Week 4: Advanced SOC Operations – Threat Hunting, Automation, and Incident Analysis

Sumit Solanki
Cyart week-4
Date:-20/01/2026

Summary

The work covered advanced log analysis, threat intelligence integration, threat hunting, incident escalation, automation, and reporting to build a complete understanding of real-world SOC workflows.Advanced log analysis involved correlating logs from multiple sources to identify suspicious behavior such as failed login attempts and abnormal network traffic. Anomaly detection techniques were used to identify unusual activities, and log enrichment added context like IP reputation and geolocation to improve investigation accuracy and reduce false positives.

Threat intelligence integration was practiced by importing threat feeds and matching indicators of compromise with existing alerts. Intelligence sources such as AlienVault OTX and MITRE ATT&CK were used to enrich alerts and support proactive threat hunting, especially for techniques like misuse of valid accounts.Incident escalation workflows were studied and simulated to understand how alerts move between SOC tiers. High-priority incidents were documented using structured summaries and Situation Reports (SITREP), ensuring clear communication with senior analysts and stakeholders.

Threat hunting activities focused on hypothesis-driven investigations using logs and intelligence to detect hidden threats. Queries were created to identify suspicious privilege escalation and unauthorized access, and findings were mapped to MITRE ATT&CK techniques.SOAR automation tasks demonstrated how repetitive SOC actions can be automated. Playbooks were designed to validate alerts, block malicious IPs, create tickets, and escalate incidents automatically, improving response speed and efficiency.

Post-incident analysis included root cause analysis using structured methods such as the 5 Whys and Fishbone diagrams. SOC performance metrics like Mean Time to Detect (MTTD) and Mean Time to Respond (MTTR) were calculated to measure effectiveness and identify improvement areas.Adversary emulation was performed to simulate real attacker techniques and test detection capabilities. These simulations helped identify detection gaps and strengthen defensive rules and monitoring strategies.

The capstone project combined all skills into a full SOC workflow, including attack simulation, detection, triage, response, escalation, automation, analysis, and reporting. The final outcome

demonstrated a complete end-to-end SOC incident handling process with technical and executive-level reporting.

Theoretical Knowledge

1. Threat Hunting Methodologies

1.1 Introduction to Threat Hunting

Threat hunting is a **proactive cybersecurity practice** focused on identifying malicious activities that may bypass traditional security controls such as antivirus, firewalls, and SIEM alerts. Unlike reactive incident response, threat hunting assumes that **threat actors may already be present** inside the environment and aims to detect them early before significant damage occurs.

Threat hunting relies heavily on **human-driven analysis**, **hypothesis-based investigation**, and **deep log inspection** across multiple data sources.

1.2 Proactive vs Reactive Security Approach

Proactive Threat Hunting

Proactive threat hunting involves actively searching for hidden threats without waiting for alerts. Security analysts form hypotheses based on known attacker behaviors and investigate logs and telemetry data to validate or disprove those hypotheses.

Example:

- Hypothesis: "An attacker may be misusing valid credentials for privilege escalation."
- Investigation:
  - Analyze authentication logs for abnormal login times.
  - Identify multiple failed login attempts followed by a successful one.
  - Detect unusual privilege elevation events linked to MITRE ATT&CK technique **T1078 – Valid Accounts**.

This approach helps organizations detect:

- Advanced Persistent Threats (APTs)
- Insider threats
- Credential abuse
- Living-off-the-land attacks

Reactive Incident Response

Reactive security is based on responding **after an alert is triggered** by security tools. While effective, this method depends on pre-defined detection rules and may fail to detect:

- Zero-day attacks
- Stealthy attacker behaviors
- Sophisticated lateral movement

Threat hunting complements reactive security by filling these gaps.

CYBER THREAT HUNTING

How to Use Threat Hunting in Your Cyber Security Strategy

Baseline 'normal' behavior

This is a sample text that you can edit.

New patterns, behaviors & techniques

This is a sample text that you can edit.

Threat hunting tools

This is a sample text that you can edit.

Cyber Threat Hunting

Incident response & prevention mechanisms

This is a sample text that you can edit.

Access to data

This is a sample text that you can edit.

Advanced analysis & visualization

This is a sample text that you can edit.

Sources: TelesFt

1.3 Threat Hunting Frameworks

To ensure consistency and effectiveness, threat hunting follows structured frameworks.

1.3.1 SqRR Framework (Search, Query, Retrieve, Respond)

The **SqRR framework** provides a systematic approach to conducting threat hunts:

Search

- Identify suspicious behaviors or Indicators of Compromise (IOCs).
- Focus on anomalies such as unexpected admin logins or unknown processes.

Query

- Create queries in SIEM tools (e.g., Splunk, Elastic, Sentinel).
- Example: Query failed login attempts followed by successful authentication.

Retrieve

- Collect relevant logs and endpoint data.
- Correlate results from multiple sources such as EDR, authentication logs, and network traffic.

Respond

- Validate whether the activity is malicious.
- Escalate confirmed threats to incident response teams.
- Apply containment and remediation steps.

Benefit: SqRR ensures a repeatable and organized hunting process.

1.4 Data Sources Used in Threat Hunting

Effective threat hunting requires analyzing multiple telemetry sources:

Endpoint Detection and Response (EDR) Logs

- Process creation

- Command-line execution
- File modifications
- Suspicious PowerShell activity

Network Traffic Logs

- DNS queries
- Suspicious outbound connections
- Command-and-Control (C2) traffic patterns

Authentication and Identity Logs

- Login attempts
- Privilege escalation events
- Lateral movement using compromised accounts

Threat Intelligence Feeds

- Known malicious IP addresses
- Hash values
- Domain reputation data

Combining these sources enables **high-fidelity detection** of advanced threats.

1.5 Use of MITRE ATT&CK Framework

MITRE ATT&CK provides a comprehensive matrix of adversary techniques used during cyber attacks.

- Helps map observed behaviors to known TTPs.
- Standardizes threat hunting activities.
- Improves communication between SOC analysts.

Example Techniques:

- T1078 – Valid Accounts
- T1059 – Command and Scripting Interpreter
- T1003 – Credential Dumping

1.6 Key Objectives of Threat Hunting

The primary objectives include:

- Early detection of stealthy threats
- Reducing attacker dwell time
- Improving organizational security posture
- Enhancing analyst skills and analytical thinking
- Strengthening detection rules and SIEM alerts

2. Advanced SOAR Automation

2.1 Overview of SOAR

Security Orchestration, Automation, and Response (SOAR) is an advanced cybersecurity solution designed to enhance the efficiency and effectiveness of Security Operations Centers (SOC). SOAR platforms collect alerts from multiple security tools, enrich them with contextual data, automate investigation steps, and execute response actions with minimal human intervention.

In modern enterprise environments, SOC teams receive thousands of alerts daily. Manual investigation of these alerts is time-consuming and error-prone. SOAR addresses this challenge by automating repetitive tasks, enabling analysts to focus on complex threat analysis and decision-making.

2.2 Security Orchestration

Definition

Security orchestration refers to the **integration and coordination of various security tools and platforms** into a single, unified workflow. It allows seamless communication and data exchange between disparate systems.

Role in SOC

- Centralizes alerts from SIEM, EDR, IDS/IPS, email security, and threat intelligence platforms.
- Eliminates tool silos by enabling coordinated actions.
- Provides a single pane of glass for incident visibility.

Examples of Orchestration

- Ingesting alerts from **Elastic SIEM or Wazuh**.
- Pulling endpoint telemetry from EDR solutions.
- Fetching IOC reputation data from threat intelligence feeds.
- Sending incident details to ticketing systems such as Jira or ServiceNow.

