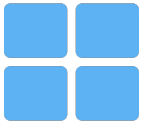
- KIBANA - VISUALIZATION

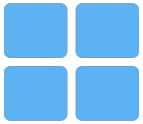
KIBANA LENS



KIBANA INTRODUCTION

Visualizing our data

Kibana Lens is an easy-to-use, intuitive UI that simplifies the process of data visualization through a drag-and-drop experience. Whether we're exploring billions of logs or spotting trends from our website traffic, Lens gets we from data to insights in just a few clicks — no prior experience in Kibana required.



KIBANA INTRODUCTION

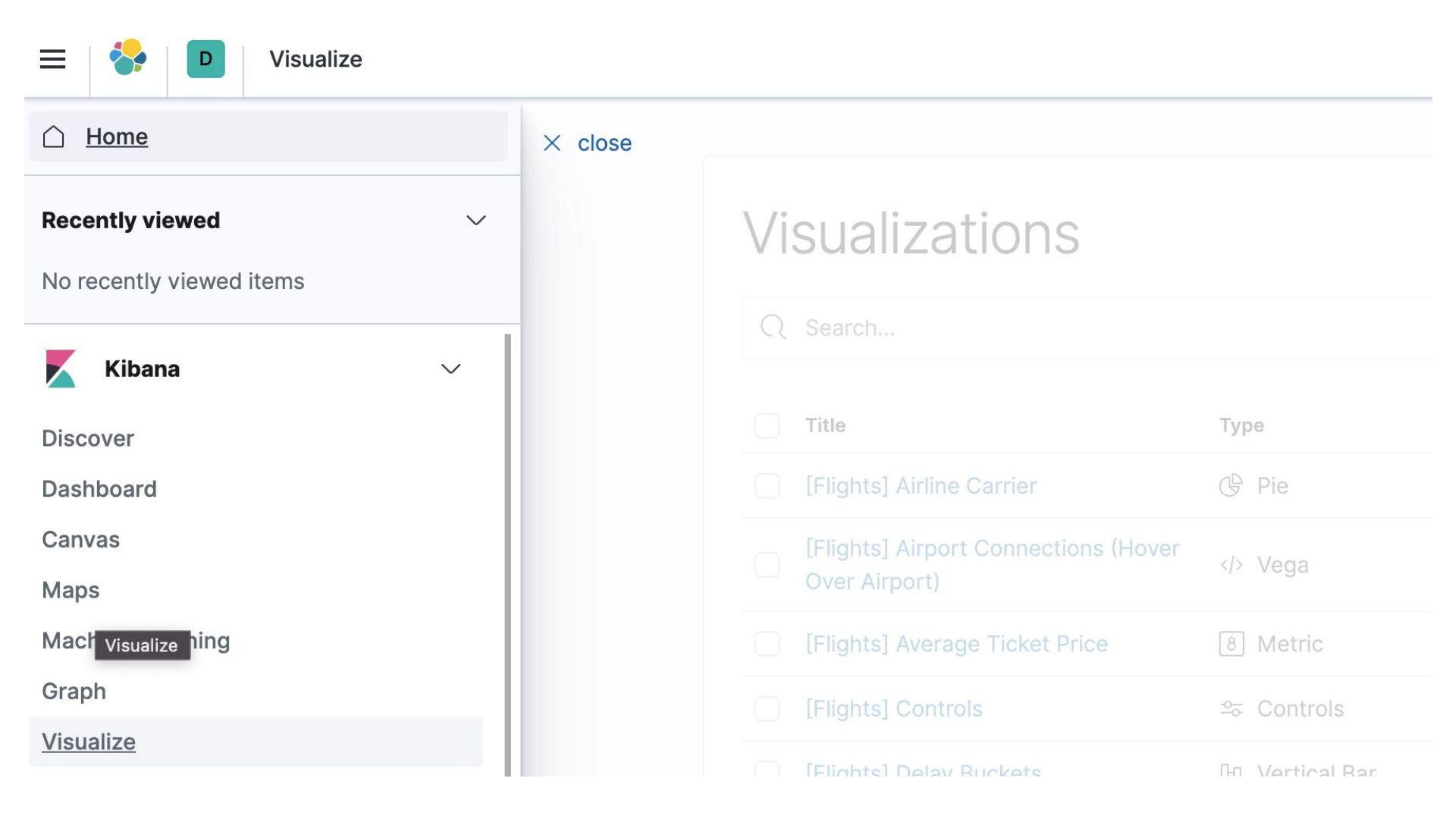
Visualizing our data

A few key benefits of Kibana Lens include:

- Convenient features for fields such as:
  - Showing their distribution of values
  - Searching fields by name for quickly tracking down the data we want
- Quick aggregation metrics like:
  - min, max, average, sum, count, and unique count
- Switching between multiple chart types after the fact, such as:
  - bar, area, line, and stacked charts
- The ability to drag and drop any field to get it immediately visualized or to breakdown the existing chart by its values
- Automatic suggestions on other possible visualization types
- Showing the raw data in data tables
- Combining the visualization with searching and filtering capabilities
- Combining data from multiple index patterns
- And quickly saving the visualization allowing for easy dashboard composition

PREPARATION

Make sure we've added
SERVER LOGS & FLIGHT SAMPLE DATA

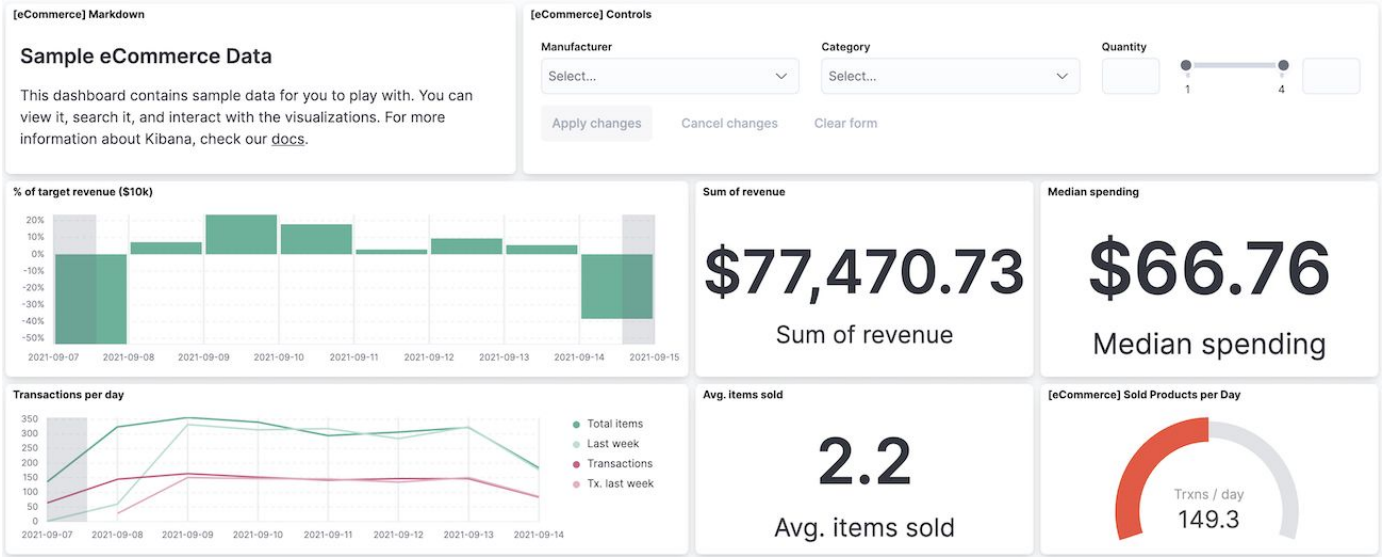

D
Visualize
Home
Recently viewed
No recently viewed items
Kibana
Discover
Dashboard
Canvas
Maps
Visualize
Graph
Visualize
close
Visualizations
Search...
Title
Type
[Flights] Airline Carrier
Pie
[Flights] Airport Connections (Hover Over Airport)
Vega
[Flights] Average Ticket Price
Metric
[Flights] Controls
Controls

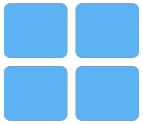
KIBANA INTRODUCTION

Visualizing our data

Visualize our data with dashboards.

The best way to understand our data is to visualize it. With dashboards, we can turn our data from one or more data views into a collection of panels that bring clarity to our data, tell a story about our data, and allow us to focus on only the data that's important to us.





KIBANA INTRODUCTION

Visualizing our data

Visualize our data with dashboards.

