — в машинном формате 167*256).
- ttl=128 — Time To Live (время жизни пакета), максимальное количество переходов между маршрутизаторами.
- (reply in 2) — Ответ пришёл в строке 2.

Подробнее (идем по всем уровням в 16-ном дампе)

```
0000   78 8c b5 e1 f3 a8 2c 7b a0 b3 50 e7 08 00 45 00   x.....,{..P...E.
0010   00 3c 65 4d 00 00 80 01 00 00 c0 a8 00 68 8c 52   .<eM.........h.R
0020   79 04 08 00 4c b4 00 01 00 a7 61 62 63 64 65 66   y...L.....abcdef
0030   67 68 69 6a 6b 6c 6d 6e 6f 70 71 72 73 74 75 76   ghijklmnopqrstuv
0040   77 61 62 63 64 65 66 67 68 69                     wabcdefghi
```

Общая структура пакета: Frame 1: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) - вся информация захвачена, нет обрезки.

Уровень 1: Ethernet (канальный уровень - кадр данных)

Ethernet II, Src: Intel_b3:50:e7 (2c:7b:a0:b3:50:e7), Dst: TPLink_e1:f3:a8 (78:8c:b5:e1:f3:a8)

- Ethernet II - тип кадра.
- Назначение (Dst) 78:8c:b5:e1:f3:a8 - MAC-адрес получателя (роутер).
- Источник (Src) 2c:7b:a0:b3:50:e7 - MAC-адрес отправителя (физический адрес сетевого адаптера).
- Протокол 0x0800 (в hex-дампе) - Указывает на IPv4.

Hex-дамп Ethernet (первые 14 байт):

```
0000   78 8c b5 e1 f3 a8 2c 7b a0 b3 50 e7 08 00    x.....,{..P...
```

- Первые 6 байт: MAC назначения (78:8c:b5:e1:f3:a8) - старшие 3 байта - ID производителя.
- Следующие 6 байт: MAC источника (2c:7b:a0:b3:50:e7).
- Последние 2 байта: 08 00 — тип Ethernet (IPv4).

Уровень 2: IPv4 (сетевой уровень - пакет)

Internet Protocol Version 4, Src: 192.168.0.104, Dst: 140.82.121.4

Hex-дамп IPv4 (байты 15–34):

```
0000                                              45 00                E.
0010   00 3c 65 4d 00 00 80 01 00 00 c0 a8 00 68 8c 52   .<eM.........h.R
0020   79 04                                           y.
```

- 45: Версия (4) + длина заголовка (5) → 20 байт весь заголовок IP-пакета.
- 00: ToS - тип сервиса (информация о приоритете трафика).
- 3c: Общая длина пакета = 60 байт (включая заголовки).
- 65 4d: Уникальный ID пакета.
- 00 00: Флаги + смещение - Фрагментация не используется.
- 80: TTL=128.
- 01: протокол (указывает какому протоколу верхнего уровня принадлежат данные IP-пакета - ICMP).
- 00 00: Контрольная сумма заголовка (для проверки целостности).
- c0 a8 00 68: IP отправителя (192.168.0.104) - компьютер.
- 8c 52 79 04: IP получателя (140.82.121.4) - сервер.

Уровень 2.1: ICMP (сетевой уровень - протокол обмена управляющими сообщениями)

Hex-дамп ICMP (байты 34–74):

```
0020          08 00 4c b4 00 01 00 a7 61 62 63 64 65 66      ..L.....abcdef
0030    67 68 69 6a 6b 6c 6d 6e 6f 70 71 72 73 74 75 76    ghijklmnopqrstuv
0040    77 61 62 63 64 65 66 67 68 69                      wabcdefghi
```

- 8 байт - заголовок:
  - 08 00: Тип+код - эхо запрос - Echo Request.
  - 4c b4: Контрольная сумма.
  - 00 01: уникальный ID сессии.
  - 00 a7: Номер последовательности (0x00a7 = 167 порядковый номер пакета).
- 32 байта - данные:
  - 61 62...: полезная нагрузка (abcdefgh...).

4.1.2 Echo reply – эхо ответ

- 140.82.121.4 — Отправитель (сервер github.com).
- 192.168.0.104 — Получатель (компьютер).
- Echo reply — Тип пакета (это ответ на ping).
- id=0x0001, seq=167/42752 — ID и seq те же, что в запросе.
- ttl=53 — пакет прошёл через 64 - 53 = 11 маршрутизаторов.

