splunk>

Searching and Reporting with Splunk 6.4 - Lab Solutions Guide

Lab typographical conventions:

`{student ID}` indicates you should replace this with your student number.

`{class#}` indicates you should substitute the server name assigned to this class.

[`sourcetype=vendor_sales`] OR [`cs_mime_type`] indicates either a source type or the name of a field.

There are a number of source types used in these lab exercises.

Please note.
This is a test environment driven by scripts with the obvious limitations. This is not a production environment.

The lab instructions refer to these source types by the types of data they represent:

Type	Sourcetype	Interesting Fields
Active Directory	`WinEventLog:Security`	`action`, `app`, `Authentication_Package`, `name`, `Reason`, `signature`, `src_ip`, `subject`
Badge reader	`history_access`	`Address_Description`, `Department`, `Device`, `Email`, `Event_Description`, `First_Name`, `last_Name`, `Rfid`, `Username`
BI server	`sales_entries`	`AcctCode`, `CustomerID`, `TransactionID`
Email	`cisco_esa`	`dcid`, `icid`, `mailfrom`, `mailto`, `mid`
Web appliance	`cisco_wsa_squid`	`action`, `bandwidth`, `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`
Online sales	`access_combined`	`action`, `bytes`, `categoryId`, `clientip`, `itemId`, `JSESSIONID`, `price`, `productId`, `product_name`, `referer`, `referer_domain`, `sale_price`, `status`, `user`, `useragent`
Retail sales	`vendor_sales`	`AcctID`, `categoryId`, `product_name`, `productId`, `sale_price`, `Vendor`, `VendorCity`, `VendorCountry`, `VendorID`, `VendorStateProvince`
Web server	`linux_secure`	`action`, `app`, `COMMAND`, `dest`, `process`, `src_city`, `src_country`, `src_ip`, `src_port`, `user`, `vendor_action`

Lab Exercise 1 – Search Fundamentals

Description

This lab exercise familiarizes you with the Buttercup Games data used in this course.

NOTE: If at any point you do not see results, check your search syntax and/or expand your time range.

Steps

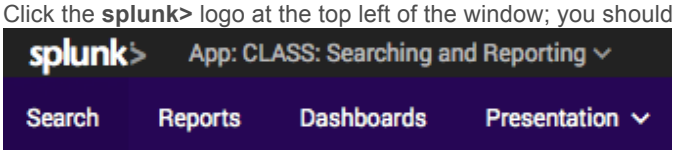
Task: **Log into Splunk on the classroom server.**

1. Direct your web browser to the class lab system (for example, `class#.splunk.com`)
2. Log in with the credentials your instructor assigned.

Task: **Make CLASS: Searching and Reporting your default app and change your account time zone setting to reflect your local time.**

3. From the Splunk bar (which is the black bar at the top of the browser window), click **your user login name**.
4. Click **Edit Account**.
5. In the **Full name** field, type your full name (i.e., first name followed by surname.)
6. From the **Time zone** menu, select your local time zone.
7. From the **Default app** drop down, select **class_SnR**.

8. Click **Save**.
9. Click the **splunk>** logo at the top left of the window; you should see:



NOTE: **CLASS: Searching and Reporting** is a custom app designed specifically for this training course. It contains custom menu options, such as the Presentation menu, which includes all of the search strings used in the slides. Searches saved in this app count toward completing this course. The dark blue menu bar indicates that you are in the **CLASS: Searching and Reporting** app.

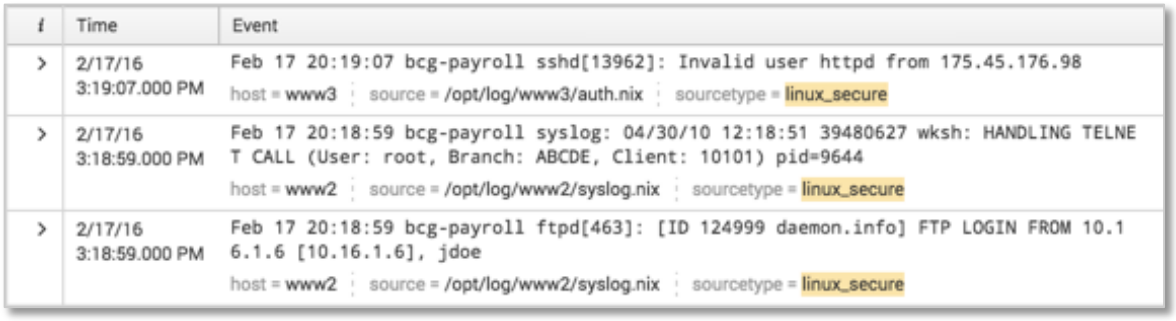
Task: **Review the different source types.**

10. From the **Search** view, click **Data Summary**.
11. Examine the available source types on the **Sourcetypes** tab.
12. Close the **Data Summary** window.

Task: **Explore web server events.**

13. Search for all web server events [`sourcetype=linux_secure`] during the **last 15 minutes**.
    NOTE: If this search does not return any events, expand the time range.
    sourcetype=linux_secure

Results Example:



i	Time	Event
>	2/17/16 3:19:07.000 PM	Feb 17 20:19:07 bcg-payroll sshd[13962]: Invalid user httpd from 175.45.176.98 host = www3 source = /opt/log/www3/auth.nix sourcetype = linux_secure
>	2/17/16 3:18:59.000 PM	Feb 17 20:18:59 bcg-payroll syslog: 04/30/10 12:18:51 39480627 wksh: HANDLING TELNET CALL (User: root, Branch: ABCDE, Client: 10101) pid=9644 host = www2 source = /opt/log/www2/syslog.nix sourcetype = linux_secure
>	2/17/16 3:18:59.000 PM	Feb 17 20:18:59 bcg-payroll ftpd[463]: [ID 124999 daemon.info] FTP LOGIN FROM 10.16.1.6 [10.16.1.6], jdoe host = www2 source = /opt/log/www2/syslog.nix sourcetype = linux_secure

NOTE: As you progress through the exercises, your results will vary from the example. However, they should look similar to it.

14. In the Fields sidebar, click **All Fields**.
15. Review the values for the following fields below. (You can do this by clicking the **>**, which is found to the left of each field name.)
    - action
    - app
    - dest
    - process
    - src_ip
    - user
    - vendor_action
16. Close the **Select Fields** window. You should now see the events that were returned by the search.
17. Expand one of the events and view the fields and values in the event. (To do this, click the **>** icon under the **i** column for the desired event.)

Task: **Explore web appliance events.**

18. To clear the previous search, click **Search** in the app navigation bar.
19. Search for all web appliance events [`sourcetype=cisco_wsa_squid`] during the **last 24 hours**.
    sourcetype=cisco_wsa_squid

Results Example:

i	Time	Event
>	2/17/16 3:42:19.701 PM	1455741739.701 16 201.3.120.132 TCP_REFRESH_HIT/200 411 GET http://www.computerworld.com/common/images/pix.gif blu@buttercupgames.com DIRECT/www.computerworld.com - ALLOW_WBRS-DefaultGroup-Demo_Clients-NONE-NONE-DefaultRouting <IW_comp,6.5,-,-,-,-,-,-,-,-,-,-,-,-,-,IW_comp,-> - http://www.computerworld.com/ host = cisco_router1 source = /opt/log/cisco_router1/cisco_ironport_web.log sourcetype = cisco_wsa_squid
>	2/17/16 3:41:41.683 PM	1455741701.683 16 123.30.108.208 TCP_REFRESH_HIT/200 927 GET http://www.computerworld.com/common/images/home/sharktank.gif blu@buttercupgames.com DIRECT/www.computerworld.com - ALLOW_WBRS-DefaultGroup-Demo_Clients-NONE-NONE-DefaultRouting <IW_comp,6.5,-,-,-,-,-,-,-,-,-,-,-,-,-,IW_comp,-> - http://www.computerworld.com/ host = cisco_router1 source = /opt/log/cisco_router1/cisco_ironport_web.log sourcetype = cisco_wsa_squid

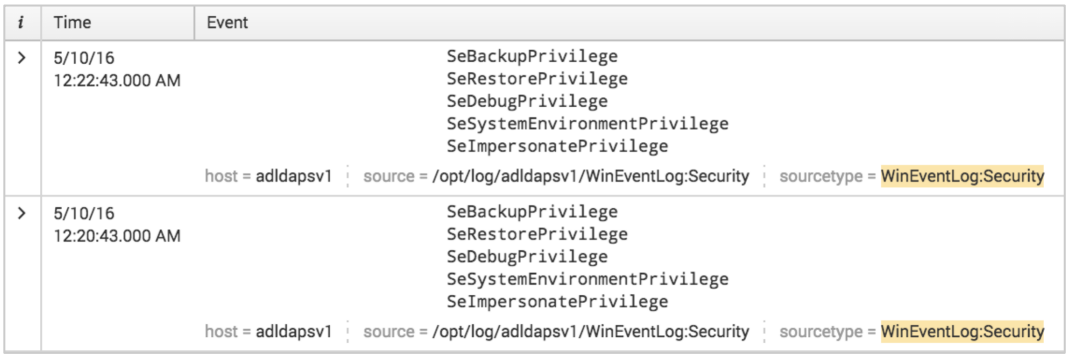
20. Examine the fields in the Fields sidebar.
21. Review the values for the following fields:
    - action
    - bcg_ip
    - bcg_workstation
    - cs_username
    - rfid
    - usage

- username
- x_webcat_code_full

Task: **Explore AD/DNS events.**

22. To clear the previous search, click **Search** in the app navigation bar.
23. Search for Active Directory events [`sourcetype=WinEventLog:Security`] during the **last 24 hours**.
    sourcetype=WinEventLog:Security

Results Example:



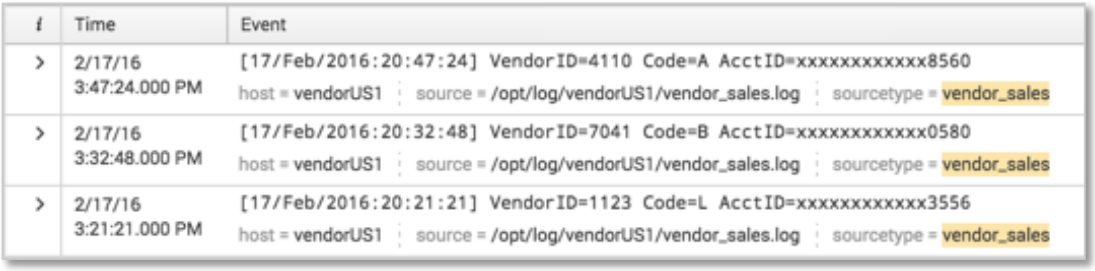
24. Review the values for the following fields:
    - action
    - app
    - Authentication_Package
    - Message
    - user

    NOTE: If you do not see a field in the Fields sidebar, remember to click **All Fields**.

Task: **Explore retail sales events.**

25. To clear the previous search, click **Search** in the app navigation bar.
26. Search for retail sales events [`sourcetype=vendor_sales`] during the **last 24 hours**.
    sourcetype=vendor_sales

Results Example:



27. Review the values for the following fields:
    - AcctID
    - categoryId
    - product_name
    - Vendor
    - VendorCountry
28. Search for retail sales from Canada.
    sourcetype=vendor_sales VendorCountry=Canada

Task: **Explore online sales events**.

29. To clear the previous search, click **Search** in the app navigation bar.
30. Search for online sales events [`sourcetype=access_combined`] during the **last 24 hours**.

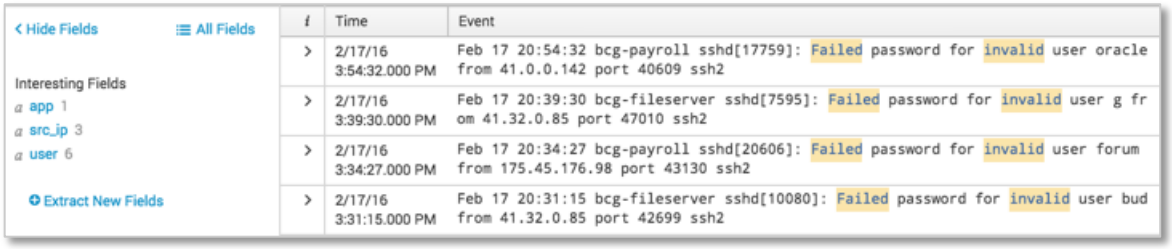
Results Example:

>	2/17/16 3:58:19.000 PM	67.133.102.54 - - [17/Feb/2016:20:58:19] "GET /category.screen?categoryId=STRATEGY&JSESSIONID=SD0SL3FF3ADFF4952 HTTP 1.1" 200 891 "http://www.buttercupgames.com/oldlink?itemId=EST-14" "Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US; rv:1.9.2.28) Gecko/20120306 YFF3 Firefox/3.6.28 (.NET CLR 3.5.30729; .NET4.0C)" 777 host = www2 source = /opt/log/www2/access.log sourcetype = access_combined
>	2/17/16 3:58:14.000 PM	67.133.102.54 - - [17/Feb/2016:20:58:14] "GET /product.screen?productId=CU-PG-G06&JSESSIONID=SD0SL3FF3ADFF4952 HTTP 1.1" 200 744 "http://www.buttercupgames.com/cart.do?action=changequantity&itemId=EST-11&productId=CU-PG-G06" "Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US; rv:1.9.2.28) Gecko/20120306 YFF3 Firefox/3.6.28 (.NET CLR 3.5.30729; .NET4.0C)" 409 host = www2 source = /opt/log/www2/access.log sourcetype = access_combined

31. Notice the number of events returned.
32. Review the values for the following fields:
    - action
    - categoryId
    - clientip
    - price
    - referer
    - referer_domain
    - status
    - useragent
33. Modify your search to return only purchase events.
    HINT: `action=purchase`
    sourcetype=access_combined action=purchase
34. Notice the number of events returned. Because you have narrowed your search to only purchases, there are fewer events than before.
    NOTE: The number of events returned depends on the time range you selected. Also, remember that unless you have selected a discrete time range, **Previous …** or **Yesterday**, when you re-execute a search, events are being added to the returned results.

Task: Check for authentication failures on the web servers in the last 60 minutes.

