CEGEP VANIER COLLEGE
CENTRE FOR CONTINUING EDUCATION
Cybersecurity
420- 950-VA

Teacher: Samir Chebbine **Lab 6** **Mar 20, 2025**

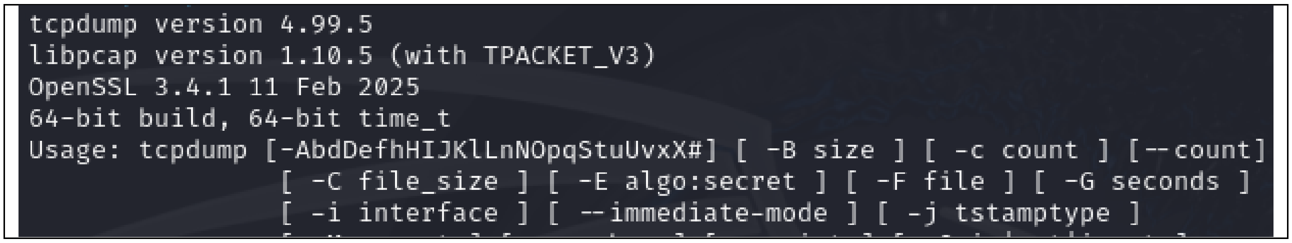
Lab 6: Network Monitoring and Intrusion Detection Systems

Complete all these following sections as explained in **class.** All *steps* were presented during class time.

Create and Submit a Word file ***Lab6CybersecurityYourName.doc*** which contains answers of Book Exercises and output screenshots for every project. Submit all Python scripts.

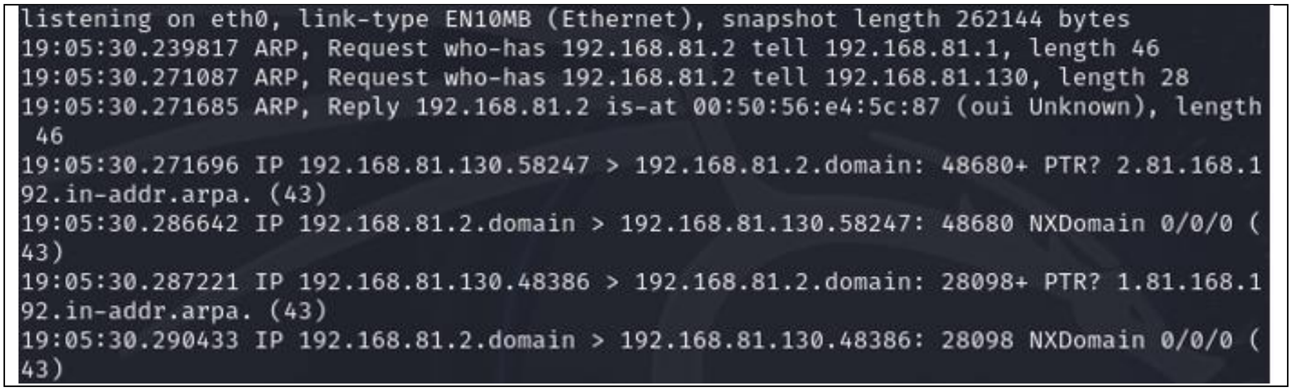
1. Network Monitoring:

a) Show command-line utility to dump the traffic on a network, allowing you to print out the headers of packets on a network interface, filter packets that match a certain expression as shown hereafter.



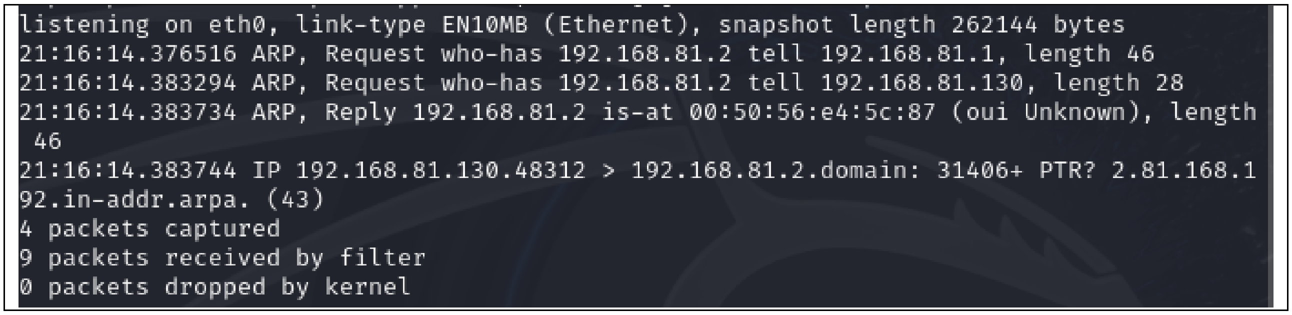
```
tcpdump version 4.99.5
libpcap version 1.10.5 (with TPACKET_V3)
OpenSSL 3.4.1 11 Feb 2025
64-bit build, 64-bit time_t
Usage: tcpdump [-AbdDefhHIJKlLnNOpqStuUvxX#] [ -B size ] [ -c count ] [--count]
                [ -C file_size ] [ -E algo:secret ] [ -F file ] [ -G seconds ]
                [ -i interface ] [ --immediate-mode ] [ -j tstamptype ]
```

b) Execute command-line to capture the packets of current network interface in Kali Linux as shown hereafter.



