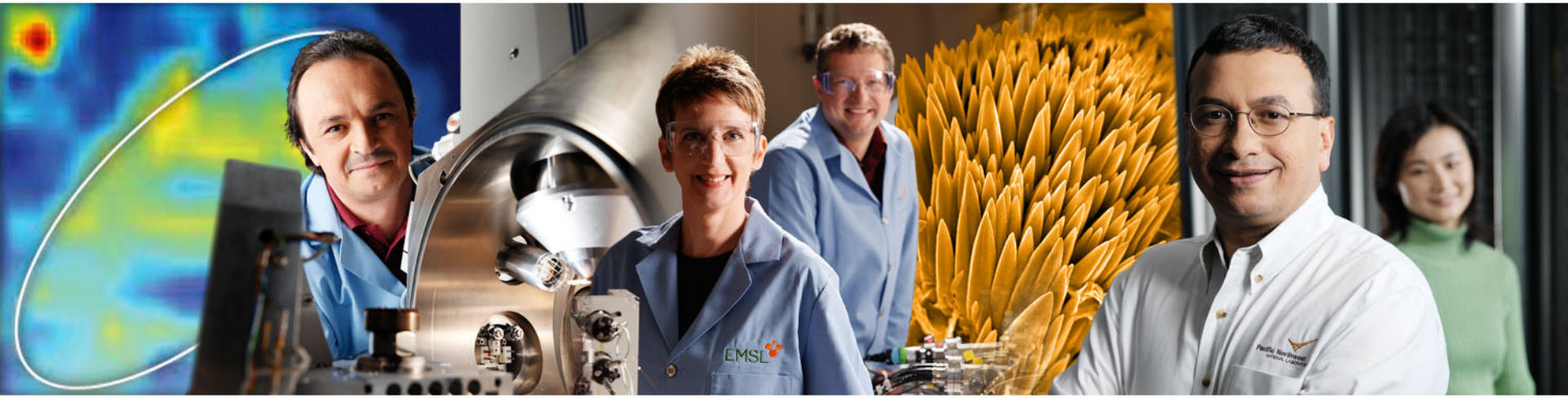
EMSL

Log Analysis with the ELK Stack (Elasticsearch, Logstash and Kibana)

Gary Smith, Pacific Northwest National Laboratory



A Little Context

EMSL

- The Five Golden Principles of Security
  - Know your system
  - Principle of Least Privilege
  - Defense in Depth
  - Protection is key but detection is a must.
  - Know your enemy.

In the Days of Chinook

- For most of Chinook's lifetime, the MSC used the "free" version of Splunk to review the syslogs.
- Splunk Inc. has an interesting licensing model that's sort of like an all-you-can-eat buffet where you pay by the pound:
  - The more you ingest, the more you pay.
- If you ingest < 500MB of logs a day, Splunk is "free".
- If you go over that limit too many times, Splunk will continue to index your logs but you can't view them until you pay them $$$ or you reinstall Splunk.
- Consequently, I was always fiddling with Syslog-NG's rules to keep the cruft out and keep the daily log data < 500MB.

The Dawning of Cascade

EMSL

- When the talk about what would later be known as Cascade started ramping up, I started looking at a replacement for Splunk because I knew that I would not be able to keep under the 500MB limit with two supercomputers in operation.
- The price for a commercial license for Splunk for the amount of log data the MSC's systems produced would be prohibitive.

Alternatives to Splunk

EMSL

- I looked at a lot of alternatives to Splunk. These are just some of them:
  - Graylog2
  - Nxlog
  - Octopussy
  - Logscape,
  - ELSA
  - LOGanalyzer
  - Logalyzer
  - Logwatcher
  - logHound
  - logReport
  - Logsurfer
  - PHP-Syslog-NG

Alternative to Splunk

- Some wouldn't build. Some wouldn’t work.
- Some were slow.
- Some had an abysmal user interface.
- Most all of them had a MySQL, PostgreSQL, or similar relational database backend for storage and retrieval.

