Shrinking the Elephant in the Room

Maximizing logs’ business value with AWS

Chris Gordon | Software Engineer, Yelp
Zach Musgrave | Technical Lead, Yelp
Patrick Shumate | Solutions Architect, Amazon Web Services

September 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.
Introduction

DevOps, Splunk, Storage, and You
Our Splunk Cluster
Multi-site, multi-cluster, 4x replication

Storage
~1 petabyte

Daily Searches
Tens of thousands

Daily Ingestion
Tens of terabytes
The Elephant in the Room

- Initial budget set at $X
- One year later, costs at 150% of $X
  - Users want to ingest all the things
- After *Shrinking the Elephant*...
  - Storage costs **down** 15%
  - Headroom for new data **up** 40%
  - Logical retention **down** 20%

- Business effects
  - Users lost no insight
  - Business value left unchanged

https://www.flickr.com/photos/44124323641@N01/246805948
It Begins
Business Cat pulls the trigger

https://www.pinterest.com/pin/298715387758064097
Presupposition is the root of all conference talks

Infrastructure
Public cloud

License
Provisioned based on daily ingestion
Amazon Web Services (AWS) is a secure cloud services platform, offering compute power, database storage, content delivery and other functionality to help businesses scale and grow.

**A Broad IT Infrastructure Platform**
The AWS Cloud provides a broad set of infrastructure services, such as computing power, storage options, networking and databases, delivered as a utility: on-demand, available in seconds, with pay-as-you-go pricing.

https://aws.amazon.com/what-is-aws/
Budgeting
Several sources of cost

Splunk License
Essentials

Storage
Petabytes aren’t free

Indexers
Throughput to storage
Budgeting
One big source of cost

An engineer, who shall remain nameless

“Can we ingest ten terabytes a day? And retain it forever?”
DevOps
Or, we’re all in this together

- Retain all data forever…
  - Users: Happy
  - Finance: Livid
- Retain all data for one day…
  - Users: Frustrated
  - Finance: Thrilled
- Retain data based on value…
  - Users: Mostly Happy
  - Finance: Placated

Us vs. Them

We
What’s Value, Anyway?

- Benefit the business
  - Close more and larger deals faster
  - Retain existing accounts
  - Limit resource loss (right-sizing)
- Improve employee productivity
  - Find insights faster
  - Discover correlations across verticals
  - Automate rote tasks
- Something else?
  - That depends on you!

https://makeameme.org/meme/what-do-you-3kq4js
Value is Intrinsic

- Your users interact with data...
  - Therefore, it’s valuable
  - Trust that they know best

- How valuable?
  - Frequency of access
  - Depth of interaction
  - Consequences if it disappeared

https://www.gulosolutions.com/2015/02/secret-making-uxd-feel-right/
How To Quantify Value
And engineer for flexibility

► Splunk isn’t your data’s final resting place
  • Next up: Next-generation ingestion from Amazon S3
► Splunk tells you a lot about access patterns
  • But only if you ask nicely: We’ll show you how
► Merely shutting off retention isn’t acceptable
  • Business is agile: You have to hedge your bets
► Pre-processing is your friend
  • Not just map/reduce
  • More like map/count/reduce/reduce/reduce/reduce…

http://trekcore.com/blog/
We Can All Get Along
It’s the DevOps mantra

http://if-you-give-a-mouse-a-cookie.wikia.com/wiki/File:If_you_give_a_mouse_a_cookie.jpg
Getting Your Data into Splunk

Amazon S3, SQS, and more
Getting Your Data into Splunk
Leveraging S3 as your data source

- Where do the logs come from? Amazon S3!
  - Cheap, always available storage
  - Useful API for retrieving and manipulating your data
Storage Is A Platform: AWS Storage

Amazon EFS
Amazon EBS
Amazon EC2 Instance Store
Amazon S3
Amazon Glacier

AWS Direct Connect
AWS Snowball ISV Connectors
Amazon Kinesis Firehose
S3 Transfer Acceleration
Storage Gateway
How do the logs get into Splunk? Amazon SQS and SNS

- SQS and S3 APIs do all the hard work
- Scripted input: Read from SQS queues
  - A config file can inform the script about what to consume and where to get it
    - Auto-generate `props.conf` and `inputs.conf`
- Scaling up is easy...
  - Just add more forwarders!

```yaml
sqs:
  my-sqs-queue:
    aws_region: "us-west-2"

logs:
  my-sourcetype:
    sqs_name: "my-sqs-queue"
    s3_prefix:
      - "s3://yelp-logs-us-west-2/logs/mylog/"
    index: "my index"
    time_prefix: "\"timestamp\":"'
    time_format: "%s"
```
Set up a bucket notification
• Simple case: Send straight to SQS queue
• Less simple case: Send to SNS topic
  • Why?
    – Allows for multiple consumers of the notifications
  • How?
    – Send to SNS topic
    – SNS topic feeds into your SQS queue
Getting Your Data into Splunk
What does all this get you?