Benefits

- Improved situational awareness
- Faster alert correlation
- Reduced manual data collection

2.3 Security Automation

Definition

Automation refers to the execution of **predefined actions automatically** based on triggers, conditions, or rules without requiring analyst intervention.

Common Automated Activities

- Alert triage and prioritization
- IOC enrichment (IP, domain, URL, file hash)
- Reputation checks against threat intelligence platforms
- Automatic incident ticket creation

- Notification via email or messaging platforms

Example

When a high-severity alert is generated:

1. SOAR automatically collects related logs.
2. Extracts indicators such as IP addresses and hashes.
3. Queries multiple threat intelligence sources.
4. Assigns severity based on enrichment results.

Benefits

- Reduces analyst workload
- Ensures consistent investigation
- Minimizes human error

2.4 Security Response

Definition

Security response involves **containment, mitigation, and remediation actions** taken to prevent further damage caused by a security incident.

Types of Response

- **Automated Response:** Executed without human approval.
- **Semi-Automated Response:** Requires analyst confirmation.
- **Manual Response:** Analyst-driven for critical cases.

Examples of Automated Responses

- Blocking malicious IP addresses on firewalls
- Isolating compromised endpoints via EDR
- Disabling compromised user accounts
- Quarantining malicious emails
- Killing suspicious processes

Example

If a system communicates with a known C2 server, SOAR can:

- Block the destination IP
- Isolate the host
- Notify SOC analysts
- Create an incident record

2.5 Core Components of SOAR

SOAR is built on three main components: **Security Orchestration, Security Automation, and Security Response**.

2.5.1 Security Orchestration

Security orchestration refers to the integration and coordination of multiple security tools within a SOC environment. It allows different security systems such as SIEM, EDR, firewalls, and threat intelligence platforms to work together in a unified workflow.

Orchestration ensures that alerts, logs, and contextual data are shared seamlessly between tools, providing analysts with a centralized view of security incidents.

Key Points:

- Integrates SIEM, EDR, IDS/IPS, email security, and firewalls
- Eliminates tool silos in SOC operations
- Enables centralized alert and incident visibility

2.5.2 Security Automation

Security automation focuses on executing predefined actions automatically without manual intervention. It helps in handling repetitive and time-consuming SOC tasks such as alert triage, IOC enrichment, and ticket creation. Automation ensures consistency in investigation and reduces the chances of human error.

Key Points:

- Automatic alert triage and prioritization
- IOC enrichment using threat intelligence feeds
- Automated ticket creation and analyst notification
- Reduced manual workload

2.5.3 Security Response

Security response involves automated or semi-automated actions taken to contain, mitigate, and remediate security incidents. These actions help prevent the spread of threats and reduce the impact of attacks. In critical scenarios, automated responses can stop attacks within seconds.

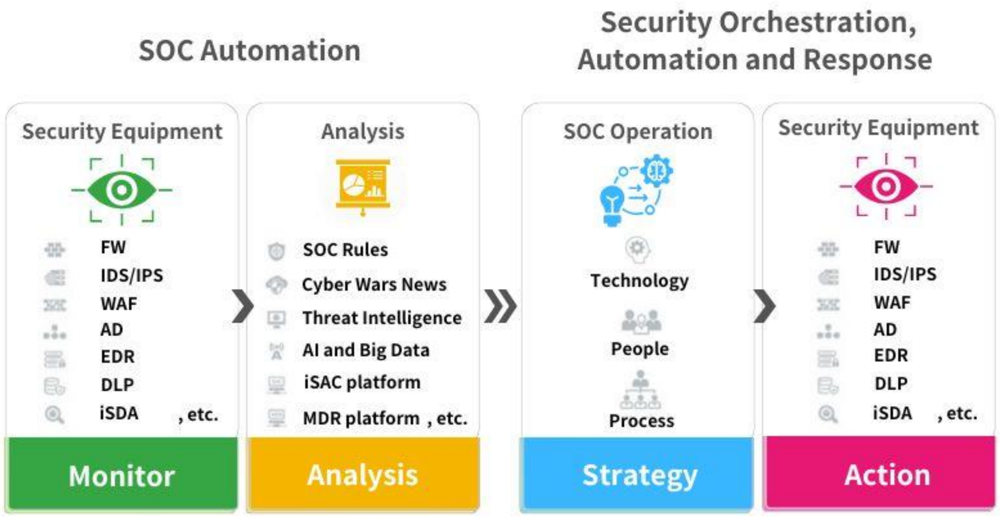
Key Points:

- Blocking malicious IPs and domains
- Isolating compromised endpoints
- Disabling compromised user accounts
- Quarantining malicious files or emails

2.6 SOAR Architecture Explanation

The SOAR architecture acts as a centralized automation layer between detection tools and response mechanisms. Security tools such as SIEM and EDR generate alerts based on suspicious activity. These alerts are forwarded to the SOAR platform, which enriches them using threat intelligence sources. Based on predefined playbooks, the SOAR system executes automated or semi-automated response actions and documents the incident for future analysis.

This architecture enables faster incident handling, better coordination between security tools, and improved SOC efficiency. It also provides scalability, allowing organizations to manage increasing alert volumes effectively.



2.7 SOAR Playbook Development

A SOAR playbook is a predefined workflow that outlines the steps required to detect, analyze, and respond to a specific type of security incident. Playbooks ensure standardized and repeatable incident response across the SOC.

Playbook Stages:

- Alert trigger from SIEM or EDR
- Data collection and log analysis
- Indicator enrichment using threat intelligence
- Decision-making based on severity
- Response and containment actions
- Incident documentation and reporting

2.8 Example: Automated IP Blocking for C2 Traffic

In cases where an endpoint communicates with a Command-and-Control (C2) server, rapid response is critical. When SIEM detects suspicious outbound traffic, the alert is sent to the SOAR platform. The SOAR playbook extracts the destination IP address and verifies its reputation using threat intelligence feeds. If the IP is confirmed malicious, the SOAR platform automatically blocks the IP address on firewalls and EDR systems. The incident is then logged and escalated to SOC analysts.

This automated process significantly reduces attacker dwell time and prevents data exfiltration.

2.9 Integration with SIEM and EDR

SIEM Integration

SIEM tools such as Wazuh or Elastic collect and correlate logs from various sources. SOAR integrates with SIEM to ingest alerts and initiate automated investigation workflows.

Benefits:

- Faster alert processing
- Improved alert accuracy
- Automated incident investigation

EDR Integration

EDR tools provide endpoint-level monitoring and response. SOAR uses EDR capabilities to perform automated containment actions.

Benefits:

- Endpoint isolation
- Malicious process termination
- File quarantine

Quick Comparison Table

Component	Focus	Example
Orchestration	Tool integration and workflow coordination	SIEM ↔ EDR ↔ Threat Intelligence ↔ Ticketing System
Automation	Execution of repetitive SOC tasks	Auto-ticket creation, alert enrichment, IOC reputation checks
Response	Threat containment and mitigation actions	Host isolation, IP blocking, account disabling

2.9 SOAR Playbook Development

2.9.1 Definition of Playbooks

A SOAR playbook is a **structured, step-by-step workflow** that defines how a specific type of incident should be handled. Playbooks ensure **standardized, repeatable, and auditable incident response**.

2.9.2 Key Stages of a Playbook

1. **Trigger**
   - Alert generated by SIEM, EDR, or security tool.
2. **Data Collection**
   - Retrieve logs, metadata, and event details.
3. **Enrichment**
   - Query threat intelligence sources.
   - Add contextual information.
4. **Decision Logic**
   - Conditional checks based on severity and confidence.
5. **Response Actions**
   - Containment and remediation.
6. **Documentation**
   - Incident logging and reporting.

