Creating Knowledge Objects Lab Guide

Overview

Welcome to the Splunk Education lab environment. These lab exercises will test your knowledge of creating knowledge objects in a Splunk environment.

Scenario

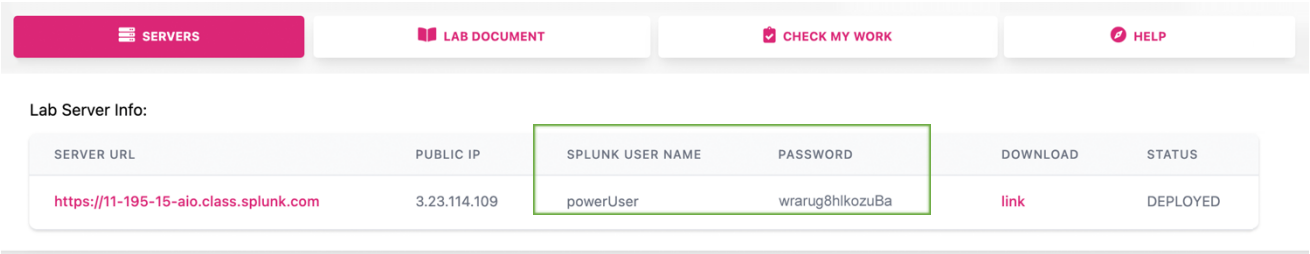
You will use data from the international video game company, Buttercup Games. A list of source types is provided below.

NOTE:	This is a lab environment driven by data generators with obvious limitations. This is not a production environment. Screenshots approximate what you should see, not the **exact** output.

Index	Type	Sourcetype	Interesting Fields
web	Online sales	**access_combined**	action, bytes, categoryId, clientip, itemId, JSESSIONID, price, productId, product_name, referer, referer_domain, sale_price, status, user, useragent
security	Web server	**linux_secure**	action, app, dest, process, src_ip, src_port, user, vendor_action
sales	Business Intelligence server	**sales_entries**	AcctCode, CustomerID, TransactionID
	Retail sales	**vendor_sales**	categoryId, product_name, productId, sale_price, Vendor, VendorCity, VendorCountry, VendorID, VendorStateProvince
network	Web security appliance data	**cisco_wsa_squid**	action, cs_method, cs_mime_type, cs_url, cs_username, sc_bytes, sc_http_status, sc_result_code, severity, src_ip, status, url, usage, x_mcafee_virus_name, x_wbrs_score, x_webcat_code_abbr
	Firewall data	**cisco_firewall**	bcg_ip, dept, Duration, fname, IP, lname, location, rfid, splunk_role, splunk_server, Username

Lab Connection Info

Access labs using the server URL, user name, and password shown in your lab environment.



splunk>

Common Commands and Functions

These commands and statistical functions are commonly used in searches but may not have been explicitly discussed in the module. Please use this table for quick reference. Click on the hyperlinked SPL to be taken to the Search Manual for that command or function.

SPL	Type	Description	Example
sort	command	Sorts results in descending or ascending order by a specified field. Can limit results to a specific number.	*Sort the first 100 src_ip values in descending order* `| sort 100 -src_ip`
where	command	Filters search results using eval-expressions.	*Return events with a count value greater than 30* `| where count > 30`
rename	command	Renames one or more fields.	*Rename SESSIONID to 'The session ID'* `| rename SESSIONID as "The session ID"`
fields	command	Keeps (+) or removes (-) fields from search results.	*Remove the host field from the results* `| fields - host`
stats	command	Calculates aggregate statistics over the results set.	*Calculate the total sales, i.e. the sum of price values* `| stats sum(price)`
eval	command	Calculates an expression and puts the resulting value into a new or existing field.	*Concatenate first_name and last_name values with a space to create a field called "full_name"* `| eval full_name=first_name." ".last_name`
table	command	Returns a table.	*Output vendorCountry, vendor, and sales values to a table* `| table vendorCountry, vendor, sales`
sum()	statistical function	Returns the sum of the values of a field. Can be used with **stats**, **timechart**, and **chart** commands.	*Calculate the sum of the bytes field* `| stats sum(bytes)`
count or count()	statistical function	Returns the number of occurrences of all events or a specific field. Can be used with **stats**, **timechart**, and **chart** commands.	*Count all events as "events" and count all events that contain a value for action as "action"* `| stats count as events, count(action) as action`

Refer to the Search Reference Manual for a full list of commands and functions.

Lab Exercise 1 – Create Event Types

Description

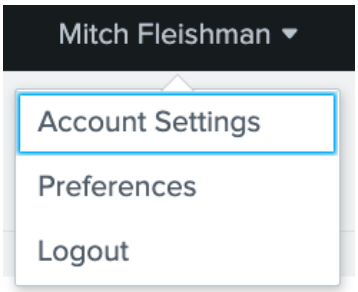
Configure the lab environment user account. Then, create event tags to monitor failed login attempts made with various administrator accounts.

Steps

Task 1: Log into Splunk and change the account name and time zone.

Set up your lab environment to fit your time zone. This also allows the instructor to track your progress and assist you if necessary.

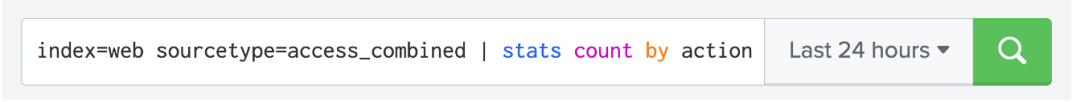
1. Log into your Splunk lab environment using the username and password provided to you.
2. You may see a pop-up window welcoming you to the lab environment. You can click **Continue to Tour** but this is not required. Click **Skip** to dismiss the window.
3. Click on the username you logged in with (at the top of the screen) and then choose **Account Settings** from the drop-down menu.
4. In the **Full name** box, enter your first and last name.
5. Click **Save**.
6. Reload your browser to reflect the recent changes to the interface. (This area of the web interface will be referred to as ***user name***.)



