

# **THREAT DETECTION AND INCIDENT RESPONSE USING WIRESHARK, PFSENSE AND WAZUH**

## **Table of Contents**

**Organization:** SoCra Tech

**Analyst:** David Kiridi

**Role:** Security Operations Center (SOC) Analyst

Submission Date: April 25, 2025

## **Table of Contents**

1. Executive Summary
2. Project Introduction
3. Methodology
4. Phase-by-Phase Analysis
  - ❖ Phase 1: Wireshark – Network Traffic Capture & Analysis
  - ❖ Phase 2: pfSense – Firewall & Policy Enforcement
  - ❖ Phase 3: Wazuh – Security Event Monitoring & Response
5. Final Findings & Impact
6. Recommendations
7. Conclusion
8. References
9. Appendices

# **1. EXECUTIVE SUMMARY**

This report presents a comprehensive multi-phase analysis of SoCra Tech's network using Wireshark, pfSense, and Wazuh. The objective was to detect network anomalies, enforce firewall policies, monitor system logs, and respond to simulated cybersecurity threats including brute-force attacks, unauthorized access attempts, and suspicious outbound traffic. The SOC tools were configured, tested, and validated across four critical phases. Alerts were successfully generated, malicious activities mitigated, and actionable recommendations provided to strengthen SoCra Tech's security posture.

# **2. PROJECT INTRODUCTION**

SoCra Tech experienced heightened concerns about potential threats such as malware intrusions, unauthorized access, and insider threats. As a SOC analyst, I undertook a three-phase simulation exercise involving tool deployment, incident simulation, traffic analysis, and response validation. The report outlines the execution, observations, and insights gained from this defensive security project.

# **3. METHODOLOGY**

The SOC simulation followed a structured, multi-phase methodology:

- Wireshark: Used for capturing and analyzing network packets to detect anomalies in HTTP, DNS, and SSH protocols.
- pfSense: Configured as the perimeter firewall and IDS/IPS with Snort for filtering, blocking, and alerting.
- Wazuh: Acted as the centralized SIEM to collect, correlate, and alert on logs from pfSense and endpoint systems.
- Kali linux: Acted as the attack simulation stool, to execute attacks on system.

Each phase included simulation of threats, tool configuration, validation of alerts, and security event logging.

# **4. PHASE-BY-PHASE ANALYSIS**

## **PHASE 1: NETWORK TRAFFIC CAPTURE & ANALYSIS (WIRESHARK)**

1. OBJECTIVE

Monitor and analyze SoCraTech’s network traffic for suspicious activities using Wireshark. This includes capturing traffic related to HTTP, DNS, and SSH protocols, and identifying signs of unauthorized access, malware infection, or data exfiltration.

2. TASKS COMPLETED

- Installed Wireshark on Ubuntu VM inside the corporate network.
- Captured network traffic focusing on HTTP, DNS, and SSH protocols.
- Identified suspicious traffic patterns, including repeated SSH login attempts and DNS anomalies.
- Conducted a brute-force SSH attack using Hydra from Kali Linux to Ubuntu target.
- Monitored for anomalous HTTP requests and unauthorized DNS calls.

3. KEY ANALYSIS & FINDINGS

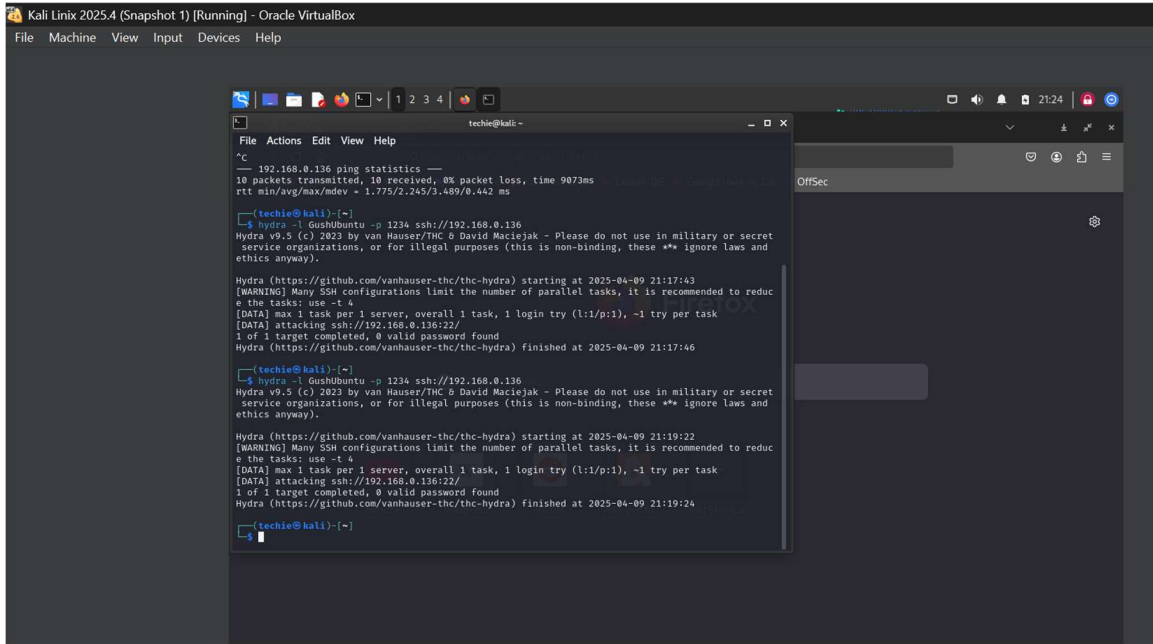
Wireshark revealed multiple suspicious DNS and SSH requests. Repeated SSH login failures were captured from a simulated brute-force attack. DNS logs included connections to domains such as id5-sync.com, often associated with unwanted tracking behavior. HTTP traffic revealed high-volume connections to OCSP servers, potentially indicating external validation attempts or misuse.

