Splunk Fundamentals 1

Outline

Module 1-2: Introducing Splunk and Splunk's Components
Module 3: Installation
Module 4: Inputs
Module 5: Searching
Module 6: Using Fields in Searches
Module 7: Best Practices for Searching
Module 8: Splunk's Search Language
Module 9: Transforming Commands
Module 10: Creating Reports and Dashboards
Module 11: Using Pivot
Module 12: Creating and Using Lookups
Module 13: Creating Scheduled Reports and Alerts

Modules 1-2: Introducing Splunk and Splunk's Components

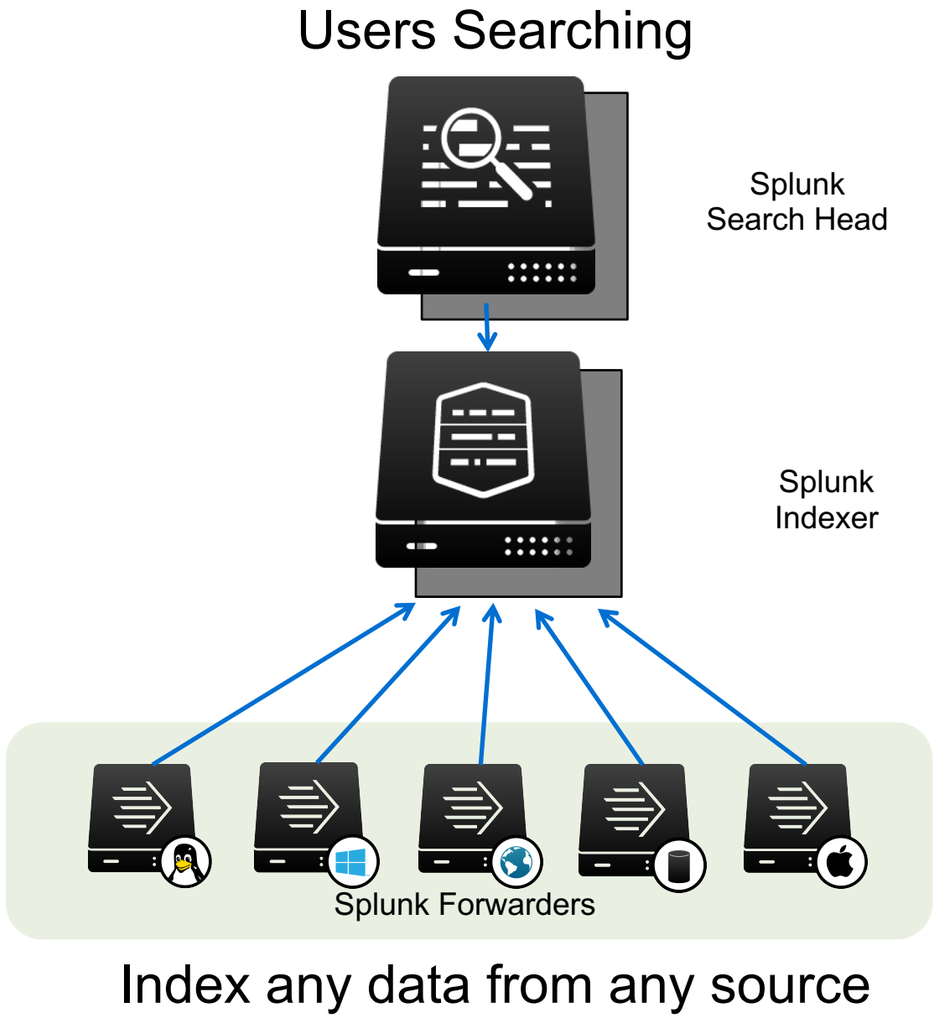
Module Objectives

- Understand the uses of Splunk
- Define Splunk apps
- Learn basic navigation in Splunk

Got Data?

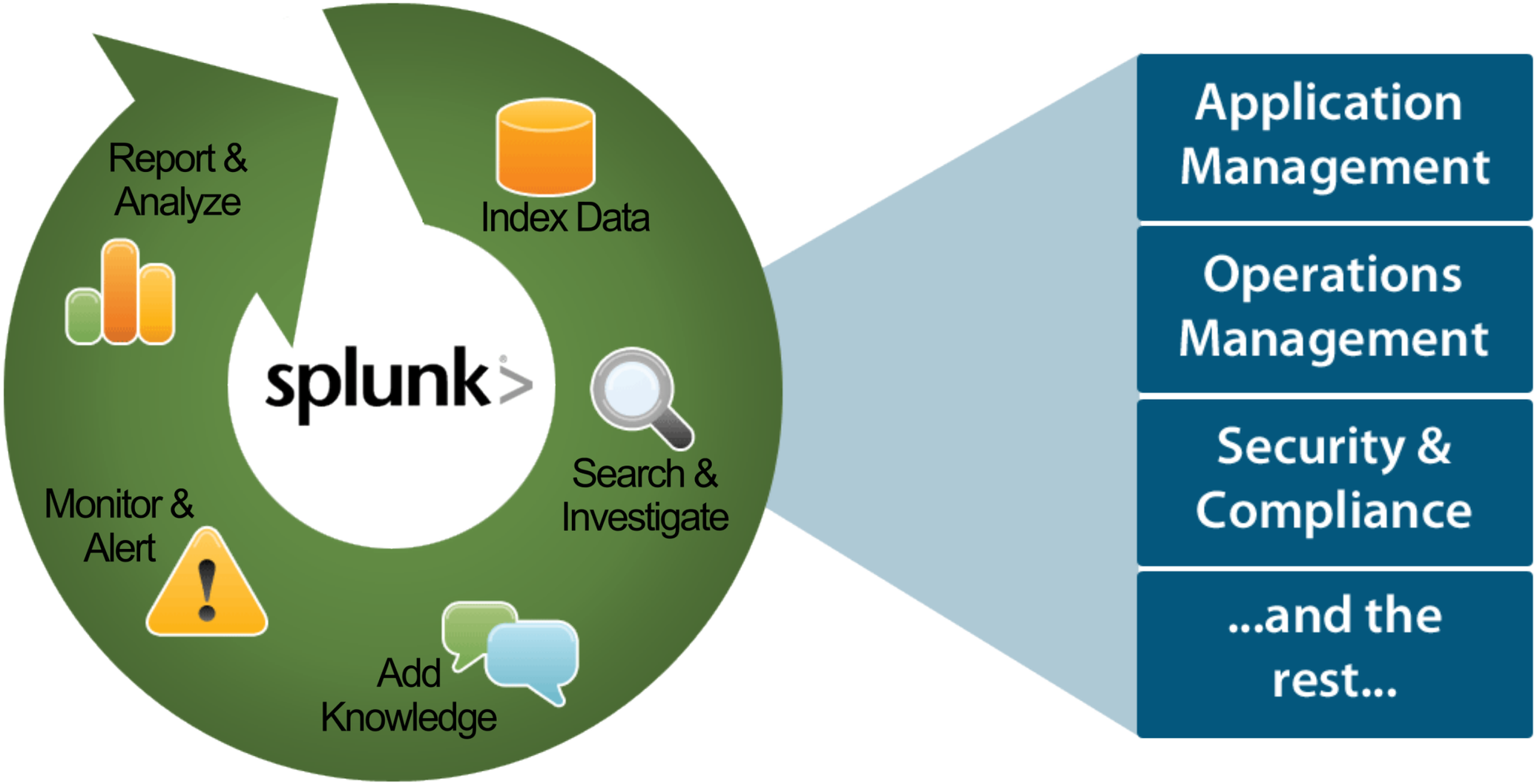
- Computers
- Network devices
- Virtual machines
- Internet devices
- Communication devices
- Sensors
- Databases
- ***Any source***

- Logs
- Configurations
- Messages
- Call detail records
- Clickstream
- Alerts
- Metrics
- Scripts
- Changes
- Tickets
- ***Any data***



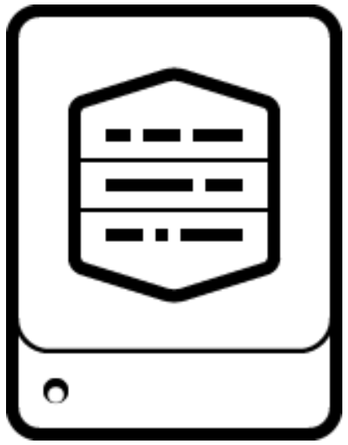
Index any data from any source

One Splunk. Many Uses.



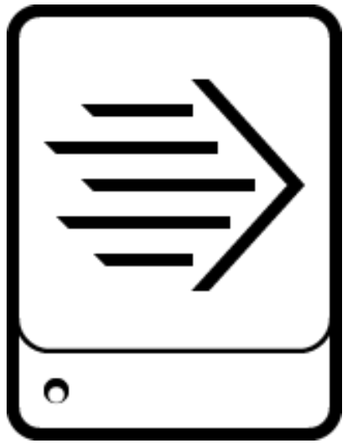
Splunk Components

- Splunk is comprised of three main processing components:



Indexer

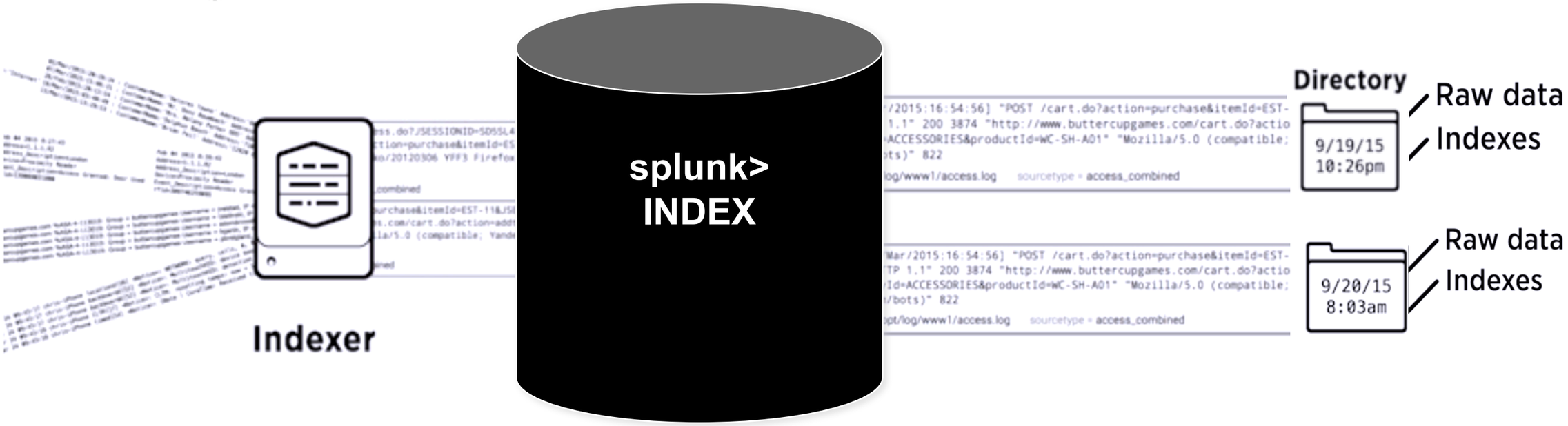
Search Head



Forwarder

Splunk Components - Indexer

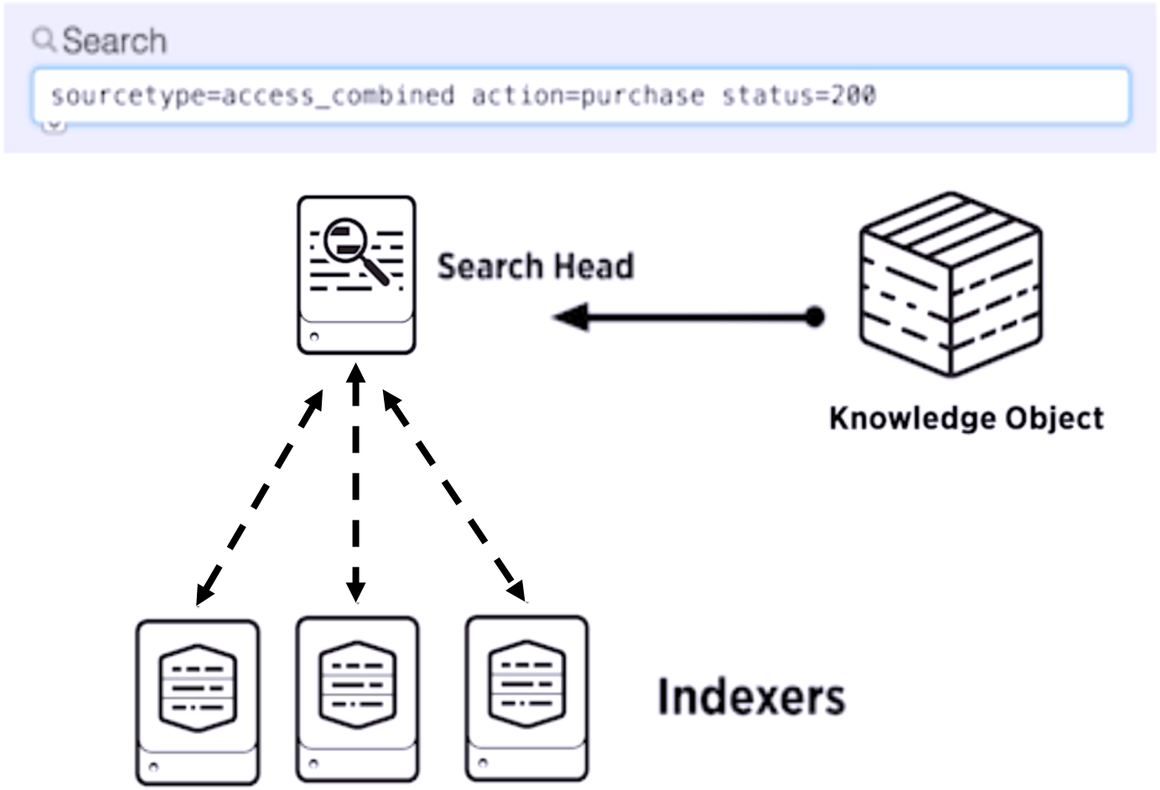
- Processes machine data, storing the results in Indexes as events, enabling fast search and analysis



- As the Indexer indexes data, it creates a number of files organized in sets of directories by age
  - Contains raw data (compressed) and Indexes (points to the raw data)

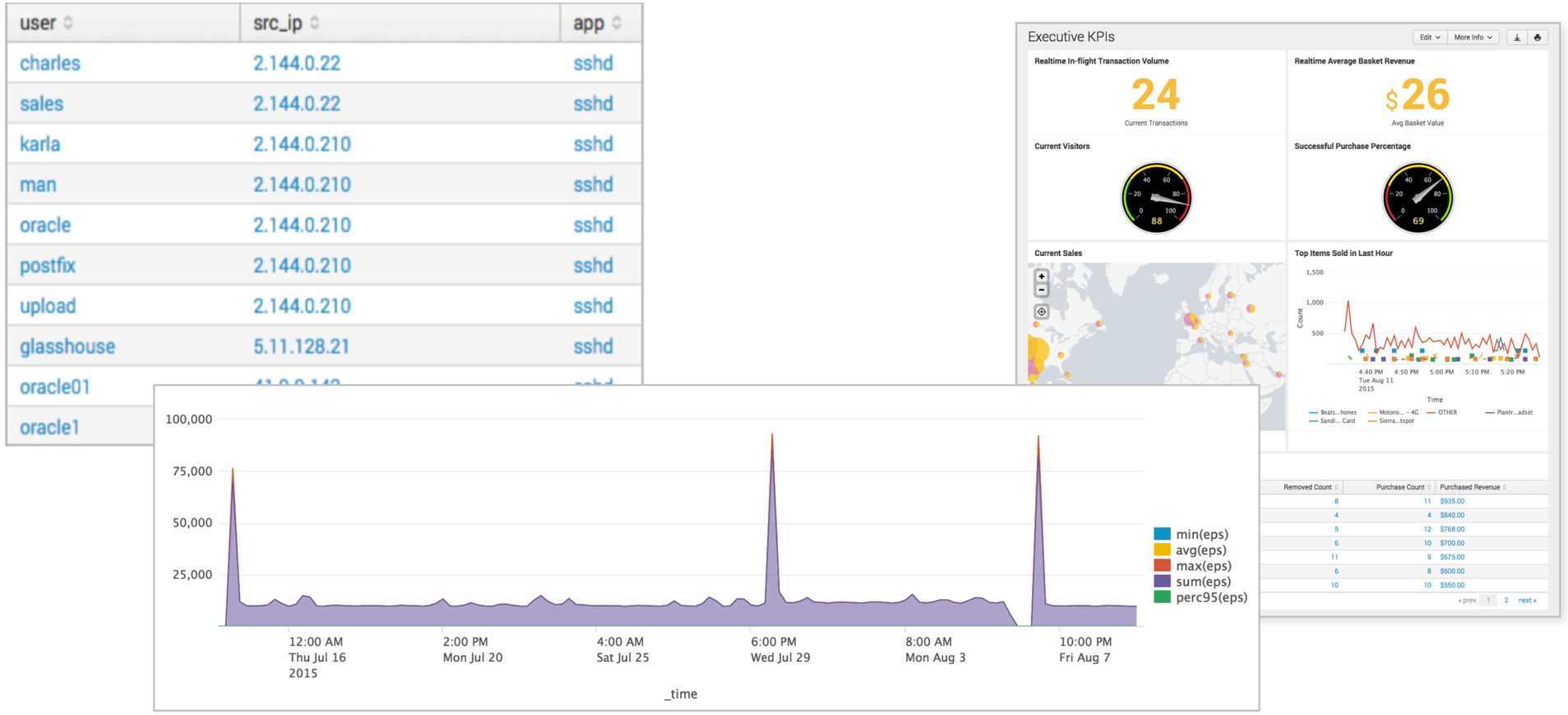
Splunk Components – Search Heads

- Allows users to use the Search language to search the indexed data
- Distributes user search requests to the Indexers
- Consolidates the results and extract field value pairs from the events to the user
- Knowledge Objects on the Search Heads can be created to extract additional fields and transform the data without changing the underlying Index data



Splunk Components – Search Heads (cont.)

- Search Heads also provide tools to enhance the search experience such as reports, dashboards and visualizations

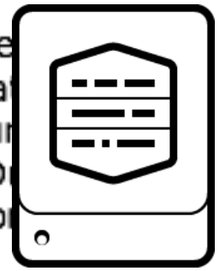


Splunk Components – Forwarders

- Splunk Enterprise instances that consume and send data to the index
- Require minimal resources and have little impact on performance
- Typically reside on the machines where the data originates
- Primary way data is suppled for indexing

Web Server with Forwarder Instance installed

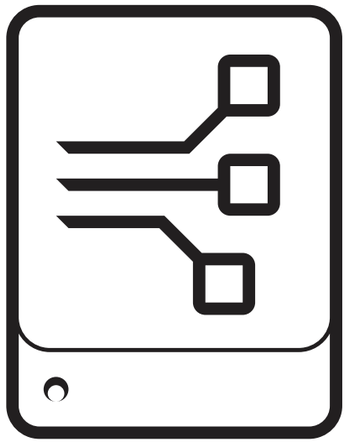
IP = 10.3.10.6, Session disconnected. Session type = TPsecOve
IP = 10.1.10.216, Session connected. Session type = SSL, Dura
s, IP = 10.1.10.133, Session connected. Session type = IKE, Du
i, IP = 10.3.10.18, Session disconnected. Session type = IKE, D
= 10.1.10.211, Session connected. Session type = SSL, Duratio



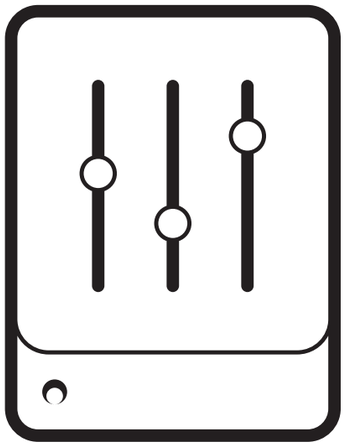
Indexer

Additional Splunk Components

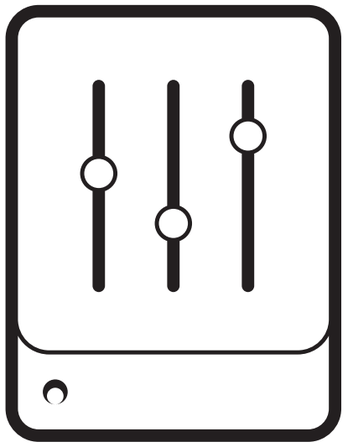
- In addition to the three main Splunk processing components, there are some less-common components including :



Deployment Server



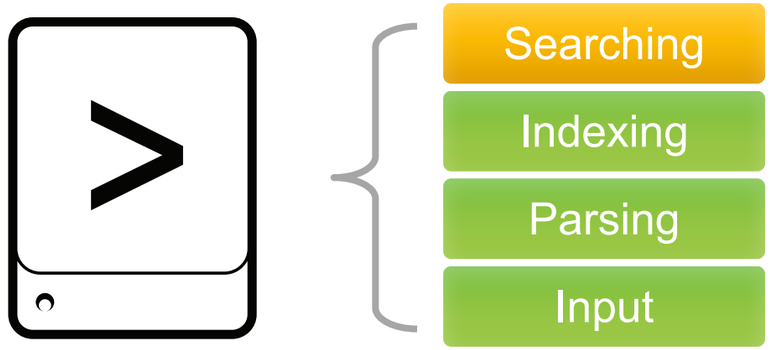
Cluster Master



License Master

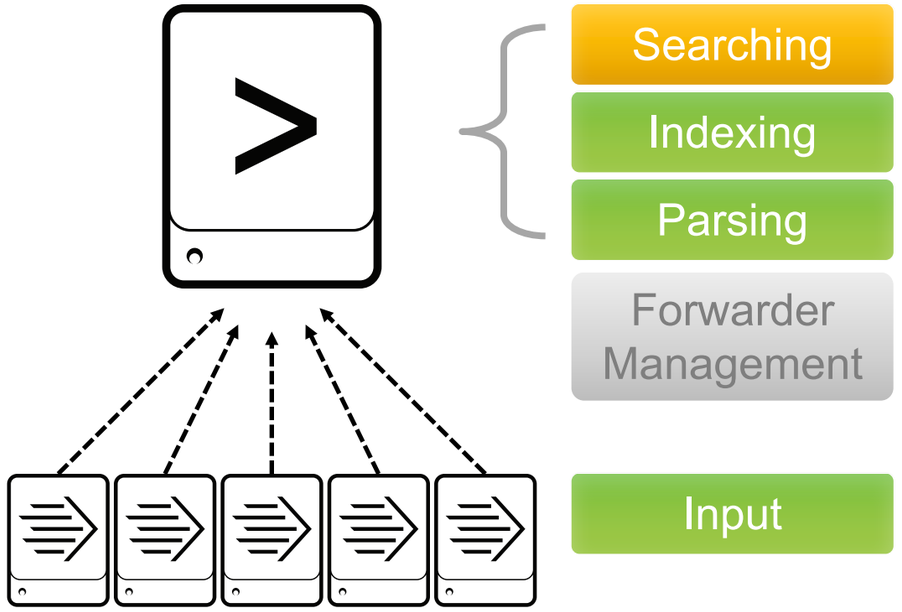
Splunk Deployment – Standalone

- **Single Server**
  - All functions in a single instance of Splunk
  - For testing, proof of concept, personal use, and learning
  - This is what you get when you download Splunk and install with default settings
- Recommendation
  - Have at least one test/development setup at your site



Splunk Deployment – Basic

- **Splunk server**
  - Similar to server in standalone configuration
  - Manage deployment of forwarder configurations
- **Forwarders**
  - Forwarders collect data and send it to Splunk servers
  - Install forwarders at data source (usually production servers)

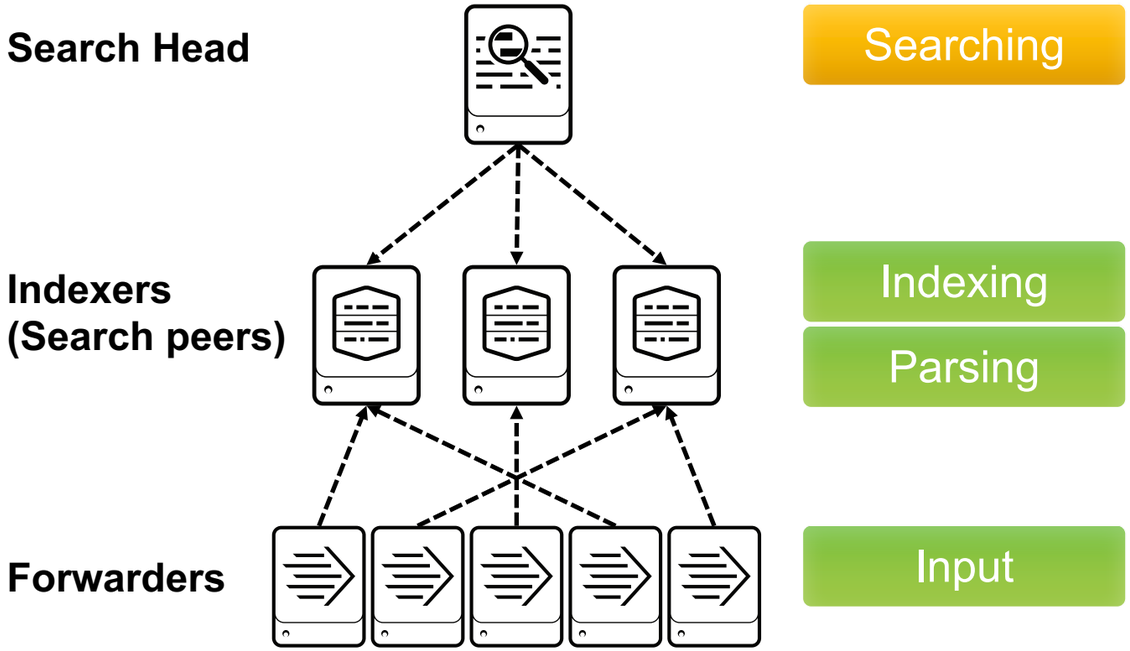


Basic Deployment for organizations:

- Indexing less than 20GB per day
- With under 20 users
- Small amount of forwarders

Splunk Deployment – Multi-Instance

- Increases indexing and searching capacity
- Search management and index functions are split across multiple machines

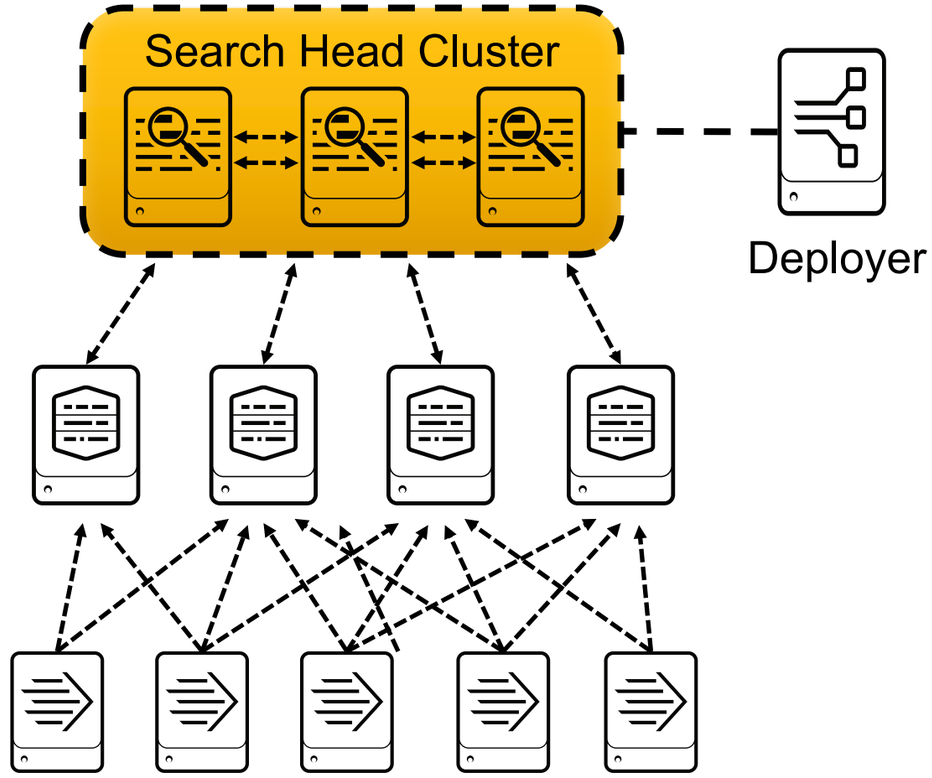


Deployment for organizations:

- Indexing up to 100 GB per day
- Supports 100 users
- Supports several hundred forwarders

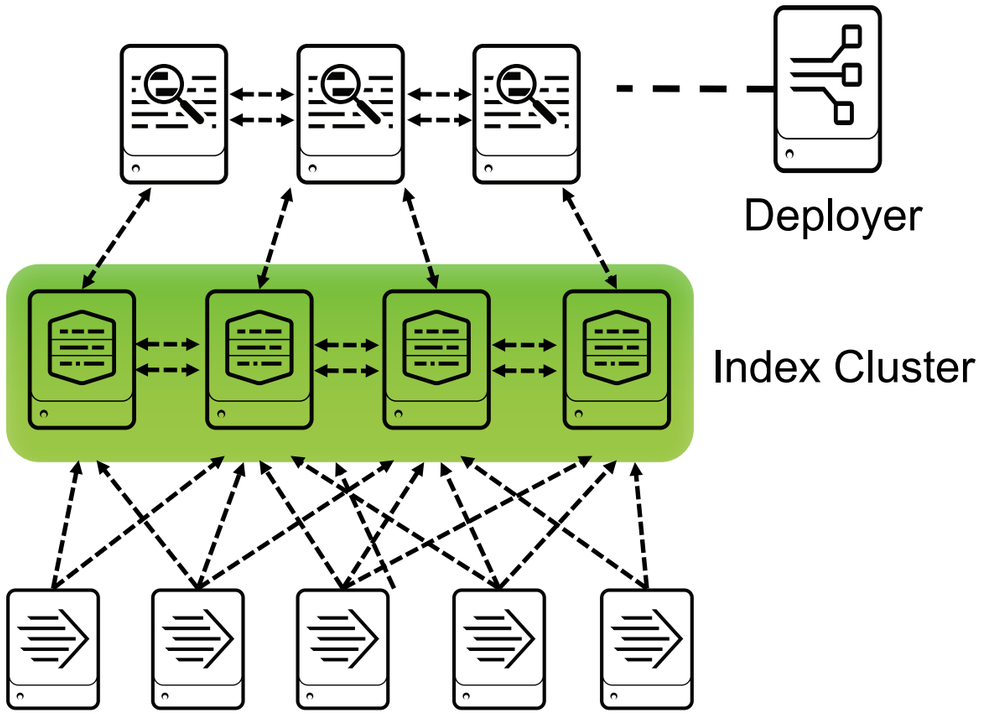
Splunk Deployment – Increasing Capacity

- Adding a Search Head Cluster:
  - services more users for increased search capacity
  - allows users and searches to share resources
  - Coordinate their activities to handle search requests and distribute the requests across the set of indexers
- Search Head Clusters require a minimum of three Search Heads
- A Deployer is used to manage and distribute apps to the members of the Search Head Cluster



Splunk Deployment – Index Cluster

- Traditional Index Clusters:
  - Configured to replicate data
  - Prevent data loss
  - Promote availability
  - Manage multiple indexers
- Non-replicating Index Clusters
  - Offer simplified management
  - Do not provide availability or data recovery



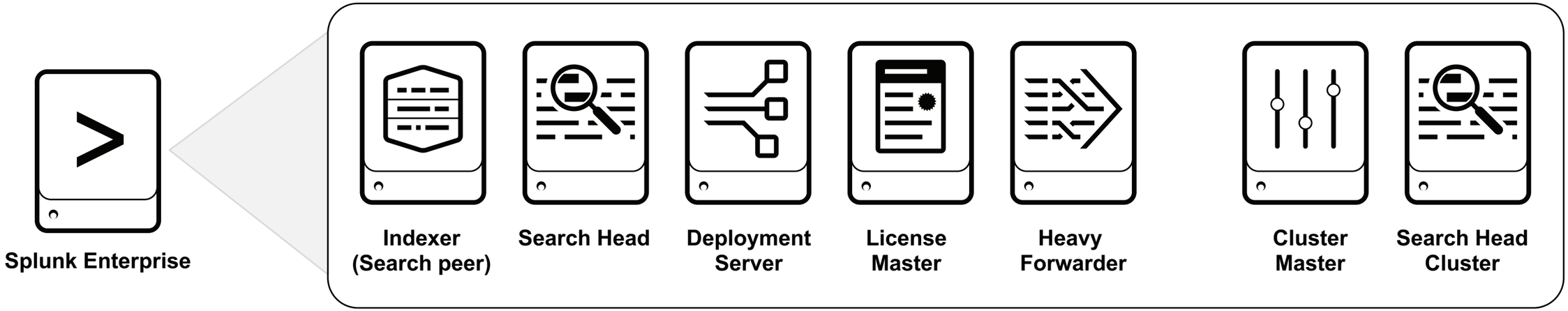
Module 3: Installation

Module Objectives

- Describe Splunk installation
- Describe Splunk component installation
- Using Splunk Web Admin
- Identify common Splunk commands
- Identify Splunk directory structure

Splunk Enterprise Install Package

- There are multiple Splunk components installed from the Splunk Enterprise package:



Splunk Enterprise Installation Overview

- Verify required ports are open (splunkweb, splunkd, forwarder) and start-up account
- Download Splunk Enterprise from www.splunk.com/download
- Installation: (as account running Splunk)
  - ***NIX** – un-compress the **.tar.gz** file in the path you want Splunk to run from
  - **Windows** – execute the **.msi** installer and follow the wizard steps
- Complete installation instructions at:
  docs.splunk.com/Documentation/Splunk/latest/Installation/Chooseyourplatform
- After installation:
  - Splunk starts automatically on Windows
  - Splunk must be manually started on *NIX until **boot-start** is enabled

Splunk Component Installation Overview

- Installing Splunk Enterprise as an Indexer or Search Head is identical an installing a single deployment instance
- The difference happens at a configuration level
  - Installation as configuration is an iterative and ongoing event as you build and scale your deployment
  - Administrators need to be in control of the environment to fulfill emerging needs
  - Before installing Indexers or Search Heads, be sure to keep in mind the different hardware requirements

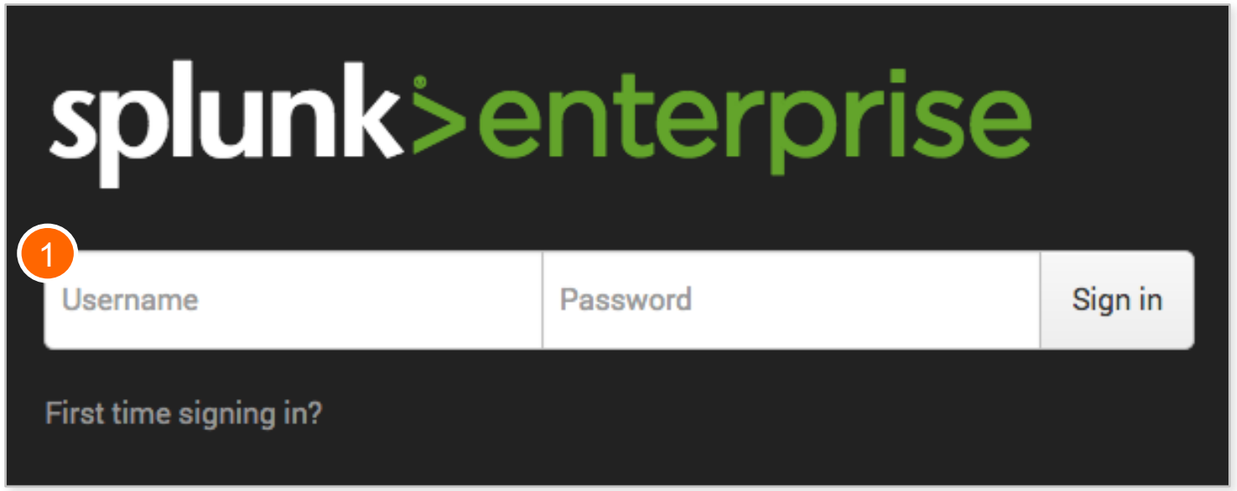
Common Splunk Commands

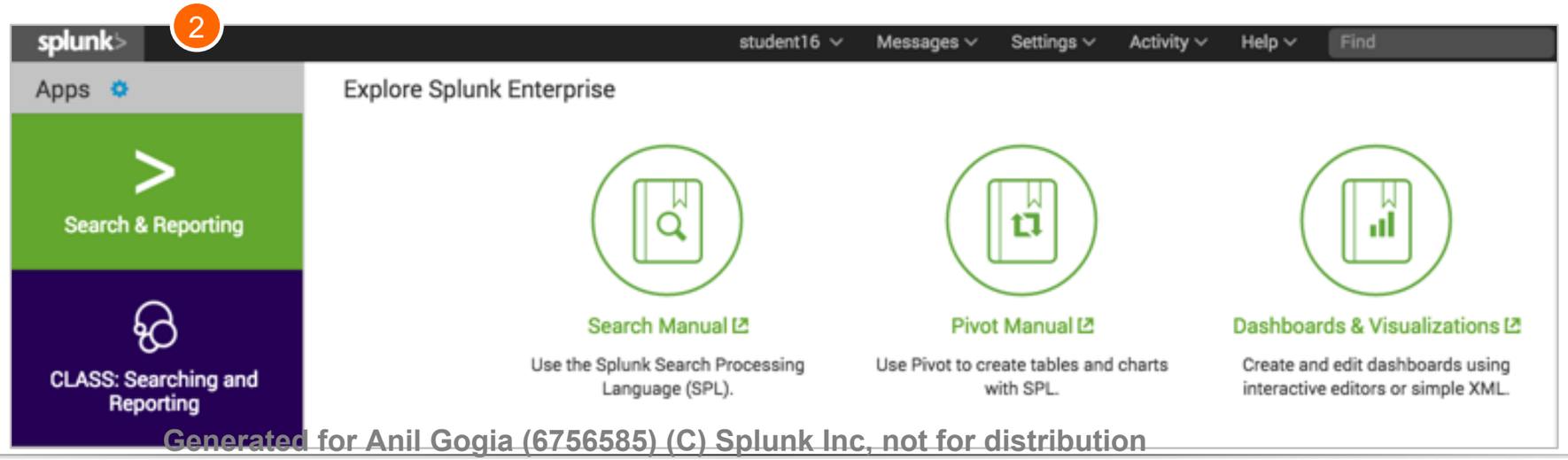
- **`splunk`** is the program in the **`bin`** directory to run the CLI

Command	Operation
`splunk help`	Display a usage summary
`splunk [start | stop | restart]`	Manages the Splunk processes
`splunk start –-accept-license`	Automatically accept the license without prompt
`splunk status`	Display the Splunk process status
`splunk show splunkd-port`	Show the port that the **splunkd** listens on
`splunk show web-port`	Show the port that Splunk Web listens on
`splunk show servername`	Show the servername of this instance
`splunk show default-hostname`	Show the default host name used for all data inputs
`splunk enable boot-start -user`	Initialize script to run Splunk Enterprise at system startup

Logging In

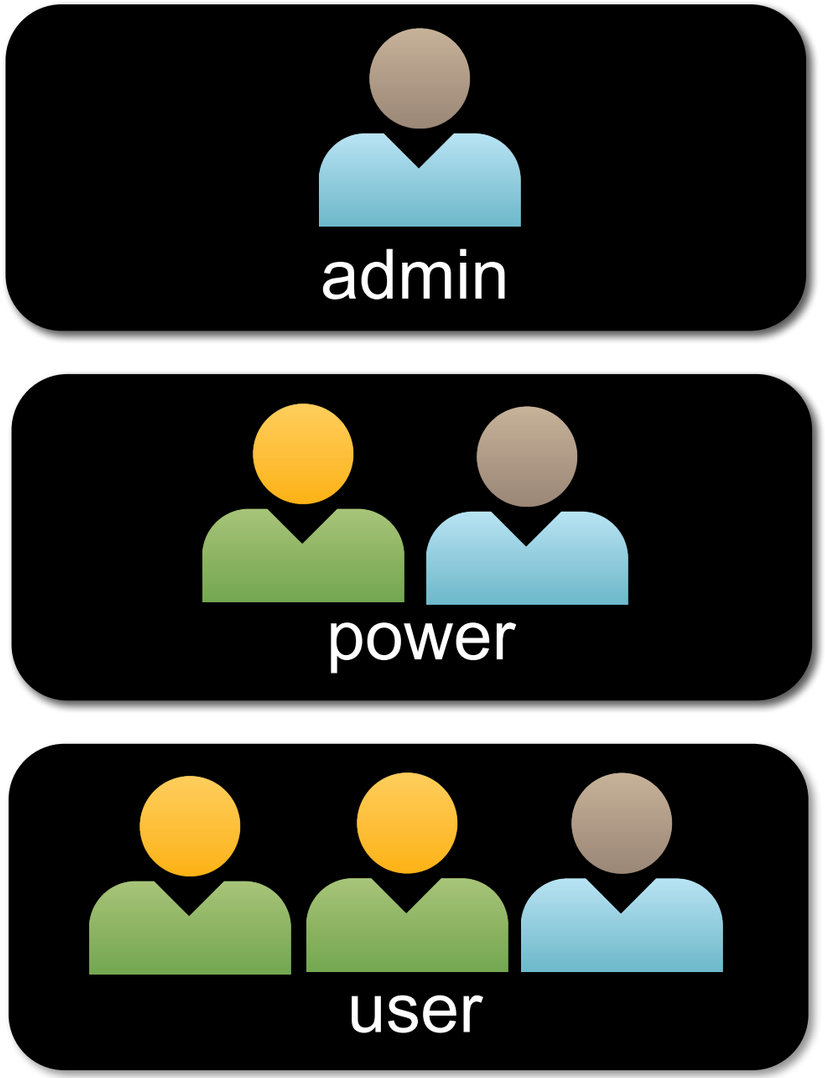
1. Log in to Splunk with a web browser
2. Based on your default app, its main view appears
   - The Home view is shown here
   - You or your organization may change your default app





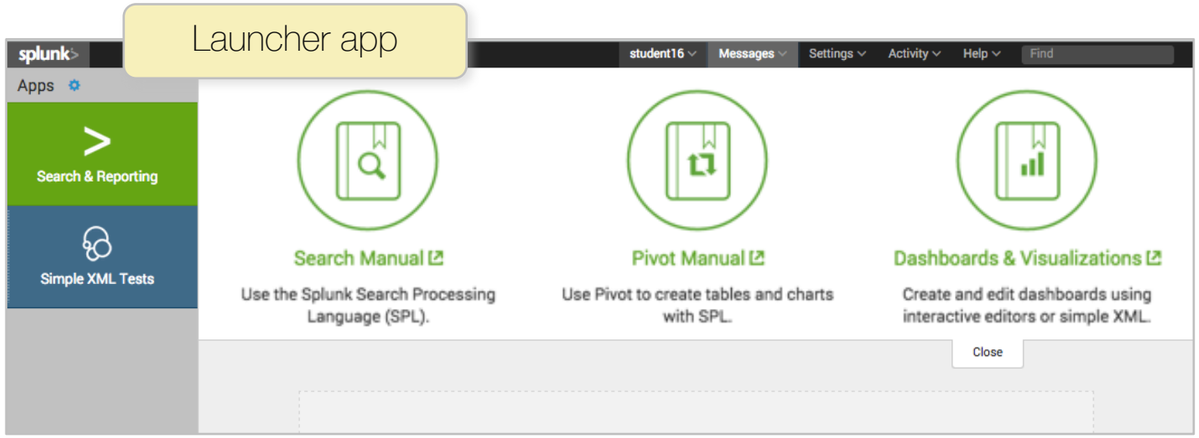
Users and Roles

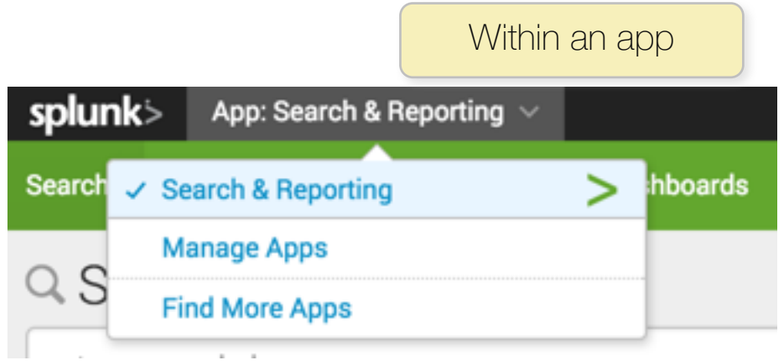
- Splunk users are assigned roles
  - Roles determine capabilities and data access
- Out of the box, there are 3 main roles:
  - Admin
  - Power
  - User
- Splunk administrators can create additional roles
- The account you use for the lab exercises has the **power** role



What Are Apps?

- Apps allow different workspaces, tailored to a specific use case or user role, to exist on a single Splunk instance
- This class focuses on the Search & Reporting app (also called the Search app)
- Administrators can create or install additional apps to your Splunk instance from http://splunkbase.splunk.com



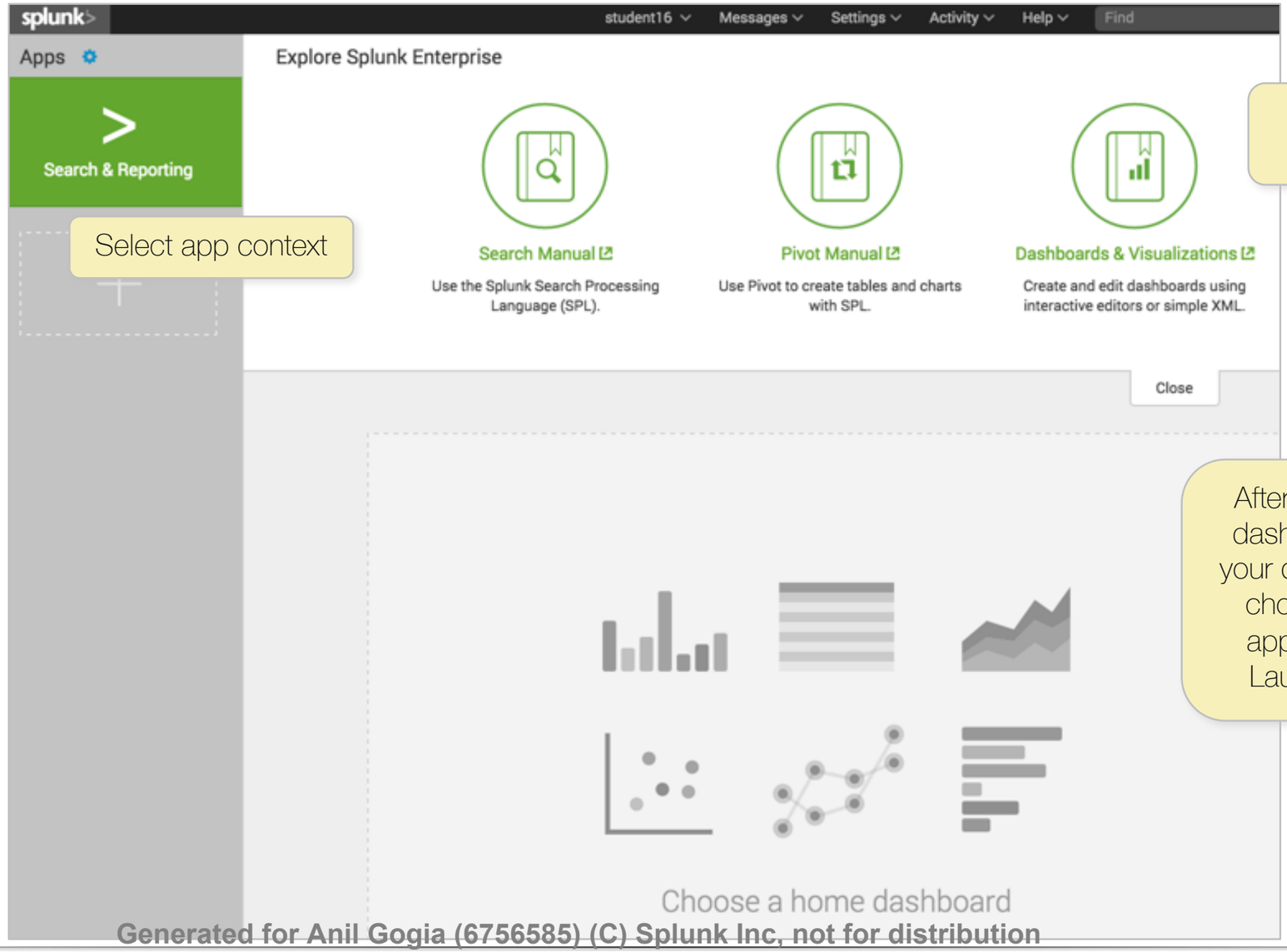


Note

Simply put, a Splunk app is a collection of files. Some apps are more robust and may contain data inputs, knowledge objects, and UI elements.
http://docs.splunk.com/Documentation/Splunk/latest/Admin/Whatsanapp

Home App

Click the Splunk logo to return to the app that is set as your default app; the default is the Launcher app



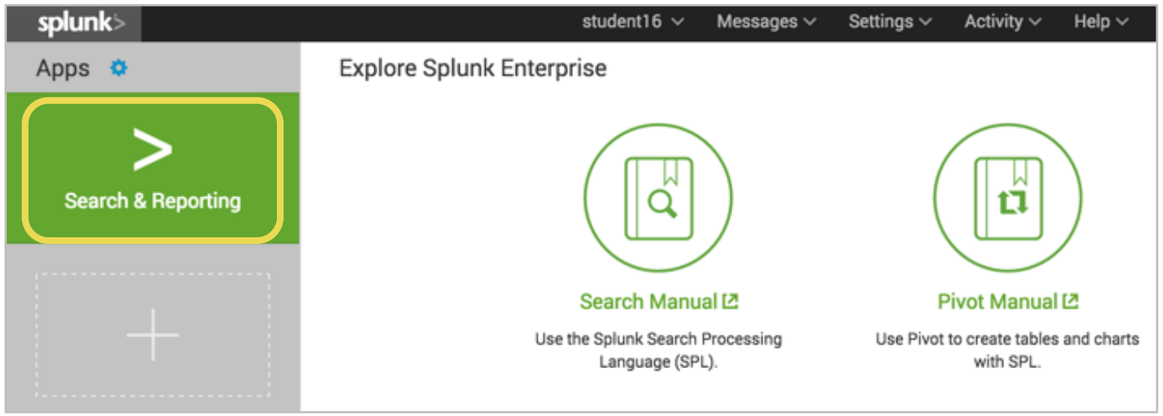
Select app context

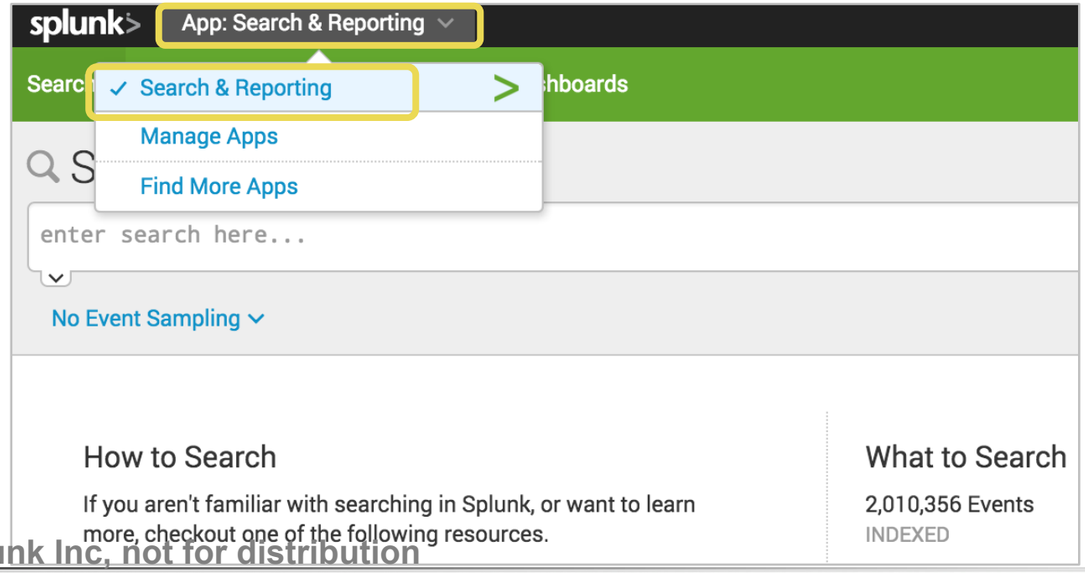
Links to several helpful resources

After you've built dashboards with your data, you can choose one to appear in your Launcher app

Search & Reporting App Overview

- Provides a default interface for searching and analyzing data
- Enables you to create knowledge objects, reports, and dashboards
- Access by selecting the **Search & Reporting** button on the Home view or from an app view, select **Apps**, then select **Search & Reporting**





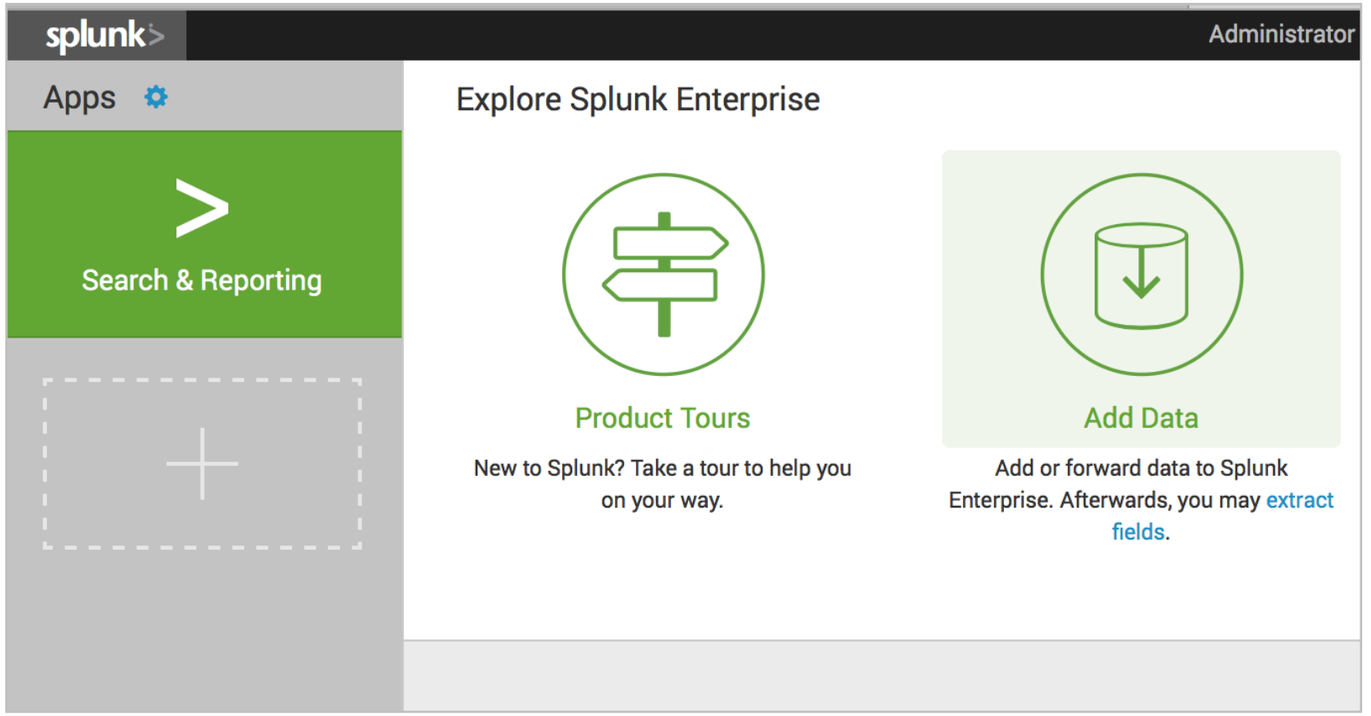
Module 4: Inputs

Module Objectives

- Identify the input types
- Uploading data using Splunk Web
- Using the Monitor option

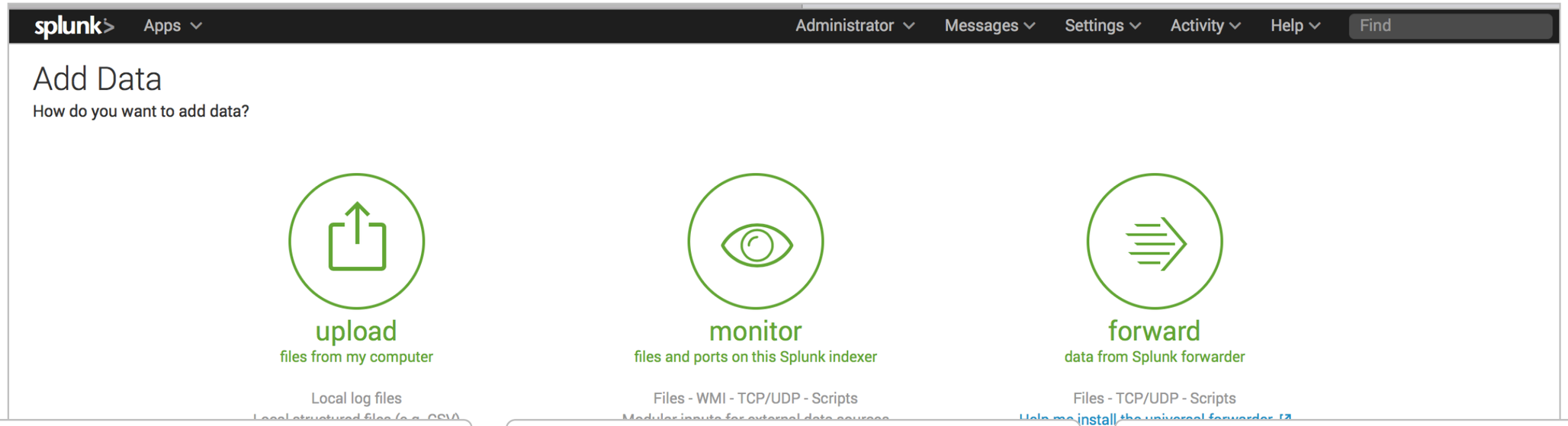
Adding Data

Administrators can access the Add Data menu by clicking the Add Data icon located on the Splunk Enterprise home app



Add Data Menu

Add Data menu provides three options depending on the source to be used



Upload Option

Upload option allows users to upload local files that only get indexed once. Useful for testing or data that is created once and never gets updated.

Monitor Option

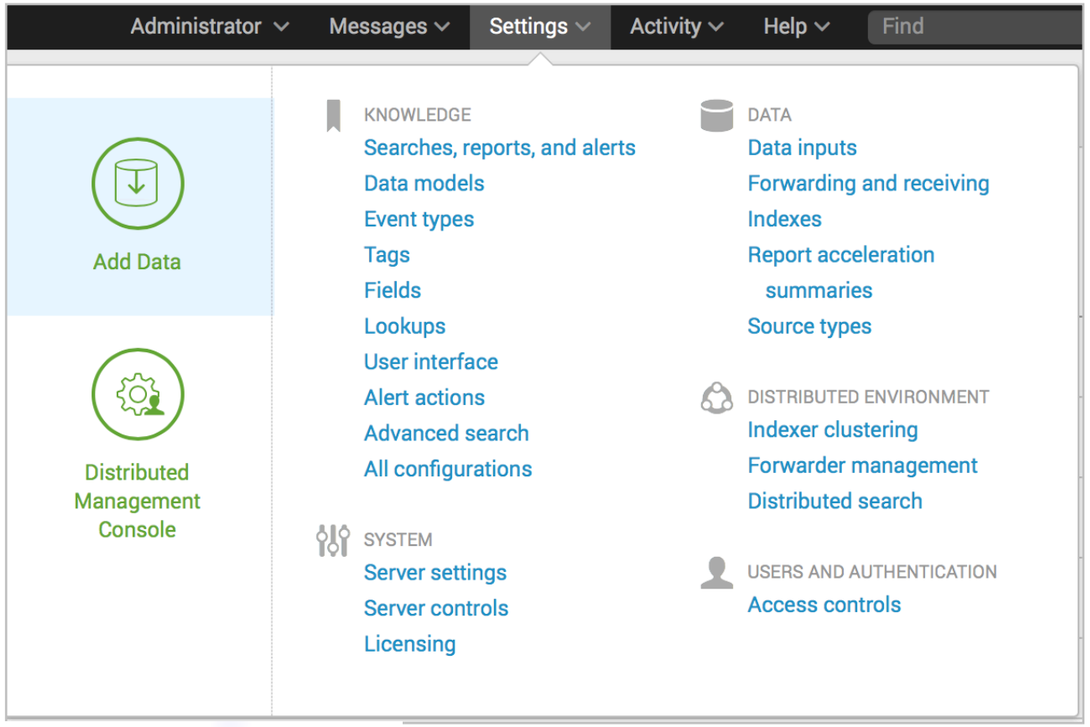
Monitors files, directories, http events, network ports, or data gathering scripts located on Splunk Enterprise instances.

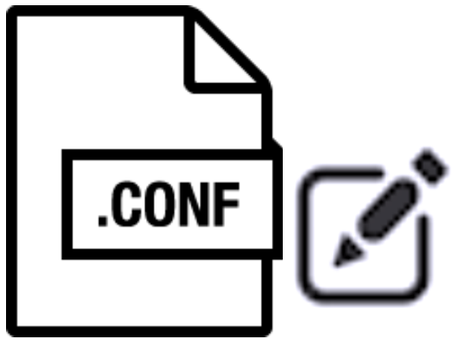
Forward Option

Main source of input in most production environments. Installed on remote machines where data is gathered on forwarded to an index over a receiving port.

Additional Data Input Management Options

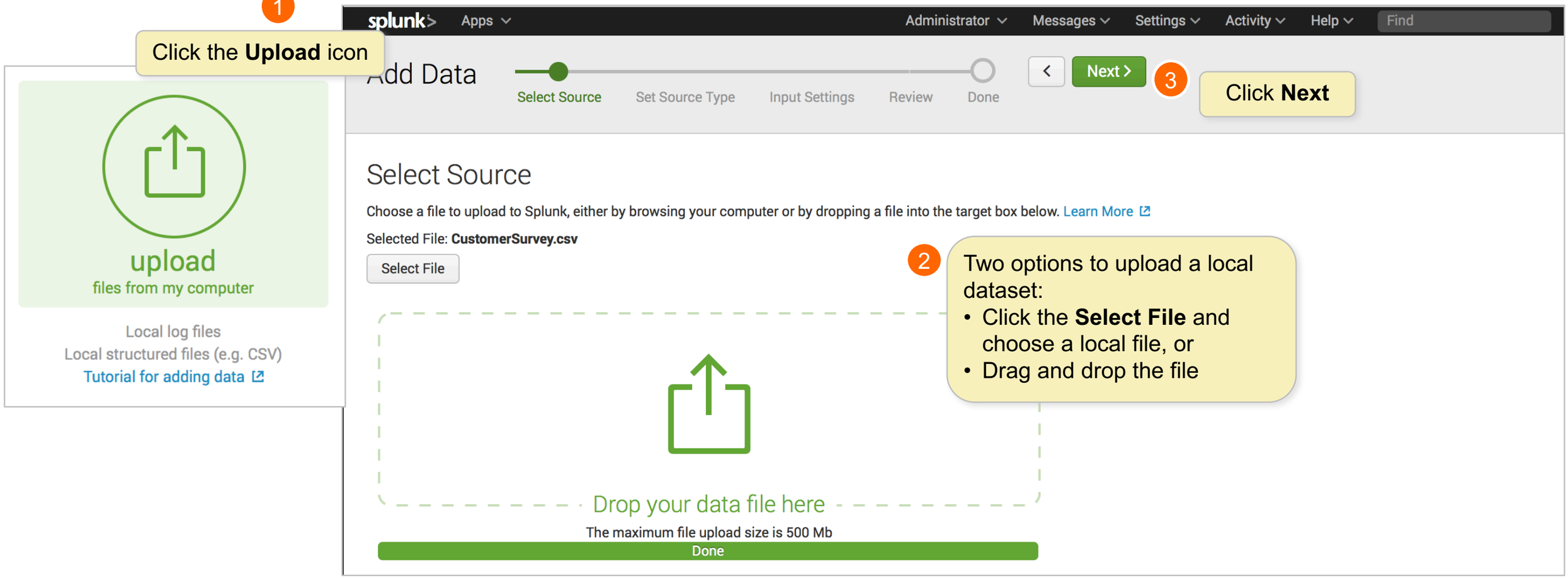
- Data can also be added and managed by:
  - **Settings > Data** Inputs below the **Data** header
  - Splunk CLI
  - Editing .conf files



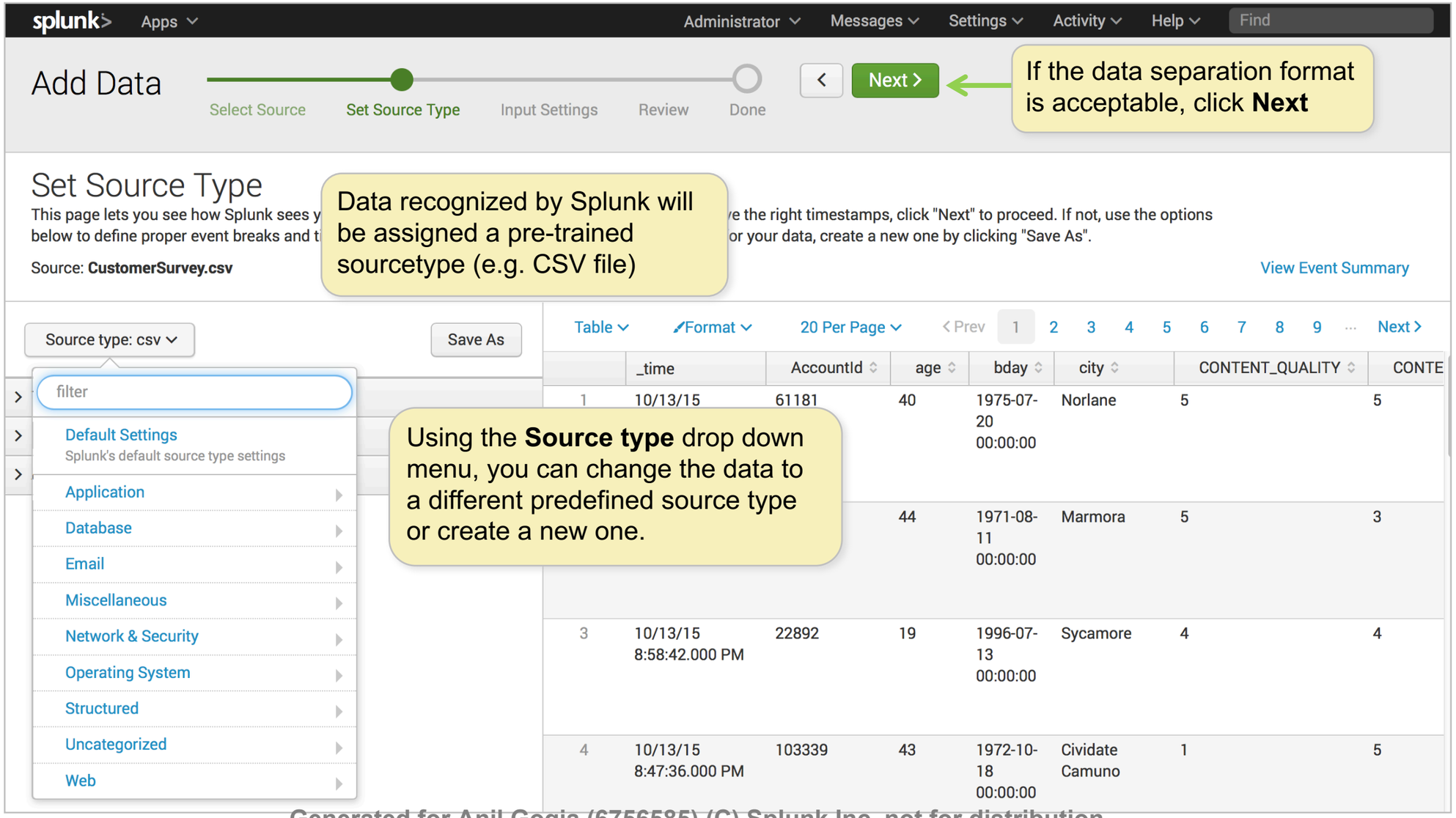


Using the Upload Option

Ideal for testing and searching small datasets that are not updated



Set Sourcetype

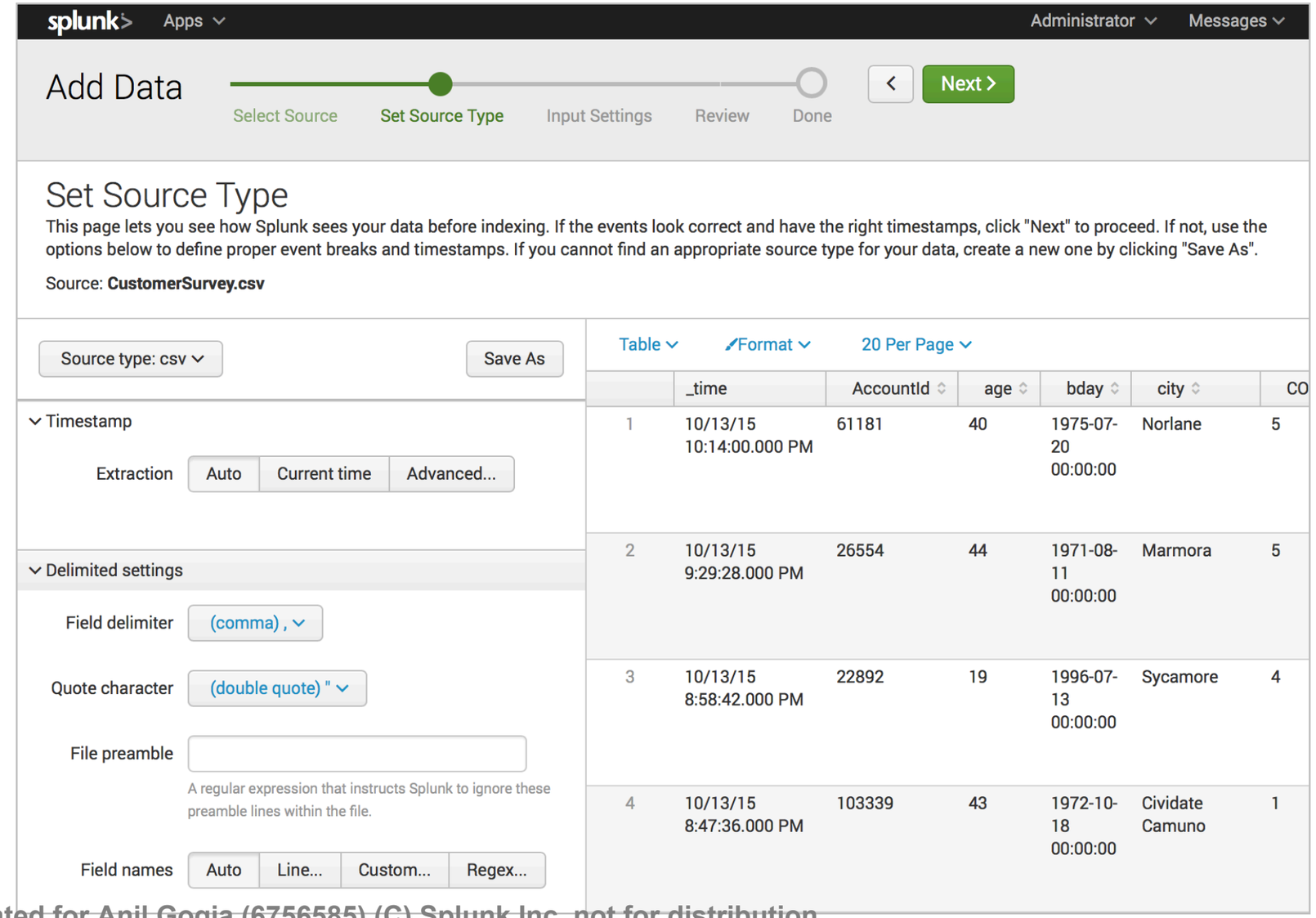


Adjusting Time Stamps and Event Breaks

Adjustments can be made to time stamps and event breaks by using the corresponding drop-down menus.

Note

The menus will change depending on the sourcetype selected.



How Splunk Uses Sourcetypes with Data

- **sourcetype** is Splunk's way of categorizing the type of data
  - Splunk indexing processes frequently reference sourcetype
  - Many searches, reports, dashboards, apps, etc. also rely on sourcetype
  - When using predefined sourcetypes, Splunk knows where to break the event, the location of the timestamp, and automatically create field value pairs.

Table ▾ Format ▾ 20 Per Page ▾ < Prev 1 2 3 4 5 6 7 8 9 ... Next >

	_time	AccountId	age	bday	city	CONTENT_QUALITY	CONTENT_QUANTITY ^	country	DESIGN	email
1	10/13/15 9:29:28.000 PM	26554	44	1971-08-11 00:00:00	Marmora	5	3	US	4	uabbott@yahoo.com
2	10/13/15 8:09:05.000 PM	55084	43	1972-04-09 00:00:00	Tarko-Sale	3	3	RU	4	kernser@gmail.com

How Splunk Uses Sourcetypes with Data (cont.)

