splunk> .conf2017

Advanced Splunk Searching for Security Hunting and Alerting

Stefan Hutchison | Sr. Security Engineer

September 26, 2017 | Washington, DC

What Makes a Security Alert

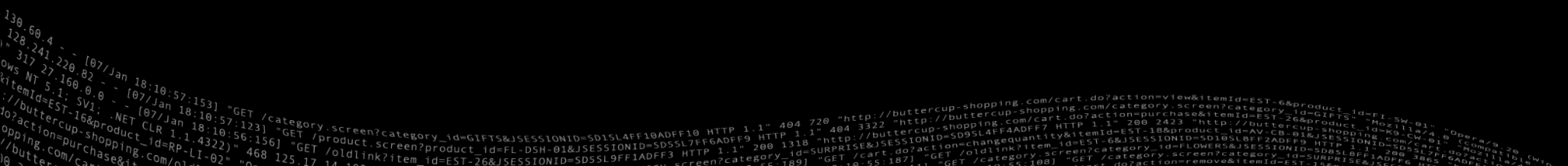
Its not just the SPL

- What question are you trying to answer?
- What are acceptable signal to noise ratios?
- What are acceptable false positive rates?

The Alert Creation Process

Step one: there's a fire!

- If you know TDD, you'll love IDD
- The best alerts start from having the data for something that is known to be bad.
- Incidents come from other alerting platforms or threat hunting
- | bucket _time span=15m | stats … by … _time
- Our SOC works closely with our Engineering team



Basic Search Commands

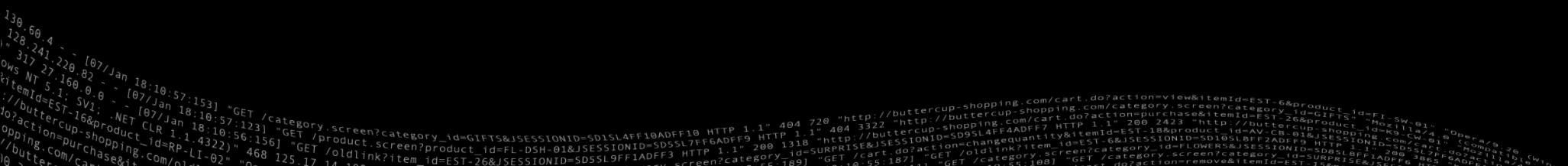
We all know and love them

- stats
  - Calculates aggregate statistics over result set
- eval
  - Calculates expression to create a new field
- where
  - Uses eval expression to filter events
- (input)lookup
  - Allows data to be pulled in from file sources

Basic Search Limitations

Stats can be your best friend, but it shouldn't be your only one

- The basic commands limit what security questions can be asked and answered.
- Alerts created with only these are often more static than can be useful.
- Streaming vs Generating commands



splunk> .conf2017

Conceptualizing data in Splunk terms

In order to master Splunk, you must think like Splunk

- Data in Splunk is a stream of events NOT rows in a table
- The order of events sometimes matters – sort is helpful here
- Tables are misleading

Advanced Search Commands

Meet your new friends

- Transaction
  - Groups events as transactions based off of constraints
- Eventstats
  - Same as stats, but applies the aggregates back to the raw data
- Streamstats
  - Same as stats, but aggregates the statistics as the events pass through the stream
- Chart
  - Applies statistics across the axes specified
- Stats (with eval clauses)
  - Using eval commands in stats aggregate functions to selectively include events in aggregates
  - | stats count, count(eval(like(src,”10.%"))) as internalSrc

splunk> .conf2017

Benefits of Advanced Commands

Level up your SPL!

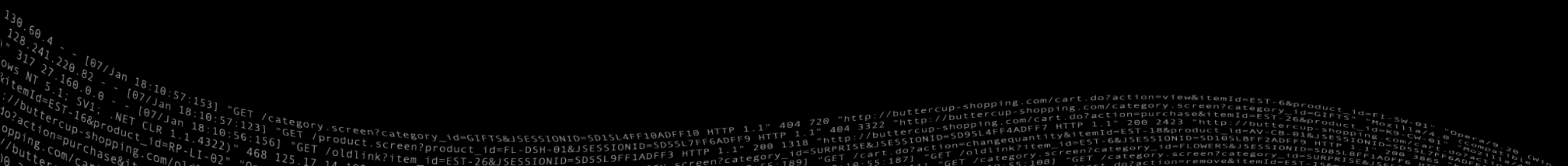
- Advanced commands expand question vocabulary
  - Allows for current events to be compared to historical trends without lookups
  - Ability to answer the question “Did x then y happen”?
    - “Is there a user that has failed logging in to 5 hosts, and was then able to successfully log in to a host?”
  - “Did a user log in from a new country with a new user agent today?”
- Eventstats and Streamstats are streaming commands, leaving the underlying data intact

splunk> .conf2017

Eventstats

Stats' more useful cousin

- Splunk comparisons in where and eval clauses are done on an event by event basis
- Makes the aggregate statistic available to compare against individual fields in the event
- This also retains all fields that were previously in the event for further analysis and context



