Минобрнауки России
Федеральное государственное бюджетное образовательное
учреждение высшего образования
НИЖЕГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
ИМ. Р.Е. АЛЕКСЕЕВА
ИНСТИТУТ РАДИОЭЛЕКТРОНИКИ И ИНФОРМАЦИОННЫХ
ТЕХНОЛОГИЙ

Сети и телекоммуникации

Отчет по лабораторной работе №1

Выполнил: Гора К.А.
Проверил: Гай В.Е.

Нижний Новгород 2021

Задание:

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.

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).

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.

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.

5. Прочесть программой tcpdump созданный в предыдущем пункте файл.

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

Работа с анализатором протоколов wireshark

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

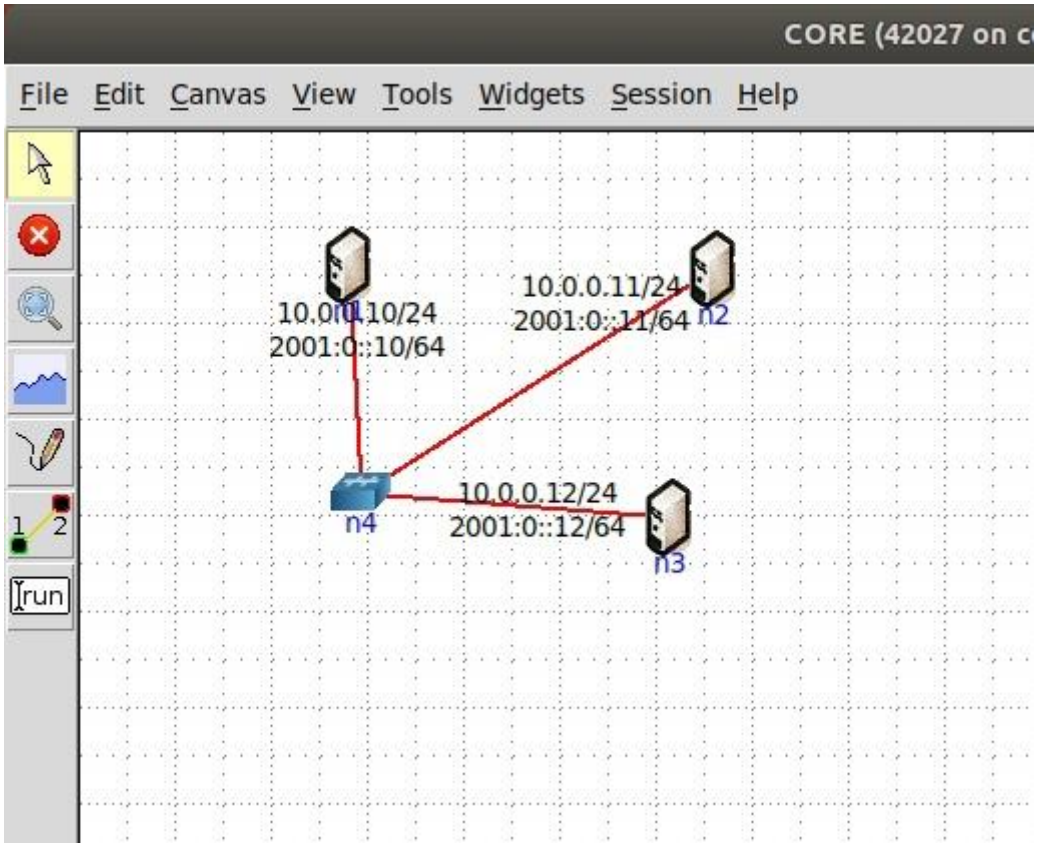
2. Захватить 3-4 пакета ICMP, полученных от определенного узла. Для генерирования пакетов воспользоваться утилитой ping. Результат сохранить в текстовый файл.

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.

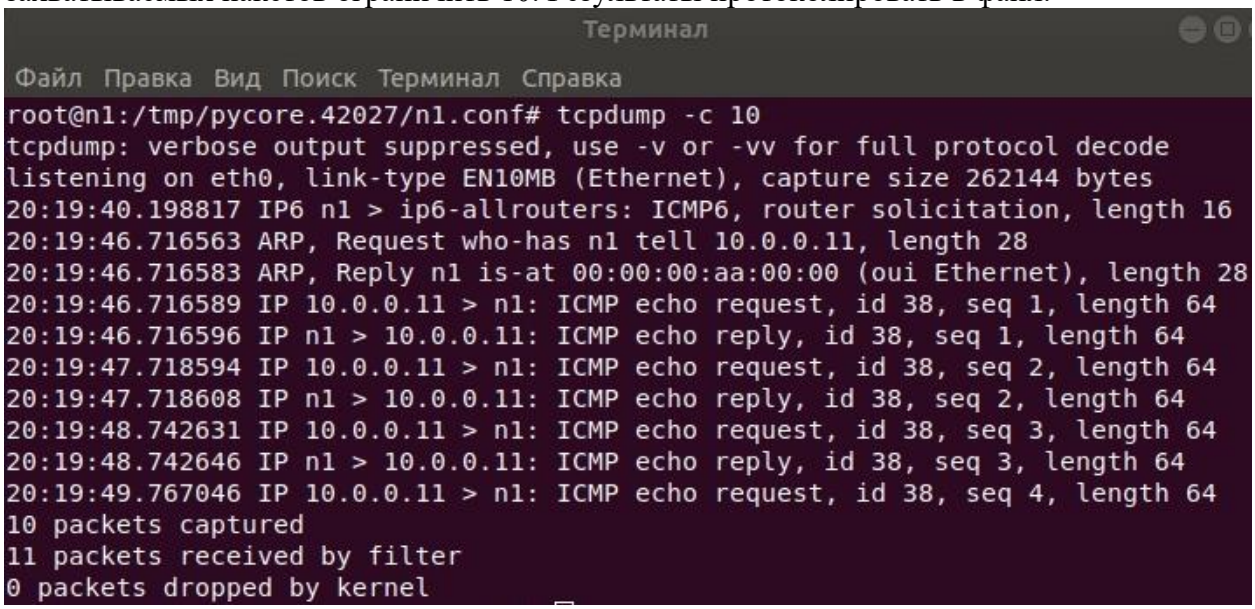
4. Прочесть файл, созданный программой tcpdump. Сравнить с тем, что было получено утилитой wireshark.

