Monitoring Docker Containers with Splunk

Marc Chéné | Product Manager

Sept 27, 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.
Who I am

Marc Chéné

• Product Manager, Engineer, APMer
• Dad/ super fan/ coach to 3, loves skiing, golfing, music and a good drink

@marcchene

https://www.linkedin.com/in/marcchene

slack id: mchene
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.
Agenda

- Monitoring Options
- Analytical Insight – Tips & Tricks
- The World of Metrics
- (coming soon!) Docker Moby - v2 logging Plugin
Monitoring Options

logs, events and perf stats
Splunk and Docker – At A Glance

Visibility in your Container Environments

- Splunk Logging Driver for Docker
  - Built into Docker – no extra software required
  - Insight into container and apps running in containers

- Docker Universal Control Plane
  - Insight into administration, changes, and composition

Monitoring for your Cloud Environments

- Deep Visibility in Amazon Web Services (AWS) and in EC2 Container Services (ECS)
  - Splunk provides support for Google Cloud Platform (GCP)

Delivering Splunk as Containers

- Make getting Splunk as easy as a single Docker pull command from the Docker Hub/Store

- Forwarders and Splunk Enterprise pre-configured to collect machine data from Container Host and Docker API
Splunk Collection Options for Docker

- Docker Native Logging – Splunk logging driver, Syslog, JSON, AWS CloudWatch, etc.
- Forwarders – App Logs, Syslog UDP forwarding, Performance, etc.
- Logging libraries in .NET, Java and node.js
- Custom (e.g., Kafka with HTTP Event Collector)
- Cloud – AWS, GCP, Azure

Use the option that is right for you!
Secure—supports TLS/SSL and tokens

Simple – config-based setup and collect data

Buffering

Scale – Based on HTTP Data Collector Based on Splunk HTTP

Configurable - Supports container labels, environment variables

docker run --log-driver=splunk \
  --log-opt splunk-token=176FCEBF-4CF5-4E8D-91BC-703796522020 \
  --log-opt splunk-url=https://splunkhost:8088 \
  --log-opt splunk-capath=/path/to/cert/cacert.pem \
  --log-opt splunk-cname=SplunkServerDefaultCert \
  --log-opt tag="{(Name)}/(FullID)" \
  --log-opt labels=location \
  --log-opt env=TEST \
  --env "TEST=false" \
  --label location=west \
  your/application
Log Streaming - Splunk Logging Driver for Docker v1.13+

- Skip verification for the valid splunk url
- Raw data collection from the native log driver
- Embedded json format support
- Performance Improvements
Splunk container images available

- Splunk Enterprise 6.6.3
- Splunk Universal Forwarder 6.6.3

Includes configuration and Docker Add-On for container monitoring out-of-the-box

docker pull store/splunk/enterprise
docker pull store/splunk/universalforwarder:6.6.3
Deep Dive: What’s Do We Monitor?

- Docker Hub: [https://hub.docker.com/r/splunk/universalforwarder/](https://hub.docker.com/r/splunk/universalforwarder/) tag: 6.5.3-monitor
- GitHub: [https://github.com/splunk/docker-itmonitoring](https://github.com/splunk/docker-itmonitoring)
  - Docker logs ([ta-dockerlogs_fileinput](#)) under “/host/containers/*/”
    - [a-f0-9]+-json.log
    - config.v2.json
    - hostconfig.json
    - hostname
    - hosts
    - resolv.conf
  - Docker stats ([ta-dockerstats](#))
  - UCP logs ([ta-ucplogs-sysloginput](#))
Demo Monitoring!
Analytical Insight – Tips & Tricks
Analytical Insight – Tips & Tricks

▶ Sample Docker Compose file
▶ Correlations
  • Docker SWARM mode
  • Amazon Web Services (AWS)
▶ Log Options
  • --log-opt tag="{{.Name}}/{{.FullID}}"
**Terminology - What is a Measurement?**

Treated natively as metrics, not log files

- **Time**
- **Metric Name**
  - `system.cpu.idle`
- **Measure**
  - *numeric data point, different types such as count, gauge, timing, sample, etc*
- **Dimensions**
  - **Host** (10.1.1.100, web01.splunk.com)
  - **Region** (e.g., us-east-1, us-west-1, us-west-2, us-central1)
  - **Instance Types** (e.g., t2.medium, t2.large, m3.large)
“Splunk provides **ONE** platform to analyze and investigate across both Logs and Metrics
## Metrics Data Shape

<table>
<thead>
<tr>
<th>Field</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>_time</td>
<td>Y</td>
<td>Microseconds since epoch</td>
</tr>
<tr>
<td>metric_name</td>
<td>Y</td>
<td>metric name</td>
</tr>
<tr>
<td>_value</td>
<td>Y</td>
<td>Value of the metric (numeric values only)</td>
</tr>
<tr>
<td>_dims</td>
<td>Y</td>
<td>Dimension names</td>
</tr>
<tr>
<td>host</td>
<td>Y</td>
<td>Origination Host</td>
</tr>
<tr>
<td>index</td>
<td>Y</td>
<td>Index to store the data</td>
</tr>
<tr>
<td>metric_type</td>
<td>N</td>
<td>Counter</td>
</tr>
<tr>
<td>source</td>
<td>N</td>
<td>the source of the data point, <a href="https://docs.splunk.com/Splexicon:Source">https://docs.splunk.com/Splexicon:Source</a></td>
</tr>
<tr>
<td>sourcetype</td>
<td>Y</td>
<td>Used for defining groupings of metrics and defining input time rules</td>
</tr>
<tr>
<td><code>&lt;fieldA&gt;..&lt;fieldZ&gt;</code></td>
<td>N</td>
<td>Arbitrary number of dimensions</td>
</tr>
</tbody>
</table>
Key Features

