Splunk® Light Search and Reporting Examples 6.6.3

Generated: 9/13/2017 2:24 pm

Table of Contents

About search and reporting examples....1
About Splunk Light Search and Reporting Examples and Scenarios....1

Search and reporting examples....2
Search for errors using Splunk Light....2
Calculate and chart statistics using Splunk Light....3
Compare week over week results using Splunk Light....4
Report on failed login attempts using Splunk Light....5

Alert examples....8
Notify when server load reaches a threshold using Splunk Light....8
Identify spikes in data and notify using Splunk Light....10

Creating an alert scenario....12
Create an alert to monitor CPU usage using Splunk Light....12
Enable the Splunk Add-On for Unix and Linux....12
Create your search....13
Save your search as an alert....14
View and edit your alert....15
View the Triggered Alerts list....15

Creating a dashboard scenario....16
Create and manage a dashboard using Splunk Light....16
Enable the Splunk Add-on for Unix and Linux....16
Create a dashboard....18
Add prebuilt panels to a dashboard....18
Add a dashboard panel from a search....19
Add tables to a dashboard....19
Add a single value panel to a dashboard....21
Edit dashboard panels....21
Organize and view a dashboard....23

About search and reporting examples

About Splunk Light Search and Reporting Examples and Scenarios

This manual includes examples and scenarios that teach you how to construct and run searches, save reports, configure alerts, and build dashboards.

Searches and Reports

Search your data and save reports. See examples:

- Search for errors using Splunk Light in *Search and Reporting Examples*.
- Calculate and chart statistics using Splunk Light in *Search and Reporting Examples*.
- Compare week over week results using Splunk Light in *Search and Reporting Examples*.
- Report on failed login attempts using Splunk Light in *Search and Reporting Examples*.

Alerts

Set up alert conditions for saved and scheduled reports, or for monitoring. See examples and scenario:

- Notify when server load reaches a threshold using Splunk Light in *Search and Reporting Examples*.
- Identify spikes in data and notify using Splunk Light in *Search and Reporting Examples*.
- Create an alert to monitor CPU usage using Splunk Light in *Search and Reporting Examples*.

Dashboards

Create dashboards and add saved reports to dashboard panels. See scenario:

- Create and manage a dashboard using Splunk Light in *Search and Reporting Examples*.

Search and reporting examples

Search for errors using Splunk Light

This topic includes examples of basic searches using keywords, phrases, booleans, fields, wildcards and comparison operators. These searches describe the events you want to retrieve from your Splunk Light indexes.

Task

Search for different types of errors or failures.

Use keywords and phrases

1. If you want to find events with "error", start by typing in the keyword.

```
error
```

2. To make the searches more efficient, use as many keywords as possible to describe the event. For example, to find specific errors described by a phrase, use the entire phrase.

```
"sshd error"

"login failed"

"failed password"

"access denied"
```

Use fields and wildcards

Fields are name and value pairs in your events. All events have the `source`, `host`, `sourcetype`, `_raw`, and `_time` fields. To search for specific field values, use the field name and field value.

You can use the asterisk wildcard with search keywords, field names, and field values to match patterns in events.

1. Search Apache web access logs for 404 status errors.

```
sourcetype=access_combined status=404
```

2. Find all client and server errors.

```
status=40* OR status=50*
```

This matches status values of 400, 401, 402, and so on, and 500, 501, 502, and so on.

Use boolean and comparison operators

Type	Operators	Description
Boolean	`AND, OR, NOT`	The operators must be written in uppercase. The AND operator is implied between search terms. You can group terms together using parentheses. When you have parentheses, the boolean expressions are evaluated inside the parentheses first. When using boolean expressions, searching for inclusion yields faster results than searching for exclusions.
Comparison	`< > <= >= != = ==`	The operators can be used to match field values for numbers and strings.

1. Find all client or server errors with a delay greater than 10 seconds.

```
status >= 40* delay > 10
```

2. Search for invalid user login attempts.

```
"invalid user" OR "failed password" OR "not allowed"
```

3. Search for only 404 or 503 status errors.

```
status=404 OR status=503
```

Calculate and chart statistics using Splunk Light

Task

Calculate metrics for different hosts.