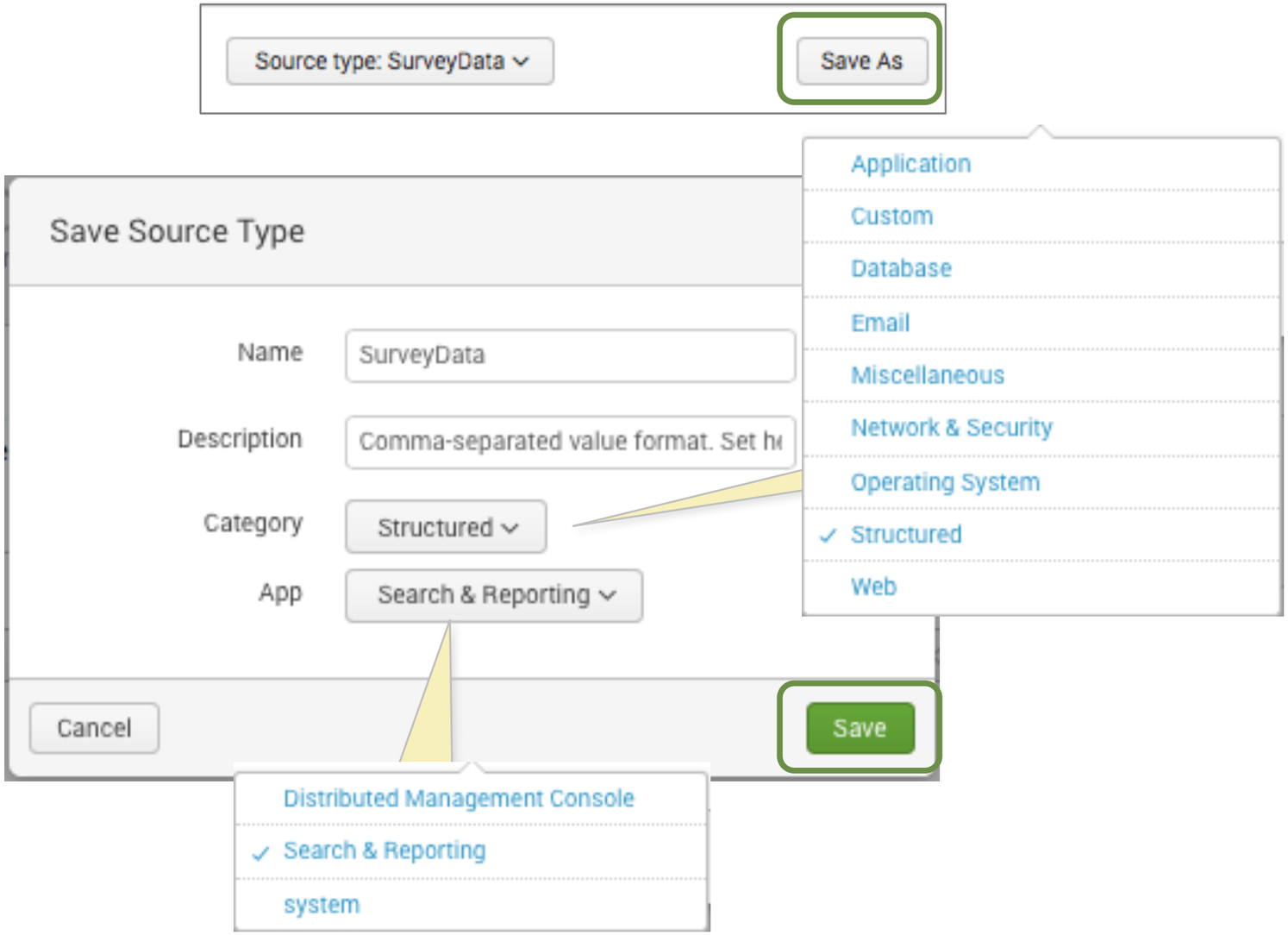
- When Splunk does not have a predefined way to break events, it looks for a time stamp to break the data
  - In the case of multiple time stamps, a regular expression can be used to extract the desired time
  - Regular expressions can be used with any expected patterns in the data to create a line break

List ▾ Format ▾ 20 Per Page ▾ < Prev 1 2 3 4 5 6 7 8 9 ... Next >

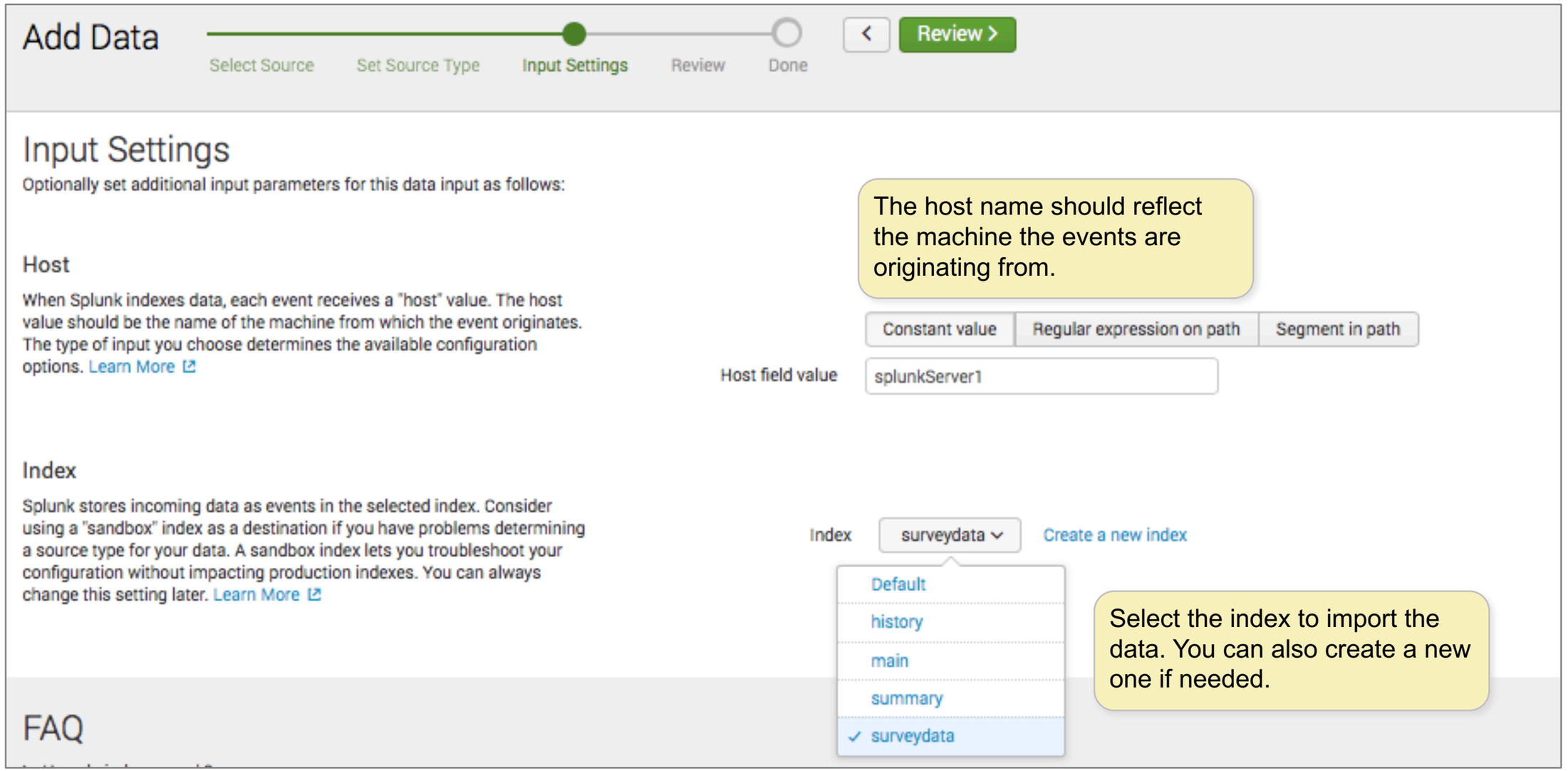
	Time	Event
1 ⚠	6/3/16 12:38:47.000 PM	AccountId,"CONTENT_QUALITY","CONTENT_QUANTITY",DESIGN,JSESSIONID,NAVIGATION,SATISFACTION,"_raw","_time",age,bday,"change_type",city,country,"date_hour","date_mday","date_minute","date_month","date_second","date_wday","date_year","date_zone",email,eventtype,fname,gender,host,index,karma,lat,linecount,lname,lon,punct,region,registered,"site_release",source,sourcetype,"splunk_server","splunk_server_group",tag,"tag::eventtype",timeendpos,timestartpos,username timestamp = none
2	10/13/15 10:14:00.000 PM	61181,5,5,4,SD9SL6FF5ADFF2948092910,3,4,"[13/Oct/2015:22:14:00] AccountId=61181 JSESSIONID=SD9SL6FF5ADFF2948092910 site_release=v4.1
3	10/13/15 5:14:00.000 PM	CONTENT_QUANTITY=5 CONTENT_QUALITY=5 NAVIGATION=3 DESIGN=4 SATISFACTION=4","2015-10-13T15:14:00.000-0700",40,"1975-07-20 00:00:00",,Norlane,AU,22,13,14,october,0,tuesday,2015,local,"jauer@kuvalis.org","nix-all-logs",Arno,M,"customer_survey",main,6807,"-38.1014",2,Mante,"144.3542","[//:::]_=_=_=.=t=t=t=t=",Victoria,"2009-01-26 00:00:00","v4.1","/opt/log/customer_survey/bcg_survey.log","bcg_survey-3","ip-10-222-134-157",,,,21,1,mhammes

Saving Sourcetypes

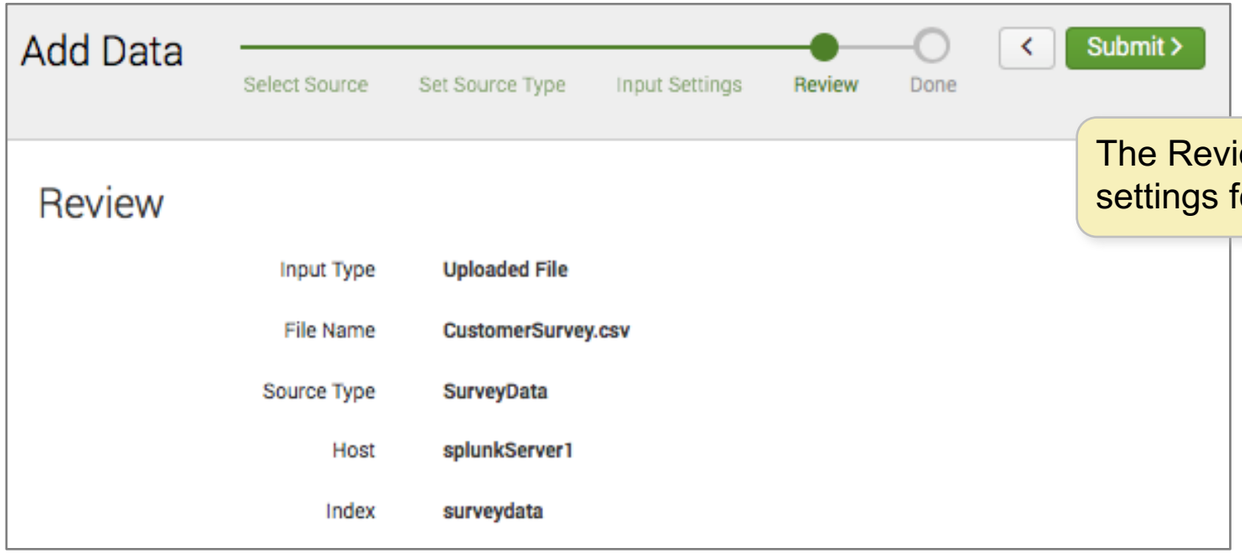
- You have the following options to save sourcetypes if any changes were made
  - Name
  - Description
  - Select a category to store in the predefined menu
  - Select which app context to save it to



Input Settings

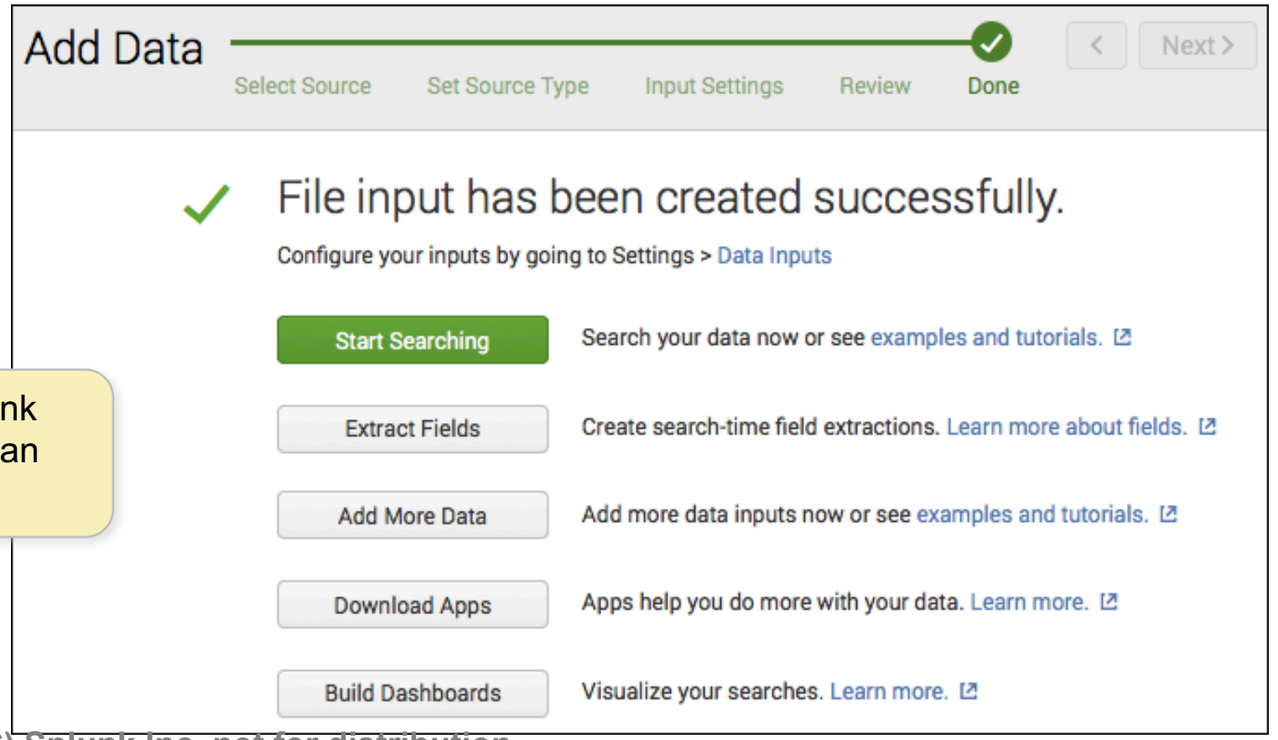


Review and Submit



The Review page displays the settings for our input.

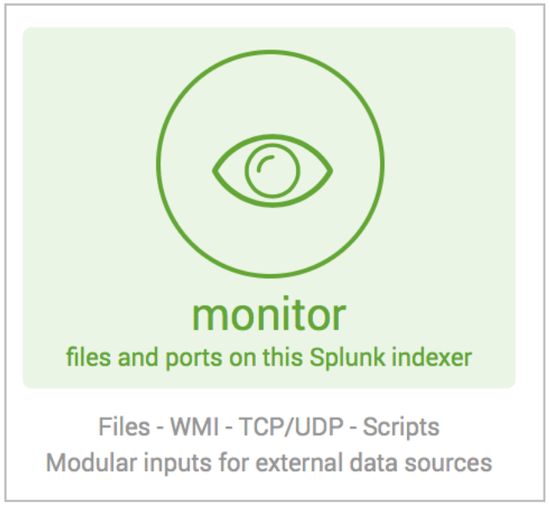
After clicking **Submit,** Splunk indexes the data, and we can start searching

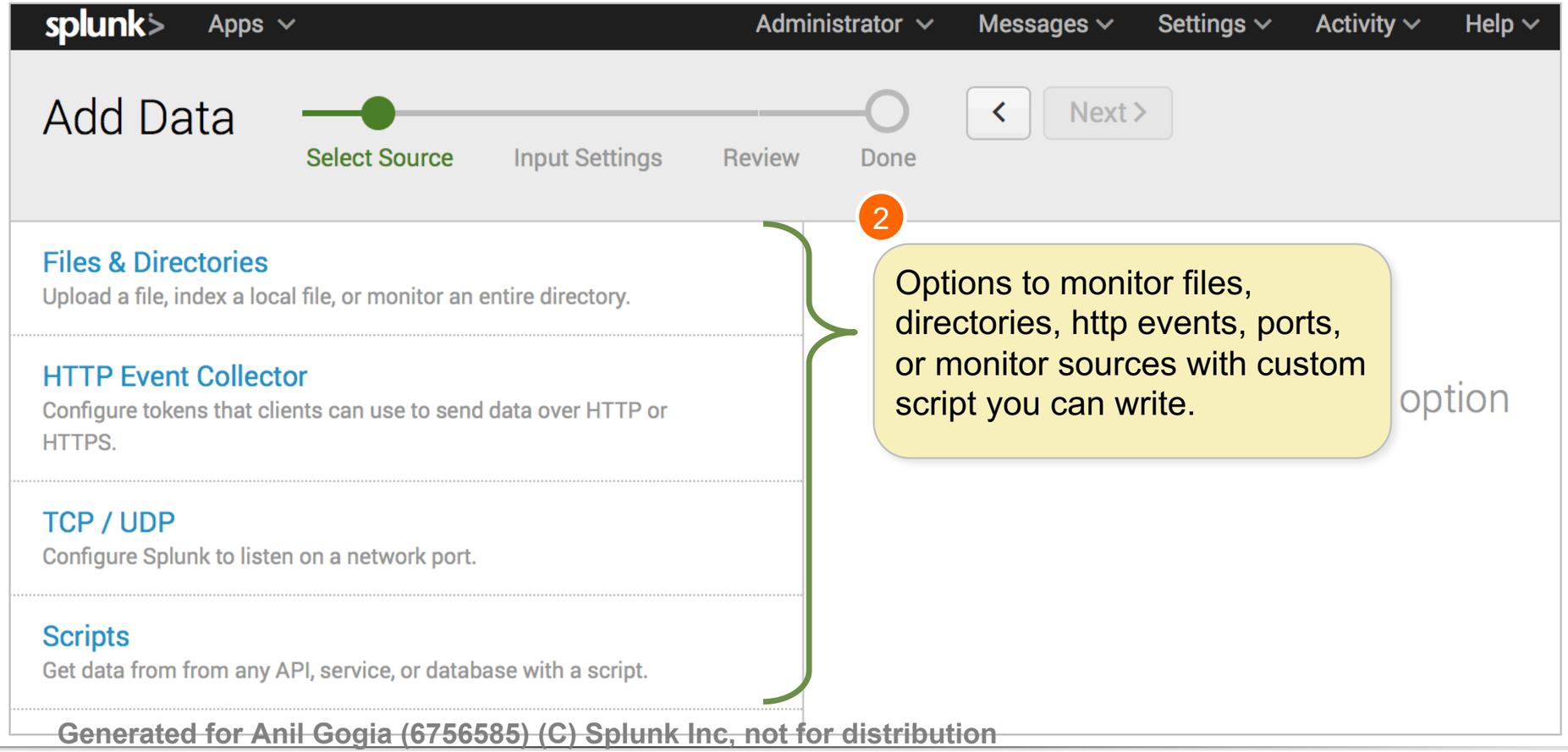


Using the Monitor Option

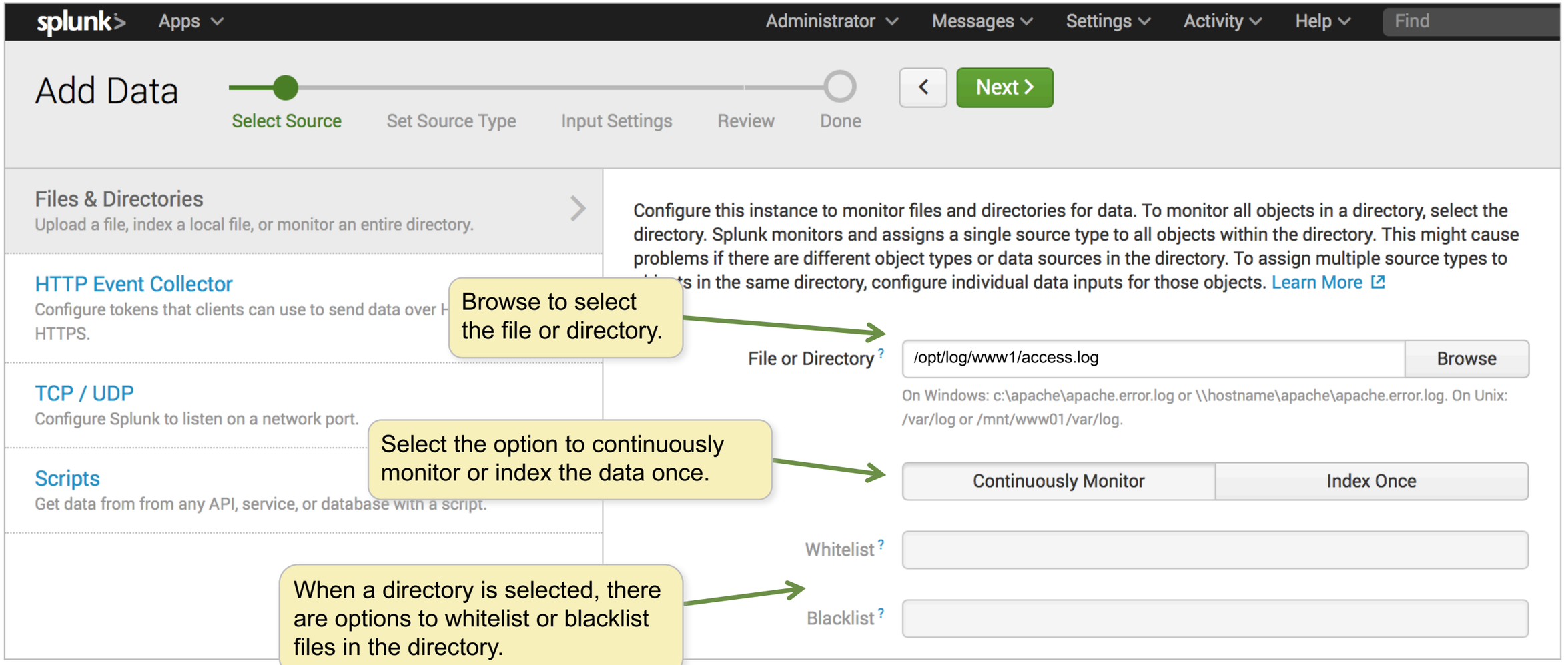
Monitors files, directories, http events, network ports, or data gathering scripts located on a Splunk indexer

1 Click the **Monitor** icon

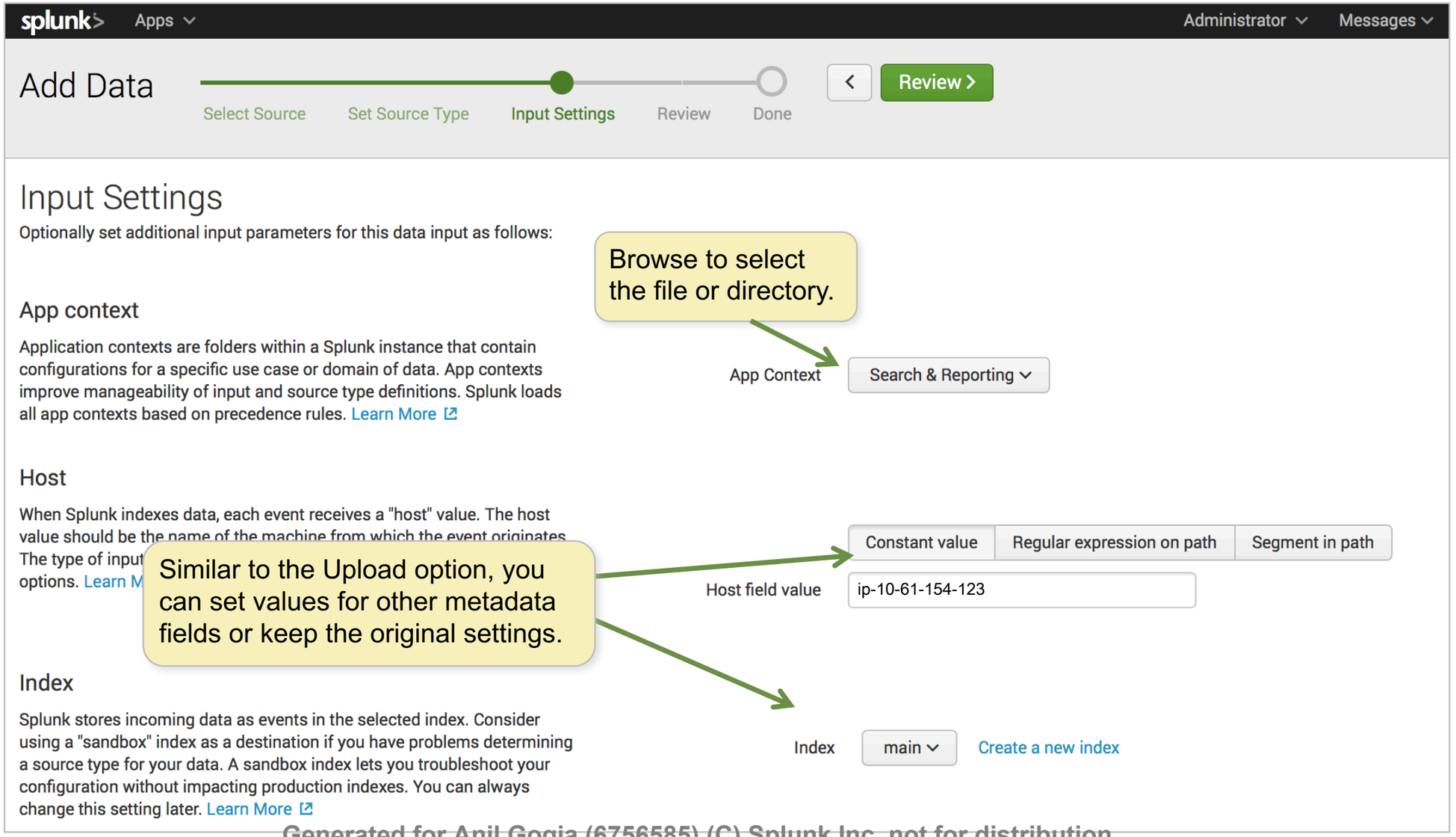




Monitoring Files or Directories

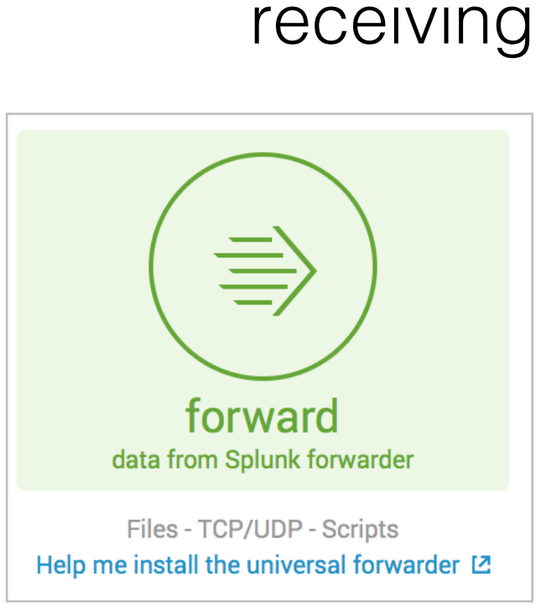


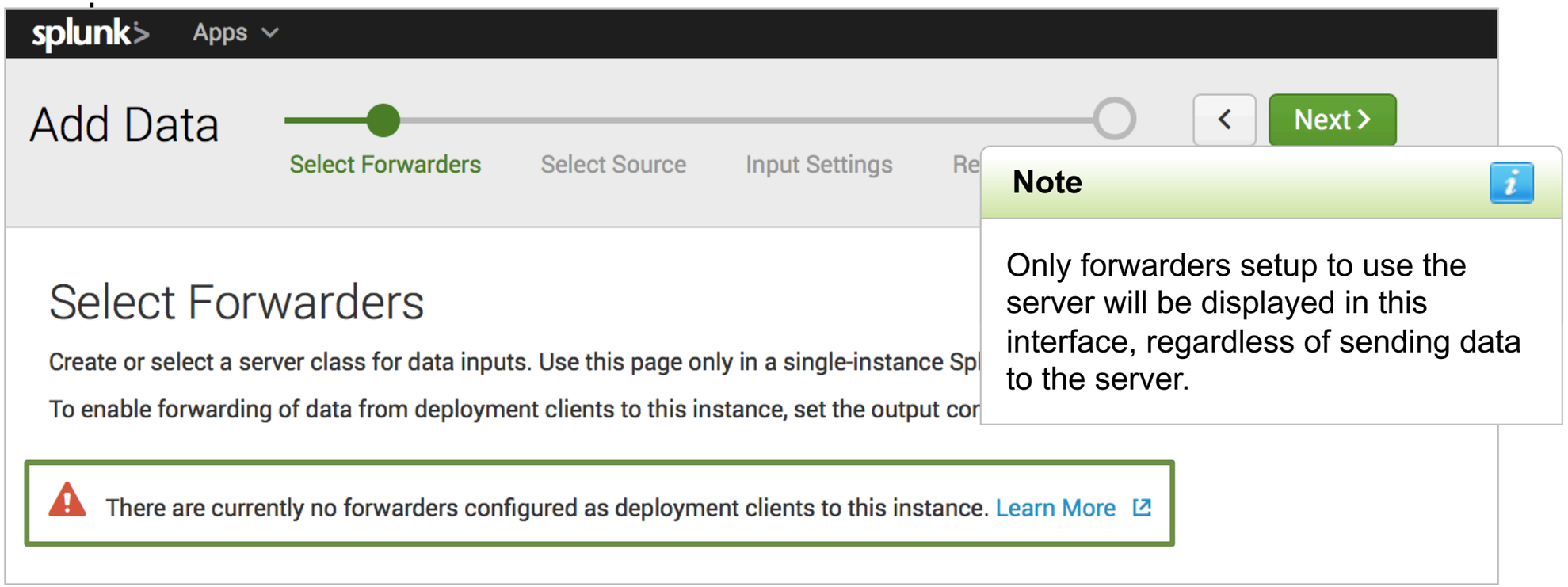
Monitoring Files or Directories (cont)



Using the Forward Option

- Production environments use forwarders as the main source of data input
  - Installed on remote machines and forward data to an indexer over a receiving



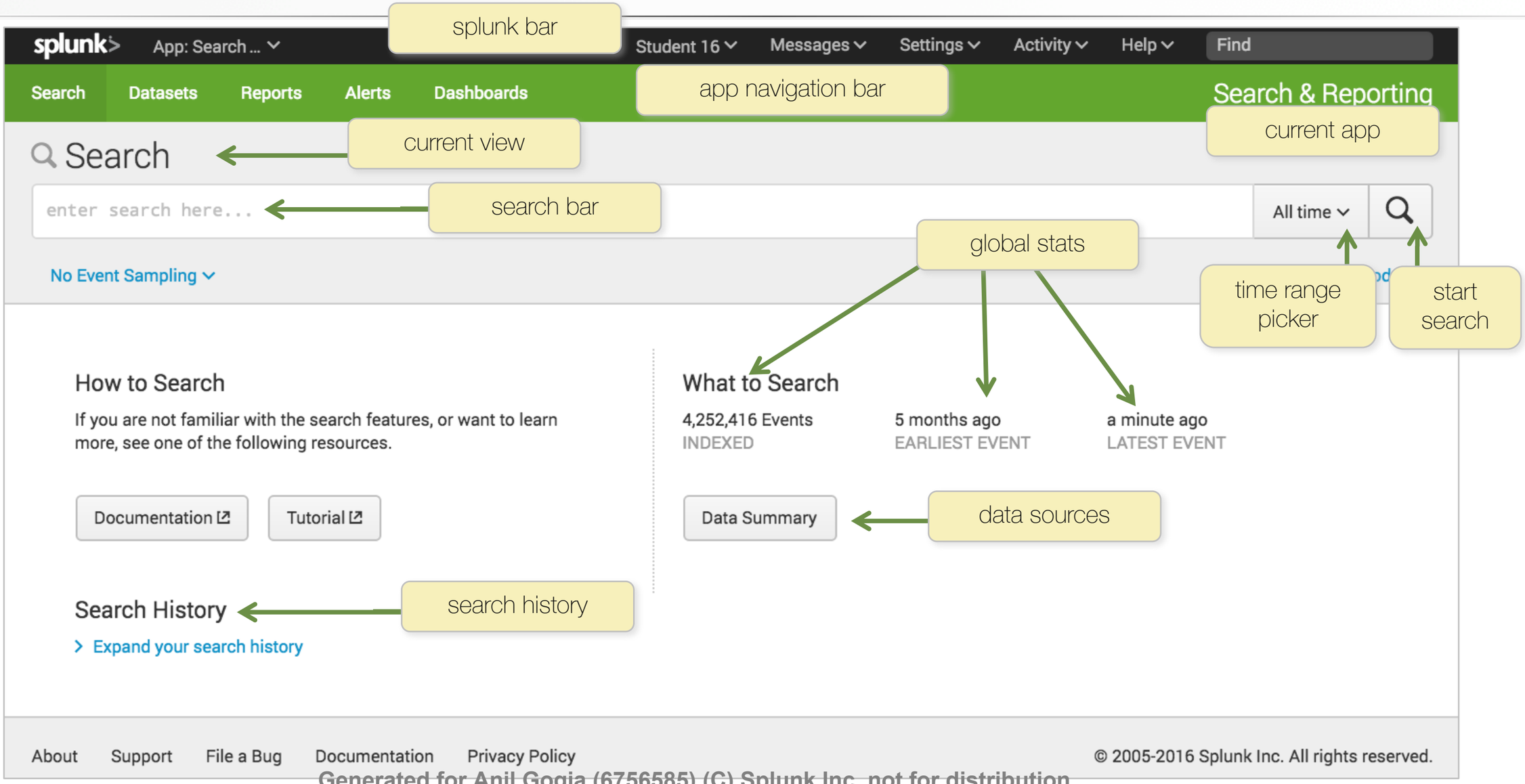


Module 5: Searching

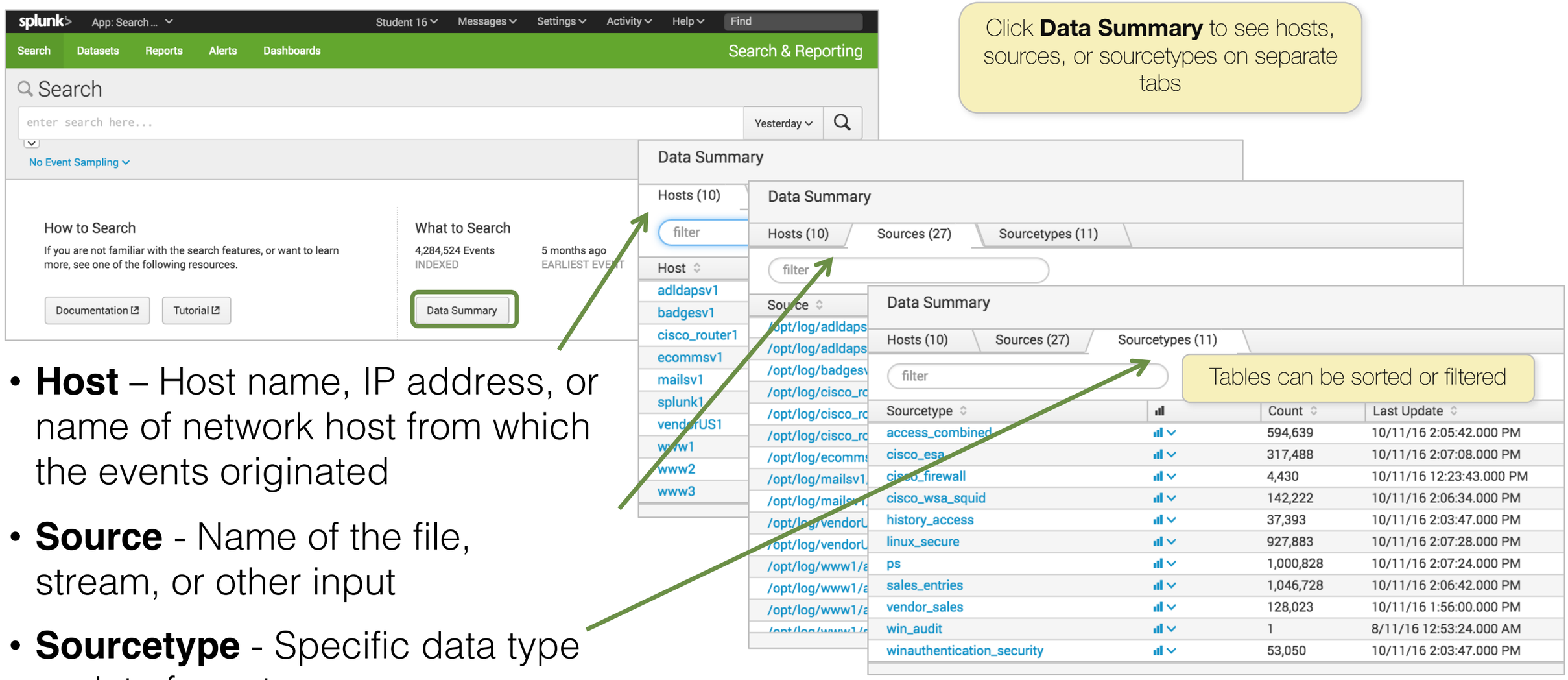
Module Objectives

- Run basic searches
- Use autocomplete to help build a search
- Set the time range of a search
- Identify the contents of search results
- Refine searches
- Use the timeline
- Control a search job
- Save search results

Search & Reporting App Overview (cont.)



Data Summary Tabs

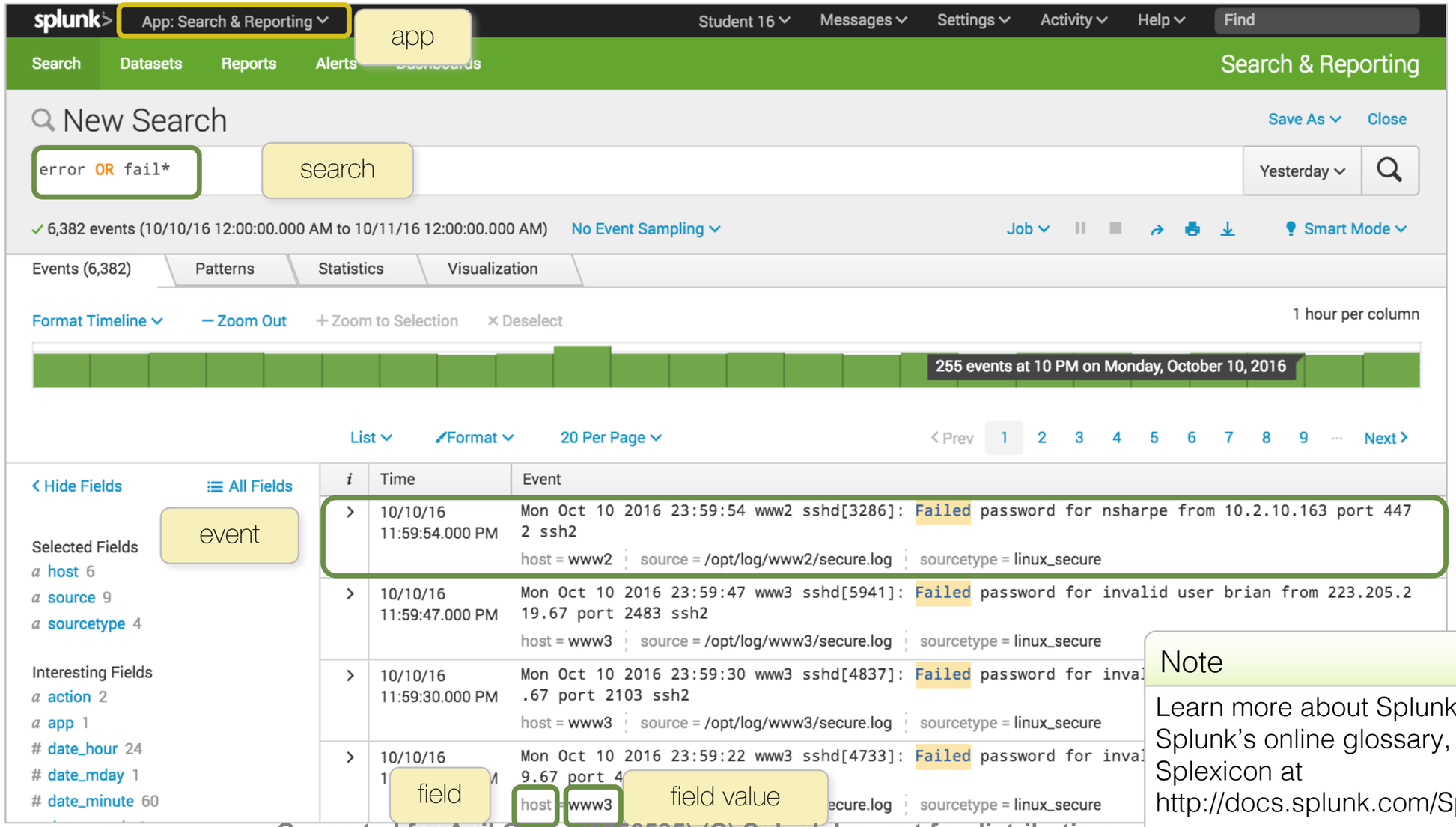


Click **Data Summary** to see hosts, sources, or sourcetypes on separate tabs

Tables can be sorted or filtered

- **Host** – Host name, IP address, or name of network host from which the events originated
- **Source** - Name of the file, stream, or other input
- **Sourcetype** - Specific data type or data format

Key Details



Note

Learn more about Splunk from Splunk's online glossary, the Splexicon at http://docs.splunk.com/Splexicon

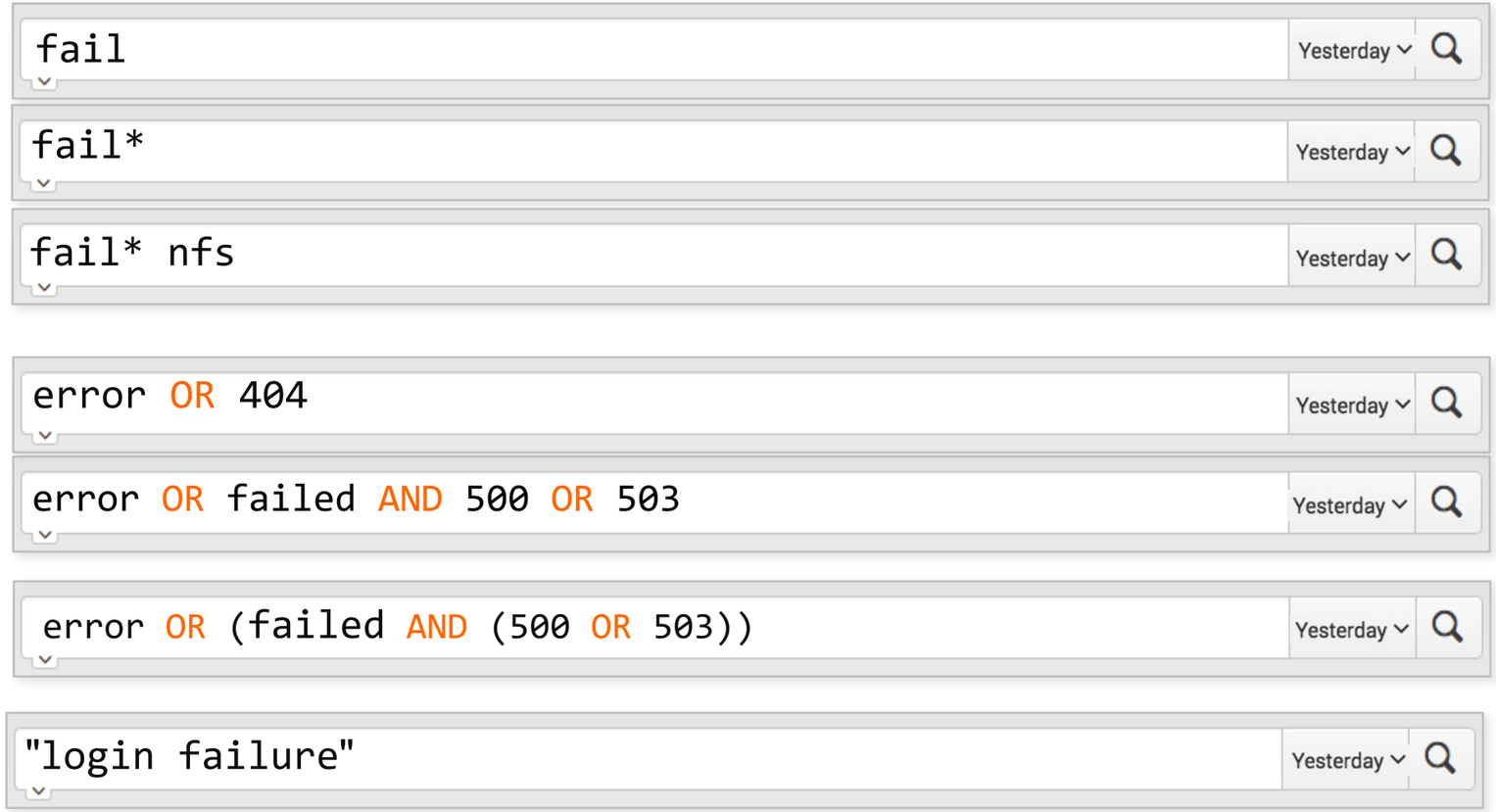
 | 5 May 2017

Why Learn to Search?

- Why is it important to be able to write searches?
  - You have questions about your data -- searches retrieve the events that can answer them
  - Every report and visualization is built based on an underlying search
  - Understanding, analyzing, and troubleshooting visualizations depends on your ability to understand the search string
  - Mastering the search language enables you to do as much as possible with your data to meet your specific needs

Search Guidelines

- * wildcard supported
- Search terms are case **in**sensitive
- Booleans AND, OR, NOT
  - Must be uppercase
  - AND is implied between terms
  - Use () for complex searches
- Quotation marks for phrases

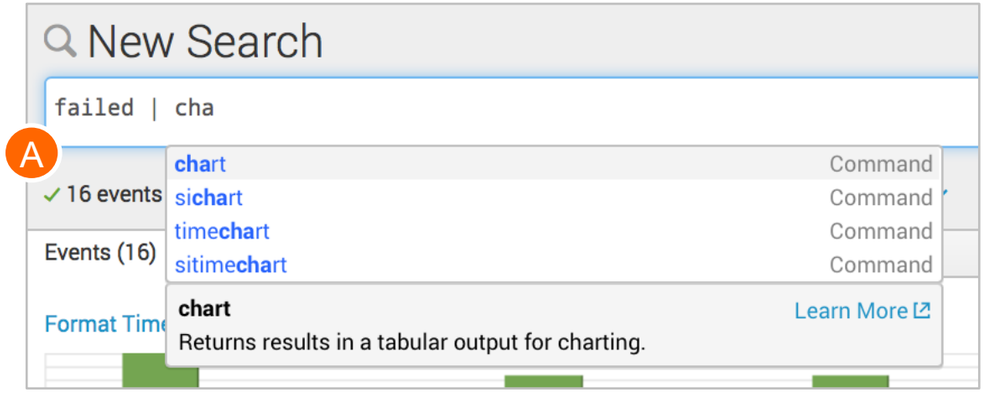


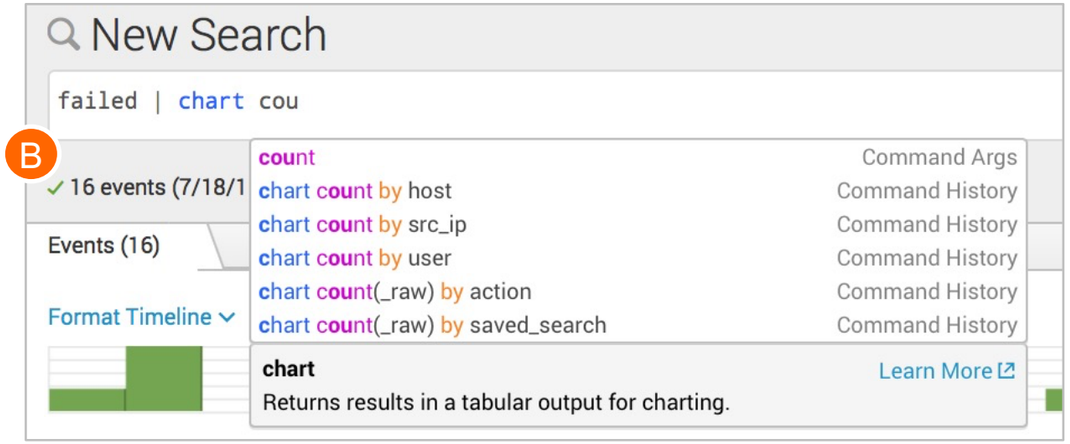
Search Assistant

- Search Assistant provides selections for how to complete the search string
- Before the first pipe (|), it will look for matching terms
- You can continue typing OR select a term from the list
  - If you select a term from the list, it is added to the search

Search Assistant (cont.)

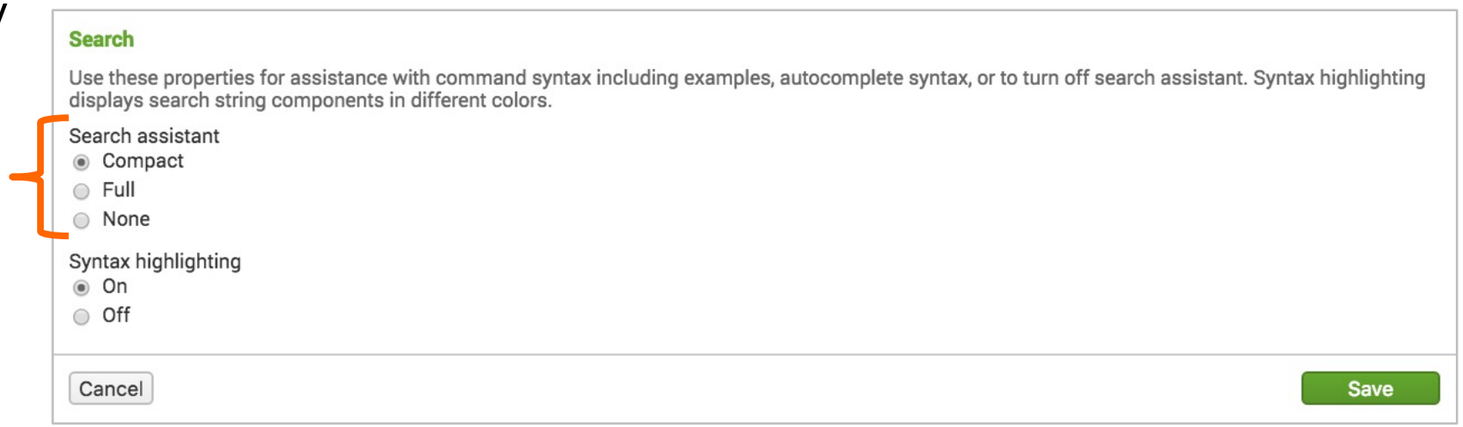
- After the first pipe, the Search Assistant will show a list of commands that can be entered into the search string
- You can continue typing OR scroll through and select a command to add
- If you mouse over a command, more information about the command is shown
- As you continue to type, Search Assistant makes more suggestions



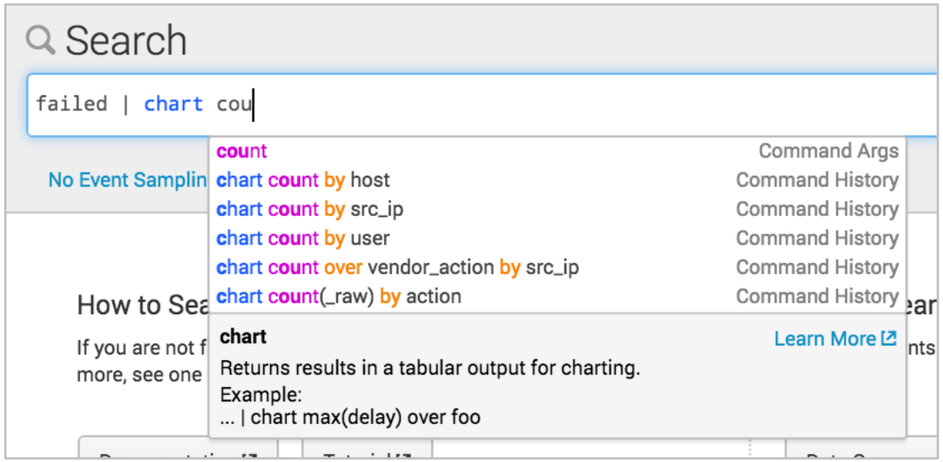


Search Assistant (cont.)

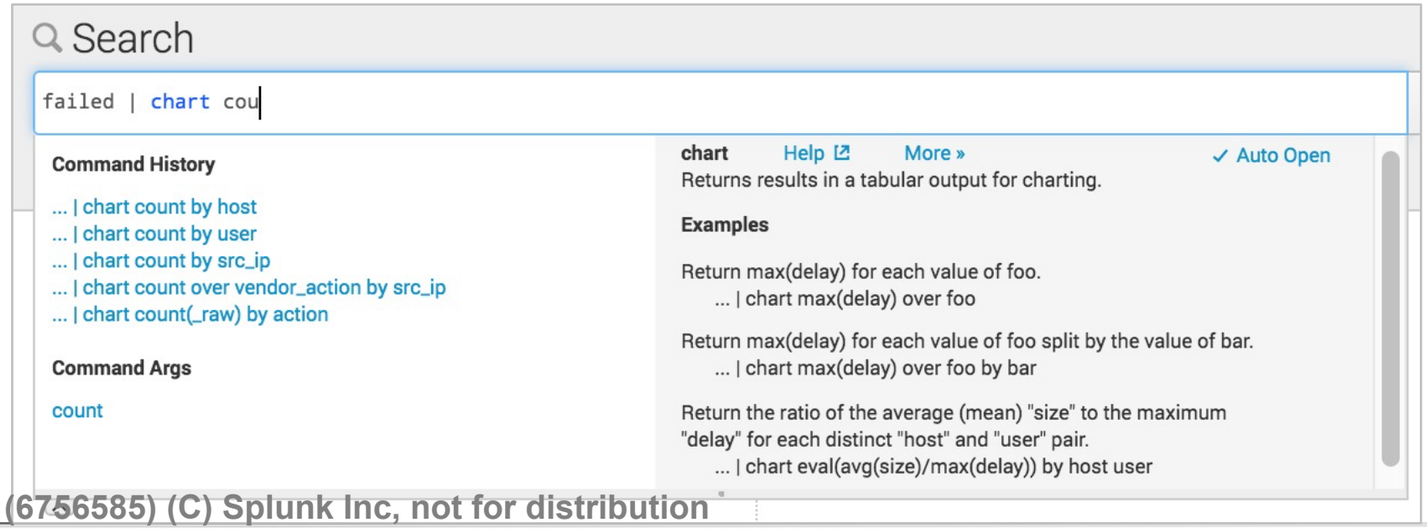
- Search Assistant is enabled by default, in the user settings
- By default, **Compact** is selected
- If desired, to show more information, choose **Full**



Compact

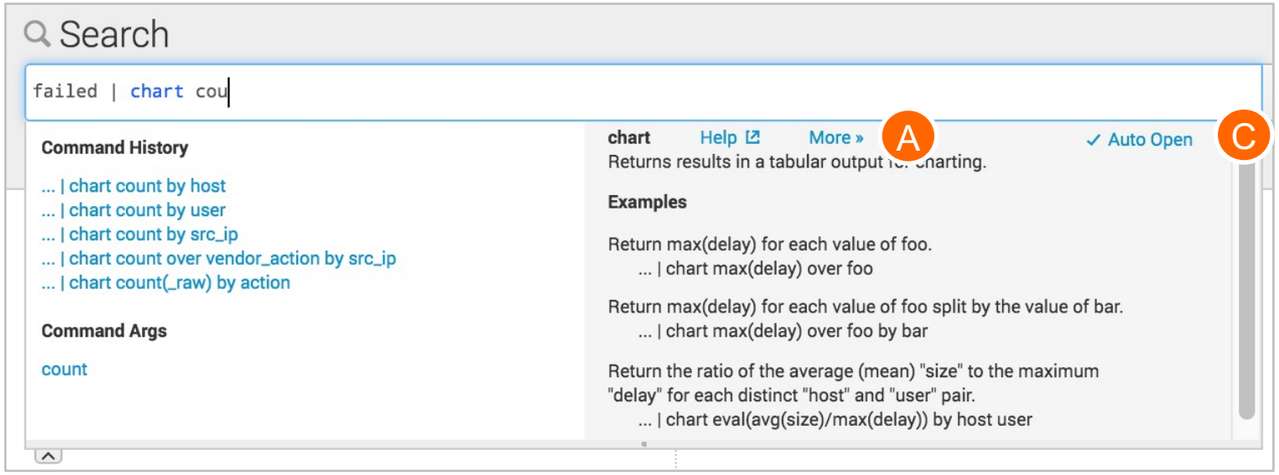


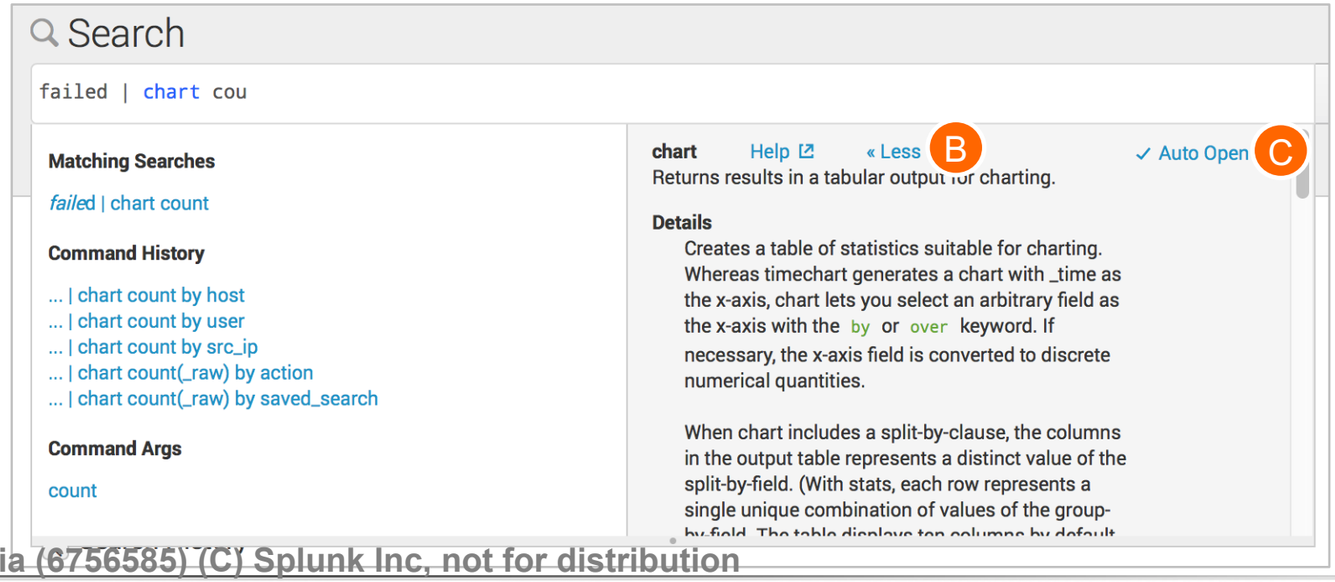
Full



Search Assistant - Full

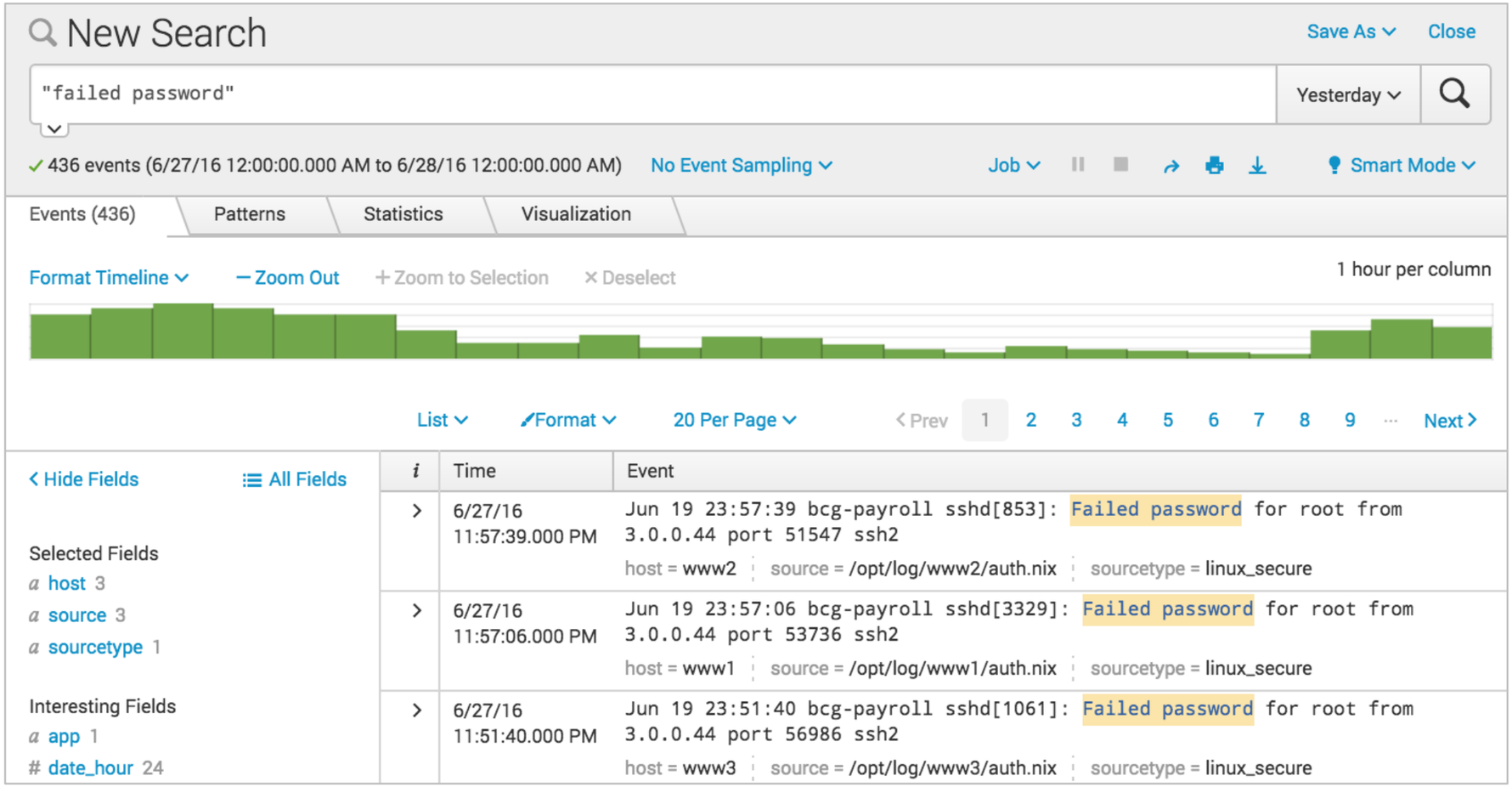
- A To show more information, click **More »**
- B To show less information, click **« Less**
- C To toggle Full mode off, de-select **Auto Open**





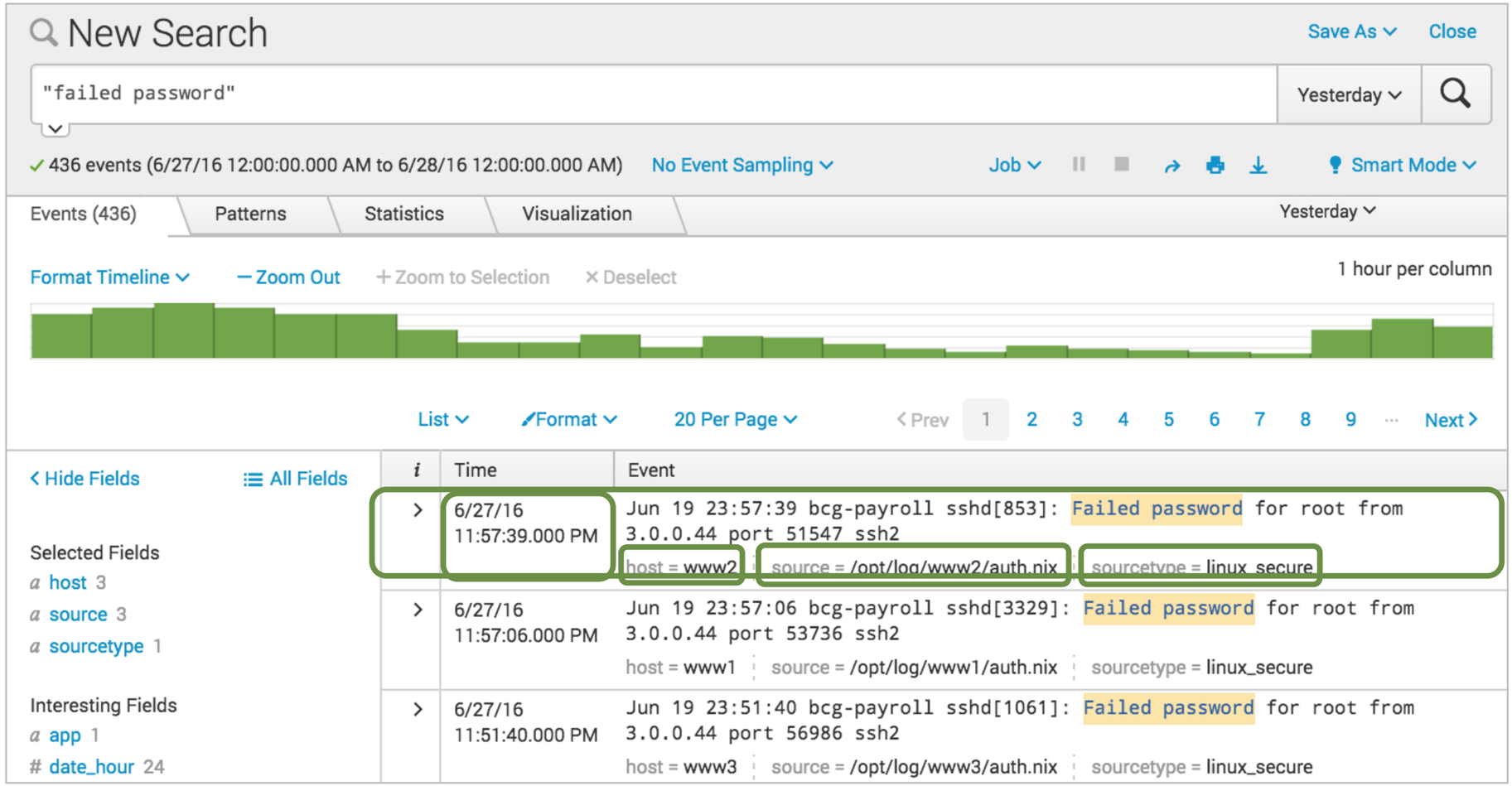
Search Results

- Matching results are returned immediately
- Displayed in reverse chronological order (newest first)
- Matching search terms are highlighted

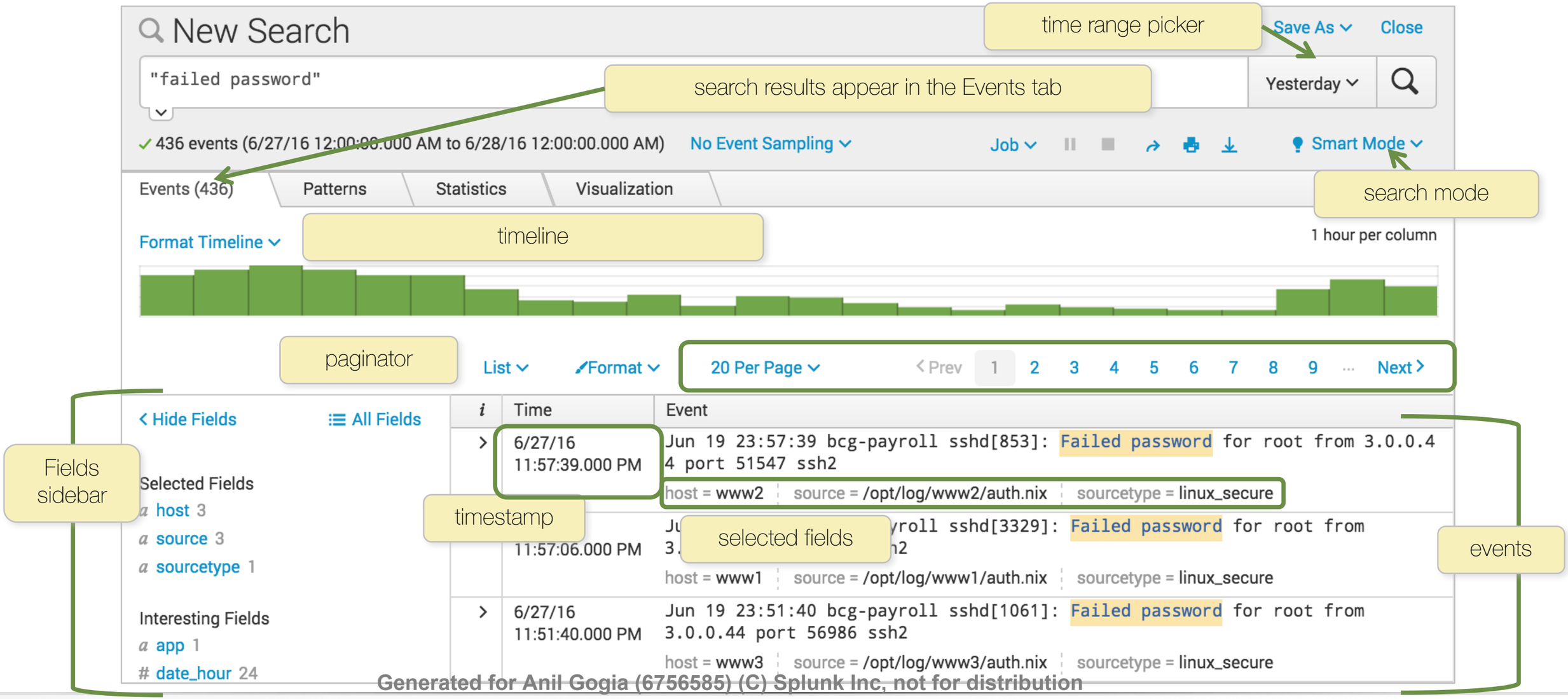


Event Details

- Splunk parses data into individual events
- Each event has a:
  - timestamp
  - host
  - source
  - sourcetype

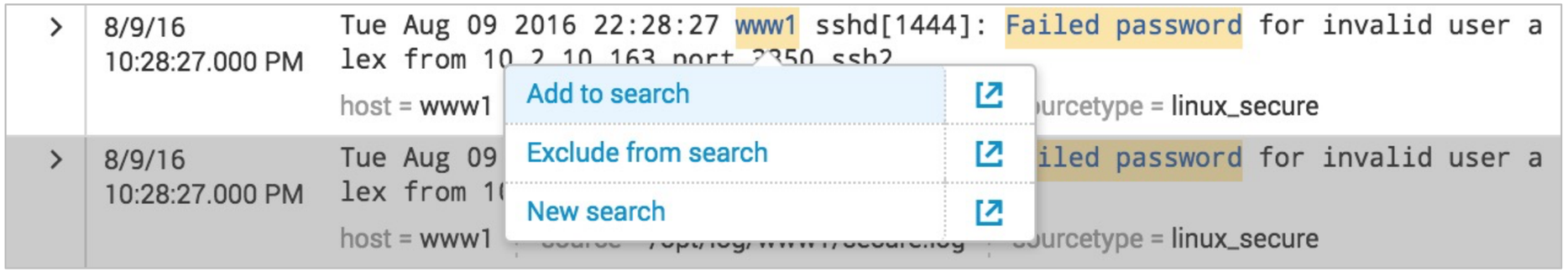


Search Results Details



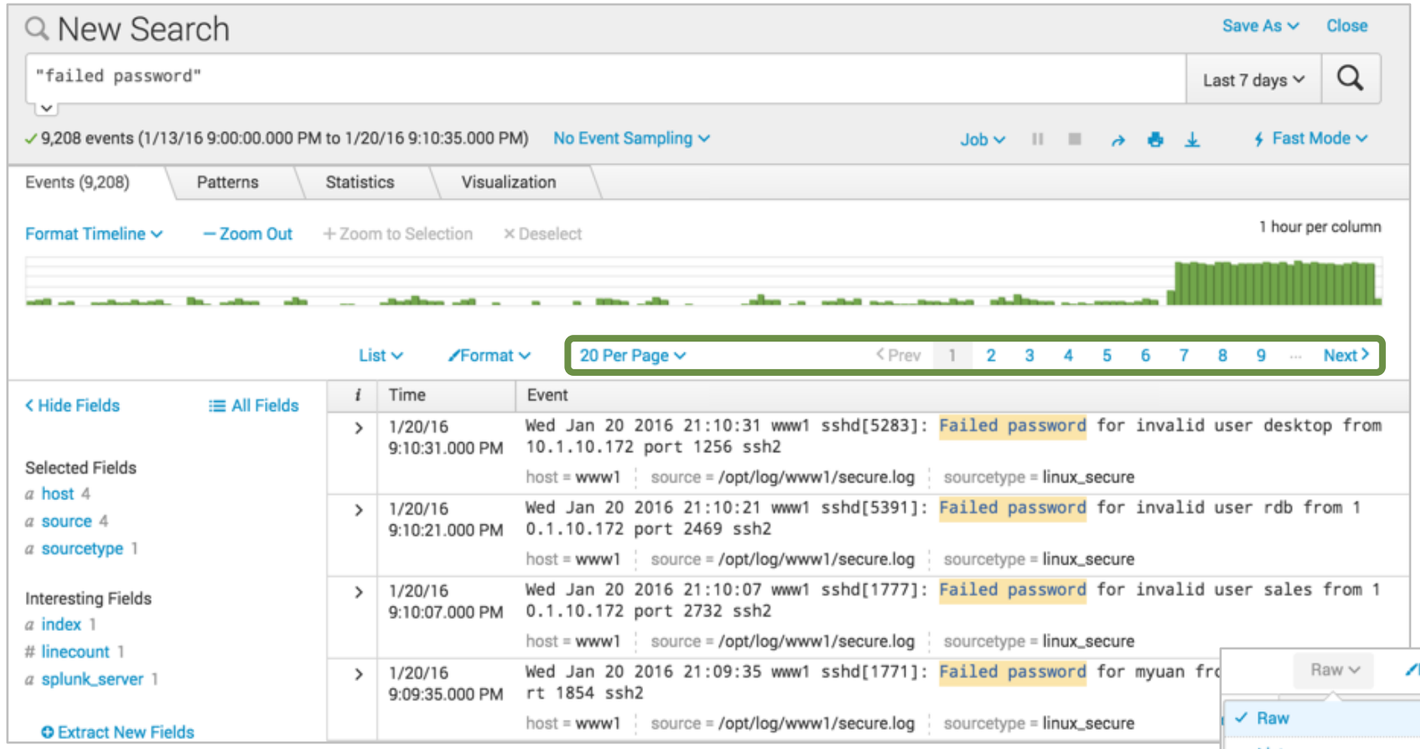
Using Search Results to Modify a Search

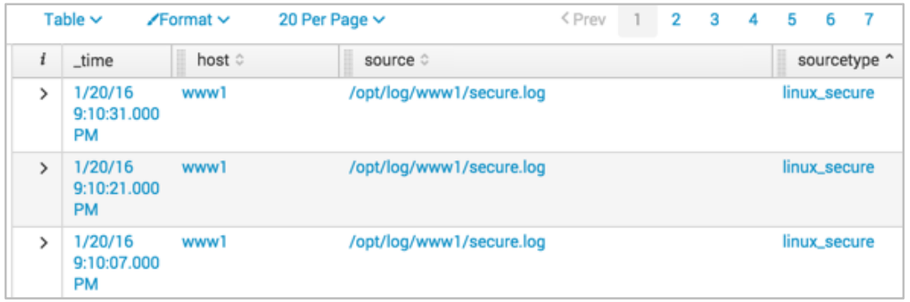
- When you mouse over search results, keywords are highlighted
- Click any item in your search results; a window appears allowing you to:
  - Add the item to the search
  - Exclude the item from the search
  - Open a new search including only that item

