Enterprise Security Health Check

Marquis Montgomery, CISSP | Sr. Staff Security Consultant, Splunk

October 2018 | Orlando, FL

Forward-Looking Statements

During the course of this presentation, we may make forward-looking statements regarding future events or the expected performance of the company. We caution you that such statements reflect our current expectations and estimates based on factors currently known to us and that actual events or results could differ materially. For important factors that may cause actual results to differ from those contained in our forward-looking statements, please review our filings with the SEC.

The forward-looking statements made in this presentation are being made as of the time and date of its live presentation. If reviewed after its live presentation, this presentation may not contain current or accurate information. We do not assume any obligation to update any forward-looking statements we may make. In addition, any information about our roadmap outlines our general product direction and is subject to change at any time without notice. It is for informational purposes only and shall not be incorporated into any contract or other commitment. Splunk undertakes no obligation either to develop the features or functionality described or to include any such feature or functionality in a future release.

splunk> .conf18

Introductions and Agenda

| rest /services/speakerinfo

Marquis Montgomery, CISSP, SSCP, GSEC

marquis@splunk.com / @trademarq

Sr. Staff Security Consultant, Splunk

- | where time@Splunk > 5 yrs
- Former customer, Manager of Corporate Security at MSSP
- Leads Enterprise Security Field Enablement
- “King of ES in PS”

Agenda

What will we be talking about today?

