20MCA136-NETWORKING & ADMINISTRATION

Analyzing network packet stream using tcpdump

SUBMITTED BY,

VIVIN V. ABRAHAM

R MCA-2020-S2

ROLL NO : 42

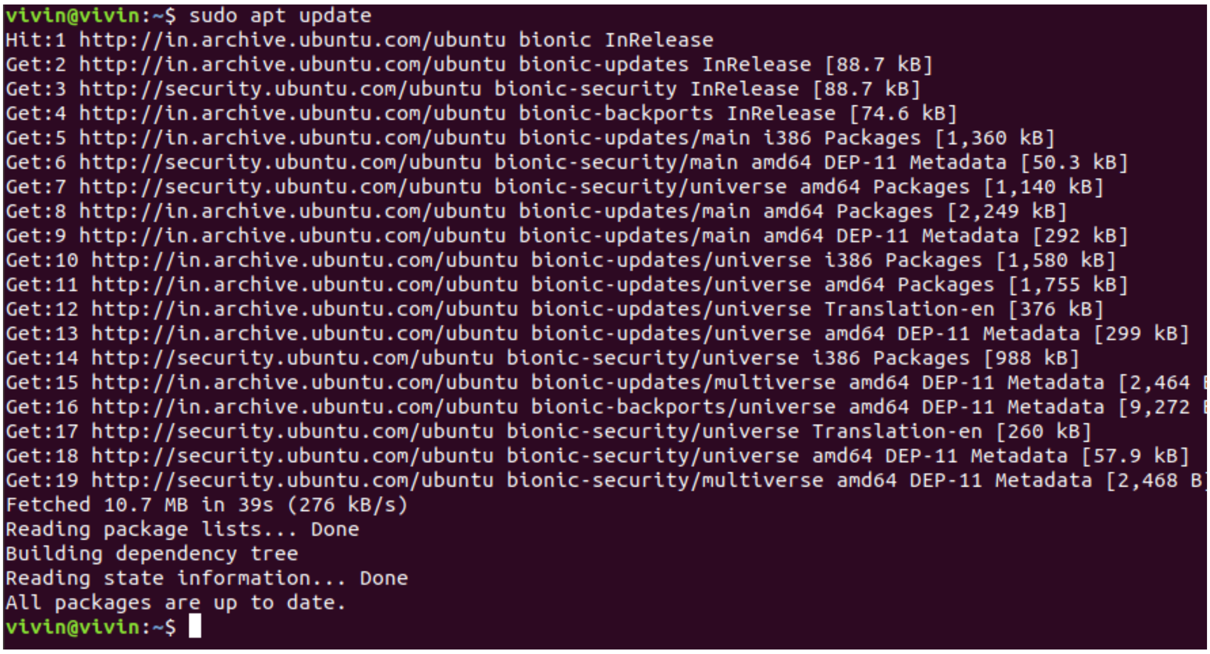
SUBMITTED TO ,

MEERA MISS

Tcpdump Installation

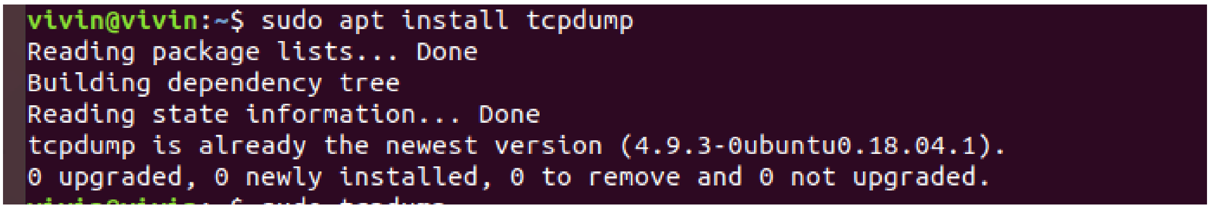
On Debian based distributions tcpdump can be installed with the APT command:

sudo apt update



```
vivin@vivin:~$ sudo apt update
Hit:1 http://in.archive.ubuntu.com/ubuntu bionic InRelease
Get:2 http://in.archive.ubuntu.com/ubuntu bionic-updates InRelease [88.7 kB]
Get:3 http://security.ubuntu.com/ubuntu bionic-security InRelease [88.7 kB]
Get:4 http://in.archive.ubuntu.com/ubuntu bionic-backports InRelease [74.6 kB]
Get:5 http://in.archive.ubuntu.com/ubuntu bionic-updates/main i386 Packages [1,360 kB]
Get:6 http://security.ubuntu.com/ubuntu bionic-security/main amd64 DEP-11 Metadata [50.3 kB]
Get:7 http://security.ubuntu.com/ubuntu bionic-security/universe amd64 Packages [1,140 kB]
Get:8 http://in.archive.ubuntu.com/ubuntu bionic-updates/main amd64 Packages [2,249 kB]
Get:9 http://in.archive.ubuntu.com/ubuntu bionic-updates/main amd64 DEP-11 Metadata [292 kB]
Get:10 http://in.archive.ubuntu.com/ubuntu bionic-updates/universe i386 Packages [1,580 kB]
Get:11 http://in.archive.ubuntu.com/ubuntu bionic-updates/universe amd64 Packages [1,755 kB]
Get:12 http://in.archive.ubuntu.com/ubuntu bionic-updates/universe Translation-en [376 kB]
Get:13 http://in.archive.ubuntu.com/ubuntu bionic-updates/universe amd64 DEP-11 Metadata [299 kB]
Get:14 http://security.ubuntu.com/ubuntu bionic-security/universe i386 Packages [988 kB]
Get:15 http://in.archive.ubuntu.com/ubuntu bionic-updates/multiverse amd64 DEP-11 Metadata [2,464 B
Get:16 http://in.archive.ubuntu.com/ubuntu bionic-backports/universe amd64 DEP-11 Metadata [9,272 B
Get:17 http://security.ubuntu.com/ubuntu bionic-security/universe Translation-en [260 kB]
Get:18 http://security.ubuntu.com/ubuntu bionic-security/universe amd64 DEP-11 Metadata [57.9 kB]
Get:19 http://security.ubuntu.com/ubuntu bionic-security/multiverse amd64 DEP-11 Metadata [2,468 B]
Fetched 10.7 MB in 39s (276 kB/s)
Reading package lists... Done
Building dependency tree
Reading state information... Done
All packages are up to date.
vivin@vivin:~$ 
```

sudo apt install tcpdump



```
vivin@vivin:~$ sudo apt install tcpdump
Reading package lists... Done
Building dependency tree
Reading state information... Done
tcpdump is already the newest version (4.9.3-0ubuntu0.18.04.1).
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
vivin@vivin:~$ sudo tcpdump
```

Capturing Packets with tcpdump

•The general syntax for the tcpdumpcommand is as follows:

tcpdump[options] [expression]

•The command options allow you to control the behavior of the command.

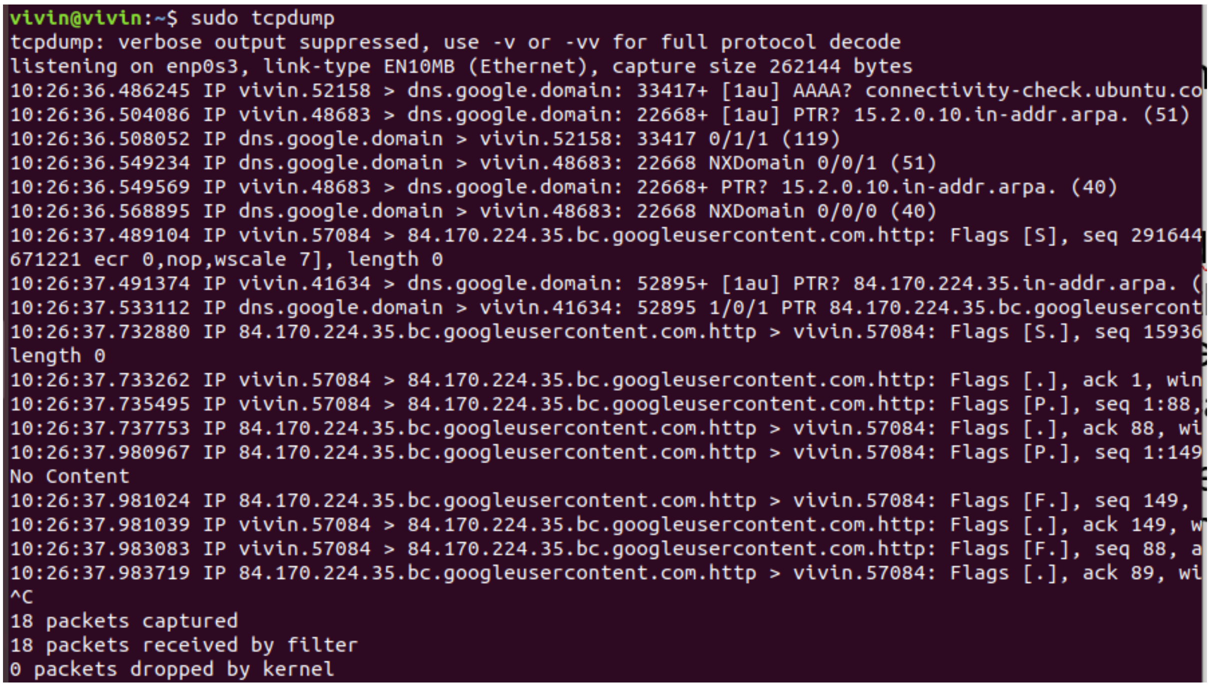
•The filter expression defines which packets will be captured.