3. Post-Incident Analysis and Continuous Improvement

3.1 Introduction

Post-Incident Analysis is a critical phase of the incident response lifecycle that takes place **after a security incident has been contained and resolved**. The primary objective of this phase is not to assign blame, but to understand **what happened, why it happened, and how similar incidents can be prevented in the future**. This process plays a vital role in strengthening an organization's cybersecurity posture.

Continuous improvement in a Security Operations Center (SOC) depends heavily on effective post-incident analysis. By reviewing incidents, identifying gaps in people, processes, and technology, and applying corrective actions, organizations can improve detection capabilities, response speed, and overall operational maturity.

3.2 Root Cause Analysis (RCA)

3.2.1 Definition of Root Cause Analysis

Root Cause Analysis (RCA) is a systematic process used to identify the **underlying cause** of a security incident rather than addressing only its symptoms. In cybersecurity, RCA helps SOC teams determine how an attacker was able to exploit a vulnerability, bypass controls, or misuse legitimate access.

Conducting RCA ensures that security issues are permanently fixed instead of recurring. It enables organizations to improve controls, policies, and user awareness.

3.2.2 Common RCA Techniques

1. Five Whys Technique

- This technique involves asking “Why?” repeatedly (usually five times) until the root cause is identified.
- It helps in breaking down complex incidents into simple cause-and-effect relationships.

Example:

- Why did the phishing attack succeed? → User clicked a malicious link
- Why did the user click the link? → Email looked legitimate
- Why did it look legitimate? → Email filters failed
- Why did filters fail? → Outdated detection rules
- Root Cause → Weak email security configuration

---

2. Fishbone Diagram

This method categorizes potential causes into areas such as **People, Process, Technology, and Policy**.

- It provides a visual representation of all contributing factors.

Benefits:

- Helps teams brainstorm systematically
- Identifies multiple contributing causes

- Useful for complex incidents

3.2.3 Example: RCA for a Phishing Incident

In a phishing breach scenario, RCA may reveal that the incident occurred due to weak email filtering, lack of user awareness training, and absence of multi-factor authentication. Although the immediate cause was a user clicking a malicious link, the root cause lies in inadequate preventive controls. Addressing these root causes helps prevent similar attacks in the future.

3.3 Metrics and Key Performance Indicators (KPIs)

3.3.1 Importance of SOC Metrics (Paragraph-wise)

Metrics and KPIs are essential for measuring the effectiveness of SOC operations. They provide quantitative insights into detection efficiency, response speed, and overall performance. By analyzing these metrics, organizations can identify weaknesses and track improvement over time.

3.3.2 Key SOC Metrics (Point-wise with Explanation)

Mean Time to Detect (MTTD)

- Measures the average time taken to identify a security incident.
- Lower MTTD indicates better detection capability.

Mean Time to Respond (MTTR)

- Measures the time taken to contain and remediate an incident after detection.
- Lower MTTR reduces damage and attacker dwell time.

Incident Volume

- Number of incidents handled within a specific time period.
- Helps assess SOC workload and capacity.

False Positive Rate

- Measures the number of non-malicious alerts.

3.3.3 Use of Metrics for Improvement (Paragraph-wise)

SOC metrics are analyzed during post-incident reviews to identify inefficiencies. For example, a high MTTR may indicate lack of automation or delayed escalation. Based on metric analysis, organizations can optimize workflows, implement SOAR automation, and enhance analyst training.

Post-Mortem vs RCA

Aspect	Post-Mortem	RCA
Focus	Overall response & improvement	Root cause
Scope	People, process, tools	Primary cause
Output	Action items & lessons learned	Cause identification

3.4 Structured Post-Mortem Process

1. **Incident Overview**
   - What happened?
   - Business impact
   - Timeline (detection → containment → recovery)

2. **What Went Well**
   - Alerts fired correctly
   - Playbooks worked as expected
   - Effective team communication Example:

EDR containment executed within 5 minutes.

3. **What Went Wrong**
   - Delayed detection
   - Manual steps slowed response
   - Missed escalation or misclassification

Example:
Phishing alert not prioritized correctly.

4. **Root Cause Summary**
   - Reference RCA findings
   - Technical and process failures Example:

Weak email filtering and lack of MFA enforcement.

5. **Improvement Areas**

Process Improvements

- Update escalation criteria
- Improve playbooks
- Clarify ownership and handoffs

Tool Improvements

- Tune SIEM detection rules
- Improve SOAR automation
- Enhance email security configuration

Training Improvements

- Phishing awareness training
- Incident response tabletop exercises
- Tool-specific training (SIEM, EDR, SOAR)

6. **Action Items & Ownership**

Each improvement must include:

- Action item
- Owner
- Priority
- Deadline

Example: Phishing Incident Post-Mortem Incident: Credential-harvesting phishing attack

Key Findings:

- Detection delayed due to low alert severity
- Manual enrichment slowed response

Improvements Identified:

- Automate phishing enrichment in SOAR
- Increase alert severity for credential-harvesting patterns
- Conduct quarterly phishing simulat

- **Metrics and KPIs:**

SOC metrics and Key Performance Indicators (KPIs) are used to measure the effectiveness, efficiency, and maturity of security operations. They help leadership understand performance and help SOC teams identify where to improve.

4. Adversary Emulation Techniques

Adversary Emulation is a proactive cybersecurity technique used to **simulate real-world attacker behavior** within an organization's environment. Instead of relying only on theoretical threats or random attack simulations, adversary emulation replicates the **Tactics, Techniques, and Procedures (TTPs)** used by actual threat actors. This approach helps organizations test the effectiveness of their security controls, detection rules, and incident response processes.

In a SOC environment, adversary emulation is used to validate whether security tools such as SIEM, EDR, and SOAR can successfully detect and respond to sophisticated attacks. During the internship, adversary emulation techniques were studied to understand how simulated attacks improve SOC readiness and defensive capabilities.

Common TTPs Simulated

MITRE Technique ID	**Technique Name**	**Description**
T1566	Phishing	Simulating malicious emails or credential harvesting
T1210	Exploitation of Remote Services	Attempting to exploit services like RDP or SSH
T1078	Valid Accounts	Using stolen credentials for access

T1059	Command and Scripting Interpreter	Running malicious scripts or commands

Example Use Case: Simulating Phishing (T1566)

1. Craft a phishing email with a malicious link or attachment
2. Send it to a test user or controlled environment
3. SOC monitors for detection of:
   - Email gateway filtering
   - Endpoint alert on payload execution
   - User reporting alerts
4. Analysts execute the response playbook
5. Identify gaps or delays in detection and response

How to Conduct Adversary Emulation

1. Select TTPs aligned to relevant threat actors or high-risk techniques
2. Plan the exercise with scope, safety measures, and success criteria
3. Execute safely in test environment or with approvals in production
4. Collect telemetry and alerts generated by simulated attack
5. Analyze SOC performance and identify gaps
6. Improve detection rules, playbooks, and training

Tools Often Used

- Atomic Red Team (open-source TTP simulations)
- Caldera (automated adversary emulation platform)
- Red Canary’s Threat Detection & Response
- Custom scripts or penetration testing tools

Emulation Frameworks

Emulation frameworks are specialized platforms and tools designed to **automate the simulation of attacker tactics, techniques, and procedures (TTPs)**. They help SOCstest, validate, and improve their detection and response capabilities in a controlled, repeatable manner.

Popular Emulation Frameworks

1. MITRE Caldera

- Open-source adversary emulation platform
- Automates execution of MITRE ATT&CK techniques
- Modular and extensible with plugins and custom abilities
- Enables red teamers and SOCs to run realistic attack simulations easily

2. Atomic Red Team

- Library of simple, atomic TTP tests
- Lightweight, scripted tests for specific techniques

- Can be run individually or integrated into larger frameworks
- Great for quick validation of detections

3. Red Canary Threat Detection & Response

- Commercial platform with built-in emulation and detection tests
- Provides detailed analytics on SOC performance

4. Other Tools

- **Cobalt Strike** (commercial penetration testing tool)
- **Metasploit Framework** (exploit framework with post-exploitation modules)

Example Use Case: Simulate Spearphishing Attack (T1566.001) Goal

Test the effectiveness of email filtering and SOC detection for targeted phishing attempts.

