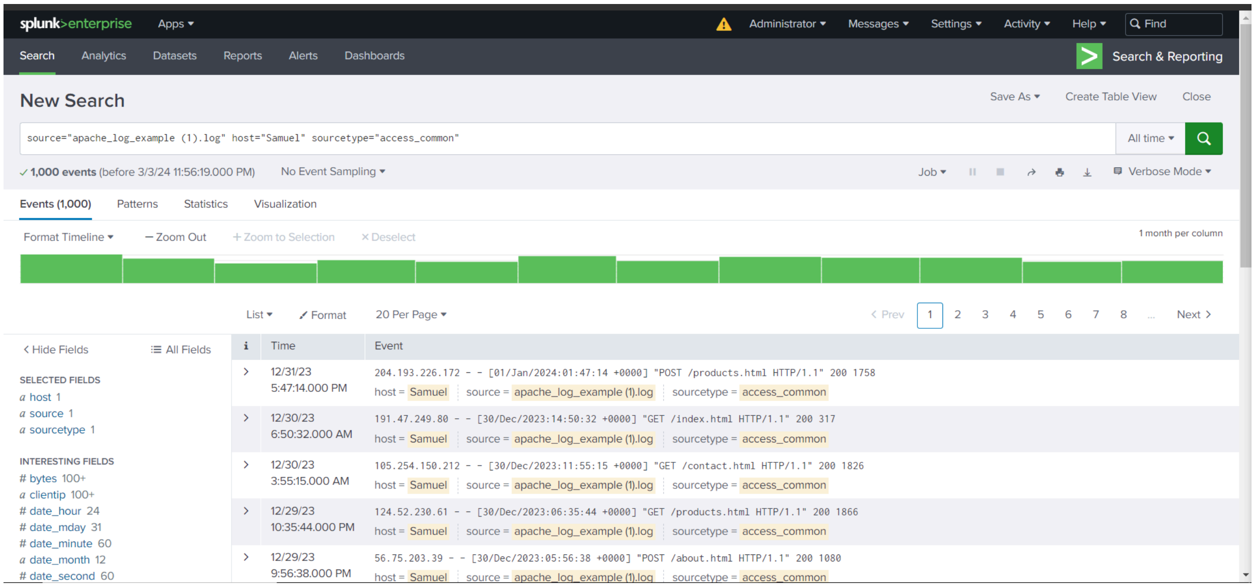
Apache Log Analysis:

1. Data Ingestion:



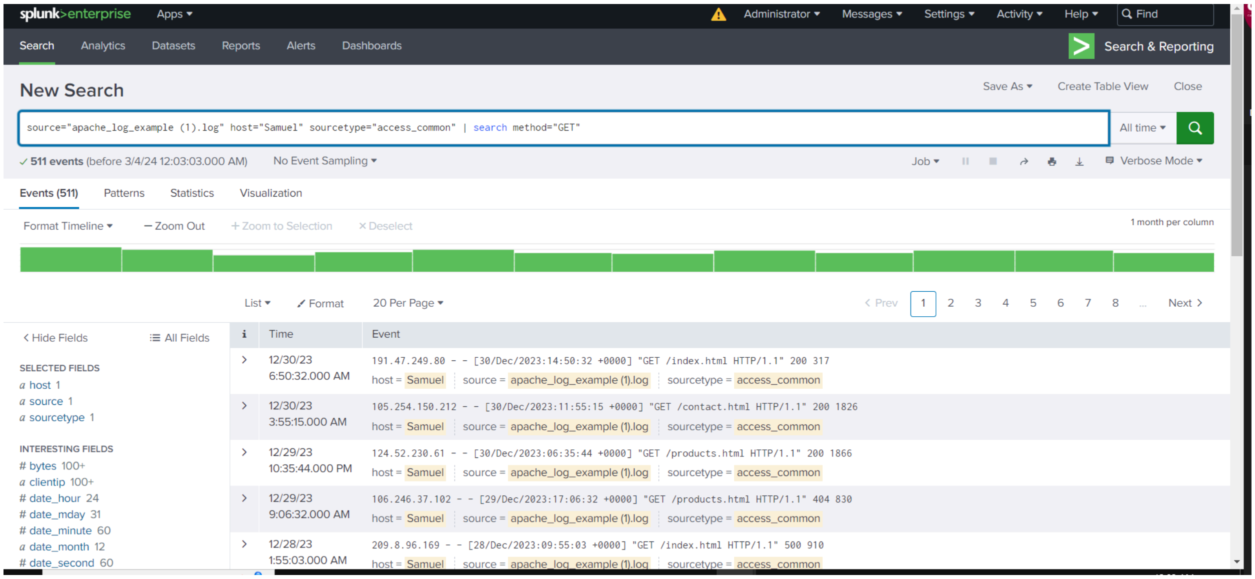
New Search

```
source="apache_log_example (1).log" host="Samuel" sourcetype="access_common"
```