Полезная информация:

- 1) TTL уменьшается, т.к.:
  Каждый маршрутизатор уменьшает TTL на 1.
  Если TTL достигает 0, пакет отбрасывается (защита от «вечной» передачи).
  Здесь 128 - 53 = 75 значит, пакет прошёл 75 узлов до сервера.
- 2) seq увеличивается, т.к.
  seq = 167, 168, 169, 170 – это номера пакетов.
  Ping отправляет несколько запросов подряд, нумеруя их для отслеживания.
- 3)Размер пакета = 74 байта:
  Заголовок Ethernet = 14 байт,
  Заголовок IP = 20 байт,
  Заголовок ICMP = 8 байт,
  Данные ping = 32 байта.

4.2 Tracert

Tracert — утилита командной строки, используемая для отслеживания маршрута, по которому пакеты данных достигают определённого пункта назначения через IP-сеть.

Tracert использует ICMP-запросы:

- ICMP Echo Request (исходящие)
- ICMP Time Exceeded (ответы от промежуточных узлов)
- ICMP Echo Reply (ответ от конечного узла)

Три числа в графе «Время» при выполнении команды представляют собой задержки (в миллисекундах) для трёх отдельных пробных пакетов, отправленных к каждому узлу на пути.

- Если все три числа близки (например, 10 мс, 11 мс, 10 мс) — соединение стабильное.
- Если разброс большой (например, 10 мс, 150 мс, 30 мс) — возможны проблемы (перегрузка сети).
- Если есть * — узел не ответил на один или несколько пакетов.

Результат выполнения в консоли представлен на Рис. 2.

```
Трассировка маршрута к biobiochile.cl [190.153.209.180]
с максимальным числом прыжков 30:

  1     3 ms     1 ms     1 ms  192.168.0.1
  2     2 ms     2 ms     2 ms  192.145.19.254
  3     *        *        *     Превышен интервал ожидания для запроса.
  4     4 ms     3 ms     3 ms  10.202.1.5
  5     4 ms     3 ms     2 ms  172-128-236-178.maloco.ru [178.236.128.172]
  6     8 ms     3 ms    76 ms  spb-ivc-cr2.ae45-426.rascom.as20764.net [80.64.98.210]

  7     4 ms     3 ms     3 ms  spb-ivc-cr2.ae667-3006.rascom.as20764.net [80.64.97.70]
  8     *        *        *     Превышен интервал ожидания для запроса.
  9   268 ms   203 ms   202 ms  195.66.227.231
 10     *        *        *     Превышен интервал ожидания для запроса.
 11   279 ms   304 ms   306 ms  190.211.167.14
 12   283 ms   305 ms   304 ms  192.168.224.221
 13   284 ms   237 ms   272 ms  re1.et2-2-0.53.cn1.gtdinternet.com [190.196.126.77]
 14   286 ms   303 ms   305 ms  static.190.153.209.180.gtdinternet.com [190.153.209.180]

Трассировка завершена.
```

Рис. 2. Trace

№ прыжка	Время (мс)	IP/Имя узла	Что это означает?
1	3, 1, 1	192.168.0.1 (роутер)	Роутер.
2	2, 2, 2	192.145.19.254	Шлюз интернет-провайдера.
3	*, *, *	Нет ответа	Узлы не отвечают на запросы
4	4, 3, 3	10.202.1.5	Внутренний IP
5	4, 3, 2	178.236.128.172	Узел в России
6	8, 3, 76	80.64.98.210	Узел в Санкт-Петербурге
7	4, 3, 3	80.64.97.70	Узел в Санкт-Петербурге
8	*, *, *	Нет ответа	Узлы не отвечают на запросы
9	268, 203, 202	195.66.227.231	Узел в Лондоне (европейский маршрутизатор)
10	*, *, *	Нет ответа	Узлы не отвечают на запросы
11	279, 304, 306	190.211.167.14	Узел в Чили
12	283, 305, 304	192.168.224.221	Узел в Чили
13	284, 237, 272	190.196.126.77	Узел в Чили
14	286, 303, 305	190.153.209.180	Конечный сервер biobiochile.cl

Сайт для просмотра данных по IP-адресу: https://ipinfo.io/.

Результат выполнения windump представлен на Рис. 3, Рис. 4, Рис. **??**.