•Only root or user with sudo privileges can run tcpdump. If you try to run the command as an unprivileged user, you’ll get an error saying: “You don’t have permission to capture on that device”.

•The most simple use case is to invoke tcpdump without any options and filters:

•**sudo tcpdump**

•tcpdumpwill continue to capture packets and write to the standard output until it receives an interrupt signal. Use the Ctrl+C key combination to send an interrupt signal and stop the command.

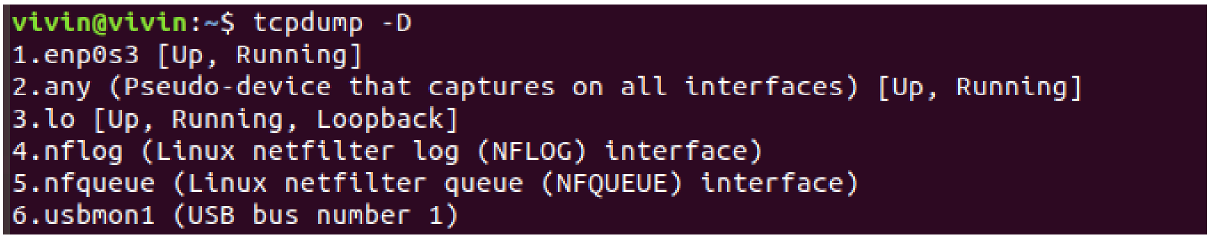


```
vivin@vivin:~$ sudo tcpdump
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on enp0s3, link-type EN10MB (Ethernet), capture size 262144 bytes
10:26:36.486245 IP vivin.52158 > dns.google.domain: 33417+ [1au] AAAA? connectivity-check.ubuntu.co
10:26:36.504086 IP vivin.48683 > dns.google.domain: 22668+ [1au] PTR? 15.2.0.10.in-addr.arpa. (51)
10:26:36.508052 IP dns.google.domain > vivin.52158: 33417 0/1/1 (119)
10:26:36.549234 IP dns.google.domain > vivin.48683: 22668 NXDomain 0/0/1 (51)
10:26:36.549569 IP vivin.48683 > dns.google.domain: 22668+ PTR? 15.2.0.10.in-addr.arpa. (40)
10:26:36.568895 IP dns.google.domain > vivin.48683: 22668 NXDomain 0/0/0 (40)
10:26:37.489104 IP vivin.57084 > 84.170.224.35.bc.googleusercontent.com.http: Flags [S], seq 291644
671221 ecr 0,nop,wscale 7], length 0
10:26:37.491374 IP vivin.41634 > dns.google.domain: 52895+ [1au] PTR? 84.170.224.35.in-addr.arpa. (
10:26:37.533112 IP dns.google.domain > vivin.41634: 52895 1/0/1 PTR 84.170.224.35.bc.googleusercont
10:26:37.732880 IP 84.170.224.35.bc.googleusercontent.com.http > vivin.57084: Flags [S.], seq 15936
length 0
10:26:37.733262 IP vivin.57084 > 84.170.224.35.bc.googleusercontent.com.http: Flags [.], ack 1, win
10:26:37.735495 IP vivin.57084 > 84.170.224.35.bc.googleusercontent.com.http: Flags [P.], seq 1:88,
10:26:37.737753 IP 84.170.224.35.bc.googleusercontent.com.http > vivin.57084: Flags [.], ack 88, wi
10:26:37.980967 IP 84.170.224.35.bc.googleusercontent.com.http > vivin.57084: Flags [P.], seq 1:149
No Content
10:26:37.981024 IP 84.170.224.35.bc.googleusercontent.com.http > vivin.57084: Flags [F.], seq 149,
10:26:37.981039 IP vivin.57084 > 84.170.224.35.bc.googleusercontent.com.http: Flags [.], ack 149, w
10:26:37.983083 IP vivin.57084 > 84.170.224.35.bc.googleusercontent.com.http: Flags [F.], seq 88, a
10:26:37.983719 IP 84.170.224.35.bc.googleusercontent.com.http > vivin.57084: Flags [.], ack 89, wi
^C
18 packets captured
18 packets received by filter
0 packets dropped by kernel
```

tcpdump command options

You need to berootto run tcpdump. It includes many options and filters. Running tcp dump without any options will capture all packets flowing through the default interface.
To see the list of network interfaces available on the system and on which tcpdump can capture packets.

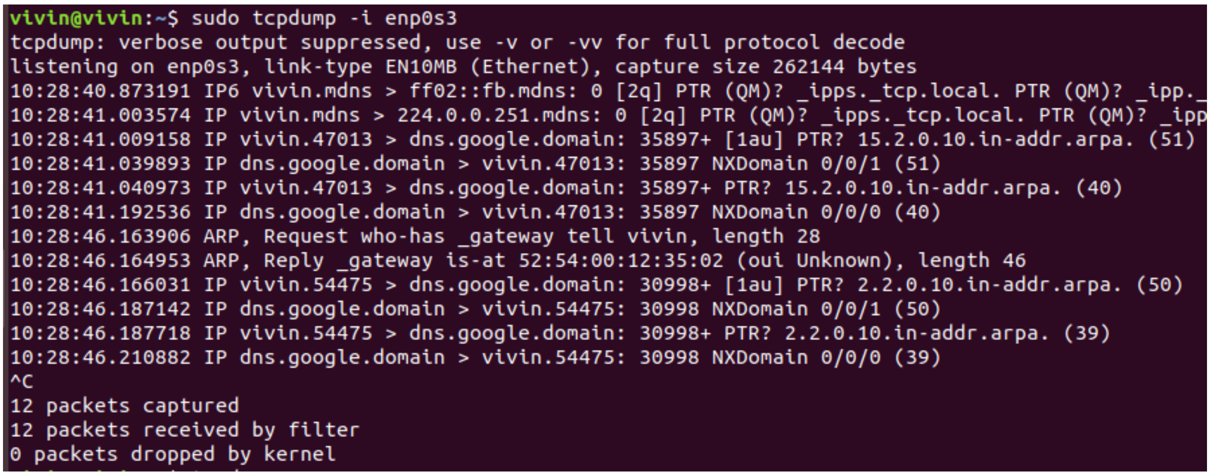
tcpdump –D



```
vivin@vivin:~$ tcpdump -D
1.enp0s3 [Up, Running]
2.any (Pseudo-device that captures on all interfaces) [Up, Running]
3.lo [Up, Running, Loopback]
4.nflog (Linux netfilter log (NFLOG) interface)
5.nfqueue (Linux netfilter queue (NFQUEUE) interface)
6.usbmon1 (USB bus number 1)
```

To capture packets flowing through a specific interface, use the -iflag with the interface name. Without the -i interface tcpdump will pick up the first network interface it comes across.

sudo tcpdump -i enp0s3



```
vivin@vivin:~$ sudo tcpdump -i enp0s3
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on enp0s3, link-type EN10MB (Ethernet), capture size 262144 bytes
10:28:40.873191 IP6 vivin.mdns > ff02::fb.mdns: 0 [2q] PTR (QM)? _ipps._tcp.local. PTR (QM)? _ipp._
10:28:41.003574 IP vivin.mdns > 224.0.0.251.mdns: 0 [2q] PTR (QM)? _ipps._tcp.local. PTR (QM)? _ipp
10:28:41.009158 IP vivin.47013 > dns.google.domain: 35897+ [1au] PTR? 15.2.0.10.in-addr.arpa. (51)
10:28:41.039893 IP dns.google.domain > vivin.47013: 35897 NXDomain 0/0/1 (51)
10:28:41.040973 IP vivin.47013 > dns.google.domain: 35897+ PTR? 15.2.0.10.in-addr.arpa. (40)
10:28:41.192536 IP dns.google.domain > vivin.47013: 35897 NXDomain 0/0/0 (40)
10:28:46.163906 ARP, Request who-has _gateway tell vivin, length 28
10:28:46.164953 ARP, Reply _gateway is-at 52:54:00:12:35:02 (oui Unknown), length 46
10:28:46.166031 IP vivin.54475 > dns.google.domain: 30998+ [1au] PTR? 2.2.0.10.in-addr.arpa. (50)
10:28:46.187142 IP dns.google.domain > vivin.54475: 30998 NXDomain 0/0/1 (50)
10:28:46.187718 IP vivin.54475 > dns.google.domain: 30998+ PTR? 2.2.0.10.in-addr.arpa. (39)
10:28:46.210882 IP dns.google.domain > vivin.54475: 30998 NXDomain 0/0/0 (39)
^C
12 packets captured
12 packets received by filter
0 packets dropped by kernel
```

tcpdump-ienp2s0

Now your devices use the "Predictable Interface Names", which are based onNames incorporating Firmware/BIOS provided index numbers for on-board devices (example: eno1)

Names incorporating Firmware/BIOS provided PCI Express hotplugslot index numbers (example: ens1)

Names incorporating physical/geographical location of the connector of the hardware (example: enp2s0)

Names incorporating the interfaces'sMAC address (example: enx78e7d1ea46da)

Classic, unpredictable kernel-native ethXnaming (example: eth0)
•tcpdumpwill continue to capture packets and write to the standard output until it receives an interrupt signal.