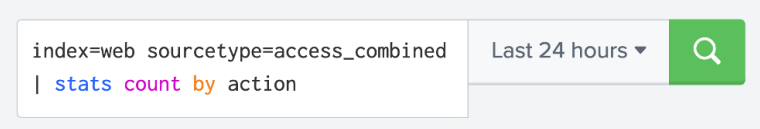
After you complete step 6, you will see your name in the web interface.

NOTE:	Sometimes there can be delays in executing an action like saving in the user interface or returning results of a search. If you are experiencing a delay, please allow the user interface a few minutes to execute your action.

7. Navigate to ***user name*** > **Preferences.**
8. Choose your local time zone from the **Time zone** drop-down menu.
9. Click **Apply**.
10. (Optional) Navigate to ***user name*** > **Preferences** > **SPL Editor** > **Search auto-format** and click on the toggle to activate auto-formatting. Then click **Apply**. When the pipe character is used in search, the SPL Editor will automatically begin the pipe on a new line.



Search auto-format disabled (default)



Search auto-format enabled

Scenario: **The IT Operations team needs to monitor failed login attempts made with any variation of admin/administrator user accounts to their network devices. To avoid lengthy searches, include all events with these user accounts and create tags.**

Task 2: Find errors on web servers/devices.

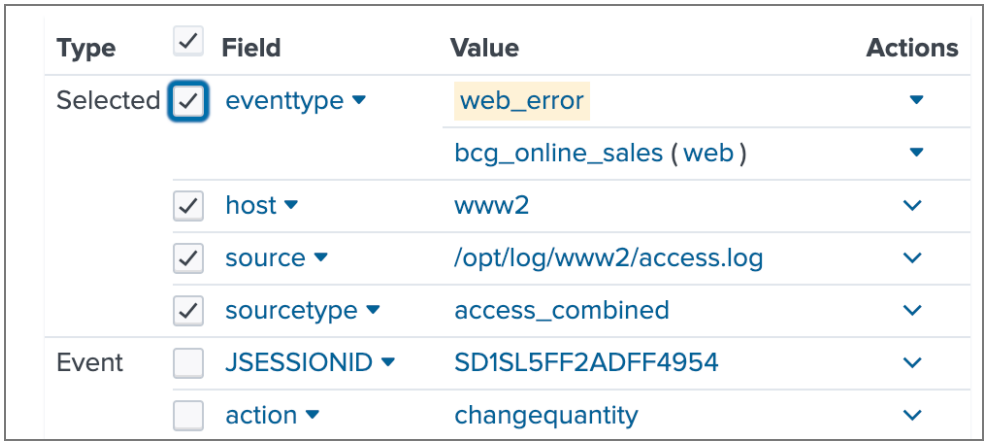
11. In the top left corner of Splunk Web, select **Apps** > **Search & Reporting**. This sets the app context to the search app.
12. Search for all online sales and Web security appliance data with status error codes greater than 500 in the **last 7 days**.

    ```
    (index=web sourcetype=access_combined) OR (index=network sourcetype=cisco_wsa_squid) status>=500
    ```

13. Select **Save As** > **Event Type**.
14. Name your event type: **`web_error`**
15. Leave the **Priority** set to 1 (Highest).
16. Click **Save**.
17. Click **Done** in the **"Your Event Type Has Been Created"** dialog box.
18. Perform a search for the **`web_error`** event type for the **Last 7 days**.

    ```
    eventtype=web_error
    ```

    a. Expand an event and click the check box next to **`eventtype`** to add it to the Selected fields.



    b. In the **Fields** side menu, how many sourcetypes are returned?

    Two sourcetypes.

NOTE:	Depending upon add-ons or apps you have installed, additional event types may be displayed.

splunk>

Lab Exercise 2: Create Workflow Actions

Description

Create GET, POST, and Search workflow actions.

Steps

Scenario: **Hackers are continually trying to log into the Linux web server. IT Ops analysts need to track ongoing attempts by external sources trying to log in with invalid credentials.**

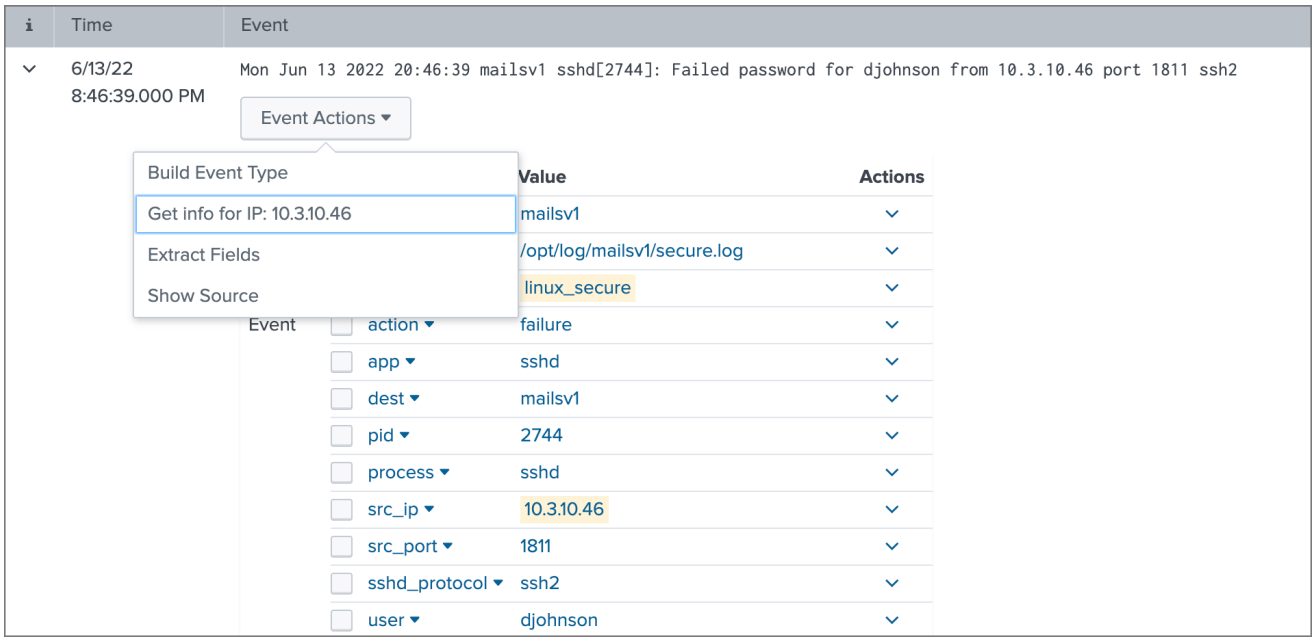
Task 1: **Create a GET workflow action that opens a new browser window with information about the source IP address.**

