

6/5/2025

Scheduling reports and alerts in Splunk: A Practical Guide

RESEARCH REPORT



Balaji Varaprasad
CYBERSAPIENS

Scheduled Reports and Alerts in Splunk

In Splunk, the overall configuration of scheduled reports and alerts revolves around defining a search query, scheduling its execution, and optionally triggering actions based on the results. Everything begins with creating a search using SPL (Search Processing Language), which filters and analyses the relevant data, for example, failed login attempts or error logs. This search can then be saved as a report, which can be scheduled to run automatically at specified intervals such as hourly, daily, or weekly. The report can be configured to output results in various formats like tables or charts, and it can be emailed, downloaded, or added to dashboards for regular monitoring.

Alerts are a specialized form of scheduled search that not only run periodically but also include trigger conditions. When these conditions are met, such as the number of failed logins exceeding a threshold, Splunk can automatically execute predefined actions. These actions include sending email notifications, running scripts, sending data to a webhook, or alerting via Slack or other integrations. Users can also configure alert throttling or suppression settings to avoid repeated alerts for the same issue. Overall, this configuration ensures that important events in your data are not only captured and summarized but also responded to in a timely and automated manner.

Types of Data Sources Monitored by Scheduled Reports and Alerts in Splunk:

In Splunk, the data sources monitored or analysed by scheduled reports and alerts come from a wide variety of machine-generated data collected across systems, applications, and networks. These sources typically include:

1. System Logs

- Operating system logs (e.g., /var/log/ in Linux, Windows Event Logs)

- Tracks login attempts, system errors, shutdowns, and user activity

2. Security Logs

- Firewall logs, antivirus logs, intrusion detection/prevention systems (IDS/IPS)
- Useful for detecting threats, failed login attempts, port scans, and policy violations

3. Application Logs

- Logs generated by custom or third-party applications (e.g., Apache, Nginx, MySQL)
- Monitor error rates, usage patterns, performance issues, and crashes

4. Network Traffic

- Data from routers, switches, and network monitoring tools (e.g., NetFlow, SNMP)
- Helps analyze bandwidth usage, detect anomalies, or identify malicious traffic

5. Authentication and Access Logs

- Logs from Active Directory, LDAP, Okta, or other identity systems
- Track who accessed what, when, and from where, key for access control monitoring

6. Cloud Services Logs

- Logs from AWS CloudTrail, Azure Monitor, Google Cloud Logging, etc.
- Provide insights into cloud resource access, configuration changes, and security events

7. Web Server Logs

- Access logs, error logs from web servers like Apache, IIS, or Nginx
- Monitor website traffic, detect attacks (e.g., SQL injection), or troubleshoot issues

Scheduled reports and alerts in Splunk typically monitor security events, system performance, user activities, and application behaviour across on-premise, cloud, and hybrid environments. The specific data sources depend on what has been onboarded into Splunk, but the goal is always to identify patterns, detect anomalies, and automate response actions where needed.

Data Sources Monitored in our case:

The primary data source for this use case is, index=_audit: This internal index logs Splunk's authentication attempts, including failed logins.

This data source is essential for tracking authentication behaviour and potential brute-force or unauthorized access attempts within Splunk itself.

Practical: Scheduling reports and creating alerts

Task: Set up a scheduled report that provides a summary of failed login attempts over the last 24 hours and configure an alert to trigger whenever the number of failed login attempts exceeds a predefined threshold. (index=_audit failedlogin).

Failed Login Query to Create a scheduled report on failed logins in 24 hours:

Splunk's internal audit logs (stored in index _audit) record web interface logins and other admin actions. Failed login attempts are captured with fields like action="login attempt" and an info=failed flag. **For example:**

```
index=_audit action="login attempt" info="failed" | stats count by user, _time
```

This query (with a time range of the last 24 hours) returns the count of failed Splunk Web logins per user. You can further refine or group by user if needed. As Splunk docs note, you should search _audit to review login events.

Step 1:

Open the Splunk Enterprise application and log in using your administrative credentials to begin the practical configuration process.

The screenshot shows the Splunk Enterprise home page. At the top, there's a navigation bar with links for Home, 127.0.0.1:8000/en-US/app/launcher/home, and various system status indicators like chatgpt, Gmail, YouTube, and Maps. Below the navigation is the Splunk logo and the text "splunk>enterprise". On the right side of the header, there are links for Administrator, Messages, Settings, Activity, Help, and a Find bar. The main content area starts with a greeting "Hello, Administrator". Below it is a "Bookmarks" section with categories: My bookmarks (0), Shared with my organization (0), Shared by me, and Shared by other administrators. A "Splunk recommended (13)" section follows, containing six cards under the heading "Common tasks": Add data, Search your data, Visualize your data, Manage alerts, Add team members, and Manage permissions.

Step 2:

- Navigate to the Search your data. Set the time range filter to Last 24 hours, then enter your SPL (Search Processing Language) query to extract failed login attempts from the internal audit logs.

This screenshot is identical to the one above, showing the Splunk Enterprise home page. It features the same header, "Hello, Administrator" message, and "Splunk recommended" section. A prominent red arrow points to the "Search your data" card in the "Common tasks" section, highlighting it as the next step in the process.

- o index=_audit action="login attempt" info=failed
- o This retrieves all failed login attempts recorded in the last 24 hours.

The screenshot shows the Splunk search interface. At the top, there is a search bar with the placeholder "enter search here...". To the right of the search bar are buttons for "Last 24 hours" and a magnifying glass icon. Below the search bar is a dropdown menu labeled "No Event Sampling". Further down are sections for "Search History" and "How to Search", which includes links to "Documentation", "Tutorial", and "Data Summary". On the right side, there is a section titled "Analyze Your Data with Table Views" with a "Create Table View" button.

The screenshot shows the search results page with the title "New Search". The search query is "index=_audit action='login attempt' info='failed'". The results section displays "3 events" from "6/4/25 12:30:00.000 PM to 6/5/25 12:38:15.000 PM". The results table has columns for "Time" and "Event". The first two events are from host "BALAJI" and the third event is from host "balaji". All events belong to index "_audit" and source "audittrail" with sourcetype "audittrail". The time for the first event is 11:34:09.507 AM and for the second event is 11:33:59.833 AM. The time for the third event is 3:28:48.709 PM. The events are Audit:[timestamp] user=[user] action=[action] info=[info].

Time	Event
6/5/25 11:34:09.507 AM	Audit:[timestamp=06-05-2025 11:34:09.507, user=asnabajshch, action=login attempt, info=failed, src=127.0.0.1 method=Splunk]
6/5/25 11:33:59.833 AM	Audit:[timestamp=06-05-2025 11:33:59.833, user=wfefw, action=login attempt, info=failed, src=127.0.0.1 method=Splunk]
6/4/25 3:28:48.709 PM	Audit:[timestamp=06-04-2025 15:28:48.709, user=balaji varaprasad, action=login attempt, info=failed, src=127.0.0.1 method=Splunk]

Explanation:

- **index=_audit**
This specifies that the search should be performed within the `_audit` index. The `_audit` index in Splunk stores logs related to user activity, authentication, and system-level operations. It is commonly used for auditing and monitoring administrative actions, including logins.
- **action="loginattempt"**
This filter narrows down the search to events where a user tried to log in. The `action` field captures the type of event that occurred, in this case, a login attempt.
- **info=failed**
This condition further filters the results to only include **failed** login attempts. The `info` field indicates the outcome of the event, and `failed` shows that the login was not successful.

Step 3:

- Now, let's further filter and organize the data to make it more insightful. We will extract and display only the username, along with the count of failed login attempts and the corresponding time.
- Add, `| stats count by user, _time` spl query to before query to filter our data.

The screenshot shows the Splunk Enterprise search interface. The search bar contains the query: `index=_audit action="login attempt" info=failed | stats count by user, _time`. The Statistics tab is selected. The search results table displays three events, each with a user name, a timestamp, and a count of 1. The columns are labeled `user`, `_time`, and `count`.

user	_time	count
asnabajshch	2025-06-05 11:34:09.507	1
balaji varaprasad	2025-06-04 15:28:48.709	1
wfefw	2025-06-05 11:33:59.833	1

Explanation:

- stats count by user, _time: This command groups the results by **user** and **timestamp**, and counts the number of failed login attempts per user at specific times.
- This helps in identifying which users are experiencing repeated login failures and when they occurred.

Step 4:

- Once the desired results are generated, proceed to save the search as a report in order to schedule it for regular execution.

To do this:

- Click on Save As → Report in the top right corner of the search window.
- Provide a suitable Report Title and optional Description.
- Set the report permissions if needed (private or shared).

Note: If you are unfamiliar with the steps to save a report in Splunk, you can refer to my previous research report published on LinkedIn for detailed guidance.

The screenshot shows the Splunk 9.4.2 interface. At the top, there's a search bar with the URL: 127.0.0.1:8000/en-US/app/search/search?q=search%20index%3D_audit%20action%3D"login%20attempt"%20info%3Dfailed%20%7C%20stats%20count%20by%20user%2C%20_time&display.p... . Below the search bar is the Splunk navigation bar with links for enterprise, Apps, Search, Analytics, Datasets, Reports, Alerts, and Dashboards. The 'Search' tab is selected. In the main area, there's a 'New Search' section with a search bar containing the command: index=_audit action="login attempt" info=failed | stats count by user, _time. Below the search bar, it says '3 events (6/4/25 12:30:00.000 PM to 6/5/25 12:48:21.000 PM)' and 'No Event Sampling'. Under the search bar, there are tabs for Events, Patterns, Statistics (3), and Visualization, with 'Statistics (3)' being the active tab. Below these tabs are buttons for 'Show: 20 Per Page', 'Format', and 'Preview: On'. The main table displays three events with columns for user, _time, and count. The user column lists 'asnbajshch', 'balaji varaprasad', and 'wfefw'. The _time column lists '2025-06-05 11:34:09.507', '2025-06-04 15:28:48.709', and '2025-06-05 11:33:59.833'. The count column lists '1', '1', and '1' respectively. In the top right corner, there's a 'Save As' button with a dropdown menu. A red arrow points from the 'Save As' button to the dropdown menu. Another red arrow points specifically to the 'Report' option in the dropdown menu. The dropdown also includes options for Alert, Job, Existing Dashboard, New Dashboard, and Event Type.

user	_time	count
asnbajshch	2025-06-05 11:34:09.507	1
balaji varaprasad	2025-06-04 15:28:48.709	1
wfefw	2025-06-05 11:33:59.833	1

The image consists of three vertically stacked screenshots of the Splunk 9.4.2 web interface. Each screenshot shows a 'Save As Report' dialog box over a search results page.

Screenshot 1: The 'Save As Report' dialog is open. It contains fields for 'Title' (empty), 'Description' (optional), 'Content' (Statistics Table), and a 'Time Range Picker' set to 'Yes'. At the bottom are 'Cancel' and 'Save' buttons. A red arrow points from the 'Title' field to the 'audit_failedlogins_report' text.

Screenshot 2: The 'Save As Report' dialog has been updated. The 'Title' field now contains 'audit_failedlogins_report' and the 'Description' field contains 'scheduled report'. A red arrow points from the 'Save' button to the 'Save' button at the bottom right of the dialog.

Screenshot 3: A 'Your Report Has Been Created' modal is displayed. It includes a message about viewing the report, additional settings (Permissions, Schedule, Acceleration, Embed), and three buttons: 'Continue Editing', 'Add to Dashboard', and a green 'View' button. A red arrow points from the 'View' button to the 'View' button in the modal.

- After creating click on view to open the created report.

Step 5:

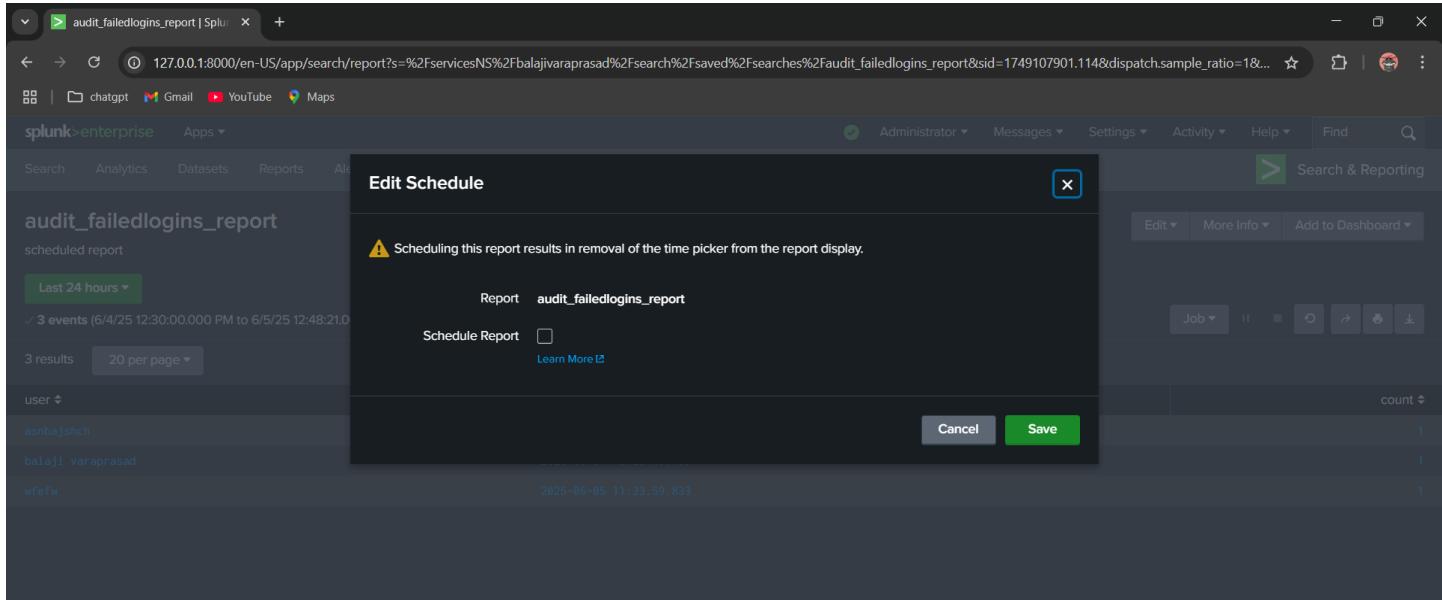
- After opening the report you just created, click on the **Edit** option located at the top-right corner of the report interface. From the dropdown, select **Edit Schedule** to configure the scheduling settings.
- This allows you to automate the report to run at specified intervals (e.g., daily, hourly, or using a custom cron expression), ensuring regular monitoring without manual execution.