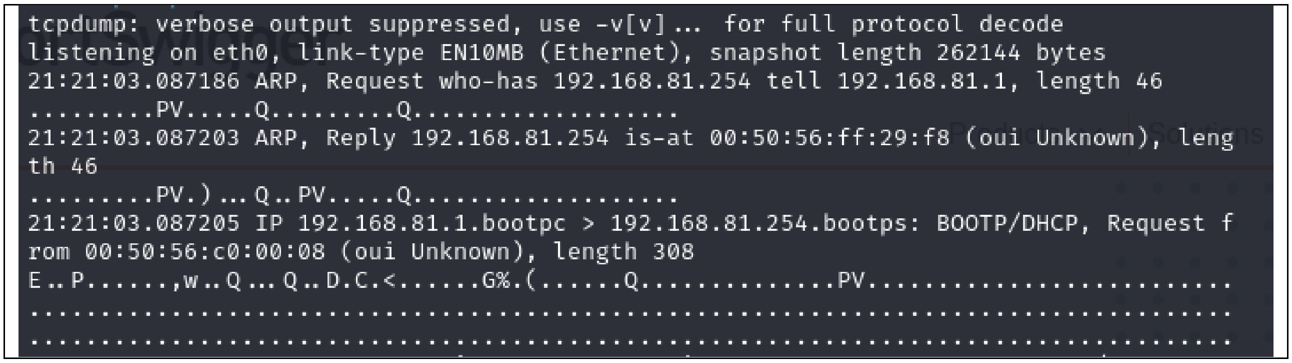
```
listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
19:05:30.239817 ARP, Request who-has 192.168.81.2 tell 192.168.81.1, length 46
19:05:30.271087 ARP, Request who-has 192.168.81.2 tell 192.168.81.130, length 28
19:05:30.271685 ARP, Reply 192.168.81.2 is-at 00:50:56:e4:5c:87 (oui Unknown), length
 46
19:05:30.271696 IP 192.168.81.130.58247 > 192.168.81.2.domain: 48680+ PTR? 2.81.168.1
92.in-addr.arpa. (43)
19:05:30.286642 IP 192.168.81.2.domain > 192.168.81.130.58247: 48680 NXDomain 0/0/0 (
43)
19:05:30.287221 IP 192.168.81.130.48386 > 192.168.81.2.domain: 28098+ PTR? 1.81.168.1
92.in-addr.arpa. (43)
19:05:30.290433 IP 192.168.81.2.domain > 192.168.81.130.48386: 28098 NXDomain 0/0/0 (
43)
```

c) Execute command-line to capture 4 packets from a specific network interface in Kali Linux as shown hereafter.



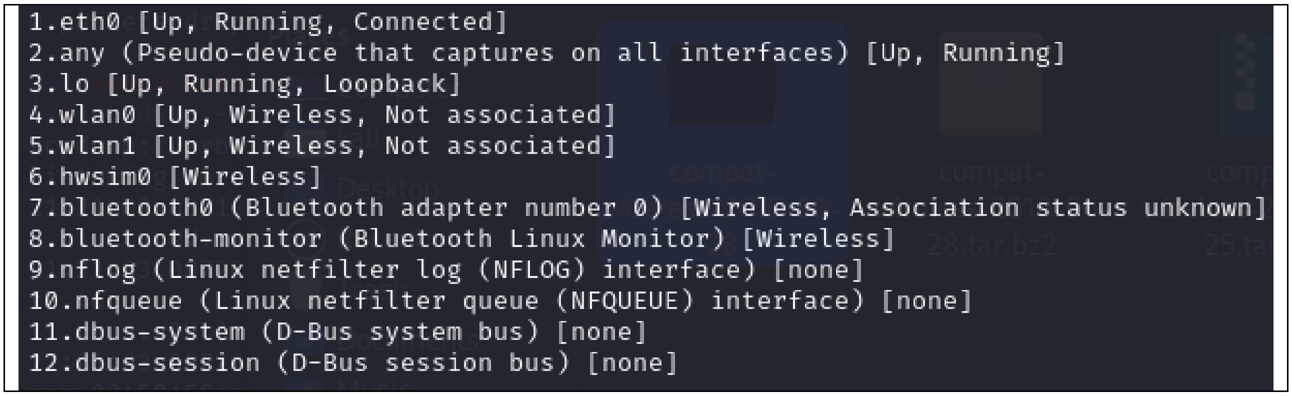
```
listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
21:16:14.376516 ARP, Request who-has 192.168.81.2 tell 192.168.81.1, length 46
21:16:14.383294 ARP, Request who-has 192.168.81.2 tell 192.168.81.130, length 28
21:16:14.383734 ARP, Reply 192.168.81.2 is-at 00:50:56:e4:5c:87 (oui Unknown), length
 46
21:16:14.383744 IP 192.168.81.130.48312 > 192.168.81.2.domain: 31406+ PTR? 2.81.168.1
92.in-addr.arpa. (43)
4 packets captured
9 packets received by filter
0 packets dropped by kernel
```