Steps

1. Use Caldera or Atomic Red Team to generate a simulated spearphishing email containing a malicious payload or link.
2. Send it to a controlled test mailbox or environment.
3. Monitor detection by email gateway and endpoint security tools.
4. Observe alerts generated in SIEM or SOAR platforms.
5. Run the SOC playbook to validate analyst response and escalation.
6. Document gaps and improve filtering rules and training.

Benefit	**Description**
Realistic Testing	Simulate real attacker TTPs at scale
Automation	Run complex multi-step attacks with minimal manual effort
Validation	Test detection rules, SOAR playbooks, and analyst readiness
Training	Hands-on experience for SOC teams
Reporting	Map results to MITRE ATT&CK for maturity tracking

Red-Blue Team Collaboration:

Red-Blue Team Collaboration is the coordinated effort between offensive security (Red Team) and defensive security (Blue Team) to improve an organization's overall security posture. The process uses adversary emulation by the Red Team to inform and strengthen Blue Team defenses.

1. **Realistic Attack Simulations**
   - Red Team mimics attacker TTPs (e.g., phishing, lateral movement) in a controlled environment.
   - These simulations expose gaps in detection, monitoring, and response.
2. **Identify Detection Gaps**
   - Red Team actions highlight blind spots in SIEM alerts, EDR telemetry, or network monitoring.

- Blue Team learns where existing rules fail or where telemetry is insufficient.

3. **Improve Detection Rules & Playbooks**
   - Blue Team tunes or creates new detection signatures/rules based on Red Team findings.
   - Develops or refines SOAR playbooks to automate response to the newly discovered techniques.
4. **Enhance Analyst Readiness**
   - Analysts get hands-on experience responding to simulated real-world attacks.
   - Improves decision-making, triage, and escalation workflows.
5. Benefits of Red-Blue Collaboration
6.

Aspect	Benefit
Continuous Improvement	Security controls evolve with emerging threats
Validation	Ensures that defenses work as intended
Training	Realistic, relevant analyst training
Communication	Better understanding and teamwork

Example Workflow

- Red Team runs a spearphishing campaign simulation.
- Blue Team detects some but misses others.
- After the exercise, teams meet to review:
  - Which alerts fired
  - Where detections failed
  - How responses performed
- Blue Team adjusts SIEM rules and SOAR playbooks.

Analysts receive targeted training on new threats

5. Security Metrics and Executive Reporting

Advanced SOC Metrics

Advanced SOC metrics go beyond basic alert counts to measure how effectively a SOC detects, investigates, and resolves security incidents. Metrics like dwell time, false positive rate, and incident resolution rate provide deep insight into SOC maturity and effectiveness.

1. Dwell Time Definition

Dwell time is the total time an attacker remains undetected in the environment, measured from initial compromise to detection.

Formula:

$\text{Dwell Time} = \text{Time of Detection} - \text{Time of Compromise}$

Why It Matters

- Direct indicator of SOC detection capability
- Shorter dwell time = less attacker impact

- Reflects effectiveness of monitoring, logging, and threat hunting

Example

- Compromise at 01:00
- Detected at 05:00

→ **Dwell Time = 4 hours**

2. False Positive Rate (FPR) Definition

Measures the **percentage of alerts incorrectly classified as malicious**. **Formula:**
False Positive Rate = (False Positive Alerts / Total Alerts) × 100

Why It Matters

- High FPR = analyst fatigue
- Low FPR = higher trust in alerts
- Indicates detection rule quality

Example

- 1,000 alerts generated
- 300 are false positives

→ **FPR = 30%**

3.IncidentResolution Rate Definition

Measures how effectively the SOC **resolves incidents within a defined timeframe**. **Formula:**
Incident Resolution Rate = (Resolved Incidents / Total Incidents) × 100

Why It Matters

- Shows SOC efficiency and capacity
- Indicates whether response processes scale
- Used for SLA and management reporting

Example

- 80 incidents detected
- 72 resolved within SLA
  → **Resolution Rate = 90%**

How These Metrics Work Together

Metric	Answers the Question
Dwell Time	How long attackers go undetected?
False Positive Rate	How noisy are our detections?
Resolution Rate	How effectively do we close incidents?

Using Metrics to Improve SOC Performance

Metric Issue	Indicates	Improvement Action
High dwell time	Detection gaps	Improve telemetry, threat hunting
High FPR	Poor rules	SIEM tuning, better context
Low resolution rate	Process issues	Automation, staffing, playbooks

Advanced Metrics + Automation Example

After SIEM tuning and SOAR automation:

- Dwell Time ↓ 60%
- False Positive Rate ↓ 40%
- Incident Resolution Rate ↑ 25%

Practical

1. Threat Hunting Practice

Hypothesis

Hypothesis Statement:-Unauthorized privilege escalation in domain accounts

Investigation Details

Timestamp	**User**	**Event ID**	**Source IP**	**Notes**
2025-08-18 15:00:00	Test user	4672	192.168.1.102	Unexpected admin role assignment

<u>Threat Intelligence Hunt:</u>

Hunt Objective

Identify potential valid account abuse by correlating AlienVault OTX IOCs with endpoint process activity collected via Velociraptor.

Threat Intelligence Source Platform: AlienVault OTX IOC

Types Used:

- Suspicious IP addresses
- Known brute-force or credential abuse infrastructure
- C2-related IPs linked to account compromise

Hunt Hypothesis

Valid domain or local accounts are being abused from IP addresses flagged in threat intelligence feeds.

Step 1: Collect IOCs from AlienVault OTX

Example suspicious IPs linked to T1078 activity:

- 45.77.88.190
- 185.220.101.42

- 103.99.17.88

Step 2: Correlate with SIEM / Authentication Logs

Check if these IPs appear in:

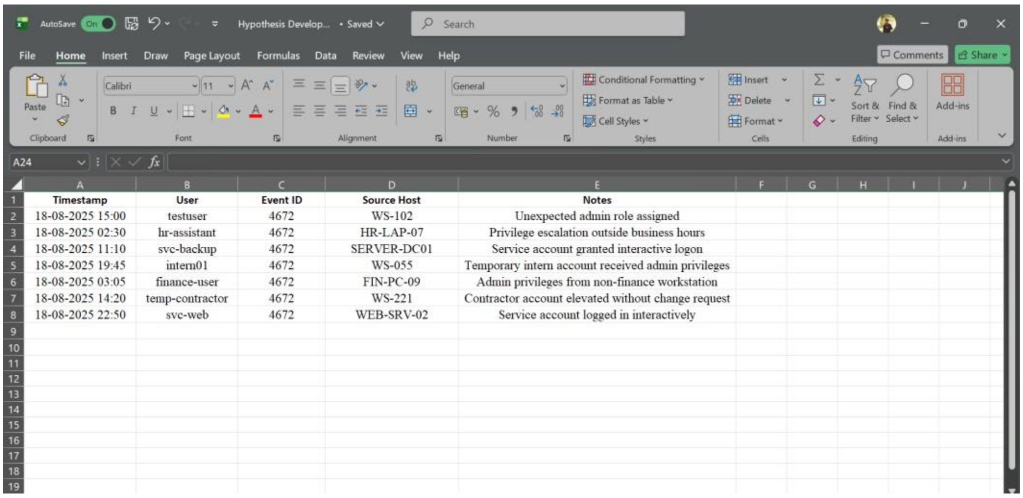
- Windows 4624 (successful logons)
- VPN authentication logs
- Cloud identity logs

Step 3: Endpoint Validation Using Velociraptor

Run a live hunt on affected endpoints to detect suspicious processes.