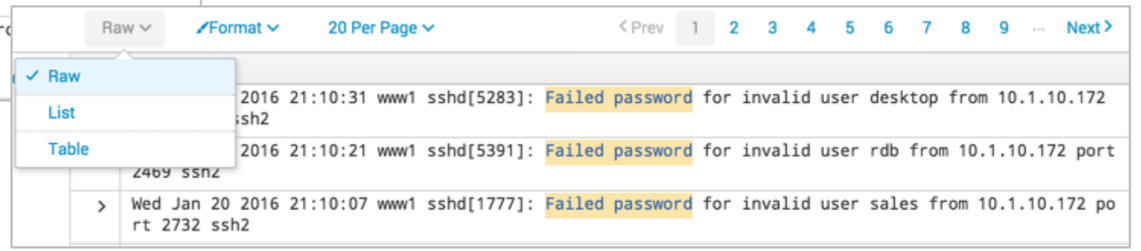


Changing Search Results View Options

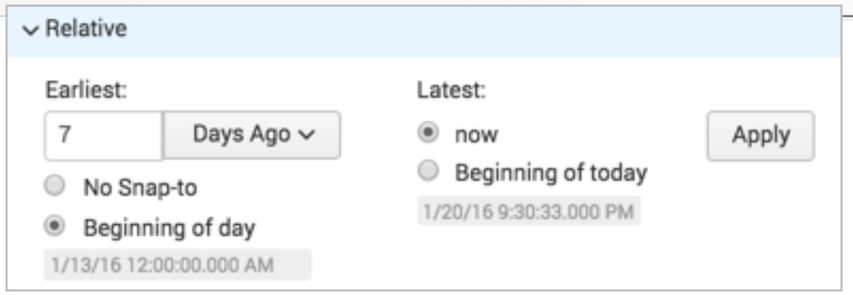
You have several layout options for displaying your search results

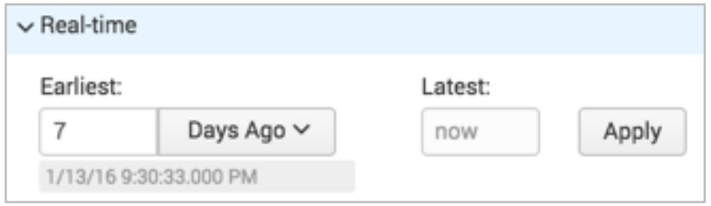


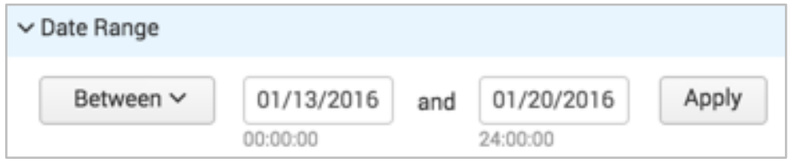


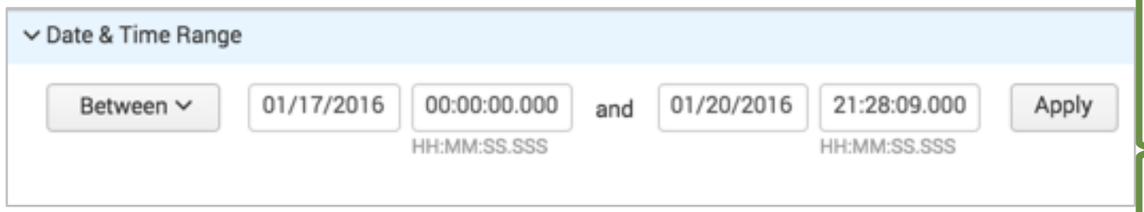


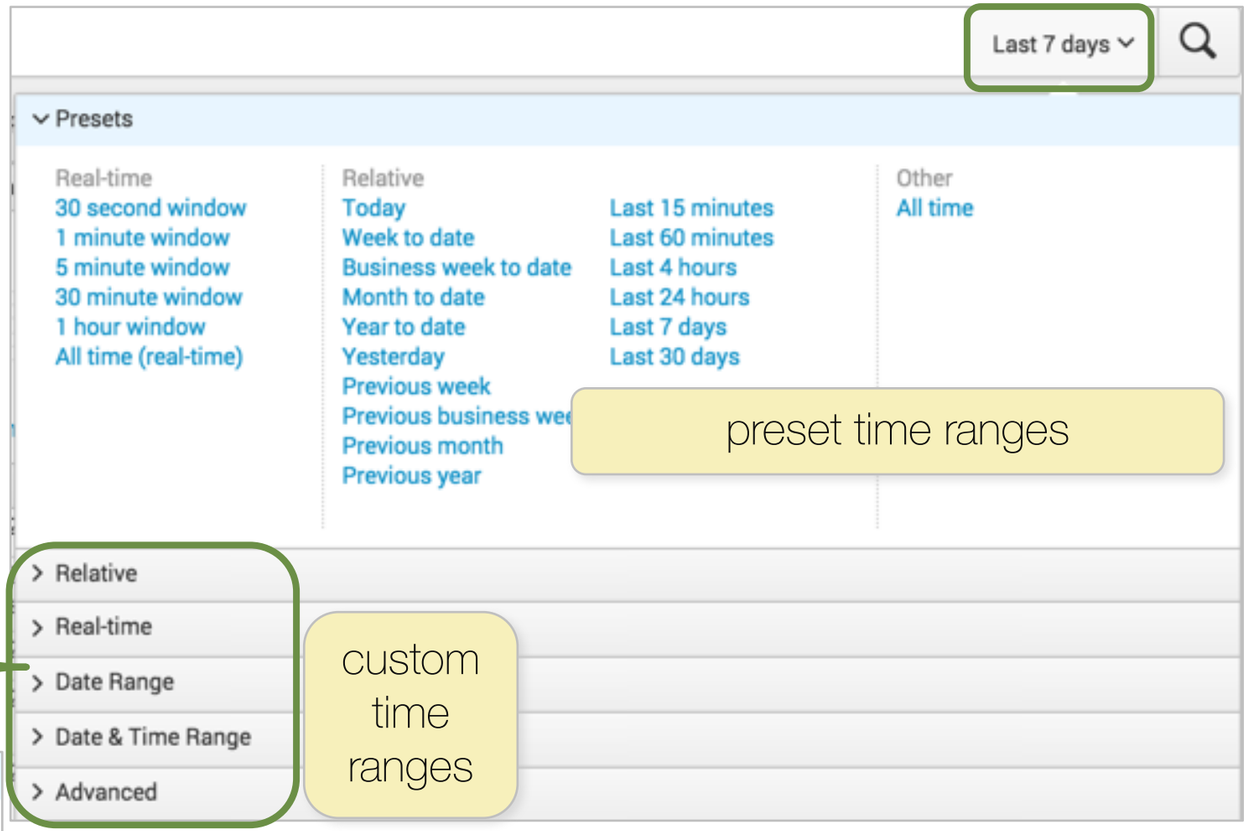
Selecting a Specific Time





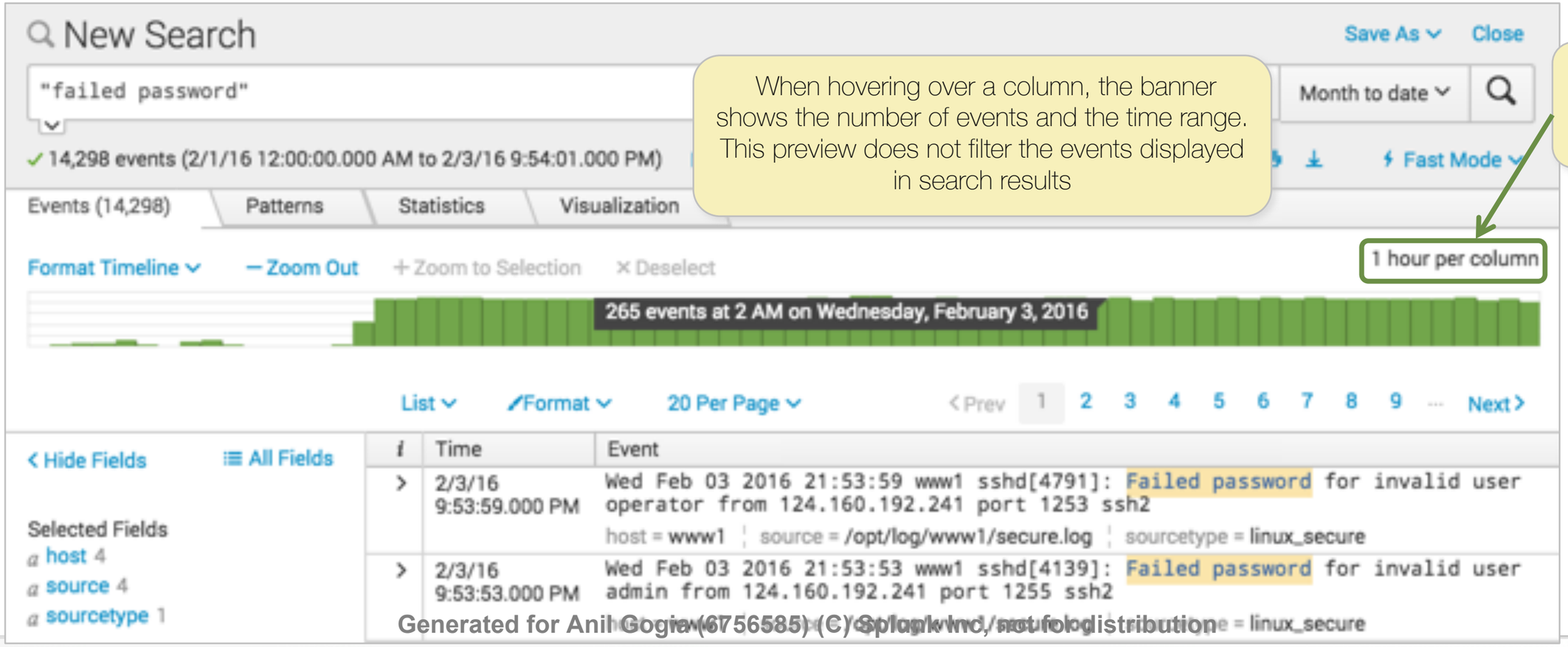






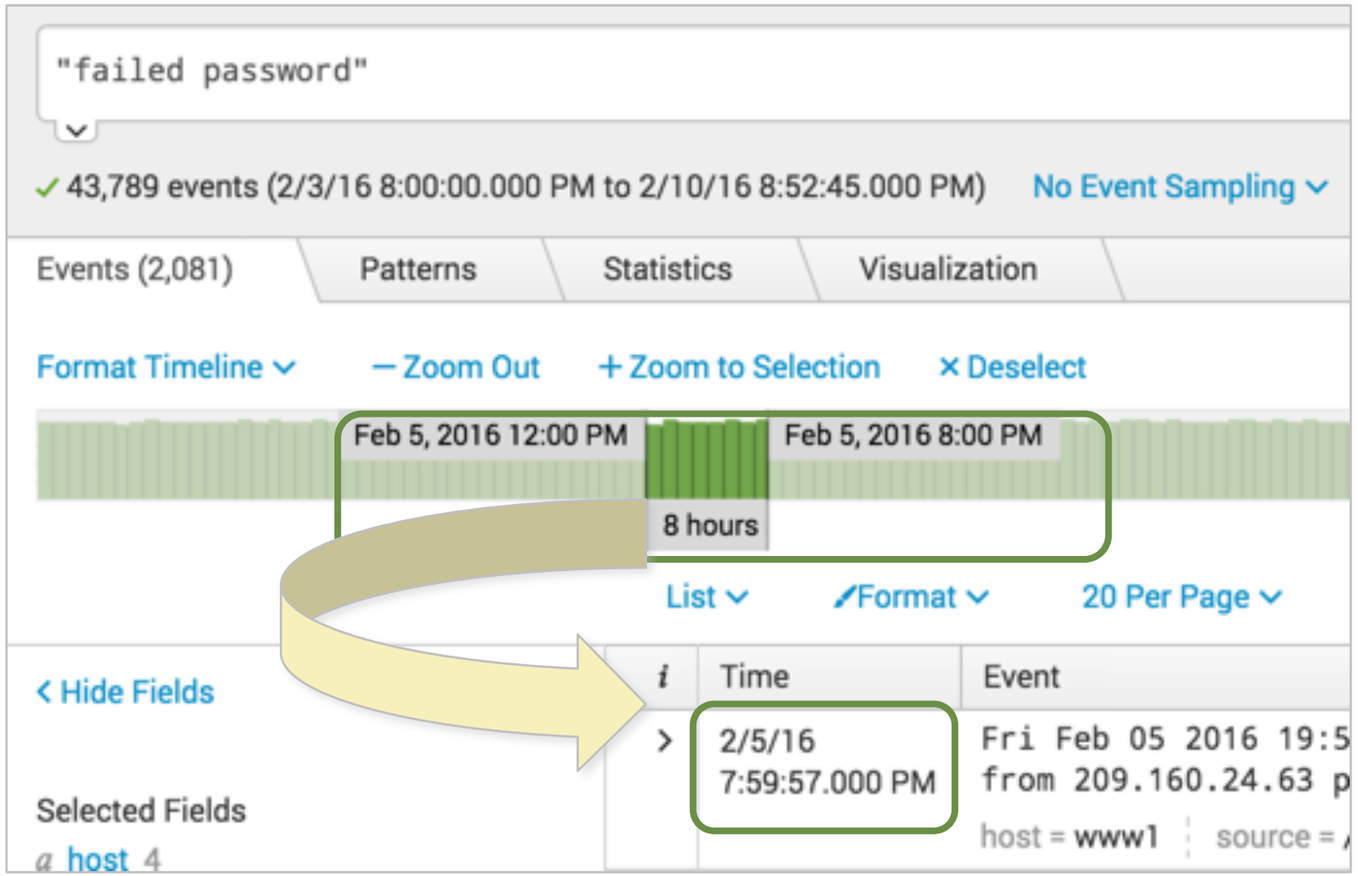
Viewing the Timeline

- Timeline shows distribution of events specified in the time range
  - Mouse over for details, or single-click to filter results for that time period



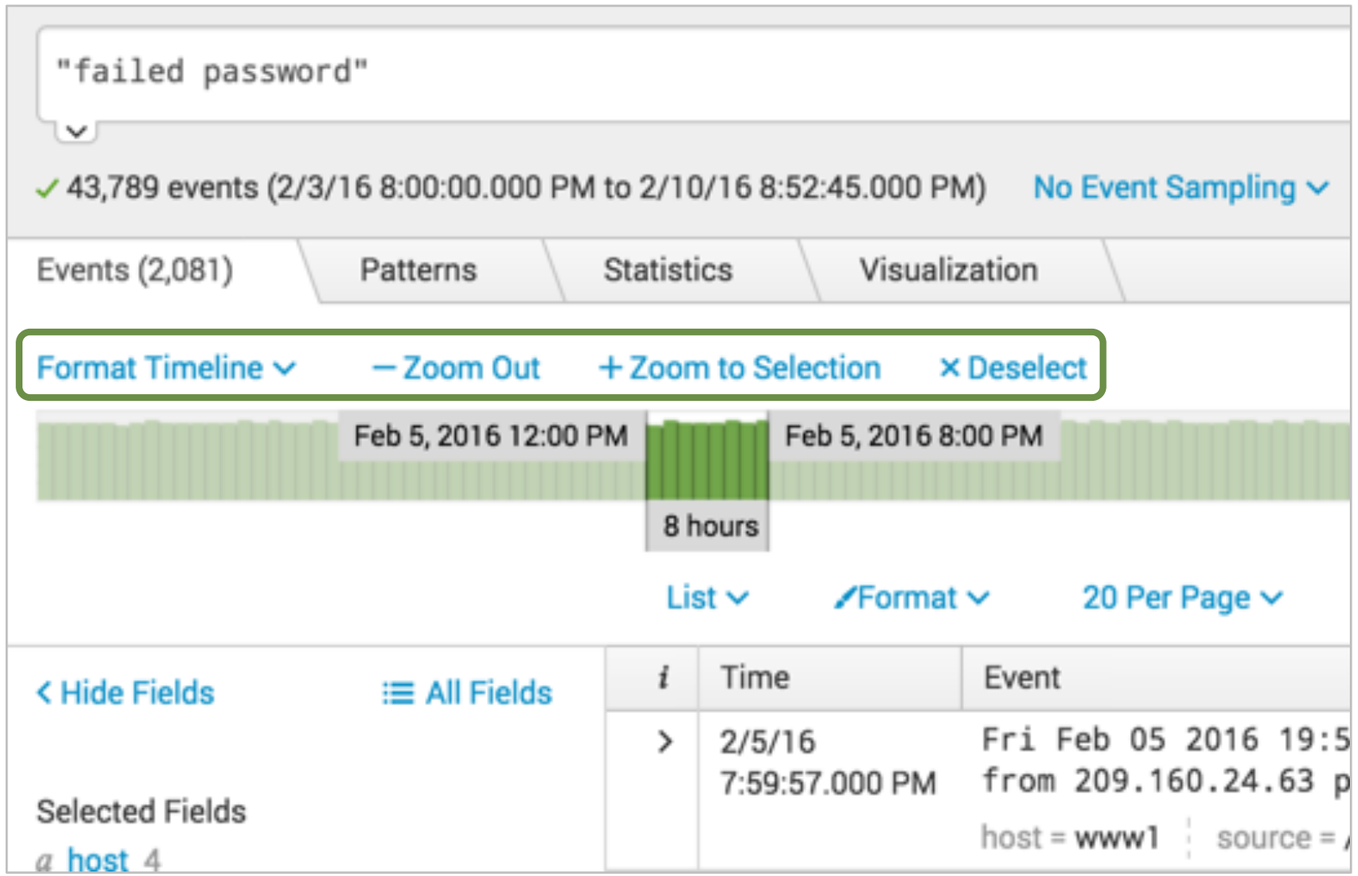
View a Subset of the Results with Timeline

- To select a narrower time range, click and drag across a series of bars
  - This action filters the current search results
    - Does not re-execute the search
  - This filters the events and displays them in reverse chronological order (most recent first)



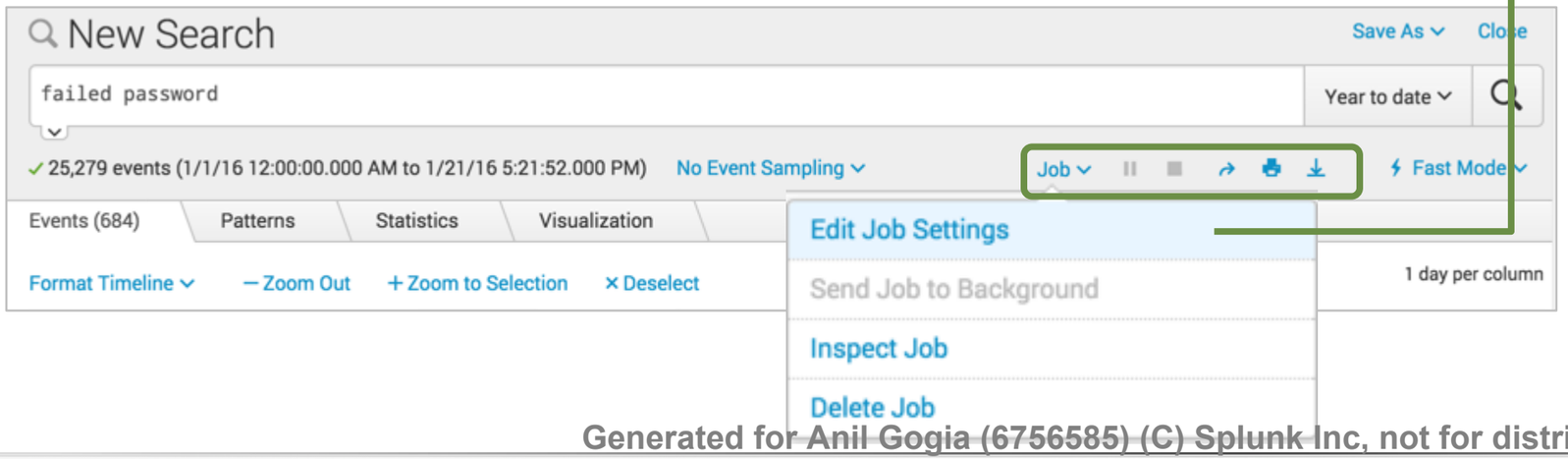
Use Other Timeline Controls

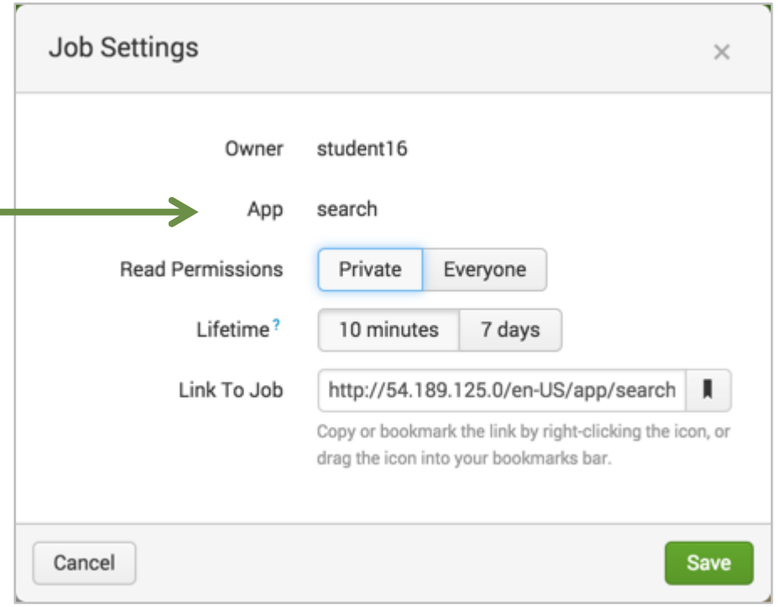
- **Format Timeline**
  - Hides or shows the timeline in different views
- **Zoom Out**
  - Expands the time focus and re-executes the search
- **Zoom to Selection**
  - Narrows the time range and re-executes the search
- **Deselect**
  - If in a drilldown, returns to the original results set
  - Otherwise, grayed out / unavailable



Control or Save Search Jobs

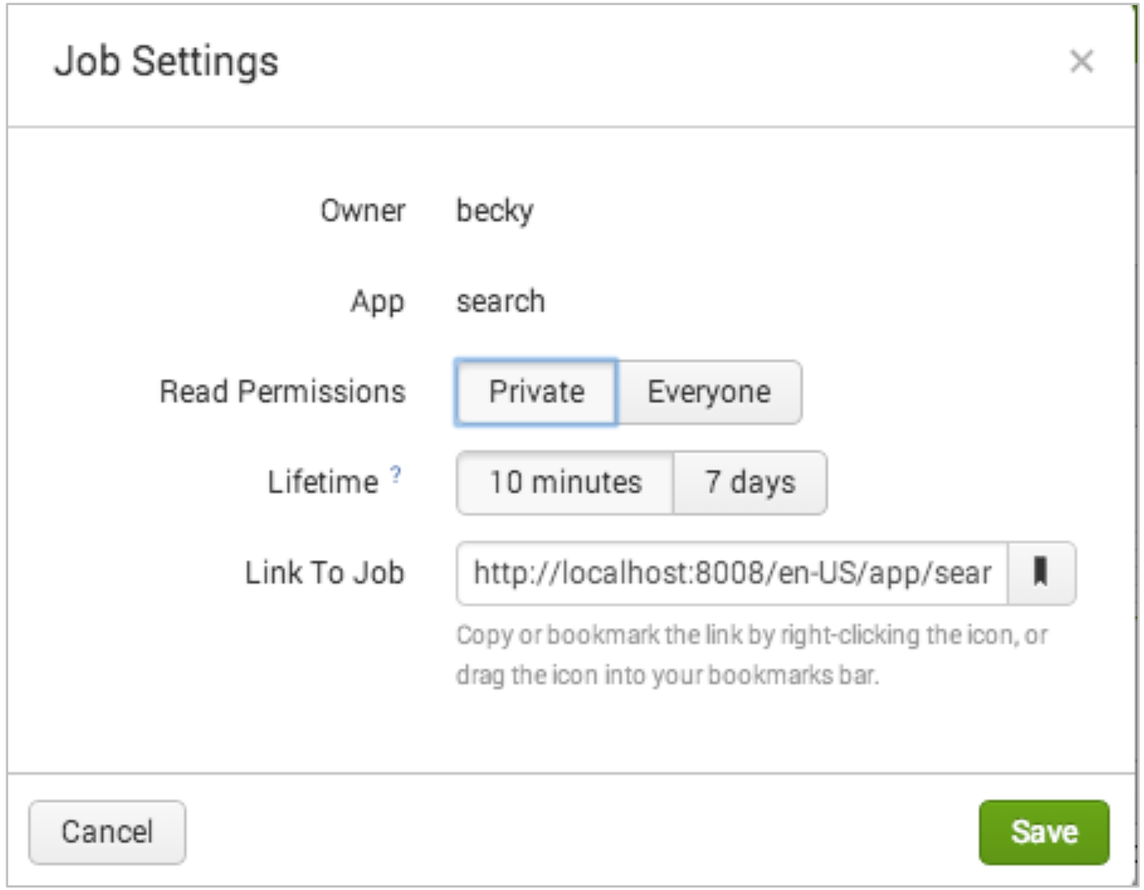
- Every search is also a **job**
- Use the Job bar to control search execution
  - **Pause** – toggles to resume the search
  - **Stop** – finalizes the search in progress
    - Jobs are available for 10 minutes (default)
    - Get a link to results from the **Job** menu





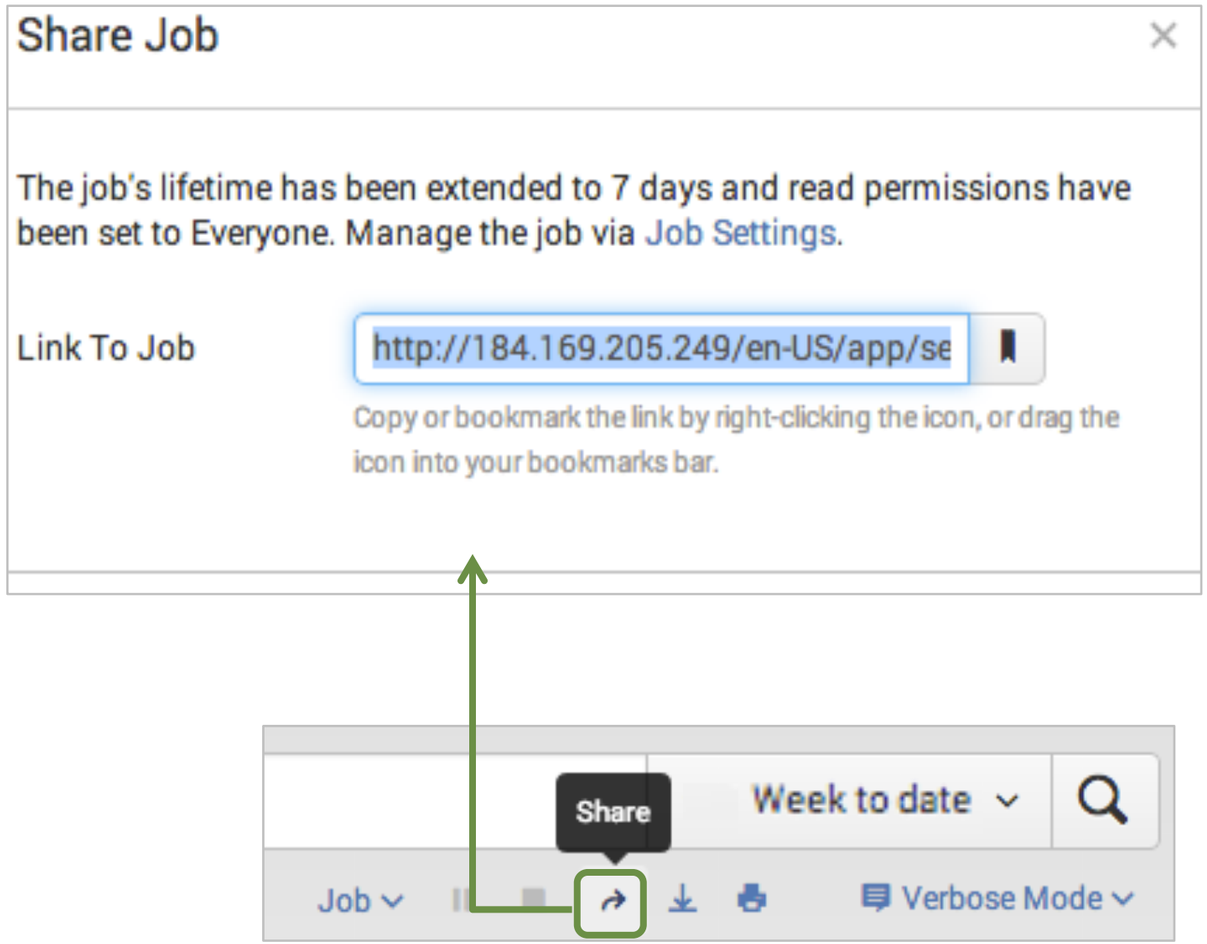
Set Permissions

- **Private** [default]
  - Only the creator can access
- **Everyone**
  - All app users can access search results
- **Lifetime**
  - Default is 10 minutes
  - Can be extended to 7 days
  - To keep your search results longer, schedule a report



Share Search Jobs

- Use the Share button next to the Job bar to quickly:
  - Apply read permissions to everyone
  - Extend the retention of the results to 7 days
  - Get a sharable link to the results
- Click the printer icon to print results or save them as PDF

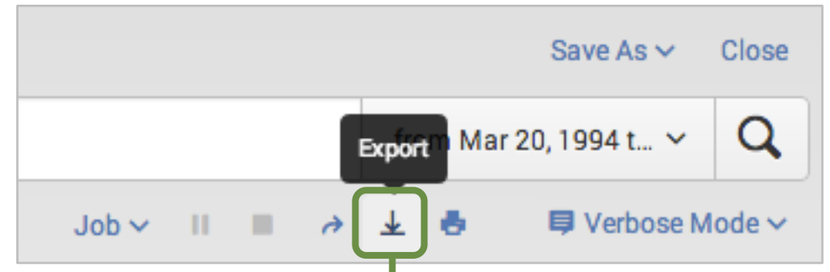


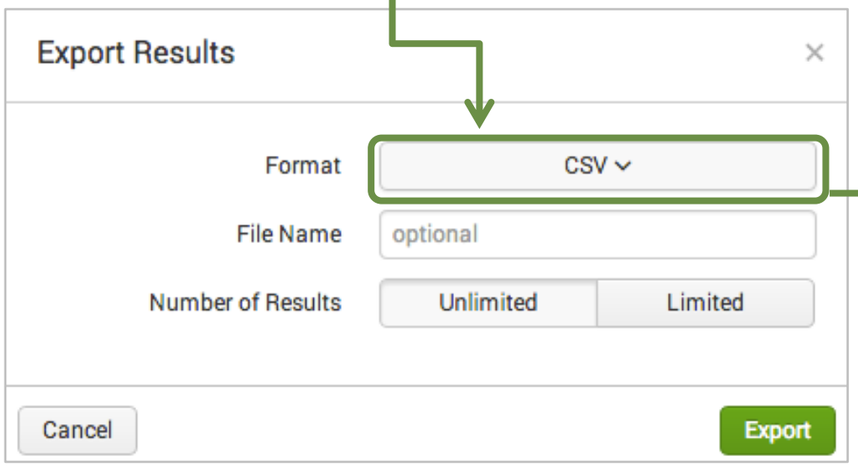
Note

Saving and sharing the actual search strings for reuse (known as **Reports**) is discussed later in this course.

Export Search Results

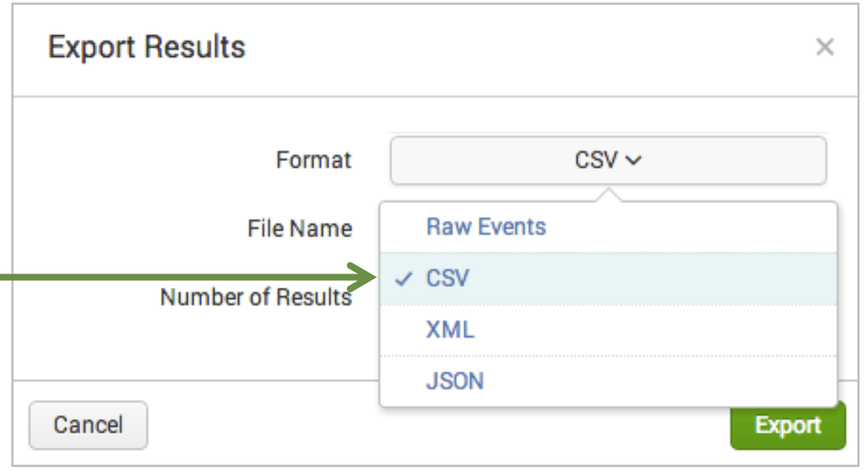
For an external copy of the results, **export** search results to Raw Events (text file), CSV, XML, or JSON format





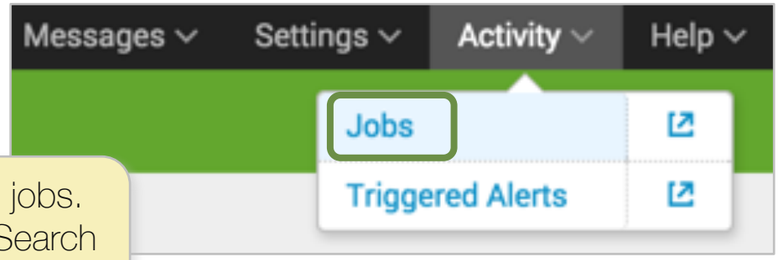
Note

Note that exporting the results of a large search is very memory-intensive!

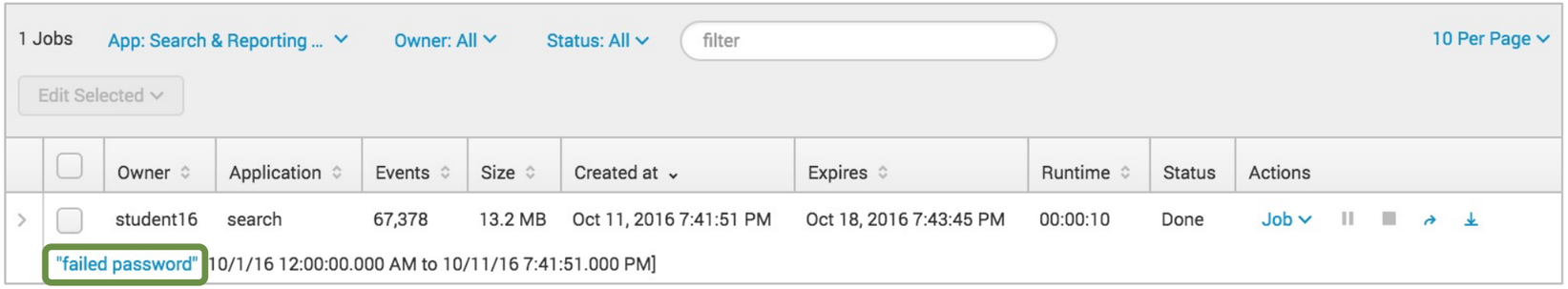


Review Your Job History

- Access saved search jobs from the **Activity** menu
- The Search Jobs view displays jobs that:
  - You have run in the last 10 minutes
  - You have extended for 7 days
- Click on a job link to view the results in the designated app view

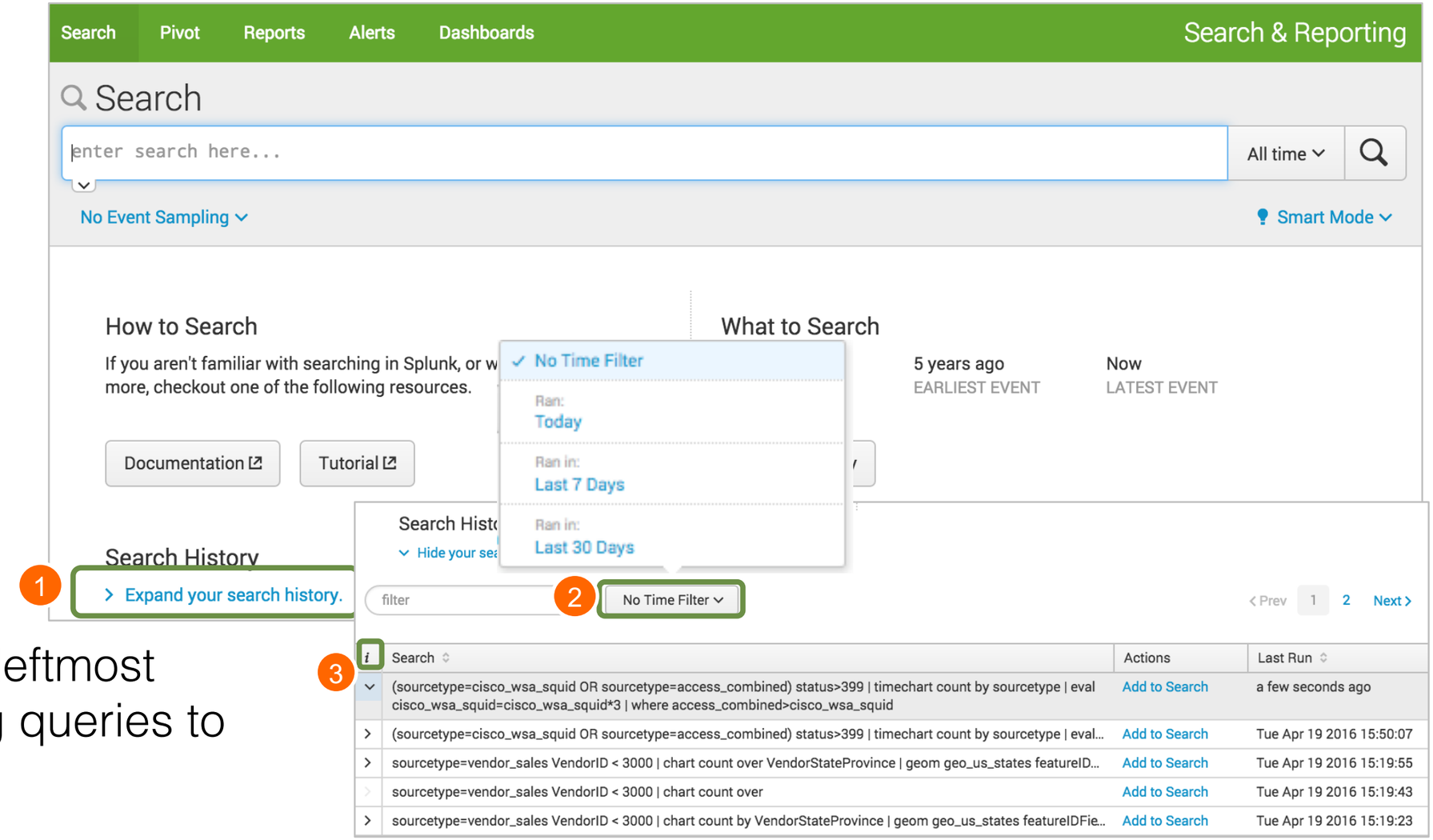


Click **Activity > Jobs** to view your saved jobs. Click the job's name to examine results in Search view. (The job name is the search string.)



Review Your Search History

1. Search History displays your most recent ad-hoc searches – 5 per page
2. You can set a time filter to further narrow your results
3. Click the > icon in the leftmost column to expand long queries to display the full text



Module 6:
Using Fields in Searches

Module Objectives

- Understand fields
- Use fields in searches
- Use the fields sidebar
- Use search modes (fast, verbose, and smart)

What Are Fields?

- Fields are searchable key/value pairs in your event data
  - Examples: **`host=www1 status=503`**
- Fields can be searched with their names, like separating an http status code of 404 from Atlanta's area code (**`area_code=404`**)
- Between search terms, unless otherwise specified, **`AND`** is implied

Field Discovery

- Splunk discovers all fields based on sourcetype and any key/value pairs found in the data
- Already stored with the event in the index (prior to search time) are:
  - Meta fields, such as **host**, **source**, and **sourcetype**
  - Meta fields, including internal fields like **_time**, **_raw**
    - Splunk may extract other fields from the raw event data that may not be directly related to your search
- **Field discovery** is directly related to each search's results
  - Some fields in the overall data may not appear within the results of a particular search

Note

While Splunk auto-extracts many fields, you can learn how to create your own in the *Splunk Fundamentals 2* course.

Identify Data-Specific Fields

- Data-specific fields come from the specific characteristics of your data
  - Sometimes, this is indicated by obvious key = value pairs (**`action = purchase`**)
  - Sometimes, this comes from data within the event, defined by the sourcetype (**`status = 200`**)

i	Time	Event
>	12/3/15 7:12:32.000 PM	207.36.232.245 - - [03/Dec/2015:19:12:32] "POST /cart/success.do?JSESSIONID=SD1SL3FF 8ADFF4966 HTTP 1.1" 200 1443 "http://www.buttercupgames.com/cart.do?action=purchase& itemId=EST-26" "Mozilla/5.0 (Windows; U; Windows NT 5.1; en-GB; rv:1.8.1.6) Gecko/20 070725 Firefox/2.0.0.6" 963

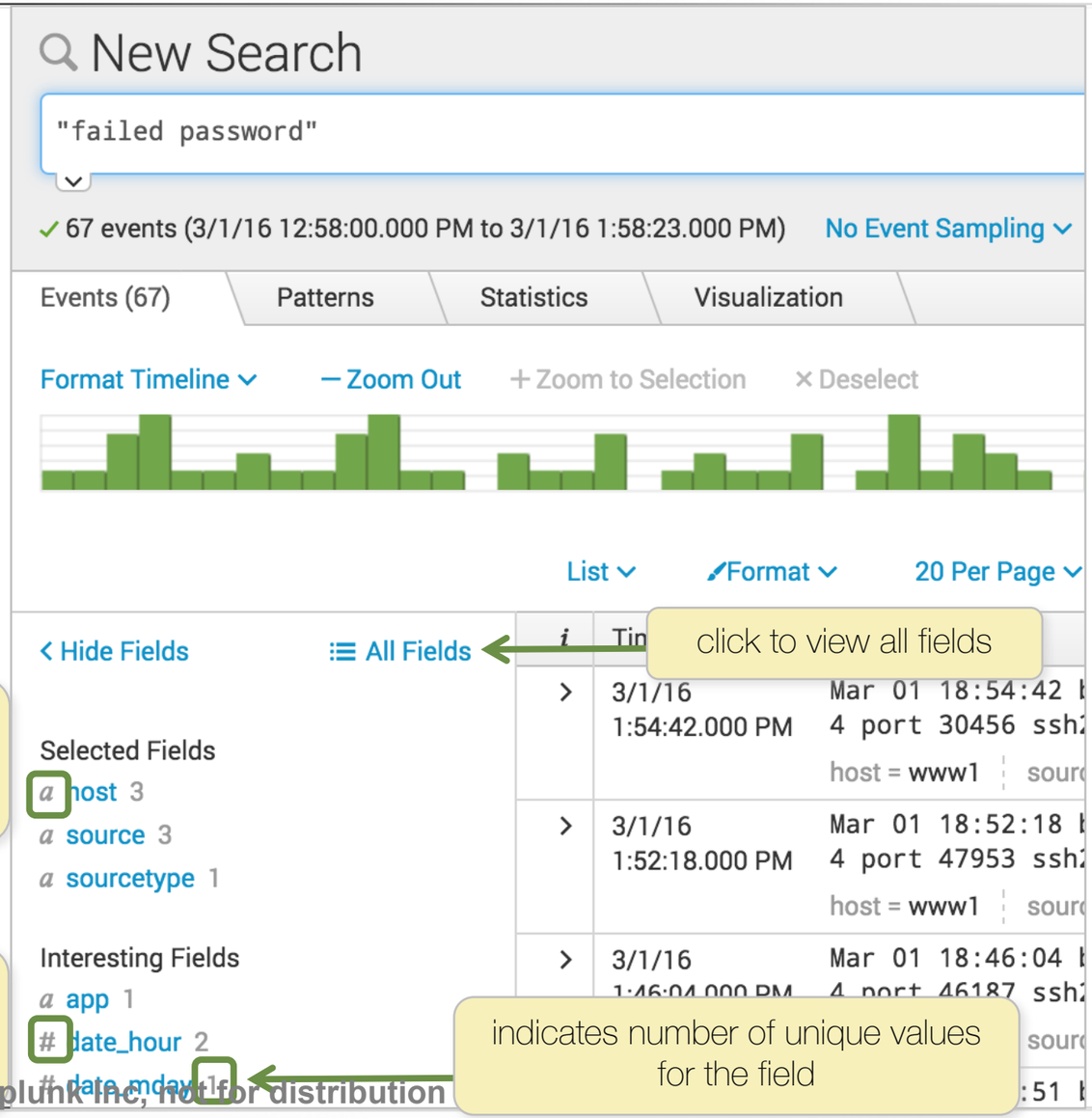
Note

For more information, please see:
http://docs.splunk.com/Documentation/Splunk/latest/Data/Listofpretrainedsourcetypes

Fields Sidebar

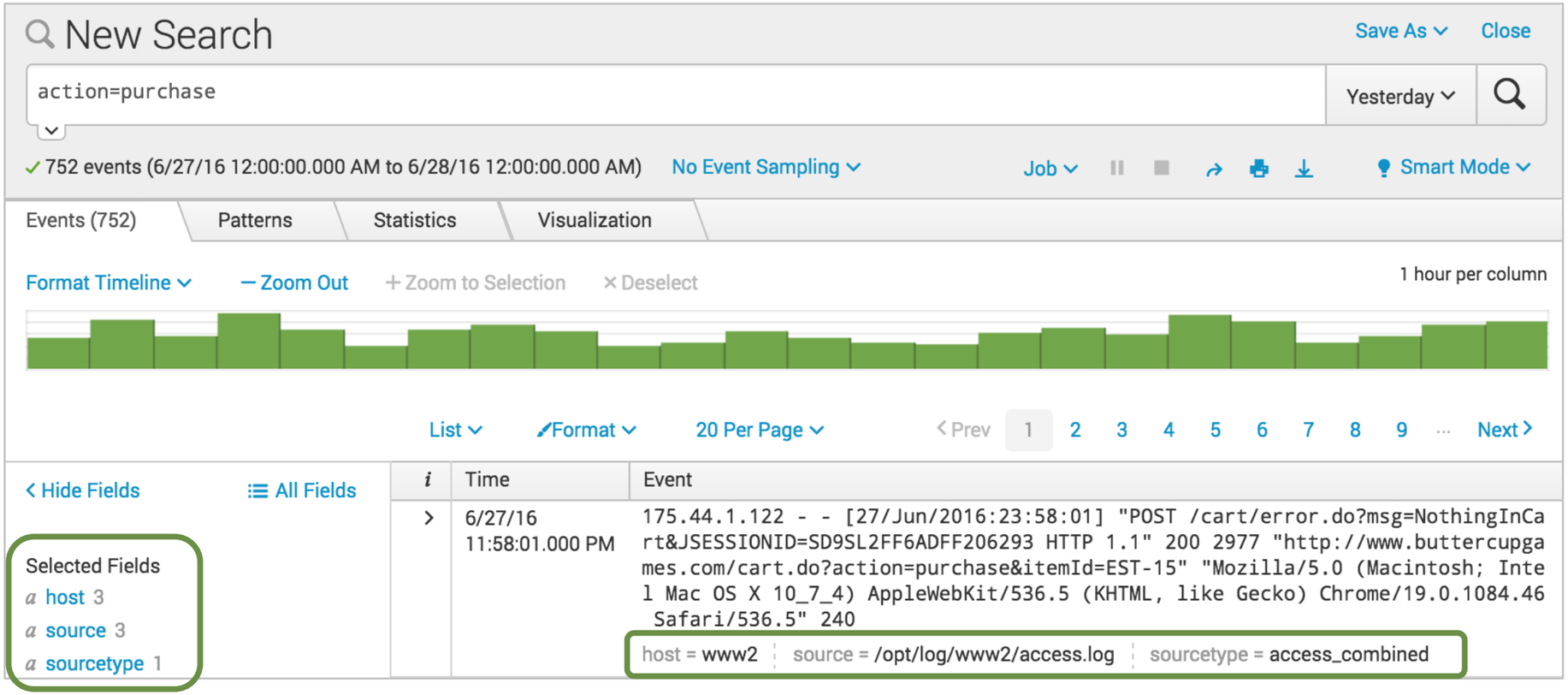
For the current search:

- **Selected Fields** – a set of configurable fields displayed for each event
- **Interesting Fields** – occur in at least 20% of resulting events
- **All Fields** link to view all fields (including non-interesting fields)



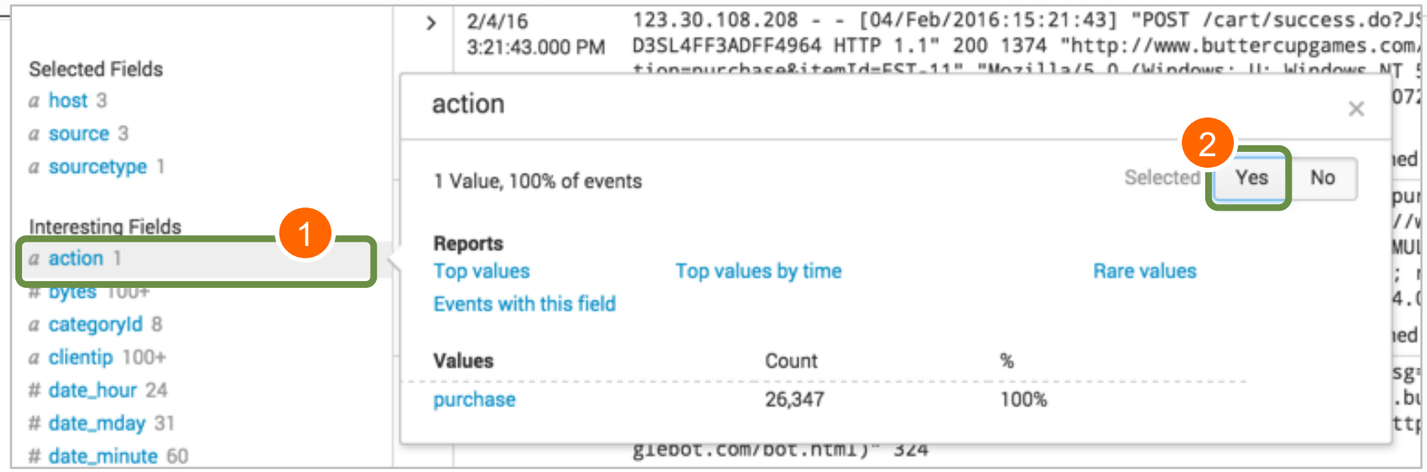
Describe Selected Fields

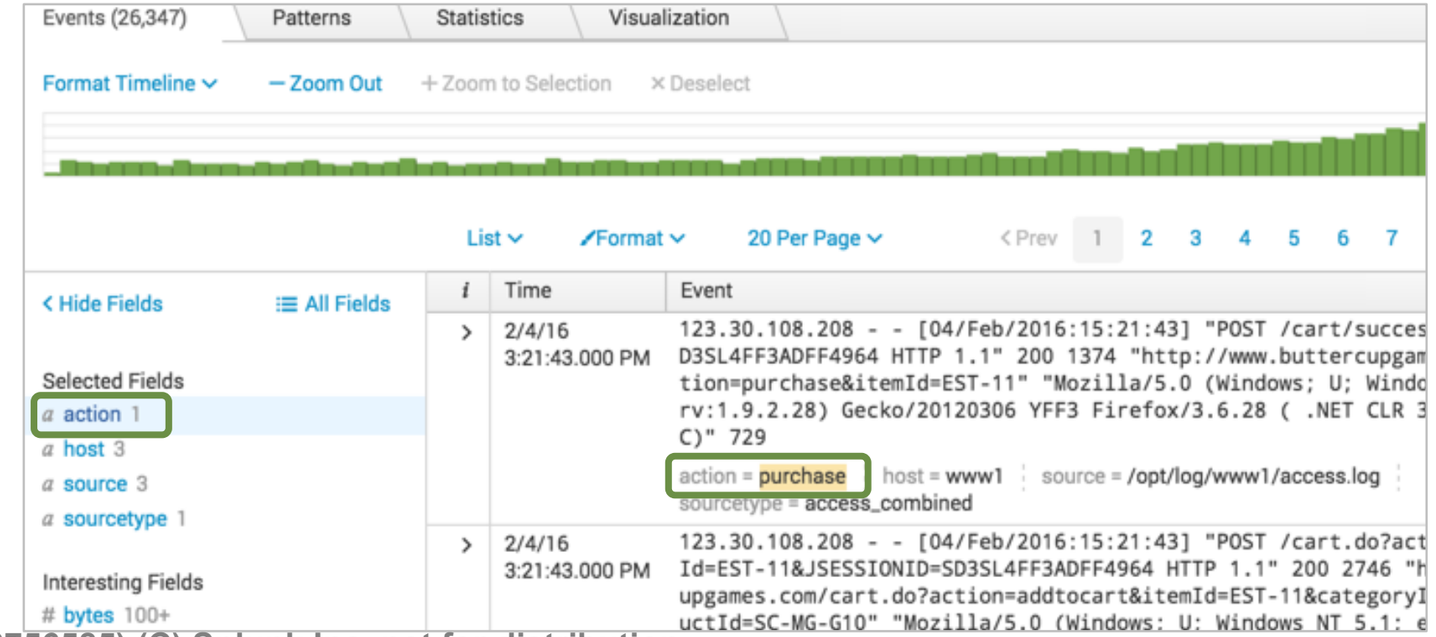
- Selected fields and their values are listed under every event that includes those fields
- By default, the selected fields are:
  - host
  - source
  - sourcetype
- You can choose any field and make it a selected field



Make an Interesting Field a Selected Field

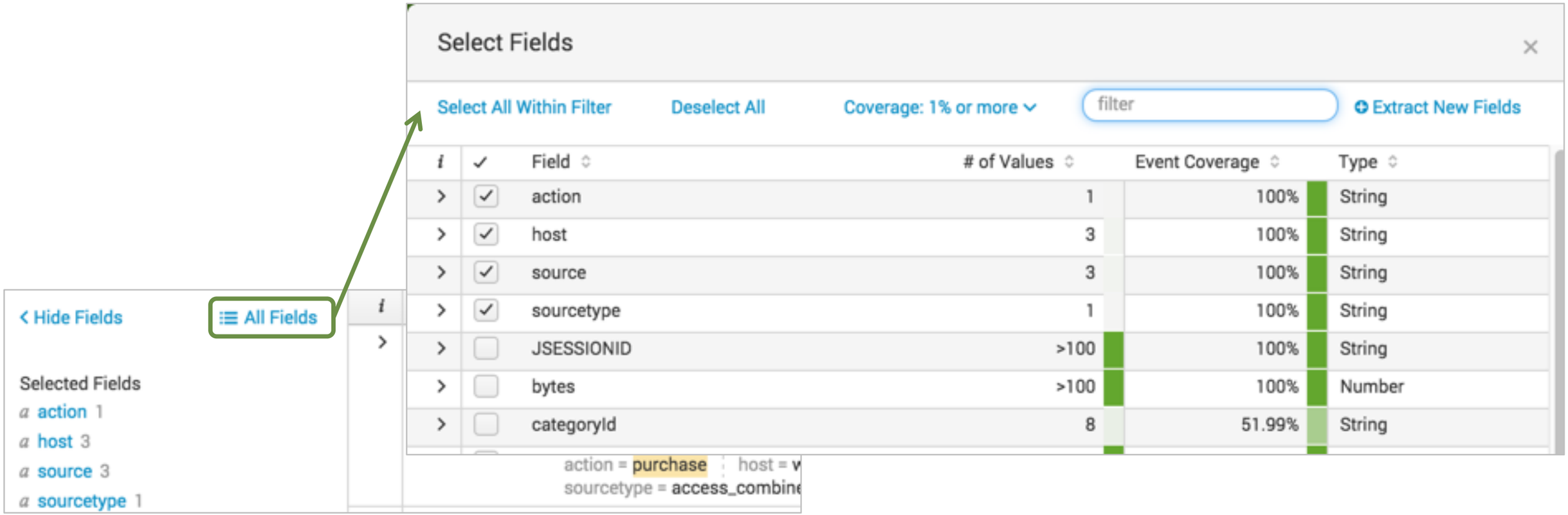
- You can modify selected fields
  1. Click a field in the Fields sidebar
  2. Click **Yes** in the upper right of the field dialog
- Now that it is a selected field, it appears:
  - In the Selected Fields section of the Fields sidebar
  - Below each event where a value exists for that field





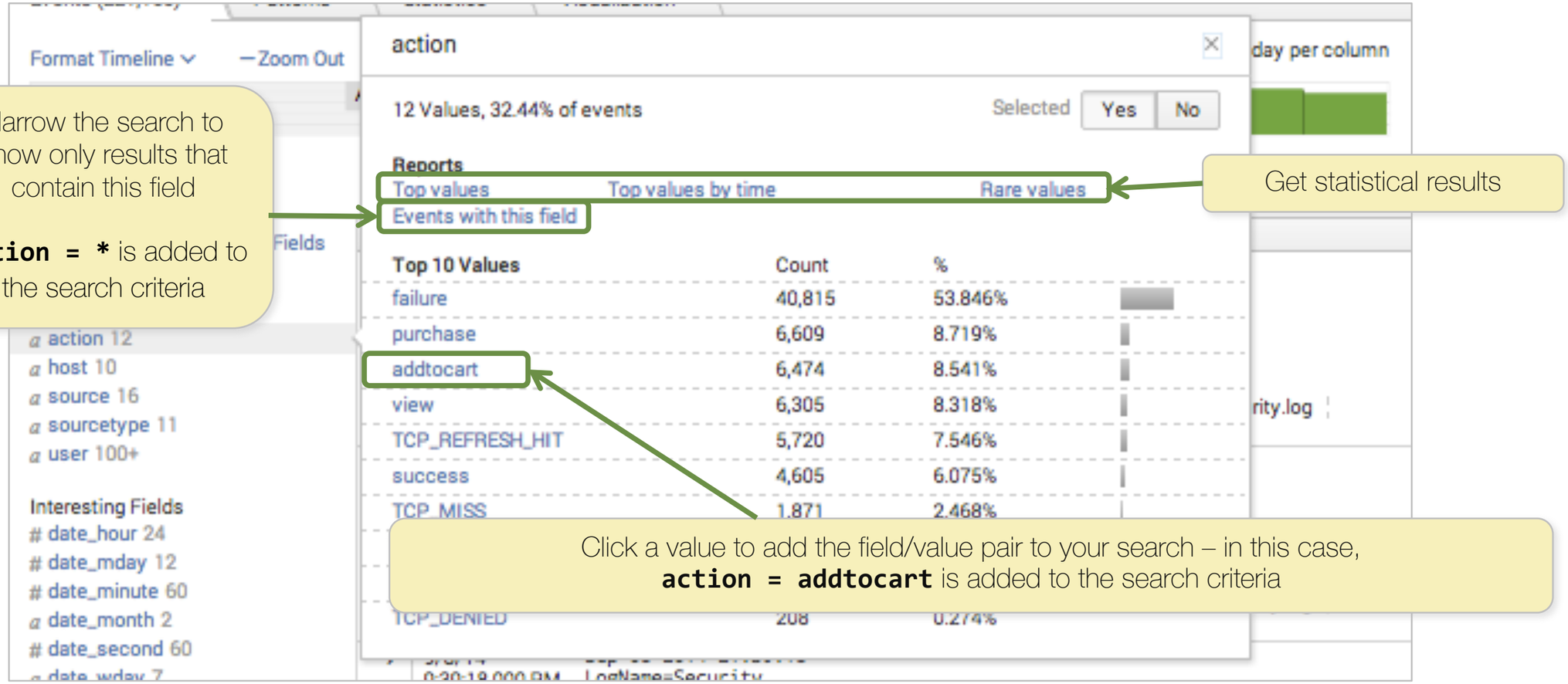
Make Any Field Selected

You can identify other fields as selected fields from All Fields (which shows all of the discovered fields)



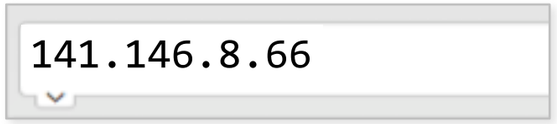
The Field Window

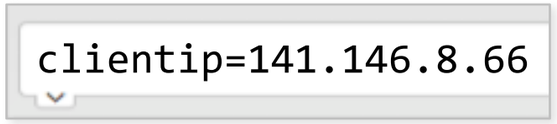
Select a field from the Fields sidebar, then:

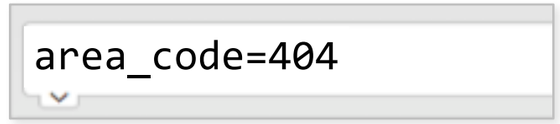


Using Fields in Searches

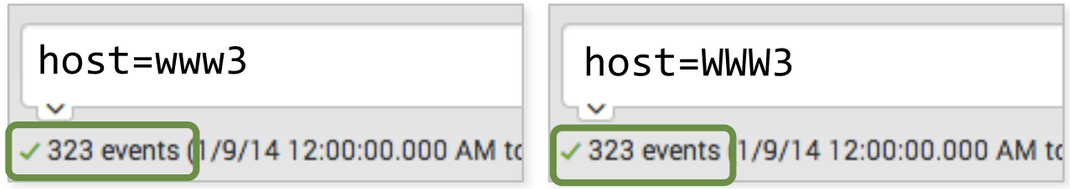
- Efficient way to pinpoint searches and refine results



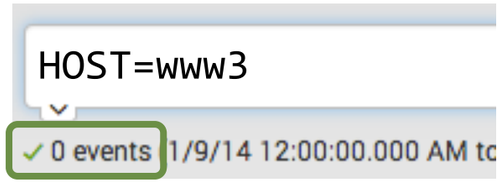




- Field names ARE case sensitive; field values are NOT
  - Example:



These two searches return results



This one does not return results

Using Fields in Searches (cont.)

- For IP fields, Splunk is subnet/CIDR aware

```
clientip="141.146.8.0/24"
```

```
clientip="141.146.8.*"
```

- Use wildcards to match a range of field values
  - Example: **`user=*`** (to display all events that contain a value for user)

```
user=* sourcetype=access* (referer_domain=*.cn OR referer_domain=*.hk)
```

All time

- Use relational operators

With numeric fields

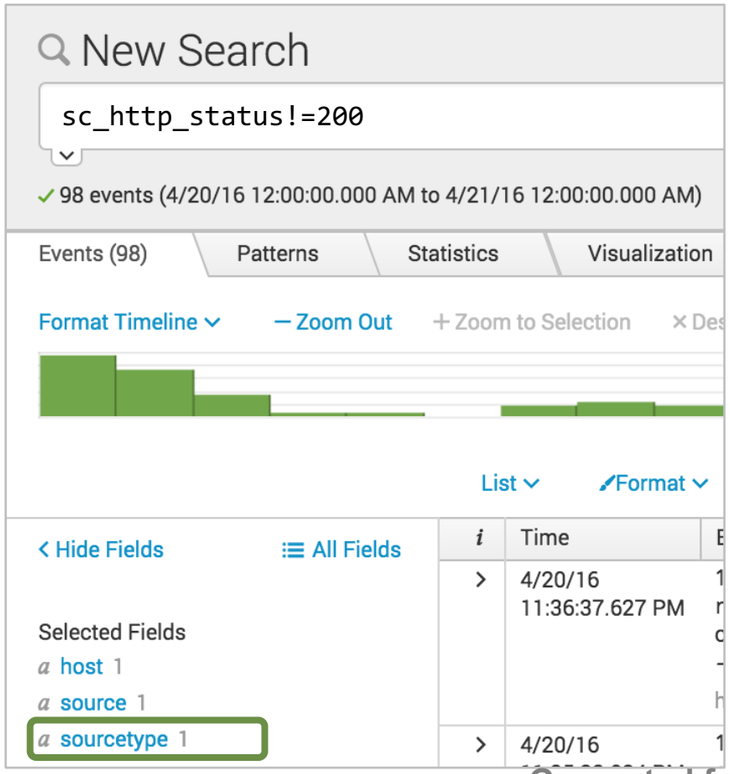
```
src_port>1000 src_port<4000
```

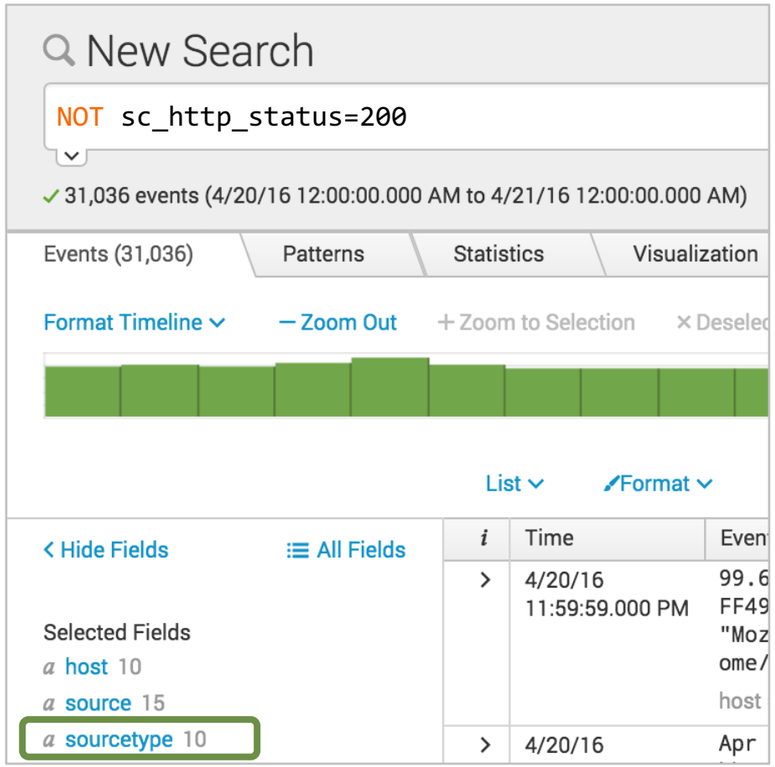
With alphanumeric fields

```
host!=www3
```


Example: != vs. NOT

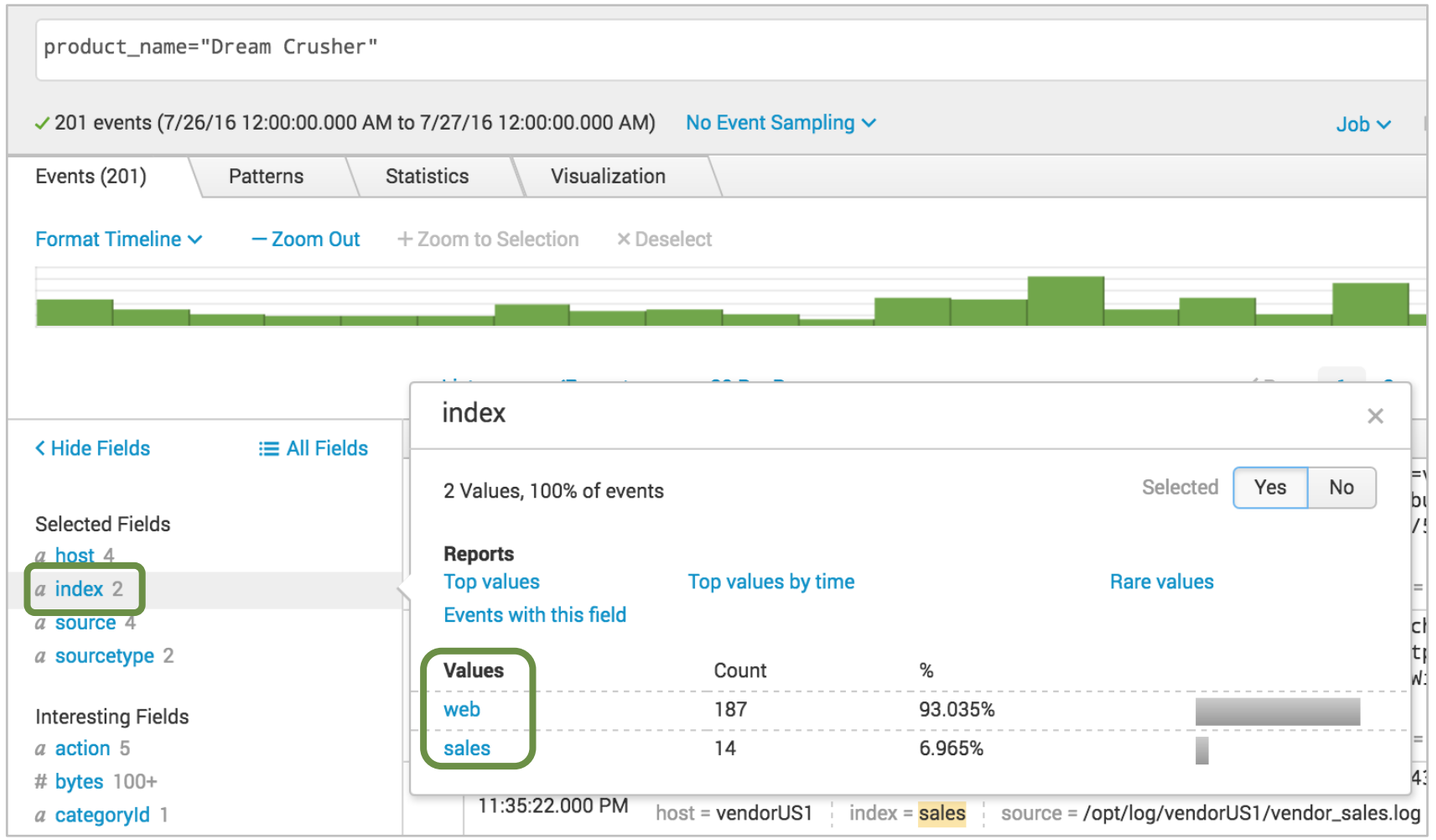
- Note that the search on the left, which uses **!=**, returns 98 events from one sourcetype
- The search on the right, using **NOT**, returns 31,036 events from *ten* sourcetypes





Searching Against the Default Index

- In the Fields sidebar, note the **index** field
- An *index* is a location where Splunk stores – and searches for – event data
- The Splunk administrator configures the index locations that you can search, by default
- In the search shown here, data is returned from two indexes: web and sales



Search Modes: Fast, Smart, Verbose

Search Mode →	Fast	Smart	Verbose
Emphasizes →	Speed	Balance of speed and completeness	Completeness (but slower)
When run with an event search, • Access to Events view? • Field discovery on? • Fields sidebar exists? • Statistics, Visualization tabs empty?	 • Yes • No • Yes • Yes	 • Yes • Yes • Yes • Yes	 • Yes • Yes • Yes • Yes
When run with a reporting/statistical search, • Access to Events view? • Field discovery on? • Fields sidebar exists?	No	No	Yes
Default Search Mode?	No	Yes	No

Module 7: Best Practices for Searching

General Search Practices

- Time is the most efficient filter
- For best performance, specify index values at the beginning of the search string
- Be specific
  - Searching for **`"access denied"`** is always better than searching for **`"denied"`**
  - To make searches more efficient, include as many terms as possible
    - If you want to find events with **`"error"`** and **`"sshd"`** and 90% of the events include **`"error"`**, but only 5% **`"sshd"`**, include both values in the search
- Inclusion is generally better than exclusion
  - Searching for **`"access denied"`** is faster than **`NOT "access granted"`**

Note

Note that search terms are case-*insensitive* and search fields are case-*sensitive.*

General Search Practices (cont.)