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.0.101	190.153.209.180	ICMP	106	Echo (ping) request id=0x0001, seq=97/24832, ttl=1 (no response found!)
2	0.003212	192.168.0.1	192.168.0.101	ICMP	126	Time-to-live exceeded (Time to live exceeded in transit)
3	0.007162	192.168.0.101	190.153.209.180	ICMP	106	Echo (ping) request id=0x0001, seq=98/25088, ttl=1 (no response found!)
4	0.008354	192.168.0.1	192.168.0.101	ICMP	126	Time-to-live exceeded (Time to live exceeded in transit)
5	0.008969	192.168.0.101	190.153.209.180	ICMP	106	Echo (ping) request id=0x0001, seq=99/25344, ttl=1 (no response found!)
6	0.010072	192.168.0.1	192.168.0.101	ICMP	126	Time-to-live exceeded (Time to live exceeded in transit)
7	5.965848	192.168.0.101	190.153.209.180	ICMP	106	Echo (ping) request id=0x0001, seq=100/25600, ttl=2 (no response found!)
8	5.968594	192.145.19.254	192.168.0.101	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
9	5.969173	192.168.0.101	190.153.209.180	ICMP	106	Echo (ping) request id=0x0001, seq=101/25856, ttl=2 (no response found!)
10	5.971220	192.145.19.254	192.168.0.101	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
11	5.971622	192.168.0.101	190.153.209.180	ICMP	106	Echo (ping) request id=0x0001, seq=102/26112, ttl=2 (no response found!)
12	5.974112	192.145.19.254	192.168.0.101	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
13	11.941595	192.168.0.101	190.153.209.180	ICMP	106	Echo (ping) request id=0x0001, seq=103/26368, ttl=3 (no response found!)
14	15.567350	192.168.0.101	190.153.209.180	ICMP	106	Echo (ping) request id=0x0001, seq=104/26624, ttl=3 (no response found!)
15	19.574084	192.168.0.101	190.153.209.180	ICMP	106	Echo (ping) request id=0x0001, seq=105/26880, ttl=3 (no response found!)
16	23.577026	192.168.0.101	190.153.209.180	ICMP	106	Echo (ping) request id=0x0001, seq=106/27136, ttl=4 (no response found!)
17	23.581583	10.202.1.5	192.168.0.101	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
18	23.582457	192.168.0.101	190.153.209.180	ICMP	106	Echo (ping) request id=0x0001, seq=107/27392, ttl=4 (no response found!)
19	23.585786	10.202.1.5	192.168.0.101	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
20	23.586706	192.168.0.101	190.153.209.180	ICMP	106	Echo (ping) request id=0x0001, seq=108/27648, ttl=4 (no response found!)
21	23.589620	10.202.1.5	192.168.0.101	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
22	29.530339	192.168.0.101	190.153.209.180	ICMP	106	Echo (ping) request id=0x0001, seq=109/27904, ttl=5 (no response found!)
23	29.534714	178.236.128.172	192.168.0.101	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
24	29.535633	192.168.0.101	190.153.209.180	ICMP	106	Echo (ping) request id=0x0001, seq=110/28160, ttl=5 (no response found!)
25	29.539283	178.236.128.172	192.168.0.101	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
26	29.540103	192.168.0.101	190.153.209.180	ICMP	106	Echo (ping) request id=0x0001, seq=111/28416, ttl=5 (no response found!)
27	29.542951	178.236.128.172	192.168.0.101	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
28	30.553055	192.168.0.101	190.153.209.180	ICMP	106	Echo (ping) request id=0x0001, seq=112/28672, ttl=6 (no response found!)
29	30.561616	80.64.98.210	192.168.0.101	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
30	30.562377	192.168.0.101	190.153.209.180	ICMP	106	Echo (ping) request id=0x0001, seq=113/28928, ttl=6 (no response found!)
31	30.565916	80.64.98.210	192.168.0.101	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
32	30.566720	192.168.0.101	190.153.209.180	ICMP	106	Echo (ping) request id=0x0001, seq=114/29184, ttl=6 (no response found!)
33	30.643026	80.64.98.210	192.168.0.101	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
34	31.581439	192.168.0.101	190.153.209.180	ICMP	106	Echo (ping) request id=0x0001, seq=115/29440, ttl=7 (no response found!)
35	31.586271	80.64.97.70	192.168.0.101	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
36	31.590236	192.168.0.101	190.153.209.180	ICMP	106	Echo (ping) request id=0x0001, seq=116/29696, ttl=7 (no response found!)
37	31.593260	80.64.97.70	192.168.0.101	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
38	31.594100	192.168.0.101	190.153.209.180	ICMP	106	Echo (ping) request id=0x0001, seq=117/29952, ttl=7 (no response found!)
39	31.597299	80.64.97.70	192.168.0.101	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
40	32.615942	192.168.0.101	190.153.209.180	ICMP	106	Echo (ping) request id=0x0001, seq=118/30208, ttl=8 (no response found!)

