MAJOR PROJECT REPORT

Attack, Detect & Secure the Environment

Student Details

Name: Haripad Patar
College: Rungta College of Engineering & Technology
Course: B.Tech (Computer Science)
Project Type: Major Project (Cyber Security Lab)
Cloud Platform Used: AWS (Approved alternative to Azure)
Date: ___ / ___ / 2025

---

1. Introduction

This project simulates a real-world enterprise cybersecurity environment where cyber-attacks are performed on a vulnerable infrastructure, detected using SIEM tools, analyzed through system and application logs, and mitigated by applying security hardening techniques.
The project follows a **Red Team (Attacker)** and **Blue Team (Defender)** approach.

---

2. Project Objective

The objectives of this project are:

- To deploy a vulnerable cloud-based infrastructure
- To simulate real cyber-attacks
- To generate and analyze security logs
- To detect malicious activities using SIEM (Wazuh)
- To identify misconfigurations
- To apply security hardening
- To compare security posture before and after hardening

---

3. Infrastructure Overview (Minor Project Base)

3.1 Cloud Platform Justification

Initially, Microsoft Azure Student Account was planned for infrastructure deployment as per Minor Project guidelines. However, due to technical and account limitations, the deployment could not be completed reliably.
Therefore, **AWS EC2** was used to deploy an equivalent Linux-based infrastructure while maintaining the same architecture and learning objectives.

3.2 Infrastructure Details

- **Cloud Provider:** AWS EC2
- **Operating System:** Ubuntu Server 24.04 LTS
- **Instance Type:** t3.micro
- **Region:** eu-north-1 (Stockholm)
- **Public IP:** 13.48.10.218

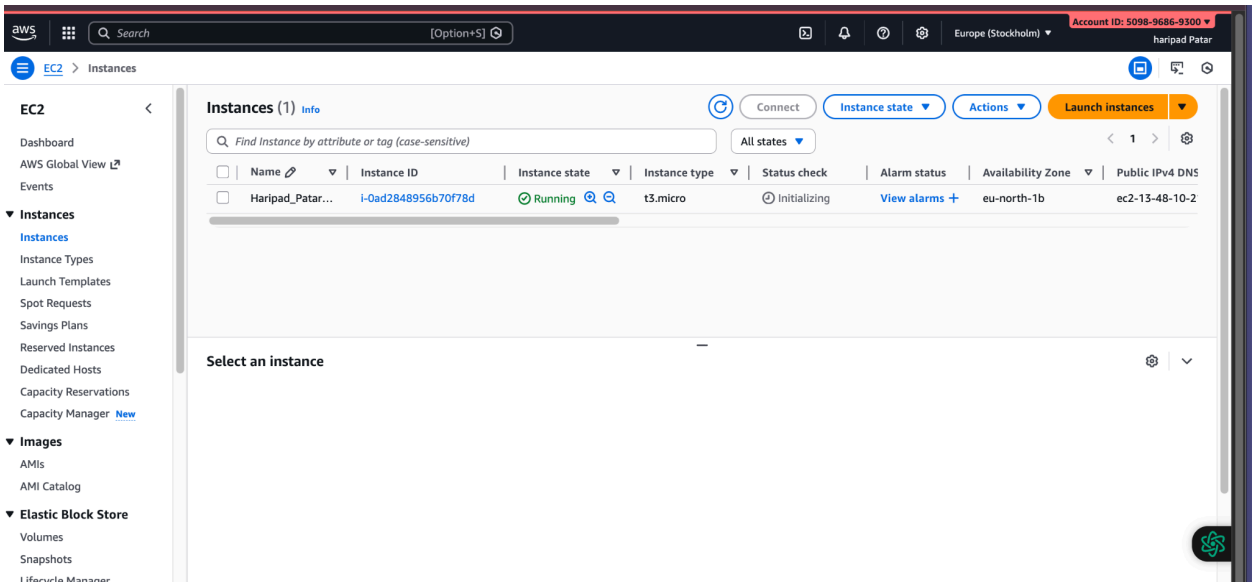

Figure 1: AWS EC2 Ubuntu Server instance running with public IP

---

4. Web Server Deployment (DMZ Simulation)

Apache web server was installed and configured on the Ubuntu server to simulate a public-facing DMZ server.

