Cybersecurity

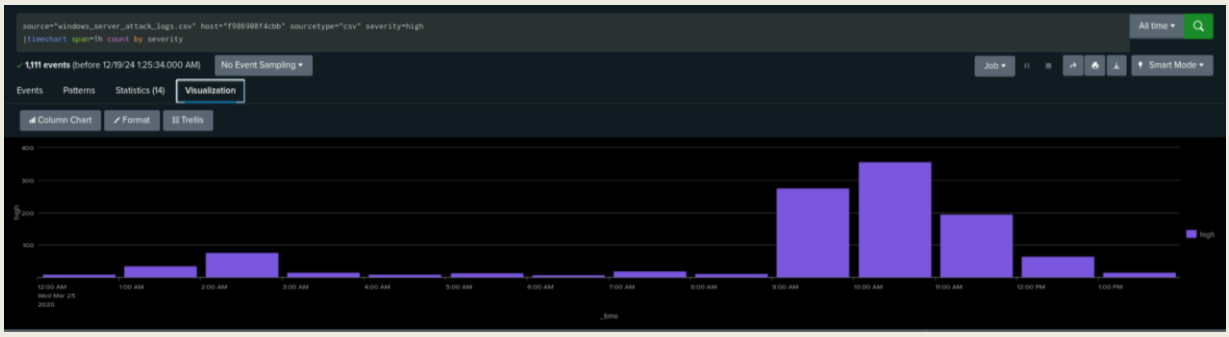
Project – Building Security Monitoring Environment in SPLUNK - Review Questions

Windows Server Log Questions

Report Analysis for Severity

- Did you detect any suspicious changes in severity?

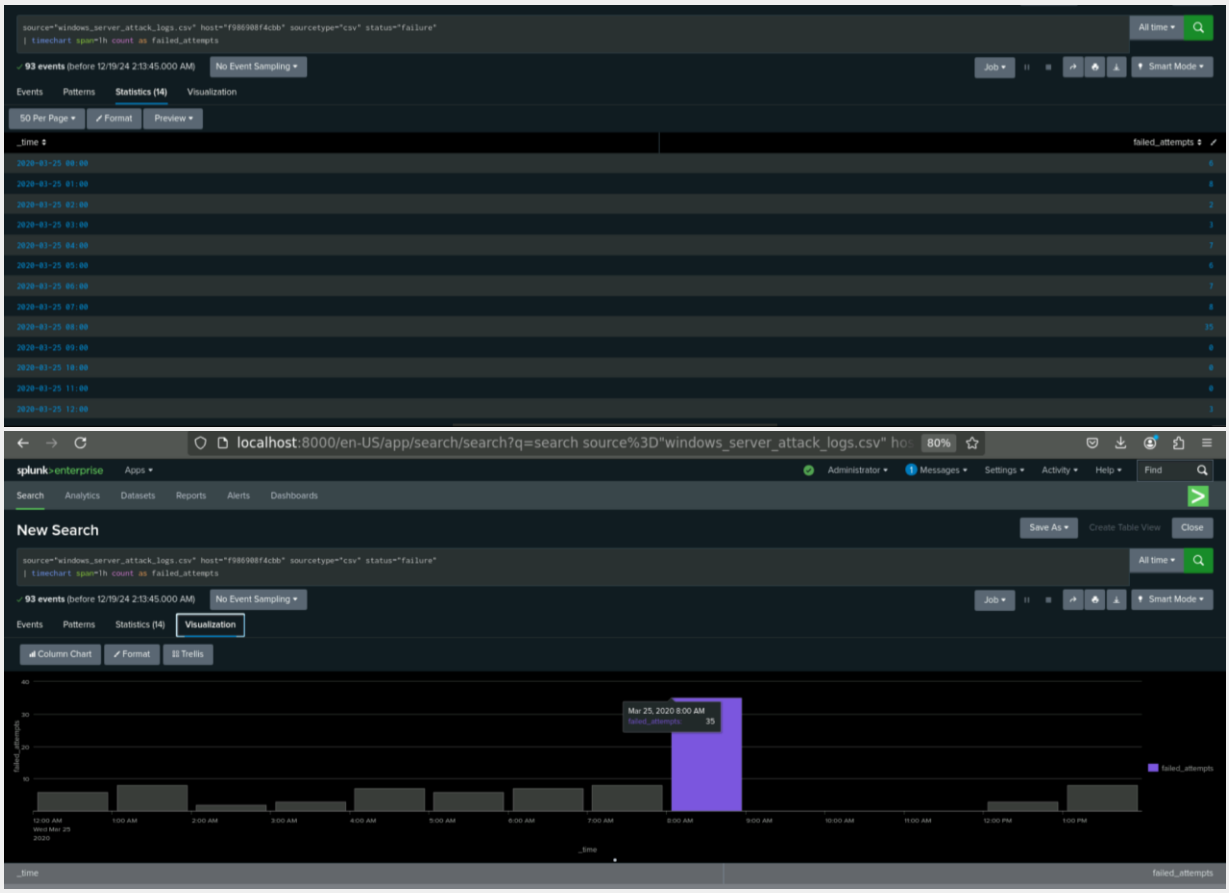
```
Yes, there is a noticeable spike in the data at 09:00 and 10:00 on March 25,
2020. Specifically:
   • 09:00: 276
   • 10:00: 357
These values are significantly higher compared to the rest of the data, with
the highest observed value of 357 at 10:00, following a lower value of 196
at 11:00. This sharp increase in severity between 09:00 and 10:00 stands out
when compared to the earlier hours, which all show values below 100, and
later hours, which remain lower in severity.
This sudden rise in numbers, especially within just one hour, could indicate
a shift in the underlying pattern or a potential anomaly that warrants
further investigation.

source="windows_server_attack_logs.csv" host="f986908f4cbb" sourcetype="csv"
severity=high
|timechart span=1h count by severity
```



Report Analysis for Failed Activities

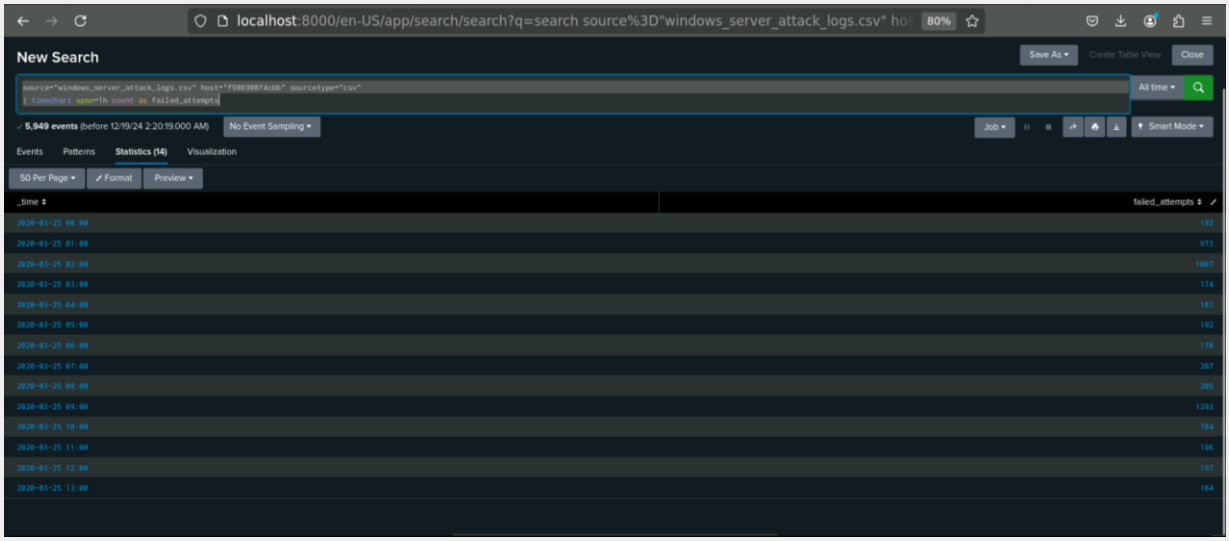
- Did you detect any suspicious changes in failed activities?

```
Yes, pls see the pics and analysis after pics.

source="windows_server_attack_logs.csv" host="f986908f4cbb" sourcetype="csv"
status="failure"
| timechart span=1h count as failed_attempts
```



With slightly different search:::

source="windows_server_attack_logs.csv" host="f986908f4cbb" sourcetype="csv" | timechart span=1h count as failed_attempts



Detailed Analysis:

1. **Spike at 01:00 (973 failures)**:
   - The count of **973** failed attempts at **01:00** is much higher than the counts seen earlier (e.g., **192** at **00:00**). This suggests an increase in failed activities around this time.
   - While not as extreme as the spike at **09:00**, this is still a significant increase when compared to the hours before it (192 -> 973). It could represent an isolated event or a rise in failed login attempts, perhaps due to a specific user or system issue.
2. **Spike at 02:00 (1007 failures)**:
   - The **1007** failed attempts at **02:00** are again notably higher than the previous hours (**973** at **01:00**). The sustained high number of failures across both **01:00 and 02:00** suggests a possible ongoing issue, such as an attack or misconfiguration, which persisted for several hours.
   - The fact that this spike is not a one-time occurrence (i.e., it continues into **02:00**) further strengthens the idea that this could be part of an unusual or suspicious event.
3. **The next few hours (03:00 to 08:00)**:
   - After the spikes at **01:00 and 02:00**, the failed attempts drop significantly to **174** at **03:00** and stay low throughout the next few hours. This sudden drop after two high spikes suggests that something either mitigated the issue (like a firewall or rate-limiting mechanism kicking in) or the event naturally subsided.
4. **09:00 Spike (1293 failures)**:
   - The massive spike at **09:00** (1293 failed attempts) is the most dramatic increase in the dataset and is the most suspicious. It far exceeds the spikes at **01:00** and **02:00**.

Conclusion:

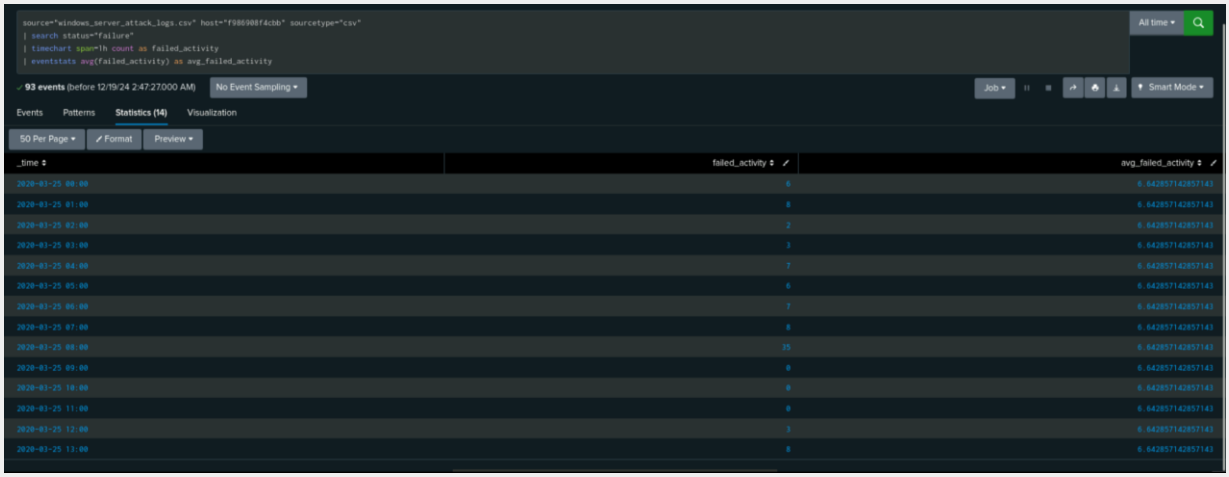
- **Suspicious Activity Detected**:
  - **01:00 and 02:00**: Both show significant increases in failed attempts (973 and 1007), suggesting a potential issue, such as

