

CODEALPHA TASK 1

PROJECT TITLE:

**Intrusion Detection and Response using
Suricata IDS on pfSense Firewall**

NAME: Onyinye Nwosu

PROGRAM: Cybersecurity Internship Project

ORGANISATION: CodeAlpha

SUBMISSION DATE: August,2025

Table Of Content

1. Introduction
2. Project Overview
3. Network Setup
4. Tools and Technologies Used
5. Suricata IDS Deployment
6. Interface Configuration & Rule Setup
7. Threat Detection & Alert Generation
8. Incident Response Implementation
9. Challenges Faced & Resolutions
10. Conclusion & Reflection
11. References

1. INTRODUCTION

This task focused on designing, deploying, and testing a Network Intrusion Detection System (NIDS) using Suricata IDS within a virtual lab environment, secured by pfSense firewall/router. The objective was to monitor and detect malicious activities from a simulated attacker (Kali Linux) targeting a small internal network with Ubuntu Desktop and Ubuntu Server machines.

The project mirrors real-world security operations in small enterprise setups, with emphasis on detection, alerting, and response mechanisms. It provides hands-on experience with tools and technologies often used by SOC analysts and network defenders.

2. PROJECT OVERVIEW

The main goal of this project was to deploy a functional intrusion detection and response system by integrating Suricata into a virtualized pfSense firewall, configure it correctly to detect network-based threats from a known attack source, and respond appropriately.

Suricata was chosen for its real-time traffic inspection, rule-based alerting, and open-source flexibility. The project simulates an enterprise LAN setup with virtual machines, where unauthorized access attempts from an attacker system (Kali Linux) must be detected and mitigated.

This task aligns with real-world objectives of a SOC team, including:

1. Detecting malicious or suspicious behavior
2. Alert generation based on configured rules
3. Implementing response actions such as blocking offenders
4. Evaluating the effectiveness of network defense tools

Key Objectives Achieved:

Objective Description

- Network Setup Properly segmented LAN and attacker networks through internal adapters
- IDS Configuration Installed and configured Suricata on pfSense
- Rule Activation Enabled ET rules and custom rule categories
- Alert Validation Successfully triggered alerts by launching scans/attacks from Kali
- Incident Response Implemented blocking and inline IPS actions through Suricata

Network Threat Simulated:**Tool Used for Attack:** Kali Linux**Attack Methods:** Ping probe, Nmap scan**Targeted VMs:** Ubuntu Desktop and Ubuntu Server**Response:** Alert generation + IPS mode drop actions**Project Outcome:**

The final setup allowed us to detect and respond to unauthorized network scans from the attacker VM, proving that Suricata was effectively inspecting traffic, logging alerts, and taking action on defined rules.

3. NETWORK SETUP & VIRTUAL INFRASTRUCTURE

Network Topology & IP Addressing

The virtual network was designed using VirtualBox, and consisted of 4 main virtual machines:

VM	Role	Network Adapter	IP Address
pfSense	Firewall Router + IDS/IPS	Adapter 1: NAT (WAN) Adapter 2: Internal Network(intnet) Adapter 3: Internal Network (KaliAttacker)	WAN: 10.0.2.15/24 (DHCP from NAT) LAN: 192.168.10.1/24 OPT1 (for Kali): 192.168.20.1/24
Ubuntu Desktop	LAN Workstation	Internal Network (intnet)	192.168.10.51/24
Ubuntu Server	LAN Server	Internal Network (intnet)	192.168.10.56/24
Kali Linux	Attacker Machine	Internal Network (Kali Attacker)	192.168.20.52/24

4. VIRTUALBOX VM NETWORK ADAPTER CONFIGURATION

VM Adapter Network Mode Assigned To

pfSense:

- Adapter 1 NAT Internet (default)
- Adapter 2 Internal Network: intnet - LAN VMs (Ubuntu Desktop & Server)
- Adapter 3 Internal Network: Kali Attacker- Kali Linux

Ubuntu Desktop/Server:

Adapter 1 Internal Network (intnet), To communicate with pfSense

Kali Linux:

Adapter 1 Internal Network: Kali Attacker, To attack through pfSense

IP Address Configuration

pfSense Interfaces

WAN: DHCP (gets IP like 10.0.2.15)

LAN: Static IP **192.168.10.1/24**

OPT1 (Renamed to KaliAttacker): Static IP **192.168.20.1/24**

LAN Devices:

Ubuntu Desktop: DHCP from pfSense LAN: **192.168.10.50**

Ubuntu Server: DHCP: **192.168.10.51**

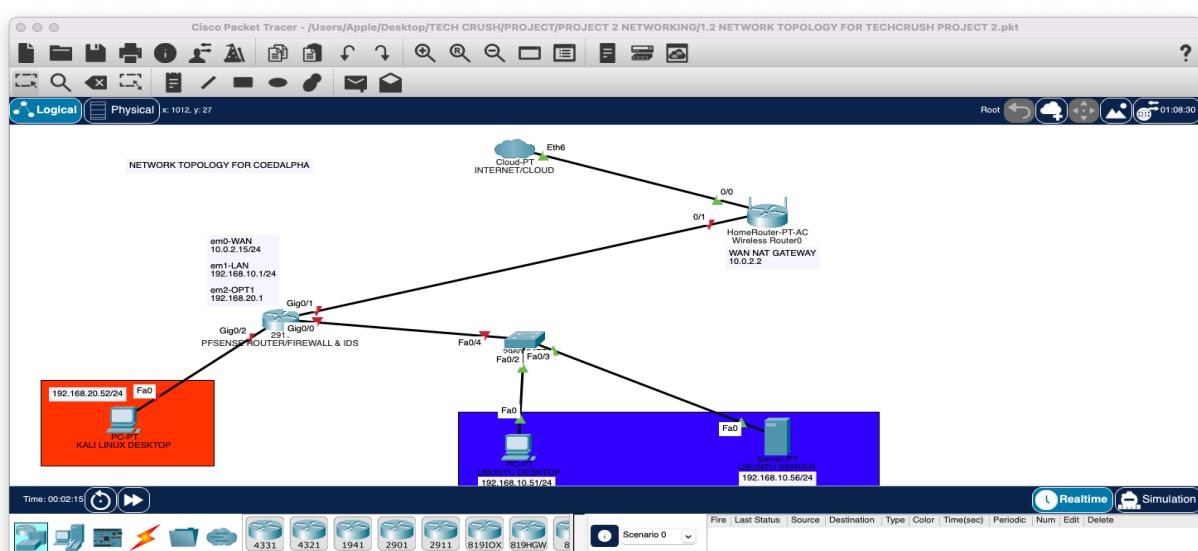
Attacker Device (Kali Linux)

