Splunk Lab Setup Documentation - Phase 1

Introduction

Objective:

The objective of Phase 1 was to set up a Security Operations Center (SOC) lab, simulate basic cyberattacks, configure log sources from different operating systems, and detect real-world threat scenarios using a SIEM solution. In this project, Splunk Enterprise was used on a Windows host and Splunk Universal Forwarder on a Linux victim machine.

What is a SIEM and Why it Matters?

What is SIEM?

A Security Information and Event Management (SIEM) system collects, aggregates, normalizes, and analyzes logs from various sources to provide real-time alerts, correlation, and dashboards for detecting cyber threats.

Key Functions of SIEM:

- Log Management
- Event Correlation
- Security Monitoring
- Incident Detection and Response

Examples of SIEM Tools:

1. Splunk

2. IBM QRadar

3. ArcSight

4. Elastic SIEM

Why SIEM is Important:

A SIEM is crucial for detecting early-stage cyberattacks such as brute-force login attempts, unauthorized access, and lateral movement. It enables centralized log collection and real-time alerting.

Lab Setup Overview

This document details the setup of a Splunk lab environment for security monitoring and threat detection.
The architecture includes a Linux virtual machine with a Splunk Universal Forwarder and a Windows host running Splunk Enterprise. The environment is designed to simulate and detect common attack scenarios such as **brute-force attacks**, **lateral movement**, **log tampering**, and **user account creation**.

Architecture

- **Windows Host Machine**:
  - Operating System: Windows 10/11
  - Software: Splunk Enterprise

- **Linux Virtual Machine**:
  - Operating System: Kali Linux/Ubuntu
  - Software: Splunk Universal Forwarder

- **Communication**: Linux VM forwards logs to Splunk Enterprise on Windows via TCP port 9997.

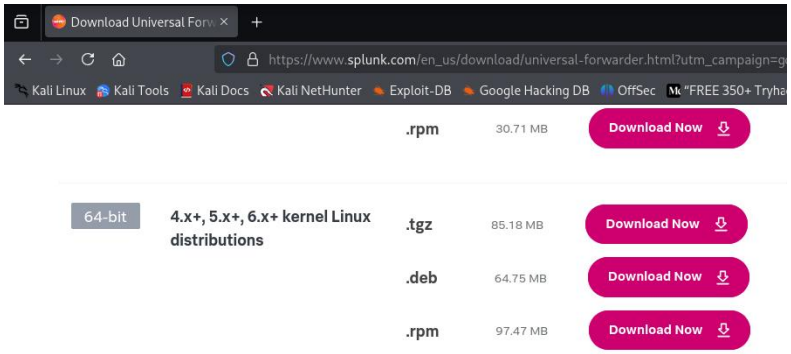
Tools Used

- Splunk Enterprise
- Splunk Universal Forwarder
- Syslog (/var/log/syslog)
- Auth logs (/var/log/auth.log)
- Audit logs (/var/log/auditd/audit.log)
- Custom attack simulation tools (ex : Hydra)

Splunk Forwarder Configuration

1. Install Splunk Forwarder on Linux VM:

- Download and extract Splunk Forwarder , according to you machine type.




```
┌──(root㉿kali)-[/home/kali/Downloads]
└─# dpkg -i splunkforwarder-9.4.2-e9664af3d956-linux-amd64.deb
Selecting previously unselected package splunkforwarder.
(Reading database ... 95%
```

2. Start Splunk Forwarder:

```
┌──(root㉿kali)-[/]
└─# /opt/splunkforwarder/bin/splunk start --accept-license
Warning: Attempting to revert the SPLUNK_HOME ownership
Warning: Executing "chown -R splunkfwd:splunkfwd /opt/splunkforwarder"

This appears to be your first time running this version of Splunk.

Splunk software must create an administrator account during startup. Ot
Create credentials for the administrator account.
Characters do not appear on the screen when you type in credentials.
```

sudo /opt/splunkforwarder/bin/splunk start --accept-license

3. Configure Splunk Forward Server:

This will forward our defined logs in inputs.conf to splunk enterprise server hosted on windows machine.

```
┌──(root㉿kali)-[/opt/splunkforwarder/bin]
└─# ./splunk add forward-server 192.168.1.34:9997
Warning: Attempting to revert the SPLUNK_HOME ownership
Warning: Executing "chown -R splunkfwd:splunkfwd /opt/splunkforwarder"
Splunk username: kali
Password:
Added forwarding to: 192.168.1.34:9997.
```

sudo /opt/splunkforwarder/bin/splunk add forward-server <Windows_IP>:9997

Note : This step , might be done again , if machine restarts with new IP_address .