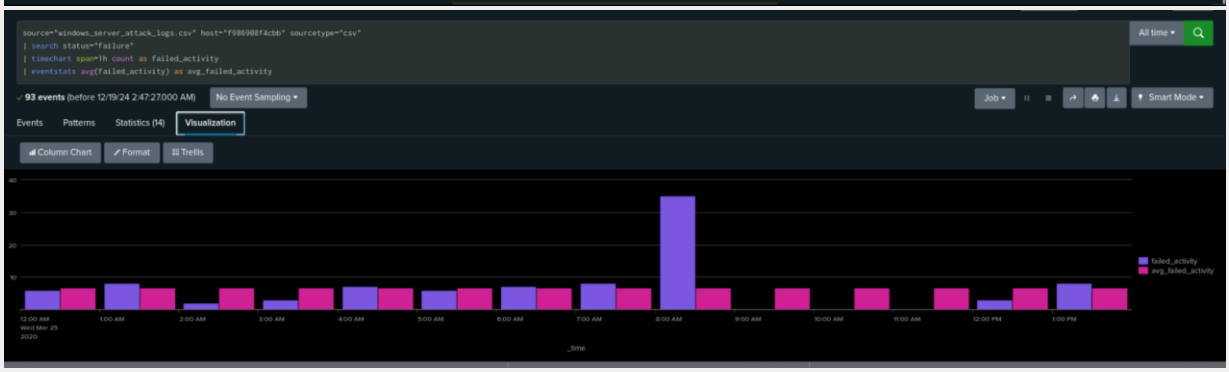
```
            an ongoing brute-force attack or system misconfiguration, which
            lasted over multiple hours.
        ○   09:00: The largest spike of 1293 failed attempts is very unusual
            and stands out even more compared to 01:00 and 02:00. It could
            indicate a more serious or targeted attack (e.g., a burst of
            brute-force login attempts).
    • The combination of multiple spikes in failed activities at 01:00,
      02:00, and 09:00 shows that there were periods of abnormal activity
      during the day. This could suggest suspicious behavior such as brute-
      force attacks, unauthorized access attempts, or service
      misconfigurations.
Final Answer:
Yes, suspicious changes in failed activities were detected. While there were
smaller spikes at 01:00 and 02:00 (973 and 1007 failures), the most notable
and suspicious spike occurred at 09:00 (1293 failures). These spikes suggest
that there may have been a brute-force attack or another form of
unauthorized access attempt, requiring further investigation.
```

Alert Analysis for Failed Windows Activity

- Did you detect a suspicious volume of failed activity?

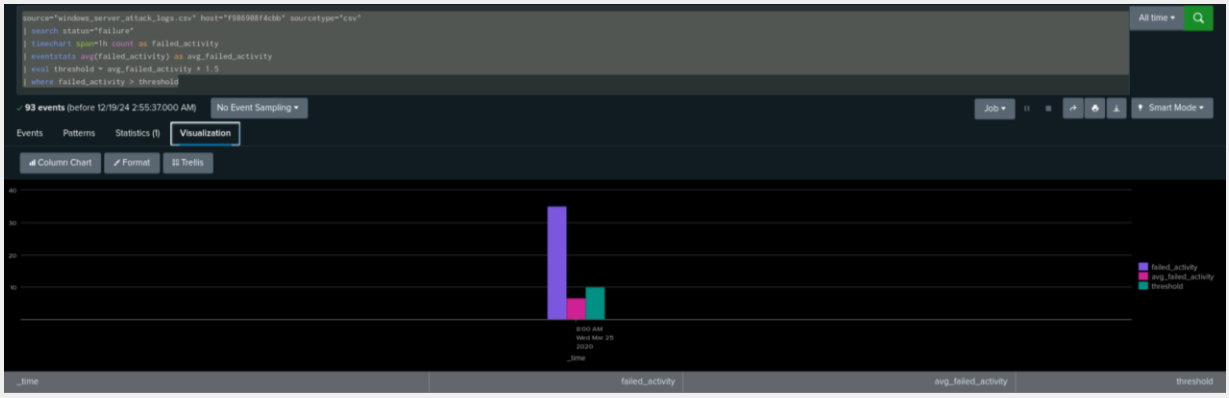
```
   Yes, the query is designed to detect a suspicious volume of failed
activity by comparing the number of failed events in each hour
(failed_activity) against the threshold, which is calculated as 1.5 times
the average of failed activity.
   The threshold is dynamic and based on the average number of failed events
over the entire time period.

Below search will establish the baseline::
source="windows_server_attack_logs.csv" host="f986908f4cbb" sourcetype="csv"
| search status="failure"
| timechart span=1h count as failed_activity
| eventstats avg(failed_activity) as avg_failed_activity
```





Alert based behavior below:::

```
source="windows_server_attack_logs.csv" host="f986908f4cbb" sourcetype="csv"
| search status="failure"
| timechart span=1h count as failed_activity
| eventstats avg(failed_activity) as avg_failed_activity
| eval threshold = avg_failed_activity * 1.5
| where failed_activity > threshold
```



- If so, what was the count of events in the hour(s) it occurred?

```
   The search will only show hours where the count of failed activities
(failed_activity) exceeded the threshold.
   For example, based on the data you've given:
   • If at 2020-03-25 08:00, the failed_activity is 35, and the average
     failed activity is 6.64, the threshold will be 6.64 * 1.5 = 9.96.
   • Since 35 > 9.96, this will trigger an alert for the 2020-03-25 08:00
     hour.
   So, the count of events in that hour would be 35.
```

- When did it occur?

```
From the results, the suspicious volume of failed activity occurred at 2020-
03-25 08:00.
```

- Would your alert be triggered for this activity?

```
Yes, the alert will be triggered because the failed activity count (35) is
greater than the threshold (9.96).
```

- After reviewing, would you change your threshold from what you previously selected?

```
After reviewing, for this volume of failures what is typical for the system,
we might adjust the threshold, either increasing or decreasing the
```

```
multiplier based on our tolerance for failure events. However, the current
threshold is aptly selected and will be kept same unless data trend changes.
```

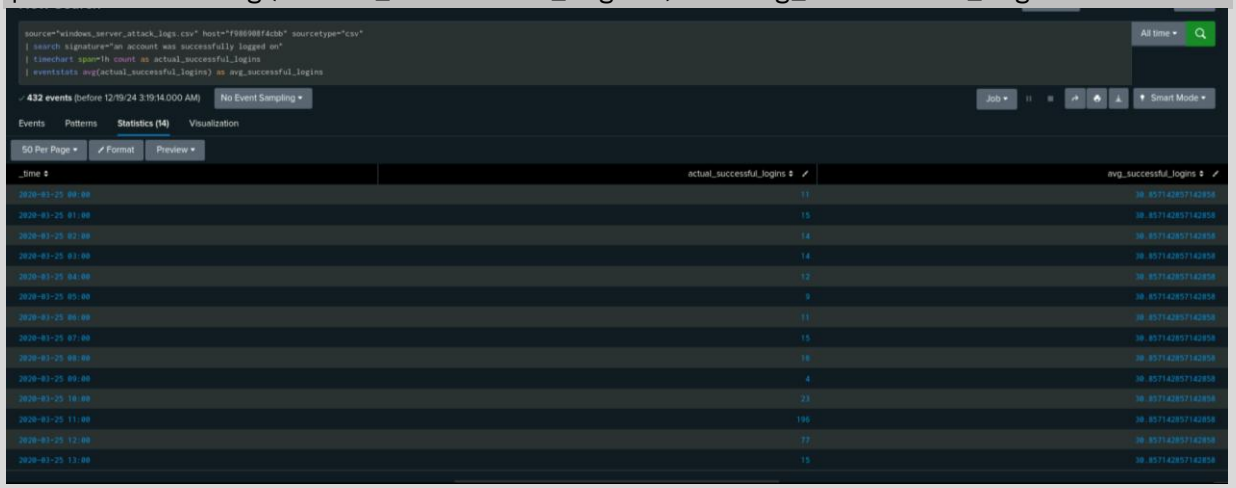
Alert Analysis for Successful Logins

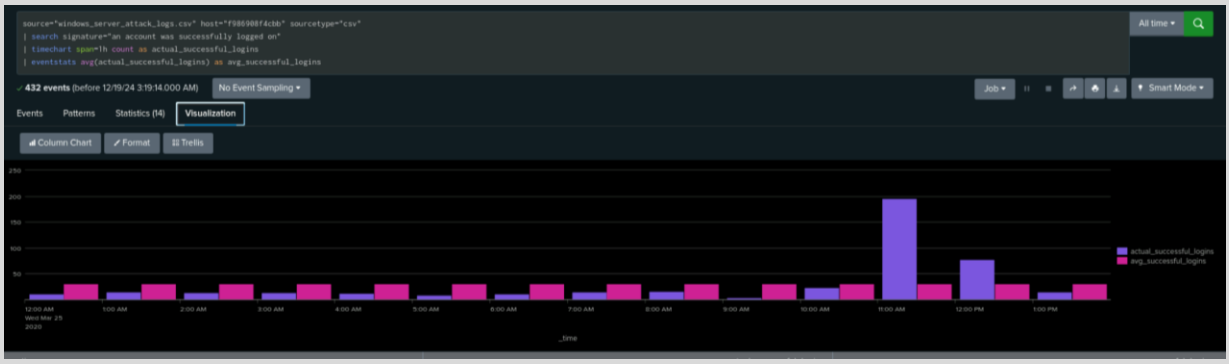
- Did you detect a suspicious volume of successful logins?

Yes, see the pics and analysis below:

This search establishes the baseline:

```
source="windows_server_attack_logs.csv" host="f986908f4cbb" sourcetype="csv"
| search signature="an account was successfully logged on"
| timechart span=1h count as actual_successful_logins
| eventstats avg(actual_successful_logins) as avg_successful_logins
```

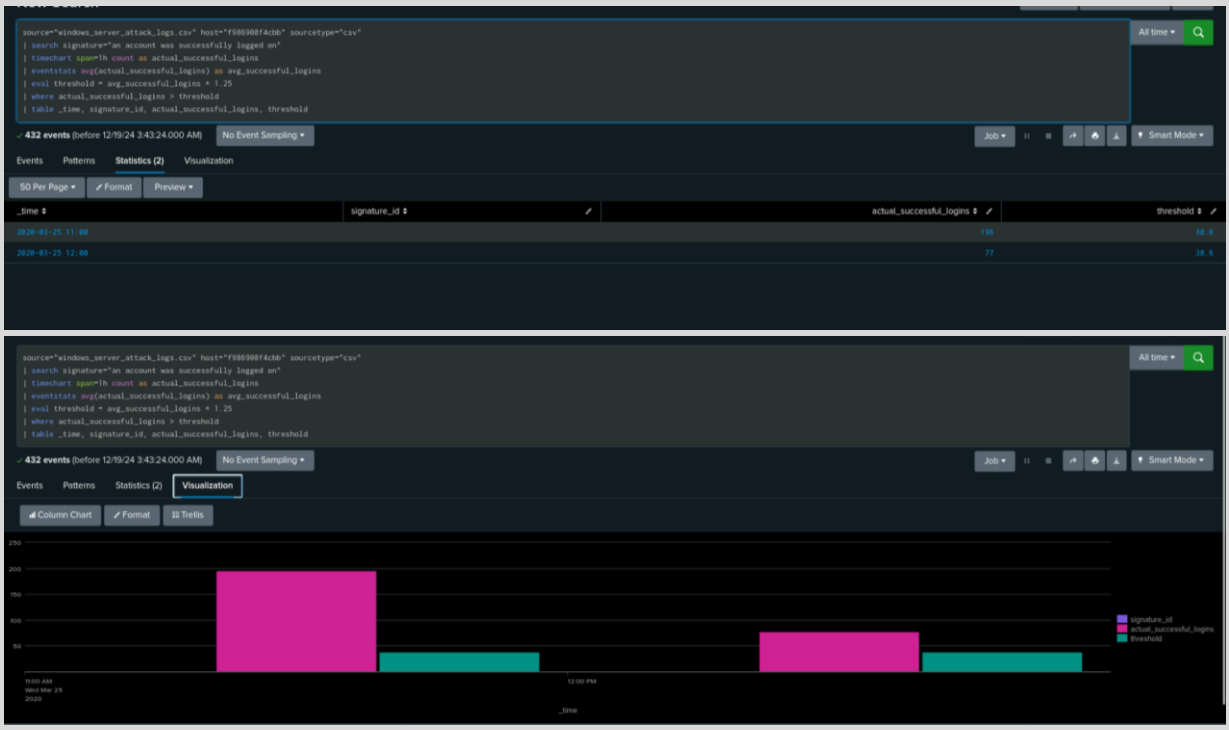




Alert based search::

```
source="windows_server_attack_logs.csv" host="f986908f4cbb" sourcetype="csv"
| search signature="an account was successfully logged on"
| timechart span=1h count as actual_successful_logins
| eventstats avg(actual_successful_logins) as avg_successful_logins
```

```
| eval threshold = avg_successful_logins * 1.25
| where actual_successful_logins > threshold
| table _time, signature_id, actual_successful_logins, threshold
```

