Splunk eval Command: A Comprehensive Guide for Apache HTTPD Logs

Introduction

The eval command in Splunk is a powerful tool for manipulating fields, creating new fields, and applying various functions and operators. In the context of Apache HTTPD logs, it can be used to extract valuable insights and perform complex analysis.

Key Functions and Examples

1. Mathematical Operations

- **Convert response time from milliseconds to seconds:**
  Splunk SPL
  ```
  index=your_index sourcetype=access_combined | eval response_time_sec
  = response_time / 1000 | table clientip, uri, response_time,
  response_time_sec
  ```

2. String Manipulation

- **Extract the domain from the referer field:**
  Splunk SPL
  ```
  index=your_index sourcetype=access_combined | eval
  referer_domain=replace(referer, "https?://([^/]+)/.*", "\1") | table
  _time, referer, referer_domain
  ```
- **Concatenate client IP and URI to form a unique identifier:**
  Splunk SPL
  ```
  index=your_index sourcetype=access_combined | eval
  unique_id=clientip . "-" . uri | table _time, clientip, uri,
  unique_id
  ```

3. Conditional Logic (if, case)

- **Flagging high response times:**
  Splunk SPL
  ```
  index=your_index sourcetype=access_combined | eval
  slow_response=if(response_time > 2000, "Yes", "No") | table _time,
  ```

```
clientip, uri, response_time, slow_response
```

- **Categorizing HTTP status codes into ranges:**
  Splunk SPL
  ```
  index=your_index sourcetype=access_combined | eval
  status_category=case(status >= 200 AND status < 300, "2xx Success",
  status >= 400 AND status < 500, "4xx Client Error", status >= 500,
  "5xx Server Error", 1=1, "Other") | table _time, clientip, uri,
  status, status_category
  ```

4. Date and Time Functions

- **Extract the hour of the day:**
  Splunk SPL
  ```
  index=your_index sourcetype=access_combined | eval
  hour=strftime(_time, "%H") | table _time, clientip, uri, hour
  ```
- **Calculate the time difference between events:**
  Splunk SPL
  ```
  index=your_index sourcetype=access_combined | streamstats current=f
  window=1 earliest(_time) as previous_time by clientip | eval
  time_diff = _time - previous_time | table _time, clientip, uri,
  time_diff
  ```

5. Geolocation (IP-based)

- **Extract country from an IP lookup:**
  Splunk SPL
  ```
  index=your_index sourcetype=access_combined | lookup geoip clientip
  as ip OUTPUT country | table _time, clientip, uri, country
  ```

6. Multi-value Fields

- **Split useragent into individual components:**
  Splunk SPL
  ```
  index=your_index sourcetype=access_combined | eval
  useragent_parts=split(useragent, "/") | table _time, clientip,
  useragent, useragent_parts
  ```
- **Expand multi-value fields into individual events:**

Splunk SPL

```
index=your_index sourcetype=access_combined | eval
referer_domains=split(referer, ",") | mvexpand referer_domains |
table _time, clientip, uri, referer_domains
```

7. Coalesce Function

- **Fallback value for missing fields:**

  Splunk SPL

  ```
  index=your_index sourcetype=access_combined | eval
  final_user=coalesce(user, "anonymous") | table _time, clientip, uri,
  final_user
  ```

8. Round Function

- **Round response time to the nearest second:**

  Splunk SPL

  ```
  index=your_index sourcetype=access_combined | eval response_time_sec
  = round(response_time/1000, 2) | table _time, clientip, uri,
  response_time_sec
  ```

9. Substr Function

- **Extract a portion of the URI:**

  Splunk SPL

  ```
  index=your_index sourcetype=access_combined | eval
  uri_prefix=substr(uri, 1, 10) | table _time, clientip, uri,
  uri_prefix
  ```

10. Length Function

- **Check the length of the URI:**

  Splunk SPL

  ```
  index=your_index sourcetype=access_combined | eval
  uri_length=len(uri) | table _time, clientip, uri, uri_length
  ```

11. Replace Function

- **Mask part of the IP address:**
  Splunk SPL
  ```
  index=your_index sourcetype=access_combined | eval
  masked_ip=replace(clientip, "\.\d+$", ".xxx") | table _time,
  clientip, masked_ip
  ```

12. Lower and Upper Functions

- **Convert useragent to lowercase:**
  Splunk SPL
  ```
  index=your_index sourcetype=access_combined | eval
  useragent_lower=lower(useragent) | table _time, clientip, uri,
  useragent_lower
  ```

Conclusion

By mastering the use of the eval command with these functions, you can manipulate and enrich your Apache HTTPD logs to gain deeper insights and build more powerful queries in Splunk.