Mini SOC project

🧪 Mini SOC Scenario: Detecting a Brute Force Attack

🎯 Objective:

Simulate and detect a brute-force SSH attack on a Linux system using Wireshark, Splunk, Nessus, and system logs. Students will:

- Capture network traffic.
- Monitor logs for failed SSH logins.
- Investigate system vulnerabilities.
- Write an incident report.

🧰 Tools Required

Tool	Purpose
Linux VM (e.g., Ubuntu)	Target system
Attacker VM (e.g., Kali Linux)	For brute-force simulation
Splunk Free	Log analysis
Wireshark	Packet capture
Nessus Essentials	Vulnerability assessment
Hydra or Ncrack	To simulate brute-force

📦 Lab Structure

✅ Task 1: Initial Setup and Configuration

1. **Target VM (Ubuntu)**
    - Ensure OpenSSH is installed and running:

```
sudo apt update
sudo apt install openssh-server
sudo systemctl status ssh
```

- Create a user for testing:

```
sudo adduser testuser
```

2. **Attacker VM (Kali)**
   - Install Hydra:

```
sudo apt install hydra
```

3. **Enable Logging and Tools**
   - On Ubuntu, ensure SSH logs go to `/var/log/auth.log`.
   - Install Splunk and Nessus on the Ubuntu VM or on a separate management VM.

✅ Task 2: Simulate Brute Force Attack (from Kali)

> Use Hydra to simulate brute force over SSH

```
hydra -l testuser -P /usr/share/wordlists/rockyou.txt ssh://<target-ip>
```

- This sends multiple SSH login attempts.
- Ensure at least 10–15 failed attempts.

✅ Task 3: Capture the Attack with Wireshark

1. Start **Wireshark** on the target or on a VM on the same network.
2. Begin packet capture before launching Hydra.
3. Apply filters to analyze traffic:

```
tcp.port == 22
```

4. Stop capture after Hydra finishes. Save the capture (`.pcapng`).

✅ Task 4: Analyze Logs with Splunk

1. **Upload `/var/log/auth.log` to Splunk**:
   - Add it via `Add Data > Upload`.
   - Set sourcetype to `linux_secure` or `linux_auth`.
2. **Run SPL Queries to Detect Attack**:
   - Find failed SSH attempts:

     ```
     index=main sourcetype=linux_auth "Failed password"
     | stats count by user, src, host
     ```

   - Get top attacking IPs:

     ```
     index=main sourcetype=linux_auth "Failed password"
     | top src
     ```

   - Timeline of login attempts:

     ```
     index=main sourcetype=linux_auth
     | timechart count by user
     ```

3. **Create Alert or Dashboard (optional)**:
   - Create a basic dashboard panel for "Failed Logins by IP".
   - Add a threshold-based alert: more than 5 failures from same IP in 2 minutes.

✅ Task 5: Scan for Vulnerabilities with Nessus

1. Run a **Basic Network Scan** with Nessus on the target VM.

2. Review for vulnerabilities like:
    - Weak SSH configurations
    - Unpatched packages
3. Export scan results (HTML or CSV).

✅ Task 6: Document the Incident

> Students should compile a Mini Incident Report

📄 Report Should Include:

Section	Details
Executive Summary	What was detected (e.g., brute force attack)
Timeline	When the attack occurred
Attack Source	IP address and tools used (Hydra)
Evidence	Wireshark packets, Splunk logs, screenshot of failed logins
System Vulnerabilities	Found using Nessus
Recommendations	e.g., IP blocking, fail2ban, strong password policies

---

📊 Grading Criteria (Optional)

Criteria	Marks
Correct execution of brute force	10
Wireshark capture with analysis	15
Log ingestion and queries in Splunk	20
Nessus scan with interpretation	15
Quality of incident report	20
Dashboard/alert setup (bonus)	+5
Total	80 (+5 Bonus)

📁 Optional Files for Submission

- `.pcap` file (Wireshark capture)
- `.csv` or `.html` Nessus report
- Splunk screenshots (queries, dashboards)
- `.pdf` incident report