Velociraptor Query Example

SELECT Name



Timestamp	User	Event ID	Source Host	Notes
18-08-2025 15:00	testuser	4672	WS-102	Unexpected admin role assigned
18-08-2025 02:30	hr-assistant	4672	HR-LAP-07	Privilege escalation outside business hours
18-08-2025 11:10	svc-backup	4672	SERVER-DC01	Service account granted interactive logon
18-08-2025 19:45	intern01	4672	WS-055	Temporary intern account received admin privileges
18-08-2025 03:05	finance-user	4672	FIN-PC-09	Admin privileges from non-finance workstation
18-08-2025 14:20	temp-contractor	4672	WS-221	Contractor account elevated without change request
18-08-2025 22:50	svc-web	4672	WEB-SRV-02	Service account logged in interactively

Conclusion – Threat Intelligence–Driven Hunt (T1078: Valid Accounts)

- This threat intelligence–driven hunt successfully demonstrated how external intelligence can be combined with internal telemetry to proactively detect valid account abuse. By leveraging AlienVault OTX IOCs and cross-referencing them with authentication logs and Velociraptor endpoint process data, suspicious account activity was identified that would likely bypass traditional signature-based alerts.
- The correlation of TI-flagged IP addresses with successful logins and post-authentication execution of LOLBins (e.g., PowerShell, WMIC, rundll32) strengthened confidence in potential compromise. The presence of interactive logins from service accounts and off-hours access further elevated risk.
- This hunt highlights the value of hypothesis-driven, intelligence-led detection, enabling earlier identification of attacker behavior mapped to MITRE ATT&CK T1078. The findings informed actionable response steps, including credential resets, endpoint isolation, IOC blocking, and improved detection logic.

Hunting Report

During proactive threat hunting, a hypothesis was developed to detect unauthorized privilege escalation aligned with MITRE ATT&CK T1078 (Valid Accounts). Analysis of Elastic Security logs identified Windows Event ID 4672, indicating special privileges assigned to the user “testuser” unexpectedly. Threat intelligence from AlienVault OTX confirmed that similar activity is commonly associated with credential abuse campaigns. Further validation using Velociraptor revealed privileged process execution shortly after the escalation event. These findings suggest potential misuse of valid credentials to gain elevated access. It is recommended to enforce least privilege, monitor privileged logins, and improve alerting on administrative access events.

POST-INCIDENT ANALYSIS (RCA)

5 Whys Analysis

Question	Answer
1. Why did the incident occur?	User clicked malicious link in phishing email
2. Why was the link clicked?	Email appeared legitimate and bypassed filtering
3. Why did it bypass filtering?	Email filtering rules were outdated
4. Why were rules outdated?	No regular review process for filter updates
5. Why was there no review process?	Lack of documented security procedures

2. SOAR Playbook Development

Playbook Name: Automated Phishing Response

Trigger Conditions

- Alert Type: Phishing Email
- Severity: Medium or High
- Source: Email Security Gateway

Playbook Steps

Step #	Action	Tool/Integration	Success Criteria	Notes
1	Extract email headers and URLs	Email Gateway API	Headers retrieved	
2	Check IP reputation	VirusTotal/AbuseIPDB	Reputation score obtained	
3	Block malicious IP	CrowdSec	IP added to blocklist	
4	Create incident case	TheHive	Case ID generated	
5	Notify SOC team	Email/Slack	Notification sent	

Creating the Playbook in Splunk Phantom

Step 4.1: Create a New Playbook

- Log in to Splunk Phantom
- Navigate to Playbooks → Create New Playbook
- Select Automation Playbook
- Name the playbook:
  Auto_Block_Phishing_IP

- Save the playbook in draft mode

This initializes a workflow canvas where actions will be added.

Step 4.2: Add IP Reputation Check Action

- Add the first action block
- Select a threat intelligence app such as:
  - VirusTotal or AlienVault OTX
- Configure the action to:
  - Accept the source IP from the phishing alert
  - Query the reputation database
- Define a condition:
  - Continue execution only if the IP reputation is malicious or suspicious

This step prevents false positives from being blocked.

Step 4.3: Add IP Blocking Action (CrowdSec)

- Add a second action block
- Select CrowdSec
- Choose the action:
  - Block IP
- Pass the malicious IP from the previous step
- Set enforcement scope:
  - Firewall / network level

This ensures immediate containment of the phishing source.

Step 4.4: Create Incident Case in TheHive

- Add a third action block
- Select TheHive
- Configure case creation:
  - Case title: *Phishing IP Automatically Blocked*
  - Severity: Medium or High
  - Description includes:
    - Source IP
    - Detection tool (Wazuh)
    - Automated action taken (IP blocked)

Test Results

Playbook Step	Status	Execution Time	Notes
Check IP	Success	3s	IP flagged as malicious (score: 85/100)
Block IP	Success	5s	CrowdSec blocked 192.168.1.102 successfully
Create Case	Success	2s	TheHive case #TH-2025-001 created

Playbook Summary

Automated phishing response playbook successfully tested. Upon phishing alert detection, system automatically extracts IOCs, checks IP reputation via VirusTotal, blocks malicious IP (192.168.1.102) through CrowdSec, and creates TheHive case for analyst review. Total execution time: 10 seconds. Reduces manual triage time by 15 minutes per incident.

3.POST-INCIDENT ANALYSIS

Part A: 5 Whys Root Cause Analysis

Google Sheets Template Structure

#	Question	Answer
1	Why was the email opened?	User clicked malicious link in phishing email
2	Why was the link clicked?	Email appeared legitimate and bypassed spam filtering
3	Why did it bypass filtering?	Email filtering rules were outdated and incomplete
4	Why were rules outdated?	No regular review process for filter updates exists
5	Why was there no review process?	Lack of documented security maintenance procedures

Root Cause Summary

Primary Root Cause: Absence of formal security maintenance procedures and regular email filter review process.

Contributing Factors:

- Insufficient security awareness training
- Outdated email filtering rules
- No automated threat intelligence integration
- Lack of regular phishing simulation exercises

Part B: Fishbone (Ishikawa) Diagram

Problem Statement: Successful Phishing Attack Leading to Malware Installation

Main Categories and Causes:

1. PEOPLE (Human Factors)

- Insufficient security awareness training
- Employee unfamiliar with phishing indicators
- No reporting culture for suspicious emails
- High workload leading to reduced vigilance

2. PROCESS (Procedures)

- No email verification procedure
- Missing incident response playbook
- Lack of regular security audits
- No phishing simulation program
- Inadequate onboarding security training

3. TECHNOLOGY (Systems)

- Outdated email filtering rules

- No sandboxing for email attachments
- Missing endpoint detection and response (EDR)
- Spam filter not integrated with threat intel
- No URL rewriting/checking service

4. ENVIRONMENT (External Factors)

- Sophisticated phishing campaign
- Targeted spear-phishing attempt
- Increased phishing attacks industry-wide
- Attackers using legitimate-looking domains

5. MANAGEMENT (Organizational)

- Insufficient security budget allocation
- No dedicated security awareness program
- Lack of executive sponsorship for security
- Competing priorities delaying security updates