d) Execute command-line to capture packets from a specific network interface in ASCII format on Kali Linux in Kali Linux as shown hereafter.



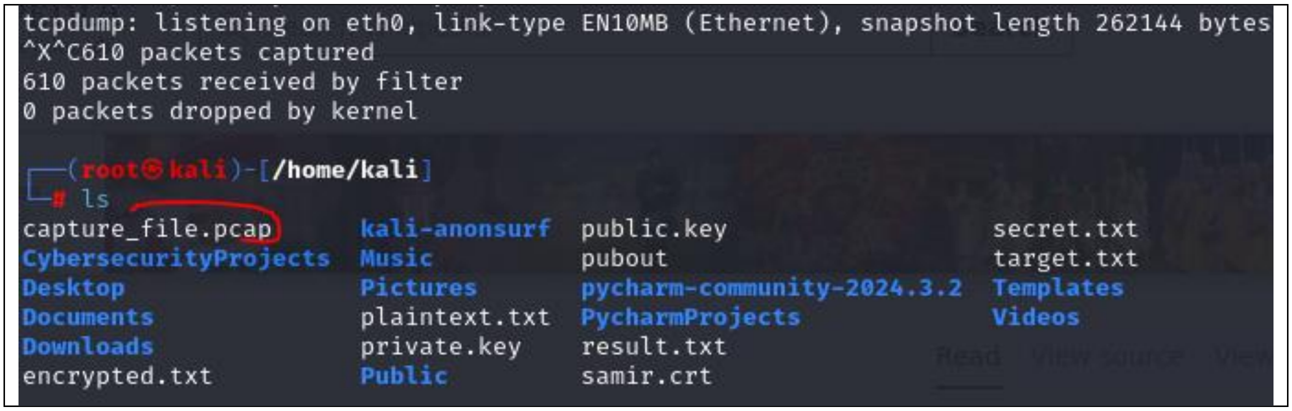
```
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
21:21:03.087186 ARP, Request who-has 192.168.81.254 tell 192.168.81.1, length 46
.........PV.....Q.........Q..................
21:21:03.087203 ARP, Reply 192.168.81.254 is-at 00:50:56:ff:29:f8 (oui Unknown), leng
th 46
.........PV.)...Q..PV.....Q..................
21:21:03.087205 IP 192.168.81.1.bootpc > 192.168.81.254.bootps: BOOTP/DHCP, Request f
rom 00:50:56:c0:00:08 (oui Unknown), length 308
E..P......,w..Q...Q..D.C.<......G%.(......Q.............PV........................
..................................................................................
..................................................................................
```

e) Execute command-line to display all available network interfaces as shown hereafter.



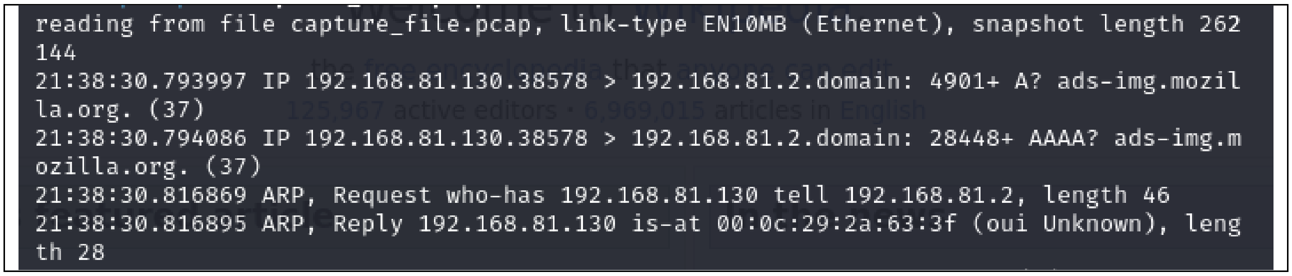
```
1.eth0 [Up, Running, Connected]
2.any (Pseudo-device that captures on all interfaces) [Up, Running]
3.lo [Up, Running, Loopback]
4.wlan0 [Up, Wireless, Not associated]
5.wlan1 [Up, Wireless, Not associated]
6.hwsim0 [Wireless]
7.bluetooth0 (Bluetooth adapter number 0) [Wireless, Association status unknown]
8.bluetooth-monitor (Bluetooth Linux Monitor) [Wireless]
9.nflog (Linux netfilter log (NFLOG) interface) [none]
10.nfqueue (Linux netfilter queue (NFQUEUE) interface) [none]
11.dbus-system (D-Bus system bus) [none]
12.dbus-session (D-Bus session bus) [none]
```

