GUIDE

STORMSHIELD LOG SUPERVISOR

DASHBOARD & VISUALIZATION GUIDE

Version 2

Document last updated: July 4, 2024

Reference: sls-en_dashboard_visualization_gde

Table of contents

Change log

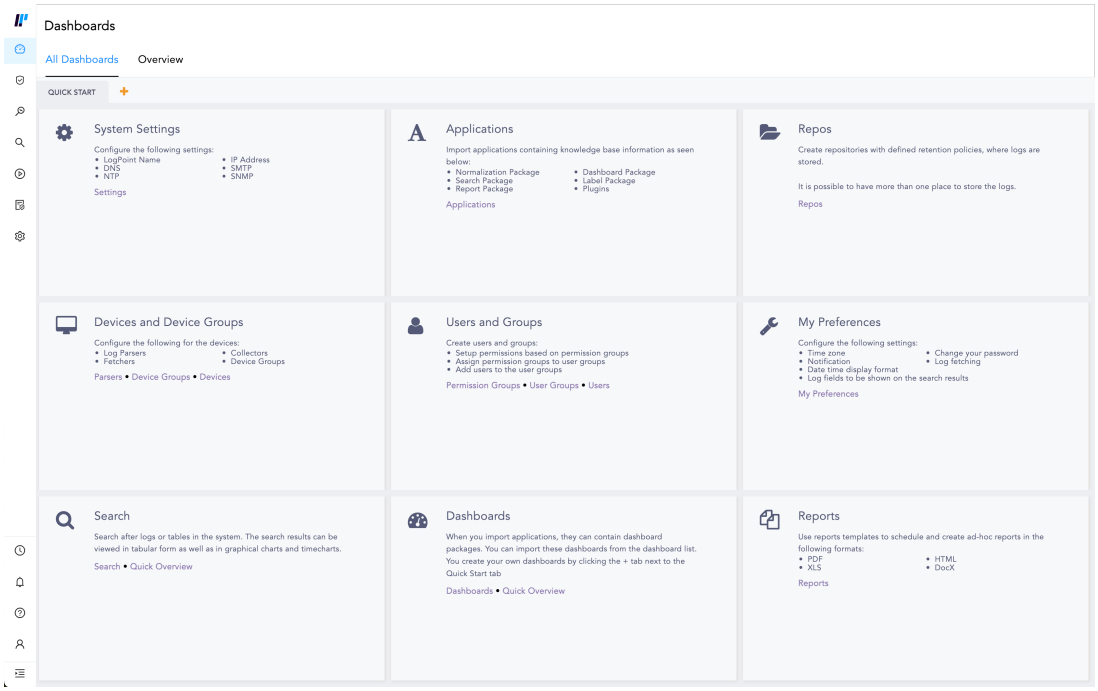
Date	Description
July 4, 2024	New document

Getting started

Welcome to the SLS version 2 Dashboard & Visualization Guide.

Dashboard is data visualization updated in real-time. SLS comes with two main, pre-configured dashboards: **All Dashboards** and **Overview**.

- All Dashboards allow you to create dynamic dashboards by adding multiple widgets. A widget can contain charts, tables, and graphs generated by a search query. A dashboard can also have diagrams, lists, and tables. If needed, you can change the widgets' height, width, and positioning.
  All Dashboards start with **Quick Start**, from which you can easily access most of the features of SLS.

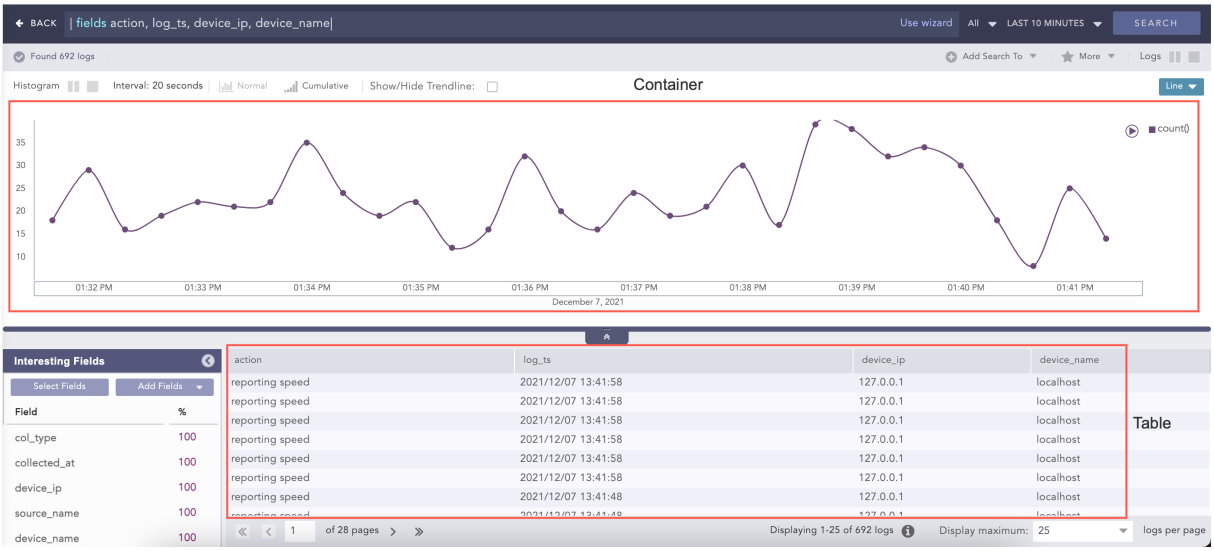


- Overview allows you to monitor your system operations and real-time cybersecurity incidents based on key measures, workflows, and behavioral patterns. This is a static dashboard that you cannot customize.

SLS provides an array of visualization options including regular bar, line and column charts, various other statistical tools have been added. Charts are not only an aesthetically pleasing way to view search results, they also help make data analysis easier.

These options are available in:

- Search Interface
- Dashboards (Widgets)
- Search Templates



In this document, Stormshield Log Supervisor is referred to in its short form SLS. Images used in this document are from the partner vendor's (Logpoint) software program. In your SLS, the graphics may vary but user experience is exactly the same.

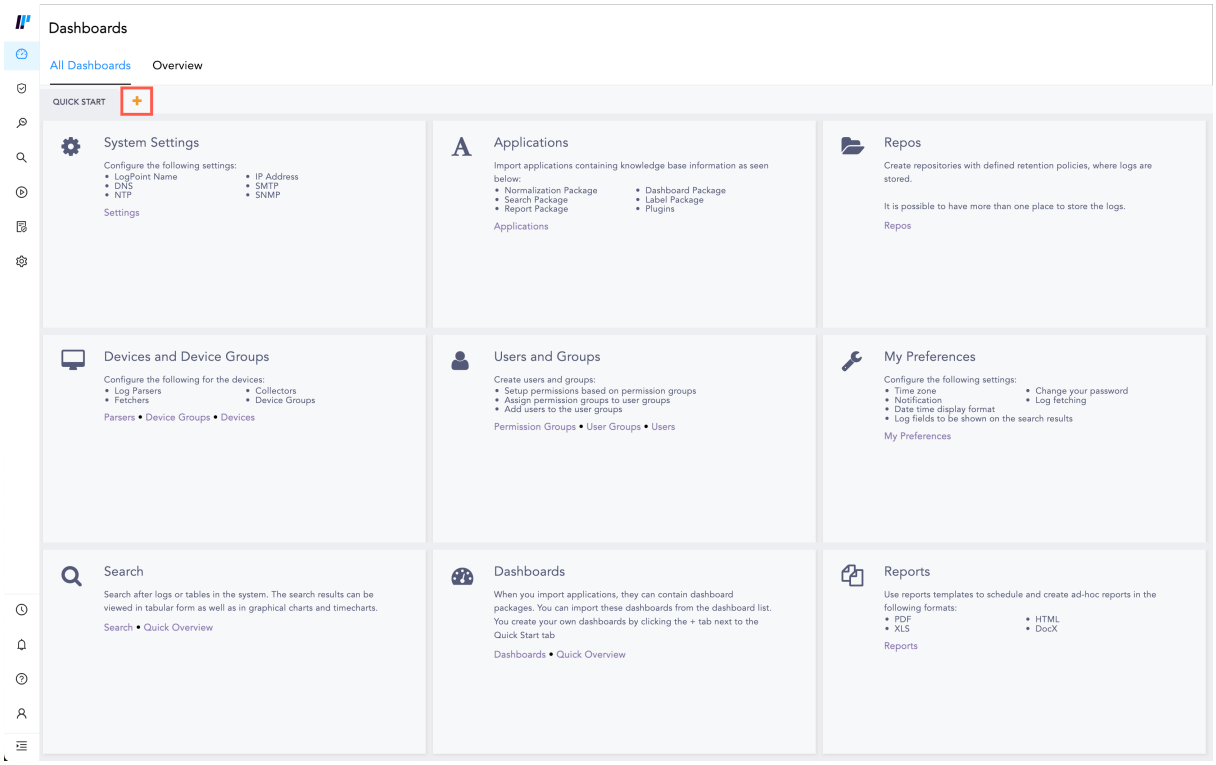
All Dashboards

All Dashboards allows you to design your own dashboards based on different analytics that provide an overview of SLS data. It displays data for a specified period of time and includes dashboards designed for different user roles. All Dashboards aggregates widgets from various sources so you can group different widgets as you would like to view them. All users can view All Dashboards. You don't need a SLS admin role.

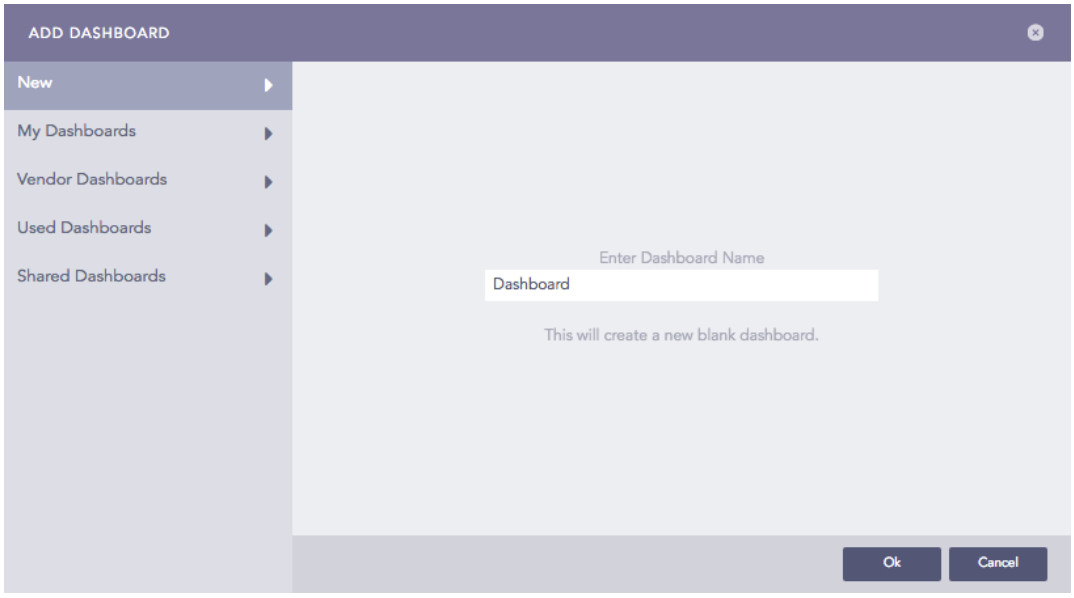
The dashboard creates an overview of any data you wish to monitor regularly, enabling you to react faster to information.

Creating a Dashboard

1. Go to `Dashboard` from the navigation bar.



2. Click on **All Dashboards**.
3. Click **+**.

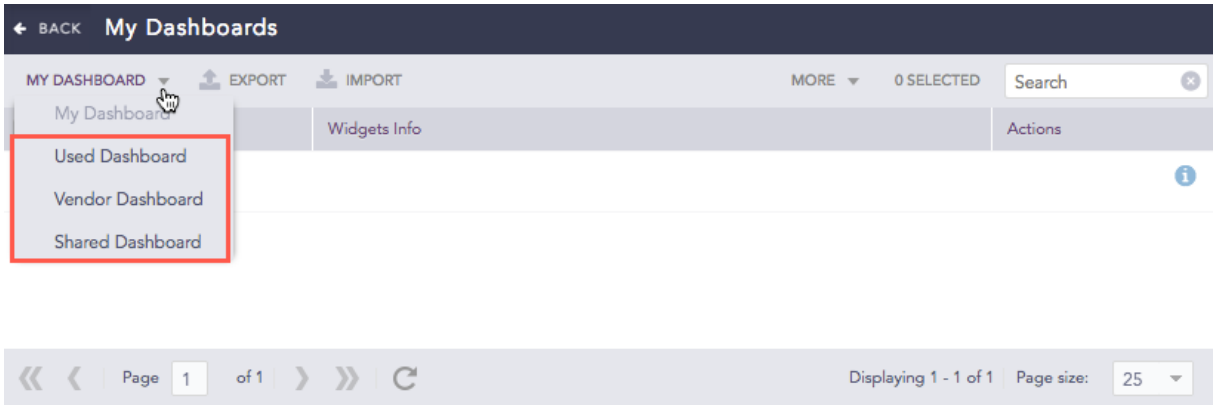


4. Enter the **Dashboard Name**. You can also pull dashboards from the tabs on the left.
5. Click **Ok**.

Dashboard Types

All Dashboards list all the dashboards created in the SLS and display the information of the widgets in each dashboard.

At the top left, you can switch among **My Dashboards**, **Used Dashboards**, **Vendor Dashboards** and **Shared Dashboards** from the drop-down.



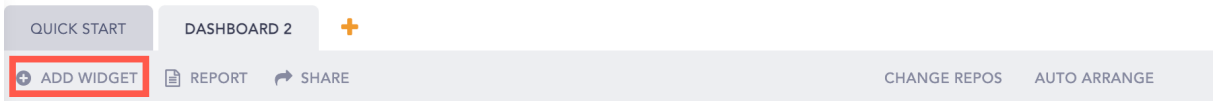
- My Dashboards: The ones created by you. You can **Clone**, **Share/Unshare**, **Lock/Unlock**, and **Delete** these dashboards from **Actions**.
- Used Dashboards: The ones you used.
- Shared Dashboards: The ones shared between users. Click the **Use** icon from **Actions** to use it.
- Vendor Dashboards: The ones included with SLS. Click the **Use** icon from **Actions** to use it. Click the **Clone** icon make a copy of the dashboard where you can apply changes.

Dashboard Tools

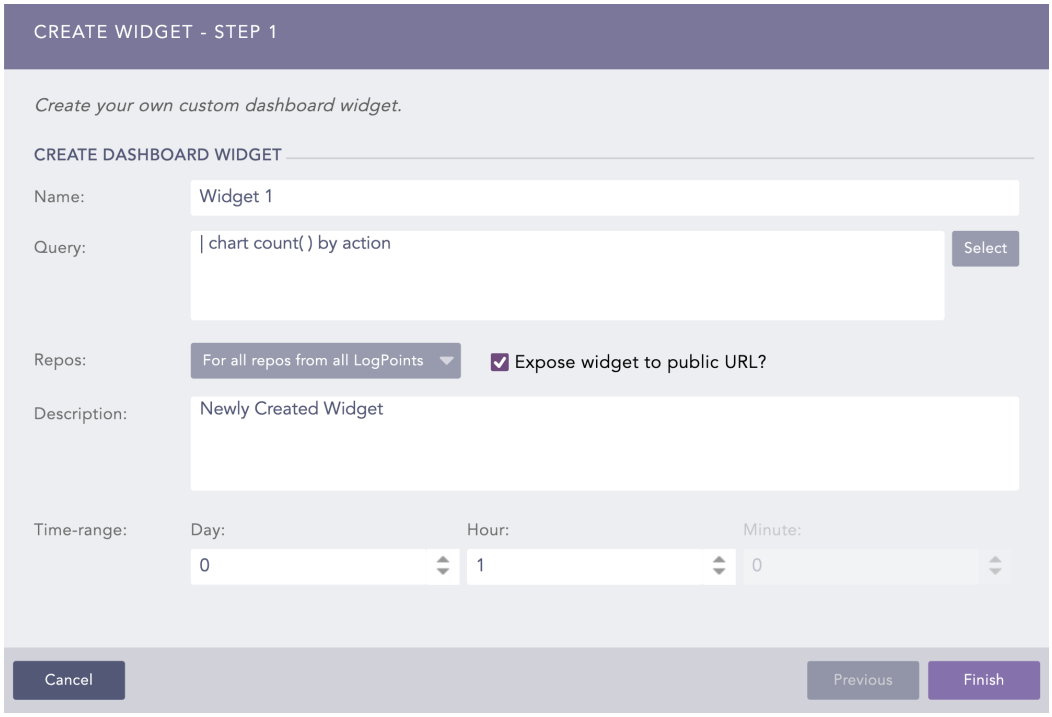
Adding a Widget

Widgets help you monitor logs in real-time. You can personally set up a widget and add it to a **Dashboard** of your choice. For example, if you want to monitor the firewall activities of devices, create a widget with the search queries related to the firewall.

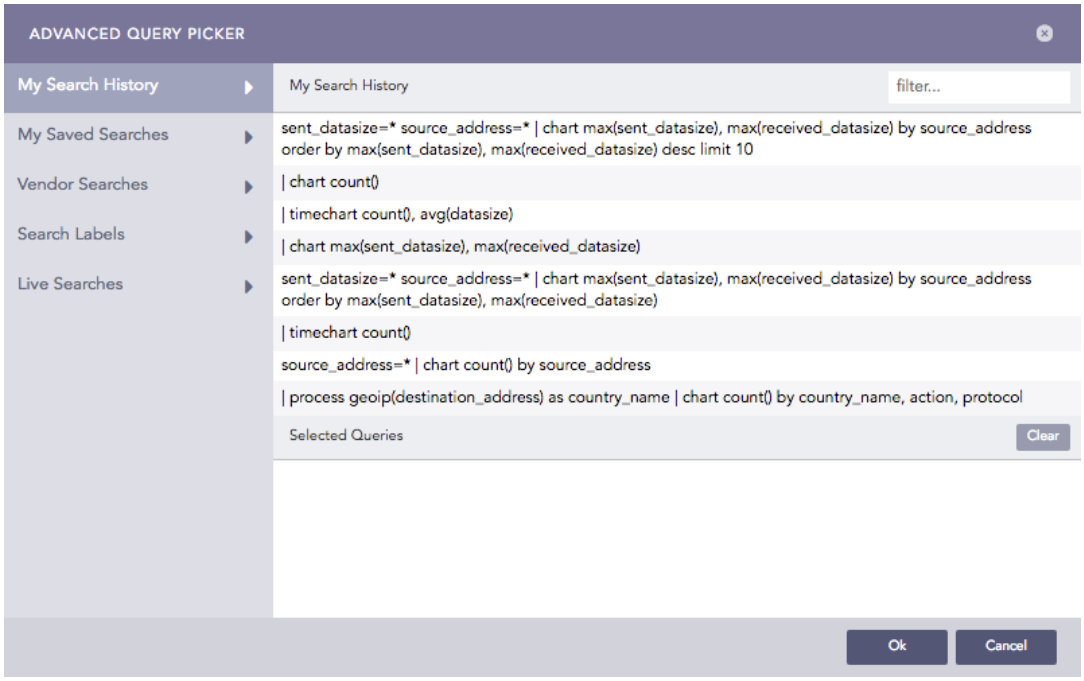
1. Go to `Dashboard` from the navigation bar.
2. Click on **All Dashboards**. Select a **Dashboard** and click **Add Widget**.



3. Enter a **Name** for the widget.

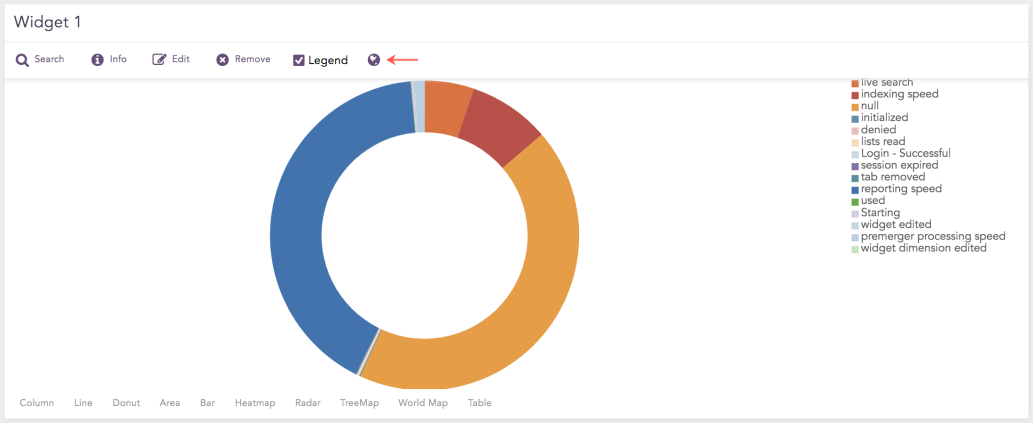


4. Enter a **Query**. Alternatively, click **Select** to choose any query from the **Advanced Query Picker**.



If you choose the **Advanced Query Picker**, select a query from the lists.

5. Choose the **Repos** from where you want to generate the logs.
6. Select a **Limit** for the number of logs generate.
7. Select **Expose widget to public URL?** to share the widget publicly. When you share a widget with other users they don't need permission to view it.

> **NOTE**
>
> - The user the widget is shared with does not need the credentials to view the shared widget.
> - If you selected **Expose widget to public URL?**, you now have the option to **Open public URL**. Click it opens the search results in a new window.
>
> 
>

8. Provide a **Description** for the widget.
9. Select a **Time Range** for the logs in the repos.

NOTE

- You can set a time range in minutes, hours, or days..
- The maximum limit of the time range for the **day** field is 30.

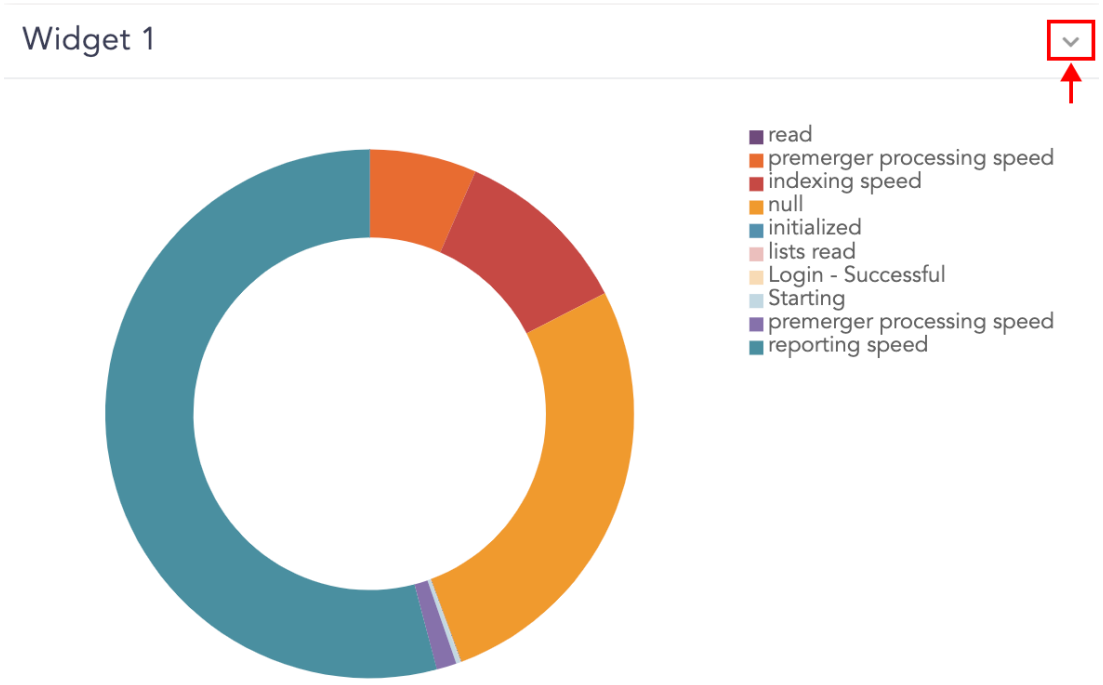
10. Click **Finish**.

NOTE
If **Data Privacy Module** is enabled, **Can Request Access users** can only view encrypted data.

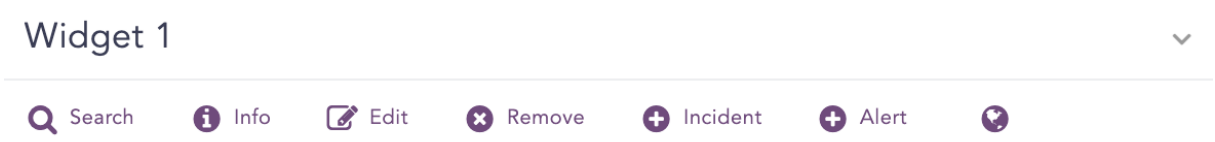
IMPORTANT
When configuring repos for a new dashboard, only select the relevant ones. Using a large number of repos impacts SLS performance.

Editing a Widget

At the top-right, click the **Widget Options** icon.



Edit a widget options include:



You can **Search** for the results, get **Info**, **Edit**, **Remove**, and open the widget in **Public URL**. You can also toggle the display of the **Legend** if there is one.

Additionally, You can create **Alerts** and **Incidents**.

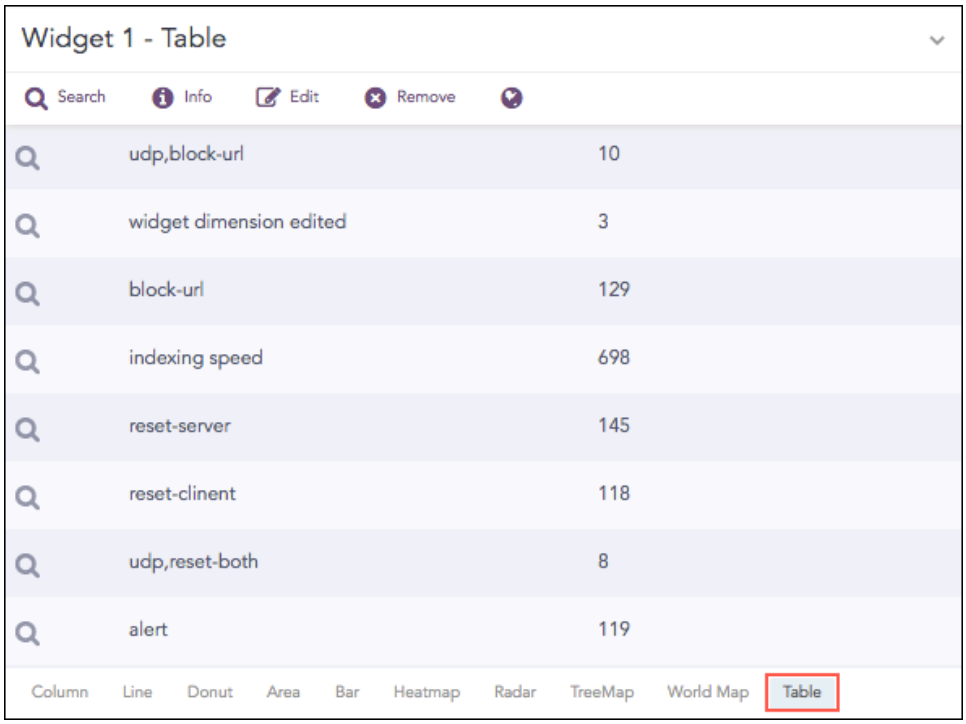
You can also create and use graphs, including tables, area charts, line charts, bar charts, column charts, gauge charts, display charts, and donut charts. The type of graph you can use depends on the type of search results.

Tables

The following query:

```
| chart count() by action
```

Generates the following table:

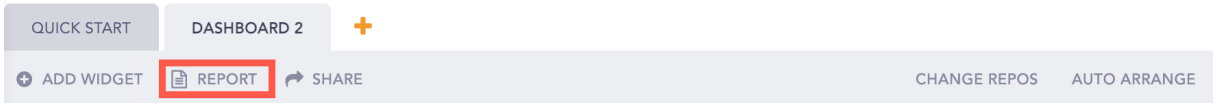


There are other graphs. Click on their link to learn more.

- Area Chart
- ATTCK chart
- Bar Chart
- Bubble Chart
- Clustered Bar Chart
- Clustered Column Chart
- Clustered Line Chart
- Column Chart
- Day/Hour Heatmap Chart
- Display Chart
- Donut Chart
- Gauge Chart
- Heatmap Chart
- Line Chart
- Parallel Coordinate chart
- Radar chart

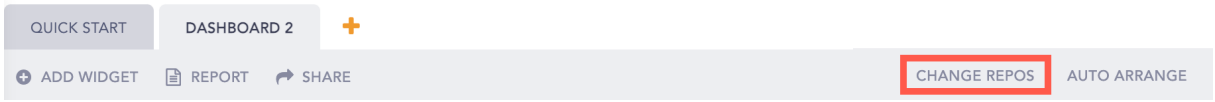
- Sankey chart
- Stacked Area Chart
- Stacked Column Chart
- TreeMap Chart
- World Map Chart

Report



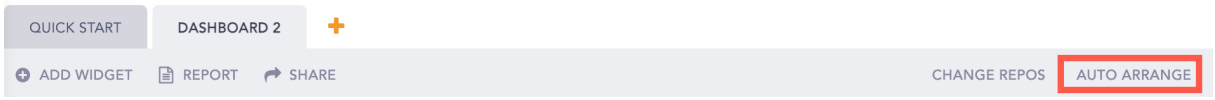
You can use dashboards to generate reports. A report replicates the contents and data in a widget. You can't schedule or change a report's layout in a widget.

Change Repos



You can change the **Repos** for log results of all the **Widgets** in the Dashboard.

Auto Arrange



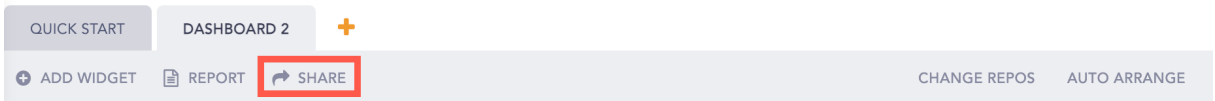
You can manually change the sizes and position of widgets, you can also use **Auto Arrange**.

Sharing a Dashboard

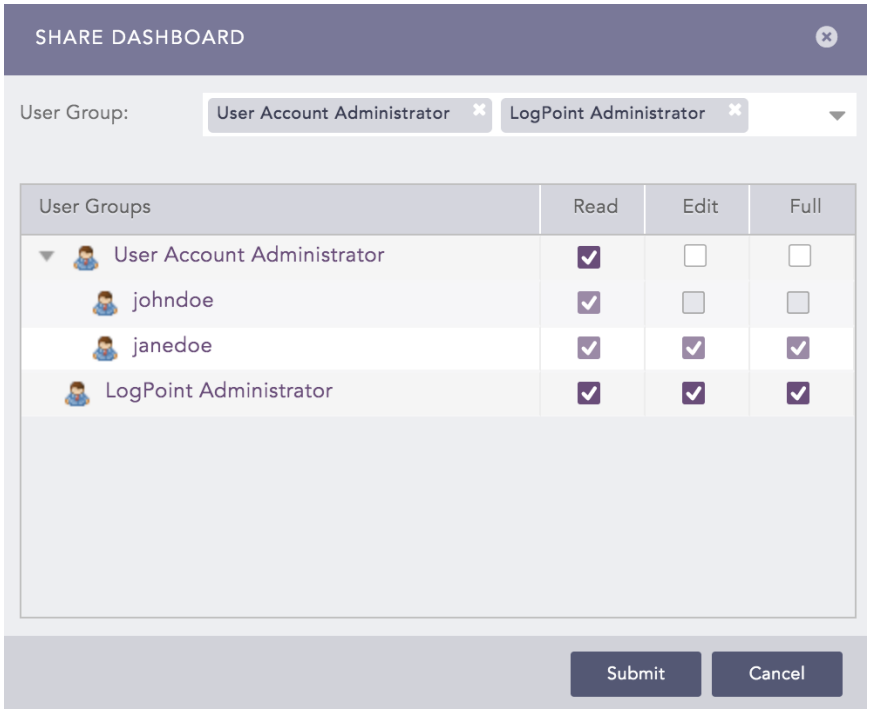
You can share a dashboard from All Dashboards with different users and give them the read, edit, or full permissions. Any changes made in the dashboard are visible to all the shared users. However, auto-arranging or re-arranging a widget's size and position is reflected only on the current user's dashboard.

Sharing a Dashboard from the Dashboard Page

1. Go to `Dashboard` from the navigation bar.
2. Select the dashboard you want to share and click **Share**.



3. Select a **User Group**. All the users in that user group are listed in the drop-down.
4. Select the **Read**, **Edit**, or **Full** permission for the users. Selecting the **Full** permission allows the user to read, edit, remove, and share the dashboard.

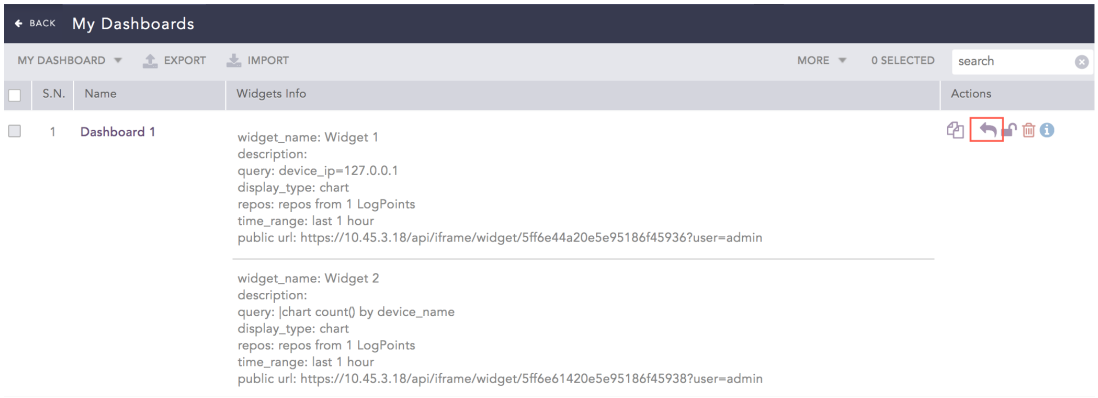


5. Click **Submit**.

> **NOTE**
>
> - You can't share UEBA or vendor dashboards with other users.
> - By default, a new user has the same permission as the user group they belong to.

Sharing a Dashboard from Knowledge Base

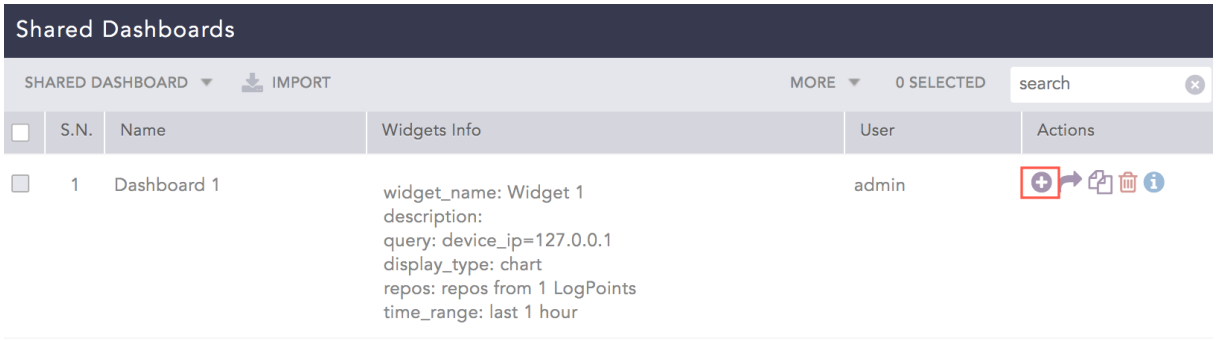
1. Go to `Settings >> Knowledge Base` from the navigation bar and click **Dashboard**.
2. Select **My Dashboard** from the drop-down.
3. Click the **Share** icon in the **Actions** column.



4. Select a **User Group**. All the users in that user group are listed in the drop-down.
5. Select the **Read**, **Edit**, or **Full** permission for the users.
6. Click **Submit**.

Using a Shared Dashboard

1. Go to `Settings >> Knowledge Base` from the navigation bar and click **Dashboard**.
2. Select **Shared Dashboard** from the drop-down.
3. Click the **Use** icon in the **Actions** column.

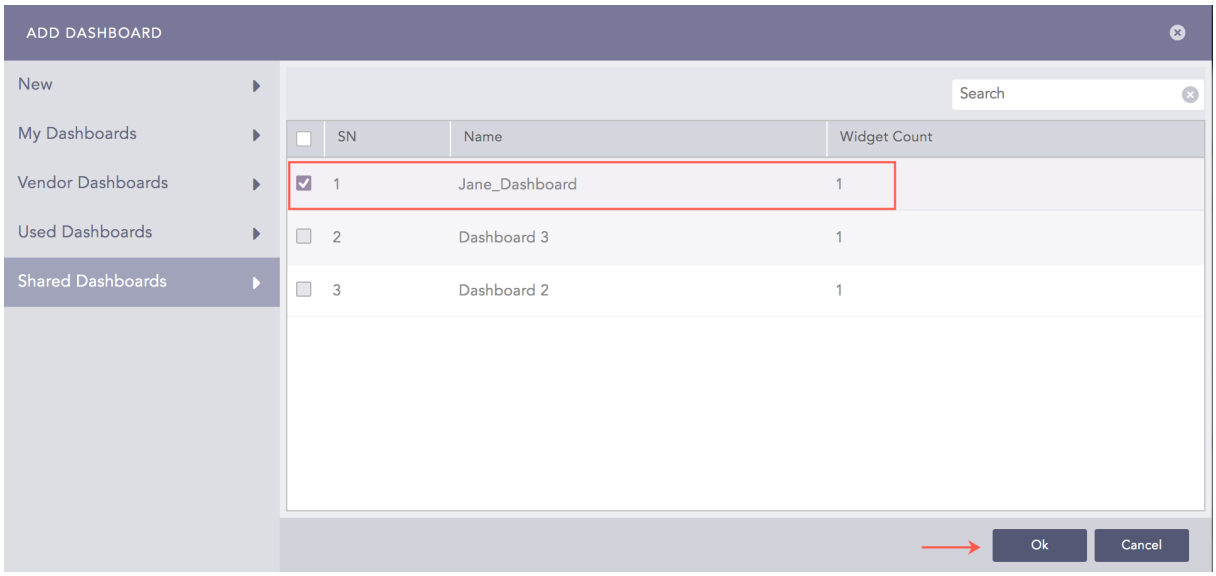


NOTE

- If a user does not have access to a repo used in a shared dashboard, the data displayed comes from repos they have access to.
- If only one repo is selected in the shared dashboard, and the user does not have access to the repo, the dashboard is empty.
- When a widget's graphs in a shared dashboard is changed by a user with the **Edit** or **Full** permission, the graphs are changed for all users. However, when a user with the Read permission changes a graph it only changes for them.

You can also use a shared dashboard from **Add Dashboard**.

1. Go to `Dashboard` from the navigation bar and click the + icon.
2. Select **Shared Dashboards**.
3. Select the dashboard you want to use and click **Ok**.



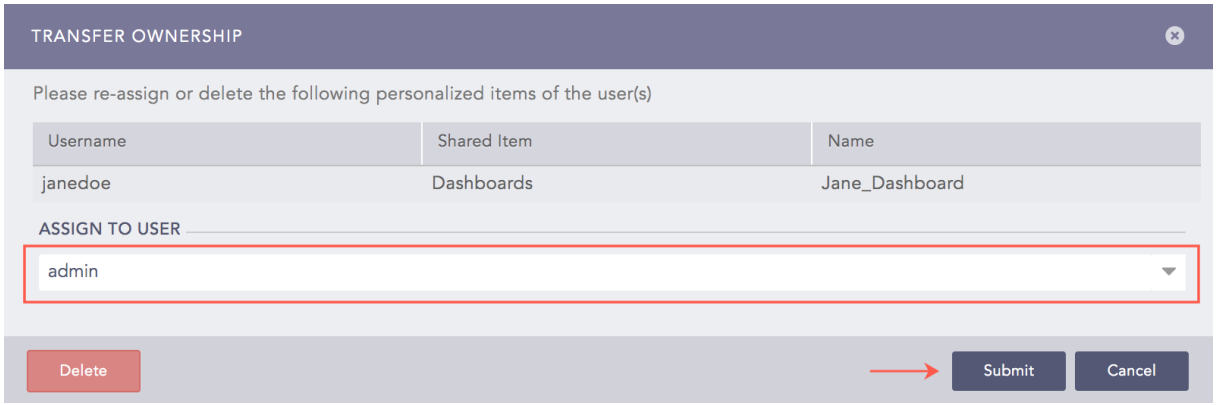
4. **Choose Repos** and click **Ok**.

Deleting a Shared Dashboard's Owner

1. Go to `Settings >> User Accounts` from the navigation bar and click **Users**.
2. De-activate the user by clicking the **De-Activate User** icon in the **Actions** column.
3. Click **Manage De-Activated Users**.
4. Click the **Delete** icon in the **Actions** column of the user.
5. Click **Yes**.

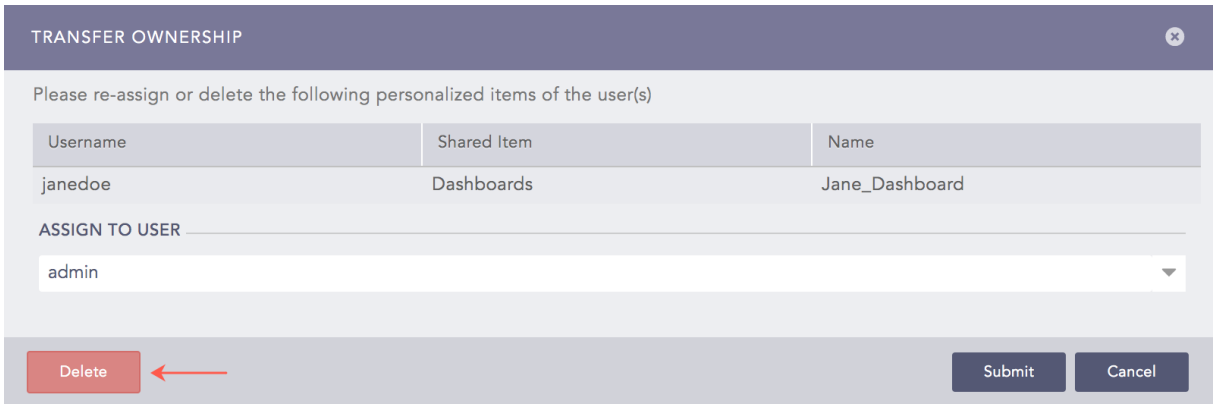
> **NOTE**
> You can do this using **Transfer Ownership** when you delete the user whose dashboard is being shared.

6. To transfer ownership, select a user from the drop-down and click **Submit**.



> **NOTE**
> The transferred dashboard is listed in **My Dashboards** of the new owner.

7. To delete the user and user's dashboard without transferring ownership, click **Delete**.

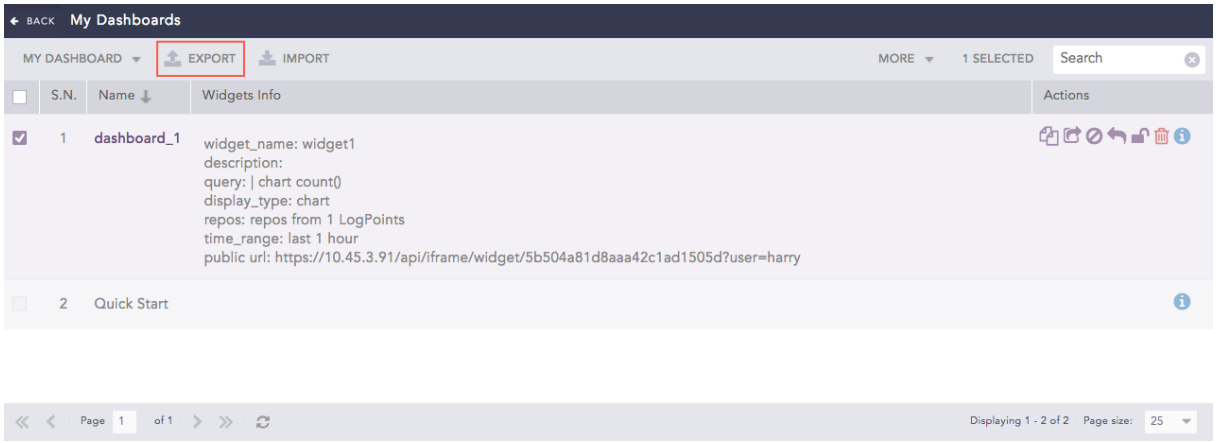


Importing and Exporting Dashboards

In addition to importing and exporting dashboards, you can also clone, share, unshare, lock, unlock, and delete them using the icons in the Actions column or from the More drop-down at the top-right corner. Click Details for more information about the dashboard.

Exporting Dashboards using Knowledge Base

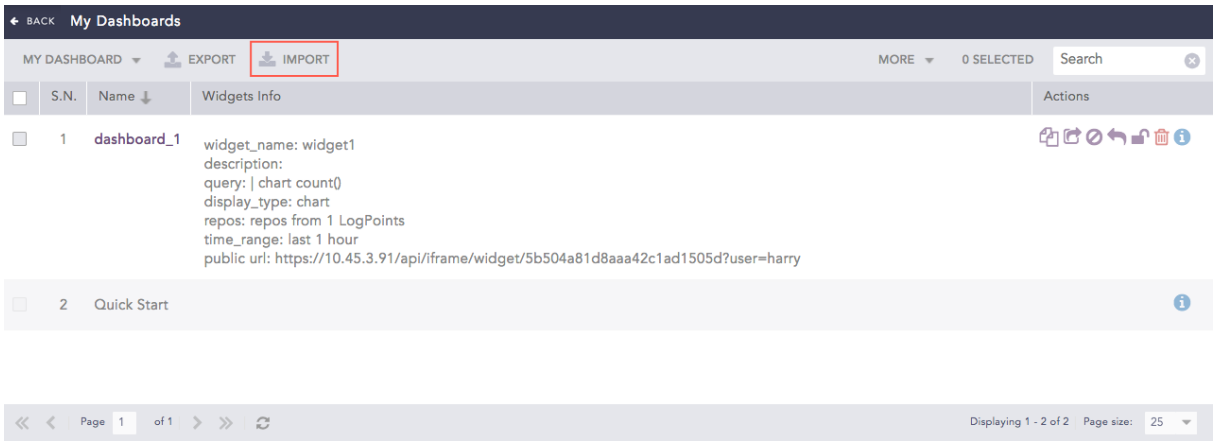
1. Go to `Settings >> Knowledge Base` from the navigation bar and click **Dashboards**.



2. Select the dashboards that you want to export.
3. Click **EXPORT**.
4. Save the **.pak** file as a backup or store in the computer system to use it in another SLS.

Importing a Dashboard using Knowledge Base

1. Go to `Settings >> Knowledge Base` from the navigation bar and click **Dashboards**.



2. Click **IMPORT**.
3. Browse and upload the **.pak** file containing the dashboards to import.
4. Click **Submit**.

NOTE

- Dashboards shared with you are listed in My Dashboards. You can edit and share this dashboard with other users. If you need to use the original dashboard, first delete yours. Then click **Shared Dashboard** and select it. You can also do this for vendor dashboards.
- You can drag and drop the widgets from one dashboard to another and must avoid dropping the widgets into a locked dashboard.

Customizable Drilldown from Dashboard Widgets

Customizable drill-down options are available in dashboard widgets. You can get more information from your search queries. Using drill down in your dashboard widgets lets you look at specific details of query results.

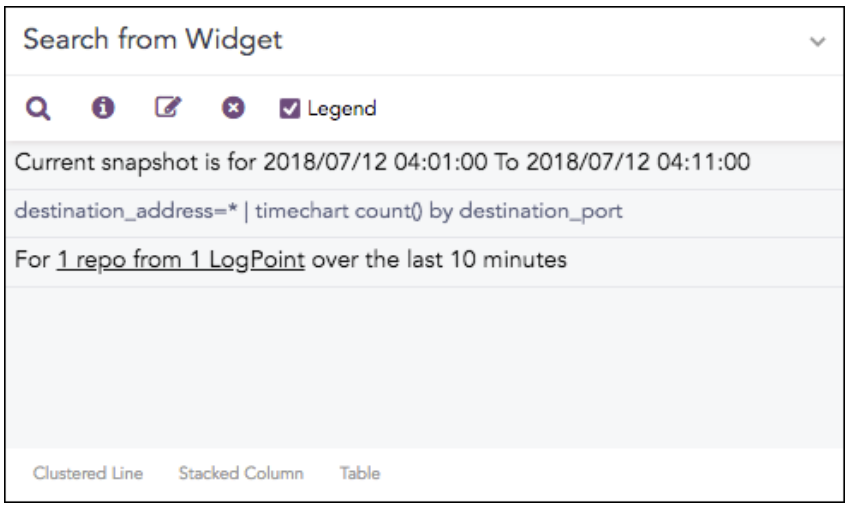
For example, when viewing results that include the **destination_address**, **destination_port**, **source_address**, and **source_port** in the query you can drill down in each individual parameter.

Here are two scenario examples.

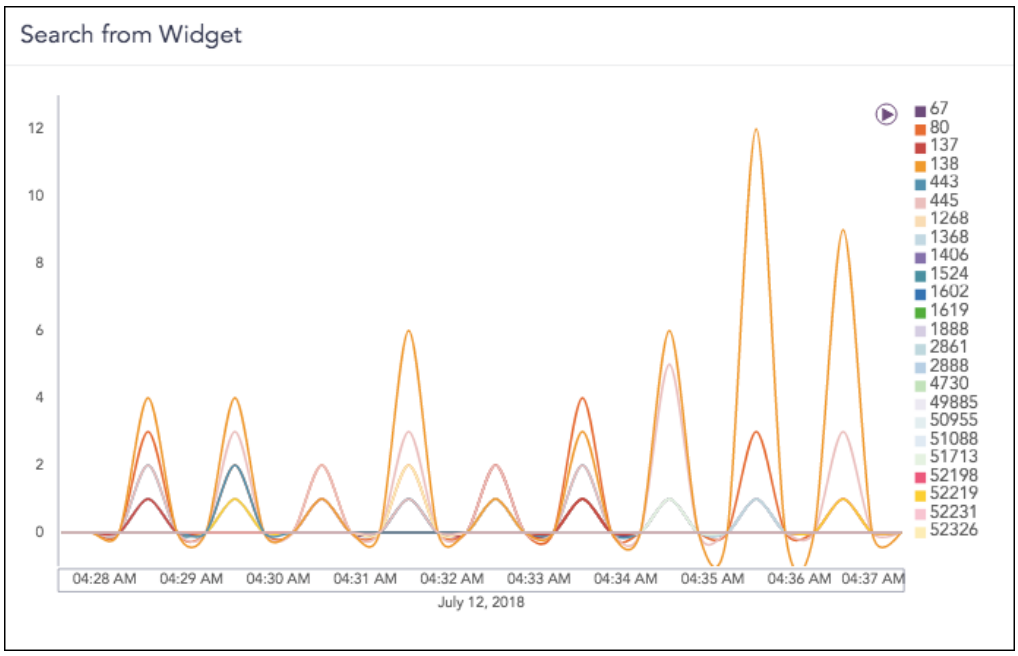
Non-Empty Search from Widget

A widget with the following search query:

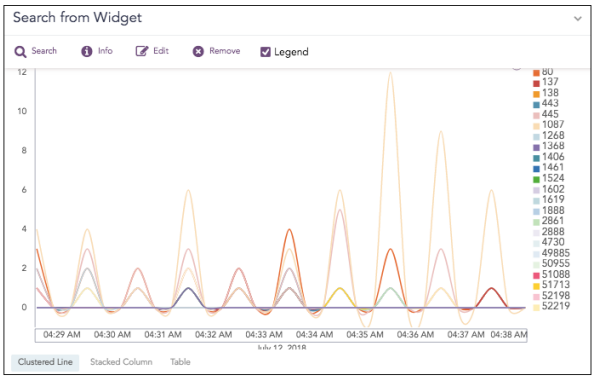
```
destination_address=* | timechart count() by destination_port
```



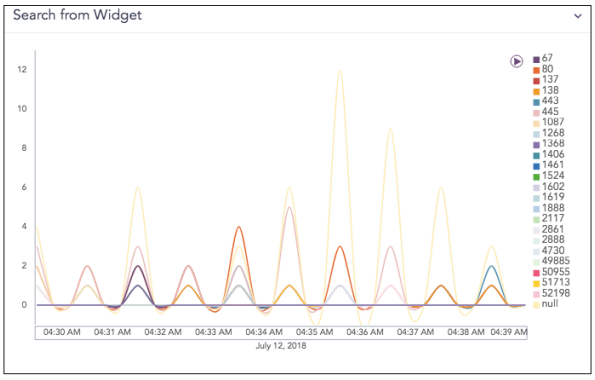
The query results are displayed as a graph.



You can toggle between **edit** and **non-edit** mode. In edit mode, you can select the graph type, for example Clustered Line Chart, Stacked Column Chart, and Tables. In non-edit mode, you can drill down.

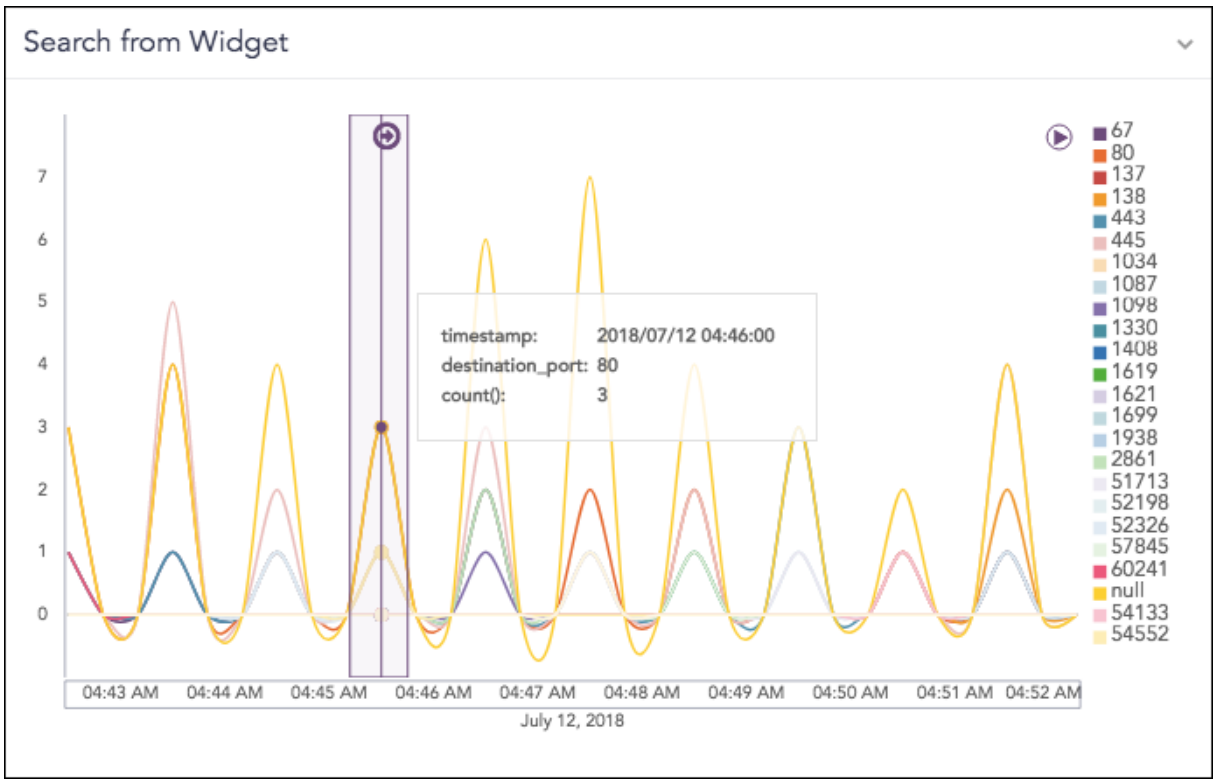


Edit Mode



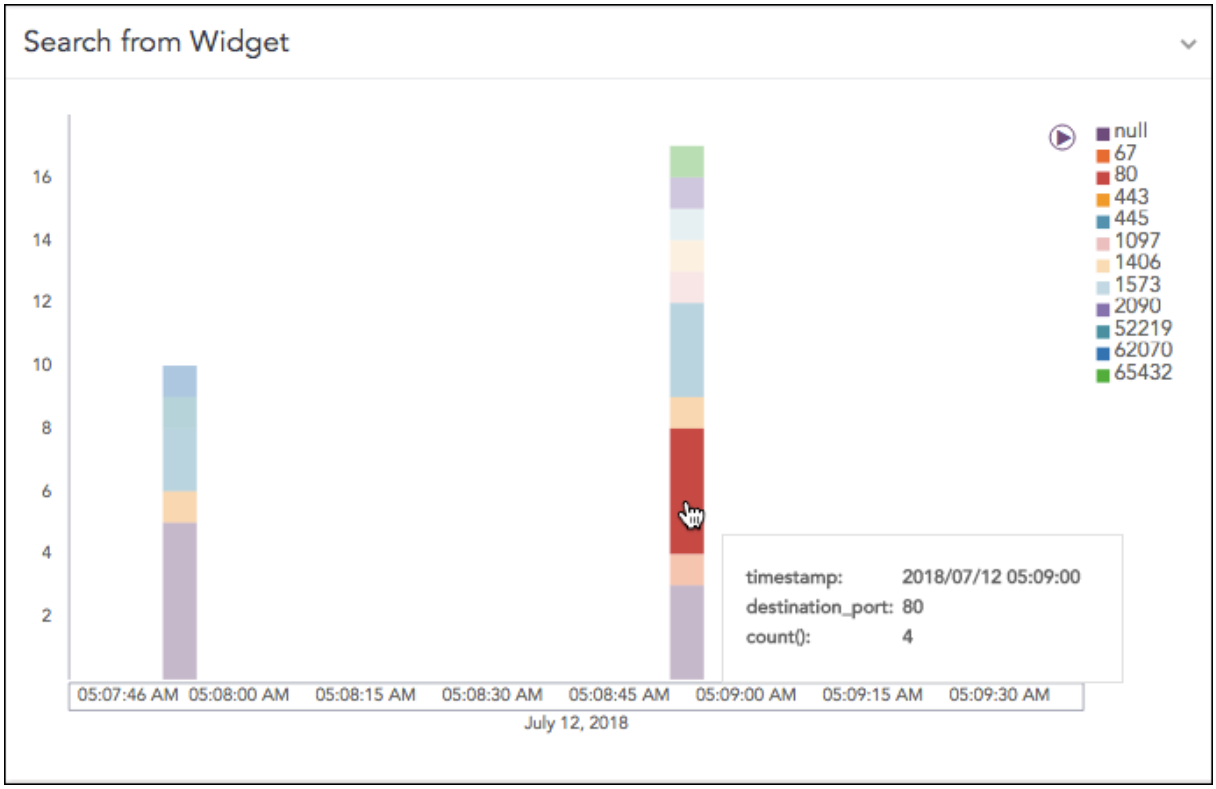
Non Edit Mode

Select **destination_port: 80** and **count(): 3** for to drill down.