4. Monitor Logs:

This step commands forwarder to forward which specific logs to splunk server.
Add the following to /opt/splunkforwarder/etc/system/local/inputs.conf:

```
┌──(root㉿kali)-[/opt/splunkforwarder/etc/system/local]
└─# cat input.conf
monitor:///var/log/syslog]
disabled = false
index = linux_logs
sourcetype = syslog

[monitor:///var/log/auth.log]
disabled = false
index = linux_logs
sourcetype = linux_secure

[monitor:///var/log/sysmon/sysmon.log]
disabled = false
index = linux_logs
sourcetype = sysmon_linux
```

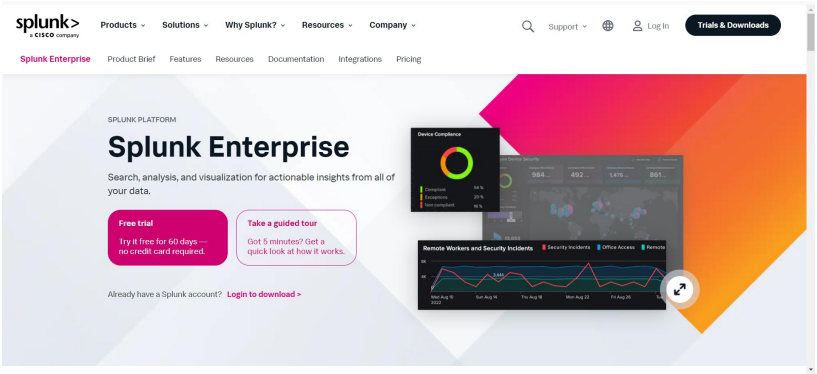
[monitor:///var/log/syslog]
disabled = false
index = linux_logs
sourcetype = syslog

[monitor:///var/log/auth.log]
disabled = false
index = linux_logs
sourcetype = linux_secure

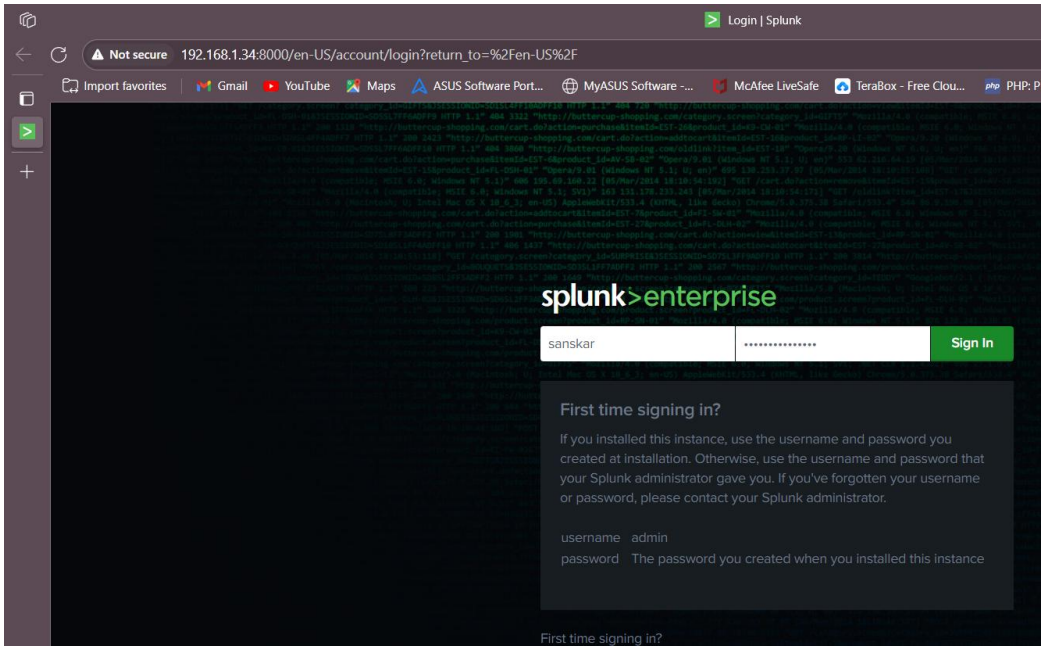
[monitor:///var/log/sysmon.log]
disabled = false
index = linux_logs
sourcetype = sysmon_linux

