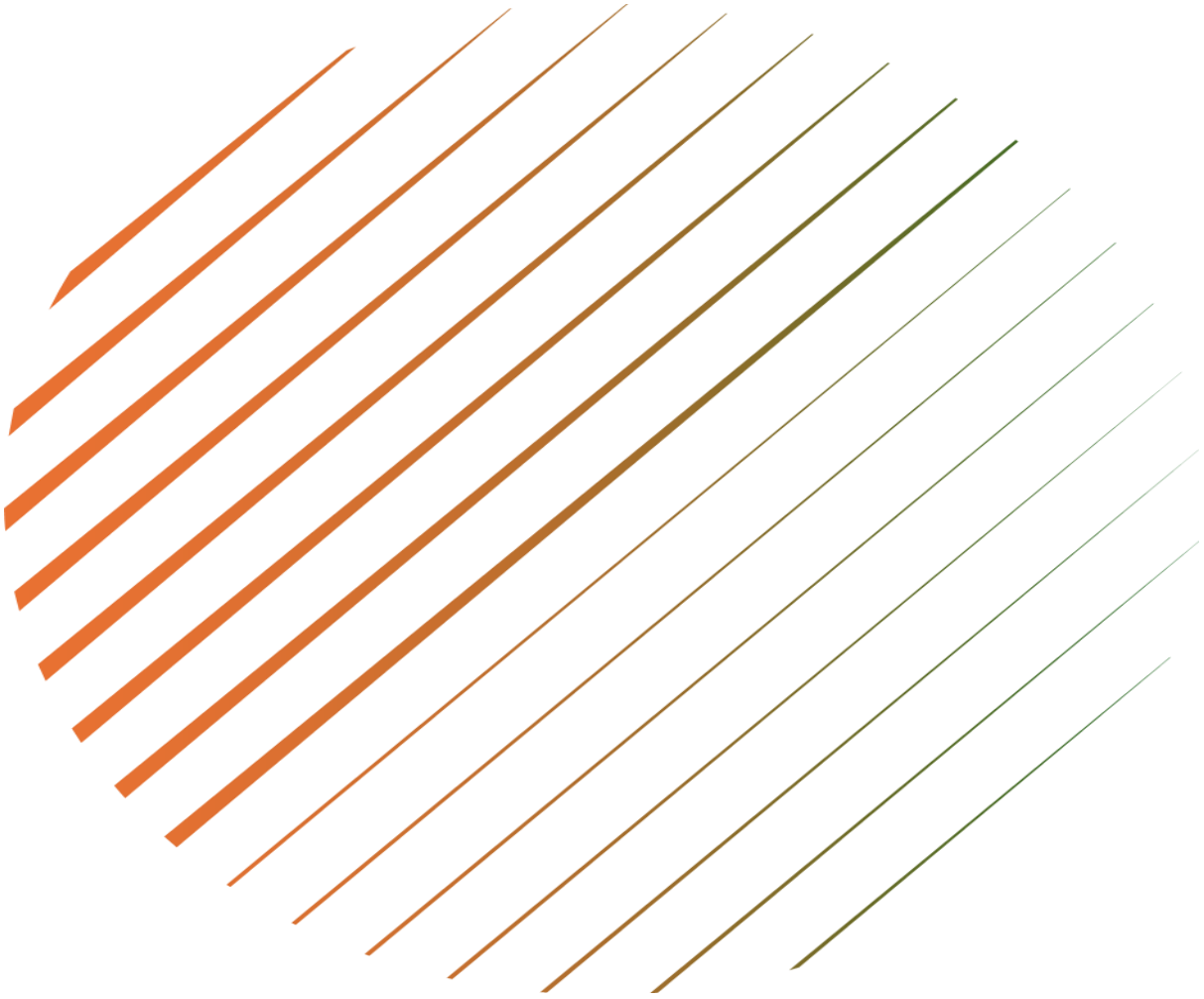
HTTP LOG ANALYSIS AND WEB SECURITY MONITORING PROJECT



PREPARED BY: PRANIT KALAMBATE

ROLE/PROGRAM: TRAINED IN CEH CURRICULUM / CYBERSECURITY ANALYST TRAINING

DATE COMPLETED: 10-12-2025

1.PROJECT OVERVIEW

1.1 Goal

The primary goal of this project is to perform **HTTP access log analysis** using Splunk Enterprise to:

- Extract critical web traffic fields (IPs, Methods, Status Codes, URIs).
- Analyze web usage patterns (most frequent URIs and methods).
- Perform security monitoring to identify unauthorized access attempts and scanning activities based on HTTP status codes.