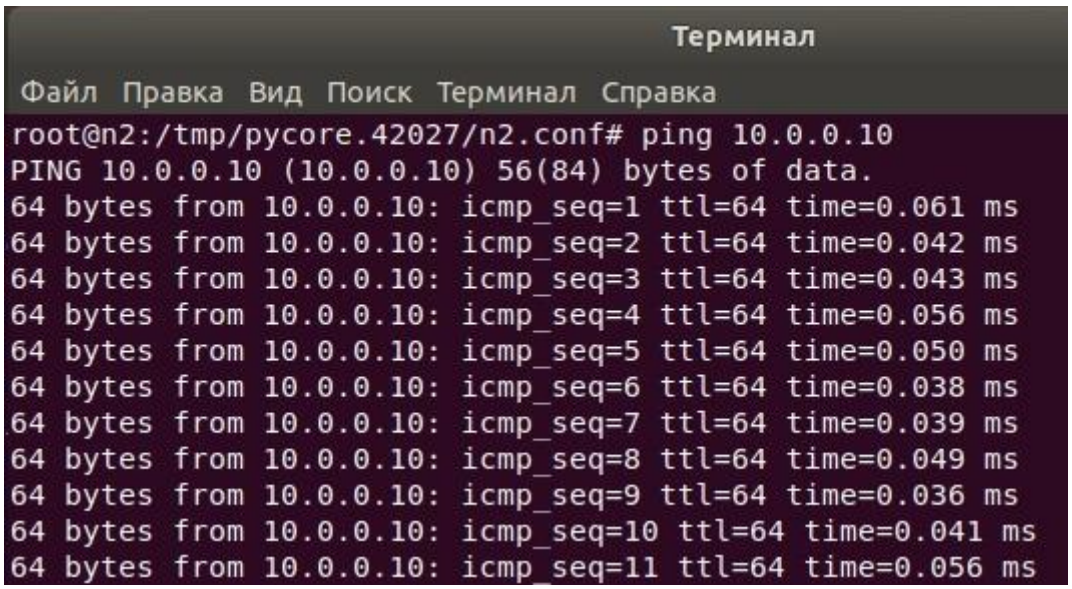
Ход работы:

Схема



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





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@n1:/tmp/pycore.42027/n1.conf# tcpdump -c 5 -e -xx
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
21:43:36.950467 00:00:00:aa:00:01 (oui Ethernet) > 00:00:00:aa:00:00 (oui Ethernet), ethertype IPv4 (0x0800), length 98: 10.0.0.11 > n1: ICMP echo request, id 39, seq 1, length 64
        0x0000:  0000 00aa 0000 0000 00aa 0001 0800 4500
        0x0010:  0054 fc70 4000 4001 2a24 0a00 000b 0a00
        0x0020:  000a 0800 d96d 0027 0001 58b6 4b60 0000
        0x0030:  0000 ad80 0e00 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
21:43:36.950481 00:00:00:aa:00:00 (oui Ethernet) > 00:00:00:aa:00:01 (oui Ethernet), ethertype IPv4 (0x0800), length 98: n1 > 10.0.0.11: ICMP echo reply, id 39, seq 1, length 64
        0x0000:  0000 00aa 0001 0000 00aa 0000 0800 4500
        0x0010:  0054 dcf9 0000 4001 899b 0a00 000a 0a00
        0x0020:  000b 0000 e16d 0027 0001 58b6 4b60 0000
        0x0030:  0000 ad80 0e00 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
21:43:37.958654 00:00:00:aa:00:01 (oui Ethernet) > 00:00:00:aa:00:00 (oui Ethernet), ethertype IPv4 (0x0800), length 98: 10.0.0.11 > n1: ICMP echo request, id 39, seq 2, length 64
        0x0000:  0000 00aa 0000 0000 00aa 0001 0800 4500
        0x0010:  0054 fcfb 4000 4001 2999 0a00 000b 0a00
        0x0020:  000a 0800 e14c 0027 0002 59b6 4b60 0000
        0x0030:  0000 a4a0 0e00 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
21:43:37.958668 00:00:00:aa:00:00 (oui Ethernet) > 00:00:00:aa:00:01 (oui Ethernet), ethertype IPv4 (0x0800), length 98: n1 > 10.0.0.11: ICMP echo reply, id 39, seq 2, length 64
        0x0000:  0000 00aa 0001 0000 00aa 0000 0800 4500
        0x0010:  0054 ddaa 0000 4001 88ea 0a00 000a 0a00
        0x0020:  000b 0000 e94c 0027 0002 59b6 4b60 0000
        0x0030:  0000 a4a0 0e00 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
21:43:38.982614 00:00:00:aa:00:01 (oui Ethernet) > 00:00:00:aa:00:00 (oui Ethernet), ethertype IPv4 (0x0800), length 98: 10.0.0.11 > n1: ICMP echo request, id 39, seq 3, length 64
        0x0000:  0000 00aa 0000 0000 00aa 0001 0800 4500
        0x0010:  0054 fdd7 4000 4001 28bd 0a00 000b 0a00
        0x0020:  000a 0800 46ee 0027 0003 5ab6 4b60 0000
        0x0030:  0000 3dfe 0e00 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
5 packets captured
6 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.

```
root@n1:/tmp/pycore.42027/n1.conf# tcpdump -c 3 -XX  'dst host 10.0.0.11 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
20:02:11.002829 IP n1 > 10.0.0.11: ICMP echo reply, id 51, seq 1, length 64
        0x0000:  0000 00aa 0001 0000 00aa 0000 0800 4500  ..............E.
        0x0010:  0054 deaa 0000 4001 87ea 0a00 000a 0a00  .T....@.........
        0x0020:  000b 0000 f79d 0033 0001 13f0 4c60 0000  .......3....L`..
        0x0030:  0000 e90a 0000 0000 0000 1011 1213 1415  ................
        0x0040:  1617 1819 1a1b 1c1d 1e1f 2021 2223 2425  ...........!"#$%
        0x0050:  2627 2829 2a2b 2c2d 2e2f 3031 3233 3435  &'()*+,-./012345
        0x0060:  3637                                     67
20:02:12.024118 IP n1 > 10.0.0.11: ICMP echo reply, id 51, seq 2, length 64
        0x0000:  0000 00aa 0001 0000 00aa 0000 0800 4500  ..............E.
        0x0010:  0054 ded4 0000 4001 87c0 0a00 000a 0a00  .T....@.........
        0x0020:  000b 0000 ce49 0033 0002 14f0 4c60 0000  .....I.3....L`..
        0x0030:  0000 115e 0000 0000 0000 1011 1213 1415  ...^............
        0x0040:  1617 1819 1a1b 1c1d 1e1f 2021 2223 2425  ...........!"#$%
        0x0050:  2627 2829 2a2b 2c2d 2e2f 3031 3233 3435  &'()*+,-./012345
        0x0060:  3637                                     67
20:02:13.047492 IP n1 > 10.0.0.11: ICMP echo reply, id 51, seq 3, length 64
        0x0000:  0000 00aa 0001 0000 00aa 0000 0800 4500  ..............E.
        0x0010:  0054 df47 0000 4001 874d 0a00 000a 0a00  .T.G..@..M......
        0x0020:  000b 0000 7ced 0033 0003 15f0 4c60 0000  ....|..3....L`..
        0x0030:  0000 61b9 0000 0000 0000 1011 1213 1415  ..a.............
        0x0040:  1617 1819 1a1b 1c1d 1e1f 2021 2223 2425  ...........!"#$%
        0x0050:  2627 2829 2a2b 2c2d 2e2f 3031 3233 3435  &'()*+,-./012345
        0x0060:  3637                                     67
3 packets captured
3 packets received by filter
0 packets dropped by kernel
```

