Mini SOC for Law Enforcement

Incident Response, Chain of Custody, and Critical Infrastructure Protection

Version: Draft v1 | Date: 10/02/2025

# 1. Project Summary

This project simulates a Security Operations Center (SOC) environment tailored for law enforcement. It provides hands-on experience with SIEM dashboards, detection rules, incident playbooks, evidence handling (chain of custody), and monitoring of critical infrastructure systems relevant to public safety. The project bridges classroom learning and real-world SOC analyst workflows for police cybersecurity units.

# 2. Problem Statement / Use Case

Law enforcement agencies face cyber threats that target critical infrastructure (traffic systems, utilities, police networks). At the same time, digital evidence collected during cyber incidents must be handled with strict chain of custody procedures to maintain integrity in legal proceedings. Entry-level SOC analysts must be prepared to detect attacks, respond effectively, and preserve evidence integrity. This project builds a mini SOC to simulate end-to-end law enforcement-style cybersecurity operations.

# 3. Goals and Objectives

- Configure a working SIEM with live log ingestion from multiple sources  
- Build dashboards and detection rules for common attack scenarios  
- Develop 4–6 incident response playbooks with escalation to law enforcement units  
- Document at least 10 investigations with full chain of custody logs  
- Simulate monitoring of critical infrastructure systems (routers, utility servers, police portals)

# 4. Key Features / Functions

- SIEM Dashboards: Failed logins, suspicious processes, unusual traffic patterns  
- Detection Rules: Brute force, port scans, phishing attempts, malware execution  
- Playbooks: SOP-style workflows with law enforcement escalation  
- Case Management: Evidence logging with chain of custody and hash verification  
- Critical Infrastructure Monitoring: Dashboards for police, traffic, and utility systems  
- Reporting: Markdown/PDF reports suitable for portfolios and courtroom evidence

# 5. Tech Stack and Tools

Languages: Python, Bash  
Platforms: VirtualBox VMs (Ubuntu, Windows, Parrot OS, pfSense)  
SIEM: ELK Stack (preferred) or Splunk Free Edition  
Tools: Nmap, Hydra, Suricata IDS, Syslog/Filebeat/Winlogbeat  
Collaboration: Monday.com (case tracking), GitHub (documentation)  
Reporting: Markdown, PDF case reports with hash verification

# 6. Architecture / Workflow

1. Log Collection (Linux, Windows, Network)  
2. SIEM Ingestion (ELK/Splunk)  
3. Dashboards & Detection Rules  
4. Alerts → Analyst Investigation  
5. Case Management with Chain of Custody Logging  
6. Critical Infrastructure Monitoring  
7. Final Reports (courtroom-style)

# 7. Timeline / Weekly Milestones

Weeks 1–2: Lab setup (VMs, SIEM install, log forwarding)  
Weeks 3–5: Build dashboards, detection rules, simulate attacks  
Weeks 6–7: Draft playbooks, integrate escalation workflows  
Weeks 8–9: Add chain of custody procedures, document 4 core cases  
Weeks 10–11: Expand monitoring to critical infrastructure, complete 10+ cases  
Week 12: Final presentation and demo (SOC workflow + evidence handling)

# 8. Risks and Mitigation

- Complex SIEM setup: Use snapshots and community guides to save time  
- Evidence mishandling: Mitigate with strict chain of custody templates and hash checks  
- Time constraints: Use phased deliverables and track progress in Monday.com  
- Tool compatibility: Test multiple OS/log sources early

# 9. Evaluation Criteria

- Functional dashboards and detection rules triggering on simulated attacks  
- At least 4 tested and documented playbooks  
- Chain of custody logs for 5+ cases  
- 10+ documented case investigations with evidence integrity  
- Demonstrated monitoring of simulated critical infrastructure

# 10. Future Considerations

- Expand case management into enterprise tools like TheHive/RTIR  
- Add anomaly detection using Zeek and ML models  
- Integrate cloud log sources (AWS, Azure Sentinel)  
- Scale to larger law enforcement SOC workflows across multiple divisions