- Filter as early as possible
  - For example, remove duplicate events, then sort
- For fastest performance, try to avoid using wildcards at the beginning of a string
- Inconsistent performance can result from using wildcards in the middle of a string, especially if the string contains punctuation or quotes

Time Range Abbreviations

- Time ranges specified in the **Advanced** tab of the time range picker
- Time unit abbreviations include:

```
s = seconds   m = minutes   h = hours   d = days   w = week   mon = months   y = year
```

- **@** symbol "snaps" to the time unit you specify
  - Snapping rounds *down* to the nearest specified unit
  - Example: Current time when the search starts is 09:37:12

    -30m@h looks back to 09:00:00

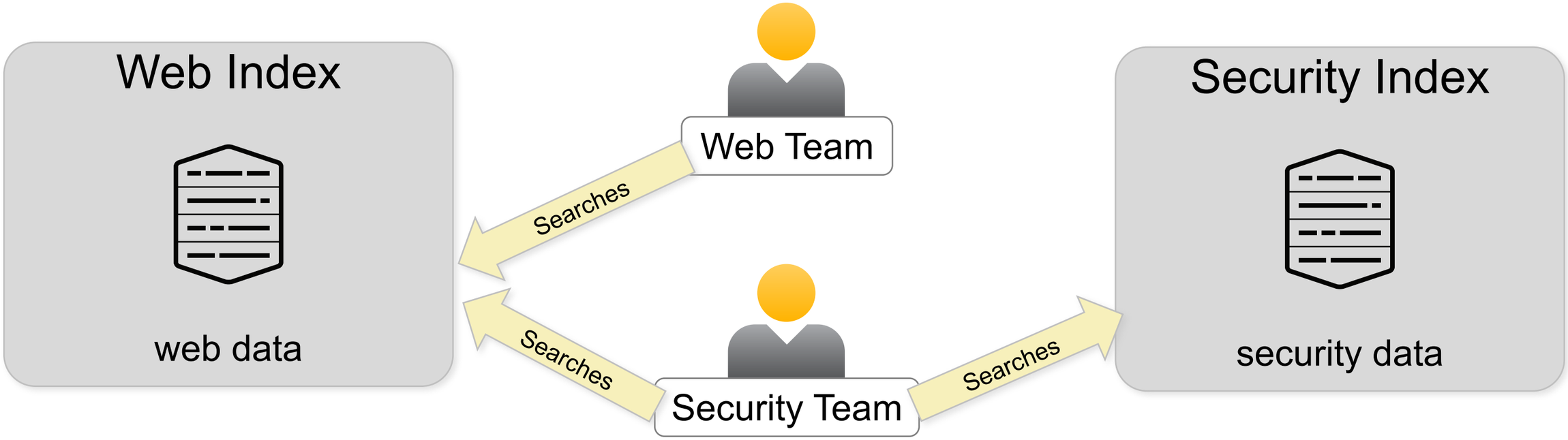
Time Range: `earliest` and `latest`

- You can also specify a time range in the search bar
- To specify a beginning and an ending for a time range, use **`earliest`** and **`latest`**
- Examples:

`earliest=-h`	looks back one hour
`earliest=-2d@d latest=@d`	looks back from two days ago, up to the beginning of today

Indexes

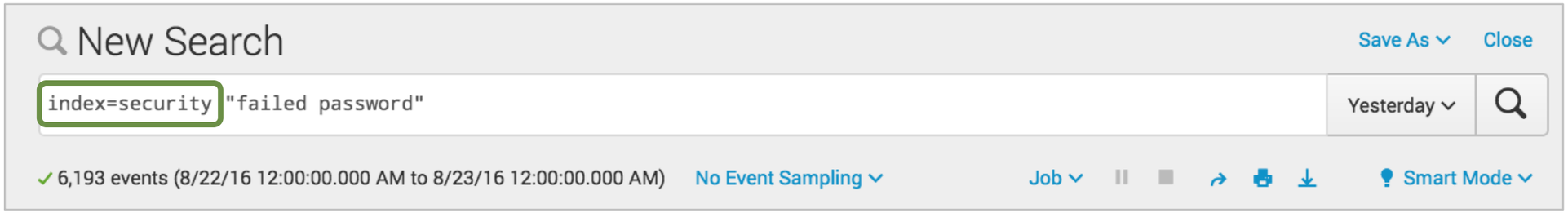
- An *index* is a location where Splunk stores and searches for event data



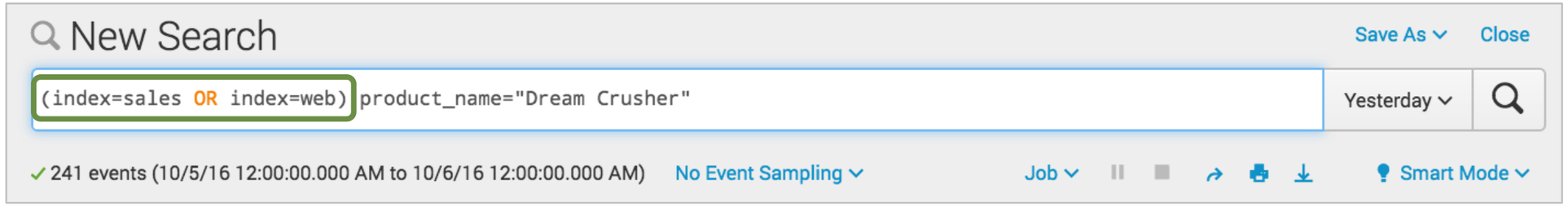
- Administrators segregate data into separate indexes to limit access by Splunk role

Working with Indexes

- This search returns event data from the security index

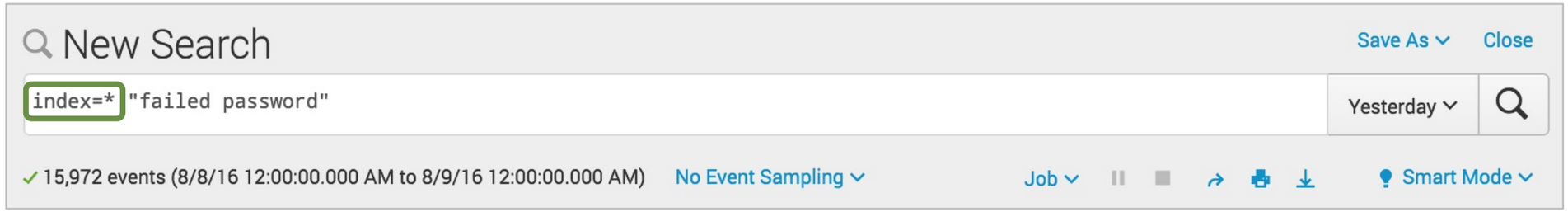


- It is possible to specify multiple index values



Working with Indexes (cont.)

It is possible to use wildcards – *, %, _, etc. – in index values



Note 1

Although `index=*` is a valid search, better performance is always obtained by specifying one or more specific index values.

Note 2

For best performance, specify the index values at the beginning of the search string.

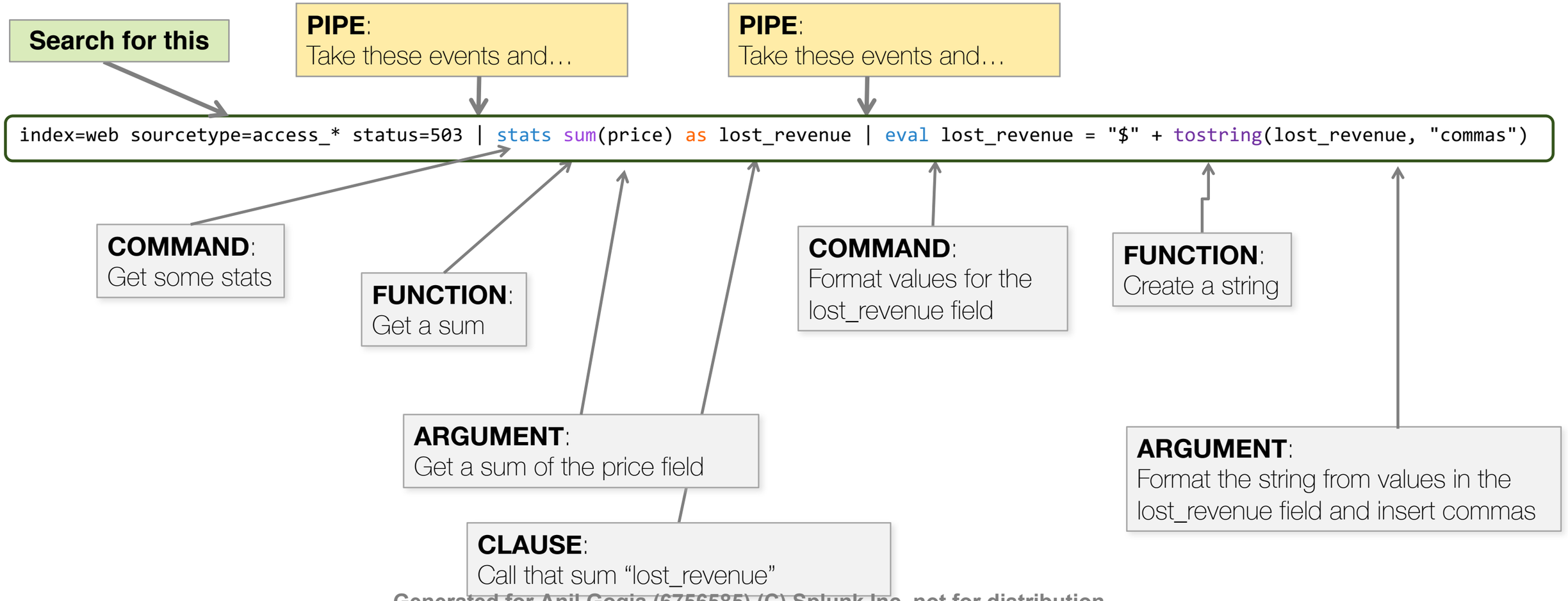
Module 8:
Splunk's Search Language

Module Objectives

- Understand the search pipeline
- Understand search syntax concepts
- Use the table, fields and sort commands

Search Pipeline Example

This diagram represents a search, broken into its syntax components:

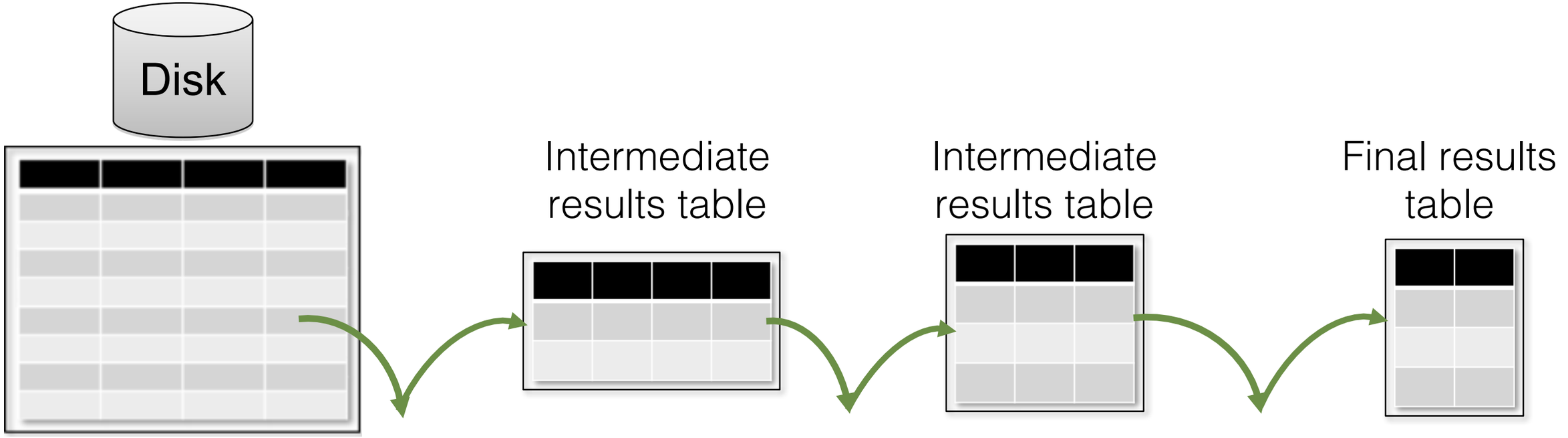


Search Language Syntax Concepts

Searches are made up of 5 basic components

- **Search terms** – what are you looking for?
  - Keywords, phrases, Booleans, etc.
- **Commands** – what do you want to do with the results?
  - Create a chart, compute statistics, evaluate and format, etc.
- **Functions** – how do you want to chart, compute, or evaluate the results?
  - Get a sum, get an average, transform the values, etc.
- **Arguments** – are there variables you want to apply to this function?
  - Calculate average value for a specific field, convert milliseconds to seconds, etc.
- **Clauses –** how do you want to group or rename the fields in the results?
  - Give a field another name or group values by or over

The Search Pipeline




```
index=security sourcetype=linux_secure fail   | top user   | fields – percent
```

Fetch events from disk that match

Summarize into table of top 10 users

Remove column showing percentage

Making the Pipeline More Readable

- Clicking **Ctrl - ** (Windows) or ⌘ - **** (MacOS) in the search box puts each pipe in the pipeline on a separate line
- For example, this:

New Search

```
index=web sourcetype=access_* status=503 | stats sum(price) as lost_revenue | eval lost_revenue = "$" + tostring(lost_revenue, "commas")
```

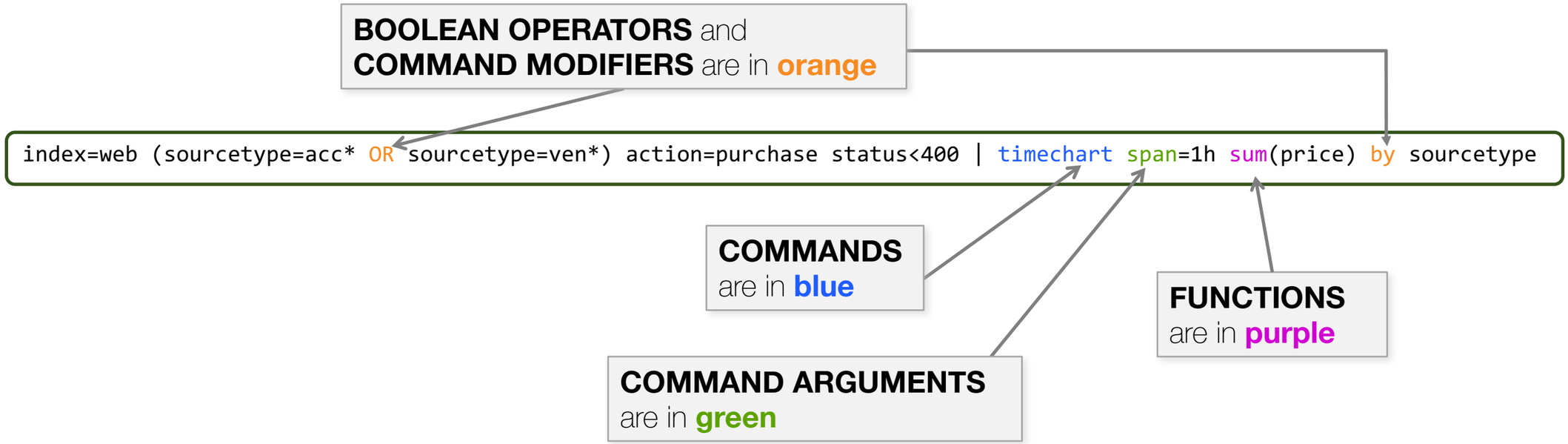
- Is transformed to this:

New Search

```
index=web sourcetype=access_* status=503
| stats sum(price) as lost_revenue
| eval lost_revenue = "$" + tostring(lost_revenue, "commas")
```

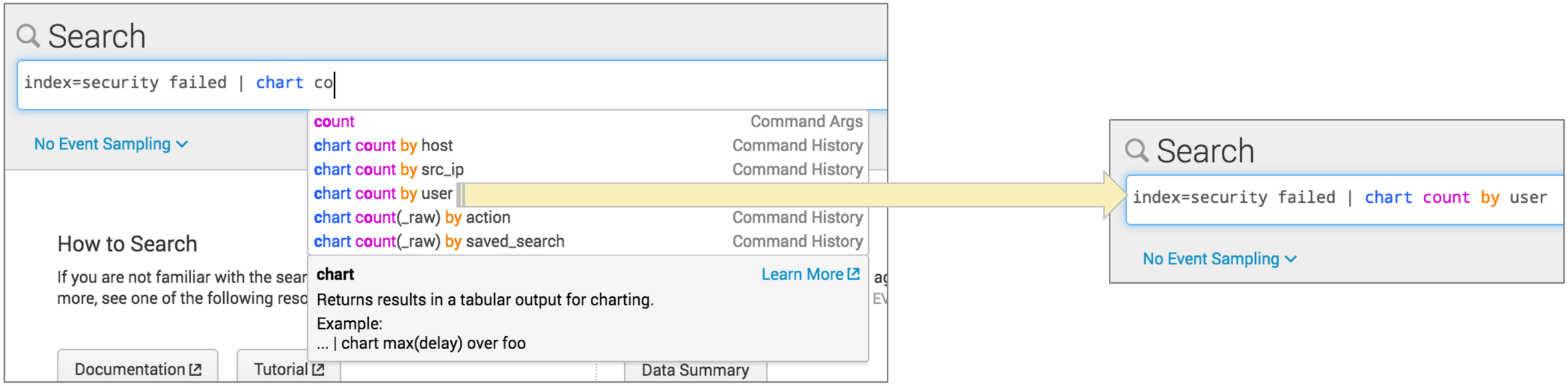
Syntax Coloring

- As you type, some parts of the search string are automatically colored
- The color is based on the search syntax
  - The rest of the search string remains black



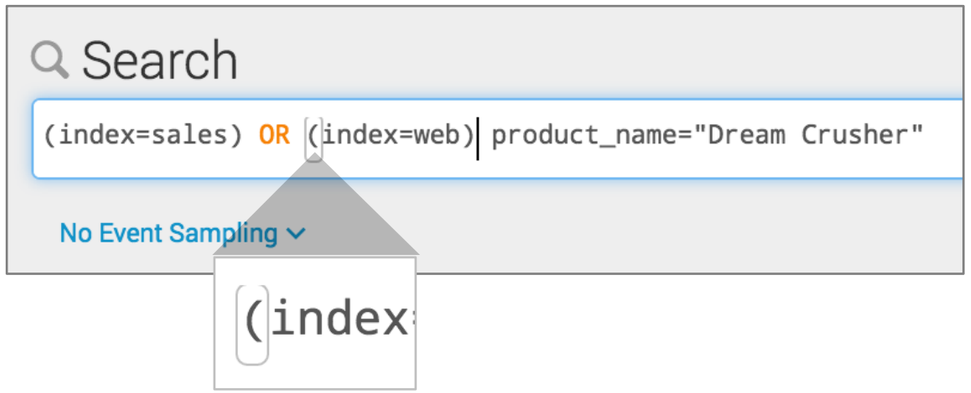
Search Assistant

- The Search Assistant provides an autocomplete feature
- It provides convenient reminders about commands available at any given point in the search string
  - If desired, click a reminder to have its contents inserted into the search



Search Assistant and Parentheses

- The Search Assistant provides help to match parentheses as you type
- When an end parenthesis is typed, the corresponding beginning parenthesis is automatically highlighted
  - If a beginning parenthesis cannot be found, *nothing* is highlighted



Beginning parenthesis found!

Beginning parenthesis NOT found!

Creating a Table

- **`table`** command returns a table formed by only fields in the argument list
- Columns are displayed in the order given in the command
  - Column headers are field names
  - Each row is an event
  - Rows are field values

Scenario

Display the **`clientip`**, **`action`**, **`productId`**, and **`status`** of customer interactions in the online store for the last 4 hours.

```
index=web sourcetype=access_combined
| table clientip, action, productId, status
```

clientip	action	productId	status
223.205.219.67			200
69.80.0.18	view	WC-SH-A02	200
69.80.0.18		SF-BVS-01	408
91.205.189.15	view	FS-SG-G03	200
91.205.189.15	view	CU-PG-G06	200
91.205.189.15	view	WC-SH-A02	200
91.205.189.15	remove	WC-SH-A01	200
91.205.189.15			200

Renaming Fields

- To change the name of a field, use the **`rename`** command
- Useful for giving fields more meaningful names
- When including spaces or special characters in field names, use double straight quotes:
  - A `rename productId as ProductID`
  - B `rename action as "Customer Action"`
  - C `rename status as "HTTP Status"`

Scenario

Display the `clientip`, `action`, `productId`, and `status` of customer interactions in the online store for the last 4 hours.

```
index=web sourcetype=access_combined
| table clientip, action, productId, status
| rename productId as ProductID, A
  action as "Customer Action", B
  status as "HTTP Status" C
```

clientip	Customer Action B	ProductID A	HTTP Status C
141.146.8.66		MB-AG-T01	200
141.146.8.66		WC-SH-A01	200
195.80.144.22		DC-SG-G02	200
141.146.8.66		WC-SH-A02	200
195.80.144.22		SC-MG-G10	200
141.146.8.66		PZ-SG-G05	200
195.80.144.22	purchase		200
195.80.144.22	purchase	SC-MG-G10	200

`fields` Command

- Field extraction is one of the most costly parts of a search
- **`fields`** command allows you to include or exclude specified fields in your search or report
- To include, use **`fields +(default)`**
  - Occurs before field extraction
  - Improves performance
- To exclude, use **`fields -`**
  - Occurs after field extraction
  - No performance benefit
  - Exclude fields used in search to make the table/display easier to read

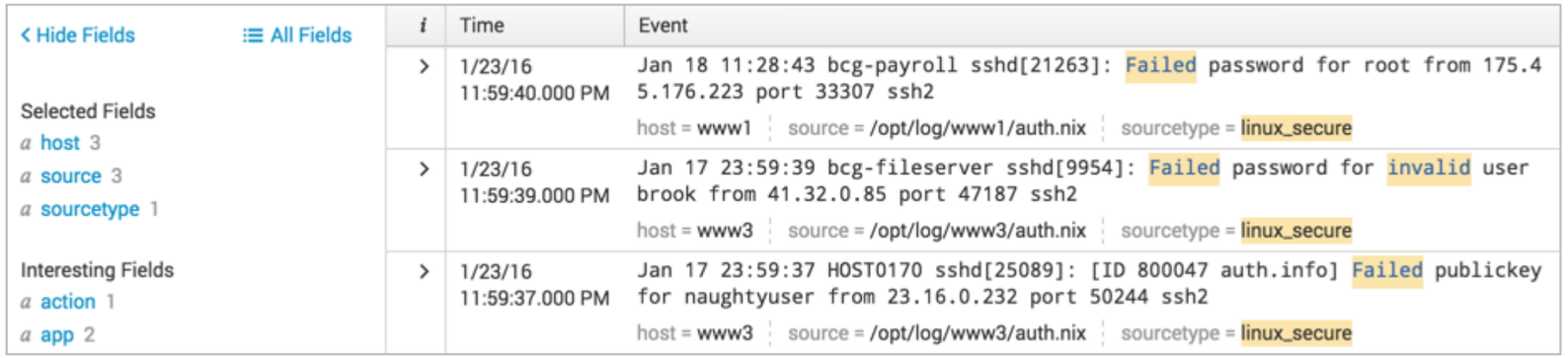
`fields` Command – Examples

Improves performance – only the fields you specify are extracted

Scenario

Display network failures during the previous week.

Returned **6,567** results by scanning **6,567** events in **1.425** seconds:



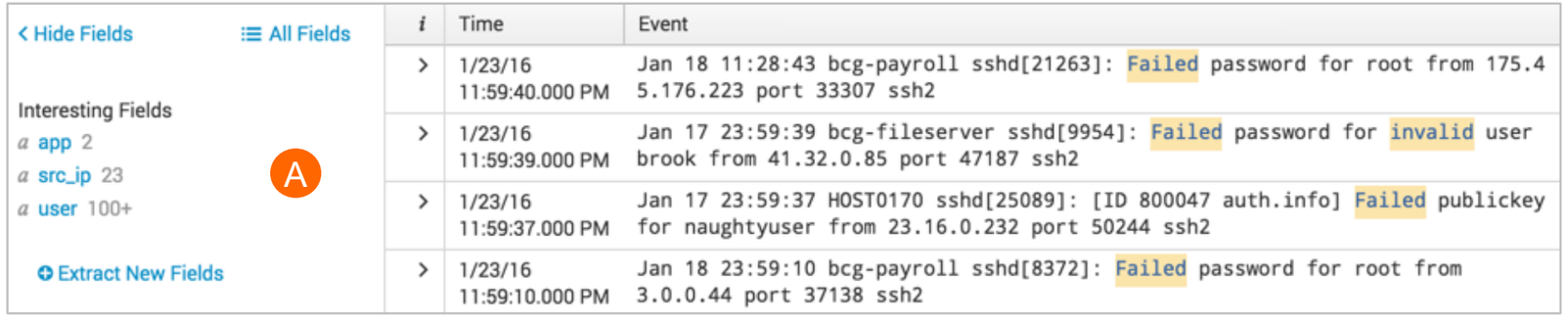
Scenario

Display network failures during the previous week. Retrieve only **user**, **app**, and **src_ip**.

```
index=security
sourcetype=linux_secure
(fail* OR invalid)
| fields user, app, src_ip
```

A

Returned **6,567** results by scanning **6,567** events in **0.753** seconds:



dedup Command

Use dedup to remove duplicates from your results

```
index=sales sourcetype=vendor_sales | table VendorCountry, VendorStateProvince, VendorCity, Vendor
```

VendorCountry	VendorStateProvince	VendorCity	Vendor
United States	Texas	Waco	Wow Games
United States	Utah	Cedar City	Woody's Games
United States	Virginia	Staunton	Woody's Games
United States	Utah	Cedar City	Woody's Games
United States	Utah	Cedar City	Woody's Games
Australia	Western Australia	Perth	Wonderland Hobbies
Australia	Western Australia	Perth	Wonderland Hobbies

…| dedup Vendor | table …

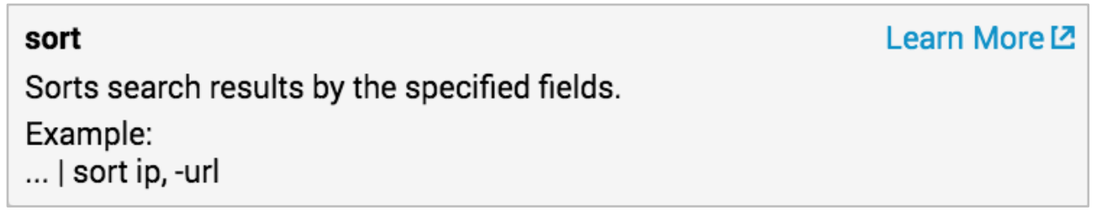
VendorCountry	VendorStateProvince	VendorCity	Vendor
United States	Texas	Waco	Wow Games
United States	Utah	Cedar City	Woody's Games
Australia	Western Australia	Perth	Wonderland Hobbies

…| dedup VendorCity, Vendor | table …

VendorCountry	VendorStateProvince	VendorCity	Vendor
United States	Texas	Waco	Wow Games
United States	Utah	Cedar City	Woody's Games
United States	Virginia	Staunton	Woody's Games
Australia	Western Australia	Perth	Wonderland Hobbies

`sort` Command

- Use **`sort`** to order your results in **`+ ascending`** (default) or **`- descending`**
- To limit the returned results, use the **`limit`** option




```
... | sort limit=20 -categoryId, product_name
... | sort 20 count
```

`sort` Command (cont.)

`sort -/+<fieldname>` sign followed by fieldname sorts results in the sign's order

`sort -/+ <fieldname>` sign followed by space and then fieldname applies sort order to all following fields without a different explicit sort order

```
index=sales sourcetype=vendor_sales
| dedup Vendor
| sort - VendorCountry, +VendorStateProvince, VendorCity, Vendor
| table VendorCountry, VendorStateProvince, VendorCity, Vendor
```

VendorCountry	VendorStateProvince	VendorCity	Vendor
United States	Arizona	Yuma	Yumster Games
United States	Arizona	Tucson	Boothill Games
United States	Arizona	Phoenix	Rising Games
United States	Arizona	Phoenix	Phoenix Games
United States	Arizona	Flagstaff	Flaggin Games

Module 9: Transforming Commands

Module Objectives

Identify and use the following commands and their functions:

- `top`
- `rare`
- `Stats`
- `dedup`

Getting Top Values

- The **top** command finds the most common values of a given field in the result set
  - By default, returns top 10 results

src_ip	count	percent
10.3.10.46	53	19.629630
10.2.10.163	50	18.518519
10.1.10.172	43	15.925926
87.194.216.51	23	8.518519
217.132.169.69	19	7.037037
188.143.232.202	11	4.074074
69.80.0.18	10	3.703704
216.221.226.11	9	3.333333
142.233.200.21	8	2.962963
84.34.159.23	7	2.592593

Scenario

During the last 60 minutes, which IP addresses generated the most attacks?

```
index=security sourcetype=linux_secure
(fail* OR invalid)
| top src_ip
```


`top` Command

- By default, output displays in table format
- Automatically returns **count** and **percent** columns
- Common constraints:

```
limit   countfield   showperc
```

Note

Refer to docs.splunk.com for the other available options.

top Learn More

Displays the most common values of a field.

Example:
... | top limit=20 url

`top` Command – Single Field Example

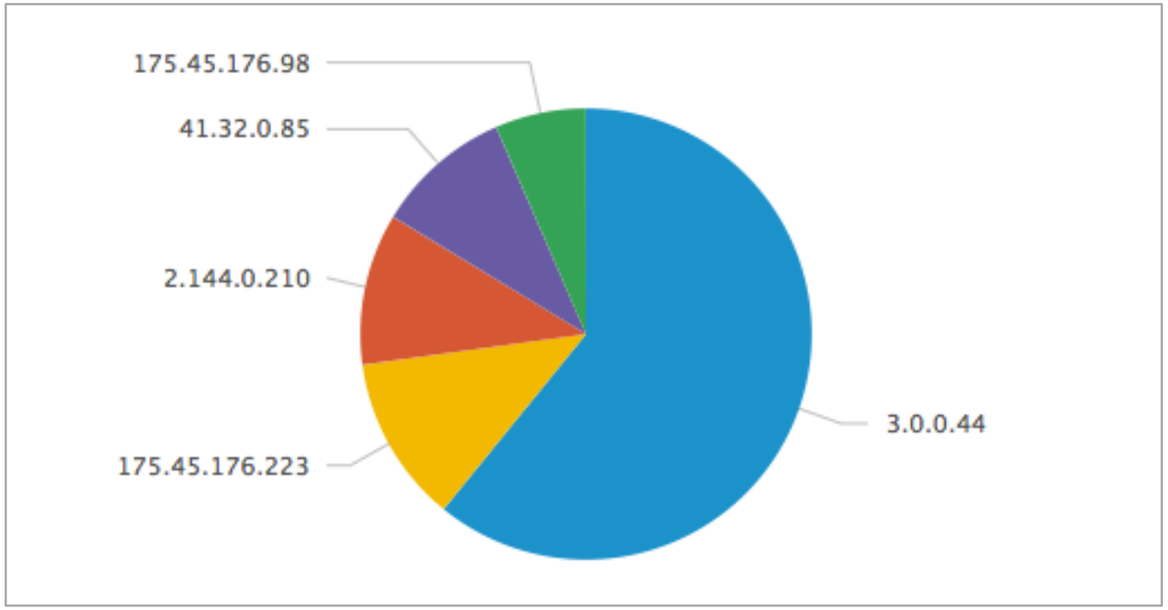
- `limit=#` returns this number of results
- By default, 10 results are displayed
- `limit=0` returns unlimited results

```
sourcetype=linux_secure index=security
(fail* OR invalid)
| top limit=5 src_ip
```

Scenario

During the last hour, display the top 5 IPs that generated the most attacks.

src_ip	count	percent
10.2.10.163	73	27.037037
10.1.10.172	42	15.555556
10.3.10.46	41	15.185185
87.194.216.51	19	7.037037
12.130.60.4	17	6.296296



top – Multiple Field Example

- If the **showperc** is not included – or it is included and set to **t** – a **percent** column is displayed
- If **showperc=f**, then a percent column is NOT displayed

Scenario

Display the top 3 common values for users and web categories browsed during the last 24 hours.

```
index=network sourcetype=cisco_wsa_squid
| top user A x_webcat_code_full limit=3 B
```

user	x_webcat_code_full	count	percent
apucci@buttercupgames.com	Games	79	6.152648
arangel@buttercupgames.com A	Society and Culture	61	4.750779
rerde@buttercupgames.com	Arts and Entertainment B	54	4.205607

top – Single Field with by Clause Example

Scenario

Display the top 3 common web categories browsed by each user during the last 24 hours.

```
index=network sourcetype=cisco_wsa_squid
| top x_webcat_code_full B by user A limit=3
```

user	x_webcat_code_full	count	percent
acurry@buttercupgames.com A	Uncategorized URLs	10	71.428571
acurry@buttercupgames.com	Sports and Recreation B	1	7.142857
acurry@buttercupgames.com	Society and Culture	1	7.142857
adombrowski@buttercupgames.com	Computers and Internet	2	33.333333
adombrowski@buttercupgames.com	Spiritual Healing	1	16.666667
adombrowski@buttercupgames.com	Shopping	1	16.666667

top – Specifying Options

- By default, the value of the **countfield** is **count**
- **countfield=*string*** provides the name of a new field to write the count value

Scenario

Display the top 3 user/web categories browsed combinations during the last 24 hours. Rename the count field and show count, but not the percentage.

```
index=network sourcetype=cisco_wsa_squid
| top user x_webcat_code_full limit=3 (A)
  countfield="Total Viewed"(B) showperc=f
```

user	x_webcat_code_full	Total Viewed
apucci@buttercupgames.com (A)	Games	79 (B)
arangel@buttercupgames.com	Society and Culture	61
rerde@buttercupgames.com	Arts and Entertainment	54

Note

A Boolean can be t/f, true/false, as well as 1/0.

`rare` Command

- The **`rare`** command returns the least common field values of a given field in the results
- Options are identical to the **`top`** command

Scenario

Which product is the least sold by Buttercup Games vendors over the last 60 minutes?

```
index=sales sourcetype=vendor_sales
  | rare product_name showperc=f limit=1
```

Events | Patterns | Statistics (1) | Visualization

20 Per Page | Format | Preview

product_name	count
Fire Resistance Suit of Provolone	1

`stats` Command

- `stats` enables you to calculate statistics on data that matches your search criteria
- Common functions include:
  - `count` – returns the number of events that match the search criteria
  - `distinct_count, dc` – returns a count of unique values for a given field
  - `sum` – returns a sum of numeric values
  - `avg` – returns an average of numeric values
  - `list` – lists all values of a given field
  - `values` – lists unique values of a given field

Note

To view all of the functions for `stats`, please see:
http://docs.splunk.com/Documentation/Splunk/Latest/SearchReference/CommonStatsFunctions

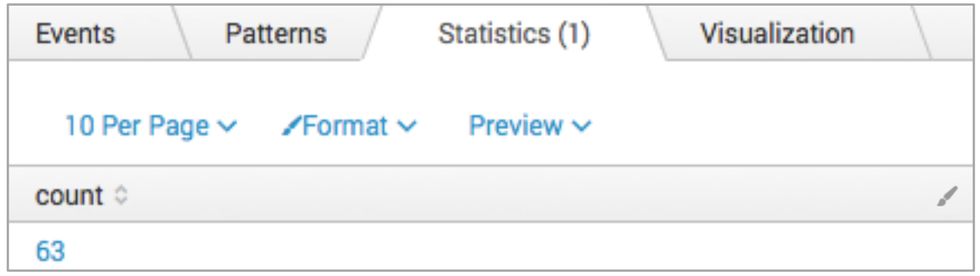
`stats` Command – `count`

- `count` returns the number of matching events based on the current search criteria
- Use the `as` clause to rename the `count` field

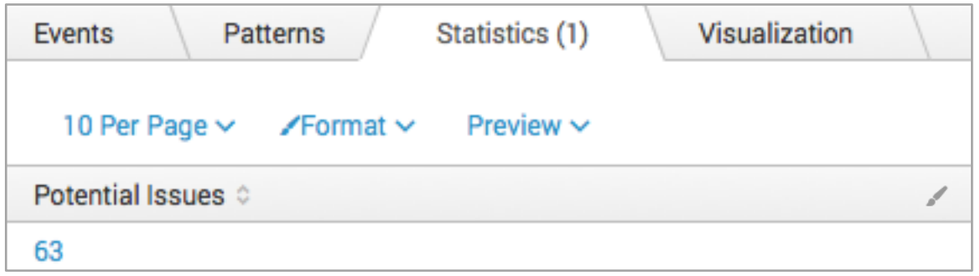
Scenario

Count the invalid or failed login attempts during the last 60 minutes.

```
index=security sourcetype=linux_secure
(invalid OR failed)
| stats count
```




```
index=security sourcetype=linux_secure
(invalid OR failed)
| stats count as "Potential Issues"
```



`stats` Command – `count(field)`

Adding a *`field`* as an argument to the **`count`** function returns the number of events where a value is present for the specified field

Scenario

Count the number of events during the last 15 minutes that contain a vendor action field. Also count the total events.

```
index=security sourcetype=linux_secure
| stats count(vendor_action) as ActionEvents,   A
  count as TotalEvents   B
```

Events (69) | Patterns | Statistics (1) | Visualization

20 Per Page | Format | Preview

A ActionEvents	B TotalEvents
61	69

stats Command – by *fields*

Scenario

Count the number of events by user, app, and vendor action during the last 15 minutes.

```
index=security sourcetype=linux_secure
| stats count by user, app, vendor_action
```

- **by** clause returns a count for each value of a named field or set of fields
- Can use any number of fields in the **by** *field* list

user	app	vendor_action	count
abc	sshd	Failed	1
admin	sshd	Failed	4
administrator	sshd	Failed	1
alex	sshd	Failed	1
apache	sshd	Failed	1
backup	sshd	Failed	1
ben	sshd	Failed	1
bin	sshd	Failed	2
britany	sshd	Failed	1
daemon	sshd	Failed	1

`stats` Command – `distinct_count(field)`

- `distinct_count()` or `dc()` provides a count of how many unique values there are for a given field in the result set
- This example counts how many unique values for `s_hostname`

Scenario

How many unique websites have our employees visited in the last 4 hours?

```
index=network sourcetype=cisco_wsa_squid
 | stats dc(s_hostname) as "Websites visited:"
```

Events | Patterns | Statistics (1) | Visualization

20 Per Page | Format | Preview

Websites visited:
25

`stats` Command – `sum(field)`

Scenario

How much bandwidth did employees spend at each website during the past week?

```
index=network sourcetype=cisco_wsa_squid
| stats sum(sc_bytes) as Bandwidth by s_hostname
| sort -Bandwidth
```

(A) (B) (C)

For fields with a numeric value, you can sum the actual values of that field

Events | Patterns | Statistics (54) | Visualization

20 Per Page | Format | Preview

s_hostname	Bandwidth
www.gctsindia.in	4866091
www.heals.co.uk	1233522
www.animationmagazine.net	1203433
www.kare11.com	1202753
www.finedinings.com	747407

`stats` Command – `sum(field)` – (cont.)

Scenario

Report the number of retail units sold and sales revenue for each product during the previous week.

```
index=sales sourcetype=vendor_sales
| stats A count(price) as "Units Sold"
B sum(price) as "Total Sales"   by product_name C
| sort -"Total Sales" D
```

- A A single `stats` command
- B can have multiple functions
- C The `by` clause is applied to both functions
- D `sort` Total Sales in descending order

product_name	Units Sold (A)	Total Sales (B)
Dream Crusher	73	2919.27
Manganiello Bros.	53	2119.47
World of Cheese	66	1649.34
SIM Cubicle	81	1619.19
Orvil the Wolverine	35	1399.65
Final Sequel	49	1224.51
Mediocre Kingdoms	42	1049.58
Curling 2014	48	959.52
Benign Space Debris	25	624.75
Manganiello Bros. Tee	62	619.38

C

D

`stats` Command – `avg(field)`

- The **avg** function provides the average numeric value for the given numeric field
- An event is not considered in the calculation if it:
  - Does not have the field
  - Has an invalid value for the field

Scenario

What is the average bandwidth used for each website usage type?

```
index=network sourcetype=cisco_wsa_squid
| stats avg(sc_bytes) as "Average Bytes" A
  by usage B
```

usage	Average Bytes
Borderline	13553.723173
Business	11277.155763
B Personal	A 14874.552763
Unknown	10724.935021
Violation	8982.340426

`stats` Command – `list(field)`

- `list` function lists all field values for a given field
- This example lists the websites visited by each employee
  - Security logs generate an event for each network request
    - This causes the same hostname to appear multiple times
  - To return a list of "unique" field values, use the `values` function

Scenario

Which websites has each employee accessed during the last 60 minutes?

```
index=network sourcetype=cisco_wsa_squid
| stats list(s_hostname) as "Websites visited:"
  by cs_username
```

cs_username	Websites visited:
basselin@buttercupgames.com	- -
blu@buttercupgames.com	www.lowermybills.com
cquinn@buttercupgames.com	- static.pochta.ru
dhale@buttercupgames.com	-
dpiazza@buttercupgames.com	www.ayles.com www.ayles.com www.ayles.com www.ayles.com www.ayles.com www.ayles.com

`stats` Command – `values(field)`

Scenario

Display by IP address the names of users who have failed access attempts in the last 60 minutes.

```
index=security sourcetype=linux_secure fail*
| stats values(user) as "User Names",
  count(user) as Attempts by src_ip
```

`values` function lists unique values for the specified field

src_ip	User Names	Attempts
1.0.32.67	root	2
10.232.44.142	twilliam	1
10.232.44.71	jsimon1	1
175.45.176.223	gbottazzi oracle root scanner user	37
175.45.176.98	abc andrew cvs enquiries logs michael test test3	8

Module 10:
Creating Reports and Dashboards

Module Objectives

- Save a search as a report
- Edit a report
- Create reports that display statistics (tables) or visualizations (charts)
- Create a dashboard
- Add a report to a dashboard
- Add a pivot to a dashboard
- Edit a dashboard

Reports

- Reports are saved searches
- Reports can show events, statistics (tables), or visualizations (charts)
- Running a report returns fresh results each time you run it
- Statistics and visualizations allow you to drill down by default to see the underlying events
- Reports can be shared and added to dashboards
- There are two ways to create a report: pivot or search

Smart Naming

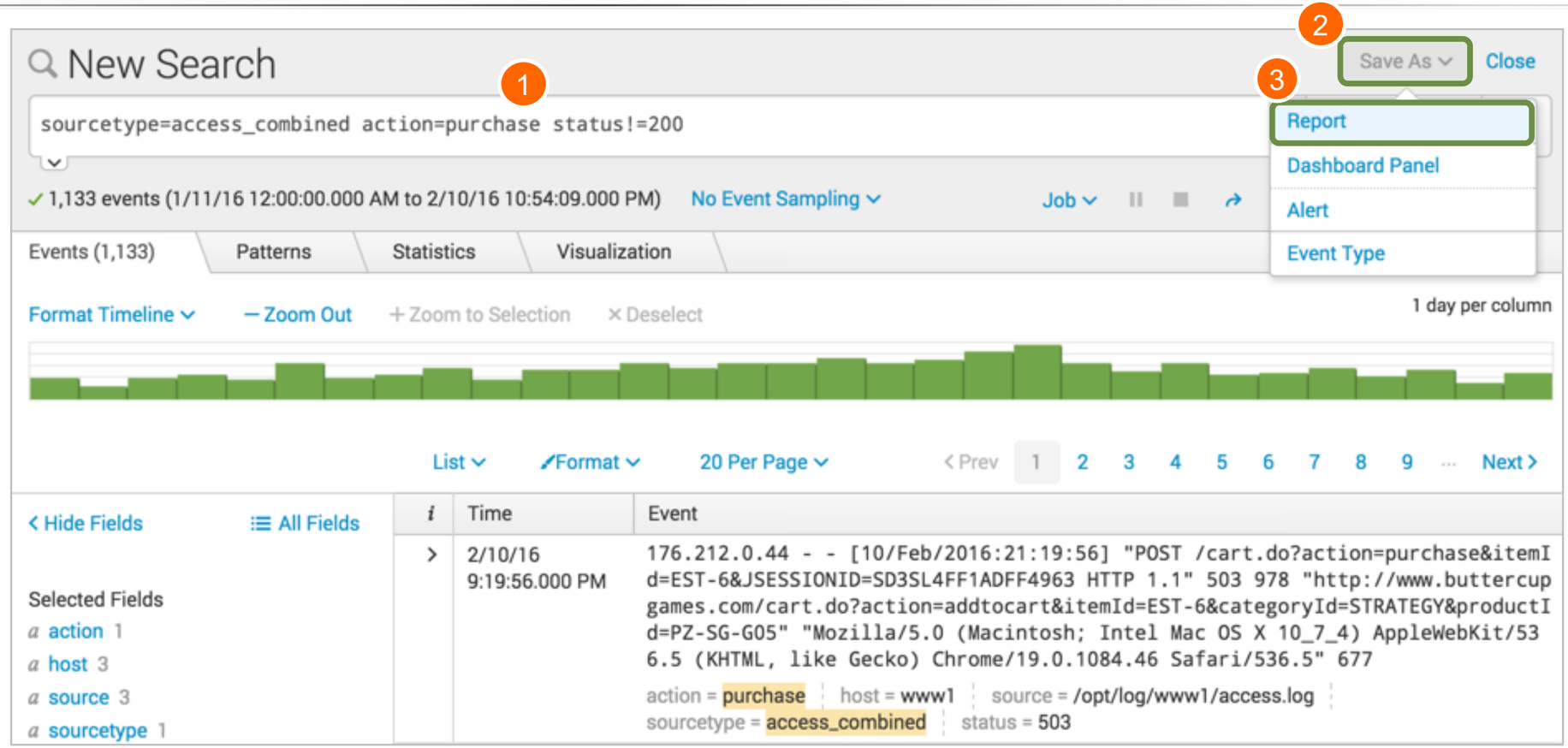
- Before you begin using Splunk on the job, define a naming convention so you can always find your reports and tell them apart
- For example, you can create something simple like this:
  - <group>_<object>_<description>
    - **group**: the name of the group or department using the knowledge object such as sales, IT, finance, etc.
    - **object**: report, dashboard, macro, etc.
    - **description**: WeeklySales, FailedLogins, etc.
  - Using this example, a quarterly sales report can be identified as:
    - Sales_Report_QuarterlySalesRevenue

Note

If you set up naming conventions early in your implementation, you can avoid some of the more challenging object naming issues. The example is a suggestion. The details are found in the Splunk product documentation: http://docs.splunk.com/Documentation/Splunk/latest/Knowledge/Developnamingconventionsforknowledgeobjecttitles

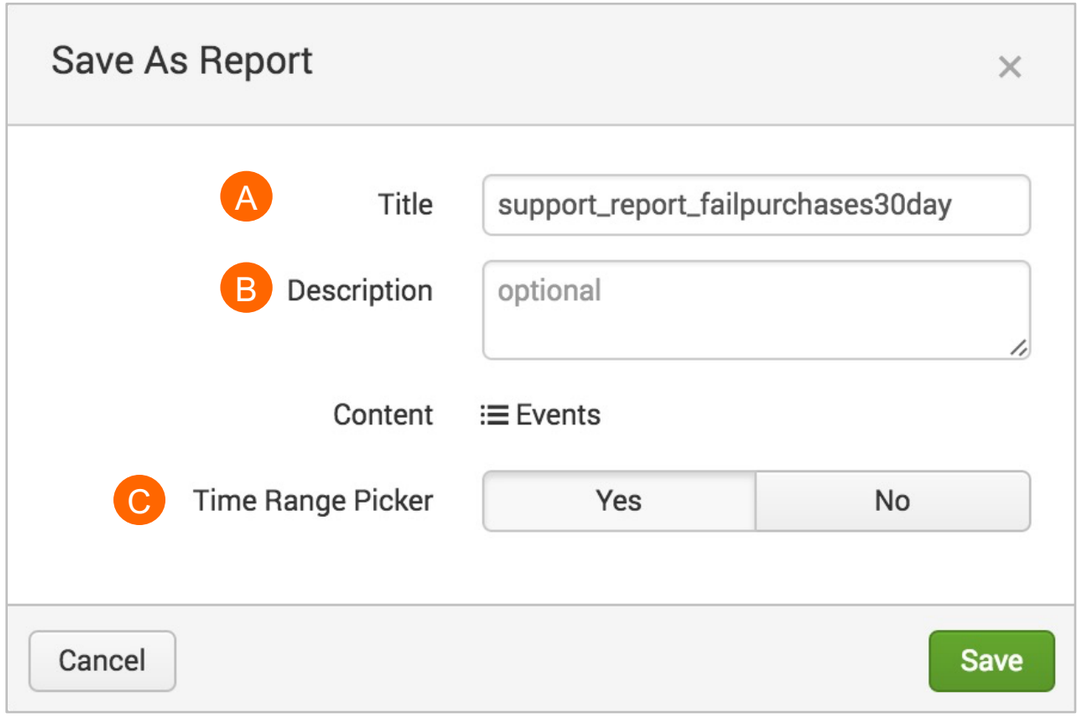
Create a Report from Search

1. Run a search
2. Select **Save As**
3. Select **Report**



Create a Report from Search (cont.)

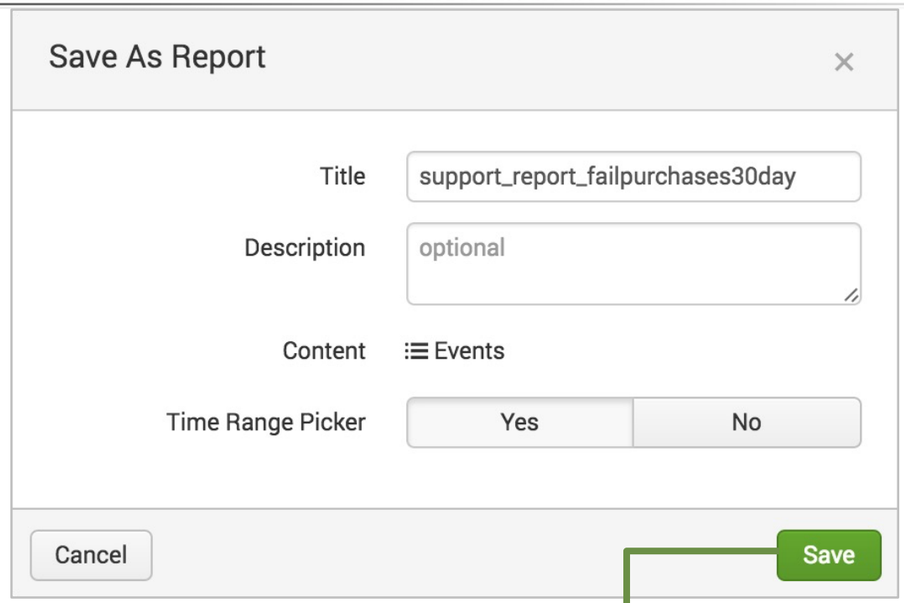
- A Give the report a meaningful title (required)
- B Specify a description (optional)
- C Select whether to include or not to include a time range picker
  - The report will be saved with the time range that was selected when it was created
  - Adding a time range picker allows you to adjust the time range of the report when you run it

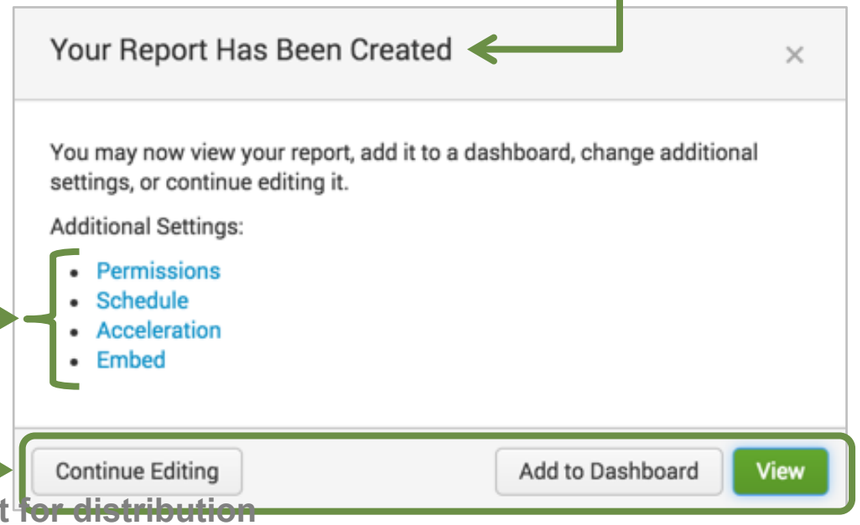


Create a Report from Search (cont.)

You can change Additional Settings, as well as use the dialog buttons:

- Click **Continue Editing** to make changes to your report
- Click **Add to Dashboard** to add your report to a dashboard
- Click **View** to display your report or run it again



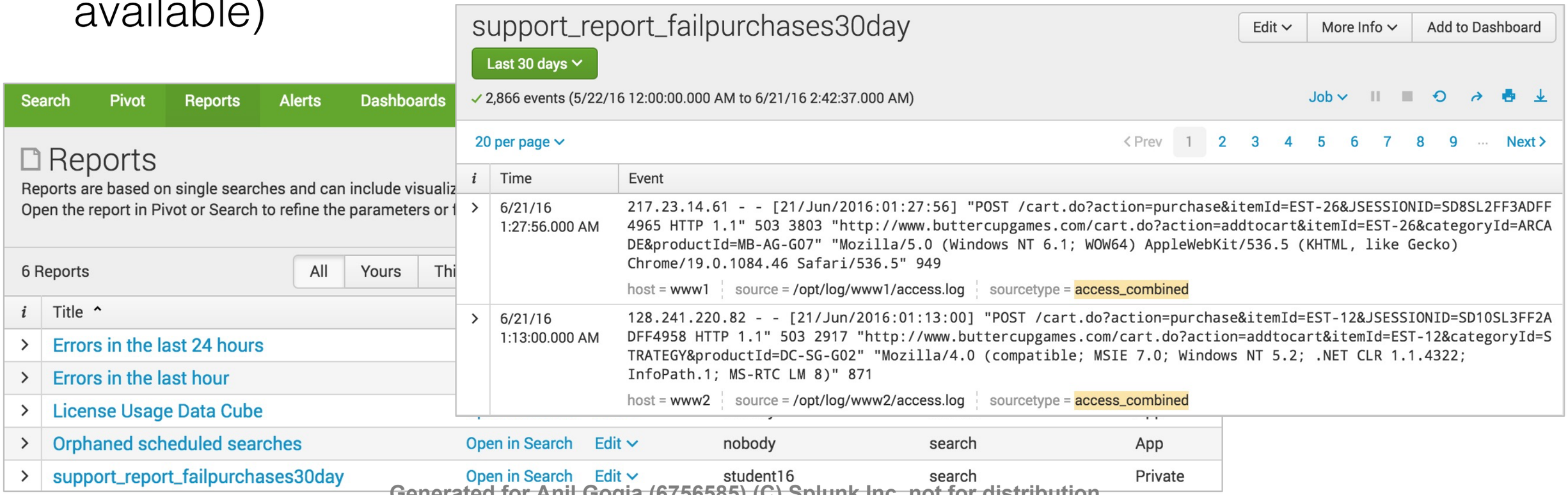


Additional Settings

Dialog buttons

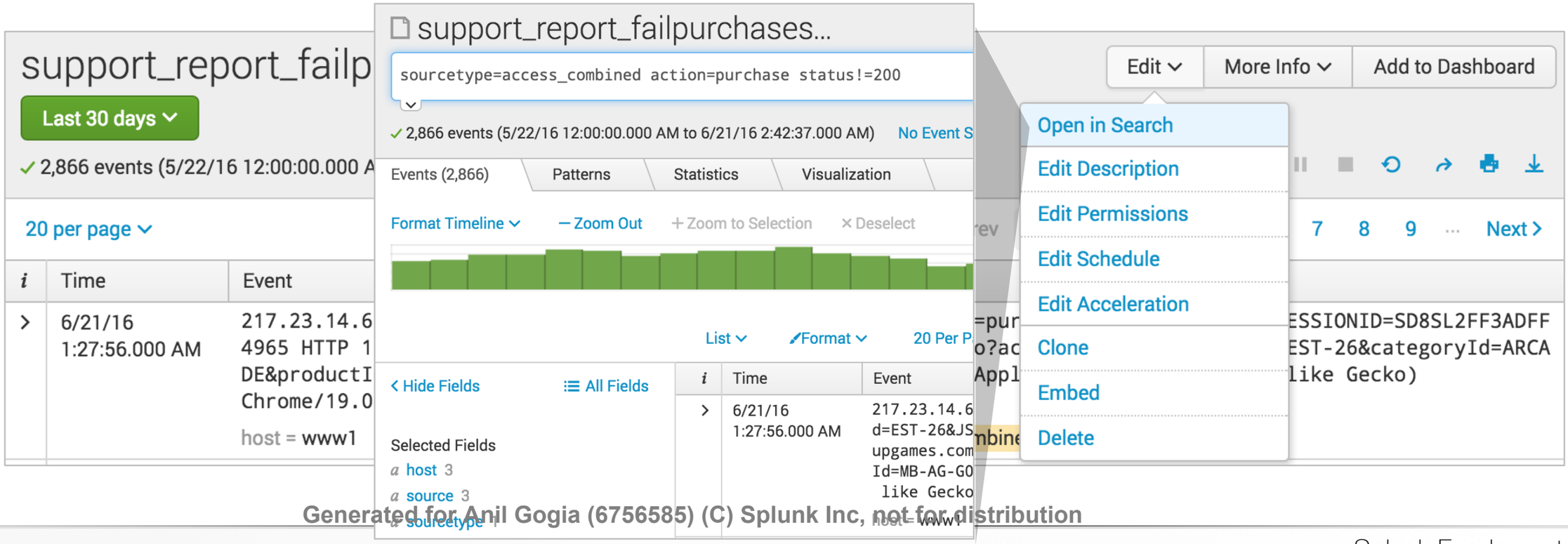
Running Reports

- Click **Reports**, then click the report title to run it
  - The report runs using the time range that was specified when it was saved
- Use the time range picker to change the time range of the report (if available)



Editing Reports

- To edit a report's underlying search, select **Edit > Open in Search**
  - You can then edit and re-save, not save, or save-as a new report
- You can also edit the description, permissions, schedule, and acceleration
- Additionally, you can clone or delete the report



Creating Tables and Visualizations

Three main methods to create tables and visualizations in Splunk are:

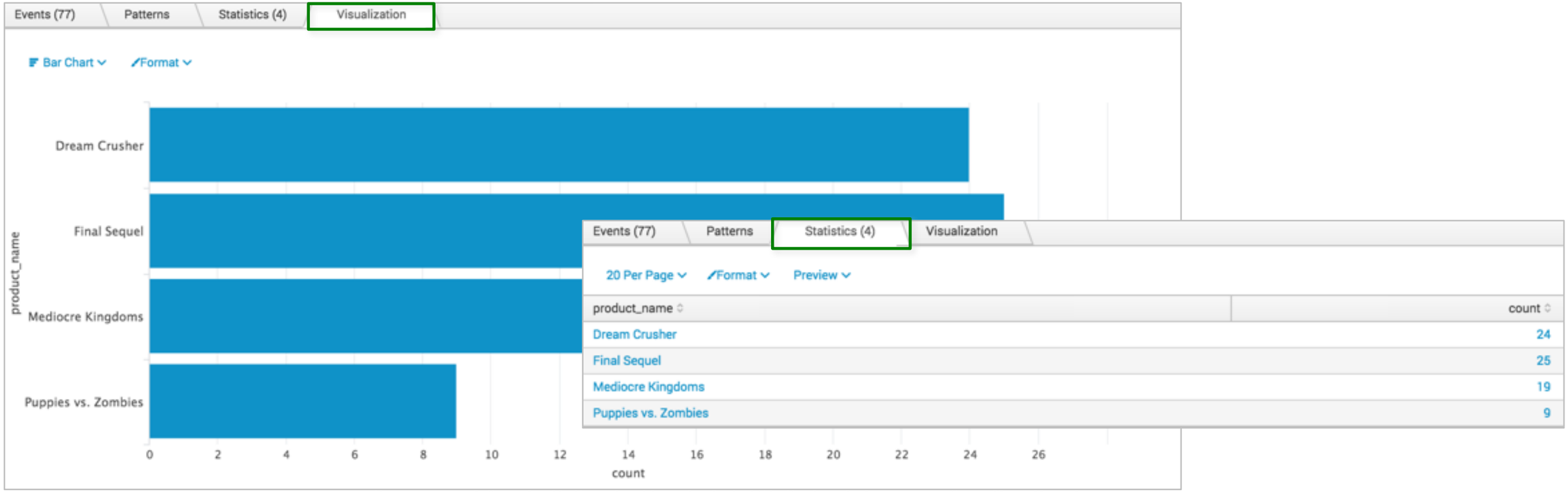
- Select a field from the fields sidebar and choose a report to run
- Use the Pivot interface
  - Start with a dataset

    or

  - Start with Instant Pivot
- Use the Splunk search language transforming commands in the Search bar
  - Transforming commands are discussed in the *Searching & Reporting with Splunk* course

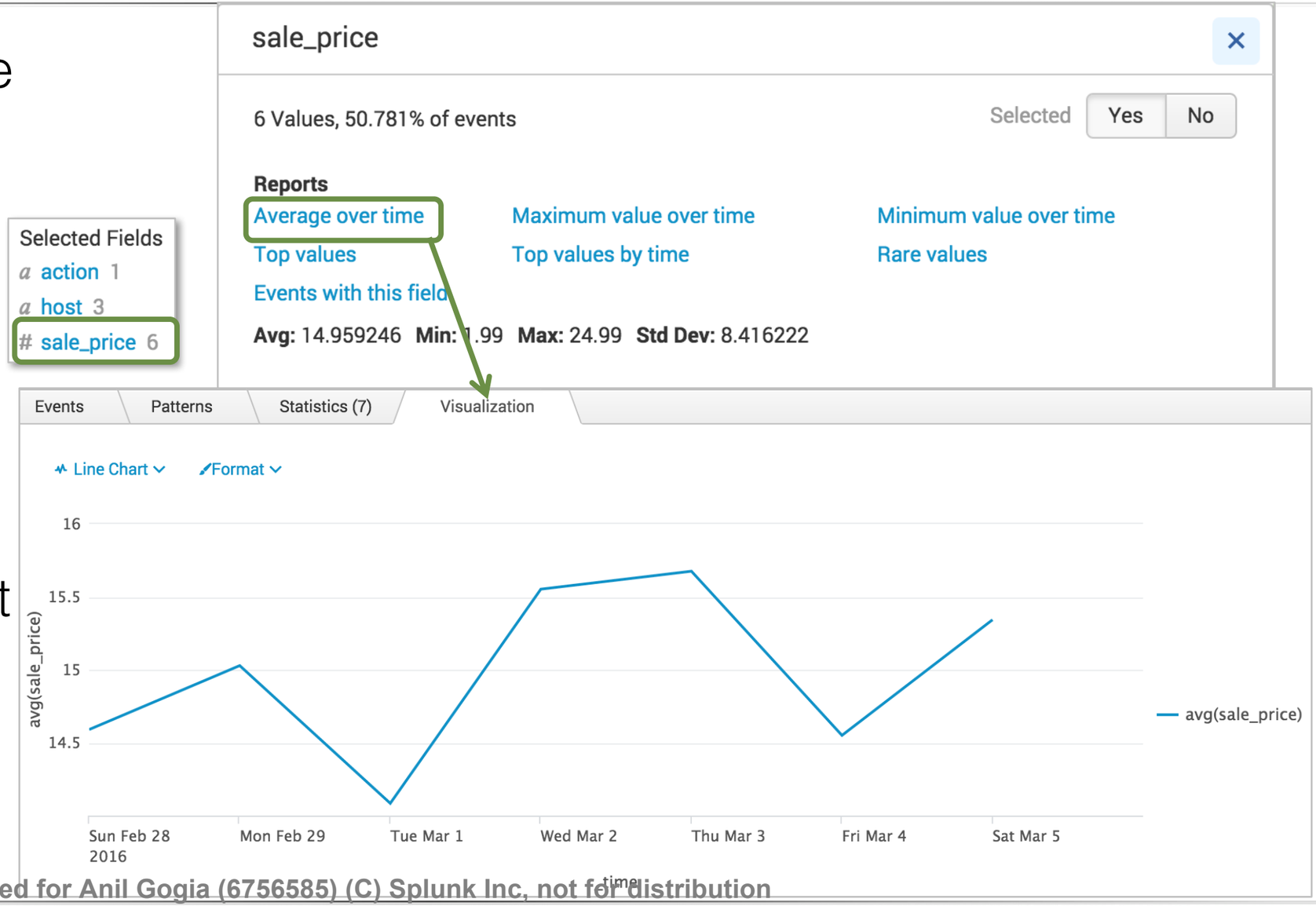
Tables and Visualizations

- Statistical reports leverage Splunk's built-in visualizations or table format
- These views give you insights into your organization's data



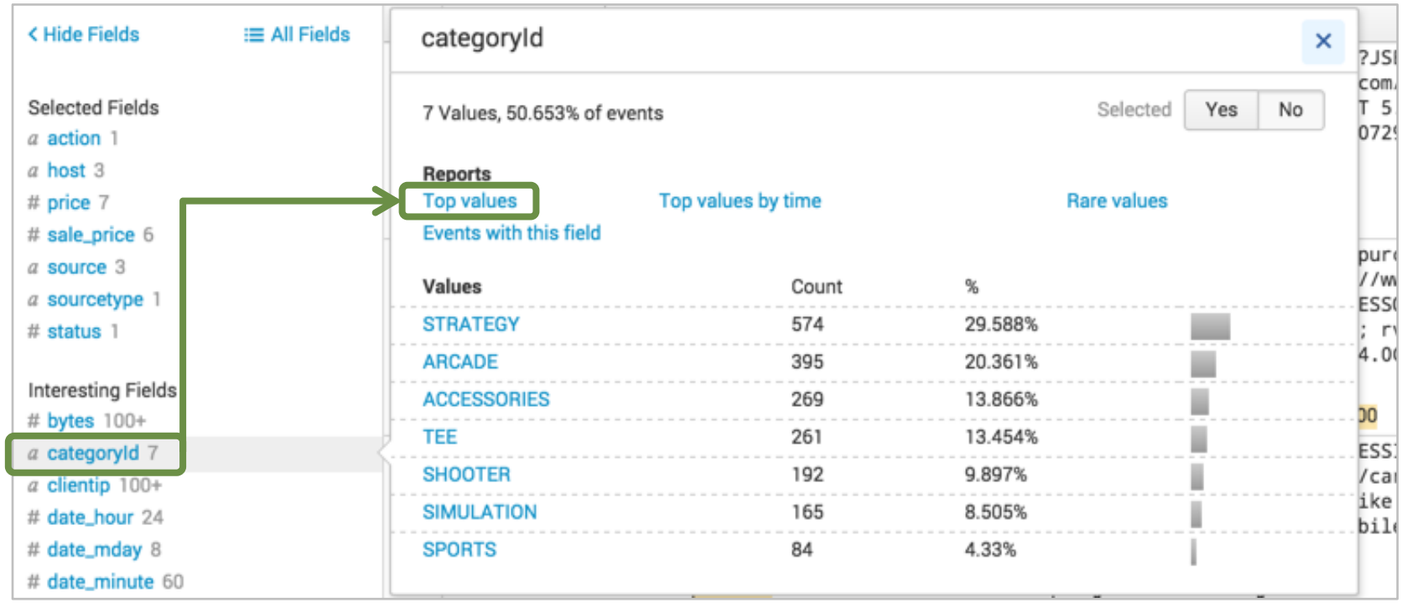
Create Reports From the Field Window

- Numeric fields: choose from six report types with mathematical functions, such as average, maximum value, and minimum value
- This example generates a report that shows the average over time
  - This is known as a **timechart**



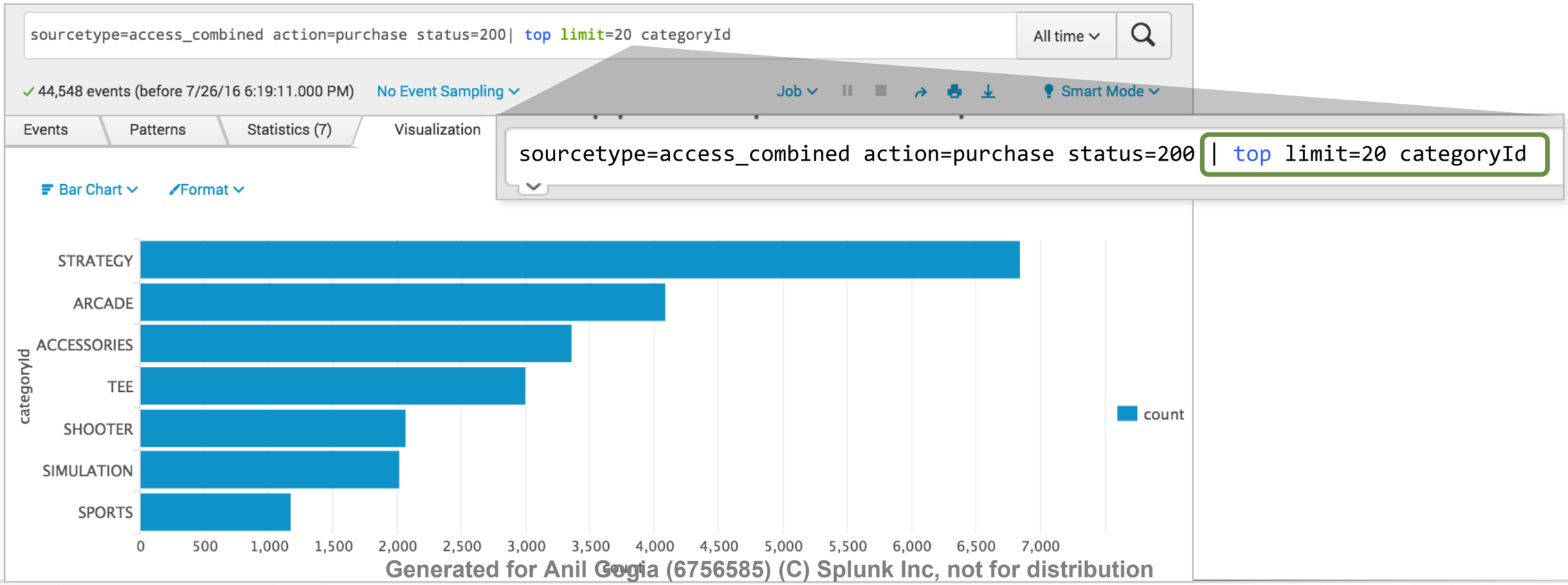
Create a Top Values Report

- For alphanumeric character fields, there are only 3 available reports
- In this example, we want a report that shows the top **categories** purchased
  - Basic search: `sourcetype=access_combined status=200 action=purchase`
  - Click the **categoryId** field
  - Click **Top values**



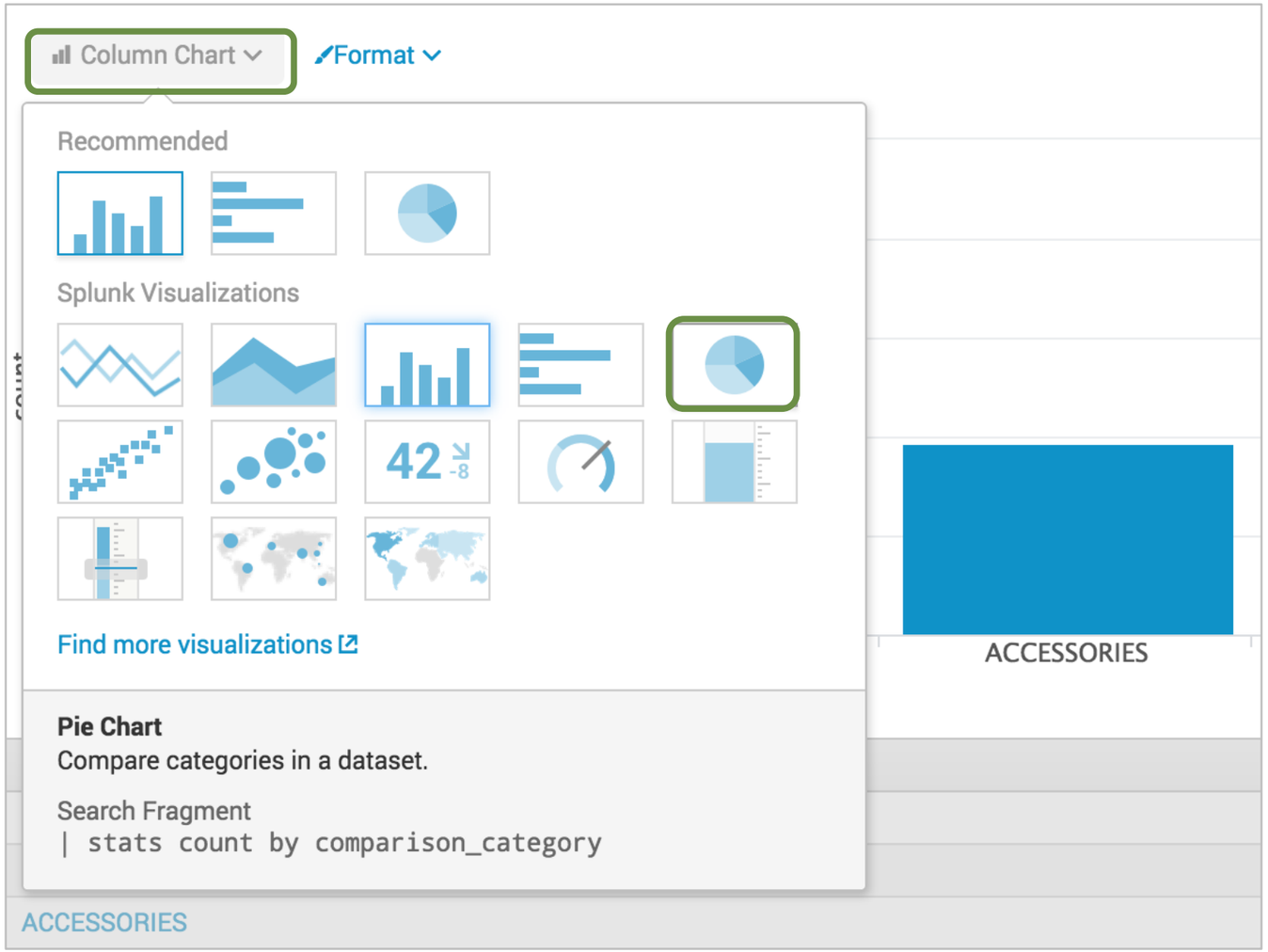
Create a Top Values Report (cont.)

