Statistical Processing

Document Usage Guidelines

- Should be used only for enrolled students
- Not meant to be a self-paced document, an instructor is needed
- Do not distribute

Course Goals

- Define data series
- Transform data with the **`chart`**, **`timechart`**, **`top`**, and **`rare`** commands
- Perform statistical aggregations using the **`stats`** command with aggregate and multivalue functions
- Manipulate data with the **`eval`** command
- Format data with the **`rename`** and **`sort`** commands

Course Outline

- Topic 1: What is a Data Series
- Topic 2: Transforming Data
- Topic 3: Statistical Aggregation with the `stats` Command
- Topic 4: Manipulating Data with `eval` Command
- Topic 5: Formatting Data

Topic 1:
What is a Data Series?

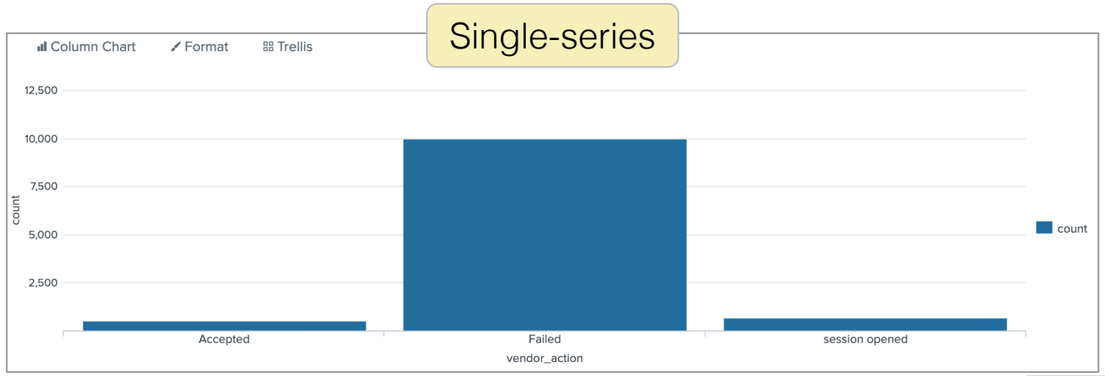
Topic Objectives

- Define data series
- Determine the difference between data series
  - Single-series
  - Multi-series
  - Time series

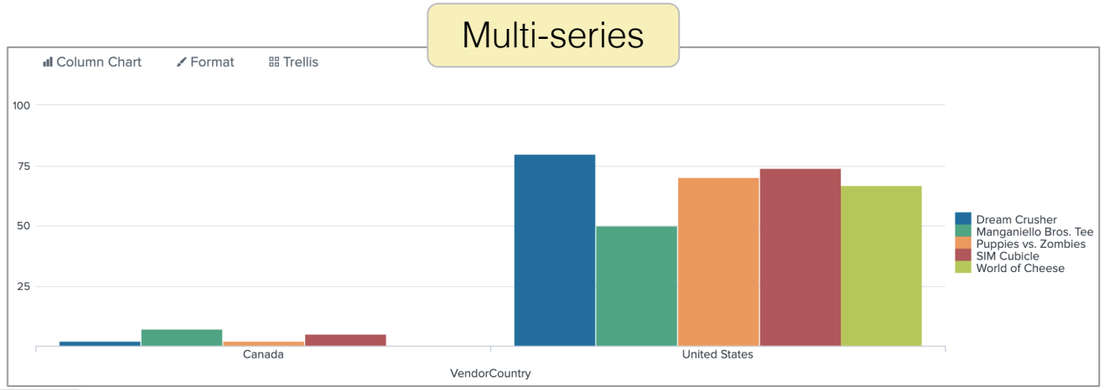
Data Series

- A sequence of related data points that are plotted in a visualization
- Not all results can be visualized as a data series
- Three types are discussed in this course

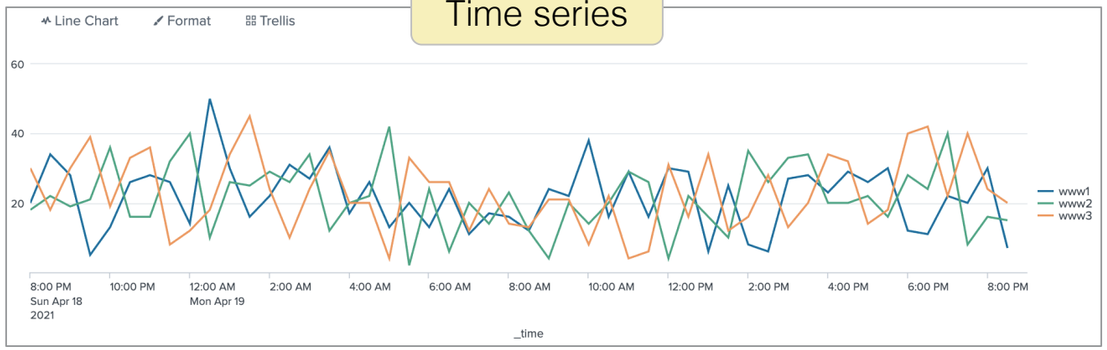
Single-series



Multi-series



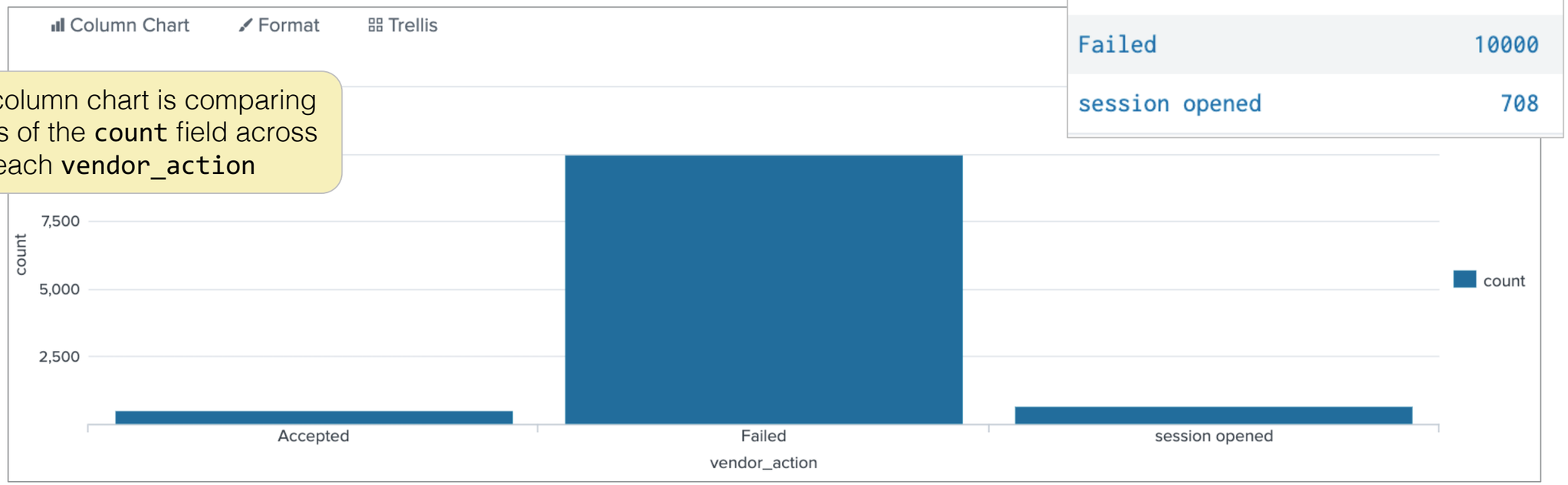
Time series



Data Series: Single-series

A sequence of related data points that compares values of a single data category

vendor_action	count
Accepted	529
Failed	10000
session opened	708



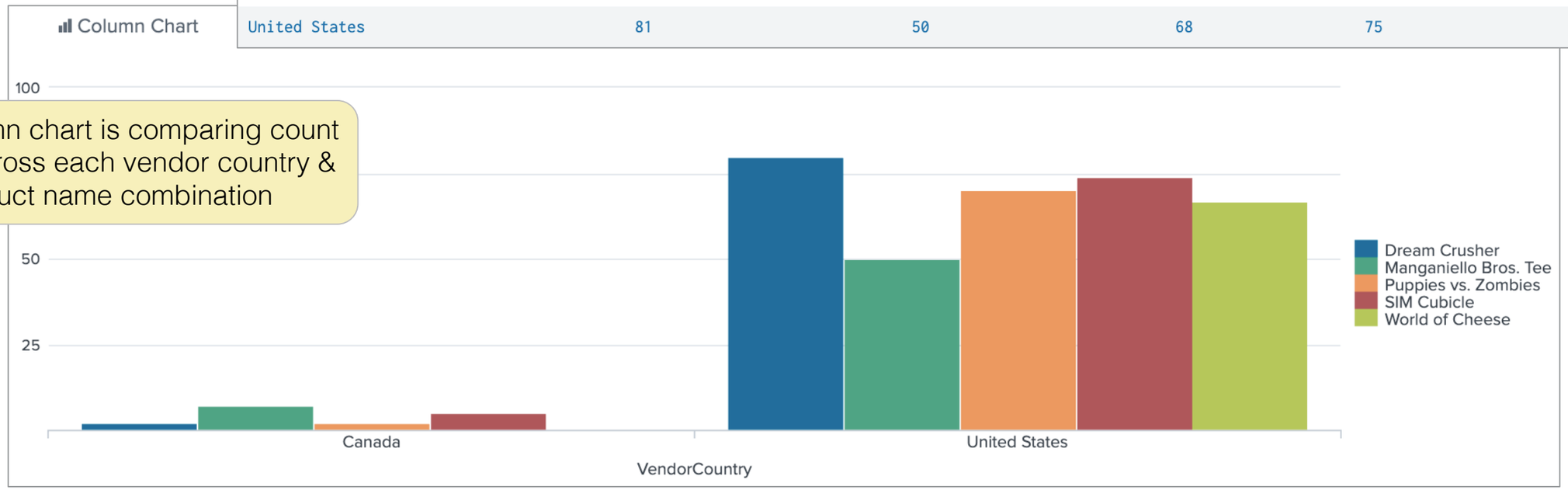
Data Series: Multi-series

A sequence of related data points that compares values of two or more data categories

Each product name represents one data category

VendorCountry	Dream Crusher	Manganiello Bros. Tee	Puppies vs. Zombies	SIM Cubicle	World of Cheese
Canada	2	8	2	4	0
United States	81	50	68	75	64

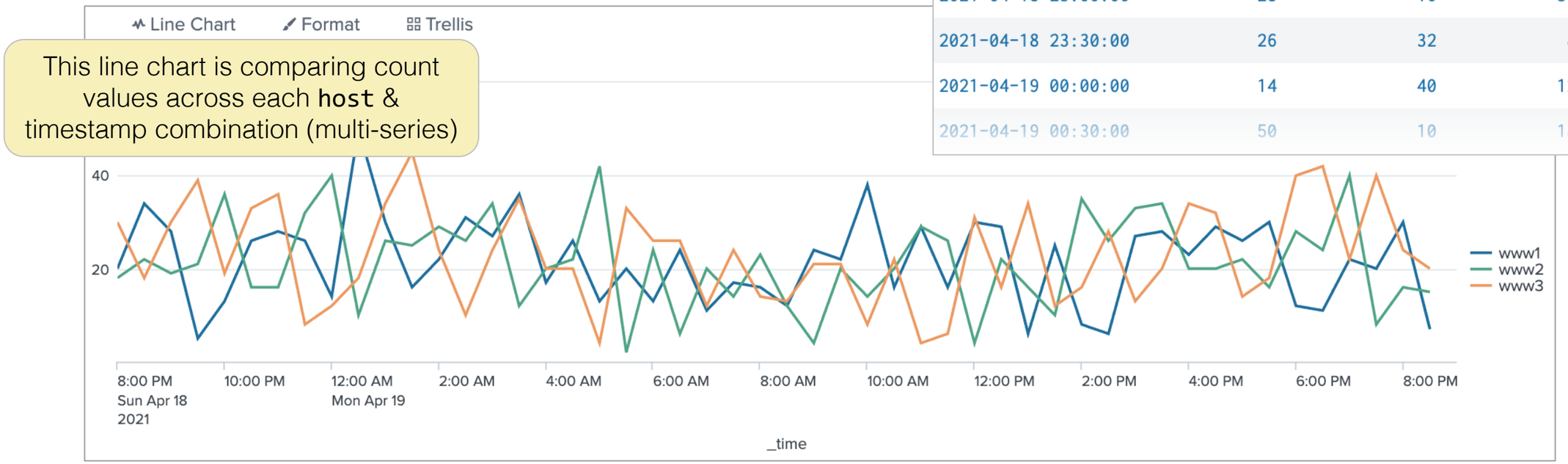
This column chart is comparing count values across each vendor country & product name combination



Data Series: Time Series

- A sequence of related data points that compares values over time
- Can be single-series or multi-series

_time	www1	www2	www3
2021-04-18 22:00:00	13	36	19
2021-04-18 22:30:00	26	16	33
2021-04-18 23:00:00	28	16	36
2021-04-18 23:30:00	26	32	8
2021-04-19 00:00:00	14	40	12
2021-04-19 00:30:00	50	10	18



Transforming Commands for Data Series

- A type of command that organizes results into a data table
- Searches that use transforming commands are called transforming searches
  - Creates statistical tables containing data series that can be visualized
  - Each command generates specific data structures

Note

Different visualizations require data structures to be set up in particular ways. For example, a search that creates a line chart might not be suitable for a pie chart. Data structures and visualizations are outside the scope of this course.