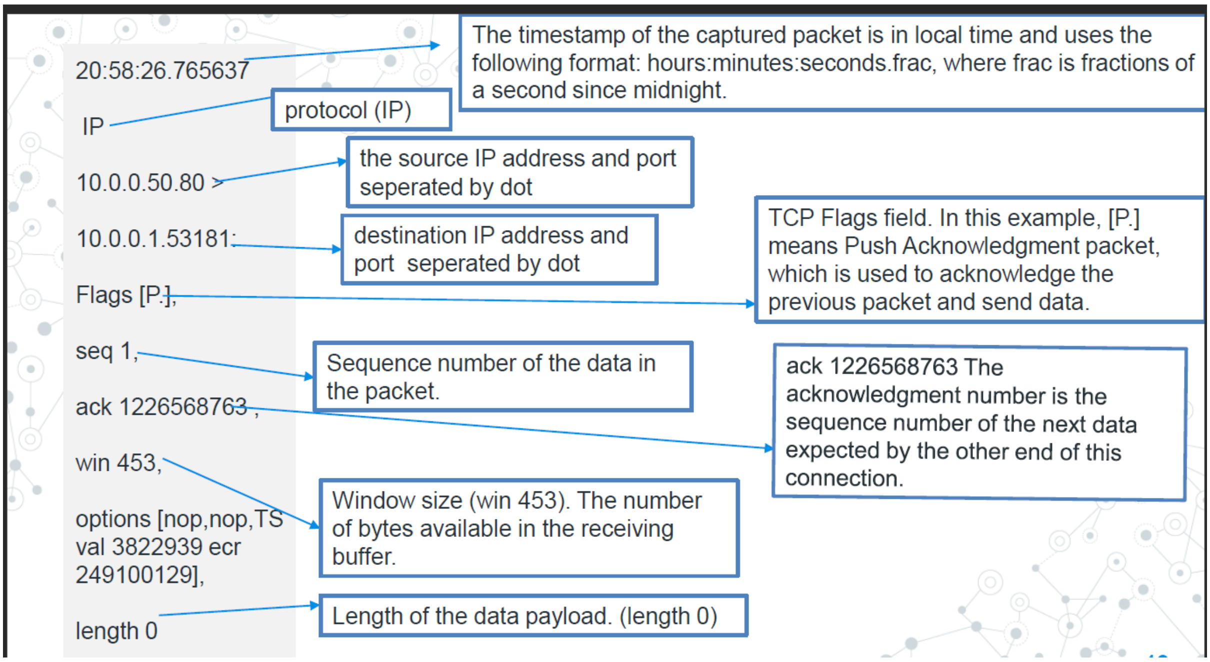
•Use the Ctrl+Ckey combination to send an interrupt signal and stop the command.

tcpdump output look like

20:58:26.765637 IP 10.0.0.50.80 > 10.0.0.1.53181: Flags [F.], seq 1, ack 2, win 453, options [nop,nop,TS val 3822939 ecr 249100129], length 0

tcpdumpoutputs information for each captured packet on a new line. Each line includes a timestamp and information about that packet, depending on the protocol.

•The typical format of a TCP protocol line is as follows:
[Timestamp] [Protocol] [SrcIP].[SrcPort] > [DstIP].[DstPort]: [Flags], [Seq], [Ack], [Win Size], [Options], [Data Length]



20:58:26.765637

The timestamp of the captured packet is in local time and uses the following format: hours:minutes:seconds.frac, where frac is fractions of a second since midnight.

IP

protocol (IP)

10.0.0.50.80 >

the source IP address and port seperated by dot

10.0.0.1.53181:

destination IP address and port seperated by dot

Flags [P.],

TCP Flags field. In this example, [P.] means Push Acknowledgment packet, which is used to acknowledge the previous packet and send data.

seq 1,

Sequence number of the data in the packet.

ack 1226568763,

ack 1226568763 The acknowledgment number is the sequence number of the next data expected by the other end of this connection.

win 453,

Window size (win 453). The number of bytes available in the receiving buffer.

options [nop,nop,TS val 3822939 ecr 249100129],

length 0

Length of the data payload. (length 0)

seq201747193:201747301 -The sequence number is in the first: lastnotation. It shows the number of data contained in the packet. Except for the first packet in the data stream where these numbers are absolute, all subsequent packets use as relative byte positions. In this example, the number is 201747193:201747301, meaning that this packet contains bytes 201747193 to 201747301 of the data stream. Use the -S option to print absolute sequence numbers.

•TCP Flags field. In this example, [P.] means Push Acknowledgment packet, which is used to acknowledge the previous packet and send data. This field can have the following values :

•S –SYN. The first step in establishing the connection.

•F –FIN. Connection termination.

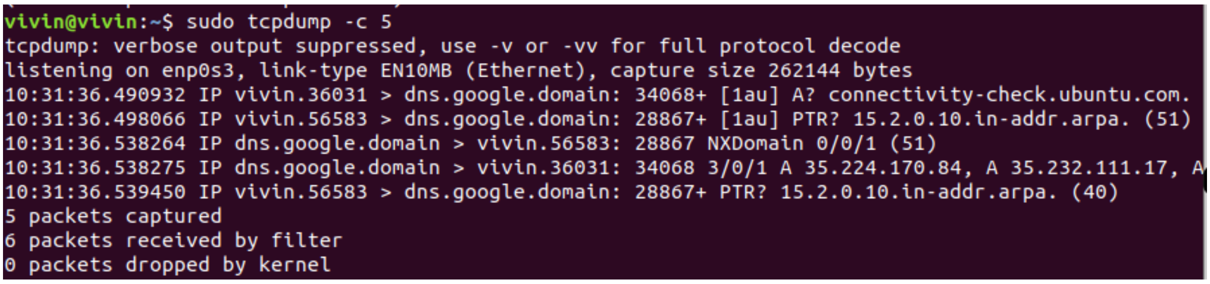
•. –ACK. Acknowledgment packet received successfully.

•P –PUSH. Tells the receiver to process packets instead of buffering them.

•R –RST. Communication stopped.

To capture only a set of lines, say 5, use the -c flag:

sudo tcpdump -c 5



```
vivin@vivin:~$ sudo tcpdump -c 5
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on enp0s3, link-type EN10MB (Ethernet), capture size 262144 bytes
10:31:36.490932 IP vivin.36031 > dns.google.domain: 34068+ [1au] A? connectivity-check.ubuntu.com.
10:31:36.498066 IP vivin.56583 > dns.google.domain: 28867+ [1au] PTR? 15.2.0.10.in-addr.arpa. (51)
10:31:36.538264 IP dns.google.domain > vivin.56583: 28867 NXDomain 0/0/1 (51)
10:31:36.538275 IP dns.google.domain > vivin.36031: 34068 3/0/1 A 35.224.170.84, A 35.232.111.17, A
10:31:36.539450 IP vivin.56583 > dns.google.domain: 28867+ PTR? 15.2.0.10.in-addr.arpa. (40)
5 packets captured
6 packets received by filter
0 packets dropped by kernel
```

tcpdump filter expressions

Filter expressions select which packet headers will be displayed.

If no filters are applied, all packet headers are displayed.

Commonly used filters are port, host, src, dst, tcp, udp, icmp.

Filters are one of the most powerful features of the tcpdumpcommand.

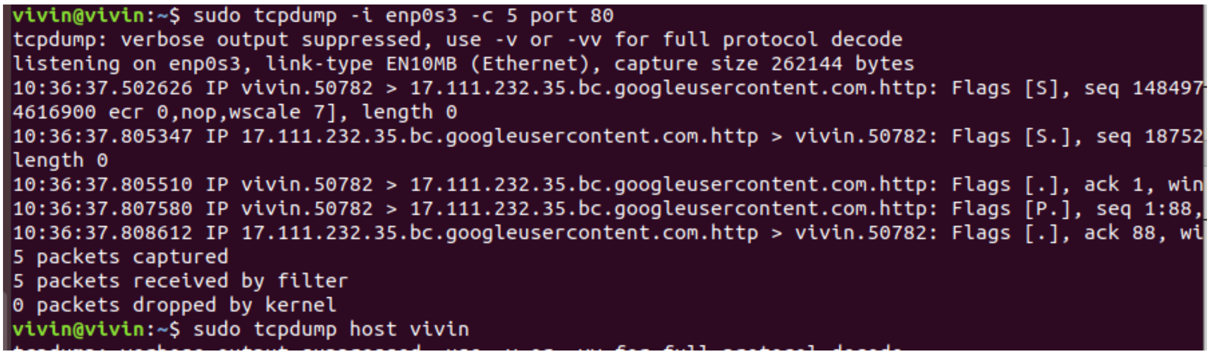
They allow you to capture only those packets matching the expression.