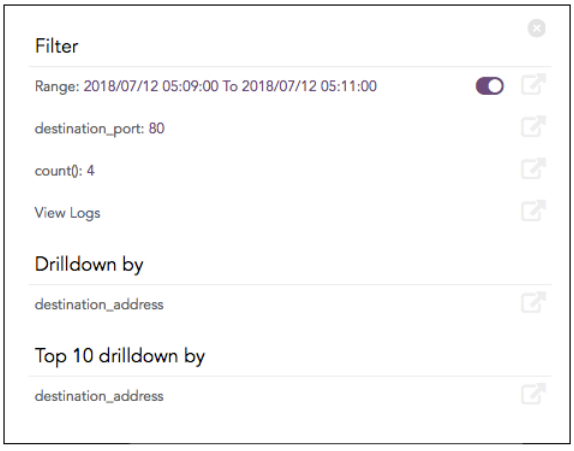
When you click on the highlighted result, you get the option to drill down one of the following specific parameters:

- Filter
- Drill down by
- Top 10 drill-down by

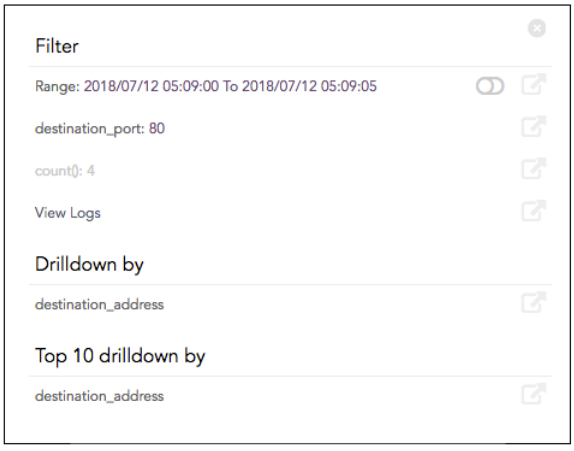


The **Filter** drilldown searches on **Range**, **destination_port** and **count()**. The **Drilldown by** and **Top 10 drill-down** searches for the destination_address.

The results of all three drilldown types can be opened and viewed in the same window or a new window. Enable **Range**.

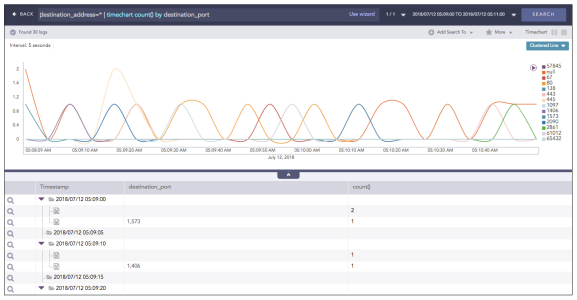


Enabled Time Range of 05:09:00 to 05:11:00

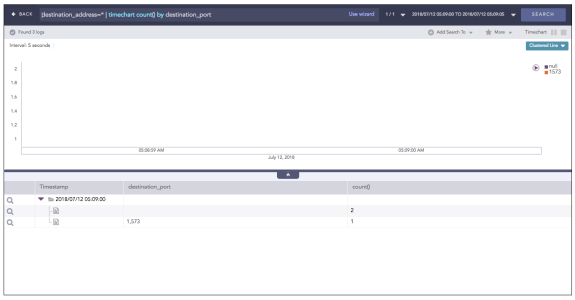


Disabled Time Range of 05:09:00 to 05:09:05

When drilling down on the **Range** value, the results opens on the same page.

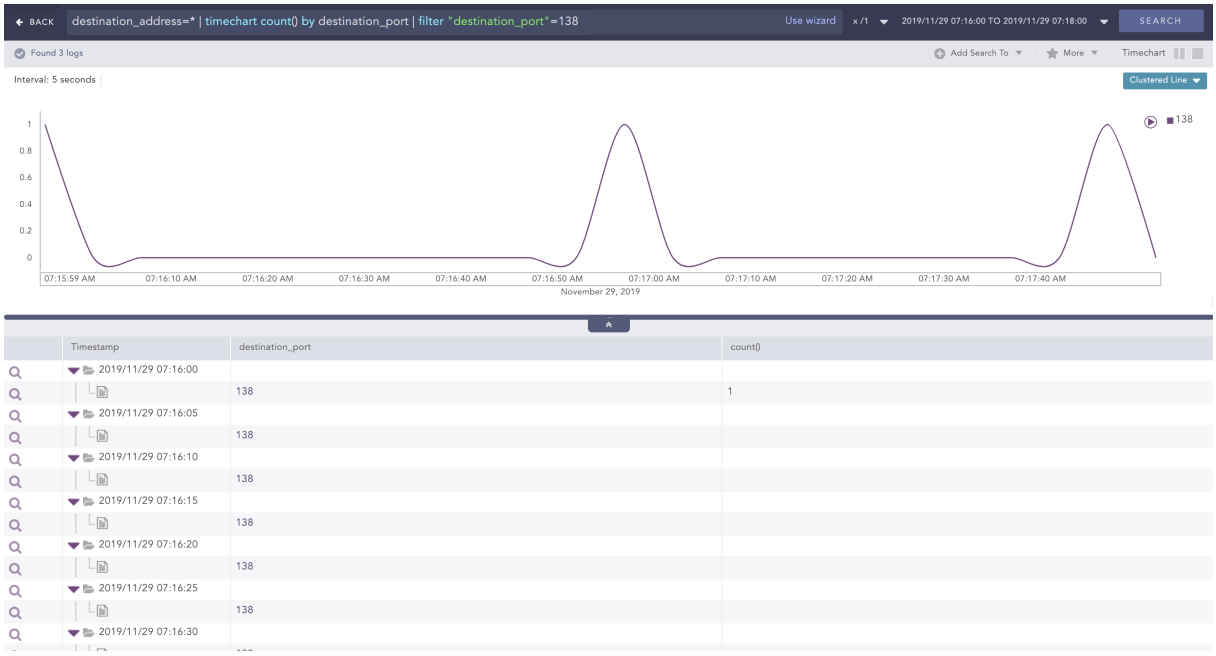


Search Result for Enabled Time Range of 05:09:00 to 05:11:00

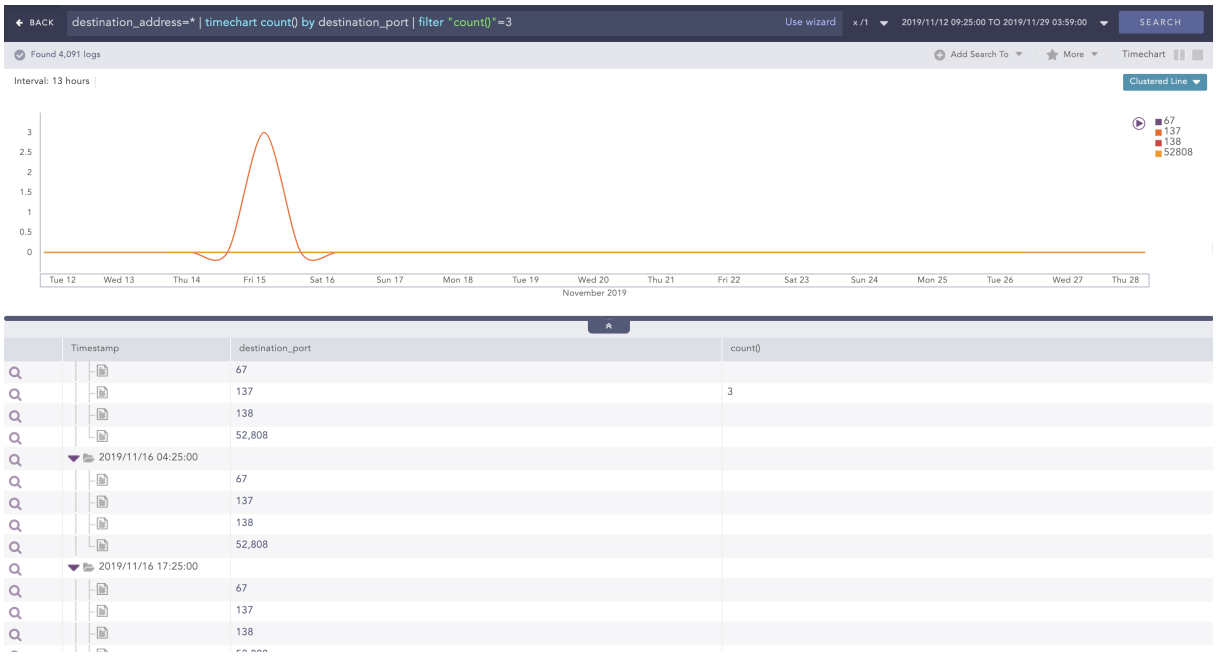


Search Result for Disabled Time Range of 05:09:00 to 05:09:05

When drilling down on **"destination_port"=138**, the results for the destination port opens in the same page.



When the drilling down is carried out on **"count()"=3**, the search results for the count open on the same page.

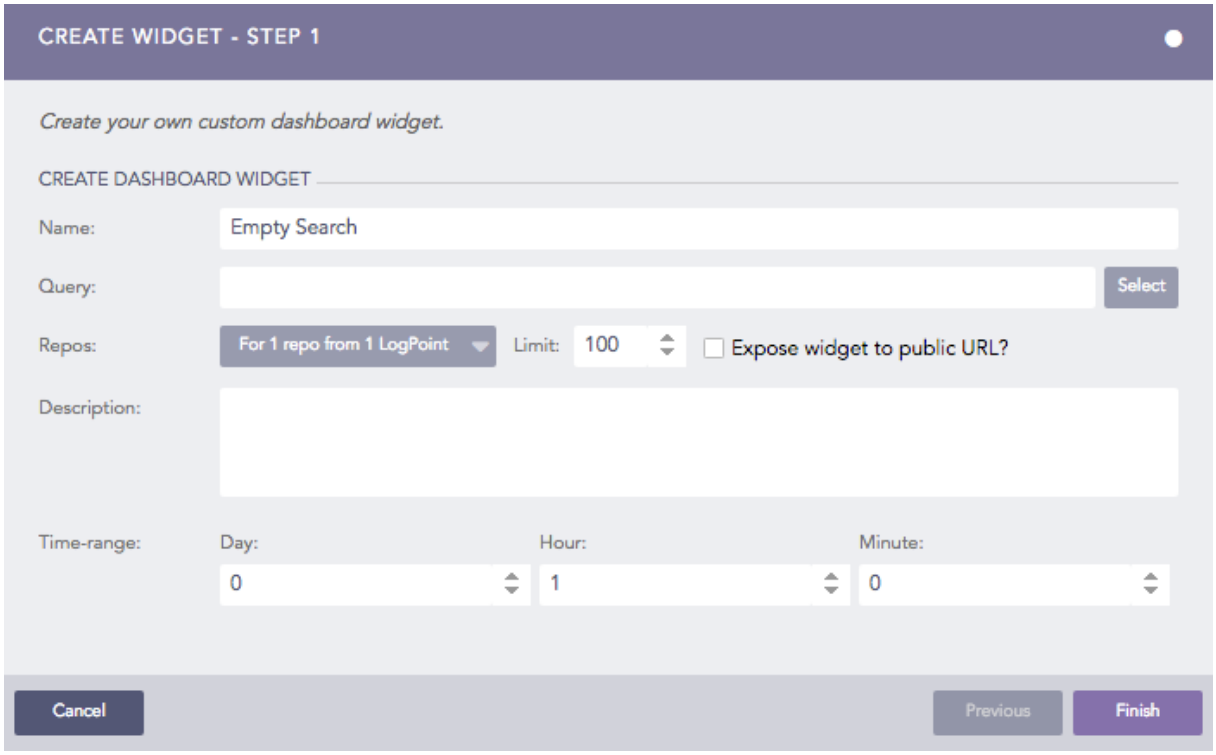


When drilling down on **destination_address**, the results open in the same page.

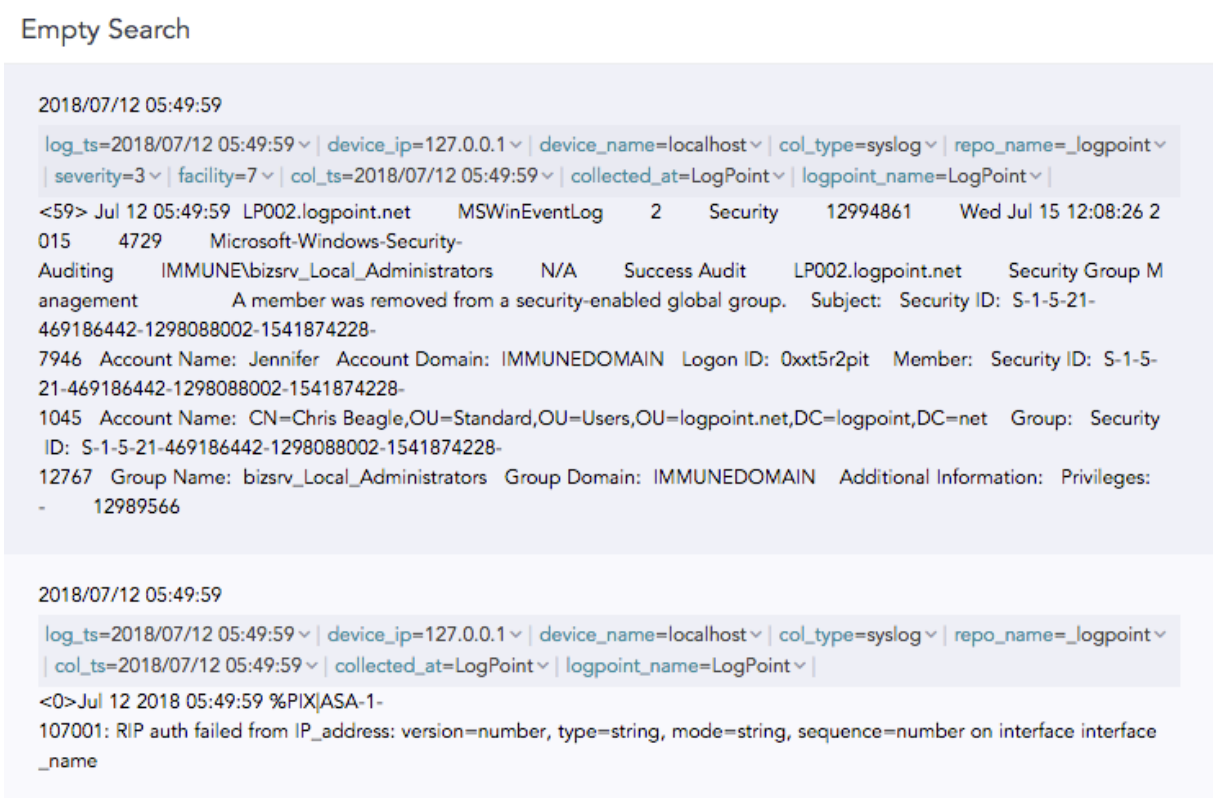
When drilling down on the **destination_address**, the results open in the same page.

Empty Search from Widget

This widget has no search query.

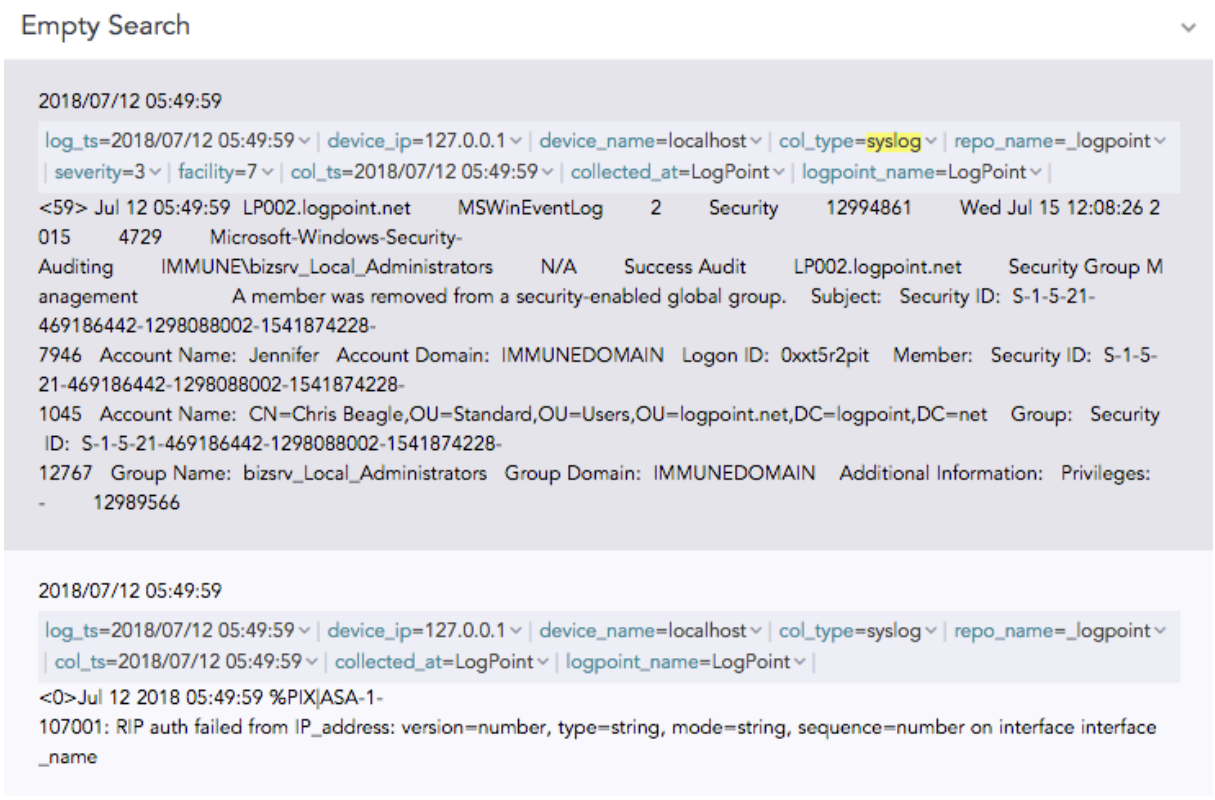


A blank query looks like this.

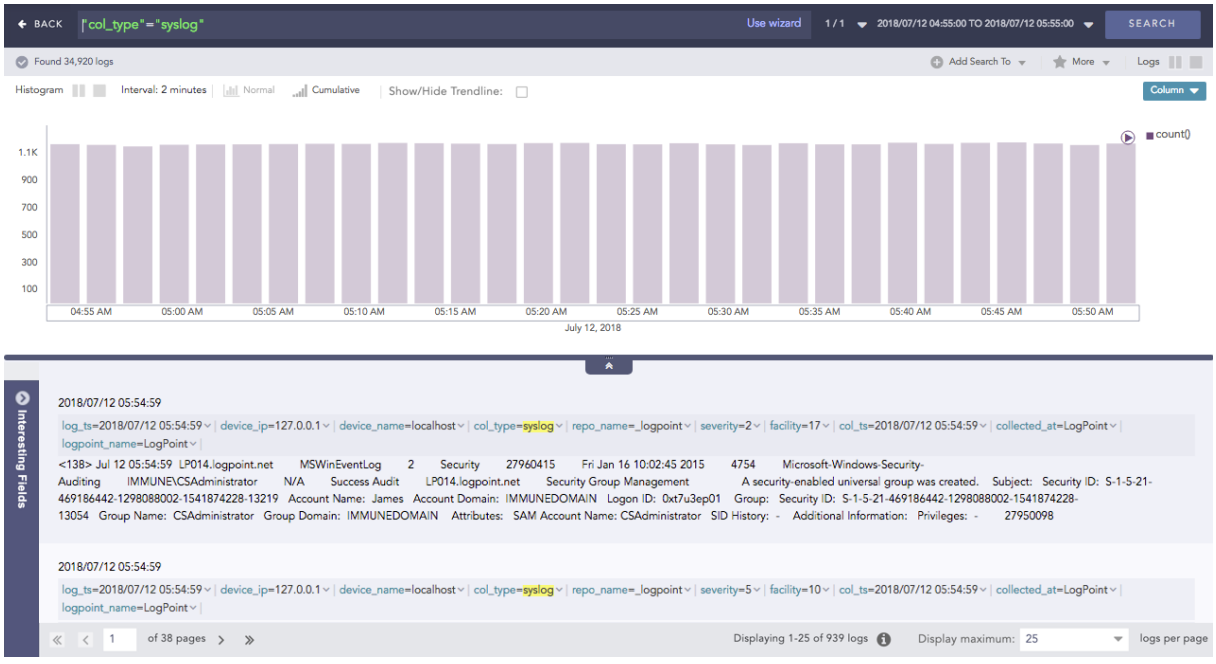


The results of a blank query are only the logs collected for the specified range of time, no graphs. You can refine the search query by clicking the on specific parts of the search results, for example key-value pair, or a raw log message. This starts a of search based on the selected parameter.

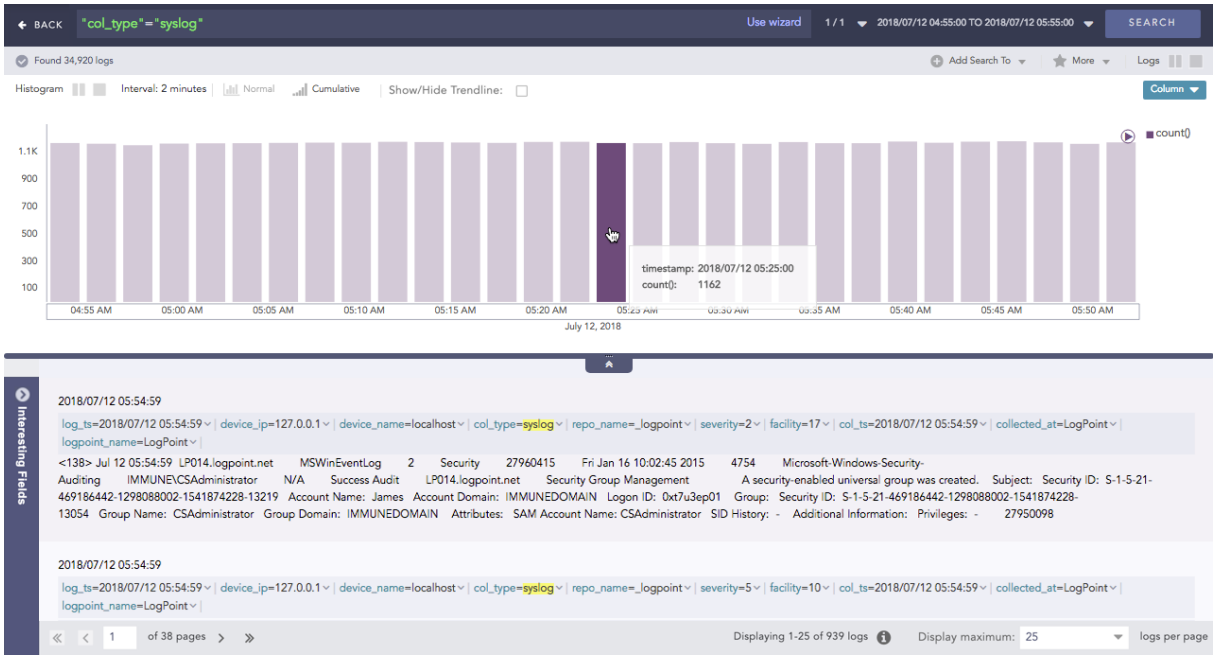
For example, if you click **syslog**:



This opens the search result of the query **"col_type"="syslog"**. The graph used depends on what you select.



Now you can drill down. When you hover a specific part of the search results, you can drill down to get more details.


BACK
"col_type"="syslog"
Use wizard
1 / 1
2018/07/12 04:55:00 TO 2018/07/12 05:55:00
SEARCH
Found 34,920 logs
Add Search To
More
Logs
Histogram
Interval: 2 minutes
Normal
Cumulative
Show/Hide Trendline:
Column
count()
1.1K
900
700
500
300
100
timestamp: 2018/07/12 05:25:00
count(): 1162
04:55 AM
05:00 AM
05:05 AM
05:10 AM
05:15 AM
05:20 AM
05:40 AM
05:45 AM
05:50 AM
July 12, 2018
Interesting Fields
2018/07/12 05:54:59
log_ts=2018/07/12 05:54:59 | device_ip=127.0.0.1 | device_name=localhost | col_type=syslog | repo_name=_logpoint | severity=2 | facility=17 | col_ts=2018/07/12 05:54:59 | collected_at=LogPoint | logpoint_name=LogPoint |
<138> Jul 12 05:54:59 LP014.logpoint.net MSWinEventLog 2 Security 27960415 Fri Jan 16 10:02:45 2015 4754 Microsoft-Windows-Security-Auditing IMMUNE\CSAdministrator N/A Success Audit LP014.logpoint.net Security Group Management A security-enabled universal group was created. Subject: Security ID: S-1-5-21-469186442-1298088002-1541874228-13219 Account Name: James Account Domain: IMMUNEDOMAIN Logon ID: 0xt7u3ep01 Group: Security ID: S-1-5-21-469186442-1298088002-1541874228-13054 Group Name: CSAdministrator Group Domain: IMMUNEDOMAIN Attributes: SAM Account Name: CSAdministrator SID History: - Additional Information: Privileges: - 27950098
2018/07/12 05:54:59
log_ts=2018/07/12 05:54:59 | device_ip=127.0.0.1 | device_name=localhost | col_type=syslog | repo_name=_logpoint | severity=5 | facility=10 | col_ts=2018/07/12 05:54:59 | collected_at=LogPoint | logpoint_name=LogPoint |
1
of 38 pages
Displaying 1-25 of 939 logs
Display maximum: 25
logs per page


Filter
Range: 2018/07/12 05:25:00 To 2018/07/12 05:27:00
count(): 1,162
View Logs
Drilldown by
col_type
Top 10 drilldown by
col_type

Overview

Overview shows the same data for a specific and non-adjustable period from one place. It includes multiple dashboards for different personas, including a SOC manager, SOC analyst and SIEM engineer. It brings together widgets from various sources, aggregates their data and lets you manage how you want to view it. You need SLS admin permission to view Overview.

To view Overview:

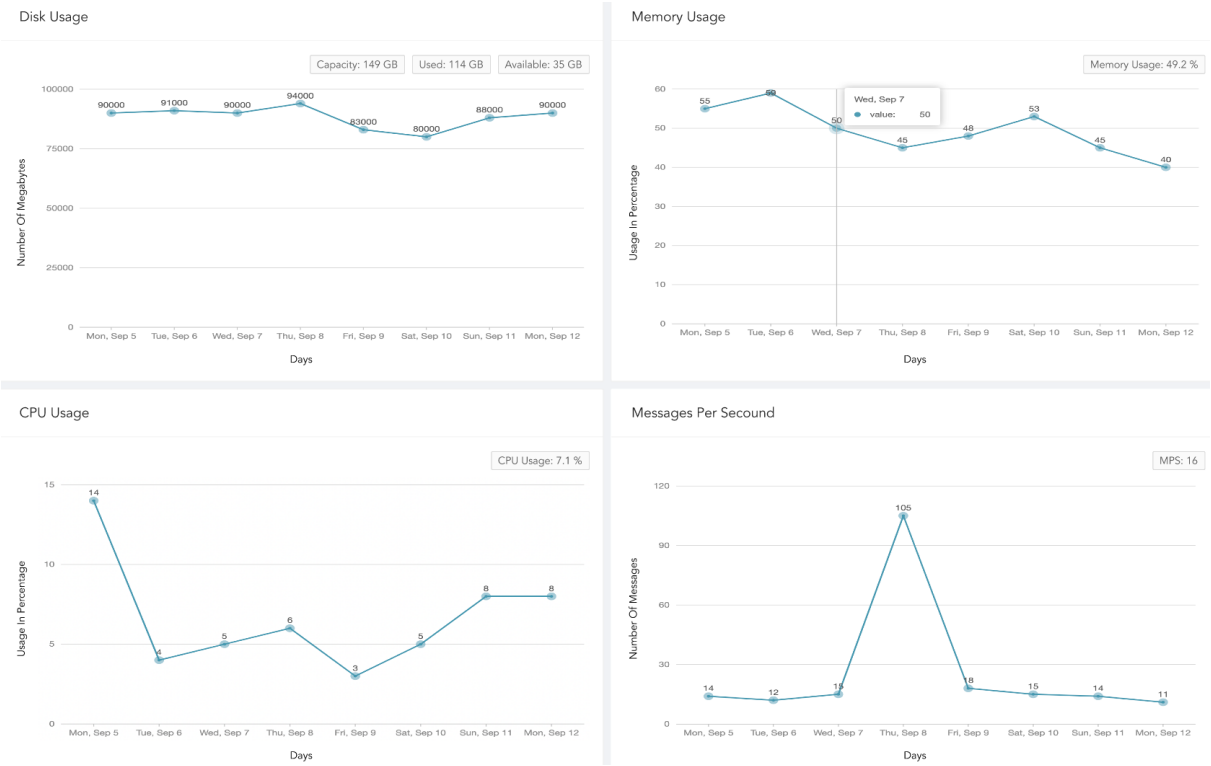
1. Click **Dashboard** from the navigation bar.
2. By default, **All Dashboards** opens. Click **Overview** on the right of **All Dashboards**.

System Health Dashboard

The System Health dashboard monitors system components such as disk usage, memory usage, CPU usage, and messages per second events, providing administrators with a high-level system health overview. These system events can help you identify unusual patterns or activities, understand whether the system is running efficiently, and detect potential threats, malware, or malicious events early so you can take corrective actions.

The dashboard's widgets are:

Widget Name	Description
Disk Usage	The total number of gigabytes SLS is using to run programs and carry out tasks daily in the specified period. Disk usage relates to hard disk performance.
Memory Usage	The trend of memory (RAM) capacity SLS uses while running processes or tasks in the specified period. This helps admin users understand system capacity and make sure there is enough memory.
CPU Usage	The total percentage of processing power in use so an admin user can check system performance, health and speed.
Messages Per Second	SLS's scalability and capacity to handle a large volume of messages within a second. It can help admin users identify peak message rates and assess capacity.



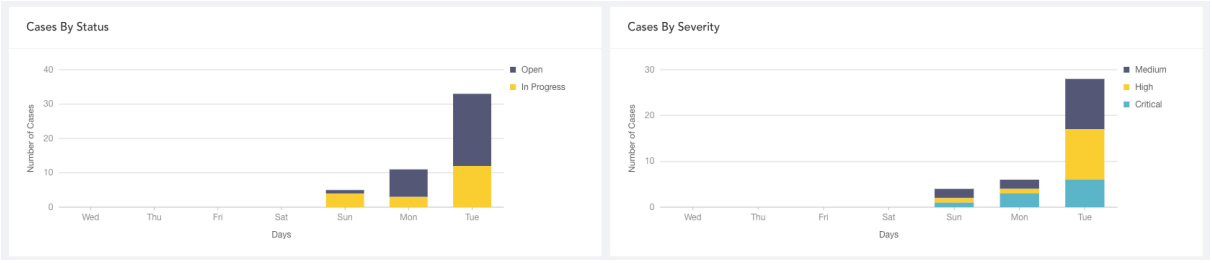
SOC Operation Dashboard

The SOC Operation dashboard is an overview of real-time cybersecurity incidents based on key measures, workflows, and behavioral patterns. The incident status/severity, cases status/severity and case response event data SLS provides is from daily activity during a specified period. You can use this dashboard to check SOC effectiveness and ensure all security operations including detections, analyses, and responses are running effectively.

The dashboard's widgets are:

Widget Name	Description
Incidents By Status	Unresolved and resolved incident trend's accumulated data collected each day over a specified period so SOC managers can use to find the number of changed incident states.
Incidents By Severity	The total number of accumulated incidents with severity (critical, high and medium) not closed daily in a specified period so a SOC manager can view risk trends associated with incidents and adjust the incident threshold.
Cases By Severity	The total number of accumulated cases with severity (critical, high and medium) not closed daily in a specified period so SOC managers can see how case severity has changed and help them prioritize case work.
Cases By Status	The accumulated data on open and in progress cases trends for each day in a specified period. SOC managers can view the proportion of cases whose status changed and evaluate the current risk level.
Automated Response vs Manual Response	The accumulated data of cases closed by playbooks (automated response) and cases closed by SOC analysts (manual response) monthly in the specified period to assess the case resolution reliability of the playbook so SOC managers can track the efficiency of automation.


Incidents By Status
Resolved
Unresolved
Number of Incidents
Days
Sat
Sun
Mon
Tue
Wed
Thu
Fri
Incidents By Severity
Medium
High
Critical


Cases By Status
Open
In Progress
Number of Cases
Days
Cases By Severity
Medium
High
Critical


Automated Response vs Manual Response
Manual
Automated
Number of Closed Cases in Percentage
Months
March
Manual: 52.67
Automated: 47.33

Visualization

Response Types in Visualization

Altogether, there are eight response types for the representation of search results in the visualization. Four of them are the regular response types, and the other four response types are the same responses grouped into time buckets for a given time-range.

Single Aggregation without Grouping

The Single Aggregation without Grouping response type is used for aggregation of an individual parameter concerning a given aggregation parameter.

The general syntax for the **Single Aggregation without Grouping** is:

```
| chart aggregation_parameter
```

This search query displays the value of the aggregation parameter over a specified range of time. The result of this response type can be represented in the form of :

Visualization Type	Sample Search Query
Display Chart	`| chart count()`
Gauge Chart	`| chart count()`

Single Aggregation with Grouping

The **Single Aggregation with Grouping** response type is used for aggregation of various grouping parameters concerning a given aggregation parameter. The general syntax for **Single Aggregation with Grouping** is:

```
| chart aggregation_parameter by grouping_parameter1, grouping_parameter2,
........, grouping_parametern
```

Example queries of Single Aggregation with Grouping are:

```
| chart count() by device_name
```

```
| chart sum(datasize) by action, protocol
```

```
| chart avg(datasize) by type, protocol, device_ip
```

The response type displays the value of the aggregation parameter, grouped by all the grouping parameter(s) in the specified time range. The result of this query can be represented in the form of :

Visualization Type	Sample Search Query
Display Chart	`| chart count() by attack_category`
Column Chart	`severity=* | chart count() by severity order by count() desc limit 5`
Line Chart	`severity=* | chart count() by severity`

Visualization Type	Sample Search Query
Donut Chart	`source_address=* | chart count() by source_address`
Area Chart	`action=* source_address=* | chart count() by action, source_address`
Bar Chart	`severity=* | chart count() by severity order by count() desc limit 5`
Heatmap Chart	`source_address=* action=* | chart count() by source_address, action order by count() desc limit 10`
Radar chart	`service=* action=* | chart count() by action, service`
TreeMap Chart	`source_address=* action=* | chart count() by source_address, action order by count() desc limit 10`
Parallel Coordinate Chart	`| process geoip(source_address) as source_country | chart count() by source_country, sub_category, destination_location`
Sankey chart	`| process geoip(source_address) as country | chart count() by country, severity, category, sub_category`
World Map Chart	`| process geoip(destination_address) as country_name | chart count(), avg(datasize) by country_name, action`
ATT&CK chart	`| chart count() by attack_id`

General Operations for Single Aggregation with Grouping

This section contains the general operations applicable to all the charts belonging to the **Single Aggregation with Grouping** response type.

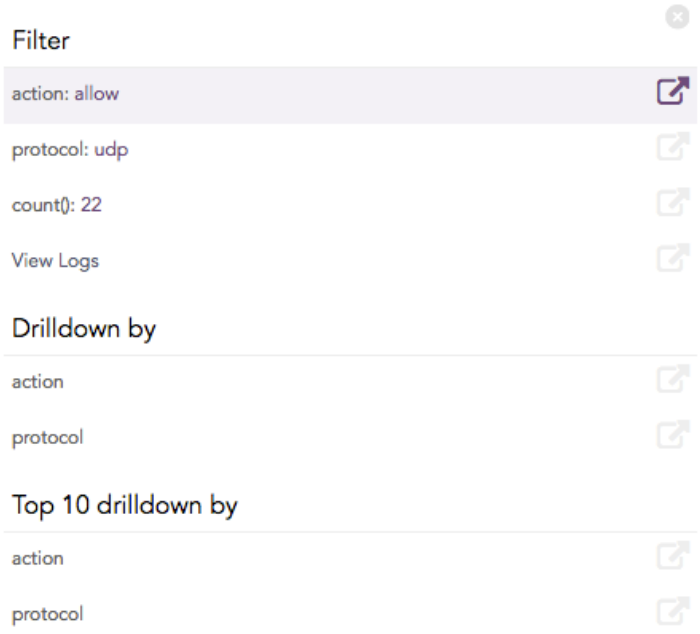
> **NOTE**
> Some charts might consist of operations that are relevant to the specific chart only. In that case, refer to the article of the particular chart.

Drill-down

In the Single Aggregation with Grouping response type, you can perform the drill-down specific value of the grouping or aggregation parameter.

When you hover over a component of a graph (including but not limited node, line, bar, point) a tooltip appears. The tooltip displays all the relevant information about the particular component.

Click the component to open a new drill-down window. The window summarizes the information of the selected node along with the option to drill down as per your preference.



Click the corresponding **Open in a new window** icon to further drill down on any field. Additionally, you can view the search results for the selected set of data by clicking **View Logs** in the same window.

Multiple Aggregation without Grouping

The Multiple Aggregation without Grouping response type is used for aggregation of multiple aggregation parameters for all the available logs or the given repo and time range. An example of a search query for such response is:

```
| chart count(), avg(datasize)
```

This query displays the total count and the average of datasize of the logs collected in the specified range of time. The result of this query can be represented in the form of :

Visualization Type	Sample Search Query
Clustered Column Chart	`| chart max(sent_datasize), max(received_datasize)`
Clustered Bar Chart	`| chart avg(sent_datasize), avg(received_datasize)`
Display Chart	`| chart count(), max(datasize), avg(datasize)`

General Operations for Multiple Aggregation without Grouping

Interactive Legend

In **Multiple Aggregation without Grouping**, when a chart is rendered, all selected aggregation parameters are displayed with a unique color for each aggregation parameter. However, you can choose to view the graphs concerning a specific aggregation parameter(s).

To hide an aggregation parameter, click the name of the parameter on the legend at the extreme right side of the container.

When you click a name of a parameter on the legend, the section referring to the respective parameter disappears, and a new chart is rendered consisting of all other aggregation parameters. You can unhide the parameter by clicking the legend again.

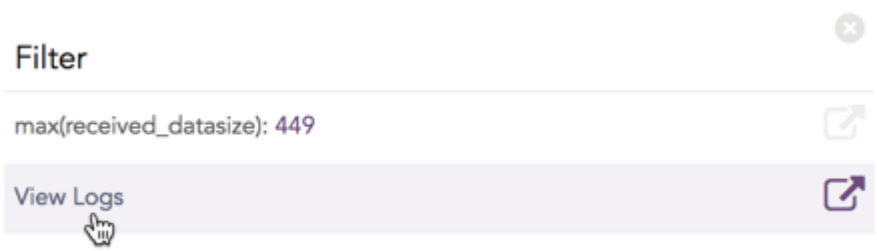
NOTE
The scale on the y-axis is auto-adjusted as per the value of the remaining aggregation parameter (s).

Drill down

Through the drill-down feature, you can choose to retrieve detailed results about a specific section of a chart. In Multiple Aggregation without Grouping response type, you can drill-down search operation over a specific value of aggregation parameter.

Hover over a component of a graph (example: node, line, bar, point) to view the tooltip. The tooltip displays all the relevant information about the particular component.

Click the segment to open a drill-down window. The window summarizes the related information of the selected section along with the option to drill down as per your preference.



Click the corresponding **Open in a new window** icon to further drill-down the search result from any field. Additionally, click the **View Logs** to view the search result for the selected set of data.

NOTE
The drill-down feature is not applicable for the Display format of Multiple Aggregations without Grouping response type.

Multiple Aggregation with Grouping

The Multiple Aggregation with Grouping response type is used for aggregation of grouping parameters concerning given multiple aggregation parameters.

The general syntax for **Multiple Aggregation with Grouping** is:

```
| chart aggregation_parameter1, aggregation_parameter2 by grouping_
parameter1, grouping_parameter2, ..., grouping_parametern
```

Example queries of Multiple Aggregation with Grouping type are:

```
| chart count(), avg(datasize) by action
```

```
user=* | chart count(label=Fail) as Failed, count(label=Successful) as
Successful by user order by Failed desc limit 10
```

This query displays the count and average datasize of the collected logs in the specified time range grouped by the actions applied. The result of this query can be represented in the form of :

Visualization Type	Sample Search Query
Clustered Column Chart	`action=Allow or action=Deny | chart count(action=allow) as AllowedConnection, count(action=deny) as DeniedConnection by source_address order by count(action=allow), count (action=deny) desc limit 10`
Clustered Bar Chart	`action=Allow or action=Deny | chart count(action=allow) as AllowedConnection, count(action=deny) as DeniedConnection by source_address order by count(action=allow), count (action=deny) desc limit 10`

Visualization Type	Sample Search Query
Clustered Line Chart	`sent_datasize=* source_address=* | chart max(sent_datasize), max(received_datasize) by source_address order by max(sent_datasize), max(received_datasize) desc limit 10`
Stacked Area Chart	`sent_datasize=* source_address=* | chart max(sent_datasize), max(received_datasize) by source_address order by max(sent_datasize), max(received_datasize) desc limit 10`
Radar chart	`"norm_id"="WinDNSDHCP"| chart count(lease_address=end), count(lease_address=start) by user`
World Map Chart	`| process geoip(destination_address) as country_name | chart count(), avg(datasize) by country_name, action`
Bubble Chart	`| chart count(), max(sig_id) by action`

General Operations for Multiple Aggregation with Grouping

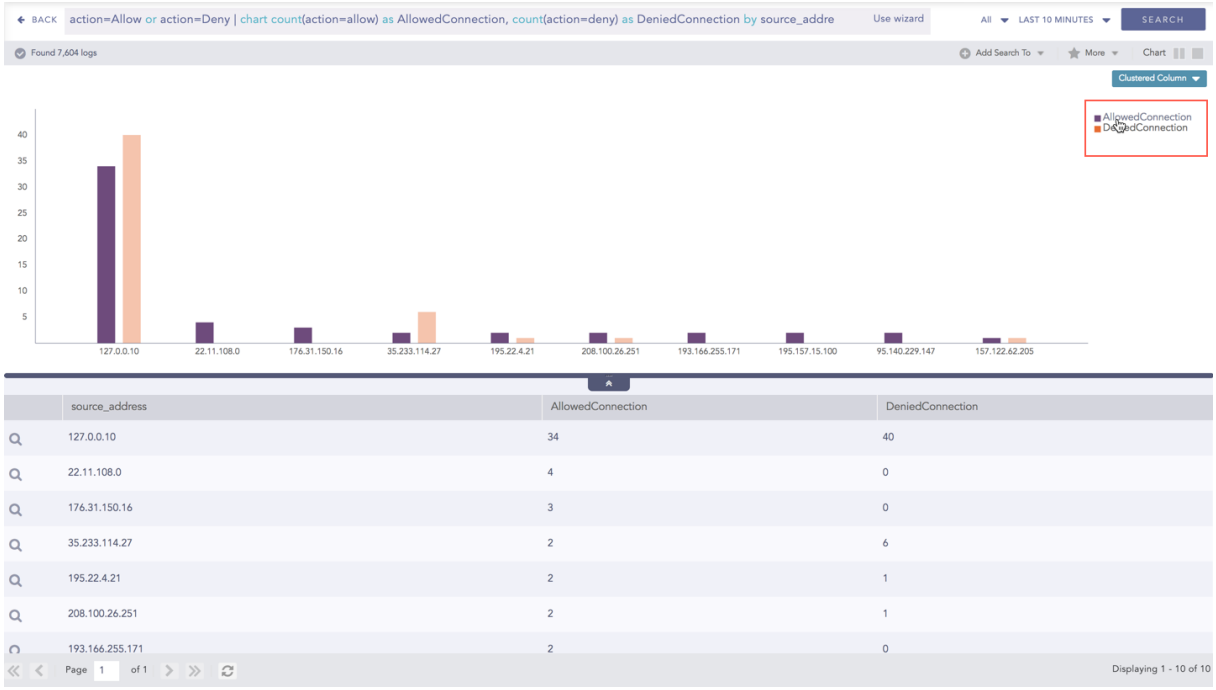
This section contains the general operations that can be applied to all the charts belonging to the Multiple Aggregation with Grouping response type.

NOTE
Some charts might consist of operations that are relevant to the specific chart only. For such operations, refer to the section of the particular chart.

Interactive Legend

In the Multiple Aggregation with Grouping response type, when a chart is rendered, values of all the selected aggregation parameters are displayed with a unique color for each aggregation parameter. However, you can choose to view the graphs concerning specific aggregation parameter(s).

To hide an aggregation parameter, click the name of the parameter on the legend at the extreme right side of the container.



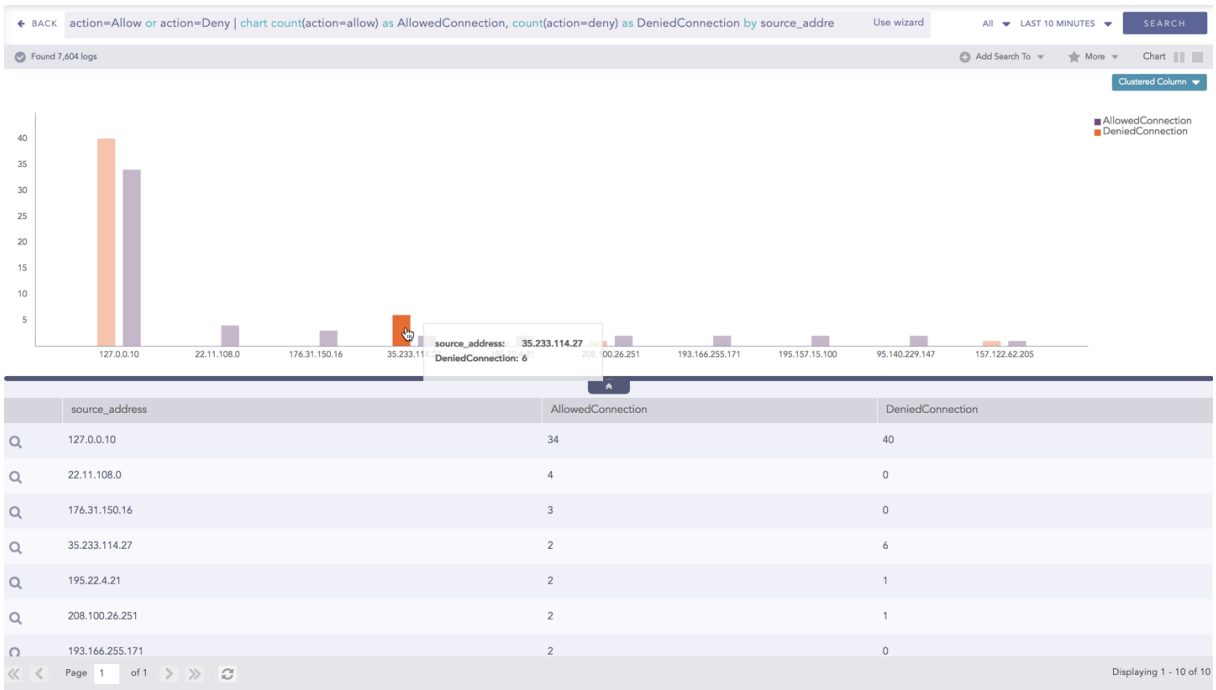
Click the name of a parameter on the legend, to hide its respective section. A new chart is rendered consisting of all other aggregation parameters. You can unhide the parameter by clicking the legend again.

NOTE
The scale on the y-axis is auto-adjusted as per value of the remaining aggregation parameter(s).

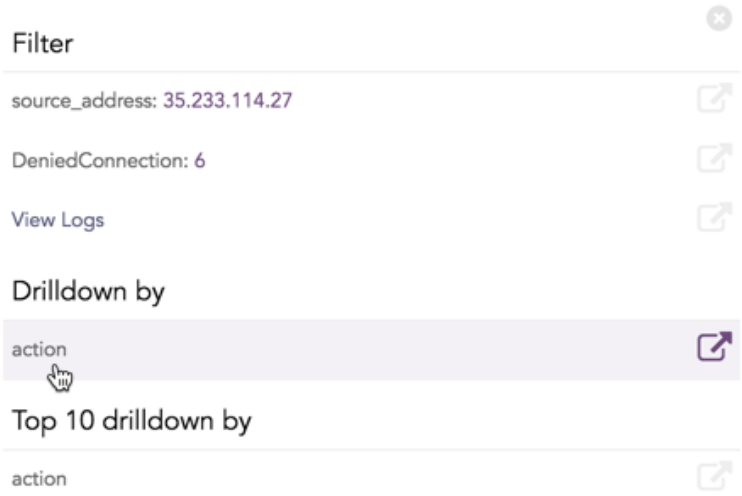
Drill-down

In the Multiple Aggregation with Grouping response type, you can drill-down search operation regarding a specific value of the grouping parameter concerning a single or multiple aggregation parameters.

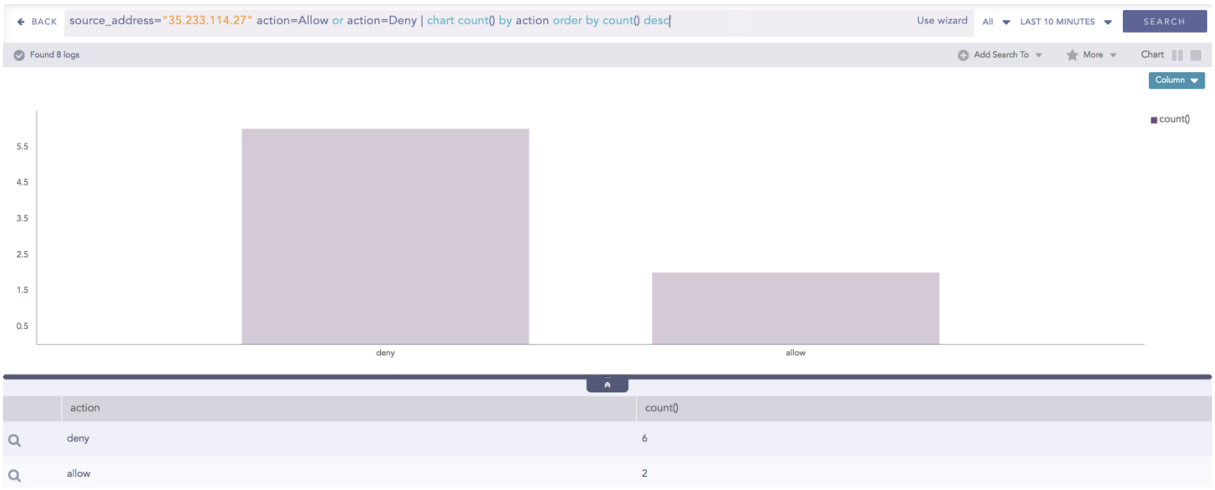
Hover over a component of a graph (example: node, line, bar, point e.t.c) to view a tooltip. The tooltip displays all the relevant information about the particular component.



Click the segment to open a drill-down window. The window summarizes the related information of the selected section along with the option to drill down as per your preference.



Click the corresponding **Open in a new window** icon to further drill-down the search result from any field. Additionally, click the **View Logs** to view the search result for the selected set of data.



Timechart Single Aggregation without Grouping

The Timechart Single Aggregation without Grouping response type is used for aggregation of processed logs to a given aggregation parameter grouped into time buckets (as a time series data) over a specified time range.

The general syntax for the **Timechart Single Aggregation without Grouping** is:

```
| timechart aggregation_parameter
```

Example queries of the Timechart Single Aggregation without Grouping type are:

```
| timechart count()
```

```
| timechart sum(datasize)
```

```
| timechart avg(datasize)
```

This response type displays the value of the aggregation parameter in the specified range of time. The charts that are used to visualize the queries belonging to this response type are : **Column**, **Line**, **Area**, **Day/Hour Heatmap**, and **Radar**

Visualization Type	Sample Search Query
Column Chart	`| timechart avg(datasize)`
Line Chart	`| timechart avg(datasize)`
Area Chart	`| timechart sum(datasize)`
Radar chart	`"norm_id"="WinDNSDHCP"| timechart count(lease_address=drop)`
Day/Hour Heatmap Chart	`| timechart sum(datasize) as TotalDatasize every 1 hour`

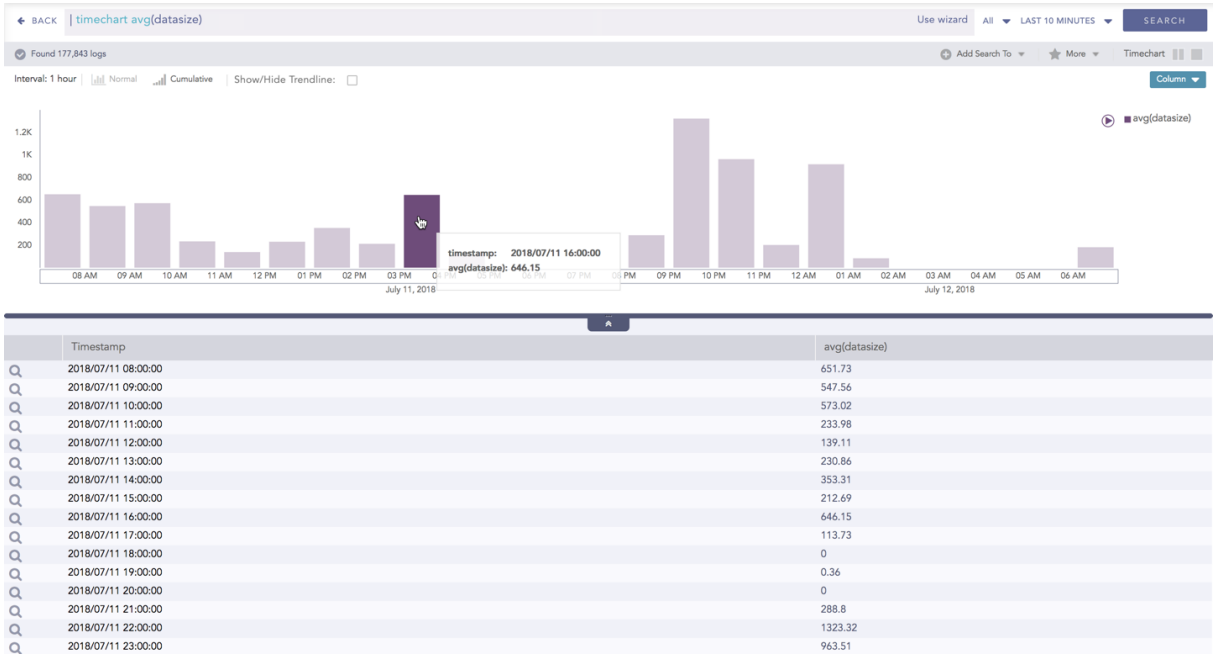
Additionally, the **Cumulative** chart option is also available along with the **Normal** chart for the **Column**, **Line**, and **Area** charts. The **Cumulative** option visualizes the results by accumulating data from the starting point to the current time-bucket for all time-buckets whereas the normal option visualizes the results as obtained from the query.

General Operations for Timechart Single Aggregation without Grouping

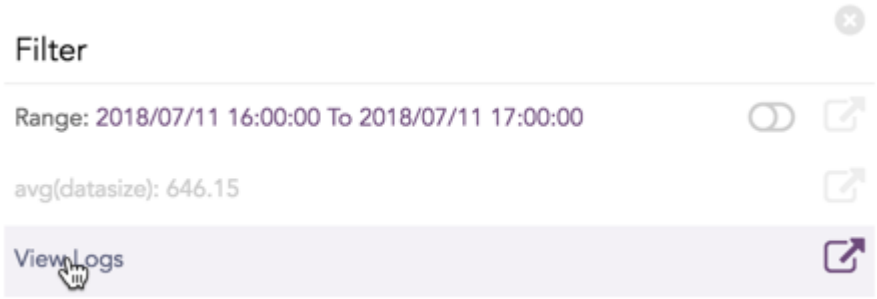
Drill-down

You can choose to view a detailed search for the response type regarding a specific value in two ways, i.e., from the line, or using a drag box.

Hover over a specific component/area of a chart to view a tool-tip. The tooltip displays all the information about the particular node.

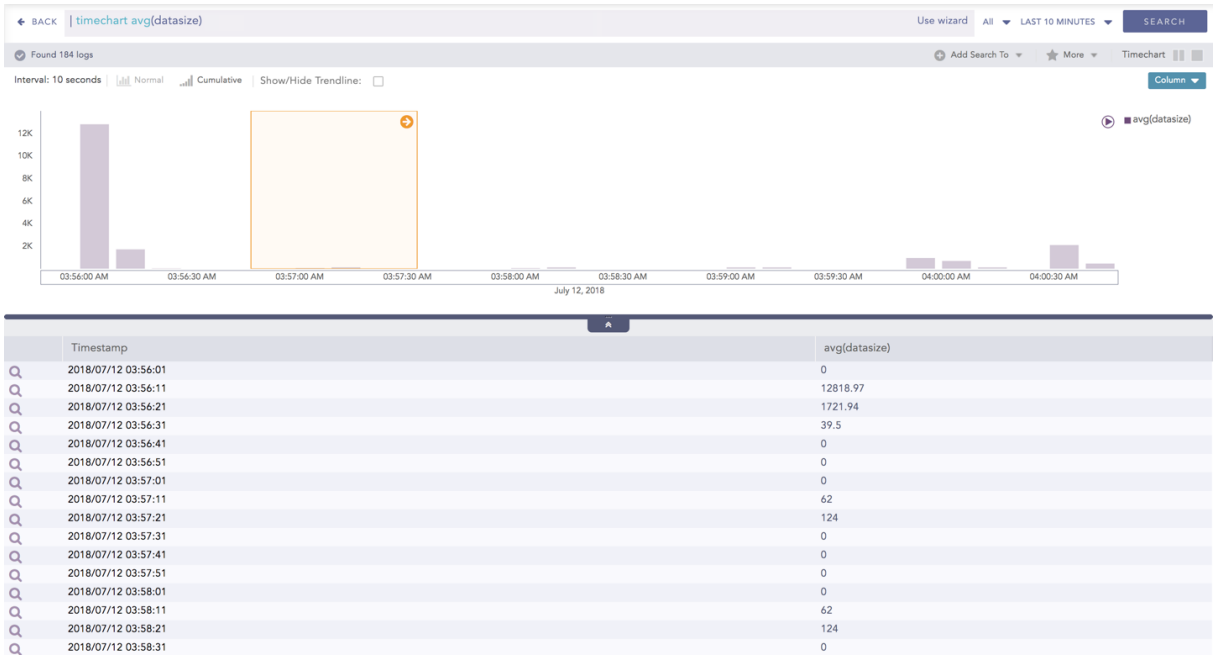


Click the component to open a drill-down window. The window summarizes the related information of the selected section along with the option to drill down as per your preference.



Click the corresponding **Open in a new window** icon to further drill-down the search result from any field. Additionally, click the **View Logs** to view the search result for the selected set of data.