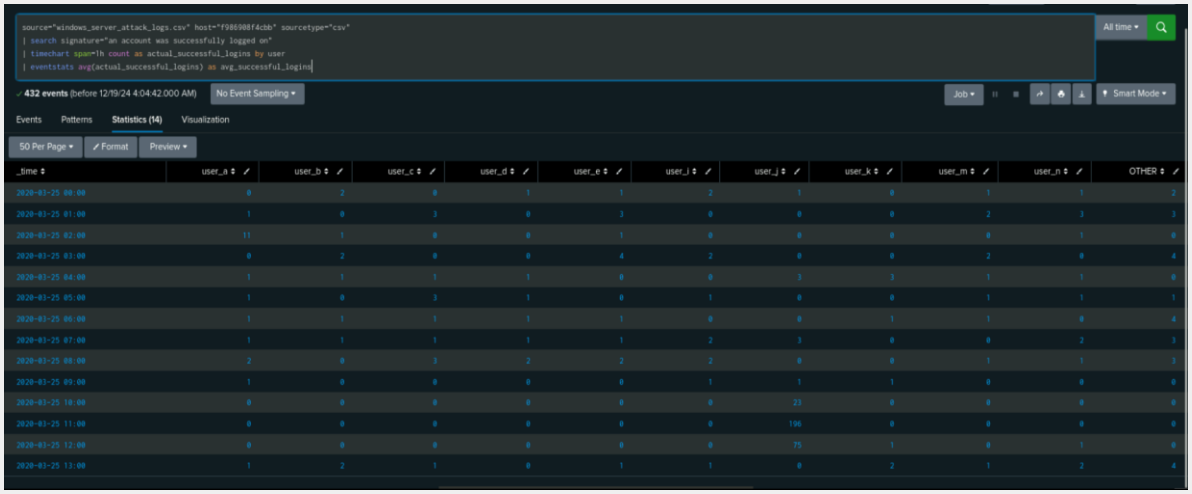


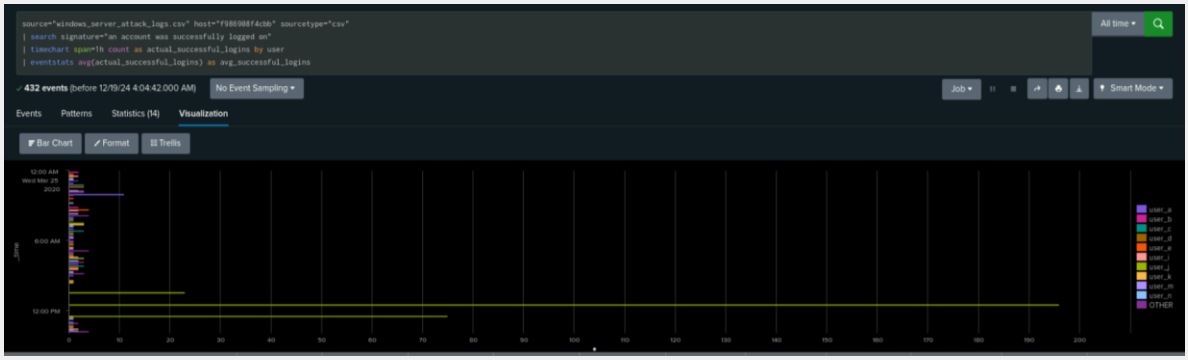
- If so, what was the count of events in the hour(s) it occurred?

196 times at 11AM on March 25, 2020

- Who is the primary user logging in?

Primary user logging in is user_j, see pics below, user_j logged in 196 times at 11AM on March 25, 2020





- When did it occur?

11AM on March 25, 2020

- Would your alert be triggered for this activity?

yes

- After reviewing, would you change your threshold from what you previously selected?

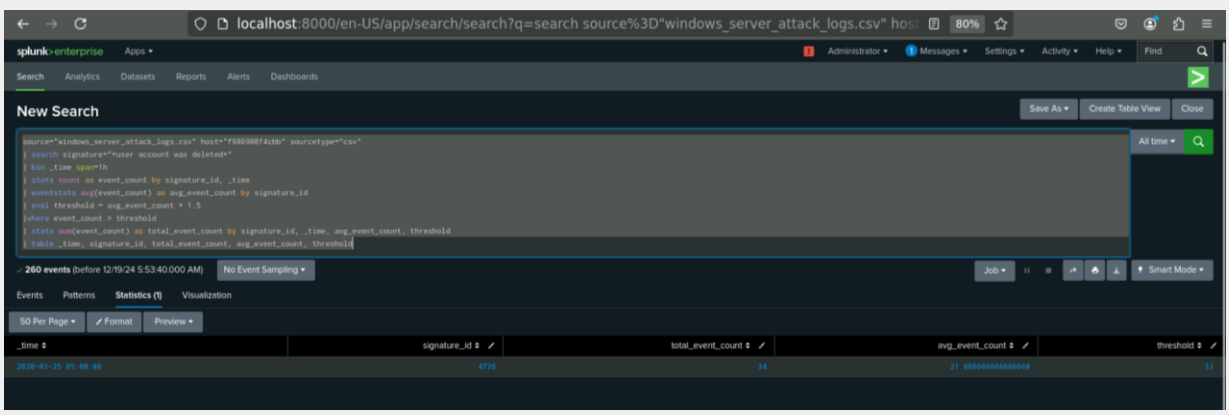
Not at the moment but it can be, alert is dynamic though but threshold can be modified.

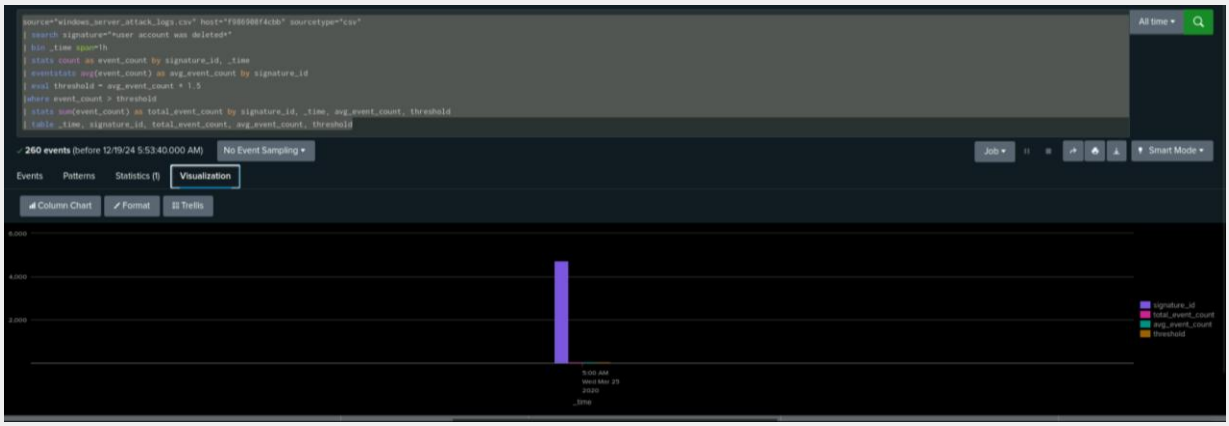
Alert Analysis for Deleted Accounts

- Did you detect a suspicious volume of deleted accounts?

```
Yes, under signature_id 4726 based on the alert criteria, spike was observed
in account deletions.At 5AM 34 account deletions were detected against a
baseline of 33.
```

```
source="windows_server_attack_logs.csv" host="f986908f4cbb" sourcetype="csv"
| search signature="*user account was deleted*"
| bin _time span=1h
| stats count as event_count by signature_id, _time
| eventstats avg(event_count) as avg_event_count by signature_id
| eval threshold = avg_event_count * 1.5
|where event_count > threshold
| stats sum(event_count) as total_event_count by signature_id, _time,
avg_event_count, threshold
| table _time, signature_id, total_event_count, avg_event_count, threshold
```



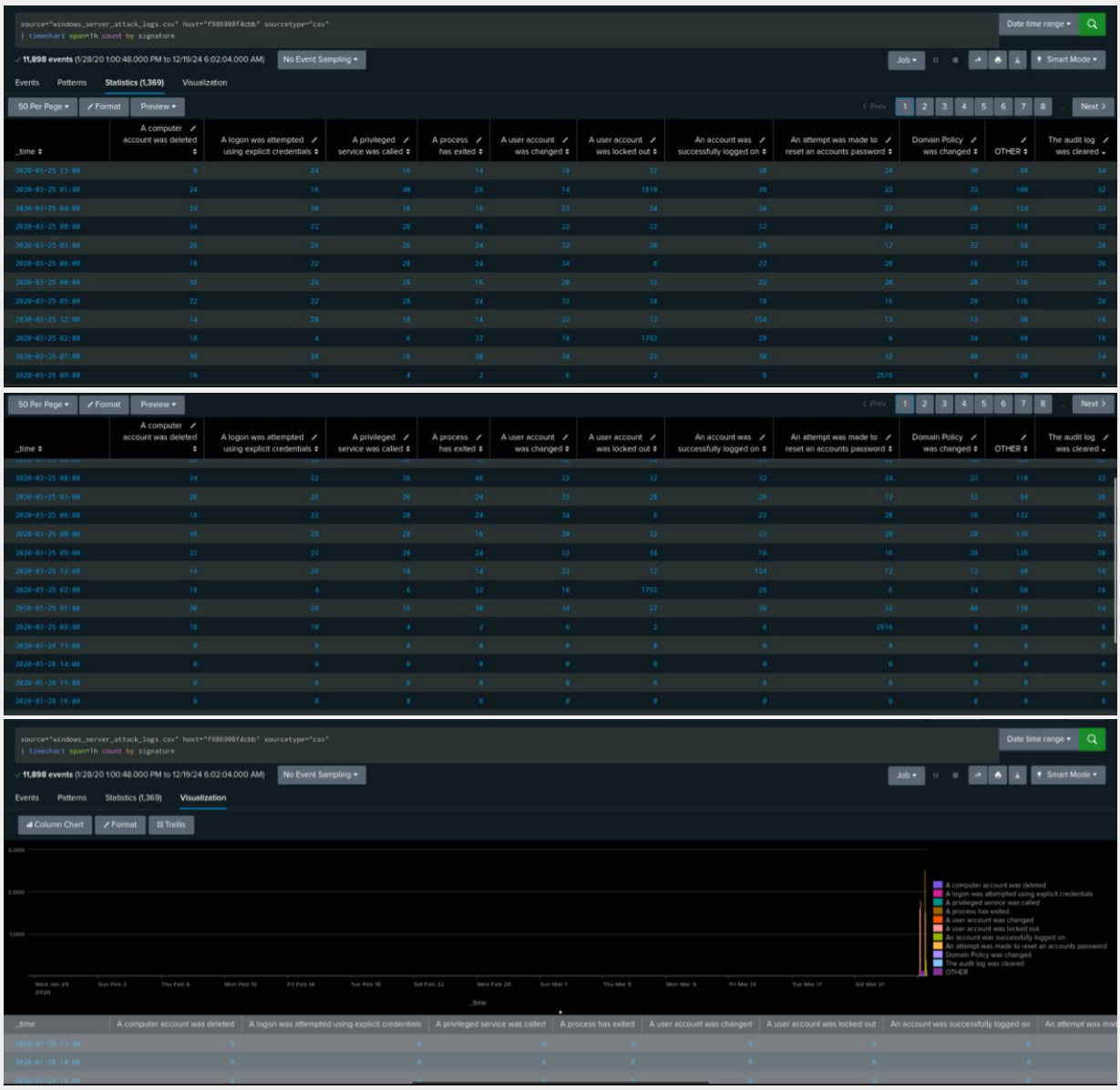


Dashboard Analysis for Time Chart of Signatures

- Does anything stand out as suspicious?