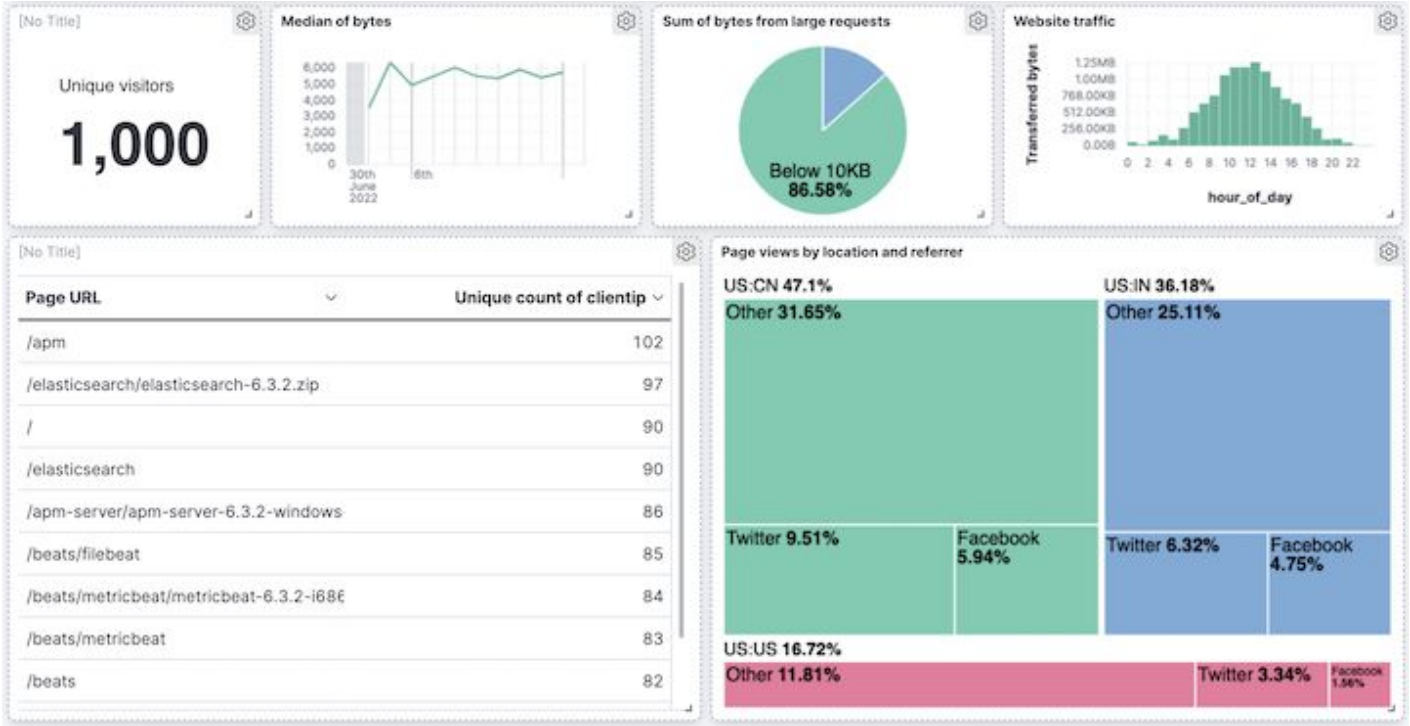
Panels display our data in charts, tables, maps, and more, which allow we to compare our data side-by-side to identify patterns and connections. Dashboards support several types of panels to display our data, and several options to create panels.

- KIBANA -
CREATING OUR FIRST DASHBOARD

KIBANA INTRODUCTION

Visualizing our data

Follow through : when we're done, we'll have a complete overview of the sample web logs data.



KIBANA INTRODUCTION

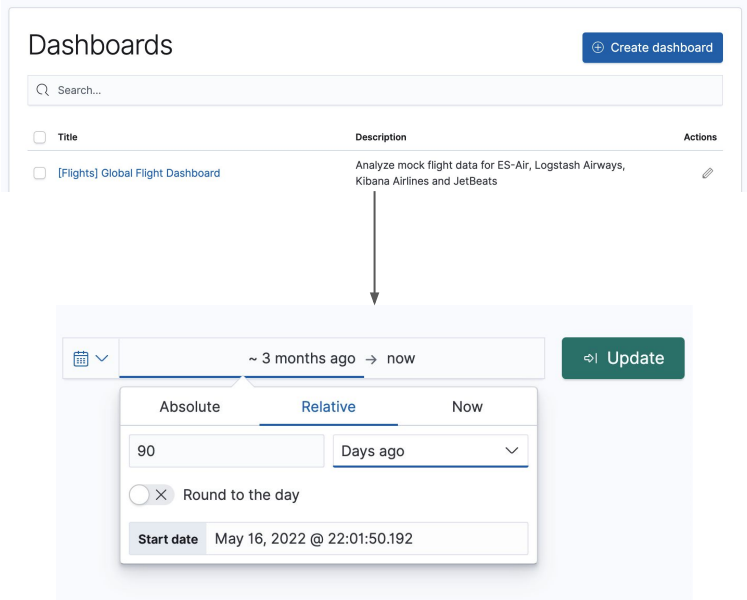
Visualizing our data

Add the data and create the dashboard
Add the sample web logs data, and create and set up the dashboard.

In case it was missed:
1. Go to the Home page, then click Try sample data.
2. **On the Sample web logs card, click Add data.**

Create the dashboard where you'll display the visualization panels.

1. Open the main menu, then click Dashboard.
2. Click Create dashboard.
3. Set the time filter to Last 90 days.



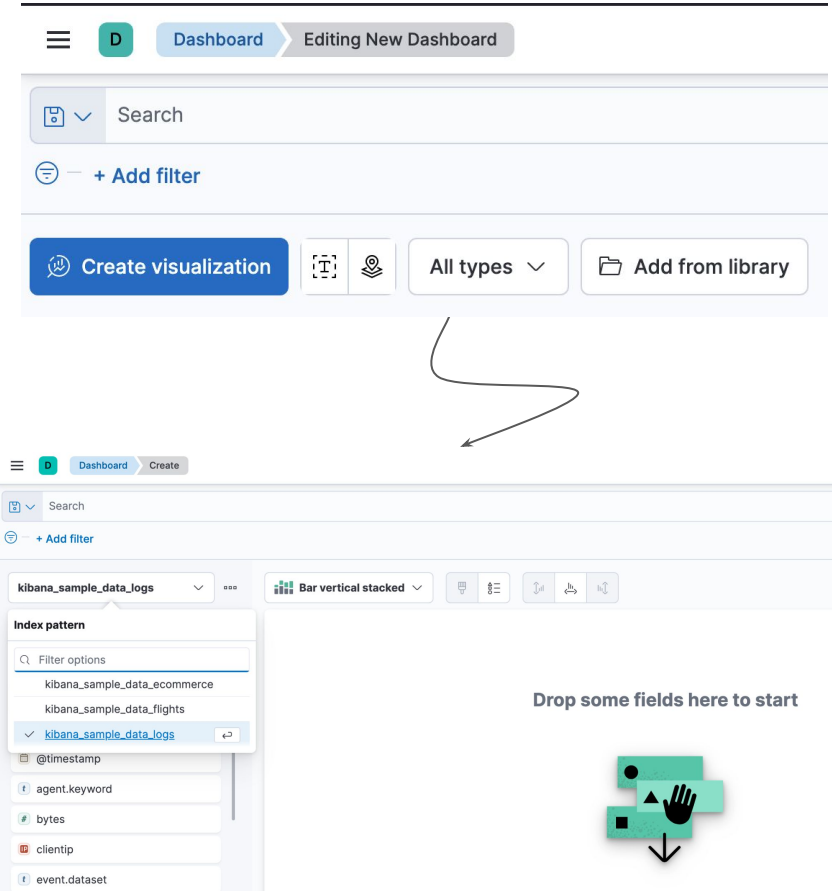
KIBANA INTRODUCTION

Visualizing our data

Open the visualization editor and get familiar with the data

Open the visualization editor, then make sure the correct fields appear.

1. On the dashboard, click Create visualization.
2. Make sure the kibana_sample_data_logs index appears.



To create the visualizations in this tutorial, you'll use the following fields:

- Records
- timestamp
- bytes
- clientip
- Referrer (Defaulted to keyword - 7.17)

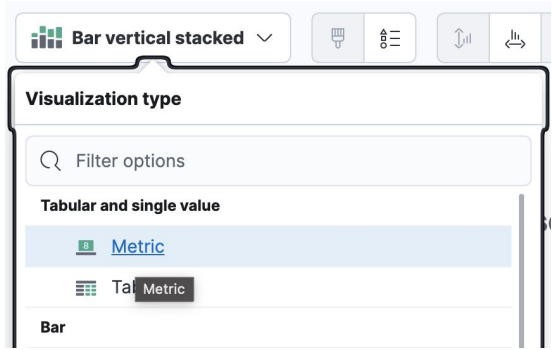
KIBANA INTRODUCTION

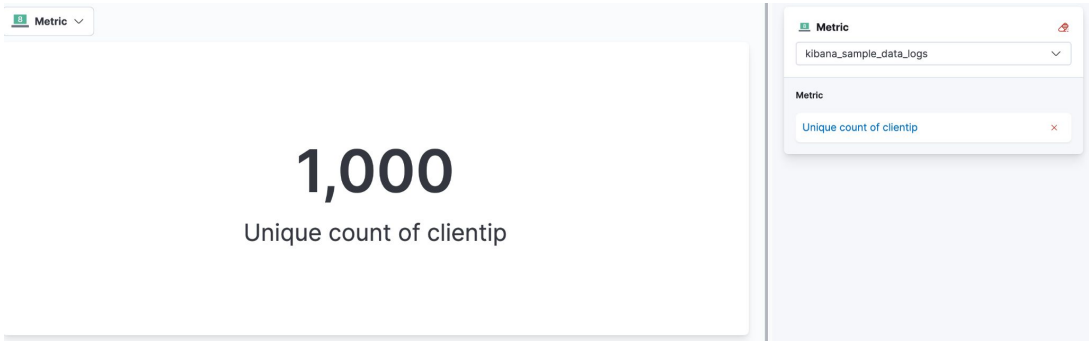
Visualizing our data

Create our first visualization
Pick a field you want to analyze, such as clientip. To analyze only the clientip field, use the Metric visualization to display the field as a number.

