Incident Detection and Response Project

Abstract

This project simulates a real-world Security Operations Center (SOC) workflow, demonstrating the crucial phases of threat detection, correlation, and automated response (SOAR). We established a monitoring pipeline using Splunk to analyse authentication logs from a Linux server under an emulated Hydra attack, triggering an immediate defensive action via the Uncomplicated Firewall (UFW).

Technical Environment & Attack Details

The project utilized three virtual machines (VMs) connected on an internal network:

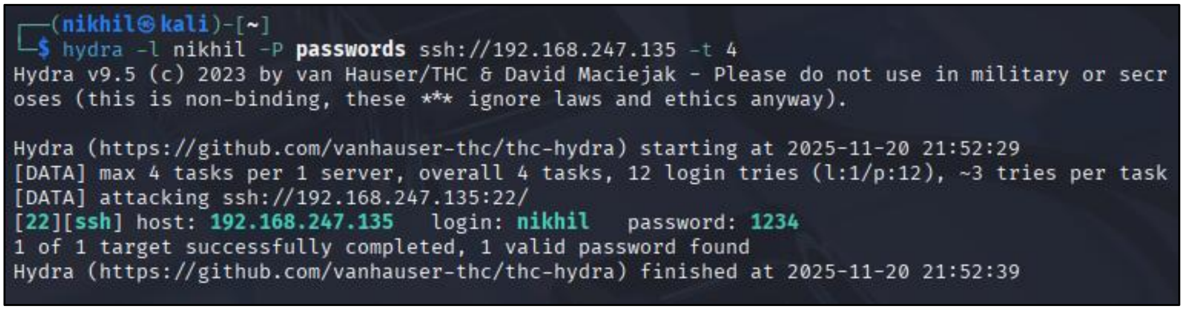
Component	Role	Operating System	Purpose
Attacker	Threat Source	Kali Linux	Simulated SSH Brute Force attempts.
Victim	Log Source & Enforcement	Ubuntu Linux	Hosted the SSH service and UFW firewall.
SIEM	Detection Engine	Ubuntu Linux (Splunk)	Ingested logs, ran correlation searches, and triggered the response.

Prerequisites

For the SSH service to be monitored and attacked, the '*openssh-server*' package must be installed and running on the Ubuntu Victim machine.

Attack Tool: Hydra

The attack was performed using the multi-protocol fast logon cracker, Hydra, specifically targeting the SSH service. The goal was to validate the detection system's ability to identify and respond to parallel, dictionary-based password guessing attempts.



```
┌──(nikhil㉿kali)-[~]
└─$ hydra -l nikhil -P passwords ssh://192.168.247.135 -t 4
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secr
oses (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-11-20 21:52:29
[DATA] max 4 tasks per 1 server, overall 4 tasks, 12 login tries (l:1/p:12), ~3 tries per task
[DATA] attacking ssh://192.168.247.135:22/
[22][ssh] host: 192.168.247.135   login: nikhil   password: 1234
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-11-20 21:52:39
```

Log Ingestion Pipeline (Splunk Universal Forwarder)

