splunk>

tutorialspoint

SIMPLY EASY LEARNING

www.tutorialspoint.com


https://www.facebook.com/tutorialspointindia


https://twitter.com/tutorialspoint

About the Tutorial

Splunk is a software used to search and analyze machine data. This machine data can come from web applications, sensors, devices or any data created by user. It serves the needs of IT infrastructure by analyzing the logs generated in various processes but it can also analyze any structured or semi-structured data with proper data modelling. It has built-in features to recognize the data types, field separators and optimize the search processes. It also provides data visualization on the search results.

Audience

This tutorial targets IT professionals, students, and IT infrastructure management professionals who want a solid grasp of essential Splunk concepts. After completing this tutorial, you will achieve intermediate expertise in Splunk, and easily build on your knowledge to solve more challenging problems.

Prerequisites

The reader should be familiar with querying language like SQL. General knowledge in typical operations in using computer applications like storing and retrieving data and reading the logs generated by computer programs will be an highly useful.

Copyright & Disclaimer

Table of Contents

1. Splunk – Overview

Splunk is a software which processes and brings out insight from machine data and other forms of big data. This machine data is generated by CPU running a webserver, IOT devices, logs from mobile apps, etc. It is not necessary to provide this data to the end users and does not have any business meaning. However, they are extremely important to understand, monitor and optimize the performance of the machines.

Splunk can read this unstructured, semi-structured or rarely structured data. After reading the data, it allows to search, tag, create reports and dashboards on these data. With the advent of big data, Splunk is now able to ingest big data from various sources, which may or may not be machine data and run analytics on big data.

So, from a simple tool for log analysis, Splunk has come a long way to become a general analytical tool for unstructured machine data and various forms of big data.

Product Categories

Splunk is available in three different product categories as follows:

- **Splunk Enterprise:** It is used by companies which have large IT infrastructure and IT driven business. It helps in gathering and analysing the data from websites, applications, devices and sensors, etc.
- **Splunk Cloud:** It is the cloud hosted platform with same features as the enterprise version. It can be availed from Splunk itself or through the AWS cloud platform.
- **Splunk Light:** It allows search, report and alert on all the log data in real time from one place. It has limited functionalities and features as compared to the other two versions.

Splunk Features

In this section, we shall discuss the important features of enterprise edition:

Data Ingestion

Splunk can ingest a variety of data formats: JSON, XML and unstructured machine data such as web and application logs. The unstructured data can be modeled into a data structure by the user as and when needed.

Data Indexing

The ingested data is indexed by Splunk for faster searching and querying on different conditions.

Data Searching

Searching in Splunk involves using the indexed data for the purpose of creating metrics, predicting future trends and identifying patterns in the data.

Using Alerts

Splunk alerts can be used to trigger emails or RSS feeds when some specific criteria are found in the data being analyzed.

Dashboards

Splunk Dashboards can show the search results in the form of charts, reports and pivots, etc.

Data Model

The indexed data can be modelled into one or more data sets that is based on specialized domain knowledge. This leads to easier navigation by the end users who analyze the business cases without learning the technicalities of the search processing language used by Splunk.

2. Splunk – Environment

In this tutorial, we will aim to install the enterprise version. This version is available for a free evaluation for 60 days with all features enabled. You can download the setup using the below link which is available for both windows and Linux platforms.

https://www.splunk.com/en_us/download/splunk-enterprise.html.

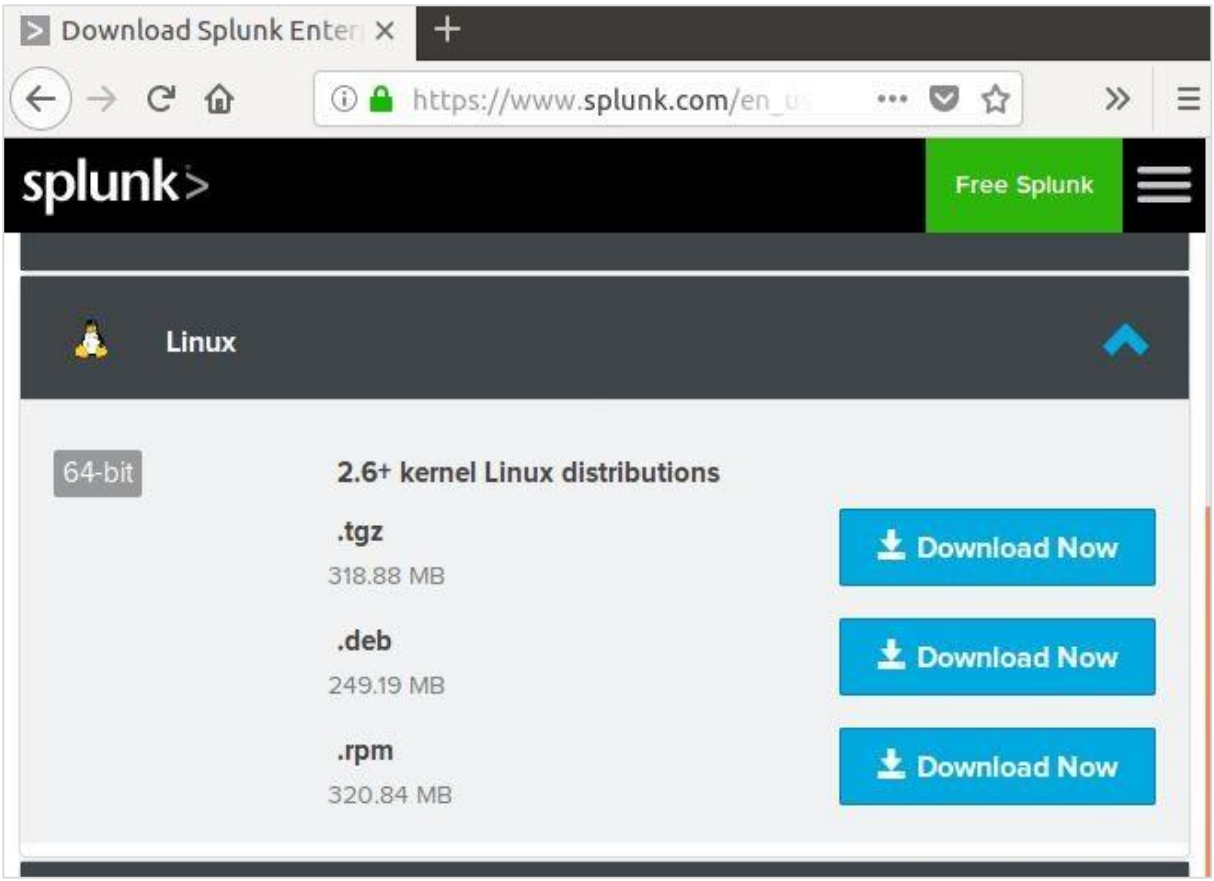
Linux Version

The Linux version is downloaded from the download link given above. We choose the .deb package type as the installation will be done in a Ubuntu platform.

We shall learn this with a step by step approach:

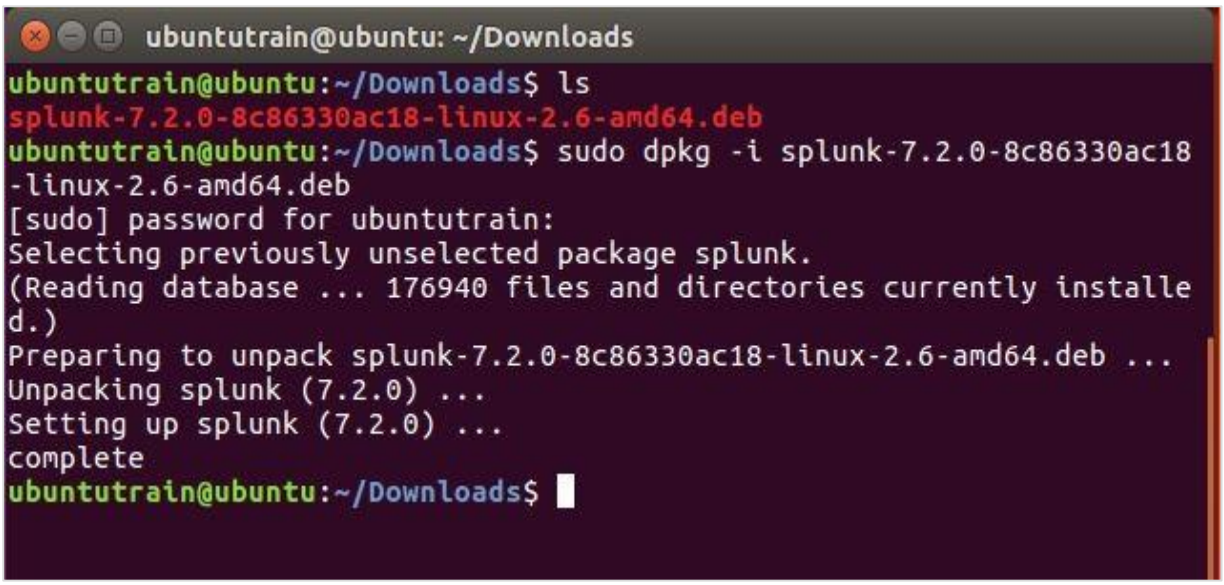
Step 1

Download the .deb package as shown in the screenshot below:



Step 2

Go to the download directory and install Splunk using the above downloaded package.



Step 3

Next, you can start Splunk by using the following command with accept license argument. It will ask for administrator user name and password which you should provide and remember.

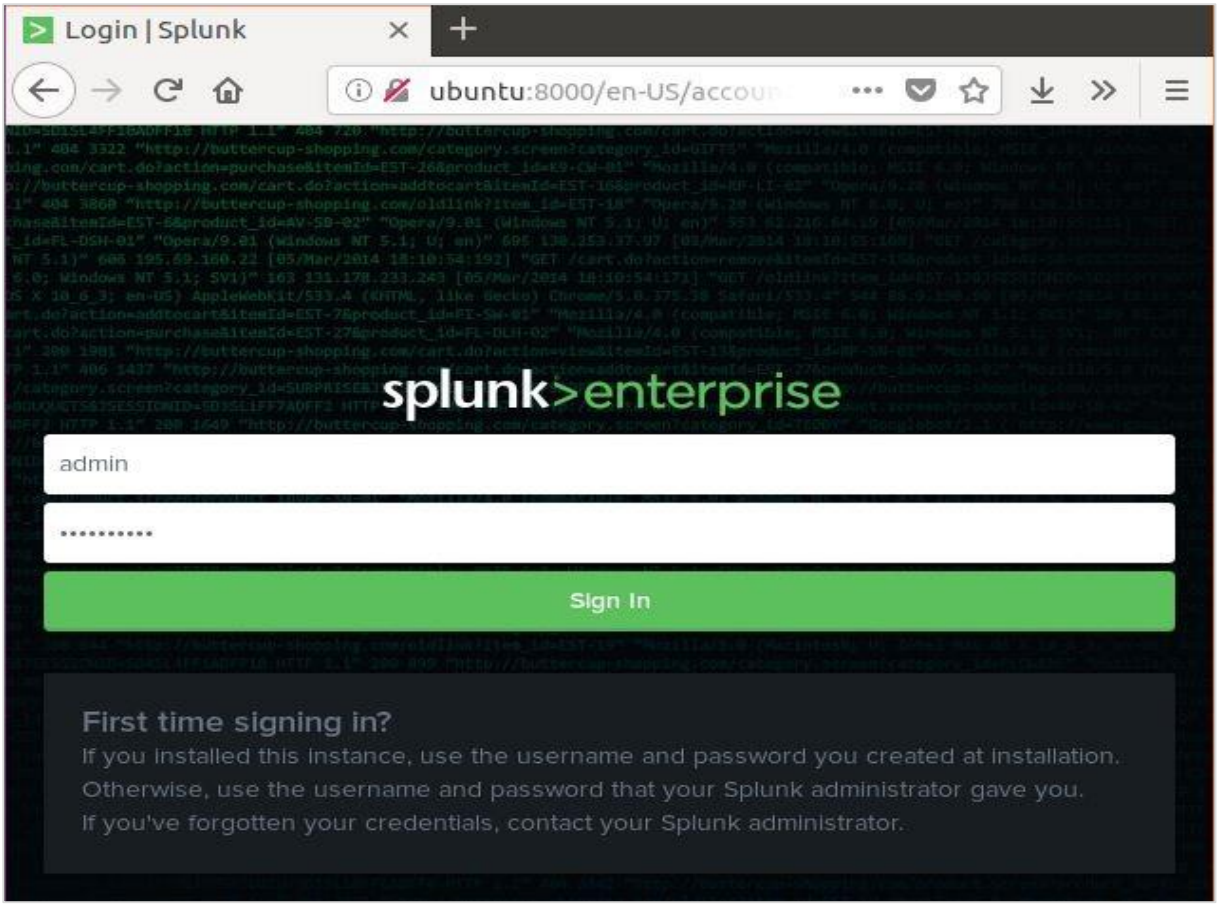
Step 4

The Splunk server starts and mentions the URL where the Splunk interface can be accessed.

```
Starting splunk server daemon (splunkd)...
Generating a 2048 bit RSA private key
...Terminal.+++
.........................+++
writing new private key to 'privKeySecure.pem'
-----
Signature ok
subject=/CN=ubuntu/O=SplunkUser
Getting CA Private Key
writing RSA key
Done


Waiting for web server at http://127.0.0.1:8000 to be available....
........ Done


If you get stuck, we're here to help.
Look for answers here: http://docs.splunk.com

The Splunk web interface is at http://ubuntu:8000
```

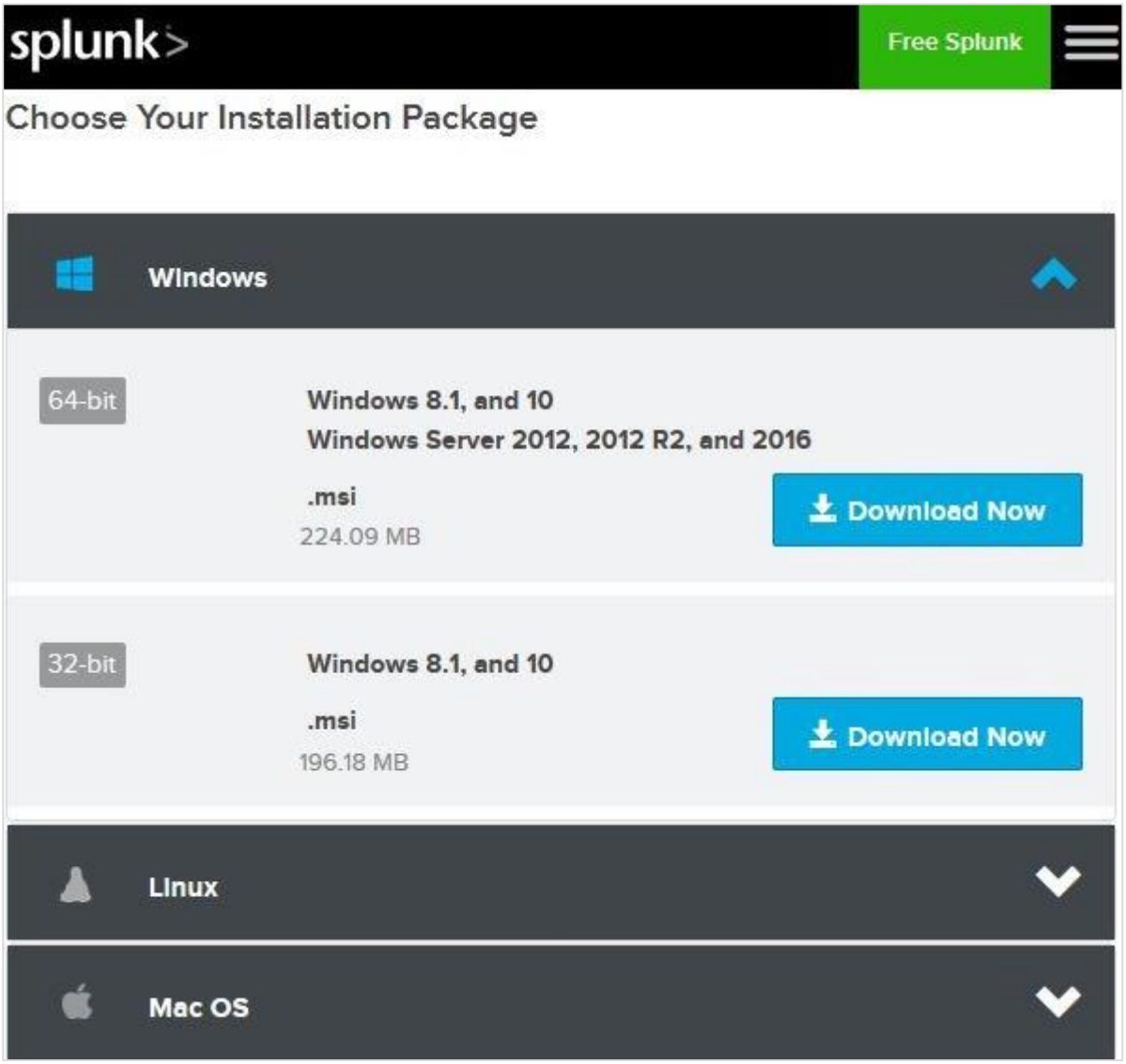
Step 5

Now, you can access the Splunk URL and enter the admin user ID and password created in step 3.



Windows Version

The windows version is available as a msi installer as shown in the below image:



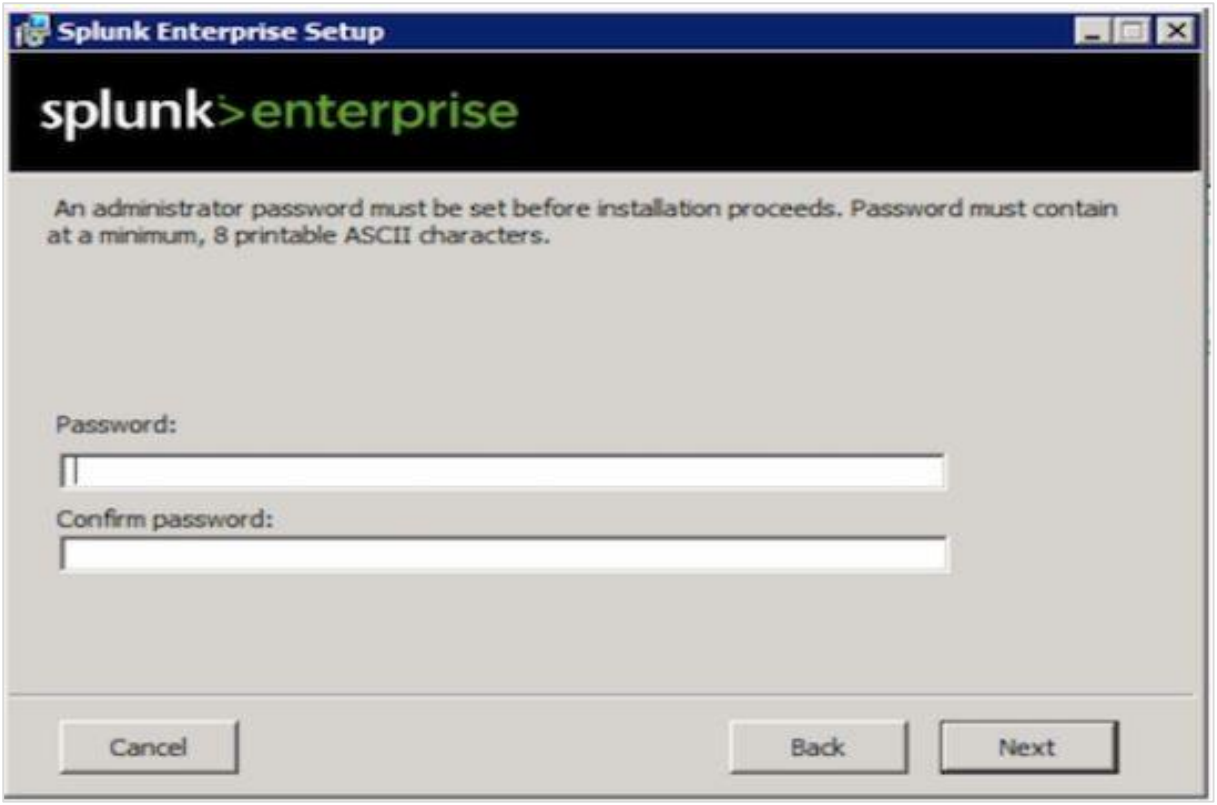
Double clicking on the msi installer installs the Windows version in a straight forward process. The two important steps where we must make the right choice for successful installation are as follows.

Step 1

As we are installing it on a local system, choose the local system option as given below:

Step 2

Enter the password for the administrator and remember it, as it will be used in the future configurations.

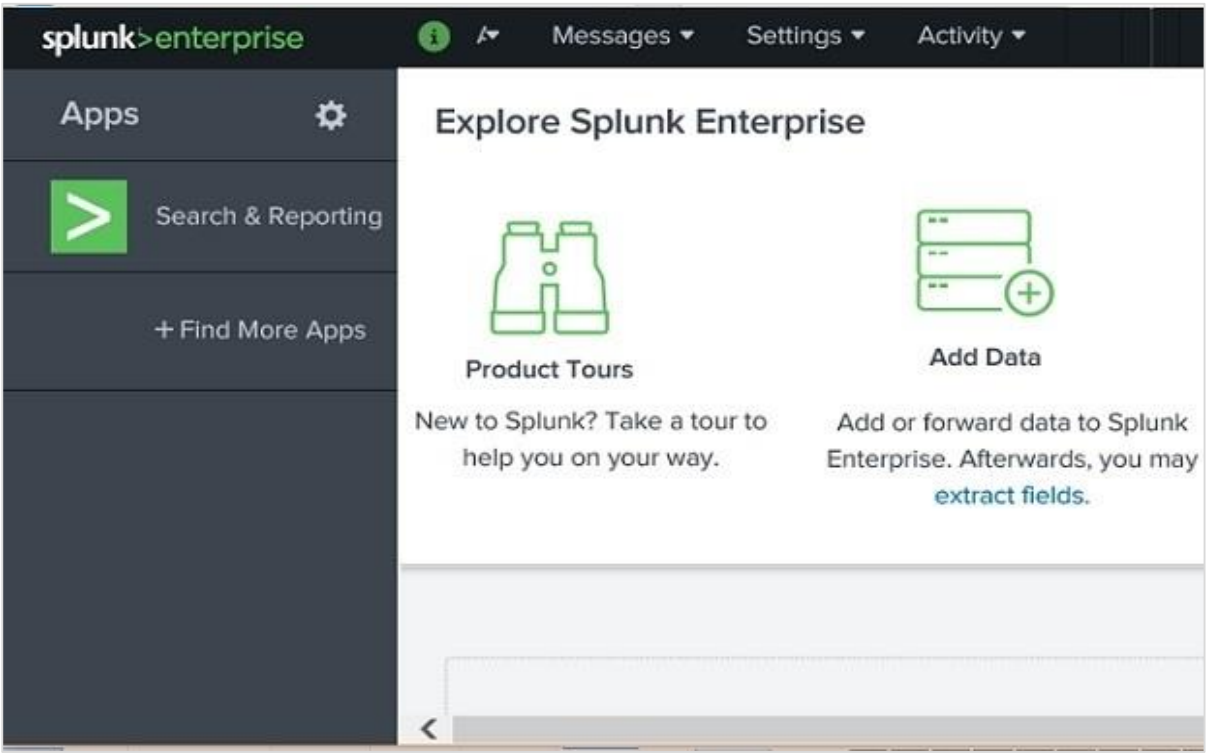


Step 3

In the final step, we see that Splunk is successfully installed and it can be launched from the web browser.

Step 4

Next, open the browser and enter the given url, http://localhost:8000, and login to the Splunk using the admin user ID and password.



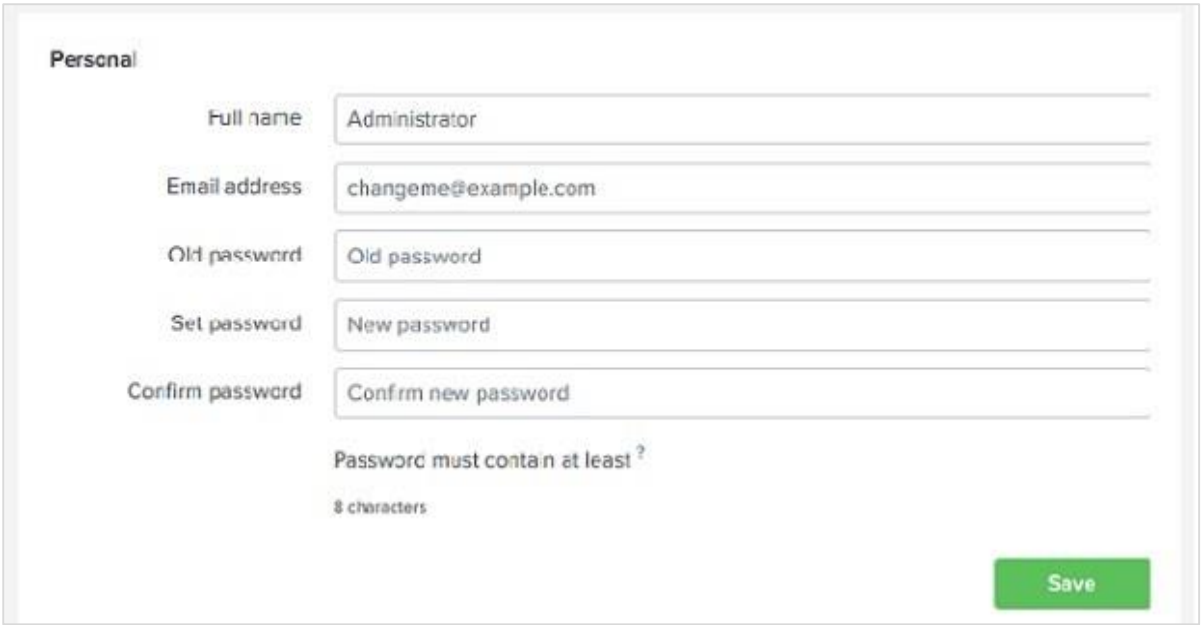
3. Splunk – Interface

The Splunk web interface consists of all the tools you need to search, report and analyse the data that is ingested. The same web interface provides features for administering the users and their roles. It also provides links for data ingestion and the in-built apps available in Splunk.

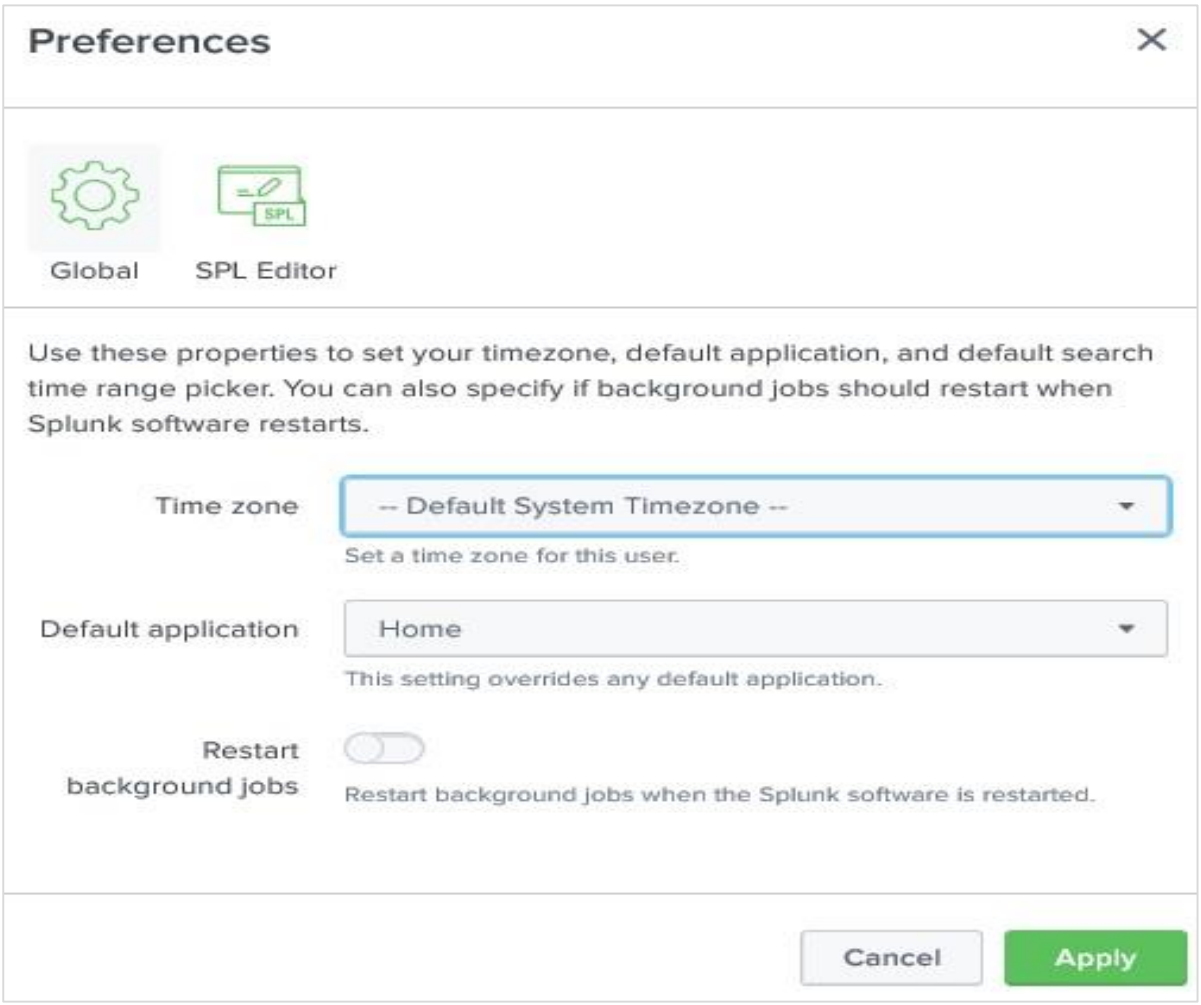
The below picture shows the initial screen after your login to Splunk with the admin credentials.

Administrator Link

The Administrator drop down gives the option to set and edit the details of the administrator. We can reset the admin email ID and password using the below screen:



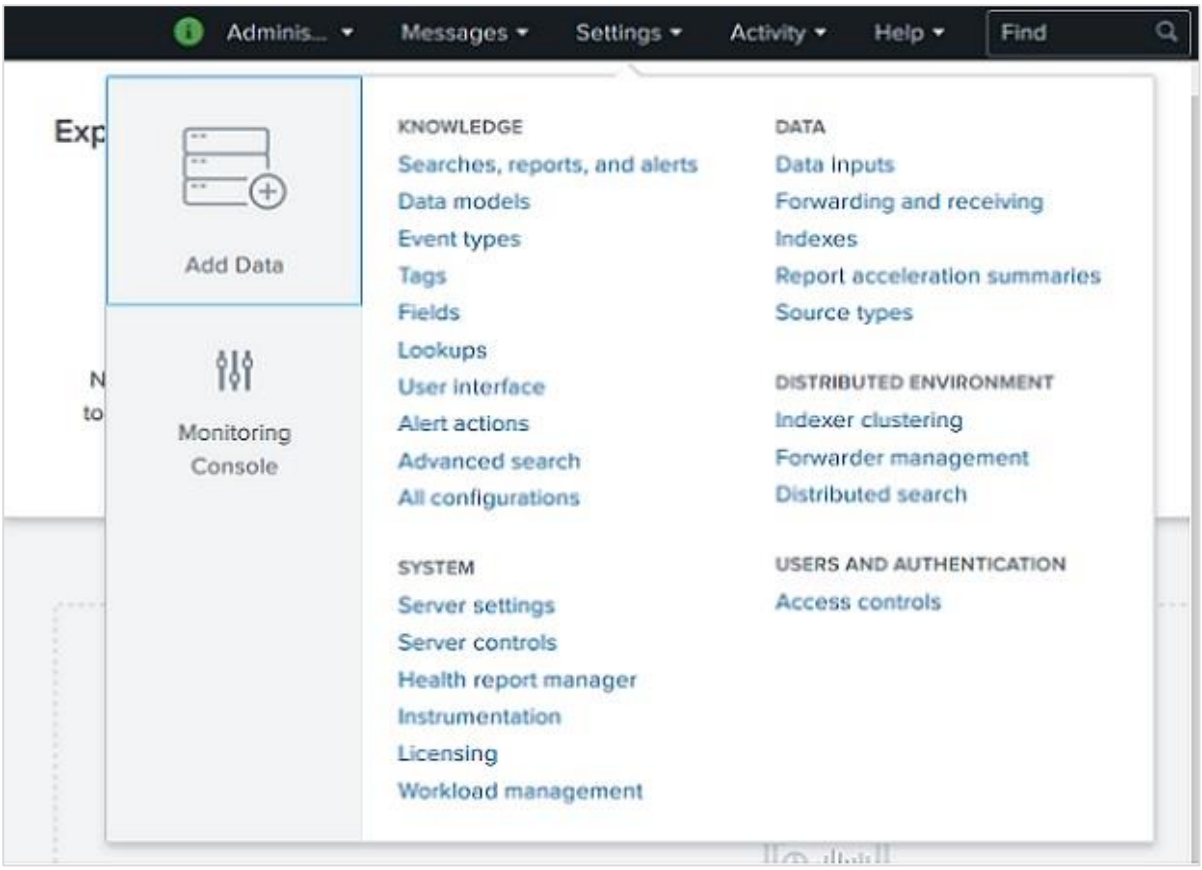
Further from the administrator link, we can also navigate to the preferences option where we can set the time zone and home application on which the landing page will open after your login. Currently, it opened on the Home page as shown below:



Settings Link

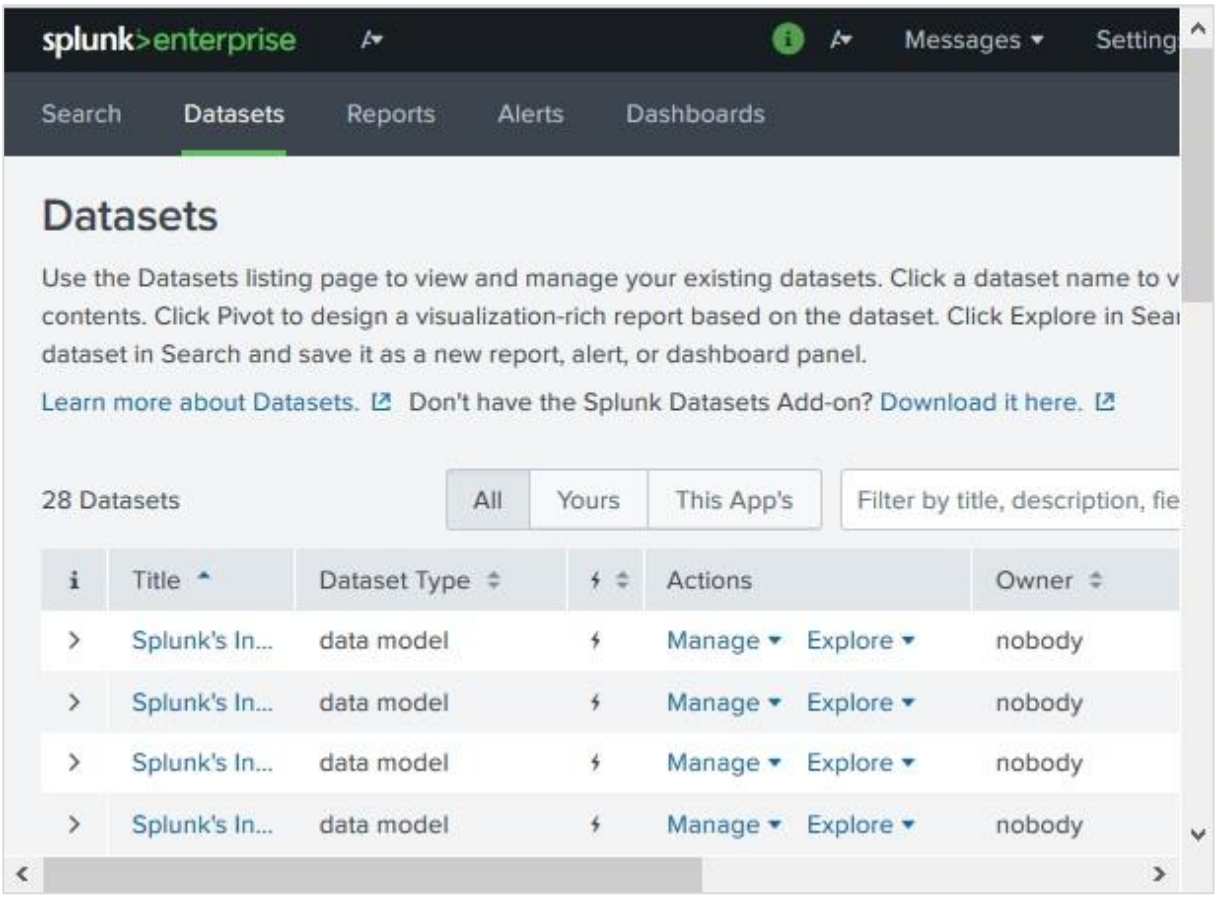
This is a link which shows all the core features available in Splunk. For example, you can add the lookup files and lookup definitions by choosing the lookup link.

We will discuss the important settings of these links in the subsequent chapters.



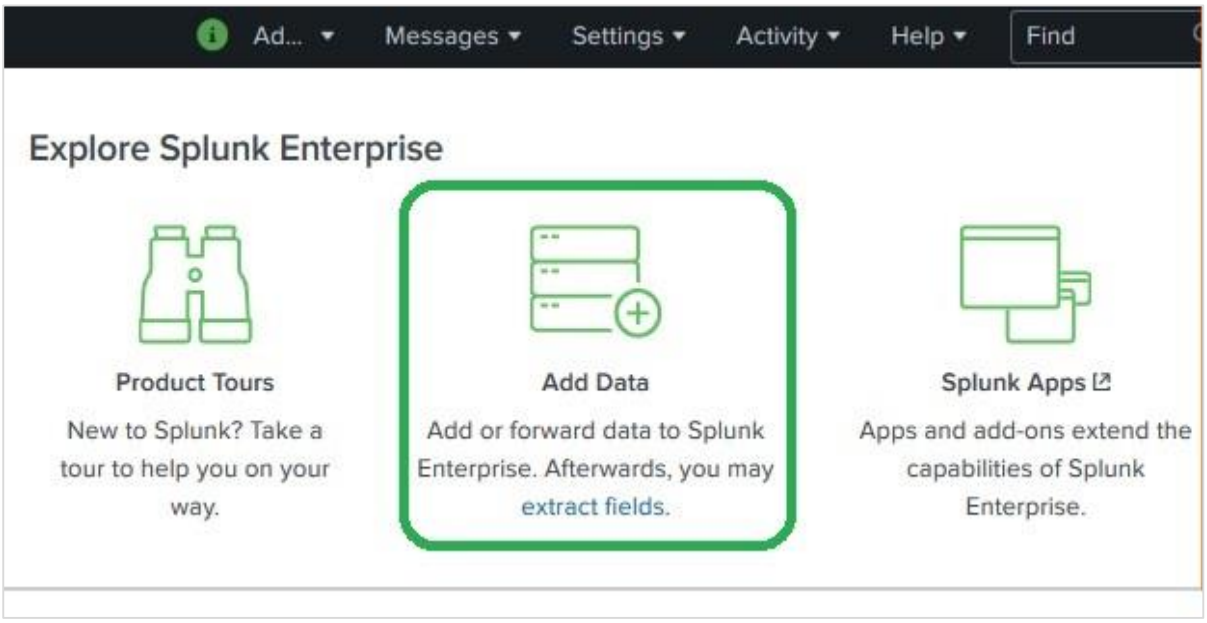
Search and Reporting Link

The search and reporting link takes us to the features where we can find the data sets that are available for searching the reports and alerts created for these searches. It is clearly shown in the below screenshot:



4. Splunk – Data Ingestion

Data ingestion in Splunk happens through the **Add Data** feature which is part of the search and reporting app. After logging in, the Splunk interface home screen shows the **Add Data** icon as shown below.



On clicking this button, we are presented with the screen to select the source and format of the data we plan to push to Splunk for analysis.

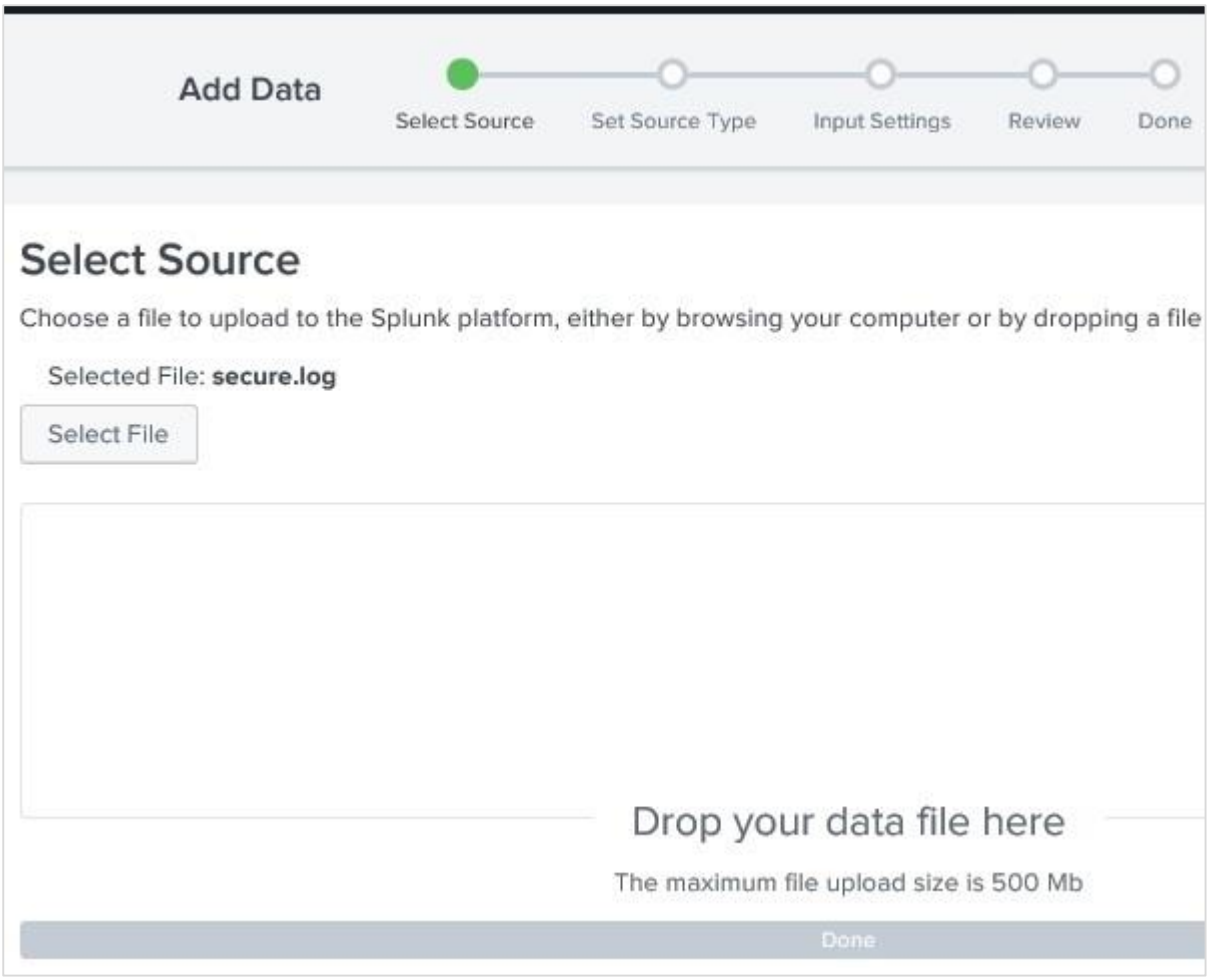
Gathering The Data

We can get the data for analysis from the Official Website of Splunk. Save this file and unzip it in your local drive. On opening the folder, you can find three files which have different formats. They are the log data generated by some web apps. We can also gather another set of data provided by Splunk which is available at from the Official Splunk webpage.

We will use data from both these sets for understanding the working of various features of Splunk.

Uploading data

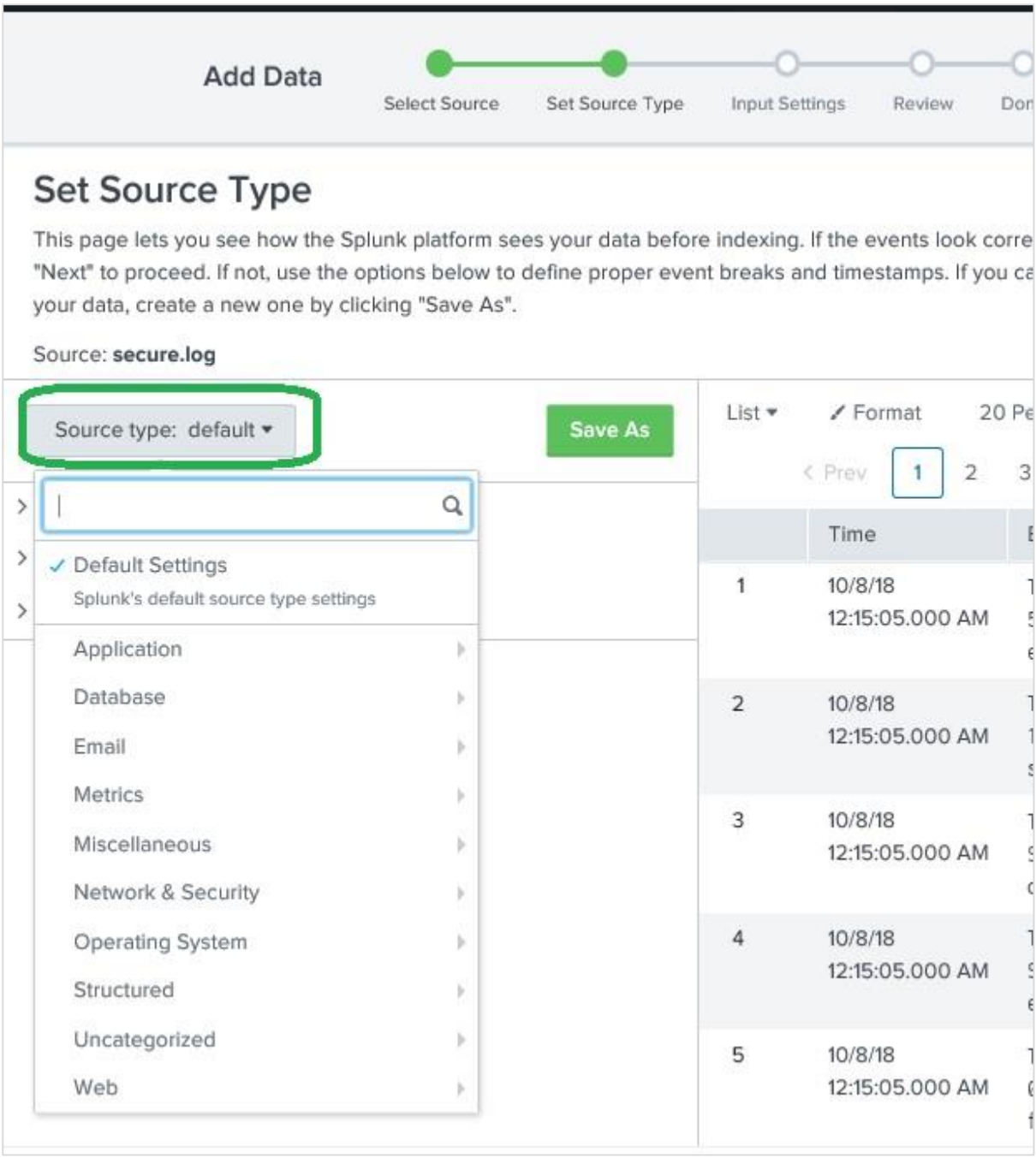
Next, we choose the file, **secure.log** from the folder, **mailsv** which we have kept in our local system as mentioned in the previous paragraph. After selecting the file, we move to next step using the green coloured next button in the top right corner.



Selecting Source Type

Splunk has an in-built feature to detect the type of the data being ingested. It also gives the user an option to choose a different data type than the chosen by Splunk. On clicking the source type drop down, we can see various data types that Splunk can ingest and enable for searching.

In the current example given below, we choose the default source type.



Input Settings

In this step of data ingestion, we configure the host name from which the data is being ingested. Following are the options to choose from, for the host name:

Constant value

It is the complete host name where the source data resides.

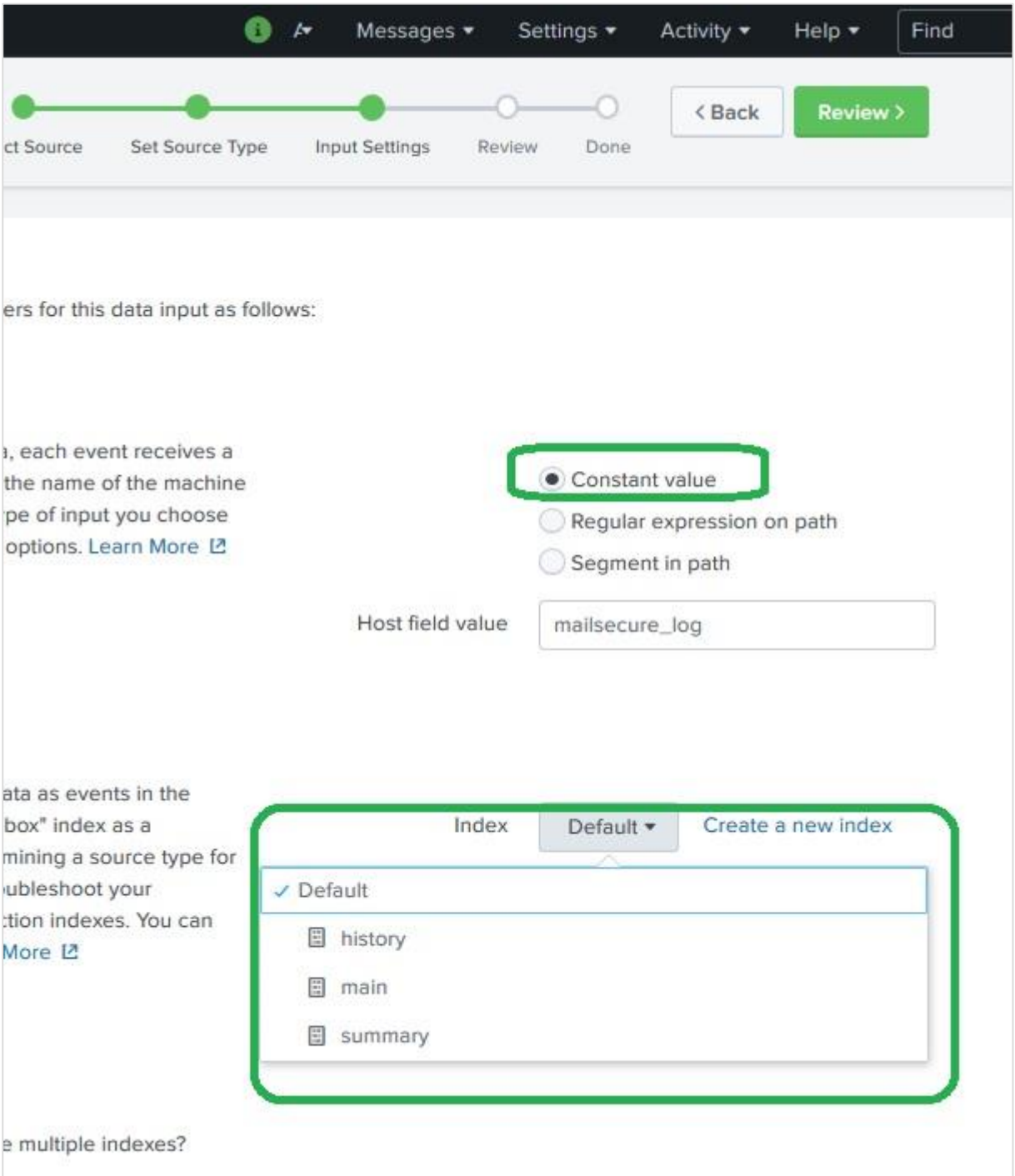
regex on path

When you want to extract the host name with a regular expression, enter the regex for in the regular expression field.

segment in path

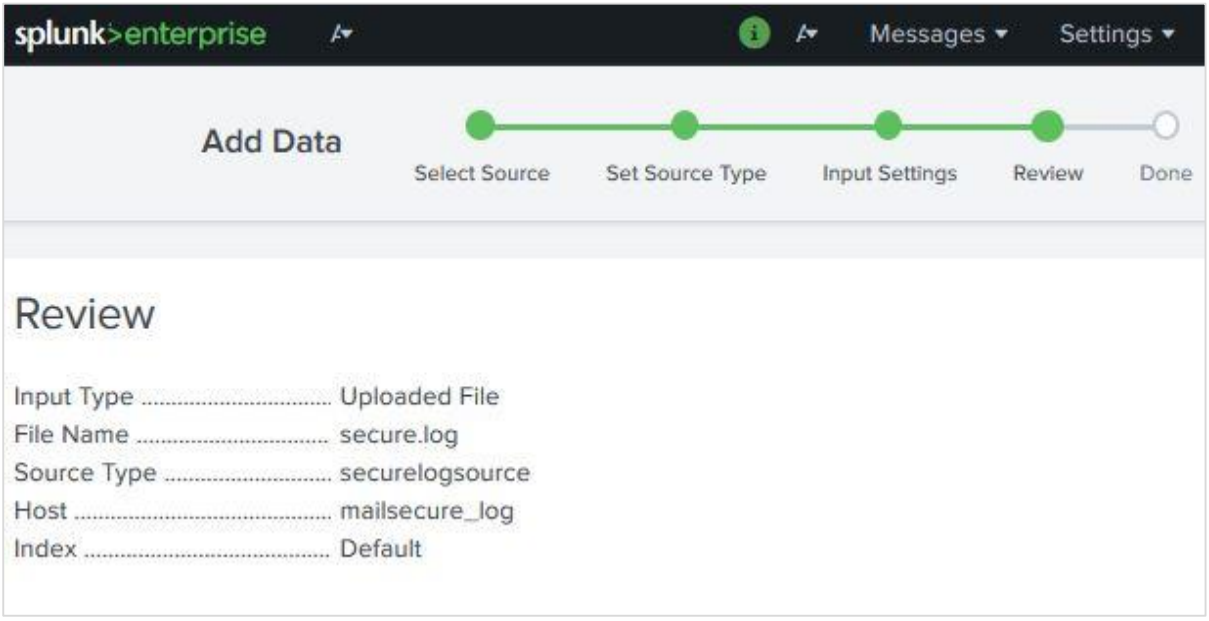
When you want to extract the host name from a segment in your data source's path, enter the segment number in the Segment number field. For example, if the path to the source is **/var/log/** and you want the third segment (the host server name) to be the host value, enter "3".

Next, we choose the index type to be created on the input data for searching. We choose the default index strategy. The summary index only creates summary of the data through aggregation and creates index on it while the history index is for storing the search history. It is clearly depicted in the image below:

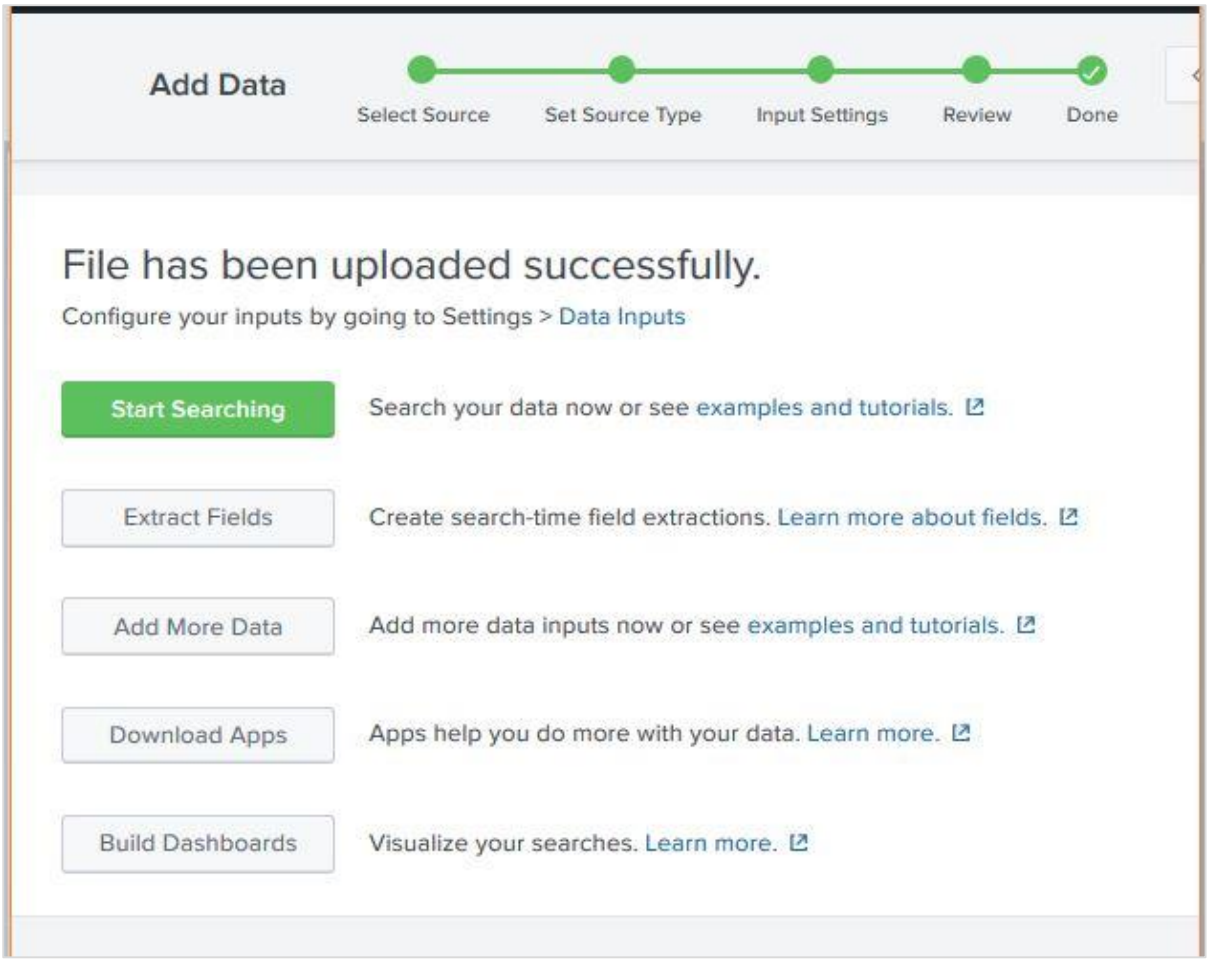


Review Settings

After clicking on the next button, we see a summary of the settings we have chosen. We review it and choose Next to finish the uploading of data.



On finishing the load, the below screen appears which shows the successful data ingestion and further possible actions we can take on the data.



5. Splunk – Source Types

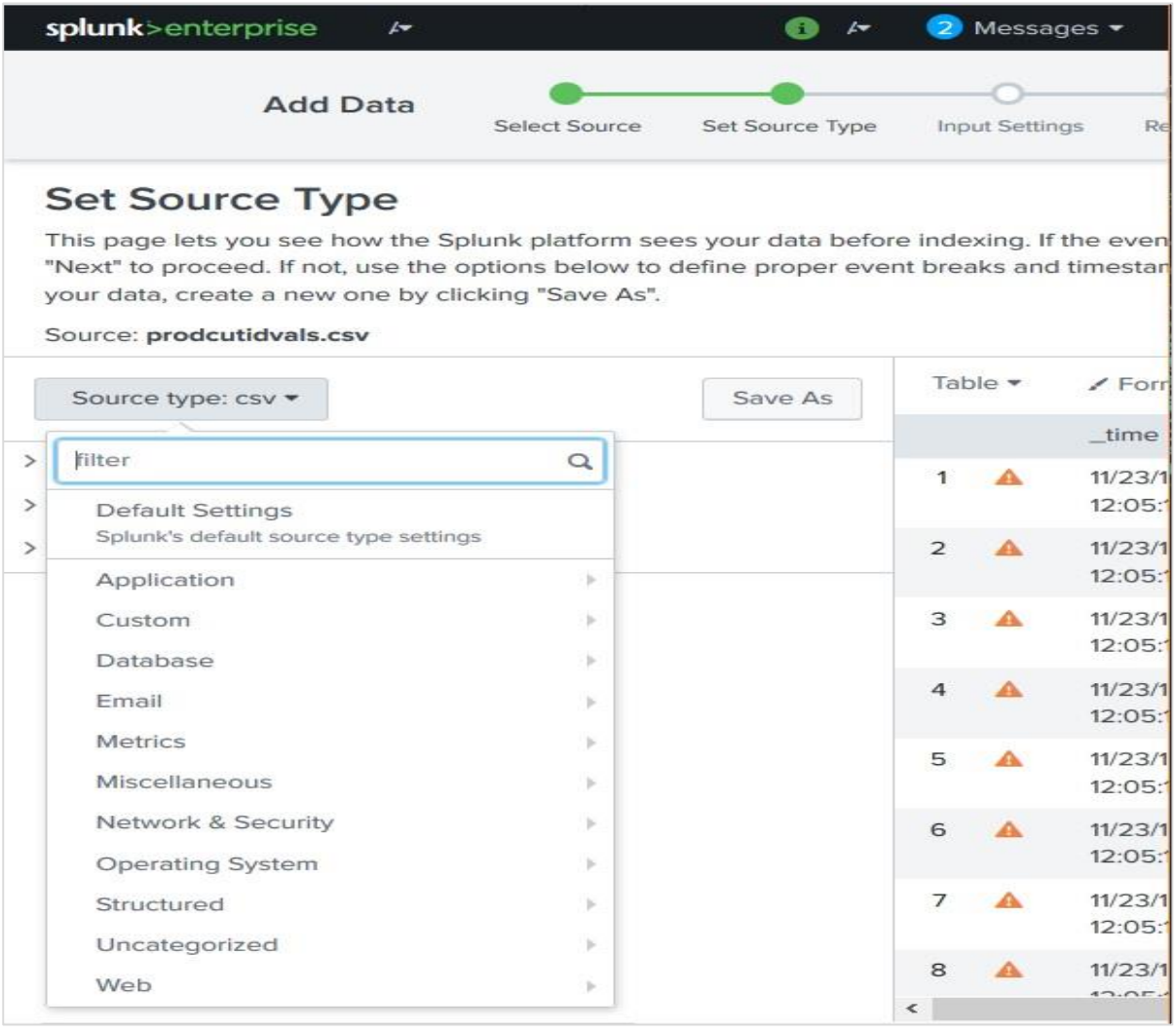
All the incoming data to Splunk are first judged by its inbuilt data processing unit and classified to certain data types and categories. For example, if it is a log from apache web server, Splunk is able to recognize that and create appropriate fields out of the data read.

This feature in Splunk is called source type detection and it uses its built-in source types that are known as "pretrained" source types to achieve this.

This makes things easier for analysis as the user does not have to manually classify the data and assign any data types to the fields of the incoming data.