The only number function that you can use with clientip is Unique count, also referred to as cardinality, which approximates the number of unique values.

1. Open the Visualization type dropdown, then select Metric.
2. From the Available fields list, drag clientip to the workspace or layer pane.

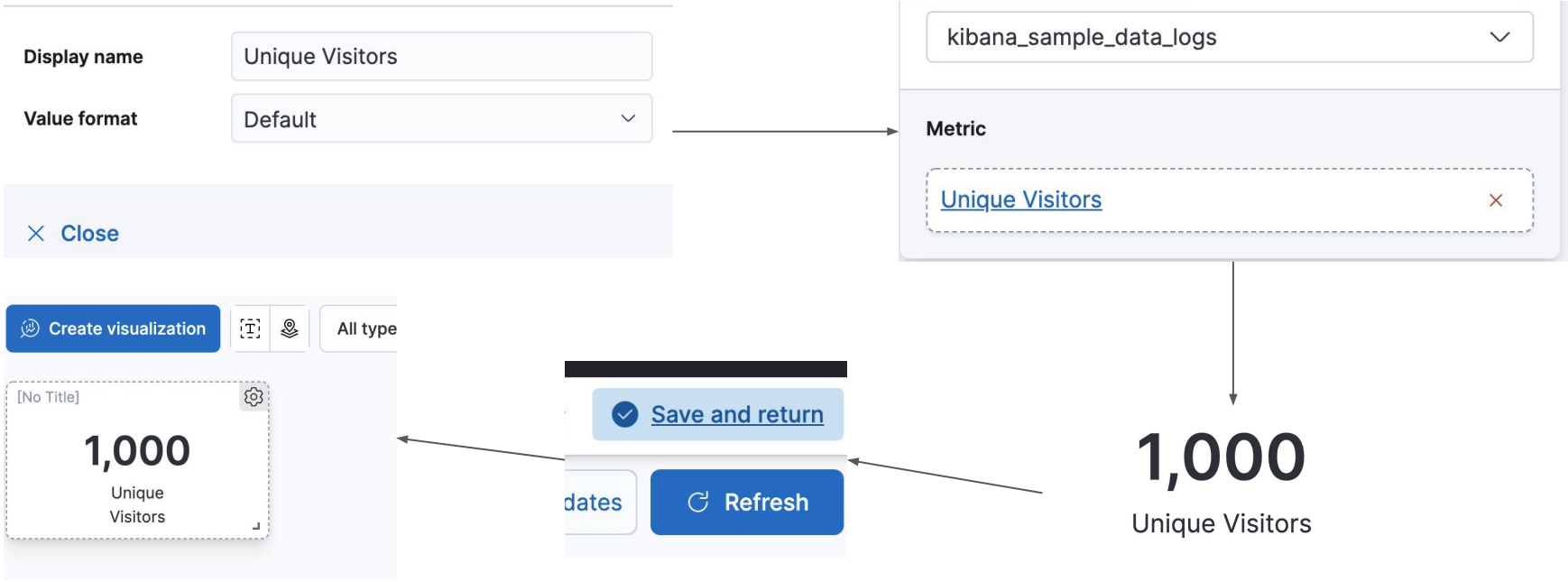




KIBANA INTRODUCTION

Visualizing our data

1. In the layer pane, click Unique count of clientip.
   a. In the Display name field, enter Unique visitors.
   b. Click Close.



KIBANA INTRODUCTION

Visualizing our data

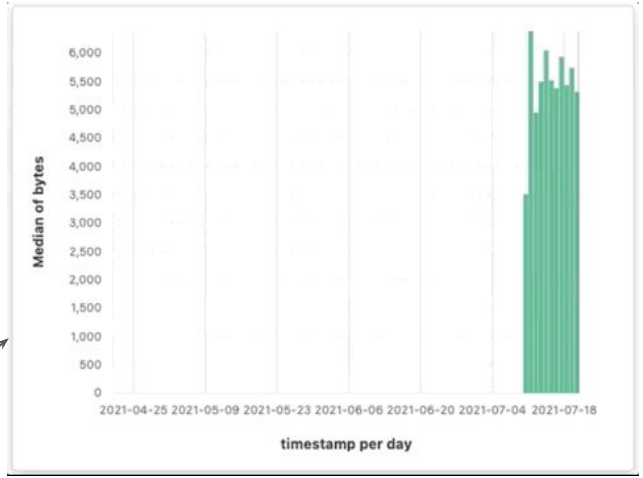
View a metric over time

There are two shortcuts you can use to view metrics over time. When you drag a numeric field to the workspace, the visualization editor adds the default time field from the index pattern. When you use the Date histogram function, you can replace the time field by dragging the field to the workspace.

To visualize the bytes field over time:

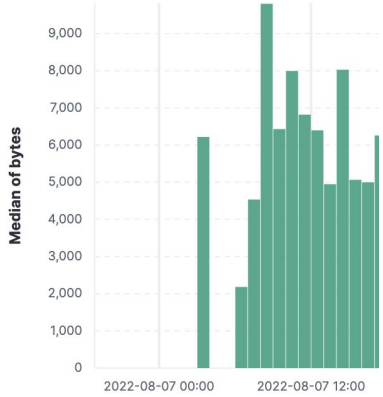
1. On the dashboard, click Create visualization.
2. From the Available fields list, drag **bytes** to the workspace.

   The visualization editor creates a bar chart with the timestamp and Median of bytes fields.
3. To zoom in on the data, click and drag your cursor across the bars.



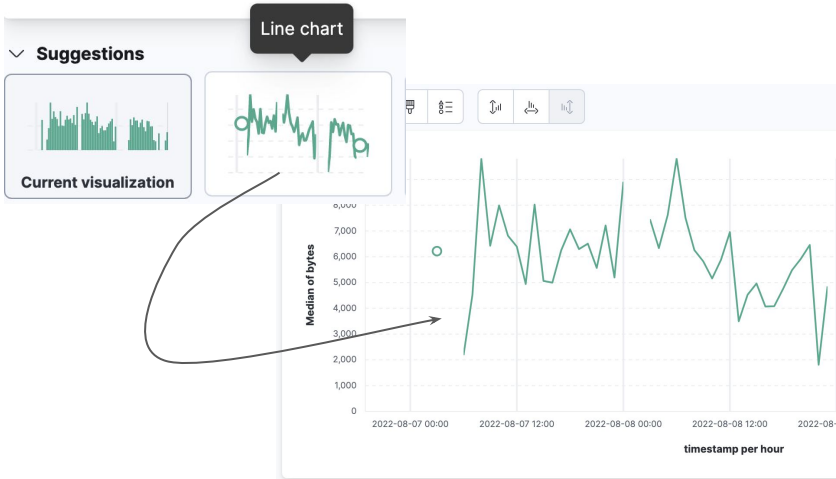
KIBANA INTRODUCTION

Visualizing our data



To emphasize the change in Median of bytes over time, change the visualization type to Line with one of the following options:

- In the Suggestions, click the line chart.
- In the editor toolbar, open the Visualization type dropdown, then select Line.
- In the layer pane, open the Layer visualization type menu, then click Line.



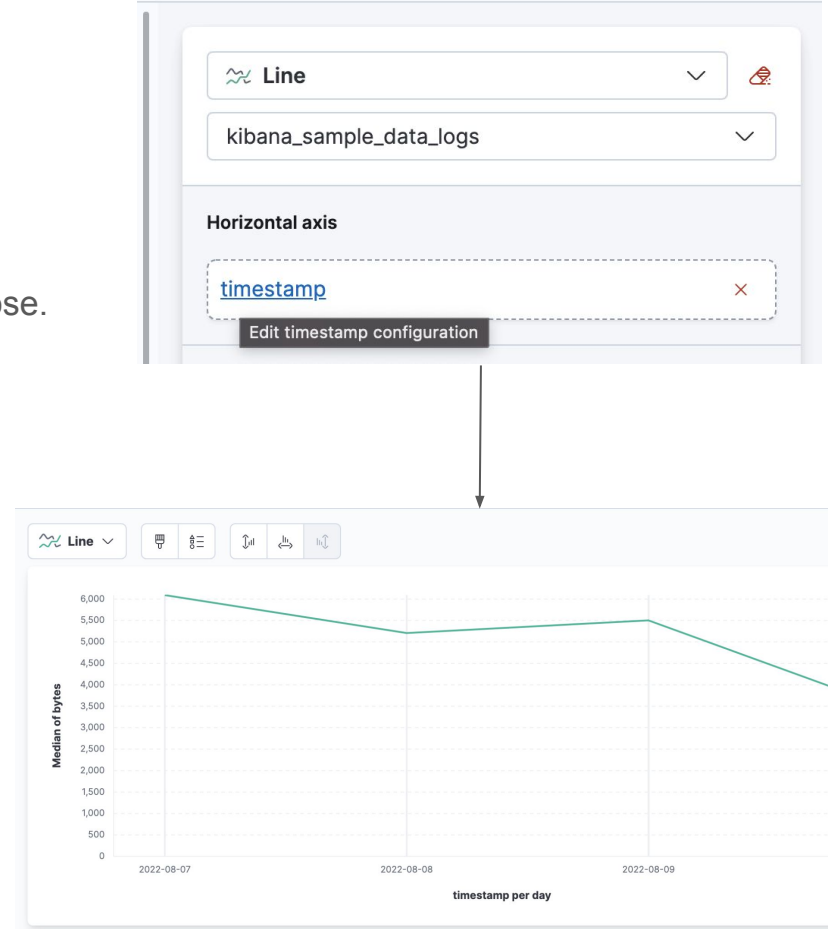
KIBANA INTRODUCTION

Visualizing our data

To increase the minimum time interval:

1. In the layer pane, click timestamp.
2. Select Customize time interval.
3. Change the Minimum interval to 1 days, then click Close.