The screenshot shows the Splunk Search & Reporting interface. At the top, there's a navigation bar with links for 'chatgpt', 'Gmail', 'YouTube', and 'Maps'. Below that is the Splunk logo and 'enterprise'. The main area displays a report titled 'audit_failedlogins_report' which is a 'scheduled report'. It shows a green button for 'Last 24 hours' and a message indicating '3 events' from June 4, 2025, to June 5, 2025. Below this, there are three rows of event data with columns for 'user', '_time', and 'count'. In the top right corner, there's an 'Edit' button with a dropdown arrow. A red arrow points from this button to the 'Edit Schedule' option in the dropdown menu. Other options in the dropdown include 'Open in Search', 'Edit Description', 'Edit Permissions', 'Edit Acceleration', 'Clone', 'Embed', and 'Delete'.

This screenshot shows the same Splunk interface as the previous one, but the 'Edit' dropdown menu is now fully open. The 'Edit Schedule' option is highlighted with a red arrow. The other options in the menu are: 'Open in Search', 'Edit Description', 'Edit Permissions', 'Edit Acceleration', 'Clone', 'Embed', and 'Delete'. The rest of the interface remains the same, showing the 'audit_failedlogins_report' scheduled report with its event data.

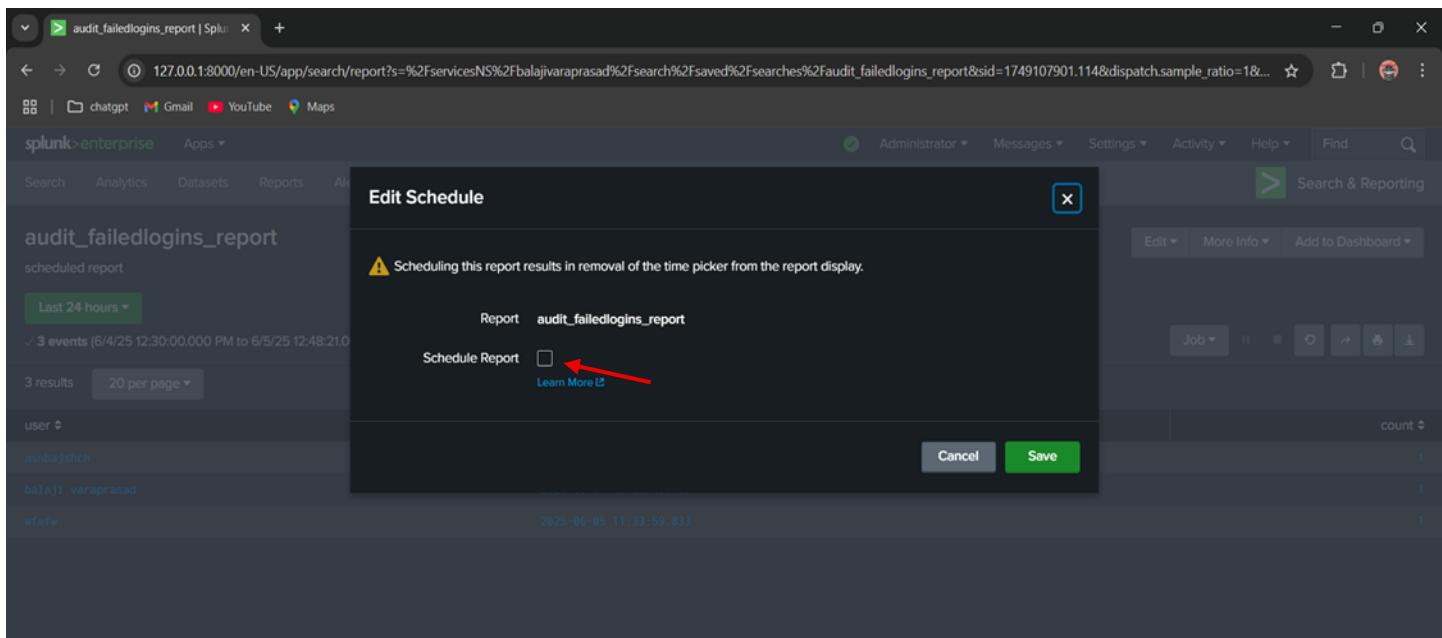
Step 6:

After clicking on Edit Schedule, a pop-up window will appear where you can define the scheduling parameters for the report.



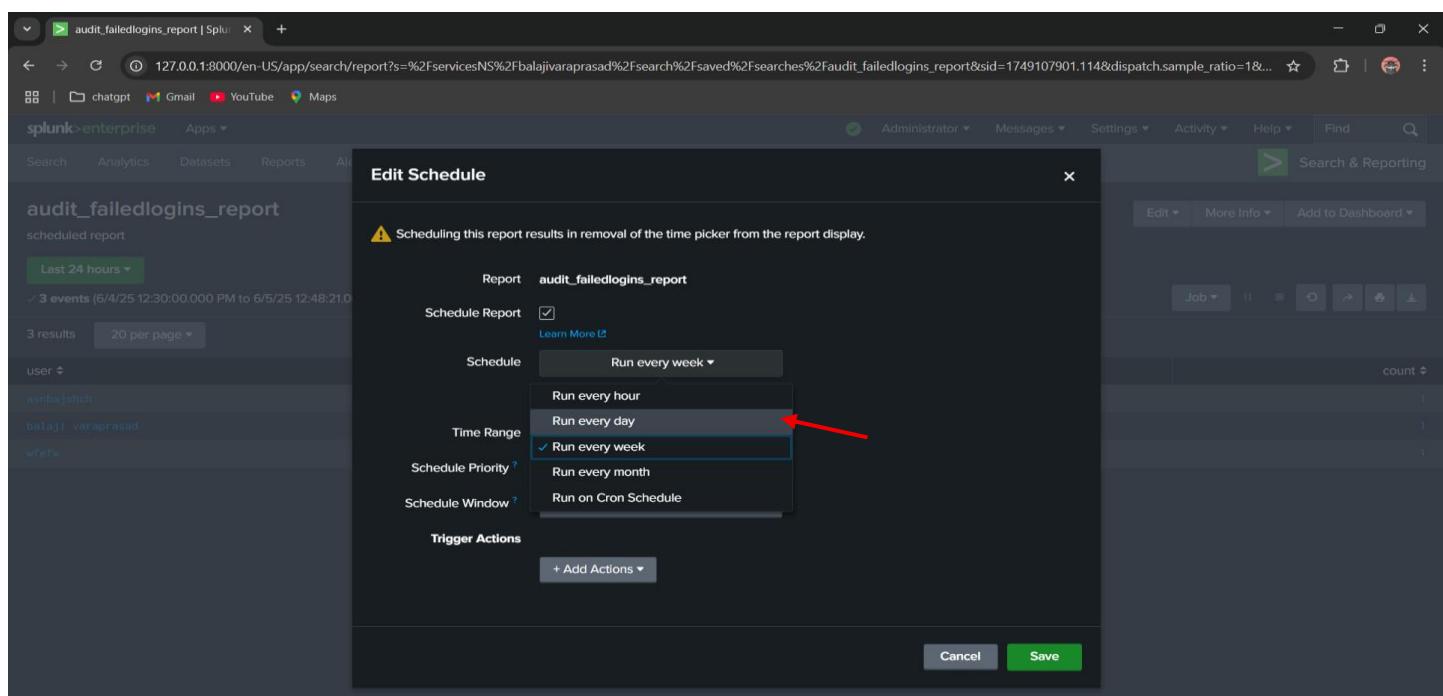
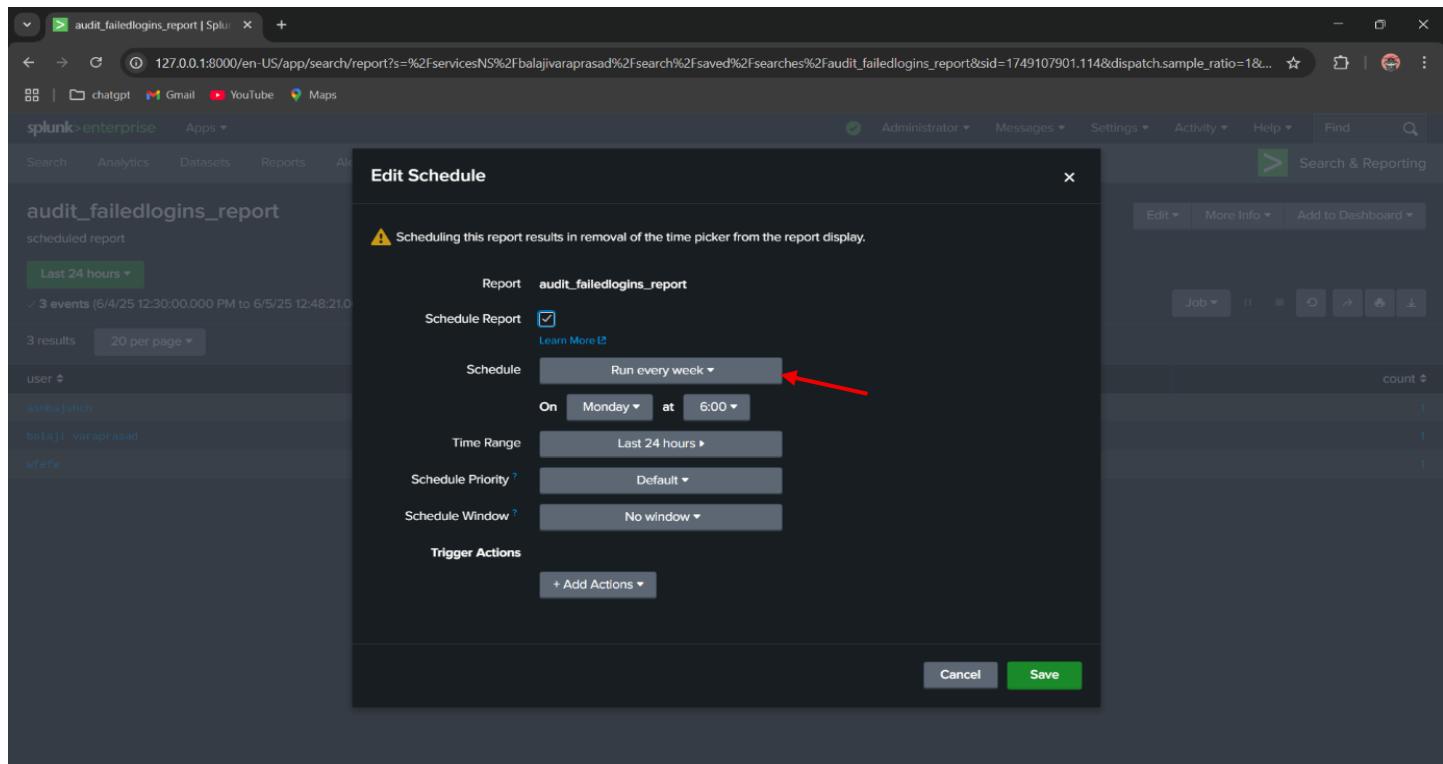
Step 7:

- In the schedule configuration pop-up, click on the checkbox labeled "Schedule Report" to enable the scheduling settings.
- Once the checkbox is selected, the scheduling fields become editable, allowing you to define when and how often the report should run. This step is essential to activate the automated scheduling feature for your report.



Step 8:

- In the same pop-up window, locate the dropdown menu next to the Schedule label. Click on it to open the options for how frequently the report should run.
- For this example, I selected Daily to ensure the report runs once every day.
- You can select the frequency based on your report's purpose and monitoring needs.



Step 9:

- After selecting the report frequency, configure the Time Range for which the report should retrieve data during each scheduled run.
- For this case, select "Last 24 hours" to ensure the report captures all failed login attempts from the previous day every time it runs.

The screenshot shows the Splunk interface with the URL http://127.0.0.1:8000/en-US/app/search/report?locale=en-US&s=audit_failedlogins_report&sid=balajivaraprasad_balajivaraprasad_search_RMD5806b870d7016ad65_at_1749116056_2&display.page. A modal window titled 'Edit Schedule' is open. Inside, under the 'Schedule Report' section, the 'Run every day' option is selected at 0:00. The 'Time Range' dropdown is set to 'Last 24 hours', which is highlighted with a red arrow. Other settings include 'Default' for 'Schedule Priority' and 'No window' for 'Schedule Window'. At the bottom right of the modal are 'Cancel' and 'Save' buttons.

The screenshot shows the Splunk interface with the same URL as the previous screenshot. A modal window titled 'Select Time Range' is open. Under the 'RELATIVE' section, the 'Last 24 hours' option is selected and highlighted with a red arrow. Other relative time options include Today, Week to date, Business week to date, Month to date, Year to date, Yesterday, Previous week, Previous business week, Previous month, and Previous year. Under the 'OTHER' section, options include Last 15 minutes, Last 60 minutes, Last 4 hours, All time, Last 7 days, and Last 30 days. At the bottom of the modal are 'Back' and 'Advanced' buttons.

Step 10:

- Once the time range is configured, proceed to set the Schedule Priority for the report.
- In the scheduling window, you'll find a Priority dropdown menu. This determines the execution priority of the report in relation to other scheduled searches in the system.

Options typically include:

- Default
 - Higher
 - Lower
- For most use cases, selecting Default is sufficient. However, if the report is critical and needs to run ahead of others, you may choose Higher priority.
 - In this example, I selected High Priority to ensure the report runs with precedence over lower-priority scheduled searches.

The screenshot shows the Splunk interface with the URL http://127.0.0.1:8000/en-US/app/search/report?locale=en-US&s=audit_failedlogins_report&sid=balajivaraprasad_balajivaraprasad_search_RMD5806b870d7016ad65_at_1749116056_2&display.page. The main area displays a search result for 'audit_failedlogins_report' with 4 events from June 4, 2025, to June 5, 2025. The 'Edit Schedule' dialog is open, showing the following configuration:

- Report:** audit_failedlogins_report
- Schedule Report:** checked
- Schedule:** Run every day
- At:** 0:00
- Time Range:** Last 24 hours
- Schedule Priority:** Default (highlighted with a red arrow)
- Schedule Window:** No window
- Trigger Actions:** + Add Actions

The 'Save' button is visible at the bottom right of the dialog.

The screenshot shows the 'Edit Schedule' dialog for a report named 'audit_failedlogins_report'. The 'Trigger Actions' dropdown menu is open, displaying three options: 'Default' (selected), 'Higher', and 'Highest'. A red arrow points to the 'Higher' option. At the bottom right of the dialog are 'Cancel' and 'Save' buttons.