Yes,
count of 2516 at 9AM was seen under "account password reset"
count of 1792 at 2AM and 1610 @1AM "under account locked out"
count of 154 at 12PM "successful logon"

```
source="windows_server_attack_logs.csv" host="f986908f4cbb" sourcetype="csv"
| timechart span=1h count by signature
```



- What signatures stand out?

"account password reset", "under account locked out" and "successful logon"

- What time did it begin and stop for each signature?

count of 2516 at 9AM was seen under "account password reset"
count of 1792 at 2AM and 1610 @1AM "under account locked out"
count of 154 at 12PM "successful logon"

- What is the peak count of the different signatures?

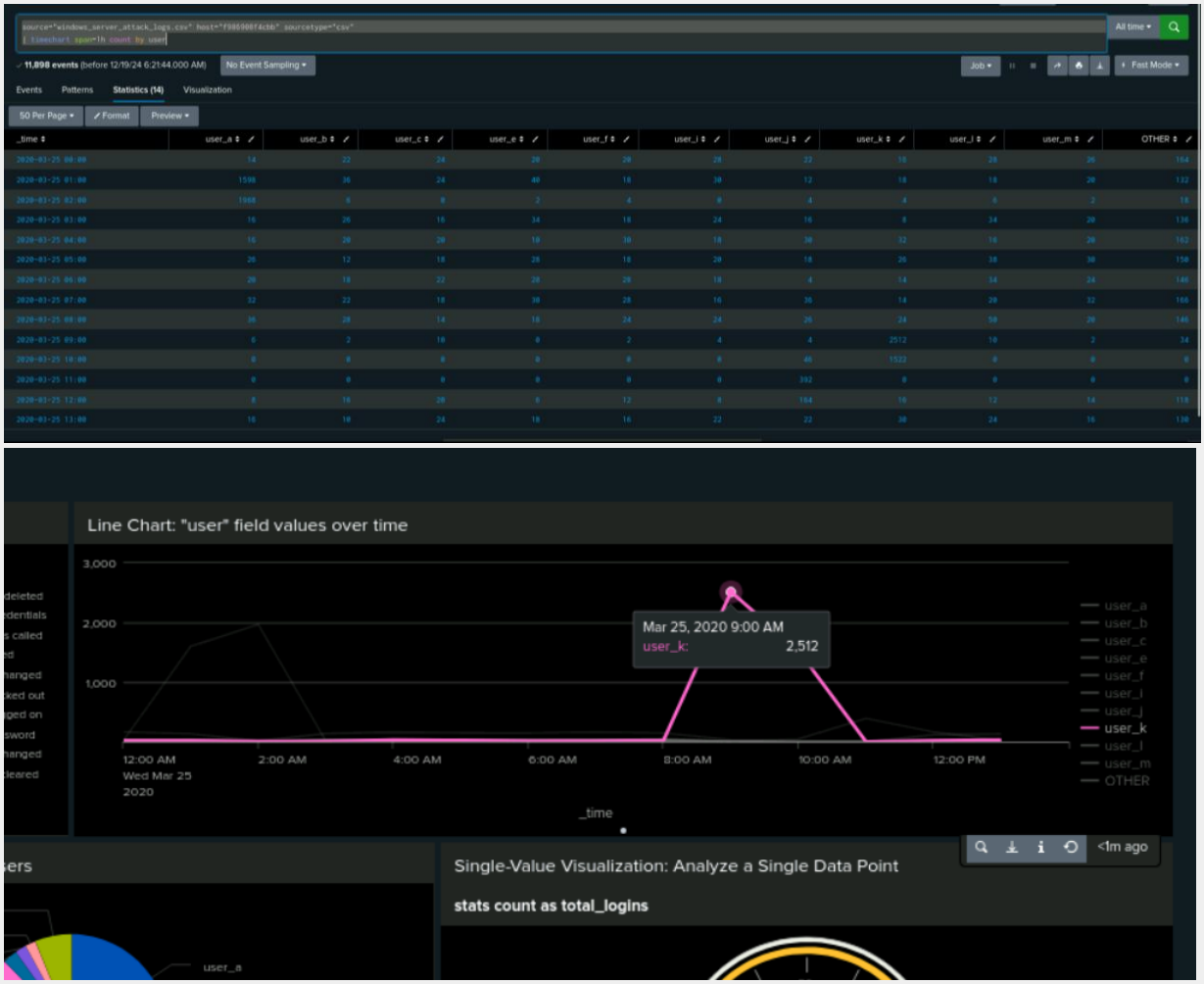
count of 2516 at 9AM was seen under "account password reset"
count of 1792 at 2AM and 1610 @1AM "under account locked out"
count of 154 at 12PM "successful logon"

Dashboard Analysis for Users

- Does anything stand out as suspicious?

Yes, there is a significant anomaly in the data around **2020-03-25 09:00** and **2020-03-25 10:00**. Specifically, **user_k** shows a drastic spike in activity at 09:00, with a count of **2512**. This is an unusually high number compared to other users in the dataset. There is also activity for **user_k** at 10:00 with **1522** events, which is notably high compared to the usual counts for other users.

```
source="windows_server_attack_logs.csv" host="f986908f4cbb" sourcetype="csv"
| timechart span=1h count by user
```



- Which users stand out?

```
User_k stands out with an abnormally high count of 2512 at 09:00 and 1522 at
10:00. User_a: Peak at 1968 at 02:00; User_j: Peak at 392 at 11AM
```

- What time did it begin and stop for each user?

```
User_k stands out with an abnormally high count of 2512 at 09:00 and 1522 at
10:00. User_a: Peak at 1968 at 02:00; User_j: Peak at 392 at 11AM
```

- What is the peak count of the different users?

User_k: The peak count is 2512 at 09:00.
For other users, the peak counts are much lower:

- User_a: Peak at 1968 at 02:00
- User_j: Peak at 392 at 09:00

Remaining users can be seen from pics above, counts are much lower.

Dashboard Analysis for Signatures with Bar, Graph, and Pie Charts

- Does anything stand out as suspicious?

Yes, there is a **suspicious spike** in events related to **signature 4726** (user account was deleted). under signature_id 4726 based on the alert criteria, spike was observed in account deletions.At 5AM 34 account deletions were detected against a baseline of 33.

Also,
count of 2516 at 9AM was seen under "account password reset"
count of 1792 at 2AM and 1610 @1AM "under account locked out"
count of 154 at 12PM "successful logon"

- Do the results match your findings in your time chart for signatures?

Yes, the results from the **time chart for signatures** align with the findings from the dashboard analysis. Both indicate elevated event counts for **signature 4726 (user account was deleted)**, and for "account password reset", "under account locked out" and "successful logon"

Dashboard Analysis for Users with Bar, Graph, and Pie Charts

- Does anything stand out as suspicious?

Yes, **User_k** stands out with a **massive spike** in activity at **09:00** and **10:00**, with counts of **2512** and **1522**, respectively. These values are abnormally high compared to other users, which could indicate suspicious behavior, such as unauthorized access or unusual login patterns.

- Do the results match your findings in your time chart for users?

Yes, the results from the **time chart for users** are consistent with the findings from the dashboard analysis. Both highlight the abnormal spikes in activity for **User_k** at **09:00** and **10:00**, confirming the suspicion of unusual activity during those times. Other users show relatively stable and typical activity.

Dashboard Analysis for Users with Statistical Charts

- What are the advantages and disadvantages of using this report, compared to the other user panels that you created?

Dashboard Analysis for Users with Statistical Charts
Advantages:

1. **Visual Representation**: Statistical charts, such as bar, pie, or line charts, provide a clear and immediate visual representation of the data, making it easier to identify trends, anomalies, and patterns in user activity.
2. **Ease of Comparison**: With statistical charts, you can quickly compare user activities side by side, such as login attempts or account deletions, which may be difficult to spot in raw data tables.
3. **Quick Detection of Outliers**: Spikes in user activity or anomalies (like unusual login attempts) can be visually identified, helping to detect potential security threats or suspicious behavior more efficiently.
4. **Interactive**: Many dashboards with statistical charts allow users to interact with the data, filtering, drilling down, or zooming into specific time periods or user activities, enhancing the user experience.
5. **Trends Over Time**: Statistical charts such as line charts or time-based visualizations allow easy observation of trends over time, helping to detect shifts in user behavior that might indicate a problem or anomaly.

Disadvantages:

1. **Loss of Detail**: Statistical charts may abstract or generalize data too much, omitting some finer details that might be important. This can make it difficult to pinpoint the exact cause of a problem without drilling down into raw data.
2. **Over-Simplification**: Complex data may be oversimplified in visual charts, leading to potential misinterpretation. Important nuances in user activity might be lost if only high-level charts are used.

3. **Limited Context**: Without proper contextual information (e.g., user roles, types of actions), the visual representation of the data may not give a complete understanding of user activity and its relevance to specific security or operational goals.
4. **Dependence on Pre-Set Filters**: The chart's value is often dependent on pre-set filters and aggregations (e.g., hourly, daily), which might not always align with the exact data granularity needed for specific analyses.
5. **Potential Misleading Trends**: In cases where the data is heavily influenced by outliers or skewed distributions, the chart may give a misleading impression of normal user behavior, requiring deeper investigation.

Comparison to Other User Panels:

- **User panels** based on raw data or tabular reports provide more granular, detailed insights, which can be essential when you need to investigate specific users or actions in-depth. These panels allow users to see exact numbers and individual event logs, which is often more useful for troubleshooting or understanding the context behind user behavior.
- **Statistical charts** offer quick insights and trend analysis, making them ideal for spotting high-level patterns, but might not provide the depth necessary for understanding specific anomalies or investigating incidents at a granular level.

In summary, **statistical charts** are excellent for visualizing trends and identifying anomalies quickly, but **raw user panels** or tables are more suitable when a detailed, precise understanding of user activity is required. Combining both approaches can give a balanced and comprehensive view.

AI and Google consulted for portions of this report.

Apache Web Server Log Questions

Report Analysis for Methods

- Did you detect any suspicious changes in HTTP methods? If so, which one?

Analysis of HTTP Methods in Apache Attack Logs:

- **Suspicious Changes Detected**: While **GET** requests (3157) are the most common method, the **POST** method (1324) stands out as potentially

suspicious due to its significant volume. An unusually high number of **POST** requests may indicate potential attempts to exploit server vulnerabilities, such as form submissions, SQL injection, or cross-site scripting (XSS) attacks.

