NETWORK TRAFFIC CAPTURE AND ANALYSIS WITH TCPDUMP

Project Description

In this project, I used Linux networking tools to capture and analyze live network traffic. I identified available network interfaces, monitored TCP traffic with ‘tcpdump’, saved packets into a ‘pcap’ file, and performed deeper inspection of HTTP requests and responses. The outputs were very detailed, so the screenshots provided are snapshots of longer results.

Identifying Network Interfaces

I began by checking the available network interfaces on the system. The output displayed both the Ethernet interface (eth0) with an assigned IP address and the loopback interface (lo) with its local IP. This confirmed that the machine was connected to the network and ready for traffic capture.

```
analyst@14fdc6933e25:~$ sudo ifconfig
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 749  bytes 13967091 (13.3 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 469  bytes 41646 (40.6 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
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 63  bytes 9128 (8.9 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 63  bytes 9128 (8.9 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

Capturing Live Network Traffic

Next, I captured live packets from the Ethernet interface. The output showed several packets in real time, including information about source and destination IP addresses, protocols, and flags. I then extended the capture to focus on HTTP traffic and saved the packets into a `.pcap` file for further analysis. The screenshot reflects only a portion of the full packet capture.

```
analyst@14fdc6933e25:~$ sudo tcpdump -D
1.eth0 [Up, Running, Connected]
2.any (Pseudo-device that captures on all interfaces) [Up, Running]
3.lo [Up, Running, Loopback]
4.bluetooth-monitor (Bluetooth Linux Monitor) [Wireless]
5.nflog (Linux netfilter log (NFLOG) interface) [none]
6.nfqueue (Linux netfilter queue (NFQUEUE) interface) [none]
7.dbus-system (D-Bus system bus) [none]
8.dbus-session (D-Bus session bus) [none]
analyst@14fdc6933e25:~$ sudo tcpdump -i eth0 -v -c5
tcpdump: listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
04:04:34.858026 IP (tos 0x0, ttl 64, id 13110, offset 0, flags [DF], proto TCP (6), length 12
1)
    14fdc6933e25.5000 > nginx-us-east1-c.c.qwiklabs-terminal-vms-prod-00.internal.33408: Flag
s [P.], cksum 0x58fa (incorrect -> 0x34c3), seq 2453684565:2453684634, ack 6180316, win 998,
options [nop,nop,TS val 1831092378 ecr 2197326373], length 69
04:04:34.858303 IP (tos 0x0, ttl 63, id 25857, offset 0, flags [DF], proto TCP (6), length 52
)
    nginx-us-east1-c.c.qwiklabs-terminal-vms-prod-00.internal.33408 > 14fdc6933e25.5000: Flag
s [.], cksum 0xa90e (correct), ack 69, win 507, options [nop,nop,TS val 2197326416 ecr 183109
2378], length 0
04:04:34.937765 IP (tos 0x0, ttl 64, id 30255, offset 0, flags [DF], proto UDP (17), length 7
1)
    14fdc6933e25.36577 > metadata.google.internal.domain: 29716+ PTR? 102.0.21.172.in-addr.ar
pa. (43)
04:04:34.939383 IP (tos 0x0, ttl 64, id 13111, offset 0, flags [DF], proto TCP (6), length 14
0)
    14fdc6933e25.5000 > nginx-us-east1-c.c.qwiklabs-terminal-vms-prod-00.internal.33408: Flag
s [P.], cksum 0x590d (incorrect -> 0xfe0f), seq 69:157, ack 1, win 998, options [nop,nop,TS v
al 1831092460 ecr 2197326416], length 88
04:04:34.939653 IP (tos 0x0, ttl 63, id 25858, offset 0, flags [DF], proto TCP (6), length 52
)
    nginx-us-east1-c.c.qwiklabs-terminal-vms-prod-00.internal.33408 > 14fdc6933e25.5000: Flag
s [.], cksum 0xa812 (correct), ack 157, win 507, options [nop,nop,TS val 2197326498 ecr 18310
92460], length 0
5 packets captured
10 packets received by filter
0 packets dropped by kernel
analyst@14fdc6933e25:~$ sudo tcpdump -i eth0 -nn -c9 port 80 -w capture.pcap &
[1] 13513
analyst@14fdc6933e25:~$ tcpdump: listening on eth0, link-type EN10MB (Ethernet), snapshot len
gth 262144 bytes
```

Saving and Reading Packet Capture Data

Once the capture file was created, I replayed it to examine the traffic in more detail. The results showed a full TCP handshake followed by an HTTP request sent to `opensource.google.com`. The server responded with a `301 Moved Permanently` message, redirecting the client to `https://opensource.google/`. This confirmed that tcpdump can be used not just to collect data, but also to clearly follow the flow of communication.