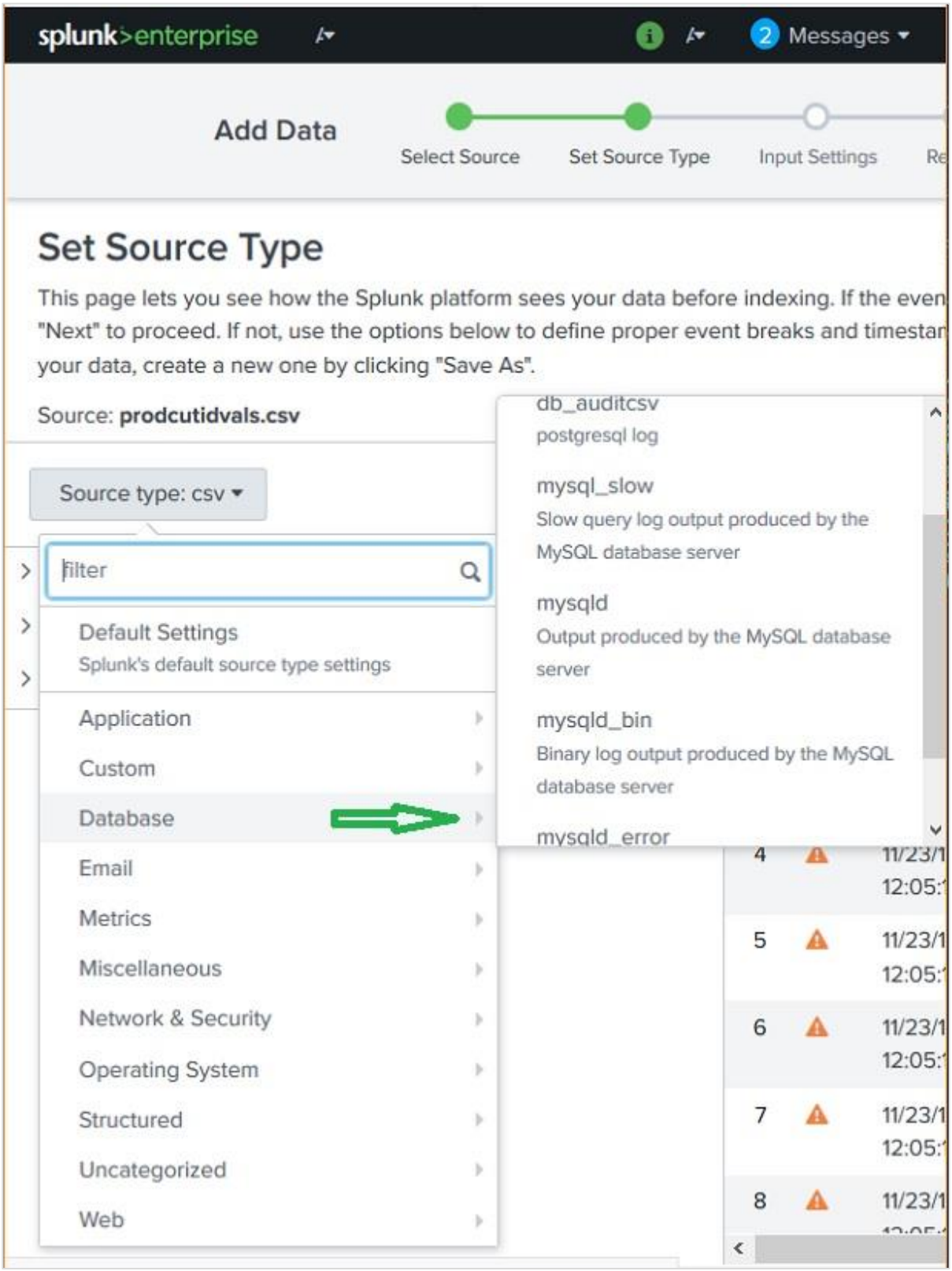
Supported Source Types

The supported source types in Splunk can be seen by uploading a file through the **Add Data** feature and then selecting the dropdown for Source Type. In the below image, we have uploaded a CSV file and then checked for all the available options.



Source Type Sub-Category

Even in those categories, we can further click to see all the sub categories that are supported. So when you choose the database category, you can find the different types of databases and their supported files which Splunk can recognize.



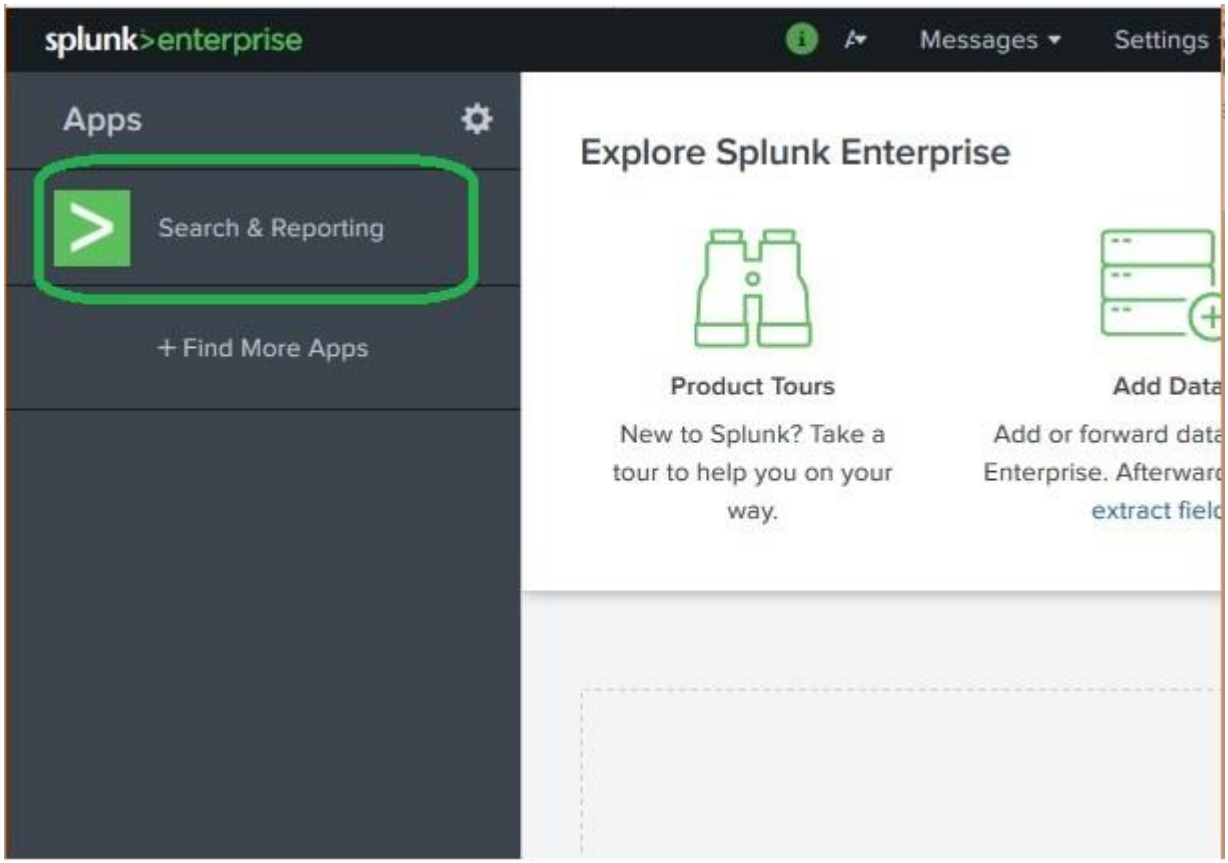
Pre-Trained Source Types

The below table lists some of the important pre-trained source types Splunk recognizes:

Source Type Name	Nature
access_combined	NCSA combined format http web server logs (can be generated by apache or other web servers)
access_combined_wcookie	NCSA combined format http web server logs (can be generated by apache or other web servers), with cookie field added at end
apache_error	Standard Apache web server error log
linux_messages_syslog	Standard linux syslog (/var/log/messages on most platforms)
log4j	Log4j standard output produced by any J2EE server using log4j
mysqld_error	Standard mysql error log

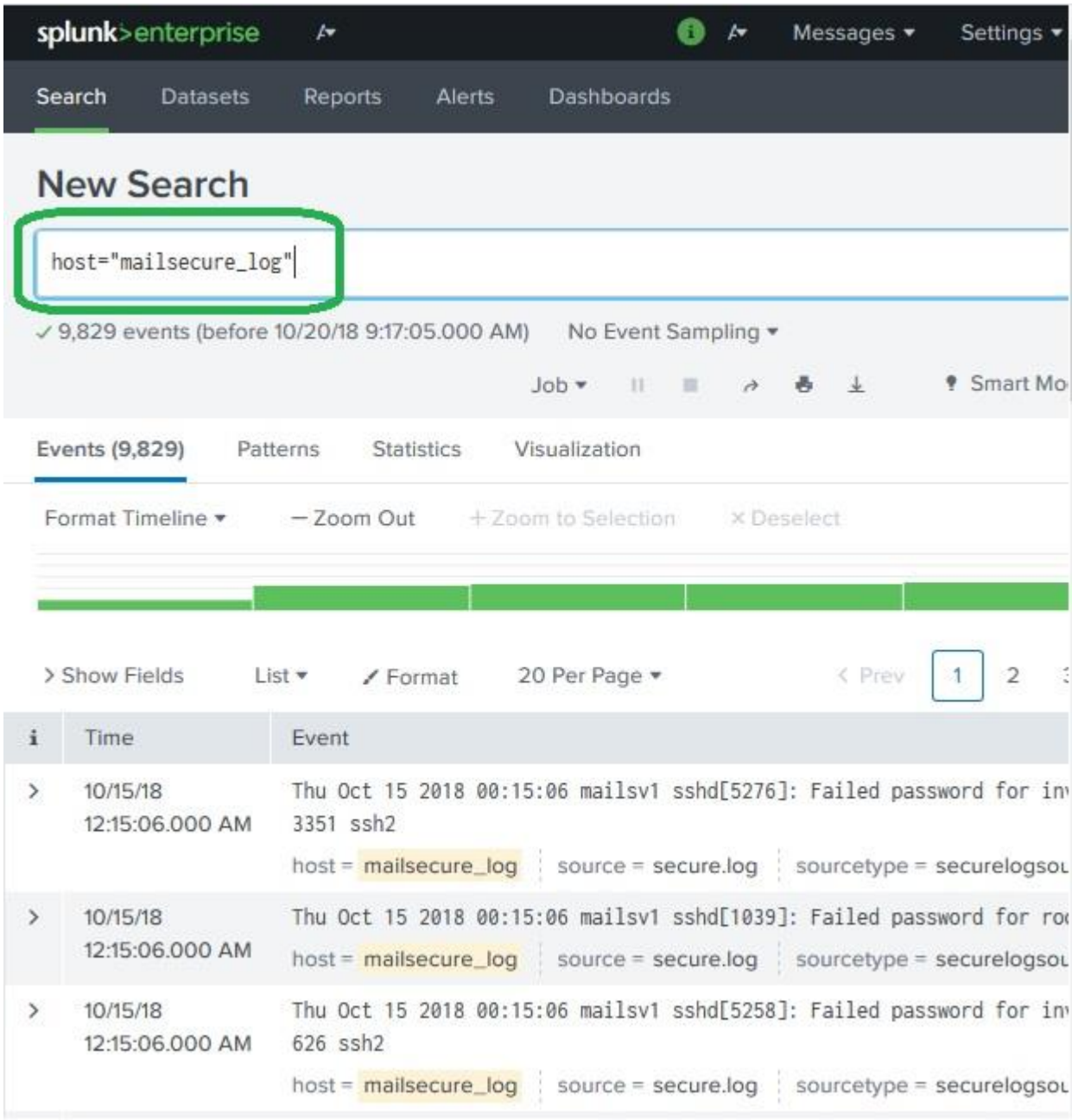
6. Splunk – Basic Search

Splunk has a robust search functionality which enables you to search the entire data set that is ingested. This feature is accessed through the app named as **Search & Reporting** which can be seen in the left side bar after logging in to the web interface.



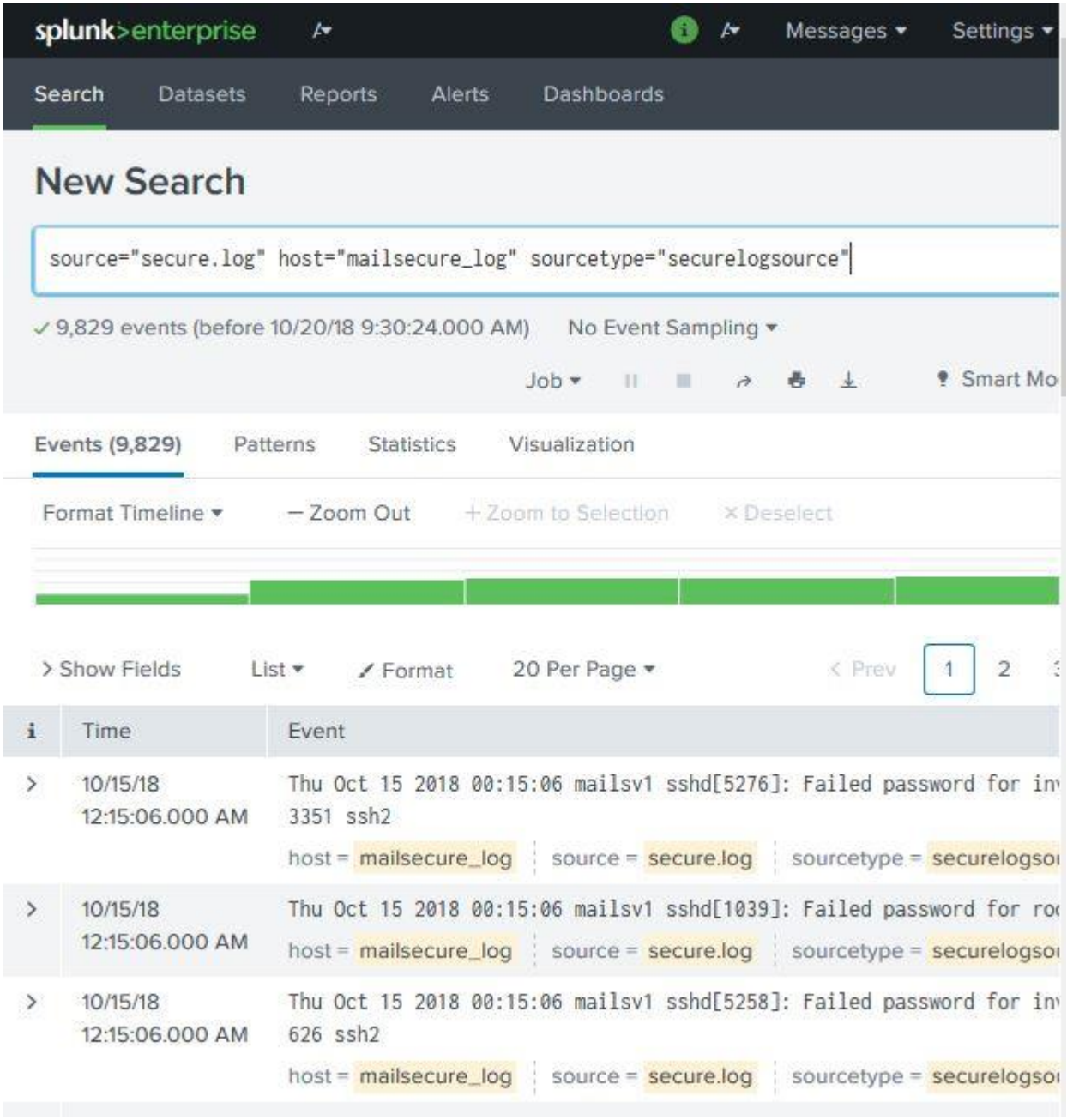
On clicking on the **search & Reporting** app, we are presented with a search box, where we can start our search on the log data that we uploaded in the previous chapter.

We type the host name in the format as shown below and click on the search icon present in the right most corner. This gives us the result highlighting the search term.



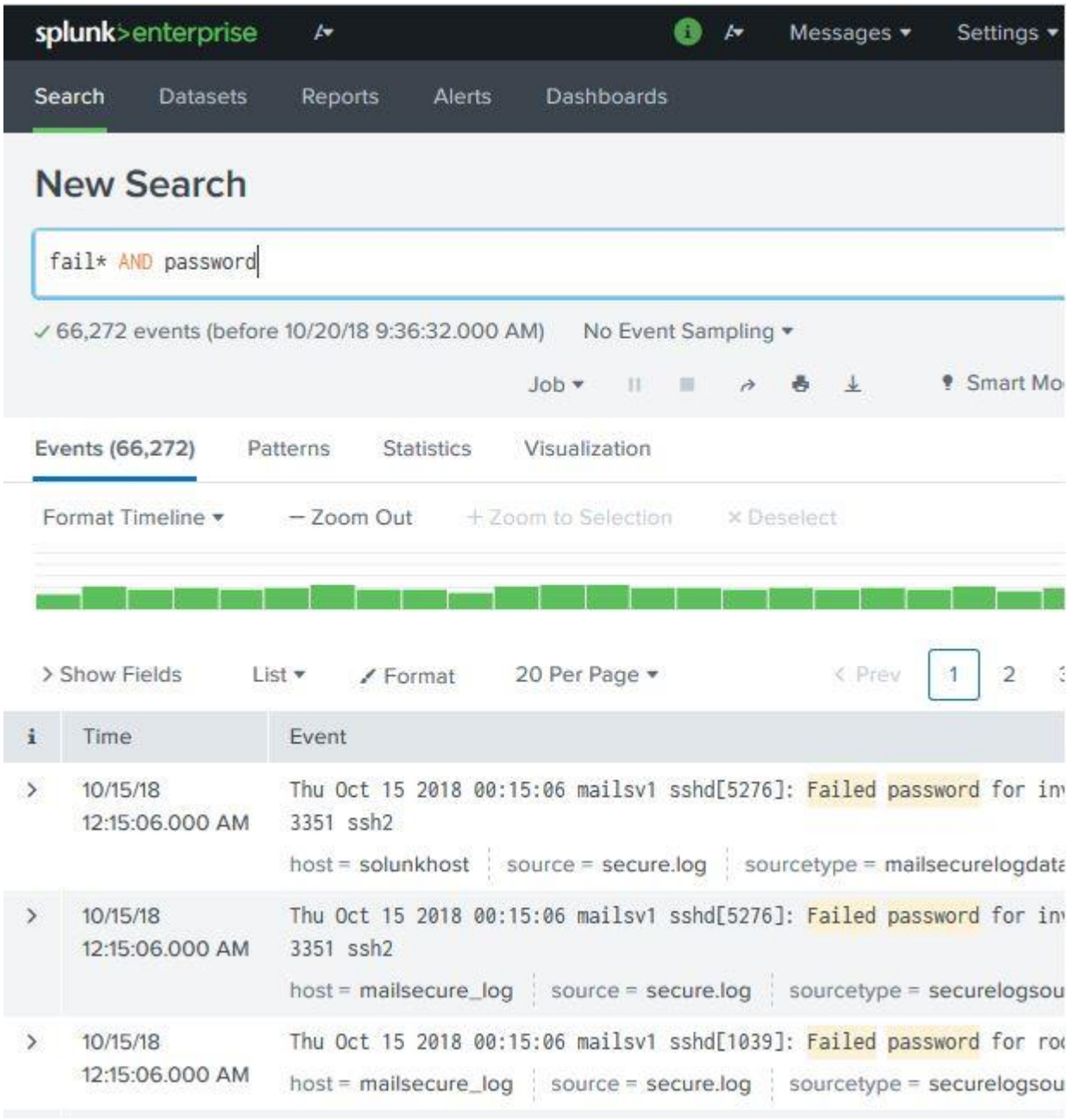
Combining Search Terms

We can combine the terms used for searching by writing them one after another but putting the user search strings under double quotes.



Using Wild Card

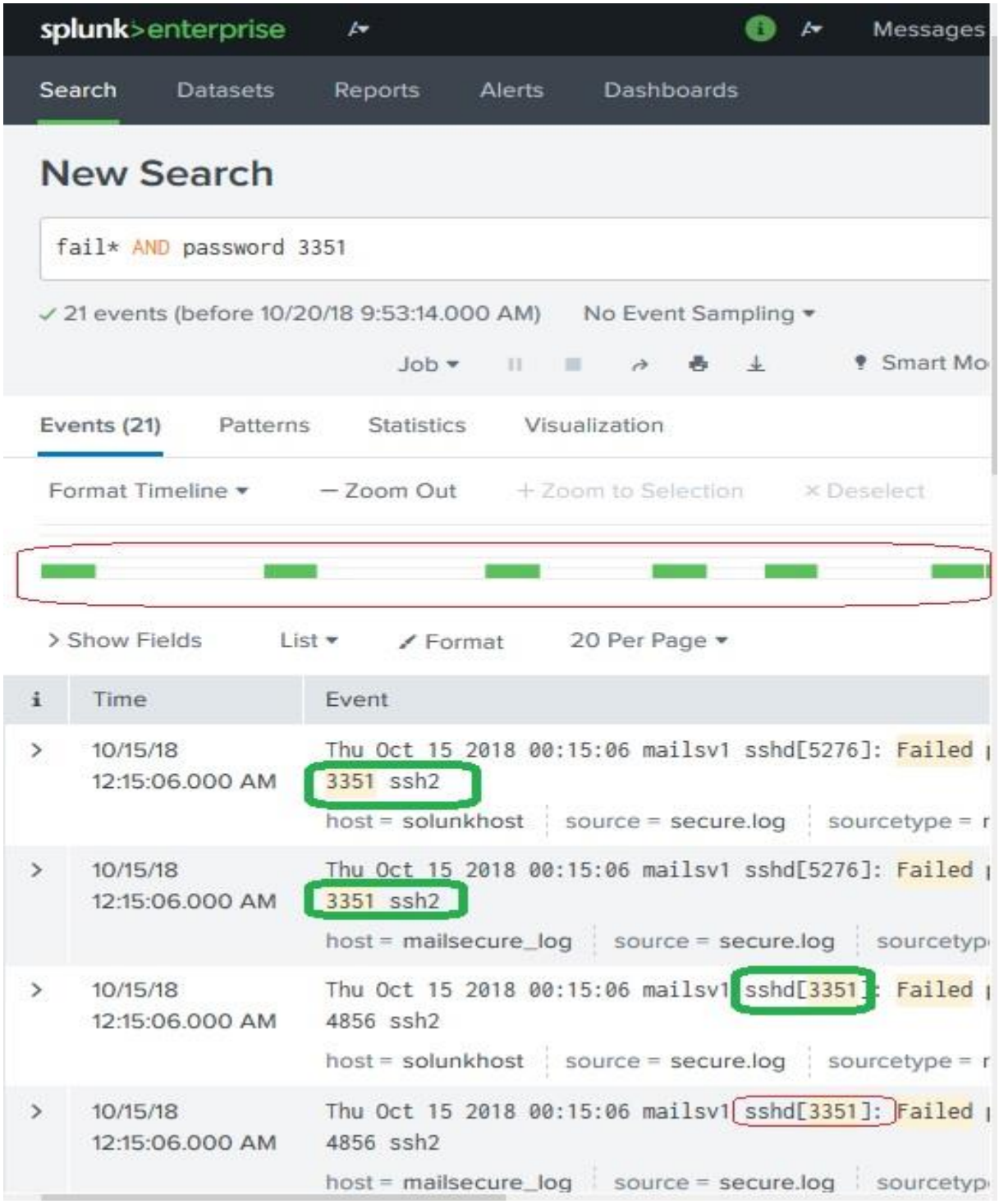
We can use wild cards in our search option combined with the **AND/OR** operators. In the below search, we get the result where the log file has the terms containing fail, failed, failure, etc., along with the term password in the same line.



Refining Search Results

We can further refine the search result by selecting a string and adding it to the search. In the below example, we click over the string **3351** and select the option **Add to Search**.

After **3351** is added to the search term, we get the below result which shows only those lines from the log containing 3351 in them. Also mark how the time line of the search result has changed as we have refined the search.

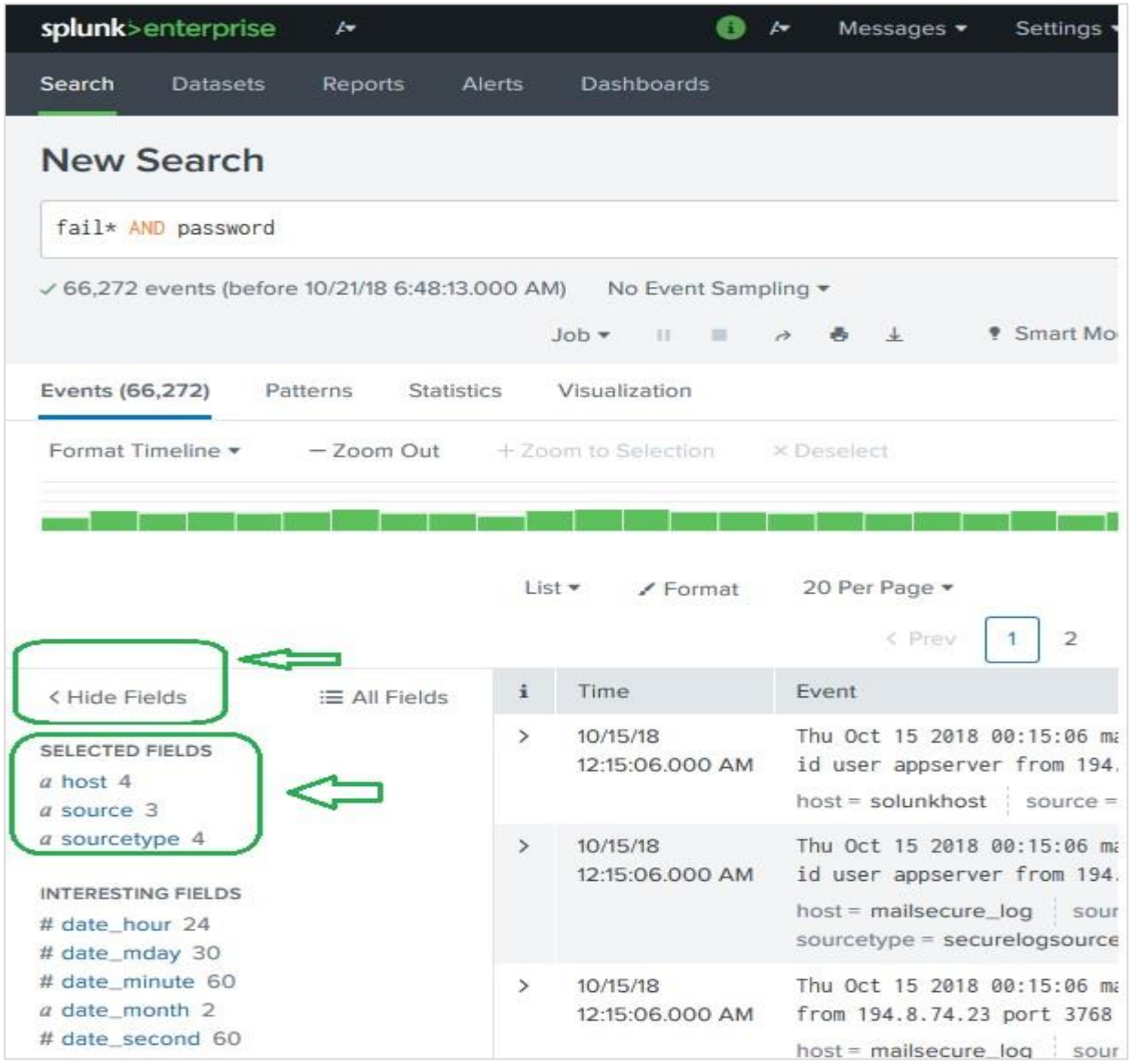


7. Splunk – Field Searching

When Splunk reads the uploaded machine data, it interprets the data and divides it into many fields which represent a single logical fact about the entire data record.

For example, a single record of information may contain server name, timestamp of the event, type of the event being logged whether login attempt or a http response, etc. Even in case of unstructured data, Splunk tries to divide the fields into key value pairs or separate them based on the data types they have, numeric and string, etc.

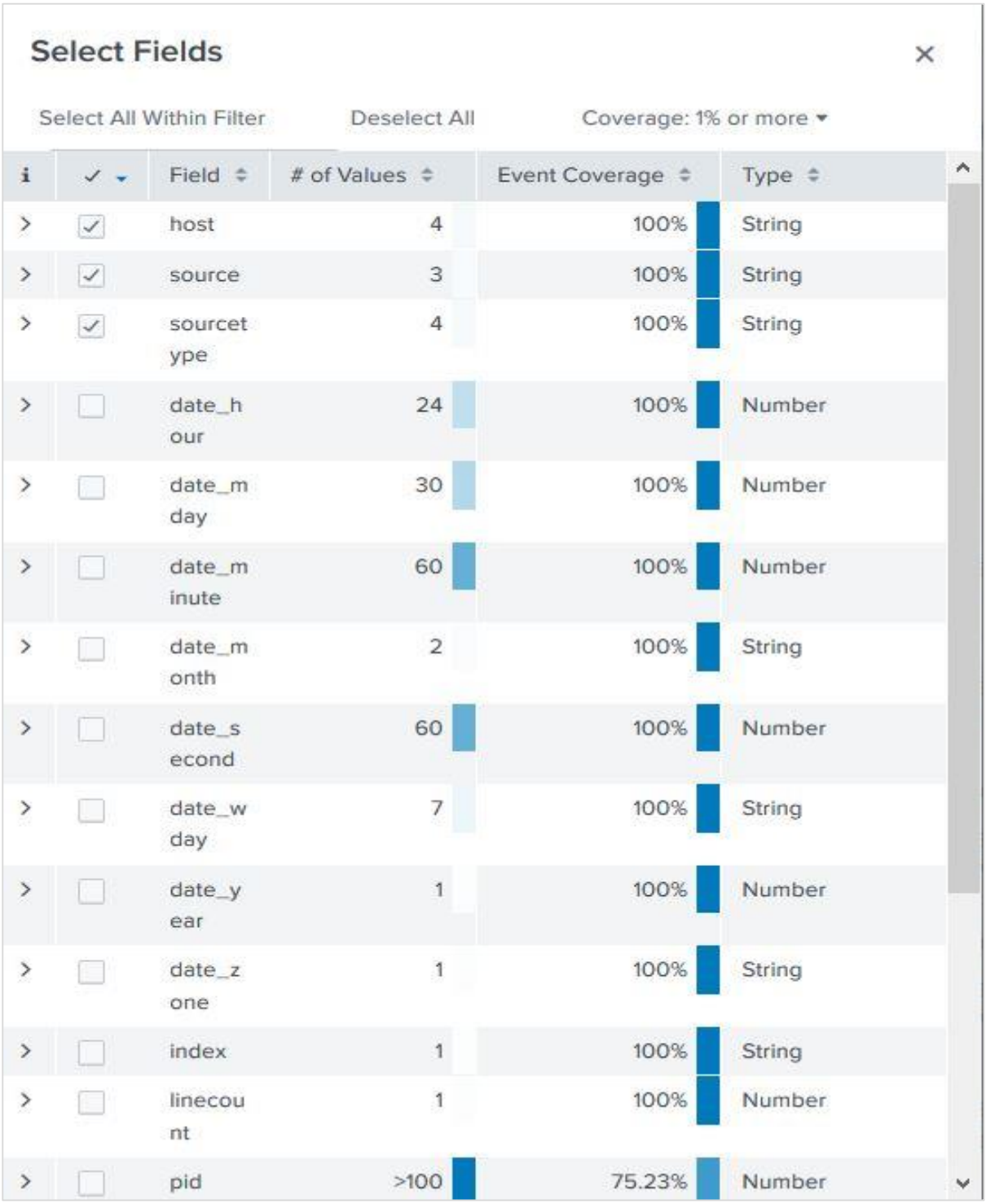
Continuing with the data uploaded in the previous chapter, we can see the fields from the **secure.log** file by clicking on the show fields link which will open up the following screen. We can notice the fields Splunk has generated from this log file.



Choosing the Fields

We can choose what fields to be displayed by selecting or unselecting the fields from the list of all fields. Clicking on **all fields** opens a window showing the list of all the fields. Some of these fields have check marks against them showing they are already selected. We can use the check boxes to choose our fields for display.

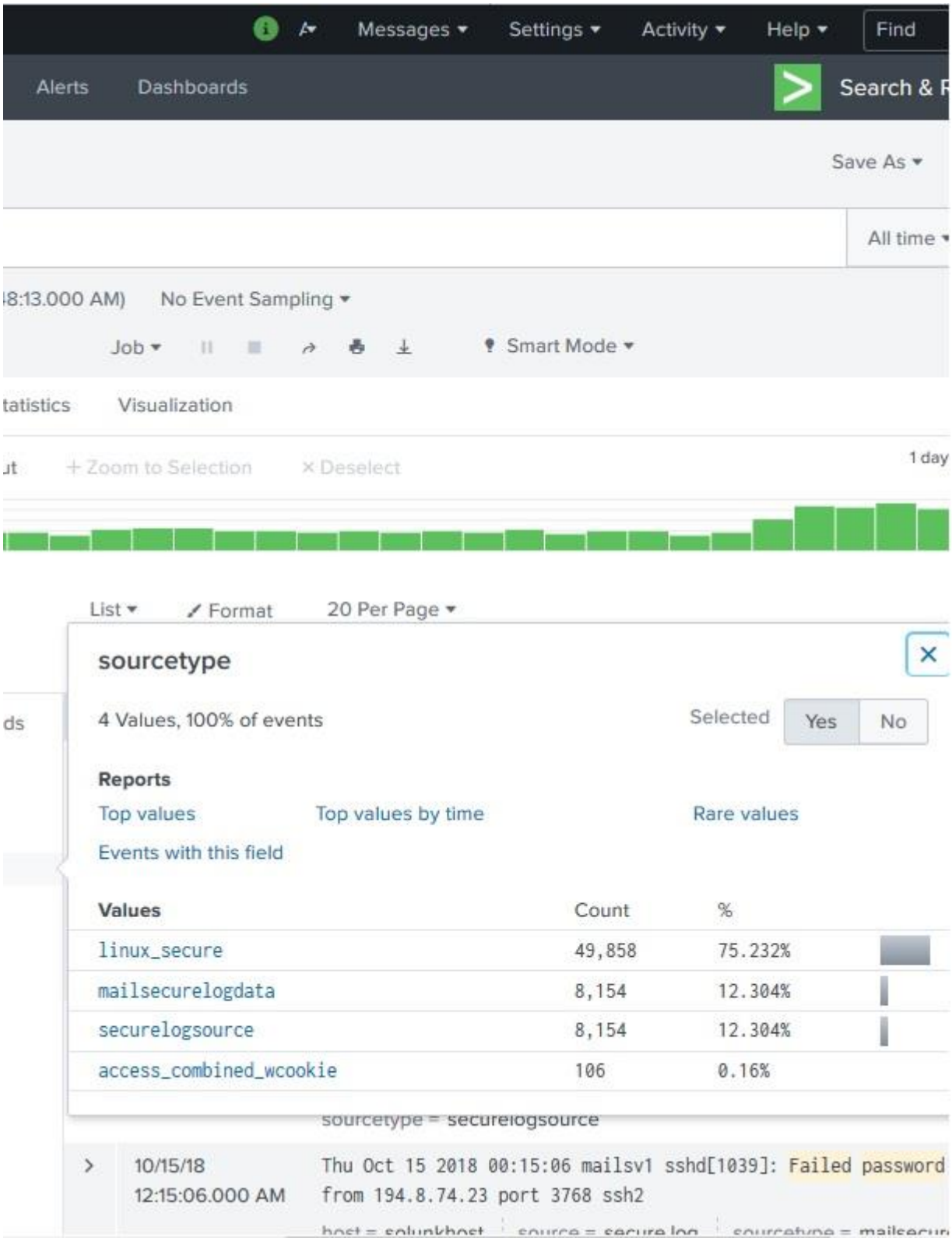
Besides the name of the field, it displays the number of distinct values the fields have, its data type and what percentage of events this field is present in.



i	✓	Field	# of Values	Event Coverage	Type
>	✓	host	4	100%	String
>	✓	source	3	100%	String
>	✓	sourcetype	4	100%	String
>		date_hour	24	100%	Number
>		date_mday	30	100%	Number
>		date_minute	60	100%	Number
>		date_month	2	100%	String
>		date_second	60	100%	Number
>		date_wday	7	100%	String
>		date_year	1	100%	Number
>		date_zone	1	100%	String
>		index	1	100%	String
>		linecount	1	100%	Number
>		pid	>100	75.23%	Number

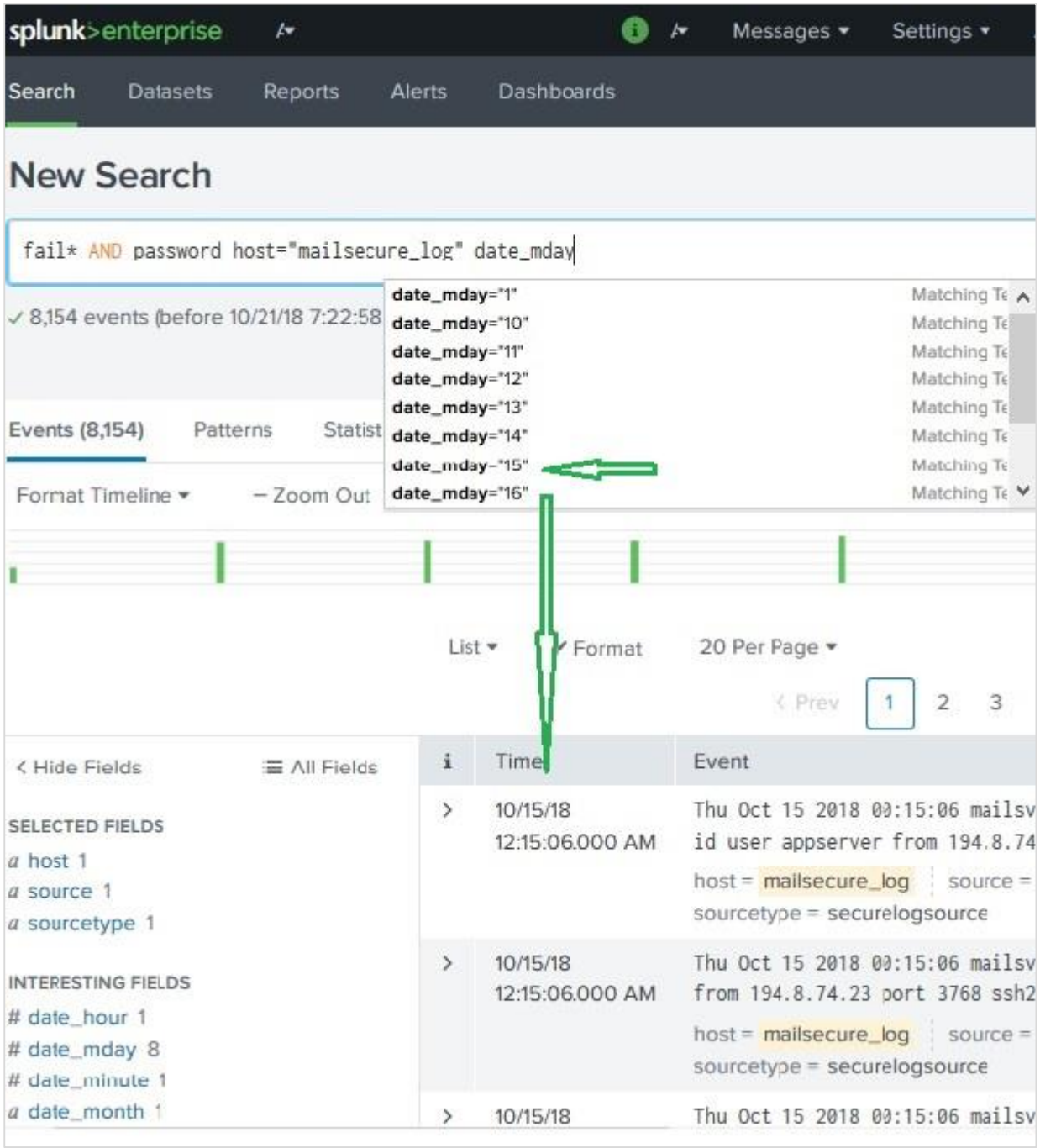
Field Summary

Very detailed stats for every selected field become available by clicking on the name of the field. It shows all the distinct values for the field, their count and their percentages.



Using Fields in Search

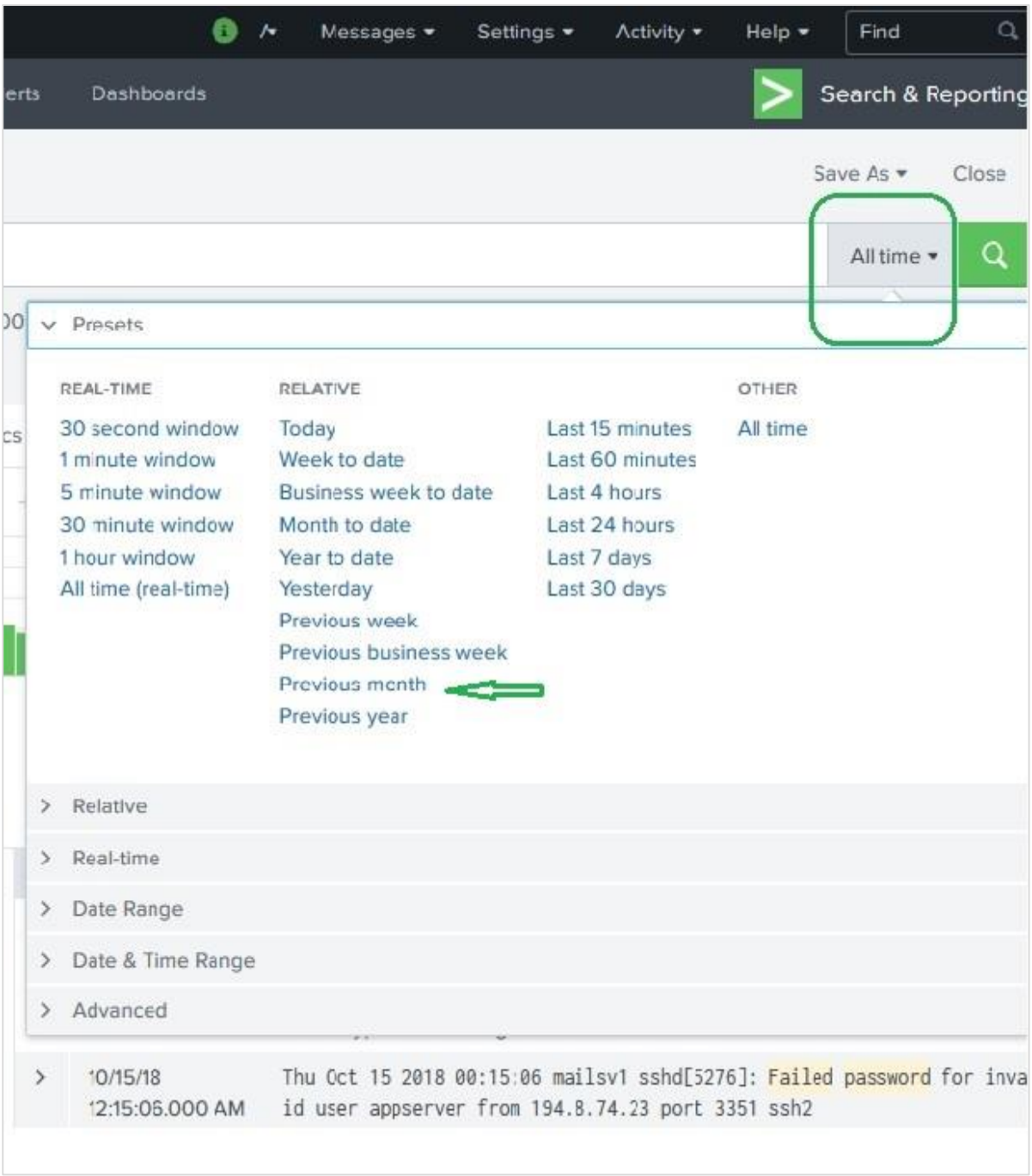
The field names can also be inserted into the search box along with the specific values for the search. In the below example, we aim to find all the records for the date, 15th Oct for the host named **mailsecure_log**. We get the result for this specific date.



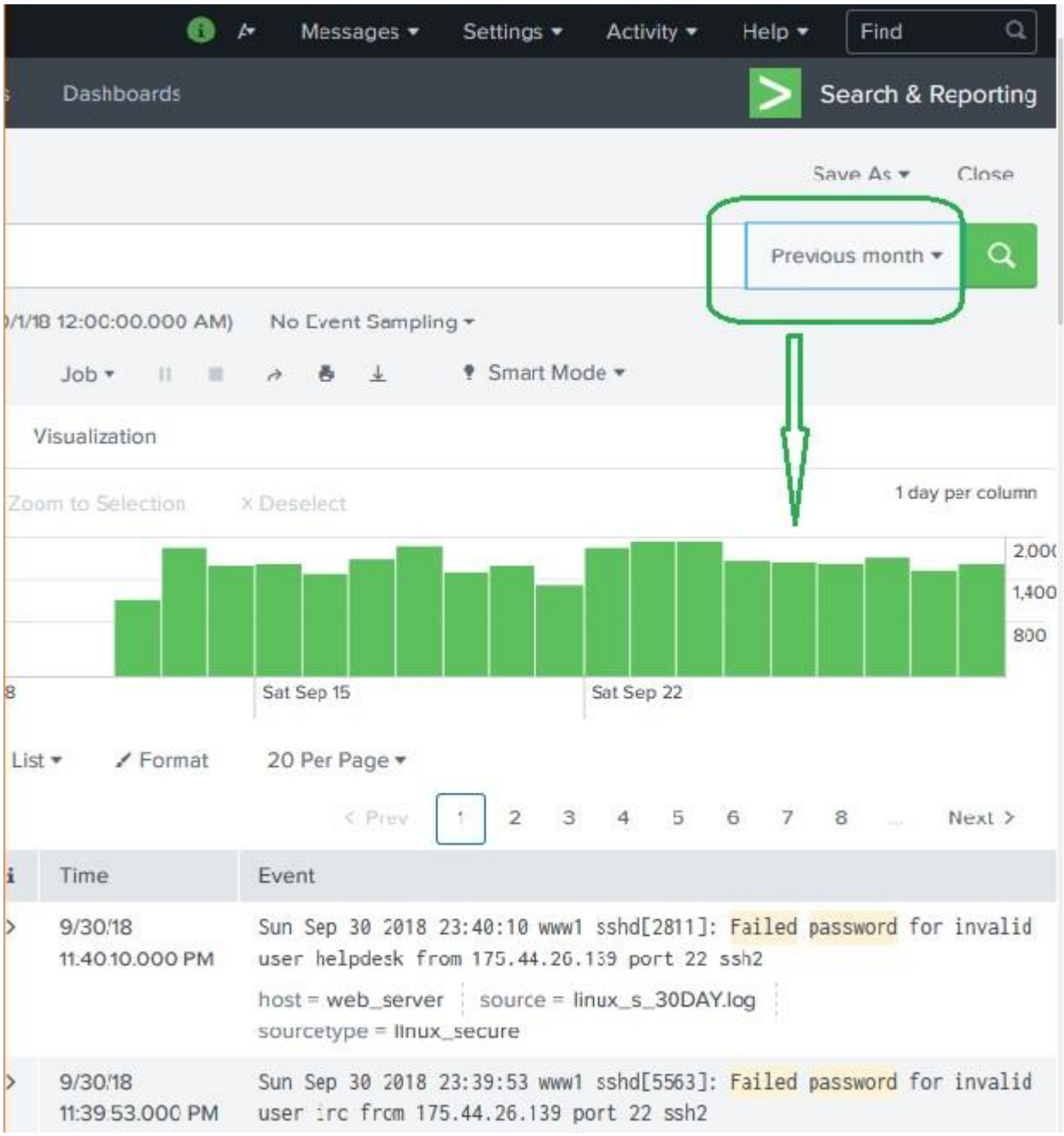
8. Splunk – Time Range Search

The Splunk web interface displays timeline which indicates the distribution of events over a range of time. There are preset time intervals from which you can select a specific time range, or you can customize the time range as per your need.

The below screen shows various preset timeline options. Choosing any of these options will fetch the data for only that specific time period which you can also analyse further, using the custom timeline options available.



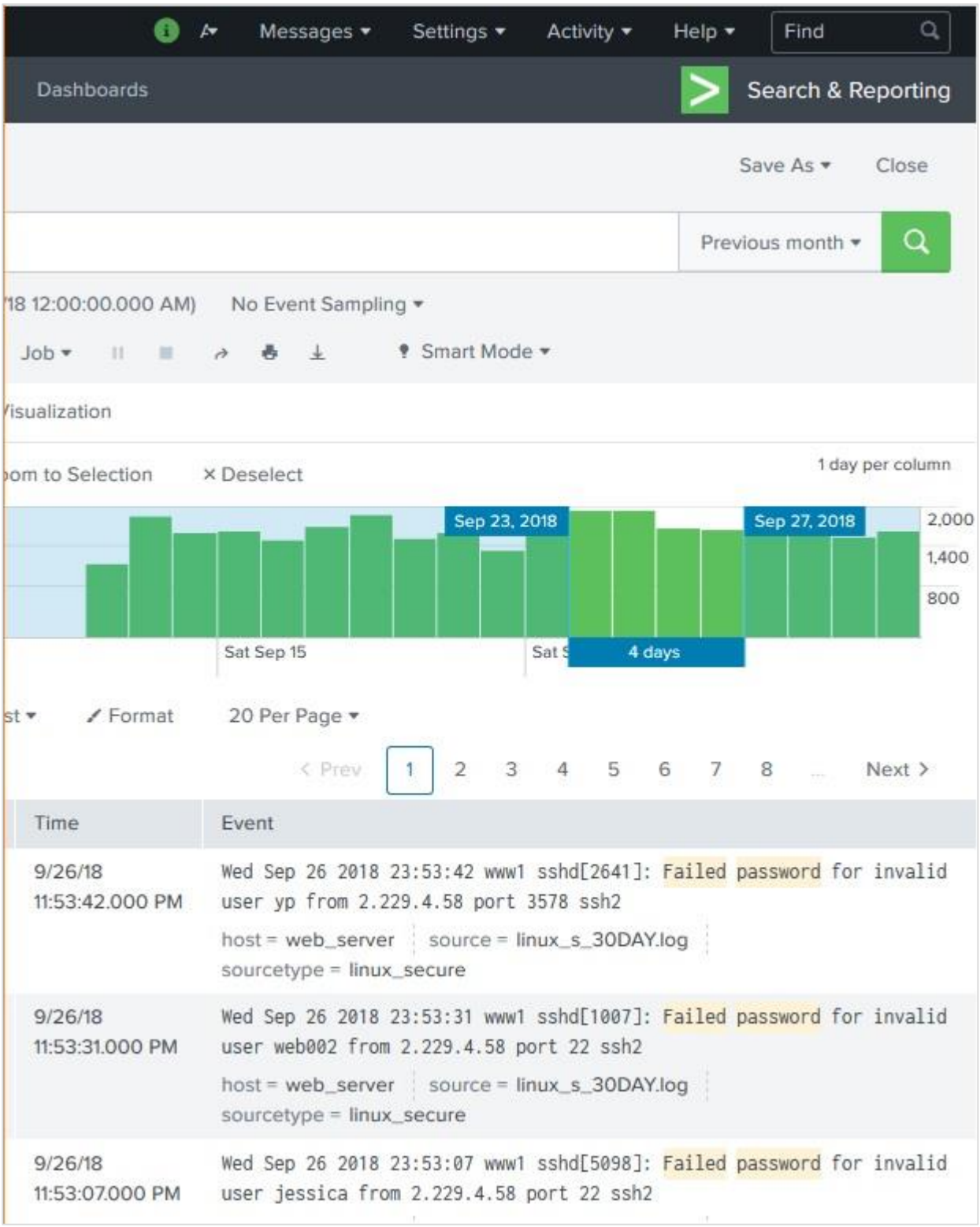
For example, choosing the previous month option gives us the result only for the previous month as you can see the in spread of the timeline graph below.



Selecting a Time Subset

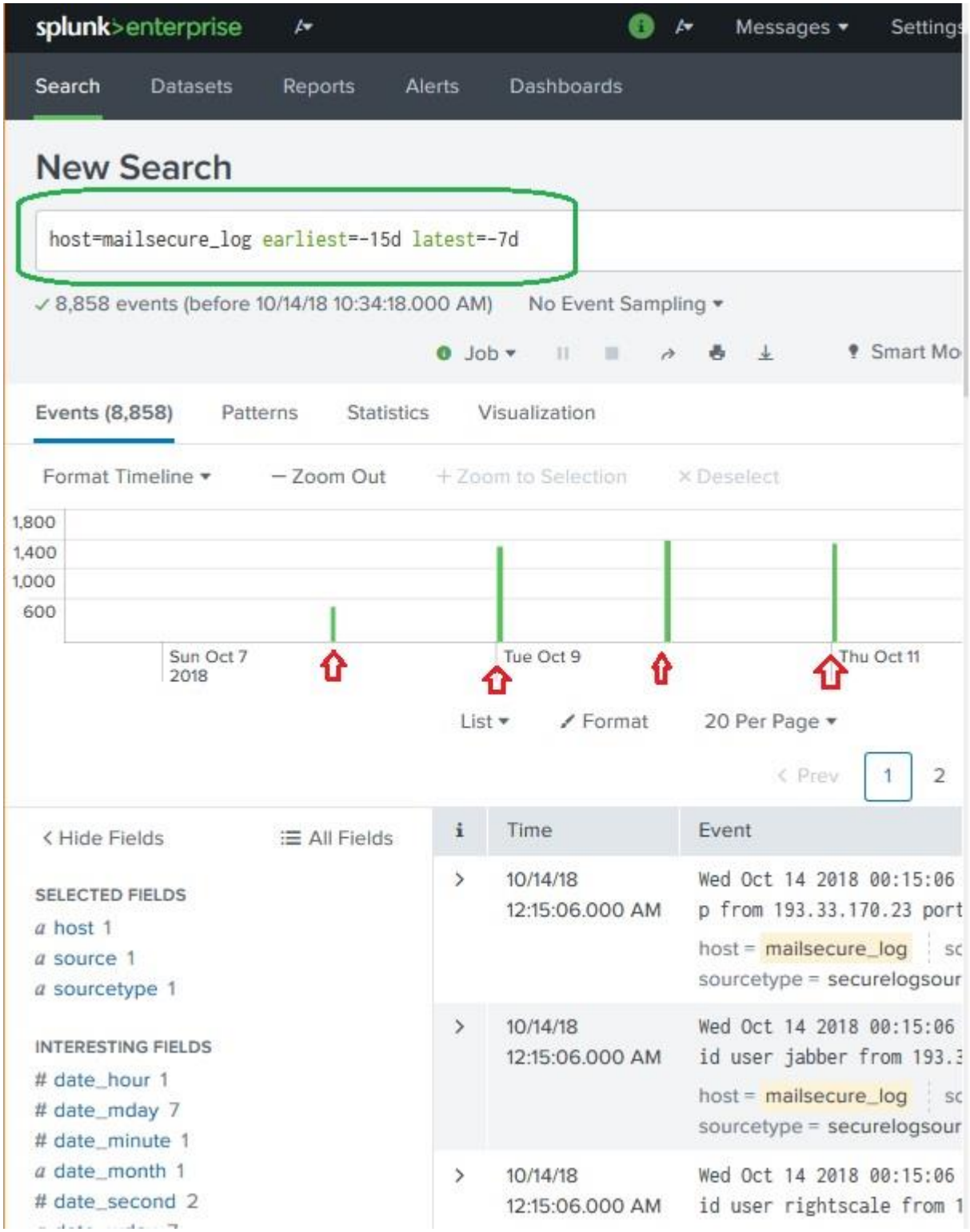
By clicking and dragging across the bars in the timeline, we can select a subset of the result that already exists. This does not cause the re-execution of the query. It only filters out the records from the existing result set.

Below image shows the selection of a subset from the result set:



Earliest and Latest

The two commands, earliest and latest can be used in the search bar to indicate the time range in between which you filter out the results. It is similar to selecting the time subset, but it is through commands rather than the option of clicking at a specific time line bar. So, it provides a finer control over that data range you can pick for your analysis.



In the above image, we give a time range between last 7 days to last 15 days. So, the data in between these two days is displayed.

Nearby Events

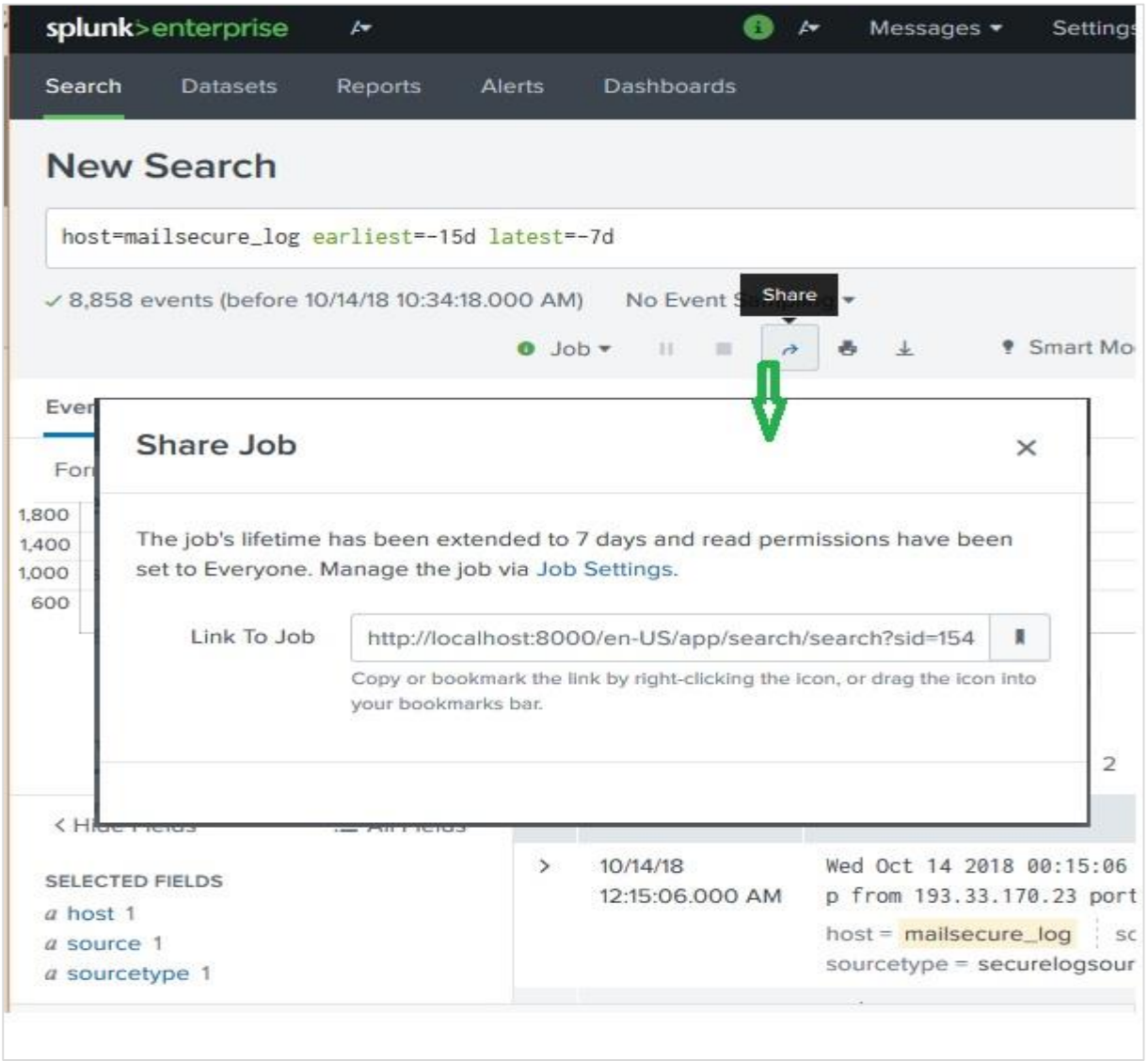
We can also find nearby events of a specific time by mentioning how close we want the events to be filtered out. We have the option of choosing the scale of the interval, like – seconds, minutes, days and week etc.

9. Splunk – Sharing Exporting

When you run a search query, the result is stored as a job in the Splunk server. While this job was created by one specific user, it can be shared across with other users so that they can start using this result set without the necessity of building the query for it again. The results can also be exported and saved as files which can be shared with users who do not use Splunk.

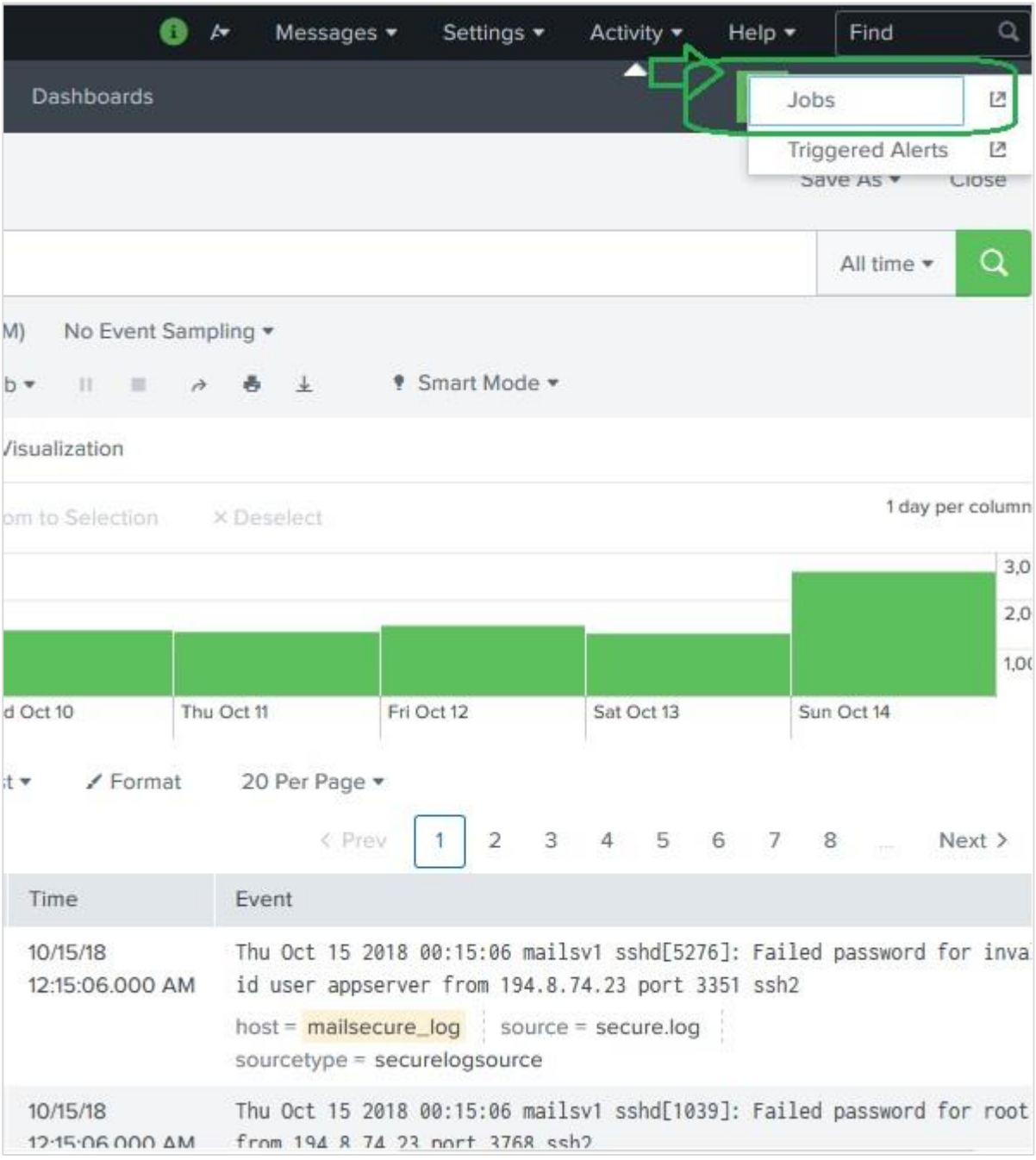
Sharing the Search Result

Once a query has run successfully, we can see a small upward arrow in the middle right of the web page. Clicking on this icon gives a URL where the query and the result can be accessed. There is a need to grant permission to the users who will be using this link. Permission is granted through the Splunk administration interface.