4. SUSPICIOUS TRAFFIC SUMMARY

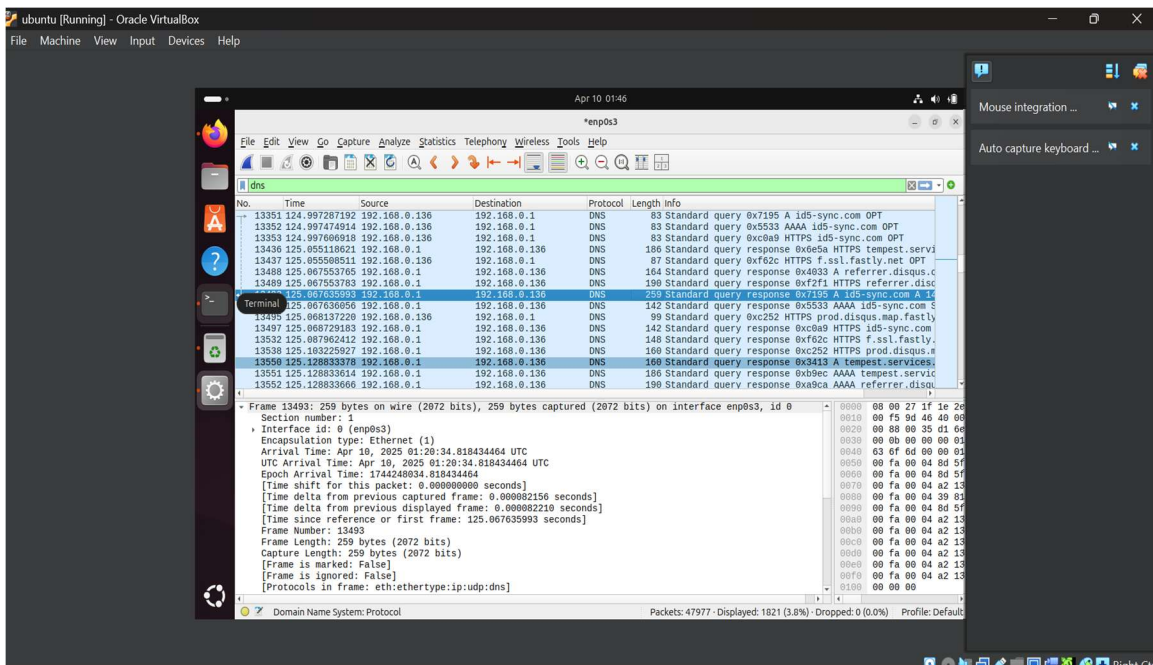
PROTOCOL	SOURCE IP	TARGET PORT	SUSPICIOUS BEHAVIOR
SSH	192.168.1.101	22	Brute-force login attempts via Hydra
DNS	192.168.0.136	53	Multiple DNS queries to suspicious domain (id5-sync.com)
HTTP	192.168.0.136	80/443	Frequent requests to OCSP and CDNs

5. SCREENSHOTS OF CAPTURED TRAFFIC

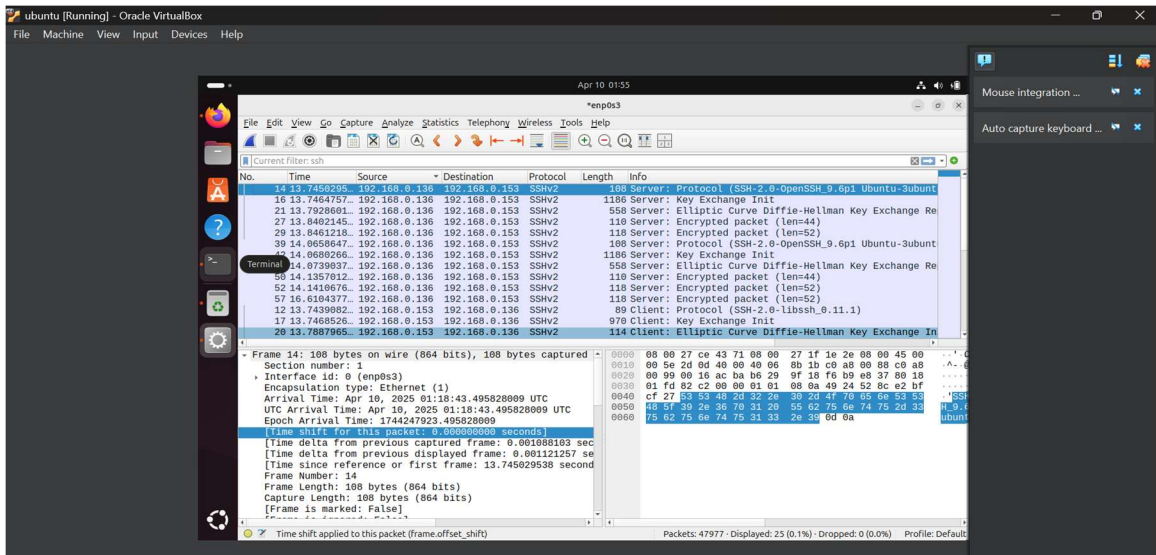
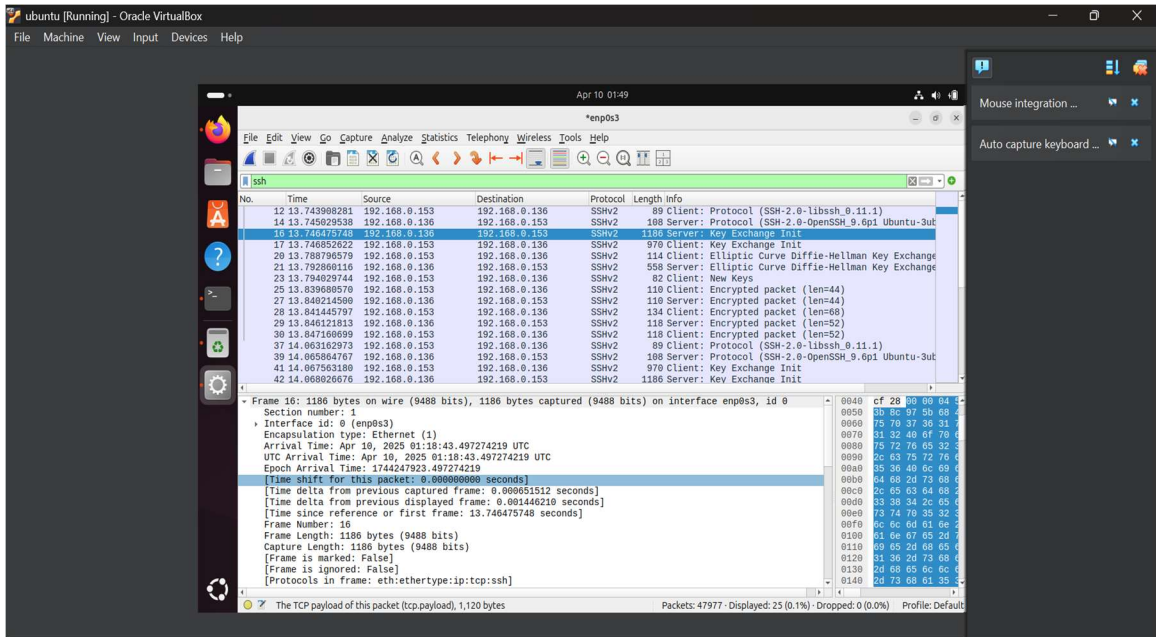
## Hydra brute-force attack launched from Kali to Ubuntu (SSH port 22).