```
analyst@14fdc6933e25:~$ 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@14fdc6933e25:~$ 9 packets captured
10 packets received by filter
0 packets dropped by kernel

[1]+  Done                    sudo tcpdump -i eth0 -nn -c9 port 80 -w capture.pcap
analyst@14fdc6933e25:~$ ls -l capture.pcap
-rw-r--r-- 1 tcpdump tcpdump 1456 Sep 15 04:10 capture.pcap
analyst@14fdc6933e25:~$ sudo tcpdump -nn -r capture.pcap -v
reading from file capture.pcap, link-type EN10MB (Ethernet), snapshot length 262144
04:10:57.552779 IP (tos 0x0, ttl 64, id 26455, offset 0, flags [DF], proto TCP (6), length 60
)
    172.17.0.2.51634 > 142.250.98.102.80: Flags [S], cksum 0x9da2 (incorrect -> 0x8119), seq
210783916, win 65320, options [mss 1420,sackOK,TS val 174127827 ecr 0,nop,wscale 6], length 0
04:10:57.553268 IP (tos 0x0, ttl 126, id 0, offset 0, flags [DF], proto TCP (6), length 60)
    142.250.98.102.80 > 172.17.0.2.51634: Flags [S.], cksum 0xb1e1 (correct), seq 3748083983,
 ack 210783917, win 65535, options [mss 1420,sackOK,TS val 1326999230 ecr 174127827,nop,wscal
e 8], length 0
04:10:57.553287 IP (tos 0x0, ttl 64, id 26456, offset 0, flags [DF], proto TCP (6), length 52
)
    172.17.0.2.51634 > 142.250.98.102.80: Flags [.], cksum 0x9d9a (incorrect -> 0xdc88), ack
1, win 1021, options [nop,nop,TS val 174127828 ecr 1326999230], length 0
04:10:57.553382 IP (tos 0x0, ttl 64, id 26457, offset 0, flags [DF], proto TCP (6), length 13
7)
    172.17.0.2.51634 > 142.250.98.102.80: Flags [P.], cksum 0x9def (incorrect -> 0x4a3c), seq
 1:86, ack 1, win 1021, options [nop,nop,TS val 174127828 ecr 1326999230], length 85: HTTP, l
ength: 85
        GET / HTTP/1.1
        Host: opensource.google.com
        User-Agent: curl/7.74.0
        Accept: */*

04:10:57.553538 IP (tos 0x0, ttl 126, id 0, offset 0, flags [DF], proto TCP (6), length 52)
    142.250.98.102.80 > 172.17.0.2.51634: Flags [.], cksum 0xdc14 (correct), ack 86, win 1051
, options [nop,nop,TS val 1326999231 ecr 174127828], length 0
04:10:57.555585 IP (tos 0x0, ttl 126, id 0, offset 0, flags [DF], proto TCP (6), length 645)
    142.250.98.102.80 > 172.17.0.2.51634: Flags [P.], cksum 0x37ed (correct), seq 1:594, ack
86, win 1051, options [nop,nop,TS val 1326999233 ecr 174127828], length 593: HTTP, length: 59
3
        HTTP/1.1 301 Moved Permanently
        X-Content-Type-Options: nosniff
        Cross-Origin-Resource-Policy: cross-origin
        Location: https://opensource.google/
        Server: sffe
```

Inspecting Packets in Hex and ASCII

For a deeper inspection, I viewed the captured packets in both hexadecimal and ASCII formats. This revealed the raw payload of the HTTP request, including headers such as `Host: opensource.google.com` and `User-Agent: curl/7.74.0`. The server's HTTP response headers were also visible, such as `X-Content-Type-Options: nosniff` and `Cross-Origin-Resource-Policy: cross-origin`. This step demonstrated how packet inspection can provide complete visibility into communication down to the byte level.

