

SIEM Visualization Example 2: Failed Logon Attempts (Disabled Users)

[? Go to Questions](#)

In this SIEM visualization example we want to create visualization to monitor failed login attempts against disabled users.

We mention "failed" because it is not possible to log in with a disabled user, so it will never be successful even if the correct credentials are provided. In a scenario where the correct credentials are provided, the Windows logs will contain an additional SubStatus value of 0xC0000072, that indicates the reason of the failure.

Navigate to the bottom of this section and click on [Click here to spawn the target system!](#).

Navigate to [http://\[Target IP\]:5601](http://[Target IP]:5601), click on the side navigation toggle, and click on "Dashboard".

A prebaked dashboard should be visible. Let's click on the "pencil"/edit icon.

Now, to initiate the creation of our first visualization, we simply have to click on the "Create visualization" button.

Upon initiating the creation of our first visualization, the following new window will appear with various options and settings.

There are four things for us to notice on this window:

1. A filter option that allows us to filter the data before creating a graph. In this case our goal is to display failed logon attempts against disabled users only. We can use a filter to only consider event IDs that match [4625 - Failed logon attempt on a Windows system](#), like we did in the previous visualization example. In this case though, we should also take into account the SubStatus ([winlog.event_data.SubStatus](#) field) that indicates, when set to 0xC0000072, that the failure is due to a logon with disabled user. The following image demonstrates how we can specify such a filter.

Table of Contents

SIEM & SOC Fundamentals

- [SIEM Definition & Fundamentals](#)
- [Introduction To The Elastic Stack](#)
- [SOC Definition & Fundamentals](#)
- [MITRE ATT&CK & Security Operations](#)
- [SIEM Use Case Development](#)

SIEM Visualization Development

- [SIEM Visualization Example 1: Failed Logon Attempts \(All Users\)](#)
- [SIEM Visualization Example 2: Failed Logon Attempts \(Disabled Users\)](#)
- [SIEM Visualization Example 3: Successful RDP Logon Related To Service Accounts](#)
- [SIEM Visualization Example 4: Users Added Or Removed From A Local Group \(Within A Specific Timeframe\)](#)

Alert Triaging

- [The Triaging Process](#)

Skills Assessment

- [Skills Assessment](#)

My Workstation

The screenshot shows the Kibana Settings interface for creating a new search query. The search bar contains the following query:

```
Field: winlog.event_data.SubStatus Operator: is Value: 0x00000072
```

Below the search bar, there are sections for "Available fields" and "Meta fields". The "Available fields" section includes fields like @timestamp and agent.ephemeral_id. A "Create custom label?" checkbox is also present.

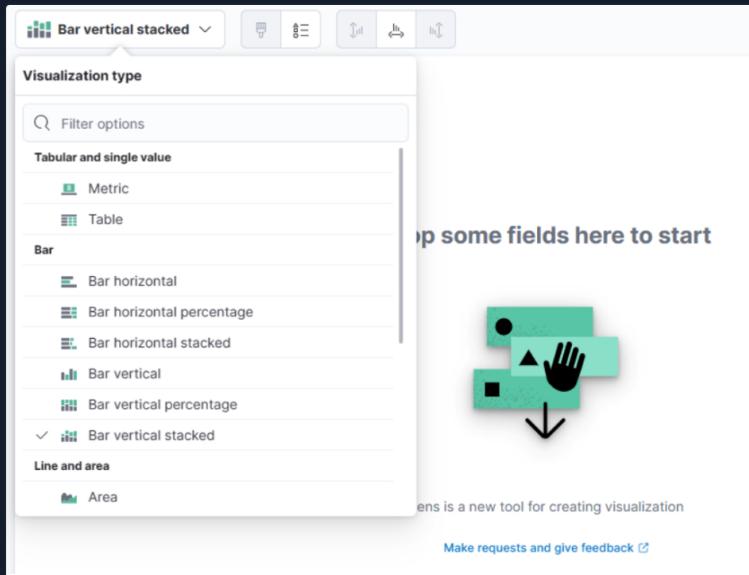
2. This field indicates the data set (index) that we are going to use. It is common for data from various infrastructure sources to be separated into different indices, such as network, Windows, Linux, etc. In this particular example, we will specify `windows*` in the "Index pattern".
3. This search bar provides us with the ability to double-check the existence of a specific field within our data set, serving as another way to ensure that we are looking at the correct data. Like in the previous visualization, we are interested in the `user.name.keyword` field. We can use the search bar to quickly perform a search and verify if this field is present and discovered within our selected data set. This allows us to confirm that we are accessing the desired field and working with accurate data.

The screenshot shows the Kibana Dashboard interface. At the top, there is a search bar with the query `event.code: 4625` and a "+ Add filter" button. Below the search bar, there is a search input field containing `windows*`. Further down, there is a search input field with the placeholder `user.|`, which is highlighted with a red border. Below these search fields, there is a "Filter by type" dropdown set to 0. The main area displays a list of "Available fields" with 4 items:

- `related.user.keyword`
- `user.domain.keyword`
- `user.id.keyword`
- `user.name.keyword` (This item is also highlighted with a red border)

Below the available fields, there are sections for "Empty fields" (15) and "Meta fields" (0).