For example, when troubleshooting issues related to a webserver, you can use filters to obtain only the HTTP traffic.

port filter

Use port filter to view packets arriving at a specific port:

sudo tcpdump -i enp0s3 -c 5 port 80.

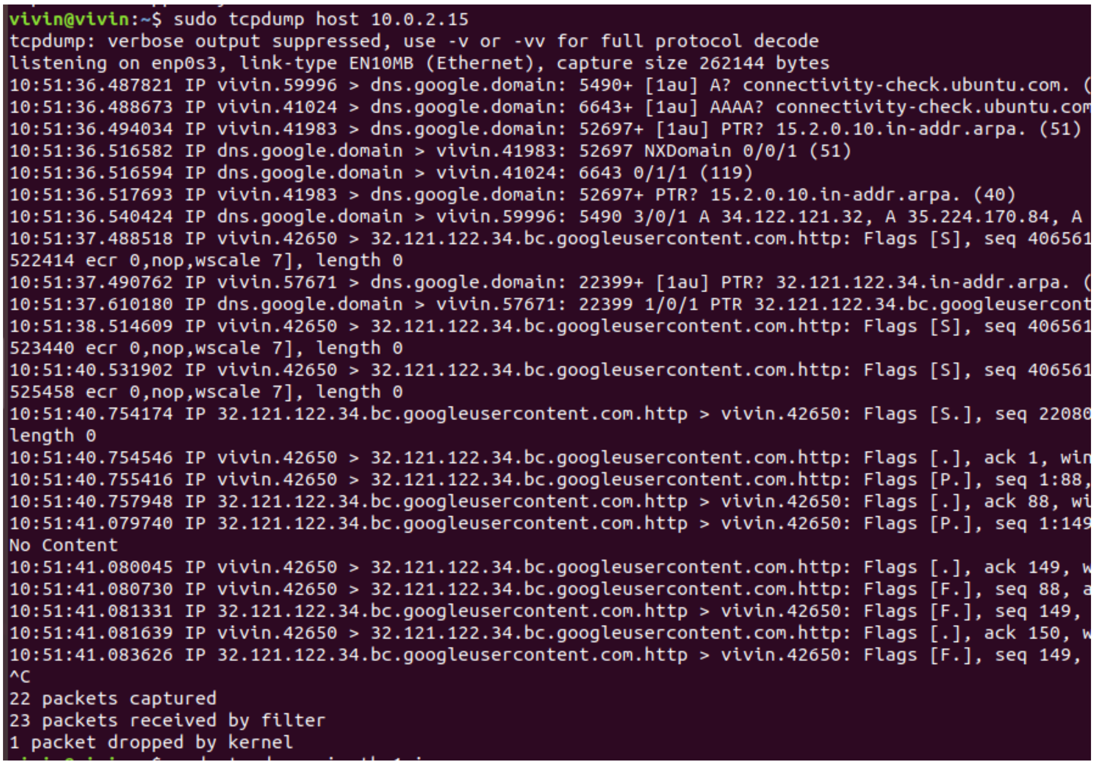


```
vivin@vivin:~$ sudo tcpdump -i enp0s3 -c 5 port 80
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on enp0s3, link-type EN10MB (Ethernet), capture size 262144 bytes
10:36:37.502626 IP vivin.50782 > 17.111.232.35.bc.googleusercontent.com.http: Flags [S], seq 148497
4616900 ecr 0,nop,wscale 7], length 0
10:36:37.805347 IP 17.111.232.35.bc.googleusercontent.com.http > vivin.50782: Flags [S.], seq 18752
length 0
10:36:37.805510 IP vivin.50782 > 17.111.232.35.bc.googleusercontent.com.http: Flags [.], ack 1, win
10:36:37.807580 IP vivin.50782 > 17.111.232.35.bc.googleusercontent.com.http: Flags [P.], seq 1:88,
10:36:37.808612 IP 17.111.232.35.bc.googleusercontent.com.http > vivin.50782: Flags [.], ack 88, wi
5 packets captured
5 packets received by filter
0 packets dropped by kernel
vivin@vivin:~$ sudo tcpdump host vivin
```

host filter

To capture all packets arriving at or leaving from the host with IP address of 10.0.2.15:

sudo tcpdump host 10.0.2.15

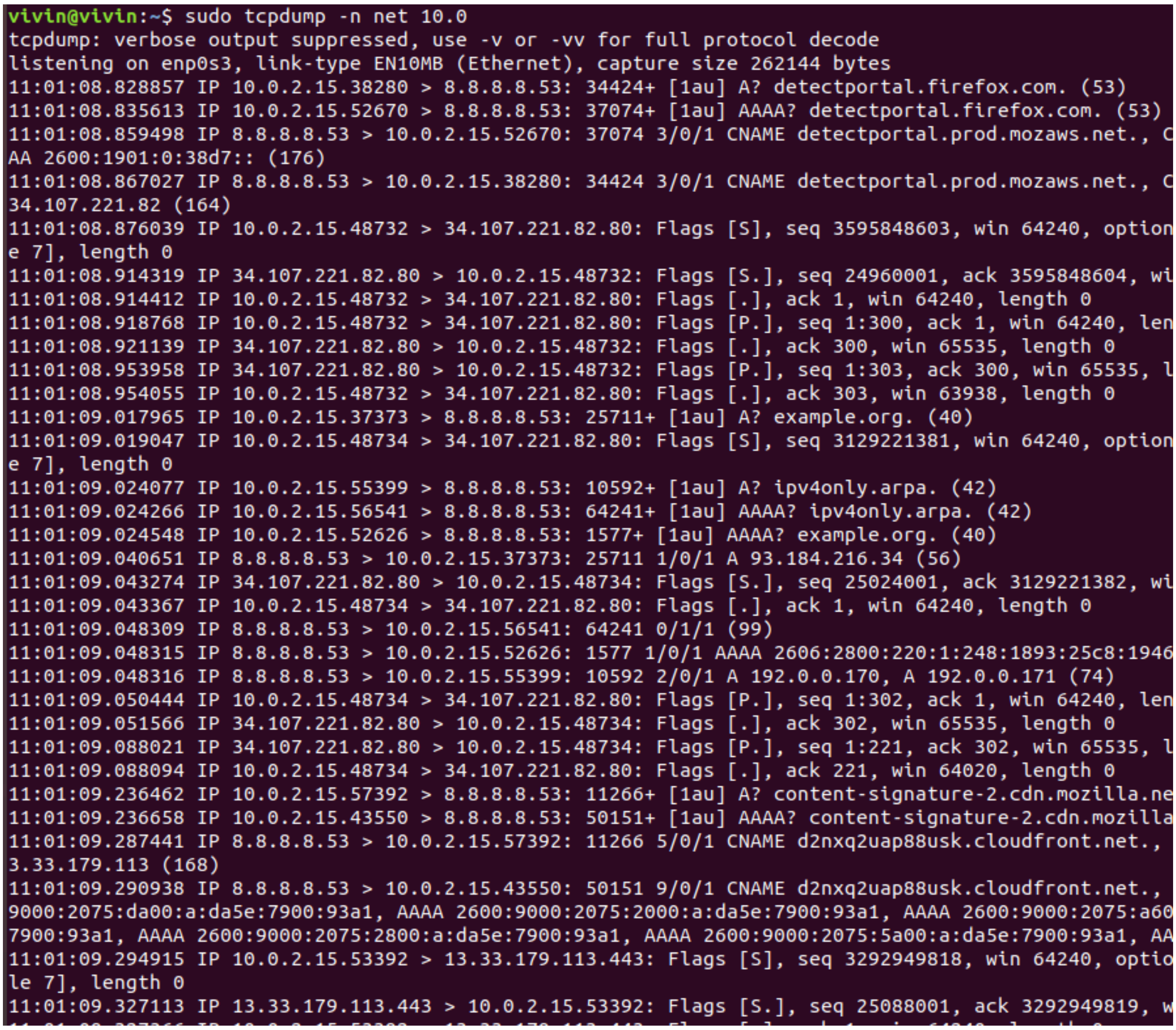


```
vivin@vivin:~$ sudo tcpdump host 10.0.2.15
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on enp0s3, link-type EN10MB (Ethernet), capture size 262144 bytes
10:51:36.487821 IP vivin.59996 > dns.google.domain: 5490+ [1au] A? connectivity-check.ubuntu.com. (
10:51:36.488673 IP vivin.41024 > dns.google.domain: 6643+ [1au] AAAA? connectivity-check.ubuntu.com
10:51:36.494034 IP vivin.41983 > dns.google.domain: 52697+ [1au] PTR? 15.2.0.10.in-addr.arpa. (51)
10:51:36.516582 IP dns.google.domain > vivin.41983: 52697 NXDomain 0/0/1 (51)
10:51:36.516594 IP dns.google.domain > vivin.41024: 6643 0/1/1 (119)
10:51:36.517693 IP vivin.41983 > dns.google.domain: 52697+ PTR? 15.2.0.10.in-addr.arpa. (40)
10:51:36.540424 IP dns.google.domain > vivin.59996: 5490 3/0/1 A 34.122.121.32, A 35.224.170.84, A
10:51:37.488518 IP vivin.42650 > 32.121.122.34.bc.googleusercontent.com.http: Flags [S], seq 406561
522414 ecr 0,nop,wscale 7], length 0
10:51:37.490762 IP vivin.57671 > dns.google.domain: 22399+ [1au] PTR? 32.121.122.34.in-addr.arpa. (
10:51:37.610180 IP dns.google.domain > vivin.57671: 22399 1/0/1 PTR 32.121.122.34.bc.googleusercont
10:51:38.514609 IP vivin.42650 > 32.121.122.34.bc.googleusercontent.com.http: Flags [S], seq 406561
523440 ecr 0,nop,wscale 7], length 0
10:51:40.531902 IP vivin.42650 > 32.121.122.34.bc.googleusercontent.com.http: Flags [S], seq 406561
525458 ecr 0,nop,wscale 7], length 0
10:51:40.754174 IP 32.121.122.34.bc.googleusercontent.com.http > vivin.42650: Flags [S.], seq 22080
length 0
10:51:40.754546 IP vivin.42650 > 32.121.122.34.bc.googleusercontent.com.http: Flags [.], ack 1, win
10:51:40.755416 IP vivin.42650 > 32.121.122.34.bc.googleusercontent.com.http: Flags [P.], seq 1:88,
10:51:40.757948 IP 32.121.122.34.bc.googleusercontent.com.http > vivin.42650: Flags [.], ack 88, wi
10:51:41.079740 IP 32.121.122.34.bc.googleusercontent.com.http > vivin.42650: Flags [P.], seq 1:149
No Content
10:51:41.080045 IP vivin.42650 > 32.121.122.34.bc.googleusercontent.com.http: Flags [.], ack 149, w
10:51:41.080730 IP vivin.42650 > 32.121.122.34.bc.googleusercontent.com.http: Flags [F.], seq 88, a
10:51:41.081331 IP 32.121.122.34.bc.googleusercontent.com.http > vivin.42650: Flags [F.], seq 149,
10:51:41.081639 IP vivin.42650 > 32.121.122.34.bc.googleusercontent.com.http: Flags [.], ack 150, w
10:51:41.083626 IP 32.121.122.34.bc.googleusercontent.com.http > vivin.42650: Flags [F.], seq 149,
^C
22 packets captured
23 packets received by filter
1 packet dropped by kernel
```

