

Network Intrusion Detection System Report

Project Name: Network Intrusion Detection System
using Suricata

Internship Program: CodeAlpha Cyber Security
Internship

Intern Name: Mahnoor Shafi

Date: [20-8-2025 - 20-9-2025]

Table of Content:

Introduction

Tools & Requirements

Project Objective

Implementation Steps

4.1 Installing Suricata

4.2 Running Suricata in IDS Mode

4.3 Creating Custom Rules

4.4 Testing with Ping/Nmap

4.5 Checking Alerts in Logs

Output & Results

Advantages of IDS

Real-World Applications

Conclusion

1. Introduction

A **Network Intrusion Detection System (NIDS)** is a cybersecurity tool that monitors network traffic and detects suspicious or malicious activities.

In this project, I implemented **Suricata**, an open-source IDS, to monitor network traffic, create custom rules, and detect attacks like **ICMP pings**.

2. Tools & Requirements

Linux VM

Suricata IDS

ping utility (for ICMP test)

Text editor (nano/vim)

Suricata-Update for community rules

3. Project Objective

The main objectives were:

Install and configure Suricata IDS.

Write and test custom detection rules.

Generate real attack traffic (ping).

Detect and log alerts in Suricata logs.

4. Implementation Steps

4.1 Installing Suricata

Suricata was installed on Linux using the official repository:

```
sudo add-apt-repository ppa:oisf/suricata-stable -y  
sudo apt update  
sudo apt install suricata -y
```

A screenshot of a terminal window on a Kali Linux system. The terminal shows the execution of the commands to add the PPA, update the package list, and install Suricata. The output shows the progress of downloading and installing the packages. The terminal window has a dark background with a Kali Linux logo watermark. The top of the window shows the title bar with 'kali@kali' and some system icons on the right.

```
kali@kali ~  
zsh: corrupt history file /home/kali/.zsh_history  
kali@kali ~  
$ sudo apt update  
[sudo] password for kali:  
$ sudo apt install suricata -y  
[sudo] password for kali:  
Get:2 https://packages.wazuh.com/4.x/apt stable InRelease [17.3 kB]  
Get:3 https://packages.wazuh.com/4.x/apt stable/main amd64 Packages [46.8 kB]  
Get:1 http://kali.download/kali kali-rolling InRelease [41.5 kB]  
Get:4 https://packages.wazuh.com/4.x/apt stable/main amd64 Contents (deb) [1,957 kB]  
Get:5 http://kali.download/kali kali-rolling/main amd64 Packages [21.3 MB]  
Get:6 http://kali.download/kali kali-rolling/main amd64 Contents (deb) [51.8 MB]  
93% [6 Contents-amd64 50.1 MB/51.8 MB 97%]  
203 kB/s 16s
```

4.2 Running Suricata in IDS Mode

Suricata was started in IDS mode using the active network interface:

```
sudo suricata -i eth0 -v
```

```
valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
   link/ether 08:00:27:6e:13:6e brd ff:ff:ff:ff:ff:ff
   inet 10.104.208.30/24 brd 10.104.208.255 scope global dynamic noprefixroute eth0
       valid_lft 3225sec preferred_lft 3225sec
   inet6 fe80::92dd:c728:695f:fd9/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
   link/ether 08:00:27:ab:f4:24 brd ff:ff:ff:ff:ff:ff

(kali@kali)~$ sudo suricata -i eth0 -v
Notice: Suricata: This is Suricata version 7.0.11 RELEASE running in SYSTEM mode
Info: cpu: CPUs/cores online: 2
Info: suricata: Setting engine mode to IDS mode by default
Info: exception-policy: master exception-policy set to: auto
Info: logopenfile: fast output device (regular) initialized: fast.log
Info: logopenfile: eve-log output device (regular) initialized: eve.json
Info: logopenfile: stats output device (regular) initialized: stats.log
Info: alert-syslog: Syslog output initialized
Warning: detect: No rule files match the pattern /var/lib/suricata/rules/suricata.rules
Warning: detect: 1 rule files specified, but no rules were loaded!
Info: threshold-config: Threshold config parsed: 0 rule(s) found
Info: detect: 0 signatures processed. 0 are IP-only rules, 0 are inspecting packet payload, 0 inspect application layer, 0 are decoder event only
Error: af-packet: fanout not supported by kernel: kernel too old or cluster-id 99 already in use.
Warning: af-packet: eth0: AF_PACKET tpacket-v3 is recommended for non-inline operation
Info: runmodes: eth0: creating 1 thread
Info: unix-manager: unix socket '/var/run/suricata-command.socket'
Info: ioctl: eth0: MTU 1500
Notice: threads: Threads created -> W: 1 FM: 1 FR: 1 Engine started.
```

