Assignment 2

- Name: Abhiroop Mukherjee
- Roll No.: 510519109
- GSuite: 510519109.abhirup@students.iiests.ac.in

- Subject: Computer Networks Lab (CS 3272)

Question (a)

Check the version of the `tcpdump` and the `libpcap` utilities. Also find the number of interfaces available with your computer. Switch the network of eth0/eth1 (or the ethernet interface name as appeared) to promiscuous mode.

Answer

1.
   - Check Version of tcpdump and libcap: `sudo tcpdump --version`
   -
```
tcpdump version 4.9.3
libpcap version 1.9.1 (with TPACKET_V3)
OpenSSL 1.1.1f  31 Mar 2020
```

2.
   - Iterfaces available: `tcpdump -D`
   -
```
1.eth0 [Up, Running]
2.lo [Up, Running, Loopback]
3.any (Pseudo-device that captures on all interfaces) [Up,
Running]
4.bluetooth-monitor (Bluetooth Linux Monitor) [none]
5.nflog (Linux netfilter log (NFLOG) interface) [none]
6.nfqueue (Linux netfilter queue (NFQUEUE) interface) [none]
7.dummy0 [none]
8.tunl0 [none]
9.sit0 [none]
10.bond0 [none]
```

3.
   - Switch the network of eth0 to promiscuous mode: `sudo ifconfig eth0 promisc`
   - Check if promiscuous mode is turned on: `ifconfig | grep PROMISC`
   -
```
eth0: flags=4419<UP,BROADCAST,RUNNING,PROMISC,MULTICAST>  mtu
1500
```
   - hence turned on promiscuous mode and verified

Question (b)

Write the tcpdump command to capture 20 packets by listening to the promiscuous mode interface of your host and save the result as *.pcap file (both with and without -n option).

Answer

- did `curl www.google.com` with both the commands

1. `sudo tcpdump -i eth0 -n -w "with-n.pcap" -c 20`
2. `sudo tcpdump -i eth0 -w "without-n.pcap" -c 20`

Question (c)

Read the above file and identify the different fields present in TCP/IP packets captured by `tcpdump`.

Answer

1. Read `with-n.pcap` : `tcpdump -n -r with-n.pcap`

```
[RAM: 15% | SWAP: 0%] .../Sem 6/CS 3272 Computer Network Lab/Assignment 2
[Batt: 79%][12:49 PM] ❯ tcpdump -n -r with-n.pcap
reading from file with-n.pcap, link-type EN10MB (Ethernet)
12:45:59.697061 ARP, Request who-has 172.22.224.1 tell 172.22.237.67, length 28
12:45:59.697310 ARP, Reply 172.22.224.1 is-at 00:15:5d:9a:7d:d9, length 28
12:46:03.702340 IP 172.22.237.67.59462 > 172.22.224.1.53: 37153+ A? www.google.com. (32)
12:46:03.702345 IP 172.22.237.67.59462 > 172.22.224.1.53: 58685+ AAAA? www.google.com. (32)
12:46:03.719965 IP 172.22.224.1.53 > 172.22.237.67.59462: 58685- 1/0/0 AAAA 2404:6800:4009:823::2004 (74)
12:46:03.721589 IP 172.22.224.1.53 > 172.22.237.67.59462: 37153- 1/0/0 A 142.250.76.164 (62)
12:46:03.721897 IP 172.22.237.67.38816 > 142.250.76.164.80: Flags [S], seq 1576795409, win 64240, options [mss 1460,sackOK,TS val 3642672798 ecr 0,nop,wscale 7], length 0
12:46:03.729393 IP 142.250.76.164.80 > 172.22.237.67.38816: Flags [S.], seq 298809061⁴, ack 1576795410, win 65535, options [mss 1412,sackOK,TS val 3857071474 ecr 3642672798,nop,wscale 8], length 0
12:46:03.729467 IP 172.22.237.67.38816 > 142.250.76.164.80: Flags [.], ack 1, win 502, options [nop,nop,TS val 3642672885 ecr 3857071474], length 0
12:46:03.729548 IP 172.22.237.67.38816 > 142.250.76.164.80: Flags [P.], seq 1:79, ack 1, win 502, options [nop,nop,TS val 3642672806 ecr 3857071474], length 78: HTTP: GET / HTTP/1.1
12:46:03.737581 IP 142.250.76.164.80 > 172.22.237.67.38816: Flags [.], ack 79, win 256, options [nop,nop,TS val 3857071482 ecr 3642672806], length 0
12:46:03.899822 IP 142.250.76.164.80 > 172.22.237.67.38816: Flags [P.], seq 1:8401, ack 79, win 256, options [nop,nop,TS val 3857071643 ecr 3642672806], length 8400: HTTP: HTTP/1.1 200 OK
12:46:03.899891 IP 172.22.237.67.38816 > 142.250.76.164.80: Flags [.], ack 8401, win 447, options [nop,nop,TS val 3642672976 ecr 3857071643], length 0
12:46:03.900161 IP 142.250.76.164.80 > 172.22.237.67.38816: Flags [P.], seq 8401:12601, ack 79, win 256, options [nop,nop,TS val 3857071643 ecr 36426728061, length 4200: HTTP
12:46:03.900195 IP 172.22.237.67.38816 > 142.250.76.164.80: Flags [.], ack 12601, win 447, options [nop,nop,TS val 3642672976 ecr 3857071643], length 0
12:46:03.900510 IP 142.250.76.164.80 > 172.22.237.67.38816: Flags [.], seq 12601:14001, ack 79, win 256, options [nop,nop,TS val 3857071643 ecr 36426728061, length 1400: HTTP
12:46:03.900528 IP 172.22.237.67.38816 > 142.250.76.164.80: Flags [.], ack 14001, win 501, options [nop,nop,TS val 3642672977 ecr 3857071643], length 0
12:46:03.907765 IP 142.250.76.164.80 > 172.22.237.67.38816: Flags [.], seq 14001:16801, ack 79, win 256, options [nop,nop,TS val 3857071653 ecr 3642672976], length 2800: HTTP
12:46:03.907765 IP 142.250.76.164.80 > 172.22.237.67.38816: Flags [P.], seq 16801:17152, ack 79, win 256, options [nop,nop,TS val 3857071653 ecr 3642672976], length 351: HTTP
12:46:03.907778 IP 172.22.237.67.38816 > 142.250.76.164.80: Flags [.], ack 16801, win 499, options [nop,nop,TS val 3642672984 ecr 3857071653], length 0
```