- A | (pipe symbol) and the **`top`** command are added to the search string
- A bar chart is returned on the Visualizations tab, displaying the top categories purchased



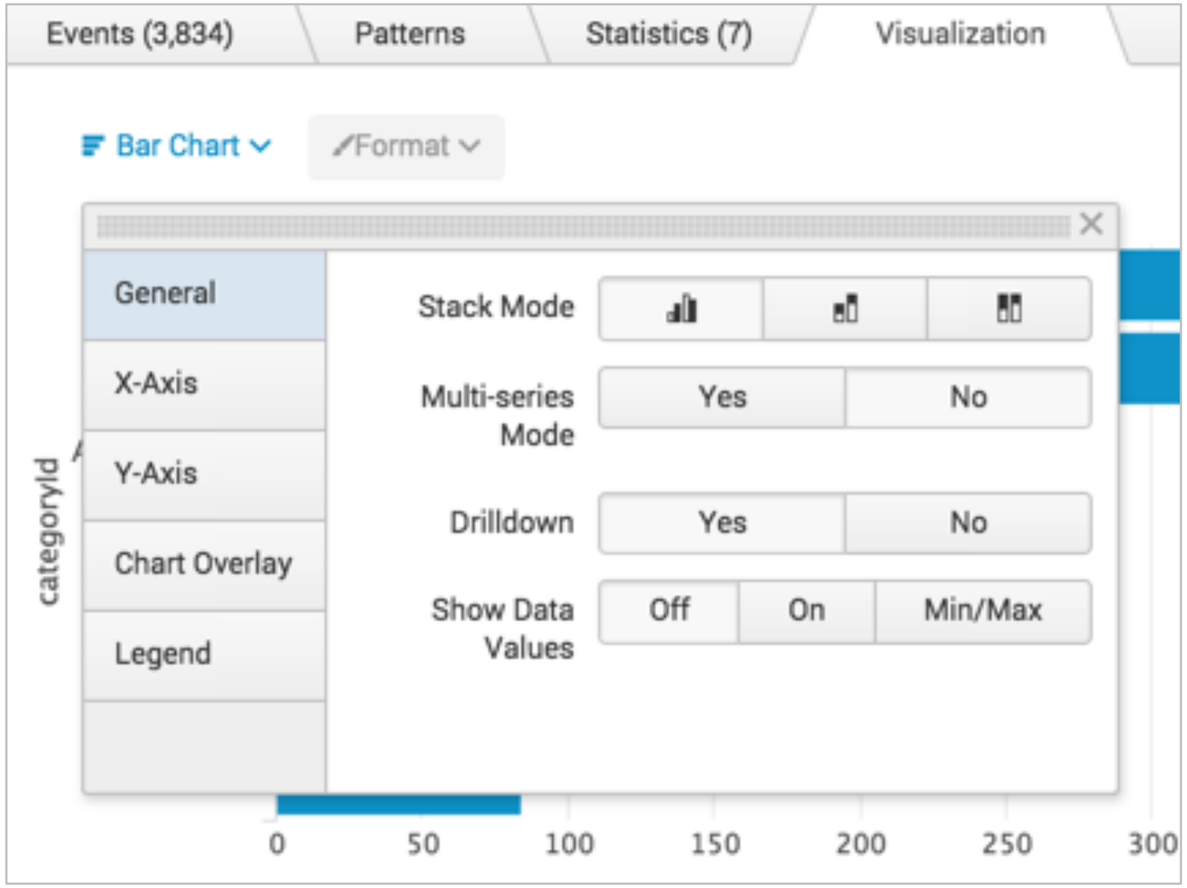
Change the Visualization

- Select a visualization from the visualization type dropdown menu
- In this example, the column chart is changed to a pie chart



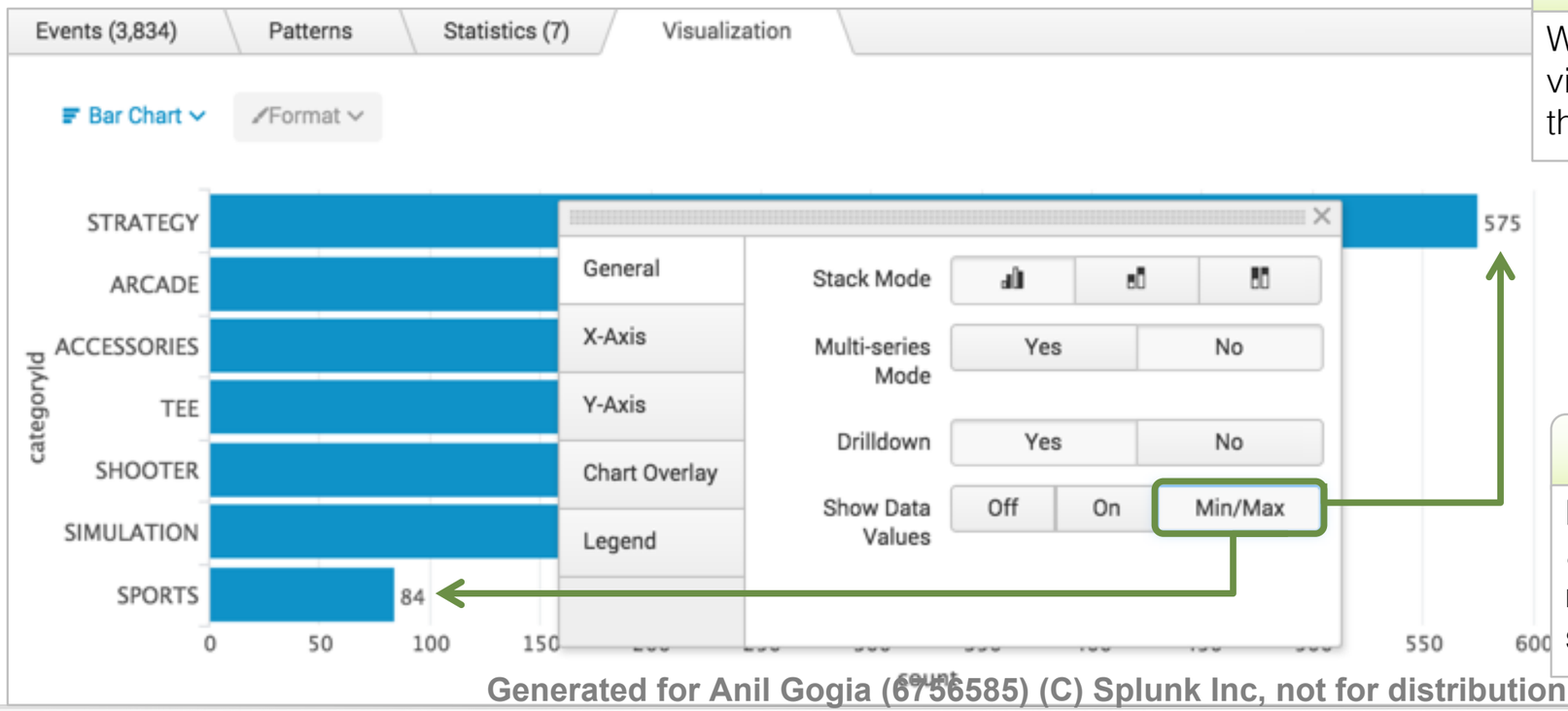
Change the Format

- The **Format** menu allows you to change formatting options
- For example, for bar and column charts:
  - The **General** tab allows you to change Stack, Multi-series, and Drilldown modes
  - The **X-Axis** and **Y-Axis** tabs allow you to change the axis labels and orientation
  - The **Legend** tab allows you to position the visualization legend as desired



Change the Format (cont.)

- **Show Data Values** determines whether to show data values in the visualization
  - If **Min/Max** is selected, data is only shown on the bars containing the minimum and maximum values



Note

When you make a change to the visualization settings – such as Min/Max – the visualization updates immediately.

Note

Learn more about modes and axes in the *Splunk Fundamentals 2* course. These modes require more sophisticated searches.

View as a Table

Switch to the **Statistics** tab to view the results as a table

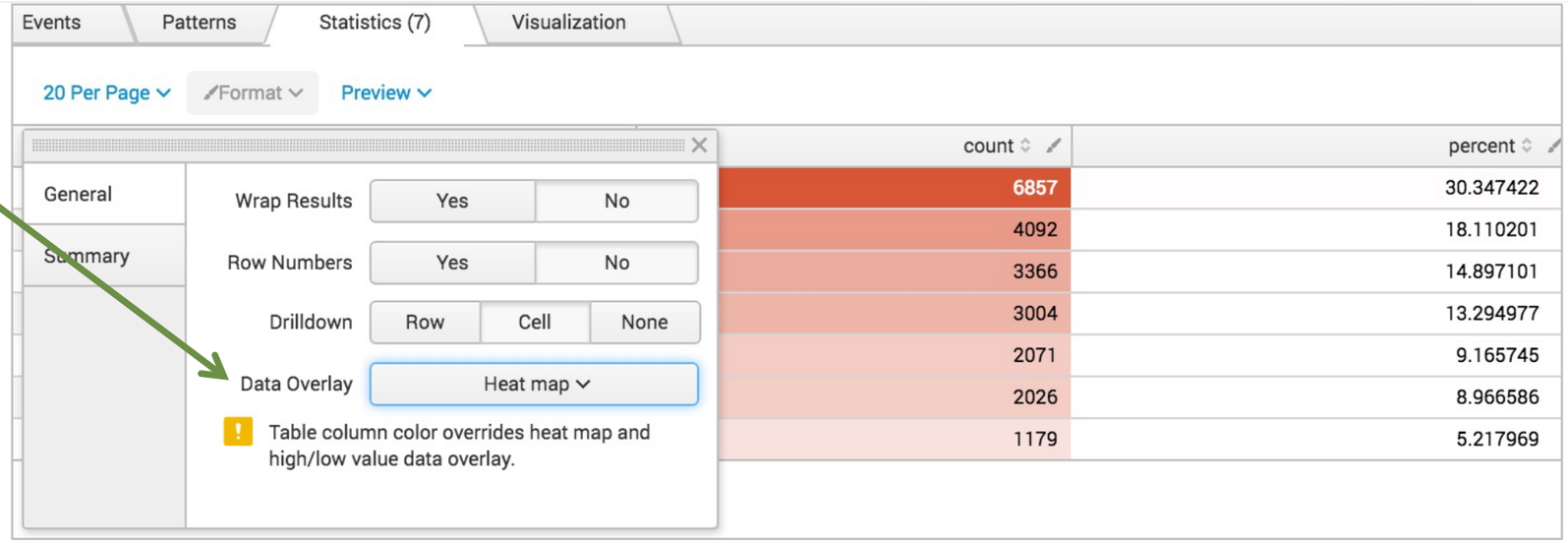
Events (1,966) | Patterns | Statistics (8) | Visualization

20 Per Page | Format | Preview

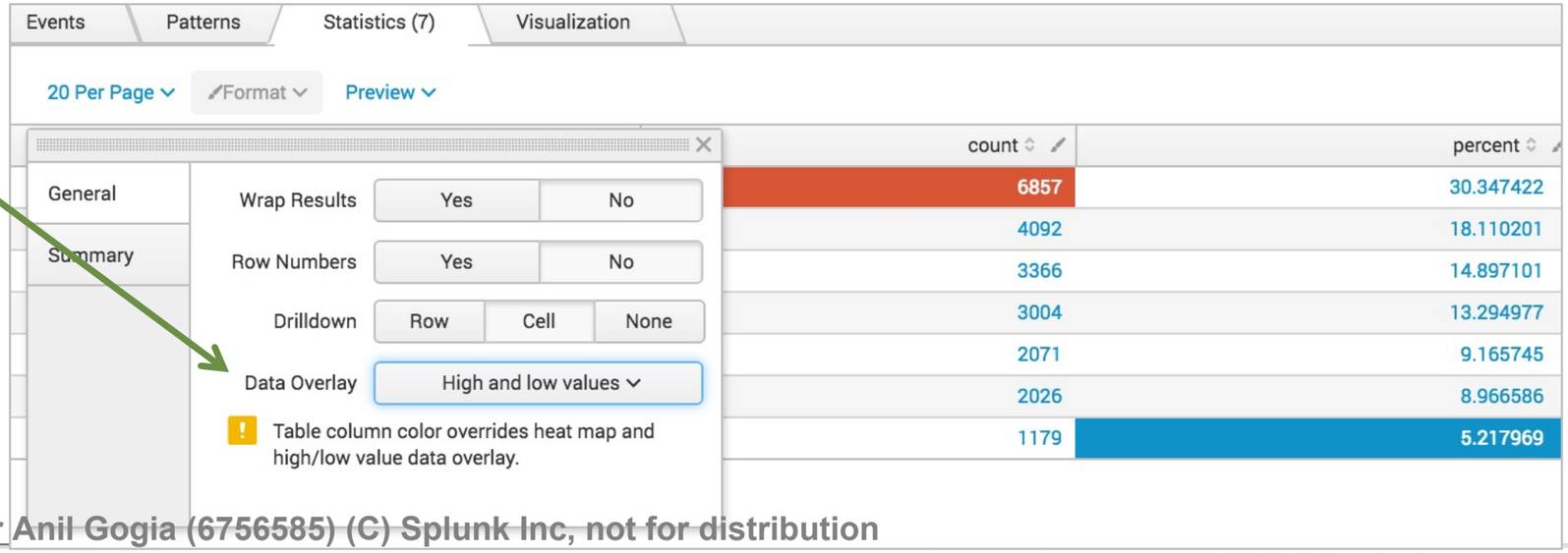
categoryId	count	percent
STRATEGY	315	23.899848
NULL	246	18.664643
ARCADE	194	14.719272
ACCESSORIES	155	11.760243
TEE	150	11.380880
SIMULATION	103	7.814871
SHOOTER	99	7.511381
SPORTS	56	4.248862

Statistics Overlay Format

- **Heat map** highlights outstanding values

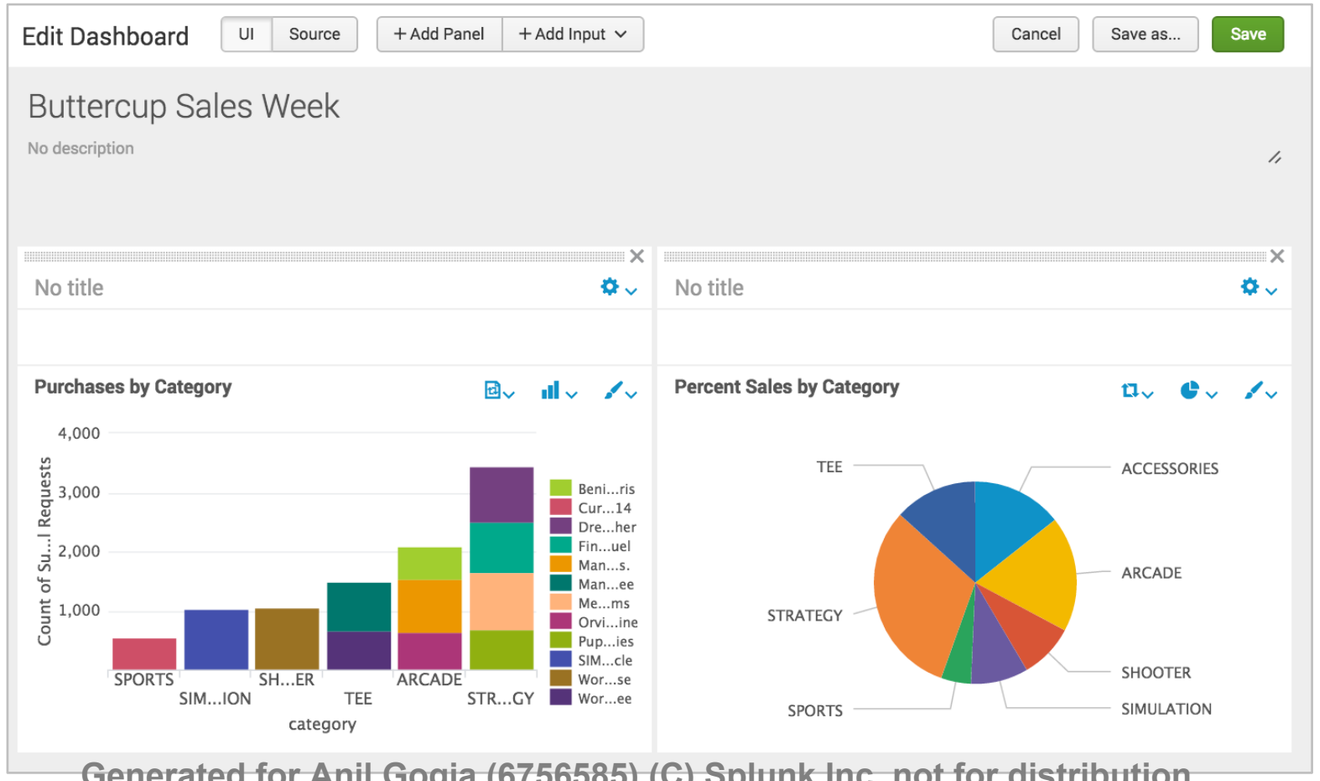


- **High and low values** highlights max and min of non zero values



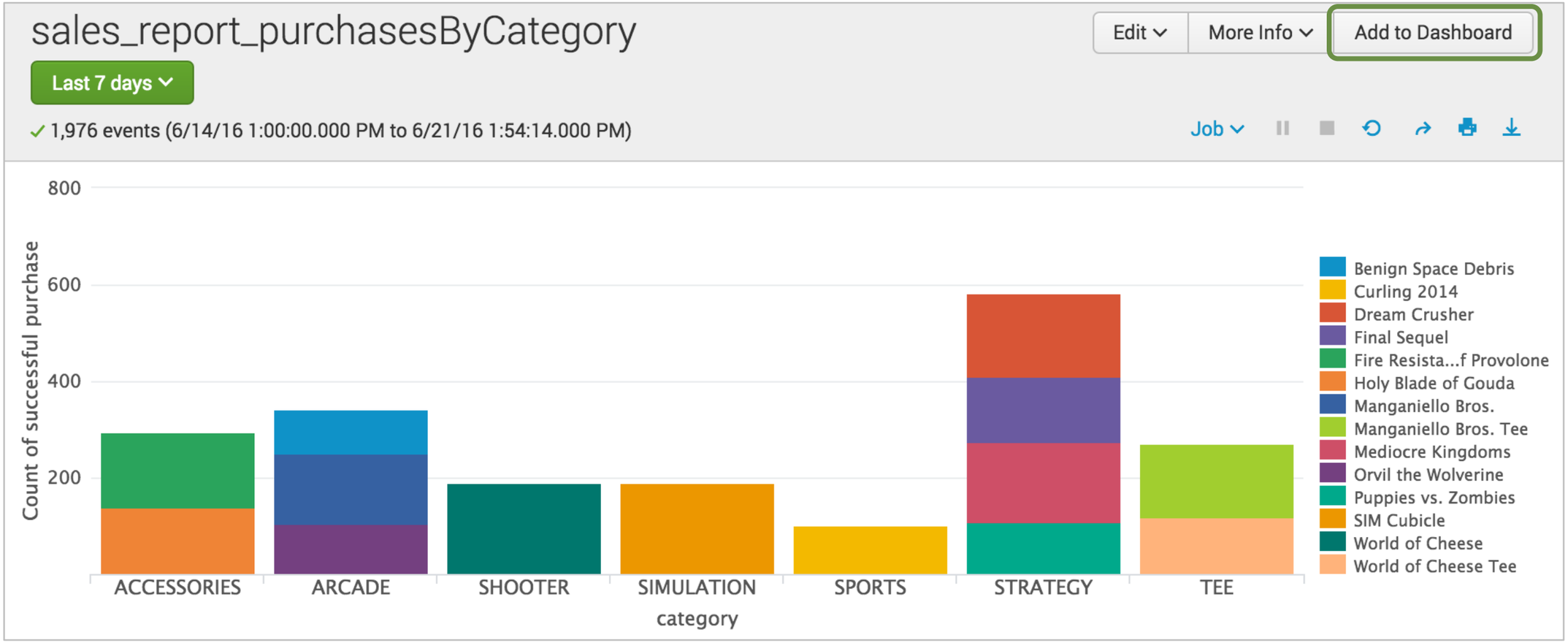
What Is a Dashboard?

- A dashboard consists of one or more panels displaying data visually in a useful way – such as events, tables, or charts
- A report or a pivot can be used to create a panel on a dashboard



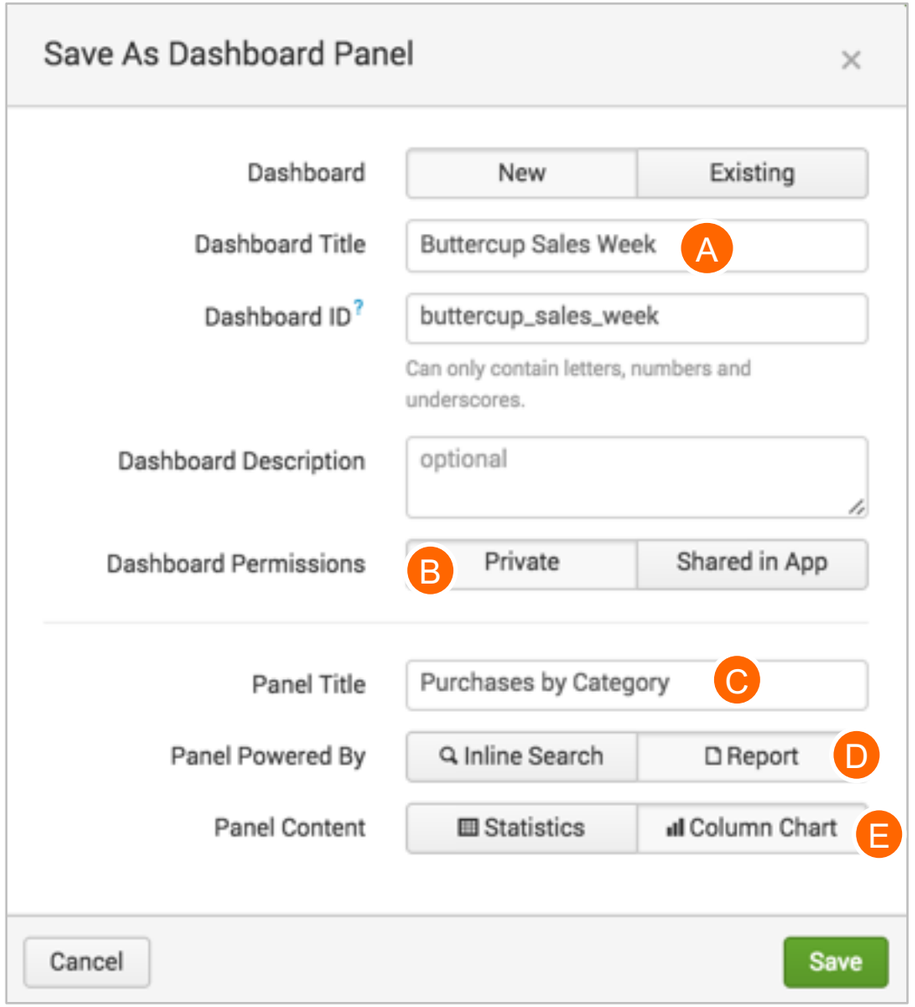
Adding a Report to a Dashboard

In the report, click **Add to Dashboard** to begin



Adding a Report to a Dashboard (cont.)

- A Name the dashboard and optionally provide a description
- B Change the permissions (use Private until tested)
- C Enter a meaningful title for the panel
- D For **Panel Powered By,** click **Report**
- E For the **Panel Content**, select **Statistics** to display as a table, or the *visualization type* (in this case, a **Column Chart**)

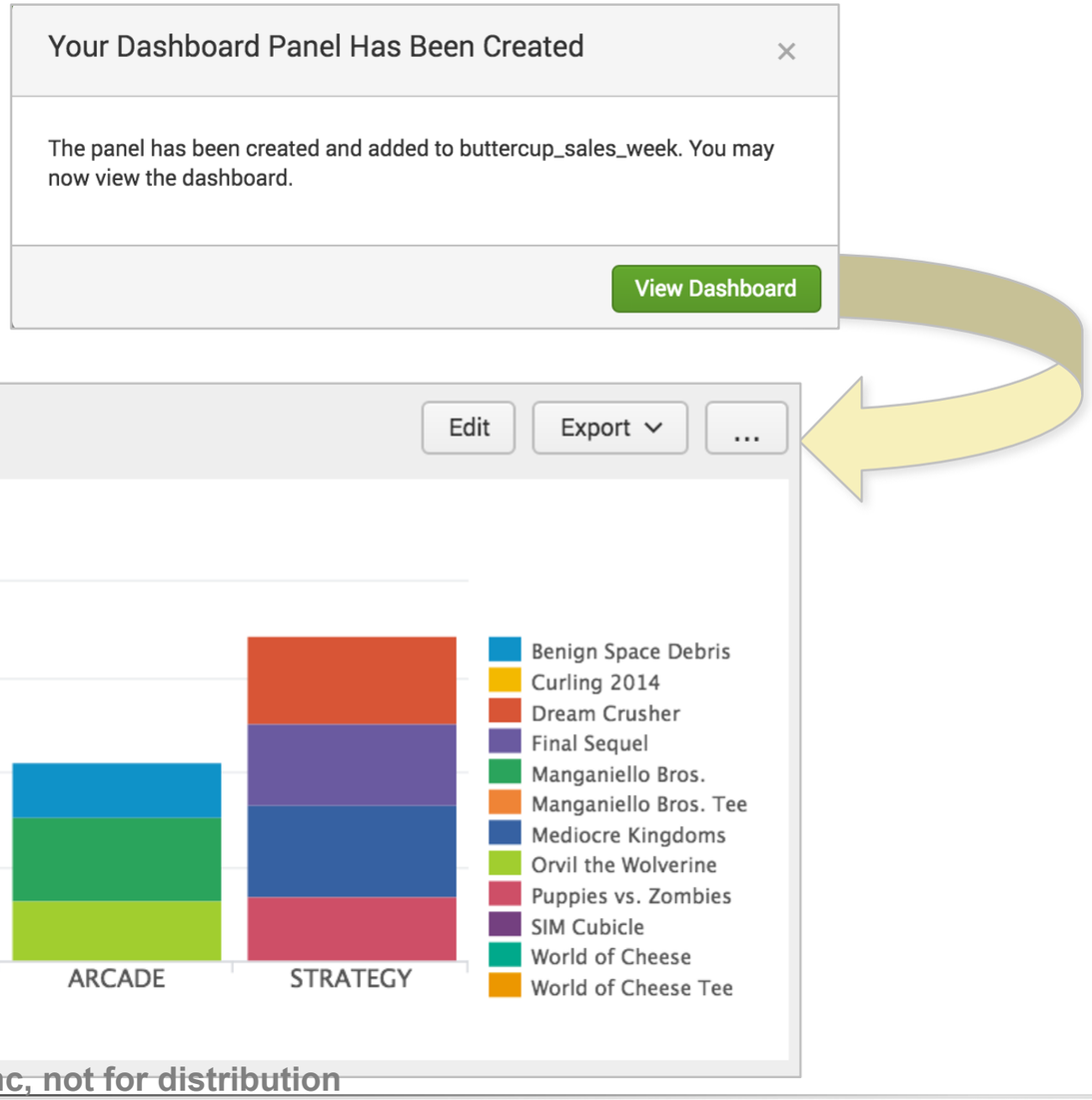


Note

The Dashboard ID is automatically populated with a unique value used by Splunk and should not be changed.

Adding a Report to a Dashboard (cont.)

After it is saved, you can view the dashboard immediately, or select the dashboard from the **Dashboards** view

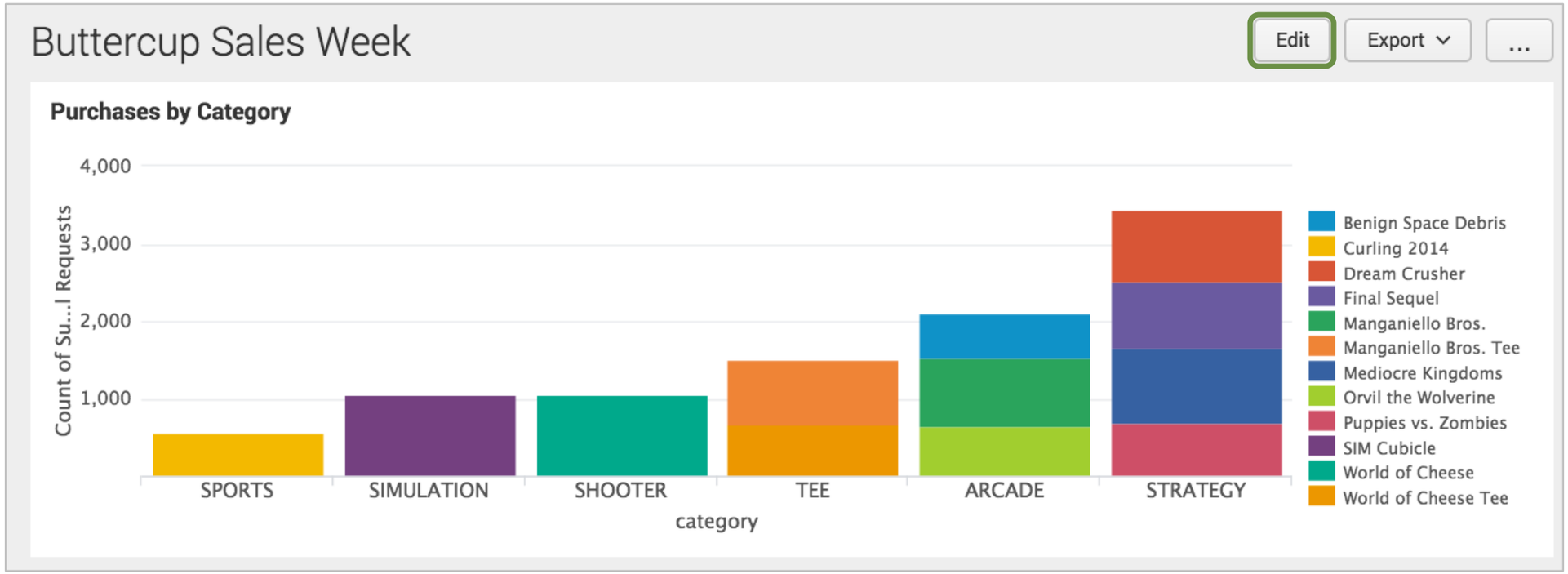


Why Create Panels from Reports?

- It is efficient to create most dashboard panels based on reports because
  - A single report can be used across different dashboards
  - This links the report definition to the dashboard
- Any change to the underlying report will affect every dashboard panel that utilizes that report

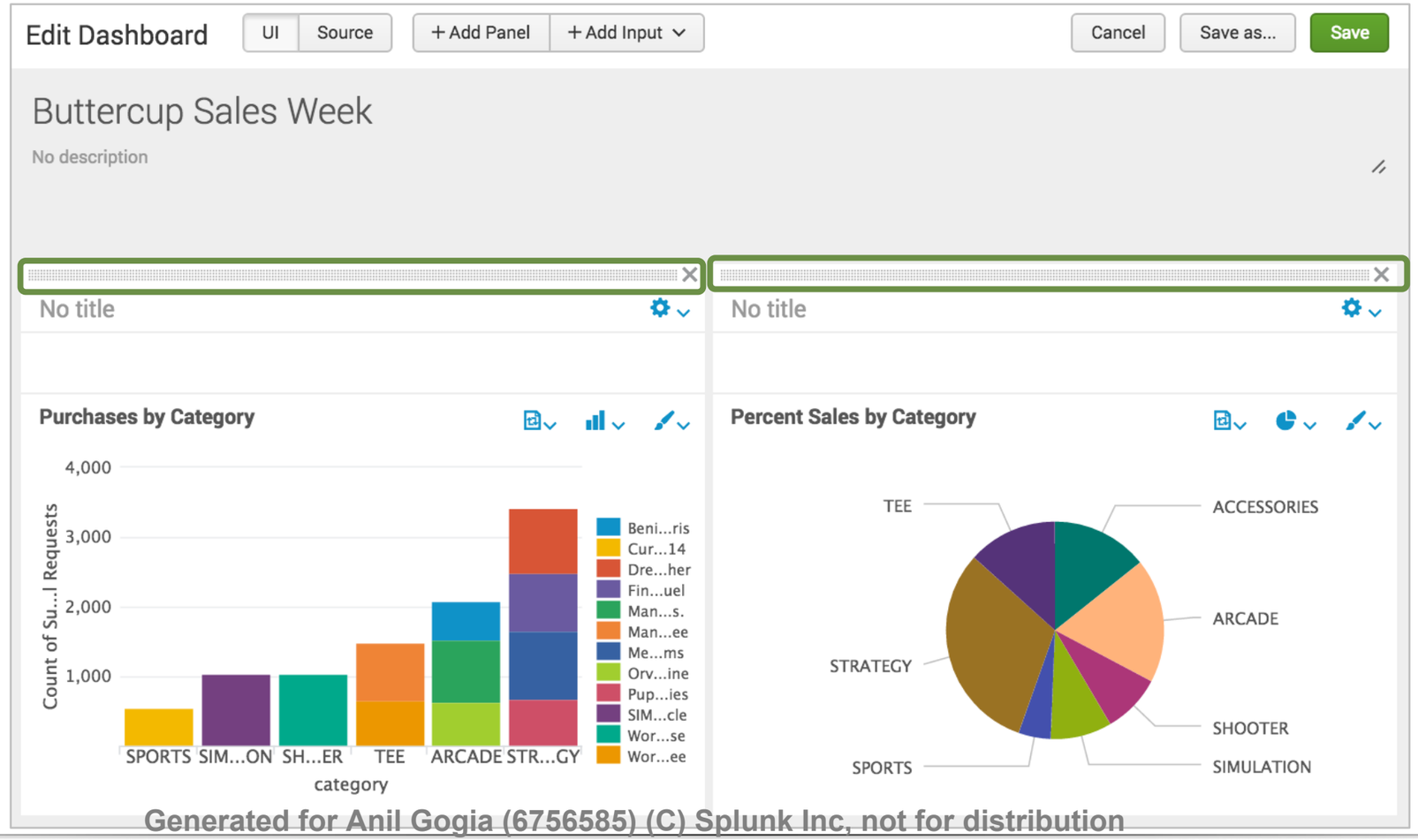
Editing Panels

- After saving the panel, a window appears from which you can view the updated dashboard
- Click **Edit** to customize the dashboard



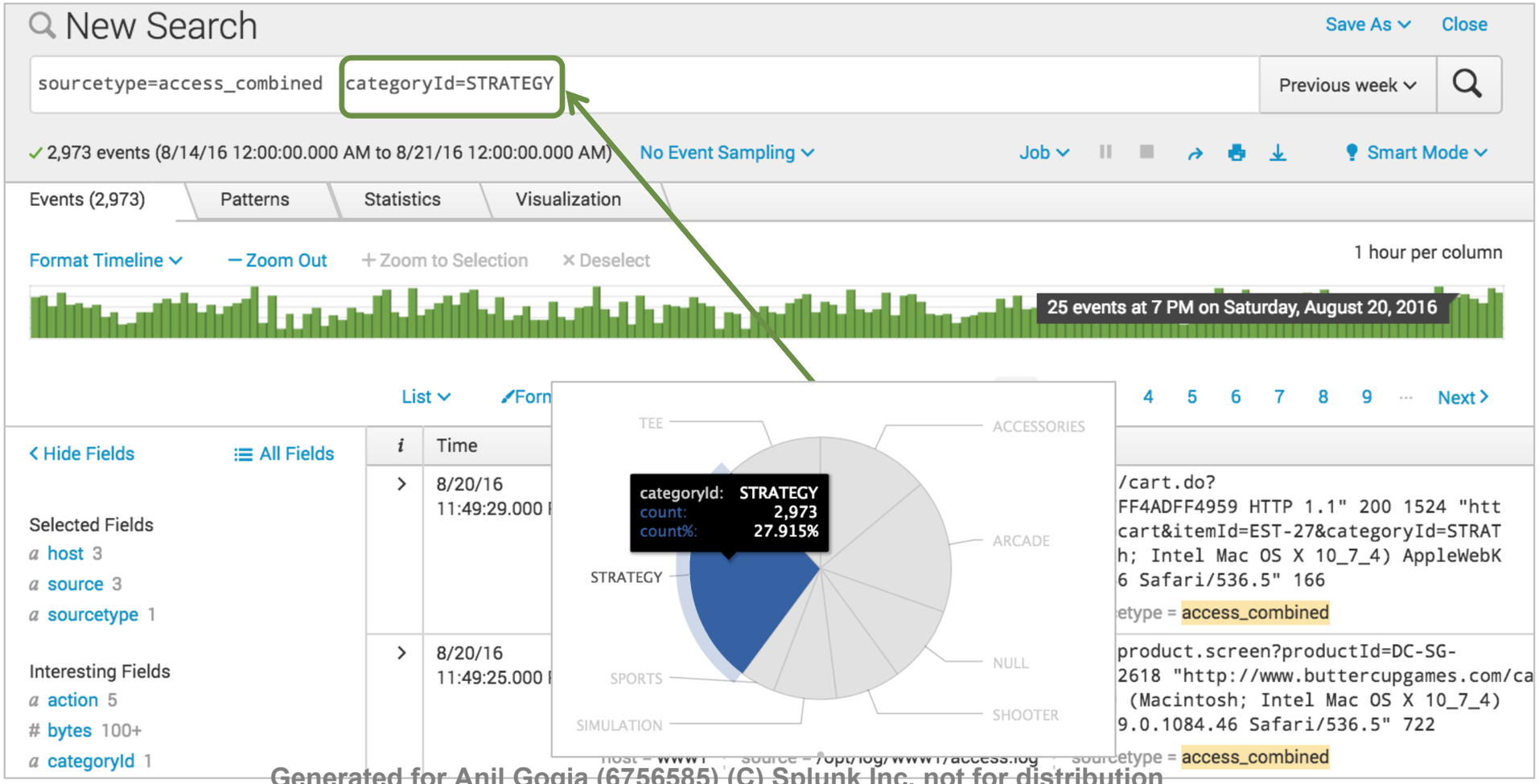
Editing Panel Layout

Click on the dotted bar on a panel to drag the panel to a new location



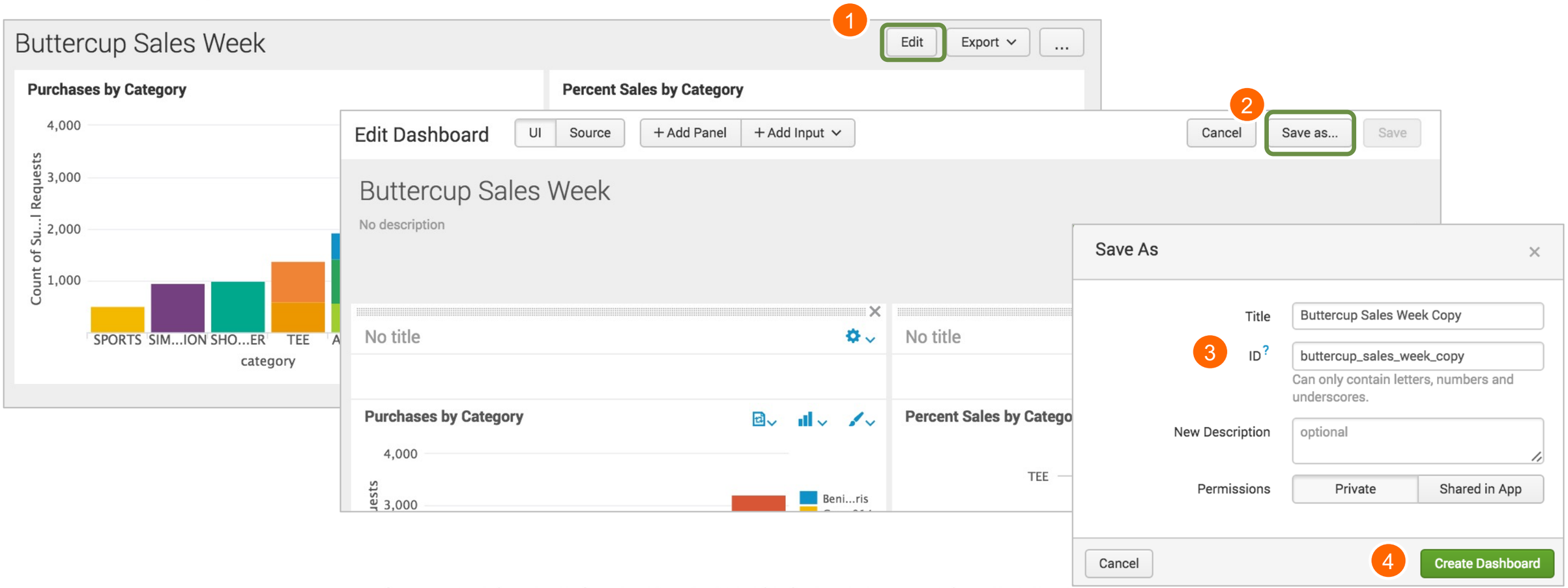
Drill Down from Visualization to Search

Click an object in a chart or table to see its underlying events in Search view



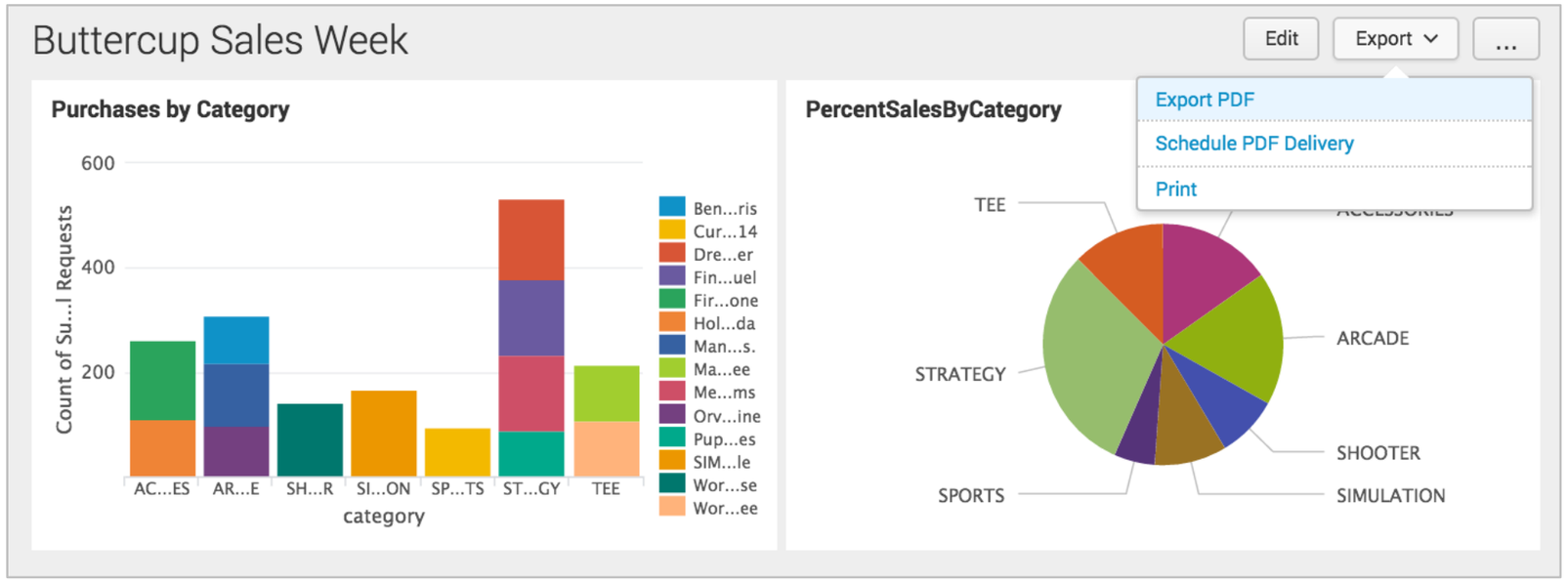
Clone a Dashboard

- To clone a dashboard, click **Edit** – and then **Save as…**
  - Change the **Title** as desired, and then click **Create Dashboard**



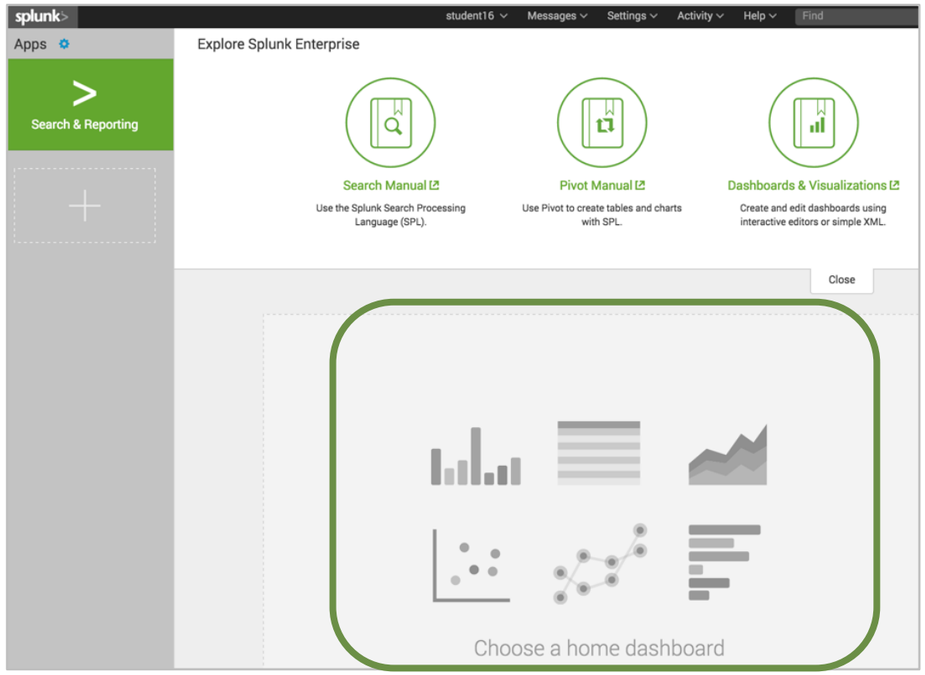
Export a Dashboard (cont.)

- Without the add-on, dashboards can be exported as PDF
  - They can also be printed



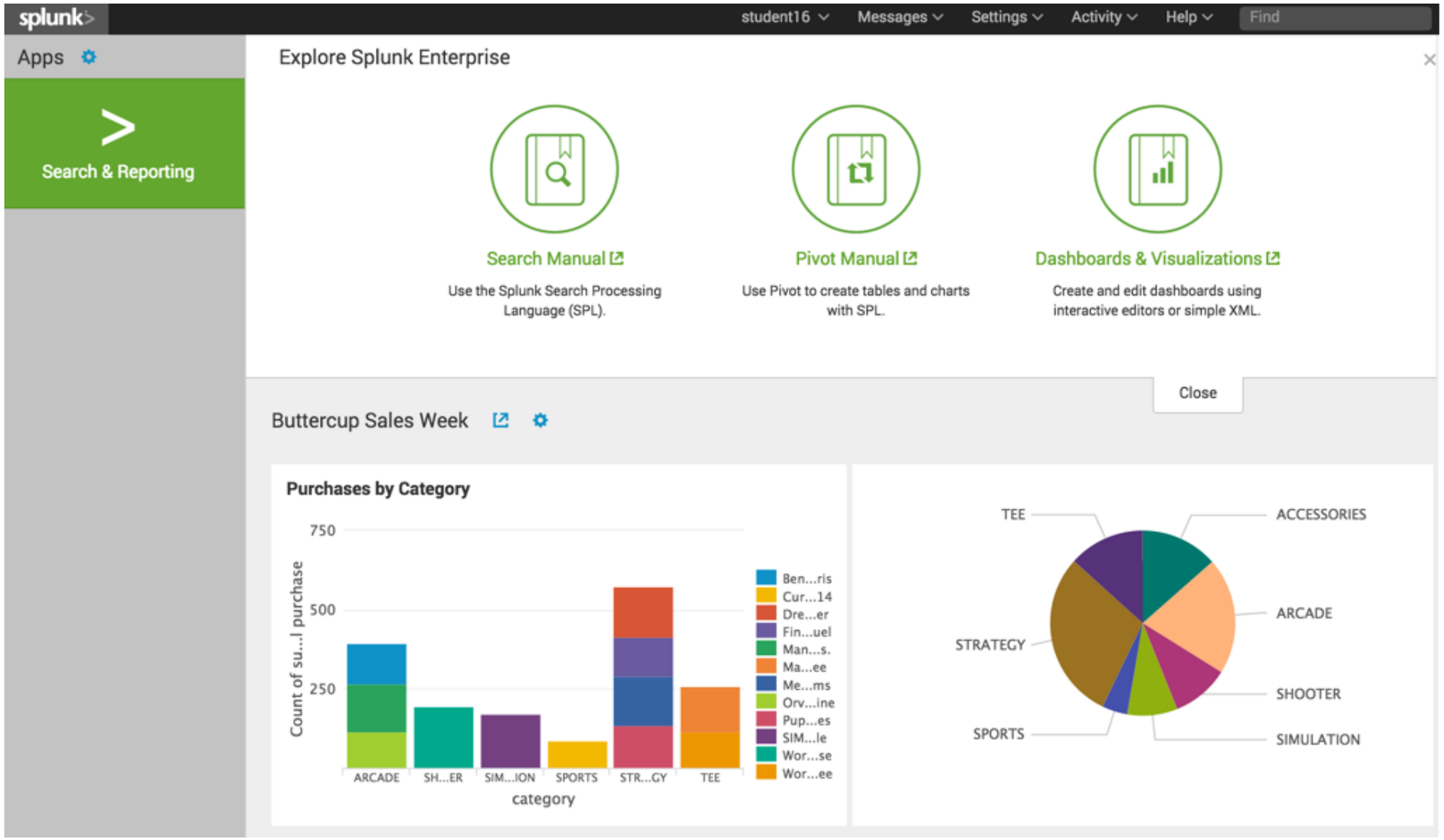
Make a Default Dashboard

- Set a dashboard to appear by default in the bottom panel of your home view
- From Home, click **Choose a home dashboard**



View Your Default Dashboard

After you've set a dashboard as default, your home view may look like this:



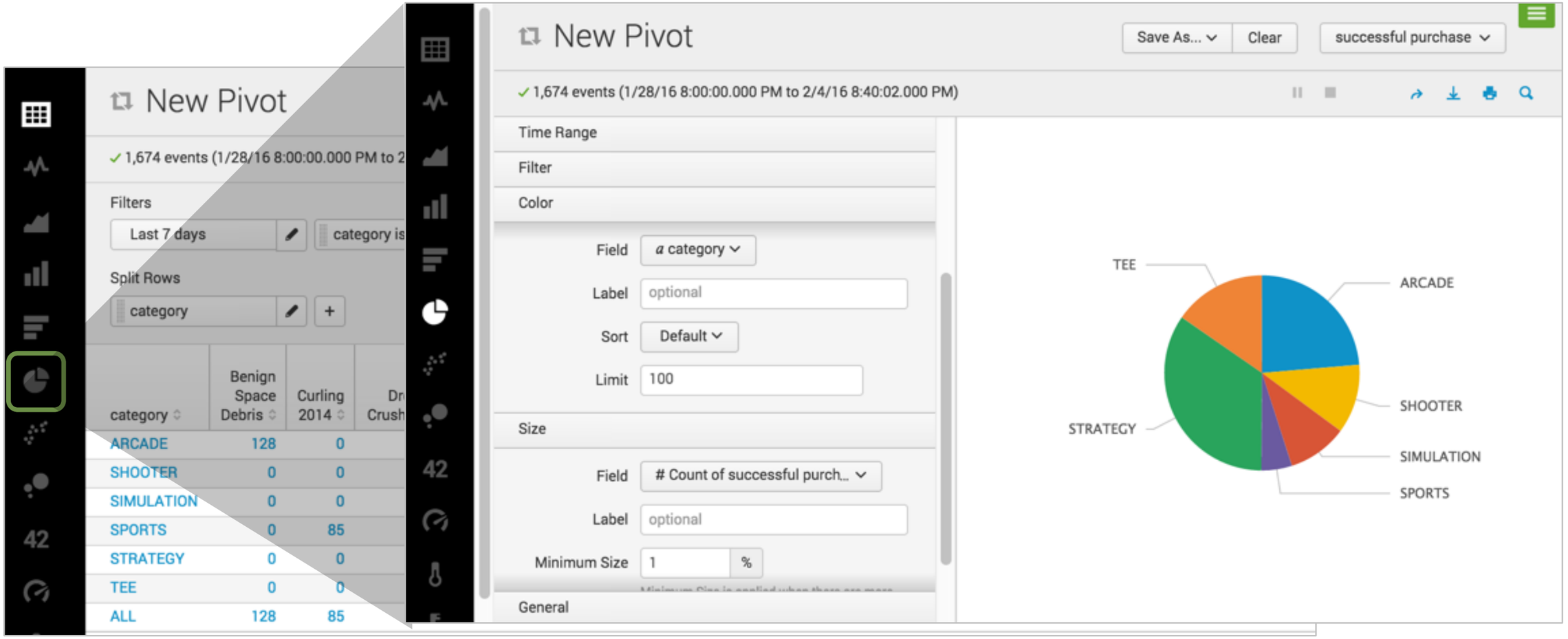
Module 11:
Using Pivot

Objectives

- Describe pivot
- Understand the relationship between the data model and the pivot
- Select a data model object
- Create a pivot report
- Use instant pivot to create a report

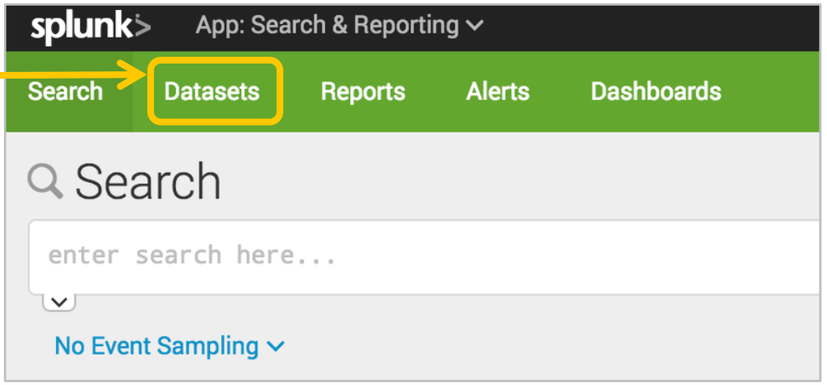
Completed Pivot

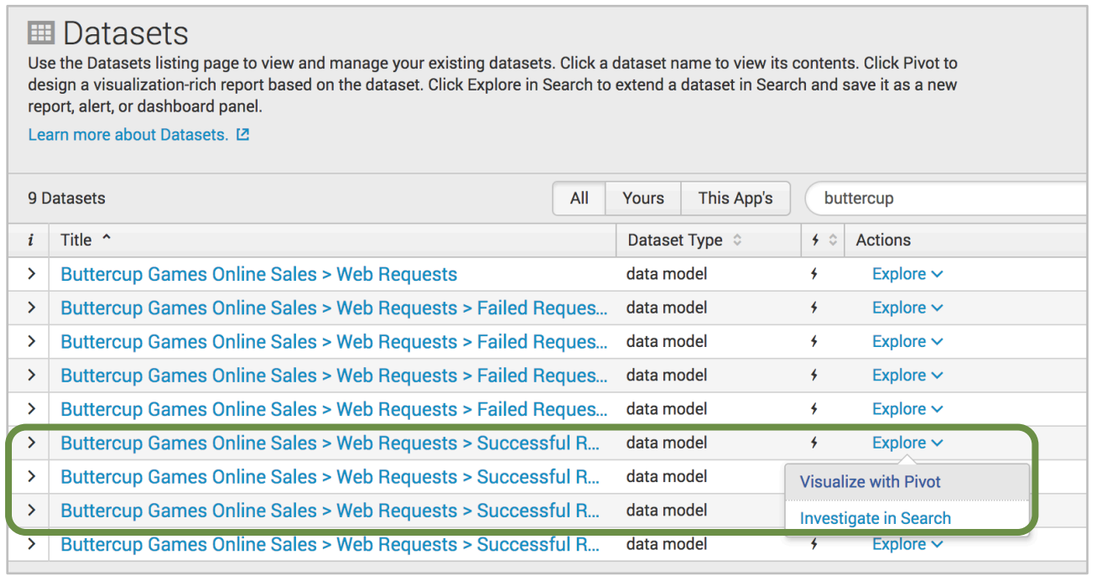
Pivot is a quick way to design visualizations of data. Let's see how.



Selecting a Dataset

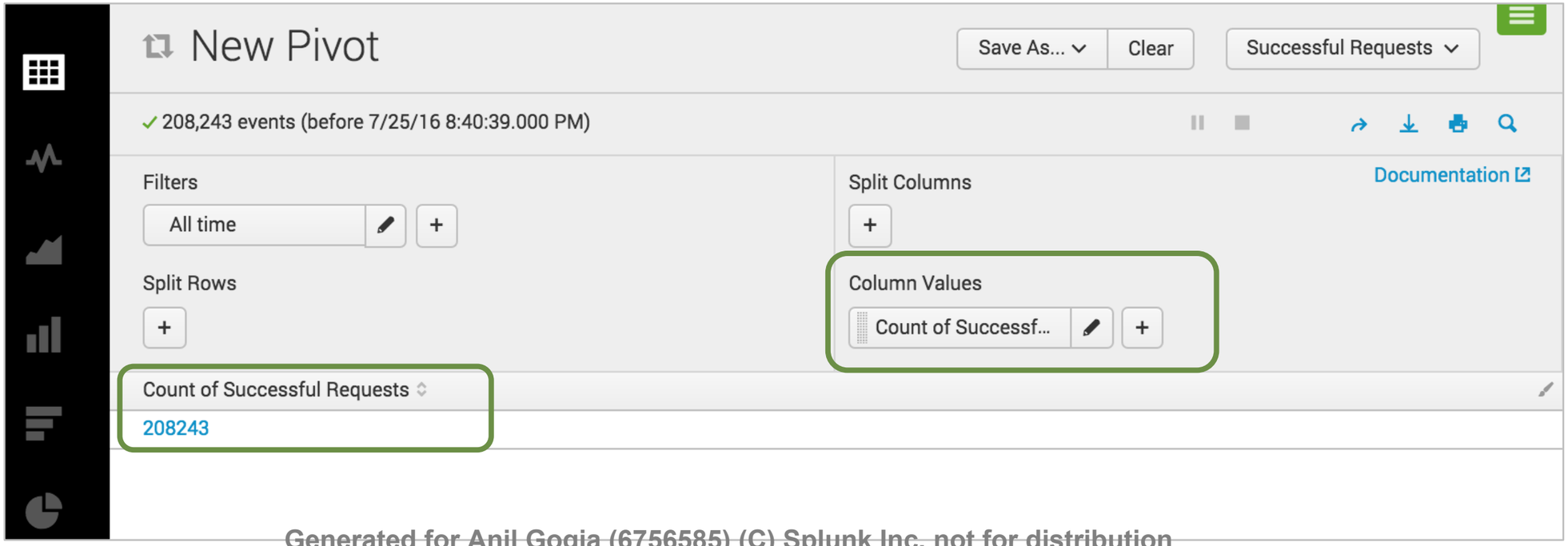
1. From the Search & Reporting app, select the **Datasets** tab
   - This displays a list of available lookup table files ("lookups") and data models
   - Each lookup and data model represent a specific category of data
     - Prebuilt lookups and data models make it easier to interact with your data
2. Click **Explore > Visualize with Pivot**





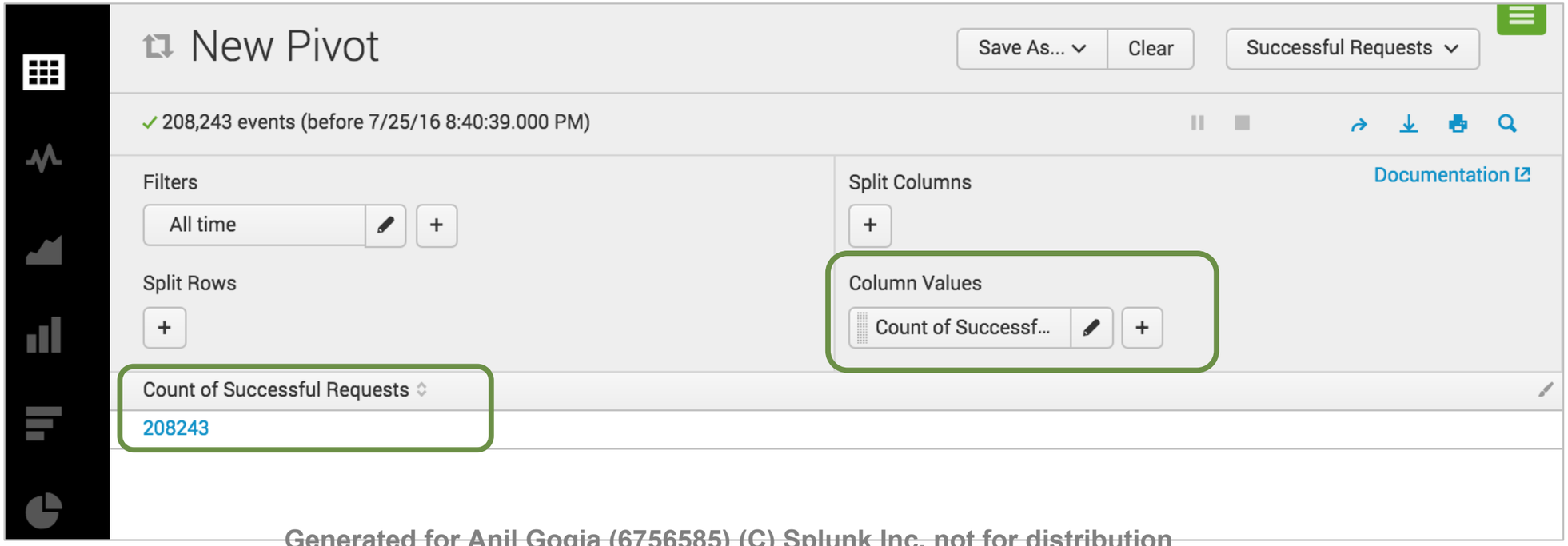
Open in Pivot

- The Pivot automatically populates with a count of events for the selected object
- In this example, it shows all successful purchase requests for all time



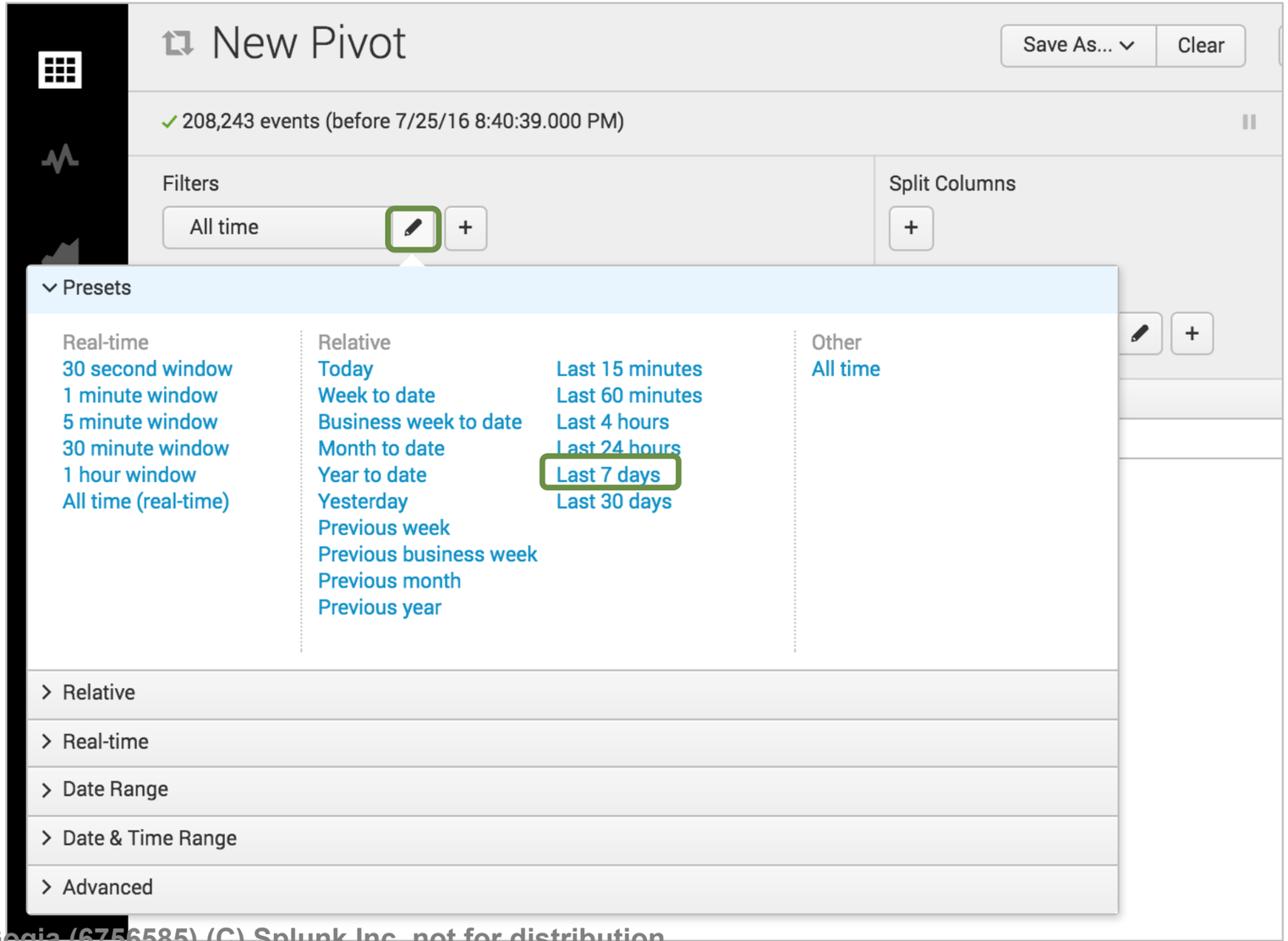
Open in Pivot

- The Pivot automatically populates with a count of events for the selected object
- In this example, it shows all successful purchase requests for all time



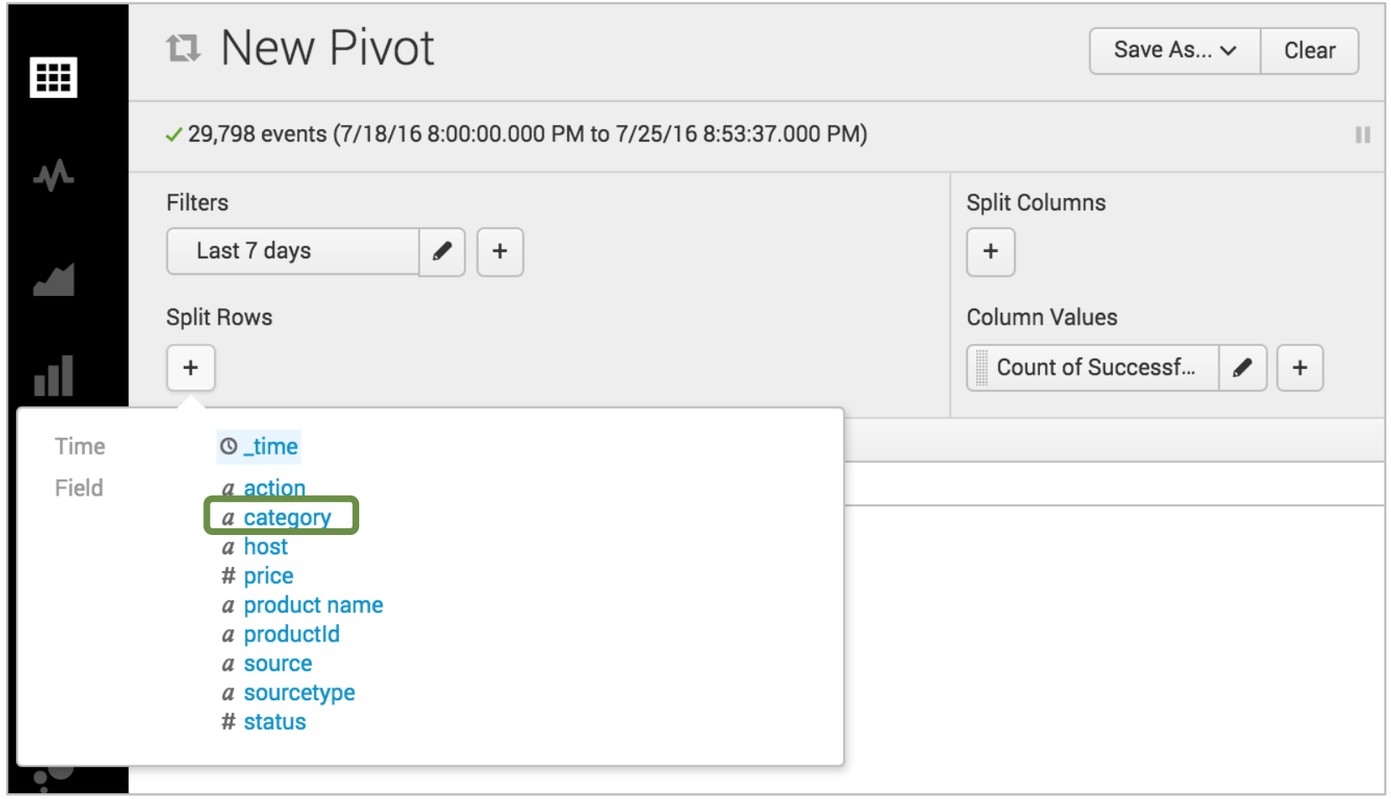
Select a Time Range

- The default is **All time**
- Click the pencil icon to select the desired time range
- The pivot runs immediately upon selecting the new time range



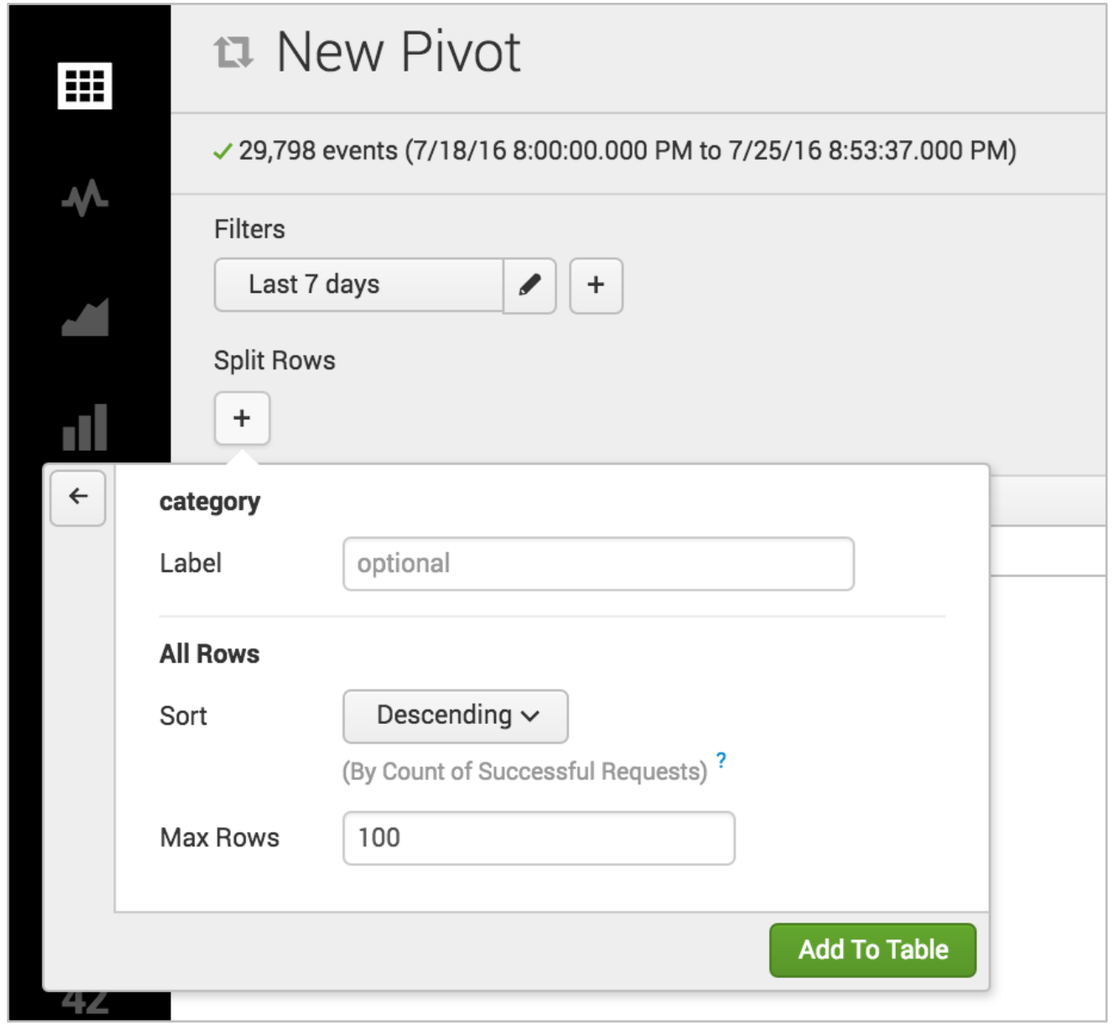
Split Rows

- Click + under **Split Rows** for a list of available attributes to populate the rows
- In this example, the rows are split by the **category** attribute, which will list
  - Each game category on a separate row
  - A count of successful requests for each game category

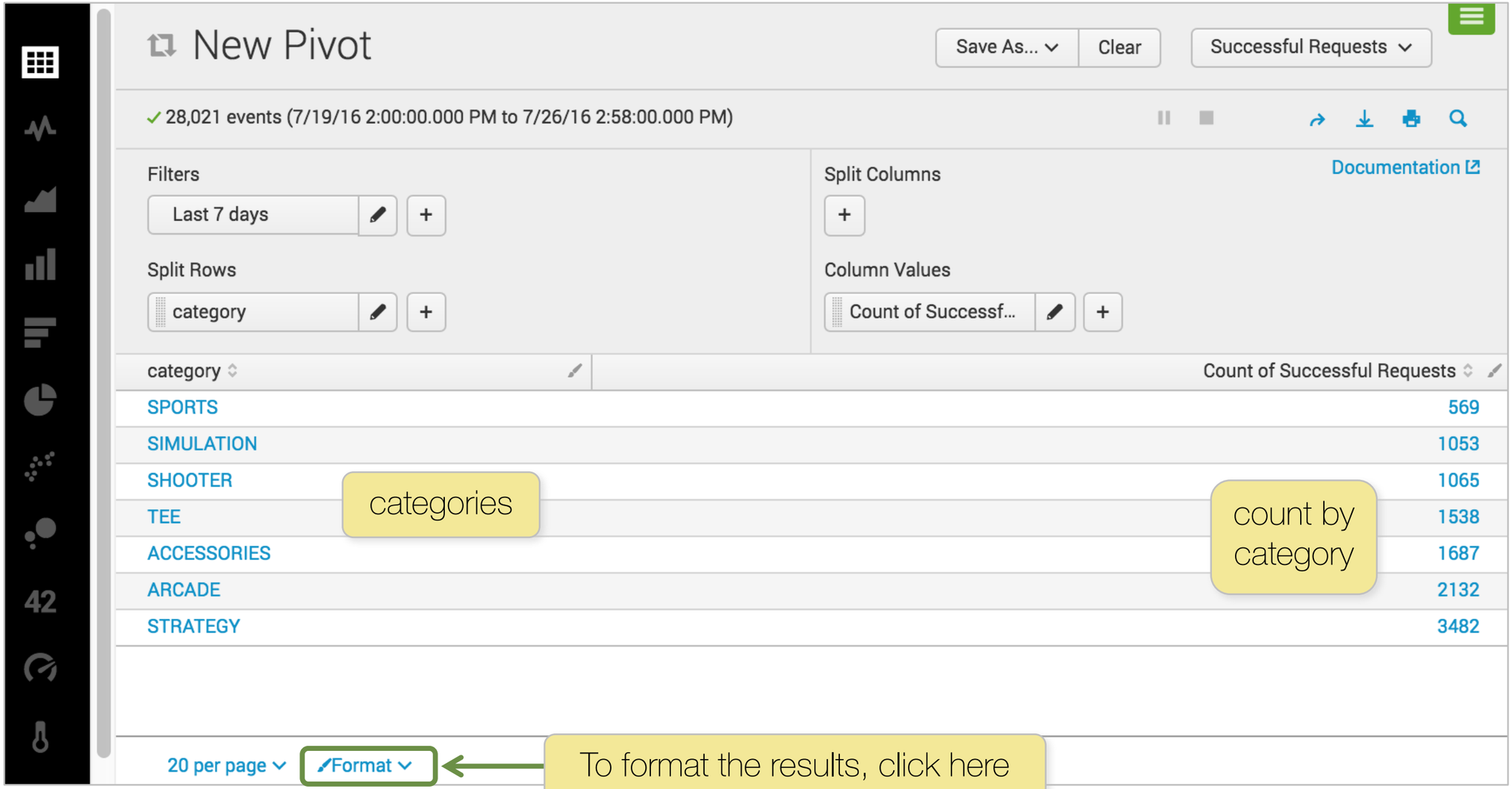


Split Rows (cont.)