Preview: The Next Topics

Transforming Data	Manipulating Data with eval	Formatting Data

Create data series with transforming commands:
`chart`
`timechart`
`stats`
`rare`
`top`
	Use eval command and evaluation functions to manipulate data series:
	`pow`
	`round`
	`min`
	`max`
	`random`
		Work with formatting commands to organize and refine fields in your data series:
		`rename`
		`sort`

Topic 2: Transforming Data

Topic Objectives

- Transform events into a data table with the following commands:
  - `chart`
  - `timechart`
  - `top`
  - `rare`

`chart` Command

```
...| chart <stats-func>(<wc-field>) over <row-split> [by <column-split>]
     [span=<int><timescale>][limit=<int>] [useother=<bool>] [usenull=<bool>]
```

- Returns results in a table format that can be displayed as a visualization
- Specify the Y-axis with **`<stats-func>(<wc-field>)`**
  - **`<wc-field>`** is a field with numeric values; supports wildcards
  - **`<stats-func>`** is a supported statistical function
- Specify X-axis with **`over <row-split>`**
- Further split data by including **`by <column-split>`**
- Control behavior with **`span`**, **`limit`**, **`useother`**, and **`usenull`**

`chart` Command: `split` Fields

- Use **`over <row-split>`** to specify the X-axis and create a single-series data series
- Further split results and create a multi-series data series by adding **`by <column-split>`**
  - Splits results of **`<stats-func>(<wc-field>)`**
  - May alter the range of the Y-axis
- Results can only be split using two fields
- Alternative syntax: **`by`** clause with two arguments

```
...| chart <stats-func>(<wc-field>) by <row-split> <column-split>
```

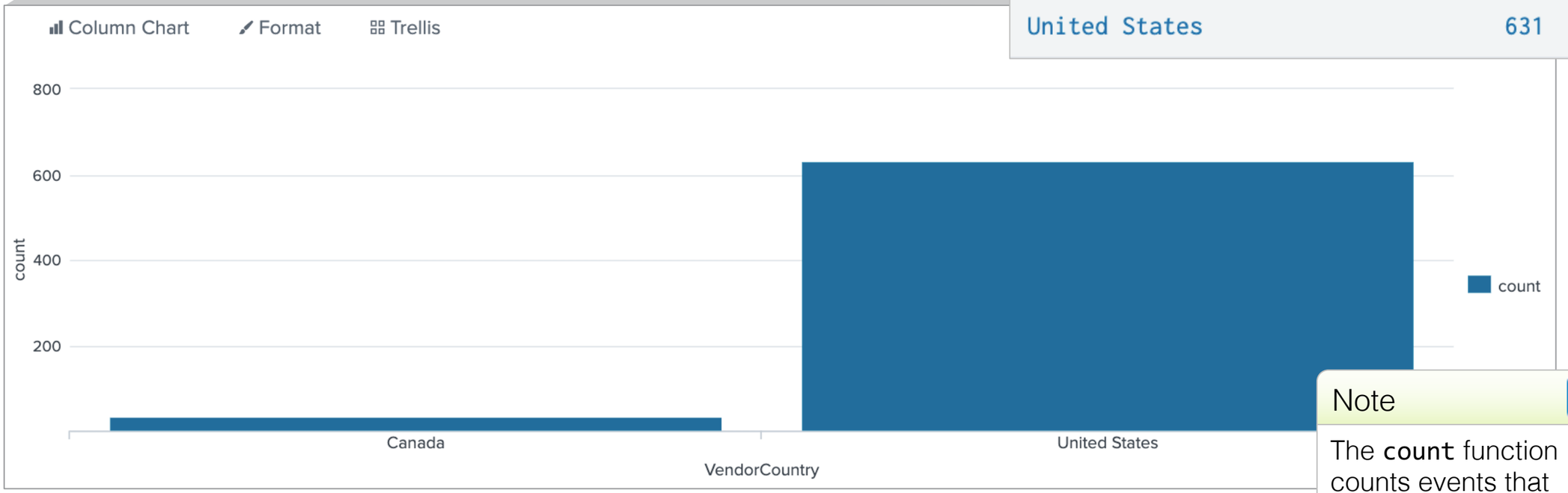
Equivalent to using an **`over`** and **`by`** clause

`chart` Command: `split` Fields Example 1

Produce a single-series data series using just the **`over`** clause

```
index=sales sourcetype=vendor_sales VendorID<4000
| chart count over VendorCountry
```

VendorCountry	count
Canada	33
United States	631



Note

The **`count`** function counts events that match search criteria.

chart Command: split Fields Example 2

Produce a multi-series data series by splitting data over two fields

```
index=sales sourcetype=vendor_sales VendorID<4000
| chart count over VendorCountry by product_name
```

```
index=sales sourcetype=vendor_sales VendorID<4000
| chart count by VendorCountry product_name
```

These searches produce the same results with different syntax

VendorCountry	Curling 2014	Dream Crusher	Final Sequel	Fire Resistance Suit of Provolone	Manganiello Bros. Tee	Orvil the Wolverine	Puppies vs. Zombies	SIM Cubicle	World of Cheese	World of Cheese Tee	OTHER
Canada	1	2	3	2	7	3	2	4	0	1	9
United States	33	79	40	40	51	33	65	74	66	42	108

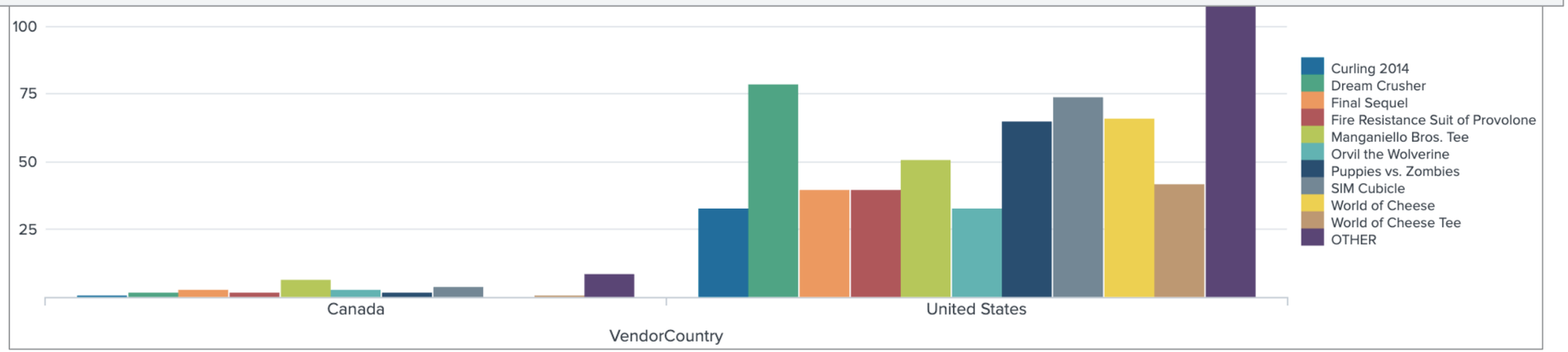
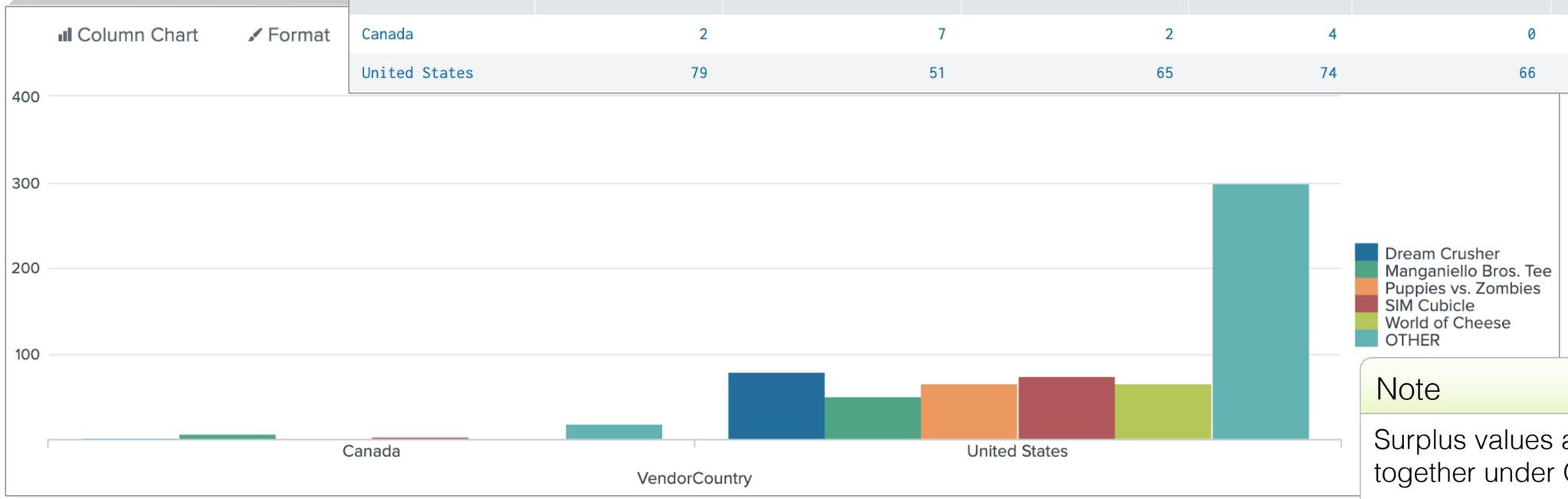


chart Command: limit

Use the **limit** option to control how many series result from the **by <column-split>** clause; defaults to **limit=10**

```
index=sales sourcetype=vendor_sales VendorID<4000
| chart count over VendorCountry by product_name limit=5
```

VendorCountry	Dream Crusher	Manganiello Bros. Tee	Puppies vs. Zombies	SIM Cubicle	World of Cheese	OTHER
Canada	2	7	2	4	0	19
United States	79	51	65	74	66	300



Note

Surplus values are grouped together under OTHER.

`chart` Command: `limit` (cont.)

For unlimited `<column-split>` values, use `limit=0`

```
index=sales sourcetype=vendor_sales VendorID<4000
| chart count over VendorCountry by product_name limit=0
```

VendorCountry	Benign Space Debris	Curling 2014	Dream Crusher	Final Sequel	Fire Resistance Suit of Provolone	Holy Blade of Gouda	Manganiello Bros.	Manga Bros.
Canada	4	1	2	3	2	1	3	
United States	17	34	81	39	38	31	30	

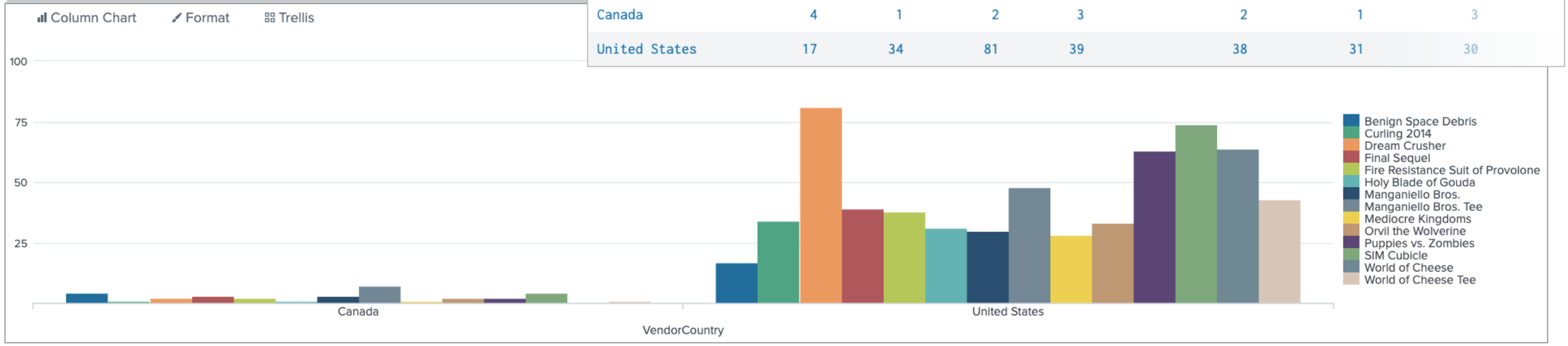


chart Command: usenull and useother

Further control column splitting behavior by specifying if you want null values or other data categories displayed

```
index=sales sourcetype=vendor_sales VendorID<4000
| chart count over VendorCountry by product_name limit=5 useother=f usenull=f
```

VendorCountry	Dream Crusher	Manganiello Bros. Tee	Puppies vs. Zombies	SIM Cubicle	World of Cheese
Canada	2	7	2	4	0
United States	80	51	65	74	66

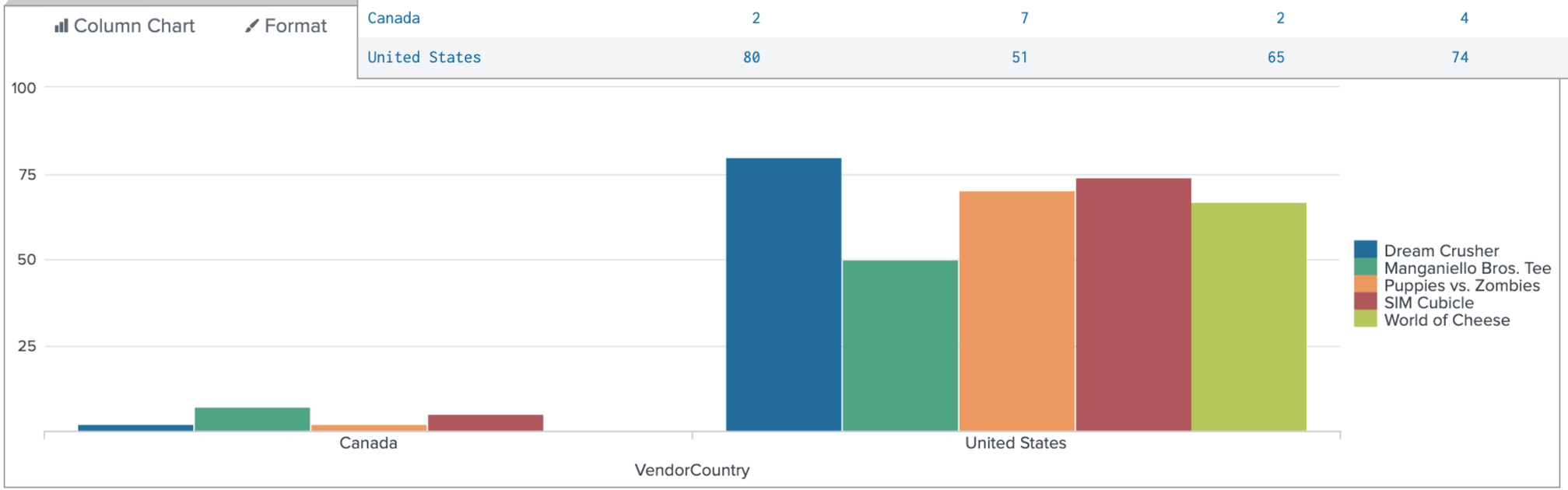


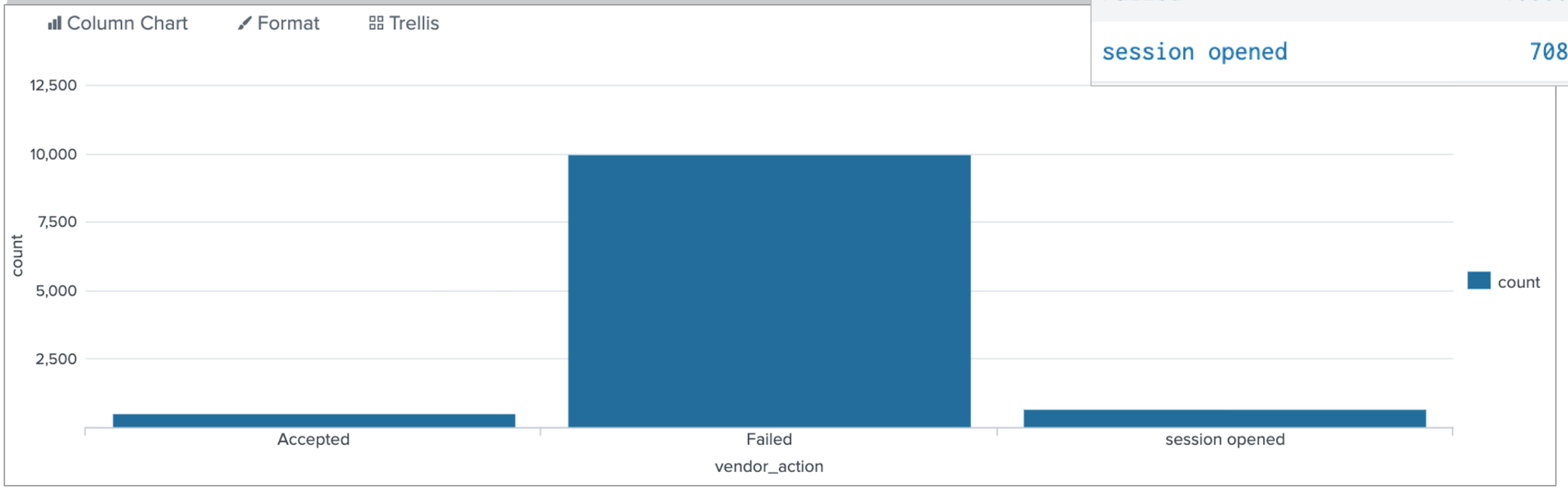
chart Command Example 1

Scenario

Display a count of vendor actions over the last 60 minutes.

```
index=security sourcetype=linux_secure
| chart count by vendor_action
```