2. Read `without-n.pcap` : `tcpdump -r without-n.pcap`

```
[RAM: 15% | SWAP: 0%] .../Sem 6/CS 3272 Computer Network Lab/Assignment 2
[Batt: 79%][12:52 PM] ❯ tcpdump -r without-n.pcap
reading from file without-n.pcap, link-type EN10MB (Ethernet)
12:46:16.331837 IP 172.22.237.67.52176 > LAPTOP-TNR28RHP.mshome.net.domain: 23943+ A? www.google.com. (32)
12:46:16.331842 IP 172.22.237.67.52176 > LAPTOP-TNR28RHP.mshome.net.domain: 56968+ AAAA? www.google.com. (32)
12:46:16.338985 IP LAPTOP-TNR28RHP.mshome.net.domain > 172.22.237.67.52176: 23943- 1/0/0 A 142.250.76.164 (62)
12:46:16.338985 IP LAPTOP-TNR28RHP.mshome.net.domain > 172.22.237.67.52176: 56968- 1/0/0 AAAA 2404:6800:4009:823::2004 (74)
12:46:16.339334 IP 172.22.237.67.38818 > bom12s09-in-f4.1e100.net.http: Flags [S], seq 2327451116, win 64240, options [mss 1460,sackOK,TS val 3642685415 ecr 0,nop,wscale 7], length 0
12:46:16.349331 IP bom12s09-in-f4.1e100.net.http > 172.22.237.67.38818: Flags [S.], seq 633643757, ack 2327451117, win 65535, options [mss 1412,sackOK,TS val 2032047857 ecr 3642685415,nop,wscale 8], length 0
12:46:16.349400 IP 172.22.237.67.38818 > bom12s09-in-f4.1e100.net.http: Flags [.], ack 1, win 502, options [nop,nop,TS val 3642685425 ecr 2032047857], length 0
12:46:16.349517 IP 172.22.237.67.38818 > bom12s09-in-f4.1e100.net.http: Flags [P.], seq 1:79, ack 1, win 502, options [nop,nop,TS val 3642685426 ecr 2032047857], length 78: HTTP: GET / HTTP/1.1
12:46:16.361304 IP bom12s09-in-f4.1e100.net.http > 172.22.237.67.38818: Flags [.], ack 79, win 256, options [nop,nop,TS val 2032047870 ecr 3642685426], length 0
12:46:16.519731 IP bom12s09-in-f4.1e100.net.http > 172.22.237.67.38818: Flags [P.], seq 1:14001, ack 79, win 256, options [nop,nop,TS val 2032048022 ecr 3642685426], length 14000: HTTP: HTTP/1.1 200 OK
12:46:16.519817 IP 172.22.237.67.38818 > bom12s09-in-f4.1e100.net.http: Flags [.], ack 14001, win 447, options [nop,nop,TS val 3642685596 ecr 2032048022], length 0
12:46:16.529770 IP bom12s09-in-f4.1e100.net.http > 172.22.237.67.38818: Flags [P.], seq 14001:16801, ack 79, win 256, options [nop,nop,TS val 2032048037 ecr 3642685596], length 2800: HTTP
12:46:16.529770 IP bom12s09-in-f4.1e100.net.http > 172.22.237.67.38818: Flags [P.], seq 16801:17166, ack 79, win 256, options [nop,nop,TS val 2032048038 ecr 3642685596], length 365: HTTP
12:46:16.529784 IP 172.22.237.67.38818 > bom12s09-in-f4.1e100.net.http: Flags [.], ack 16801, win 499, options [nop,nop,TS val 3642685606 ecr 2032048037], length 0
12:46:16.529814 IP 172.22.237.67.38818 > bom12s09-in-f4.1e100.net.http: Flags [.], ack 17166, win 497, options [nop,nop,TS val 3642685606 ecr 2032048038], length 0
12:46:16.530014 IP 172.22.237.67.38818 > bom12s09-in-f4.1e100.net.http: Flags [F.], seq 79, ack 17166, win 501, options [nop,nop,TS val 3642685606 ecr 2032048038], length 0
12:46:16.538975 IP bom12s09-in-f4.1e100.net.http > 172.22.237.67.38818: Flags [F.], seq 17166, ack 80, win 256, options [nop,nop,TS val 2032048048 ecr 3642685606], length 0
12:46:16.539003 IP 172.22.237.67.38818 > bom12s09-in-f4.1e100.net.http: Flags [.], ack 17167, win 501, options [nop,nop,TS val 3642685615 ecr 2032048048], length 0
12:46:20.486394 IP 20.84.171.179.https > 172.22.237.67.45814: Flags [.], ack 3650756280, win 1096, options [nop,nop,TS val 904982087 ecr 1411957013], length 0
12:46:20.486410 IP 172.22.237.67.45814 > 20.84.171.179.https: Flags [.], ack 1, win 4133, options [nop,nop,TS val 1419722471 ecr 904921247], length 0
```