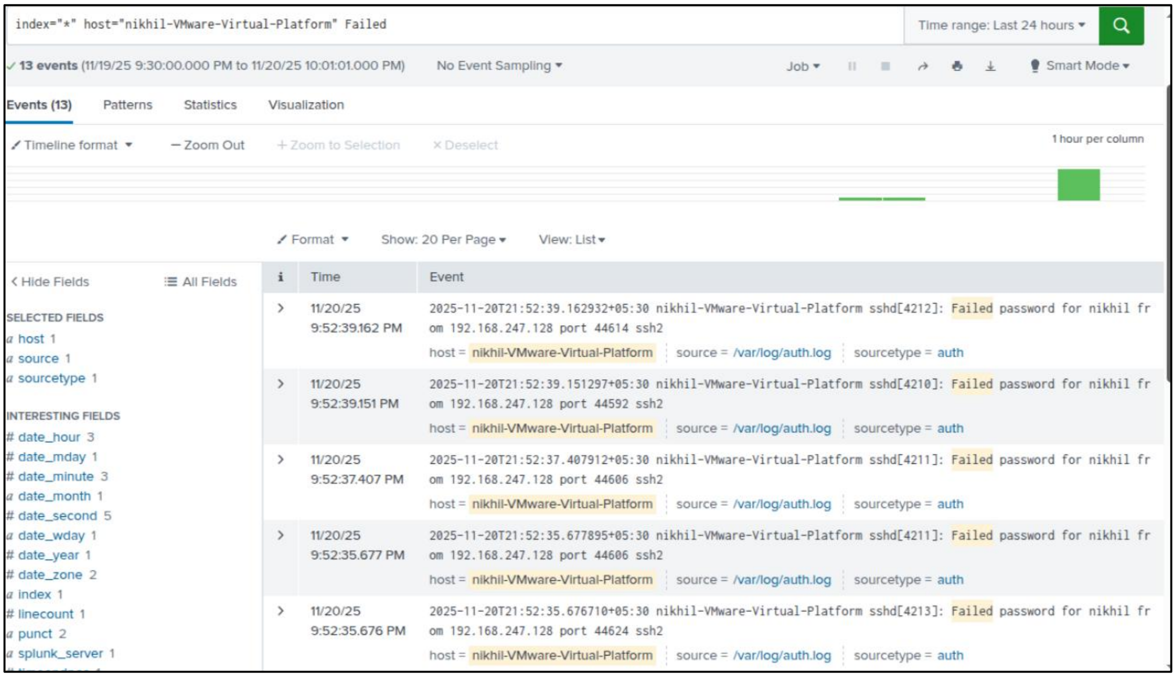
The core of the monitoring system relied on the Splunk Universal Forwarder (UF) installed on the Ubuntu Victim machine.

- The UF was required on the Linux host to efficiently and securely monitor the local system logs and stream them to the central Splunk Indexer.
- The UF was configured to monitor the SSH authentication log file: '*/var/log/auth.log*'.
- As shown in the Splunk dashboard, the UF successfully forwarded events like '*Failed password for nikhil fr on 192.168.247.128*', providing the essential timestamp, host, and attacker IP for analysis.

Detection Logic (Splunk SPL)

The primary detection mechanism was a correlation search designed to identify patterns typical of a Hydra-based brute force attack. We used Splunk Search Processing Language (SPL) to achieve this.

The detection rule operates as follows: Identify a single source IP that generates events consistent with a brute-force attack.



```
index="*" host="nikhil-VMware-Virtual-Platform" Failed
```

Time range: Last 24 hours

✓ **13 events** (11/19/25 9:30:00.000 PM to 11/20/25 10:01:01.000 PM)

No Event Sampling

SELECTED FIELDS
a host 1
a source 1
a sourcetype 1

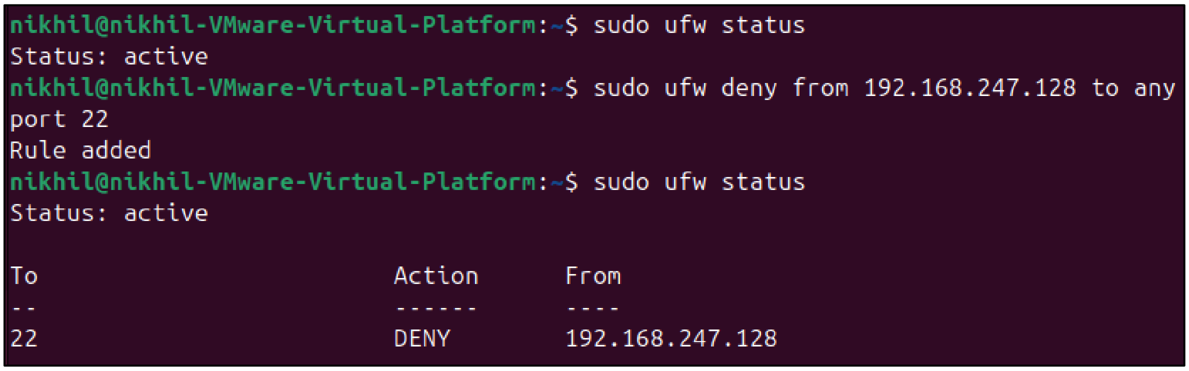
INTERESTING FIELDS
# date_hour 3
# date_mday 1
# date_minute 3
a date_month 1
# date_second 5
a date_wday 1
# date_year 1
# date_zone 2
a index 1
# linecount 1
a punct 2
a splunk_server 1

