Contents

[Splunk Tutorial 2](#_Toc353959059)

[The Search App 2](#_Toc353959060)

[What is it? 2](#_Toc353959061)

[Terminology 2](#_Toc353959062)

[Events 2](#_Toc353959063)

[Fields 2](#_Toc353959064)

[The Search Language 2](#_Toc353959065)

[Basic Search Syntax 3](#_Toc353959066)

[Refining the Results: Commands 4](#_Toc353959067)

[Refining the Results: Functions 8](#_Toc353959068)

[Searches 10](#_Toc353959069)

[Saved Searches 10](#_Toc353959070)

[Macros 11](#_Toc353959071)

[Dashboards 12](#_Toc353959072)

[Creating a dashboard from a search 12](#_Toc353959073)

# Splunk Tutorial

This tutorial is meant to teach a user the basics of using Splunk to collect useful information for a particular application, and use Splunk overall more effectively and efficiently. It is mainly aimed at people who have previously used Splunk in at least some capacity, and are familiar with getting around in Splunk. For those who need to get access to Splunk, instructions can be found on the [wiki page](https://sfwikis.opr.statefarm.org/sites/systems/Wiki%20Pages/SplunkUserGuide.aspx).

**Splunk by environment:**

* [Infrastructure](http://sfsplunkinf.opr.test.statefarm.org/)
* [Development](http://sfsplunkdev.opr.test.statefarm.org/)
* [Implementation & Performance](http://sfsplunkperf.opr.test.statefarm.org/)
* [Production](http://sfsplunk.opr.statefarm.org/)

## The Search App

### What is it?

The Search App is the main view, and most likely where you will be spending most of your time. From this view you can run searches, build reports, create dashboards, and pretty much everything else you can do, starts here. If you are unfamiliar with it, take a moment to read through the [Search App overview](http://docs.splunk.com/Documentation/Splunk/4.3.3/User/Searchapptutorial).

## Terminology

### Events

Events are what a search returns. If you consider a log for an application, each entry in the log would be considered an event. When you construct a search, the events that are returned to you, are the events that matched the criteria that you provided.

### Fields

Fields refer to pieces of information that Splunk knows how to pick out of an event. Consider the following snippet from an event:

<*context* *ear=*"*PUXRQA06\_mhk32-p73.ear*" *displayName=*"*PUXRQA06*" *threadId=*"*WebContainer* *:* *3*" *user=*"*UJC9@AGCY.STATEFARM.ORG*" *executionEnvironment=*"*cea57p73/nea571-ws73pezl/mhk32c1-ws73pezl*" *location=*"*ws73pezl*" */*>

This snippet contains the fields: ear, displayName, threadId, user, executionEnvironment, and location.

## The Search Language

Without even knowing anything about Splunk you can probably type in some keywords related to what you are trying to find, and you will get some results that are relevant. This would kind of be a brute force approach to using Splunk. Splunk would basically take whatever you put into the search bar and look for any events that contained the words or values you typed in. In all likelihood a lot of bloat is going to be returned, that isn’t relevant to what you intended to search for. To make sure Splunk returns only events that you are interested, you need to know how to tell Splunk what to look for and where.

Splunk has what it calls a “Search Language,” which is used to construct a query that will return some information. This language can be used to get something as simple as a list of events that match the criteria provided, or it can get a result set, perform statistical analysis on it, and then return those statistics in the form of a chart. Knowing how to use it will increase the quality of the results you get back from a search.

For more info: [Search Reference](http://docs.splunk.com/Documentation/Splunk/4.3.3/SearchReference/WhatsInThisManual)

### Basic Search Syntax

#### Fundamentals

* Splunk will attempt to match against each term in the search.
* Spaces between search terms are treated as an implied AND.
* If you are trying to search for a phrase use quotation marks, i.e. "login failure"
* Use the time picker to select a timeframe to search within.
* For more complicated searches use parenthesis.
* The pipe ("|") is used to chain together actions. This is similar to how the pipe is used in the Linux shell, where using pipes sends the output of the program on the left as the input to the program on the right.