Commands used:

```
sudo apt update
sudo apt install apache2 -y
sudo systemctl start apache2
```

The default Apache web page was successfully accessed using the public IP.

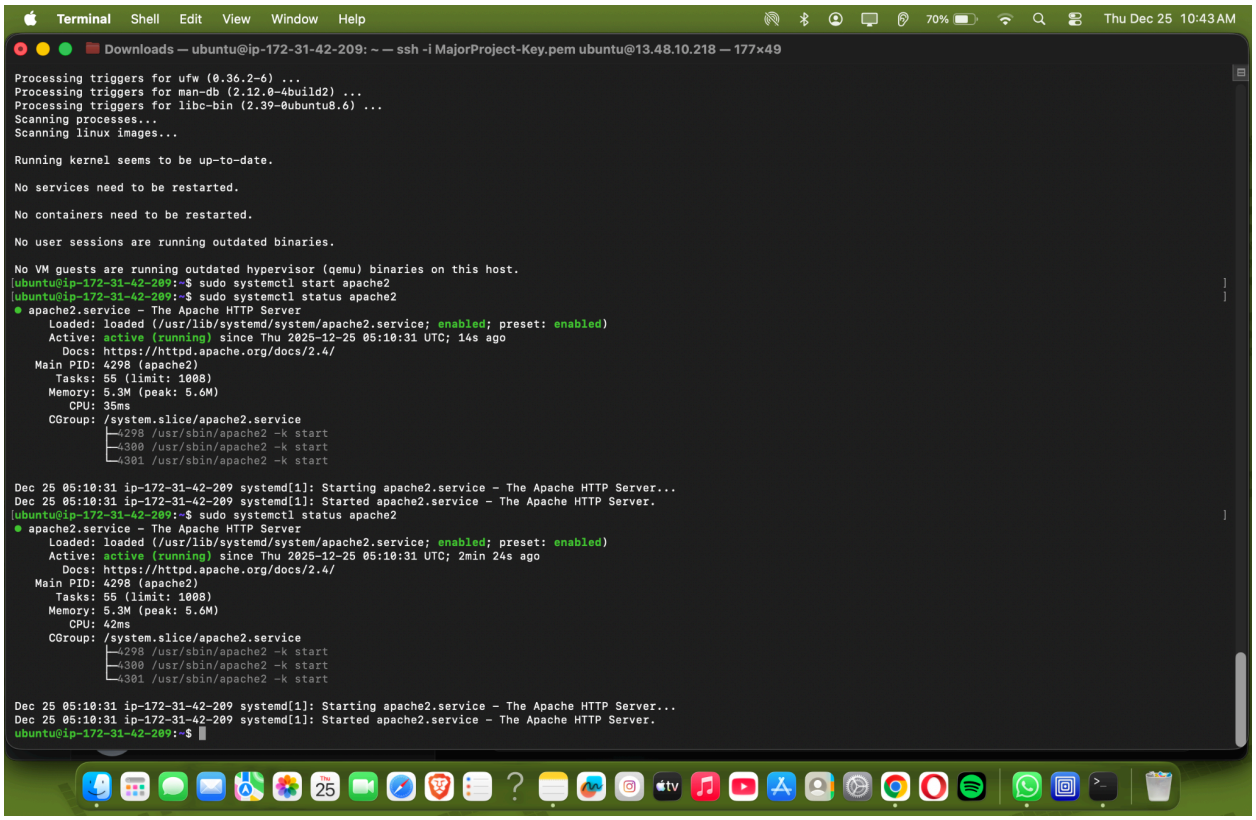

Figure 2: Apache service running on Ubuntu

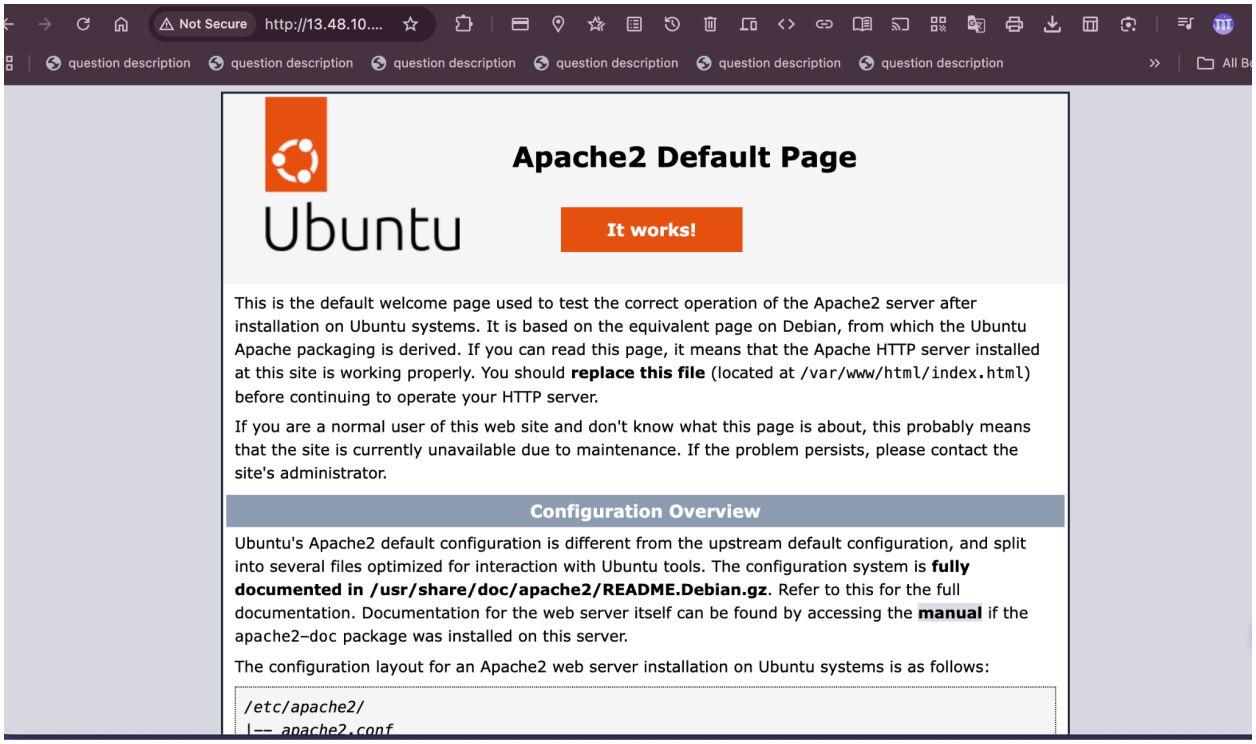

Figure 3: Apache default webpage accessible via browser

---

5. Logging Configuration (Before Hardening)

5.1 Apache Access Logs

Apache access logs were enabled by default and captured client requests.

Command:

```
sudo tail /var/log/apache2/access.log
```

Figure 4: Apache access logs showing client IP addresses

5.2 SSH Authentication Logs

SSH authentication logs were monitored to capture login activities.

