**LAB-5 REPORT**

Course Code - Course Name: - COMP4040 – Introduction to SOC

Program: T433 - Cybersecurity

Term: - Winter 2024

Student Names - ID:

* Prabhjot Singh Sains – 101495218

1. **use\_sourceCategory=linux/system and write queries to retrieve data from the past 3 days (i.e. "-3d"):**

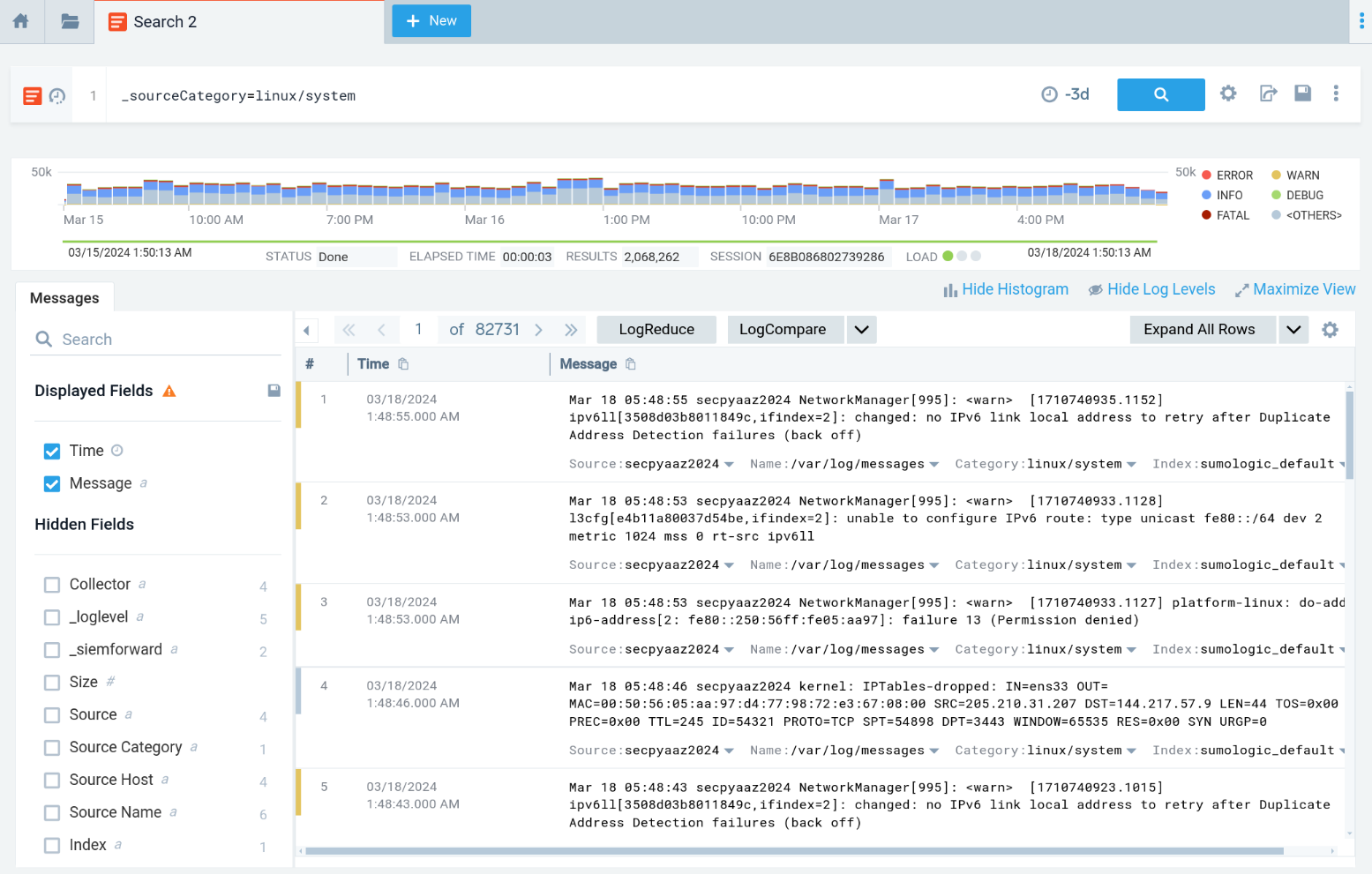


Fig 1. Logs of Linux System from past 3 days

1. **Filter for "Failed password" and then:**

\_sourceCategory=linux/system "Failed password")

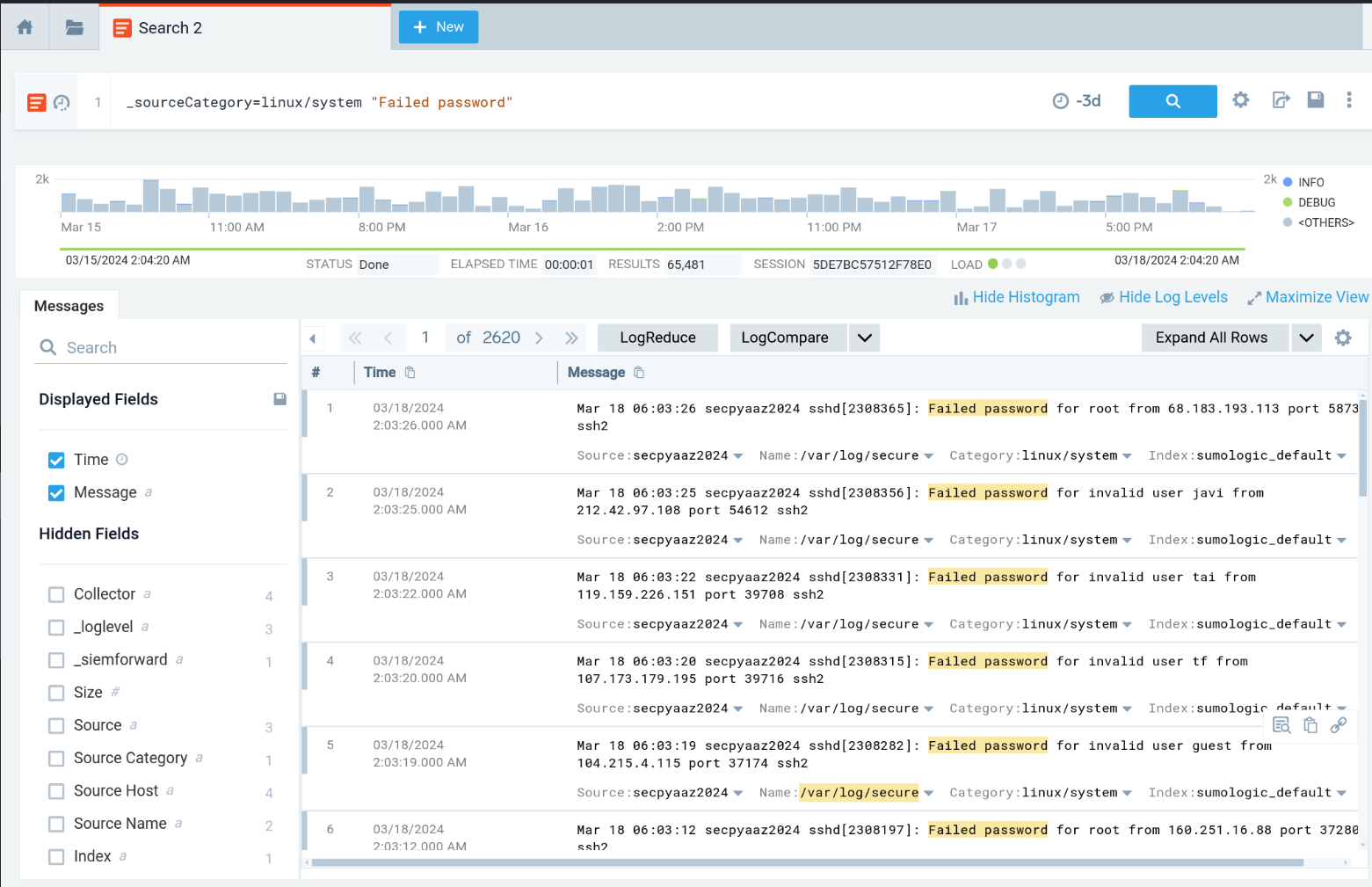


Fig 2. A total of 2620 logs of failed password login attempts on the Linux system.