#### Using Fields to Search

By including fields in a search, you can tell Splunk exactly where to look for a specific value. This is much the same as SQL, where you would construct a query using key/value pairs to retrieve information. Let’s say we want a search that only returns events with a displayName that has the value of PUXRQA06. To do this we would simply put the following in the search bar:

displayName="PUXRQA06"

You do not have to use the quotes, if you do not have a space in the value you are searching for. There are a couple of things to keep in mind when using a field in a search. First, the name of the field **is case-sensitive**. So, displayname and displayName will not be treated as being the same field by Splunk. However, the value to the right of the equal sign **is not case-sensitive**, so puxrqa06 and PUXRQA06 are considered being equal.

When you run a search, you will see along the left side of the screen the Fields bar. This shows the fields that Splunk is aware of. If you click on a field name, it will show you the top values that were found for that field, and how many of each instance occurred. Using the Fields bar can be useful for constructing a search. If you find a part of an event that you think should be a field, but is currently not recognized by Splunk, there is a way to add it, so that Splunk will recognize it.

#### Source-type

One of the first things you want to specify in your search is the source-type. Source-type refers to a specific data type or format. For example, if you wanted to look at the logs for one of our J2EE applications, you would want to look in the "eventing" source-type.

Example:

sourcetype="eventing"

Although not required, specifying the source-type immediately limits the scope of the search. By not specifying it, Splunk will be searching through all available source-types, even if they are not relevant to what you are trying to search for. You can find a list of all of the available source-types by going to the Summary view.

#### Arithmetic Operators

The standard arithmetic operators are available: +, -, \*, /, %

#### Boolean Operators

The basic Boolean operators are available, such as: AND, OR, XOR, LIKE, and NOT. The AND operator is implied between terms, and therefore does not need to be explicitly used.

#### Evaluation Operators

The standard evaluators are available: <, <=, >, >=, =, ==, and !=.

### Refining the Results: Commands

There are going to be times when the raw events returned from a search are not enough. You are going to want to be able to run some statistics for analysis, and perhaps you will want to look beyond the raw values and create charts to display the information. There are many commands that can be used to help shape this data, this will just touch on a few that you are more likely to use. You can find a full list of commands [here](http://docs.splunk.com/Documentation/Splunk/4.3.3/SearchReference/ListOfSearchCommands).

#### eval

The eval command can be used to evaluate an expression and store the result in a new field. [Available functions.](http://docs.splunk.com/Documentation/Splunk/4.3.3/SearchReference/CommonEvalFunctions)

**Example:**

(sourcetype="eventing" displayName="PUXRQA06" logLevel="error") OR

(sourcetype="ihs\_access" contextRoot="CreditCardApp\_jrf")

| stats count(logLevel) as errors, count(contextRoot) as total

| eval percentage = (1 - (errors / total)) \* 100

**Explanation:**

This is the search that is used to calculate the throughput for our applications on the app chooser dashboard. The percentage is calculated using the eval command. The search first gets all of the error events and requests sent to the context root in the IHS logs. Then it counts up the errors and the requests. Finally, eval is used to perform a calculation using the values obtained in the previous step, and then stores it in a new field called percentage.

#### stats

The stats command can be used in conjunction with functions to present results in the form of a table. [Available functions.](http://docs.splunk.com/Documentation/Splunk/4.3.3/SearchReference/CommonStatsFunctions)

**Example:**

sourcetype="eventing" displayName="PUXRQA06" logLevel="error" | stats count as errors

**Explanation:**

This is going to find all events for PUXRQA06 that are errors, count the number of events found (naming the resulting value errors), and finally displays the results in a table.

#### chart

The chart command can be used in conjunction with functions to present results in the form of a chart. The chart can be formatted after the results have been calculated or by using the report builder. [Available functions.](http://docs.splunk.com/Documentation/Splunk/4.3.3/SearchReference/CommonStatsFunctions)

**Example:**

sourcetype="eventing" displayName="PUXRQA06" logLevel="error" | chart count by text

**Explanation:**

This is going to find all events for PUXRQA06 that are errors, count the number of events found, and then group them by the text field. If you try a search like this in the regular search view, the results will be returned in a table. You can switch to the Results Chart view from the chart icon at the top of the results panel. When using the chart command, you will probably want to use the Advanced Charting View instead. You can find this view under either of the workgroup view menus, towards the bottom.

#### timechart

The timechart command can be used in conjunction with functions to show a result over a defined period of time. [Available functions.](http://docs.splunk.com/Documentation/Splunk/4.3.3/SearchReference/CommonStatsFunctions)

**Example:**

sourcetype="eventing" displayName="PUXRQA06" logLevel="error" | timechart count as Errors

**Explanation:**

This is the same search we used for the stats command, only now using the timechart command. What this is going to do is include the \_time field for each of the events. So if you go into the Results Chart or the Advanced Charting view, it will now show the results over time.