ES Under-The-Hood

Checking out the engine

Core Splunk Optimizations

Splunk Enterprise Platform Enhancements

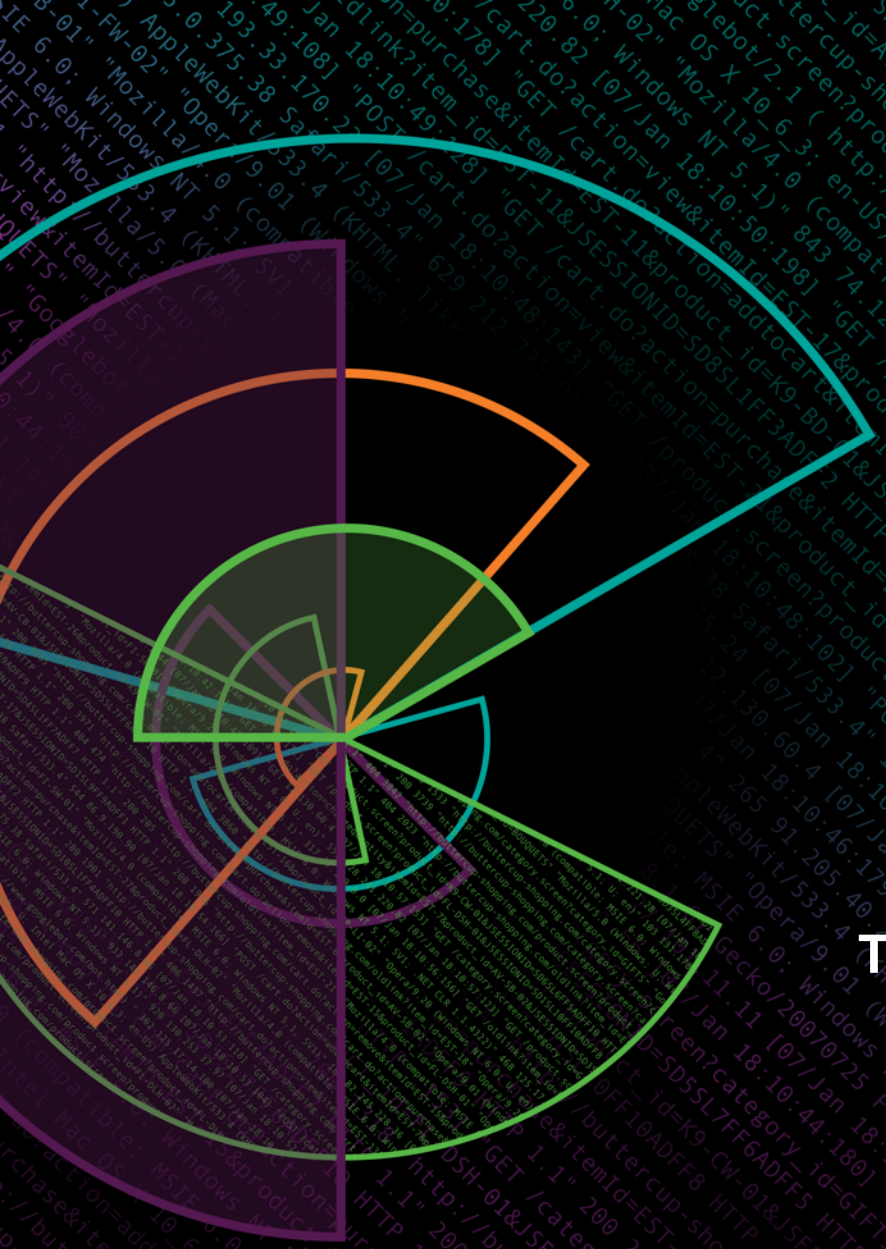
ES Specific Optimizations

Enhancements specific to the ES application

Key Takeaways and Q&A

Tying it all together

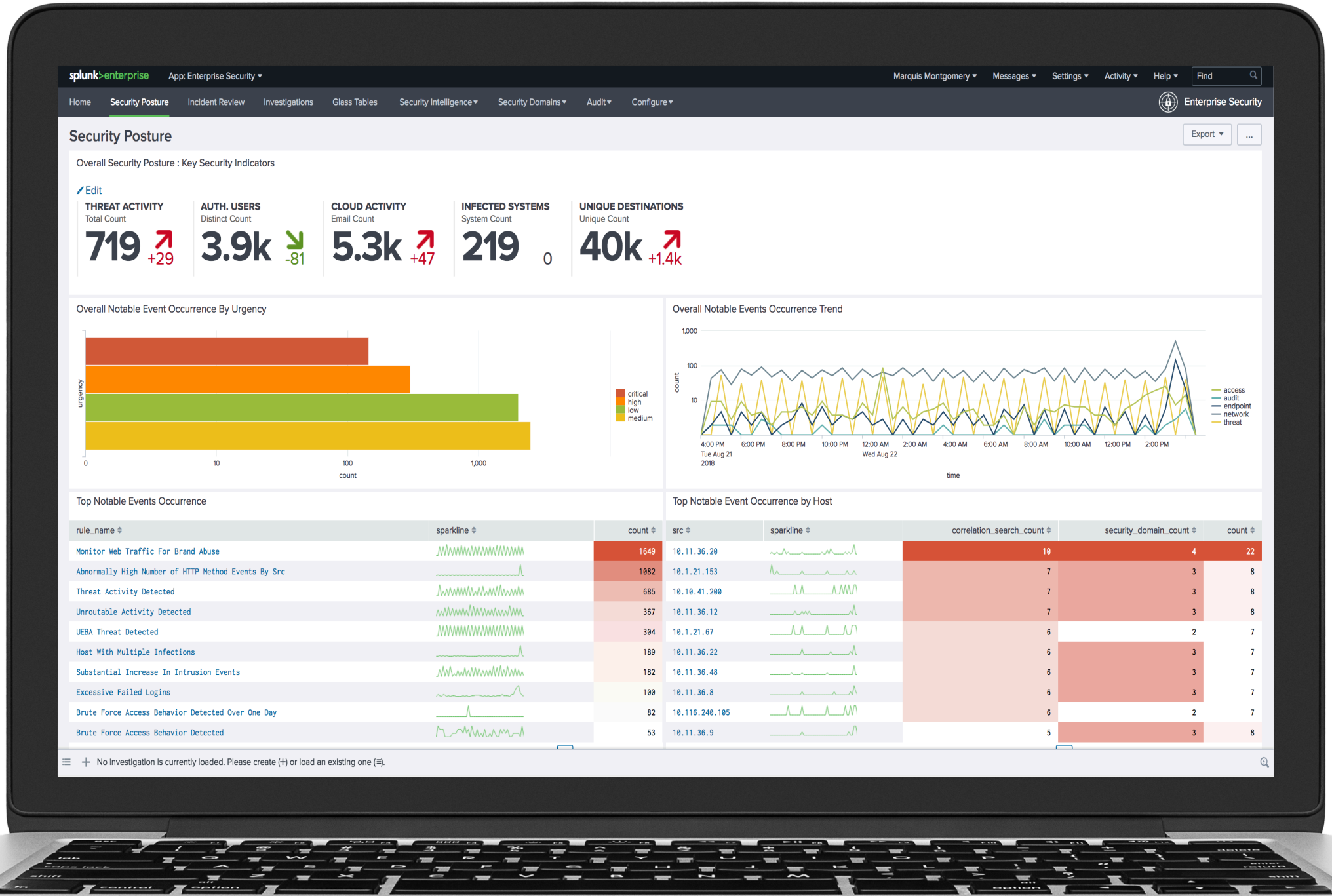
splunk> .conf18



Enterprise Security Under-The-Hood

To tune the engine, you need to understand the engine

splunk> .conf18



Things You Should Know About ES and Performance

Splunk Enterprise Security is a complex group of apps that work together, but at its core, it consists of the following components:

- Lots of Dashboards (and the ad-hoc searches that come with them)
- Scheduled Searches
  - Correlation Searches
  - Lookup Generator Searches
  - Context Generator Searches
  - Threat Generator Searches
  - Data Model Acceleration
- Lookup Tables
  - Assets & Identities Tables
  - Trackers
- KV Store Collections
  - Incident Review
  - Investigations

splunk> .conf18

Things You Should Know About ES and Performance

Splunk Enterprise Security is a complex group of apps that work together, but at its core, it consists of the following components:

- Lots of Dashboards (and the ad-hoc searches that come with them)
- Scheduled Searches
  - Correlation Searches
  - Lookup Generator Searches
  - Context Generator Searches
  - Threat Generator Searches
  - Data Model Acceleration
- Lookup Tables
  - Assets & Identities Tables
  - Trackers
- KV Store Collections
  - Incident Review
  - Investigations

All of these use CPU cycles and memory. You can directly control all of these.

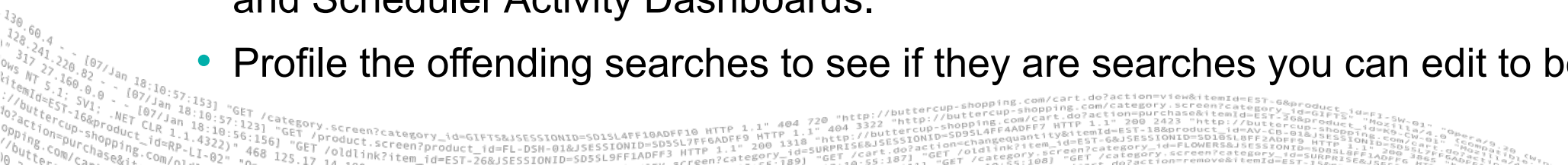
Understanding your profile and what you can control is half the battle!

splunk> .conf18

Things You Should Know About ES and Performance

So what can I control, and how do I know I need to make a change?

- In terms of searches, there is a finite number of concurrent searches a search head ***should*** run, and there are a finite number of concurrent searches a pool of indexers ***can*** run.
- You should be aware of how many concurrent searches your environment is running. The Monitoring Console instance is your friend! Use the Search > Search Activity dashboards to see these metrics.
- If you are getting great performance but you are approaching the limits, you can change them – we will talk about that soon.
- If you are not getting great performance, use other dashboards in Monitoring Console to understand why:
  - Look for Top Memory Consuming Searches and High Runtime Searches in Search Activity and Scheduler Activity Dashboards.
  - Profile the offending searches to see if they are searches you can edit to be more efficient.



splunk> .conf18

Things You Should Know About ES and Performance

So what can I control, and how do I know I need to make a change?

- In terms of Lookup Tables – this manifests itself in terms of Assets and Identities.
- These lookup tables can impact the amount of memory each search process requires. There is also a network and search setup impact in terms of compressing, transferring, and decompressing very large lookups.
- If you find that search setup time is taking too long, or your lookup performance is too long, evaluate whether or not the size of your lookup tables is absolutely necessary.
- Maybe it is acceptable to enrich only critical assets and identities in an automatic lookup, and leave enrichment for all others on an adhoc basis via a workflow action?

How Enterprise Security Works

Raw Event is Indexed

Data is generated, forwarded, and indexed in Splunk

Data Model Summary Search Runs

CIM DM normalization is applied, CIM DM field key/value pairs are stored in DM TSIDX

Data is now available for ES

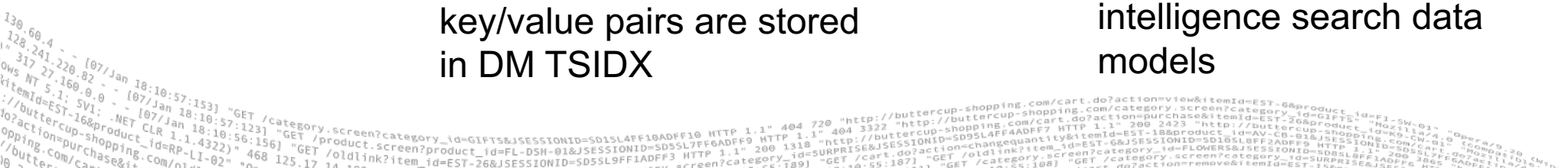
| tstats queries and dashboards now see data

ES Background Searches Process Data

Correlation Searches, trackers, and threat intelligence search data models

ES learns about threats and anomalies

ES writes these results in to notable events, summary indexing, and data models



Places We Can Increase Performance

HERE

Raw Event is Indexed

Data is generated, forwarded, and indexed in Splunk

HERE

Data Model Summary Search Runs

CIM DM normalization is applied, CIM DM field key/value pairs are stored in DM TSIDX

