

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение высшего  
образования



НИЖЕГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ  
УНИВЕРСИТЕТ им. Р.Е.АЛЕКСЕЕВА

Институт радиоэлектроники и информационных технологий  
Кафедра информатики и систем управления

## ОТЧЕТ

По лабораторной работе №1

РУКОВОДИТЕЛЬ:

\_\_\_\_\_  
(подпись)

Гай В.Е.  
(фамилия, и.,о.)

СТУДЕНТ:

\_\_\_\_\_  
(подпись)

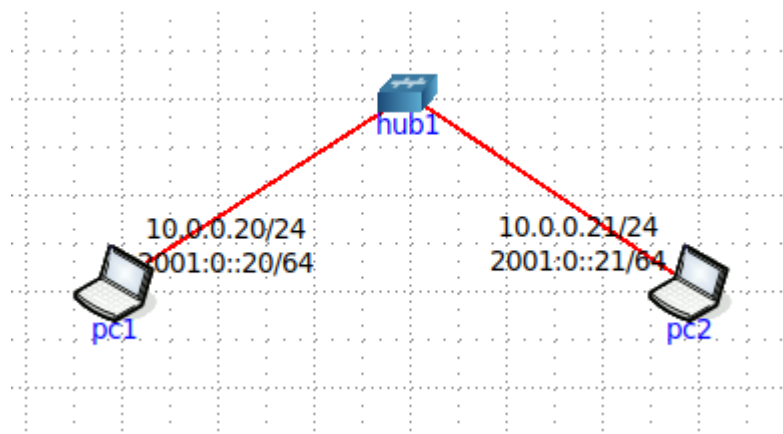
Захарова Е.Д.  
(фамилия, и.,о.)

18-B1  
(шифр группы)

Работа защищена «\_\_» \_\_\_\_\_

С оценкой \_\_\_\_\_

Нижний Новгород 2021



## Работа с анализатором протоколов tcpdump

1. Запустить tcpdump в режиме захвата всех пакетов, проходящих по сети. Количество захватываемых пакетов ограничить 10. Результаты протоколировать в файл.

```

Терминал
root@pc1:/tmp/pycore.37817/pc1.conf# tcpdump -c 10 -w out.cap
tcpdump: listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
10 packets captured
10 packets received by filter
0 packets dropped by kernel
root@pc1:/tmp/pycore.37817/pc1.conf#

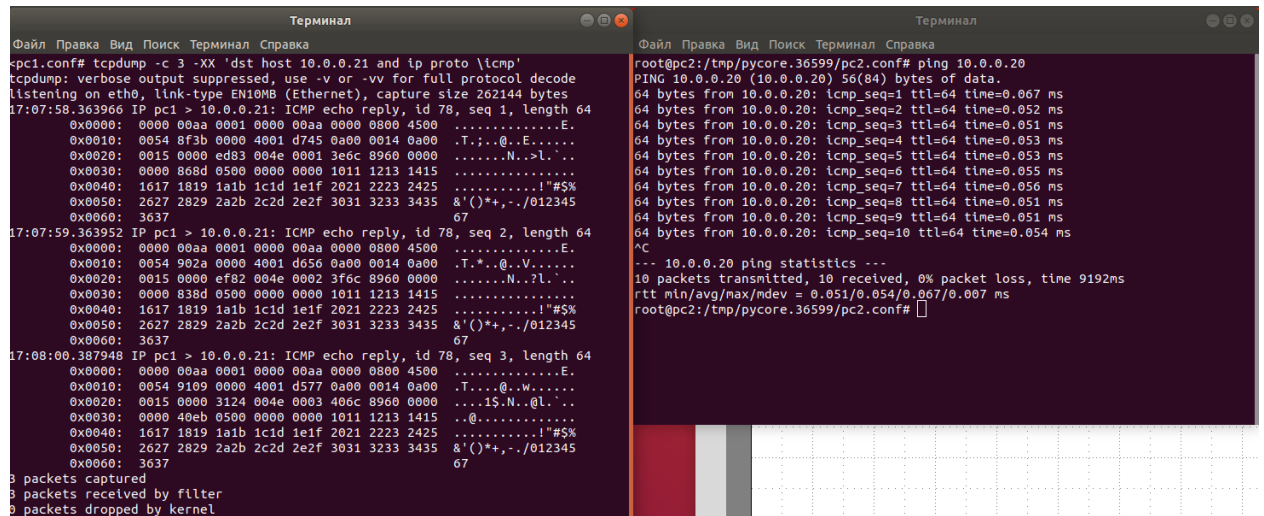
Терминал
root@pc2:/tmp/pycore.37817/pc2.conf# ping 10.0.0.20
PING 10.0.0.20 (10.0.0.20) 56(84) bytes of data:
64 bytes from 10.0.0.20: icmp_seq=1 ttl=64 time=0.044 ms
64 bytes from 10.0.0.20: icmp_seq=2 ttl=64 time=0.055 ms
64 bytes from 10.0.0.20: icmp_seq=3 ttl=64 time=0.053 ms
64 bytes from 10.0.0.20: icmp_seq=4 ttl=64 time=0.057 ms
64 bytes from 10.0.0.20: icmp_seq=5 ttl=64 time=0.058 ms
64 bytes from 10.0.0.20: icmp_seq=6 ttl=64 time=0.048 ms
64 bytes from 10.0.0.20: icmp_seq=7 ttl=64 time=0.047 ms
64 bytes from 10.0.0.20: icmp_seq=8 ttl=64 time=0.047 ms
64 bytes from 10.0.0.20: icmp_seq=9 ttl=64 time=0.049 ms
64 bytes from 10.0.0.20: icmp_seq=10 ttl=64 time=0.049 ms
64 bytes from 10.0.0.20: icmp_seq=11 ttl=64 time=0.049 ms
  
