BUILDING HA ELK STACK FOR DRUPAL

Marji Cermak

DevOps track, Experience level: Intermediate

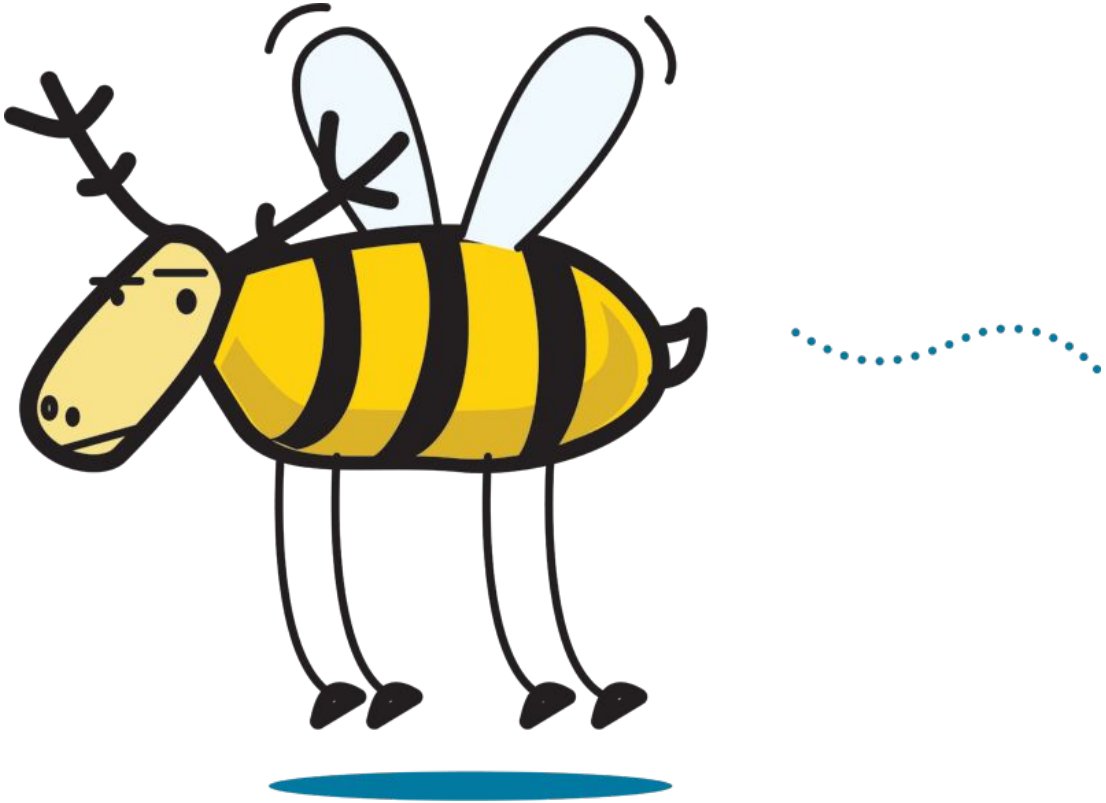
What is this ...
... ELK again?

Source: "Family of Elk on Grassland" (CC BY-NC-ND 2.0) by Princess-Lodges

The ELK stack

Elasticsearch **L**ogstash **K**ibana

Marji Cermak @cermakm



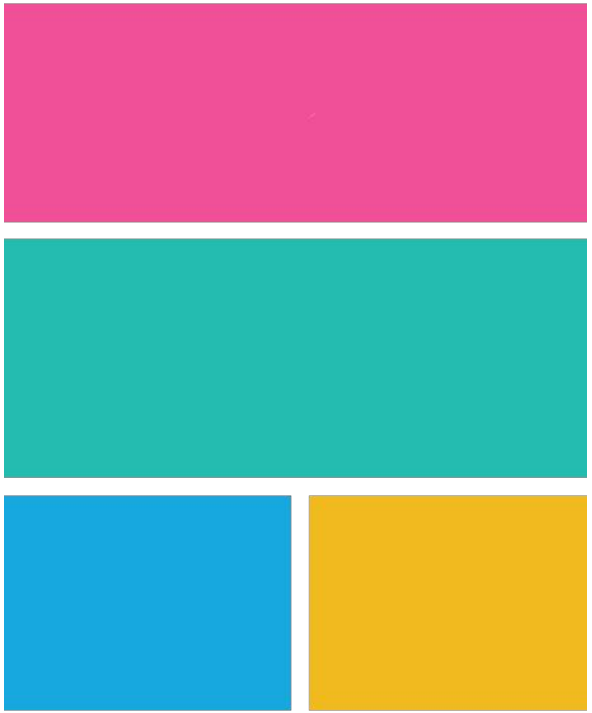
Source: https://www.elastic.co/blog/heya-elastic-stack-and-x-pack

The BELK stack

Beats **E**lasticsearch **L**ogstash **K**ibana

Marji Cermak @cermakm

The elastic stack



The elastic stack

The stack’s goal

- Take data from any source, any format,

The stack’s goal

- Take data from any source, any format,
- process, transform and enrich it,

The stack's goal

- Take data from any source, any format,
- process, transform and enrich it,
- store it,

The stack's goal

- Take data from any source, any format,
- process, transform and enrich it,
- store it,
- so you can search, analyse and visualise it in real time.

The four components of the BELK

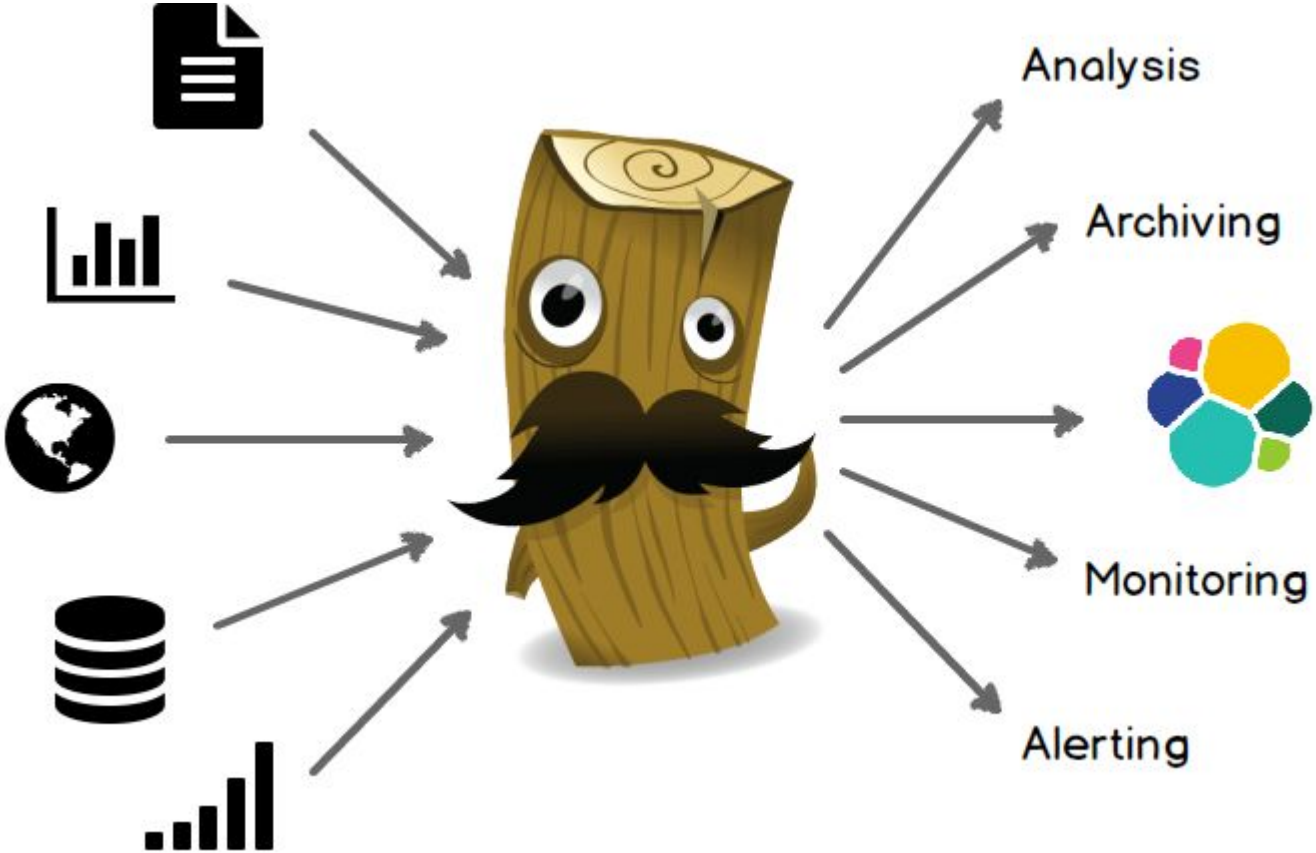
Elasticsearch

- open source, full-text search analytic engine
- distributed, High Availability
- designed for horizontal scalability and reliability
- based on Apache Lucene (like Apache solr)
- written in Java
- Plugins - a way to enhance ES functionality

elasticsearch

Logstash

- tool to collect, process, and forward events and log messages
- data collection, enrichment and transformation pipeline
- configurable input and output plugins
- e.g. logfile, MS windows eventlog, socket, Syslog, redis, salesforce, Drupal DBLog



Source: https://www.elastic.co/guide/en/logstash/current/introduction.html

Logstash

dozens of **input** plugins

- **Beats**
- file
- TCP, UDP, websocket
- syslog
- **redis**
- MS windows eventlog
- **drupal_dblog**

Logstash

dozens of **input** plugins

dozens of **output** plugins

- file
- TCP, UDP, websocket
- syslog
- **redis, SQS**
- graphite, influxdb
- nagios, zabbix
- jira, redmine
- s3
- **elasticsearch**

Logstash

dozens of **input** plugins

dozens of **output** plugins

dozens of **filter** plugins

- **grok**
- mutate
- drop
- date
- geoip

Kibana

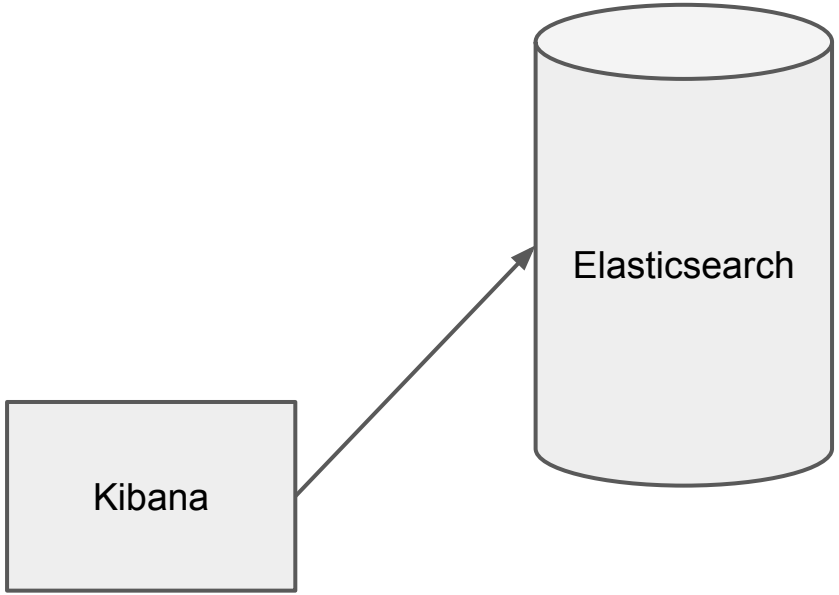
- open source data visualisation platform
- allows to interact with data through powerful graphics
- brings data to life with visuals

kibana

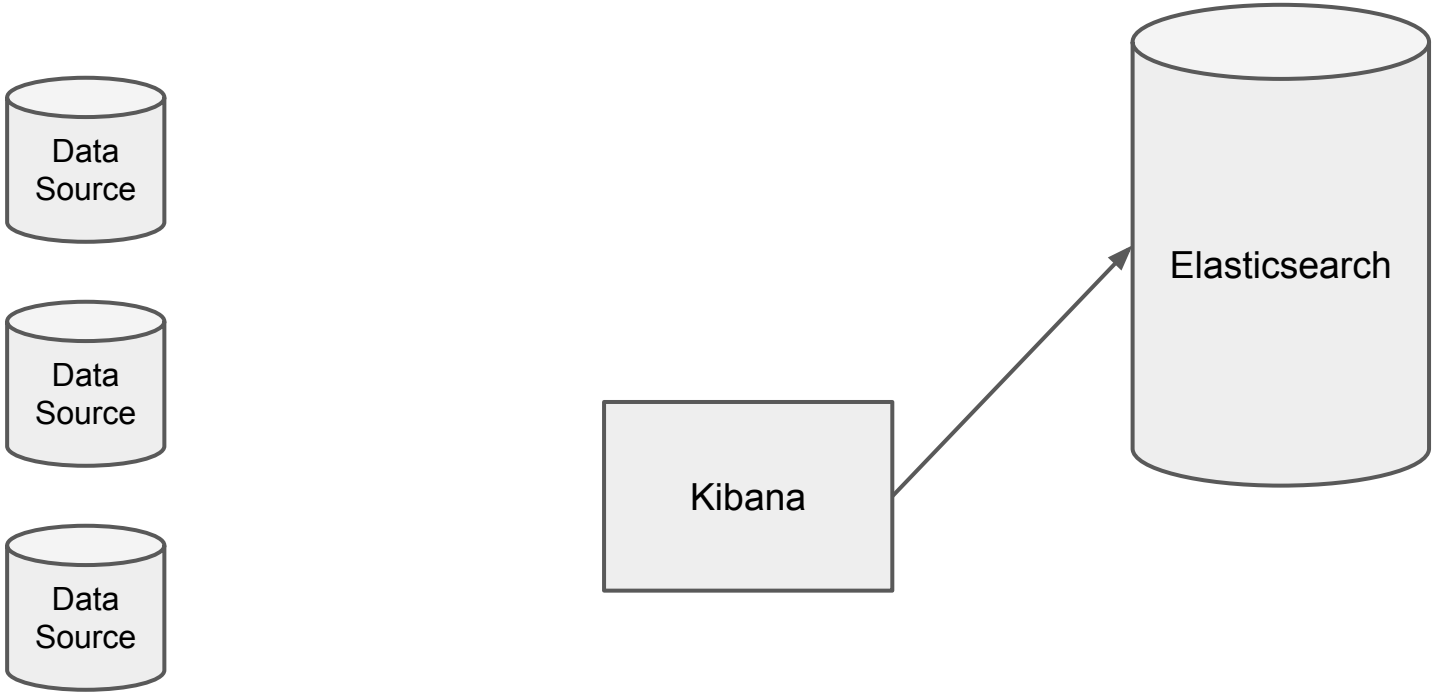
Beats

- Open source data shippers
- Lightweight
- Different beats:
  Filebeat, Topbeat, Packetbeat,
  Winlogbeat, Libbeat

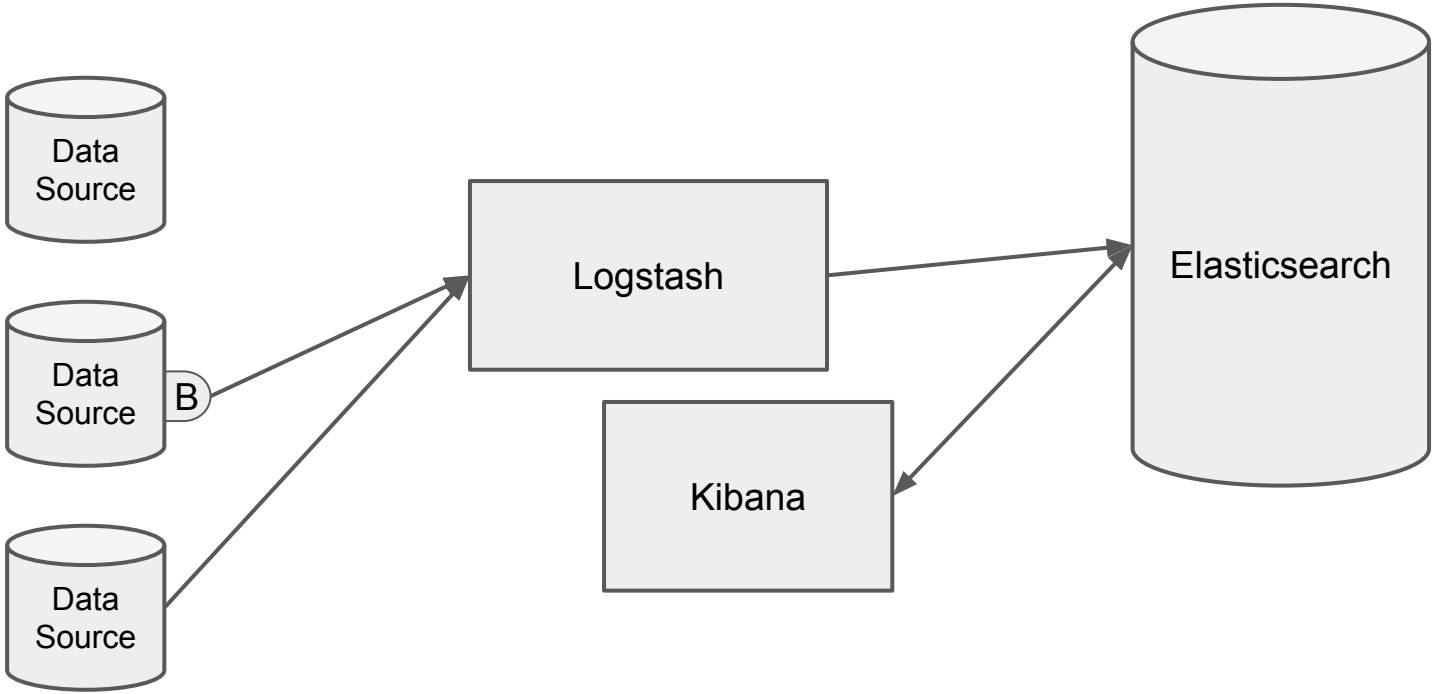
The BELK flow



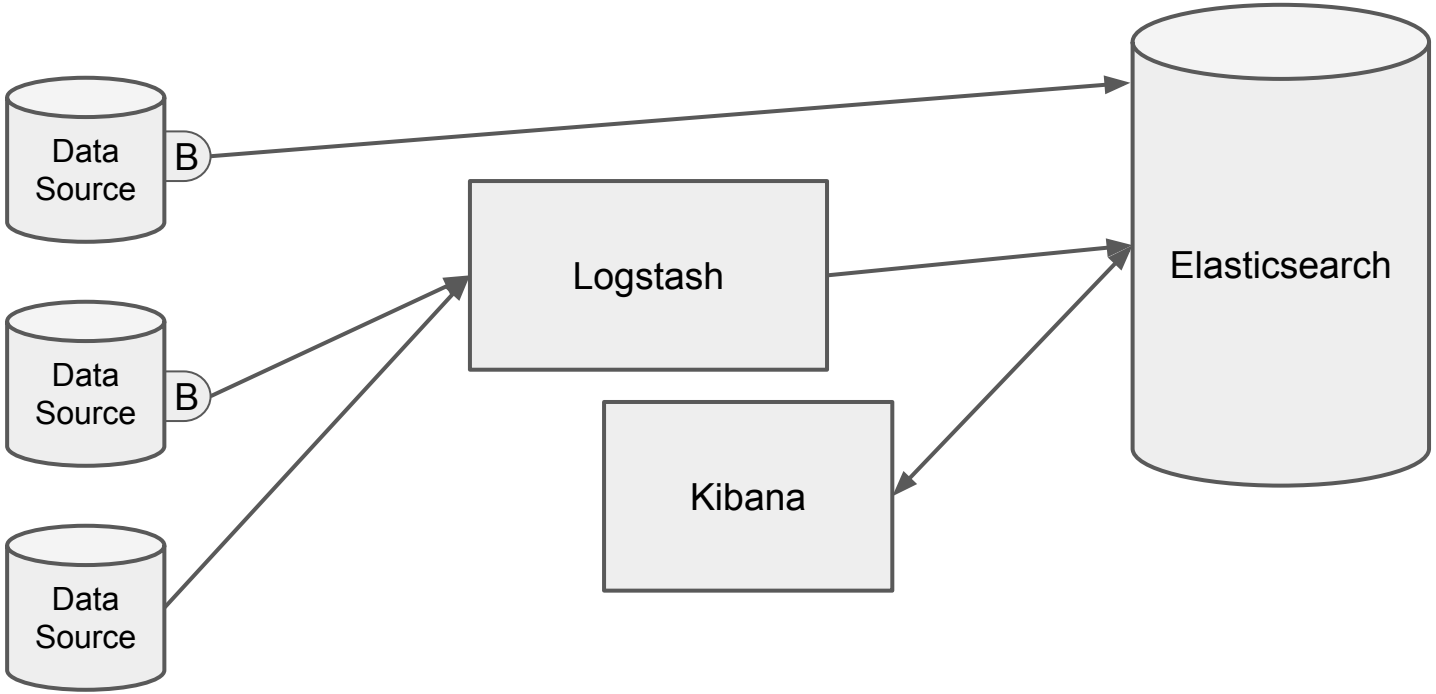
The BELK flow



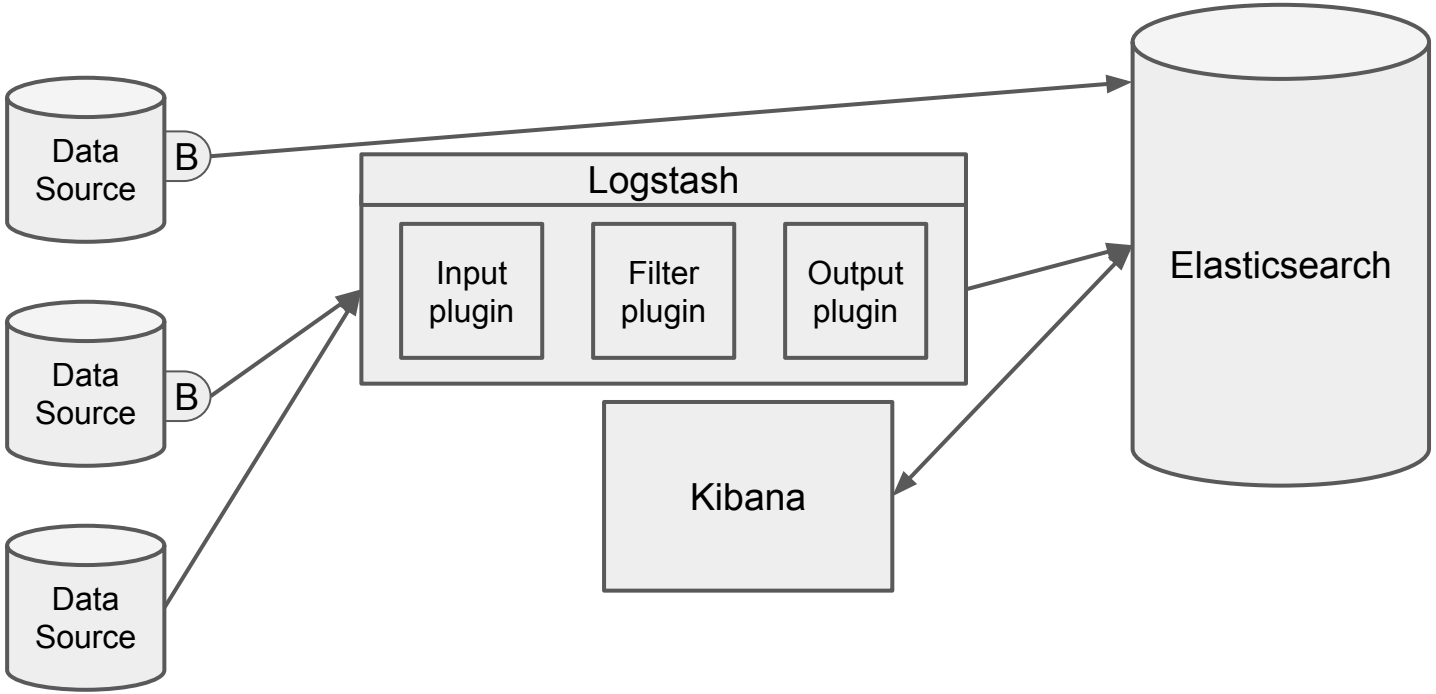
The BELK flow



The BELK flow



The BELK flow




DUBLIN
DRUPALCON
Tell me something new...
How do I build a HA ELK?

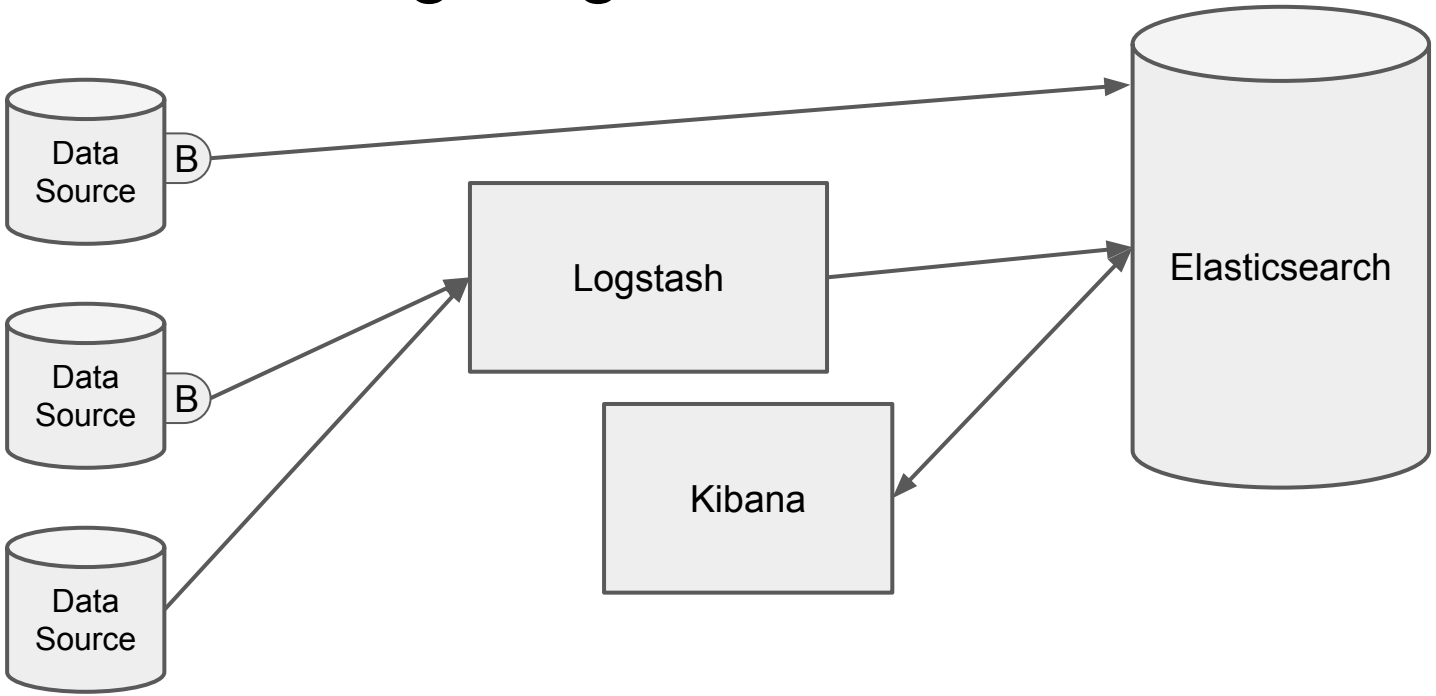
Why would you want a HA ELK (use case)

Imagine a banking industry client with a few dozens of sites (and servers).

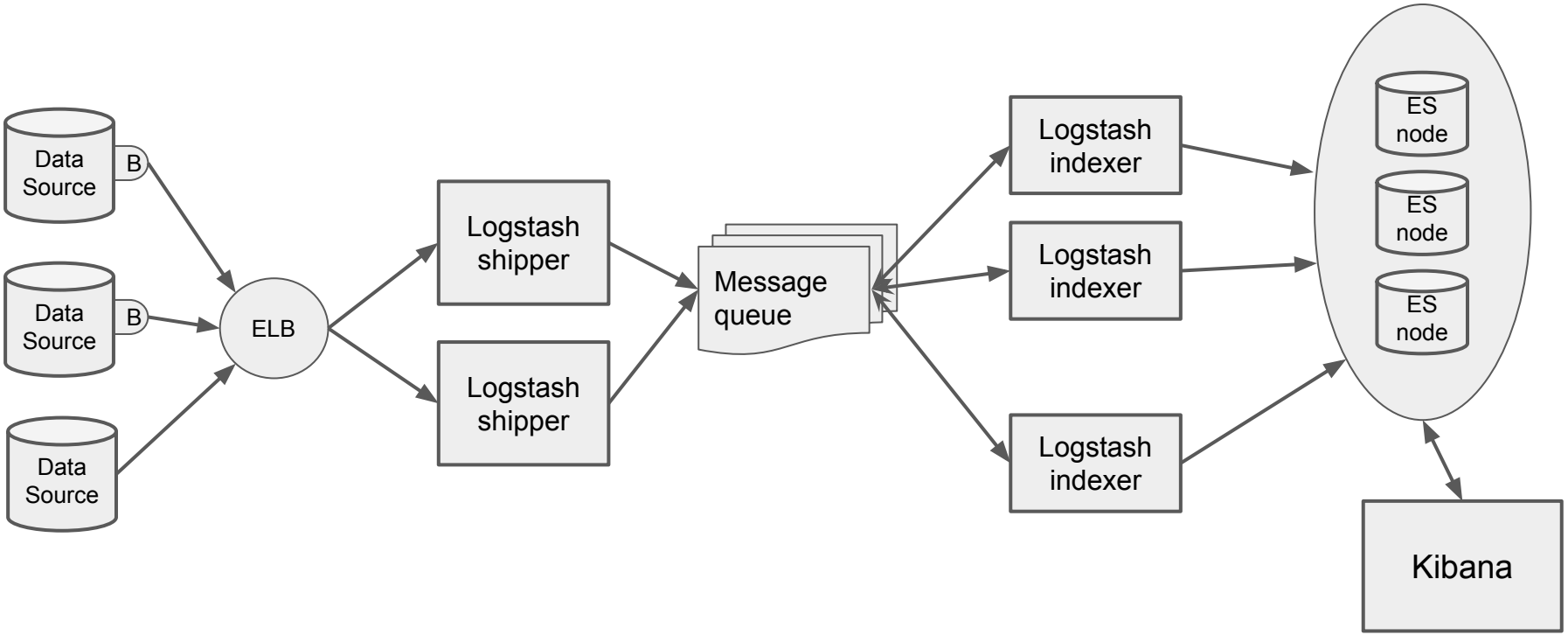
They want all logs in one place. They cannot lose any log. They might have data retention requirements.

Audits, customer complaints.

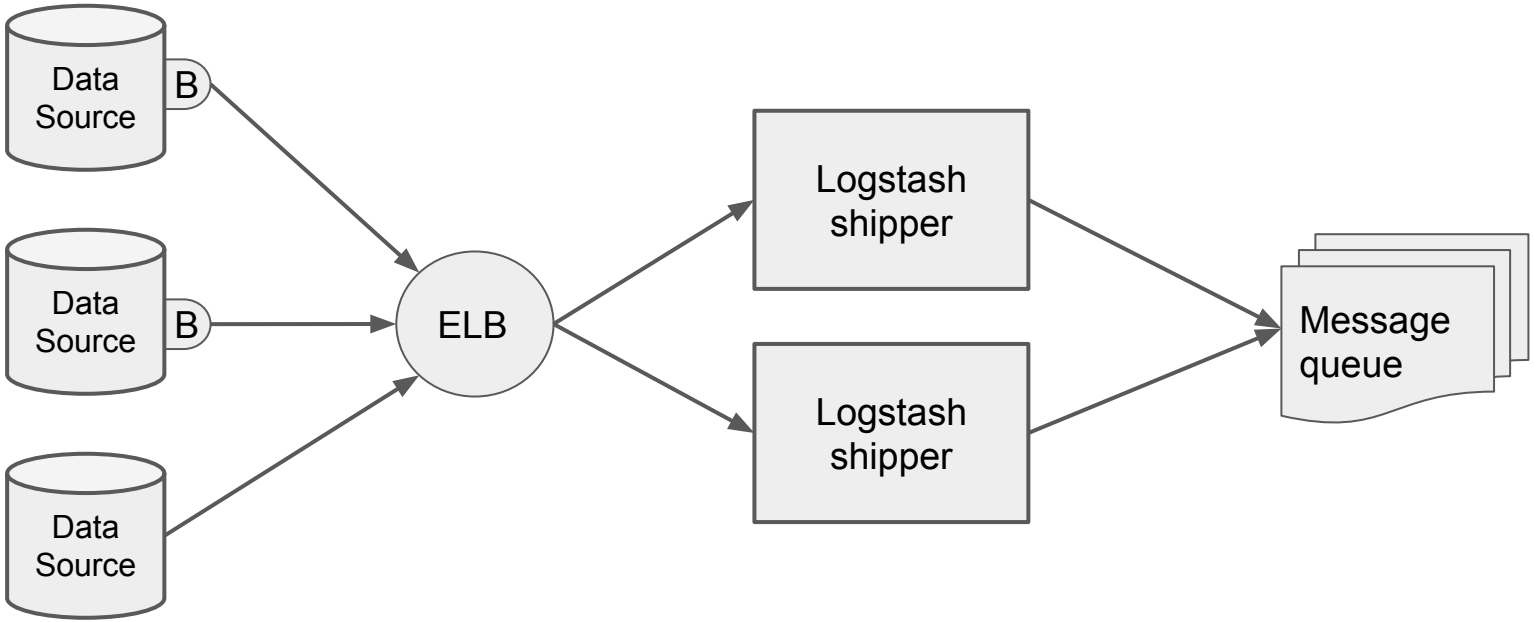
Let’s make things high available



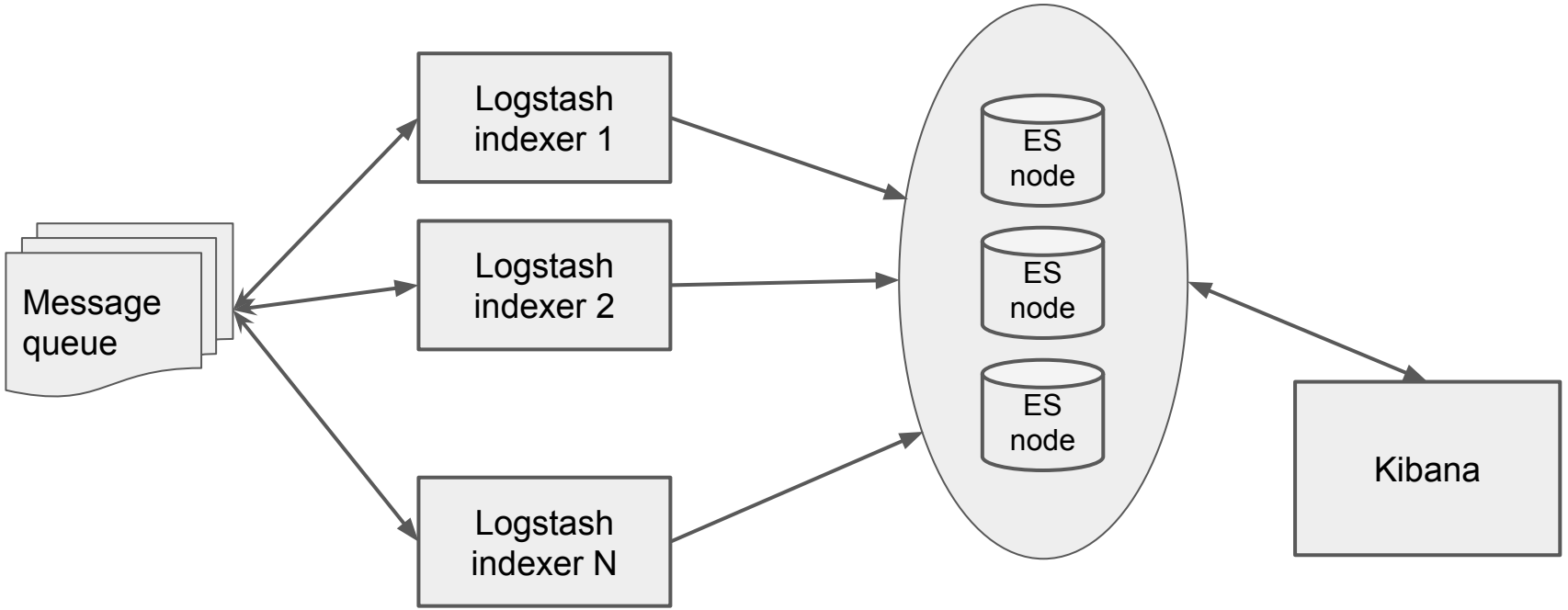
High Available ELK



High Available ELK (logs receiving part)



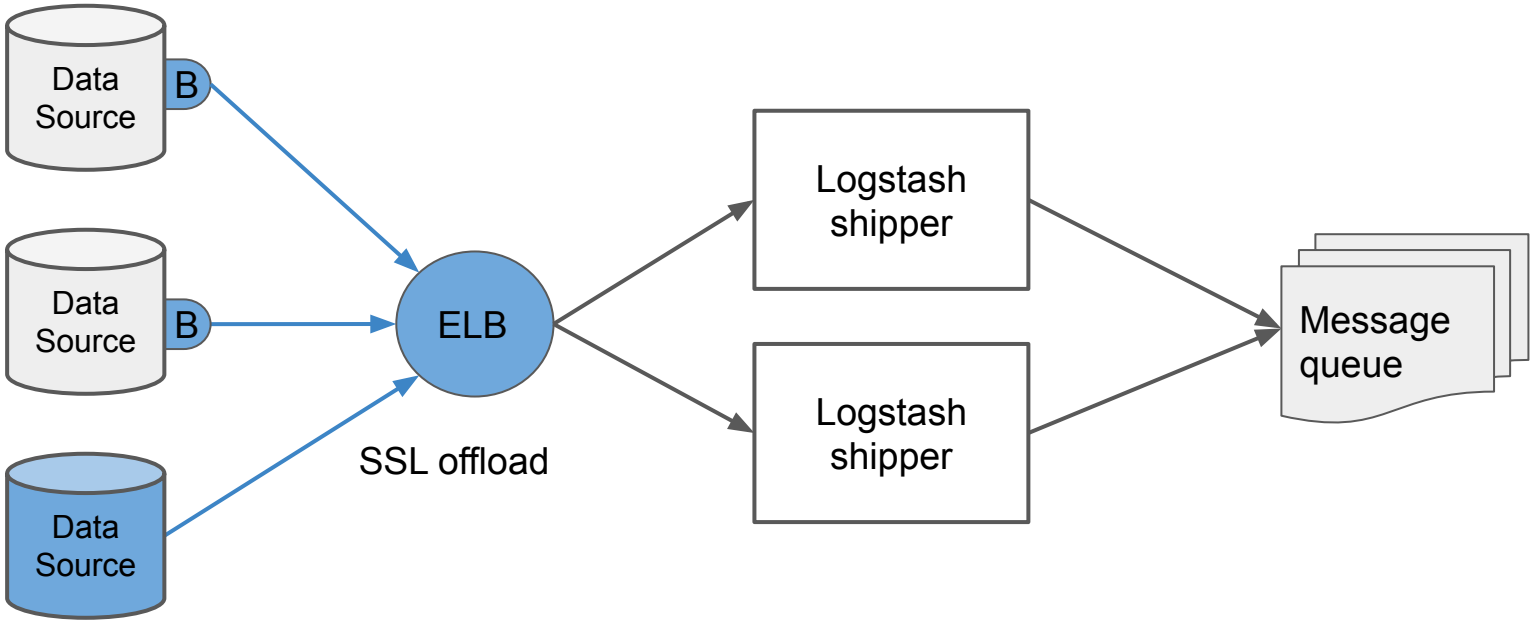
High Available ELK (logs processing part)



High Available ELK

Diving in

Shipping data

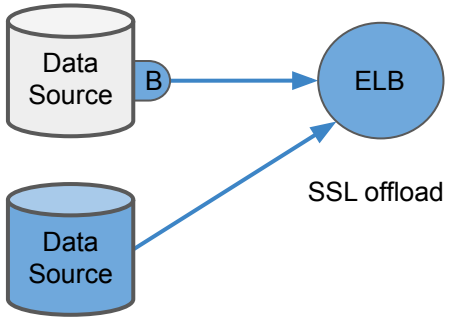


Shipping data

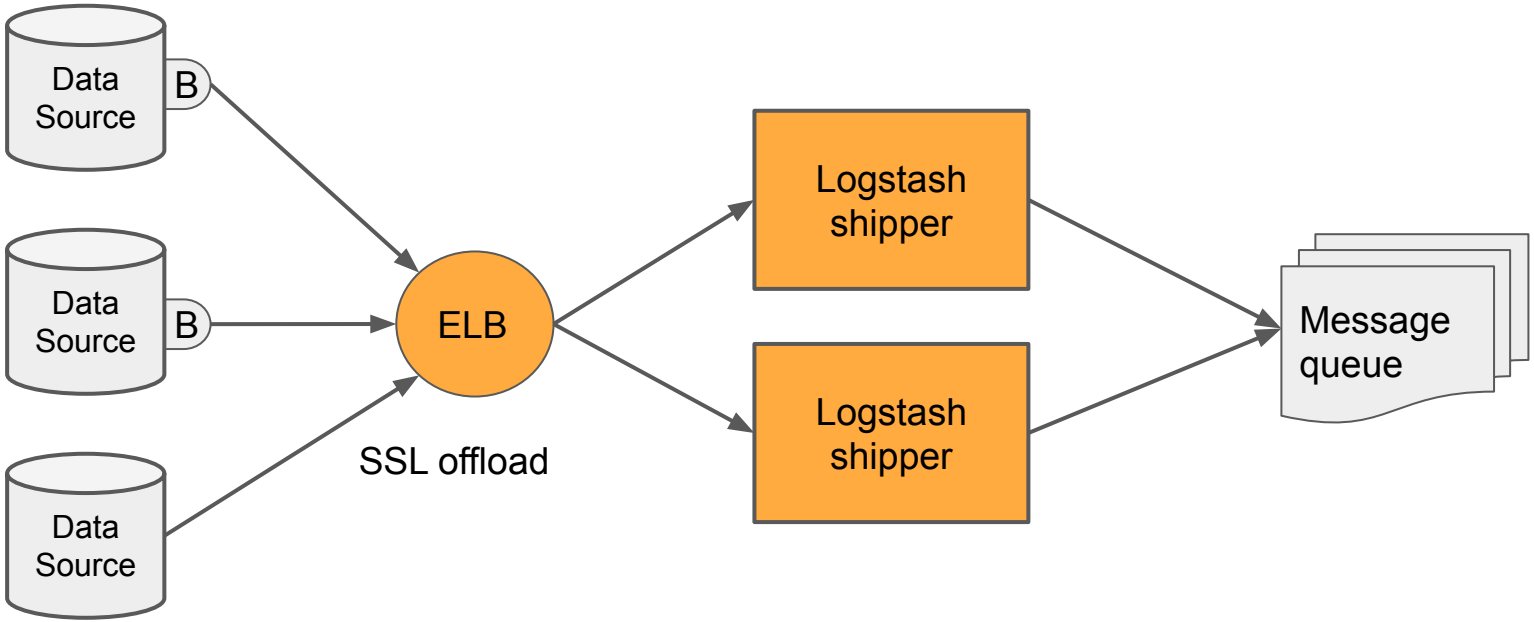
HA way of shipping

- Beats
- Syslog
- Avoid UDP

SSL encryption



ELB and multiple logstash shippers



ELB and multiple logstash shippers

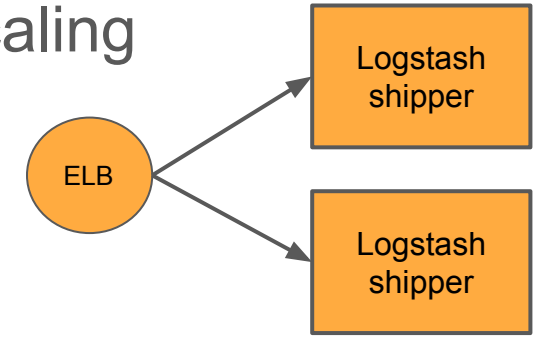
Logstash shipper

- Main purpose is to store events in the message queue
- Very lightweight - minimal processing

ELB and multiple logstash shippers

Elastic Load Balancer

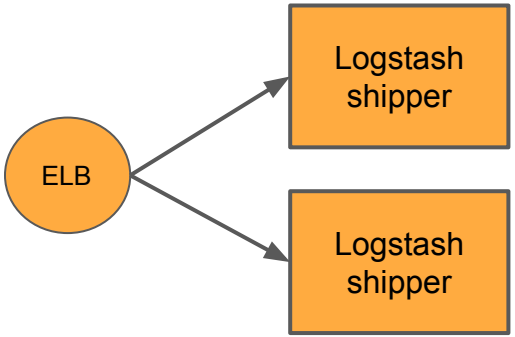
- Enable shipper failure / update / reboot / reprovision
- ELB can protect you from a zone failure
- SSL offload on the ELB - CPU auto scaling built in ELB



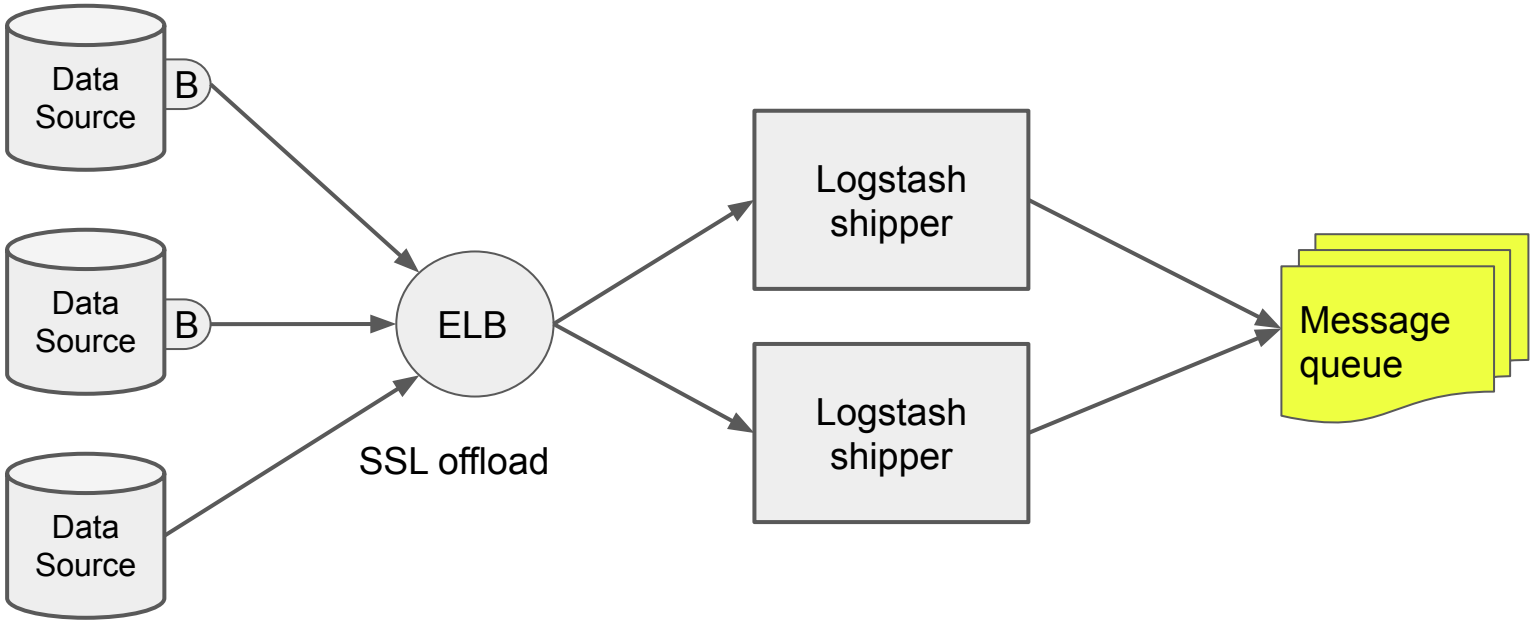
ELB and multiple logstash shippers

Cons

- No static IP / range - cannot whitelist in FW
- ELB does not support client side SSL Authentication (2-way SSL authentication)



Message queue



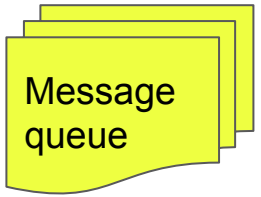
Message queue

SQS

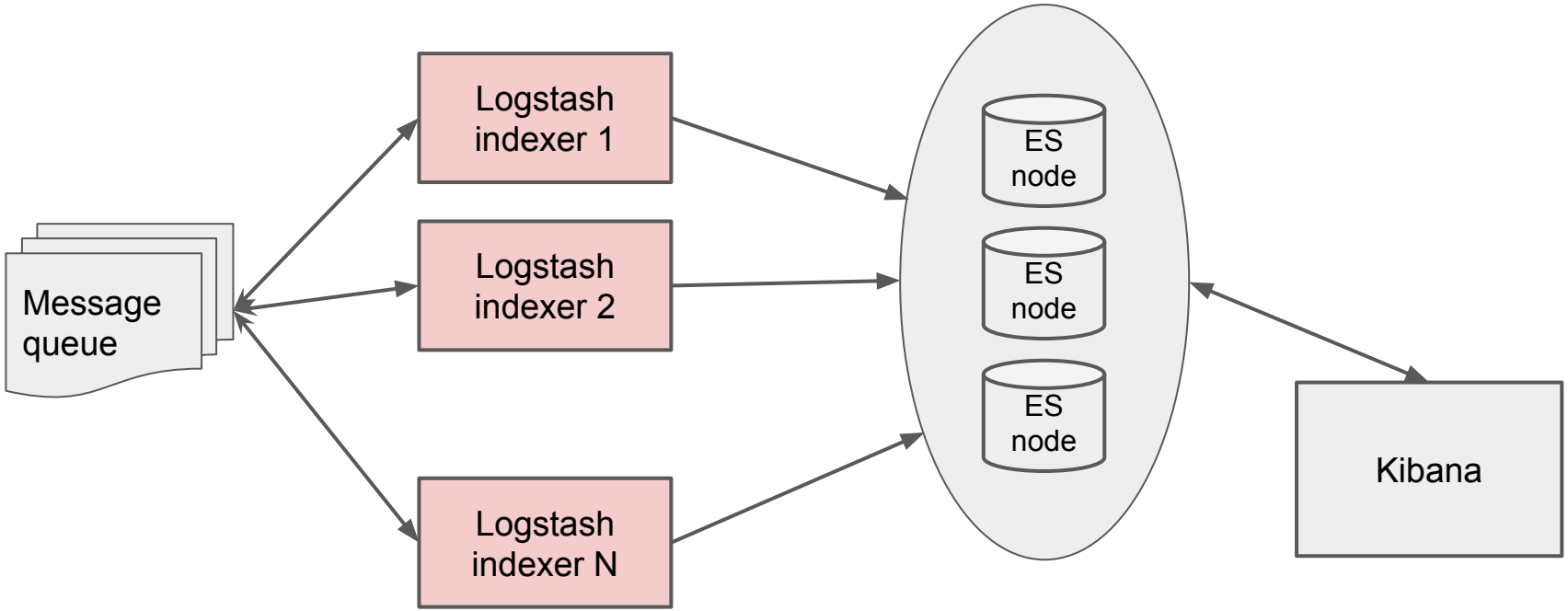
- fast, reliable, scalable, fully managed message queuing service
- unlimited number of services and messages

Cons

- Not supported by beats (while Redis is)



Logstash indexers



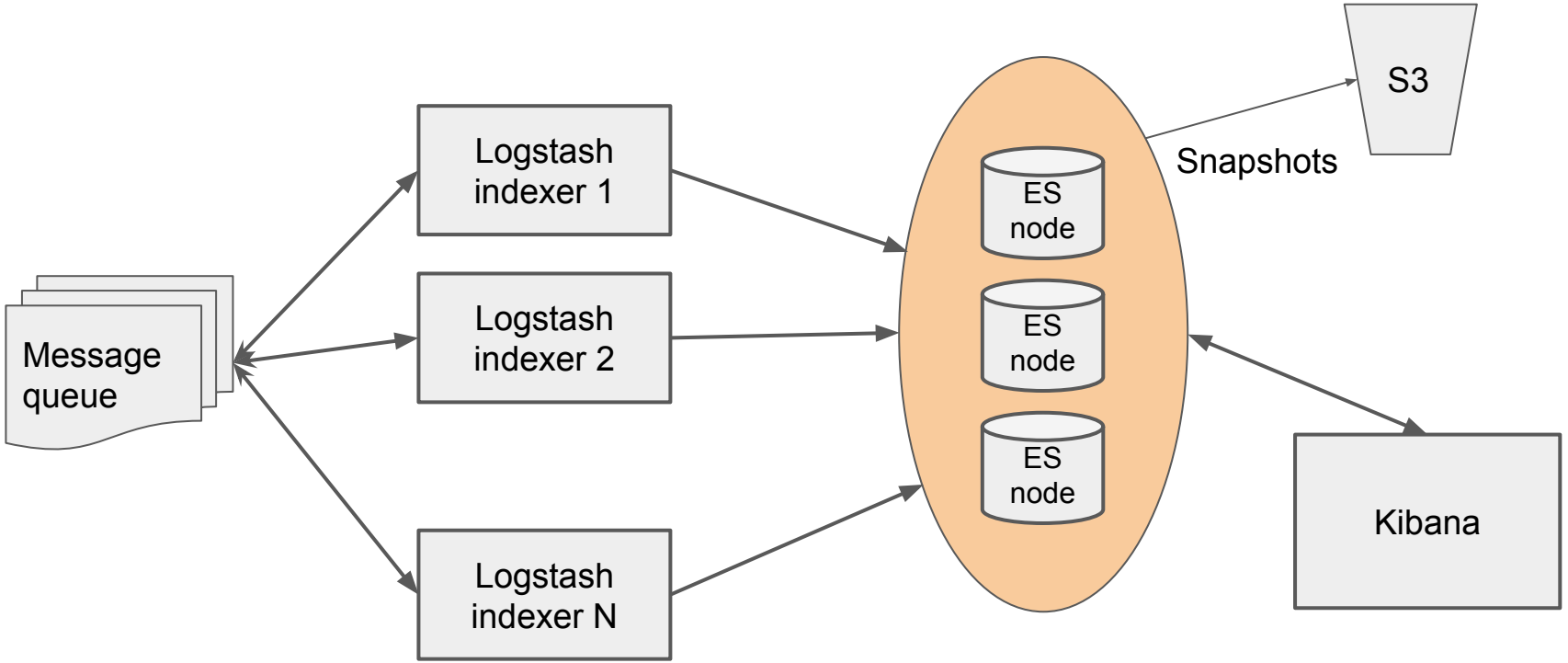
Logstash indexers

Provision more instances if the queue grows

HA here means “logs are processed close to real-time”

Auto-scaling policy automatically adding extra instance when queue grows

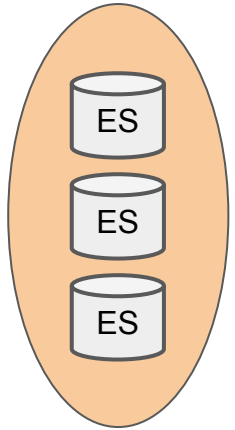
Elasticsearch cluster



Elasticsearch cluster

Avoid 2 nodes - either split-brain possibility or no HA

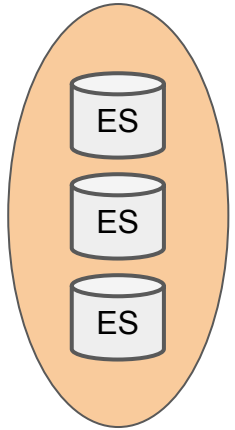
3 master-eligible nodes is the minimum



Elasticsearch cluster

No need for ELB:

- ES Cluster has load balancing built in
- Kibana recommends running a local ES node
- Logstash supports



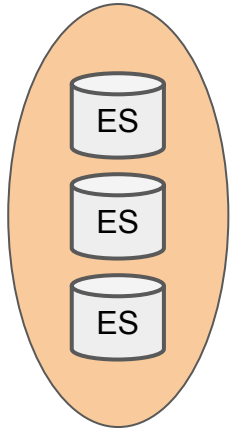
Elasticsearch - data storage

directory(ies) where ES stores data

Use SSD instance store if you can

If not, then SSD EBS :

- provisioned IOPS SSD (io1)
- max size General Purpose SSD (gp2)



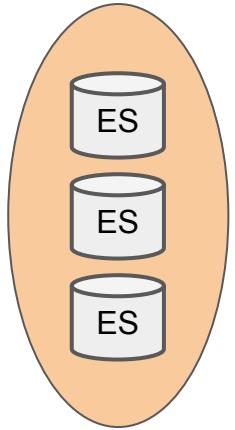
Elasticsearch - data storage maintenance

Avoid using more than 80% of disk space

Snapshot and restore module

- Allows to create snapshots into a remote repo
- Several backends - shared FS, AWS cloud, HDFS, Azure cloud

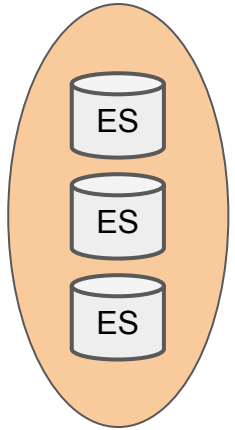
AWS Cloud plugin - S3 backup



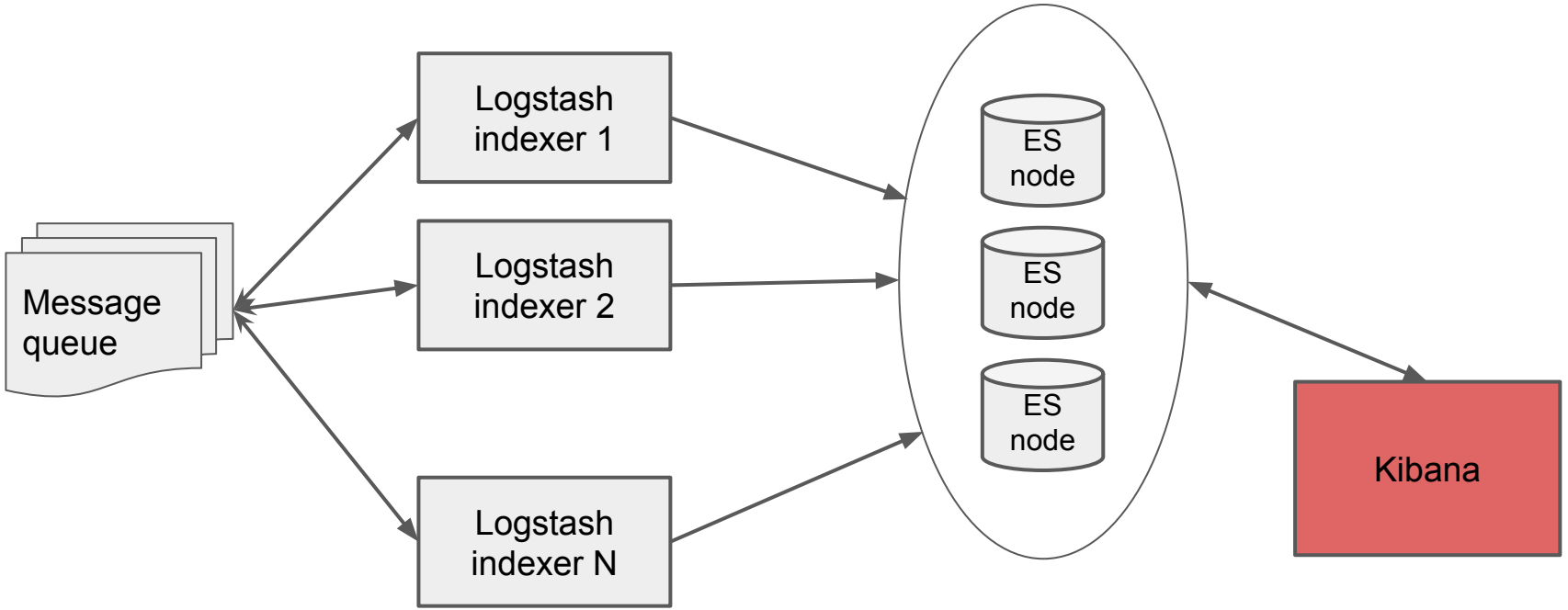
Elasticsearch - data storage maintenance

Curator

- Tool to curate ES indices and snapshots
- Perfect for creating and deleting snapshots



Kibana


Message queue
Logstash indexer 1
Logstash indexer 2
Logstash indexer N
ES node
ES node
ES node
Kibana

Kibana

Single instance (ready to be reprovisioned)

If you have many heavy users, load balance across multiple Kibana instances

Kibana

Don’t run kibana on existing ES node (master/data)

Instead, install Kibana and ES client node on the same machine (ES client nodes are smart LB that are part of the cluster)

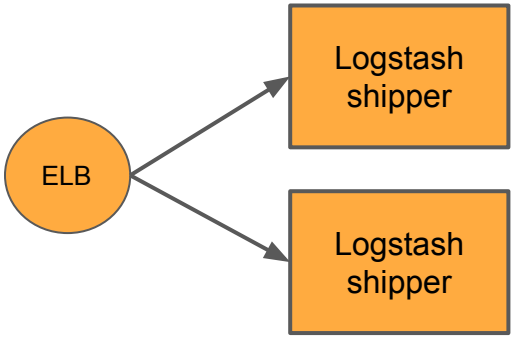
Upgrading / Patching ELK

without losing data

Patching Logstash servers

Shippers

- ELB with “Connection draining” enabled
- Add new (updated) instances
- Deregistering old instances



Patching Logstash servers

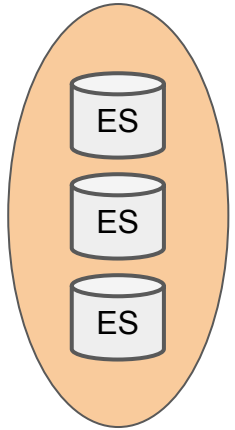
Indexers

- Provision a new instance or take it offline (no data lost, they consume from the queue)

Patching Elasticsearch nodes

Rolling upgrade (no service interruption) or Full cluster restart

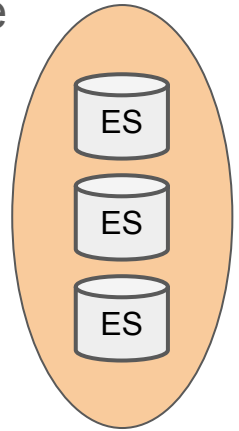
Plugins must be upgraded alongside Elasticsearch



Patching Elasticsearch nodes

Live migration

- Provision new ES cluster
- Have logstash indexers write to both old and new cluster for a while
- Load data from snapshot
- Make Kibana use new cluster
- Terminate old cluster



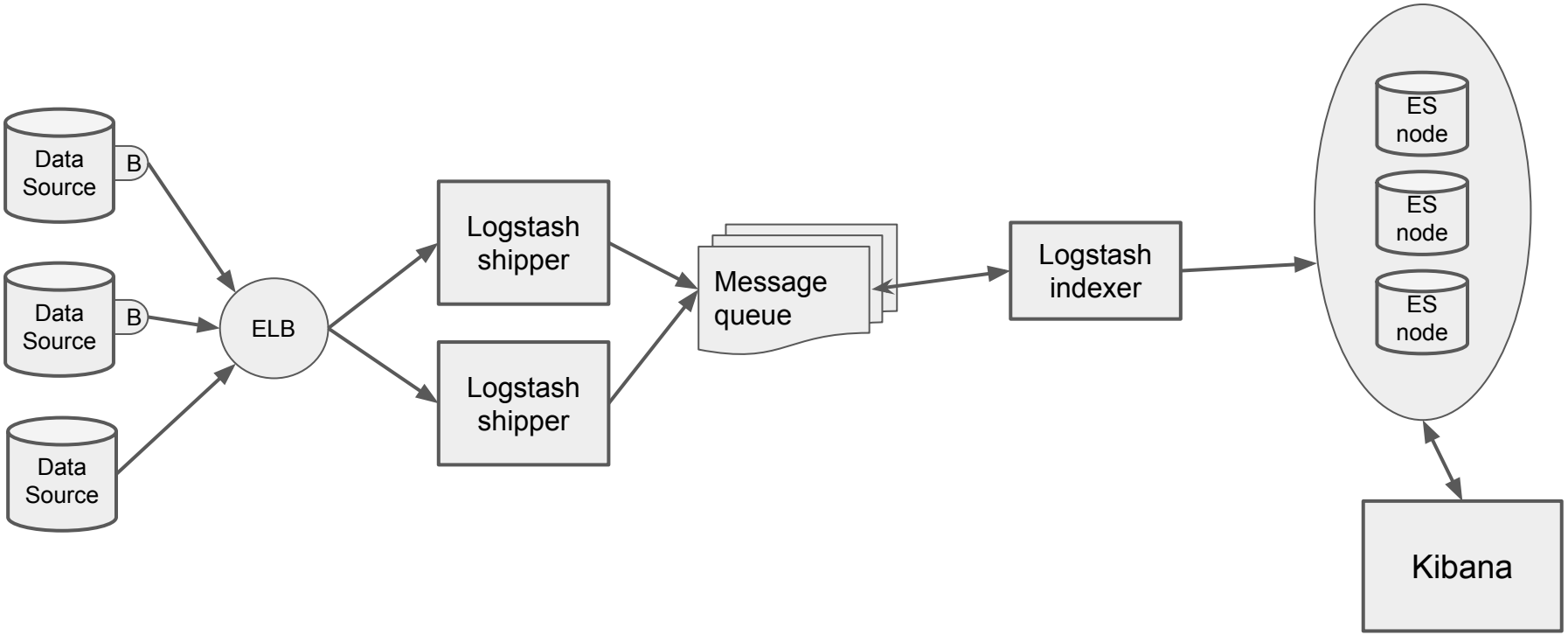
Patching Kibana

Provision new kibana server and

- take over the EIP or
- update Kibana’s DNS record

Cost estimate

Cost estimate



Cost estimate

https://calculator.s3.amazonaws.com/index.html

	USD per month
1 x indexer: c4.large	$77
2 x shipper: c4.large	$154
3 x ES node: m4.xlarge ($175 each)	$525
1 x kibana: t2.small	$20
3 x SSD EBS (gp2), 1TB	$350
S3, ELB, traffic	~ $80
TOTAL per month	~ $1200

ELK Alternatives

ELK alternatives

Elastic Cloud

- AKA “Hosted Elasticsearch & Kibana on AWS”
- no logstash
- starts at $45 per month

Loggly, Sumo Logic, Papertrail, Logentries

Complements to HA ELK

Monitoring ELK

Cluster health

```
GET _cluster/health
```

green

yellow

red


```
{
   "cluster_name": "cluster02",
   "status": "green",
   "timed_out": false,
   "number_of_nodes": 1,
   "number_of_data_nodes": 1,
   "active_primary_shards": 10,
   "active_shards": 10,
   "relocating_shards": 0,
   "initializing_shards": 0,
   "unassigned_shards": 0
}
```


Monitoring ELK

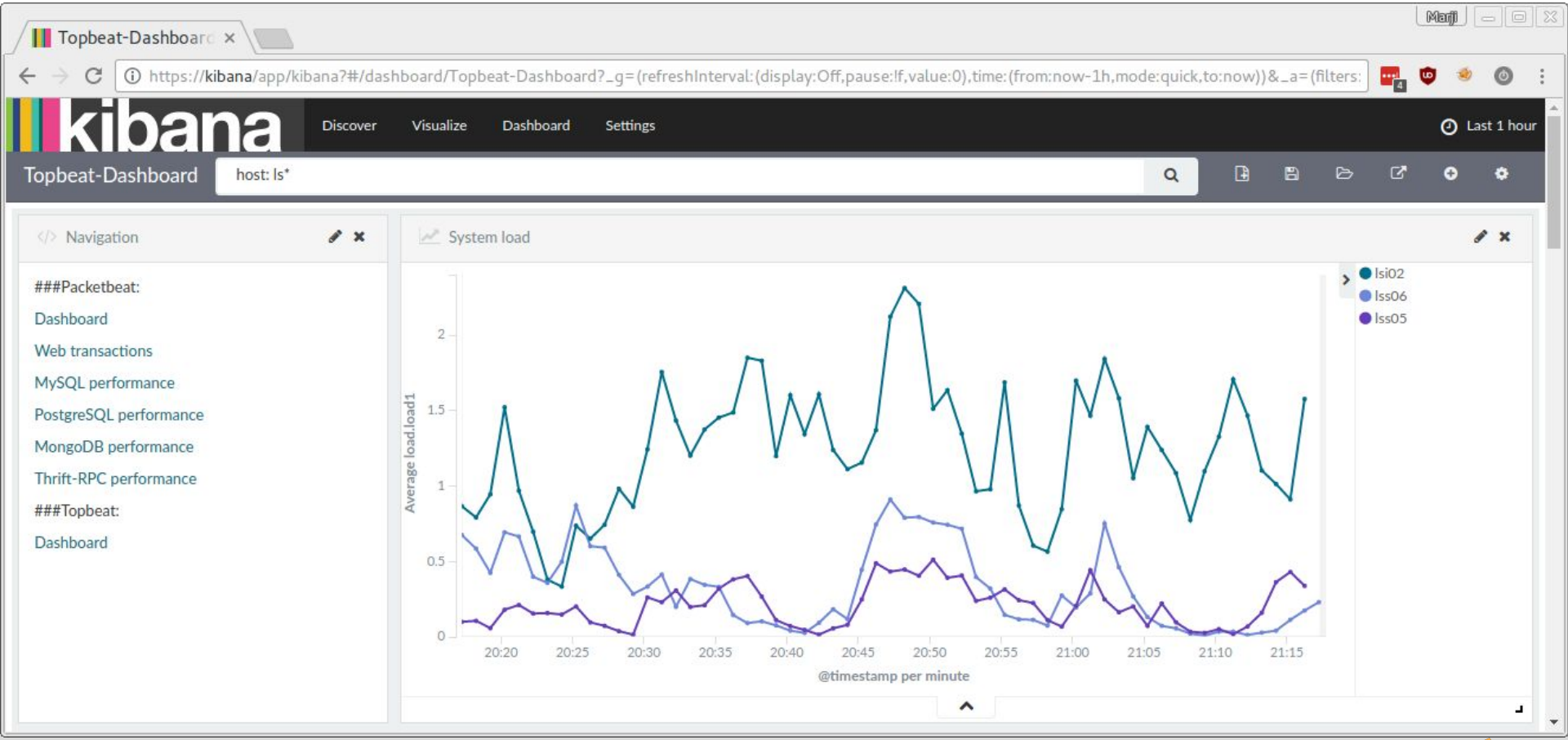
Alerting on

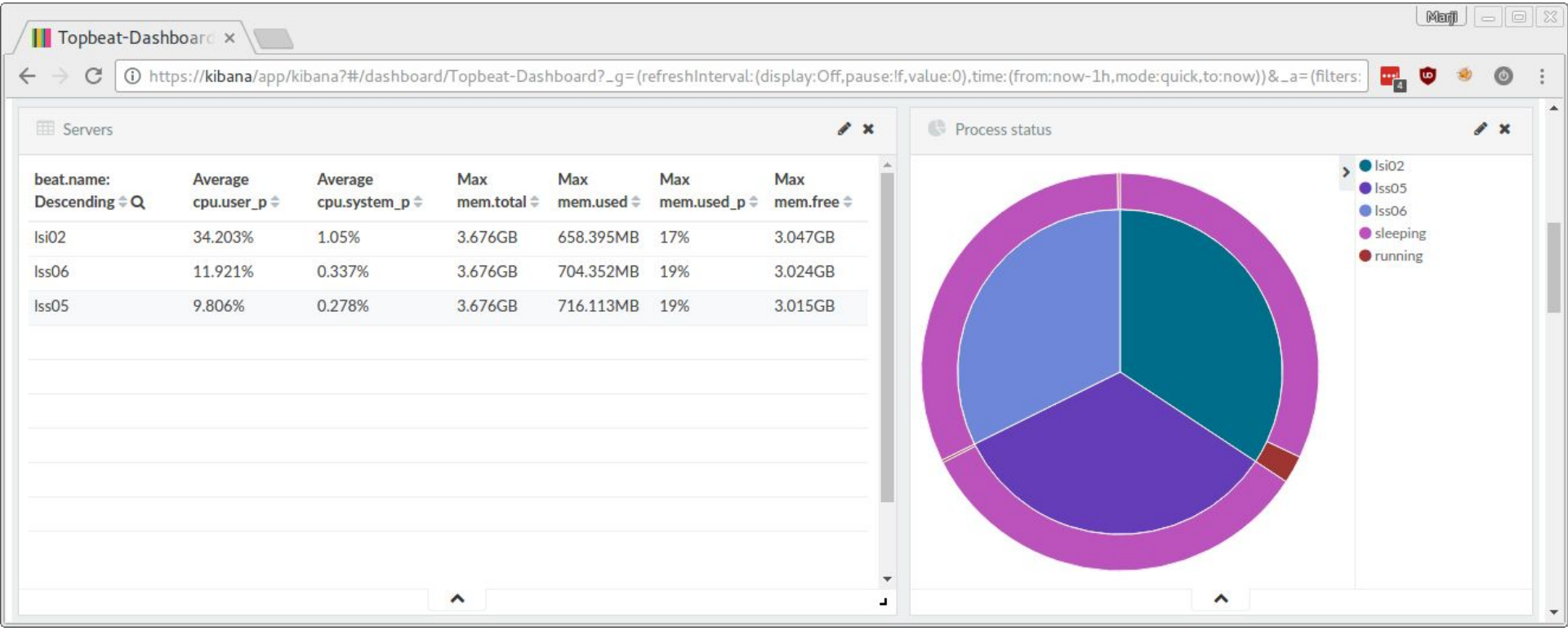
- ES cluster status
- ES disk space and inode usage
- Logstash heartbeat
- Timestamp of the most recent record in ES cluster
- Kibana availability

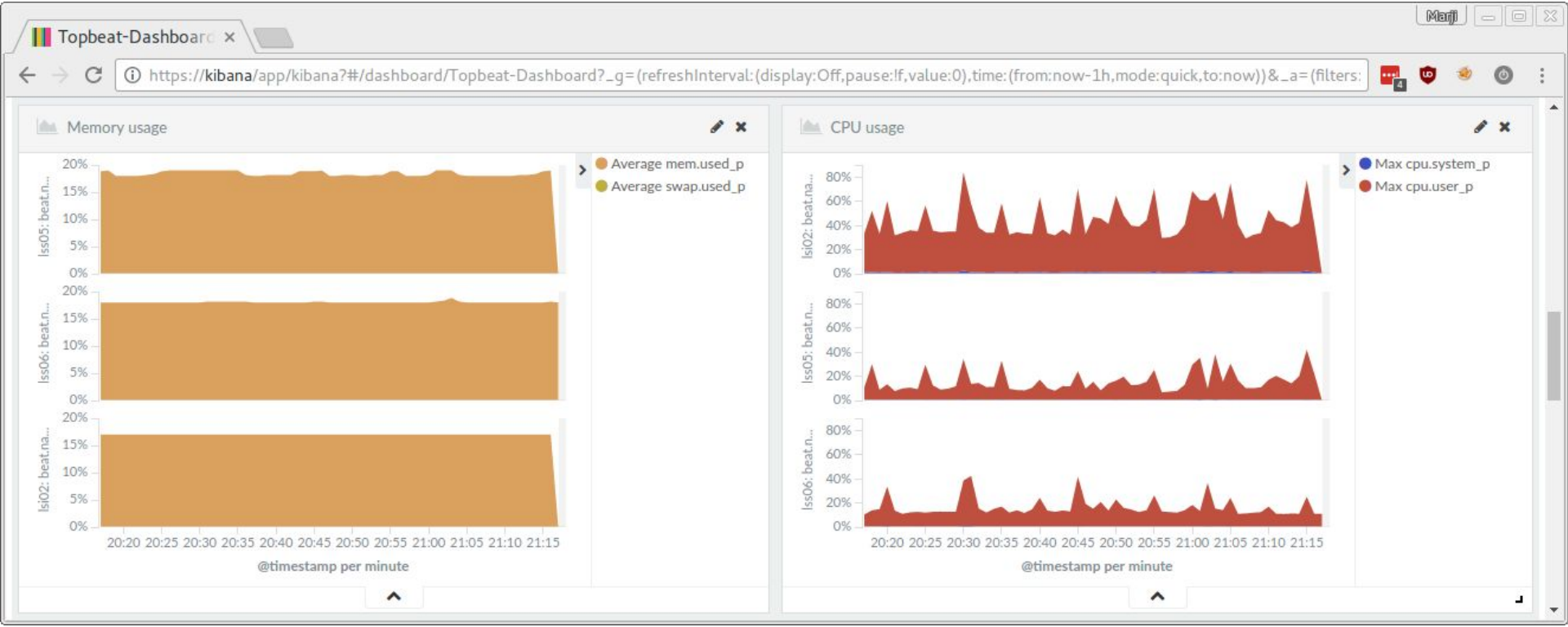
Monitoring ELK

Metrics

- be able to compare utilisation of cluster members
- memory and CPU, load, swap, descriptors trends
- ES monitoring - dozens of metrics, e.g. JVM performance


Topbeat-Dashboard
https://kibana/app/kibana?#/dashboard/Topbeat-Dashboard?_g=(refreshInterval:(display:Off,pause:!f,value:0),time:(from:now-1h,mode:quick,to:now))&_a=(filters:
kibana
Discover
Visualize
Dashboard
Settings
Last 1 hour
Topbeat-Dashboard
host: ls*
Navigation
###Packetbeat:
Dashboard
Web transactions
MySQL performance
PostgreSQL performance
MongoDB performance
Thrift-RPC performance
###Topbeat:
Dashboard
System load
lsi02
lss06
lss05
Average load.load1
0
0.5
1
1.5
2
20:20
20:25
20:30
20:35
20:40
20:45
20:50
20:55
21:00
21:05
21:10
21:15
@timestamp per minute

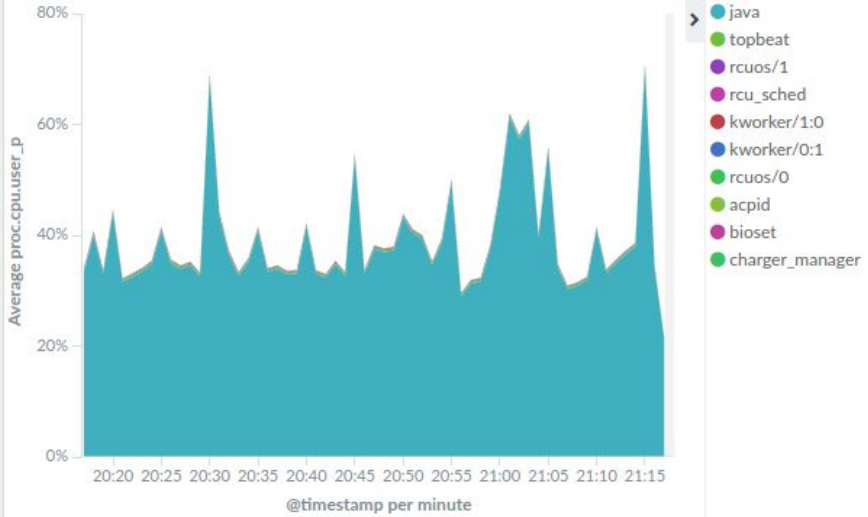

Topbeat-Dashboard
Marji
https://kibana/app/kibana?#/dashboard/Topbeat-Dashboard?_g=(refreshInterval:(display:Off,pause:!f,value:0),time:(from:now-1h,mode:quick,to:now))&_a=(filters:
Servers
beat.name: Descending
Average cpu.user_p
Average cpu.system_p
Max mem.total
Max mem.used
Max mem.used_p
Max mem.free
lsi02 34.203% 1.05% 3.676GB 658.395MB 17% 3.047GB
lss06 11.921% 0.337% 3.676GB 704.352MB 19% 3.024GB
lss05 9.806% 0.278% 3.676GB 716.113MB 19% 3.015GB
Process status
lsi02
lss05
lss06
sleeping
running


Topbeat-Dashboard
https://kibana/app/kibana?#/dashboard/Topbeat-Dashboard?_g=(refreshInterval:(display:Off,pause:!f,value:0),time:(from:now-1h,mode:quick,to:now))&_a=(filters:
Memory usage
Average mem.used_p
Average swap.used_p
lss05: beat.n...
lss06: beat.n...
lsi02: beat.na...
20%
15%
10%
5%
0%
20:20 20:25 20:30 20:35 20:40 20:45 20:50 20:55 21:00 21:05 21:10 21:15
@timestamp per minute
CPU usage
Max cpu.system_p
Max cpu.user_p
80%
60%
40%
20%
0%

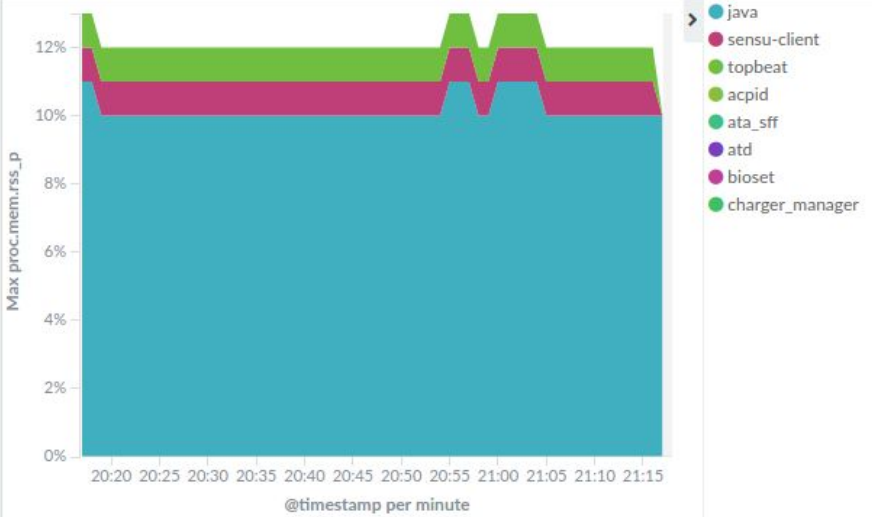
Topbeat-Dashboard

https://kibana/app/kibana?#/dashboard/Topbeat-Dashboard?_g=(refreshInterval:(display:Off,pause:!f,value:0),time:(from:now-1h,mode:quick,to:now))&_a=(filters:

CPU usage per process



Memory usage per process



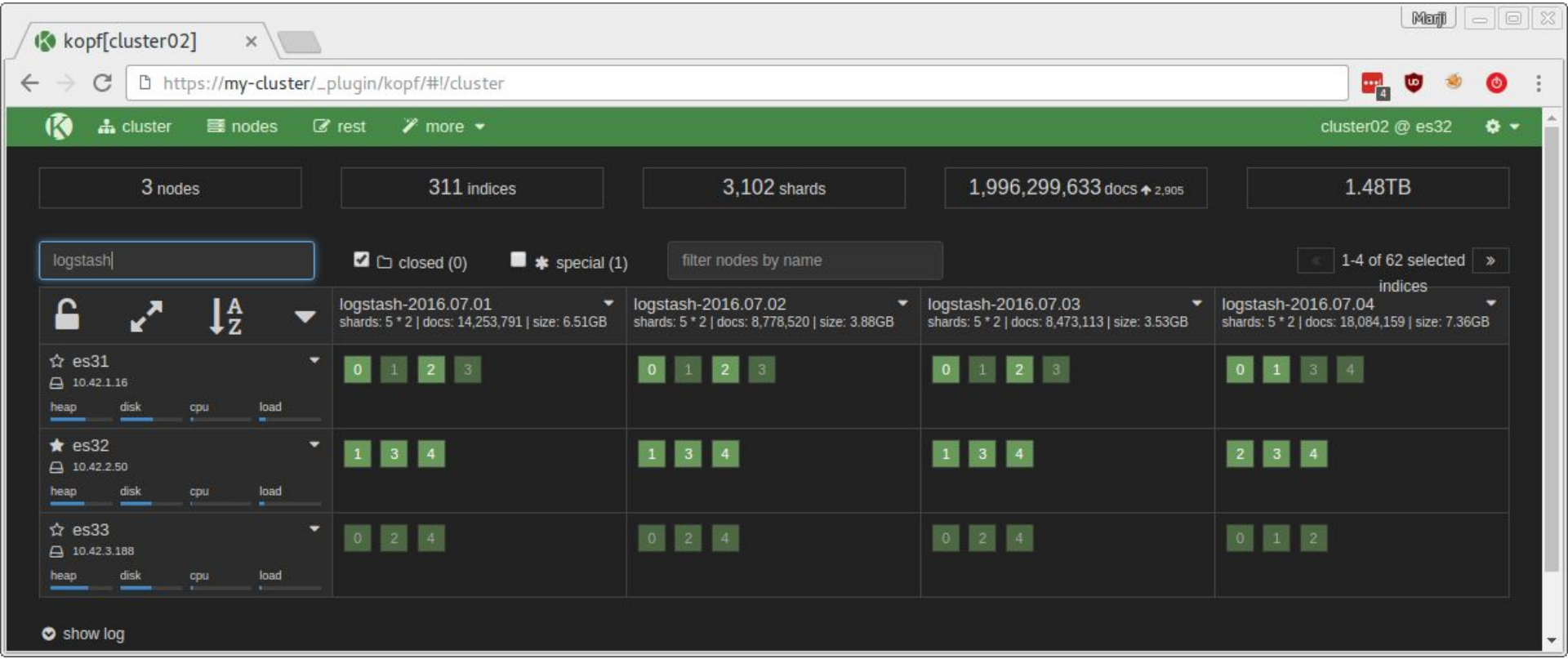
Top processes

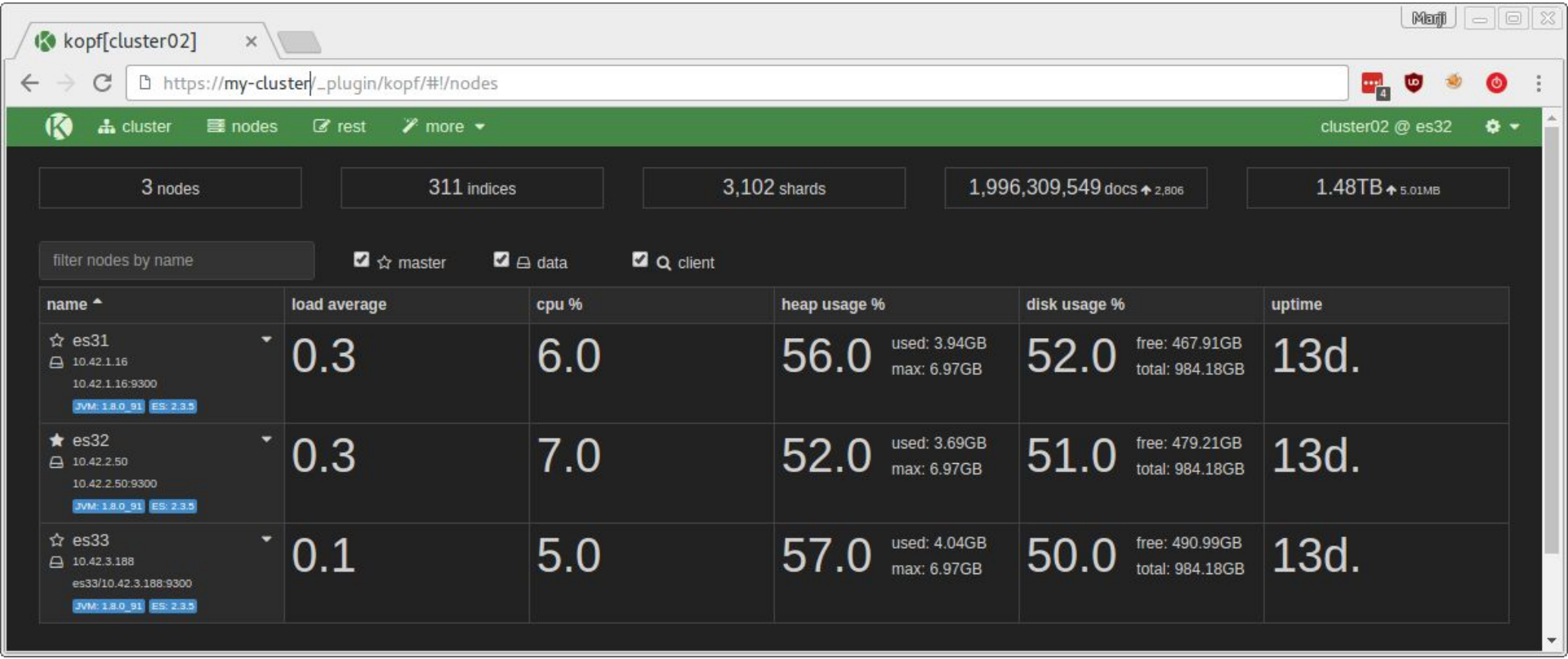
proc.name: Descending	Max proc.cpu.user_p	Max proc.mem.rss	Max proc.mem.rss_p	Max proc.mem.share
java	168.89%	398.555MB	11%	15.094MB
check-cpu.rb	0.5%	9.453MB	0%	2.723MB
topbeat	0.5%	21.32MB	1%	4.504MB

Disk utilization over time

Elasticsearch web admin plugins

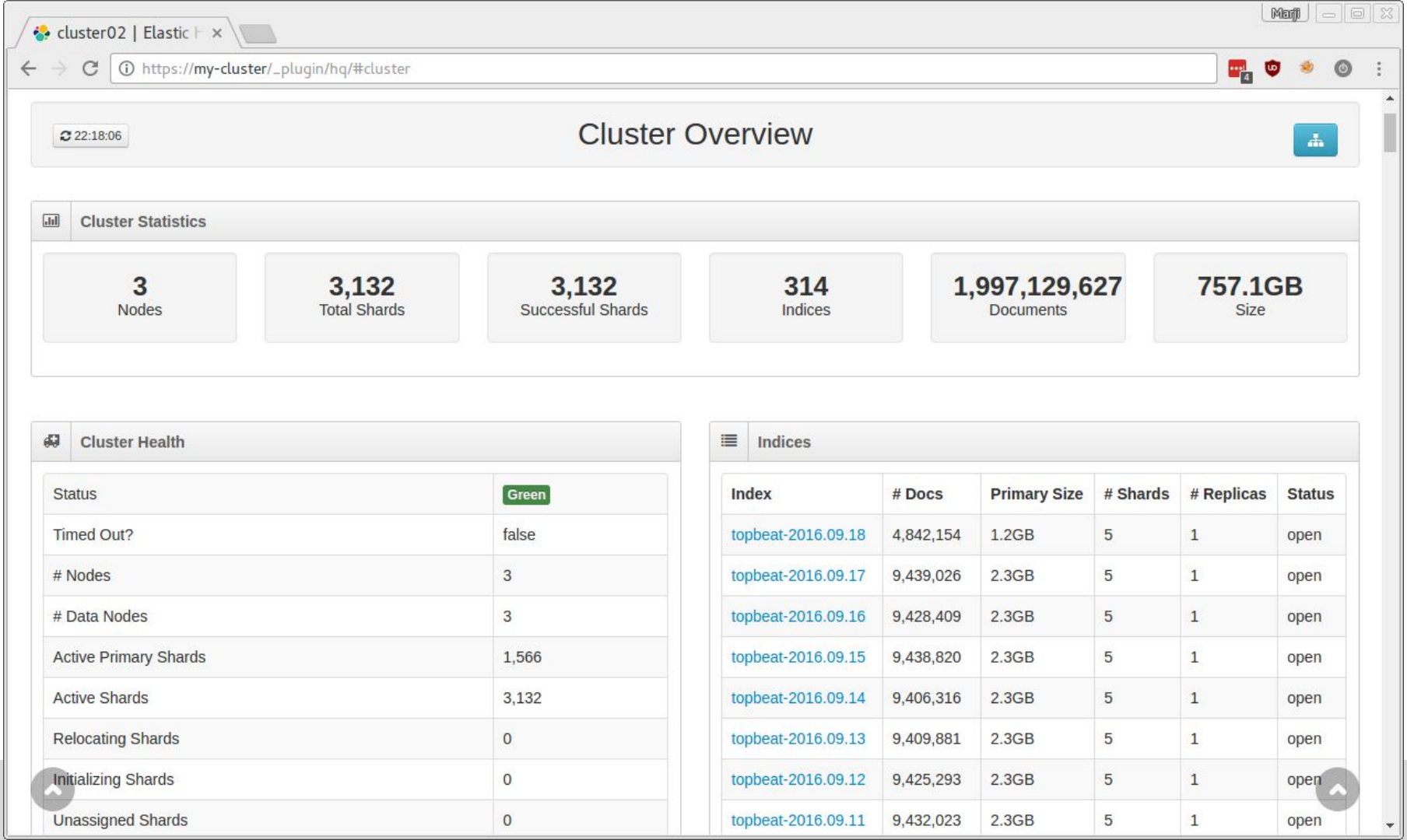
Kopf

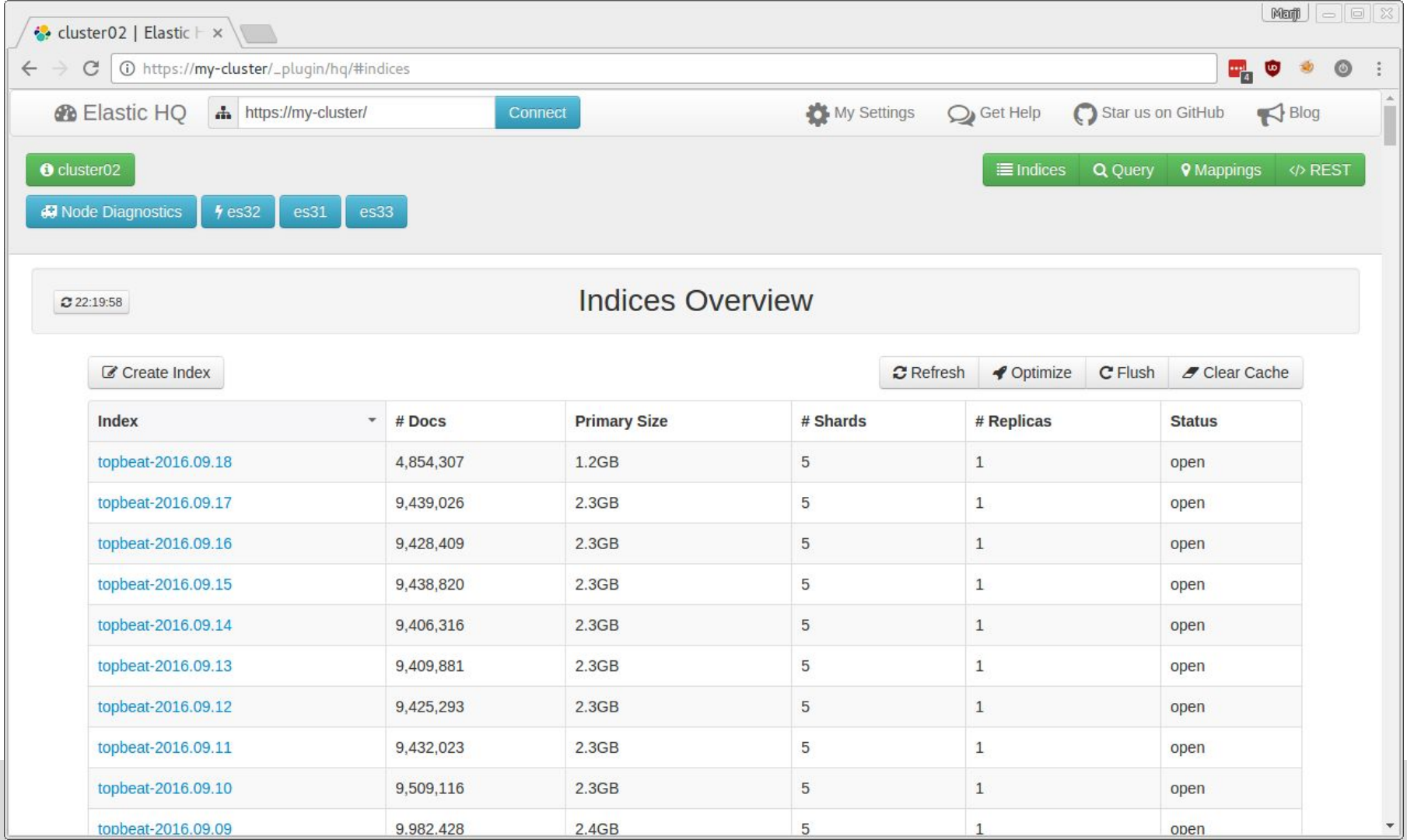

kopf[cluster02]
Marji
https://my-cluster/_plugin/kopf/#!/cluster
cluster
nodes
rest
more
cluster02 @ es32
3 nodes
311 indices
3,102 shards
1,996,299,633 docs ↑ 2,905
1.48TB
logstash
closed (0)
special (1)
filter nodes by name
1-4 of 62 selected indices
logstash-2016.07.01
shards: 5 * 2 | docs: 14,253,791 | size: 6.51GB
logstash-2016.07.02
shards: 5 * 2 | docs: 8,778,520 | size: 3.88GB
logstash-2016.07.03
shards: 5 * 2 | docs: 8,473,113 | size: 3.53GB
logstash-2016.07.04
shards: 5 * 2 | docs: 18,084,159 | size: 7.36GB
es31
10.42.1.16
es32
10.42.2.50
es33
10.42.3.188
heap
disk
cpu
load
show log


kopf[cluster02]
Marji
https://my-cluster/_plugin/kopf/#!/nodes
cluster
nodes
rest
more
cluster02 @ es32
3 nodes
311 indices
3,102 shards
1,996,309,549 docs 2,806
1.48TB 5.01MB
filter nodes by name
master
data
client
name
load average
cpu %
heap usage %
disk usage %
uptime
es31
10.42.1.16
10.42.1.16:9300
JVM: 1.8.0_91
ES: 2.3.5
0.3
6.0
56.0
used: 3.94GB
max: 6.97GB
52.0
free: 467.91GB
total: 984.18GB
13d.
es32
10.42.2.50
10.42.2.50:9300
JVM: 1.8.0_91
ES: 2.3.5
0.3
7.0
52.0
used: 3.69GB
max: 6.97GB
51.0
free: 479.21GB
total: 984.18GB
13d.
es33
10.42.3.188
es33/10.42.3.188:9300
JVM: 1.8.0_91
ES: 2.3.5
0.1
5.0
57.0
used: 4.04GB
max: 6.97GB
50.0
free: 490.99GB
total: 984.18GB
13d.

Elasticsearch web admin plugins

- Kopf
- Elastic HQ


cluster02 | Elastic
https://my-cluster/_plugin/hq/#cluster
22:18:06
Cluster Overview
Cluster Statistics
3
Nodes
3,132
Total Shards
3,132
Successful Shards
314
Indices
1,997,129,627
Documents
757.1GB
Size
Cluster Health
Status
Green
Timed Out?
false
Nodes
3
Data Nodes
3
Active Primary Shards
1,566
Active Shards
3,132
Relocating Shards
0
Initializing Shards
0
Unassigned Shards
0
Indices
Index
Docs
Primary Size
Shards
Replicas
Status
topbeat-2016.09.18
4,842,154
1.2GB
5
1
open
topbeat-2016.09.17
9,439,026
2.3GB
5
1
open
topbeat-2016.09.16
9,428,409
2.3GB
5
1
open
topbeat-2016.09.15
9,438,820
2.3GB
5
1
open
topbeat-2016.09.14
9,406,316
2.3GB
5
1
open
topbeat-2016.09.13
9,409,881
2.3GB
5
1
open
topbeat-2016.09.12
9,425,293
2.3GB
5
1
open
topbeat-2016.09.11
9,432,023
2.3GB
5
1
open


cluster02 | Elastic H
https://my-cluster/_plugin/hq/#indices
Elastic HQ
https://my-cluster/
Connect
My Settings
Get Help
Star us on GitHub
Blog
cluster02
Indices
Query
Mappings
REST
Node Diagnostics
es32
es31
es33
22:19:58
Indices Overview
Create Index
Refresh
Optimize
Flush
Clear Cache
Index
Docs
Primary Size
Shards
Replicas
Status
topbeat-2016.09.18 4,854,307 1.2GB 5 1 open
topbeat-2016.09.17 9,439,026 2.3GB 5 1 open
topbeat-2016.09.16 9,428,409 2.3GB 5 1 open
topbeat-2016.09.15 9,438,820 2.3GB 5 1 open
topbeat-2016.09.14 9,406,316 2.3GB 5 1 open
topbeat-2016.09.13 9,409,881 2.3GB 5 1 open
topbeat-2016.09.12 9,425,293 2.3GB 5 1 open
topbeat-2016.09.11 9,432,023 2.3GB 5 1 open
topbeat-2016.09.10 9,509,116 2.3GB 5 1 open
topbeat-2016.09.09 9,982,428 2.4GB 5 1 open

Getting logs from Drupal to ELK

Drupal Watchdog logs - shipping

Logstash drupal_dblog input filter

- not for production!

```
input {
  drupal_dblog {
     databases =>
       ["site1", "mysql://usr:pass@host/db"]
     interval => "1"
   }
}
```


Drupal Watchdog logs - shipping

Via syslog

1) Enable Drupal syslog module
2) Configure server rsyslog to write to dedicated logfile:

```
create e.g. /etc/rsyslog.d/60-drupal.conf:

local0.* /var/log/drupal.log
```


Drupal Watchdog logs - shipping

Via syslog

3) Use filebeat to stream the log lines to logstash

```
filebeat:
  prospectors:
    -
      paths:
         - /var/log/drupal.log

      input_type: drupalsyslog

output:
  logstash:
    hosts: ["logstash.example.com:9876"]
```


Drupal Watchdog logs - processing

Logstash grok filter - many pre-defined patterns:

```
- GREEDYDATA .*
- USERNAME [a-zA-Z0-9._-]+
- POSINT \b(?:[1-9][0-9]*)\b
```

Drupal Watchdog logs - processing

Logstash grok filter - define your owns:

WATCHDOG

```
https?://%{HOSTNAME:drupal_vhost}\|%{NUMBER:drupal_timestamp}\|(?<drupal_action>[^\|]*)\|%{IP:drupal_ip}\|(?<drupal_request_uri>[^\|]*)\|(?<drupal_referer>[^\|]*)\|(?<drupal_uid>[^\|]*)\|(?<drupal_link>[^\|]*)\|(?<drupal_message>.*)
```

```
https://stg.d8.com|1474269512|cron|127.0.0.1|https://stg.d8.com/||0||Cron run completed.
```

Drupal Watchdog logs - processing

Logstash grok filter - define your owns:

```
WATCHDOG
https?://%{HOSTNAME:drupal_vhost}\|%{NUMBER:drupal_timestamp}\|(?<
drupal_action>[^\|]*)\|%{IP:drupal_ip}\|(?<drupal_request_uri>[^\|
]*)\|(?<drupal_referer>[^\|]*)\|(?<drupal_uid>[^\|]*)\|(?<drupal_l
ink>[^\|]*)\|(?<drupal_message>.*)

SYSLOGWATCHDOG %{SYSLOGTIMESTAMP:logdate} %{IPORHOST:logsource}
%{SYSLOGHOST:syslogprog}: %{WATCHDOG}
```

Drupal Watchdog logs - processing

Logstash grok filter - use yours

```
filter {
  if [type] == "drupalsyslog" {
    grok {
      match => { "message" => "%{SYSLOGWATCHDOG}" }
    }
}
```

Drupal Watchdog logs - shipping

Via the “Logs HTTP” module

- Provides JSON event pushing to Logs via the tag/http endpoint.
- when the Logs syslog agent is not an option

Wrapping up

Wrapping up

It is not easy, evaluate SaaS options

Monitoring is a must

Links

Main docs area for the ELK stack:
https://www.elastic.co/guide/index.html

Deploying and Scaling Logstash
https://www.elastic.co/guide/en/logstash/current/deploying-and-scaling.html

Follow up blog post:
http://morpht.com/posts/ha-elk-drupal

Links

Logs for Drupal: Why You Need Them and How to Do It
https://www.loggly.com/blog/logs-for-drupal-why-you-need-them-and-how-to-do-it/

Questions?

Thank you!

@cermakm

DUBLIN
DRUPALCON

JOIN US FOR CONTRIBUTION SPRINTS

First Time Sprinter Workshop - 9:00-12:00 - Room Wicklow 2A

Mentored Core Sprint - 9:00-18:00 - Wicklow Hall 2B

General Sprints - 9:00 - 18:00 - Wicklow Hall 2A


DUBLIN
DRUPALCON
WHAT DID YOU THINK?
Evaluate This Session
events.drupal.org/dublin2016/schedule
THANK YOU!