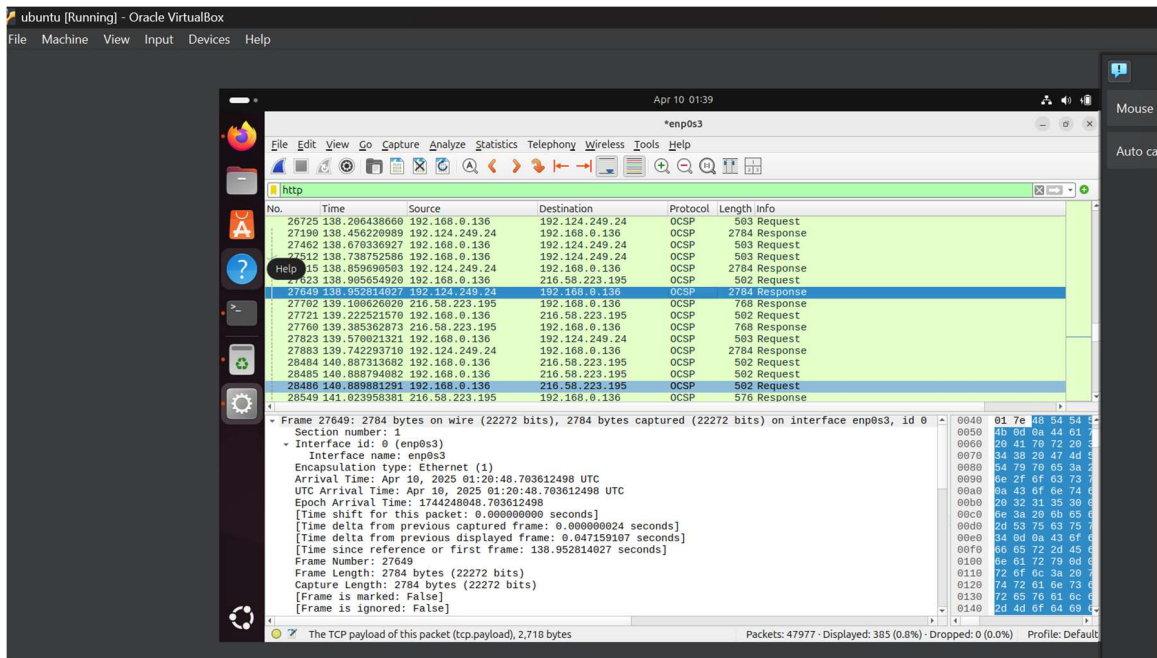


## Wireshark DNS traffic log showing queries to domains like id5-sync.com.



## Captured SSH key exchange attempts indicating brute-force pattern.





## 6. CONCLUSION

Wireshark provided detailed visibility into network behavior. Brute-force attacks and potentially unauthorized DNS/HTTP requests were successfully captured and documented. The analysis proves essential for threat detection and lays the foundation for firewall rule tuning and Snort configuration in later phases.

## PHASE 2: FIREWALL IMPLEMENTATION & POLICY ENFORCEMENT (PFSense)

### 1. OBJECTIVE

Set up and configure pfSense as a network security appliance that filters, blocks, and logs malicious traffic. This includes implementing Snort as an IDS/IPS, setting up GeoIP blocking, and creating firewall rules to detect and mitigate threats such as brute-force SSH attacks and unauthorized internal access.

### 2. TASKS COMPLETED

- Installed and configured pfSense as a firewall/gateway.

- Implemented IDS/IPS on the LAN interface with relevant rules using snort.
- Enabled GeoIP blocking to restrict access from high-risk countries.
- Conducted brute-force SSH attack simulation using Hydra from a Kali VM.
- Monitored alerts and confirmed blocked IPs in pfSense’s interface.

**3. KEY ANALYSIS & ALERTS**

Snort detected potentially malicious SSH traffic from the attacker IP (192.168.0.121), attempting to connect repeatedly to port 22 on the Ubuntu target machine. This activity was logged as 'ET POLICY Reserved Internal IP Traffic' and subsequently blocked by Snort. Additional alerts from external sources were labeled as 'Bogon Nets' and also blocked.