Рис. 3. Tracert: windump

41	36.573091	192.168.0.101	190.153.209.180	ICMP	106	Echo (ping) request id=0x0001, seq=119/30464, ttl=8 (no response found!)
42	40.575227	192.168.0.101	190.153.209.180	ICMP	106	Echo (ping) request id=0x0001, seq=120/30720, ttl=8 (no response found!)
43	44.567482	192.168.0.101	190.153.209.180	ICMP	106	Echo (ping) request id=0x0001, seq=121/30976, ttl=9 (no response found!)
44	44.835445	195.66.227.231	192.168.0.101	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
45	44.837258	192.168.0.101	190.153.209.180	ICMP	106	Echo (ping) request id=0x0001, seq=122/31232, ttl=9 (no response found!)
46	45.040954	195.66.227.231	192.168.0.101	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
47	45.043811	192.168.0.101	190.153.209.180	ICMP	106	Echo (ping) request id=0x0001, seq=123/31488, ttl=9 (no response found!)
48	45.246473	195.66.227.231	192.168.0.101	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
49	51.036666	192.168.0.101	190.153.209.180	ICMP	106	Echo (ping) request id=0x0001, seq=124/31744, ttl=10 (no response found!)
50	54.575476	192.168.0.101	190.153.209.180	ICMP	106	Echo (ping) request id=0x0001, seq=125/32000, ttl=10 (no response found!)
51	58.572712	192.168.0.101	190.153.209.180	ICMP	106	Echo (ping) request id=0x0001, seq=126/32256, ttl=10 (no response found!)
52	62.578965	192.168.0.101	190.153.209.180	ICMP	106	Echo (ping) request id=0x0001, seq=127/32512, ttl=11 (no response found!)
53	62.858563	190.211.167.14	192.168.0.101	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
54	62.860350	192.168.0.101	190.153.209.180	ICMP	106	Echo (ping) request id=0x0001, seq=128/32768, ttl=11 (no response found!)
55	63.165049	190.211.167.14	192.168.0.101	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
56	63.167796	192.168.0.101	190.153.209.180	ICMP	106	Echo (ping) request id=0x0001, seq=129/33024, ttl=11 (no response found!)
57	63.474045	190.211.167.14	192.168.0.101	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
58	69.128106	192.168.0.101	190.153.209.180	ICMP	106	Echo (ping) request id=0x0001, seq=130/33280, ttl=12 (no response found!)
59	69.411781	192.168.224.221	192.168.0.101	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
60	69.413465	192.168.0.101	190.153.209.180	ICMP	106	Echo (ping) request id=0x0001, seq=131/33536, ttl=12 (no response found!)
61	69.719004	192.168.224.221	192.168.0.101	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
62	69.720731	192.168.0.101	190.153.209.180	ICMP	106	Echo (ping) request id=0x0001, seq=132/33792, ttl=12 (no response found!)
63	70.025300	192.168.224.221	192.168.0.101	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
64	75.681375	192.168.0.101	190.153.209.180	ICMP	106	Echo (ping) request id=0x0001, seq=133/34048, ttl=13 (no response found!)
65	75.965507	190.196.126.77	192.168.0.101	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
66	75.967262	192.168.0.101	190.153.209.180	ICMP	106	Echo (ping) request id=0x0001, seq=134/34304, ttl=13 (no response found!)
67	76.204594	190.196.126.77	192.168.0.101	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
68	76.206402	192.168.0.101	190.153.209.180	ICMP	106	Echo (ping) request id=0x0001, seq=135/34560, ttl=13 (no response found!)
69	76.478909	190.196.126.77	192.168.0.101	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
70	78.034942	192.168.0.101	190.153.209.180	ICMP	106	Echo (ping) request id=0x0001, seq=136/34816, ttl=14 (reply in 71)
71	78.321326	190.153.209.180	192.168.0.101	ICMP	106	Echo (ping) reply id=0x0001, seq=136/34816, ttl=48 (request in 70)
72	78.323967	192.168.0.101	190.153.209.180	ICMP	106	Echo (ping) request id=0x0001, seq=137/35072, ttl=14 (reply in 73)
73	78.627789	190.153.209.180	192.168.0.101	ICMP	106	Echo (ping) reply id=0x0001, seq=137/35072, ttl=48 (request in 72)
74	78.629402	192.168.0.101	190.153.209.180	ICMP	106	Echo (ping) request id=0x0001, seq=138/35328, ttl=14 (reply in 75)
75	78.935086	190.153.209.180	192.168.0.101	ICMP	106	Echo (ping) reply id=0x0001, seq=138/35328, ttl=48 (request in 74)