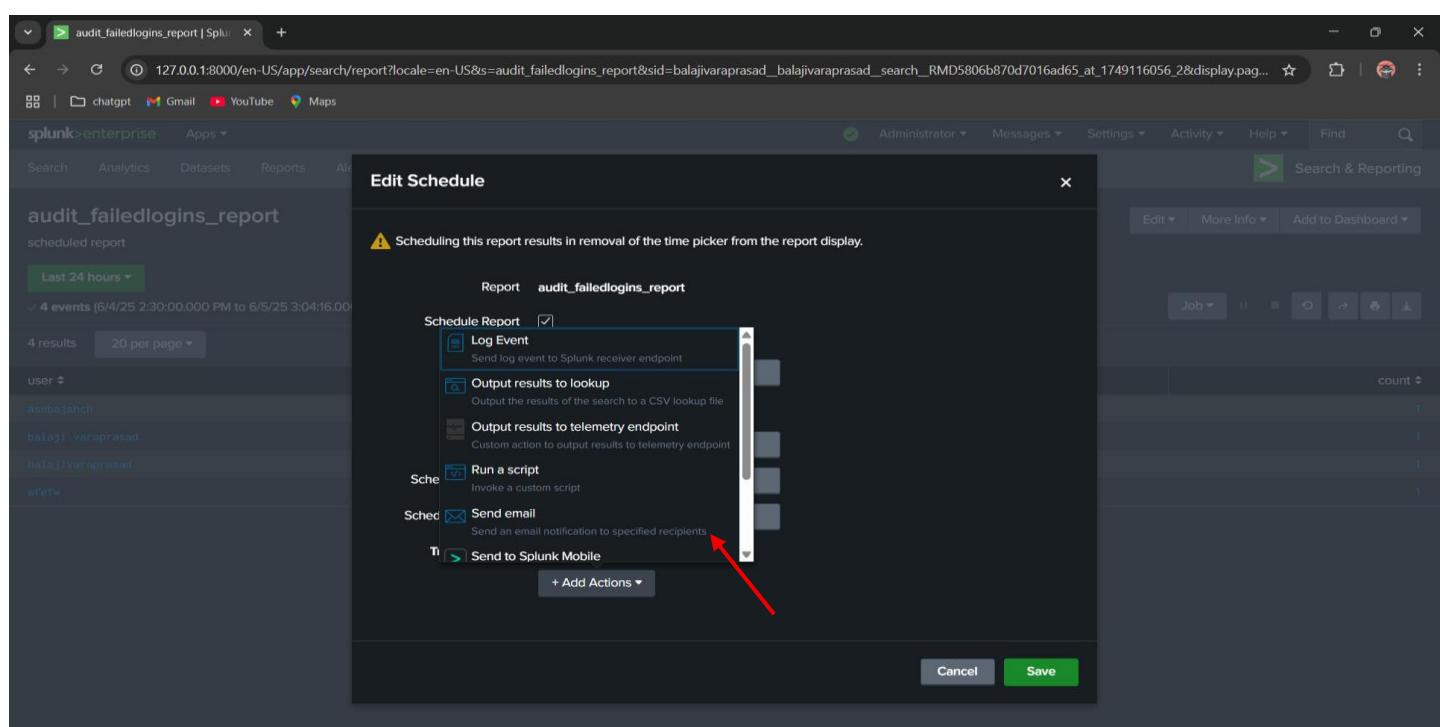
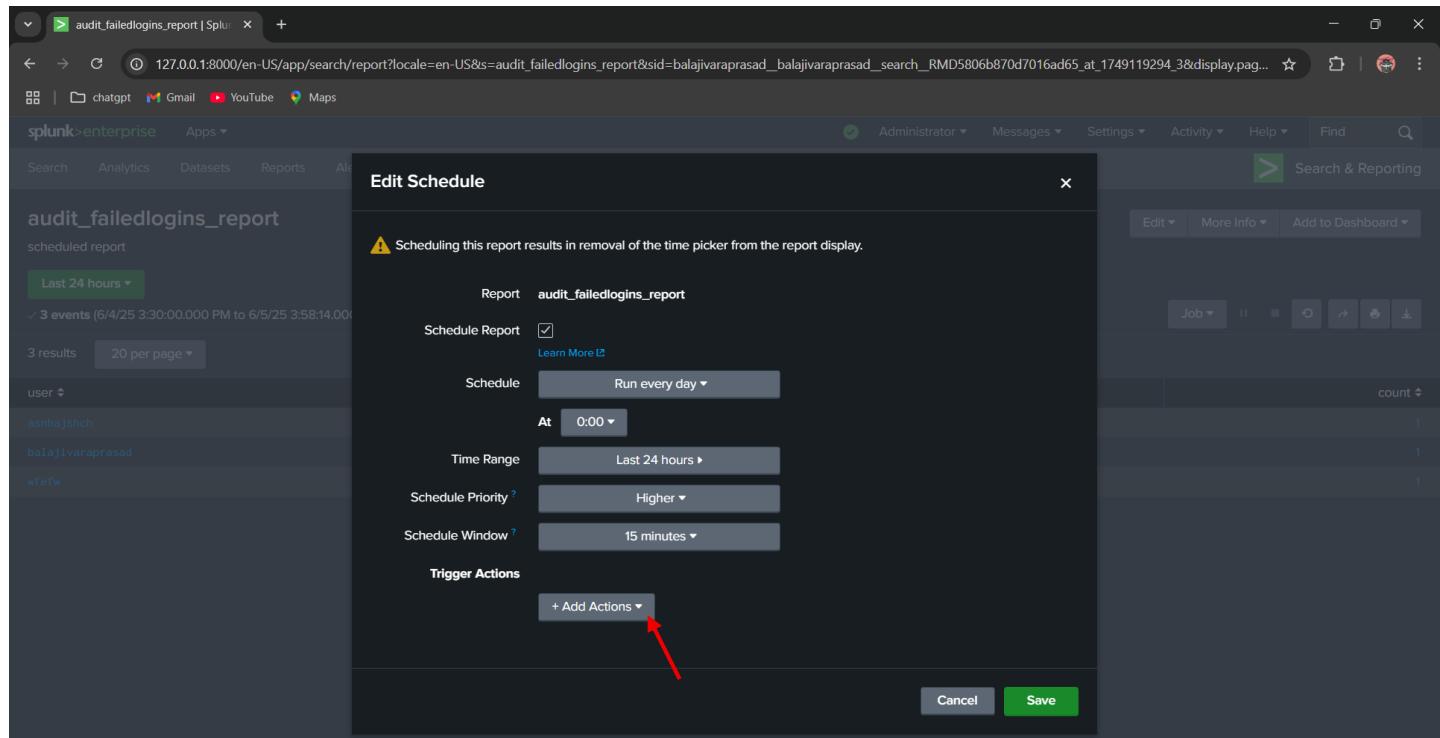
Step 11:

- Next, set the Schedule Window, which defines the allowed time range during which Splunk can run the scheduled report if it misses the exact scheduled time.
- This is useful in environments with limited resources or heavy search loads. For example, if the report is scheduled to run at 6:00 AM but the system is busy, the schedule window allows Splunk to execute the report slightly later (e.g., within 5 or 10 minutes).

The screenshot shows the 'Edit Schedule' dialog for the same report. The 'Schedule Window' dropdown menu is open, listing several options: 'Auto', 'No window', '5 minutes', '15 minutes' (selected), '30 minutes', '1 hour', '2 hours', '4 hours', and '8 hours'. Two red arrows point to the '15 minutes' option and the '15 minutes' button in the 'Schedule Window' field. The 'Save' button is visible at the bottom right.

Step 12:

- In the Trigger Actions section of the scheduling window, click on “Add Actions” and select “Send Email” to configure email notifications for the report.
- This allows Splunk to automatically send the generated report to your specified email address upon each scheduled run.



Step 13:

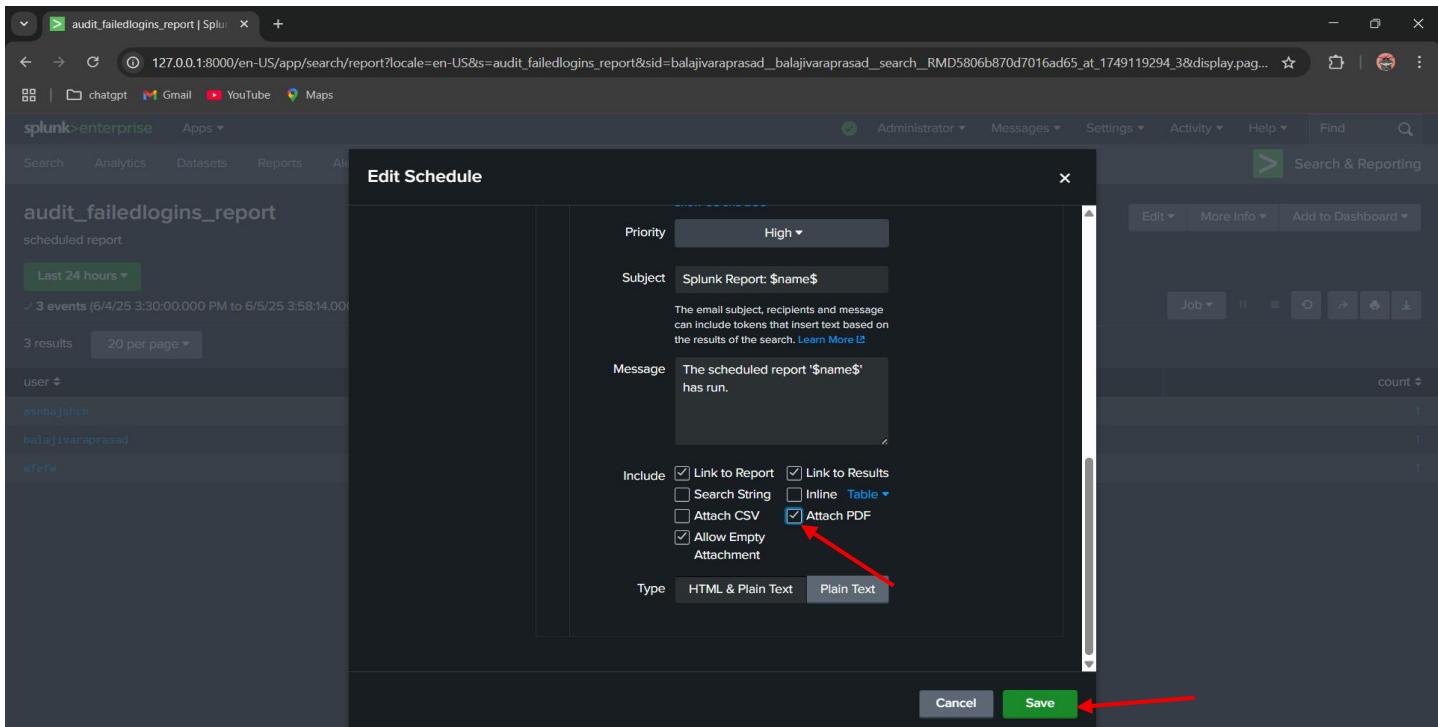
- After selecting the Send Email action, the next step is to configure the email settings. Begin by entering the recipient's email address in the "To" field, this is where the scheduled report will be delivered. You can add multiple addresses if needed, separated by commas.
- Finally, set the email priority to High to ensure the alert is treated with urgency by the recipient's mail system. These settings help ensure that critical security insights are promptly delivered and noticed.

The screenshot shows the Splunk interface with the search bar at the top containing the query 'audit_failedlogins_report'. Below the search bar, there is a sidebar with 'Audit' selected. The main area displays a search result for 'audit_failedlogins_report' with 4 events found over the last 24 hours. A modal window titled 'Edit Schedule' is open, specifically for the 'Send email' action. The 'To' field is populated with 'changeme@example.com'. The 'Priority' dropdown is set to 'Normal'. A red arrow points to the 'Priority' dropdown, and another red arrow points to the 'To' field. The 'Subject' field contains 'Splunk Report: \$name\$', and the 'Message' field contains 'The scheduled report '\$name\$' has run.' At the bottom of the modal, there are 'Cancel' and 'Save' buttons.

This screenshot is similar to the one above, showing the 'Edit Schedule' dialog for the 'Send email' action. The 'To' field now contains 'techinfo@you9@gmail.com'. The 'Priority' dropdown is now set to 'High'. A red arrow points to the 'Priority' dropdown, and another red arrow points to the 'To' field. The other fields ('Subject' and 'Message') and the 'Include' section remain the same as in the previous screenshot. The 'Save' button is visible at the bottom of the dialog.

Step 14:

- In this final step, you can customize how the report will be shared via email.
- For this example, I chose to attach the report as a PDF, ensuring the recipient receives a clean, readable summary of failed login attempts.
- You may select whichever format suits your reporting needs. Once all the necessary options are selected, simply click the Save button to complete the configuration. This finalizes the scheduled report setup with automated email delivery.



Step 15:

- Your automated report with email delivery is now fully configured and ready to run as per the defined schedule.
- Below image shows, the scheduled report named audit_failedlogins_report has been successfully configured. It is set to run daily at 00:00 hours with a time range of the last 24 hours.
- Since the first scheduled run has not yet occurred, there are currently no results to display. This message confirms that the setup is complete and the system is waiting for the scheduled time to execute the report for the first time.
- Once the report runs at the scheduled time, Splunk will automatically generate the report and send it via email to the configured recipients.

The screenshot shows the Splunk Enterprise web interface. At the top, the URL is 127.0.0.1:8000/en-US/app/search/report?locale=en-US&s=audit_failedlogins_report&display.page.search.mode=smart&dispatch.sample_ratio=1&earliest=-24h%40h&latest=now. The page title is 'audit_failedlogins_report'. The top navigation bar includes links for Search, Analytics, Datasets, Reports, Alerts, Dashboards, and a 'Search & Reporting' button. On the right side of the header, there are links for Administrator, Messages, Settings, Activity, Help, and Find. Below the header, a message says 'This scheduled report runs daily, at 0:00. Its time range is last 24 hours.' A large central message states 'There are no results because the first scheduled run of the report has not completed.' Below this message is a grey button labeled 'Open in Search'.

- This confirms the report scheduling is active and that you'll start receiving automated reports from the next scheduled execution onward.