Static IP: **192.168.20.52/24**

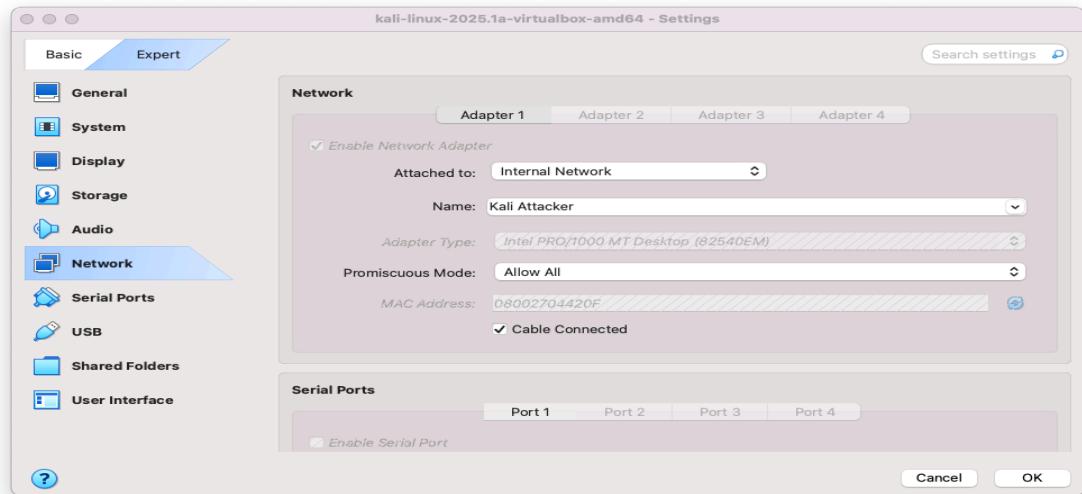
Gateway: **192.168.20.1**

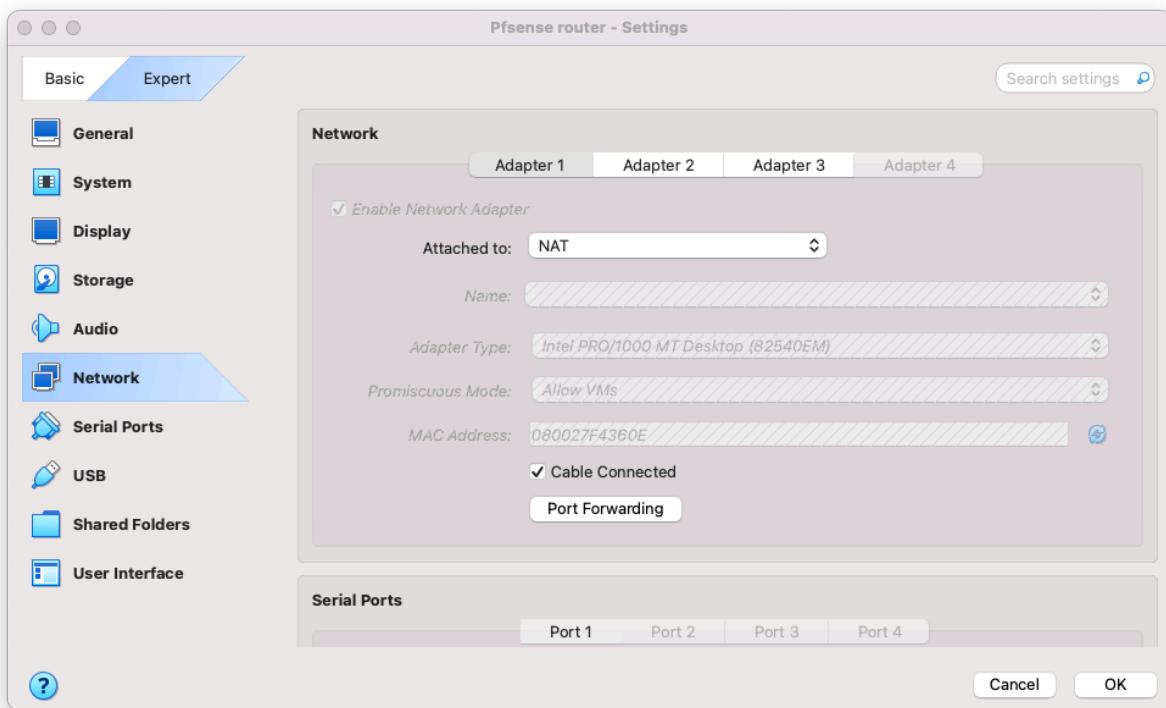
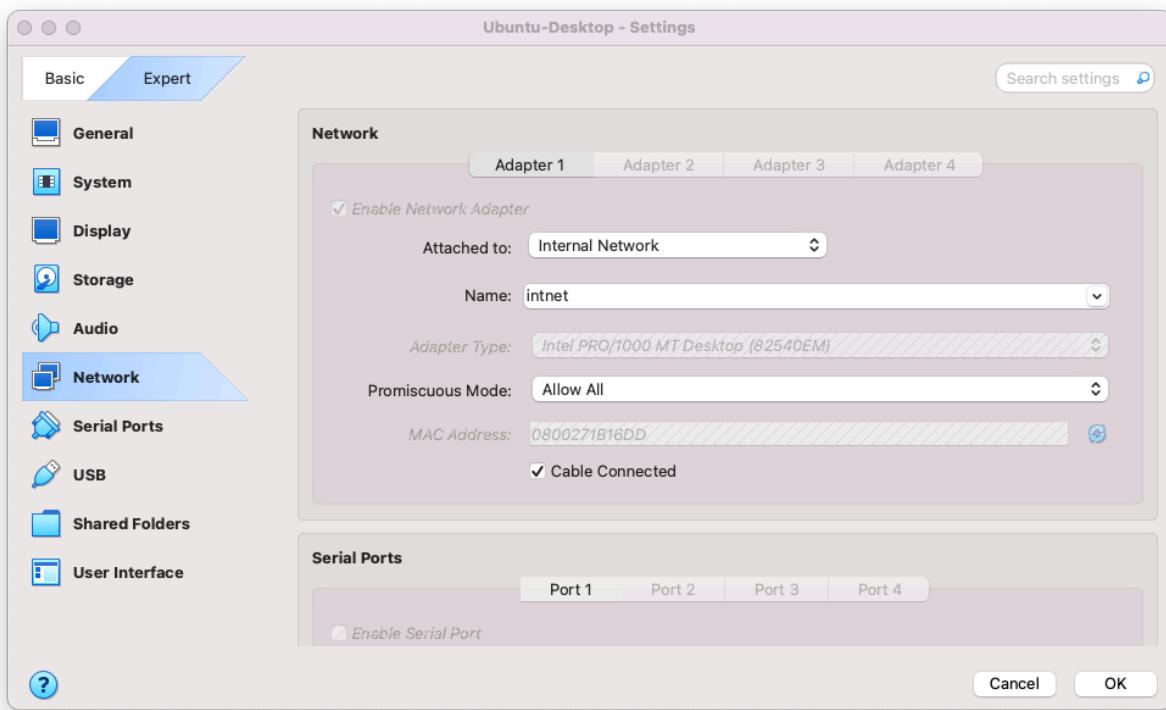
DNS: **8.8.8.8** or use pfSense gateway

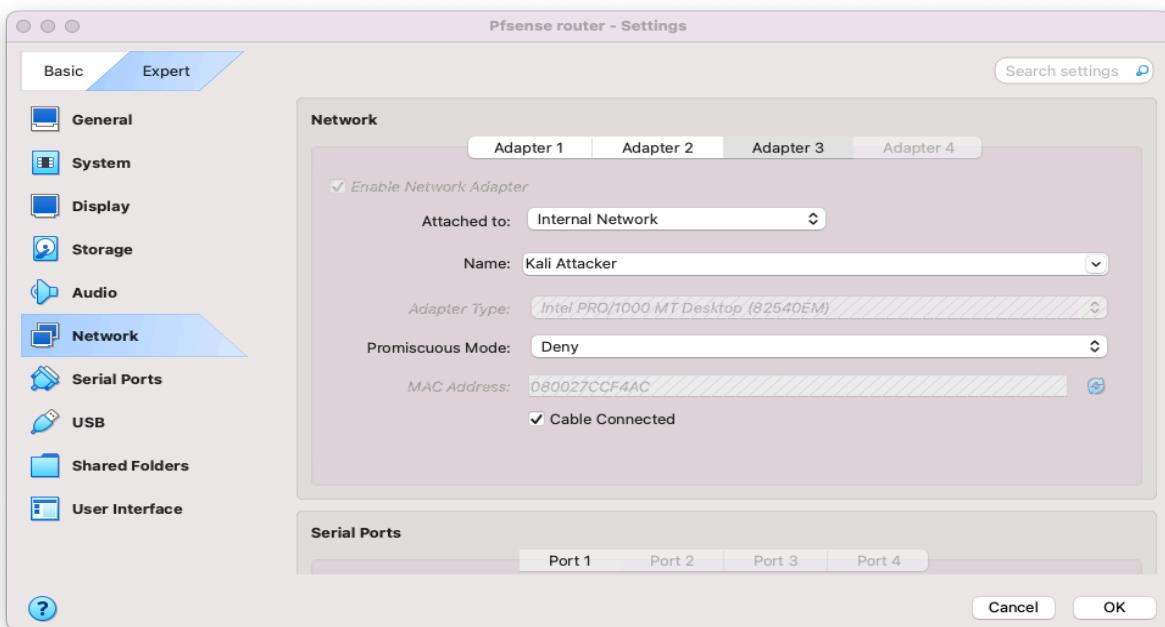
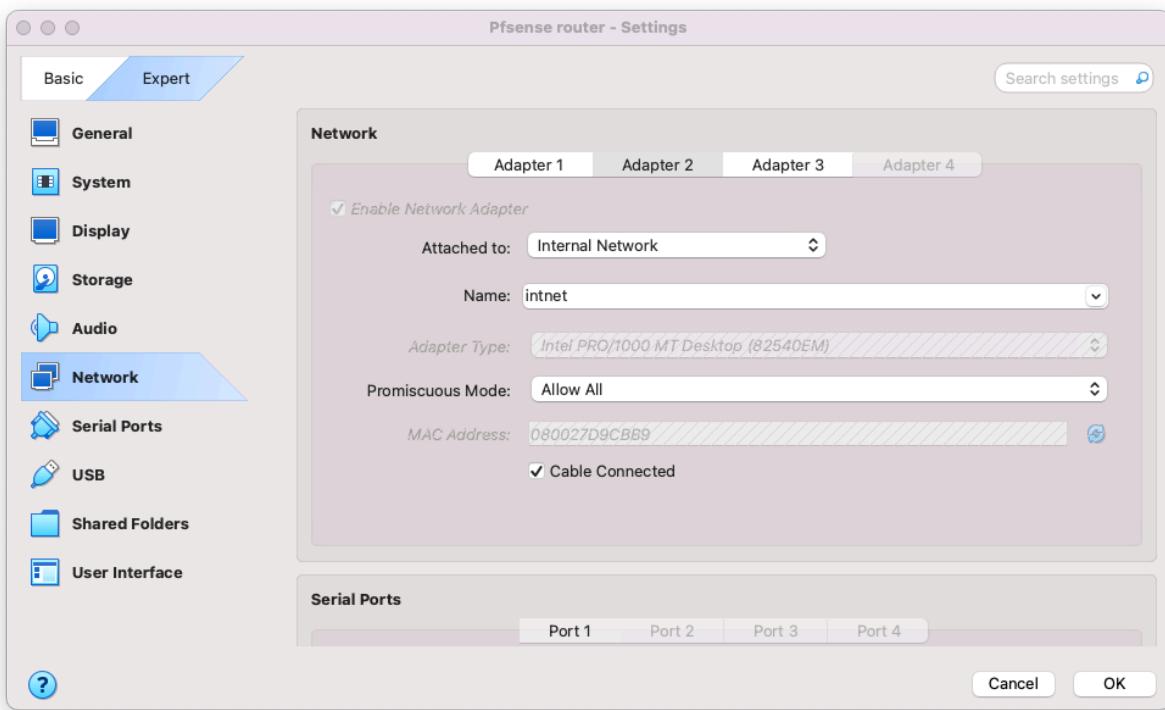
Network Topology



