Extracting Web Server Log Data in Splunk

This document explains how to use regular expressions (regex) with the rex command in Splunk to extract specific fields from web server access logs. This allows you to analyze and understand website traffic patterns and server health.

1. Extracting Client IP Address

Log Format: We'll assume logs are in Apache HTTPD access log format, like this:

```
192.168.1.1 - - [26/Sep/2024:12:34:56 +0000] "GET /index.html
HTTP/1.1" 200 2326 "http://example.com" "Mozilla/5.0 (Windows NT 10.0;
Win64; x64)"
```

SPL Query:

```
index=ashu_web earliest=-30d
| rex field=_raw "^(?P<client_ip>\d{1,3}\.\d{1,3}\.\d{1,3}\.\d{1,3})"
| stats count by client_ip
```

Explanation:

- rex field=_raw "^(?P<client_ip>\d{1,3}\.\d{1,3}\.\d{1,3}\.\d{1,3})": This line extracts the client IP address using regex.
  - ^: Matches the beginning of the line.
  - (?P<client_ip>\d{1,3}\.\d{1,3}\.\d{1,3}\.\d{1,3}): Captures 1-3 digits separated by dots as the client_ip named group.
- stats count by client_ip: Groups and counts events by the extracted client IP address.

Result: This query counts the number of requests from each unique IP address.

2. Extracting HTTP Status Code

SPL Query:

```
index=ashu_web earliest=-30d
| rex field=_raw "status\s(?P<http_status>\d{3})"
| stats count by http_status
```

Explanation:

- rex field=_raw "status\s(?P<http_status>\d{3})": This line extracts the HTTP status code.
  - status\s: Matches the word "status" followed by a space.
  - (?P<http_status>\d{3}): Captures a 3-digit number (status code) as the http_status named group.
- stats count by http_status: Groups and counts events by the extracted HTTP status code.

Result: This query counts the number of requests for each HTTP status code (e.g., 200 for success, 404 for not found).

3. Extracting URL Path

Assuming: Logs have a request field containing the URL path.

SPL Query:

```
index=ashu_web earliest=-30d
| rex field=_raw "GET\s(?P<url_path>/\S+)\sHTTP"
| stats count by url_path
```

Explanation:

- rex field=_raw "GET\s(?P<url_path>/\S+)\sHTTP": This line extracts the URL path after the GET method.
  - GET\s: Matches the GET method followed by a space.
  - (?P<url_path>/\S+): Captures any non-whitespace characters following a forward slash (/) as the url_path named group.
- stats count by url_path: Groups and counts events by the extracted URL path.

Result: This query counts the number of requests for each unique URL path.

4. Extracting User-Agent Information

Assuming: Logs include a User-Agent string indicating the browser or client.

SPL Query:

```
index=ashu_web earliest=-30d
| rex field=_raw "\"\s(?P<user_agent>[^\"]+)\"$"
| stats count by user_agent
```

Explanation:

- rex field=_raw "\"\s(?P<user_agent>[^\"]+)\"$": This line extracts the User-Agent string.
    - "\s: Matches a double quote followed by a space.
    - (?P<user_agent>[^\"]+): Captures any characters except double quotes as the user_agent named group.
- stats count by user_agent: Groups and counts events by the