To capture packets of a specific protocol type, for example, icmp, on eth1 interface:

(tcpdump-ieth1 icmp)

sudo tcpdump -n net 10.10



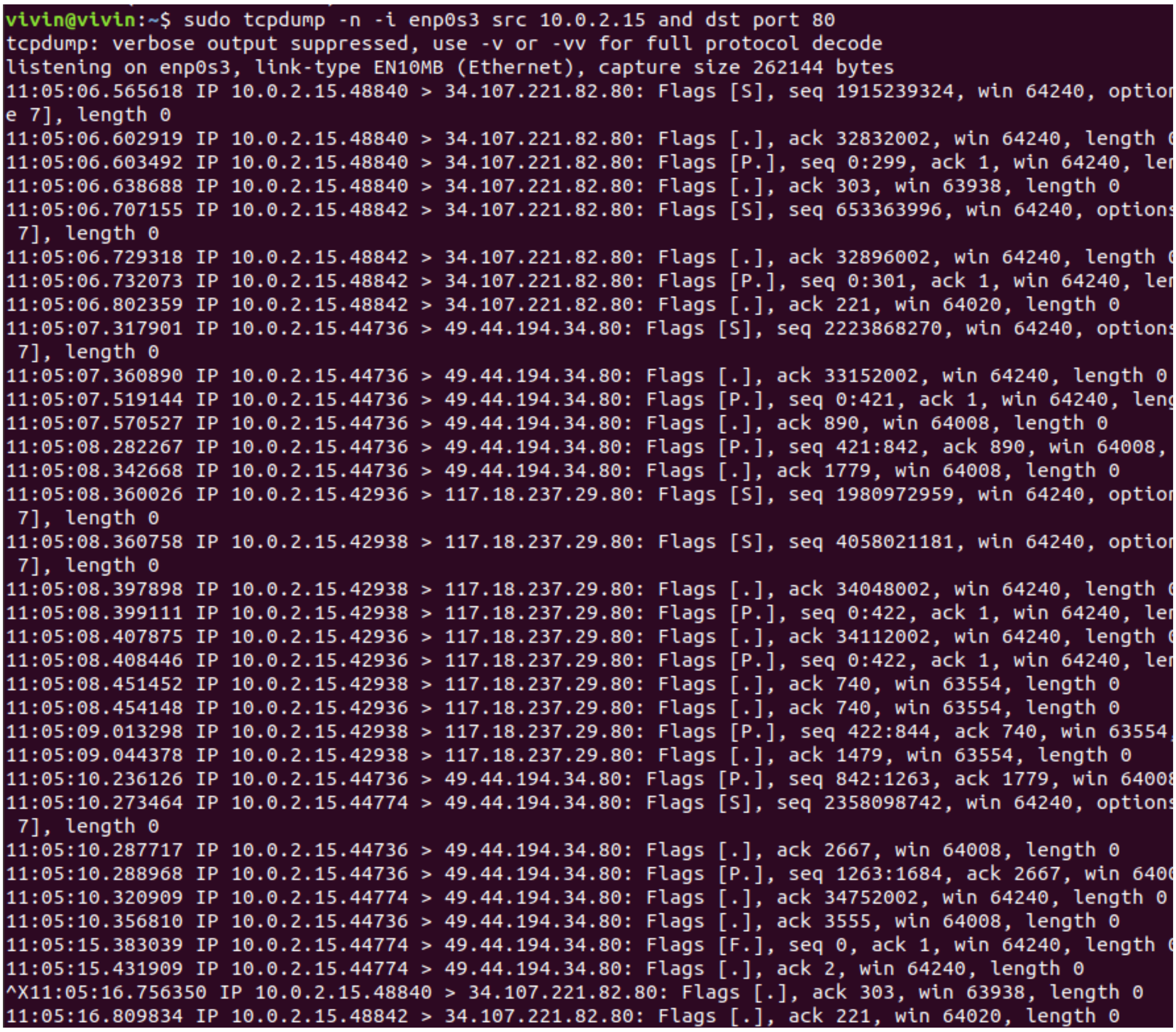
```
vivin@vivin:~$ sudo tcpdump -n net 10.0
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on enp0s3, link-type EN10MB (Ethernet), capture size 262144 bytes
11:01:08.828857 IP 10.0.2.15.38280 > 8.8.8.8.53: 34424+ [1au] A? detectportal.firefox.com. (53)
11:01:08.835613 IP 10.0.2.15.52670 > 8.8.8.8.53: 37074+ [1au] AAAA? detectportal.firefox.com. (53)
11:01:08.859498 IP 8.8.8.8.53 > 10.0.2.15.52670: 37074 3/0/1 CNAME detectportal.prod.mozaws.net., C
AA 2600:1901:0:38d7:: (176)
11:01:08.867027 IP 8.8.8.8.53 > 10.0.2.15.38280: 34424 3/0/1 CNAME detectportal.prod.mozaws.net., C
34.107.221.82 (164)
11:01:08.876039 IP 10.0.2.15.48732 > 34.107.221.82.80: Flags [S], seq 3595848603, win 64240, option
e 7], length 0
11:01:08.914319 IP 34.107.221.82.80 > 10.0.2.15.48732: Flags [S.], seq 24960001, ack 3595848604, wi
11:01:08.914412 IP 10.0.2.15.48732 > 34.107.221.82.80: Flags [.], ack 1, win 64240, length 0
11:01:08.918768 IP 10.0.2.15.48732 > 34.107.221.82.80: Flags [P.], seq 1:300, ack 1, win 64240, len
11:01:08.921139 IP 34.107.221.82.80 > 10.0.2.15.48732: Flags [.], ack 300, win 65535, length 0
11:01:08.953958 IP 34.107.221.82.80 > 10.0.2.15.48732: Flags [P.], seq 1:303, ack 300, win 65535, l
11:01:08.954055 IP 10.0.2.15.48732 > 34.107.221.82.80: Flags [.], ack 303, win 63938, length 0
11:01:09.017965 IP 10.0.2.15.37373 > 8.8.8.8.53: 25711+ [1au] A? example.org. (40)
11:01:09.019047 IP 10.0.2.15.48734 > 34.107.221.82.80: Flags [S], seq 3129221381, win 64240, option
e 7], length 0
11:01:09.024077 IP 10.0.2.15.55399 > 8.8.8.8.53: 10592+ [1au] A? ipv4only.arpa. (42)
11:01:09.024266 IP 10.0.2.15.56541 > 8.8.8.8.53: 64241+ [1au] AAAA? ipv4only.arpa. (42)
11:01:09.024548 IP 10.0.2.15.52626 > 8.8.8.8.53: 1577+ [1au] AAAA? example.org. (40)
11:01:09.040651 IP 8.8.8.8.53 > 10.0.2.15.37373: 25711 1/0/1 A 93.184.216.34 (56)
11:01:09.043274 IP 34.107.221.82.80 > 10.0.2.15.48734: Flags [S.], seq 25024001, ack 3129221382, wi
11:01:09.043367 IP 10.0.2.15.48734 > 34.107.221.82.80: Flags [.], ack 1, win 64240, length 0
11:01:09.048309 IP 8.8.8.8.53 > 10.0.2.15.56541: 64241 0/1/1 (99)
11:01:09.048315 IP 8.8.8.8.53 > 10.0.2.15.52626: 1577 1/0/1 AAAA 2606:2800:220:1:248:1893:25c8:1946
11:01:09.048316 IP 8.8.8.8.53 > 10.0.2.15.55399: 10592 2/0/1 A 192.0.0.170, A 192.0.0.171 (74)
11:01:09.050444 IP 10.0.2.15.48734 > 34.107.221.82.80: Flags [P.], seq 1:302, ack 1, win 64240, len
11:01:09.051566 IP 34.107.221.82.80 > 10.0.2.15.48734: Flags [.], ack 302, win 65535, length 0
11:01:09.088021 IP 34.107.221.82.80 > 10.0.2.15.48734: Flags [P.], seq 1:221, ack 302, win 65535, l
11:01:09.088094 IP 10.0.2.15.48734 > 34.107.221.82.80: Flags [.], ack 221, win 64020, length 0
11:01:09.236462 IP 10.0.2.15.57392 > 8.8.8.8.53: 11266+ [1au] A? content-signature-2.cdn.mozilla.ne
11:01:09.236658 IP 10.0.2.15.43550 > 8.8.8.8.53: 50151+ [1au] AAAA? content-signature-2.cdn.mozilla
11:01:09.287441 IP 8.8.8.8.53 > 10.0.2.15.57392: 11266 5/0/1 CNAME d2nxq2uap88usk.cloudfront.net.,
3.33.179.113 (168)
11:01:09.290938 IP 8.8.8.8.53 > 10.0.2.15.43550: 50151 9/0/1 CNAME d2nxq2uap88usk.cloudfront.net.,
9000:2075:da00:a:da5e:7900:93a1, AAAA 2600:9000:2075:2000:a:da5e:7900:93a1, AAAA 2600:9000:2075:a60
7900:93a1, AAAA 2600:9000:2075:2800:a:da5e:7900:93a1, AAAA 2600:9000:2075:5a00:a:da5e:7900:93a1, AA
11:01:09.294915 IP 10.0.2.15.53392 > 13.33.179.113.443: Flags [S], seq 3292949818, win 64240, optio
le 7], length 0
11:01:09.327113 IP 13.33.179.113.443 > 10.0.2.15.53392: Flags [S.], seq 25088001, ack 3292949819, w
```

