

Real Time Monitoring Of A Cloud Based Micro Service Architecture Using Splunkcloud And The HTTP Eventcollector

Mike Sclementi

Experian Consumer Services, Splunk Inc.

Matt Poland

Experian Consumer Services, Splunk Inc.

.conf2016

splunk>

Disclaimer

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 the 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.

It should also be noted that the views expressed in this presentation are solely those of the author in his private capacity and do not in any way represent the views of ConsumerInfo.com, Inc. (aka: Experian Consumer Services), any other entity of Experian, or its Affiliates.

All logos used in this presentation are property of their respective companies.

This talk will go through the architecture and the lessons learned while deploying SplunkCloud using the AWS App (S3), Kinesis, Lambda functions, and the HTTP Event Collector. It will also show how we went from 15 minutes of latency, on our production dashboards, to sub-5 seconds of latency sending the logs directly from Kinesis, via Lambda, to the HTTP Event Collector.

.conf2016

splunk>

Agenda

- Who Are We?
- The Audience...
- Why Present @ .Conf 2016?
- Why Splunk Cloud?
- Cloud Services Architecture
- The S3 Connector
- .Conf 2015 – The Great Shake Off
- The HTTP Event Collector
- Lessons Learned & Tuning
- Q & A

Who Is Mike?

- Senior Systems Engineer
 - Experian Consumer Services
- Education
 - BA @ CSULB
- Background
 - IT Systems Administration/Engineering for 20+ years
 - › Highly Scalable Infrastructure Deployments & Disaster Recovery
 - › Large Scale VMware & Symantec (Veritas) NetBackup Environments
 - › Application Deployments, Systems Management, Active Directory, etc.
 - Monitoring Systems 1.5+ years
- Splunk Customer
 - User for 7 years
 - Admin for 1.5 years (Splunk 6.1, 6.3)
- Hobby
 - Mountain Bike Racing



Who Is Matt?

- Senior Sales Engineer @ Splunk (Southwest Major Accounts)
- Education
 - BS in Computer Science – University of Colorado, Boulder
- Background
 - Sales Engineer for SIEM tools for over 12 years as the SIEM market
 - Working with customers to monitor and secure cloud based applications
- Splunk Speaker
 - Presented at .conf 2015 on getting data from AWS into Splunk
- Hobbies
 - Waterskiing
 - Dirt Biking
 - Basically Anything Outdoors

splunk>

presenter
.conf2015



About The Audience

Let us get to know you...

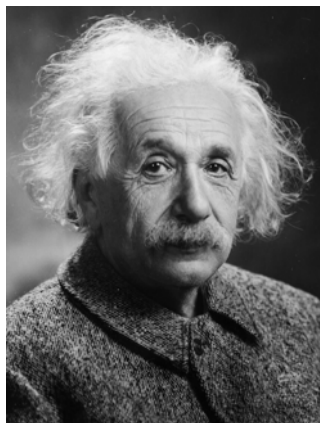
- User?
- Power User?
- Admin?
- Groupie?



Why Present About The HTTP Event Collector?

Well, it's pretty simple...

Because we want you to learn from my mistakes!



“A PERSON WHO NEVER MADE A MISTAKE
NEVER TRIED ANYTHING NEW.”

- *Albert Einstein*

Why SplunkCloud?

- Dedicated SAAS in AWS
- 100% Uptime SLA
- Encryption (available add-on)
- Hybrid Capable
- Splunk CloudOps manages your SaaS hardware and software (search heads, indexers, etc.)
- No more late nights upgrading hardware/software!!!

