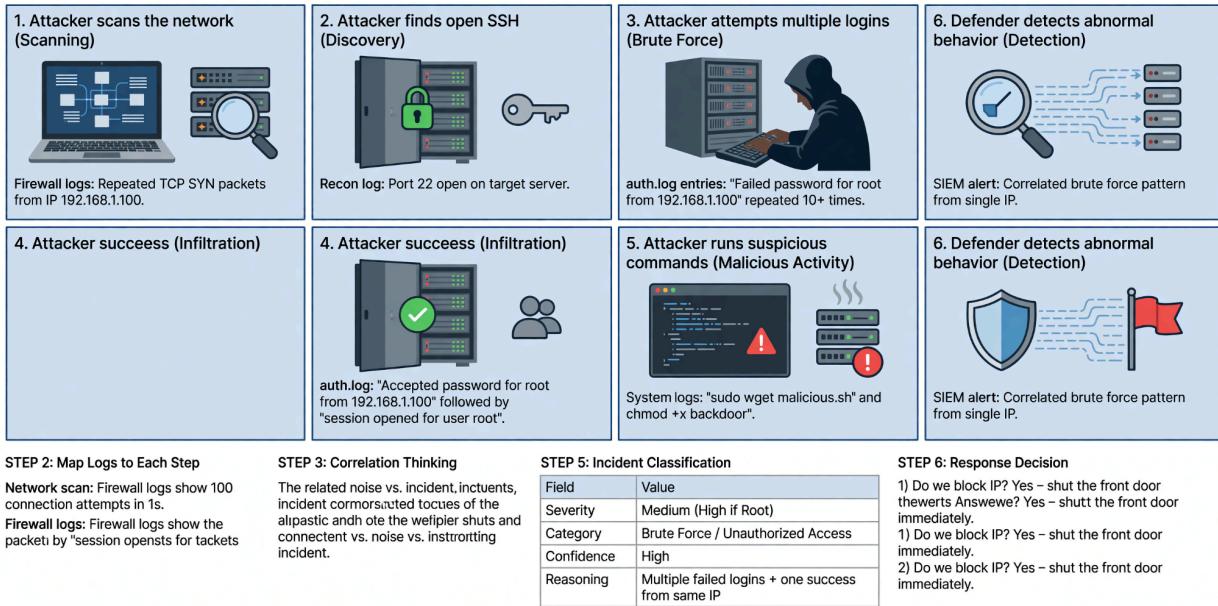


## DAY 12 – SIEM THINKING & ATTACK STORY CORRELATION



## DAY 12 – SIEM THINKING & ATTACK STORY CORRELATION

*This training session is about thinking like a security engineer. By the end, you'll be able to look at any incident and instantly pinpoint what happened, prove it, and execute the next steps.*

### STEP 1 ATTACK STORY CORRELATION :

1. Attacker scans the network (**Scanning**)
2. Attacker finds open SSH (**Discovery**)
3. Attacker attempts multiple logins (**Brute Force**)
4. Attacker succeeds (**Infiltration**)
5. Attacker runs suspicious commands (**Malicious Activity**)
6. Defender detects abnormal behavior (**Detection**)

## **Why**

I am doing this to connect raw data into a story. Security isn't just about logs; it's about seeing the attacker's journey from the outside to the inside.

## ***STEP 2: Map Logs to Each Step***

What log would show this?

1. Network scan(Scout):

Firewall logs

IDS logs

Connection attempts from one IP:

### **why:**

Because the Firewall is the Front Gate. It records every single knock.

If you see 100 knocks in 1 second, the firewall log proves it is a bot, not a person.

2. SSH brute force (Guessing):

auth.log

Multiple failed login entries

### **Why:**

This specific file records every time someone tries to log in.

This log shows Failed password over and over, which is the proof of an attack.

3. Successful login (Entry):

auth.log (Accepted password)

New session started

### **Why:**

The same camera that saw them fail now sees a line saying Accepted password. This is the most important log because it tells you exactly what time you were hacked.

### ***STEP 3: Correlation Thinking:***

One failed login is noise. (The user forgot their password).

Ten failed logins is suspicious. (A bot is trying to guess).

Ten failed + one success = INCIDENT. (The bot got in).

### **Why:**

Computers are "noisy." Thousands of things happen every second.