```

2. Запустить tcpdump в режиме перехвата широковещательного трафика (фильтр по MAC-адресу). Количество захватываемых пакетов ограничить 5. Включить распечатку пакета в шестнадцатеричной системе (включая заголовок канального уровня).

```

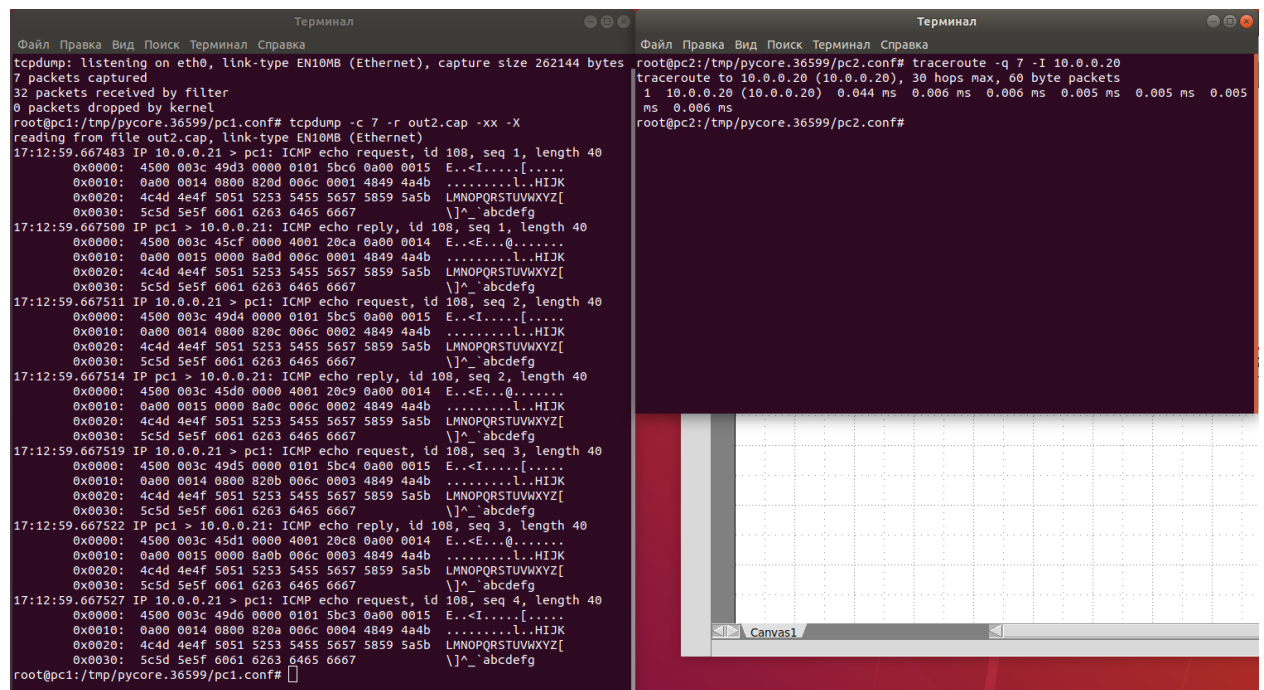
Терминал
Файл Правка Вид Поиск Терминал Справка
<pc1.conf# tcpdump -c 5 -x 'ether dst ff:ff:ff:ff:ff:ff'
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
15:46:00.221042 ARP, Request who-has pc1 tell 10.0.0.21, length 46
  0x0000:  0001 0800 0604 0001 0000 00aa 0001 0a00
  0x0010:  0015 0000 0000 0000 0a00 0014 0000 0000
  0x0020:  0000 0000 0000 0000 0000 0000 0000
15:49:02.614393 ARP, Request who-has pc1 (00:00:00:aa:00:00 (oui Ethernet)) tell 10.0.0.21, length 46
  0x0000:  0001 0800 0604 0001 0000 00aa 0001 0a00
  0x0010:  0015 0000 00aa 0000 0a00 0014 0000 0000
  0x0020:  0000 0000 0000 0000 0000 0000 0000
15:52:06.065315 ARP, Request who-has pc1 tell 10.0.0.21, length 46
  0x0000:  0001 0800 0604 0001 0000 00aa 0001 0a00
  0x0010:  0015 0000 0000 0000 0a00 0014 0000 0000
  0x0020:  0000 0000 0000 0000 0000 0000 0000
15:52:07.154874 ARP, Request who-has pc1 tell 10.0.0.21, length 46
  0x0000:  0001 0800 0604 0001 0000 00aa 0001 0a00
  0x0010:  0015 0000 0000 0000 0a00 0014 0000 0000
  0x0020:  0000 0000 0000 0000 0000 0000 0000
16:01:45.349472 ARP, Request who-has pc1 (Broadcast) tell 10.0.0.21, length 46