Final Results Example:



Hide Fields | All Fields

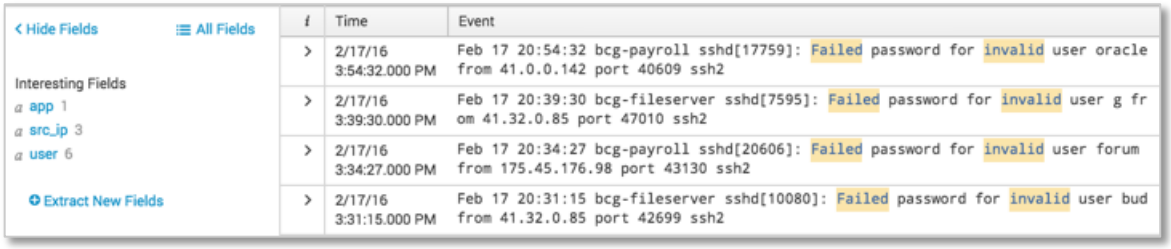
Interesting Fields
a app 1
a src_ip 3
a user 6

Extract New Fields

i	Time	Event
>	2/17/16 3:54:32.000 PM	Feb 17 20:54:32 bcg-payroll sshd[17759]: Failed password for invalid user oracle from 41.0.0.142 port 40609 ssh2
>	2/17/16 3:39:30.000 PM	Feb 17 20:39:30 bcg-fileserver sshd[7595]: Failed password for invalid user g from 41.32.0.85 port 47010 ssh2
>	2/17/16 3:34:27.000 PM	Feb 17 20:34:27 bcg-payroll sshd[20606]: Failed password for invalid user forum from 175.45.176.98 port 43130 ssh2
>	2/17/16 3:31:15.000 PM	Feb 17 20:31:15 bcg-fileserver sshd[10080]: Failed password for invalid user bud from 41.32.0.85 port 42699 ssh2

35. Search the web server [`sourcetype=linux_secure`] during the **last 60 minutes**
36. Modify your search to look for `failed` password attempts by `invalid` users.
    sourcetype=linux_secure fail* invalid
37. Using the `fields` command, extract only the `user`, `src_ip`, and `app` fields.
    sourcetype=linux_secure fail* invalid
    | fields user, src_ip, app

Results Example:



38. Save your search as report, with the title **L1S1**. (**Save As > Report**)
39. Click **Save** and then click **View**.

NOTE: All your saved searches will appear in the Student Labs menu (assuming you name your searches as directed). If View does not refresh the Student Labs menu, you may need to refresh your browser.

Task: **Display the last search in a table, with duplicate results removed and with new headings.**

Final Results Example:

Potential Attacker	Name Used	Application Used
1.0.32.67	test	sshd
1.0.32.67	update	sshd
1.0.32.67	user	sshd
2.144.0.22	abc	sshd
2.144.0.22	amanda	sshd
2.144.0.22	charles	sshd

40. To expand your search history, click **Search** -- and then, on the lower left of the Search window, click **Expand your search history**.
41. Find your last search and click **Add to Search**.
42. Modify the search to display the results in a table, using the same fields, `user`, `src_ip`, and `app`, during the **last 4 hours**.
    sourcetype=linux_secure fail* invalid
    | table user, src_ip, app

Results Example:

user	src_ip	app
operator	41.32.0.85	sshd
amanda	175.45.176.98	sshd
emma	175.45.176.98	sshd
forum	175.45.176.98	sshd
oracle	41.0.0.142	sshd
g	41.32.0.85	sshd

43. To view which IPs may be attackers, sort the results by `src_ip` and `user`.
    sourcetype=linux_secure fail* invalid
    | table user, src_ip, app
    | sort src_ip, user

Results Example:

user	src_ip	app
test	1.0.32.67	sshd
update	1.0.32.67	sshd
user	1.0.32.67	sshd
abc	2.144.0.22	sshd
amanda	2.144.0.22	sshd
amanda	2.144.0.22	sshd
charles	2.144.0.22	sshd

44. Notice that the results contain duplicates. Use `dedup` to remove duplicate IPs and user name combinations from the results.

    Best Practice: For best performance, place `dedup` as early in the search as possible.

    ```
    sourcetype=linux_secure fail* invalid
    | dedup user, src_ip
    | table user, src_ip, app
    | sort src_ip, user
    ```

    Results Example:

user	src_ip	app
test	1.0.32.67	sshd
update	1.0.32.67	sshd
user	1.0.32.67	sshd
abc	2.144.0.22	sshd
amanda	2.144.0.22	sshd
charles	2.144.0.22	sshd

45. Finally, to make the table easier for non-IT employees to read and understand, reorder the fields to `src_ip`, `user`, and `app`.

    ```
    sourcetype=linux_secure fail* invalid
    | dedup user, src_ip
    | table src_ip, user, app
    | sort src_ip, user
    ```

46. Rename the app field to "Application Used", user to "Name Used", and src_ip to "Potential Attacker".
    NOTE: Remember, if the new field name includes spaces, it must be enclosed in double-quotes.

    ```
    sourcetype=linux_secure fail* invalid
    | dedup user, src_ip
    | table src_ip, user, app
    | sort scr_ip, user
    | rename  app as "Application Used", user as "Name Used",  src_ip as "Potential Attacker"
    ```

    Results Example:

Potential Attacker	Name Used	Application Used
1.0.32.67	test	sshd
1.0.32.67	update	sshd
1.0.32.67	user	sshd
2.144.0.22	abc	sshd
2.144.0.22	amanda	sshd
2.144.0.22	charles	sshd

47. Save your search as report, **L1S2**. (**Save As > Report**)
48. Click **View**.

Task: **Check for issues with customer purchases in the online store.**

49. To display the search bar, in the app navigation bar, click **Search**.
50. Search online sales [`sourcetype=access_combined`] during the **last 4 hours**, and using the `table` command, display only the `clientip`, `host`, `action`, and `status` fields.
    sourcetype=access_combined
    | table clientip, host, action, status

Results Example:

clientip	host	action	status
94.229.0.20	www3	changequantity	406
94.229.0.20	www3	purchase	200
94.229.0.20	www3	purchase	200
94.229.0.20	www3	addtocart	200
94.229.0.20	www3		200
94.229.0.20	www3	changequantity	200

51. Notice your results may show events with no value in the `action` field. These events can represent a number of different issues. For now, modify your search to only view purchase events [`action=purchase`]. Display only the `clientip`, `host`, and `status` fields.
    sourcetype=access_combined action=purchase
    | table clientip, host, status

Results Example:

clientip	host	status
94.229.0.20	www3	200
94.229.0.20	www3	200
175.44.24.82	www3	200
175.44.24.82	www3	200
175.44.24.82	www3	200
175.44.24.82	www3	200

52. To make the search results easier for non-IT employees to understand, rename the `clientip` field to `"Customer IP"`, `host` to `"Web Server"`, and `status` to `"HTTP Status"`.
    sourcetype=access_combined action=purchase
    | table clientip host status
    | rename clientip as "Customer IP", host as "Web Server", status as "HTTP Status"

Results Example:

Customer IP	Web Server	HTTP Status
123.30.108.208	www1	503
123.30.108.208	www1	200
123.30.108.208	www1	200
94.229.0.20	www3	200

53. With the new results, you can explore and investigate whether any of the web servers are experiencing issues. Modify your current search to look for problems [`status>399`] on the servers.
    NOTE: If your search returns no results, expand your time range to **last 24 hours**.

```
sourcetype=access_combined action=purchase status>399
| table clientip host status
| rename clientip as "Customer IP", host as "Web Server", status as "HTTP Status"
```

Results Example:

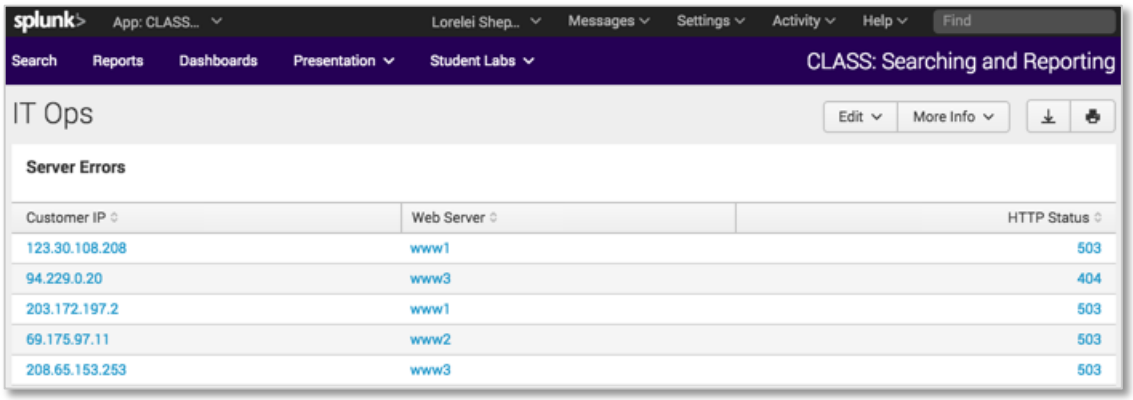
Customer IP	Web Server	HTTP Status
123.30.108.208	www1	503
94.229.0.20	www3	404
203.172.197.2	www1	503

54. Click **Save As** to save your search as report, **L1S3**.
55. Do not dismiss the dialog.

Task: Add your search to a new dashboard.

56. From the dialog box, click **Add to Dashboard**.
57. Save the dashboard with these values:
    - **Dashboard:** *New*
    - **Dashboard Title:** *IT Ops*
    - **Panel Title:** *Server Errors*
    - **Panel Powered by:** *Inline Search*

    NOTE: You are adding a report that will be inline. This means that it will be independent from the report you just saved called **L1S3**.
58. Click **Save** and then click **View Dashboard** to view your dashboard.



Task: Check how employees are using corporate resources.

Final Results Example:

username	usage
acurry	Unknown
basselin	Personal
blu	Borderline
blu	Unknown
dhale	Personal
djohnson	Borderline

59. To display the search bar, in the app navigation bar, click **Search**.
60. Search for web appliance events `[sourcetype=cisco_wsa_squid]` during the **last 4 hours**.
    sourcetype=cisco_wsa_squid

61. Create a `table` with the web appliance events that displays [`username`] and [`usage`].
    NOTE: `usage` is a classification of websites.
    sourcetype=cisco_wsa_squid
    | table username, usage

Results Example:

username	usage
gvoronoff	Personal
gvoronoff	Personal
gvoronoff	Personal
gvoronoff	Personal
gvoronoff	Personal
gvoronoff	Personal

62. As you did previously, remove duplicates based on username and usage, and sort by username.
    sourcetype=cisco_wsa_squid
    | dedup username usage
    | table username, usage
    | sort username

Results Example:

username	usage
acurry	Unknown
basselin	Personal
blu	Borderline
blu	Unknown
dhale	Personal
djohnson	Borderline

63. Save your search as report, **L1S4**.

Lab Exercise 2 – Transforming Commands, Part 1 Deriving Statistics

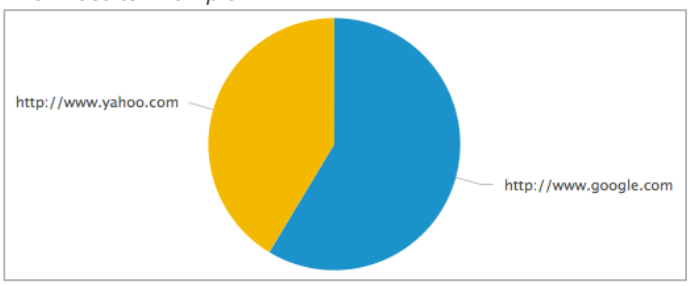
Description

This lab exercise reinforces the `top`, `rare`, and `stats` commands.

Steps

Task: **Find out from where visitors to our website are coming.**

Final Results Example:



1. Search online sales [`sourcetype=access_combined`] during the **last 24 hours** for all referer domains [`referer_domain`] except `http://www.buttercupgames.com`.
   sourcetype=access_combined referer_domain!="http://www.buttercupgames.com"
2. Use the `top` command `limit` option to display the top two referer domains.
   sourcetype=access_combined referer_domain!="http://www.buttercupgames.com"
   | top limit=2 referer_domain

Results Example:

referer_domain	count	percent
http://www.google.com	139	50.545455
http://www.yahoo.com	98	35.636364

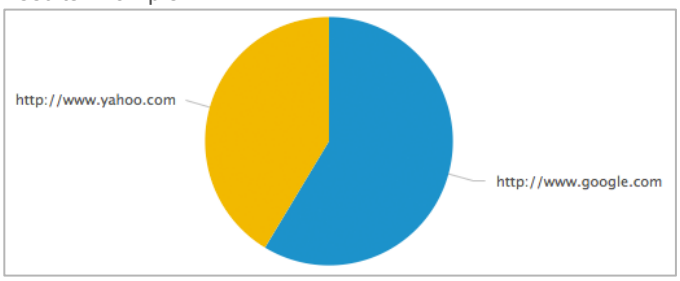
3. Use the `showperc` option of `top` to remove `percent` from the display.
   sourcetype=access_combined referer_domain!="http://www.buttercupgames.com"
   | top limit=2 referer_domain showperc=0

Results Example:

referer_domain	count
http://www.google.com	139
http://www.yahoo.com	98

4. Switch to the **Visualization** tab. The Column Chart visualization is selected.
5. Click **Column Chart**; a list of visualizations appear.
6. From the list of visualizations, click the **pie chart**  .