Searches

These searches start with a search for "error", but you can replace this search with other search terms.

1. Count the number of errors seen on each host.

```
error | stats count by host
```

2. Search for outliers. Here, outliers are hosts with a count of errors that is two standard deviations from the mean.

```
error | stats count by host | eventstats avg(count) as avg_count
stdev(count) as std_count | where count>(2*avg_count + std_count)
```

Compare week over week results using Splunk Light

Task

Determine how this week's average download compare with last week's results.

Solutions

1. Find events in the total time period.

In this case, the time period is two weeks. Use time modifiers in your search.

```
earliest=-2w@w latest=@w
```

2. Differentiate events between the two weeks.

Use the eval command to create new fields, "this week" and "last week".

```
earliest=-2w@w latest=@w | eval marker=if
(_time<relative_time(now(),"-w@w"), "last week","this week")
```

3. To graph the two weeks on the same time range, adjust last week's events to look like they occurred this week.

```
earliest=-2w@w latest=@w | eval marker=if
(_time<relative_time(now(),"-w@w"), "last week","this week") | eval
```

```
_time=if(marker=="last week", _time + 7*24*60*60, _time)
```

4. Chart the average download for each week.

```
earliest=-2w@w latest=@w | eval marker=if
(_time<relative_time(now(),"-w@w"), "last week","this week") | eval
_time=if(marker=="last week", _time + 7*24*60*60, _time) | timechart
avg(bytes) by marker
```

This produces a timechart with two series, "last week" and "this week".

Report on failed login attempts using Splunk Light

This example uses LDAP data with source type `winauthentication_security`. The search monitors users of a fictitious online company, called Buttercup Games, who have multiple login failures over the past 24 hours.

The data contains Windows Event Codes, such as:

- 540 Successful Network Logon
- 4624 Successful Network Logon
- 4625 Failure
- 4634 Successful Network Logoff

To search for failures, you can specify the Event Code:

```
sourcetype=winauthentication_security EventCode=4625
```

You can search for the top failed logins based on the User:

```
sourcetype=winauthentication_security EventCode=4625 | top User
```

By default, the `top` command will return 10 results. You can change this limit. Or, you can use the `stats` command to see all users with failed login attempts.

```
sourcetype=winauthentication_security EventCode=4625 | stats count by
User
```

In this search, `stats` counts the number of failed login attempts by User. Then, you can use the `where` command to show only the Users who attempted and failed more than once.

```
sourcetype=winauthentication_security EventCode=4625 | stats count by
User | where count > 1
```

If there are users you want to exclude, you can filter them out in the original search. For example, you may not want to include User=Administrator in your report.

```
sourcetype=winauthentication_security EventCode=4625
User!=Administrator | stats count by User | where count > 1
```

Alert examples

Notify when server load reaches a threshold using Splunk Light

Task

Configure Splunk Light to notify you when a server's load reaches a predefined threshold, such as 80%.

Part 1: Run the search

The following search retrieves events with load averages above 80% and calculates the maximum value for each host.

```
sourcetype=top load_avg>80 | stats max(load_avg) by host
```

Part 2: Configure an alert

Save the search as an alert and configure the alert condition and alert actions as follows.

- Alert condition: Alert if the search returns at least one result.
- Alert actions: Email and set subject to "Server load above 80%."
- Suppress: 1 hour.

1. After you run the search, click **Save as** and select **Alert**.

2. In the **Save As Alert** dialog box, enter a **Title** and (Optional) **Description**.

3. Next to **Alert Type**, select **Real Time**.

4. Next to **Trigger condition**, select **Per-Result**.

5. Click **Next**.

6. Under **Enable Actions**, select **Send Email**.

6a. Next to **To**, enter the email recipients.

6b. (Optional) Change the **Priority** level for this alert.

6c. Next to **Subject**, enter "Server load above 80%."

6d. (Optional) Enter a **Message** to include with the email.

6e. Next to **Include**, select **Inline** and choose **Raw** to include the event that triggered the alert in the email.

7. Under **Action Options**, select **Throttle**.

7a. Next to **Suppress triggering for**, enter 1 and select **hour(s)**.