Splunk Enterprise Configuration

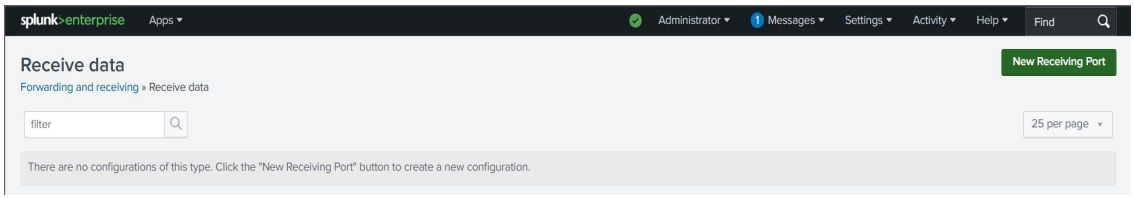
1. **Download Splunk Enterprise for windows from official site :**

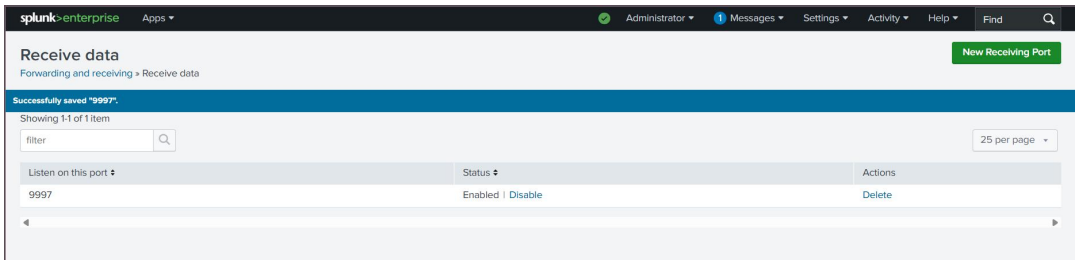


2. **Set username and password to login on splunk server later.**



3. **Enable Receiving on Port 9997:**

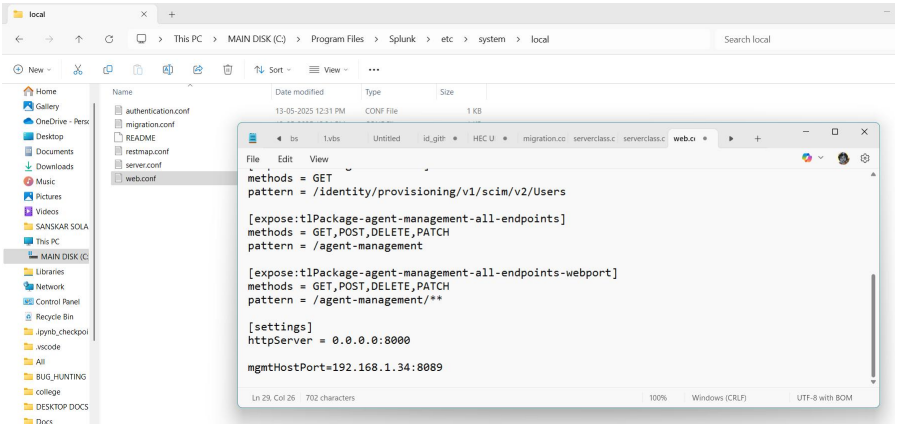




Settings > Forwarding and Receiving > Configure Receiving > new receiving port > Port 9997