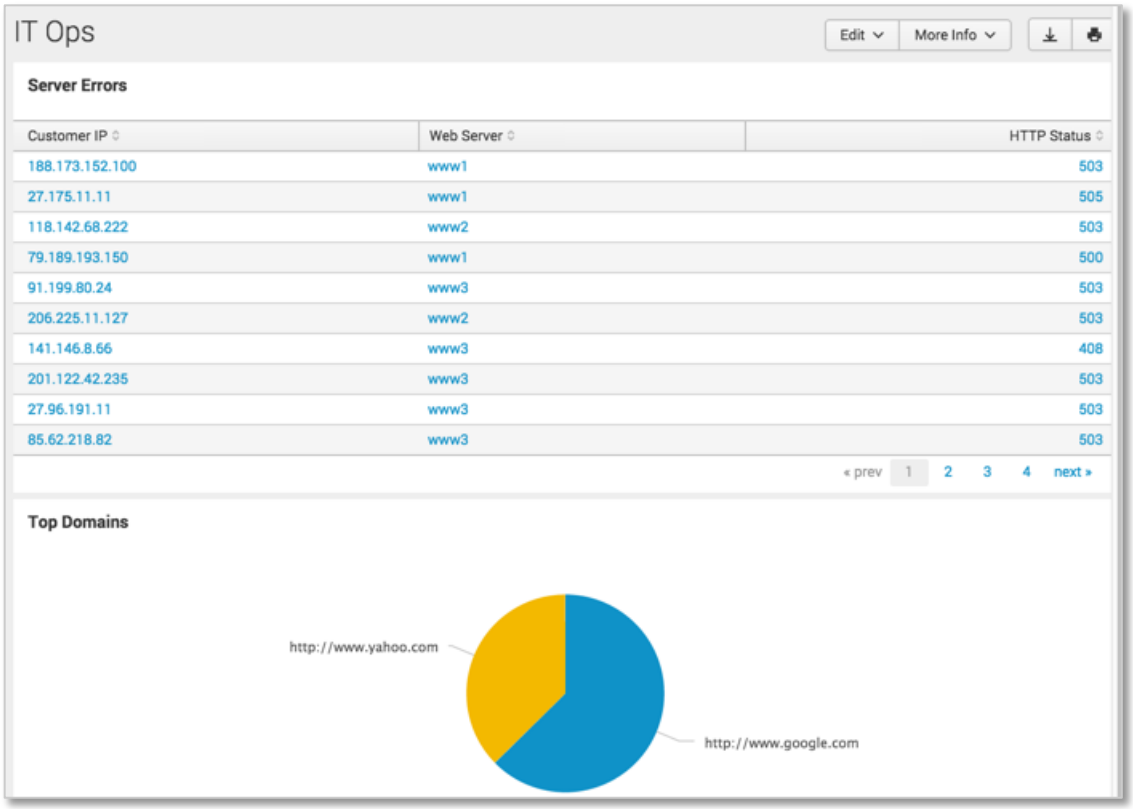
Results Example:



7. Save your search as report, **L2S1**.

Task: **Add this search to your IT Ops dashboard.**

8. Add this report to your IT Ops dashboard. Name the panel: **Top Domains**. Save the dashboard. Your dashboard should now look like this:



Task: **Display the top status codes for each of our web servers.**

Final Results Example:

host	status	count
www1	200	1176
www3	200	1056
www2	200	1017
www1	503	26
www3	503	23
www2	408	22

9. Search online sales [`sourcetype=access_combined`] during the **last 24 hours**.
10. Use `top` to display the top 2 `status` codes and hide the percent field.
    sourcetype=access_combined
    | top limit=2 status showperc=f

Results Example:

status	count
200	3246
503	67

11. Add a `by` clause to display the top two `status` codes for each `host`.
    sourcetype=access_combined
    | top limit=2 status by host showperc=f

Results Example:

host	status	count
www1	200	1174
www1	503	26
www2	200	1017
www2	408	22
www3	200	1056
www3	503	23

12. Sort the results in descending order on the `count` column.
    sourcetype=access_combined
    | top limit=2 status by host showperc=f
    | sort -count

Results Example:

host	status	count
www1	200	1176
www3	200	1056
www2	200	1017
www1	503	26
www3	503	23
www2	408	22

13. Save your search as report, **L2S2**.

Task: **Identify the types of content employees are viewing. Report the rare types, as these can potentially be malicious.**

14. Search the web appliance events [`sourcetype=cisco_wsa_squid`] during the **last 24 hours**.
    sourcetype=cisco_wsa_squid
15. Use the `rare` command to display the 3 least common content types [`cs_mime_type`].
    sourcetype=cisco_wsa_squid
    | rare limit=3 cs_mime_type

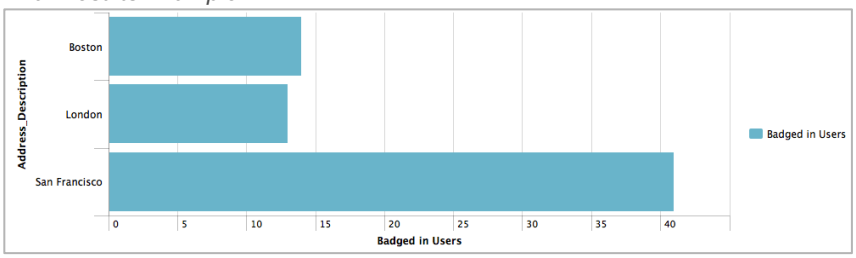
Results Example:

cs_mime_type	count	percent
image/bmp	1	0.084818
text/javascript	2	0.169635
application/x-shockwave-flash	3	0.254453

16. Save your search as report, **L2S3**.

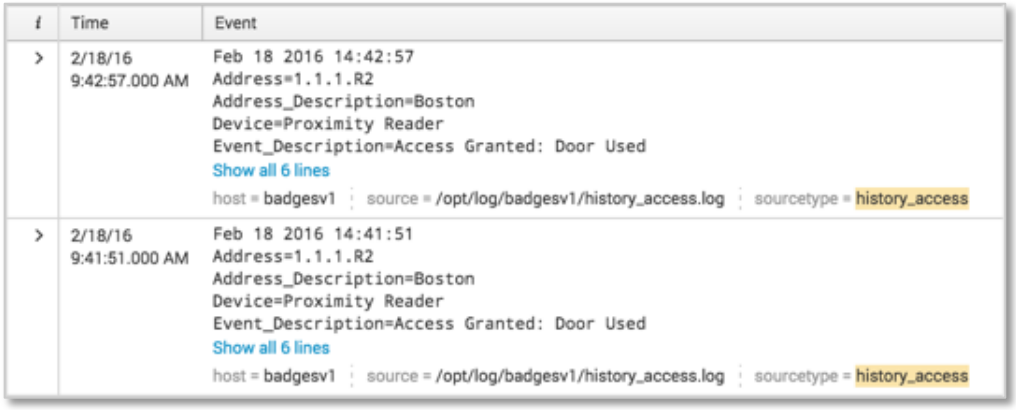
Task: Count the number of distinct employee badge swipes, by location, during the last 24 hours.

Final Results Example:



17. Search for employee badge swipes [`sourcetype=history_access`] during the **24 hours**.

Results Example:



18. Modify your search to display the distinct count of `Username` by location (`Address_Description`).
    sourcetype=history_access
    | stats dc(Username) by Address_Description

Results Example:

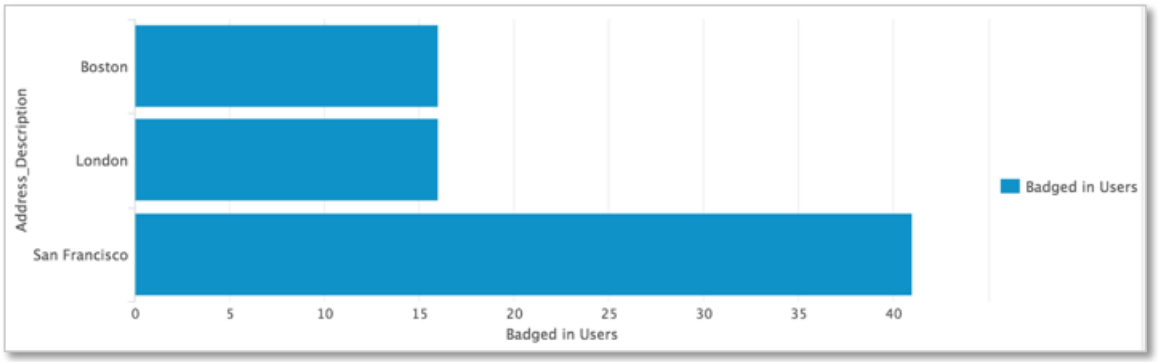
Address_Description	dc(Username)
Boston	16
London	16
San Francisco	41

19. Without using a separate `rename` command, rename the `dc(Username)` column to `"Badged in Users"`.
    sourcetype=history_access
    | stats dc(Username) as "Badged in Users" by Address_Description

Results Example:

Address_Description	Badged in Users
Boston	16
London	16
San Francisco	41

20. Click the **Visualization** tab, and change the visualization to a **Bar Chart**.
    Results Example:



21. Save your search as report, **L2S4**.

Task: **List the actions (without duplicates) on the AD/DNS server during the 60 minutes.**

22. Search Active Directory data [`sourcetype=WinEventLog:Security`] for events during the **last 60 minutes**.
    sourcetype=WinEventLog:Security

Results Example:

i	Time	Event
>	5/10/16 12:18:17.000 AM	Account Domain: SPLUNK Additional Information: Privileges: - host = adldapsv1 source = /opt/log/adldapsv1/WinEventLog:Security sourcetype = WinEventLog:Security
>	5/10/16 12:16:43.000 AM	SeBackupPrivilege SeRestorePrivilege SeDebugPrivilege SeSystemEnvironmentPrivilege SeImpersonatePrivilege host = adldapsv1 source = /opt/log/adldapsv1/WinEventLog:Security sourcetype = WinEventLog:Security

23. Use `stats` to display a list of unique subjects [`subject`]. Rename the column `Subject`.
    sourcetype=WinEventLog:Security
    | stats values(subject) as Subject

Results Example:

Subject
Logon attempt using explicit credentials The logon attempt failed for other reasons. Unknown user name or bad password

24. Save your search as report, **L2S5**.

****CHALLENGE Exercise: Calculate the number of events, the average price, and the total price for each action in the online store during the previous week.**

Final Results Example:

Action	Total Events	Average Price	Total Amount
addtocart	3896	21.255869	76351.08
changequantity	967	20.832627	16811.93
purchase	3854	21.188686	41911.22
remove	946	21.998974	17159.20
view	3908	21.230758	69488.27

25. Search online sales [`access_combined`] for events containing a value in the `action` field during the **previous week**.
    sourcetype=access_combined action=*
26. Count the results by `action`.
    sourcetype=access_combined action=*
    | stats count by action

    Results Example:

action	count
addtocart	3896
changequantity	967
purchase	3854
remove	946
view	3908

27. Rename the count column (that displays the action counts) to "Total Events" and action to Action.
    sourcetype=access_combined action=*
    | stats count by action
    | rename action as Action, count as "Total Events"

Results Example:

Action	Total Events
addtocart	3896
changequantity	967
purchase	3854
remove	946
view	3908

28. Modify your search to also compute the average price as “Average Price” and the sum of price as "Total Amount".
    sourcetype=access_combined action=*
    | stats count, avg(price), sum(price) by action
    | rename count as "Total Events",
      avg(price) as "Average Price",
      sum(price) as "Total Amount",
      action as Action

Results Example:

Action	Total Events	Average Price	Total Amount
addtocart	3896	21.255869	76351.08
changequantity	967	20.832627	16811.93
purchase	3854	21.188686	41911.22
remove	946	21.998974	17159.20
view	3908	21.230758	69488.27

29. Save your search as report, **L2C1**.

Lab Exercise 3 – Transforming Commands, Part 2 Creating Visualizations

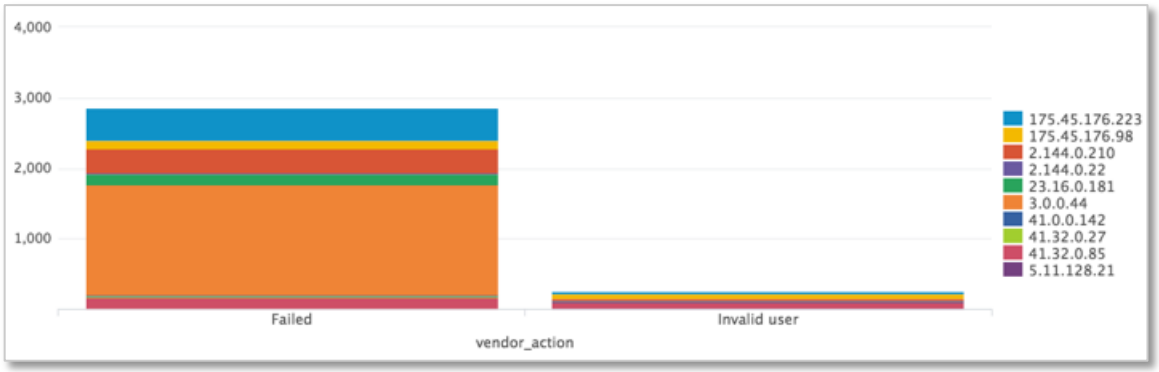
Description

In this lab exercise, you use the `chart` and `timechart` command.

Steps

Task: **Report the failures on the web server during the last 24 hours and add it to a new security dashboard as a column chart.**

Final Results Example:



1. Search the web server [`sourcetype=linux_secure`] that have either failed or "invalid user" [`vendor_action`] for events during the **last 24 hours**.
   sourcetype=linux_secure (vendor_action=failed OR vendor_action="invalid user")

Results Example:

i	Time	Event
>	2/18/16 10:56:14.000 AM	Feb 18 15:56:14 bcg-fileserver sshd[7768]: Invalid user accept from 41.32.0.85 host = www1 source = /opt/log/www1/auth.nix sourcetype = linux_secure
>	2/18/16 10:55:15.000 AM	Feb 18 15:55:15 bcg-payroll sshd[22365]: Failed password for root from 175.45.176.223 port 43264 ssh2 host = www2 source = /opt/log/www2/auth.nix sourcetype = linux_secure
>	2/18/16 10:50:14.000 AM	Feb 18 15:50:14 bcg-payroll sshd[32522]: Failed password for rjayaraman from 3.0.0.44 port 45954 ssh2 host = www1 source = /opt/log/www1/auth.nix sourcetype = linux_secure

2. Using the `chart` command, display a count for each action [`vendor_action`] the users performed by IP [`src_ip`].
   HINT: Use `over … by`
   sourcetype=linux_secure (vendor_action=failed OR vendor_action="invalid user")
   | chart count over vendor_action by src_ip

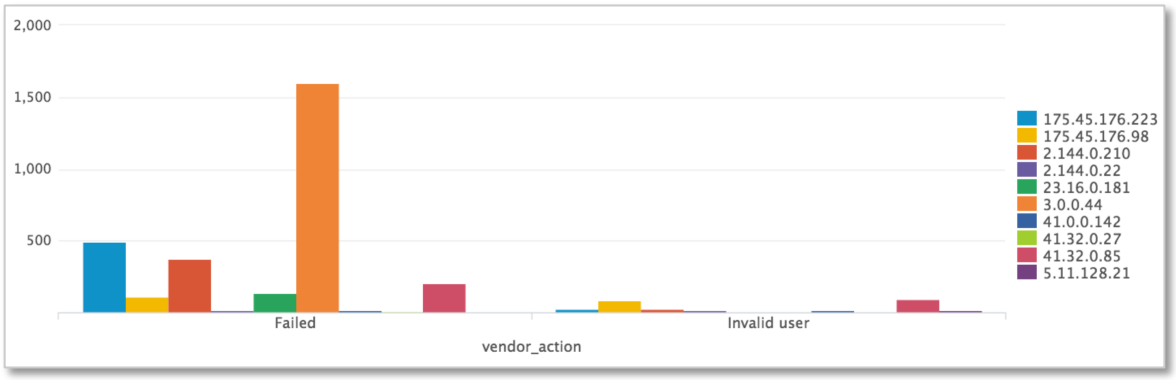
Results Example:

vendor_action	175.45.176.223	175.45.176.98	2.144.0.210	2.144.0.22	23.16.0.181	3.0.0.44	41.0.0.142	41.32.0.27	41.32.0.85	5.11.128.21	OTHER
Failed	466	104	352	11	149	1560	16	11	173	3	19
Invalid user	23	74	29	10	0	2	13	4	85	11	2

