Visualizations Lab Guide

Overview

Welcome to the Splunk Education lab environment. These lab exercises will test your knowledge of formatting commands and your ability to visualize data using transforming and mapping commands.

Scenario

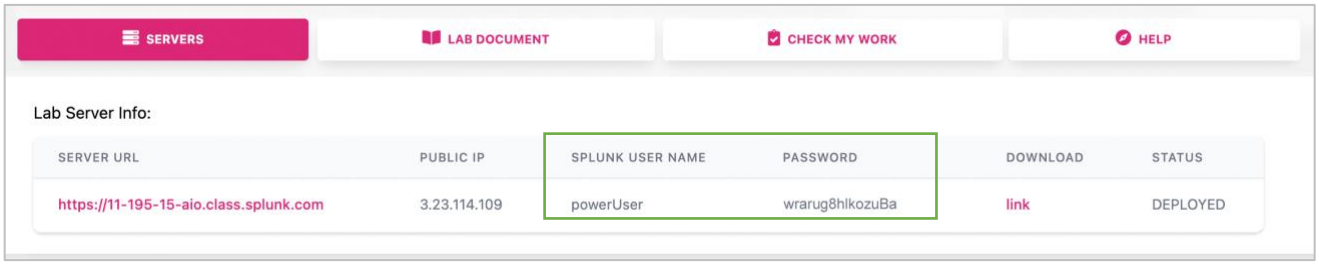
You will use data from the international video game company, Buttercup Games. A list of source types is provided below.

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	Badge reader	**history_access**	Address_Description, Department, Device, Email, Event_Description, First_Name, last_Name, Rfid, Username
	Active Directory	**winauthentication_security**	LogName, SourceName, EventCode, EventType, User
	Web server	**linux_secure**	action, app, dest, process, src_ip, src_port, user, vendor_action
sales	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
games	Game logs	**SimCubeBeta**	date_hour, date_mday, date_minute, date_month, date_second, data_wday, data_year, date_zone, eventtype, index, linecount, punct, splunk_server, timeendpos, timestartpos

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 Exercises

Description

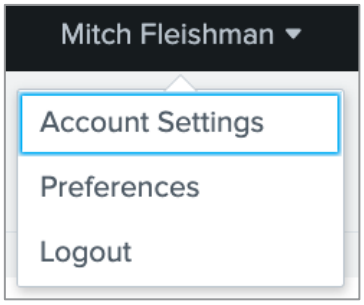
Configure the lab environment user account. Then, transform data using **`chart`**, **`timechart`**, **`top`**, **`rare`**, and **`stats`** commands.

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 UI or returning results of a search. If you are experiencing a delay, please allow the UI 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)

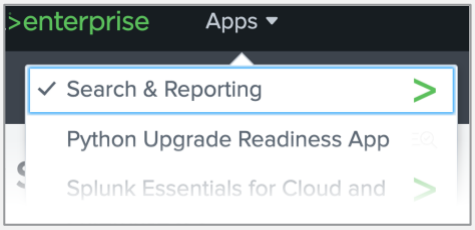
Search auto-format enabled

Scenario: The Sales team wants to see online sales and retail sales from the last 24 hours visualized as a bar chart.

Task 2: Use the `timechart` command to visualize data in an area chart. Then use advanced formatting options to add a chart overlay.

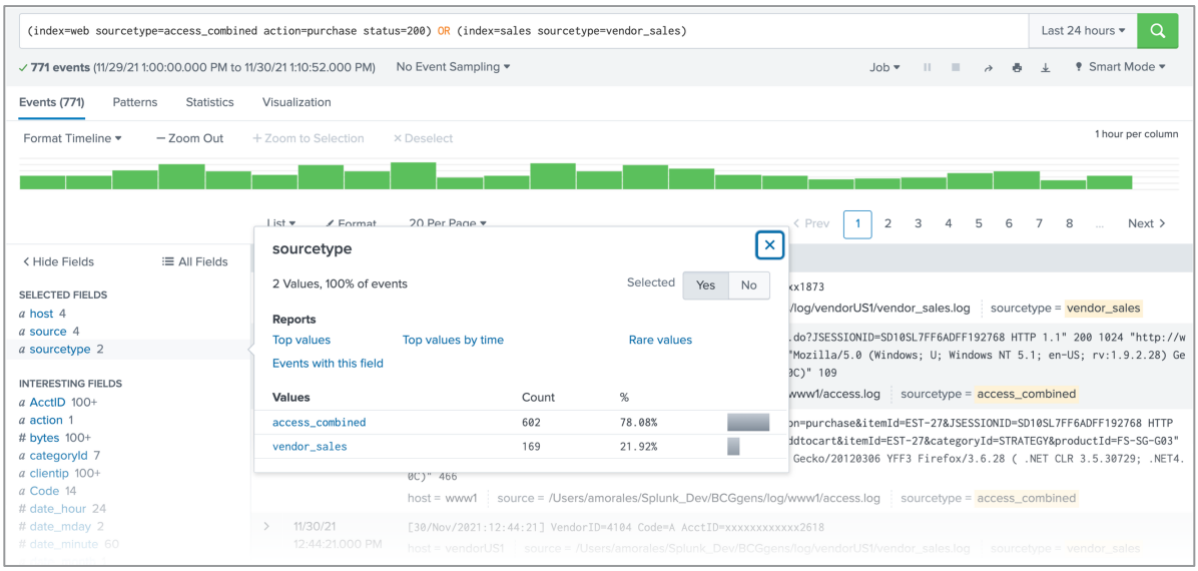
11. In the top left corner of Splunk Web, select **Apps** > **Search & Reporting**. This sets the app context to the search app.

NOTE: If **Search & Reporting** has a checkmark then you are in the right app and do not need to perform step 11.



12. Execute the following search over the **Last 24 hours**. This search will return successful online purchase events from the **`access_combined`** sourcetype and retail sales events from the **`vendor_sales`** sourcetype.