3. consider the following packet (from `tcpdump -n -r with-n.pcap`):

    - `12:46:03.899822 IP 142.250.76.164.80 > 172.22.237.67.38816: Flags [P.], seq 1:8401, ack 79, win 256, options [nop,nop,TS val 3857071643 ecr 3642672806], length 8400: HTTP: HTTP/1.1 200 OK`

Data	Description
`12:46:03.899822`	Timestamp of the packet dumped
`IP`	IP protocol used, here its IPv4
`172.22.237.67.38816`	Source IP address and port
`142.250.76.164.80`	Destination IP address and port
`Flags [P.]`	Flags of the packet, P means `PUSH` and . means `ACK`
`seq 1:8401`	Sequence number of the packet
`ack 79`	Acknowledgement number of the packet. 79 represents the next expected byts(data) on the network flow
`win 256`	Window size of the packet, represents the no. of bytes available in the receiving buffer
`length 8400`	Length of the packet, reptesents the length in bytes of the payload data
`HTTP: HTTP/1.1 200 OK`	Payload data, here HTTP 200 OK response to my laptop

Question (d)

Extract packet arrival time, source IP address, destination IP address and port.

Answer

- `tcpdump -tttt -n -r with-n.pcap -c 3`

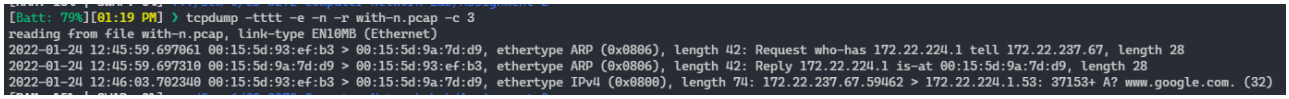
```
[RAM: 15% | SWAP: 0%] .../Sem 6/CS 3272 Computer Network Lab/Assignment 2
[Batt: 79%][01:19 PM] ❯ tcpdump -tttt -n -r with-n.pcap -c 3
reading from file with-n.pcap, link-type EN10MB (Ethernet)
2022-01-24 12:45:59.697061 ARP, Request who-has 172.22.224.1 tell 172.22.237.67, length 28
2022-01-24 12:45:59.697310 ARP, Reply 172.22.224.1 is-at 00:15:5d:9a:7d:d9, length 28
2022-01-24 12:46:03.702340 IP 172.22.237.67.59462 > 172.22.224.1.53: 37153+ A? www.google.com. (32)
```

- From the last record:
    - packet arrival time: `12:46:03.702340`
    - source IP address: `172.22.237.67`
    - destination IP address: `172.22.224.1`
    - destination port: `53`

Question (e)

Extract source MAC address and destination MAC addresses.

