

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
«сети и телекоммуникации»

РУКОВОДИТЕЛЬ:

(подпись)

Гай В.Е.
(фамилия, и.,о.)

СТУДЕНТ:

(подпись)

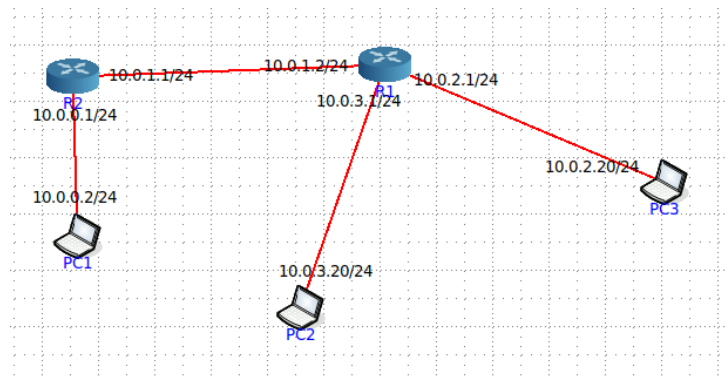
Рукавишников М.А..
(фамилия, и.,о.)

18-АС
(шифр группы)

Работа защищена «__» _____

С оценкой _____

Нижний Новгород 2021



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

```

Файл Правка Вид Поиск Терминал Справка
root@PC1:/tmp/pycore.43971/PC1.conf# tcpdump -l -c 10
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
15:59:17.664624 IP6 fe80::200:ff:feaa:2 > ff02::5: OSPFv3, Hello, length 36
15:59:17.754318 IP_gateway > 224.0.0.5: OSPFv2, Hello, length 44
15:59:19.760265 IP_gateway > 224.0.0.5: OSPFv2, Hello, length 44
15:59:21.800637 IP_gateway > 224.0.0.5: OSPFv2, Hello, length 44
15:59:23.801780 IP_gateway > 224.0.0.5: OSPFv2, Hello, length 44
15:59:24.072183 IP6 fe80::b821:43ff:febd:2bdd > ip6-allrouters: ICMP6, router so
licitation, length 16
15:59:25.803036 IP_gateway > 224.0.0.5: OSPFv2, Hello, length 44
15:59:26.375032 IP6 PC1 > ip6-allrouters: ICMP6, router solicitation, length 16
15:59:27.664480 IP6 fe80::200:ff:feaa:2 > ff02::5: OSPFv3, Hello, length 36
15:59:27.813073 IP_gateway > 224.0.0.5: OSPFv2, Hello, length 44
10 packets captured
10 packets received by filter
0 packets dropped by kernel
root@PC1:/tmp/pycore.43971/PC1.conf#