#### sort

As the name suggests, this command is used to sort results in ascending/descending order by some field(s). Default is ascending order, you can precede the field with the minus sign to reverse the order.

**Example:**

sourcetype="eventing" displayName="PUXRQA06" logLevel="error" | stats count(text) as errors by text | sort –errors

**Explanation:**

This will get all of the errors for PUXRQA06, count them grouped by the field text, and then it will sort them in descending order according to the value of errors.

#### top

The top command returns the top values of the field that you specify.

**Example:**

sourcetype="eventing" displayName="PUXRQA06" logLevel="error" | top text

**Explanation:**

This will get all of the errors for PUXRQA06, and return a table with the top occurring values for text, the number of occurrences for that value, and what percentage that is of all values.

#### rex

The rex command allows you to use a regular expression to extract a field during a search. You would use this if there was a field you wanted to extract from the event that Splunk itself does not recognize in the Fields bar. For more information on the rex command, go [here](http://docs.splunk.com/Documentation/Splunk/4.3.3/SearchReference/Rex).

**Example:**

sourcetype="eventing" displayName="PUVVNA00" logLevel="warn"

| rex field=\_raw "B2C ID - (?<b2cid>\d+)"

**Explanation:**

In the logs for OAC, there are warning messages that have the B2C ID somewhere in the event as seen below.

<message text="Single Signon to CeB failed for B2C ID - 6344651645167600" />

The search is going to find all warnings for PUVVNA00, and then attempt to extract a new field called b2cid anywhere that it finds the characters "B2C ID - " followed by one or more digits.

### Refining the Results: Functions

There are a lot of different functions that can be used with the various commands. Most operate across commands, and are not tied to a specific one. This is going to cover some of the functions that you are most likely to need. For a full list, go [here](http://docs.splunk.com/Documentation/Splunk/4.3.3/SearchReference/CommonStatsFunctions).

#### avg

The average function returns the average value for a specified field.

#### count

The count function counts the number of events that have the specified field.

**Example:**

(sourcetype="eventing" displayName="PUXRQA06" logLevel="error")

| stats count(text) as errors by text

**Explanation:**

First it finds all of the errors, and then it is going to count them, and group them by the value of text.

#### dc or distinct\_count

The count function counts the number of events that have the specified field.

**Example:**

(sourcetype="eventing" displayName="PUXRQA06" logLevel="error")

| stats count(text) as errors by text

**Explanation:**

First it finds all of the errors, and then it is going to count them, and group them by the value of text found in no particular order.

#### min & max

The min and max functions return the lowest and highest value, respectively, for a given field.

#### sum

The sum function will sum all of the values for a specified field.

#### values

The values function will return a list of all of the values for a specified field.

**Example:**

sourcetype="eventing" displayName="PUXRQA06" | stats values(text) as "error message" by logLevel

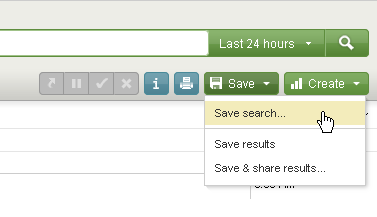
**Explanation:**

This will find all of the events for PUXRQA06, and return a table with all of the values for the text field grouped according to the logLevel (info, warn, and error).

## Searches

### Saved Searches

If you have a search that you find yourself running regularly, it would be a good candidate for a saved search. From the Search app, you can save a search that you have run by clicking on the Save button and selecting "Save search…" A dialog box will then pop up that will allow you to give the search a name.



Once a search has been saved, it can either be run from the Splunk Manager under "Searches and Reports," or you can run it directly from the search bar using the following syntax:

savedsearch=<name\_of\_search>

### Alerts

Splunk has the ability to monitor the logs and send a notification based on the results of a search through alerts. An alert is merely a saved search that has been scheduled and set to send a notification based on some criteria. Currently it appears that alerts are restricted to being created by power users only, so you may have to fill out a form if you need to create one (found in the wiki link at the beginning of this tutorial).

You can create an alert either when creating a saved search or by editing one. Either way there will be a section called "Schedule and alert" with a checkbox for scheduling the search. When clicked, it will expand with further options for scheduling and setting the notifications. You can either use a cron expression or use the basic type and select how frequently the search will run.