1.2 Data Source

The project utilizes web server access logs ingested into a Splunk index.

- Sourcetype: "httplog"
- Index:"index=* OR index=sourcetype=httplog" (Inferred from search queries).
- Total Events Analyzed: Approximately 875,960 events.

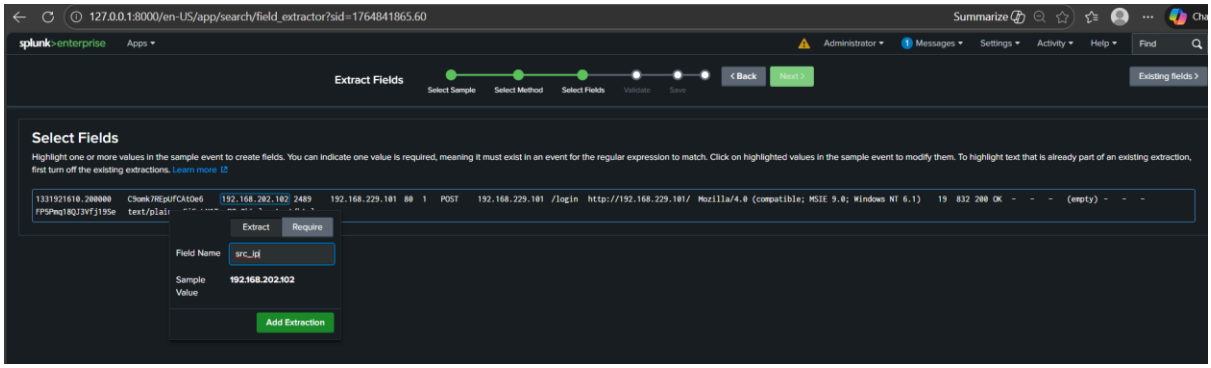
2.FIELD EXTRACTION PROCESS

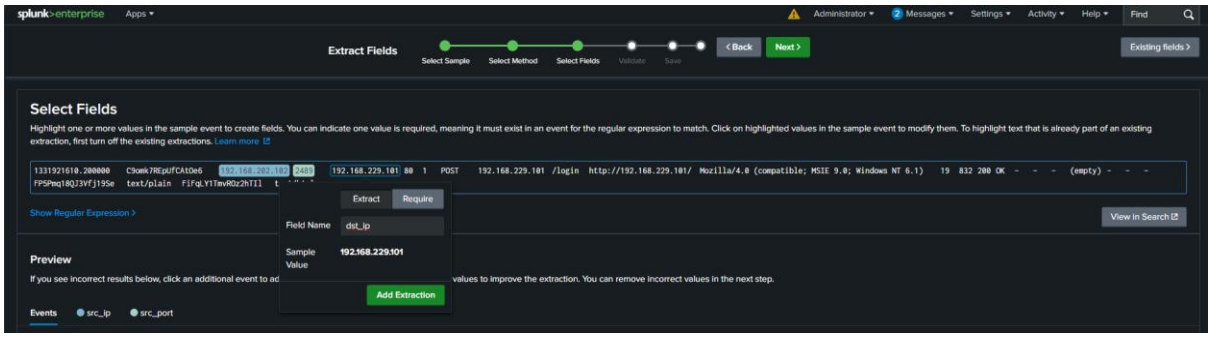
The raw HTTP logs required manual field extraction to isolate key components of the web requests and responses. This ensures that the data is structured and available for granular searching and statistical analysis.