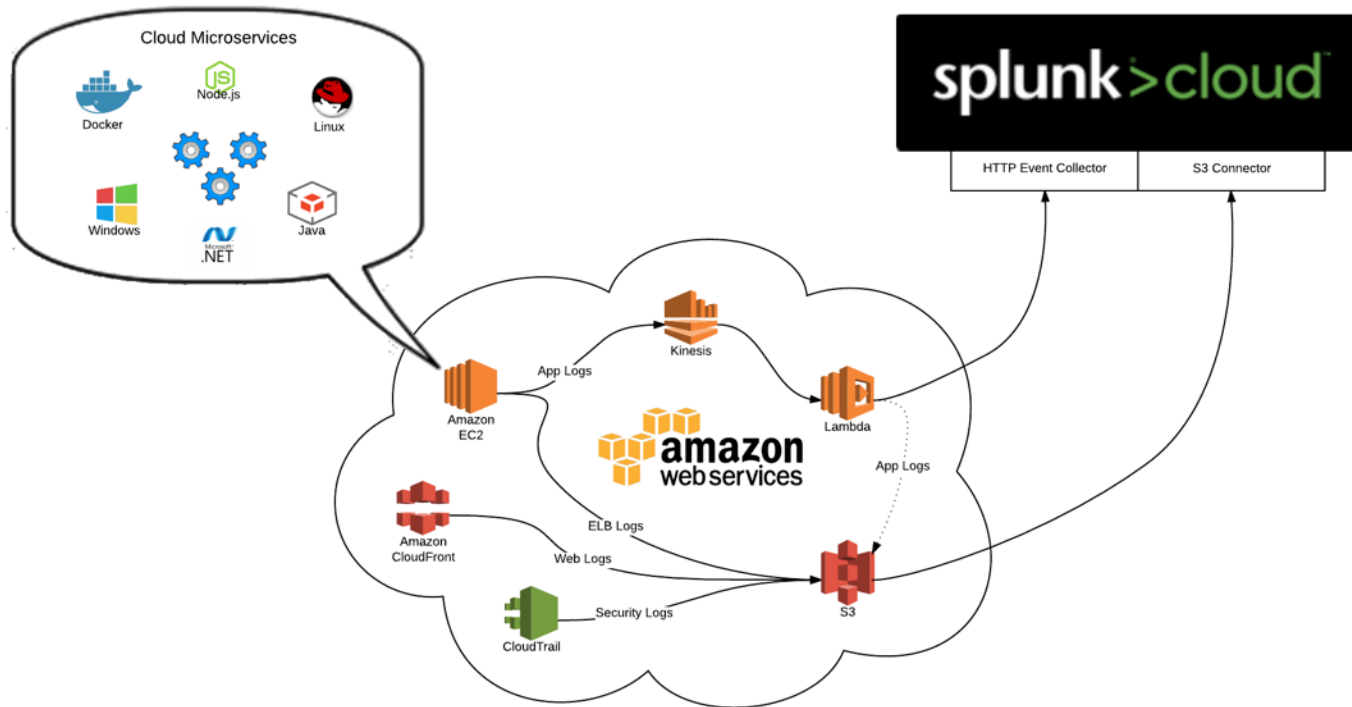
Cloud Service Providers



Google Cloud Platform



Cloud Services Architecture



Splunk's S3 Connector

The S3 Connector is efficient for:

- CloudFront
- ELB (Elastic Load Balancer)
- CloudWatch & CloudWatch Logs
- Cloudtrail
- Billing

This screenshot shows the 'Add AWS S3 Input' dialog box with the 'Settings' tab selected. The 'Name' field is labeled 'Input a name' with a link to 'Learn more about configuring this input.' Below the tabs, the 'Interval' is set to 1800, 'Source Type' is set to 'aws:s3', and 'Index' is set to 'default'. 'Cancel' and 'Create' buttons are at the bottom.

This screenshot shows the 'Add AWS S3 Input' dialog box with the 'Buckets' tab selected. It contains three dropdown menus: 'AWS Account' (labeled 'Select an AWS Account'), 'S3 Host Name' (set to 's3.amazonaws.com'), and 'S3 Bucket' (labeled 'Select a Bucket'). 'Cancel' and 'Create' buttons are at the bottom.

This screenshot shows the 'Add AWS S3 Input' dialog box with the 'Templates' tab selected. It contains four input fields: 'S3 Key Prefix', 'Start Date/Time' (set to '2016-05-16T15:35:04-0700'), 'Blacklist', and 'Whitelist'. 'Cancel' and 'Create' buttons are at the bottom.