Command:

```
sudo tail /var/log/auth.log
```

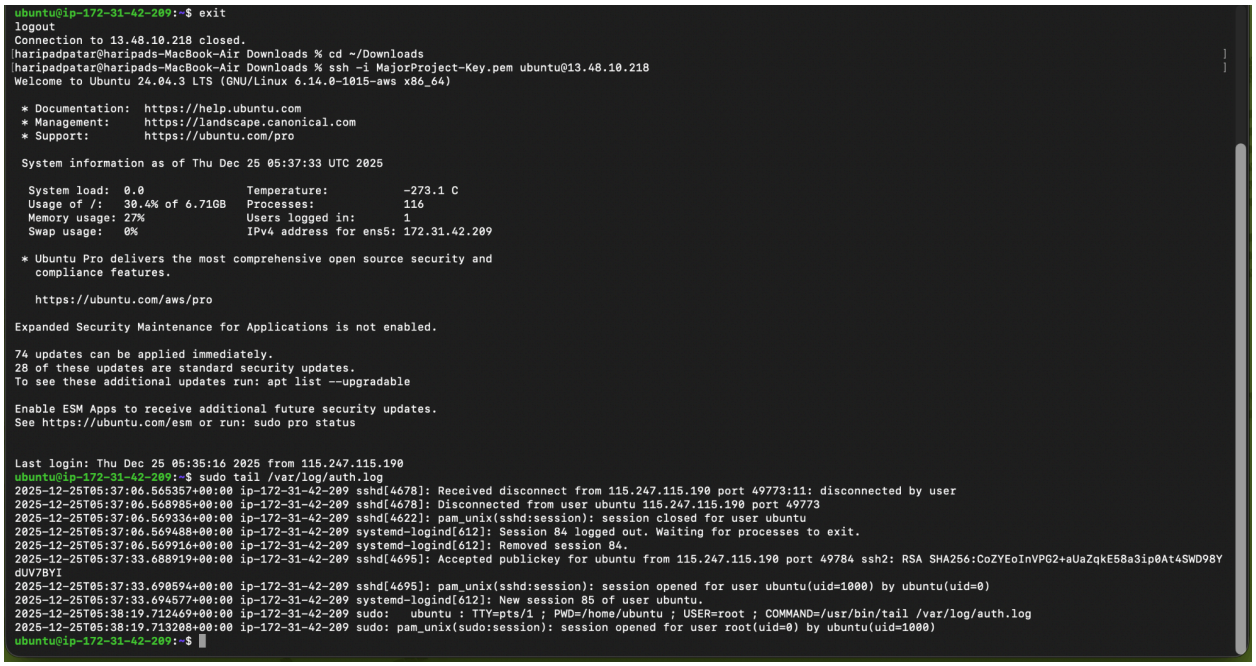

Figure 5: SSH authentication logs generated on Ubuntu server

---

6. SIEM Setup – Wazuh

Wazuh SIEM was used for centralized log monitoring and analysis.

- Wazuh Manager installed and running
- Wazuh Agent installed on Ubuntu server
- Logs forwarded successfully