Alert Criteria and Configuration:



Alert Condition: Configure the alert to fire when failed logins exceed the threshold. For example, run the spl over the last 24h and set “Trigger alert when number of results > 3.” This ensures that any surge in failures (beyond 3) raises

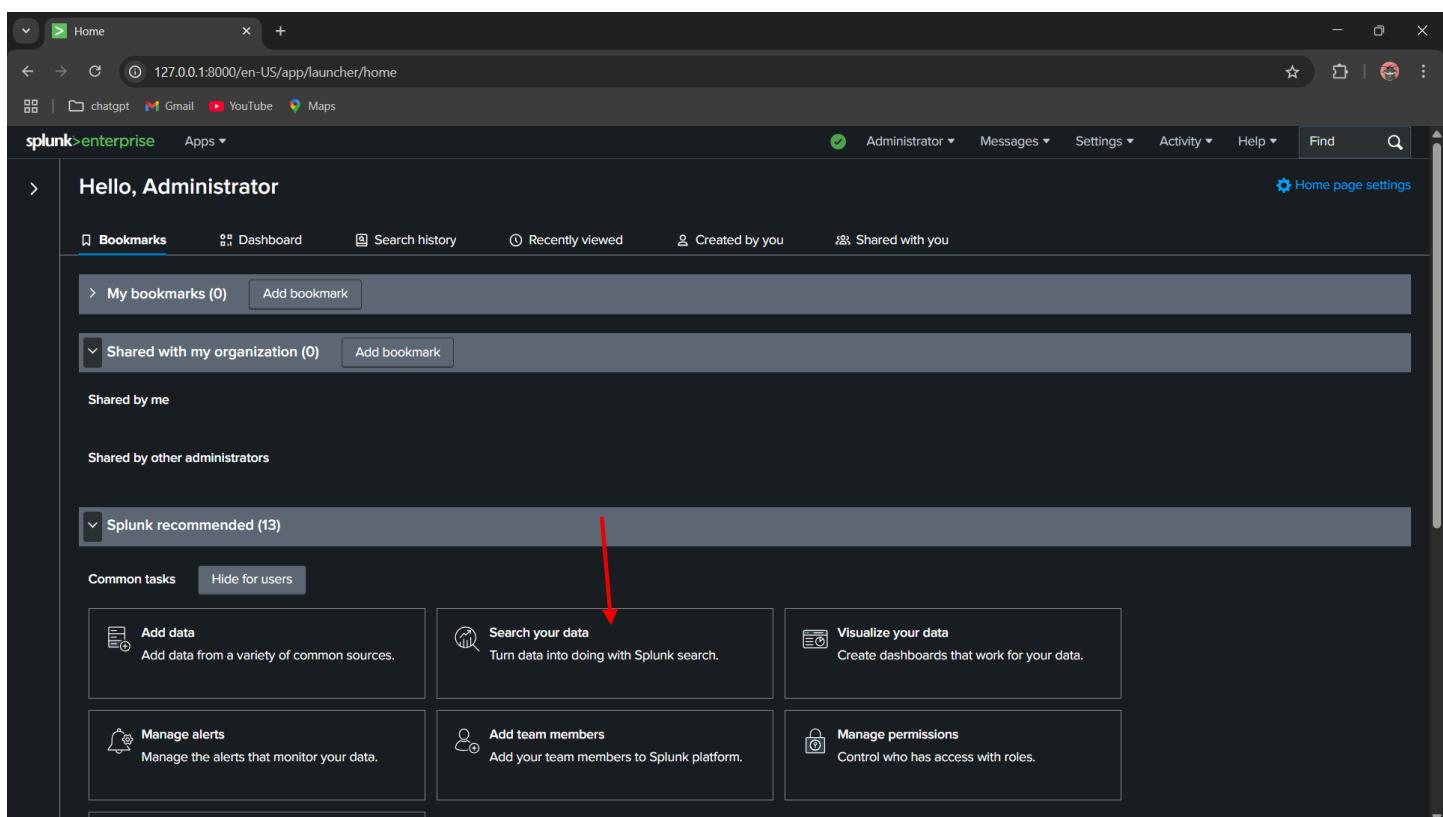
an alarm. Splunk will run the saved search on schedule and count the results, if the count is above 3, the alert triggers.

Example spl query:

```
index=_audit action="login attempt" info="failed" | stats count as failed_attempts by user, _time | where failed_attempts > 3
```

Step 1:

- Start by logging in to your Splunk instance using your credentials. Once logged in, navigate to the Search & Reporting app from the homepage.
- This is where you will run your search query, configure trigger conditions, and begin creating the alert based on specific criteria, in this case, failed login attempts.



Step 2:

- In the Search & Reporting app, enter the following SPL (Search Processing Language) query to extract failed login attempts from the internal audit index:
- `index=_audit action="login attempt" info="failed" | stats count as failed_attempts by user, _time`
- This query filters the audit logs to display only failed login attempts. It then uses the stats command to count the number of failed login attempts for each user over time.
- Set time filter to 24hrs and click on search.
- This is the same query used earlier for generating the scheduled report. Reusing it ensures consistency in data analysis and alerting.

The screenshot shows the Splunk 9.4.2 interface. The search bar contains the SPL query: `index=_audit action="login attempt" info="failed" | stats count as failed_attempts by user, _time`. A red arrow points from the text "Last 24 hours" in the search bar to the "Time Range" dropdown menu, which is set to "Last 24 hours". Another red arrow points from the green "Search" button to the "Search" button icon in the search bar. The interface includes a navigation bar with links like "chatgpt", "Gmail", "YouTube", and "Maps". The main menu bar has items like "splunk>enterprise", "Apps", "Administrator", "Messages", "Settings", "Activity", "Help", "Find", and a magnifying glass icon. Below the search bar, there are sections for "How to Search" and "Analyze Your Data with Table Views".

Step 3:

- The "Statistics" tab shows a table with three columns: user, _time, and failed_attempts. Each row represents a failed login event by a specific user along with the timestamp and the count of failed attempts.
- In this case, three users (asnbajshch, balajivaraPrasad, and wfefw) each have one failed login attempt within the last 24 hours.

The screenshot shows the Splunk 9.4.2 search interface. The search bar contains the SPL query: `index=_audit action="login attempt" info="failed" | stats count as failed_attempts by user, _time`. The results table displays 3 events from June 4, 2025, to June 5, 2025. The columns are user, _time, and failed_attempts. The data is as follows:

user	_time	failed_attempts
asnbajshch	2025-06-05 11:34:09.507	1
balajivaraprasad	2025-06-05 15:02:58.418	1
wfefw	2025-06-05 11:33:59.833	1

Step 4:

- To configure an alert, we need a condition that defines when the alert should be triggered. In this example, we'll use a threshold value of 3 failed login attempts. This means the alert will only trigger if the number of failed login attempts by any user exceeds 3 within the specified time range.
- Update your previous SPL query by appending a where clause as shown:
- `index=_audit action="login attempt" info="failed" | stats count as failed_attempts by user, _time | where failed_attempts > 3`
- This refined query filters out all results where failed login attempts are 3 or fewer and retains only the events that exceed the threshold, which is essential for setting up a meaningful and actionable alert.

The screenshot shows the Splunk 9.4.2 search interface. The search bar contains the SPL query: `index=_audit action="login attempt" info="failed" | stats count as failed_attempts by user, _time | where failed_attempts > 3`. A red arrow points from the text "No results found." at the bottom to the green search button icon in the top right corner. The results table is empty, displaying "No results found.".

Step 5:

- Even though there are 3 events displayed initially, we see no results after applying the threshold filter. This is because none of the users in the dataset have more than 3 failed login attempts, which is the threshold we defined in the where clause.
- Since our query now filters results to only include users with more than 3 failed login attempts, and none meet this condition, the final output is empty. This confirms that the alert logic is working as intended, it will only trigger when a genuine threshold breach occurs.

The screenshot shows the Splunk 9.4.2 search interface. The search bar contains the following query:
index=_audit action="login attempt" info="failed"
| stats count as failed_attempts by user, _time | where failed_attempts > 3

The results pane indicates "3 events" from "6/4/25 4:30:00.000 PM to 6/5/25 5:23:37.000 PM". Below the events, a message says "No results found." A red circle highlights the event count "3 events" and another red circle highlights the message "No results found."

Step 6:

Now that we've applied the threshold condition to detect users with more than 3 failed login attempts, the next step is to configure this search as an alert.

Step 7:

- To do this, click on the “Save As” button at the top right of the search window and select “Alert” from the dropdown.
- This allows us to save the search query and define the conditions under which an alert should be triggered, in this case, whenever failed login attempts exceed our set threshold.

The screenshot shows the Splunk search interface with a search bar containing a query: `index=_audit action="login attempt" info="failed" | stats count as failed_attempts by user, _time | where failed_attempts > 3`. Below the search bar, it says "3 events (6/4/25 4:30:00.000 PM to 6/5/25 5:23:37.000 PM) No Event Sampling". On the right, there's a "Save As" button with a dropdown menu. The "Alert" option is highlighted with a red arrow. Other options in the menu include "Report", "Existing Dashboard", "New Dashboard", and "Event Type". The status bar at the bottom says "No results found."

Step 8:

After clicking on “Save As → Alert,” a configuration dialog box appears where you need to provide the necessary details to set up the alert.

The screenshot shows the "Save As Alert" configuration dialog box. It has two main sections: "Settings" and "Trigger Conditions".
Settings:

- Title:** Title
- Description:** Optional
- Permissions:** Private (selected)
- Alert type:** Scheduled (selected)
 - Run every week
- On:** Monday
- at:** 6:00
- Expires:** 24 hour(s)

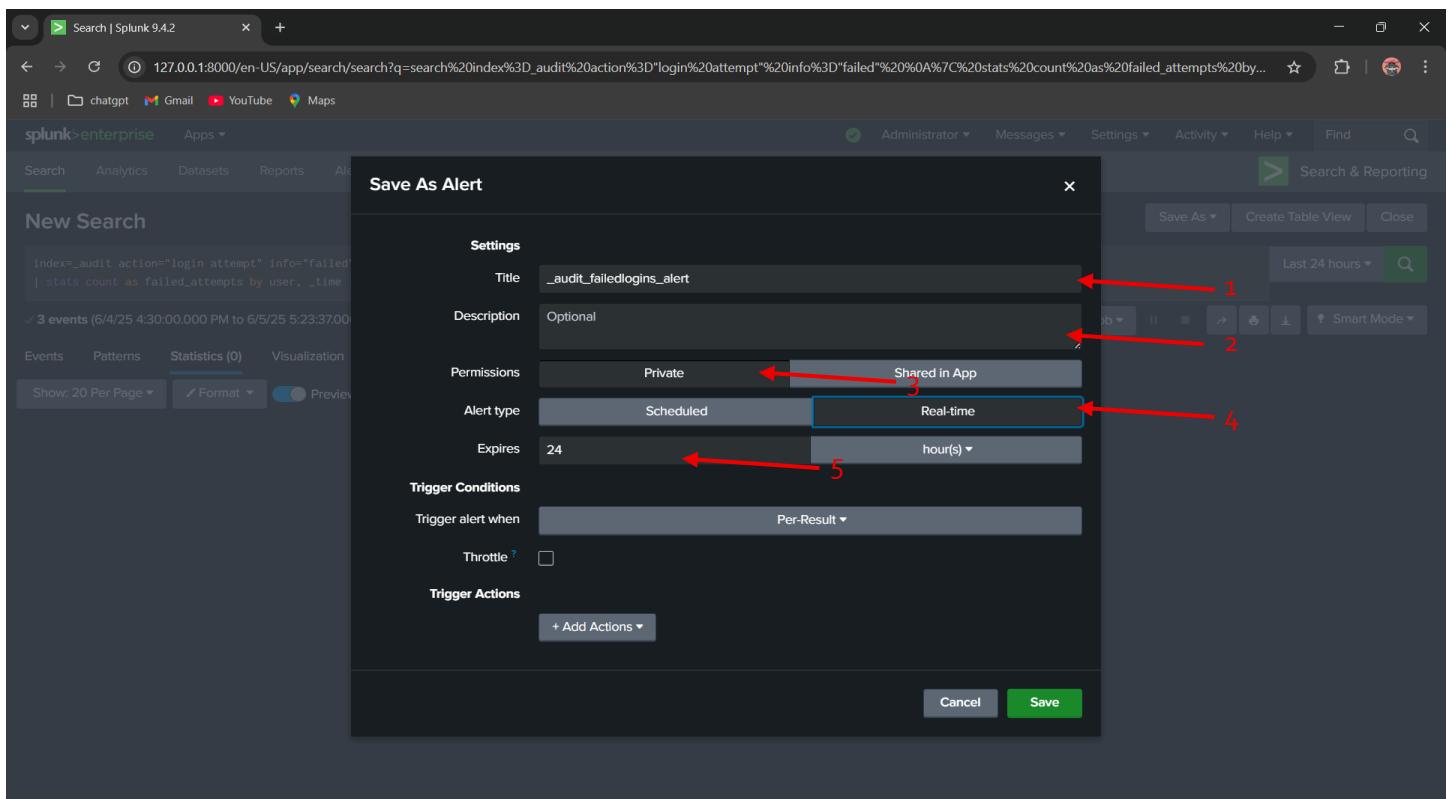
Trigger Conditions:

- Trigger alert when:** Number of Results
 - is greater than 0
- Trigger:** Once
- For each result:** (This part is partially visible)

At the bottom are "Cancel" and "Save" buttons.

Step 9:

1. **Title:** Enter a meaningful name for the alert. Here, I used `_audit_failedlogin_alert`.
2. **Description:** This field is optional. You can provide a brief explanation of the alert if needed.
3. **Permissions:** Leave it as Private to restrict visibility to only you.
4. **Alert Type:** Select Real-time to enable continuous monitoring and trigger alerts instantly when conditions are met.
5. **Expiration Time:** You can adjust this based on how long the alert should stay active. In this case, it is left at the default 24 hours, but you can set it to specific days or hours if required.



Step 10:

- In the "Trigger alert when" section, choose Per Result as the trigger condition.
- This means the alert will be triggered each time a single search result meets the defined condition, in this case, when a user has more than 3 failed login

attempts. This setting is ideal for real-time monitoring scenarios where you want to be notified every time a matching event occurs.

