

Cybersecurity Internship – Task 4

Name: Tirth Vaja

Task: 4 – Network Intrusion Detection System (NIDS)

Task Objective:

To set up a Network-based Intrusion Detection System (NIDS) using Snort on a CentOS environment, configure detection rules for suspicious activity such as ICMP packets, monitor network traffic in real-time, and observe alerts generated by Snort.

Tools and Technologies Used:

- Snort 2.9.20 (installed from source)
- CentOS 8 (running in VMware)
- Terminal & Ping utility (for traffic generation)
- Text editors: nano

Implementation Steps:

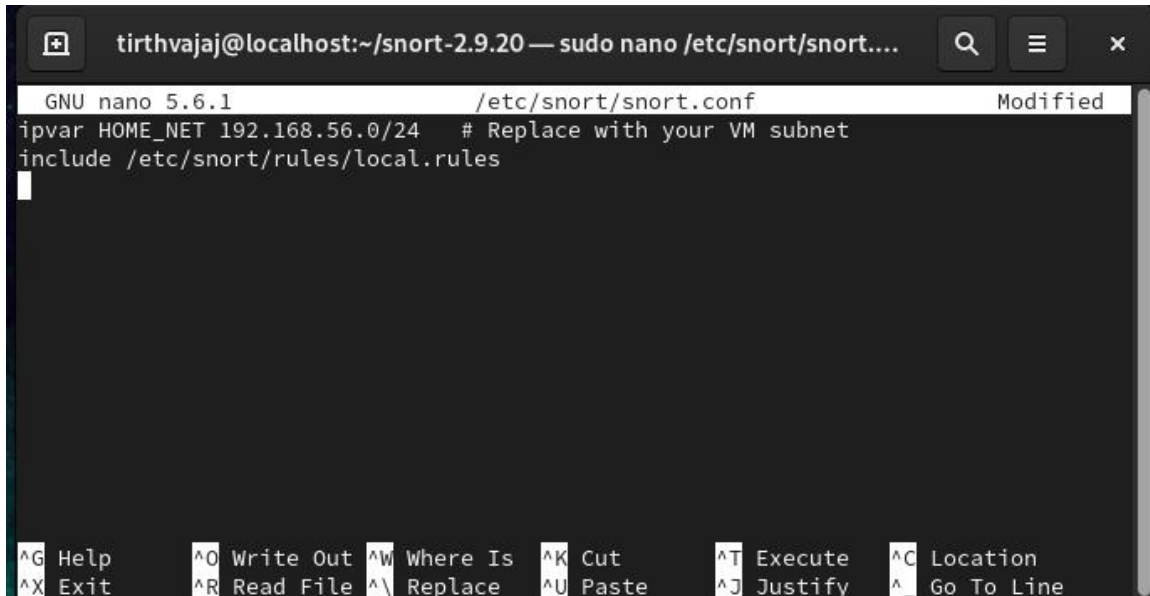
1. Installed Required Packages and Dependencies for Snort using YUM.
2. Downloaded and compiled DAQ 2.0.7 and Snort 2.9.20 from source.
3. Configured Snort using snort.conf and set the HOME_NET variable according to VM IP range.
4. Created a custom rule in /etc/snort/rules/local.rules to detect ICMP (ping) traffic:

```
alert icmp any any -> $HOME_NET any (msg:"ICMP Packet Detected"; sid:1000001; rev:1;)
```

5. Ran Snort in console mode using the command:

```
sudo snort -A console -q -c /etc/snort/snort.conf -i ens160
```

6. Triggered a ping from host system to VM and successfully received real-time alerts.

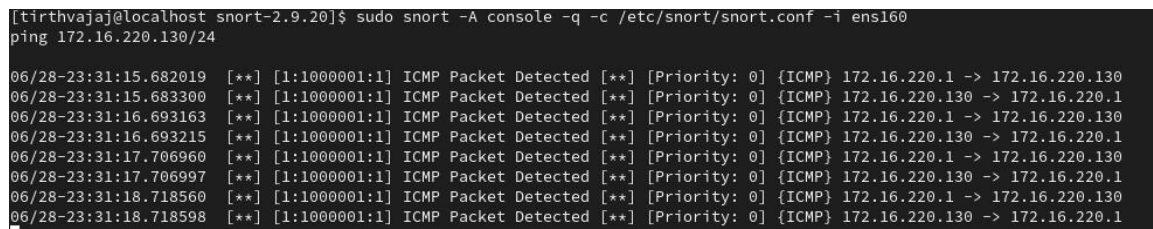


```
tirthvajaj@localhost:~/snort-2.9.20 — sudo nano /etc/snort/snort....
GNU nano 5.6.1 /etc/snort/snort.conf Modified
ipvar HOME_NET 192.168.56.0/24 # Replace with your VM subnet
include /etc/snort/rules/local.rules
^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute   ^C Location
^X Exit      ^R Read File ^\ Replace   ^U Paste     ^J Justify   ^_ Go To Line
```

Observations and Alerts:

Snort successfully detected ICMP traffic. Each ping generated an alert message in the terminal such as:

```
[**] [1:1000001:1] ICMP Packet Detected [**]
```



```
[tirthvajaj@localhost snort-2.9.20]$ sudo snort -A console -q -c /etc/snort/snort.conf -i ens160
ping 172.16.220.130/24
06/28-23:31:15.682019  [**] [1:1000001:1] ICMP Packet Detected [**] [Priority: 0] {ICMP} 172.16.220.1 -> 172.16.220.130
06/28-23:31:15.683300  [**] [1:1000001:1] ICMP Packet Detected [**] [Priority: 0] {ICMP} 172.16.220.130 -> 172.16.220.1
06/28-23:31:16.693163  [**] [1:1000001:1] ICMP Packet Detected [**] [Priority: 0] {ICMP} 172.16.220.1 -> 172.16.220.130
06/28-23:31:16.693215  [**] [1:1000001:1] ICMP Packet Detected [**] [Priority: 0] {ICMP} 172.16.220.130 -> 172.16.220.1
06/28-23:31:17.706960  [**] [1:1000001:1] ICMP Packet Detected [**] [Priority: 0] {ICMP} 172.16.220.1 -> 172.16.220.130
06/28-23:31:17.706997  [**] [1:1000001:1] ICMP Packet Detected [**] [Priority: 0] {ICMP} 172.16.220.130 -> 172.16.220.1
06/28-23:31:18.718560  [**] [1:1000001:1] ICMP Packet Detected [**] [Priority: 0] {ICMP} 172.16.220.1 -> 172.16.220.130
06/28-23:31:18.718598  [**] [1:1000001:1] ICMP Packet Detected [**] [Priority: 0] {ICMP} 172.16.220.130 -> 172.16.220.1
```

Response Mechanism:

The Snort engine responded by logging the ICMP activity with timestamp and source-destination IPs, fulfilling real-time alerting functionality.

Visualization (Optional):

As per task guidelines, visualization was optional. For this task, console-based alerts were used instead of graphical dashboards.

Conclusion:

The objective of setting up a working NIDS using Snort was successfully achieved. By writing custom detection rules, capturing live network traffic, and responding with console-based alerts, the system was tested thoroughly. This task enhanced practical knowledge in intrusion detection and network monitoring, and served as a foundation for deeper security implementation.