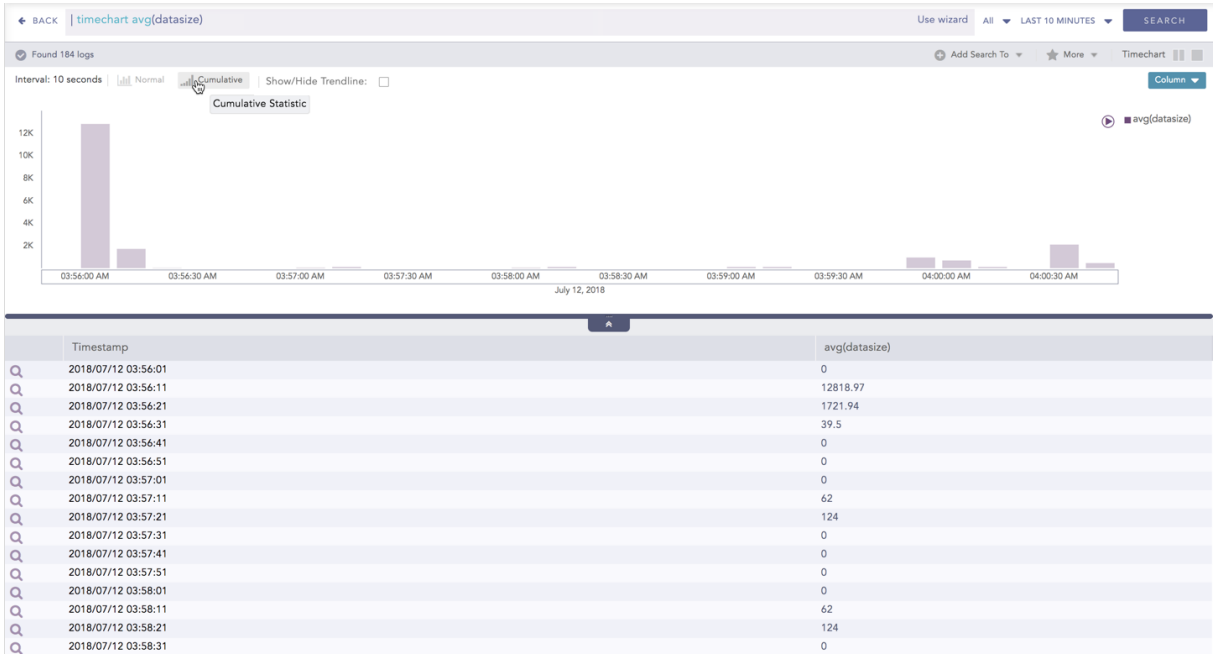
In addition to that, you can also drill down any chart of the response type using the drag box. Click and drag the mouse inside the graph, a yellow colored transparent drag box appears. You can drill-down the selected section of the chart by clicking the drill-down icon on the top-right corner of the box. You can resize or move the drag box as per your requirement.



Timestamp	avg(datasize)
2018/07/12 03:56:01	0
2018/07/12 03:56:11	12818.97
2018/07/12 03:56:21	1721.94
2018/07/12 03:56:31	39.5
2018/07/12 03:56:41	0
2018/07/12 03:56:51	0
2018/07/12 03:57:01	0
2018/07/12 03:57:11	62
2018/07/12 03:57:21	124
2018/07/12 03:57:31	0
2018/07/12 03:57:41	0
2018/07/12 03:57:51	0
2018/07/12 03:58:01	0
2018/07/12 03:58:11	62
2018/07/12 03:58:21	124
2018/07/12 03:58:31	0

Cumulative chart

The Cumulative chart displays the accumulated data values throughout the given time range. To view the cumulative chart, click **Cumulative** on the left side of the container of a chart.



Timestamp	avg(datasize)
2018/07/12 03:56:01	0
2018/07/12 03:56:11	12818.97
2018/07/12 03:56:21	1721.94
2018/07/12 03:56:31	39.5
2018/07/12 03:56:41	0
2018/07/12 03:56:51	0
2018/07/12 03:57:01	0
2018/07/12 03:57:11	62
2018/07/12 03:57:21	124
2018/07/12 03:57:31	0
2018/07/12 03:57:41	0
2018/07/12 03:57:51	0
2018/07/12 03:58:01	0
2018/07/12 03:58:11	62
2018/07/12 03:58:21	124
2018/07/12 03:58:31	0

Click **Normal** to view the regular chart.

Trendline

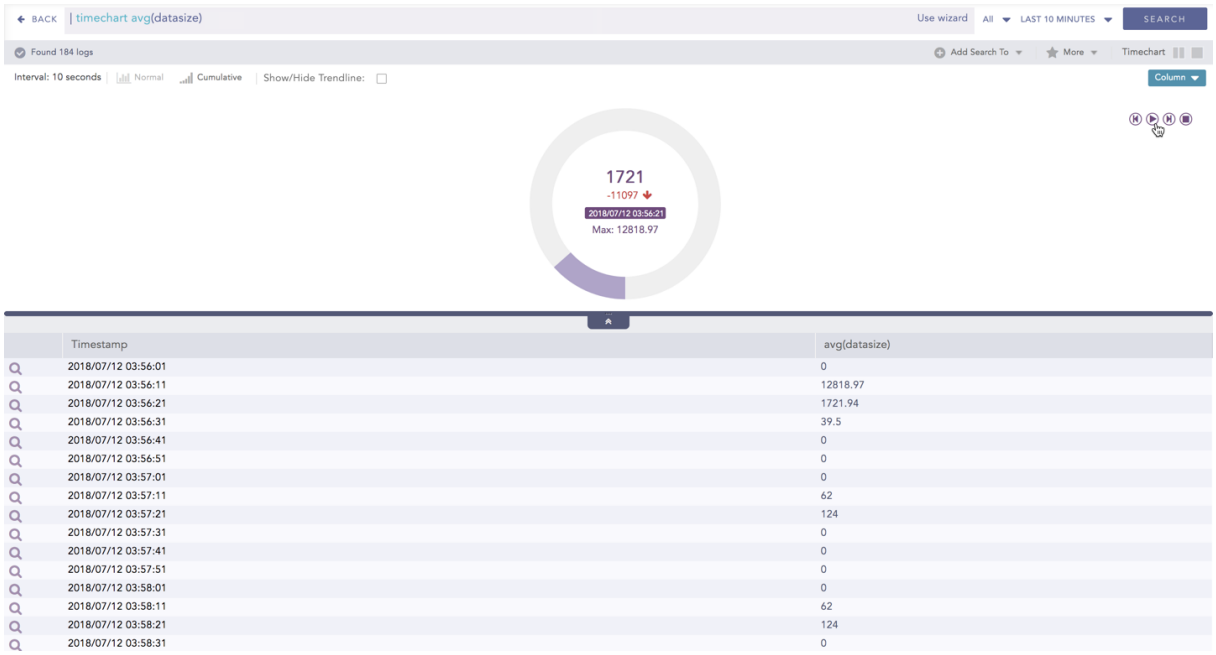
You can select the **Show/Hide Trendline** checkbox to identify whether the time-series data is likely to increase, decrease, or remain constant over a time period. The data on an increasing trend forms an upsloping line, whereas, on a decreasing trend, it forms a downsloping line. The **Show/Hide Trendline** checkbox is available for **Column**, **Line**, and **Area** charts of this response type only.

> **NOTE**
> The **Show/Hide Trendline** checkbox is also available for **Column**, **Line**, and **Area** charts resulted from Simple search queries and a blank search query.

Interactive Animation

The charts belonging to the **Timechart** response type include an interactive play button. The button allows you to slide through values of the charts concerning time buckets known as **Interval**.

Click **Play** on the right side of the container to start the animation. The graph is refreshed every four seconds, i.e., that graph shifts from one time-bucket to another time bucket every four seconds. Value of the time bucket is dependent upon the time-range specified in the **Interval**.



You can also click **Pause**, **Stop**, **Previous**, **Next**, **Replay** as required.

> **NOTE**
> The operations **Cumulative chart** and **Interactive Animation** are not available for the Radar chart.

Timechart Single Aggregation with Grouping

The Timechart Single Aggregation with Grouping response type is used for aggregation of processed logs by an individual grouping parameter concerning given a single aggregation parameter grouped into time buckets (as a time series data) over a specified time range.

The general syntax for **Timechart Single Aggregation with Grouping** is:

```
| timechart aggregation_parameter by grouping_parameter1, grouping_
parameter2, ....., grouping_parametern
```

Example queries of Timechart Single Aggregation with Grouping type are:

```
| timechart count() by action
```

This query displays the count of the logs generated by the individual action, for an individual time bucket over a specified range of time. The result of this query can be represented in the form of :

Visualization Type	Sample Search Query
Clustered Line Chart	`event_category=* | timechart count() by event_category`
Stacked Column Chart	`source_address=* | timechart count() by source_address`

General Operations for Timechart Single Aggregation with Grouping

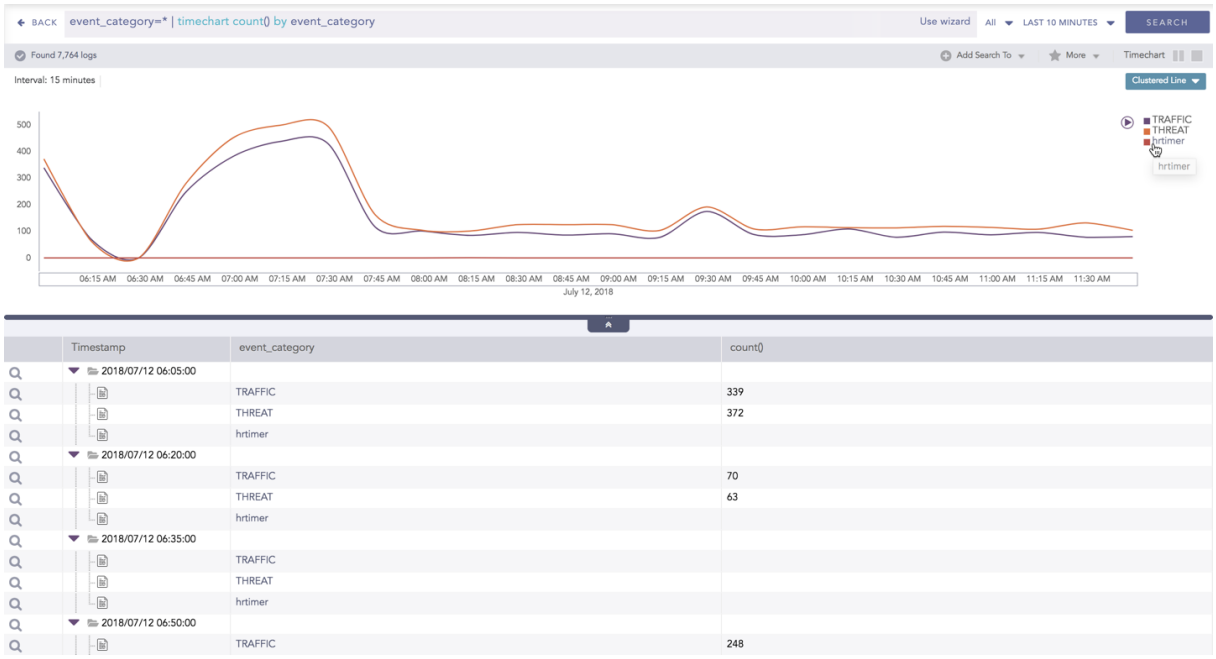
This section contains the general operations that can be applied to all the charts belonging to the Timechart Single Aggregation with Grouping response type.

> **NOTE**
> Some charts might consist of operations that are relevant to the specific chart only. In this case, refer to the section of the particular chart.

Interactive Legend

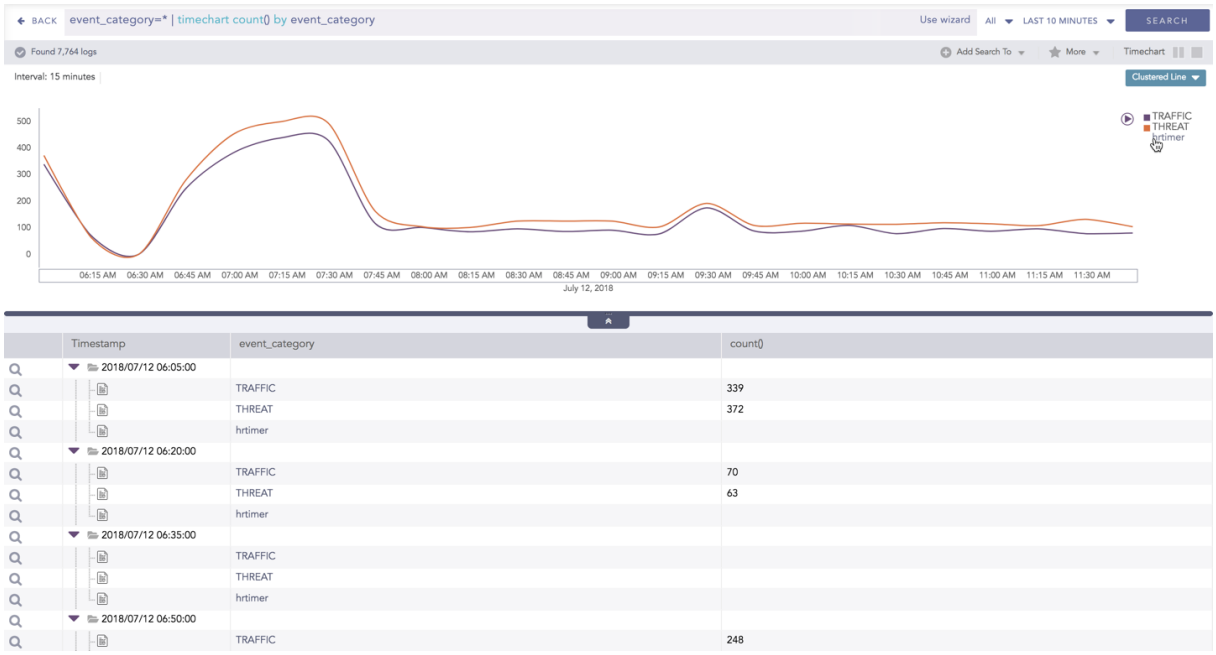
In the Timechart Single Aggregation with Grouping response type, when a chart is rendered, all the aggregation values of the selected grouping parameter(s) are displayed with a unique color for each value of the grouping parameter(s). However, you can choose to view the graphs concerning a specific value of grouping parameter(s).

To hide the value of a grouping parameter, click the name of the parameter on the legend at the extreme right side of the container.



Timestamp	event_category	count()
2018/07/12 06:05:00		
	TRAFFIC	339
	THREAT	372
	hrtimer	
2018/07/12 06:20:00		
	TRAFFIC	70
	THREAT	63
	hrtimer	
2018/07/12 06:35:00		
	TRAFFIC	
	THREAT	
	hrtimer	
2018/07/12 06:50:00		
	TRAFFIC	248

When you click a name of a parameter on the legend, the section (line, bar) referring to the respective parameter disappears, and a new chart is rendered consisting all other values of the grouping parameter(s). Click the legend again to unhide the particular value.



Timestamp	event_category	count()
2018/07/12 06:05:00		
	TRAFFIC	339
	THREAT	372
	hrtimer	
2018/07/12 06:20:00		
	TRAFFIC	70
	THREAT	63
	hrtimer	
2018/07/12 06:35:00		
	TRAFFIC	
	THREAT	
	hrtimer	
2018/07/12 06:50:00		
	TRAFFIC	248

NOTE

The scale on the y-axis is auto-adjusted as per the value of the remaining values of grouping parameter(s).

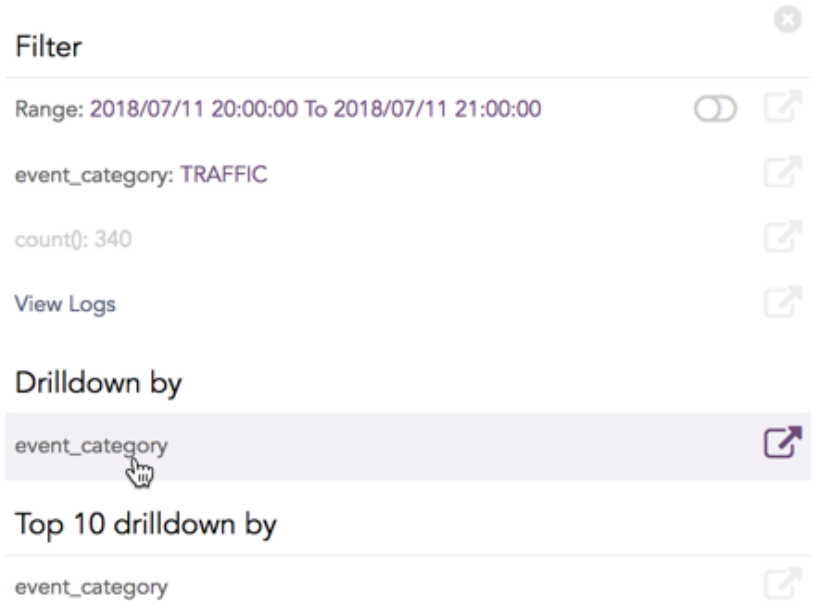
Drill-down

You can choose to view a detailed search for the response type regarding a specific value in two ways, i.e., from the line, or using a drag box.

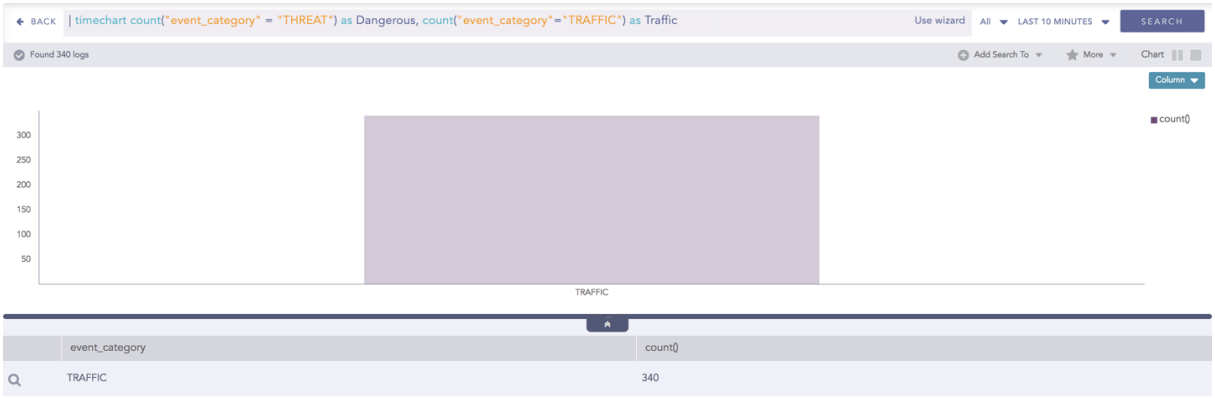
Hover over a specific component/area of a chart to view a tool-tip. The tooltip displays all the information about the particular node.

Timestamp	event_category	count()
2018/07/11 11:00:00		
	TRAFFIC	1,044
	THREAT	984
	RPC	
	hrtimer	
2018/07/11 12:00:00		
	TRAFFIC	178
	THREAT	212
	RPC	
	hrtimer	

Click the component to open a drill-down window. The window summarizes the related information of the selected section along with the option to drill down as per your preference.



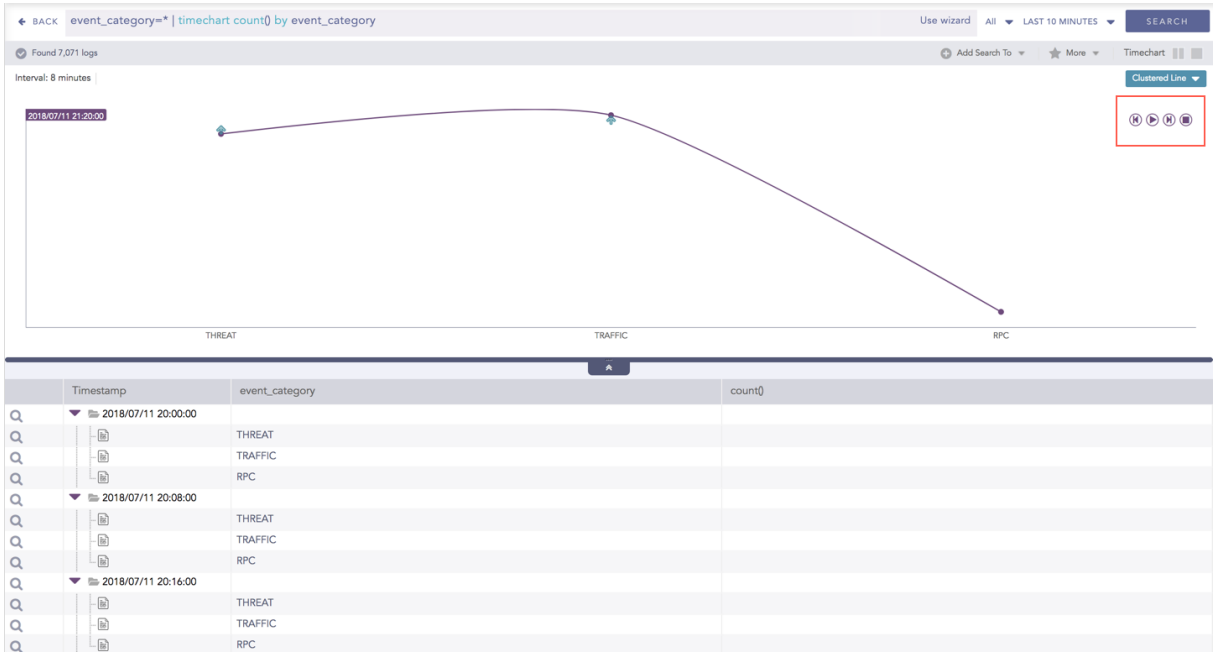
Click the corresponding **Open in a new window** icon to further drill-down the search result from any field. Additionally, click the **View Logs** to view the search result for the selected set of data.



In addition to that, you can also drill-down any chart of the response type using the drag box. Click and drag the mouse inside the graph, a yellow colored transparent drag box appears. You can drill-down the selected section of the chart by clicking the drill-down icon on the top-right corner of the box. You can resize or move the drag box as per your requirement.

Interactive Animation

The charts belonging to the **Timechart** response type include an interactive Play button. It allows you to slide through values of the charts concerning time buckets known as **Interval**.



Click the **Play** on the right side of the container to start the animation. The graph is refreshed every four seconds, i.e., that graph shifts from one time-bucket to another time bucket every four seconds. Value of the time bucket is dependent upon the time-range specified in the **Interval**.

You can also click **Pause**, **Stop**, **Previous**, **Next**, **Replay** as required.

Timechart Multiple Aggregation without Grouping

The Timechart Multiple Aggregation without Grouping response type is used for aggregation of processed logs related to the given parameters. The logs are grouped into time buckets (as a

time series data) over a specified time-range.

The general syntax for **Timechart Multiple Aggregation without Grouping** is:

```
| timechart aggregation_parameter1, aggregation_parameter2, ....
aggregation_parametern
```

Example queries of Timechart Multiple Aggregation without Grouping type are:

```
| timechart count(), avg(datasize)
```

This query displays the count of total logs generated and the average datasize of collected logs for individual time bucket over a specified range of time. The result of this query can be represented in the form of :**Clustered Column**, **Clustered Line**, **Radar**, and **Stacked Area** charts.

Visualization Type	Sample Search Query
Clustered Column Chart	`norm_id=WinDNSDHCP | timechart count(lease_address=drop) as Dropped, count(lease_address=start) as Started,count(lease_address=end) as ENDED`
Clustered Line Chart	`| timechart count("event_category" = "THREAT") as Dangerous, count("event_category" ="TRAFFIC") as Traffic`
Radar chart	`norm_id=WinDNSDHCP | timechart count(lease_address=drop) as Dropped, count(lease_address=start) as Started, count(lease_address=end) as ENDED`
Stacked Area Chart	`sent_datasize=* source_address=* | chart max(sent_datasize), max(received_datasize) by source_address order by max(sent_datasize), max(received_datasize) desc limit 10`

General Operations of Timechart Multiple Aggregation without Grouping

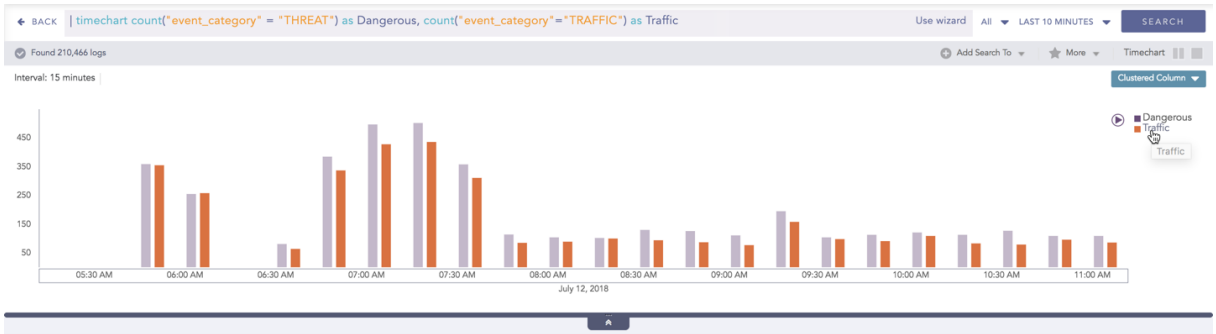
This section contains the general operations that can be applied to all the charts belonging to the Timechart Multiple Aggregation without Grouping response type.

NOTE
Some charts might consist of operations that are relevant to the specific chart only. For such operations, refer to the section of the particular chart.

Interactive Legend

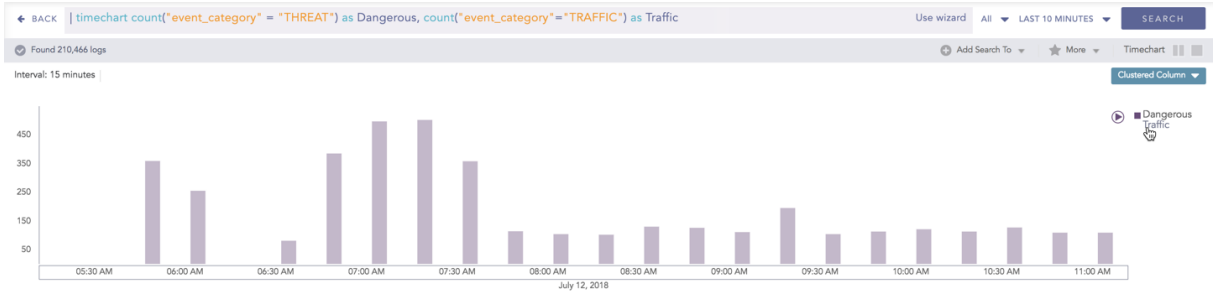
In the Timechart Multiple Aggregation without Grouping response type, when a chart is rendered, all the values of the selected aggregation parameter(s) are displayed with a unique color for each value of the aggregation parameter(s). However, you can view the graphs for specific aggregation parameter(s).

To hide an aggregation parameter, click the name of the parameter on the legend at the extreme right side of the container.



Timestamp	Dangerous	Traffic
2018/07/12 05:26:58	0	0
2018/07/12 05:41:58	0	0
2018/07/12 05:56:58	359	355
2018/07/12 06:11:58	255	258
2018/07/12 06:26:58	0	0
2018/07/12 06:41:58	81	64
2018/07/12 06:56:58	385	337
2018/07/12 07:11:58	497	428
2018/07/12 07:26:58	502	436
2018/07/12 07:41:58	358	311
2018/07/12 07:56:58	114	85
2018/07/12 08:11:58	104	89
2018/07/12 08:26:58	102	100
2018/07/12 08:41:58	130	94
2018/07/12 08:56:58	126	87
2018/07/12 09:11:58	111	77

When you click a name of a parameter on the legend, the section (line, column, bar) referring to the respective parameter disappears, and a new chart is rendered consisting all other aggregation parameters (s). Click the legend again to unhide the value.



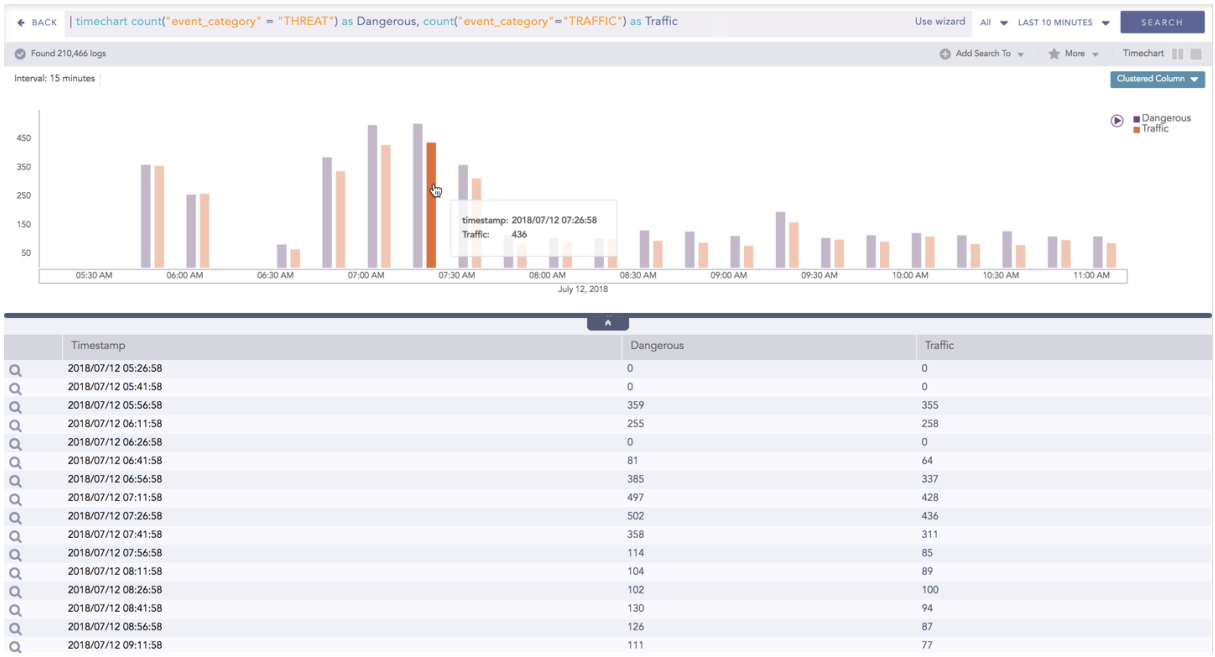
Timestamp	Dangerous	Traffic
2018/07/12 05:26:58	0	0
2018/07/12 05:41:58	0	0
2018/07/12 05:56:58	359	355
2018/07/12 06:11:58	255	258
2018/07/12 06:26:58	0	0
2018/07/12 06:41:58	81	64
2018/07/12 06:56:58	385	337
2018/07/12 07:11:58	497	428
2018/07/12 07:26:58	502	436
2018/07/12 07:41:58	358	311
2018/07/12 07:56:58	114	85
2018/07/12 08:11:58	104	89
2018/07/12 08:26:58	102	100
2018/07/12 08:41:58	130	94
2018/07/12 08:56:58	126	87
2018/07/12 09:11:58	111	77

NOTE
The scale on the y-axis is auto-adjusted as per the value of the remaining aggregation parameter (s).

Drill-down

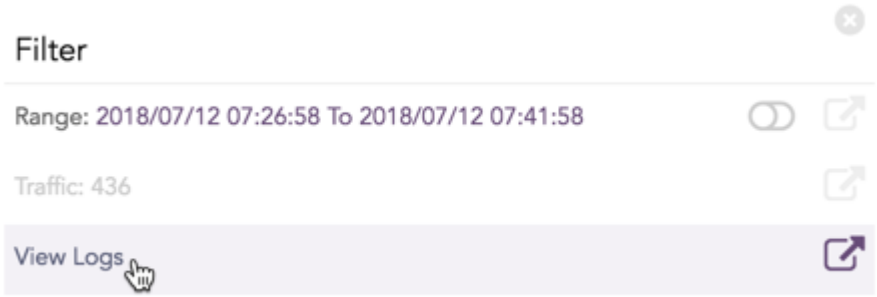
You can choose to view a detailed search for the response type regarding a specific value in two ways, i.e., from the line, or using a drag box.

Hover over a specific component/area of a chart to view a tool-tip. The tooltip displays all the information about the particular node.



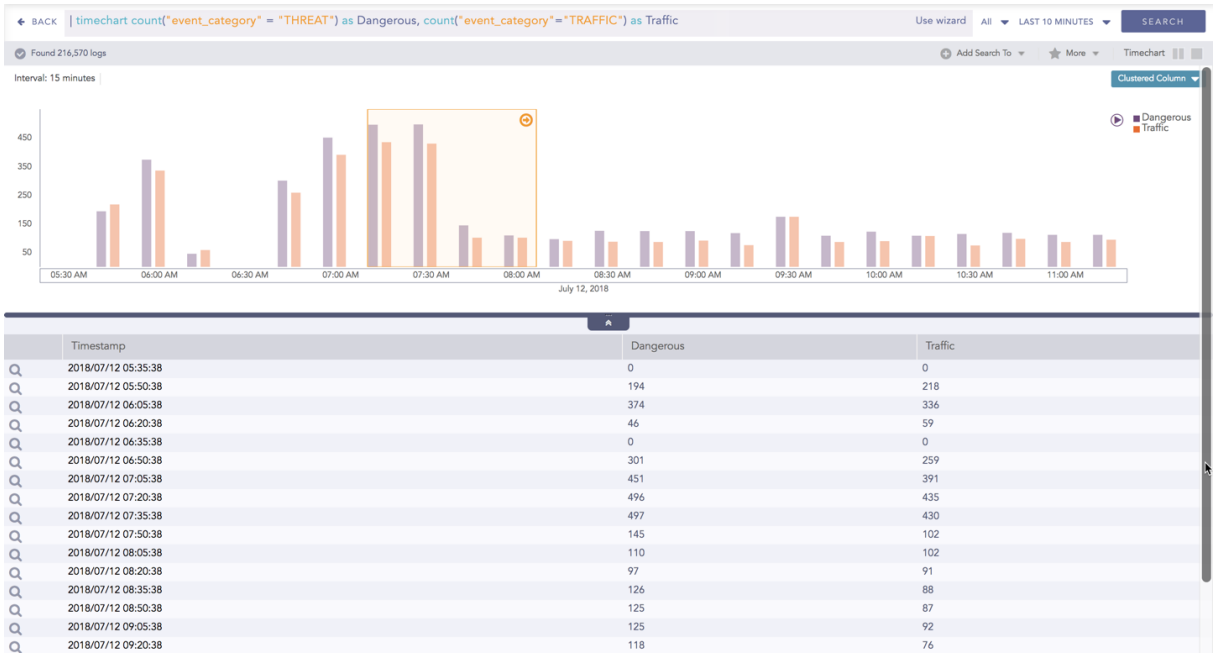
Timestamp	Dangerous	Traffic
2018/07/12 05:26:58	0	0
2018/07/12 05:41:58	0	0
2018/07/12 05:56:58	359	355
2018/07/12 06:11:58	255	258
2018/07/12 06:26:58	0	0
2018/07/12 06:41:58	81	64
2018/07/12 06:56:58	385	337
2018/07/12 07:11:58	497	428
2018/07/12 07:26:58	502	436
2018/07/12 07:41:58	358	311
2018/07/12 07:56:58	114	85
2018/07/12 08:11:58	104	89
2018/07/12 08:26:58	102	100
2018/07/12 08:41:58	130	94
2018/07/12 08:56:58	126	87
2018/07/12 09:11:58	111	77

Click the component to open a drill-down window. The window summarizes the related information of the selected section along with the option to drill down as per your preference.



Click the corresponding **Open in a new window** icon to further drill-down the search result from any field. Additionally, click the **View Logs** to view the search result for the selected set of data.

In addition to that, you can also drill-down any chart of the response type using the drag box. Click and drag the mouse inside the graph, a yellow colored transparent drag box appears. You can drill-down the selected section of the chart by clicking the drill-down icon at the top-right corner of the box. You can resize or move the drag box as per your requirement.



Timestamp	Dangerous	Traffic
2018/07/12 05:35:38	0	0
2018/07/12 05:50:38	194	218
2018/07/12 06:05:38	374	336
2018/07/12 06:20:38	46	59
2018/07/12 06:35:38	0	0
2018/07/12 06:50:38	301	259
2018/07/12 07:05:38	451	391
2018/07/12 07:20:38	496	435
2018/07/12 07:35:38	497	430
2018/07/12 07:50:38	145	102
2018/07/12 08:05:38	110	102
2018/07/12 08:20:38	97	91
2018/07/12 08:35:38	126	88
2018/07/12 08:50:38	125	87
2018/07/12 09:05:38	125	92
2018/07/12 09:20:38	118	76

Interactive Animation

The charts belonging to the **Timechart** response type include an interactive play button. It allows you to slide through values of the charts concerning time buckets known as **Interval**.

Timestamp	Dangerous	Traffic
2018/07/12 05:35:38	0	0
2018/07/12 05:50:38	194	218
2018/07/12 06:05:38	374	336
2018/07/12 06:20:38	46	59

Click **Play** on the right side of the container to start the animation. The graph is refreshed every four seconds, i.e., that graph shifts from one time-bucket to another time bucket every four seconds. Value of the time bucket is dependent upon the time-range specified in the **Interval**.

You can also click **Pause**, **Stop**, **Previous**, **Next**, **Replay** as required.

Timechart Multiple Aggregation with Grouping

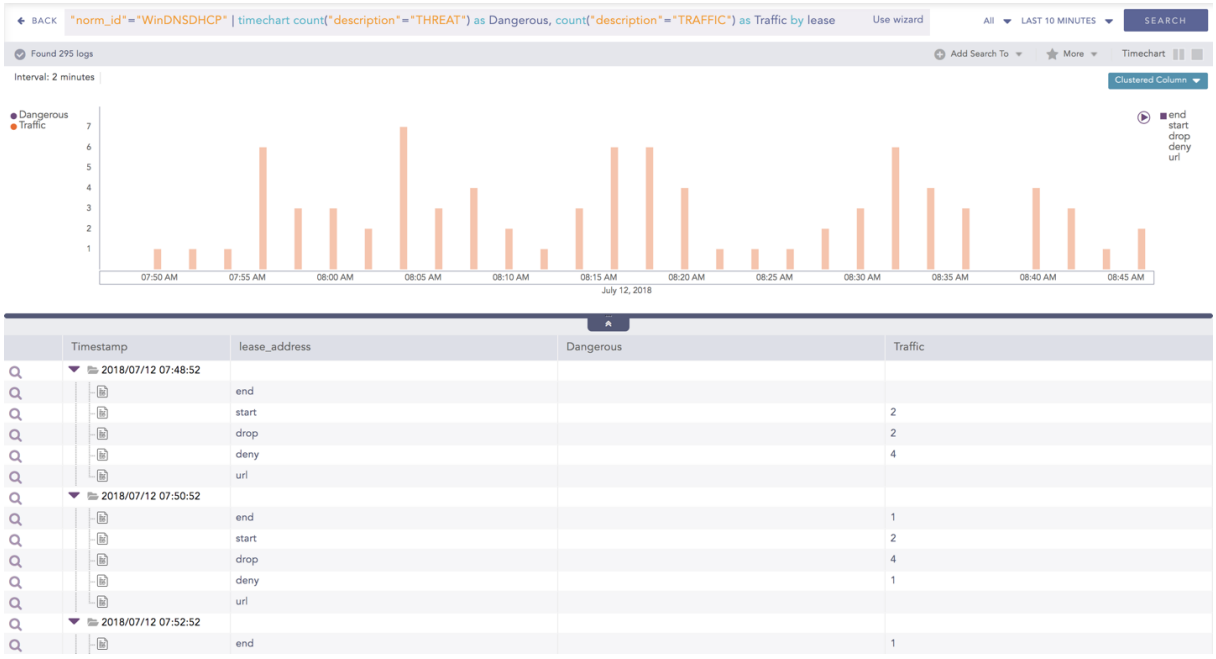
The Timechart Multiple Aggregation with Grouping response type is used for aggregation of an individual grouping parameter for given multiple aggregation parameters grouped into time buckets over a specified time range.

The general syntax for **Timechart Multiple Aggregation without Grouping** is:

```
| timechart aggregation_parameter1, aggregation_parameter2, ....,
aggregation_parametern by grouping_parameter1, grouping_parameter2, ....,
grouping_parametern
```

An example of a search query for the response is:

```
"norm_id"="WinDNSDHCP" | timechart count("description" = "THREAT") as
Dangerous, count("description" = "TRAFFIC") as Traffic by lease_address
```



Timestamp	lease_address	Dangerous	Traffic
2018/07/12 07:48:52			
	end		
	start		2
	drop		2
	deny		4
	url		
2018/07/12 07:50:52			
	end		1
	start		2
	drop		4
	deny		1
	url		
2018/07/12 07:52:52			
	end		1

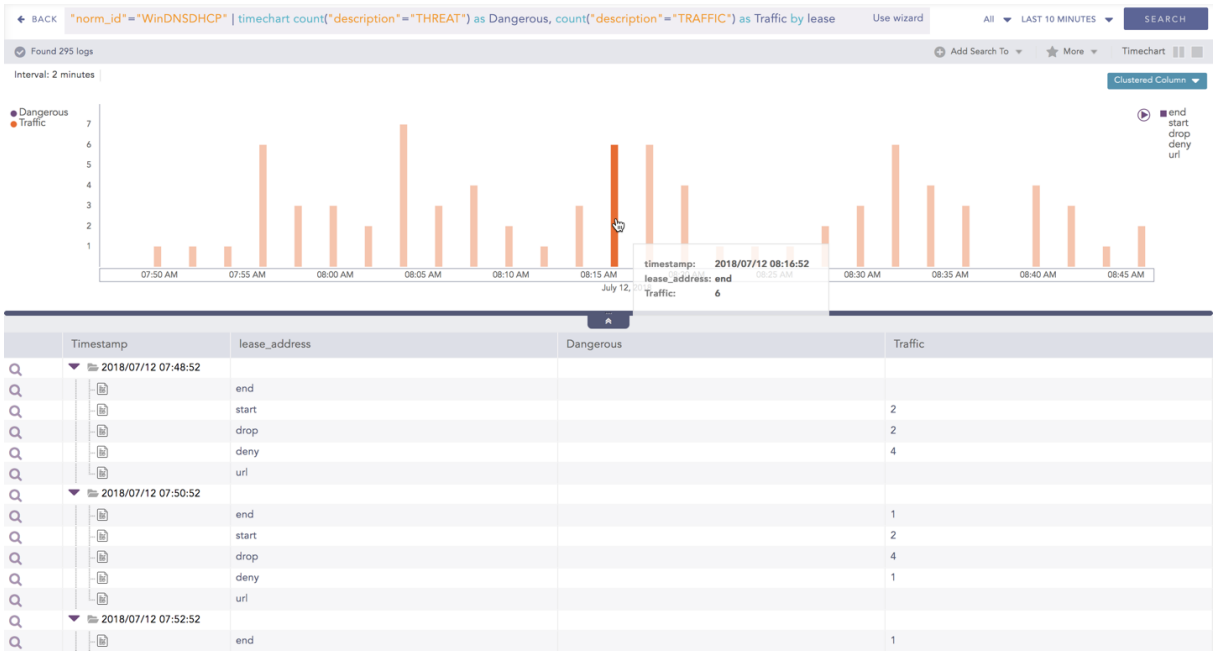
The result of this query can be represented in the form of : **Clustered Column** and **Bubble** charts.

Visualization Type	Sample Search Query
Clustered Column Chart	`norm_id=WinDNSDHCP | timechart count(lease_address=drop) as Dropped, count(lease_address=start) as Started,count(lease_address=end) as ENDED`
Bubble Chart	`| timechart count(), avg(sig_id), max(datasize), distinct_count(sig_id) by status_code`

General operations for Timechart Multiple Aggregation with Grouping

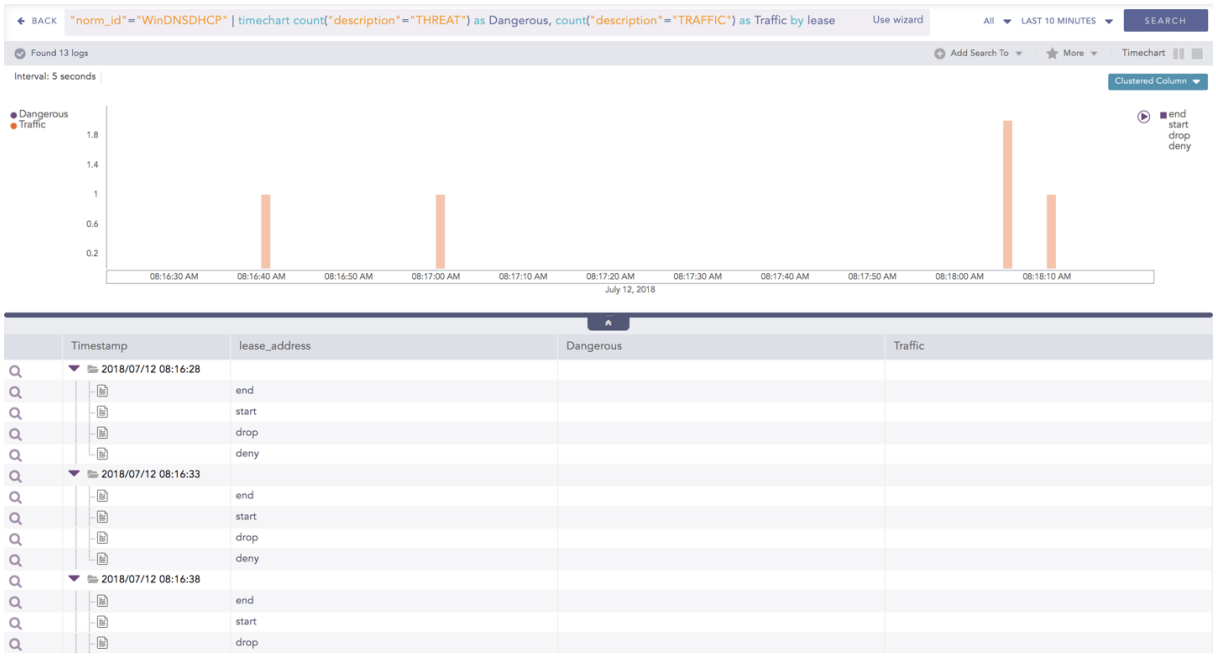
Drill-down

Like in the search results of other responses, when you hover on any section (here, any count () or avg(doable_mps)), the selected section is highlighted, and the information for the selected section is as shown in the tooltip.

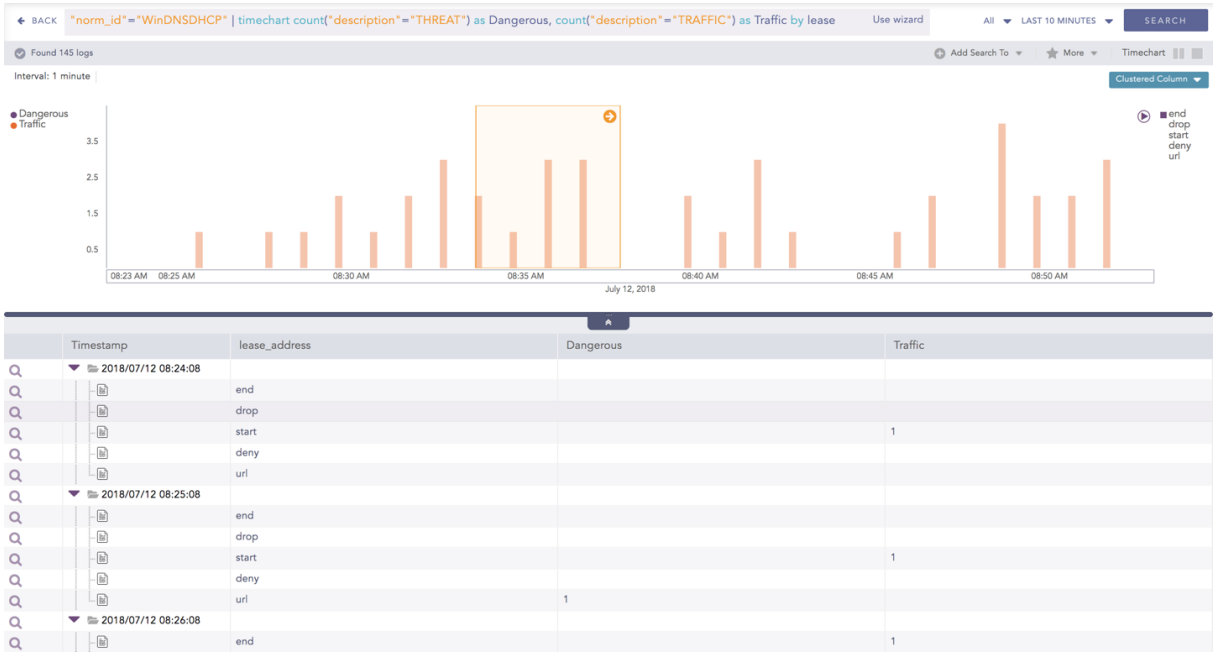


Click the component to open a drill-down window. The window summarizes the related information of the selected section along with the option to drill down as per your preference.

Click the corresponding **Open in a new window** icon to further drill-down the search result from any field. Additionally, click the **View Logs** to view the search result for the selected set of data.



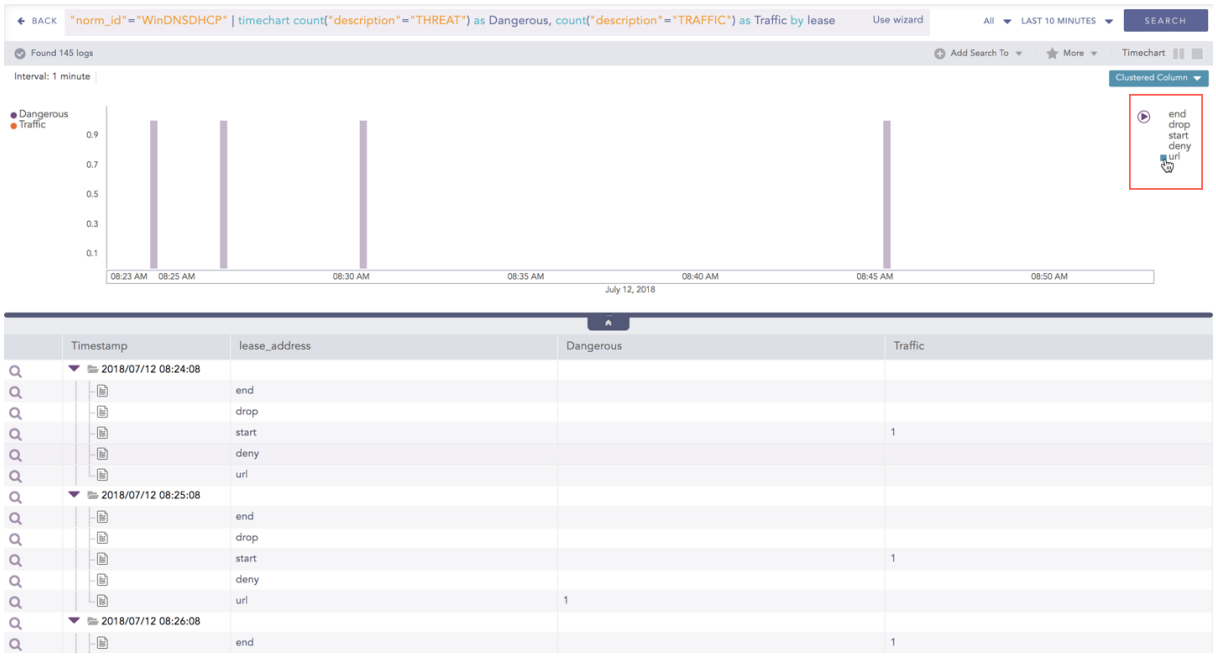
In addition to that, you can also drill-down any chart of the response type using the drag box. Click and drag the mouse inside the graph, a yellow colored transparent drag box appears. You can drill-down the selected section of the chart by clicking the drill-down icon on the top-right corner of the box. You can resize or move the drag box as per your requirement.



Interactive Legend

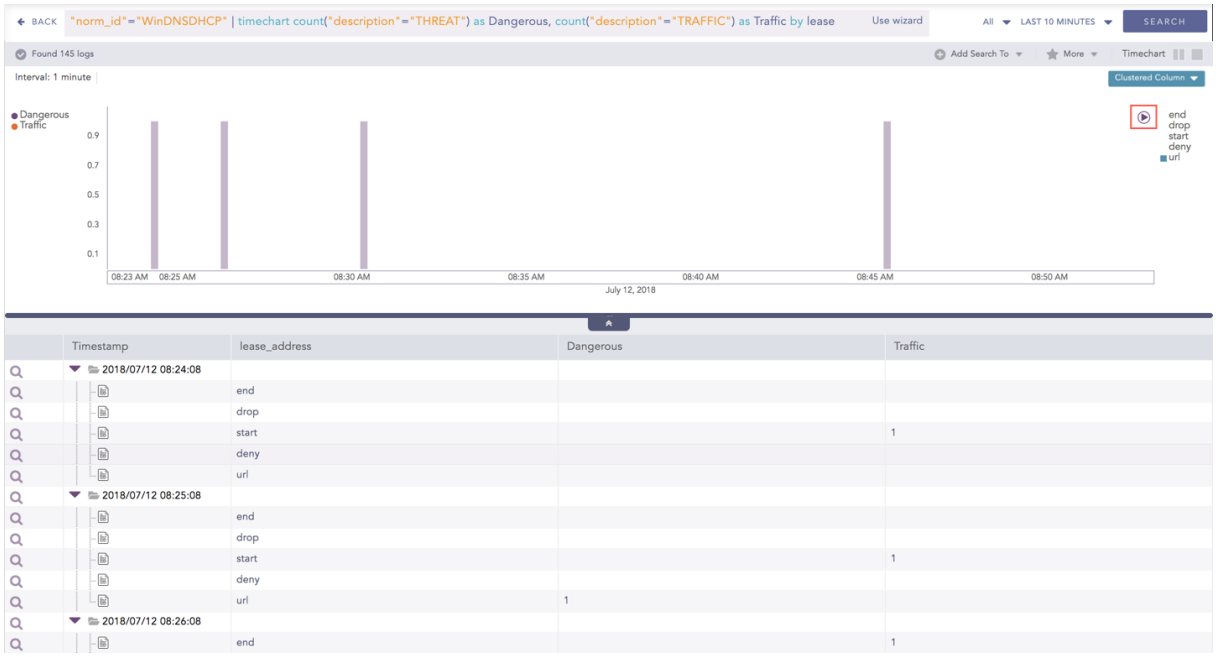
For the responses of **Timechart Multiple Aggregation with Grouping**, the legend is displayed on either side of the search graph. The aggregation parameter(s) is shown on the left-hand side whereas the grouping parameter is shown on the right-hand side.

An important thing to note here is that at an instant, the result of only one of the grouping parameters is displayed. Moreover, only the legends of the grouping parameter (on the right) are interactive. The legends of aggregation parameters (on the left) are not interactive.



Interactive Animation

The charts belonging to the **Timechart** response type include an interactive Play button. It allows you to slide through values of the charts concerning time buckets known as **Interval**.



Click the **Play** on the right side of the container to start the animation. The graph is refreshed every four seconds, i.e., that graph shifts from one time-bucket to another time bucket every four seconds. Value of the time bucket is dependent upon the time-range specified in the **Interval**.

You can also click **Pause**, **Stop**, **Previous**, **Next**, **Replay** as required.

Features of Visualization

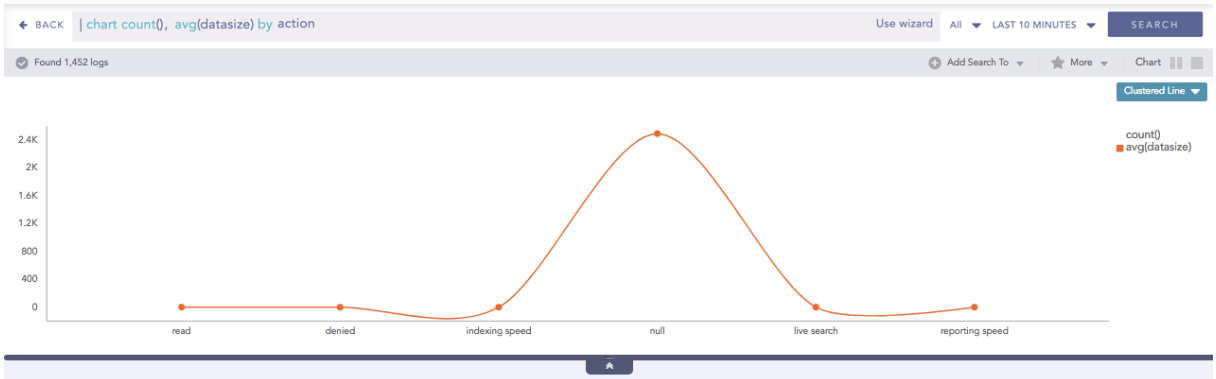
Some features of the new visualization of search graphs are provided below:

The legend for the search results is interactive in multiple ways.

You can toggle the display of the legend as **ON** and **OFF**. Click the desired legend to hide/unhide it. For example,

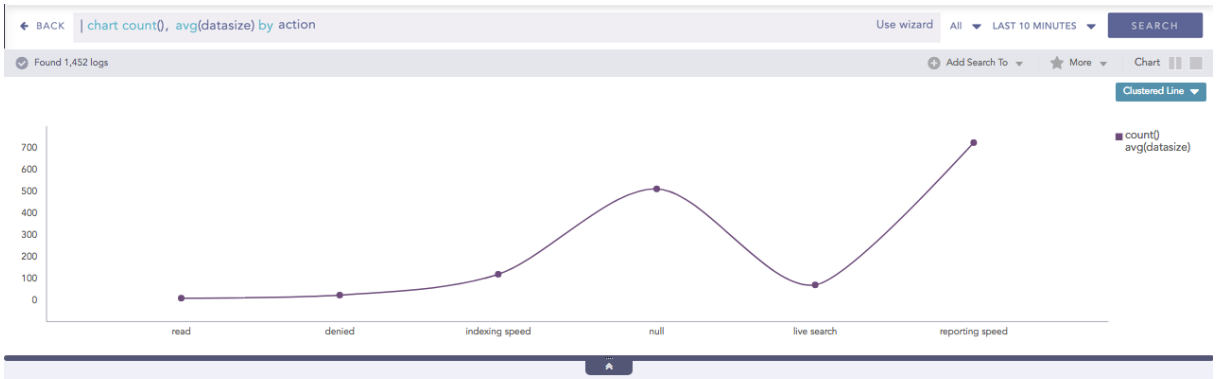
	action	count()	avg(datasize)
	read	8	0
	denied	22	0
	indexing speed	118	0
	null	511	2489.78
	live search	69	0

Click the legend of **count()** to hide its corresponding result.



	action	count()	avg(datasize)
	read	8	0
	denied	22	0
	indexing speed	118	0
	null	511	2489.78
	live search	69	0

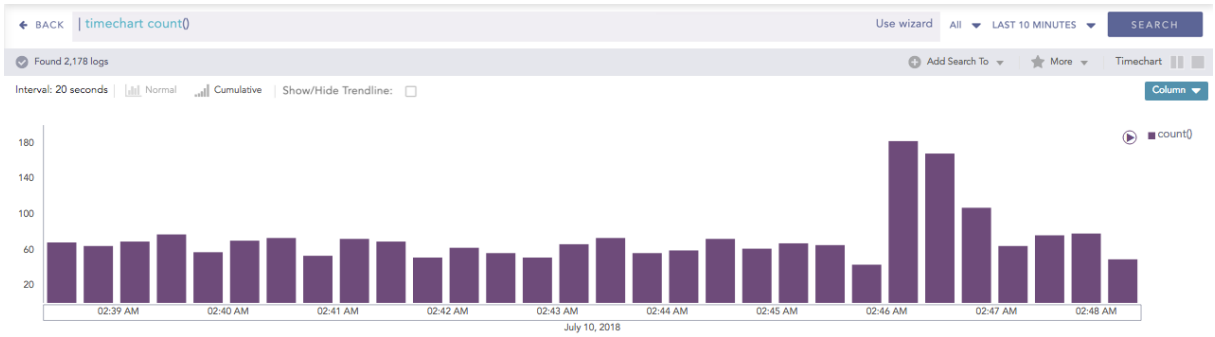
Click the legend of **avg(datasize)** to hide its corresponding result.



	action	count()	avg(datasize)
	read	8	0
	denied	22	0
	indexing speed	118	0
	null	511	2489.78
	live search	69	0

The status of the legend (either ON or OFF) is saved for the result which is dynamically populated in the widgets in the dashboards.