i	Time	Event
>	11/20/25 9:52:39.162 PM	2025-11-20T21:52:39.162932+05:30 nikhil-VMware-Virtual-Platform sshd[4212]: Failed password for nikhil from 192.168.247.128 port 44614 ssh2 host = nikhil-VMware-Virtual-Platform source = /var/log/auth.log sourcetype = auth
>	11/20/25 9:52:39.151 PM	2025-11-20T21:52:39.151297+05:30 nikhil-VMware-Virtual-Platform sshd[4210]: Failed password for nikhil from 192.168.247.128 port 44592 ssh2 host = nikhil-VMware-Virtual-Platform source = /var/log/auth.log sourcetype = auth
>	11/20/25 9:52:37.407 PM	2025-11-20T21:52:37.407912+05:30 nikhil-VMware-Virtual-Platform sshd[4211]: Failed password for nikhil from 192.168.247.128 port 44606 ssh2 host = nikhil-VMware-Virtual-Platform source = /var/log/auth.log sourcetype = auth
>	11/20/25 9:52:35.677 PM	2025-11-20T21:52:35.677895+05:30 nikhil-VMware-Virtual-Platform sshd[4211]: Failed password for nikhil from 192.168.247.128 port 44606 ssh2 host = nikhil-VMware-Virtual-Platform source = /var/log/auth.log sourcetype = auth
>	11/20/25 9:52:35.676 PM	2025-11-20T21:52:35.676710+05:30 nikhil-VMware-Virtual-Platform sshd[4213]: Failed password for nikhil from 192.168.247.128 port 44624 ssh2 host = nikhil-VMware-Virtual-Platform source = /var/log/auth.log sourcetype = auth

Automated Response (SOAR)

The project achieved full automation by linking the triggered Splunk alert to a defensive action on the victim machine.

- Orchestration Setup: SSH key-based authentication was configured between the Splunk user and a privileged user on the Ubuntu Victim machine.
- Shell Script Execution: When the Splunk alert fired, it executed an external shell script that passed the detected attacker's ‘*src_ip*’ as a variable.
- Enforcement (UFW Block): The script remotely ran the following UFW command on the victim machine, which was verified in the console to instantly block all further SSH connections from the attacker: ‘*sudo ufw deny from 192.168.247.128 to any port 22*’.



```
nikhil@nikhil-VMware-Virtual-Platform:~$ sudo ufw status
Status: active
nikhil@nikhil-VMware-Virtual-Platform:~$ sudo ufw deny from 192.168.247.128 to any
port 22
Rule added
nikhil@nikhil-VMware-Virtual-Platform:~$ sudo ufw status
Status: active

To                         Action      From
--                         ------      ----
22                         DENY        192.168.247.128
```

Note:

For the UFW deny rule to be effective against brute-force attacks, it must be processed before any existing ALLOW rules for the same service (e.g., SSH port 22). To ensure the deny rule takes precedence, the ‘*ufw insert 1*’ command is used, explicitly placing the block rule at the very beginning of the firewall chain.

Conclusion

This project successfully demonstrated the capability to monitor security events, leverage advanced SIEM functionality with SPL to identify complex attacks (specifically a Hydra brute force), and integrate with system controls (UFW) to perform autonomous, real-time threat mitigation.