  0x0000:  0001 0800 0604 0001 0000 00aa 0001 0a00
  0x0010:  0015 ffff ffff ffff 0a00 0014 0000 0000
  0x0020:  0000 0000 0000 0000 0000 0000 0000
5 packets captured
5 packets received by filter
0 packets dropped by kernel
  
```

3. Запустить tcpdump так, чтобы он перехватывал только пакеты протокола ICMP, отправленные на определенный IP-адрес. При этом включить распечатку пакета в шестнадцатеричной системе и ASCII-формате (включая заголовок канального уровня). Количество захватываемых пакетов ограничить 3. Для генерирования пакетов воспользоваться утилитой ping.



The image shows two terminal windows. The left window displays the output of the command `tcpdump -c 3 -XX 'dst host 10.0.0.21 and ip proto icmp'`. It shows three captured ICMP echo replies from 10.0.0.21 to 10.0.0.20, each 64 bytes long. The output includes hex and ASCII representations of the packets. The right window shows the output of the command `ping 10.0.0.20`, displaying 10 successful pings with 0% packet loss and a round-trip time of approximately 0.05 ms.

4. Запустить tcpdump в режиме сохранения данных в двоичном режиме так, чтобы он перехватывал пакеты, созданные утилитой traceroute для определения маршрута к заданному в варианте узлу. Включить распечатку пакета в шестнадцатеричной системе и ASCII-формате (включая заголовок канального уровня). Количество захватываемых пакетов ограничить 7. Результат работы программы писать в файл.