3. Configure web.conf :

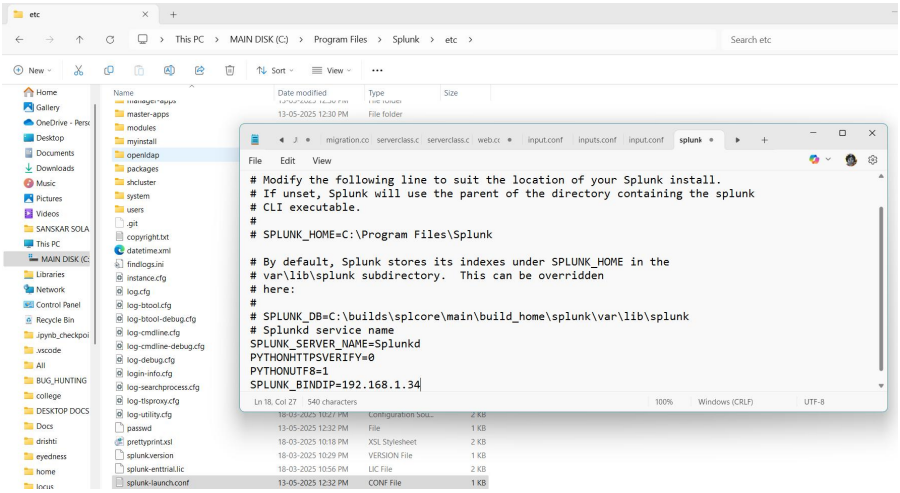
To allow access to splunk forwarder in linux via actual IP of windows where splunk server is hosted , edit C:\Program Files\Splunk\etc\system\local\web.conf:



[settings]
enableSplunkWebSSL = false
httpport = 8000
mgmtHostPort = 192.168.1.34:8089

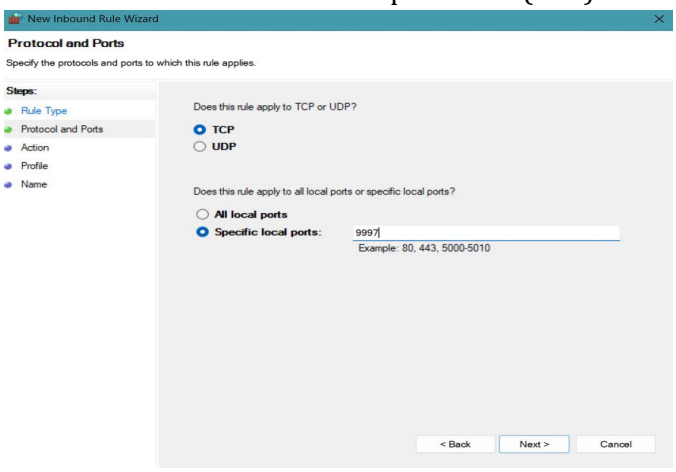
4. splunk-launch.conf :

This is necessary , if we are changing IP in above step , to start splunk server with that actual IP.
Configure environmental variables if needed in splunk-launch.conf , i.e , add bind_ip

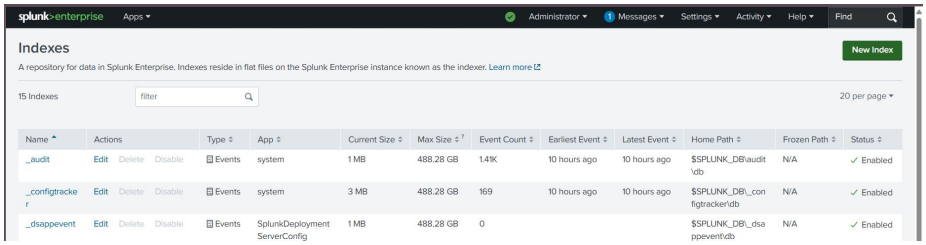


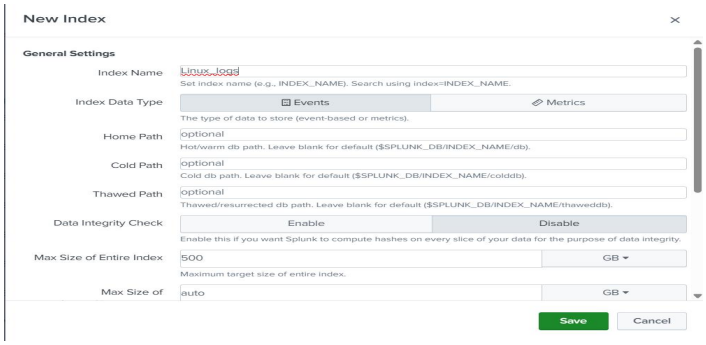
5. Firewall Configuration(this is optional):

Allow inbound connections on port 9997 (TCP) in Windows Defender Firewall.