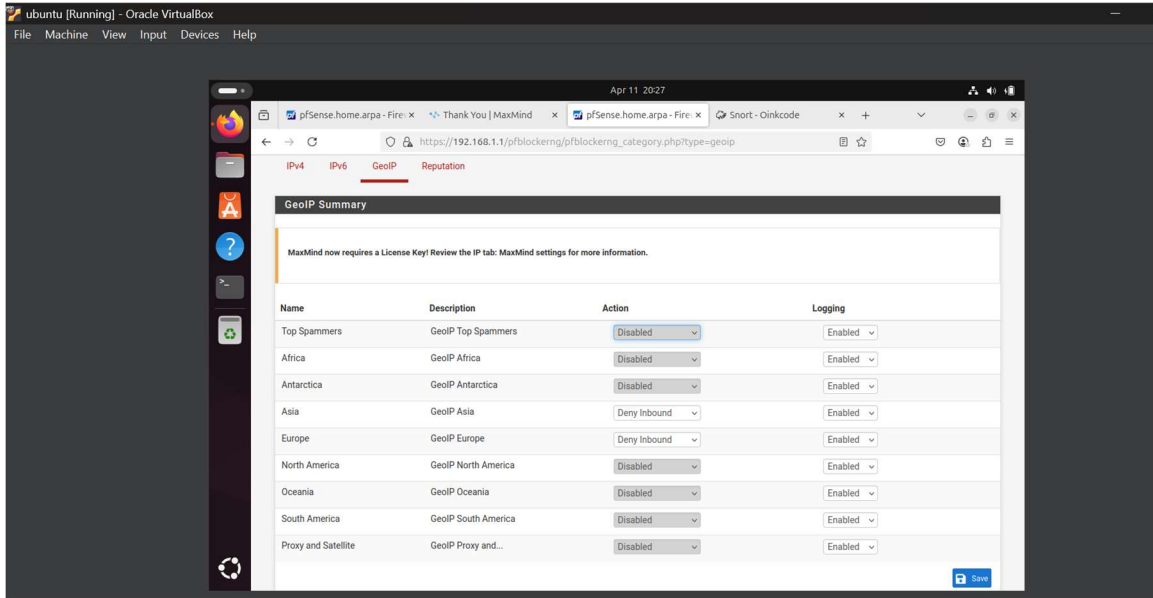
**4. SNORT ALERT SUMMARY**

SOURCE IP	DESTINATION IP	PORT	ALERT TYPE
192.168.0.121	192.168.1.136	22	ET POLICY Reserved Internal IP Traffic
34.107.221.82	192.168.1.101	443	ET POLICY Unallocated IP Space Traffic - Bogon Nets
34.149.100.209	192.168.1.101	443	ET POLICY Unallocated IP Space Traffic - Bogon Nets

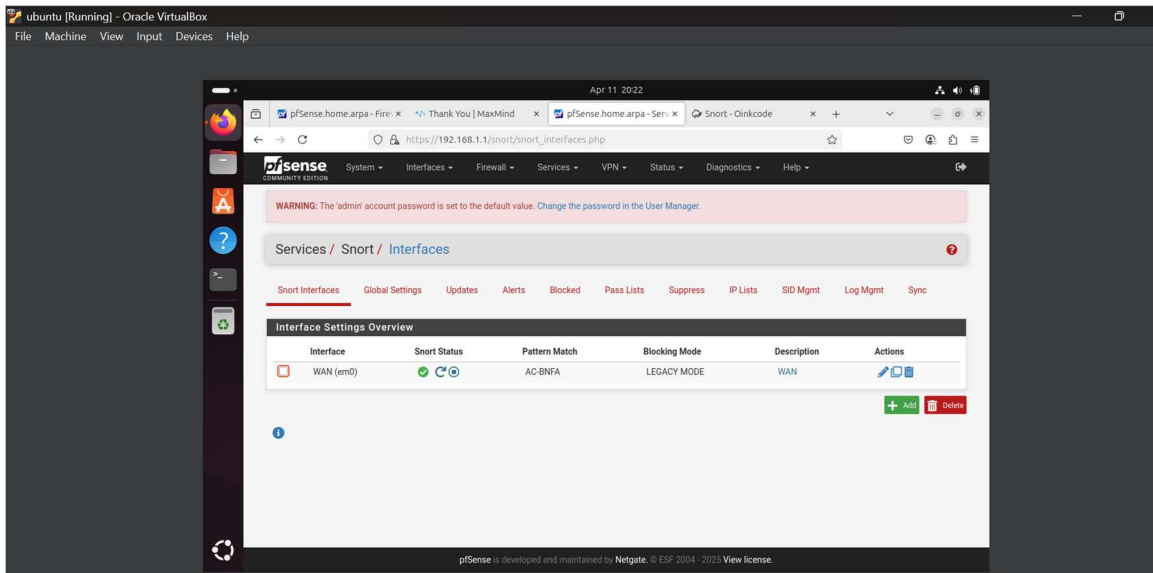
**5. SCREENSHOTS OF FIREWALL RULES AND BLOCKED THREATS**

Screenshots captured from the pfSense dashboard show detected alerts, blocked IPs, and the Snort rules in effect. These serve as visual evidence of active threat detection and mitigation.