The S3 Connector was working...

but then...

I went to .Conf 2015

.conf 2015 – The Great Shake Off



The HTTP Event Collector

Agentless, direct data onboarding via a standard developer API



Applications



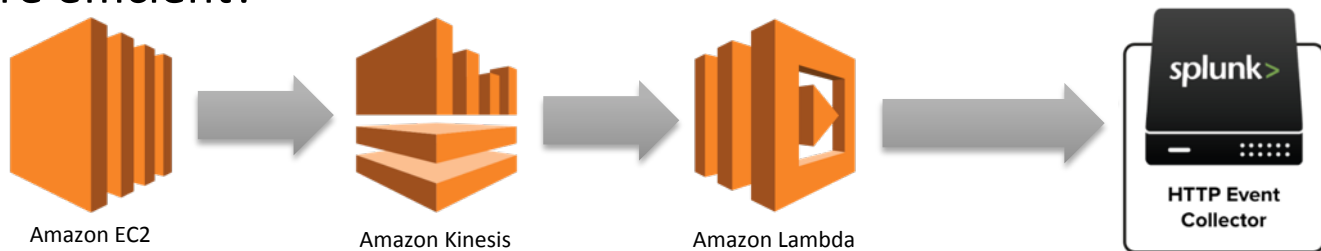
IoT Devices

```
curl -k https://<host>:8080/services/collector -H 'Authorization: Splunk  
<token>' -d '{"event": "Hello Event Collector"}'
```

The HTTP Event Collector

So, as I was sitting in the Keynote session on Day 1, I thought to myself:

- Could I go directly to the HTTP Event Collector from the application?
 - No more Universal Forwarders to install or update
 - Less agents running on the EC2 instances
- Would logging to Kinesis and then to the HTTP event collector be more efficient?



The HTTP Event Collector (cont.)

- Got back to the office and began doing further research
- Started planning our migration from the S3 Connector to the HTTP Event Collector
- We began seeing some latency issues with the ingest from S3 while running some load tests
- Timeline for migration accelerated due to latency of 15 minutes ingesting logs from S3



The HTTP Event Collector (cont.)



HOUSTON WE HAVE A PROBLEM

The HTTP Event Collector (cont.)

- We were running Splunk Cloud version 6.2
- The HTTP Event Collector did not exist in Splunk Cloud version 6.2
- Installed the HTTP Event Collector on a Heavy Forwarder running the Splunk Enterprise 6.3



HTTP Event Collector (cont.)

All was running well...

until...

we cranked up our traffic...

Luckily SplunkCloud made version 6.3 available for Production!!!



The HTTP Event Collector (cont.)

Splunk Cloud

- SplunkCloud 6.2 was upgraded to 6.3
- HTTP Event Collector was enabled on indexers
- Lambda functions updated
- Tuning began... (which we will discuss shortly)

Lambda Configuration

Select blueprint



Blueprints are sample configurations of event sources and Lambda functions. Choose a blueprint that best aligns with your desired scenario and customize as needed, or skip this step if you want to author a Lambda function and configure an event source separately. Except where otherwise noted, blueprints are licensed under [CC0](#).

Select runtime ▼

Filter

« < Viewing 1-9 of 46 > »

s3-get-object-python An Amazon S3 trigger that retrieves metadata for the object that has been updated. python2.7 · s3	config-rule-change-triggered An AWS Config rule that is triggered by configuration changes to EC2 instances. Checks instance types. nodejs · config	dynamodb-process-stream An Amazon DynamoDB trigger that logs the updates made to a table. nodejs · dynamodb
microservice-http-endpoint A simple backend (read/write to DynamoDB) with a RESTful API endpoint using Amazon API Gateway. nodejs · api-gateway	node-exec Demonstrates running an external process using the Node.js child_process module. nodejs	slack-echo-command-python A function that handles a Slack slash command and echoes the details back to the user. python2.7 · api-gateway · slack
simple-mobile-backend A simple mobile backend (read/write to DynamoDB). nodejs · mobile	kinesis-process-record-python An Amazon Kinesis stream processor that logs the data being published. python2.7 · kinesis	splunk-kinesis-logging Demonstrates logging events streamed from AWS Kinesis to Splunk's HTTP Event Collector. nodejs · splunk · kinesis


[Cancel](#)[Next](#)

Lambda Configuration


Configure triggers

Configure an optional trigger to automatically invoke your function.