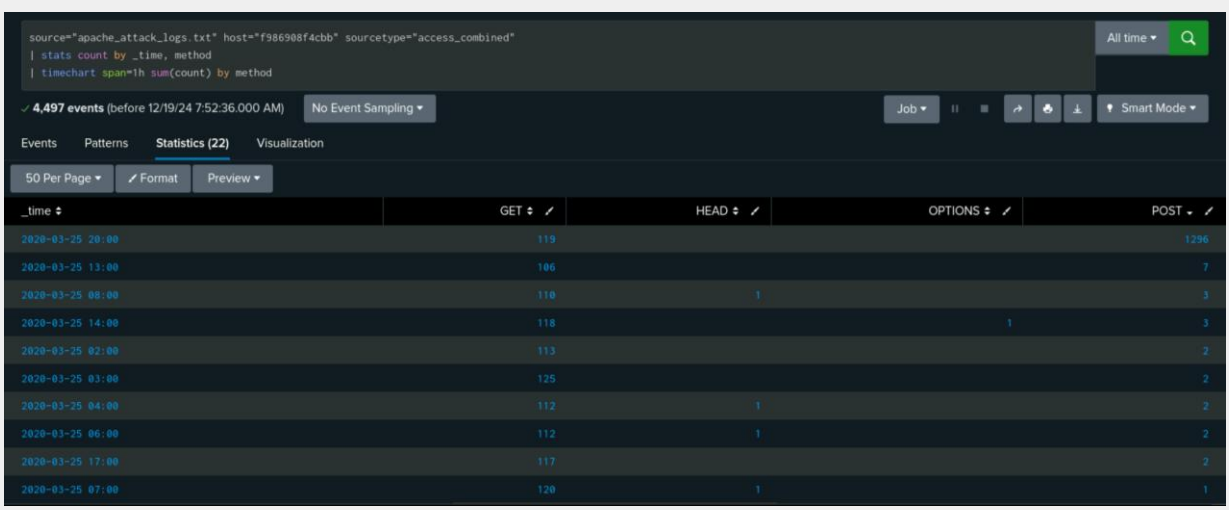
- **High GET Request Count**: The volume of **GET** requests (3157) is expected in normal web traffic, as **GET** is typically used to retrieve resources. However, if there are sudden spikes in **GET** traffic at certain times, it could be an indicator of reconnaissance activity or DDoS (Distributed Denial of Service) attacks.
- **Low HEAD and OPTIONS Counts**: The **HEAD** (15) and **OPTIONS** (1) methods are relatively low, which is normal since these methods are generally used for specific tasks like checking headers or probing for available methods. A spike in these could indicate probing or scanning attempts by attackers.

Conclusion:

While **GET** is expected to be the most common method, an unusual volume of **POST** requests may signal potentially suspicious activity, such as data manipulation or exploitation attempts. Regular monitoring and further investigation into the nature of **POST** requests could help in detecting malicious behavior.

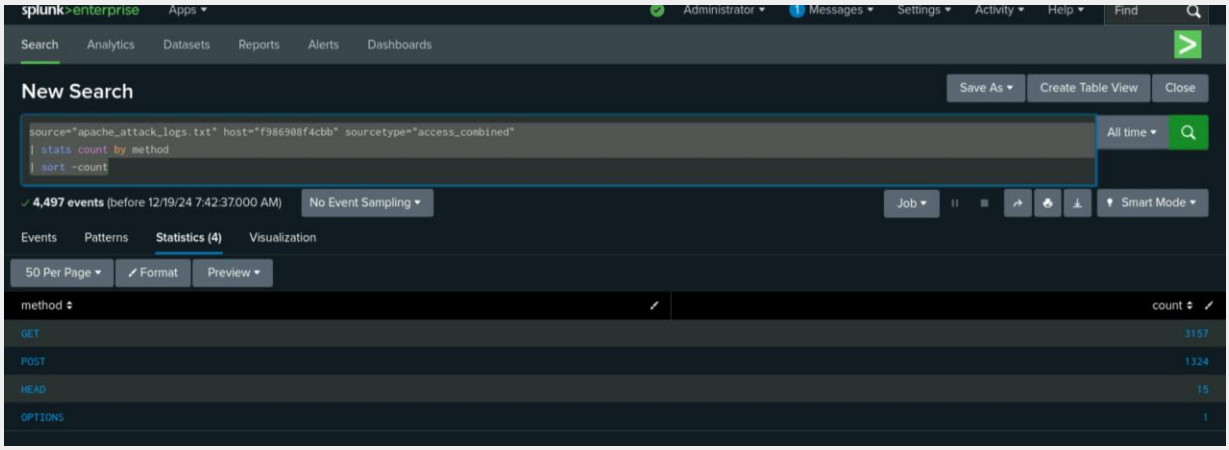
Further, drilling clearly shows a possible sql injection or bruteforce attack as shown in the pic below. Attack is visible via HTTP Post method which shows heavy departure from normal with a count of 1296 @2000 hrs on March 25, 2020.

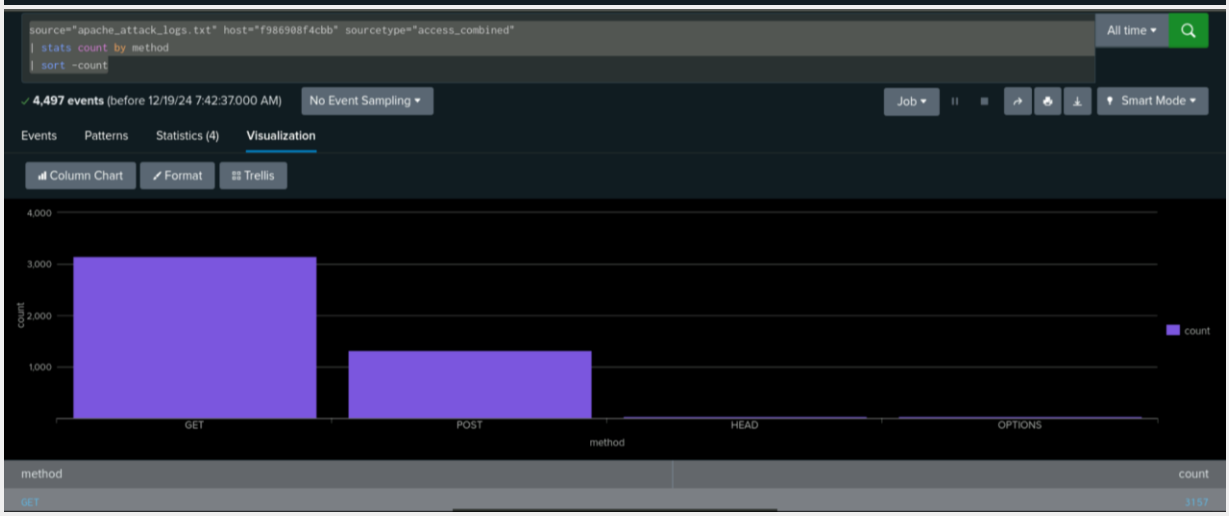
```
source="apache_attack_logs.txt" host="f986908f4cbb"
sourcetype="access_combined"
| stats count by _time, method
| timechart span=1h sum(count) by method
```



_time	GET	HEAD	OPTIONS	POST
2020-03-25 20:00	119			1296
2020-03-25 13:00	106			7
2020-03-25 08:00	110	1		3
2020-03-25 14:00	118		1	3
2020-03-25 02:00	113			2
2020-03-25 03:00	125			2
2020-03-25 04:00	112	1		2
2020-03-25 06:00	112	1		2
2020-03-25 17:00	117			2
2020-03-25 07:00	120	1		1

```
source="apache_attack_logs.txt" host="f986908f4cbb"
sourcetype="access_combined"
| stats count by method
| sort -count
```





- What is that method used for?

The **POST** method is one of the most common HTTP methods used in web communication. It is primarily used for sending data to a server to be processed or stored, usually as part of an action or transaction. Here are the key purposes and uses of the **POST** method:

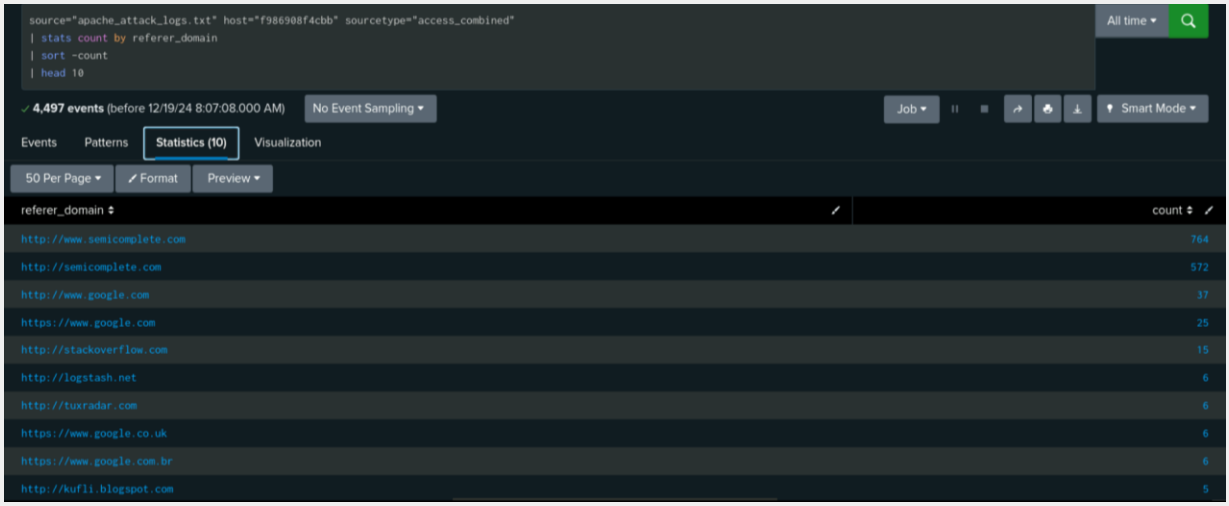
1. **Submitting Form Data**:
    - The **POST** method is commonly used in HTML forms to submit data from a web page to a server. For example, when a user enters information (e.g., username, password, email) in a form and submits it, the data is sent via the **POST** method to the server for processing.
2. **Creating or Modifying Resources**:
    - **POST** is used to create or modify resources on the server. When a client (browser or application) sends data to the server using **POST**, it can result in the creation of new entries in a database or updates to existing records. For example, when submitting a new post to a social media platform, **POST** is used to send the content to the server.
3. **Sending Large Amounts of Data**:
    - Unlike **GET**, which appends data to the URL (and thus has size limitations), **POST** allows data to be sent in the request body. This makes it suitable for sending large amounts of data, including binary data such as images or files.
4. **Triggering Actions**:
    - **POST** is also used to trigger actions on the server, such as submitting a payment, processing a search query, or initiating an API request. It is typically employed when a server needs to perform an action that may have side effects, such as saving data or executing a function.
5. **Security**:
    - **POST** is considered more secure than **GET** for sensitive data (like passwords or credit card details) because the data is sent in the body of the request, not in the URL, which can be logged or cached. However, it's important to note that **POST** requests should still use HTTPS for encryption during transmission to ensure security.

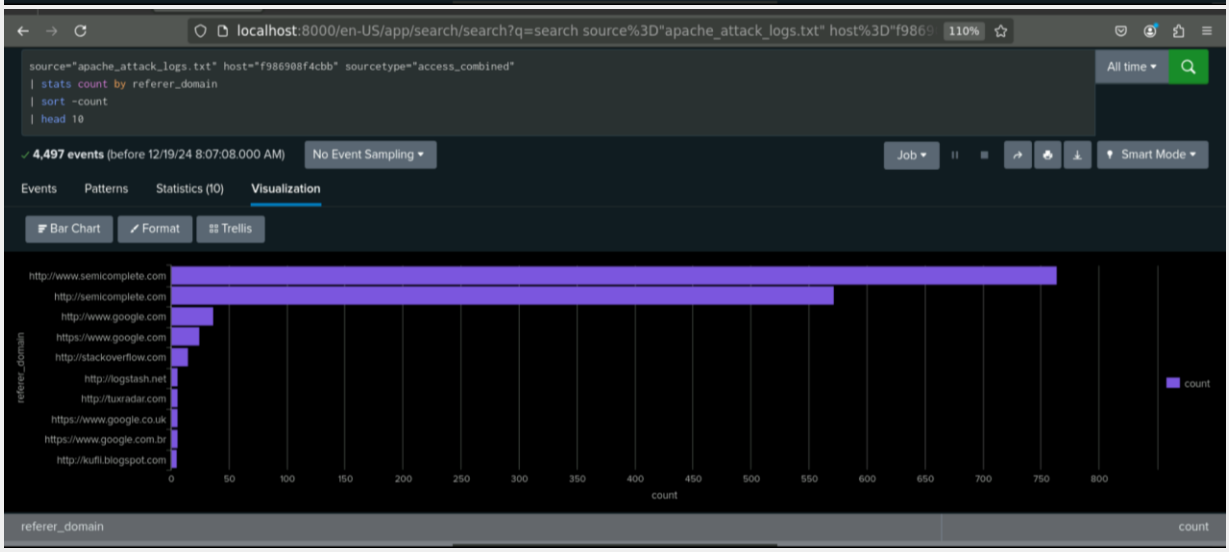
In summary, the **POST** method is used to send data to the server for various purposes, such as submitting forms, creating or updating resources, uploading files, and triggering server-side actions.