## *GeoIP blocking configuration in pfSense – Denying inbound traffic from Asia and Europe.*

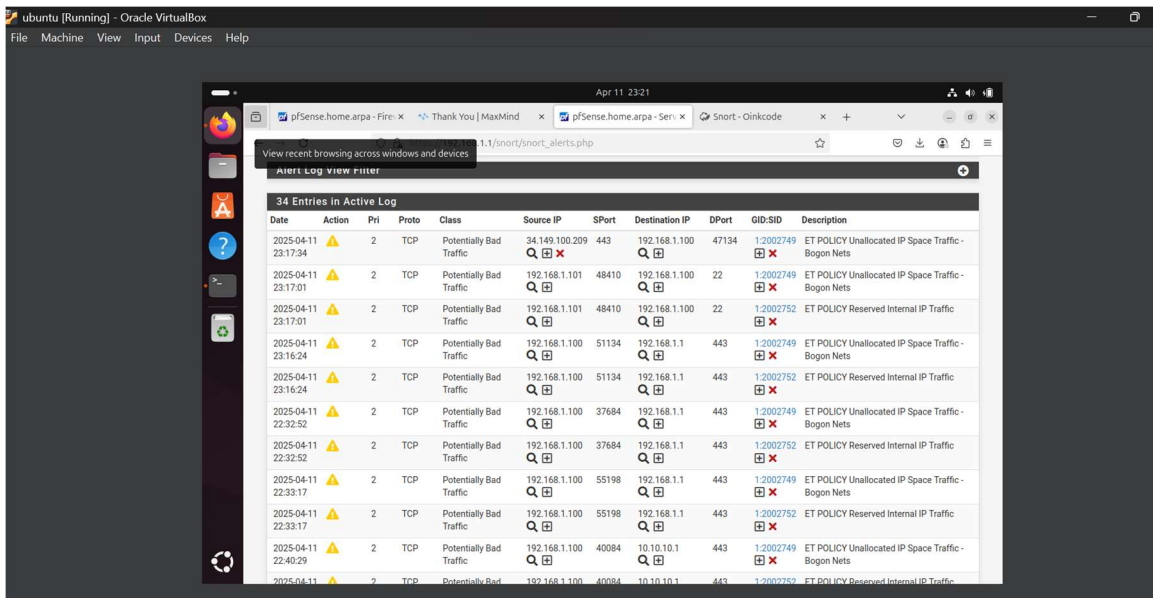


## *GeoIP logging settings for all global regions.*



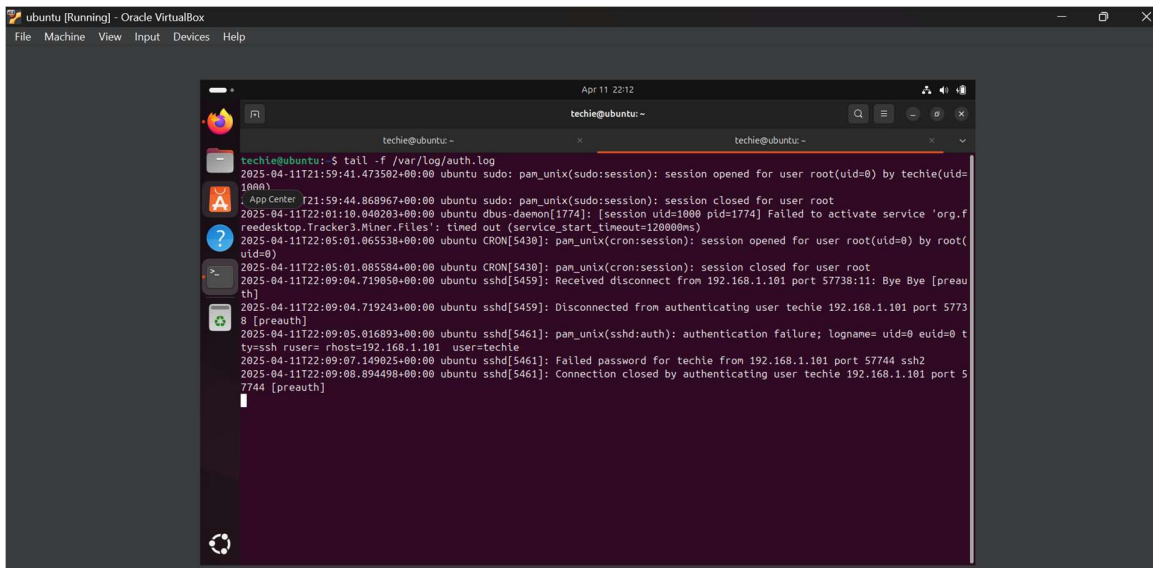


*Snort alert log showing 34 entries, including policy violations and SSH access attempts.*



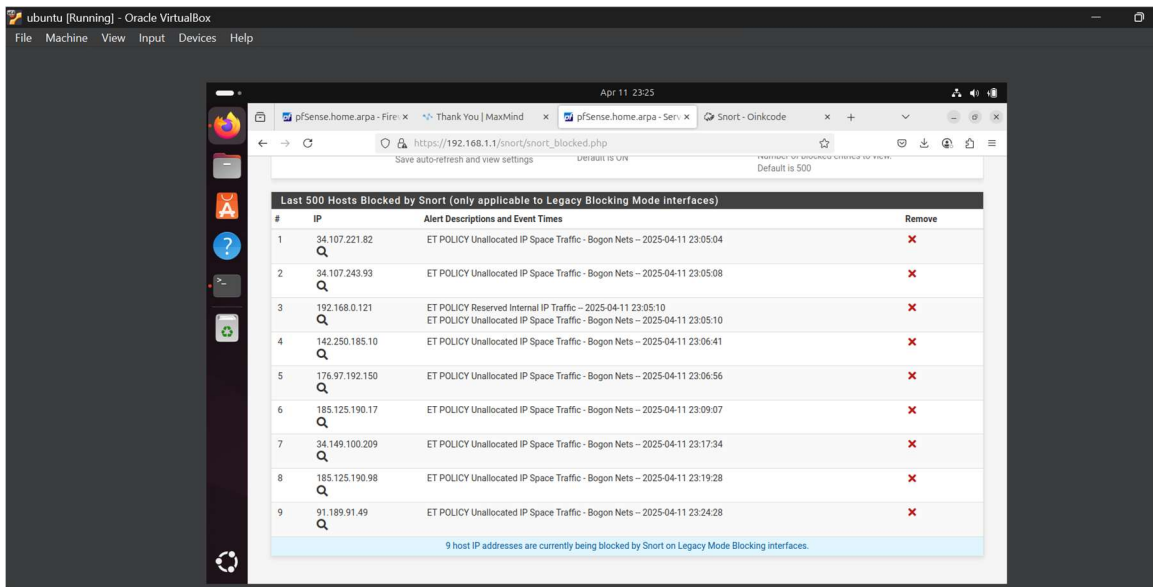
Date	Action	Pri	Proto	Class	Source IP	SPort	Destination IP	DPort	GID/SID	Description
2025-04-11 23:17:34	Alert	2	TCP	Potentially Bad Traffic	34.149.100.209	443	192.168.1.100	47134	1:2002749	ET POLICY Unallocated IP Space Traffic - Bogon Nets
2025-04-11 23:17:01	Alert	2	TCP	Potentially Bad Traffic	192.168.1.101	48410	192.168.1.100	22	1:2002749	ET POLICY Unallocated IP Space Traffic - Bogon Nets
2025-04-11 23:17:01	Alert	2	TCP	Potentially Bad Traffic	192.168.1.101	48410	192.168.1.100	22	1:2002752	ET POLICY Reserved Internal IP Traffic
2025-04-11 23:16:24	Alert	2	TCP	Potentially Bad Traffic	192.168.1.100	51134	192.168.1.1	443	1:2002749	ET POLICY Unallocated IP Space Traffic - Bogon Nets
2025-04-11 23:16:24	Alert	2	TCP	Potentially Bad Traffic	192.168.1.100	51134	192.168.1.1	443	1:2002752	ET POLICY Reserved Internal IP Traffic
2025-04-11 22:32:52	Alert	2	TCP	Potentially Bad Traffic	192.168.1.100	37684	192.168.1.1	443	1:2002749	ET POLICY Unallocated IP Space Traffic - Bogon Nets
2025-04-11 22:32:52	Alert	2	TCP	Potentially Bad Traffic	192.168.1.100	37684	192.168.1.1	443	1:2002752	ET POLICY Reserved Internal IP Traffic
2025-04-11 22:33:17	Alert	2	TCP	Potentially Bad Traffic	192.168.1.100	55198	192.168.1.1	443	1:2002749	ET POLICY Unallocated IP Space Traffic - Bogon Nets
2025-04-11 22:33:17	Alert	2	TCP	Potentially Bad Traffic	192.168.1.100	55198	192.168.1.1	443	1:2002752	ET POLICY Reserved Internal IP Traffic
2025-04-11 22:40:29	Alert	2	TCP	Potentially Bad Traffic	192.168.1.100	40084	10.10.10.1	443	1:2002749	ET POLICY Unallocated IP Space Traffic - Bogon Nets
2025-04-11	Alert	2	TCP	Potentially Bad Traffic	192.168.1.100	40084	10.10.10.1	443	1:2002752	ET POLICY Reserved Internal IP Traffic