Kinesis



▶



Lambda

Remove

Kinesis stream

examplestream

▼

i

Batch size

5000

←

i

Starting position

Latest

←

▼

i

In order to read from the Kinesis stream, your execution role must have proper permissions.

[Click here to add these permissions to your role.](#) **You must have popups enabled for this to work.** When the screen appears, review the policy and click "Allow" to add it to your role.

Enable trigger

☐

i

Cancel

Previous

Next

Lambda Configuration

Configure function

A Lambda function consists of the custom code you want to execute. [Learn more](#) about Lambda functions.

Name* Splunk_Lambda

Description Kinesis to Splunk's HTTP Event Collector

Runtime* Node.js 4.3

Lambda function code

```
156 var loggerInfo = {
157     splunkHost: 'https://http-inputs-example.splunkcloud.com:443', // Fill in with your Splunk host
158     base64EncodedEncryptedToken: '1111AB11-222B-333C-4D4E-55555555F555', // Fill in with base64-encoded, encrypted Splunk token here (step 1 above)
159     lambdaFunctionName: 'aws:lambda-example-logtype' // Fill in with your function name
160 };
```

```
162 initSplunkTokenAsync(loggerInfo);
163
164 var logger = new Logger({ functionName: loggerInfo.lambdaFunctionName });
165 logger.log('Loading function');
166 logger.flushAsync();
167
168 exports.handler = function(event, context) {
169     // Your code goes here
170 }
```

Advanced settings

These settings allow you to control the code execution performance and costs for your Lambda function. Changing your resource settings (by selecting memory) or changing the timeout may impact your function cost. [Learn more](#) about how Lambda pricing works.

Memory (MB)* 512

Timeout* 0 min 30 sec

All AWS Lambda functions run securely inside a default system-managed VPC. However, you can optionally configure Lambda to access resources, such as databases, within your custom VPC. [Learn more](#) about accessing VPCs within Lambda. **Please ensure your role has appropriate permissions to configure VPC.**

VPC No VPC

* These fields are required.