vendor_action	count
Accepted	529
Failed	10000
session opened	708



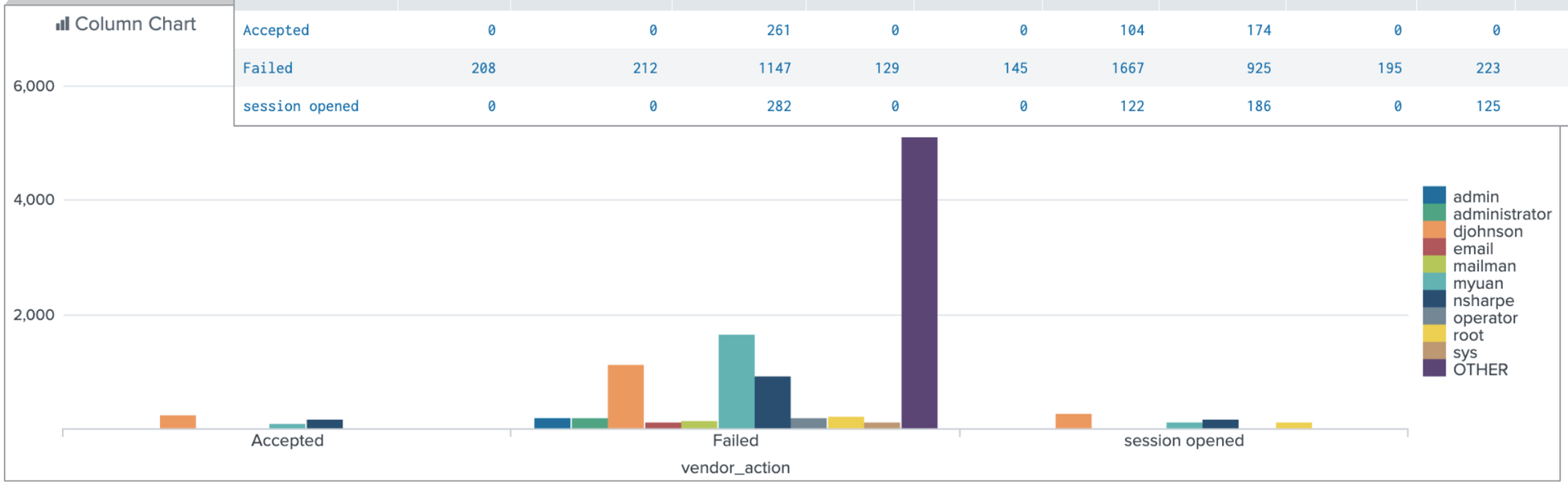
`chart` Command Example 2

Scenario

Display a count of vendor actions by user over the last 60 minutes.

```
index=security sourcetype=linux_secure
| chart count by vendor_action user
```

vendor_action	admin	administrator	djohnson	email	mailman	myuan	nsharpe	operator	root	sys	OTHER
Accepted	0	0	261	0	0	104	174	0	0	0	0
Failed	208	212	1147	129	145	1667	925	195	223	124	5112
session opened	0	0	282	0	0	122	186	0	125	0	0



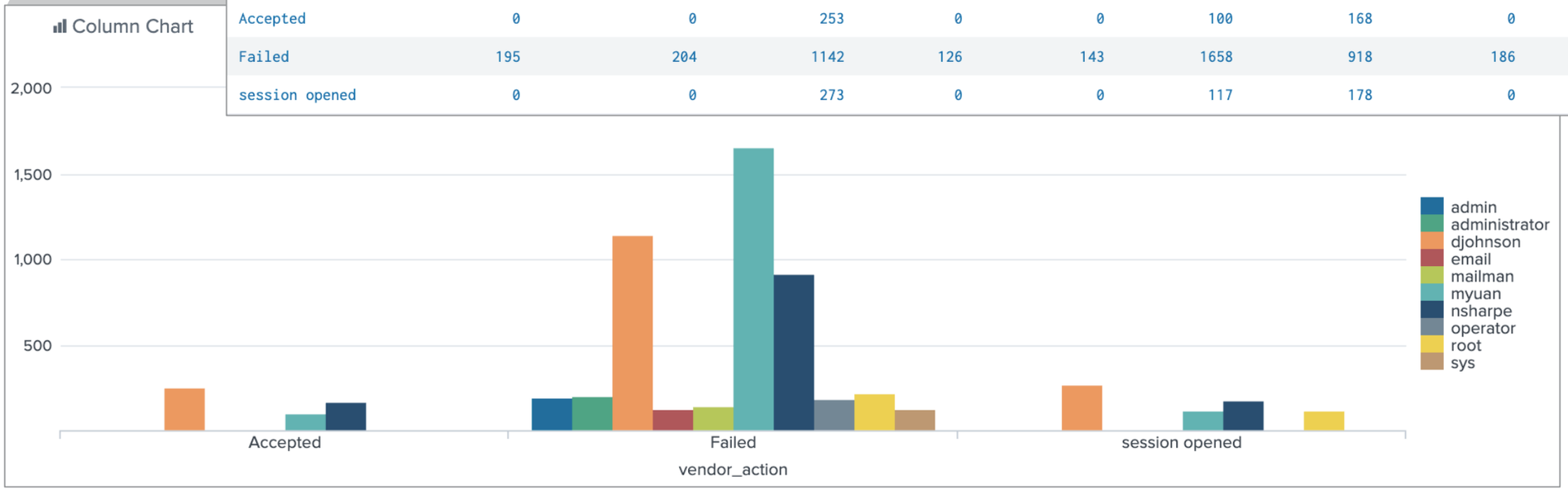
`chart` Command Example 2 (cont.)

Scenario

The last report was skewed by surplus categories. Remove these categories from your report.

```
index=security sourcetype=linux_secure
| chart count by vendor_action user useother=f
```

vendor_action	admin	administrator	djohnson	email	mailman	myuan	nsharpe	operator	root	sys
Accepted	0	0	253	0	0	100	168	0	0	0
Failed	195	204	1142	126	143	1658	918	186	221	123
session opened	0	0	273	0	0	117	178	0	121	0



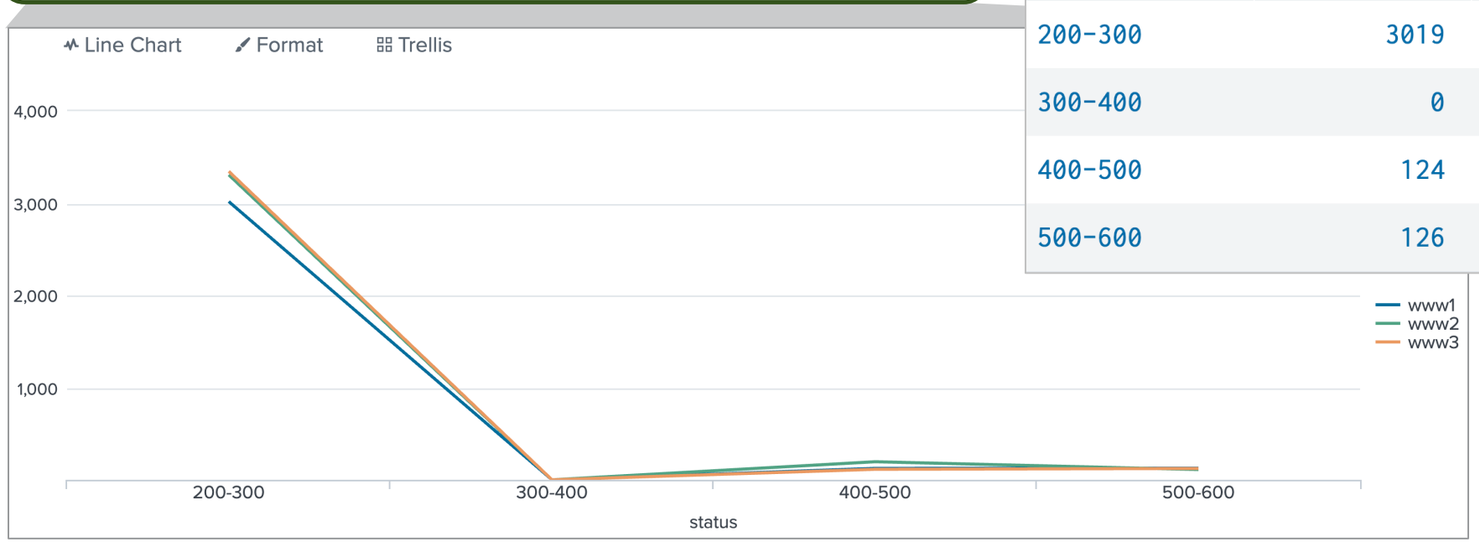
`chart` Command: `span` Option

- If the `<row-split>` field (X-axis) is numeric, use the **`span`** option with the **`chart`** command to group events into buckets
- Splunk shifts overlapped values to the higher grouping

Scenario

The IT manager wants to see the volume of web activity over the past 24 hours by ranges of status codes.

```
index=web sourcetype=access_combined
| chart span=100 count over status by host
```



status	www1	www2	www3
200-300	3019	3309	3347
300-400	0	0	0
400-500	124	196	114
500-600	126	111	123

`timechart` Command

```
...| timechart <stats-func>(<field>) by <split-by-field>
     [span=<int><timescale>] [limit=<int>]
```

- Performs statistical aggregations against time and returns a time series chart where **`_time`** is always the X-axis
- **`<stats-func>(<field>)`** applies a function to a single field and populates the Y-axis
  - If using the **`count`** function, a field does not need to be specified

Note

Functions and arguments used with **`chart`** can also be used with **`timechart`**. However, unlike **`chart`**, **`timechart`** does not support wildcarded fields.

`timechart` Command (cont.)

```
...| timechart <stats-func>(<field>) [by <split-by-field>]
     [span=<int><timescale>] [limit=<int>]
```

- Use by **`<split-by-field>`** to further split results of the statistical aggregation
  - Only one **`<split-by-field>`** can be specified
  - Each distinct value of **`<split-by-field>`** becomes a series
- Further control timechart behavior with **`span`** and **`limit`** options

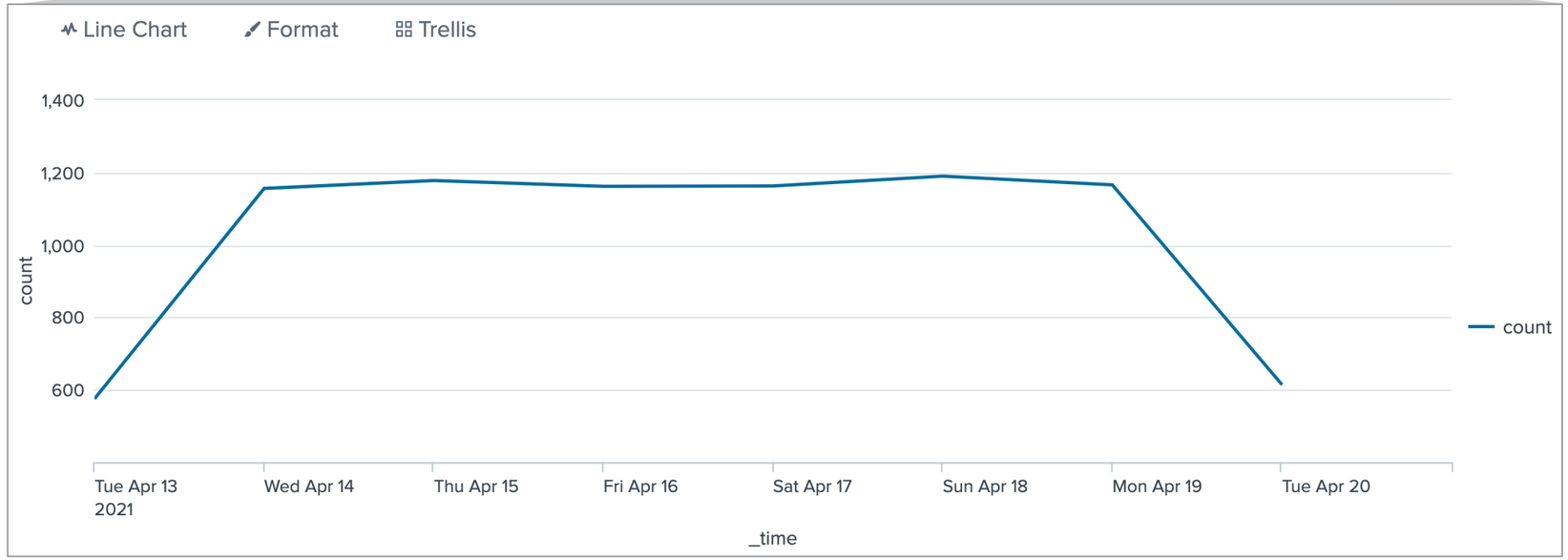
`timechart` Command Example

Use `timechart` with just `<stats-func>(<field>)` to produce a single-series time chart

Scenario

How many events have been logged on the cisco web security appliance during the last 7 days?

```
index=network sourcetype=cisco_wsa_squid
| timechart count
```



Note

`_time` will ALWAYS populate the X-axis when using `timechart`.

`timechart` Command Example (cont.)

Including a `by <split-by-field>` clause splits the result of the statistical aggregation and creates a multi-series time chart

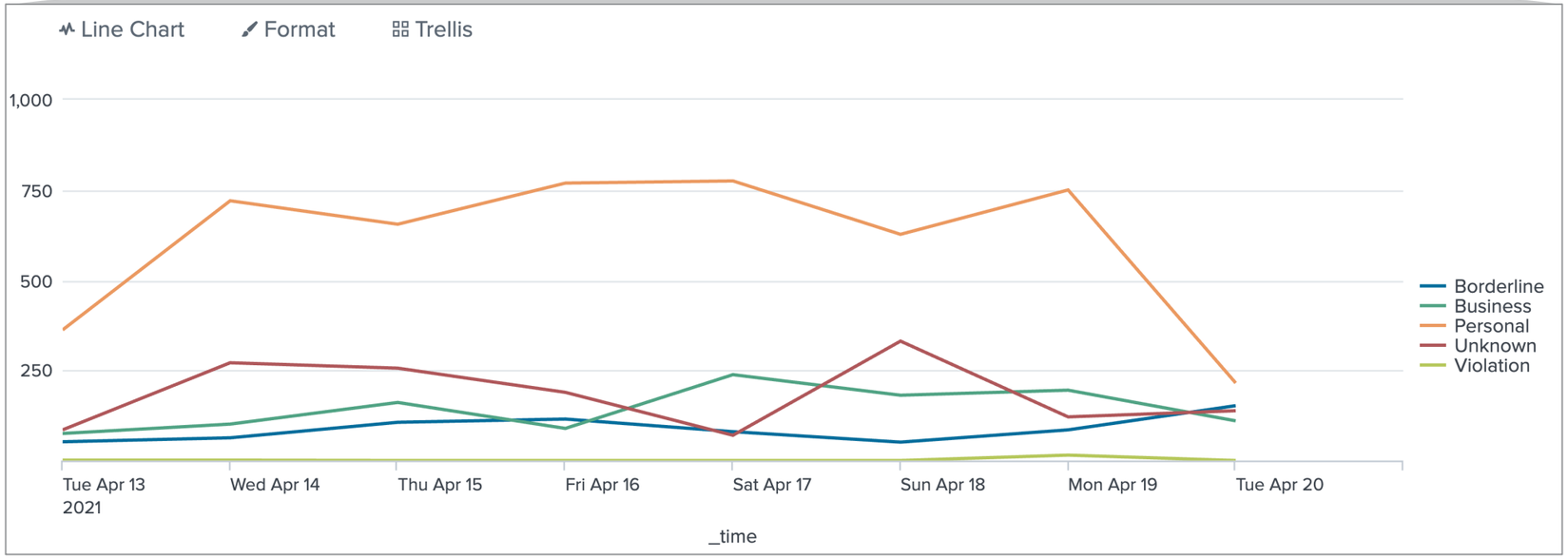
Scenario

SecOps wants to see the previous report split by usage type.