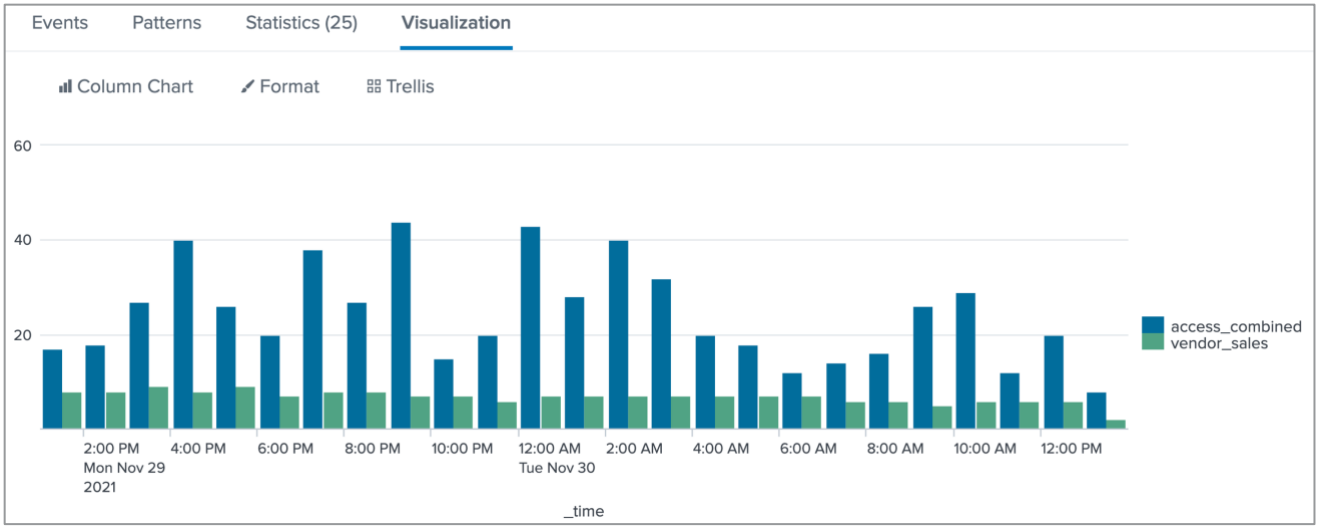
```
(index=web sourcetype=access_combined action=purchase status=200) OR (index=sales sourcetype=vendor_sales)
```



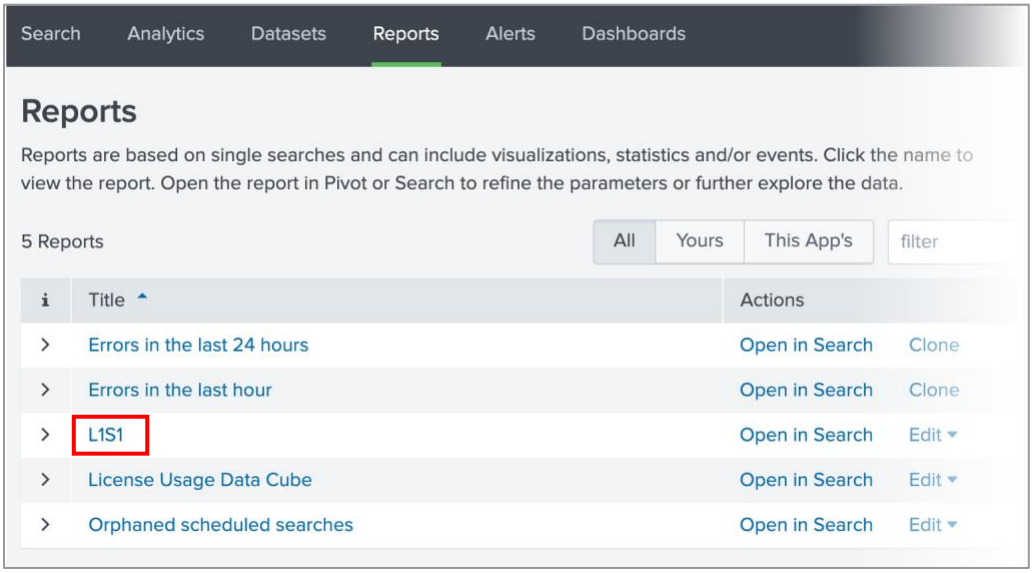
13. Pipe the results to the **`timechart`** command so that events are counted by **`sourcetype`** with a **`span`** of 1 hour.

_time	access_combined	vendor_sales
2021-11-29 13:00	17	8
2021-11-29 14:00	18	8
2021-11-29 15:00	27	9
2021-11-29 16:00	40	8
2021-11-29 17:00	26	9
2021-11-29 18:00	20	7

14. Click on the **Visualization** tab and display results as a **Column Chart**.



15. Save your search as a report with the name **L1S1**.
    a. Click **Save As** > **Report**
    b. For **Title**, enter L1S1.
    c. **Save**.
    d. You can **View** your report or exit out of the **Your Report Has Been Created** window by clicking the **X** in the upper-right corner.
    e. You can access your saved reports using the **Reports** tab in the application bar.

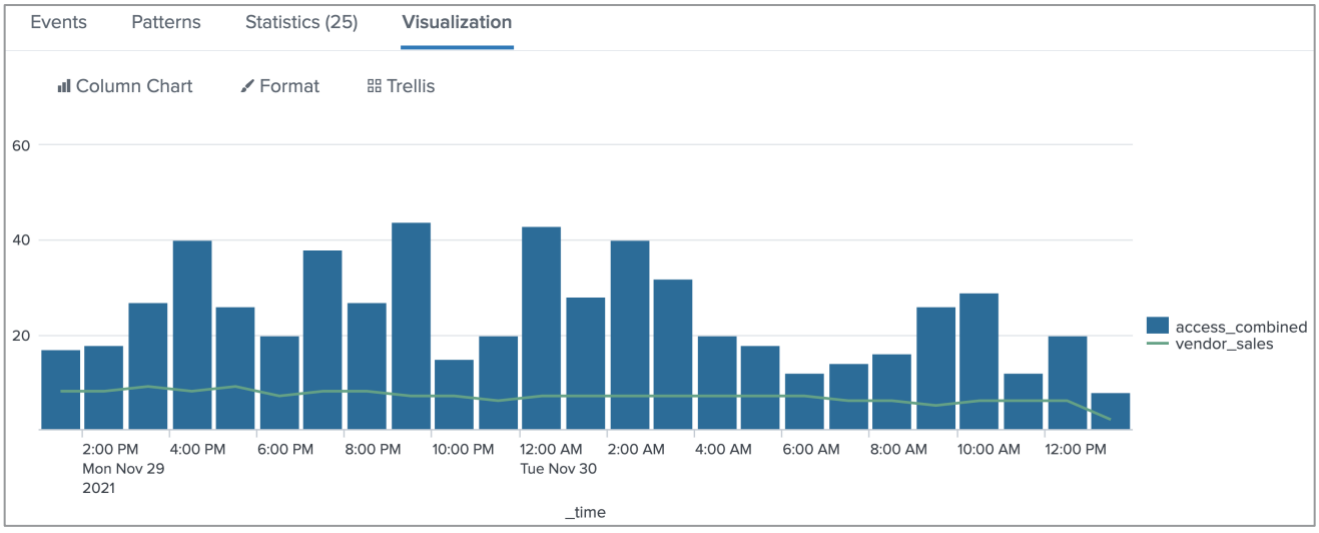


*Your recently saved **L1S1** report will be visible in the **Reports** tab.*

Scenario: The Sales team manager liked the visualization but wants to know if you can "make the retail sales data easier to read."