You can increase and decrease the minimum interval, but you are unable to decrease the interval below the Advanced Settings.

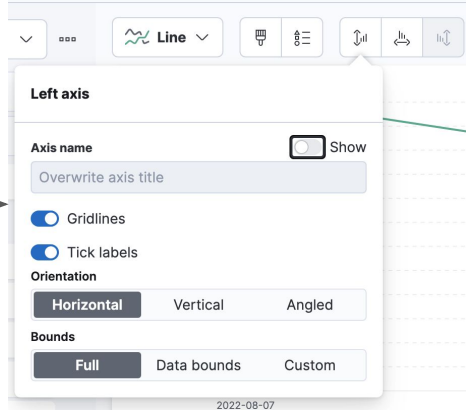


KIBANA INTRODUCTION

Visualizing our data

To save space on the dashboard, hide the axis labels.

1. Open the Left axis menu, then deselect Show.



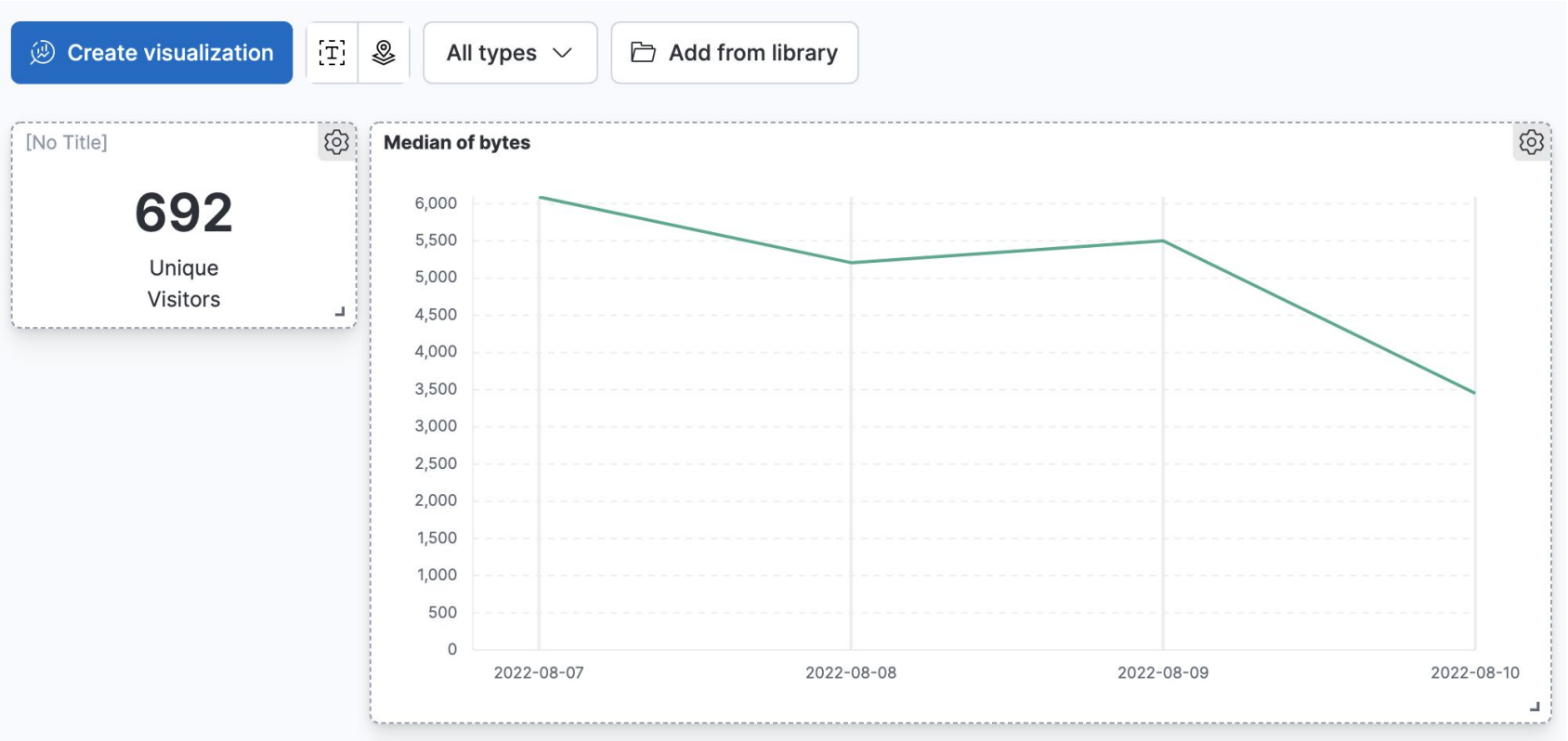
2. Click Sa**ve and return** to dashboard

Since you removed the axis labels, add a panel title:

1. Open the panel menu, then select Edit panel title.
2. In the Panel title field, enter **Median of bytes**, then click Save.

KIBANA INTRODUCTION

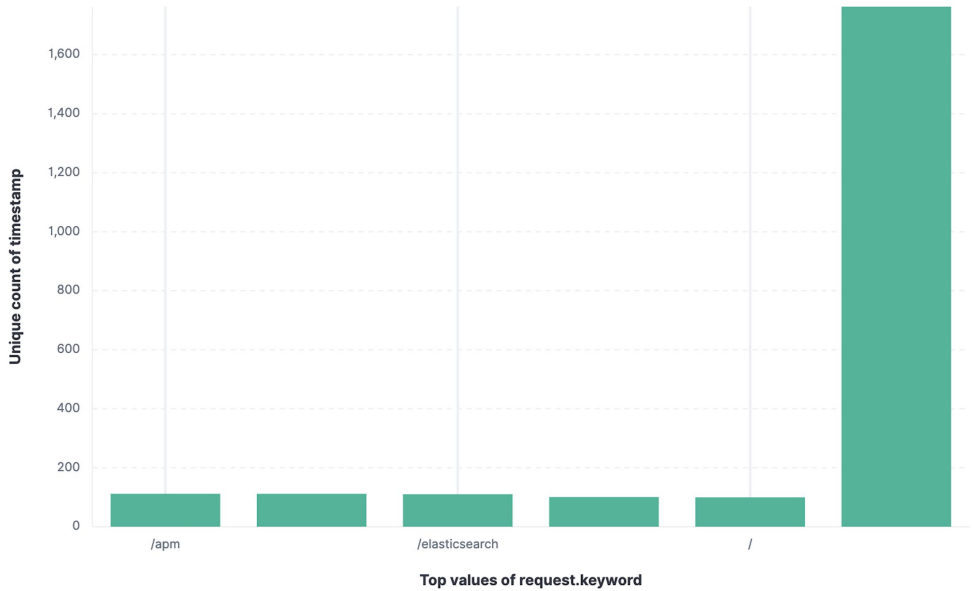
Visualizing our data



KIBANA INTRODUCTION

Visualizing our data

View the top values of a field
Create a visualization that displays the most frequent values of **request.keyword** on your website, ranked by the unique visitors.



KIBANA INTRODUCTION

Visualizing our data

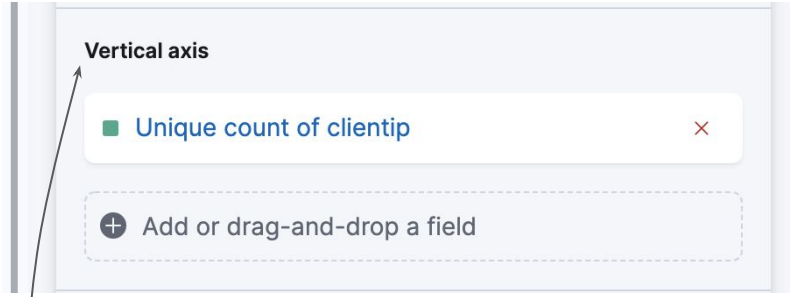
To create the visualization, use T**op values of request.keyword** ranked by **Unique count of clientip**, instead of being ranked by **Count of records**.

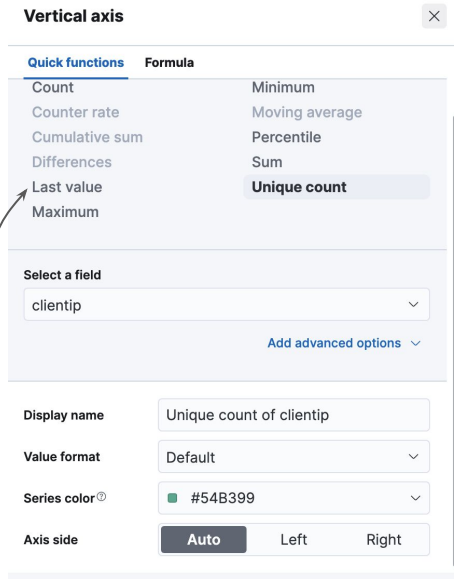
The Top values function ranks the unique values of a field by another function. The values are the most frequent when **ranked by a Count function**, and the largest when ranked by the Sum function.