Stats Limitations

```
index=[redacted]_logs src_user=[redacted]\\stefan.hutchison
| stats max(bytes), count, avg(bytes) by dest
```

Last 4 hours

✓ 225 events (8/14/17 11:39:00.000 PM to 8/15/17 3:39:20.000 AM) No Event Sampling

Job | Fast Mode

Events | Patterns | Statistics (5) | Visualization

100 Per Page | Format | Preview

dest	max(bytes)	count	avg(bytes)
[redacted].10.83	102332	60	19027.20588235294
[redacted].10.84	24845	36	4634.916666666667
[redacted].138.11	13598	7	3620.2
[redacted].239.138	82	54	80.81481481481481
[redacted].239.139	82	68	80.58823529411765

splunk> .conf2017

What Eventstats Provides

```
index=[redacted]_logs src_user=[redacted]\stefan.hutchison
| eventstats max(bytes) as max_bytes, count, avg(bytes) as avg_bytes by dest
| where bytes=max_bytes
| table src, app, dest, bytes, max_bytes, count, avg_bytes
```

Last 4 hours

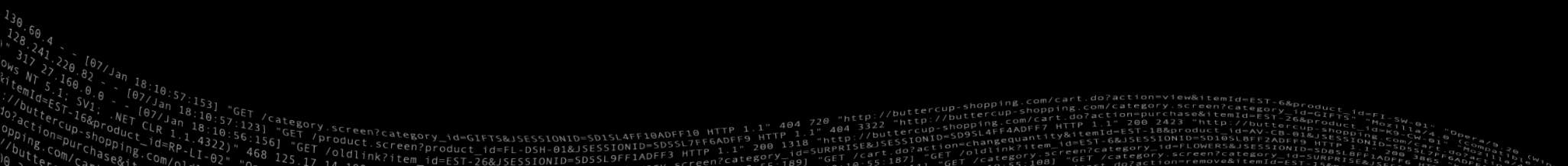
✓ 3 events (8/14/17 11:42:00.000 PM to 8/15/17 3:42:49.000 AM) No Event Sampling

Job Fast Mode

Events | Patterns | Statistics (3) | Visualization

100 Per Page | Format | Preview

src	app	dest	bytes	max_bytes	count	avg_bytes
[redacted].1.125	ssl	[redacted].10.83	102332	102332	71	16345.075
[redacted].1.125	splunk	[redacted].138.11	13598	13598	7	3620.2
[redacted].1.125	ssl	[redacted].10.84	24845	24845	36	4634.916666666667



splunk> .conf2017

Streamstats

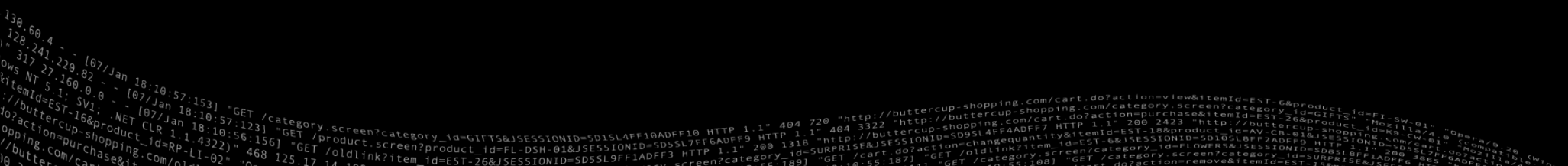
Stats' other more useful cousin… it's a big family

- Similar to eventstats where the aggregate functions are applied to the original events
- The value of the field changes as more events pass through the stream
- Example: " | streamstats count" will count 1, 2, 3, etc. until the last event

Transaction

One of the most expensive commands

- Joins events based off of shared fields to form a logical transaction
- Quite expensive, use it sparingly
- Adds the fields duration and eventcount
- Very useful when trying to determine if x then y



splunk> .conf2017

Eventstats and Transaction in Action

```
index=os sourcetype=sshd "Failed " OR "Accepted " host=[redacted]
| sort 0 src, dest, user, _time desc
| streamstats count as contiguous_action by src dest user action current=true window=50 reset_on_change=true
| where (tag="success" AND contiguous_action>3) OR (tag="error")
| transaction src dest user maxevents=2 startswith=eval(like(tag,"success")) endswith=eval(like(tag,"error"))
| rename contiguous_action as success_attempts
| stats count dc(dest) as distinct_dests sum(success_attempts) as "Total Successes" by user
```

Last 24 hours

✓ 1 event (8/13/17 9:00:00.000 PM to 8/14/17 9:17:14.000 PM) No Event Sampling Job Verbose Mode

Events (1) Patterns Statistics (1) Visualization

100 Per Page Format Preview

user	count	distinct_dests	Total Successes
stefan.hutchison	1	1	5

splunk> .conf2017

Under the Hood

```
index=os sourcetype=sshd "Failed " OR "Accepted " host=[redacted] stefan.hutchison
| sort 0 src, dest, user, _time asc
| streamstats count as contiguous_action by src dest user action current=true window=50 reset_on_change=true
| search src=*
| table _time src dest user tag contiguous_action
```

Last 24 hours

✓ 5 events (8/13/17 9:00:00.000 PM to 8/14/17 9:34:29.000 PM) No Event Sampling Job Verbose Mode

Events (5) Patterns Statistics (5) Visualization

100 Per Page Format Preview

_time ^	src	dest	user	tag	contiguous_action
2017-08-14 21:09:02	[redacted].2.12	[redacted]	stefan.hutchison	authentication remote success	1
2017-08-14 21:11:07	[redacted].2.12	[redacted]	stefan.hutchison	authentication remote success	2
2017-08-14 21:11:12	[redacted]2.12	[redacted]	stefan.hutchison	authentication remote success	3
2017-08-14 21:11:17	[redacted]2.12	[redacted]	stefan.hutchison	authentication remote success	4
2017-08-14 21:11:46	[redacted]2.12	[redacted]	stefan.hutchison	authentication error remote	1

Chart

And you thought we were done with the Stats family

- Calculates statistics (or eval values) based on multiple axes of data
- This can allow us to pivot the data
  - This is different from the splunk pivot functionality
- Allows us to compare the results of aggregate functions of 2 different groups

Chart in Action

```
| tstats `summariesonly` dc(All_Traffic.dest) as distinct_host from datamodel=Network_Traffic where nodename=All_Traffic  by All_Traffic.src, _time span=1h
| `drop_dm_object_name("All_Traffic")`
| eval StartTime=relative_time(now(),"-1h")
| eval Series=
    if(_time>=StartTime,"now","prior")
| chart avg(distinct_host) as avg, stdev(distinct_host) as stdev, count by src, Series
| where 'count: prior'>20 AND 'avg: now' > ('avg: prior' + 'stdev: prior')
```

Last 7 days

✓ 2,401,031,932 events (8/7/17 10:00:00.000 PM to 8/14/17 10:44:25.000 PM) No Event Sampling

Job Fast Mode

Events | Patterns | Statistics (2,697) | Visualization

100 Per Page | Format | Preview

< Prev 1 2 3 4 5 6 7 8 9 ... Next >

src	avg: now	avg: prior	count: now	count: prior	stdev: now	stdev: prior
[redacted].232.2	29641	593.6501706484642	1	8204	0	3031.7632627377234
[redacted].50.14	217	36.149950714637754	1	8116	0	82.3831277029888
[redacted].1.249	49	8.081126469448392	1	7741	0	15.324386725032245
[redacted].1.12	1422	149.99801429706116	1	7554	0	368.9941694255707
[redacted].1.131	822	69.99583836756612	1	7449	0	180.9665491188129
[redacted].232.3	492	88.5518817204301	1	7440	0	166.06792500755066
[redacted].0.131	774	68.7271112325772	1	7318	0	172.56760090292187
[redacted].56.2	22	3.7436437490597263	1	6647	0	5.9589978413830496

splunk> .conf2017

Key Takeaways

Hopefully

1. IDD and a process to keep Alerts relevant
2. For robust alerting, go beyond the basic search commands
3. It is necessary to think about your data the way Splunk does

Thank You

Don't forget to rate this session in the .conf2017 mobile app