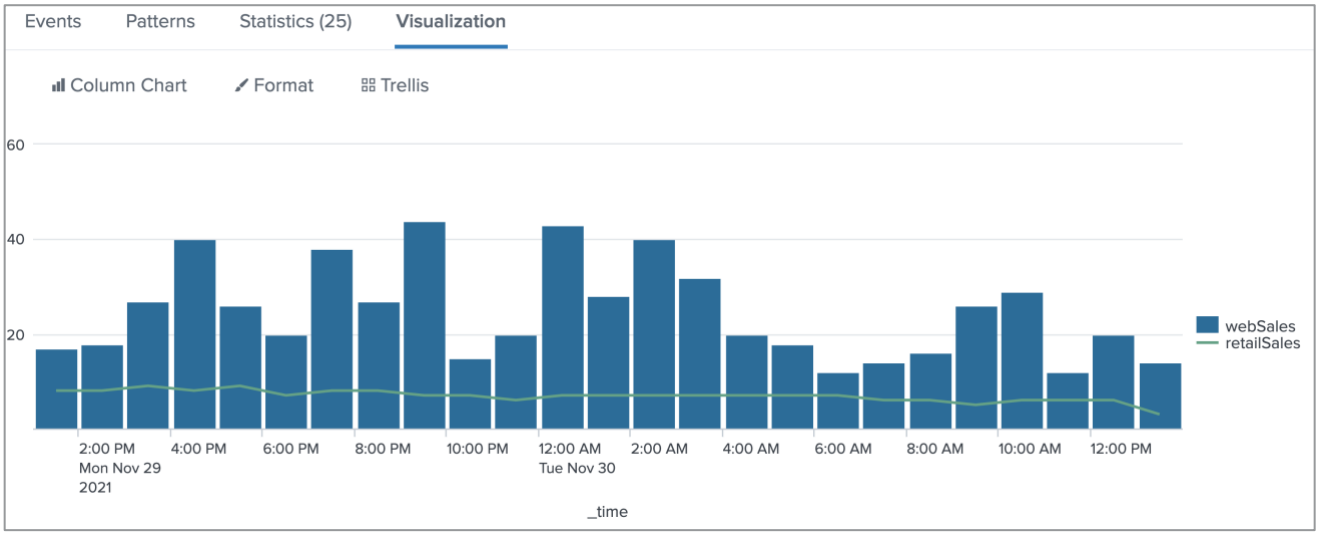
Task 3: Convert the retail sales data to a chart overlay. Optionally, use the `rename` command to rename the sourcetypes.

16. (Optional) If you are currently on the **Reports** tab then you will need to navigate back to your L1S1 report. Under **Actions** for your **L1S1** report, click on **Open in Search**. Then, confirm that you are in the **Visualization** tab.
17. Under **Visualization**, click on the **Format** tab and use the **Chart Overlay** tab to assign **`vendor_sales`** data as the **Overlay**.



18. This report needs to be easier to read for non-IT employees. Use the **`rename`** command to rename **`vendor_sales`** as "retailSales" and **`access_combined`** as "webSales" to make the legend easier to read. Use the **Common Commands and Functions** table in this document for details on the **`rename`** command.

19. Change the **Overlay** field to **`retailSales`**.



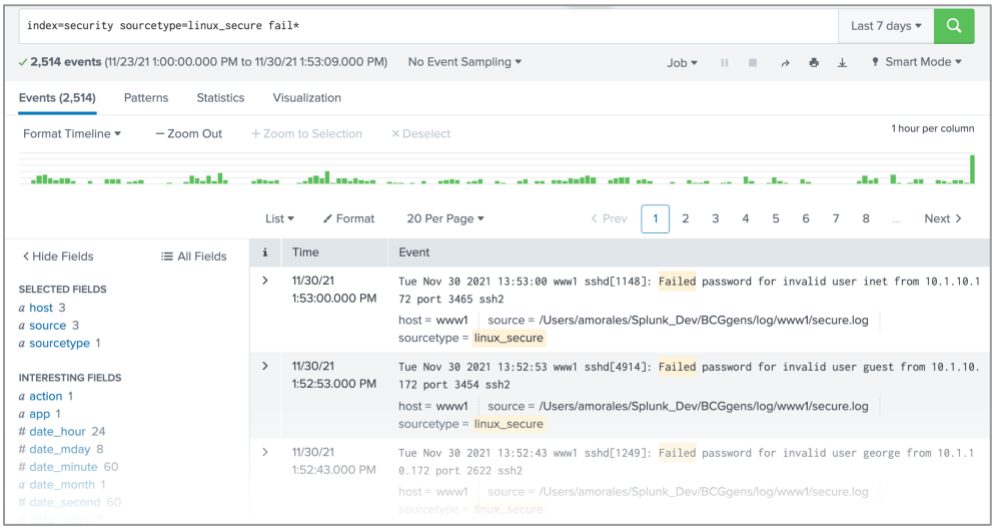
20. Click on **Save As** to save your search as a report with the name **L1S2**. (Do not click **Save** or you will overwrite your L1S1 report.)

Scenario: The Security Operations team would like a visualization of authentication failures that occurred over the last week with a simple moving average trendline.

Task 4: Use the `timechart` and `trendline` commands to visualize authentication failures.

21. Re-initialize the search window by clicking **Search** in the application bar. This step should be done every time you save a report so that you do not accidentally overwrite a previous report.
22. Execute the following search over the **Last 7 days**. This search will return results from the web server that contain "fail" in the raw data.

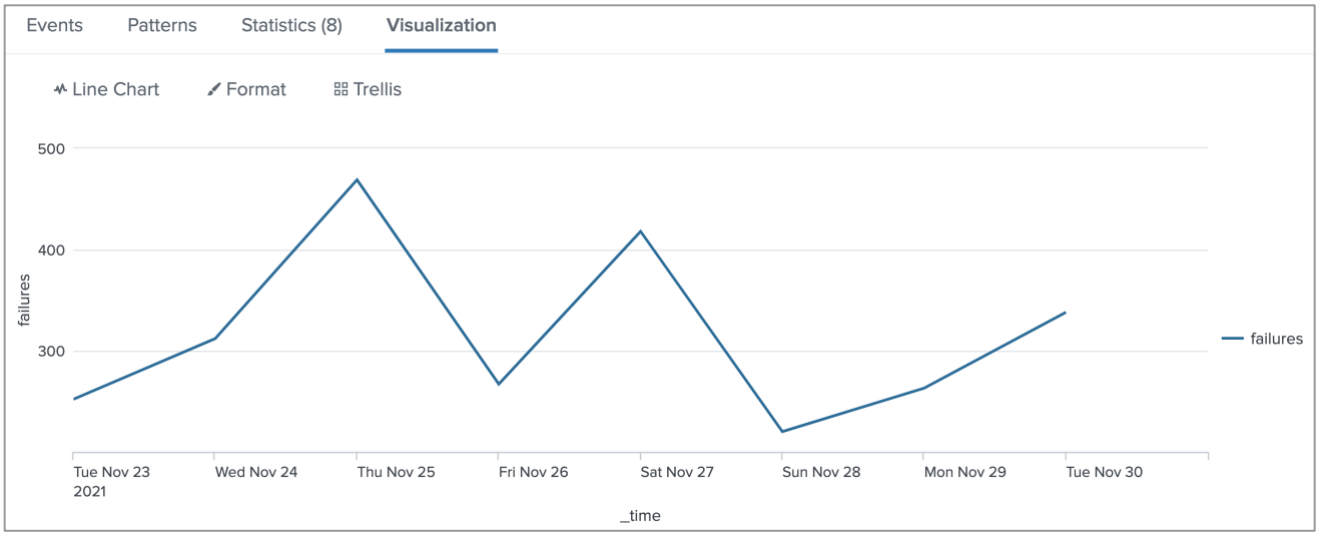
```
index=security sourcetype=linux_secure fail*
```