Рис. 4. Tracert: windump

Подробнее (идем по всем уровням в 16-ном дампе) для 1-го пакета

Общая структура пакета: Frame 1: 106 bytes on wire (848 bits), 106 bytes captured (848 bits).

```
0000   78 8c b5 e1 f3 a8 2c 7b a0 b3 50 e7 08 00 45 00   x.....,{..P...E.
0010   00 5c 1c b7 00 00 01 01 00 00 c0 a8 00 65 be 99   .\...........e..
0020   d1 b4 08 00 f7 9d 00 01 00 61 00 00 00 00 00 00   .........a......
0030   00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
0040   00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
0050   00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
0060   00 00 00 00 00 00 00 00 00 00                     ..........
```

По сути все аналогично ping, только будет TTL уменьшаться и увеличиваться в пакетах.

Уровень 1: Ethernet (канальный уровень - кадр данных)

Hex-дамп Ethernet (первые 14 байт):

```
0000   78 8c b5 e1 f3 a8 2c 7b a0 b3 50 e7 08 00         x.....,{..P...
```

- Первые 6 байт: MAC назначения (78 8c b5 e1 f3 a8) - старшие 3 байта - ID производителя.
- Следующие 6 байт: MAC источника (2c 7b a0 b3 50 e7).
- Последние 2 байта: 08 00 — тип Ethernet (IPv4).

Уровень 2: IPv4 (сетевой уровень - пакет)

Internet Protocol Version 4, Src: 192.168.0.175, Dst: 195.153.209.180

Hex-дамп IPv4 (байты 15–34):

```
0000                                              45 00                E.
0010   00 5c 1c b7 00 00 01 01 00 00 c0 a8 00 65 be 99   .\...........e..
0020   d1 b4                                           ..
```

- 45: Версия (4) + длина заголовка (5) → 20 байт весь заголовок IP-пакета.
- 00: ToS - тип сервиса (информация о приоритете трафика) - обычный трафик.
- 5c: Общая длина пакета = 92 байта (включая заголовки).
- 8a 1a: Уникальный ID пакета (0x8a1a = 35354 в 10сс).
- 00 00: Флаги + смещение - фрагментация не используется.
- 01: TTL=1.

- 01: протокол (указывает какому протоколу верхнего уровня принадлежат данные IP-пакета - ICMP).
- 00 00: Контрольная сумма (для проверки целостности).
- c0 a8 00 65: IP отправителя (192.168.0.101) - ПК.
- be 99 d1 b4: IP получателя (190.153.209.180) - biobiochile.cl.

Уровень 2.1: ICMP (сетевой уровень - протокол обмена управляющими сообщениями)

Hex-дамп ICMP (байты 34–106):

```
0020          08 00 f7 9d 00 01 00 61 00 00 00 00 00 00   ................
0030   00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
0040   00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
0050   00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
0060   00 00 00 00 00 00 00 00 00 00                     ..........
```

- 8 байт - заголовок:
  - 08 00: Тип+код - эхо запрос - Echo Request.
  - f7 9d: Контрольная сумма.
  - 00 01: уникальный ID сессии.
  - 00 61: Номер последовательности (0x0061 = 97 десятичное).
- данные: 00 00...: Данные - нули.

Подробнее (идем по всем уровням в 16-ном дампе) для 2-го пакета - ответ на 1ый

Frame 2: 126 bytes on wire (1008 bits), 126 bytes captured (1008 bits)

```
0000   2c 7b a0 b3 50 e7 78 8c b5 e1 f3 a8 08 00 45 00   ,{..P.x.......E.
0010   00 70 ec c0 00 00 40 01 0c 16 c0 a8 00 01 c0 a8   .p....@.........
0020   00 65 0b 00 f4 ff 00 00 00 00 45 00 00 5c 1c b7   .e........E..\..
0030   00 00 01 01 4b 8f c0 a8 00 65 be 99 d1 b4 08 00   ....K....e......
0040   f7 9d 00 01 00 61 00 00 00 00 00 00 00 00 00 00   .....a..........
0050   00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
0060   00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
0070   00 00 00 00 00 00 00 00 00 00 00 00 00 00         ..............
```

Уровень 1: Ethernet (канальный уровень - кадр данных)

Hex-дамп Ethernet (первые 14 байт):

```
0000   2c 7b a0 b3 50 e7 78 8c b5 e1 f3 a8 08 00         ,{..P.x.......
```

- Первые 6 байт: MAC назначения (2c 7b a0 b3 50 e7) - ПК.
- Следующие 6 байт: MAC источника (78 8c b5 e1 f3 a8) - роутер.
- Последние 2 байта: 08 00 — тип Ethernet (IPv4).