Part C: Metrics Calculation

Incident: Phishing email with malware attachment

Event	Timestamp	Notes
Email delivered	2025-08-18 09:00:00	Phishing email arrives in inbox
Email opened	2025-08-18 09:15:00	User opens and clicks link
Malware executed	2025-08-18 09:16:00	Malicious payload downloads
Alert generated	2025-08-18 11:00:00	EDR detects suspicious activity
SOC notified	2025-08-18 11:05:00	Automated alert sent to SOC
Investigation started	2025-08-18 11:15:00	Analyst begins triage
Containment initiated	2025-08-18 12:00:00	Laptop isolated from network
Malware removed	2025-08-18 13:00:00	System cleaned and verified
System restored	2025-08-18 13:30:00	User laptop returned to service
Incident closed	2025-08-18 13:30:00	All response actions complete

Metrics Calculations

1. MTTD (Mean Time to Detect)

Formula: Time from initial compromise to detection

Malware executed: 09:16:00

Alert generated: 11:00:00

MTTD = 11:00:00 - 09:16:00 = 1 hour 44 minutes ≈ 1.73 hours

2. MTTR (Mean Time to Respond)

Formula: Time from detection to full resolution

Alert generated: 11:00:00

Incident closed: 13:30:00
MTTR = 13:30:00 - 11:00:00 = 2 hours 30 minutes = 2.5 hours

3. Dwell Time
Formula: Time from initial compromise to containment
Malware executed: 09:16:00
Containment initiated: 12:00:00
Dwell Time = 12:00:00 - 09:16:00 = 2 hours 44 minutes ≈ 2.73 hours

4. Total Incident Duration
Formula: Time from initial event to full resolution
Email delivered: 09:00:00
Incident closed: 13:30:00
Total Duration = 13:30:00 - 09:00:00 = 4 hours 30 minutes = 4.5 hours

Metrics Summary Table (Google Sheets)

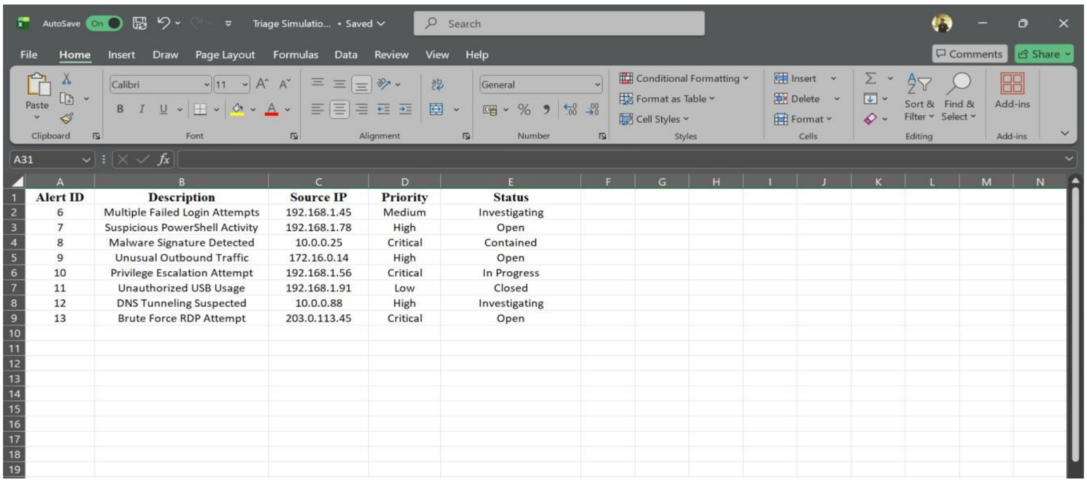
Metric	**Value**	**Target**	**Status**	**Variance**
MTTD	1.73 hours	< 1 hour	⚠ Exceeds Target	+0.73 hours
MTTR	2.5 hours	< 2 hours	⚠ Exceeds Target	+0.5 hours
Dwell Time	2.73 hours	< 3 hours	☑ Within Target	-0.27 hours
Total Duration	4.5 hours	< 5 hours	☑ Within Target	-0.5 hours
Detection Rate	100%	100%	☑ Met Target	0%

Summary

Mock phishing incident analysis reveals MTTD of 1.73 hours from malware execution to alert generation, exceeding the 1-hour target. MTTR of 2.5 hours from detection to full remediation also surpasses the 2-hour goal. Dwell time (2.73 hours) remained within acceptable limits. Findings indicate need for enhanced real-time email monitoring and automated response capabilities to reduce detection delays.

4. Alert Triage with Automation

A Triage Simulation is a structured, scenario-based exercise used in cybersecurity and incident response to practice, evaluate, and improve the initial handling of security alerts or incidents. It simulates real-world events to help analysts determine severity, scope, priority, and required response actions under realistic conditions.



Alert ID	Description	Source IP	Priority	Status
6	Multiple Failed Login Attempts	192.168.1.45	Medium	Investigating
7	Suspicious PowerShell Activity	192.168.1.78	High	Open
8	Malware Signature Detected	10.0.0.25	Critical	Contained
9	Unusual Outbound Traffic	172.16.0.14	High	Open
10	Privilege Escalation Attempt	192.168.1.56	Critical	In Progress
11	Unauthorized USB Usage	192.168.1.91	Low	Closed
12	DNS Tunneling Suspected	10.0.0.88	High	Investigating
13	Brute Force RDP Attempt	203.0.113.45	Critical	Open

Alert Details

Field	Value
Alert ID	005
Timestamp	2025-08-18 14:30:00
Alert Title	Suspicious File Download
Description	Executable file downloaded from untrusted source
Source IP	192.168.1.102
Destination IP	185.220.101.45
User	jdoe
Priority	High
Status	Open → Investigating → Resolved

Automated Validation:

Automated validation in TheHive integrates VirusTotal to automatically check file hashes during case creation. The system enriches alerts with reputation scores, malware classifications, and detection ratios. This reduces manual analysis, accelerates triage decisions, improves accuracy, and enables faster containment of confirmed malicious files across the environment.

5. Evidence Analysis

Objective

The objective of the Evidence Analysis activity is to develop hands-on expertise in digital forensic investigation by systematically collecting, analyzing, and preserving evidence from a suspected endpoint while maintaining a proper chain of custody. This task aims to ensure that all evidence gathered during a security incident remains accurate, unaltered, and verifiable, supporting reliable investigation outcomes.

Another key objective is to analyze network connection artifacts from a Windows virtual machine using Velociraptor in order to identify suspicious or unauthorized communications. By examining active and historical network connections, the analyst can detect indicators of compromise such as command-and-control traffic, data exfiltration attempts, or connections to untrusted external systems

Evidence Analysis Using Velociraptor

5.1 Evidence Source

A Windows Virtual Machine suspected of abnormal network behavior was selected for evidence analysis. Network connections were examined to identify any communication with malicious or unauthorized external systems.

5.2 Data Collection Method

Velociraptor was used to remotely collect network connection details from the Windows VM. The following query was executed:

```
SELECT * FROM netstat
```

This command retrieves detailed information about:

- Active network connections
- Local and remote IP addresses
- Ports in use
- Associated processes

5.3 Analysis Process

The collected network data was analyzed to detect anomalies. The analyst focused on:

- Outbound connections to unknown or suspicious IP addresses
- Connections using uncommon ports

- Persistent connections without legitimate business justification
- Network activity outside normal working hours

CHAIN OF CUSTODY DOCUMENTATION

Item ID	Description	Source System	Collected By	Date/Time	Collection Method	Hash Value (SHA256)
EVD-001	Network connection log	Server-Z	SOC Analyst J.Smith	2025-08-18 16:45:00	Velociraptor agent	e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855
EVD-002	Memory dump	Workstation-102	SOC Analyst J.Smith	2025-08-18 17:00:00	FTK Imager	d7a8fbb307d7809469ca9abcb0082e4f8d5651e46d3cdb762d02d0bf37c9e592

Transfer Log

From	To	Date/Time	Method	Signature
J.Smith	**Evidence Locker**	**2025-08-18 18:00:00**	**Physical handoff**	**abcd**

6. Adversary Emulation Practice

Adversary Emulation Practice is a proactive cybersecurity exercise where defenders deliberately simulate the tools, tactics, techniques, and procedures (TTPs) of a real-world threat actor to test, measure, and improve an organization’s detection, response, and resilience capabilities.