f) Execute command-line to save captured packets into a pcap file named (capture_file.pcap) in Kali Linux as shown hereafter.



```
tcpdump: listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
^X^C610 packets captured
610 packets received by filter
0 packets dropped by kernel

┌──(root㉿kali)-[/home/kali]
└─# ls
capture_file.pcap        kali-anonsurf   public.key                   secret.txt
CybersecurityProjects    Music           pubout                       target.txt
Desktop                  Pictures        pycharm-community-2024.3.2   Templates
Documents                plaintext.txt   PycharmProjects              Videos
Downloads                private.key     result.txt
encrypted.txt            Public          samir.crt
```

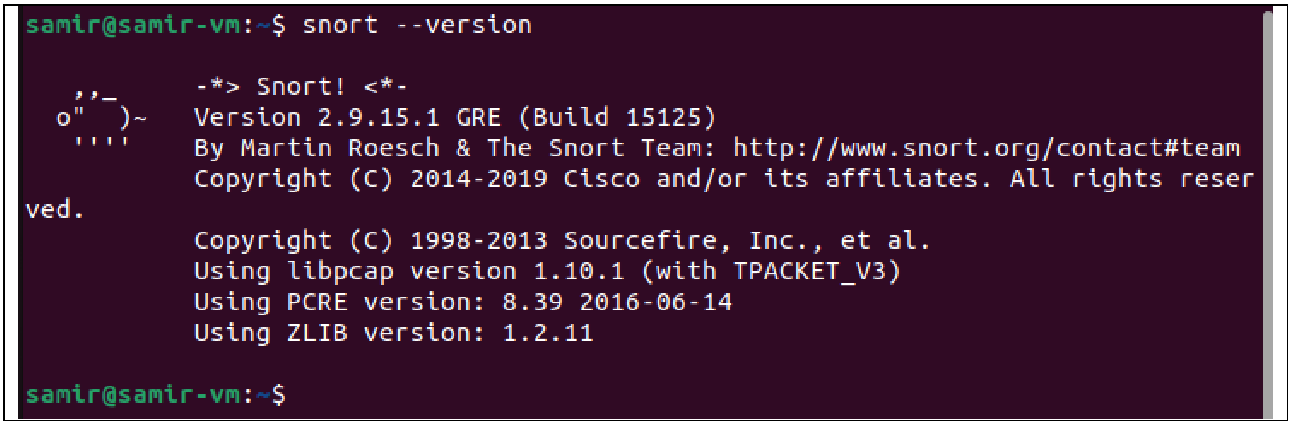
g) Execute command-line to read from captured file capture_file.pcap in Kali Linux as shown hereafter.



```
reading from file capture_file.pcap, link-type EN10MB (Ethernet), snapshot length 262
144
21:38:30.793997 IP 192.168.81.130.38578 > 192.168.81.2.domain: 4901+ A? ads-img.mozil
la.org. (37)
21:38:30.794086 IP 192.168.81.130.38578 > 192.168.81.2.domain: 28448+ AAAA? ads-img.m
ozilla.org. (37)
21:38:30.816869 ARP, Request who-has 192.168.81.130 tell 192.168.81.2, length 46
21:38:30.816895 ARP, Reply 192.168.81.130 is-at 00:0c:29:2a:63:3f (oui Unknown), leng
th 28
```

2. Snort as Intrusion Detection System:

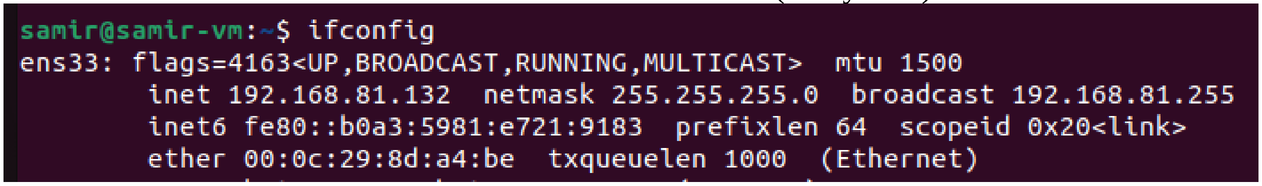
a) Install Linux Ubuntu Operating System as virtual machine. Do research on how to install and configure Snort 2.X on Ubuntu machine as shown hereafter.



```
samir@samir-vm:~$ snort --version

   ,,_     -*> Snort! <*-
  o"  )~   Version 2.9.15.1 GRE (Build 15125)
   ''''    By Martin Roesch & The Snort Team: http://www.snort.org/contact#team
           Copyright (C) 2014-2019 Cisco and/or its affiliates. All rights reser
ved.
           Copyright (C) 1998-2013 Sourcefire, Inc., et al.
           Using libpcap version 1.10.1 (with TPACKET_V3)
           Using PCRE version: 8.39 2016-06-14
           Using ZLIB version: 1.2.11

samir@samir-vm:~$
```