Уровень 2: IPv4 (сетевой уровень - пакет)

Hex-дамп IPv4 (байты 15–34):

```
0000                                                 45 00                E.
0010   00 70 ec c0 00 00 40 01 0c 16 c0 a8 00 01 c0 a8   .p....@.........
0020   00 65                                            .e
```

- 45: Версия (4) + длина заголовка (5) → 20 байт весь заголовок IP-пакета.
- 00: ToS - тип сервиса управляющий трафик.
- 00 70: Общая длина пакета = 112 байт (включая заголовки).
- ec c0: Уникальный ID пакета.
- 00 00: Флаги + смещение - фрагментация не используется.
- 40: TTL (64 - стандартное значение для Linux-роутеров).
- 01: протокол (указывает какому протоколу верхнего уровня принадлежат данные IP-пакета - ICMP).
- 0c 16: Контрольная сумма (для проверки целостности).
- c0 a8 00 01: IP отправителя 192.168.0.1 (роутер).
- c0 a8 00 65: IP получателя 192.168.0.101 (ПК).

Уровень 2.1: ICMP (сетевой уровень - протокол обмена управляющими сообщениями)

Hex-дамп ICMP (байты 34–128):

```
0020         0b 00 f4 ff 00 00 00 00 45 00 00 5c 1c b7     ........E..\..
0030   00 00 01 01 4b 8f c0 a8 00 65 be 99 d1 b4 08 00   ....K....e......
0040   f7 9d 00 01 00 61 00 00 00 00 00 00 00 00 00 00   .....a..........
0050   00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
0060   00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
0070   00 00 00 00 00 00 00 00 00 00 00 00 00 00         ..............
```

- 0b 00: Тип+код - 11 0 - время истекло.
- f4 ff: Контрольная сумма.

- 00 00: уникальный ID сессии.
- 00 00: Номер последовательности.
- А дальше точная копия исходного запроса со 2го уровеня IPv4 (Frame 1) с изменениями:
  - TTL: Уменьшен с 0x01 на 0x00.
  - Контрольная сумма IP: Изменилась с 0x0000 на 0xde28 (т.к. изменилось TTL).

Полезная информация:

- Как работает определение следующего хопа?

  tracert полагается исключительно на увеличение TTL и ответы от промежуточных узлов.

  1) Отправка пакета с TTL=N+1. После получения ответа от хопа N, компьютер отправляет новый пакет с TTL=N+1 на тот же самый адрес назначения (например, biobiochile.cl).

  2) Автоматическая маршрутизация. Каждый следующий маршрутизатор в пути:
  - Уменьшает TTL на 1.
  - При TTL=0 возвращает Time Exceeded со своим IP.
  - При TTL>0 пересылает пакет дальше по своей таблице маршрутизации.
- Как работает TTL в трассировке?
  - Отправляются пакеты с последовательно увеличивающимся TTL (1, 2, 3...).
  - Промежуточные узлы отвечают «Time Exceeded» и показывают свой TTL (обычно 64 или 255).
  - Конечный сервер (biobiochile.cl) отвечает «Echo Reply» со своим исходным TTL.

4.3 Web-доступ

Результат выполнения и подробный вывод для 1-го пакета представлен на Рис. 5.