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@n2:/tmp/pycore.42027/n2.conf# traceroute -q  7 -I 10.0.0.10
traceroute to 10.0.0.10 (10.0.0.10), 30 hops max, 60 byte packets
 1  10.0.0.10 (10.0.0.10)  0.126 ms  0.095 ms  0.090 ms  0.085 ms  0.081 ms  0.0
76 ms  0.071 ms
```

```
root@n1:/tmp/pycore.42027/n1.conf# tcpdump -c 7 -w lab5.cap
7 packets captured
32 packets received by filter
0 packets dropped by kernel
root@n1:/tmp/pycore.42027/n1.conf# tcpdump -c 7 -r lab5.cap -xx -X
13:07:55.421955 IP 10.0.0.11 > n1: ICMP echo request, id 93, seq 1, length 40
        0x0000:  4500 003c 16ae 0000 0101 8eff 0a00 000b  E..<............
        0x0010:  0a00 000a 0800 821c 005d 0001 4849 4a4b  .........]..HIJK
        0x0020:  4c4d 4e4f 5051 5253 5455 5657 5859 5a5b  LMNOPQRSTUVWXYZ[
        0x0030:  5c5d 5e5f 6061 6263 6465 6667            \]^_`abcdefg
13:07:55.421963 IP n1 > 10.0.0.11: ICMP echo reply, id 93, seq 1, length 40
        0x0000:  4500 003c 319f 0000 4001 350e 0a00 000a  E..<1...@.5.....
        0x0010:  0a00 000b 0000 8a1c 005d 0001 4849 4a4b  .........]..HIJK
        0x0020:  4c4d 4e4f 5051 5253 5455 5657 5859 5a5b  LMNOPQRSTUVWXYZ[
        0x0030:  5c5d 5e5f 6061 6263 6465 6667            \]^_`abcdefg
13:07:55.421971 IP 10.0.0.11 > n1: ICMP echo request, id 93, seq 2, length 40
        0x0000:  4500 003c 16af 0000 0101 8efe 0a00 000b  E..<............
        0x0010:  0a00 000a 0800 821b 005d 0002 4849 4a4b  .........]..HIJK
        0x0020:  4c4d 4e4f 5051 5253 5455 5657 5859 5a5b  LMNOPQRSTUVWXYZ[
        0x0030:  5c5d 5e5f 6061 6263 6465 6667            \]^_`abcdefg
13:07:55.421972 IP n1 > 10.0.0.11: ICMP echo reply, id 93, seq 2, length 40
        0x0000:  4500 003c 31a0 0000 4001 350d 0a00 000a  E..<1...@.5.....
        0x0010:  0a00 000b 0000 8a1b 005d 0002 4849 4a4b  .........]..HIJK
        0x0020:  4c4d 4e4f 5051 5253 5455 5657 5859 5a5b  LMNOPQRSTUVWXYZ[
        0x0030:  5c5d 5e5f 6061 6263 6465 6667            \]^_`abcdefg
13:07:55.421977 IP 10.0.0.11 > n1: ICMP echo request, id 93, seq 3, length 40
        0x0000:  4500 003c 16b0 0000 0101 8efd 0a00 000b  E..<............
        0x0010:  0a00 000a 0800 821a 005d 0003 4849 4a4b  .........]..HIJK
        0x0020:  4c4d 4e4f 5051 5253 5455 5657 5859 5a5b  LMNOPQRSTUVWXYZ[
        0x0030:  5c5d 5e5f 6061 6263 6465 6667            \]^_`abcdefg
13:07:55.421978 IP n1 > 10.0.0.11: ICMP echo reply, id 93, seq 3, length 40
        0x0000:  4500 003c 31a1 0000 4001 350c 0a00 000a  E..<1...@.5.....
        0x0010:  0a00 000b 0000 8a1a 005d 0003 4849 4a4b  .........]..HIJK
        0x0020:  4c4d 4e4f 5051 5253 5455 5657 5859 5a5b  LMNOPQRSTUVWXYZ[
        0x0030:  5c5d 5e5f 6061 6263 6465 6667            \]^_`abcdefg
13:07:55.421983 IP 10.0.0.11 > n1: ICMP echo request, id 93, seq 4, length 40
        0x0000:  4500 003c 16b1 0000 0101 8efc 0a00 000b  E..<............
        0x0010:  0a00 000a 0800 8219 005d 0004 4849 4a4b  .........]..HIJK
        0x0020:  4c4d 4e4f 5051 5253 5455 5657 5859 5a5b  LMNOPQRSTUVWXYZ[
        0x0030:  5c5d 5e5f 6061 6263 6465 6667            \]^_`abcdefg
```