Next you will need to set the conditions that will trigger the alert. Typically for this you will select "if number of events" and set the threshold for the number of events required to trigger the alert. There is also an option to not trigger the alert again for a specified period of time if it has just been triggered, kind of like a sleep function.

Finally, for the alert actions you can select "send email" and enter a list of recipients to receive the alert. By selecting "include results in email" it will include a list of the events that were returned by the search.

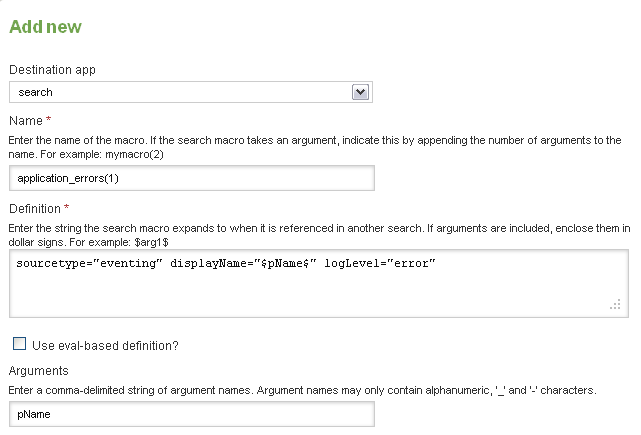
If you have already created an alert you can toggle it on/off from the manager by clicking enable/disable in the "Status" column of "Searches and reports" view.

### Macros

If you have a search that you are running for multiple applications, then it would be a good candidate for a macro. Macros are similar to saved searches, except that they can accept parameters as input to the search.

Creating a macro:

1. When defining the name, you need to follow it with parenthesis and the number of arguments it accepts. If it will not take any arguments, then you can just leave the parenthesis empty.
2. In the definition you will put your search. You will put placeholders in the search with the name for the variable wrapped in $.
3. For the arguments you will put a comma delimited list of the variables that you have in the search.



You can then execute the macro from the search bar by using the grave accent (`).

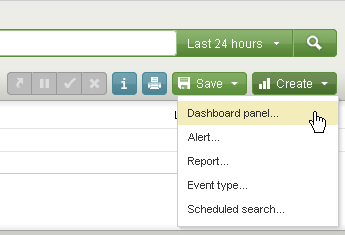
`application\_errors(pName="PUXRQA06")`

## Dashboards

A dashboard is simply a page that displays some sort of information. Most of the screens you will interact with in Splunk are actually dashboards, even the Search screen. Typically you are going to create a dashboard when you want to display data from different searches or present the data in different ways all on one page. There are different ways to create dashboards.

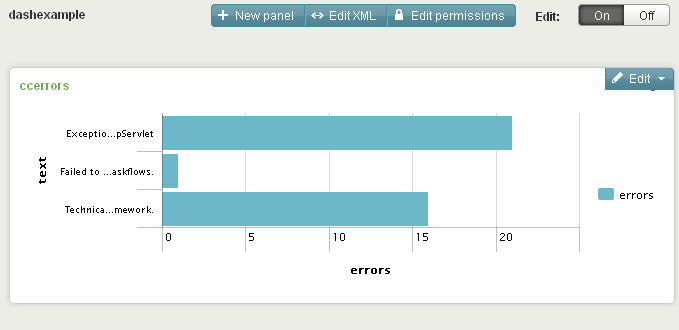
### Creating a dashboard from a search

Let's say that you have just run a search, and you would like to create a dashboard that would then utilize this search, and perhaps render the results in a chart of some kind. We can do this from the Search App.



This will take you through a wizard type of interface that will let you save the current search as a Saved Search, create a dashboard, create a panel for that search to run in, and then lets you choose the visualization for the search results.

If you'd like to make some changes to the dashboard you can click the Edit toggle to on, which will allow you to interact with the panels of the dashboard. In this mode you can move the panels around, change the visualizations, etc. You can also add new panels.



If later you discover another search that you would like to add to this dashboard, you can do that in one of two ways. Either from the dashboard itself while in edit mode, or you can follow the previous process that we used to create the dashboard, only this time select "add to existing dashboard" on the dashboard screen in the wizard.

For more information on creating dashboards, go [here](http://docs.splunk.com/Documentation/Splunk/4.3.3/Developer/DashboardIntro).