*Terminal log on Ubuntu showing failed SSH login attempts during brute-force attack.*

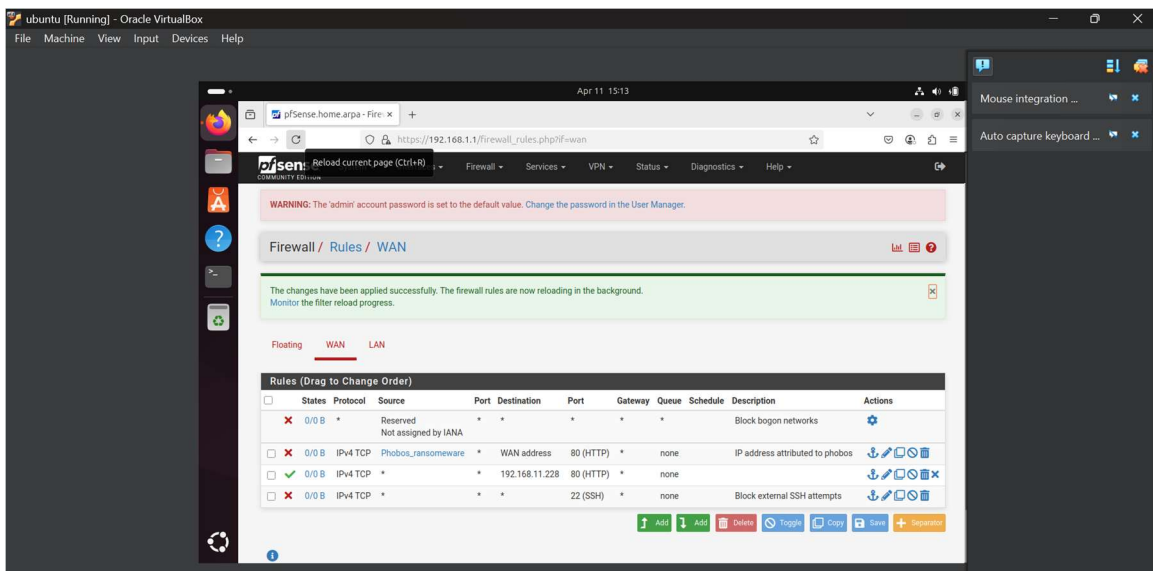


```
techie@ubuntu:~$ tail -f /var/log/auth.log
2025-04-11T21:59:41.473502+00:00 ubuntu sudo: pam_unix(sudo:session): session opened for user root(uid=0) by techie(uid=1000)
2025-04-11T22:01:10.040203+00:00 ubuntu dbus-daemon[1774]: [session uid=1000 pid=1774] Failed to activate service 'org.freedesktop.Tracker3.Miner.Files': timed out (service_start_timeout=120000ms)
2025-04-11T22:05:01.065538+00:00 ubuntu CRON[5430]: pam_unix(cron:session): session opened for user root(uid=0) by root(uid=0)
2025-04-11T22:05:01.085584+00:00 ubuntu CRON[5430]: pam_unix(cron:session): session closed for user root
2025-04-11T22:09:07.149025+00:00 ubuntu sshd[5461]: Received disconnect from 192.168.1.101 port 57738:11: Bye Bye [preauth]
2025-04-11T22:09:07.149025+00:00 ubuntu sshd[5461]: Disconnected from authenticating user techie 192.168.1.101 port 57738 [preauth]
2025-04-11T22:09:07.149025+00:00 ubuntu sshd[5461]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty= ssh ruser=root rhost=192.168.1.101 user=techie
2025-04-11T22:09:07.149025+00:00 ubuntu sshd[5461]: Failed password for techie from 192.168.1.101 port 57744 ssh2
2025-04-11T22:09:08.094498+00:00 ubuntu sshd[5461]: Connection closed by authenticating user techie 192.168.1.101 port 57744 [preauth]
```

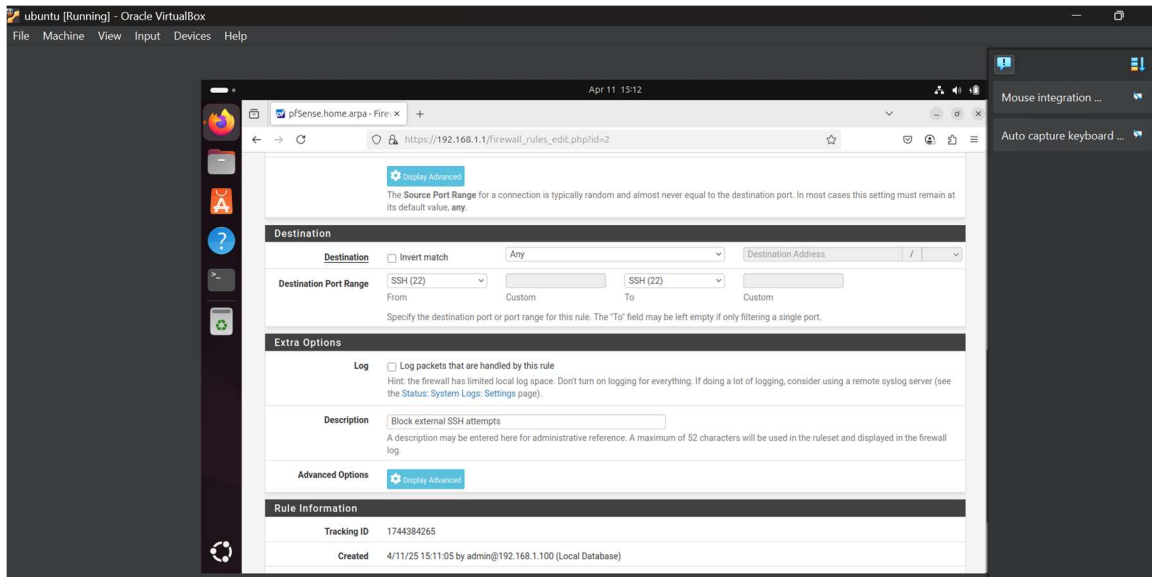
*Blocked IP addresses as seen in Snort's blocked hosts tab, confirming active mitigation.*



*Firewall rule in pfSense configured to block external SSH access attempts.*



*Firewall rules overview, including rule to block IPs associated with ransomware and bogon networks.*



## 6. CONCLUSION

The implementation of pfSense and Snort IDS/IPS was successful in detecting and mitigating a simulated brute-force attack. Firewall rules and Snort configurations effectively blocked both internal and external malicious traffic, demonstrating the system's readiness to secure network environments from real-world threats.