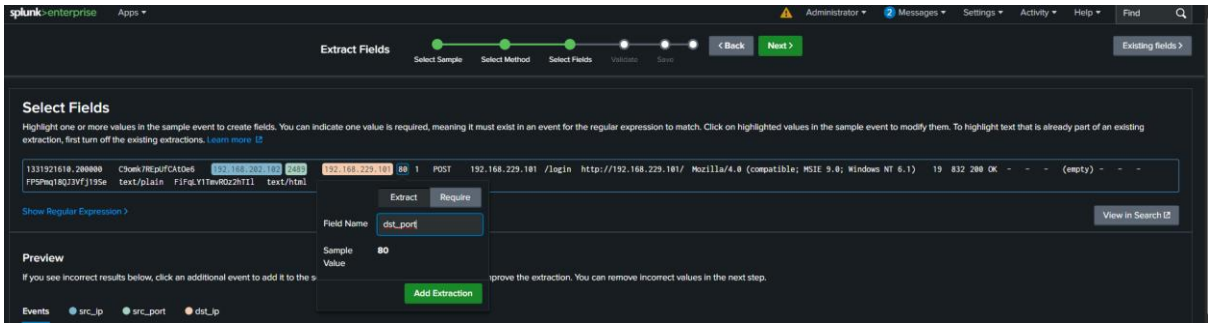
2.1 Extracted Fields

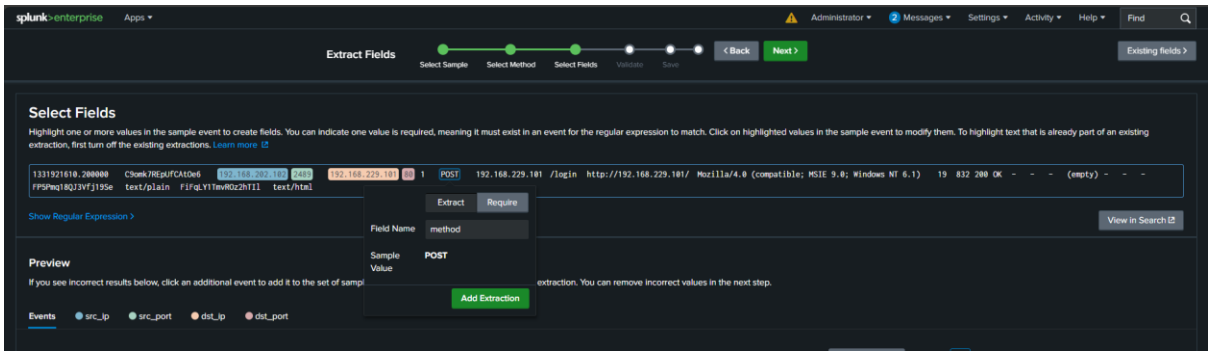
The following essential fields were manually extracted using the Splunk Field Extractor utility:

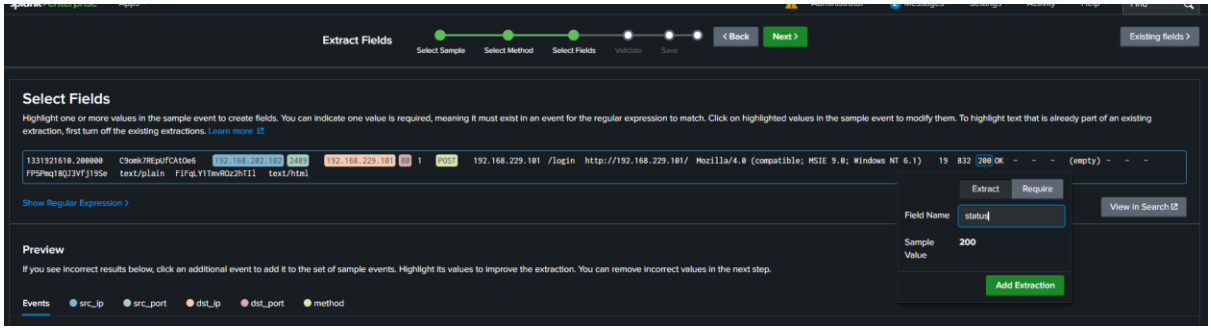
Field Name	Description
src_ip	The IP address of the client making the request.
dst_ip	The IP address of the destination web server.
dst_port	The destination port (e.g., 80 or 443).
method	The HTTP method used (e.g., GET, POST, HEAD).
status	The three-digit HTTP response code (e.g., 200, 404, 500).