The Solution: ELK Stack [Elasticsearch, Logstash, Kibana]

EMSL

- Elasticsearch: Indexing, storage and retrieval engine
- Logstash: Log input slicer and dicer and output writer
- Kibana: Data displayer

Early Experiments With The Stack

EMSL

- In the early stages of the ELK stack, the pieces didn't play well together.
- Early versions of Logstash needed specific versions of Elasticsearch and those weren't the latest ones.
- This caused some problems because Kibana wanted the latest version of Elasticsearch.
- So I tried a couple of alternatives to ELK.

EFK

- EFK => Elasticsearch FluentD Kibana
- This worked pretty well.
- Good points:
  - With FluentD, you install it, point it at Elasticsearch, point your syslogs at Fluentd and you're good to go.
- Bad Points:
  - There's not much you can do to extend FluentD to do things with the syslog events coming in.

ERK

EMSL

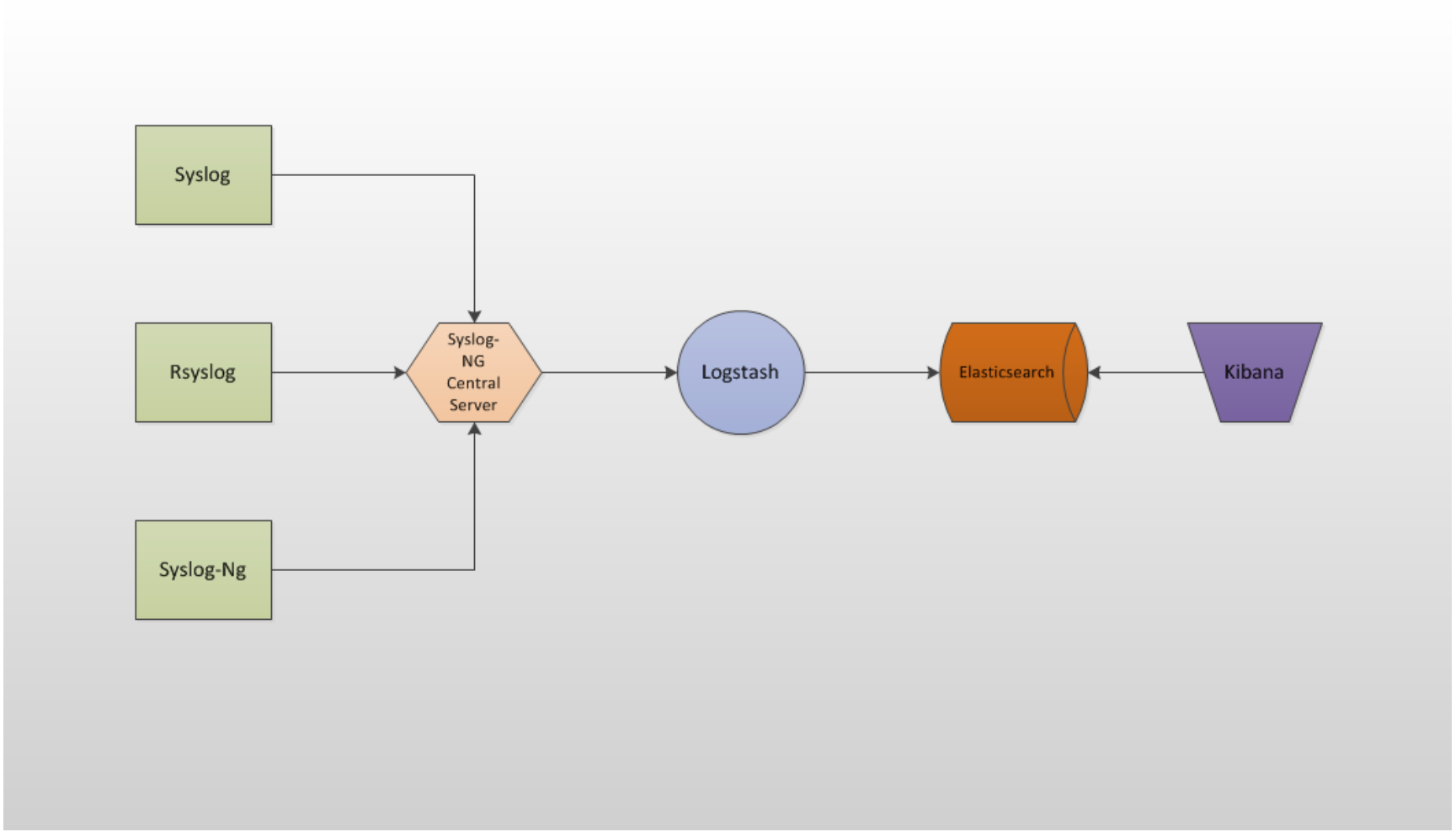
- ERK => Elasticsearch Rsyslogd Kibana
- There's an Rsyslogd plugin that takes syslog events and sends them to Elasticsearch.
- Good points:
  - Much like FluentD, you install it, point it at Elasticsearch and point your syslogs at Rsyslogd and you're good to go.
- Bad Points:
  - The plugin requires the very latest version of Rsyslogd, so you have to build the latest version of Rsyslogd and the plugin.
  - Then, you have to maintain the version of Rsyslogd and the plugin since it's two major revisions above what's available in RHEL.