The screenshot shows the 'Save As Alert' dialog in Splunk 9.4.2. The 'Trigger Conditions' section is expanded, showing a dropdown menu with the following options:

- Per-Result** (selected, highlighted with a red arrow)
- Number of Results
- Number of Hosts
- Number of Sources
- Custom

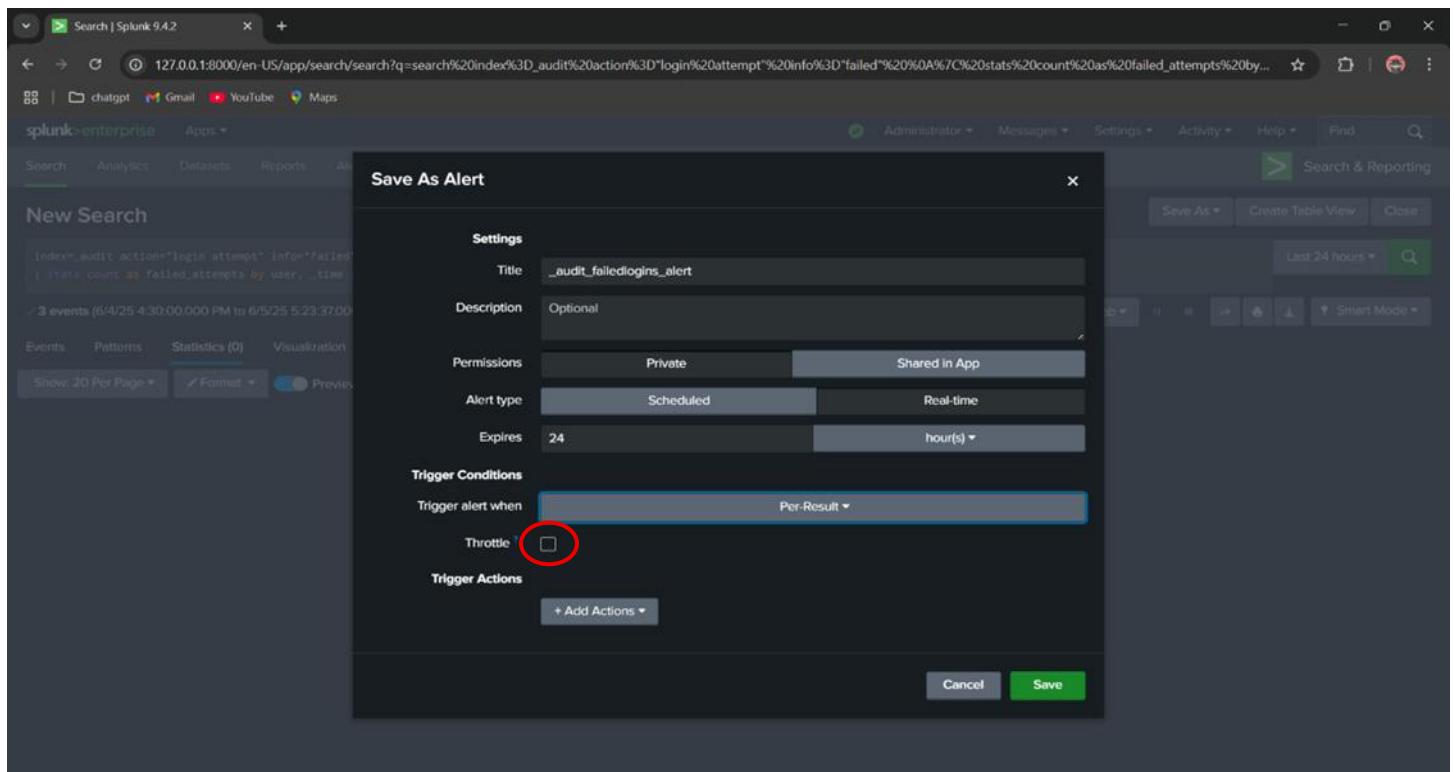
The 'Trigger Conditions' dropdown also has a red arrow pointing to it, indicating the selection of the 'Per-Result' option.

The screenshot shows the 'Save As Alert' dialog in Splunk 9.4.2 with the following settings:

- Settings**:
 - Title: `_audit_failedlog`
 - Description: Optional
 - Permissions: Shared in App
 - Alert type: Real-time
 - Expires: 24 hour(s)
- Trigger Conditions**: Trigger alert when: Per-Result (selected)
- Trigger Actions**: + Add Actions

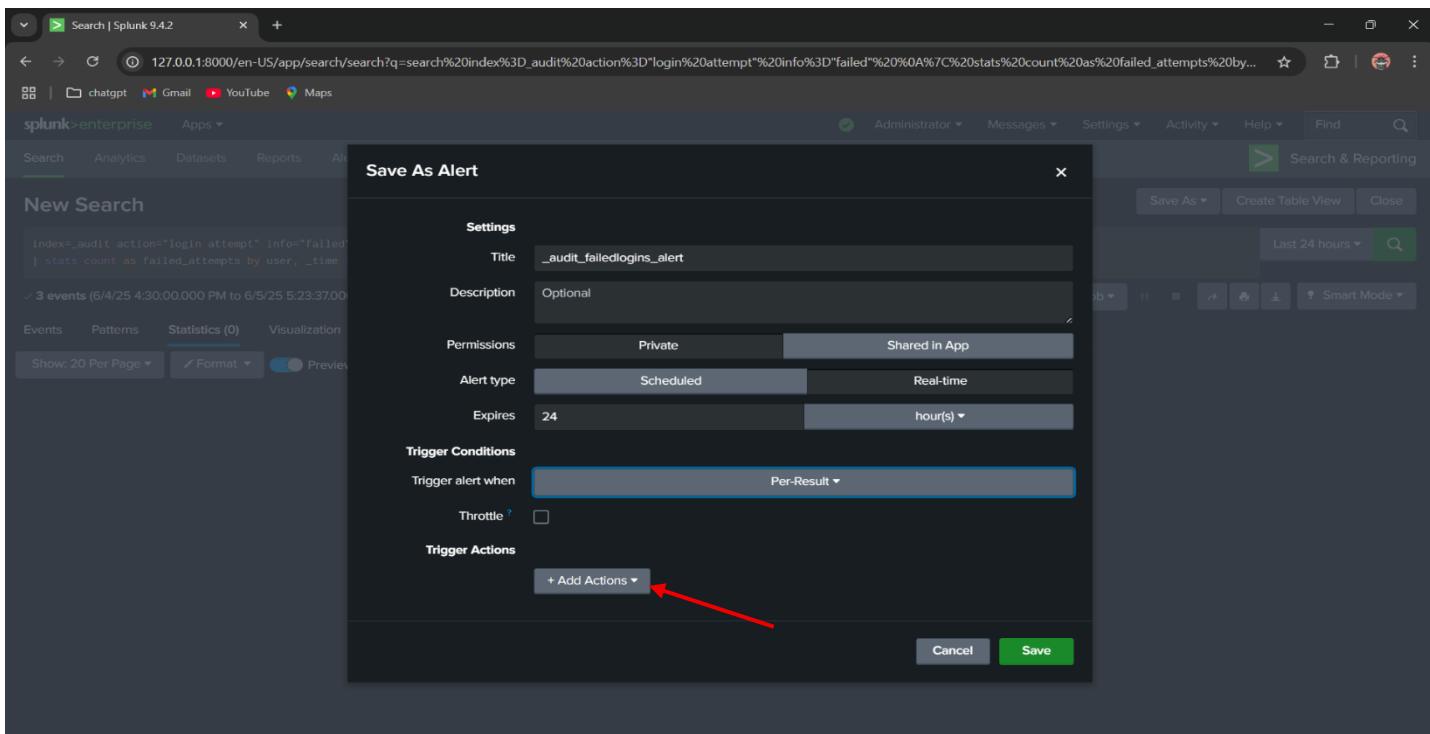
Step 11:

- In this step, I did not enable the throttle option. However, if you prefer, you can configure it based on your alerting needs.
- Throttle is used to prevent duplicate or repeated alerts from being triggered within a short time frame for the same condition. When enabled, it suppresses additional alerts for a specific time period (e.g., 1 hour) even if the condition is still being met.
- This is especially useful to avoid alert fatigue when the same issue continuously triggers the alert. For example, if a user repeatedly fails to log in, throttling ensures you receive just one alert instead of dozens in quick succession.



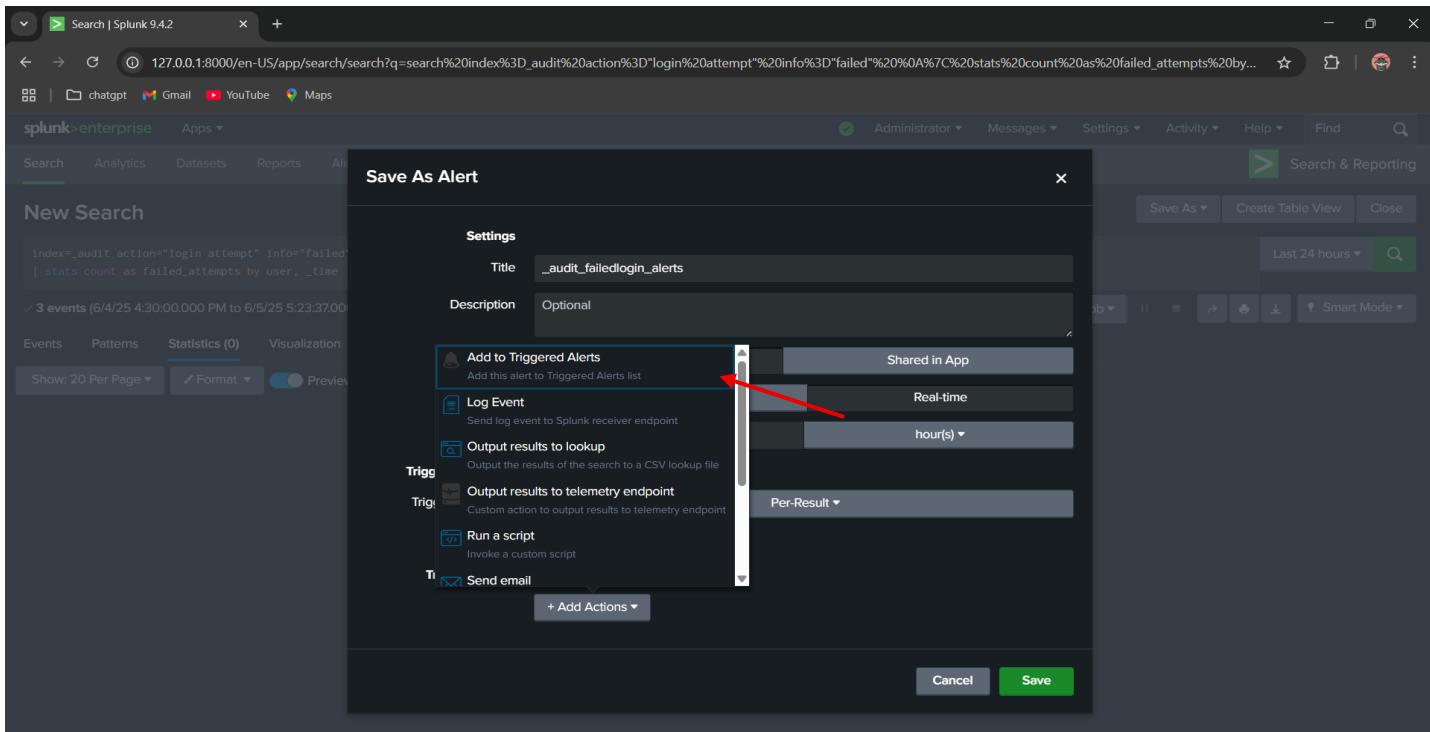
Step 12:

Now it's time to set how the alert should notify you when triggered. Under the Trigger Actions section, click on the "Add Actions" button. A dropdown menu will appear with several available options.



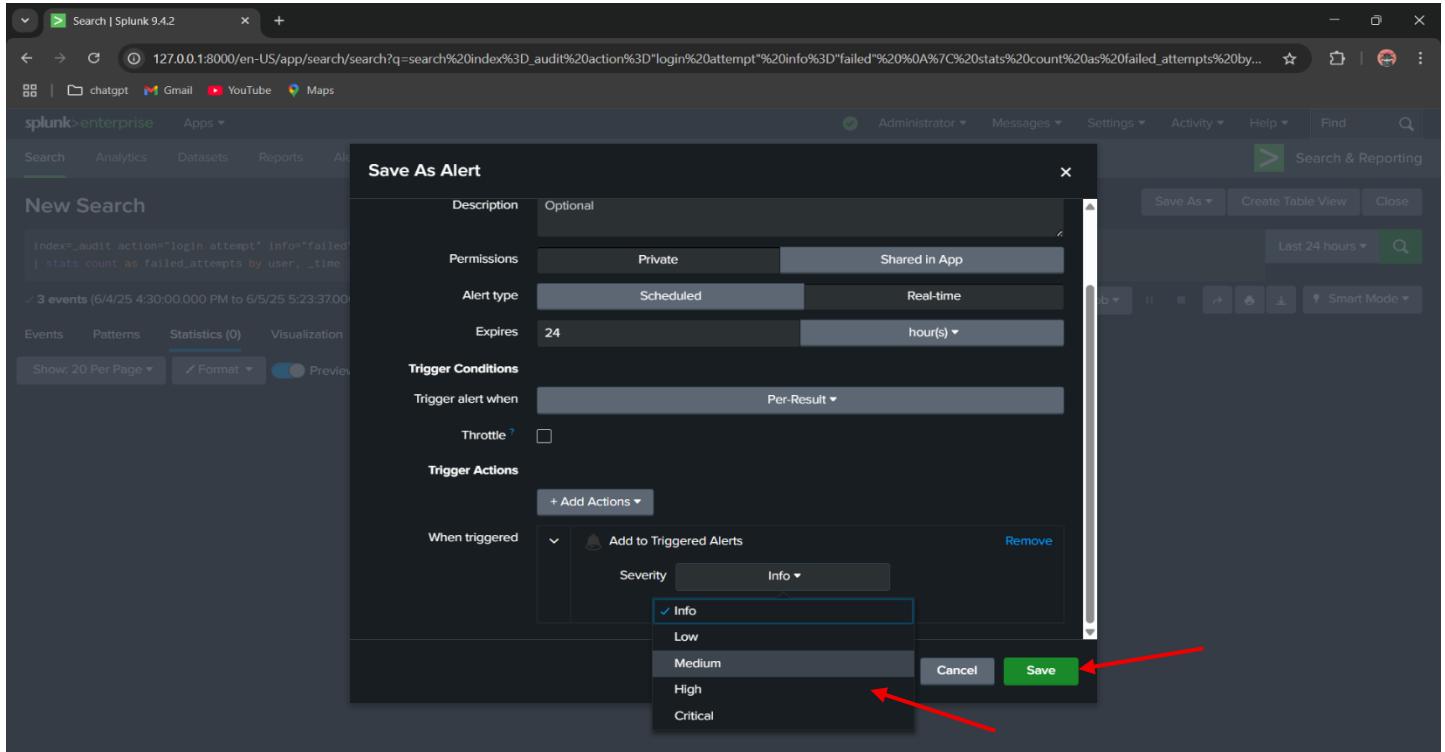
Step 13:

Now simply click on "Add to Trigger Alerts". This confirms and attaches the action to your alert configuration. Once added, the action becomes active and will execute whenever the alert condition is met.