```
analyst@14fdc6933e25:~$ sudo tcpdump -nn -r capture.pcap -X
reading from file capture.pcap, link-type EN10MB (Ethernet), snapshot length 262144
04:10:57.552779 IP 172.17.0.2.51634 > 142.250.98.102.80: Flags [S], seq 210783916, win 65320,
 options [mss 1420,sackOK,TS val 174127827 ecr 0,nop,wscale 6], length 0
        0x0000:  4500 003c 6757 4000 4006 35f1 ac11 0002  E..<gW@.@.5.....
        0x0010:  8efa 6266 c9b2 0050 0c90 4eac 0000 0000  ..bf...P..N.....
        0x0020:  a002 ff28 9da2 0000 0204 058c 0402 080a  ...(............
        0x0030:  0a60 fad3 0000 0000 0103 0306            .`..........
04:10:57.553268 IP 142.250.98.102.80 > 172.17.0.2.51634: Flags [S.], seq 3748083983, ack 2107
83917, win 65535, options [mss 1420,sackOK,TS val 1326999230 ecr 174127827,nop,wscale 8], len
gth 0
        0x0000:  4500 003c 0000 4000 7e06 5f48 8efa 6266  E..<..@.~._H..bf
        0x0010:  ac11 0002 0050 c9b2 df67 390f 0c90 4ead  .....P...g9...N.
        0x0020:  a012 ffff b1e1 0000 0204 058c 0402 080a  ................
        0x0030:  4f18 66be 0a60 fad3 0103 0308            O.f..`......
04:10:57.553287 IP 172.17.0.2.51634 > 142.250.98.102.80: Flags [.], ack 1, win 1021, options
[nop,nop,TS val 174127828 ecr 1326999230], length 0
        0x0000:  4500 0034 6758 4000 4006 35f8 ac11 0002  E..4gX@.@.5.....
        0x0010:  8efa 6266 c9b2 0050 0c90 4ead df67 3910  ..bf...P..N..g9.
        0x0020:  8010 03fd 9d9a 0000 0101 080a 0a60 fad4  .............`..
        0x0030:  4f18 66be                                O.f.
04:10:57.553382 IP 172.17.0.2.51634 > 142.250.98.102.80: Flags [P.], seq 1:86, ack 1, win 102
1, options [nop,nop,TS val 174127828 ecr 1326999230], length 85: HTTP: GET / HTTP/1.1
        0x0000:  4500 0089 6759 4000 4006 35a2 ac11 0002  E...gY@.@.5.....
        0x0010:  8efa 6266 c9b2 0050 0c90 4ead df67 3910  ..bf...P..N..g9.
        0x0020:  8018 03fd 9def 0000 0101 080a 0a60 fad4  .............`..
        0x0030:  4f18 66be 4745 5420 2f20 4854 5450 2f31  O.f.GET./.HTTP/1
        0x0040:  2e31 0d0a 486f 7374 3a20 6f70 656e 736f  .1..Host:.openso
        0x0050:  7572 6365 2e67 6f6f 676c 652e 636f 6d0d  urce.google.com.
        0x0060:  0a55 7365 722d 4167 656e 743a 2063 7572  .User-Agent:.cur
        0x0070:  6c2f 372e 3734 2e30 0d0a 4163 6365 7074  l/7.74.0..Accept
        0x0080:  3a20 2a2f 2a0d 0a0d 0a                   :.*/*....
04:10:57.553538 IP 142.250.98.102.80 > 172.17.0.2.51634: Flags [.], ack 86, win 1051, options
 [nop,nop,TS val 1326999231 ecr 174127828], length 0
        0x0000:  4500 0034 0000 4000 7e06 5f50 8efa 6266  E..4..@.~._P..bf
        0x0010:  ac11 0002 0050 c9b2 df67 3910 0c90 4f02  .....P...g9...O.
        0x0020:  8010 041b dc14 0000 0101 080a 4f18 66bf  ............O.f.
        0x0030:  0a60 fad4                                .`..
04:10:57.555585 IP 142.250.98.102.80 > 172.17.0.2.51634: Flags [P.], seq 1:594, ack 86, win 1
051, options [nop,nop,TS val 1326999233 ecr 174127828], length 593: HTTP: HTTP/1.1 301 Moved
Permanently
        0x0000:  4500 0285 0000 4000 7e06 5cff 8efa 6266  E.....@.~.\...bf
        0x0010:  ac11 0002 0050 c9b2 df67 3910 0c90 4f02  .....P...g9...O.
        0x0020:  8018 041b 37ed 0000 0101 080a 4f18 66c1  ....7.......O.f.
        0x0030:  0a60 fad4 4854 5450 2f31 2e31 2033 3031  .`..HTTP/1.1.301
        0x0040:  204d 6f76 6564 2050 6572 6d61 6e65 6e74  .Moved.Permanent
        0x0050:  6c79 0d0a 582d 436f 6e74 656e 742d 5479  ly..X-Content-Ty
        0x0060:  7065 2d4f 7074 696f 6e73 3a20 6e6f 736e  pe-Options:.nosn
        0x0070:  6966 660d 0a43 726f 7373 2d4f 7269 6769  iff..Cross-Origi
        0x0080:  6e2d 5265 736f 7572 6365 2d50 6f6c 6963  n-Resource-Polic
```