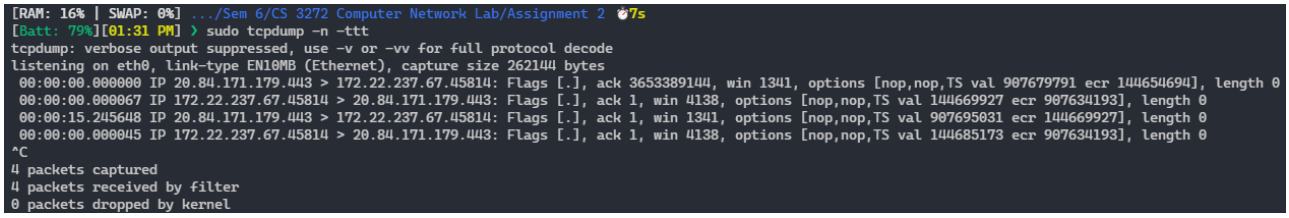
Answer

- we will use the `-e` tag to extract the MAC address: `tcpdump -tttt -e -n -r with-n.pcap -c 3`
- 
```
[Batt: 79%][01:19 PM] ❯ tcpdump -tttt -e -n -r with-n.pcap -c 3
reading from file with-n.pcap, link-type EN10MB (Ethernet)
2022-01-24 12:45:59.697061 00:15:5d:93:ef:b3 > 00:15:5d:9a:7d:d9, ethertype ARP (0x0806), length 42: Request who-has 172.22.224.1 tell 172.22.237.67, length 28
2022-01-24 12:45:59.697310 00:15:5d:9a:7d:d9 > 00:15:5d:93:ef:b3, ethertype ARP (0x0806), length 42: Reply 172.22.224.1 is-at 00:15:5d:9a:7d:d9, length 28
2022-01-24 12:46:03.702340 00:15:5d:93:ef:b3 > 00:15:5d:9a:7d:d9, ethertype IPv4 (0x0800), length 74: 172.22.237.67.59462 > 172.22.224.1.53: 37153+ A? www.google.com. (32)
```
- From the last record from the image:
  - source MAC address: `00:15:5d:93:ef:b3`
  - destination MAC address: `00:15:5d:9a:7d:d9`

Question (f)

Get the inter-arrival times while capturing packets.

Answer

- the `-ttt` tag shows inter-arrival time instead of arrival time in the result
- `sudo tcpdump -n -ttt`
- 
```
[RAM: 16% | SWAP: 0%] .../Sem 6/CS 3272 Computer Network Lab/Assignment 2 7s
[Batt: 79%][01:31 PM] ❯ sudo tcpdump -n -ttt
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
 00:00:00.000000 IP 20.84.171.179.443 > 172.22.237.67.45814: Flags [.], ack 36533889144, win 1341, options [nop,nop,TS val 907679791 ecr 144654694], length 0
 00:00:00.000067 IP 172.22.237.67.45814 > 20.84.171.179.443: Flags [.], ack 1, win 4138, options [nop,nop,TS val 144669927 ecr 907634193], length 0
 00:00:15.245648 IP 20.84.171.179.443 > 172.22.237.67.45814: Flags [.], ack 1, win 1341, options [nop,nop,TS val 907695031 ecr 144669927], length 0
 00:00:00.000045 IP 172.22.237.67.45814 > 20.84.171.179.443: Flags [.], ack 1, win 4138, options [nop,nop,TS val 144685173 ecr 907634193], length 0
^C
4 packets captured
4 packets received by filter
0 packets dropped by kernel
```

Question (g)

Use `tcpdump` to capture HTTP/HTTPS request and reply from www.google.com. Also print the packet content in ASCII format.

Answer

- We will use the `-A` flag to also print the ASCII data of the packet payload
- HTTP Default Port: 80
- HTTPS Default Port 443
- Host: www.google.com
- Hence we will use the following command:
    - `sudo tcpdump -A 'host www.google.com and (port 80 or port 443)' -c 1`

```
[RAM: 16% | SWAP: 0%] .../Sem 6/CS 3272 Computer Network Lab/Assignment 2
[Batt: 79%][01:44 PM] ❯ sudo tcpdump -A 'host www.google.com and (port 80 or port 443)' -c 1
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
13:44:21.506940 IP 172.22.237.67.39334 > bom12s09-in-f4.1e100.net.http: Flags [S], seq 2631450235, win 64240, options [mss 1460,sackOK
,TS val 3646170583 ecr 0,nop,wscale 7], length 0
E..<3.@.@......C..L....P...{........u'.........
.T%.........
1 packet captured
25 packets received by filter
0 packets dropped by kernel
```

Question (h)

For each command, use `tcpdump` to capture the associated packets, and explain the different fields of each request and reply: (i) ping (ii) wget (iii) traceroute

Answer

`ping`

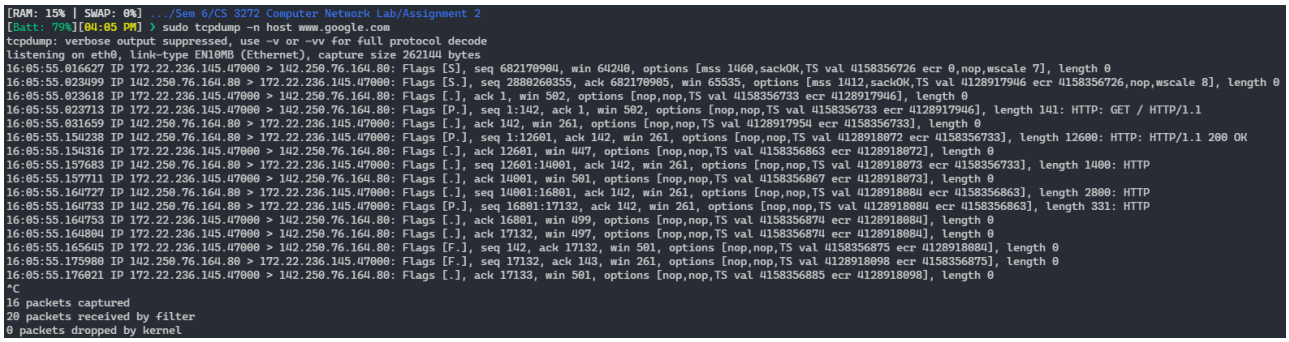
- `sudo tcpdump -n host www.google.com -c 6` and `ping www.google.com -c 5`

```
[RAM: 14% | SWAP: 0%] .../Sem 6/CS 3272 Computer Network Lab/Assignment 2
[Batt: 79%][03:48 PM] ❯ sudo tcpdump -n host www.google.com -c 6
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
15:49:00.794798 IP 172.22.236.145 > 142.250.76.164: ICMP echo request, id 600, seq 23, length 64
15:49:00.806128 IP 142.250.76.164 > 172.22.236.145: ICMP echo reply, id 600, seq 23, length 64
15:49:01.796925 IP 172.22.236.145 > 142.250.76.164: ICMP echo request, id 600, seq 24, length 64
15:49:01.809005 IP 142.250.76.164 > 172.22.236.145: ICMP echo reply, id 600, seq 24, length 64
15:49:02.798378 IP 172.22.236.145 > 142.250.76.164: ICMP echo request, id 600, seq 25, length 64
15:49:02.807198 IP 142.250.76.164 > 172.22.236.145: ICMP echo reply, id 600, seq 25, length 64
6 packets captured
10 packets received by filter
0 packets dropped by kernel
```