- Once selected, you can:
  - Modify the label
  - Change the sort order
    - **Default** – sorts by the field value in ascending order
    - **Ascending** - sorts by the count in ascending order
    - **Descending** – sorts by the count in descending order
  - Define maximum # of rows to display
- Click **Add to Table** to view the results

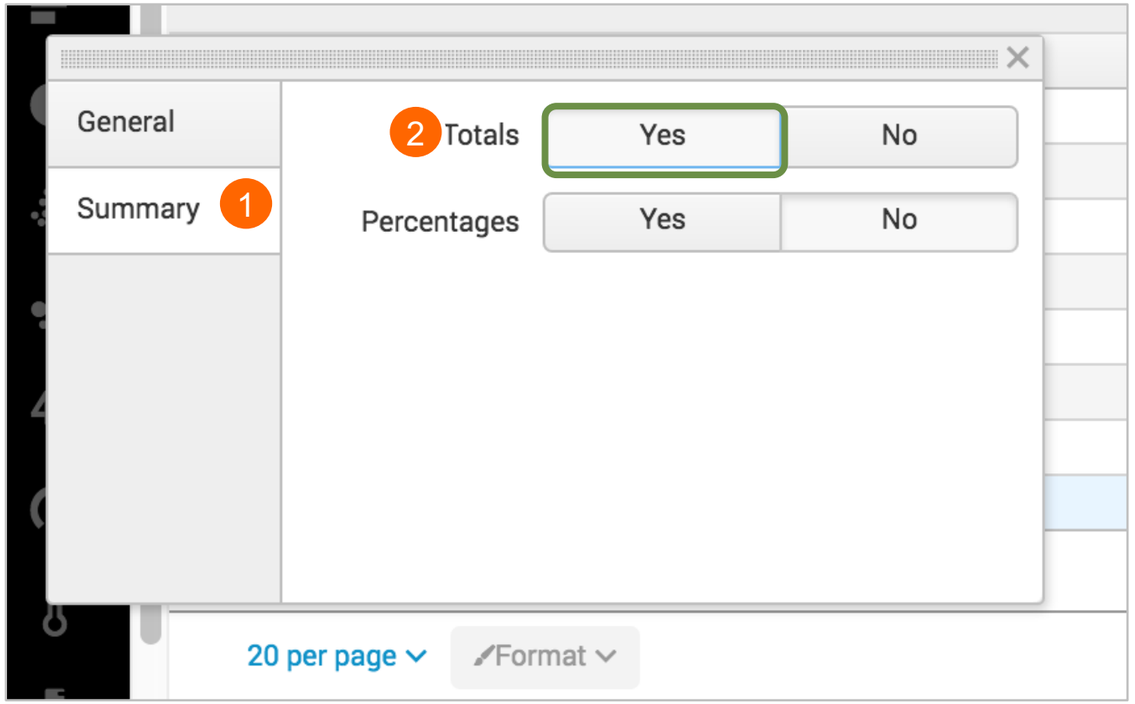


Results

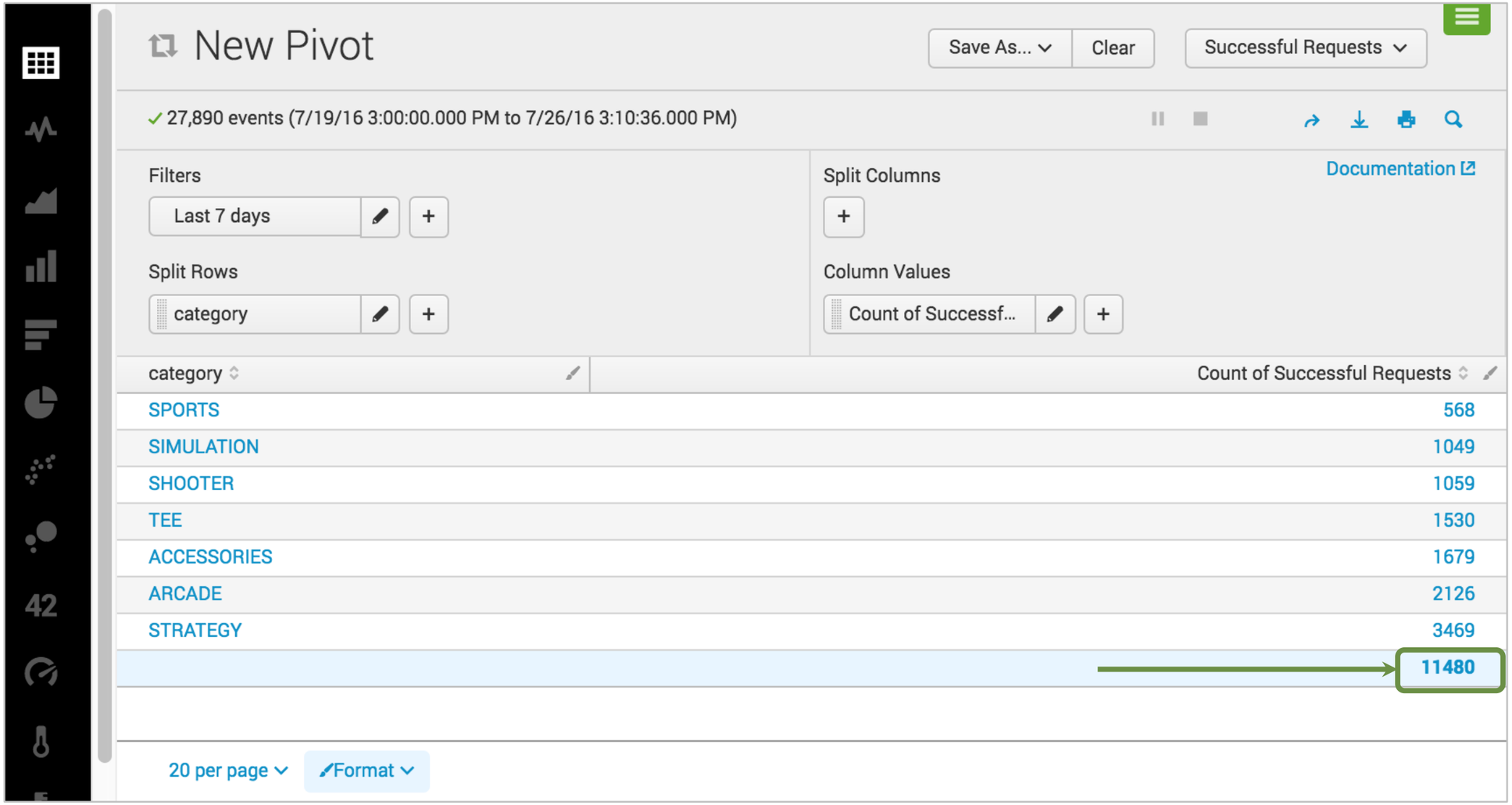


Formatting the Results

For example, to add totals on the **Summary** tab, click **Yes** next to **Totals**

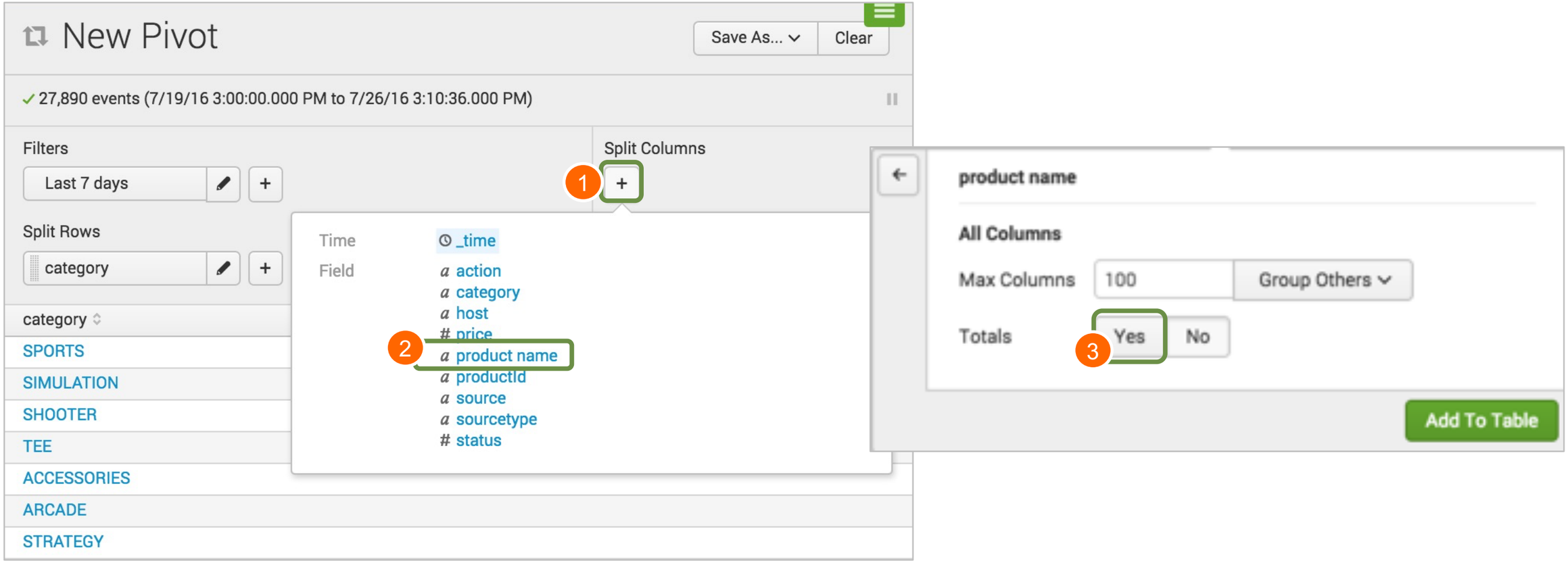


Updated Results (with Total)

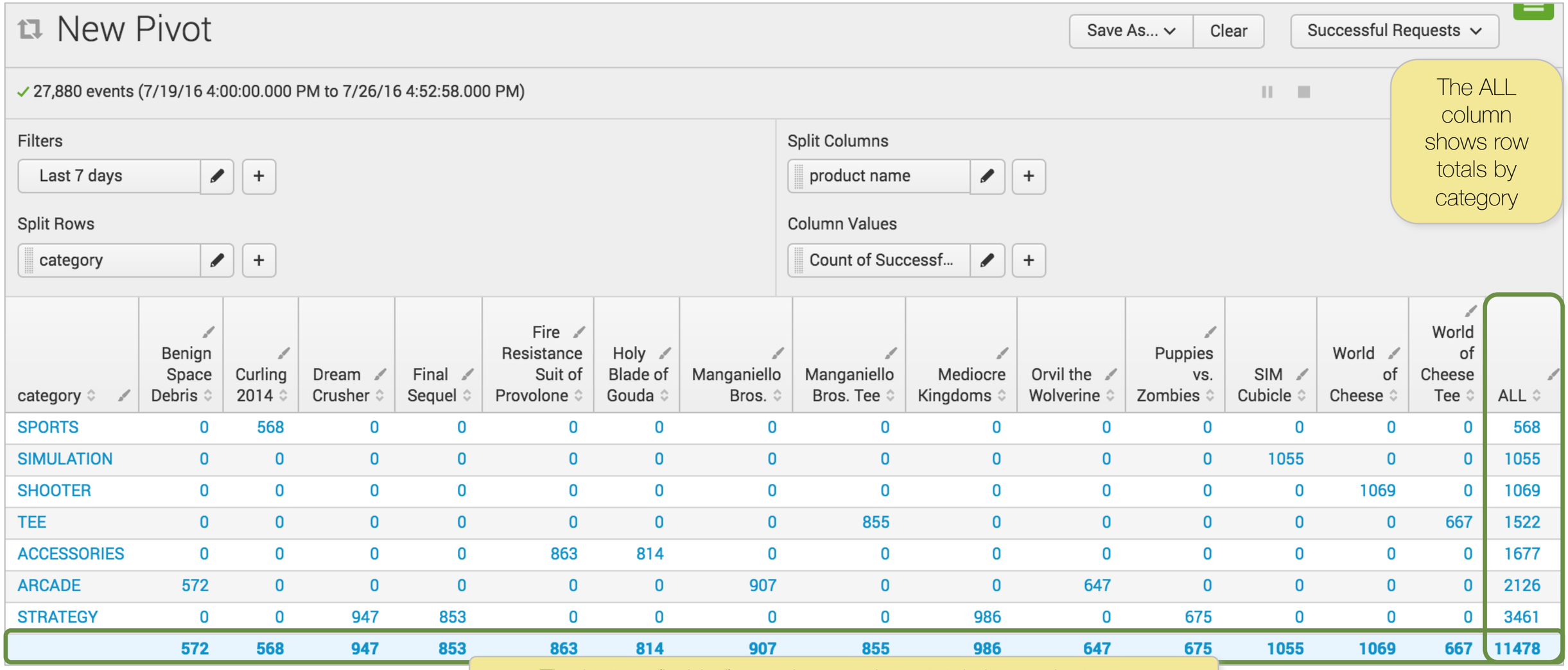


Split Columns

- Click + under **Split Columns** and select the desired split
- Specify the maximum number of columns and whether you want Totals



Results

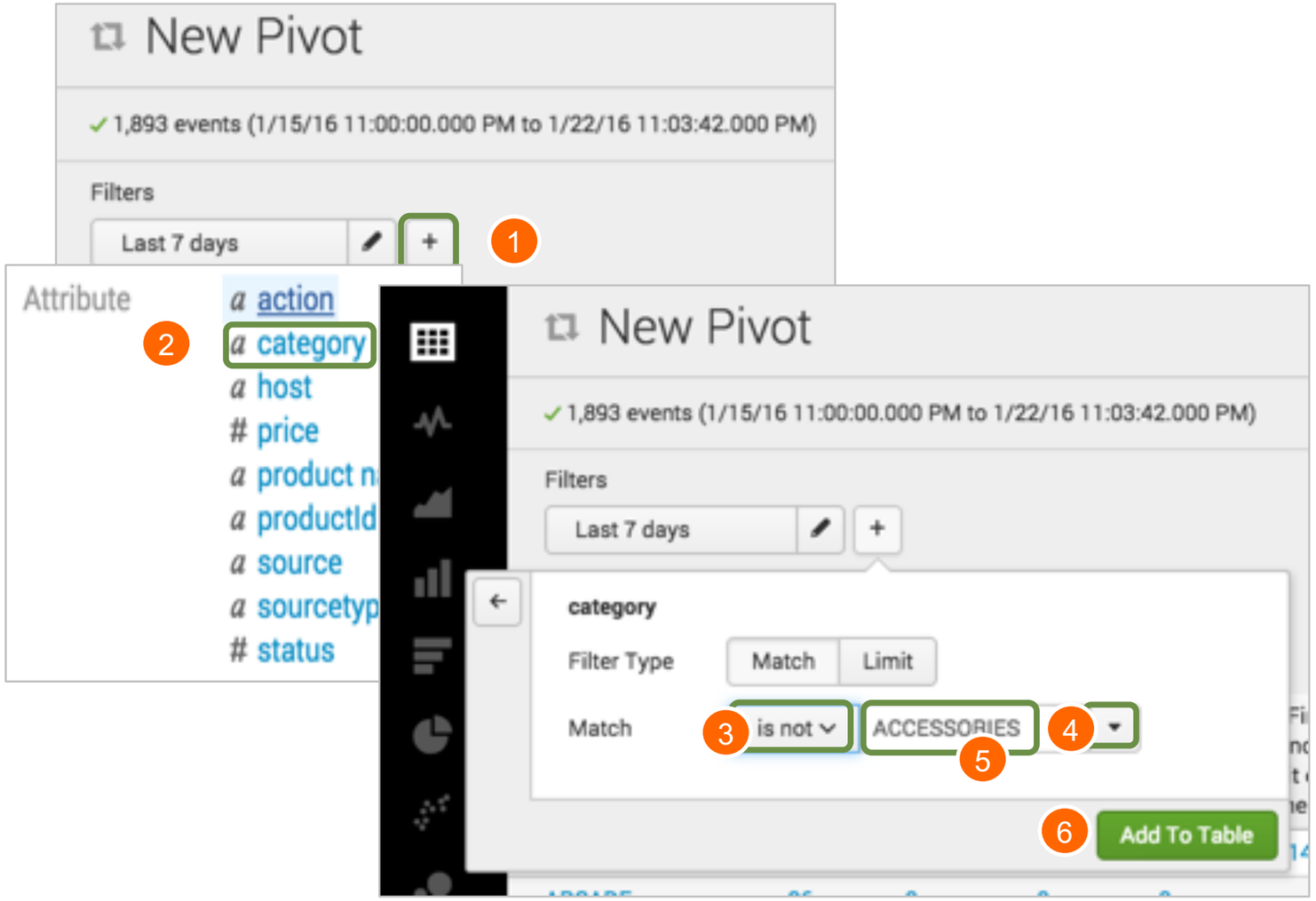


category	Benign Space Debris	Curling 2014	Dream Crusher	Final Sequel	Fire Resistance Suit of Provolone	Holy Blade of Gouda	Manganiello Bros.	Manganiello Bros. Tee	Mediocre Kingdoms	Orvil the Wolverine	Puppies vs. Zombies	SIM Cubicle	World of Cheese	World of Cheese Tee	ALL
SPORTS	0	568	0	0	0	0	0	0	0	0	0	0	0	0	568
SIMULATION	0	0	0	0	0	0	0	0	0	0	0	1055	0	0	1055
SHOOTER	0	0	0	0	0	0	0	0	0	0	0	0	1069	0	1069
TEE	0	0	0	0	0	0	0	855	0	0	0	0	0	667	1522
ACCESSORIES	0	0	0	0	863	814	0	0	0	0	0	0	0	0	1677
ARCADE	572	0	0	0	0	0	907	0	0	647	0	0	0	0	2126
STRATEGY	0	0	947	853	0	0	0	0	986	0	675	0	0	0	3461
	572	**568**	**947**	**853**	**863**	**814**	**907**	**855**	**986**	**647**	**675**	**1055**	**1069**	**667**	**11478**

The bottom (bolded) row shows column totals by product name

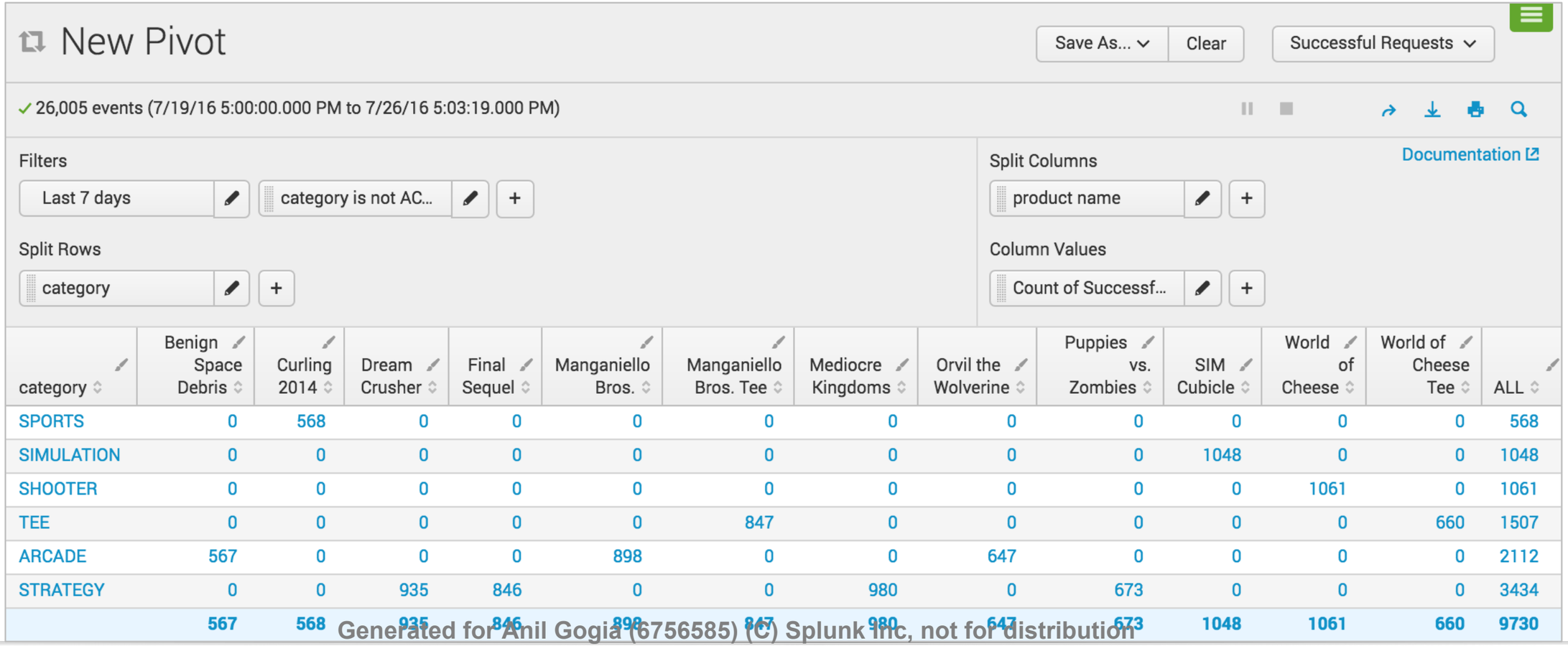
Add Additional Filters

- You can refine a pivot by filtering on key/value pairs
  - Think of 'split by' as rows and columns as the fields to display
  - Think of filters as a field=value inclusion, exclusion or specific condition to apply to the search (=, <, >, !=, *)
- In the example, the pivot is filtered to exclude events from the ACCESSORIES category



Filtered Pivot

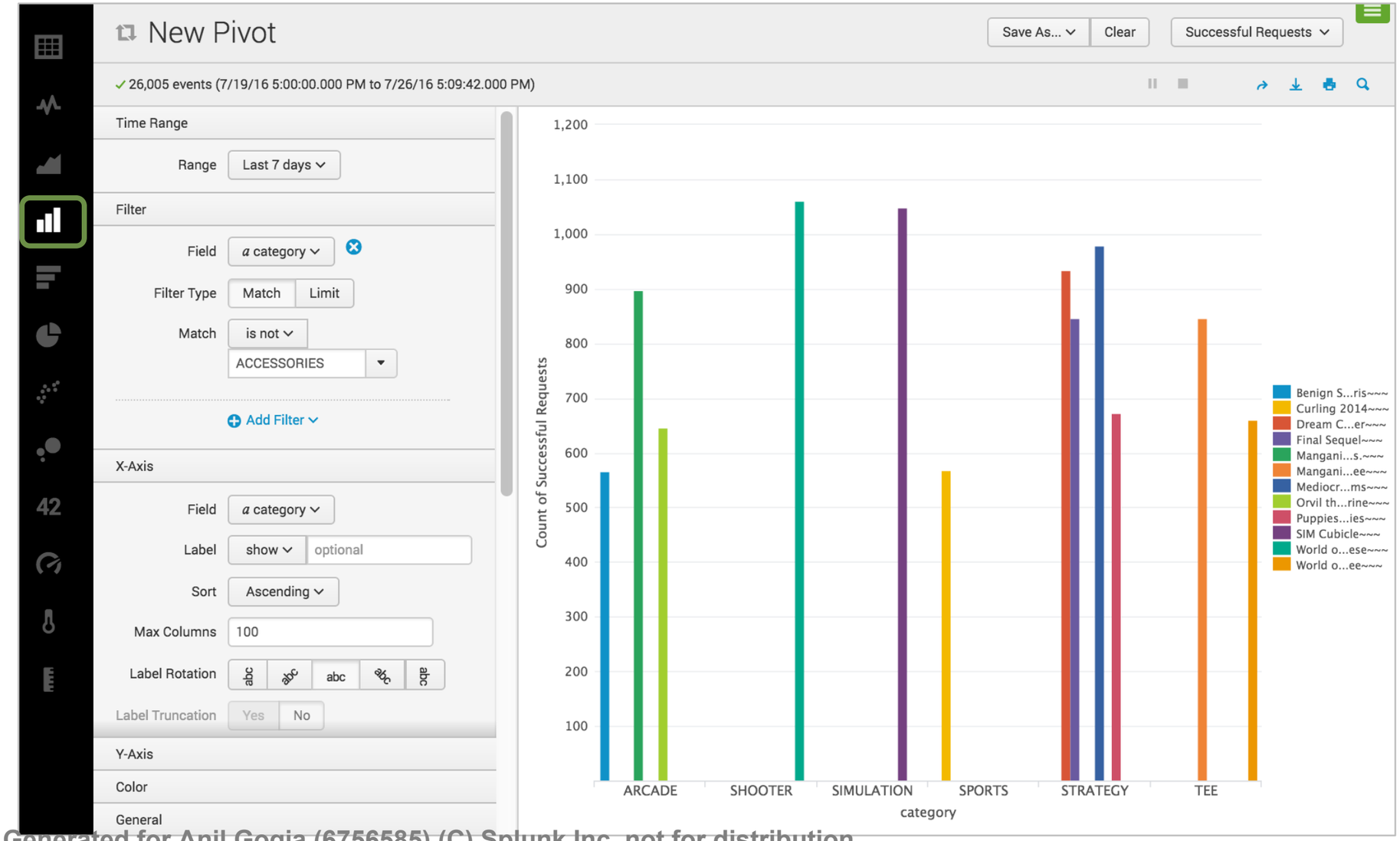
- The ACCESSORIES category is filtered out
- All the other categories remain



category	Benign Space Debris	Curling 2014	Dream Crusher	Final Sequel	Manganiello Bros.	Manganiello Bros. Tee	Mediocre Kingdoms	Orvil the Wolverine	Puppies vs. Zombies	SIM Cubicle	World of Cheese	World of Cheese Tee	ALL
SPORTS	0	568	0	0	0	0	0	0	0	0	0	0	568
SIMULATION	0	0	0	0	0	0	0	0	0	1048	0	0	1048
SHOOTER	0	0	0	0	0	0	0	0	0	0	1061	0	1061
TEE	0	0	0	0	0	847	0	0	0	0	0	660	1507
ARCADE	567	0	0	0	898	0	0	647	0	0	0	0	2112
STRATEGY	0	0	935	846	0	0	980	0	673	0	0	0	3434
	567	568	935	846	898	847	980	647	673	1048	1061	660	9730

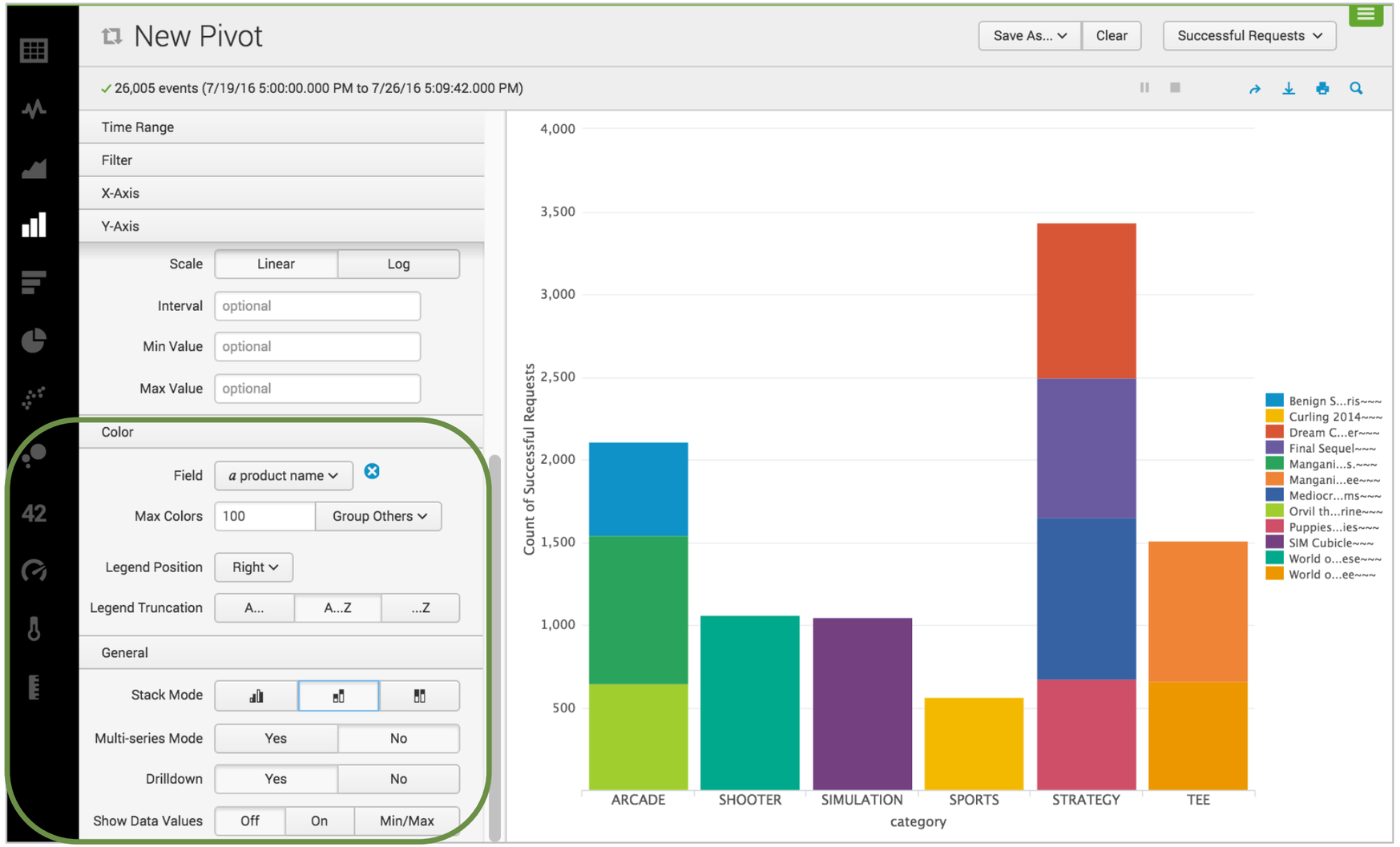
Select a Visualization Format

You can display your pivot as a table or a visualization, such as a column chart



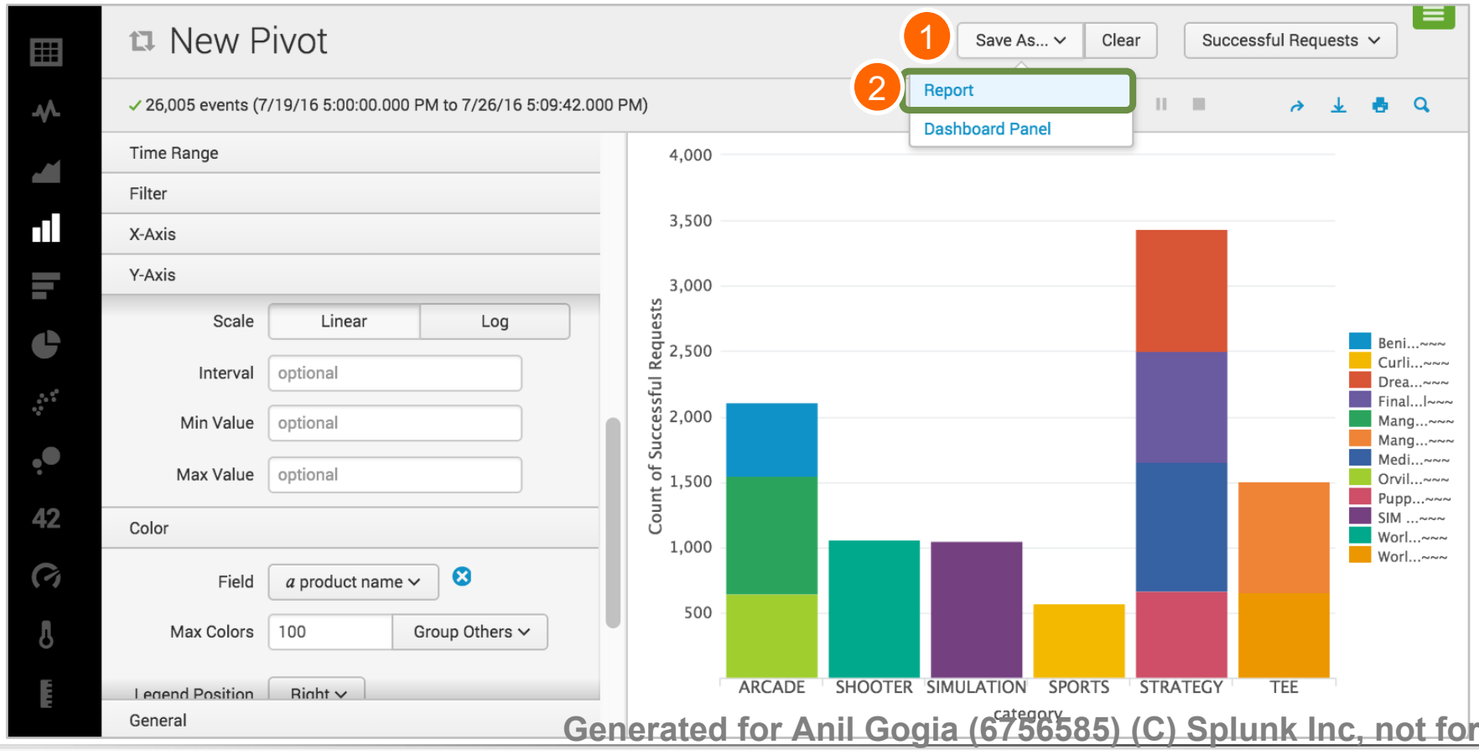
Modify Visualization Settings

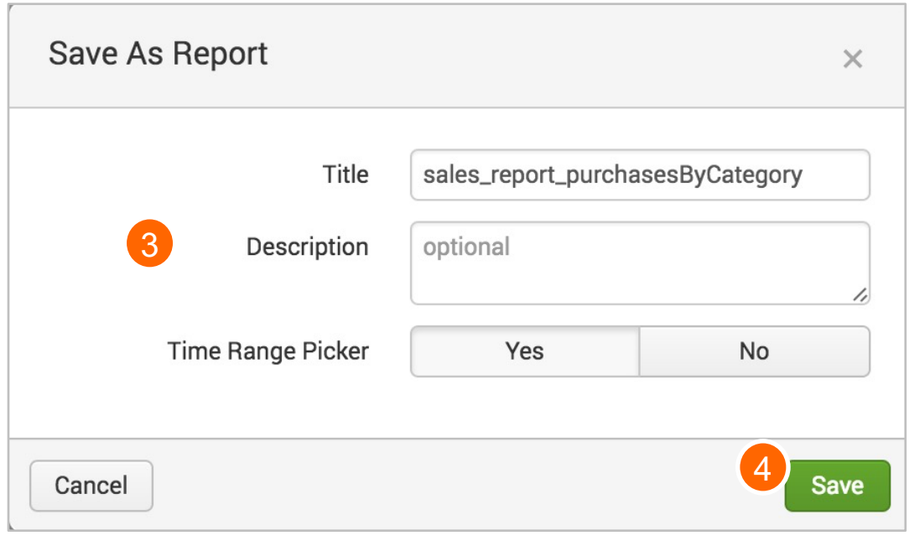
- When a visualization control is selected, panels appear that let you configure its settings
- In this example:
  - The results for each category are broken down by **product_name**
  - The stack mode is set to stacked 



Saving a Pivot

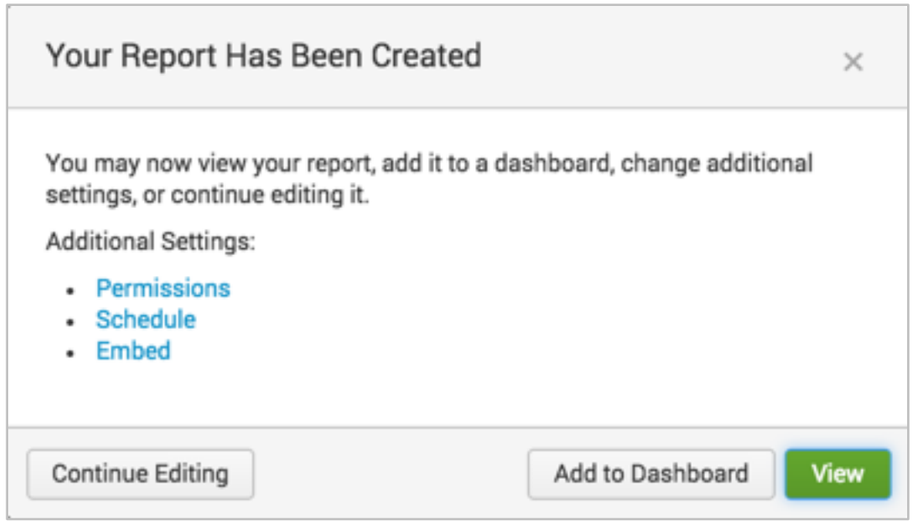
- Pivots can be saved as reports
  - You can choose to include a Time Range Picker in the report to allow people who run it to change the time range (default is Yes)
  - You will learn more about reports later in this course

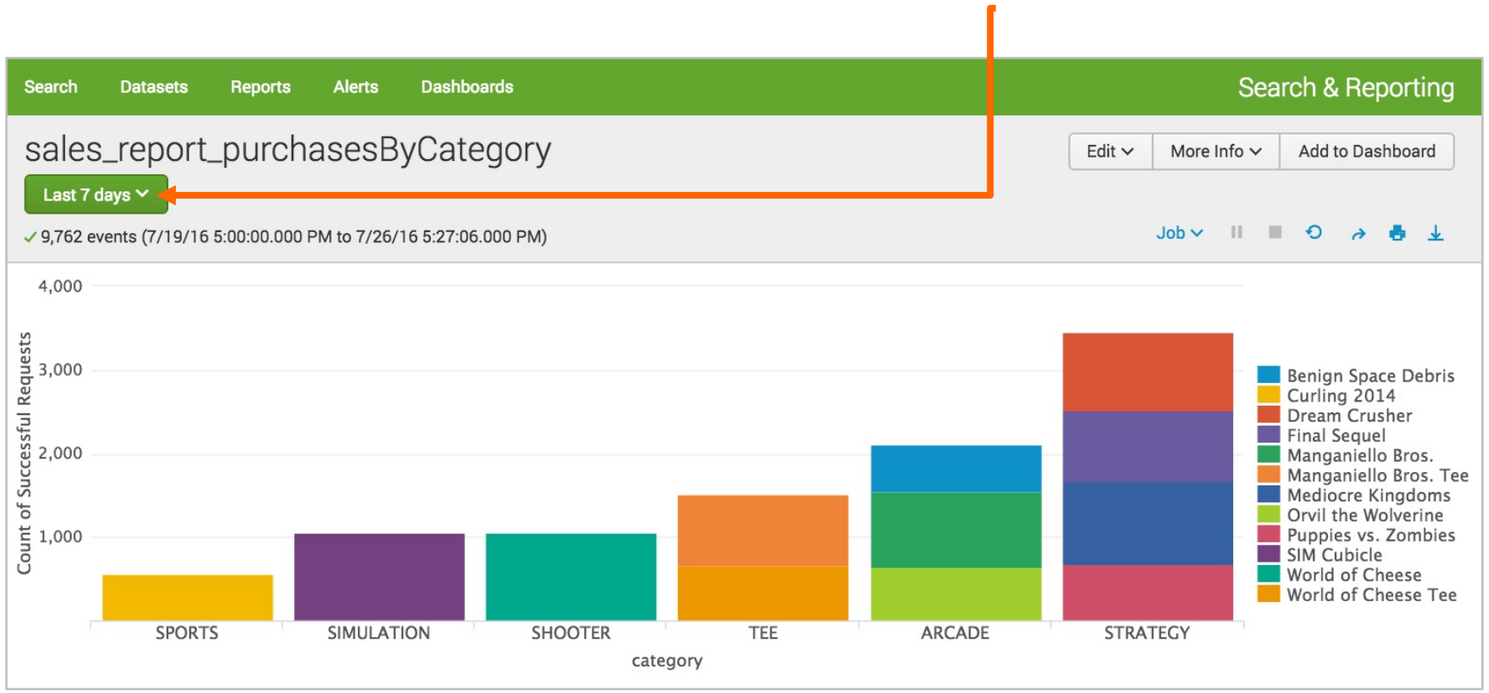




Saving a Pivot (cont.)

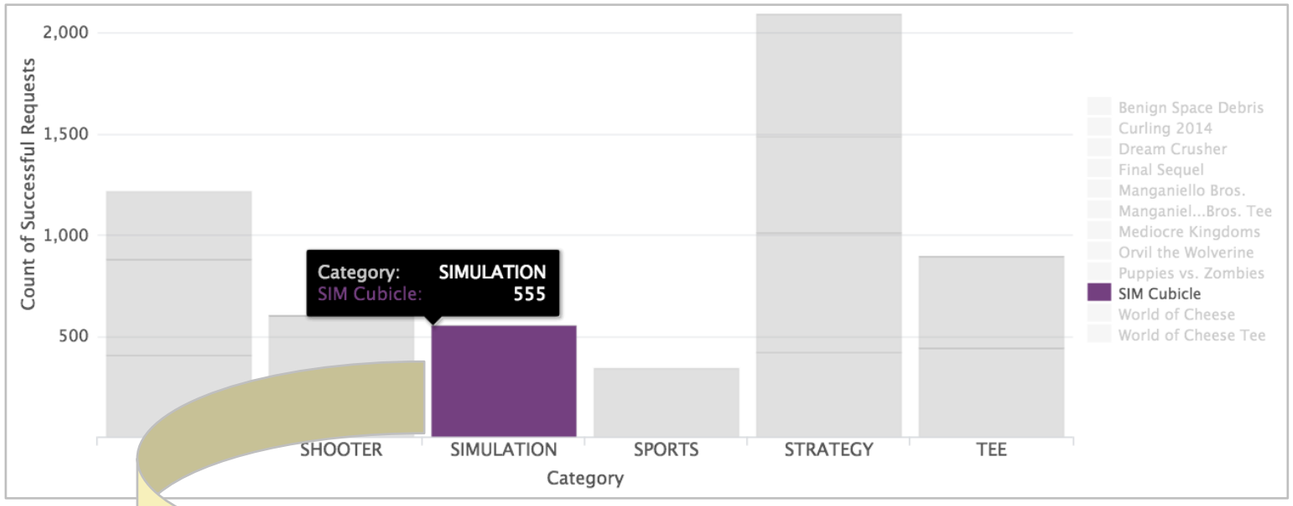
When you click **View**, the report is displayed with a Time Range Picker

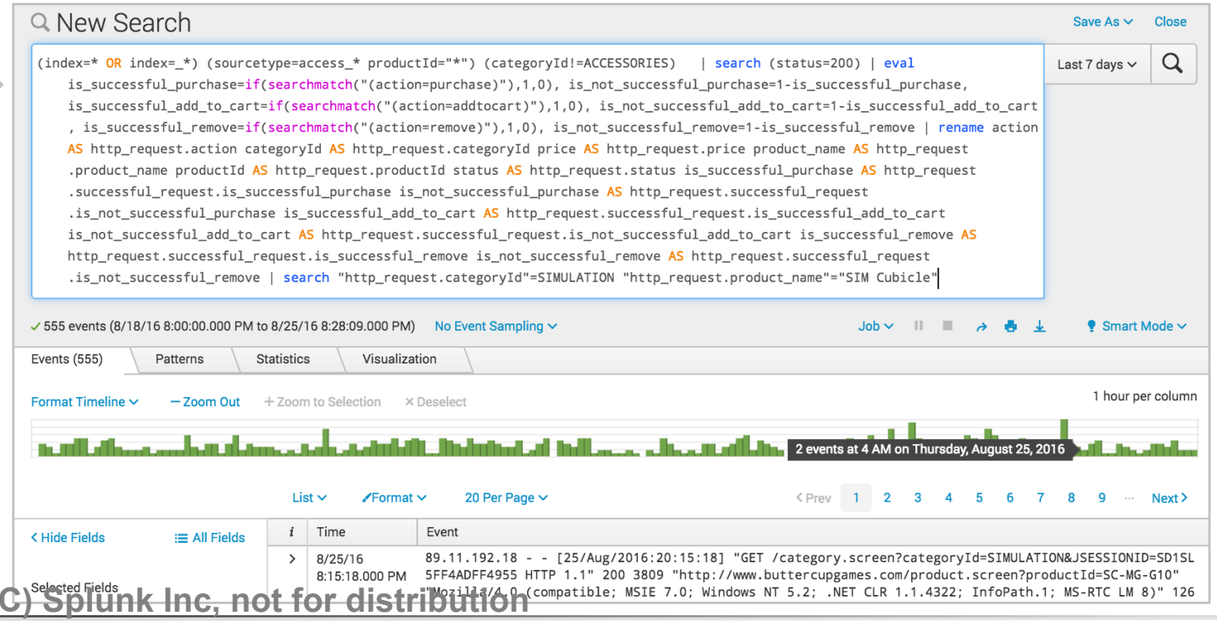




Mouse Actions

- Mouse over an object to reveal its details
- If drilldown is enabled (default), it is possible to click on the object to expose the underlying search





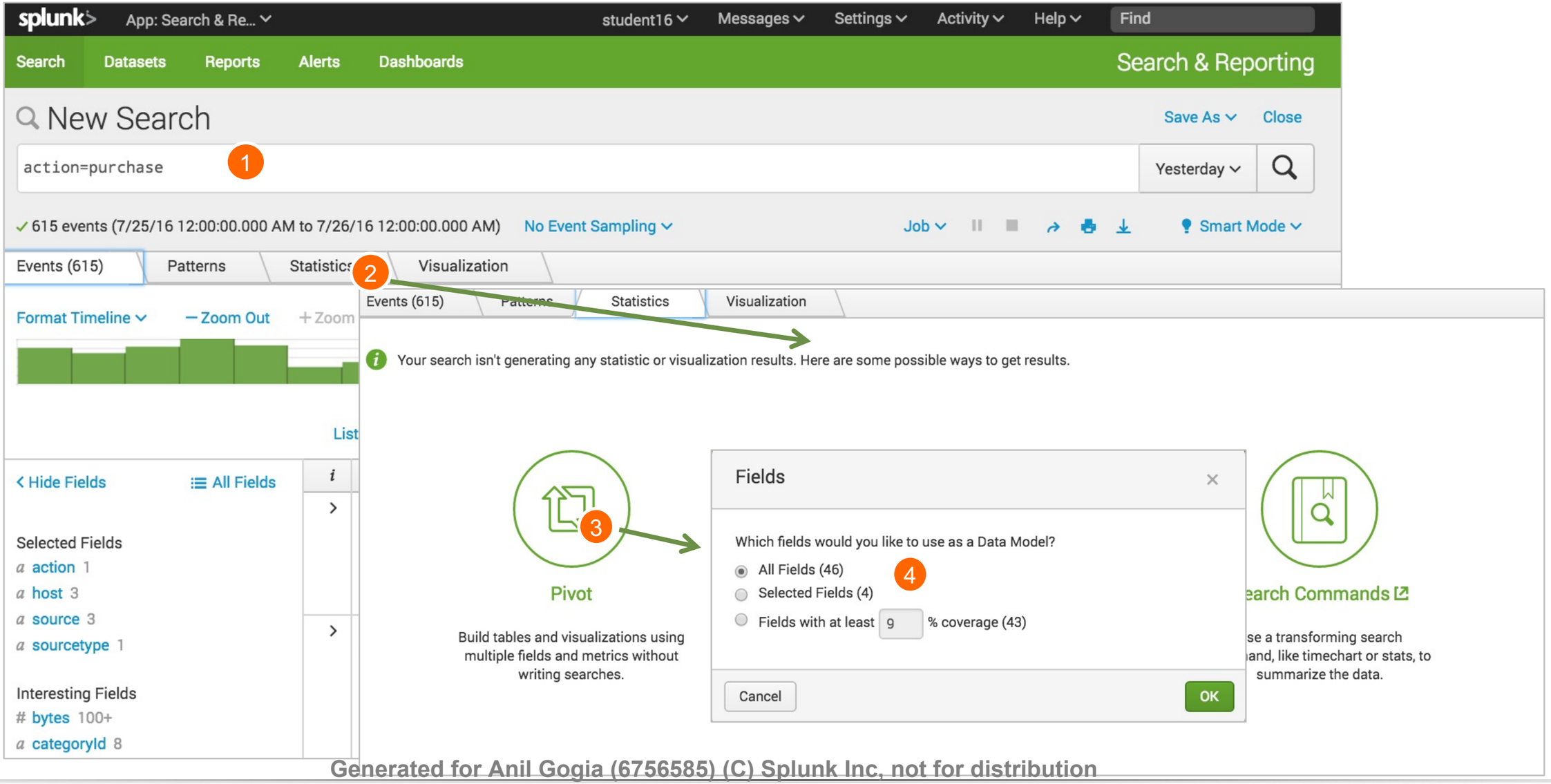
Note

The search generated by drilldown may be more detailed than your original search. However, it produces the same results.

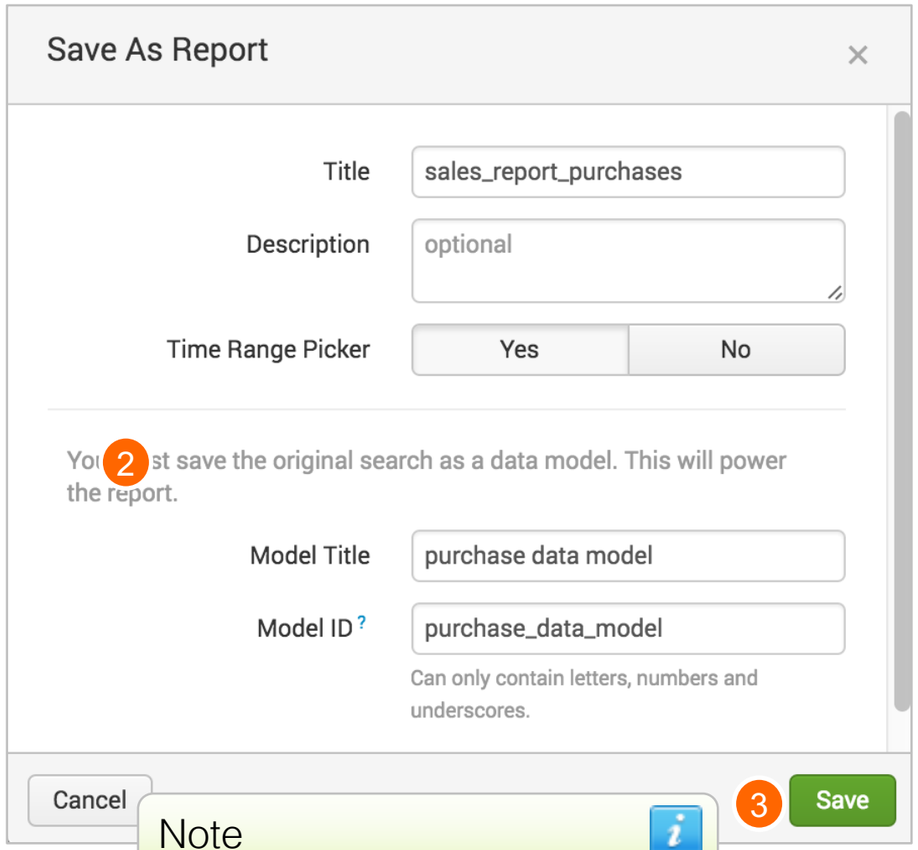
Instant Pivot Overview

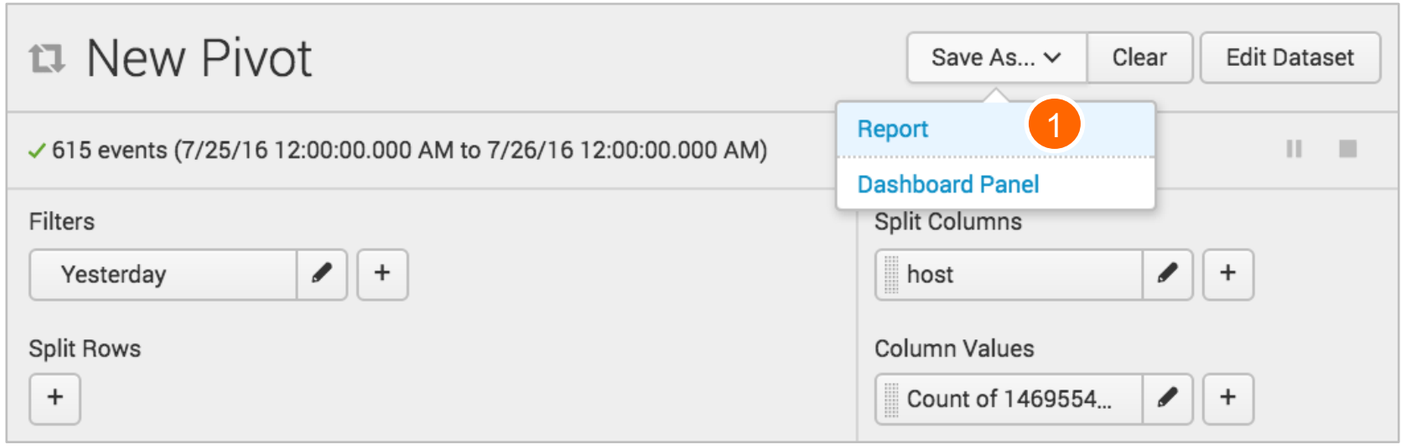
- Instant pivot allows you to utilize the pivot tool without a preexisting data model
  - Instant pivot creates an underlying data model utilizing the search criteria entered during the initial search
- How to create an Instant Pivot
  1. Execute a search (search criteria only, no search commands)
  2. Click the **Statistics** or **Visualization** tab
  3. Click the **Pivot** icon
  4. Select the fields to be included in the data model object
  5. Create the pivot (table or chart)

Open Instant Pivot



Saving a Pivot as a Report

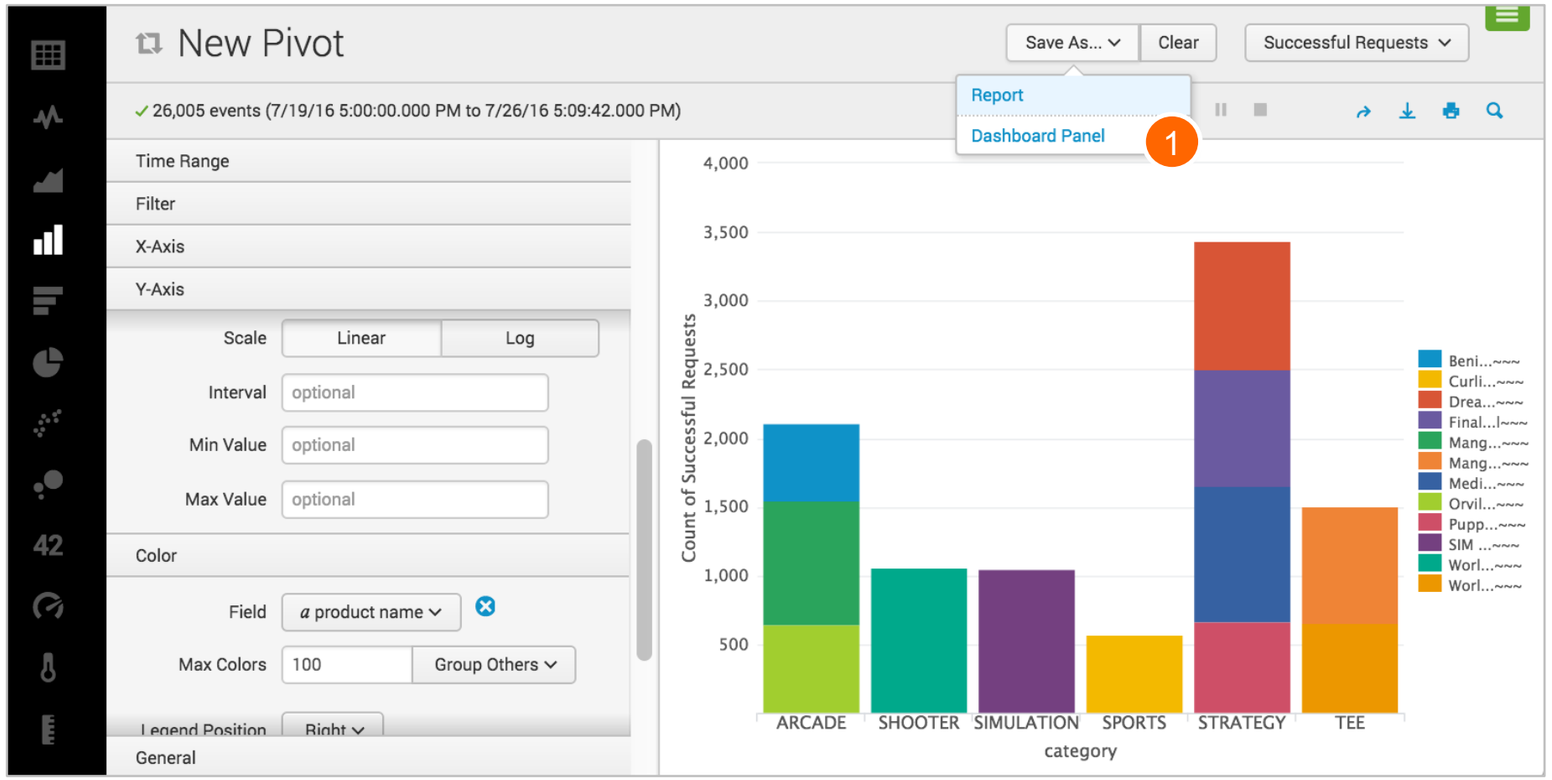


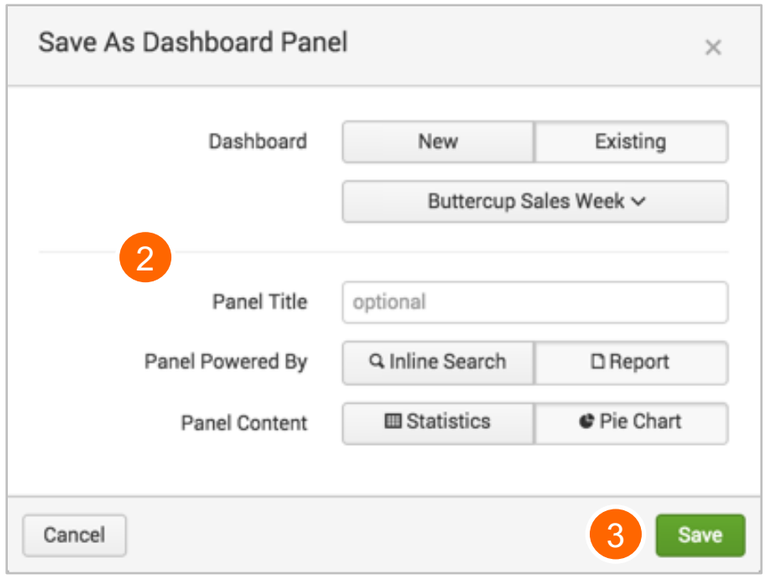


- When saving as a report, the **Model Title** is required
  - This is used to create a data model, which is required by the pivot report
- The **Model ID** is automatically generated based on the **Model Title**

Add a Pivot to a Dashboard

Similarly, you can save any pivot to a new or existing dashboard





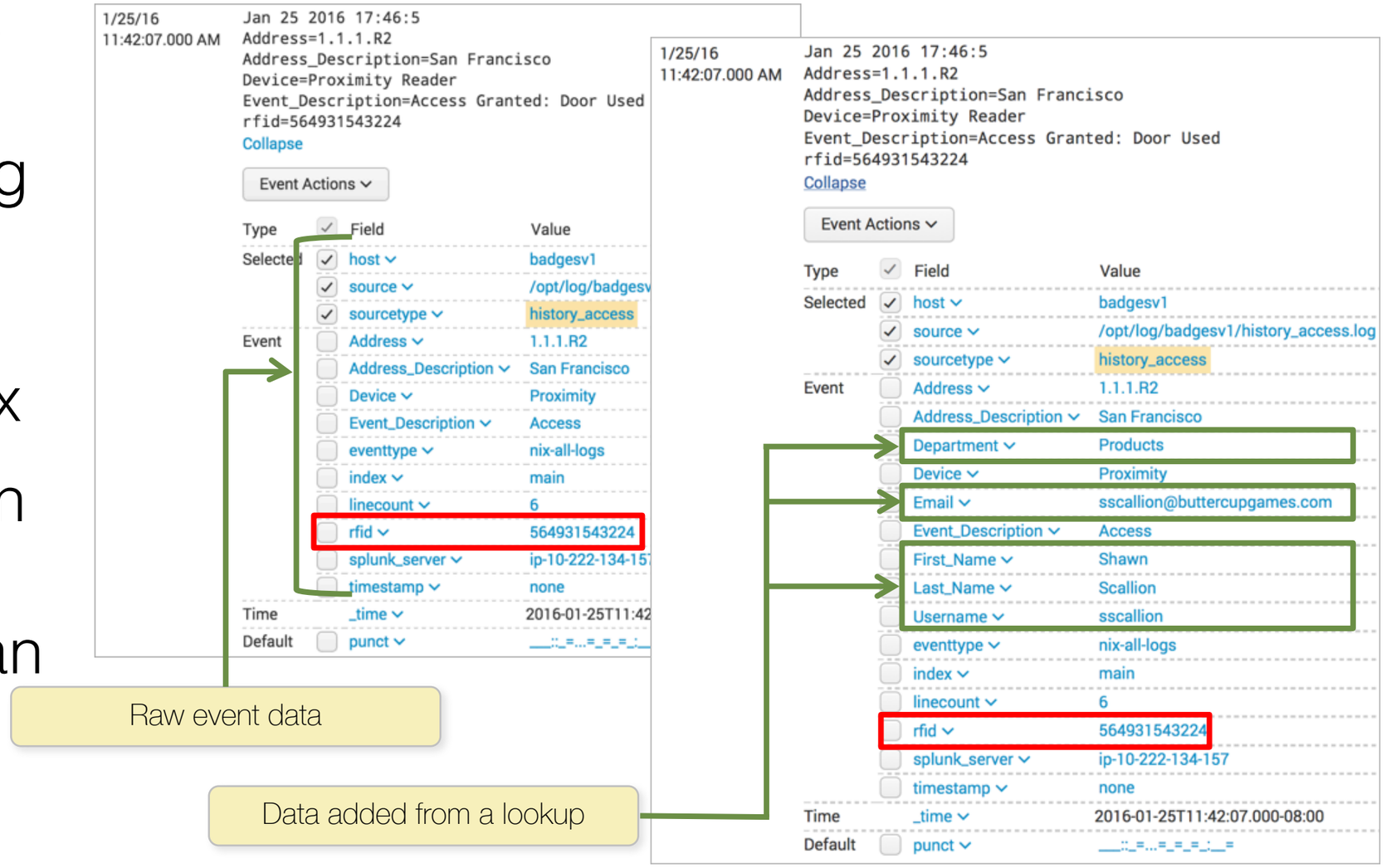
Module 12: Creating and Using Lookups

Module Objectives

- Describe lookups
- Examine a lookup file example
- Create a lookup file and definition
- Configure an automatic lookup
- Use the lookup in searches

Describing Lookups

- There are use cases where static or relatively unchanging data is required for searches, but is not available in the index
- For example, from an RFID in a badge reader event, you can look up employee information

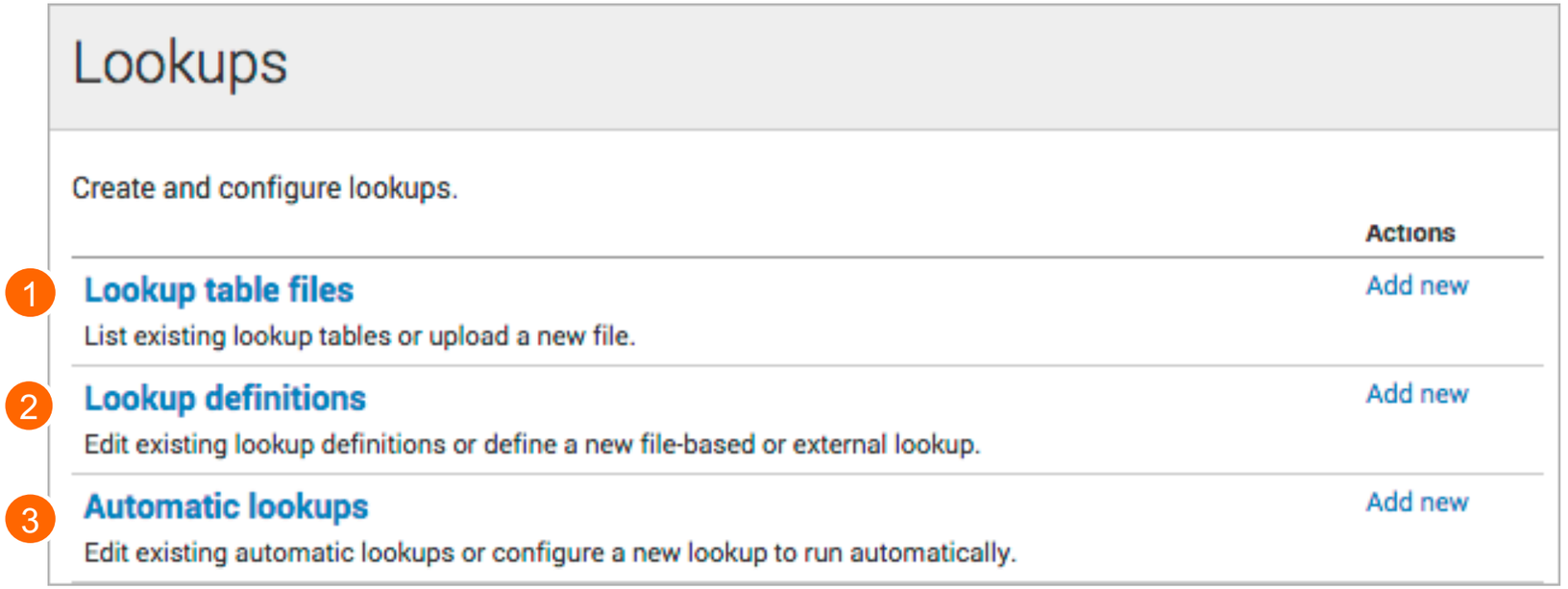


Describing Lookups (cont.)