VM Settings showing network adapter types

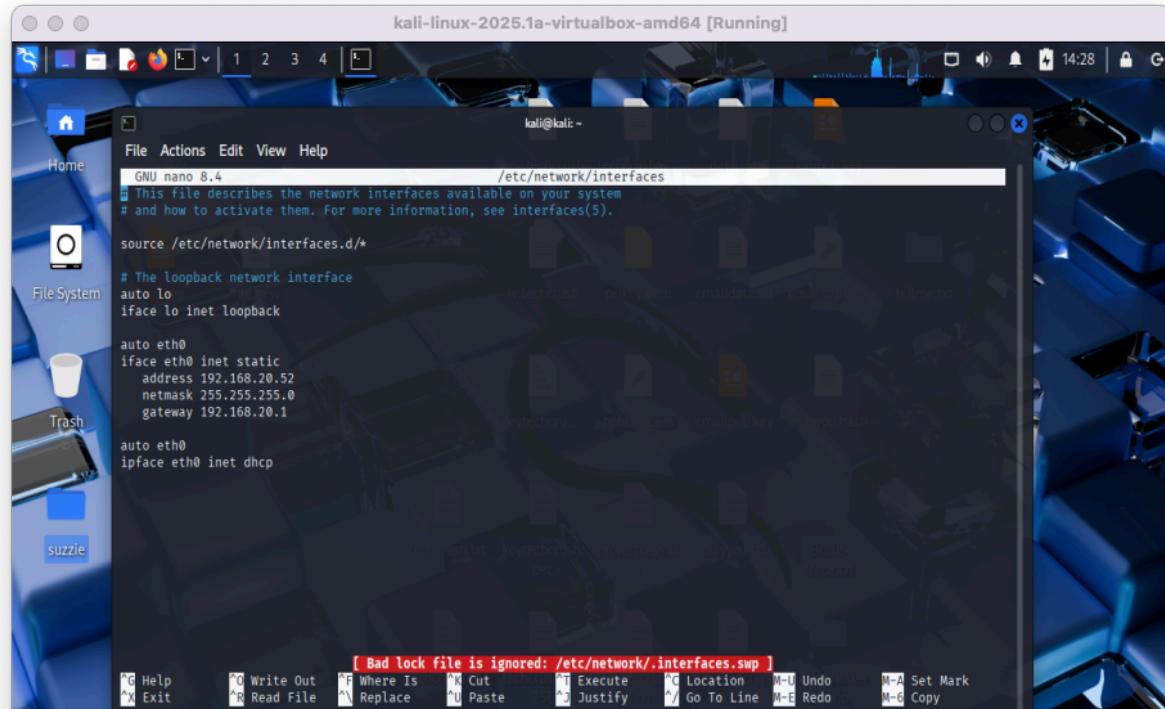






Network Manager & ip a output on each VM

Kali VM:



The screenshot shows a Kali Linux desktop environment. A terminal window titled "kali-linux-2025.1a-virtualbox-amd64 [Running]" is open, displaying the contents of the `/etc/network/interfaces` file. The file configuration includes loopback, eth0 (static IP), and eth0 (DHCP). The terminal window has a dark blue background and a standard Linux-style menu bar.

```
GNU nano 8.4 /etc/network/interfaces
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

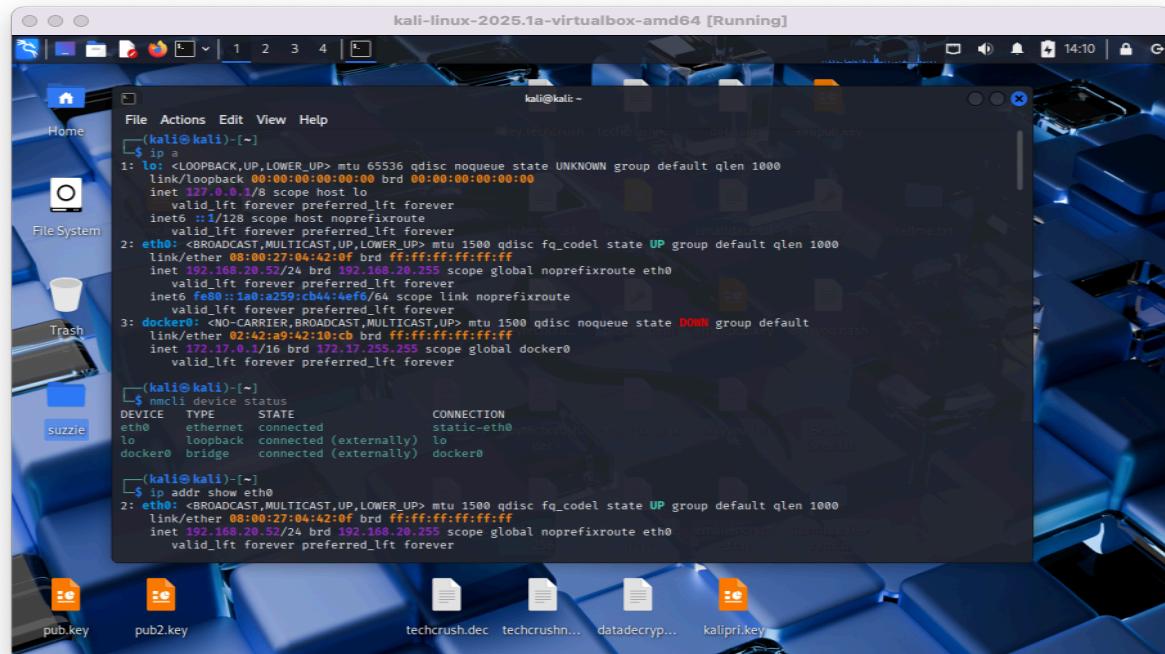
source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

auto eth0
iface eth0 inet static
    address 192.168.20.52
    netmask 255.255.255.0
    gateway 192.168.20.1

auto eth0
iface eth0 inet dhcp

[ Bad lock file is ignored: /etc/network/.interfaces.swp ]
```



