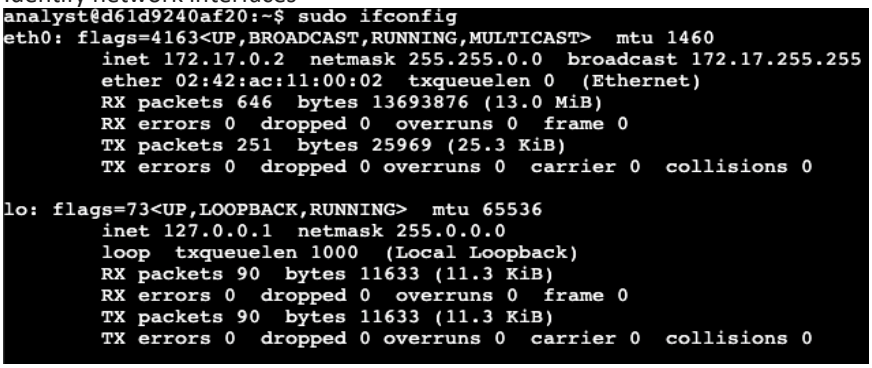
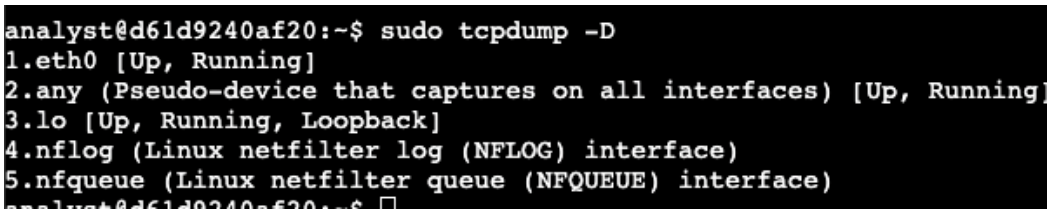
Incident handler's journal

This journal guides incident handlers through the complexities of managing cyber threats, offering insights into strategies, tools, and case studies. It's an essential read for enhancing skills and staying updated in the cybersecurity landscape.

Date: July 23, 2024	**Entry:** #1
Description	Documenting a cybersecurity incident
Tool(s) used	None.
The 5 W's	• **Who**: An organized group of unethical hackers • **What**: A ransomware security incident • **Where**: At a health care company • **When**: Tuesday 9:00 a.m. • **Why**: The incident happened because unethical hackers were able to access the company's systems using a phishing attack. After gaining access, the attackers launched their ransomware on the company's systems, encrypting critical files. The attackers' motivation appears to be financial because the ransom note they left demanded a large sum of money in exchange for the decryption key.
Additional notes	1. How could the health care company prevent an incident like this from occurring again? 2. Should the company pay the ransom to retrieve the decryption key?

Date: 30/10/2023	**Entry:** #2
Description	Documenting a cybersecurity incident
Tool(s) used	For this activity, I used VirusTotal, which is an investigative tool that analyzes files and URLs for malicious content such as viruses, worms, trojans, and more. It's a very helpful tool to use if you want to quickly check if an indicator of compromise like a website or file has been reported as malicious by others in the cybersecurity community. For this activity, I used VirusTotal to analyze a file hash, which was reported as malicious. This incident occurred in the **Detection and Analysis** phase. The scenario put me in the place of a security analyst at a SOC investigating a suspicious file hash. After the suspicious file was detected by the security systems in place, I had to perform deeper analysis and investigation to determine if the alert signified a real threat.
The 5 W's	● **Who:** An employee downloaded a suspicious file on his computer**.** ● **What**: A malicious payload was executed on the employee's computer. ● **When** Monday 10:00 am ● **Where**: At a financial services company. ● **Why**: The employee received an email containing an attachment. The attachment was a password-protected spreadsheet file. The spreadsheet's password was provided in the email. The employee downloaded the file, then entered the password to open the file.
Additional notes	● SHA256 file hash: 54e6ea47eb04634d3e87fd7787e2136ccfbcc80ade34f246a12cf93bab527f6b ● The file hash has been reported as malicious by over 56 vendors. Upon further investigation, this file hash is known as the malware Flagpro, which has been commonly used by the advanced threat actor BlackTech.

Date: 11/11/2023	**Entry:** #3

Description	capture and filter network traffic in a Linux environment.
Tool(s) used	tcpdump
Task	This activity consists of performing tasks associated with using tcpdump to capture network traffic. The data is captured in a packet capture (p-cap) file and then the content of the captured packet data is examined focusing on specific types of traffic.
Additional notes	1. Identify network interfaces  2. Identify the interface options available for package capture.(if systems do not include ifconfig)  3. Inspect network traffic (eth0): -i eth0: Capture data specifically from the eth0 interface. -v: Display detailed packet data. -c5: Capture 5 packets of data.

```
analyst@d61d9240af20:~$ 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 646  bytes 13693876 (13.0 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 251  bytes 25969 (25.3 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 90  bytes 11633 (11.3 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 90  bytes 11633 (11.3 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

```
analyst@d61d9240af20:~$ 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@d61d9240af20:~$ sudo tcpdump -i eth0 -v -c5
tcpdump: listening on eth0, link-type EN10MB (Ethernet), capture size 262144 byt
09:18:18.019643 IP (tos 0x0, ttl 64, id 4749, offset 0, flags [DF], proto TCP (6
    d61d9240af20.5000 > nginx-us-west1-c.c.qwiklabs-terminal-vms-prod-00.interna
um 0x5891 (incorrect -> 0xa9b2), seq 3302246382:3302246445, ack 2931523961, win
 val 1264142509 ecr 244937377], length 63
09:18:18.019878 IP (tos 0x0, ttl 63, id 59724, offset 0, flags [DF], proto TCP (
    nginx-us-west1-c.c.qwiklabs-terminal-vms-prod-00.internal.38256 > d61d9240af
m 0x20fd (correct), ack 63, win 507, options [nop,nop,TS val 244937418 ecr 12641
09:18:18.030131 IP (tos 0x0, ttl 64, id 4750, offset 0, flags [DF], proto TCP (6
    d61d9240af20.5000 > nginx-us-west1-c.c.qwiklabs-terminal-vms-prod-00.interna
um 0x58ae (incorrect -> 0xece5), seq 63:155, ack 1, win 501, options [nop,nop,TS
37418], length 92
09:18:18.030397 IP (tos 0x0, ttl 63, id 59725, offset 0, flags [DF], proto TCP (
    nginx-us-west1-c.c.qwiklabs-terminal-vms-prod-00.internal.38256 > d61d9240af
m 0x208b (correct), ack 155, win 507, options [nop,nop,TS val 244937429 ecr 1264
09:18:18.038952 IP (tos 0x0, ttl 64, id 49446, offset 0, flags [DF], proto UDP (
    d61d9240af20.56683 > metadata.google.internal.domain: 24186+ PTR? 2.0.22.172
5 packets captured
10 packets received by filter
0 packets dropped by kernel
```

4. Capture network traffic

-i eth0: Capture data from the eth0 interface.

-nn: Do not attempt to resolve IP addresses or ports to names.This is best practice from a security perspective, as the lookup data may not be valid. It also prevents malicious actors from being alerted to an investigation.

-c9: Capture 9 packets of data and then exit.

port 80: Filter only port 80 traffic. This is the default HTTP port.

-w capture.pcap: Save the captured data to the named file.

&: This is an instruction to the Bash shell to run the command in the background.