8. Click **Save**.

Identify spikes in data and notify using Splunk Light

You want to identify spikes in your data. Spikes can show you where you have peaks (or troughs) the indicate that some metric is rising or falling sharply. Traffic spikes, sales spikes, spikes in the number of returns, spikes in database load. Whatever type of spike you are interested in, you want to watch for it, set up alerts to notify you, and then perhaps take some action to address those spikes.

Search for spikes in your data

Use a moving trendline to help you see the spikes. Run a search followed by the trendline command using a field you want to create a trendline for.

For example, on web access data, you could chart an average of the `bytes` field.

```
sourcetype=access* | timechart avg(bytes) as avg_bytes
```

To add another line or bar series to the chart for the simple moving average (sma) of the last 5 values of `bytes`, use this command:

```
... | trendline sma5(avg_bytes) as moving_avg_bytes
```

If you want to clearly identify spikes, you might add an additional series for spikes. This is when the current value is more than twice the moving average.

```
... | eval spike=if(avg_bytes > 2 * moving_avg_bytes, 10000, 0)
```

The 10000 here is arbitrary and you should choose a value relevant to your data that makes the spike noticeable. Changing the formatting of the y-axis to Log scale also helps.

Putting this all together, the search is:

```
sourcetype=access* | timechart avg(bytes) as avg_bytes | trendline
sma5(avg_bytes) as moving_avg_bytes | eval spike=if(avg_bytes > 2 *
moving_avg_bytes, 10000, 0)
```

This search uses a simple moving average for the last 5 results (sma5). Consider a different number of values, for example sma20.

The trendline command also supports the exponential moving average (ema) and the weighted moving average (wma).

Alternatively, you can bypass the charting altogether and replace the `eval` command with the `where` command to filter your results.

```
... | where avg_bytes > 2 * moving_avg_bytes
```

And by looking at the table view or as an alert, you will only see the times when the `avg_bytes` spiked.

Creating an alert scenario

Create an alert to monitor CPU usage using Splunk Light

Alerts actively monitor your data and notify you when an alert is triggered. You can schedule alerts, or run an alert in real-time. Both provide insight into your data.

In this scenario you create an alert from a search, configure that alert, and add alert actions.

What you need for this scenario

To complete this scenario, first ensure that you have Splunk Light installed and running. You must be using an on-premise version of Splunk Light. Additionally, ensure that you can enable the Splunk Add-On for Unix and Linux.

Scenario overview

Complete each of these steps to reach the goal of creating a useful alert

1. Enable the Splunk Add-On for Unix and Linux.
2. Create your search.
3. Save your search as an alert.
4. View and edit your alert.
5. View Triggered Alerts list.

Enable the Splunk Add-On for Unix and Linux

By enabling the Splunk Add-On for Unix and Linux, you can analyze data from your deployment and set up alerts to notify you when a change occurs.

1. Log in to Splunk Light.
2. Navigate to **Data > Add-Ons**.
3. Click **Enable** on the Splunk Add-On for Unix and Linux.
4. Click **Set Up**.
5. Click **Enable** on all inputs.
6. Click **Save** at the bottom of the page.

7. Restart your Splunk Light instance for the changes to take effect.

By enabling the inputs, you set bash scripts to run at specific intervals and collect event data in the os index. Splunk Light can now access and analyze your data.

Confirm the Splunk Add-On for Unix and Linux is enabled

Ensure that your inputs are enabled.

1. Log in to Splunk Light.
2. Select to **Data > Add-Ons**.
3. Click **Objects** in the **Splunk Add-On for Unix and Linux**.
4. Click **Indexes**. The index **os** should have an event count greater than zero.

Create your search

To create an alert, you need to create your search. In this search, look for CPU usage over 75%.

1. Click **Search** on the Splunk Light bar.
2. Type the following search

   ```
   index=os sourcetype=cpu host=*
   ```

   to display events of source type CPU from any host.
3. Click **All fields**.
4. Select **PercentIleTime**. Exit the dialog box.
5. Type the following search

   ```
   index=os sourcetype=cpu host=* | multikv fields PercentIdleTime |
   eval Percent_CPU_Usage = 100 - PercentIdleTime
   ```

   to create the field **Percent_CPU_Usage**.
6. Click **All fields**
7. Select **Percent_CPU_Usage**. Exit the dialog box.
8. Type the following search.