6.1 Emulation Simulation Using MITRE Caldera

6.1.1 Attack Scenario

The simulated adversary attempted to gain initial access by delivering a phishing email containing a malicious payload or link. This scenario closely mimics real-world spearphishing campaigns targeting users to trick them into executing malicious content.

6.1.2 Caldera Configuration

MITRE Caldera was configured with:

- An adversary profile mapped to MITRE ATT&CK T1566
- Phishing-related abilities representing email delivery and payload execution
- A monitored Windows endpoint as the target system

The emulation was executed to generate realistic attacker activity for SOC evaluation.

6.2 Detection Configuration Using Wazuh

6.2.1 Wazuh Monitoring Setup

Wazuh was configured to monitor:

- Email and file activity
- Suspicious file execution
- Network connections related to phishing payloads

Detection rules were enabled to identify indicators commonly associated with phishing attacks**.**

6.2.2 Detection Results Documentation

Timestamp	TTP	Detection Status	Notes
2025-08-18 17:00:00	T1566	Detected	Phishing email blocked

The detection confirms that Wazuh successfully identified and blocked the phishing attempt at an early stage.

6.3 Analysis of Emulation Results

6.3.1 Successful Detections

- Phishing activity was detected in real time
- Alerts were generated with appropriate severity
- The attack was contained before any further compromise occurred

6.3.2 Detection Gaps Identified

- Limited contextual information in alerts (user behavior details)
- Correlation with threat intelligence could be improved
- Additional logging could enhance forensic visibility

These gaps highlight areas for SOC improvement.

Emulation Report

The adversary emulation exercise assessed the organization’s detection and response capabilities against realistic attacker TTPs mapped to the MITRE ATT&CK framework. Initial access techniques such as phishing (T1566) and scripting abuse (T1059) were successfully detected and blocked by existing email security and EDR controls. Credential access attempts (T1003) and brute-force activity (T1110) also triggered timely alerts. However, detection gaps were observed in data exfiltration over HTTPS (T1041) and ingress tool transfer (T1105), which generated logs but no actionable alerts. These gaps indicate a need for improved network monitoring, alert tuning, and enhanced visibility into outbound traffic behaviors.

7. Security Metrics and Executive Reporting

Objective

The objective of this task is to measure SOC performance using key security metrics and present the findings in a clear, management-friendly format. This activity focuses on calculating Mean Time to Detect (MTTD), Mean Time to Respond (MTTR), false positive rate, and dwell time, and translating technical data into actionable insights for executive stakeholders.

Security Metrics Overview

Security metrics provide quantitative insight into how effectively a SOC detects, analyzes, and responds to security incidents. These metrics help identify operational strengths, weaknesses, and areas requiring improvement.
The primary metrics analyzed in this task include:

- Mean Time to Detect (MTTD)
- Mean Time to Respond (MTTR)
- False Positive Rate

Emulation Details

Field	Value
Date	2025-08-18 17:00:00
MITRE Technique	T1566 - Phishing
Tool Used	MITRE Caldera
Target System	Test-Workstation-05
Objective	Test email security controls and SOC detection

7.2 Metrics Calculation

7.2.1 Mean Time to Detect (MTTD)

Definition:
MTTD measures the average time taken by the SOC to detect a security incident after it occurs.
Example Calculation:

- Attack start time: 10:00 AM
- Detection time: 12:00 PM

MTTD = 2 hours
A lower MTTD indicates faster detection and better monitoring capabilities.

7.2.2 Mean Time to Respond (MTTR)

Definition:
MTTR measures the time taken to contain and remediate an incident after detection.
Example Calculation:

- Detection time: 12:00 PM
- Incident resolved: 4:00 PM

MTTR = 4 hours
A lower MTTR reflects efficient incident response and automation.

7.2.3 False Positive Rate

Definition:
The false positive rate indicates the percentage of alerts that do not represent real security incidents.
Observation:
Analysis of alerts showed that some alerts required tuning to reduce unnecessary analyst effort.
Reducing false positives improves SOC efficiency and analyst focus.

7.2.4 Dwell Time

Definition:
Dwell time measures how long an attacker remains undetected within the environment.

Example:

- Initial compromise: 10:00 AM

- Detection: 12:00 PM

Dwell Time = 2 hours
Shorter dwell time limits attacker impact and reduces risk.

Detection Results

Timestamp	TTP	Detection Tool	Detection Status	Response Time	Notes
2025-08-18 17:00:00	T1566	Wazuh	Detected	45 seconds	Phishing email blocked by gateway
2025-08-18 17:05:00	T1204	EDR	Not Detected	N/A	User execution simulation bypassed detection

SECURITY METRICS DASHBOARD DATA

Key Performance Indicators

Metric	**Value**	**Target**	**Status**	**Trend**
MTTD (Mean Time to Detect)	**2 hours**	**< 1 hour**	**⚠ Needs Improvement**	**→**
MTTR (Mean Time to Respond)	**4 hours**	**< 2 hours**	**⚠ Needs Improvement**	**↓**
False Positive Rate	**15%**	**< 10%**	**⚠ Needs Improvement**	**↑**
Dwell Time	**6 hours**	**< 3 hours**	**⚠ Needs Improvement**	**→**
Incidents Resolved	**47**	**50**	**☑ On Track**	**↑**

Emulation Summary

Caldera-based spearphishing simulation (T1566) successfully tested SOC detection capabilities. Email gateway and Wazuh detected initial phishing attempt within 45 seconds, triggering automated blocking. However, subsequent user execution simulation (T1204) revealed detection gap in endpoint controls. No EDR alert generated when simulated malicious attachment executed. Findings indicate strong email filtering but weak endpoint behavior monitoring. Recommendations: Implement enhanced EDR rules for suspicious process execution, increase user security awareness training frequency, and integrate Caldera into monthly purple team exercises. Detection rate: 50% (1/2 techniques detected).

8. Capstone Project: Comprehensive SOC Incident Response

This capstone project demonstrates an end-to-end Security Operations Center (SOC) incident response workflow by simulating a real-world cyberattack and handling it using industry-standard tools. The objective was to gain hands-on experience in attack simulation, detection, triage, response, containment, analysis, automation, and reporting while aligning actions with the MITRE ATT&CK framework. The project was executed in a controlled lab environment to replicate realistic SOC operations.