Correlation means connecting the dots to find the one thing that matters.

### **STEP 4: Manual Correlation Exercise (Hands-on, No Tools)**

On your Ubuntu VM:

```
sudo cat /var/log/auth.log
```

Look for:

- failed password
- accepted password
- session opened

Now imagine:

- Same IP

- Short time window

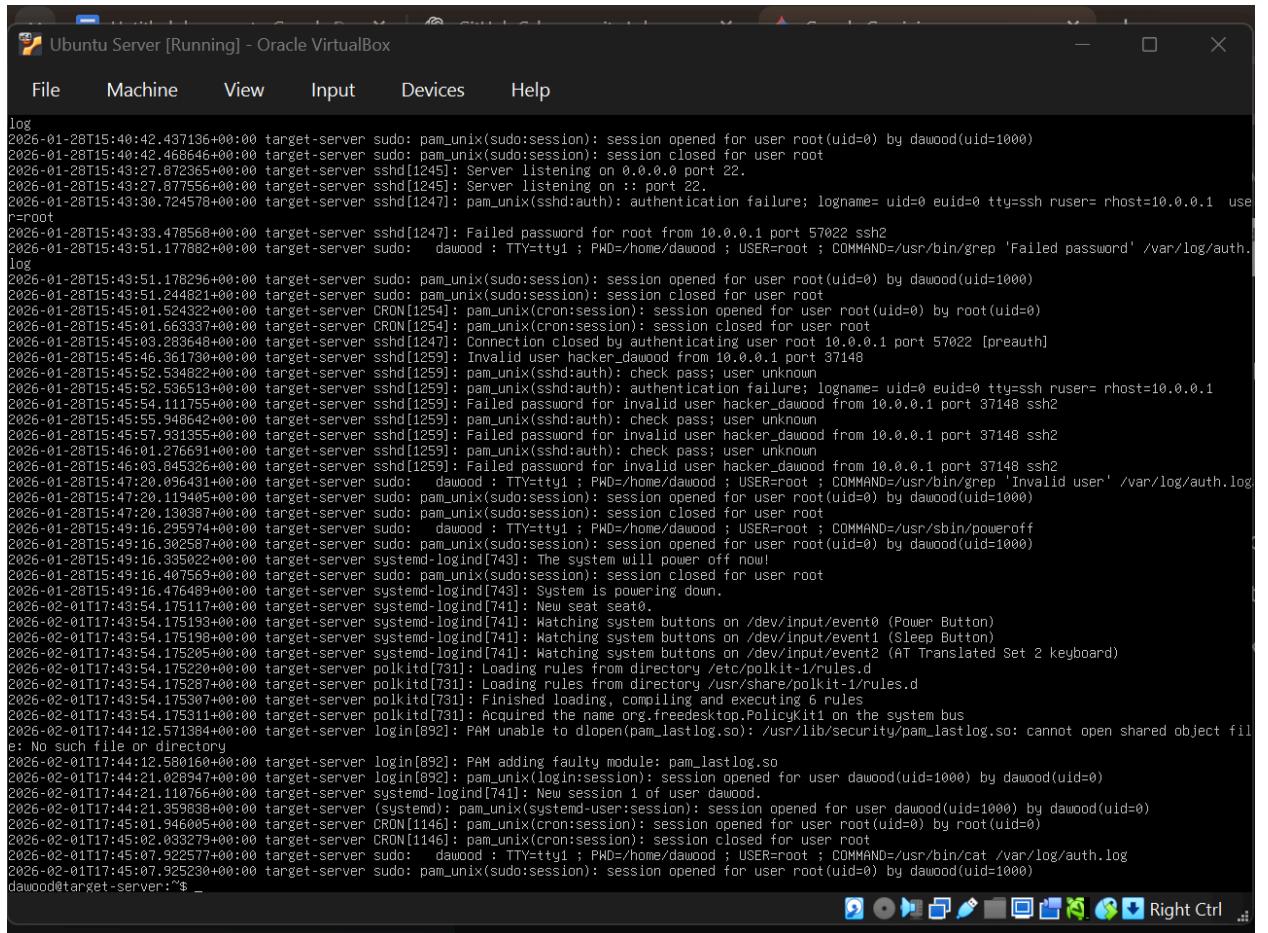
- Pattern

You are now doing **manual SIEM analysis**.