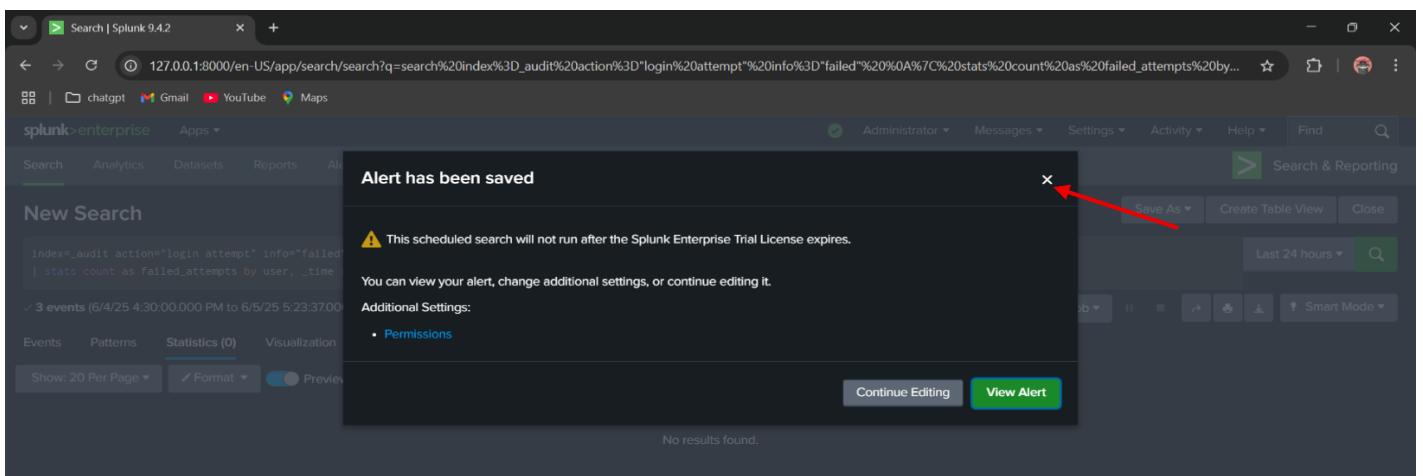
Step 14:

After clicking on Add to configure alert triggers, set the severity level according to your preference. In this case, I selected High. Finally, click Save to apply the settings.



Step 15:

After saving, a pop-up window will appear confirming that the alert has been successfully created. You can either close this window and access the alert later from the dashboard, or click on View in the pop-up to open the alert immediately.



Step 16:

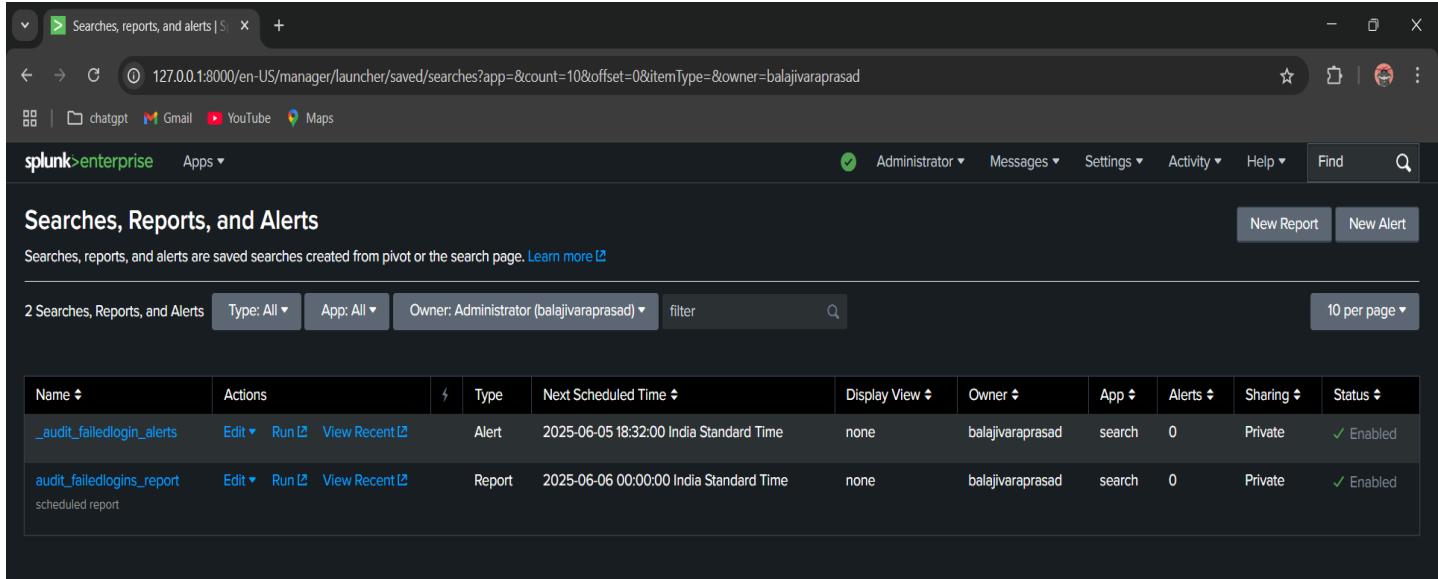
Navigate to the Splunk homepage by clicking on the Splunk icon. Then, go to Settings and under the Knowledge section, click on Searches, Reports, and Alerts.

The screenshot shows the Splunk homepage with a dark theme. At the top right, there is a navigation bar with links for 'Administrator', 'Messages', 'Settings' (which has a red arrow pointing to it), 'Activity', 'Help', and 'Find'. Below the navigation bar, the main content area displays 'Hello, Administrator' and various sections like 'Bookmarks', 'Dashboard', 'Search history', and 'Splunk recommended (13)'. A red arrow points from the 'Settings' link in the top navigation to the 'Knowledge' section in the sidebar.

The screenshot shows the Splunk homepage with the 'Knowledge' sidebar open. The sidebar is divided into several sections: 'Add Data', 'Monitoring Console', 'KNOWLEDGE' (with 'Searches, reports, and alerts' highlighted in blue and a red arrow pointing to it), 'DATA', 'DISTRIBUTED ENVIRONMENT', 'SYSTEM', and 'USERS AND AUTHENTICATION'. The 'Searches, reports, and alerts' section includes options like 'Data models', 'Event types', 'Tags', 'Fields', 'Lookups', 'User interface', 'Alert actions', 'Advanced search', and 'All configurations'. The rest of the page shows the same layout as the previous screenshot, including the 'Hello, Administrator' greeting and 'Splunk recommended' section.

Step 17:

You can now view the alert that was just created. Additionally, the scheduled report created earlier will also be visible in the list.



Screenshots, reports, and alerts | 127.0.0.1:8000/en-US/manager/launcher/saved/searches?app=&count=10&offset=0&itemType=&owner=balajivaraprasad

Administrator Messages Settings Activity Help Find

Searches, Reports, and Alerts

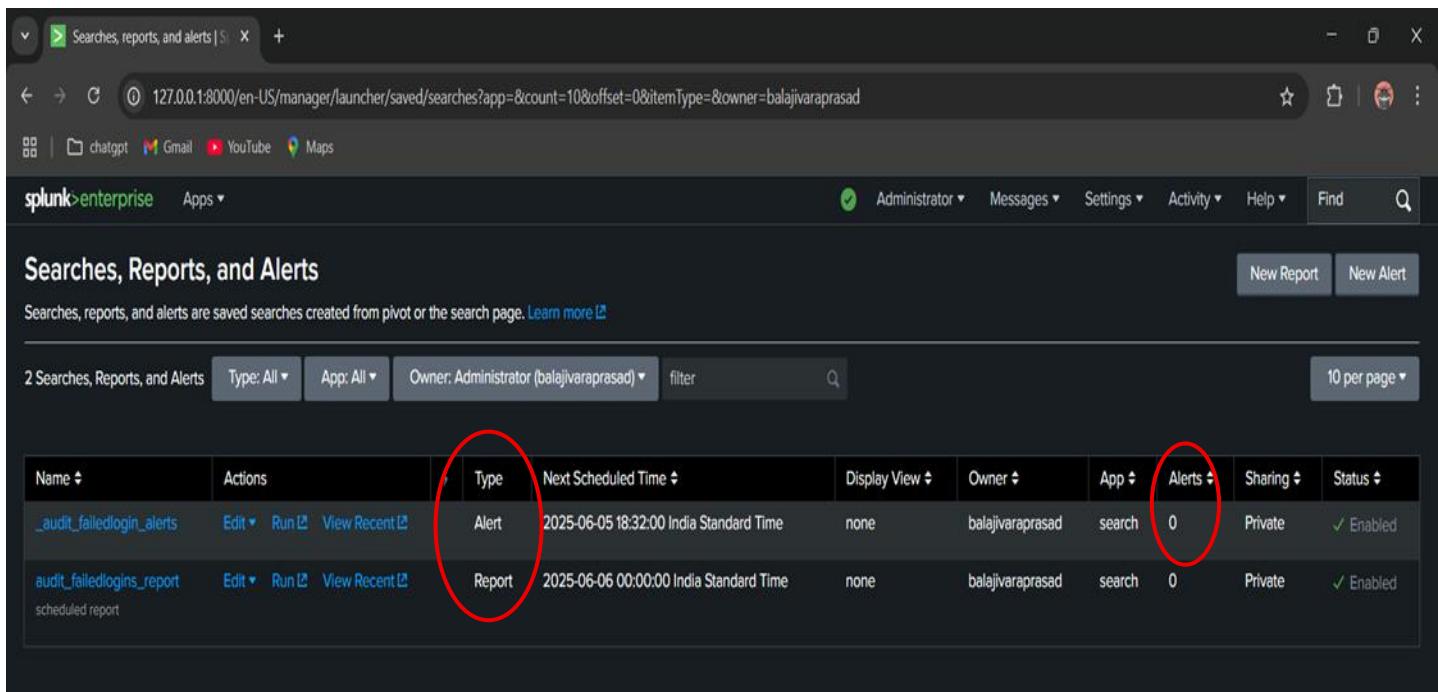
Searches, reports, and alerts are saved searches created from pivot or the search page. [Learn more](#)

New Report New Alert

Name	Actions	Type	Next Scheduled Time	Display View	Owner	App	Alerts	Sharing	Status
_audit_failedlogin_alerts	Edit Run View Recent	Alert	2025-06-05 18:32:00 India Standard Time	none	balajivaraprasad	search	0	Private	Enabled
audit_failedlogins_report	Edit Run View Recent	Report	2025-06-06 00:00:00 India Standard Time	none	balajivaraprasad	search	0	Private	Enabled

Step 18:

You can view the type of each data and the number of times the alert has been triggered. Currently, the alert count is zero because no failed login attempts have occurred since the alert was created.



Screenshots, reports, and alerts | 127.0.0.1:8000/en-US/manager/launcher/saved/searches?app=&count=10&offset=0&itemType=&owner=balajivaraprasad

Administrator Messages Settings Activity Help Find

Searches, Reports, and Alerts

Searches, reports, and alerts are saved searches created from pivot or the search page. [Learn more](#)

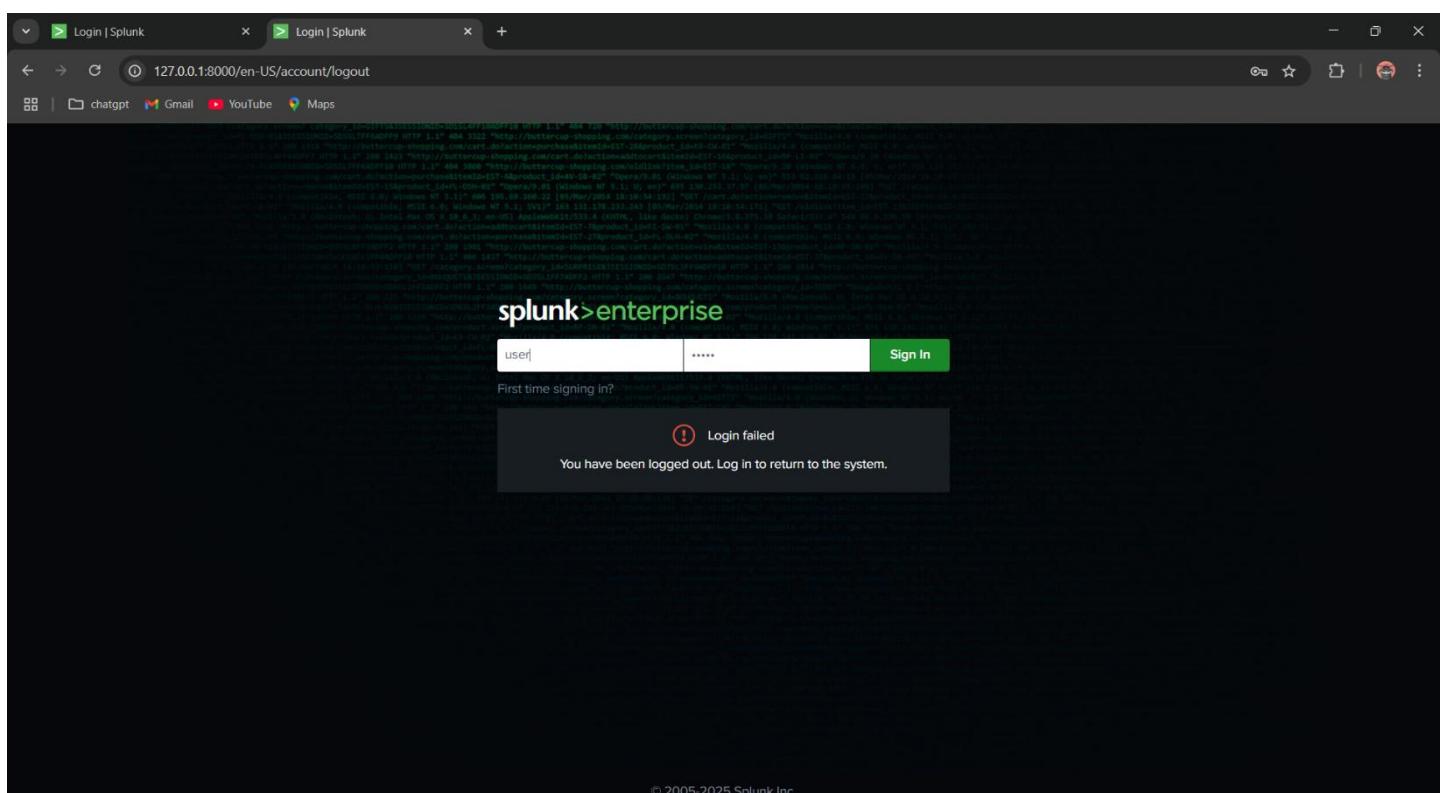
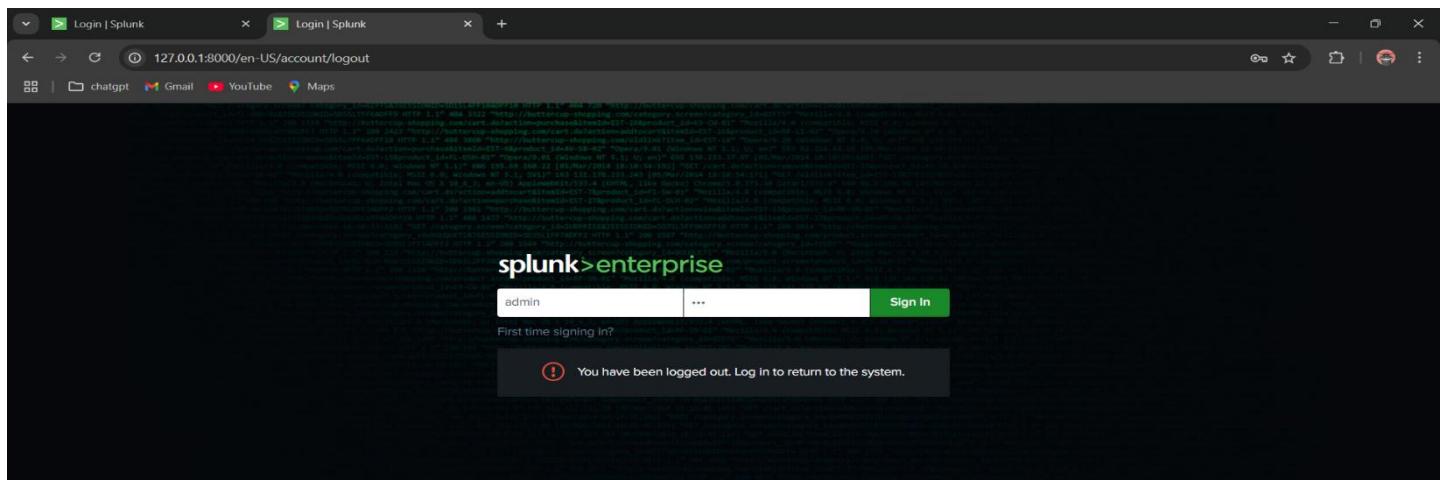
New Report New Alert