Combining filter expressions

We can combine filter expressions with AND, OR, and NOT operators. This will enable you to write commands which can isolate packets more precisely:

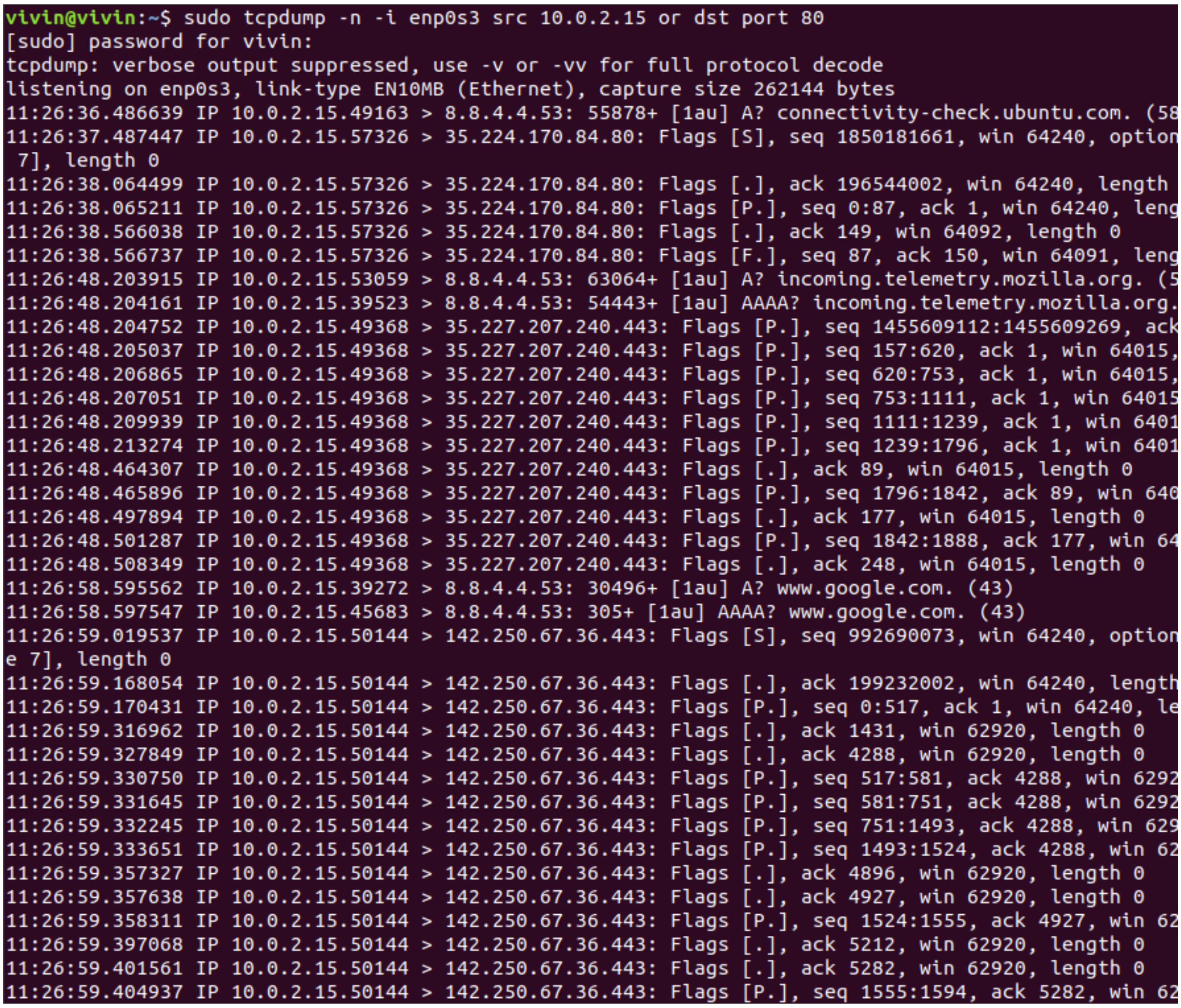
Packets from a specific IP and destined for a specific port:

sudo tcpdump-n -i enp0s3 src 10.0.2.15 and dst port 80



```
vivin@vivin:~$ sudo tcpdump -n -i enp0s3 src 10.0.2.15 and dst port 80
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on enp0s3, link-type EN10MB (Ethernet), capture size 262144 bytes
11:05:06.565618 IP 10.0.2.15.48840 > 34.107.221.82.80: Flags [S], seq 1915239324, win 64240, option
e 7], length 0
11:05:06.602919 IP 10.0.2.15.48840 > 34.107.221.82.80: Flags [.], ack 32832002, win 64240, length
11:05:06.603492 IP 10.0.2.15.48840 > 34.107.221.82.80: Flags [P.], seq 0:299, ack 1, win 64240, le
11:05:06.638688 IP 10.0.2.15.48840 > 34.107.221.82.80: Flags [.], ack 303, win 63938, length 0
11:05:06.707155 IP 10.0.2.15.48842 > 34.107.221.82.80: Flags [S], seq 653363996, win 64240, options
 7], length 0
11:05:06.729318 IP 10.0.2.15.48842 > 34.107.221.82.80: Flags [.], ack 32896002, win 64240, length
11:05:06.732073 IP 10.0.2.15.48842 > 34.107.221.82.80: Flags [P.], seq 0:301, ack 1, win 64240, le
11:05:06.802359 IP 10.0.2.15.48842 > 34.107.221.82.80: Flags [.], ack 221, win 64020, length 0
11:05:07.317901 IP 10.0.2.15.44736 > 49.44.194.34.80: Flags [S], seq 2223868270, win 64240, options
 7], length 0
11:05:07.360890 IP 10.0.2.15.44736 > 49.44.194.34.80: Flags [.], ack 33152002, win 64240, length 0
11:05:07.519144 IP 10.0.2.15.44736 > 49.44.194.34.80: Flags [P.], seq 0:421, ack 1, win 64240, leng
11:05:07.570527 IP 10.0.2.15.44736 > 49.44.194.34.80: Flags [.], ack 890, win 64008, length 0
11:05:08.282267 IP 10.0.2.15.44736 > 49.44.194.34.80: Flags [P.], seq 421:842, ack 890, win 64008,
11:05:08.342668 IP 10.0.2.15.44736 > 49.44.194.34.80: Flags [.], ack 1779, win 64008, length 0
11:05:08.360026 IP 10.0.2.15.42936 > 117.18.237.29.80: Flags [S], seq 1980972959, win 64240, option
 7], length 0
11:05:08.360758 IP 10.0.2.15.42938 > 117.18.237.29.80: Flags [S], seq 4058021181, win 64240, option
 7], length 0
11:05:08.397898 IP 10.0.2.15.42938 > 117.18.237.29.80: Flags [.], ack 34048002, win 64240, length
11:05:08.399111 IP 10.0.2.15.42938 > 117.18.237.29.80: Flags [P.], seq 0:422, ack 1, win 64240, le
11:05:08.407875 IP 10.0.2.15.42936 > 117.18.237.29.80: Flags [.], ack 34112002, win 64240, length
11:05:08.408446 IP 10.0.2.15.42936 > 117.18.237.29.80: Flags [P.], seq 0:422, ack 1, win 64240, le
11:05:08.451452 IP 10.0.2.15.42938 > 117.18.237.29.80: Flags [.], ack 740, win 63554, length 0
11:05:08.454148 IP 10.0.2.15.42936 > 117.18.237.29.80: Flags [.], ack 740, win 63554, length 0
11:05:09.013298 IP 10.0.2.15.42938 > 117.18.237.29.80: Flags [P.], seq 422:844, ack 740, win 63554,
11:05:09.044378 IP 10.0.2.15.42938 > 117.18.237.29.80: Flags [.], ack 1479, win 63554, length 0
11:05:10.236126 IP 10.0.2.15.44736 > 49.44.194.34.80: Flags [P.], seq 842:1263, ack 1779, win 6400
11:05:10.273464 IP 10.0.2.15.44774 > 49.44.194.34.80: Flags [S], seq 2358098742, win 64240, options
 7], length 0
11:05:10.287717 IP 10.0.2.15.44736 > 49.44.194.34.80: Flags [.], ack 2667, win 64008, length 0
11:05:10.288968 IP 10.0.2.15.44736 > 49.44.194.34.80: Flags [P.], seq 1263:1684, ack 2667, win 640
11:05:10.320909 IP 10.0.2.15.44774 > 49.44.194.34.80: Flags [.], ack 34752002, win 64240, length 0
11:05:10.356810 IP 10.0.2.15.44736 > 49.44.194.34.80: Flags [.], ack 3555, win 64008, length 0
11:05:15.383039 IP 10.0.2.15.44774 > 49.44.194.34.80: Flags [F.], seq 0, ack 1, win 64240, length
11:05:15.431909 IP 10.0.2.15.44774 > 49.44.194.34.80: Flags [.], ack 2, win 64240, length 0
^X11:05:16.756350 IP 10.0.2.15.48840 > 34.107.221.82.80: Flags [.], ack 303, win 63938, length 0
11:05:16.809834 IP 10.0.2.15.48842 > 34.107.221.82.80: Flags [.], ack 221, win 64020, length 0
```