✓ 1,000 events (before 3/3/24 11:56:19.000 PM) No Event Sampling ▾

i	Time	Event
›	12/31/23 5:47:14.000 PM	204.193.226.172 - - [01/Jan/2024:01:47:14 +0000] "POST /products.html HTTP/1.1" 200 1758 host = Samuel source = apache_log_example (1).log sourcetype = access_common
›	12/30/23 6:50:32.000 AM	191.47.249.80 - - [30/Dec/2023:14:50:32 +0000] "GET /index.html HTTP/1.1" 200 317 host = Samuel source = apache_log_example (1).log sourcetype = access_common
›	12/30/23 3:55:15.000 AM	105.254.150.212 - - [30/Dec/2023:11:55:15 +0000] "GET /contact.html HTTP/1.1" 200 1826 host = Samuel source = apache_log_example (1).log sourcetype = access_common
›	12/29/23 10:35:44.000 PM	124.52.230.61 - - [30/Dec/2023:06:35:44 +0000] "GET /products.html HTTP/1.1" 200 1866 host = Samuel source = apache_log_example (1).log sourcetype = access_common
›	12/29/23 9:56:38.000 PM	56.75.203.39 - - [30/Dec/2023:05:56:38 +0000] "POST /about.html HTTP/1.1" 200 1080 host = Samuel source = apache_log_example (1).log sourcetype = access_common

2. Searching:



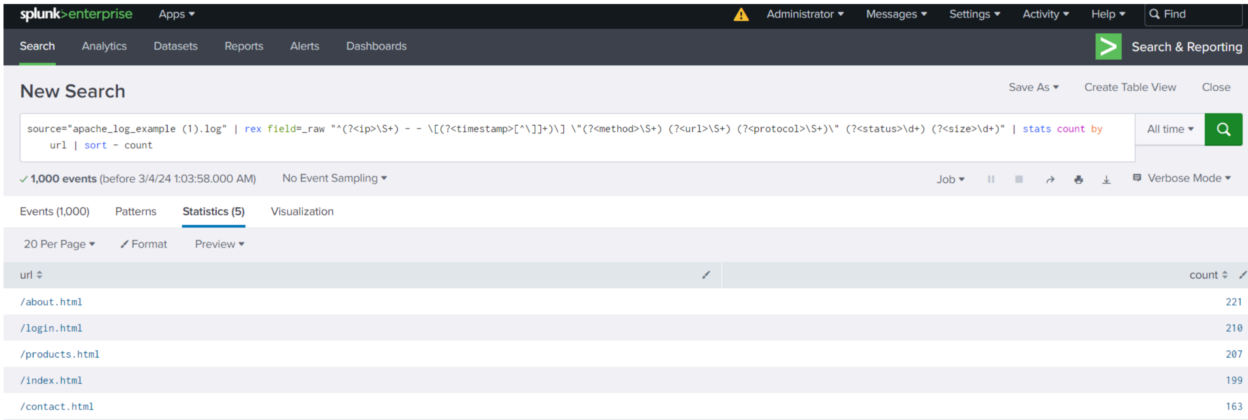
New Search

```
source="apache_log_example (1).log" host="Samuel" sourcetype="access_common" | search method="GET"
```

✓ 511 events (before 3/4/24 12:03:03.000 AM) No Event Sampling ▾

i	Time	Event
›	12/30/23 6:50:32.000 AM	191.47.249.80 - - [30/Dec/2023:14:50:32 +0000] "GET /index.html HTTP/1.1" 200 317 host = Samuel source = apache_log_example (1).log sourcetype = access_common
›	12/30/23 3:55:15.000 AM	105.254.150.212 - - [30/Dec/2023:11:55:15 +0000] "GET /contact.html HTTP/1.1" 200 1826 host = Samuel source = apache_log_example (1).log sourcetype = access_common
›	12/29/23 10:35:44.000 PM	124.52.230.61 - - [30/Dec/2023:06:35:44 +0000] "GET /products.html HTTP/1.1" 200 1866 host = Samuel source = apache_log_example (1).log sourcetype = access_common
›	12/29/23 9:06:32.000 AM	106.246.37.102 - - [29/Dec/2023:17:06:32 +0000] "GET /products.html HTTP/1.1" 404 830 host = Samuel source = apache_log_example (1).log sourcetype = access_common
›	12/28/23 1:55:03.000 AM	209.8.96.169 - - [28/Dec/2023:09:55:03 +0000] "GET /index.html HTTP/1.1" 500 910 host = Samuel source = apache_log_example (1).log sourcetype = access_common