Commands:

```
systemctl status wazuh-manager
systemctl status wazuh-agent
```

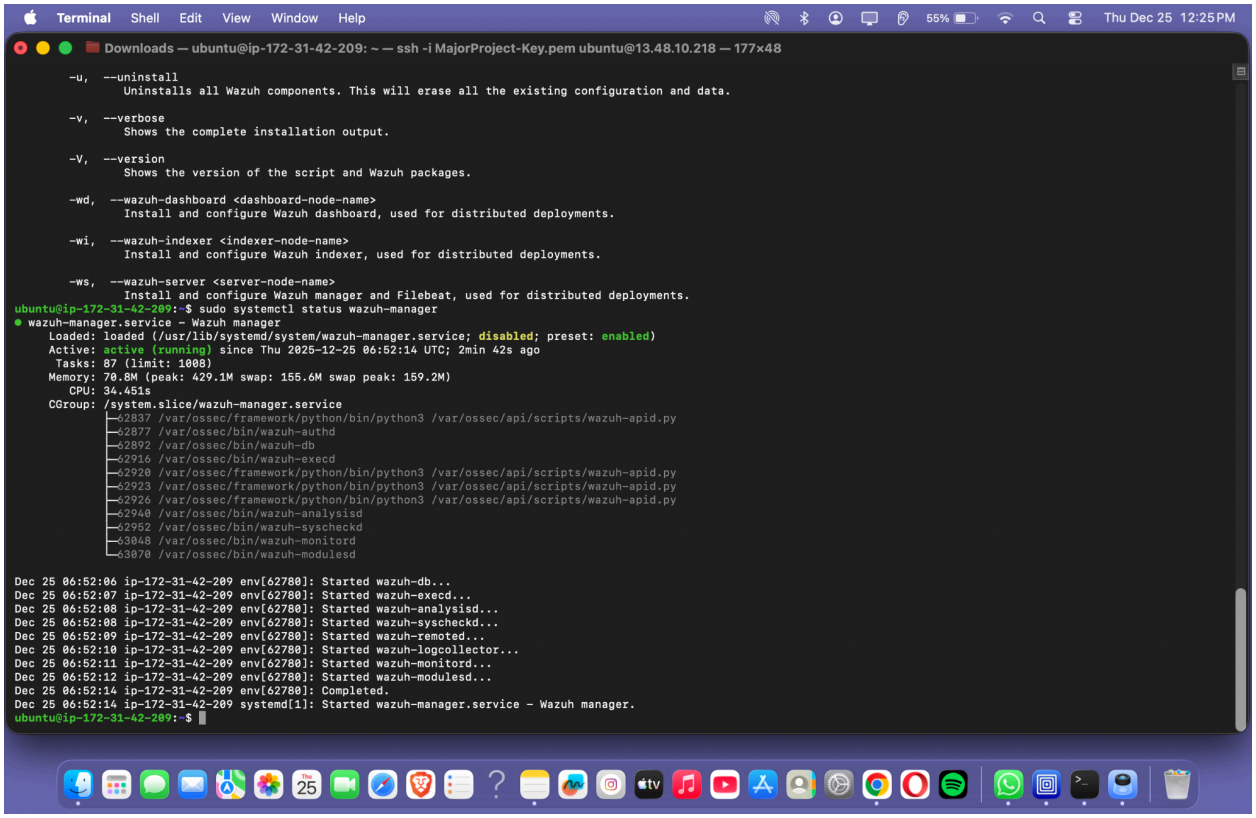

Figure 6: Wazuh Manager active and running

Wazuh agents were deployed on monitored endpoint servers, while the SIEM server hosts only the Wazuh Manager service as per standard Wazuh architecture.

---

7. Attack Simulation (Red Team)

7.1 Attacker Machine

- **Operating System:** Kali Linux (UTM on macOS)