- my IP: 172.22.236.145
- We can observe that `ping` uses ICMP Protocol for contacting hosts
- We can observe that for every ICMP echo request given by my computer, it receives an ICMP echo request by the destination IP; all request having same size wich is the default 64 bytes given by ping

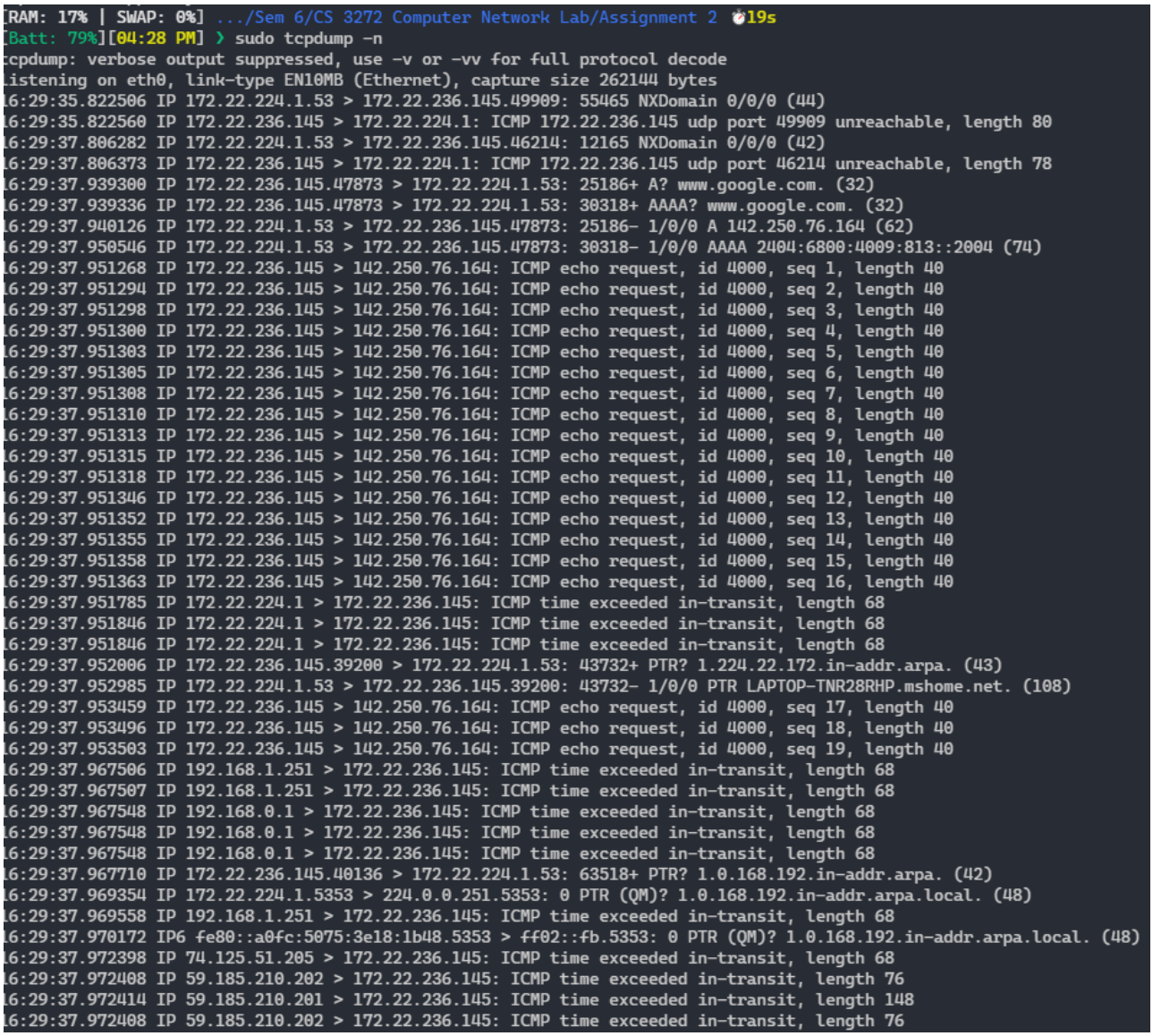
wget

- `sudo tcpdump -n host www.google.com` and `wget www.google.com -O /tmp/index.html`