   ```
   index=os sourcetype=cpu host=* | multikv fields PercentIdleTime |
   eval Percent_CPU_Usage = 100 - pctIdle | where Percent_CPU_Usage
   > 75
   ```

Your search now shows all events where CPU usage is greater than 75%.

Next, save your search and create an alert to actively monitor CPU usage.

Save your search as an alert

Now that your search displays all events where the percent of CPU Usage is greater than 75%, save your search as an alert. Alerts monitor your data and alert you when the specified trigger conditions are met.

After completing the previous step:

1. Click **Save As** from the search page.
2. Click **Alert**.
3. Name your alert CPU Usage > 75%.
4. Under **Alert type** select **Real-time**.

Set trigger conditions

Trigger conditions lets you specify what triggers your alert. You can trigger an alert on a per-result basis, by the number of results, by the number of hosts, by the number of sources, or even with a custom trigger condition.

1. Select **Number of Results**.
2. Select the **is greater than** menu, and select **is equal to**. Enter **1**.
3. Select **For each result**.

Every time there is an event where CPU usage is greater than 75%, you receive an alert.

Set trigger actions

Trigger Actions let you specify how your alert notifies you. You can add your alert to the triggered alerts list, send a log event to a splunk receiver endpoint, run a script, utilize a webhook, or send yourself an email. You can add multiple trigger actions to an alert.

1. To add the email alert action, go to Trigger Actions and select **Add Actions > Send email**.
2. To add the triggered alerts list action, go to Trigger Actions and select **Add Actions > Add to Triggered Alerts**.

3. To add the log event action, go to Trigger Actions and select **Add Actions > Log event**.
4. Click **Save**.

Your alert sends you an email, adds your alert to the list of triggered alerts, and logs the event every time your set conditions trigger the alert.

View and edit your alert

After you save your alert, you can view and edit it in the **Alerts** page. In this step, remove the **Log event** trigger action from your alert.

1. Click **Alerts** on the Splunk Light bar.
2. Find your alert and click **Open**.
3. Edit alert type and trigger conditions by clicking the **Edit** menu. In Trigger history every event that triggered an alert and their respective timestamps are stored.
4. Under **Actions** click **Edit**.
5. Click **Remove** next to the **Log event** trigger action.
6. Click **Save**.

Your alert reflects your changes.

View the Triggered Alerts list

When you select **Add to Triggered Alerts** in Trigger Actions, the alert populates the Triggered Alerts page. Use this page to keep track of alerts across different searches.

1. Click the sidebar menu on the Splunk Light bar.
2. Select **Activity > Triggered Alerts**.
3. Sort by **Owner**, **Severity**, and **Alert**.
4. (Optional) Click **View results** and **Open alert** on your alert.

You have completed this scenario.

Creating a dashboard scenario

Create and manage a dashboard using Splunk Light

Dashboards let you easily visualize your data. Dashboards are made up of panels powered by searches, which can display search boxes, fields, charts, tables, and lists.

In this scenario, you construct a dashboard made up of two prebuilt panels, a panel from an inline search, two table panels, and a single value panel. You also learn how to view and edit your dashboard to best fit your needs.

What you need for this scenario

To complete this scenario, first ensure that you have Splunk Light installed and running. You must be using an on-premise version of Splunk Light. Additionally, ensure that you can enable the Splunk Add-On for Unix and Linux.

Scenario overview

Complete each of these steps to reach the goal of a completed functional dashboard.

1. Enable the Splunk Add-On for Unix and Linux.
2. Create a dashboard.
3. Add prebuilt panels to a dashboard.
4. Add a dashboard panel from a search.
5. Add tables to a dashboard.
6. Add a single value panel to a dashboard.
7. Edit dashboard panels.
8. Organize and view a dashboard.

Enable the Splunk Add-on for Unix and Linux

The first step in this scenario is to enable the Splunk Add-on for Unix and Linux. You can analyze data from your deployment as well as understand the functionality of dashboards.

1. Log in to Splunk Light.
2. Navigate to **Data > Apps and Add-ons**.
3. Click **Enable** on the Splunk Add-on for Unix and Linux.
4. Click **Set Up**.
5. Click **Enable** on all inputs.
6. Click **Save**.
7. Restart your Splunk Light instance for the changes to take effect.