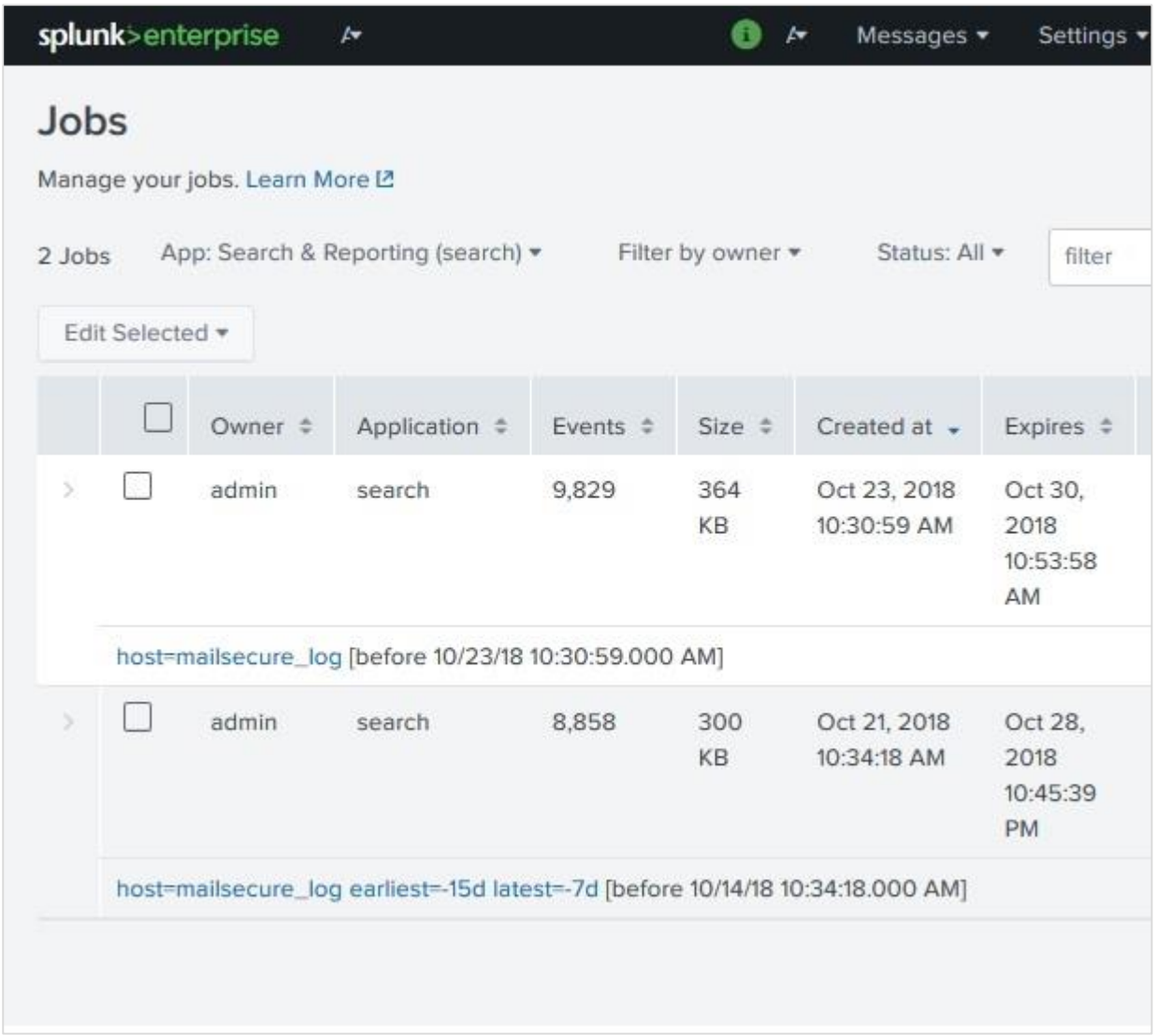


Finding the Saved Results

The jobs that are saved to be used by all users with appropriate permissions can be located by looking for the jobs link under the activity menu in the top right bar of the Splunk interface. In the below image, we click on the highlighted link named jobs to find the saved jobs.

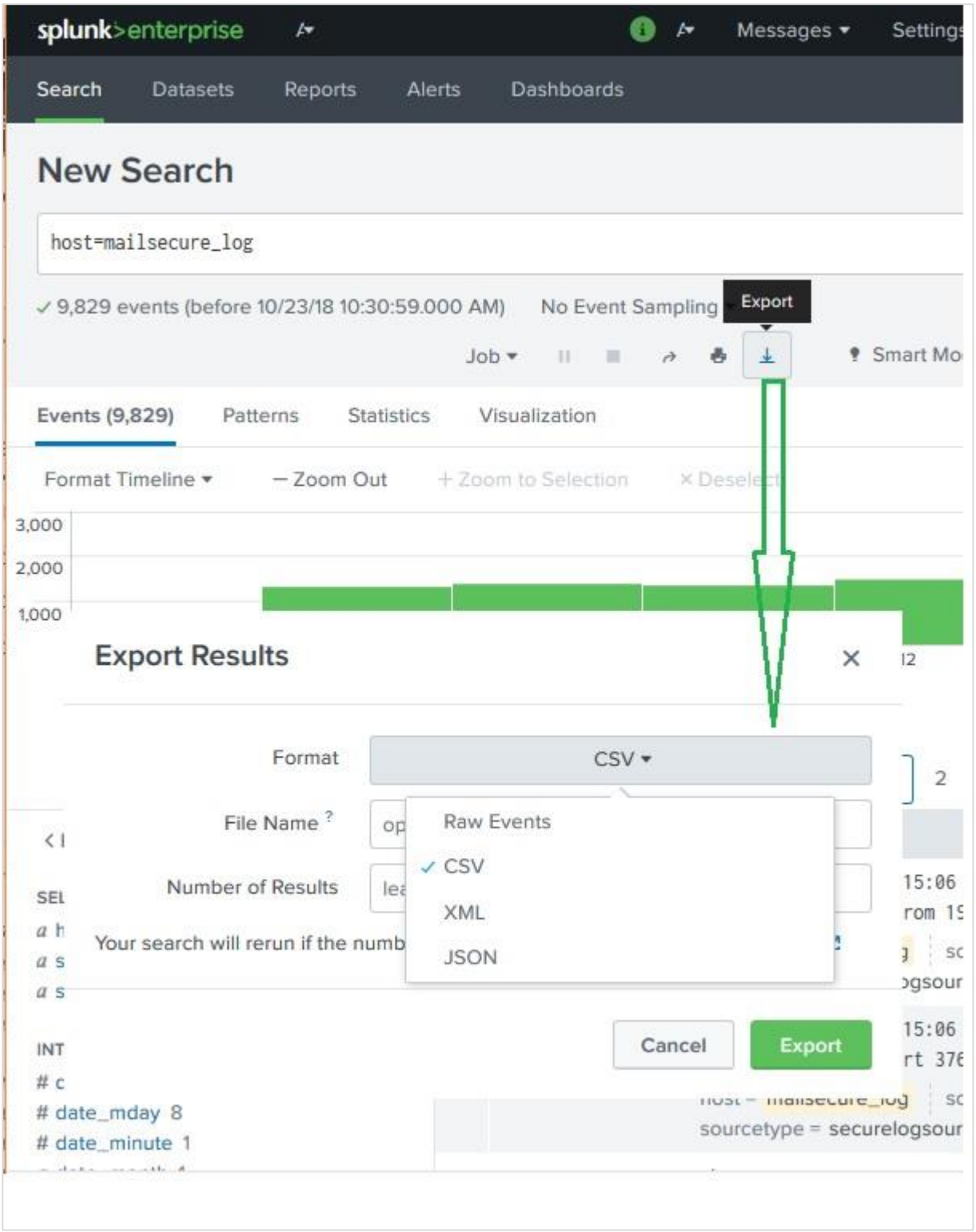


After the above link is clicked, we get the list of all the saved jobs as shown below. He, we have to note that there is an expiry date post where the saved job will automatically get removed from Splunk. You can adjust this date by selecting the job and clicking on Edit selected and then choosing Extend Expiration.



Exporting the Search Result

We can also export the results of a search into a file. The three different formats available for export are: CSV, XML and JSON. Clicking on the Export button after choosing the formats downloads the file from the local browser into the local system. This is explained in the below image:



10. Splunk – Search Language

The Splunk Search Processing Language (SPL) is a language containing many commands, functions, arguments, etc., which are written to get the desired results from the datasets. For example, when you get a result set for a search term, you may further want to filter some more specific terms from the result set. For this, you need some additional commands to be added to the existing command. This is achieved by learning the usage of SPL.

Components of SPL

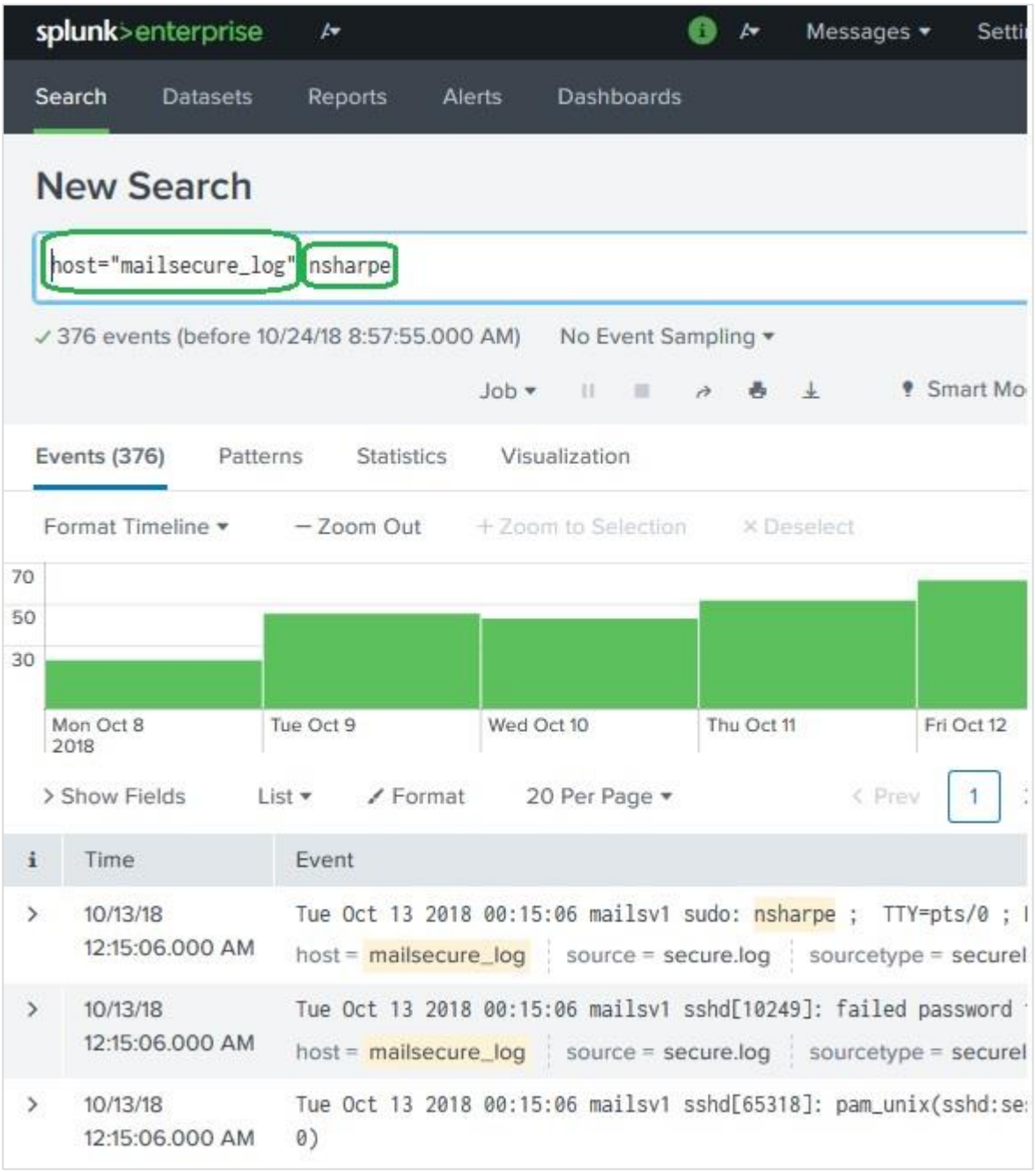
SPL has the following components:

- **Search Terms** – These are the keywords or phrases you are looking for.
- **Commands** – The action you want to take on the result set like format the result or count them.
- **Functions** – What are the computations you are going to apply on the results. Like Sum, Average etc.
- **Clauses** – How to group or rename the fields in the result set.

Let us discuss all the components with the help of images in the below section:

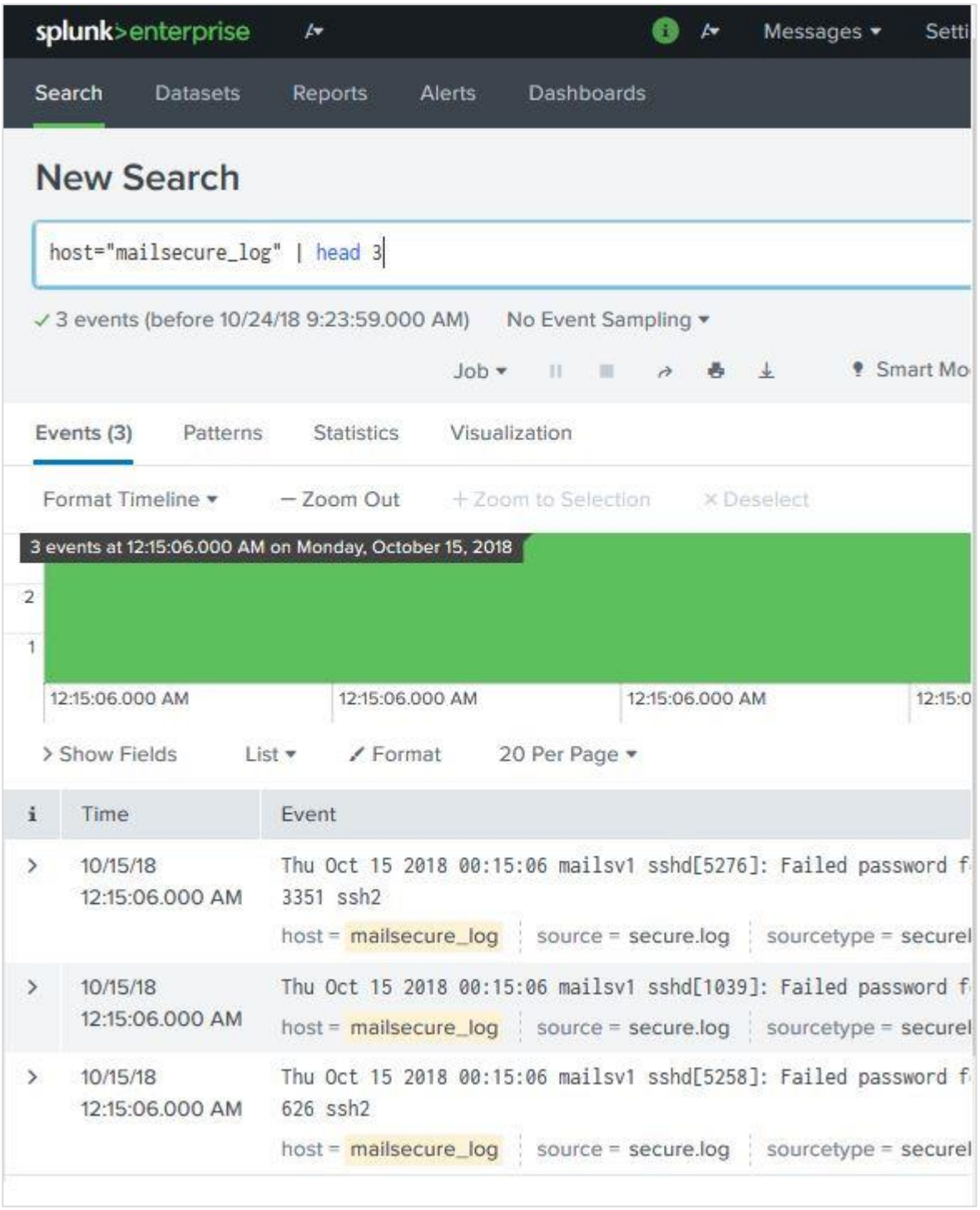
Search Terms

These are the terms you mention in the search bar to get specific records from the dataset which meet the search criteria. In the below example, we are searching for records which contain two highlighted terms.



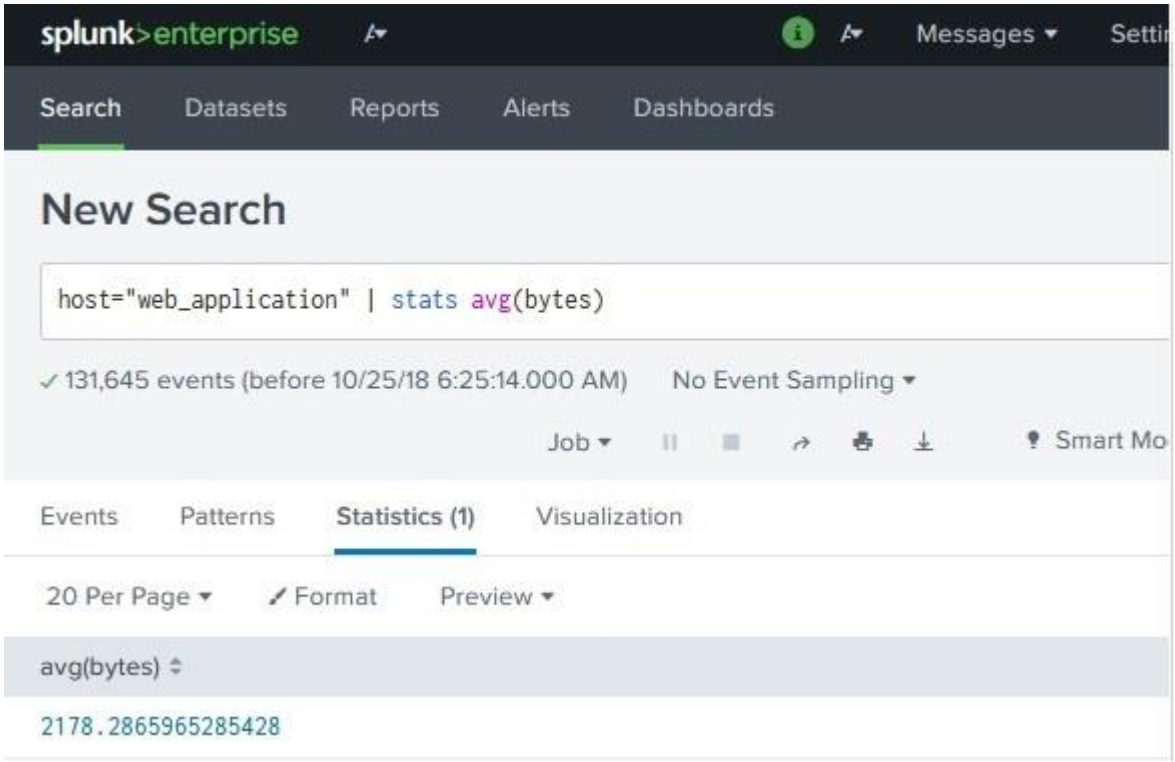
Commands

You can use many in-built commands that SPL provides to simplify the process of analysing the data in the result set. In the below example we use the head command to filter out only the top 3 results from a search operation.



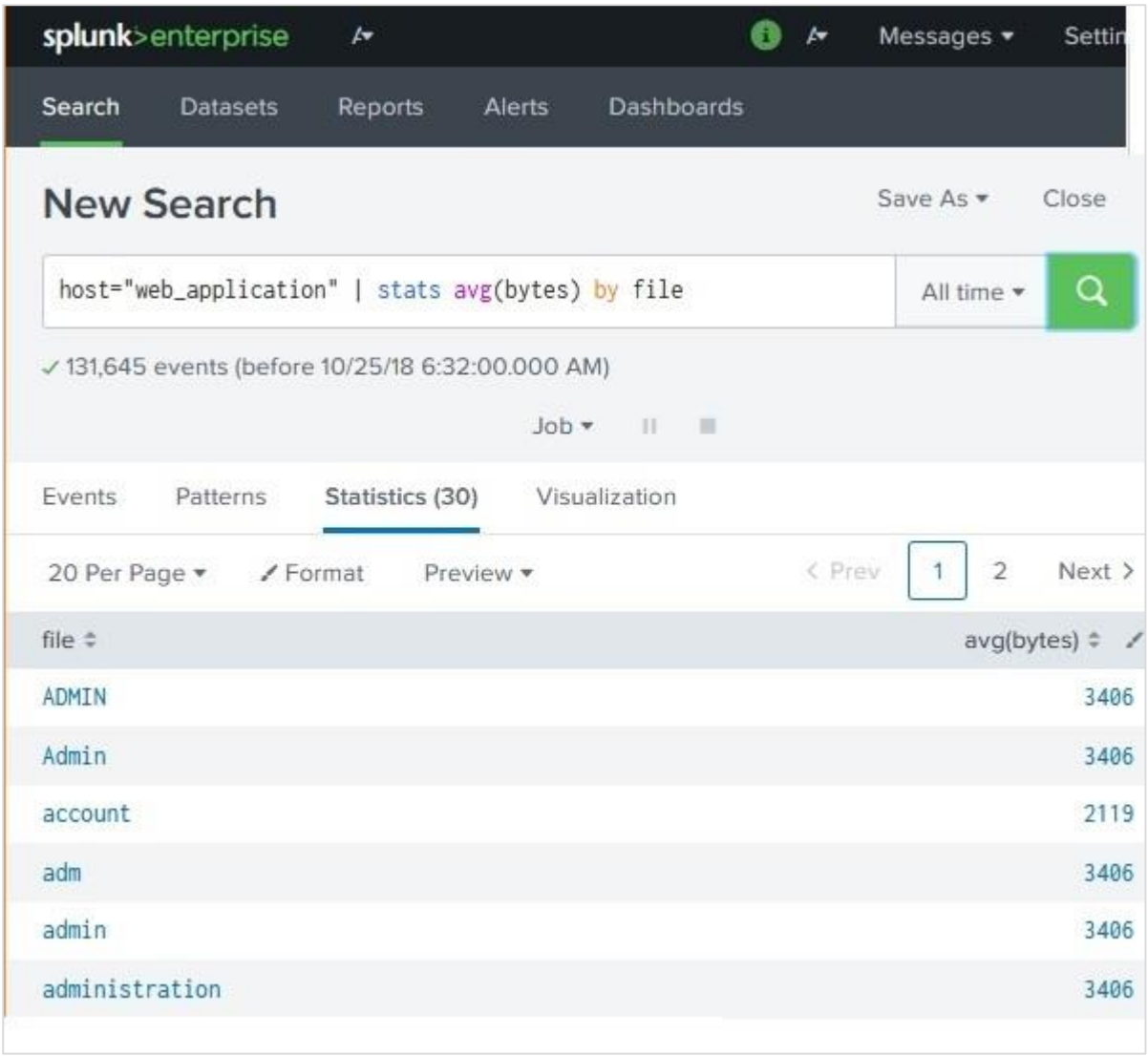
Functions

Along with commands, Splunk also provides many in-built functions which can take input from a field being analysed and give the output after applying the calculations on that field. In the below example, we use the **Stats avg()**function which calculates the average value of the numeric field being taken as input.



Clauses

When we want to get results grouped by some specific field or we want to rename a field in the output, we use the **group by** clause and the **as** clause respectively. In the below example, we get the average size of bytes of each file present in the **web_application** log. As you can see, the result shows the name of each file as well as the average bytes for each file.



11. Splunk – Search Optimization

Splunk already includes the optimization features, analyses and processes your searches for maximum efficiency. This efficiency is mainly achieved through the following two optimization goals:

- **Early Filtering**: These optimizations filter the results very early so that the amount of data getting processed is reduced as early as possible during the search process. This early filter avoids unnecessary lookup and evaluation calculations for events that are not part of final search results.
- **Parallel Processing**: The built-in optimizations can reorder search processing, so that as many commands as possible are run in parallel on the indexers before sending the search results to the search head for final processing.

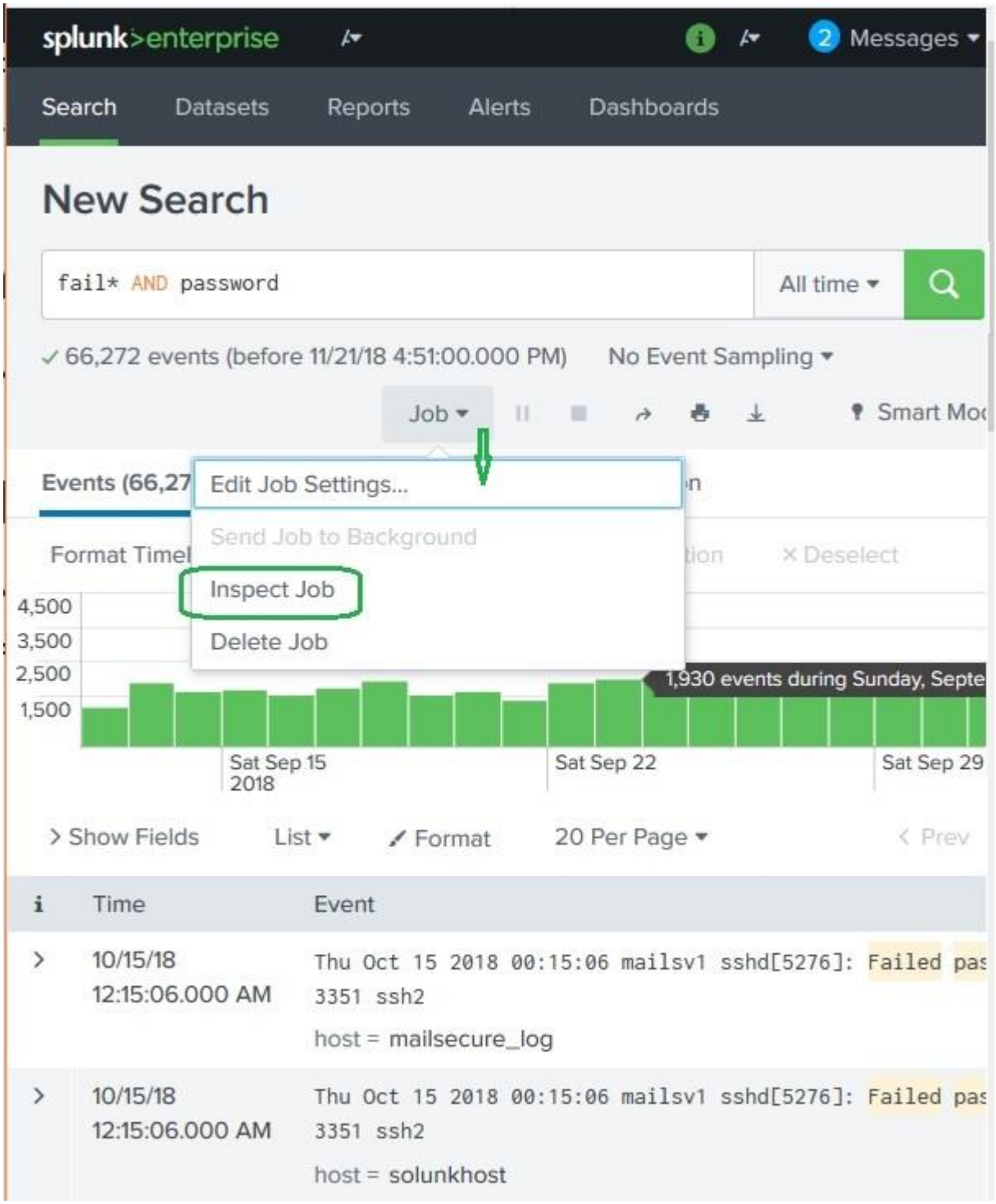
Analysing Search Optimisations

Splunk has given us tools to analyse how the search optimization works. These tools help us figure out how the filter conditions are used and what is the sequence of these optimisation steps. It also gives us the cost of the various steps involved in the search operations.

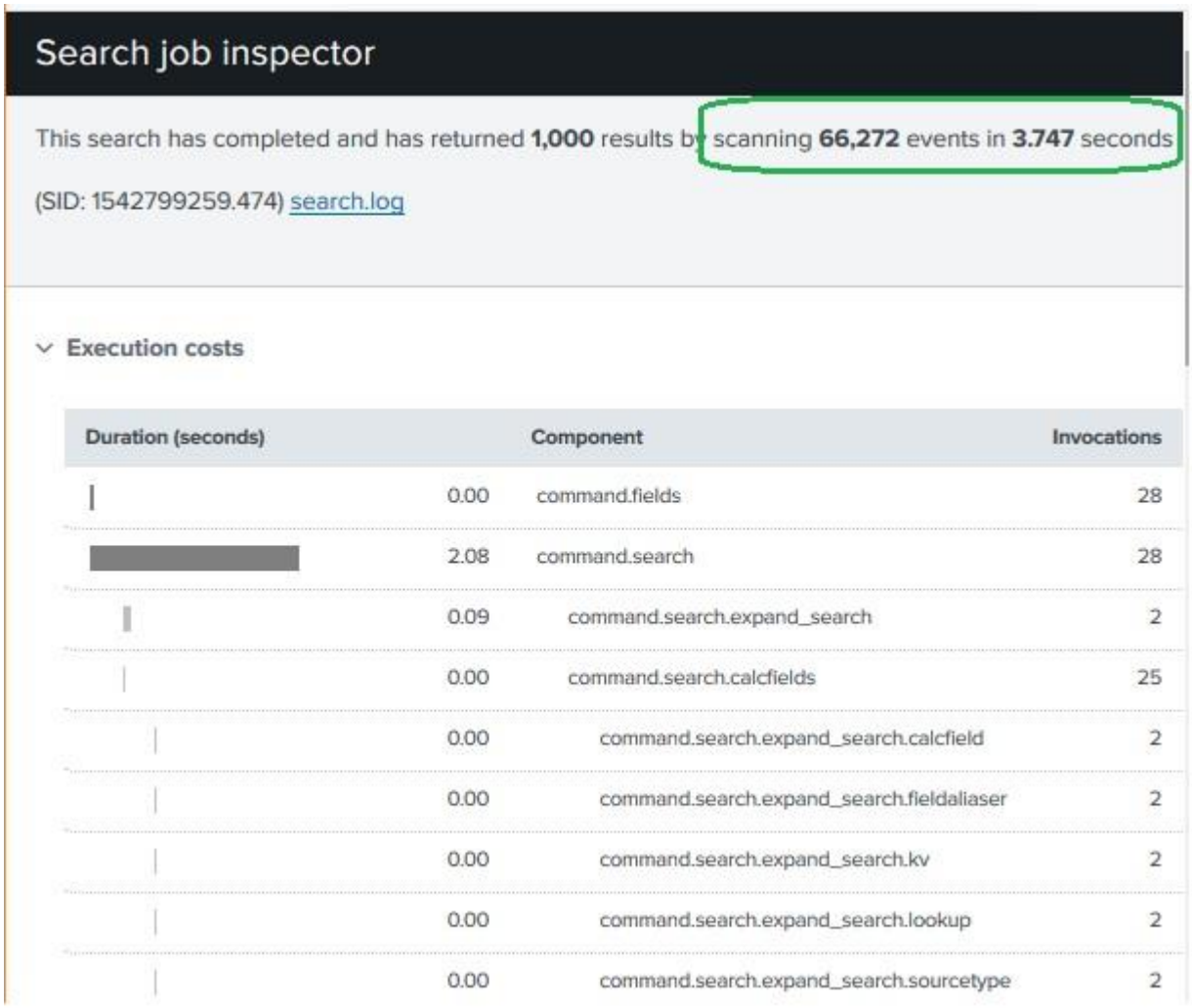
Example

Consider a search operation to find the events which contain the words: fail, failed or password. When we put this search query in the search box, the built-in optimizers act automatically to decide the path of the search. We can verify how long the search took to return a specific number of search results and if needed can go on to check each and every step of the optimization along with the cost associated with it.

We follow the path of **Search -> Job -> Inspect Job** to get these details as shown below:



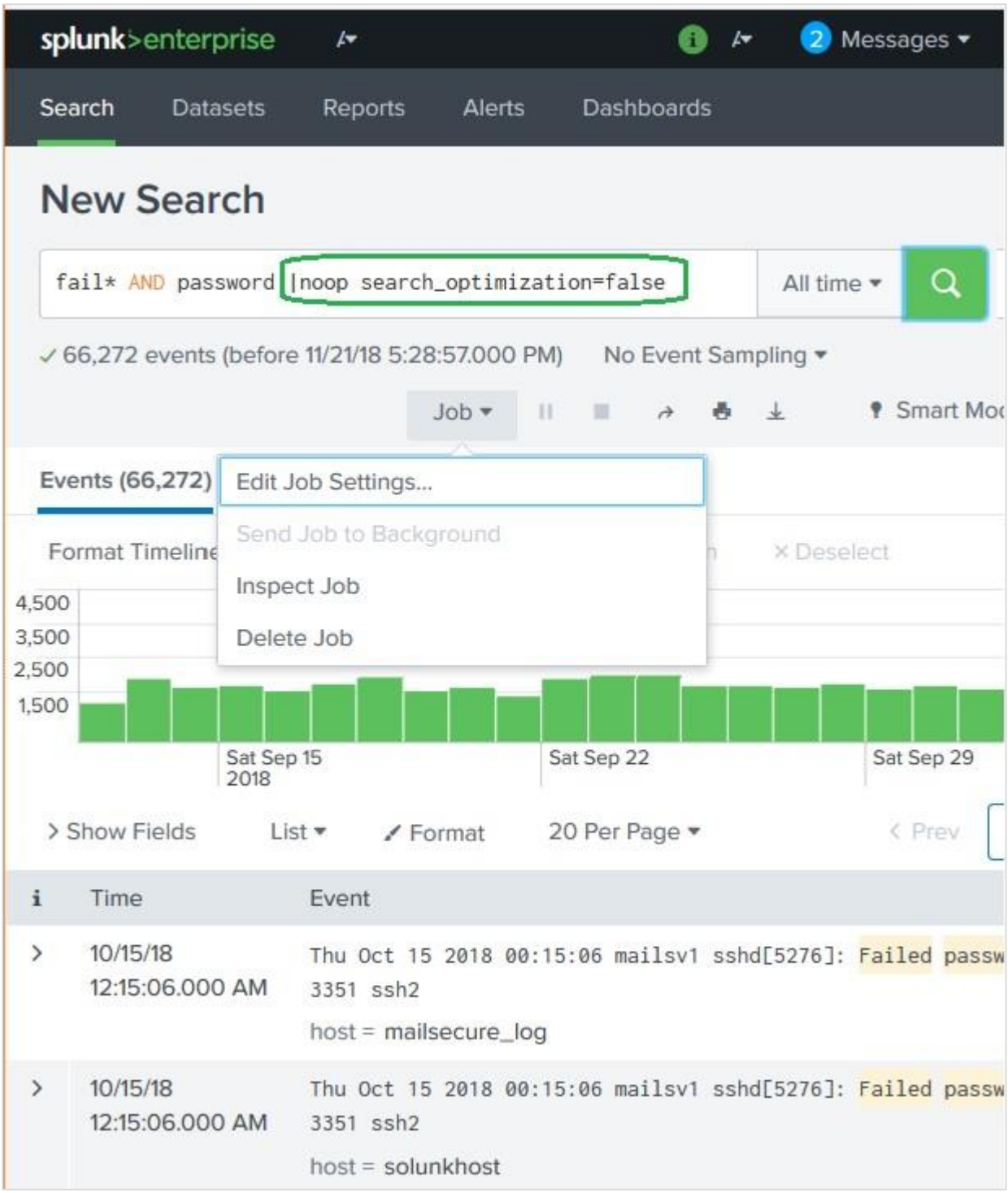
The next screen gives details of the optimization that has occurred for the above query. Here, we need to note the number of events and the time taken to return the result.



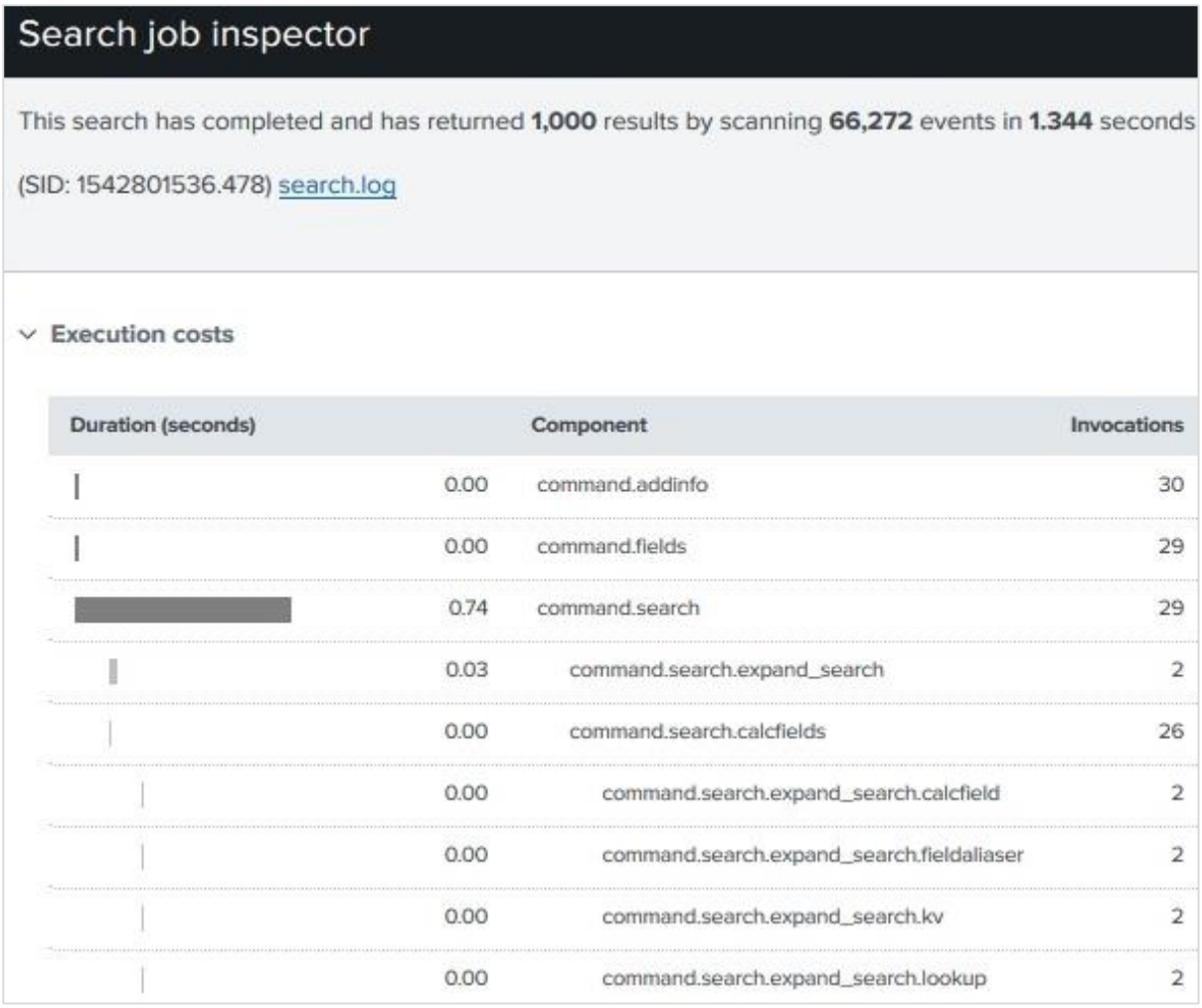
Turning Off Optimization

We can also turn off the in-built optimization and notice the difference in the time taken for the search result. The result may or may not be better than the in-built search. In case it is better, we may always choose this option of turning off the optimization for only this specific search.

In the below diagram, we use the No Optimization command presented as **noop** in the search query.



The next screen gives us the result of using no optimization. For this given query, the results come faster without using in-built optimizations.



Search job inspector

This search has completed and has returned **1,000** results by scanning **66,272** events in **1.344** seconds

(SID: 1542801536.478) search.log

Execution costs

Duration (seconds)	Component	Invocations
0.00	command.addinfo	30
0.00	command.fields	29
0.74	command.search	29
0.03	command.search.expand_search	2
0.00	command.search.calcfields	26
0.00	command.search.expand_search.calcfield	2
0.00	command.search.expand_search.fieldaliaser	2
0.00	command.search.expand_search.kv	2
0.00	command.search.expand_search.lookup	2

12. Splunk – Transforming Commands

These are the commands in Splunk which are used to transform the result of a search into such data structures which will be useful in representing the statistics and data visualizations.

Examples of Transforming Commands

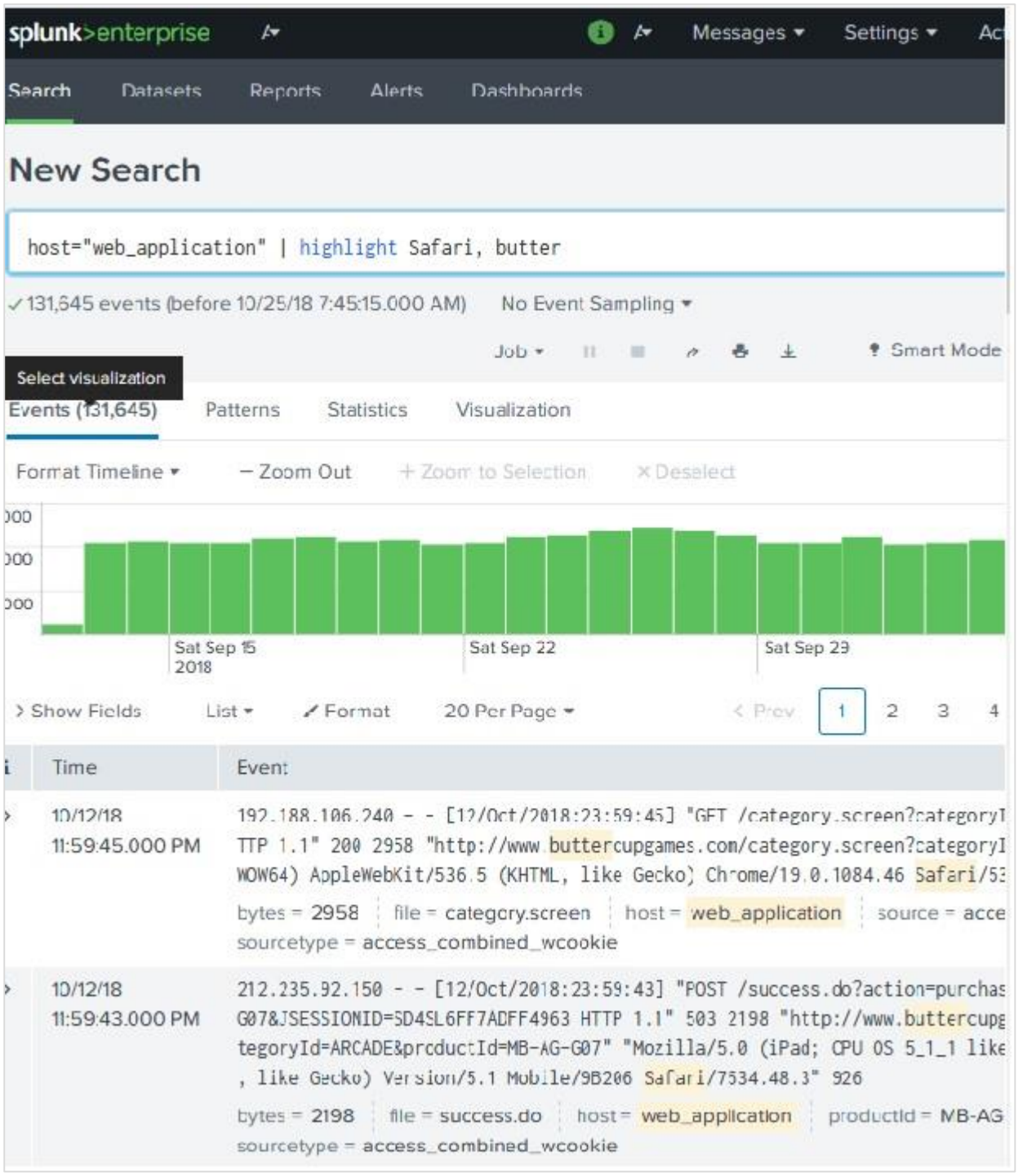
Following are some of the examples of transforming commands:

- **Highlight** – To highlight the specific terms in a result.
- **Chart** – To create a chart out of the search result.
- **Stats** – To create statistical summaries from the search result.

Highlight

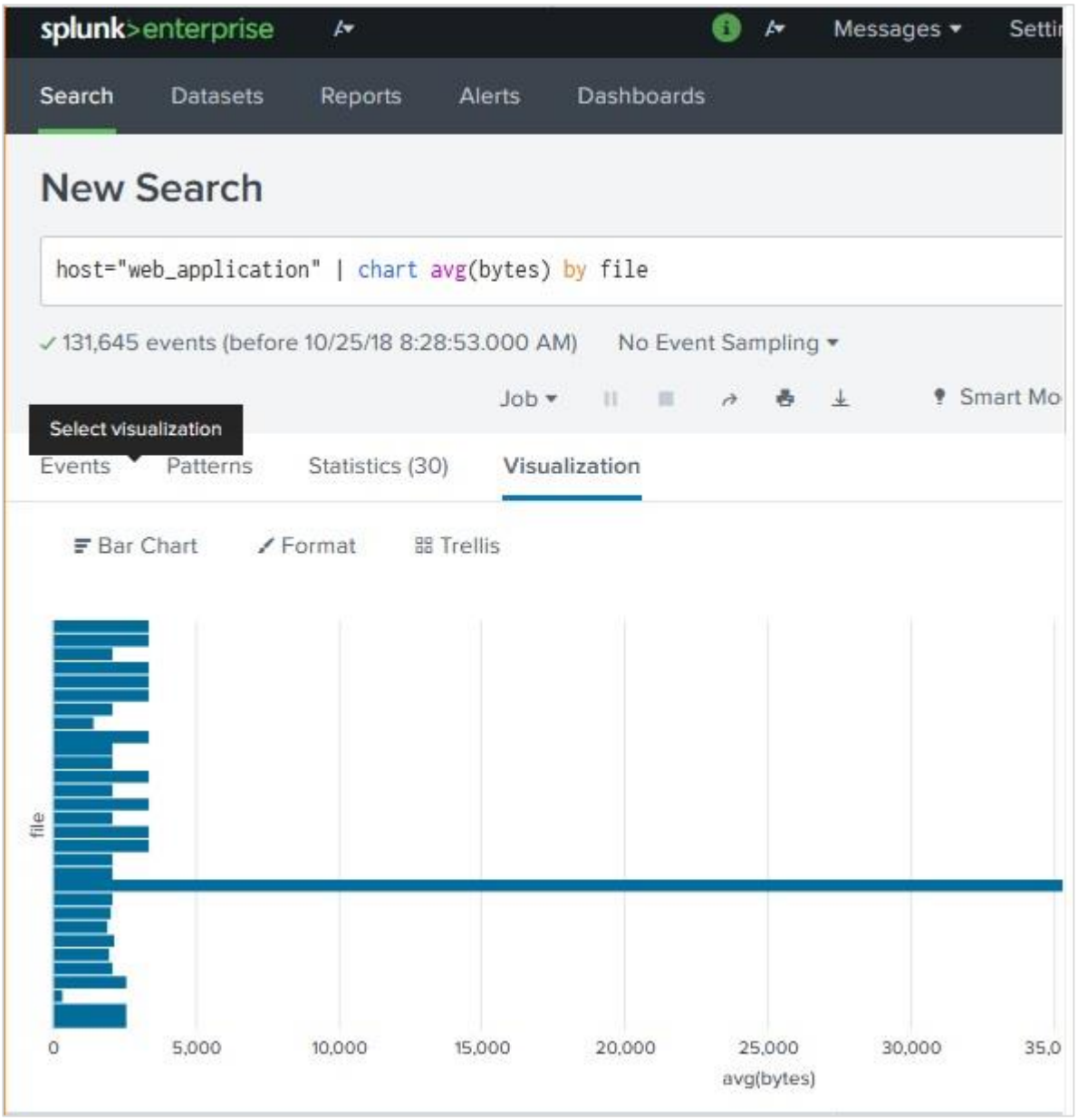
This command is used to **highlight specific terms in the search result set**. It is used by supplying the search terms as arguments to the highlight function. Multiple search terms are supplied by separating them with comma.

In the below example, we search for the terms, **safari** and **butter** in the result set.



Chart

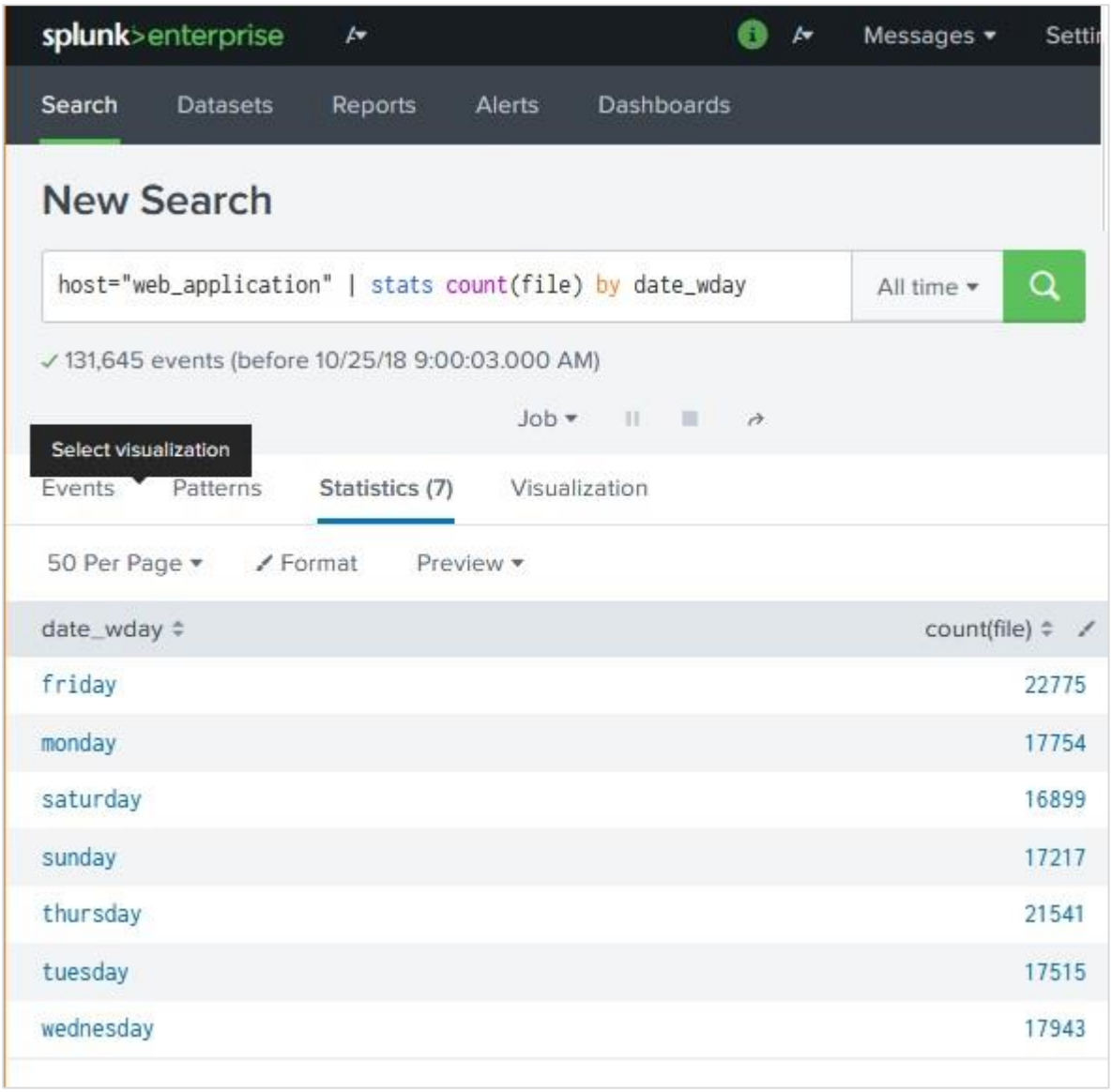
The **chart** command is a transforming command that returns your results in a table format. The results can then be used to display the data as a chart, such as column, line, area, etc. In the below example, we create a horizontal bar chart by plotting the average size of bytes for each file type.



Stats

The Stats command transforms the search result data set into various statistical representations depending on the types of arguments we supply for this command.

In the below example, we use the stats command with count function which is then grouped by another field. Here, we are counting the number of file names created on each week day. The result of the search string come out in a tabular from with rows created for each day.



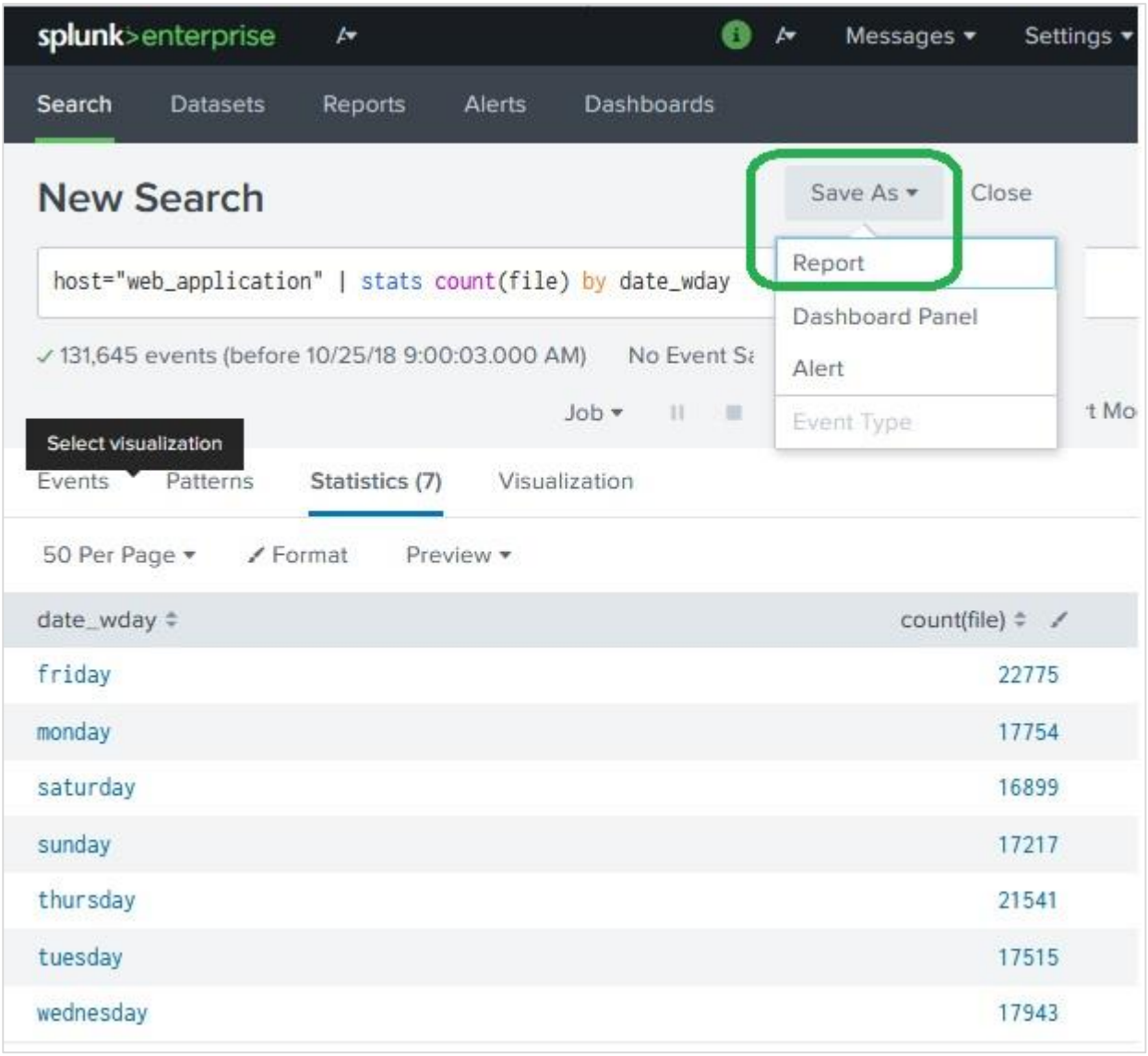
13. Splunk – Reports

Splunk reports are results saved from a search action which can show statistics and visualizations of events. Reports can be run anytime, and they fetch fresh results each time they are run. The reports can be shared with other users and can be added to dashboards. More sophisticated reports can allow a drill down function to see underlying events which create the final statistics.

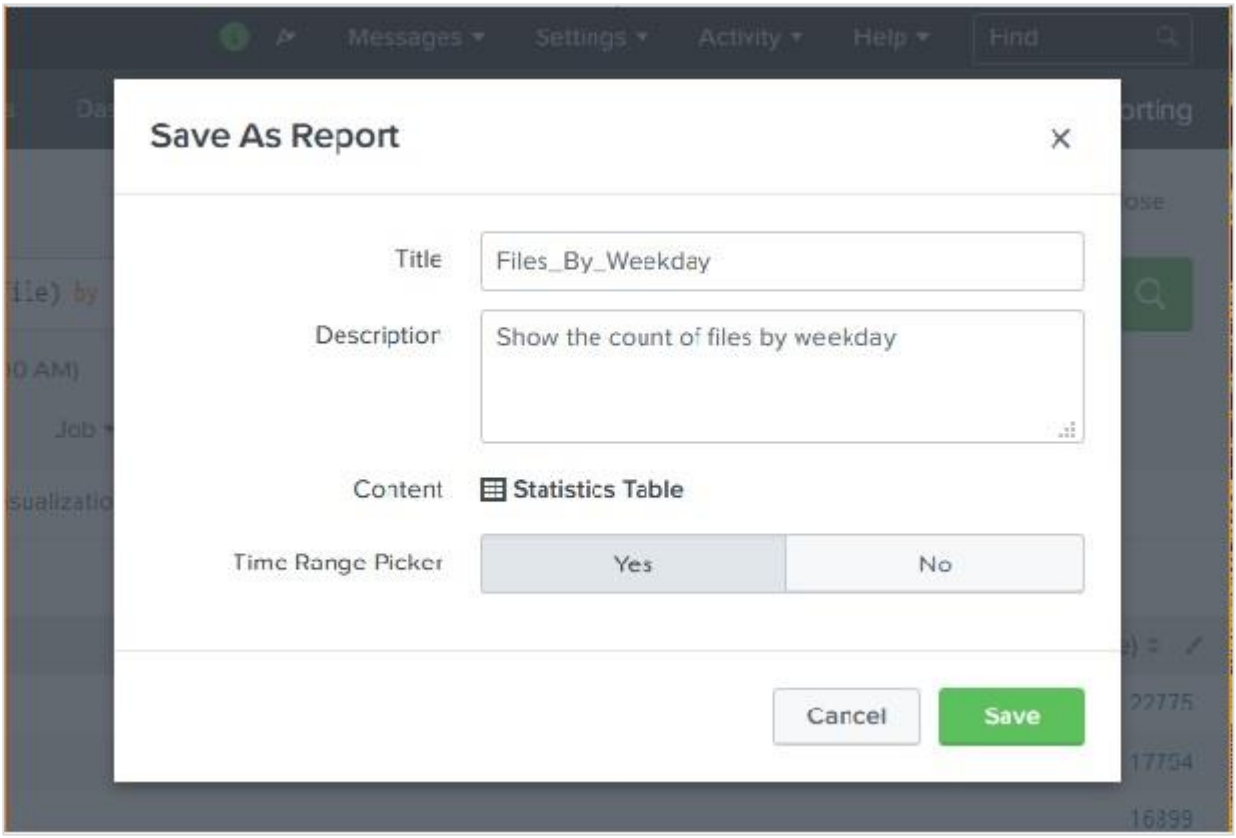
In this chapter, we will see how to create and edit a sample report.

Report Creation

Report creation is a straight forward process where we use the **Save As** option to save the result of a search operation choosing the Reports option. The below diagram shows the **Save As** option.

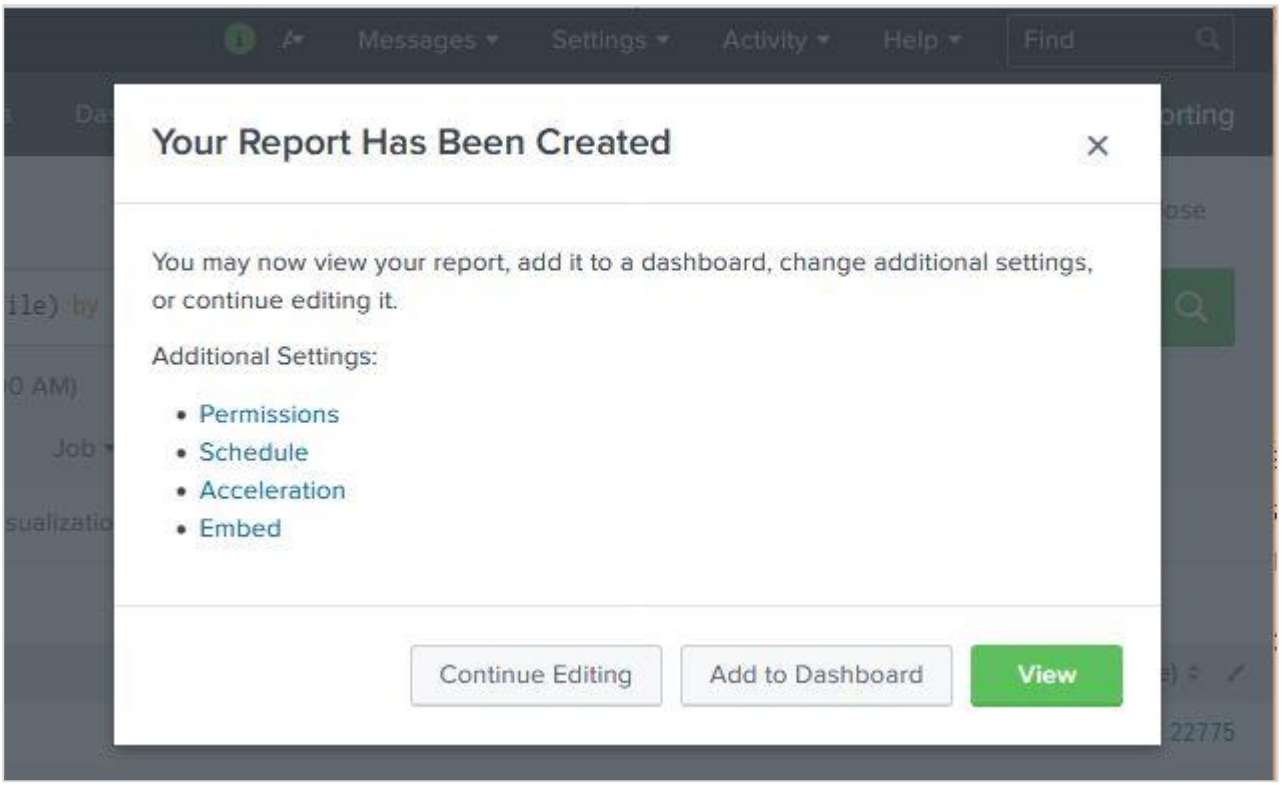


By clicking on the Reports option from the dropdown, we get the next window which asks for additional inputs like the name of the report, the description and choosing the time picker. If we choose the time picker, it allows the time range to be adjusted when we run the report. Below diagrams show how we fill the required details and then click save.

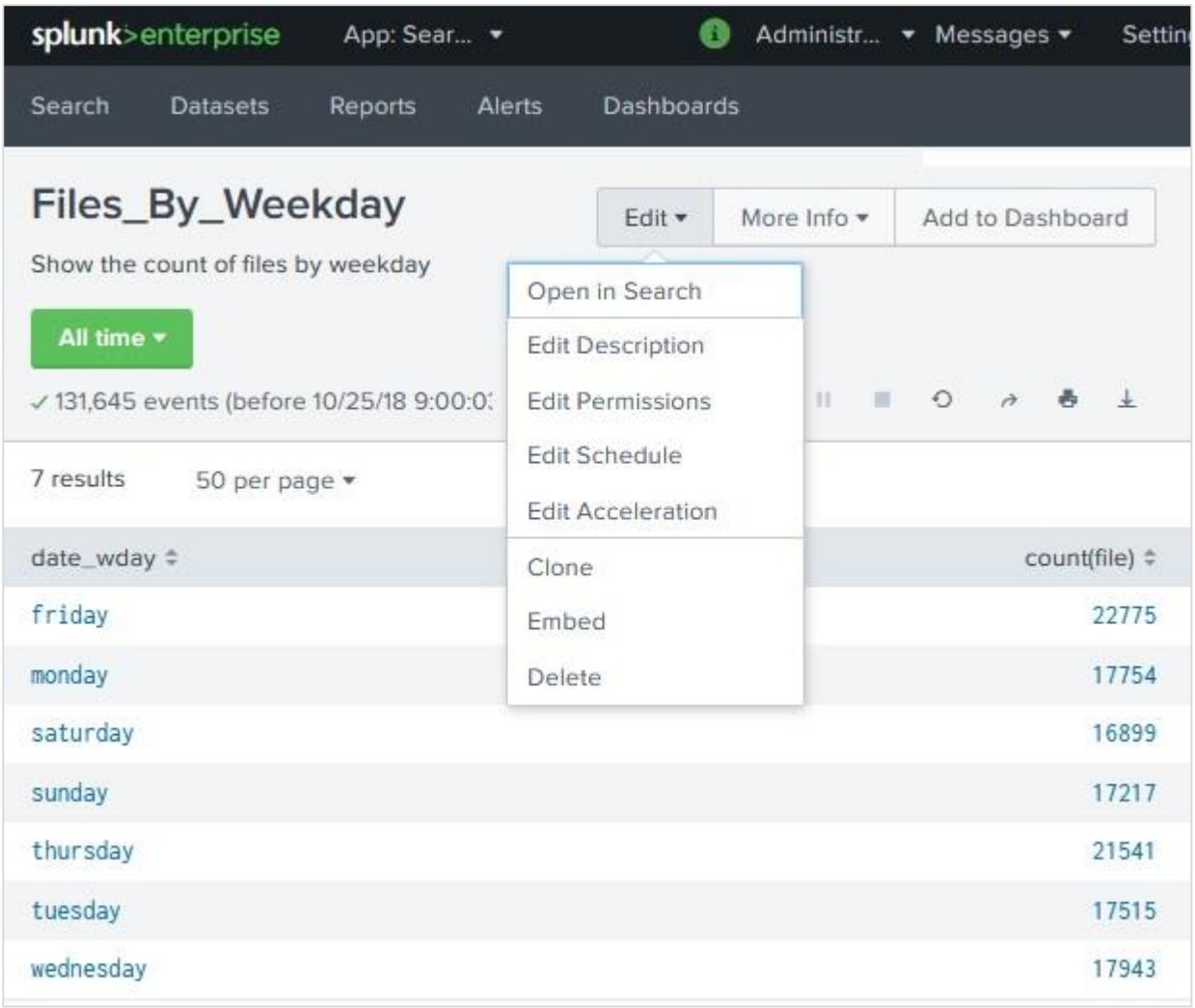


Report Configuration

After clicking save to create the report in the above step, we get the next screen asking for configuring the report as shown below. Here, we can configure the permissions, scheduling the report, etc. We also get an option to go to the next step and add the report to a dashboard.