1. **Extract month, date, time, hostname, service, pid, username, src\_ip, port, and protocol:**

(\_sourceCategory=linux/system "Failed password")

| parse "\* \* \* \* \*[\*]: Failed password for invalid user \* from \* port \* \*" as month,date,time,hostname,service,pid,username,src\_ip,port,protocol

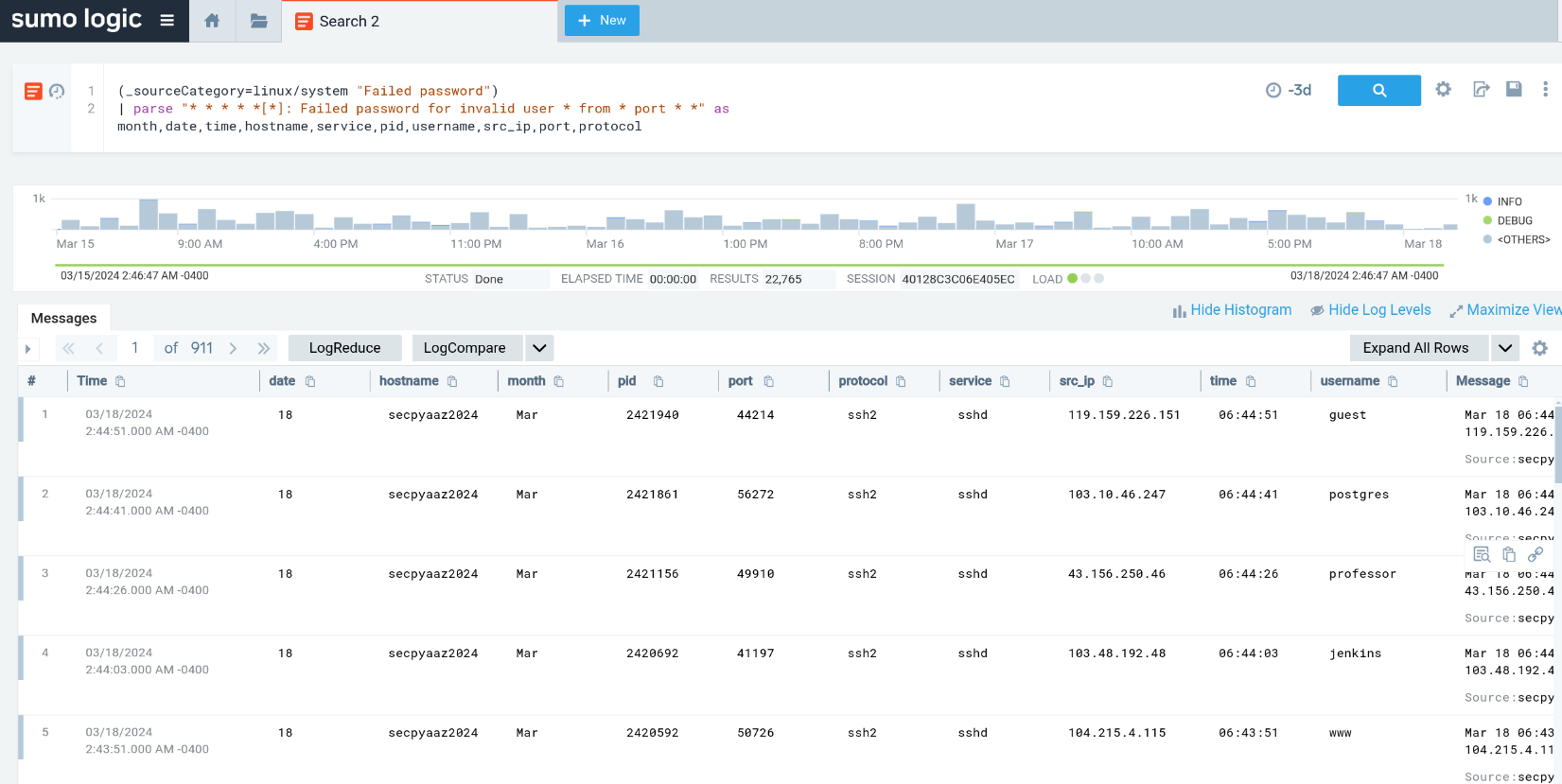


Fig 3. The Parse log data as per the query

1. **Create a table based on username count (Panel01):**

(\_sourceCategory=linux/system "Failed password")

| parse "\* \* \* \* \*[\*]: Failed password for invalid user \* from \* port \* \*" as month,date,time,hostname,service,pid,username,src\_ip,port,protocol | count by username