sudo tcpdump-n -i enp0s3 src 10.0.2.15 or dst port 80



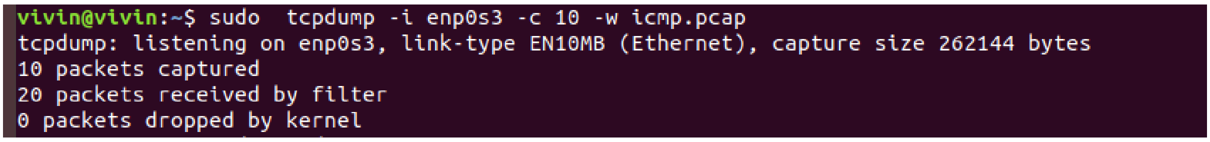
```
vivin@vivin:~$ sudo tcpdump -n -i enp0s3 src 10.0.2.15 or dst port 80
[sudo] password for vivin:
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on enp0s3, link-type EN10MB (Ethernet), capture size 262144 bytes
11:26:36.486639 IP 10.0.2.15.49163 > 8.8.4.4.53: 55878+ [1au] A? connectivity-check.ubuntu.com. (58
11:26:37.487447 IP 10.0.2.15.57326 > 35.224.170.84.80: Flags [S], seq 1850181661, win 64240, option
 7], length 0
11:26:38.064499 IP 10.0.2.15.57326 > 35.224.170.84.80: Flags [.], ack 196544002, win 64240, length
11:26:38.065211 IP 10.0.2.15.57326 > 35.224.170.84.80: Flags [P.], seq 0:87, ack 1, win 64240, leng
11:26:38.566038 IP 10.0.2.15.57326 > 35.224.170.84.80: Flags [.], ack 149, win 64092, length 0
11:26:38.566737 IP 10.0.2.15.57326 > 35.224.170.84.80: Flags [F.], seq 87, ack 150, win 64091, leng
11:26:48.203915 IP 10.0.2.15.53059 > 8.8.4.4.53: 63064+ [1au] A? incoming.telemetry.mozilla.org. (5
11:26:48.204161 IP 10.0.2.15.39523 > 8.8.4.4.53: 54443+ [1au] AAAA? incoming.telemetry.mozilla.org.
11:26:48.204752 IP 10.0.2.15.49368 > 35.227.207.240.443: Flags [P.], seq 1455609112:1455609269, ack
11:26:48.205037 IP 10.0.2.15.49368 > 35.227.207.240.443: Flags [P.], seq 157:620, ack 1, win 64015,
11:26:48.206865 IP 10.0.2.15.49368 > 35.227.207.240.443: Flags [P.], seq 620:753, ack 1, win 64015,
11:26:48.207051 IP 10.0.2.15.49368 > 35.227.207.240.443: Flags [P.], seq 753:1111, ack 1, win 64015
11:26:48.209939 IP 10.0.2.15.49368 > 35.227.207.240.443: Flags [P.], seq 1111:1239, ack 1, win 6401
11:26:48.213274 IP 10.0.2.15.49368 > 35.227.207.240.443: Flags [P.], seq 1239:1796, ack 1, win 6401
11:26:48.464307 IP 10.0.2.15.49368 > 35.227.207.240.443: Flags [.], ack 89, win 64015, length 0
11:26:48.465896 IP 10.0.2.15.49368 > 35.227.207.240.443: Flags [P.], seq 1796:1842, ack 89, win 640
11:26:48.497894 IP 10.0.2.15.49368 > 35.227.207.240.443: Flags [.], ack 177, win 64015, length 0
11:26:48.501287 IP 10.0.2.15.49368 > 35.227.207.240.443: Flags [P.], seq 1842:1888, ack 177, win 64
11:26:48.508349 IP 10.0.2.15.49368 > 35.227.207.240.443: Flags [.], ack 248, win 64015, length 0
11:26:58.595562 IP 10.0.2.15.39272 > 8.8.4.4.53: 30496+ [1au] A? www.google.com. (43)
11:26:58.597547 IP 10.0.2.15.45683 > 8.8.4.4.53: 305+ [1au] AAAA? www.google.com. (43)
11:26:59.019537 IP 10.0.2.15.50144 > 142.250.67.36.443: Flags [S], seq 992690073, win 64240, option
e 7], length 0
11:26:59.168054 IP 10.0.2.15.50144 > 142.250.67.36.443: Flags [.], ack 199232002, win 64240, length
11:26:59.170431 IP 10.0.2.15.50144 > 142.250.67.36.443: Flags [P.], seq 0:517, ack 1, win 64240, le
11:26:59.316962 IP 10.0.2.15.50144 > 142.250.67.36.443: Flags [.], ack 1431, win 62920, length 0
11:26:59.327849 IP 10.0.2.15.50144 > 142.250.67.36.443: Flags [.], ack 4288, win 62920, length 0
11:26:59.330750 IP 10.0.2.15.50144 > 142.250.67.36.443: Flags [P.], seq 517:581, ack 4288, win 6292
11:26:59.331645 IP 10.0.2.15.50144 > 142.250.67.36.443: Flags [P.], seq 581:751, ack 4288, win 6292
11:26:59.332245 IP 10.0.2.15.50144 > 142.250.67.36.443: Flags [P.], seq 751:1493, ack 4288, win 629
11:26:59.333651 IP 10.0.2.15.50144 > 142.250.67.36.443: Flags [P.], seq 1493:1524, ack 4288, win 62
11:26:59.357327 IP 10.0.2.15.50144 > 142.250.67.36.443: Flags [.], ack 4896, win 62920, length 0
11:26:59.357638 IP 10.0.2.15.50144 > 142.250.67.36.443: Flags [.], ack 4927, win 62920, length 0
11:26:59.358311 IP 10.0.2.15.50144 > 142.250.67.36.443: Flags [P.], seq 1524:1555, ack 4927, win 62
11:26:59.397068 IP 10.0.2.15.50144 > 142.250.67.36.443: Flags [.], ack 5212, win 62920, length 0
11:26:59.401561 IP 10.0.2.15.50144 > 142.250.67.36.443: Flags [.], ack 5282, win 62920, length 0
11:26:59.404937 IP 10.0.2.15.50144 > 142.250.67.36.443: Flags [P.], seq 1555:1594, ack 5282, win 62
```

Saving packet headers to a file

Store packet headers to a file with the -w flag. The files to save the output use pcap format and have an extension of .pcap.

PCAP stands for packet capture. The following command saves 10 lines of output on the eth1 interface to icmp.pcap.