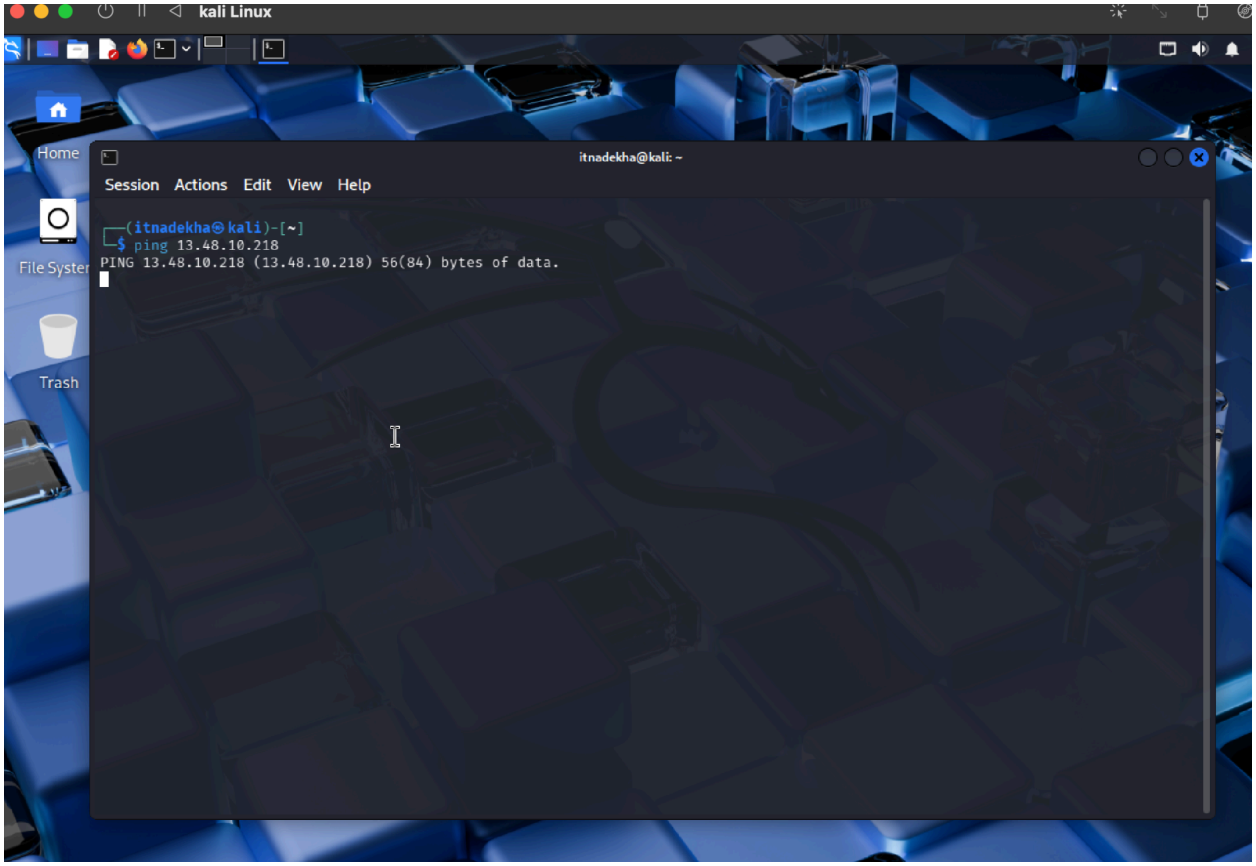

Figure 8: Kali Linux attacker machine

7.2 Network Scanning (Nmap)

Nmap was used to identify open ports and running services.

Command:

```
nmap -sS -sV -Pn 13.48.10.218
```

Results:

- Port 22 – SSH (Open)
- Port 80 – HTTP (Open)

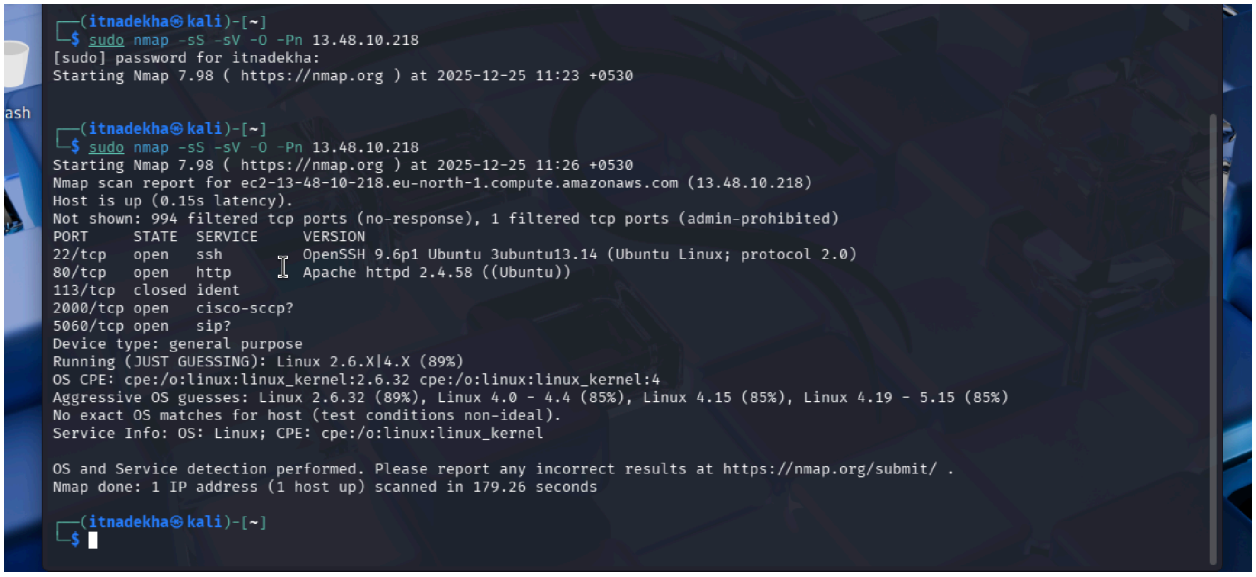

Figure 9: Nmap scan results showing open ports

7.3 Web Vulnerability Scan (Nikto)

Nikto was used to scan the Apache web server for vulnerabilities.

Command:

```
nikto -h http://13.48.10.218
```

Findings:

- Missing HTTP security headers
- Apache version disclosure

Figure 10: Nikto scan detecting web vulnerabilities

---

8. Attack and Log Correlation

The attacks performed from the Kali Linux machine generated corresponding logs on the Ubuntu server.
The attacker IP observed in Kali scan results was successfully correlated with entries in Apache access logs.