Data is now available for ES

| tstats queries and dashboards now see data

HERE

ES Background Searches Process Data

Correlation Searches, trackers, and threat intelligence search data models

ES learns about threats and anomalies

ES writes these results in to notable events, summary indexing, and data models

HERE

Core Splunk Optimizations

The Machine Data Platform

splunk> .conf18

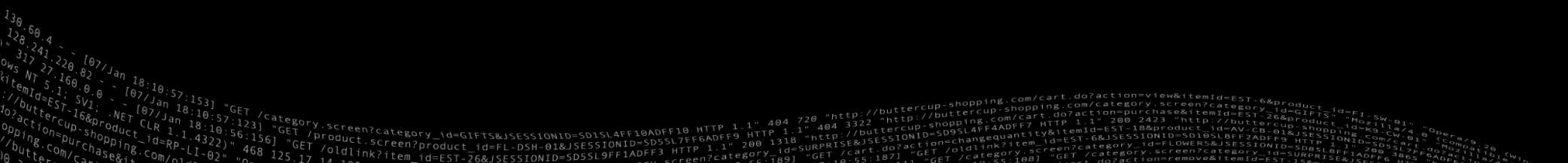
What Are Search Slots

- A very important metric to monitor and maintain are Search Slots.
- Search Slots are the number of concurrent searches that can run on a search head. This number is based on a formula defined by attributes in limits.conf:
  - `max_searches_per_cpu(# of cpus) + base_max_searches = total search slots`
  - Typical Configuration (1 * 16) + 6 = 22 search slots = 22 searches I can run at once.
- Never modify `max_searches_per_cpu`. Adjust `base_max_searches` sparingly.
- Earlier, we mentioned you could tweak the number of concurrent searches the search head will run – this is how. Do so at your own risk.

What Are Search Slots

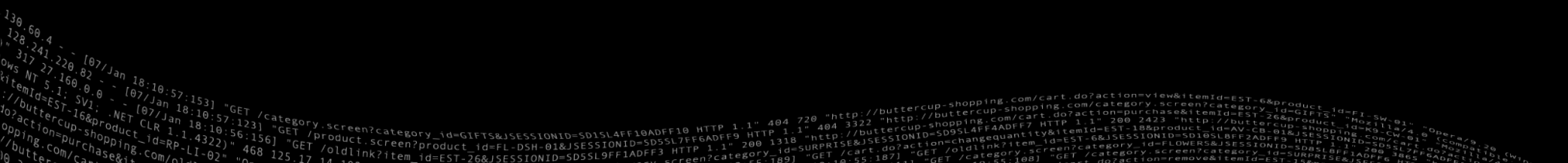
- Remember…
  - Ad-Hoc Searches
  - Correlation Searches
  - Lookup Generator Searches
  - Context Generator Searches
  - Threat Generator Searches
  - Data Model Acceleration

… are all searches and count against your 22 concurrent searches limit!
(if you have a 16 core search head, you may have more)



Also, Note the *Artificial Limits…*

- `max_searches_perc`
  - The maximum number of searches the scheduler can run, as a percentage of the maximum number of concurrent searches
  - Default: 50
- `auto_summary_perc`
  - The maximum number of concurrent searches to be allocated for auto summarization, as a percentage of the concurrent searches that the scheduler can run. Auto summary searches include: Searches which generate the data for the Report Acceleration feature. * Searches which generate the data for **Data Model Acceleration.**
  - Default: 50



Also, Note the *Artificial Limits…*

- Let's do the math…
  - 22 total search slots
  - 50% limit (max_searches_perc) for any scheduled search == 11 concurrent background searches allowed, 11 reserved for users.
  - 50% limit of available background searches (auto_summary_perc) == 5 concurrent report acceleration or data model searches
  - Often an untenable combination – tweak the limits to give ES some breathing room:

```
[scheduler]
auto_summary_perc = 100
max_searches_perc = 75
```

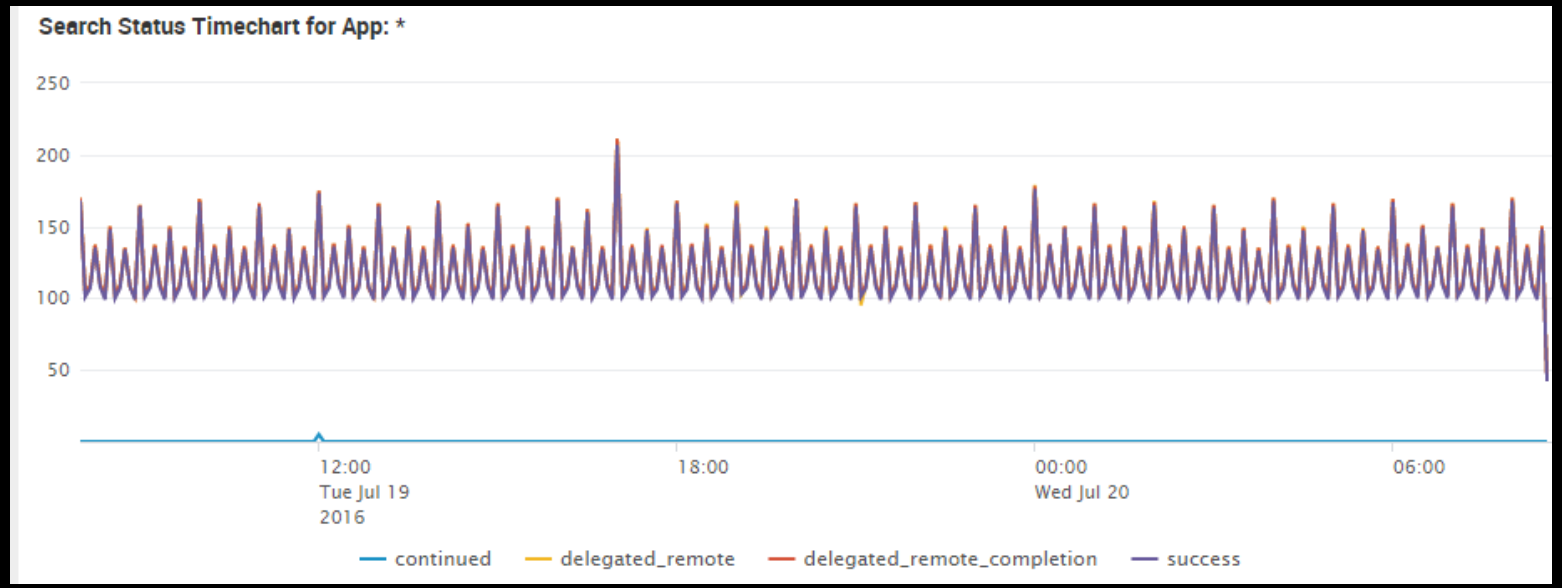
Search Scheduler Tuning

- Problem: Searches usually start at the top of the hour or obvious segments, such as every 10 minutes, 15 minutes, 30 minutes, etc.
  - 60 minutes in an hour, 1440 minutes in a day – We should use them all for our work
- This can be applied to ALL scheduled searches (alerts, DMAs, correlation searches etc.)
- Provided you have enough search slots, it turns out we can get some serious benefit by spreading out scheduled search executions manually.

Search Scheduler Tuning

How much benefit could we possibly get??