The image shows two terminal windows. The left window displays the output of the command `tcpdump -c 7 -r out2.cap -XX -X`. It shows seven captured packets, including ICMP echo requests and replies, and traceroute packets. The output includes hex and ASCII representations of the packets. The right window shows the output of the command `traceroute -q 7 -I 10.0.0.20`, displaying the route from 10.0.0.20 to 10.0.0.20, showing 30 hops max and 60 byte packets.

5. Прочитать программой tcpdump созданный в предыдущем пункте файл.

```
root@pc1:/tmp/pycore.36599/pc1.conf# tcpdump -c 7 -r out2.cap
reading from file out2.cap, link-type EN10MB (Ethernet)
17:12:59.667483 IP 10.0.0.21 > pc1: ICMP echo request, id 108, seq 1, length 40
17:12:59.667500 IP pc1 > 10.0.0.21: ICMP echo reply, id 108, seq 1, length 40
17:12:59.667511 IP 10.0.0.21 > pc1: ICMP echo request, id 108, seq 2, length 40
17:12:59.667514 IP pc1 > 10.0.0.21: ICMP echo reply, id 108, seq 2, length 40
17:12:59.667519 IP 10.0.0.21 > pc1: ICMP echo request, id 108, seq 3, length 40
17:12:59.667522 IP pc1 > 10.0.0.21: ICMP echo reply, id 108, seq 3, length 40
17:12:59.667527 IP 10.0.0.21 > pc1: ICMP echo request, id 108, seq 4, length 40
```

6. Придумать три задания для фильтрации пакетов на основе протоколов ARP, TCP, UDP, ICMP

1) Запустить tcpdump так, чтобы он перехватывал только пакеты протоколов ICMP, отправленные на определенный IP-адрес, чтобы он перехватывал пакеты, созданные утилитой traceroute. Включить распечатку пакета в шестнадцатеричной системе и ASCII-формате (не включая заголовок канального уровня). Количество захватываемых пакетов ограничить 5.

The image shows two terminal windows side-by-side. The left window displays the output of a tcpdump command: `tcpdump -c 5 -X 'dst host 10.0.0.21 and ip proto \icmp'`. It shows five captured ICMP packets (echo requests and replies) between 10.0.0.21 and pc1. The output includes hex and ASCII representations of the packet data. At the bottom, it states '5 packets captured', '12 packets received by filter', and '4 packets dropped by kernel'. The right window shows the output of a traceroute command: `traceroute -q 7 -I 10.0.0.20`. It shows a single hop from 10.0.0.20 to 10.0.0.20 with various timing statistics. Below the terminal output, there is a network diagram showing a connection between two nodes, with one node labeled 'pc2'.

2) Запустить `tcpdump` так, чтобы он перехватывал только пакеты протокола UDP, отправленные на определенный IP-адрес, чтобы он перехватывал пакеты, созданные утилитой `traceroute`. Включить распечатку пакета в шестнадцатеричной системе (включая заголовок канального уровня). Количество захватываемых пакетов ограничить 4.

The image shows two terminal windows. The left window shows the execution of `tcpdump -c 4 -xx 'dst host 10.0.0.20 and ip proto {udp}'` on interface `eth0`. It displays four captured packets in hexadecimal and ASCII, all destined for 10.0.0.20 and using the UDP protocol. The right window shows the execution of `traceroute -q 4 10.0.0.20`, which shows a single hop from 10.0.0.20 to 10.0.0.20 with a time of 0.045 ms.

3) Запустить `tcpdump` так, чтобы он перехватывал только пакеты протокола ARP, отправленные на определенный IP-адрес, чтобы он перехватывал пакеты, созданные утилитой `ping`. Включить распечатку пакета в шестнадцатеричной системе (включая заголовок канального уровня). Количество захватываемых пакетов ограничить 4.