3. Click on the **Visualization** tab and make sure **Column Chart**  is selected.

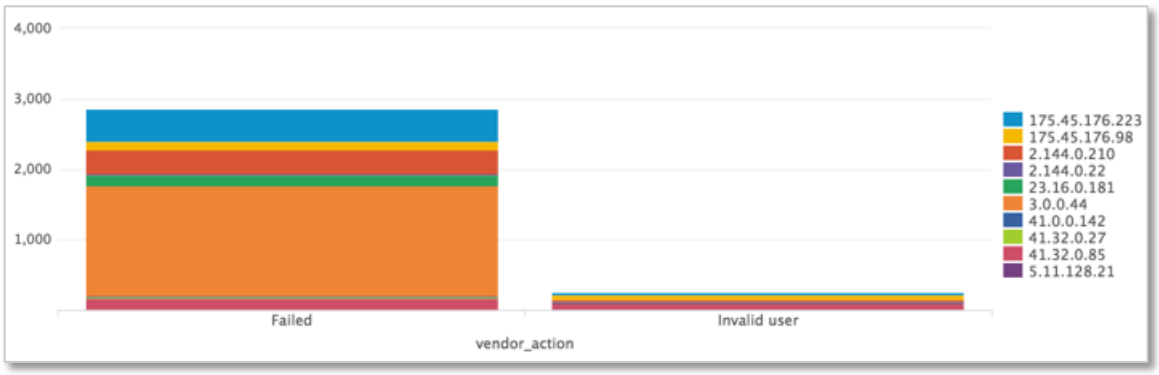
4. As you can see, there is an OTHER column at the end of the results. Set the `useother` option to `f`, to remove this column.
   sourcetype=linux_secure (vendor_action=failed OR vendor_action="invalid user")
   | chart count over vendor_action by src_ip useother=f

Results Example:



5. Click **Format**; in the General section, set the visualization to **Stacked**.

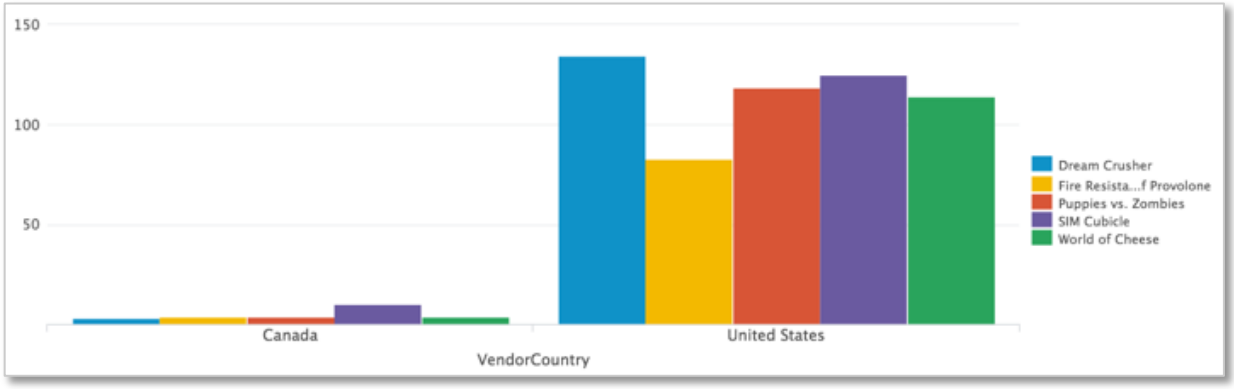
Results Example:



6. Save your search as report, **L3S1**, but instead of continuing to edit, click **Add to Dashboard**.
7. Save the dashboard with these values:
   - **Dashboard:** *New*
   - **Dashboard Title:** *Sec Ops*
   - **Panel Title:** *Potential Penetration Attempts*
   - **Panel Powered By:** *Report*
8. View your dashboard.
9. Mouse over your column chart and click one of the bars. Notice the drilldown feature is activated.
10. Use your browser's Back button to return to your dashboard.
11. Click Edit > Edit Panels.
12. Click  .
13. For **Drilldown**, click **No**.
14. Close the dialog.
15. Verify that the drilldown is not working.
16. Click **Done** to save your changes.

Task: **Chart by country the five best selling products for our vendors in North America during the last 7 days.**

Final Results Example:



VendorID:

1000-2999	USA
3000-3999	Canada
4000-4999	Caribbean, Central & South America
5000-6999	Europe and the Middle East
7000-8999	Asia and Pacific Region
9000-9900	Africa
9901-9999	Outliers, such as the South Pole

17. Search for retail store events [`vendor_sales`] from North America (United States and Canada) during the **last 7 days**.
    sourcetype=vendor_sales VendorID<4000

    Results Example:

i	Time	Event
>	2/18/16 11:21:04.000 AM	[18/Feb/2016:16:21:04] VendorID=1090 Code=A AcctID=xxxxxxxxxxxx2923 host = vendorUS1 source = /opt/log/vendorUS1/vendor_sales.log sourcetype = vendor_sales
>	2/18/16 11:05:42.000 AM	[18/Feb/2016:16:05:42] VendorID=1026 Code=C AcctID=xxxxxxxxxxxx1620 host = vendorUS1 source = /opt/log/vendorUS1/vendor_sales.log sourcetype = vendor_sales
>	2/18/16 10:39:36.000 AM	[18/Feb/2016:15:39:36] VendorID=1132 Code=D AcctID=xxxxxxxxxxxx2410 host = vendorUS1 source = /opt/log/vendorUS1/vendor_sales.log sourcetype = vendor_sales

18. Using the `chart` command, count the events over `VendorCountry`.
    sourcetype=vendor_sales VendorID<4000
    | chart count over VendorCountry

Results Example:

VendorCountry	count
Canada	77
United States	1109

19. To see the products sold in each country, add a `by` clause to further split the data by `product_name`.
    sourcetype=vendor_sales VendorID<4000
    | chart count over VendorCountry by product_name

Results Example:

VendorCountry	Dream Crusher	Fire Resistance Suit of Provolone	Holy Blade of Gouda	Manganiello Bros.	Manganiello Bros. Tee	OTHER	Orvil the Wolverine	Puppies vs. Zombies	SIM Cubicle	World of Cheese	World of Cheese Tee
Canada	3	4	4	3	2	25	9	4	10	4	9
United States	135	83	71	69	75	187	64	119	125	114	67

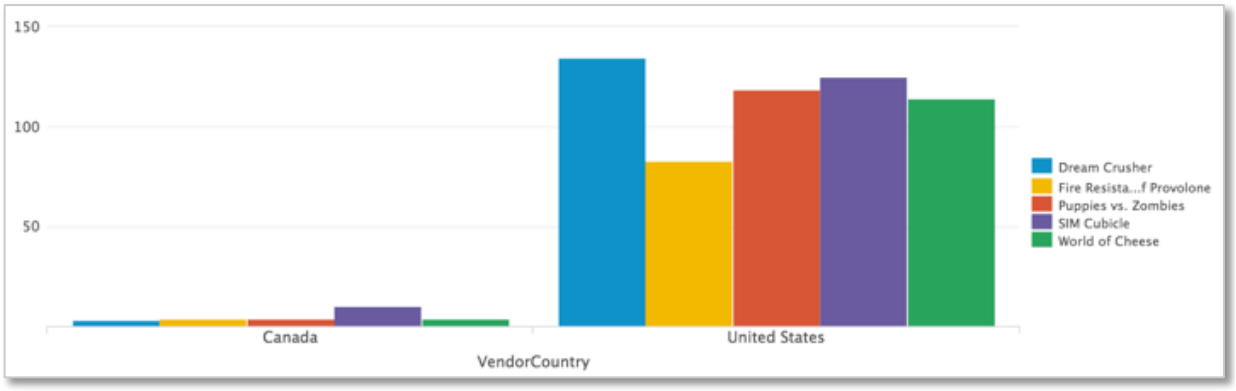
20. Use a `limit` option to include only the 5 best selling products.
NOTE: Splunk automatically adds the columns and filters based on the largest number.
```
sourcetype=vendor_sales VendorID<4000
| chart count over VendorCountry by product_name limit=5
```
Results Example:

VendorCountry	Dream Crusher	Fire Resistance Suit of Provolone	OTHER	Puppies vs. Zombies	SIM Cubicle	World of Cheese
Canada	3	4	52	4	10	4
United States	135	83	533	119	125	114

21. Remove the OTHER column from your table.
```
sourcetype=vendor_sales VendorID<4000
| chart count over VendorCountry by product_name limit=5 useother=f
```
Results Example:

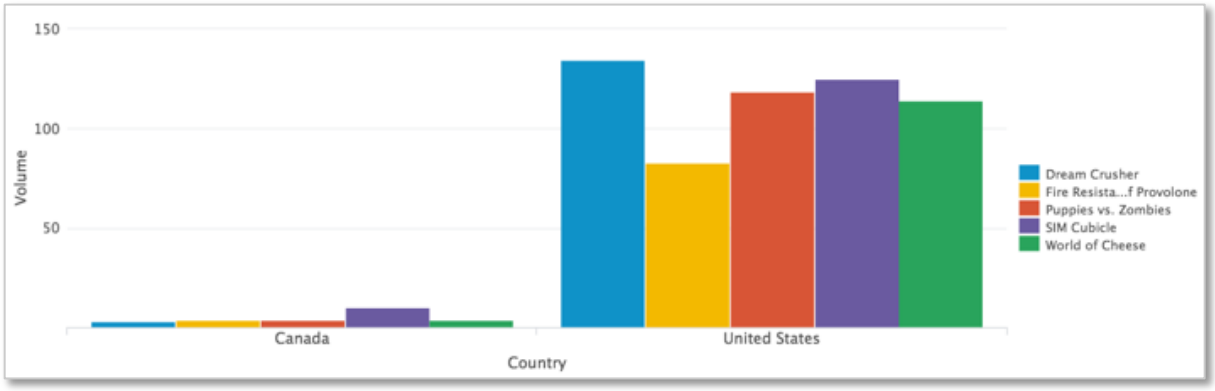
VendorCountry	Dream Crusher	Fire Resistance Suit of Provolone	Puppies vs. Zombies	SIM Cubicle	World of Cheese
Canada	3	4	4	10	4
United States	135	83	119	125	114

22. Switch to the **Visualization** tab and, if a column chart was not automatically shown, set the chart type to **Column Chart**.
Results Example:



23. Use the **Format** options to define custom labels of **Country** and **Volume** for the X and Y axes, respectively.

Results Example:

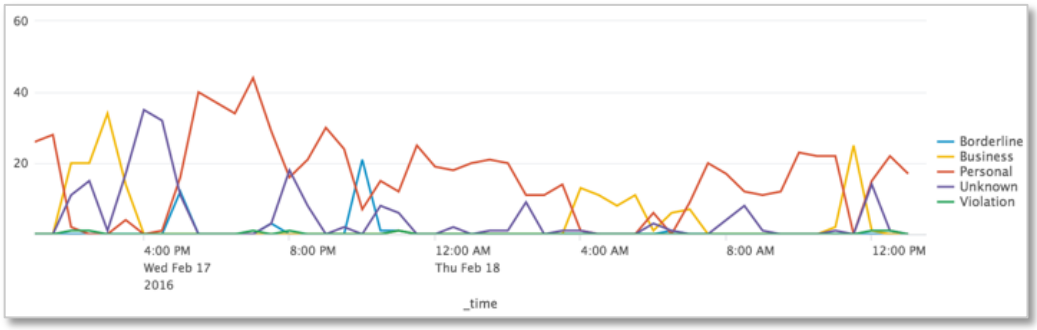


24. Save your search as report, **L3S2**.

Task: **Display Internet usage in a timechart during the last 24 hours.**

25. Search for web appliance events [`cisco_wsa_squid`] during the **last 24 hours**.
    sourcetype=cisco_wsa_squid
26. Use the `timechart` command to count the events by `usage`.
    sourcetype=cisco_wsa_squid
    | timechart count by usage
27. Change the visualization to **Line Chart**.

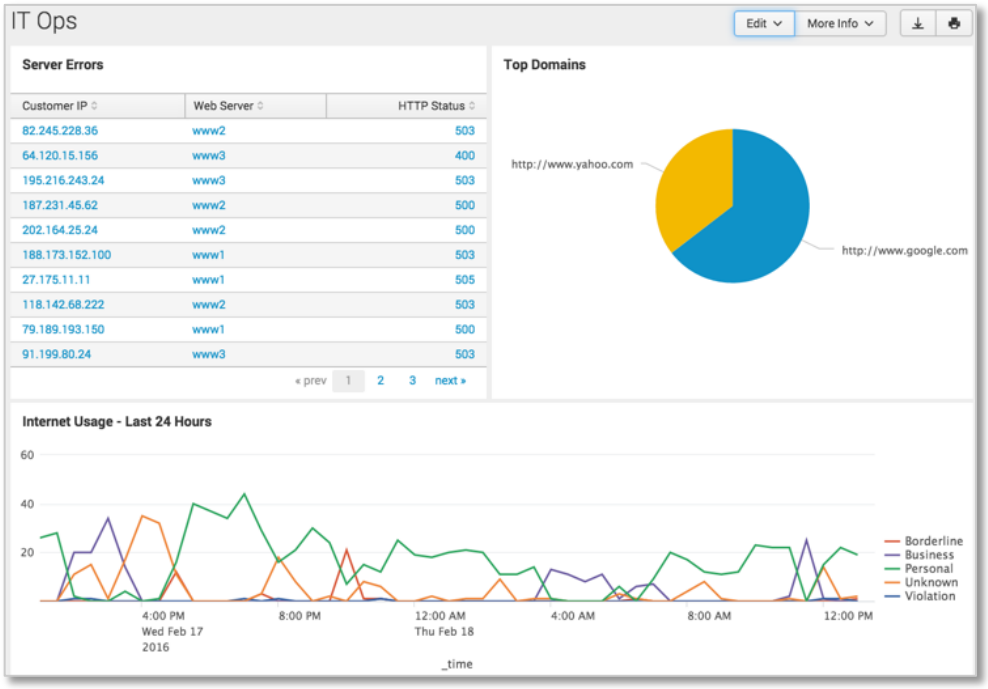
Results Example:



28. Save the search as report, **L3S3**.
29. Add this report to your *IT Ops* dashboard in a panel named: **Internet Usage - Last 24 Hours**