Report Analysis for Referrer Domains

- Did you detect any suspicious changes in referrer domains?

There are **two entries for "semicomplete.com"**—http://www.semicomplete.com and http://semicomplete.com—which could indicate **referrer duplication** due to variations in the domain (with and without "www"). This could suggest possible misconfiguration or **potential spam** traffic, where different variations of the domain are treated separately.
However, in terms of suspicious changes, there doesn't seem to be any **significant malicious referrers** like known spam sites or bots among the listed domains. The traffic appears mostly legitimate with recognized websites like **Google** and **StackOverflow**.
The primary concern might be the **"www" vs non-"www" variations** of your own domain (semicomplete.com), which can be streamlined for better analysis.





Report Analysis for HTTP Response Codes

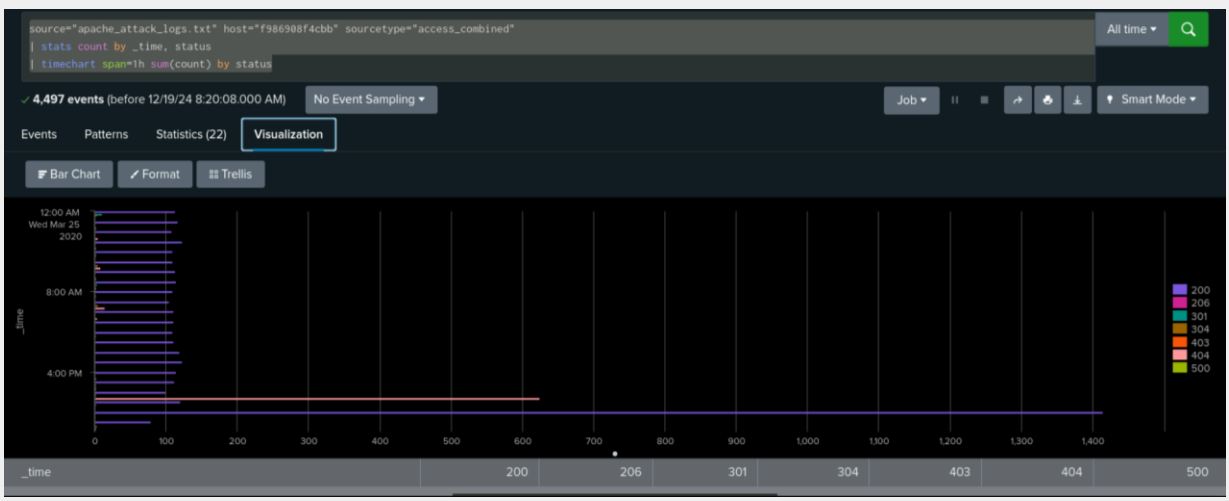
- Did you detect any suspicious changes in HTTP response codes?

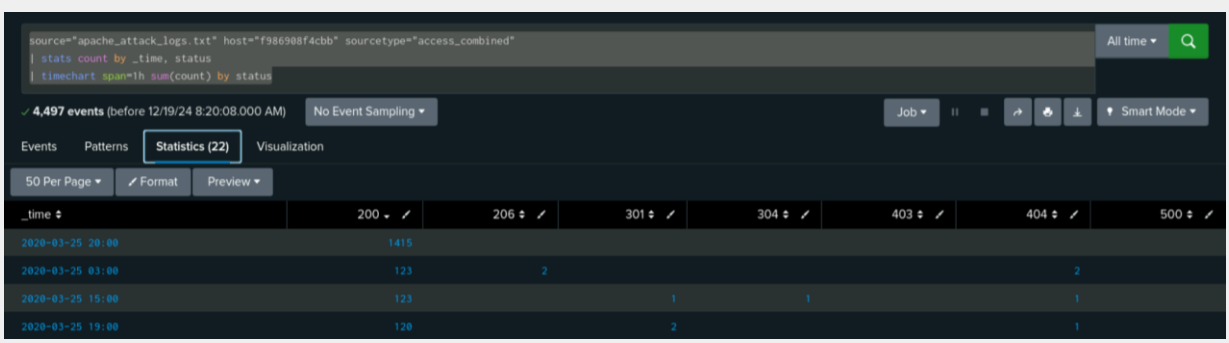
- Yes, the **404** responses at **18:00** (624) and **05:00** (9) stand out as potentially suspicious, as they are much higher than usual, which could suggest probing or scanning for vulnerabilities.
- The presence of **500** and **403** errors, though low, could also indicate misconfigurations or unauthorized access attempts. Further investigation is recommended to ensure these do not signal a targeted attack.

Peak Count of HTTP Methods by Status:

- **200**: Highest count observed at **20:00** (1415 responses).
- **404**: Peaks at **18:00** (624 responses), suggesting an anomaly in non-existent resource requests.

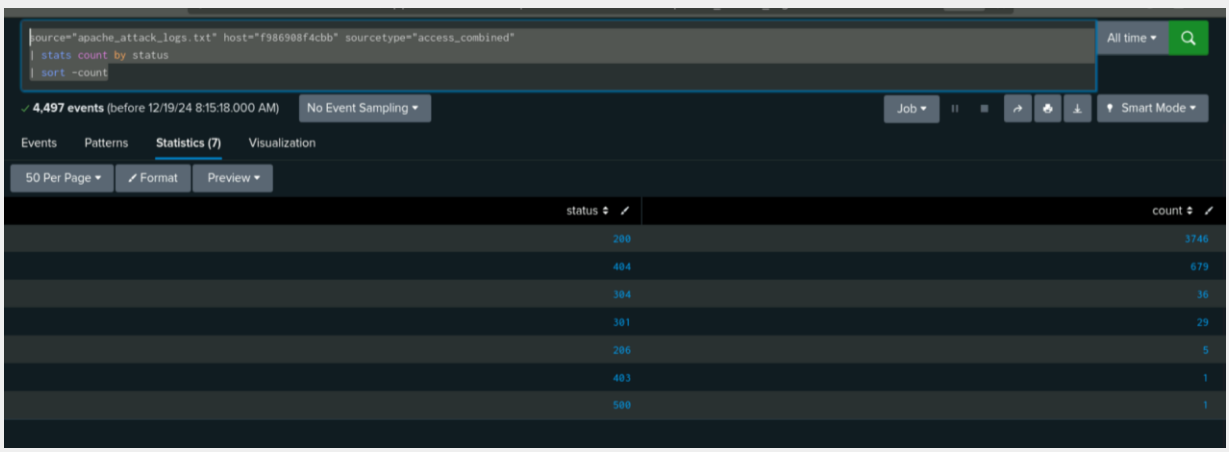
```
source="apache_attack_logs.txt" host="f986908f4cbb"
sourcetype="access_combined"
| stats count by _time, status
| timechart span=1h sum(count) by status
```

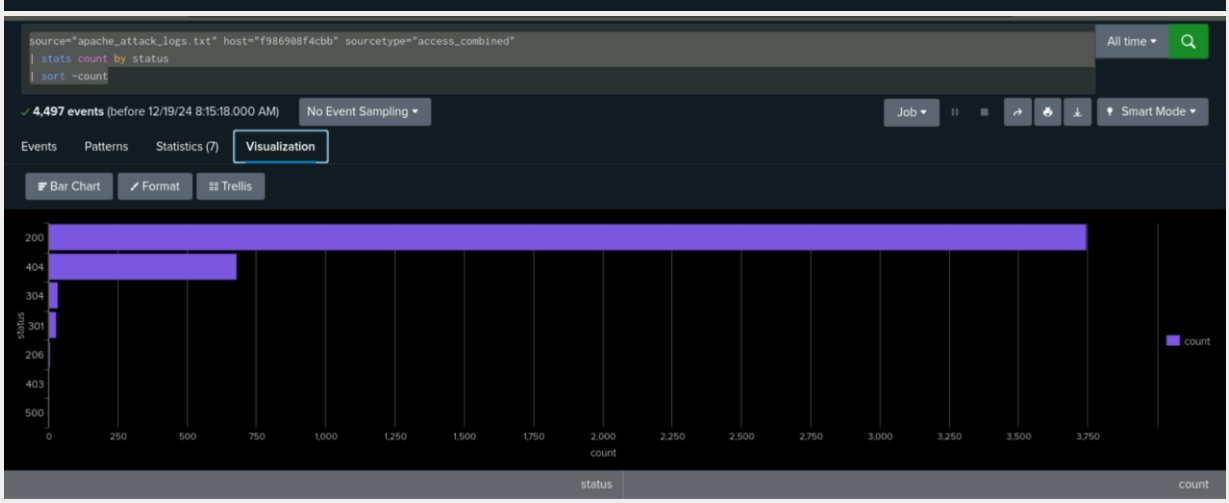




_time	200	206	301	304	403	404	500
2020-03-25 00:00	113		11	4			
2020-03-25 06:00	113					2	
2020-03-25 17:00	112		2	3		2	
2020-03-25 10:00	111				1	4	
2020-03-25 11:00	111			1			
2020-03-25 13:00	111			1		1	
2020-03-25 08:00	110			3		1	
2020-03-25 04:00	109		2	3		1	
2020-03-25 05:00	109		1	5		9	
2020-03-25 12:00	109			2		1	
2020-03-25 02:00	108		1	1		5	
2020-03-25 09:00	105			5		15	
2020-03-25 18:00	100	3	3			624	
2020-03-25 21:00	79			4		3	

```
source="apache_attack_logs.txt" host="f986908f4cbb"
sourcetype="access_combined"
| stats count by status
| sort -count
```