```

Запись в файл и чтение из него:

```

Терминал
Файл Правка Вид Поиск Терминал Справка
root@PC2:/tmp/pycore.43971/PC2.conf# ping 10.0.0.2
PING 10.0.0.2 (10.0.0.2) 56(84) bytes of data.
64 bytes from 10.0.0.2: icmp_seq=1 ttl=62 time=0.144 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=62 time=0.145 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=62 time=0.200 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=62 time=0.130 ms
64 bytes from 10.0.0.2: icmp_seq=5 ttl=62 time=0.159 ms
64 bytes from 10.0.0.2: icmp_seq=6 ttl=62 time=0.138 ms
64 bytes from 10.0.0.2: icmp_seq=7 ttl=62 time=0.134 ms
64 bytes from 10.0.0.2: icmp_seq=8 ttl=62 time=0.099 ms
64 bytes from 10.0.0.2: icmp_seq=9 ttl=62 time=0.130 ms
64 bytes from 10.0.0.2: icmp_seq=10 ttl=62 time=0.307 ms
64 bytes from 10.0.0.2: icmp_seq=11 ttl=62 time=0.137 ms
^C
--- 10.0.0.2 ping statistics ---
11 packets transmitted, 11 received, 0% packet loss, time 10384ms
rtt min/avg/max/mdev = 0.099/0.156/0.307/0.054 ms
root@PC2:/tmp/pycore.43971/PC2.conf#

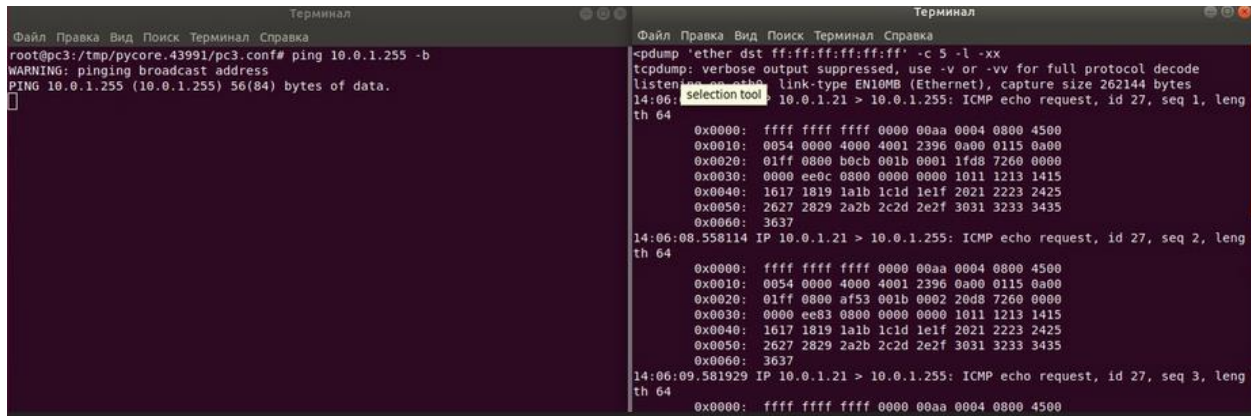
```

```

root@PC1:/tmp/pycore.43971/PC1.conf# tcpdump -l -c 10 -w logfile.log
tcpdump: listening on eth0, link-type EN10MB (Ethernet), capture size 262144 byt
es
10 packets captured
11 packets received by filter
0 packets dropped by kernel
root@PC1:/tmp/pycore.43971/PC1.conf# tcpdump -l -c 10 -r logfile.log
reading from file logfile.log, link-type EN10MB (Ethernet)
16:03:56.604902 IP_gateway > 224.0.0.5: OSPFv2, Hello, length 44
16:03:57.994840 IP 10.0.3.20 > PC1: ICMP echo request, id 28, seq 1, length 64
16:03:57.994849 IP PC1 > 10.0.3.20: ICMP echo reply, id 28, seq 1, length 64
16:03:58.204244 IP6 fe80::200:ff:feaa:2 > ff02::5: OSPFv3, Hello, length 36
16:03:58.707391 IP_gateway > 224.0.0.5: OSPFv2, Hello, length 44
16:03:58.763187 IP6 fe80::c4cf:d1ff:fe0a:8598 > ip6-allrouters: ICMP6, router so
licitation, length 16
16:03:59.016219 IP 10.0.3.20 > PC1: ICMP echo request, id 28, seq 2, length 64
16:03:59.016231 IP PC1 > 10.0.3.20: ICMP echo reply, id 28, seq 2, length 64
16:04:00.050522 IP 10.0.3.20 > PC1: ICMP echo request, id 28, seq 3, length 64
16:04:00.050547 IP PC1 > 10.0.3.20: ICMP echo reply, id 28, seq 3, length 64
root@PC1:/tmp/pycore.43971/PC1.conf#