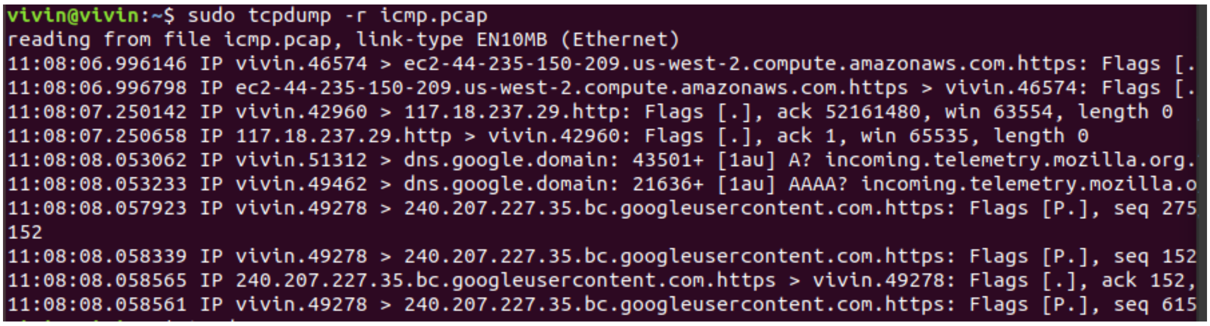
sudo tcpdump -i enp0s3 -c 10 -w icmp.pcap



```
vivin@vivin:~$ sudo  tcpdump -i enp0s3 -c 10 -w icmp.pcap
tcpdump: listening on enp0s3, link-type EN10MB (Ethernet), capture size 262144 bytes
10 packets captured
20 packets received by filter
0 packets dropped by kernel
```

You can read this file with -r flag:

sudo tcpdump -r icmp.pcap

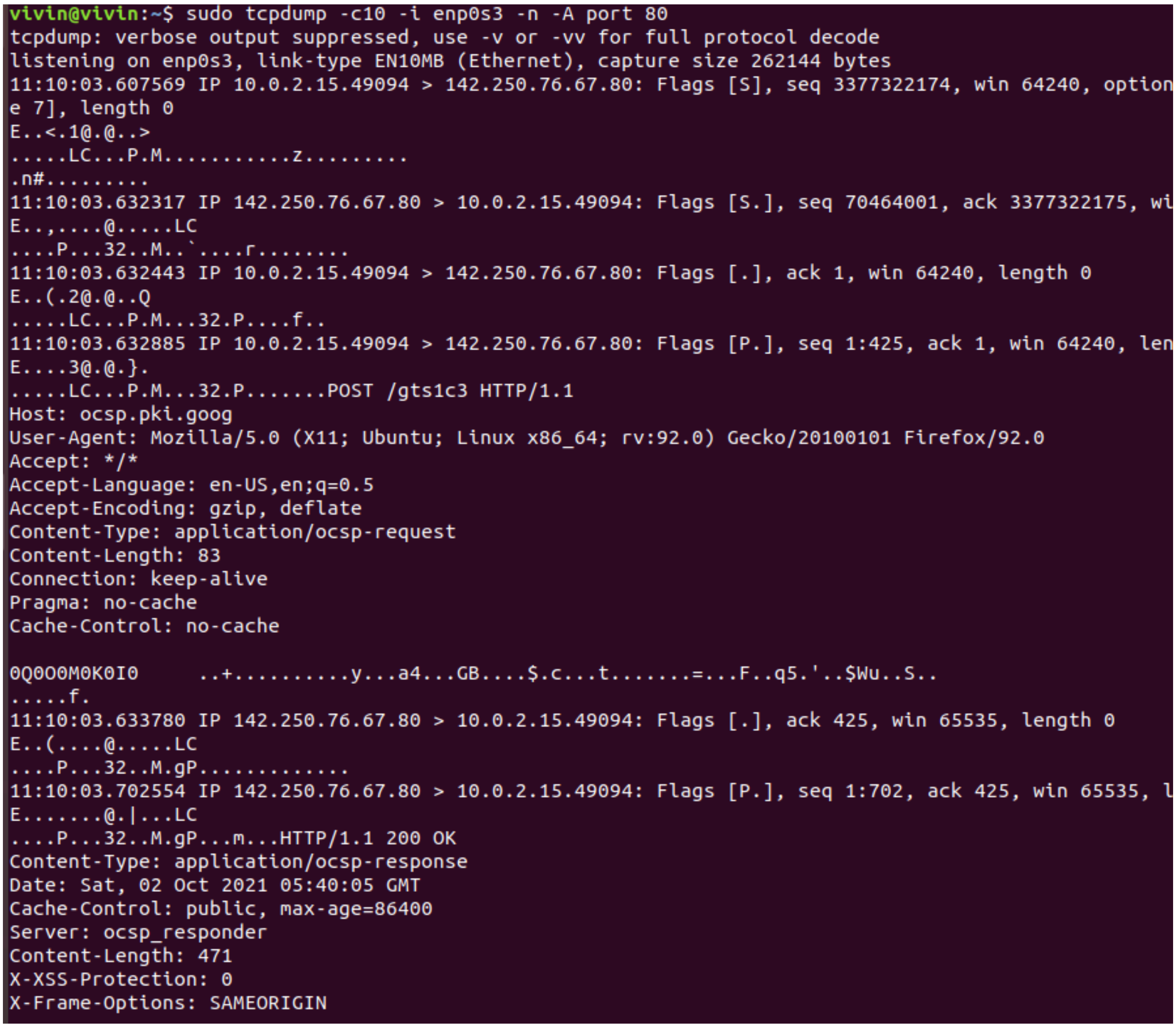


```
vivin@vivin:~$ sudo tcpdump -r icmp.pcap
reading from file icmp.pcap, link-type EN10MB (Ethernet)
11:08:06.996146 IP vivin.46574 > ec2-44-235-150-209.us-west-2.compute.amazonaws.com.https: Flags [.
11:08:06.996798 IP ec2-44-235-150-209.us-west-2.compute.amazonaws.com.https > vivin.46574: Flags [.
11:08:07.250142 IP vivin.42960 > 117.18.237.29.http: Flags [.], ack 52161480, win 63554, length 0
11:08:07.250658 IP 117.18.237.29.http > vivin.42960: Flags [.], ack 1, win 65535, length 0
11:08:08.053062 IP vivin.51312 > dns.google.domain: 43501+ [1au] A? incoming.telemetry.mozilla.org.
11:08:08.053233 IP vivin.49462 > dns.google.domain: 21636+ [1au] AAAA? incoming.telemetry.mozilla.o
11:08:08.057923 IP vivin.49278 > 240.207.227.35.bc.googleusercontent.com.https: Flags [P.], seq 275
152
11:08:08.058339 IP vivin.49278 > 240.207.227.35.bc.googleusercontent.com.https: Flags [P.], seq 152
11:08:08.058565 IP 240.207.227.35.bc.googleusercontent.com.https > vivin.49278: Flags [.], ack 152,
11:08:08.058561 IP vivin.49278 > 240.207.227.35.bc.googleusercontent.com.https: Flags [P.], seq 615
```

Viewing packet details

To view packet contents use -A option. This prints the packet contents in ASCII, which can be of help in network troubleshooting. Also -X flag can be used to display output in hex format. This may not be of much help if the connection is encrypted.

sudo tcpdump -c10 -i enp0s3 -n -A port 80



```
vivin@vivin:~$ sudo tcpdump -c10 -i enp0s3 -n -A port 80
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on enp0s3, link-type EN10MB (Ethernet), capture size 262144 bytes
11:10:03.607569 IP 10.0.2.15.49094 > 142.250.76.67.80: Flags [S], seq 3377322174, win 64240, option
e 7], length 0
E..<.1@.@..>
.....LC...P.M...........z.........
.n#.........
11:10:03.632317 IP 142.250.76.67.80 > 10.0.2.15.49094: Flags [S.], seq 70464001, ack 3377322175, wi
E..,....@.....LC
....P...32..M..`....r........
11:10:03.632443 IP 10.0.2.15.49094 > 142.250.76.67.80: Flags [.], ack 1, win 64240, length 0
E..(.2@.@..Q
.....LC...P.M...32.P....f..
11:10:03.632885 IP 10.0.2.15.49094 > 142.250.76.67.80: Flags [P.], seq 1:425, ack 1, win 64240, len
E....3@.@.}.
.....LC...P.M...32.P.......POST /gts1c3 HTTP/1.1
Host: ocsp.pki.goog
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:92.0) Gecko/20100101 Firefox/92.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: application/ocsp-request
Content-Length: 83
Connection: keep-alive
Pragma: no-cache
Cache-Control: no-cache

0Q0O0M0K0I0     ..+..........y...a4...GB....$.c...t.......=...F..q5.'..$Wu..S..
.....f.
11:10:03.633780 IP 142.250.76.67.80 > 10.0.2.15.49094: Flags [.], ack 425, win 65535, length 0
E..(....@.....LC
....P...32..M.gP.............
11:10:03.702554 IP 142.250.76.67.80 > 10.0.2.15.49094: Flags [P.], seq 1:702, ack 425, win 65535, l
E.......@.|...LC
....P...32..M.gP...m...HTTP/1.1 200 OK
Content-Type: application/ocsp-response
Date: Sat, 02 Oct 2021 05:40:05 GMT
Cache-Control: public, max-age=86400
Server: ocsp_responder
Content-Length: 471
X-XSS-Protection: 0
X-Frame-Options: SAMEORIGIN
```