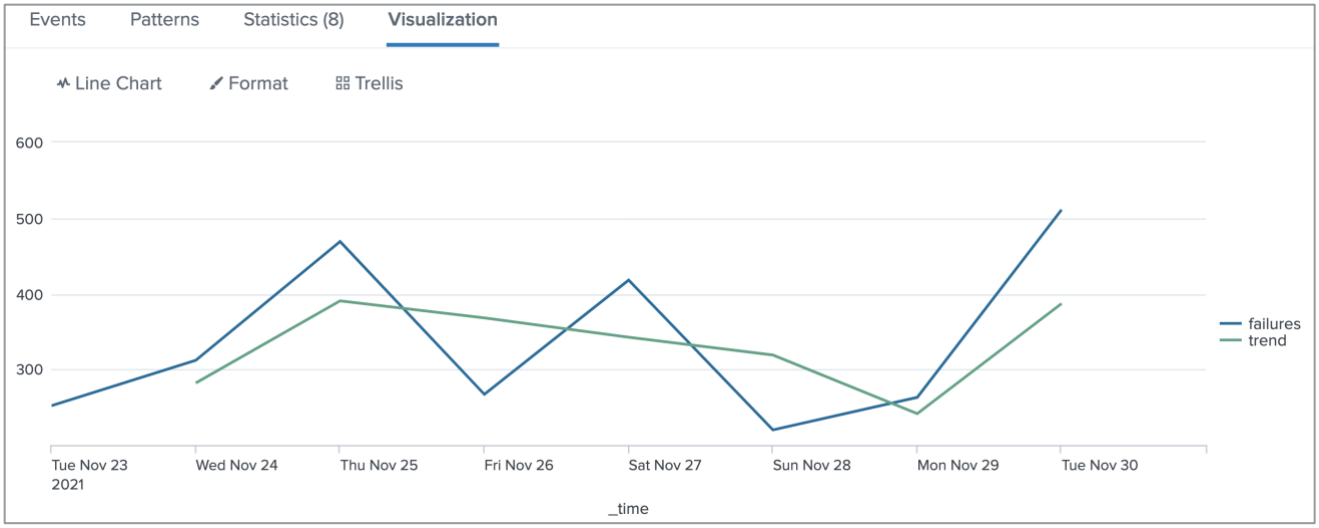
23. Use the **`timechart`** command to count events with a **`span`** of 1 day. Use an **`as`** clause to name the output "failures."

_time	failures
2021-11-23	252
2021-11-24	312
2021-11-25	469
2021-11-26	267
2021-11-27	418
2021-11-28	220
2021-11-29	263
2021-11-30	338

24. Visualize your results as a **Line Chart**.



25. Find the trendline of **`failures`** with the **`trendline`** command. Use the simple moving average trendtype with a period of 2 days. Use an **`as`** clause to label this output as "trend."



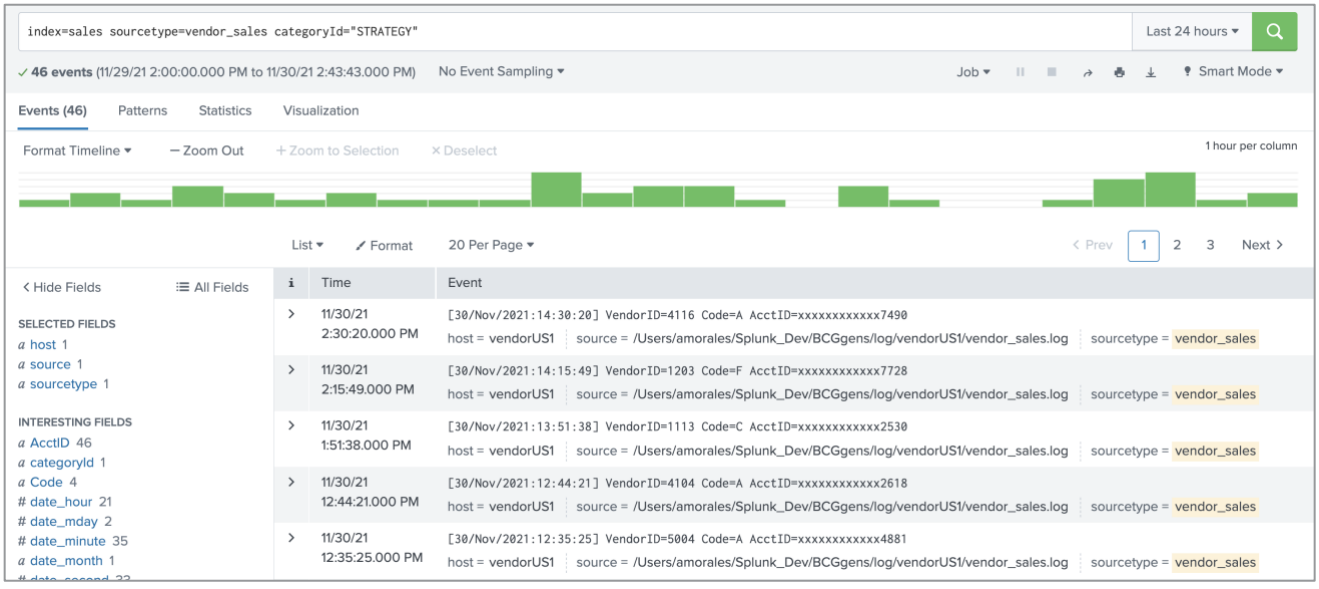
26. Save your search as a report with the name **L1S3**.

Scenario: Display the daily sales count of strategy games at Buttercup Games retail locations (i.e., not online) during the previous week, and display the sales count and trend for the previous day.

Task 5: Use the `timechart` command to count retail sales events for strategy games and visualize results as a single value visualization with a sparkline and trendline.