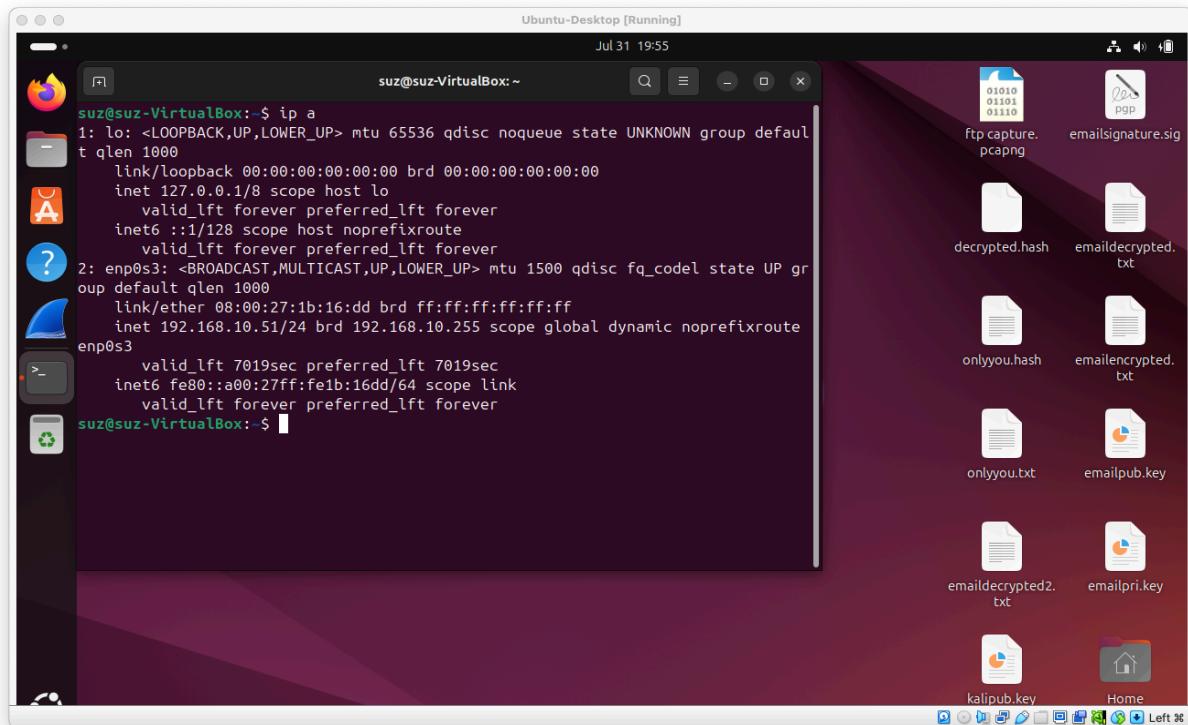
The screenshot shows a Kali Linux desktop environment. A terminal window titled "kali-linux-2025.1a-virtualbox-amd64 [Running]" is open, displaying the output of the `ip a` command. The output lists network interfaces (lo, eth0, docker0) with their respective link layer, MTU, queueing discipline (qdisc), state, broadcast domain, and link layer information. Below this, the `nmcli device status` command is run, showing the connection status for each interface. The terminal window has a dark blue background and a standard Linux-style menu bar.

```
(kali㉿kali)-[~]
$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
        inet 127.0.0.1/8 scope host lo
            valid_lft forever preferred_lft forever
        inet6 ::1/128 scope host noprefixroute
            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 08:00:27:04:42:0f brd ff:ff:ff:ff:ff:ff
        inet 192.168.20.52/24 brd 192.168.20.255 scope global noprefixroute eth0
            valid_lft forever preferred_lft forever
        inet6 fe80::a00:27ff:fe04:42%eth0 brd ff:ff:ff:ff:ff:ff scope link noprefixroute
            valid_lft forever preferred_lft forever
3: docker0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue state DOWN group default
    link/ether 02:42:a9:42:10:cb brd ff:ff:ff:ff:ff:ff
        inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
            valid_lft forever preferred_lft forever

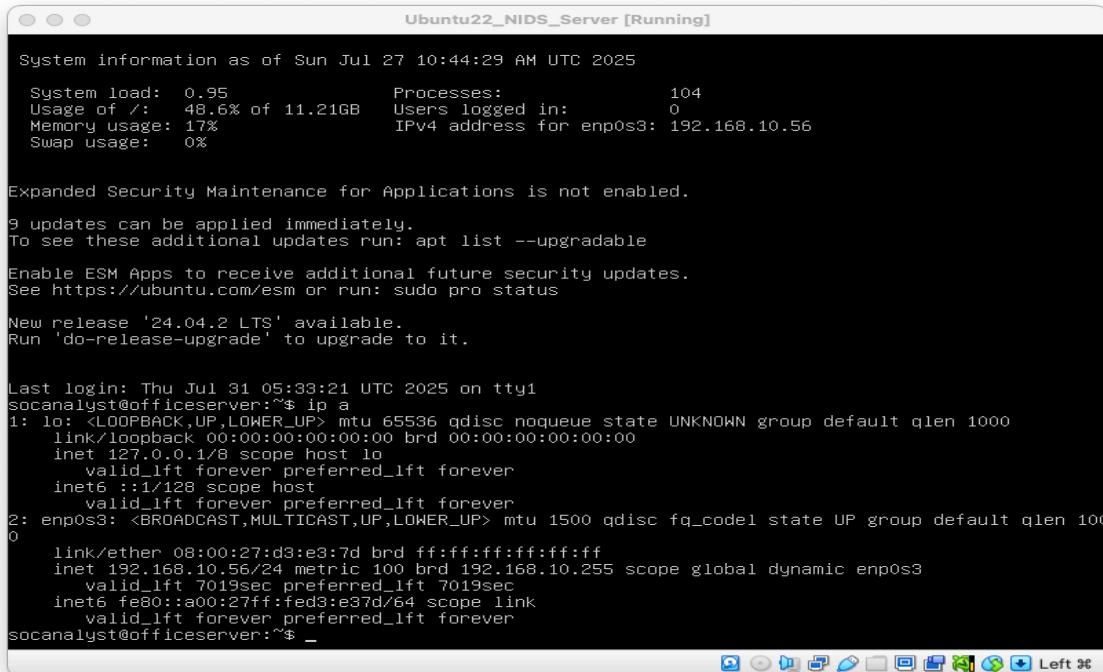
(kali㉿kali)-[~]
$ nmcli device status
DEVICE      TYPE      STATE           CONNECTION
eth0        ethernet  connected       static-eth0
lo          loopback  connected (externally) to docker0
docker0     bridge    connected (externally) to docker0

(kali㉿kali)-[~]
$ ip addr show eth0
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:04:42:0f brd ff:ff:ff:ff:ff:ff
        inet 192.168.20.52/24 brd 192.168.20.255 scope global noprefixroute eth0
            valid_lft forever preferred_lft forever
```

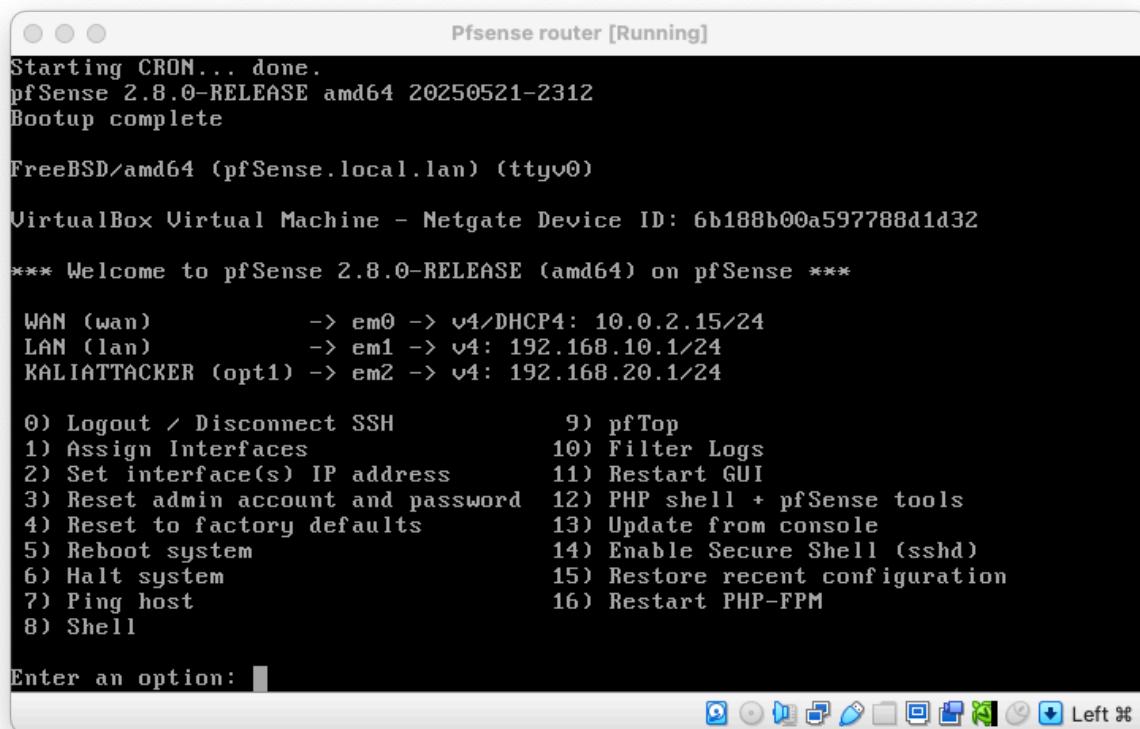
Ubuntu Desktop:



Ubuntu Server



Interfaces page in pfSense showing WAN, LAN, and OPT1



Pfsense router [Running]

```
Starting CRON... done.
pfSense 2.8.0-RELEASE amd64 20250521-2312
Bootup complete

FreeBSD/amd64 (pfSense.local.lan) (ttyv0)

VirtualBox Virtual Machine - Netgate Device ID: 6b188b00a597788d1d32

*** Welcome to pfSense 2.8.0-RELEASE (amd64) on pfSense ***

WAN (wan)      -> em0 -> v4/DHCP4: 10.0.2.15/24
LAN (lan)      -> em1 -> v4: 192.168.10.1/24
KALIATTACKER (opt1) -> em2 -> v4: 192.168.20.1/24

0) Logout / Disconnect SSH          9) pfTop
1) Assign Interfaces                10) Filter Logs
2) Set interface(s) IP address     11) Restart GUI
3) Reset admin account and password 12) PHP shell + pfSense tools
4) Reset to factory defaults       13) Update from console
5) Reboot system                   14) Enable Secure Shell (sshd)
6) Halt system                     15) Restore recent configuration
7) Ping host                       16) Restart PHP-FPM
8) Shell

Enter an option: ■
```

The screenshot shows the pfSense terminal interface. It displays boot logs, network interface configurations, and a command-line menu with 16 options. The menu includes options for logging out, assigning interfaces, setting IP addresses, rebooting, and updating the system. At the bottom, there is a text input field labeled "Enter an option:" followed by a small selection bar. Below the input field is a toolbar with various icons.

5: IDS/IPS CONFIGURATION ON PFSENSE WITH SURICATA

Installation of Suricata on pfSense

Tool Used: Suricata (via pfSense Package Manager)

Steps:

1. Login to pfSense Web UI via browser using <https://192.168.10.1> from any LAN VM.
2. Navigate to System > Package Manager > Available Packages.
3. Search for Suricata and click Install.
4. After installation, go to Services > Suricata.

6. SURICATA INTERFACE CONFIGURATION

I created two separate interfaces for inspection:

Interface Purpose IP/Subnet Monitored by Suricata?