4. Lastly, this drop-down menu enables us to select the type of visualization we want to create. The default option displayed in the earlier image is "Bar vertical stacked". If we click on that button, it will reveal additional available options (image redacted as not all options fit on the screen). From this expanded list, we can choose the desired visualization type that best suits our requirements and data presentation needs.



For this visualization, let's select the "Table" option. After selecting the "Table", we can proceed to click on the "Rows" option. This will allow us to choose the specific data elements that we want to include in the table view.

A screenshot of the 'Table' configuration interface. At the top, it says 'Table'. Below that is a search bar with 'windows*' in it. The main area is divided into three sections: 'Rows' (highlighted with a red box), 'Columns', and 'Metrics'. Each section has a box with a plus sign and the text 'Add or drag-and-drop a field'. The 'Rows' section is currently active, indicated by a red border around its box.

Let's configure the "Rows" settings as follows.

Rows

X

Select a function

Date histogram Intervals **Top values**

Select a field

user.name.keyword

Number of values **1000**

Rank by **Count of records**

Rank direction **Descending**

> Advanced

Display name **Top values of user.name.keyword**

Text alignment **Left** Center Right

Hide column

Moving forward, let's close the "Rows" window and proceed to enter the "Metrics" configuration.

Table

windows*

Rows **Top values of user.name.keyword**

+ Add or drag-and-drop a field

Columns

+ Add or drag-and-drop a field

Metrics

+ **Add or drag-and-drop a field**

Required dimension

In the "Metrics" window, let's select "count" as the desired metric.

In the "Metrics" window, let's select "Count" as the desired metric.

The screenshot shows the "Metrics" window with a red box highlighting the title bar. Below it, there are two tabs: "Quick functions" (which is selected) and "Formula". Under "Select a function", the "Count" option is highlighted with a red box. Other options include Average, Median, Minimum, Counter rate, Moving average, Cumulative sum, Percentile, Differences, Sum, Last value, Unique count, and Maximum. At the bottom, there is a "Select a field" section with a dropdown menu labeled "Field".

One final addition to the table is to include another "Rows" setting to show the machine where the failed logon attempt occurred. To do this, we will select the `host.hostname.keyword` field, which represents the computer reporting the failed logon attempt. This will allow us to display the hostname or machine name alongside the count of failed logon attempts, as shown in the image.

The screenshot shows a "Table" visualization. On the left, there is a table with three columns: "Top values of user.name.keyword", "Top values of host.hostname.keyword", and "Count of records". The first row shows "arnli" and "WS001" with a count of 1. On the right, there is a "Rows" configuration panel with two dropdown menus: "Top values of user.name.keyword" and "Top values of host.hostname.keyword", both of which are highlighted with red boxes. Below the dropdowns are sections for "Add or drag-and-drop a field" under "Columns" and "Rows".

Now we can see three columns in the table, which contain the following information:

1. The disabled user whose credentials generated the failed logon attempt event.
2. The machine on which the logon attempt occurred.
3. The number of times the event has occurred (based on the specified time frame or the entire data set, depending on the settings).

Finally, click on "Save and return", and you will observe that the new visualization is added to the dashboard.

Please allow 3-5 minutes for Kibana to become available after spawning the target of the questions below.

VPN Servers

⚠ Warning: Each time you "Switch", your connection keys are regenerated and you must re-download your VPN connection file.

All VM instances associated with the old VPN Server will be terminated when switching to a new VPN server.

Existing PwnBox instances will automatically switch to the new VPN server.

US Academy 3

Recommended

Medium Load

PROTOCOL

UDP 1337

TCP 443

DOWNLOAD VPN CONNECTION FILE



Connect to Pwnbox

Your own web-based Parrot Linux instance to play our labs.

Pwnbox Location

UK

161ms

Terminate Pwnbox to switch location

Start Instance

∞ / 1 spawns left

Waiting to start...

Enable step-by-step solutions for all questions  

Questions

Answer the question(s) below to complete this Section and earn cubes!

 Download VPN Connection File

Target(s): [Click here to spawn the target system!](#)

+ 1  Navigate to [http://\[Target IP\]:5601](http://[Target IP]:5601), click on the side navigation toggle, and click on "Dashboard". Either create a new visualization or edit the "Failed logon attempts [Disabled user]" visualization, if it is available, so that it includes failed logon attempt data related to disabled users including the logon type.

What is the logon type in the returned document?

Interactive

 Submit

+ 1  Navigate to [http://\[Target IP\]:5601](http://[Target IP]:5601), click on the side navigation toggle, and click on "Dashboard". Either create a new visualization or edit the "Failed logon attempts [Admin users only]" visualization, if it is available, so that it includes failed logon attempt data where the username field contains the keyword "admin" anywhere within it. What should you specify after user.name: in the KQL query?

admin

 Submit

[← Previous](#) [Next →](#)

 [Mark Complete & Next](#)