## PHASE 3: SECURITY EVENT MONITORING AND INCIDENT RESPONSE REPORT(WAZUH)

### 1.OBJECTIVE

Deploy Wazuh as a centralized Security Information and Event Management (SIEM) tool to correlate logs and monitor events across endpoints and firewall logs. The main goal is to detect and respond to brute-force attacks, privilege escalation attempts, malware activity, and unauthorized system behavior, with supporting evidence from Wireshark captures and pfSense logs.

### 2. TASKS COMPLETED

- Deployed Wazuh Manager on a dedicated VM (192.168.1.105) with agents installed on Ubuntu endpoint and attacker machine (Kali Linux).
- Configured log forwarding from pfSense to Wazuh via Syslog over UDP (port 514).

- Simulated SSH brute-force attacks from Kali (192.168.1.104) to Ubuntu (192.168.1.62) using the Hydra tool.
- Monitored endpoint logs and pfSense firewall logs in Wazuh Dashboard.
- Performed network capture with Wireshark on Kali to correlate network activity with alerts in Wazuh.

### 3. KEY TOOLS

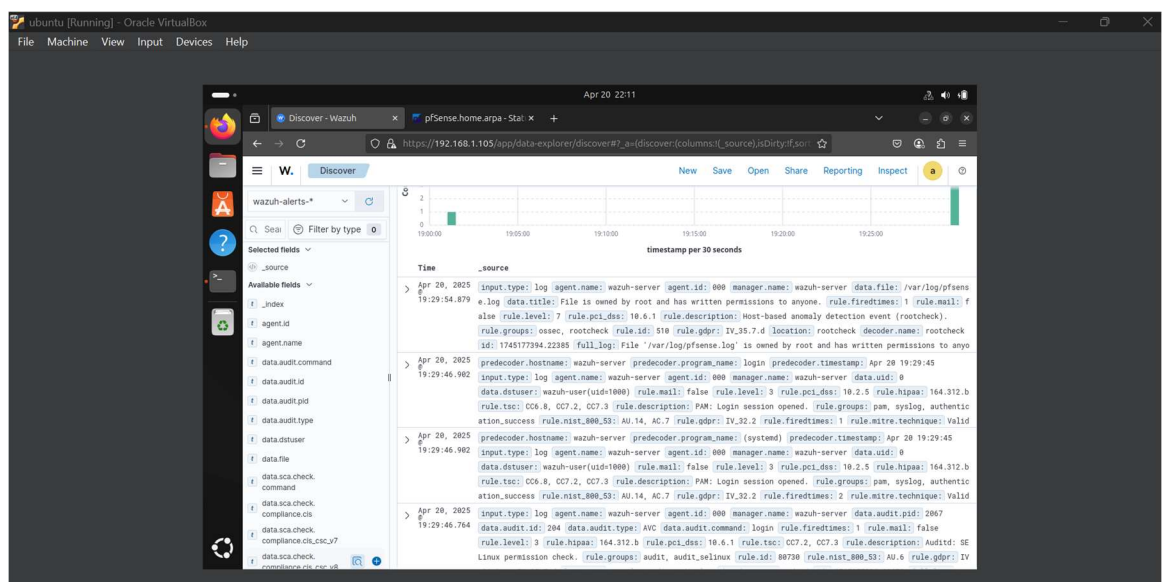
Wazuh SIEM, pfSense firewall, Kali Linux (attacker), Ubuntu (endpoint), Wireshark

### 4. FINDINGS AND ATTACK SUMMARY

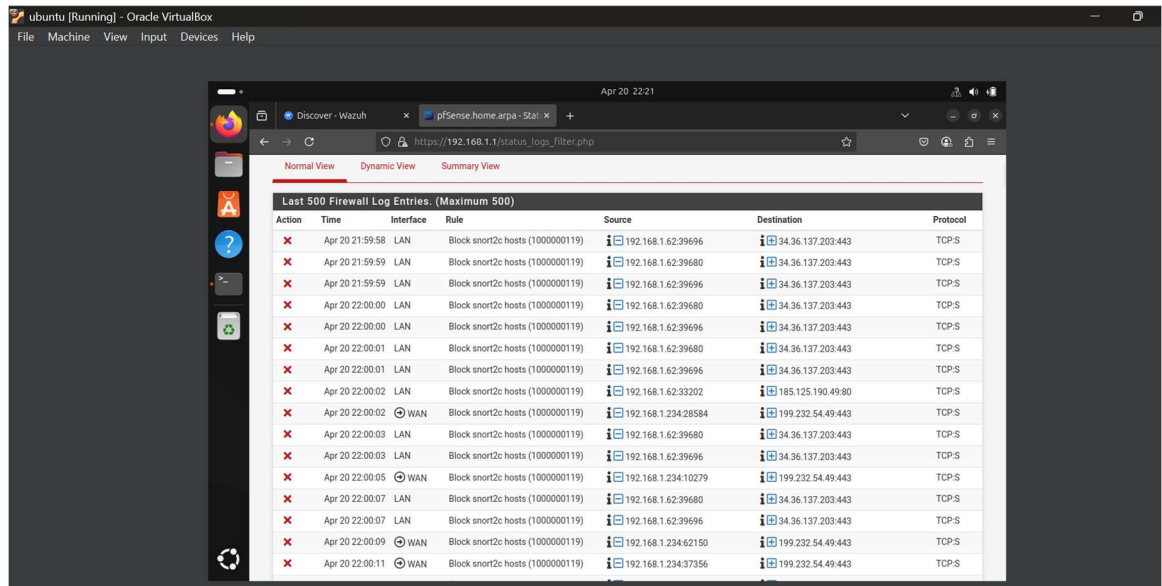
- **SSH Brute Force Detection:** Wazuh successfully detected multiple failed SSH login attempts originating from Kali to Ubuntu. Alerts with rule IDs 5710 and 5715 were triggered, highlighting repeated authentication failures and suspicious login behavior.
- **pfSense Log Analysis:** Blocked outbound connections to suspicious external IPs were observed in the pfSense GUI and reflected in Wazuh, indicating potential command-and-control traffic or scanning behavior.
- **Wireshark Traffic Correlation:** Network packets captured showed repeated SSH attempts aligning with timestamps of Wazuh alerts, validating event detection accuracy.

