Squeezing all the Juice out of Splunk Enterprise Security

Marquis Montgomery, CISSP | Sr. Staff Security Consultant, Splunk
Jae Jung | Professional Services Consultant, Splunk

September 23 – 25, 2017 | Washington, DC
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, Splunk>, Listen to Your Data, The Engine for Machine Data, Splunk Cloud, Splunk Light and SPL are trademarks and registered trademarks of Splunk Inc. in the United States and other countries. All other brand names, product names, or trademarks belong to their respective owners. © 2017 Splunk Inc. All rights reserved.
Introductions and Agenda

Who are these guys, anyway?
Welcome!

Marquis Montgomery, CISSP
marquis@splunk.com / @trademarq
Sr. Staff Security Consultant, Splunk
▶ 4+ years in Splunk PS
▶ Former customer, Manager of Corporate Security at MSSP
▶ Leads Enterprise Security Field Enablement

Jae Jung, Splunk Certified Consultant
jajung@splunk.com / @jaestwitteraccount
Professional Services Consultant, Splunk
▶ 2+ years with Splunk Professional Services
▶ Review, remediation and re-platforming of the largest ES client in Australia
▶ High involvement with Splunk customers across the Australian FinServ market
Agenda
What will we be talking about today?

ES Under-the-Hood
Checking out the engine

ES Specific Optimizations
Enhancements specific to the ES application

Core Splunk Optimizations
Splunk Enterprise Platform Enhancements

Key Takeaways and Q&A
Tying it all together
To tune the engine, you need to understand the engine
Splunk Enterprise Security is a complex group of Splunk apps that work together, but at its core, it consists of the following components:

- **LOTS of Dashboards**
- **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
Key Processes in Enterprise Security

Where can I tune for better performance?

- Data Models
- Scheduled Searches
- Lookup Tables
- KV Stores
How ES Works

Raw Event is Indexed
Data is generated, forwarded and indexed in Splunk

Data is now available for ES
tlstats queries and dashboards now see data

Data Model Summary Search Runs
CIM DM normalization is applied, CIM DM field key/value pairs are stored in DM TSIDX

ES Background Searches Process Data
Correlation Searches, trackers and threat intelligence search data models

ES learns about threats and anomalies
ES writes these results to notable events, summary indexing and data models
Places We Can Squeeze More Juice

- **Raw Event is Indexed**
  - Data is generated, forwarded and indexed in Splunk

- **Data is now available for ES**
  - tstats queries and dashboards now see data

- **Data Model Summary Search Runs**
  - CIM DM normalization is applied, CIM DM field key/value pairs are stored in DM TSIDX

- **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
Okay. How!?

Get to the goods!!!
Core Splunk Optimizations

The Machine Data Platform
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.
What Are Search Slots

Remember...

- 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!
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
```
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
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
```
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
Performance Related Enhancements in Splunk Enterprise by version

6.3/6.4
- Search Parallelization
- Index Parallelization
- Distributed Lookups/KV Store
- Data Model Summary Replication

6.5
- Improved UF Load balancing
- Indexer Cluster rebalancing

6.6
- Predicate splitting and search optimization
- Projection elimination search optimization
- Volume based data forwarding
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.