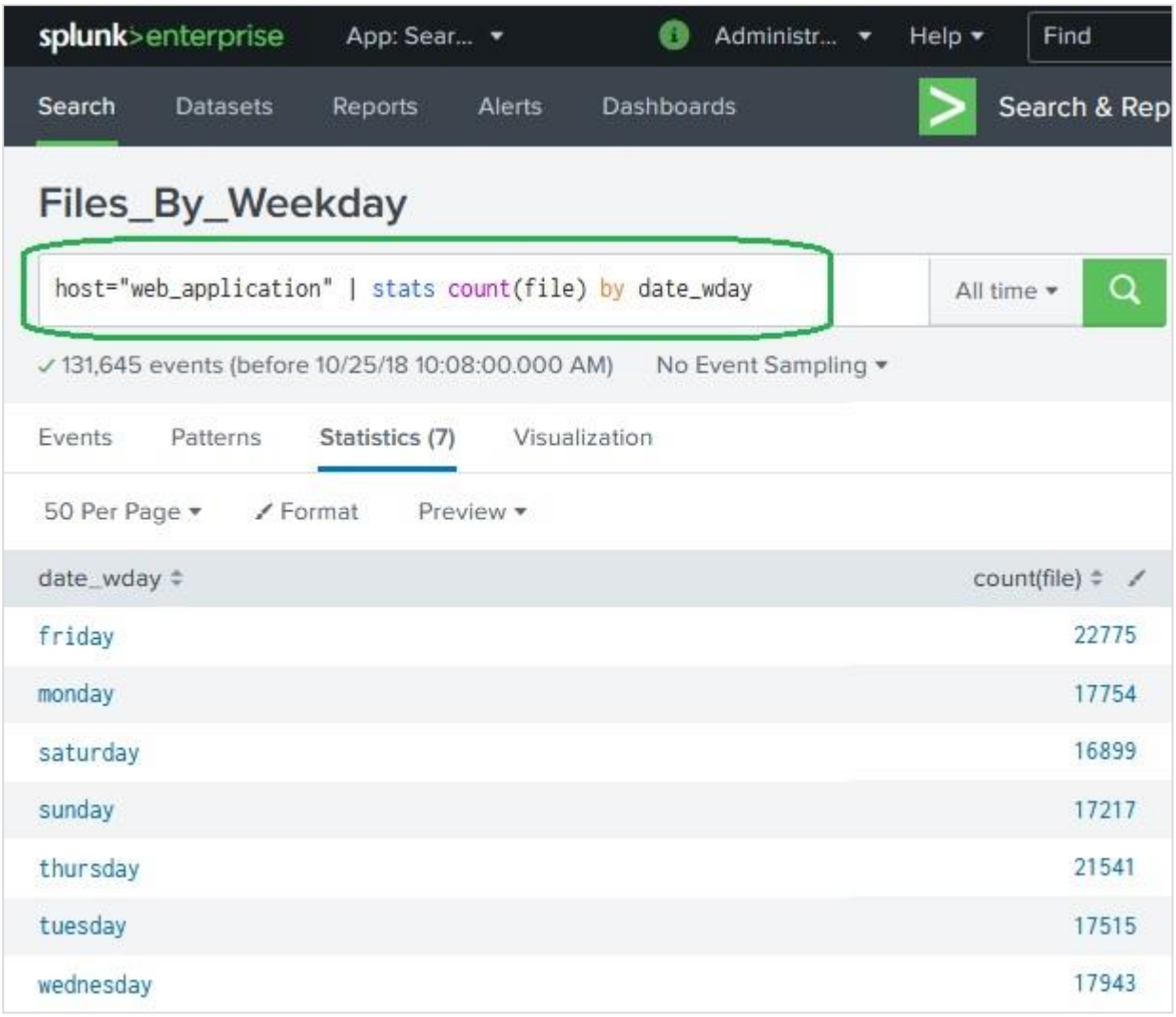


If we click on **View** in the above step, we can see the report. We also get configuration options after the report is created.



Modifying Report Search Option

While we can edit the permissions, schedule, etc., sometimes we need to modify the original search string. This can be done by choosing the **Open in Search** option as given in the above image. That will open the original search option again which we can be edited to a new search. Refer to the below image:



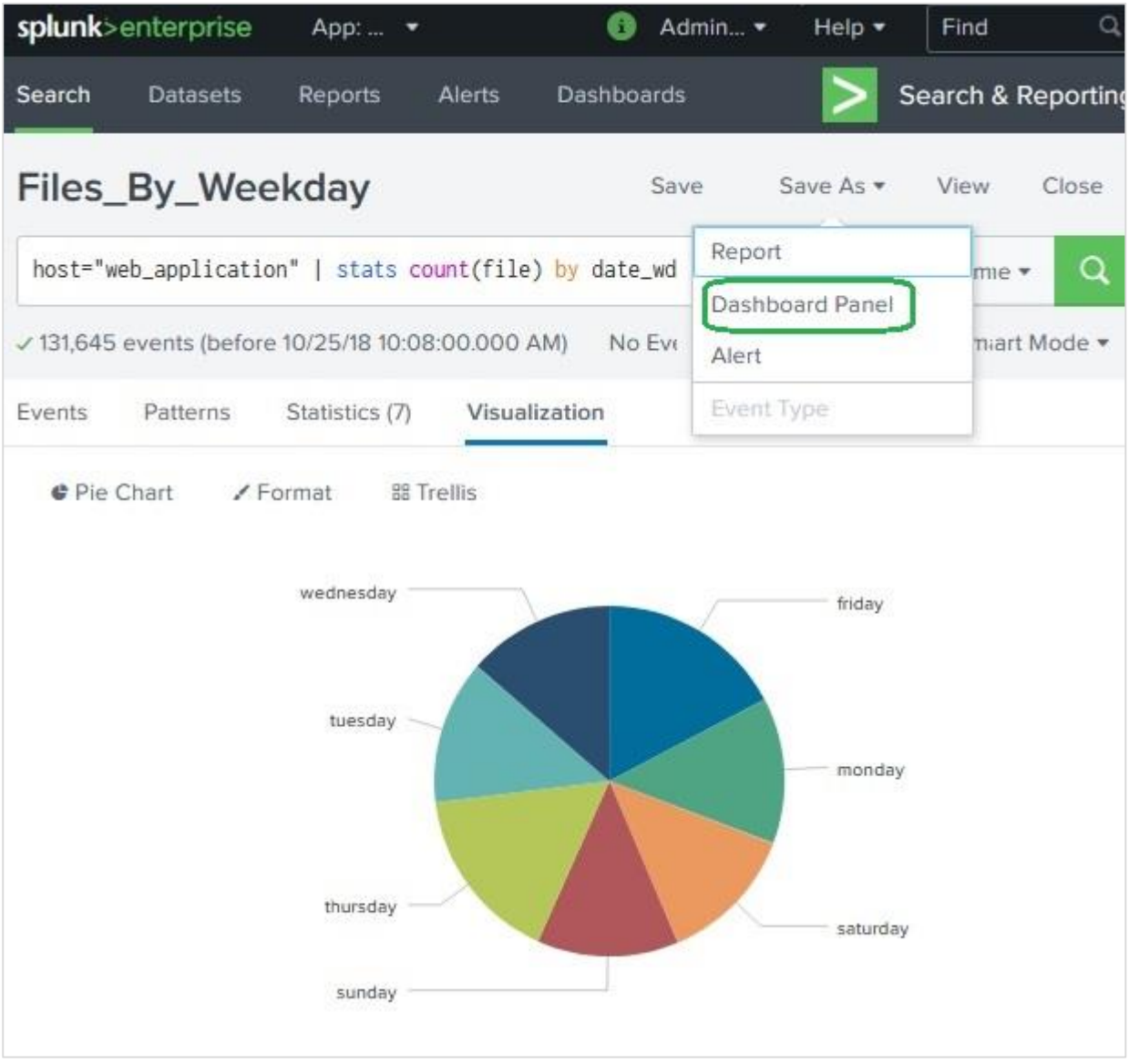
14. Splunk – Dashboards

A dashboard is used to represent tables or charts which are related to some business meaning. It is done through panels. The panels in a dashboard hold the chart or summarized data in a visually appealing manner. We can add multiple panels, and hence multiple reports and charts to the same dashboard.

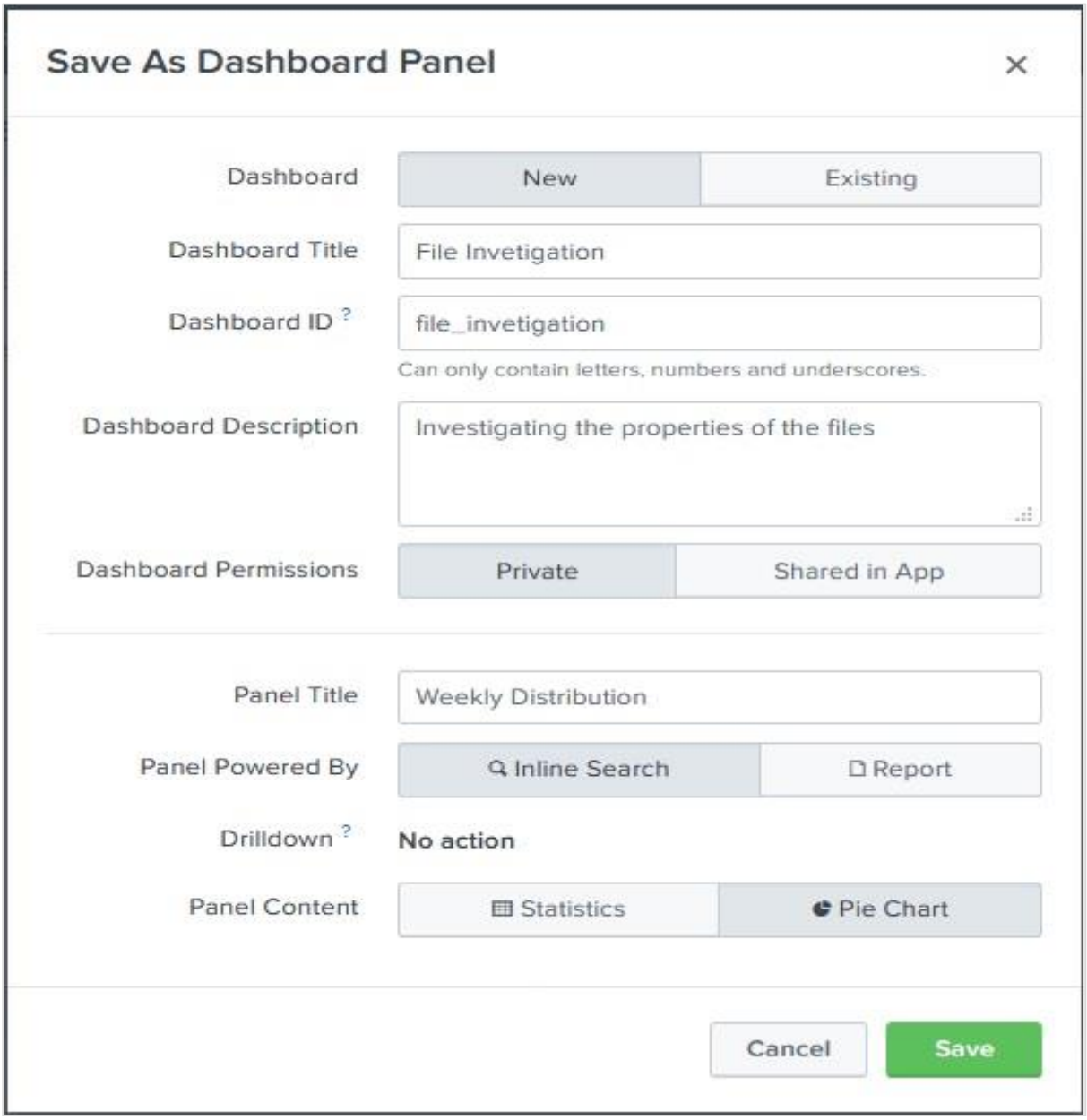
Creating Dashboard

We will continue with the search query from the previous chapter which shows the count of files by week days.

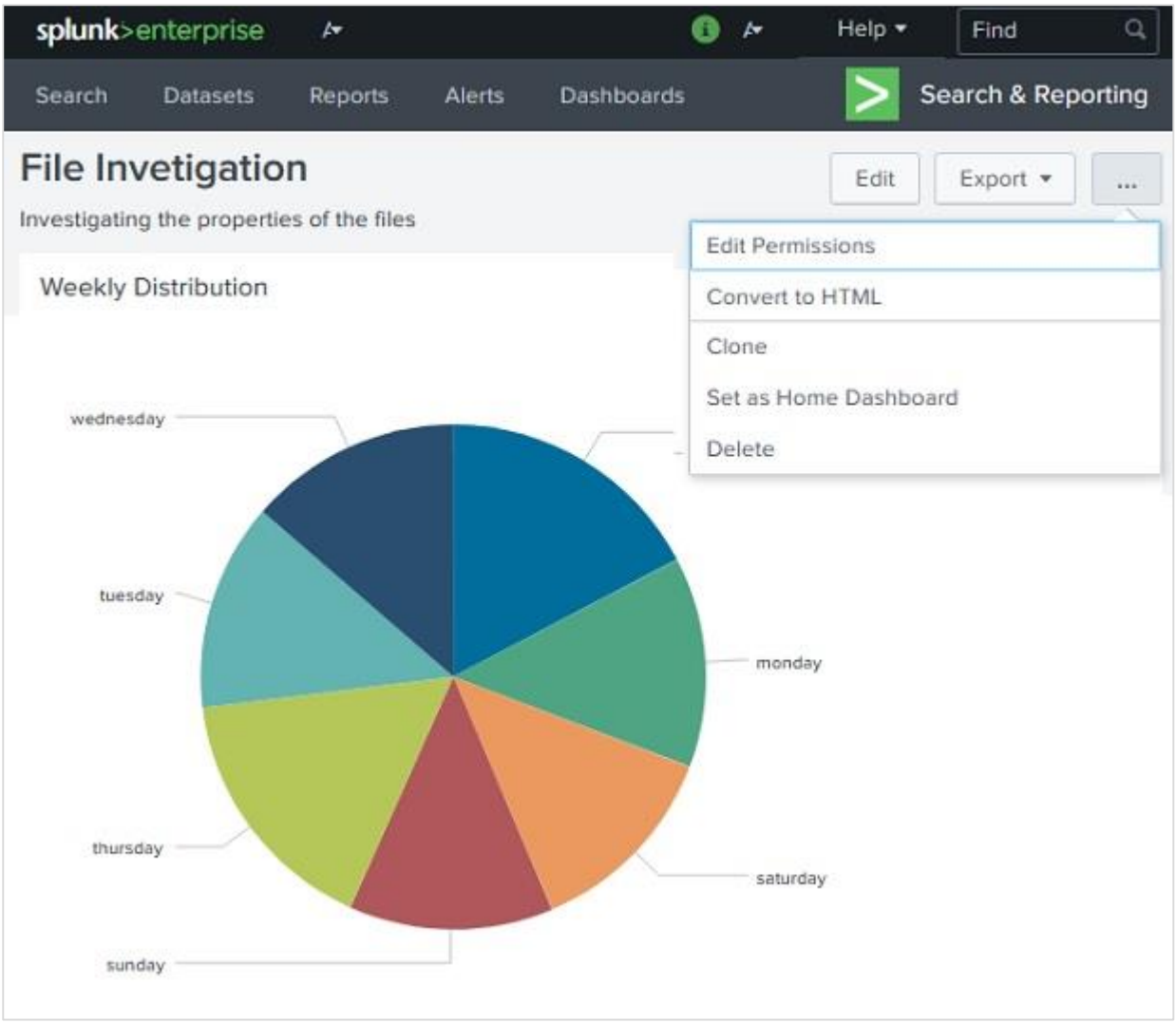
We choose the Visualization tab to see the result as a pie chart. To put the chart on a dashboard, we can choose the option **Save As -> Dashboard Panel** as shown below.



The next screen will ask for fillings the details of the dashboard and the panel in it. We fill the screen with details as shown below.

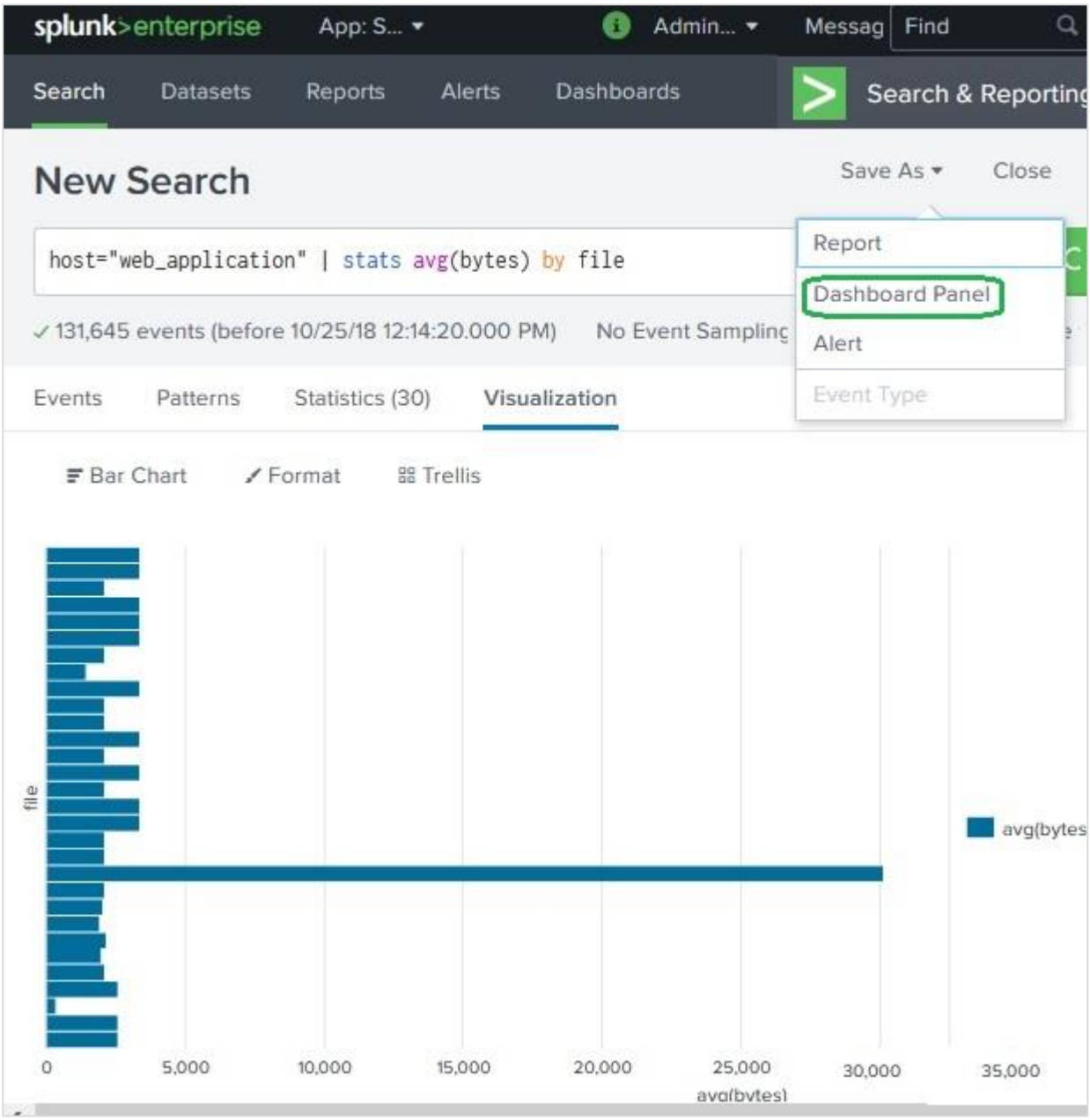


On clicking on Save button, the next screen gives an option to view dashboard. On choosing to view dashboard, we get the following output where we can see the dashboard and options to edit, export or delete.

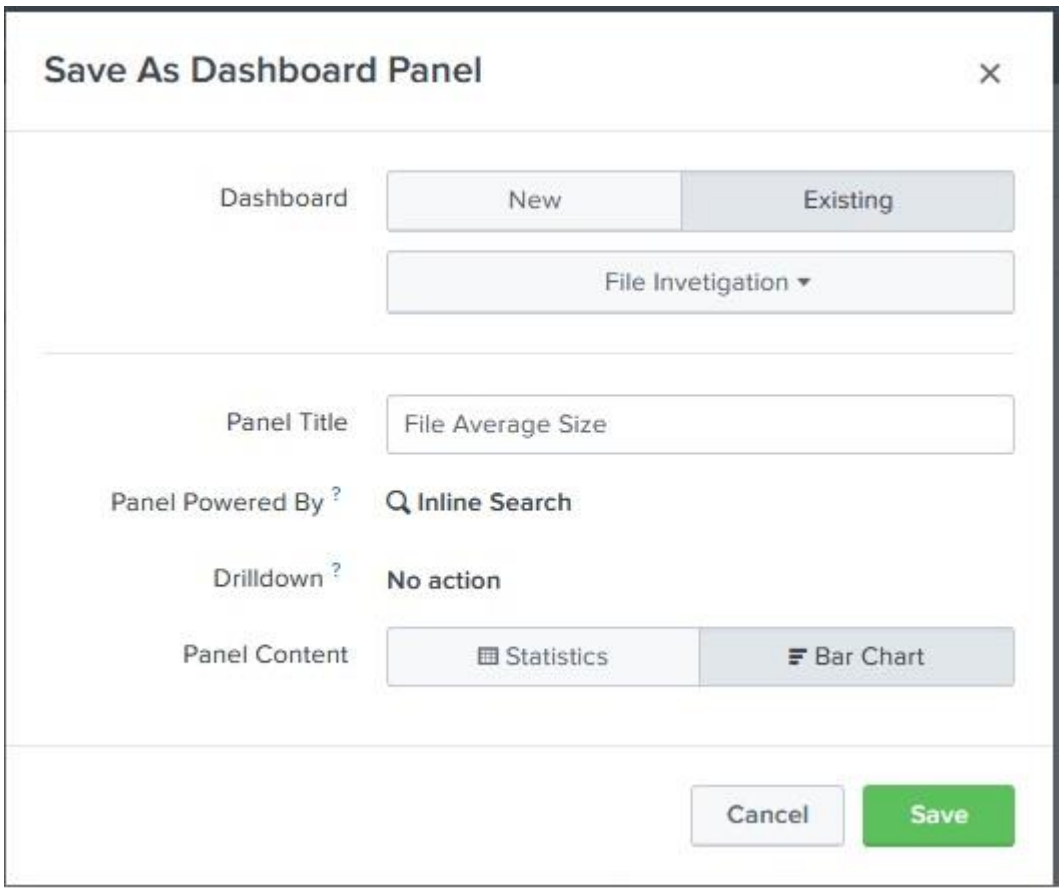


Adding Panel to Dashboard

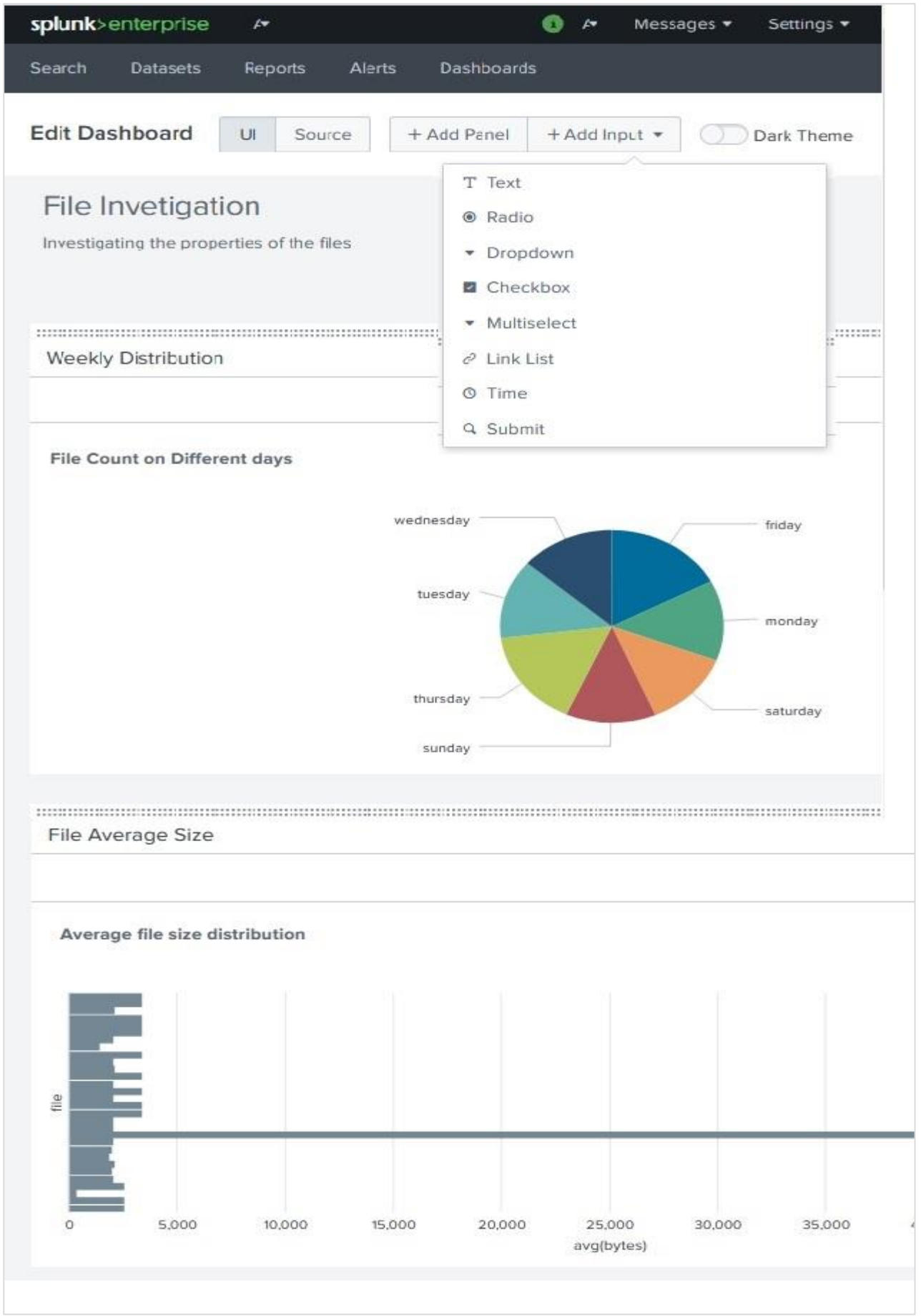
We can add a second chart to the dashboard by adding a new panel containing the chart. Below is the bar chart and its query which we are going to add to the above dashboard.



Next, we fill up the details for the second chart and click **Save** as shown in the below image:



Finally, we get the dashboard which contains both the charts in two different panels. As you can see in the image below, we can edit the dashboard to add more panels and you can add more input elements: Text, Radio and Dropdown buttons to create more sophisticated dashboards.

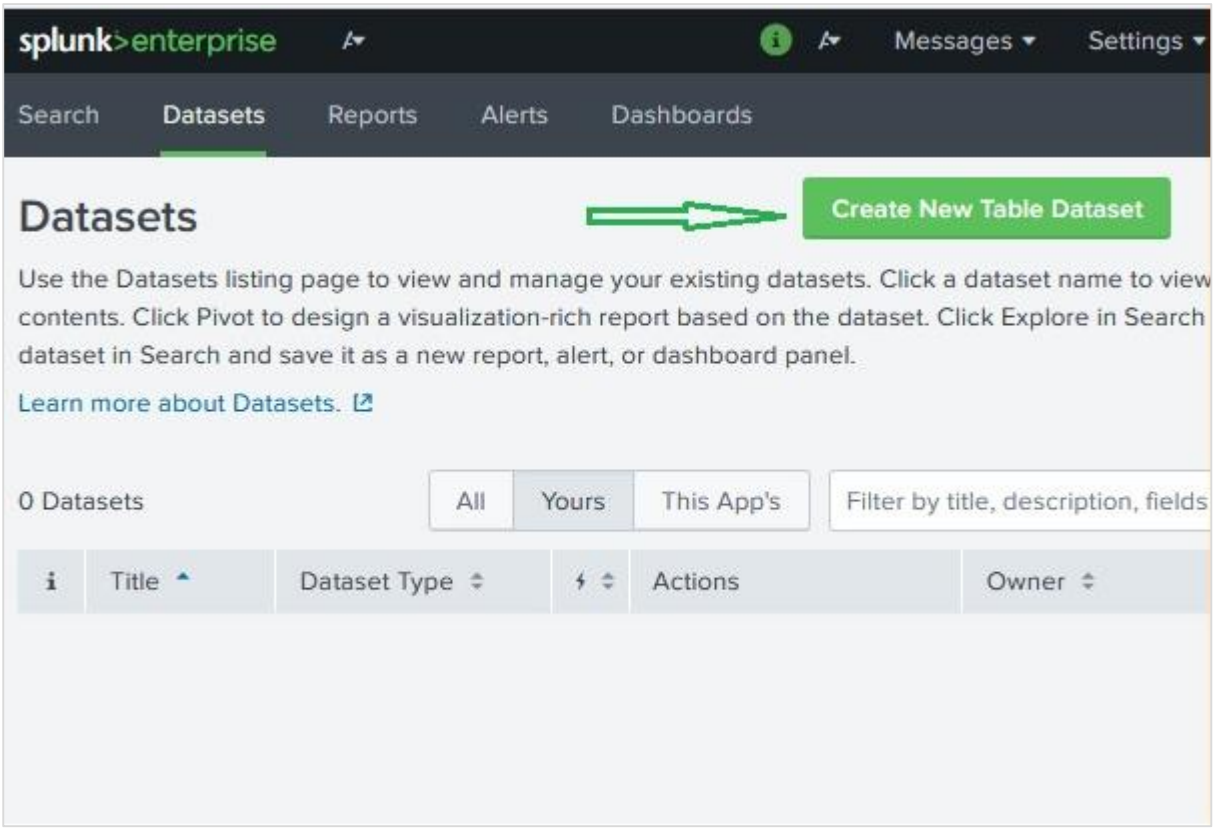


15. Splunk – Pivot and Datasets

Splunk can ingest different types of data sources and build tables which are similar to relational tables. These are called **table dataset** or just **tables**. They provide easy ways to analyse and filter the data and lookups, etc. These table data sets are also used in creating pivot analysis which we learn in this chapter.

Creating a Dataset

We use a Splunk Add-on named **Splunk Datasets Add-on** to create and manage the datasets. It can be downloaded from the Splunk website, https://splunkbase.splunk.com/app/3245/#/details. It has to be installed by following the instructions given in the details tab in this link. On successful installation, we see a button named **Create New Table Dataset**.



Selecting a Dataset

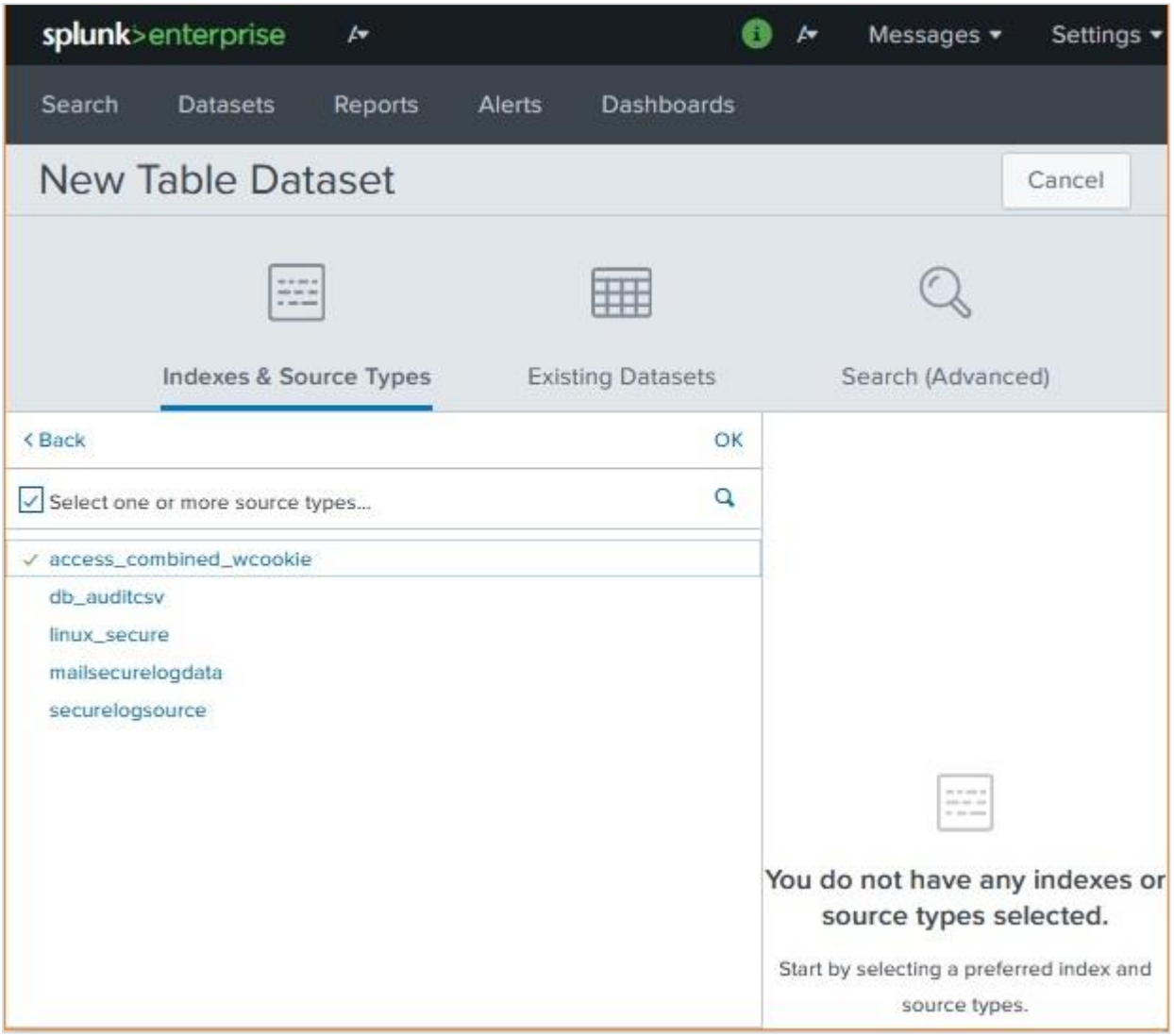
Next, we click on the **Create New Table Dataset** button and it gives us the option to choose from the below three options.

- **Indexes and Source Types** – Choose from an existing index or source type which are already added to Splunk through Add Data app.

- **Existing Datasets** – You might have already created some dataset previously which you want to modify by creating a new dataset from it.

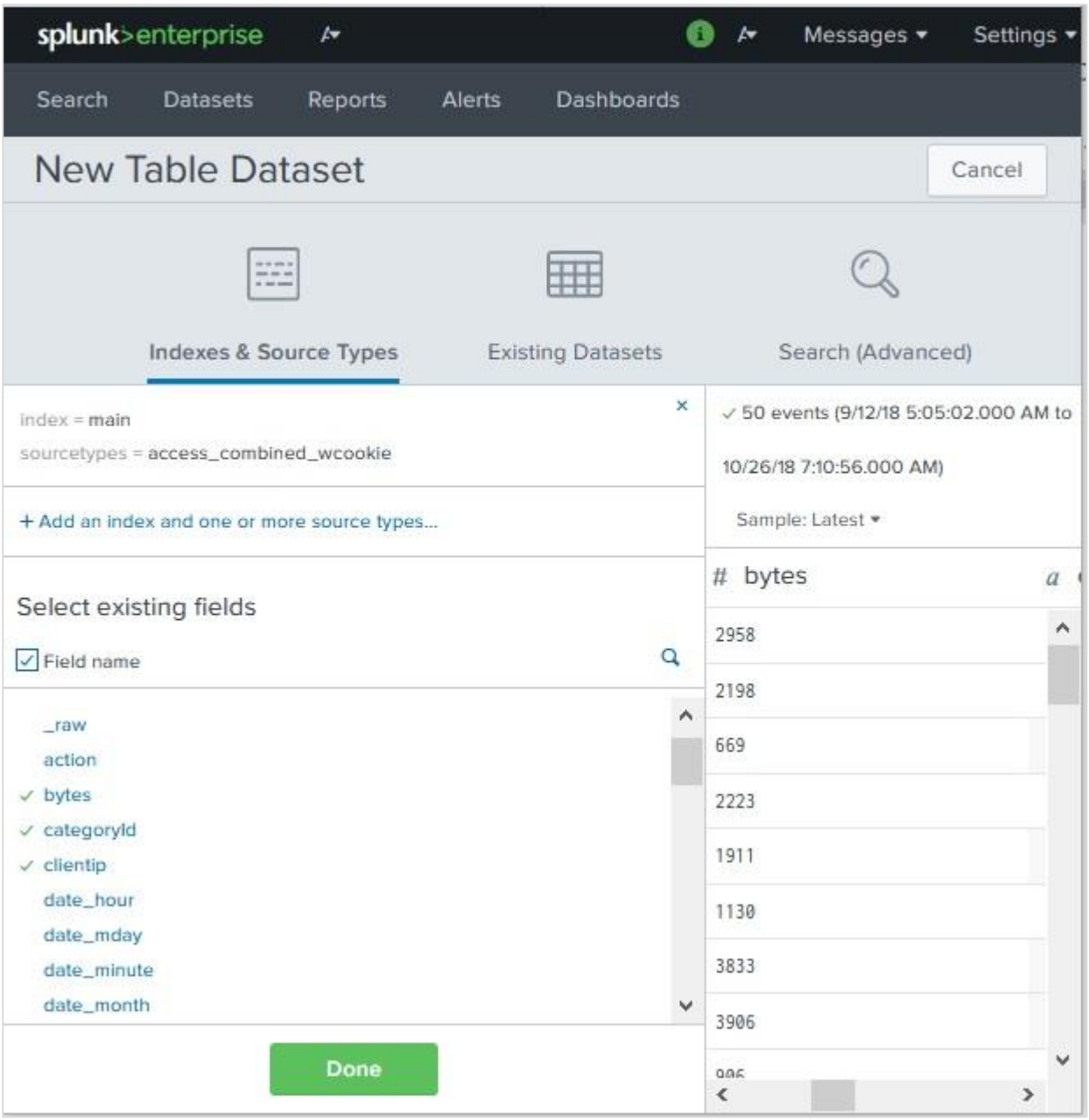
- **Search** – Write a search query and the result can be used to create a new dataset.

In our example, we choose an index to be our source of data set as shown in the image below:

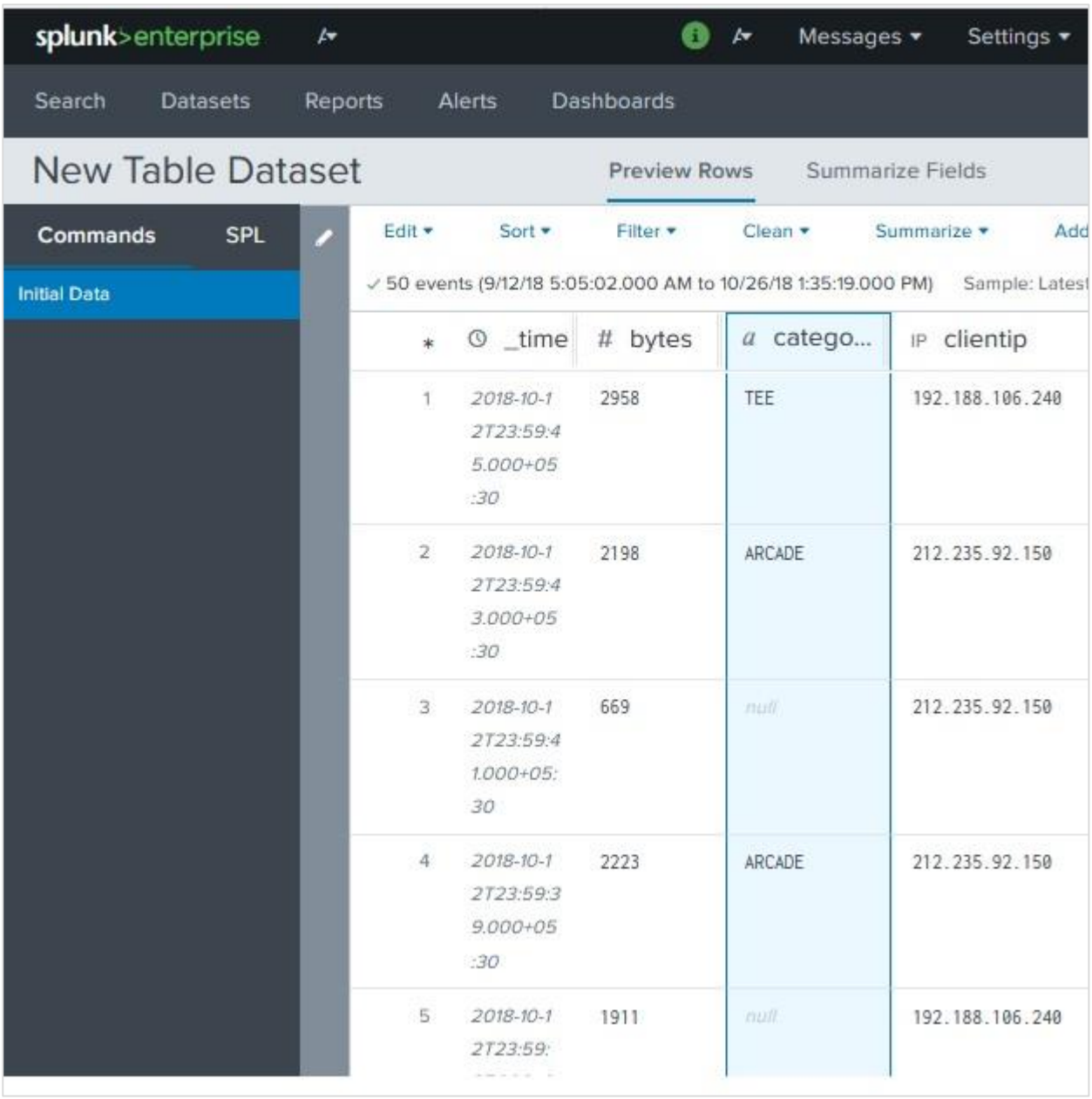


Choosing Dataset Fields

On clicking OK in the above screen, we are presented with an option to choose the various fields we want to finally get into the Table Dataset. The _time field is selected by default and this field cannot be dropped. We choose the fields: **bytes**, **categoryID**, **clientIP** and **files**.



On clicking done in the above screen, we get the final dataset table with all the selected fields, as seen below. Here the dataset has become similar to a relational table. We save the dataset with **save as** option available in the top right corner.

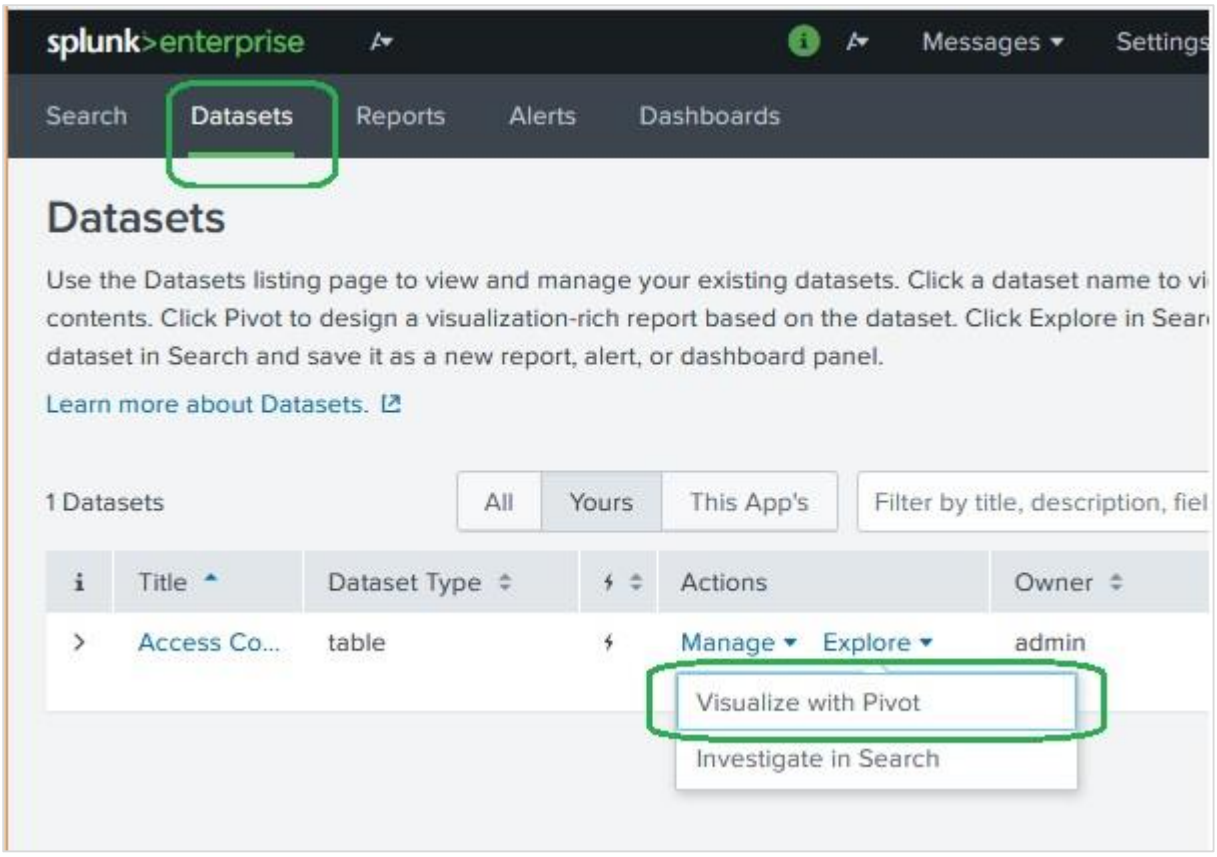


Creating Pivot

We use the above dataset to create a pivot report. The pivot report reflects aggregation of values of one column with respect to the values in another column. In other words, one columns values are made into rows and another columns values are made into rows.

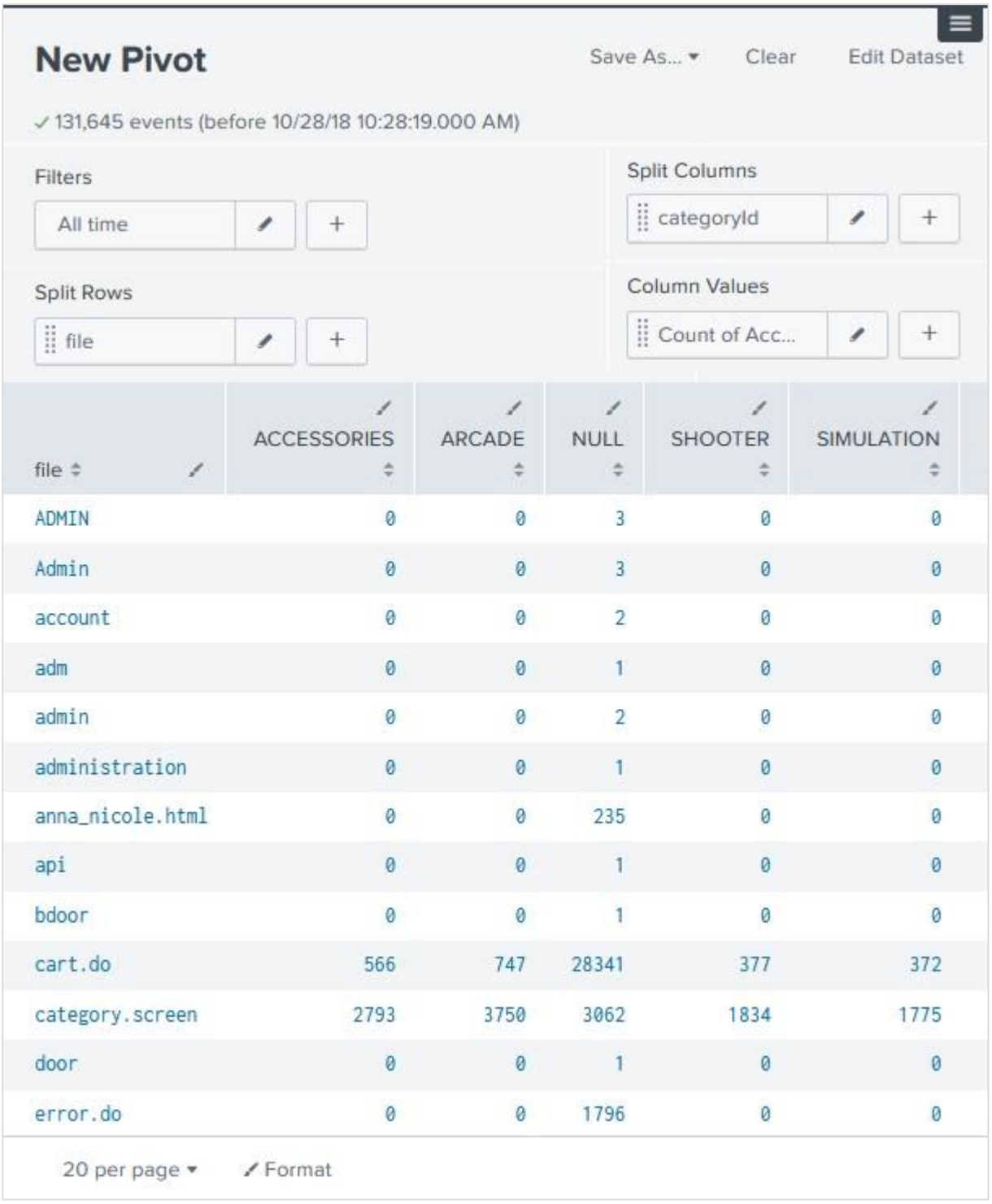
Choose Dataset Action

To achieve this, we first select the dataset using the dataset tab and then choose the option **Visualize with Pivot** from the Actions column for that data set.



Choose the Pivot Fields

Next, we choose the appropriate fields for creating the pivot table. We choose category ID in the **split columns** option as this is the field whose values should appear as different columns in the report. Then we choose File in the **Split Rows** option as this is the field whose values should be presented in rows. The result shows count of each categoryid values for each value in the file field.



file	ACCESSORIES	ARCADE	NULL	SHOOTER	SIMULATION
ADMIN	0	0	3	0	0
Admin	0	0	3	0	0
account	0	0	2	0	0
adm	0	0	1	0	0
admin	0	0	2	0	0
administration	0	0	1	0	0
anna_nicole.html	0	0	235	0	0
api	0	0	1	0	0
bdoor	0	0	1	0	0
cart.do	566	747	28341	377	372
category.screen	2793	3750	3062	1834	1775
door	0	0	1	0	0
error.do	0	0	1796	0	0

Next, we can save the pivot table as a Report or a panel in an existing dashboard for future reference.

16. Splunk – Lookups

In the result of a search query, we sometimes get values which may not clearly convey the meaning of the field. For example, we may get a field which lists the value of product id as a numeric result. These numbers will not give us any idea of what kind of product it is. But if we list the product name along with the product id, that gives us a good report where we understand the meaning of the search result.

Such linking of values of one field to a field with same name in another dataset using equal values from both the data sets is called a lookup process. The advantage is, we retrieve the related values from two different data sets.

Steps to Create and Use Lookup File

In order to successfully create a lookup field in a dataset, we need to follow the below steps:

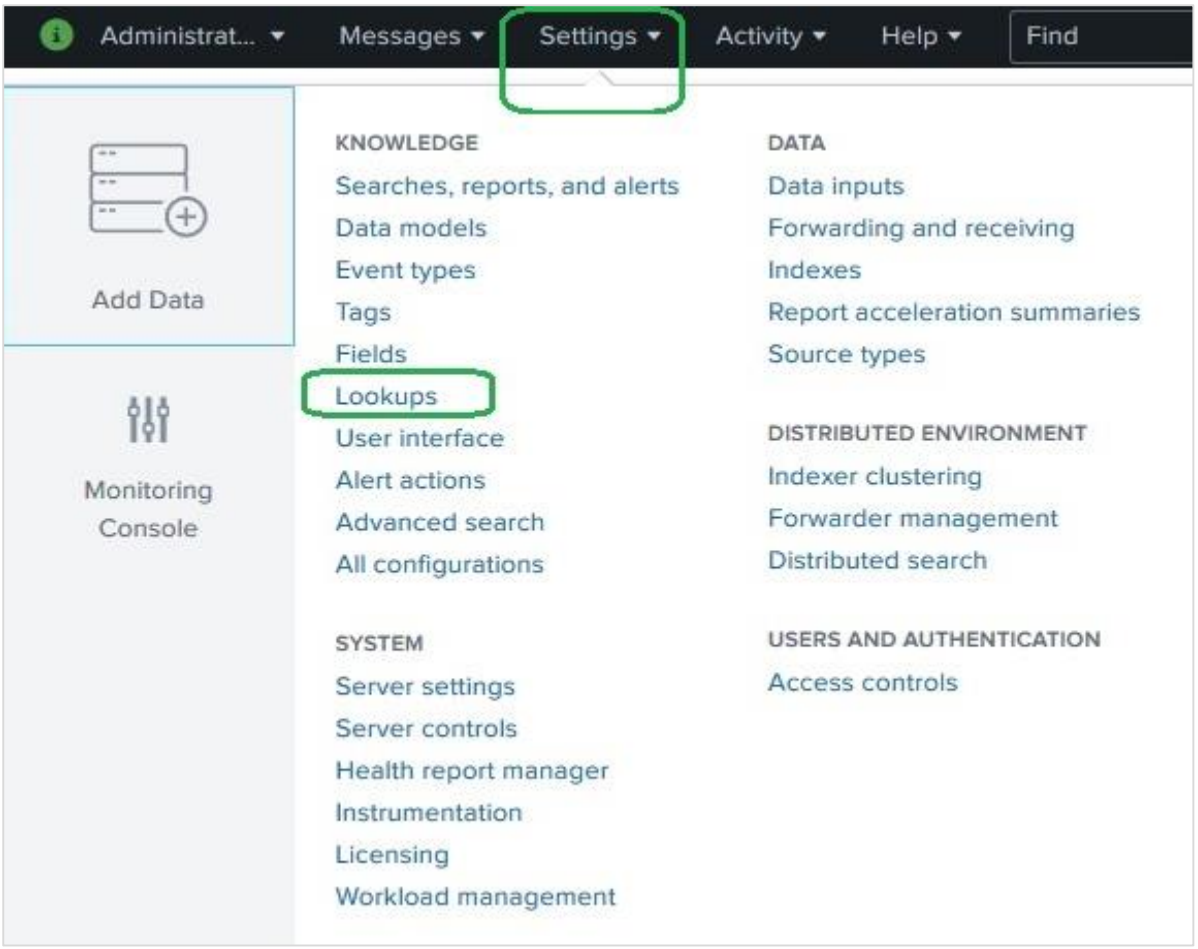
Create Lookup File

We consider the dataset with host as web_application, and look at the productid field. This field is just a number, but we want product names to be reflected in our query result set. We create a lookup file with the following details. Here, we have kept the name of the first field as **productid** which is same as the field we are going to use from the dataset.

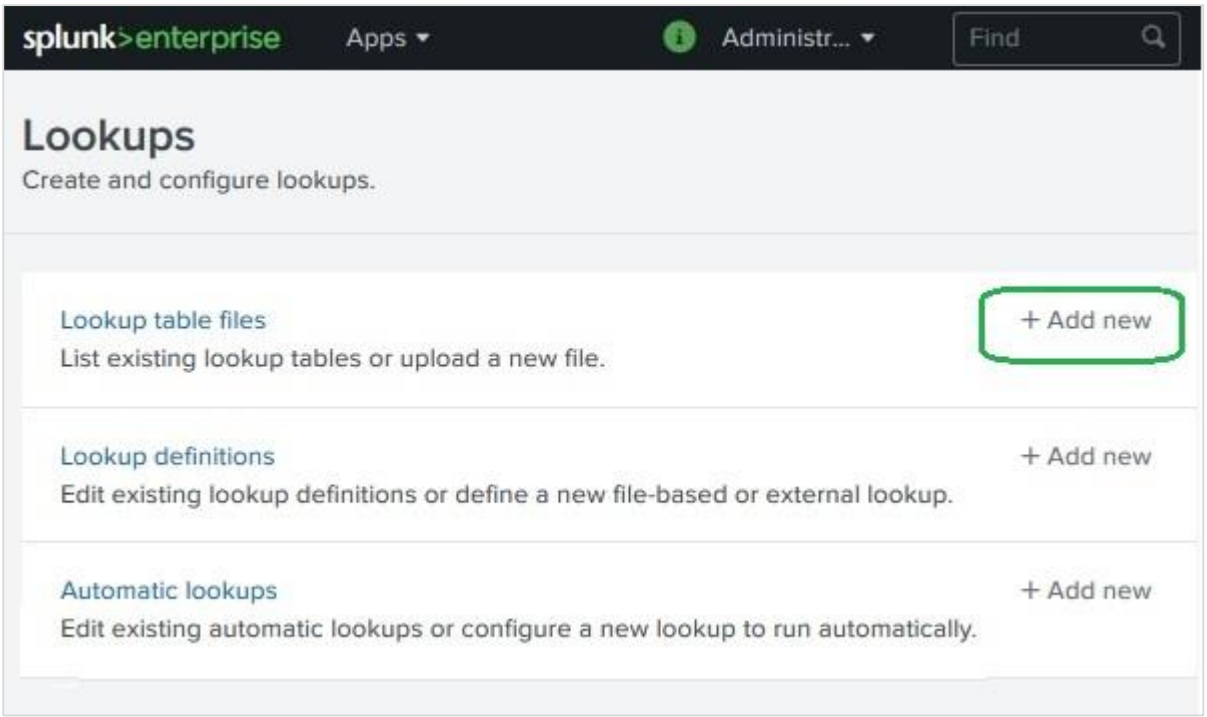
```
productId,productdescription
WC-SH-G04,Tablets
DB-SG-G01,PCs
DC-SG-G02,MobilePhones
SC-MG-G10,Wearables
WSC-MG-G10,Usb Light
GT-SC-G01,Battery
SF-BVS-G01,Hard Drive
```

Add the Lookup File

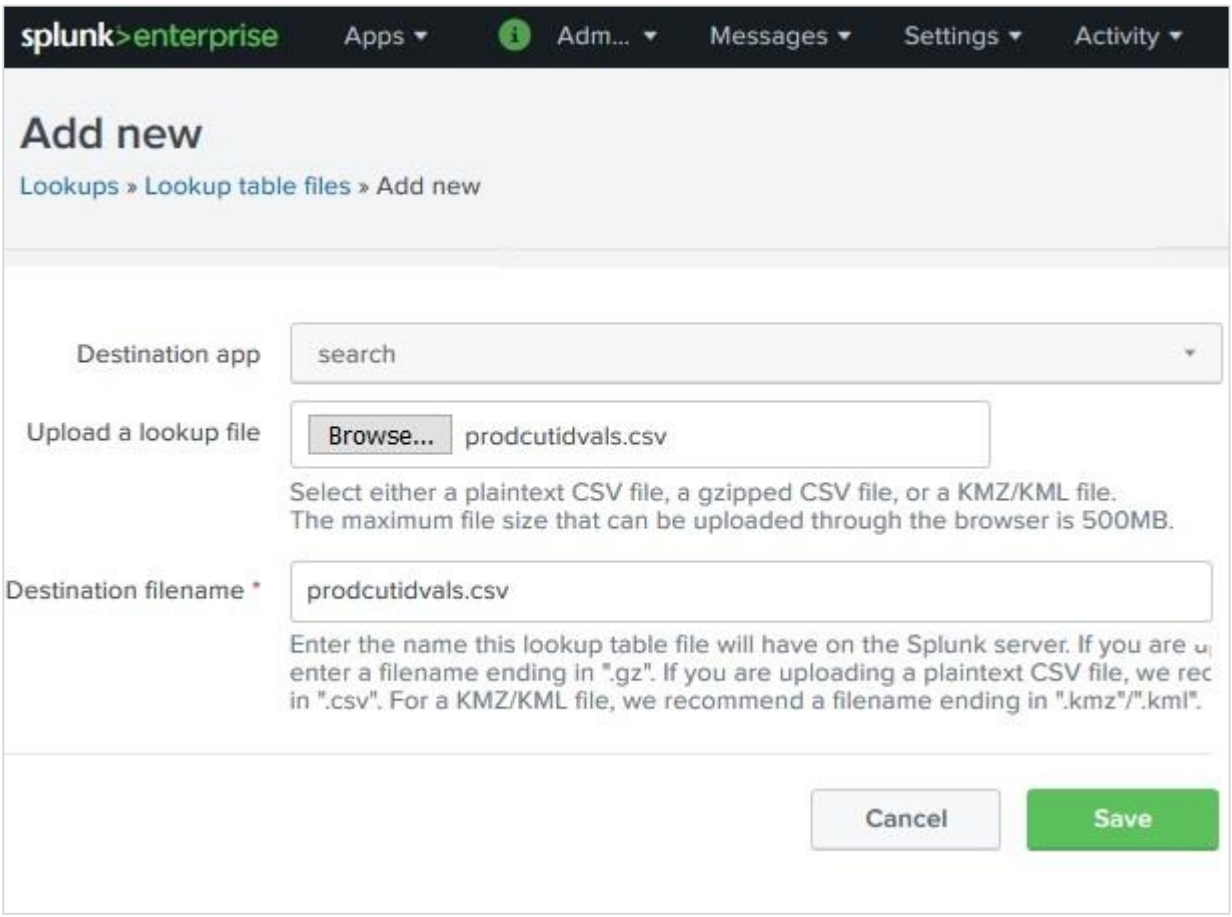
Next, we add the lookup file to Splunk environment by using the Settings screens as shown below:



After selecting the Lookups, we are presented with a screen to create and configure lookup. We select lookup table files as shown below.



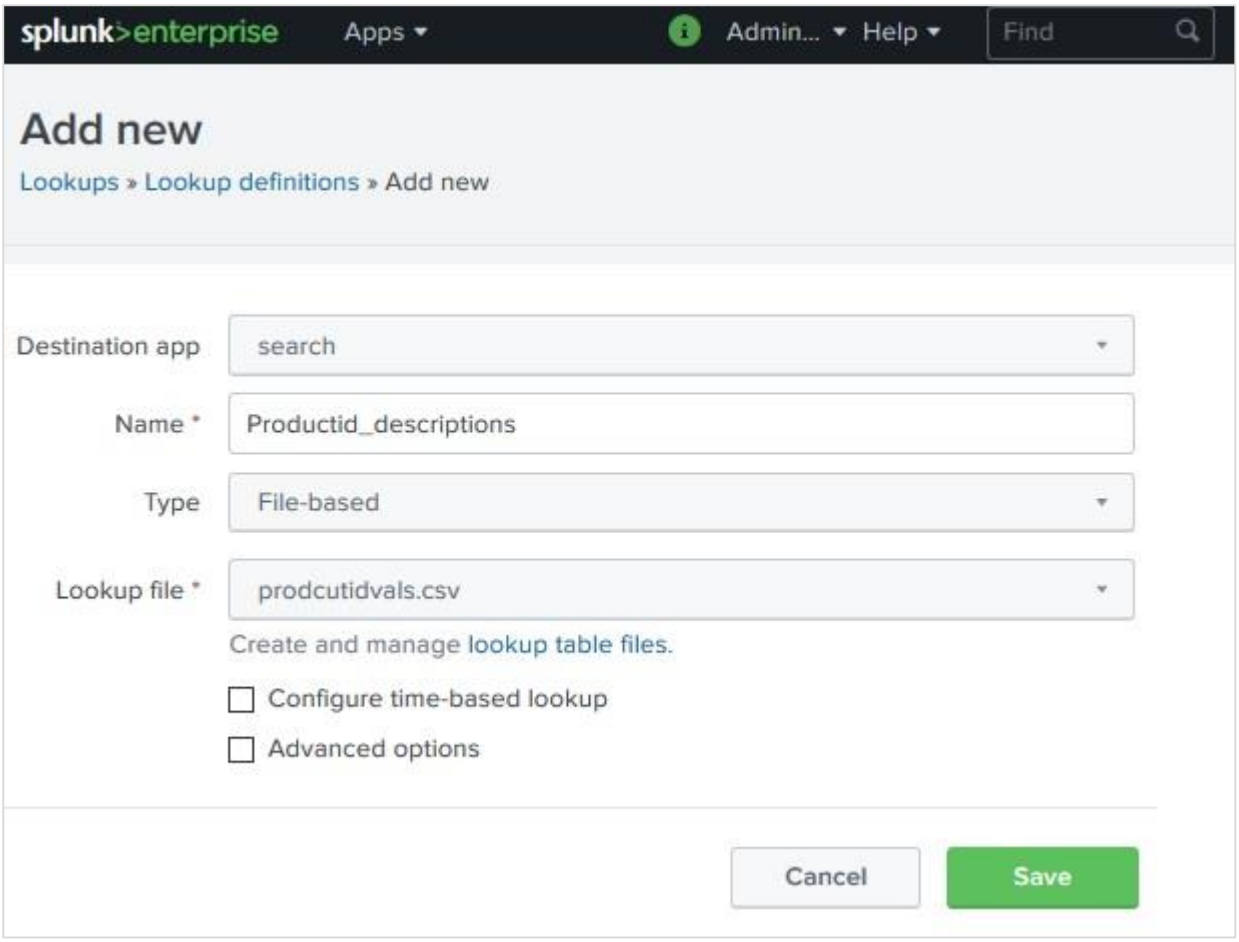
We browse to select the file **productidvals.csv** as our lookup file to be uploaded and select search as our destination app. We also keep the same destination file name.