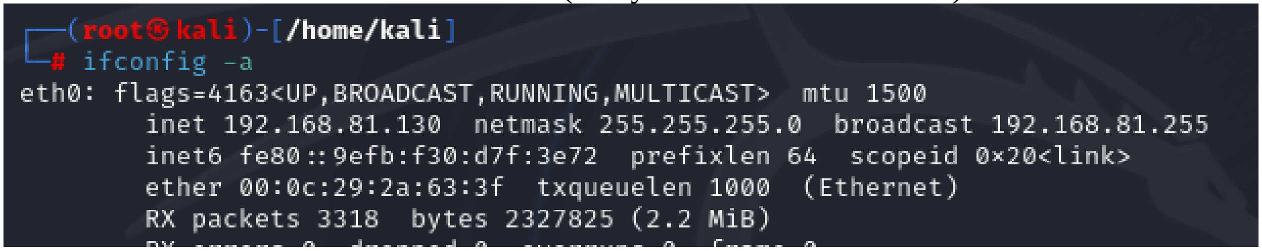
b) Start Snort as Instruction Detection program on Ubuntu machine as shown hereafter.

Ubuntu Linux machine IP address (in my case)



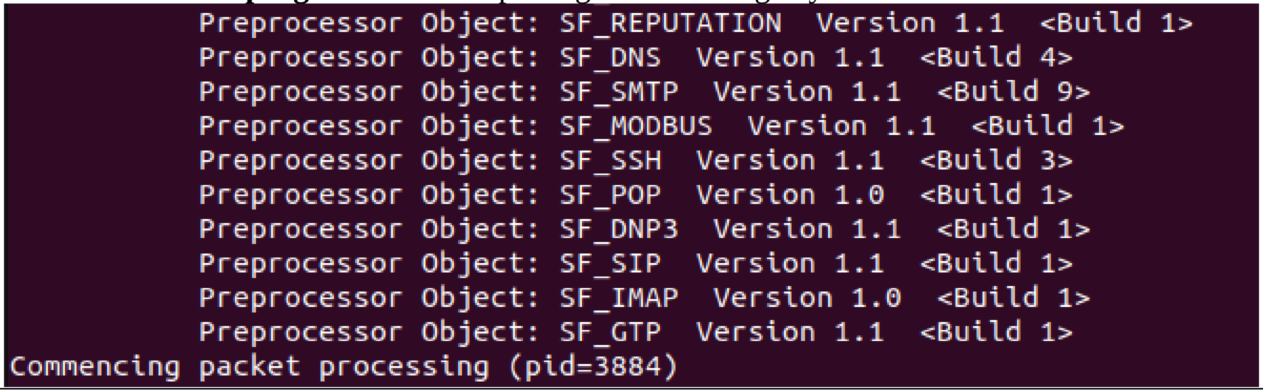
```
samir@samir-vm:~$ ifconfig
ens33: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.81.132  netmask 255.255.255.0  broadcast 192.168.81.255
        inet6 fe80::b0a3:5981:e721:9183  prefixlen 64  scopeid 0x20<link>
        ether 00:0c:29:8d:a4:be  txqueuelen 1000  (Ethernet)
```

Kali Linux machine (in my case Attacker machine)