4.3 Creating Custom Rules

A custom rule was added in /etc/suricata/rules/local.rules:

alert icmp any any -> any any (msg:"ICMP Ping detected";
sid:1000001; rev:1;)

```
GNU nano 8.2 /etc/suricata/rules/local.rules *
alert icmp any any -> any any (msg:"ICMP Ping detected"; sid:1000001; rev:1;)
```

```
GNU nano 8.2 /etc/suricata/suricata.yaml
# The most common hashmode commands are: hash2tuple, hash2tuplesorted,
# hash5tuple, hash5tuplesorted and roundrobin.
#
# See Napatech NTPL documentation other hashmodes and details on their use.
#
# This parameter has no effect if auto-config is disabled.
#
hashmode: hash5tuplesorted

##
## Configure Suricata to load Suricata-Update managed rules.
##
default-rule-path: /etc/suricata/rules
rule-files:
- local.rules

##
## Auxiliary configuration files.
##
classification-file: /etc/suricata/classification.config
reference-config-file: /etc/suricata/reference.config
# threshold-file: /etc/suricata/threshold.config

[ Wrote 2211 lines ]
Help Write Out Where Is Cut Execute Location Undo Set Mark To Bracket
Exit Read File Replace Paste Justify Go To Line Redo Copy Copy Where Was
```

Suricata IDS mode run *Engine started:*

```
zsh: corrupt history file /home/kali/.zsh_history
(kali@kali)~$ sudo suricata -c /etc/suricata/suricata.yaml -i eth0 -v

[sudo] password for kali:
Notice: suricata: This is Suricata version 7.0.11 RELEASE running in SYSTEM mode
Info: cpu: CPUs/cores online: 2
Info: suricata: Setting engine mode to IDS mode by default
Info: exception-policy: master exception-policy set to: auto
Info: logopenfile: fast output device (regular) initialized: fast.log
Info: logopenfile: eve-log output device (regular) initialized: eve.json
Info: logopenfile: stats output device (regular) initialized: stats.log
Info: alert-syslog: Syslog output initialized
Info: detect: 1 rule files processed. 1 rules successfully loaded, 0 rules failed, 0
Info: threshold-config: Threshold config parsed: 0 rule(s) found
Info: detect: 1 signatures processed. 1 are IP-only rules, 0 are inspecting packet payload, 0 inspect application layer, 0 are decoder event only
Error: af-packet: Tunnel not supported by kernel: kernel too old or cluster-id 99 already in use.
Warning: af-packet: eth0: AF_PACKET tpacket-v3 is recommended for non-inline operation
Info: runmodes: eth0: creating 1 thread
Info: unix-manager: unix socket '/var/run/suricata-command.socket'
Info: loctl: eth0: MTU 1500
Notice: threads: Threads created -> W: 1 FM: 1 FR: 1 Engine started.
```

4.4 Testing with Ping

Ping test:

ping -c 4 8.8.8.8

Creating traffic for testing

```
kali@kali:~$ zsh: corrupt history file /home/kali/.zsh_history
(kali@kali)~$ ping -c 4 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data:
64 bytes from 8.8.8.8: icmp_seq=1 ttl=113 time=106 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=113 time=60.7 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=113 time=71.0 ms
64 bytes from 8.8.8.8: icmp_seq=4 ttl=113 time=242 ms

--- 8.8.8.8 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3018ms
rtt min/avg/max/ndev = 60.662/119.988/242.474/72.069 ms

(kali@kali)~$
```