1. Navigate to **Settings** > **Fields** > **Workflow actions**.
    a. Click **New Workflow Action**.
    b. For the **Destination App**, select **search**.
    c. For **Name**, type: **get_whois_info**
    d. For **Label**, type: **Get info for IP: src_ip**
    e. For **Apply only to the following fields**, type: **src_ip**
    f. For **Action type**, make sure **link** is selected.
    g. For URI, type: **https://who.is/whois-ip/ip-address/src_ip**
    h. From the **Open** link in drop-down menu, verify **New window** is selected.
    i. From the **Link Method** drop-down menu, verify **get** is selected.
    j. Save your workflow action.
2. Verify your workflow action works as expected. Return to the **search** app and perform the following search over the **Last 24 hours**:

```
index=security sourcetype=linux_secure src_ip=*
```

3. Expand the first event containing a value for **src_ip** and click **Event Actions**.
    a. Click **Get info for IP**: {src_ip}. A secondary browser window or tab should open to the URI and display the IP address information.

NOTE:	If whois is not behaving as expected, try https://whois.domaintools.com/src_ip.



Scenario: **The revenue accounting department is having issues with sales transactions not posting to the accounting system. This issue is causing revenue recognition discrepancies and the IT department is tasked with notifying the accounting system administrators when there is a transaction error in the system.**

Task 2: Create a POST workflow action that uses fields from events with errors to create a ticket in the IT ticket tracking system.

4. Perform a search in the **Last 7 days** on the `sales_entries` sourcetype for events posting errors. These events contain two fields that are needed when creating tickets in the tracking system: `TransactionID` and `CustomerID`.

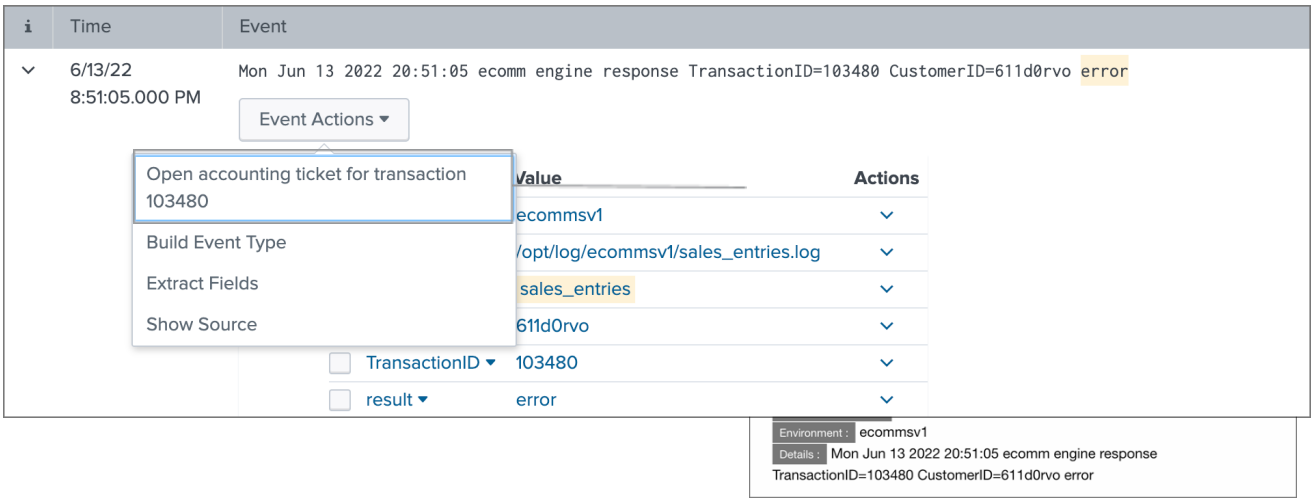
   ```
   index=sales sourcetype=sales_entries error
   ```

NOTE:	This lab exercise requires that a field extraction with a field name of result for the string "error." This extraction allows you to easily search for events where result=error. The result=error field extraction has already been created for you in this lab environment.

5. Create a new workflow action. Navigate to **Settings** > **Fields** > **Workflow actions**. Select **New Workflow Action**.
   a. For the **Destination App**, select **search**.
   b. For **Name**, type: **Create accounting system ticket**
   c. For **Label**, type: **Open accounting ticket for transaction $TransactionID$**
   d. For **Apply only to the following fields**, type: **result**
   e. For **Show Action** in, select **Event menu**.
   f. For **Action type**, make sure **link** is selected.
   g. For **URI**, type: https://tickets.students.splunk.education/
   h. From the **Open link** in drop-down menu, select **New window**.
   i. From the **Link Method** drop-down menu, select **post**.
   j. Enter the following values for the **Post arguments (enter field names and values exactly as they are presented as follows)**:
      - details = $_raw$
      - environment = $host$
      - occurred = $_time$
      - priority = Urgent
      - summary = sales transaction error on $host$

   k. Click **Save**.
6. Rerun your search for events where **result=error** in the **Last 7 days** and view the details of one of the returned events.
7. Expand the Event Actions. Does your POST workflow action appear?
8. Click on your workflow action. A new browser window should appear with the ticket details.