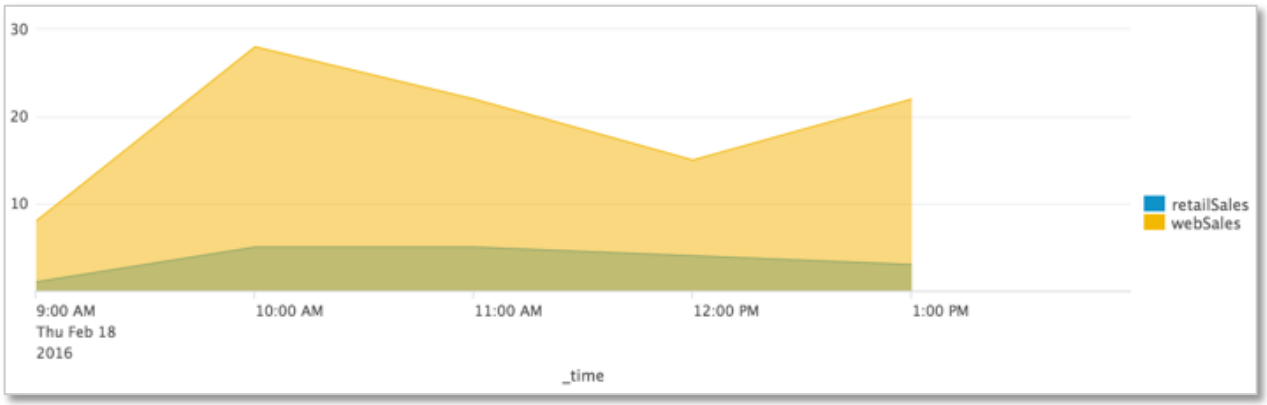
30. View your dashboard and arrange your panels so that your dashboard looks like this:

Results Example:



****CHALLENGE Exercise: Display and compare on-line and vendor sales during the last 4 hours.**

Final Results Example:

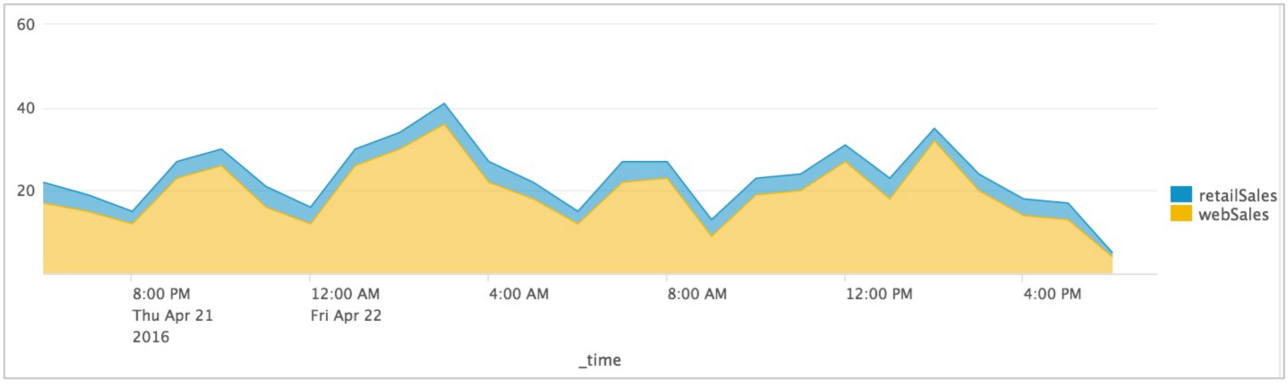


31. Search for successful online purchase events [`access_combined`] during the **last 4 hours** and enclose the entire search string in parentheses. (As you continue to modify this search string in the upcoming lab steps, the parentheses will be helpful.)
    (sourcetype=access_combined action=purchase status=200)
32. Modify the search string to also search for all retail sales [`vendor_sales`].
    (sourcetype=access_combined action=purchase status=200) OR sourcetype=vendor_sales
33. Use `timechart` to count the sales events by `sourcetype`. Change the sampling interval to 1 hour. **HINT**: View the results in the **Statistics** tab to see the time values.
    (sourcetype=access_combined action=purchase status=200) OR sourcetype=vendor_sales
    | timechart span=1h count by sourcetype
34. Rename the access_combined column to webSales and the vendor_sales column to retailSales.

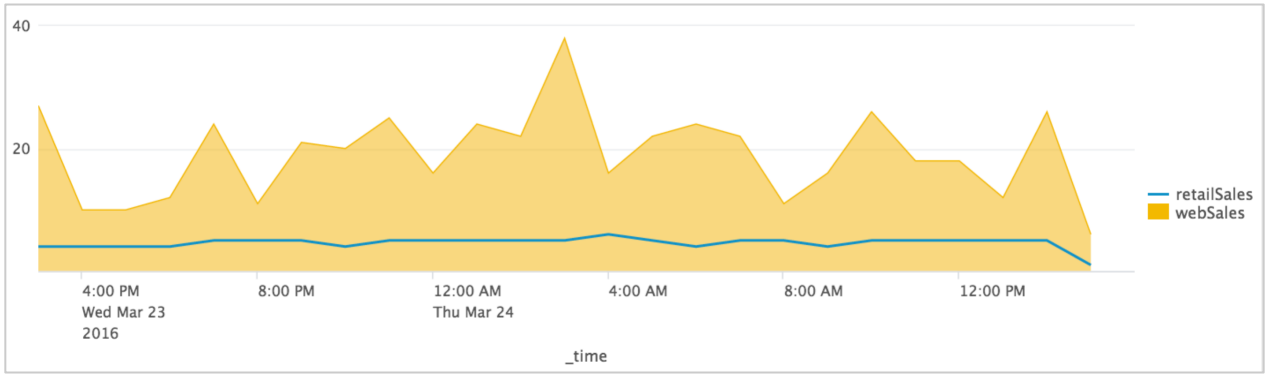
```
(sourcetype=access_combined action=purchase status=200) OR sourcetype=vendor_sales
| timechart span=1h count by sourcetype
| rename access_combined as webSales, vendor_sales as retailSales
```

35. Display the results as an **Area Chart**.

Results Example:



36. Save the search as report, **L3C1**.
37. Optionally: revise the formatting to show retail sales as a chart overlay, and save as **L3C2**.



splunk>

Lab Exercise 4 – Transforming Commands, Part 3 Enriching Visualizations

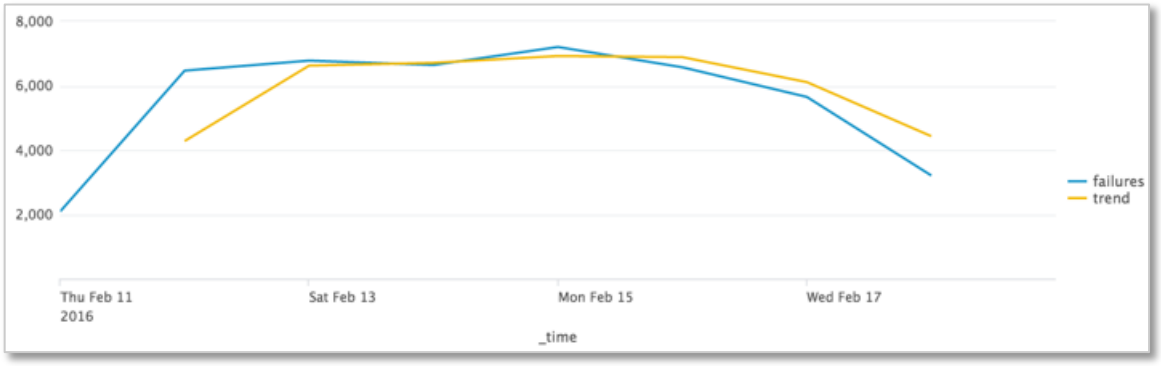
Description

In this lab exercise, you use `trendline`, `iplocation`, `geostats`, `geom` and `addtotals` commands.

Steps

Task: **Show the failures and the trend on the web server during the last 7 days.**

Final Results Example:



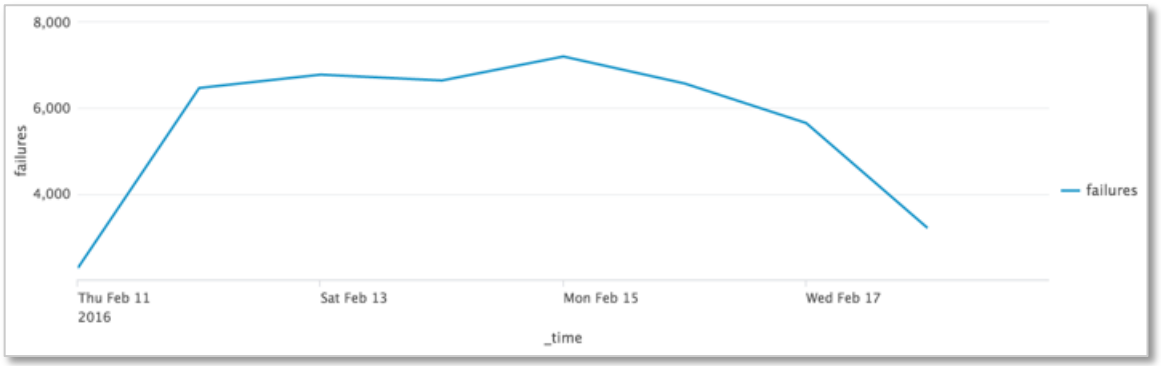
1. Search for failures on the web server [`linux_secure`] during the **last 7 days**.
   sourcetype=linux_secure fail*

Results Example:

i	Time	Event
>	2/18/16 1:49:43.000 PM	Feb 18 18:49:43 bcg-payroll sshd[907]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=3.0.0.44 user=root host = www3 source = /opt/log/www3/auth.nix sourcetype = linux_secure
>	2/18/16 1:49:17.000 PM	Feb 18 18:49:17 bcg-payroll sshd[5327]: Failed password for root from 3.0.0.44 port 48468 ssh2 host = www1 source = /opt/log/www1/auth.nix sourcetype = linux_secure

2. Using `timechart` count the events as failures. Change the visualization to **Line Chart**. (In the result, note that the default span is one day -- that is, you see the results charted by day.)
   sourcetype=linux_secure fail*
   | timechart count as failures

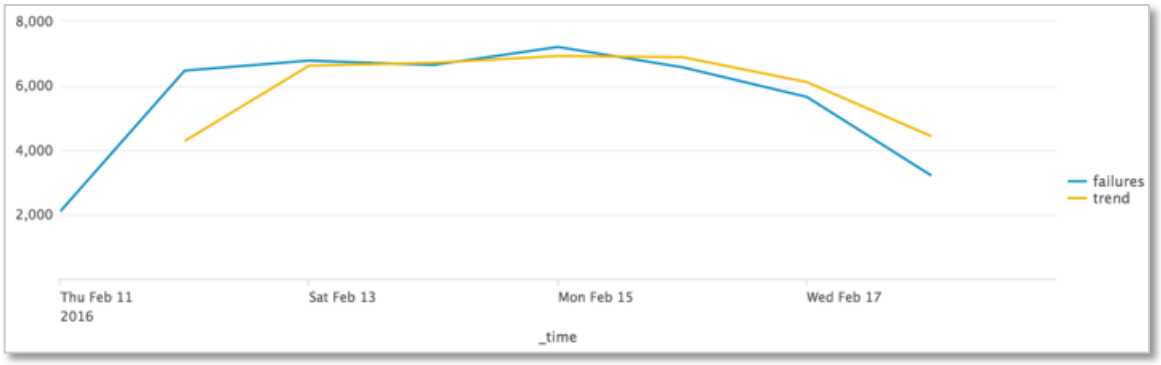
Results Example:



3. Using `trendline` with `sma2`, trend the failures as trend.

```
sourcetype=linux_secure fail*
| timechart count as failures
| trendline sma2(failures) as trend
```

Results Example:



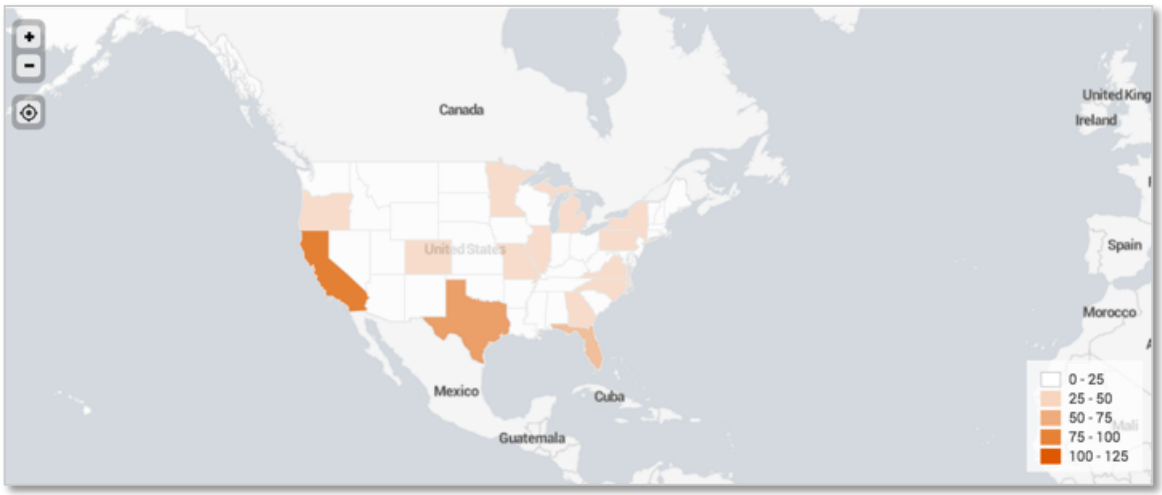
4. Save your search as report, **L4S1** and continue editing.
5. Change the visualization to single value with the following format:
   - **Caption:** *Web Server Failures - Previous Day*
   - **Show Trend indicator:** *Yes*
   - **Show Sparkline:** *Yes*
   - **Color:** *Set ranges, 100, 500, 1000, 2000*
   - **Color Mode:** *Set so that the background shows the color based on the range (e.g., red)*

Results Example:

6. Save your search as report, **L4S2**.

Task: **Display a choropleth map of United States retail sales during the last 7 Days.**

Final Results Example:



7. Search for United States retail sales during the **last 7 Days**.

   HINT: North American vendors have a VendorID less than 3000.
   sourcetype=vendor_sales VendorID < 3000

Results Example:

i	Time	Event
>	2/18/16 2:19:43.000 PM	[18/Feb/2016:19:19:43] VendorID=1100 Code=J AcctID=xxxxxxxxxxx8067 host = vendorUS1 source = /opt/log/vendorUS1/vendor_sales.log sourcetype = vendor_sales
>	2/18/16 1:49:35.000 PM	[18/Feb/2016:18:49:35] VendorID=1193 Code=L AcctID=xxxxxxxxxxx3642 host = vendorUS1 source = /opt/log/vendorUS1/vendor_sales.log sourcetype = vendor_sales

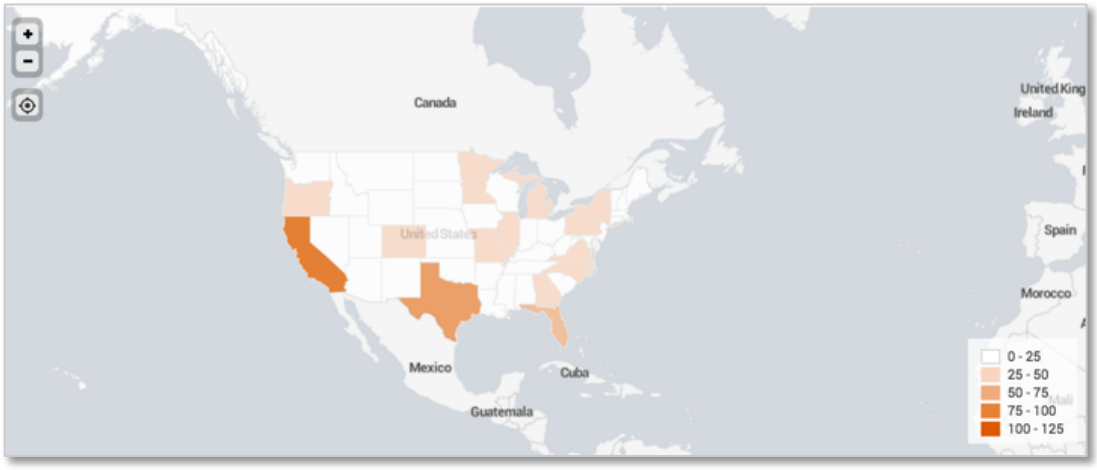
8. Using the `chart` command, count the events over `VendorStateProvince`.
   sourcetype=vendor_sales VendorID<3000
   | chart count over VendorStateProvince

Results Example:

VendorStateProvince	count
Alabama	16
Alaska	18
Alberta	14
Arizona	21
Arkansas	8
British Columbia	9

9. To display the data as a choropleth map, use the `geom` command to map VendorStateProvince to the geo_us_states KMZ file (`geom geo_us_states featureIdField=VendorStateProvince`).
   sourcetype=vendor_sales VendorID<3000
   | chart count by VendorStateProvince
   | geom geo_us_states featureIdField=VendorStateProvince
10. Click the **Visualization** tab.
11. Change the visualization to use the **Choropleth Map**. 
12. Zoom in the on the map so you can see clearly the United States.