Cancel

Previous

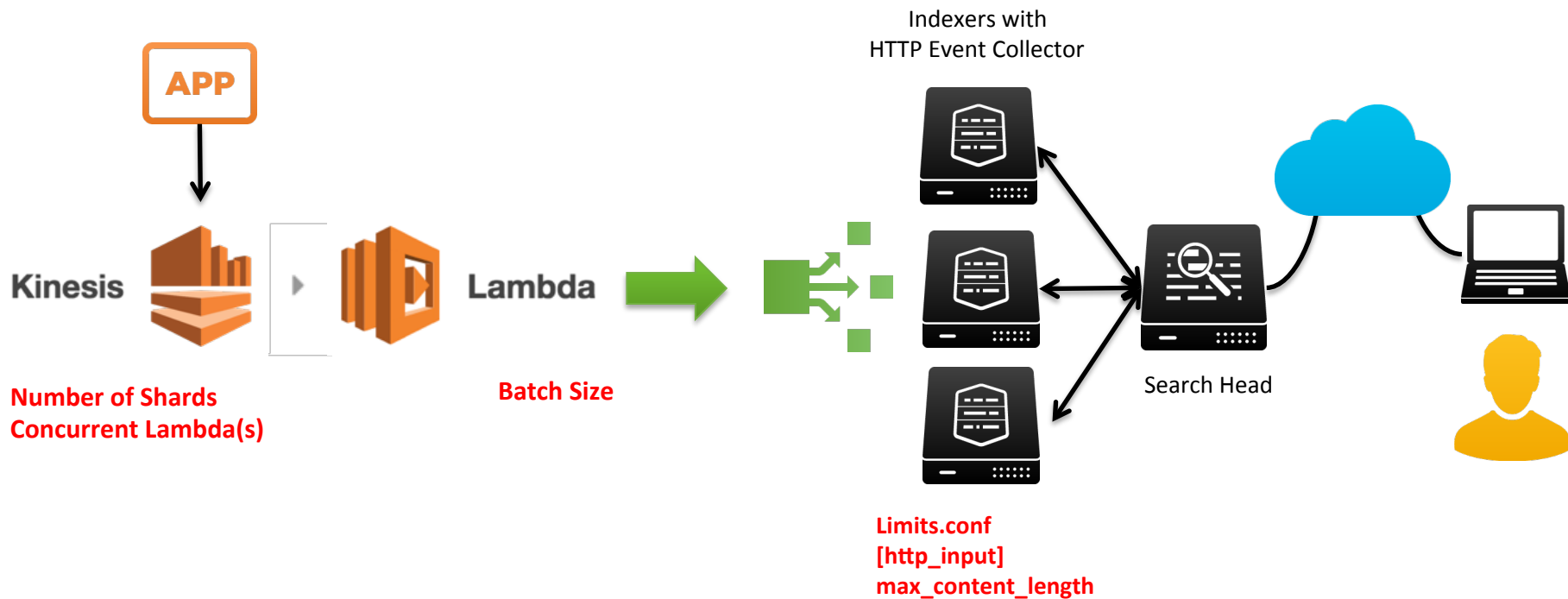
Next

Lambda Batch Size

- Batch size is the max number of events that sent for single invocation of the Lambda function
- Increased it from 100 to 1000 to 5000 to 10000 then back to **5000**
- 646 bytes average event size but then HTTP event collector started to error sometimes because of the default `max_content_length = 1,000,000` bytes
- $1,000,000 / 646 = 1548$ events in batch

```
sourcetype=applogs host=http-inputs.splunkcloud.com earliest=-24h latest=now |  
eval event_size=len(_raw) | stats avg(event_size)
```

Tuning The HTTP Event Collector



HTTP Event Collector Scaling

Limits.conf

[http_input]

max_content_length = 1000000 (bytes)

<http://docs.splunk.com/Documentation/Splunk/latest/Admin/Limitsconf>



Increase the max_content_length = 5,000,000 bytes (~5MB)

Batch size = 5000, memory for the Lambda at 512MB

HTTP Event Collector Scaling (con't)

- OS - Linux is 30% faster than Windows
- HTTP/HTTP(S) - HTTPS is 30% slower
- # of clients. Around 20K on a single box (if HTTP)

Lambda Tuning

- Make sure you use https/SSL between Lambda and HTTP Event Collector
- Set an appropriate batch size! “1000” is better than “100”
- Set Lambda Function to “**Latest**” NOT “Trim Horizon”
- Give your Lambda function the right amount of memory
- Change the timeout from “10” to “30”

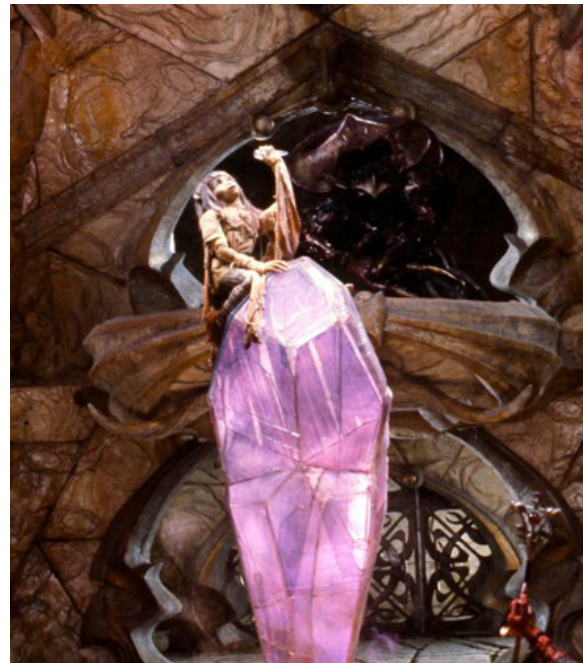
AWS Kinesis Shards

Each Shard can support:

- Up to 5 transactions per second for reads
- Up to a max total data read rate of 2MB/sec
- Up to 1K records per second for writes
- Up to a max total data write rate of 1MB/sec

<http://docs.aws.amazon.com/streams/latest/dev/service-sizes-and-limits.html>

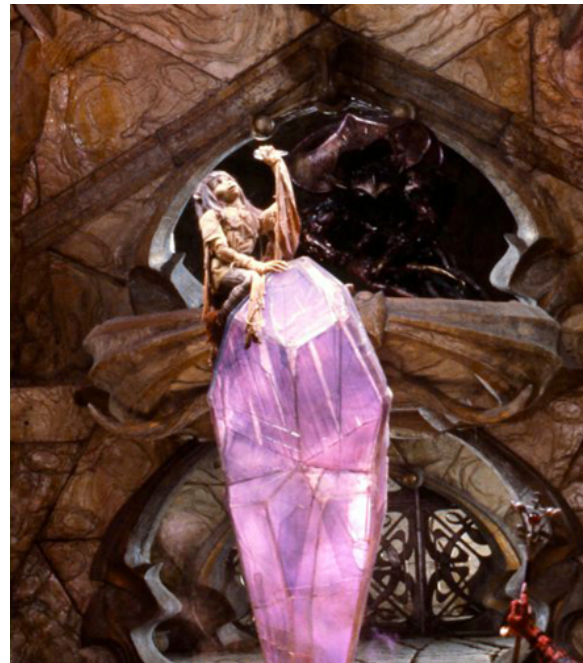
- 2MB/sec per Shard
- Plan for peaks



AWS Kinesis Shards

Make sure you split Kinesis into enough Shards so that it can handle:

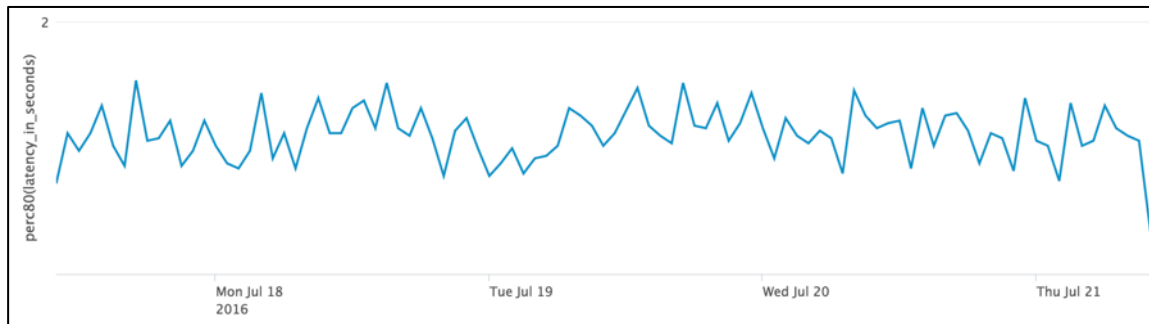
- Inbound streams from your Application
- Outbound streams to S3 and/or the HTTP Event Collector



