

puppetconf²⁰¹⁵

Puppet Reporting with Elasticsearch, Logstash, and Kibana

Indeed.com

Alex Elman

Devops Engineer, Operations

Who we are

Alex Elman



@_pkill



github.com/pkill

Connor Kelly



@cllyross



github.com/clly

I help
people
get jobs.

What this talk will cover

- Using Puppet reporting to reduce technical debt
- Installing and configuring Elasticsearch, Logstash, and Kibana
- Getting reports into Elasticsearch
- Using Puppet reports to
 - Troubleshoot failures
 - Reveal bottlenecks in catalog application
 - Identify git commits that broke Puppet

What this talk won't be covering

- Architecture of Elasticsearch or Logstash
- In-depth exploration of the Elasticsearch query API
- How to scale Elasticsearch and Logstash
- Opinions on the best Puppet report processor

Technical Debt

puppet dashboard • 1.2.23 • Home • Nodes • Groups • Classes • Reports • File Search • Enable autorefresh

Background Tasks
All systems go

Nodes

- Unresponsive
- Failed
- Pending
- Changed
- Unchanged
- Unreported

All

Add node Radiator View

Group
Database Servers (14)
Add group

Class
Add class

Nodes

Daily run status
Number and status of runs during the last 30 days:

Nodes

Export nodes as CSV

Node	Latest report	Total	Failed	Pending	Changed	Unchanged
Total		2967208	997	6764	12084	2953080
✓	2015-09-21 13:50 CDT	1918	0	0	1	1917
✓	2015-09-21 13:50 CDT	1918	0	0	1	1917
✓	2015-09-21 13:50 CDT	1979	0	0	1	1978
✓	2015-09-21 13:50 CDT	1979	0	0	1	1978
✓	2015-09-21 13:50 CDT	3759	0	0	4	3755
✓	2015-09-21 13:21 CDT	2370	40	1	16	2314
✓	2015-09-21 13:21 CDT	2369	1	0	1	2367
✓	2015-09-21 12:48 CDT	2186	0	0	3	2183
✓	2015-09-21 12:46 CDT	2186	0	0	3	2183
✓	2015-09-21 12:38 CDT	2325	1	0	2	2322
✓	2015-09-21 12:21 CDT	1868	0	1	0	1867
✓	2015-09-21 12:16 CDT	1890	0	1	0	1889
✓	2015-09-21 12:00 CDT	1384	1	0	0	1383
✓	2015-09-21 11:59 CDT	2039	0	0	4	2035
✓	2015-09-21 11:57 CDT	2052	1	0	0	2051
✓	2015-09-21 11:57 CDT	1886	0	0	3	1883
✓	2015-09-21 11:56 CDT	1843	2	0	2	1839
✓	2015-09-21 11:56 CDT	2039	0	0	4	2035
✓	2015-09-21 11:55 CDT	2035	0	0	0	2035
✓	2015-09-21 11:55 CDT	1682	1	0	2	1679

« Previous 1 2 3 4 5 6 7 8 9 ... 90 91 Next » Per page: 20 100 all



“When technical debt accrues, everybody loses.”
- Gene Kim

Puppet Reporting & Technical Debt

- What change was introduced that broke Puppet?
- What are the slowest resources applying on each node?
- What resources are causing the most issues?
- Which nodes haven't been enforcing their catalog the longest?

Puppet Report processors and integrations

There are lots of them...

- Puppet dashboard
- Puppet Enterprise
- PuppetDB
- Riemann
- Ganglia
- tagmail
- graphite/graphana
- HTTP



Puppet Enterprise Configuration Manager

The screenshot shows the Puppet Enterprise Configuration Manager interface. On the left is a dark sidebar with navigation links: Configuration Management (selected), Nodes, Overview, Events, Reports, License, Access control, Help, My account, (Log out), and v4.0.0. The main content area is titled "Overview" and displays the last run status of nodes. It shows 5 total nodes under Puppet management. A summary table indicates 0 Failed, 0 Changed, 1 Unchanged, 0 No-op, 4 Unresponsive, and 0 Unreported nodes. Below this is a table listing node names, their last report times, and links to view the node graph.

Configuration Management

Gain insights and visibility into your Puppet-managed infrastructure.

Overview

Events

Reports

License

Access control

Help

My account

(Log out)

v4.0.0

Overview

View the last run status of your nodes and drill in to their configuration details.

Reporting on **5** total nodes under Puppet management.

Browser refreshed 6 minutes ago.

Filter by run status **all** ▾ [Filter by fact value](#)

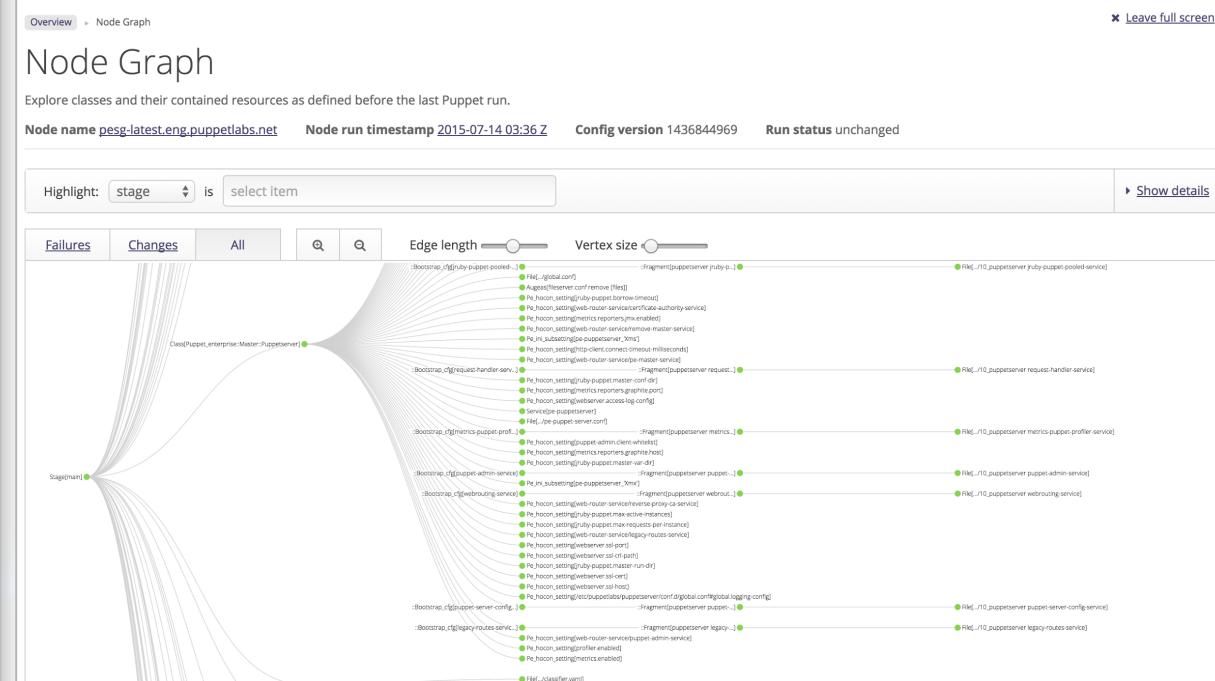
Run status	Node name	Last report	Node graph
● Failed	pesg-latest.eng.puppetlabs.net	2015-07-14 03:36 Z	View graph
● Changed	ix9a3f69sg4wju8.delivery.puppetlabs.net	2015-07-11 05:18 Z	View graph
● Unchanged	s3ha07himm6pv72.delivery.puppetlabs.net	2015-07-09 23:42 Z	View graph
● No-op	il3xq2krlr0emd4.delivery.puppetlabs.net	2015-07-09 19:45 Z	View graph
● Unresponsive	tfe8rux4cfb84t3.delivery.puppetlabs.net	2015-07-09 18:24 Z	View graph

Radiator

Export data

Configuration Manager Node Graph

- Exploded view of resources and their dependencies
 - Easily pinpoint cause of long chains of dependency failures
 - Analyze interactions between classes and defined types

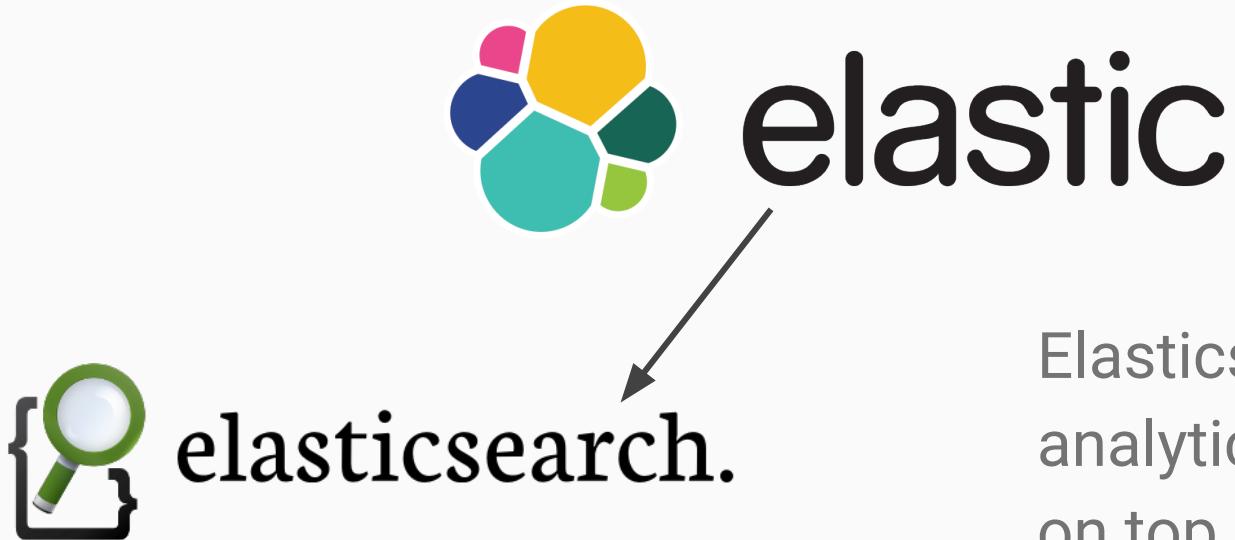


Elasticsearch, Logstash and Kibana



elastic

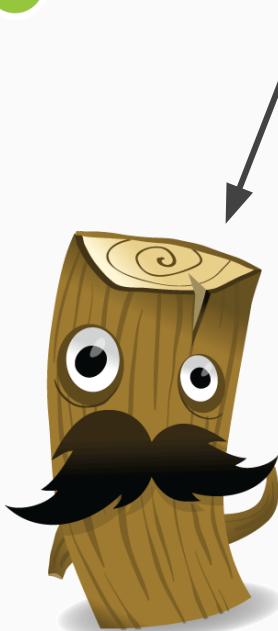
Maintained by a
company called Elastic.



Elasticsearch is an analytics platform built on top of a distributed search engine.



elastic



Logstash is a log processing engine. It takes incoming logs and uses grok filters to create structured data from unstructured messages.



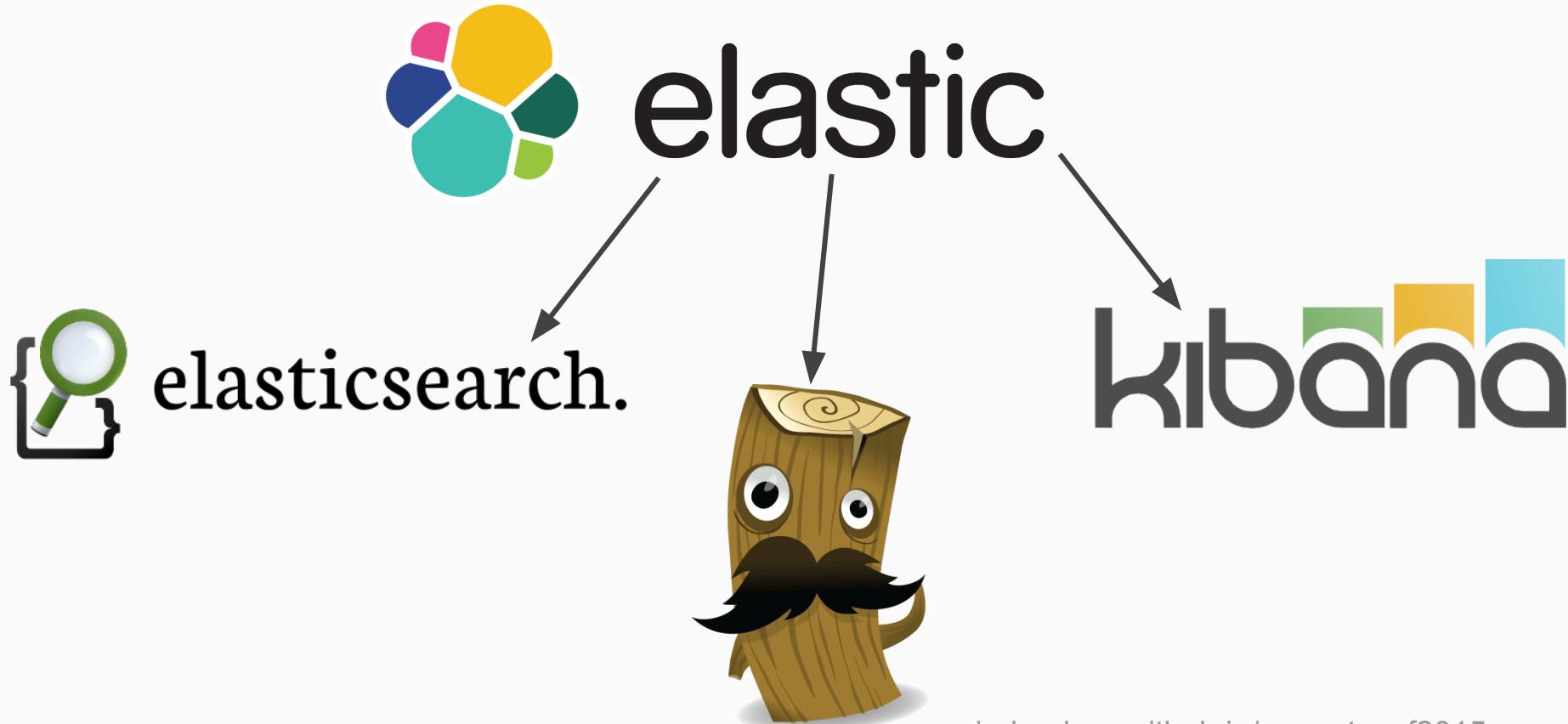
elastic



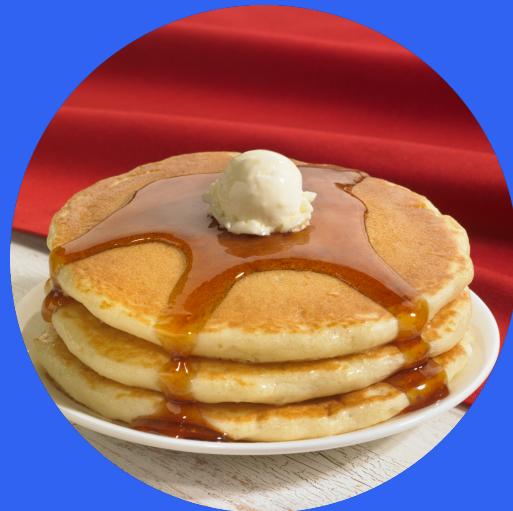
Kibana allows you to analyze and visualize data in Elasticsearch.

These visualizations can be grouped into dashboards.

Elasticsearch, Logstash and Kibana



Elasticsearch, Logstash and Kibana

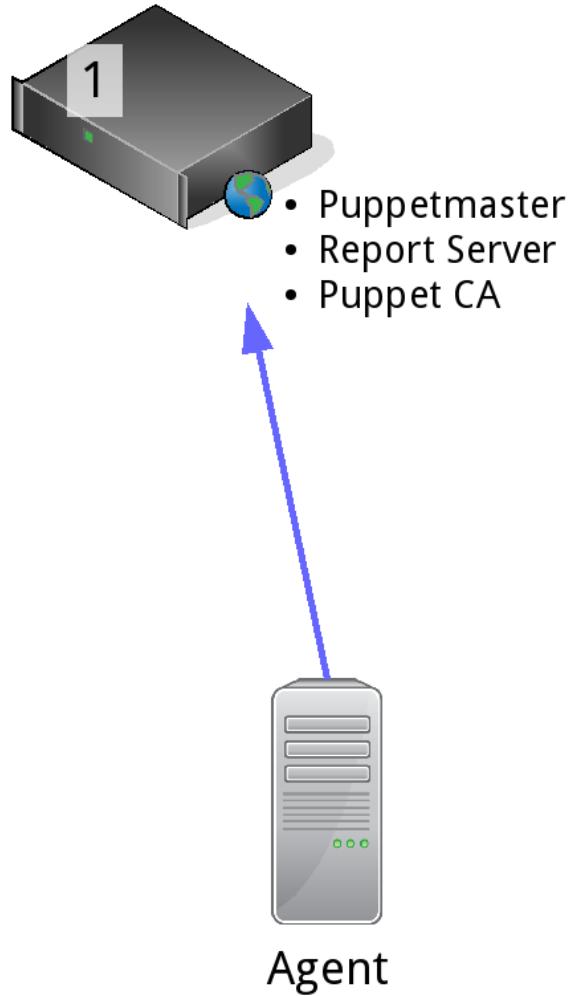


“ELK
STACK”

Getting Puppet reports into Elasticsearch

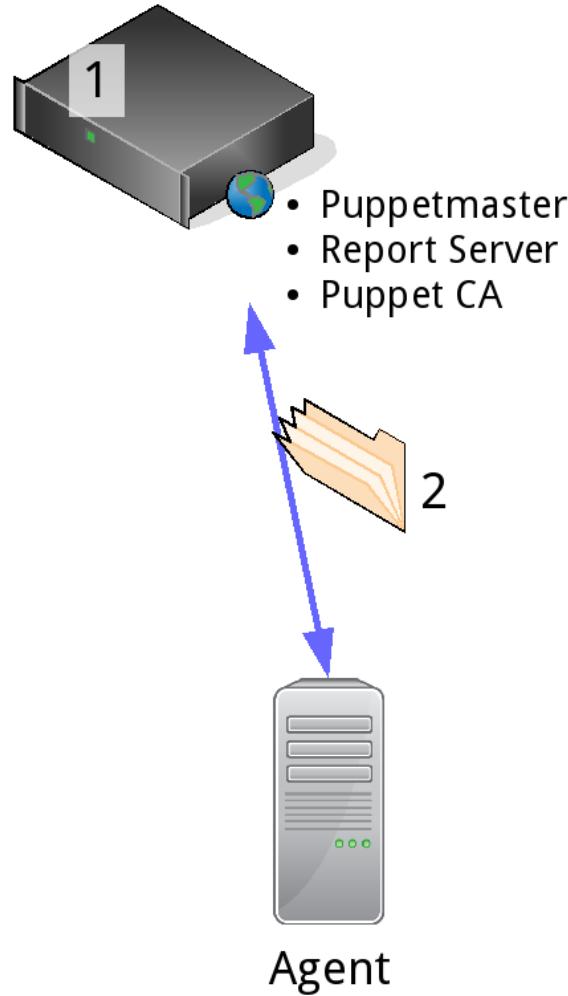
Getting reports into Elasticsearch

1. Agent requests catalog,
Master compiles catalog



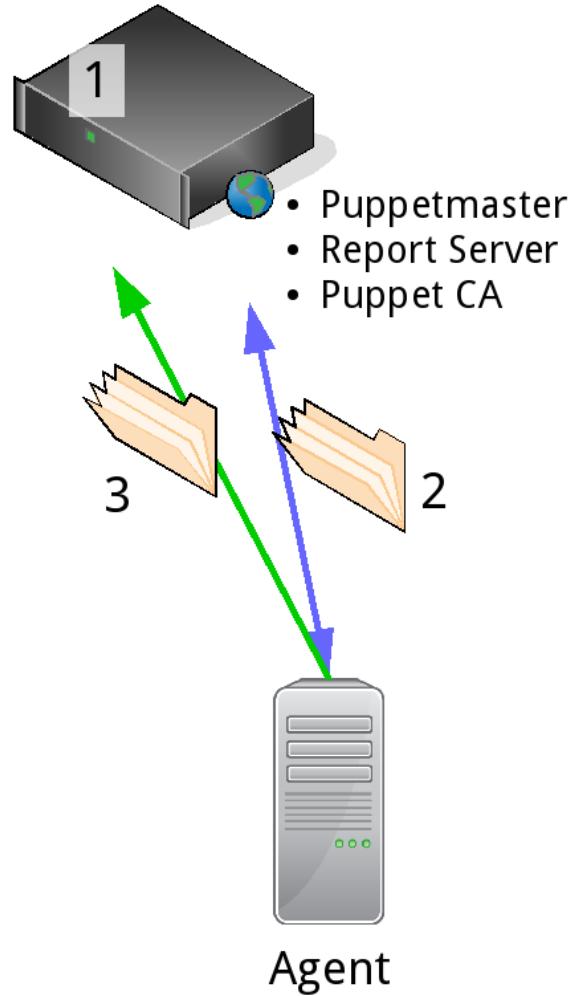
Getting reports into Elasticsearch

1. Agent requests catalog,
Master compiles catalog
2. Catalog is sent to Agent
which applies it



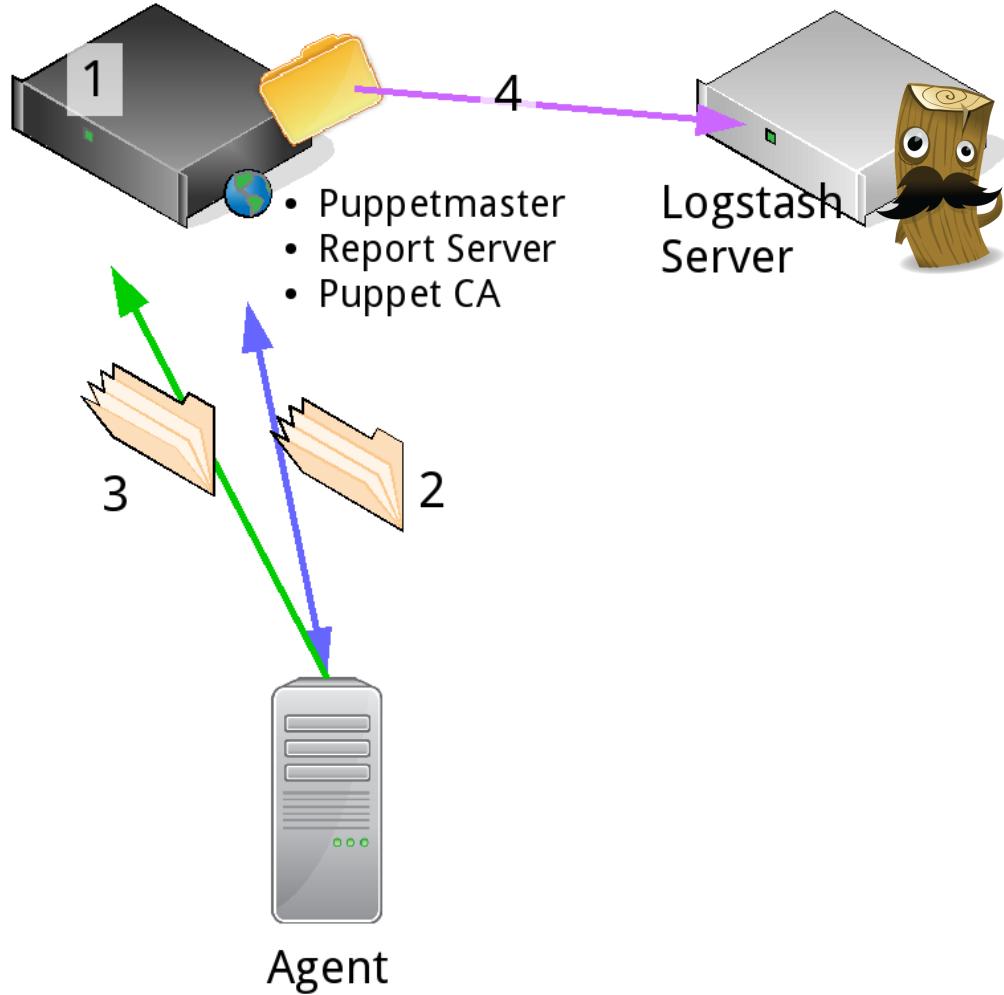
Getting reports into Elasticsearch

1. Agent requests catalog,
Master compiles catalog
2. Catalog is sent to Agent
which applies it
3. Run logs and metrics sent to
Report Server from Agent



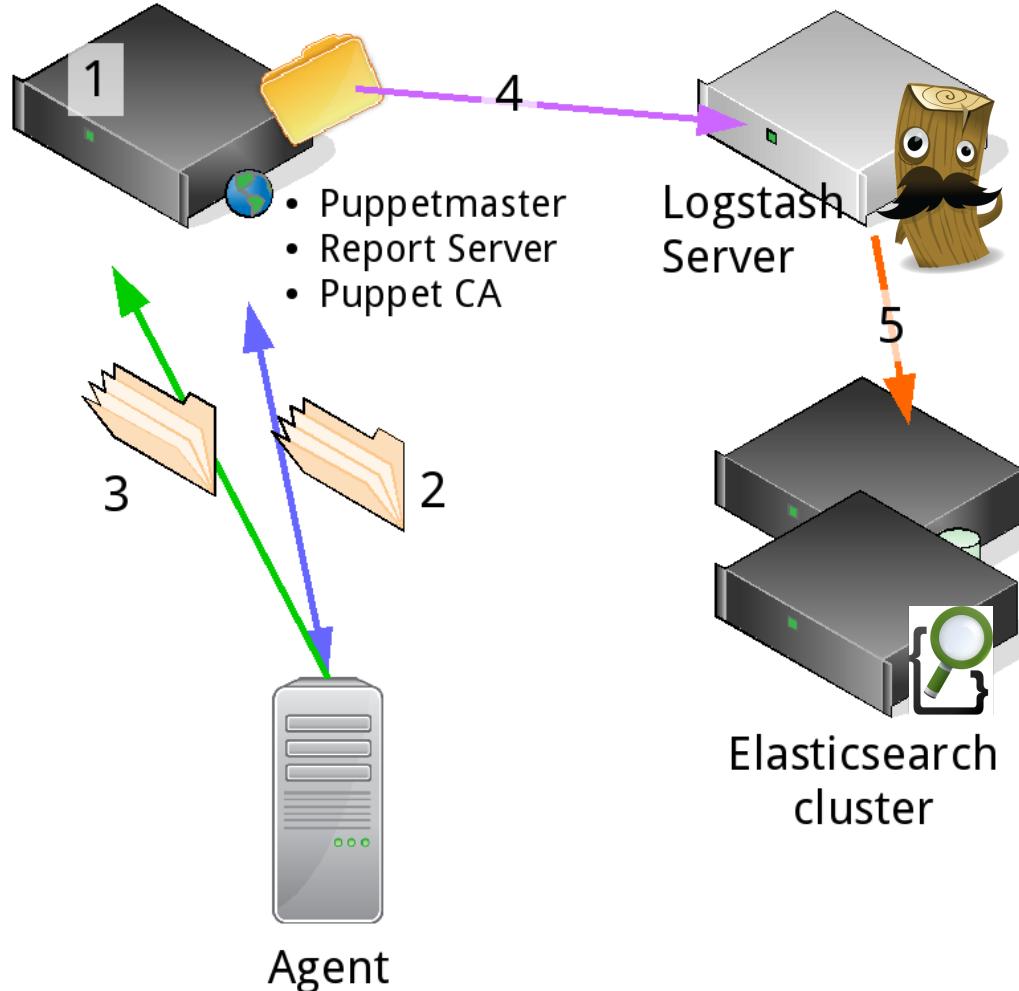
Getting reports into Elasticsearch

1. Agent requests catalog,
Master compiles catalog
2. Catalog is sent to Agent
which applies it
3. Run logs and metrics sent to
Report Server from Agent
4. Report Server sends report to
Logstash server as json



Getting reports into Elasticsearch

1. Agent requests catalog,
Master compiles catalog
2. Catalog is sent to Agent
which applies it
3. Run logs and metrics sent to
Report Server from Agent
4. Report Server sends report to
Logstash server as json
5. Elasticsearch indexes report



Installation

- Install Elasticsearch and Logstash
 - Place on separate servers for scaling
- Install the puppet-logstash-reporter
 - <https://github.com/elastic/puppet-logstash-reporter>
 - Configure the master and agent configs
 - Setup logstash host and port
- Install Kibana
 - Setup a proxying webserver to proxy requests
 - Tie into existing authentication system
 - Terminate SSL at the webserver

Master configuration

```
[master]
report = true
reports = logstash
pluginsync = true
```

Agent configuration

```
[agent]
report = true
pluginsync = true
```

/etc/puppet/logstash.yaml

```
---
:host: puppetreports.logstashserver.net
port: 5603
```

Maintenance

- Prune retired nodes from Elasticsearch
 - Easily becomes an automatic decommissioning task
 - Dead nodes become cruft and add noise
- Retire old indices or set a TTL on documents
- Keep everything up-to-date
- Monitor and alert on uptime of report infrastructure
- Be mindful of report format changes

What's in a Puppet Report?

- A resource describes some aspect of a system such as a package or a user
- An event is a change that happens to a resource
- Metrics are statistics generated about resources such as apply duration

```
status: failed
environment: production
kind: apply
report_format: 4
time: 2015-09-24 04:18:57.464790 -05:00
version:
  config: "master=puppetmaster
environment=production c8f9242 @ 2015-09-24 04:19:06
(-0500)"
  puppet: "3.8.0"
metrics:
  changes: !ruby/object:Puppet::Util::Metric
    total: 0
  resources: !ruby/object:Puppet::Util::Metric
    changed: 0
    failed_to_restart: 0
    out_of_sync: 8
    skipped: 7
    failed: 1
    total: 1408
  time: !ruby/object:Puppet::Util::Metric
    file: 6.655260
    package: 397.245294
    config_retrieval: 43.456929
    user: 0.098480
    group: 0.008287
    service: 10.116711
    exec: 10.337963
    firewall: 0.642523
    total: 471.79689
  events: !ruby/object:Puppet::Util::Metric
    total: 8
    noop: 7
    success: 0
    failure: 1
  puppet_version: "3.8.0"
  host: devserver4.dev
```

Aggregate
resource times

Aggregate Resource Times

```
file: 6.655260
package: 397.245294
config_retrieval: 43.456929
user: 0.098480
group: 0.008287
service: 10.116711
exec: 10.337963
firewall: 0.642523
total: 471.79689
```

Aggregate Resource Times

Whoa! We'll
be analyzing
this a bit later.

```
file: 6.655260
package: 397.245294
config_retrieval: 43.456929
user: 0.098480
group: 0.008287
service: 10.116711
exec: 10.337963
firewall: 0.642523
total: 471.79689
```

config_version

```
version:  
  config: "master=puppetmaster  
environment=production c8f9242 @  
2015-09-24 04:19:06 (-0500)"
```

- config_version is a Puppet variable that is populated by a script
- Uniquely identifies a catalog
 - Usually associated with a revision in your revision-controlled puppet repo
- Useful in troubleshooting failures and verifying changes
- Test for idempotency
 - comparing reports with same config_version

Tools we're open sourcing

- **puppet-show.py**
 - Answers many questions about Puppet runs by querying elasticsearch.
- **puppet-disof.py**
 - Did the last catalog run or resource succeed or fail?
- **puppet-blame.py**
 - A tool for identifying commits that broke Puppet.

puppet-show.py

A script for querying Elasticsearch

- Unresponsive nodes
 - Cross reference to find decommissioned nodes?
- Catalog compilation failures
- Idempotency violations
- Nodes that have been failing the longest
- Nodes with the most failures
- Resource responsible for the most recent failures

Why might a node go “unresponsive”?

- Puppet daemon isn't running
- Stale lockfile or pidfile prevent agent startup
- Master not sending node's report
- Catalog hasn't compiled in a long time
- Misconfiguration
- Linux OOMKiller is killing Puppet

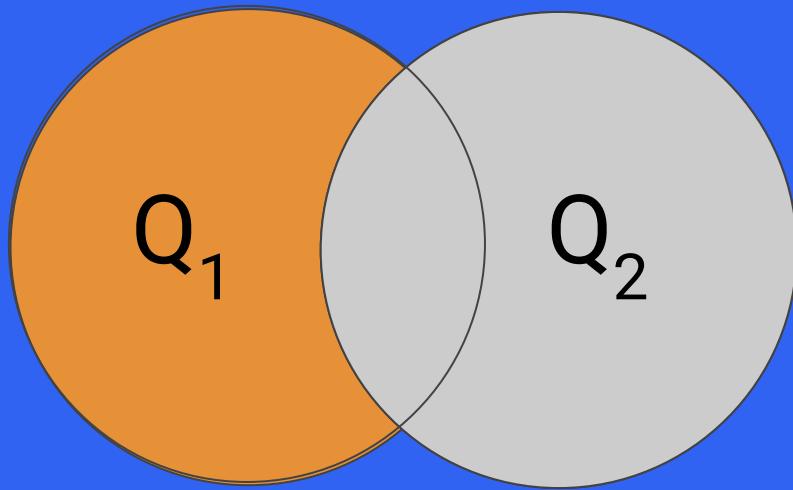
Q₁: Unique host aggregation

```
"size":0,  
"aggs": {  
    "by_hosts": {  
        "terms": {  
            "size": 0,  
            "field": "host"  
        }  
    }  
},
```

Q₂: Set a range over the time threshold

```
"query" : {  
    "range" : {  
        "@timestamp": {  
            "gt": "now-60m"  
        }  
    },  
},
```

Unresponsive nodes: First set less the second set



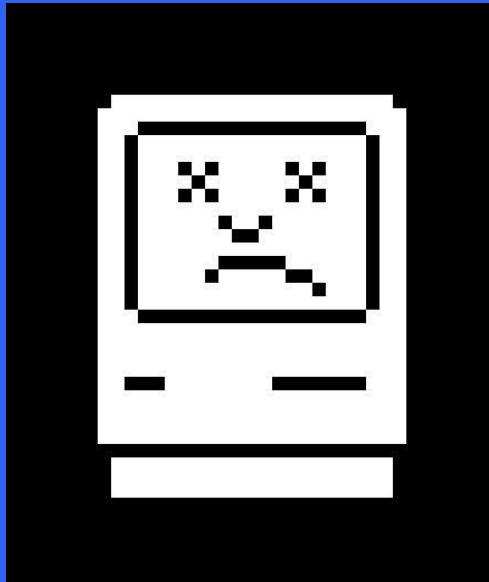
$$Q_1 / Q_2$$

Query for unresponsive nodes

```
$> python puppet-show.py --unresponsive 60

==== sfo-dev1.myserver.net exceeded 60 minutes and last
submitted report at 2015-08-29 21:28:07 ====
==== sfo-tools2.myserver.net exceeded 60 minutes and last
submitted report at 2015-09-01 21:42:24 ====
==== pdx-puppet1.myserver.net exceeded 60 minutes and last
submitted report at 2015-09-12 11:41:37 ====
==== atx-foreman.myserver.net exceeded 60 minutes and last
submitted report at 2015-09-16 17:10:55 ====
```

If the catalog didn't
compile how do we know?



Could not retrieve catalog from remote
server: Error 400 on SERVER

Algorithm

1. Perform query aggregation for unique hosts

Algorithm

1. Perform query aggregation for unique hosts
2. Iterate over hosts retrieving newest doc per host

Algorithm

1. Perform query aggregation for unique hosts
2. Iterate over hosts retrieving newest doc per host
3. Check newest catalog run status is a failure

Algorithm

1. Perform query aggregation for unique hosts
2. Iterate over hosts retrieving newest doc per host
3. Check newest catalog run status is a failure
4. If failure retrieve index of “Could not retrieve catalog” in logs field

Algorithm

1. Perform query aggregation for unique hosts
2. Iterate over hosts retrieving newest doc per host
3. Check newest catalog run status is a failure
4. If failure retrieve index of “Could not retrieve catalog” in logs field
5. Populate associative array where key is hash of failure message

Query for unresponsive nodes

```
$> python puppet-show.py --catalog-failure
```

```
=====
```

```
Could not retrieve catalog from remote server: Error 400  
on SERVER: lv_size for disk01 must end in M, G or T at  
/var/lib/puppet/production/manifests/filesystem.pp:91  
COUNT: 3
```

```
NODES AFFECTED:
```

```
sfo-doc1.myserver.net on 2015-09-07 08:11:29
```

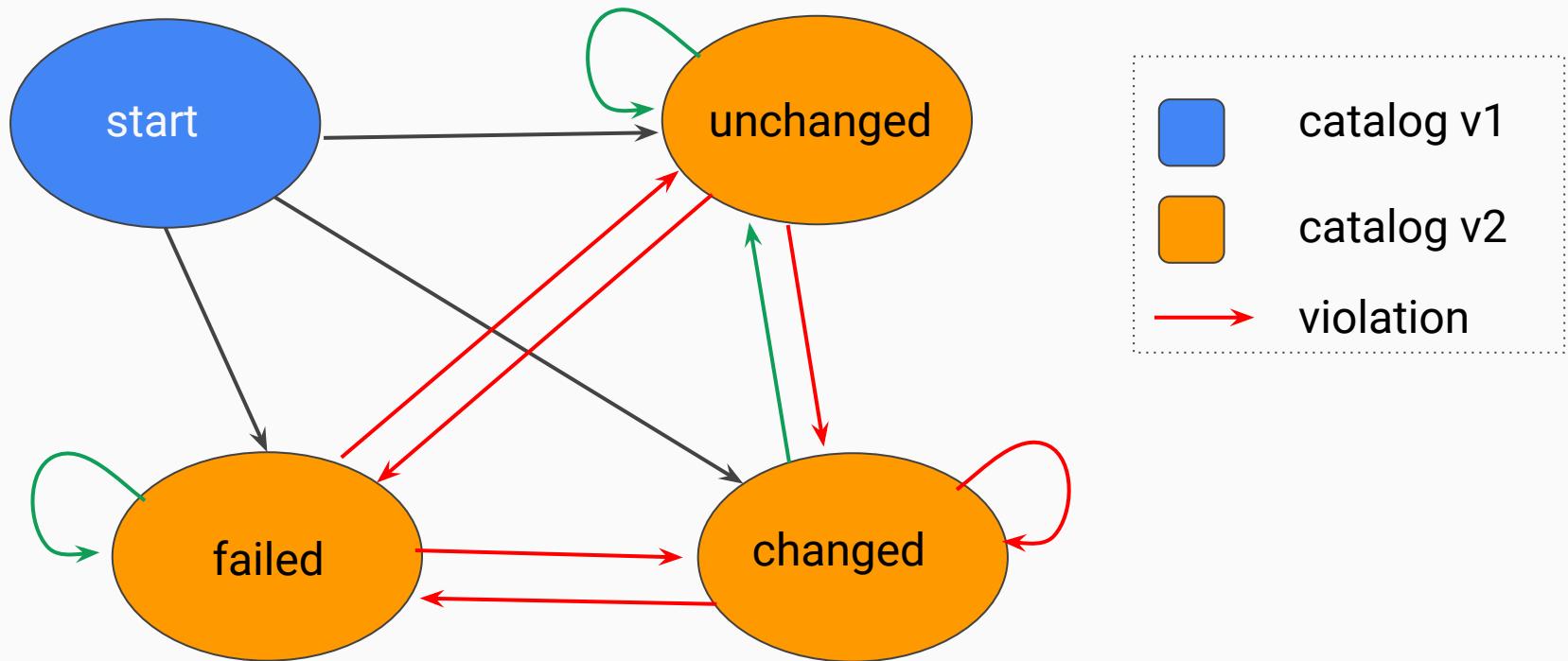
```
atx-log3.myserver.net on 2015-09-07 06:21:25
```

```
atx-dev2.myserver.net on 2015-09-08 15:23:34
```

Checking for idempotence

- Running puppet twice against the same catalog should yield the same result
- When is idempotence violated?

Idempotence state transition diagram



Idempotence violation matrix

First run status		Second run status
changed	→	changed
changed	→	failed
unchanged	→	failed
unchanged	→	changed
failed	→	unchanged
failed	→	changed

Algorithm

1. Perform query aggregation for unique hosts
2. Query each host for config_version and status
3. Group documents by identical config_version
4. Pairwise compare catalog status transitions temporally
5. Check for transition violations

Making the query

```
{  
  "fields" : ["configuration_version", "status"],  
  "query": {  
    "bool": {  
      "must": [  
        { "match": { "host": "node1.myserver.net" } },  
        { "match": { "environment": "production" } }  
      ]  
    }  
  },  
  "sort": [{"@timestamp":{"order":"desc"}}]  
}
```

Interrogating Puppet runs with Kibana

Profiling Puppet Performance



Time Filter

Refresh Interval

Quick

Relative

Absolute

Today

This week

This month

This year

The day so far

Week to date

Month to date

Year to date

Yesterday

Day before yesterday

This day last week

Previous week

Previous month

Previous year

Last 15 minutes

Last 30 minutes

Last 1 hour

Last 4 hours

Last 12 hours

Last 24 hours

Last 7 days

Last 30 days

Last 60 days

Last 90 days

Last 6 months

Last 1 year

Last 2 years

Last 5 years



Profiling Puppet Performance

- How many unique resources do we have?

The screenshot shows a user interface for monitoring Puppet performance. At the top, there's a dark bar with a hand icon holding a smartphone. Below it, a light gray header bar says "metrics". Underneath, a dropdown menu is open, showing "Metric" and "Aggregation". The "Aggregation" section contains a dropdown menu set to "Unique count". Below this, there's a "Field" section with a dropdown menu containing "resource_statuses.title". To the right of the field dropdown, there's an orange warning icon with the text "⚠ Analyzed Field". At the bottom of the interface, there's a dark bar with a back arrow and the word "Advanced".

Profiling Puppet Performance

- How many unique resources do we have?

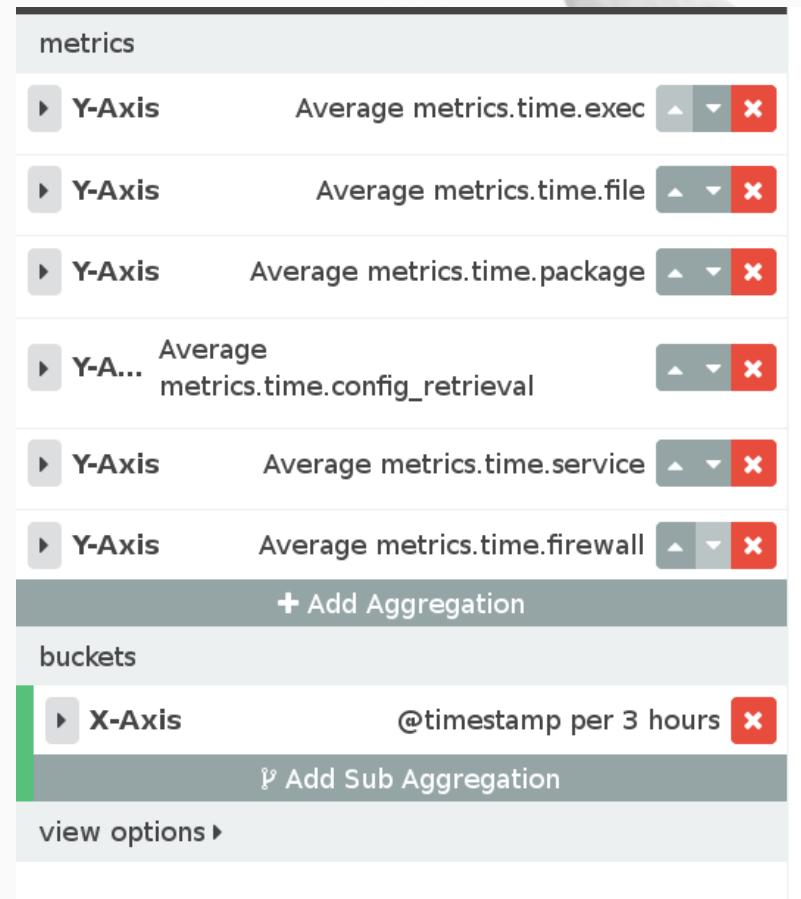


6095

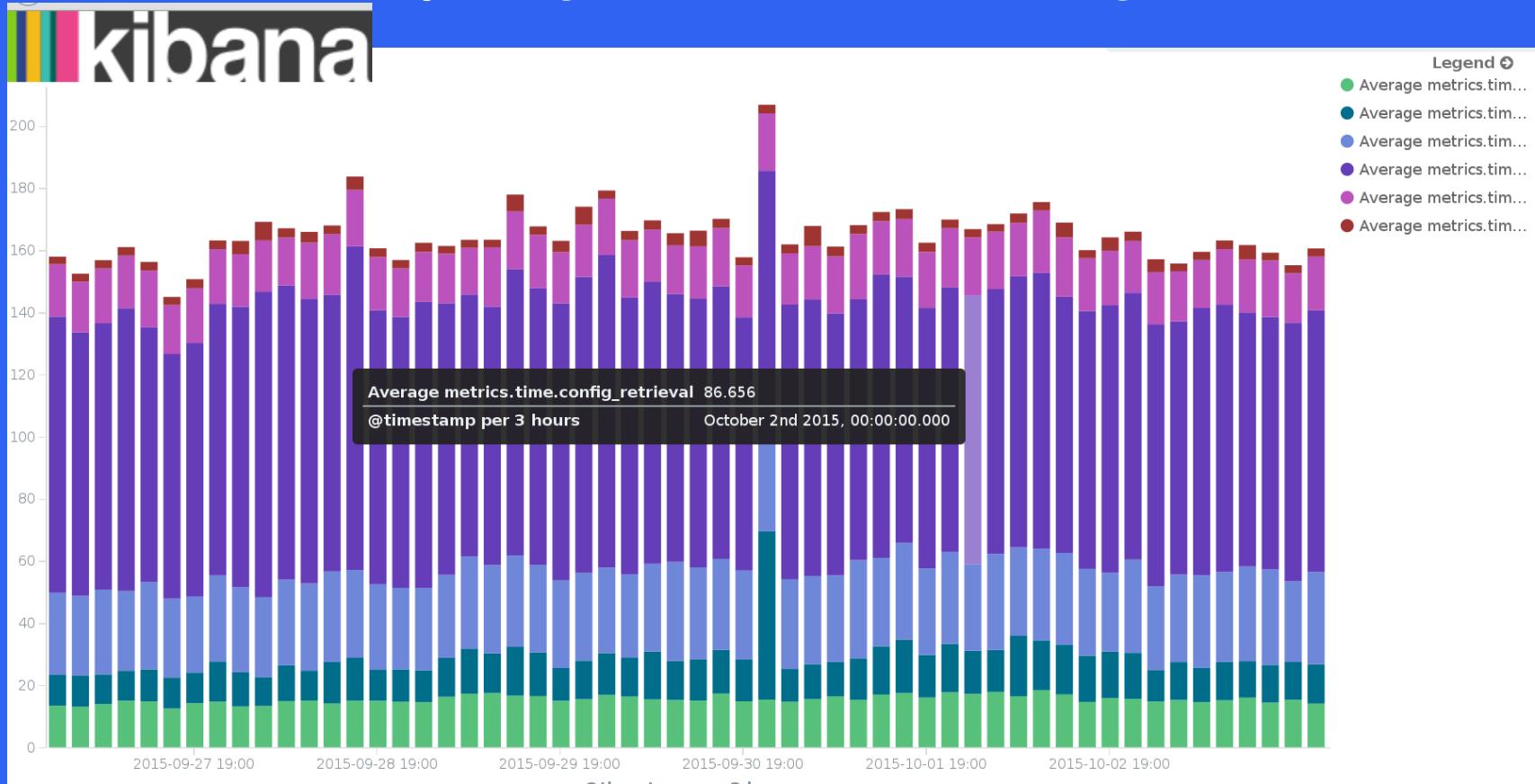
Unique count of `resource_statuses.title`

Profiling Puppet Performance

- How many unique resources do we have?
- What are the nodes are spending the most time doing?



Analyzing Resource Timings



Profiling Puppet Performance

- How many unique resources do we have?
- What are the nodes are spending the most time doing?
- Which nodes have the most resource out of compliance?

metrics

Y-Axis

Aggregation

Unique count

Field ⚠ Analyzed Field

resource statuses.title

Advanced

+ Add Aggregation

buckets

X-Axis ✖

Aggregation

Terms

Field ⚠ Analyzed Field

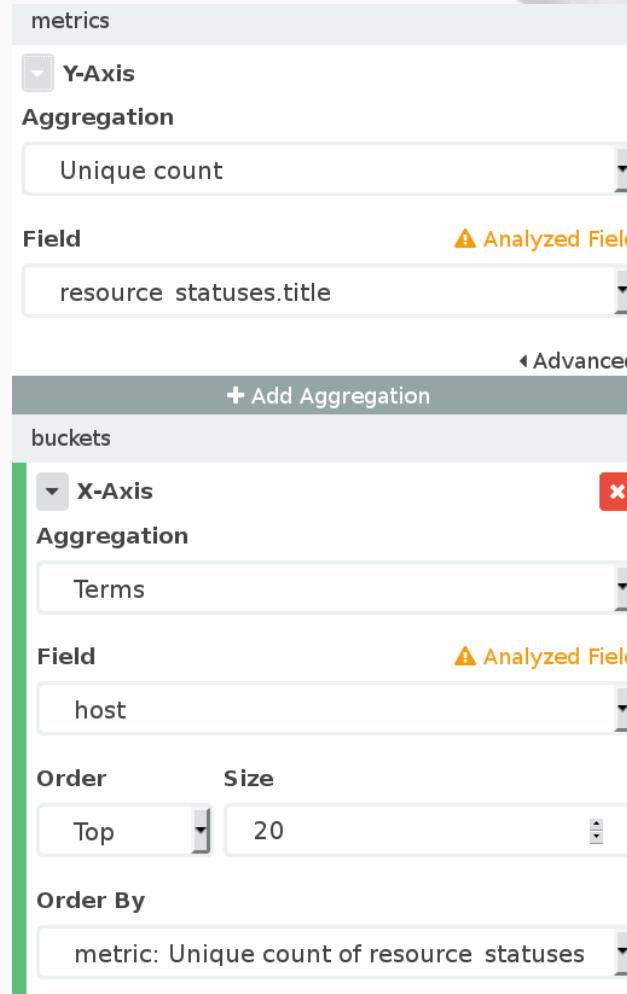
host

Order Size

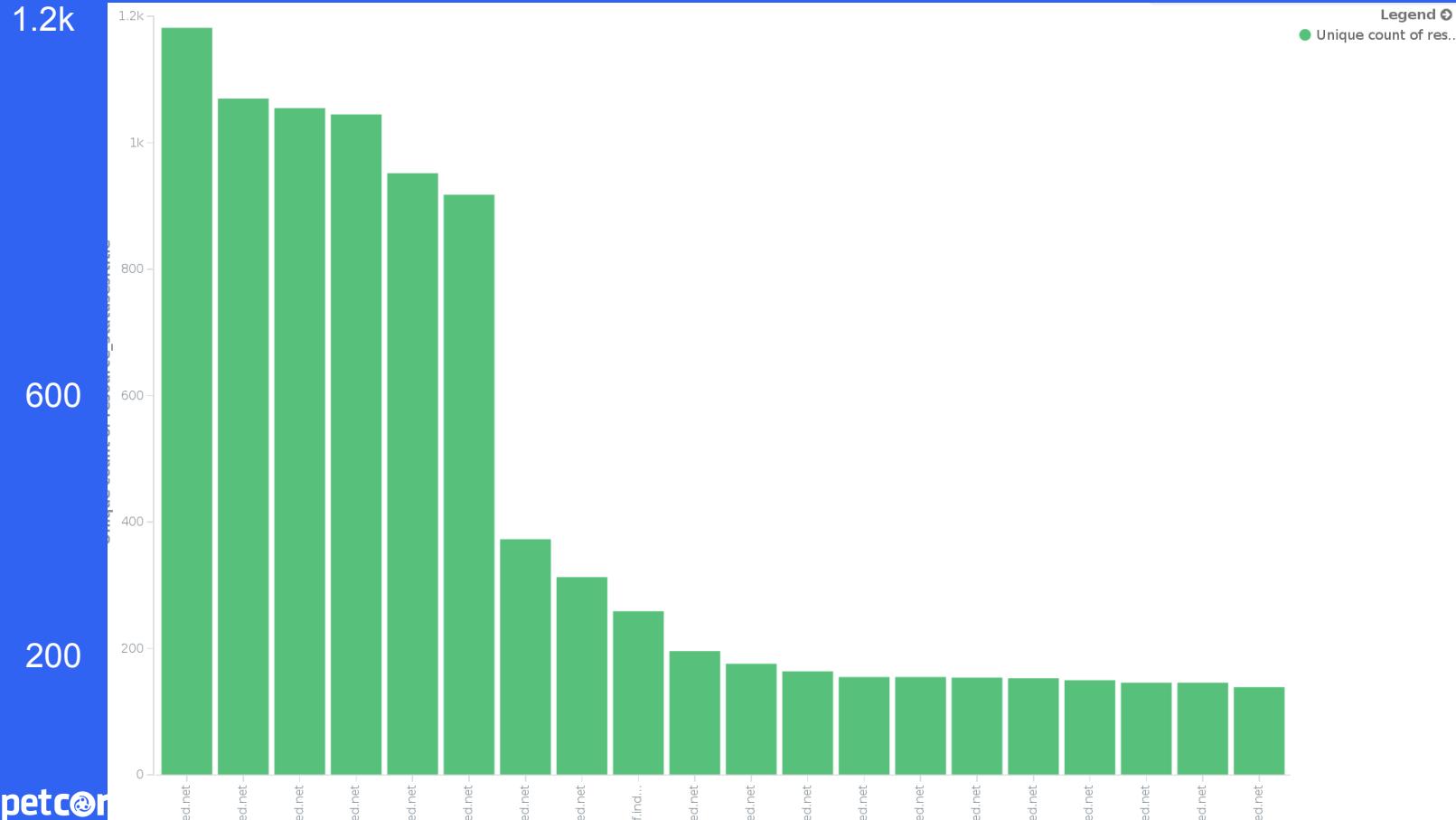
Top 20

Order By

metric: Unique count of resource statuses

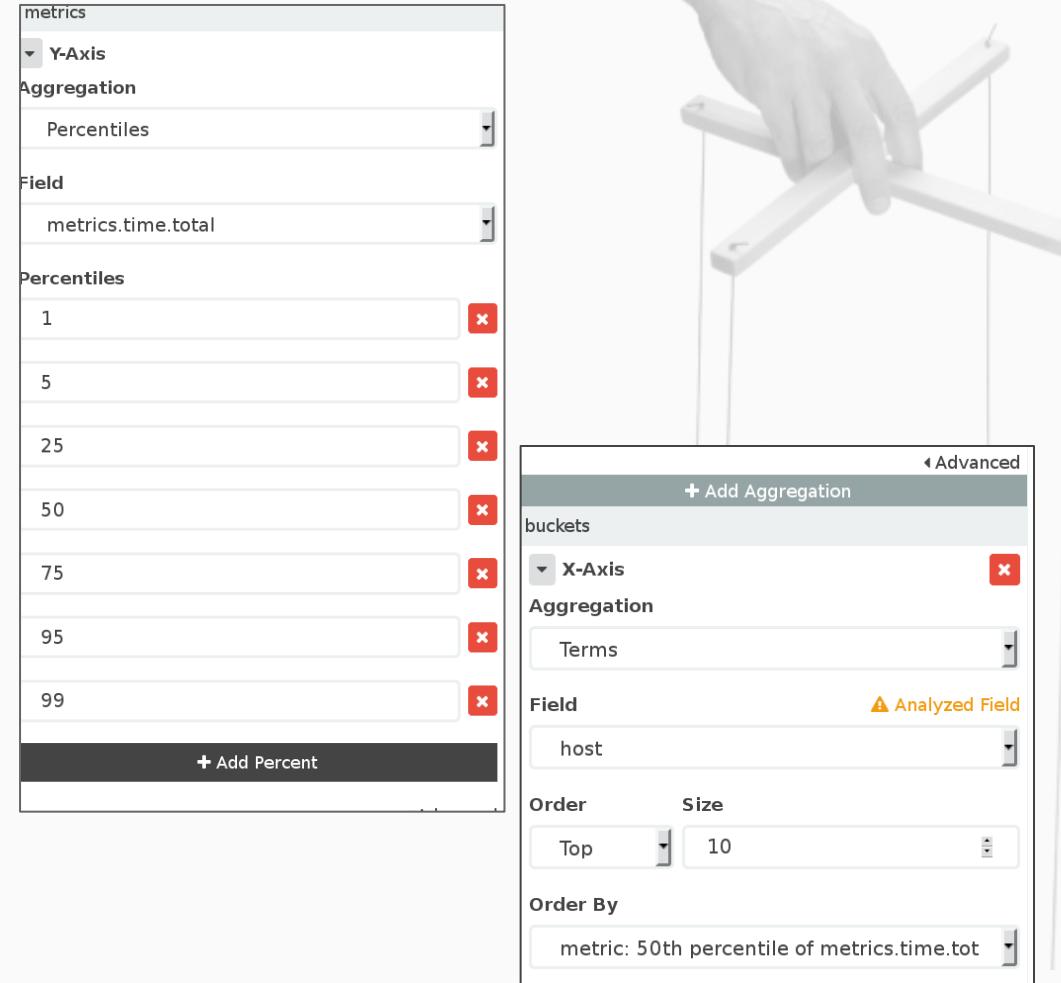


Number of resources out of compliance by host

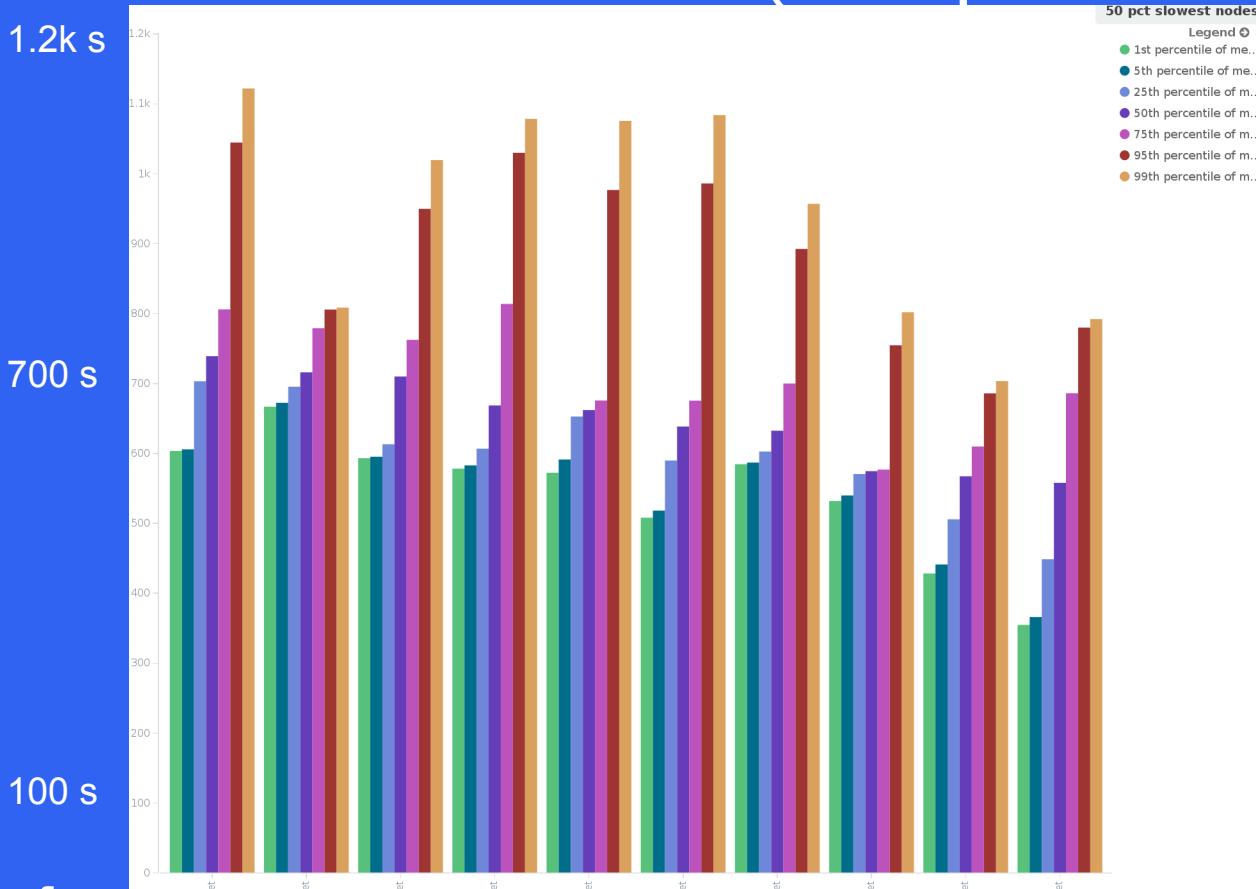


Profiling Puppet Performance

- How many unique resources do we have?
- What are the nodes are spending the most time doing?
- Which nodes have the most resource out of compliance?
- What resource is responsible for the most failures?

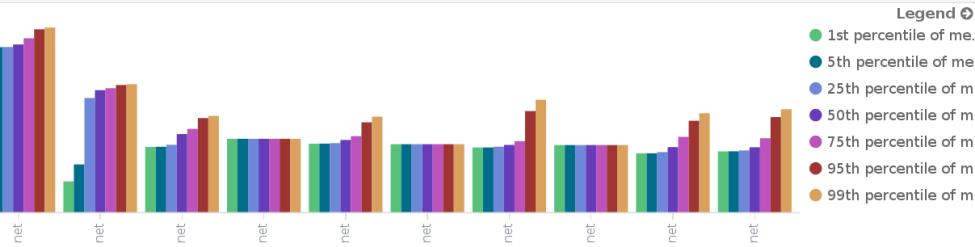


Slowest nodes (50% percentile)

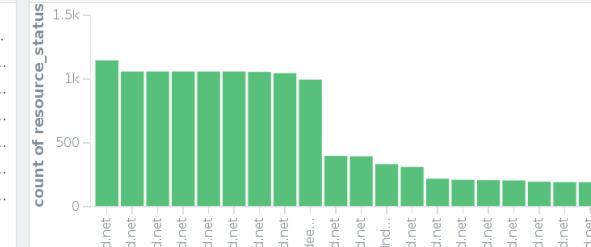


Dashboards: Putting it all together

50 pct slowest nodes



Most out-of-sync resources by host



Top 10 host

Nodes responding in the last 24 hours

1389

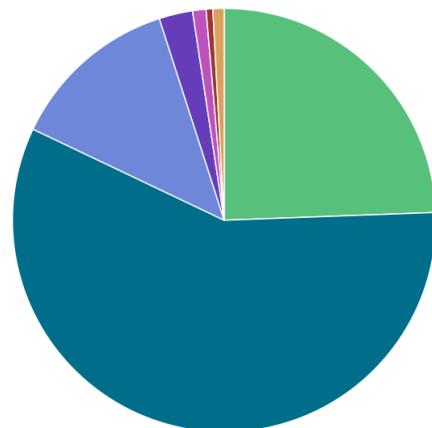
Unique count of host

Number of unique hosts

1552

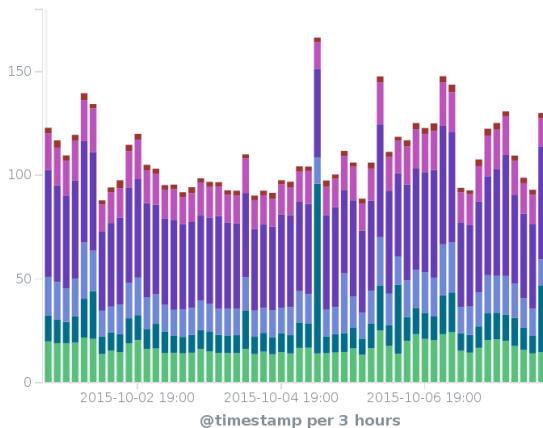
Unique count of host

Percentages of total time in seconds to apply the catalog



Top 20 host

Relative times it takes to apply resources



Did Puppet succeed or fail?

puppet-disof.py: did I succeed or fail?

- If the last run was a failure, puppet-disof.py will print the log output of the failure

```
echo -e "sfo-puppet1.myserver.net" | python puppet-disof.py

===== sfo-puppet1.myserver.net failed 2015-09-20T21:24:36.000-05:00 =====

Failed Nodes:
  sfo-puppet1.myserver.net:

    Could not update: Execution of '/usr/bin/yum -d 0 -e 0 -y install curl' returned 1: Error: Package: libcurl-devel-7.19.7-46.el6.x86_64 (centos6)

      Requires: libcurl = 7.19.7-46.el6
      You could try using --skip-broken to work around the problem
      You could try running: rpm -Va --nofiles --nodigest
    Wrapped exception:
      Execution of '/usr/bin/yum -d 0 -e 0 -y install curl' returned 1: Error: Package: libcurl-devel-7.19.7-46.el6.x86_64 (centos6)
```

puppet-disof.py: did I succeed or fail?

- The --summary flag will simply summarize the status of the last Puppet run.

```
echo -e "sfo-prod1.myserver.net\nsfo-prod2.  
myserver.net\pdx-tools1.myserver.net\pdx-mongo1.  
myserver.net" | python puppet-disof.py --summary  
===== sfo-prod1.myserver.net succeeded 2015-09-  
18T06:59:55.000-05:00 =====  
===== sfo-prod2.myserver.net failed 2015-09-20T12:  
21:51.000-05:00 =====  
===== pdx-tools1.myserver.net failed 2015-09-20T12:  
33:41.000-05:00 =====  
===== pdx-mongo1.myserver.net failed 2015-09-20T12:  
35:51.000-05:00
```

Failed Nodes:

```
sfo-prod2.myserver.net  
pdx-tools1.myserver.net  
pdx-mongo1.myserver.net
```

Using Mcollective

- Server orchestration framework
- Built into Puppet Enterprise
- Can be used on the command line to discover nodes across certain Facter facts
- Helps us build and refine our queries for Elasticsearch

Story time!

An upgrade to an NRPE script that checks RAID volumes became available and was applied everywhere.

That upgrade didn't go so well...

PROBLEM: A new version of NRPE command check-raid has a bug in it causing Nagios to report false positives

Before:

```
$> check_nrpe -H sfo-prod2.myserver.net -c check-raid  
Logical Device 0 Optimal,Controller Optimal
```

After:

```
$> check_nrpe -H sfo-prod2.myserver.net -c check-raid  
No output from aarccconf!
```

Don't worry, we have a patch that fixes the upgrade. Let's push this out with Puppet!

An hour later
pagers are still
going off...



Puppet, you had
one job!

Mcollective + puppet-disof.py

This runs the mcollective NRPE plugin against all nodes that have the has_raid fact set.

```
$> sudo mc-nrpe -S has_raid=true check-raid | grep myserver.net | awk '{ print $1 }' | python puppet-disof.py --resource check_raid

===== Nrpe[check_raid] succeeded on sfo-prod1.myserver.net at 2015-09-18T06:59:55 =====

===== Nrpe[check_raid] failed on sfo-prod2.myserver.net at 2015-09-20T12:21:51 =====

===== Nrpe[check_raid] failed on pdx-tools1.myserver.net at 2015-09-20T12:33:41 =====

===== Nrpe[check_raid] failed on pdx-mongo1.myserver.net at 2015-09-20T12:35:51 =====
```

Mcollective + puppet-disof.py

mc-nrpe will run check-raid and output only the server names failing the check. Here we massage the output into a newline separated list

```
$> sudo mc-nrpe -S has_raid=true check-raid | grep myserver.net | awk '{ print $1 }' | python puppet-disof.py --resource check_raid

===== Nrpe[check_raid] succeeded on sfo-prod1.myserver.net at 2015-09-18T06:59:55 =====

===== Nrpe[check_raid] failed on sfo-prod2.myserver.net at 2015-09-20T12:21:51 =====

===== Nrpe[check_raid] failed on pdx-tools1.myserver.net at 2015-09-20T12:33:41 =====

===== Nrpe[check_raid] failed on pdx-mongo1.myserver.net at 2015-09-20T12:35:51 =====
```

Mcollective + puppet-disof.py

Of the nodes with
false-positives what is
the status of Nrpe
[check_raid] resource
during their last
Puppet run?

```
$> sudo mc-nrpe -S has_raid=true check-raid | grep  
myserver.net | awk '{ print $1 }' | python puppet-  
disof.py --resource check_raid  
  
===== Nrpe[check_raid] succeeded on sfo-prod1.  
myserver.net at 2015-09-18T06:59:55 =====  
  
===== Nrpe[check_raid] failed on sfo-prod2.myserver.  
net at 2015-09-20T12:21:51 =====  
  
===== Nrpe[check_raid] failed on pdx-tools1.myserver.  
net at 2015-09-20T12:33:41 =====  
  
===== Nrpe[check_raid] failed on pdx-mongo1.myserver.  
net at 2015-09-20T12:35:51 =====
```

Mcollective + puppet-disof.py

The resource during the last Puppet run on sfo-prod1 was successful but notice that there hasn't been a run in several days.

```
$> sudo mc-nrpe -S has_raid=true check-raid | grep myserver.net | awk '{ print $1 }' | python puppet-disof.py --resource check_raid

===== Nrpe[check_raid] succeeded on sfo-prod1.myserver.net at 2015-09-18T06:59:55 =====
===== Nrpe[check_raid] failed on sfo-prod2.myserver.net at 2015-09-20T12:21:51 =====
===== Nrpe[check_raid] failed on pdx-tools1.myserver.net at 2015-09-20T12:33:41 =====
===== Nrpe[check_raid] failed on pdx-mongo1.myserver.net at 2015-09-20T12:35:51 =====
```

Mcollective + puppet-disof.py

These three nodes all have recent Puppet runs but the resources are failing to be applied.

```
$> sudo mc-nrpe -S has_raid=true check-raid | grep myserver.net | awk '{ print $1 }' | python puppet-disof.py --resource check_raid

===== Nrpe[check_raid] succeeded on sfo-prod1.myserver.net at 2015-09-18T06:59:55 =====

===== Nrpe[check_raid] failed on sfo-prod2.myserver.net at 2015-09-20T12:21:51 =====

===== Nrpe[check_raid] failed on pdx-tools1.myserver.net at 2015-09-20T12:33:41 =====

===== Nrpe[check_raid] failed on pdx-mongo1.myserver.net at 2015-09-20T12:35:51 =====
```

```
Error: /Stage[main]/Main/Resources[firewall]: Failed to generate additional resources using 'generate': Execution of '/sbin/ip6table
Info: Applying configuration version 'master=          /ironment=production f7b75d9 @ 2015-09-26 18:46:43 (-0500)'
Notice: /Stage[main]/Main/Nagios_service[LogRepoReader lag prod archive]/ensure: removed
Error: Could not prefetch package provider 'pip': #<Puppet::ExecutionFailure:0x2ad5811f6b80>
Error: /Package[setuptools]: Could not evaluate: #<Puppet::ExecutionFailure:0x2ad5811e38f0>
Error: /Package[pip]: Could not evaluate: #<Puppet::ExecutionFailure:0x2ad5811a4880>
Error: /Package[pyelasticsearch]: Could not evaluate: #<Puppet::ExecutionFailure:0x2ad5808effc8>
Error: Could not start Service[rsyslog]: Execution of '/sbin/service rsyslog start' returned 1: Shut down sysklog before you run rs
Wrapped exception:
Execution of '/sbin/service rsyslog start' returned 1: Shut down sysklog before you run rsyslog
Error: /Stage[main]/Rsyslog::Service/Service[rsyslog]/ensure: change from stopped to running failed: Could not start Service[rsyslog]
returned 1: Shut down sysklog
Notice: /Stage[main]/Nagios::S
Info: /Stage[main]/Nagios::Se
Notice: /Stage[main]/Base/Pack
Error: /Package[meld3]: Could
Error: /Package[pymongo]: Cou
Notice: /Stage[main]/Nagios::S
s: true
Warning: /Stage[main]/Nagios::Scripts/Nagios::Install_Title[custom/ccheck_mongodb.py] File /usr/local/nagios/libexec/custom/check_mong
s
Error: /Package[elementtree]: Could not evaluate: #<Puppet::ExecutionFailure:0x2ad57f5afdc8>
Error: /Package[cryptography]: Could not evaluate: #<Puppet::ExecutionFailure:0x2ad57f3c3fa0>
Error: /Package[kazoo]: Could not evaluate: #<Puppet::ExecutionFailure:0x2ad57e3d5558>
Error: /Package[requests]: Could not evaluate: #<Puppet::ExecutionFailure:0x2ad57e325c70>
Error: /Package[klein]: Could not evaluate: #<Puppet::ExecutionFailure:0x2ad57e2b03f8>
Error: /Package[mock]: Could not evaluate: #<Puppet::ExecutionFailure:0x2ad57e283290>
Error: /Package[PyMySQL]: Could not evaluate: #<Puppet::ExecutionFailure:0x2ad57e21e610>
Error: /Package[trellis]: Could not evaluate: #<Puppet::ExecutionFailure:0x2ad57e0629e8>
Notice: /Stage[main]/Deploy/Package[deploytool]: Dependency Package[mock] has failures: true
Notice: /Stage[main]/Deploy/Package[deploytool]: Dependency Package[klein] has failures: true
Notice: /Stage[main]/Deploy/Package[deploytool]: Dependency Package[requests] has failures: true
Notice: /Stage[main]/Deploy/Package[deploytool]: Dependency Package[kazoo] has failures: true
Notice: /Stage[main]/Deploy/Package[deploytool]: Dependency Package[PyMySQL] has failures: true
Notice: /Stage[main]/Deploy/Package[deploytool]: Dependency Package[trellis] has failures: true
Warning: /Stage[main]/Deploy/Package[deploytool]: Skipping because of failed dependencies
Notice: /Stage[main]/Deploy/Package[deploytool2]: Dependency Package[mock] has failures: true
Notice: /Stage[main]/Deploy/Package[deploytool2]: Dependency Package[klein] has failures: true
Notice: /Stage[main]/Deploy/Package[deploytool2]: Dependency Package[requests] has failures: true
Notice: /Stage[main]/Deploy/Package[deploytool2]: Dependency Package[kazoo] has failures: true
Notice: /Stage[main]/Deploy/Package[deploytool2]: Dependency Package[PyMySQL] has failures: true
Notice: /Stage[main]/Deploy/Package[deploytool2]: Dependency Package[trellis] has failures: true
Warning: /Stage[main]/Deploy/Package[deploytool2]: Skipping because of failed dependencies
```

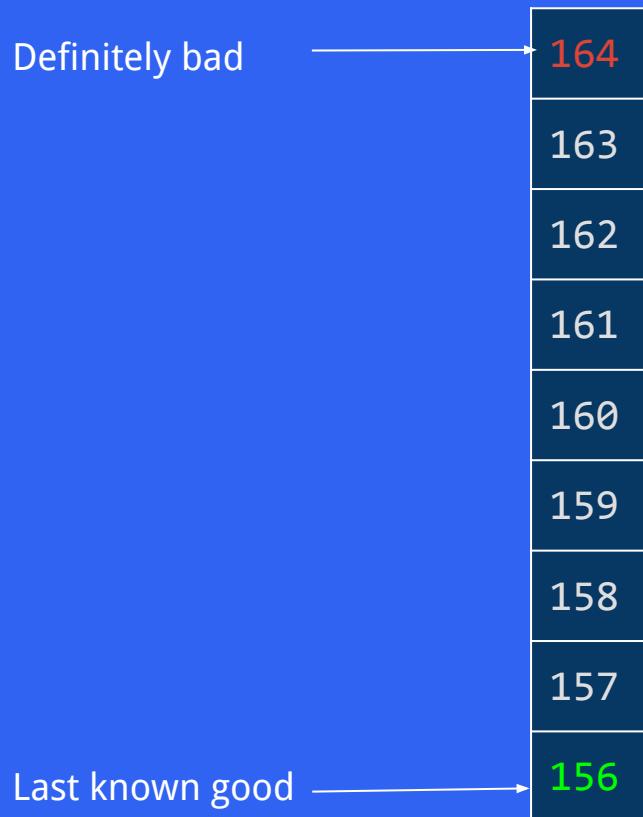
A bad commit escaped into production. Now what?

puppet-blame.py

A script for tracking down culpable commits

- Uses git bisect
- Requires git, setting configuration version, and ability compile catalog
- puppet-blame.py requires access to git repo
- Pass list of nodes on stdin

binary search with git bisect



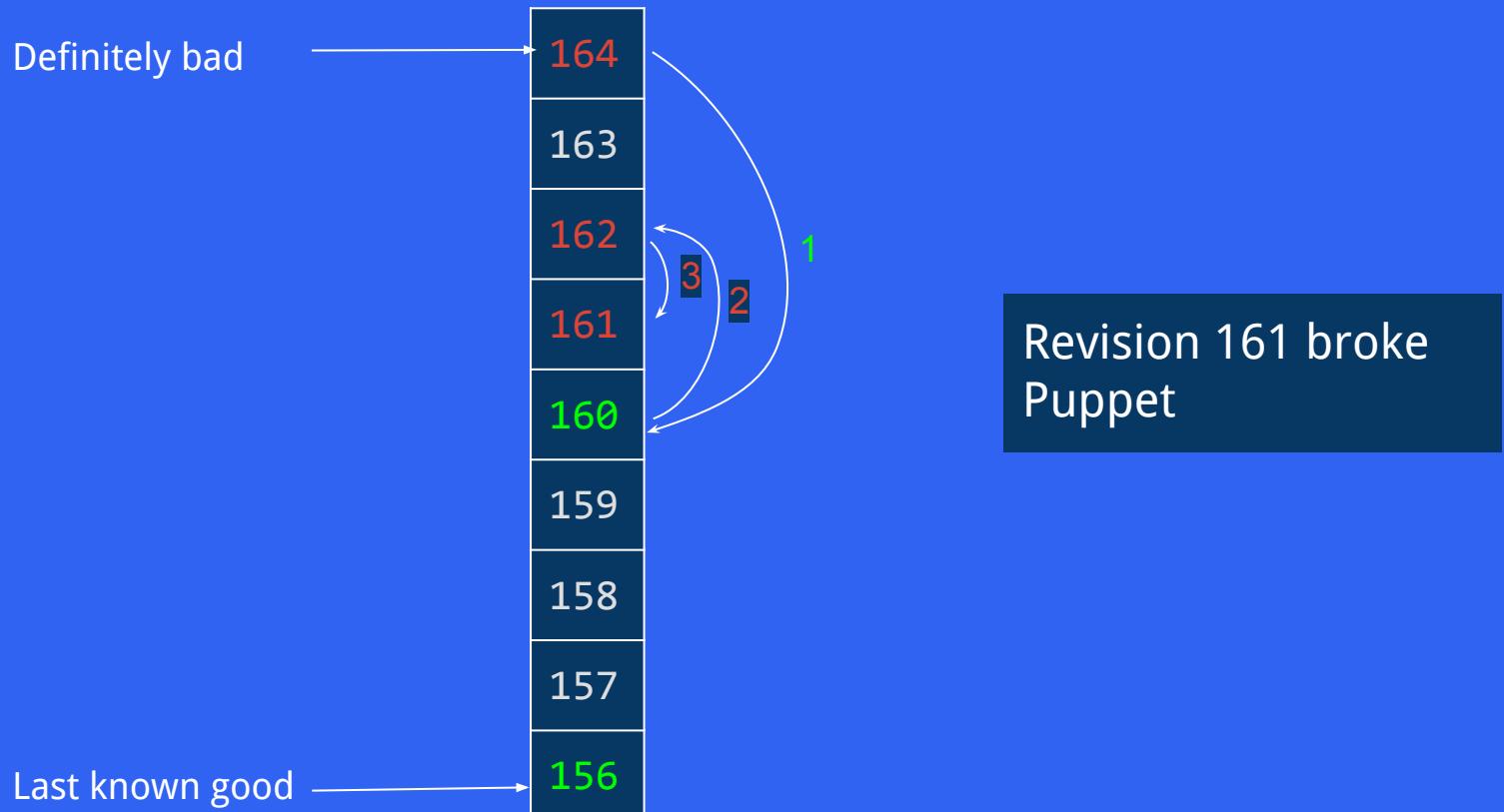
binary search with git bisect



binary search with git bisect



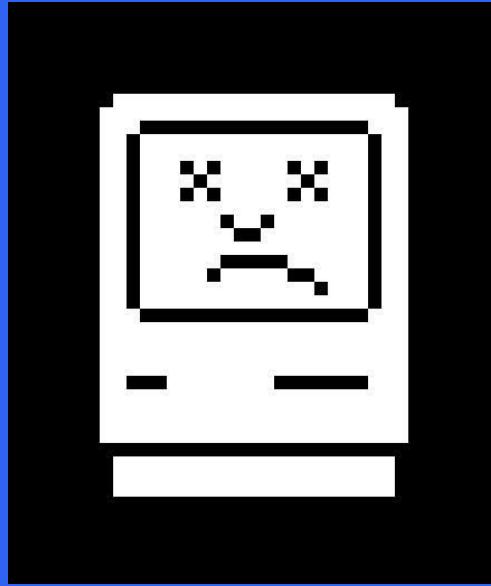
binary search with git bisect



Running interactively

- Accepts node name(s) on stdin
- Combines two datasets
 - git log
 - elasticsearch
- Queries Elasticsearch for version of last success and subsequent failure
- Wraps git-bisect interactively

```
$> cd puppet-git  
$> echo "dev-node2.dev" | python puppet-blame.py  
  
Oldest success: f1a658 @ Jun 6th  
First failure: 241b1f @ Jun 7th  
Running git bisect...  
  
Perform dry-run against this environment.  
Did the run succeed (s) or fail (f)? f  
  
Perform a dry-run against this environment.  
Did the run succeed (s) or fail (f)? f  
  
Perform a dry-run against this environment.  
Did the run succeed (s) or fail (f)? s  
  
The breaking change on dev-node2.dev was  
version: 6bc814 by Bad Committer
```



Could not retrieve catalog from remote server: Error 400 on SERVER



But the agent doesn't report what version of the catalog it tried to compile! Now what?

Could not retrieve catalog from remote server: Error 400 on SERVER

Running automatically

- If a node can't even compile its catalog, puppet-blame can run git-bisect on its own to find the bad commit
- We still can know:
 - last good revision
 - bad revision: `git log --before @timestamp -1 --pretty=format:%h`
- git-bisect can use a shell script that compiles the catalog on master
- shell script returns 0 for successful compile

No interaction required!

```
$> echo "node1.devserver.net" | python puppet-blame.py
```

```
Oldest success: 1a660b @ Sep 18th  
First failure: 8408cd @ Sep 18th
```

No interaction required!

```
$> echo "node1.devserver.net" | python puppet-blame.py
```

```
Oldest success: 1a660b @ Sep 18th  
First failure: 8408cd @ Sep 18th
```



No interaction required!

```
$> echo "node1.devserver.net" | python puppet-blame.py
```

```
Oldest success: 1a660b @ Sep 18th  
First failure: 8408cd @ Sep 18th
```

```
Bisection: 7 revisions left to test after this  
(roughly 3 steps)  
[6c0cba] Some commit message  
running puppet_compile.sh node1.devserver.net
```

No interaction required!

```
Bisection: 3 revisions left to test after this  
(roughly 2 steps)
```

```
[a13130] More changes here  
running puppet_compile.sh node1.devserver.net
```

```
Bisection: 1 revision left to test after this  
(roughly 1 step)
```

```
[f23f33] Changes that broke Puppet  
running puppet_compile.sh node1.devserver.net
```

```
Bisection: 0 revisions left to test after this  
(roughly 0 steps)
```

```
[e63b0a] Other changes  
running puppet_compile.sh node1.devserver.net
```

No interaction required!

```
Bisecting: 0 revisions left to test after this  
(roughly 0 steps)  
[e63b0a] Other changes  
running puppet_compile.sh node1.devserver.net
```

```
f23f33 is the first bad commit  
commit f23f33  
Author: Bad Committer <bc@devserver.com>  
Date:   Fri Sep 18 11:42:29 2015 -0500
```

Changes that broke Puppet

```
bisect run success
```

No interaction required!

```
Bisection: 0 revisions left to test after this  
(roughly 0 steps)  
[e63b0a] Other changes in this commit  
running puppet_compile.sh node1.devserver.net
```

```
f23f33 is the first bad commit  
commit f23f33  
Author: Bad Committer <bc@devserver.com>  
Date:   Fri Sep 18 11:42:29 2015 -0500
```

Changes that broke Puppet

```
bisect run success
```

git-bisect run puppet_compile.sh

- git-bisect run will automatically check out a version and simply run this script
- Exit 0 and git-bisect will mark revision as GOOD
- Exit 1-127 and git-bisect will mark revision as BAD
- envvars set by outer python script

```
#!/bin/sh
# puppet_compile.sh

sudo puppet master --compile $1 --environment
${ENVIRONMENT} > /dev/null
```

We've reduced our technical debt by

- Locating nodes that aren't submitting reports
- Alerting on nodes that are failing catalog compilation
- Ensuring puppet runs aren't violating idempotency
- Identifying why agents are taking so long to apply the catalog
- Tracking down commits that broke puppet

Future work

- Making puppet-blame more automatic
 - Mcollective Puppet plugin
 - Beaker
- Add more use cases to puppet-show.py
 - Catalog compiles by entire catalog can't be applied
- Using Puppet reporting for patch management



“When technical debt accrues, everybody loses.”
- Gene Kim

Puppet Reporting with Elasticsearch, Logstash, and Kibana

Indeed.com

Alex Elman

Devops Engineer, Operations

Find us here

Alex Elman



@_pkill



github.com/pkill

Connor Kelly



@cllyross



github.com/clly

Links and References

- https://docs.puppetlabs.com/puppet/3/reference/format_report.html
- <https://github.com/elastic/puppet-logstash-reporter>
- <https://puppetlabs.com/presentations/introducing-puppet-enterprises-event-inspector>
- https://docs.puppetlabs.com/pe/latest/CM_overview.html
- [Puppet Reporting Book](#)
- <https://git-scm.com/docs/git-bisect>
- Jordan Sissel @ PuppetConf [2012](#), [2013](#), [2014](#)
- Gene Kim @ [PuppetConf 2014](#)