1. On the dashboard, click Create visualization.
2. From the Available fields list, drag **clientip** to the **Vertical axis** field in **the layer pane.**

   The visualization editor automatically applies the **Unique count function**. If you drag clientip to the workspace, the editor adds the field to the incorrect axis.

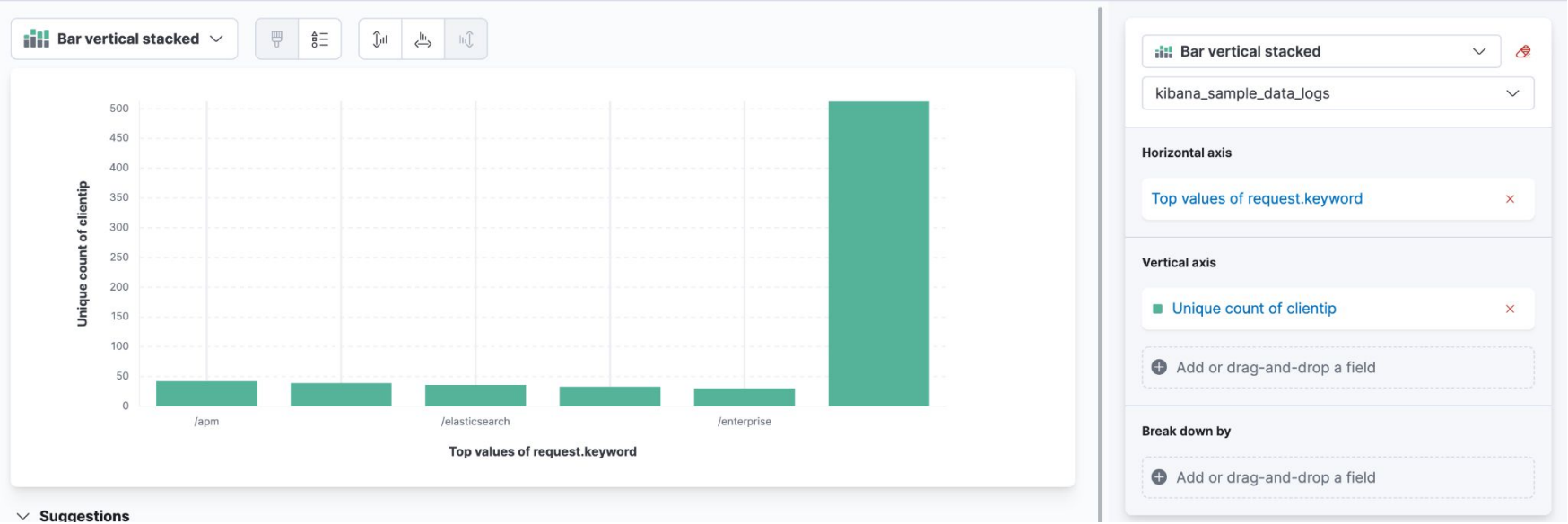
3. Drag request.keyword to the workspace.





KIBANA INTRODUCTION

Visualizing our data

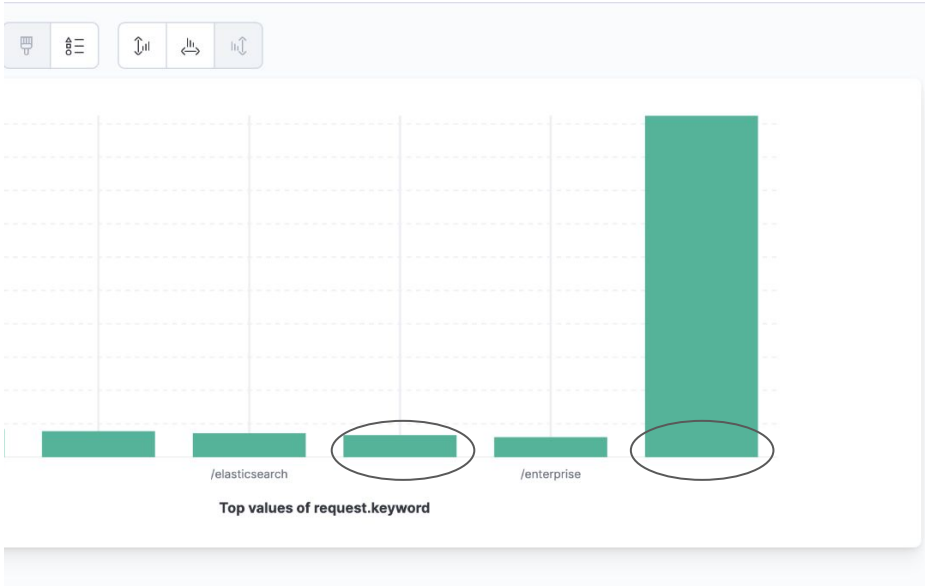


KIBANA INTRODUCTION

Visualizing our data

When you drag a text or IP address field to the workspace, the editor adds the Top values function ranked by **Count of records** to show the most frequent values.

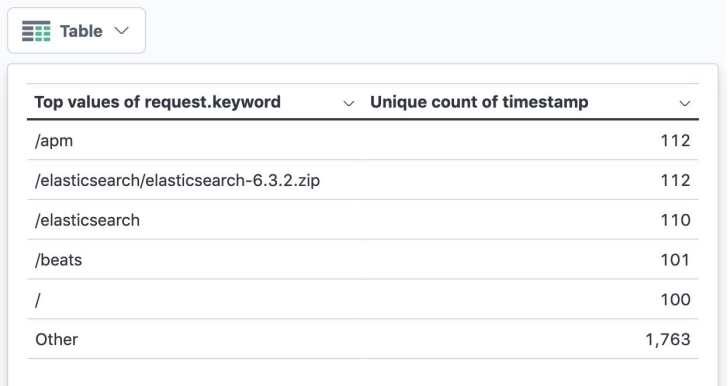
NOTE
The chart labels are unable to display because the **request.keyword** field contains long text fields. You could use one of the **Suggestions**, but the suggestions also have issues with long text. The best way to display long text fields is with the **Table visualization**.



KIBANA INTRODUCTION

Visualizing our data

1. Open the Visualization type dropdown, then select Table.



2. In the layer pane, click Top values of request.keyword.

   a. In the Number of values field, enter 10.
   b. In the Display name field, enter Page URL.
   c. Click Close.

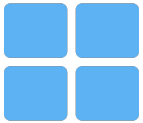
Page URL	Unique count of clientip
/apm	131
/elasticsearch/elasticsearch-6.3.2.zip	126
/elasticsearch	121
/beats/metricbeat	118
/beats/metricbeat/metricbeat-6.3.2-i686.rpm	112
/apm-server/apm-server-6.3.2-windows-x86.zip	110
/beats	109
/	108
/beats/filebeat	108
/enterprise	106
Other	843

3. Click Save and return.

Since the table columns are labeled, you do not need to add a panel title.

- LAB -

CONTINUE ON YOUR OWN



KIBANA INTRODUCTION

Visualizing our data

View the distribution of a number field

The distribution of a number can help you find patterns. For example, you can analyze the website traffic per hour to find the best time for routine maintenance.

1. On the dashboard, click Create visualization.
2. From the Available fields list, drag bytes to Vertical axis field in the layer pane.
3. In the layer pane, click Median of bytes.
    a. Click the Sum function.
    b. In the Display name field, enter Transferred bytes.
    c. From the Value format dropdown, select Bytes (1024), then click Close.
4. From the Available fields list, drag hour_of_day to Horizontal axis field in the layer pane.
5. In the layer pane, click hour_of_day, then slide the Intervals granularity slider until the horizontal axis displays hourly intervals.