**Metric Store**
Ability to ingest and store metric measurements at scale

**mstats**
`tstats` equivalent to query time series from metrics indexes

**Metrics Catalog**
REST APIs to query lists of ingested metrics and dimensions
Metrics Store

- Based on splunkd
- Dedicated Indexes for Metrics and Logs
- Full part of the platform
  - RBAC
  - Clustering
  - Index Management
  - Central Administration
- Optimized for fast time series queries and ingestion of metrics at scale
**SPL: mstats**

- New SPL command
- Syntax
  ```splunk
  | mstats <stats-fun>...
  [WHERE index=<mymetricindex> metric_name=<metricname>...] [BY <dimension-list> [span=<timespan>] ]
  ```
- Sample
  - Stats:
    ```splunk
    | mstats avg(_value), count(_value)
    WHERE metric_name="*.cpu.percent" by metric_name span=30s
    ```
  - Time Series Visualization:
    ```splunk
    | mstats avg(_value), count(_value)
    WHERE metric_name="*.cpu.percent" by metric_name span=30s
    | timechart first(avg(_value)) as "avg" span=30s by metric_name
    ```
**Metrics Catalog: Discovery & Search**

- **GET /services/catalog/metricstore/metrics**
  - List all metric names
    ```
curl -k -u admin/pass
https://localhost:8089/services/catalog/metricstore/metrics
```
  - List all metric names that apply to a given dimension name "dc"
    ```
curl -k -u admin/pass
https://localhost:8089/services/catalog/metricstore/metrics?dimension=dc
```

- **GET /services/catalog/metricstore/dimensions**
  - List all dimension names
    ```
curl -k -u admin/pass
https://localhost:8089/services/catalog/metricstore/dimensions
```
  - List all the dimension names that are compatible with a given metric name "mem.free":
    ```
curl -k -u admin/pass
https://localhost:8089/services/catalog/metricstore/dimensions?metric=mem.free
```
  - List all the dimension values for a given dimension name "dc"
    ```
curl -k -u admin/pass
https://localhost:8089/services/catalog/metricstore/dimensions/dc/values
```
  - List all the dimension values for a given dimension name "dc" and metric name "mem.free"
    ```
curl -k -u admin/pass
```
GDI - Metric Ingestion Protocol: Collectd – Write HTTP plugin

- Collectd, https://collectd.org - ~100 frontend plugins
- Scheduled push interval: 30secs
- # of metrics collected: ~350 (~1M measurements per day per server)
- Enabled plugins configurations, collectd.conf

1. csv
2. cpu
3. df
4. disk
5. Interface
6. irq
7. load
8. Logfile
9. memory
10. Network
11. processes
12. protocols
13. Syslog
14. swap
15. tcpconns
16. thermal
17. ptime
GDI: collectd write_http plugin

- Sample write_http event
  ```json
  {"values":[98.9363841194414],"dstypes": ["derive"], "dsnames": ["value"], "time": 1474401106.556, "interval": 10.000, "host": "C5819124-66AE-4B28-8E13-914C3961E46C", "plugin": "cpu", "plugin_instance": "0", "type": "cpu", "type_instance": "idle"}
  ```

- Sample Result
  - metric_name=cpu.idle.value
  - _value=98.9363841194414
  - Host=C5819124-66AE-4B28-8E13-914C3961E46C
GDI Deployment Options: Collectd & HEC

Server Farm

Cloud

Splunk Collectd Package

write_http plugin

HTTPS - HEC

Splunk Indexing Tier
cAdvisor

► Provides container users an understanding of the resource usage and performance characteristics of their running containers

► It is a running daemon that collects, aggregates, processes, and exports information about running containers
DEMO Docker Metrics!
Docker Moby - v2 logging Plugin

- Docker Hub: [https://github.com/splunk/docker-logging-plugin](https://github.com/splunk/docker-logging-plugin)
- Running the logging plugin

```
docker run --log-driver=splunk-log-driver:next \
  --log-opt splunk-token=176FCEBF-4CF5-4EDF-91BC-703796522D20 \
  --log-opt splunk-url=https://splunkhost:8088 \
  --log-opt splunk-capath=/path/to/cert/cacert.pem \
  --log-opt splunk-caname=SplunkServerDefaultCert \
  --log-opt tag='{{.Name}}/{{.FullID}}' \
  --log-opt labels=location \
  --log-opt env=TEST \
  --env "TEST=false" \
  --label location=west your/application
```
Key Takeaways

1. Docker Monitoring – You have options!
2. Analytical Driven Insight
3. Metrics
4. Docker v2 logging API plugin
Thank You

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