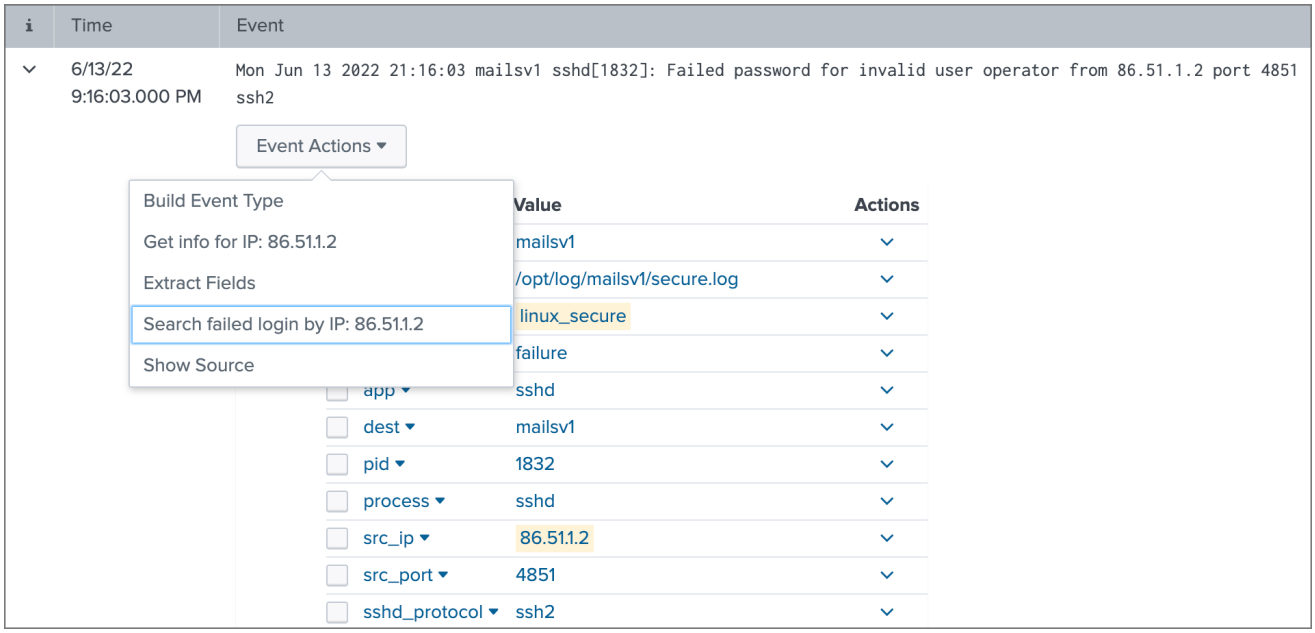
Results Example:

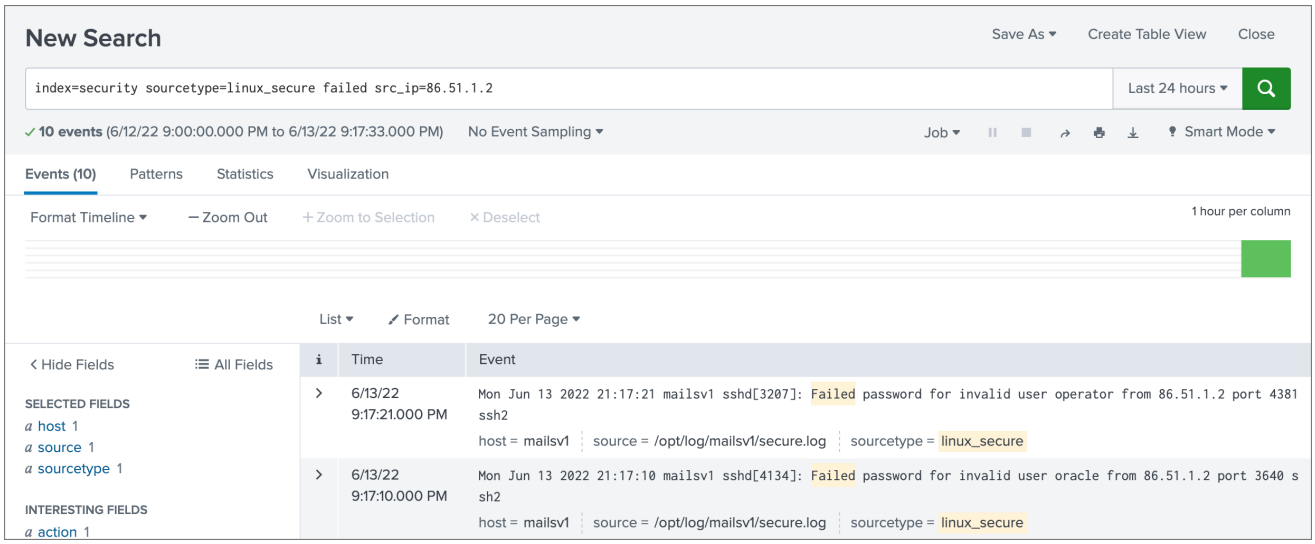


Task 3: Create a Search workflow action that performs a search for all failed password events associated with a specific IP address.

9. Navigate to **Settings** > **Fields** > **Workflow actions**.
    a. Click **New Workflow Action**.
    b. For the **Destination App**, select **search**.
    c. For **Name**, type: **search_access_by_ipaddress**
    d. For **Label**, type: **Search failed login by IP: src_ip**
    e. For **Apply only to the following fields**, type: **src_ip**
    f. From the **Action Type** drop-down menu, select **search**.
    g. In the **Search string** field, type: **index=security sourcetype=linux_secure failed src_ip=src_ip**
    h. From the **Run in** app drop-down menu, select **search**.
    i. From the **Run search in** drop-down menu, verify **New window** is selected.
    j. Select the **Use the same time range** as the search that created the field listing checkbox.
    k. Save your workflow action.
10. Verify your workflow action works as expected. Return to the **Search & Reporting** app and search for **`index=security sourcetype=linux_secure src_ip=*`** over the **Last 24 hours**. (You may need to refresh your browser for the workflow action to appear.)
    a. Expand an event with an IP address field and click **Event Actions**.
    b. Select **Search failed login by IP: {src_ip}**
    c. A secondary search window should open with the search results for the IP address.