Note

You can only split by one field with `timechart`.

```
index=network sourcetype=cisco_wsa_squid
| timechart count by usage
```



`timechart` Command: `span` Option

- The **`timechart`** command "buckets" the values of the **`_time`** field based on time range if no **`span`** argument is specified
- Examples:
  - A **Last 60 minutes** defaults to **`span=1m`**
  - B **Last 24 hours** defaults to **`span=30m`**

```
index=security sourcetype=linux_secure vendor_action=*
| timechart count by vendor_action
```

Last 60 minutes

_time	Accepted	Failed	session opened
2021-04-15 01:43:00	0	14	0
2021-04-15 01:44:00	0	0	1
2021-04-15 01:45:00	0	0	1
2021-04-15 01:46:00	1	11	0

Last 24 hours

_time	Accepted	Failed	session opened
2021-04-14 02:00:00	13	133	15
2021-04-14 02:30:00	10	250	13
2021-04-14 03:00:00	10	154	16
2021-04-14 03:30:00	8	227	22

`timechart` Command: `span` Option (cont.)

Manually adjust the interval using the **`span`** option

```
index=security sourcetype=linux_secure vendor_action=*
| timechart span=15m count by vendor_action
```

_time	Accepted	Failed	session opened
2021-04-15 01:45:00	3	80	1
2021-04-15 02:00:00	5	113	7
2021-04-15 02:15:00	2	62	8
2021-04-15 02:30:00	3	67	14
2021-04-15 02:45:00	2	173	10

`timechart` Command: `limit` Option

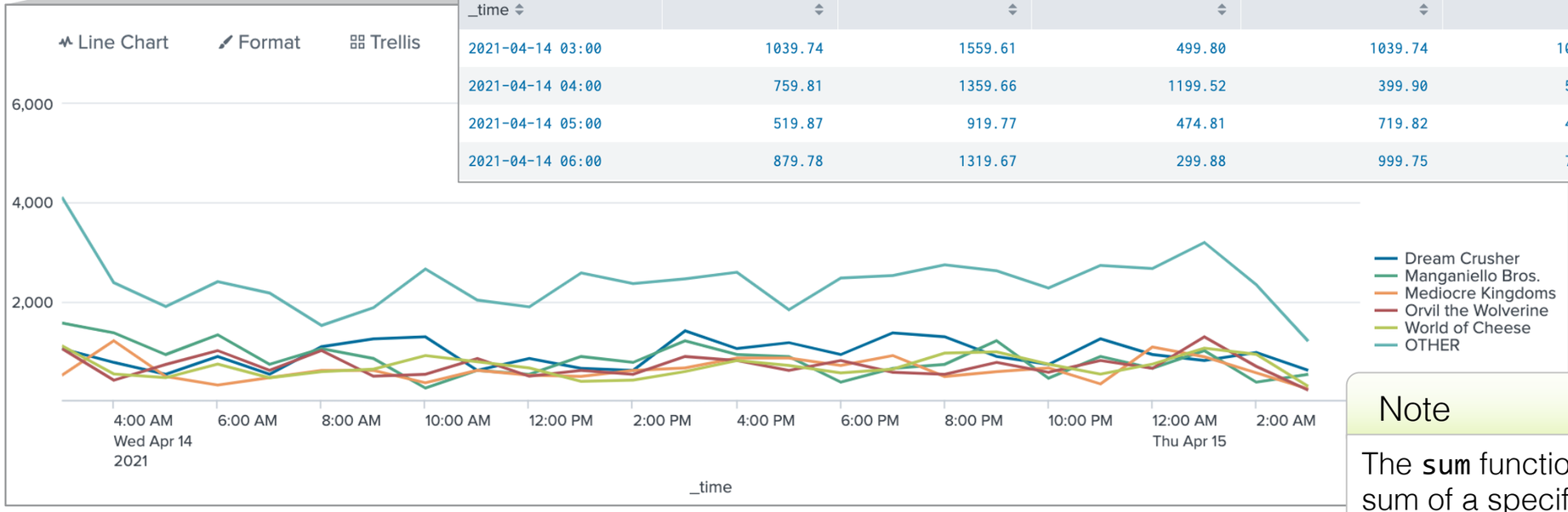
The **`limit`** option controls the number of distinct values returned by the **`by`** clause field

Scenario

What was the total web sales for the top products during the last 24 hours?

```
index=web sourcetype=access_combined
| timechart span=1h sum(price) by product_name limit=5
```

_time	Dream Crusher	Manganiello Bros.	Mediocre Kingdoms	Orvil the Wolverine	World of Cheese	OTHER
2021-04-14 03:00	1039.74	1559.61	499.80	1039.74	1099.56	4111.23
2021-04-14 04:00	759.81	1359.66	1199.52	399.90	524.79	2376.17
2021-04-14 05:00	519.87	919.77	474.81	719.82	449.82	1894.52
2021-04-14 06:00	879.78	1319.67	299.88	999.75	724.71	2398.30



Note

The **`sum`** function returns the sum of a specified field.

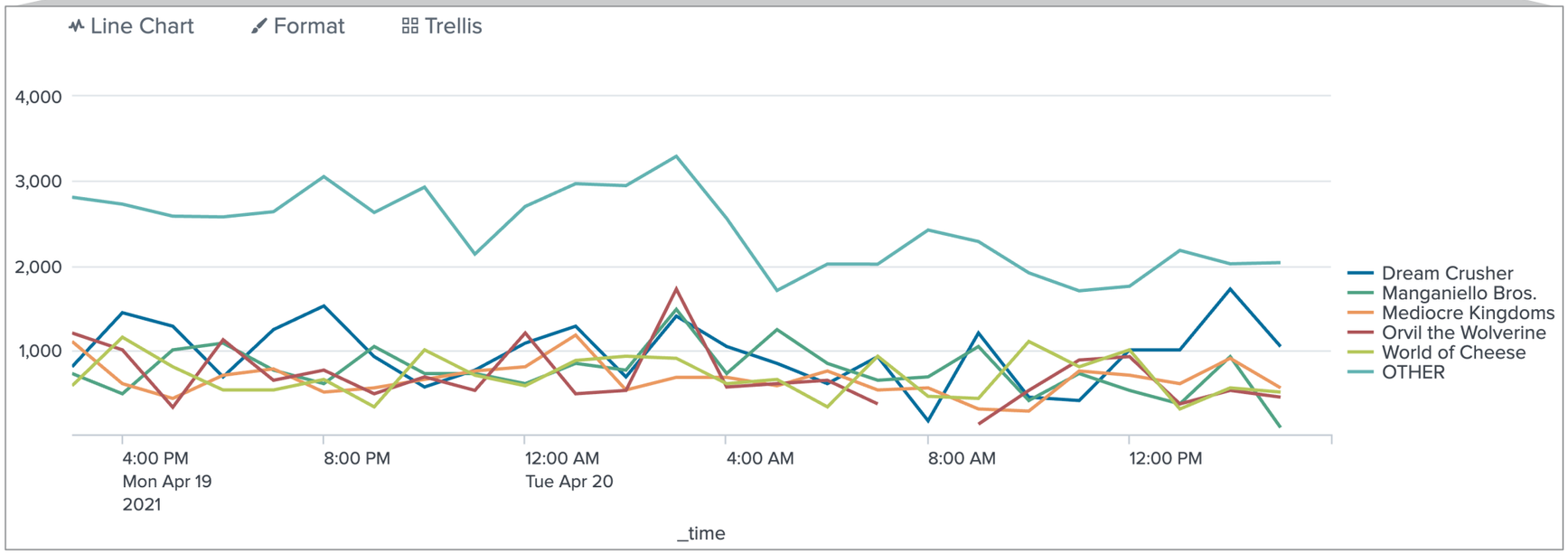
`timechart` Command Example: `<stats-func>`

`chart` and `timechart` can use the same statistical functions

Scenario

What was the total web sales for our top products during the last 24 hours?

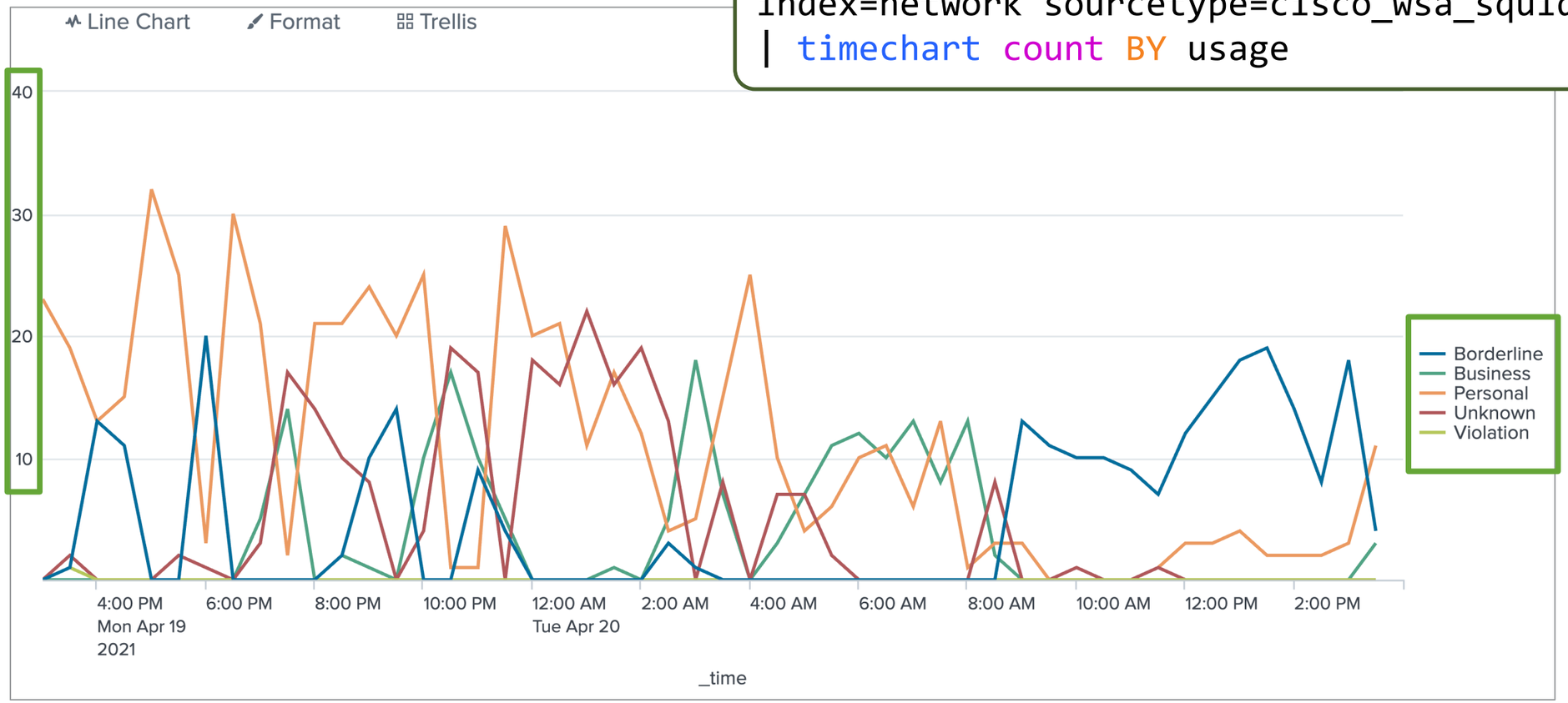
```
index=web sourcetype=access_combined
| timechart span=1h sum(price) by product_name limit=5
```



Toggle Multi-series Mode

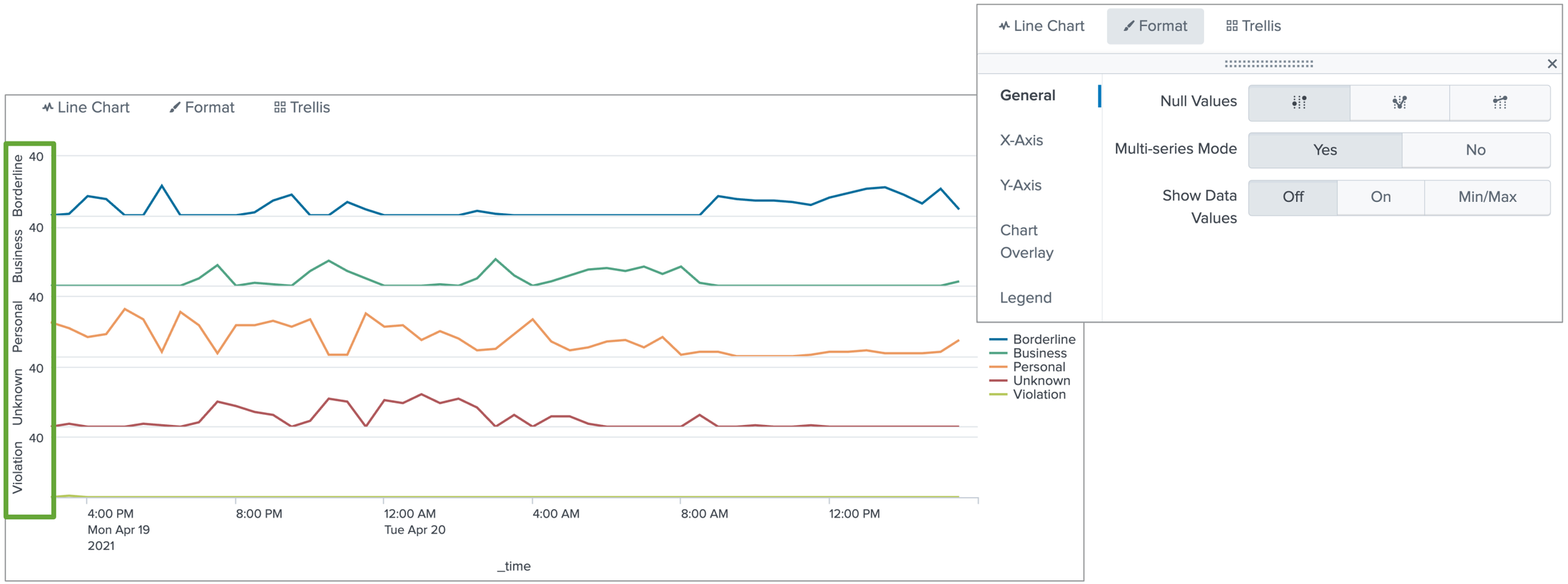
By default, multi-series time charts share the same Y-axis

```
index=network sourcetype=cisco_wsa_squid
| timechart count BY usage
```



Toggle Multi-series Mode (cont.)