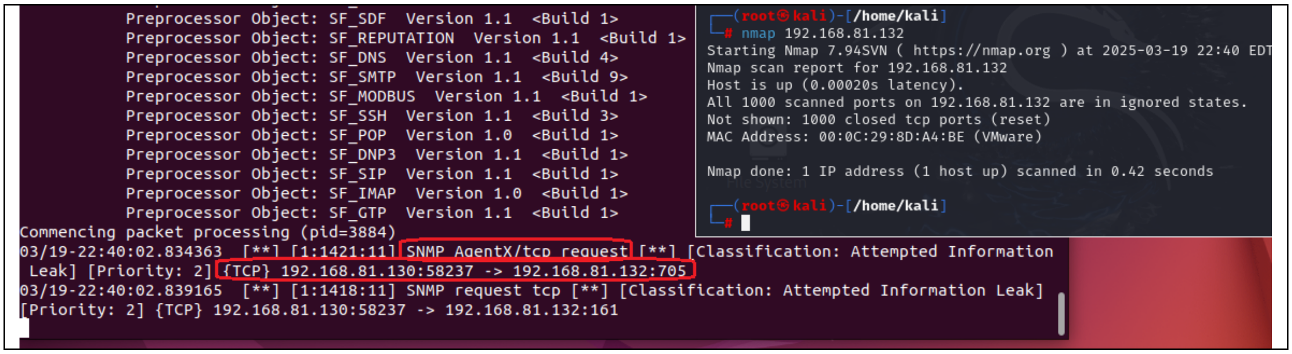
```
┌──(root㉿kali)-[/home/kali]
└─# ifconfig -a
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.81.130  netmask 255.255.255.0  broadcast 192.168.81.255
        inet6 fe80::9efb:f30:d7f:3e72  prefixlen 64  scopeid 0x20<link>
        ether 00:0c:29:2a:63:3f  txqueuelen 1000  (Ethernet)
        RX packets 3318  bytes 2327825 (2.2 MiB)
```

Execute Snort program to start capturing and detecting any Intrusion in **Ubuntu machine**



```
           Preprocessor Object: SF_REPUTATION  Version 1.1  <Build 1>
           Preprocessor Object: SF_DNS  Version 1.1  <Build 4>
           Preprocessor Object: SF_SMTP  Version 1.1  <Build 9>
           Preprocessor Object: SF_MODBUS  Version 1.1  <Build 1>
           Preprocessor Object: SF_SSH  Version 1.1  <Build 3>
           Preprocessor Object: SF_POP  Version 1.0  <Build 1>
           Preprocessor Object: SF_DNP3  Version 1.1  <Build 1>
           Preprocessor Object: SF_SIP  Version 1.1  <Build 1>
           Preprocessor Object: SF_IMAP  Version 1.0  <Build 1>
           Preprocessor Object: SF_GTP  Version 1.1  <Build 1>
Commencing packet processing (pid=3884)
```

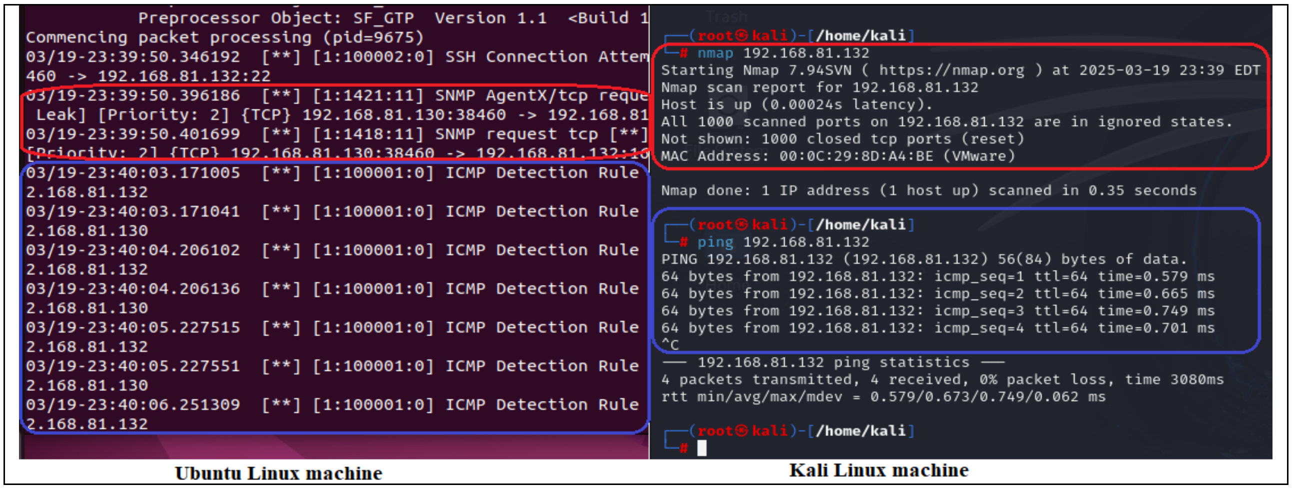
c) Perform intrusion from Kali Linux attacker machine by sending SNMP message toward Ubuntu machine (nmap IPUbuntoMachine for ex) and you should notice detection of SNMP messages in Ubuntu machine as shown hereafter.



```
          Preprocessor Object: SF_SDF  Version 1.1  <Build 1>
          Preprocessor Object: SF_REPUTATION  Version 1.1  <Build 1>
          Preprocessor Object: SF_DNS  Version 1.1  <Build 4>
          Preprocessor Object: SF_SMTP  Version 1.1  <Build 9>
          Preprocessor Object: SF_MODBUS  Version 1.1  <Build 1>
          Preprocessor Object: SF_SSH  Version 1.1  <Build 3>
          Preprocessor Object: SF_POP  Version 1.0  <Build 1>
          Preprocessor Object: SF_DNP3  Version 1.1  <Build 1>
          Preprocessor Object: SF_SIP  Version 1.1  <Build 1>
          Preprocessor Object: SF_IMAP  Version 1.0  <Build 1>
          Preprocessor Object: SF_GTP  Version 1.1  <Build 1>
Commencing packet processing (pid=3884)
03/19-22:40:02.834363  [**] [1:1421:11] SNMP AgentX/tcp request [**] [Classification: Attempted Information
 Leak] [Priority: 2] {TCP} 192.168.81.130:58237 -> 192.168.81.132:705
03/19-22:40:02.839165  [**] [1:1418:11] SNMP request tcp [**] [Classification: Attempted Information Leak]
[Priority: 2] {TCP} 192.168.81.130:58237 -> 192.168.81.132:161
```

```
┌──(root㉿kali)-[/home/kali]
└─# nmap 192.168.81.132
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-03-19 22:40 EDT
Nmap scan report for 192.168.81.132
Host is up (0.00020s latency).
All 1000 scanned ports on 192.168.81.132 are in ignored states.
Not shown: 1000 closed tcp ports (reset)
MAC Address: 00:0C:29:8D:A4:BE (VMware)

Nmap done: 1 IP address (1 host up) scanned in 0.42 seconds

┌──(root㉿kali)-[/home/kali]
└─#
```