Results Example:





Lab Exercise 3: Create Tags and Aliases

Description

This lab exercise walks you through the process of creating field aliases and tags.

Steps

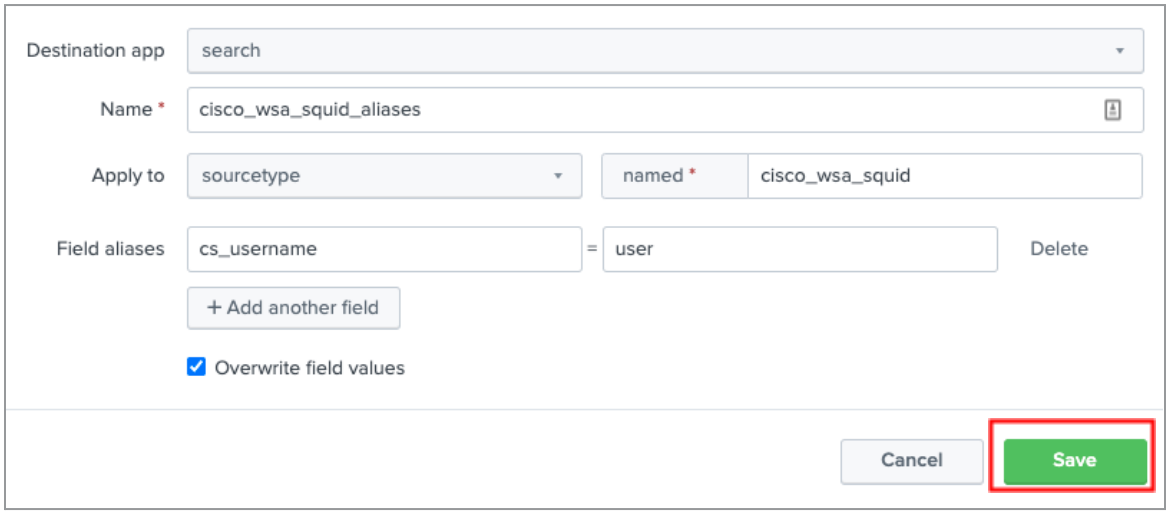
Scenario: **The IT Ops team runs reports for all employee access, but the user name field is not consistent across the different source types.**

Task 1: **Create a field alias so that `cs_username` also appears as `user`.**

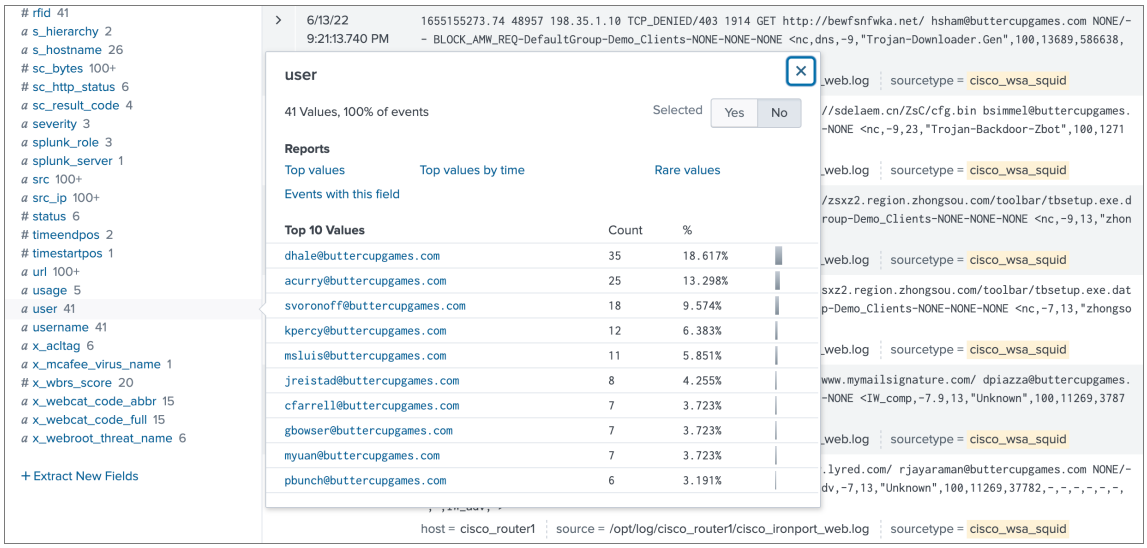
1. Search all events in the **`cisco_wsa_squid`** sourcetype over the **Last 7 days**.

   ```
   index=network sourcetype=cisco_wsa_squid
   ```

2. Note the **`cs_username`** field values.
3. Go to **Settings** > **Fields** > **Field aliases**.
4. Click **New Field Alias**
5. Create a new field alias with the following values:
   - Destination app: **search**
   - Name: cisco_wsa_squid_aliases
   - Apply to: **sourcetype**
   - Named: cisco_wsa_squid
   - Field aliases: **cs_username = user**
6. Select the **Overwrite field values** check box and click **Save**.



7. Return to **Search**. Re-run your search and examine the user field and values.



8. Perform the following search for all events over the **Last 30 days**:

   index=network sourcetype=cisco_firewall

9. Note the **Username** field values.
10. Create another field alias for sourcetype **cisco_firewall** with the following values:
    - Destination app: **search**
    - Name: cisco_firewall_aliases
    - Apply to: **sourcetype**
    - Named: cisco_firewall
    - Field aliases: **Username = user**
11. Perform the following search: **index=network sourcetype=cisco* user=*** over the **Last 30 days**. You should receive results from the **cisco_wsa_squid** and **cisco_firewall** sourcetypes.