```
analyst@14fdc6933e25:~$ sudo tcpdump -nn -r capture.pcap -v
reading from file capture.pcap, link-type EN10MB (Ethernet), snapshot length 262144
04:10:57.552779 IP (tos 0x0, ttl 64, id 26455, offset 0, flags [DF], proto TCP (6), length 60
)
    172.17.0.2.51634 > 142.250.98.102.80: Flags [S], cksum 0x9da2 (incorrect -> 0x8119), seq
210783916, win 65320, options [mss 1420,sackOK,TS val 174127827 ecr 0,nop,wscale 6], length 0
04:10:57.553268 IP (tos 0x0, ttl 126, id 0, offset 0, flags [DF], proto TCP (6), length 60)
    142.250.98.102.80 > 172.17.0.2.51634: Flags [S.], cksum 0xb1e1 (correct), seq 3748083983,
 ack 210783917, win 65535, options [mss 1420,sackOK,TS val 1326999230 ecr 174127827,nop,wscal
e 8], length 0
04:10:57.553287 IP (tos 0x0, ttl 64, id 26456, offset 0, flags [DF], proto TCP (6), length 52
)
    172.17.0.2.51634 > 142.250.98.102.80: Flags [.], cksum 0x9d9a (incorrect -> 0xdc88), ack
1, win 1021, options [nop,nop,TS val 174127828 ecr 1326999230], length 0
04:10:57.553382 IP (tos 0x0, ttl 64, id 26457, offset 0, flags [DF], proto TCP (6), length 13
7)
    172.17.0.2.51634 > 142.250.98.102.80: Flags [P.], cksum 0x9def (incorrect -> 0x4a3c), seq
 1:86, ack 1, win 1021, options [nop,nop,TS val 174127828 ecr 1326999230], length 85: HTTP, l
ength: 85
        GET / HTTP/1.1
        Host: opensource.google.com
        User-Agent: curl/7.74.0
        Accept: */*

04:10:57.553538 IP (tos 0x0, ttl 126, id 0, offset 0, flags [DF], proto TCP (6), length 52)
    142.250.98.102.80 > 172.17.0.2.51634: Flags [.], cksum 0xdc14 (correct), ack 86, win 1051
, options [nop,nop,TS val 1326999231 ecr 174127828], length 0
04:10:57.555585 IP (tos 0x0, ttl 126, id 0, offset 0, flags [DF], proto TCP (6), length 645)
    142.250.98.102.80 > 172.17.0.2.51634: Flags [P.], cksum 0x37ed (correct), seq 1:594, ack
86, win 1051, options [nop,nop,TS val 1326999233 ecr 174127828], length 593: HTTP, length: 59
3
        HTTP/1.1 301 Moved Permanently
        X-Content-Type-Options: nosniff
        Cross-Origin-Resource-Policy: cross-origin
        Location: https://opensource.google/
        Server: sffe
        Content-Length: 223
        X-XSS-Protection: 0
        Date: Mon, 15 Sep 2025 03:51:44 GMT
        Expires: Mon, 15 Sep 2025 04:21:44 GMT
        Cache-Control: public, max-age=1800
        Content-Type: text/html; charset=UTF-8
        Age: 1153

        <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>
```

Summary

In this project, I identified network interfaces, captured live traffic, saved and analyzed packet data, and inspected raw packet contents. These steps demonstrated how tcpdump can be used for both high-level monitoring and detailed deep packet inspection, making it a critical tool for cybersecurity and network analysis.