```

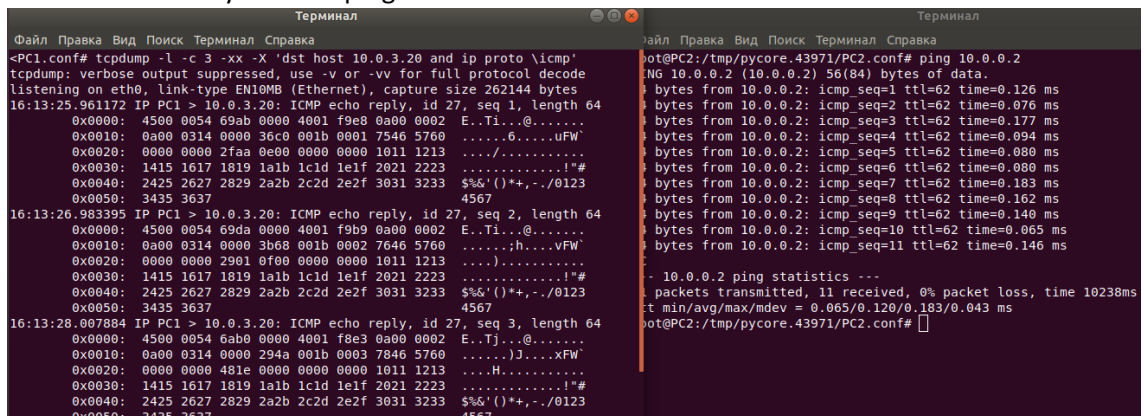
2. Запустить tcpdump в режиме перехвата широковещательного трафика (фильтр по MAC-адресу). Количество захватываемых пакетов ограничить 5. Включить распечатку пакета в шестнадцатеричной системе (включая заголовок канального уровня).



```
root@pc3:/tmp/pycore.43991/pc3.conf# ping 10.0.1.255 -b
WARNING: pinging broadcast address
PING 10.0.1.255 (10.0.1.255) 56(84) bytes of data.
```

```
tcpdump 'ether dst ff:ff:ff:ff:ff:ff' -c 5 -l -xx
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
14:06:08.558114 IP 10.0.1.21 > 10.0.1.255: ICMP echo request, id 27, seq 1, length 64
0x0000: ffff ffff ffff 0000 00aa 0004 0800 4500
0x0010: 0054 0000 4000 4001 2396 0a00 0115 0a00
0x0020: 01ff 0800 b0cb 001b 0001 1fd8 7260 0000
0x0030: 0000 e0c0 0800 0000 0000 1011 1213 1415
0x0040: 1617 1819 1a1b 1c1d 1e1f 2021 2223 2425
0x0050: 2627 2829 2a2b 2c2d 2e2f 3031 3233 3435
0x0060: 3637
14:06:08.558114 IP 10.0.1.21 > 10.0.1.255: ICMP echo request, id 27, seq 2, length 64
0x0000: ffff ffff ffff 0000 00aa 0004 0800 4500
0x0010: 0054 0000 4000 4001 2396 0a00 0115 0a00
0x0020: 01ff 0800 af53 001b 0002 20d8 7260 0000
0x0030: 0000 ee83 0800 0000 0000 1011 1213 1415
0x0040: 1617 1819 1a1b 1c1d 1e1f 2021 2223 2425
0x0050: 2627 2829 2a2b 2c2d 2e2f 3031 3233 3435
0x0060: 3637
14:06:09.581929 IP 10.0.1.21 > 10.0.1.255: ICMP echo request, id 27, seq 3, length 64
0x0000: ffff ffff ffff 0000 00aa 0004 0800 4500
```