NOTE:	It may take a minute before the field aliases are applied and appear in searches.

Scenario: The IT Operations team needs to monitor failed login attempts made with any variation of admin/administrator user accounts to their network devices. To avoid lengthy searches, include all events with these user accounts and create tags.

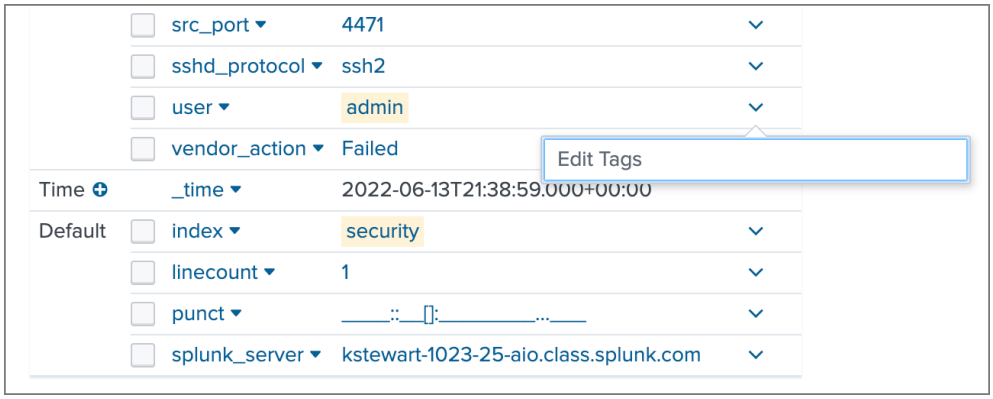
Task 2: Create tags to identify all admin accounts.

12. Run a search over the **Last 24 hours** for all failed login attempts for any variation of the user *admin* under the security index. You should see the following five users: admin, administrator, sysadmin, itmadmin, and sapadmin.

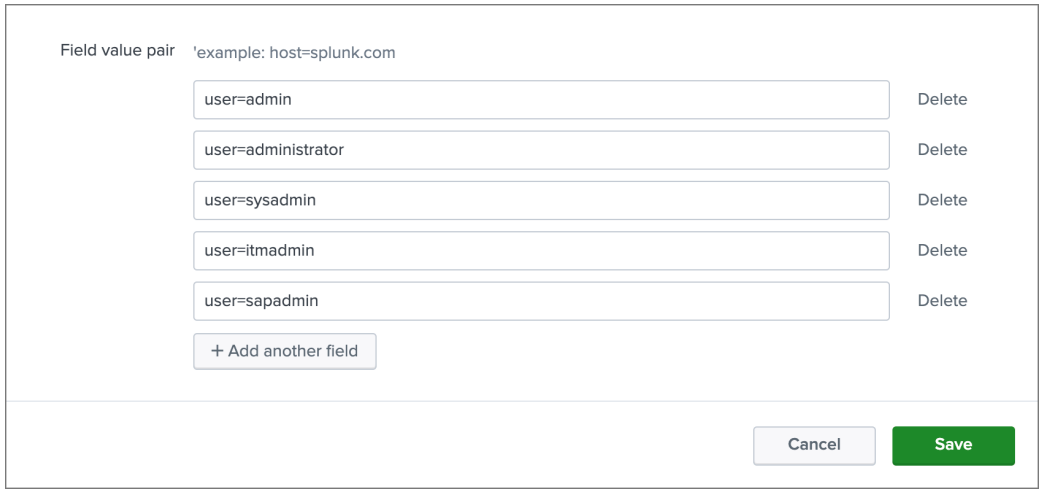
    index=security failed user=*admin*

NOTE:	Only trailing wildcards make efficient use of indexes. For that reason, it is generally a best practice *not* to use wildcards at the beginning of a string, as such searches must scan all events within the specified timeframe. However, doing a search with a wildcard at the beginning of a string is *possible* and sometimes necessary in particular scenarios. Be advised, however, that such searches are inefficient and, in general, should be avoided. Performing an occasional inefficient ad hoc search shouldn't have too much of a performance impact, but such searches certainly shouldn't be used in reports, dashboards, dataset constraints, etc.

13. Expand an event and find the row for the **user** field. Click the **down arrow** under the **Actions** column and select **Edit Tags**.



14. In the **Tag(s)** field, type **privileged_user** and click **Save**.
15. Create tags for each variation of the user *admin (*admin, administrator, sysadmin, itmadmin, and sapadmin). You can create the subsequent tags the same way you created the first one, from the Events tab of the search results. Alternatively, to create the subsequent tags, go to the **Settings** > **Tags** > **List by tag name** screen, choose the newly created **privileged_user** tag, add the other four types of admins, and click **Save**.



16. Run the search again for the **Last 24 hours** and check to see that the **`privileged_user`** tag was created.

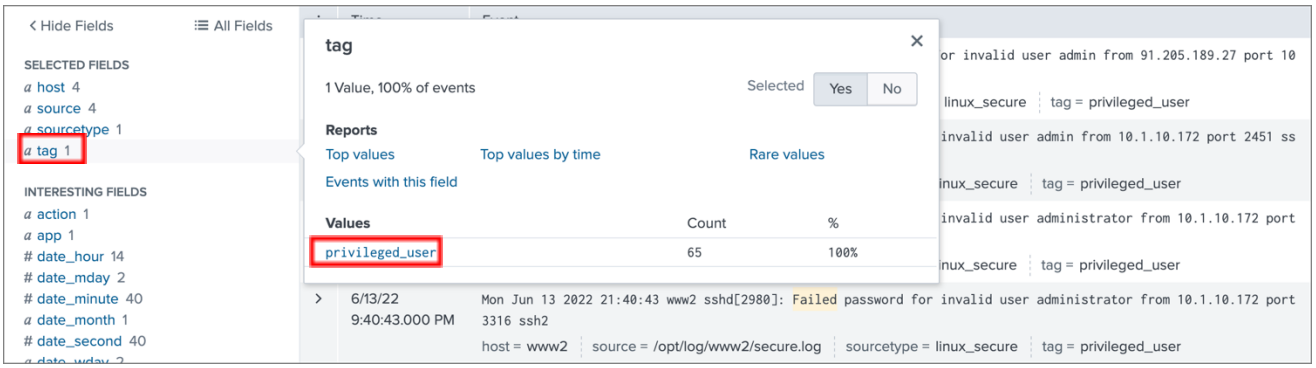
    ```
    index=security failed user=*admin*
    ```

17. Add **tag** to your list of **Selected Fields** if not already present.

Task 3: Use tags in a search.