- Lookups allow you to add more fields to your events:
  - Provide descriptions for http status codes (“file not found”, “service unavailable”)
  - Define sale prices for products
  - Associate RFIDs with user names, IP addresses, and workstation IDs
- Lookups can be defined in a static.csv file, or it can be the output of a Python script
- After a field lookup is configured, you can use the lookup fields in searches
- The lookup fields also appear in the Fields sidebar
- Lookup field values are case-sensitive by default
  - Admins can change the `case_sensitive_match` option to `false` in transforms.conf

Defining a File-based Lookup

1. Upload the file required for the lookup
2. Define the lookup type
3. Optionally, configure the lookup to run automatically



Lookup File – Example

- This example displays a lookup `.csv` file used to associate product information with productId
- First row represents field names (header)

  `productId, product_name, categoryId, price, sale_price, Code`

- The `productId` field exists in the `access_combined` events
  - This is the **input** field
- All of the fields listed above are available to search after the lookup is defined
  - These are the **output** fields

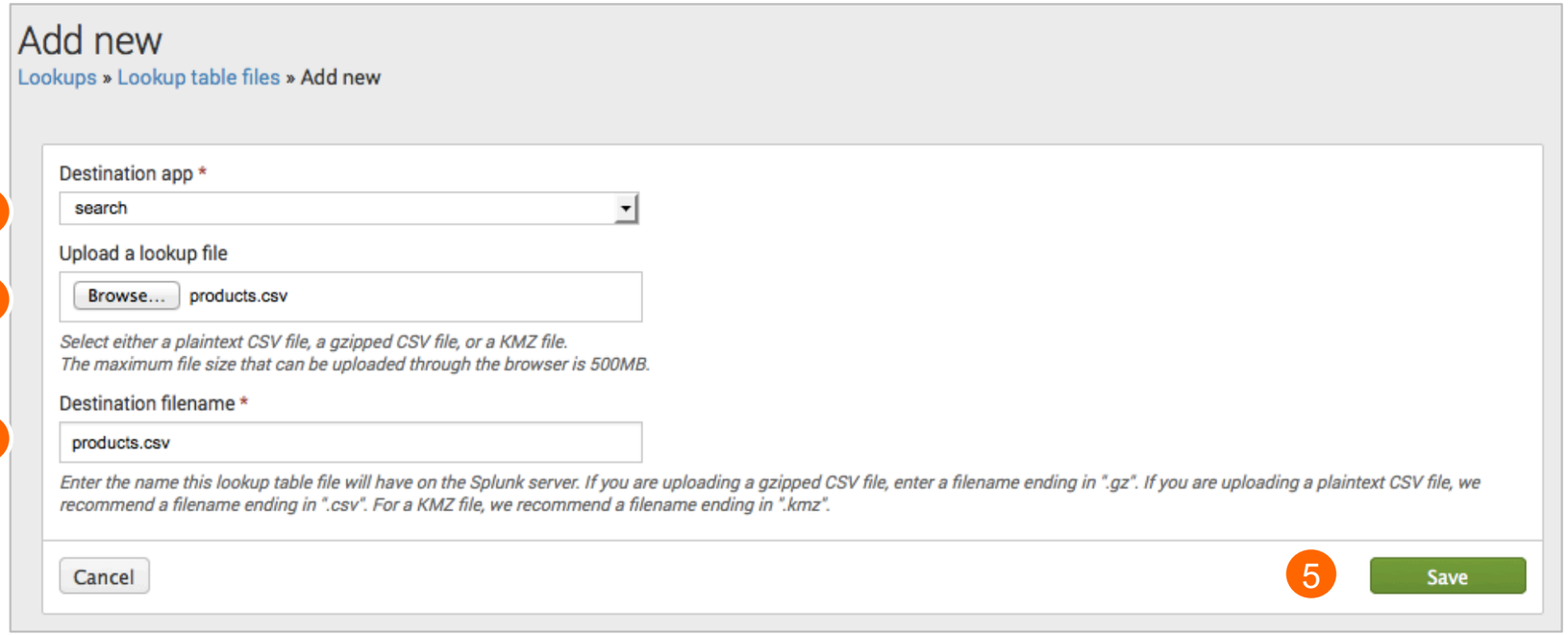
```
 GNU nano 2.3.1                File: products.csv

productId,product_name,categoryId,price,sale_price,Code
DB-SG-G01,Mediocre Kingdoms,STRATEGY,24.99,19.99,A
DC-SG-G02,Dream Crusher,STRATEGY,39.99,24.99,B
FS-SG-G03,Final Sequel,STRATEGY,24.99,16.99,C
WC-SH-G04,World of Cheese,SHOOTER,24.99,19.99,D
WC-SH-T02,World of Cheese Tee,TEE,9.99,6.99,E
PZ-SG-G05,Puppies vs. Zombies,STRATEGY,4.99,1.99,F
CU-PG-G06,Curling 2014,SPORTS,19.99,16.99,G
MB-AG-G07,Manganiello Bros.,ARCADE,39.99,24.99,H
MB-AG-T01,Manganiello Bros. Tee,TEE,9.99,6.99,I
FI-AG-G08,Orvil the Wolverine,ARCADE,39.99,24.99,J
BS-AG-G09,Benign Space Debris,ARCADE,24.99,19.99,K
SC-MG-G10,SIM Cubicle,SIMULATION,19.99,16.99,L
WC-SH-A01,Holy Blade of Gouda,ACCESSORIES,5.99,2.99,M
WC-SH-A02,Fire Resistance Suit of Provolone,ACCESSORIES,3.99,1.99,N
```


Creating a Lookup Table

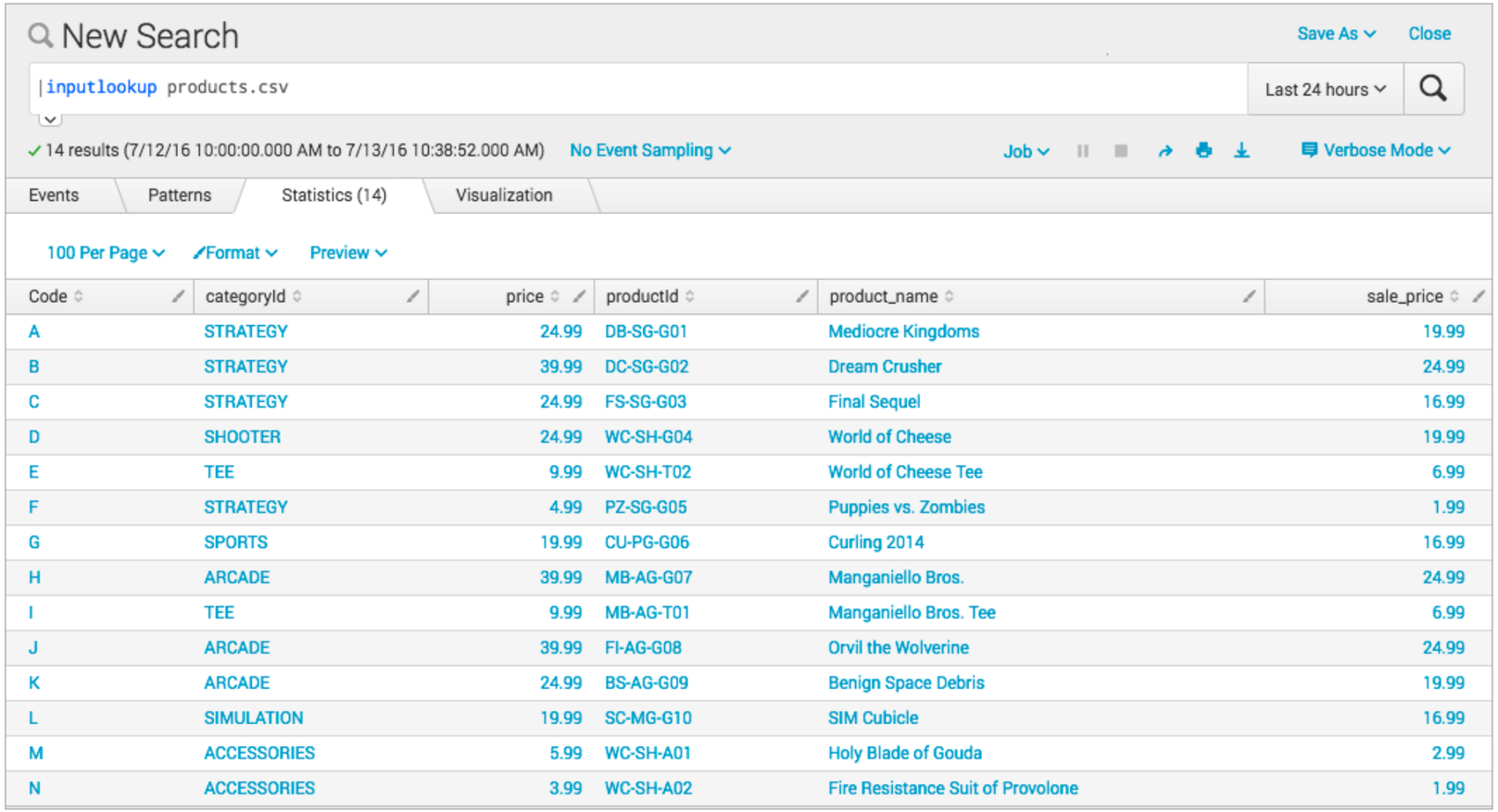
Settings > Lookups > Lookup table files

1. Click **New**
2. Select a destination app
3. Browse and select the `.csv` file to use for the lookup table
4. Enter a name for the lookup file
5. Save



`inputlookup` Command

- Use the **`inputlookup`** command to load the results from a specified static lookup
- Useful to:
  - Review the data in the .csv file
  - Validate the lookup



Code	categoryId	price	productId	product_name	sale_price
A	STRATEGY	24.99	DB-SG-G01	Mediocre Kingdoms	19.99
B	STRATEGY	39.99	DC-SG-G02	Dream Crusher	24.99
C	STRATEGY	24.99	FS-SG-G03	Final Sequel	16.99
D	SHOOTER	24.99	WC-SH-G04	World of Cheese	19.99
E	TEE	9.99	WC-SH-T02	World of Cheese Tee	6.99
F	STRATEGY	4.99	PZ-SG-G05	Puppies vs. Zombies	1.99
G	SPORTS	19.99	CU-PG-G06	Curling 2014	16.99
H	ARCADE	39.99	MB-AG-G07	Manganiello Bros.	24.99
I	TEE	9.99	MB-AG-T01	Manganiello Bros. Tee	6.99
J	ARCADE	39.99	FI-AG-G08	Orvil the Wolverine	24.99
K	ARCADE	24.99	BS-AG-G09	Benign Space Debris	19.99
L	SIMULATION	19.99	SC-MG-G10	SIM Cubicle	16.99
M	ACCESSORIES	5.99	WC-SH-A01	Holy Blade of Gouda	2.99
N	ACCESSORIES	3.99	WC-SH-A02	Fire Resistance Suit of Provolone	1.99

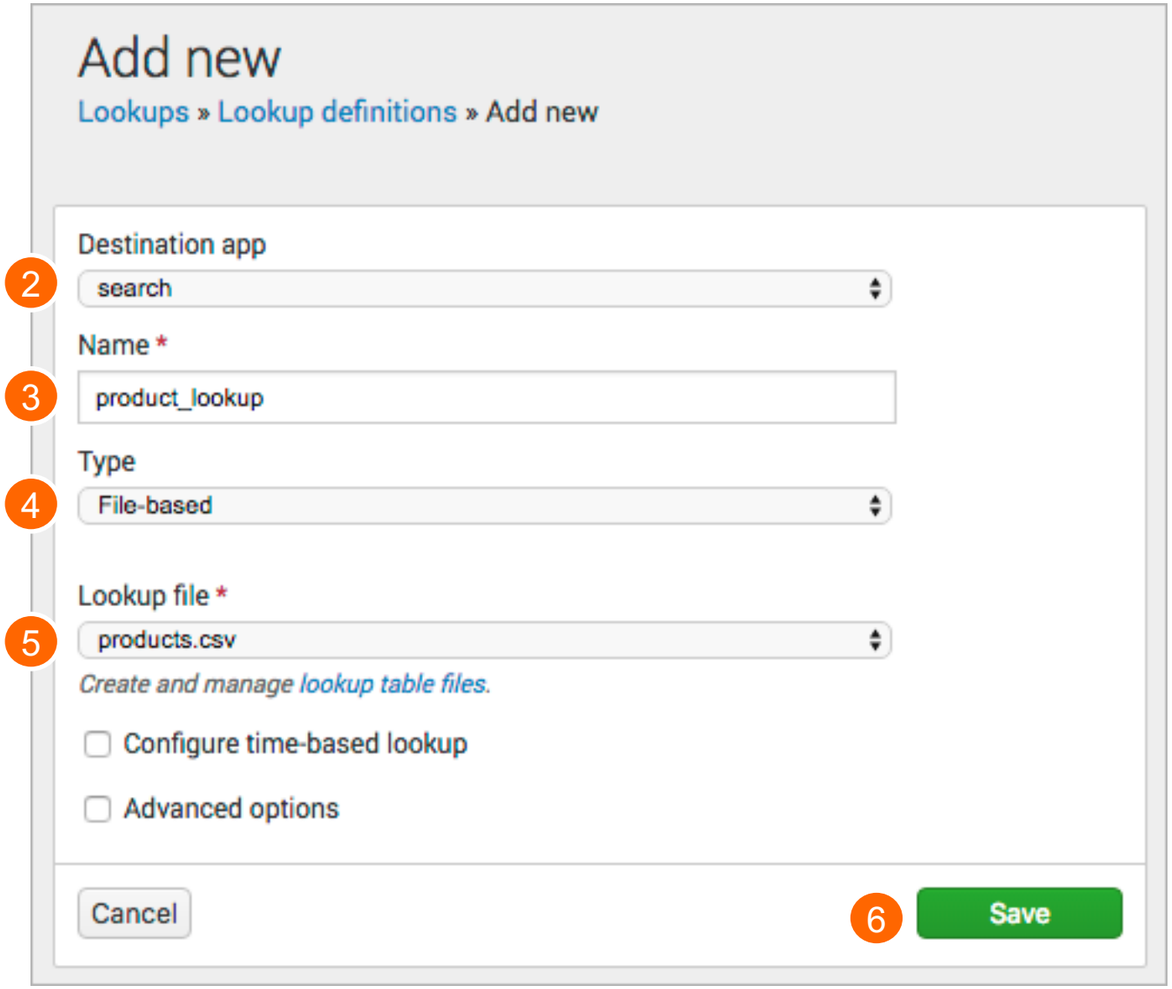
Note

When using the **`inputlookup`** command, you can specify the filename ending with .csv or the lookup definition name.

Creating a Lookup Definition

Settings > Lookups > Lookup definitions

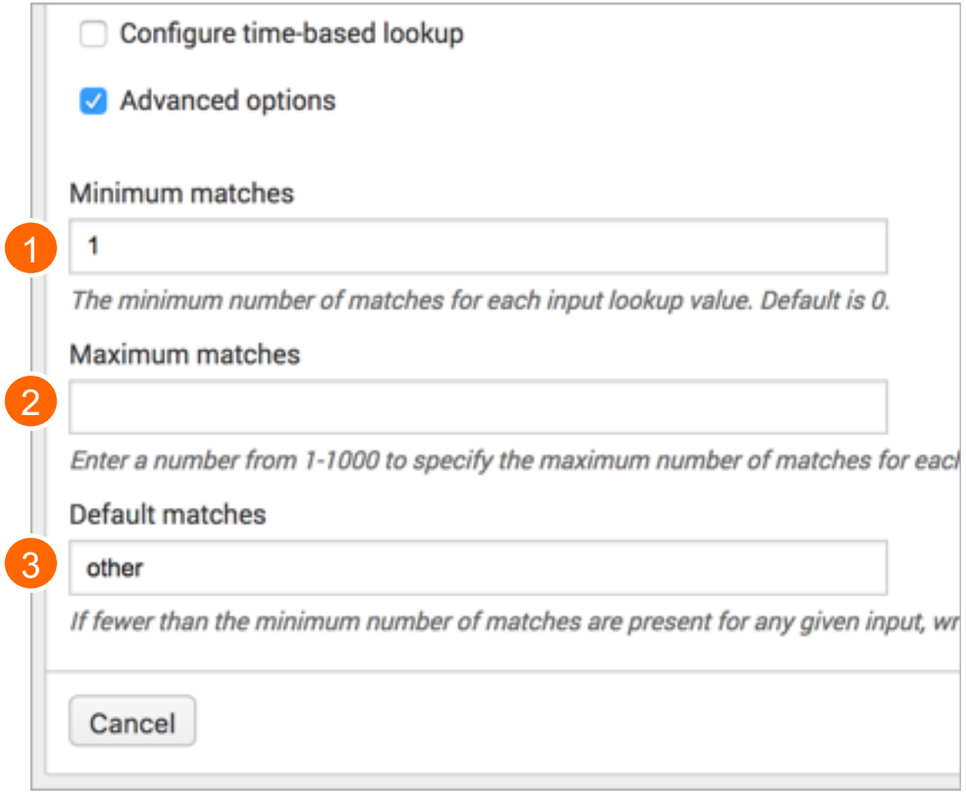
1. Click **New**
2. Select a destination app
3. Name the lookup definition
4. Select the lookup type, either File-based or External
5. From the drop-down, select a lookup file
6. Save



Applying Advanced Options

Under Advanced Options, you can specify:

1. Minimum number of matches for each input lookup value
2. Maximum number of matches for each input lookup value
3. Default value to output, if fewer than the minimum number of matches are present for a given input



`lookup` Command

- If a lookup is not configured to run automatically, use the **`lookup`** command in your search to use the lookup fields
- **`OUTPUT`** - If an **`OUTPUT`** clause is not specified, all fields in the lookup table that are not the match field are used as output fields
- If **`OUTPUT`** is specified, the fields overwrite existing fields
- The output lookup fields exist only for the current search
- Use **`OUTPUTNEW`** when you do not want to overwrite existing fields

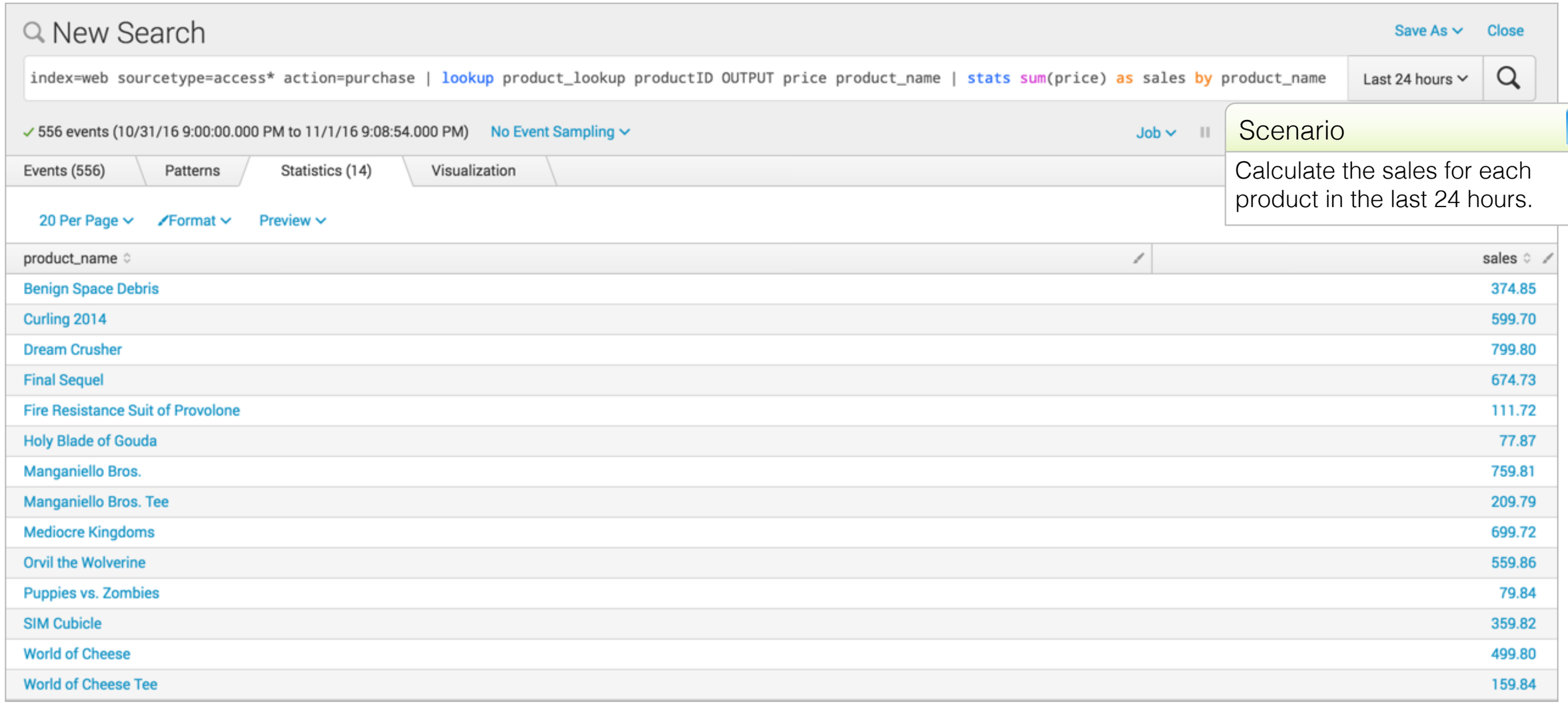
lookup Help More »
Explicitly invokes field value lookups.

Examples

There is a lookup table specified in a stanza name 'usertogroup' in transform.conf. This lookup table contains (at least) two fields, 'user' and 'group'. For each event, we look up the value of the field 'local_user' in the table and for any entries that matches, the value of the 'group' field in the lookup table will be written to the field 'user_group' in the event.

... | lookup usertogroup user as local_user OUTPUT group as user_group

Using the `lookup` Command



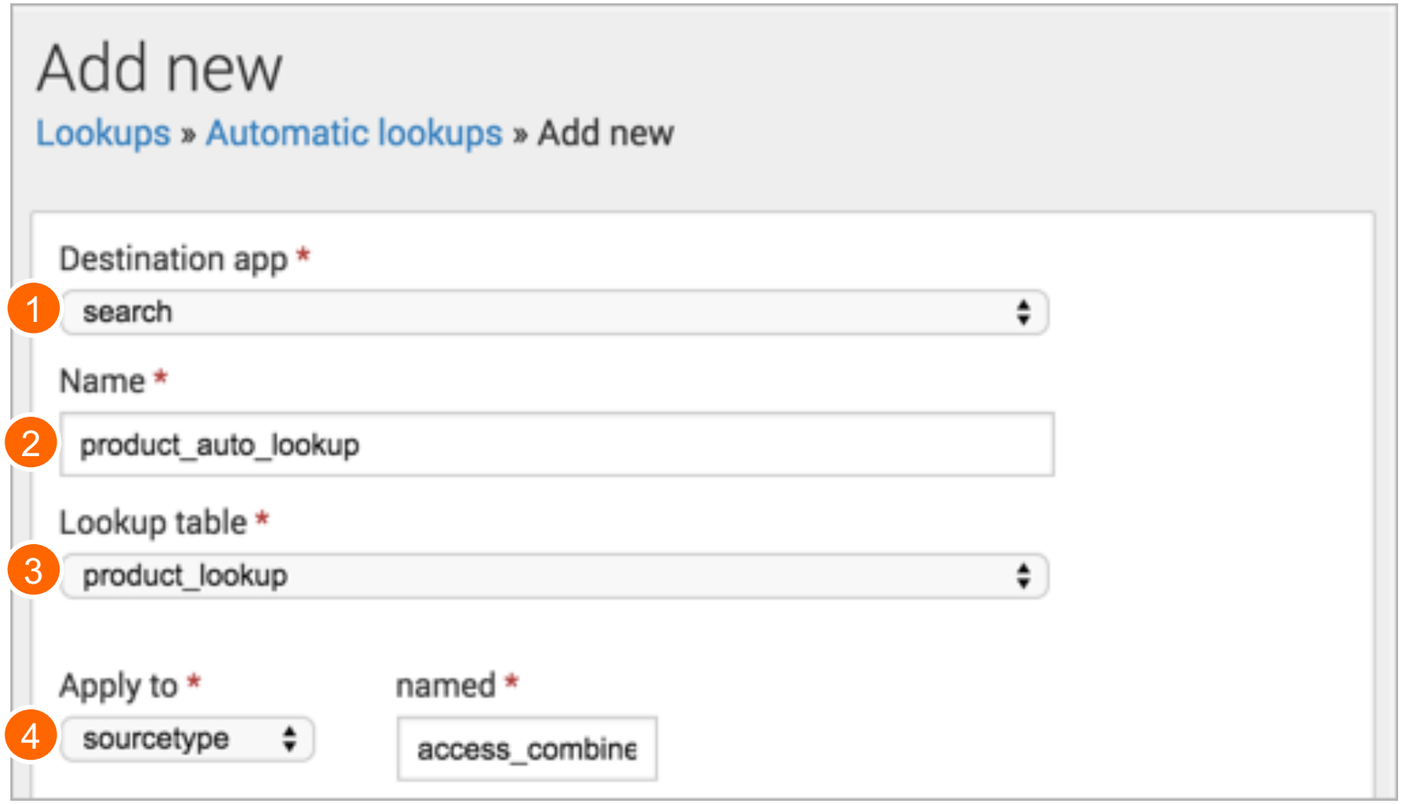
Scenario

Calculate the sales for each product in the last 24 hours.

Creating an Automatic Lookup

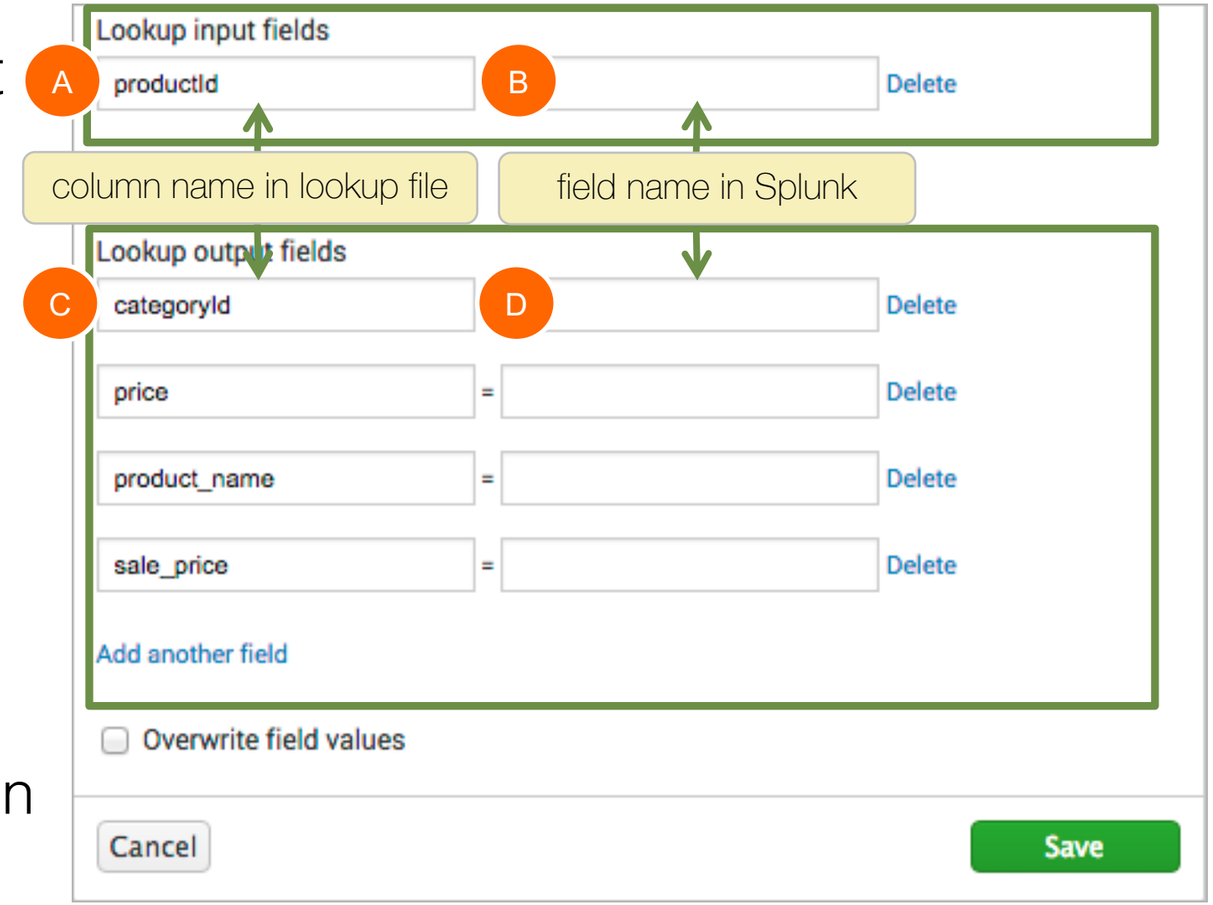
Settings > Lookups > Automatic lookups > New

1. Select the Destination app
2. Enter a Name for the lookup
3. Select the Lookup table definition
4. Select host, source, or sourcetype to apply the lookup and specify the name



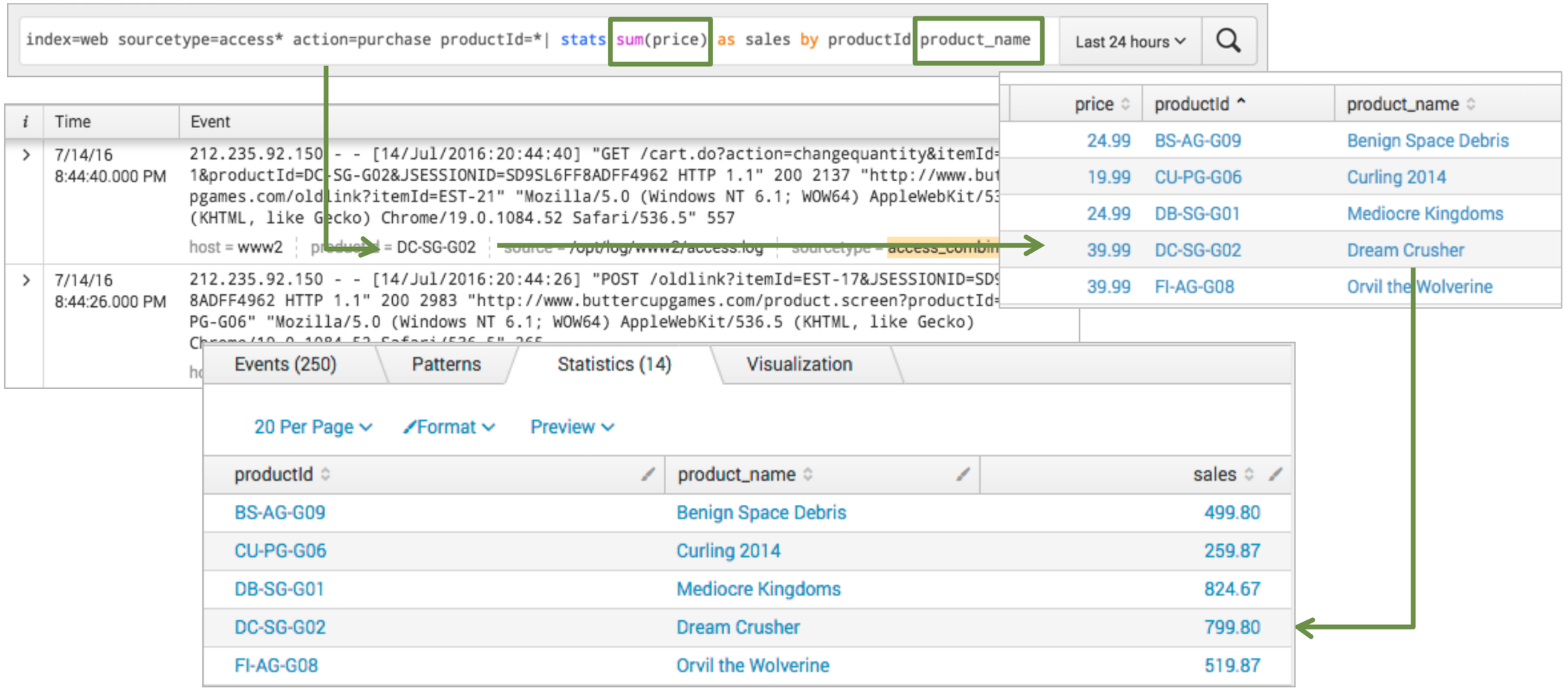
Creating an Automatic Lookup (cont.)

5. Define the Lookup input fields
   - Field(s) that exist in your events that you are relating to the lookup table
     A. Column name in CSV
     B. Field name in Splunk, if different from column name
6. Define the Lookup output fields
   - Field(s) from your lookup table that are added to the events
     C. Field name in lookup table
     D. Name you want displayed in Splunk; otherwise it inherits the column name
7. Save



Using the Automatic Lookup

To use an automatic lookup, specify the output fields in your search



Time-based Lookups

- If a field in the lookup table represents a timestamp, you can create a time-based lookup
- In this example, the search retrieved events for February and March and calculated the sales based on the correct unit price for those dates.

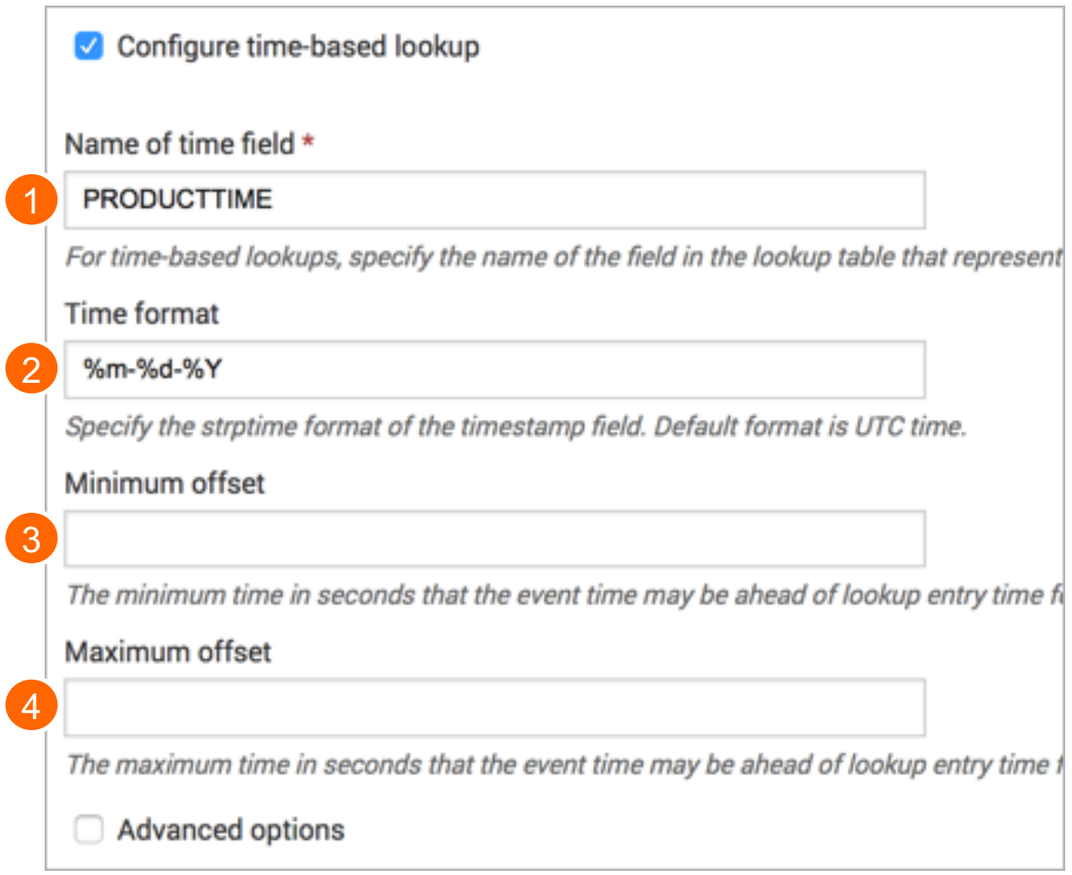
products.csv

PRODUCTTIME	productId	product_name	categoryId	price	sale_price
1/1/10	DB-SG-G01	Mediocre Kingdoms	STRATEGY	24.99	19.99
1/1/10	DC-SG-G02	Dream Crusher	STRATEGY	39.99	24.99
1/1/10	FS-SG-G03	Final Sequel	STRATEGY	24.99	16.99
1/1/10	WC-SH-G04	World of Cheese	SHOOTER	24.99	19.99
1/1/10	WC-SH-T02	World of Cheese Tee	TEE	9.99	6.99
1/1/10	PZ-SG-G05	Puppies vs. Zombies	STRATEGY	4.99	1.99
1/1/10	CU-PG-G06	Curling 2014	SPORTS	19.99	16.99
1/1/10	MB-AG-G07	Manganiello Bros.	ARCADE	39.99	24.99
1/1/10	MB-AG-T01	Manganiello Bros. Tee	TEE	9.99	6.99
1/1/10	FI-AG-G08	Orvil the Wolverine	ARCADE	39.99	24.99
1/1/10	BS-AG-G09	Benign Space Debris	ARCADE	24.99	19.99
3/1/16	DB-SG-G01	Mediocre Kingdoms	STRATEGY	26.55	21.55
3/1/16	DC-SG-G02	Dream Crusher	STRATEGY	41.55	36.55
3/1/16	FS-SG-G03	Final Sequel	STRATEGY	26.55	21.55
3/1/16	WC-SH-G04	World of Cheese	SHOOTER	26.55	21.55
3/1/16	WC-SH-T02	World of Cheese Tee	TEE	11.55	8.55
3/1/16	PZ-SG-G05	Puppies vs. Zombies	STRATEGY	5.55	2.55
3/1/16	CU-PG-G06	Curling 2014	SPORTS	21.55	18.55
3/1/16	MB-AG-G07	Manganiello Bros.	ARCADE	41.55	26.55
3/1/16	MB-AG-T01	Manganiello Bros. Tee	TEE	11.55	8.55
3/1/16	FI-AG-G08	Orvil the Wolverine	ARCADE	41.55	26.55
3/1/16	BS-AG-G09	Benign Space Debris	ARCADE	26.55	21.55

product_name	Month	price	count	sales	SubTotal Sales
Benign Space Debris	Feb	24.99	402	10,045.98	
Benign Space Debris	Mar	26.55	548	14,549.40	
Benign Space Debris Subtotal					24,595.38
Curling 2014	Feb	19.99	420	8,395.80	
Curling 2014	Mar	21.55	575	12,391.25	
Curling 2014 Subtotal					20,787.05
Dream Crusher	Feb	39.99	691	27,633.09	
Dream Crusher	Mar	41.55	852	35,400.60	
Dream Crusher Subtotal					63,033.69
Final Sequel	Feb	24.99	513	12,819.87	
Final Sequel	Mar	26.55	766	20,337.30	
Final Sequel Subtotal					33,157.17

Configuring Time-based Lookups

1. Specify the name of the time field in the lookup
2. Enter the strptime format of the time field
3. Define the minimum offset in seconds
   - Default is 0
4. Define the maximum offset in seconds
   - There is no maximum offset by default

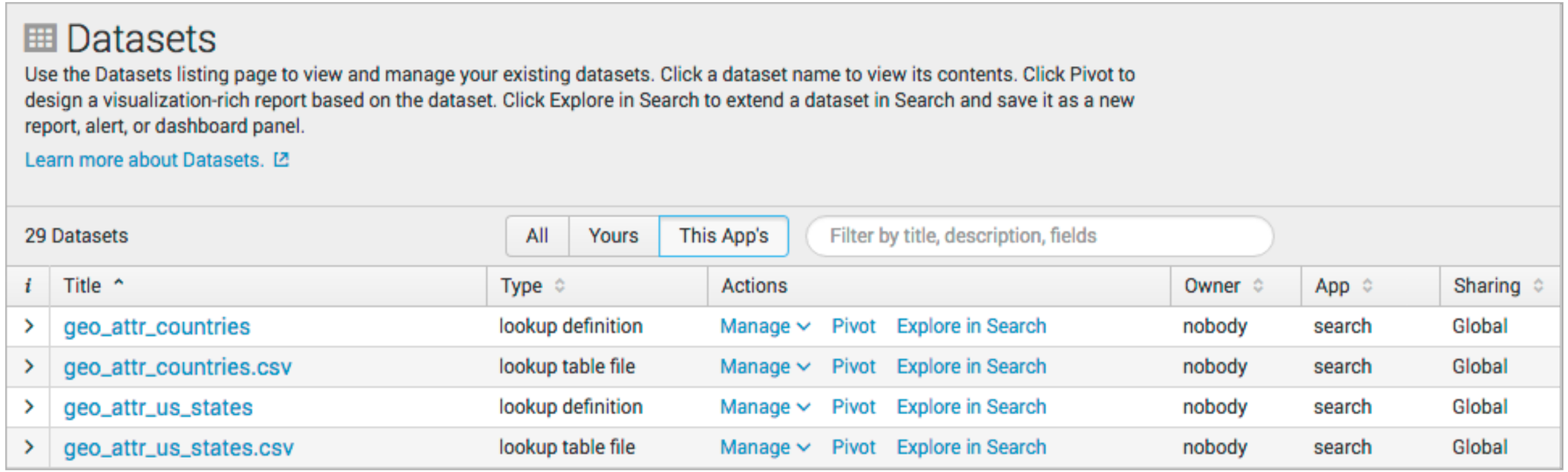


Note

The offset is the minimum and maximum amounts of time that an event may be ahead of a lookup entry.

Using the Lookup as a Dataset

- A lookup is categorized as a dataset
  - Manage
  - Pivot
  - View the lookup in a search (**`inputlookup`**)



Additional Lookup Options

In addition to creating and using a file-based lookup, you can also:

- Populate a lookup table with search results
  - **`outputlookup`** is discussed in more detail in the *Advanced Searching & Reporting* class
- Define a field lookup based on an external command; Python- and binary-based scripts
  - For more information, see the *Knowledge Manager Manual*

  docs.splunk.com/Documentation/Splunk/latest/Knowledge/Addfieldsfromexternaldatasources
- Use the Splunk DB Connect app to create lookups with data from external SQL databases

Additional Lookup Options (cont.)

- Use Geospatial lookups to create queries that can be used to generate choropleth map visualizations

  http://docs.splunk.com/Documentation/Splunk/latest/Knowledge/Configuregeospatiallookups

- Populate events with fields from an App Key Value Store (KV Store) collection
  - KV Store lookups can only be invoked through REST endpoints or by using search commands such as lookup, inputlookup, and outputlookup; therefore, cannot be set up as automatic
  - For more information, see the *Knowledge Manager Manual*

  http://docs.splunk.com/Documentation/Splunk/latest/Knowledge/ConfigureKVstorelookups

Module 13: Creating Scheduled Reports and Alerts

Module Objectives

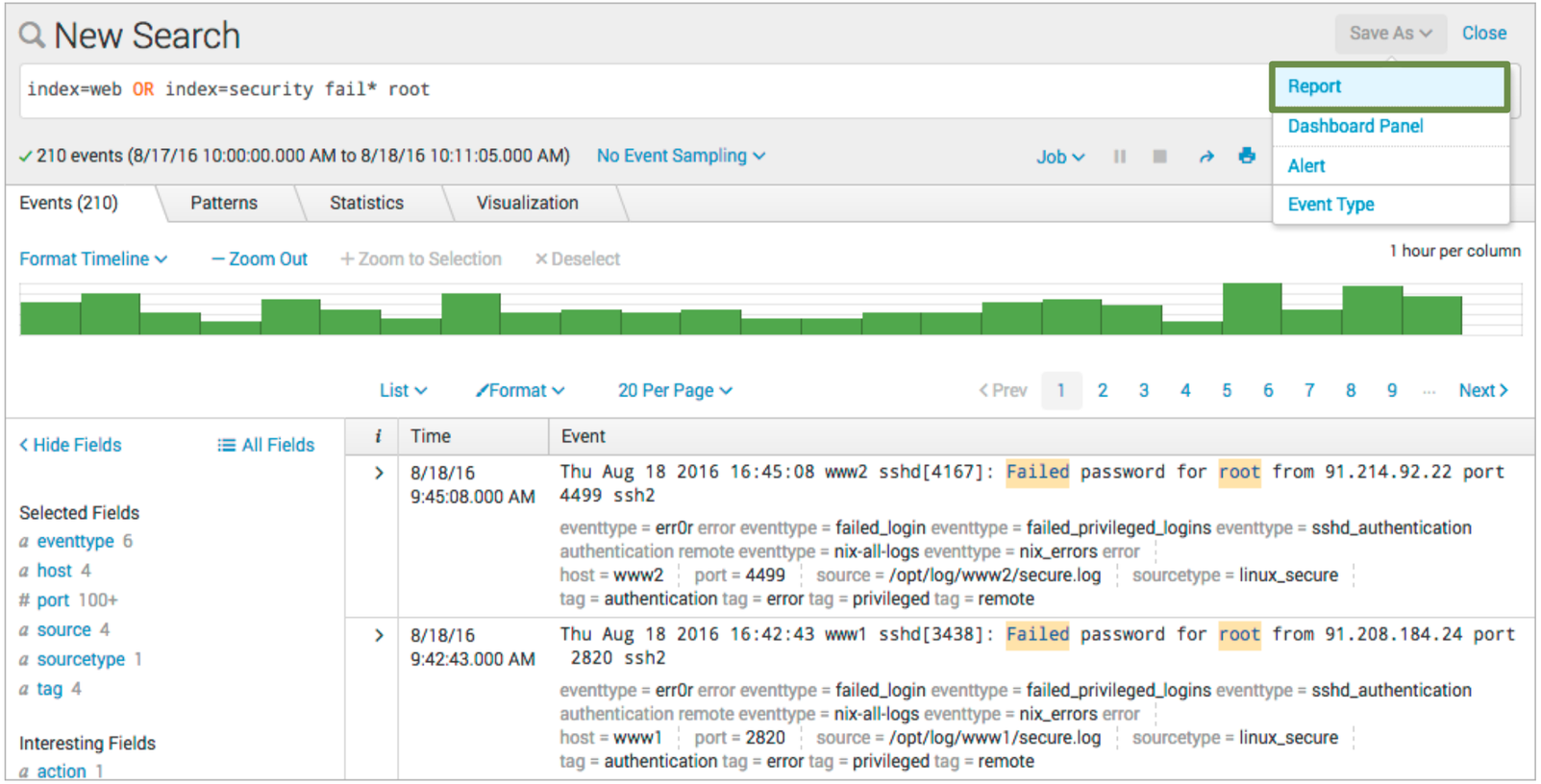
- Describe scheduled reports and alerts
- Create scheduled reports and alerts
  - Run the underlying search
  - Set the schedule, conditions, and actions
- View fired, scheduled reports and alerts

Using Scheduled Reports

- Scheduled Reports are useful for:
  - Monthly, weekly, daily executive/managerial roll up reports
  - Dashboard performance
  - Automatically sending reports via email

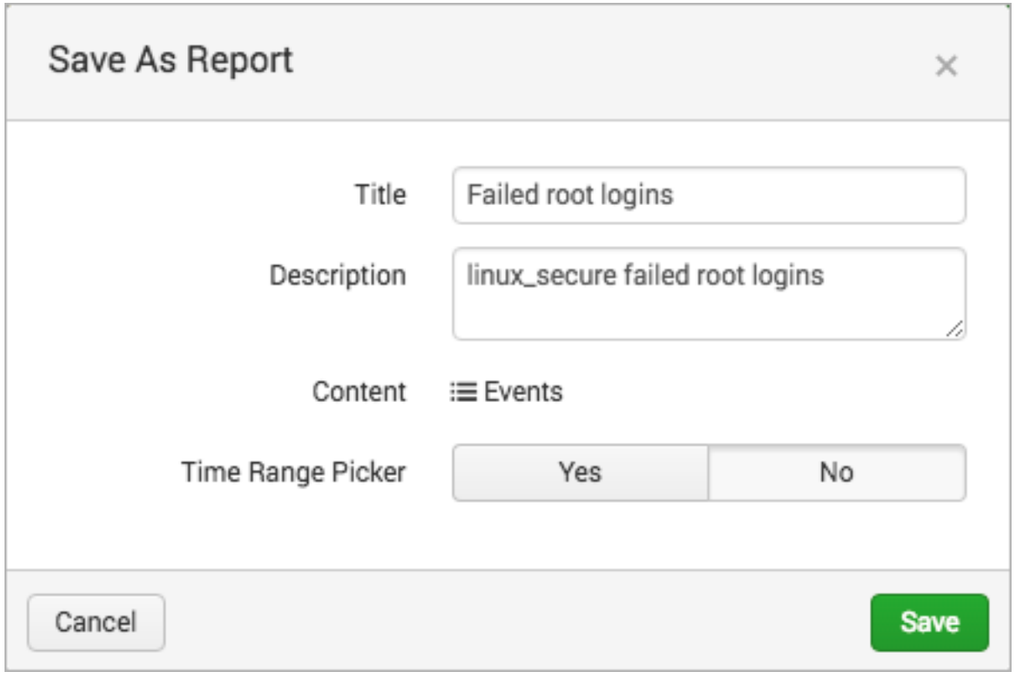
Creating a Scheduled Report

- Create your search
- From the **Save As** menu, select **Report**



Creating a Scheduled Report (cont.)

- **Title** – enter a title for your report
- **Description** – provide a description
- **Time Range Picker** – you can add a time range picker to the report
- Click **Save**

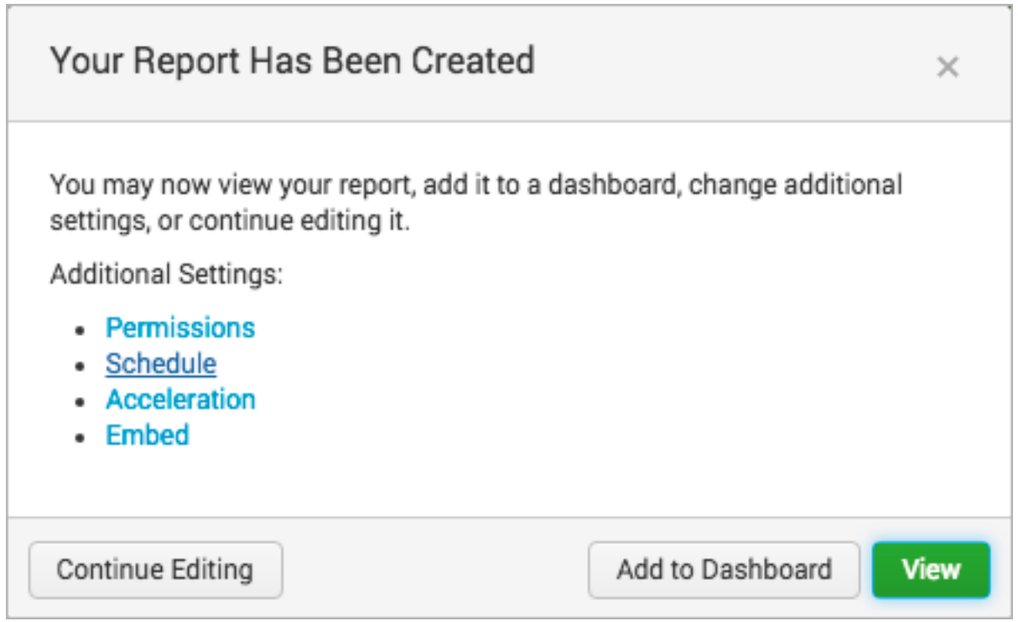


Note

When you schedule a report, the Time Range Picker will not be available.

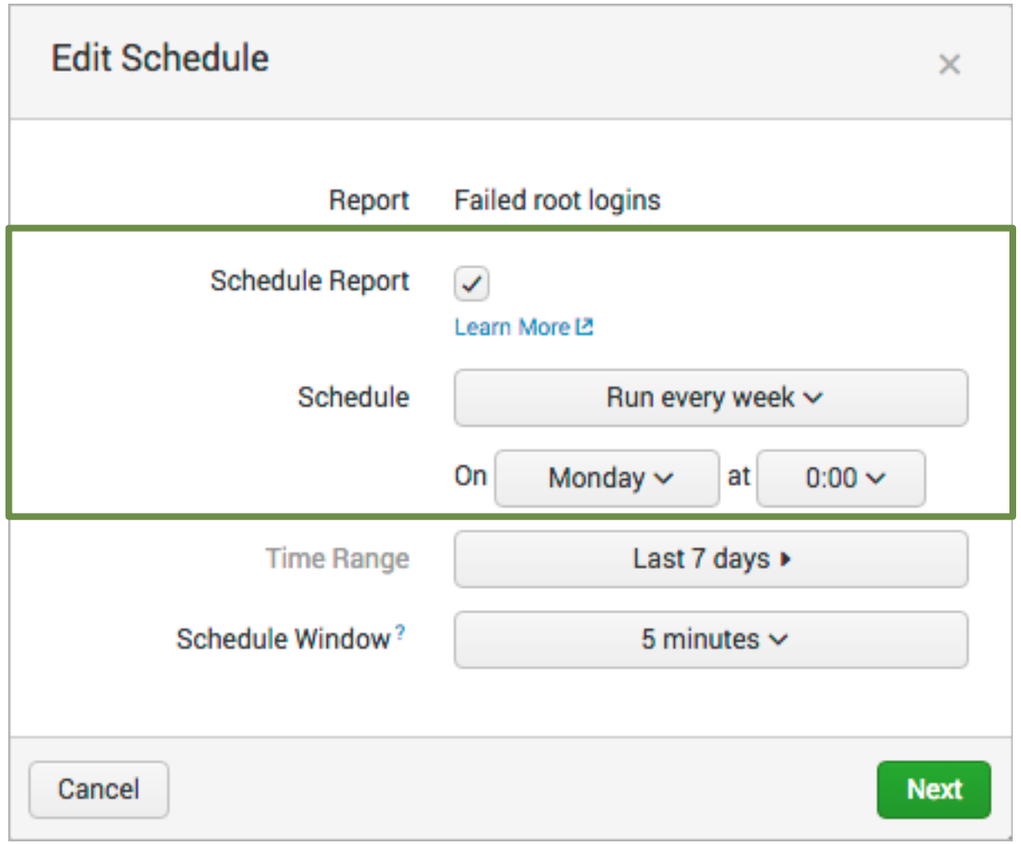
Creating a Scheduled Report (cont.)

After the report is created, click **Schedule**



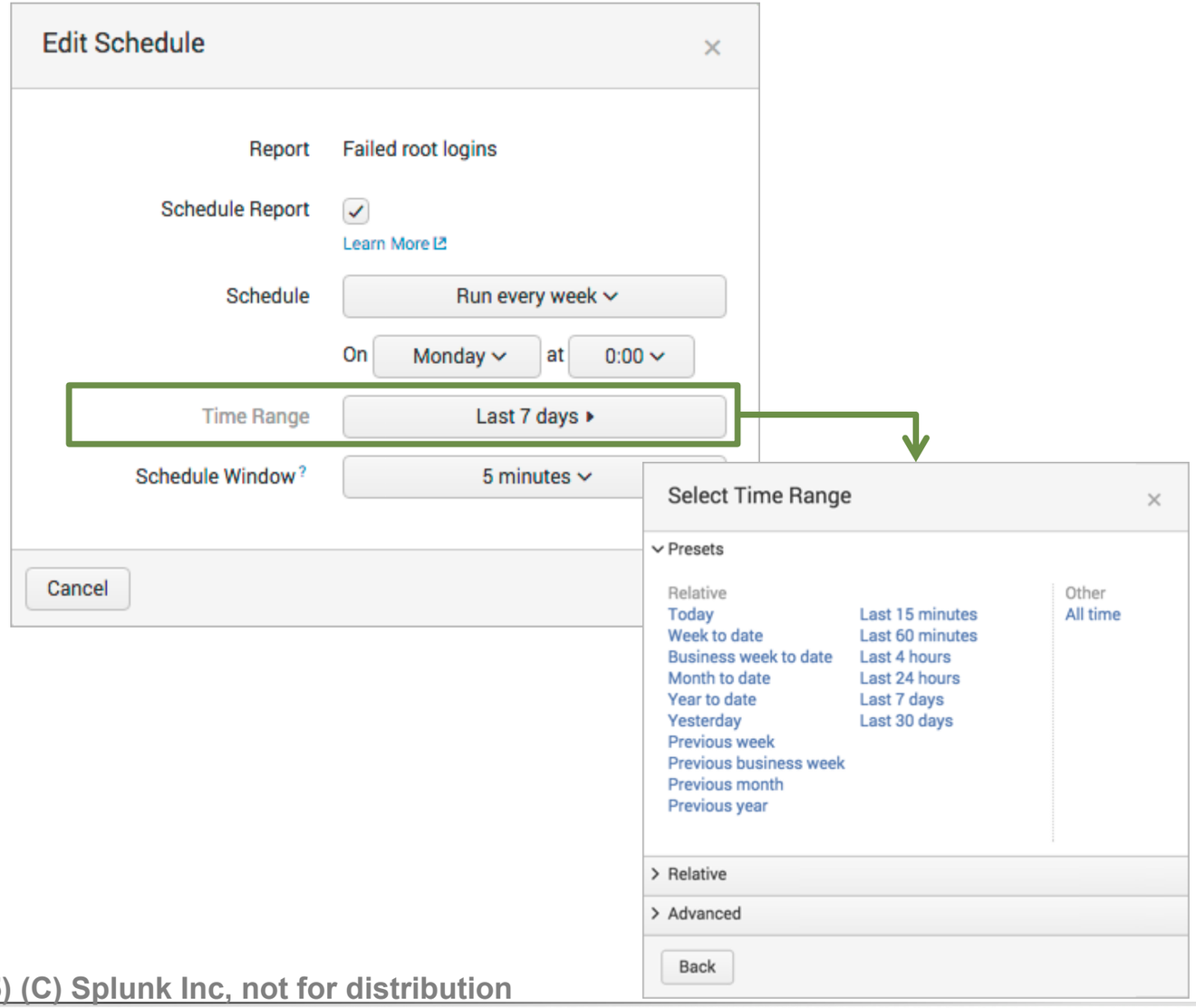
Creating a Scheduled Report – Define Schedule

- **Schedule Report** – select this checkbox
- **Schedule** – select the frequency to run the report
  - Run every hour
  - Run every day
  - Run every week
  - Run every month
  - Run on Cron Schedule



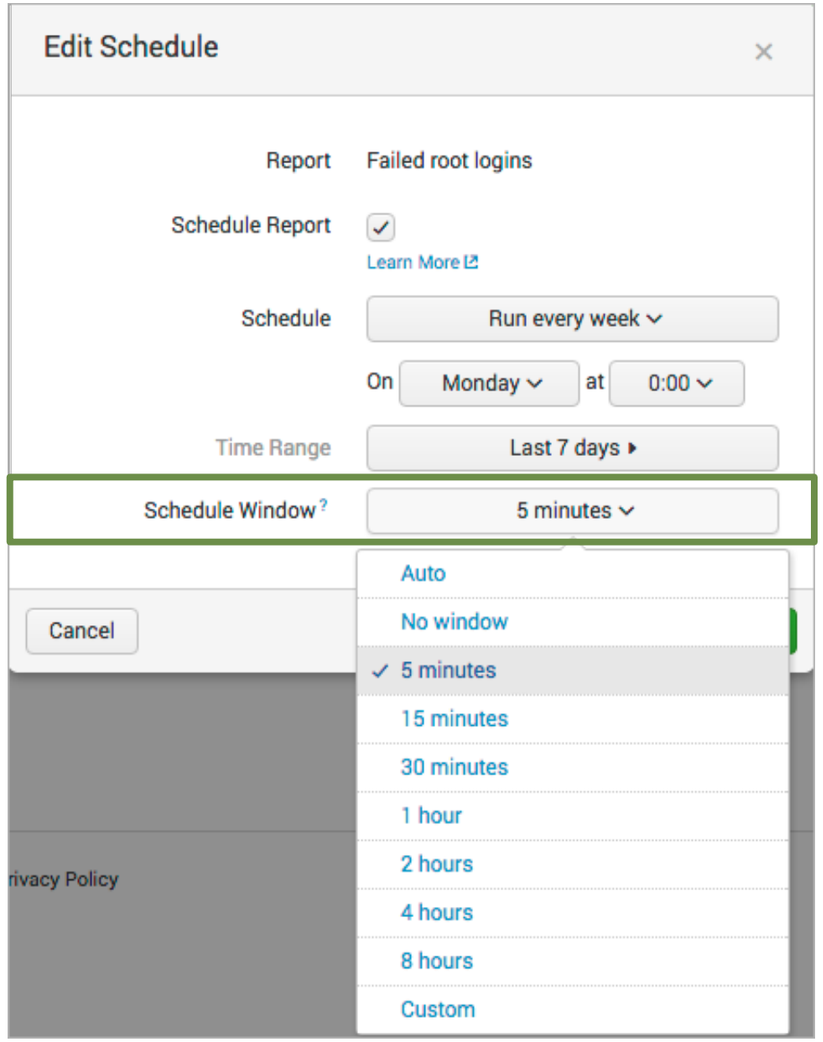
Creating a Scheduled Report – Select Time Range

- **Time Range** – By default, the search time range is used
  - Click the time range button to change the time range
  - You can select a time range from Presets, Relative, or Advanced
  - Typically, the time range is relative to the Schedule



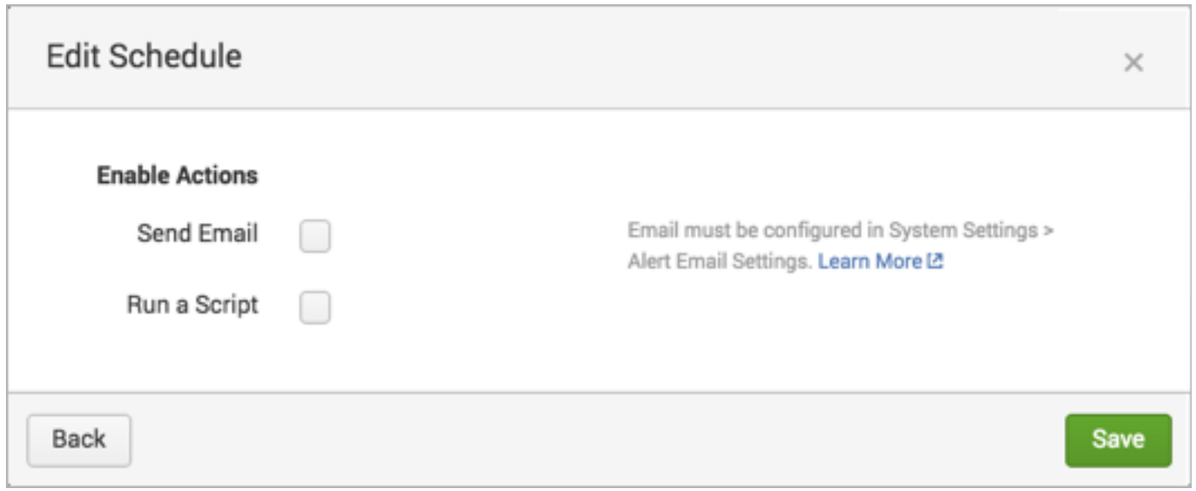
Creating a Scheduled Report – Schedule Window

- **Schedule Window** – This setting determines a time frame to run the report
  - If there are other reports scheduled to run at the same time, you can provide a window in which to run the report
  - This setting provides efficiency when scheduling several reports to run
- After you configure the schedule report, click **Next**



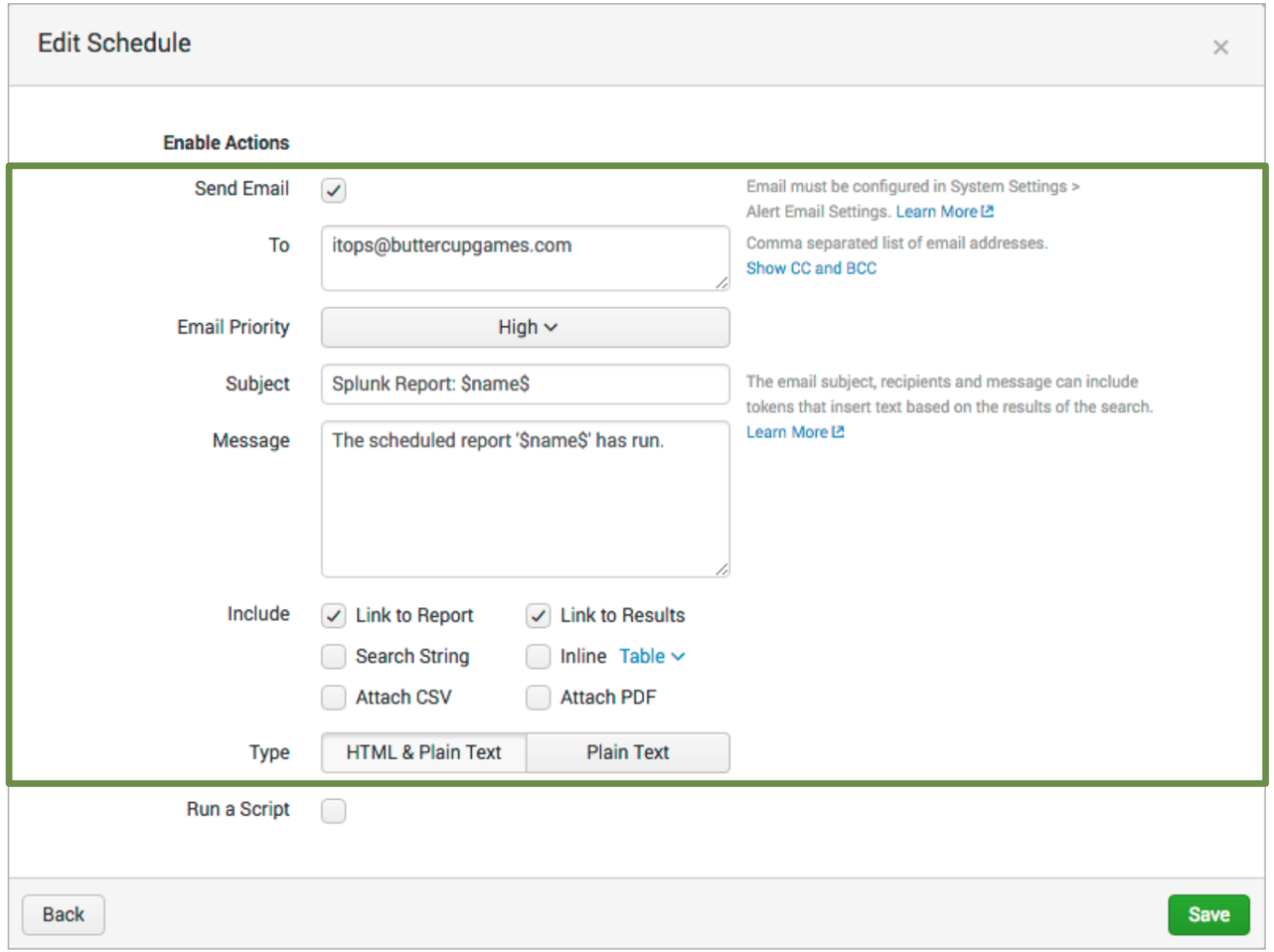
Creating a Scheduled Report – Enable Actions

- **Enable Actions**
  - **Send Email**: When a report runs, an email is sent to the specified recipient(s)
  - **Run a script**: A script is launched when a report runs



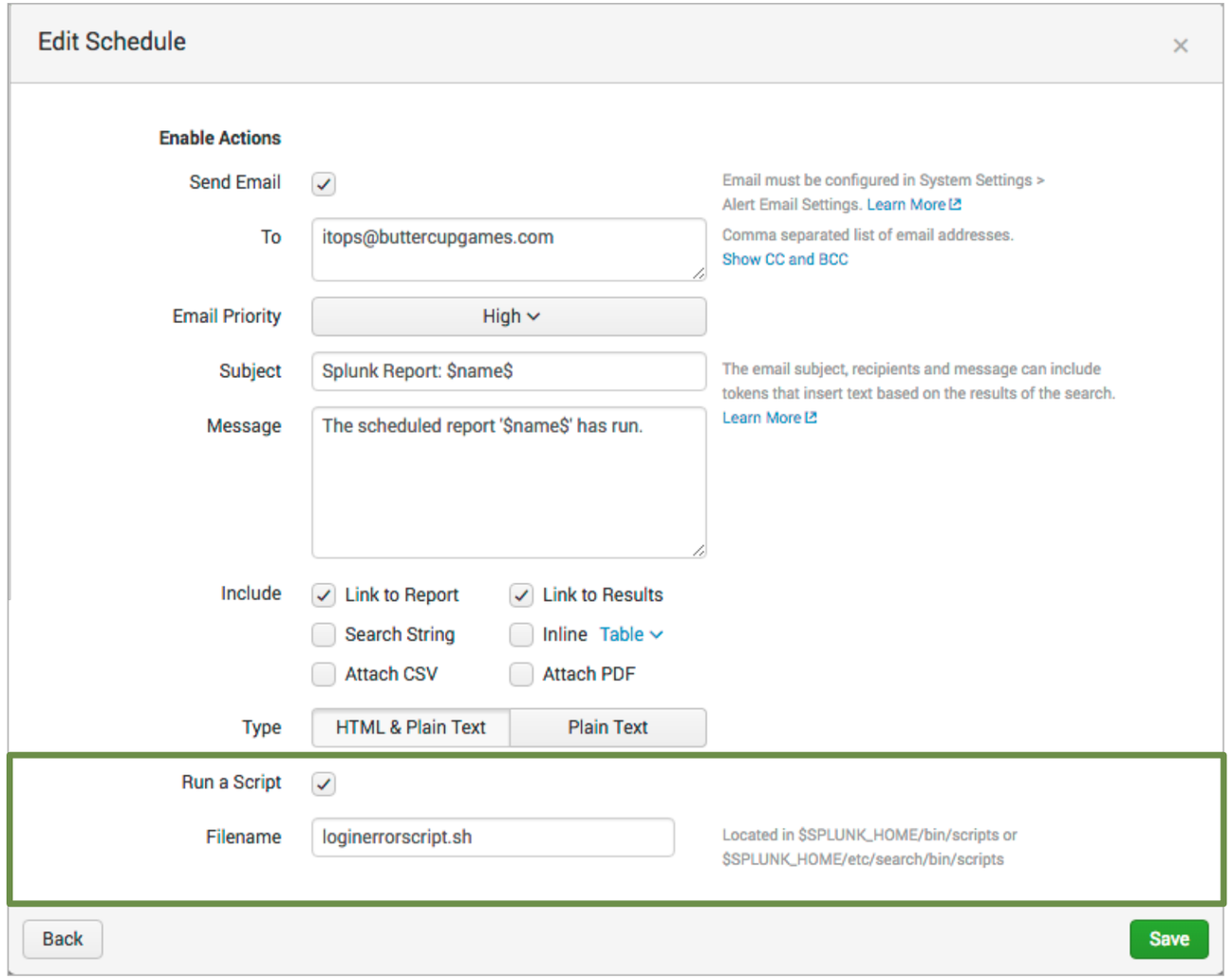
Creating a Scheduled Report – Send Email

1. Enter addresses in the **To** field, separated by a comma
2. Set the priority
3. Edit or keep the default subject
   - The $name$ variable includes the name of the report
   - In addition to a message, you can include other options like an inline table of the results, etc.
4. Define the email text type
5. After you have configured the actions, click **Save**



Creating a Scheduled Report – Run a Script

1. Enter the file name of the script
   - The script must reside in the **`$SPLUNK_HOME/bin/scripts`** directory
2. Click **Save**

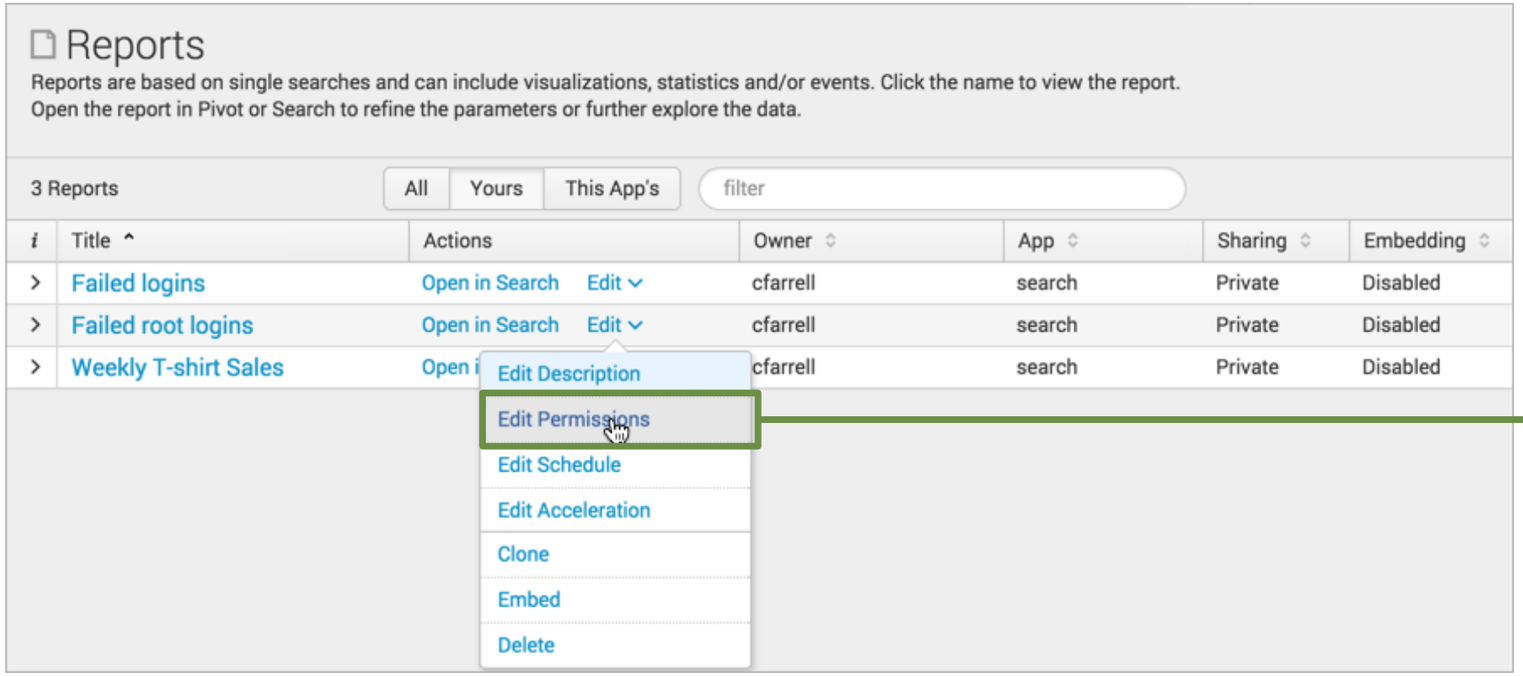


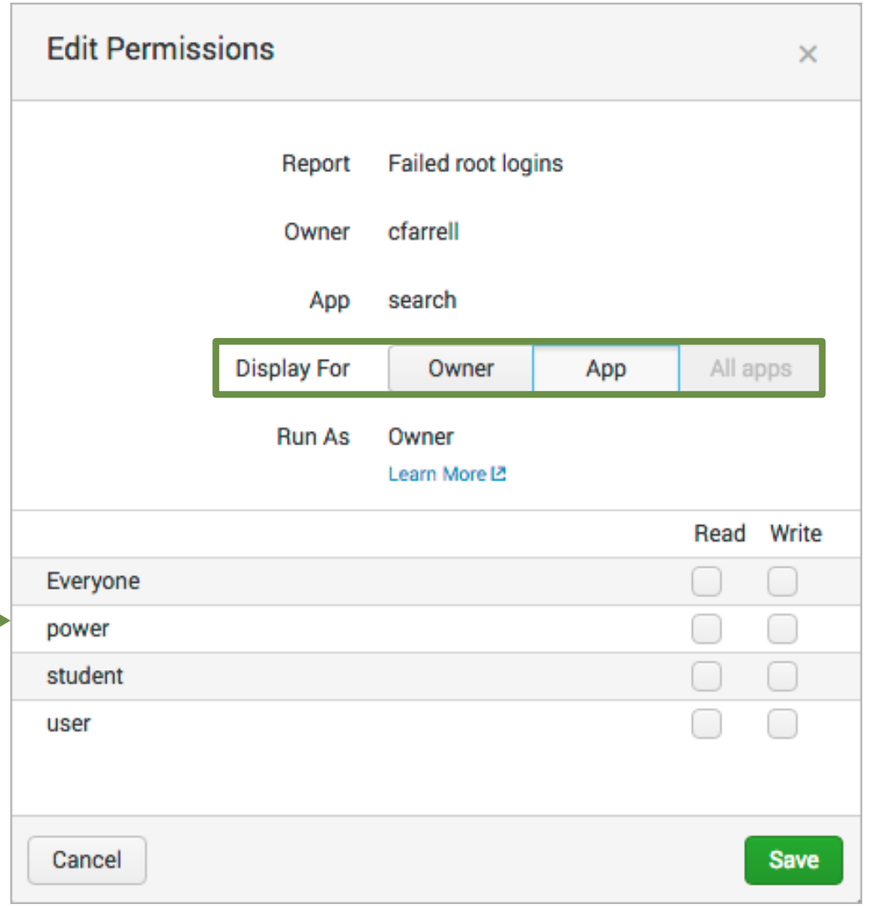
Note

The proper permissions for your Splunk server are required to upload your script to this Splunk directory.