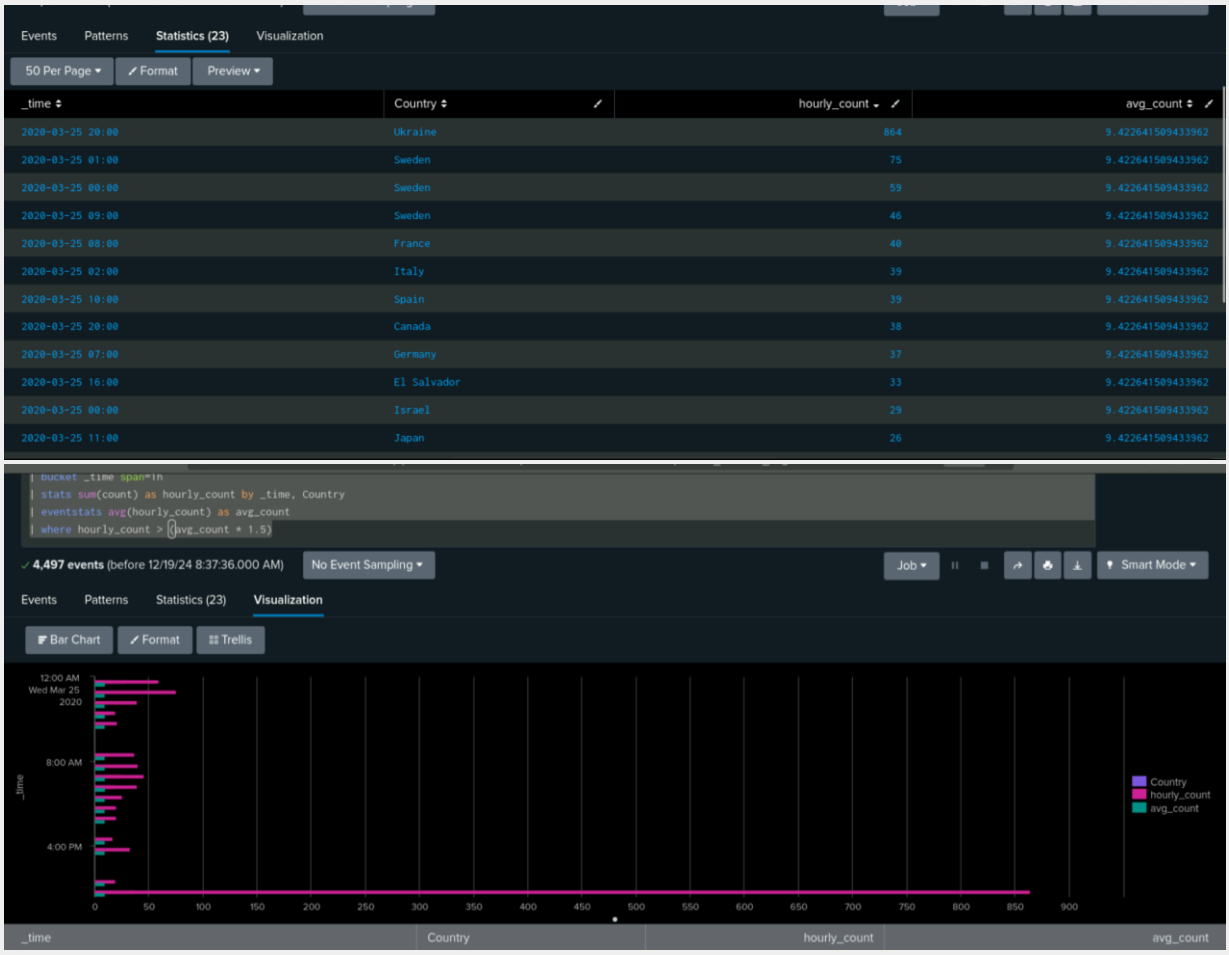


Alert Analysis for International Activity

- Did you detect a suspicious volume of international activity?

Yes, there appears to be a suspicious volume of international activity, particularly from **Ukraine**. The highest volume of activity (864 requests) occurred at **20:00** on March 25, 2020, which is notably higher than any other country listed.

```
source="apache_attack_logs.txt" host="f986908f4cbb"
sourcetype="access_combined"
| iplocation clientip
| stats count by Country, _time
| where Country != "United States"
| bucket _time span=1h
| stats sum(count) as hourly_count by _time, Country
| eventstats avg(hourly_count) as avg_count
| where hourly_count > (avg_count * 1.5)
```



- If so, what was the count of the hour(s) it occurred in?

```
   The suspicious activity peaked at 20:00, with 864 requests from Ukraine.
   Other notable international activity includes 75 requests from Sweden at
01:00 and 59 requests at 00:00.
```

- Would your alert be triggered for this activity?

```
Yes, this alert would likely be triggered due to the unusually high volume
of requests from Ukraine at 20:00. The count of 864 requests from a single
country is a significant anomaly compared to other countries and could
indicate possible malicious activity, such as a brute-force attack or a
botnet.
```

- After reviewing, would you change the threshold that you previously selected?

```
After reviewing, it would be advisable to adjust the threshold for
international traffic, especially focusing on large spikes from a single
country. A threshold alert for requests above a certain number (e.g., 100-
200 requests per hour) from any given country could help in detecting
unusual activity earlier. This would refine the detection of abnormal
traffic volumes more effectively.
```

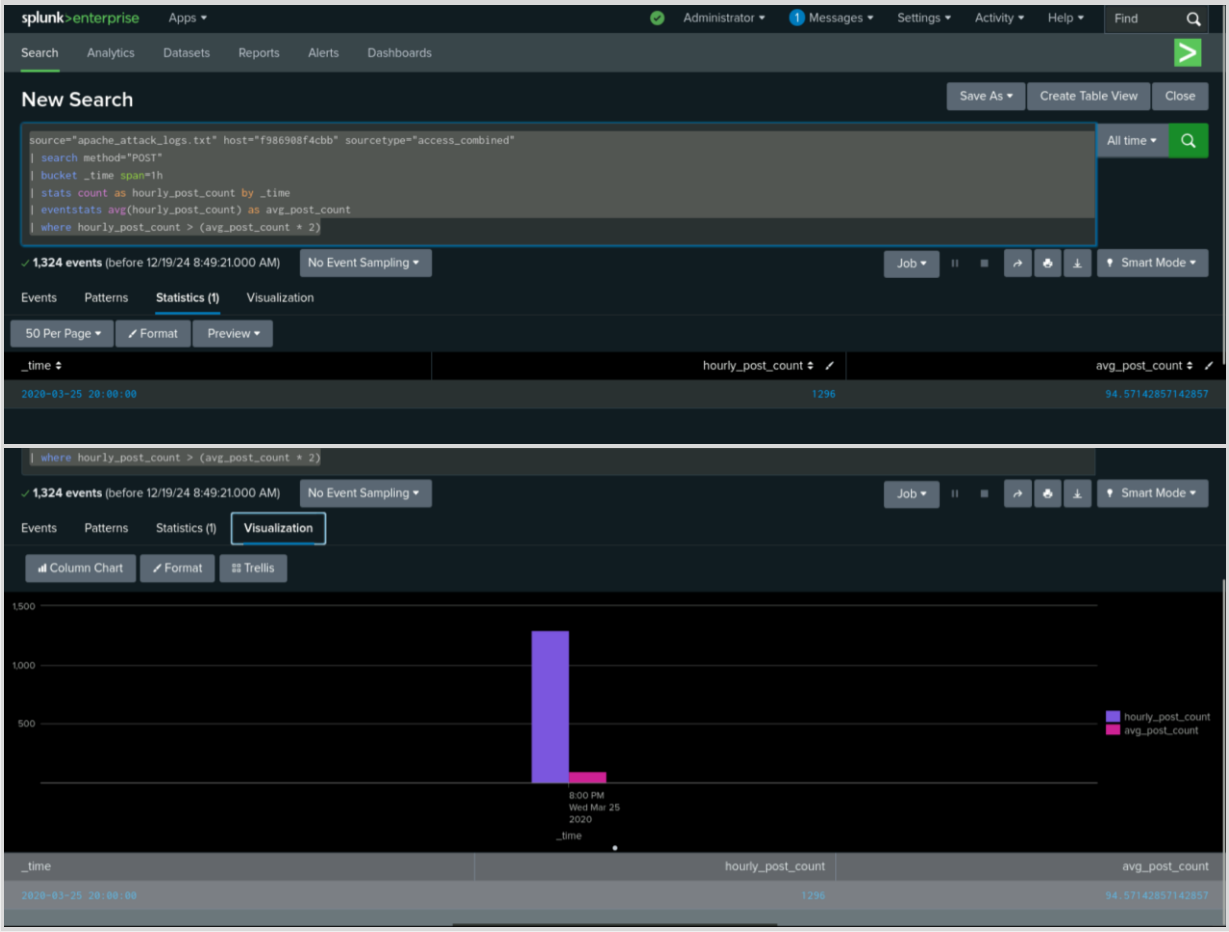
Alert Analysis for HTTP POST Activity

- Did you detect any suspicious volume of HTTP POST activity?

```
   Yes, a suspicious volume of HTTP POST activity was detected. The hourly
POST count at 20:00 on 2020-03-25 was 1296, which is significantly higher
than the average POST count of 94.57.
```

```
source="apache_attack_logs.txt" host="f986908f4cbb"
sourcetype="access_combined"
| search method="POST"
| bucket _time span=1h
| stats count as hourly_post_count by _time
```

```
| eventstats avg(hourly_post_count) as avg_post_count
| where hourly_post_count > (avg_post_count * 2)
```



- If so, what was the count of the hour(s) it occurred in?

```
The count of events in the hour was 1296 POST requests at 20:00 on March 25, 2020.
```

- When did it occur?

```
The suspicious activity occurred at 20:00 on 2020-03-25.
```

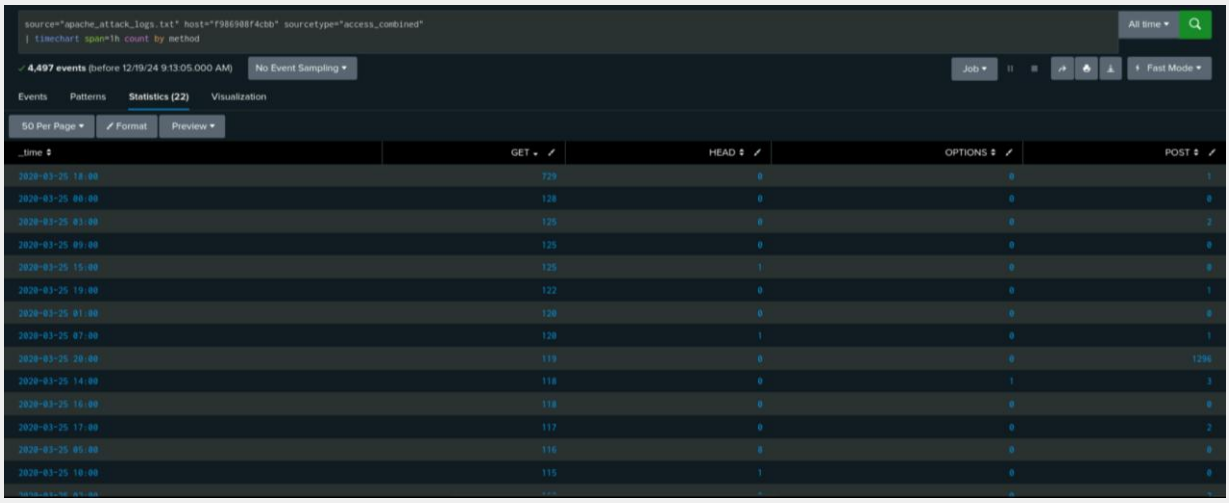
- After reviewing, would you change the threshold that you previously selected?

After reviewing, the threshold should likely be adjusted to account for spikes that are more than **2x the average** POST request count. The threshold may need to be lowered or recalibrated depending on the context of the average traffic and to better capture anomalous spikes, such as the one at 20:00 with **1296 POST requests**. A threshold around **double the average** (i.e., greater than ~189) seems effective for capturing significant anomalies in POST activity.

Dashboard Analysis for Time Chart of HTTP Methods

- Does anything stand out as suspicious?

Yes, there is suspicious activity, particularly at **20:00 on March 25, 2020**. At this hour, there is an abnormally high volume of **POST requests** (**1296 POSTs**) compared to other times, especially considering the other hours show relatively normal POST activity with counts typically between 0 and 2 POST requests per hour.