All the related data can be highlighted at once by hovering over the legend. If you hover the mouse over the legend of **count()** then all the data of average is highlighted.

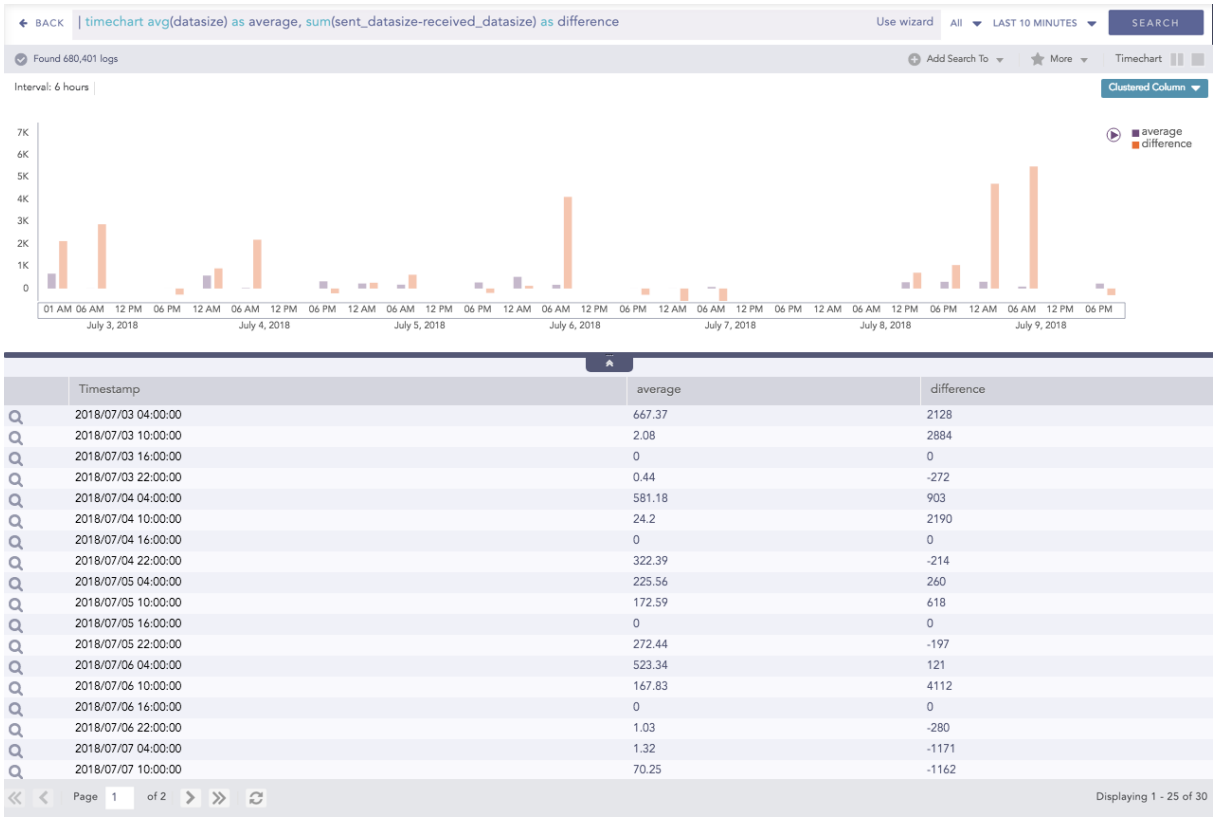


	Timestamp	count()
	2018/07/10 02:38:59	68
	2018/07/10 02:39:19	64
	2018/07/10 02:39:39	69
	2018/07/10 02:39:59	77
	2018/07/10 02:40:19	57
	2018/07/10 02:40:39	70
	2018/07/10 02:40:59	73
	2018/07/10 02:41:19	53
	2018/07/10 02:41:39	72
	2018/07/10 02:41:59	69

The Pan and Zoom feature in the axes is provided for better visibility of the results.

Pan is the ability to click and drag the cursor over the search result visualization to select the desired area. With this feature, the axes can be moved to cover a larger area of the timespan of the search result.

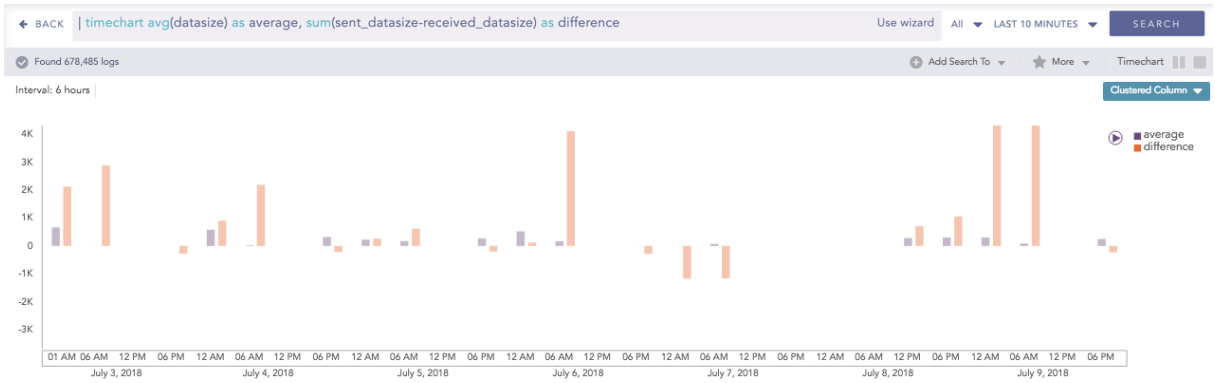
Consider the following search result:



	Timestamp	average	difference
	2018/07/03 04:00:00	667.37	2128
	2018/07/03 10:00:00	2.08	2884
	2018/07/03 16:00:00	0	0
	2018/07/03 22:00:00	0.44	-272
	2018/07/04 04:00:00	581.18	903
	2018/07/04 10:00:00	24.2	2190
	2018/07/04 16:00:00	0	0
	2018/07/04 22:00:00	322.39	-214
	2018/07/05 04:00:00	225.56	260
	2018/07/05 10:00:00	172.59	618
	2018/07/05 16:00:00	0	0
	2018/07/05 22:00:00	272.44	-197
	2018/07/06 04:00:00	523.34	121
	2018/07/06 10:00:00	167.83	4112
	2018/07/06 16:00:00	0	0
	2018/07/06 22:00:00	1.03	-280
	2018/07/07 04:00:00	1.32	-1171
	2018/07/07 10:00:00	70.25	-1162

Page 1 of 2 Displaying 1 - 25 of 30

If you click and drag the y-axis, it is displayed as in the following screenshot:

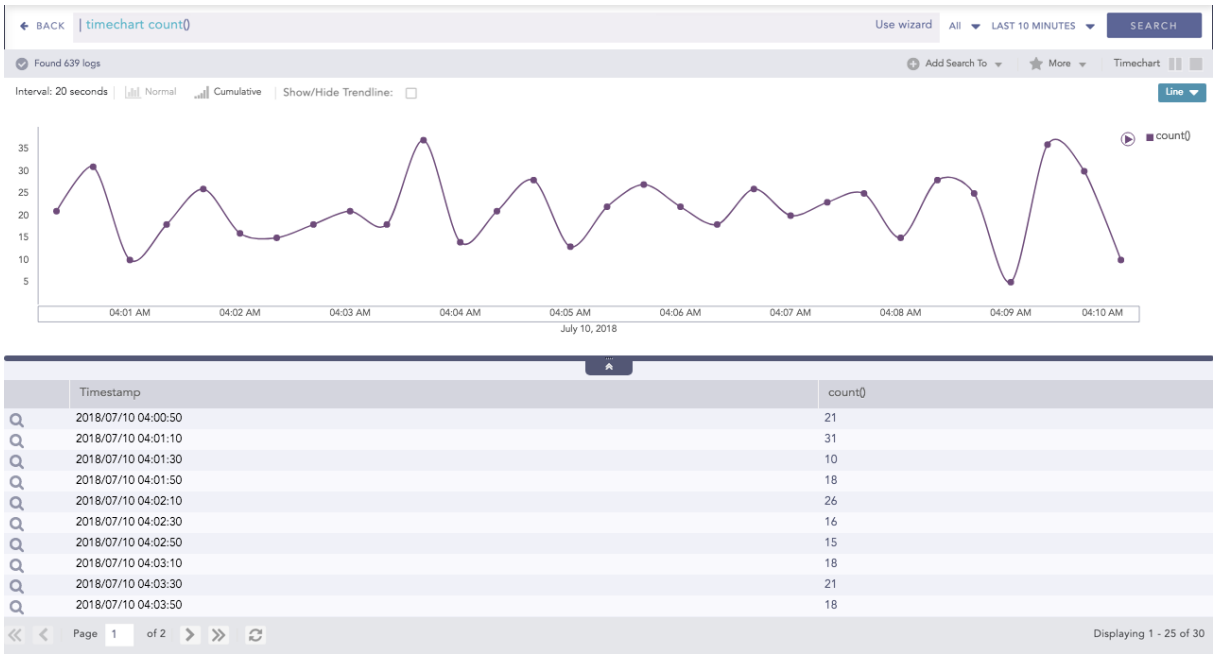


	Timestamp	average	difference
	2018/07/03 04:00:00	667.37	2128
	2018/07/03 10:00:00	2.08	2884
	2018/07/03 16:00:00	0	0
	2018/07/03 22:00:00	0.44	-272
	2018/07/04 04:00:00	581.18	903
	2018/07/04 10:00:00	24.2	2190
	2018/07/04 16:00:00	0	0
	2018/07/04 22:00:00	322.39	-214
	2018/07/05 04:00:00	225.56	260
	2018/07/05 10:00:00	172.59	618
	2018/07/05 16:00:00	0	0
	2018/07/05 22:00:00	272.44	-197
	2018/07/06 04:00:00	523.34	121
	2018/07/06 10:00:00	167.83	4112
	2018/07/06 16:00:00	0	0
	2018/07/06 22:00:00	1.03	-280
	2018/07/07 04:00:00	1.32	-1171
	2018/07/07 10:00:00	70.25	-1162

Page 1 of 2 Displaying 1 - 25 of 30

Zoom is the ability to expand and shrink the scale of the axis. With this feature, the scale of the axes can be zoomed in and out for better visibility of the search results.

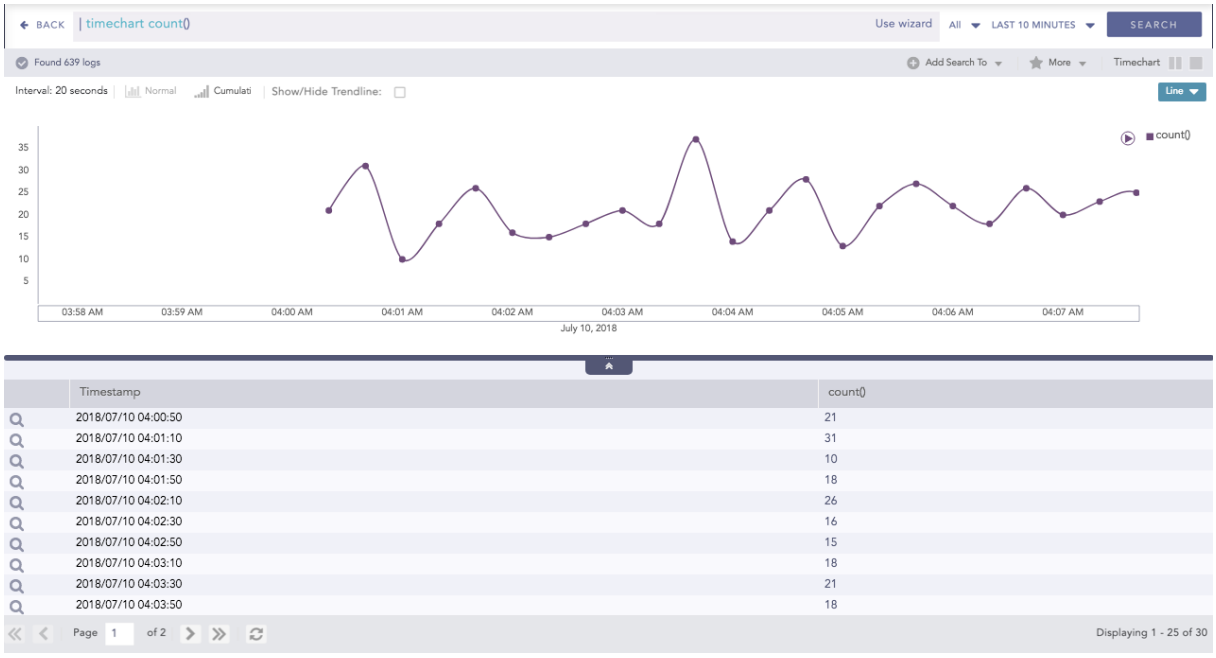
Consider the following search result:



Timestamp	count()
2018/07/10 04:00:50	21
2018/07/10 04:01:10	31
2018/07/10 04:01:30	10
2018/07/10 04:01:50	18
2018/07/10 04:02:10	26
2018/07/10 04:02:30	16
2018/07/10 04:02:50	15
2018/07/10 04:03:10	18
2018/07/10 04:03:30	21
2018/07/10 04:03:50	18

Page 1 of 2 — Displaying 1 - 25 of 30

If the you zoom over the axes individually, it is displayed as in the following screenshot:

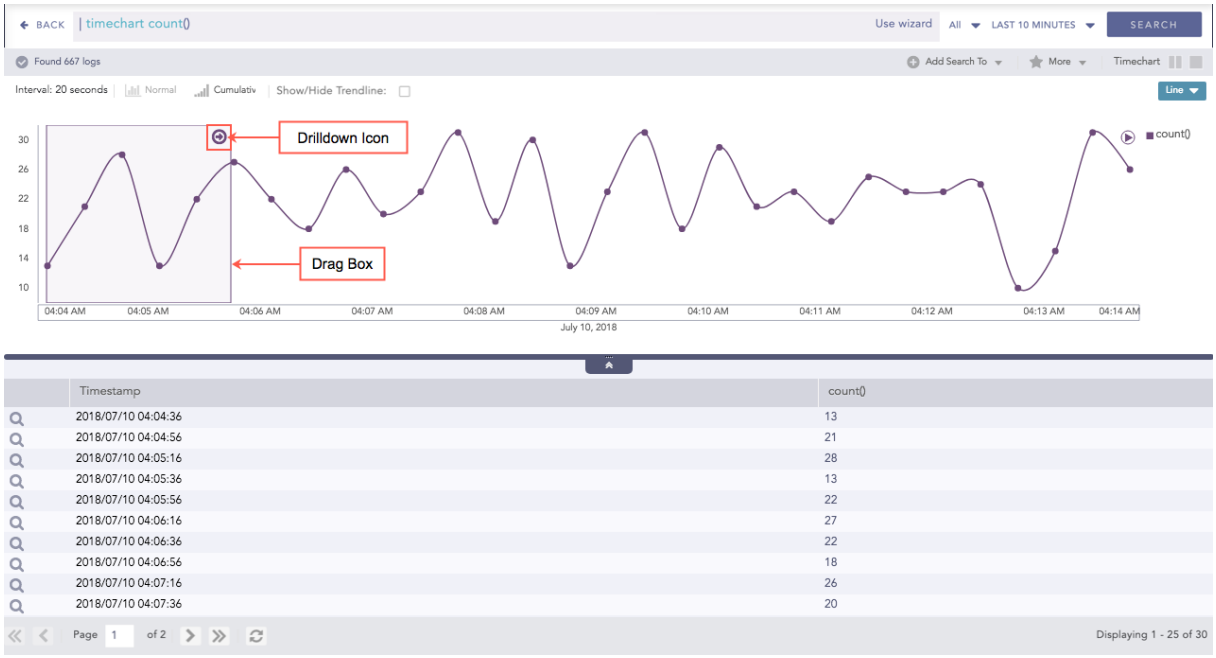


Timestamp	count()
2018/07/10 04:00:50	21
2018/07/10 04:01:10	31
2018/07/10 04:01:30	10
2018/07/10 04:01:50	18
2018/07/10 04:02:10	26
2018/07/10 04:02:30	16
2018/07/10 04:02:50	15
2018/07/10 04:03:10	18
2018/07/10 04:03:30	21
2018/07/10 04:03:50	18

Page 1 of 2 — Displaying 1 - 25 of 30

In the Timechart responses, a new feature called "Drilldown via Drag Box" has been introduced.

If you click and drag the mouse inside the container, a transparent drag box appears. This drag box is movable and resizable.

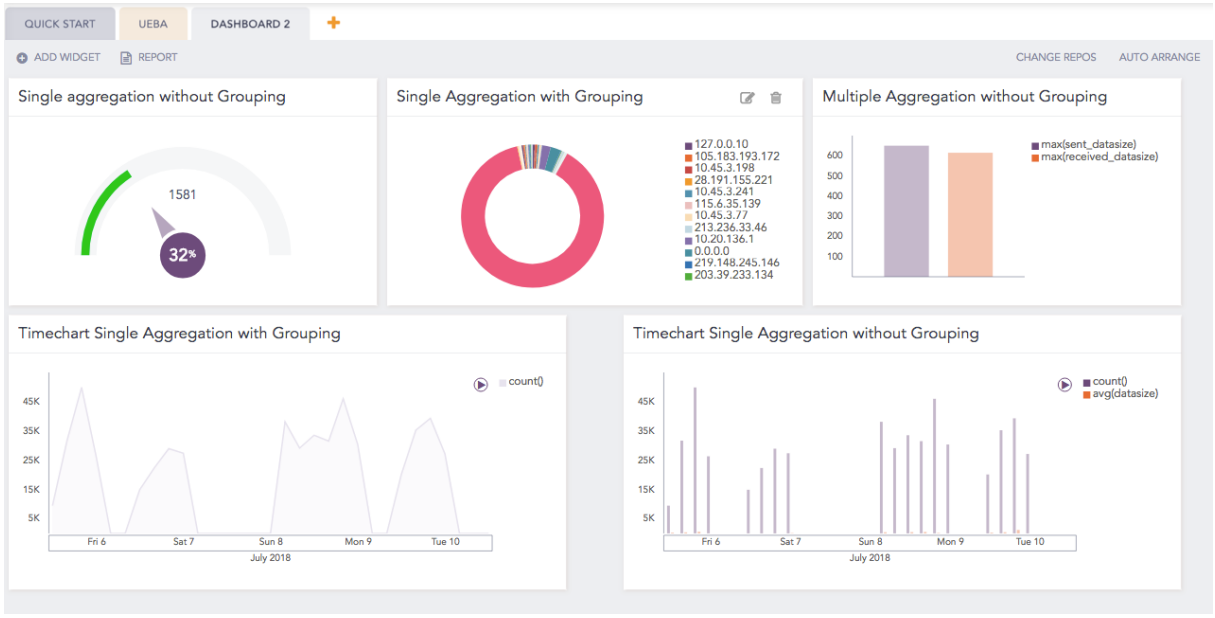


The main purpose of the drag box is to further drill-down within a custom time-range which is a subset of the previous time-range. Once the desired vicinity of the drag box is set, click the drill-down icon. This displays the search results of the particular time-frame tapped by the drag box.

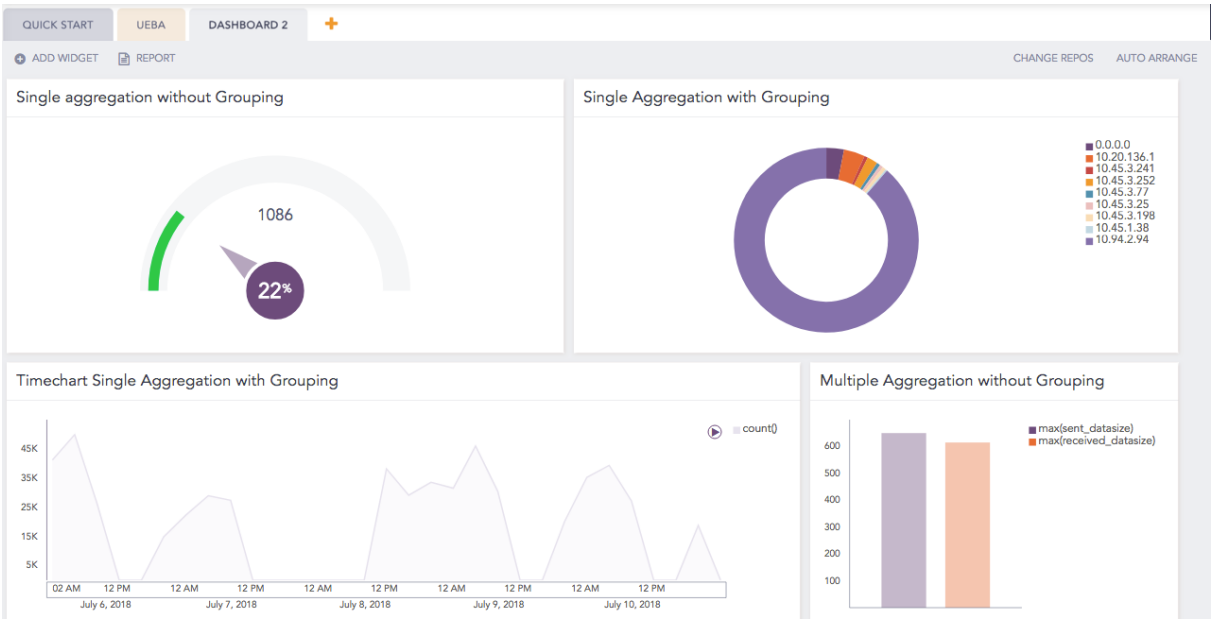
The axes label auto-adjusts as per the size of the container.

This feature is especially useful for dashboards with many widgets where the size of a widget is user-configurable. Whenever you resize a widget or click the **Auto-arrange** option, the labels of both the axes auto-adjust as per the space occupied by the search graph whenever applicable.

Consider the third widget (Multiple Aggregation with Grouping) of the following dashboard:



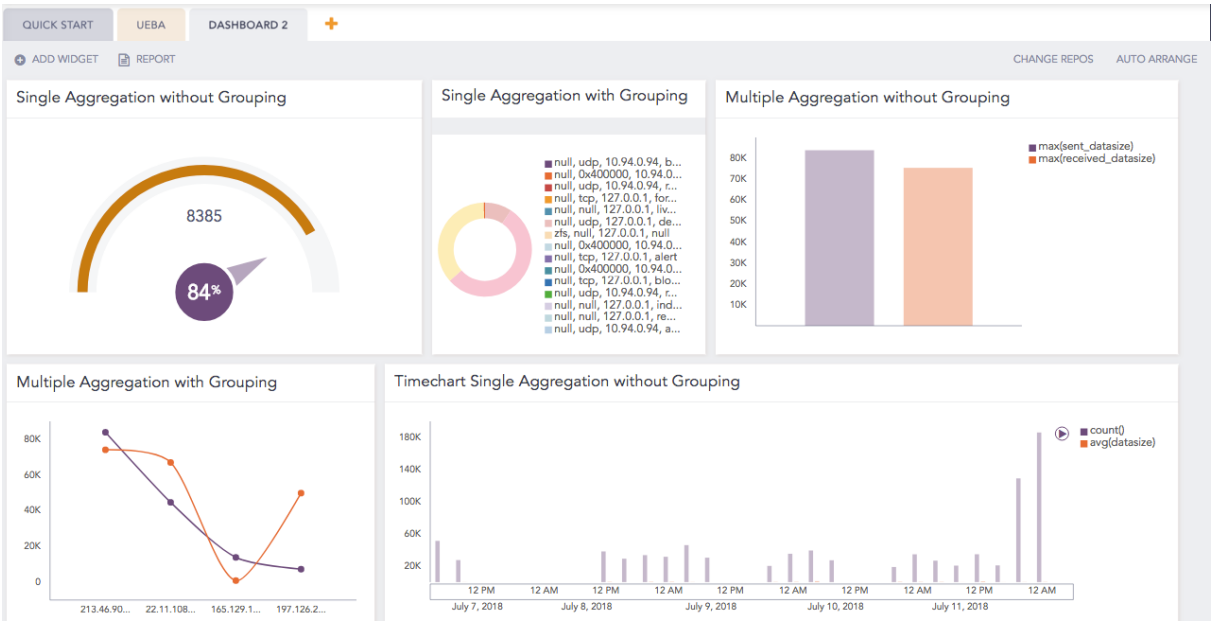
For the same dashboard, if the widget's size is increased, the labels of the axes are auto-adjusted.



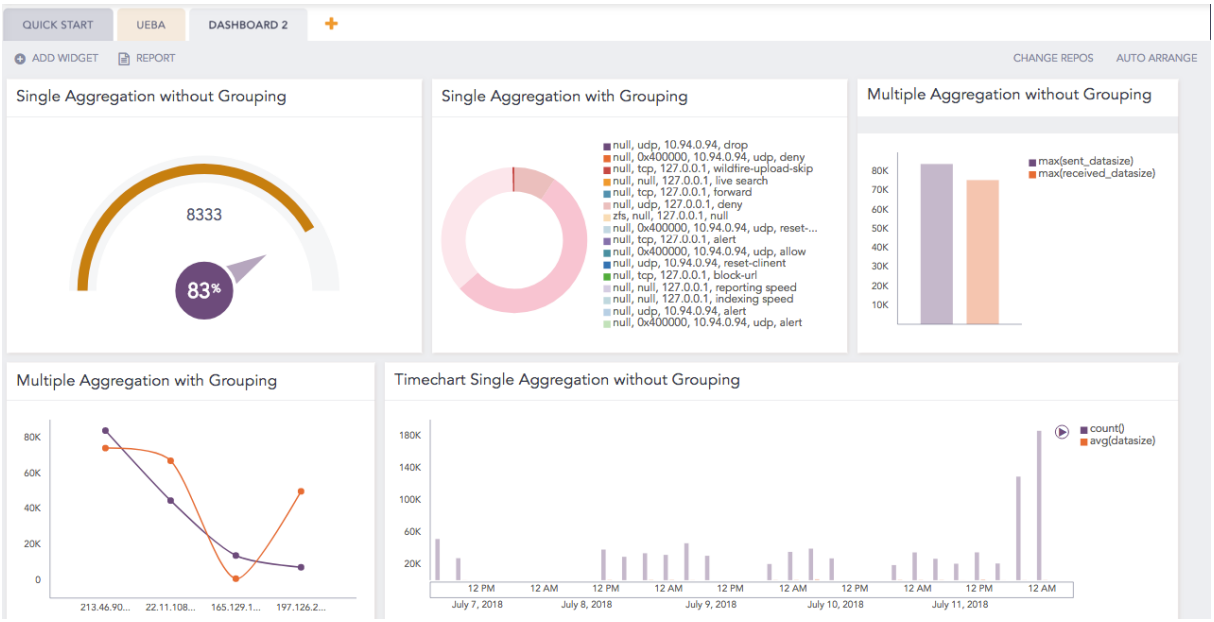
The legend's text auto adjusts as per the widget's dimension.

When the container's dimension is expanded or shrunk, the legend's text auto-adjusts without blocking the search result.

Consider the following dashboard:



As you customize the size of a widget in the first row, you can see that the legend of the donut chart automatically adjusts.

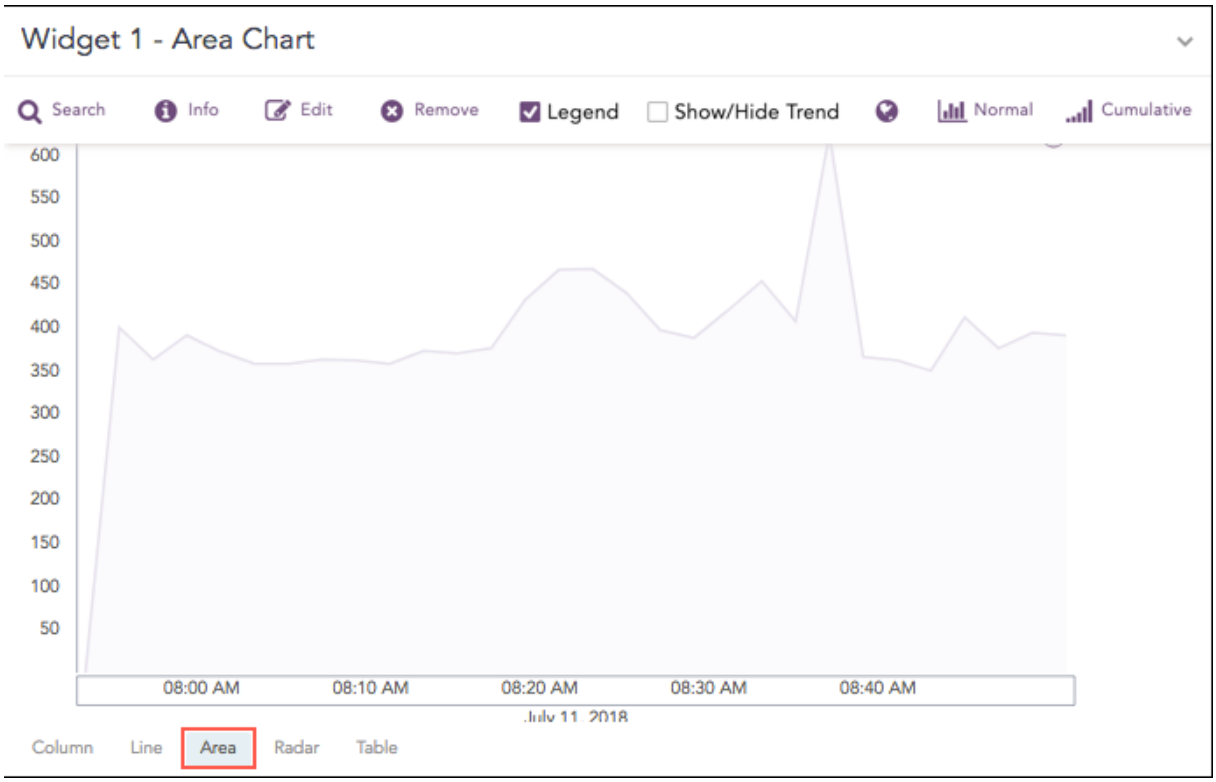

QUICK START
UEBA
DASHBOARD 2
ADD WIDGET
REPORT
CHANGE REPOS
AUTO ARRANGE
Single Aggregation without Grouping
8333
83%
Single Aggregation with Grouping
null, udp, 10.94.0.94, drop
null, 0x400000, 10.94.0.94, udp, deny
null, tcp, 127.0.0.1, wildfire-upload-skip
null, null, 127.0.0.1, live search
null, tcp, 127.0.0.1, forward
null, udp, 127.0.0.1, deny
zfs, null, 127.0.0.1, null
null, 0x400000, 10.94.0.94, udp, reset-...
null, tcp, 127.0.0.1, alert
null, 0x400000, 10.94.0.94, udp, allow
null, udp, 10.94.0.94, reset-clinent
null, tcp, 127.0.0.1, block-url
null, null, 127.0.0.1, reporting speed
null, null, 127.0.0.1, indexing speed
null, udp, 10.94.0.94, alert
null, 0x400000, 10.94.0.94, udp, alert
Multiple Aggregation without Grouping
max(sent_datasize)
max(received_datasize)
80K
70K
60K
50K
40K
30K
20K
10K
Multiple Aggregation with Grouping
80K
60K
40K
20K
0
213.46.90...
22.11.108...
165.129.1...
197.126.2...
Timechart Single Aggregation without Grouping
count()
avg(datasize)
180K
140K
100K
60K
20K
12 PM
12 AM
July 7, 2018
July 8, 2018
July 9, 2018
July 10, 2018
July 11, 2018

Area Chart

The Area chart is used to represent quantitative data graphically. The graph is used to interpret the quantitative statistics graphically. The graph is based on a Line graph, and the area between the x-axis and lines are emphasized with colors, textures or hatchings.

Area charts are used to represent accumulated totals using numbers and percentages. It is also used to show the trends over time along with all related attributes.

The x-axis of the Area chart represents the grouping parameter(s), and the y-axis represents values of the aggregation parameter.



The following query gives the output shown above.

```
| timechart count()
```

Response Types Supported

The **Area** chart supports two aggregation response types for representation of search results in the visualization. They are:

Response Type	General Syntax
Single Aggregation with Grouping	`| chart aggregation_parameter by grouping_parameter1, grouping_parameter2, ........, grouping_parametern`
Timechart Single Aggregation without Grouping	`| timechart aggregation_parameter`

Single Aggregation with Grouping

Example:

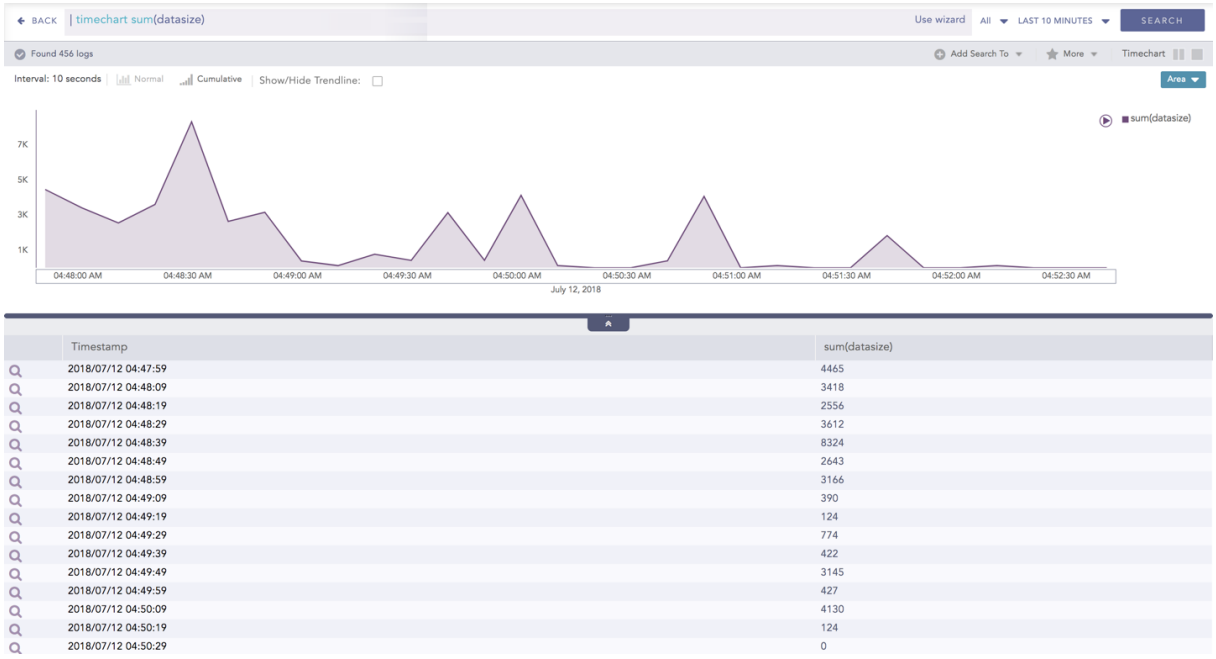
```
action=* source_address=* | chart count() by action, source_address
```

You can refer to Single Aggregation with Grouping for more details.

Timechart Single Aggregation without Grouping

Example:

```
| timechart sum(datasize)
```



You can refer to Timechart Single Aggregation without Grouping for more details.

ATT&CK chart

The **ATT&CK** chart is a heatmap describing the attacks carried out on a system in the form of attack tactics, techniques, and sub-techniques described by MITRE. You can select the **ATT&CK** chart from the search page only if you provide **attack_id** as a grouping parameter.

To populate the ATT&CK chart, SLS adds the following fields to the corresponding logs each time an alert is triggered:

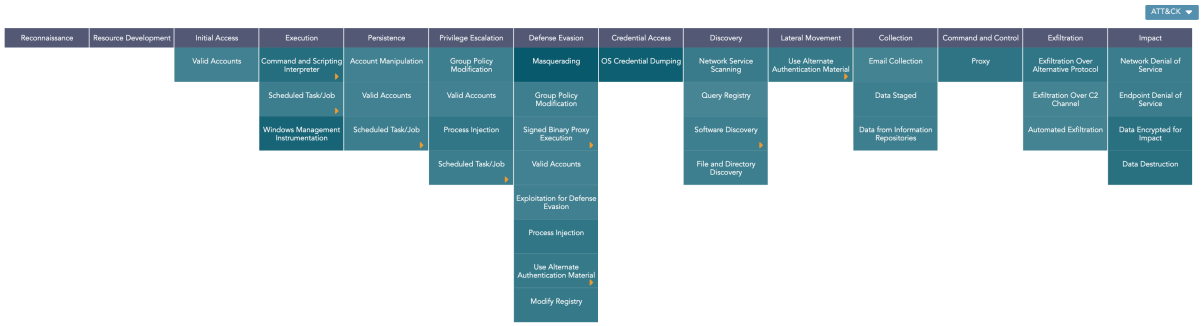
1. **attack_id**: An ID for the attack.
2. **attack_category**: The type of attack tactic used.
3. **attack_tag**: The type of attack technique used.

Description

The header row of the **ATT&CK** chart contains the tactics that may be used to perform an attack. The body of the chart displays the techniques used to execute the corresponding tactics.

The ATT&CK chart also displays a drop-down icon on some cells. You can select the icon to see the sub-techniques involved in the relevant attack technique.

> **NOTE**
> The color intensity for each cell is based on the frequency of the corresponding technique and sub-technique.



The following query gives the output shown above.

```
| chart count() by attack_id
```

> **NOTE**
> You can drill down on the ATT&CK chart by clicking on each tactic on the header row and each technique and sub-technique cell on the heatmap.

Grouping by Entities

SLS also provides the option to further analyze the ATT&CK chart based on multiple entities. You can do this by grouping the results using the required entity. SLS currently supports grouping by the following entities:

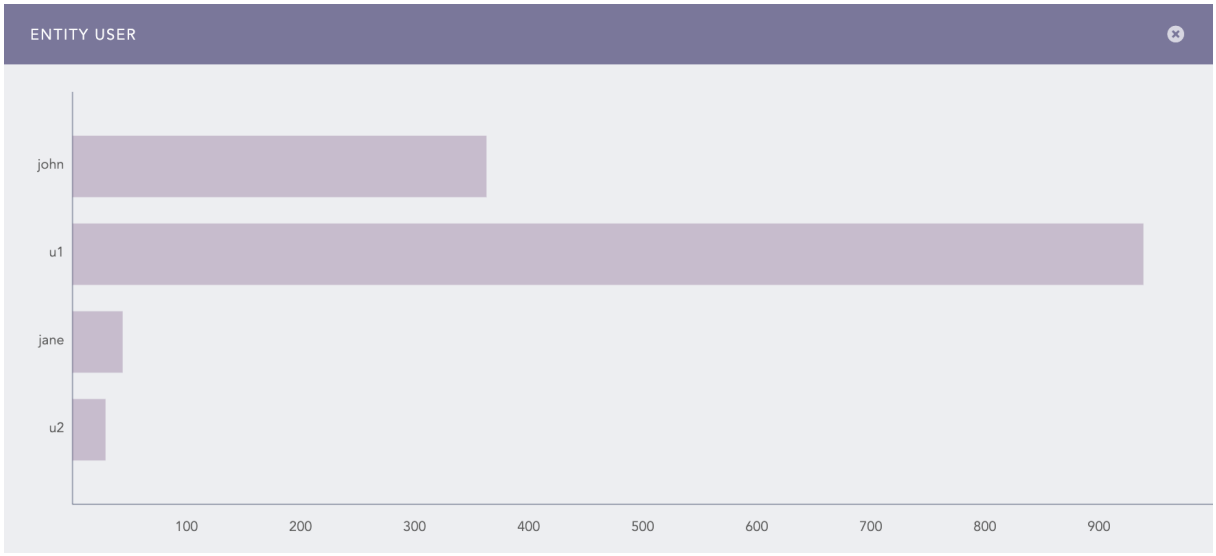
1. user
2. ip_address
3. workstation

```
| chart count() by attack_id, user
```

If you group the results by an entity, the ATT&CK chart provides the following additional features:

1. The **Entity** icon at the top-right corner of the visualization.

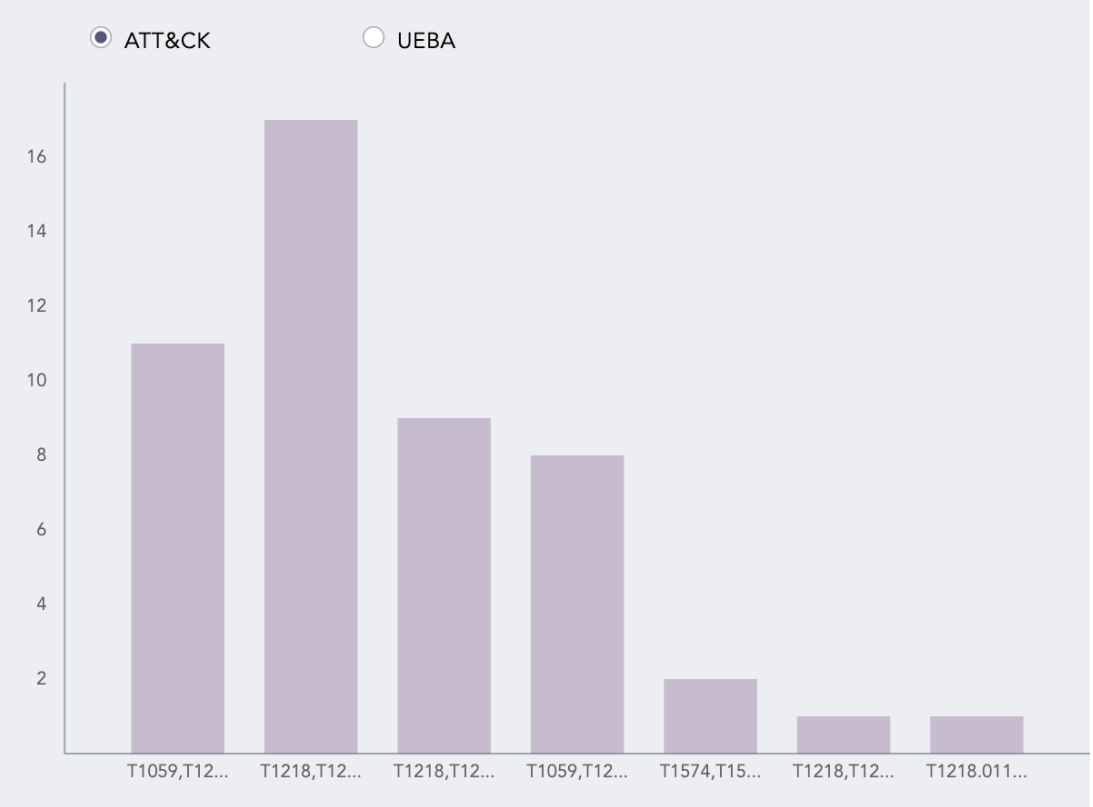
Clicking the icon displays a bar chart describing the contribution of each entity to the overall results. Here, the y-axis represents the entities and the x-axis represents the count of the entities in the overall results.



2. The **Info** icon displayed while hovering on each cell of the ATT&CK chart.

Clicking the icon displays a bar chart describing the contribution of the entities to the corresponding attack technique or sub-technique. Here, the x-axis represents the entities and

the y-axis represents the count of the entities in the results for the selected technique or sub-technique.



NOTE
The bar chart shows the data for the selected technique or sub-technique under all the attack tactics. To display the data for only the selected tactic, you must group the results by **attack_category** as well.

If you have enabled SLS UEBA, you can also view the risk scores for the entities by clicking **UEBA**. This chart displays only the entities that have a risk score of more than one.

NOTE

- The **entity** and **info** icons are displayed only if you group the results by **user**, **ip_address**, or **workstation** fields. Therefore, if you have a field named differently in the search results, make sure to rename the field to one of the required names using the **rename** command.
- If you group the results by multiple valid entities, the results are grouped only by one of the provided entities. In this case, SLS prioritizes the entities in the following order:
  - **user**
  - **ip_address**
  - **workstation**
- You can drill down on each entity's results by clicking the corresponding column of the entity bar charts. Additionally, you can drill down onto the UEBA dashboard by clicking the risk score for each entity.
- The **entity** icons are **not** displayed in the **Search Templates** and **Dashboard**.

Response Types Supported

The **ATT&CK** chart supports a single aggregation response type for the representation of search results in the visualization. It is:

Response Type	General Syntax
Single Aggregation with Grouping	`| chart aggregation_parameter by attack_id`

Single Aggregation with Grouping

Example:

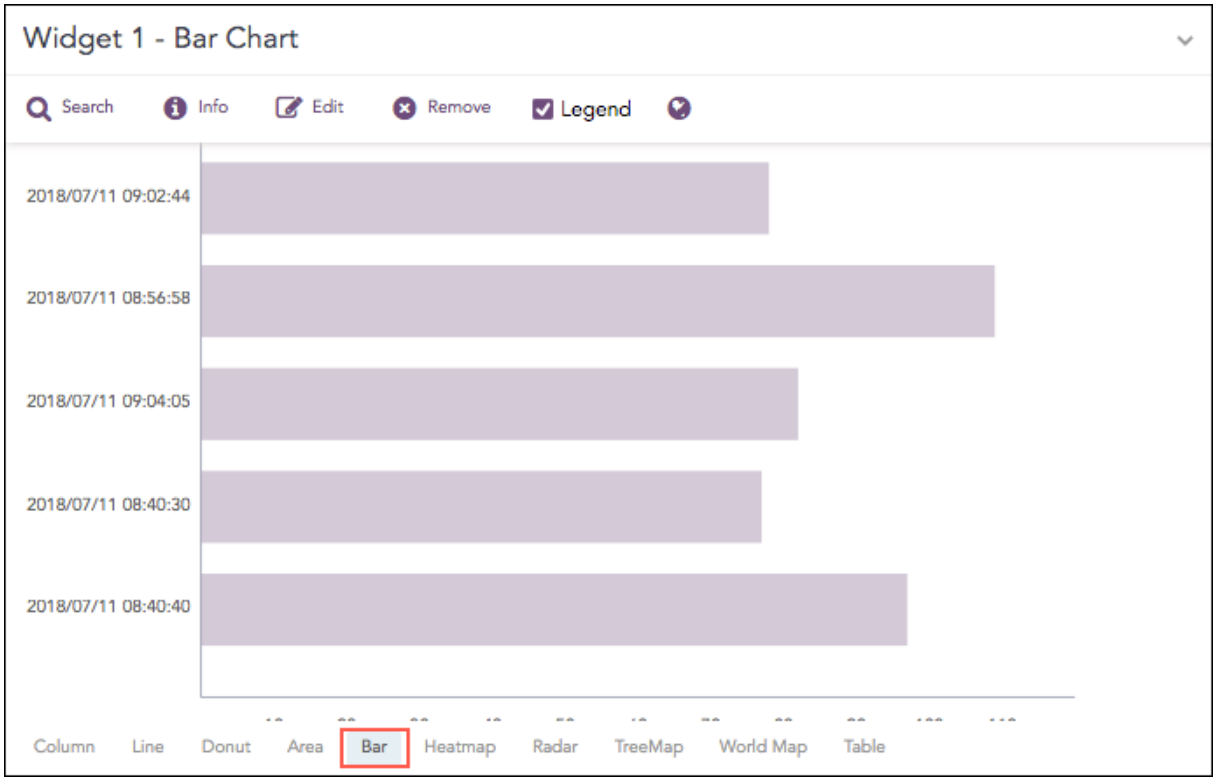
```
| chart count() by attack_id
```

You can refer to Single Aggregation with Grouping for more details.

Bar Chart

The Bar chart is a horizontal bar graph that visualizes categorical data in a rectangular bar with the width proportional to the value.

In a Bar Chart, the x-axis represents the aggregation parameter and the y-axis represents the grouping parameter(s). Besides this, it is similar to the Column Chart.



The following query gives the output shown above.

```
| chart count() by col_ts limit 5
```

Response Types Supported

The **Bar** chart supports a single aggregation response types for representation of search results in the visualization. It is :

Response Type	General Syntax
Single Aggregation with Grouping	`| chart aggregation_parameter by grouping_parameter1, grouping_parameter2, ........, grouping_parametern`

Single Aggregation with Grouping

Example:

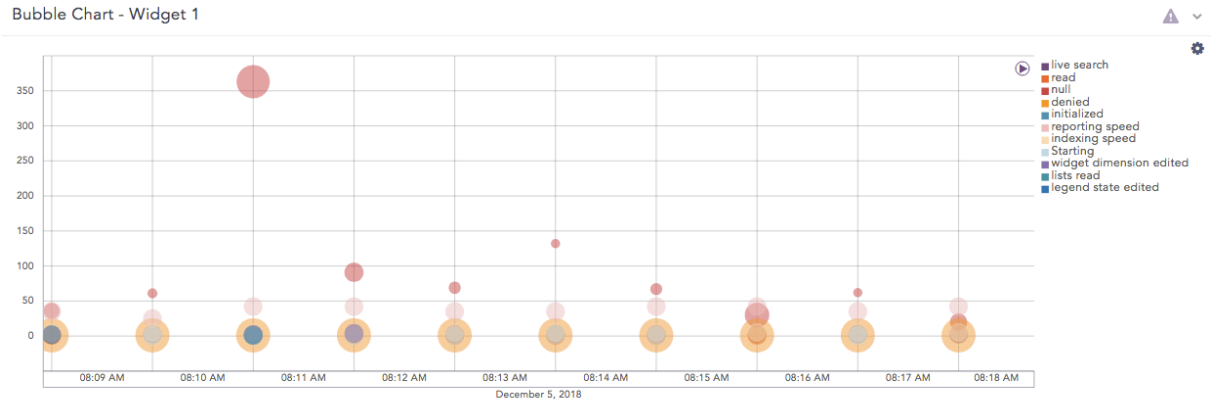
```
severity=* | chart count() by severity order by count() desc limit 5`
```

You can refer to Single Aggregation with Grouping for more details.

Bubble Chart

The Bubble Chart is a scatter chart that shows the relationship between variables using three dimensions: the x-axis, the y-axis, and the bubble radius. The chart can display different groups of data at once. Based on the grouping parameter, the chart groups the data into bubbles of different colors with each color representing a single group.

You can see the group names and their corresponding colors in the legend to the right of the chart.



The following query gives the output shown above.

```
timechart count(), avg(sig_id) by action
```

Response Types Supported

The **Bubble** chart supports two aggregation response types for representation of search results in the visualization. They are:

Response Type	General Syntax
Multiple Aggregation with Grouping	`| chart aggregation_parameter1, aggregation_parameter2 by grouping_parameter1, grouping_parameter2, ...,grouping_parametern`
Timechart Multiple Aggregation with Grouping	`| timechart aggregation_parameter1, aggregation_parameter2, ...., aggregation_parametern by grouping_parameter1, grouping_parameter2, ...., grouping_parametern`

Multiple Aggregation with Grouping

Example:

```
| chart count(), max(sig_id) by action
```

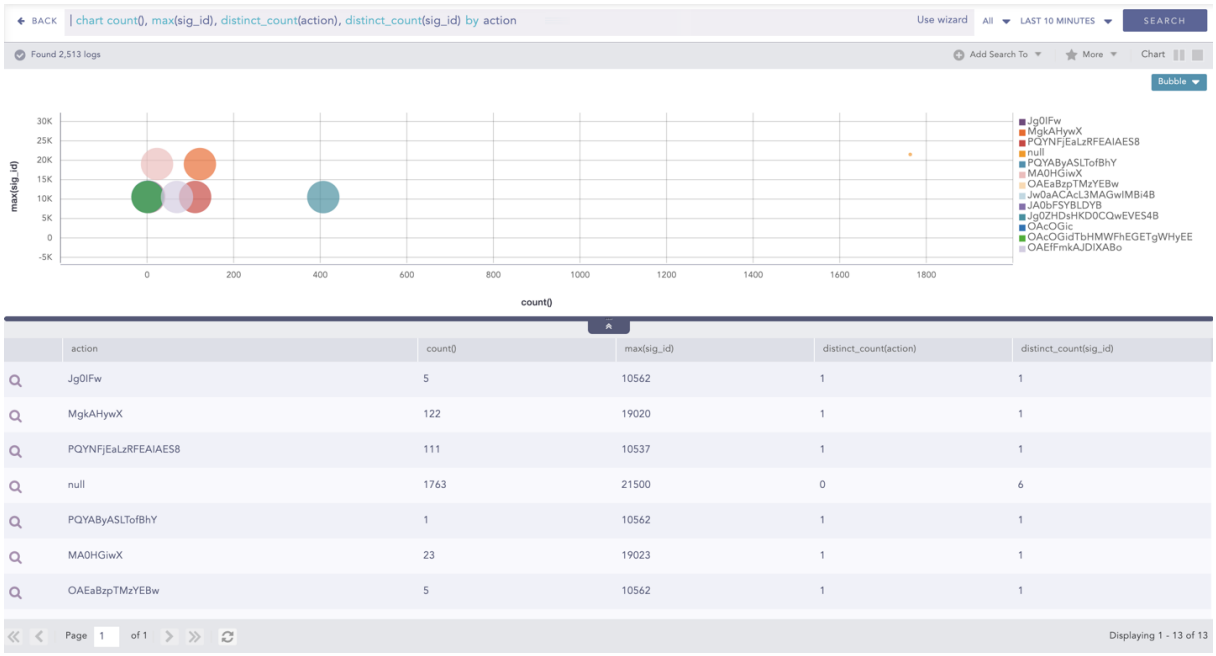
By default, in the search command for the Bubble Chart, the first aggregation parameter represents the x-axis while the next two parameters represent the y-axis and the bubble radius respectively.

You can also use the Bubble Chart with more than three aggregation parameters. To see the values of the other parameters, hover over a bubble in the chart. A tooltip appears, displaying all the values of the parameters associated with the bubble chart.

NOTE
The radii with negative values are represented in the red-colored text. However, the system takes the modulus of the negative value and plots it in the chart.

Example:

```
| chart count(), max(sig_id), distinct_count(action), distinct_count(sig_
id) by action
```

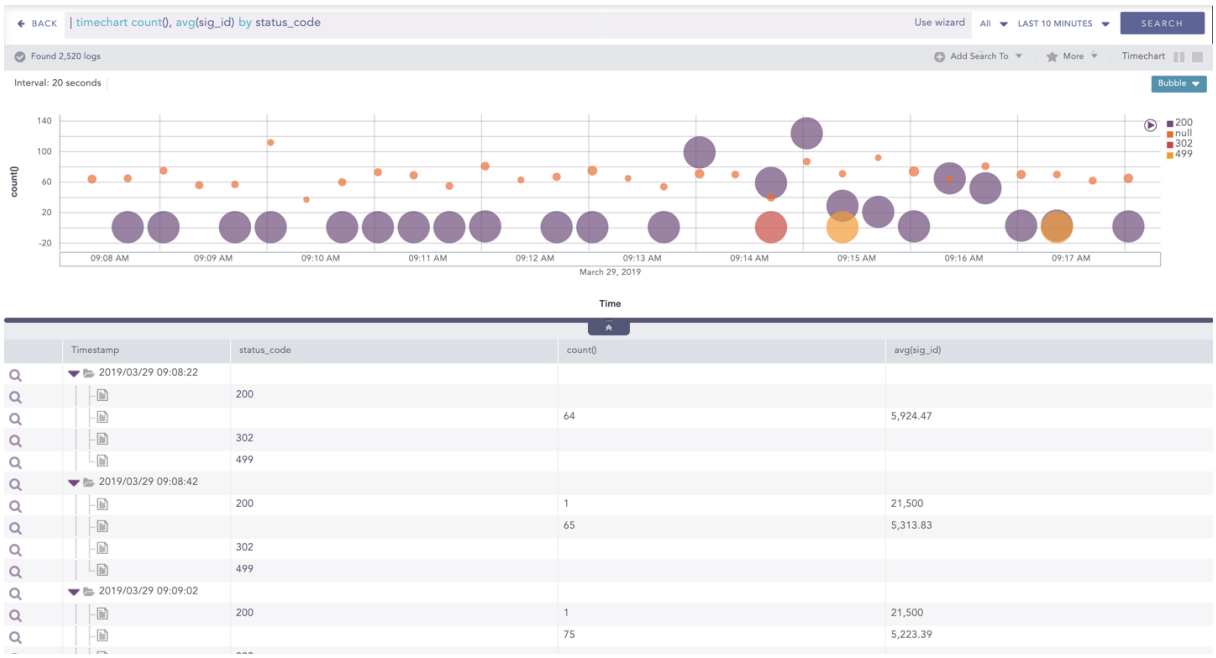


You can refer to Multiple Aggregation with Grouping for more details.

Timechart Multiple Aggregation with Grouping

Example:

```
| timechart count(), avg(sig_id) by status_code
```



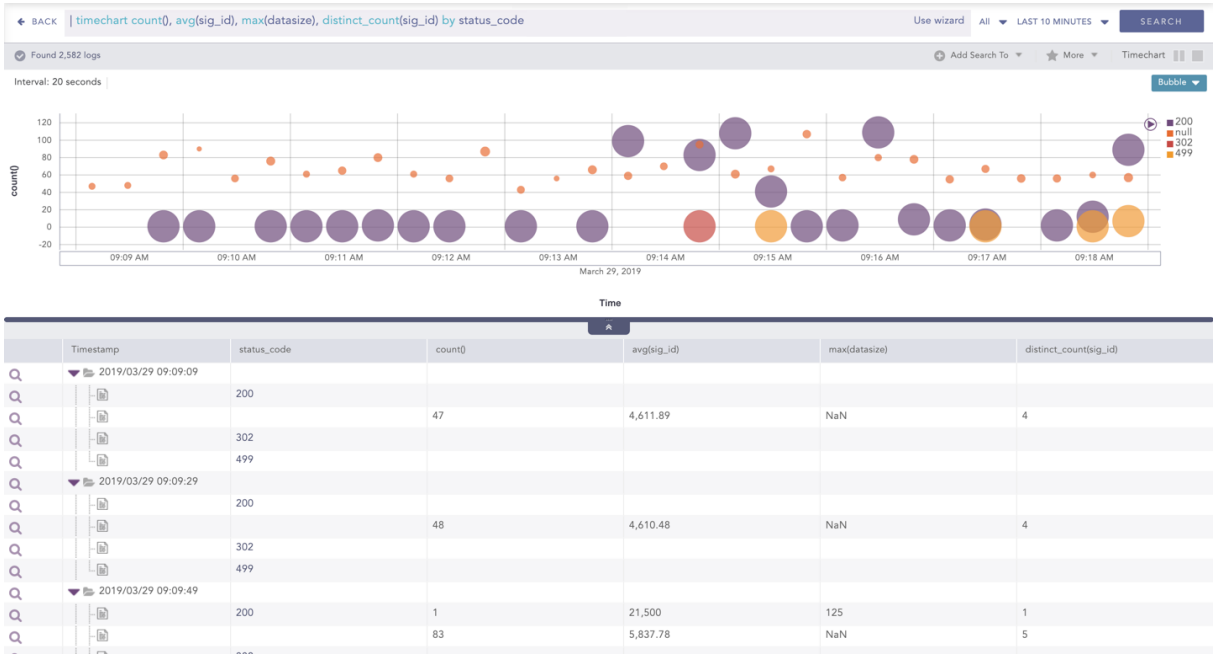
By default, in the search command for the Bubble Chart, the timechart represents the **Time** paramter in the x-axis while the next two parameters represent the y-axis and the bubble radius respectively.

You can also use the Bubble Chart with more than three aggregation parameters. To see the values of the other parameters, hover over a bubble in the chart. A tooltip appears, displaying all the values of the parameters associated with the bubble.

NOTE
The radii with negative values are represented in the red-colored text. However, the system takes the modulus of the negative value and plots it in the chart.