- Total 511 GET request were discovered in the apache logs using query: ***source="apache_log_example (1).log" host="Samuel" sourcetype="access_common" | search method="GET"***

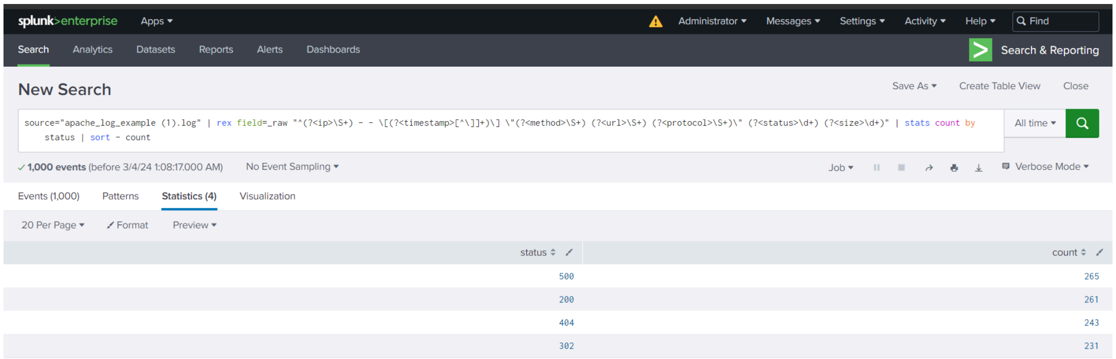


```
source="apache_log_example (1).log" | rex field=_raw "^(?<ip>\S+) - - \[(?<timestamp>[^\]]+)\] \"(?<method>\S+) (?<url>\S+) (?<protocol>\S+)\" (?<status>\d+) (?<size>\d+)" | stats count by
    url | sort - count
```

✓ 1,000 events (before 3/4/24 1:03:58.000 AM) No Event Sampling

url	count
/about.html	221
/login.html	210
/products.html	207
/index.html	199
/contact.html	163

In the apache logs the */about.html* is most requested URL with over 221 requests. This was discovered using the search query: ***source="apache_log_example (1).log" host="Samuel" sourcetype="access_common" | rex field=_raw "^(?<ip>\S+) - - \[(?<timestamp>[^\]]+)\] \"(?<method>\S+) (?<url>\S+) (?<protocol>\S+)\" (?<status>\d+) (?<size>\d+)" | stats count by url | sort - count***



```
source="apache_log_example (1).log" | rex field=_raw "^(?<ip>\S+) - - \[(?<timestamp>[^\]]+)\] \"(?<method>\S+) (?<url>\S+) (?<protocol>\S+)\" (?<status>\d+) (?<size>\d+)" | stats count by
    status | sort - count
```

✓ 1,000 events (before 3/4/24 1:08:17.000 AM) No Event Sampling

status	count
500	265
200	261
404	243
302	231

The most common response code in the apache logs is the response code 500 which indicates internal server error which happened total 265 times for the given logs.

After the response code 500, the most common response code is 200 followed by 404 and 302.