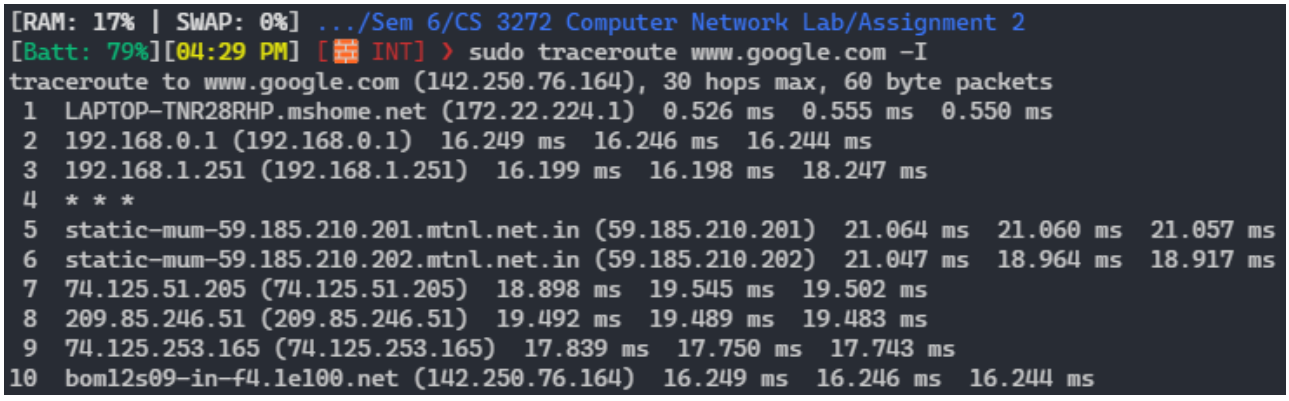


Packet No.	Source	Destination	Packet Type	Packet Length
1	172.22.236.145.47000	142.250.76.164.80	[S]-> SYN	0
2	142.250.76.164.80	172.22.236.145.47000	[S.] -> SYN ACK	0
3	172.22.236.145.47000	142.250.76.164.80	[.] -> ACK	0
4	172.22.236.145.47000	142.250.76.164.80	[P.] -> PUSH ACK	141 -> GET REQ
5	142.250.76.164.80	172.22.236.145.47000	[.] -> ACK	0
6	142.250.76.164.80	172.22.236.145.47000	[P.] -> PUSH ACK	12600 -> 200 OK RES
7	172.22.236.145.47000	142.250.76.164.80	[.] -> ACK	0
8	142.250.76.164.80	172.22.236.145.47000	[.] -> ACK	1400 -> HTTP
9	172.22.236.145.47000	142.250.76.164.80	[.] -> ACK	0
10	142.250.76.164.80	172.22.236.145.47000	[.] -> ACK	2800 -> HTTP
11	142.250.76.164.80	172.22.236.145.47000	[P.] -> PUSH ACK	331 -> HTTP
12	172.22.236.145.47000	142.250.76.164.80	[.] -> ACK	0
13	172.22.236.145.47000	142.250.76.164.80	[.] -> ACK	0
14	172.22.236.145.47000	142.250.76.164.80	[F.] -> FIN ACK	0
15	142.250.76.164.80	172.22.236.145.47000	[F.] -> FIN ACK	0
16	172.22.236.145.47000	142.250.76.164.80	[.] -> ACK	0