```
tcpdump: listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes

reading from file lab5.cap, link-type EN10MB (Ethernet)
```

5. Прочесть программой tcpdump созданный в предыдущем пункте файл.

```
root@n1:/tmp/pycore.42027/n1.conf# tcpdump -c 7 -r lab5.cap
reading from file lab5.cap, link-type EN10MB (Ethernet)
13:07:55.421955 IP 10.0.0.11 > n1: ICMP echo request, id 93, seq 1, length 40
13:07:55.421963 IP n1 > 10.0.0.11: ICMP echo reply, id 93, seq 1, length 40
13:07:55.421971 IP 10.0.0.11 > n1: ICMP echo request, id 93, seq 2, length 40
13:07:55.421972 IP n1 > 10.0.0.11: ICMP echo reply, id 93, seq 2, length 40
13:07:55.421977 IP 10.0.0.11 > n1: ICMP echo request, id 93, seq 3, length 40
13:07:55.421978 IP n1 > 10.0.0.11: ICMP echo reply, id 93, seq 3, length 40
13:07:55.421983 IP 10.0.0.11 > n1: ICMP echo request, id 93, 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 протокола UDP, отправленные на определенный IP-адрес. При этом включить распечатку пакета в шестнадцатеричной системе (включая заголовок канального уровня). Количество захватываемых пакетов ограничить 7. Для генерирования пакетов воспользоваться утилитой traceroute.

```
root@n1:/tmp/pycore.42027/n1.conf# tcpdump -c 7 -xx 'dst host 10.0.0.10 and 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
13:24:36.286480 IP 10.0.0.11.39055 > n1.33434: UDP, length 32
        0x0000:  0000 00aa 0000 0000 00aa 0001 0800 4500
        0x0010:  003c 0130 0000 0111 a46d 0a00 000b 0a00
        0x0020:  000a 988f 829a 0028 db5a 4041 4243 4445
        0x0030:  4647 4849 4a4b 4c4d 4e4f 5051 5253 5455
        0x0040:  5657 5859 5a5b 5c5d 5e5f
13:24:36.286537 IP 10.0.0.11.33443 > n1.33435: UDP, length 32
        0x0000:  0000 00aa 0000 0000 00aa 0001 0800 4500
        0x0010:  003c 0131 0000 0111 a46c 0a00 000b 0a00
        0x0020:  000a 82a3 829b 0028 f145 4041 4243 4445
        0x0030:  4647 4849 4a4b 4c4d 4e4f 5051 5253 5455
        0x0040:  5657 5859 5a5b 5c5d 5e5f
13:24:36.286551 IP 10.0.0.11.58185 > n1.33436: UDP, length 32
        0x0000:  0000 00aa 0000 0000 00aa 0001 0800 4500
        0x0010:  003c 0132 0000 0111 a46b 0a00 000b 0a00
        0x0020:  000a e349 829c 0028 909e 4041 4243 4445
        0x0030:  4647 4849 4a4b 4c4d 4e4f 5051 5253 5455
        0x0040:  5657 5859 5a5b 5c5d 5e5f
13:24:36.286578 IP 10.0.0.11.50488 > n1.33437: UDP, length 32
        0x0000:  0000 00aa 0000 0000 00aa 0001 0800 4500
        0x0010:  003c 0133 0000 0111 a46a 0a00 000b 0a00
        0x0020:  000a c538 829d 0028 aeae 4041 4243 4445
        0x0030:  4647 4849 4a4b 4c4d 4e4f 5051 5253 5455
        0x0040:  5657 5859 5a5b 5c5d 5e5f
13:24:36.286592 IP 10.0.0.11.57320 > n1.33438: UDP, length 32
        0x0000:  0000 00aa 0000 0000 00aa 0001 0800 4500
        0x0010:  003c 0134 0000 0111 a469 0a00 000b 0a00
        0x0020:  000a dfe8 829e 0028 93fd 4041 4243 4445
        0x0030:  4647 4849 4a4b 4c4d 4e4f 5051 5253 5455
        0x0040:  5657 5859 5a5b 5c5d 5e5f
13:24:36.286605 IP 10.0.0.11.60821 > n1.33439: UDP, length 32
        0x0000:  0000 00aa 0000 0000 00aa 0001 0800 4500
        0x0010:  003c 0135 0000 0111 a468 0a00 000b 0a00
        0x0020:  000a ed95 829f 0028 864f 4041 4243 4445
        0x0030:  4647 4849 4a4b 4c4d 4e4f 5051 5253 5455
        0x0040:  5657 5859 5a5b 5c5d 5e5f
13:24:36.286618 IP 10.0.0.11.55209 > n1.33440: UDP, length 32
        0x0000:  0000 00aa 0000 0000 00aa 0001 0800 4500
        0x0010:  003c 0136 0000 0111 a467 0a00 000b 0a00
        0x0020:  000a d7a9 82a0 0028 9c3a 4041 4243 4445
        0x0030:  4647 4849 4a4b 4c4d 4e4f 5051 5253 5455
root@n2:/tmp/pycore.42027/n2.conf# traceroute -q 7 10.0.0.10
traceroute to 10.0.0.10 (10.0.0.10), 30 hops max, 60 byte packets
 1  10.0.0.10 (10.0.0.10)  0.299 ms  0.221 ms  0.209 ms  0.196 ms  0.173 ms  0.161 ms *
```

