Task:

In this lab activity, you'll capture and analyze live network traffic using tcpdump. You'll use Linux commands in the Bash shell to complete these steps.

Scenario:

You're a network analyst who needs to use `tcpdump` to capture and analyze live network traffic from a Linux virtual machine.

The lab starts with your user account, called `analyst`, already logged in to a Linux terminal.

Your Linux user's home directory contains a sample packet capture file that you will use at the end of the lab to answer a few questions about the network traffic that it contains.

Here's how you'll do this: First, you'll identify network interfaces to capture network packet data. Second, you'll use `tcpdump` to filter live network traffic. Third, you'll capture network traffic using `tcpdump`. Finally, you'll filter the captured packet data.

After you click the **Start Lab** button, you will see a shell, where you will be performing further steps in the lab.

Task 1. Identify network interfaces

1. **Use sudo ifconfig to identify the interfaces that are available:**

```
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1460
        inet 172.17.0.2  netmask 255.255.0.0  broadcast 172.17.255.255
        ether 02:42:ac:11:00:02  txqueuelen 0  (Ethernet)
        RX packets 784  bytes9379957 (8.9 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 683  bytes 56880 (55.5 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 400  bytes 42122 (041.1 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 400  bytes 42122 (041.1 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0 collisions 0
```

2. **Use tcpdump to identify the interface options available for packet capture:**

```
analyst@8544d954e232:~$ sudo tcpdump -D
1.eth0 [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)
analyst@8544d954e232:~$
```

Task 2. Inspect the network traffic of a network interface with tcpdump

- Filter live network packet data from the `eth0` interface with `tcpdump`:

```
analyst@8544d954e232:~$ sudo tcpdump -i eth0 -v -c5
tcpdump: listening on eth0, link-type EN10MB (Ethernet), capture size 262
44 bytes
03:46:45.085782 IP (tos 0x0, ttl 64, id 21210, offset 0, flags [DF], proto
 TCP (6), length 113)
    8544d954e232.5000 > nginx-us-west1-b.c.qwiklabs-terminal-vms-prod-00.i
nternal.54888: Flags [P.], cksum 0x588c (incorrect -> 0xeed1), seq 266694
473:2666949534, ack 4044447386, win 492, options [nop,nop,TS val 27716386
 ecr 4222342233], length 61
03:46:45.086236 IP (tos 0x0, ttl 63, id 10595, offset 0, flags [DF], proto
 TCP (6), length 52)
```

Task 3. Capture network traffic with tcpdump

1. **Capture packet data into a file called `capture.pcap`:**

```
analyst@8544d954e232:~$ sudo tcpdump -i eth0 -nn -c9 port 80 -w capture.pc
ap &
[1] 12796
analyst@8544d954e232:~$ tcpdump: listening on eth0, link-type EN10MB (Ethe
rnet), capture size 262144 bytes
```

2. **Use `curl` to generate some HTTP (port 80) traffic:**

```
curl opensource.google.com
<HTML><HEAD><meta http-equiv="content-type" content="text/html;charset=utf
-8">
<TITLE>301 Moved</TITLE></HEAD><BODY>
<H1>301 Moved</H1>
The document has moved
<A HREF="https://opensource.google/">here</A>.
</BODY></HTML>
analyst@8544d954e232:~$ 9 packets captured
10 packets received by filter
0 packets dropped by kernel
```

3. Verify that packet data has been captured:

```
ls -l capture.pcap
-rw-r--r-- 1 root root 1401 Nov 30 03:52 capture.pcap
[1]+  Done                    sudo tcpdump -i eth0 -nn -c9 port 80 -w capt
ure.pcap
```