LAN Monitors internal LAN 192.168.10.1/24

KaliAttacker (OPT1) Monitors attacker subnet 192.168.20.1/24

Steps:

1. Navigate to Services > Suricata > Interfaces.

2. Click + Add to configure each interface:

Interface: LAN and KALIATTACKER

Description: Suricata_LAN and Suricata_Kali_Attacker

Enable Interface:

IPS Mode: (to allow active blocking)

Promiscuous Mode:

Block Offenders: (Important!)

Enabled loggings

Choose Home Net: default or define manually as 192.168.10.0/24,192.168.20.0/24

3. Scroll down to Logging Settings and enable logging for all relevant alert types.

4. Save and apply changes.

Rules Setup For Detection

I used Emerging Threats (ET Open) rules:

Steps:

1. Go to Services > Suricata > Global Settings.

2. In Rules Update Settings:

Enable ET Open Rules:

Update Frequency: 12 hours (or as needed)

3. Save and then click Update Rules button.

4. Navigate back to Interfaces > LAN > Categories.

5. Select categories like:

- Emerging-attacks
- Emerging-malware
- Emerging-scans
- Emerging-shellcode

6. Use Enable All button for the selected category.

Rule Management

Rule Category Activation:

- Under the Rules tab for each interface, categories such as Active Rules and Attack Response were enabled.
- Categories were manually reviewed to activate relevant rules against port scans, brute-force attempts, and suspicious payloads.

7. ATTACK SIMULATION, THREAT DETECTION AND ALERT GENERATION

This section outlines the simulation of a malicious scan from an untrusted host, the detection of this activity using Suricata IDS on pfSense, and the security response implemented via firewall rules. Here, I simulated unauthorized access attempts from an

attacker VM (Kali Linux) to the internal network, trigger detection using Suricata IDS, and respond by blocking the attacker via pfSense firewall rules.

Attack Simulation from Kali VM

➤ Tools Used:

Kali Linux VM (192.168.20.52)

nmap utility for reconnaissance

➤ Commands Executed:

- ping 192.168.10.1
- nmap -A -sS 192.168.10.1

ping tested host reachability.

nmap -A -sS performed a stealth scan with OS detection, version detection, script scanning, and traceroute.

➤ Goal:

Simulate an attacker probing a critical device — the pfSense gateway at 192.168.10.1.

Detection via Suricata on pfSense

- Suricata was active on both the LAN and Kali Interface.
- Alerts were generated for suspicious scanning behavior, visible in the pfSense > Suricata > Alerts section.

Sample alert:

[1:2024364] ET SCAN Nmap User-Agent Observed

Alerts confirmed successful detection of the scan from Kali.

8. INCIDENT RESPONSE IMPLEMENTATION

SID Management (Selective Blocking)

Due to UI limitations under the KaliAttacker interface (where SID Management was unavailable), we enabled drop actions manually using the central SID Mgmt tab.

Steps:

1. Go to Services > Suricata > SID Mgmt.
2. Click Add SID.
3. Add the SIDs (rule IDs) of signatures you want to enforce drop action on.
4. Apply changes using Apply SID Mgmt.
5. Confirm under interface settings that the rules are now enforcing drop.

SID Management Attempts & Outcome:

While Suricata's Global SID Management tool was used to attempt setting rules to drop, the expected behavior was not consistently achieved. This was due to the absence of the SID Management tab under the Kali interface (DMZ), which limited per-interface rule

control. Used the Global SID Mgmt section to add specific SIDs to be set to drop. However, these actions did not consistently result in alerts or active blocking, possibly due to interface mismatch or Suricata behavior with pfSense's packet flow handling.

SID Management Limitations:

- No SID Management section appeared under the Kali interface in Suricata. Drop rules were added manually through Global SID Management, but this proved unreliable.
- Alerts were seen occasionally (e.g., QUIC decrypt fail), but no consistent block behavior was observed from SID actions alone.

Final Implementation That Achieved Blocking

To fulfill the detection and response objective of this project, a firewall-based mitigation strategy was adopted.

Effective Countermeasure:

A pfSense Firewall Rule was created to block all traffic originating from the attacker network.

Interface: Kali Interface (192.168.20.1/24)

Action: Block

Source: 192.168.20.0/24

Destination: 192.168.10.0/24

Protocol: Any

This rule successfully blocked Kali VM from accessing both Ubuntu Desktop and Server, achieving containment and demonstrating a security response to an identified threat.

Validation of Block

- Repeated ping and nmap scans from Kali to 192.168.10.1 failed.
- No more Suricata alerts were generated for Kali after the block.
- Attacker was effectively isolated from the internal network.

1. Suricata package installed on pfSense

The screenshot shows the pfSense Package Manager interface. The 'Installed Packages' tab is selected. A single package, 'suricata', is listed in the table. The table columns include Name, Category, Version, Description, and Actions. The 'suricata' entry has a checkmark in the Name column and is categorized under 'security'. The version is 7.0.8_2. The description states: 'High Performance Network IDS, IPS and Security Monitoring engine by OISF.' Below the table, there is a note: 'Package Dependencies: suncata-7.0.8'. At the bottom, there are icons for Update, Current, Remove, Information, Reinstall, and a note about a newer version available.

2. Interfaces where Suricata is enabled

The screenshot shows the pfSense Services / Suricata interface. The 'Interfaces' tab is selected. Two interfaces are listed in the table: 'KALIATTACKER (em2)' and 'LAN (em1)'. The 'Suricata Status' column shows green checkmarks and circular icons. The 'Pattern Match' column shows 'AUTO' for both. The 'Blocking Mode' column shows 'INLINE IPS' for both. The 'Description' column provides specific names for each interface: 'Suricata_Kali_Attacker' for em2 and 'Suricata_LAN' for em1. There are 'Edit' and 'Delete' buttons at the bottom right of the table.

4. Rules categories and tab showing selected categories (e.g., attack-response.rules)

The screenshot shows the pfSense Services / Suricata / Global Settings interface. The 'Global Settings' tab is selected. The page displays a section titled 'Please Choose The Type Of Rules You Wish To Download'. It lists four categories: 'Install ETOpen Emerging Threats rules', 'Install ETPro Emerging Threats rules', 'Install Snort rules', and 'Install Snort GPLv2'. Each category has a checkbox for selecting the rule set and a checkbox for using a custom URL. There are also links for 'Sign Up for a free Registered User Rules Account' and 'Sign Up for paid Snort Subscriber Rule Set (by Talos)'.

5. Force rule update

The screenshot shows the pfSense web interface with the URL `192.168.10.1/suricata/suricata_download_updates.php`. The page title is "Services / Suricata / Updates". The "Updates" tab is selected in the navigation bar. Below the tabs are two buttons: "Sync" and "IP Lists". The main content area is titled "INSTALLED RULE SET MD5 SIGNATURES" and displays a table of installed rule sets:

Rule Set Name/Publisher	MD5 Signature Hash	MD5 Signature Date
Emerging Threats Open Rules	bccc1fc385948d31f77e4a8d7d94261d	Saturday, 26-Jul-25 13:32:39 WAT
Snort Subscriber Rules	Not Enabled	Not Enabled
Snort GPLv2 Community Rules	Not Enabled	Not Enabled
Feodo Tracker Botnet C2 IP Rules	Not Enabled	Not Enabled
ABUSE.ch SSL Blacklist Rules	Not Enabled	Not Enabled