Unused Apps/event types removed
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

```bash
[root@master run]# ls -lah /opt/splunk/var/run
total 943M
```

```bash
drwxr-x--- 5 root root 4.0K Aug 8 21:41 .
```

```bash
drwxr-x-x- 7 root root 4.0K Oct 25 2016 ..
```

```bash
-rw------- 1 root root 50K Aug 1 03:29 master.splunktools.com-1501554602-1501554602
-rw------- 1 root root 235M Aug 1 03:29 master.splunktools.com-1501558190-1501558190
-rw------- 1 root root 45 Aug 1 03:29 master.splunktools.com-1501558190-1501558190
-rw------- 1 root root 235M Aug 8 21:37 master.splunktools.com-1502228244-1502228244
-rw------- 1 root root 44 Aug 8 21:37 master.splunktools.com-1502228244-1502228244
```

```
-rw------- 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
-rw------- 1 root root 237M Jun 23 16:52 sh.splunktools.com-1498236990.bundle
-rw------- 1 root root 45 Jun 23 16:52 sh.splunktools.com-1498236990.bundle
```

```
-rw------- 13 root root 4.0K Aug 9 21:08 splunk
drwxr-x--- 3 root root 4.0K Oct 15 2015 tmp
```

```
[root@master run]# ```
Bundle Size Matters

- Review Search bundle size and techniques to reduce the total size
  - `$SPLUNK_HOME/var/run` with `.bundle` extensions (but actually tar files)
  - `untar` and `du` . -h
- `distsearch.conf`
  - `[replicationBlacklist]`
  - `[replicationWhitelist]`
  - `[replicationSettings:refineConf stanza]`
ES Optimizations
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 Common Information Model Add-on Setup

By default, a datamodel will search across all indexes. Use the configuration panel below to constrain data model searches to specific indexes.

#### Data Models

**Alerts**
- No restriction

**Application State**
- No restriction

**Authentication**
- Restricted to: main, risk, twitter

**Certificates**
- No restriction

**Change Analysis**
- No restriction

**Compute Inventory**
- No restriction

**Databases**
- No restriction

**DLP**
- No restriction

**Email**
- No restriction

---

#### Indexes

<table>
<thead>
<tr>
<th>Name</th>
<th>App</th>
<th>Current Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>main</td>
<td>org_allindexes</td>
<td>5,081 MB</td>
</tr>
<tr>
<td>risk</td>
<td>org_allindexes</td>
<td>3 MB</td>
</tr>
<tr>
<td>twitter</td>
<td>org_allindexes</td>
<td>1 MB</td>
</tr>
<tr>
<td>_audit</td>
<td>org_allindexes</td>
<td>520 MB</td>
</tr>
<tr>
<td>_internal</td>
<td>org_allindexes</td>
<td>2,435 MB</td>
</tr>
<tr>
<td>_introspection</td>
<td>system</td>
<td>1,161 MB</td>
</tr>
<tr>
<td>_telemetry</td>
<td>system</td>
<td>1 MB</td>
</tr>
<tr>
<td>_theishbucket</td>
<td>org_allindexes</td>
<td>1 MB</td>
</tr>
<tr>
<td>add_on_builder_index</td>
<td>splunk_app_addon-builder</td>
<td>1 MB</td>
</tr>
<tr>
<td>cim_modactions</td>
<td>org_allindexes</td>
<td>3 MB</td>
</tr>
<tr>
<td>cim_summary</td>
<td>org_allindexes</td>
<td>1 MB</td>
</tr>
<tr>
<td>endpoint_summary</td>
<td>org_allindexes</td>
<td>1 MB</td>
</tr>
<tr>
<td>history</td>
<td>org_allindexes</td>
<td>1 MB</td>
</tr>
<tr>
<td>ioc</td>
<td>org_allindexes</td>
<td>1 MB</td>
</tr>
</tbody>
</table>
Data Model Activity consumes search slots that you may need for ad hoc search.

Sometimes, it's 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.
**datamodels.conf acceleration.backfill_time**

- `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.
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)
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)
<table>
<thead>
<tr>
<th>Asset/Identity</th>
<th>Table Lookup</th>
<th>Performance</th>
</tr>
</thead>
<tbody>
<tr>
<td>command.search.filter</td>
<td>64</td>
<td>-</td>
</tr>
<tr>
<td>command.search.caiched</td>
<td>64</td>
<td>193,555</td>
</tr>
<tr>
<td>command.search.fieldalias</td>
<td>64</td>
<td>193,555</td>
</tr>
<tr>
<td>command.search.index.usec_1_9</td>
<td>0.00</td>
<td>362,541</td>
</tr>
<tr>
<td>command.search.index.usec_4006_32768</td>
<td>0.00</td>
<td>11</td>
</tr>
<tr>
<td>command.search.index.usec_512_4096</td>
<td>0.00</td>
<td>6,808</td>
</tr>
<tr>
<td>command.search.index.usec_64_512</td>
<td>0.00</td>
<td>22,853</td>
</tr>
<tr>
<td>command.search.index.usec_8_64</td>
<td>0.00</td>
<td>31,369</td>
</tr>
<tr>
<td>command.search.rawdata</td>
<td>59.73</td>
<td>64</td>
</tr>
<tr>
<td>command.search.kv</td>
<td>5.22</td>
<td>64</td>
</tr>
<tr>
<td>command.search.lookups</td>
<td>4.51</td>
<td>64</td>
</tr>
<tr>
<td>command.search.type</td>
<td>0.15</td>
<td>64</td>
</tr>
<tr>
<td>command.search.tags</td>
<td>0.05</td>
<td>64</td>
</tr>
<tr>
<td>command.search.summary</td>
<td>0.02</td>
<td>71</td>
</tr>
<tr>
<td>dispatch.check_disk_usage</td>
<td>0.00</td>
<td>2</td>
</tr>
<tr>
<td>dispatch.createdSearchResultInfrastructure</td>
<td>0.02</td>
<td>1</td>
</tr>
<tr>
<td>dispatch.evaluate</td>
<td>0.17</td>
<td>1</td>
</tr>
<tr>
<td>dispatch.evaluate.search</td>
<td>0.17</td>
<td>1</td>
</tr>
<tr>
<td>dispatch.fetch</td>
<td>10.99</td>
<td></td>
</tr>
<tr>
<td>dispatch.finalizeRemoteTimeline</td>
<td>20.30</td>
<td>1</td>
</tr>
<tr>
<td>dispatch.parserThread</td>
<td>0.07</td>
<td>71</td>
</tr>
</tbody>
</table>
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
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
Making Machine Data Accessible, Usable And Valuable To Everyone.
Thank You

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

splunk> .conf2017