27. Re-initialize the search window by clicking **Search** in the application bar. This step should be done every time you save a report so that you do not accidentally overwrite a previous report.
28. Execute the following search over the **Previous week**. This search will return retail sales events for strategy games.

```
index=sales sourcetype=vendor_sales categoryId="STRATEGY"
```

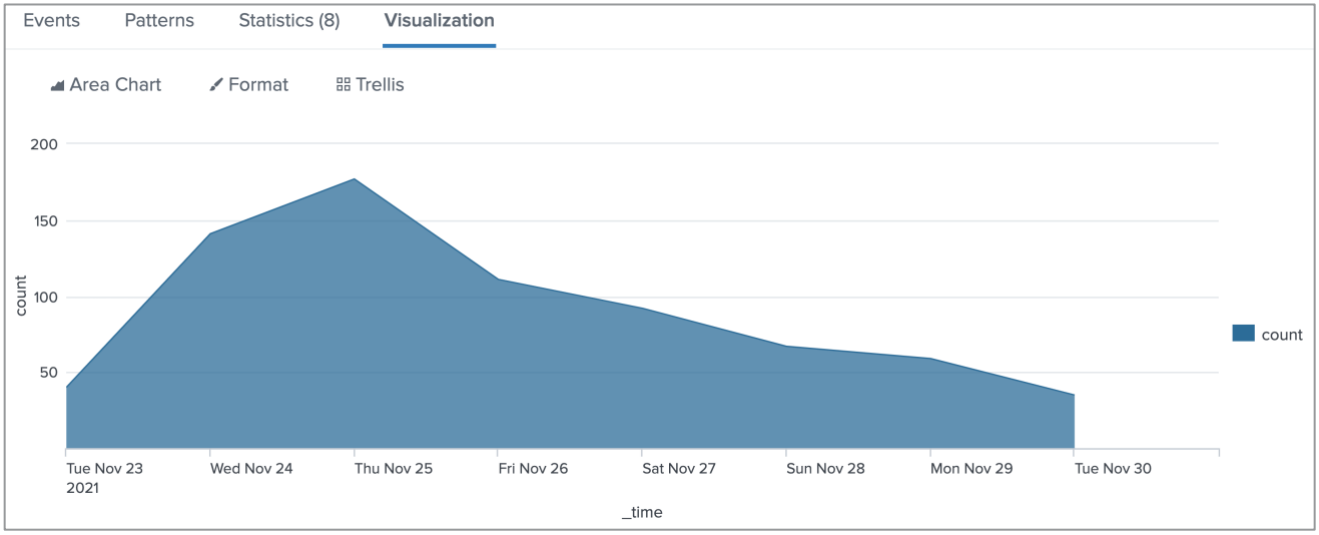


NOTE: Since the **`categoryId`** comes from a lookup, the value being matched is case sensitive. Therefore, “STRATEGY” must be uppercase.

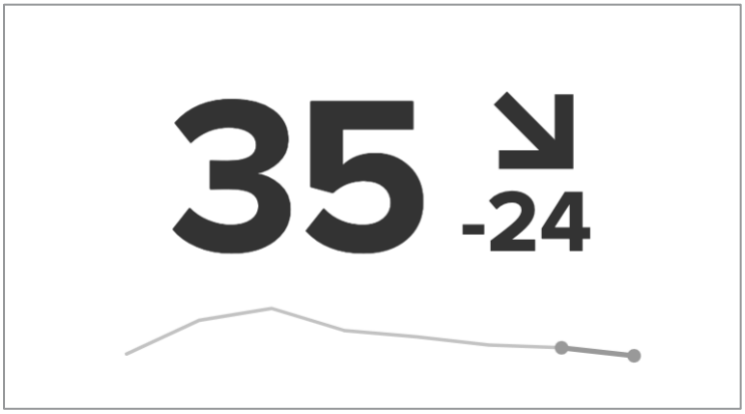
29. Use the **`timechart`** command to count events with a **`span`** of 1 day.

_time	count
2021-11-23	70
2021-11-24	141
2021-11-25	177
2021-11-26	111
2021-11-27	92
2021-11-28	67
2021-11-29	59
2021-11-30	31

30. Visualize your results as an **Area Chart**.



31. Change your visualization to **Single Value**. You should see the most recent **`count`** value, a sparkline with the difference between today's **`count`** value and yesterday's **`count`** value, and a trendline under these two values.



32. Adjust the following **Format** options:

General

a. **Show Trend Indicator**: Yes
b. **Show Trend in**: Absolute
c. **Caption**: Strategy Games Sales – Previous Day
d. **Show Sparkline**: Yes

Color

e. **Use Colors**: Yes
f. **Color By**: Trend
g. **Color Mode**: Block background (i.e. white numbers on a color background)

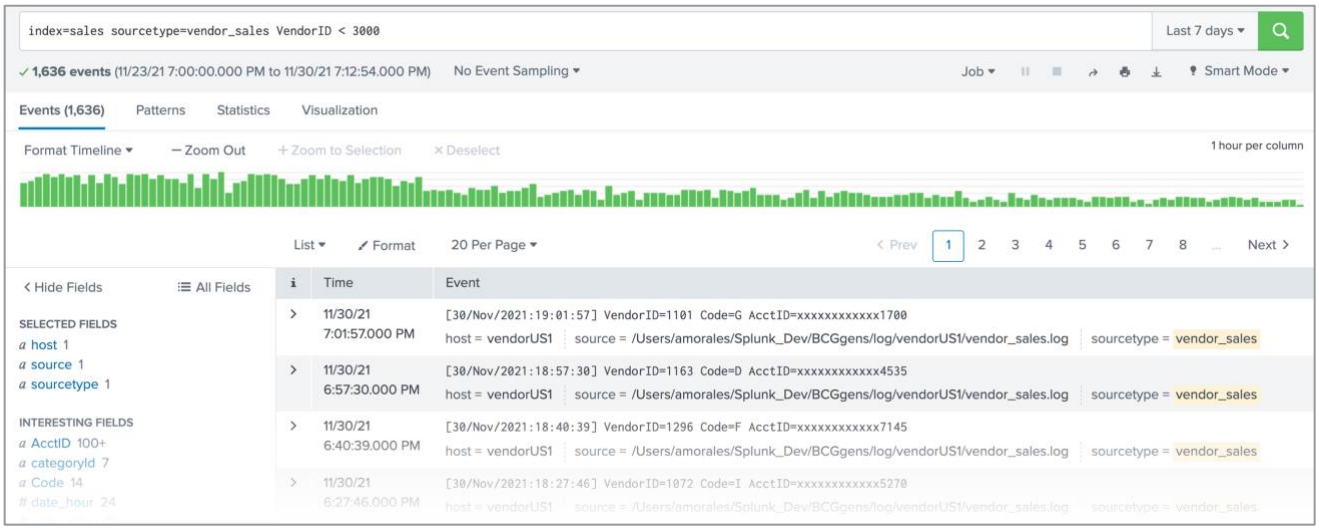
33. Save your search as a report with the name **L1S4**.