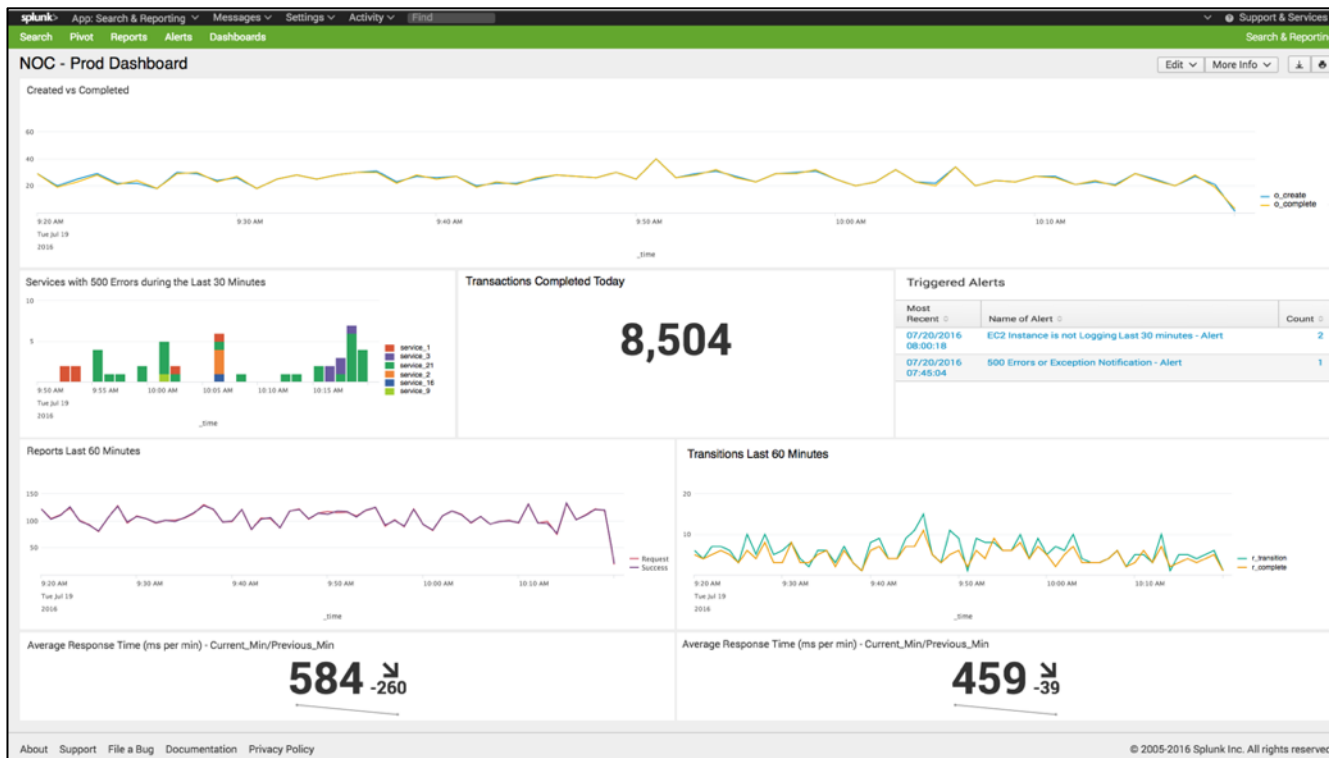
Measuring Our Progress

- Latency Search

```
sourcetype=applogs host=http-inputs.splunkcloud.com earliest=-2m latest=now |  
eval latency_in_seconds=( _indextime - _time ) |  
stats perc80(latency_in_seconds) as 80th_percentile_latency_in_seconds
```



Sample NOC Monitoring Dashboard



Resources

- .Conf 2015 “The Great Shake Off”

<http://www.ustream.tv/recorded/73893599> (starts at the 22min mark)

- Splunk’s HTTP Event Collector

<http://dev.splunk.com/view/event-collector/SP-CAAAE6M>

- AWS Lambda

<http://docs.aws.amazon.com/lambda/latest/dg/welcome.html>

- AWS Kinesis Shard Limits

<http://docs.aws.amazon.com/streams/latest/dev/service-sizes-and-limits.html>

Things to Remember

- S3 Works but the HTTP Event Collector is faster
- You must be using Splunk Cloud OR Splunk Enterprise 6.3 (or higher)
- Tune your Lambda Function (may impact your function \$\$\$)
- Scale up your HTTP Event Collector
- Make sure you have enough Kinesis Shards (may impact your Kinesis \$\$\$)
- Measure your progress through Dashboards and Alerts
- And as Albert Einstein suggests...

Don't be afraid to make mistakes!

Question & Answer



THANK YOU

.conf2016