4.5 Checking Alerts in Logs

Suricata logs were checked in fast.log:

```
sudo tail -f /var/log/suricata/fast.log
```

Detected alert:

```
[**] [1:1000001:1] ICMP Ping detected [**]
```

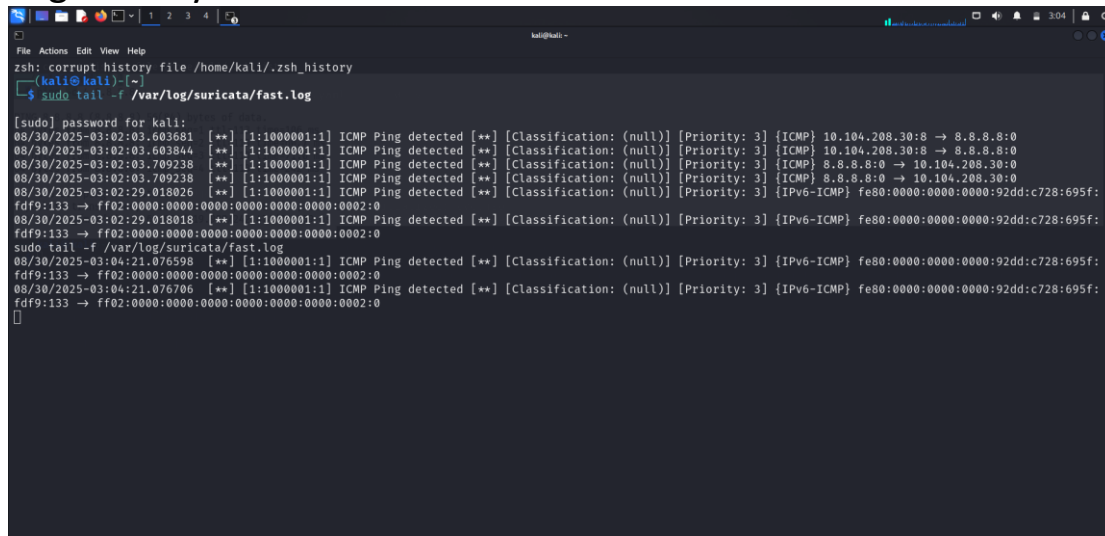
```
kali@kali:~$ sudo tail -f /var/log/suricata/fast.log
[sudo] password for kali:
08/30/2025-03:02:03.603681 [**] [1:1000001:1] ICMP Ping detected [**] [Classification: (null)] [Priority: 3] {ICMP} 10.104.208.30:8 → 8.8.8.8:0
08/30/2025-03:02:03.603844 [**] [1:1000001:1] ICMP Ping detected [**] [Classification: (null)] [Priority: 3] {ICMP} 10.104.208.30:8 → 8.8.8.8:0
08/30/2025-03:02:03.709238 [**] [1:1000001:1] ICMP Ping detected [**] [Classification: (null)] [Priority: 3] {ICMP} 8.8.8.8:0 → 10.104.208.30:0
08/30/2025-03:02:03.709238 [**] [1:1000001:1] ICMP Ping detected [**] [Classification: (null)] [Priority: 3] {ICMP} 8.8.8.8:0 → 10.104.208.30:0
08/30/2025-03:02:29.018026 [**] [1:1000001:1] ICMP Ping detected [**] [Classification: (null)] [Priority: 3] {IPv6-ICMP} fe80:0000:0000:0000:92dd:c728:695f:
fdf9:133 → ff02:0000:0000:0000:0000:0000:0000:0002:0
08/30/2025-03:02:29.018018 [**] [1:1000001:1] ICMP Ping detected [**] [Classification: (null)] [Priority: 3] {IPv6-ICMP} fe80:0000:0000:0000:92dd:c728:695f:
fdf9:133 → ff02:0000:0000:0000:0000:0000:0000:0002:0
sudo tail -f /var/log/suricata/fast.log
08/30/2025-03:04:21.076598 [**] [1:1000001:1] ICMP Ping detected [**] [Classification: (null)] [Priority: 3] {IPv6-ICMP} fe80:0000:0000:0000:92dd:c728:695f:
fdf9:133 → ff02:0000:0000:0000:0000:0000:0000:0002:0
08/30/2025-03:04:21.076706 [**] [1:1000001:1] ICMP Ping detected [**] [Classification: (null)] [Priority: 3] {IPv6-ICMP} fe80:0000:0000:0000:92dd:c728:695f:
fdf9:133 → ff02:0000:0000:0000:0000:0000:0000:0002:0
```

5. Output & Results

Suricata successfully detected **ICMP** pings.