```
analyst@d61d9240af20:~$ sudo tcpdump -i eth0 -nn -c9 port 80 -w capture.pcap &
[1] 12828
analyst@d61d9240af20:~$ tcpdump: listening on eth0, link-type EN10MB (Ethernet), capture size 262
es
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@d61d9240af20:~$ 9 packets captured
12 packets received by filter
0 packets dropped by kernel
```

5. Verify that data has been capture

```
analyst@d61d9240af20:~$ ls -l capture.pcap
-rw-r--r-- 1 root root 1455 Dec 12 09:22 capture.pcap
```

6. Filter the captured packet data

   -nn: Disable port and protocol name lookup.

   -r: Read capture data from the named file.

   -v: Display detailed packet dat

```
nalyst@bb88ce441a42:~$ sudo tcpdump -nn -r capture.pcap -v
eading from file capture.pcap, link-type EN10MB (Ethernet)
9:33:10.305672 IP (tos 0x0, ttl 64, id 3455, offset 0, flags [DF], proto TCP (6),
   172.17.0.2.50098 > 74.125.142.101.80: Flags [S], cksum 0x8524 (incorrect -> 0xd
win 65320, options [mss 1420,sackOK,TS val 1775507883 ecr 0,nop,wscale 7], length
9:33:10.306673 IP (tos 0x60, ttl 126, id 0, offset 0, flags [DF], proto TCP (6), l
   74.125.142.101.80 > 172.17.0.2.50098: Flags [S.], cksum 0xb92c (correct), seq 3
446, win 65535, options [mss 1420,sackOK,TS val 1291387135 ecr 1775507883,nop,wsca
9:33:10.306726 IP (tos 0x0, ttl 64, id 3456, offset 0, flags [DF], proto TCP (6),
   172.17.0.2.50098 > 74.125.142.101.80: Flags [.], cksum 0x851c (incorrect -> 0xe
options [nop,nop,TS val 1775507884 ecr 1291387135], length 0
9:33:10.306775 IP (tos 0x0, ttl 64, id 3457, offset 0, flags [DF], proto TCP (6),
   172.17.0.2.50098 > 74.125.142.101.80: Flags [P.], cksum 0x8571 (incorrect -> 0x
, win 511, options [nop,nop,TS val 1775507884 ecr 1291387135], length 85: HTTP, le
      GET / HTTP/1.1
      Host: opensource.google.com
      User-Agent: curl/7.64.0
      Accept: */*
```

Date: 11/11/2023	**Entry:** #3
Description	capture and filter network traffic in a Linux environment.
Tool(s) used	tcpdump
Task	This activity consists of performing tasks associated with using tcpdump to capture network traffic. The data is captured in a packet capture (p-cap) file and then the content of the captured packet data is examined focusing on specific types of traffic.
Additional notes	7. Identify network interfaces 8. Identify the interface options available for package capture.(if systems do not include ifconfig)

```
analyst@d61d9240af20:~$ 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 646  bytes 13693876 (13.0 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 251  bytes 25969 (25.3 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 90  bytes 11633 (11.3 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 90  bytes 11633 (11.3 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

```
analyst@d61d9240af20:~$ 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)
```

9. Inspect network traffic (eth0):
   -i eth0: Capture data specifically from the eth0 interface.
   -v: Display detailed packet data.
   -c5: Capture 5 packets of data.

```
analyst@d61d9240af20:~$ sudo tcpdump -i eth0 -v -c5
tcpdump: listening on eth0, link-type EN10MB (Ethernet), capture size 262144 byt
09:18:18.019643 IP (tos 0x0, ttl 64, id 4749, offset 0, flags [DF], proto TCP (6
    d61d9240af20.5000 > nginx-us-west1-c.c.qwiklabs-terminal-vms-prod-00.interna
um 0x5891 (incorrect -> 0xa9b2), seq 3302246382:3302246445, ack 2931523961, win
 val 1264142509 ecr 244937377], length 63
09:18:18.019878 IP (tos 0x0, ttl 63, id 59724, offset 0, flags [DF], proto TCP (
    nginx-us-west1-c.c.qwiklabs-terminal-vms-prod-00.internal.38256 > d61d9240af
m 0x20fd (correct), ack 63, win 507, options [nop,nop,TS val 244937418 ecr 12641
09:18:18.030131 IP (tos 0x0, ttl 64, id 4750, offset 0, flags [DF], proto TCP (6
    d61d9240af20.5000 > nginx-us-west1-c.c.qwiklabs-terminal-vms-prod-00.interna
um 0x58ae (incorrect -> 0xece5), seq 63:155, ack 1, win 501, options [nop,nop,TS
37418], length 92
09:18:18.030397 IP (tos 0x0, ttl 63, id 59725, offset 0, flags [DF], proto TCP (
    nginx-us-west1-c.c.qwiklabs-terminal-vms-prod-00.internal.38256 > d61d9240af
m 0x208b (correct), ack 155, win 507, options [nop,nop,TS val 244937429 ecr 1264
09:18:18.038952 IP (tos 0x0, ttl 64, id 49446, offset 0, flags [DF], proto UDP (
    d61d9240af20.56683 > metadata.google.internal.domain: 24186+ PTR? 2.0.22.172
5 packets captured
10 packets received by filter
0 packets dropped by kernel
```