▶ The obvious
  • You automatically ingest whatever S3 data you want!

▶ The less obvious
  • You can **backfill** whatever data you want with little to no effort
    • Simple: A script generates a *bucket notification* for anything you want to ingest
    • Add a new log to Splunk? Instantly ingest any historical data

▶ The least obvious
  • You can **re-ingest** data that’s rolled out of retention on a whim
    • Really make use of that unlimited license

http://www.backtothefuture.com/
Allocate a short retention index for ad-hoc ingestion and search

- Ingest the data, do what you want with it, then let it roll out again soon after
  - Ingest the data, run a summary report to collect a bunch of stats to be stored long-term
    - More on this later
- Don’t keep data around longer than you need it

http://btulp.com/12208/cake-graphic/
Are there any downsides? *Sort of…*

- Can’t think of index retention windows the same
  - Data rolls out based on index time, not time of the event
    - You’ve broken time
- What does this mean?
  - You will need to control retention by setting the appropriate size limit
  - You can set up alerts to let you know when data is starting to roll out too early

Minimizing Storage Costs without Losing Value

Or, Summary Indexing 101
Minimizing Storage Costs

What is Summary Indexing?

What is summary indexing?

- Think of it as computing statistical rollups of your logs
- As data gets older, you don’t care about individual logs
  - Trends become more important
- These stats have insignificant storage cost
  - You can essentially store them forever
- Dashboards based on summary indexes load extremely quickly
Minimizing Storage Costs
Leveraging Summary Indexes

▶ Take summary indexes a step further
  • Use them to enable storing super verbose logs in Splunk
  • Lots of DevOps tools produce insane amounts of logs
    • Puppet, CloudTrail, HAProxy, NGINX, etc…

▶ How to ingest them cost effectively?
  • You guessed it, summary indexes!
Minimizing Storage Costs
Leveraging Summary Indexes (cont)

- These individual logs typically stop being useful after a few days
  - So, only store them for a few days!

- Perform daily (or weekly) summary reports to persist any statistics or trends
  - These can be stored in a summary index at insignificant cost

- How long should retention be?
  - We'll come back to this

- What if someone needs the logs longer?
  - No problem, just re-ingest them!
Measuring the Business Value of Your Logs

Analyzing the cost/value of your data
Measuring Cost/Value
Mapping log ingest and usage

▶ So we’ve covered how to minimize cost, but…
  • How to attribute a cost to each log?
    • *license_usage* logs!

  • How to determine a log’s value relative to its cost?
    • *audit* logs!

▶ You still likely need to talk to stakeholders, but it’s a starting point
  • Who are the stakeholders?
    • We’ll come back to this
How to attribute a cost to each log?

- Splunk indexer deployments: Relatively static
  - Most of the cost is here
  - For on-site deployments, costs don’t change as ingestion goes up or down
    - Costs change only when you scale storage/indexer count
- Solution: Attribute a percentage of cost to each log based on their ingestion ratios

Note: Numbers have been changed to protect the innocent…
How to determine a log’s value?
- Count how often logs are searched
- Re-use the cost per log data
- Result is a Cost per Search metric
- Useful for finding candidates for removal

Types of high CPS logs
- Unsearched: Can be removed easily
- Seldom searched: Require further investigation

Measuring Cost/Value
Which logs are being used?
Cost Per Search
- Useful for finding high cost/low value logs
- **But**... once identified, what do you do with them?
  - Ask stakeholders if logs can just be removed

Which brings us back to the question:
- Who are the stakeholders?
  - Instead of just counting searches, count by user!
We’ve identified the users of our removal candidate
• What if they push back?
  • This log is useful to somebody
  • Put it on cheaper hardware instead of removing completely
  • Lower QOS while still supporting developer needs

Where else can you reduce costs without impacting users?
• Drop your unused retention!
  • But how to identify unused retention? Back to the audit logs!
How far into the past is a log useful?

- The **audit** log has the answer!
- Each search has a start time
  - *Retention window = time of search – start time*
- Count the number of searches in **tiered** windows
  - One day, three days, one week, two weeks, three weeks…

<table>
<thead>
<tr>
<th>stream</th>
<th>Total Count</th>
<th>costPerMonth</th>
<th>OneDay Count</th>
<th>TotalCostMinusCostOfStoringOneDay</th>
</tr>
</thead>
<tbody>
<tr>
<td>nginx_access</td>
<td>2863</td>
<td>$1,390.65</td>
<td>120</td>
<td>$1,488.34</td>
</tr>
<tr>
<td>nginx_access</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Note:** Numbers have been changed to protect the innocent…

**stream = index.sourcetype**

**Total Cost**
Many logs’ search counts will drop off after a few days
  - But not completely
  - How do we lower retention without also removing value?
    - Remember summary indexes?