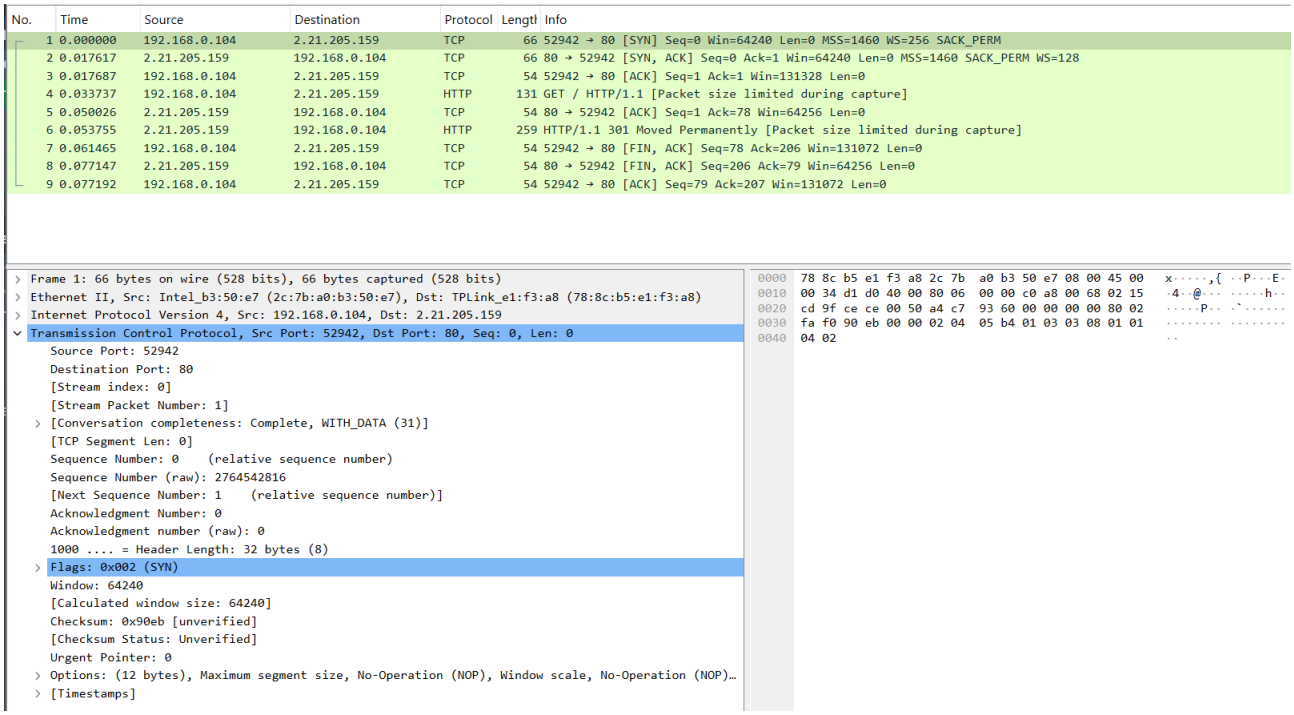

Рис. 5. Web

4.3.1 Устанавливаем TCP соединение (трехкратное рукопожатие)

Пакет 1: Клиент → Сервер ([SYN])
Источник: 192.168.0.104 (компьютер)
Назначение: 2.21.205.159 (сервер newzeland.com)
Протокол: TCP
Порт клиента: 52942 (случайный динамический порт)
Порт сервера: 80 (HTTP)
Флаги: [SYN] - запрос на установку соединения (синхронизация номера последовательности).
Sequence Number = 0 - начальный номер последовательности (относительный).
Sequence Number (raw): 2764542816
Acknowledgment Number: 0
Acknowledgment number (raw): 0
Win=64240 - размер окна приёма клиента.
MSS=1460 - максимальный размер сегмента.
WS=256 - Window Scaling (масштабирование окна).
SACK_PERM - поддержка Selective ACK (при потере пакета можно будет запросить конкретный, а не все, которые уже в том числе доставлены после потерянного).

Пакет 2: Сервер → Клиент ([SYN, ACK])
Флаги: [SYN, ACK] - сервер подтверждает соединение и отправляет свой SYN.
Seq=0 - начальный номер последовательности сервера.
Sequence Number (raw): 3242837979

Acknowledgment number (raw): 2764542817
Ack=1 - подтверждение получения SYN от клиента.
Win=64240 - размер окна сервера.
MSS=1460 - максимальный размер сегмента сервера.
WS = 128 - Window Scaling сервера.

Пакет 3: Клиент → Сервер ([ACK])
Флаги: [ACK] - подтверждение от клиента.
Seq=1 - следующий ожидаемый байт.
Sequence Number (raw): 2764542817
Ack=1 - подтверждение SYN сервера.
Acknowledgment number (raw): 3242837980