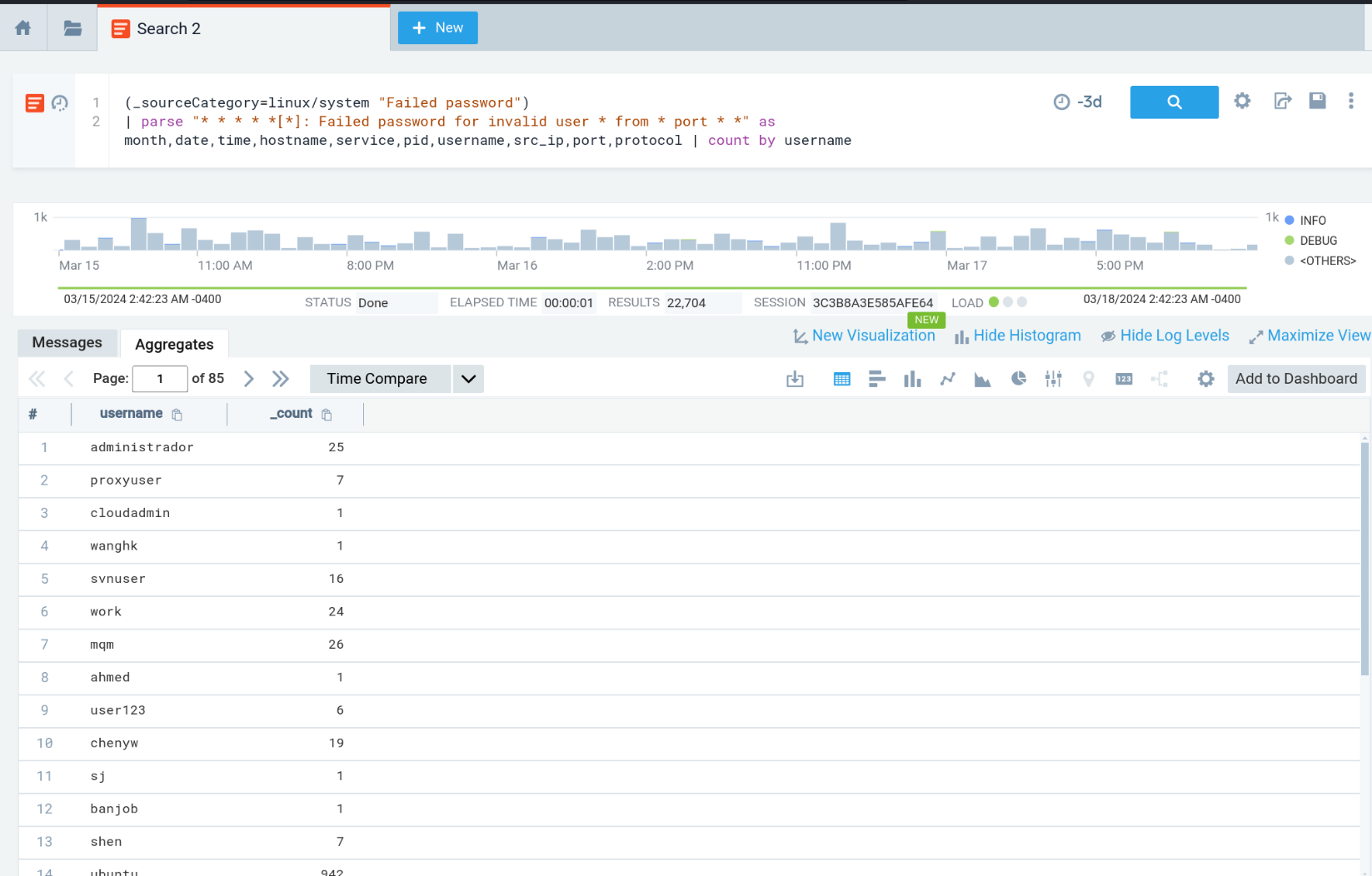


Fig 4. A table Based on username count

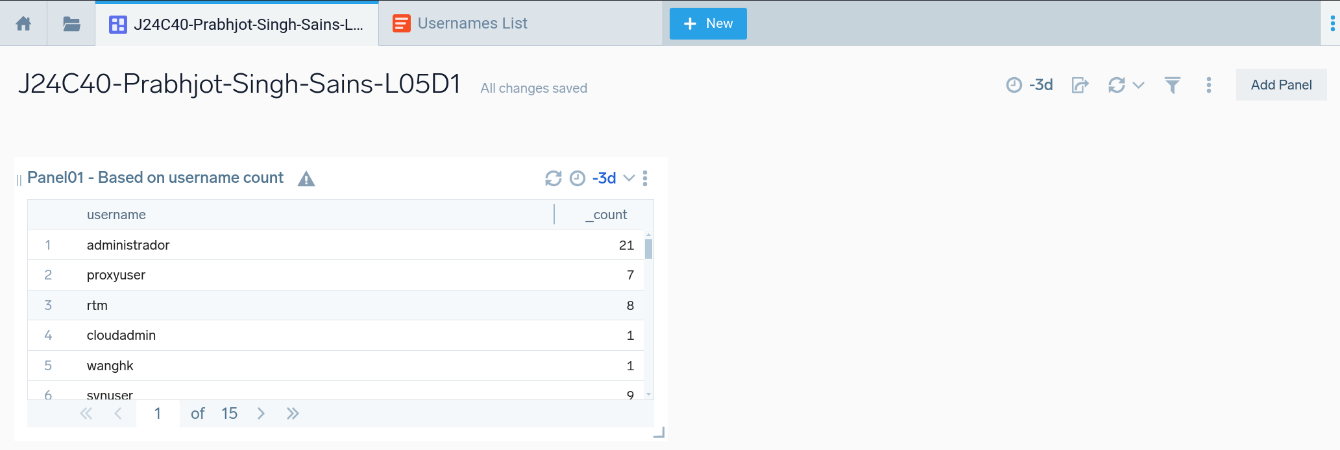


Fig 5. Dashboard with Panel 1 showing a table with username

1. **Download the usernames list:**

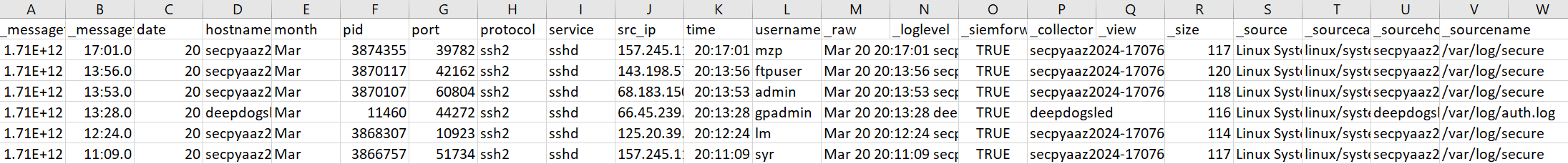


Fig 6. Some of the downloaded username shown in excel

1. **Find 5 most tried usernames (Panel02):**

(\_sourceCategory=linux/system "Failed password")

| parse "\* \* \* \* \*[\*]: Failed password for invalid user \* from \* port \* \*" as month,date,time,hostname,service,pid,username,src\_ip,port,protocol

| count by username | order by \_count | top 5 username by \_count

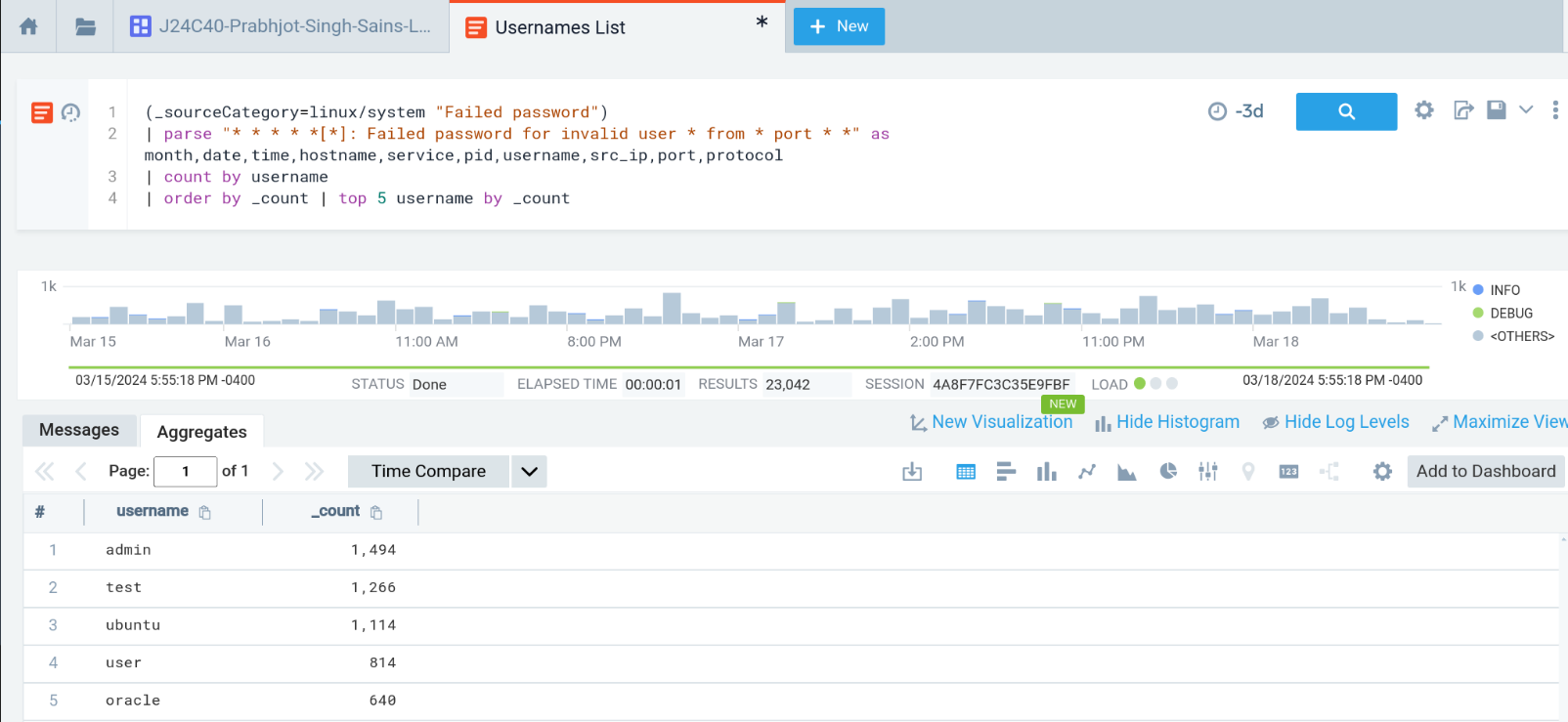


Fig 7. Query and Output showing top 5 usernames used in Linux system

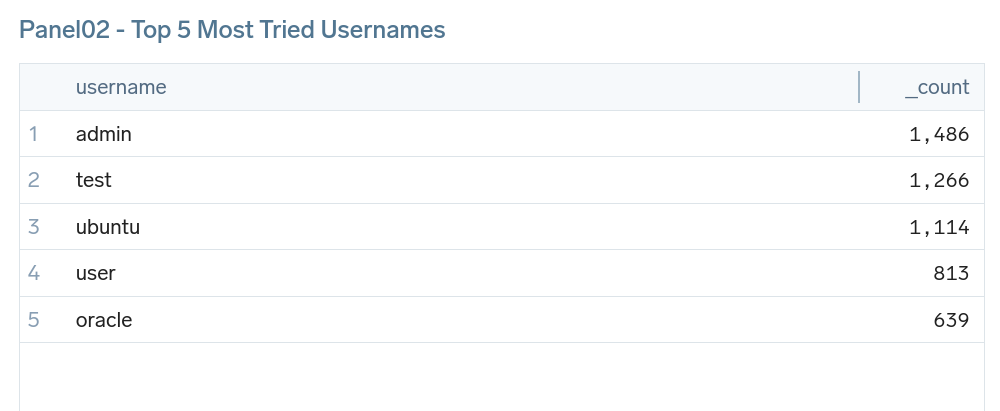


Fig 8. Panel 2 for top 5 usernames used in Linux system

1. **Create a graph to visualize Failed attempts every 30 minutes (Panel03):**

(\_sourceCategory=linux/system "Failed password")