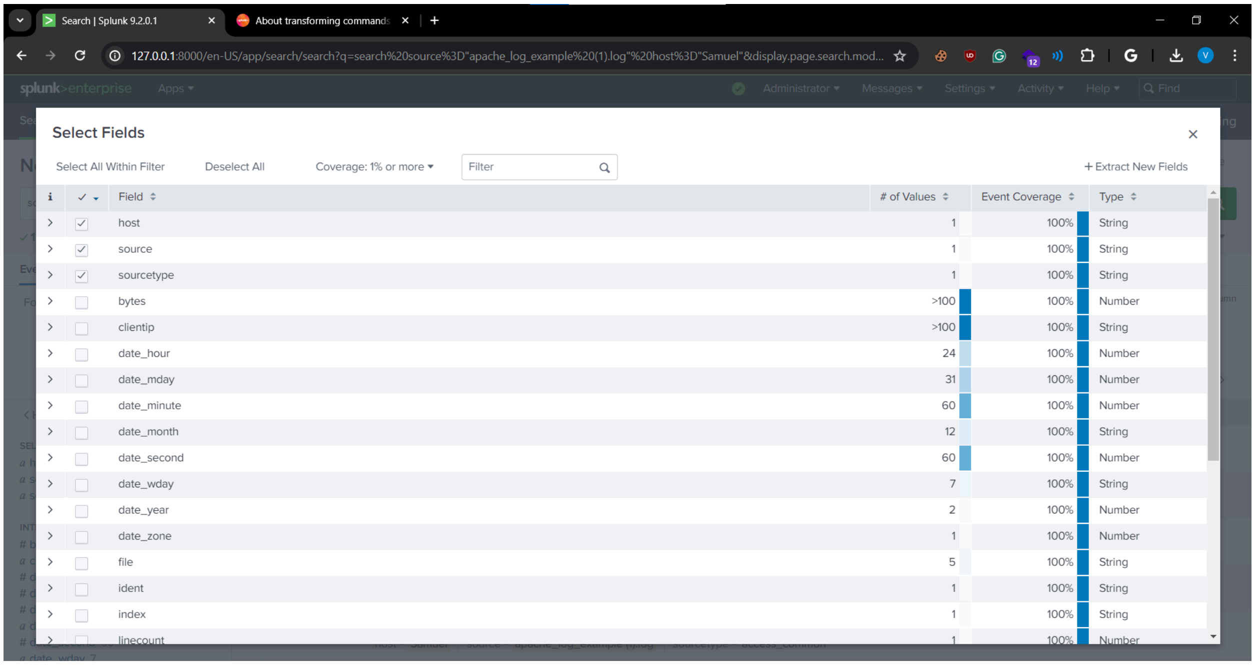
In order to search for most common request code use the following query:

- ***source="apache_log_example (1).log" | rex field=_raw "^(?<ip>\S+) - - \[(?<timestamp>[^\]]+)\] \"(?<method>\S+) (?<url>\S+)***

(?<protocol>\S+)\" (?<status>\d+) (?<size>\d+)" | stats count by status | sort - count

- Where, rex indicated the regular expression of the URL, stats counts how many times a given status code is encountered in the logs, and ‘sort’ helps sort the counts of the response code

3. Field Extraction:



Search | Splunk 9.2.0.1 — About transforming commands

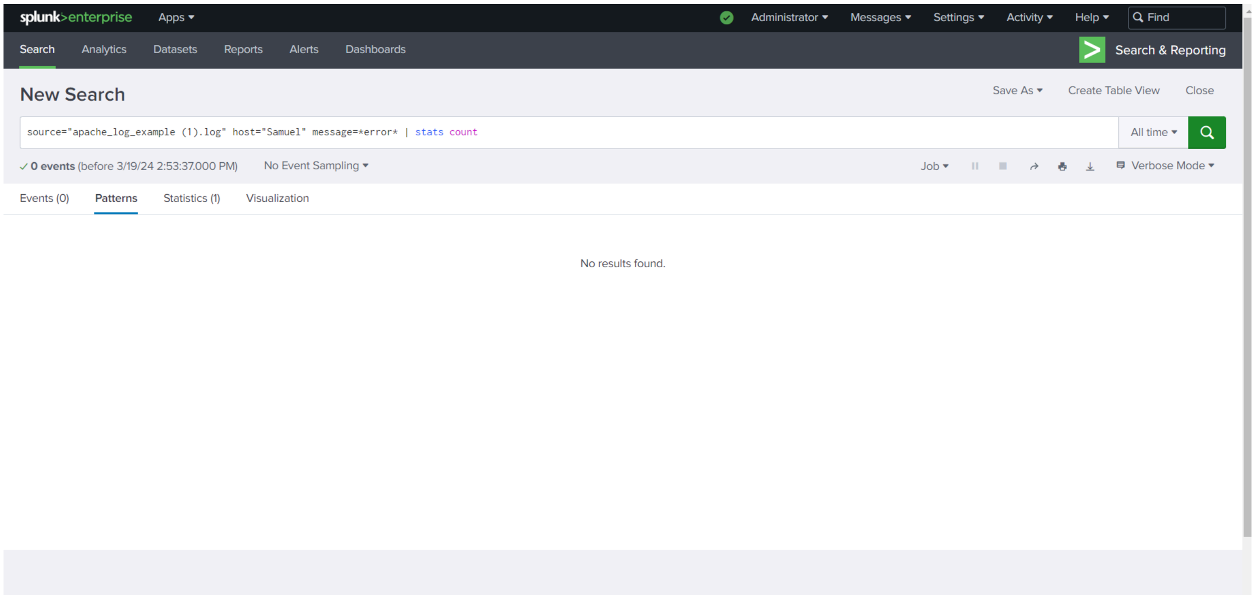
127.0.0.1:8000/en-US/app/search/search?q=search%20source%3D"apache_log_example%20(1).log"%20host%3D"Samuel"&display.page.search.mod...