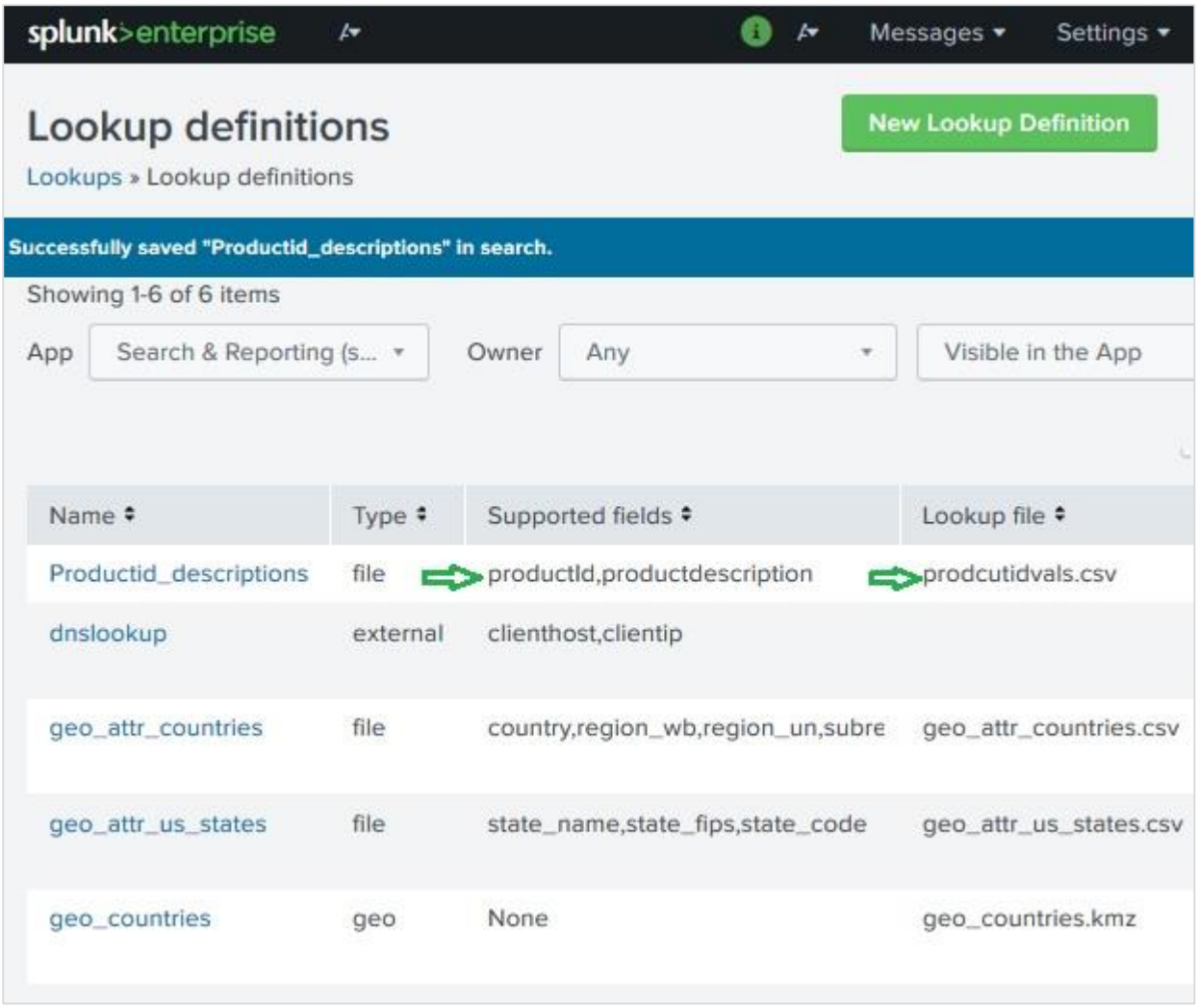
On clicking the save button, the file gets saved to the Splunk repository as a lookup file.

Create Lookup Definitions

For a search query to be able to lookup values from the Lookup file we just uploaded above, we need to create a lookup definition. We do this by again going to **Settings -> Lookups -> Lookup Definition -> Add New.**

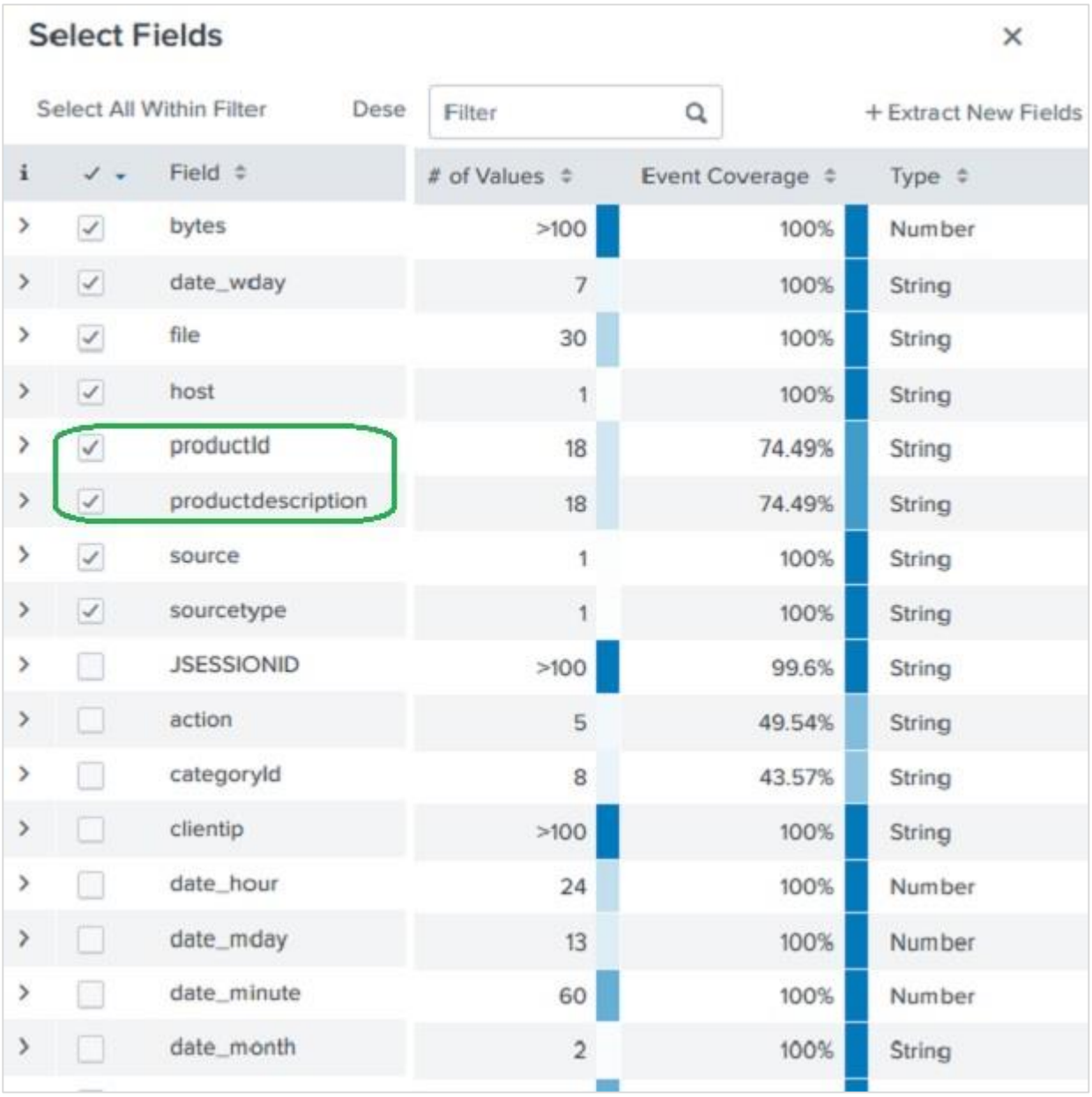


Next, we check the availability of the lookup definition we added by going to **Settings -> Lookups -> Lookup Definition**.



Selecting Lookup Field

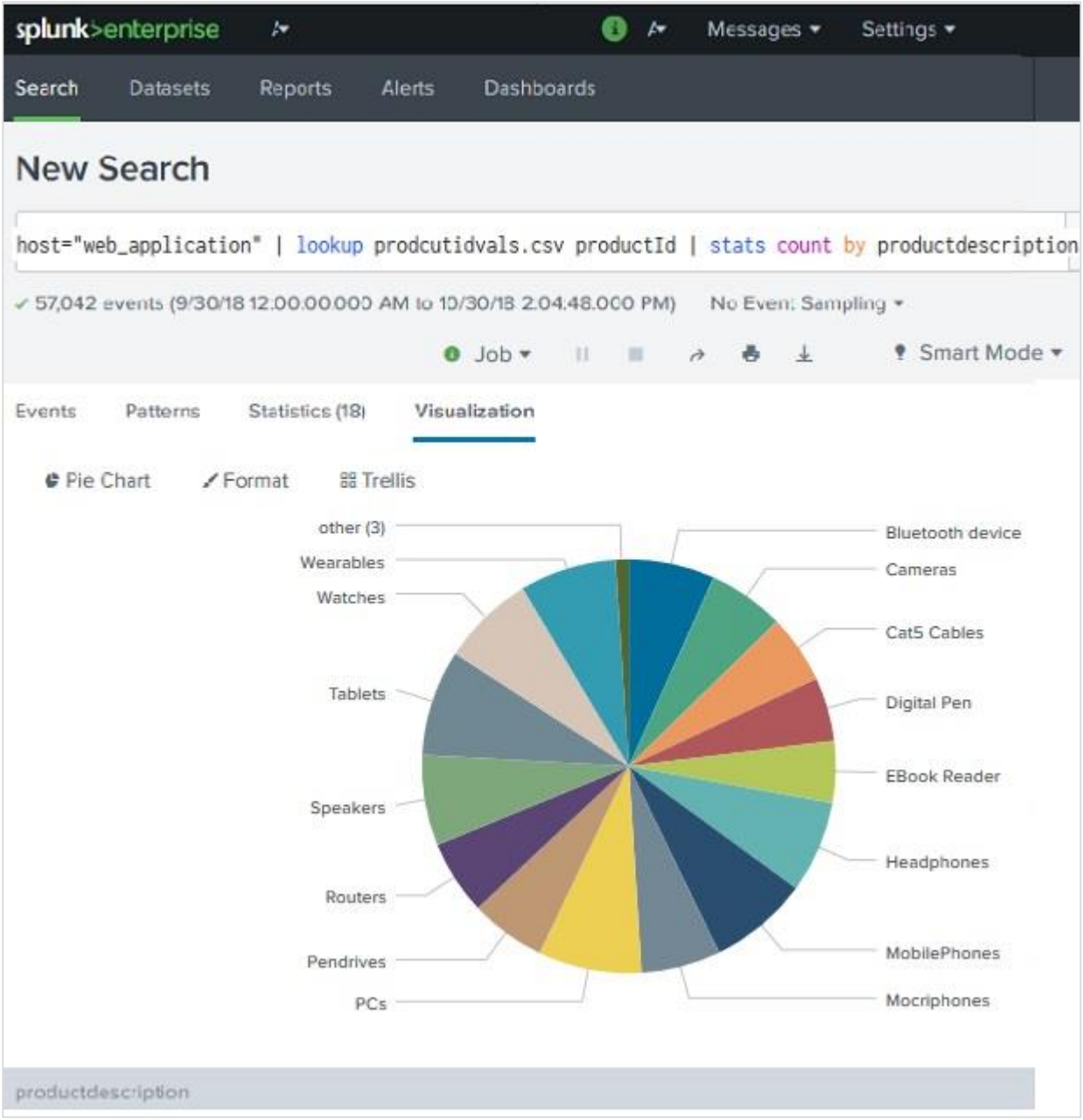
Next, we need to select the lookup field for our search query. This is done my going to **New search -> All Fields**. Then check the box for **productid** which will automatically add the **productdescription** field from the lookup file also.



Field	# of Values	Event Coverage	Type
bytes	>100	100%	Number
date_wday	7	100%	String
file	30	100%	String
host	1	100%	String
productId	18	74.49%	String
productdescription	18	74.49%	String
source	1	100%	String
sourcetype	1	100%	String
JSESSIONID	>100	99.6%	String
action	5	49.54%	String
categoryId	8	43.57%	String
clientip	>100	100%	String
date_hour	24	100%	Number
date_mday	13	100%	Number
date_minute	60	100%	Number
date_month	2	100%	String

Using the Lookup Field

Now we use the Lookup field in the search query as shown below. The visualization shows the result with productdescription field instead of productid.



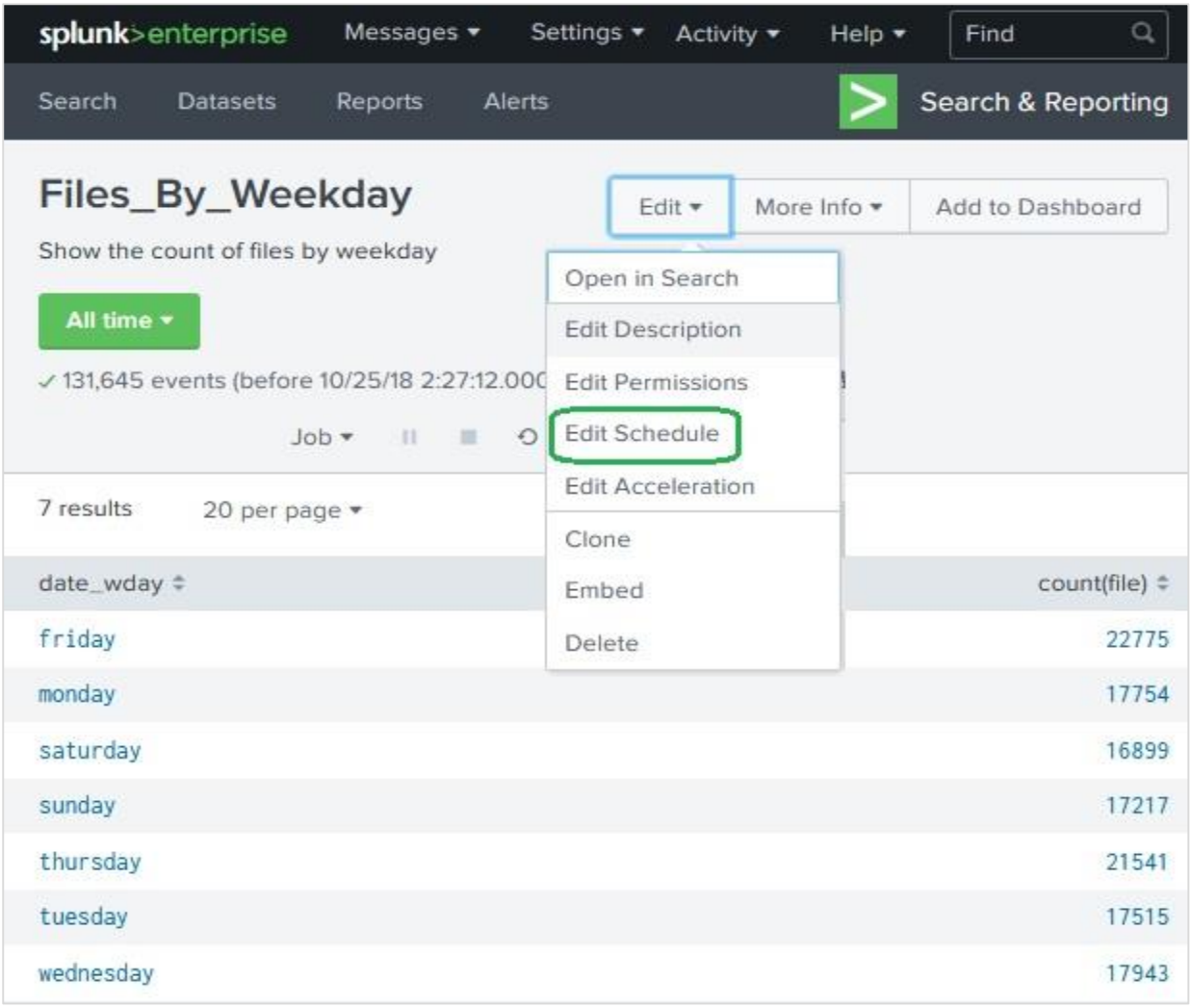
17. Splunk – Schedules and Alerts

Scheduling is the process of setting up a trigger to run the report automatically without the user's intervention. Below are the uses of scheduling a report:

- By running the same report at different intervals: monthly, weekly or daily, we can get results for that specific period.
- Improved performance of the dashboard as the reports finish running in the background before the dashboard is opened by the users.
- Sending of reports automatically via email after it finishes running.

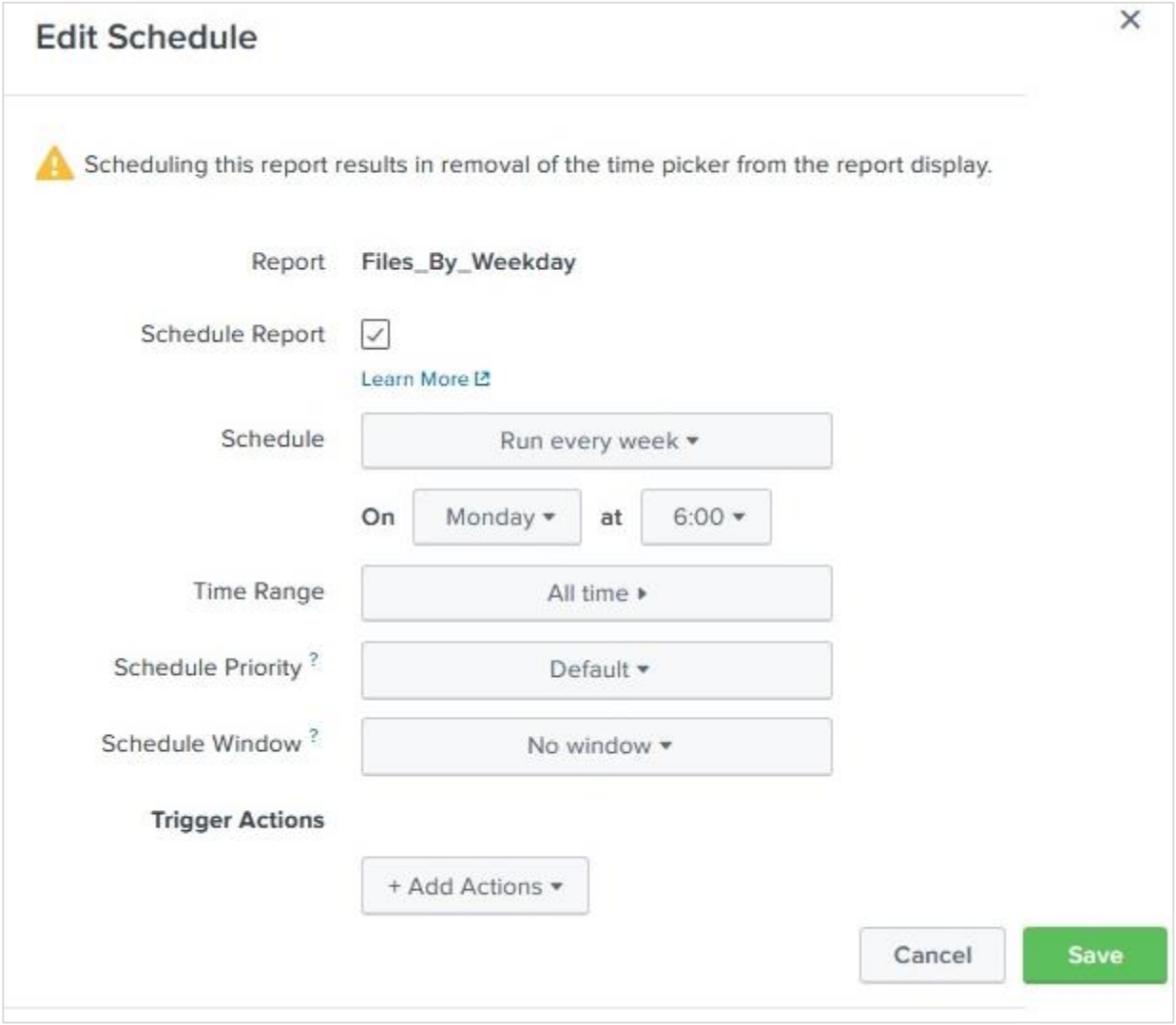
Creating a Schedule

A schedule is created by editing the report's schedule feature. We go to the **Edit Schedule** option on the Edit button as shown in the image below.



On clicking the edit schedule button, we get the next screen which lays out all the options for creating the schedule.

In the below example, we take all the default options and the report is scheduled to run every week on Monday at 6 AM.



Important Features of Scheduling

The following are the important features of scheduling:

Time Range – It indicates the time range from which the report must fetch the data. It can be last 15 minutes, last 4 hours or last week etc.

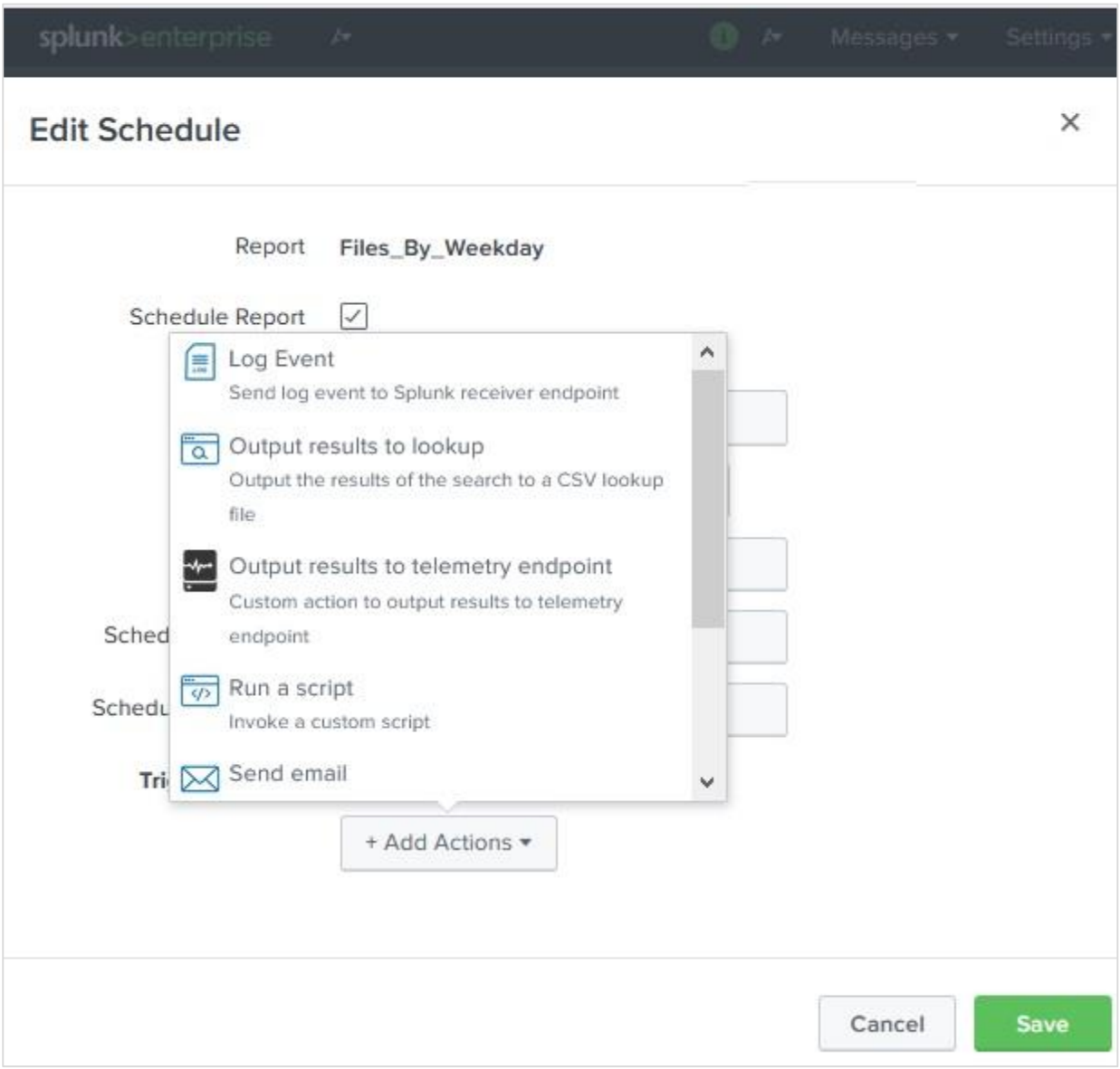
Schedule Priority – If more than one report is scheduled at the same time then this will determine the priority of a specific report.

Schedule Window – When there are multiple report schedules with same priority then we can choose a time window which will help the report to run at anytime during this window. If it is 5 minutes, then the report will run within 5 minutes of its scheduled time.

This helps in enhancing the performance of the scheduled reports by spreading their run time.

Schedule Actions

The schedule actions are meant to take some steps after the report is run. For example, you may want to send an email stating the run status of the report or run another script. Such actions can be carried out by setting the option by clicking on **Add Actions** button as shown below:

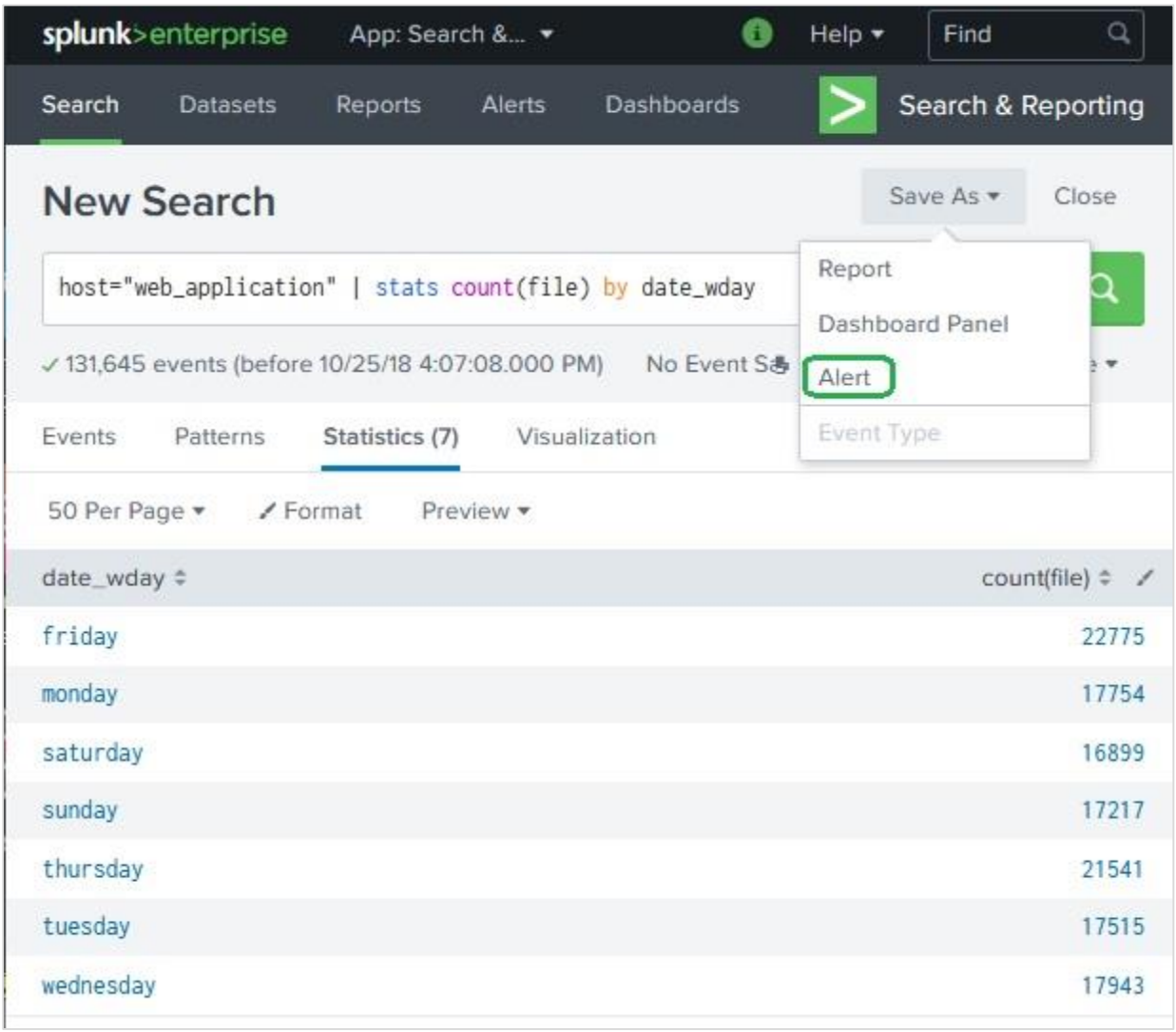


Alerts

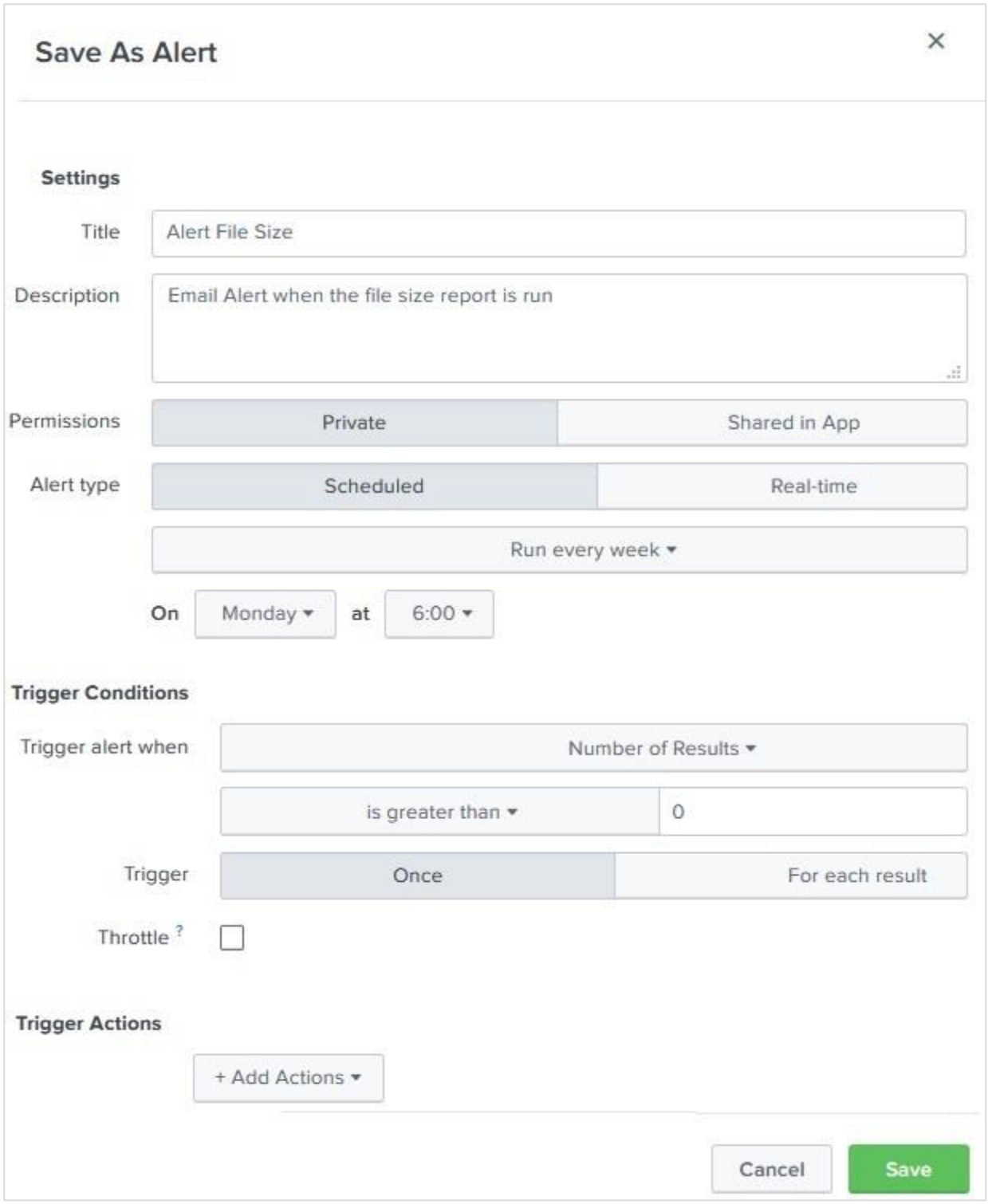
Splunk alerts are actions which get triggered when a specific criterion is met which is defined by the user. The goal of alerts can be logging an action, sending an email or output a result to a lookup file, etc.

Creating an Alert

You create an alert by running a search query and saving its result as an alert. In the below screenshot, we take the search for daywise file count and save the result as an alert by choosing the **Save As** option.



In the next screenshot, we configure the alert properties. The below image shows the configuration screen:



The purpose and choices of each of these options is explained below:

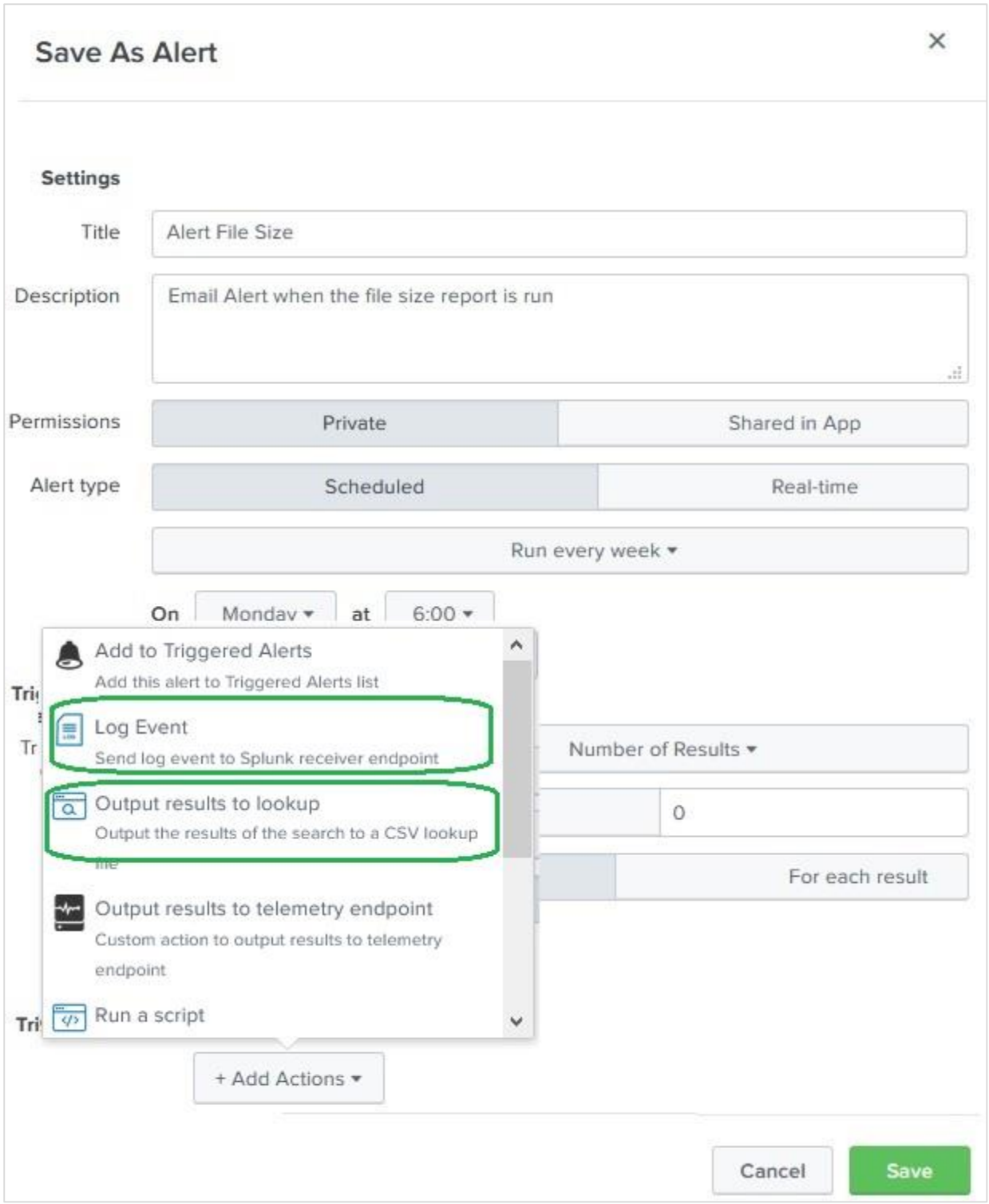
- **Title**: It is the name of the alert.
- **Description:** It is the detailed description of what the alert does.
- **Permission:** Its value decided who can access, run or edit the alert. If declared private, then only the creator of the alert has all the permissions. To be accessed

by others the option should be changed to **Shared in App**. In this case everyone has read access but only power user has the edit access for the alert.

- **Alert Type:** A scheduled alert runs at a pre-defined interval whose run time is defined by the day and time chosen from the drop downs. But the other option on real-time alert causes the search to run continuously in the background. Whenever the condition is met, the alert action is executed.

- **Trigger condition:** The trigger condition checks for the criteria mentioned in the trigger and sets off the alter only when the alert criteria is met. You can define number of results or number of sources or number of hosts in the search result to trigger the alert. If it is set for once, it will execute only once when the result condition is met but if it is set to **For** each Result, then it will run for every row in the result set where the trigger condition is met.

- **Trigger Actions:** The trigger actions can give a desired output or send an email when the trigger condition is met. The below image shows some of the important trigger actions available in Splunk.



18. Splunk – Knowledge Management

Splunk knowledge management is about maintenance of knowledge objects for a Splunk Enterprise implementation.

Below are the **main features of knowledge management:**

- Ensure that knowledge objects are being shared and used by the right groups of people in the organization.
- Normalize event data by implementing knowledge object naming conventions and retiring duplicate or obsolete objects.
- Oversee strategies for improved search and pivot performance (report acceleration, data model acceleration, summary indexing, batch mode search).
- Build data models for Pivot users.

Knowledge Object

It is a Splunk object to get specific information about your data. When you create a knowledge object, you can keep it private or you can share it with other users. The examples of knowledge object are: saved searches, tags, field extractions, lookups, etc.

Uses of Knowledge Objects

On using the Splunk software, the knowledge objects are created and saved. But they may contain duplicate information, or they may not be used effectively by all the intended audience. To address such issues, we need to manage these objects. This is done by classifying them properly and then using proper permission management to handle them. Below are the uses and classification of various knowledge objects:

Fields and field extractions

Fields and field extractions is the first layer of Splunk software knowledge. The fields automatically extracted from the Splunk software from the IT data help bring meaning to the raw data. The manually extracted fields expand and improve upon this layer of meaning.

Event types and transactions

Use event types and transactions to group together interesting sets of similar events. Event types group together sets of events discovered through searches. Transactions are collections of conceptually-related events that span time.

Lookups and workflow actions

Lookups and workflow actions are categories of knowledge objects that extend the usefulness of your data in various ways. Field lookups enable you to add fields to your data from external data sources such as static tables (CSV files) or Python-based

commands. Workflow actions enable interactions between fields in your data and other applications or web resources, such as a WHOIS lookup on a field containing an IP address.

Tags and aliases

Tags and aliases are used to manage and normalize sets of field information. You can use tags and aliases to group sets of related field values together, and to give extracted field tags that reflect different aspects of their identity. For example, you can group events from set of hosts in a particular location (such as a building or city) together by giving the same tag to each host.

If you have two different sources using different field names to refer to same data, then you can normalize your data by using aliases (by aliasing clientip to ipaddress, for example).

Data models

Data models are representations of one or more datasets, and they drive the Pivot tool, enabling Pivot users to quickly generate useful tables, complex visualizations, and robust reports without needing to interact with the Splunk software search language. Data models are designed by knowledge managers who fully understand the format and semantics of their indexed data. A typical data model makes use of other knowledge object types.

We will discuss some of the examples of these knowledge objects in the subsequent chapters.

19. Splunk – Subsearching

Subsearch is a special case of the regular search when the result of a secondary or inner query is the input to the primary or outer query. It is similar to the concept of subquery in case of SQL language. In Splunk, the primary query should return one result which can be input to the outer or the secondary query.

When a search contains a subsearch, the subsearch is run first. Subsearches must be enclosed in square brackets in the primary search.

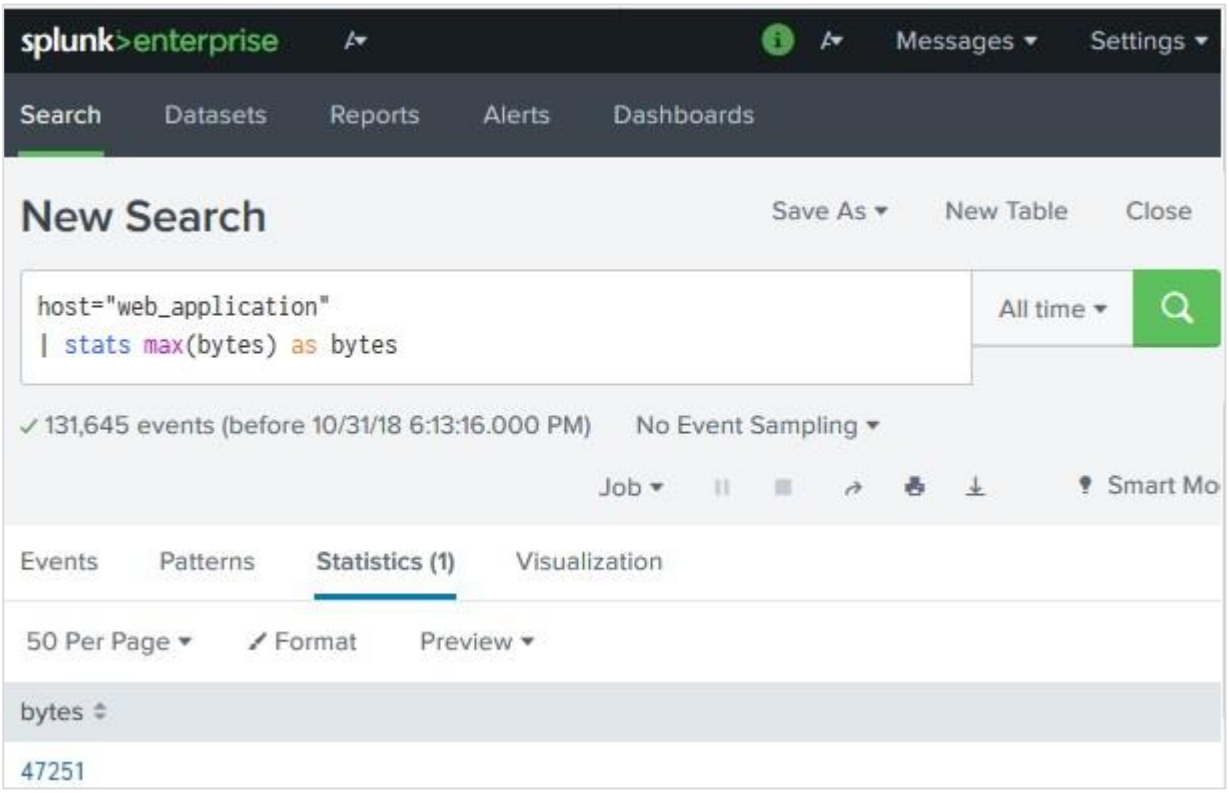
Example

We consider the case of finding a file from web log which has maximum byte size. But that may vary every day. Then we want to find only those events where the file size is equal to the maximum size, and is a Sunday.

Create the Subsearch

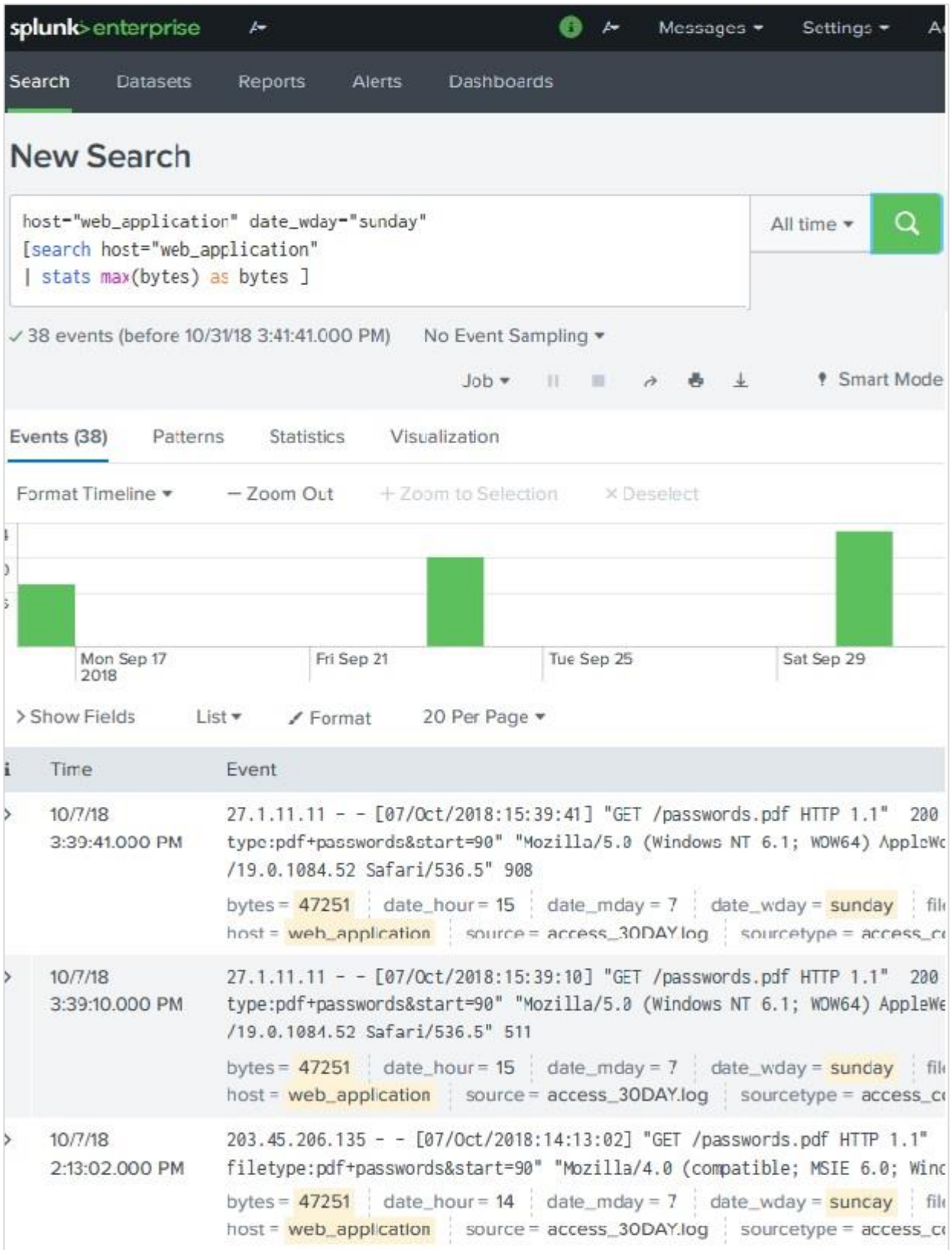
We first create the subsearch to find the maximum file size. We use the function **Stat max** with the field named bytes as the argument. This identifies the maximum size of the file for the time frame for which the search query is run.

The below image shows the search and the result of this subsearch:



Adding the Subsearch

Next, we add the subsearch query to the primary or the outer query by putting the subsearch inside square brackets. Also the search clause is added to the subsearch query.



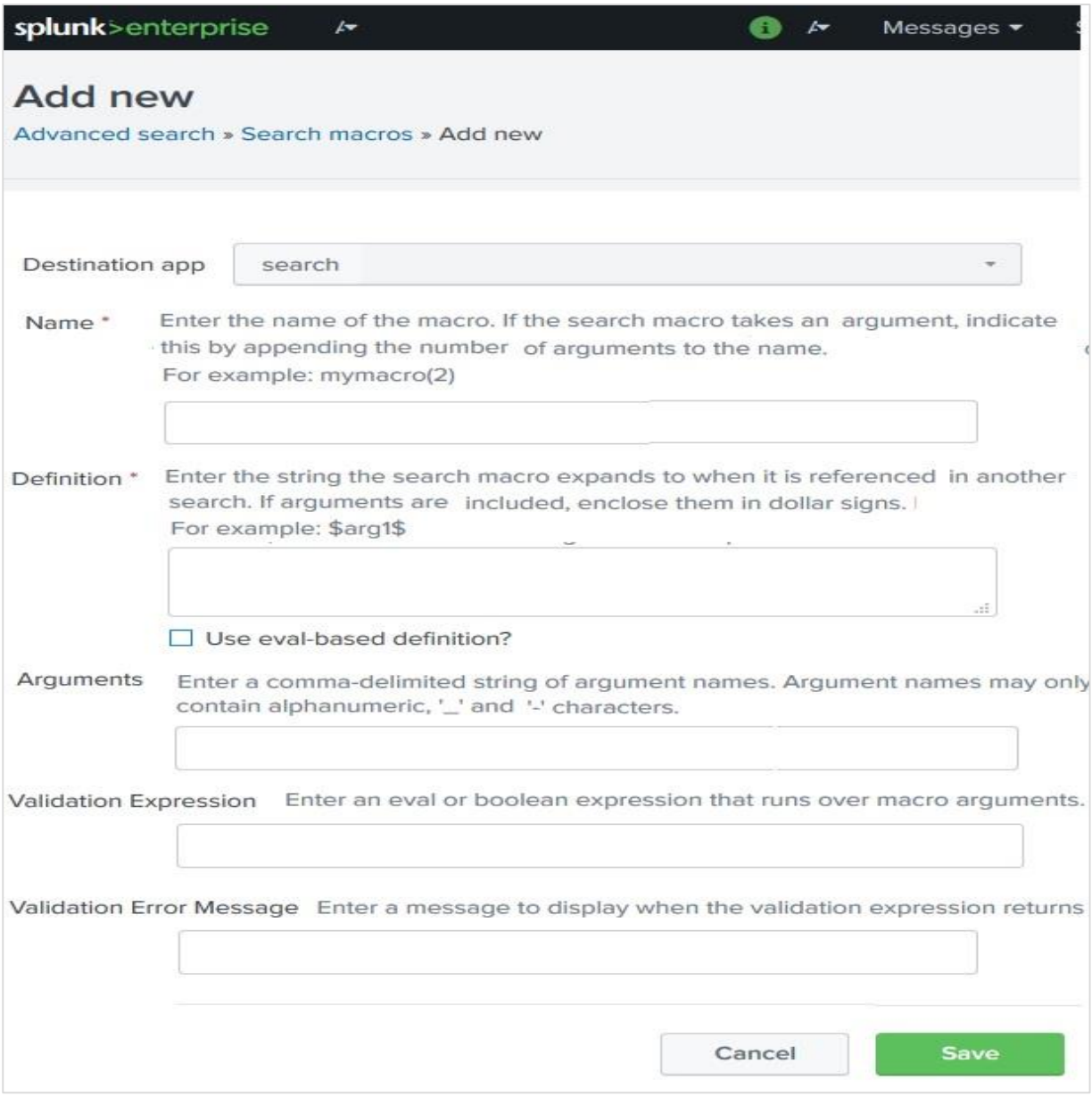
As we see, the result contains only the events where the file size is equal to the max file size found by considering all the events, and the event day is a Sunday.

20. Splunk – Search Macros

Search macros are reusable blocks of Search Processing Language (SPL) that you can insert into other searches. They are used when you want to use the same search logic on different parts or values in the data set dynamically. They can take arguments dynamically and the search result will be updated as per the new values.

Macro Creation

To create the search macro, we go to the **Settings -> Advanced Search -> Search Macros -> Add New**. This brings up the below screen where we start creating the macro.



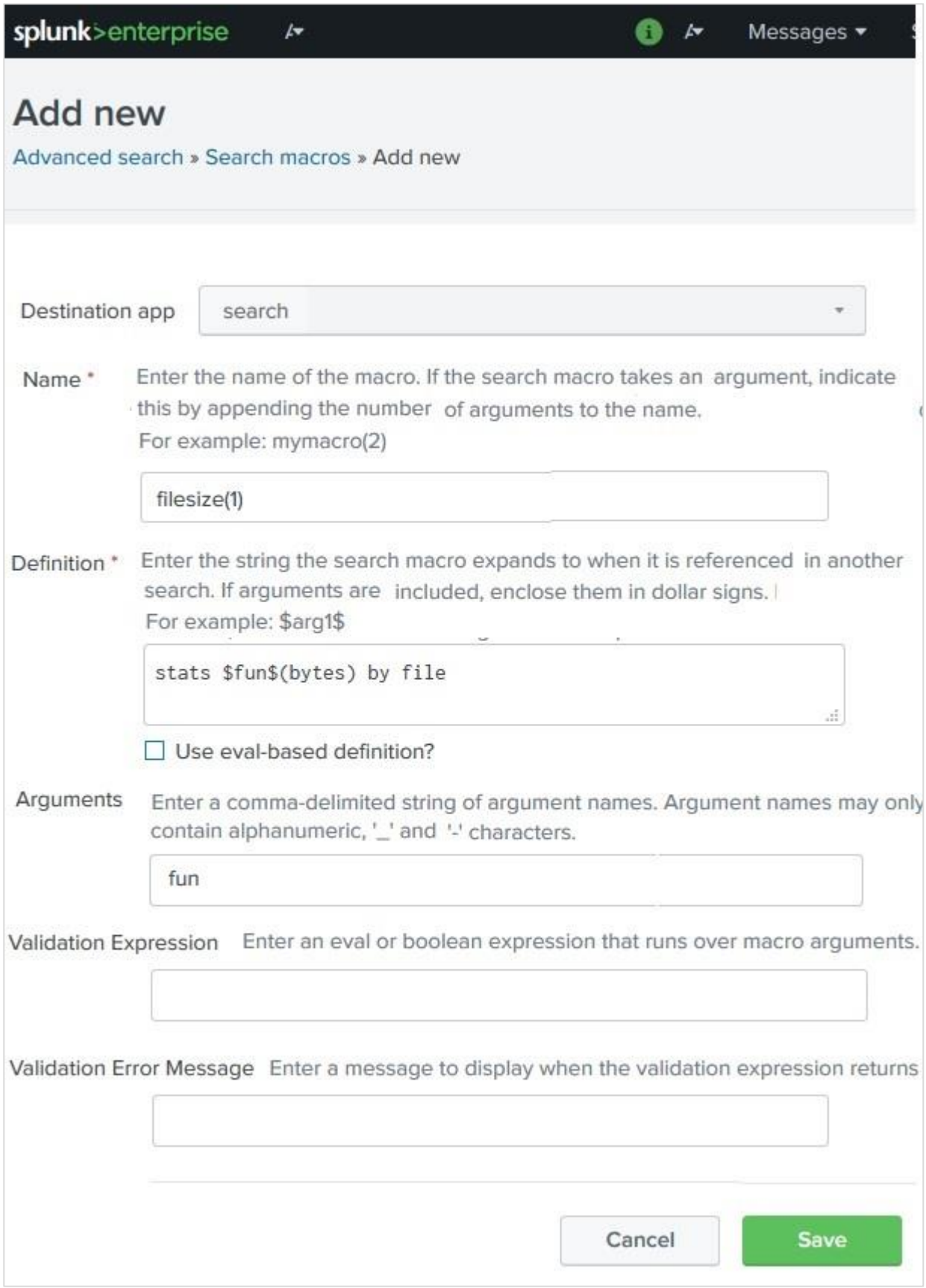
Macro Scenario

We want to show various stats about the file size from the **web_applications** log. The stats are about max, min and avg value of the filesize using the bytes field in the log. The result should display these stats for each file listed in the log.

So here the type of the stats is dynamic in nature. The name of the stats function will be passed as an argument to the macro.

Defining the Macro

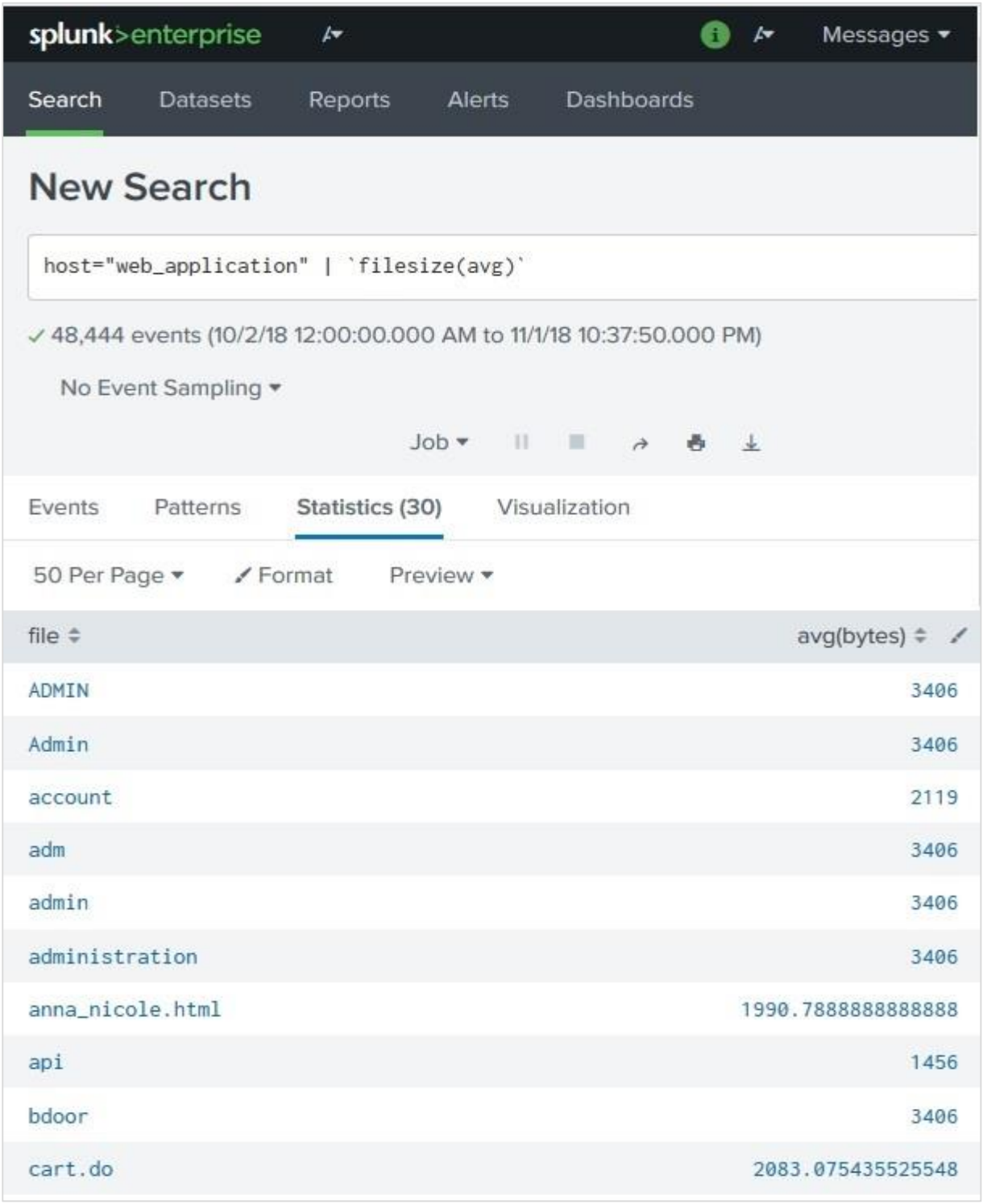
Next, we define the macro by setting various properties as shown in the below screen. The name of the macro contains (1), indicating that there is one argument to be passed into the macro when it is used in the search string. **fun** is the argument which will be passed on to the macro during execution in the search query.



Using the Macro

To use the macro, we make it a part of the search string. On passing different values for the argument we see different results as expected.

Consider finding the average size in bytes of the files. We pass avg as the argument and get the result as shown below. The macro has been kept under ` sign as part of the search query.



Similarly, if we want the maximum file size for each of the files present in the log, then we use **max** as the argument. The result is as shown below.

21. Splunk – Event Types

In Splunk search, we can design our own events from a dataset based on certain criteria. For example, we search for only the events which have a http status code of 200. This event now can be saved as an event type with a user defined name as **status200** and use this event name as part of future searches.

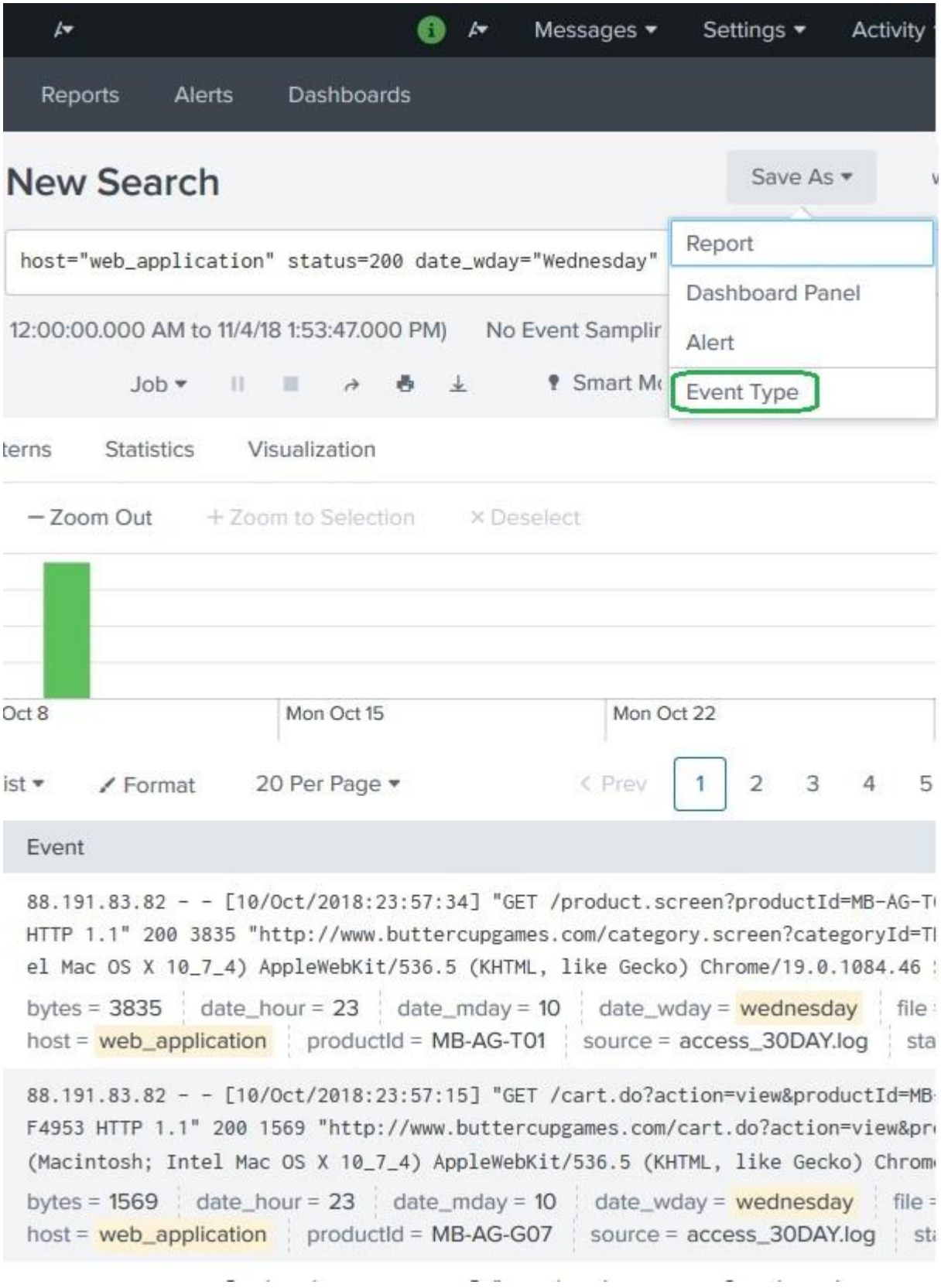
In short, an event type represents a search that returns a specific type of event or a useful collection of events. Every event that can be returned by the search gets an association with that event type.