127.0.0.1:8000/en-US/app/search/field_extractor?sid=1764841865.60
splunk>enterprise
Extract Fields
Select Sample
Select Method
Select Fields
Validate
Save
Back
Next
Existing fields
Select Fields
Highlight one or more values in the sample event to create fields. You can indicate one value is required, meaning it must exist in an event for the regular expression to match. Click on highlighted values in the sample event to modify them. To highlight text that is already part of an existing extraction, first turn off the existing extractions. Learn more
Extract
Require
Field Name
src_ip
Sample Value
192.168.202.102
Add Extraction


splunk>enterprise
Extract Fields
Select Sample
Select Method
Select Fields
Validate
Save
Back
Next
Existing fields
Select Fields
Highlight one or more values in the sample event to create fields. You can indicate one value is required, meaning it must exist in an event for the regular expression to match. Click on highlighted values in the sample event to modify them. To highlight text that is already part of an existing extraction, first turn off the existing extractions. Learn more
Show Regular Expression
View in Search
Extract
Require
Field Name
dst_ip
Sample Value
192.168.229.101
Add Extraction
Preview
Events
src_ip
src_port


splunk>enterprise
Extract Fields
Select Sample
Select Method
Select Fields
Validate
Save
Back
Next
Existing fields
Select Fields
Highlight one or more values in the sample event to create fields. You can indicate one value is required, meaning it must exist in an event for the regular expression to match. Click on highlighted values in the sample event to modify them. To highlight text that is already part of an existing extraction, first turn off the existing extractions. Learn more
Show Regular Expression
View in Search
Extract
Require
Field Name
dst_port
Sample Value
80
Add Extraction
Preview
Events
src_ip
src_port
dst_ip


splunk>enterprise
Extract Fields
Select Sample
Select Method
Select Fields
Validate
Save
Back
Next
Existing fields
Select Fields
Highlight one or more values in the sample event to create fields. You can indicate one value is required, meaning it must exist in an event for the regular expression to match. Click on highlighted values in the sample event to modify them. To highlight text that is already part of an existing extraction, first turn off the existing extractions. Learn more
Show Regular Expression
View in Search
Extract
Require
Field Name
method
Sample Value
POST
Add Extraction
Preview
Events
src_ip
src_port
dst_ip
dst_port


Extract Fields
Select Sample
Select Method
Select Fields
Validate
Save
< Back
Next >
Existing fields >
Select Fields
Highlight one or more values in the sample event to create fields. You can indicate one value is required, meaning it must exist in an event for the regular expression to match. Click on highlighted values in the sample event to modify them. To highlight text that is already part of an existing extraction, first turn off the existing extractions. Learn more
1331921610.200000 C9omk7REpUfCAtOe6 192.168.202.102 2488 192.168.229.101 80 1 POST 192.168.229.101 /login http://192.168.229.101/ Mozilla/4.0 (compatible; MSIE 9.0; Windows NT 6.1) 19 832 200 OK - - - (empty) - - -
FPSPmq18QJ3Vfj195e text/plain FiFqLYITmvROz2hTIl text/html
Show Regular Expression >
Extract
Require
Field Name
status
Sample Value
200
Add Extraction
View in Search
Preview
If you see incorrect results below, click an additional event to add it to the set of sample events. Highlight its values to improve the extraction. You can remove incorrect values in the next step.
Events
src_ip
src_port
dst_ip
dst_port
method

3.ANALYSIS OF WEB TRAFFIC BASELINES

3.1 Most Frequent URIs Queried

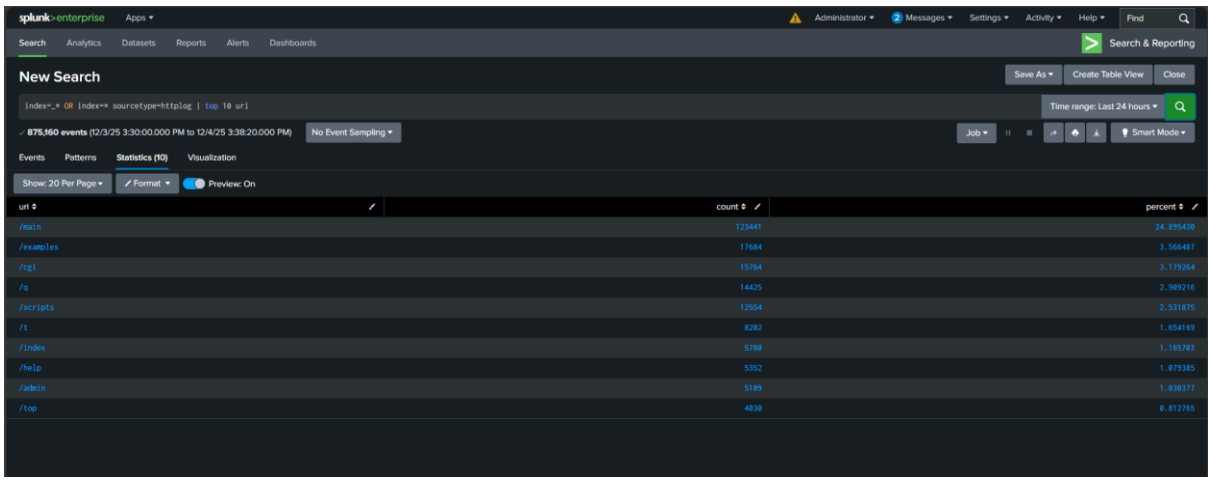
This analysis identifies which Uniform Resource Identifiers (URIs) or pages are most frequently accessed, indicating popular content or key application endpoints.