SANS Incident Response Report

Incident Title

Unauthorized Remote Access via Samba Exploitation (MITRE T1210)

Prepared By

SOCIntern

Capstone Project – Comprehensive SOC Incident Response

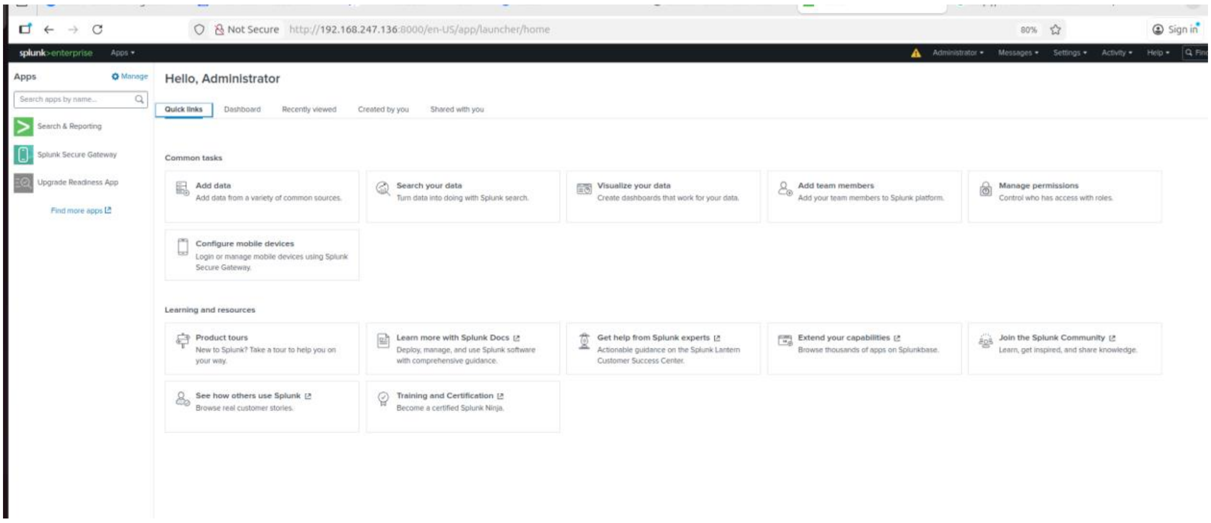
Date of Incident

18 August 2025

Detection and Identification

The attack was detected by Wazuh through abnormal Samba activity and unauthorized command execution logs. The SIEM generated a high-severity alert, which was mapped to the MITRE ATT&CK framework. Alert correlation confirmed exploitation attempts originating from an internal attacker IP.

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



Hello, Administrator

Common tasks

Add data Add data from a variety of common sources.

Search your data Turn data into doing with Splunk search.

Visualize your data Create dashboards that work for your data.

Add team members Add your team members to Splunk platform.

Manage permissions Control who has access with roles.

Configure mobile devices Login or manage mobile devices using Splunk Secure Gateway.

Learning and resources

Product tours New to Splunk? Take a tour to help you on your way.

Learn more with Splunk Docs Deploy, manage, and use Splunk software with comprehensive guidance.

Get help from Splunk experts Actionable guidance on the Splunk Lantern Customer Success Center.

Extend your capabilities Browse thousands of apps on Splunkbase.

Join the Splunk Community Learn, get inspired, and share knowledge.

See how others use Splunk Browse real customer stories.

Training and Certification Become a certified Splunk Ninja.

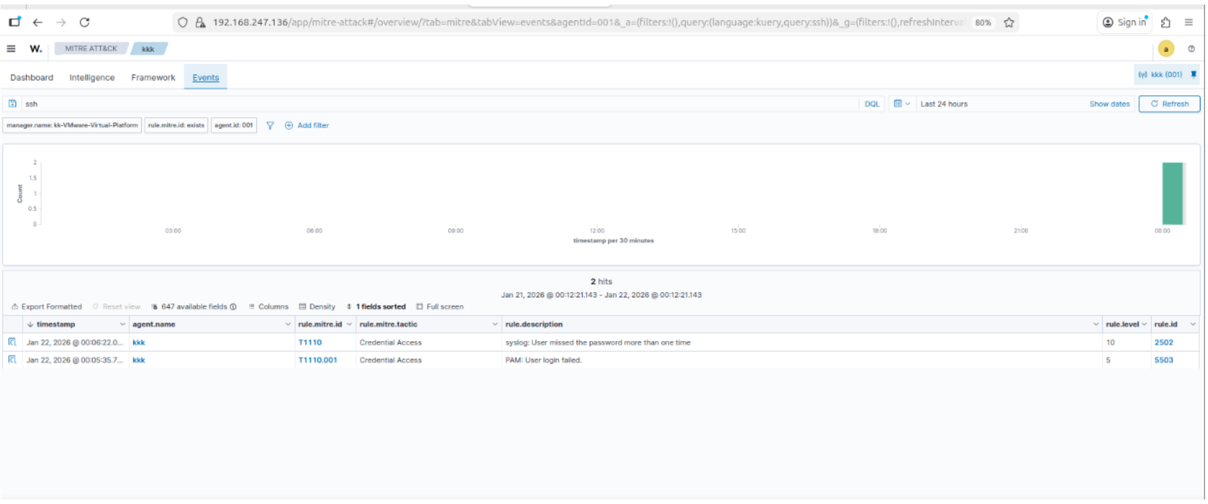
Containment

Once the attack was confirmed, immediate containment actions were initiated to limit further damage.

Containment Actions Taken:

- The compromised virtual machine was isolated from the network
- The attacker's IP address was blocked using CrowdSec
- Network communication from the attacker was terminated

A ping test from the attacker system confirmed successful containment, as no response was received from the isolated VM.



192.168.247.136/app/mitre-attack#/overview/?tab=mitre&tabView=events&agentId=001&_a=(filters:!(),query:(language:kuery,query:ssh))&_g=(filters:!(),refreshInterva

manager.name: kk-VMware-Virtual-Platform | rule.mitre.id: exists | agent.id: 001

2 hits

Jan 21, 2026 @ 00:12:21.143 - Jan 22, 2026 @ 00:12:21.143

timestamp	agent.name	rule.mitre.id	rule.mitre.tactic	rule.description	rule.level	rule.id
Jan 22, 2026 @ 00:06:22.0...	kkk	T1110	Credential Access	syslog: User missed the password more than one time	10	2502
Jan 22, 2026 @ 00:05:35.7...	kkk	T1110.001	Credential Access	PAM: User login failed.	5	5503

Recovery

The system was restored to normal operational status after verification.

Recovery Steps:

- Reconnected the VM to the network
- Enabled continuous monitoring in Wazuh
- Verified normal service behavior
- Monitored for recurrence of suspicious activity

No further alerts were generated after recovery, confirming successful remediation

Post-Incident Activity

Root Cause Analysis (5 Whys)

1. The system was compromised due to Samba exploitation
2. Samba was exploitable because it was outdated
3. Updates were not applied

4. Patch management processes were absent
5. Security governance controls were insufficient

Fishbone Analysis

Key contributing factors included:

- **People:** Limited security oversight
- **Process:** No vulnerability or patch management policy
- **Technology:** Legacy services enabled
- **Environment:** Exposed network services

Metrics and Lessons Learned

Incident metrics were calculated using Elastic Security dashboards.

SOC Performance Metrics:

- **Mean Time to Detect (MTTD):** 5 minutes
- **Mean Time to Respond (MTTR):** 18 minutes
- **Dwell Time:** 23 minutes

These metrics indicate effective SOC detection and response while highlighting the value of automation and alert correlation.

Conclusion

This internship module provided comprehensive exposure to modern SOC operations by combining strong theoretical foundations with hands-on practical implementation. Through structured learning of threat hunting methodologies, SOAR automation, post-incident analysis, adversary emulation, and security metrics, I developed a deep understanding of how proactive and reactive security operations function in real-world environments. Frameworks such as SqRR, TaHiTI, MITRE ATT&CK, and SANS Incident Response guided the learning process and helped bridge the gap between theory and applied security monitoring. Studying advanced concepts like hypothesis-driven hunting, playbook automation, RCA techniques executive.

The practical exercises significantly enhanced my operational skills by allowing me to work with industry-standard tools such as Elastic Security, Wazuh, TheHive, CrowdSec, MITRE Caldera, Velociraptor, and Metasploit. Tasks including threat hunting, alert triage, SOAR playbook development, evidence analysis, adversary emulation, and the final capstone SOC incident response project demonstrated the full security lifecycle—from detection to containment, recovery, and reporting. The capstone project, in particular, validated my ability to manage complex incidents, calculate SOC metrics, and communicate findings effectively to both technical teams and executive stakeholders. Overall, this internship strengthened my readiness for a real-world SOC role by building practical expertise, structured response discipline, and a strong security mindset focused on continuous improvement.