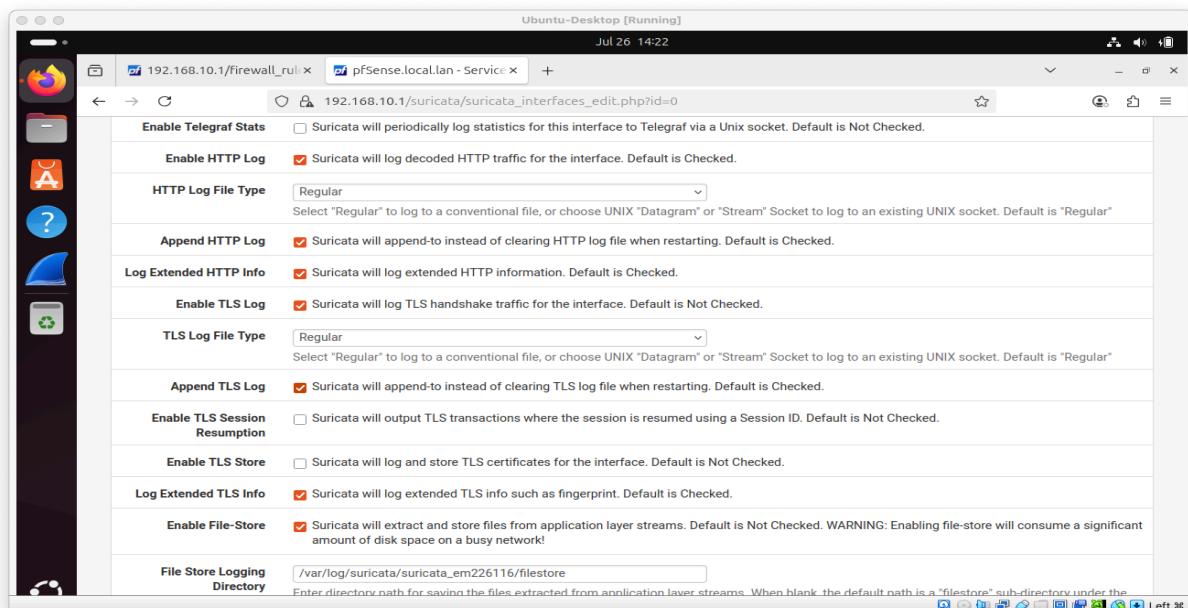
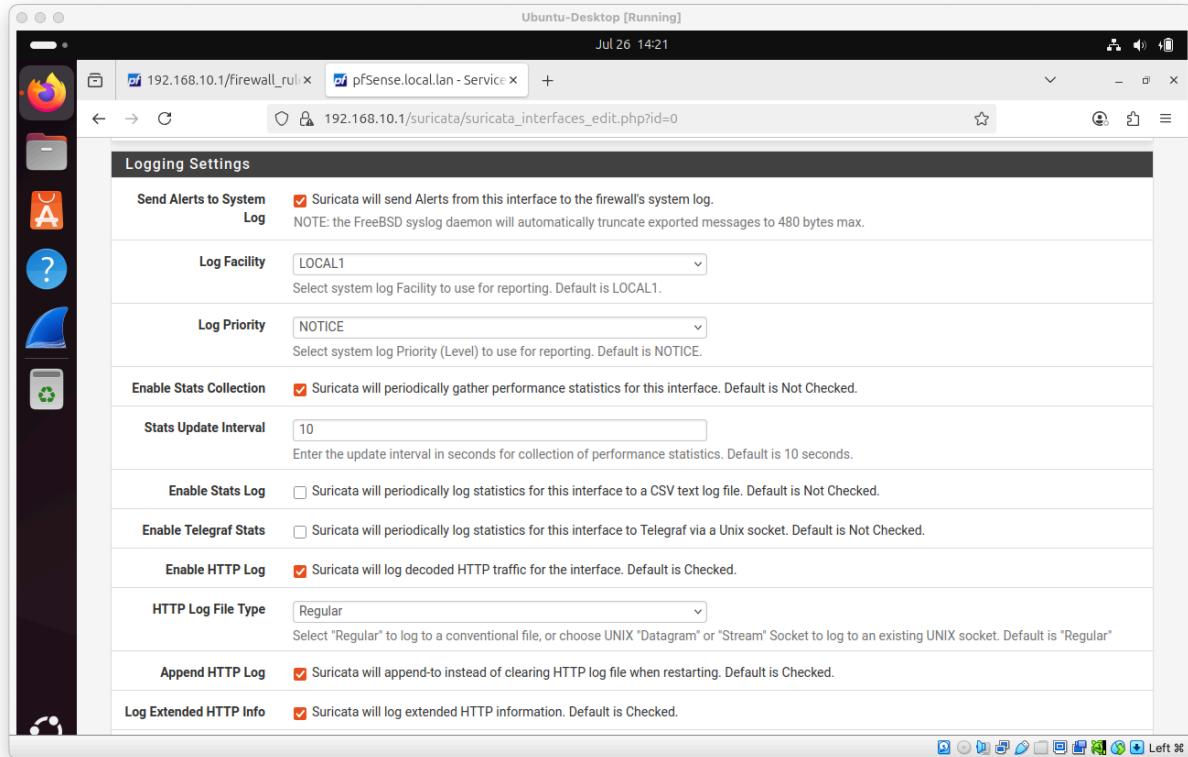
Below this is a section titled "UPDATE YOUR RULE SET" with the message "Last Update: Jul-26 2025 13:32" and "Result: success". At the bottom are two buttons: "Update" and "Force".

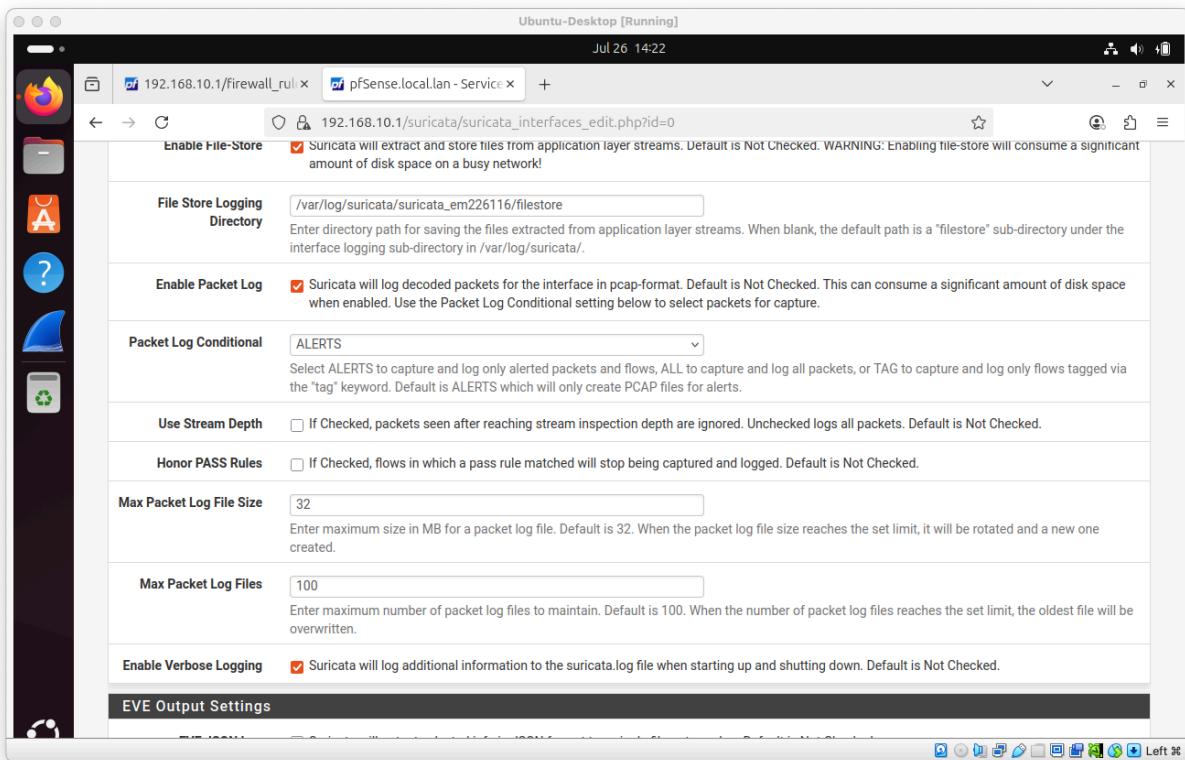
6. Enable ET rules

The screenshot shows the pfSense web interface with the URL `192.168.10.1/suricata/suricata_rulesets.php`. The page lists various rule sets with checkboxes next to them. Some checkboxes are checked, indicating they are enabled. The list includes:

- emerging-ja3.rules
- emerging-malware.rules
- emerging-misc.rules
- emerging-mobile_malware.rules
- emerging-netbios.rules
- emerging-p2p.rules
- emerging-phishing.rules
- emerging-pop3.rules
- emerging-remote_access.rules
- emerging-retired.rules
- emerging-rpc.rules
- emerging-scada.rules
- emerging-scan.rules
- emerging-shellcode.rules
- emerging-smtp.rules
- emerging-snmp.rules
- emerging-sql.rules
- emerging-ta_abused_services.rules
- emerging-telnet.rules

6B. Enabled loggings and alerts





7. Terminal output showing ping and nmap -A -sS 192.168.10.1 from Kali before rule

The screenshot shows a terminal window on a Kali Linux desktop. The terminal title is 'kali-linux-2025.1a-virtualbox-amd64 [Running]'. The command run is 'sudo nmap -A -sS 192.168.10.0/24'. The output shows a scan of the 192.168.10.0/24 network, identifying several hosts and services, including a pfSense device at 192.168.10.1.

```
(kali㉿kali)-[~]
$ sudo nmap -A -sS 192.168.10.0/24
[sudo] password for kali:
Starting Nmap 7.95 ( https://nmap.org ) at 2025-07-26 09:33 EDT
Nmap scan report for 192.168.10.1
Host is up (0.0085s latency).
Not shown: 997 filtered tcp ports (no-response)
port      STATE SERVICE VERSION
80/tcp    open  http    nginx
|_http-title: Did not follow redirect to https://192.168.10.1/
443/tcp   open  ssl/http nginx
|_tls-alpn:
|   h2
|   http/1.1
|   http/1.0
|   http/0.9
|_ssl-date: TLS randomness does not represent time
| ssl-cert: Subject: commonName=pfSense-683845607dc0c/organizationName=pfSense GUI default Self-Signed Certificate
|   Subject Alternative Name: DNS:pfSense-683845607dc0c
|   Not valid before: 2025-05-29T11:30:41
|   Not valid after:  2026-05-29T11:30:41
|_http-title: pfSense - Login
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Device type: general purpose
Running (JUST GUESSING): FreeBSD 11.X (97%)
OS CPE: cpe:/o:freebsd:freebsd:11.2
Aggressive OS guesses: FreeBSD 11.2-RELEASE (97%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 1 hop
```

```

kali@kali:~$ ping 192.168.10.1
PING 192.168.10.1 (192.168.10.1) 56(84) bytes of data.
64 bytes from 192.168.10.1: icmp_seq=1 ttl=64 time=4.15 ms
64 bytes from 192.168.10.1: icmp_seq=2 ttl=64 time=2.66 ms
^C
--- 192.168.10.1 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1005ms
rtt min/avg/max/mdev = 2.662/3.405/4.148/0.743 ms