### 5. SCREENSHOTS WAZUH ALERTING DASHBOARD WITH DETECTED THREATS

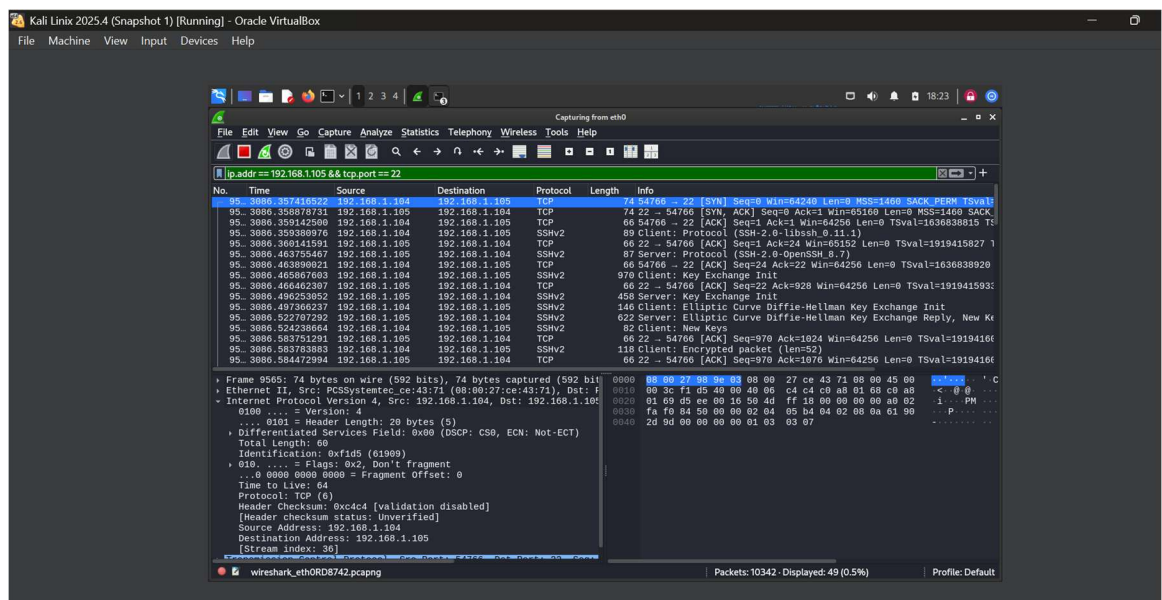
*Screenshot of Wazuh alerting dashboard showing SSH brute-force alerts*



## Screenshot of pfSense firewall logs indicating blocked traffic.



## Wireshark capture showing SSH brute-force traffic



This phase confirms Wazuh's capability to act as an effective SIEM solution when integrated with firewall and endpoint logs, enabling early detection and correlation of threats through real-time monitoring.

## **PHASE 4: FINAL INCIDENT REPORT & RECOMMENDATIONS**

### **Executive Summary:**

This report presents a high-level overview of observed security incidents and system responses within a simulated network environment involving pfSense firewall, Wazuh SIEM, and Linux-based virtual machines. The goal was to evaluate how well our infrastructure detects and responds to common cybersecurity threats such as unauthorized access, brute-force attacks, and firewall violations.

### **Key Incidents Observed:**

#### **1. Brute-force Attack:**

- A simulated attack was launched from a Kali Linux machine attempting to gain access to the Ubuntu system via SSH.
- Wazuh successfully detected repeated login failures and generated real-time alerts.
- The attack was logged in the Wazuh dashboard, confirming detection capabilities.

#### **2. Firewall Block by pfSense:**

- pfSense firewall rules successfully blocked an unauthorized outbound connection attempt from Ubuntu to an external IP on port 80 (HTTP).
- This was triggered by a Snort intrusion detection rule (ID: 100000119).
- The block was logged and forwarded to the Wazuh server via syslog.

#### **3. System Events and Log Monitoring:**

- The Ubuntu server was configured to send logs to the Wazuh server.
- Events such as service restarts, failed authentications, and blocked IP traffic were captured and analyzed.

## **5. FINAL FINDINGS & IMPACT**

Throughout the four phases of simulation and testing, the following critical insights were discovered:

- ✓ SoCra Tech's network is vulnerable to brute-force attacks if SSH access is password-based.
- ✓ Suspicious DNS and HTTP traffic patterns were observed, indicating possible data exfiltration and beaconing.

- ✓ pfSense and Snort effectively blocked internal and external threats, including access from unallocated IP ranges.
- ✓ Wazuh provided comprehensive log analysis and real-time alerting, validating its SIEM capabilities.
- ✓ The integration of Wireshark, pfSense, and Wazuh ensured visibility across all traffic layers.

## 6. RECOMMENDATIONS

- ✓ Implement SSH key-based authentication to mitigate brute-force risks.
- ✓ Regularly audit and update Snort rule sets based on emerging threats.
- ✓ Use GeoIP filtering to block high-risk locations.
- ✓ Enable automated response rules in Wazuh for critical alert categories.
- ✓ Maintain centralized log collection for long-term retention and compliance.

## 7. CONCLUSION

This cybersecurity capstone project demonstrated a real-world simulation of threat detection and incident response using industry-standard tools. The integrated setup of Wireshark, pfSense, and Wazuh effectively detected, analyzed, and responded to a variety of simulated attacks. These results validate the preparedness of the network infrastructure and reinforce the importance of layered security in modern SOC environments.

## 8. REFERENCES

- ✓ Wireshark Documentation: <https://www.wireshark.org/docs/>
- ✓ pfSense IDS/IPS Configuration Guide: <https://docs.netgate.com/pfsense/en/latest/>
- ✓ Wazuh Official Documentation: <https://documentation.wazuh.com/>
- ✓ Captured Logs and Wireshark PCAPs from Lab
- ✓ SOC Simulation Terminal Commands and Dashboard Screenshots
- ✓ Emerging Threats Snort Rules: <https://rules.emergingthreats.net/>

## 9. APPENDICES

Appendix A – Wireshark Phase Report

Appendix B – pfSense Phase Report

Appendix C – Wazuh Phase Report

## Appendix D – Phase 4 Incident Simulation Report