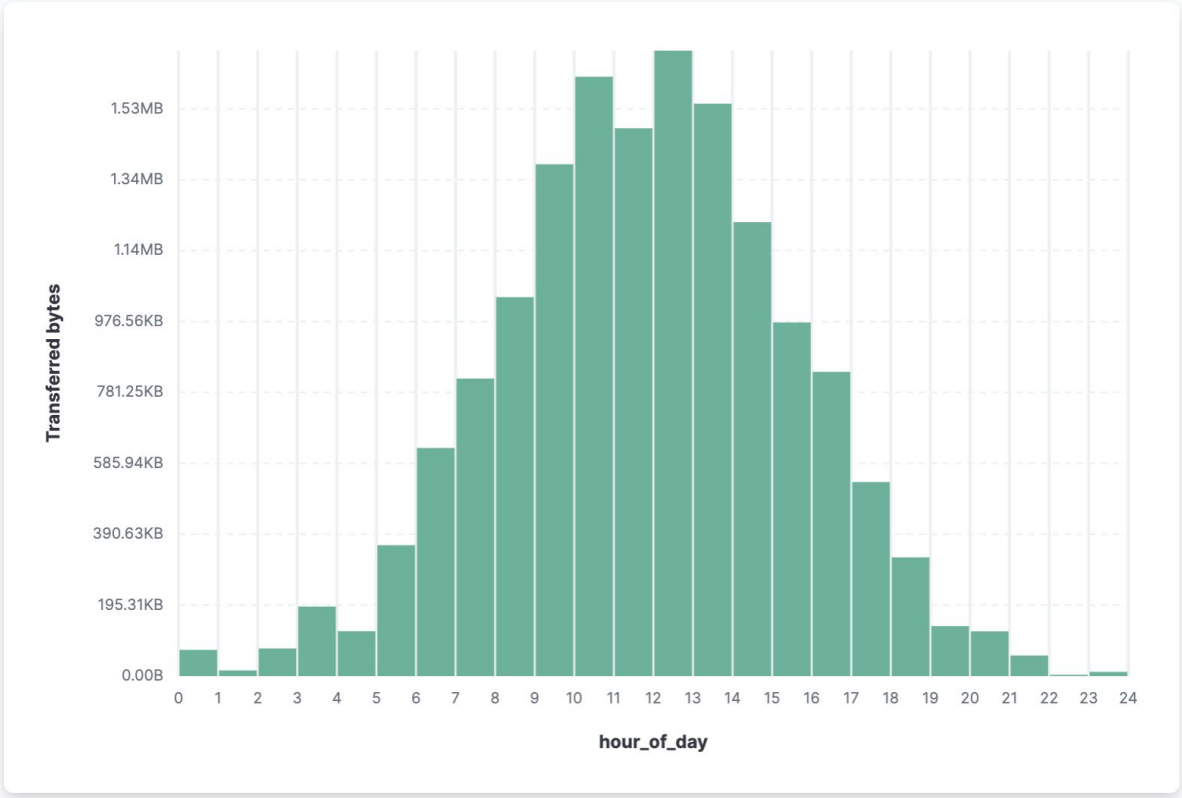
KIBANA INTRODUCTION

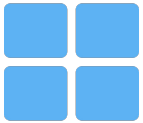
Visualizing our data

Click Save and return.
Add a panel title:

Open the panel menu, then select Edit panel title.
In the Panel title field, enter Website traffic, then click Save.

Final Results





KIBANA INTRODUCTION

Visualizing our data

Create a multi-level chart

Table and Proportion visualizations support multiple functions. For example, to create visualizations that break down the data by website traffic sources and user geography, apply the Filters and Top values functions.

1. On the dashboard, click Create visualization.
2. Open the Visualization type dropdown, then select Treemap.
3. From the Available fields list, drag Records to the Size by field in the layer pane.
4. In the editor, click Add or drag-and-drop a field for Group by.

KIBANA INTRODUCTION

Visualizing our data

Create a filter for each website traffic source:

1. From Select a function, click Filters.
2. Click All records, enter the following in the query bar, then press Return:

- KQL — referer : *facebook.com*
- Label — Facebook

3. Click Add a filter, enter the following in the query bar, then press Return:

- KQL — referer : *twitter.com*
- Label — Twitter

4. Click Add a filter, enter the following in the query bar, then press Return:

- KQL — NOT referer : *twitter.com* OR NOT referer: *facebook.com*
- Label — Other

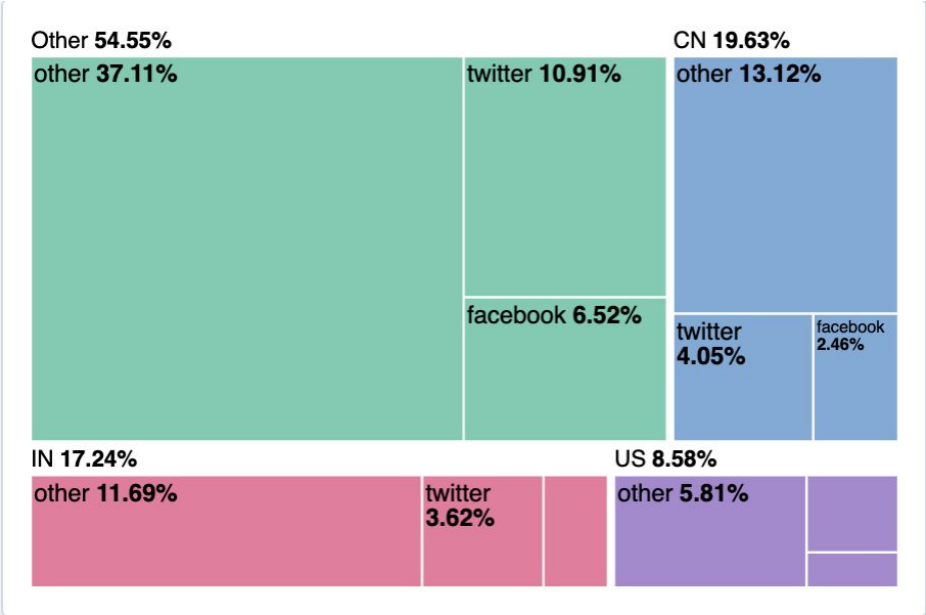
Click Close.

KIBANA INTRODUCTION

Visualizing our data

Add the user geography grouping:

1. From the Available fields list, drag geo.srcdest to the workspace.
2. To change the Group by order, drag Top values of geo.srcdest in the layer pane so that appears first.



KIBANA INTRODUCTION

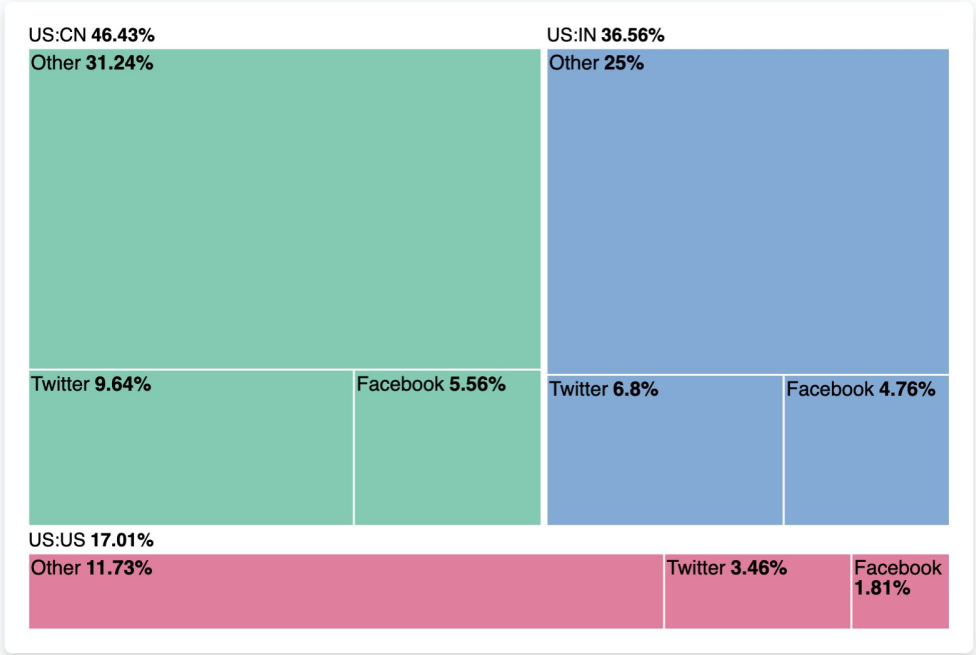
Visualizing our data

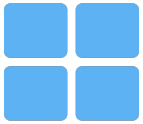
Remove the documents that do not match the filter criteria:

1. In the layer pane, click Top values of geo.srcdest.
2. Click Advanced, then deselect Group other values as "Other", the click Close.
3. Click Save and return.

Add a panel title:

1. Open the panel menu, then select Edit panel title.
2. In the Panel title field, enter Page views by location and referrer, then click Save.



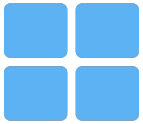


KIBANA INTRODUCTION

Visualizing our data

Add on your own
PIE that needs to have the following:

Bytes by large filter (from X to *)



KIBANA INTRODUCTION

Visualizing our data

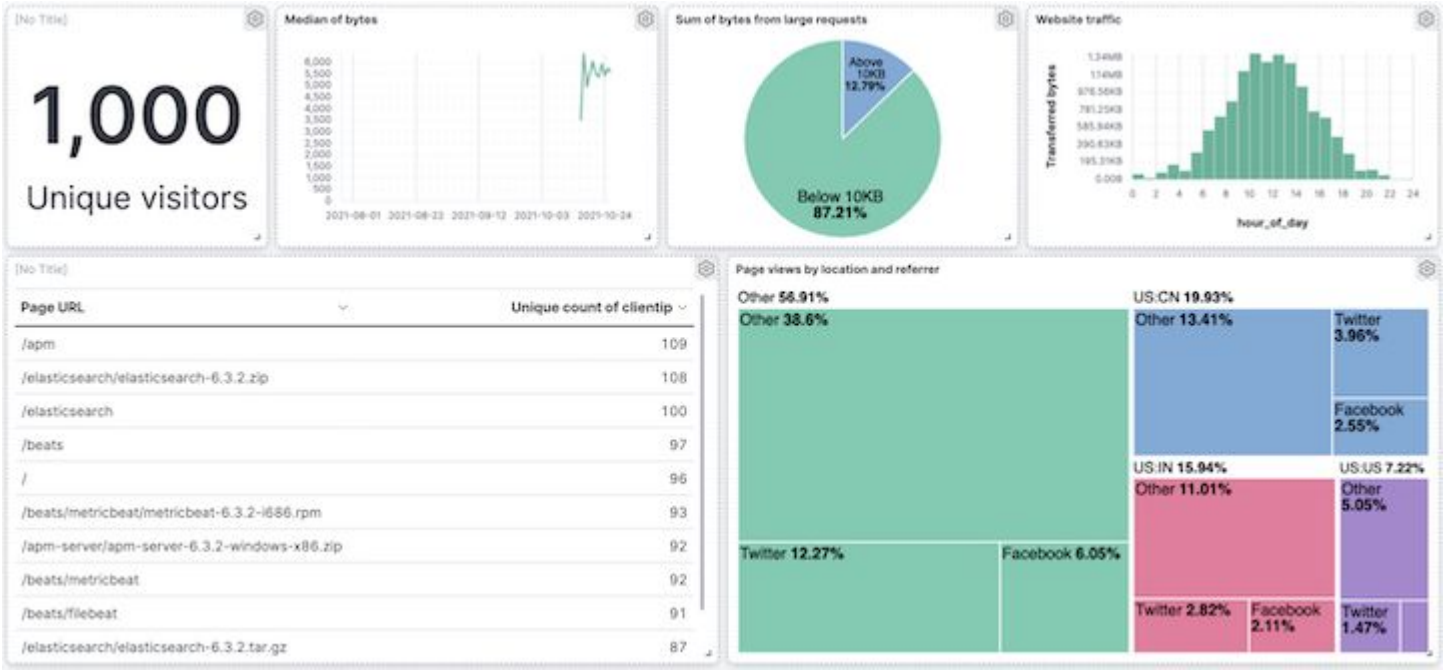
Arrange the dashboard panels
Resize and move the panels so they all appear on the dashboard without scrolling.