```

8. Suricata alert log showing triggered detection

Alert Log View Settings

Instance to View: (KALIATTACKER) Suricata_Kali_Attacker

Save or Remove Logs: Download, Clear

Save Settings: Save, Refresh (Default is ON), Number of alerts to display: 250

Alert Log View Filter

Last 250 Alert Entries. (Most recent entries are listed first)

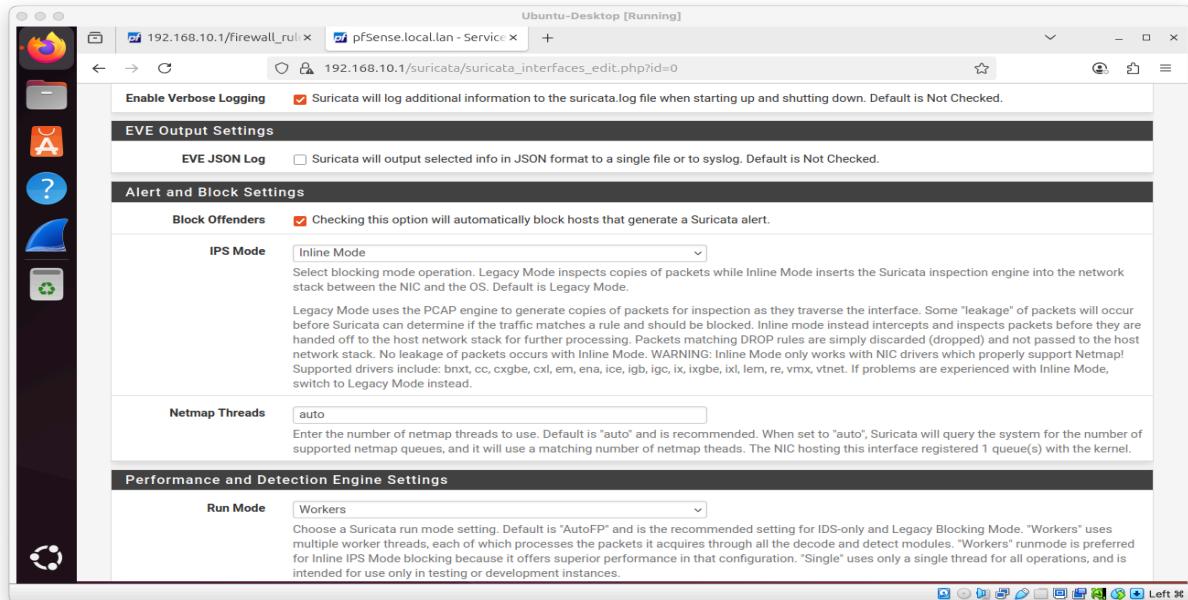
Date	Action	Pri	Proto	Class	Src	SPort	Dst	DPort	GID:SID	Description
07/26/2025 14:34:13	▲	1	TCP	Web Application Attack	192.168.20.52	60388	192.168.10.1	80	1:2024364	ET SCAN Possible Nmap User-Agent Observed
07/26/2025 14:34:13	▲	1	TCP	Web Application Attack	192.168.20.52	60386	192.168.10.1	80	1:2024364	ET SCAN Possible Nmap User-Agent Observed
07/26/2025 14:34:13	▲	1	TCP	Web Application Attack	192.168.20.52	60370	192.168.10.1	80	1:2024364	ET SCAN Possible Nmap User-Agent Observed

Ubuntu-Desktop [Running] Jul 26 14:40

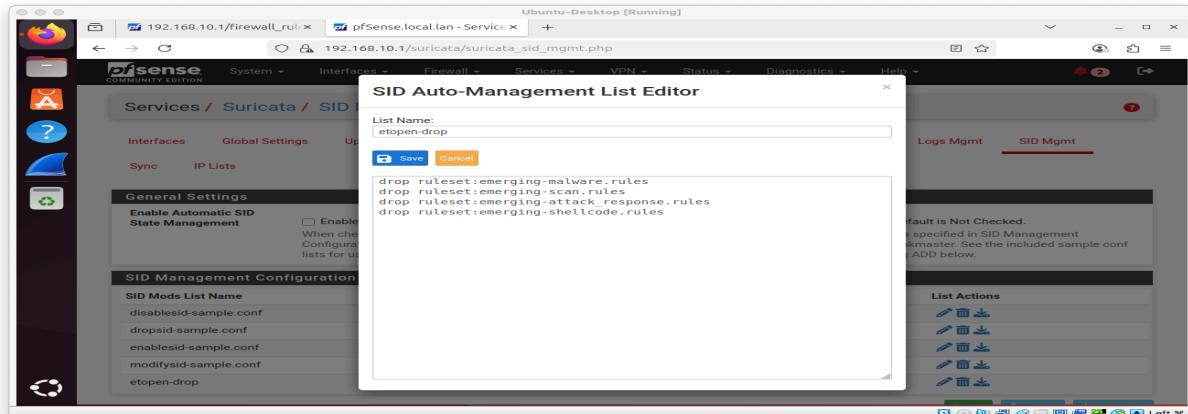
192.168.10.1/Firewall_rules - Service - 192.168.10.1/suricata/suricata_alerts.php?instance=0

Date	Action	Pri	Proto	Class	Src	SPort	Dst	DPort	GID:SID	Description
07/26/2025 14:34:13	▲	1	TCP	Web Application Attack	192.168.20.52	60358	192.168.10.1	80	1:2024364	ET SCAN Possible Nmap User-Agent Observed
07/26/2025 14:34:13	▲	1	TCP	Web Application Attack	192.168.20.52	60350	192.168.10.1	80	1:2024364	ET SCAN Possible Nmap User-Agent Observed
07/26/2025 14:34:12	▲	1	TCP	Web Application Attack	192.168.20.52	60336	192.168.10.1	80	1:2024364	ET SCAN Possible Nmap User-Agent Observed
07/26/2025 14:34:12	▲	1	TCP	Web Application Attack	192.168.20.52	60334	192.168.10.1	80	1:2024364	ET SCAN Possible Nmap User-Agent Observed
07/26/2025 14:34:12	▲	1	TCP	Web Application Attack	192.168.20.52	60314	192.168.10.1	80	1:2024364	ET SCAN Possible Nmap User-Agent Observed
07/26/2025 14:34:12	▲	1	TCP	Web Application Attack	192.168.20.52	60264	192.168.10.1	80	1:2024364	ET SCAN Possible Nmap User-Agent Observed
07/26/2025 14:34:12	▲	1	TCP	Web Application Attack	192.168.20.52	60320	192.168.10.1	80	1:2024364	ET SCAN Possible Nmap User-Agent Observed
07/26/2025 14:34:12	▲	1	TCP	Web Application Attack	192.168.20.52	60300	192.168.10.1	80	1:2024364	ET SCAN Possible Nmap User-Agent Observed
07/26/2025 14:34:12	▲	1	TCP	Web Application Attack	192.168.20.52	60286	192.168.10.1	80	1:2024364	ET SCAN Possible Nmap User-Agent Observed
07/26/2025 14:34:12	▲	1	TCP	Web Application Attack	192.168.20.52	60278	192.168.10.1	80	1:2024364	ET SCAN Possible Nmap User-Agent Observed
07/26/2025 14:34:12	▲	3	TCP	Generic Protocol Command Decode	192.168.20.52	60278	192.168.10.1	80	1:2260002	SURICATA Applayer Detect protocol only one direction
07/26/2025 14:34:12	▲	1	TCP	Web Application Attack	192.168.20.52	60258	192.168.10.1	80	1:2024364	ET SCAN Possible Nmap User-Agent Observed

9. IPS mode and Block Offenders checkbox activated



10. Global SID Management area with manually added SIDs and drop rules



The screenshot shows the pfSense web interface for managing Suricata SID Management Configuration Lists. The URL is 192.168.10.1/suricata/suricata_sid_mgmt.php. The page displays a table of existing configuration lists and allows for adding, importing, or downloading new ones. It also includes an interface SID management list assignments section.