Scenario: The Marketing manager wants to see how the new marketing campaign is affecting retail sales across the U.S.

Task 6: Use the `chart` command to calculate retail sales events for each state and the `geom` command to create a choropleth map of the United States.

34. Re-initialize the search window by clicking **Search** in the application bar. This step should be done every time you save a report so that you do not accidentally overwrite a previous report.
35. Execute the following search over the **Last 7 days**. This search returns retail sales events from Vendors in the United States.

```
index=sales sourcetype=vendor_sales VendorID < 3000
```



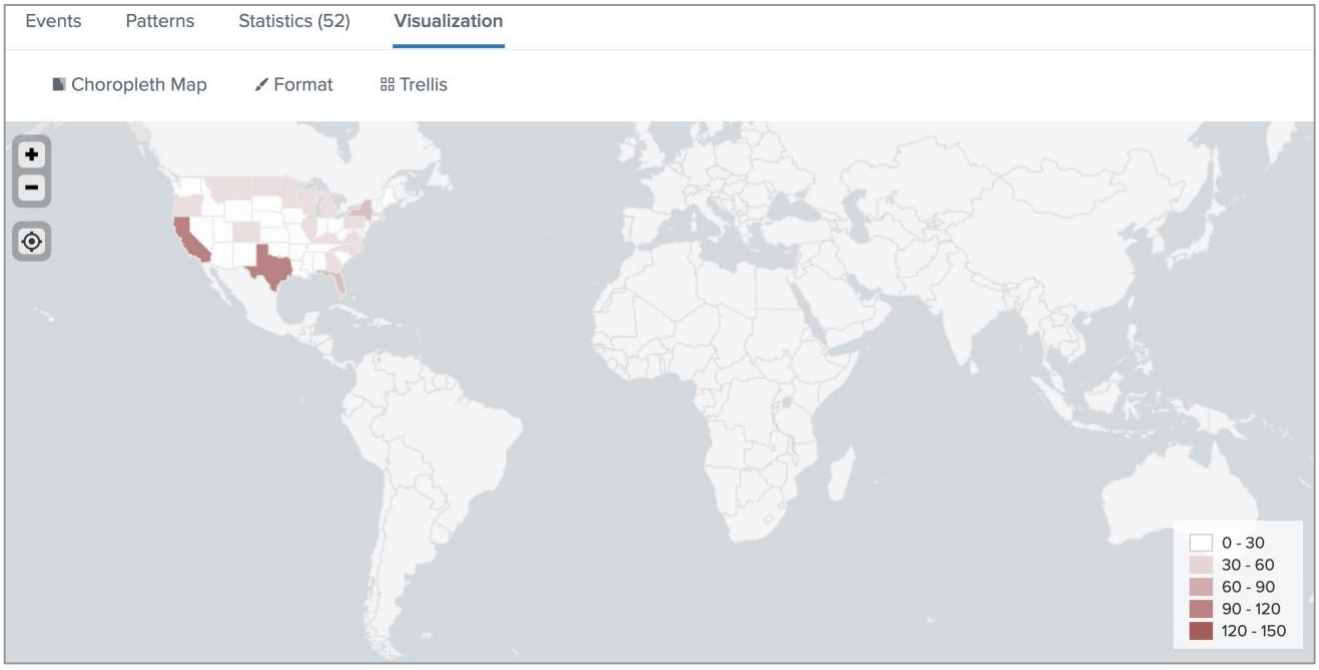
36. Use the **`chart`** command to count events by **`VendorStateProvince`**.

VendorStateProvince	count
Alabama	24
Alaska	34
Arizona	29
Arkansas	17
California	148
Colorado	54
Connecticut	7
Florida	79

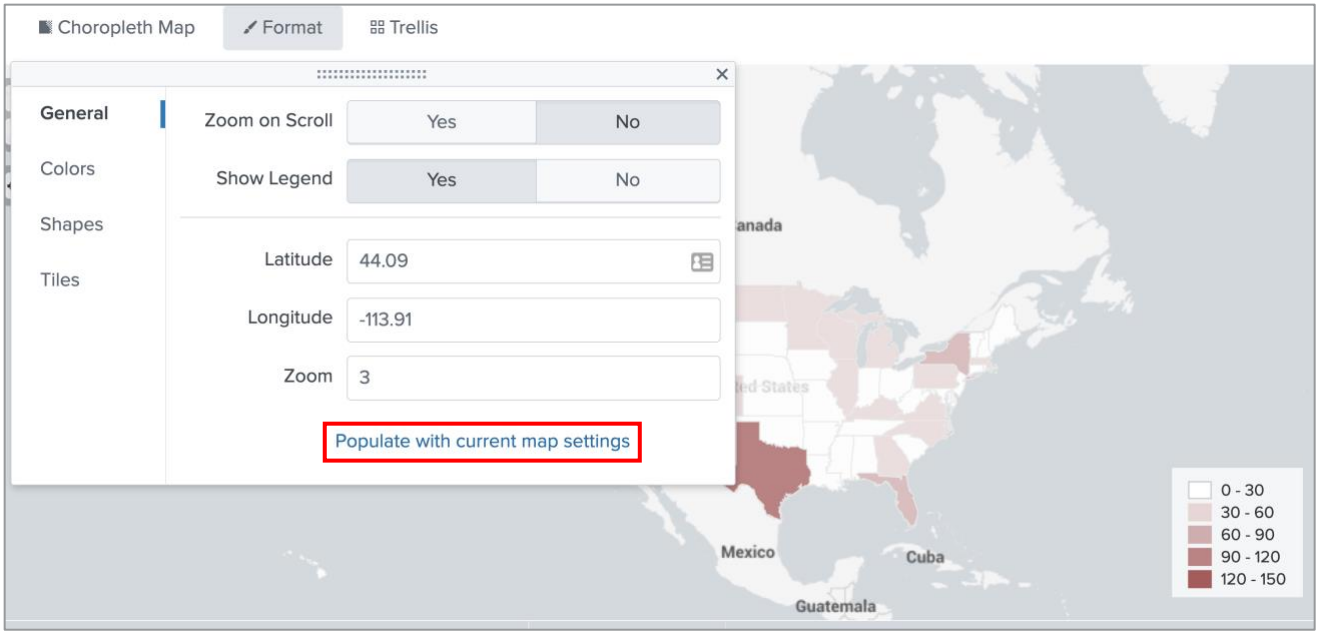
37. To display the results as a choropleth map, use the **`geom`** command to map **`VendorStateProvince`** to the geo_us_states KMZ file.