d) Do research on how to add LOCAL RULES to include custom ICMP rule to detect incoming ICMP packets in Snort. Submit the new edited local rule configuration file.
Perform ping from Kali Linux attacker machine by sending ICMP message toward Ubuntu machine (ping IPUbuntoMachine for ex) and you should notice detection of ICMP messages in Ubuntu machine as shown hereafter.



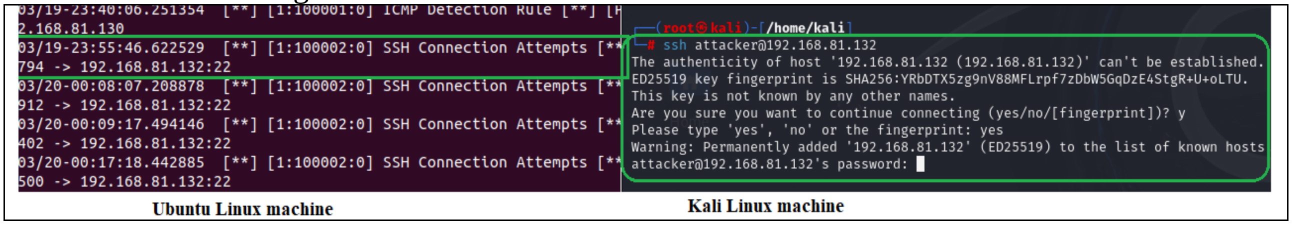
```
          Preprocessor Object: SF_GTP  Version 1.1  <Build 1
Commencing packet processing (pid=9675)
03/19-23:39:50.346192  [**] [1:100002:0] SSH Connection Attem
460 -> 192.168.81.132:22
03/19-23:39:50.396186  [**] [1:1421:11] SNMP AgentX/tcp reque
 Leak] [Priority: 2] {TCP} 192.168.81.130:38460 -> 192.168.81
03/19-23:39:50.401699  [**] [1:1418:11] SNMP request tcp [**]
[Priority: 2] {TCP} 192.168.81.130:38460 -> 192.168.81.132:16
03/19-23:40:03.171005  [**] [1:100001:0] ICMP Detection Rule
2.168.81.132
03/19-23:40:03.171041  [**] [1:100001:0] ICMP Detection Rule
2.168.81.130
03/19-23:40:04.206102  [**] [1:100001:0] ICMP Detection Rule
2.168.81.132
03/19-23:40:04.206136  [**] [1:100001:0] ICMP Detection Rule
2.168.81.130
03/19-23:40:05.227515  [**] [1:100001:0] ICMP Detection Rule
2.168.81.132
03/19-23:40:05.227551  [**] [1:100001:0] ICMP Detection Rule
2.168.81.130
03/19-23:40:06.251309  [**] [1:100001:0] ICMP Detection Rule
2.168.81.132
```