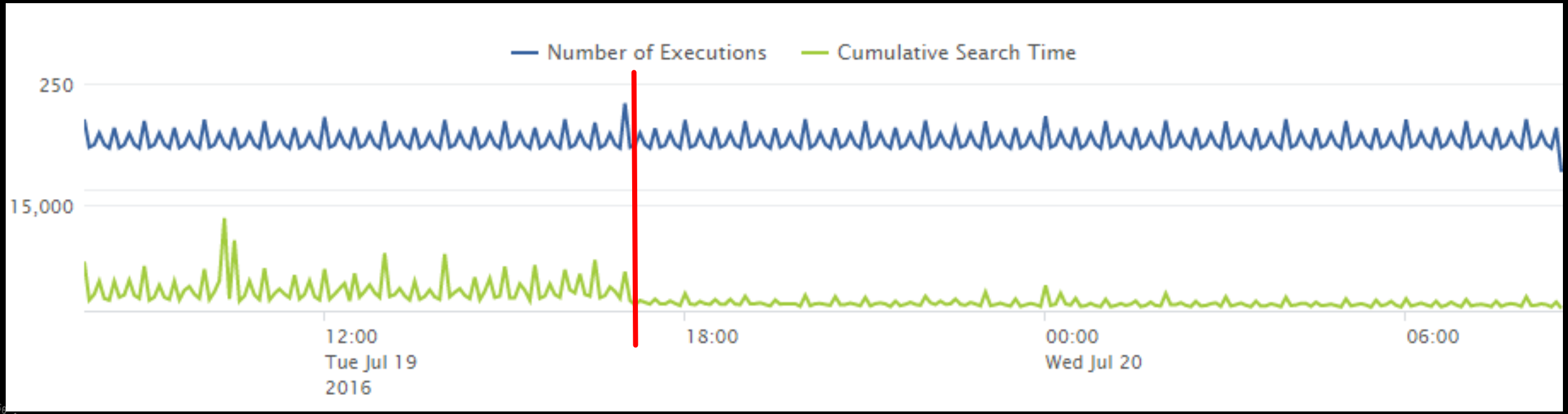
- In this real-world example, each 1 minute “bucket” has 17-18 concurrent scheduled searches running
- Observe around the 5pm mark, and notice relatively uniform search executions



Search Scheduler Tuning

How much benefit could we possibly get??

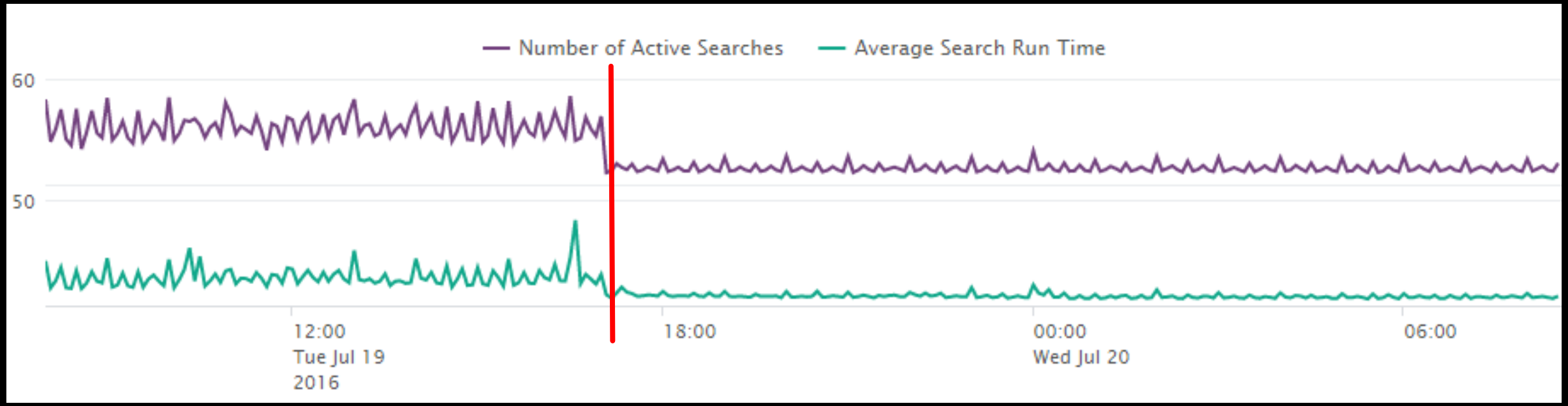
- Search performance though? Not so great until we spread out the searches to run evenly over time
- AGGREGATE (Cumulative) search time… 😳



Search Scheduler Tuning

How much benefit could we possibly get??

- AVERAGE search time… 😯



- The number of Active Searches also is reduced because of the reduction in Average Search Run Time

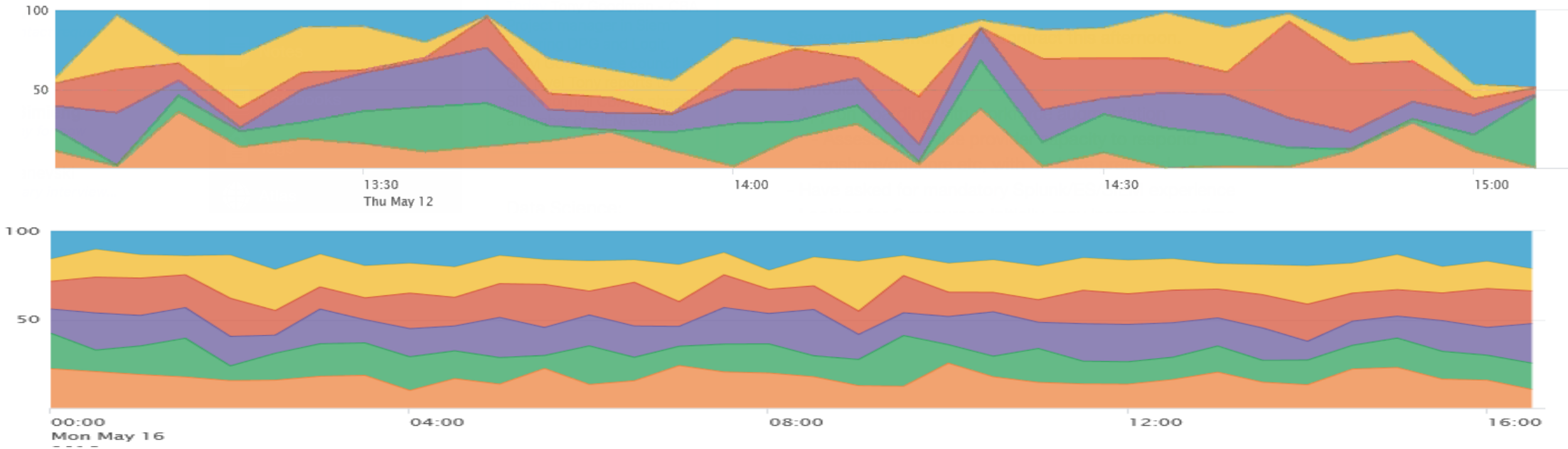
Data Balancing

Use the resources at our disposal

- Even data distribution is crucial in parallel computing
  - We have powerful indexers at our disposal, we should be using them
- Ways to improve data distribution:
  - Enable parallel pipelines on intermediate forwarders (UF and HF)(In server.conf)
  - Route directly from Universal Forwarders to Indexers where possible
  - Consider the following changes to forwarders' outputs.conf:
    - `forceTimebasedAutoLB = true`
    - `autoLBFrequency`
    - `autoLBVolume` (6.6 only)