SPL Query:

index=* OR index=* sourcetype="httplog" | top 10 uri

Results (Partial):

The top URIs accessed include "/main", "/examples", and "/cgi", indicating heavy traffic toward these specific web resources.



3.2 Analysis of HTTP Methods

HTTP methods define the action requested by the client. Analyzing their frequency helps identify if traffic is primarily for browsing ("GET") or involves data submission ("POST").

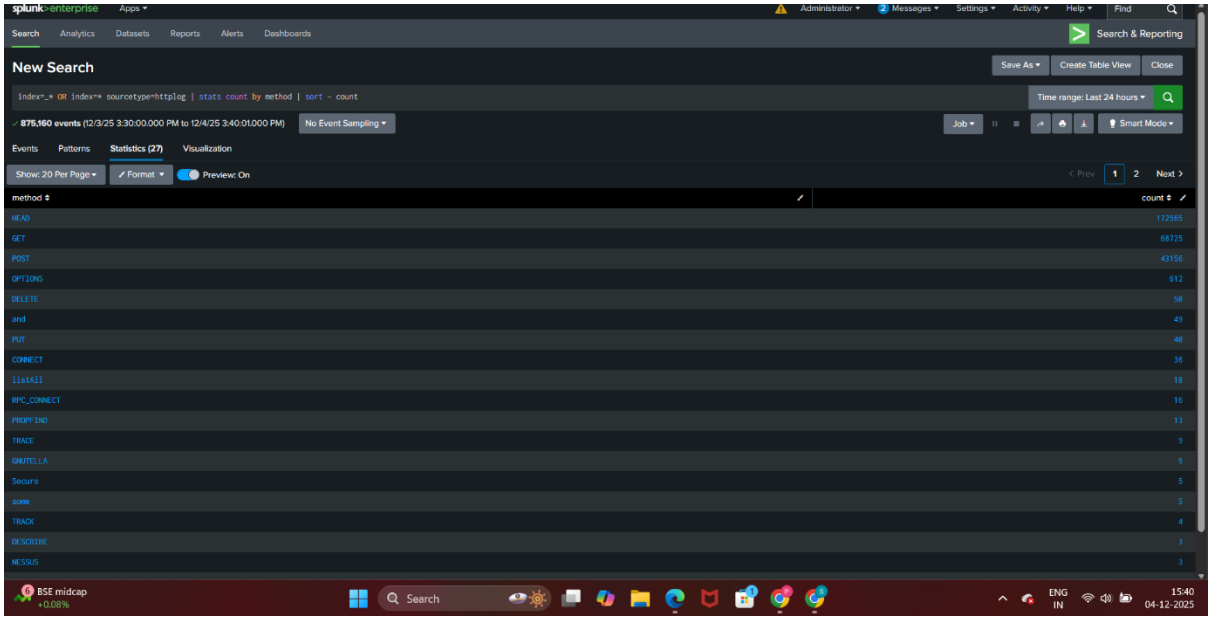
SPL Query:

```
index=* OR index=* sourcetype="httplog" | stats count by method | sort - count
```

Results (Partial):

The results confirm the primary traffic pattern:

- HEAD and GET are the most dominant methods (used for fetching content).
- POST follows, indicating user interactions or data submissions.
- The presence of less common methods like DELETE, PUT, and CONNECT warrants further security investigation, as these are often restricted.



3.3 Frequency of HTTP Status Codes

Analyzing status codes is the quickest way to gauge the health and accessibility of the web server.