SID Mods List Name	Last Modified Time	List Actions
disableSID-sample.conf	May-22 2025 2:18 am	[Edit] [Delete]
dropSID-sample.conf	May-22 2025 2:18 am	[Edit] [Delete]
enableSID-sample.conf	May-22 2025 2:18 am	[Edit] [Delete]
modifySID-sample.conf	May-22 2025 2:18 am	[Edit] [Delete]
etopen-drop	Jul-28 2025 10:59 am	[Edit] [Delete]

Interface SID Management List Assignments

Rebuild	Interface	SID State Order	Enable SID List	Disable SID List	Modify SID List	Drop SID List	Reject SID List
<input type="checkbox"/>	KALIATTACKER	Disable, Enab	None	None	None	etopen-drop	None

Save Remember to save changes before exiting this page

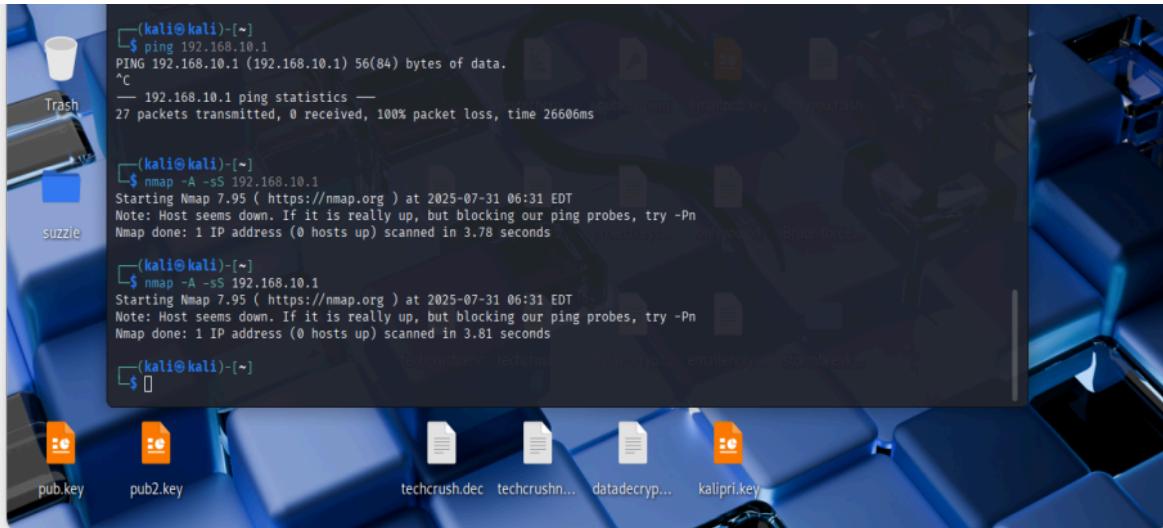
11. pfSense Firewall Rule blocking Kali traffic

The screenshot shows the pfSense web interface for managing Firewall Rules. The URL is 192.168.10.1/firewall_rules.php?if=opt1. A success message indicates that changes have been applied successfully and the firewall rules are reloading. The KALIATTACKER tab is selected, showing two rules: one to block Kali from LAN and another to allow Kali traffic.

The changes have been applied successfully. The firewall rules are now reloading in the background.
Monitor the filter reload progress.

States	Protocol	Source	Port	Destination	Port	Gateway	Queue	Schedule	Description	Actions
0/0 B	IPv4 *	192.168.20.52	*	192.168.10.0/24	*	*	none		Block kali from LAN	[Edit] [Copy] [Delete]
0/1.37 MiB	IPv4 *	KALIATTACKER subnets	*	*	*	*	none		Allow kali Attacker	[Edit] [Copy] [Delete]

12. Failed re-attempted scan or ping from Kali



The screenshot shows a terminal window on a Kali Linux desktop environment. The terminal displays three command-line sessions:

- The first session shows a failed ping attempt to 192.168.10.1, resulting in 100% packet loss.
- The second session shows a failed nmap scan (-A) of 192.168.10.1, indicating the host seems down.
- The third session shows another failed nmap scan (-A) of 192.168.10.1, with similar results.

The desktop background features a blue abstract pattern, and the taskbar at the bottom shows icons for various files and applications.

9. LESSON LEARNED AND RECOMMENDATIONS

Lessons Learned

1. Correct Network Segmentation Enables Effective Isolation

Placing the Kali attacker VM on a separate internal network (192.168.20.0/24) made it easier to detect and control traffic flows, proving how segmentation enhances visibility and security.

2. Suricata on pfSense Can Detect Basic Threats Efficiently

Suricata successfully picked up stealth scans and ping probes, reinforcing its value as a lightweight and powerful intrusion detection tool when properly configured on pfSense.

3. Detection Alone Isn't Enough - Response is Key

While Suricata detected attacks, it didn't block them automatically. Implementing a targeted pfSense firewall rule on the Kali interface provided an effective and immediate response mechanism.

4. SID Management Was Not Essential for This Use Case

Though Suricata supports SID Management for rule tuning and inline blocking, this project achieved its goal without using that feature. Blocking was instead implemented via firewall, showing there's more than one effective response path.

5. Simplicity and Clarity in Configuration Prevents Downtime

Keeping the Suricata configuration straightforward (monitoring only relevant interfaces) and using minimal, targeted scanning from Kali ensured a focused learning experience with observable outcomes.

6. Tool Synergy Is Powerful

Combining Suricata for detection with pfSense firewall for enforcement demonstrates a layered defense strategy — a foundational concept in real-world network security.

Recommendations for Future Implementations

1. Enable Logging and Monitoring on All Interfaces Early

Ensure the correct interfaces are monitored by Suricata from the beginning (e.g., LAN, attacker interface) to avoid blind spots during attack simulation.

2. Consider SID Management for Production Environments

For real-time blocking or deeper customization, SID Management can enhance Suricata's effectiveness by controlling rule actions (alert/drop/pass) at a granular level.

3. Use Wireshark for Deeper Traffic Analysis

While not needed here, incorporating Wireshark in future simulations can help with deeper traffic inspection and forensic analysis, especially when IDS alerts seem unclear.

4. Automate Response Where Possible

Integration of IDS with automated response systems (e.g., scripts, firewall APIs) can reduce manual intervention and enhance real-time threat mitigation.

5. Expand Threat Simulation Variety

Future tests should explore other forms of malicious traffic such as brute-force login attempts, malware downloads, or DNS tunneling to test detection rules more extensively.

This project successfully demonstrates the deployment and use of Suricata IDS on pfSense to detect and respond to malicious network activity in a controlled virtual lab environment. By simulating a basic reconnaissance attack using nmap and ICMP ping probes from a Kali attacker machine, we confirmed the ability of Suricata to identify suspicious activity on monitored interfaces.

Despite challenges such as the absence of SID Management options under the Kali interface, we effectively implemented a responsive mitigation strategy using pfSense firewall rules to block the attacker from further interacting with the internal LAN.

The entire process reinforced key cybersecurity concepts, including network segmentation, traffic monitoring, intrusion detection, and layered defense. The project also highlighted the value of simplicity, precision, and adaptability in configuring tools for specific network scenarios.

In real-world scenarios, integrating IDS with automated response systems and wider traffic analysis tools such as Wireshark would enhance visibility and security posture. However, for a lab-based educational task, this setup met the core objectives and provided a solid foundation in network threat detection and response.

Next Steps (Optional Enhancements for Future Work)

- Integrate Wireshark for full packet capture and traffic flow analysis.
- Simulate more advanced attacks (e.g., brute force, malware injection) for comprehensive rule validation.
- Explore automated blocking using Suricata's inline mode or integration with firewall APIs.
- Implement logging and alert forwarding to a central SIEM (e.g., Wazuh or ELK Stack) for enterprise-style threat monitoring.

10. CONCLUSION AND REFLECTION

Conclusion

This project was executed as Task 4 of the Code Alpha Internship Program, focusing on practical network defense through the use of Suricata IDS on pfSense. The goal was to set up a small virtualized network environment, simulate an attacker using Kali Linux, and implement detection and response mechanisms.

Through proper configuration of Suricata on pfSense, we achieved accurate detection of Nmap reconnaissance scans and ping probes. Although we initially explored Suricata's inline blocking capabilities, we opted for a manual response using pfSense firewall rules to block the attacker's IP address — demonstrating practical defensive action in a networked environment.

Key tasks completed:

- Setup of a multi-VM virtual network using Internal Networks on VirtualBox
- Installation and configuration of Suricata IDS on the pfSense router/firewall
- Execution of simulated attacks from Kali Linux
- Successful generation of detection alerts via Suricata
- Manual response by applying firewall rules on the attacker's interface

Reflection

This task provided hands-on exposure to critical cybersecurity operations including network segmentation, threat simulation, intrusion detection, and incident response. The challenge of ensuring proper interface selection and troubleshooting alert failures emphasized the importance of precise system configuration and real-time problem-solving.

Key Takeaways:

- **Traffic direction matters:** Suricata must monitor the correct interface where the threat enters
- **Tool limitations require adaptability:** Lack of SID Management on interface settings was mitigated by leveraging pfSense firewall
- **Basic attacks are useful for validation:** Simple tools like nmap and ping can effectively test detection pipelines
- **Firewall rules are still powerful:** Even without inline blocking from Suricata, we implemented solid mitigation

Overall, this task improved my technical agility, strengthened my IDS configuration skills, and deepened my understanding of how to pair detection tools with enforcement controls. It was a vital learning experience in building foundational competence for real-world cybersecurity roles.

11. REFERENCES

1. pfSense Documentation

Netgate Documentation. Configuring and Managing pfSense Firewall.

<https://docs.netgate.com/pfsense/en/latest/>

2. Suricata IDS Official Documentation

Open Information Security Foundation (OISF). Suricata - Open Source Threat Detection Engine.

<https://docs.suricata.io/>

3. Nmap Network Scanning Tool

Gordon Lyon (Fyodor). Nmap Security Scanner Manual and Examples.

<https://nmap.org/book/man.html>

4. VirtualBox User Manual

Oracle. VirtualBox Networking and VM Configuration.

<https://www.virtualbox.org/manual/UserManual.html>

5. Network Security Concepts

Cisco Press. Introduction to Intrusion Detection Systems and Basic Network Security.

(Used for background knowledge on IDS vs IPS, not directly cited)