Example:

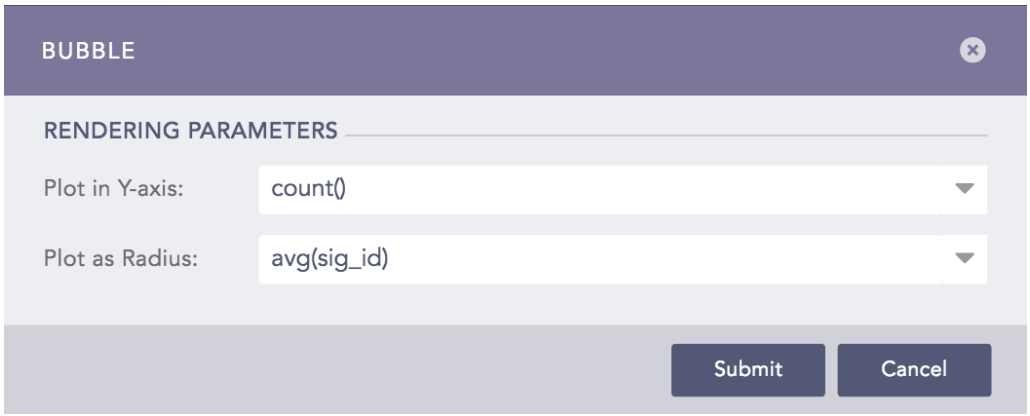
```
| timechart count(), avg(sig_id), max(datasize), distinct_count(sig_id) by
status_code
```



You can refer to Timechart Multiple Aggregation with Grouping for more details.

Rendering Parameters

Click the settings icon at the top-right corner of the Bubble Chart to open a dialog box. The dialog box allows you to configure the rendering parameters of the Bubble Chart.



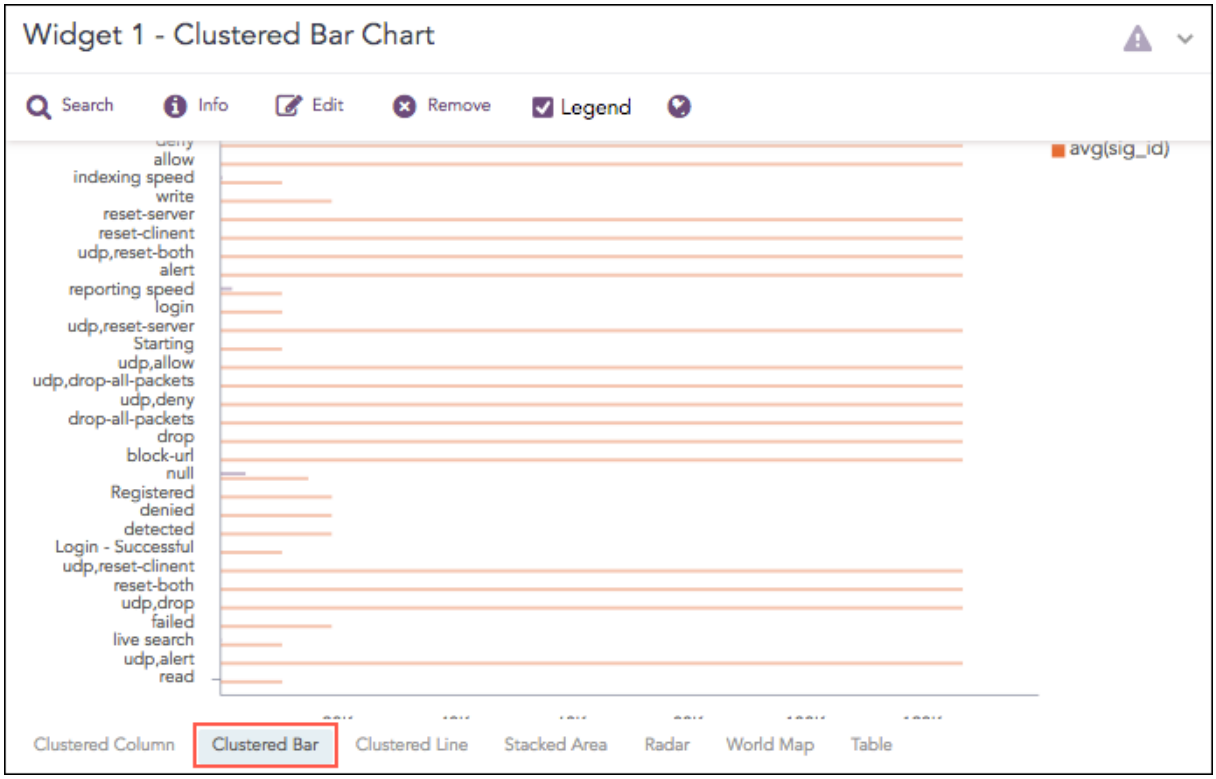
You can select the required parameters from the **Plot in Y-axis** and **Plot in Radius** drop-down menus to represent the y-axis and the bubble radius respectively.

NOTE
Make sure you select different parameters to represent the y-axis and the bubble radius.

Clustered Bar Chart

The Clustered Bar chart is a horizontal bar graph that represents multiple categorical data in a rectangular bar with the width proportional to the value.

The only difference between a Clustered Bar chart and a Clustered Column Chart is the placement of parameters. In a **Clustered Column Chart**, the aggregation parameter is placed on the x-axis whereas, in a Clustered Bar chart, the parameters are placed in the y-axis.



The following query gives the output shown above.

```
| chart count(), avg(sig_id) by action
```

Response Types Supported

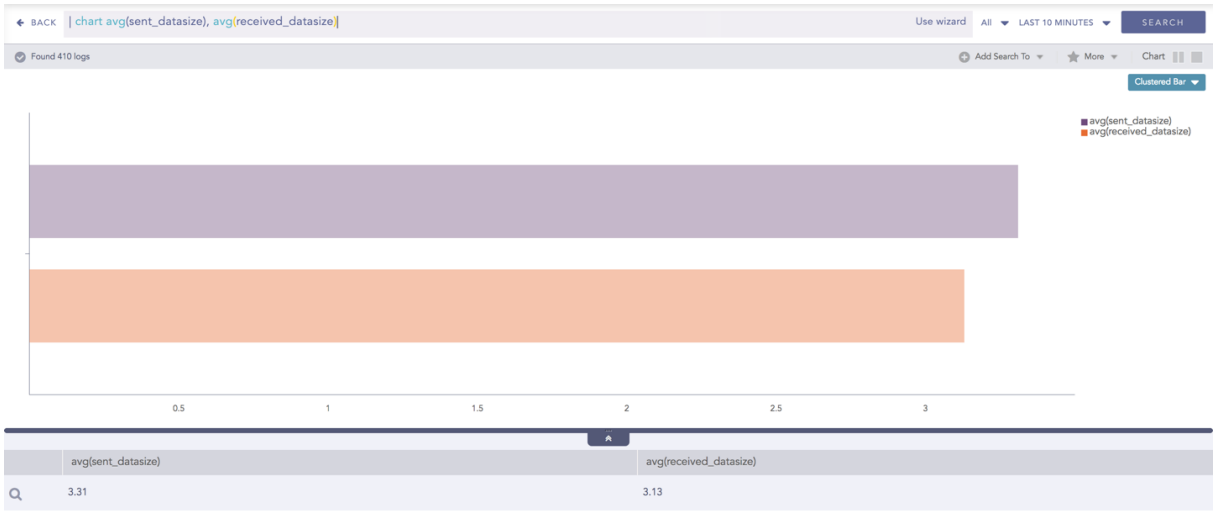
The **Clustered Bar** chart supports two aggregation response types for representation of search results in the visualization. They are:

Response Type	General Syntax
Multiple Aggregation without Grouping	`| chart count(), avg(datasize)`
Multiple Aggregation with Grouping	`| chart aggregation_parameter1, aggregation_parameter2 by grouping_parameter1, grouping_parameter2, ..., grouping_parametern`

Multiple Aggregation without Grouping

Example:

```
| chart avg(sent_datasize), avg(received_datasize)
```

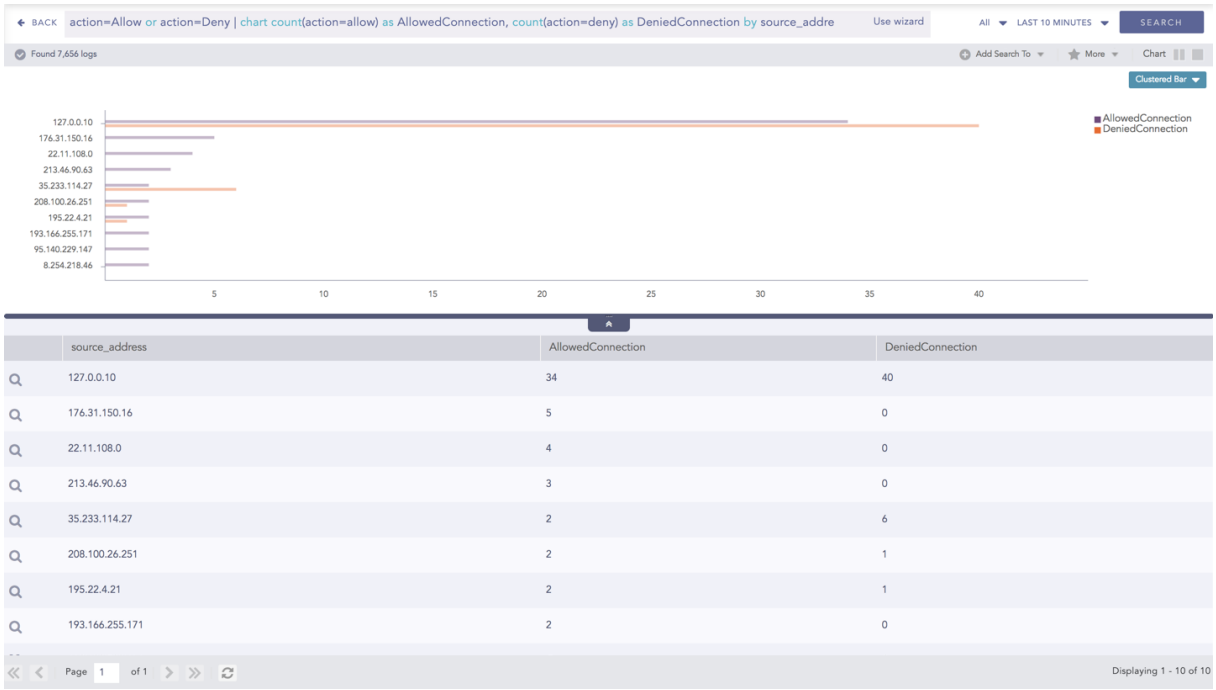


You can refer to Multiple Aggregation without Grouping for more details.

Multiple Aggregation with Grouping

Example:

```
action=Allow or action=Deny | chart count(action=allow) as
AllowedConnection, count(action=deny) as DeniedConnection by source_
address order by count(action=allow), count(action=deny) desc limit 10
```

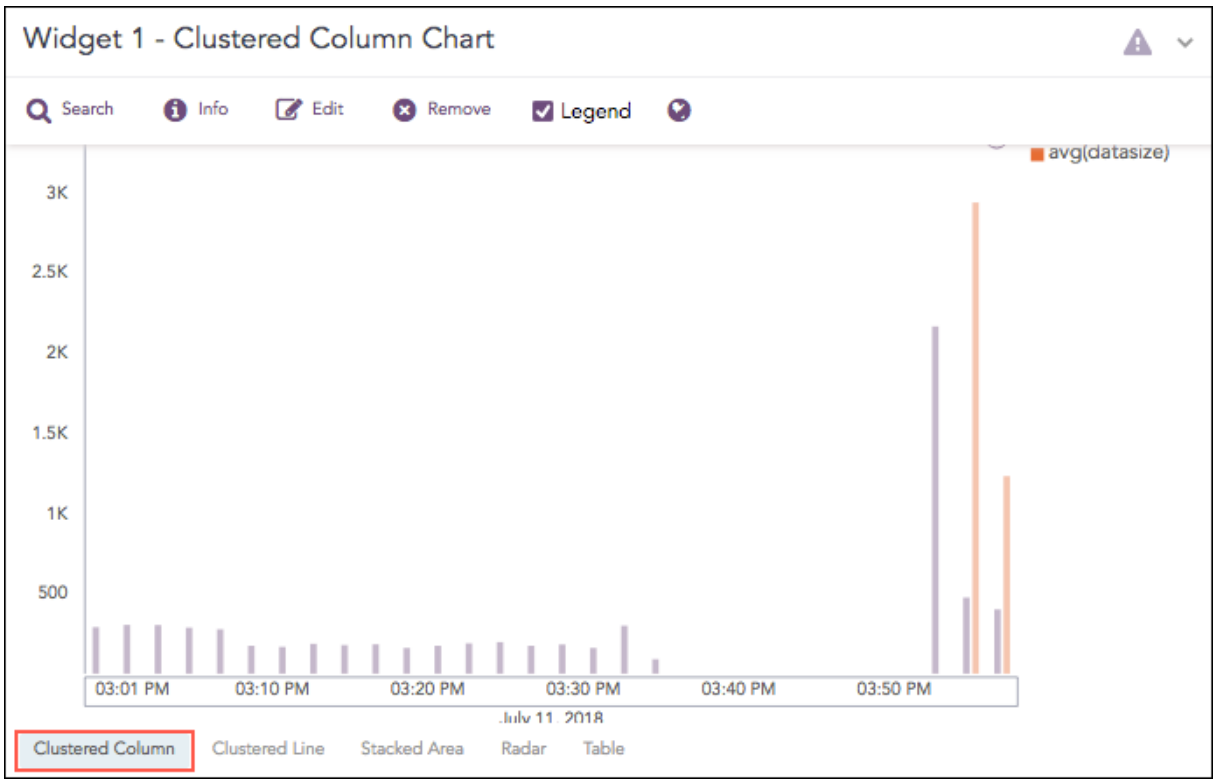


You can refer to Multiple Aggregation with Grouping for more details.

Clustered Column Chart

The Clustered Column chart is a type of Column Chart which allows you to display multiple quantitative variables.

Unlike a standard Column Chart, where only one variable is used to mark x-axis, a Clustered Column chart uses multiple variables on the x-axis with a different color for each variable.



The following query gives the output shown above.

```
| timechart count(), avg(datasize)
```

Response Types Supported

The **Clustered Column** chart supports four aggregation response types for representation of search results in the visualization. They are:

Response Type	General Syntax
Multiple Aggregation without Grouping	`| chart count(), avg(datasize)`
Multiple Aggregation with Grouping	`| chart aggregation_parameter1, aggregation_parameter2 by grouping_parameter1, grouping_parameter2, ...,grouping_parametern`
Timechart Multiple Aggregation without Grouping	`| timechart aggregation_parameter1, aggregation_parameter2, .... aggregation_parametern`

Response Type	General Syntax
Timechart Multiple Aggregation with Grouping	`| timechart aggregation_parameter1, aggregation_parameter2, ...., aggregation_parametern by grouping_parameter1, grouping_parameter2, ...., grouping_parametern`

Multiple Aggregation without Grouping

For the Multiple Aggregation without Grouping response type, the x-axis represents the different aggregation parameter, and y-axis contains the scale that denotes the value of the aggregation parameter.

Example:

```
| chart max(sent_datasize), max(received_datasize)
```

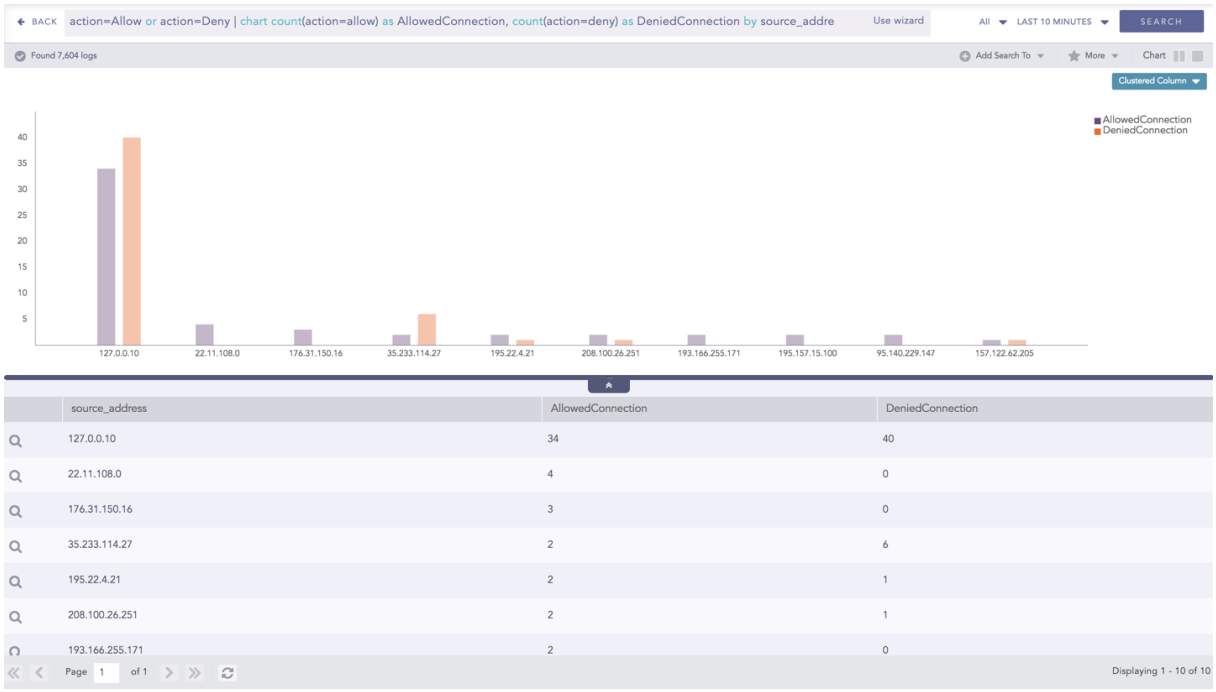
You can refer to Multiple Aggregation without Grouping for more details.

Multiple Aggregation with Grouping

For the Multiple Aggregation with Grouping response type, the x-axis contains the values of grouping parameter(s) with a vertical bar for each aggregation parameter. The height of the bar determines the value of the aggregation parameter for the specific value of a grouping parameter. The y-axis contains the scale that denotes the value of the aggregation parameter.

Example:

```
action=Allow or action=Deny | chart count(action=allow) as
AllowedConnection, count(action=deny) as DeniedConnection by source_
address order by count(action=allow), count(action=deny) desc limit 10
```



You can refer to Multiple Aggregation with Grouping for more details.

Timechart Multiple Aggregation without Grouping

For the Timechart Multiple Aggregation without Grouping response type, the x-axis represents the different time buckets within the specified time range, and the y-axis contains the scale that denotes the value of the aggregation parameter. The bars indicate the different values of the aggregation parameter at different timestamps. The vertical length of a bar signifies its value of the aggregation parameter at that particular timestamp.

Example:

```
norm_id=WinDNSDHCP | timechart count(lease_address=drop) as Dropped, count
(lease_address=start) as Started, count(lease_address=end) as ENDED
```

You can refer to Timechart Multiple Aggregation without Grouping for more details.

Timechart Multiple Aggregation with Grouping

For the Timechart Multiple Aggregation with Grouping response type, the x-axis represents the different time buckets within the specified time range, and the y-axis contains the scale that denotes the value of the aggregation parameter. The bars indicate the different values of the aggregation parameter at different timestamps. The vertical length of a bar signifies its value of the aggregation parameter at that particular timestamp.

Example:

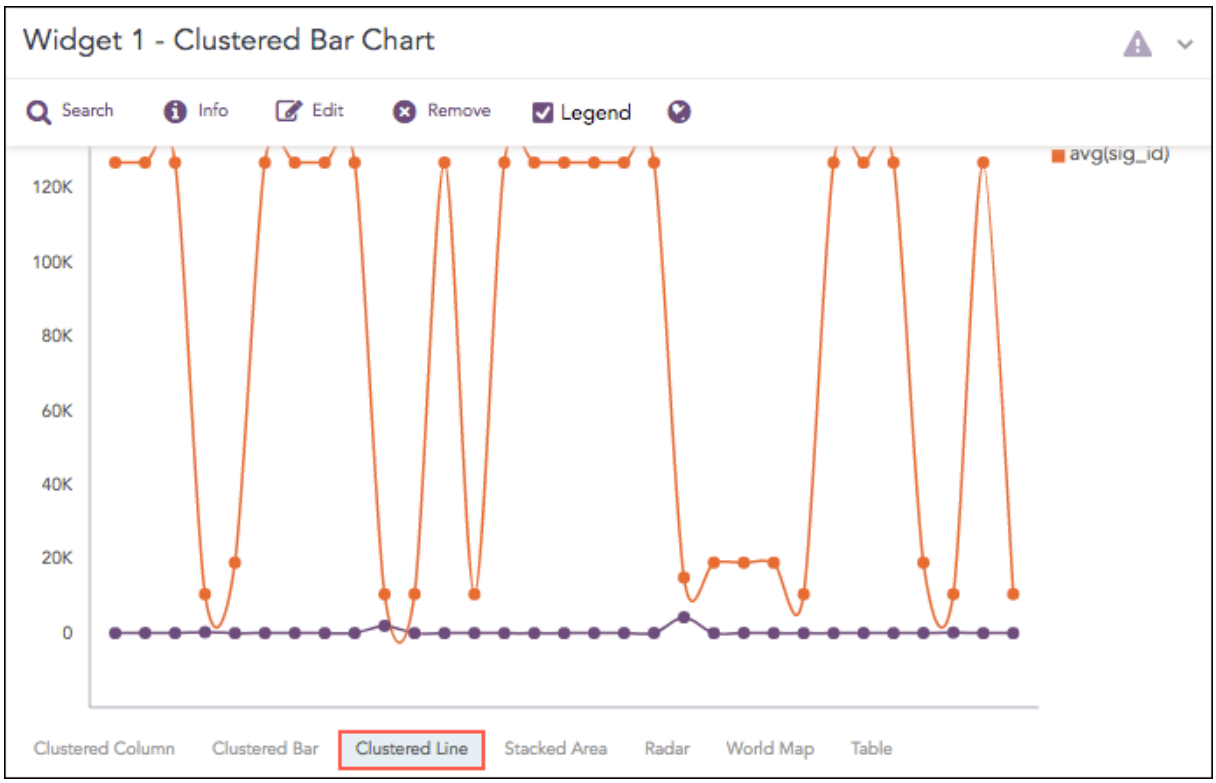
```
norm_id=WinDNSDHCP | timechart count(lease_address=drop) as Dropped, count
(lease_address=start) as Started, count(lease_address=end) as ENDED
```

Timestamp	Dangerous	Traffic
2018/07/12 05:26:58	0	0
2018/07/12 05:41:58	0	0
2018/07/12 05:56:58	359	355
2018/07/12 06:11:58	255	258
2018/07/12 06:26:58	0	0
2018/07/12 06:41:58	81	64
2018/07/12 06:56:58	385	337
2018/07/12 07:11:58	497	428
2018/07/12 07:26:58	502	436
2018/07/12 07:41:58	358	311
2018/07/12 07:56:58	114	85

You can refer to Timechart Multiple Aggregation with Grouping for more details.

Clustered Line Chart

The Clustered Line chart is an extension of the Line Chart in which multiple lines are used to represent values of different categories. Silimar to the Clustered Column Chart, in a Clustered Line chart the x-axis contains the values of the grouping parameters, and the y-axis contains the scale to measure the value of an aggregation parameter of the particular grouping parameter.



The following query gives the output shown above. :

```
| chart count(), avg(sig_id) by action
```

Response Types Supported

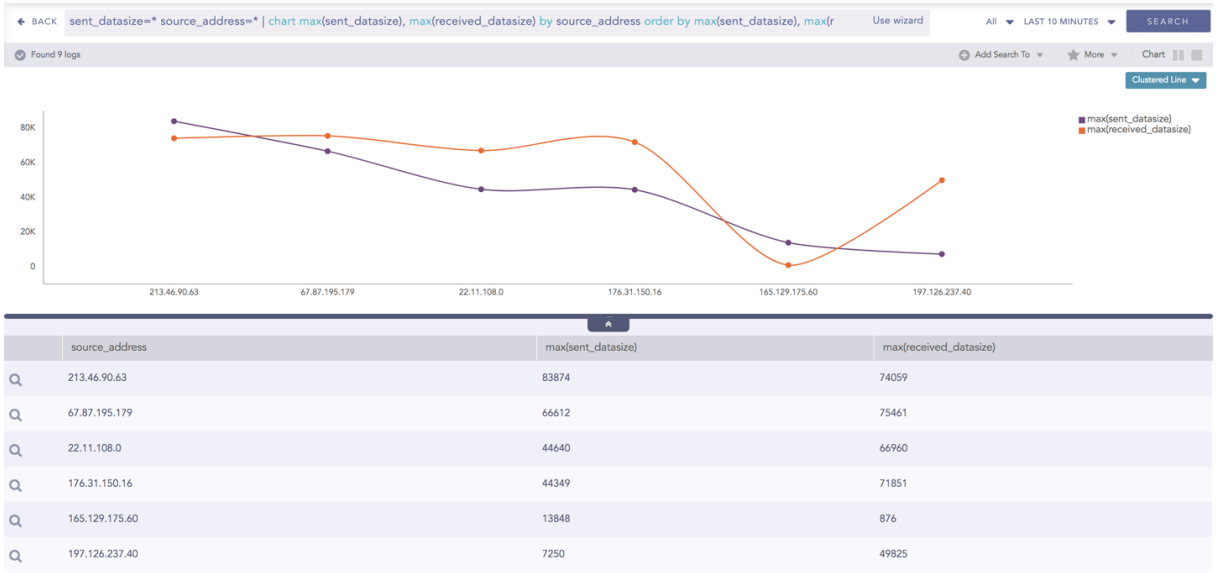
The **Clustered Line** chart supports three aggregation response types for representation of search results in the visualization. They are:

Response Type	General Syntax
Multiple Aggregation with Grouping	`| chart aggregation_parameter1, aggregation_parameter2 by grouping_parameter1, grouping_parameter2, ...,grouping_parametern`
Timechart Single Aggregation with Grouping	`| timechart aggregation_parameter by grouping_parameter1, grouping_parameter2, ....., grouping_parametern`
Timechart Multiple Aggregation without Grouping	`| timechart aggregation_parameter1, aggregation_parameter2, .... aggregation_parametern`

Multiple Aggregation with Grouping

Example:

```
sent_datasize=* source_address=* | chart max(sent_datasize), max(received_
datasize) by source_address order by max(sent_datasize), max(received_
datasize) desc limit 10
```



source_address	max(sent_datasize)	max(received_datasize)
213.46.90.63	83874	74059
67.87.195.179	66612	75461
22.11.108.0	44640	66960
176.31.150.16	44349	71851
165.129.175.60	13848	876
197.126.237.40	7250	49825

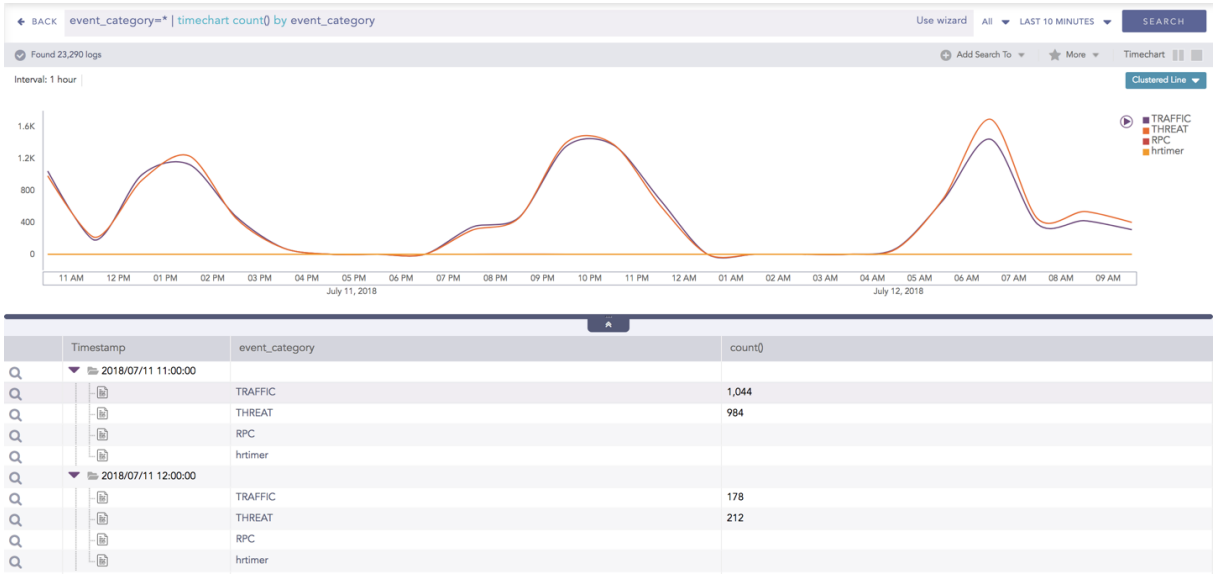
You can refer to Multiple Aggregation with Grouping for more details.

Timechart Single Aggregation with Grouping

In the Timechart Single Aggregation with Grouping, the y-axis represents the aggregation value for every grouping parameter, and the x-axis displays the value of the timestamps. Similarly, the lines represent the values of the grouping parameter(s).

Example:

```
event_category=* | timechart count() by event_category
```



Timestamp	event_category	count()
2018/07/11 11:00:00		
	TRAFFIC	1,044
	THREAT	984
	RPC	
	hrtimer	
2018/07/11 12:00:00		
	TRAFFIC	178
	THREAT	212
	RPC	
	hrtimer	

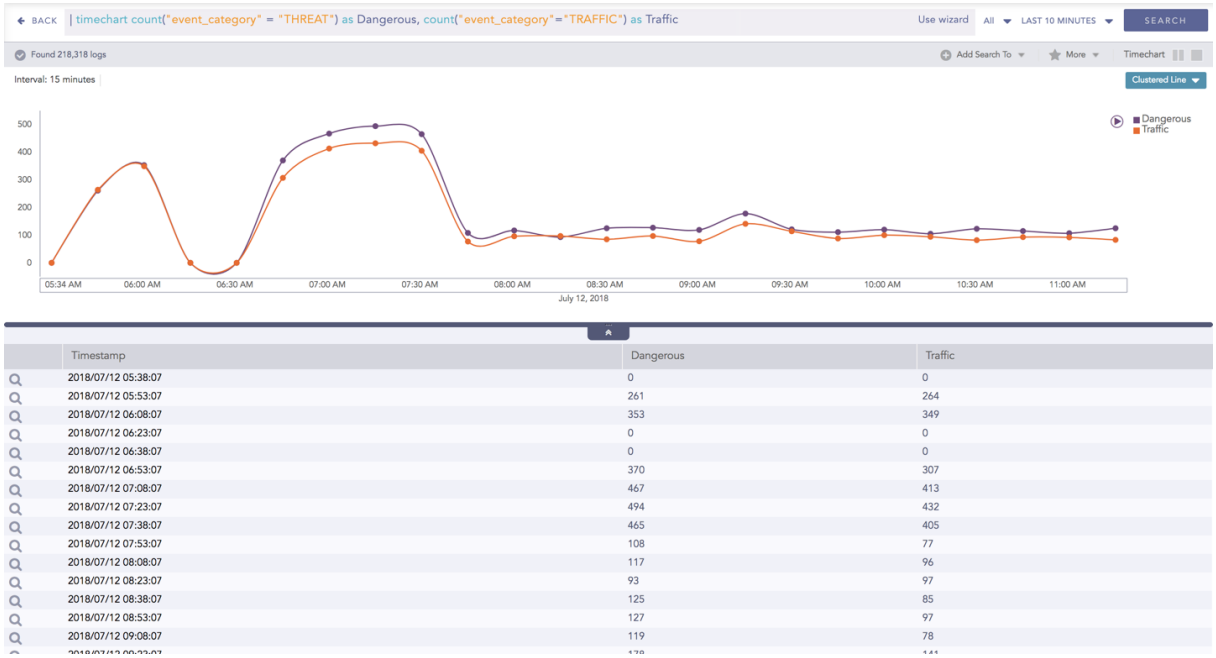
You can refer to Timechart Single Aggregation with Grouping for more details.

Timechart Multiple Aggregation without Grouping

Alike to the Clustered Column chart, the y-axis represents values of the aggregation parameter, and the x-axis displays the value of the timestamps. Similarly, the lines represent the values of the aggregation parameters at a particular timestamp.

Example:

```
| timechart count("event_category" = "THREAT") as Dangerous, count("event_
category" ="TRAFFIC") as Traffic
```



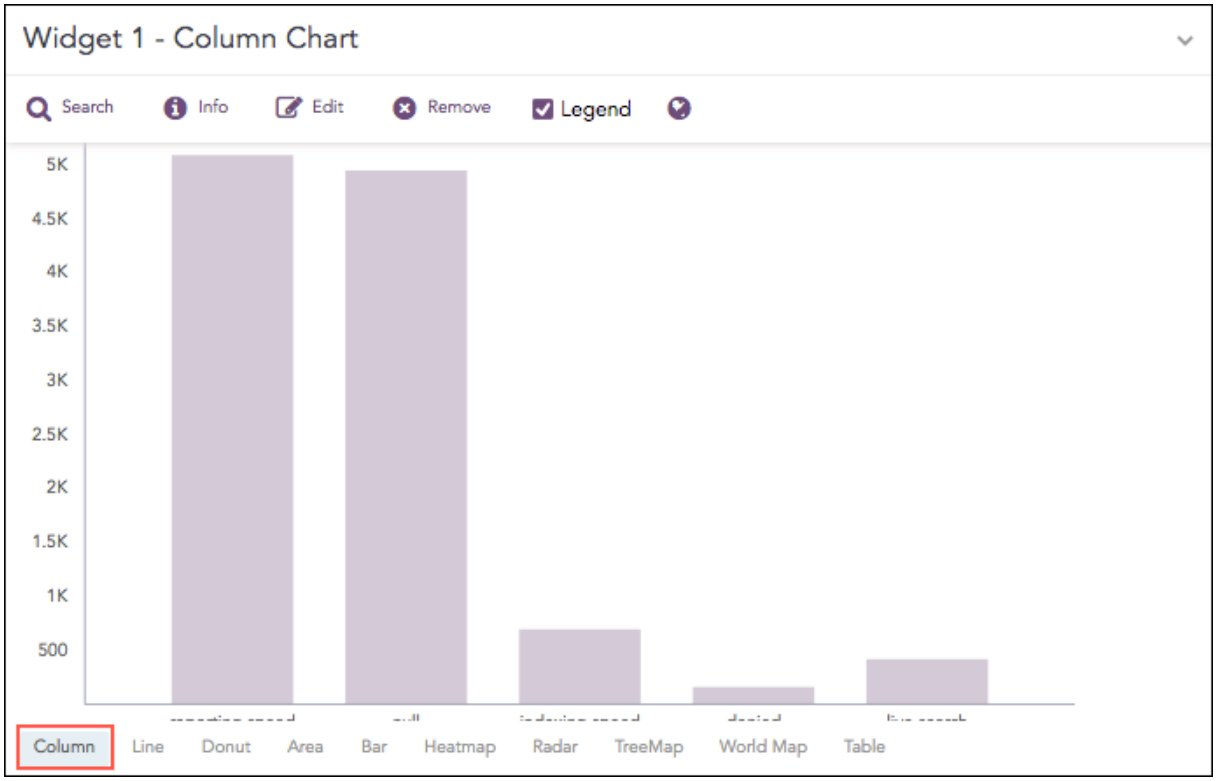
Timestamp	Dangerous	Traffic
2018/07/12 05:38:07	0	0
2018/07/12 05:53:07	261	264
2018/07/12 06:08:07	353	349
2018/07/12 06:23:07	0	0
2018/07/12 06:38:07	0	0
2018/07/12 06:53:07	370	307
2018/07/12 07:08:07	467	413
2018/07/12 07:23:07	494	432
2018/07/12 07:38:07	465	405
2018/07/12 07:53:07	108	77
2018/07/12 08:08:07	117	96
2018/07/12 08:23:07	93	97
2018/07/12 08:38:07	125	85
2018/07/12 08:53:07	127	97
2018/07/12 09:08:07	119	78

You can refer to Timechart Multiple Aggregation without Grouping for more details.

Column Chart

The Column Chart is a vertical bar graph that represents categorical data in rectangular bars with heights proportional to the values that they represent.

The Column Chart shows comparisons among discrete categories. It is a two-dimensional graph in which one axis of the graph shows the specific groups being compared and another one represents the measured value.



The following query gives the output shown above.

```
| chart count() by action limit 5
```

Response Types Supported

The **Column** chart supports two aggregation response types for representation of search results in the visualization. They are:

Response Type	General Syntax
Single Aggregation with Grouping	`| chart aggregation_parameter by grouping_parameter1, grouping_parameter2, ........, grouping_parametern`
Timechart Single Aggregation without Grouping	`| timechart aggregation_parameter`

Single Aggregation with Grouping

In the Single Aggregation with Grouping response type, the x-axis of a Column chart represents the values of the grouping parameter(s) whereas the y-axis represents the values of the aggregation parameter.

Example:

```
severity=* | chart count() by severity order by count() desc limit 5
```

	severity	count()
	6	236
	high	47
	low	41
	critical	41
	medium	36

Page 1 of 1 — Displaying 1 - 5 of 5

You can refer to Single Aggregation with Grouping for more details.

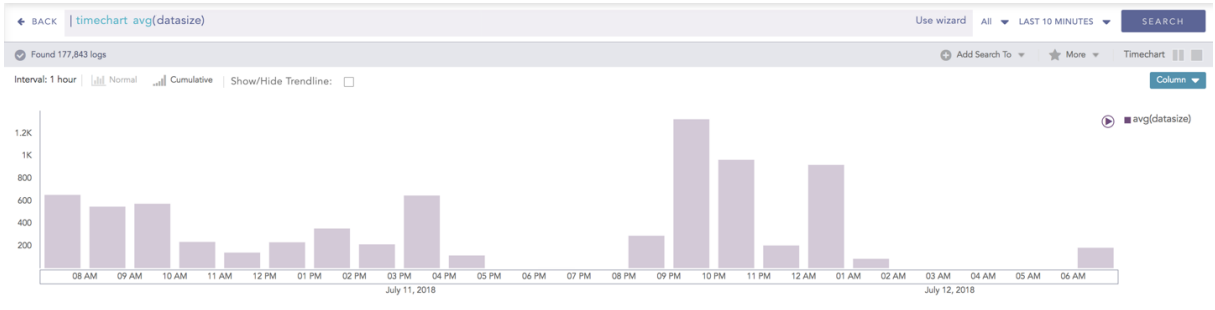
Timechart Single Aggregation without Grouping

In the Timechart Single Aggregation without Grouping response type, the x-axis of the Column chart represents the value of timestamps whereas the y-axis represents the values of the aggregation parameter.

Each bar represents the value of the aggregation parameter in a given **Interval**. The **Interval** is calculated automatically as per the time range selected in the **Search Bar**. The value of the **Interval** is displayed on the extreme left of the container.

Example:

```
| timechart avg(datasize)
```



	Timestamp	avg(datasize)
	2018/07/11 08:00:00	651.73
	2018/07/11 09:00:00	547.56
	2018/07/11 10:00:00	573.02
	2018/07/11 11:00:00	233.98
	2018/07/11 12:00:00	139.11
	2018/07/11 13:00:00	230.86
	2018/07/11 14:00:00	353.31
	2018/07/11 15:00:00	212.69
	2018/07/11 16:00:00	646.15
	2018/07/11 17:00:00	113.73
	2018/07/11 18:00:00	0
	2018/07/11 19:00:00	0.36
	2018/07/11 20:00:00	0
	2018/07/11 21:00:00	288.8
	2018/07/11 22:00:00	1323.32
	2018/07/11 23:00:00	963.51

You can refer to Timechart Single Aggregation without Grouping for more details.

Day/Hour Heatmap Chart

Heatmaps are used to visualize individual values contained in a matrix and represent them using different shades of a single color.

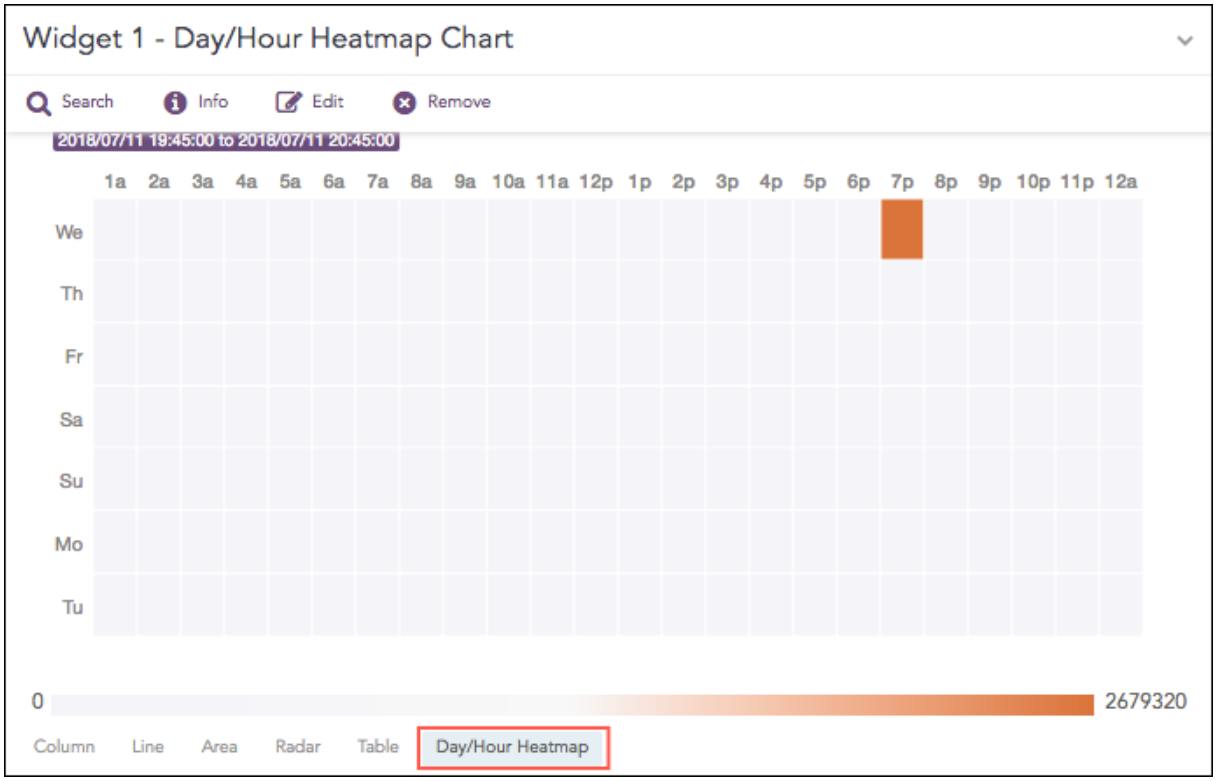
The Day/Hour Heatmap is an extension of a regular heatmap in which results are displayed in the day/hour format. It has seven rows and 24 columns. Each row represents a day of the week and each column represents an hour of the day. Therefore, each cell represents a single hour of a particular day.

The query format for the Day/Hour Heatmap is:

```
| timechart aggregation_parameter1 every 1 hour
```

Example:

```
| timechart sum(datasize) as TotalDatasize every 1 hour
```



Response Types Supported

The **Day/Hour Heatmap** chart supports a single aggregation response types for representation of search results in the visualization. It is :

Response Type	General Syntax
Timechart Single Aggregation without Grouping	`| timechart aggregation_parameter`

Timechart Single Aggregation without Grouping

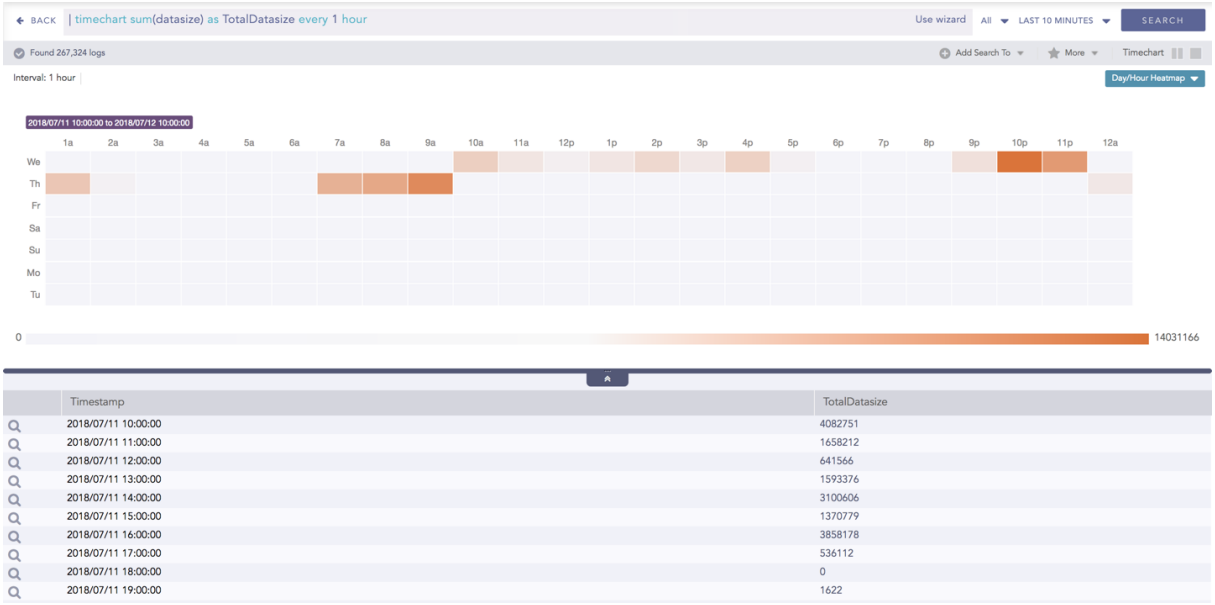
The Day/Hour Heatmap only works for the Timechart Single Aggregation with Grouping response type with **every 1 hour** suffixed to the query.

The values of the aggregation parameter are displayed in the cells as per their timestamps.

The intensity of the color is dependent upon the relative value of the aggregation parameters.

Example:

```
| timechart sum(datasize) as TotalDatasize every 1 hour
```

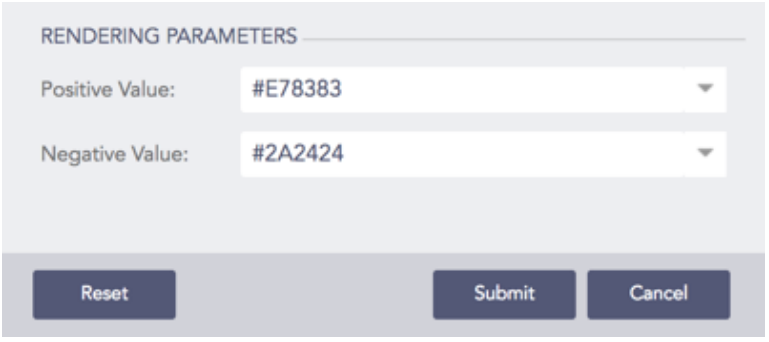


When the selected time range is more than a week, a slider appears on the right end of the container that allows the user to slide over the particular days.

You can refer to Timechart Single Aggregation without Grouping for more details.

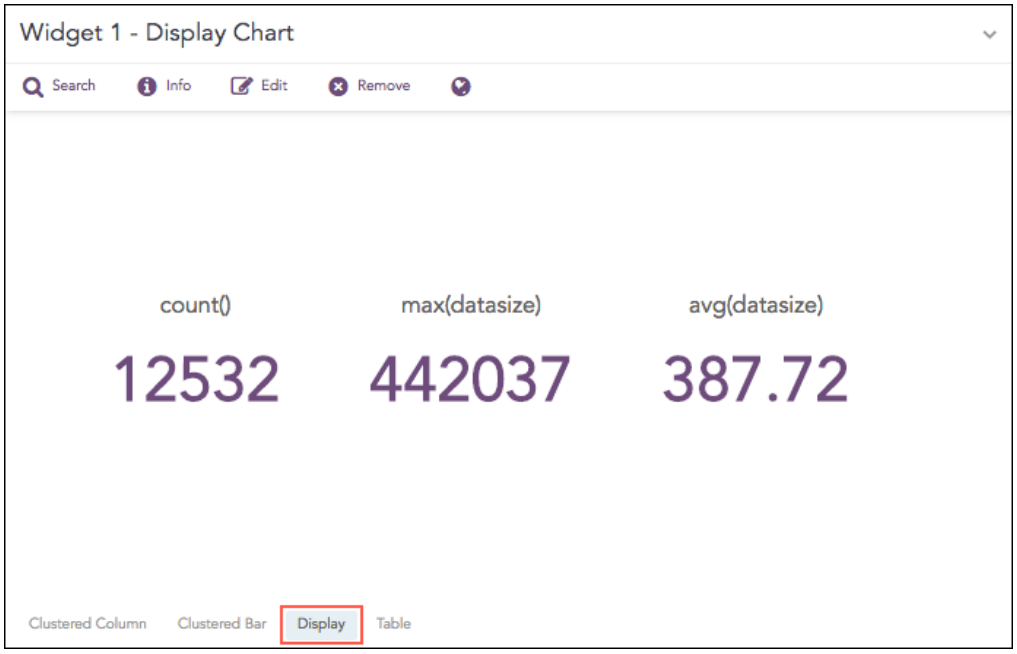
Rendering Parameters

You can assign custom colors to the Day/Hour heatmap for both positive and negative values. SLS uses the selected color to represent the maximum value of the data obtained, and lesser values have the same color with linear transparency.



Display Chart

The **Display** format shows the value of the aggregation parameter in the container.



The following query gives the output shown above.

```
| chart count(), max(datasize), avg(datasize)
```

To view the search results in display format, select **Display** from the drop-down at the top-right corner of the **Search Result** page.

Response Types Supported

The Display chart supports three aggregation response types for representation of search results in the visualization. They are:

Response Type	General Syntax
Single Aggregation without Grouping	`| chart aggregation_parameter`
Single Aggregation with Grouping	`| chart aggregation_parameter by grouping_parameter`
Multiple Aggregation without Grouping	`| chart count(), avg(datasize)`

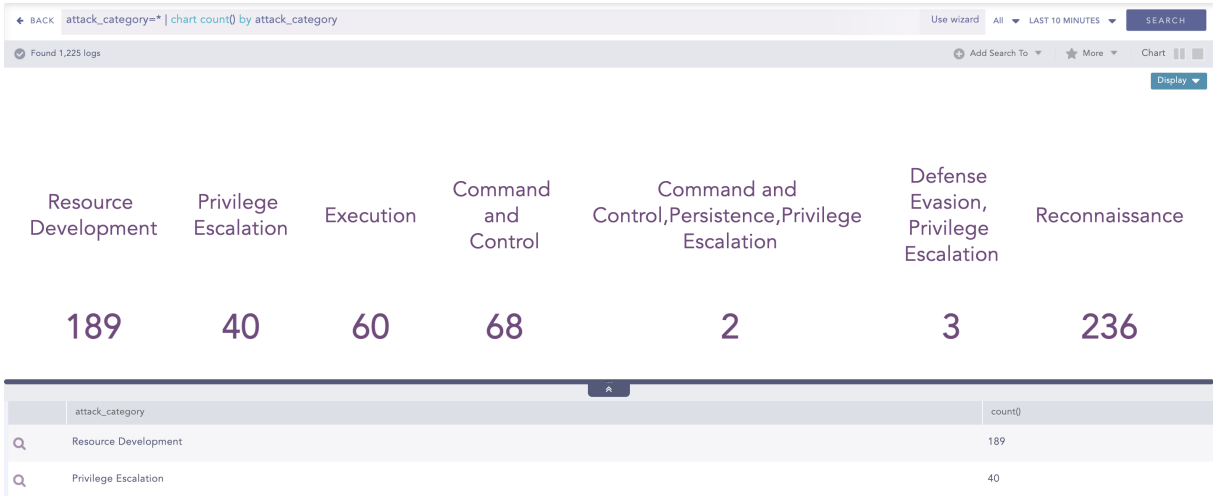
Single Aggregation without Grouping

To view the search results in display format, select **Display** from the drop-down on the top-right corner of the **Search Result** page.

> **NOTE**
> By default, SLS renders the display format for all queries of the **Single Aggregation without Grouping** type.

You can refer to Single Aggregation without Grouping for more details.

Single Aggregation with Grouping



To view the search results in display format, select **Display** from the drop-down on the top-right corner of the **Search Result** page.

> **NOTE**
> The Display chart is available in **Search**, **Dashboards**, and **Search Templates**.

You can refer to Single Aggregation with Grouping for more details.

Multiple Aggregation without Grouping

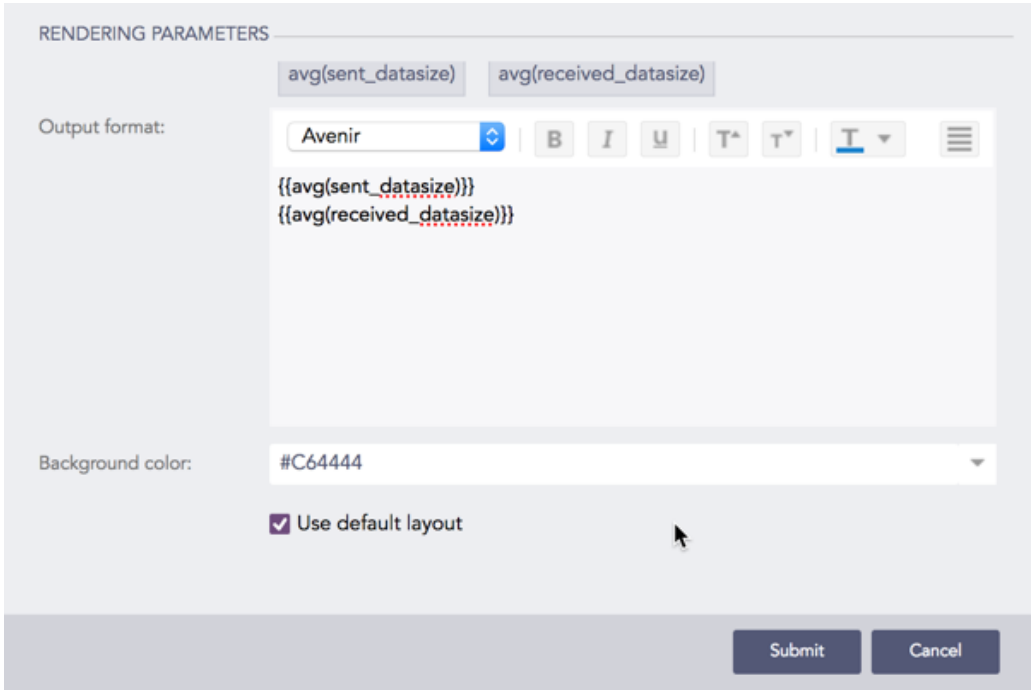
For Multiple Aggregation without Grouping response type, the value of the first aggregation parameter is displayed in the container.

To view the search results in display format, select **Display** from the drop-down on the top right corner of the Search result page.

You can refer to Multiple Aggregation without Grouping for more details.

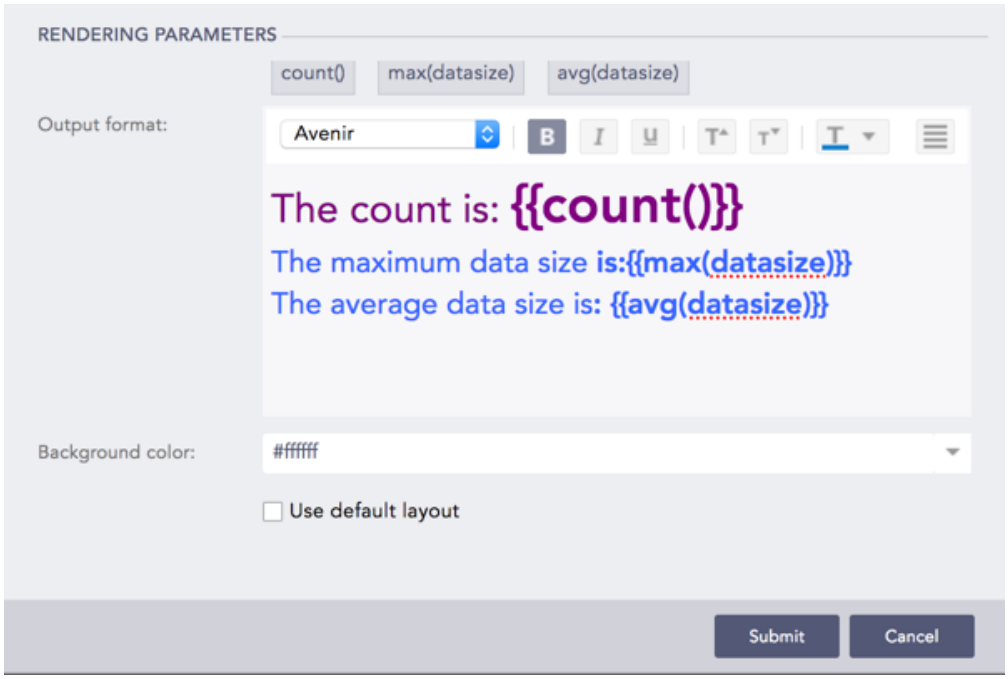
Rendering Parameters

Click the **Settings** icon at the top-right corner to change the **Rendering Parameters**.

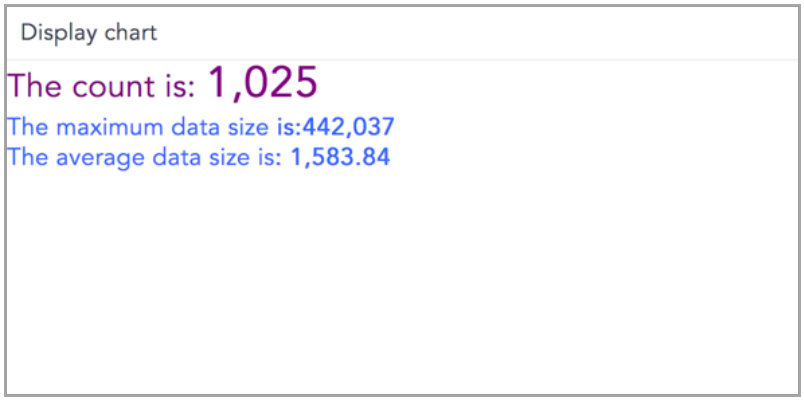


You can choose the output format, font and the background color from the rendering parameter section.

Customize the output of the display by configuring the **Output format**. In this section, you can choose the fonts and color of the result. Additionally, you can use the display template similar to "jinja" to customize the search result.