SPL Query:

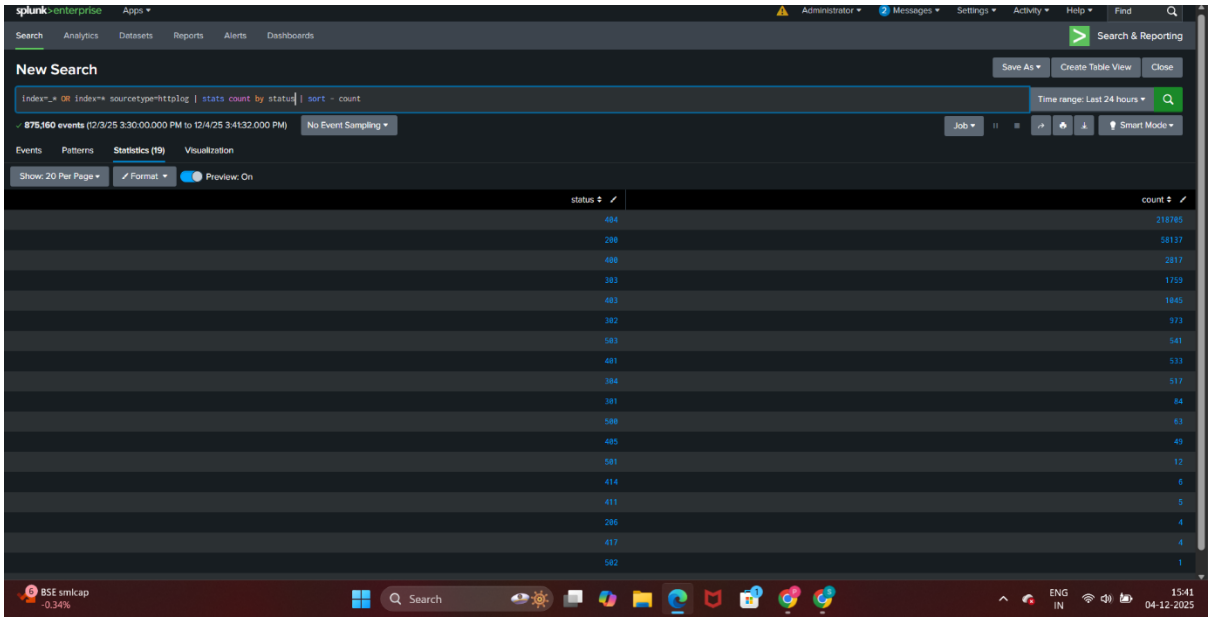
index=* OR index=sourcetype=httplog | stats count by status | sort - count

Results (Partial):

The results provide a distribution of response types:

- 404 (Not Found) is the most frequent code, suggesting numerous requests for non-existent pages or files. This requires further investigation.

- 200 (OK) is the second most frequent, indicating successful requests.
- 400 (Bad Request) and 302/303 (Redirection) are also prominent.



4.SECURITY ANALYSIS AND THREAT HUNTING

The most critical part of this project involves using the analyzed status codes to hunt for malicious activity, demonstrating my role as a security analyst.

4.1 Identifying Failed Authentication and Authorization Attempts.

Security professionals focus on 4xx errors, as they indicate client-side issues, often representing scanning or unauthorized access attempts.

A. Analysis of Top Failed URIs (4xx Errors)

This query identifies which specific pages or directories are being targeted by failed requests (e.g., directory brute-forcing).