The image shows two terminal windows. The left window shows the execution of `tcpdump -c 4 -xx 'ether proto {arp}'` on interface `eth0`. It displays four captured ARP packets in hexadecimal and ASCII, including requests and replies. The right window shows the execution of `ping 10.0.0.20`, which shows 15 successful pings with a 0% packet loss and a round-trip time of approximately 0.044 ms.

## Работа с анализатором протоколов wireshark

1. Захватить 5-7 пакетов широковещательного трафика (фильтр по IP-адресу). Результат сохранить в текстовый файл.

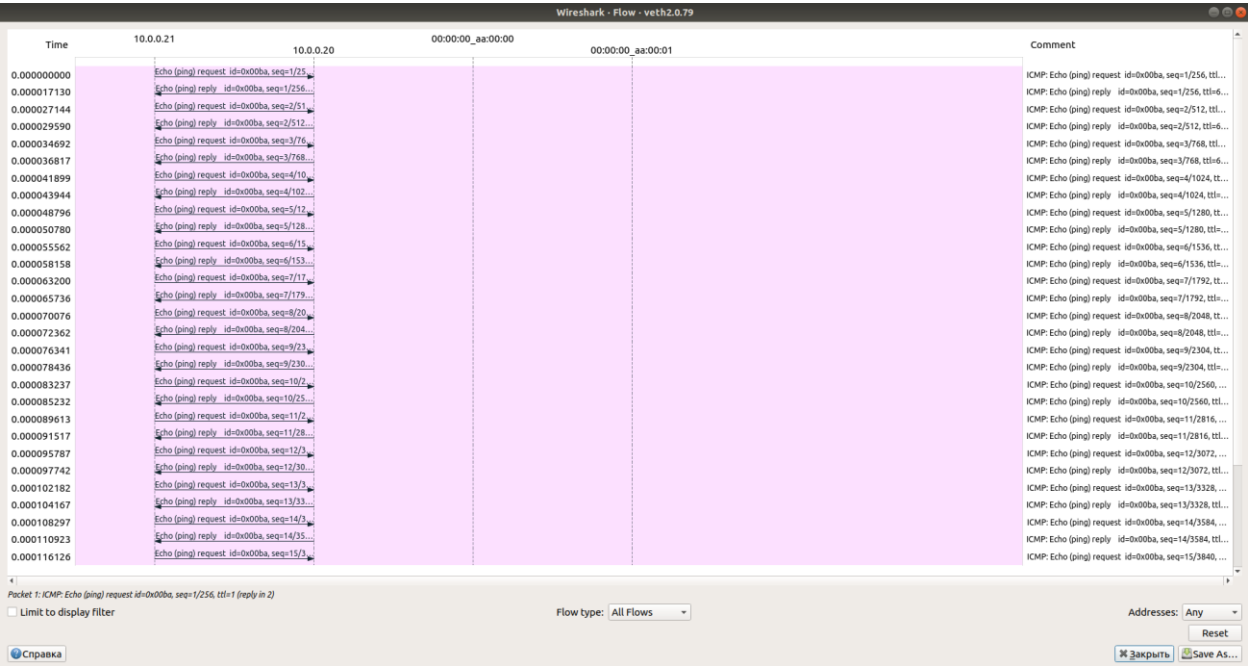
| ip.addr == 10.0.0.20 |             |           |             |          |        |   |
|----------------------|-------------|-----------|-------------|----------|--------|---|
| No.                  | Time        | Source    | Destination | Protocol | Length | Info  |
| 1                    | 0.000000000 | 10.0.0.21 | 10.0.0.20   | ICMP     | 98     | Echo (ping) request id=0x00ae, seq=1/256, ttl=64 (reply in 2) |
| 2                    | 0.000015256 | 10.0.0.20 | 10.0.0.21   | ICMP     | 98     | Echo (ping) reply id=0x00ae, seq=1/256, ttl=64 (request in 1) |
| 3                    | 1.009155710 | 10.0.0.21 | 10.0.0.20   | ICMP     | 98     | Echo (ping) request id=0x00ae, seq=2/512, ttl=64 (reply in 4) |
| 4                    | 1.009179186 | 10.0.0.20 | 10.0.0.21   | ICMP     | 98     | Echo (ping) reply id=0x00ae, seq=2/512, ttl=64 (request in 3) |
| 5                    | 2.033686289 | 10.0.0.21 | 10.0.0.20   | ICMP     | 98     | Echo (ping) request id=0x00ae, seq=3/768, ttl=64 (reply in 6) |
| 6                    | 2.033705986 | 10.0.0.20 | 10.0.0.21   | ICMP     | 98     | Echo (ping) reply id=0x00ae, seq=3/768, ttl=64 (request in 5) |