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



```
<PC1.conf# tcpdump -l -c 3 -xx -X 'dst host 10.0.3.20 and ip proto \icmp'
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
16:13:25.961172 IP PC1 > 10.0.3.20: ICMP echo reply, id 27, seq 1, length 64
0x0000: 4500 0054 69ab 0000 4001 f9e8 0a00 0002 E..Ti...@.....
0x0010: 0a00 0314 0000 36c0 001b 0001 7546 5760 .....6....uFW'
0x0020: 0000 0000 2faa 0e00 0000 0000 1011 1213 ...../.....
0x0030: 1415 1617 1819 1a1b 1c1d 1e1f 2021 2223 .....!..H.....
0x0040: 2425 2627 2829 2a2b 2c2d 2e2f 3031 3233 %&()*+,-./0123
0x0050: 3435 3637 4567
16:13:26.983395 IP PC1 > 10.0.3.20: ICMP echo reply, id 27, seq 2, length 64
0x0000: 4500 0054 69da 0000 4001 f9b9 0a00 0002 E..Ti...@.....
0x0010: 0a00 0314 0000 36b8 001b 0002 7046 5760 .....;h....VFW'
0x0020: 0000 0000 2901 0f00 0000 0000 1011 1213 ...../.....
0x0030: 1415 1617 1819 1a1b 1c1d 1e1f 2021 2223 .....!..H.....
0x0040: 2425 2627 2829 2a2b 2c2d 2e2f 3031 3233 %&()*+,-./0123
0x0050: 3435 3637 4567
16:13:28.007884 IP PC1 > 10.0.3.20: ICMP echo reply, id 27, seq 3, length 64
0x0000: 4500 0054 6a0b 0000 4001 f8e3 0a00 0002 E..Ti...@.....
0x0010: 0a00 0314 0000 294a 001b 0003 7846 5760 .....J....XFW'
0x0020: 0000 0000 481e 0000 0000 0000 1011 1213 ...../.....
0x0030: 1415 1617 1819 1a1b 1c1d 1e1f 2021 2223 .....!..H.....
0x0040: 2425 2627 2829 2a2b 2c2d 2e2f 3031 3233 %&()*+,-./0123
0x0050: 3435 3637 4567
```

```
root@PC2:/tmp/pycore.43971/PC2.conf# ping 10.0.0.2
PING 10.0.0.2 (10.0.0.2) 56(84) bytes of data:
0 bytes from 10.0.0.2: icmp: seq=1 ttl=62 time=0.126 ms
0 bytes from 10.0.0.2: icmp: seq=2 ttl=62 time=0.076 ms
0 bytes from 10.0.0.2: icmp: seq=3 ttl=62 time=0.177 ms
0 bytes from 10.0.0.2: icmp: seq=4 ttl=62 time=0.094 ms
0 bytes from 10.0.0.2: icmp: seq=5 ttl=62 time=0.080 ms
0 bytes from 10.0.0.2: icmp: seq=6 ttl=62 time=0.080 ms
0 bytes from 10.0.0.2: icmp: seq=7 ttl=62 time=0.183 ms
0 bytes from 10.0.0.2: icmp: seq=8 ttl=62 time=0.162 ms
0 bytes from 10.0.0.2: icmp: seq=9 ttl=62 time=0.140 ms
0 bytes from 10.0.0.2: icmp: seq=10 ttl=62 time=0.065 ms
0 bytes from 10.0.0.2: icmp: seq=11 ttl=62 time=0.146 ms
--- 10.0.0.2 ping statistics ---
11 packets transmitted, 11 received, 0% packet loss, time 10238ms
rt min/avg/max/mdev = 0.065/0.120/0.183/0.043 ms
root@PC2:/tmp/pycore.43971/PC2.conf#
```