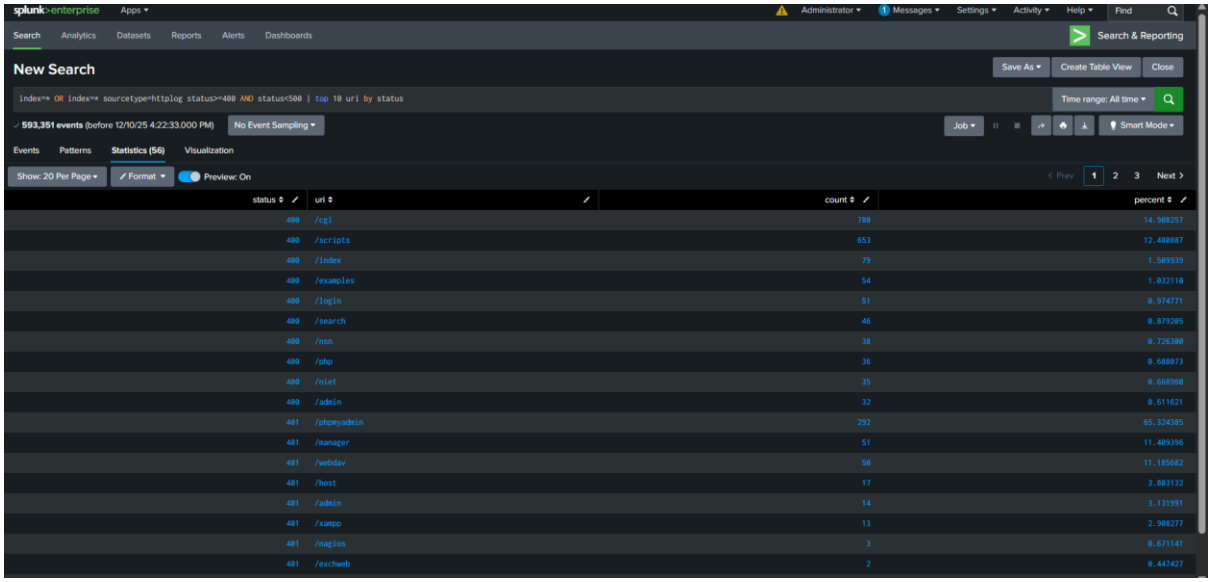
SPL Query:

index=* OR index=sourcetype=httplog status>=400 AND status<500 | top 10 uri by status

Security Interpretation:

The search reveals high request counts for sensitive or common administrative paths paired with error codes:

- 408 (Request Timeout): High volumes against "/cgi", "/scripts", and "/index" may indicate slow scanning or network congestion.
- 401 (Unauthorized): Failures against paths like "/manager", "/phpmyadmin", and "/admin" strongly suggest brute-forcing or directory enumeration attempts against restricted web administration interfaces.



B. Analyzing Specific Failed Login Attempts

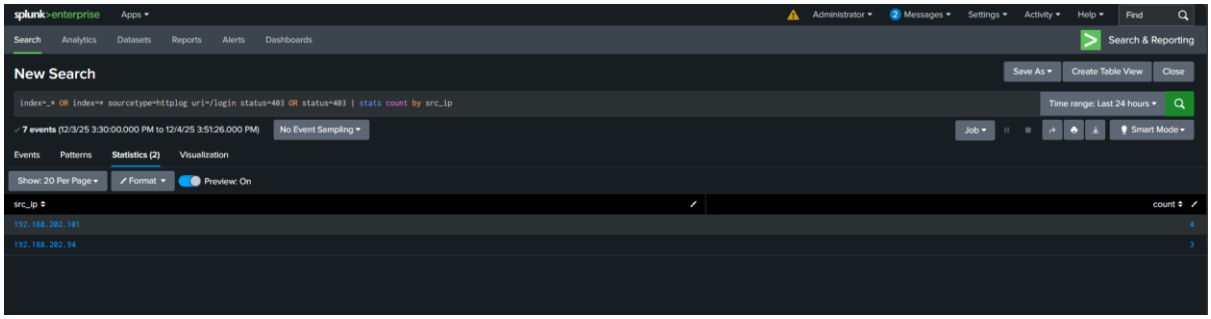
Targeting a specific failure status, such as 403 (Forbidden), combined with sensitive paths like "/login", helps pinpoint specific failed unauthorized access attempts.

SPL Query:

index=* OR index=* sourcetype=httplog uri=/login status=403 OR status=403 | stats count by src_ip

Results and Security Interpretation:

The query successfully isolated the two source IP addresses ("192.168.202.101" and "192.168.202.94") responsible for generating the 403 Forbidden errors against the "/login" path. This provides immediate, actionable intelligence for blocking or investigating these specific client hosts.



5.CONCLUSION

This project effectively demonstrated the use of Splunk for comprehensive HTTP log analysis. By successfully extracting fields, establishing a baseline of normal traffic patterns (Methods and URIs), and executing targeted security searches (analyzing 401/403 status codes), I proved the ability to detect common web application threats, including brute-force attempts and directory enumeration. This is a vital skill set for monitoring web infrastructure and securing applications in a real-world environment.