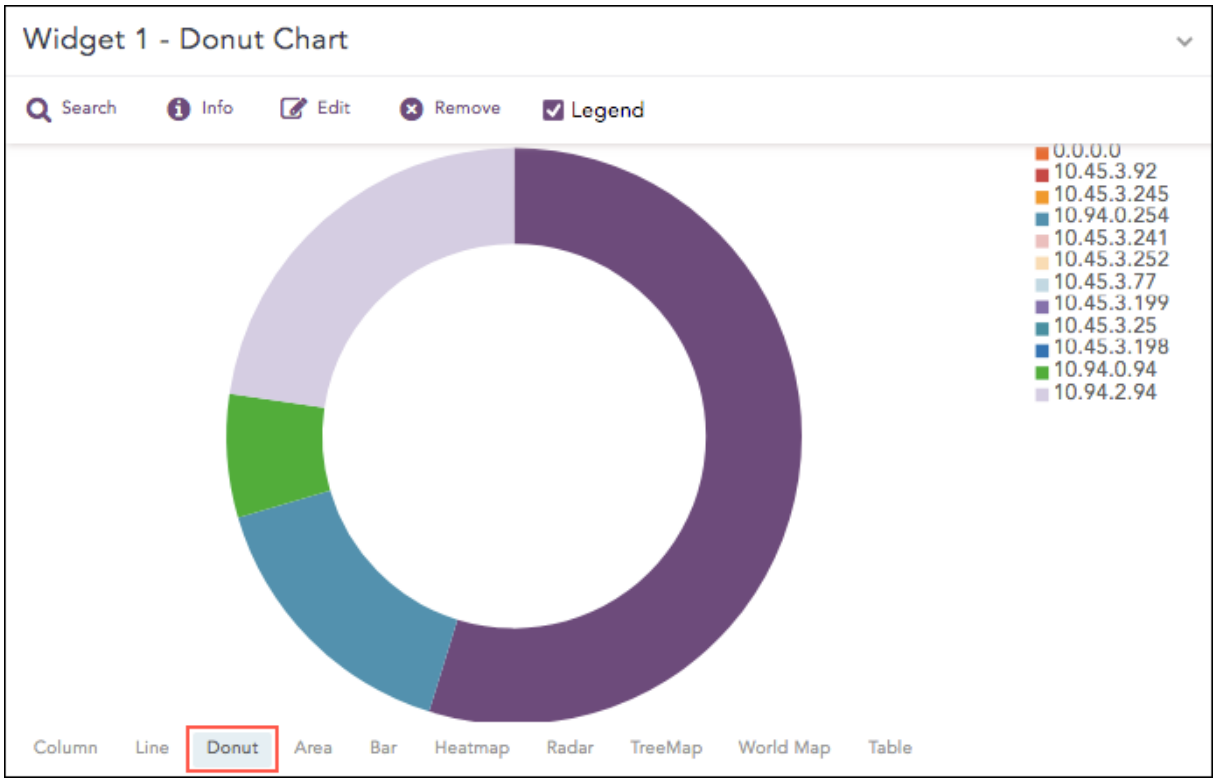
The result of the configuration looks like:



NOTE
The **Output Format** section is disabled when you select the default layout.

Donut Chart

The Donut Chart shows the data distribution based on the length of its arc. It was introduced in SLS to replace the Pie Chart. The reason for this is that pie charts can be hard to interpret as they focus on the proportional areas of the slices. Donut charts de-emphasize the use of area and focus on the lengths of the arcs of each individual element



The following query gives the output shown above.

```
source_address=* | chart sum(datasize) as Datasize by source_address
```

Response Types Supported

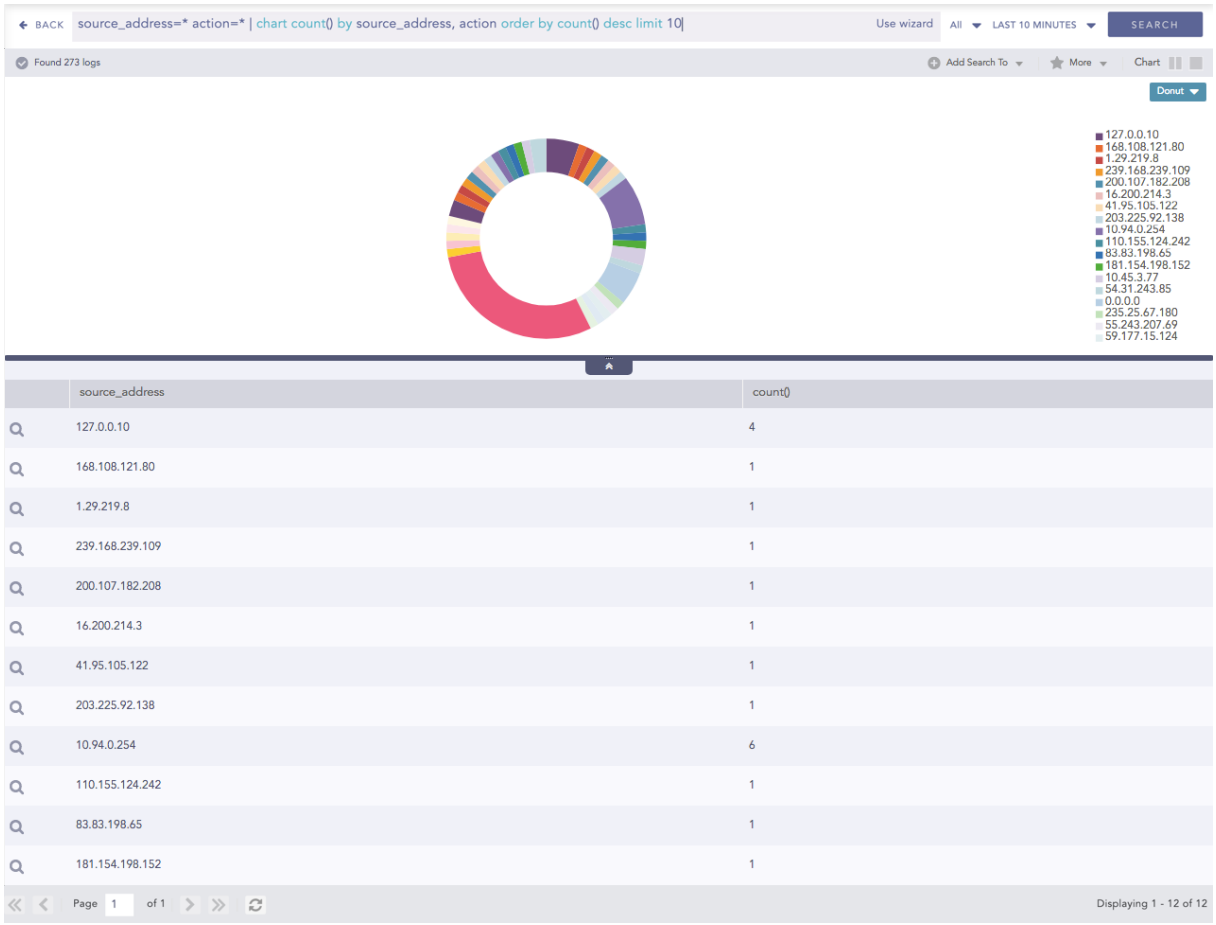
The **Donut** chart supports a single aggregation response types for representation of search results in the visualization. It is :

Response Type	General Syntax
Single Aggregation with Grouping	`| chart aggregation_parameter by grouping_parameter1, grouping_parameter2, ........, grouping_parametern`

Single Aggregation with Grouping

Example:

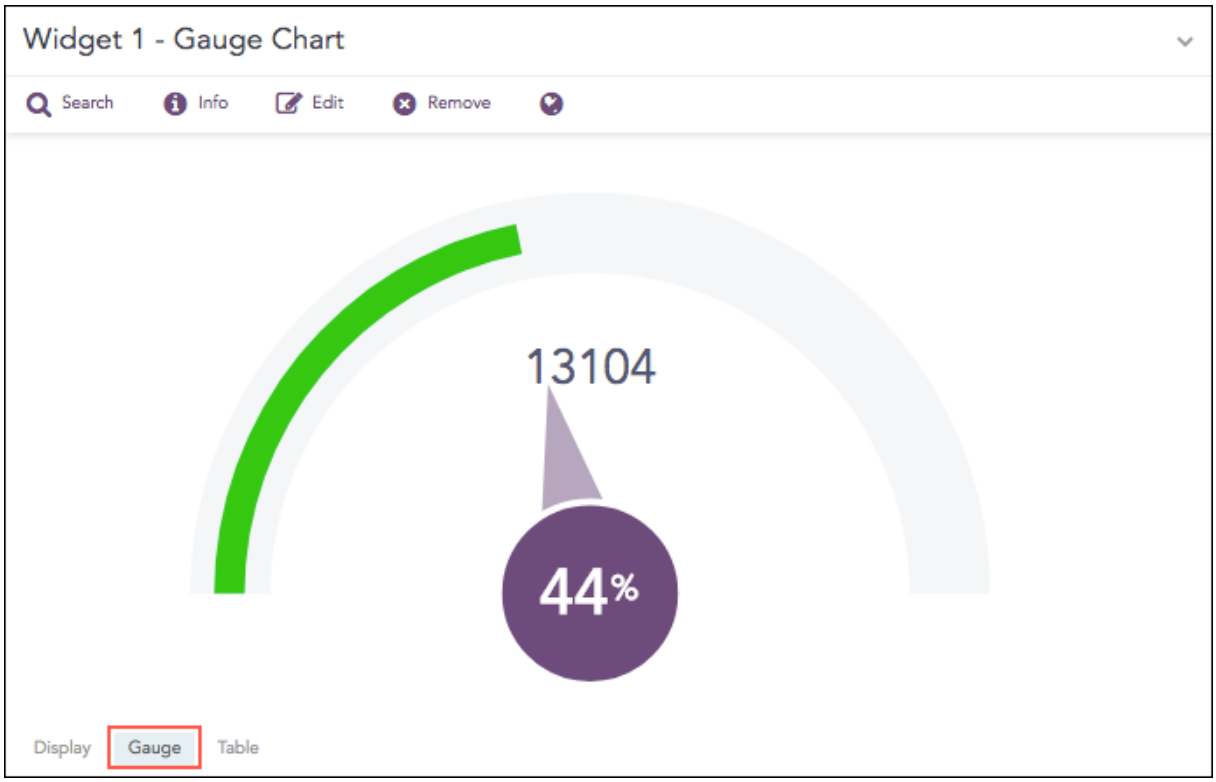
```
source_address=* | chart count() by source_address
```



You can refer to Single Aggregation with Grouping for more details.

Gauge Chart

Gauge chart, also known as speedometer chart, uses a single needle to show the information as a reading on a dial. The graph is used to visualize percentage values as well as a fixed range of data.



The following query gives the output shown above.

```
| chart count()
```

The value of the aggregation parameter determines the value pointed by the needle. You can configure the maximum value of the dial from **Max value** while rendering parameters. When a value of the aggregation parameter is equal to, or greater than the **Max value**, the percentage value of the needle is displayed as 100%.

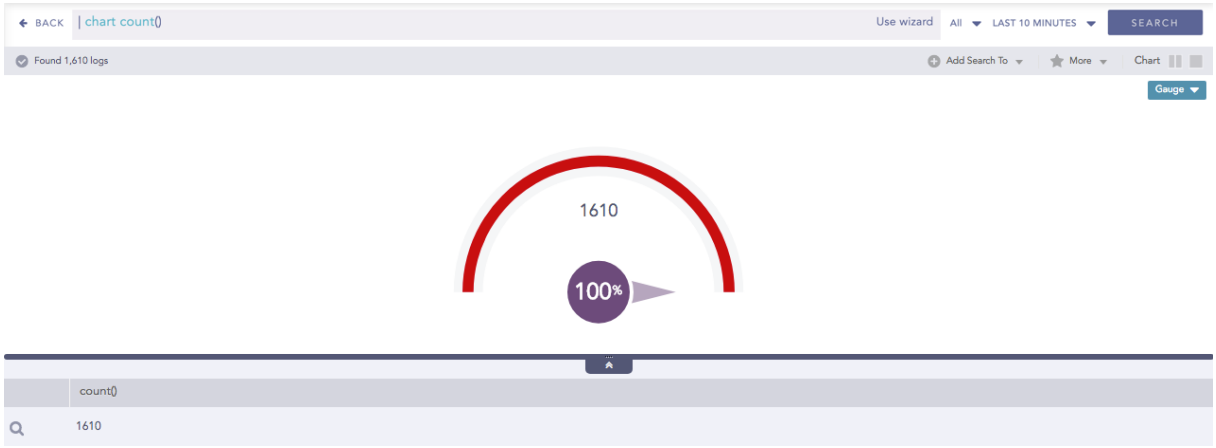
Three different colors, green, yellow, and red are used to represent the limits for the data being depicted in the gauge. By default, the green, yellow, and red colors represent the low, mid, and high range of values respectively. However, you can configure the threshold value (in percentage) to display the dial in the yellow and red colors.

Response Types Supported

The **Gauge** chart supports a single aggregation response types for representation of search results in the visualization. It is:

Response Type	General Syntax
Single Aggregation without Grouping	`| chart aggregation_parameter`

Single Aggregation without Grouping

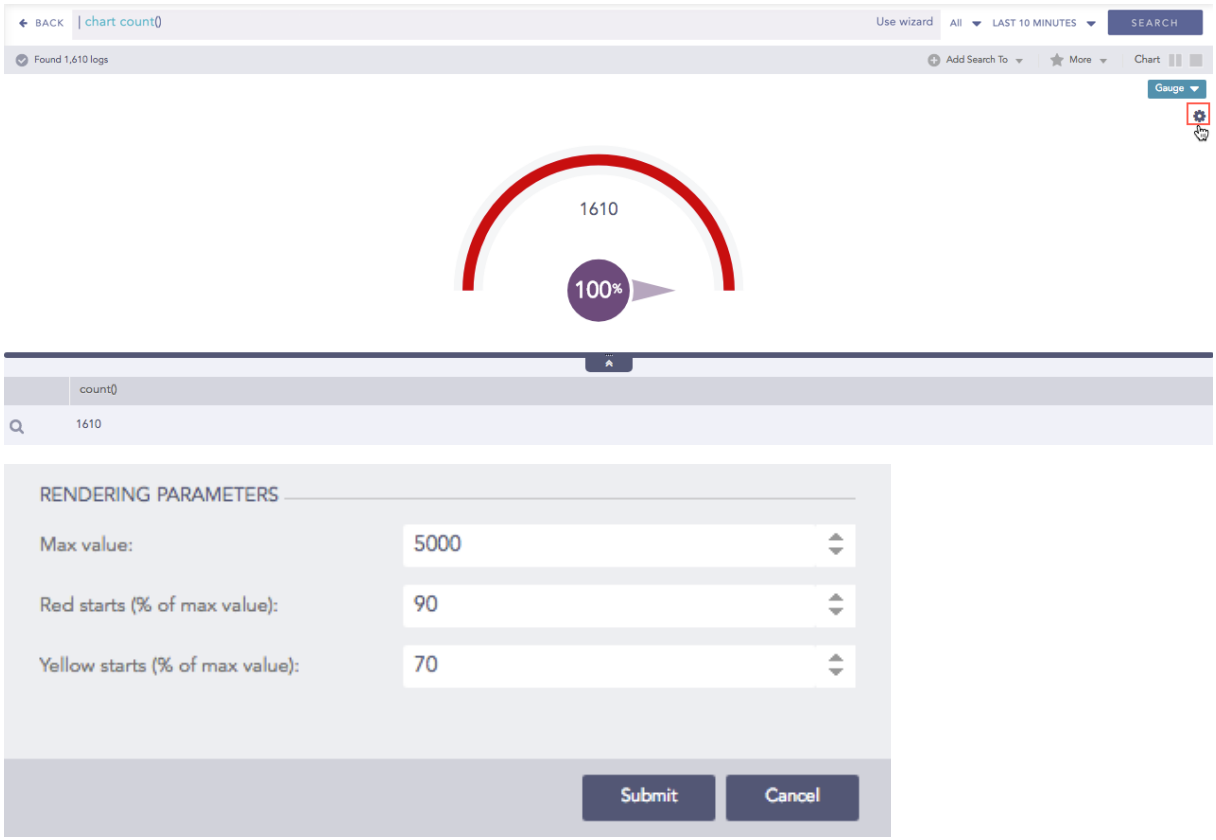


To view the search results in a Gauge chart, select **Gauge** from the drop-down menu on the top-right corner of **Search Result** page.

You can refer to Single Aggregation without Grouping for more details.

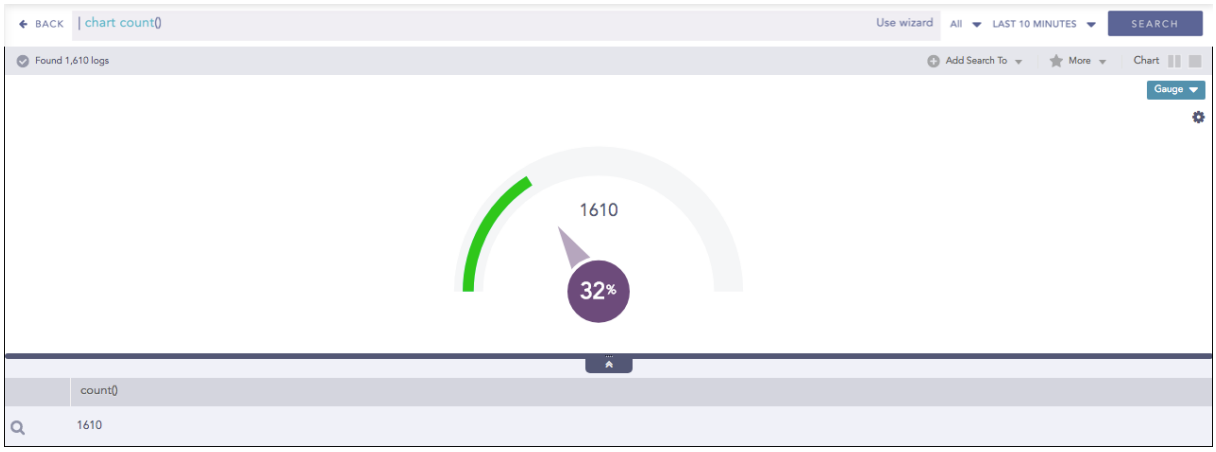
Rendering Parameters

Click the settings icon on the right side of the chart container to open the **Rendering Parameters** panel.



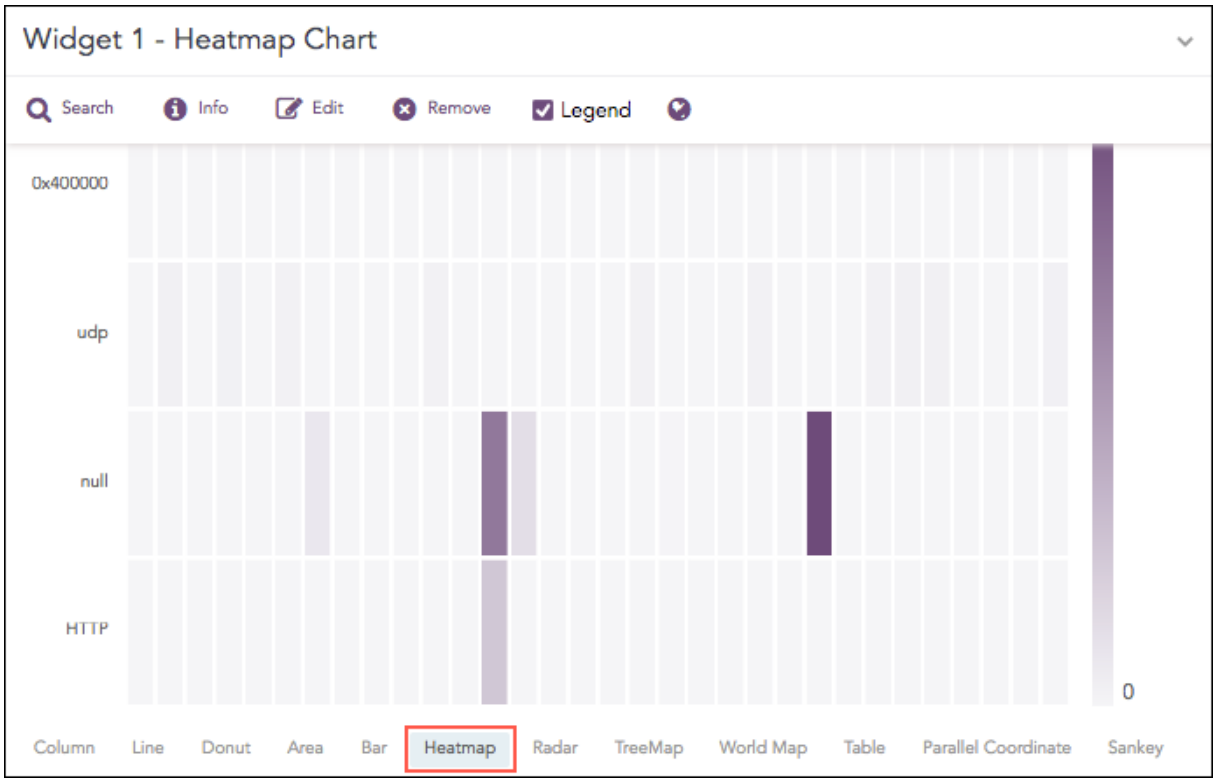
You can specify the threshold value for the red and yellow colors in the **Red Starts** and **Yellow Starts** configuration fields on the **Rendering Parameters** panel.

NOTE
The value of **Red starts** is 90% and **Yellow starts** is 70% by default.



Heatmap Chart

Heatmap visualizes individual values in a matrix and represents them through different color shades based on their intensity. Use it to analyze the differences across multiple variables, reveal patterns, and detect correlations between them.



The following query gives the output shown above.

```
| chart count() by action, protocol
```

Response Types Supported

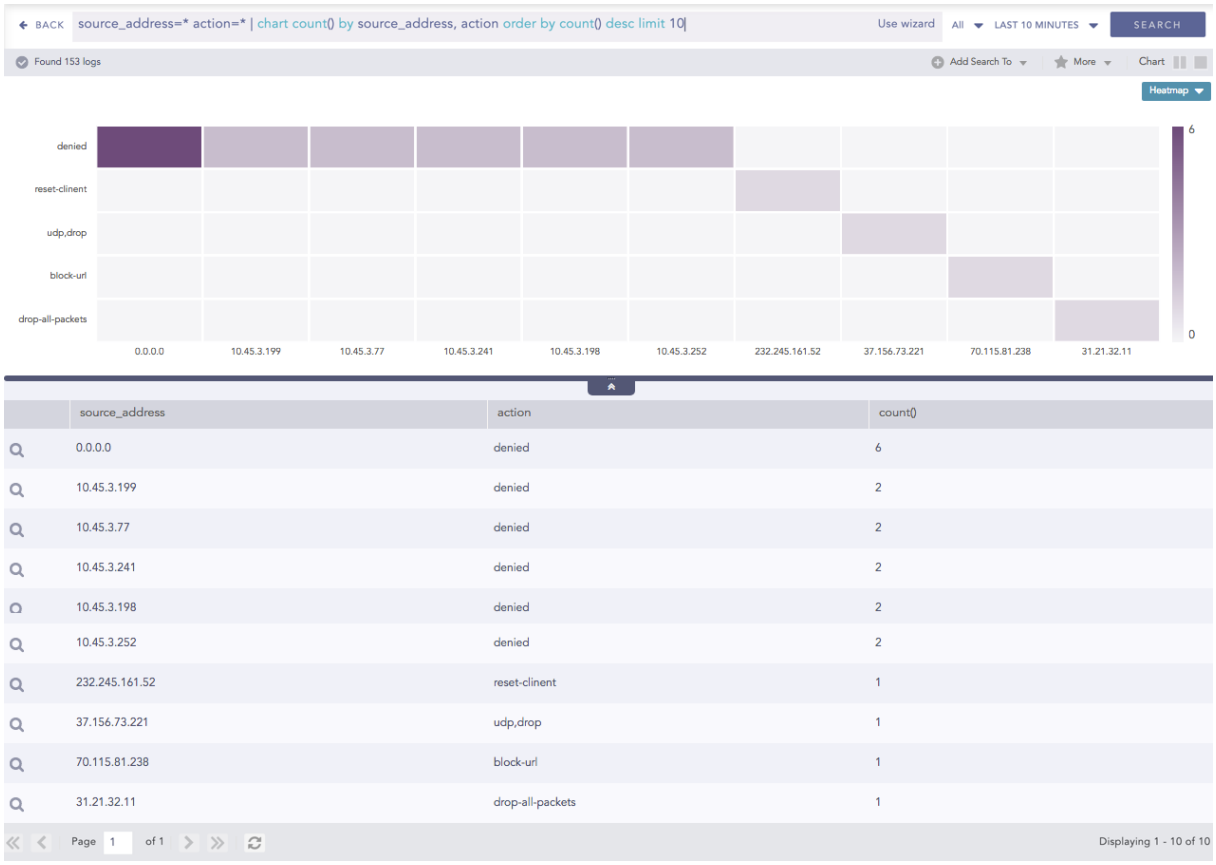
The **Heatmap** chart supports a single aggregation response types for representation of search results in the visualization. It is :

Response Type	General Syntax
Single Aggregation with Grouping	`| chart aggregation_parameter by grouping_parameter1, grouping_parameter2, ........, grouping_parametern`

Single Aggregation with Grouping

Example:

```
source_address=* action=* | chart count() by source_address, action order by count() desc limit 10
```



You can refer to Single Aggregation with Grouping for more details.

Rendering Parameters

Click the settings icon at the top-right corner of the heatmap chart to open a dialog box. The dialog box allows you to configure the rendering parameters of the Heatmap chart.

The Rendering Parameters such as **X-axis Group**, **Positive Value**, and **Negative Value** provide a custom settings option to view data in different formats.

By default, the first grouping parameter of the query is assigned to the X-axis of the Heatmap. For example, the grouping parameter **source_name** is assigned in the X-axis of the Heatmap for the query:

```
| chart count() by source_name, action
```

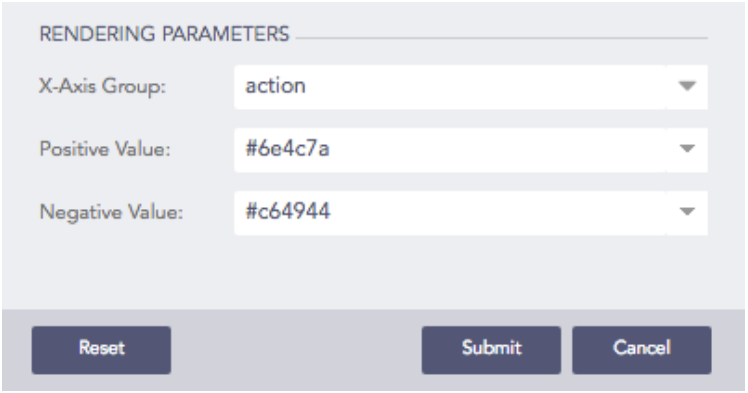
However, by selecting a value from the drop-down menu of the **X-axis Group**, you can choose the grouping parameter to be placed on the X-axis of the chart. For example,

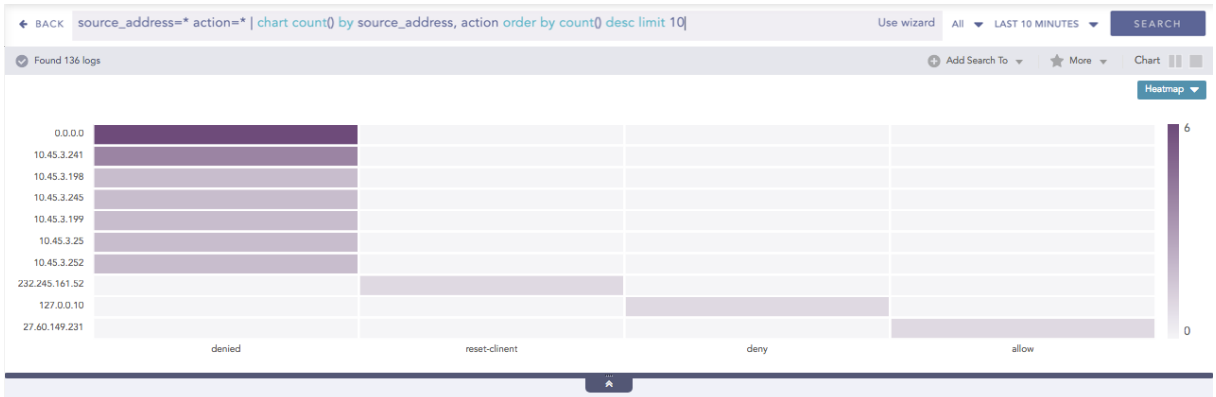
```
| chart count() by source_name, action
```

The query above contains two grouping parameters: **source_name** and **action**. If you choose **action** for x-axis, **source_name** is shown on y-axis. The **count()** value is represented according to the transparency level of the chosen cell color.

NOTE
If a query contains three or more than three grouping parameters, and you choose to keep **grouping_parameter_1** on the x-axis, then the combination of **grouping_parameter_2**, ..., **grouping_parameter_n** is shown on the y-axis.

Furthermore, you can assign custom colors to the Heatmap for both positive and negative values. SLS uses the selected color to represent the maximum value of the data obtained, and lesser values have the same color with linear transparency.





	source_address	action	count()
	0.0.0.0	denied	6
	10.45.3.241	denied	4
	10.45.3.198	denied	2
	10.45.3.245	denied	2
	10.45.3.199	denied	2
	10.45.3.25	denied	2
	10.45.3.252	denied	2
	232.245.161.52	reset-clinent	1
	127.0.0.10	deny	1
	27.60.149.231	allow	1

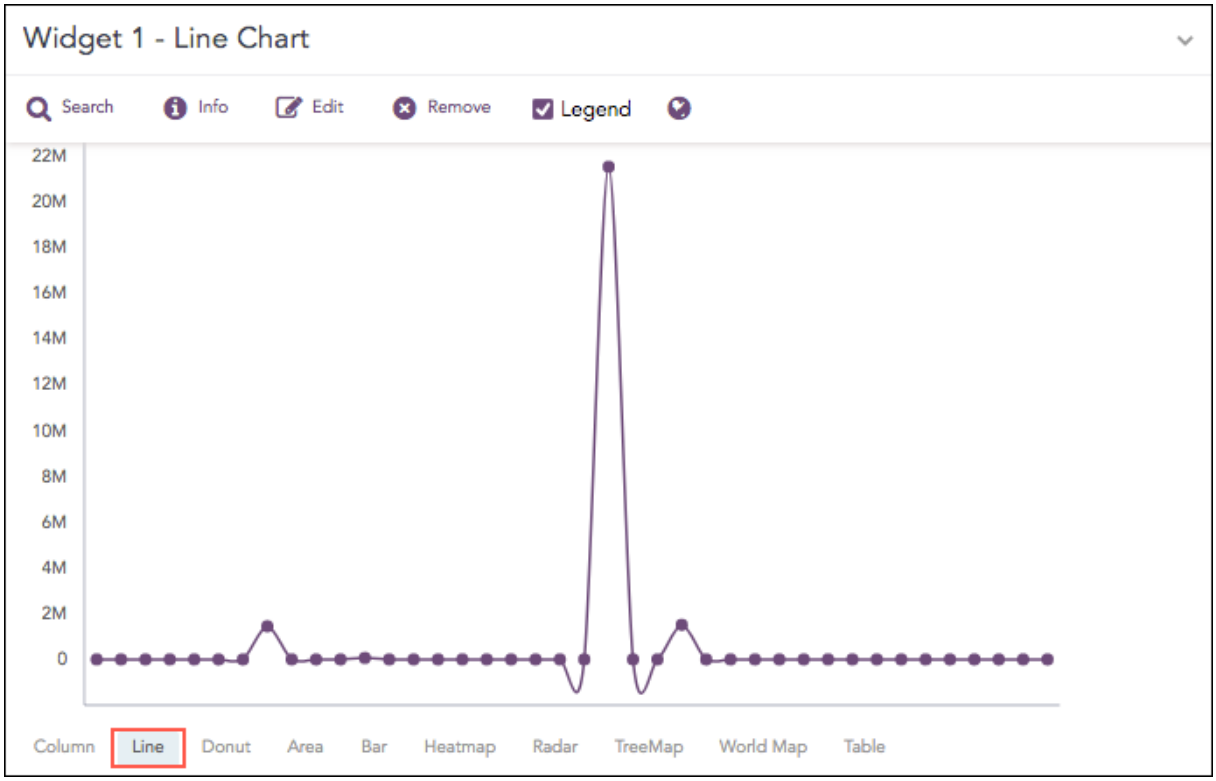
Page 1 of 1

Displaying 1 - 10 of 10

Line Chart

The Line chart displays information as a series of data points called markers. The markers are connected to each other by a line.

The Line chart consists of two axes, in which x-axis contains the value of the grouping parameter(s) and the y-axis contains the values of the aggregation parameter. It is similar to a Column chart, except that, a Column chart usually displays discrete values, whereas a line chart visualizes a trend in continuous data.



The following query gives the output shown above.

```
source_address=* | chart sum(datasize) as Datasize by source_address
```

Response Types Supported

The **Line** chart supports two aggregation response types for representation of search results in the visualization. They are:

Response Type	General Syntax
Single Aggregation with Grouping	`| chart aggregation_parameter by grouping_parameter1, grouping_parameter2, ........, grouping_parametern`
Timechart Single Aggregation without Grouping	`| timechart aggregation_parameter`

Single Aggregation with Grouping

Example:

```
severity=* | chart count() by severity
```

You can refer to Single Aggregation with Grouping for more details.

Timechart Single Aggregation without Grouping

Example:

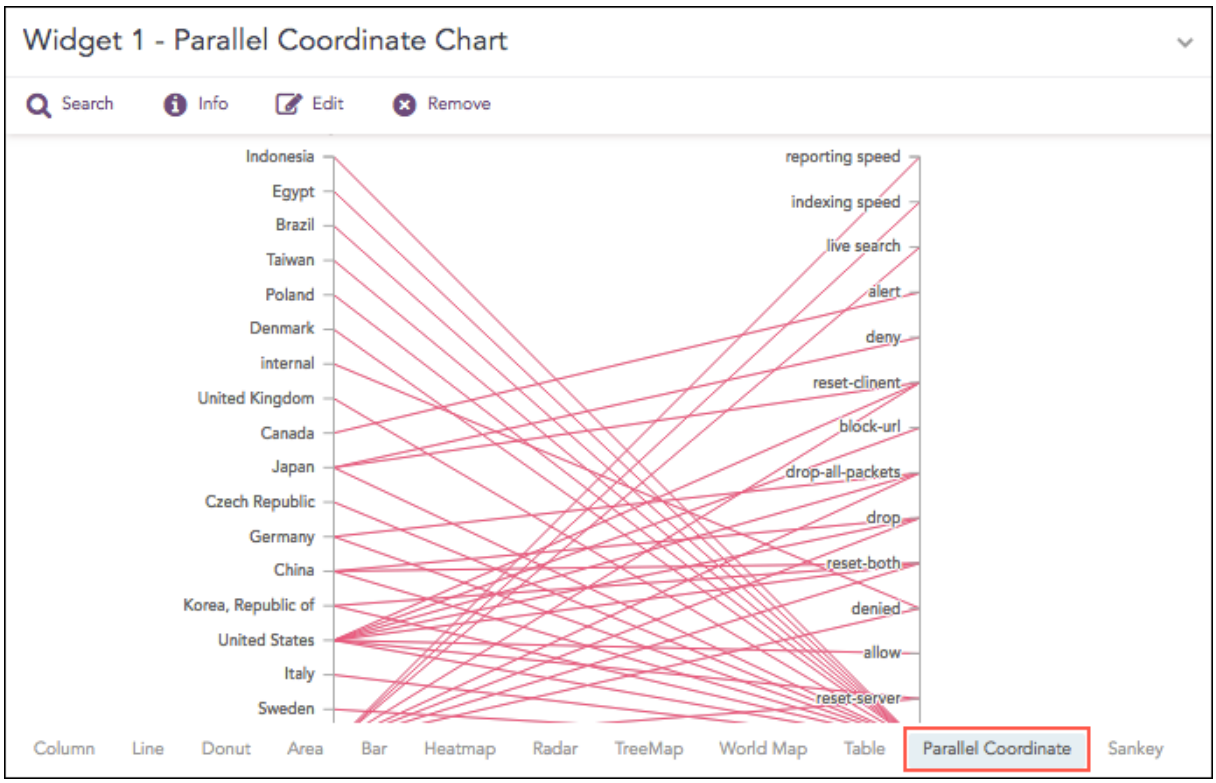
```
| timechart avg(datasize)
```

You can refer to Timechart Single Aggregation without Grouping for more details.

Parallel Coordinate Chart

The Parallel Coordinate Chart is a visualization technique used to plot individual data elements across multiple dimensions. The charts are ideal for comparing many grouping parameters and analyzing the relationships between them. Each grouping parameter has its axis, and all the axes are placed in parallel to each other. Values are plotted as a series of lines that are connected across all the axes. This means that each line is a collection of points placed on each axis, which have all been linked together.

The Parallel Coordinate chart shows both the forest and the tree. You can see the big picture in the patterns of the lines. You can highlight the individual lines to see the performance of a specific value of parameters. It is useful in the situations when the behavior of particular parameters may not be of concern, but a combination of those parameters may emphasize an abnormal pattern or relationship.



The following query gives the output shown above.

```
| process geoip(source_address) as source_country | chart count() by
source_country, sub_category, destination_ location
```

NOTE

1. Each line represents a relationship between two parameters rather than a trend or change in value.
2. As the number of values increase, the graph may be cluttered or may even overlap at times, which makes it difficult to perceive. In such a case, use the Brushing feature to highlight an individual or a group of values for better understanding.
3. You can view the value of the aggregation parameter by hovering over a relationship line.

Response Types Supported

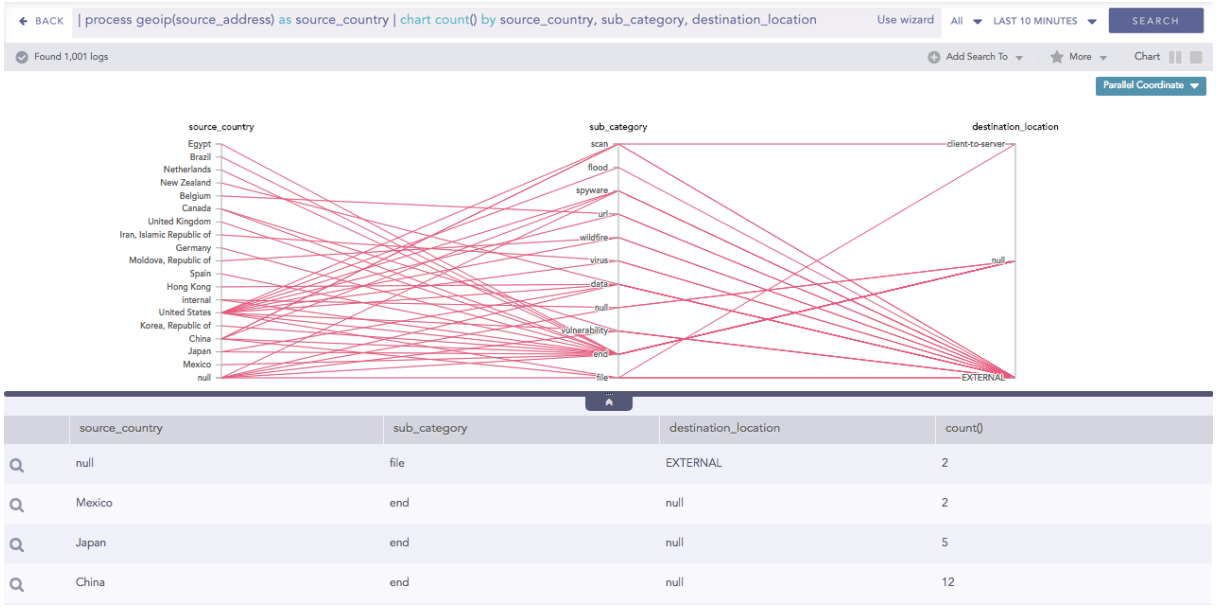
The **Parallel Coordinate** chart supports a single aggregation response types for representation of search results in the visualization. It is :

Response Type	General Syntax
Single Aggregation with Grouping	`| chart aggregation_parameter by grouping_parameter1, grouping_parameter2, ........, grouping_parametern`

Single Aggregation with Grouping

Example:

```
| process geoip(source_address) as source_country | chart count() by source_country, sub_category, destination_ location
```



	source_country	sub_category	destination_location	count()
	null	file	EXTERNAL	2
	Mexico	end	null	2
	Japan	end	null	5
	China	end	null	12

Some notable points about the Parallel Coordinate chart are as follows:

1. Each line represents a relationship between two parameters rather than a trend or a change in value.
2. As the number of values increase, the graph may be cluttered or even overlapped at times, which makes it difficult to analyze. In this case, use the Brushing feature to highlight an individual or a group of values for better understanding.
3. You can view the value of the aggregation parameter by hovering over a relationship line.

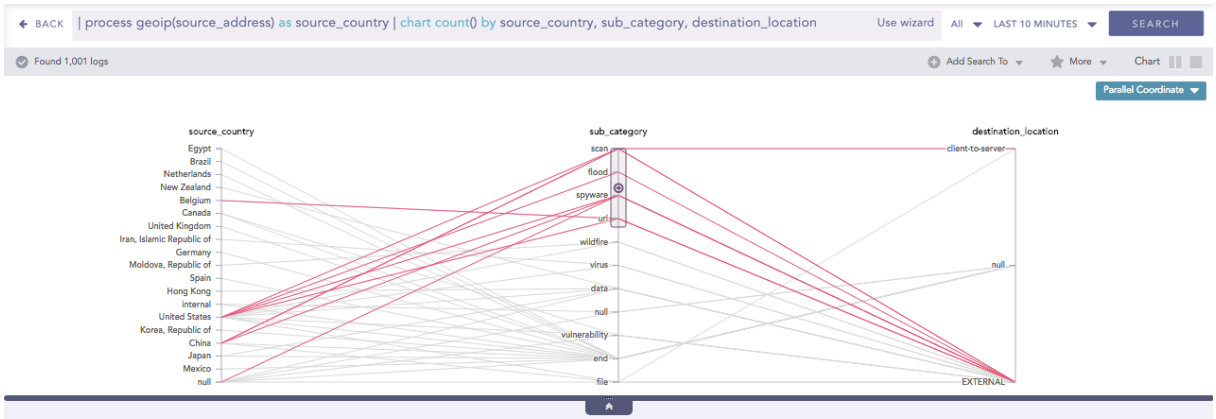
You can refer to Single Aggregation with Grouping for more details.

Operations

Brushing

The Brushing feature eliminates one of the primary drawbacks of the Parallel Coordinate chart, which is cluttering and overlapping of the graph. When the number of data items in a Parallel Coordinate chart gets very high, lines get cluttered and even overplotted which eventually becomes difficult to understand. Using the brushing feature, you can select an area containing one or many data points.

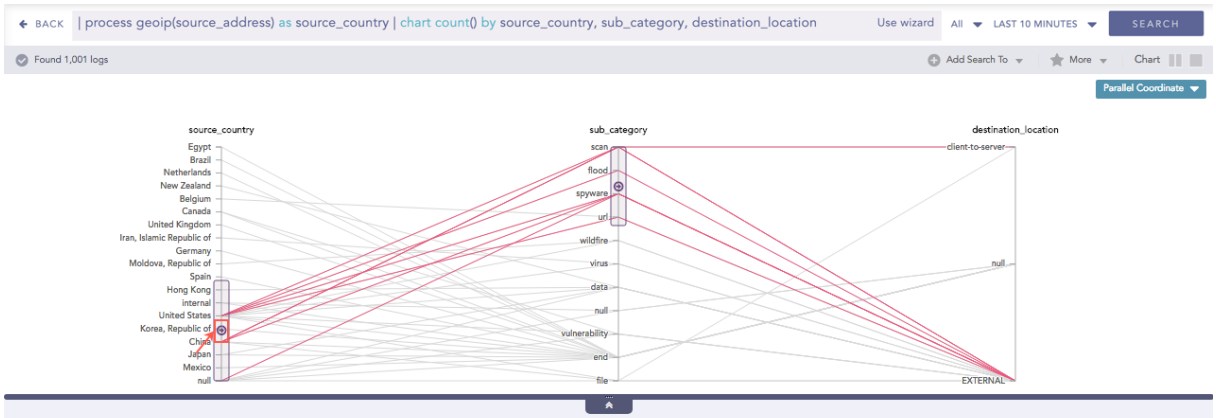
The line(s) under the brushed area is highlighted. You can view the details of the stressed relationship by hovering over the particular line. In addition to that, you can further drill-down to its details of the relationship by clicking it.



	source_country	sub_category	destination_location	count()
	null	file	EXTERNAL	2
	Mexico	end	null	2
	Japan	end	null	5
	China	end	null	12

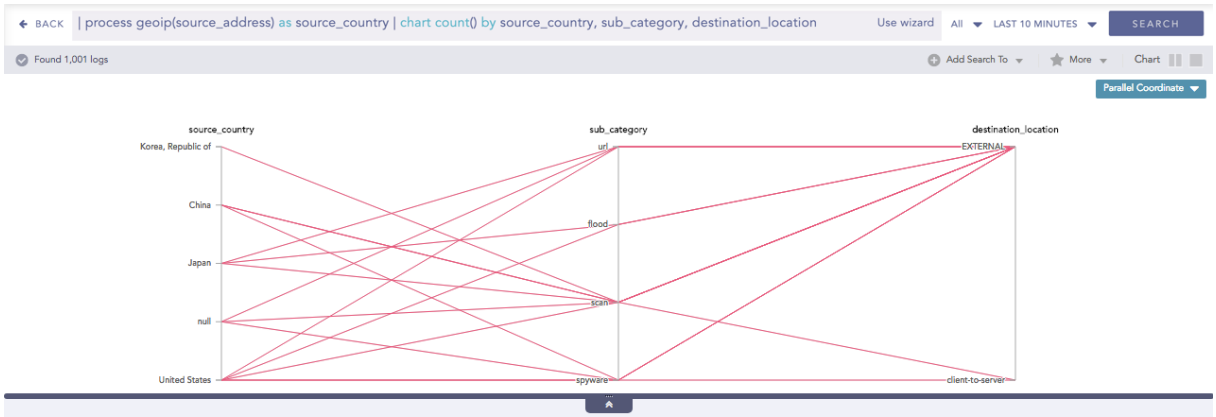
Combined Drill-down

In addition to the regular drill down operation, you can also perform a combined drill-down using the **Brush**. Select a range of values in multiple axes using a brush and click the brushed area, to drill down.



	source_country	sub_category	destination_location	count()
	null	file	EXTERNAL	2
	Mexico	end	null	2
	Japan	end	null	5
	China	end	null	12

The results of the drill-down filters down to the combination of the selected grouping parameter values.



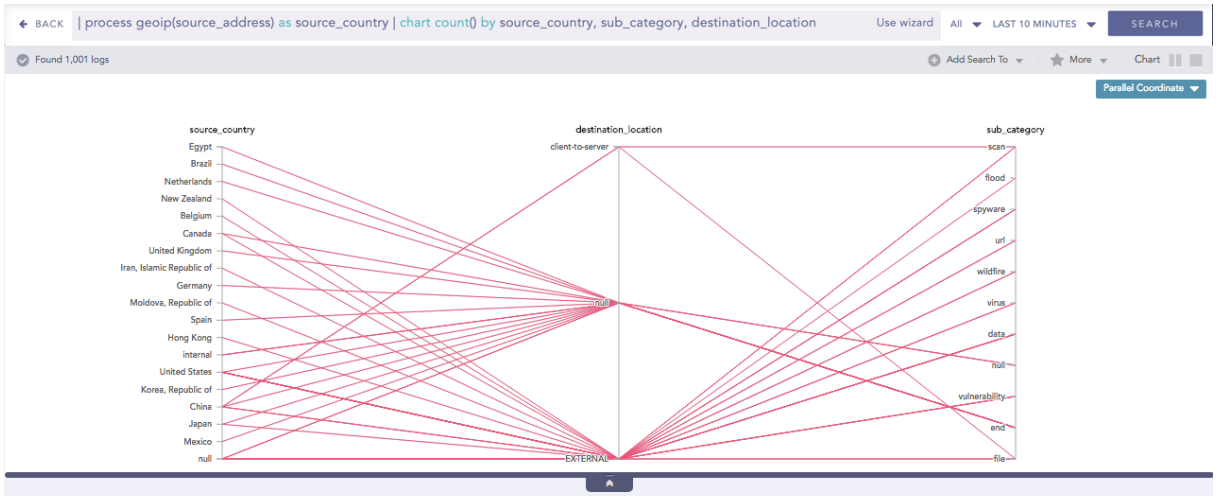
	source_country	sub_category	destination_location	count()
	United States	spyware	client-to-server	1
	null	scan	EXTERNAL	1
	Japan	flood	EXTERNAL	1
	United States	url	EXTERNAL	7

Changing order of the parameter

By default, the first grouping parameter of the query is assigned to the first axis of the Parallel Coordinate chart, followed by the other grouping parameters.

For example, in the above query, the grouping parameter **source_address**, **sub_category** and **destination_location** are placed in first, second and third axes respectively.

You can also change the order of parameters by dragging them across the parameters with which the value is to be exchanged.



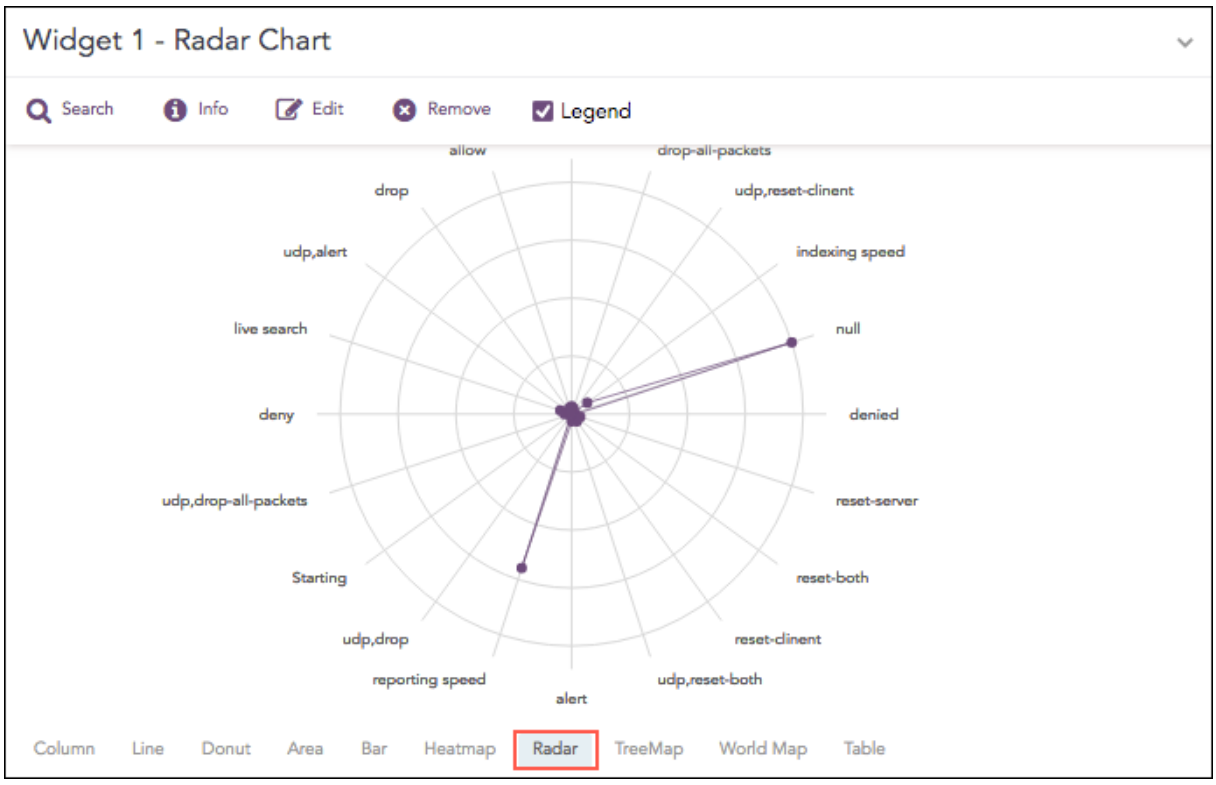
	source_country	sub_category	destination_location	count()
	null	file	EXTERNAL	2
	Mexico	end	null	2
	Japan	end	null	5
	China	end	null	12
	Korea, Republic of	end	null	2

Radar chart

The Radar chart is a graphical representation of multivariate data in the form of a two-dimensional graph, in which one or more quantitative variables are represented on axes starting from the same point.

Each value of grouping parameter(s) forms an individual axis which is arranged radially around a point. These axes are equiangular to each other and known as spoke or radii. Each node depicts the value of a spoke, and the lines are drawn to connect the nodes to each other.

The Radar chart is best for visualizing outliers in a dataset, especially in cases of operation related analysis such as performance metrics and quality improvement. The line between the origin points and radii can be used as the scale for data points.



The following query gives the output shown above.

```
| chart count by action()
```

Response Types Supported

The **Radar** chart supports four aggregation response types for representation of search results in the visualization. They are:

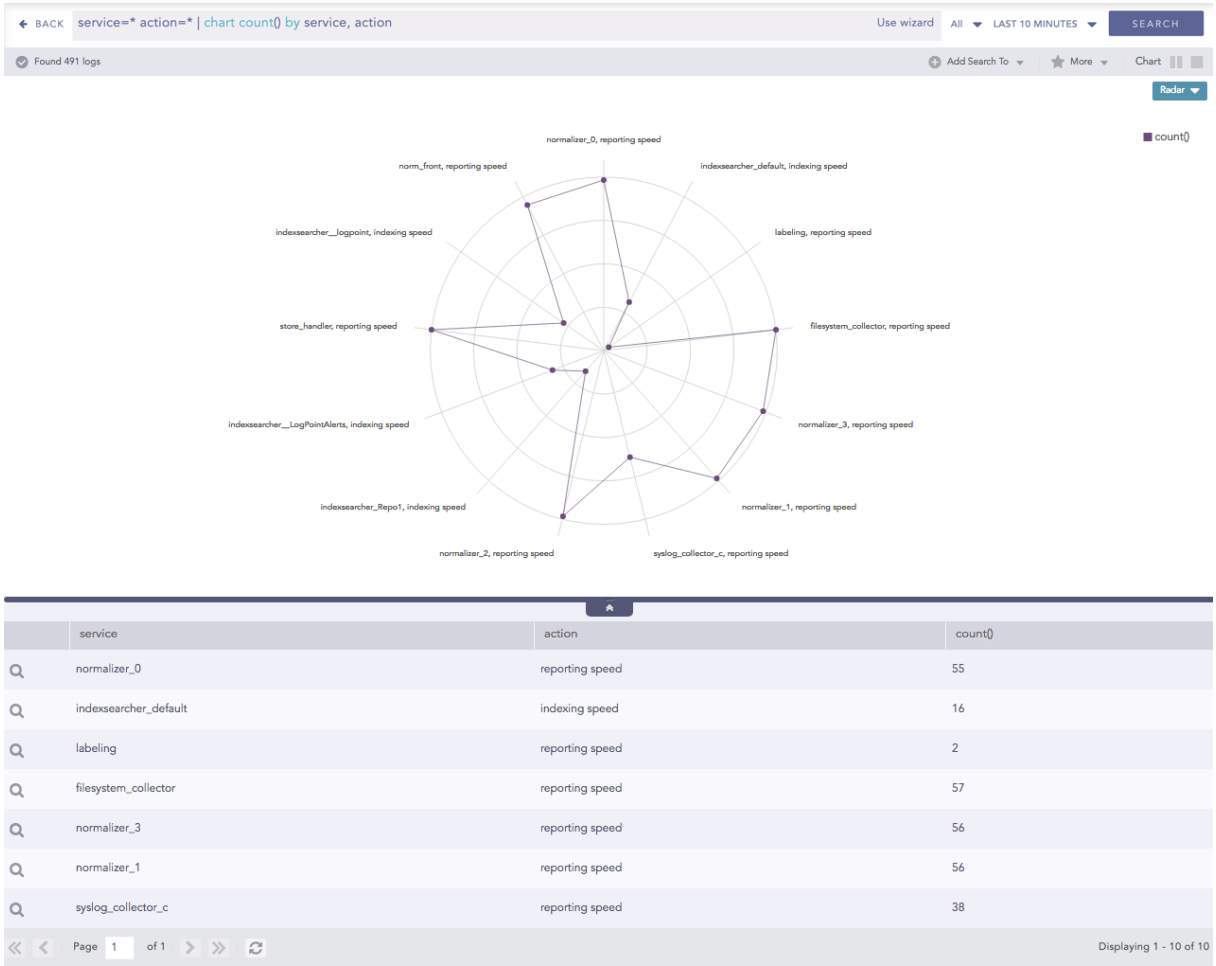
Response Type	General Syntax
Single Aggregation with Grouping	`| chart aggregation_parameter by grouping_parameter1, grouping_parameter2, ........, grouping_parametern`
Multiple Aggregation with Grouping	`| chart aggregation_parameter1, aggregation_parameter2 by grouping_parameter1, grouping_parameter2, ..., grouping_parametern`

Response Type	General Syntax
Timechart Single Aggregation without Grouping	`| timechart aggregation_parameter`
Timechart Multiple Aggregation without Grouping	`| timechart aggregation_parameter1, aggregation_parameter2, .... aggregation_parametern`

Single Aggregation with Grouping

Example:

```
service=* action=* | chart count() by action, service
```



service	action	count()
normalizer_0	reporting speed	55
indexsearcher_default	indexing speed	16
labeling	reporting speed	2
filesystem_collector	reporting speed	57
normalizer_3	reporting speed	56
normalizer_1	reporting speed	56
syslog_collector_c	reporting speed	38

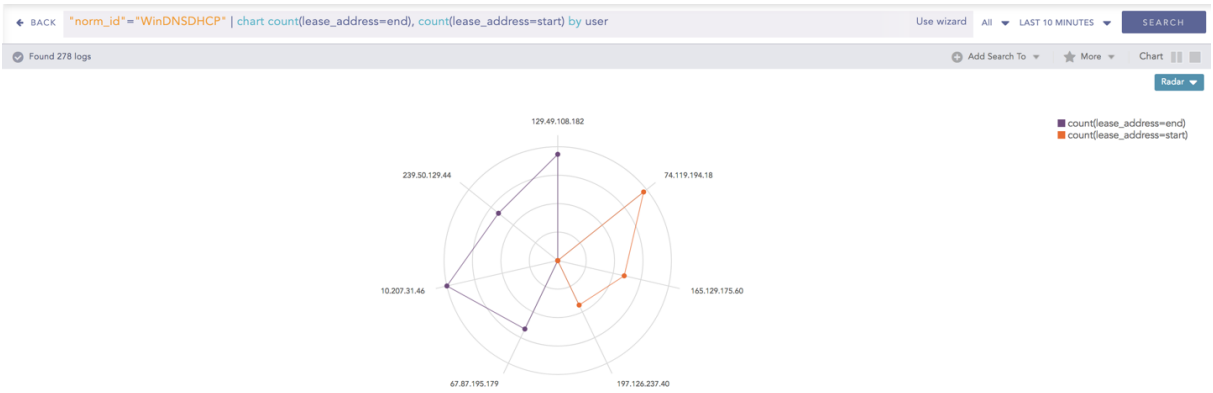
Page 1 of 1 — Displaying 1 - 10 of 10

You can refer to Single Aggregation with Grouping for more details.

Multiple Aggregation with Grouping

Example:

```
"norm_id"="WinDNSDHCP"| chart count(lease_address=end), count(lease_address=start) by user
```



	user	count(lease_address=end)	count(lease_address=start)
	129.49.108.182	28	0
	74.119.194.18	0	29
	165.129.175.60	0	18
	197.126.237.40	0	13
	67.87.195.179	20	0

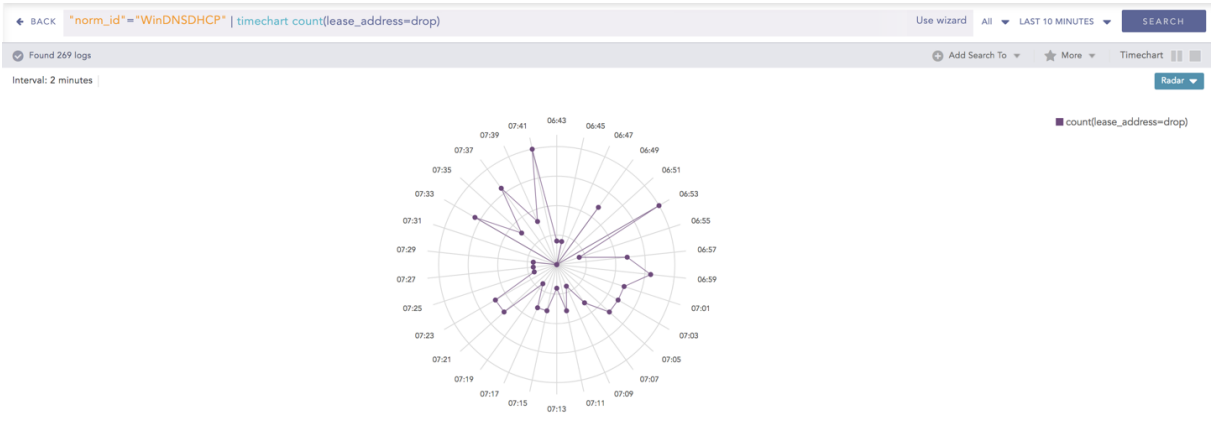
You can refer to Multiple Aggregation with Grouping for more details.