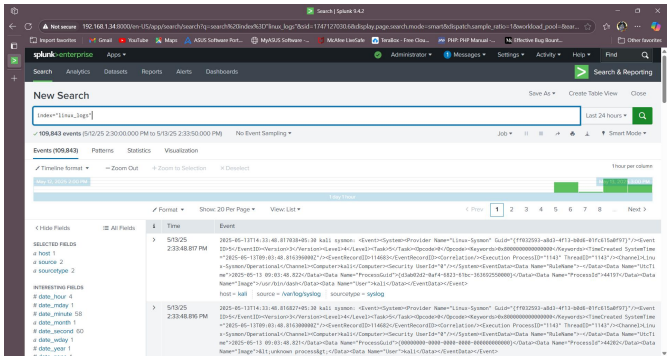


6. Create new index (namespace or data repository where forwarded logs are collected and analyzed) in splunk enterprise , with name as linux_logs.





7. Now restart splunk enterprise and splunk forwarder.



Detection Use Case 1 - Brute Force Login

Simulation:

- Used Hydra on the Linux victim to simulate SSH brute-force attempts.

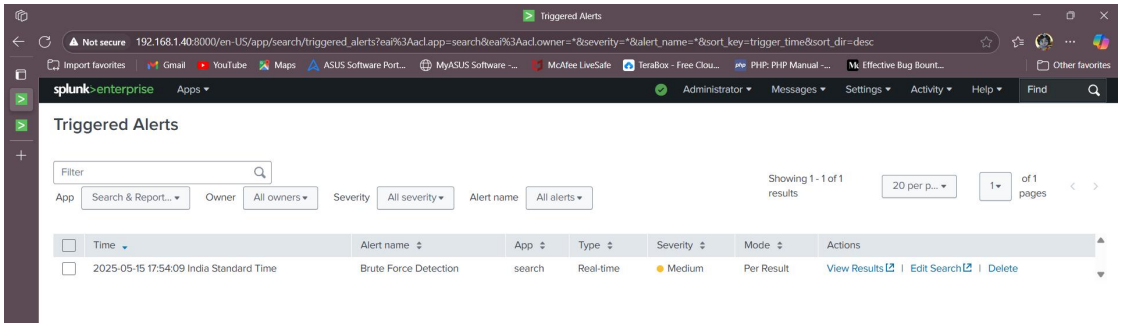
```
┌──(root㉿kali)-[~]
└─# hydra -l kali -P pass.txt ssh://192.168.1.6
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military
this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-05-13 09:06:38
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommende
[DATA] max 12 tasks per 1 server, overall 12 tasks, 12 login tries (l:1/p:12), ~1 try
[DATA] attacking ssh://192.168.1.6:22/
[22][ssh] host: 192.168.1.6   login: kali   password: kali
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-05-13 09:06:42
```

Detection:

- Splunk query searched for multiple "Failed password" events within a short time frame from the same IP.

- Triggered alert for potential brute-force login.



Detection Use Case 2 - After-Hours Login

Definition:

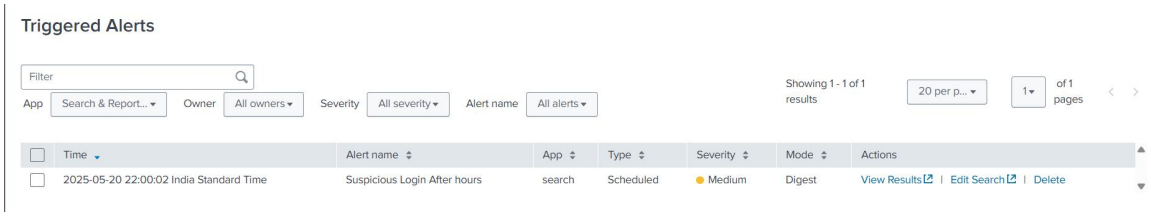
- Business hours defined as 9:00 AM to 7:00 PM.

```
index="linux_logs" sourcetype="auth"
("Accepted password for kali" OR "Accepted publickey for kali" OR
 "session opened for user root(uid=0) by kali(uid=1000)" OR
 "COMMAND=")
| eval hour_12=strftime(_time, "%I"), ampm=strftime(_time, "%p"), time_12hr=strftime(_time, "%I:%M:%S %p")
| eval hour_12_num=tonumber(hour_12)
| eval event_type=case(
    like(_raw, "%Accepted password for kali%") OR like(_raw, "%Accepted publickey for kali%"), "login",
    like(_raw, "%session opened for user root(uid=0) by kali(uid=1000)%"), "root_session",
    like(_raw, "%COMMAND=%") AND like(_raw, "%kali%"), "command_as_root",
    true(), "other"
)
| where event_type IN ("login", "root_session", "command_as_root")
    AND (
        (ampm="AM" AND hour_12_num < 9) OR
        (ampm="PM" AND hour_12_num > 7 AND hour_12_num < 12)
    )
| sort _time
```

Detection Logic:

- Detected SSH login events (e.g., "Accepted password") occurring outside business hours.