Data Balancing

Use the resources at our disposal



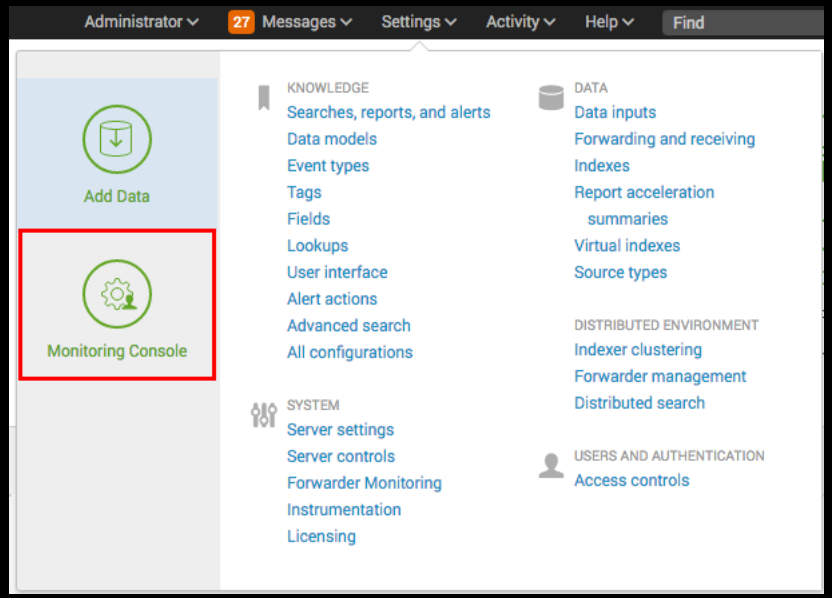
- `| tstats summariesonly=t count WHERE index=* by splunk_server _time | timechart span=5m sum(count) by splunk_server`

splunk> .conf18

Splunk Monitoring Console

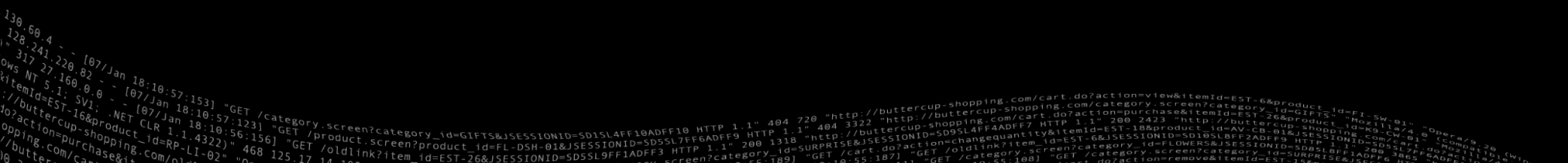
Where can I monitor all of these metrics??

- An invaluable tool for monitoring Splunk Health
- Key Dashboards:
  - Search Activity: Instance
  - Search Usage Statistics: Instance
  - Scheduler Activity: Instance
  - KV Store: Instance



Upgrade Splunk Core!

- Noticeable jumps and improvements at every major release
  - Staying up to date can be tiresome but the types of updates can be worthwhile
  - Do not be shy about updating, particularly Splunk Core
  - Numerous instances where functionality or performance enhancements have improved the ES experience for customers



splunk> .conf18

ES Performance Related Enhancements in Splunk Enterprise by Version

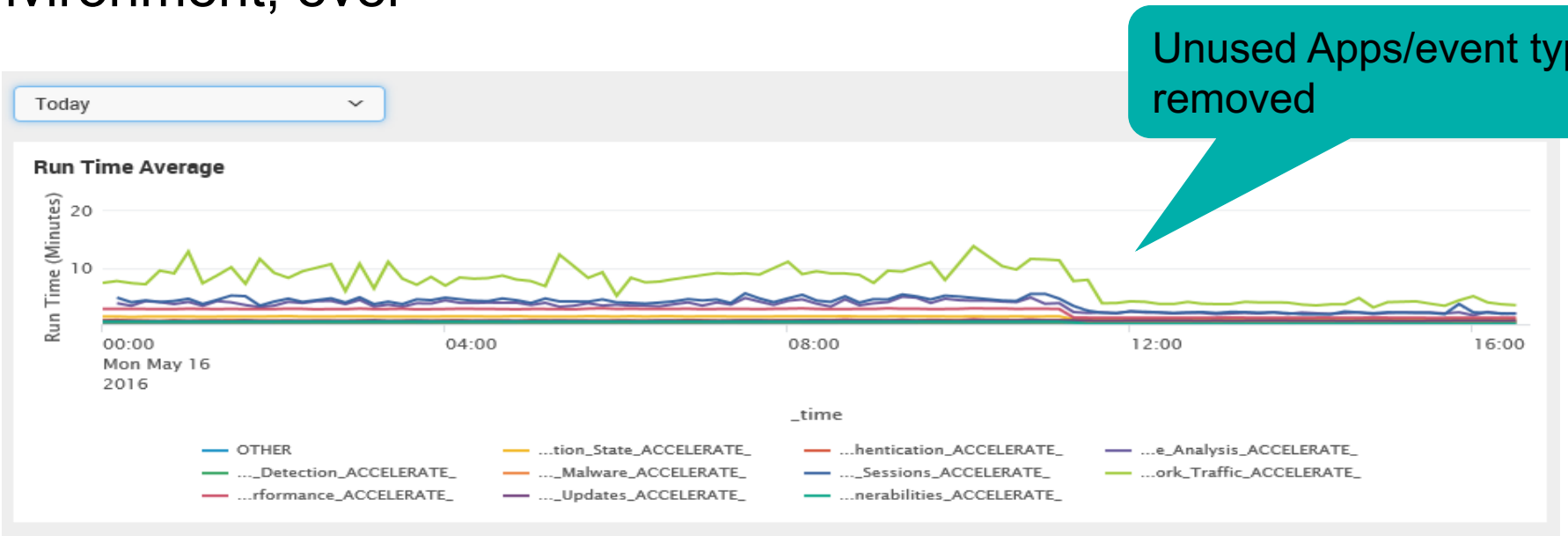
splunk>

- 6.3/6.4
  - Search Parallelization
  - Index Parallelization
  - Distributed Lookups/KV Store
  - Data Model Summary Replication
- 6.6
  - Predicate splitting and search optimization
  - Projection elimination search optimization
  - Volume based data forwarding
- 7.0
  - Faster Data Model Acceleration (3X!)
  - Reduced Summarization Lag
- 7.1
  - UI/UX Refresh Compatible with ES 5.1
  - Minimal Service Disruption for Indexer Cluster and SHC upgrades
  - KVStore Live Backup
  - Improved Data Model Drilldown

splunk> .conf18

Remove Unnecessary TAs

- Splunk ES makes use of tagged eventtypes within applications to generate syntax for searches and data models
- An excessive amount of tags will add to execution time of searches and data model acceleration time
- ADVANCED Tip: Disable eventtypes that will not actually reference any data in your environment, ever



Bundle Size Matters

- Search performance at the SH and IDX tier is greatly impacted by the bundle
  - The larger it is, the greater the impact
  - Large bundles over WAN links (such as indexers in the cloud) simply exacerbate the problem
- Bundle size blowouts can be caused by a number of factors
  - Large lookups
  - ”backups” of configuration changes
  - Core dumps
  - Sneaky files like .git versioning metadata that could be included in automation process
  - Support files used in complex apps (DBX or Tripwire)

Bundle Size Matters

Contents of $SPLUNK_HOME/var/run

```
[root@master run]#
[root@master run]#
[root@master run]#
[root@master run]#
[root@master run]#
[root@master run]#
[root@master run]# pwd
/opt/splunk/var/run
[root@master run]# ls -lah
total 943M
drwx--x--x  5 root root 4.0K Aug  8 21:41 .
drwx--x--x 7 root root 4.0K Oct 25  2016 ..
-rw-------  1 root root  50K Aug  1 03:29 master.splunktools.com-1501554602-1501
-rw-------  1 root root 235M Aug  1 03:29 master.splunktools.com-1501558190.bund
-rw-------  1 root root   45 Aug  1 03:29 master.splunktools.com-1501558190.bund
-rw-------  1 root root 235M Aug  8 21:37 master.splunktools.com-1502228244.bund
-rw-------  1 root root   44 Aug  8 21:37 master.splunktools.com-1502228244.bund
drwx------  7 root root 4.0K Jun 27 10:16 searchpeers
-rw-------  1 root root  299 Jul 11 18:56 serverclass.xml
-rw-------  1 root root 237M Jun 23 16:52 sh.splunktools.com-1498236732.bundle
-rw-------  1 root root   44 Jun 23 16:52 sh.splunktools.com-1498236732.bundle.i
-rw-------  1 root root 237M Jun 23 16:56 sh.splunktools.com-1498236990.bundle
-rw-------  1 root root   45 Jun 23 16:56 sh.splunktools.com-1498236990.bundle.i
drwx--x--x 13 root root 4.0K Aug  9 21:08 splunk
drwx------  3 root root 4.0K Oct 15  2015 tmp
[root@master run]#
```

splunk> .conf18

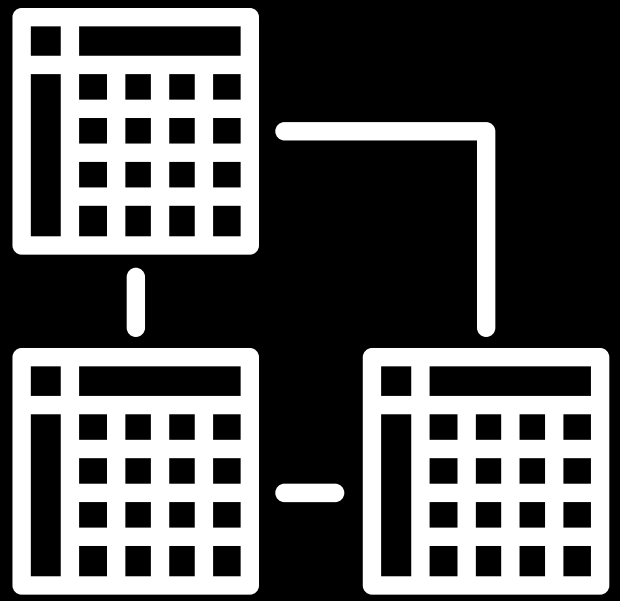
Bundle Size Matters

- Review Search bundle size and techniques to reduce the total size
  - `$SPLUNK_HOME/var/run` with .bundle extentions (but actually tar files)
  - untar and du . -h
- `distsearch.conf`
  - `[replicationBlacklist]`
  - `[replicationWhitelist]`
  - `[replicationSettings:refineConf stanza]`

Enterprise Security Optimizations

splunk> .conf18

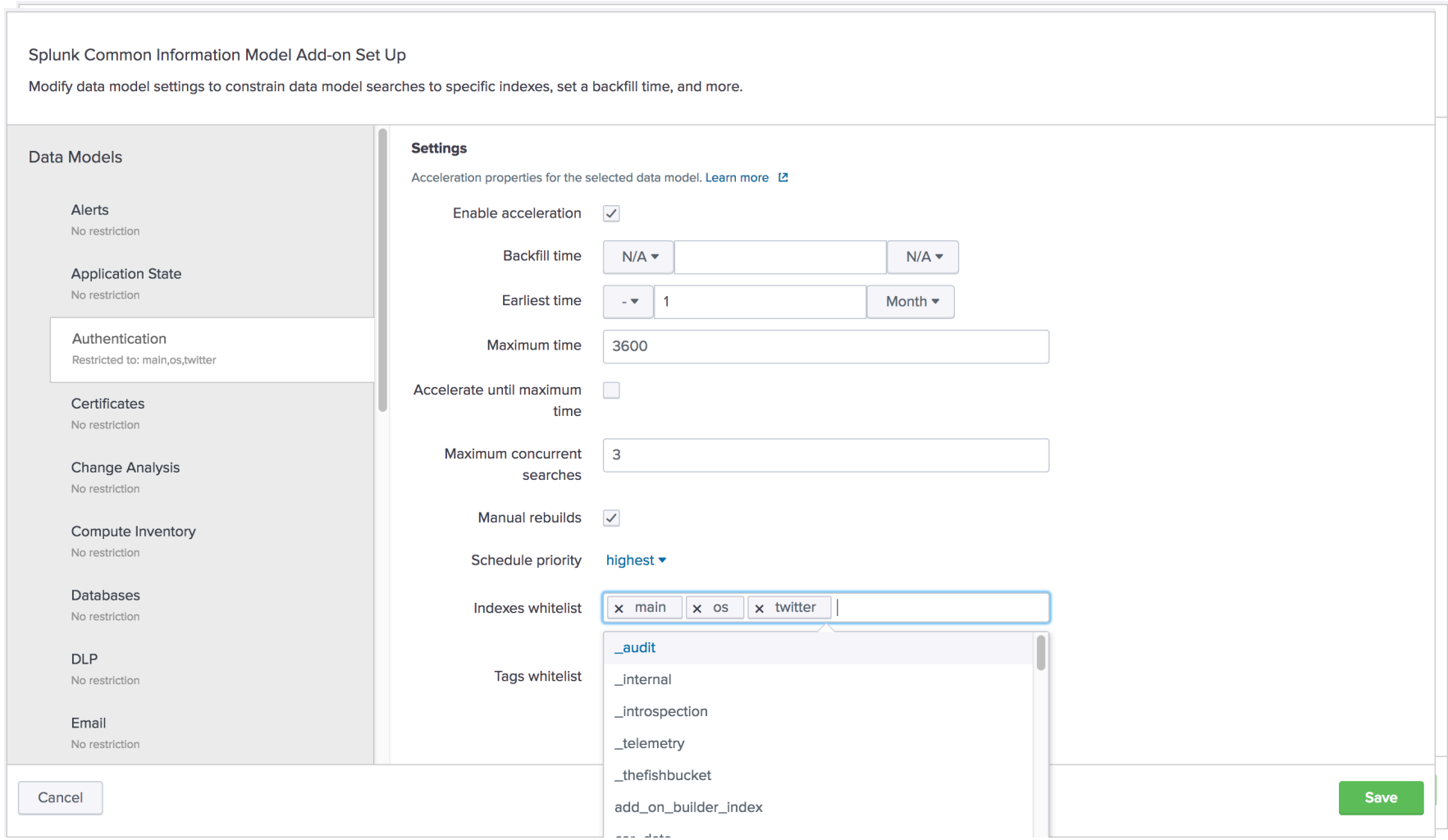
Data Model Tuning



- ES utilizes several Data Models from the Splunk Common Information Model.
- Data Model Acceleration summarizes all events in scope down to key value pairs of specific fields, as defined in the Data Model.
- By default, Splunk **searches all indexes for data** relevant to a particular data model, and is normally filtered by special tags.
- Data Models can be **tuned to specific indexes for each data model**, resulting in better efficiency in summarizing the key value pairs needed for the Data Model.

Data Model Tuning

Use the Configure > CIM Setup menu in ES

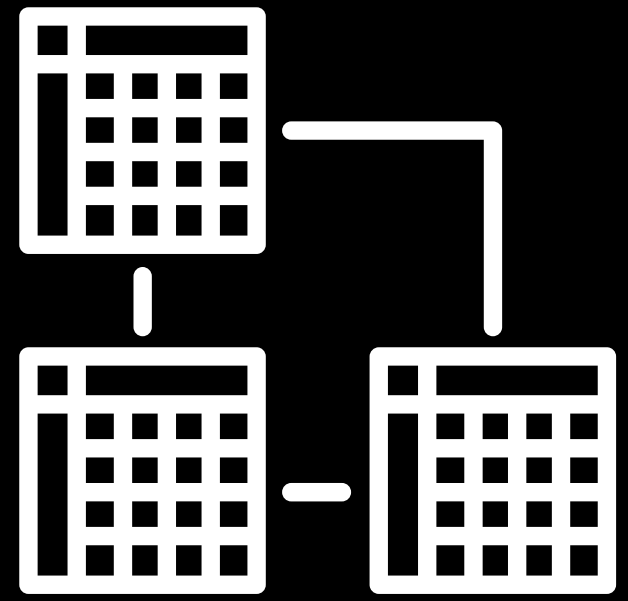


splunk> .conf18

datamodels.conf acceleration.backfill_time

Limit the impact of Data Model Acceleration

- **Data Model Activity consumes search slots that you may need for ad-hoc search**
- Sometimes, its better to not backfill old data model summaries all at once.
- You can limit how far back Splunk attempts to summarize datamodels with backfill_time in `datamodels.conf`



datamodels.conf acceleration.backfill_time

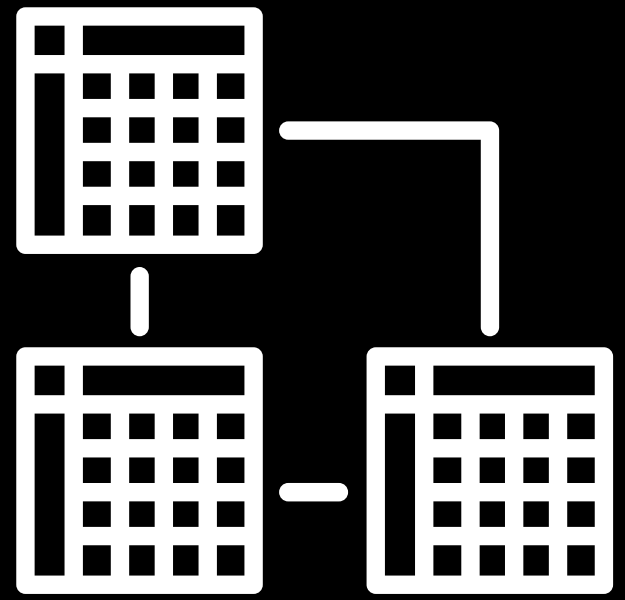
Limit the impact of Data Model Acceleration

- acceleration.backfill_time = <relative-time-str>
- * ADVANCED: Specifies how far back in time the Splunk software should create its column stores. * ONLY set this parameter if you want to backfill less data than the retention period set by 'acceleration.earliest_time'. You may want to use this parameter to limit your time window for column store creation in a large environment where initial creation of a large set of column stores is an expensive operation.
- * WARNING: Do not set 'acceleration.backfill_time' to a narrow time window. If one of your indexers is down for a period longer than this backfill time, you may miss accelerating a window of your incoming data.
- * MUST be set to a more recent time than 'acceleration.earliest_time'. For example, if you set 'acceleration.earliest_time' to '-1y' to retain your column stores for a one year window, you could set 'acceleration.backfill_time' to '-20d' to create column stores that only cover the last 20 days. However, you cannot set 'acceleration.backfill_time' to '-2y', because that goes farther back in time than the 'acceleration.earliest_time' setting of '-1y'. * Defaults to empty string (unset).
- When 'acceleration.backfill_time' is unset, the Splunk software always backfills fully to 'acceleration.earliest_time'.

datamodels.conf acceleration.backfill_time

Limit the impact of Data Model Acceleration

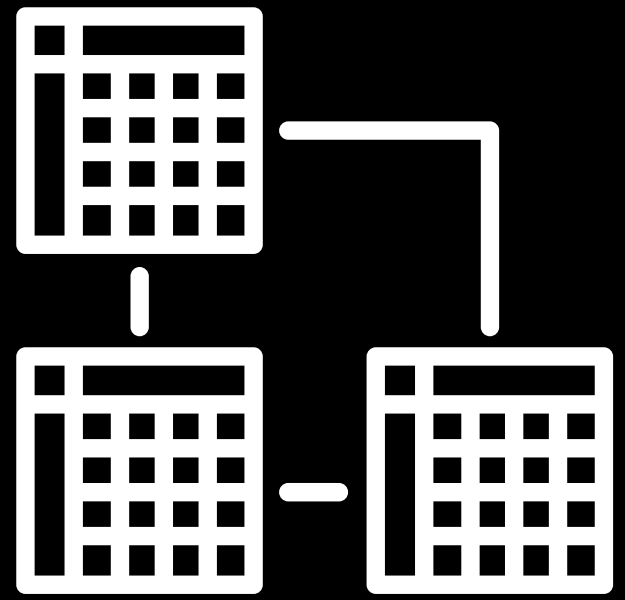
- **When is backfill_time relevant?**
- **Almost Never.** Only when you need to artificially “slow down” data model acceleration because you do not have the available CPU and search slots to do it the normal way.