10. Capture network traffic

-i eth0: Capture data from the eth0 interface.
-nn: Do not attempt to resolve IP addresses or ports to names.This is best practice from a security perspective, as the lookup data may not be valid. It also prevents malicious actors from being alerted to an investigation.
-c9: Capture 9 packets of data and then exit.
port 80: Filter only port 80 traffic. This is the default HTTP port.
-w capture.pcap: Save the captured data to the named file.
&: This is an instruction to the Bash shell to run the command in the background.

```
analyst@d61d9240af20:~$ sudo tcpdump -i eth0 -nn -c9 port 80 -w capture.pcap &
[1] 12828
analyst@d61d9240af20:~$ tcpdump: listening on eth0, link-type EN10MB (Ethernet), capture size 262
es
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@d61d9240af20:~$ 9 packets captured
12 packets received by filter
0 packets dropped by kernel
```

11. Verify that data has been capture

```
analyst@d61d9240af20:~$ ls -l capture.pcap
-rw-r--r-- 1 root root 1455 Dec 12 09:22 capture.pcap
```

12. Filter the captured packet data
    - -nn: Disable port and protocol name lookup.
    - -r: Read capture data from the named file.
    - -v: Display detailed packet dat

```
nalyst@bb88ce441a42:~$ sudo tcpdump -nn -r capture.pcap -v
eading from file capture.pcap, link-type EN10MB (Ethernet)
9:33:10.305672 IP (tos 0x0, ttl 64, id 3455, offset 0, flags [DF], proto TCP (6),
   172.17.0.2.50098 > 74.125.142.101.80: Flags [S], cksum 0x8524 (incorrect -> 0xd
win 65320, options [mss 1420,sackOK,TS val 1775507883 ecr 0,nop,wscale 7], length
9:33:10.306673 IP (tos 0x60, ttl 126, id 0, offset 0, flags [DF], proto TCP (6), l
   74.125.142.101.80 > 172.17.0.2.50098: Flags [S.], cksum 0xb92c (correct), seq 3
446, win 65535, options [mss 1420,sackOK,TS val 1291387135 ecr 1775507883,nop,wsca
9:33:10.306726 IP (tos 0x0, ttl 64, id 3456, offset 0, flags [DF], proto TCP (6),
   172.17.0.2.50098 > 74.125.142.101.80: Flags [.], cksum 0x851c (incorrect -> 0xe
options [nop,nop,TS val 1775507884 ecr 1291387135], length 0
9:33:10.306775 IP (tos 0x0, ttl 64, id 3457, offset 0, flags [DF], proto TCP (6),
   172.17.0.2.50098 > 74.125.142.101.80: Flags [P.], cksum 0x8571 (incorrect -> 0x
, win 511, options [nop,nop,TS val 1775507884 ecr 1291387135], length 85: HTTP, le
      GET / HTTP/1.1
      Host: opensource.google.com
      User-Agent: curl/7.64.0
      Accept: */*
```