VendorStateProvince	count	featureCollection	geom
Alabama	24	geo_us_states	{"type":"MultiPolygon","coordinates":[[[[-88.31002807617188,30.233232498168945],[-88.31002807617188,30.2332324981689 45]]],[[[-88.47322845458984,31.893856048583984],[-88.20295715332031,35.008026123046875],[-85.60516357421875,34.984676361083984],[-85.00250244140625,31.0006828308105 47],[-88.02840423583984,30.221132278442383],[-88.47322845458984,31.893856048583984]]]}
Alaska	34	geo_us_states	{"type":"MultiPolygon","coordinates":[[[[179.4824676513672,51.98283386230469],[179.4824676513672,51.9828338623046 9]]],[[[178.6255340576172,51.63730239868164],[178.6255340576172,51.6373023986816 4]]],[[[178.44696044921875,51.97822189331055],[178.44696044921875,51.9782218933105 5]]],[[[178.2369384765625,51.828208923339844],[178.2369384765625,51.828208923339 844]]],[[[178.0946044921875,52.033294677734375],[178.0946044921875,52.0332946777343 75]]],[[[177.2033233642578,51.89656066894531],[177.2033233642578,51.8965606689453 1]]],[[[175.87435913085938,52.371002197265625],[175.87435913085938,52.3710021972656 25]]],[[[174.06918334960938,52.7348861694335 94],

38. Click on the **Visualizations** tab and display the results as a **Choropleth Map**.



39. Zoom into the United States using the **+** button. Then navigate to **Format** > **General** and click **Populate with current map settings**.



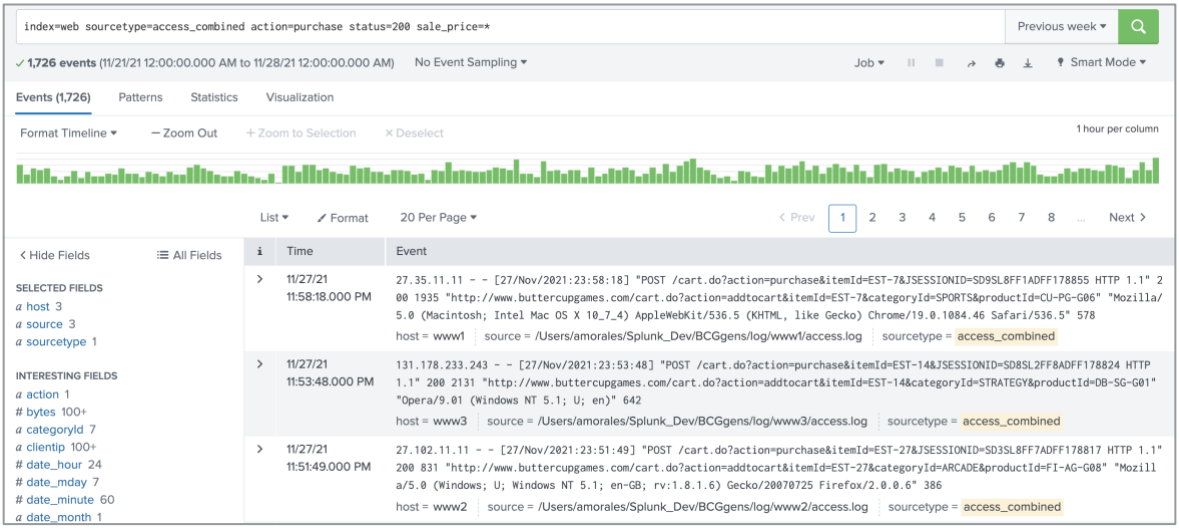
40. Save your search as a report with the name **L1S5**.

Scenario: Buttercup Games is currently running a global Cyber Monday sale and the Chief Financial Officer wants to see how online sales are performing across the globe.

Task 7: Use the `iplocation` and `geostats` commands to create a cluster map of online retail sales.

41. Re-initialize the search window by clicking **Search** in the application bar. This step should be done every time you save a report so that you do not accidentally overwrite a previous report.
42. Execute the following search over the **Previous week**. This search returns all online sales events (**`index=web sourcetype=access_combined action=purchase`**) that were successful (**`status=200`**) and on sale (**`sale_price=*`**).

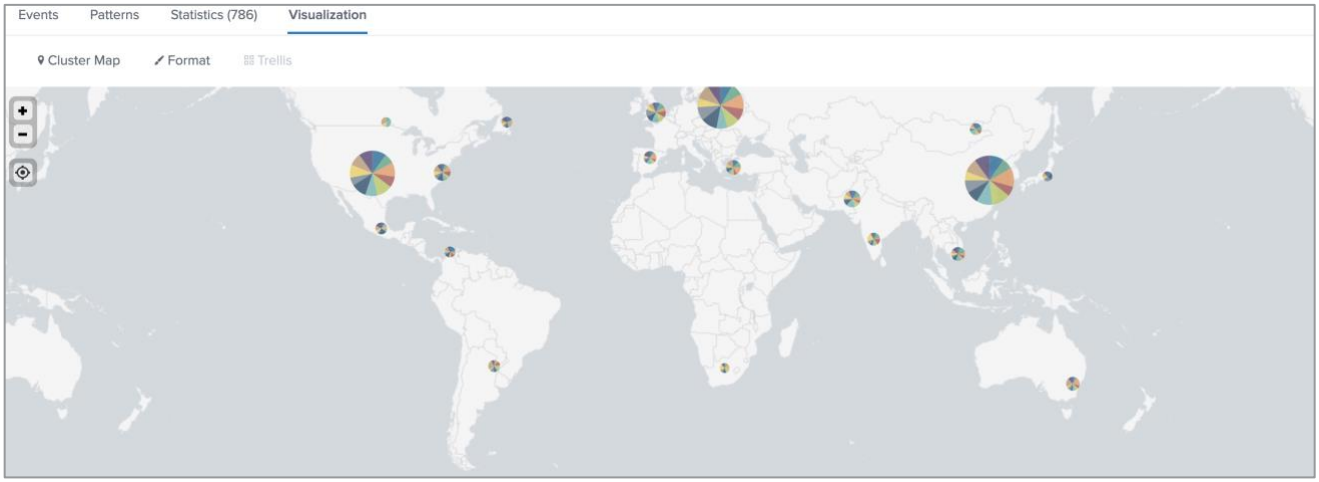
```
index=web sourcetype=access_combined action=purchase status=200 sale_price=*
```



43. Use the **`iplocation`** command to extract the location of purchases based on the **`clientip`** field. After you execute the search you should see **`lat`** and **`lon`** fields in the **Interesting Fields** list.

44. Use the **`geostats`** command to count events by **`product_name`.**

NOTE: The view changes to the **Visualization** tab when you run the **`geostats`** command. You may need to change the visualization to **Cluster Map**.



45. You may notice that some of the clusters have an **`OTHER`** field. This is because the number of named categories in a cluster is limited to 10 fields. Remove this limit using the **`globallimit`** argument.

*Before using **`globallimit`***

*After using **`globallimit`***

46. Save your search as a report with the name **L1S6**.

Scenario: The retail sales manager wants a report of all retail sales events during the last 4 hours by country. The final row in the report should contain the total count of retail events with a "Total" label.

Task 8: Complete a search using the `stats` and `addtotals` commands.

47. Re-initialize the search window by clicking **Search** in the application bar. This step should be done every time you save a report so that you do not accidentally overwrite a previous report.
48. Complete the missing portion of the following search so that:
    a. The **`stats`** command counts all events by **`VendorCountry`**.
    b. The count values calculated by the **`stats`** command are listed under a column called "Retail Events" without using the **`rename`** command.
    c. The last row of the results displays "Total" in the **`VendorCountry`** column and a sum of **`Retail Events`** values in the **`Retail Events`** column.

```
index=sales sourcetype=vendor_sales
| ???
| ???
```

49. Execute the search over the **Last 4 hours**.

VendorCountry	Retail Events
Brazil	1
China (PRC)	1
Mexico	1
Netherlands	1
South Africa	1
United States	14
Total	19

50. Save your search as a report with the name **L1S7**.

Challenge: Buttercup Games has hired a "Productivity Manager" and their first order of business is to monitor employee's web usage. They want a visualization of all non-Business activity over the last 24 hours. Create a trellised single-value visualization of this data.

51. Complete the missing portion of this search so that all non-Business web security appliance events are counted for each **`location`** and split by **`usage`** categories over the **Last 24 hours**.

```
index=network sourcetype=cisco_wsa_squid usage!=Business
| ???
```

location	Borderline	Personal	Unknown	Violation
Boston	0	39	2	1
London	0	14	11	1
SanFrancisco	21	49	15	2

52. Visualize your results by **`usage`** as pie charts in a trellis layout.

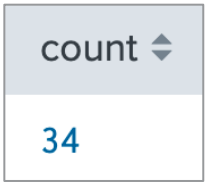
53. Save your search as **LX1**.

Challenge: The Buttercup Games San Francisco office recently reopened after being closed due to a chicken pox outbreak among the Engineering team. The CEO wants all 41 San Francisco employees back in the office now that quarantine is over. They want to see this data visualized as a single value with any number under 40 shown in red. (One employee is on parental leave after having quadruplets.)

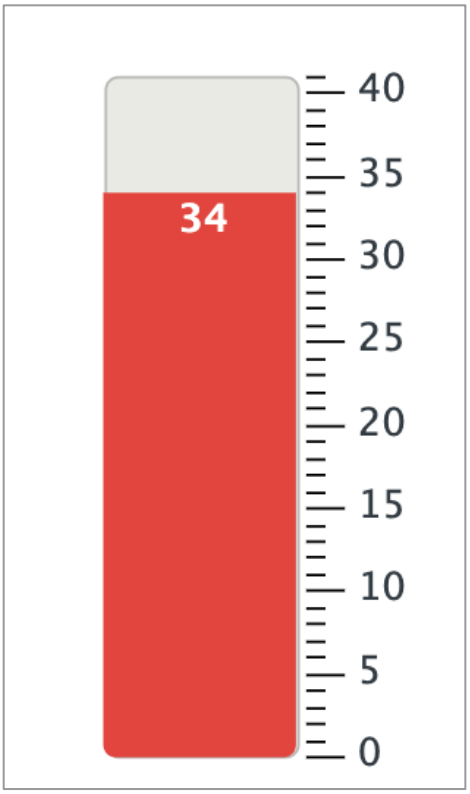
54. Complete the missing portion of this search so that the search:
    a. Removes duplicate values of **`rfid`** (the unique ID associated with each employee's badge.)
    b. Counts all events and produces a single value as the result.

```
index=security sourcetype=history_access Address_Description="San Francisco"
| ???
| ???
```

55. Execute your search over the **Last 24 hours**.



56. Visualize your results as a **Filler Gauge** and format the colors so that anything equal to or greater than 40 is green and anything less than 40 is red.



57. Save your search as **LX2**.

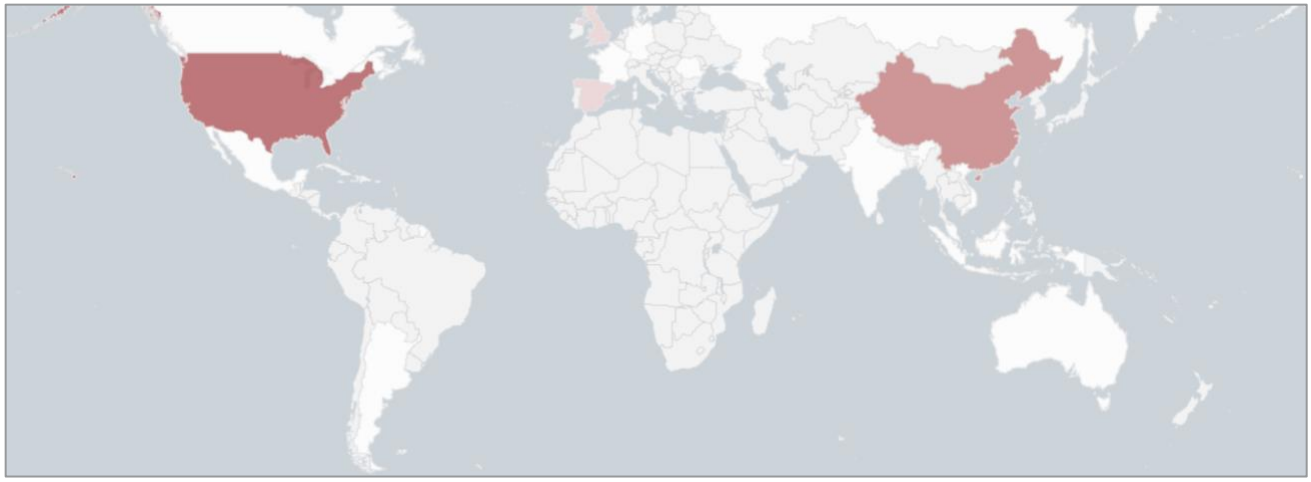
Choose your Challenge: The Sim Cubicle Game team recently launched a BETA and the Director of Engineering is eager to see how many unique events have occurred on the server in the past month.

58. Fulfill the scenario request by creating a choropleth map or a cluster map. The basic search for both solutions will be **`index=games sourcetype=SimCubeBeta`**. Run your search over the **Last 30 days**.

    Hints for each challenge:

    a. If you create a choropleth map, you will use 3 commands, 1 function, 1 **`by`** clause, the **`featureIdField`** argument, the **`geo_countries`** lookup and 3 fields (**`user_ip`** and **`Country`**.)

    b. If you create the cluster map, you will use 2 commands, 1 function and the **`user_ip`** field.



Choropleth Map

Cluster Map

59. Save your search as **LX3**.