- Split the Y-axis for each value by toggling Multi-series Mode to **Yes**
- Y-axis is divided into sections of equal span



`top` Command

```
...| top (<field>|<field-list>) [by <field>] [countfield=<string>]
     [limit=<int>] [showperc=<bool>]
```

- Finds the most common values for a **`<field>`** or **`<field-list>`**
- By default, outputs top 10 results in table format
- Group results with a **`by <field>`** clause
- Control behavior with **`countfield`**, **`limit`**, and **`showperc`** options

top Command Example 1

Scenario

Determine which IP addresses generated the most attacks in the last 60 minutes.

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

src_ip	count	percent
132.55.227.221	105	27.131783
214.156.206.45	9	2.325581
118.6.85.68	9	2.325581
87.194.216.51	8	2.067183
108.65.113.83	6	1.550388
89.11.192.18	5	1.291990
211.166.11.101	5	1.291990
195.69.160.22	5	1.291990
81.18.148.190	4	1.033592
223.205.219.67	4	1.033592

`top` Command: `limit` Example

- Control # of results displayed with `limit` option
- Use `limit=0` for unlimited results

Scenario

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

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

src_ip	count	percent
64.185.135.189	221	2.187902
87.194.216.51	173	1.712702
211.166.11.101	167	1.653302
214.63.137.251	152	1.504802
194.124.246.74	145	1.435501

`top` Command: `field-list` Example

Find top values for specific combinations of fields by listing two or more fields

```
index=network sourcetype=cisco_wsa_squid
| top cs_username
```

cs_username	count	percent
yschonegge@buttercupgames.com	78	6.388206
acurry@buttercupgames.com	56	4.586405
pbridgland@buttercupgames.com	46	3.767404
syoungin@buttercupgames.com	43	3.521704
kjoslin@buttercupgames.com	41	3.357903
sscallion@buttercupgames.com	38	3.112203
moh@buttercupgames.com	35	2.866503
basselin@buttercupgames.com	33	2.702703
pleuchs@buttercupgames.com	32	2.620803
dpiazza@buttercupgames.com	31	2.538903

All `cs_username` values are unique

```
index=network sourcetype=cisco_wsa_squid
| top cs_username x_webcat_code_full
```

cs_username	x_webcat_code_full	count	percent
kjoslin@buttercupgames.com	Reference	36	2.948403
syoungin@buttercupgames.com	Uncategorized URLs	34	2.784603
yschonegge@buttercupgames.com	Shopping	31	2.538903
pbridgland@buttercupgames.com	Uncategorized URLs	31	2.538903
yschonegge@buttercupgames.com	Infrastructure	30	2.457002
basselin@buttercupgames.com	Arts and Entertainment	29	2.375102
apucci@buttercupgames.com	Shopping	26	2.129402
moh@buttercupgames.com	Health and Nutrition	25	2.047502
cganttchart@buttercupgames.com	Uncategorized URLs	25	2.047502
blu@buttercupgames.com	Uncategorized URLs	23	1.883702

All combinations of `cs_username` and `x_webcat_code_full` values are unique

`top` Command: `showperc` Example

- By default, a percent column is displayed; **`showperc=t`**
- Use **`showperc=f`** to remove the percent column

Scenario

Display the top 5 common values for users and web categories browsed during the last 24 hours.

```
index=network sourcetype=cisco_wsa_squid
| top cs_username x_webcat_code_full limit=5 showperc=f
```

cs_username	x_webcat_code_full	count
kjoslin@buttercupgames.com	Reference	36
syoungin@buttercupgames.com	Uncategorized URLs	34
yschonegge@buttercupgames.com	Shopping	31
pbridgland@buttercupgames.com	Uncategorized URLs	31
yschonegge@buttercupgames.com	Infrastructure	30

`top` Command: `countfield` Example

Rename the count field by specifying a string with `countfield`

Scenario

Display the top 5 common values for users and web categories browsed during the last 24 hours. Label the count field as "Total Viewed".

```
index=network sourcetype=cisco_wsa_squid
| top cs_username x_webcat_code_full limit=5 showperc=f
  countfield="Total Viewed"
```

cs_username	x_webcat_code_full	Total Viewed
syoungin@buttercupgames.com	Uncategorized URLs	34
yschonegge@buttercupgames.com	Shopping	31
pbridgland@buttercupgames.com	Uncategorized URLs	31
yschonegge@buttercupgames.com	Infrastructure	30
basselin@buttercupgames.com	Arts and Entertainment	29

top Command: by Clause Example

Scenario

Display the top 5 users for each web category during the last 24 hours.

```
index=network sourcetype=cisco_wsa_squid
| top cs_username by x_webcat_code_full
  limit=5 showperc=f countfield="Total Viewed"
```

x_webcat_code_full	cs_username	Total Viewed
Advertisements	pleuchs@buttercupgames.com	10
Advertisements	gzuyeva@buttercupgames.com	3
Advertisements	acurry@buttercupgames.com	2
Arts and Entertainment	basselin@buttercupgames.com	29
Arts and Entertainment	myuan@buttercupgames.com	9
Arts and Entertainment	svoronoff@buttercupgames.com	4
Arts and Entertainment	cberztiss@buttercupgames.com	4
Arts and Entertainment	yschonegge@buttercupgames.com	3
Business and Industry	myuan@buttercupgames.com	23

All web categories are listed with the top 5 users of each category

Scenario

Display the top 5 web categories browsed by each user during the last 24 hours.

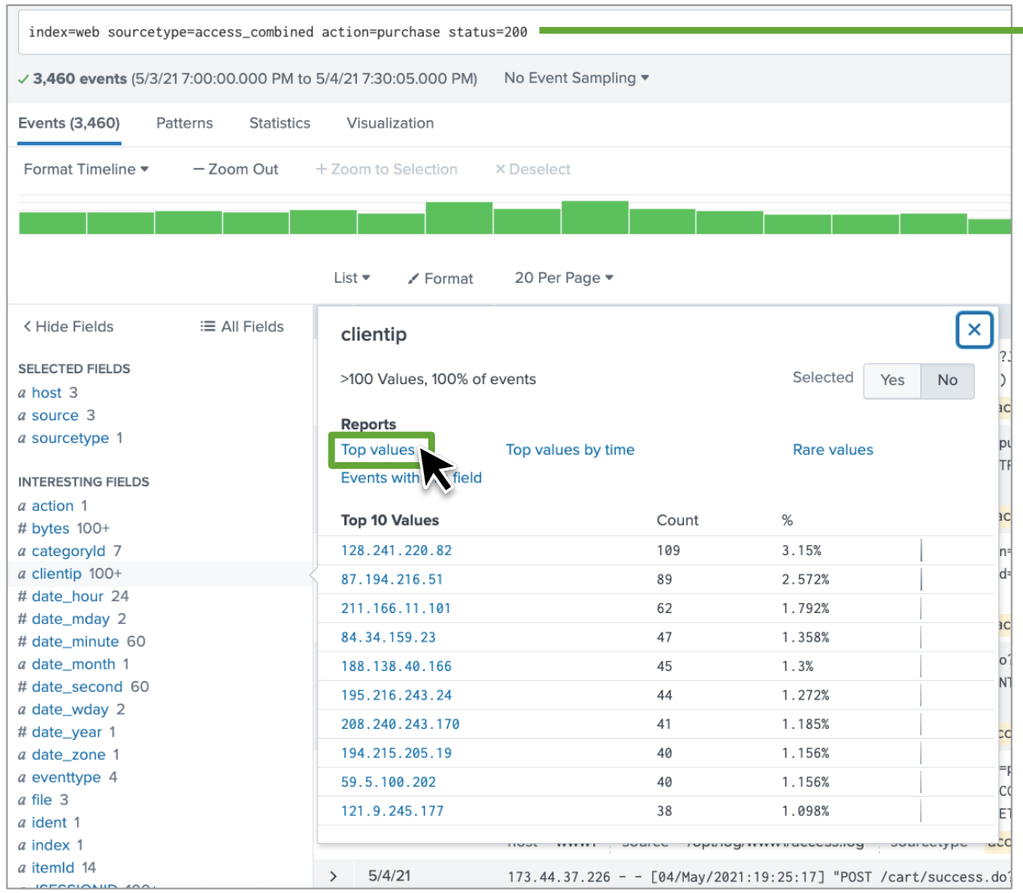
```
index=network sourcetype=cisco_wsa_squid
| top x_webcat_code_full by cs_username
  limit=5 showperc=f countfield="Total Viewed"
```

cs_username	x_webcat_code_full	Total Viewed
acurry@buttercupgames.com	Uncategorized URLs	23
acurry@buttercupgames.com	Shopping	19
acurry@buttercupgames.com	Science and Technology	3
acurry@buttercupgames.com	News	2
acurry@buttercupgames.com	Advertisements	2
adombrowski@buttercupgames.com	Transportation	2
adombrowski@buttercupgames.com	Travel	1
adombrowski@buttercupgames.com	News	1
adombrowski@buttercupgames.com	Business and Industry	1

All users listed with their top 5 most visited web categories

`top` Command: Top Values

Click **Top Values** in the Field Window to automatically add the top 20 values to your search




```
index=web sourcetype=access_combined action=purchase status=200| top limit=20 clientip
```

Last 24 hours

Note

After clicking **Top Values**, your view changes to the **Visualization** tab by default.

`rare` Command

```
...| rare (<field>|<field-list>)
```

- Finds the least common field values for a **`<field>`** or **`<field-list>`**
- By default, results are sorted in ascending order based on count
- Uses the same options as the **`top`** command

Scenario

Identify which product is the least sold by Buttercup Games vendors over the last 60 minutes.

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

product_name	count
Benign Space Debris	44

Take a break!

 |

Topic 3: Statistical Aggregation with the `stats` Command

Topic Objectives

- Explore the stats command and its various functions
  - `count`
  - `dc`
  - `sum`
  - `min`
  - `max`
  - `avg`
  - `values`
  - `list`

What is Aggregation?

- "An aggregation is a process in which numbers are gathered for statistical purposes and are expressed as one number."

Beginners: Statistical Concept – Aggregate. (2020, February 14). Eurostat Statistics Explained. https://ec.europa.eu/eurostat/statistics-explained/index.php?title=Beginners:Statistical_concept_-_Aggregate.

- "Aggregation denotes the compounding of primary data into an aggregate, usually for the purpose of expressing them in a summary form."

Aggregation. (2013, June 10). Organization for Economic Co-Operation and Development. https://stats.oecd.org/glossary/detail.asp?ID=68.

- Examples in this topic will demonstrate how to use the functions of the `stats` command to aggregate:
  - All event data into a single number
  - Data from a specified field into a single number
  - All or some event data over values of a specific field(s)

`stats` Command

```
...| stats <stats-function>(<wc-field>) [as <wc-field>] [by <field-list>]
```

- Uses various functions to calculate statistics on search results
- Returns a results table containing the output of **`<stats-function>`** on **`<wc-field>`**
- Group results with a **`by <field-list>`** clause

Note

The **`wc`** in **`<wc-field>`** means the stats command supports wildcarded fields.

`stats` Command: Functions

- There are 4 categories of statistical functions:
  - Aggregate: summarizes event values to create a single value
  - Event Order: returns values from fields based on processing order
  - Multivalue: returns lists of values for a field as a multivalue entry
  - Time: returns values based on time

Note

Two aggregate functions, **`count`** and **`sum`**, have been used in previous examples with **`chart`** and **`timechart`**.

`stats` Command: Functions (cont.)

This topic primarily focuses on aggregate functions

Function Category	Function	Description
Aggregate	`count`	Returns the count of events
	`count(X)`	Returns the number of events with a field value for the field `X`
	`dc(X)`	Returns a count of unique values for `X`
	`sum(X)`	Returns a sum of numeric values for `X`
	`min(X)`	Returns the minimum value of `X`
	`max(X)`	Returns the maximum value of `X`
	`avg(X)`	Returns the average value of `X`
	`median(X)`	Returns the middle-most value of `X`
	`range(X)`	Returns the difference between the min and max values of `X`
	`stdev(X)`	Returns the standard deviation of `X`
	`var(X)`	Returns the variance of `X`
Multivalue	`list(X)`	Lists all values of `X`
	`values(X)`	Lists unique values of `X`

Note

This is not a full list of supported functions.

`stats` Command: `count` Example 1

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

Scenario

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

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

count
367

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

Potential Issues
367

`stats` Command: `as` Clause

If no `as` clause is used, fields are named using the search syntax

```
index=security sourcetype=linux_secure
| stats count(vendor_action), count
```

count(vendor_action)	count
68	79

Note

This applies to all `stats` functions, not just `count`.

`stats` Command: `count` Example 2

Use with a field to count the number of events where a value is present for the specified field

Scenario

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

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

ActionEvents	TotalEvents
68	79

Note

Use multiple functions with the `stats` command, separated by commas.

`stats` Command: `by` Clause

- Groups count values by a named field or set of fields
- Fields display in the same order they are listed

Scenario

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

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

user	app	vendor_action	count
adm	sshd	Failed	1
admin	sshd	Failed	1
administrator	sshd	Failed	2
backup	sshd	Failed	1
beyonce	sshd	Failed	1
bfsuser	sshd	Failed	1

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

app	user	vendor_action	count
sshd	adm	Failed	1
sshd	admin	Failed	1
sshd	administrator	Failed	2
sshd	backup	Failed	1
sshd	beyonce	Failed	1
sshd	bfsuser	Failed	1

`stats` Command: `dc` Example

`dc()` or `distinct_count()` provides a count of the unique values for a given field in the results set

Scenario

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

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

Websites visited:
17

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

Websites visited:
247

`stats` Command: `sum` Example

Calculates the sum of fields containing only numeric values

Scenario

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

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

A

s_hostname	Bandwidth
-	776393
1zs0ewvqcget52rl1z1n.cn	1899
8b5a18ozw9smaleujdvm2m9m30.hop.clickbank.net	1168
abapharm.net	1608
ad.doubleclick.net	320
afflvwetib.com	1872

Output from `stats` is not sorted

B

s_hostname	Bandwidth
zcomcdl.zcominc.com	2335808
www.adfusion.com	1930508
www.cartoonstock.com	1866622
www.billboard.com	1829816
www.xtremepowersports.com	1755650
www.lakeholiday.com	1718544

 |

`stats` Command: Multiple Functions

- Use multiple functions with the **`stats`** command
- If a **`by`** clause is used, the by clause applies to all functions

Scenario

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

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

product_name	Units Sold	Total Sales
Dream Crusher	606	24233.94
Manganiello Bros.	483	19315.17
World of Cheese	689	17218.11
Orvil the Wolverine	354	14156.46
SIM Cubicle	685	13693.15
Final Sequel	444	11095.56

Units Sold	Total Sales
6778	140052.22

Results without **by** clause

`stats` Command: `min`, `max`, `avg`

Scenario

What is the minimum, maximum, and average bandwidth used for each website usage type?

```
index=network sourcetype=cisco_wsa_squid
| stats min(sc_bytes) as "Minimum Bytes",
  max(sc_bytes) as "Maximum Bytes",
  avg(sc_bytes) as "Average Bytes" by usage
```

usage	Minimum Bytes	Maximum Bytes	Average Bytes
Borderline	395	195976	11673.848101265823
Business	304	149222	12769.720720720721
Personal	256	312773	11322.620955315871
Unknown	0	2335808	34007.03370786517
Violation	468	62311	6278.214285714285

Note

The avg function ignores any event or row that does not have a value for the specified field or has a non-numerical value for the field.