Select Fields

Select All Within Filter · Deselect All · Coverage: 1% or more ▾ · Filter · + Extract New Fields

i	✓	Field	# of Values	Event Coverage	Type
›	☑	host	1	100%	String
›	☑	source	1	100%	String
›	☑	sourcetype	1	100%	String
›	□	bytes	>100	100%	Number
›	□	clientip	>100	100%	String
›	□	date_hour	24	100%	Number
›	□	date_mday	31	100%	Number
›	□	date_minute	60	100%	Number
›	□	date_month	12	100%	String
›	□	date_second	60	100%	Number
›	□	date_wday	7	100%	String
›	□	date_year	2	100%	Number
›	□	date_zone	1	100%	Number
›	□	file	5	100%	String
›	□	ident	1	100%	String
›	□	index	1	100%	String
›	□	linecount	1	100%	Number

4. Finding errors:



New Search

```
source="apache_log_example (1).log" host="Samuel" message=*error* | stats count
```

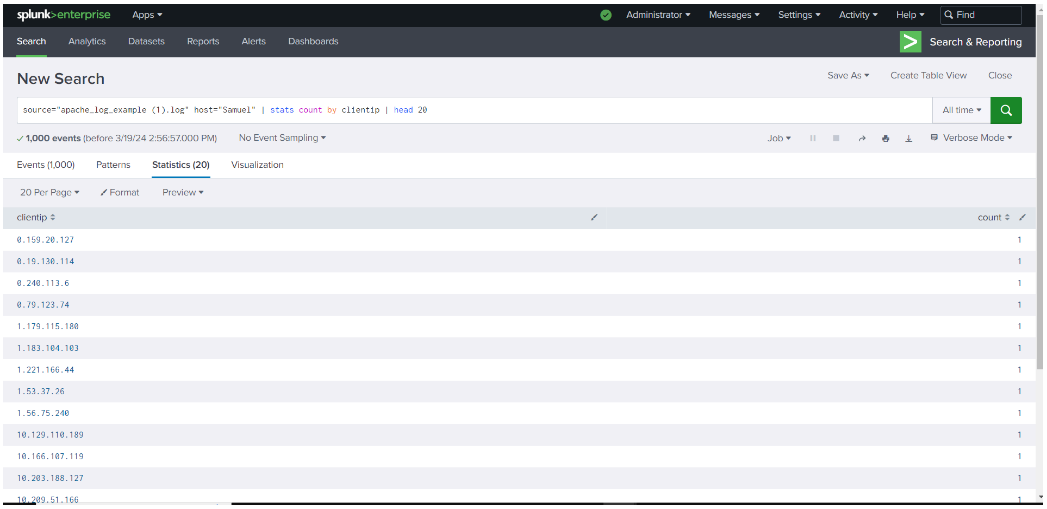
✓ **0 events** (before 3/19/24 2:53:37.000 PM) No Event Sampling ▾

No results found.

- ***source=”apache_log_example (1).log” host=”Samuel” message=*error****
- The search for errors in the specified log file did not yield any results, indicating no errors were found in the logs for the specified host.

5. Analysing Client IP:

- Using command ***source="apache_log_example (1).log" host="Samuel" | stats count by clientip | head 20*** no IP addresses were discovered that had frequent access attempts



New Search

```
source="apache_log_example (1).log" host="Samuel" | stats count by clientip | head 20
```

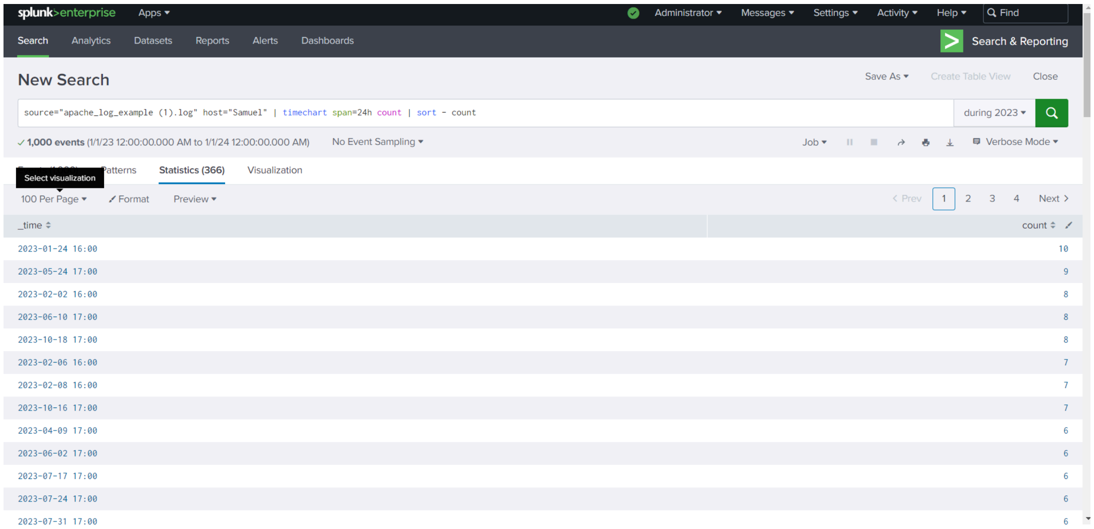
✓ 1,000 events (before 3/19/24 2:56:57.000 PM) No Event Sampling ▾

Events (1,000) Patterns Statistics (20) Visualization

clientip	count
0.159.20.127	1
0.19.130.114	1
0.240.113.6	1
0.79.123.74	1
1.179.115.180	1
1.183.104.103	1
1.221.166.44	1
1.53.37.26	1
1.56.75.240	1
10.129.110.189	1
10.166.107.119	1
10.203.188.127	1
10.209.51.166	1

6. Time Analysis:

- Conducting time analysis of the request with a span of 24 hours using search query: ***source="apache_log_example (1).log" host="Samuel" | timechart span=24h count | sort - count***
- Search results in increasing order of the requests per 24 hours



New Search

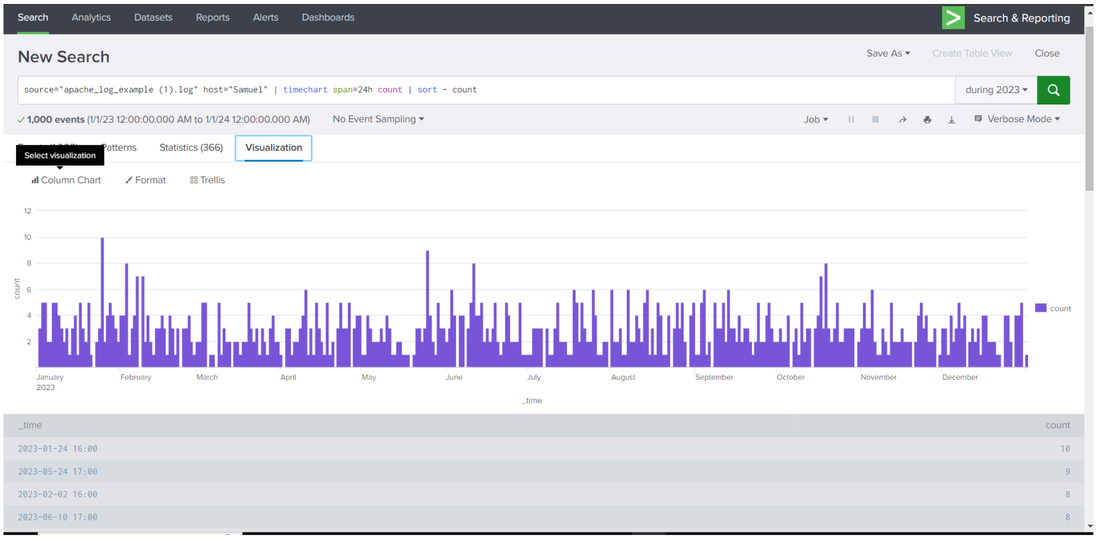
```
source="apache_log_example (1).log" host="Samuel" | timechart span=24h count | sort - count
```

✓ 1,000 events (1/1/23 12:00:00.000 AM to 1/1/24 12:00:00.000 AM) No Event Sampling ▾

Patterns Statistics (366) Visualization

_time	count
2023-01-24 16:00	10
2023-05-24 17:00	9
2023-02-02 16:00	8
2023-06-10 17:00	8
2023-10-18 17:00	8
2023-02-06 16:00	7
2023-02-08 16:00	7
2023-10-16 17:00	7
2023-04-09 17:00	6
2023-06-02 17:00	6
2023-07-17 17:00	6
2023-07-24 17:00	6
2023-07-31 17:00	6

- Visualizing the search results:

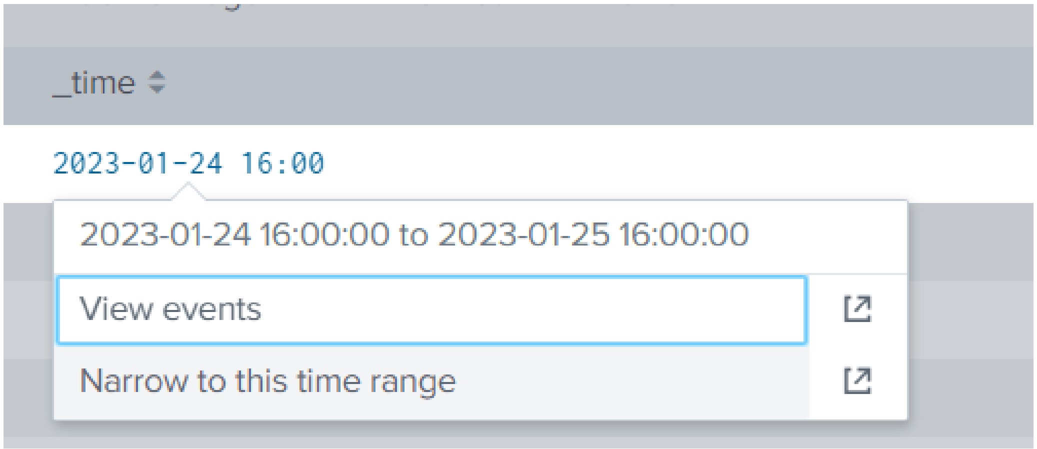


```
source="apache_log_example (1).log" host="Samuel" | timechart span=24h count | sort - count
```

✓ **1,000 events** (1/1/23 12:00:00.000 AM to 1/1/24 12:00:00.000 AM) No Event Sampling ▾

_time	count
2023-01-24 16:00	10
2023-05-24 17:00	9
2023-02-02 16:00	8
2023-06-10 17:00	8

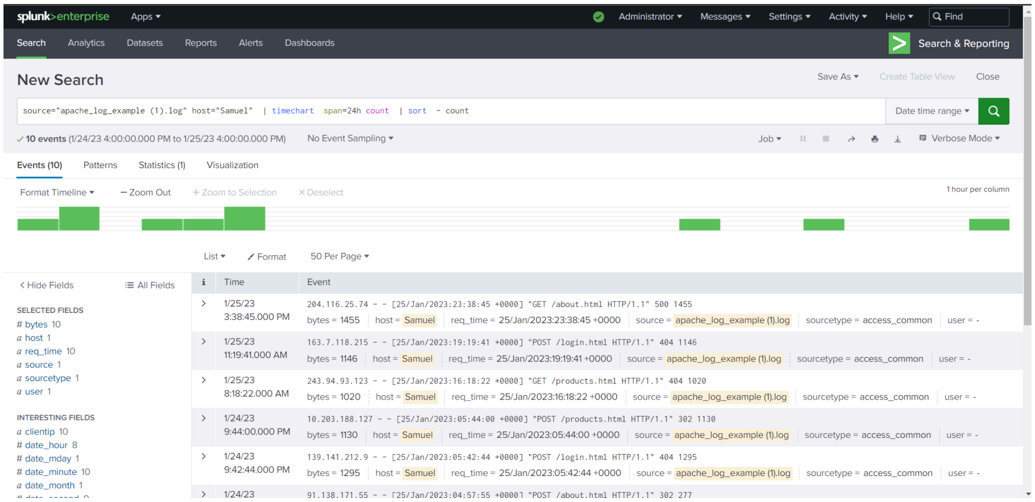
- 2023-01-24 has the highest number of requests for further analysis