| timeslice 30m | count by \_timeslice | order by \_count

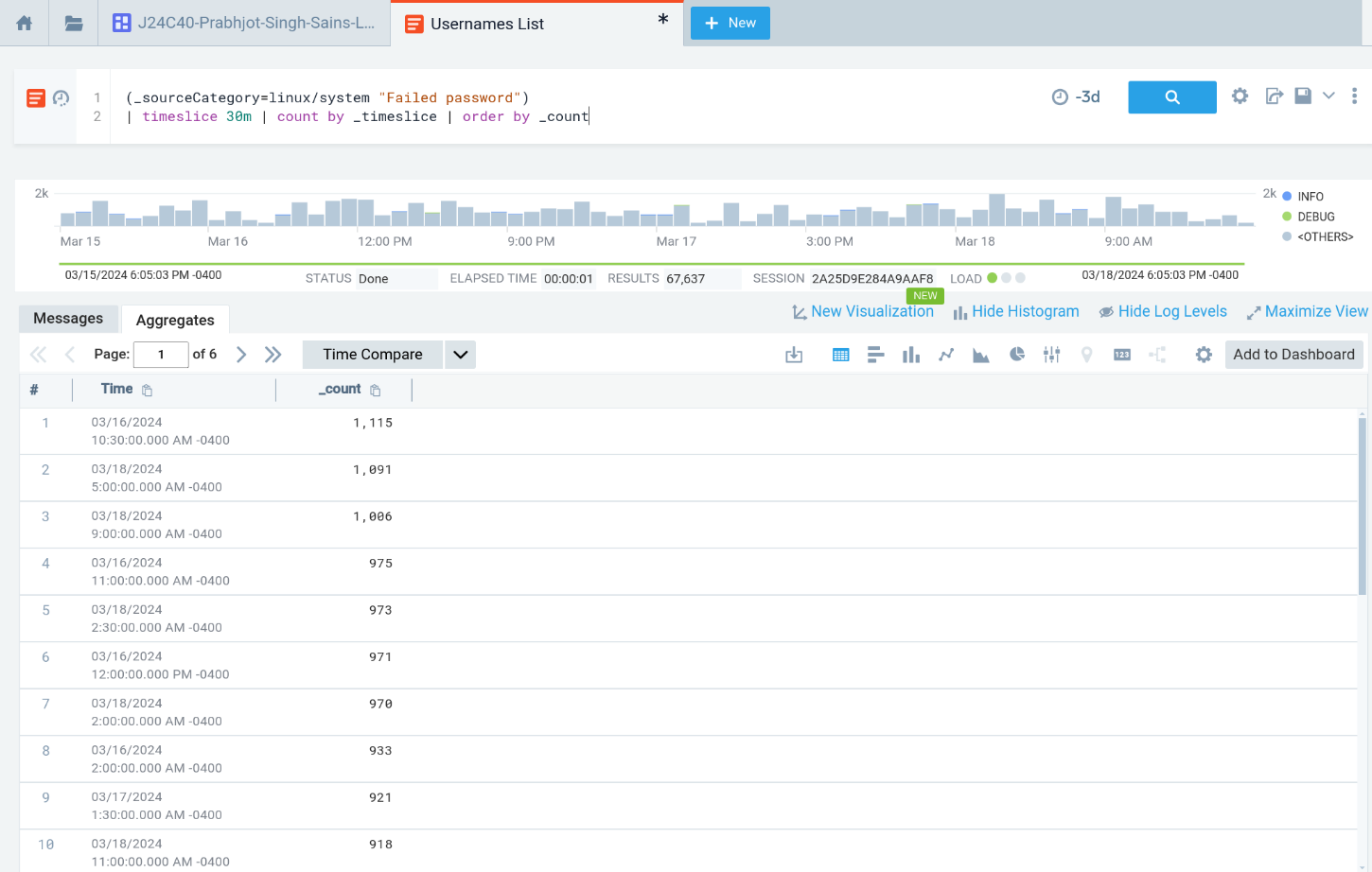


Fig 9. Query and Output showing login failed attempts every 30 minutes

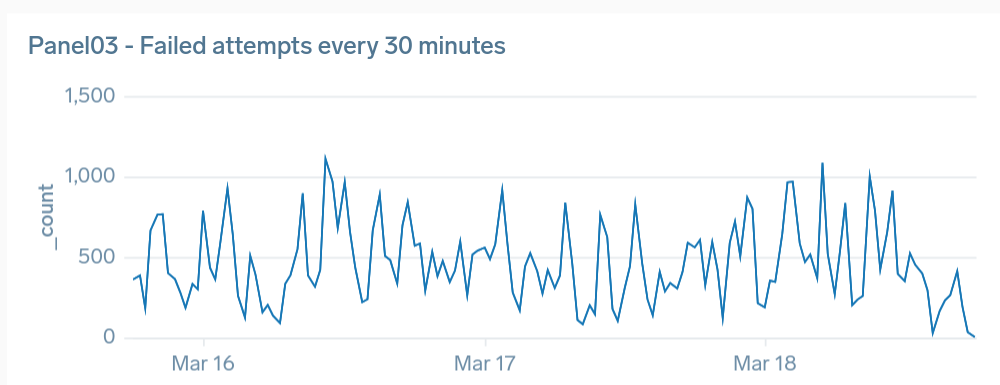


Fig 10. A Graph showing login failed attempts every 30 minutes

1. **Extract geolocation information from src\_ip:**

(\_sourceCategory=linux/system "Failed password")

| parse "\* \* \* \* \*[\*]: Failed password for invalid user \* from \* port \* \*" as month,date,time,hostname,service,pid,username,src\_ip,port,protocol

| lookup latitude, longitude from geo://location on ip = src\_ip | count by latitude, longitude

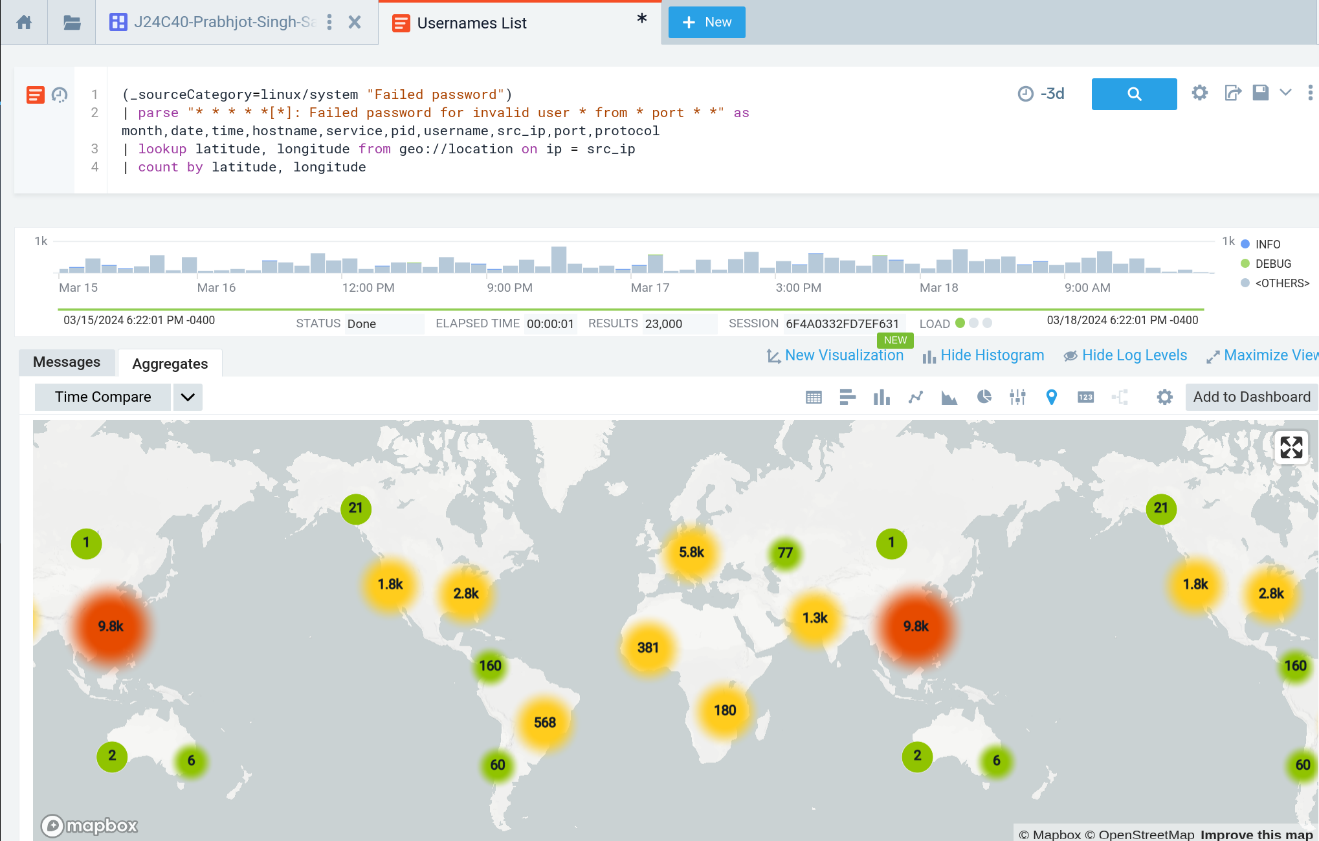


Fig 11. Query and Output showing geolocation information from src\_ip in world map

1. **Order attempts from each city (Panel04):**

(\_sourceCategory=linux/system "Failed password")

| parse "\* \* \* \* \*[\*]: Failed password for invalid user \* from \* port \* \*" as month,date,time,hostname,service,pid,username,src\_ip,port,protocol

| lookup latitude, longitude, city from geo://location on ip = src\_ip | count by city | order by \_count

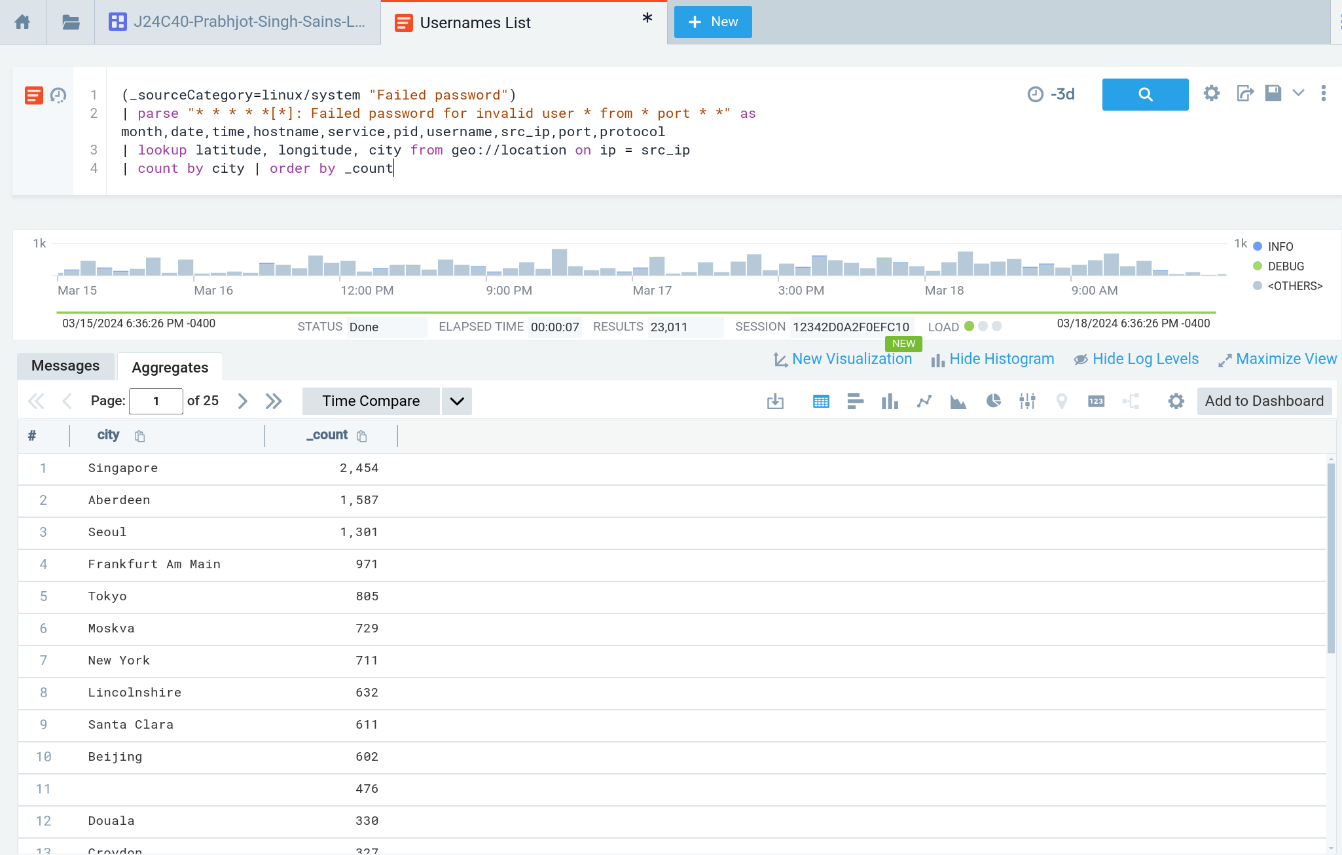


Fig 12. Query and Output showing attempts from cities

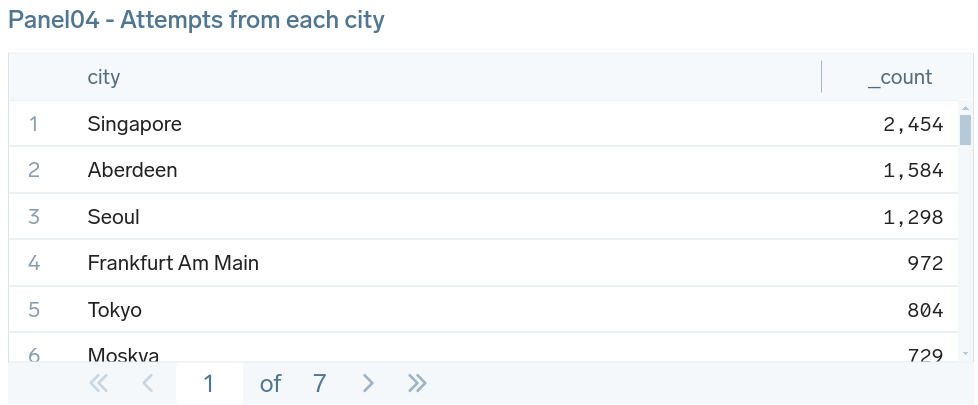


Fig 13. Panel showing from cities

1. **Order attempts from each country (Panel05):**

(\_sourceCategory=linux/system "Failed password")

| parse "\* \* \* \* \*[\*]: Failed password for invalid user \* from \* port \* \*" as month,date,time,hostname,service,pid,username,src\_ip,port,protocol

| lookup latitude, longitude, country\_name from geo://location on ip = src\_ip

| count by country\_name | order by \_count

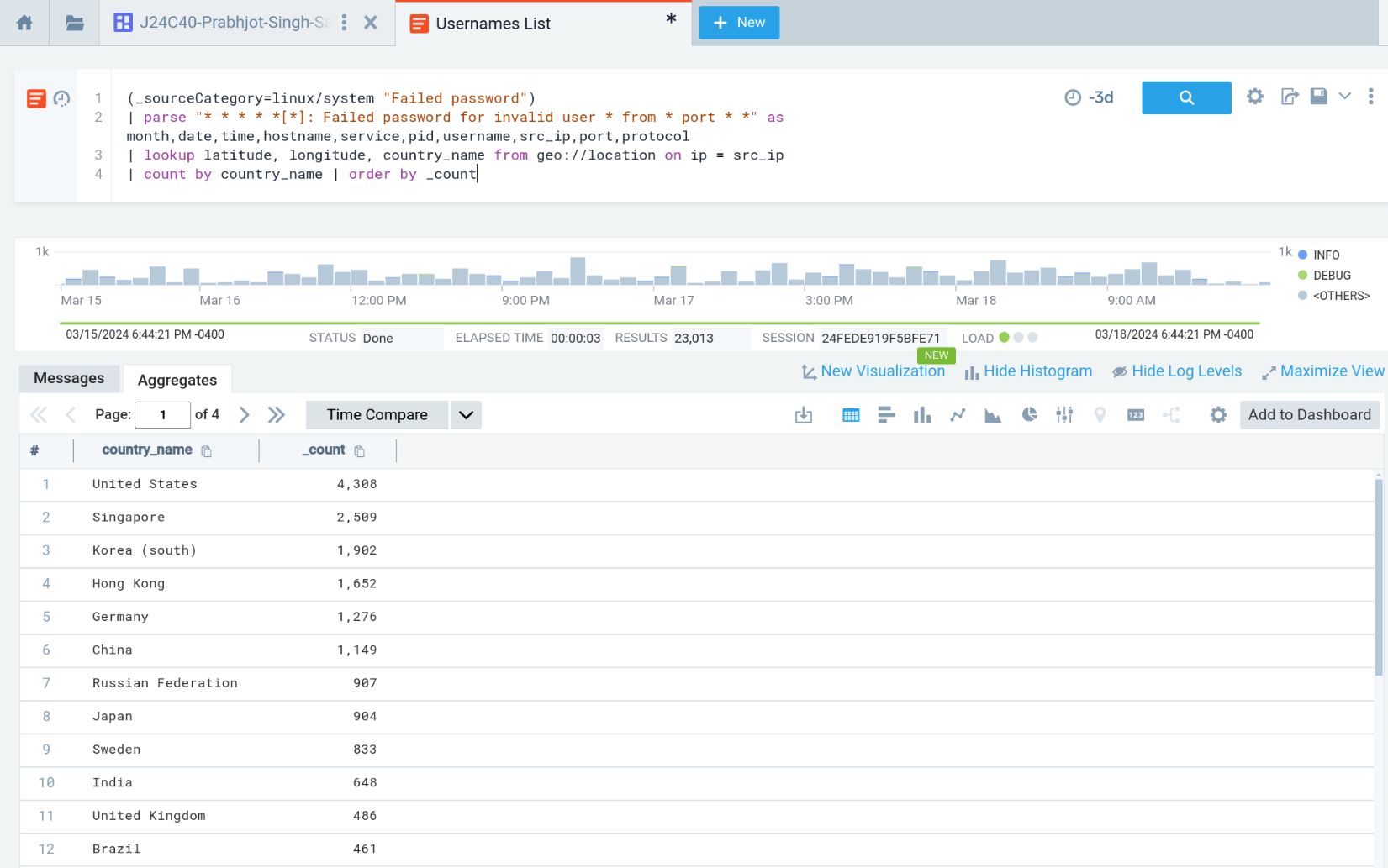


Fig 14. Query and Output showing attempts from countries



Fig 15. Panel showing from countries

1. **Visualize src\_ip on a map (Panel06):**

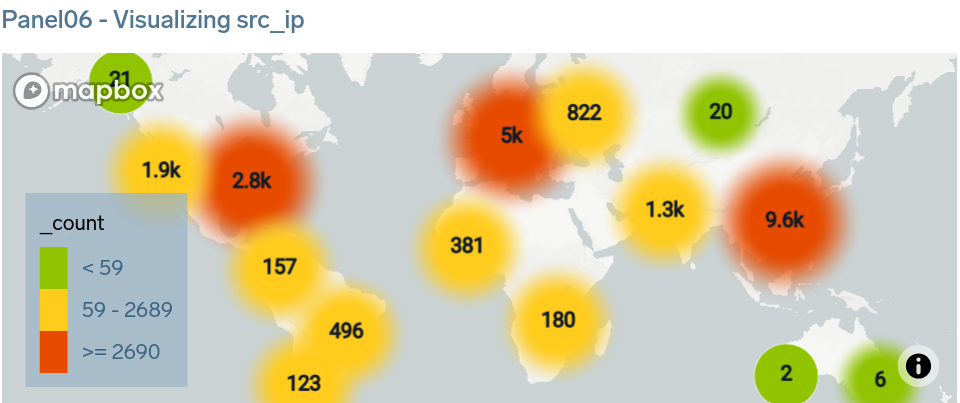


Fig 16. Showing src\_ip on a map

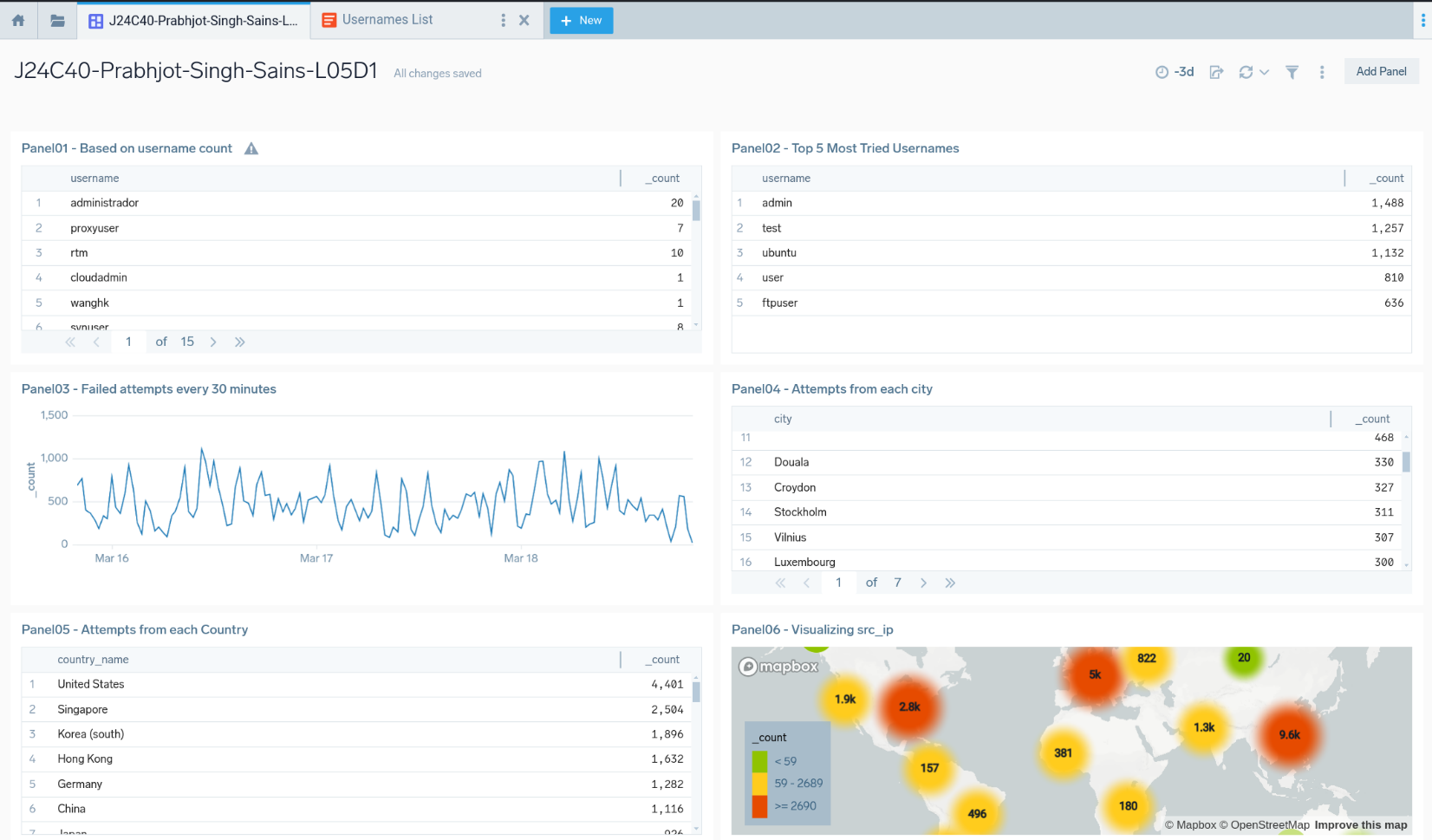


Fig 17. Dashboard with six panels showing data from logs

1. **Use the source and collector with the identifier "\_sourceCategory=windows/events" and write queries to retrieve data from the past 5 days (i.e. "-5d").**

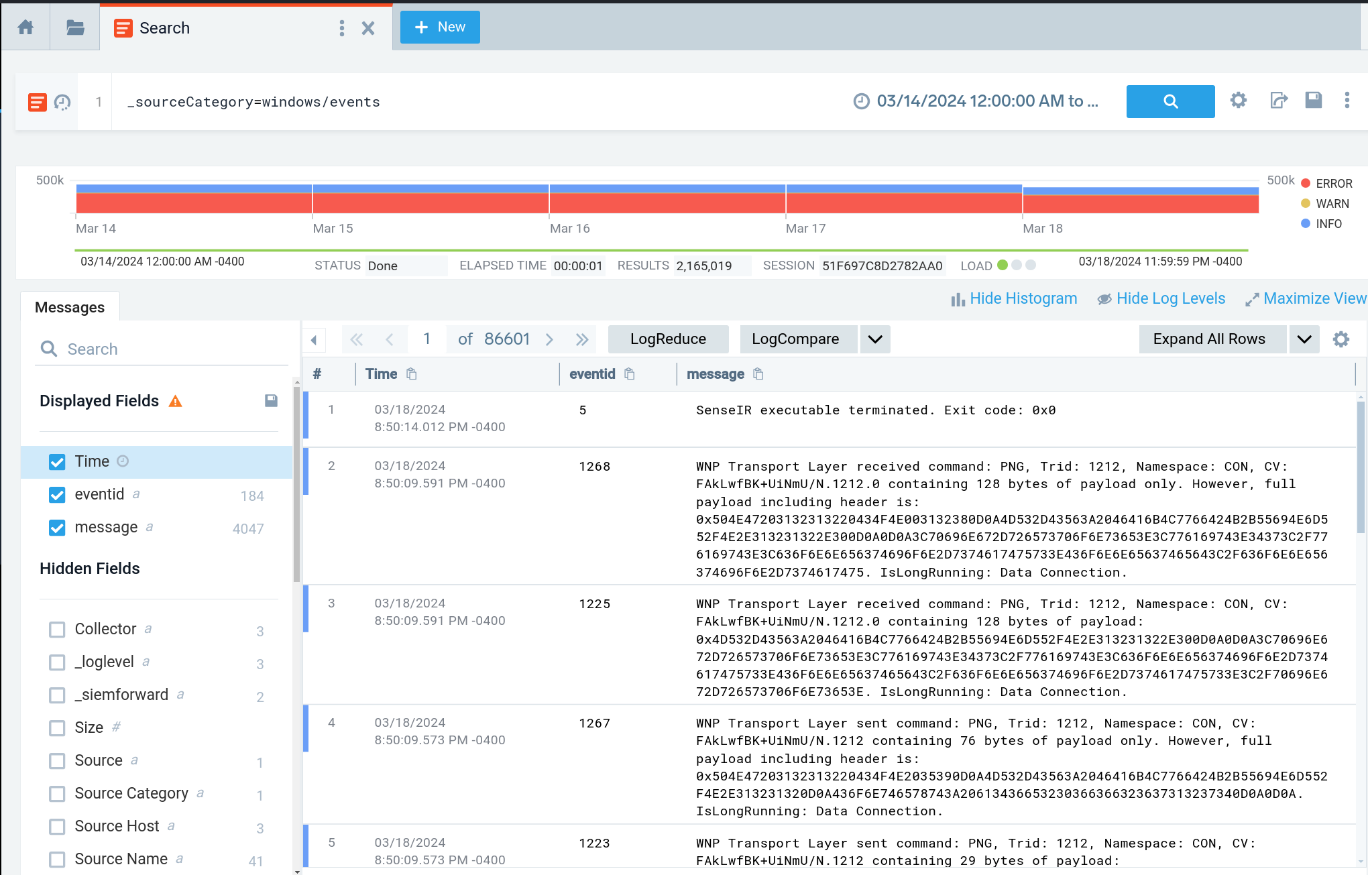


Fig 18. Windows log events from past 5 days

1. **Create a table view that displays the top 10 most frequently occurring events based on their ID (eventID). (Panel01):**

\_sourceCategory=windows/events | count by eventid | order by \_count | top 10 eventid by \_count

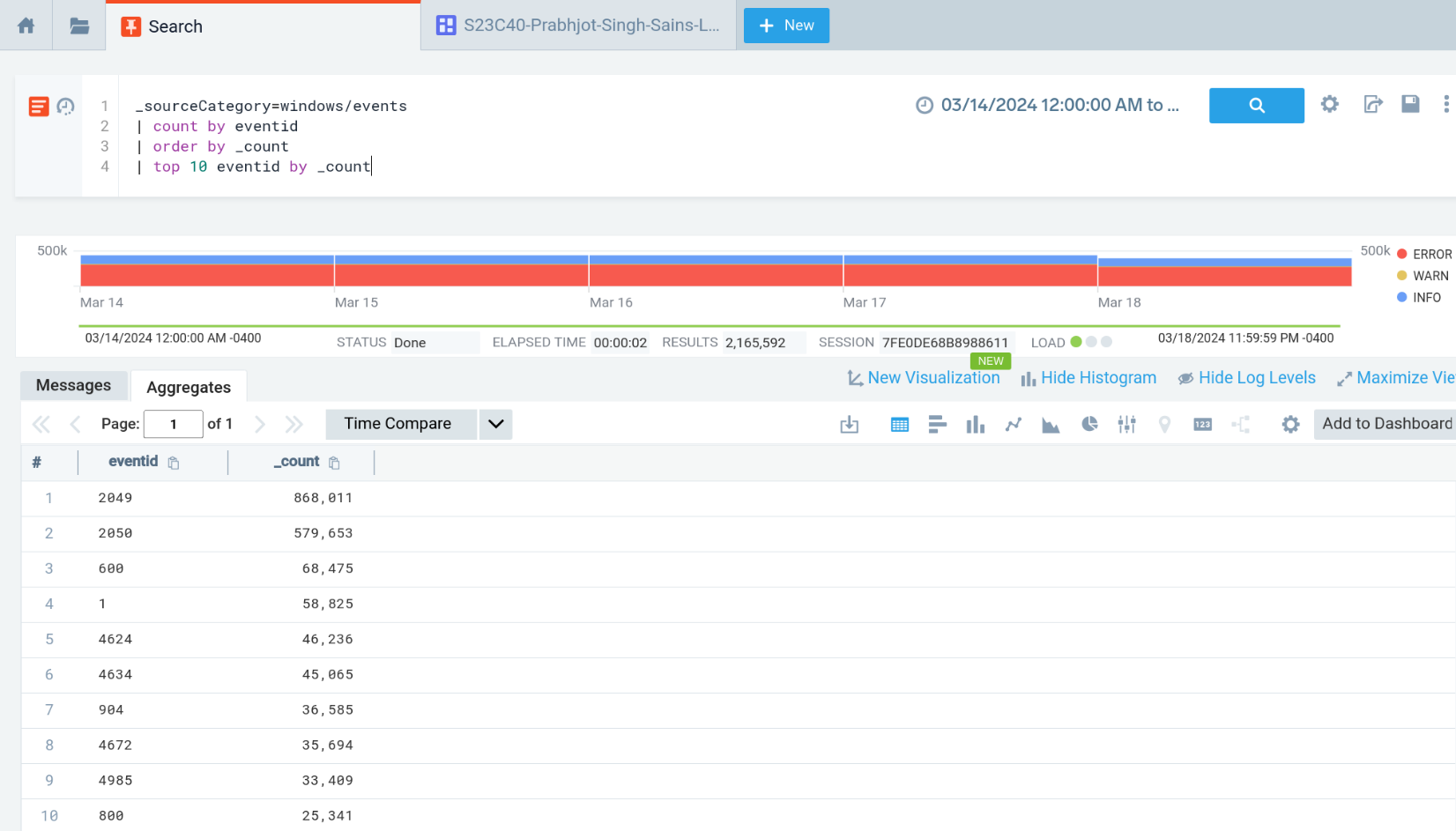


Fig 19. Query showing table view of Top 10 frequently occurring events based on ID

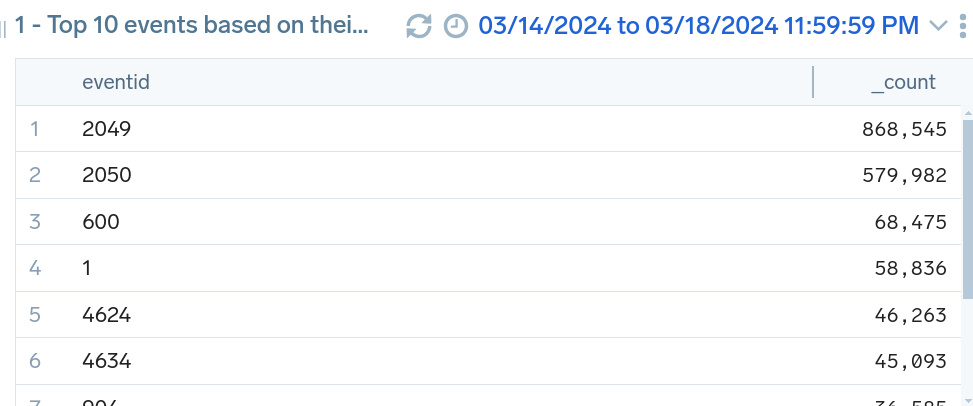


Fig 20. Panel 1 showing table for Top 10 frequently occurring events based on ID

1. **Write a query and create a graph (panel) to show the number of events each day. (Panel02):**

\_sourceCategory=windows/events

| timeslice 1d | count by \_timeslice | order by \_count

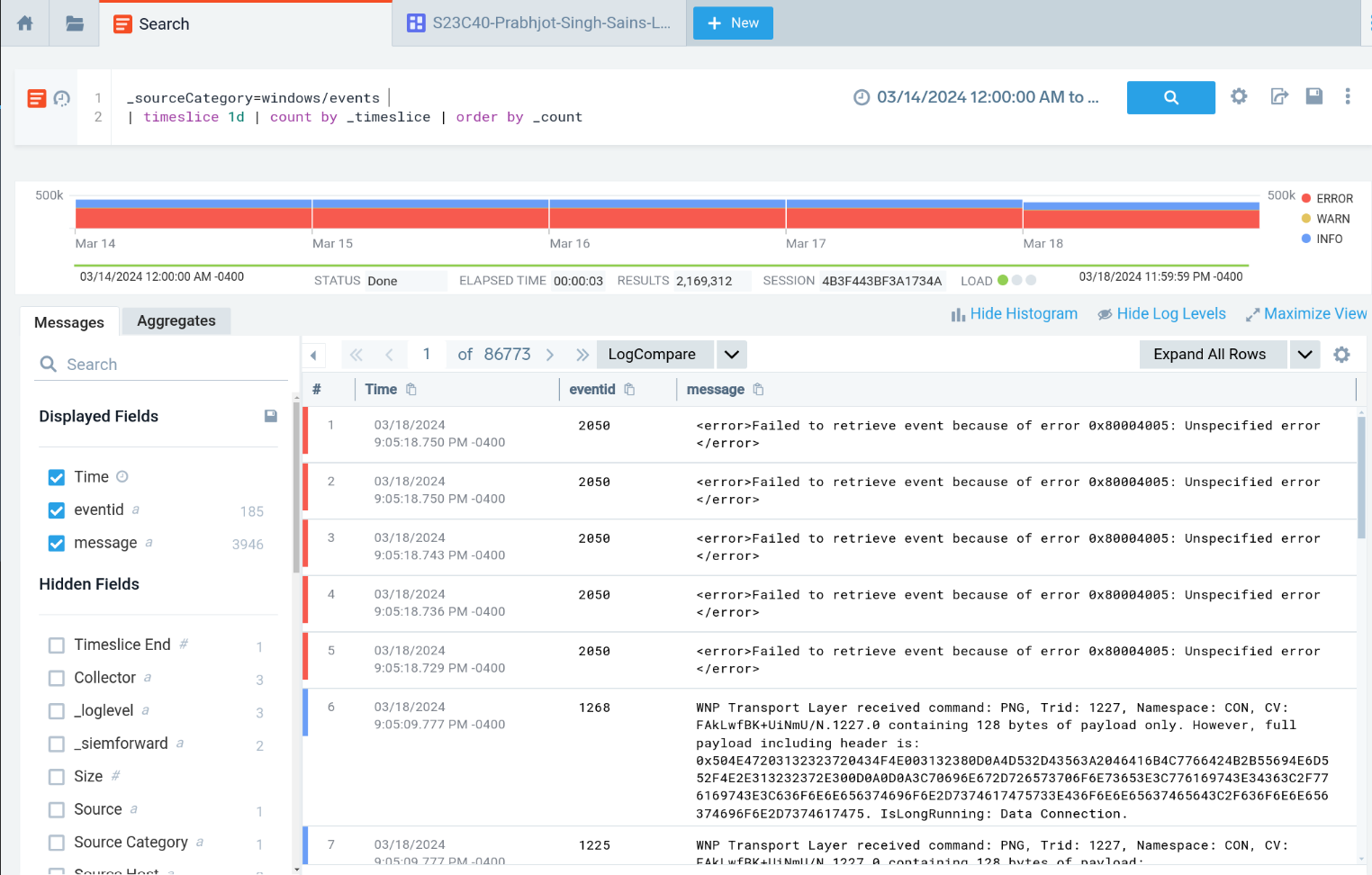


Fig 21. A query showing the number of events each day.

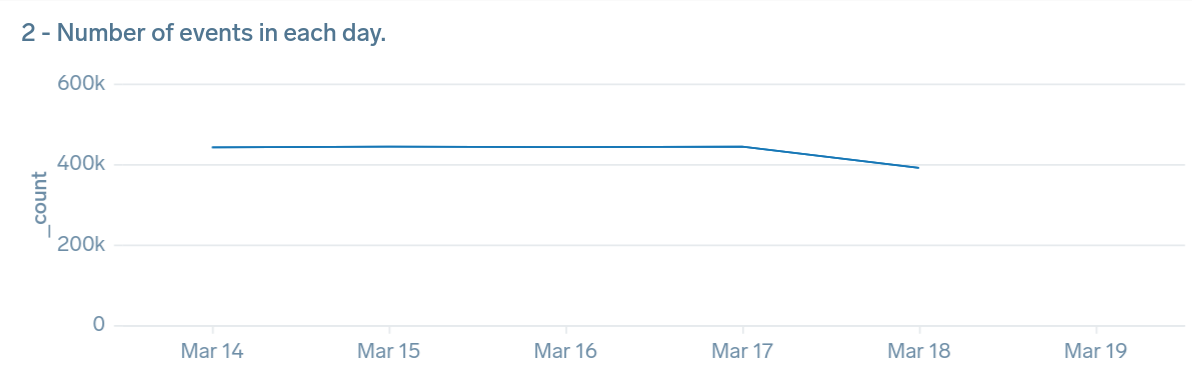


Fig 22. Panel 2 Showing the number of events each day.

1. **Write a query and visualize the** **percentage of each severity level. (Panel03):**

\_sourceCategory=windows/events

| json field=\_raw "Level" as severity\_level | json field=\_raw "TimeCreated" | count by severity\_level

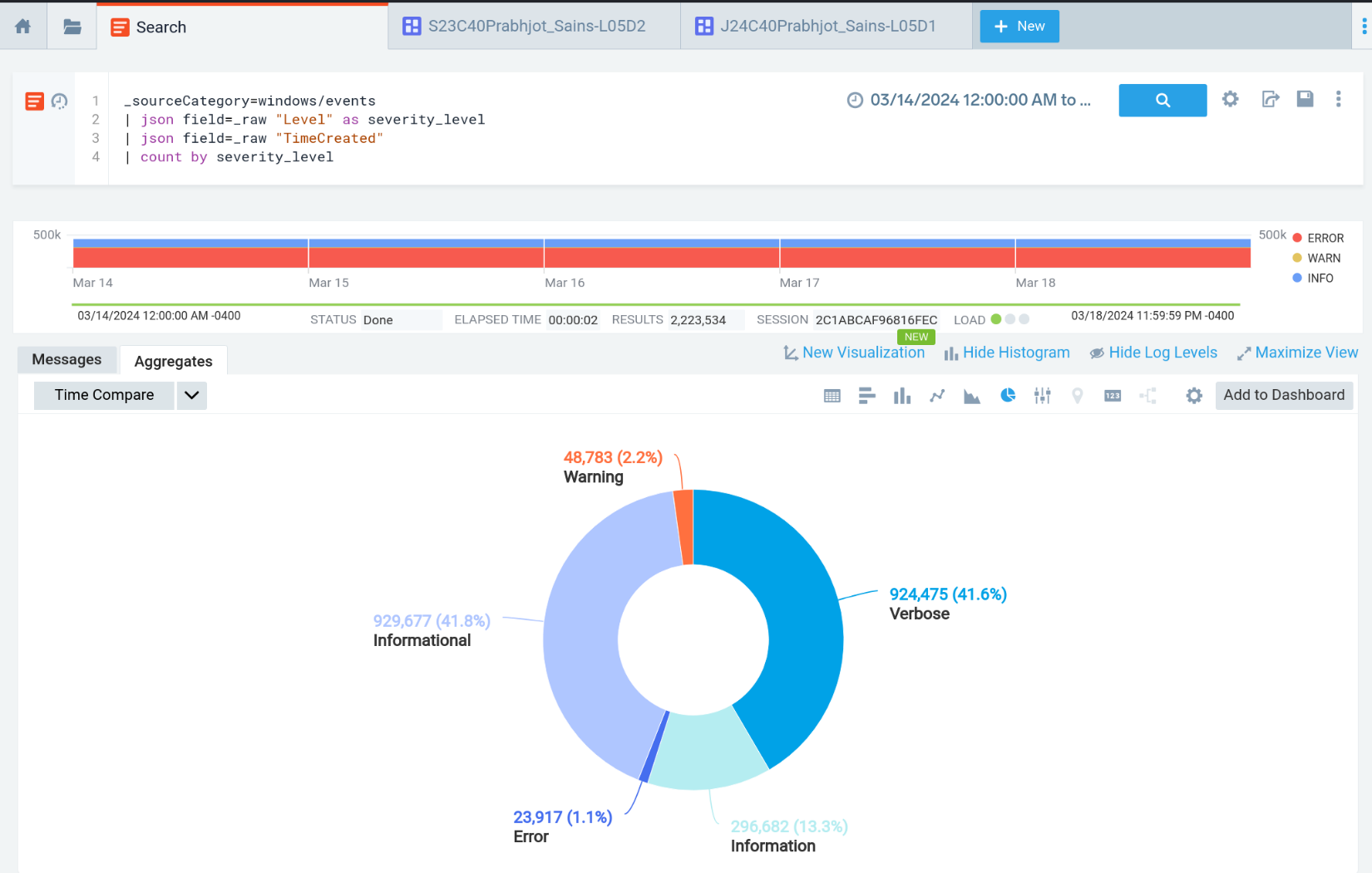


Fig 23. Query showing percentage of each severity level.

1. **Shared the Dashboard as S24C40Prabhjot\_Sains-L05D2 on SumoLogic**

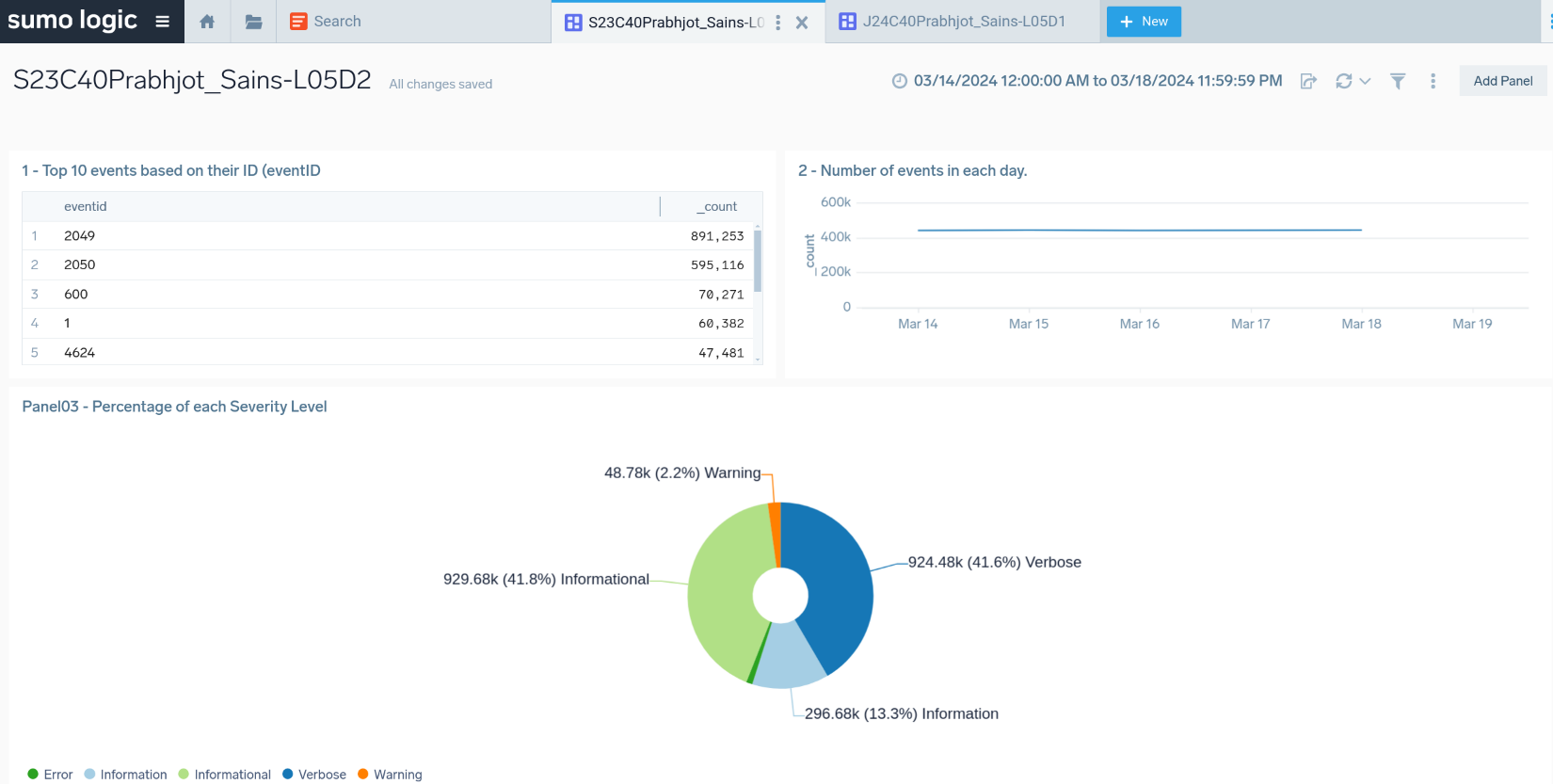


Fig 24. Dashboard for windows events