Task 4. Filter the captured packet data

1. Use the `tcpdump` command to filter the packet header data from the `capture.pcap` capture file:

```
analyst@8544d954e232:~$ sudo tcpdump -nn -r capture.pcap -v
reading from file capture.pcap, link-type EN10MB (Ethernet)
03:52:16.843874 IP (tos 0x0, ttl 64, id 28426, offset 0, flags [DF], proto
 TCP (6), length 60)
    172.17.0.2.53180 > 74.125.199.102.80: Flags [S], cksum 0xbe25 (incorr
ect -> 0x0fc7), seq 2751877779, win 32660, options [mss 1420,sackOK,TS val
2720502280 ecr 0,nop,wscale 6], length 0
03:52:16.845220 IP (tos 0x0, ttl 126, id 0, offset 0, flags [DF], proto TC
P (6), length 60)
    74.125.199.102.80 > 172.17.0.2.53180: Flags [S.], cksum 0x59a0 (corre
t), seq 326131319, ack 2751877780, win 65535, options [mss 1420,sackOK,TS
val 312062247 ecr 2720502280,nop,wscale 8], length 0
03:52:16.845260 IP (tos 0x0, ttl 64, id 28427, offset 0, flags [DF], proto
 TCP (6), length 52)
    172.17.0.2.53180 > 74.125.199.102.80: Flags [.], cksum 0xbe1d (incorr
ct -> 0x8644), ack 1, win 511, options [nop,nop,TS val 2720502282 ecr 312
62247], length 0
03:52:16.845318 IP (tos 0x0, ttl 64, id 28428, offset 0, flags [DF], proto
 TCP (6), length 137)
    172.17.0.2.53180 > 74.125.199.102.80: Flags [P.], cksum 0xbe72 (incor
ect -> 0xf4f7), seq 1:86, ack 1, win 511, options [nop,nop,TS val 2720502
82 ecr 312062247], length 85: HTTP, length: 85
        GET / HTTP/1.1
        Host: opensource.google.com
        User-Agent: curl/7.64.0
        Accept: */*
```

2. Use the `tcpdump` command to filter the extended packet data from the `capture.pcap` capture file:

```
analyst@8544d954e232:~$ sudo tcpdump -nn -r capture.pcap -X
reading from file capture.pcap, link-type EN10MB (Ethernet)
03:52:16.843874 IP 172.17.0.2.53180 > 74.125.199.102.80: Flags [S], seq 27
51877779, win 32660, options [mss 1420,sackOK,TS val 2720502280 ecr 0,nop,
wscale 6], length 0
        0x0000:  4500 003c 6f0a 4000 4006 0dbb ac11 0002  E..<o.@.@.......
        0x0010:  4a7d c766 cfbc 0050 a406 5293 0000 0000  J}.f...P..R.....
        0x0020:  a002 7f94 be25 0000 0204 058c 0402 080a  .....%..........
        0x0030:  a227 9208 0000 0000 0103 0306            .'..........
03:52:16.845220 IP 74.125.199.102.80 > 172.17.0.2.53180: Flags [S.], seq 3
26131319, ack 2751877780, win 65535, options [mss 1420,sackOK,TS val 31206
2247 ecr 2720502280,nop,wscale 8], length 0
        0x0000:  4500 003c 0000 4000 7e06 3ec5 4a7d c766  E..<..@.~.>.J}.f
        0x0010:  ac11 0002 0050 cfbc 1370 5e77 a406 5294  .....P...p^w..R.
        0x0020:  a012 ffff 59a0 0000 0204 058c 0402 080a  ....Y...........
        0x0030:  1299 b127 a227 9208 0103 0308            ...'.'......
03:52:16.845260 IP 172.17.0.2.53180 > 74.125.199.102.80: Flags [.], ack 1,
 win 511, options [nop,nop,TS val 2720502282 ecr 312062247], length 0
        0x0000:  4500 0034 6f0b 4000 4006 0dc2 ac11 0002  E..4o.@.@.......
        0x0010:  4a7d c766 cfbc 0050 a406 5294 1370 5e78  J}.f...P..R..p^x
```