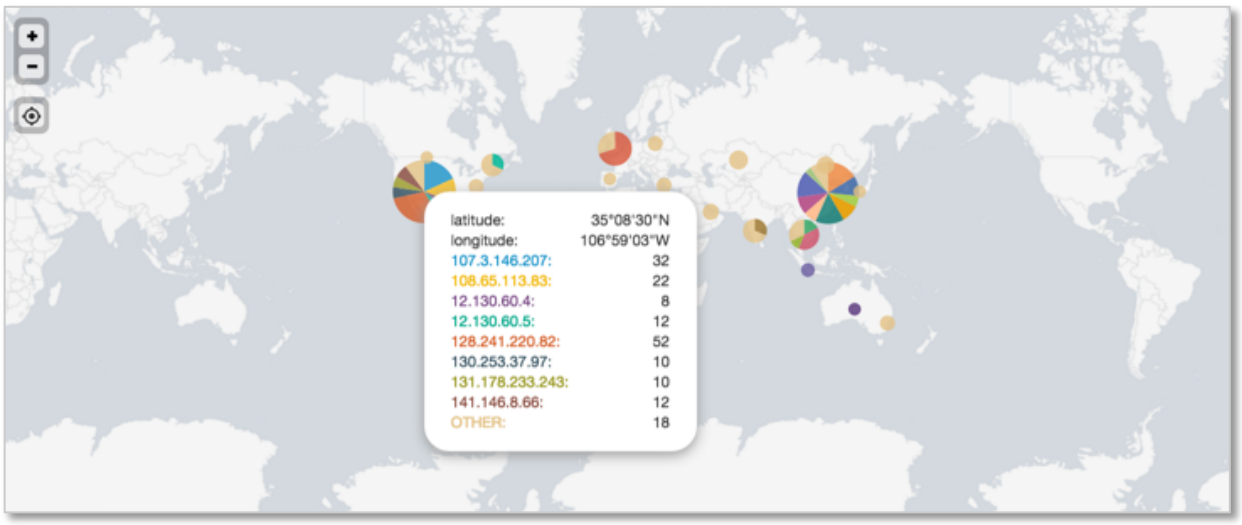
Results Example:



13. Save your search as report, **L4S3**.

Task: **Display a map of online sales by country during the previous week.**

Final Results Example:



14. Find successful online purchases [`access_combined`] during the previous week.
HINT: You can use the Fields sidebar to narrow your search results. From `action`, select purchase and from `status`, 200.
sourcetype=access_combined action=purchase status=200

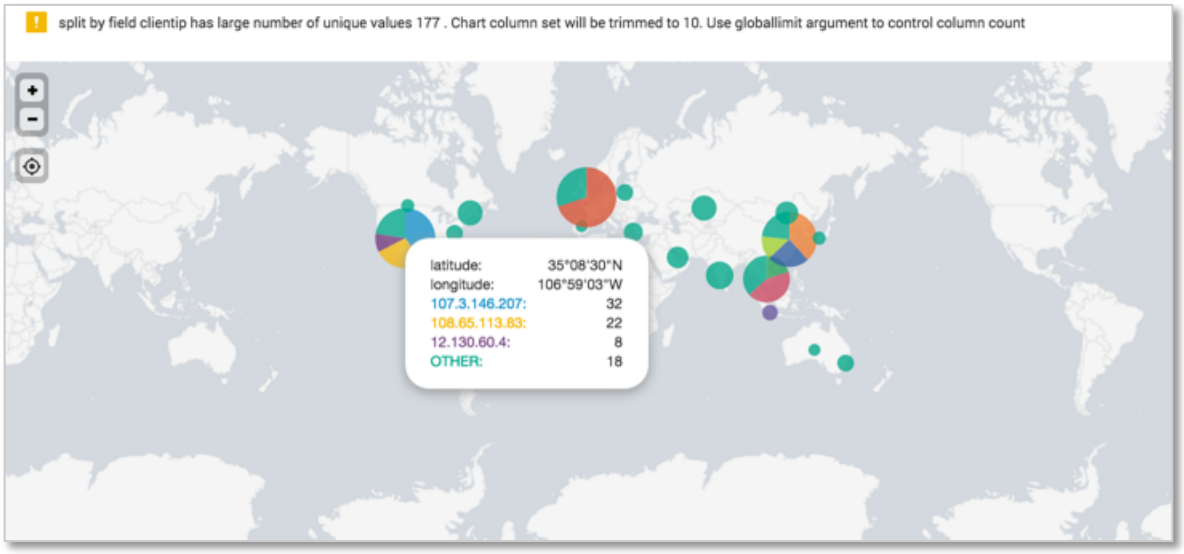
Results Example:

i	Time	Event
>	2/13/16 11:57:45.000 PM	71.192.86.205 - - [14/Feb/2016:04:57:45] "POST /cart/success.do?JSESSIONID=SD9SL6FF3ADFF180963 HTTP 1.1" 200 873 "http://www.buttercupgames.com/cart.do?action=purchase&itemId=EST-11" "Googlebot/2.1 (http://www.googlebot.com/bot.html)" 729 host = www1 source = /opt/log/www1/access.log sourcetype = access_combined
>	2/13/16 11:57:44.000 PM	71.192.86.205 - - [14/Feb/2016:04:57:44] "POST /cart.do?action=purchase&itemId=EST-11&JSESSIONID=SD9SL6FF3ADFF180963 HTTP 1.1" 200 2719 "http://www.buttercupgames.com/cart.do?action=addtocart&itemId=EST-11&categoryId=ARCADE&productId=BS-AG-G09" "Googlebot/2.1 (http://www.googlebot.com/bot.html)" 460 host = www1 source = /opt/log/www1/access.log sourcetype = access_combined

15. Use `iplocation` to extract the location of the purchases based on **clientip**. (You will see the `lat` and `lon` fields on the Fields sidebar.)
sourcetype=access_combined action=purchase status=200
| iplocation clientip

16. To place the events on a map, use `geostats` to count by `clientip`. (Note that you may need to change the visualization to a Cluster Map,  .)
sourcetype=access_combined action=purchase status=200
| iplocation clientip
| geostats count by clientip

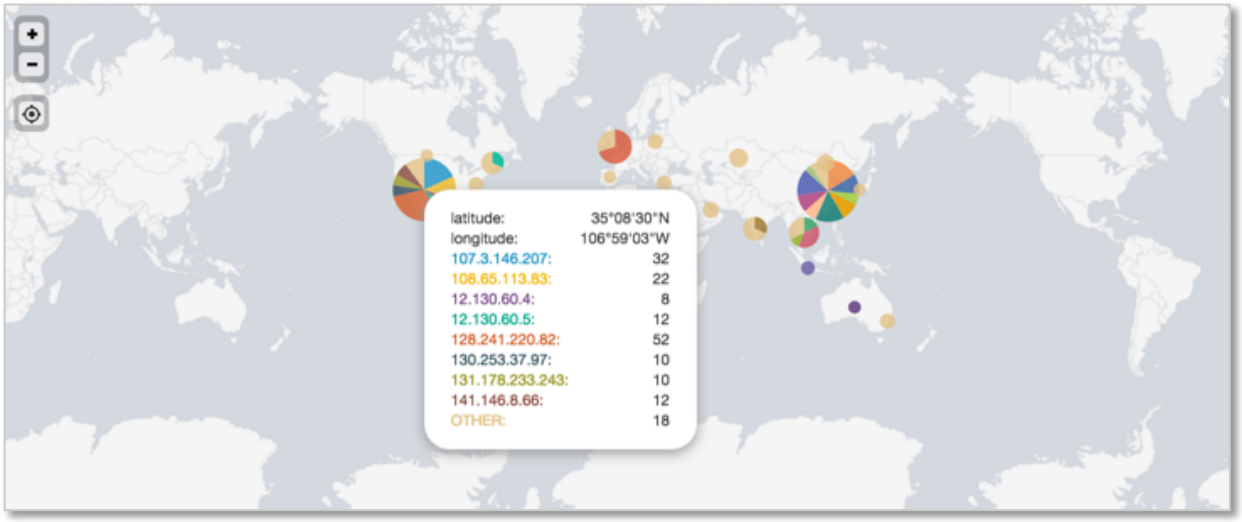
Results Example:



17. Note the message at the top of the screen. The column chart is trimmed to 10 by default. To remove this message, set the `globallimit` to 25

```
sourcetype=access_combined action=purchase status=200
| iplocation clientip
| geostats count by clientip globallimit=25
```

Results Example:



18. Save your search as report, **L4S4**.

Task: **Count the retail sales units sold by country and include a grand total row.**

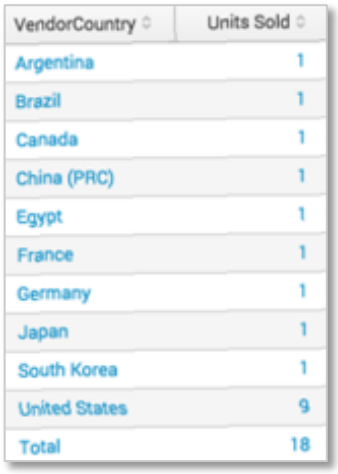
19. Count the number of retail store purchases [`vendor_sales`] as "`Units Sold`" by `VendorCountry`, during the **last 4 hours**.
    sourcetype=vendor_sales
    | stats count as "Units Sold" by VendorCountry

Results Example:

VendorCountry	Units Sold
Argentina	1
Brazil	1
Canada	1
China (PRC)	1

20. Use `addtotals` with the col and row options to display the column total and suppress the row total. Modify the search to include a `Total` label for the last row of the table, as shown in the results below.
    sourcetype=vendor_sales
    | stats count as "Units Sold" by VendorCountry
    | addtotals col=t row=f labelfield="VendorCountry"

Results Example:



VendorCountry	Units Sold
Argentina	1
Brazil	1
Canada	1
China (PRC)	1
Egypt	1
France	1
Germany	1
Japan	1
South Korea	1
United States	9
Total	18

21. Save your search as report, **L4S5**.

Lab Exercise 5 – Manipulating and Filtering Results

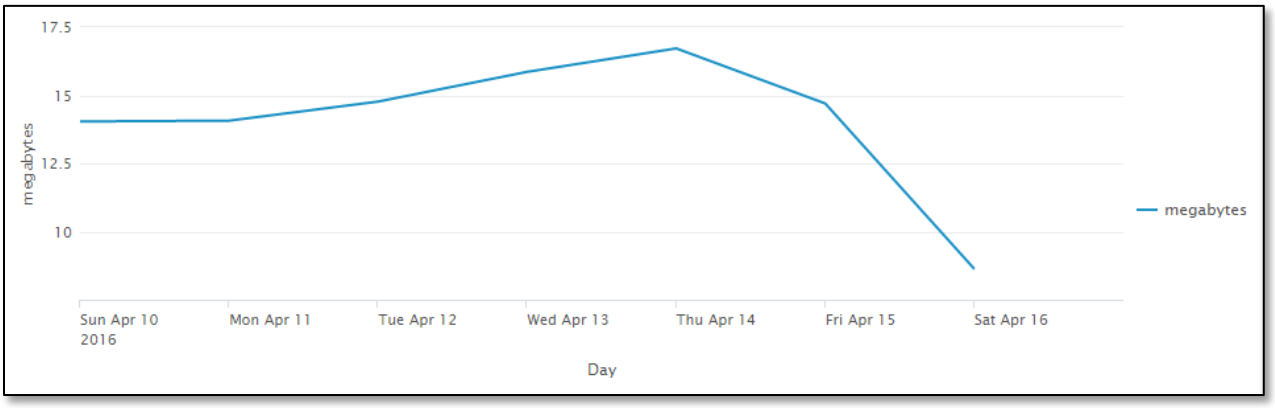
Description

In this lab exercise, you use `eval`, `search`, and `where` commands.