Decrease the size of the following panels, then move the panels to the first row:

- Unique visitors
- Median of bytes
- Sum of bytes from large requests
- Website traffic

KIBANA INTRODUCTION

Visualizing our data



KIBANA INTRODUCTION

Visualizing our data

Save the dashboard

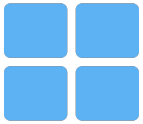
Now that you have a complete overview of your web server data, save the dashboard.

1. In the toolbar, click Save.
2. On the Save dashboard window, enter Logs dashboard in the Title field.
3. Select Store time with dashboard.
4. Click Save.
5. Send me the link to your Dashboard

- LAB -

END

- KIBANA -
WATCHER

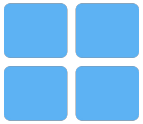


KIBANA INTRODUCTION

WATCHER

INTRO

Watcher is an Elasticsearch feature that we can use to create actions based on conditions, which are periodically evaluated using queries on our data. Watches are helpful for analyzing mission-critical and business-critical streaming data. For example, we might watch application logs for performance outages or audit access logs for security threats.



KIBANA INTRODUCTION

WATCHER

Watch definition

Trigger
Determines when the watch is checked. A watch must have a trigger.

Input
Loads data into the watch payload. If no input is specified, an empty payload is loaded.

Condition
Controls whether the watch actions are executed. If no condition is specified, the condition defaults to always.

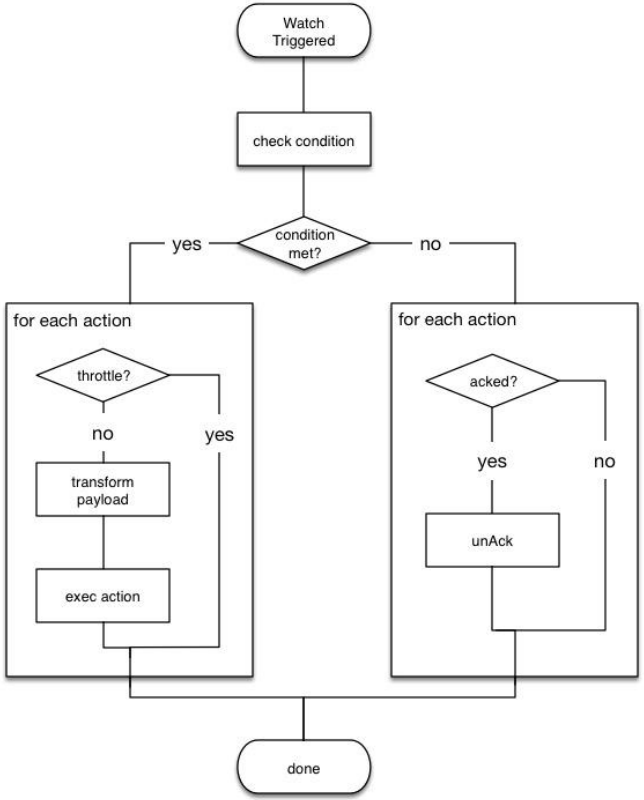
Transform
Processes the watch payload to prepare it for the watch actions. We can define transforms at the watch level or define action-specific transforms. Optional.

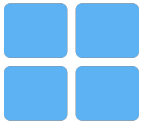
Actions
Specify what happens when the watch condition is met.

KIBANA INTRODUCTION

WATCHER

The following diagram shows the watch execution process:





KIBANA INTRODUCTION

WATCHER

BASICS

```
PUT _watcher/watch/watch_name
{
  "trigger" : {},
  "input" : {},
  "condition" : { },
  "transform" : { },
  "actions" : {}
}
```

This is a simple elastic call that can be executed in Kibana console. This call will create an Elastic Watcher with the name “watch_name”.
We will discuss each element separately:

KIBANA INTRODUCTION

WATCHER

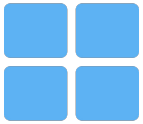
TRIGGER

```
"trigger" : {
  "schedule" : {
    "interval" : "5m"
  }
}
```

Determines when the watch execution process should start. We can either provide a cron expression or specify a period in terms of intervals.
This is an interval based trigger that will execute in a five-minute interval once the Watcher is created.

INPUT

Input is the source from where we fetch the data and load it into the execution context. The result from this input is called a “watcher payload” or “context payload”.
Watcher supports four types of inputs:
>>>>>



KIBANA INTRODUCTION

WATCHER

INPUT CONTINUE

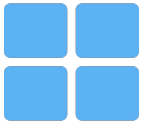
simple: load static data into the execution context
search: load the results of a search into the execution context
http: load the results of an HTTP request into the execution context
chain: use a series of inputs to load data into the execution context

CONDITION

It is used to decide whether we should invoke a Watcher action or not.
Default is true.

The way it's been calculated is by checking the payload of the search and loading it
Into ELASTICSEARCH CTX

```
"condition": {
   "compare": {
     "ctx.payload.hits.total": {
       "gt": 0
     }
   }
 }
```



KIBANA INTRODUCTION

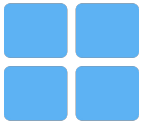
WATCHER

ELASTICSEARCH CTX?

ctx is a special variable that allows we to access the source of the object

Accessing the search results
Conditions, transforms, and actions can access the search results through the watch execution context. For example:

- To load all of the search hits into an email body, use ctx.payload.hits.
- To reference the total number of hits, use ctx.payload.hits.total.
- To access a particular hit, use its zero-based array index. For example, to get the third hit, use ctx.payload.hits.hits.2.
- To get a field value from a particular hit, use ctx.payload.hits.hits.<index>.fields.<fieldname>. For example, to get the message field from the first hit, use ctx.payload.hits.hits.0.fields.message.



KIBANA INTRODUCTION

WATCHER

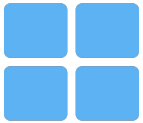
TRANSFORMATION

A “Transform” processes and changes the payload in the watch execution context to prepare it for the watch actions.

ACTION

It is executed when conditions are met. Eg. Send an email, send a slack message, call a rest service, add loggers, etc.

```
"actions": {
    "log": {
      "logging": {
        "text": "We got the expected
error"
      }
    }
  }
```



KIBANA INTRODUCTION

WATCHER

WATCHER APIs

Create Watcher

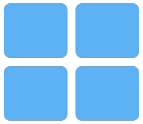
```
CREATE:
PUT _watcher/watch/watch_name
{
  \\Body

}

GET _watcher/watch/watch_name

ACTIVATE/DEACTIVATE:  PUT _watcher/watch/watch_name/_activate/ OR _deactivate

EXECUTE WATCHER:  PUT _watcher/watch/watch_name/_execute
```



KIBANA INTRODUCTION

WATCHER

WATCHER APIs

WATCHER EXAMPLE >>> COPY IT TO KIBANA DEV FOR REVIEW

Highlights about the Watcher definition on the right:

- This Watcher will execute once every 10 minutes
- It will look for a particular log with the specified status in the input i.e. the search query
- In the event that the condition is fulfilled, it will execute the action
- In the action, we simply add a logger in the elastic log, which we can check in the elastic console

```
PUT /_watcher/watch/log_error_watch
{
  "trigger": {
    "schedule": {
      "interval": "10m"
    }
  },
  "input": {
    "search": {
      "request": {
        "indices": ["logs"],
        "body": {
          "query": {
            "match": {
              "message": "error"
            }
          }
        }
      }
    }
  },
  "condition": {
    "compare": {
      "ctx.payload.hits.total": {
        "gt": 0
      }
    }
  },
  "actions": {
    "log": {
      "logging": {
        "text": "We got the expected error"
      }
    }
  }
}
```

KIBANA INTRODUCTION

WATCHER

TESTING

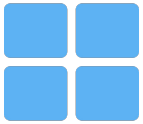
1. Let’s add a new INDEX named LOG with the following mapping

```
"request":     { "type": "keyword" },
"status_code": { "type": "keyword" },
"message":     { "type": "keyword" },
"timestamp":   { "type": "date"    }
```