Managing Reports – Edit Permissions

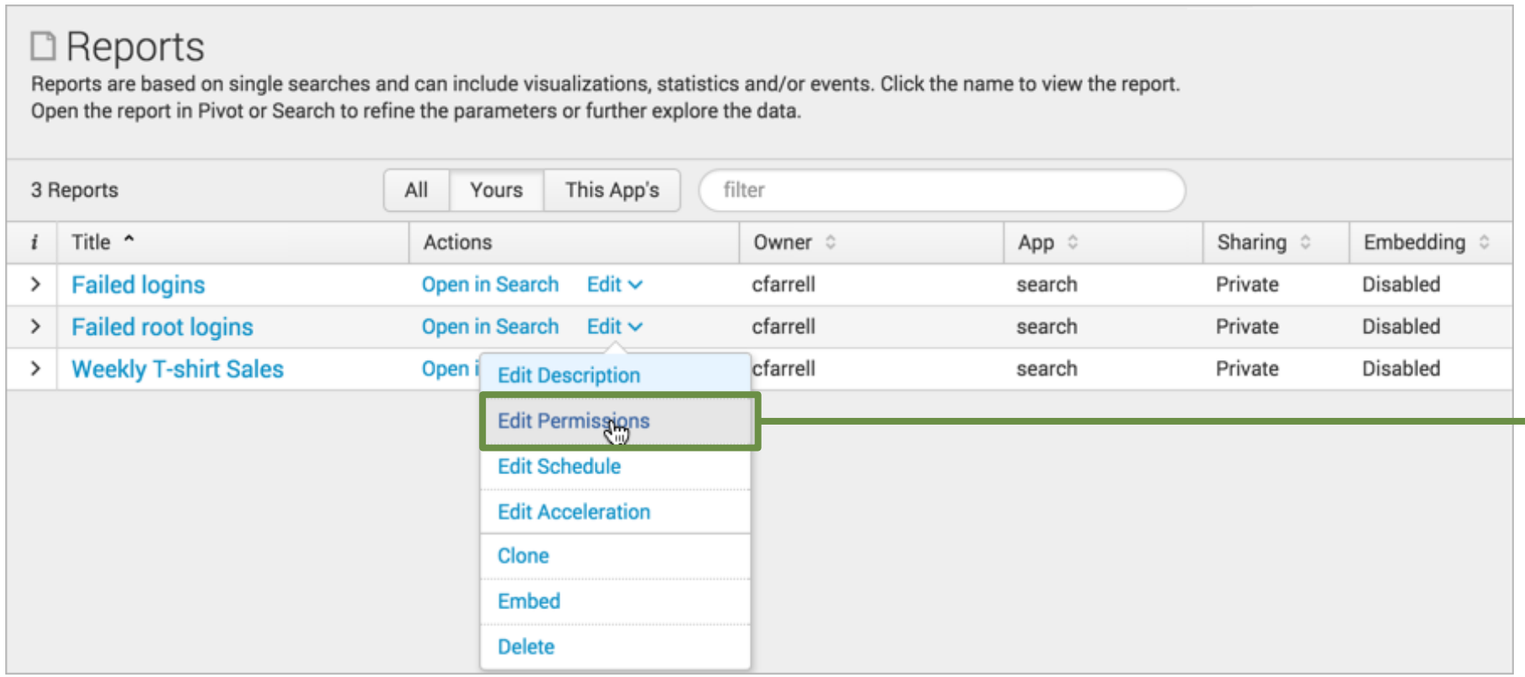
Display For determines who sees the scheduled report

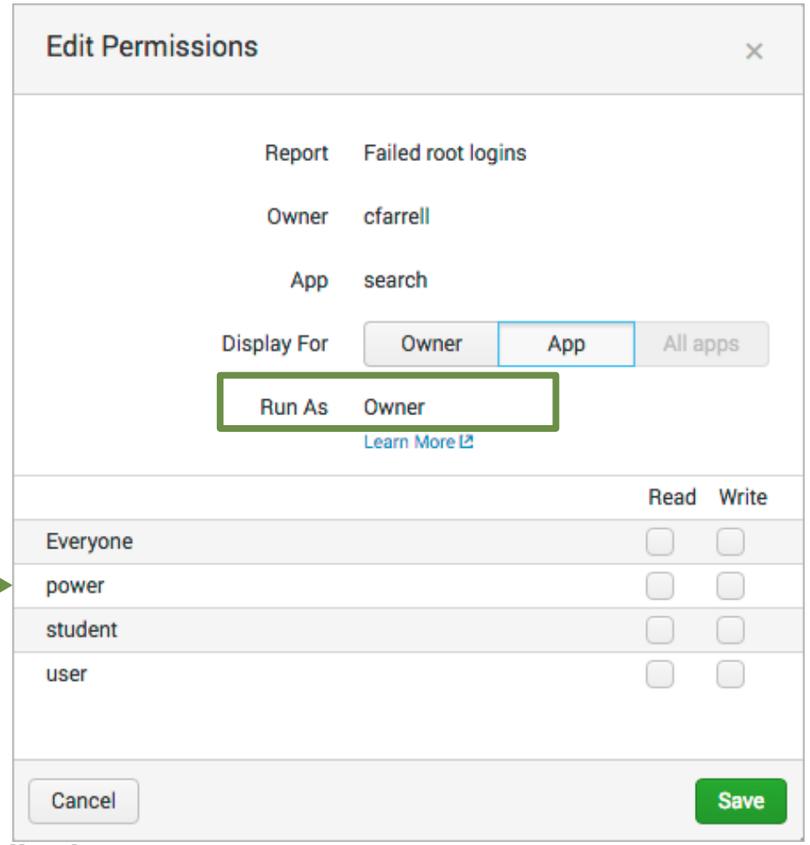




Managing Reports – Edit Permissions (cont.)

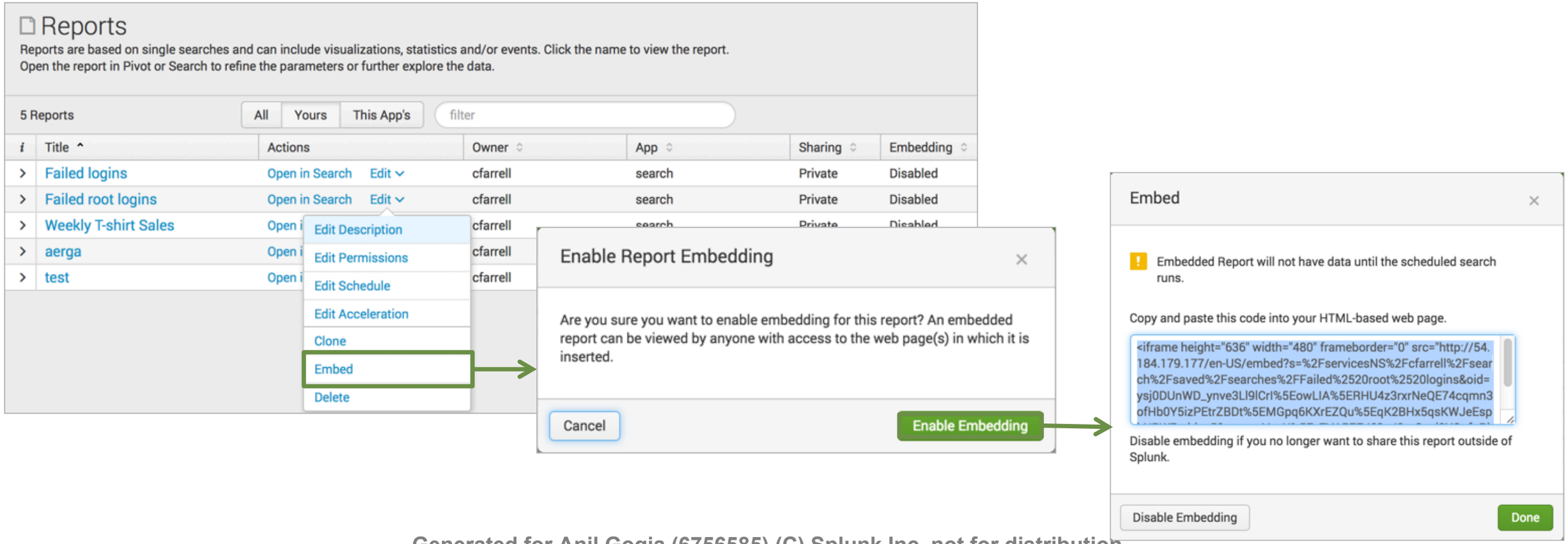
- **Run As** – determines which user profile is used at run time
  - Owner – all data accessible by the owner appears in the report
  - User – only data allowed to be accessed by the user role appears





Managing Reports – Embed

- To access the report results from a webpage, click **Edit** > **Embed**
  - Before a report can be embedded, it must be scheduled

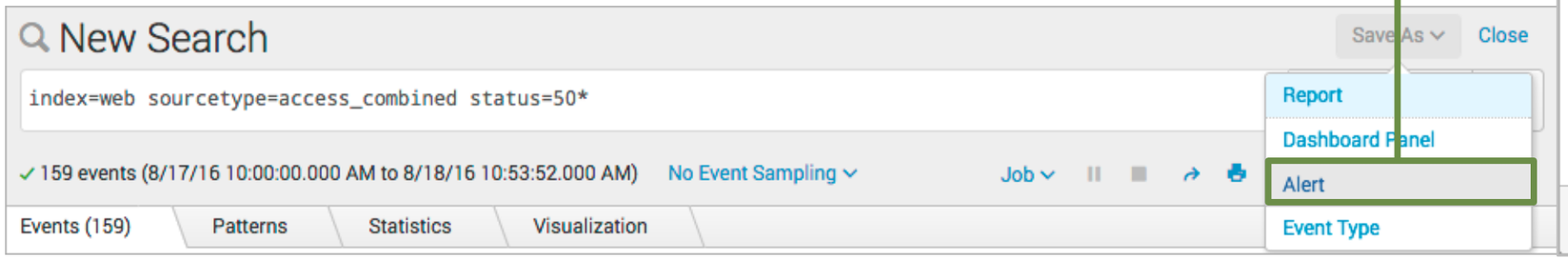


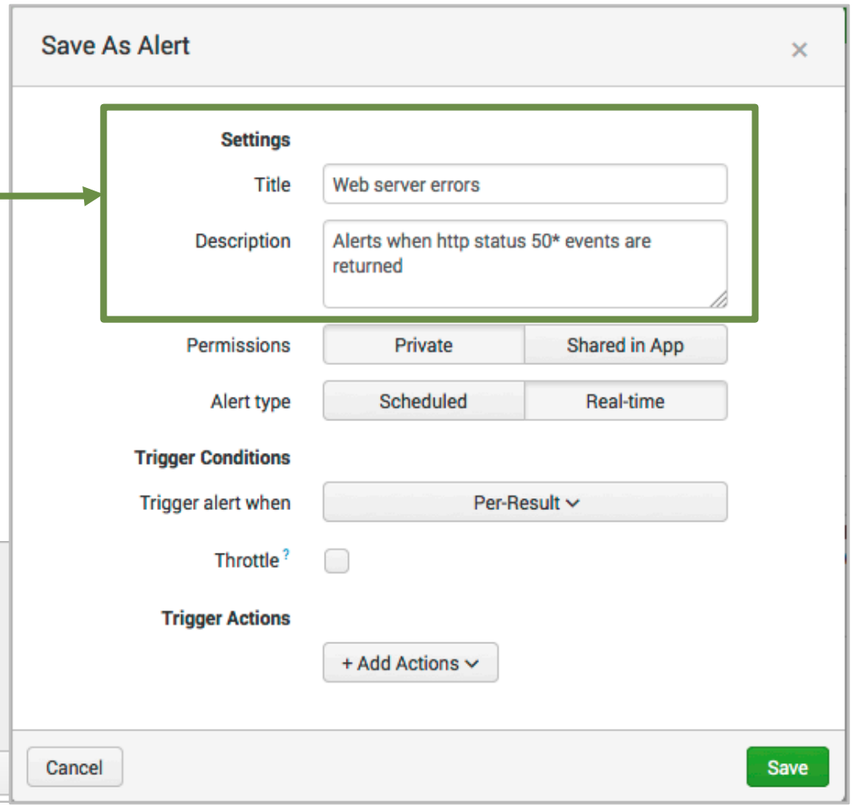
Alerting Overview

- Splunk alerts are based on searches that can run either:
  - On a regular **scheduled interval**
  - In **real-time**
- Alerts are triggered when the results of the search meet a specific condition that you define
- Based on your needs, alerts can:
  - List in triggered alerts
  - Send emails
  - Trigger scripts
  - Use a webhook
  - Run a custom alert

Creating an Alert

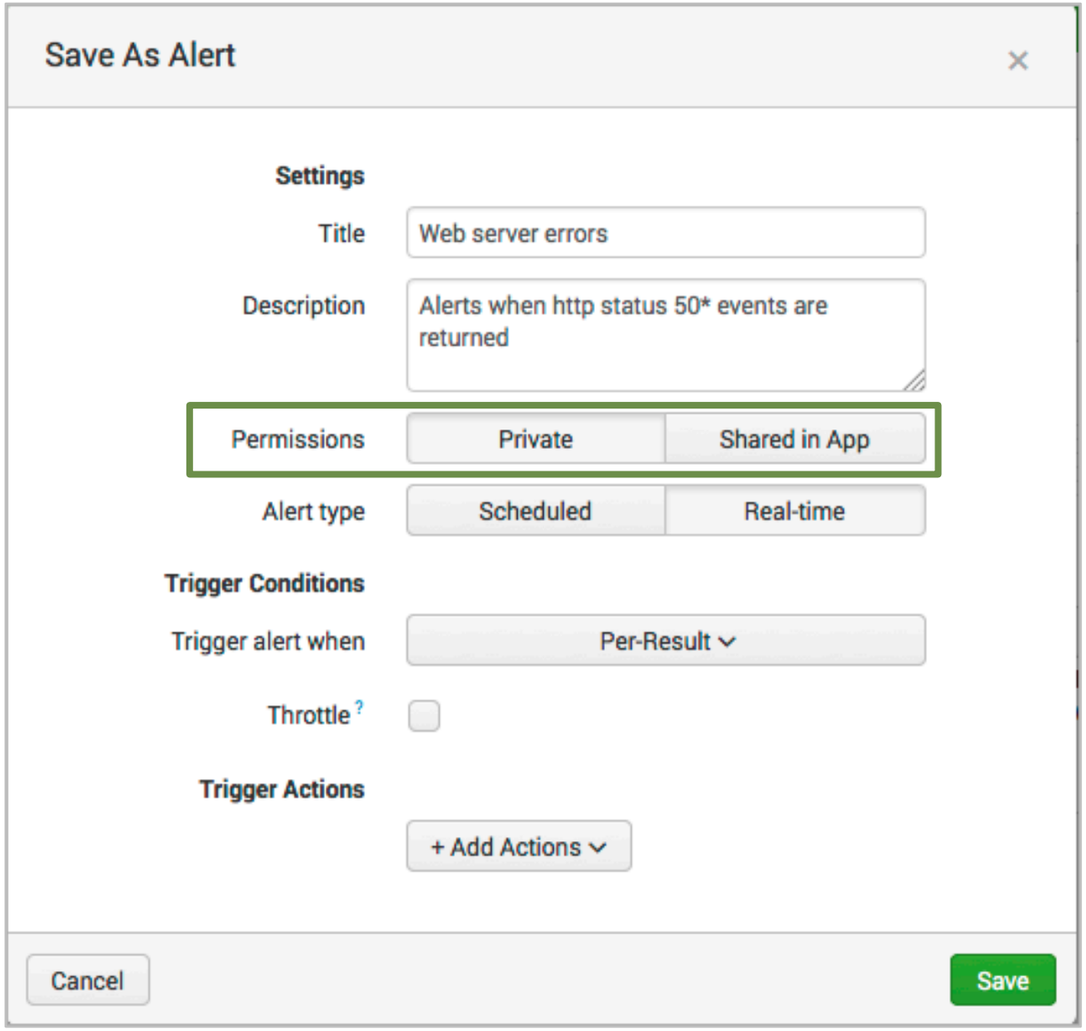
- Run a search
  - In this example, you're searching for server errors: any http request status that begins with 50 over the last 5 minutes
- Select **Save As** > **Alert**
- Give the alert a Title and Description





Setting Alert Permissions

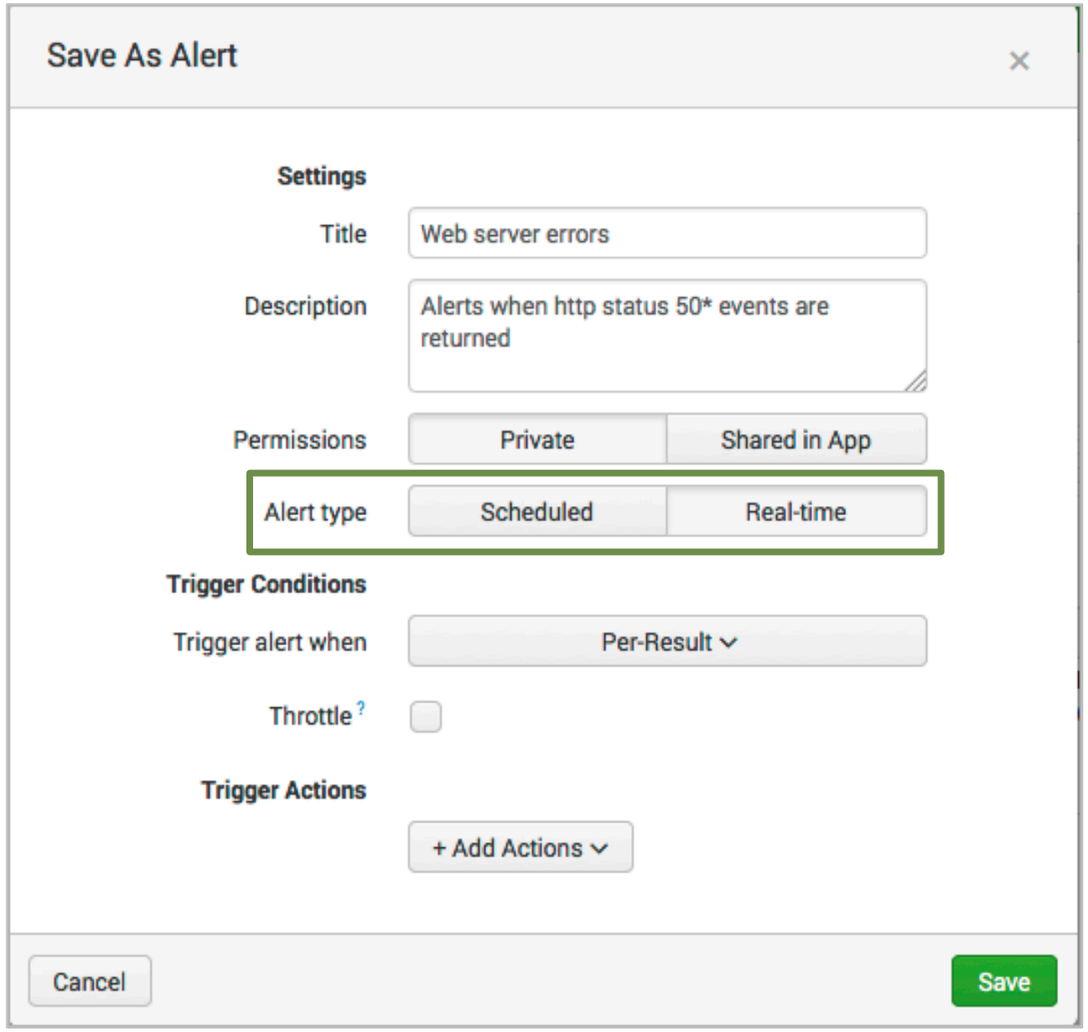
- Set the alert permissions
  - **Private** – only you can access, edit, and view triggered alerts
  - **Shared in app**
    - All users of the app can view triggered alerts
    - By default, everyone has read access and power has write access to the alert



Choosing Real Time or Scheduled Alert Type

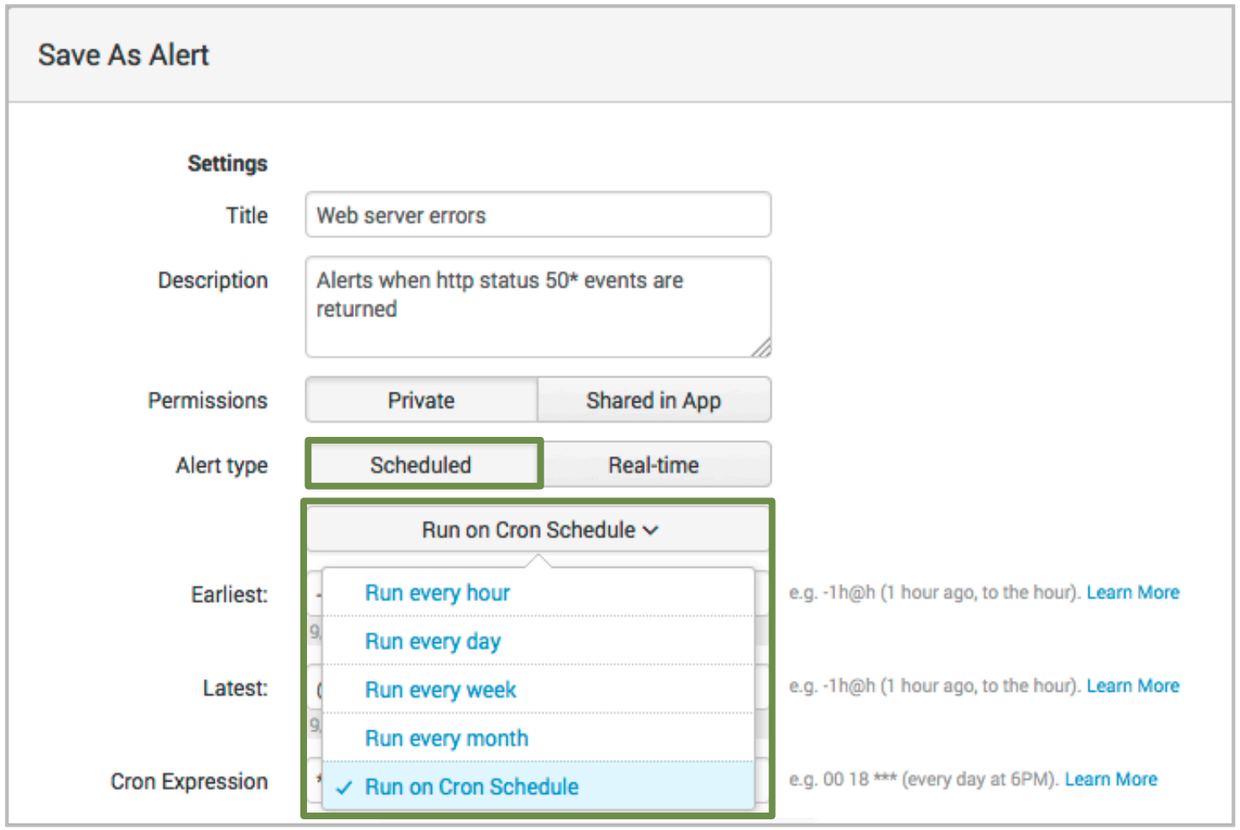
Choose an **Alert type** to determine how Splunk searches for events that may match your alert

- **Scheduled** alerts
  - Search runs at a defined interval
  - Evaluates trigger condition when the search completes
- **Real-time** alerts
  - Search runs constantly in the background
  - Evaluates trigger conditions within a window of time based on the conditions you define



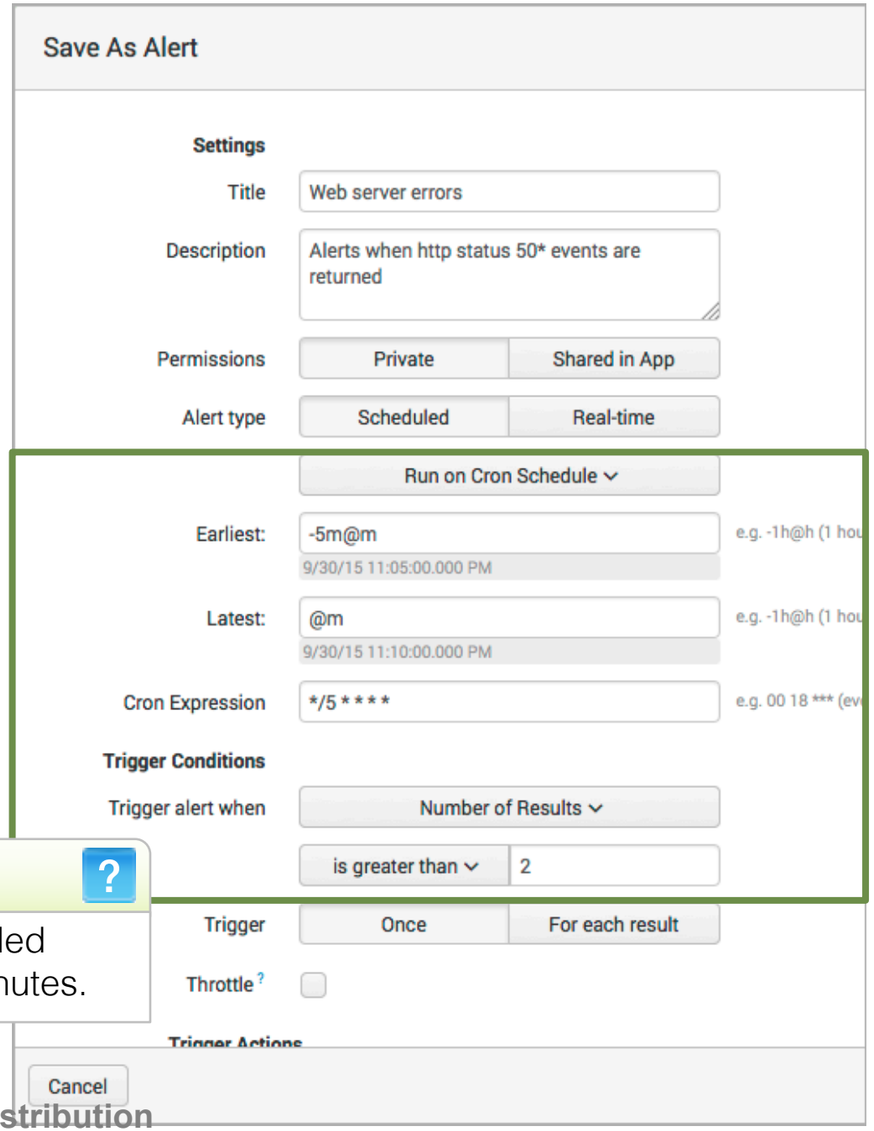
Setting the Alert Type – Scheduled

- From the frequency menu, choose to run the search every hour, day, week, month, or on a cron schedule
  - For the scheduled interval options, select the time the search will run
  - For cron schedule, define the cron expression



Setting Trigger Conditions – Scheduled

- For the cron schedule, enter the **earliest** and **latest** values to define the time range of the results
- Set trigger conditions for scheduled alerts (same steps outlined for real-time alerts)
  - The alert examines the complete results set after the search is run

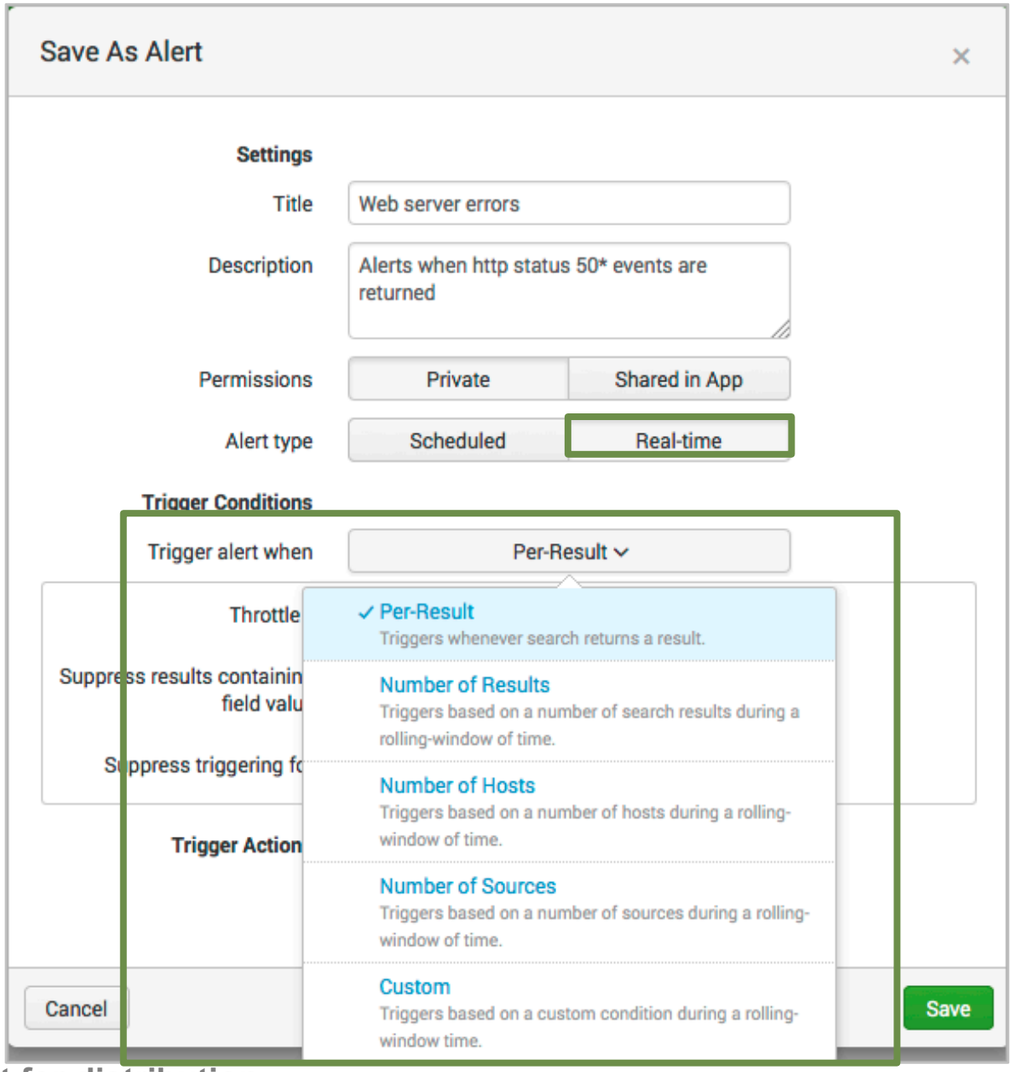


Scenario

In this example, a scheduled search will run every 5 minutes.

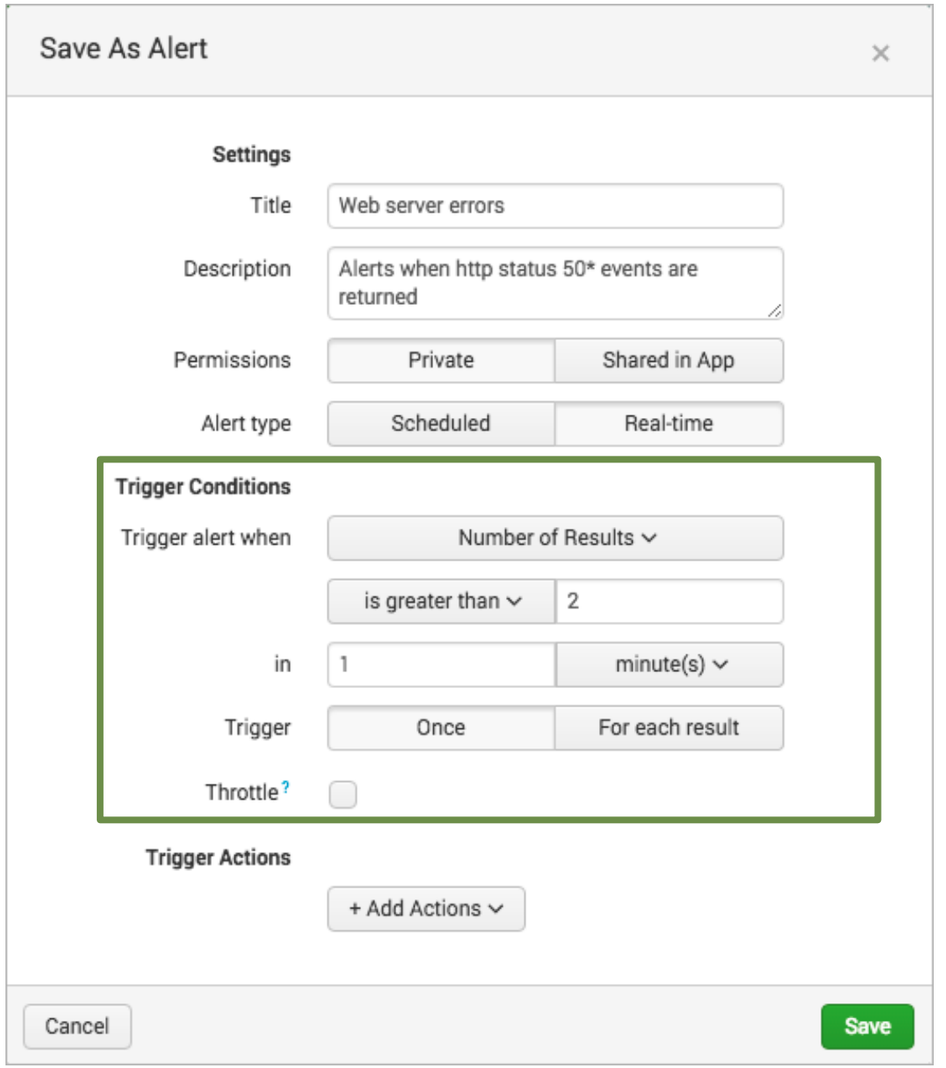
Setting Trigger Conditions – Real-time

- Trigger conditions allow you to capture a larger data set, then apply more stringent criteria to results before executing the alert
- You can set alerts to trigger:
  - **Per Result –** triggers when a result is returned
  - **Number of Results** – define how many results are returned before the alert triggers
  - **Number of Hosts** – define how many unique hosts are returned before the alert triggers
  - **Number of Sources** – define how many unique sources are returned before the alert triggers
  - **Custom** – define custom conditions using the search language



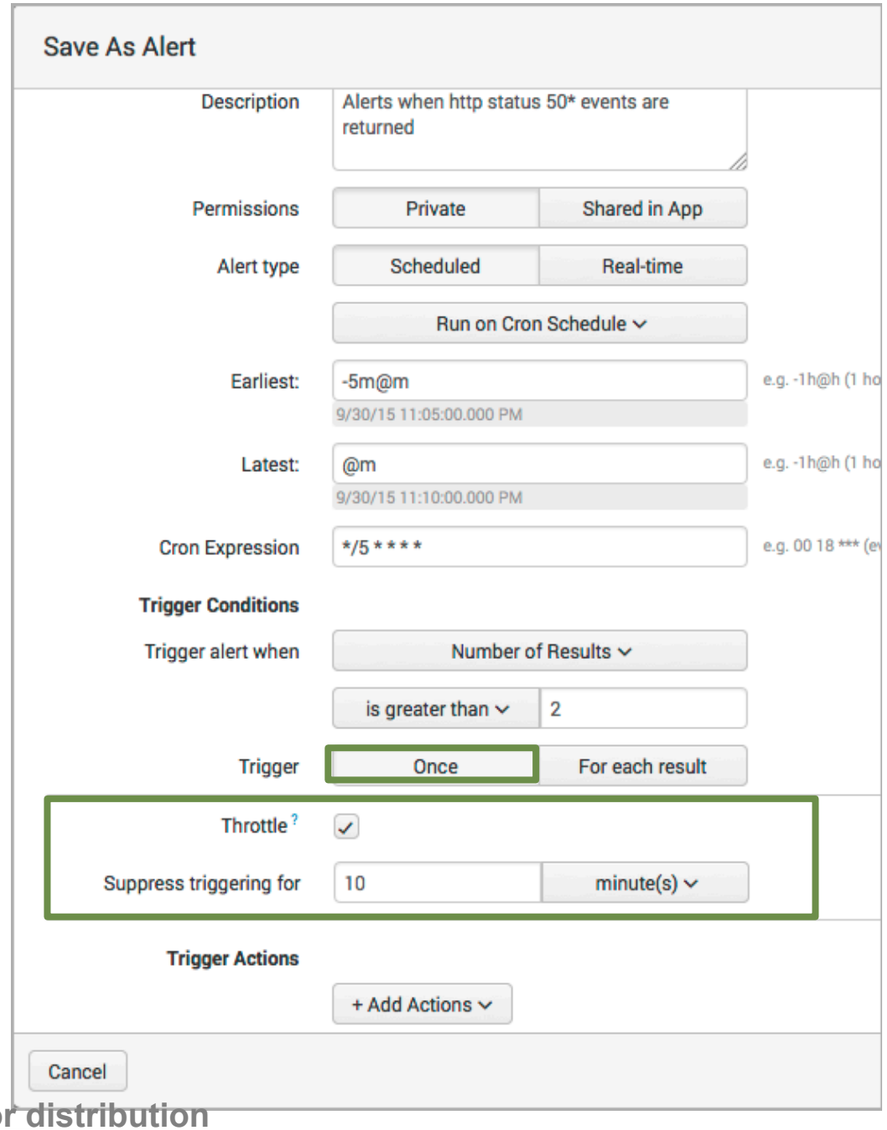
Setting Trigger Conditions – Real-time (cont.)

- In this example, the trigger condition is set to **Number of Results**
- In this **Real Time** alert example, if the number of results is greater than **2** within **1** minute, the alert triggers



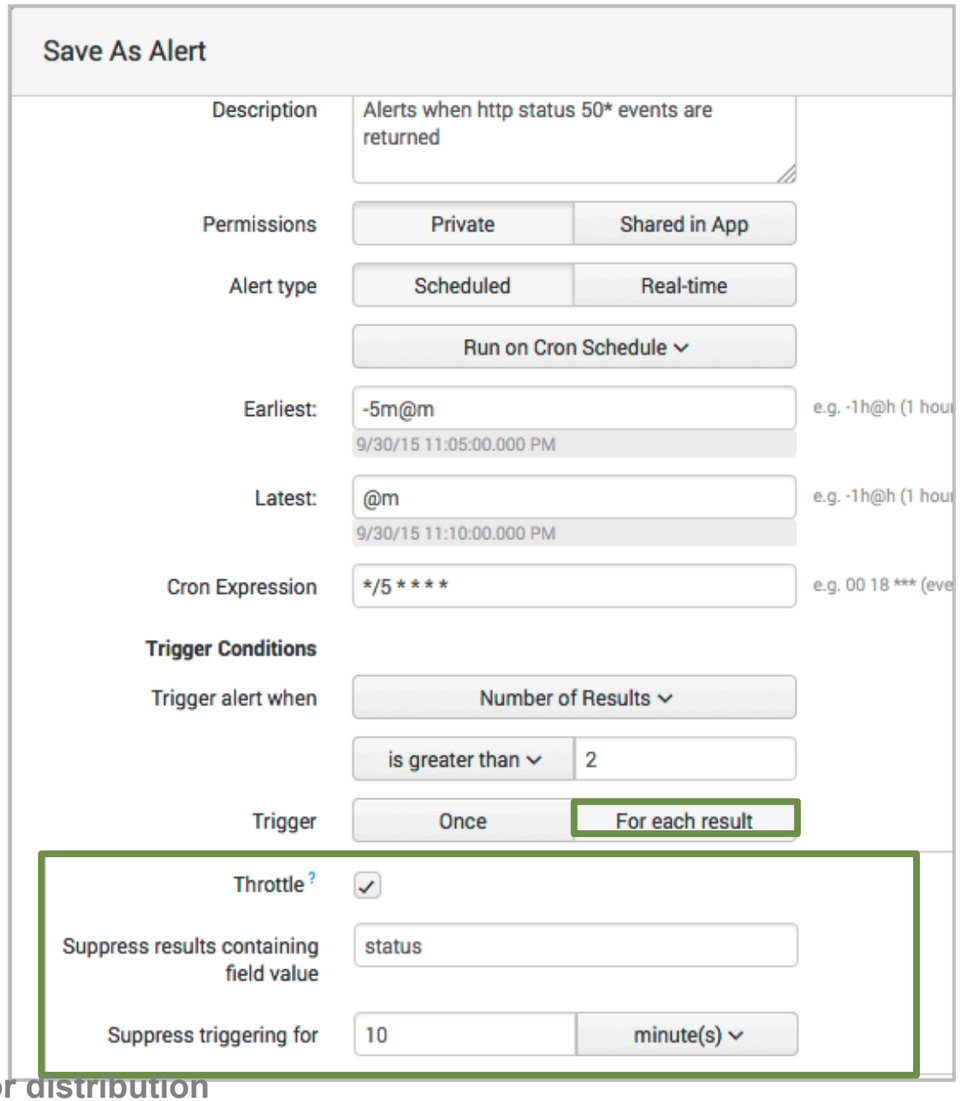
Alert Actions – Trigger Conditions: Once

- **Once** executes actions *one time* for all matching events within the scheduled time and conditions
  - Example: If your alert is scheduled to run every 5 minutes, and 40 results are returned, the alert only triggers and executes actions one time
- Select the **Throttling** option to suppress the actions for results within a specified time range



Alert Actions – Trigger Conditions: For Each Result

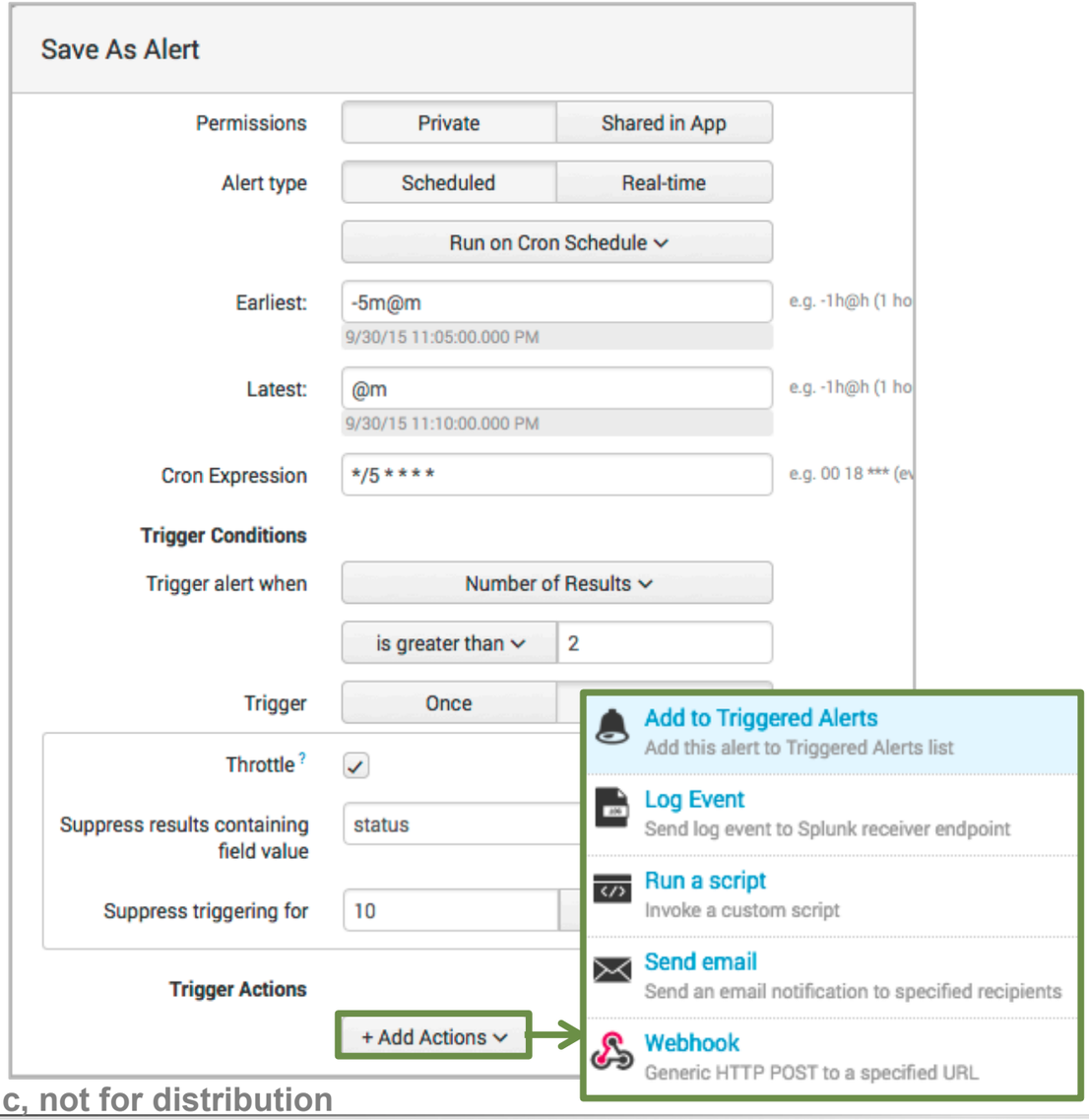
- **For each result** – executes the alert actions once *for each result* that matches the conditions
- Select the **Throttling** option to suppress the actions for results that have the same field value, within a specified time range
  - Certain situations can cause a flood of alerts, when really you only want one
- In this example, the search runs every 5 minutes:
  - 70 events are returned in a 5 minute window
  - 50 events with status=**500** and 20 include status=**503**
  - 2 actions will trigger, once for each status



Add Trigger Actions

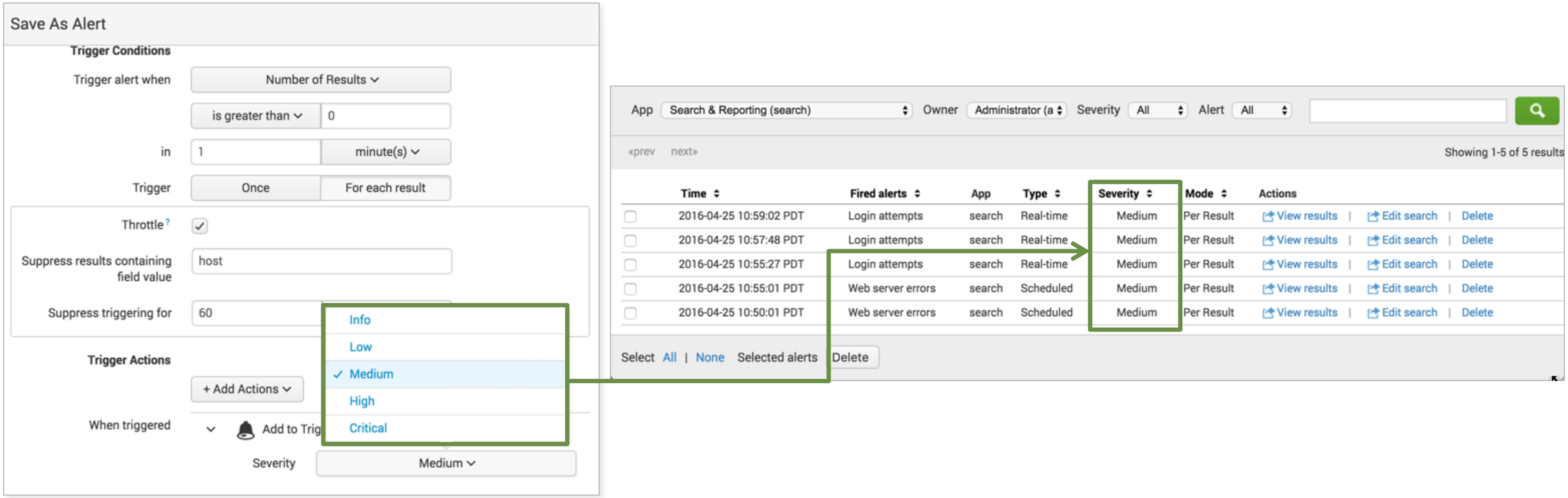
Add Actions

- **Add to Triggered Alerts** – adds the alert to the *Activity > Triggered alerts*
- **Log Event –** creates a log event to index and search
- **Run a script** – runs a script that can perform some other action
- **Send Email** – sends an email with results to recipients that you define
- **Webhook** – calls a rest endpoint using http post request



Alert Actions – Triggered Alerts

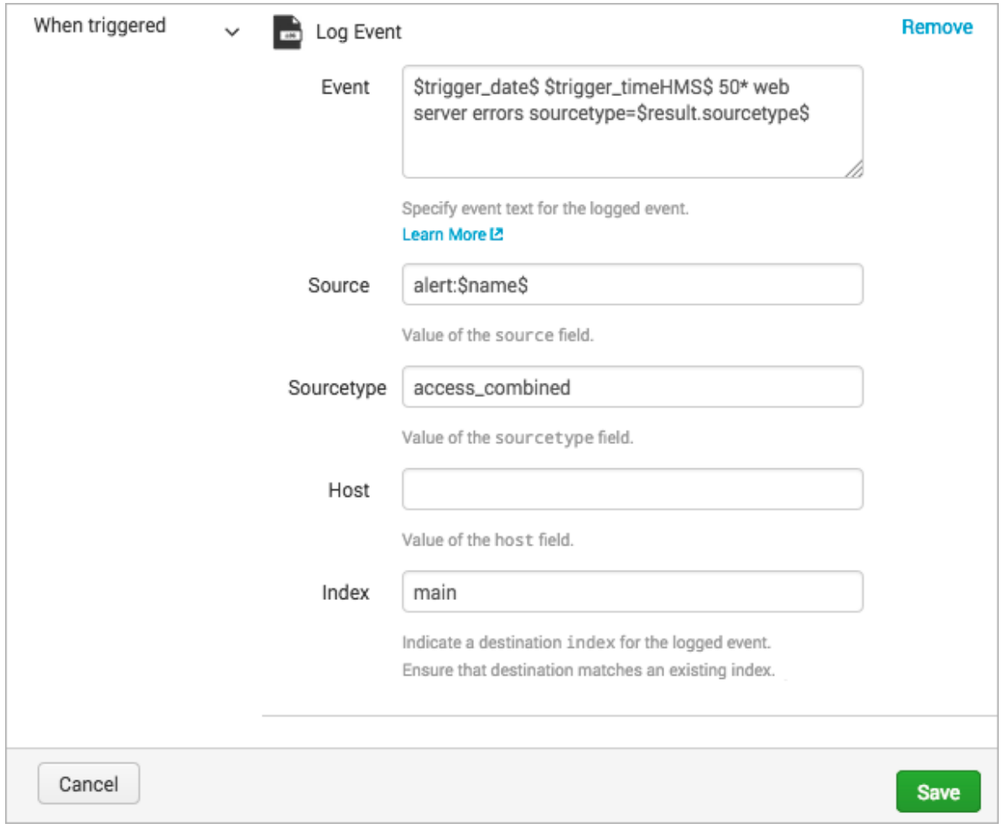
Choose an appropriate severity for the alert



Alert Actions – Log Event

If you have *administrator privileges*, you can use a log event action

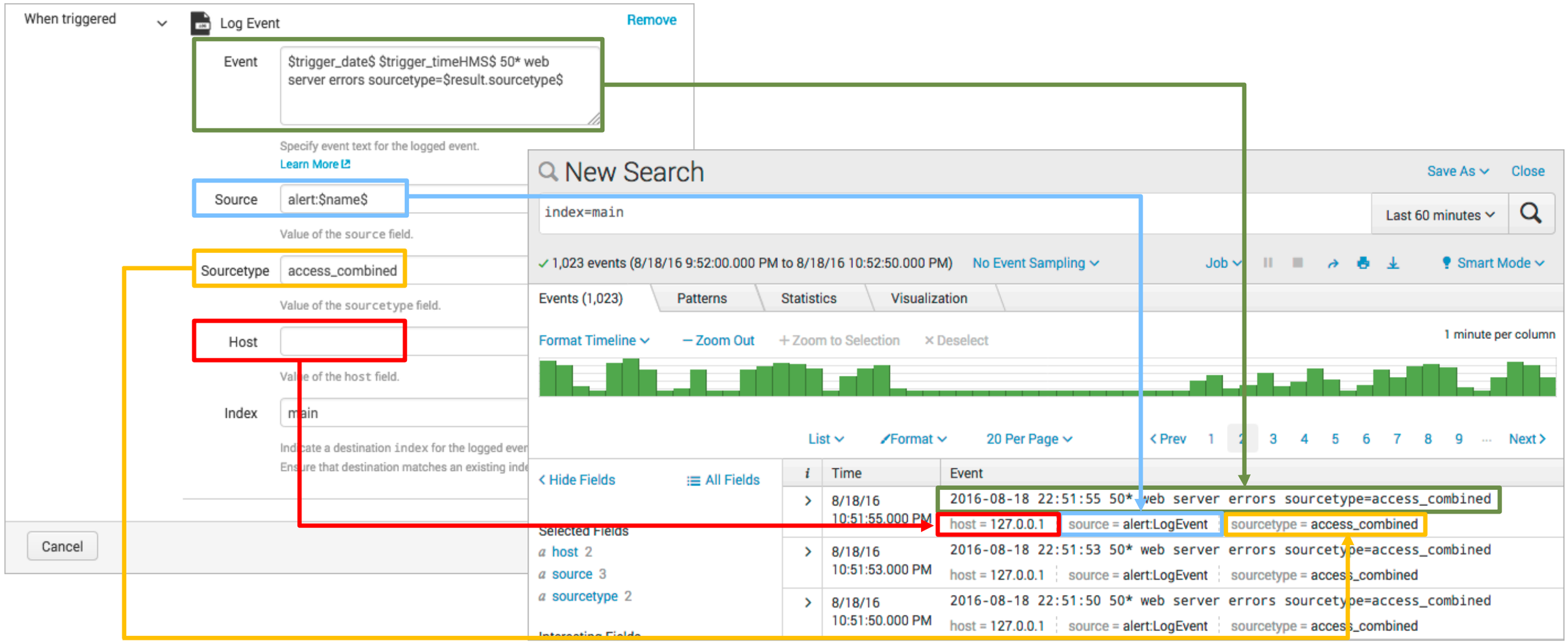
- **Event** – Enter the information that will be written to the event log
- **Source** – The name of the source (alert name is used by default)
- **Sourcetype** – The name of the sourcetype used in the alert
- **Host** – The IP address of the host of the alert
- **Index** – The target index for the log event (default value is main)



Note

For a complete list of available tokens, go to: http://docs.splunk.com/Documentation/Splunk/latest/Alert/EmailNotificationTokens

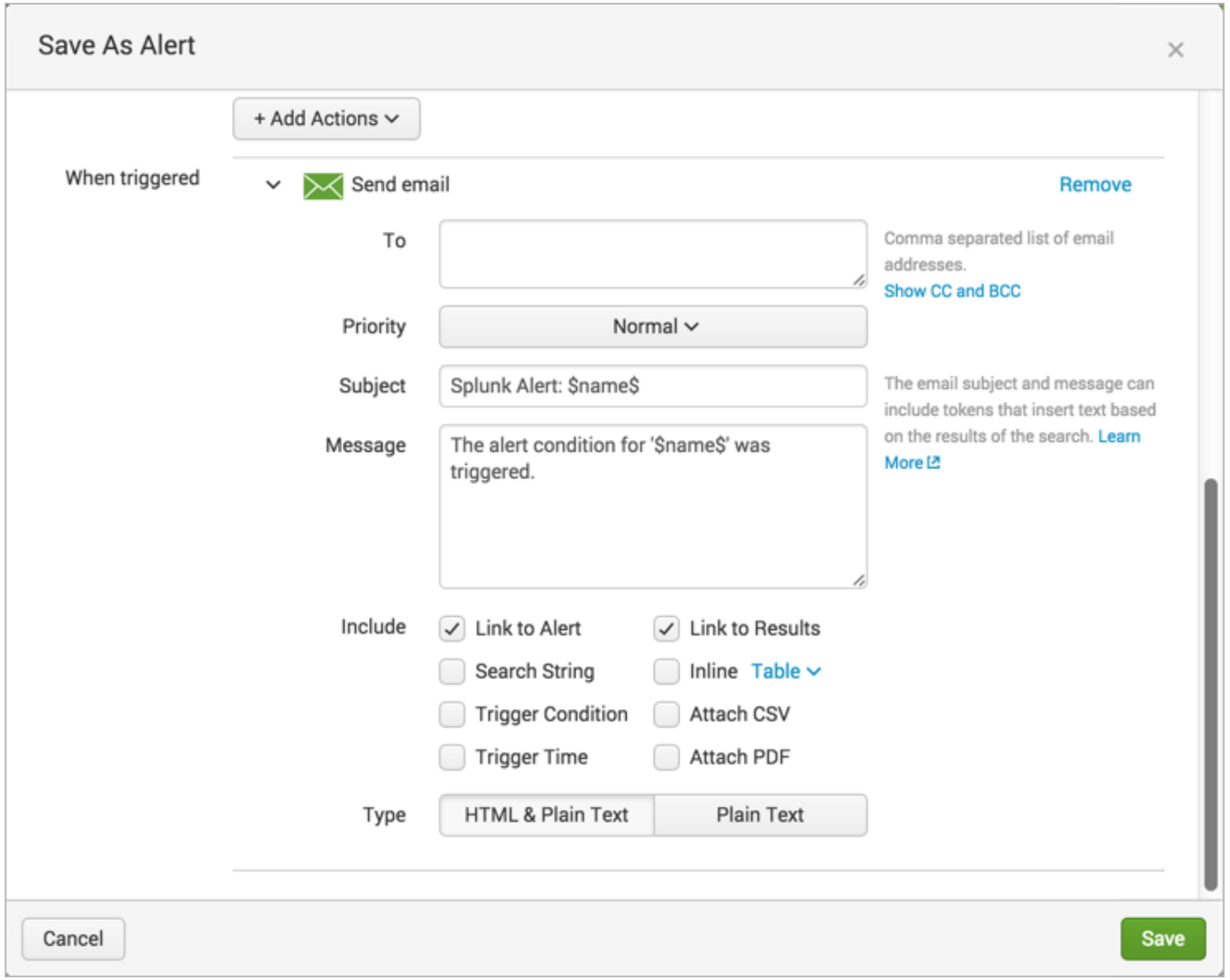
Alert Actions – Log Event (cont.)



Alert Actions – Send Email

Customize the content of email alerts

- **To** - enter the email address(es) of the alert recipients
- **Priority** – select the priority
- **Subject** – edit the subject of the email (the $name$ token is the title of the alert)
- **Message** – provide the message body of the email
- **Include** - select the format of the alert
- **Type** – select the format of the text message

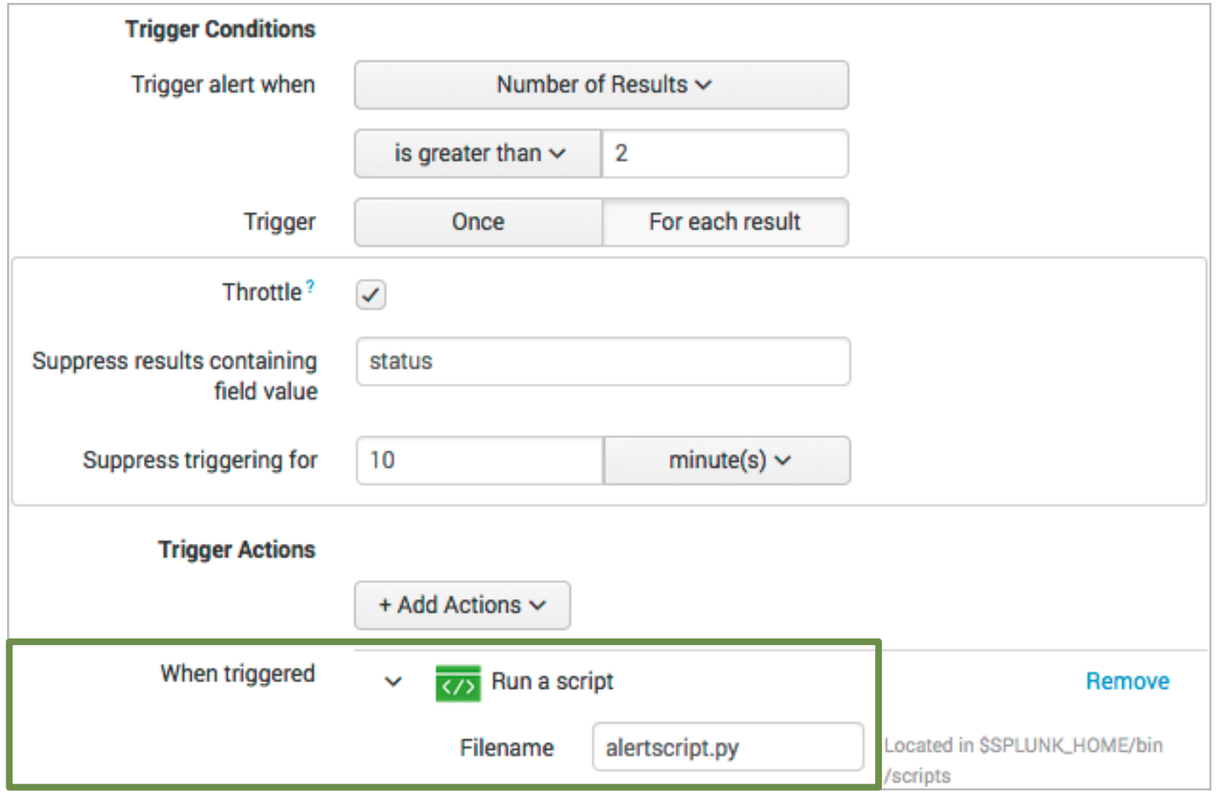


Alert Actions – Run a Script

- When an alert is triggered, you can launch a script
  - Enter the name of the script
  - All alert scripts need to reside in either of the following locations:

```
$SPLUNK_HOME/bin/scripts

$SPLUNK_HOME/etc/apps/
    <Appname>/bin/scripts
```

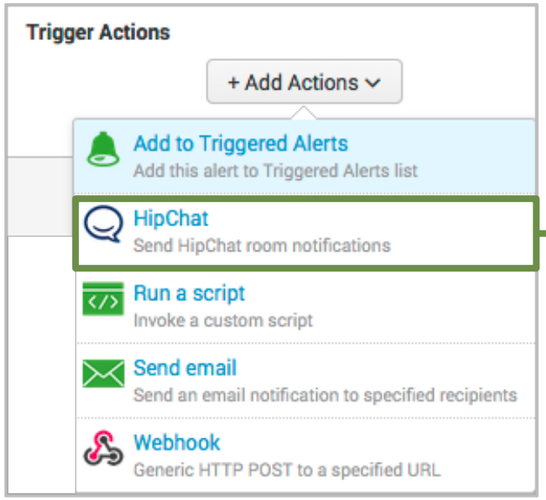


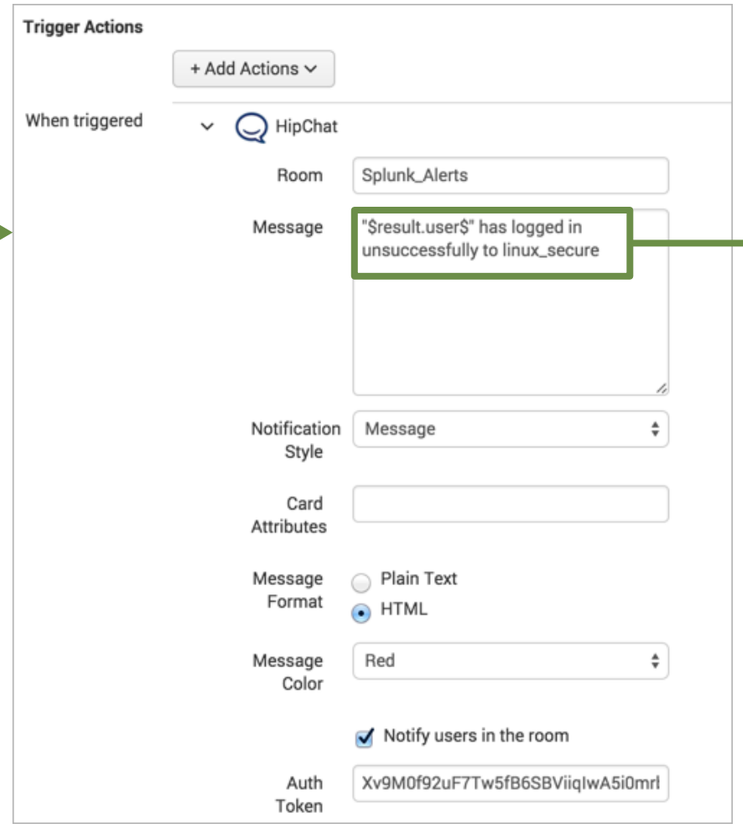
Note

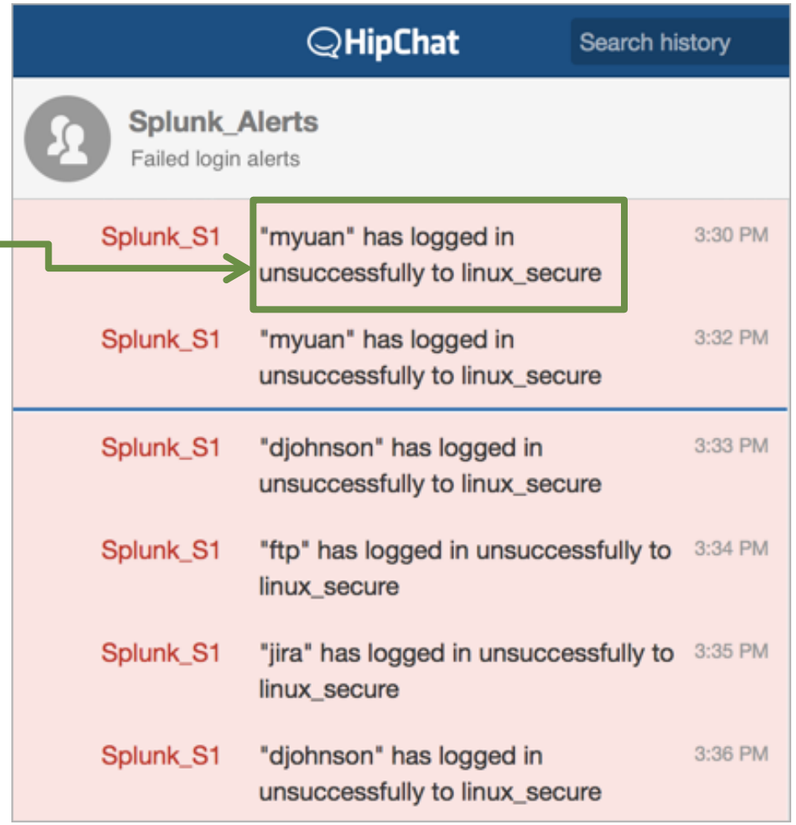
The proper permissions for your Splunk server are required to upload your script to these Splunk directories.

Custom Alert Action - Example

- A custom alert action can be created or an admin can install and configure app from Splunkbase
- In this example, the HipChat Room Notification Alert app is used





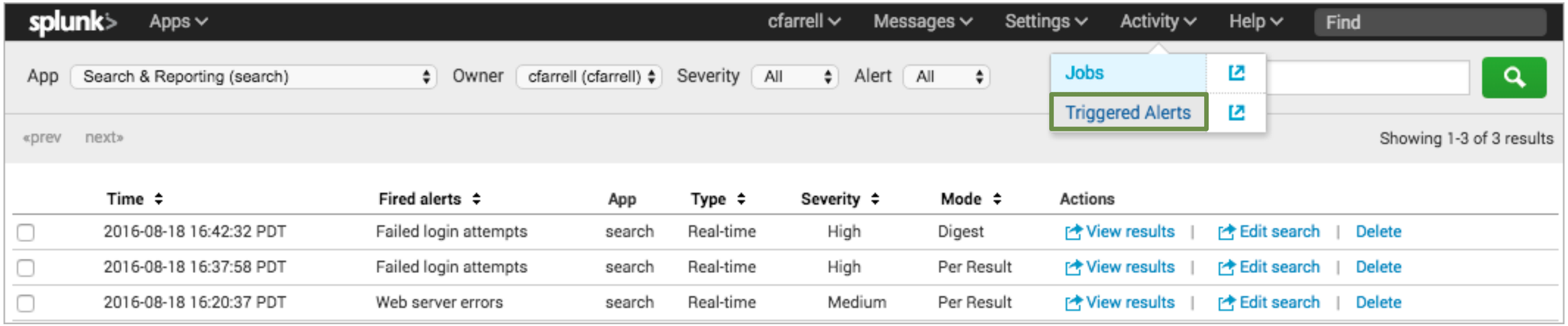


Note

For more information about Custom Alerts, see docs.splunk.com/Documentation/Splunk/latest/AdvancedDev/ModAlertsIntro

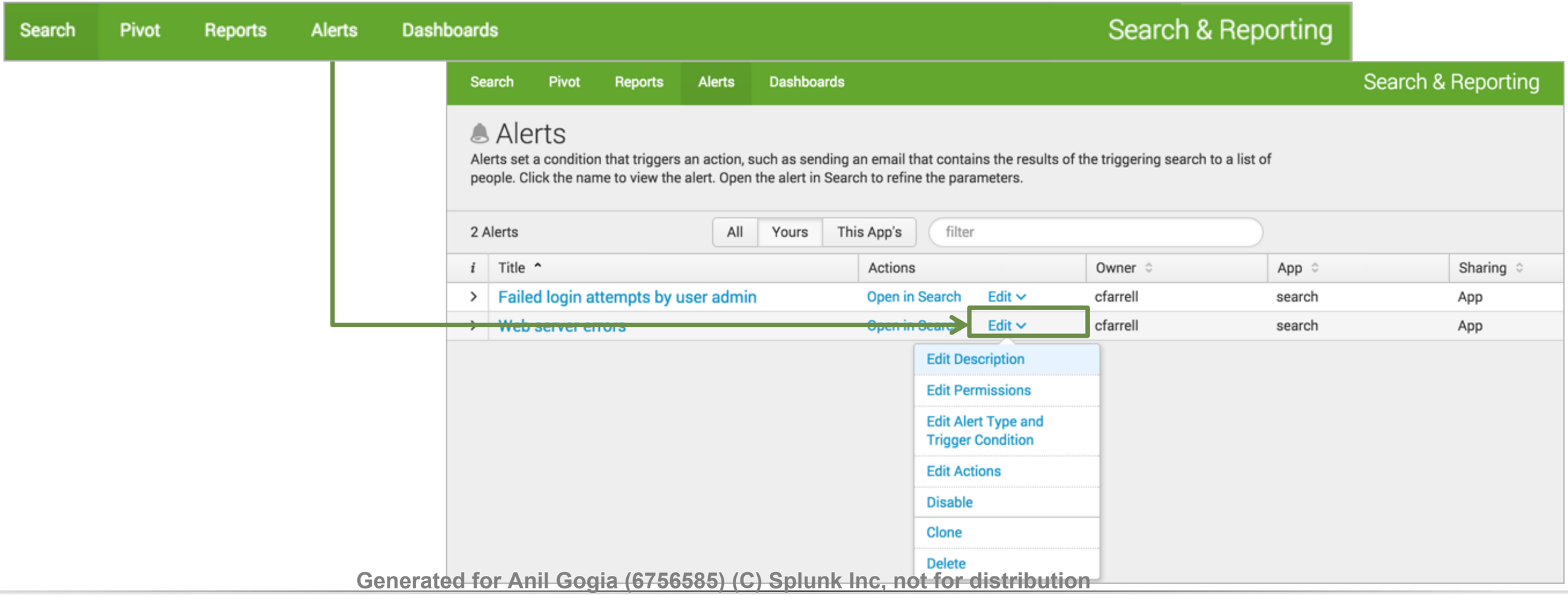
Viewing Triggered Alerts

- If you elected to list in triggered alerts, you can view the results by accessing **Activity > Triggered Alerts**
- Click **View results** to see the matching events that triggered the alert
- Click **Edit search** to modify the alert definition



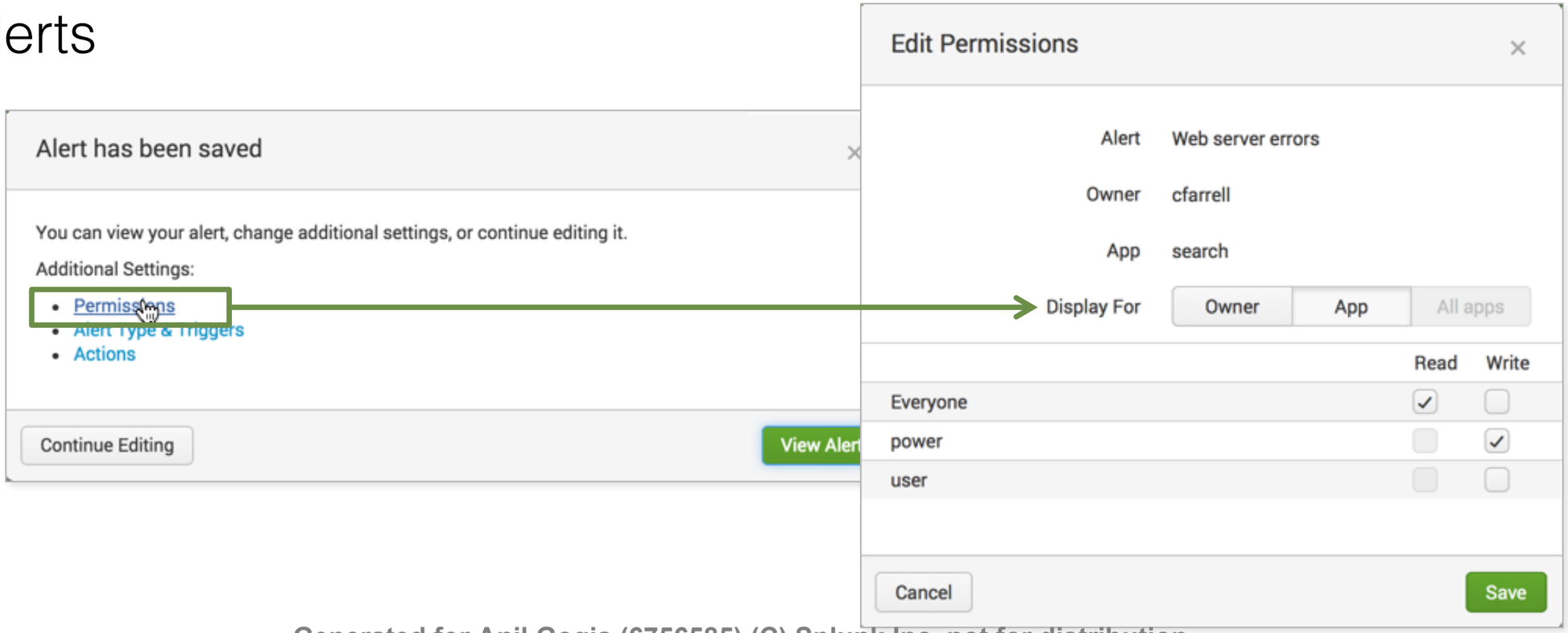
Editing Alerts

1. From the search bar, click **Alerts**
2. Select the alert and click **Edit**



Editing Alert Permissions

- Edit permissions
  - **Owner** – only you can access, edit, and view triggered alerts
  - **App** – users of the app can access, edit, and view triggered alerts



Support Programs

- **Community**
  - **Splunk Answers:** answers.splunk.com
    Post specific questions and get them answered by Splunk community experts.
  - **Splunk Docs:** docs.splunk.com
    These are constantly updated. Be sure to select the version of Splunk you are using.
  - **Wiki:** wiki.splunk.com
    A community space where you can share what you know with other Splunk users.
  - **IRC Channel:** #splunk on the EFNet IRC server Many well-informed Splunk users “hang out” here.
- **Global Support**
  Support for critical issues, a dedicated resource to manage your account – 24 x 7 x 365.
  - **Phone: (855) SPLUNK-S or (855) 775-8657**
  - **Web:** http://www.splunk.com/index.php/submit_issue
- **Enterprise Support**
  Access your customer support team by phone and manage your cases online 24 x 7 (depending on support contract.)