Comprehensive SOC Incident Response

This capstone project demonstrates an **end-to-end SOC incident response lifecycle**.
A complex cyberattack is simulated against a vulnerable system, detected using SIEM, triaged through case management, contained via automated response, analyzed for root cause, and reported to stakeholders.

Objective:

- Validate SOC readiness
- Measure detection and response effectiveness
- Demonstrate SOAR and adversary emulation capabilities

2 Lab Environment

Component	Details
Attacker Machine	Kali Linux
Victim Machine	Metasploitable2
SIEM	Wazuh
Case Management	TheHive
Threat Prevention	CrowdSec
Adversary Emulation	MITRE Caldera
Analytics	Elastic Security
Documentation	Google Docs

3 Attack Simulation (Metasploit)

Attack Vector: Samba Usermap Script Vulnerability
Technique: Remote Code Execution

Exploit Used:

```
exploit/multi/samba/usermap_script
```

Steps Performed:

1. Attacker scans Metasploitable2.
2. Samba service detected as vulnerable.
3. Exploit executed via Metasploit.
4. Reverse shell access gained on victim.

Result:
Unauthorized remote access successfully achieved.

4 Adversary Emulation (MITRE Caldera)

MITRE Technique: T1210 – Exploitation of Remote Services

Caldera simulated post-exploitation behavior to validate detection consistency.

Detection Logged in Wazuh

Timestamp	Source IP	Alert Description	MITRE Technique
2025-08-18 16:00:00	192.168.1.102	Samba exploit detected	T1210

5 Detection & Triage

Wazuh Detection

- Logs correlated from Samba service
- Abnormal command execution detected
- Alert severity: **High**

TheHive Triage

Field	Value
Case Title	Samba Exploitation – Metasploitable2
Severity	High
Status	Confirmed Incident
Tactics	Initial Access, Lateral Movement
Analyst Action	Escalated for containment

6 Response & Containment

Actions Taken:

- Victim VM isolated from network
- Attacker IP blocked using CrowdSec
- Firewall rules enforced automatically

Verification:

```
ping 192.168.1.102
Request timed out
```

Containment successful

7 SOAR Automation

Automated Playbook Flow:

1. Wazuh alert received
2. TheHive case auto-created
3. IP reputation checked
4. CrowdSec blocks malicious IP
5. Case updated with response status

Execution Status: ✅ Successful

8 Post-Incident Analysis

5 Whys – Root Cause Analysis

Why	Answer
Why was system compromised?	Vulnerable Samba service
Why vulnerable?	Outdated software
Why not updated?	No patch management
Why no monitoring?	Lack of asset visibility
Root Cause	Poor vulnerability management

Fishbone Diagram (Draw.io)

Causes Identified:

- Technology: Unpatched Samba
- Process: No vulnerability scans
- People: Misconfiguration
- Policy: Weak update policy

9 Metrics Reporting (Elastic Security)

Metric	Value
MTTD (Mean Time to Detect)	15 minutes
MTTR (Mean Time to Respond)	30 minutes
Dwell Time	45 minutes

A SOC dashboard was created showing:

- Alert timeline
- Detection latency

- Response effectiveness

10 Incident Report

Executive Summary

On August 18, 2025, a simulated cyberattack exploiting a Samba vulnerability was detected in the SOC environment. The attack resulted in unauthorized remote access to a Metasploitable2 system. Security controls successfully detected, contained, and remediated the incident within acceptable response timelines.

Timeline

- 15:45 – Exploit executed
- 16:00 – Alert generated
- 16:10 – Case triaged
- 16:30 – IP blocked
- 16:45 – Incident closed

Root Cause

The incident occurred due to an unpatched Samba service and lack of vulnerability management.

Recommendations

- Enforce patch management
- Continuous vulnerability scanning
- Expand SOAR automation
- Improve asset visibility

11 Stakeholder Briefing (150 Words)

On August 18, 2025, our security team identified and contained a simulated cyberattack targeting a vulnerable system. The attack exploited an outdated service to gain unauthorized access. Our monitoring systems detected the activity within 15 minutes, and automated response mechanisms blocked the attacker within 30 minutes.

No business impact occurred due to rapid containment. Analysis revealed the root cause was missing software updates and insufficient vulnerability monitoring. Moving forward, we recommend strengthening patch management, increasing automation, and enhancing continuous monitoring. These improvements will reduce risk exposure and improve response efficiency.

Overall, the exercise confirmed that our SOC processes are effective and capable of responding to real-world threats.