Ubuntu Linux machine

```
┌──(root㉿kali)-[/home/kali]
└─# nmap 192.168.81.132
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-03-19 23:39 EDT
Nmap scan report for 192.168.81.132
Host is up (0.00024s latency).
All 1000 scanned ports on 192.168.81.132 are in ignored states.
Not shown: 1000 closed tcp ports (reset)
MAC Address: 00:0C:29:8D:A4:BE (VMware)

Nmap done: 1 IP address (1 host up) scanned in 0.35 seconds

┌──(root㉿kali)-[/home/kali]
└─# ping 192.168.81.132
PING 192.168.81.132 (192.168.81.132) 56(84) bytes of data.
64 bytes from 192.168.81.132: icmp_seq=1 ttl=64 time=0.579 ms
64 bytes from 192.168.81.132: icmp_seq=2 ttl=64 time=0.665 ms
64 bytes from 192.168.81.132: icmp_seq=3 ttl=64 time=0.749 ms
64 bytes from 192.168.81.132: icmp_seq=4 ttl=64 time=0.701 ms
^C
── 192.168.81.132 ping statistics ──
4 packets transmitted, 4 received, 0% packet loss, time 3080ms
rtt min/avg/max/mdev = 0.579/0.673/0.749/0.062 ms

┌──(root㉿kali)-[/home/kali]
└─#
```

Kali Linux machine

e) Do research on how to add LOCAL RULES to detect in Snort SSH connection running on TCP 22 by default. Submit the new edited local rule configuration file.
Perform ssh test from Kali Linux attacker machine by sending ICMP message toward Ubuntu machine (ssh attacker@192.168.81.132 for ex) and you should notice detection of SSH connection messages in Ubuntu machine as shown hereafter.



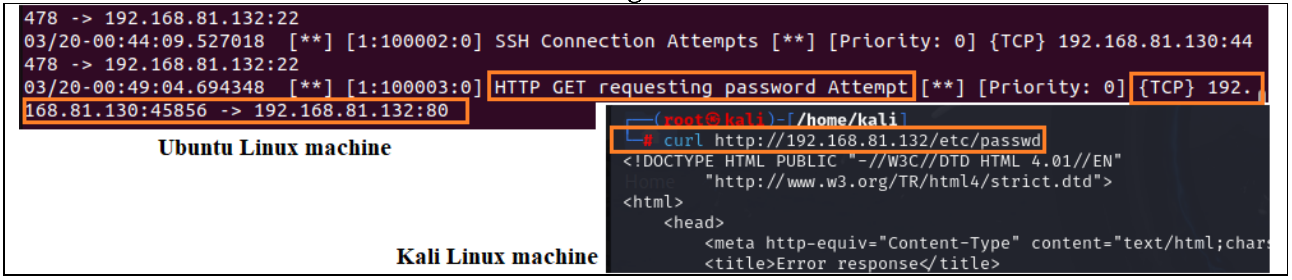
```
03/19-23:40:06.251354  [**] [1:100001:0] ICMP Detection Rule [**] [H
2.168.81.130
03/19-23:55:46.622529  [**] [1:100002:0] SSH Connection Attempts [**
794 -> 192.168.81.132:22
03/20-00:08:07.208878  [**] [1:100002:0] SSH Connection Attempts [**
912 -> 192.168.81.132:22
03/20-00:09:17.494146  [**] [1:100002:0] SSH Connection Attempts [**
402 -> 192.168.81.132:22
03/20-00:17:18.442885  [**] [1:100002:0] SSH Connection Attempts [**
500 -> 192.168.81.132:22
```

Ubuntu Linux machine

```
┌──(root㉿kali)-[/home/kali]
└─# ssh attacker@192.168.81.132
The authenticity of host '192.168.81.132 (192.168.81.132)' can't be established.
ED25519 key fingerprint is SHA256:YRbDTX5zg9nV88MFLrpf7zDbW5GqDzE4StgR+U+oLTU.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? y
Please type 'yes', 'no' or the fingerprint: yes
Warning: Permanently added '192.168.81.132' (ED25519) to the list of known hosts
attacker@192.168.81.132's password:
```

Kali Linux machine

f) Do research on how to add LOCAL RULES to detect HTTP connection running on TCP 80 by default in Snort. Submit the new edited local rule configuration file.
Perform curl test from Kali Linux attacker machine by sending HTTP GET request containing content passwd (for ex) toward Ubuntu machine (http server) and you should notice detection of HTTP connection messages in Ubuntu machine as shown hereafter.



```
478 -> 192.168.81.132:22
03/20-00:44:09.527018  [**] [1:100002:0] SSH Connection Attempts [**] [Priority: 0] {TCP} 192.168.81.130:44
478 -> 192.168.81.132:22
03/20-00:49:04.694348  [**] [1:100003:0] HTTP GET requesting password Attempt [**] [Priority: 0] {TCP} 192.
168.81.130:45856 -> 192.168.81.132:80
```

Ubuntu Linux machine

```
┌──(root㉿kali)-[/home/kali]
└─# curl http://192.168.81.132/etc/passwd
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
        "http://www.w3.org/TR/html4/strict.dtd">
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html;char
        <title>Error response</title>
```

Kali Linux machine