**Lab Exercise 4- Create an Index in Splunk, Upload Logs, and Run Queries for Security Use Cases**

**Objective**

In this lab, you'll follow a step-by-step process to:

1. Create an index in Splunk.
2. Upload log files (authentication logs and firewall logs).
3. Run queries to detect security issues like brute force attacks, privileged account usage, and suspicious network traffic.

**Step 1: Set Up Your Splunk Environment**

1. **Log into your Splunk instance**:
   * Open your browser and log into your Splunk instance (typically http://<your\_splunk\_server>:8000).
   * Use your admin credentials.
2. **Create an Index**:
   * Go to **Settings** (top-right corner) > **Indexes**.
   * Click **New Index**.
   * In the **Index Name** field, enter security (or any other name you prefer).
   * Click **Save** to create the index.

**Step 2: Upload Log Files to Splunk**

1. **Download the Sample Log Files**:
   * Save the following log data as two separate log files.

**Sample Authentication Log File (authentication\_logs.log):**

192.168.1.1 - - [22/Dec/2024:12:00:00 +0000] "POST /login HTTP/1.1" 200 512

192.168.1.2 - - [22/Dec/2024:12:01:00 +0000] "POST /login HTTP/1.1" 401 512

192.168.1.3 - - [22/Dec/2024:12:02:00 +0000] "POST /login HTTP/1.1" 401 512

192.168.1.1 - - [22/Dec/2024:12:03:00 +0000] "POST /login HTTP/1.1" 401 512

192.168.1.4 - - [22/Dec/2024:12:04:00 +0000] "POST /login HTTP/1.1" 200 512

192.168.1.1 - - [22/Dec/2024:12:05:00 +0000] "POST /login HTTP/1.1" 401 512

192.168.1.1 - - [22/Dec/2024:12:06:00 +0000] "POST /login HTTP/1.1" 401 512

192.168.1.2 - - [22/Dec/2024:12:07:00 +0000] "POST /login HTTP/1.1" 200 512

192.168.1.3 - - [22/Dec/2024:12:08:00 +0000] "POST /login HTTP/1.1" 401 512

192.168.1.1 - - [22/Dec/2024:12:09:00 +0000] "POST /login HTTP/1.1" 200 512

**Sample Firewall Log File (firewall\_logs.log):**

22/Dec/2024:12:00:00 192.168.1.1 10.0.0.1 80 TCP ALLOW

22/Dec/2024:12:00:01 192.168.1.2 10.0.0.1 80 TCP ALLOW

22/Dec/2024:12:00:02 192.168.1.3 10.0.0.1 80 TCP ALLOW

22/Dec/2024:12:00:03 192.168.1.1 10.0.0.1 80 TCP DENY

22/Dec/2024:12:00:04 192.168.1.1 10.0.0.1 80 TCP ALLOW

22/Dec/2024:12:00:05 192.168.1.2 10.0.0.1 80 TCP ALLOW

22/Dec/2024:12:00:06 192.168.1.3 10.0.0.1 80 TCP ALLOW

22/Dec/2024:12:00:07 192.168.1.4 10.0.0.1 80 TCP ALLOW

22/Dec/2024:12:00:08 192.168.1.1 10.0.0.1 80 TCP ALLOW

22/Dec/2024:12:00:09 192.168.1.1 10.0.0.1 80 TCP DENY

22/Dec/2024:12:00:10 192.168.1.1 10.0.0.1 80 TCP ALLOW

22/Dec/2024:12:00:11 192.168.1.2 10.0.0.1 80 TCP ALLOW

22/Dec/2024:12:00:12 192.168.1.5 10.0.0.1 80 TCP ALLOW

22/Dec/2024:12:00:13 192.168.1.1 10.0.0.1 80 TCP DENY

22/Dec/2024:12:00:14 192.168.1.1 10.0.0.1 80 TCP ALLOW

22/Dec/2024:12:00:15 192.168.1.2 10.0.0.1 80 TCP ALLOW

1. **Upload the Log Files to Splunk**:
   * Navigate to **Settings** > **Add Data**.
   * Select **Upload** and browse to select authentication\_logs.log and firewall\_logs.log files.
   * Choose the index security (or create a new one).
   * Proceed with the upload and make sure the data is indexed under the correct sourcetype.

**Step 3: Run Queries to Detect Security Use Cases**

**Use Case 1: Detect Brute Force Attacks from Failed Logins**

1. **Query to Detect Failed Logins**:
   * Run the following search to detect brute-force attacks by identifying multiple failed login attempts from the same IP.

index=security sourcetype=authentication "failure"

| stats count by user, src\_ip, \_time

| where count > 5

1. **Explanation**:
   * This query searches for failure messages in the authentication logs and groups them by user and src\_ip.
   * It counts the number of failed login attempts for each user and source IP, and the where count > 5 filter ensures that it only returns results where there are more than 5 failed attempts, indicating a potential brute-force attack.
2. **Save as Alert**:
   * Once the query runs successfully, click **Save As** > **Alert**.
   * Configure the alert with:
     + **Trigger Condition**: Trigger the alert when the count exceeds 5.
     + **Schedule**: Set the alert to run every 5 minutes.
     + **Alert Action**: Send an email or a webhook for notification.
   * Click **Save** to create the alert.

**Use Case 2: Monitor Privileged Account Usage (e.g., root or admin)**

1. **Query to Detect Privileged Account Logins**:
   * Run the following query to detect when privileged accounts such as root or admin are used to log in.

index=security sourcetype=authentication "root" OR "admin"

| stats count by user, src\_ip, \_time

1. **Explanation**:
   * This query searches for the usage of root or admin in authentication logs. If these accounts are used to log in, they will be flagged.
   * The query counts the login attempts by each user and source IP.
2. **Save as Alert**:
   * Save the search as an alert with the following configuration:
     + **Trigger Condition**: Trigger when root or admin is found in the logs.
     + **Schedule**: Set to run every 10 minutes.
     + **Alert Action**: Send an email or create a webhook to notify security teams.
   * Click **Save**.

**Use Case 3: Detect Suspicious Network Traffic (DDoS Detection)**

1. **Query to Detect Suspicious Traffic**:
   * Run this query to detect high numbers of requests from a single IP address, which could indicate a DDoS attack.

index=security sourcetype=firewall

| stats count by src\_ip

| where count > 100

1. **Explanation**:
   * This query counts the number of requests made by each source IP (src\_ip), which could potentially highlight a DDoS attack if an IP is making more than 100 requests in a short time frame.
2. **Save as Alert**:
   * Save the query as an alert and configure:
     + **Trigger Condition**: When the count exceeds 100.
     + **Schedule**: Run the alert every 5 minutes.
     + **Alert Action**: Email or Webhook.

**Step 4: Create a Dashboard to Visualize Security Events**

1. **Create a New Dashboard**:
   * Go to **Dashboards** > **Create New Dashboard**.
   * Name it Security Monitoring Dashboard.
   * Add a panel for each use case.
2. **Panel for Failed Logins**:
   * Add a panel to show failed login attempts over time.
   * Use the following search for the panel:

index=security sourcetype=authentication "failure"

| timechart span=5m count by user

1. **Panel for Privileged Account Usage**:
   * Add a panel to visualize privileged account logins.
   * Use this query:

index=security sourcetype=authentication "root" OR "admin"

| timechart span=1h count by user

1. **Panel for Suspicious Traffic**:
   * Add a panel to visualize the source IPs with the highest traffic.
   * Use this query:

index=security sourcetype=firewall

| stats count by src\_ip

1. **Save the Dashboard**:
   * Save your dashboard and monitor it to detect security incidents in real time.

**Conclusion**

In this lab, you learned how to create an index in Splunk, upload sample security log data, and create searches to detect failed logins, privileged account usage, and suspicious network traffic. You also created alerts and visualized the data through dashboards to monitor potential security incidents.

By continuously monitoring these use cases, you can proactively respond to security threats in your environment.