By enabling the inputs, you set bash scripts to run at specific intervals and collect event data in the os index. Now that you have these inputs set to enable, Splunk Light can access and analyze your data.

Confirm the add-on is enabled

Ensure that your inputs are enabled.

1. Select **Data > Apps and Add-ons**.
2. Click **Objects** in the Splunk Add-on for Unix and Linux.
3. Click **Indexes**. The index os should have an event count greater than zero.

Create a dashboard

Before you start adding panels, you need to create your dashboard. When creating your dashboard you can edit the permissions and description of the dashboard, but this is outside the scope of this scenario.

1. Click **Dashboards** in the Splunk Light bar.
2. Click **Create New Dashboard**.
3. Type a title for your dashboard.
4. Click **Create Dashboard**.

You are now in your new dashboard.

Add prebuilt panels to a dashboard

You can begin populating your dashboard using prebuilt panels. Prebuilt panels are simple XML code that can be shared among dashboards. You cannot edit the title, search, or visualizations of the panel from the dashboard reference. The inputs that you enabled in the Splunk Add-On for Unix and Linux power the prebuilt panels.

1. Ensure you are within your new dashboard. If you are not, go to **Dashboards > Your dashboard** and click **Edit**.
2. Select **Add Panel > Add Prebuilt Panel**. You have four prebuilt panels available. The inputs that you enabled earlier power these prebuilt panels.
3. Select Unix - Input ingestion over 24 hours.
4. Click **Add to Dashboard**.

5. Select Unix - Network Information - Top Inbound/Outbound Hosts.
6. Click **Add to Dashboard**.
7. Click **Save**.

Your dashboard now contains two prebuilt panels.

Add a dashboard panel from a search

To visualize your data, you first need to execute a search. Splunk software provides visualizations that together suit all types of data. In this scenario, search for all processes that are running under the root user.

1. Click **Search** in the Splunk Light bar.
2. Type the following into the search bar.

   ```
   sourcetype=ps user=root
   ```

   You now see all processes that are running under the **root** user.
3. Click **Save As**.
4. Click **Dashboard Panel**.
5. Add your list of events to your existing dashboard.
6. Name your panel Processes running under the root user.
7. Click **Save**.
8. To view your changes, click **View Dashboard**.

Your dashboard now contains three panels: two prebuilt panels, and one powered by an inline search.

Add tables to a dashboard

Tables are an easy way to visualize your data. In this scenario, you create two tables using different search syntax.

Use the table command to generate a table

To build a table, you can use a `table` command. The `table` command is a generating command. Generating commands fetch information from the indexes, without any transformations. In this scenario, make a table of your event type, source type and hour data to visualize your activity.

1. Click **Search** on the Splunk Light bar.
2. Type the following into the search bar.

   ```
   index=os /var/log sourcetype!=ps
   ```

   This lists all events within `/var/log` that are not of source type ps.
3. To add fields to the **Selected Fields**, click **All Fields**.
4. Select **date_hour** and **event type**. These fields are now included in the search results.
5. Type the following into the search bar.

   ```
   index=os /var/log sourcetype!=ps | table eventtype sourcetype
   date_hour | sort –date_hour
   ```

   This displays a table with the columns in the same order as they are typed.
6. Click **Save As** and click **Dashboard Panel**.
7. Add your table to your existing dashboard.
8. Name your panel Event and source type by time.
9. Click **Save**.
10. To view your changes, click **View Dashboard**.

Create a table from a search

You can create a table using a series of pipes. In this scenario, create a table of process counts by user.

1. Click **Search** on the Splunk Light bar.
2. Type the following into the search bar.

   ```
   sourcetype=ps | stats count(user) by user | sort –count(user)
   ```

   This creates a table of users and process counts, organized by highest process count.
3. Click **Save As**, and click **Dashboard Panel**.
4. Add your table to your existing dashboard.
5. Name your panel Process counts by user.
6. Click **Save**.
7. To view your changes, click **View Dashboard**.

Your dashboard now contains five panels: two prebuilt panels, one powered by an inline search, and two table panels.