Finally, One Big Happy Family

EMSL

- The dysfunctional aspects of the ELK stack got worked out.
- Now the members of the ELK stack play well together after being unified with help from the Elasticsearch people.

Components of The ELK Stack [Elasticsearch Logstash Kibana]



Logstash

- Logstash was developed by Jordan Sissel when he was a system administrator at Dreamhost.
- Jordan needed something that could handle a peak of 20,000 messages per second.
- Logstash is easy to set up, scalable, and easy to extend.

Logstash Hosts

- In most cases there are two broad classes of Logstash hosts:
  - Hosts running the Logstash agent as an event forwarder that sends you application, service, and host logs to a central Logstash server.
  - Central Logstash hosts running some combination of archiver, indexer, search, storage, and web interface software which receive, process, and store your logs.

Logstash Basic Configuration File

EMSL

- A basic configuration file for Logstash has 3 sections:
  - input
  - filter
  - output

The Input Section

- Inputs are the mechanism for passing log data to Logstash. Some of the more useful, commonly-used ones are:
  - **file**: reads from a file on the filesystem, much like the UNIX command "tail -f"
  - **syslog**: listens on the well-known port 514 for syslog messages and parses according to RFC3164 format
  - **lumberjack**: processes events sent in the lumberjack protocol. Now called *logstash-forwarder*.

The Filter Section

- Filters are workhorses for processing inputs in the Logstash chain.
- They are often combined with conditionals in order to perform a certain action on an event, if it matches particular criteria.
- Some useful filters:
  - **grok**: parses arbitrary text and structures it.
    - Grok is currently the best way in Logstash to parse unstructured log data into something structured and queryable.
    - With 120 patterns shipped built-in to Logstash, it's more than likely you'll find one that meets your needs!
  - **mutate**: The mutate filter allows you to do general mutations to fields. You can rename, remove, replace, and modify fields in your events.
  - **drop**: Drop an event completely, for example, *debug* events.
  - **geoip**: Adds information about geographical location of IP addresses (and displays amazing charts in Kibana)

The Output Section

- Outputs are the final phase of the Logstash pipeline.
- An event may pass through multiple outputs during processing, but once all outputs are complete, the event has finished its execution.
- Some commonly used outputs include:
  - **elasticsearch**: If you're planning to save your data in an efficient, convenient and easily queryable format… Elasticsearch is the way to go. Period. Yes, we're biased :)
  - **file**: writes event data to a file on disk.
  - **statsd**: a service which "listens for statistics, like counters and timers, sent over UDP and sends aggregates to one or more pluggable backend services".
    - If you're already using statsd, this could be useful for you!

Elasticsearch: Hard made Easy

EMSL

- Elasticsearch is a powerful indexing and search tool.
- The Elasticsearch team says, "Elasticsearch is a response to the claim, 'Search is hard'".
- Elasticsearch is easy to set up, has search and index data available RESTfully as JSON over HTTP and is easy to scale and extend.
- It's released under the Apache 2.0 license and is built on top of Apache's Lucene project.

Elasticsearch: How it works!

- Elasticsearch is a text indexing search engine.
- The best metaphor to describe Elasticsearch is the index of a book.
- You flip to the back of a book, look up a word and then find the reference page.
- This means that rather than searching text strings directly, Elasticsearch creates an index from incoming text and performs searches on the index rather than the content.
- As a result, it is very fast.

Elasticsearch Configuration

EMSL

- Elasticsearch is started with a default cluster name of "elasticsearch" and a random node name based on characters from the X-Men.
- A new random node name is selected each time Elasticsearch is restarted if one has not been chosen.
- If you want to customize your cluster and node names to ensure unique names, edit `/etc/elasticsearch/elasticsearch.yml`.
- This is Elasticsearch's YAML-based configuration file.

The Kibana Web Interface

- The Kibana web interface is a customizable dashboard that you can extend and modify to suit your environment.
- It allows the querying of events, creation of tables and graphs as well as sophisticated visualizations.
- The Kibana web interface uses the Apache Lucene query syntax to allow you to make queries.
- You can search any of the fields contained in a Logstash event, for example, message, syslog_program, etc.
- You can use Boolean logic with AND, OR, NOT as well as fuzzy searches and wildcard searches.
- You can:
  - Build complex queries (including saving them and displaying the results as a new panel)
  - Graph and visualize data
  - Produce tables and display data on maps and charts.

Troubleshooting: Is It Running?

- How do you tell if Elasticsearch is running?
- Do this: `curl http://10.0.0.1:9200/_status?pretty=true`
- This will return a page that contains a variety of information about the state and status of your Elasticsearch server.

Troubleshooting: Are Logstash And Elasticsearch Working Together?

EMSL

- How can you check to see if Logstash is getting events to Elasticsearch and they are getting indexed?
- Do this:

```
curl "http://localhost:9200/_search q=type:syslog&pretty=true"
```

Troubleshooting: Syntax Checking Your Logstash Configuration File

- After you've written/modified your Logstash configuration file, how do you know it's syntactically correct before you put it into production
- Do this:

```
/opt/logstash/bin/logstash agent -f logstash.conf --configtest
```

Getting Rid Of Old Data

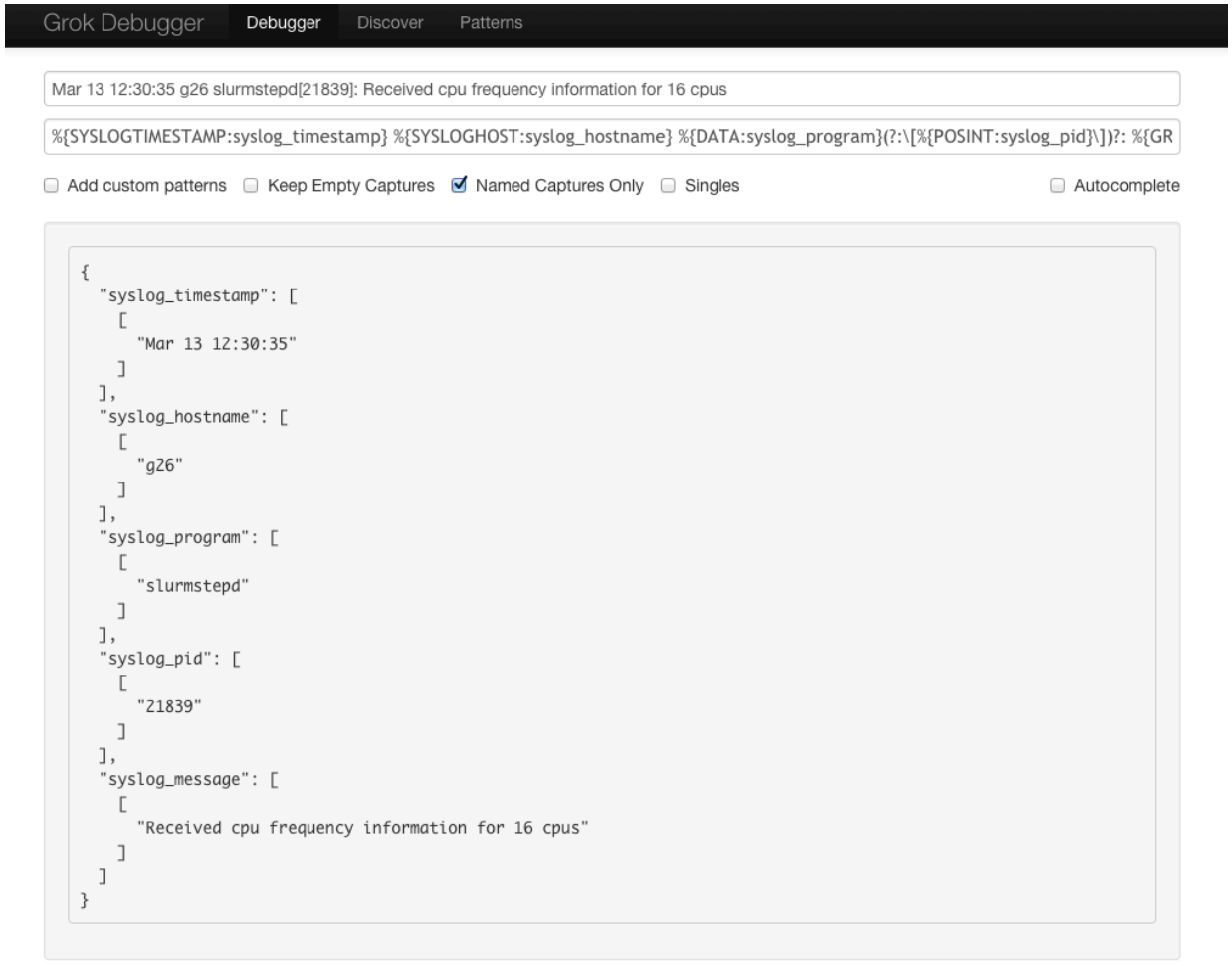
EMSL

- One of the things I could never figure out with Splunk is "How do I get expire old data out of Splunk?"
- What about Elasticsearch? Can I expire old data out of Elasticsearch and keep only what's recent and relevant?
- As it turns out, like most things in the Linux-Sphere, there's more than one way to do it.
- I have a daily cron job that runs a Perl script that deletes data out of Elasticsearch older that 31 days.
- There is also a python program called *Curator* by Aaron Mildenstein that helps you curate, or manage your Elasticsearch indices like a museum curator manages the exhibits and collections on display.

Debugging Grok Patterns

- Remember those patterns I was using in the grok filter to parse out the fields in a syslog event? How did I come up with those?
- I used the Grok Debugger at `http://grokdebug.herokuapp.com/` to work out the construction of the pattern.
- The Grok Debugger is an interactive web page that facilitates rapid creation of patterns based on sample input.

The Grok Debugger In Action



References

- http://www.elastic.co/
- http://logstash.net/docs/latest
- https://www.elastic.co/products/kibana
- https://github.com/elastic/curator/wiki
- http://www.fluentd.org/
- http://www.rsyslog.com/
- http://grokdebug.herokuapp.com/

Questions?

Gary Smith

Information System Security Officer, Molecular Science Computing, Pacific Northwest National Laboratory

Richland, WA

gary.smith@pnnl.gov