traceroute

- `sudo tcpdump -n` and `sudo traceroute www.google.com -I`
- `-I` will give ICMP requests for traceroute
- 

```
[RAM: 17% | SWAP: 0%] .../Sem 6/CS 3272 Computer Network Lab/Assignment 2 19s
[Batt: 79%][04:28 PM] ❯ sudo tcpdump -n
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
16:29:35.822506 IP 172.22.224.1.53 > 172.22.236.145.49909: 55465 NXDomain 0/0/0 (44)
16:29:35.822560 IP 172.22.236.145 > 172.22.224.1: ICMP 172.22.236.145 udp port 49909 unreachable, length 80
16:29:37.806282 IP 172.22.224.1.53 > 172.22.236.145.46214: 12165 NXDomain 0/0/0 (42)
16:29:37.806373 IP 172.22.236.145 > 172.22.224.1: ICMP 172.22.236.145 udp port 46214 unreachable, length 78
16:29:37.939300 IP 172.22.236.145.47873 > 172.22.224.1.53: 25186+ A? www.google.com. (32)
16:29:37.939336 IP 172.22.236.145.47873 > 172.22.224.1.53: 30318+ AAAA? www.google.com. (32)
16:29:37.940126 IP 172.22.224.1.53 > 172.22.236.145.47873: 25186- 1/0/0 A 142.250.76.164 (62)
16:29:37.950546 IP 172.22.224.1.53 > 172.22.236.145.47873: 30318- 1/0/0 AAAA 2404:6800:4009:813::2004 (74)
16:29:37.951268 IP 172.22.236.145 > 142.250.76.164: ICMP echo request, id 4000, seq 1, length 40
16:29:37.951294 IP 172.22.236.145 > 142.250.76.164: ICMP echo request, id 4000, seq 2, length 40
16:29:37.951298 IP 172.22.236.145 > 142.250.76.164: ICMP echo request, id 4000, seq 3, length 40
16:29:37.951300 IP 172.22.236.145 > 142.250.76.164: ICMP echo request, id 4000, seq 4, length 40
16:29:37.951303 IP 172.22.236.145 > 142.250.76.164: ICMP echo request, id 4000, seq 5, length 40
16:29:37.951305 IP 172.22.236.145 > 142.250.76.164: ICMP echo request, id 4000, seq 6, length 40
16:29:37.951308 IP 172.22.236.145 > 142.250.76.164: ICMP echo request, id 4000, seq 7, length 40
16:29:37.951310 IP 172.22.236.145 > 142.250.76.164: ICMP echo request, id 4000, seq 8, length 40
16:29:37.951313 IP 172.22.236.145 > 142.250.76.164: ICMP echo request, id 4000, seq 9, length 40
16:29:37.951315 IP 172.22.236.145 > 142.250.76.164: ICMP echo request, id 4000, seq 10, length 40
16:29:37.951318 IP 172.22.236.145 > 142.250.76.164: ICMP echo request, id 4000, seq 11, length 40
16:29:37.951346 IP 172.22.236.145 > 142.250.76.164: ICMP echo request, id 4000, seq 12, length 40
16:29:37.951352 IP 172.22.236.145 > 142.250.76.164: ICMP echo request, id 4000, seq 13, length 40
16:29:37.951355 IP 172.22.236.145 > 142.250.76.164: ICMP echo request, id 4000, seq 14, length 40
16:29:37.951358 IP 172.22.236.145 > 142.250.76.164: ICMP echo request, id 4000, seq 15, length 40
16:29:37.951363 IP 172.22.236.145 > 142.250.76.164: ICMP echo request, id 4000, seq 16, length 40
16:29:37.951785 IP 172.22.224.1 > 172.22.236.145: ICMP time exceeded in-transit, length 68
16:29:37.951846 IP 172.22.224.1 > 172.22.236.145: ICMP time exceeded in-transit, length 68
16:29:37.951846 IP 172.22.224.1 > 172.22.236.145: ICMP time exceeded in-transit, length 68
16:29:37.952006 IP 172.22.236.145.39200 > 172.22.224.1.53: 43732+ PTR? 1.224.22.172.in-addr.arpa. (43)
16:29:37.952985 IP 172.22.224.1.53 > 172.22.236.145.39200: 43732- 1/0/0 PTR LAPTOP-TNR28RHP.mshome.net. (108)
16:29:37.953459 IP 172.22.236.145 > 142.250.76.164: ICMP echo request, id 4000, seq 17, length 40
16:29:37.953496 IP 172.22.236.145 > 142.250.76.164: ICMP echo request, id 4000, seq 18, length 40
16:29:37.953503 IP 172.22.236.145 > 142.250.76.164: ICMP echo request, id 4000, seq 19, length 40
16:29:37.967506 IP 192.168.1.251 > 172.22.236.145: ICMP time exceeded in-transit, length 68
16:29:37.967507 IP 192.168.1.251 > 172.22.236.145: ICMP time exceeded in-transit, length 68
16:29:37.967548 IP 192.168.0.1 > 172.22.236.145: ICMP time exceeded in-transit, length 68
16:29:37.967548 IP 192.168.0.1 > 172.22.236.145: ICMP time exceeded in-transit, length 68
16:29:37.967548 IP 192.168.0.1 > 172.22.236.145: ICMP time exceeded in-transit, length 68
16:29:37.967710 IP 172.22.236.145.40136 > 172.22.224.1.53: 63518+ PTR? 1.0.168.192.in-addr.arpa. (42)
16:29:37.969354 IP 172.22.224.1.5353 > 224.0.0.251.5353: 0 PTR (QM)? 1.0.168.192.in-addr.arpa.local. (48)
16:29:37.969558 IP 192.168.1.251 > 172.22.236.145: ICMP time exceeded in-transit, length 68
16:29:37.970172 IP6 fe80::a0fc:5075:3e18:1b48.5353 > ff02::fb.5353: 0 PTR (QM)? 1.0.168.192.in-addr.arpa.local. (48)
16:29:37.972398 IP 74.125.51.205 > 172.22.236.145: ICMP time exceeded in-transit, length 68
16:29:37.972408 IP 59.185.210.202 > 172.22.236.145: ICMP time exceeded in-transit, length 76
16:29:37.972414 IP 59.185.210.201 > 172.22.236.145: ICMP time exceeded in-transit, length 148
16:29:37.972408 IP 59.185.210.202 > 172.22.236.145: ICMP time exceeded in-transit, length 76
```

- 

```
[RAM: 17% | SWAP: 0%] .../Sem 6/CS 3272 Computer Network Lab/Assignment 2
[Batt: 79%][04:29 PM] [🧱 INT] ❯ sudo traceroute www.google.com -I
traceroute to www.google.com (142.250.76.164), 30 hops max, 60 byte packets
 1  LAPTOP-TNR28RHP.mshome.net (172.22.224.1)  0.526 ms  0.555 ms  0.550 ms
 2  192.168.0.1 (192.168.0.1)  16.249 ms  16.246 ms  16.244 ms
 3  192.168.1.251 (192.168.1.251)  16.199 ms  16.198 ms  18.247 ms
 4  * * *
 5  static-mum-59.185.210.201.mtnl.net.in (59.185.210.201)  21.064 ms  21.060 ms  21.057 ms
 6  static-mum-59.185.210.202.mtnl.net.in (59.185.210.202)  21.047 ms  18.964 ms  18.917 ms
 7  74.125.51.205 (74.125.51.205)  18.898 ms  19.545 ms  19.502 ms
 8  209.85.246.51 (209.85.246.51)  19.492 ms  19.489 ms  19.483 ms
 9  74.125.253.165 (74.125.253.165)  17.839 ms  17.750 ms  17.743 ms
10  bom12s09-in-f4.1e100.net (142.250.76.164)  16.249 ms  16.246 ms  16.244 ms
```