Nmap scans generated intrusion alerts.

Logs clearly showed detection

A screenshot of a Kali Linux terminal window. The terminal shows the user running 'zsh: corrupt history file /home/kali/.zsh_history' and then 'sudo tail -f /var/log/suricata/fast.log'. The output displays several log entries for ICMP ping detections. Each entry includes a timestamp, a source IP, a destination IP, and a classification. The logs show multiple instances of ping requests from 10.104.208.30 to 8.8.8.8 and from 8.8.8.8 to 10.104.208.30. The classification for these events is '[Classification: (null)]'.

```
[sudo] password for kali:
08/30/2025-03:02:03.603681 [**] [1:1000001:1] ICMP Ping detected [**] [Classification: (null)] [Priority: 3] {ICMP} 10.104.208.30:8 → 8.8.8.8:0
08/30/2025-03:02:03.603844 [**] [1:1000001:1] ICMP Ping detected [**] [Classification: (null)] [Priority: 3] {ICMP} 10.104.208.30:8 → 8.8.8.8:0
08/30/2025-03:02:03.709238 [**] [1:1000001:1] ICMP Ping detected [**] [Classification: (null)] [Priority: 3] {ICMP} 8.8.8.8:0 → 10.104.208.30:0
08/30/2025-03:02:03.709238 [**] [1:1000001:1] ICMP Ping detected [**] [Classification: (null)] [Priority: 3] {ICMP} 8.8.8.8:0 → 10.104.208.30:0
08/30/2025-03:02:29.018026 [**] [1:1000001:1] ICMP Ping detected [**] [Classification: (null)] [Priority: 3] {IPv6-ICMP} fe80:0000:0000:0000:92dd:c728:695f:
fdf9:133 → ff02:0000:0000:0000:0000:0000:0000:0002:0
08/30/2025-03:02:29.018018 [**] [1:1000001:1] ICMP Ping detected [**] [Classification: (null)] [Priority: 3] {IPv6-ICMP} fe80:0000:0000:0000:92dd:c728:695f:
fdf9:133 → ff02:0000:0000:0000:0000:0000:0000:0002:0
sudo tail -f /var/log/suricata/fast.log
08/30/2025-03:04:21.076598 [**] [1:1000001:1] ICMP Ping detected [**] [Classification: (null)] [Priority: 3] {IPv6-ICMP} fe80:0000:0000:0000:92dd:c728:695f:
fdf9:133 → ff02:0000:0000:0000:0000:0000:0000:0002:0
08/30/2025-03:04:21.076706 [**] [1:1000001:1] ICMP Ping detected [**] [Classification: (null)] [Priority: 3] {IPv6-ICMP} fe80:0000:0000:0000:92dd:c728:695f:
fdf9:133 → ff02:0000:0000:0000:0000:0000:0000:0002:0
```

6. Advantages of IDS

Detects malicious activity in real-time.

Helps in early warning for network intrusions.

Generates logs useful for forensic investigation.

7. Real-World Applications

Used in **Security Operations Centers (SOCs)**.

Detecting **malware traffic** and **data exfiltration**.

Helps organizations meet **compliance and monitoring standards**.

Forms part of **defense-in-depth strategy**.

8. Conclusion

This project successfully demonstrated setting up and using Suricata as a **Network Intrusion Detection System**.

Custom rules were implemented and tested with real traffic, confirming that Suricata can detect suspicious activities effectively.

This hands-on task improved my understanding of **network security monitoring** and **threat detection systems**.