Steps

Task: **Chart the total daily volume (in MB) of the web servers during the previous week.**

Final Results Example:



1. Search online sales [`access_combined`] during the **previous week**.
   sourcetype=access_combined
2. Use `timechart` to calculate the total `bytes` and name the field: `bytes`
   sourcetype=access_combined
   | timechart sum(bytes) as bytes

Results Example:

_time	bytes
2016-02-07	6551357
2016-02-08	6466309
2016-02-09	6750901
2016-02-10	6914829
2016-02-11	8199802
2016-02-12	10830095
2016-02-13	10874717

3. Use `eval` to convert the `bytes` field to `megabytes`.
   ```
   sourcetype=access_combined
   | timechart sum(bytes) as bytes
   | eval megabytes=bytes/(1024*1024)
   ```

Results Example:

_time	bytes	megabytes
2016-02-07	6551357	6.247861
2016-02-08	6466309	6.166753
2016-02-09	6750901	6.438161
2016-02-10	6914829	6.594495
2016-02-11	8199802	7.819941
2016-02-12	10830095	10.328383
2016-02-13	10874717	10.370938

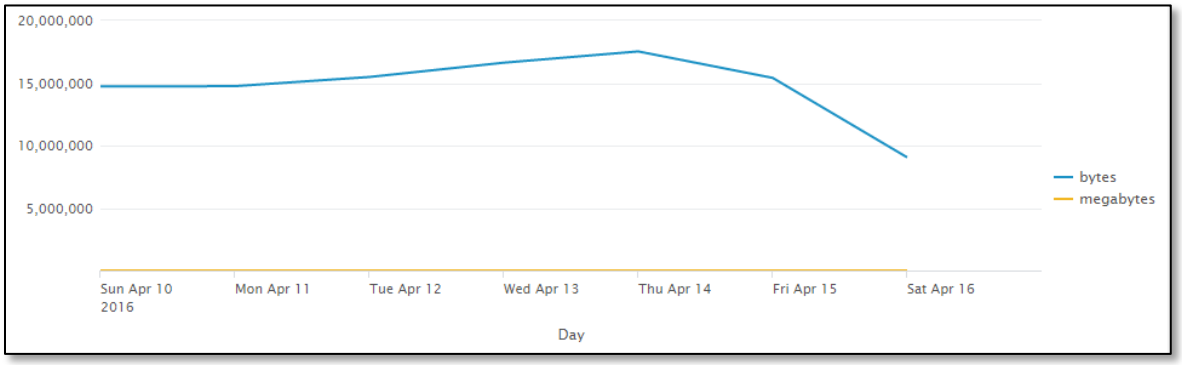
4. Use the `round` function to round the `megabytes` field values to two decimal places.
   ```
   sourcetype="access_combined"
   | timechart sum(bytes) as bytes
   | eval megabytes=round(bytes/(1024*1024),2)
   ```

Results Example:

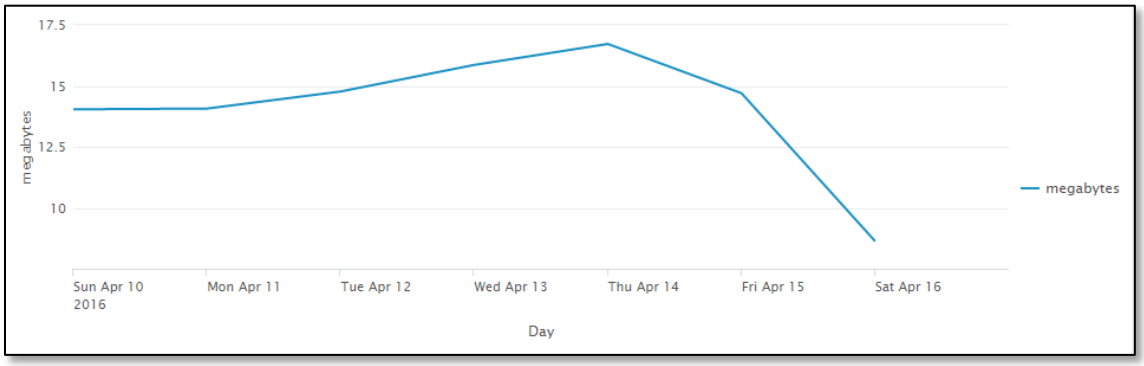
_time	bytes	megabytes
2016-02-07	6551357	6.25
2016-02-08	6466309	6.17
2016-02-09	6750901	6.44
2016-02-10	6914829	6.59
2016-02-11	8199802	7.82
2016-02-12	10830095	10.33
2016-02-13	10874717	10.37

5. Switch to the **Visualization** tab and display the data as a **Line Chart**. Set the X-axis label to **Day**. Notice that the `bytes` field still displays.

Results Example.



6. Use the `fields` command to remove the `bytes` field.
```
sourcetype=access_combined
| timechart sum(bytes) as bytes
| eval megabytes=round(bytes/(1024*1024),2)
| fields - bytes
```
Results Example.



7. Save your search as report, **L5S1**.

Task: **Calculate the ratio of GET requests to POST requests for each web server.**

Final Results Example:

host	GET	POST	Ratio
www1	751	446	1.68
www2	885	569	1.56
www3	592	389	1.52

8. Search online sales [`access_combined`] during the **last 24 hours**.
```
sourcetype=access_combined
```
9. Use `chart` to count events over `host` by `method`.
```
sourcetype=access_combined
| chart count over host by method
```

Results Example:

host	GET	POST
www1	748	444
www2	885	569
www3	592	389

10. Use `eval` to create a new column called `Ratio`, which divides `GET` by `POST`.
```
sourcetype=access_combined
| chart count over host by method
| eval Ratio=GET/POST
```

Results Example:

host	GET	POST	Ratio
www1	749	444	1.686937
www2	885	569	1.555360
www3	592	389	1.521851

11. Round the `Ratio` field to two decimal places.
```
sourcetype=access_combined
| chart count over host by method
| eval Ratio=round(GET/POST,2)
```

Results Example:

host	GET	POST	Ratio
www1	751	446	1.68
www2	885	569	1.56
www3	592	389	1.52

12. Save your search as report, **L5S2**.

Task: **Identify users with more than 3 failed logins during the last 4 hours, sort in descending order.**

Final Results Example:

user	count
root	3933
lsagers	177
showser	163
rerde	96
pdabbeville	84
yowen	82
acurry	80

13. Search the web server [linux_secure] for failures during the last 4 hours.
    sourcetype=linux_secure fail*

    Results Example:

i	Time	Event
>	2/19/16 9:33:43.000 AM	Feb 19 14:33:43 bcg-payroll sshd[5854]: Failed password for acurry from 3.0.0.44 port 52598 ssh2 host = www1 source = /opt/log/www1/auth.nix sourcetype = linux_secure
>	2/19/16 9:33:08.000 AM	Feb 19 14:33:08 bcg-payroll sshd[21085]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=175.45.176.223 user=root host = www1 source = /opt/log/www1/auth.nix sourcetype = linux_secure

14. Use `stats` to count the number of failures by user.
    sourcetype=linux_secure fail*
    | stats count by user

Results Example:

user	count
Alli	1
a	1
aaa	1
acurry	80

15. Using the `search` command, filter the results to include only users with more than three failures and sort in descending order.
    sourcetype=linux_secure fail*
    | stats count by user
    | search count>3
    | sort -count

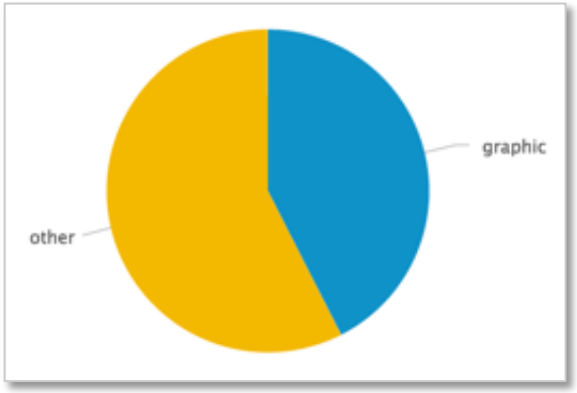
Results Example:

user	count
root	3933
lsagers	177
showser	163
rerde	96
pdabbeville	84
yowen	82
acurry	80

16. Save your search as report, **L5S3**.

****CHALLENGE Exercise: Classify and report employee web traffic by content type during the previous business week.**

Final Results Example:



17. Search web appliance data [`cisco_wsa_squid`] during the **previous business week**.
    sourcetype=cisco_wsa_squid
18. Use `stats` or `chart` to count events by the `http_content_type` field.
    sourcetype=cisco_wsa_squid
    | stats count by http_content_type

Results Example:

http_content_type	count
-	188
application/javascript	16
application/octet-stream	4
application/x-elc	1
application/x-javascript	29
application/x-shockwave-flash	6
image/gif	116
image/jpeg	117

19. Use the `if` function of `eval` to create a new column named `type`. If the `http_content_type` value begins with “image”, set the `type` field to “graphic.” Otherwise, set the value to “other”.
    HINT: Use the LIKE operator and the % wildcard to define the expression as follows:
    `http_content_type LIKE "image%"`
    sourcetype=cisco_wsa_squid
    | stats count by http_content_type
    | eval type=if(http_content_type LIKE "image%","graphic","other")

Results Example:

http_content_type	count	type
-	188	other
application/javascript	16	other
application/octet-stream	4	other
application/x-elc	1	other
application/x-javascript	29	other
application/x-shockwave-flash	6	other
image/gif	116	graphic
image/jpeg	117	graphic

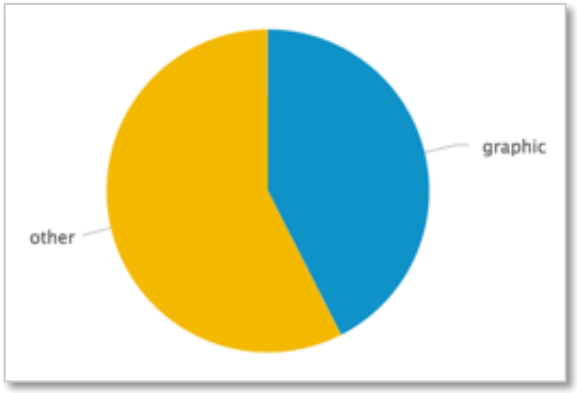
20. Use another `stats` or `chart` command to sum the `count` column by the `type` field. Rename the sum of the `count` calculation to `total`.
```
sourcetype=cisco_wsa_squid
| stats count by http_content_type
| eval type=if(http_content_type LIKE "image%","graphic","other")
| stats sum(count) as total by type
```

Results Example:

type	total
graphic	255
other	345

21. Change the visualization to a **Pie Chart**.

Results Example:



22. Save your search as report, **L5C1**.

****CHALLENGE Exercise: Report which days during the previous week the online store experienced more than three times the number of internal HTTP errors.**

Final Results Example:

_time	access_combined	cisco_wsa_squid
2016-02-07	430	24
2016-02-08	429	12
2016-02-09	434	18
2016-02-10	455	66

23. Search online sales [`access_combined`] and web appliance data [`cisco_wsa_squid`] for HTTP status errors [`status>399`] during the **previous week**.
```
(sourcetype=cisco_wsa_squid OR sourcetype=access_combined) status>399
```

24. Use `timechart` to count events by the `sourcetype` field.
    (sourcetype=cisco_wsa_squid OR sourcetype=access_combined) status>399
    | timechart count by sourcetype

Results Example:

_time	access_combined	cisco_wsa_squid
2016-02-07	430	8
2016-02-08	429	4
2016-02-09	434	6
2016-02-10	455	22

25. Use `eval` to multiply the values in the `cisco_wsa_squid` column by 3.
    (sourcetype=cisco_wsa_squid OR sourcetype=access_combined) status>399
    | timechart count by sourcetype
    | eval cisco_wsa_squid=cisco_wsa_squid*3

Results Example:

_time	access_combined	cisco_wsa_squid
2016-02-07	430	24
2016-02-08	429	12
2016-02-09	434	18
2016-02-10	455	66

26. Use `where` to keep only rows where the value in `access_combined` is greater than the value in `cisco_wsa_squid`.
    NOTE: If no rows are filtered, in the eval clause, multiple cisco_wsa_squid by 25. If that fails to produce results, multiple cisco_wsa_squid by 50. If no results are returned, continue multiplying cisco_wsa_squid by larger integers until results are produced.
    (sourcetype=cisco_wsa_squid OR sourcetype=access_combined) status>399
    | timechart count by sourcetype
    | eval cisco_wsa_squid=cisco_wsa_squid*3
    | where access_combined>cisco_wsa_squid

Results Example:

_time	access_combined	cisco_wsa_squid
2016-02-07	430	24
2016-02-08	429	12
2016-02-09	434	18
2016-02-10	455	66

27. Save your search as report, **L5C2**.
28. Modify your previous search to use `search` instead of `where`. Observe that the search produces no results. Why does this search produce no results?
    No results are found because the `search` command cannot compare values from two different fields. (As you saw earlier, the `where` command can do this.)

Lab Exercise 6 – Correlating Events

Description

Use the transaction command to correlate events.

Steps

Task: Analyze transactions in the online store during the last 60 minutes.

Final Results Example:

JSESSIONID	clientip	action
SD5SL2FF6ADFF4955	86.9.190.90	addtocart purchase
SD6SL2FF8ADFF4963	62.216.64.19	addtocart purchase view
SD0SL3FF3ADFF4950	61.164.73.20	purchase view

1. Search for all events in the online store [`access_combined`] during the **last 60 minutes**.
   sourcetype=access_combined
2. Display a table that shows the `_time`, `clientip`, `JSESSIONID`, and the `action`.
   sourcetype=access_combined
   | table _time, clientip, JSESSIONID, action

Results Example:

_time	clientip	JSESSIONID	action
2016-05-12 15:16:47	148.107.2.20	SD7SL6FF6ADFF4953	view
2016-05-12 15:15:49	76.89.103.115	SD5SL8FF2ADFF4961	
2016-05-12 15:15:39	76.89.103.115	SD5SL8FF2ADFF4961	view
2016-05-12 15:15:32	76.89.103.115	SD5SL8FF2ADFF4961	view
2016-05-12 15:15:15	76.89.103.115	SD5SL8FF2ADFF4961	

Note that the actions are listed from most recent to least recent, i.e. reverse chronological order.