Name	Actions	Type	Next Scheduled Time	Display View	Owner	App	Alerts	Sharing	Status
_audit_failedlogin_alerts	Edit Run View Recent	Alert	2025-06-05 18:32:00 India Standard Time	none	balajivaraprasad	search	0	Private	Enabled
audit_failedlogins_report	Edit Run View Recent	Report	2025-06-06 00:00:00 India Standard Time	none	balajivaraprasad	search	0	Private	Enabled

Let's simulate failed login attempts in Splunk to trigger the alert and verify that it functions as expected.

Step 19:

Open Splunk in a new browser tab and navigate to the login page. Enter a username of your choice and intentionally provide an incorrect password. Click the login button repeatedly, around 3 to 5 times or more, to simulate failed login attempts. This action generates failed login events that exceed the threshold value specified in your alert query.



Step 20:

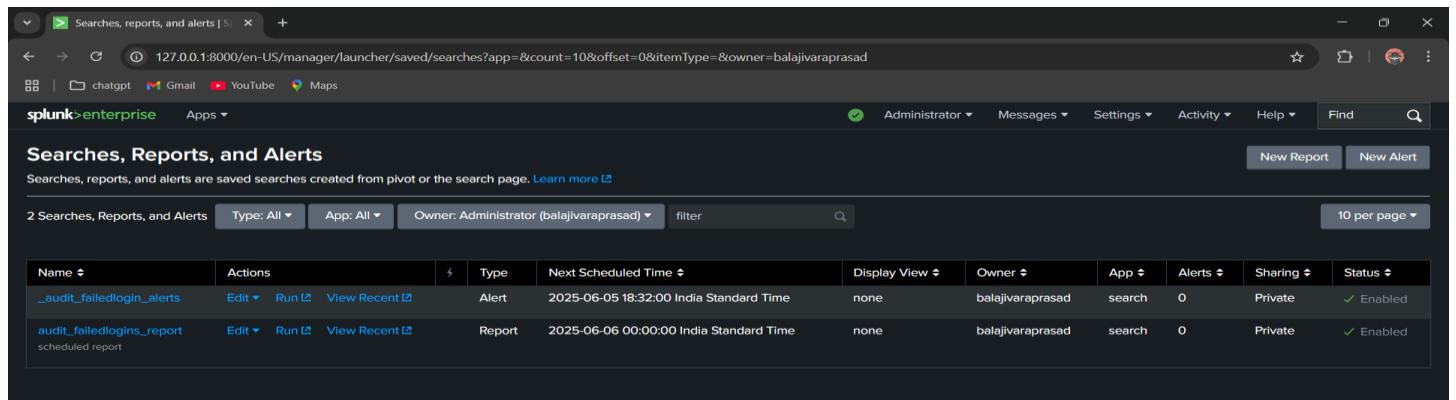
After simulating the failed login attempts, log in to Splunk using your original credentials to access the platform again.

Step 21:

Now, repeat the steps from Step 16 to Step 18 to navigate back to the Searches, Reports, and Alerts section. There, you can verify that the alert has been triggered as a result of the simulated failed login attempts.

Step 22:

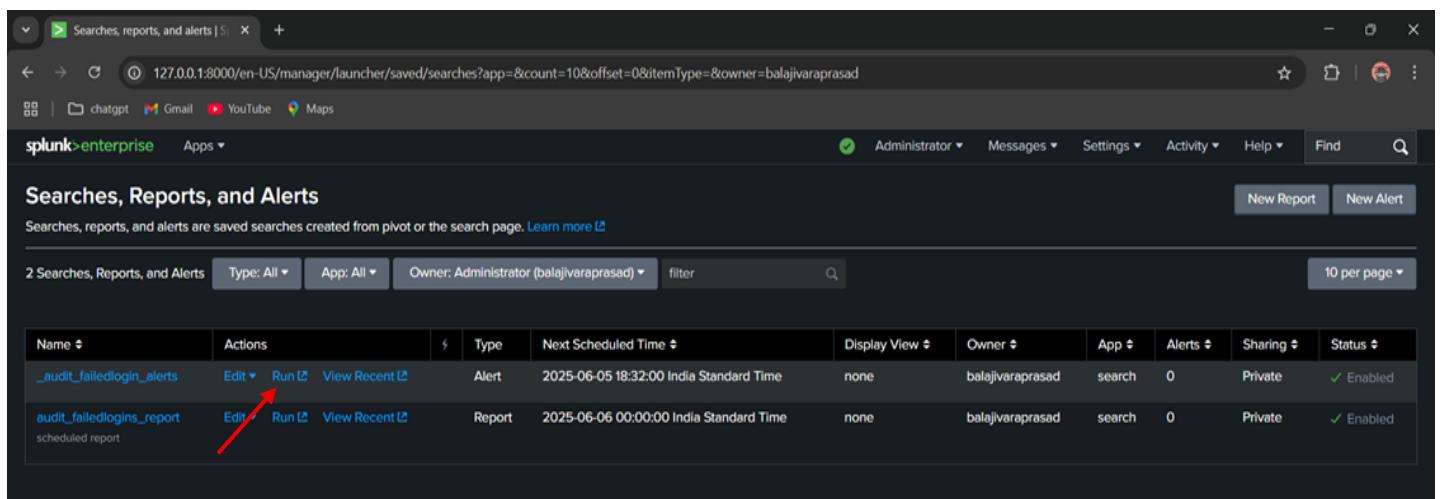
Now, we can still see there are zero alerts.



Name	Actions	Type	Next Scheduled Time	Display View	Owner	App	Alerts	Sharing	Status
_audit_failedlogin_alerts	Edit Run View Recent	Alert	2025-06-05 18:32:00 India Standard Time	none	balajivaraprasad	search	0	Private	Enabled
audit_failedlogins_report	Edit Run View Recent	Report	2025-06-06 00:00:00 India Standard Time	none	balajivaraprasad	search	0	Private	Enabled

Step 23:

Now click on the run option to update the alerts.



Name	Actions	Type	Next Scheduled Time	Display View	Owner	App	Alerts	Sharing	Status
_audit_failedlogin_alerts	Edit Run View Recent	Alert	2025-06-05 18:32:00 India Standard Time	none	balajivaraprasad	search	0	Private	Enabled
audit_failedlogins_report	Edit Run View Recent	Report	2025-06-06 00:00:00 India Standard Time	none	balajivaraprasad	search	0	Private	Enabled

Step 24:

You will now be redirected to the Search page. Set the time filter to Last 24 hours, then click on the Search button to execute the query.

The screenshot shows the Splunk 9.4.2 search interface. The top navigation bar includes tabs for 'Search, reports, and alerts' and 'Search | Splunk 9.4.2'. The address bar displays the URL: `127.0.0.1:8000/en-US/app/search/search?s=%2FservicesNS%2Fbalajivaraprasad%2Fsearch%2Fsaved%2Fsearches%2F_audit_failedlogin_alert&sid=rt_1749136322.233&display.page=search.mod...`. The main header features the 'splunk>enterprise' logo, user status 'Administrator', and navigation links for 'Messages', 'Settings', 'Activity', 'Help', 'Find', and a search icon.

The search bar contains the query: `index=_audit action="login attempt" info="failed" | stats count as failed_attempts by user, _time | where failed_attempts > 3`. Below the search bar, it says 'All time (real-time)'. The results section indicates '0 of 0 events matched' and 'No Event Sampling'. The interface includes standard search controls like 'Save', 'Save As', 'View', 'Create Table View', and 'Close'. Below the search bar, there are tabs for 'Events', 'Patterns', 'Statistics (0)', and 'Visualization'. The 'Statistics (0)' tab is currently selected. At the bottom left, there are buttons for 'Show: 20 Per Page' and 'Format'. A message at the bottom center states 'No results in current time range.'

The screenshot shows the Splunk 9.4.2 interface with the following details:

- Search Bar:** The URL is `127.0.0.1:8000/en-US/app/search/search?s=%2FservicesNS%2Fbalajivaraprasad%2Fsearch%2Fsearch%2Fsaved%2Fsearches%2F_audit_failedlogin_alert&display.page.search.mode=smart&dispatch.sample...`. The search term is `_index=_audit action="login attempt" info="failed" | stats count as Failed_attempts by user, _time | where failed_attempts > 3`.
- Header:** Shows the user is an **Administrator**. Navigation links include Search, Analytics, Datasets, Reports, Alerts, Dashboards, and a **Search & Reporting** button.
- Job Control:** Buttons for Save, Save As, View, Create Table View, and Close.
- Time Range:** Set to "Last 24 hours". A red arrow points to this dropdown.
- Event Summary:** 25 events from 6/4/25 8:30:00.000 PM to 6/5/25 8:42:22.000 PM. A red arrow points to the event count.
- Filter:** No Event Sampling.
- Statistics:** Statistics (0).
- Visualizations:** Visualization.
- Page Options:** Show: 20 Per Page, Format, Preview: On.
- Results:** No results found.

Step 25:

- o Even though the time filter is set to Last 24 hours, you may see a message indicating “No results found.”
 - o Let’s troubleshoot this issue. The most likely causes are:
 1. An error in the SPL (Search Processing Language) query.
 - 2.Improper filtering or thresholds that are excluding the relevant data.

- o However, we can see that the number of events has increased to **25**, which confirms that the failed login attempts we simulated have been successfully captured in the audit logs.
- o To verify this, let's simplify the query by temporarily removing the threshold condition from the main SPL query and rerun the search to check if the events appear.

The screenshot shows the Splunk 9.4.2 search interface. The search bar contains the query: `index=_audit action="login attempt" info="failed" | stats count as failed_attempts by user, _time`. The results panel displays 25 events for the user "admin" between June 4, 2025, and June 5, 2025. Each event shows a timestamp and a value of 1 for "failed_attempts". Red arrows point to the "Last 24 hours" time range selector and the green search button.

user	_time	failed_attempts
admin	2025-06-05 19:35:00.572	1
admin	2025-06-05 19:35:01.422	1
admin	2025-06-05 19:35:01.969	1
admin	2025-06-05 19:35:02.429	1
admin	2025-06-05 19:35:03.099	1
admin	2025-06-05 19:35:03.303	1
admin	2025-06-05 19:35:03.467	1
admin	2025-06-05 19:35:03.633	1
admin	2025-06-05 19:35:03.805	1
admin	2025-06-05 19:35:03.975	1
admin	2025-06-05 19:35:04.863	1
admin	2025-06-05 19:35:05.041	1
admin	2025-06-05 19:35:05.185	1

Step 26:

- o After running the updated query, the data appears as expected. This confirms that the issue was due to the use of `_time` (timestamp) in the SPL query.
- o When `_time` is included in the `stats` or `by` clause, each failed login attempt is recorded as a separate event, even for the same user, because `_time` differentiates entries down to the millisecond.
- o As a result, the data was fragmented across multiple timestamps, preventing the alert from being triggered, even though there were multiple failed login attempts by the same user.

The screenshot shows the Splunk Search interface with a search bar containing the query: `index=_audit action="login attempt" info="failed" | stats count as failed_attempts by user, _time`. The results section displays 25 events from June 4, 2025, between 7:30:00.000 PM and 7:41:14.000 PM. The results are grouped by user ('admin') and ordered by time (_time). A red oval highlights the first seven rows, which represent multiple failed login attempts for the user 'admin' on the same timestamp (2025-06-05 19:35:00.572).

user	_time	failed_attempts
admin	2025-06-05 19:35:00.572	1
admin	2025-06-05 19:35:01.422	1
admin	2025-06-05 19:35:01.969	1
admin	2025-06-05 19:35:02.429	1
admin	2025-06-05 19:35:03.099	1
admin	2025-06-05 19:35:03.303	1
admin	2025-06-05 19:35:03.467	1
admin	2025-06-05 19:35:03.633	1
admin	2025-06-05 19:35:03.805	1
admin	2025-06-05 19:35:03.975	1
admin	2025-06-05 19:35:04.863	1
admin	2025-06-05 19:35:05.91	1
admin	2025-06-05 19:35:05.185	1

Step 27:

Now, let's correct the SPL query and create a new version that properly groups the failed login attempts. This updated query will ensure that the alert is triggered when the defined threshold is reached.