2. Захватить 3-4 пакета ICMP, полученных от определенного узла. Для генерирования пакетов воспользоваться утилитой ping. Результат сохранить в текстовый файл.

| icmp |             |           |             |          |        |   |
|------|-------------|-----------|-------------|----------|--------|---|
| No.  | Time        | Source    | Destination | Protocol | Length | Info  |
| 1    | 0.000000000 | 10.0.0.21 | 10.0.0.20   | ICMP     | 98     | Echo (ping) request id=0x00af, seq=1/256, ttl=64 (reply in 2) |
| 2    | 0.000019877 | 10.0.0.20 | 10.0.0.21   | ICMP     | 98     | Echo (ping) reply id=0x00af, seq=1/256, ttl=64 (request in 1) |
| 3    | 1.006420111 | 10.0.0.21 | 10.0.0.20   | ICMP     | 98     | Echo (ping) request id=0x00af, seq=2/512, ttl=64 (reply in 4) |
| 4    | 1.006436560 | 10.0.0.20 | 10.0.0.21   | ICMP     | 98     | Echo (ping) reply id=0x00af, seq=2/512, ttl=64 (request in 3) |

3. Перехватить пакеты, созданные утилитой traceroute для определения маршрута к заданному в варианте узлу. По результатам построить диаграмму Flow Graph. Диаграмму сохранить либо в виде текстового файла либо в виде изображения.



```
Терминал

Файл Правка Вид Поиск Терминал Справка

root@pc2:/tmp/pycore.36599/pc2.conf# traceroute -I 10.0.0.20
traceroute to 10.0.0.20 (10.0.0.20), 30 hops max, 60 byte packets
 1  10.0.0.20 (10.0.0.20)  0.043 ms  0.006 ms  0.005 ms
```