## Why:

Security isn't just about reading logs; it's about connecting them to tell a story. This logic is the core of professional Cloud SIEM tools like Splunk and Microsoft Sentinel.

## Proof:



The screenshot shows a Windows desktop environment with an Oracle VirtualBox window open. Inside the window, there is a terminal session running on an Ubuntu Server. The terminal displays a long log of system events and user activity. The log includes entries for sudo sessions, cron jobs, and various system services like sshd and polkitd. The text is in a monospaced font, typical of a terminal window.

```

Ubuntu Server [Running] - Oracle VirtualBox
File Machine View Input Devices Help

log
2026-01-28T15:40:42.437136+00:00 target-server sudo: pam_unix(sudo:session): session opened for user root(uid=0) by dawood(uid=1000)
2026-01-28T15:40:42.468646+00:00 target-server sudo: pam_unix(sudo:session): session closed for user root
2026-01-28T15:43:27.872365+00:00 target-server sshd[1245]: Server listening on 0.0.0.0 port 22.
2026-01-28T15:43:27.877556+00:00 target-server sshd[1245]: Server listening on :: port 22.
2026-01-28T15:43:30.724578+00:00 target-server sshd[1247]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=10.0.0.1 use
rroot
2026-01-28T15:43:33.478568+00:00 target-server sshd[1247]: Failed password for root from 10.0.0.1 port 57022 ssh2
2026-01-28T15:43:51.177882+00:00 target-server sudo:  dawood : TTY=tty1 ; PWD=/home/dawood ; USER=root ; COMMAND=/usr/bin/grep 'Failed password' /var/log/auth.log
2026-01-28T15:43:51.178296+00:00 target-server sudo: pam_unix(sudo:session): session opened for user root(uid=0) by dawood(uid=1000)
2026-01-28T15:43:51.244821+00:00 target-server sudo: pam_unix(sudo:session): session closed for user root
2026-01-28T15:45:01.524322+00:00 target-server CRON[1254]: pam_unix(cron:session): session opened for user root(uid=0) by root(uid=0)
2026-01-28T15:45:01.563337+00:00 target-server CRON[1254]: pam_unix(cron:session): session closed for user root
2026-01-28T15:45:03.361730+00:00 target-server sshd[1259]: Connection closed by authenticating user root 10.0.0.1 port 57022 [preauth]
2026-01-28T15:45:46.361730+00:00 target-server sshd[1259]: Invalid user hacker_dawood from 10.0.0.1 port 37148
2026-01-28T15:45:52.534822+00:00 target-server sshd[1259]: pam_unix(sshd:auth): check pass; user unknown
2026-01-28T15:45:52.536513+00:00 target-server sshd[1259]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=10.0.0.1
2026-01-28T15:45:54.111755+00:00 target-server sshd[1259]: Failed password for invalid user hacker_dawood from 10.0.0.1 port 37148 ssh2
2026-01-28T15:45:55.348642+00:00 target-server sshd[1259]: pam_unix(sshd:auth): check pass; user unknown
2026-01-28T15:45:57.581955+00:00 target-server sshd[1259]: Failed password for invalid user hacker_dawood from 10.0.0.1 port 37148 ssh2
2026-01-28T15:46:01.276651+00:00 target-server sshd[1259]: pam_unix(sshd:auth): check pass; user unknown
2026-01-28T15:46:03.845826+00:00 target-server sshd[1259]: Failed password for invalid user hacker_dawood from 10.0.0.1 port 37148 ssh2
2026-01-28T15:47:20.096431+00:00 target-server sudo:  dawood : TTY=tty1 ; PWD=/home/dawood ; USER=root ; COMMAND=/usr/bin/grep 'Invalid user' /var/log/auth.log
2026-01-28T15:47:20.119405+00:00 target-server sudo: pam_unix(sudo:session): session opened for user root(uid=0) by dawood(uid=1000)
2026-01-28T15:47:20.130387+00:00 target-server sudo: pam_unix(sudo:session): session closed for user root
2026-01-28T15:49:16.235974+00:00 target-server sudo:  dawood : TTY=tty1 ; PWD=/home/dawood ; USER=root ; COMMAND=/usr/sbin/poweroff
2026-01-28T15:49:16.302587+00:00 target-server sudo: pam_unix(sudo:session): session opened for user root(uid=0) by dawood(uid=1000)
2026-01-28T15:49:16.335622+00:00 target-server systemd-logind[743]: The system will power off now!
2026-01-28T15:49:16.407569+00:00 target-server sudo: pam_unix(sudo:session): session closed for user root
2026-01-28T15:49:16.476489+00:00 target-server systemd-logind[743]: System is powering down.
2026-02-01T17:43:54.175117+00:00 target-server systemd-logind[741]: New seat seat0.
2026-02-01T17:43:54.175193+00:00 target-server systemd-logind[741]: Watching system buttons on /dev/input/event0 (Power Button)
2026-02-01T17:43:54.175198+00:00 target-server systemd-logind[741]: Watching system buttons on /dev/input/event1 (Sleep Button)
2026-02-01T17:43:54.175205+00:00 target-server systemd-logind[741]: Watching system buttons on /dev/input/event2 (AT Translated Set 2 keyboard)
2026-02-01T17:43:54.175209+00:00 target-server polkitd[731]: Loading rules from directory /etc/polkit-1/rules.d
2026-02-01T17:43:54.175209+00:00 target-server polkitd[731]: Loading rules from directory /usr/share/polkit-1/rules.d
2026-02-01T17:43:54.175307+00:00 target-server polkitd[731]: Finished loading, compiling and executing 6 rules
2026-02-01T17:43:54.175311+00:00 target-server polkitd[731]: Acquired the name org.freedesktop.PolicyKit1 on the system bus
2026-02-01T17:44:12.571384+00:00 target-server login[892]: PAM unable to dlopen(pam_lastlog.so): /usr/lib/security/pam_lastlog.so: cannot open shared object file: No such file or directory
2026-02-01T17:44:12.580156+00:00 target-server login[892]: PAM adding faulty module: pam_lastlog.so
2026-02-01T17:44:21.028947+00:00 target-server login[892]: pam_unix(login:session): session opened for user dawood(uid=1000) by dawood(uid=0)
2026-02-01T17:44:21.110766+00:00 target-server systemd-logind[741]: New session 1 of user dawood.
2026-02-01T17:44:21.359838+00:00 target-server (systemd): pam_unix(systemd-user:session): session opened for user dawood(uid=1000) by dawood(uid=0)
2026-02-01T17:45:01.946095+00:00 target-server CRON[1146]: pam_unix(cron:session): session opened for user root(uid=0) by root(uid=0)
2026-02-01T17:45:02.03279+00:00 target-server CRON[1146]: pam_unix(cron:session): session closed for user root
2026-02-01T17:45:07.322577+00:00 target-server sudo:  dawood : TTY=tty1 ; PWD=/home/dawood ; USER=root ; COMMAND=/usr/bin/cat /var/log/auth.log
2026-02-01T17:45:07.322530+00:00 target-server sudo: pam_unix(sudo:session): session opened for user root(uid=0) by dawood(uid=1000)
dawood@target-server:~$ 

```

## **STEP 5: Incident Classification**

Write this table in your document:

Field	Value
<b>Severity</b>	<b>Medium</b> (High if it's a Root account)
<b>Category</b>	Brute Force / Unauthorized Access
<b>Confidence</b>	<b>High</b>
<b>Reasoning</b>	Multiple "Failed password" logs followed by "Accepted password" from the same IP.

### **Why:**

Management doesn't have time to read thousands of logs. They need this 4-line summary to make quick decisions. You are translating Digital

Evidence into Business Risk.

## ***STEP 6: Response Decision :***

Answer these questions:

1)Do we block IP?

**Yes,** If a specific address is trying to break in, you shut the front door on them immediately. This stops the attack from getting worse right now.

2)Do we reset credentials?

**Yes,** Even if the attacker is kicked out, they still know the password. You must change the locks. This ensures the attacker cannot just log back in later from a different address.

3)Do we monitor further?

**Yes,** You need to check if the attacker left a "backdoor" or created a new user while they were inside. You watch the logs for the next 24-48 hours to make sure the system is truly clean.

4)Do we escalate?

**Yes (to the IT/Legal team).**

If data was stolen, the company might have to tell the government or the customers.In a big company, you never hide a hack,you report it so the whole team can help.