Add a single value panel to a dashboard

Some searches return a single value output. You can visualize this data with Splunk Light?s single value display options. In this scenario, use your data to determine the number of root users.

1. Click **Search** in the Splunk Light bar.
2. Type the following into the search bar.

   ```
   sourcetype=ps root | stats count
   ```
3. Click the **Visualization** tab.
4. In the chart type menu, select **Single Value** (appears as a 42). You can format the color of the display for specific number ranges, indicating when a value gets too high or too low.
5. (Optional) To see your data as a radial gauge, select **Radial Gauge** from the chart type menu. Format it for the correct number ranges.
6. Click **Save As**, and click **Dashboard Panel**.
7. Add your visualization to your existing dashboard.
8. Name your panel Number of root users.
9. Click **Save**.
10. To view your changes, click **View Dashboard**.

Your dashboard now contains six panels: two prebuilt panels, an in-line search panel, two table panels, and a single value data panel.

Edit dashboard panels

After charts and visualizations are in your dashboard, you can edit the presentation of your panels. You can edit the title, style, formatting, and time range of your dashboard panels directly from the dashboard reference. In this scenario, you edit all aspects of your dashboard panels.

Step 1: Modify panel settings

You can edit the visualization type, visualization format, and search time range from the dashboard reference.

Navigate to your dashboard.

1. Click **Dashboards** in the Splunk Light bar.
2. Select **Yours**.

3. Select your dashboard.
4. Click **Open**.
5. Select **Edit > Edit Panels**.

You can now edit the various aspects of your dashboard panels

Edit visualization type

Edit a visualization to configure its appearance. Edit your Process counts by user table panel to display it as a bar chart.

1. Scroll to your Process counts by user table panel.
2. Click the visualizations menu, which appears as a table icon.
3. Select the **Bar Chart** visualization option. Certain visualizations are recommended for the data type.

Edit visualization format

You can edit the visualization format and behavior. Format options vary by visualization type. Edit your Processes running under the root user panel to display more clearly.

1. Scroll to your Processes running under the root user panel.
2. Click the paint brush menu.
3. Edit your visualization formatting by selecting options. The visualization shows your changes as you make them.
4. Click **Yes** for **Row Numbers**.
5. Click **No** for **Wrap Results**.

Edit search time range

You can edit the time ranges of the searches that power your visualizations. To edit a prebuilt panel from the dashboard reference, first convert it to an inline panel. Edit your Unix - Input ingestion over 24 hours panel to only include results from the past hour.

1. Click the lock icon on the top right of your Unix - Input ingestion over 24 hours panel.
2. Click **Convert to Inline Panel**.
3. Click **Convert**.
4. Click the magnifying glass icon menu.
5. Click **Edit Search String**.

6. Click **Time Range** to select or create the time range over which your search is applied.
7. Click **Last 60 minutes**.
8. Click **Save**.

Step 2: Edit panel titles

Edit prebuilt panel titles

Edit your Unix - Input Ingestion over 24 hours panel title to reflect the changes you made to the search.

1. Click the lock icon on the top right of your Unix - Input Ingestion over 24 hours panel.
2. Ensure that you converted this panel to a prebuilt panel. If not, then click **Convert to Inline Panel**.
3. Click the title box of the panel, and rename your panel accordingly to Unix - Input Ingestion over 1 hour.

You can edit panels you generate from a search or report can be edited directly in the dashboard reference. To edit all other panel type titles, click on the title of the panel and type your new title.

Step 3: Save all of your changes

When finished with the above, click **Done** at the top right of the page to save the changes you made to your dashboard panels.

Organize and view a dashboard

Rearrange panels

You can rearrange panels within your dashboard.

1. Click **Dashboards** in the Splunk Light bar.
2. Select your dashboard and open it.
3. Select **Edit > Edit Panels**.
4. Click the dotted-line bar at the top of each panel to drop and rearrange your panels.
5. Click **Done**.

View a dashboard

Now that your dashboard is complete, you can view your various data visualization panels in one place. To view your dashboard:

1. Click **Dashboards** in the Splunk Light bar.
2. Click **Yours**.
3. Select your dashboard.
4. Click **Open**.

You are now within your dashboard and have completed this scenario.