

# Log Analysis with SIEM

This lab, provided by TryHackMe, demonstrates some of my skills in Splunk and SPL (Search Processing Language) by investigating alerts and identifying suspicious activities across different systems.

## Windows Logs Investigation

You are an SOC Level 1 Analyst on shift and have received an alert indicating a suspicious network connection using port 5678 on the WIN-105 host. Your task is to conduct an investigation and determine whether this activity is suspicious.

Q1: Which IP address was the connection established with?

To identify the IP address, I filtered Windows Sysmon logs using EventCode 3 and the specific destination port 5678. The search revealed one event showing a network connection from WIN-105 to an external IP.

Command:

```
index="task4" EventCode=3 ComputerName=WIN-105 DestinationPort=5678
```

Answer: 10.10.114.80

Q2: Which process initiated this suspicious connection?

From the same filtered event, I reviewed the Image field which identified the process that initiated the connection.

Answer: SharePoint.exe

Q3: What is the MD5 hash of the malicious process from the previous question?

I expanded the search to include ProcessId and MD5 hash fields using a regex extraction command.

Command:

```
index=task4 (EventCode=3 OR EventCode=1) ComputerName=WIN-105 | transaction ProcessGuid maxspan=5m | search EventCode=3 DestinationPort=5678 | rex field=Hashes "(?i)MD5=(?[A-Fa-f0-9]{32})" | table _time ComputerName Image SourceIp DestinationIp DestinationPort Protocol ProcessId ProcessGuid md5
```

Answer: 770D14FF41A2F09730B415506249E7D1

Q4: What is the name of the scheduled task that was created on the system?

I filtered EventCode 1 and searched for 'schtasks.exe' within command lines to detect scheduled task creation attempts.

Command:

```
index=task4 ComputerName=WIN-105 EventCode=1 (Image="*\schtasks.exe" OR  
CommandLine="*schtasks*") | table _time ComputerName User Image CommandLine  
ParentImage ProcessGuid
```

Answer: Office365 Install

## Linux Logs Investigation

You are an SOC Level 1 Analyst on shift and have received an alert indicating possible persistence through the creation of a new remote-ssh user on an Ubuntu server.  
Your task is to dive into the logs and determine exactly what happened on the system.

Q1: What was the timestamp of the remote-ssh account creation?

I searched authentication logs for 'su' activity and sorted events by time to find when a new SSH session for the root user was initiated.

Command:

```
index=task5 source=auth.log *su* | sort + _time
```

Answer: 2025-08-12 09:52:57

Q2: Which user successfully escalated their privileges to root prior to the action from the first question?

Reviewing the previous event details revealed the user 'jack-brown' as the one who executed the sudo command leading to privilege escalation.

Answer: jack-brown

Q3: From which IP address did the user from the previous question successfully log in to the system?

By checking the SSHD authentication logs, I confirmed that the user connected from IP 10.14.94.82.

Answer: 10.14.94.82

Q4: How many failed login attempts occurred prior to this successful login?

I searched for 'Failed password for jack-brown' in the logs and identified four failed attempts before the successful authentication.

Answer: 4

Q5: Which port is the persistence mechanism configured to connect to?

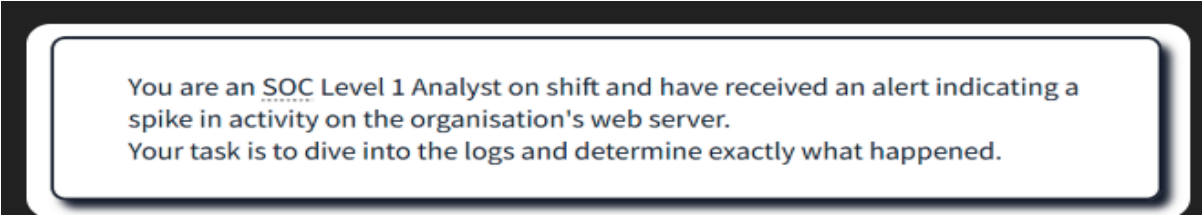
I analyzed cron job activity and found that the persistence mechanism was configured to connect to port 7654.

Command:

```
index=task5 sourcetype=syslog process=CRON
```

Answer: 7654

## Web Application Logs Investigation



You are an SOC Level 1 Analyst on shift and have received an alert indicating a spike in activity on the organisation's web server.  
Your task is to dive into the logs and determine exactly what happened.

Q1: Which URI path had the highest number of requests?

I analyzed the web server logs for POST requests to identify which URI was most frequently accessed. The results showed that '/wp-login.php' had the highest number of requests.

Command:

```
index=task6 method=POST uri_path="*/wp-login.php*" | bin _time span=5m |  
stats values(referer_domain) as referer_domain values(status) as status  
values(useragent) as UserAgent values(uri_path) as uri_path count by clientip  
_time | where count > 25 | table referer_domain clientip UserAgent uri_path  
count status
```

Answer: /wp-login.php

Q2: Which IP address was the source of the activity?

The same query also revealed that the source IP responsible for the excessive POST requests was 10.10.243.134.

Answer: 10.10.243.134

Q3: How can this activity be classified?

Given the repeated POST requests to the login page, this activity is clearly identified as a brute-force attack.

Answer: Brute Force Attack

Q4: Which tool did the threat actor use?

By analyzing the 'UserAgent' field, it became evident that the attacker used WPScan v3.8.28, a popular WordPress security scanner tool often used to automate brute-force attempts.

Answer: WPScan