Timechart Single Aggregation without Grouping

The Radar chart can be used in time queries to graphically represent the change in values of the aggregation parameter over a period.

Example:

```
"norm_id"="WinDNSDHCP"| timechart count(lease_address=drop)
```



	Timestamp	count(lease_address=drop)
	2018/07/12 06:43:14	1
	2018/07/12 06:45:14	1
	2018/07/12 06:47:14	0
	2018/07/12 06:49:14	3
	2018/07/12 06:51:14	0
	2018/07/12 06:53:14	5
	2018/07/12 06:55:14	1

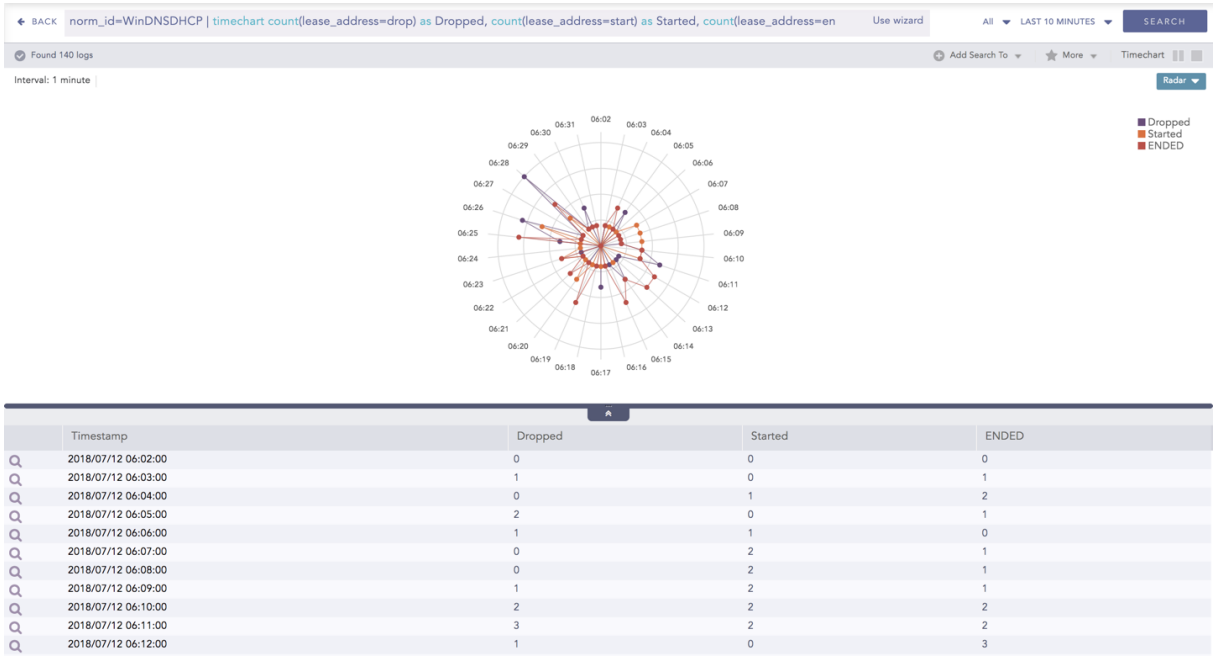
You can refer to Timechart Single Aggregation without Grouping for more details.

Timechart Multiple Aggregation without Grouping

The Radar chart can be used in time queries to graphically represent the change in values of aggregation over a period. For Timechart Multiple Aggregation without grouping type, each aggregation parameter is represented by a unique color.

Example:

```
norm_id=WinDNSDHCP | timechart count(lease_address=drop) as Dropped, count
(lease_address=start) as Started, count(lease_address=end) as ENDED
```



Timestamp	Dropped	Started	ENDED
2018/07/12 06:02:00	0	0	0
2018/07/12 06:03:00	1	0	1
2018/07/12 06:04:00	0	1	2
2018/07/12 06:05:00	2	0	1
2018/07/12 06:06:00	1	1	0
2018/07/12 06:07:00	0	2	1
2018/07/12 06:08:00	0	2	1
2018/07/12 06:09:00	1	2	1
2018/07/12 06:10:00	2	2	2
2018/07/12 06:11:00	3	2	2
2018/07/12 06:12:00	1	0	3

You can refer to Timechart Multiple Aggregation without Grouping for more details.

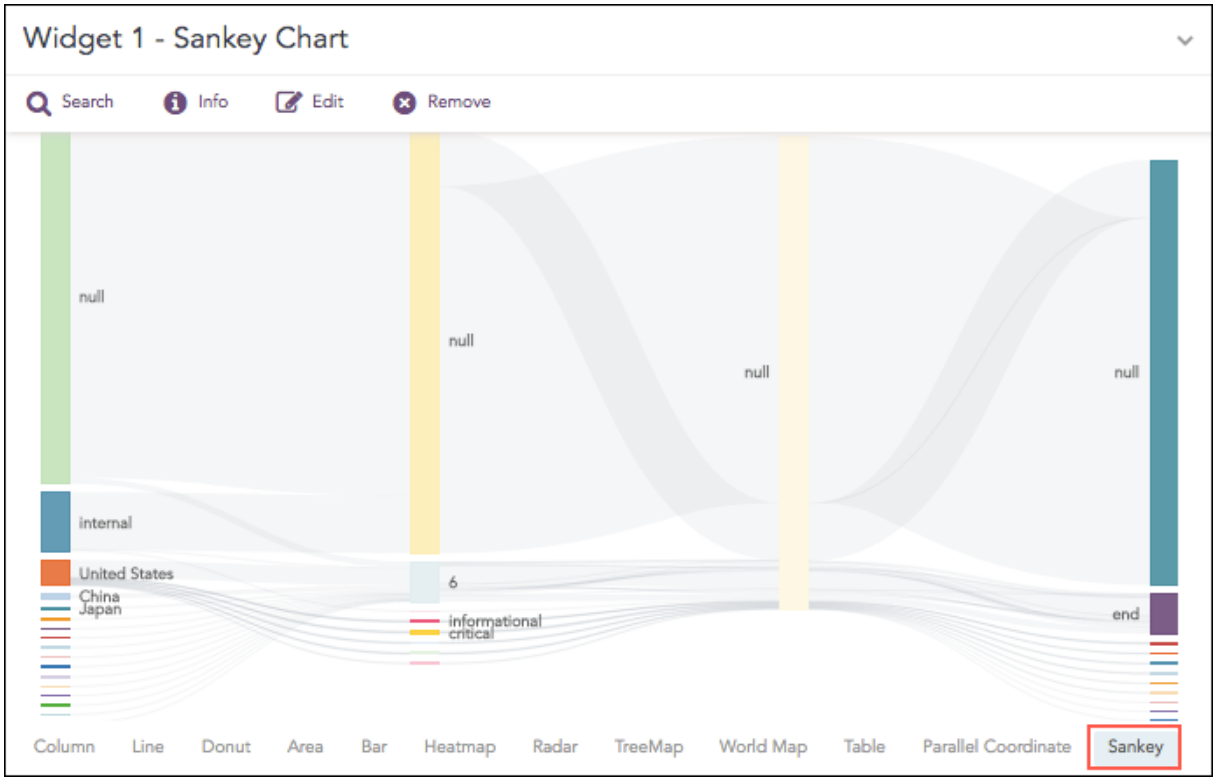
Sankey chart

Sankey chart is a flow diagram used to depict a flow from one set of values to another. The connected values are called **nodes** and the connections are called **links**. It displays the corresponding grouping parameters on top of each node of the chart. The width of the link shows the magnitude of the flow. Colors are used to divide the diagram into different nodes or to show the transition from one state of the process to another.

Use the Sankey chart to show a **many to many** mapping between two or more nodes. The **aggregation parameter** is used to define the width of the flow between a source node and the destination node.

Example:

```
| process geoip(source_address) as country | chart count() by country,
severity, category, sub_category
```



Response Types Supported

The **Sankey** chart supports a single aggregation response types for representation of search results in the visualization. It is :

Response Type	General Syntax
Single Aggregation with Grouping	`| chart aggregation_parameter by grouping_parameter1, grouping_parameter2, ........, grouping_parametern`

Single Aggregation with Grouping

Example:

```
| process geoip(source_address) as country | chart count() by country,
severity, category, sub_category
```

You can refer to Single Aggregation with Grouping for more details.

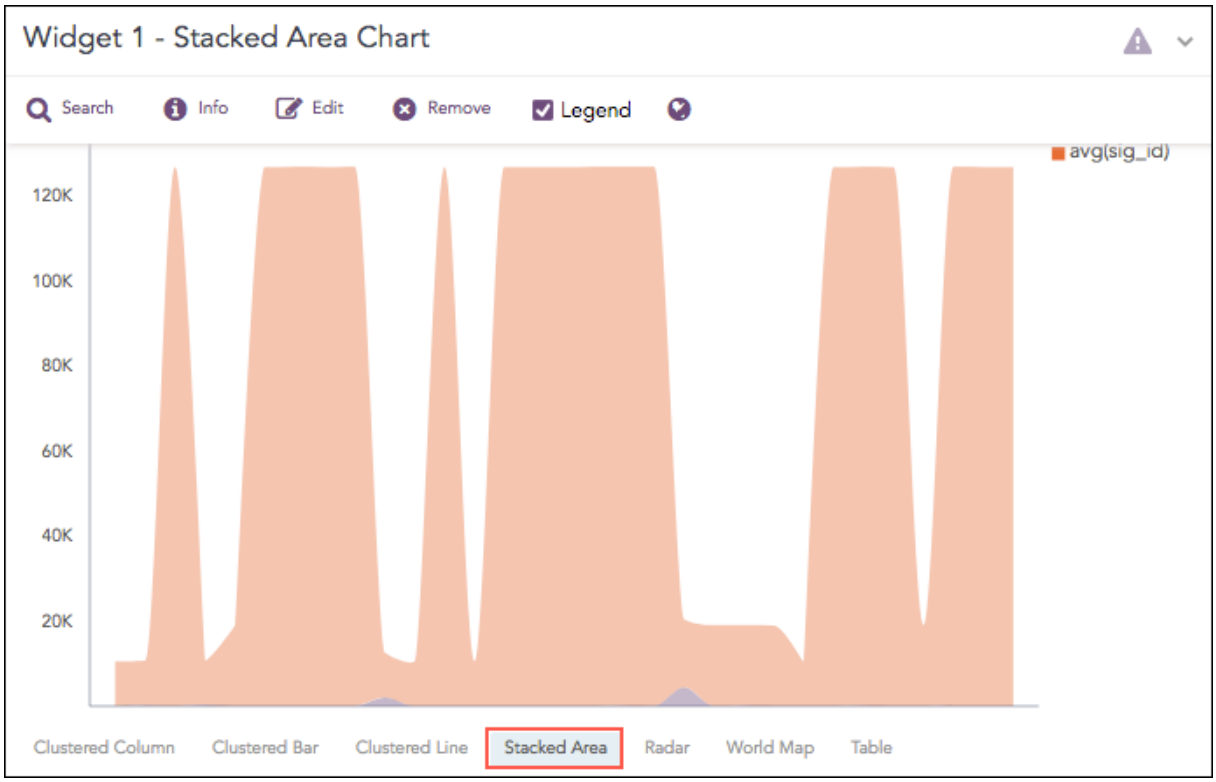
Operations

Vertical Reposition

You can change the vertical position of the nodes by dragging them in the upward or the downward direction. You can either overlap the nodes or place them distinctly.

Stacked Area Chart

Stacked Area charts are fundamentally similar to a standard Area Chart, except for the use of multiple variables in the x-axis instead of a single variable.



The following query gives the output shown above.

```
| chart count(), avg(sig_id) by action
```

Response Types Supported

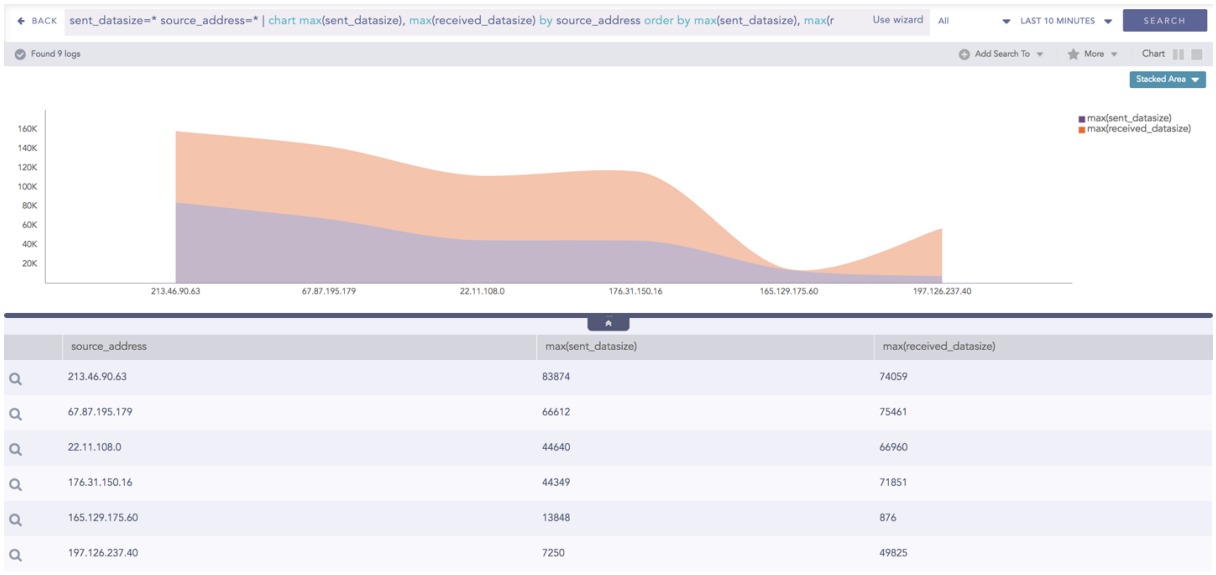
The **Stacked Area** chart supports two aggregation response types for representation of search results in the visualization. They are:

Response Type	General Syntax
Multiple Aggregation with Grouping	`| chart aggregation_parameter1, aggregation_parameter2 by grouping_parameter1, grouping_parameter2, ...,grouping_parametern`
Timechart Multiple Aggregation without Grouping	`| timechart aggregation_parameter1, aggregation_parameter2, .... aggregation_parametern`

Multiple Aggregation with Grouping

In the Multiple Aggregation with Grouping response type, the x-axis contains the values of the grouping parameter(s), whereas the y-axis consists the scale to measure the value of the aggregation parameters. A unique color represents each aggregation parameter.

```
sent_datasize=* source_address=* | chart max(sent_datasize), max(received_datasize) by source_ address order by max(sent_datasize), max(received_datasize) desc limit 10`
```



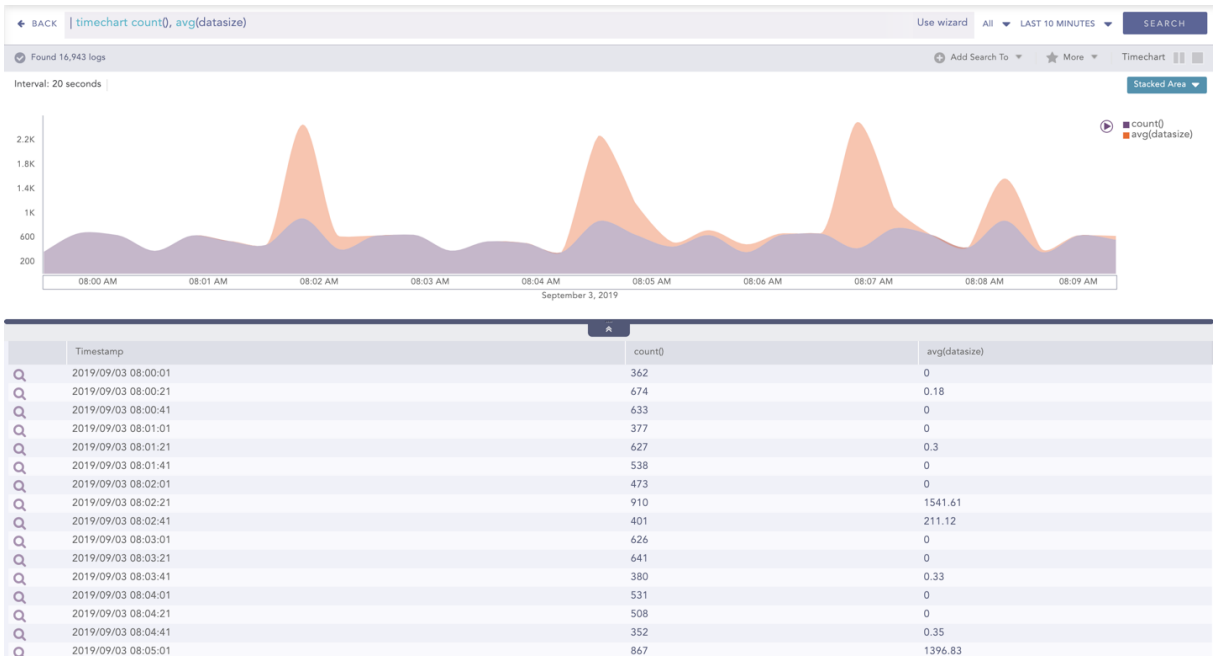
source_address	max(sent_datasize)	max(received_datasize)
213.46.90.63	83874	74059
67.87.195.179	66612	75461
22.11.108.0	44640	66960
176.31.150.16	44349	71851
165.129.175.60	13848	876
197.126.237.40	7250	49825

You can refer to Multiple Aggregation with Grouping for more details.

Timechart Multiple Aggregation without Grouping

Example:

```
| timechart count(), avg(datasize)
```



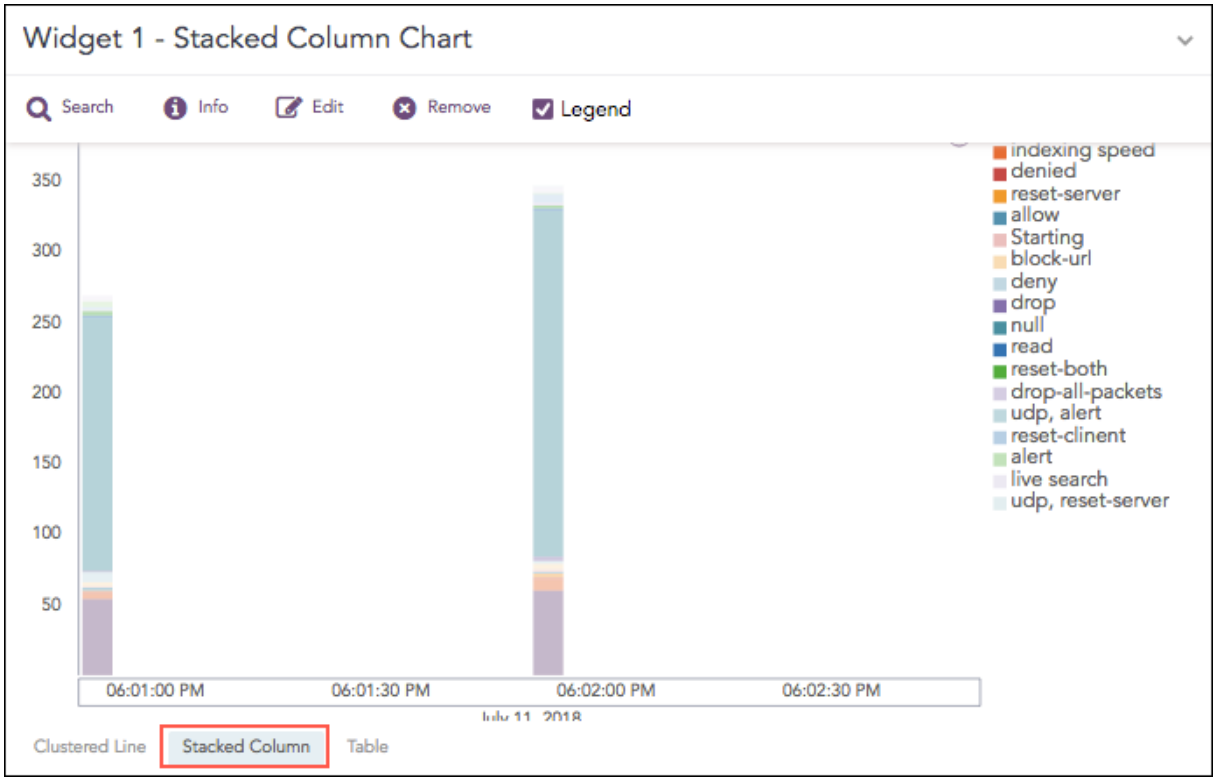
Timestamp	count()	avg(datasize)
2019/09/03 08:00:01	362	0
2019/09/03 08:00:21	674	0.18
2019/09/03 08:00:41	633	0
2019/09/03 08:01:01	377	0
2019/09/03 08:01:21	627	0.3
2019/09/03 08:01:41	538	0
2019/09/03 08:02:01	473	0
2019/09/03 08:02:21	910	1541.61
2019/09/03 08:02:41	401	211.12
2019/09/03 08:03:01	626	0
2019/09/03 08:03:21	641	0
2019/09/03 08:03:41	380	0.33
2019/09/03 08:04:01	531	0
2019/09/03 08:04:21	508	0
2019/09/03 08:04:41	352	0.35
2019/09/03 08:05:01	867	1396.83

You can refer to Timechart Multiple Aggregation without Grouping for more details.

Stacked Column Chart

A Stacked Column chart uses bars to show the comparisons between categories of data but with an ability to break down and compare parts of a whole. Each bar in the chart represents a whole, and segments in the bar represent different parts or categories of that whole.

Similar to the Clustered Line Chart, the y-axis represents value of the aggregation parameter, and the x-axis displays value of the timestamps.



The following query gives the output shown above.

```
| timechart count() by action
```

You can use this chart to display the following response type:

Response Types Supported

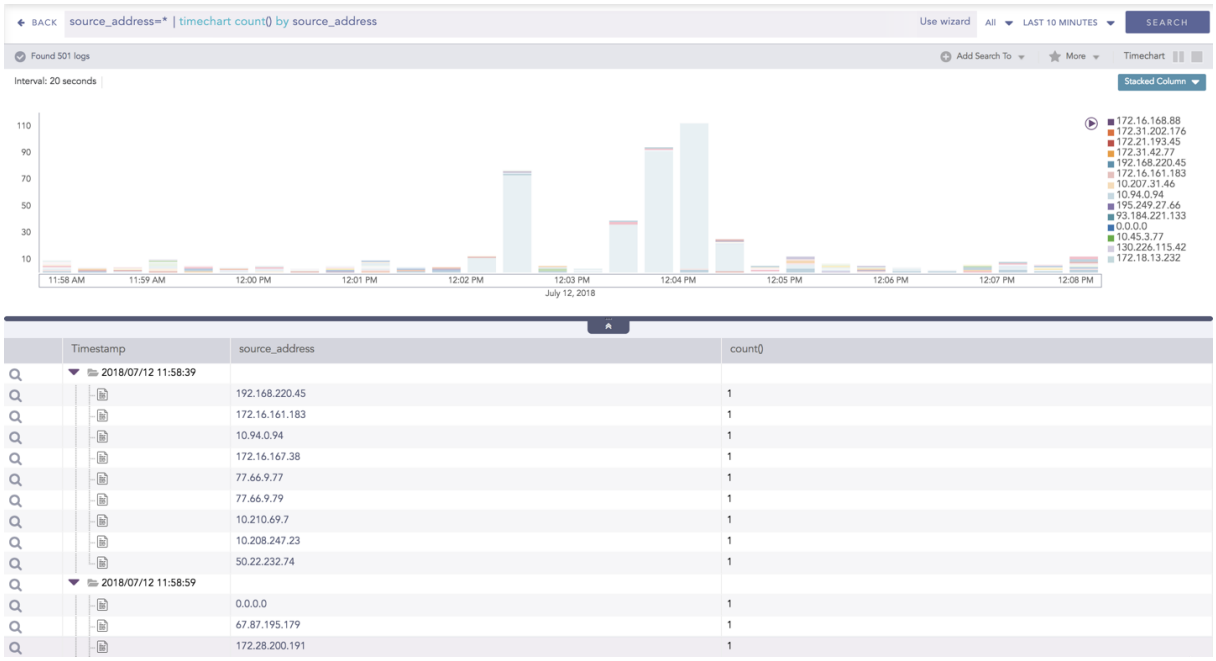
The **Stacked Column** chart supports a single aggregation response type for representation of search results in the visualization. They are:

Response Type	General Syntax
Timechart Single Aggregation with Grouping	`| timechart aggregation_parameter by grouping_parameter1, grouping_parameter2, ....., grouping_parametern`

Timechart Single Aggregation with Grouping

Example:

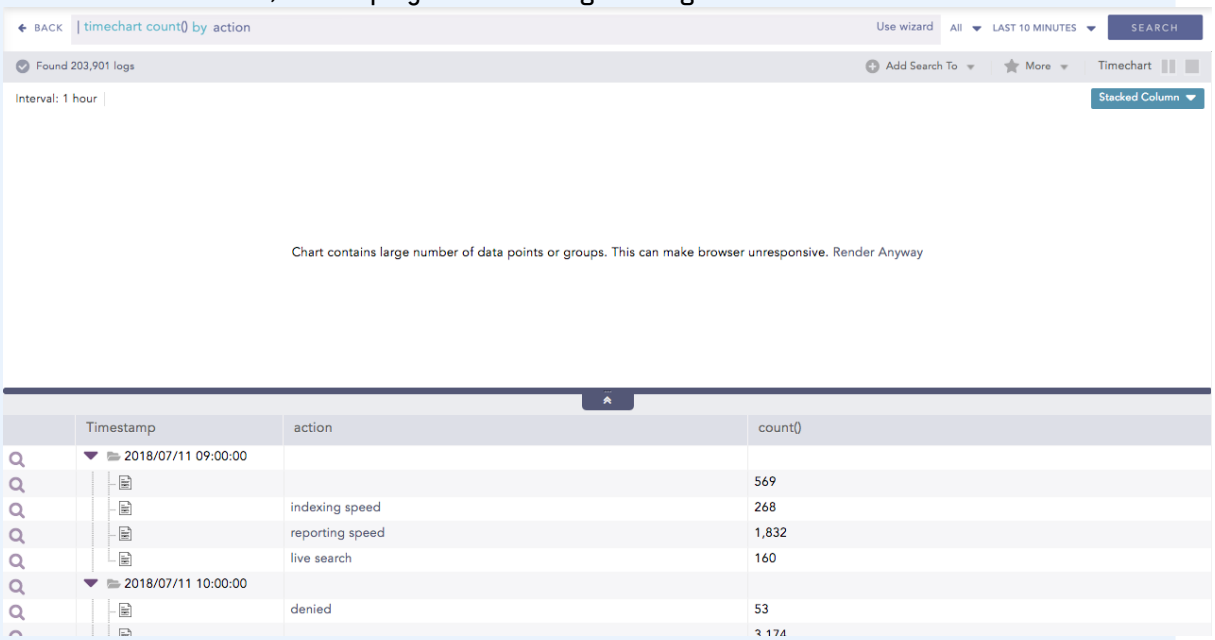
```
source_address=* | timechart count() by source_address
```



Timestamp	source_address	count()
2018/07/12 11:58:39		
	192.168.220.45	1
	172.16.161.183	1
	10.94.0.94	1
	172.16.167.38	1
	77.66.9.77	1
	77.66.9.79	1
	10.210.69.7	1
	10.208.247.23	1
	50.22.232.74	1
2018/07/12 11:58:59		
	0.0.0.0	1
	67.87.195.179	1
	172.28.200.191	1

NOTE

If the search result contains a large number of data points (more than 50) or groups (more than 20), switching from the Clustered Line to Stacked Column consumes a large amount of CPU resources. In this case, SLS displays the following message.

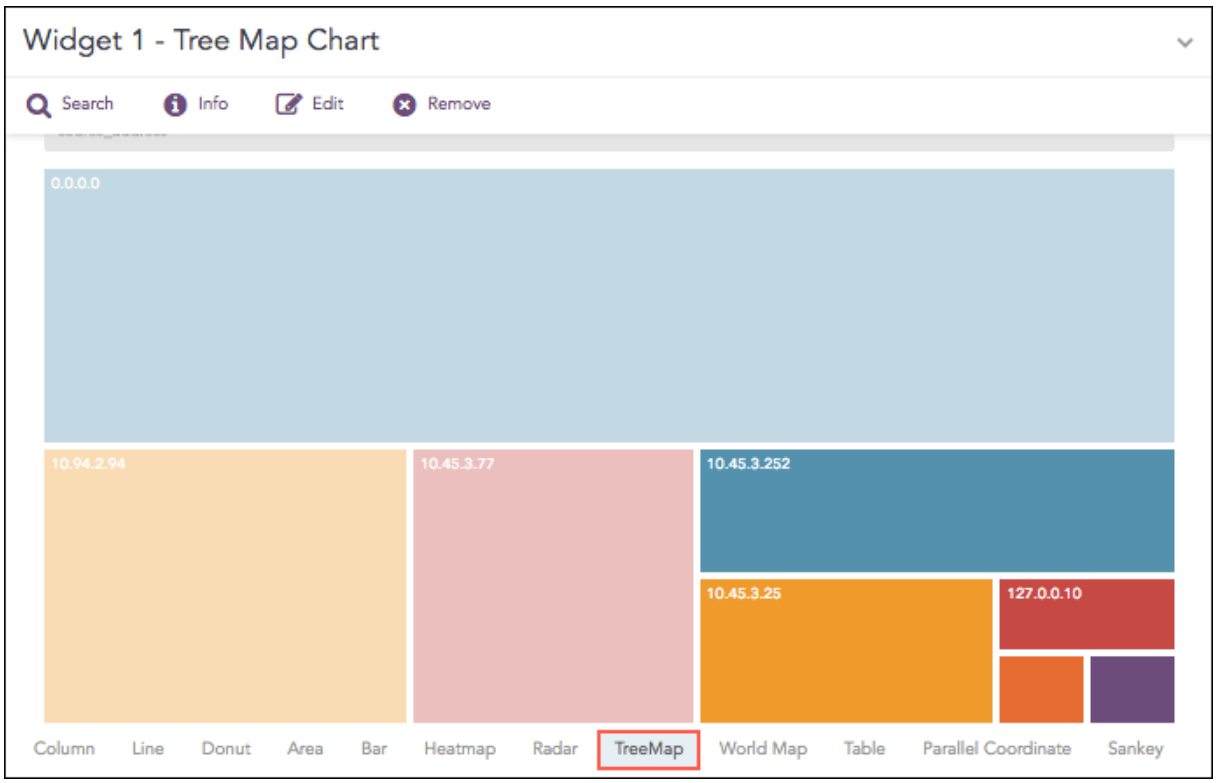


Timestamp	action	count()
2018/07/11 09:00:00		
		569
	indexing speed	268
	reporting speed	1,832
	live search	160
2018/07/11 10:00:00		
	denied	53
		3,174

You can refer to Timechart Single Aggregation with Grouping for more details.

TreeMap Chart

The **TreeMap** chart visualizes the hierarchical structure of a tree diagram. It displays the weight of each node in the form of the area size. Each node is assigned a rectangular area with their child nodes nested inside. The space of each node inside a parent node is displayed with proportion to all other nodes within the same parent node. If the weight of a child node is zero, the node is not included in the diagram.



The following query gives the output shown above.

```
source_address=* action=* | chart count() by source_address, action order
by count() desc limit 10
```

The first grouping parameter is the parent node of a TreeMap diagram, and all its successive parameters are the child nodes.

The name of the first grouping parameter is displayed in the breadcrumb, while all its fields are displayed in the containers as individual nodes.

Response Types Supported

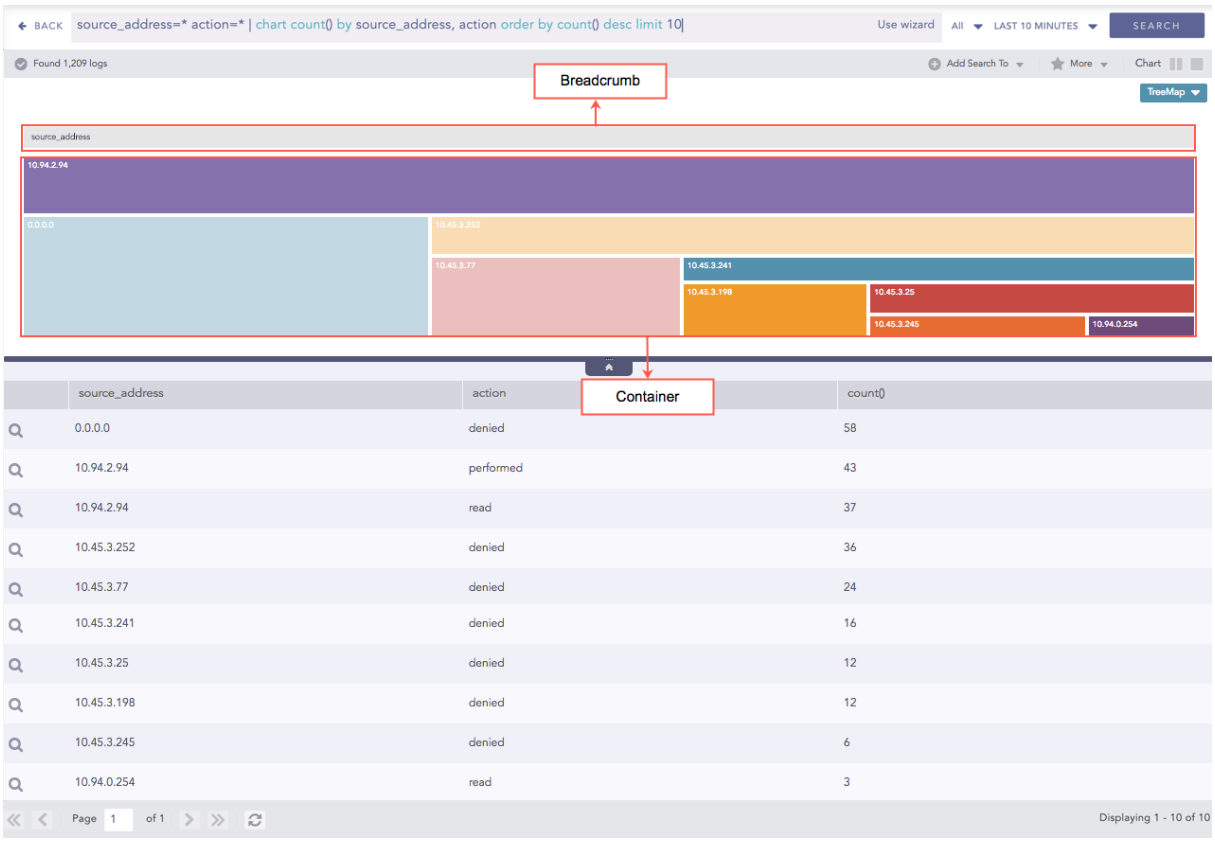
The **Treemap** chart supports a single aggregation response types for representation of search results in the visualization. It is :

Response Type	General Syntax
Single Aggregation with Grouping	`| chart aggregation_parameter by grouping_parameter1, grouping_parameter2, ........, grouping_parametern`

Single Aggregation with Grouping

Example:

```
source_address=* action=* | chart count() by source_address, action order
by count() desc limit 10
```

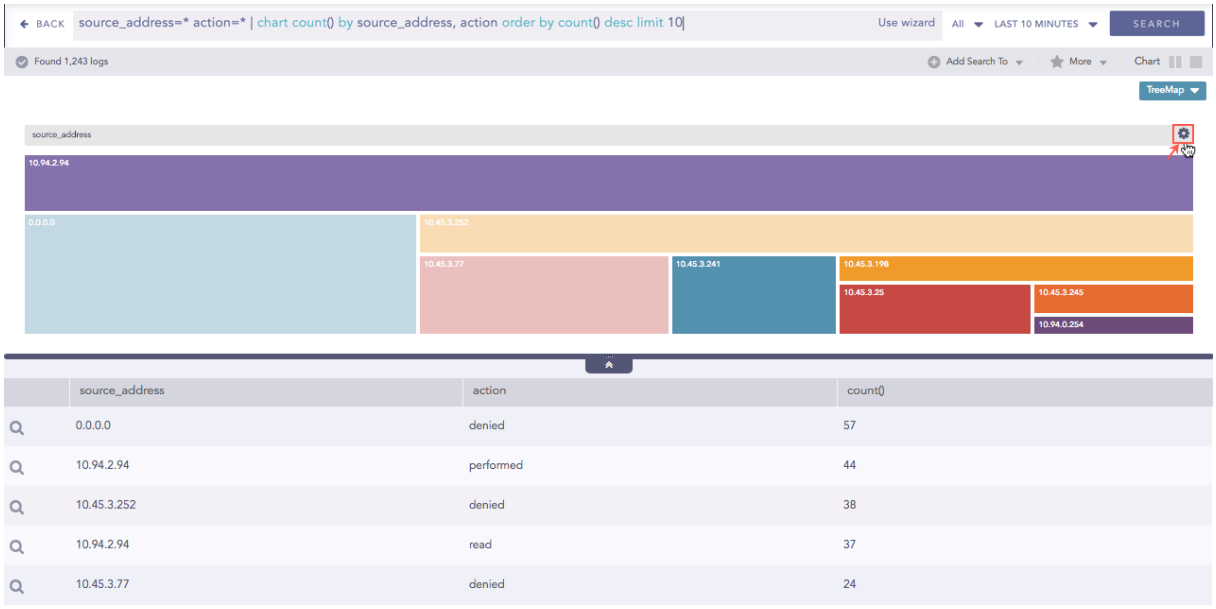


NOTE
The aggregation parameter determines the area size of each node in the container.

You can refer to Single Aggregation with Grouping for more details.

Rendering Parameters

Click the **gear** icon on the right side of the breadcrumb to select the rendering parameters for the nodes of the treemap chart.

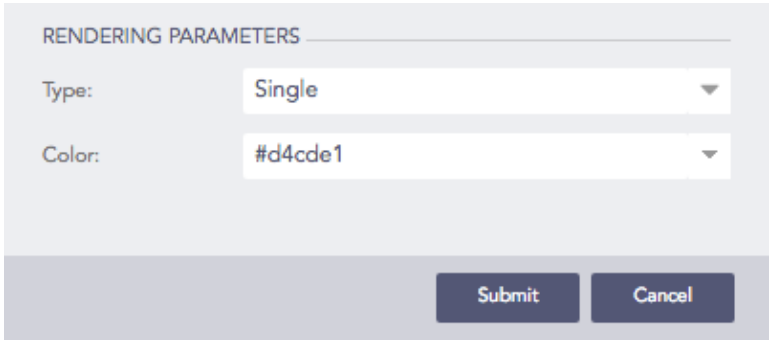


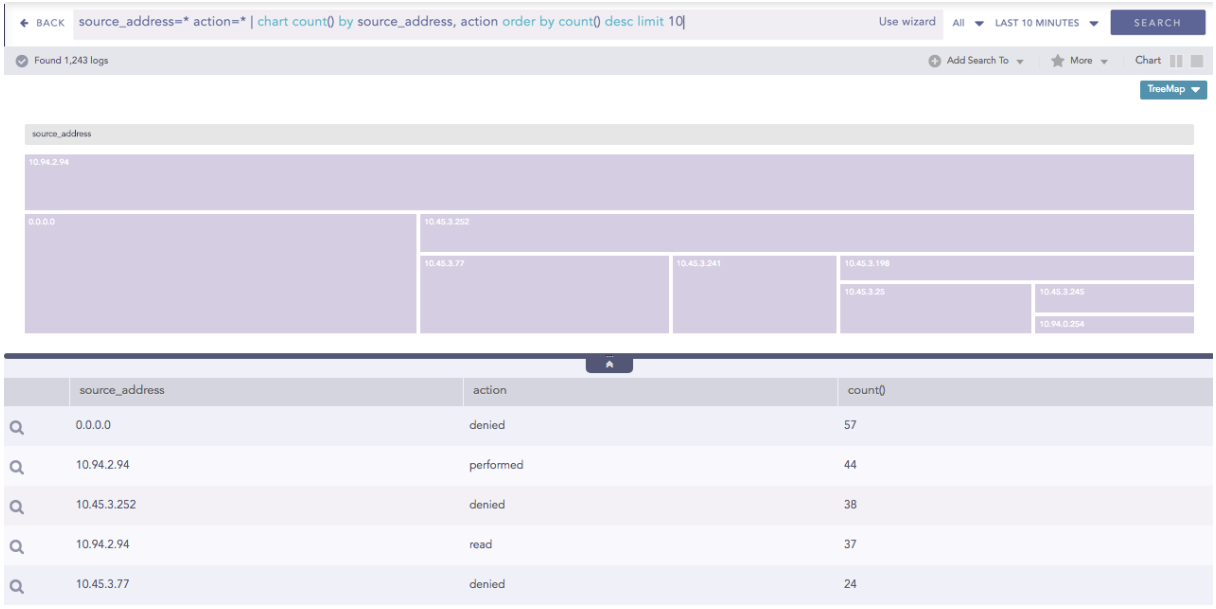
	source_address	action	count()
	0.0.0.0	denied	57
	10.94.2.94	performed	44
	10.45.3.252	denied	38
	10.94.2.94	read	37
	10.45.3.77	denied	24

You can choose one of the following type in the rendering parameters

1. Single
2. Unique
3. Gradient

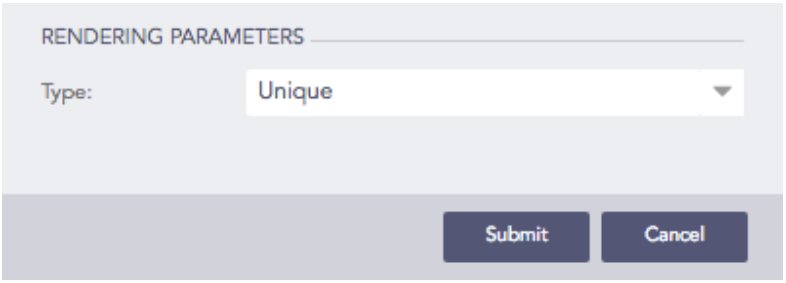
If the **Single** type is selected, all the nodes in the container are represented by a single color.
You can also select the color to represent the nodes from the **Color** picker tool.



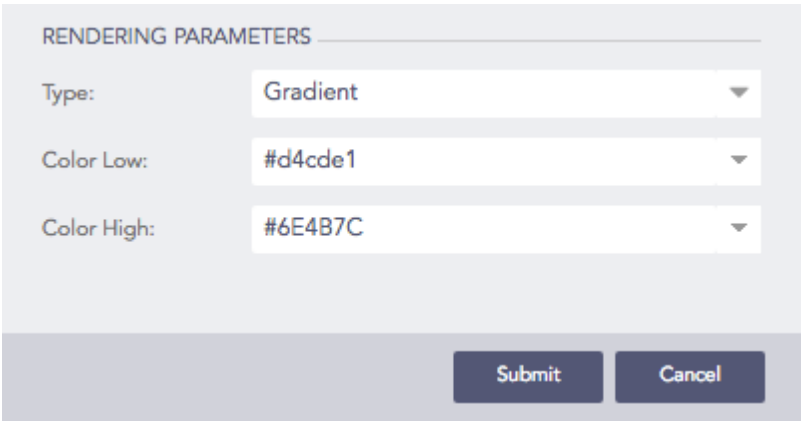


	source_address	action	count()
	0.0.0.0	denied	57
	10.94.2.94	performed	44
	10.45.3.252	denied	38
	10.94.2.94	read	37
	10.45.3.77	denied	24

If the **Unique** type rendering parameter is selected, all the nodes in the container are represented by a unique color. The colors are chosen randomly by the SLS itself.



If you select the **Gradient** type rendering parameter, the **Color High** represents the node with the most significant area size and **Color Low** represents the node with the least area size.



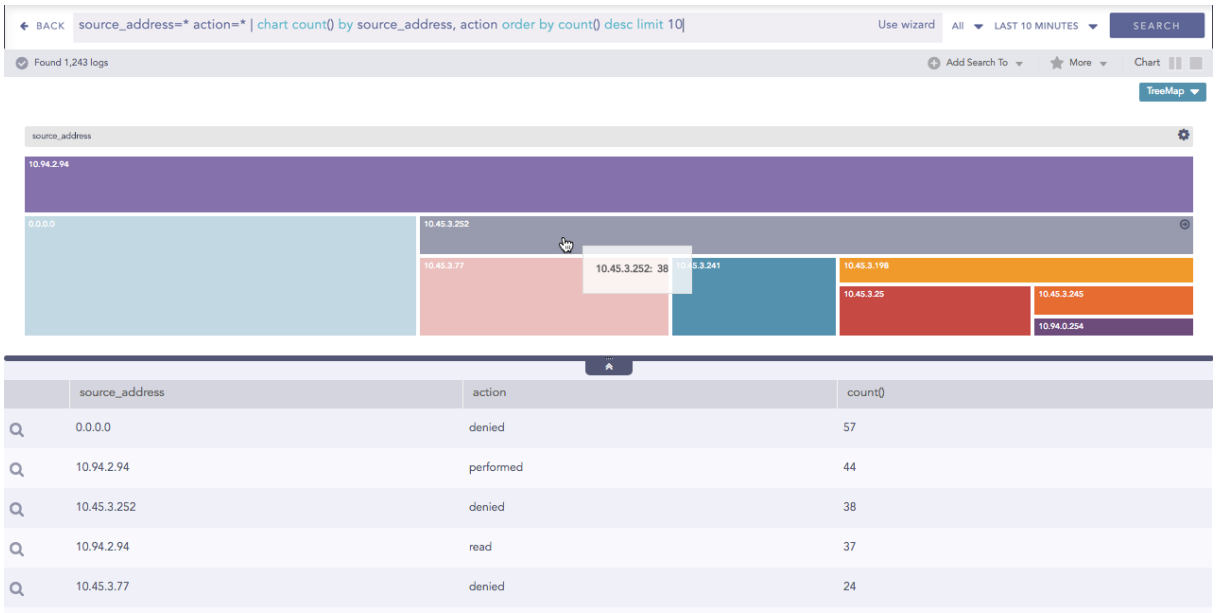
Each section has a defined color, and different shades of color represent all the nodes of the division. The darkest shade represents the node with the most significant area size, and the shade of the color fades as the area size of the nodes decrease.

You can select the color for the nodes of high area value and low area value from **Color High** and **Color Low** drop-down menu respectively.

Operations

Zoom In and Zoom Out

The Zoom In feature allows you to click any node of the container and expand the chart further.



source_address	action	count()
0.0.0.0	denied	57
10.94.2.94	performed	44
10.45.3.252	denied	38
10.94.2.94	read	37
10.45.3.77	denied	24

The expanded diagram displays the nodes of the successive grouping parameter associated with the selected parent node. The new node is shifted to the breadcrumb, and the container is updated with the fields of the node in the breadcrumb.

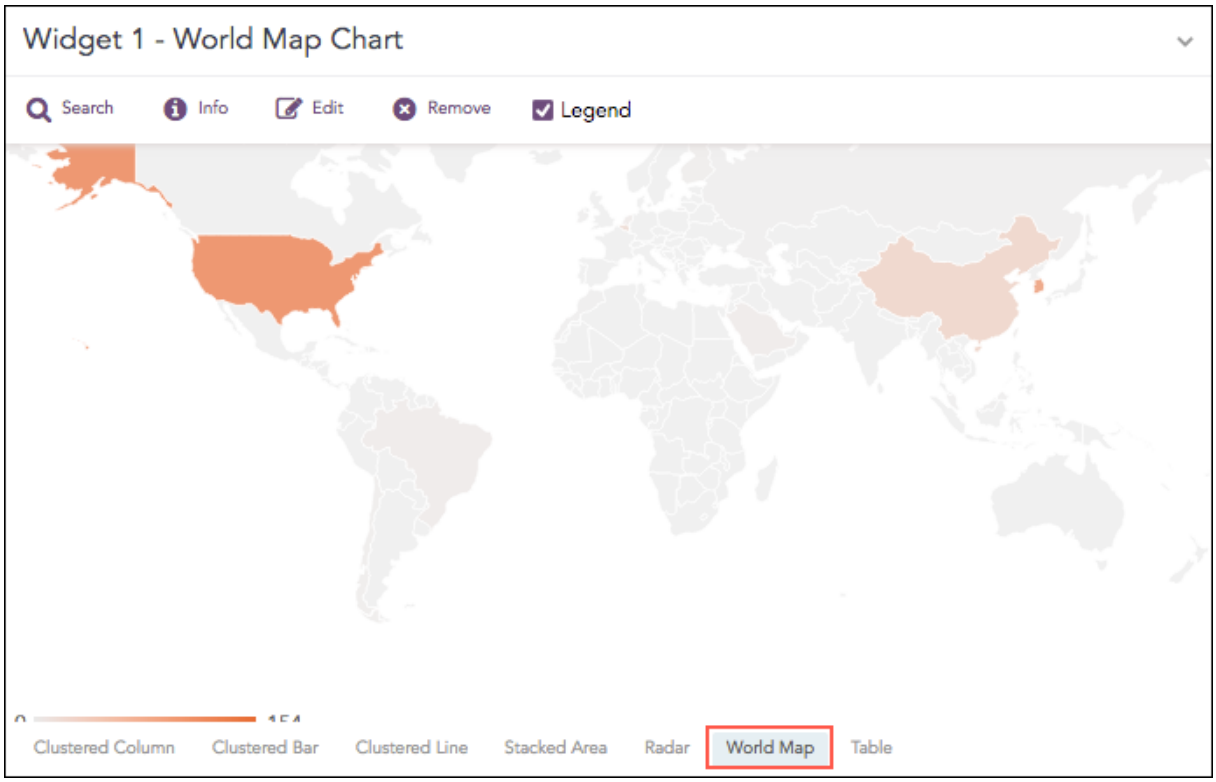
For example: In the diagram above, when the user clicks **source_address**`10.45.3.252`, it is shifted to the breadcrumb and all its related fields are displayed in the container.

source_address	action	count()
0.0.0.0	denied	57
10.94.2.94	performed	44
10.45.3.252	denied	38
10.94.2.94	read	37
10.45.3.77	denied	24

With the Zoom Out feature, you can go back to the previous state of the diagram by clicking the breadcrumb.

World Map Chart

A World Map is a map of a country, a continent, or a region, with colors and values assigned to specific regions. Values are displayed as a color scale, and you can see the name of the country by hovering over a particular part.



The following query gives the output shown above.

```
| process geoip(destination_address) as country_name | chart count(), avg
(datasize) by country_name, action
```

Response Types Supported

The **World Map** chart supports two aggregation response types for representation of search results in the visualization. They are:

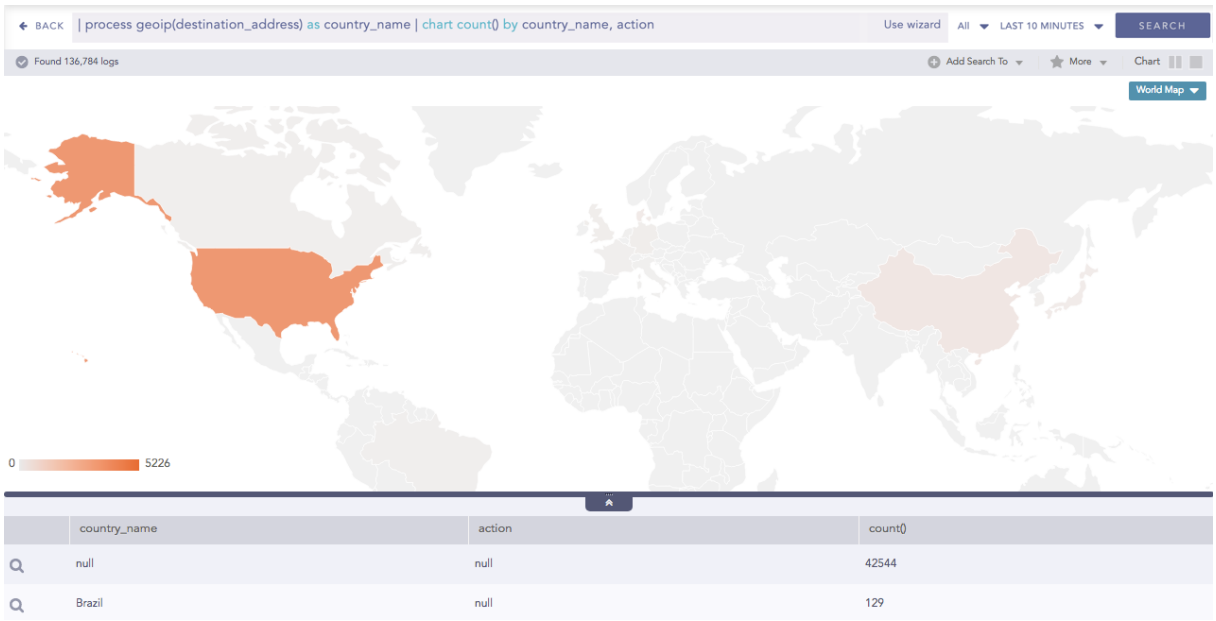
Response Type	General Syntax
Single Aggregation with Grouping	`| chart aggregation_parameter by grouping_parameter1, grouping_parameter2, ........, grouping_parametern`
Multiple Aggregation with Grouping	`| chart aggregation_parameter1, aggregation_parameter2 by grouping_parameter1, grouping_parameter2, ..., grouping_parametern`

Single Aggregation with Grouping

In Single Aggregation with Grouping, the color shade on each region of a World Map displays the value of the aggregation parameter, i.e., higher the value of the aggregation parameter, darker the color.

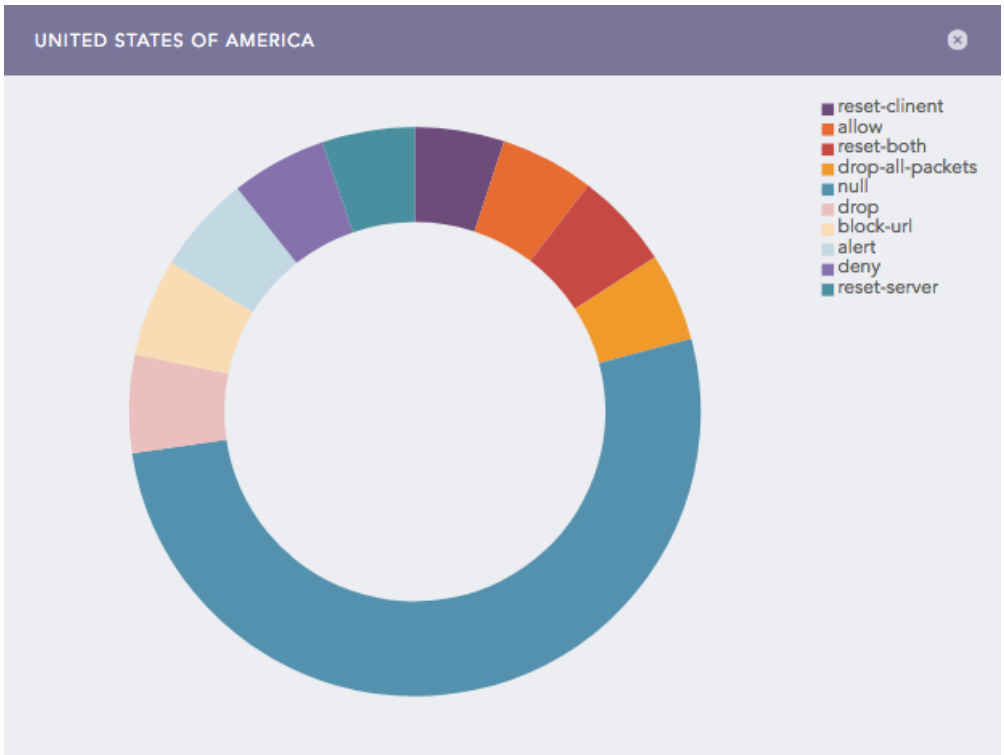
Example:

```
| process geoip(destination_address) as country_name | chart count() by
country_name, action
```

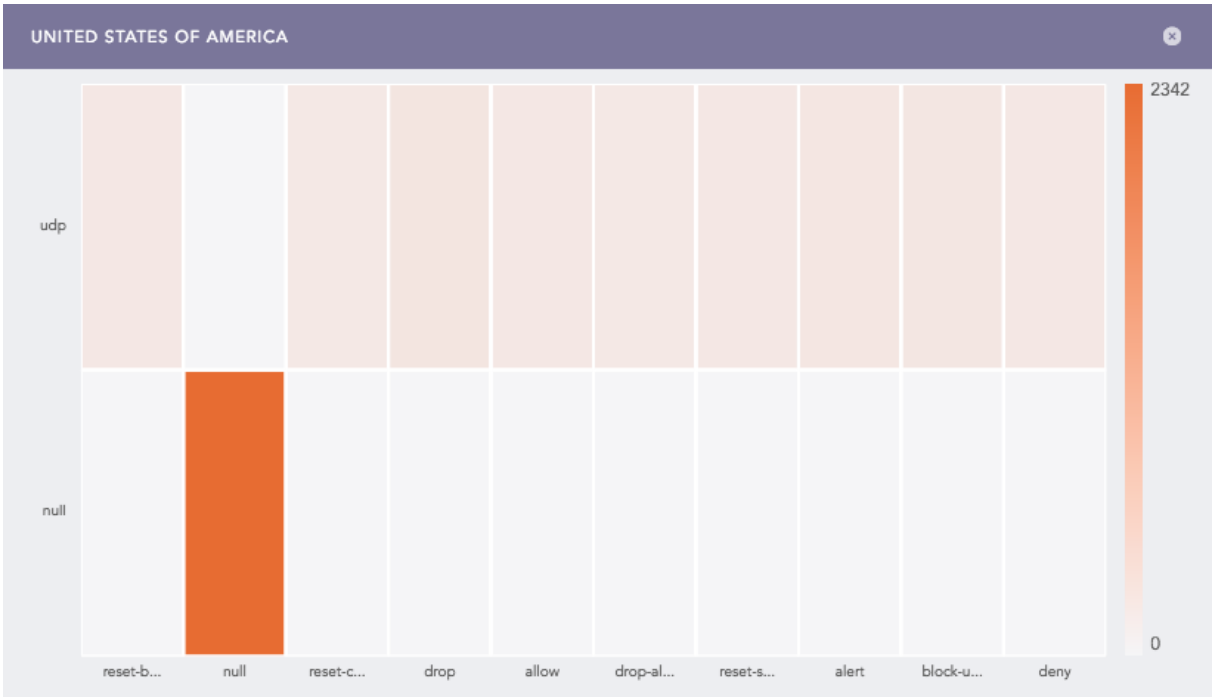


Some notable points about the World Map:

1. Sections of a graph are only clickable if they have some value of aggregation parameter and the search query contains two or three grouping parameter meaning that you cannot click and further drill-down the chart for a query with one grouping parameter or more than three grouping parameters.
2. For search queries with a single aggregation parameter and two grouping parameters, you can view a Donut chart by clicking on any region of a World Map (with some value for the aggregation parameter).



3. For search queries with a Single aggregation parameter and three grouping parameters, you can view a Heatmap by clicking on any region of a World Map(with some value for the aggregation parameter).



