Splunk Query Language Cheat Sheet

Splunk Query Language Cheat Sheet

Splunk's Search Processing Language (SPL) is the backbone of data analysis and monitoring in the platform. Whether you're a seasoned analyst or just getting started, having a quick reference for the most common commands, functions, and syntax can save you hours of trial and error. This cheat sheet distills the essentials of SPL into a single, easy-to-read guide that you can keep handy while working on dashboards, alerts, or ad-hoc investigations.

1. Basic Search Structure

At its core, every SPL query follows a simple pattern:

```
search [criteria] | [command] [options] | [command] [options]
```

Here's a quick rundown:

- **search** – Optional keyword that initiates the search. Often omitted when the first term is a field or event.
- **criteria** – Field filters (e.g., `host=webserver`) or time modifiers (e.g., `earliest=-24h`).
- **commands** – Functions that transform, aggregate, or visualize the data.

2. Common Search Commands

- **where** – Advanced filtering using boolean logic. Example: `| where status=200 AND response_time < 500`
- **stats** – Aggregation functions. Example: `| stats count, avg(response_time) by status`
- **chart** – Creates a bar or line chart directly in the search results. Example: `| chart count by status`
- **timechart** – Aggregates data over time. Example: `| timechart span=1h count`
- **eval** – Calculates new fields. Example: `| eval error_rate = (errors / total_requests) * 100`
- **rename** – Renames fields for clarity. Example: `| rename user_id as uid`
- **dedup** – Removes duplicate events based on specified fields. Example: `| dedup user_id`
- **sort** – Orders results. Example: `| sort -response_time`
- **head/tail** – Limits the number of events. Example: `| head 10`
- **rex** – Extracts fields using regular expressions. Example: `| rex field=_raw "(?\d{3})"`

3. Field Extraction & Manipulation

Field extraction is often the first step before analysis. Use the `rex` command for regex extraction or rely on Splunk's automatic extraction when possible.

1. **Automatic Extraction** – Splunk parses event data into fields based on delimiters.
2. **rex** – Manual extraction. Syntax: `| rex field=field_name "(?pattern)"`
3. **eval** – Transform fields. Example: `| eval month=strftime(_time, "%b")`

4. Time Modifiers

Time is critical in Splunk searches. Use the following modifiers to define the search window:

- **earliest** – Start time. Example: `earliest=-7d@d` (last 7 days, rounded to the day).
- **latest** – End time. Example: `latest=now` (current moment).
- **span** – Time bucket size for `timechart`. Example: `span=5m` (5 minutes).

5. Advanced Functions

Beyond basic stats, SPL offers powerful functions for complex analytics:

- **mvexpand** – Expands multivalue fields into separate events.
- **mvcount** – Counts values in a multivalue field.
- **spath** – Parses JSON or XML data. Example: `| spath input=payload output=user_id`
- **transaction** – Groups events into transactions based on keys and time. Example: `| transaction session_id maxspan=30m`
- **streamstats** – Performs running calculations across events.

6. Example Queries

1. **Count of HTTP Status Codes**
   `index=web_logs | stats count by status | sort -count`
2. **Average Response Time per Host**
   `index=web_logs | stats avg(response_time) as avg_time by host | sort avg_time`
3. **Top 5 Users by Request Count**
   `index=auth_logs | stats count by user_id | sort -count | head 5`
4. **Alert on Error Rate Spike**
   `index`

Downloaded from hotelcommunityforum.com by Lillianna & Conner • Books are a uniquely portable magic. • Page 1 • 2025-06-06