4.3.2 HTTP-запрос и ответ

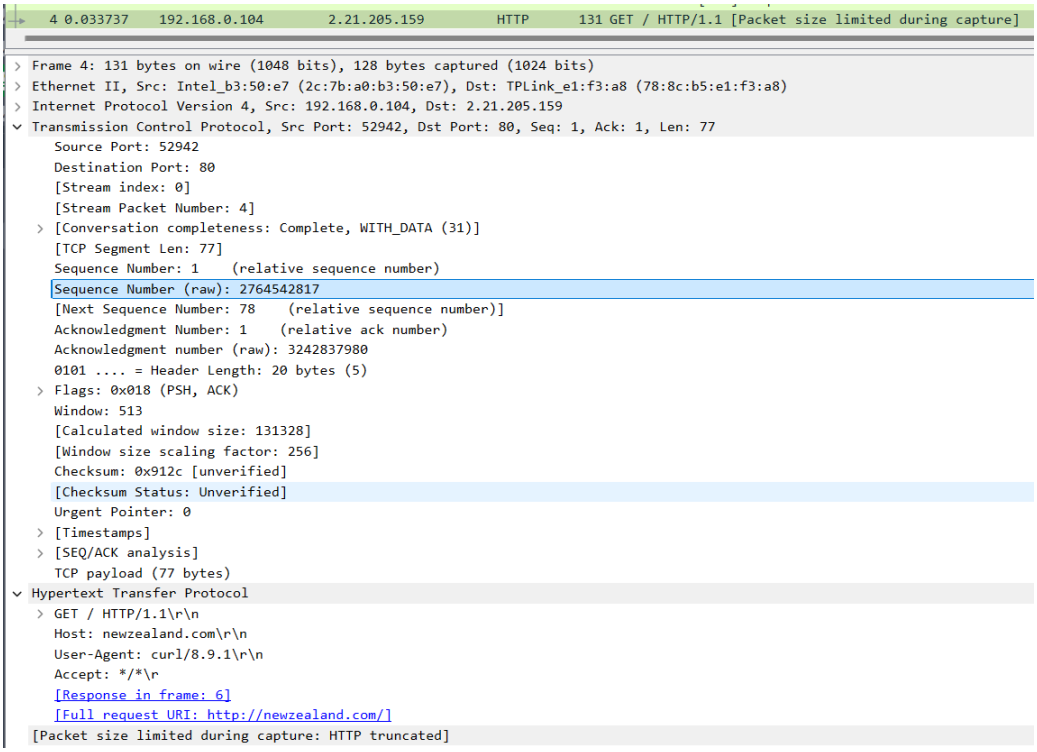

Рис. 6. Web: HTTP-запрос

Браузер запрашивает главную страницу сайта.
Пакет 4: Клиент → Сервер (GET / HTTP/1.1)
Протокол: HTTP
Метод: GET / - запрос главной страницы.
Версия: HTTP/1.1
TCP Segment Len: 77
Sequence Number (raw): 2764542817

[Next Sequence Number: 78   (relative sequence number)]
Acknowledgment number (raw): 3242837980
Длина: 131 байт (заголовки HTTP).

Сервер подтверждает получение запроса.
Пакет 5: Сервер → Клиент ([ACK])
Флаги: [ACK] - подтверждение получения HTTP-запроса.
Ack=78 - сервер подтвердил получение первых 77 байт (HTTP-заголовков).
Acknowledgment number (raw): 2764542894
Sequence Number (raw): 3242837980

Сервер перенаправляет на https адрес.
Пакет 6: Сервер → Клиент (HTTP/1.1 301 Moved Permanently)
Статус: 301 Moved Permanently - перенаправление.
Причина: сервер перенаправляет http://newzealand.com → https://newzealsnd.com.
Длина: 259 байт.
Sequence Number (raw): 3242837980
Acknowledgment Number: 78   (relative ack number)
Acknowledgment number (raw): 2764542894

4.3.3 Закрытие соединения

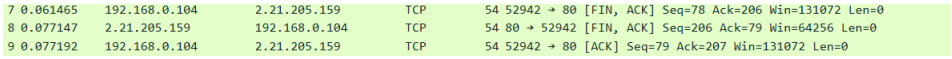

Рис. 7. Web: закрытие соединения

Пакет 7: Клиент → Сервер ([FIN, ACK])
Флаги: [FIN, ACK] - клиент инициирует завершение соединения.
Seq=78, Ack=206 - подтверждение получения HTTP-ответа.
Acknowledgment number (raw): 3242838185
Sequence Number (raw): 2764542894

Пакет 8: Сервер → Клиент ([FIN, ACK])
Флаги: [FIN, ACK] - сервер согласен на разрыв.
Seq=206, Ack=79 - подтверждение.
Acknowledgment number (raw): 2764542895
Sequence Number (raw): 3242838185

Пакет 9: Клиент → Сервер ([ACK])
Флаги: [ACK] - финальное подтверждение.
Ack=207 - подтверждение получения FIN от сервера.

```
Sequence Number (raw): 2764542895
Acknowledgment number (raw): 3242838186
```