NOTE
You can drill-down operations from these sub-charts.

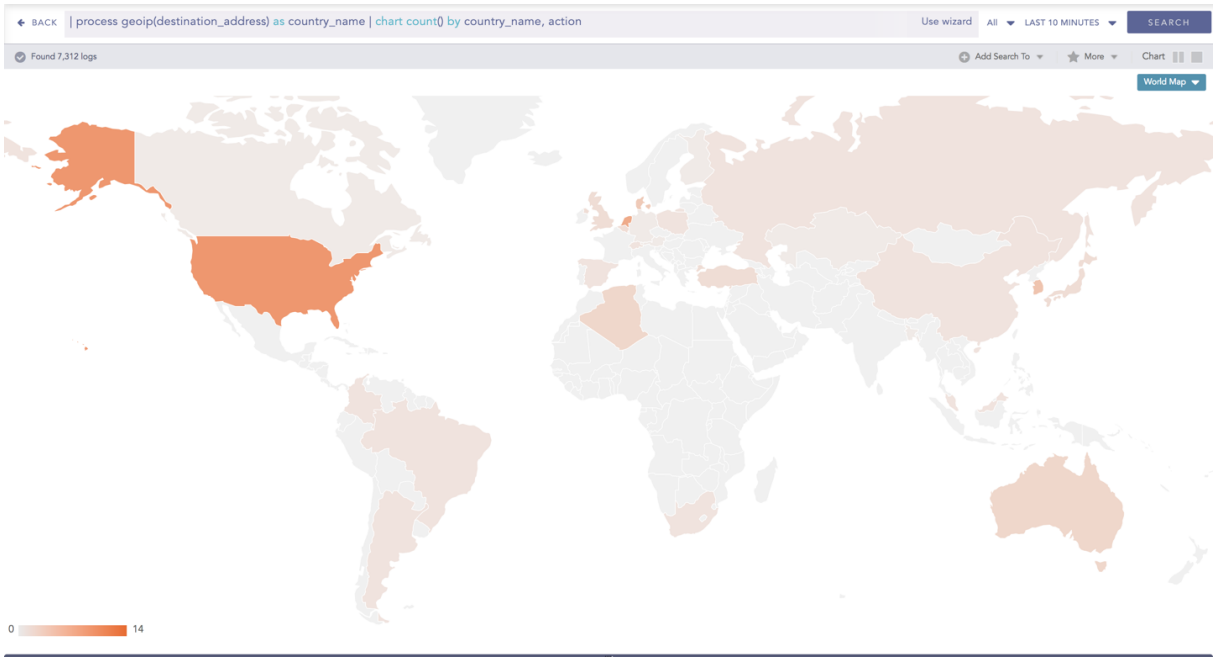
You can refer to Single Aggregation with Grouping for more details.

Multiple Aggregation with Grouping

In Multiple Aggregation with Grouping, the color shade on each region of a World Map displays the value of the first aggregation parameter, i.e., higher the value of the aggregation parameter, darker the color. The values of all other successive aggregation parameters can be viewed using the sub-charts.

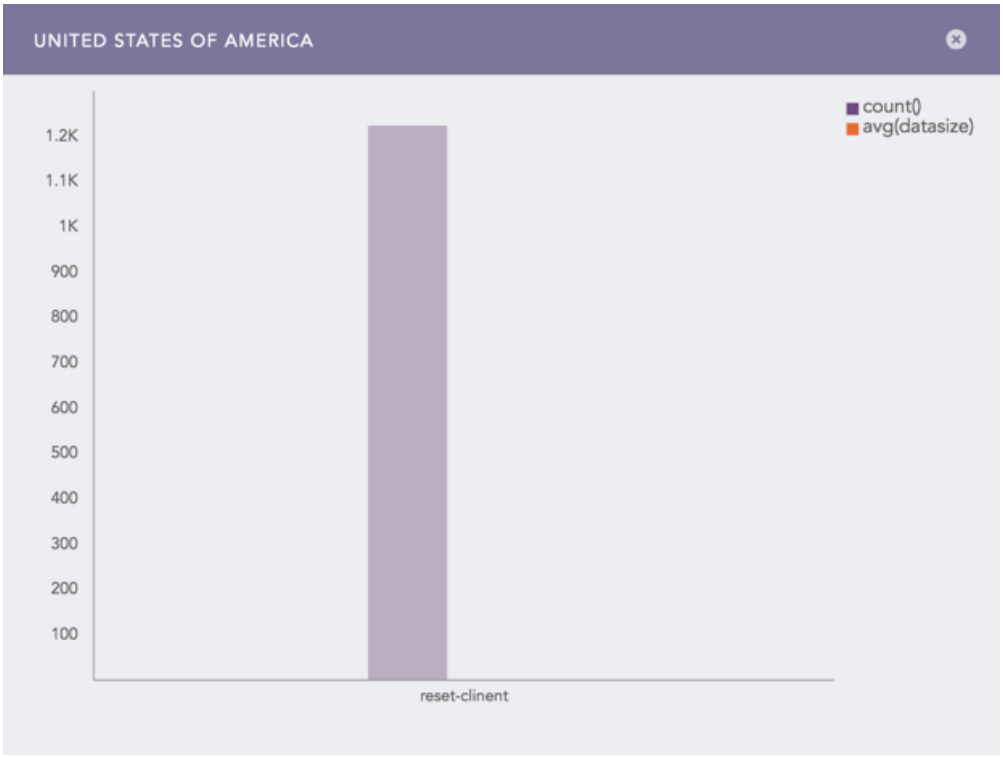
Example:

```
| process geoip(destination_address) as country_name | chart count(), avg
(datasize) by country_name, action
```



NOTE

1. The value of the first grouping parameter can only be depicted in the chart if the search query contains only one grouping parameter. However, the value can be viewed from the **Search Table**.
2. Sections of a graph are only clickable if they have some value of aggregation parameter and the search query contains two or three grouping parameters. Thus, you cannot click and further drill-down the chart for a query with one grouping parameter or more than three grouping parameters.
3. For search queries with multiple aggregation parameters and two or three grouping parameters, you can view a Clustered Column chart by clicking on any region of a World Map (with some value for the aggregation parameter).



NOTE
You can further drill down from these charts.

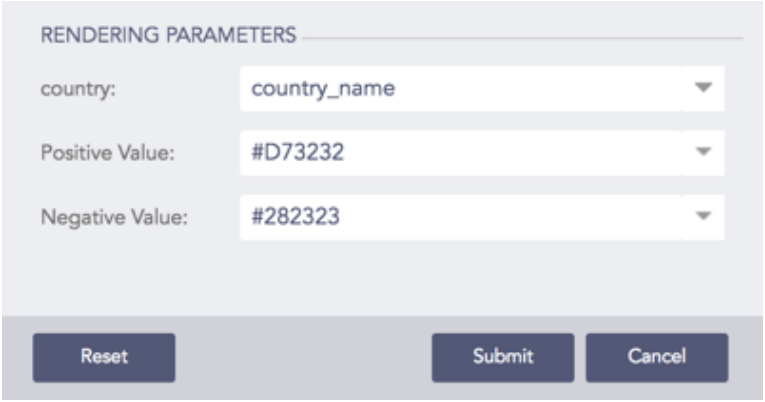
You can refer to Multiple Aggregation with Grouping for more details.

Rendering Parameters

Click the gear icon at the top right corner of the World Map to open the rendering parameters panel.

The Rendering Parameters such as **Country**, **Positive Value**, and **Negative Value** provide a custom settings option to view data in different formats.

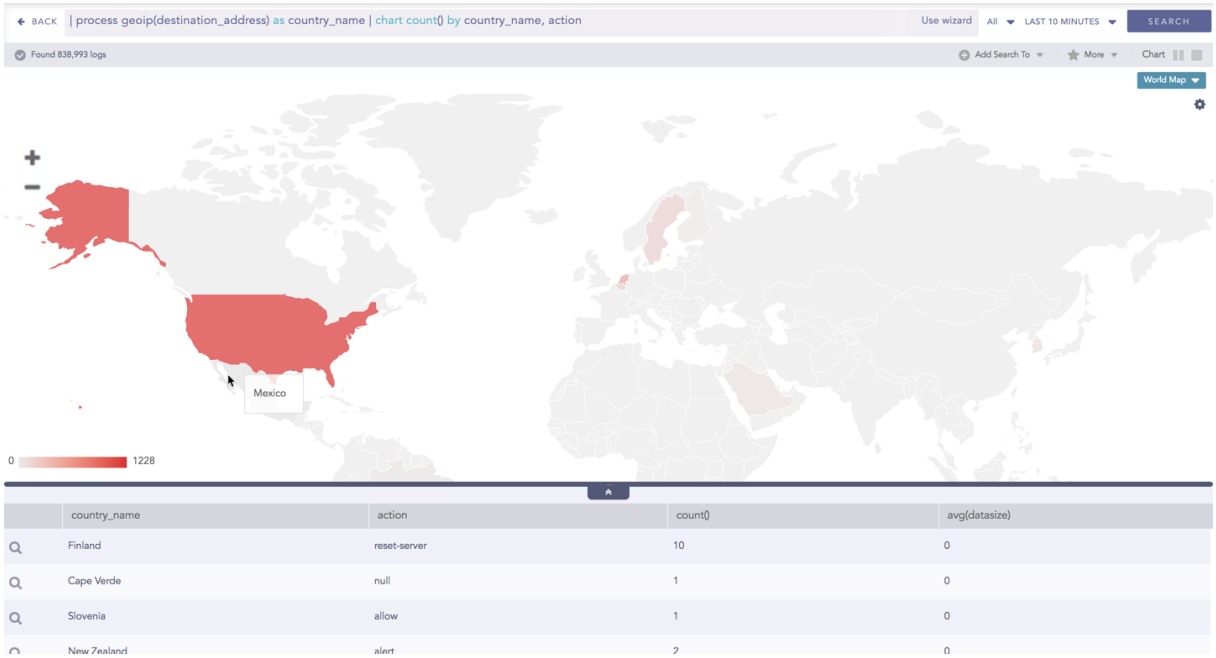
Through the **Country** option, you can specify the grouping parameter containing the names of the countries. Whereas, the **Positive Value** and **Negative Value** options allow you to select the color to represent the positive values of the aggregation parameter and the negative value of the aggregation parameter respectively.



Operations

Pan and Zoom

The **Pan and Zoom** feature allows you to zoom in and out on a specific section on the world map and shift from one section to another.



Drilldown from Search Visualization

SLS provides a number of options for search result visualization. While visualizing the search results or the content of a widget, it is possible to dive deeper into the results by clicking the graphical representation. For example, while viewing a search result which includes the fields such as **destination_address**, **destination_port**, **source_address**, and **source_port** in the search query, it is possible to drill down to the results based on these parameters. Use the keys from the original query to drill down.

Common Features of Drill-down

Depending on the original query chosen to drill down from, the contents in the drill-down context menu varies. There are 3 types of drill-down options in SLS visualization:

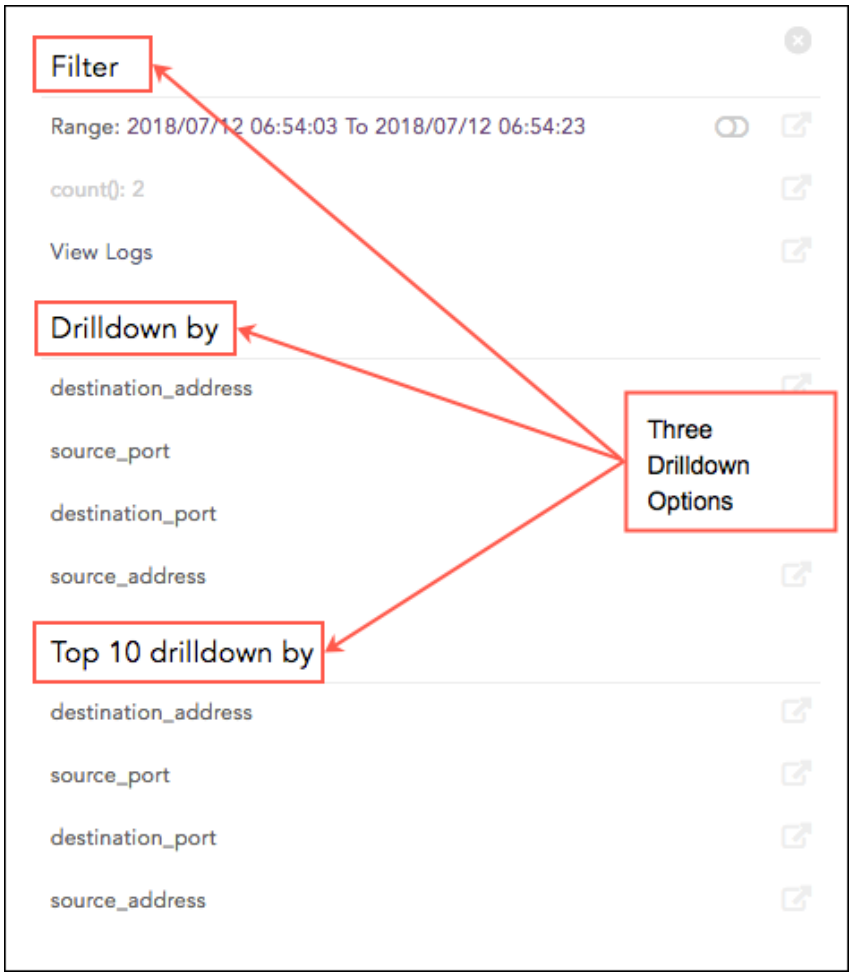
1. Filter
2. Drilldown by
3. Top 10 drilldown by

The **Filter** type drill-down searches on the **Range**, the **Field**, and the **count()**. The **Drilldown by** and the **Top 10 drilldown by** types drill down on the **fields** and the **labels** respectively.

For example:

```
destination_address=* source_port=* destination_port=* source_address=*
```

While performing drill-down from this query, the following context menu appears on the screen. It lists all three possible sections in a drill-down context menu.

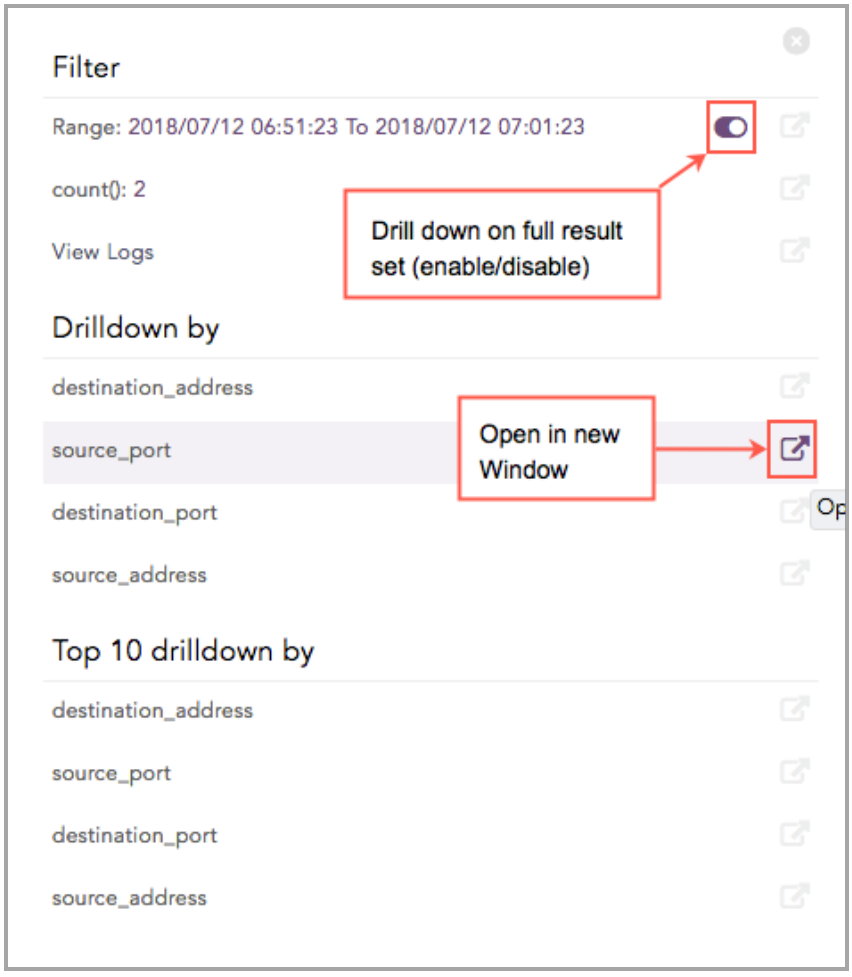


1. **Filter**
   This section contains the following components depending on the original query:
   - **Range**: Displays the subset of the time-period from which you have chosen to drill-down. It is only displayed for queries containing the timechart command or logs plotted in a time series manner.
   - **count()**: Total number of logs.
   - **View Logs**: Lets you view the drilled-down logs. You can view them in the same or a new window by clicking **View Logs** in the context menu for the given time-range.

   > **NOTE**
   > By default, the **Drilldown on full result set** slider and **count()** are disabled (grayed out).

2. **Drilldown by**
   This section contains the fields or labels present in the original query.
3. **Top 10 Drilldown by**
   This section contains the fields or labels present in the original query.

Besides these, the context menu also contains some other options for the following.



Drilldown on Full Result Set

It is possible to drill down on the full result. The slider icon present next to the **Range** value lets you drill down on the full result set in addition to the subset.

Open drilldown in a New Window

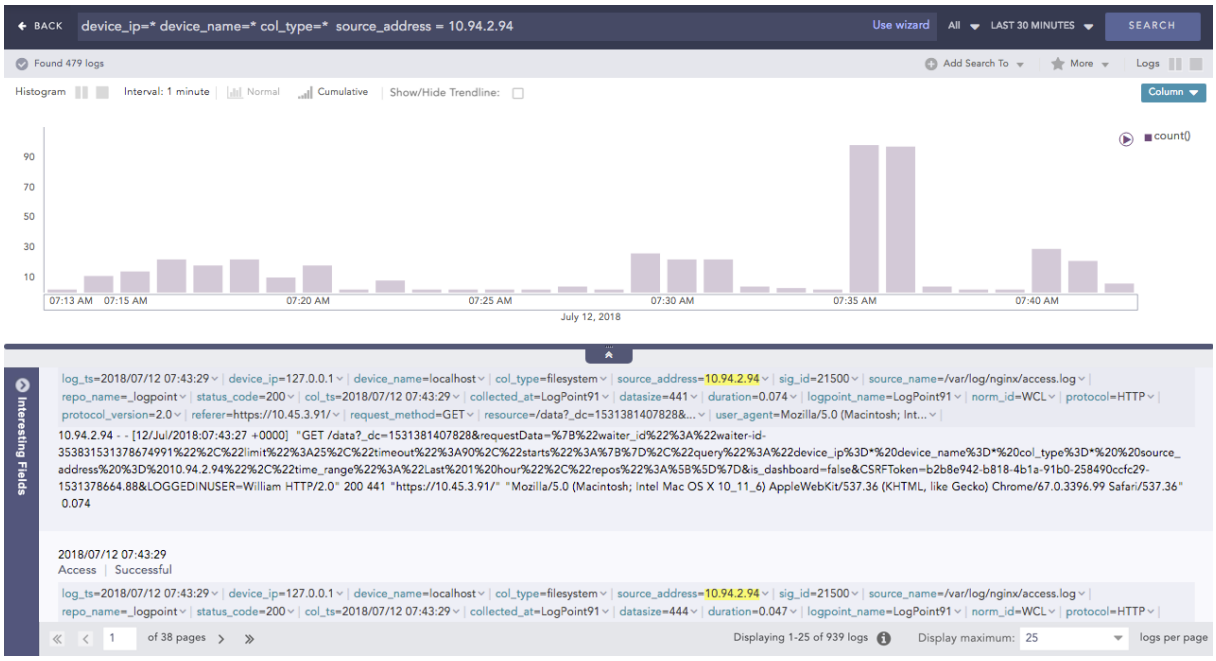
While performing drill-down, it executes in the same window by default. However, you can click the **Open in New Window** icon to open the results in the new window.

Demonstration of Customizable Drilldown from Search Visualization

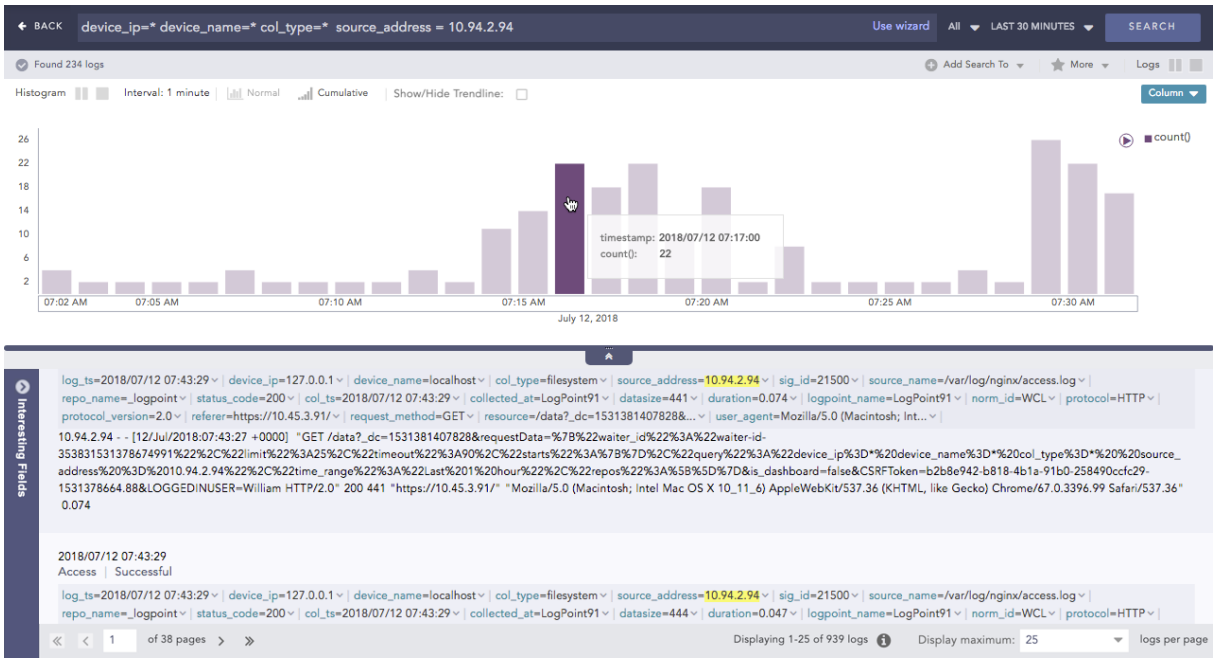
Consider the following search query:

```
device_ip=* device_name=* col_type=* source_address = 10.94.2.94
```

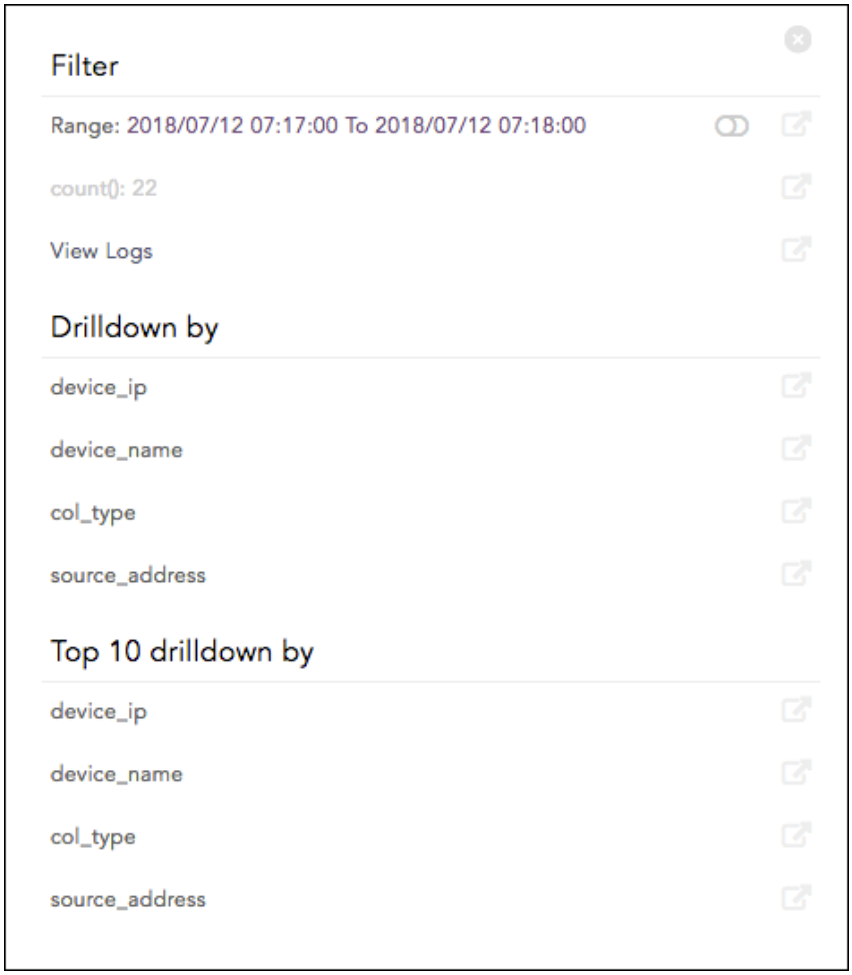
This query displays the following visualization.



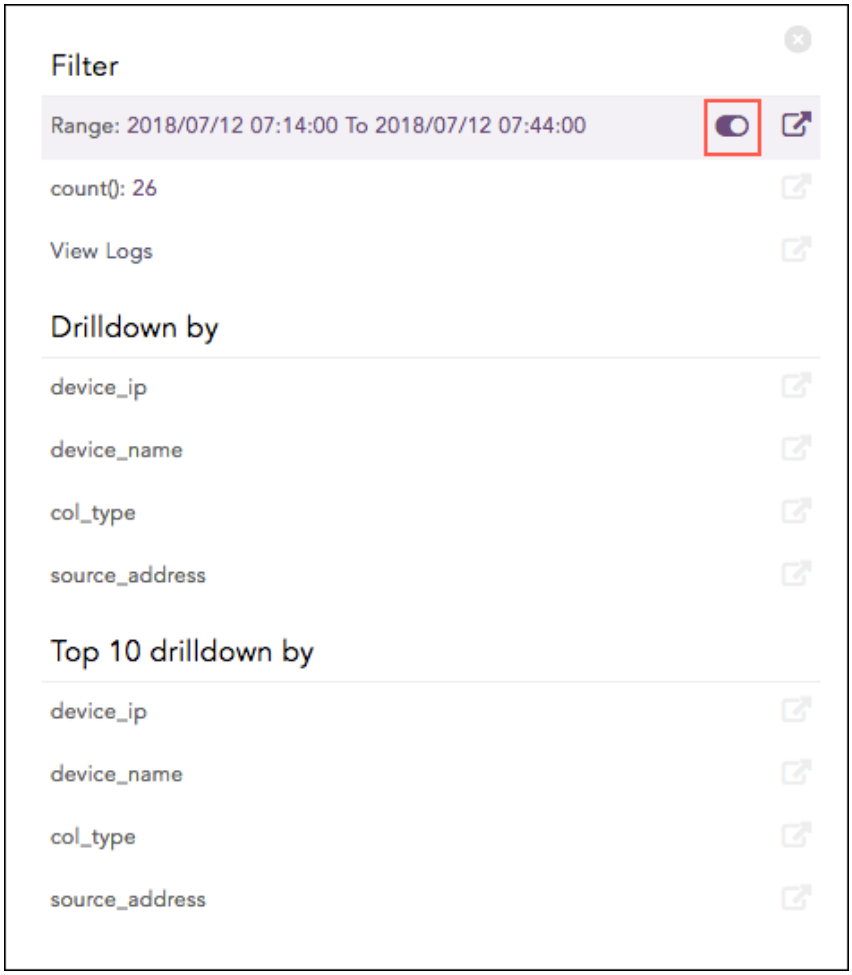
SLS's search result drill-down actions let you dive deeper into the details of the information presented in the visualization. If you hover over the search graph, the related information of the selected area is summarized in a tooltip.

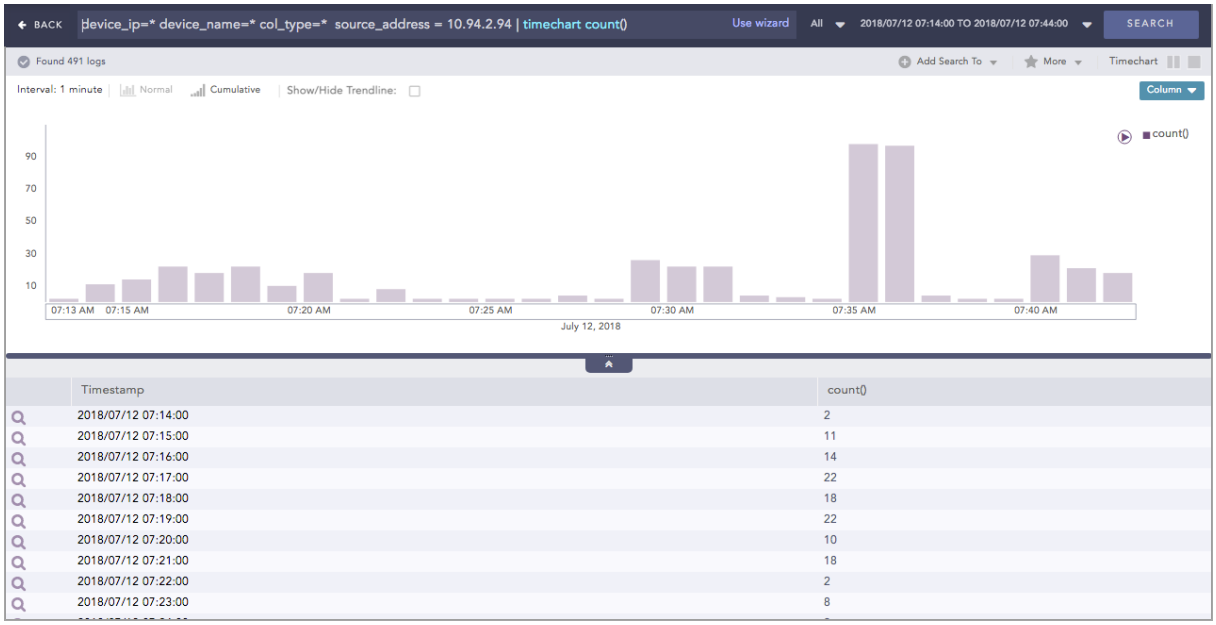


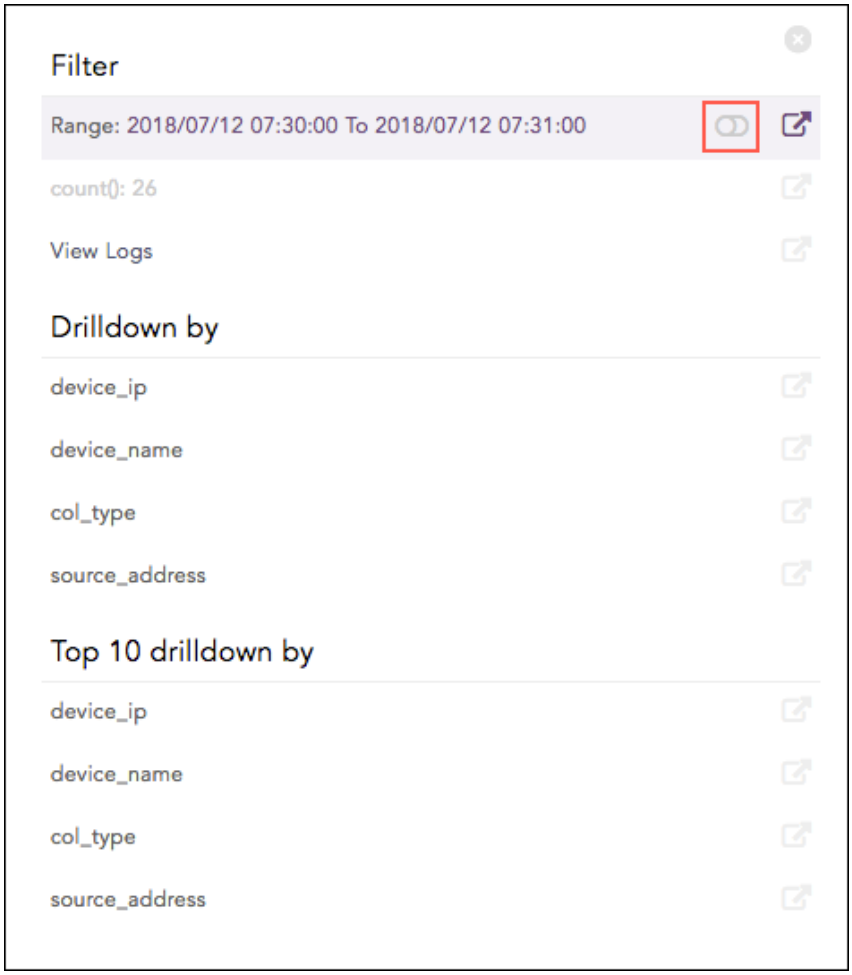
Click the highlighted section of the result.

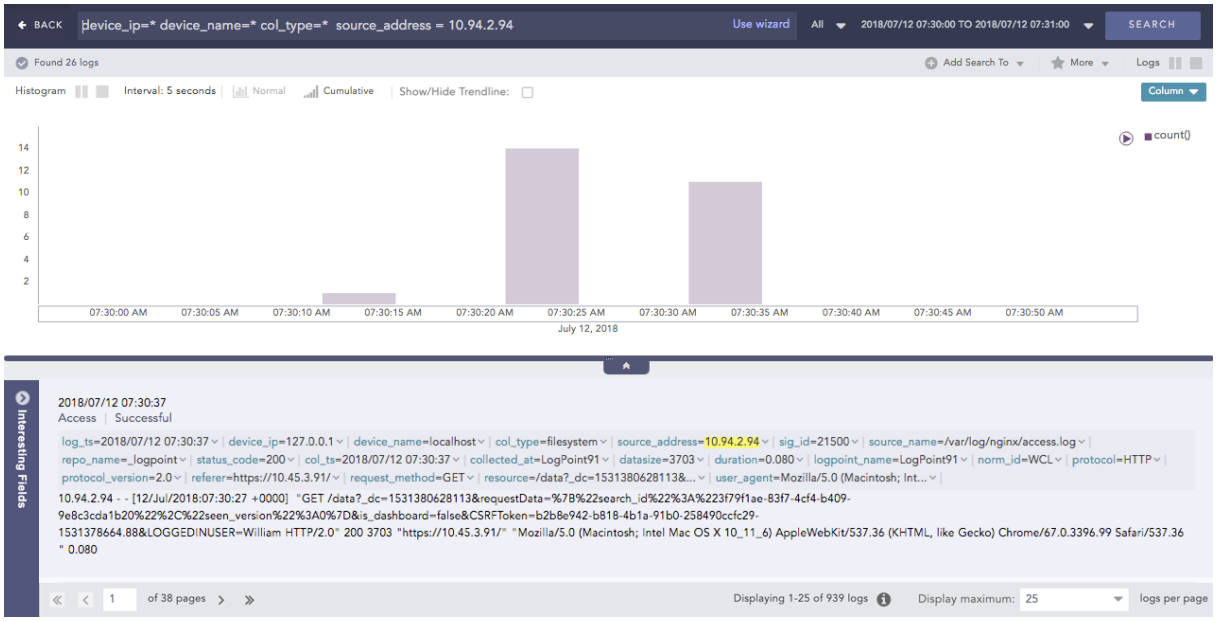


In the context menu, enable or disable the drill-down on the **Range** value by clicking the slider icon. The corresponding search visualization for the **Range** is shown below:

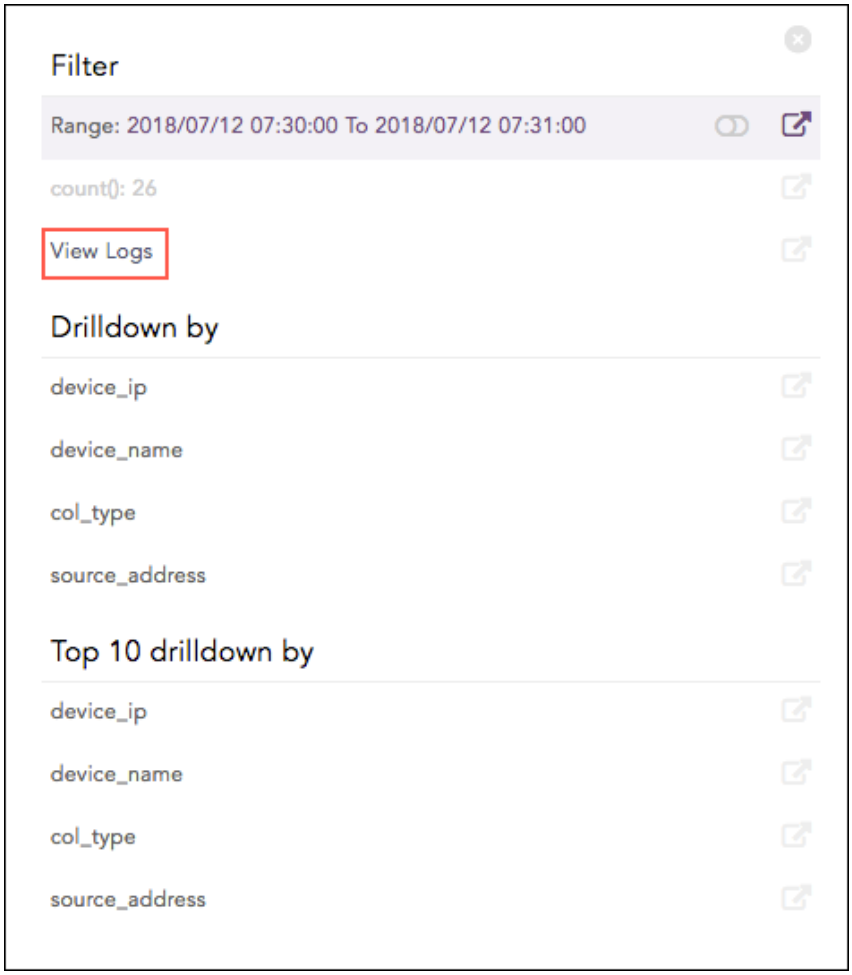

Filter
Range: 2018/07/12 07:14:00 To 2018/07/12 07:44:00
count(): 26
View Logs
Drilldown by
device_ip
device_name
col_type
source_address
Top 10 drilldown by
device_ip
device_name
col_type
source_address

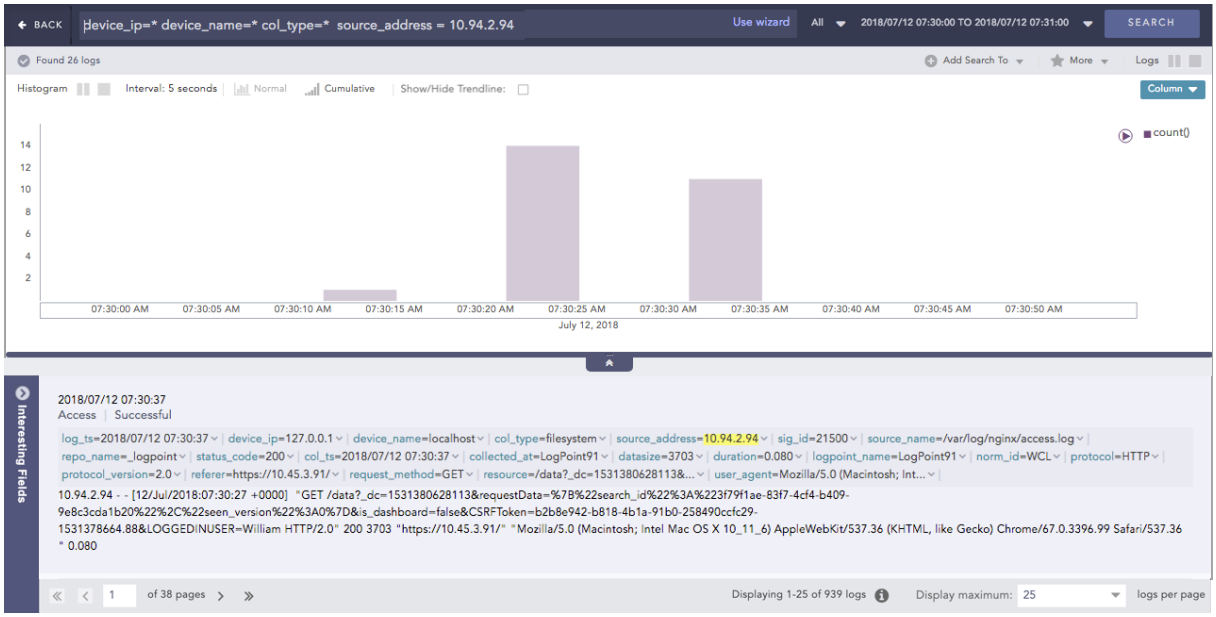

BACK
device_ip=* device_name=* col_type=* source_address = 10.94.2.94 | timechart count()
Use wizard
All
2018/07/12 07:14:00 TO 2018/07/12 07:44:00
SEARCH
Found 491 logs
Add Search To
More
Timechart
Interval: 1 minute
Normal
Cumulative
Show/Hide Trendline:
Column
count()
90
70
50
30
10
07:13 AM
07:15 AM
07:20 AM
07:25 AM
07:30 AM
07:35 AM
07:40 AM
July 12, 2018
Timestamp
count()
2018/07/12 07:14:00 2
2018/07/12 07:15:00 11
2018/07/12 07:16:00 14
2018/07/12 07:17:00 22
2018/07/12 07:18:00 18
2018/07/12 07:19:00 22
2018/07/12 07:20:00 10
2018/07/12 07:21:00 18
2018/07/12 07:22:00 2
2018/07/12 07:23:00 8



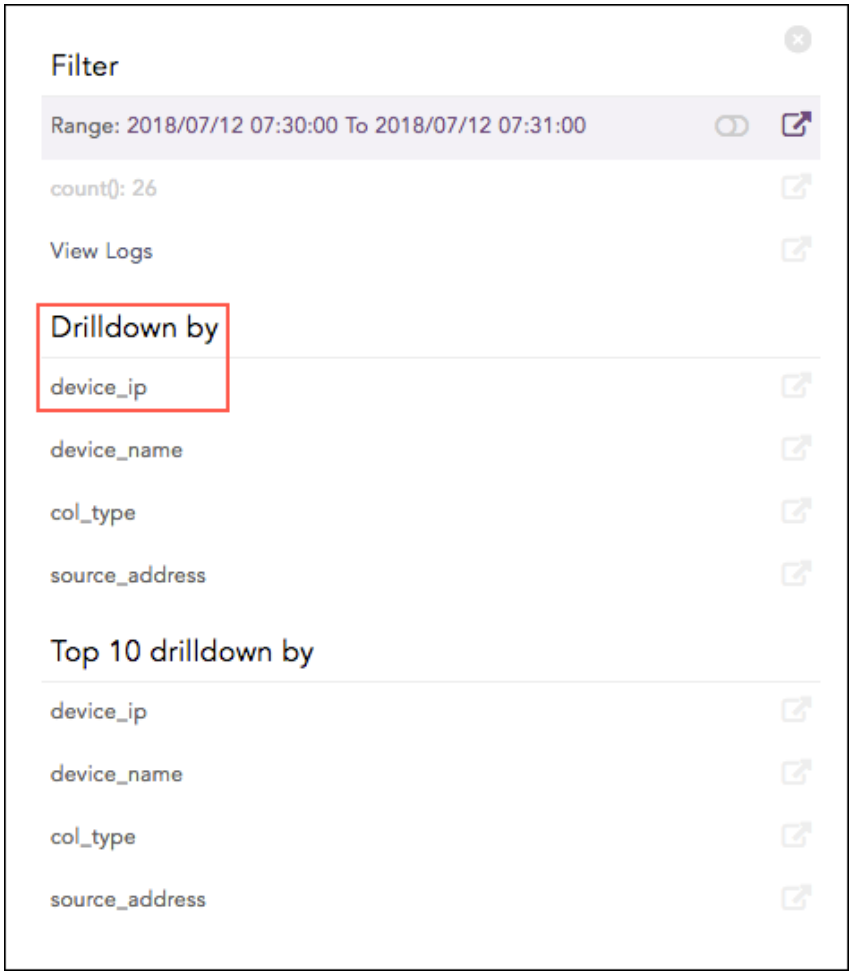


Click **View Logs** to see the corresponding log results. The results can be viewed in the same window or in a new one.





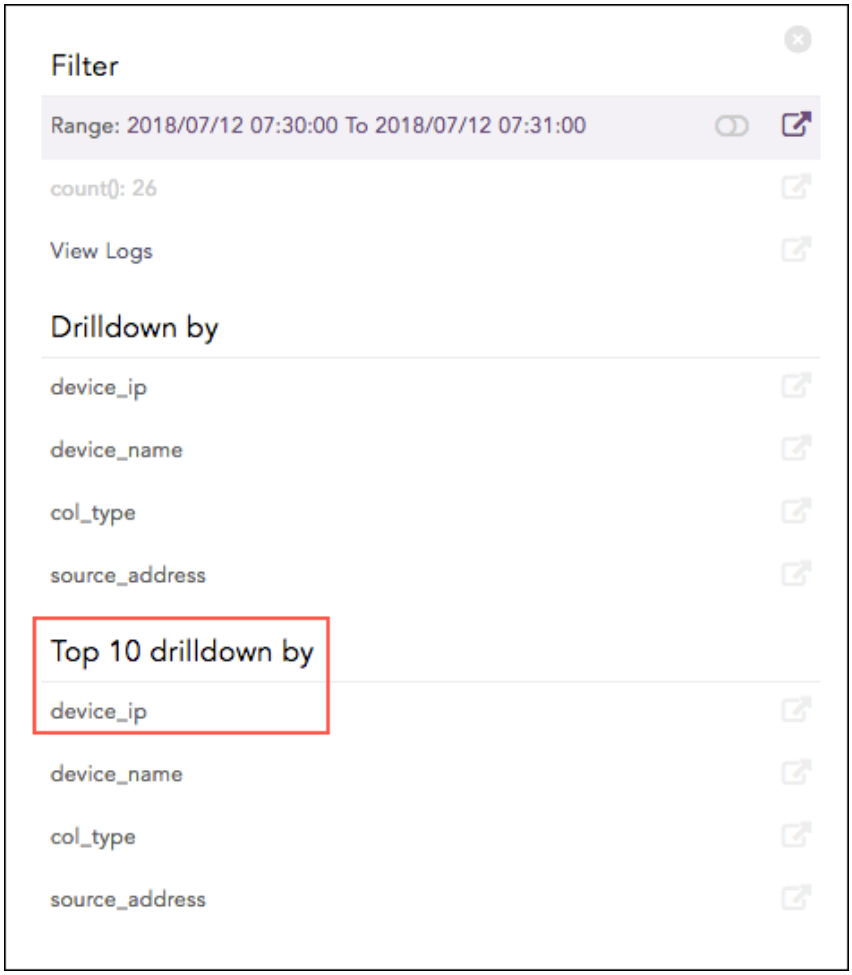
Click the required **Field-values** in the **Drilldown by** section to see the corresponding search results. The results can be viewed in the same window or in a new one.



Click the **device_ip** in the **Drilldown by** section to append **chart count() by device_ip order by count() desc** in the search query. The search result can be viewed in the same window or in a new one.

```
device_ip=* device_name=* col_type=* source_address = 10.94.2.94 | chart
count() by device_ip order by count() desc
```

Click the required **labels** in the **Top 10 drilldown by** section to see the corresponding search results. These results can be viewed in the same window or in a new one.



Click **device_ip** in the **Top 10 Drilldown by** section to append **| chart count() by device_ip order by count() limit 10 desc** to the search query. Choosing **device_ip** results in the following query.

```
device_ip=* device_name=* col_type=* source_address = 10.94.2.94 | chart
count() by device_ ip order by count() desc limit 10
```

Similarly, the search results can be drilled down on the basis of the **source_port**, **destination_port**, and the **source_address**.

The search result can be further drilled down by clicking any part of the result set.

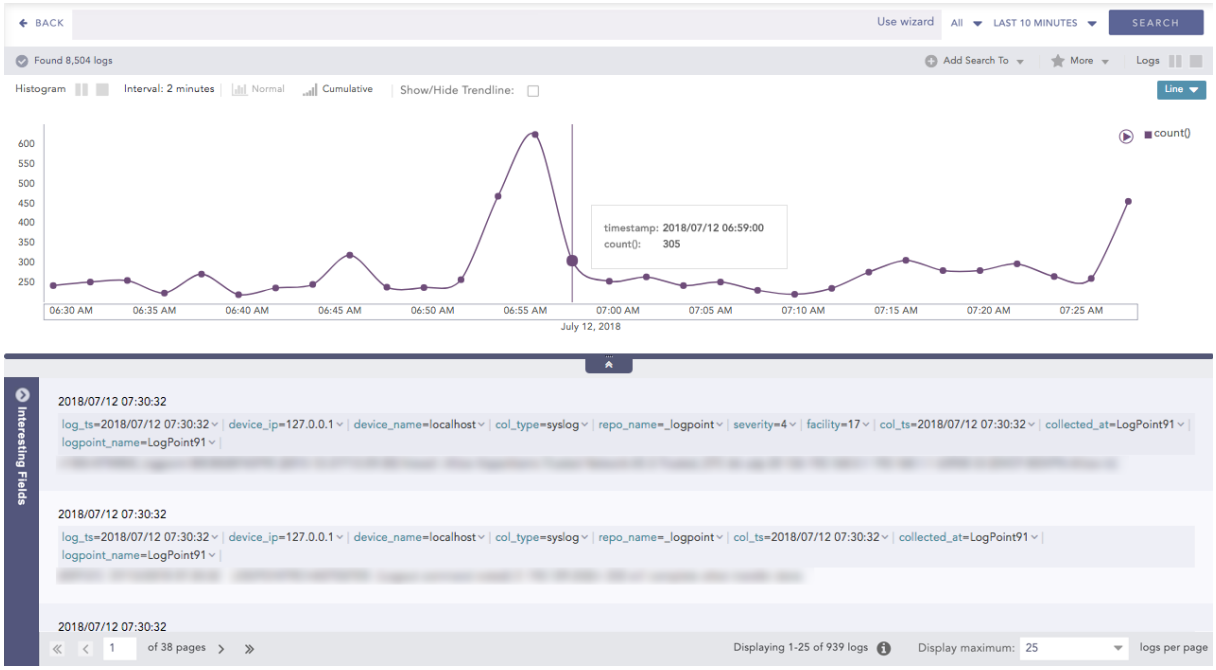
```
device_ip=127.0.0.1 device_name=* col_type=* source_address = 10.94.2.94 |
chart count() by device_ name order by count() desc
```

Special Drilldown Scenarios

Filter Drilldown

Example 1

For **Filter Drilldown**, if you drill down on the **Range** and open the results in the same page, the search is executed in the selected time-range. If you open the search in a new window, it is executed in the selected time-range with **| timechart count()** appended to it. The command is appended only for simple queries.



Select a bar to drill down from. The following context menu appears.

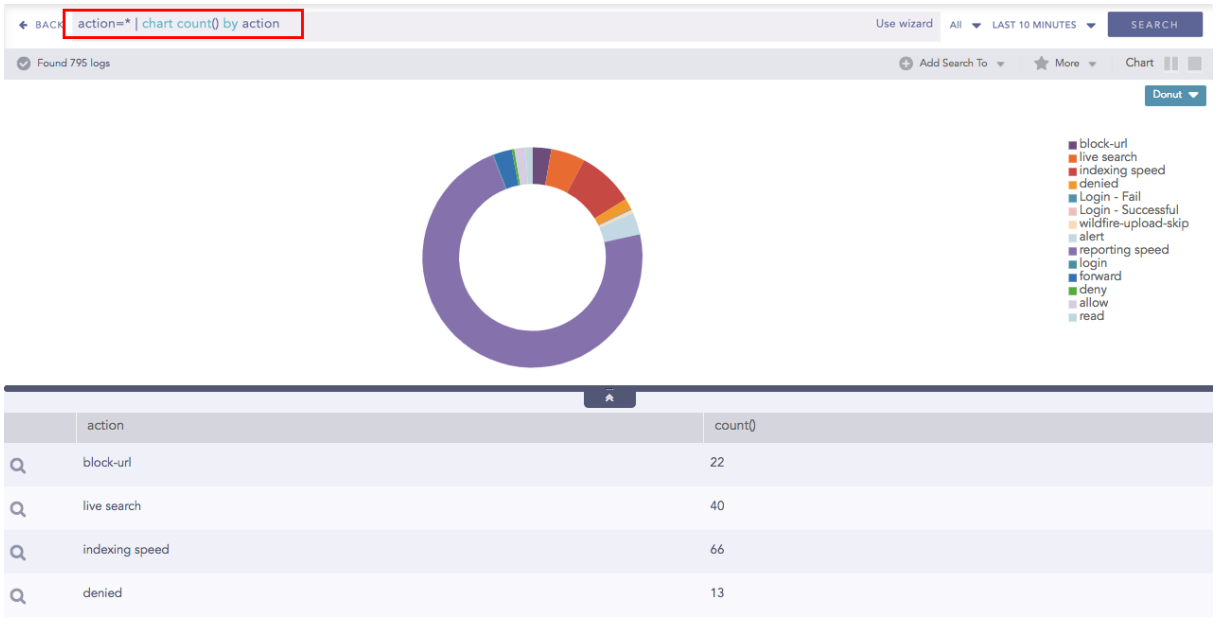
Once you drill down, you can see results similar to the following example.

For the filter type, when the drill-down is executed on **Field**, search is executed with **| filter <field> = <value>**

Consider the following query:

```
action=*|chart count() by action
```

The following visualization appears.



	action	count()
	block-url	22
	live search	40
	indexing speed	66
	denied	13

If you drill down on the **reporting Speed**, the following context menu appears.

If you drill down on the **reporting speed**, the appended search query is:

```
action=* | chart count() by action | filter "action"="reporting speed"
```

Example 2

When the drill-down is executed on **count()** for the **Filter** type, the search is executed with **| search count() = <value>**. Consider the following example:

```
action=* | chart count() by action
```

The following visualization appears.

The context menu for this drilldown is:

When the drill-down is executed on **count(): 544**, the new appended query is:

```
action=*|chart count() by action | filter "count()" = 544
```

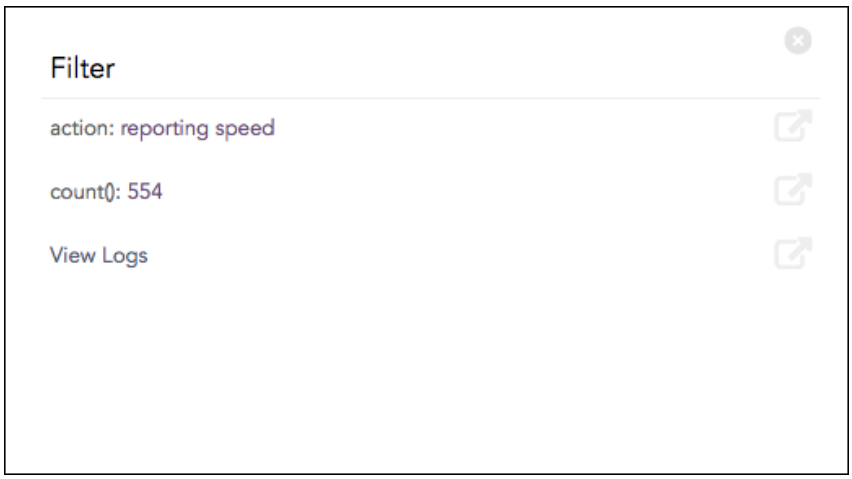
Example 3

When the drill-down is conducted for **<empty query> | chart count() by group**, the customizable drill-down options differ. Consider the following:

```
| chart count() by action
```

The following visualization appears.

Clicking drill-down for a bar opens up the following context menu. In this case, only the Filter section with **field, count()** and **View Logs** is displayed as shown.



If you click "action: reporting speed", the new query becomes:

```
| chart count() by action | filter "action"="reporting speed"
```

If you click "count(): 544", the new query becomes:

```
| chart count() by action | filter "count()"=544
```

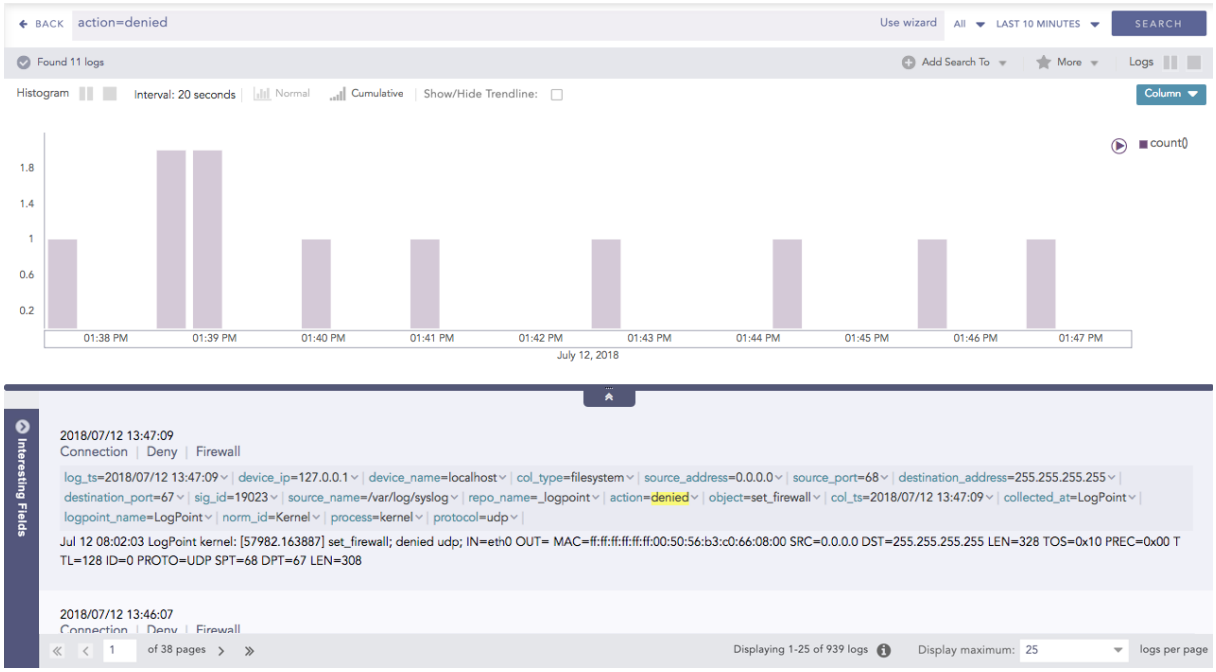
Drilldown by

For **Drilldown by**, when the drill-down is executed on **fields** or **label**, search is executed with the given query followed by **| chart count() by <field> order by count() desc**

For example:

```
action = denied
```

The following visualization appears. Hover over the required result and click to drill down.



In the **Drilldown Context Menu**, click **action** under the Drilldown by section.

The search results of the drilldown appear.

New query:

```
action = denied| chart count() by action order by count() desc
```

Top 10 Drilldown by

For **Top 10 Drilldown by**, when you execute the drill-down on **field-values** or **label**, the search is executed with the given query followed by **| chart count() by <field> order by count() desc limit 10**.

Execute a query and click the search result visualization to dive deeper. In the context menu, click the field under the Top 10 Drilldown by section. The search result of the drill-down appears on the screen.

New query:

```
action = denied| chart count() by action order by count() desc limit 10
```

Further reading

Additional information and answers to questions you may have about SLS are available in the Stormshield knowledge base (authentication required).

STORMSHIELD

documentation@stormshield.eu