2. Add new Document into the Index

```
"timestamp" : "2015-05-17T18:12:07.613Z",
"request" : "GET index.html",
"status_code" : 404,
"message" : "Error: File not found"
```

```
PUT /_watcher/watch/log_error_watch
{
  "trigger": {
    "schedule": {
      "interval": "10m"
    }
  },
  "input": {
    "search": {
      "request": {
        "indices": ["logs"],
        "body": {
          "query": {
            "match": {
              "message": "error"
            }
          }
        }
      }
    }
  },
  "condition": {
    "compare": {
      "ctx.payload.hits.total": {
        "gt": 0
      }
    }
  },
  "actions": {
    "log": {
      "logging": {
        "text": "We got the expected error"
      }
    }
  }
}
```



KIBANA INTRODUCTION

WATCHER

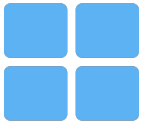
TESTING HANDS ON

3. Execute the Watcher
4. View the log in elasticsearch logs (docker logs -f elkstack-docker_elasticsearch_1)

Advanced WATCHER ACTIONS:
https://www.elastic.co/guide/en/elasticsearch/reference/master/actions.html

```
PUT /_watcher/watch/log_error_watch
{
  "trigger": {
    "schedule": {
      "interval": "10m"
    }
  },
  "input": {
    "search": {
      "request": {
        "indices": ["logs"],
        "body": {
          "query": {
            "match": {
              "message": "error"
            }
          }
        }
      }
    }
  },
  "condition": {
    "compare": {
      "ctx.payload.hits.total": {
        "gt": 0
      }
    }
  },
  "actions": {
    "log": {
      "logging": {
        "text": "We got the expected error"
      }
    }
  }
}
```

- KIBANA - MANAGEMENT

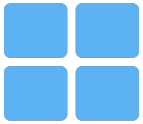


KIBANA INTRODUCTION

MANAGING KIBANA

SPACES

Kibana is considered the “window” to Elasticsearch and indeed it’s a powerful UI for searching, filtering, analyzing, and visualizing Elasticsearch data, but Kibana settings are also used to configure, administer and monitor the Elasticsearch cluster. In this lesson, we’re going to explore how Kibana settings can be tweaked for collaborative teamwork.

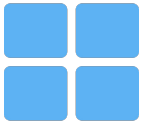


KIBANA INTRODUCTION

MANAGING KIBANA

In Kibana spaces we will learn about

- Spaces enable we to organize various Dashboards, Visualizations, Searches, and other so-called Saved Objects on our Kibana instance, into distinct “spaces” where different user groups can access what they need.
- This can be especially useful for segregating various business users according to departments like Marketing, Security, Development, Operations, Finance etc.
- We can also assign access to multiple spaces for specific users, in which case they’ll see their accessible spaces when logging in.



KIBANA INTRODUCTION

MANAGING KIBANA

In Kibana spaces we will learn about

Export Kibana Dashboards

- In relation to Spaces, Kibana allows we to transfer a selected set of Saved Objects either by copying them between Spaces within a single Kibana instance or by exporting them in JSON format from one Kibana instance and importing into another instance. This can be useful in multi-cluster environments.

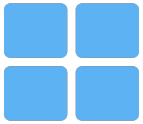
Advanced Kibana Settings

- Lastly, we'll take a look at some settings related to user setup and spaces such as setting a default landing page when entering a space or switching to Dark mode.

KIBANA SPACES

Let’s start by navigating to the Management app in Kibana → and from there, head over to the Spaces configuration in the Kibana section. It can be found here running on our remote host and default port http://REMOTEHOST:5601/app/kibana#/management/spaces/list?_g=()

As we can see right away, there’s already one space created for us by default (named Default). It’s always present if we haven’t explicitly disabled spaces. All objects that are contained inside the default space are shared between all users.

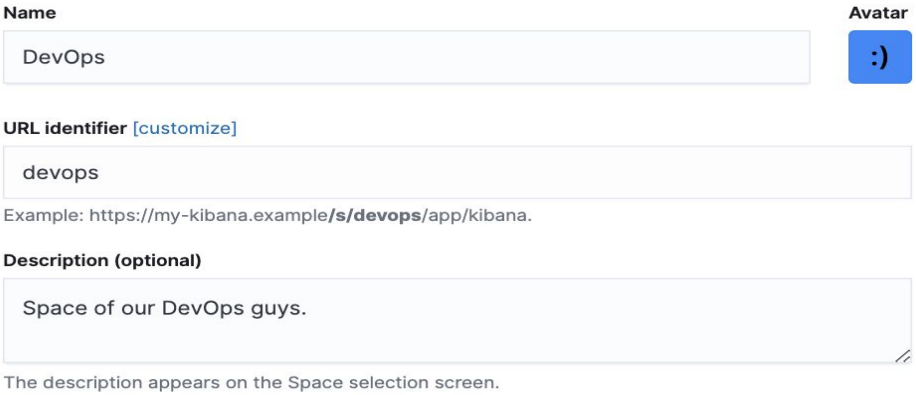


KIBANA INTRODUCTION

MANAGING KIBANA

KIBANA SPACES

Let’s create a new space by clicking on Create a space. We’ll add the name “DevOps” and we can optionally provide a description. Lastly, we have the option to choose a color for space. This is helpful to visually differentiate spaces for users who have access to multiple spaces.



KIBANA SPACES

Then, scroll down to the Filters section where we can also filter out any sections that are not relevant for the specific target group. Our DevOps example would likely need access to most or all of these, but if we would be preparing a space for marketing, for example, we'd probably want to hide most of these and just enable access to the Dashboard menu option.

Bear in mind that this only hides or displays the menu options in the Kibana UI, but users would still be able to reach those areas via a direct URL. In other words, these Filters are not a replacement for access control features which would need to be handled separately.

KIBANA INTRODUCTION

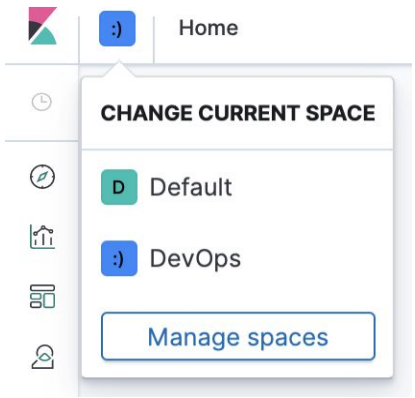
MANAGING KIBANA

KIBANA SPACES

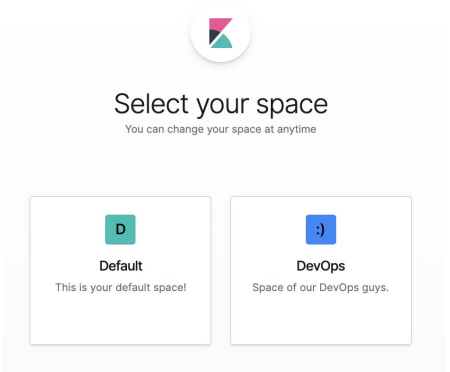
Feature	Show? (change all)
Discover	on
Visualize	on
Dashboard	on
Dev Tools	on
Advanced Settings	on
Index Pattern Management	on
Saved Objects Management	on
Graph	off
Stack Monitoring	on
Machine Learning	off
APM	on
Maps	off
Canvas	off
Metrics	on
Logs	on
SIEM	off
Uptime	on

Once finished click create

Now we can see the new space in the top left menu by clicking on the Spaces icon next to the Kibana logo. This menu is great for quickly jumping between different spaces.



NEW LOGIN SCREEN

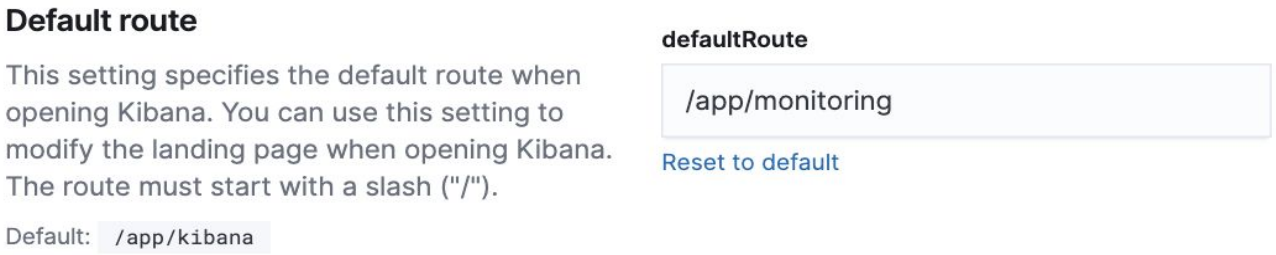


Advanced Kibana settings

For our last example, let's head over to the Advanced Settings section in the Management app where many settings are concentrated.

First of all, bear in mind the warning box that is displayed on top of the page saying "Caution: We can break stuff here". This is true. All sorts of settings, both "high" and "low" level, are found here in one place, so always double-check what we're about to tweak. Secondly, these settings are mostly applied to the active space (i.e. not globally), which is useful for our case.

Advanced Kibana settings



We can change the default landing page to which the users are redirected when they switch to a given space. Do so with the Default route setting. Here we have switched the default landing page to the Monitoring app which aggregates information about our Elastic Stack components. For example, our Operations team can go straight to the overview of the cluster, similarly, we can guide marketing users to their main dashboard.

Advanced Kibana settings

More advanced options can be seen in the following link
https://www.elastic.co/guide/en/kibana/current/advanced-options.html