18. Search for all failed login attempts by privileged user accounts for the **Last 7 days**. You should see the following five users: **`admin`**, **`administrator`**, **`sysadmin`**, **`itmadmin`**, **`sapadmin`**

    ```
    index=security failed tag=privileged_user
    ```



Lab Exercise 4: Create Search Macros

Description

This lab exercise walks you through the steps for creating a basic macro and a macro with arguments.

Steps

Scenario: **The VP of Sales wants to run ad-hoc searches to determine the value of products sold in a given month in various countries. He also wants to easily convert US Dollars to the same value in another currency.**

Task 1: **Use the provided search to create a macro that will create a table displaying the total sales of each product sold in certain European countries.**

19. This search finds all retail sales events from Germany, France, and Italy (**`index=sales sourcetype=vendor_sales VendorCountry IN (Germany, France, Italy)`**) and calculates the total sales by each product with the **`stats`** command. Then, the **`eval`** command converts the numeric sales values to string values with commas and a "$" sign. Run this search over the **Last 30 days**. (Hint: After typing this search string, you may want to copy it into a notepad, as you'll be using it to create a macro later in this exercise.)

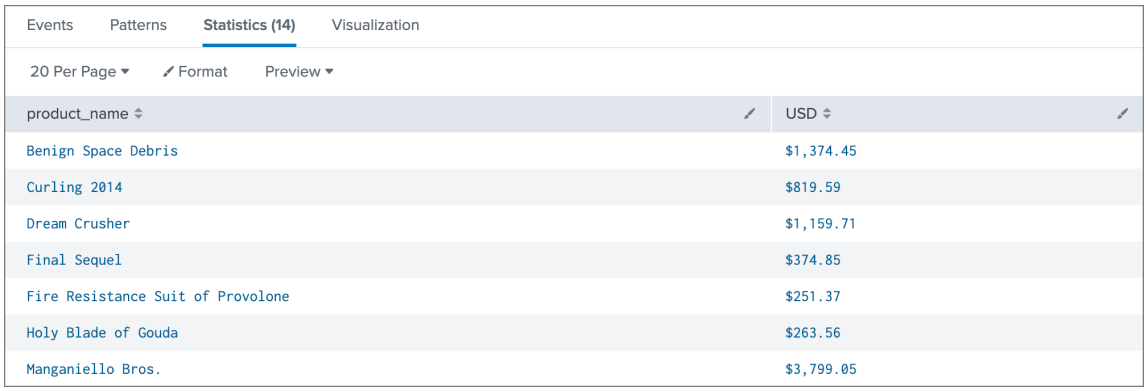
```
index=sales sourcetype=vendor_sales VendorCountry IN (Germany, France, Italy)
| stats sum(price) as USD by product_name
| eval USD = "$".tostring(USD,"commas")
```

20. Navigate to **Settings > Advanced search > Search macros**. Click **New Search Macro**.
    a. Verify the **Destination app** is set to **search**.
    b. Name the macro: **Europe_sales**
    c. In the **Definition** field, type or paste the search string from Step 1.
    d. Save the macro.

Task 2: **Use your macro.**

21. Return to the **Search & Reporting** app. In the search bar, type **`` `Europe_sales` ``** and search over the **Last 30 days**. Examine the results.

NOTE:	Remember to type the macro name between backticks, not single quotes.  Mac Windows



Task 3: Create a macro that allows users to specify currency when performing a search. This macro uses currency, currency symbol, and rate as variables (arguments).

22. Run the following search to determine total sales for each product from vendors in Europe in the **Last 30 days**:

```
index=sales sourcetype=vendor_sales VendorCountry IN (Germany, France, Italy)
| stats sum(price) as USD by product_name
| eval euro = "€".tostring(round(USD*0.79,2), "commas"), USD = "$".tostring(USD,
  "commas")
```

Now you're going to use the second portion of this search string, where the evaluations are done, to create a dynamic macro with arguments.

23. Navigate to. Click **Settings** > **Advanced search** > **Search macros** > **New Search Macro**.
    a. Verify the **Destination app** is set to **search**.
    b. Name the macro: **convert_sales(3)**
    c. To make things easy for the user, the currency, currency symbol and exchange rate are arguments. In the **Definition** field, enter the following search string (the arguments are encapsulated by the $ signs):

```
stats sum(price) as USD by product_name
| eval $currency$="$symbol$".tostring(round(USD*$rate$,2),"commas"),
  USD="$".tostring(USD,"commas")
```

NOTE:	Be sure to include the pipe symbol (|) before the `eval` command.

    d. In the **Arguments** field, type the arguments, separated by commas. (Hint: The order of variables must match the order in which the arguments are passed.)

       currency,symbol,rate

    e. Save the macro.

Task 4: Use your macro with arguments in a search.