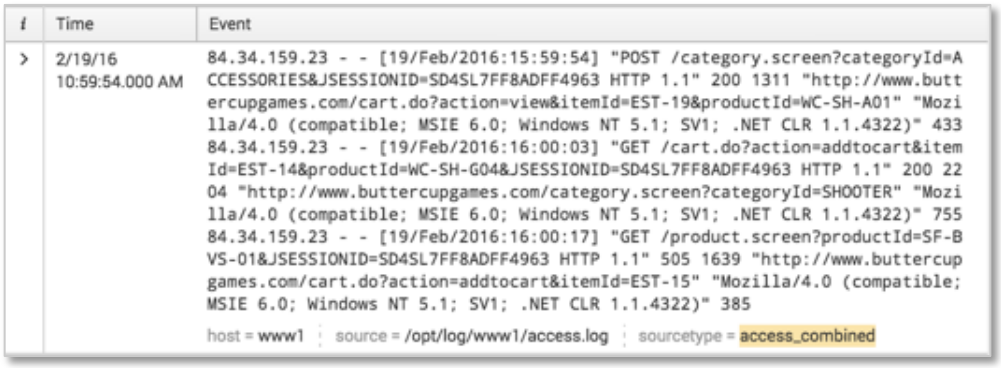
3. Search again for online events with a value in the `action` field. (In the table, you do not need to show the `_time`.)
   sourcetype=access_combined action=*
   | table clientip, JSESSIONID, action

Results Example:

clientip	JSESSIONID	action
84.34.159.23	SD4SL7FF8ADFF4963	addtocart
84.34.159.23	SD4SL7FF8ADFF4963	addtocart
84.34.159.23	SD4SL7FF8ADFF4963	view
62.216.64.19	SD6SL2FF8ADFF4963	view
62.216.64.19	SD6SL2FF8ADFF4963	view

4. Remove the `table` command and, using the `transaction` command, create groups of transactions based on the Java session ID field.
   sourcetype="access_combined" action=*
   | transaction JSESSIONID

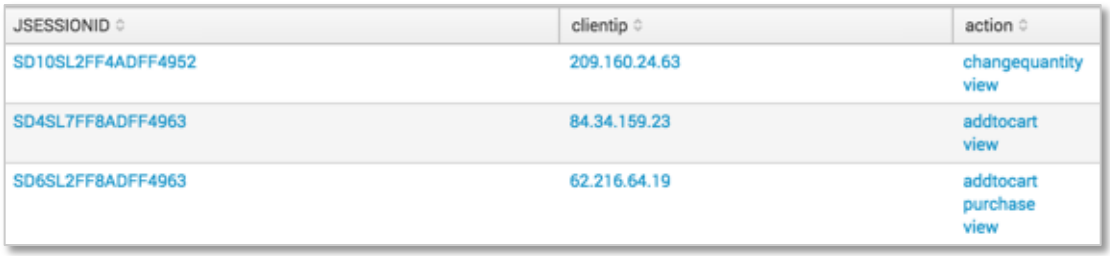
Results Example:



i	Time	Event
>	2/19/16 10:59:54.000 AM	84.34.159.23 - - [19/Feb/2016:15:59:54] "POST /category.screen?categoryId=ACCESSORIES&JSESSIONID=SD4SL7FF8ADFF4963 HTTP 1.1" 200 1311 "http://www.buttercupgames.com/cart.do?action=view&itemId=EST-19&productId=WC-SH-A01" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1; .NET CLR 1.1.4322)" 433 84.34.159.23 - - [19/Feb/2016:16:00:03] "GET /cart.do?action=addtocart&itemId=EST-14&productId=WC-SH-G04&JSESSIONID=SD4SL7FF8ADFF4963 HTTP 1.1" 200 2204 "http://www.buttercupgames.com/category.screen?categoryId=SHOOTER" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1; .NET CLR 1.1.4322)" 755 84.34.159.23 - - [19/Feb/2016:16:00:17] "GET /product.screen?productId=SF-BVS-01&JSESSIONID=SD4SL7FF8ADFF4963 HTTP 1.1" 505 1639 "http://www.buttercupgames.com/cart.do?action=addtocart&itemId=EST-15" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1; .NET CLR 1.1.4322)" 385 host = www1 source = /opt/log/www1/access.log sourcetype = access_combined

5. Modify your search to display the transactions in a table. Include `JSESSIONID`, `clientip`, and `action`.
   ```
   sourcetype=access_combined action=*
   | transaction JSESSIONID
   | table JSESSIONID, clientip, action
   ```

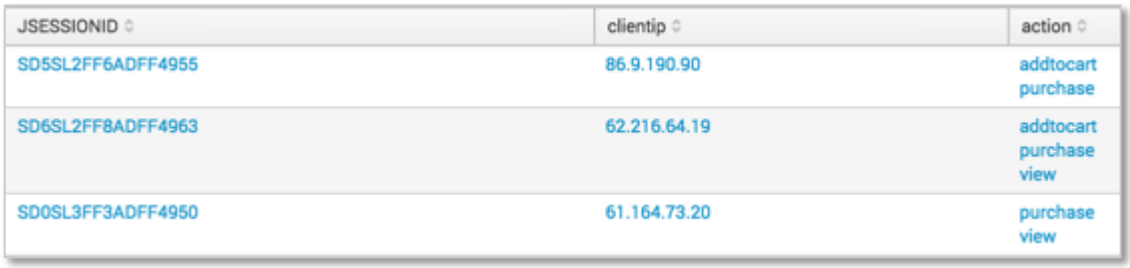
Results Example:



JSESSIONID	clientip	action
SD10SL2FF4ADFF4952	209.160.24.63	changequantity view
SD4SL7FF8ADFF4963	84.34.159.23	addtocart view
SD6SL2FF8ADFF4963	62.216.64.19	addtocart purchase view

6. View only transactions that contain at least one purchase event. Use the `search` command to find transactions containing a purchase.
   NOTE: The `search` command must be downstream from the `transaction` command.
   ```
   sourcetype="access_combined" action=*
   | transaction JSESSIONID
   | table JSESSIONID, clientip, action
   | search action=purchase
   ```

Results Example:



JSESSIONID	clientip	action
SD5SL2FF6ADFF4955	86.9.190.90	addtocart purchase
SD6SL2FF8ADFF4963	62.216.64.19	addtocart purchase view
SD0SL3FF3ADFF4950	61.164.73.20	purchase view

7. Save your search as report, **L6S1**. Click **View.**

Task: **Display the online store purchase transactions lasting more than one minute and include the number of events in each transaction.**

Final Results Example:

clientip	JSESSIONID	action	eventcount	durationMinutes
79.189.193.150	SD1SL7FF2ADFF4950	addtocart purchase view	4	1.7
206.225.11.127	SD10SL10FF10ADFF4955	addtocart purchase remove view	7	1.3

8. If not already displayed, run your **L6S1** search again.
9. In the report, select **Edit > Open in Search**.
10. Set the search mode to **Verbose Mode**, which will re-execute your search. Notice the new fields generated by the `transaction` command: `duration` and `eventcount`
11. Modify your search to add the `duration` and `eventcount` fields to your table and run your search in **Smart Mode**.
    ```
    sourcetype="access_combined" action=*
    | transaction JSESSIONID
    | table JSESSIONID, clientip, duration, eventcount, action
    | search action=purchase
    ```

Results Example:

JSESSIONID	clientip	duration	eventcount	action
SD1SL7FF2ADFF4950	79.189.193.150	102	4	addtocart purchase view
SD9SL5FF4ADFF4954	87.194.216.51	23	3	addtocart purchase remove
SD2SL8FF8ADFF4964	206.225.11.127	58	8	addtocart purchase remove view

12. Use `eval` to create a new field named `durationMinutes`, which is the rounded value of `duration` divided by 60. Round to one decimal place.
    ```
    sourcetype=access_combined action=*
    | transaction JSESSIONID
    | table JSESSIONID, clientip, duration, eventcount, action
    | search action=purchase
    | eval durationMinutes=round(duration/60,1)
    ```

Results Example:

JSESSIONID	clientip	duration	eventcount	action	durationMinutes
SD9SL5FF4ADFF4954	87.194.216.51	23	3	addtocart purchase remove	0.4
SD2SL8FF8ADFF4964	206.225.11.127	58	8	addtocart purchase remove view	1.0
SD10SL10FF10ADFF495!	206.225.11.127	75	7	addtocart purchase remove view	1.3

13. Modify your search to display the columns `clientip`, `JSESSIONID`, `action`, `eventcount`, and `durationMinutes`. To limit the results, search for rows with `durationMinutes` greater than one minute.
```
sourcetype=access_combined action=*
| transaction JSESSIONID
| search action=purchase
| eval durationMinutes=round(duration/60,1)
| table JSESSIONID, clientip, action, durationMinutes, eventcount
| where durationMinutes > 1
```

Results Example:

JSESSIONID	clientip	action	durationMinutes	eventcount
SD1SL5FF2ADFF4956	188.143.232.202	addtocart changequantity purchase view	1.1	6
SD7SL1FF2ADFF4954	216.221.226.11	addtocart purchase view	1.7	5

14. Save your search as report, **L6S2**.

Task: **Search for online store transactions that begin with an `addtocart` action and end with a `purchase` action.**

Final Results Example:

clientip	JSESSIONID	product_name	action	duration	eventcount	price
200.6.134.23	SD9SL10FF9ADFF4954	World of Cheese Tee	addtocart purchase	4	2	9.99
200.6.134.23	SD9SL10FF9ADFF4954	Benign Space Debris	addtocart purchase	3	2	24.99
200.6.134.23	SD9SL10FF9ADFF4954	Final Sequel	addtocart purchase	4	2	24.99

15. Search for all events from the online store [`access_combined`] in the **last 60 minutes** and correlate the events based on `clientip`.
```
sourcetype=access_combined
| transaction clientip
```
16. Use the `startswith` and `endswith` options of the `transaction` command to display transactions that begin with an `addtocart` action and end with a `purchase` action.
```
sourcetype=access_combined
| transaction clientip startswith=action="addtocart" endswith=action="purchase"
```

17. In a table, display clientip, JSESSIONID, product_name, action, duration, eventcount, and price.
```
sourcetype=access_combined
| transaction clientip startswith=action="addtocart" endswith=action="purchase"
| table clientip, JSESSIONID, product_name, action, duration, eventcount, price
```

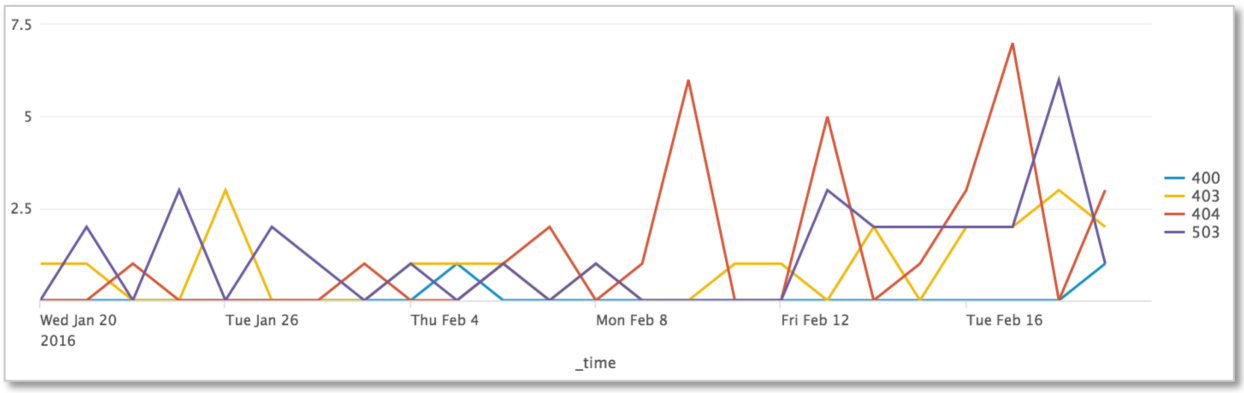
Results Example:

clientip	JSESSIONID	product_name	action	duration	eventcount	price
200.6.134.23	SD9SL10FF9ADFF4954	World of Cheese Tee	addtocart purchase	4	2	9.99
200.6.134.23	SD9SL10FF9ADFF4954	Benign Space Debris	addtocart purchase	3	2	24.99
200.6.134.23	SD9SL10FF9ADFF4954	Final Sequel	addtocart purchase	4	2	24.99

18. Save your search as report, **L6S3**.
19. Click **View**.

****CHALLENGE Exercise: Report common HTTP status errors that occurred during the last 30 days on the online sales web servers and the internal web appliance within a proximity of 5 minutes or less. Omit days with no common errors.**

Final Results Example:



20. Search HTTP status error events from the online sales web servers [`access_combined`] and the web appliance [`cisco_wsa_squid`] during the **last 30 days**. For best performance, limit extracted fields to only `sourcetype` and `status`.
```
sourcetype=cisco_wsa_squid OR sourcetype=access_combined status>399
| fields sourcetype, status
```
21. Create transactions based on `status` field values and limit the span to 5 minutes.
NOTE: If you do not see results, increase the `maxspan` value.
```
sourcetype=cisco_wsa_squid OR sourcetype=access_combined status>399
| fields sourcetype, status
| transaction status maxspan=5m
```
22. Limit the results to only transactions that contain at least one event from each sourcetype.
```
sourcetype=cisco_wsa_squid OR sourcetype=access_combined status>399
| fields sourcetype, status
| transaction status maxspan=5m
| search sourcetype=access_combined AND sourcetype=cisco_wsa_squid
```

23. Use `timechart` to count events by `status`.

```
sourcetype=cisco_wsa_squid OR sourcetype=access_combined status>399
| fields sourcetype, status
| transaction status maxspan=5m
| search sourcetype=access_combined AND sourcetype=cisco_wsa_squid
| timechart count by status
```

Results Example:

_time	400	403	404	503
2016-01-20	0	1	0	0
2016-01-21	0	0	0	0
2016-01-22	0	1	0	2
2016-01-23	0	0	1	0

24. Discard rows that have no (zero) errors for all `status` values.
HINT: Use `addtotals`.

```
sourcetype=cisco_wsa_squid OR sourcetype=access_combined status>399
| fields sourcetype, status
| transaction status maxspan=5m
| search sourcetype=access_combined AND sourcetype=cisco_wsa_squid
| timechart count by status
| addtotals
| search Total>0
```

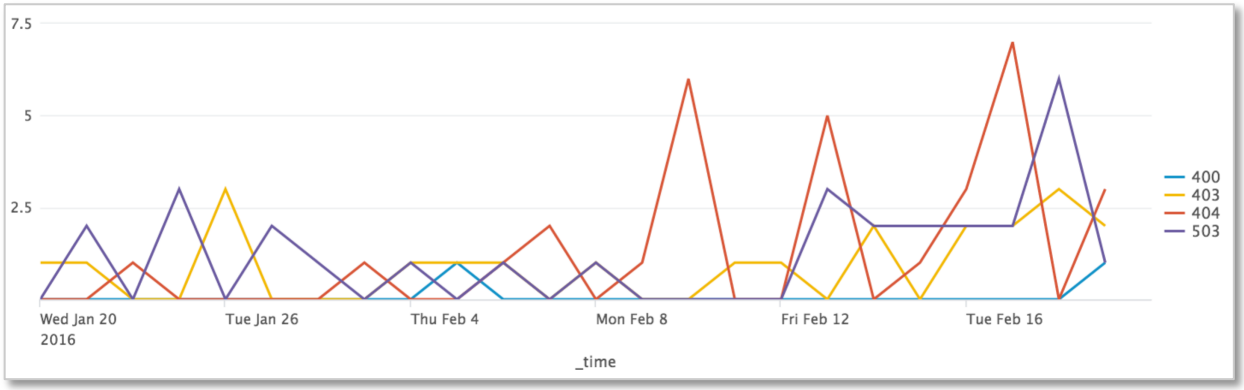
Results Example:

_time	400	403	404	503	Total
2016-01-20	0	1	0	0	1
2016-01-22	0	1	0	2	3
2016-01-23	0	0	1	0	1
2016-01-25	0	0	0	3	3

25. Remove the `Total` column and display the data as a **Line chart**.

```
sourcetype=cisco_wsa_squid OR sourcetype=access_combined status>399
| fields sourcetype, status
| transaction status maxspan=5m
| search sourcetype=access_combined AND sourcetype=cisco_wsa_squid
| timechart count by status
| addtotals
| search Total>0
| fields - Total
```

Results Example:



26. Save your search as report, **L6C1**.
27. Optionally: for this line chart, set **Multi-series Mode** to **Yes**. (Hint: it's one of the **Format** options on the **General** tab.) Observe the change in how the lines are represented.