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.

```

root@PC2:/tmp/pycore.37425/PC2.conf# tcpdump -c 7 -l -w log2.cap
tcpdump: listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
7 packets captured
7 packets received by filter
0 packets dropped by kernel
root@PC2:/tmp/pycore.37425/PC2.conf# tcpdump -c 7 -l -xx -X -r log2.cap
reading from file log2.cap, link-type EN10MB (Ethernet)
14:13:42.628927 IP6 fe80::200:ff:feaa:2 > ff02::5: OSPFv3, Hello, length 36
    0x0000:  6c08 e62e 0024 5901 fe80 0000 0000 0000  l....$Y.....
    0x0010:  0200 00ff feaa 0002 ff02 0000 0000 0000  .....
    0x0020:  0000 0000 0000 0005 0301 0024 0a00 0001  .....$.
    0x0030:  0000 0000 a5c3 4100 0000 001e 0100 0113  .....A.....
    0x0040:  000a 0028 0a00 0001 0000 0000          ...(.
14:13:52.673454 IP6 fe80::200:ff:feaa:2 > ff02::5: OSPFv3, Hello, length 36
    0x0000:  6c08 e62e 0024 5901 fe80 0000 0000 0000  l....$Y.....
    0x0010:  0200 00ff feaa 0002 ff02 0000 0000 0000  .....
    0x0020:  0000 0000 0000 0005 0301 0024 0a00 0001  .....$.
    0x0030:  0000 0000 a5c3 4100 0000 001e 0100 0113  .....A.....
    0x0040:  000a 0028 0a00 0001 0000 0000          ...(.
14:13:55.178203 ARP, Request who-has PC2 tell 10.0.1.21, length 28
    0x0000:  0001 0800 0604 0001 0000 00aa 0004 0a00  .....
    0x0010:  0115 0000 0000 0000 0a00 0114          .....
14:13:55.178220 ARP, Reply PC2 is-at 00:00:00:aa:00:03 (oui Ethernet), length 28
    0x0000:  0001 0800 0604 0002 0000 00aa 0003 0a00  .....
    0x0010:  0114 0000 00aa 0004 0a00 0115          .....
14:14:02.695534 IP6 fe80::200:ff:feaa:2 > ff02::5: OSPFv3, Hello, length 36
    0x0000:  6c08 e62e 0024 5901 fe80 0000 0000 0000  l....$Y.....
    0x0010:  0200 00ff feaa 0002 ff02 0000 0000 0000  .....
    0x0020:  0000 0000 0000 0005 0301 0024 0a00 0001  .....$.
    0x0030:  0000 0000 a5c3 4100 0000 001e 0100 0113  .....A.....

```

5. Прочитать программой tcpdump созданный в предыдущем пункте файл.

```

root@pc1:/tmp/pycore.32993/pc1.conf# tcpdump -c 7 -r task.cap
reading from file task.cap, link-type EN10MB (Ethernet)
22:31:10.529321 IP 10.0.3.20 > pc1: ICMP echo request, id 75, seq 1, length 64
22:31:10.529328 IP pc1 > 10.0.3.20: ICMP echo reply, id 75, seq 1, length 64
22:31:11.557286 IP 10.0.3.20 > pc1: ICMP echo request, id 75, seq 2, length 64
22:31:11.557296 IP pc1 > 10.0.3.20: ICMP echo reply, id 75, seq 2, length 64
22:31:12.585327 IP 10.0.3.20 > pc1: ICMP echo request, id 75, seq 3, length 64
22:31:12.585356 IP pc1 > 10.0.3.20: ICMP echo reply, id 75, seq 3, length 64
22:31:13.610534 IP 10.0.3.20 > pc1: ICMP echo request, id 75, seq 4, length 64
root@pc1:/tmp/pycore.32993/pc1.conf#

```

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

- Перехват всех широковещательных ARP пакетов без отображения времени в каждой строке: `tcpdump -l -t 'ether proto \arp and ether dst ff:ff:ff:ff:ff'`