2. 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 и ASCII-

формате (включая заголовок канального уровня). Количество захватываемых пакетов ограничить 7.

```
root@n1:/tmp/pycore.42027/n1.conf# tcpdump -c 7 -XX 'ether proto \arp'
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
13:32:41.485381 ARP, Request who-has 10.0.0.11 tell n1, length 28
        0x0000:  0000 00aa 0001 0000 00aa 0000 0806 0001  ................
        0x0010:  0800 0604 0001 0000 00aa 0000 0a00 000a  ................
        0x0020:  0000 0000 0000 0a00 000b                 ..........
13:32:41.485414 ARP, Request who-has n1 tell 10.0.0.11, length 28
        0x0000:  0000 00aa 0000 0000 00aa 0001 0806 0001  ................
        0x0010:  0800 0604 0001 0000 00aa 0001 0a00 000b  ................
        0x0020:  0000 0000 0000 0a00 000a                 ..........
13:32:41.485423 ARP, Reply n1 is-at 00:00:00:aa:00:00 (oui Ethernet), length 28
        0x0000:  0000 00aa 0001 0000 00aa 0000 0806 0001  ................
        0x0010:  0800 0604 0002 0000 00aa 0000 0a00 000a  ................
        0x0020:  0000 00aa 0001 0a00 000b                 ..........
13:32:41.485424 ARP, Reply 10.0.0.11 is-at 00:00:00:aa:00:01 (oui Ethernet), length 28
        0x0000:  0000 00aa 0000 0000 00aa 0001 0806 0001  ................
        0x0010:  0800 0604 0002 0000 00aa 0001 0a00 000b  ................
        0x0020:  0000 00aa 0000 0a00 000a                 ..........
13:33:10.157382 ARP, Request who-has 10.0.0.11 tell n1, length 28
        0x0000:  0000 00aa 0001 0000 00aa 0000 0806 0001  ................
        0x0010:  0800 0604 0001 0000 00aa 0000 0a00 000a  ................
        0x0020:  0000 0000 0000 0a00 000b                 ..........
13:33:10.157464 ARP, Reply 10.0.0.11 is-at 00:00:00:aa:00:01 (oui Ethernet), length 28
        0x0000:  0000 00aa 0000 0000 00aa 0001 0806 0001  ................
        0x0010:  0800 0604 0002 0000 00aa 0001 0a00 000b  ................
        0x0020:  0000 00aa 0000 0a00 000a                 ..........
13:33:12.205407 ARP, Request who-has n1 tell 10.0.0.11, length 28
        0x0000:  0000 00aa 0000 0000 00aa 0001 0806 0001  ................
        0x0010:  0800 0604 0001 0000 00aa 0001 0a00 000b  ................
        0x0020:  0000 0000 0000 0a00 000a                 ..........
7 packets captured
8 packets received by filter
0 packets dropped by kernel
```

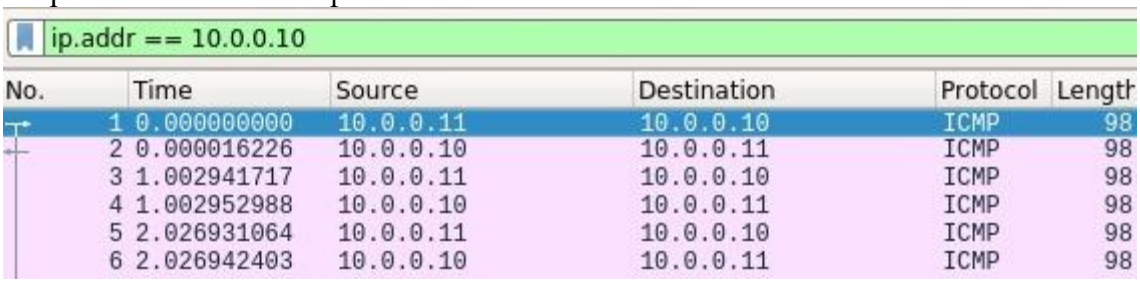
```
root@n2:/tmp/pycore.42027/n2.conf# ping 10.0.0.10
PING 10.0.0.10 (10.0.0.10) 56(84) bytes of data.
64 bytes from 10.0.0.10: icmp_seq=1 ttl=64 time=0.035 ms
64 bytes from 10.0.0.10: icmp_seq=2 ttl=64 time=0.041 ms
64 bytes from 10.0.0.10: icmp_seq=3 ttl=64 time=0.051 ms
64 bytes from 10.0.0.10: icmp_seq=4 ttl=64 time=0.040 ms
64 bytes from 10.0.0.10: icmp_seq=5 ttl=64 time=0.036 ms
64 bytes from 10.0.0.10: icmp_seq=6 ttl=64 time=0.042 ms
64 bytes from 10.0.0.10: icmp_seq=7 ttl=64 time=0.039 ms
64 bytes from 10.0.0.10: icmp_seq=8 ttl=64 time=0.039 ms
64 bytes from 10.0.0.10: icmp_seq=9 ttl=64 time=0.039 ms
```

3. Запустить tcpdump так, чтобы он перехватывал только пакеты протокола ICMP, отправленные на определенный IP-адрес. Включить распечатку пакета в шестнадцатеричной системе (включая заголовок канального уровня). Количество захватываемых пакетов ограничить 7. Для генерирования пакетов воспользоваться утилитой traceroute.

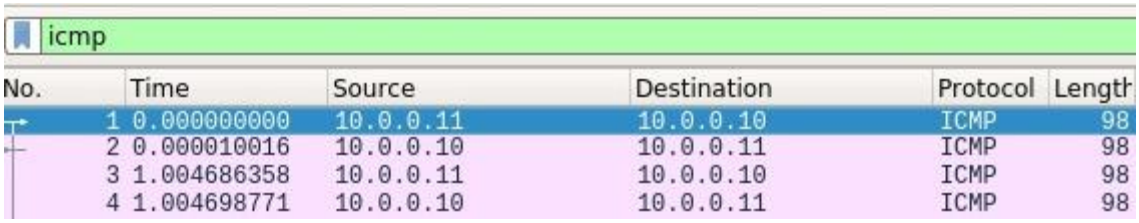
```
root@n1:/tmp/pycore.42027/n1.conf# tcpdump -c 7 -xx 'dst host 10.0.0.10 and ip  proto \icmp'
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
13:38:38.827336 IP 10.0.0.11 > n1: ICMP echo request, id 113, seq 1, length 40
        0x0000:  0000 00aa 0000 0000 00aa 0001 0800 4500
        0x0010:  003c 0ae1 0000 0101 9acc 0a00 000b 0a00
        0x0020:  000a 0800 8208 0071 0001 4849 4a4b 4c4d
        0x0030:  4e4f 5051 5253 5455 5657 5859 5a5b 5c5d
        0x0040:  5e5f 6061 6263 6465 6667
13:38:38.827356 IP 10.0.0.11 > n1: ICMP echo request, id 113, seq 2, length 40
        0x0000:  0000 00aa 0000 0000 00aa 0001 0800 4500
        0x0010:  003c 0ae2 0000 0101 9acb 0a00 000b 0a00
        0x0020:  000a 0800 8207 0071 0002 4849 4a4b 4c4d
        0x0030:  4e4f 5051 5253 5455 5657 5859 5a5b 5c5d
        0x0040:  5e5f 6061 6263 6465 6667
13:38:38.827362 IP 10.0.0.11 > n1: ICMP echo request, id 113, seq 3, length 40
        0x0000:  0000 00aa 0000 0000 00aa 0001 0800 4500
        0x0010:  003c 0ae3 0000 0101 9aca 0a00 000b 0a00
        0x0020:  000a 0800 8206 0071 0003 4849 4a4b 4c4d
        0x0030:  4e4f 5051 5253 5455 5657 5859 5a5b 5c5d
        0x0040:  5e5f 6061 6263 6465 6667
13:38:38.827368 IP 10.0.0.11 > n1: ICMP echo request, id 113, seq 4, length 40
        0x0000:  0000 00aa 0000 0000 00aa 0001 0800 4500
        0x0010:  003c 0ae4 0000 0101 9ac9 0a00 000b 0a00
        0x0020:  000a 0800 8205 0071 0004 4849 4a4b 4c4d
        0x0030:  4e4f 5051 5253 5455 5657 5859 5a5b 5c5d
        0x0040:  5e5f 6061 6263 6465 6667
13:38:38.827374 IP 10.0.0.11 > n1: ICMP echo request, id 113, seq 5, length 40
        0x0000:  0000 00aa 0000 0000 00aa 0001 0800 4500
        0x0010:  003c 0ae5 0000 0101 9ac8 0a00 000b 0a00
        0x0020:  000a 0800 8204 0071 0005 4849 4a4b 4c4d
        0x0030:  4e4f 5051 5253 5455 5657 5859 5a5b 5c5d
        0x0040:  5e5f 6061 6263 6465 6667
13:38:38.827380 IP 10.0.0.11 > n1: ICMP echo request, id 113, seq 6, length 40
        0x0000:  0000 00aa 0000 0000 00aa 0001 0800 4500
        0x0010:  003c 0ae6 0000 0101 9ac7 0a00 000b 0a00
        0x0020:  000a 0800 8203 0071 0006 4849 4a4b 4c4d
        0x0030:  4e4f 5051 5253 5455 5657 5859 5a5b 5c5d
        0x0040:  5e5f 6061 6263 6465 6667
13:38:38.827386 IP 10.0.0.11 > n1: ICMP echo request, id 113, seq 7, length 40
        0x0000:  0000 00aa 0000 0000 00aa 0001 0800 4500
        0x0010:  003c 0ae7 0000 0101 9ac6 0a00 000b 0a00
        0x0020:  000a 0800 8202 0071 0007 4849 4a4b 4c4d
        0x0030:  4e4f 5051 5253 5455 5657 5859 5a5b 5c5d
        0x0040:  5e5f 6061 6263 6465 6667
root@n2:/tmp/pycore.42027/n2.conf# traceroute -q 7 -I 10.0.0.10
traceroute to 10.0.0.10 (10.0.0.10), 30 hops max, 60 byte packets
 1  10.0.0.10 (10.0.0.10)  0.127 ms  0.093 ms  0.088 ms  0.083 ms  0.078 ms  0.0
73 ms  0.069 ms
```

Работа с анализатором протоколов wireshark

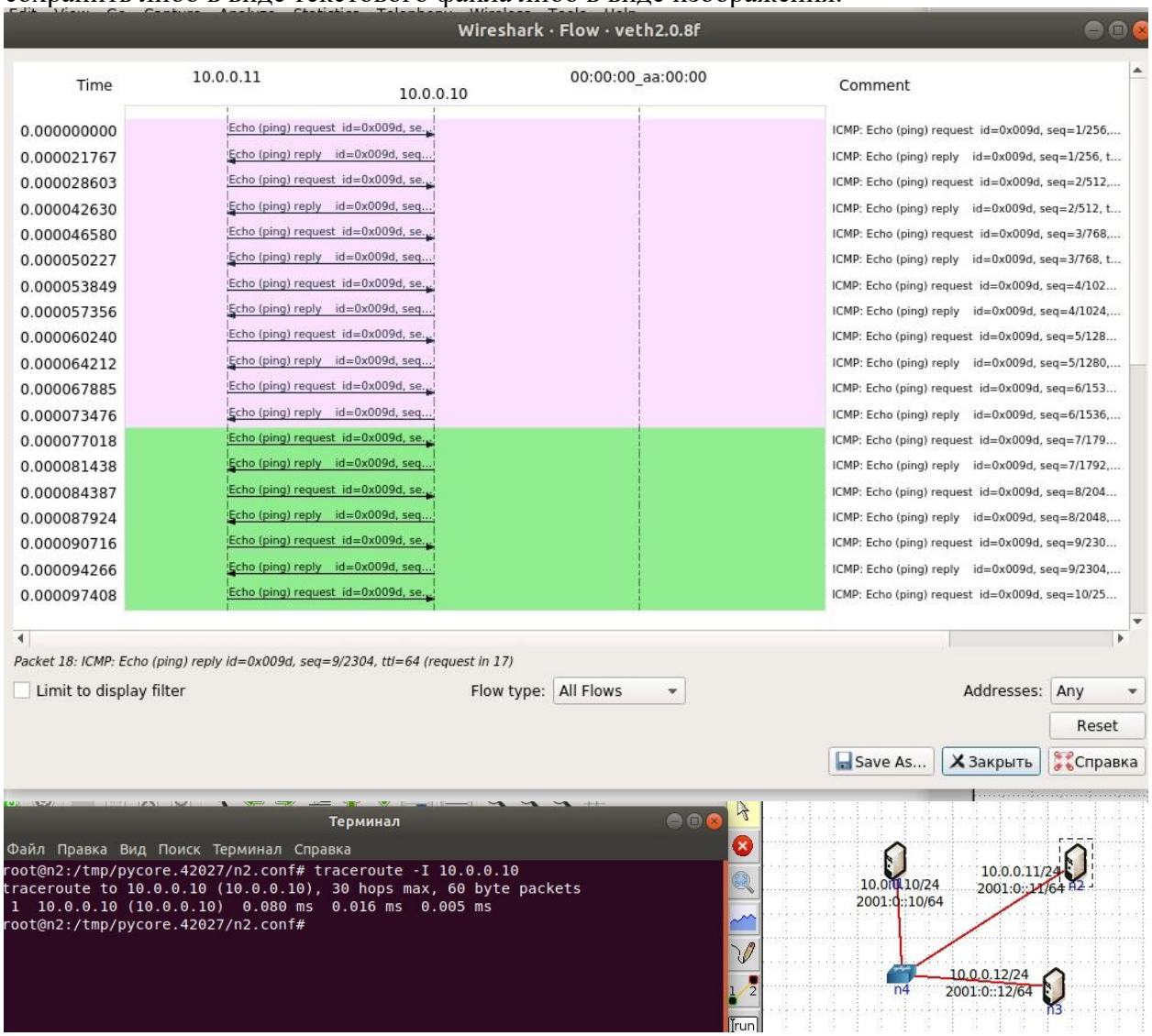
1. Захватить 5-7 пакетов широковещательного трафика (фильтр по IP-адресу). Результат сохранить в текстовый файл.



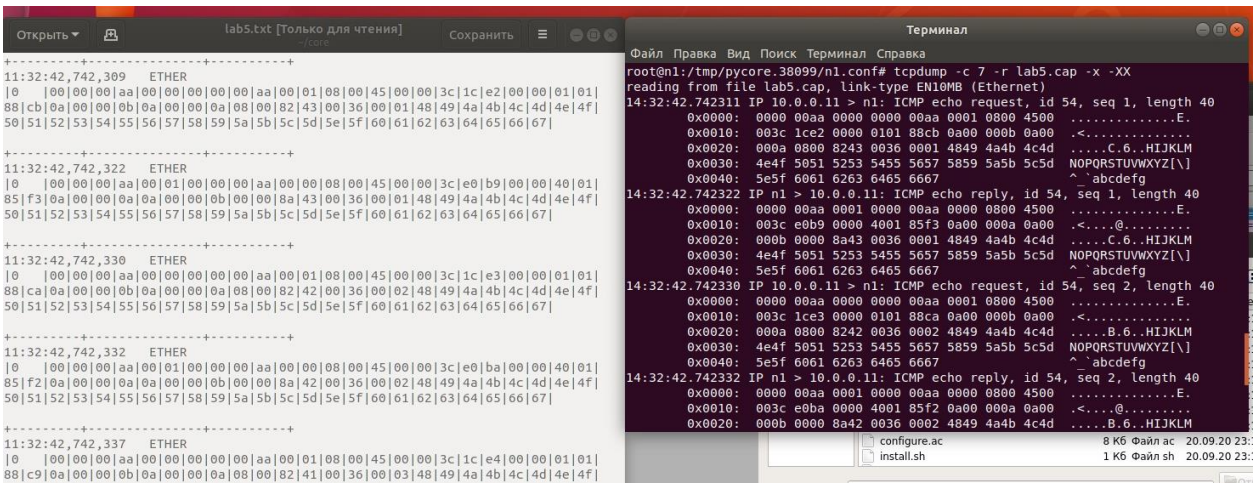
2. Захватить 3-4 пакета ICMP, полученных от определенного узла. Для генерирования пакетов воспользоваться утилитой ping. Результат сохранить в текстовый файл.