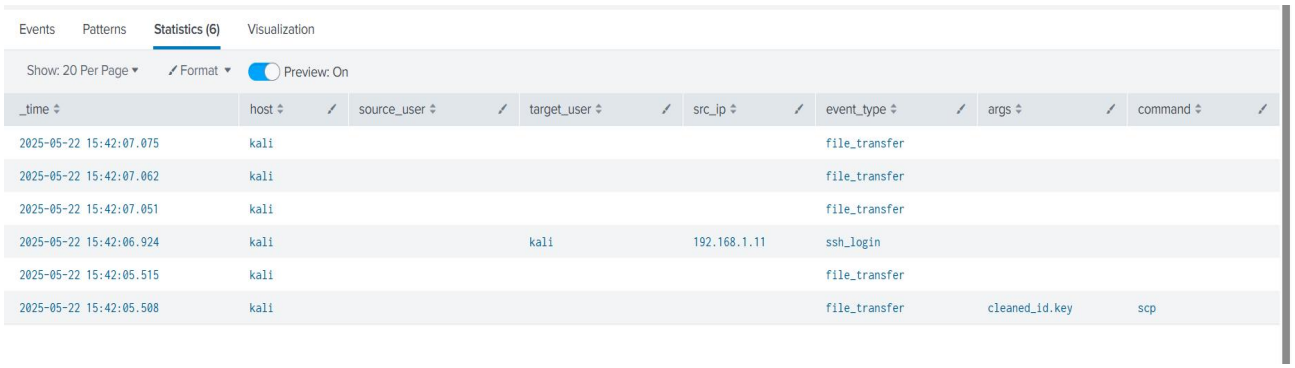
- Alerted for potential suspicious access.



Detection Use Case 3 - SSH Lateral Movement

Simulation:

- SSH and SCP/SFTP usage from internal IP addresses.



Events Patterns Statistics (6) Visualization

Show: 20 Per Page ▾ Format ▾ Preview: On

_time	host	source_user	target_user	src_ip	event_type	args	command
2025-05-22 15:42:07.075	kali				file_transfer		
2025-05-22 15:42:07.062	kali				file_transfer		
2025-05-22 15:42:07.051	kali				file_transfer		
2025-05-22 15:42:06.924	kali		kali	192.168.1.11	ssh_login		
2025-05-22 15:42:05.515	kali				file_transfer		
2025-05-22 15:42:05.508	kali				file_transfer	cleaned_id.key	scp

Detection:

- Splunk parsed "Accepted password" and "scp"/"sftp" logs.

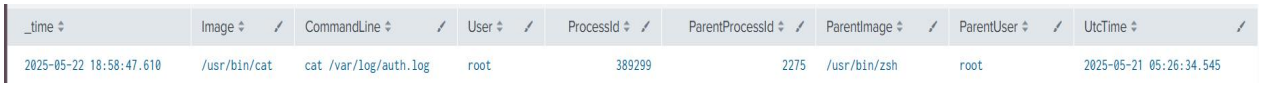
- Monitored for user logins and file transfers between internal systems.

- Mapped to MITRE ATT&CK T1021.004 (SSH).

Detection Use Case 4 - Log Tampering

Simulation:

- Commands like cat /var/log/syslog , or clearing log files with > /var/log/auth.log.



_time	Image	CommandLine	User	ProcessId	ParentProcessId	ParentImage	ParentUser	UtcTime
2025-05-22 18:58:47.610	/usr/bin/cat	cat /var/log/auth.log	root	389299	2275	/usr/bin/zsh	root	2025-05-21 05:26:34.545

Detection:

- Splunk monitored for service stops and file size drops.

- Alerts triggered on suspicious commands or cleared logs.

2025-05-15 19:15:17 India Standard Time Log tampering search Real-time Medium Digest View Results | Edit Search | Delete