- First some DNS Resolution happens

```
16:29:37.939300 IP 172.22.236.145.47873 > 172.22.224.1.53: 25186+ A?
www.google.com. (32)
16:29:37.939336 IP 172.22.236.145.47873 > 172.22.224.1.53: 30318+ AAAA?
www.google.com. (32)
16:29:37.940126 IP 172.22.224.1.53 > 172.22.236.145.47873: 25186- 1/0/0 A
142.250.76.164 (62)
16:29:37.950546 IP 172.22.224.1.53 > 172.22.236.145.47873: 30318- 1/0/0
AAAA 2404:6800:4009:813::2004 (74)
```

- Then many ICMP echo requests are sent

```
16:29:37.951268 IP 172.22.236.145 > 142.250.76.164: ICMP echo request, id
4000, seq 1, length 40
16:29:37.951294 IP 172.22.236.145 > 142.250.76.164: ICMP echo request, id
4000, seq 2, length 40
16:29:37.951298 IP 172.22.236.145 > 142.250.76.164: ICMP echo request, id
4000, seq 3, length 40
16:29:37.951300 IP 172.22.236.145 > 142.250.76.164: ICMP echo request, id
4000, seq 4, length 40
16:29:37.951303 IP 172.22.236.145 > 142.250.76.164: ICMP echo request, id
4000, seq 5, length 40
16:29:37.951305 IP 172.22.236.145 > 142.250.76.164: ICMP echo request, id
4000, seq 6, length 40
...
```

- We then recieve some ICMP time exceeded in-transit, which shows us the working of traceroute

```
16:29:37.951785 IP 172.22.224.1 > 172.22.236.145: ICMP time exceeded in-
transit, length 68
16:29:37.951846 IP 172.22.224.1 > 172.22.236.145: ICMP time exceeded in-
transit, length 68
16:29:37.951846 IP 172.22.224.1 > 172.22.236.145: ICMP time exceeded in-
transit, length 68
...
```

Question (i)

Write the `tcpdump` command that captures packets containing TCP packets with a specific IP address as (i) both source and destination, (ii) only source, and (iii) only destination.

Answer

- My IP: 172.22.236.145
- Both Source and Destination
  - `sudo tcpdump -n src 172.22.236.145 and dst www.google.com`
- Only Source
  - `sudo tcpdump -n src www.google.com`
- Only Destination
  - `sudo tcpdump -n dst 172.22.236.145`

Question (j)

Write the `tcpdump` command that captures packets containing ICMP packets between two hosts with different IP addresses.

Answer

- `sudo tcpdump -n icmp -c 5`
- we specify the icmp protocol as a filter in `tcpdump`
- ```
  [RAM: 18% | SWAP: 0%] .../Sem 6/CS 3272 Computer Network Lab/Assignment 2 22s
  [Batt: 79%][04:52 PM] ❯ sudo tcpdump -n icmp -c 5
  tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
  listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
  16:53:13.119933 IP 172.22.236.145 > 142.250.76.164: ICMP echo request, id 6661, seq 1, length 64
  16:53:13.128235 IP 142.250.76.164 > 172.22.236.145: ICMP echo reply, id 6661, seq 1, length 64
  16:53:14.121581 IP 172.22.236.145 > 142.250.76.164: ICMP echo request, id 6661, seq 2, length 64
  16:53:14.134128 IP 142.250.76.164 > 172.22.236.145: ICMP echo reply, id 6661, seq 2, length 64
  16:53:15.123402 IP 172.22.236.145 > 142.250.76.164: ICMP echo request, id 6661, seq 3, length 64
  5 packets captured
  5 packets received by filter
  0 packets dropped by kernel
  ```
  ```

Question (k)

Write the `tcpdump` command to capture packets containing SSH request and reply between two specific IP addresses (hint: use port number 22 for SSH).

Answer

- `sudo tcpdump -n port 22`