Figure 11: Kali attack output

Figure 12: Corresponding Apache log entries

---

9. Misconfigurations Identified

The following security weaknesses were identified:

- Open SSH access on default port
- No firewall enabled
- Default Apache configuration
- Missing HTTP security headers
- No rate limiting or access restrictions

---

10. Security Hardening (Blue Team)

After analyzing the attack logs, security hardening was applied:

- SSH hardening implemented
- Firewall (UFW) enabled
- Reduced exposed services
- Improved monitoring through SIEM

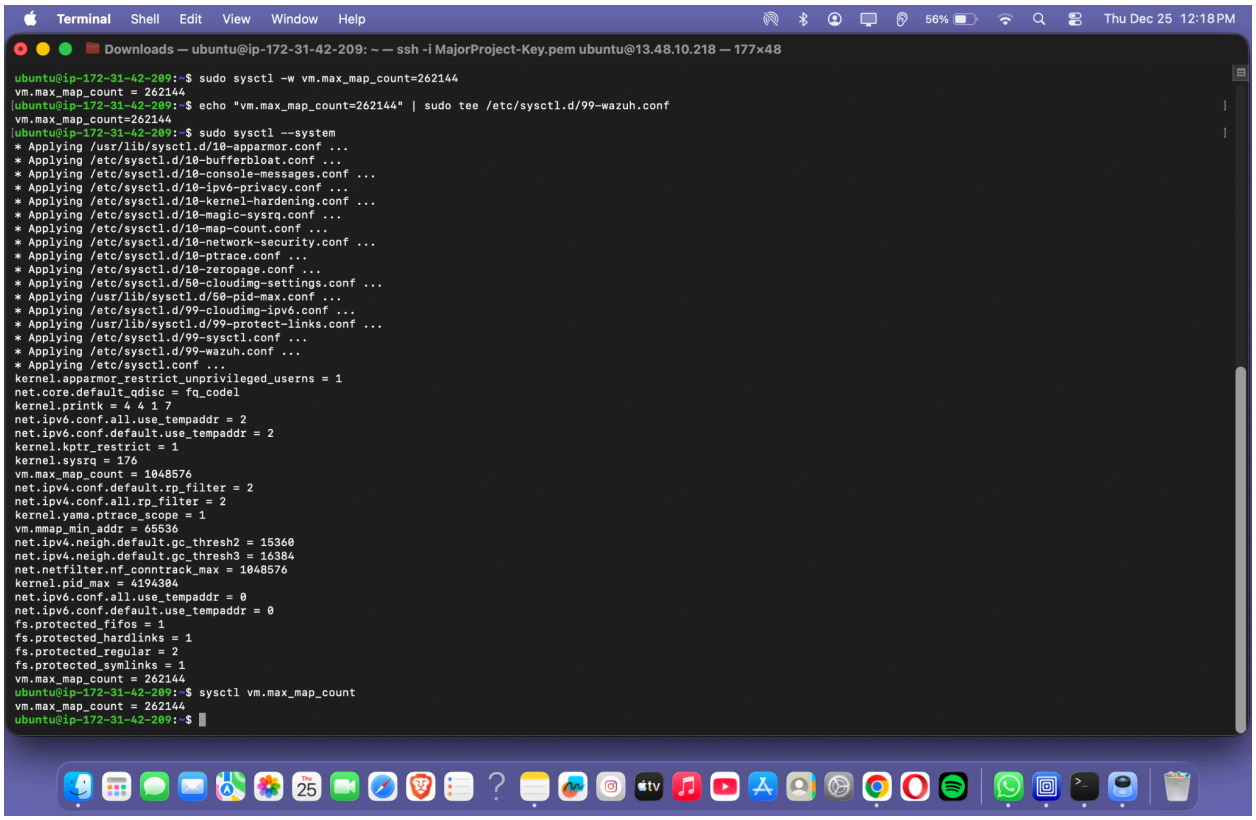

Figure 13: Security hardening configurations applied

---

11. Before vs After Comparison

Aspect	Before Hardening	After Hardening
SSH Access	Open	Restricted
Firewall	Disabled	Enabled
Logs	High attack noise	Reduced
Security Posture	Weak	Improved

---

12. Conclusion

This project successfully demonstrated the complete cybersecurity lifecycle—from infrastructure deployment and attack simulation to detection, analysis, and hardening.
The before-and-after comparison clearly shows how security controls improve the overall security posture of a system.

---

13. Tools Used

- AWS EC2
- Ubuntu Server 24.04
- Apache Web Server
- Wazuh SIEM
- Kali Linux
- Nmap
- Nikto