2023-01-24 16:00

2023-01-24 16:00:00 to 2023-01-25 16:00:00

- Further Analyzing the events from 2023-01-24



New Search

```
source="apache_log_example (1).log" host="Samuel"  | timechart  span=24h count  | sort  - count
```

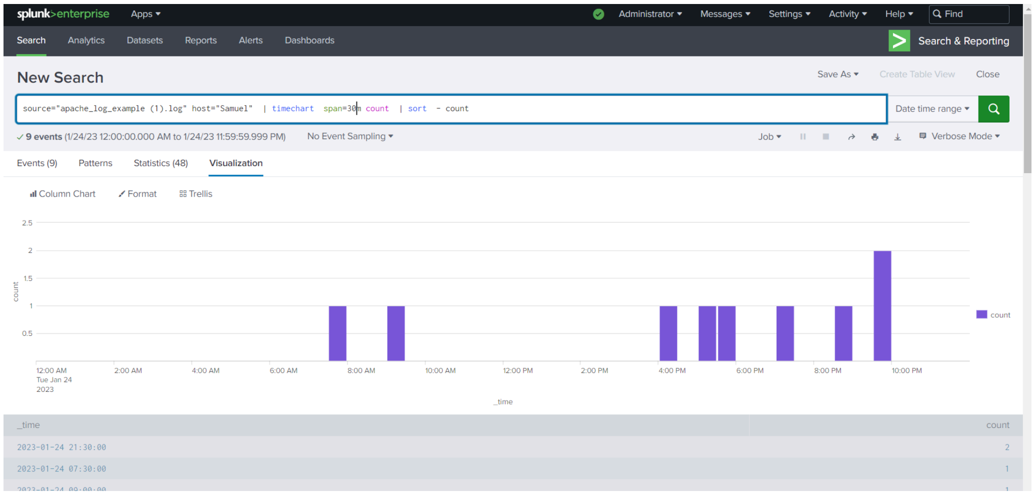
✓ **10 events** (1/24/23 4:00:00.000 PM to 1/25/23 4:00:00.000 PM)

SELECTED FIELDS
# bytes 10
a host 1
a req_time 10
a source 1
a sourcetype 1
a user 1

INTERESTING FIELDS
a clientip 10
# date_hour 8
# date_mday 1
# date_minute 10
a date_month 1

Time	Event
1/25/23 3:38:45.000 PM	204.116.25.74 - - [25/Jan/2023:23:38:45 +0000] "GET /about.html HTTP/1.1" 500 1455 bytes = 1455 host = Samuel req_time = 25/Jan/2023:23:38:45 +0000 source = apache_log_example (1).log sourcetype = access_common user = -
1/25/23 11:19:41.000 AM	163.7.118.215 - - [25/Jan/2023:19:19:41 +0000] "POST /login.html HTTP/1.1" 404 1146 bytes = 1146 host = Samuel req_time = 25/Jan/2023:19:19:41 +0000 source = apache_log_example (1).log sourcetype = access_common user = -
1/25/23 8:18:22.000 AM	243.94.93.123 - - [25/Jan/2023:16:18:22 +0000] "GET /products.html HTTP/1.1" 404 1020 bytes = 1020 host = Samuel req_time = 25/Jan/2023:16:18:22 +0000 source = apache_log_example (1).log sourcetype = access_common user = -
1/24/23 9:44:00.000 PM	10.203.188.127 - - [25/Jan/2023:05:44:00 +0000] "POST /products.html HTTP/1.1" 302 1130 bytes = 1130 host = Samuel req_time = 25/Jan/2023:05:44:00 +0000 source = apache_log_example (1).log sourcetype = access_common user = -
1/24/23 9:42:44.000 PM	139.141.212.9 - - [25/Jan/2023:05:42:44 +0000] "POST /login.html HTTP/1.1" 404 1295 bytes = 1295 host = Samuel req_time = 25/Jan/2023:05:42:44 +0000 source = apache_log_example (1).log sourcetype = access_common user = -
1/24/23	91.138.171.55 - - [25/Jan/2023:04:57:55 +0000] "POST /about.html HTTP/1.1" 302 277

- Visualizing the events using time chart from the day 24th Jan 2023 with span of 30m



New Search

```
source="apache_log_example (1).log" host="Samuel"  | timechart  span=30m count  | sort  - count
```

✓ **9 events** (1/24/23 12:00:00.000 AM to 1/24/23 11:59:59.999 PM)

_time	count
2023-01-24 21:30:00	2
2023-01-24 07:30:00	1
2023-01-24 09:00:00	1