---

### Measuring Cost/Value

**Drop Retention Without Losing Value**

Table:

<table>
<thead>
<tr>
<th>Percentile of CPS to show</th>
<th>Minimum Cost per Month</th>
</tr>
</thead>
<tbody>
<tr>
<td>95</td>
<td>60</td>
</tr>
</tbody>
</table>

#### Display the logs in the Nth percentile of CPS for any retention window

**Note:** Numbers have been changed to protect the innocent…

<table>
<thead>
<tr>
<th>stream</th>
<th>Users</th>
<th>SearchCounts</th>
<th>costPerMonth</th>
<th>TotalCount</th>
<th>OneDayCount</th>
<th>CostOfStoringPastOneDay</th>
<th>CPS_OneDay</th>
<th>ThreeDayCount</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>user_288, user_299</td>
<td>261</td>
<td>$2,603.46</td>
<td>878</td>
<td>78</td>
<td>$2,603.68</td>
<td>33.38</td>
<td>68</td>
</tr>
<tr>
<td>2</td>
<td>user_82</td>
<td>15910</td>
<td>$1,866.24</td>
<td>16025</td>
<td>136</td>
<td>$1,803.07</td>
<td>13.26</td>
<td>46</td>
</tr>
<tr>
<td>3</td>
<td>user_64</td>
<td>21690</td>
<td>$1,578.43</td>
<td>22201</td>
<td>20</td>
<td>$1,525.82</td>
<td>76.29</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>user_154</td>
<td>5789</td>
<td>$1,292.75</td>
<td>5890</td>
<td>41</td>
<td>$1,249.66</td>
<td>30.48</td>
<td>37</td>
</tr>
<tr>
<td>5</td>
<td>user_238</td>
<td>1238</td>
<td>$954.00</td>
<td>2563</td>
<td>120</td>
<td>$922.20</td>
<td>7.68</td>
<td>52</td>
</tr>
<tr>
<td>6</td>
<td>user_332, user_355</td>
<td>75</td>
<td>$884.61</td>
<td>178</td>
<td>54</td>
<td>$855.13</td>
<td>15.84</td>
<td>40</td>
</tr>
<tr>
<td>7</td>
<td>user_514, user_532</td>
<td>10</td>
<td>$622.66</td>
<td>44</td>
<td>18</td>
<td>$601.91</td>
<td>33.44</td>
<td>18</td>
</tr>
</tbody>
</table>
Remember: Attributed costs are just a percentage of total infrastructure costs

- Costs don’t actually go down until you scale down the cluster
- Use cost as a proxy for expected value
- Instead of thinking about savings, think about what else you can ingest
  - Provide more value

What if you do need to reduce costs?

- Keep a spreadsheet to inform your decisions
- How many indexers you need to drop to meet a goal
- How much data you need to drop to fit into a smaller cluster

<table>
<thead>
<tr>
<th>Index</th>
<th>Current Total Size (all indexers)</th>
<th>Current Median Oldest Age on Indexers (days)</th>
<th>Desired Retention (days)</th>
<th>Desired Headroom (protection from log explosions/indexer losses)</th>
<th>Estimated total size (when at full retention)</th>
<th>Space needed (total size + headroom)</th>
<th>Space required per index (GB)</th>
<th>Percentage of Whole</th>
<th>Cost (per month)</th>
<th>Space to Allocate MB (indexes, cont.)</th>
</tr>
</thead>
<tbody>
<tr>
<td>nginx_access</td>
<td>60,061.00</td>
<td>30</td>
<td>7</td>
<td>5.00%</td>
<td>14,014.23</td>
<td>14,714.95</td>
<td>294.30</td>
<td>4.10%</td>
<td>$2,225.67</td>
<td>303,128.00</td>
</tr>
</tbody>
</table>

Note: Numbers have been changed to protect the innocent…
Wrapping Up

The Elephant Has Been Shrunk
1. Our cluster is far more efficient
   • 40% more headroom for future growth
   • In line with company budget expectations

2. Our users retain all valuable data
   • We know what that is now
   • We can leverage it to further optimize

3. Our ingestion scales along with us
   • We add, remove, *re-add* data easily
   • Special cases are easy: *Put it in S3!*

Final Results
The elephant has been shrunk
1. Don’t be an Admin. Be an Owner.
   • Take responsibility for your Splunk!

2. Make cluster decisions with data
   • Measure real use of your cluster
   • If they use it, then it’s important.

3. Ingest intelligently
   • Use Amazon S3 to do this cheaply and flexibly
Thank You

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