Step 28:

- o Use the following updated SPL query to detect users with more than three failed login attempts:
- o `index=_audit action="login attempt" info="failed" | stats count as failed_attempts by user | where failed_attempts > 3`
- o Enter this query into the search bar and run it.
- o The key difference in this version is that we removed `_time` from the `stats` clause, which allows the query to group all failed attempts by user, rather than splitting them by timestamp. This ensures the alert will be triggered once the threshold is exceeded.

Step 29:

You can now clearly see that the data has been generated as expected. The query successfully groups failed login attempts by user, making it suitable for setting up an alert based on the defined threshold.

The screenshot shows the Splunk 9.4.2 search interface. The search bar contains the command: `index=_audit action="login attempt" info="failed" | stats count as failed_attempts by user | where failed_attempts > 3`. Below the search bar, it says "25 events (6/4/25 8:30:00.000 PM to 6/5/25 9:09:35.000 PM) No Event Sampling". The "Statistics (3)" tab is selected. The results table has two columns: "user" and "failed_attempts". The data is as follows:

user	failed_attempts
admin	13
balaji varaprasad	5
user	4

There are red arrows pointing to the "Smart Mode" button in the top right corner and the "Format" dropdown menu.

Step 30:

- o Now, delete the previously created alert to avoid confusion.
To delete an alert:

1. Go to Settings and navigate to the Alerts section.
2. Locate the alert you created earlier.
3. Click on the Edit option next to it.
4. From the dropdown, select Delete to remove the alert.

Step 31:

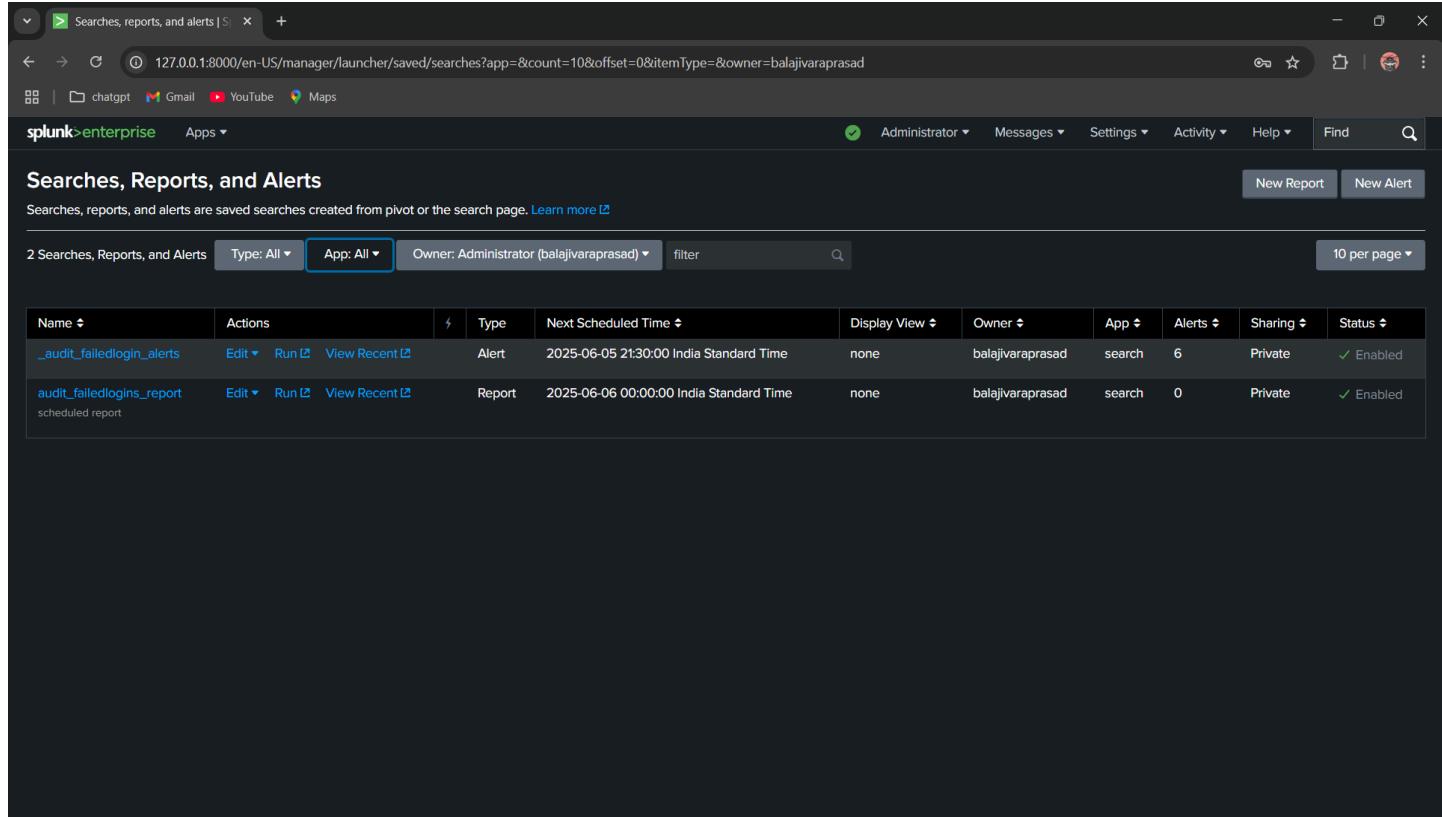
To create the alert, please follow the instructions outlined in Steps 5 through 18.

Step 32:

Now, simulate the failed login attempts again by following the procedure described in Steps 19 through 22. Try to use different usernames this time.

Step 33:

After navigating to Search, Reports, and Alerts in the Settings, you can view the alert created using the new SPL query.

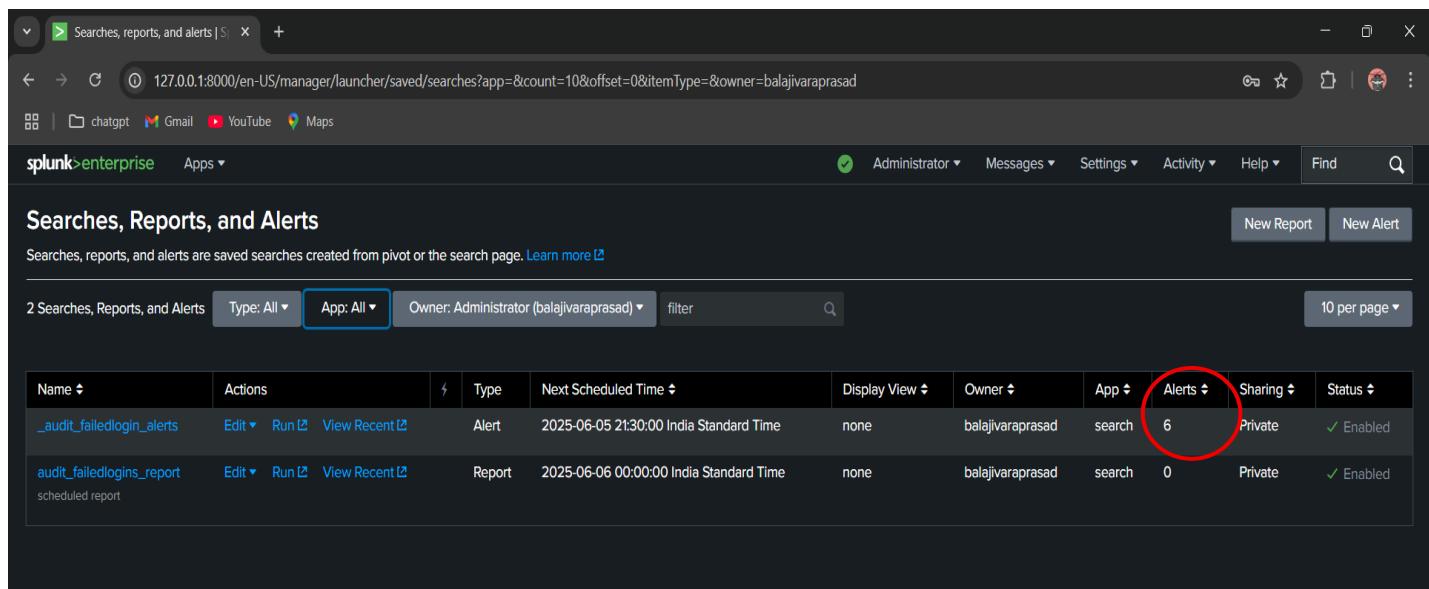


Screenshots of the Splunk interface showing the 'Searches, Reports, and Alerts' page. The page lists two items:

Name	Type	Next Scheduled Time	Display View	Owner	App	Alerts	Sharing	Status
_audit_failedlogin_alerts	Alert	2025-06-05 21:30:00 India Standard Time	none	balajivaraprasad	search	6	Private	Enabled
audit_failedlogins_report	Report	2025-06-06 00:00:00 India Standard Time	none	balajivaraprasad	search	0	Private	Enabled

Step 34:

You can now see that six alerts have been triggered in the alert you created.



Screenshot of the Splunk interface showing the 'Searches, Reports, and Alerts' page. The 'Alerts' column for the first item is circled in red.

Name	Type	Next Scheduled Time	Display View	Owner	App	Alerts	Sharing	Status
_audit_failedlogin_alerts	Alert	2025-06-05 21:30:00 India Standard Time	none	balajivaraprasad	search	6	Private	Enabled
audit_failedlogins_report	Report	2025-06-06 00:00:00 India Standard Time	none	balajivaraprasad	search	0	Private	Enabled

Step 35:

Click on the Run option, set the time filter to the last 24 hours, and then click Search. You will now be able to view the data from the simulated failed login attempts.

Screenshots illustrating the steps to run a search for failed login attempts:

The first screenshot shows the 'Searches, Reports, and Alerts' page. It lists two items: '_audit_failedlogin_alerts' (Alert) and 'audit_failedlogins_report' (Report). The '_audit_failedlogin_alerts' item has a 'Run' button highlighted with a red arrow.

The second screenshot shows the search results for the '_audit_failedlogin_alerts' search. The search bar contains the query: `index=_audit action="login attempt" info="failed" | stats count as failed_attempts by user | where failed_attempts > 3`. The results table is empty, showing '0 of 0 events matched'. A red arrow points to the 'Smart Mode' button in the top right corner of the search results panel.

The third screenshot shows the search results for the '_audit_failedlogin_alerts' search with a time filter set to 'Last 24 hours'. The search bar contains the same query: `index=_audit action="login attempt" info="failed" | stats count as failed_attempts by user | where failed_attempts > 3`. The results table shows 55 events, with the first few entries being: admin (13), admin1 (15), admin2 (10), admin3 (5), balaji varaprasad (5), and user (4).

We successfully created a real-time alert in Splunk to monitor failed login attempts exceeding a defined threshold. By setting a condition to trigger the alert when any user experiences more than three failed login attempts, the alert helps in promptly identifying potential unauthorized access attempts. The configuration includes specifying a meaningful title, setting the alert type to real-time, and defining the trigger condition as “per result.” Though no action was added in this instance, the alert is now active and will execute whenever the condition is met, allowing for quick visibility into critical security events.

Alert Actions and Escalation:

Once triggered, Splunk can notify teams or systems via alert actions. For example:

- o **Email Notification:** Splunk can send an email with alert details to specified recipients. (You must configure SMTP in Splunk’s Settings → Server Settings → Email.) Emails can include the search results or a summary.
- o **Webhooks/Integrations:** Use the webhook or custom alert actions (via Splunkbase apps) to invoke external services. For instance, you might call a ServiceNow API to open an incident ticket, or post to a Slack/PagerDuty webhook. Splunk also offers an official ServiceNow and PagerDuty add-on for automated ticket creation.
- o **Custom Scripts:** You can run a script action to perform any automated response (e.g. restart a service, disable an account) when the alert fires.

To handle escalation if the alert is not addressed, consider:

- o **Repeated Notifications:** Configure the alert to repeat (e.g. daily) until the condition clears, or schedule a follow-up search. This ensures ongoing issues keep alerting.
- o **On-Call Systems:** Route alerts into an on-call/incident management tool (like Splunk On-Call/VictorOps or PagerDuty). These platforms support multi-level escalation policies so that if the first responder doesn’t acknowledge the incident, it automatically pages the next person.
- o **Ticketing Integration:** Auto-create a ticket in ITSM (ServiceNow, Jira, etc.) with the alert details. The ticket system can then enforce SLA-based escalation (reminders, re-assignments) if the issue remains open.

By combining Splunk's alert actions with external tools, you can ensure that a "failed login" alert not only notifies the team immediately (via email or chat) but also gets tracked in a ticketing system or on-call rotation. Splunk's alert framework makes it easy to plug in these mechanisms, for example, an email alert action will "send an email notification to specified recipients when an alert triggers", and similar actions exist for scripts or webhooks.

Conclusion:

Scheduling reports and alerts in Splunk is a fundamental practice that significantly enhances operational efficiency, security monitoring, and real-time data analysis. By automating the process of generating insights, such as tracking failed login attempts, organizations can ensure continuous visibility into critical events without manual intervention. This not only saves time but also ensures timely detection and response to anomalies or threats, such as unauthorized access attempts. The integration of alerts with trigger conditions and automated actions like email notifications further empowers teams to stay proactive and maintain high system reliability.

Moreover, the detailed step-by-step implementation covered in this guide demonstrates that setting up such monitoring mechanisms in Splunk is both practical and accessible, even for those without deep technical expertise. Splunk's intuitive interface, combined with its powerful Search Processing Language (SPL), enables users to create targeted reports and effective alerts tailored to their operational or security requirements. In today's data-driven world, leveraging such tools ensures that important insights are not only uncovered but also acted upon, paving the way for smarter, faster decision-making.

Reference:

[https://docs.splunk.com/Documentation/Splunk/9.4.2/Alert/Definescheduledalerts#:~:text/Create%20scheduled%20alerts](https://docs.splunk.com/Documentation/Splunk/9.4.2/Alert/Definescheduledalerts#:~:text>Create%20scheduled%20alerts)

<https://docs.splunk.com/Documentation/Splunk/9.4.2/Alert/Definescheduledalerts#:~:text=1,This%20setting%20controls%20the%20lifespan>

<https://docs.splunk.com/Documentation/Splunk/9.4.2/Alert/Definescheduledalerts#:~:text=8,happen%20when%20the%20alert%20triggers>

<https://docs.splunk.com/Documentation/Splunk/9.4.2/Security/Searchforauditevents#:~:text=1>

<https://docs.splunk.com/Documentation/Splunk/9.4.2/Alert>Emailnotification#:~:text=Send%20an%20email%20notification%20to,directly%20in%20a%20search%20command>

THANK YOU