`stats` Command: `values` Example

Lists all unique values for a specified field

Scenario

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

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

src_ip	User Names	Attempts
107.3.146.207	db dbase jessica postgres	4
111.187.254.243	nsharpe	102
109.169.32.135	admin info	2
110.138.30.229	jabber system vpxuser	3

 |

`stats` Command: `list` Example

Lists every value, including duplicates, for a specified field

Scenario

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

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

cs_username	Websites visited:
adombrowski@buttercupgames.com	www.ebgames.com www.ebgames.com
apucci@buttercupgames.com	www.ebgames.com www.ebgames.com www.ebgames.com
blu@buttercupgames.com	www.ebgames.com
cberztiss@buttercupgames.com	www.ebgames.com
cmunson@buttercupgames.com	www.collegegrad.com
cquinn@buttercupgames.com	www.ebgames.com www.ebgames.com
djohnson@buttercupgames.com	www.ebgames.com
edutra@buttercupgames.com	www.ebgames.com

Events that contain multiple values for **`Websites visited:`** are treated as multivalue entries

Note

The list function is very useful for creating easy-to-read data tables.

stats Command: list versus values

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

cs_username	Websites visited:
adombrowski@buttercupgames.com	www.ebgames.com www.ebgames.com
apucci@buttercupgames.com	www.ebgames.com www.ebgames.com www.ebgames.com
blu@buttercupgames.com	www.ebgames.com
cberztiss@buttercupgames.com	www.ebgames.com
cmunson@buttercupgames.com	www.collegegrad.com
cquinn@buttercupgames.com	www.ebgames.com www.ebgames.com
djohnson@buttercupgames.com	www.ebgames.com
edutra@buttercupgames.com	www.ebgames.com
ewarwick@buttercupgames.com	www.collegegrad.com www.ebgames.com

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

cs_username	Websites visited:
adombrowski@buttercupgames.com	www.ebgames.com
apucci@buttercupgames.com	www.ebgames.com
blu@buttercupgames.com	www.ebgames.com
cberztiss@buttercupgames.com	www.ebgames.com
cmunson@buttercupgames.com	www.collegegrad.com
cquinn@buttercupgames.com	www.ebgames.com
djohnson@buttercupgames.com	www.ebgames.com
edutra@buttercupgames.com	www.ebgames.com
ewarwick@buttercupgames.com	www.collegegrad.com www.ebgames.com

stats Command Example

Scenario

Marketing wants to perform a statistical analysis on all APAC retail sales by country over the last 24 hours.

```
index=sales sourcetype=vendor_sales VendorID>=7000 AND VendorID<9000
| stats count(price) as count, sum(price) as sum, min(price) as Minimum, max(price) as Maximum, range(price) as Range, mean(price) as Average, median(price) as Median, stdev(price) as sdev, var(price) as Variance by VendorCountry
| eval Average = round(Average,2), sdev=round(sdev,2), Variance=round(Variance,2)
| sort -sum
| rename count as "Units Sold", sum as "Total Sales", sdev as "Standard Deviation"
```

VendorCountry	Units Sold	Total Sales	Minimum	Maximum	Range	Average	Median	Standard Deviation	Variance
China (PRC)	26	585.74	3.99	39.99	36.00	22.53	24.99	12.22	149.22
India	24	453.76	3.99	39.99	36.00	18.91	17.49	13.81	190.60
Japan	16	336.84	4.99	39.99	35.00	21.05	24.99	12.24	149.93
South Korea	7	184.93	9.99	39.99	30.00	26.42	24.99	10.69	114.29
Australia	10	183.90	3.99	39.99	36.00	18.39	22.49	11.93	142.27
Taiwan (ROC)	6	139.94	3.99	39.99	36.00	23.32	24.99	15.72	247.07
Pakistan	8	124.92	3.99	24.99	21.00	15.61	17.49	10.17	103.41
New Zealand	5	94.95	3.99	39.99	36.00	18.99	5.99	19.18	368.00
Armenia	3	89.97	9.99	39.99	30.00	29.99	39.99	17.32	300.00
Sri Lanka	2	79.98	39.99	39.99	0.00	39.99	39.99	0.00	0.00

Transforming Commands Summary

Feature	`chart`	`timechart`	`stats`
by clause arguments	2	1	many
Limit series shown	`limit=int` (default `limit=10`)	`limit=int` (default `limit=10`)	NA
Filter other series	`useother=t`	`useother=t`	NA
Filter null values	`usenull=t`	`usenull=t`	NA
Set value groups along the x-axis	`span`	`span`	NA

Transforming Data Lab Exercise

Time: 15 minutes

Tasks:

- Use the **`chart`** and **`timechart`** commands to create multi-series results and visualizations
- Use the **`top`** command to identify the domains customers are using to get to the Buttercup Games online store
- Use the **`stats`** command to calculate badge swipes at all three Buttercup Games offices

Topic 4: Manipulating Data with `eval` Command

Topic Objectives

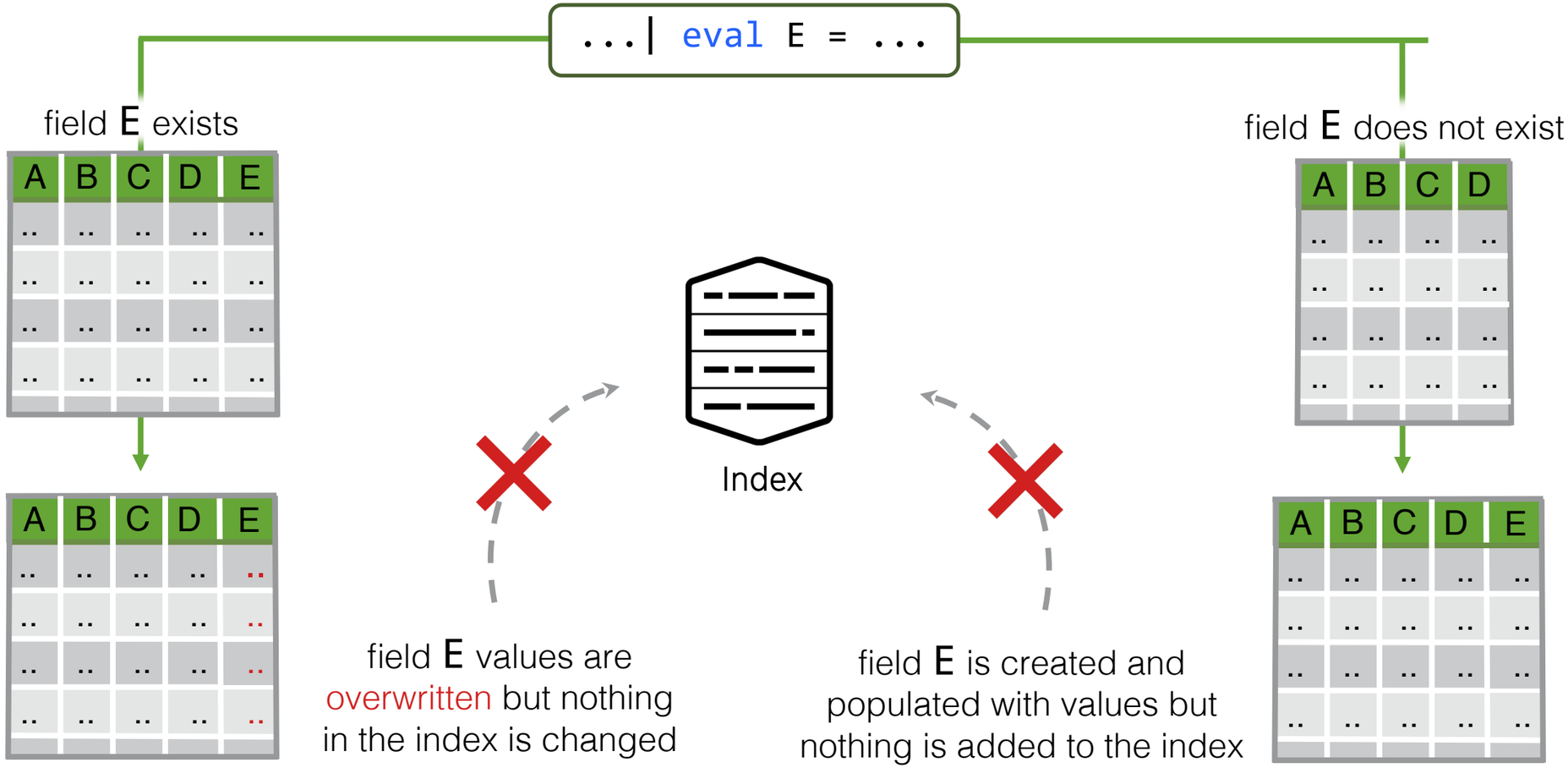
- Explore the **`eval`** command
- Explore evaluation functions for performing calculations:
  - `pow`
  - `round`
  - `min`
  - `max`
  - `random`
- Calculate and concatenate field values without functions
- Use **`eval`** as a function with **`stats`**

`eval` Command

```
...| eval <field1>=<expression1>[, <field2>=<expression2>]
```

- Calculates an expression and puts the resulting value into a new field or overwrites an existing field
- Fields created by **`eval`** can be reused in the search pipeline
- Extremely useful command that supports a vast assortment of functions for performing specific tasks
- Can exist as a nested function in certain scenarios

`eval` Command (cont.)



`eval` Command (cont.)

The `eval` command supports various operators

Type	Operators
arithmetic	`+ - * / %`
concatenation	`+ .`
Boolean	`AND OR NOT XOR`
comparison	`< > <= >= != = LIKE`

Note

This topic focuses on using concatenation operators with certain `eval` functions.

 |

eval Command Syntax

```
index=sales sourcetype=vendor_sales VendorCountry IN("United States", Canada)
| stats sum(price) as "USA+Canada Sales", count as "Total Products Sold"
  count(eval(VendorCountry = "United States")) as "Products Sold in US",
  count(eval(VendorCountry = "Canada")) as "Products Sold in Canada" by product_name
| eval "USA+Canada Sales" = "$".'USA+Canada Sales'
```

- Field values are treated in a case-sensitive manner
- String values must be "double-quoted"
- Field names must be unquoted or single quoted when they include a special character like a space
- Use a period (.) instead of (+) when concatenating strings and numbers to avoid conflicts

Ways to Write Multiple `eval`s

Expressions can be separate, nested, or linked with a comma

Separate **eval** pipeline segments

```
index=network sourcetype=cisco_wsa_squid
| stats sum(sc_bytes) as bytes by usage
| eval bandwidth = bytes/(1024*1024)
| eval bandwidth = round(bandwidth, 2)
```

Nested **eval** commands targeting the same field

```
index=network sourcetype=cisco_wsa_squid
| stats sum(sc_bytes) as bytes by usage
| eval bandwidth = round(bytes/(1024*1024), 2)
```

Combining **eval** commands with commas

```
index=network sourcetype=cisco_wsa_squid
| stats sum(sc_bytes) as bytes by usage
| eval bandwidth = bytes/(1024*1024),
  bandwidth = round(bandwidth, 2)
```

usage	bytes	bandwidth
Borderline	1298542	1.24
Business	2909449	2.77
Personal	9771346	9.32
Unknown	997092	0.95
Violation	495606	0.47

Note

round is a mathematical function that rounds field values up to a specified decimal place.

Referencing `eval` Fields

Temporary fields created using **`eval`** can be referenced in the search pipeline by succeeding commands

```
index=network sourcetype=cisco_wsa_squid
| stats sum(sc_bytes) as Bytes by usage
| eval bandwidth = round(Bytes/pow(1024,2), 2)
| sort -bandwidth
| rename bandwidth as "Bandwidth (MB)"
```

usage	Bytes	Bandwidth (MB)
Personal	299584913	285.71
Unknown	77187989	73.61
Business	66844576	63.75
Borderline	54011022	51.51
Violation	3203231	3.05

Note

`pow(X,Y)` is a mathematical function that returns the `X` to the power of `Y`.

Evaluation Functions

- Evaluates an expression based on your events and returns a result
- There are 11 categories of evaluation functions:
  - Conversion
  - Comparison and Conditional
  - Mathematical
  - Informational
  - Statistical
  - Text
  - etc.

Evaluation Functions (cont.)

This course discusses how to calculate with **`eval`** using functions and expressions

Category	Function Syntax	Description
Mathematical	`round(X,Y)`	Rounds `X` to `Y` decimal places, otherwise returns `X` as a whole number
	`pow(X,Y)`	Returns `X` to the power of `Y`
Statistical	`max(X,…)`	Takes an arbitrary number of arguments and returns the maximum
	`min(X,…)`	Takes an arbitrary number of arguments and returns the minimum
	`random()`	Takes no arguments and returns a random integer

Note

Not all functions are discussed in this course. This is not a full list.

`eval` Command: `pow` Function

```
...| eval <field> = pow(X,Y)
```

Returns `X` to the power of `Y`

Scenario

What types of websites used the most bandwidth in bytes during the previous month?

A

```
index=network sourcetype=cisco_wsa_squid
| stats sum(sc_bytes) as Bytes by usage
| eval bandwidth = Bytes/pow(1024,2)
```

usage	Bytes	bandwidth
Borderline	46922039	44.7483434677124
Business	62129350	59.25116539001465
Personal	304522017	290.41482639312744
Unknown	74295120	70.85334777832031
Violation	2830926	2.6997814178466797

New **`bandwidth`** field stores easy-to-read megabyte values

A

usage	Bytes
Borderline	46922039
Business	62129350
Personal	304522017
Unknown	74295120
Violation	2830926

Hard to determine bandwidth usage with just **`Bytes`**

`eval` Command: `round` Function

```
...| eval <field> = round(X,Y)
```

- Rounds **X** to the decimal places specified by **Y**
- Returns rounded whole numbers if no **Y** is specified

Scenario

What types of websites used the most bandwidth in bytes during the previous month?

```
index=network sourcetype=cisco_wsa_squid
| stats sum(sc_bytes) as Bytes by usage
| eval bandwidth = Bytes/pow(1024,2)
| eval bandwidth = round(bandwidth,2)
| sort -bandwidth
```

usage	Bytes	bandwidth
Personal	304522017	290.41
Unknown	74295120	70.85
Business	62129350	59.25
Borderline	46922039	44.75
Violation	2830926	2.70

A

usage	Bytes	bandwidth
Borderline	46922039	44.7483434677124
Business	62129350	59.25116539001465
Personal	304522017	290.41482639312744
Unknown	74295120	70.85334777832031
Violation	2830926	2.6997814178466797

Results before rounding and sorting

`eval` Command: `round` Function (cont.)

It is a common practice to wrap the **`round`** function around another statistical function

Scenario

What types of websites used the most bandwidth in bytes during the previous month?

```
index=network sourcetype=cisco_wsa_squid
| stats sum(sc_bytes) as Bytes by usage
| eval bandwidth = round(Bytes/pow(1024,2),2)
| sort -bandwidth
```

usage	Bytes	bandwidth
Personal	304522017	290.41
Unknown	74295120	70.85
Business	62129350	59.25
Borderline	46922039	44.75
Violation	2830926	2.70

`eval` Command: `max`, `min`, `random`

`max`: takes an arbitrary number of numeric or string arguments and returns the maximum

```
...| eval field1 = max(X1,X2,...)
```

`min`: takes an arbitrary number of numeric or string arguments and returns the minimum

```
...| eval field1 = min(X1,X2...)
```

`random`: takes no arguments and returns a pseudo-random integer ranging from zero to $2^{31}-1$

```
...| eval field1 = random()
```

Note

For both **`max`** and **`min`**, strings are considered greater than numbers.