- Отправка пакетов через TCP с фильтром на прием UDP: (ничего не поймано)

```

root@pc1:/tmp/pycore.35267/pc1.conf# tcpdump -c 5 'ip proto \udp'
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
^C
0 packets captured
0 packets received by filter
0 packets dropped by kernel
root@pc1:/tmp/pycore.35267/pc1.conf# tcpdump -c 5 'ip proto \udp'

root@pc4:/tmp/pycore.35267/pc4.conf# traceroute -q 7 -T 10.0.5.20
traceroute to 10.0.5.20 (10.0.5.20), 30 hops max, 60 byte packets
 1  _gateway (10.0.9.1)  0.170 ms  0.146 ms  0.141 ms  0.137 ms  0.134 ms  0.130 ms
 2  10.0.7.1 (10.0.7.1)  0.123 ms  0.109 ms  0.104 ms  0.099 ms  0.095 ms  0.090 ms
 3  10.0.5.20 (10.0.5.20)  0.081 ms  0.057 ms  0.050 ms  0.035 ms  0.029 ms  0.021 ms
root@pc4:/tmp/pycore.35267/pc4.conf#

```

- Перехват UDP пакетов без протокола IPv6 с отображением их в ASCII и hex формате без заголовков канального уровня: `tcpdump 'ether proto \udp' -l -X`

Работа с анализатором протоколов wireshark

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

PC1:

socat - UDP-DATAGRAM:10.0.3.20:80,broadcast

"test"

PC2:

sudo -s

ufw allow 80/udp

ufw reload

sudo ufw enable

ufw status

wireshark -> UDP

| No. | Time | Source | Destination | Protocol | Length | Info |
|-----|--------------|-----------|-------------|----------|--------|---------------|
| 4 | 4.828223176 | 10.0.0.21 | 10.0.0.255 | UDP | 60 | 80 → 80 Len=0 |
| 10 | 13.158303050 | 10.0.0.21 | 10.0.0.255 | UDP | 60 | 80 → 80 Len=0 |
| 11 | 13.791550208 | 10.0.0.21 | 10.0.0.255 | UDP | 60 | 80 → 80 Len=0 |
| 12 | 13.992650178 | 10.0.0.21 | 10.0.0.255 | UDP | 60 | 80 → 80 Len=0 |
| 14 | 14.172875048 | 10.0.0.21 | 10.0.0.255 | UDP | 60 | 80 → 80 Len=0 |

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

The terminal window shows the execution of a ping command from 10.0.2.2 to 10.0.2.2. The output indicates 5 packets transmitted, 5 received, 0% packet loss, and a time of 4103ms. The Wireshark window shows a capture of 10 packets, including 4 ICMP Echo (ping) requests and 6 ICMP Echo (ping) replies, all from 10.0.0.2 to 10.0.0.2.

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:

The terminal window shows the execution of a traceroute command from 10.0.0.1 to 10.0.2.2. The output shows 3 hops: 1 gateway (10.0.0.1), 2 10.0.1.2 (10.0.1.2), and 3 10.0.2.2 (10.0.2.2). The Wireshark window shows a capture of 12 packets, including 4 UDP packets and 8 ICMP packets, all from 10.0.0.1 to 10.0.2.2.

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

Файл:

| No. | Time | Source | Destination | Protocol | Length | Info |
|-----|----------|-----------|-------------|----------|--------|--|
| 1 | 0.000000 | 10.8.0.28 | 10.8.0.29 | ICMP | 74 | 8475d -> 33437 Len=32 |
| 2 | 0.000012 | 10.8.0.28 | 10.8.0.29 | ICMP | 162 | Destination unreachable (Port unreachable) |
| 3 | 0.000041 | 10.8.0.28 | 10.8.0.29 | ICMP | 74 | 42438 -- 33438 Len=32 |
| 4 | 0.000046 | 10.8.0.28 | 10.8.0.29 | ICMP | 162 | Destination unreachable (Port unreachable) |
| 5 | 0.000071 | 10.8.0.28 | 10.8.0.29 | ICMP | 74 | 49051 -- 33438 Len=32 |
| 6 | 0.000075 | 10.8.0.28 | 10.8.0.29 | ICMP | 162 | Destination unreachable (Port unreachable) |
| 7 | 0.000381 | 10.8.0.28 | 10.8.0.29 | ICMP | 74 | 83478 -- 33448 Len=32 |

FlowGraph:

