

Computer Security Project Report

CSE 406

Tool: Snort

Supervisor:

Sayem Hasan
Adjunct Lecturer
CSE, BUET

Presented By:

Farhan Tanvir - 1805073
Kowsar Mahmud Pappu - 1805075

Level-4 Term-1

Department of CSE

Bangladesh University of Engineering & Technology

13 September, 2023

1 Tool Overview

Snort is an open source network intrusion detection and prevention system (IDS/IPS) tool. It is widely used for real time traffic analysis and packet logging on IP networks. Here is an overview of Snort's key features and functionalities:

1. **Intrusion Detection:** Snort can detect and alert on various types of network-based attacks, including
 - Unauthorized accesses
 - DoS attacks
 - Malware infections
2. **Rule-Based Detection:** Snort uses a rule-based detection engine to analyze network traffic and compare it against a set of predefined rules. These rules can be customized to match specific attack signatures or patterns
3. **Protocol Analysis:** Snort supports the analysis of multiple network protocols, including TCP, UDP, ICMP, and more. It can inspect packet headers and payloads to identify anomalies or malicious activities
4. **Traffic Analysis and Statistics:** Snort provides detailed statistics and reports on network traffic, including the number of alerts triggered, traffic volume, and attack trends. This information can be used for network monitoring and security analysis

2 Source Code Overview

The central component of Snort setup resides in its configuration file, typically located at */etc/snort/snort.conf*. Within this configuration file, crucial details regarding both the internal and external network settings utilized by the tool are specified. Additionally, it serves as the repository for defining ports and protocols. Furthermore, provisions for implementing whitelists and blacklists are embedded in this configuration file. **It comes**

equipped with a set of default rules, while users have the flexibility to craft their own custom rules in */etc/snort/rules/local.rules*. The rules directory contains an array of rule definitions distributed across multiple files and the local.rules file (initially empty) serves as a canvas for users to append their personalized rule set as per their requirement.

3 Feature Description

Snort can be configured to operate in three different modes.

1. Sniffer Mode:

- In Sniffer mode, Snort functions as a passive network sniffer or packet logger
- It generates alerts and logs based on predefined rules but does not actively block or prevent network traffic
- Sniffer mode is typically used for network monitoring, packet capture, and analysis to understand network activity and identify potential security threats

2. Logger Mode:

- In Logger Mode, Snort captures and logs network packets but does not generate alerts or take immediate action
- It is useful for network administrators who want to record network traffic for later analysis or auditing without immediate intrusion detection or prevention
- While it does not actively prevent attacks, it provides valuable information for post-incident analysis

3. Network Intrusion Detection System (NIDS) Mode:

- In this mode, Snort actively monitors network traffic, analyzes packets in real-time, and generates alerts for suspicious or malicious activity based on predefined rules

- Unlike the previous modes, NIDS Mode can take action to block or drop malicious traffic, making it an intrusion prevention system (IPS)

4 Feature Documentation

1. Packet Sniffing (Sniffer Mode):

- **sudo nano /etc/snort/snort.conf**
 - This command opens snort.conf in the nano editor

```
seed@CSE406:/home/ftp$ sudo nano /etc/snort/snort.conf
```

- Check if `ipvar HOME_NET` is set to your subnet (e.g. 10.0.0.0/24)

```
ipvar HOME_NET 10.0.0.0/24
```

- Comment all the included custom rules like **include \$RULE_PATH/<filename>.rule** by appending **#** at the front so that we can write our own rules and test them

[illegible]

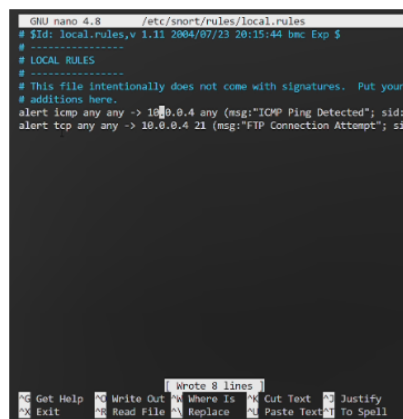
- Control + O → Enter → Save changes in nano
Control + X → Exit nano
- **sudo snort -T -i eth0 -c /etc/snort/snort.conf**
- This command tests if everything is alright in the configuration file

```
seed@CSE406:/home/ftp$ sudo snort -T -i eth0 -c /etc/snort/snort.conf
```

- **sudo nano /etc/snort/rules/local.rules**
- This command opens the **local.rules** file in nano

```
seed@CSE406:/home/ftp$ sudo nano /etc/snort/rules/local.rules
```

- **alert icmp any any -> 10.0.0.4 any (msg:"ICMP Ping Detected"; sid:100001; rev:1;)**
- Write this rule in the last line of the local.rules file. This is a rule that detects any ICMP packet, that is, ping towards the home network from anywhere and generates an alert
- **alert tcp any any -> 10.0.0.4 21 (msg:"FTP Connection Attempt"; sid:100003; rev:1;)**
- This rule detects any FTP connection attempt



```
GNU nano 4.8 /etc/snort/rules/local.rules
# $Id: local.rules,v 1.11 2004/07/23 20:15:44 bmc Exp $
#
# LOCAL RULES
#
# This file intentionally does not come with signatures. Put your
# additions here.
alert icmp any any -> 10.0.0.4 any (msg:"ICMP Ping Detected"; sid:100001; rev:1;)
alert tcp any any -> 10.0.0.4 21 (msg:"FTP Connection Attempt"; sid:100003; rev:1;)

Wrote 8 lines
Get Help Write Out Where Is Cut Text Justify
Exit Read File Replace Paste Text To Spell
```

- Save and Exit Nano

- **sudo snort -q -i eth0 -A console -c /etc/snort/snort.conf**
- This command starts Snort in quiet mode, sniffs packets from the eth0 interface and prints an alert on the console

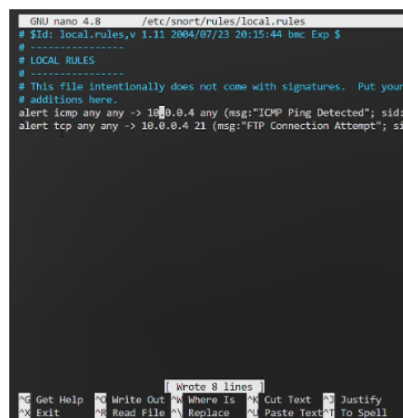
```
seed@CSE406:/home/ftp$ sudo snort -q -i eth0 -A console -c /etc/snort/snort.conf
```

2. Packet Logging (Logger Mode):

- **sudo nano /etc/snort/rules/local.rules**
- This command opens the local rules file in nano

```
seed@CSE406:/home/ftp$ sudo nano /etc/snort/rules/local.rules
```

- **alert tcp any any -> 10.0.0.4 21 (msg:"FTP Connection Attempt"; sid:100003; rev:1;)**
- Write this rule in the last line of the local.rules file. This rule detects any FTP connection attempt



```
GNU nano 4.8 /etc/snort/rules/local.rules
# $Id: local.rules,v 1.11 2004/07/23 20:15:44 bmc Exp $
#
# LOCAL RULES
# -----
# This file intentionally does not come with signatures. Put your
# additions here.
alert icmp any any -> 10.0.0.4 any (msg:"ICMP Ping Detected"; sid:
alert tcp any any -> 10.0.0.4 21 (msg:"FTP Connection Attempt"; sid:
Write 8 lines
Get Help Write Out Where Is Cut Text Justify
Exit Read File Replace Paste Text To Spell
```

- **sudo snort -q -l /var/log/snort -i eth0 -A console -c /etc/snort/snort.conf**
- This command starts Snort in quiet mode, sniffs packets from the eth0 interface, logs everything about it in the /var/log/snort directory and prints an alert on the console

```
seed@CSE406:/home/ftp$ sudo snort -q -l /var/log/snort -i eth0 -A console -c /etc/snort/snort.conf
```

5 Feature Demonstration

1. Packet Sniffing (Sniffer Mode):

- From attacker machine:
ping 10.0.0.4
- This command sends ICMP packets to the host machine (10.0.0.4)

```
seed@CSE406New:/home/kowsar156$ ping 10.0.0.4
PING 10.0.0.4 (10.0.0.4) 56(84) bytes of data.
64 bytes from 10.0.0.4: icmp_seq=1 ttl=64 time=1.45 ms
64 bytes from 10.0.0.4: icmp_seq=2 ttl=64 time=1.15 ms
64 bytes from 10.0.0.4: icmp_seq=3 ttl=64 time=1.27 ms
```

- Now go to the terminal of the host machine (which is snort enabled) and see alerts

```
seed@CSE406:/home/kowsar156$ sudo snort -q -i eth0 -A console -c /etc/snort/snort.conf
09/13-05:27:23.642137  [**] [1:100001:1] ICMP Ping Detected [**] [Priority: 0] {ICMP} 10.0.0.5 -> 10.0.0.4
09/13-05:27:24.643650  [**] [1:100001:1] ICMP Ping Detected [**] [Priority: 0] {ICMP} 10.0.0.5 -> 10.0.0.4
09/13-05:27:25.645043  [**] [1:100001:1] ICMP Ping Detected [**] [Priority: 0] {ICMP} 10.0.0.5 -> 10.0.0.4
```

2. Packet Logging (Logger Mode):

- From attacker machine:
ping 10.0.0.4
- This command sends ICMP packets to the host machine (10.0.0.4)
ftp 10.0.0.4
- This command tries to establish an FTP connection with the host machine (10.0.0.4)

```
seed@CSE406New:/home/kowsar156$ ping 10.0.0.4
PING 10.0.0.4 (10.0.0.4) 56(84) bytes of data.
64 bytes from 10.0.0.4: icmp_seq=1 ttl=64 time=2.03 ms
64 bytes from 10.0.0.4: icmp_seq=2 ttl=64 time=0.926 ms
64 bytes from 10.0.0.4: icmp_seq=3 ttl=64 time=2.32 ms
^C
--- 10.0.0.4 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2023ms
rtt min/avg/max/mdev = 0.926/1.760/2.320/0.601 ms
seed@CSE406New:/home/kowsar156$ ftp 10.0.0.4
```

- Now go to the terminal of the host machine (which is snort enabled) and see alerts

```
seed@CSE406:/home/kowsar156$ sudo snort -q -l /var/log/snort
-i eth0 -A console -c /etc/snort/snort.conf
09/13-06:18:36.251240  [**] [1:100001:1] ICMP Ping Detected [
**] [Priority: 0] {ICMP} 10.0.0.5 -> 10.0.0.4
09/13-06:18:37.252008  [**] [1:100001:1] ICMP Ping Detected [
**] [Priority: 0] {ICMP} 10.0.0.5 -> 10.0.0.4
09/13-06:18:38.275259  [**] [1:100001:1] ICMP Ping Detected [
**] [Priority: 0] {ICMP} 10.0.0.5 -> 10.0.0.4
09/13-06:19:09.667456  [**] [1:100003:1] FTP Connection Attem
pt [**] [Priority: 0] {TCP} 10.0.0.5:36956 -> 10.0.0.4:21
```

- From host machine:
cd /var/log - This command opens the directory where Snort is waiting with the log files
sudo chmod 777 snort - This command gives necessary permissions to enter into the Snort folder
cd snort - Through this command we enter the /var/log/snort directory where all the log files are kept
sudo chmod 777 snort.log.1694527894 - This command serves with necessary permissions to work on Wireshark with the log file

```
seed@CSE406:/home/ftp$ cd /var/log
seed@CSE406:/var/log$ sudo chmod 777 snort
seed@CSE406:/var/log$ cd snort
seed@CSE406:/var/log/snort$ sudo chmod 777 snort.log.1694527894
```

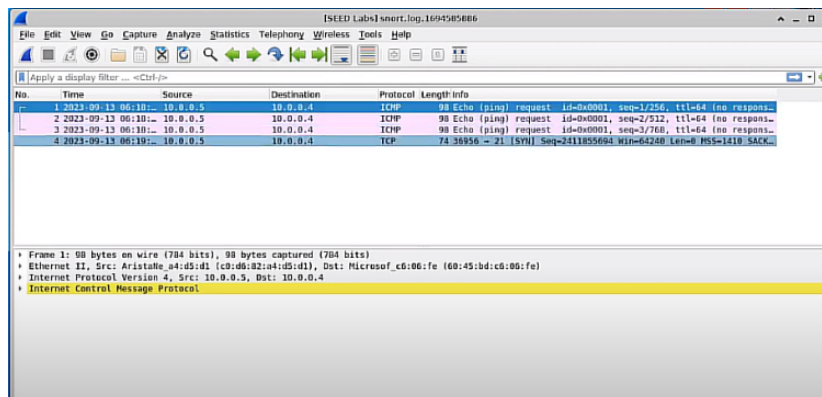
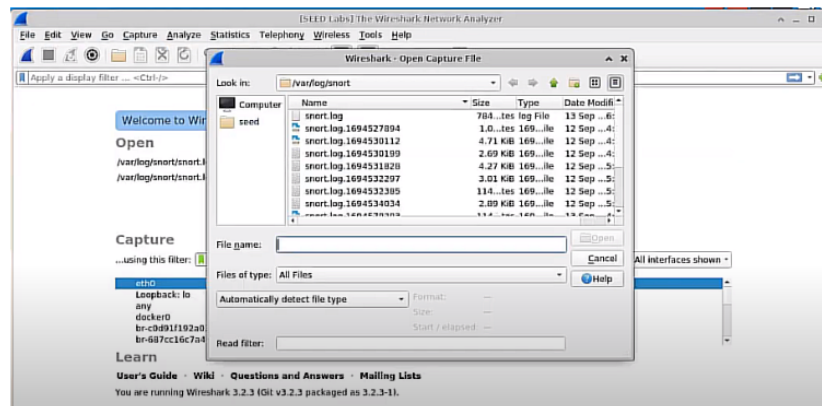
- **vncserver -localhost no**
- This allows the VM to be viewed by the Virtual Network viewer

applications (e.g. TigerShark, RealVNC)

```
seed@CSE406:/var/log/snort$ vncserver -localhost no

New Xtigervnc server 'CSE406.ga4qdyugyre5oc12vrxyhuod.hx.internal.cloudapp.net:1 (seed)' on port 5901 for display :1.
Use xtigervncviewer -SecurityTypes VncAuth,TLSTVnc -passwd /home/seed/.vnc/passwd CSE406.ga4qdyugyre5oc12vrxyhuod.hx.in
ternal.cloudapp.net:1 to connect to the VNC server.
```

- Connect to your VM through TigerShark/RealVNC
- After entering the VM through TigerShark/RealVNC, open WireShark → File → Open → /var/log/snort → snort.log.1694527894



6 Youtube Link

<https://www.youtube.com/watch?v=XaeAOnjMl-o>