4. Прочитать файл, созданный программой tcpdump. Сравнить с тем, что было получено утилитой wireshark.

```
lab1.txt
~Документы
Открыть
Сохранить
Файл Правка Вид Поиск Терминал Справка

14:46:39.620.093  ETHER
+-----+
|0|00|00|00|aa|00|00|00|00|00|aa|00|01|08|00|45|00|00|54|6c|af|40|00|40|01|b5|d1|0a|00|00|15|
|0a|00|00|14|08|00|30|1b|00|c5|00|01|4f|75|89|60|00|00|00|26|76|09|00|00|00|00|10|11|12|13|
|14|15|16|17|18|19|1a|1b|1c|1d|1e|1f|20|21|22|23|24|25|26|27|28|29|2a|2b|2c|2d|2e|2f|30|31|32|33|
|34|35|36|37|
+-----+

14:46:39.620.113  ETHER
+-----+
|0|00|00|00|aa|00|01|00|00|00|aa|00|00|08|00|45|00|00|54|a6|08|00|00|40|01|c0|78|0a|00|00|14|
|0a|00|00|15|00|00|30|1b|00|c5|00|01|4f|75|89|60|00|00|00|26|76|09|00|00|00|00|10|11|12|13|
|14|15|16|17|18|19|1a|1b|1c|1d|1e|1f|20|21|22|23|24|25|26|27|28|29|2a|2b|2c|2d|2e|2f|30|31|32|33|
|34|35|36|37|
+-----+

14:46:40.643.939  ETHER
+-----+
|0|00|00|00|aa|00|00|00|00|00|aa|00|01|08|00|45|00|00|54|6d|56|40|00|40|01|b9|2a|0a|00|00|15|
|0a|00|00|14|08|00|15|bd|00|c5|00|02|50|75|89|60|00|00|00|3f|d3|09|00|00|00|00|10|11|12|13|
|14|15|16|17|18|19|1a|1b|1c|1d|1e|1f|20|21|22|23|24|25|26|27|28|29|2a|2b|2c|2d|2e|2f|30|31|32|33|
|34|35|36|37|
+-----+

14:46:40.643.962  ETHER
+-----+
|0|00|00|00|aa|00|01|00|00|00|aa|00|00|08|00|45|00|00|54|a6|26|00|00|40|01|c0|5a|0a|00|00|14|
|0a|00|00|15|00|00|1d|bd|00|c5|00|02|50|75|89|60|00|00|00|3f|d3|09|00|00|00|00|10|11|12|13|
|14|15|16|17|18|19|1a|1b|1c|1d|1e|1f|20|21|22|23|24|25|26|27|28|29|2a|2b|2c|2d|2e|2f|30|31|32|33|
|34|35|36|37|
+-----+

14:46:41.668.374  ETHER
+-----+
|0|00|00|00|aa|00|00|00|00|00|aa|00|01|08|00|45|00|00|54|6d|b5|40|00|40|01|b8|cb|0a|00|00|15|
|0a|00|00|14|08|00|a0|5c|00|c5|00|03|51|75|89|60|00|00|00|b3|32|0a|00|00|00|00|10|11|12|13|
|14|15|16|17|18|19|1a|1b|1c|1d|1e|1f|20|21|22|23|24|25|26|27|28|29|2a|2b|2c|2d|2e|2f|30|31|32|33|
|34|35|36|37|
+-----+

14:46:41.668.396  ETHER
+-----+
|0|00|00|00|aa|00|01|00|00|00|aa|00|00|08|00|45|00|00|54|a6|71|00|00|40|01|c0|0f|0a|00|00|14|
|0a|00|00|15|00|00|a0|5c|00|c5|00|03|51|75|89|60|00|00|00|b3|32|0a|00|00|00|00|10|11|12|13|
|14|15|16|17|18|19|1a|1b|1c|1d|1e|1f|20|21|22|23|24|25|26|27|28|29|2a|2b|2c|2d|2e|2f|30|31|32|33|
|34|35|36|37|
+-----+

14:46:42.691.983  ETHER
+-----+
|0|00|00|00|aa|00|00|00|00|00|aa|00|01|08|00|45|00|00|54|6e|8a|40|00|40|01|b7|f6|0a|00|00|15|
|0a|00|00|14|08|00|69|ff|00|c5|00|04|52|75|89|60|00|00|00|e8|0e|0a|00|00|00|00|10|11|12|13|
|14|15|16|17|18|19|1a|1b|1c|1d|1e|1f|20|21|22|23|24|25|26|27|28|29|2a|2b|2c|2d|2e|2f|30|31|32|33|
|34|35|36|37|
+-----+

root@pc1:/tmp/pycore.36599/pc1.conf# tcpdump -c 7 -r out2.cap -xx -X
reading from file out2.cap, link-type EN10MB (Ethernet)
17:46:39.620098 IP 10.0.0.21 > pc1: ICMP echo request, id 197, seq 1, length 64
  0x0000: 4500 0054 6caf 4000 4001 b9d1 0a00 0015  E...TL.@.X....
  0x0010: 0a00 0014 0800 301b 00c5 0001 4f75 8960  .....8.....Ou.
  0x0020: 0000 0000 2676 0900 0000 0000 1011 1213  .....&v.....