Data Model Acceleration Summary Replication

A very relevant Data Model Acceleration feature

- **Problem:** If an indexer with a summary goes down (or is restarted), bucket primaries move to another searchable copy, and searches will not have access to the summaries (until they get regenerated), thereby searches run slow.
- **Answer**: Replicate summaries so that they exist with all searchable copies.
- To turn on summary replication, make `summary_replication=true` under clustering stanza in server.conf on cluster master. By default summary replication is turned off.
- Config changes are reloadable (i.e. does not require a splunk restart)



Assets and Identities Table Lookup Performance

- ES carries along with it a number of lookup tables, two of which could become very large.
- The process of “indexing” large lookups could slow down ES
- If you see a long period of time in Job Inspector for search.command.lookups, preventing indexing of large lookups may provide a performance improvement.
- limits.conf tweak max_memtable_bytes slightly larger than your assets/identities
  - `max_memtable_bytes = <integer>`
  - `* Maximum size, in bytes, of static lookup file to use an in-memory index for.`
    `* Lookup files with size above max_memtable_bytes will be indexed on disk`
  - `* A large value results in loading large lookup files in memory leading to bigger process memory footprint.`
  - `* Caution must be exercised when setting this parameter to arbitrarily high values!`
  - `* Default: 10000000 (10MB)`

splunk> .conf18

Assets and Identities Table Lookup Performance

search.command.lookups

	2.69	command.search.filter	64	-	-
	1.51	command.search.calcfields	64	193,555	193,555
	0.54	command.search.fieldalias	64	193,555	193,555
	0.00	command.search.index.usec_1_8	363,541	-	-
	0.00	command.search.index.usec_4096_32768	11	-	-
	0.00	command.search.index.usec_512_4096	6,808	-	-
	0.00	command.search.index.usec_64_512	22,853	-	-
	0.00	command.search.index.usec_8_64	31,369	-	-
	59.73	command.search.rawdata	64	-	-
	5.22	command.search.kv	64	-	-
	4.51	command.search.lookups	64	193,555	193,555
	0.15	command.search.typer	64	2,460	2,460
	0.06	command.search.tags	64	2,460	2,460
	0.02	command.search.summary	71	-	-
	0.00	dispatch.check_disk_usage	2	-	-
	0.02	dispatch.createdSearchResultInfrastructure	1	-	-
	0.17	dispatch.evaluate	1	-	-
	0.17	dispatch.evaluate.search	1	-	-
	18.98	dispatch.fetch	72	-	-
	20.30	dispatch.finalizeRemoteTimeline	1	-	-
	0.07	dispatch.parserThread	71	-	-

splunk> .conf18

Search Optimization Techniques

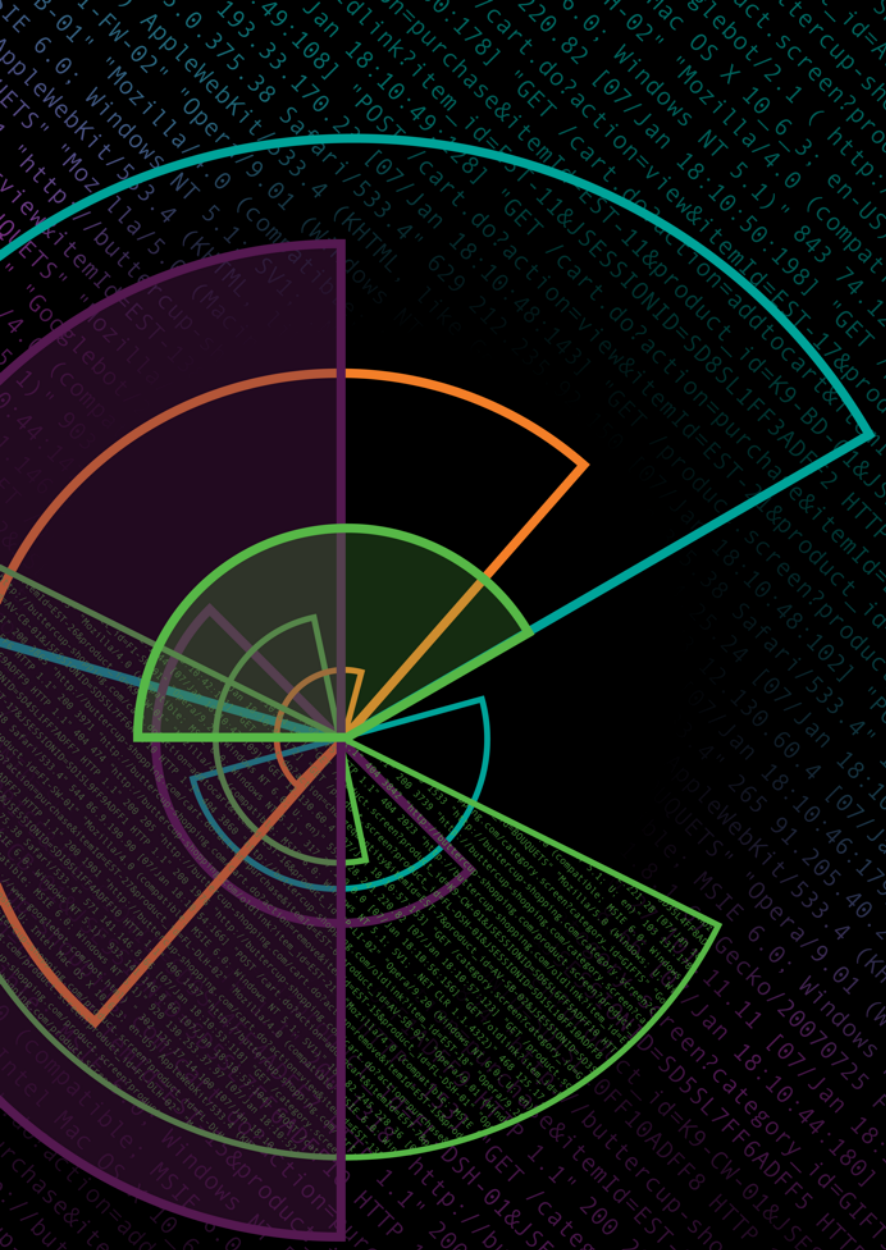
- **What Correlation Searches should I run?**
  - (answer: not all of them. Quality > quantity. 50 notable events > 60,000 notable events.)
- **Optimizing Slow Running ES Panels**
- **Profiling and Resolving Slow Correlation Search Performance**

Health Check List

- Uninstall the unnecessary add-ons.
- Tune the artificial limits.
- Profile search slots and skipped searches using Monitoring Console.
- Deal with long running searches by optimizing.
- Consider rescheduling your searches on a more even schedule if you find skipped searches.
- Balance your data.
- Upgrade Splunk.
- Watch your bundle size.
- Tune your data models (indexes, and perhaps backfill range).
- If necessary, make your lookups smaller.

Key Takeaways

- Getting more “juice” out of Enterprise Security is really about Splunk optimization.
- Understanding the under-the-hood inner workings make ES easier to tune and optimize.
- There are a few easy knobs you can turn that drastically impact performance – make one change at a time and test!



Q&A

splunk> .conf18

Thank You

Don't forget to rate this session in the .conf18 mobile app