Creating Event Type

There are two ways to create an event type after we have decided the search criteria. One is to **run a** search and then save it as an Event Type. Another is to **add a new Event Type from the settings tab**. We will see both the ways of creating it in this section.

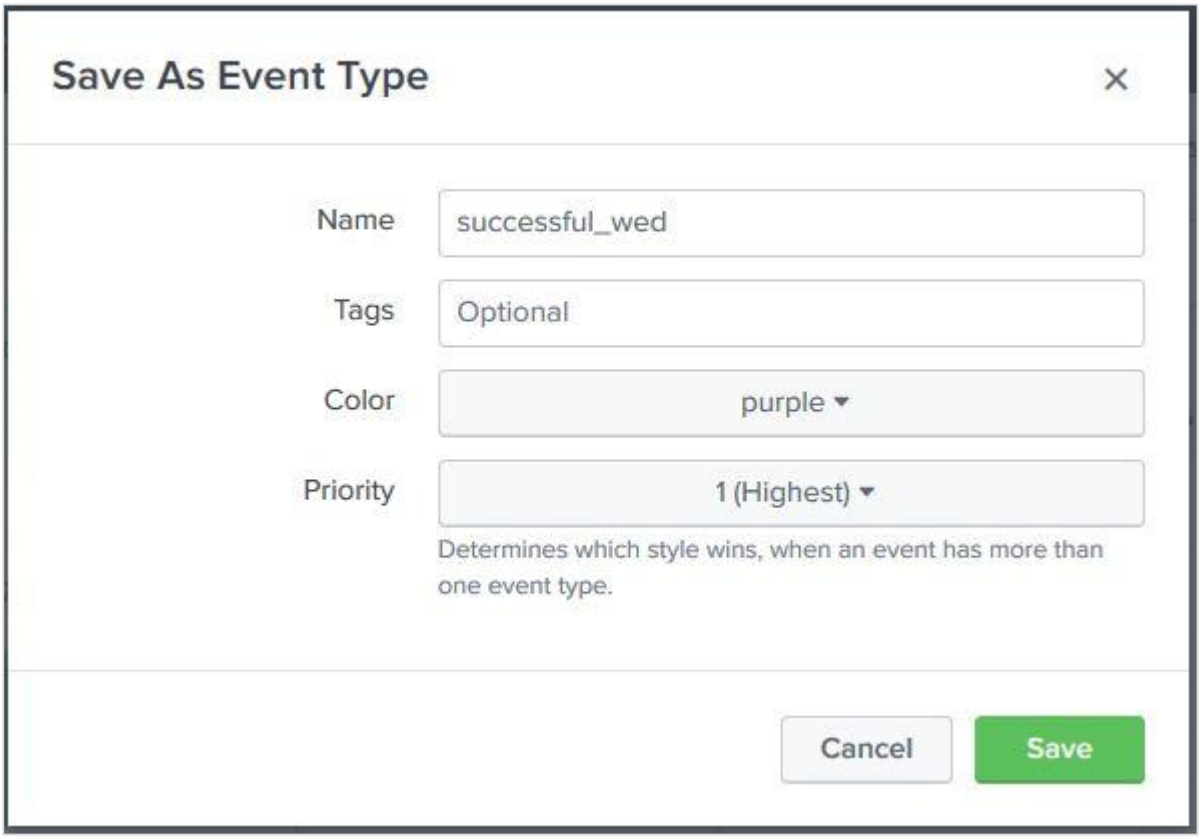
Using a Search

Consider the search for the events which have the criteria of successful http status value of 200 and the event type run on a Wednesday. After running the search query, we can choose **Save As** option to save the query as an Event Type.



The next screen prompts to give a name for the Event Type, choose a Tag which is optional and then choose a colour with which the events will be highlighted. The priority option

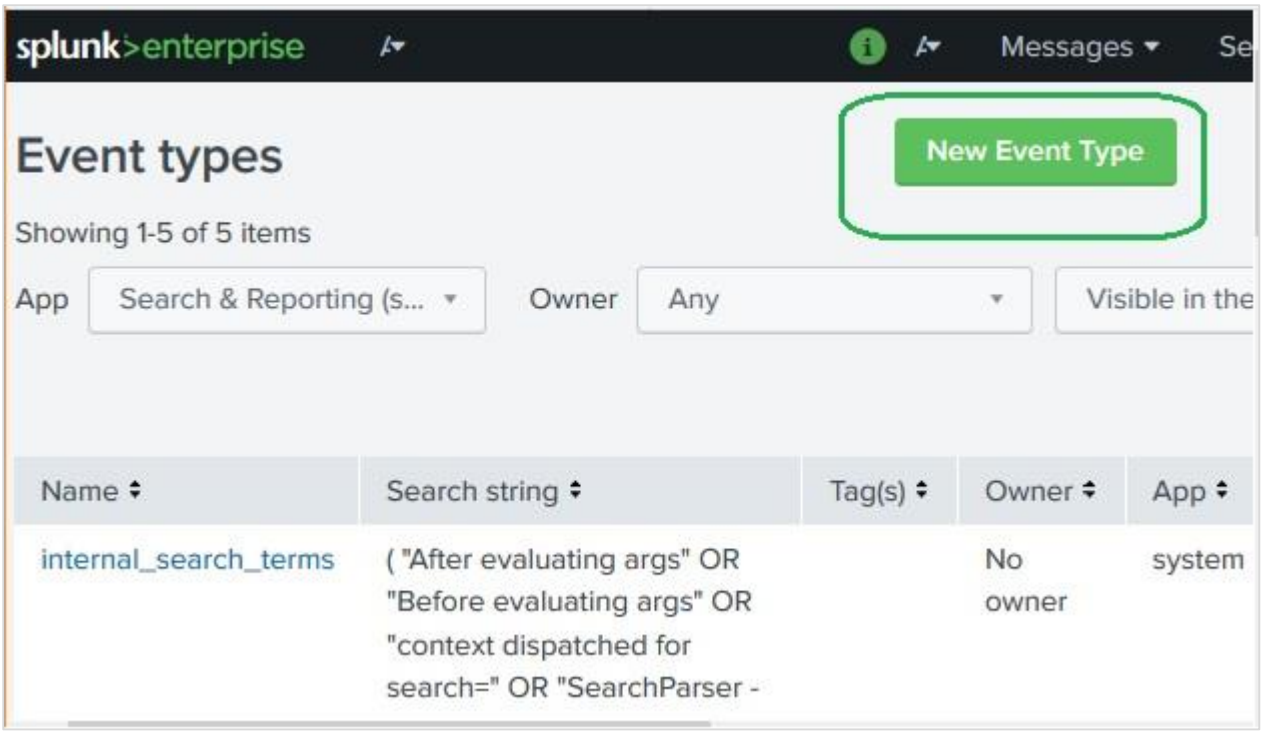
decides which event type will be displayed first in case two or more event types match the same event.



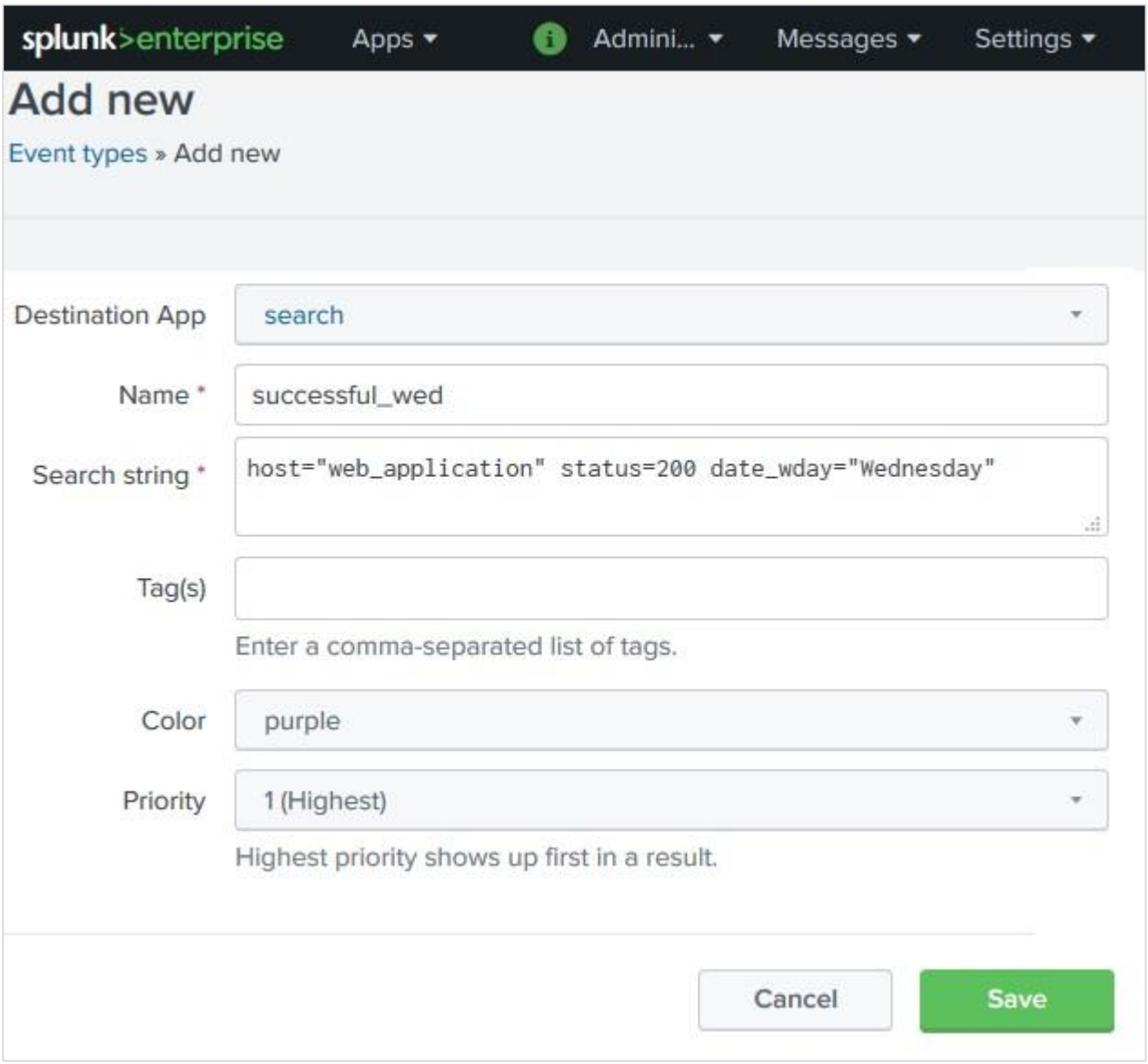
Finally, we can see the Event Type has been created by going to the **Settings -> Event Types** option.

Using New Event Types

The other option to create a new Event Type is to use the **Settings -> Event Types** option as shown below where we can add a new Event Type:

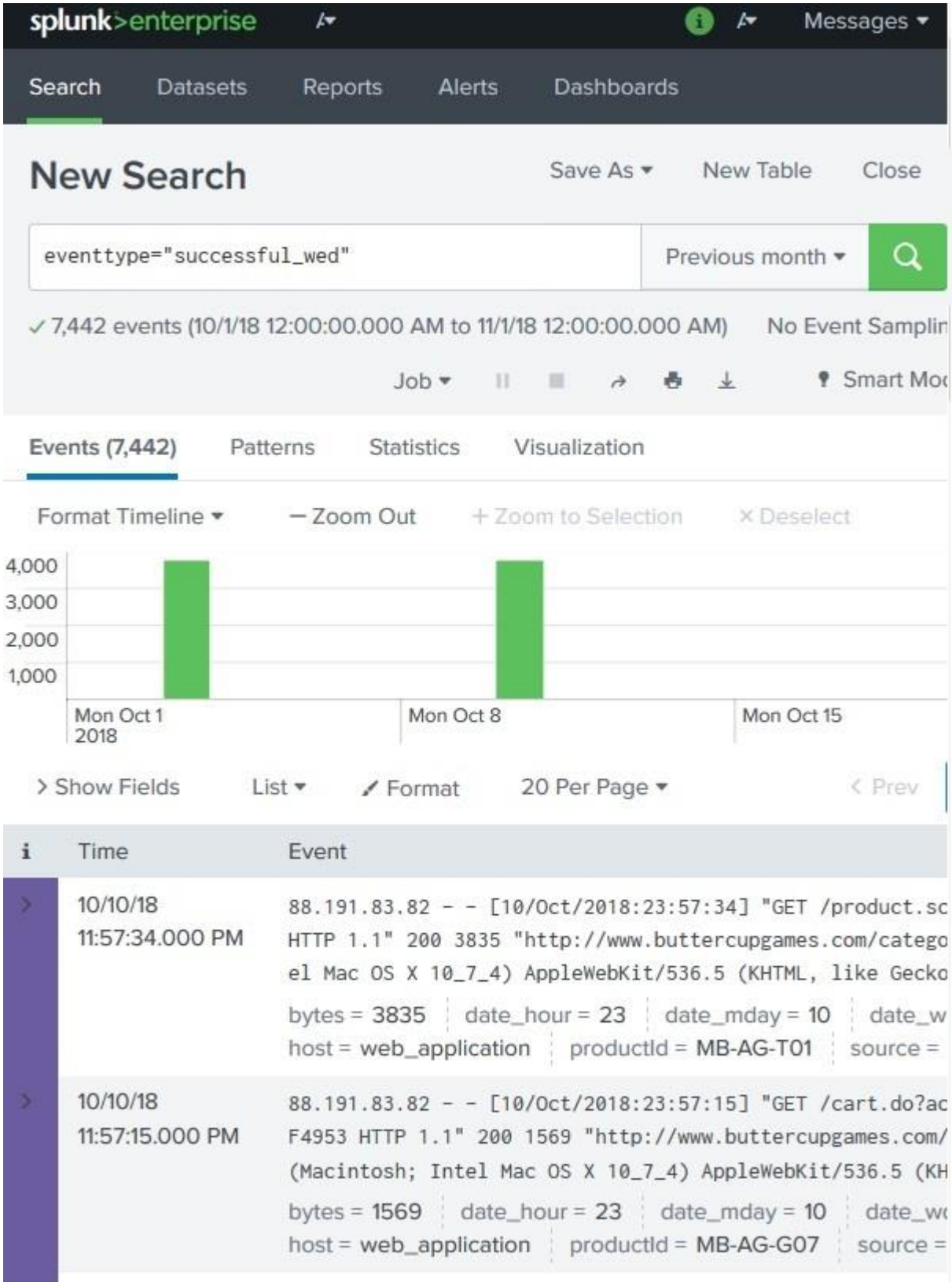


On clicking the button **New Event Type,** we get the following screen to add the same query as in the previous section.



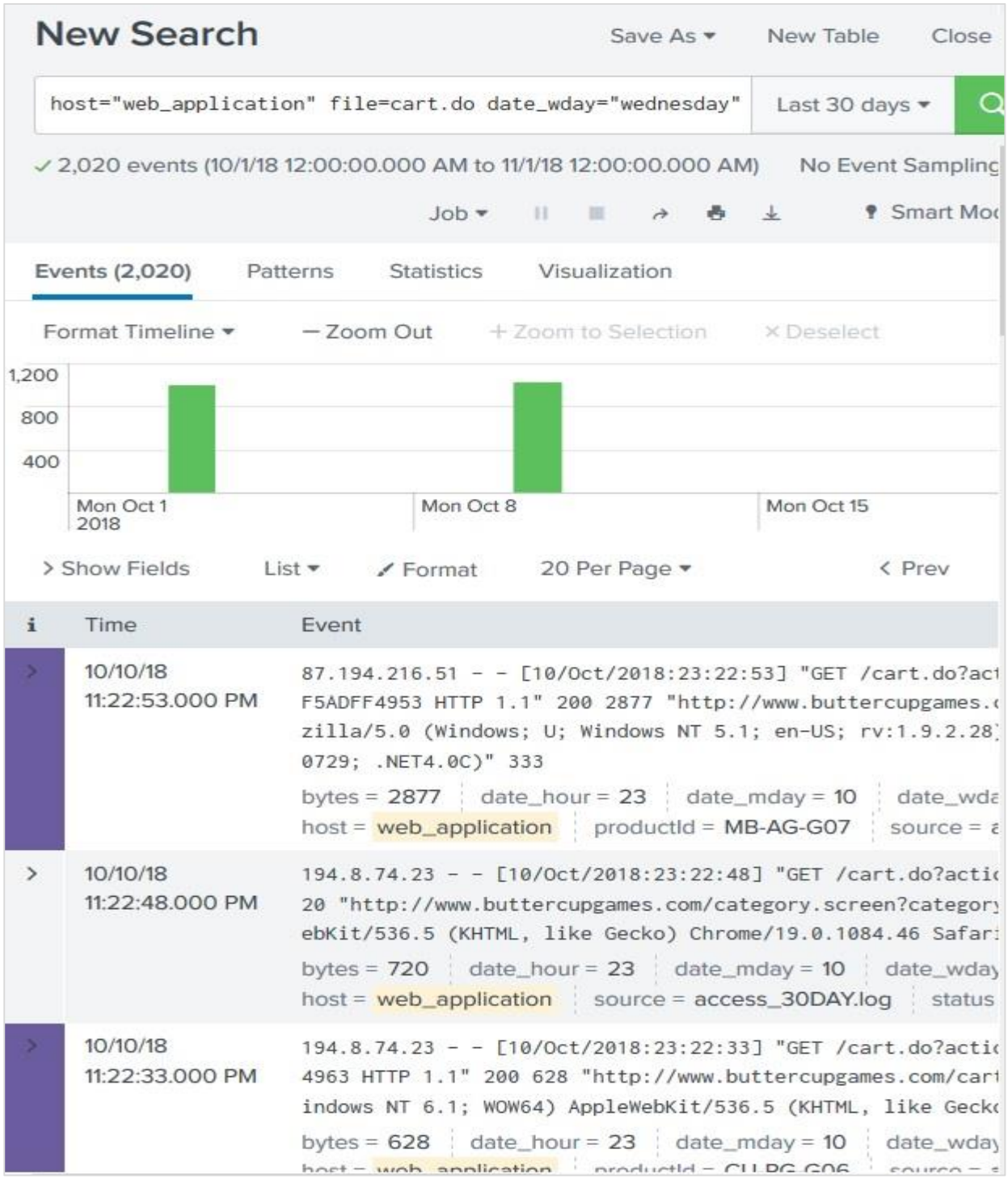
Viewing the Event Type

To view the event we just created above, we can write the below search query in the search box and we can see the resulting events along with the colour we have chosen for the event type.



Using the Event Type

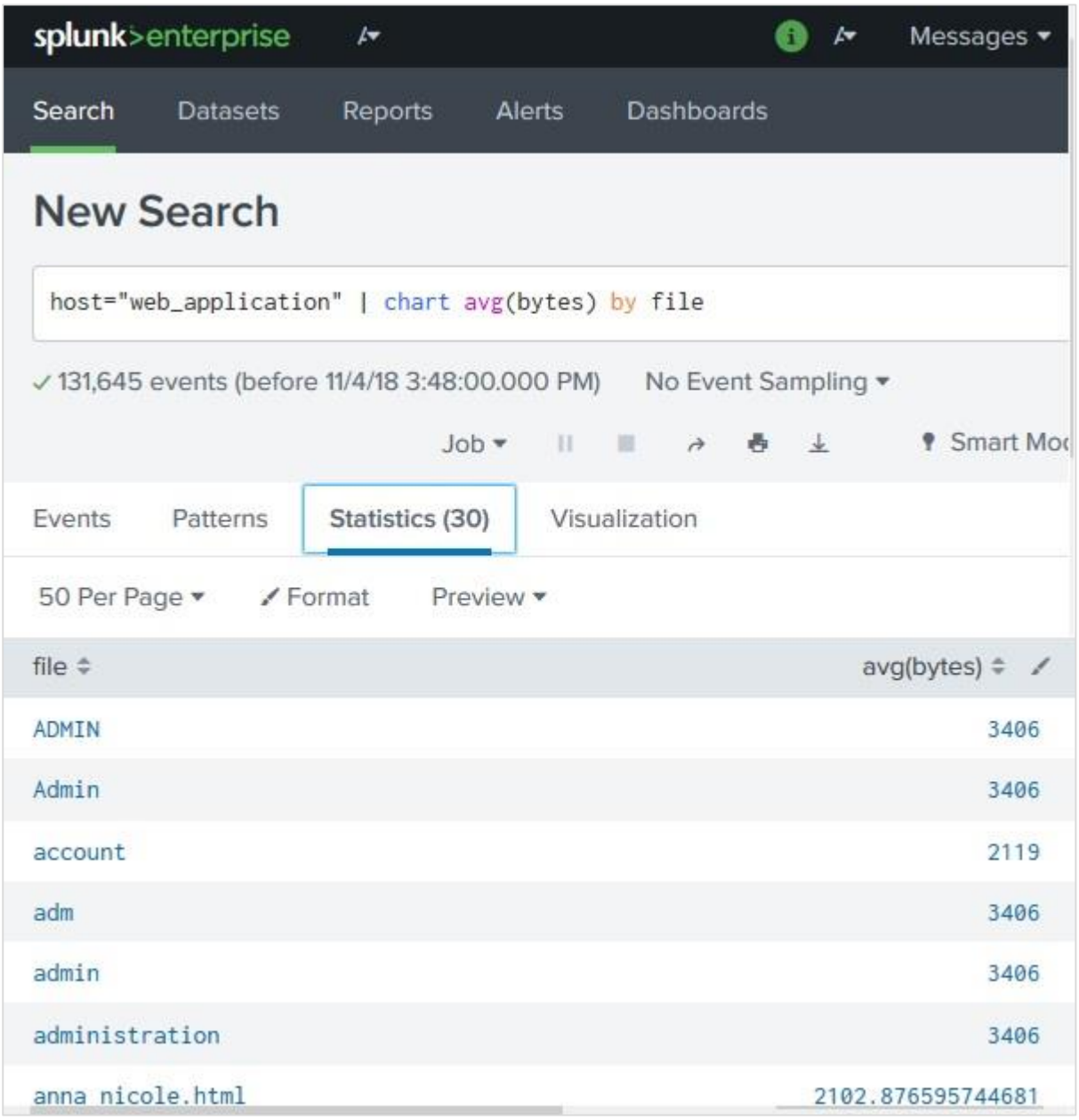
We can use the Event type along with other queries. Here we specify some partial criteria from the Event Type and the result is a mix of events which shows the coloured and non-coloured events in the result.



22. Splunk – Basic Chart

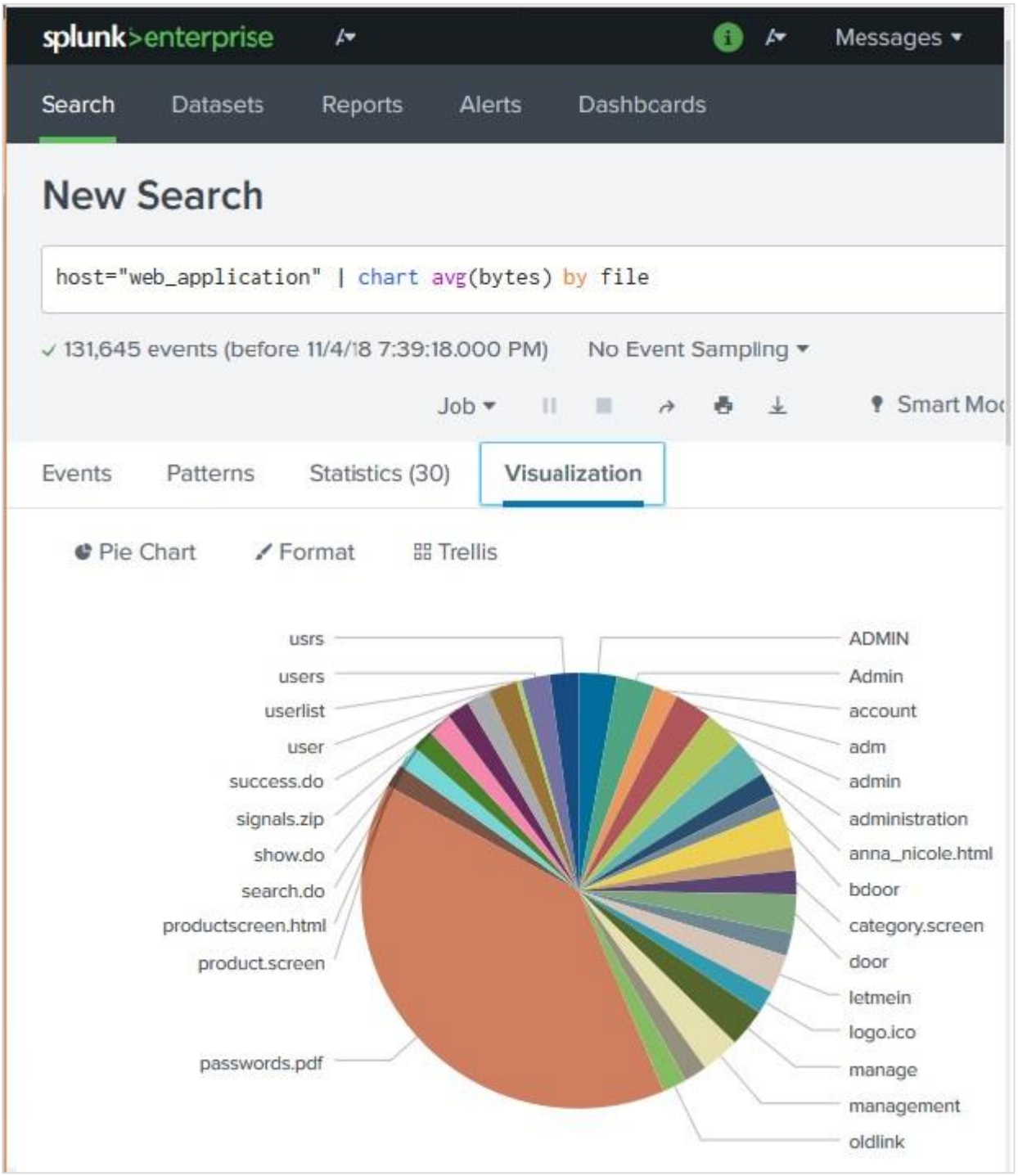
Splunk has great visualization features which shows a variety of charts. These charts are created from the results of a search query where appropriate functions are used to give numerical outputs.

For example, if we look for the average file size in bytes from the data set named web_applications, we can see the result in the statistics tab as shown below:



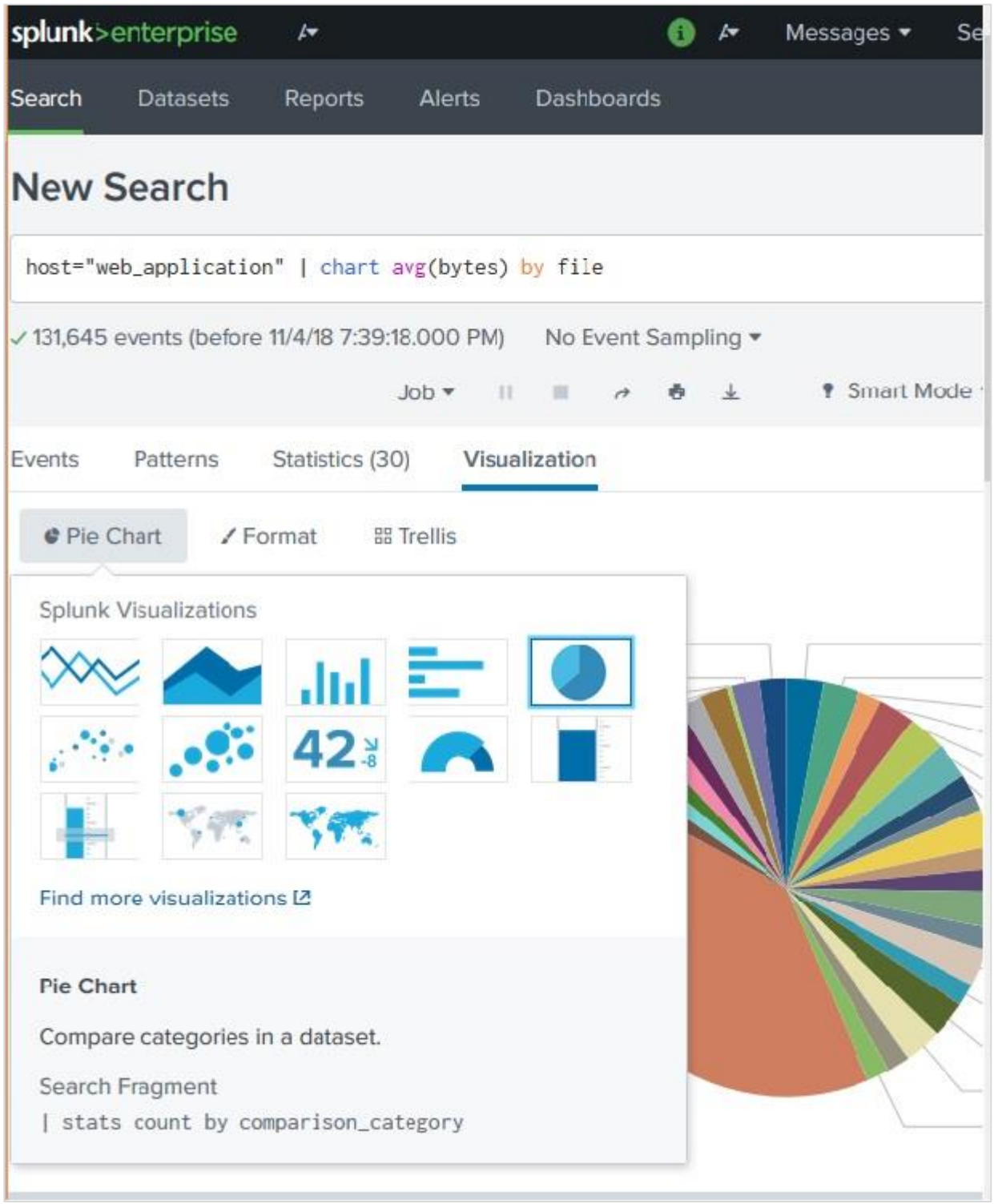
Creating Charts

In order to create a basic chart, we first ensure that the data is visible in the statistics tab as shown above. Then we click on the Visualization tab to get the corresponding chart. The above data produces a pie chart by default as shown below.



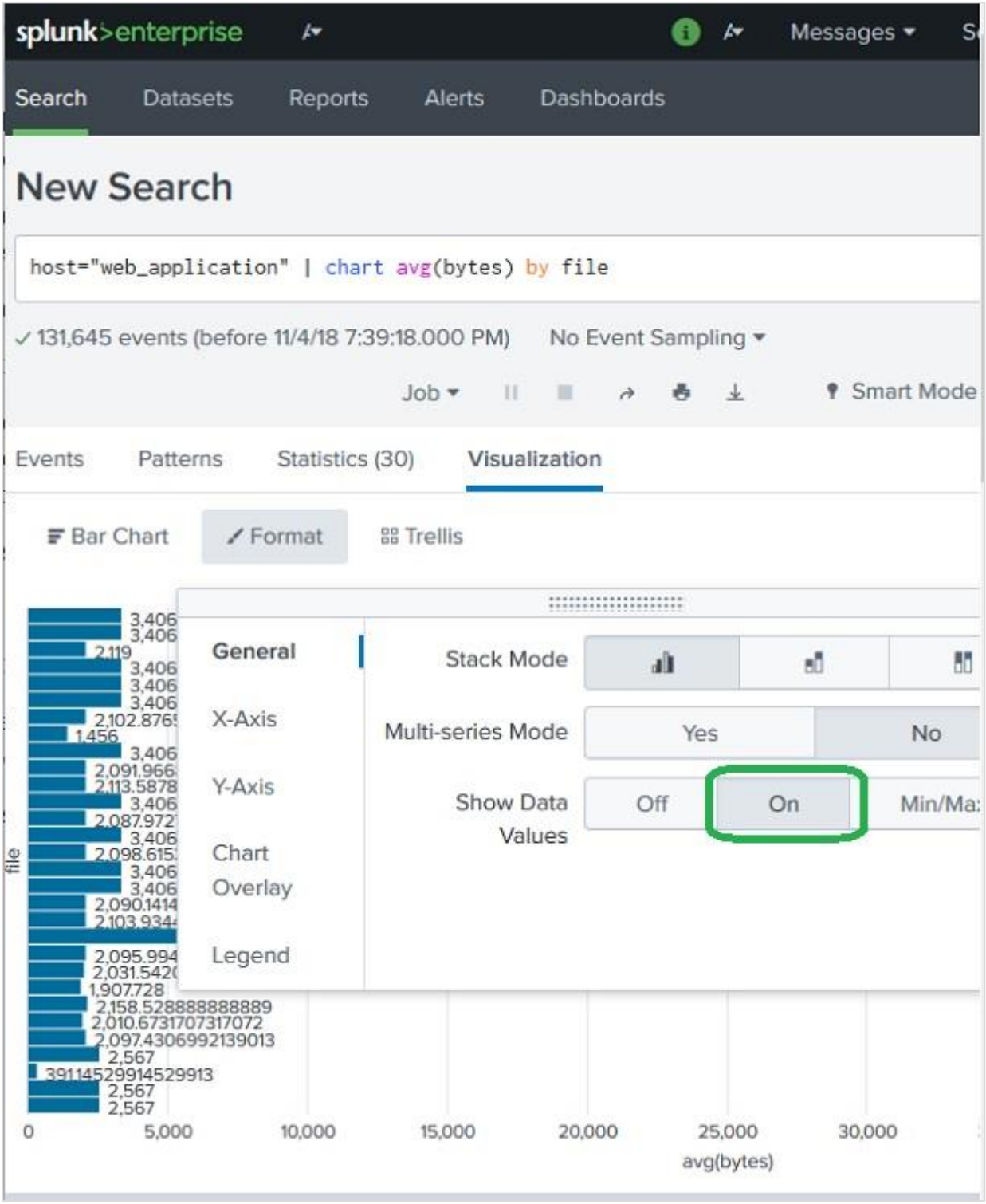
Changing the Chart Type

We can change the chart type by selecting a different chart option from the chart name. Clicking on one of these options will produce the chart for that type of graph.



Formatting a Chart

The charts can also be formatted by using the Format option. This option allows to set the values for the axes, set the legends or show the data values in the chart. In the below example, we have chosen the horizontal chart and selected the option to show the data values as a Format option.

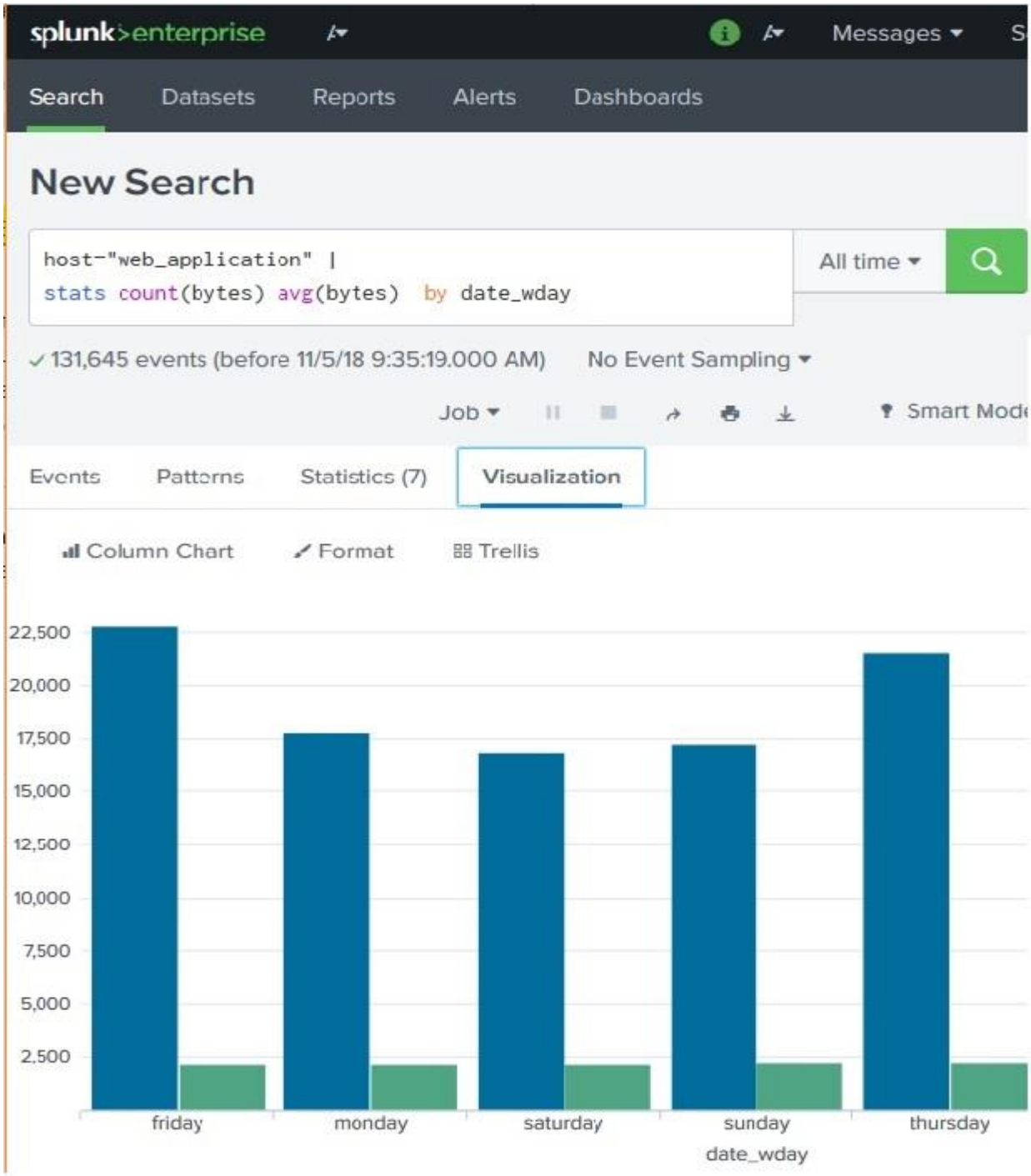


23. Splunk – Overlay Chart

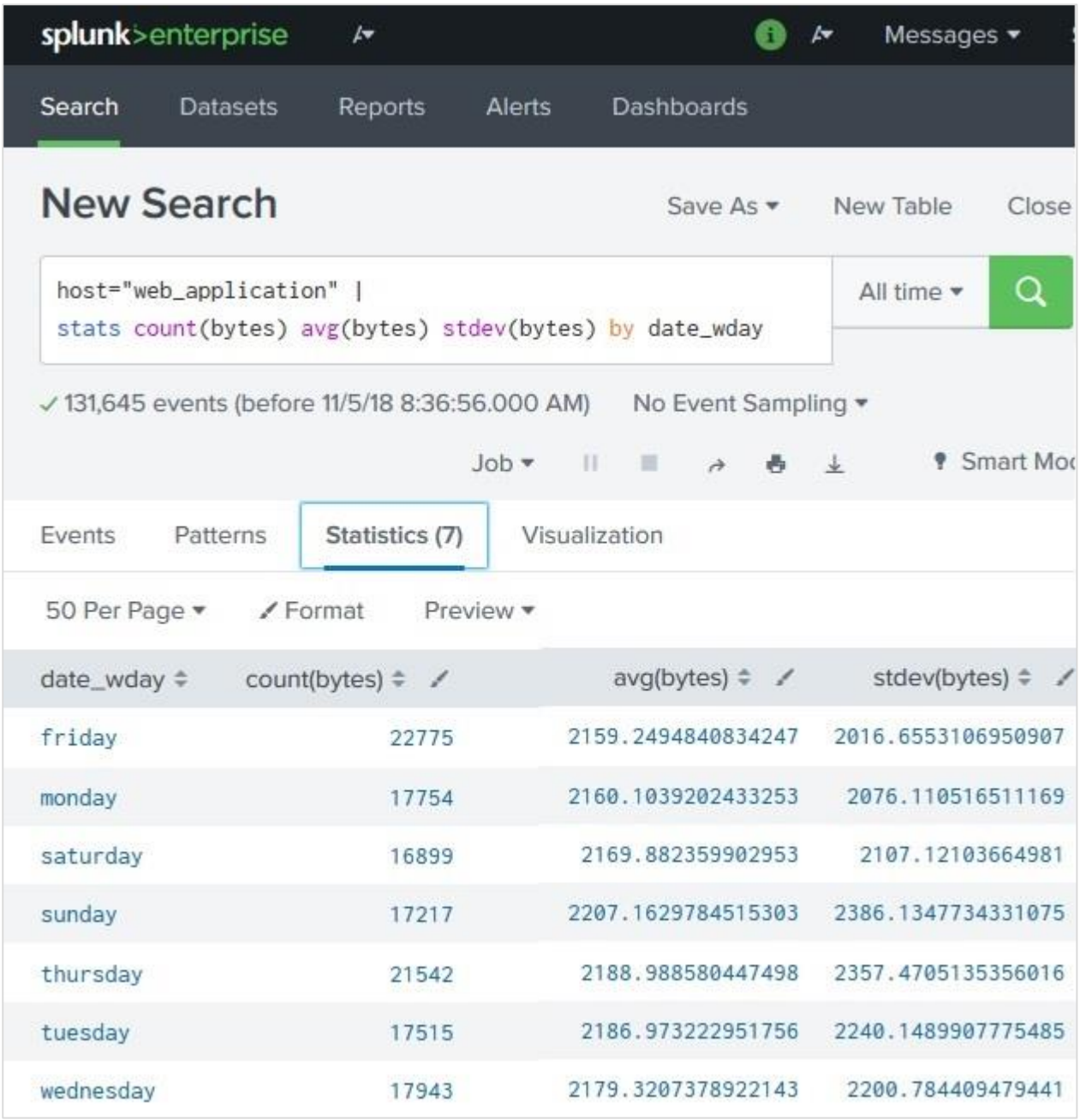
Many times, we need to put one chart over another to compare or see the trend of the two charts. Splunk supports this feature through the chart overlay feature available in its visualization tab. To create such a chart, we need to first make a chart with two variables and then add a third variable which can create the overlay chart.

Chart Scenario

Continuing the examples from previous chapter, we find out the byte size of the files on different week days and then also add the average byte size for those days. The below image shows the chart showing the byte size versus average byte size of files on different days of the week.



Next, we are going to add the statistical function called standard deviation to the above search query. This will bring the additional variable needed to create the chart overlay. The below image shows the statistics of the query result which will be used in the visualization.

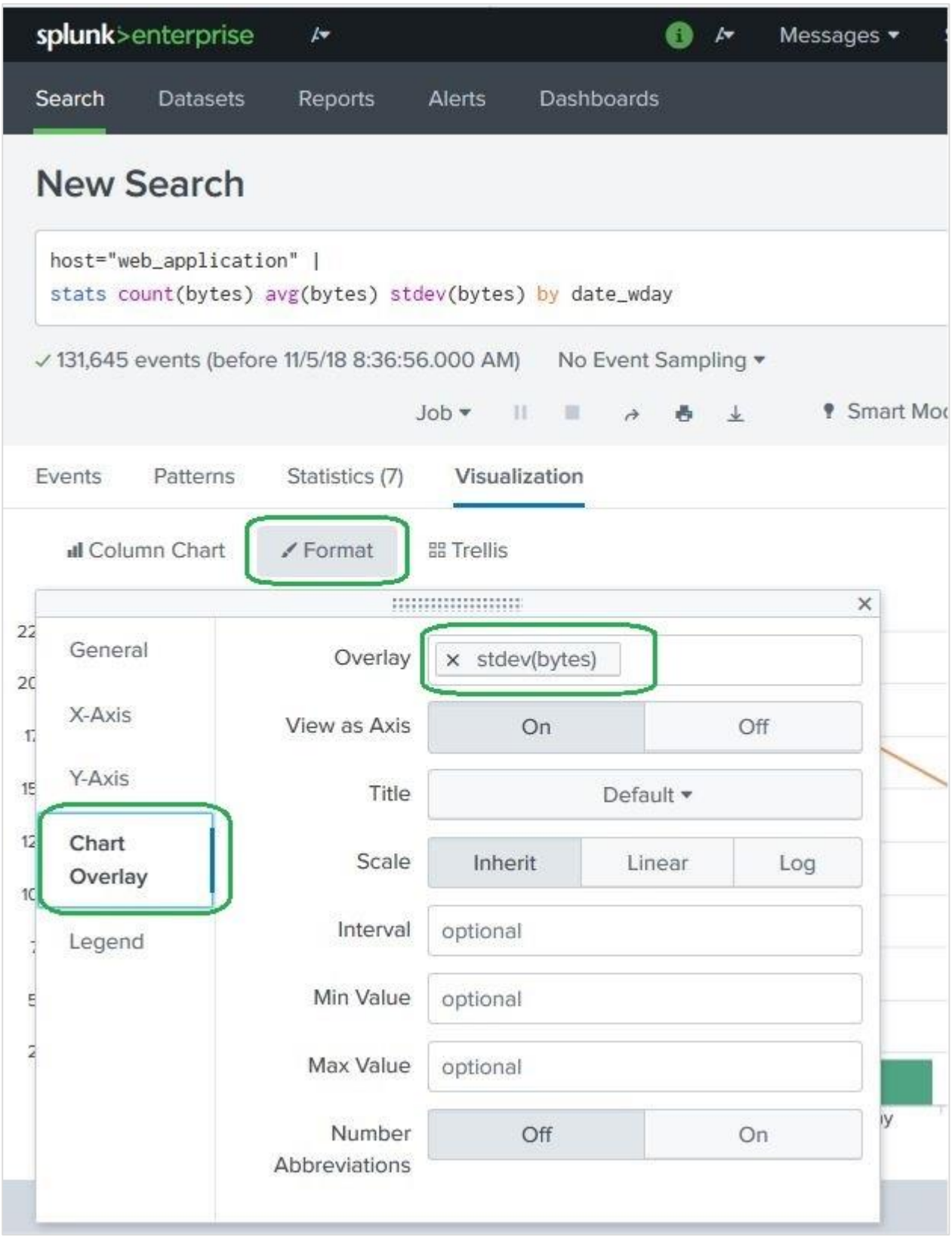


date_wday	count(bytes)	avg(bytes)	stdev(bytes)
friday	22775	2159.2494840834247	2016.6553106950907
monday	17754	2160.1039202433253	2076.110516511169
saturday	16899	2169.882359902953	2107.12103664981
sunday	17217	2207.1629784515303	2386.1347734331075
thursday	21542	2188.988580447498	2357.4705135356016
tuesday	17515	2186.973222951756	2240.1489907775485
wednesday	17943	2179.3207378922143	2200.784409479441

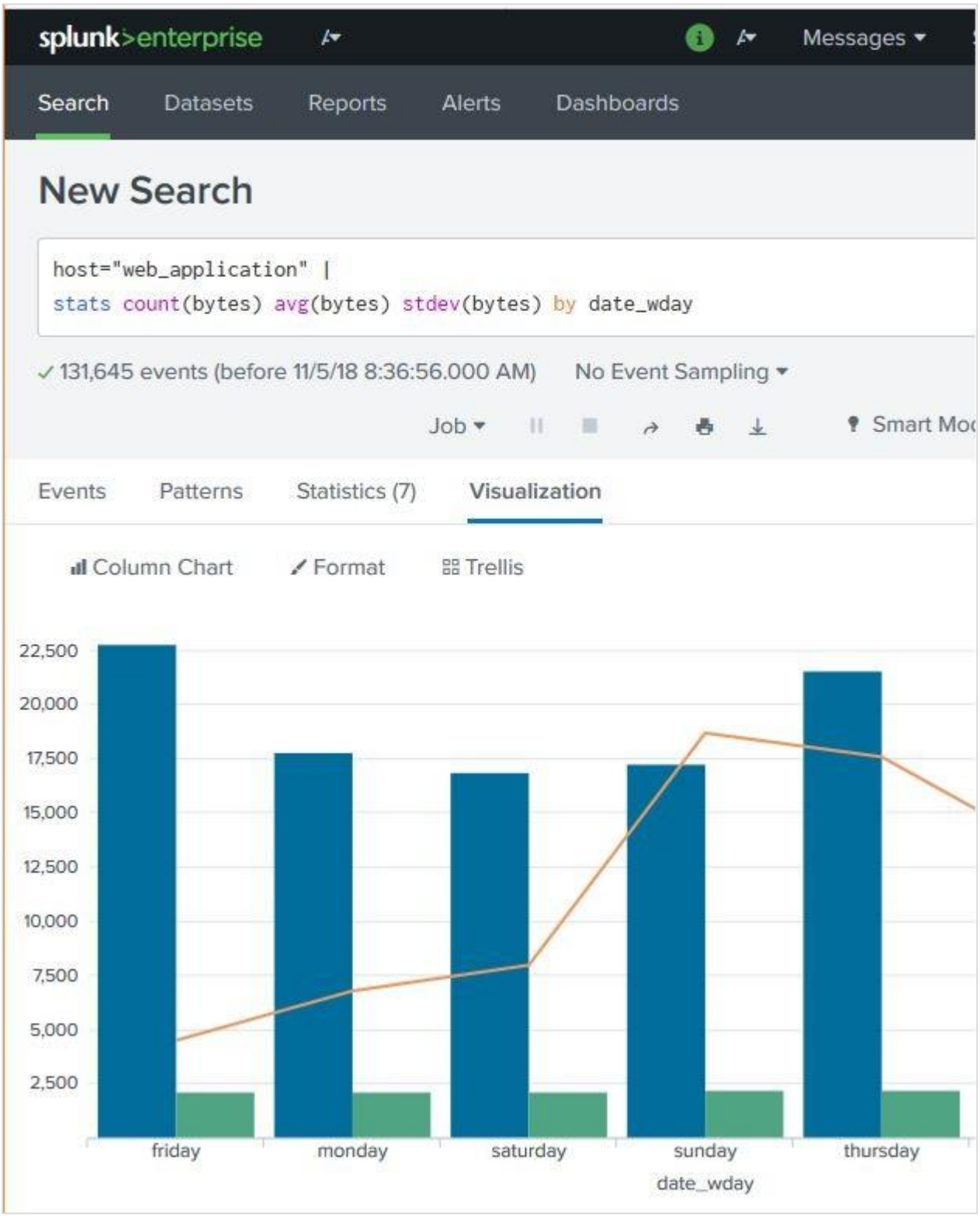
Creating Chart Overlay

To create the chart overlay, we follow **Visualization -> Format -> Chart Overlay**

This brings up a pop-up window where we need to choose the field which will be the overlay chart. In this case, we choose stdev(bytes) as the field as shown in the image below. We can also fill in other values: title, scale and their intervals, minimum values, maximum values, etc. For our example, we choose the default values after selecting the field for the overlay option.



After selecting the above options, we can close the chart overlay pop-up window and see the final chart as shown below:

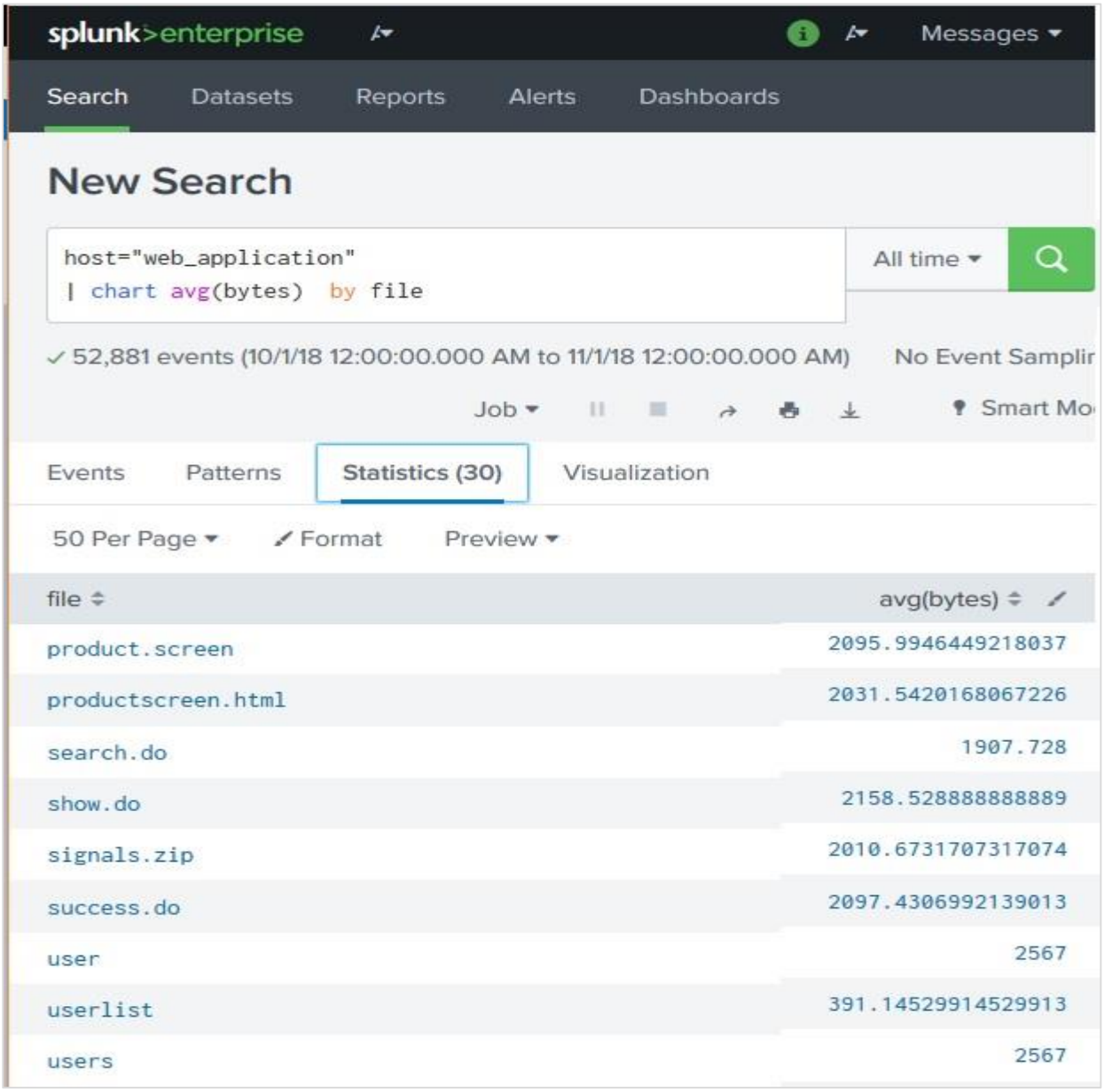


24. Splunk – Sparklines

A sparkline is a small representation of some statistical information without showing the axes. It generally appears as a line with bumps just to indicate how certain quantity has changed over a period of time. Splunk has in-built function to create sparklines from the events it searches. It is a part of the chart creation function.

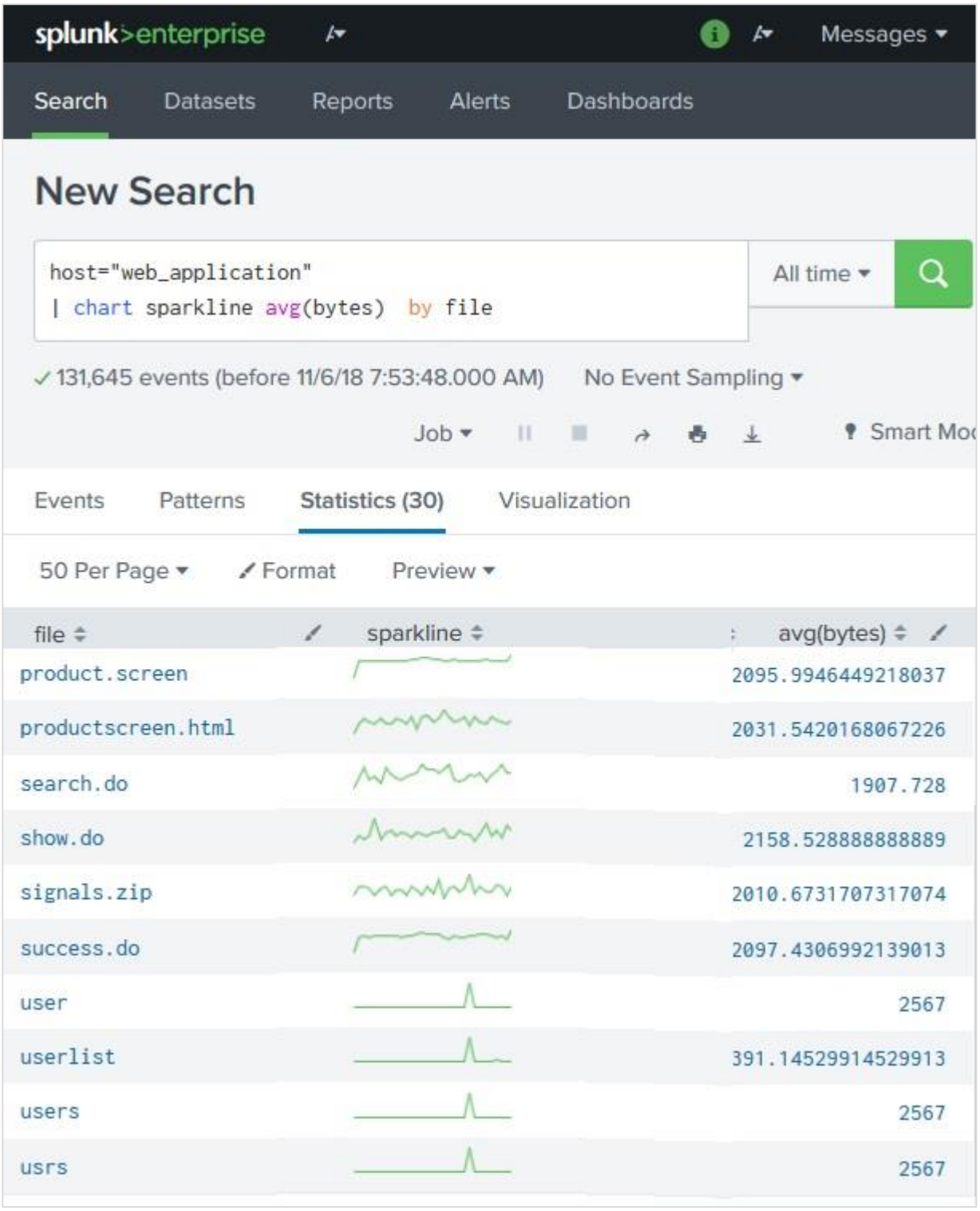
Selecting the Fields

We need to select the field and the search formula which will be used in creating the sparkline. The below image shows the average byte size values of the some of the files in the web_application host.



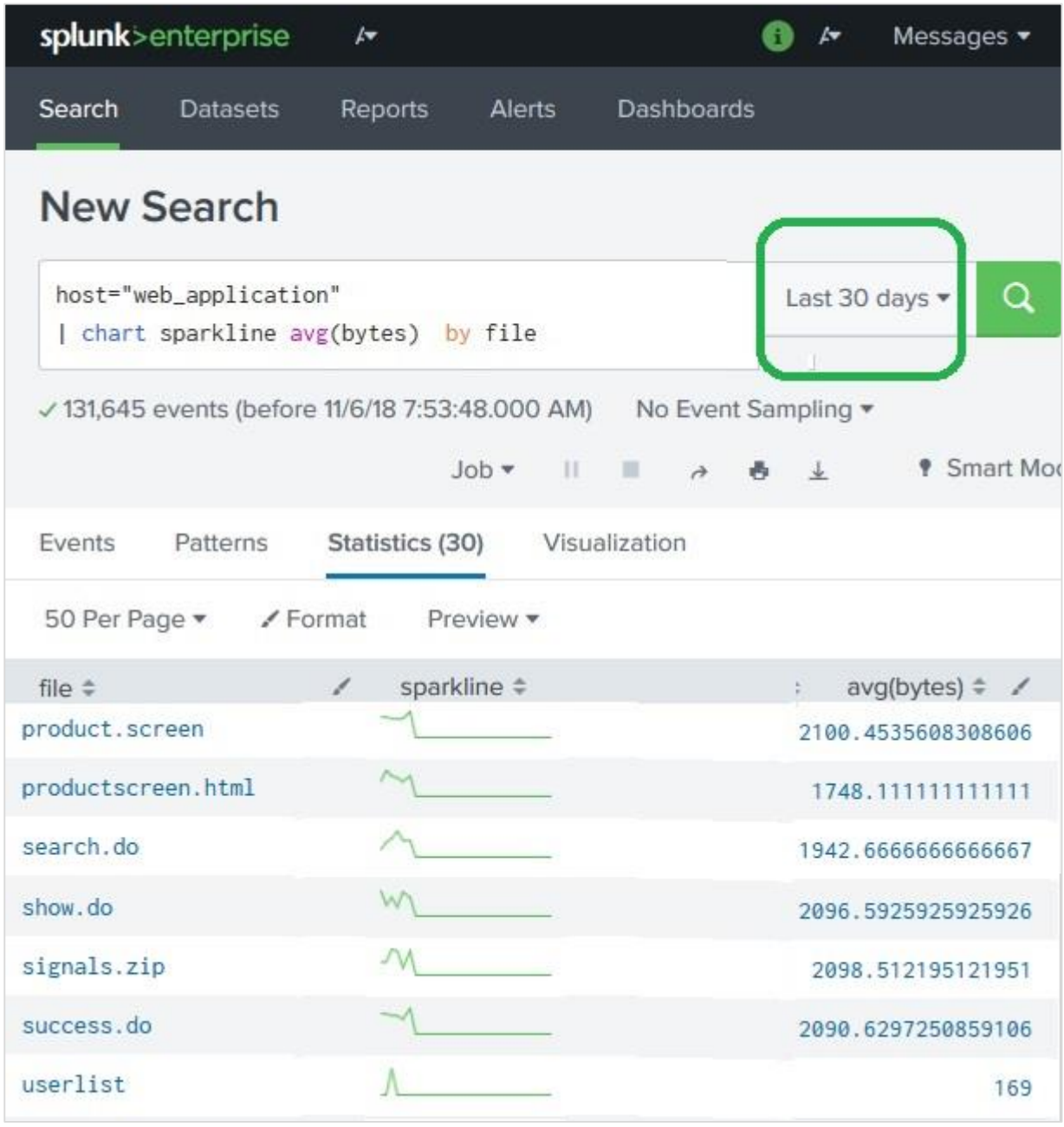
Creating the Sparkline

To create the Sparklines from above statistics, we add the Sparkline function to the search query as shown in the image below. The table view of the above statistics now starts displaying the sparklines for average byte size of those files. Here, we have taken **All Time** as the time period for calculating the variation in average byte size of files. If we change this time period, then the nature of the graphs will change.



Changing the Time Period

If we change the time period for the above graph from All Time to Last 30 days, we will see the sparklines to be little different as shown below. Here we need to note, how few file names have vanished from the list as those files were not available in that time period.