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.



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


Открыть
lab5.txt [Только для чтения]
~/core
Сохранить
+---------+--------------+----------+
11:32:42,742,309 ETHER
|0 |00|00|00|aa|00|00|00|00|00|aa|00|01|08|00|45|00|00|3c|1c|e2|00|00|01|01|
88|cb|0a|00|00|0b|0a|00|00|0a|08|00|82|43|00|36|00|01|48|49|4a|4b|4c|4d|4e|4f|
50|51|52|53|54|55|56|57|58|59|5a|5b|5c|5d|5e|5f|60|61|62|63|64|65|66|67|
11:32:42,742,322 ETHER
|0 |00|00|00|aa|00|01|00|00|00|aa|00|00|08|00|45|00|00|3c|e0|b9|00|00|40|01|
85|f3|0a|00|00|0a|0a|00|00|0b|00|00|8a|43|00|36|00|01|48|49|4a|4b|4c|4d|4e|4f|
50|51|52|53|54|55|56|57|58|59|5a|5b|5c|5d|5e|5f|60|61|62|63|64|65|66|67|
11:32:42,742,330 ETHER
|0 |00|00|00|aa|00|00|00|00|00|aa|00|01|08|00|45|00|00|3c|1c|e3|00|00|01|01|
88|ca|0a|00|00|0b|0a|00|00|0a|08|00|82|42|00|36|00|02|48|49|4a|4b|4c|4d|4e|4f|
50|51|52|53|54|55|56|57|58|59|5a|5b|5c|5d|5e|5f|60|61|62|63|64|65|66|67|
11:32:42,742,332 ETHER
|0 |00|00|00|aa|00|01|00|00|00|aa|00|00|08|00|45|00|00|3c|e0|ba|00|00|40|01|
85|f2|0a|00|00|0a|0a|00|00|0b|00|00|8a|42|00|36|00|02|48|49|4a|4b|4c|4d|4e|4f|
50|51|52|53|54|55|56|57|58|59|5a|5b|5c|5d|5e|5f|60|61|62|63|64|65|66|67|
11:32:42,742,337 ETHER
|0 |00|00|00|aa|00|00|00|00|00|aa|00|01|08|00|45|00|00|3c|1c|e4|00|00|01|01|
88|c9|0a|00|00|0b|0a|00|00|0a|08|00|82|41|00|36|00|03|48|49|4a|4b|4c|4d|4e|4f|
Терминал
Файл Правка Вид Поиск Терминал Справка
root@n1:/tmp/pycore.38099/n1.conf# tcpdump -c 7 -r lab5.cap -x -XX
reading from file lab5.cap, link-type EN10MB (Ethernet)
14:32:42.742311 IP 10.0.0.11 > n1: ICMP echo request, id 54, seq 1, length 40
0x0000: 0000 00aa 0000 0000 00aa 0001 0800 4500 ....E.
0x0010: 003c 1ce2 0000 0101 88cb 0a00 000b 0a00 .<..............
0x0020: 000a 0800 8243 0036 0001 4849 4a4b 4c4d .....C.6..HIJKLM
0x0030: 4e4f 5051 5253 5455 5657 5859 5a5b 5c5d NOPQRSTUVWXYZ[\]
0x0040: 5e5f 6061 6263 6465 6667 ^_`abcdefg
14:32:42.742322 IP n1 > 10.0.0.11: ICMP echo reply, id 54, seq 1, length 40
0x0000: 0000 00aa 0001 0000 00aa 0000 0800 4500 ....E.
0x0010: 003c e0b9 0000 4001 85f3 0a00 000a 0a00 .<....@.........
0x0020: 000b 0000 8a43 0036 0001 4849 4a4b 4c4d .....C.6..HIJKLM
0x0030: 4e4f 5051 5253 5455 5657 5859 5a5b 5c5d NOPQRSTUVWXYZ[\]
0x0040: 5e5f 6061 6263 6465 6667 ^_`abcdefg
14:32:42.742330 IP 10.0.0.11 > n1: ICMP echo request, id 54, seq 2, length 40
0x0000: 0000 00aa 0000 0000 00aa 0001 0800 4500 ....E.
0x0010: 003c 1ce3 0000 0101 88ca 0a00 000b 0a00 .<..............
0x0020: 000a 0800 8242 0036 0002 4849 4a4b 4c4d .....B.6..HIJKLM
0x0030: 4e4f 5051 5253 5455 5657 5859 5a5b 5c5d NOPQRSTUVWXYZ[\]
0x0040: 5e5f 6061 6263 6465 6667 ^_`abcdefg
14:32:42.742332 IP n1 > 10.0.0.11: ICMP echo reply, id 54, seq 2, length 40
0x0000: 0000 00aa 0001 0000 00aa 0000 0800 4500 ....E.
0x0010: 003c e0ba 0000 4001 85f2 0a00 000a 0a00 .<....@.........
0x0020: 000b 0000 8a42 0036 0002 4849 4a4b 4c4d .....B.6..HIJKLM
configure.ac 8 Кб Файл ac 20.09.20 23:
install.sh 1 Кб Файл sh 20.09.20 23: