**Lab Exercise 7- Working with Multiple Log Files and Complex SPL Queries**

**Objective**

Ingest multiple log files into a single Splunk index and perform advanced SPL queries across different data scenarios, covering search, statistics, visualization, and data transformation.

**Log Files**

**File 1: web\_access.log**

192.168.1.1 - - [23/Mar/2024:10:01:45] "GET /home HTTP/1.1" 200 524

192.168.1.2 - - [23/Mar/2024:10:02:12] "POST /login HTTP/1.1" 401 128

192.168.1.3 - - [23/Mar/2024:10:03:33] "GET /profile HTTP/1.1" 403 256

192.168.1.4 - - [23/Mar/2024:10:04:45] "GET /dashboard HTTP/1.1" 200 1024

192.168.1.5 - - [23/Mar/2024:10:05:15] "POST /update HTTP/1.1" 200 400

**File 2: auth\_activity.log**

Mar 23 10:01:45 server1 sshd[1234]: Failed password for user root from 192.168.1.1 port 22 ssh2

Mar 23 10:02:12 server1 sshd[1256]: Accepted password for user admin from 192.168.1.2 port 22 ssh2

Mar 23 10:03:33 server1 sshd[1278]: Failed password for invalid user guest from 192.168.1.3 port 22 ssh2

Mar 23 10:04:45 server1 sshd[1300]: Failed password for user alice from 192.168.1.4 port 22 ssh2

Mar 23 10:05:15 server1 sshd[1322]: Accepted password for user bob from 192.168.1.5 port 22 ssh2

**File 3: firewall.log**

Mar 23 10:01:45 fw1 connection allowed from 192.168.1.1 to 172.16.0.1

Mar 23 10:02:12 fw1 connection denied from 192.168.1.2 to 172.16.0.2

Mar 23 10:03:33 fw1 connection allowed from 192.168.1.3 to 172.16.0.3

Mar 23 10:04:45 fw1 connection denied from 192.168.1.4 to 172.16.0.4

Mar 23 10:05:15 fw1 connection allowed from 192.168.1.5 to 172.16.0.5

**Steps for the Lab Exercise**

**1. Create Index**

1. Log in to **Splunk Web**.
2. Navigate to **Settings > Indexes > New Index**.
3. Create an index with the following details:
   * **Index Name**: multi\_logs\_index
   * **Data Type**: Events
4. Save the index.

**2. Upload Log Files**

1. Navigate to **Settings > Add Data > Upload**.
2. Upload all three log files (web\_access.log, auth\_activity.log, firewall.log).
3. Assign the data to the **multi\_logs\_index**.
4. Verify data ingestion by running the following query:

index=multi\_logs\_index

**Complex SPL Queries**

**General Search and Filtering**

1. **Search for HTTP GET Requests Only**

index=multi\_logs\_index "GET"

1. **Search for Failed Logins**

index=multi\_logs\_index "Failed password"

**Statistical Analysis**

1. **Count of Events by Log Source**

index=multi\_logs\_index | stats count by sourcetype

1. **Top 3 Users with Most Failed Logins**

index=multi\_logs\_index "Failed password" | stats count by user | sort -count | head 3

1. **HTTP Status Code Distribution**

index=multi\_logs\_index | rex "HTTP/1.1\" (?<status\_code>\d{3})" | stats count by status\_code

**Data Transformation**

1. **Extract IP Addresses from Web Logs**

index=multi\_logs\_index | rex field=\_raw "from (?<src\_ip>\d{1,3}(\.\d{1,3}){3})"

1. **Tagging Events Based on HTTP Status**

index=multi\_logs\_index | eval status\_category=if(status\_code<400, "Success", "Failure") | stats count by status\_category

**Time-Based Queries**

1. **Failed Logins Over Time**

index=multi\_logs\_index "Failed password" | timechart span=1m count

1. **Connections Allowed and Denied Over Time**

index=multi\_logs\_index "connection" | eval status=if(match(\_raw, "allowed"), "Allowed", "Denied") | timechart span=1m count by status

**Advanced Correlation**

1. **Correlation Between Auth and Firewall Logs**

index=multi\_logs\_index | transaction src\_ip maxspan=2s | stats count by src\_ip

1. **Identify IPs with Failed Logins and Denied Connections**

index=multi\_logs\_index "Failed password" OR "connection denied" | stats dc(src\_ip) as unique\_ips values(src\_ip) as ip\_list

**Top/Bottom Analysis**

1. **Top 5 IPs by Event Count**

index=multi\_logs\_index | stats count by src\_ip | sort -count | head 5

1. **Bottom 3 Users by Login Attempts**

index=multi\_logs\_index | stats count by user | sort count | head 3

**Dashboard Queries**

1. **Create a Panel for Firewall Connections**

index=multi\_logs\_index "connection" | stats count by status

1. **Create a Panel for Top Users by Successful Logins**

index=multi\_logs\_index "Accepted password" | stats count by user | sort -count

**Visualization**

1. **Pie Chart for HTTP Status Distribution**

index=multi\_logs\_index | rex "HTTP/1.1\" (?<status\_code>\d{3})" | stats count by status\_code | chart count by status\_code

1. **Bar Chart for Connections Over Time**

index=multi\_logs\_index "connection" | timechart span=1m count by status

**Advanced SPL Features**

1. **Create a New Field for Byte Conversion**

index=multi\_logs\_index | eval kb\_transferred=bytes/1024 | stats sum(kb\_transferred) by src\_ip

1. **Aggregate Data from Multiple Sources**

index=multi\_logs\_index | stats count by sourcetype

1. **Anomaly Detection for Failed Logins**

index=multi\_logs\_index "Failed password" | anomalydetection method=iqr

**Deliverables**

1. **Log Files**: Ensure they are uploaded to the specified index.
2. **Query Results**: Submit screenshots for each SPL query output.
3. **Dashboard**: Create a dashboard with panels for at least 5 queries from this lab.
4. **Insights**: Write a short summary of findings from the queries.

This lab provides a comprehensive exercise covering multiple aspects of Splunk data analysis.