25. Splunk – Managing Indexes

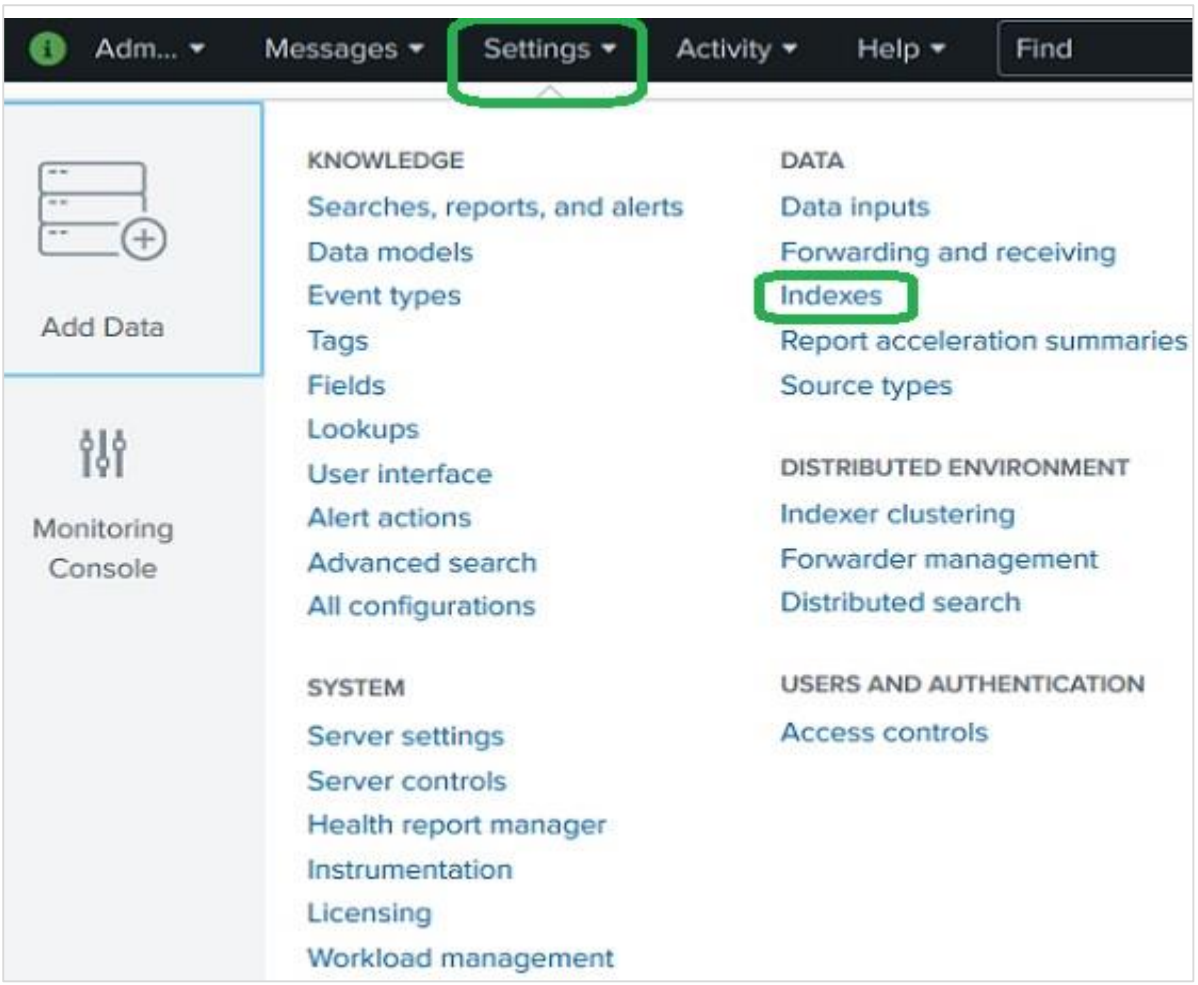
Indexing is a mechanism to speed up the search process by giving numeric addresses to the piece of data being searched. Splunk indexing is similar to the concept of indexing in databases. The installation of Splunk creates three default indexes as follows.

- **main**: This is Splunk's default index where all the processed data is stored.
- **Internal**: This index is where Splunk's internal logs and processing metrics are stored.
- **audit**: This index contains events related to the file system change monitor, auditing, and all user history.

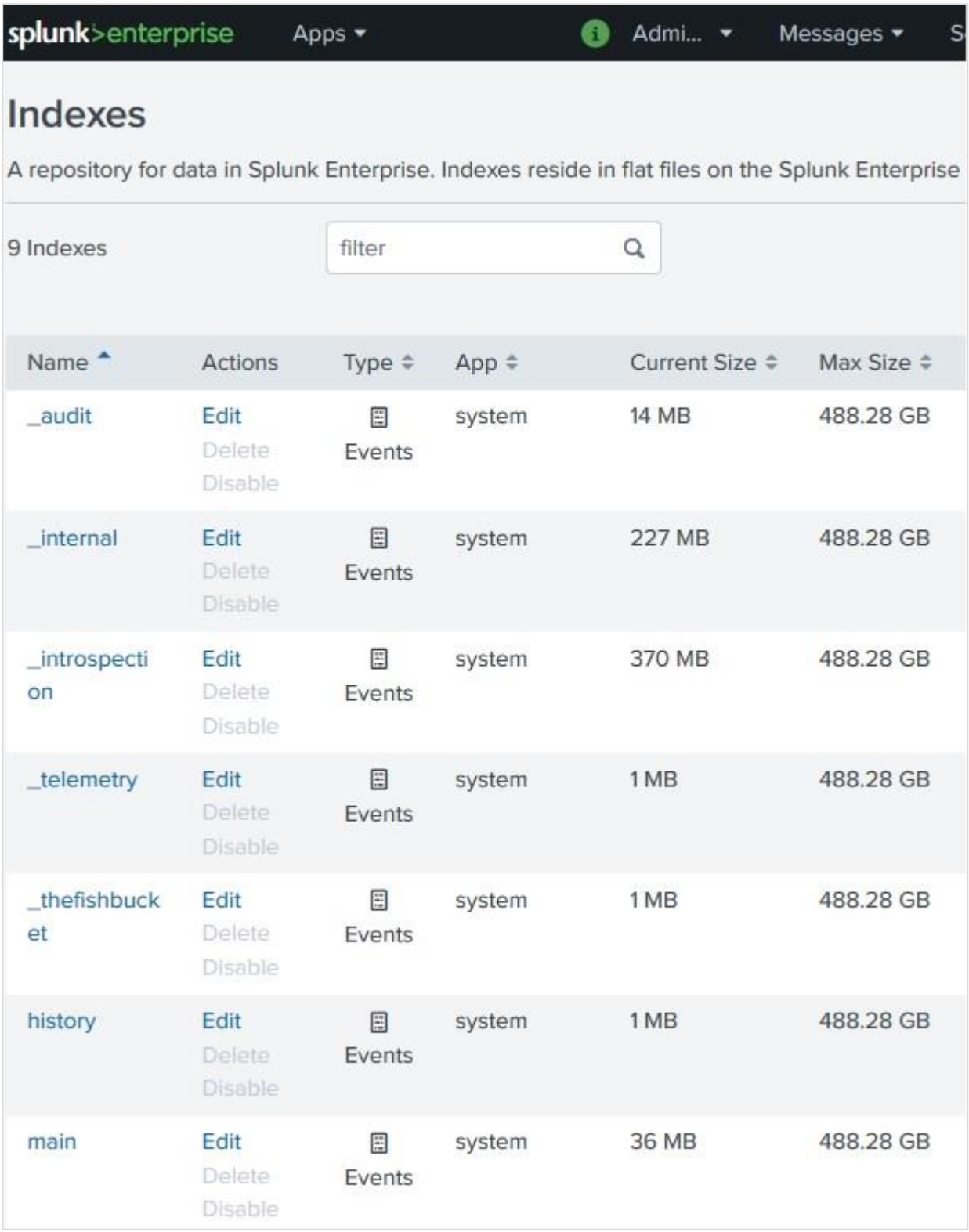
The Splunk Indexers create and maintain the indexes. When you add data to Splunk, the indexer processes it and stores it in a designated index (either, by default, in the main index or in the one that you identify).

Checking Indexes

We can have a look at the existing indexes by going to **Settings -> Indexes** after logging in to Splunk. The below image shows the option.



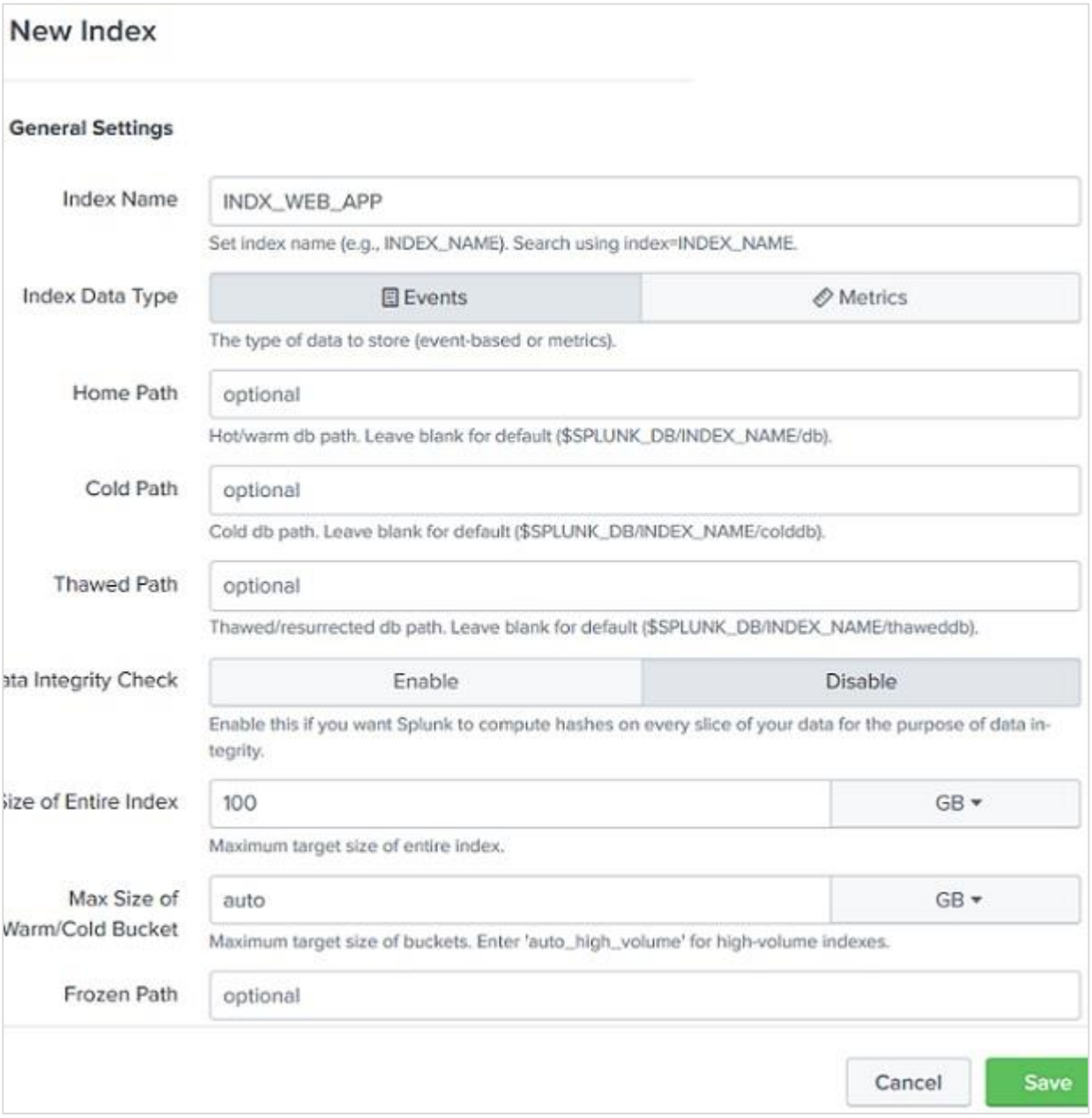
On further clicking on the indexes, we can see the list of indexes Splunk maintains for the data that is already captured in Splunk. The below image shows such a list.



Creating a New Index

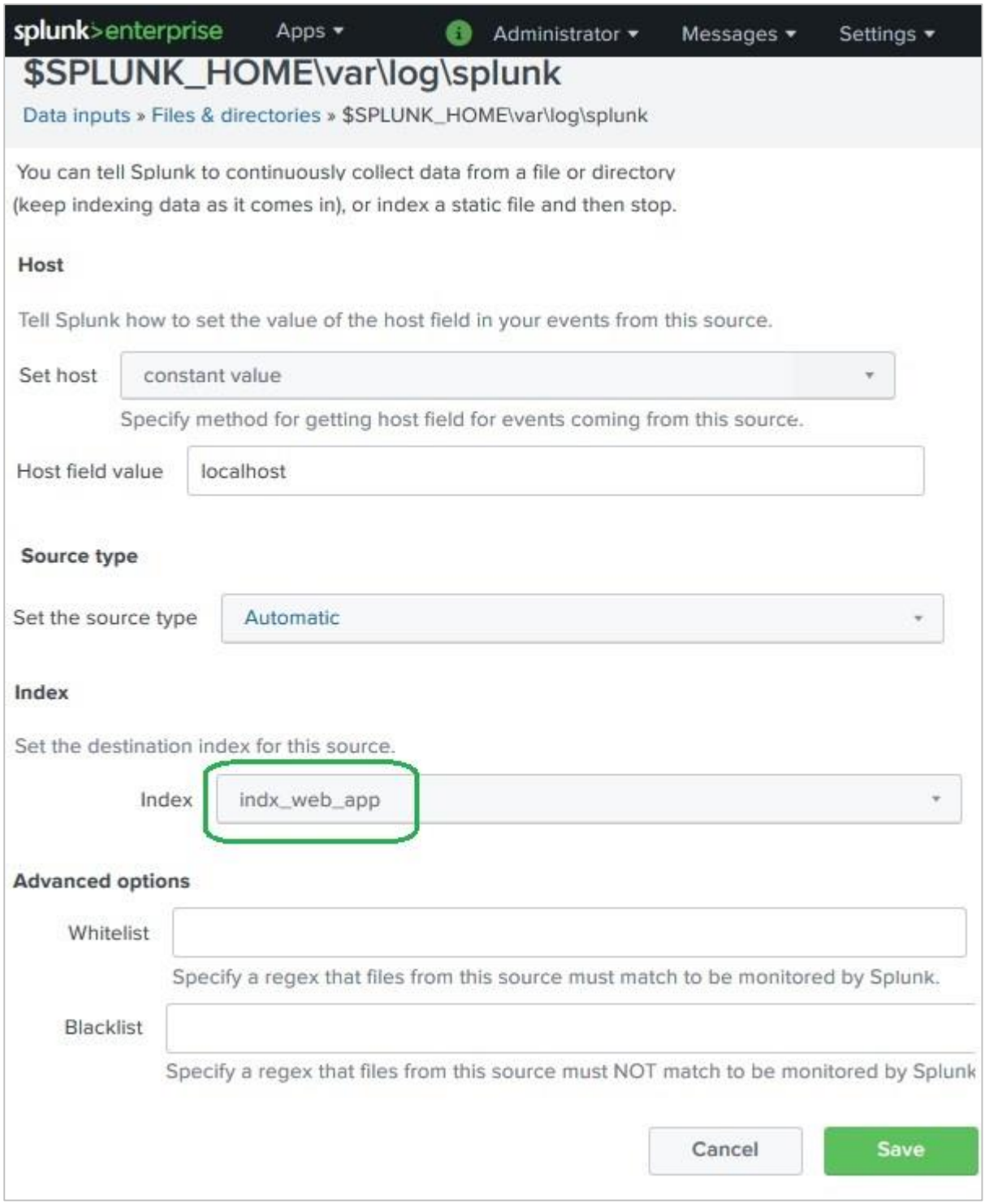
We can create a new index with desired size by the data that is stored in Splunk. The additional data that comes in can use this newly created index but better search functionality. The steps to create an index is **Settings -> Indexes -> New Index**. The

below screen appears where we mention the name of the index and memory allocation etc.



Indexing the Events

After creating the index above we can configure the events to be indexed by this specific index. We choose the event type. Use the path **Settings -> Data Inputs -> Files & Directories.** Then we choose the specific file of the events which we want to attach to the newly created event. As you can see in the below image, we have assigned the index named **index_web_app** to this specific file.



26. Splunk – Calculated Fields

Many times, we will need to make some calculations on the fields that are already available in the Splunk events. We also want to store the result of these calculations as a new field to be referred later by various searches. This is made possible by using the concept of calculated fields in Splunk search.

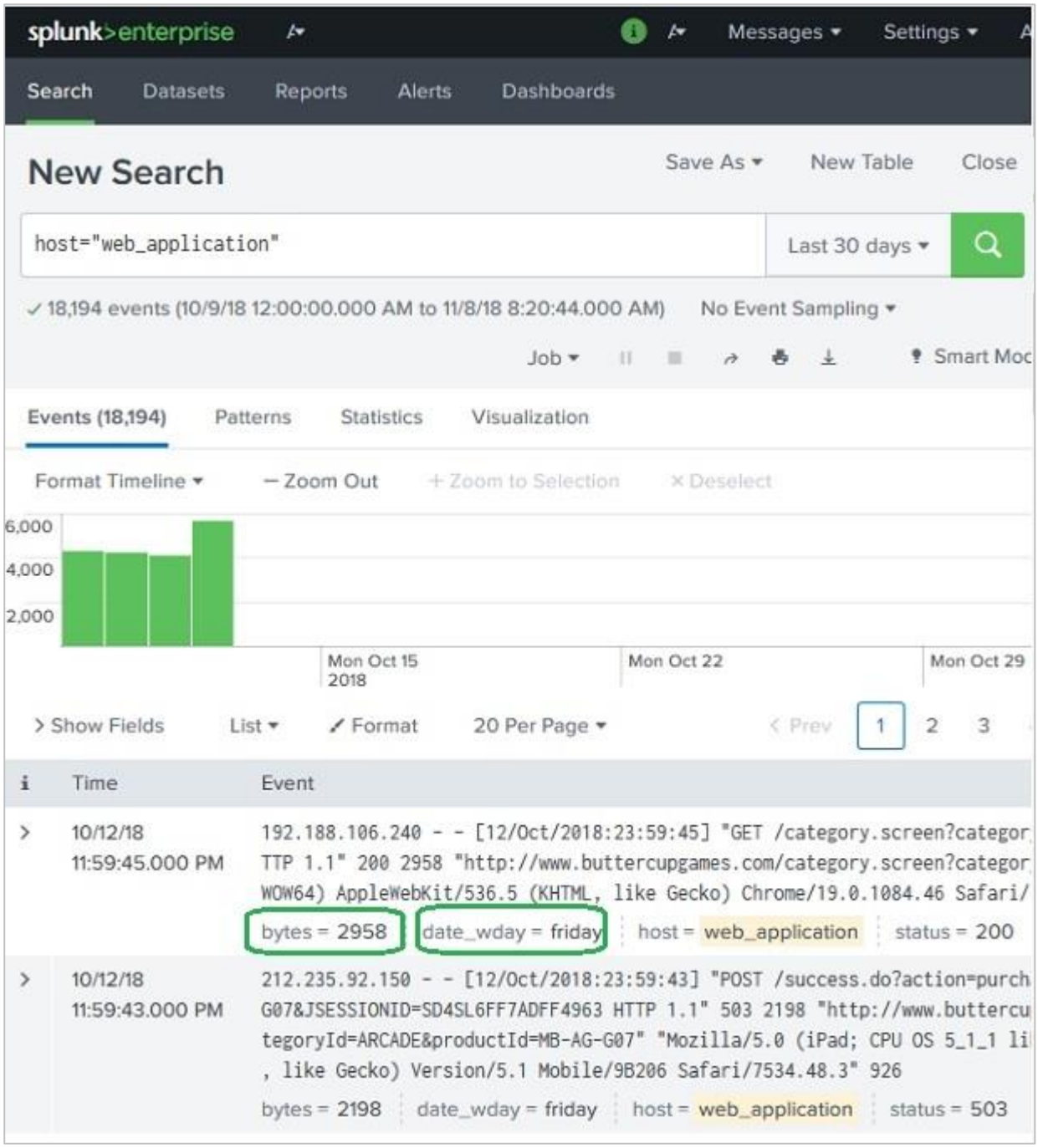
A simplest example is to show the first three characters of a week day instead of the complete day name. We need to apply certain Splunk function to achieve this manipulation of the field and store the new result under a new field name.

Example

The Web_application log file has two fields named bytes and date_wday. The value in the bytes field is the number of bytes. We want to display this value as GB. This will require the field to be divided by 1024 to get the GB value. We need to apply this calculation to the bytes field.

Similarly, the date_wday displays complete name of the week day. But we need to display only the first three characters.

The existing values in these two fields is shown in the image below:



Using the eval Function

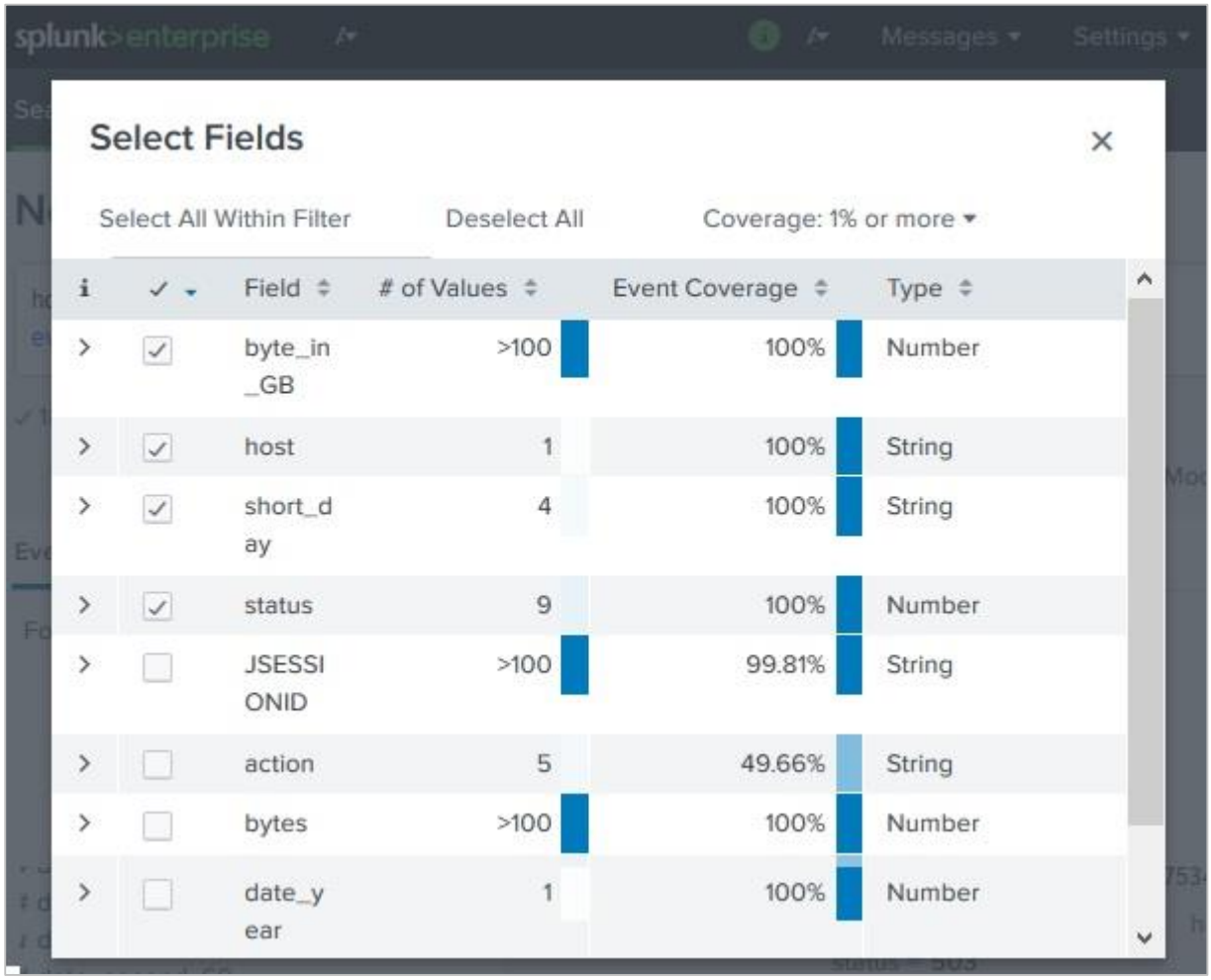
To create calculated field, we use the eval function. This function stores the result of the calculation in a new field. We are going to apply the below two calculations:

```
# divide the bytes with 1024 and store it as a field named byte_in_GB

Eval byte_in_GB = (bytes/1024)
```

```
# Extract the first 3 characters of the name of the day.
Eval short_day=substr(date_wday,1,3)
```

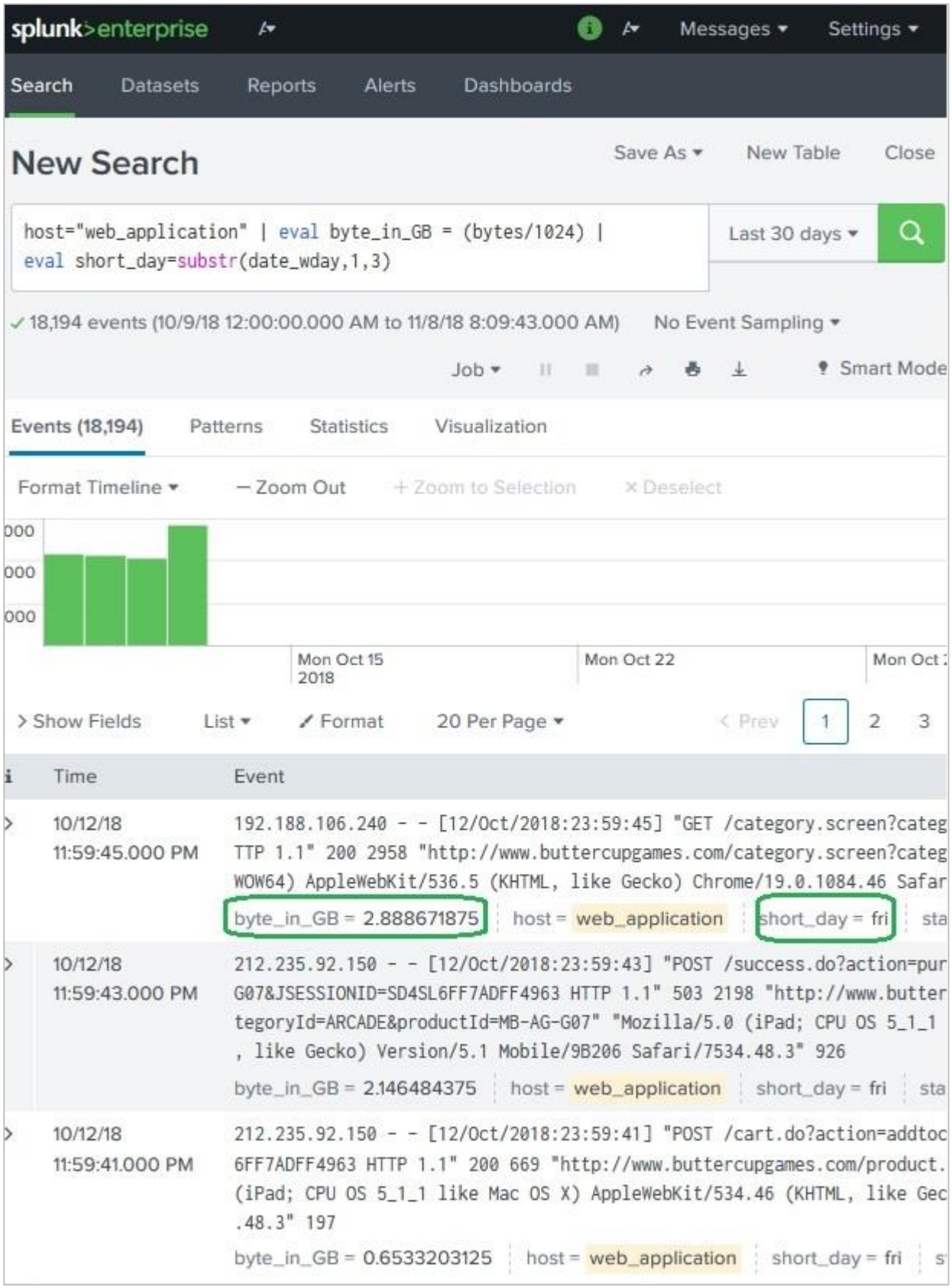
Adding New Fields

We add new fields created above to the list of fields we display as part of the search result. To do this, we choose **All fields** options and tick check mark against the name of these new fields as shown in below image:



Displaying the calculated Fields

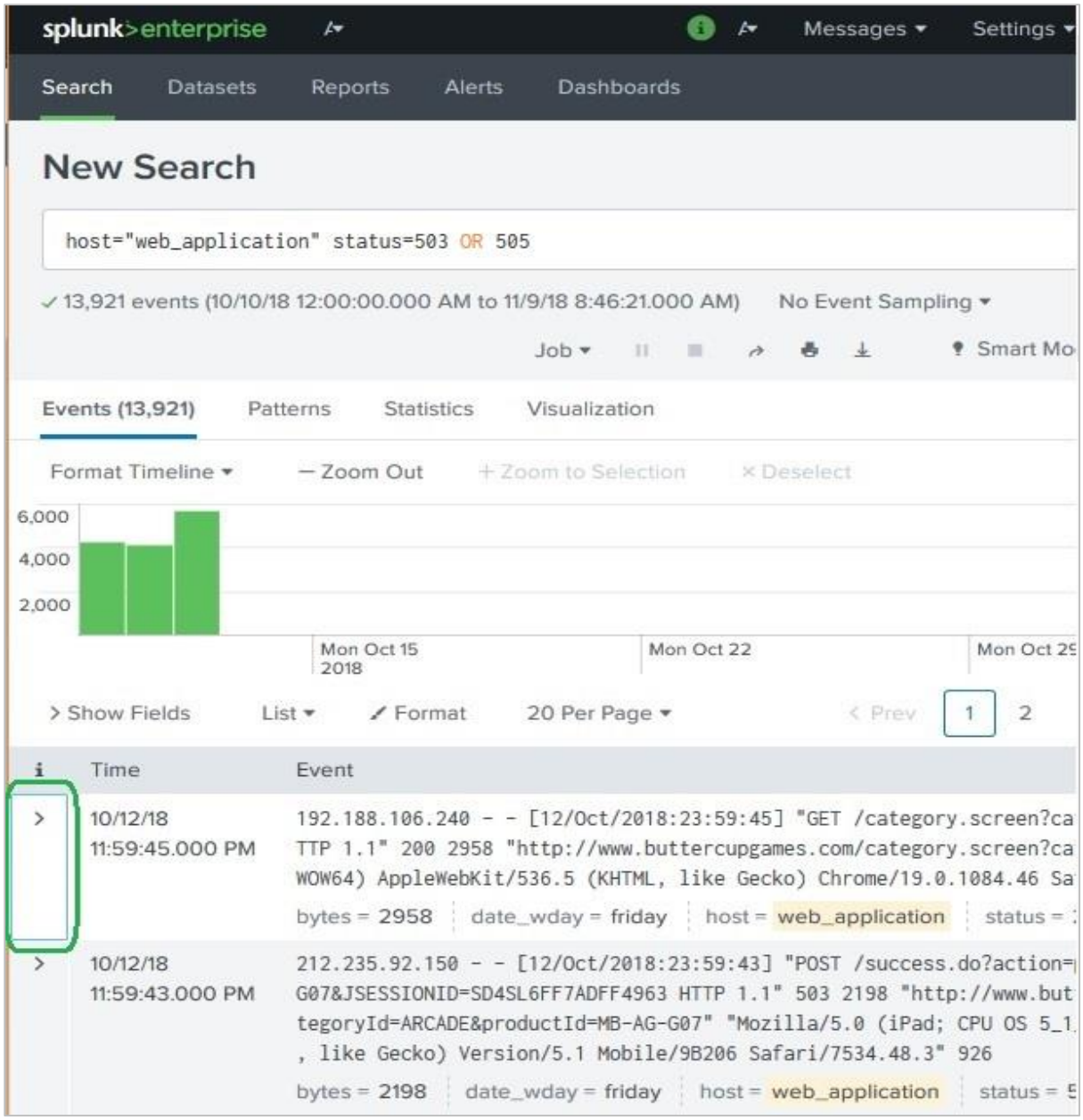
After choosing the fields above, we are able to see the calculated fields in the search result as shown below. The search query displays the calculated fields as shown below:



27. Splunk – Tags

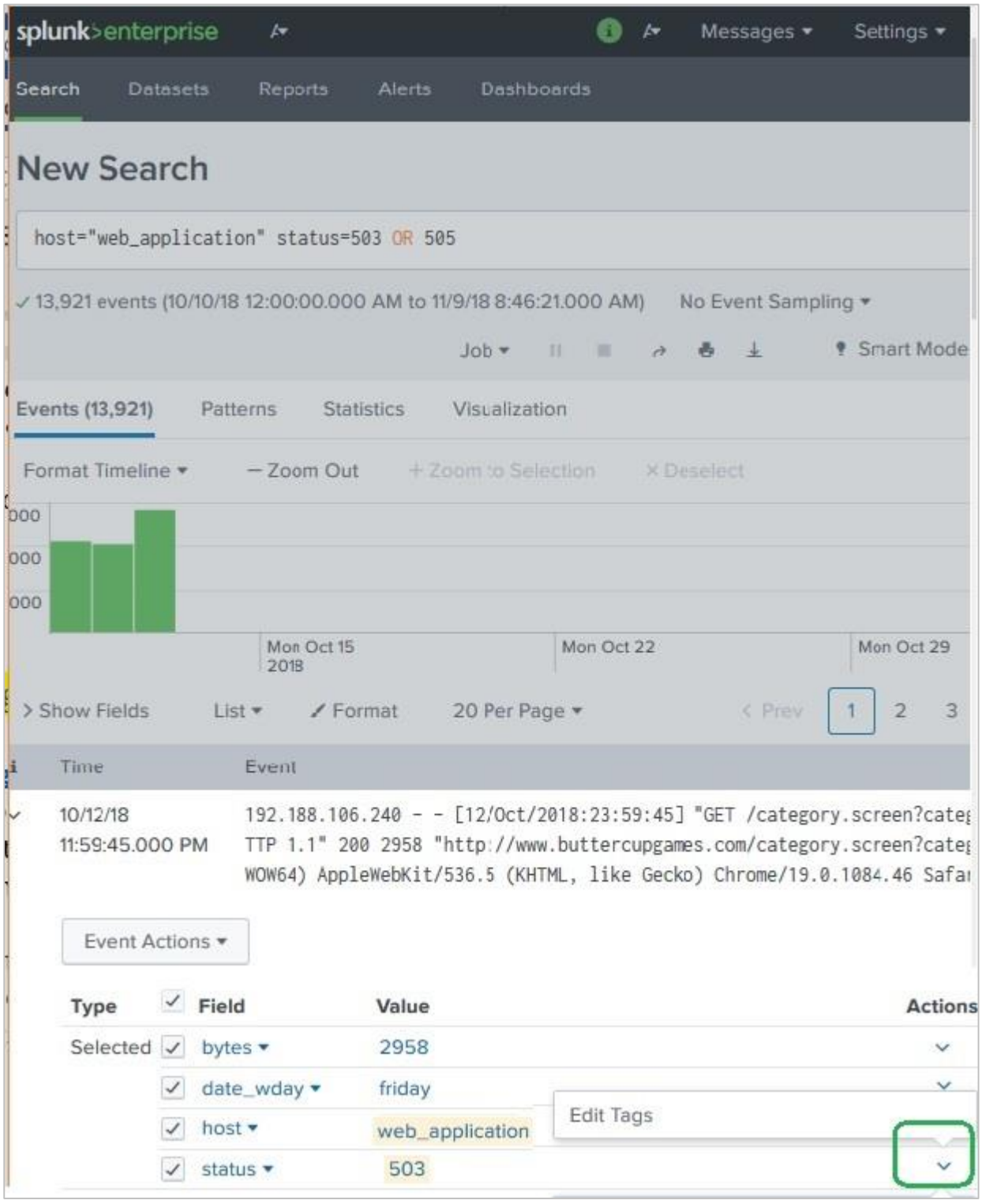
Tags are used to assign names to specific field and value combinations. These fields can be event type, host, source, or source type, etc. You can also use a tag to group a set of field values together, so that you can search for them with one command. For example, you can tag all the different files generated on Monday to a tag named mon_files.

To find the field-value pair which we are going to tag, we need to expand the events and locate the field to be considered. The below image shows how we can expand an event to see the fields:



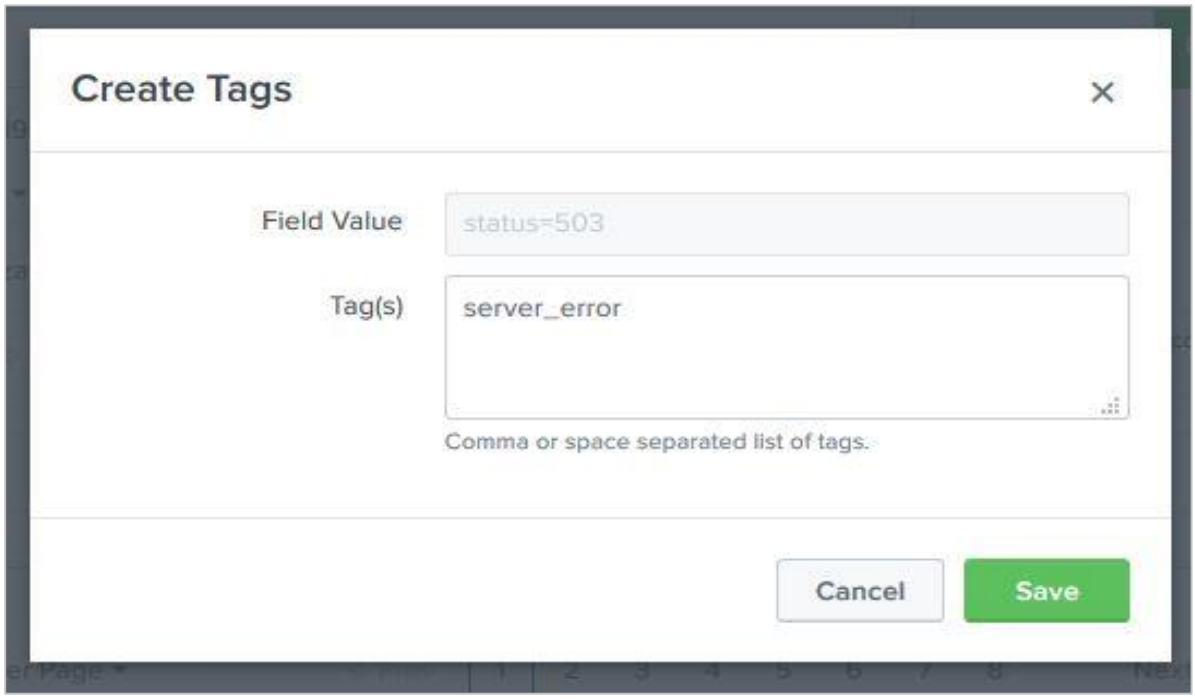
Creating Tags

We can create tags by adding the tag value to field-value pair using **Edit Tags** option as shown below. We choose the field under the Actions column.



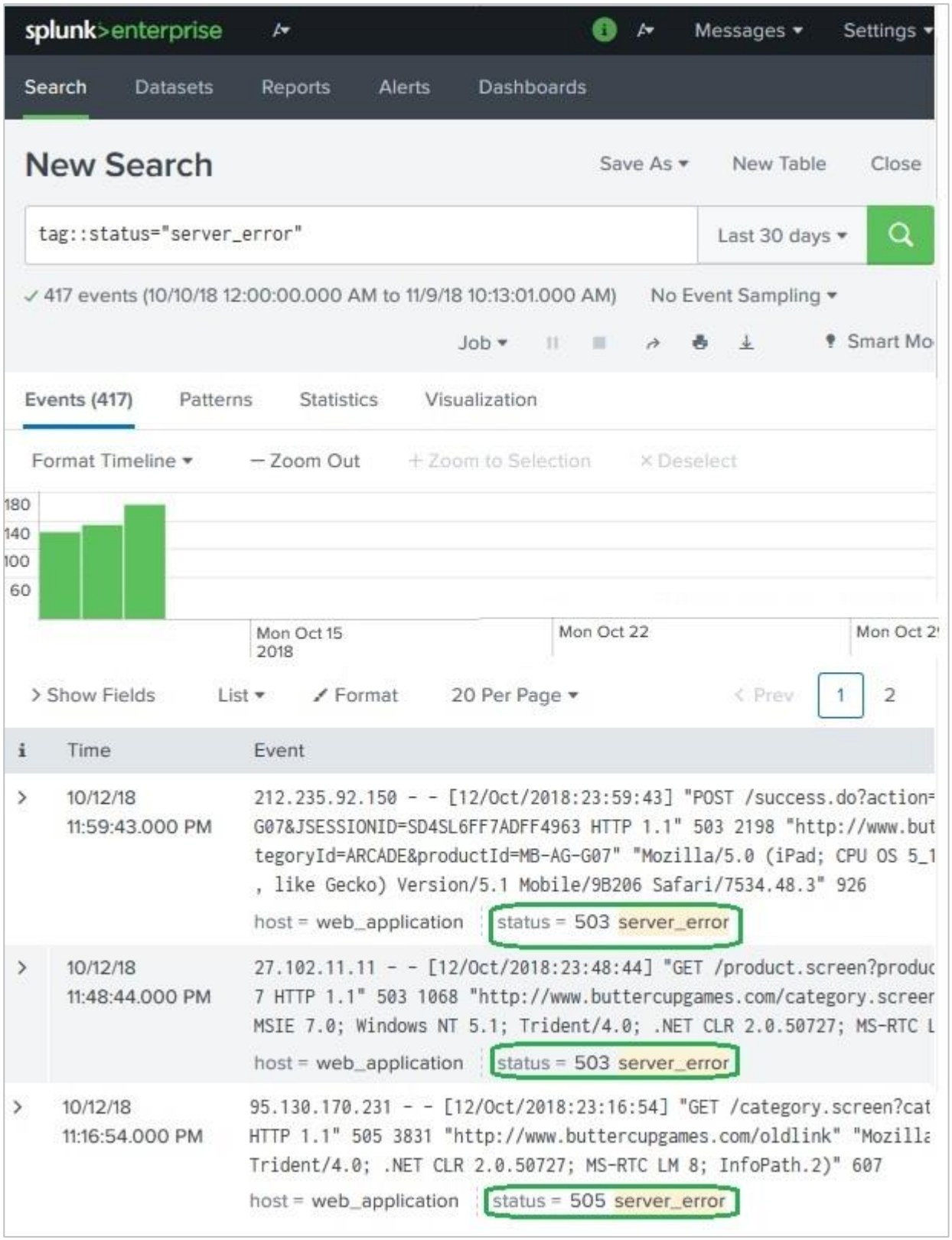
The next screen prompts us to define the tag. For the Status field, we choose the status value of 503 or 505 and assign a tag named server_error as shown below. We have to do

it one by one by choosing two events, each with the events with status value 503 and 505. The image below shows the method for status value as 503. We have to repeat the same steps for an event with status value as 505.



Search Using Tags

Once the tags are created, we can search for events containing the Tag by simply writing the Tag name in the search bar. In the below image, we see all the events which have status: 503 or 505.



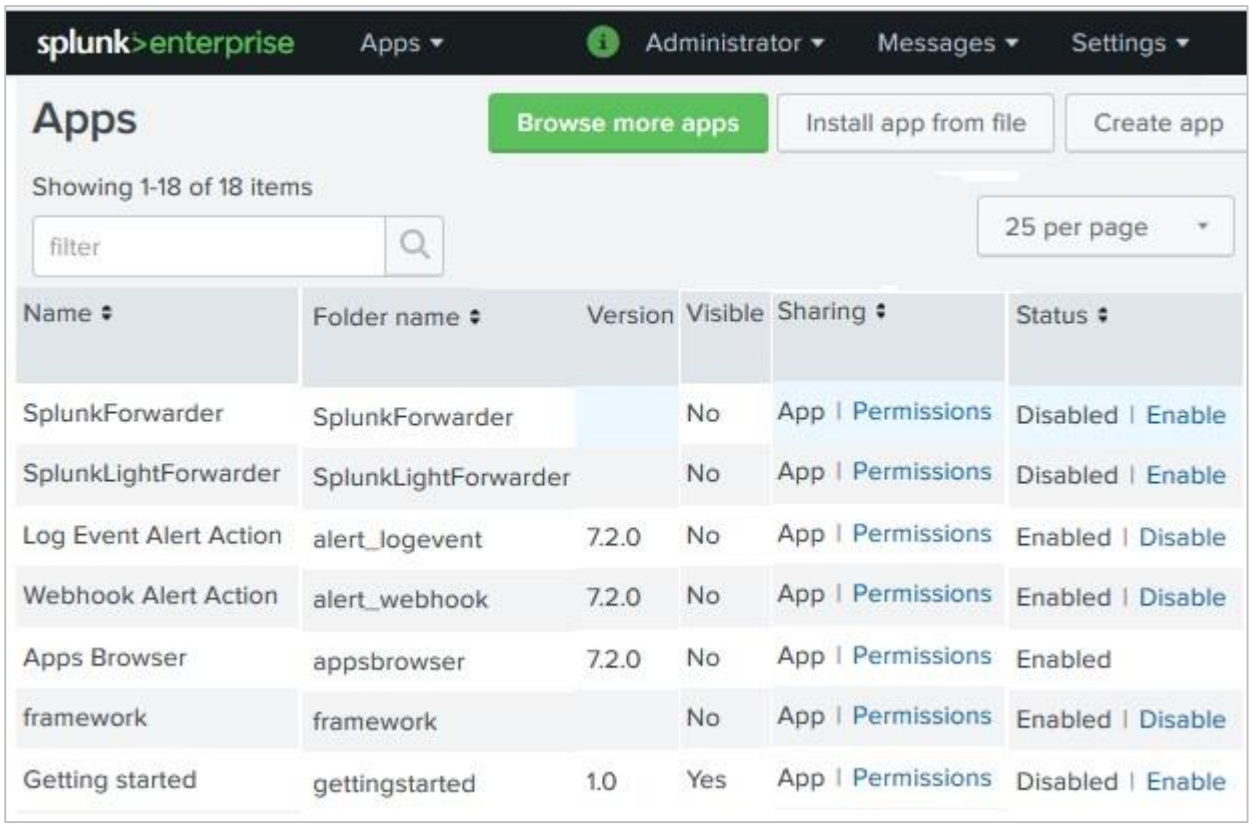
28. Splunk – Apps

A Splunk app is an extension of Splunk functionality which has its own in-built UI context to serve a specific need. Splunk apps are made up of different Splunk knowledge objects (lookups, tags, eventtypes, savedsearches, etc). Apps themselves can utilize or leverage other apps or add-ons. Splunk can run any number of apps simultaneously.

When you log in to Splunk, you land on an app which is typically, the **Splunk Search app**. So, almost everytime you are inside the Splunk interface, you are using an app.

Listing Splunk Apps

We can list the available apps in Splunk by using the option **Apps -> Manage Apps**. Navigating this option brings out the following screen which lists the existing apps available in Splunk interface.



Following are important values associated with the Splunk apps:

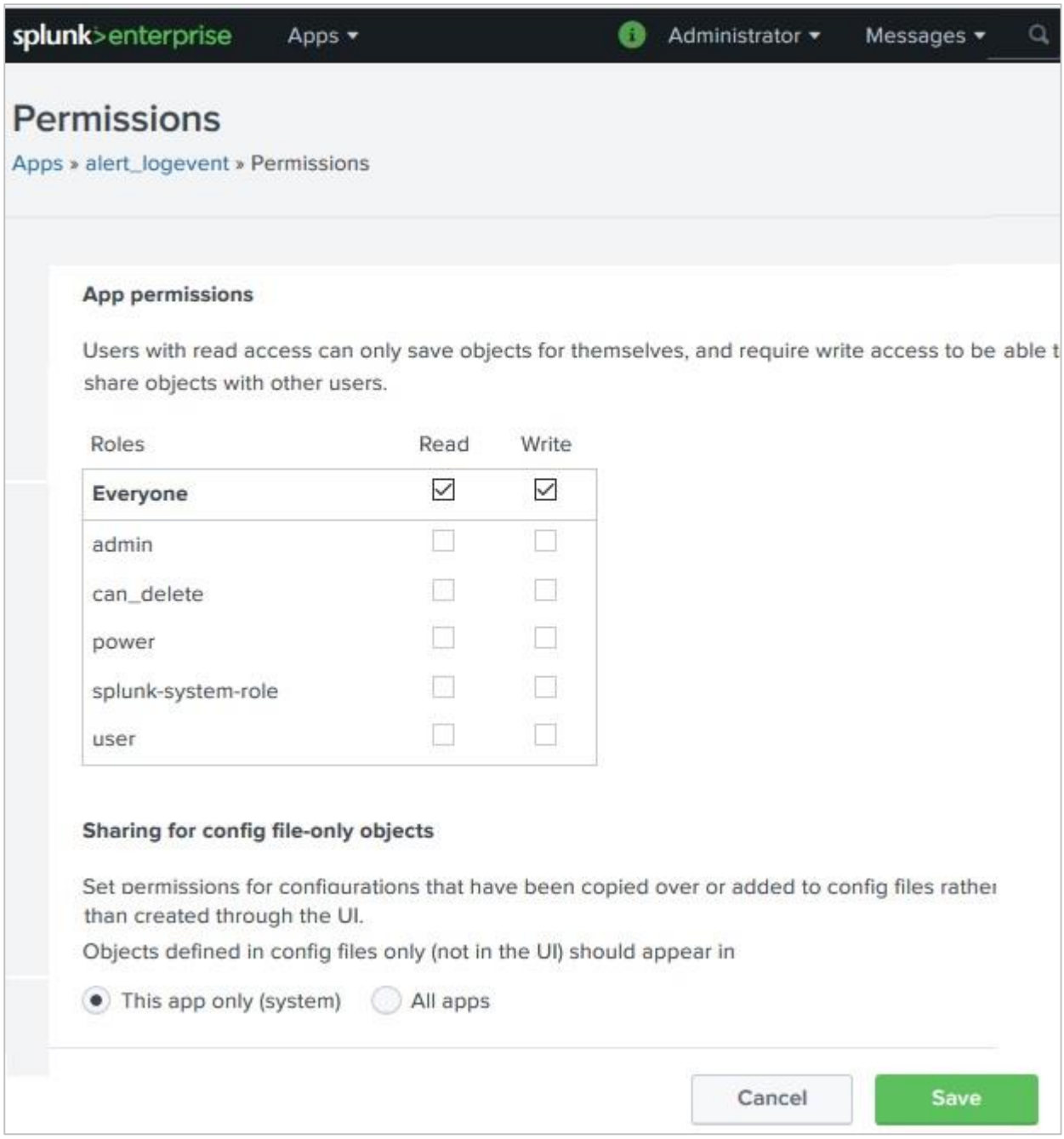
- **Name:** It is the name of the App and unique for each App.
- **Folder Name**: It is the name to use for the directory in $SPLUNK_HOME/etc/apps/. The name of the folder cannot contain "dot" (.) character.
- **Version**: It is the app version string. Visible Indicates whether the app should be visible in Splunk Web. Apps that contain a user interface should be visible.

- **Sharing:** It is the level of permissions (read or write) given to different Splunk users for that specific app.

- **Status:** It is the current status of availability of the App. It may be enabled or disabled for use.

App Permissions

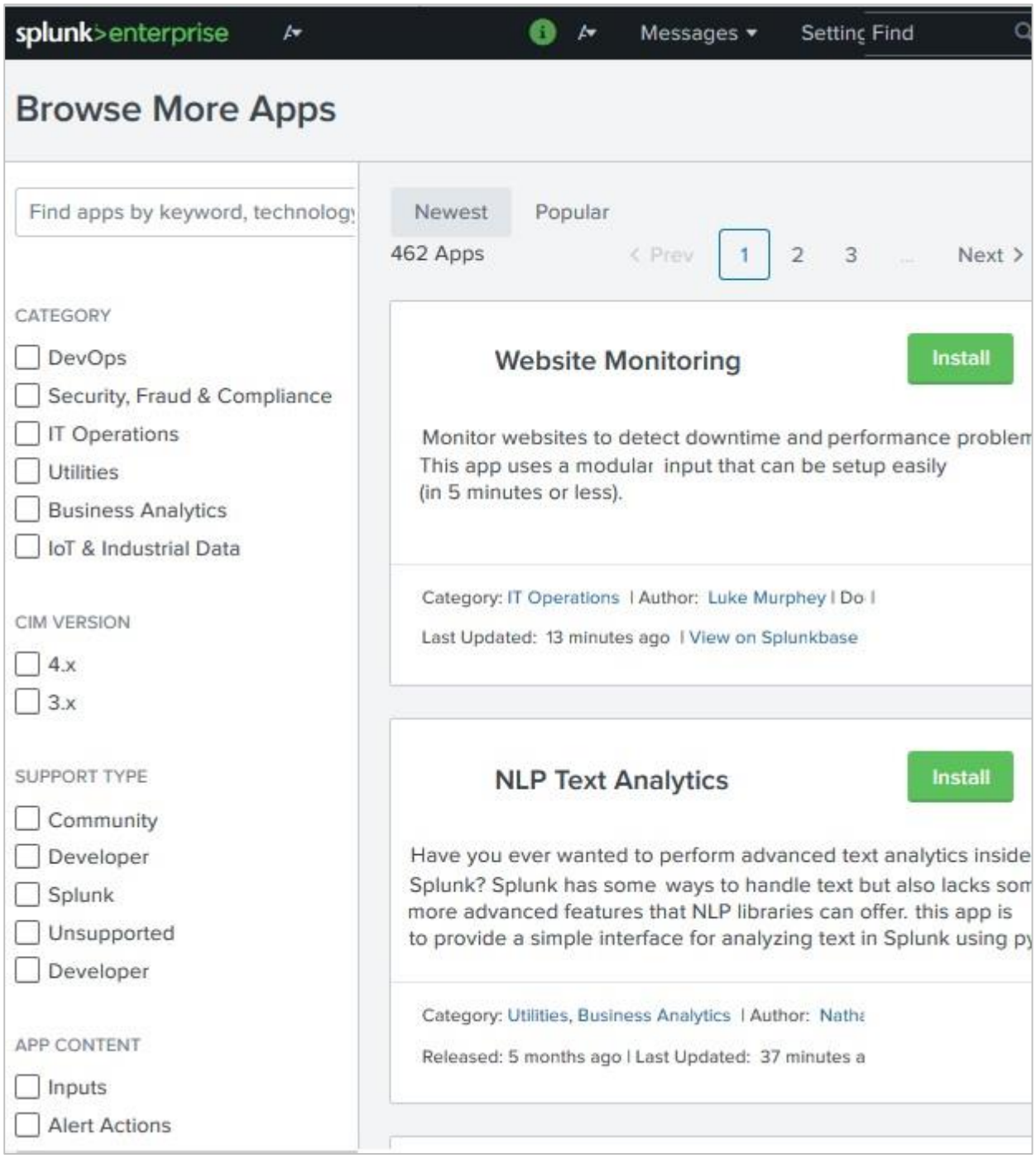
A proper setting of permissions for using the app is important. We can restrict the app to be used by a single user or by multiple users including all users. The below screen which appears after clicking on the permissions link in the above is used to modify the access to different roles.



By default, the check marks for Read and Write option is available for Everyone. But we can change that by going to each role and selecting appropriate permission for that specific role.

App Marketplace

There is a wide variety of needs for which the Splunk search functionalities are used. So, there is a Splunk App market place which has come into existence show casing many different apps created by individual and organizations. They are available in both free and paid versions. We can browse those apps by choosing the option **Apps -> Manage Apps -> Browse More Apps**. The below screen comes up.



As you can see, the App name along with a brief description of the functionality of the App appears. This helps you decide which app to use. Also, note how the Apps are categorized in the left bar to help choose the type of App faster.

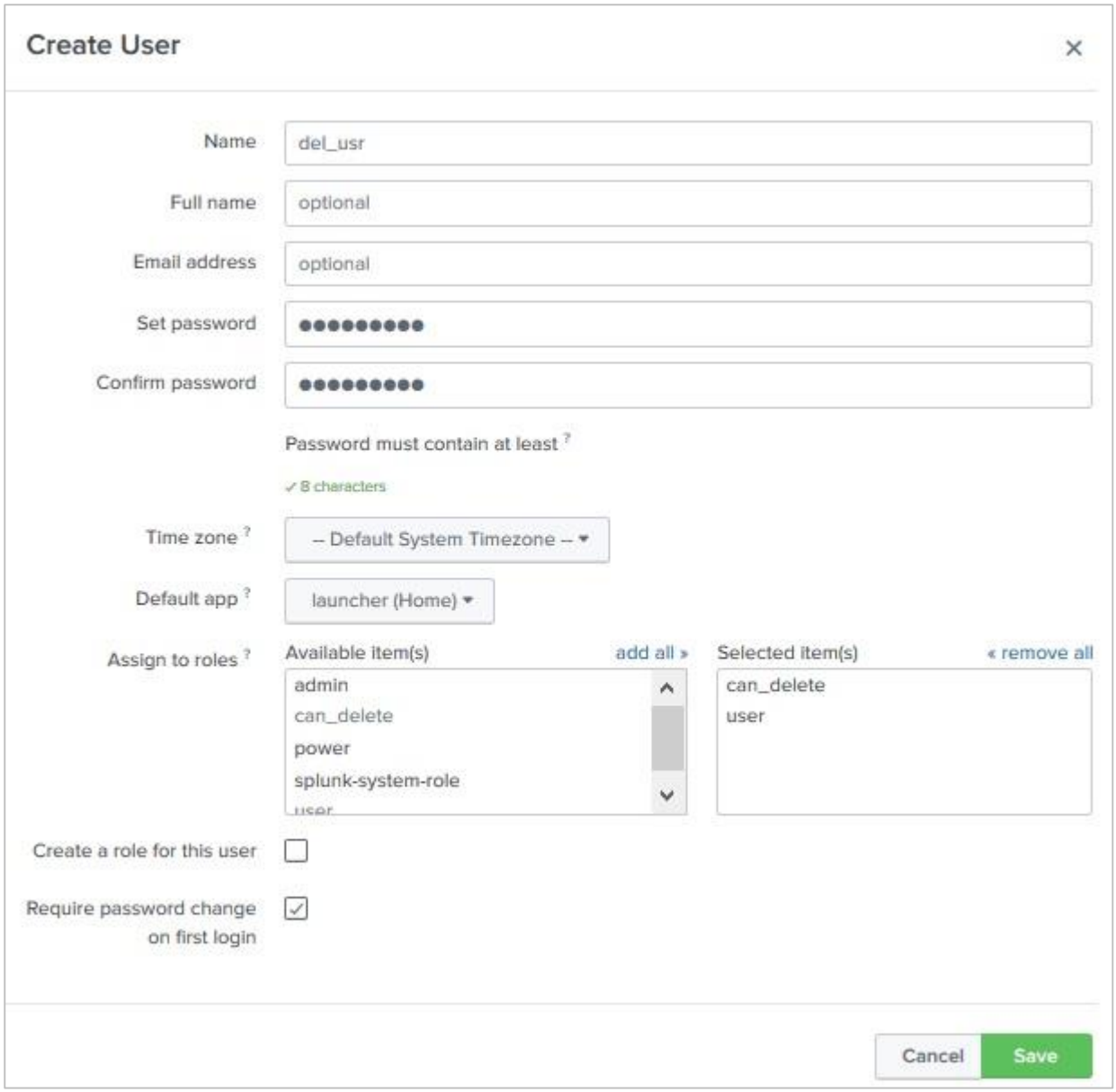
29. Splunk – Removing Data

Removing data from Splunk is possible by using the **delete** command. We first create the search condition to fetch the events we want to mark for delete. Once the search condition is acceptable, we add the delete clause at the end of the command to remove those events from Splunk. After deletion, not even a user with admin privilege is able to view this data in Splunk.

Removal of data is irreversible. If you still want the removed data back into Splunk then you should have the original source data copy with you which can be used to re-index the data in Splunk. It will be a process similar to creating a new index.

Assigning Delete Privilege

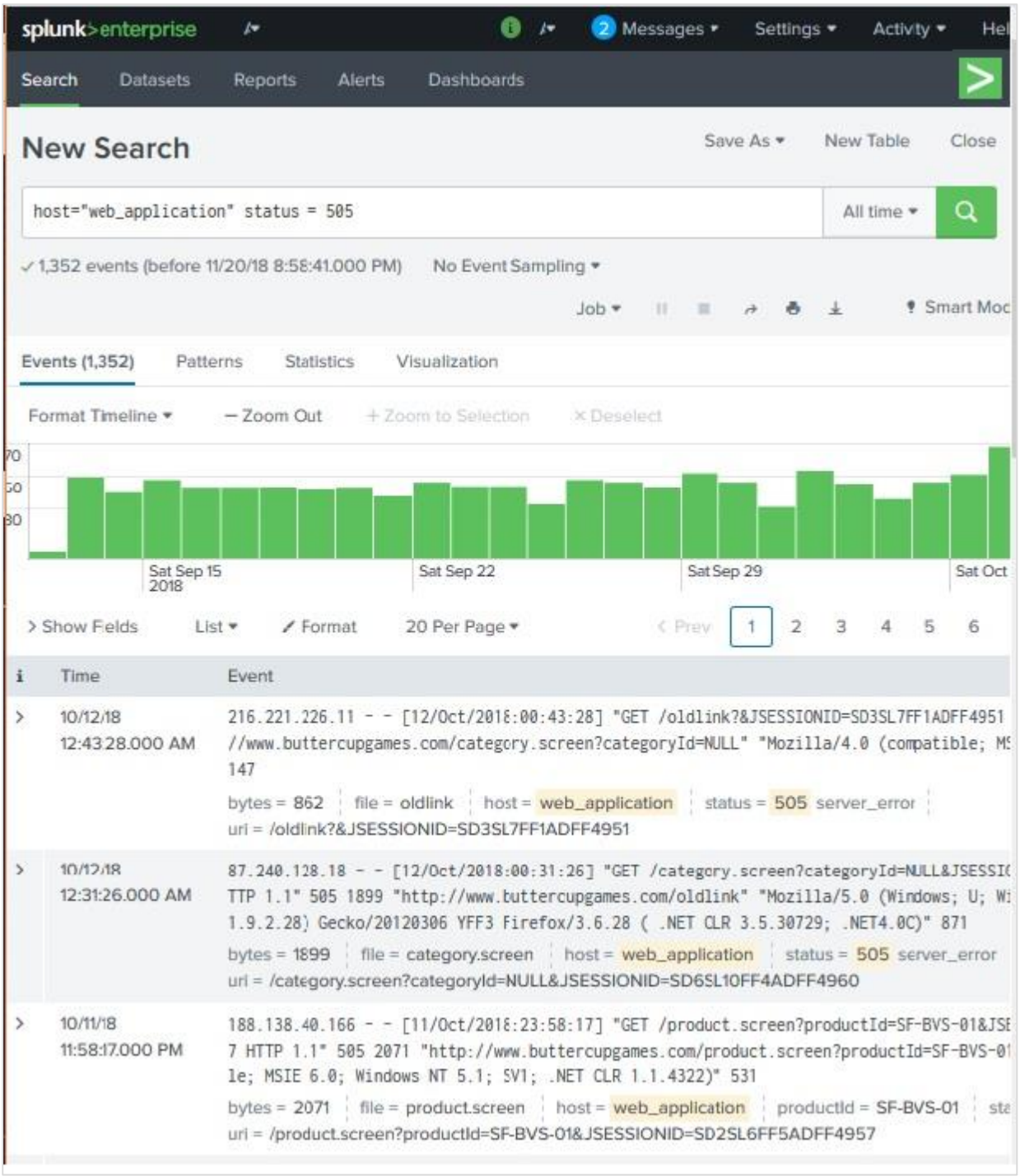
Any user including admin user does not have access to delete the data by default. By default, only the "**can_delete**" role has the ability to delete events. So, we create a new user, assign this role and then login with the credentials of this new user to perform the delete operation. The below image shows how we create a new user with "can_delete" role. We arrive at this screen by following the path **Settings -> Access Controls -> Users -> New User**.



We then log out of Splunk interface and login back with this newly created user.