`eval` Command: `random` Function Example

Scenario

The Sim Cubicle Beta team wants to test a new feature on random groups of users. Assign a random group number to unique user emails from the past 60 minutes.

```
index=games sourcetype=SimCubeBeta
| table User
| dedup User
| eval group = (random() % 5) + 1
| stats list(User) as Users by group
```

group	Users
1	chocolateswife@verizon.net sarah@jebian.com quint@msn.com ted@wesellmops.com kealyhogarth@live.com
2	guinita@tower2.com kdickerson@wca.org bashful@demo.com CEDWARDS@napavalleycomputers.com
3	fiveanddogs@gmail.com trulyblessed1@charter.net ebuceci@gmail.com ms2jb@comcast.net tmcbreen@mhcable.com paul@idrankyourmilkshake.com news@jebian.com mr@cbreshears.com

Note

The **%** (modulo) operator returns the remainder of a division. The remainder is always less than the divisor (in this example, the number 5) so using a modulo operator can give you control over your random numbers. If you want a range of random numbers, then just add the first range number to your operation. In this case, you want a range of 1 – 5, so you add 1.

 |

`eval` Expressions Without Functions

- Expressions do not require the use of a function
- Use operators to perform calculations or concatenate field values

Scenario

Calculate total online sales for last week; include price, sales price, and discount percentage. Sort by descending discount value.

```
index=web sourcetype=access_combined product_name=* action=purchase
| stats sum(price) as tlp, sum(sale_price) as tsp by product_name
| eval Discount = (((tlp - tsp)/ tlp)*100)
| eval Discount = round(Discount)
| sort -Discount
| eval Discount = Discount."%"
```

product_name	tlp	tsp	Discount
Puppies vs. Zombies	4151.68	1655.68	60%
Fire Resistance Suit of Provolone	4061.82	2025.82	50%
Holy Blade of Gouda	5103.48	2547.48	50%
Dream Crusher	41429.64	25889.64	38%
Manganiello Bros.	32631.84	20391.84	38%
Orvil the Wolverine	33271.68	20791.68	38%

Note

Sort numerical field values before converting to strings.

`eval` Command with Multiple Expressions

- Link two or more expressions together to create multiple fields
- Use functions or perform simple calculations and concatenations

Scenario

Calculate a new sale price that is 5% less than the current discount percentage for online sales data over the last hour.

```
index=web sourcetype=access_combined price=*
| stats values(price) as list_price, values(sale_price)
  as current_sale_price by product_name
| eval current_discount=round((list_price - current_sale_price)/list_price*100,2),
  new_discount=(current_discount - 5),
  new_sale_price=list_price - (list_price * (new_discount/100))
```

product_name	list_price	current_sale_price	current_discount	new_discount	new_sale_price
Benign Space Debris	24.99	19.99	20.01	15.01	21.24
Curling 2014	19.99	16.99	15.01	10.01	17.99
Dream Crusher	39.99	24.99	37.51	32.51	26.99
Final Sequel	24.99	16.99	32.01	27.01	18.24
Fire Resistance Suit of Provolone	3.99	1.99	50.13	45.13	2.19
Holy Blade of Gouda	5.99	2.99	50.08	45.08	3.29

`eval` Command as a Function

- Nest within the **`stats count`** function to count events with a specific field value
- Field values are case sensitive and should be wrapped in double quotes
- Requires an **`as`** clause to rename the field

Scenario

Count the number of events that occurred yesterday where the vendor action was Accepted, Failed, or session opened.

```
index=security sourcetype=linux_secure vendor_action=*
| stats count(eval(vendor_action="Accepted")) as Accepted,
  count(eval(vendor_action="Failed")) as Failed,
  count(eval(vendor_action="session opened")) as SessionOpened
```

Accepted	Failed	SessionOpened
15691	291694	20893

 |

Manipulating Data Lab Exercise

Time: 15 minutes

Tasks:

- Use the **`stats`** and **`eval`** commands to transform and manipulate online sales data
- Use **`timechart`** and **`eval`** commands to convert bytes to megabytes and visualize the results

Topic 5:
Formatting Data

Topic Objectives

- Use the **`rename`** command
- Use the **`sort`** command

`rename` Command

```
...| rename <field> as <newfield>
```

- Useful for giving fields more meaningful and user-friendly names
- When including spaces or special characters in field names, use double straight quotes, " "

`rename` Command Example

Scenario

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

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

clientip	Customer Action	ProductID	HTTP Status
12.130.60.4			200
27.102.11.11		SC-MG-G10	200
27.102.11.11		DC-SG-G02	200
27.102.11.11			200
27.102.11.11	view	FI-AG-G08	200
99.61.68.230	purchase		200
99.61.68.230	purchase	WC-SH-A01	200
99.61.68.230	addtocart	WC-SH-A01	200

 |

`rename` Command Example (cont.)

Once you rename a field, the new field name must be used in the rest of the search string

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

No results found.

```
index=web sourcetype=access_combined
| table clientip, action, productId, status
| rename productId AS ProductID,
  action AS "Customer Action",
  status AS "HTTP Status"
| sort "Customer Action", "HTTP Status"
```

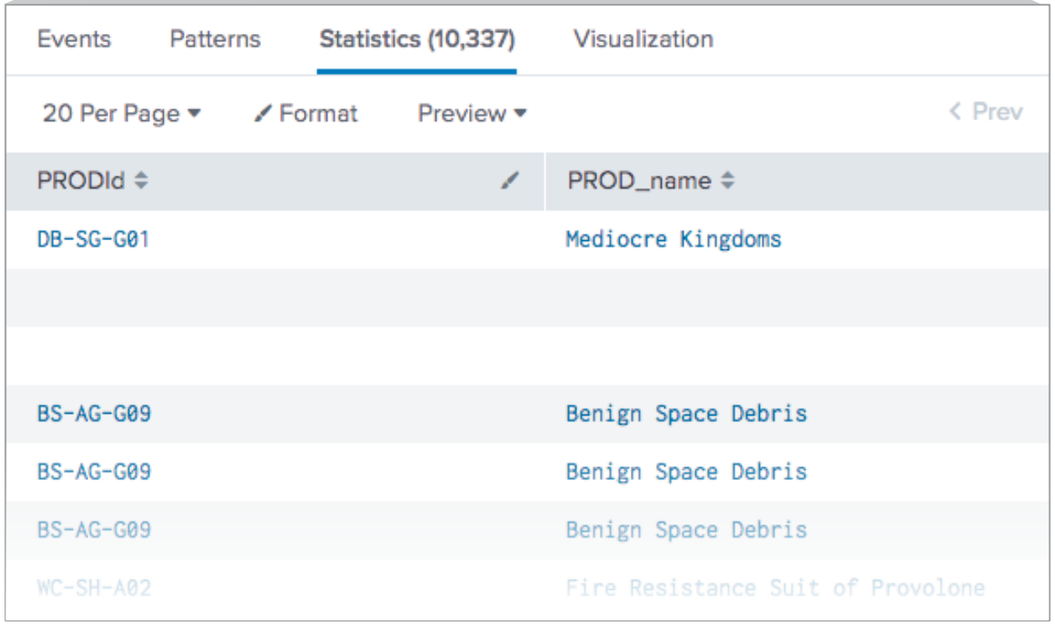
clientip	Customer Action	ProductID	HTTP Status
70.38.1.235	addtocart	WC-SH-A02	200
203.45.206.135	addtocart	DC-SG-G02	200
91.205.189.15	addtocart	MB-AG-G07	200

 |

`rename` Command Example (cont.)

Use wildcards (*) to rename multiple fields that match a pattern

```
index=web sourcetype=access_combined
| table productId, product_name
| rename product* AS PROD*
| table PROD*
```



`sort` Command

```
...| sort (-|+) <field> [limit=<int> | <int>]
```

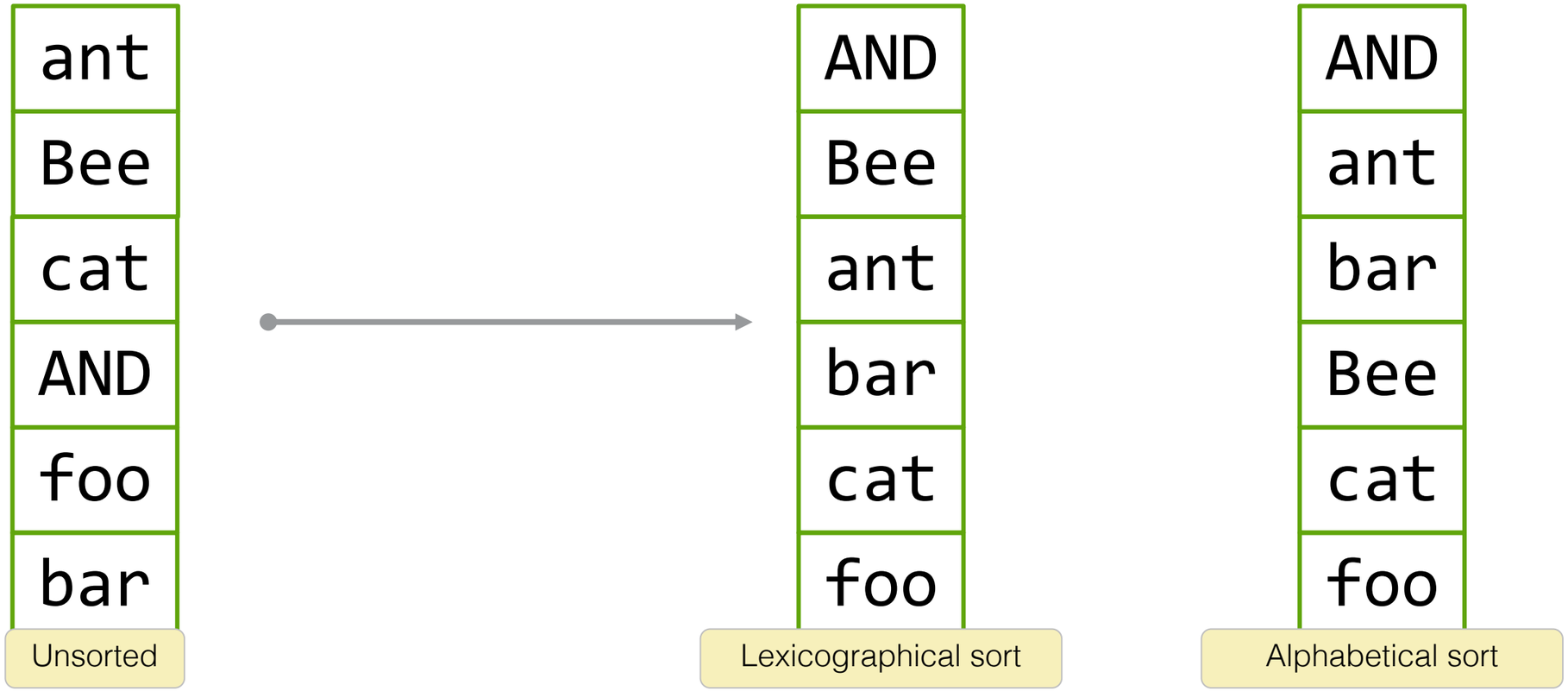
- Sorts results of a **`<field>`** in descending (**`-`**) or ascending (**`+`**) order
- Sorts in ascending order by default
- Limit results returned with the **`limit`** option or just include a number
- Determines field type then determines sorting type
  - Alphabetic strings = lexicographic sort
  - Numeric = numerical sort
  - Combination = lexicographic or numeric sort based on first character

Note

If the field name on which you're sorting contains spaces or non-alphanumeric characters, enclose it in double quotes.

`sort` Command: Alphabetic Strings

Sorted lexicographically so that uppercase letters appear before lowercase letters



sort Command Example

Scenario

Display vendor information for vendors that begin with '"Bea." Sort results in descending order.

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

A

A

Vendor	VendorCity	VendorStateProvince	VendorCountry
Beads & Games	Ft. Riley	Kansas	United States
Beauty Games	Butte	Montana	United States
Beantown Games	Boston	Massachusetts	United States
Beach Games	Miami	Florida	United States
Beach Games	Fort Lauderdale	Florida	United States

Before sorting

Vendor	VendorCity	VendorStateProvince	VendorCountry
Beauty Games	Butte	Montana	United States
Beantown Games	Boston	Massachusetts	United States
Beads & Games	Ft. Riley	Kansas	United States
Beach Games	Miami	Florida	United States
Beach Games	Fort Lauderdale	Florida	United States

After sorting

 |

`sort` Command Example (cont.)

Include a space after +/- if you want to apply the sort order to more than one field

Vendor	VendorCity	VendorStateProvince	VendorCountry
Beads & Games	Ft. Riley	Kansas	United States
Beauty Games	Butte	Montana	United States
Beantown Games	Boston	Massachusetts	United States
Beach Games	Miami	Florida	United States
Beach Games	Fort Lauderdale	Florida	United States

```
...
| sort -Vendor, VendorCity
```

Vendor	VendorCity	VendorStateProvince	VendorCountry
Beauty Games	Butte	Montana	United States
Beantown Games	Boston	Massachusetts	United States
Beads & Games	Ft. Riley	Kansas	United States
Beach Games	Fort Lauderdale	Florida	United States
Beach Games	Miami	Florida	United States

Different sort order as **Vendor** (ascending)

```
...
| sort - Vendor, VendorCity
```

Vendor	VendorCity	VendorStateProvince	VendorCountry
Beauty Games	Butte	Montana	United States
Beantown Games	Boston	Massachusetts	United States
Beads & Games	Ft. Riley	Kansas	United States
Beach Games	Miami	Florida	United States
Beach Games	Fort Lauderdale	Florida	United States

Same sort order as **Vendor** (descending)

Formatting Data Lab Exercise

Time: 10 minutes

Tasks:

- Use **`timechart`** and **`sort`** commands to find and sort events where web sales were twice as profitable as retail sales
- Use the **`stats`**, **`eval`**, **`sort`**, and **`rename`** commands to fulfill a request from the Sales department

Wrap-up Slides

Wrap-up

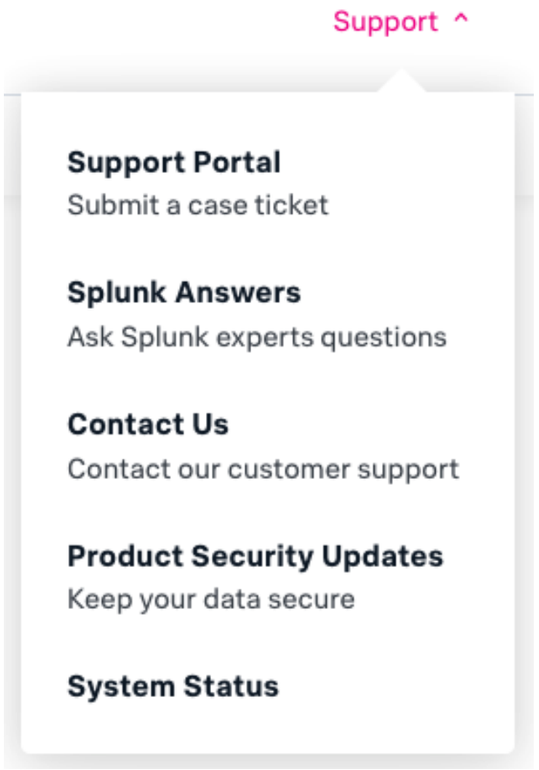
- You should now be able to:
  - Create single, multi, and time data series
  - Create chartable results with **`chart`** and **`timechart`**
  - Perform statistical aggregations with **`stats`**
  - Manipulate data with **`eval`**
  - Use eval as a function with **`stats`**
  - Sort results in descending or ascending order
  - Rename fields

Community

- Splunk Community Portal
  community.splunk.com
  - Answers
  - Discussions
  - Splunk Trust
  - User Groups
  - Ideas
- Splunk Blogs
  splunk.com/blog/
- Splunk Apps
  splunkbase.com
- Splunk Dev Google Group
  groups.google.com/forum/#!forum/splunkdev
- Splunk Docs on Twitter
  twitter.com/splunkdocs
- Splunk Dev on Twitter
  twitter.com/splunkdev
- Splunk Live!
  splunklive.splunk.com
- .conf
  conf.splunk.com

Support Programs

- **Web**
  - Documentation: dev.splunk.com and docs.splunk.com
  - Wiki: wiki.splunk.com
- **Splunk Lantern**
  Guidance from Splunk experts
  - lantern.splunk.com
- **Global Support**
  Support for critical issues, a dedicated resource to manage your account – 24 x 7 x 365
  - Web: splunk.com/index.php/submit_issue
- **Enterprise, Cloud, ITSI, Security Support**
  - Web: splunk.com/en_us/about-splunk/contact-us.html#tabs/customersupport
  - Phone: (855) SPLUNK-S or (855) 775-8657



Learning Paths

Search Expert - Recommended Courses

Free eLearning courses are in blue and courses with an * are present in both learning paths.

- What is Splunk *
- Introduction to Splunk *
- Using Fields *
- Scheduling Reports and Alerts
- Visualizations
- Statistical Processing
- Working with Time
- Comparing Values
- Result Modification
- Leveraging Lookups and Subsearches
- Correlation Analysis
- Search Under the Hood
- Multivalue Fields
- Search Optimization *

Learning Paths (cont.)

Knowledge Manager - Recommended Courses

Free eLearning courses are in blue and courses with an * are present in both learning paths.

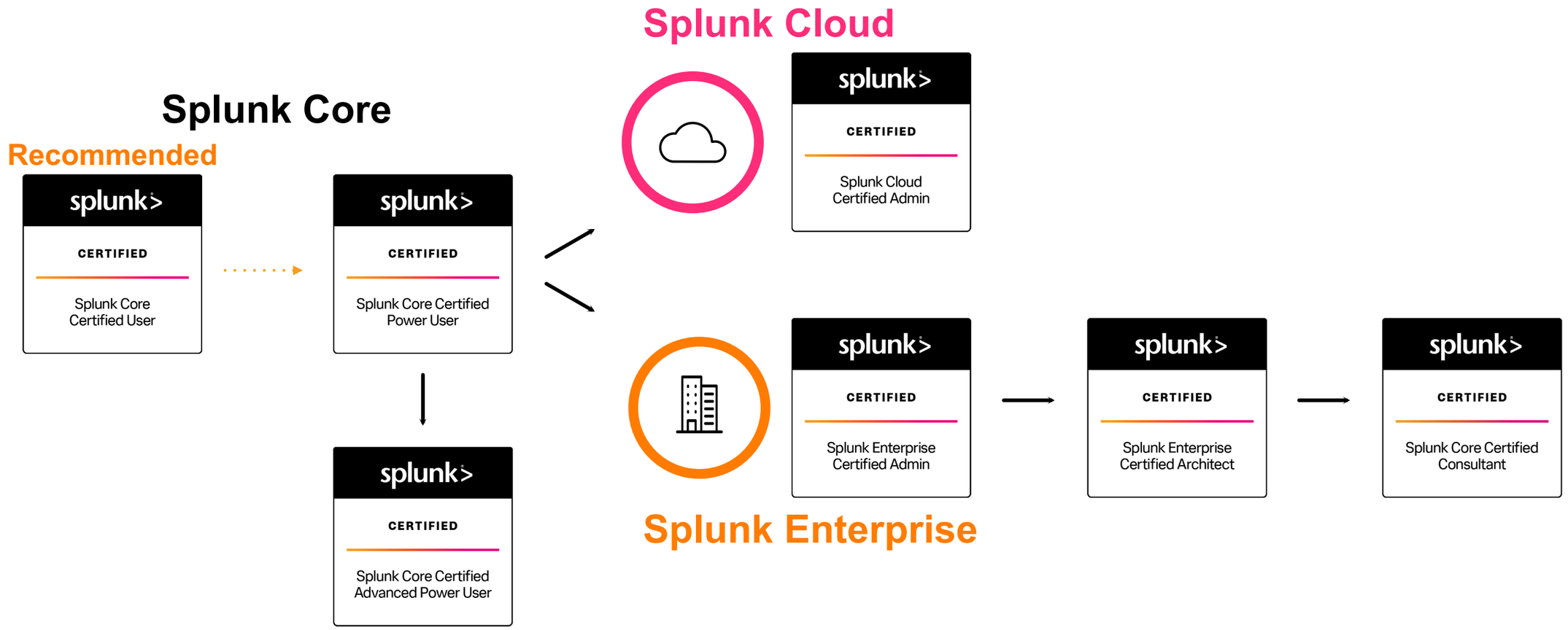
- What is Splunk *
- Introduction to Splunk *
- Using Fields *
- Introduction to Knowledge Objects
- Creating Knowledge Objects
- Creating Field Extractions
- Enriching Data with Lookups
- Data Models
- Introduction to Dashboards
- Dynamic Dashboards
- Using Choropleth
- Search Optimization *

Splunk Certification

Offerings & Requirements

Splunk Core and Beyond

Regardless of which Splunk product you use, it all starts with Splunk Core



App-Specific Offerings

For Splunk Add-Ons

App Developer

ES Administration

ITSI Administration

SOAR Automation Developer

Splunk Core Certified User

This entry-level certification demonstrates an individual's basic ability to navigate and use Splunk software

Prerequisite Certification(s):

- None

Prerequisite Course(s):

- None

Splunk Core Certified User Exam

Time to study! We suggest candidates looking to prepare for this exam complete Fundamentals 1 **or** the following courses:

- What is Splunk?
- Intro to Splunk
- Using Fields
- Scheduling Reports and Alerts
- Visualizations
- Statistical Processing
- Working with Time
- Leveraging Lookups and Subsearches
- Search Optimization
- Enriching Data with Lookups
- Data Models

See here for registration assistance.

Congratulations! You are a...

Recommended Next Step

- Splunk Core Certified Power User

Splunk Core Certified Power User

This entry-level certification demonstrates an individual's foundational competence of Splunk's core software

Prerequisite Certification(s):

- None

Prerequisite Course(s):

- None

Splunk Core Certified Power User Exam

Time to study! We suggest candidates looking to prepare for this exam complete Fundamentals 2 **or** the following courses:

- Visualizations
- Statistical Processing
- Working with Time
- Comparing Values
- Result Modification
- Correlation Analysis
- Search Under the Hood
- Introduction to Knowledge Objects
- Creating Knowledge Objects
- Creating Field Extractions
- Data Models
- Using Choropleth

See here for registration assistance.

Congratulations! You are a...

Recommended Next Steps

- Splunk Core Certified Advanced Power User
- Splunk Enterprise Certified Admin
- Splunk Cloud Certified Admin

 | 26 September 2022

Splunk Core Certified Advanced Power User

This certification demonstrates an individual's ability to generate complex searches, reports, and dashboards with Splunk's core software to get the most out of their data

Prerequisite Certification(s):

- Splunk Core Certified Power User

Prerequisite Course(s):

- None

Splunk Core Certified Advanced Power User Exam

Time to study! We suggest candidates looking to prepare for this exam complete Fundamentals 3, Creating Dashboards, and Advanced Searching & Reporting **or** the following courses:

- Using Fields
- Working with Time
- Comparing Values
- Result Modification
- Leveraging Lookups and Subsearches
- Correlation Analysis
- Search Under the Hood
- Multivalue Fields
- Search Optimization
- Creating Field Extractions
- Enriching Data with Lookups
- Data Models
- Using Choropleth
- Introduction to Dashboards
- Dynamic Dashboards

See here for registration assistance.

Congratulations! You are a...

Recommended Next Steps

- Splunk Enterprise Certified Admin
- Splunk Cloud Certified Admin

 | 26 September 2022

Splunk Cloud Certified Admin

This certification demonstrates an individual's ability to support the day-to-day administration and health of a Splunk Cloud environment

Prerequisite Certification(s):

- Splunk Core Certified Power User

Prerequisite Course(s):

- None

Splunk Cloud Certified Admin Exam

Time to study! We suggest candidates looking to prepare for this exam complete **either** the Splunk Cloud Administration **or** the Transitioning to Splunk Cloud course.

Both courses will equally prepare candidates for the exam, but are tailored to meet the needs of the individual based on prior Splunk experience.

Splunk Cloud Administration is designed for net-new administrators working in a Splunk Cloud environment. **Transitioning to Splunk Cloud** is for experienced Enterprise administrators looking to maximize their success in migrating to a Cloud environment.

See here for registration assistance.

Congratulations! You are a...

Recommended Next Steps

- Splunk Certified Developer

Splunk Enterprise Certified Admin

This certification demonstrates an individual's ability to support the day-to-day administration and health of a Splunk Enterprise environment

Prerequisite Certification(s):

- Splunk Core Certified Power User

Prerequisite Course(s):

- None

Splunk Enterprise Certified Admin Exam

Time to study! We suggest candidates looking to prepare for this exam complete the following courses:

- Splunk System Administration
- Splunk Data Administration

See here for registration assistance.

Congratulations! You are a...

Recommended Next Steps

- Splunk Enterprise Certified Architect
- Splunk Certified Developer

Splunk Certified Architect

This certification demonstrates an individual's ability to deploy, manage, and troubleshoot complex Splunk Enterprise environments

Prerequisite Certification(s):

- Splunk Core Certified Power User
- Splunk Enterprise Certified Admin

Prerequisite Course(s):

- Architecting Splunk Enterprise Deployments
- Troubleshooting Splunk Enterprise
- Splunk Cluster Administration
- Splunk Deployment Practical Lab

Splunk Enterprise Certified Architect Exam

Time to study! We **require** candidates looking to register for this exam to complete the following prerequisite courses:

- Architecting Splunk Enterprise Deployments
- Troubleshooting Splunk Enterprise
- Splunk Cluster Administration
- Splunk Deployment Practical Lab

Candidates who are **Splunk Enterprise Certified Admin** and have completed all of the above courses will automatically receive an exam authorization for the Splunk Enterprise Certified Architect exam within 5-7 business days of receiving their passing lab results.

See here for registration assistance.

Congratulations! You are a...

Recommended Next Steps

- Splunk Core Certified Consultant

Splunk Core Certified Consultant

This certification demonstrates an individual's ability to properly size, install, and implement Splunk environments and to advise others on how to utilize the product and maximize its value for their needs

Prerequisite Certification(s):

- Splunk Core Certified Power User
- Splunk Enterprise Certified Admin
- Splunk Enterprise Certified Architect

Prerequisite Course(s):

- *Advanced Power User courses **or** digital badge**
- Core Consultant Labs
  - Indexer Cluster Implementation
  - Distributed Search Migration
  - Implementation Fundamentals
  - Architect Implementation 1-3
- Services Core Implementation

Splunk Core Certified Consultant Exam

Time to study! We **require** candidates looking to register for this exam to complete the following prerequisite courses:

- *Fundamentals 3*, *Creating Dashboards*, *Advanced Searching & Reporting**
- Core Consultant Labs
- Services Core Implementation

Candidates who are **Splunk Enterprise Certified Architects** and have completed all of the above courses must contact certification@splunk.com to request their Core Consultant exam authorization.

See here for registration assistance.

These Advanced Power User courses can be replaced with a Splunk Certified Advanced Power User badge **or completion of the following courses:*

- Using Fields
- Creating Field Extractions
- Enriching Data with Lookups
- Data Models
- Search Optimization
- Working with Time
- Leveraging Lookups and Subsearches
- Comparing Values
- Correlation Analysis
- Result Modification
- Multivalue Fields
- Search Under the Hood
- Introduction to Dashboards
- Dynamic Dashboards
- Using Choropleth

Congratulations! You are a...

Recommended Next Steps

- None

 |

Splunk Certified Developer

This certification demonstrates an individual's expertise in drilldowns, advanced behaviors and visualizations, planning, creating, and packaging apps, and REST endpoints

Prerequisite Certification(s):

- Splunk Core Certified Power User

AND

- Splunk Enterprise Certified Admin

OR

- Splunk Cloud Certified Admin

Prerequisite Course(s):

- None

Splunk Certified Developer Exam

Time to study! We suggest candidates looking to prepare for this exam complete the following courses:

- Creating Dashboards with Splunk*
- Advanced Dashboards & Visualizations
- Building Splunk Apps
- Developing with Splunk's REST API

This course may also be substituted with the following newly-launched courses:

- Introduction to Dashboards
- Dynamic Dashboards
- Using Choropleth

See here for registration assistance.

Congratulations! You are a...

Recommended Next Steps

- None

Splunk Enterprise Security Certified Admin

This certification demonstrates an individual's ability to install, configure, and manage a Splunk Enterprise Security deployment

Prerequisite Certification(s):

- None

Prerequisite Course(s):

- None

Splunk Enterprise Security Certified Admin Exam

Time to study! We suggest candidates looking to prepare for this exam complete the following course:

- Administering Splunk Enterprise Security

Please note: all candidates are expected to have working knowledge and experience as either Splunk Cloud or Splunk Enterprise Administrators.

See here for registration assistance.

Congratulations! You are a...

Recommended Next Steps

- Splunk Phantom Certified Admin

Splunk IT Service Intelligence Certified Admin

This certification demonstrates an individual's ability to deploy, manage, and utilize Splunk ITSI to monitor mission-critical services

Prerequisite Certification(s):

- None

Prerequisite Course(s):

- None

Splunk IT Service Intelligence Certified Admin Exam

Time to study! We suggest candidates looking to prepare for this exam complete the following course:

- Implementing Splunk IT Service Intelligence

Please note: all candidates are expected to have working knowledge and experience as either Splunk Cloud or Splunk Enterprise Administrators.

See here for registration assistance.

Congratulations! You are a...

Recommended Next Steps

- Courses on Observability

Splunk SOAR Certified Automation Developer

This certification demonstrates an individual's ability to install and configure a SOAR server, integrate it with Splunk, and plan, design, create, and debug playbooks

Prerequisite Certification(s):

- None

Prerequisite Course(s):

- None

Splunk SOAR Certified Automation Developer Exam

Time to study! We suggest candidates looking to prepare for this exam complete the following courses:

- Administering SOAR (Phantom)
- Developing SOAR (Phantom) Playbooks
- Advanced SOAR (Phantom) Implementation

Please note: all candidates are expected to have working knowledge and experience as either Splunk Cloud or Splunk Enterprise Administrators.

See here for registration assistance.

Congratulations! You are a...

Recommended Next Steps

- None

Thank You