  0x0030: 1415 1617 1819 1a1b 1c1d 1e1f 2021 2223  .....!"#
  0x0040: 2425 2627 2829 2a2b 2c2d 2e2f 3031 3233  S88()*+,-./0123
  0x0050: 3435 3637                                     4567
17:46:39.620112 IP pc1 > 10.0.0.21: ICMP echo reply, id 197, seq 1, length 64
  0x0000: 4500 0054 a608 0000 4001 c078 0a00 0014  E...T.@.X....
  0x0010: 0a00 0015 0000 301b 00c5 0001 4f75 8960  .....8.....Ou.
  0x0020: 0000 0000 2676 0900 0000 0000 1011 1213  .....&v.....
  0x0030: 1415 1617 1819 1a1b 1c1d 1e1f 2021 2223  .....!"#
  0x0040: 2425 2627 2829 2a2b 2c2d 2e2f 3031 3233  S88()*+,-./0123
  0x0050: 3435 3637                                     4567
17:46:40.643945 IP 10.0.0.21 > pc1: ICMP echo request, id 197, seq 2, length 64
  0x0000: 4500 0054 6d56 4000 4001 b92a 0a00 0015  E..Tm@.@.X....
  0x0010: 0a00 0014 0800 15bd 00c5 0002 5075 8960  .....7.....Pu.
  0x0020: 0000 0000 3fd3 0900 0000 0000 1011 1213  .....?.....
  0x0030: 1415 1617 1819 1a1b 1c1d 1e1f 2021 2223  .....!"#
  0x0040: 2425 2627 2829 2a2b 2c2d 2e2f 3031 3233  S88()*+,-./0123
  0x0050: 3435 3637                                     4567
17:46:40.643961 IP pc1 > 10.0.0.21: ICMP echo reply, id 197, seq 2, length 64
  0x0000: 4500 0054 a626 0000 4001 c05a 0a00 0014  E..Tm@.@.X....
  0x0010: 0a00 0015 0000 1dbd 00c5 0002 5075 8960  .....7.....Pu.
  0x0020: 0000 0000 3fd3 0900 0000 0000 1011 1213  .....?.....
  0x0030: 1415 1617 1819 1a1b 1c1d 1e1f 2021 2223  .....!"#
  0x0040: 2425 2627 2829 2a2b 2c2d 2e2f 3031 3233  S88()*+,-./0123
  0x0050: 3435 3637                                     4567
17:46:41.668379 IP 10.0.0.21 > pc1: ICMP echo request, id 197, seq 3, length 64
  0x0000: 4500 0054 6db5 4000 4001 b8cb 0a00 0015  E..Tm.@.@.X....
  0x0010: 0a00 0014 0800 a05c 00c5 0003 5175 8960  .....2.....Qu.
  0x0020: 0000 0000 b332 0a00 0000 0000 1011 1213  .....2.....
  0x0030: 1415 1617 1819 1a1b 1c1d 1e1f 2021 2223  .....!"#
  0x0040: 2425 2627 2829 2a2b 2c2d 2e2f 3031 3233  S88()*+,-./0123
  0x0050: 3435 3637                                     4567
17:46:41.668395 IP pc1 > 10.0.0.21: ICMP echo reply, id 197, seq 3, length 64
  0x0000: 4500 0054 a671 0000 4001 c00f 0a00 0014  E..T.q.@.@.X....
  0x0010: 0a00 0015 0000 a85c 00c5 0003 5175 8960  .....2.....Qu.
  0x0020: 0000 0000 b332 0a00 0000 0000 1011 1213  .....2.....
  0x0030: 1415 1617 1819 1a1b 1c1d 1e1f 2021 2223  .....!"#
  0x0040: 2425 2627 2829 2a2b 2c2d 2e2f 3031 3233  S88()*+,-./0123
  0x0050: 3435 3637                                     4567
17:46:42.691989 IP 10.0.0.21 > pc1: ICMP echo request, id 197, seq 4, length 64
  0x0000: 4500 0054 6e8a 4000 4001 b7f6 0a00 0015  E..Tm.@.@.X....
  0x0010: 0a00 0014 0800 69ff 00c5 0004 5275 8960  .....l.....Ru.
  0x0020: 0000 0000 e80e 0a00 0000 0000 1011 1213  .....2.....
  0x0030: 1415 1617 1819 1a1b 1c1d 1e1f 2021 2223  .....!"#
  0x0040: 2425 2627 2829 2a2b 2c2d 2e2f 3031 3233  S88()*+,-./0123
  0x0050: 3435 3637                                     4567
```