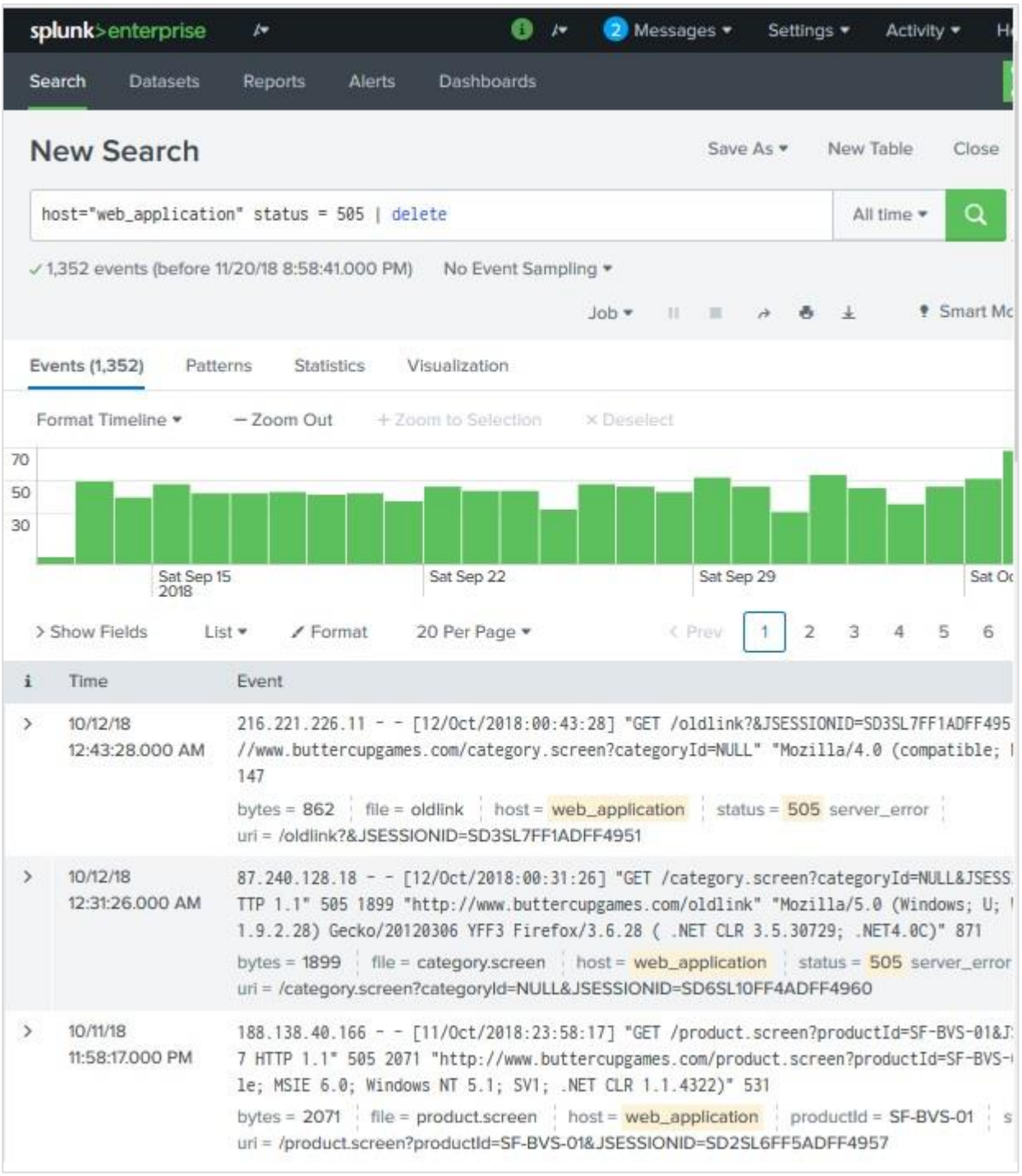
Identifying the data to be removed

First, we need to identify the list of events we want to remove. It is done using a normal search query specifying the filter condition. In the below example, we choose to look for the events from the host web_application which has the field http status value as **505**. Our goal is to delete only the set of data containing these values to be removed from the search result. The below image shows this set of data selected.

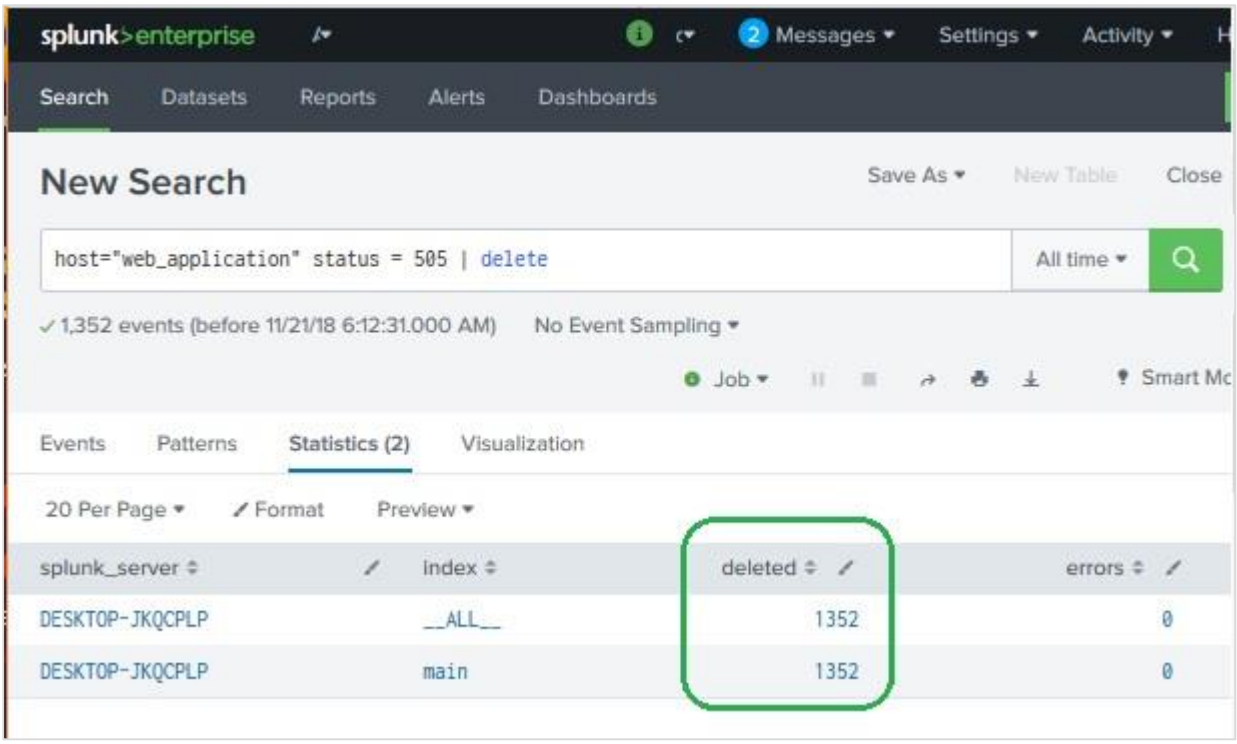


Deleting the Selected Data

Next, we use the delete command to remove the above selected data from the result set. It involves just adding the word delete after '|' at the end of the search query as shown below:



After running the search query above, we can see the next screen where those events have got deleted.

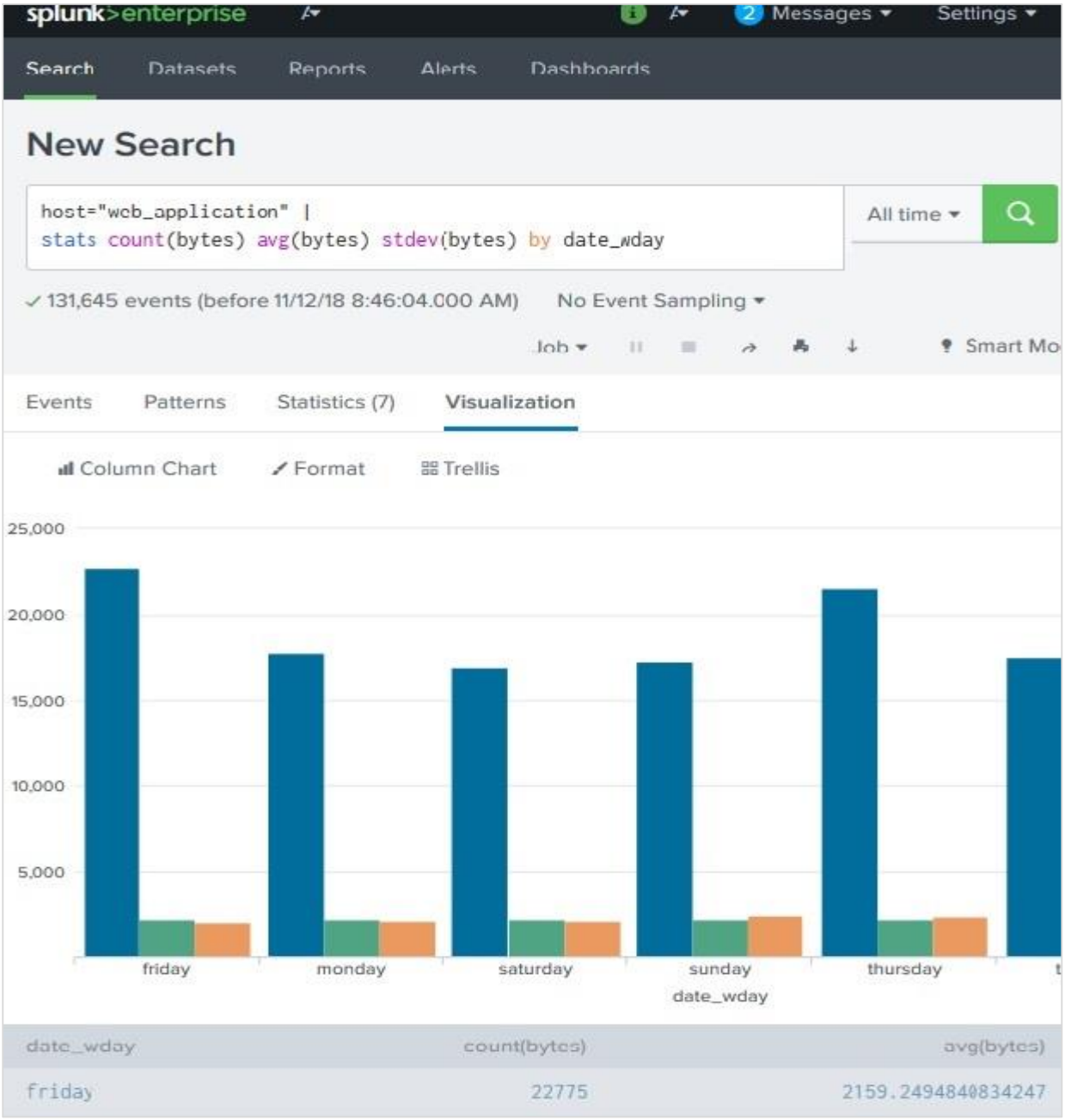


You can also further run the search query to verify that these events are not returned in the result set.

30. Splunk – Custom Chart

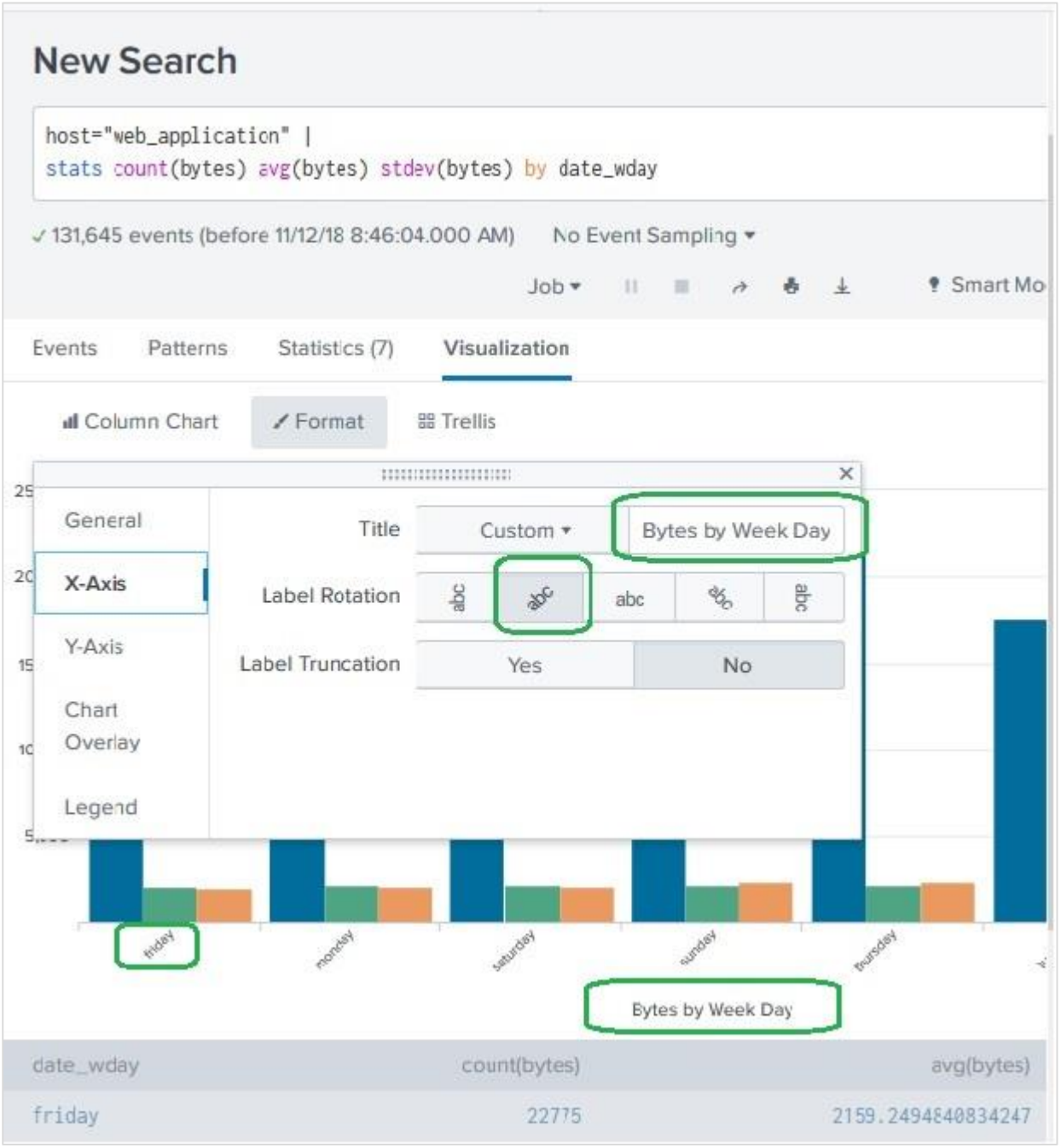
The charts created in Splunk has many features to customize them as per the user need. These customizations help in displaying the data completely or changing the interval for which the data is calculated. After initially creating the chart, we dive into the customization features.

Let us consider the below search query for getting the statistics of various measurements of byte size of the files by week day. We choose a column chart to display the graph and see the default values in the X-axis and Y-Axis values.



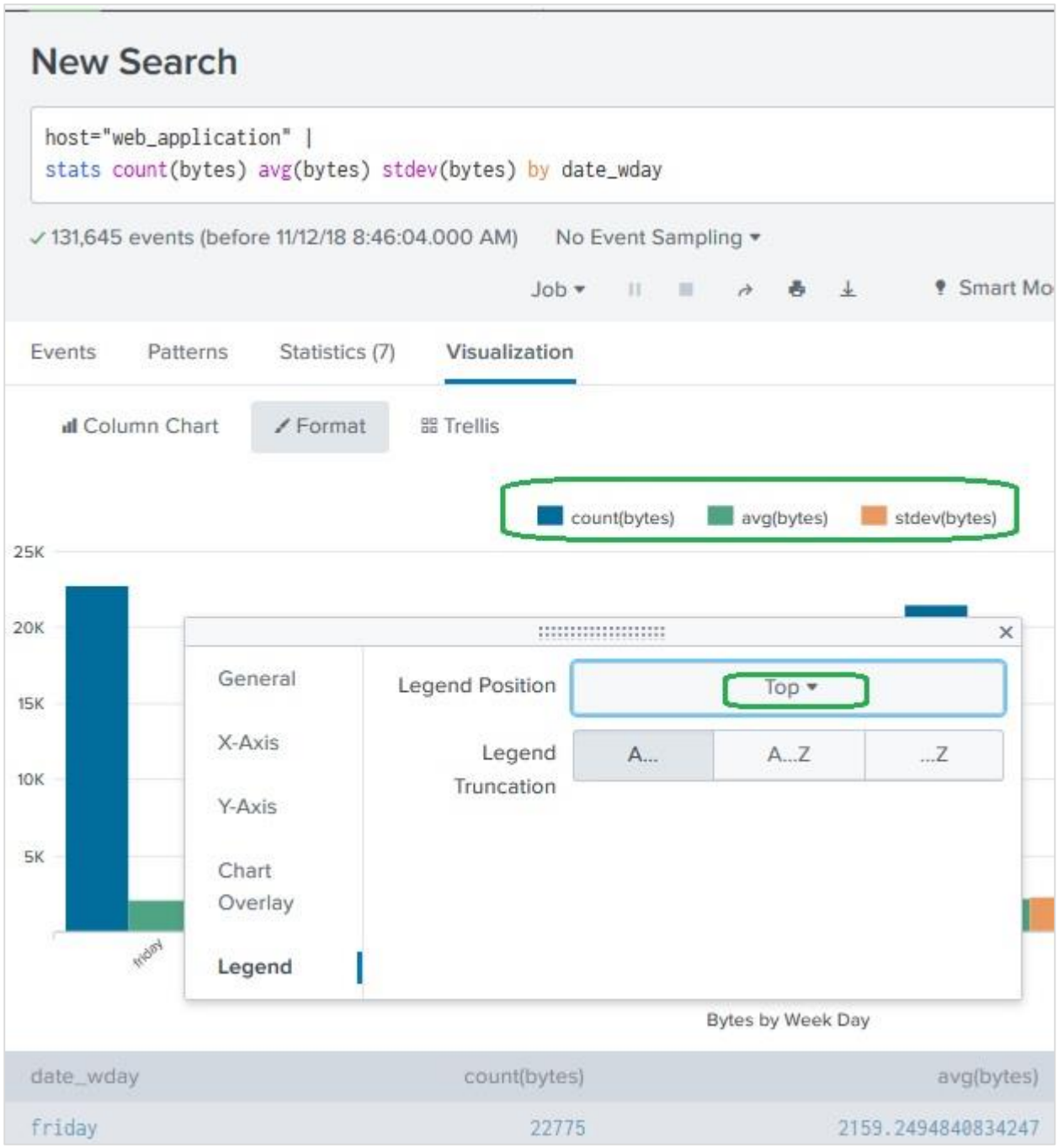
Axis Customization

We can customize the axes displayed in the chart by choosing the **Format -> X-axis** button. Here, we edit the Title of the chart. We also edit the Label Rotation option to choose an inclined label to fit better into the chart. After editing these, results can be seen in the chart as highlighted using the green boxes below.



Legend Customization

The legends of the chart can also be customized by using the option **Format -> Legend**. We edit the option Legend Position to mark it at Top. We also edit the Legend Truncation option to Truncate the End of the legend if required. The below cart shows the legends displayed at the top with colors and values.



31. Splunk – Monitor Files

Splunk Enterprise monitors and indexes the file or directory as new data appears. You can also specify a mounted or shared directory, including network file systems, as long as Splunk Enterprise can read from the directory. If the specified directory contains subdirectories, the monitor process recursively examines them for new files, as long as the directories can be read.

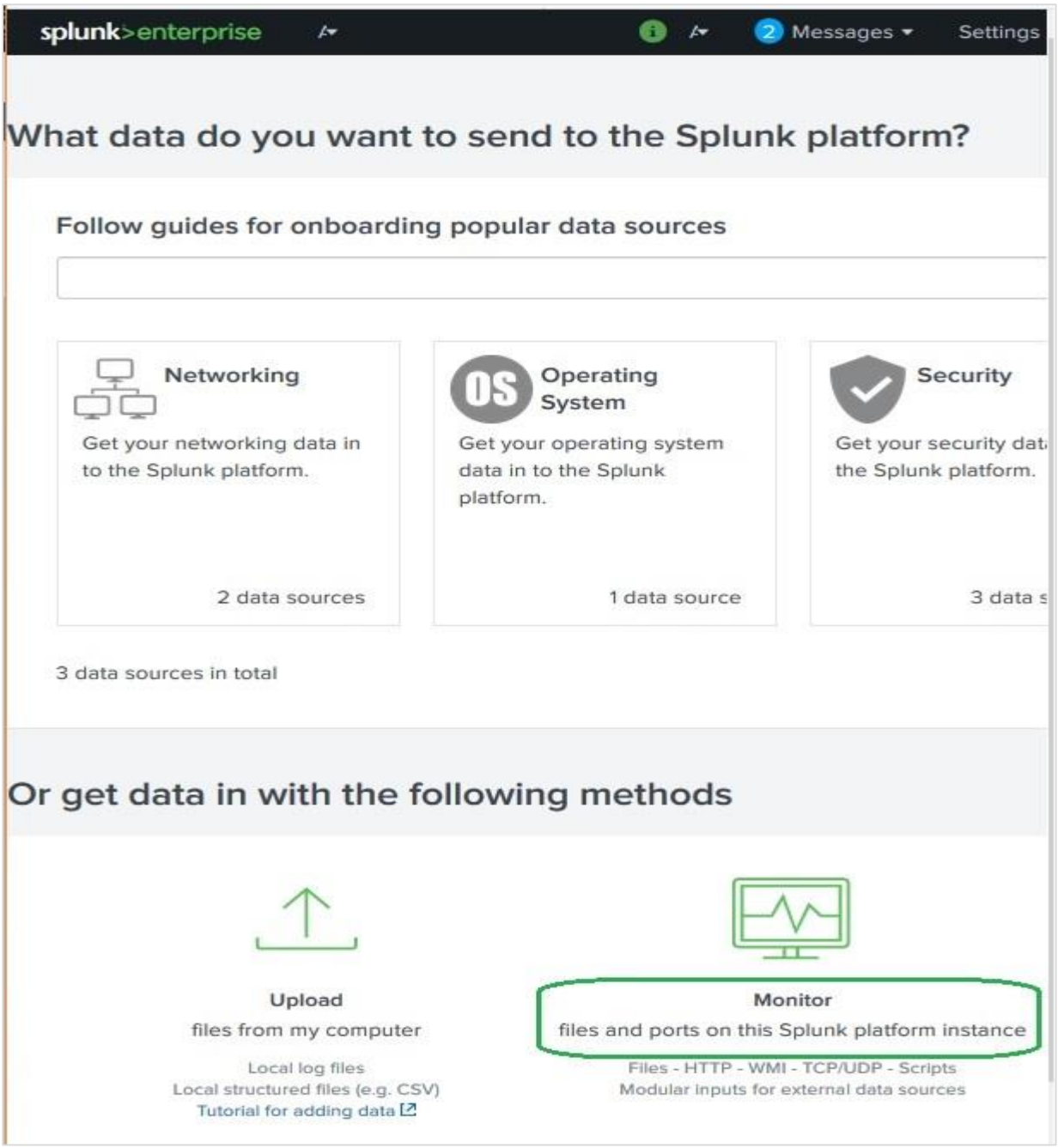
You can include or exclude files or directories from being read by using whitelists and blacklists.

If you disable or delete a monitor input, Splunk Enterprise does not stop indexing the files: input references. It only stops checking those files again.

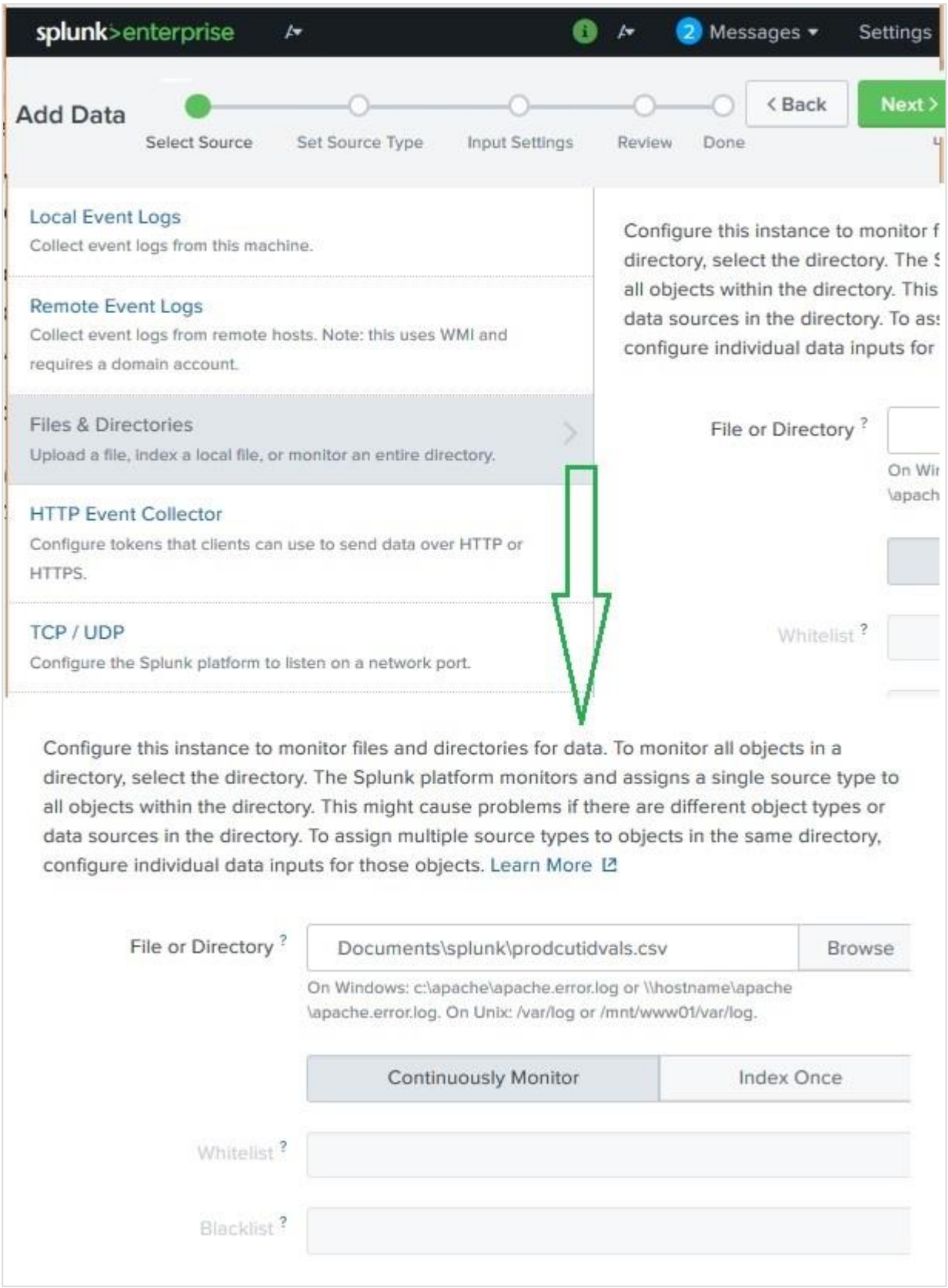
You specify the path to a file or directory and the monitor processor consumes any new data written to that file or directory. This is how you can monitor live application logs such as those coming from Web access logs, Java 2 Platform or .NET applications, and so on.

Add files to Monitor

Using Splunk web interface, we can add files or directories to be monitored. We go to **Splunk Home -> Add Data -> Monitor** as shown in the below image:

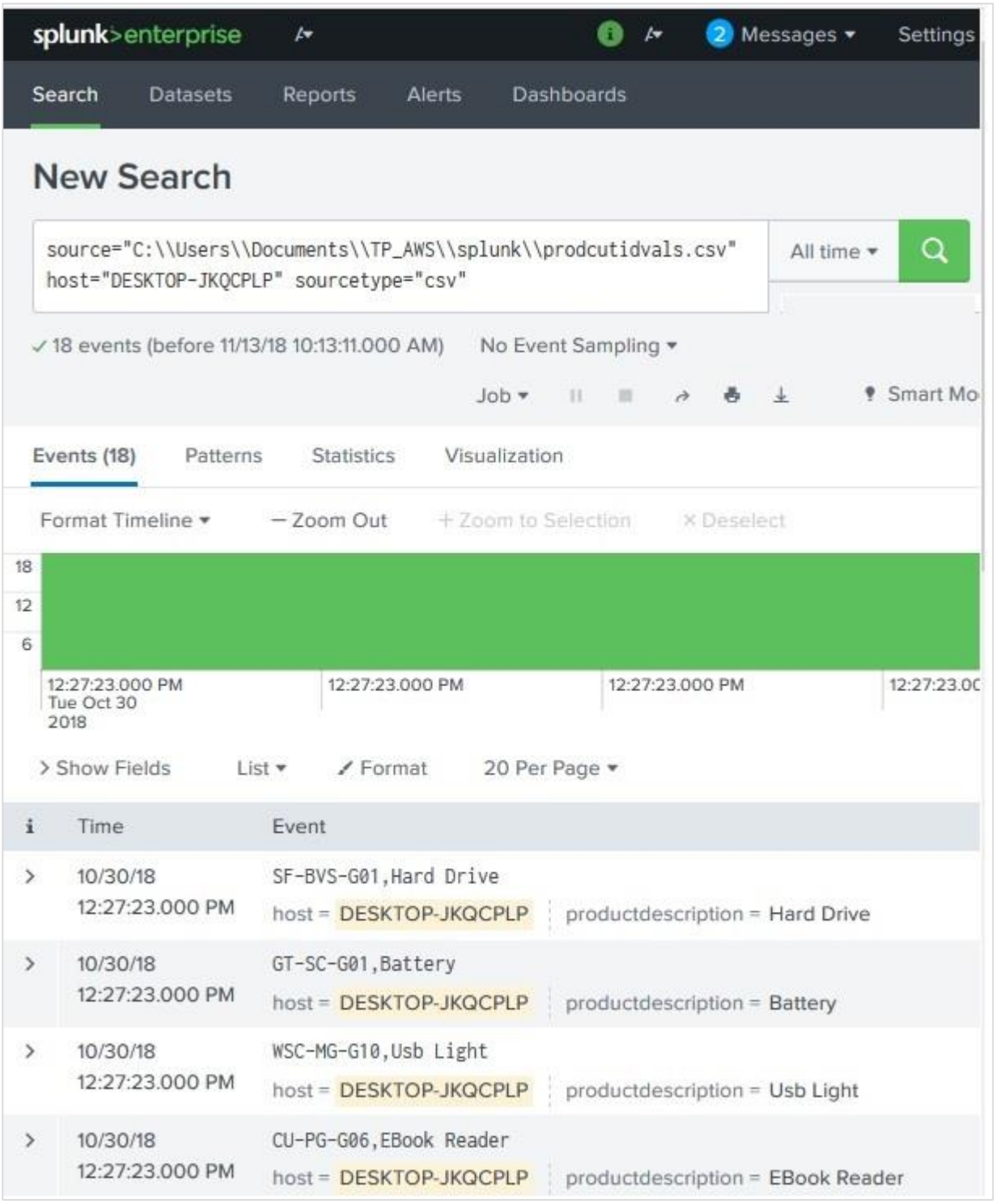


On clicking Monitor, it brings up the list of types of files and directory you can use to monitor the files. Next, we choose the file we want to monitor.



Next, we choose the default values as Splunk is able to parse the file and configure the options for monitoring automatically.

After the final step, we see the below result which captures the events from the file to be monitored.



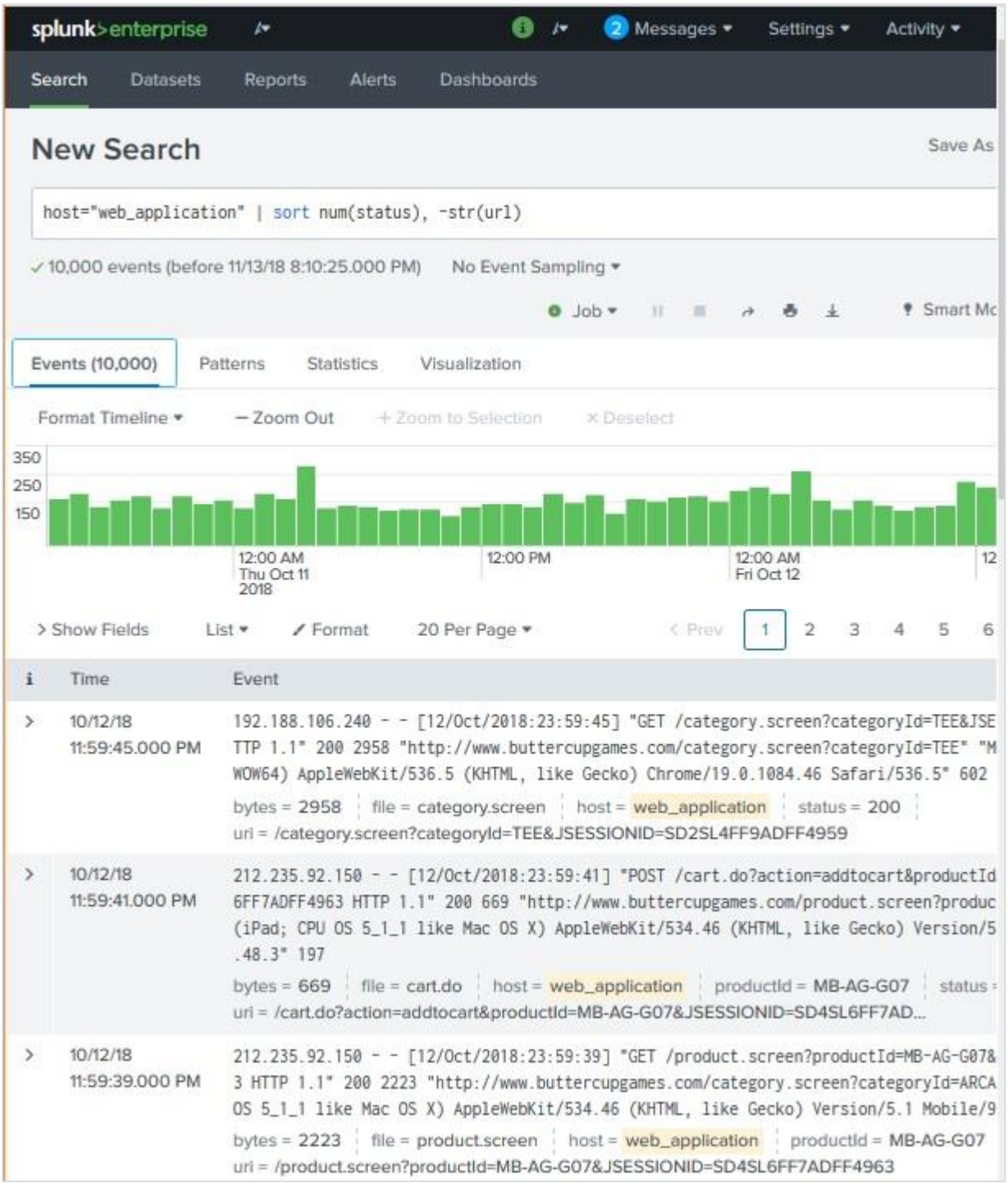
If any of the value in the event changes, then the above result gets updated to show the latest result.

32. Splunk – Sort Command

The **sort** command sorts all the results by specified fields. The missing fields are treated as having the smallest or largest possible value of that field if the order is descending or ascending, respectively. If the first argument to the sort command is a number, then at most that many results are returned, in order. If no number is specified, the default limit of 10000 is used. If the number 0 is specified, all of the results are returned.

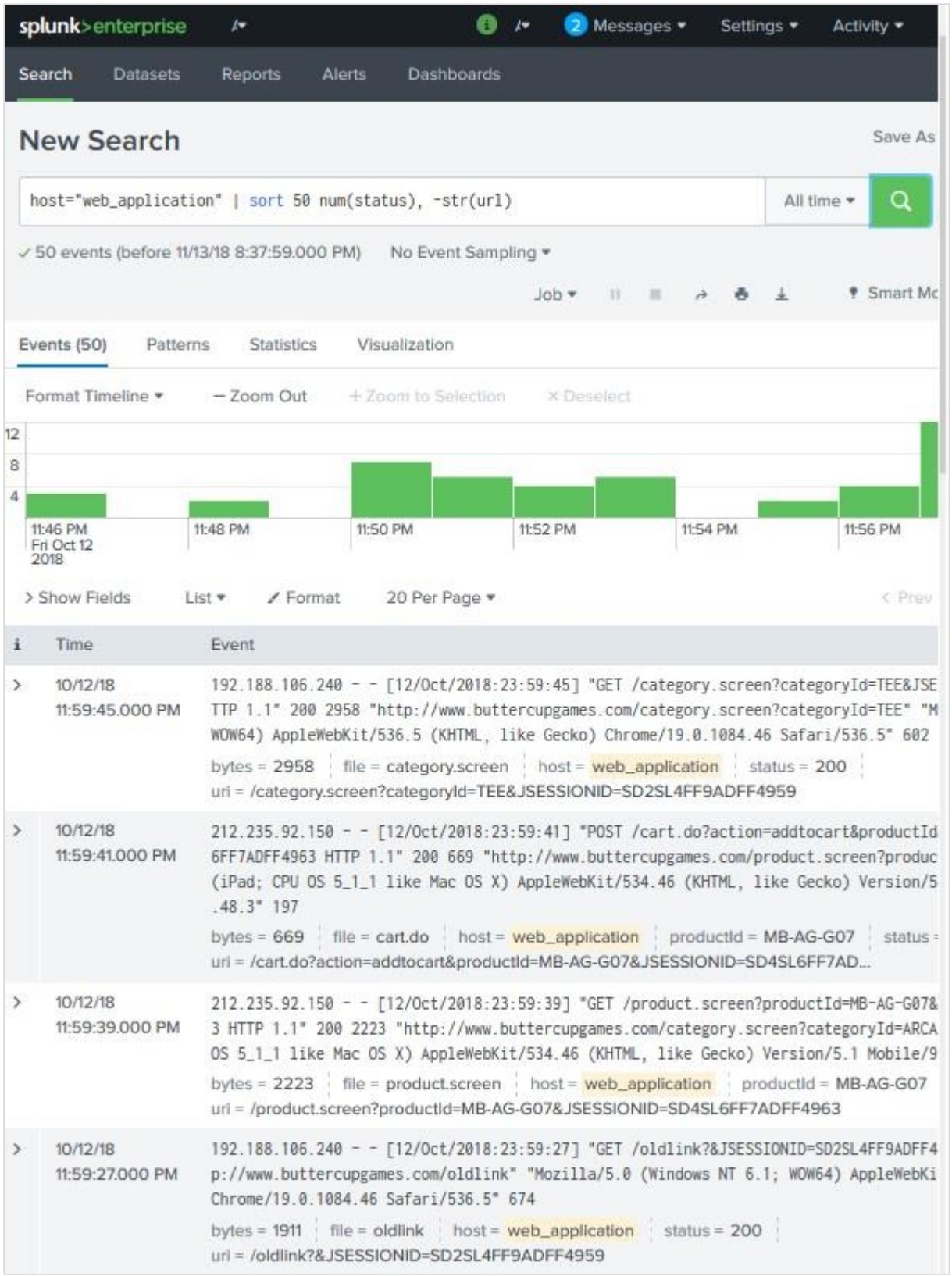
Sorting by Field Types

We can assign specific data type for the fields being searched. The existing data type in the Splunk dataset may be different than the data type we enforce in the search query. In the below example, we sort the status field as numeric in ascending order. Also, the field named url is searched as a string and the negative sign indicates descending order of sorting.



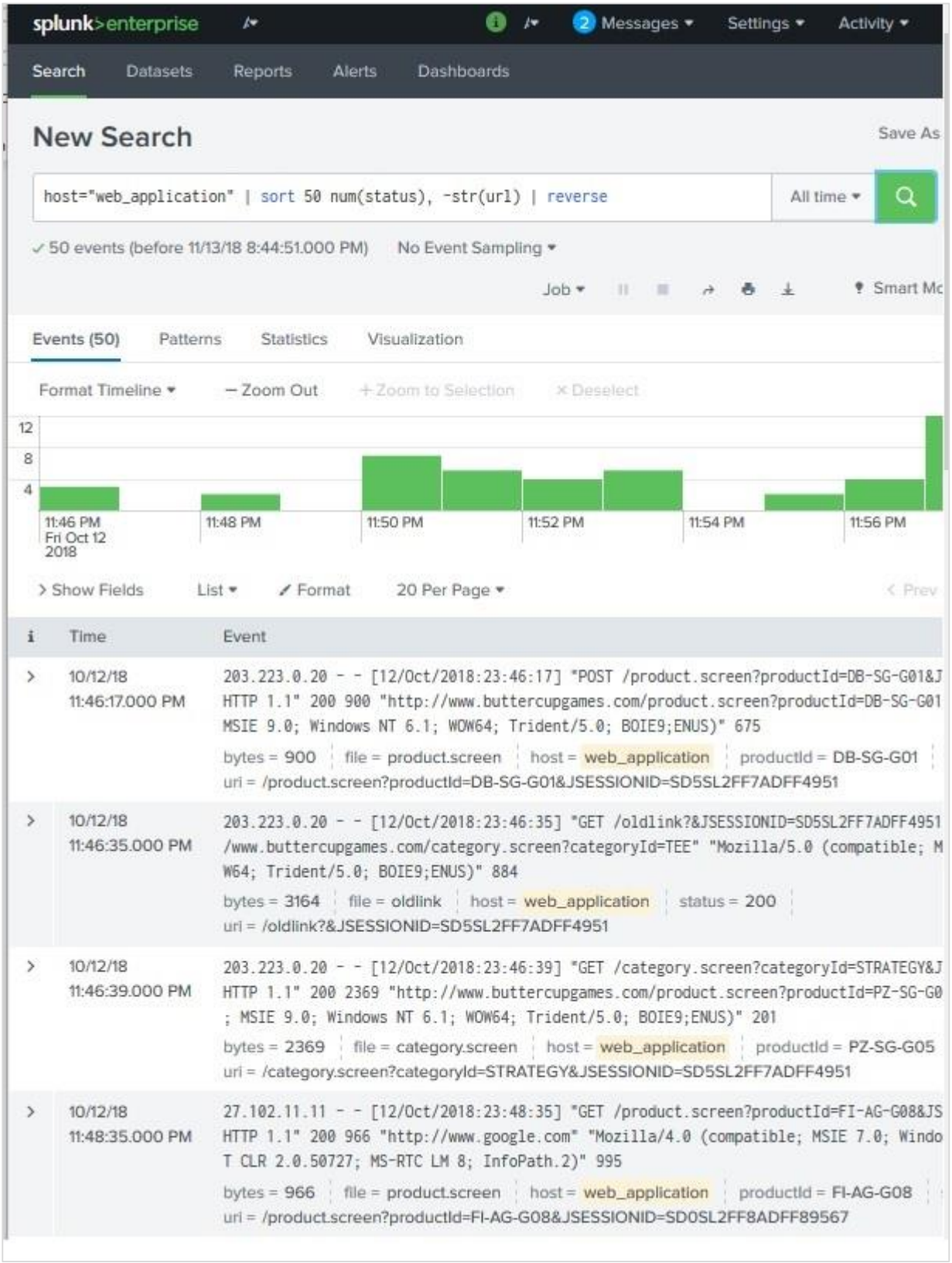
Sorting up to a Limit

We can also specify the number of results that will be sorted instead of the entire search result. The below search result shows the sorting of only 50 events with **status** as ascending and **url** as descending.



Using Reverse

We can toggle the result of an entire search query by using the reverse clause. It is useful to use the existing query without altering and reversing the sort result as and when needed.

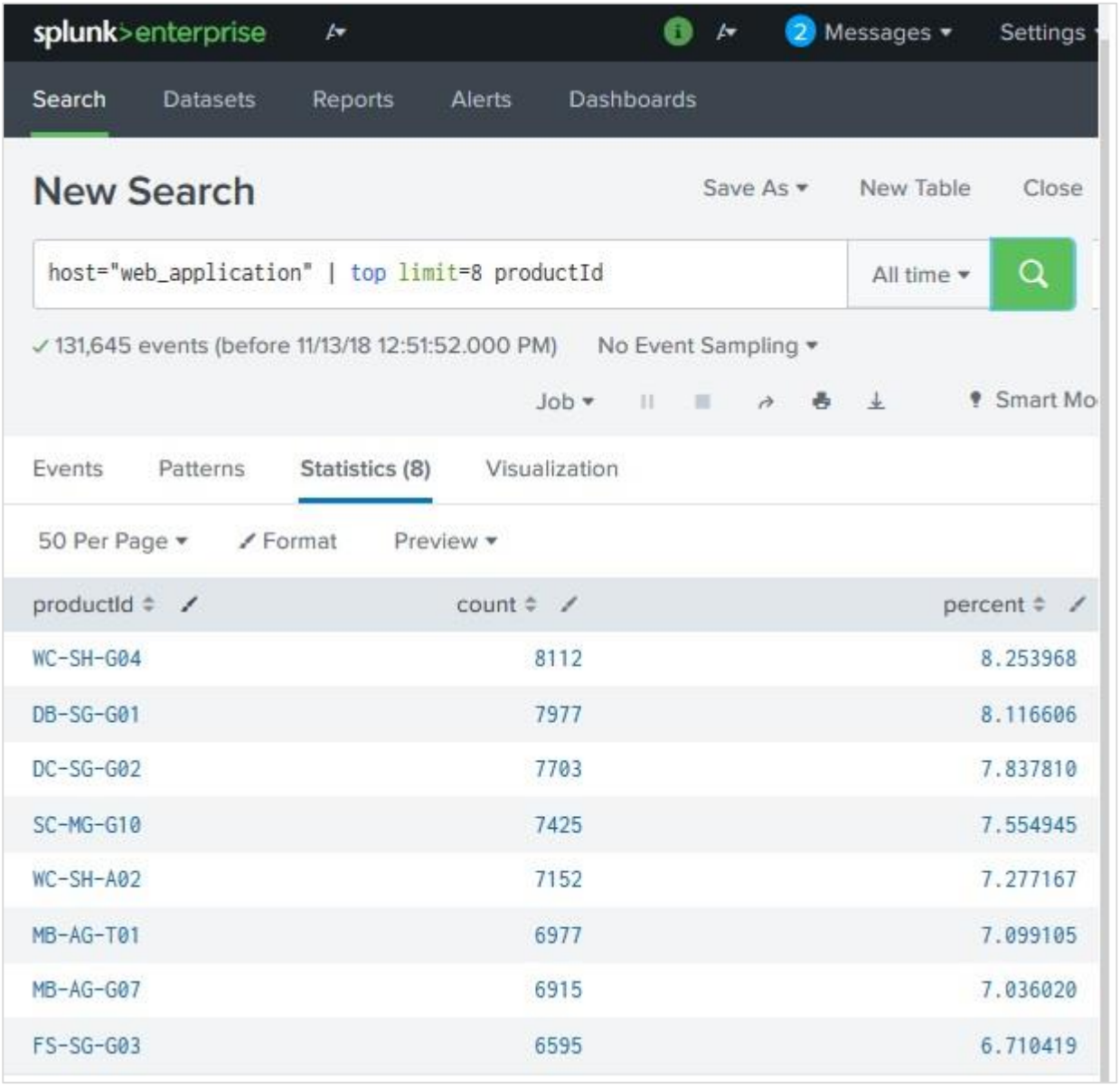


33. Splunk – Top Command

Many times, we are interested in finding the most common values available in a field. The **top** command in Splunk helps us achieve this. It further helps in finding the count and percentage of the frequency the values occur in the events.

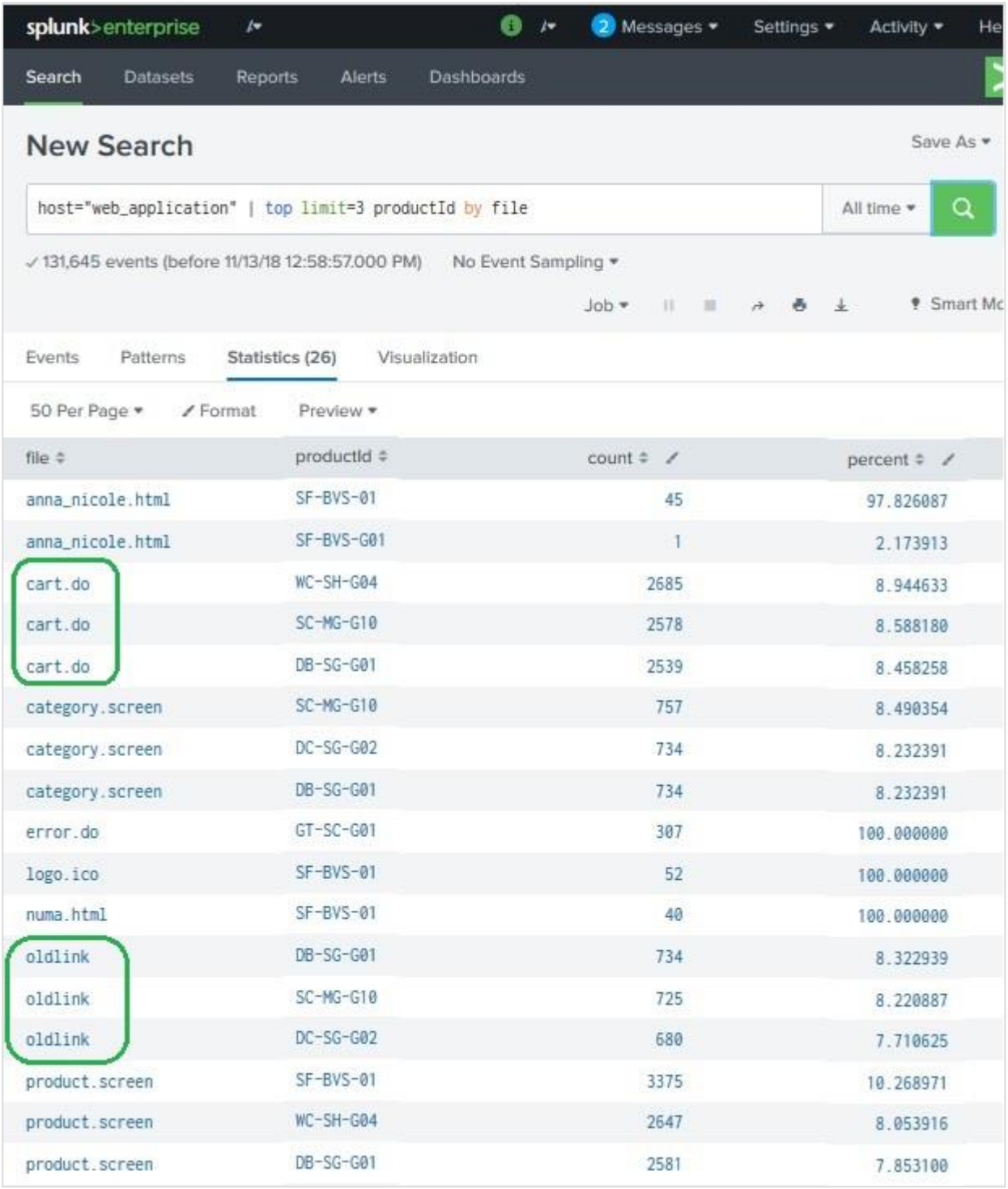
Top Values for a Field

In its simplest form, we just get the count and the percentage of such count as compared to the total number of events. In the below example, we find 8 top most productid values.



Top Values for a Field by a Field

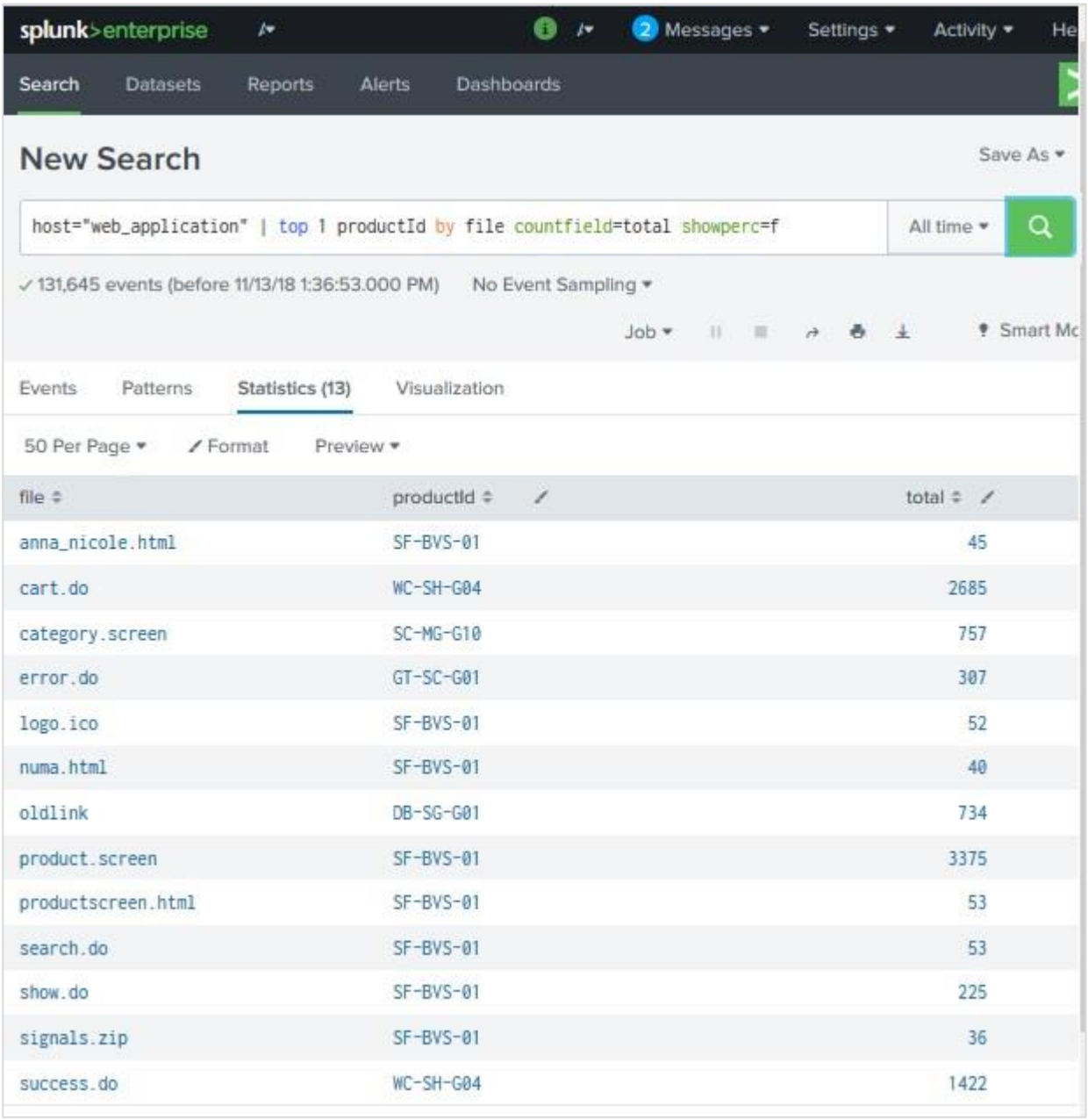
Next, we can also include another field as part of this top command's by clause to display the result of field1 for each set of field2. In the below search, we find top 3 productids for each file name. Note how the file names are repeated 3 times showing different productid for that file.



file	productId	count	percent
anna_nicole.html	SF-BVS-01	45	97.826087
anna_nicole.html	SF-BVS-G01	1	2.173913
cart.do	WC-SH-G04	2685	8.944633
cart.do	SC-MG-G10	2578	8.588180
cart.do	DB-SG-G01	2539	8.458258
category.screen	SC-MG-G10	757	8.490354
category.screen	DC-SG-G02	734	8.232391
category.screen	DB-SG-G01	734	8.232391
error.do	GT-SC-G01	307	100.000000
logo.ico	SF-BVS-01	52	100.000000
numa.html	SF-BVS-01	40	100.000000
oldlink	DB-SG-G01	734	8.322939
oldlink	SC-MG-G10	725	8.220887
oldlink	DC-SG-G02	680	7.710625
product.screen	SF-BVS-01	3375	10.268971
product.screen	WC-SH-G04	2647	8.053916
product.screen	DB-SG-G01	2581	7.853100

Show Options

We can also decide to show specific columns by using additional options available in Splunk with the Top Command. In the below command, we disable to show the percentage option and display only the top product ID by File name.



file	productId	total
anna_nicole.html	SF-BVS-01	45
cart.do	WC-SH-G04	2685
category.screen	SC-MG-G10	757
error.do	GT-SC-G01	307
logo.ico	SF-BVS-01	52
numa.html	SF-BVS-01	40
oldlink	DB-SG-G01	734
product.screen	SF-BVS-01	3375
productscreen.html	SF-BVS-01	53
search.do	SF-BVS-01	53
show.do	SF-BVS-01	225
signals.zip	SF-BVS-01	36
success.do	WC-SH-G04	1422

34. Splunk – Stats Command

The stats command is used to calculate summary statistics on the results of a search or the events retrieved from an index. The stats command works on the search results as a whole and returns only the fields that you specify.

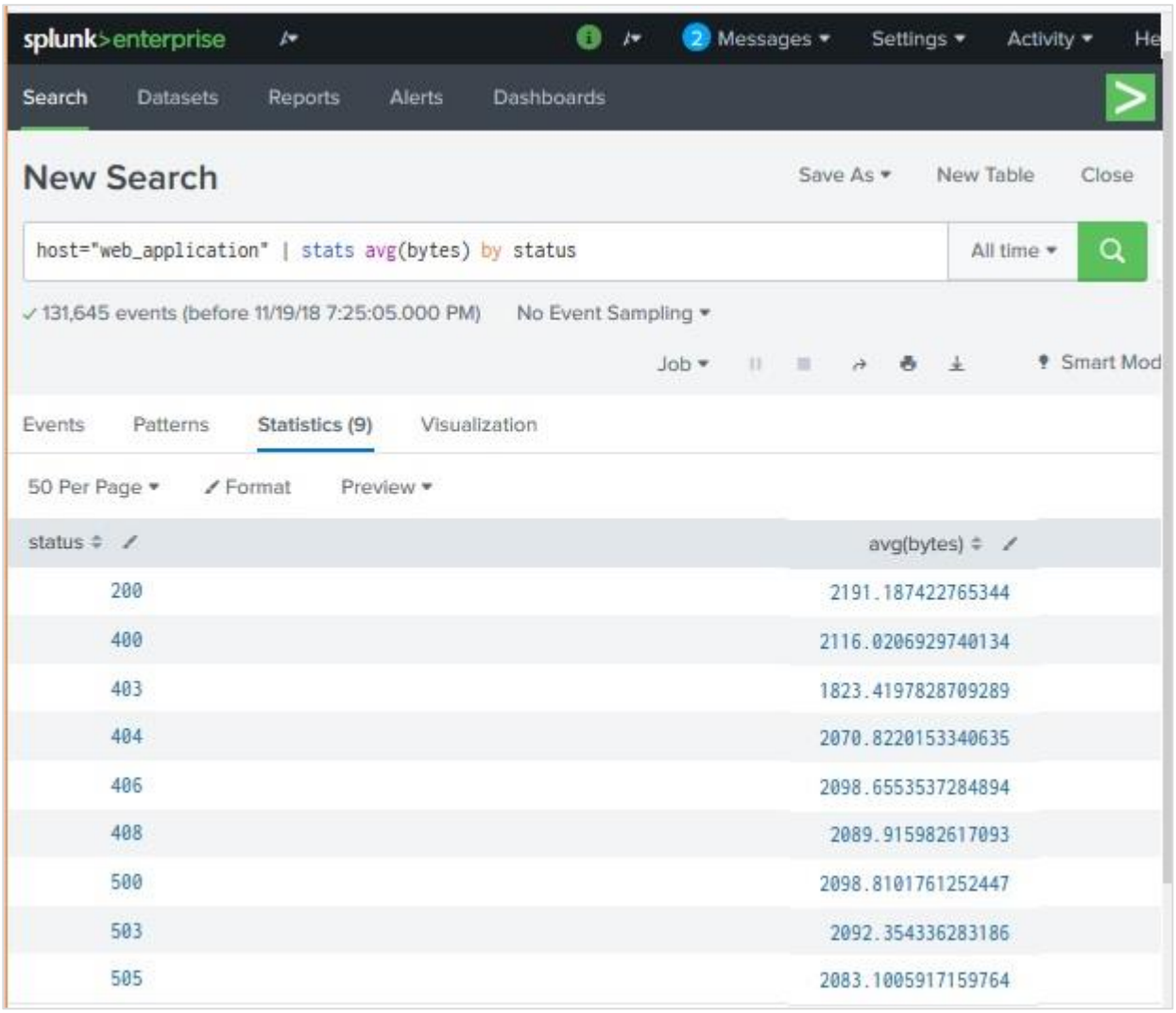
Each time you invoke the stats command, you can use one or more functions. However, you can only use one BY clause. If the stats command is used without a BY clause, only one row is returned, which is the aggregation over the entire incoming result set. If a BY clause is used, one row is returned for each distinct value specified in the BY clause.

Below we see the examples on some frequently used stats command.

Finding Average

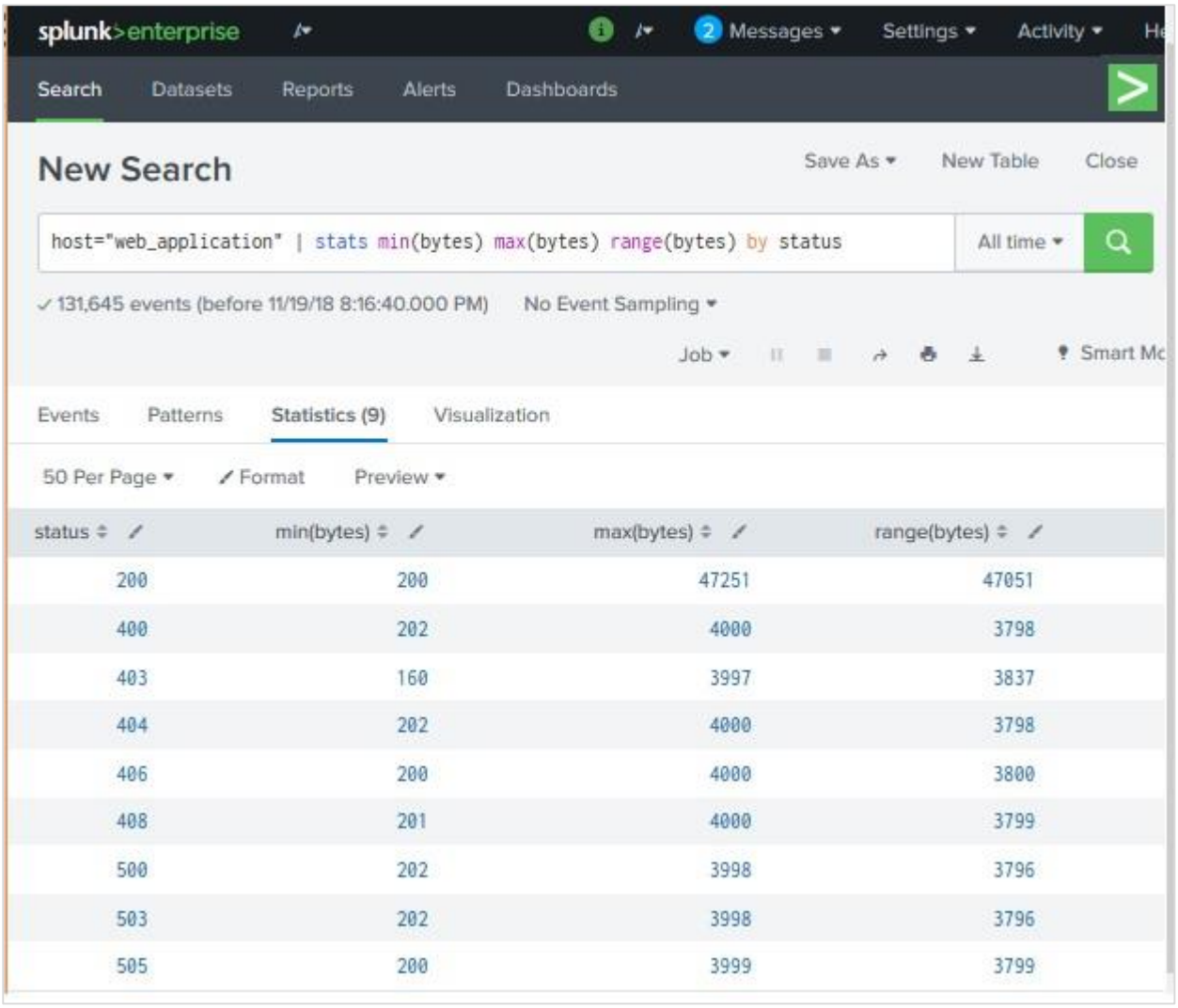
We can find the average value of a numeric field by using the **avg()** function. This function takes the field name as input. Without a BY clause, it will give a single record which shows the average value of the field for all the events. But with a by clause, it will give multiple rows depending on how the field is grouped by the additional new field.

In the below example, we find the average byte size of the files grouped by the various http status code linked to the events associated with those files.



Finding Range

The stats command can be used to display the range of the values of a numeric field by using the **range** function. We continue the previous example but instead of average, we now use the **max(), min()** and **range** function together in the stats command so that we can see how the range has been calculated by taking the difference between the values of max and min columns.



status	min(bytes)	max(bytes)	range(bytes)
200	200	47251	47051
400	202	4000	3798
403	160	3997	3837
404	202	4000	3798
406	200	4000	3800
408	201	4000	3799
500	202	3998	3796
503	202	3998	3796
505	200	3999	3799

Finding Mean and Variance

Statistically focused values like the mean and variance of fields is also calculated in a similar manner as given above by using appropriate functions with the stats command. In the below example, we use the functions **mean()** and **var()** to achieve this. We continue using the same fields as shown in the previous examples. The result shows the mean and variance of the values of the field named bytes in rows organized by the http status values of the events.