_time	GET	HEAD	OPTIONS	POST
2020-03-25 18:00	729	0	0	1
2020-03-25 00:00	128	0	0	0
2020-03-25 03:00	125	0	0	2
2020-03-25 09:00	125	0	0	0
2020-03-25 15:00	125	1	0	0
2020-03-25 19:00	122	0	0	1
2020-03-25 01:00	120	0	0	0
2020-03-25 07:00	120	1	0	1
2020-03-25 20:00	119	0	0	1296
2020-03-25 14:00	118	0	1	3
2020-03-25 16:00	118	0	0	0
2020-03-25 17:00	117	0	0	2
2020-03-25 05:00	116	0	0	0
2020-03-25 10:00	115	1	0	0

- Which method seems to be used in the attack?

```
The POST method is the one primarily used in the attack, as evidenced by the
massive spike at 20:00, where the POST count jumps to 1296. This is far
above the normal baseline of 0-2 POST requests seen in other hours.
```

- At what times did the attack start and stop?

```
   Start time: The attack seems to begin at 20:00 when the POST requests
spike to 1296.
   End time: There is no clear end time provided from the dashboard, but
based on the data, the POST activity significantly drops after 20:00,
returning to low counts at subsequent hours (e.g., 19:00 and 21:00).
```

- What is the peak count of the top method during the attack?

```
The peak count of the POST method during the attack is 1296 at 20:00. This
is the highest volume of POST requests recorded and is considered highly
suspicious compared to the typical traffic seen in other hours.
```

Dashboard Analysis for Cluster Map

- Does anything stand out as suspicious?

```
Suspicious Observations:
```

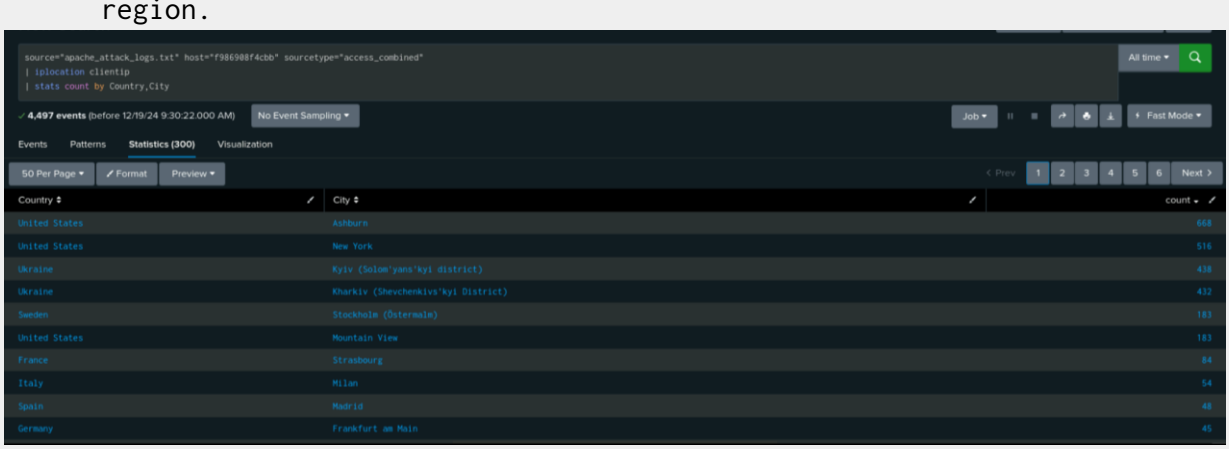
- **High volume of activity** from the following locations stands out:
  - **Ashburn, United States** (668 events)
  - **New York, United States** (516 events)
  - **Kyiv, Ukraine** (438 events)
  - **Kharkiv, Ukraine** (432 events)

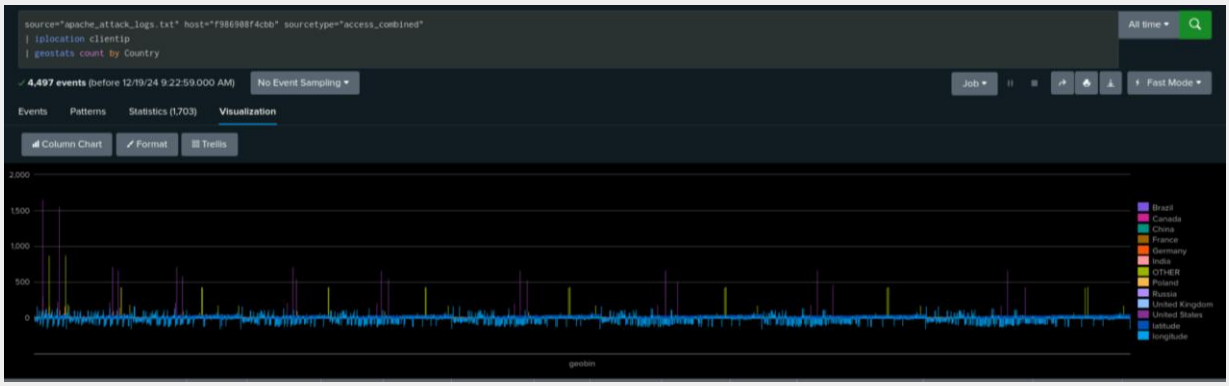
New Locations with High Volume:

- **Ashburn, United States** has the **highest volume of activity**, with **668** events.
- It is notable because Ashburn is a well-known data center hub in the U.S., and a high volume of traffic from there could indicate traffic originating from automated systems or servers (e.g., bots, attack infrastructure).

Summary:

- **Suspicious activity** might be coming from **Ashburn, United States** with a **count of 668** events, which could represent an unusually high volume of requests.
- **Ukraine** cities (Kyiv and Kharkiv) also show high activity, which might warrant further investigation due to the geopolitical context of the region.





- Which new location (city, country) on the map has a high volume of activity? (**Hint**: Zoom in on the map.)

- **Ashburn, United States** has the **highest volume of activity**, with **668** events.
- It is notable because Ashburn is a well-known data center hub in the U.S., and a high volume of traffic from there could indicate traffic originating from automated systems or servers (e.g., bots, attack infrastructure).

- What is the count of that city?

668

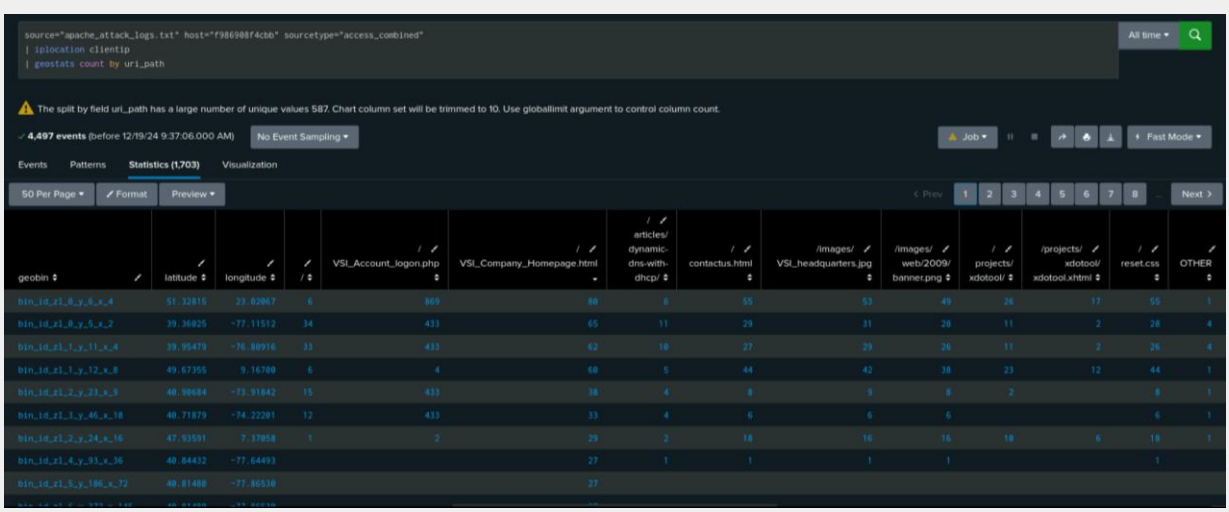
Dashboard Analysis for URI Data

- Does anything stand out as suspicious?

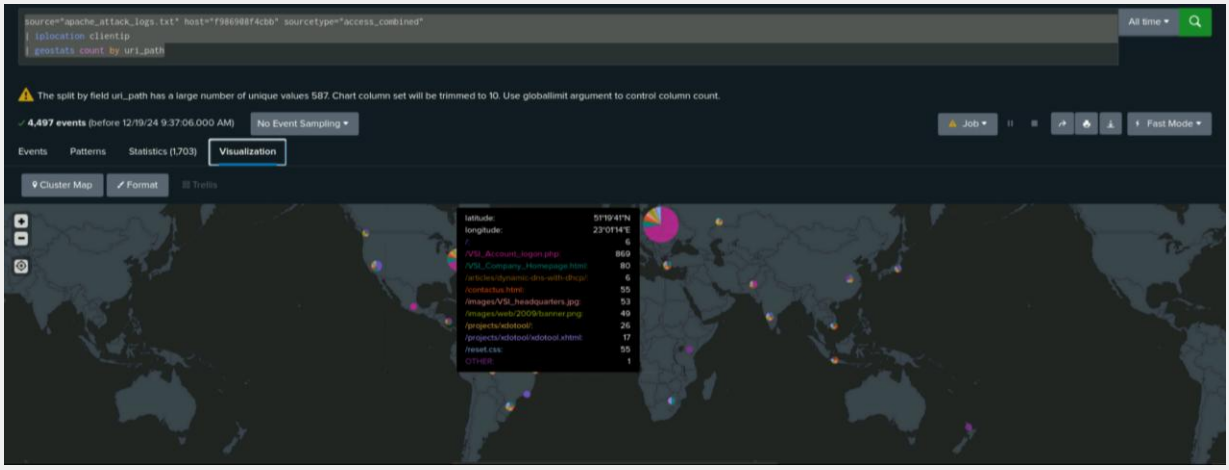
The URI **"/VSI_Account_logon.php"** appears to be hit the most, with counts of **869** events from the first bin (bin_id_zl_0_y_6_x_4).

This is a critical and potentially sensitive URI, typically used for logging into accounts. If it's targeted heavily, it could indicate attempts to exploit login vulnerabilities, such as **brute-force login attempts** or **credential stuffing attacks**.

```
source="apache_attack_logs.txt" host="f986908f4cbb"
sourcetype="access_combined"
| iplocation clientip
| geostats count by uri_path
```



geobin	latitude	longitude	/	VSI_Account_logon.php	VSI_Company_Homepage.html	articles/dynamic-dns-with-dhcp/	contactus.html	/images/VSI_headquarters.jpg	/images/web/2009/banner.png	projects/xdotool/	/projects/xdotool/xdotool.xhtml	reset.css	OTHER
bin_id_zl_0_y_6_x_4	51.32815	23.02067	6	869	80	6	55	53	49	26	17	55	1
bin_id_zl_0_y_5_x_2	39.36025	-77.11512	34	433	65	11	29	31	28	11	2	28	4
bin_id_zl_1_y_11_x_4	39.95479	-76.80916	33	433	62	10	27	29	26	11	2	26	4
bin_id_zl_1_y_12_x_8	49.67355	9.16700	6	4	60	5	44	42	38	23	12	44	1
bin_id_zl_2_y_23_x_9	40.90684	-73.91842	15	433	38	4	8	9	8	2		8	1
bin_id_zl_3_y_46_x_18	40.71879	-74.22201	12	433	33	4	6	6	6			6	1
bin_id_zl_2_y_24_x_16	47.93591	7.37058	1	2	29	2	18	16	16	10	6	18	1
bin_id_zl_4_y_93_x_36	40.84432	-77.64493			27	1	1	1	1			1	
bin_id_zl_5_y_186_x_72	40.81480	-77.86530			27								



- What URI is hit the most?

The URI that is hit the most in the provided log data is:
"/VSI_Account_logon.php"
It has the highest count of **869** events, indicating that it is accessed most frequently compared to other URIs in the logs.

- Based on the URI being accessed, what could the attacker potentially be doing?

Exploitation of Account Login:

- The attacker could be attempting to **compromise accounts** by repeatedly trying different credentials or exploiting any vulnerabilities in the account login system.
- Given the high number of hits to the /VSI_Account_logon.php endpoint, this might point to **brute-force attacks**, where automated tools are used to guess usernames and passwords.

Other URIs of Interest:

- The presence of other URIs like /VSI_Company_Homepage.html, /images/VSI_headquarters.jpg, and /images/web/2009/banner.png suggests the attacker might also be probing for other resources or attempting to scrape the website for information, such as finding vulnerabilities in other parts of the site or gathering intelligence.