24. Return to the **Search & Reporting** app. Perform a search for **`sourcetype=vendor_sales`** where the **`VendorCountry`** is Germany, France, or Italy. Use the macro and pass the arguments **`euro`**, **`€`**, and **`0.79`** for results in the **Last 30 days**. (Hint: You can copy and paste the **€** symbol from this document.)

```
index=sales sourcetype=vendor_sales VendorCountry IN (Germany, France, Italy)
| `convert_sales(euro,€,.79)`
```

25. Run the search again for sales in the UK with the following arguments **`GBP`**, **`£`**, and **`0.64`**. Copy/paste the **`£`** symbol from this document.

```
index=sales sourcetype=vendor_sales VendorCountry="United Kingdom"
|`convert_sales(GBP,£,.64)`
```

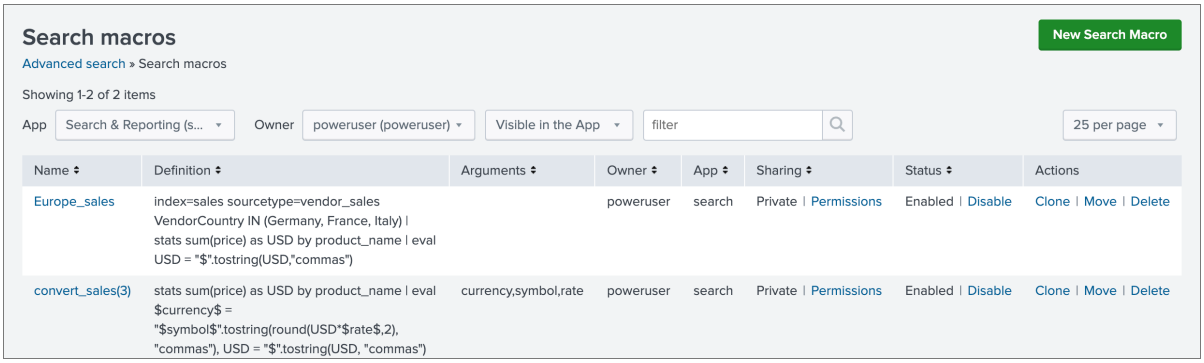
Events Patterns **Statistics (14)** Visualization

20 Per Page ▾ Format Preview ▾

product_name	USD	GBP
Benign Space Debris	$324.87	£207.92
Curling 2014	$499.75	£319.84
Dream Crusher	$519.87	£332.72
Final Sequel	$149.94	£95.96
Fire Resistance Suit of Provolone	$103.74	£66.39
Holy Blade of Gouda	$161.73	£103.51
Manganiello Bros.	$1,439.64	£921.37
Manganiello Bros. Tee	$289.71	£185.41
Mediocre Kingdoms	$499.80	£319.87
Orvil the Wolverine	$1,119.72	£716.62

Task 5: Edit your macro and use the `isnum` expression to validate the rate field.

26. Navigate to **Settings** > **Advanced search** > **Search macros**. Choose your ***user name*** from the **Owner** drop-down menu.



Search macros — New Search Macro

Advanced search » Search macros

Showing 1-2 of 2 items

App: Search & Reporting (s... ▾ Owner: poweruser (poweruser) ▾ Visible in the App ▾ filter 25 per page ▾

Name	Definition	Arguments	Owner	App	Sharing	Status	Actions
Europe_sales	index=sales sourcetype=vendor_sales VendorCountry IN (Germany, France, Italy) | stats sum(price) as USD by product_name | eval USD = "$".tostring(USD,"commas")		poweruser	search	Private | Permissions	Enabled | Disable	Clone | Move | Delete
convert_sales(3)	stats sum(price) as USD by product_name | eval $currency$ = "$symbol$".tostring(round(USD*$rate$,2), "commas"), USD = "$".tostring(USD, "commas")	currency,symbol,rate	poweruser	search	Private | Permissions	Enabled | Disable	Clone | Move | Delete

27. Click on the **convert_sales(3)** link.
28. In the Validation Expression text box, type: **isnum($rate$)**

NOTE:	**isnum** is an informational evaluation function that accepts a single argument and returns TRUE if the argument is a numerical value. Refer to the Search Reference manual for more information about **isnum** and other informational functions.

29. In the **Validation Error Message** text box, type:

    This macro is expecting to be called as `` `convert_sales(currency,symbol,rate)` `` where rate is a numeric value.

30. Click **Save**.
31. Return to the **Search & Reporting** app. Perform a search for **sourcetype=vendor_sales** for the **Last 30 days** where the **VendorCountry** is Germany, France, or Italy. Use the macro, but deliberately pass a non-numeric value for the rate argument (for example, pass the arguments **euro**, **€**, and **.xxx**).

    ```
    index=sales sourcetype=vendor_sales VendorCountry IN (Germany, France, Italy)
    | `convert_sales(euro,€,.xxx)`
    ```

32. Check to see that your error message displays.

Results Example:

Lab Exercise 5: Create Calculated Fields

Description

This lab exercise walks you through the steps for creating calculated fields.

Steps

Scenario: **The IT Ops team is monitoring bandwidth usage for all users for the last month, but the data is reported in bytes. The team needs the usage to be measured in megabytes.**

Task 1: Create a calculated field that converts bytes to MB.

33. Search for all events in the **Last 7 days** for the **`cisco_wsa_squid`** sourcetype.

```
index=network sourcetype=cisco_wsa_squid
```

34. Note the **`sc_bytes`** field. This field displays the amount of bytes used for that event.
35. Go to **Settings** > **Fields** > **Calculated fields** > **New Calculated Field**.
36. Create a calculated field named **`sc_megabytes`** that converts the value of **`sc_bytes`** to MB with the following values:
    - Destination app: **search**
    - Apply to: **sourcetype**
    - Named: cisco_wsa_squid
    - Name: sc_megabytes
    - Eval expression: sc_bytes/(1024*1024)
37. Save the new calculated field.
38. Return to the **Search & Reporting** app. Run this search using **`sc_megabytes`** over the **Last 7 days.**

```
index=network sourcetype=cisco_wsa_squid
| stats sum(sc_megabytes) as "Bandwidth (MB)", sum(sc_bytes) as sc_bytes by usage
```

usage	Bandwidth (MB)	sc_bytes
Borderline	1.66202926635742190000	1742764
Business	3.03161239624023440000	3178876
Personal	7.06902980804443400000	7412415
Unknown	2.17674255371093750000	2282480
Violation	0.0175428390502929700	18395