elastic

Monitoring All the Things! on your Linux system with the Elastic Stack

Joshua Rich
Principal Support Engineer, Elastic
linux.conf.au 2018 Sysadmin Miniconf

THE HYPE
IS OVER 9000
memegenerator.net

Agenda

What we **will** discuss today :)

1 About the Elastic Stack

2 About Metricbeat

3 Elasticsearch for metrics?

4 Quick walkthrough of Metricbeat

5 Where to go from here...

What is the Elastic Stack?

Kibana gives shape to your data and is the extensible user interface for configuring and managing all aspects of the Elastic Stack.

Elasticsearch is a distributed, JSON-based search and analytics engine designed for horizontal scalability, maximum reliability, and easy management.

Beats is a platform for lightweight shippers (note: plural) that send data from edge machines to Logstash and Elasticsearch.

Logstash is a dynamic data collection pipeline with an extensible plugin ecosystem.

What is Metricbeat?

- One of many shippers built on the Beats platform
- Reads from local data sources
- Sends data to Elasticsearch or Logstash or Redis
- Deploy once per server, consume many local services
- Single binary and a few config files
- Docker ready!

What can Metricbeat monitor?

A lot of things...

Apache	Ceph	Couchbase	Docker
Etcd	HAproxy	Kafka	Memcached
MongoDB	MySQL	Nginx	PHP-FPM
PostgreSQL	RabbitMQ	Redis	**System***

This is just a sample of, full list here:

https://www.elastic.co/guide/en/beats/metricbeat/current/metricbeat-modules.html

The Metricbeat System Module

What it monitors:

- CPU usage statistics totals and per core breakdown, system load
- Filesystem usage and statistics (no. of files, used/free)
- Disk IO totals and per device
- Memory usage (total and free/used)
- Network IO totals and per device breakdown
- Per process statistics (can be filtered)
- TCP sockets
- Uptime

> *I can use Elasticsearch for metrics?*
>
> *Isn’t Elasticsearch for text search, not number search?*

Everybody. All the time.

Elasticsearch Loves Numbers!

Storage

- BKD-tree data structures for numeric types
- Range data types for storing a value range
- Half-float and scaled-float data types for confined numeric ranges
- Sparse field storage improvements
- Index sorting

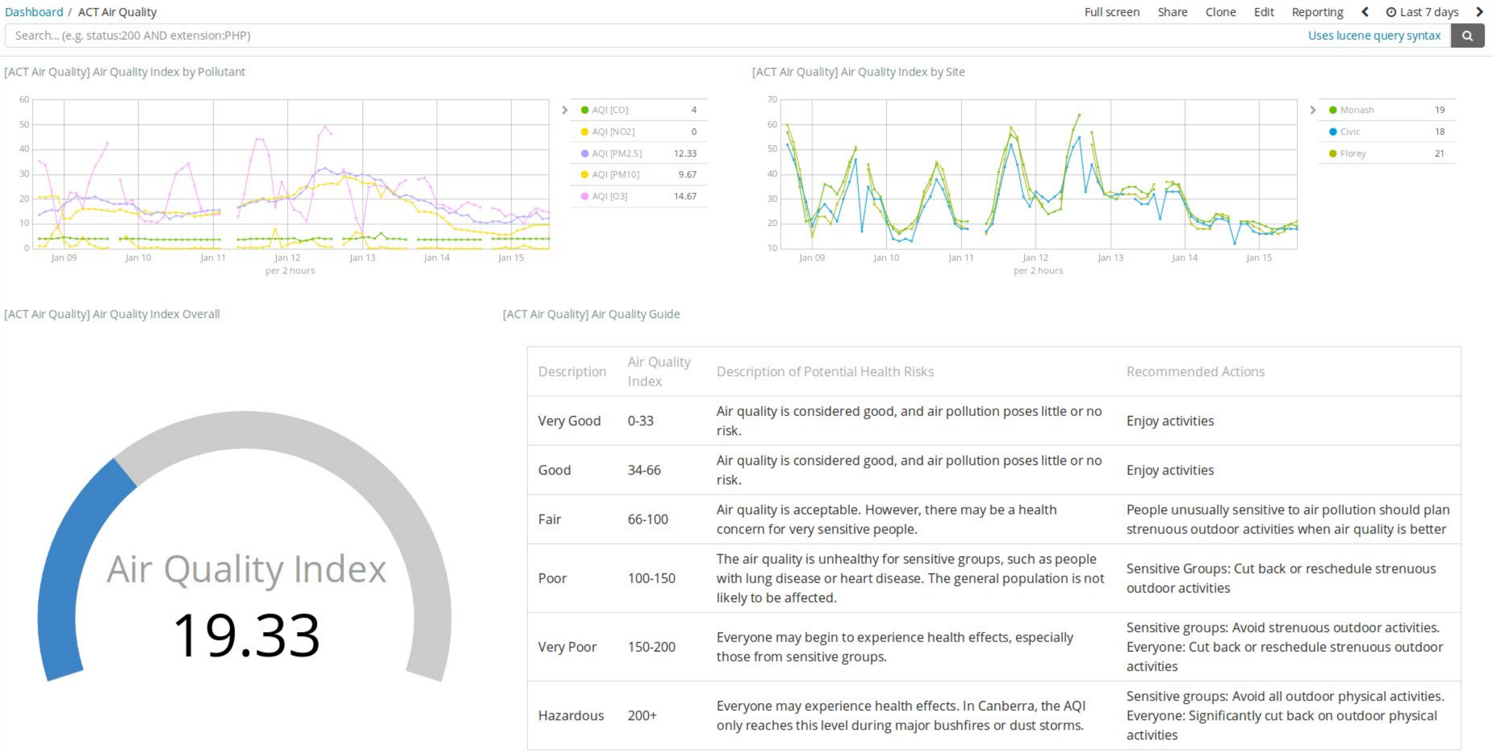
Search

- Per shard query and filter cache
- Query rewriting to avoid shards that don’t match

Management

- Rollover API for ease of retention requirements
- Shrink/Split APIs for helping with data growth
- Reindex API and Ingest Node to help with data management

TSVB In Kibana makes for lovely metric visualisations!



Description	Air Quality Index	Description of Potential Health Risks	Recommended Actions
Very Good	0-33	Air quality is considered good, and air pollution poses little or no risk.	Enjoy activities
Good	34-66	Air quality is considered good, and air pollution poses little or no risk.	Enjoy activities
Fair	66-100	Air quality is acceptable. However, there may be a health concern for very sensitive people.	People unusually sensitive to air pollution should plan strenuous outdoor activities when air quality is better
Poor	100-150	The air quality is unhealthy for sensitive groups, such as people with lung disease or heart disease. The general population is not likely to be affected.	Sensitive Groups: Cut back or reschedule strenuous outdoor activities
Very Poor	150-200	Everyone may begin to experience health effects, especially those from sensitive groups.	Sensitive groups: Avoid strenuous outdoor activities. Everyone: Cut back or reschedule strenuous outdoor activities
Hazardous	200+	Everyone may experience health effects. In Canberra, the AQI only reaches this level during major bushfires or dust storms.	Sensitive groups: Avoid all outdoor physical activities. Everyone: Significantly cut back on outdoor physical activities

Walkthrough: Monitoring your Linux System Stats

with Elasticsearch, Kibana and Metricbeat

elastic

Start Elasticsearch and Kibana

Let’s use Docker containers for easy deploy:

```
# Start Elasticsearch:
docker run -p 9200:9200 -p 9300:9300 \
    -e "discovery.type=single-node" \
    --name elasticsearch \
    docker.elastic.co/elasticsearch/elasticsearch-oss:6.1.1

# Start Kibana:
docker run -p 5601:5601 \
    --name kibana --link elasticsearch \
    docker.elastic.co/kibana/kibana-oss:6.1.1
```

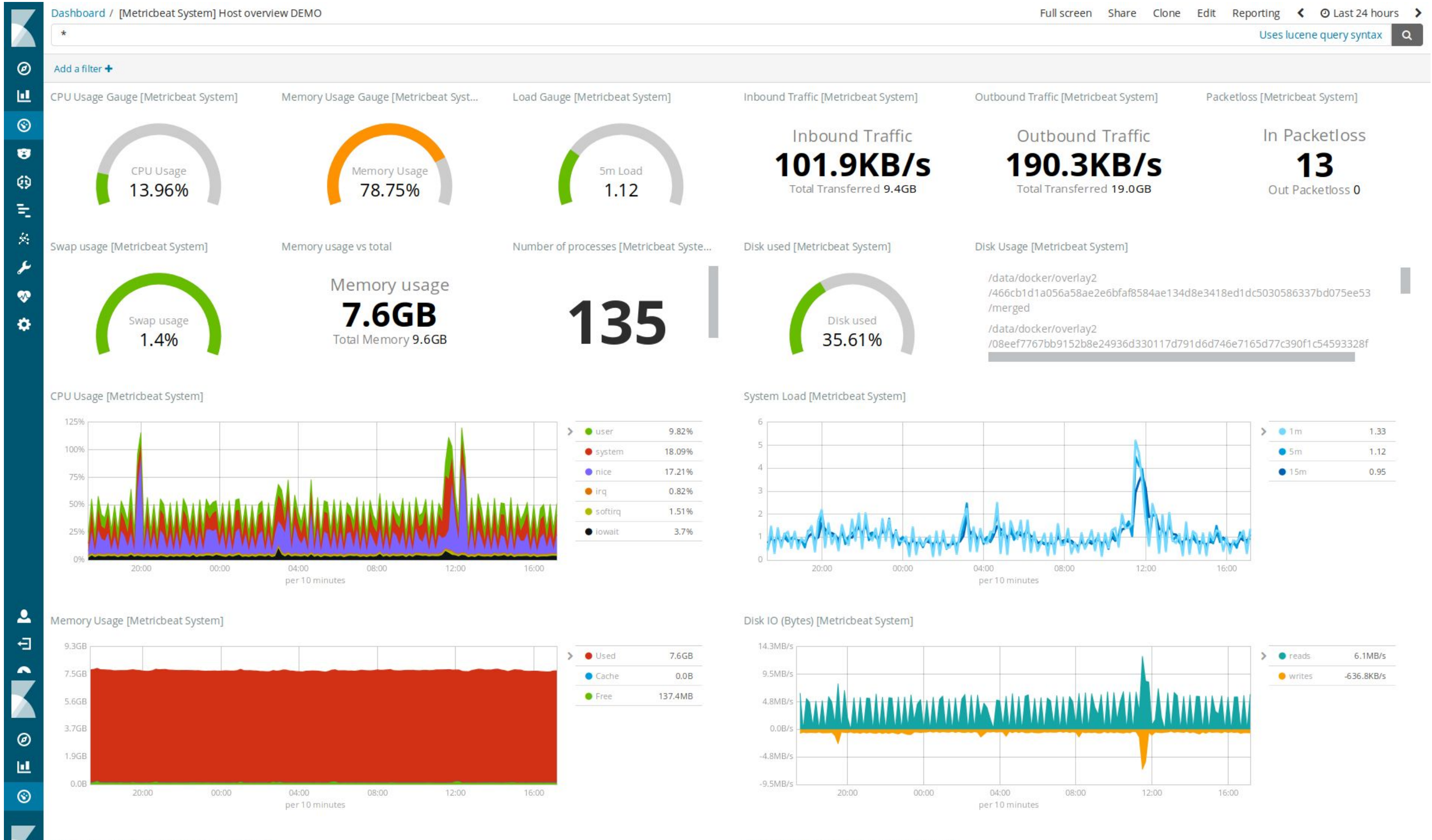
Start Metricbeat

Let’s use Docker containers for easy deploy:

```
# Start Metricbeat:
docker run \
       --volume="/proc:/hostfs/proc:ro" \
       --volume="/sys/fs/cgroup:/hostfs/sys/fs/cgroup:ro" \
       --volume="/:/hostfs:ro" \
       --net=host --name metricbeat \
       docker.elastic.co/beats/metricbeat:6.1.1 \
       -system.hostfs=/hostfs -E \
       output.elasticsearch.hosts="[http://localhost:9200]" \
       -setup

# Note: no need to pass -setup on future runs...
```

Open http://localhost:5601 & view the pretty dashboards



CPU Usage Gauge [Metricbeat System]: CPU Usage 13.96%

Memory Usage Gauge [Metricbeat Syst...: Memory Usage 78.75%

Load Gauge [Metricbeat System]: 5m Load 1.12

Inbound Traffic [Metricbeat System]: Inbound Traffic 101.9KB/s, Total Transferred 9.4GB

Outbound Traffic [Metricbeat System]: Outbound Traffic 190.3KB/s, Total Transferred 19.0GB

Packetloss [Metricbeat System]: In Packetloss 13, Out Packetloss 0

Swap usage [Metricbeat System]: Swap usage 1.4%

Memory usage vs total: Memory usage 7.6GB, Total Memory 9.6GB

Number of processes [Metricbeat Syste...: 135

Disk used [Metricbeat System]: Disk used 35.61%

Disk Usage [Metricbeat System]:

/data/docker/overlay2
/466cb1d1a056a58ae2e6bfaf8584ae134d8e3418ed1dc5030586337bd075ee53
/merged

/data/docker/overlay2
/08eef7767bb9152b8e24936d330117d791d6d746e7165d77c390f1c54593328f

CPU Usage [Metricbeat System]

● user	9.82%
● system	18.09%
● nice	17.21%
● irq	0.82%
● softirq	1.51%
● iowait	3.7%

System Load [Metricbeat System]

● 1m	1.33
● 5m	1.12
● 15m	0.95

Memory Usage [Metricbeat System]

● Used	7.6GB
● Cache	0.0B
● Free	137.4MB

Disk IO (Bytes) [Metricbeat System]

● reads	6.1MB/s
● writes	-636.8KB/s

Where do you go from here?

Listen to the beat of all your infrastructure...

Install

- Metricbeat on all your servers.
- Separate Elasticsearch and Kibana servers.

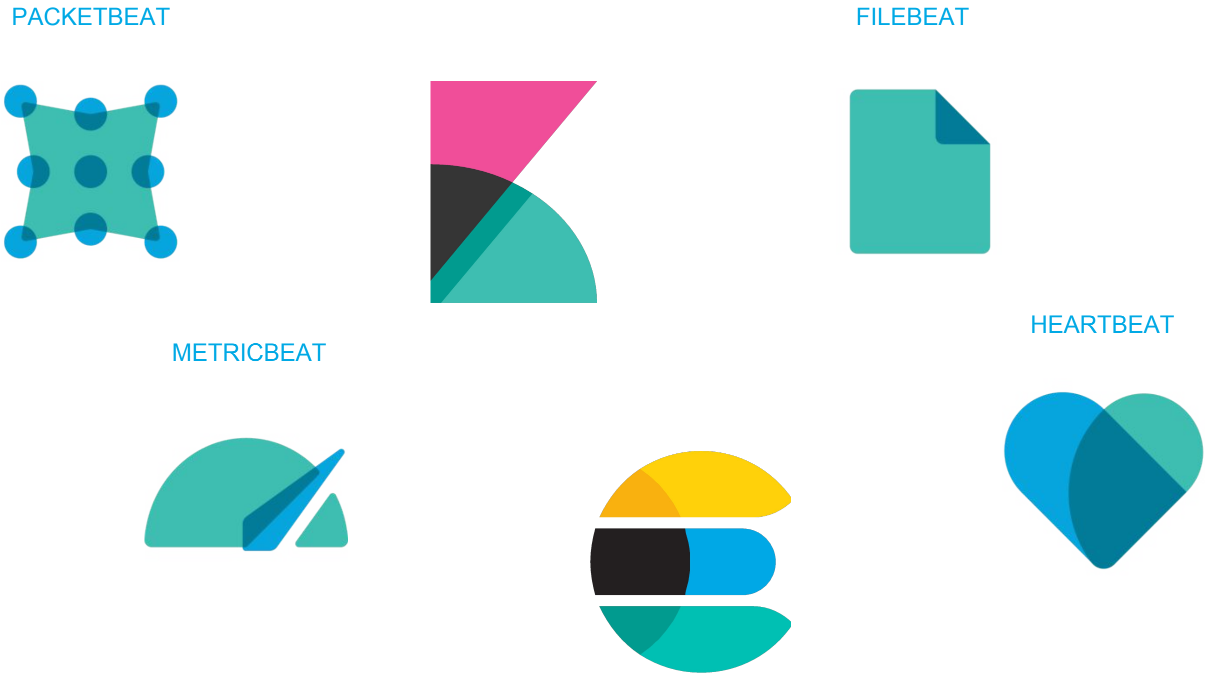
Configure

- Configure Metricbeat on each server according to the services it runs.
- Configure all Metricbeat instances to index to Elasticsearch.

Visualise

- Run Metricbeat setup to import the dashboards once only.
- Visualise, monitor and explore your servers and services in a central web UI.

But we can go further...



PACKETBEAT

FILEBEAT

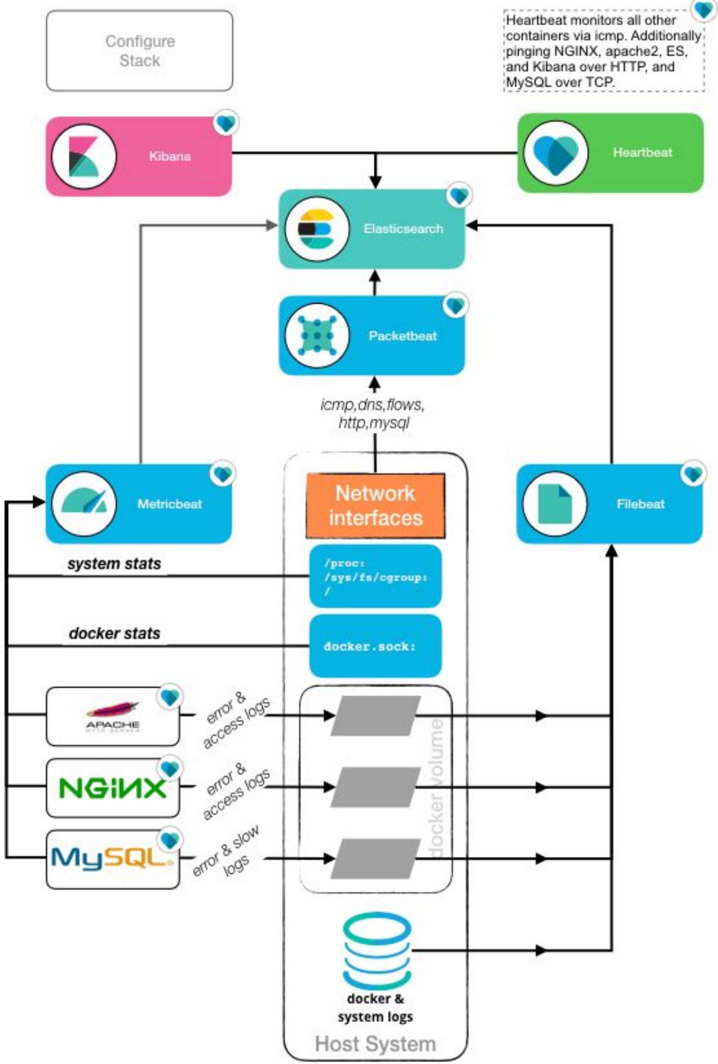
METRICBEAT

HEARTBEAT

“Full Stack” Example

Dockerized Elastic+Web+DB Stack

- Elasticsearch, Kibana:
- Metricbeat monitoring:
  - System stats
  - Docker stats (i.e., everything here)
- + Filebeat monitoring:
  - Apache/Nginx error/access logs
  - MySQL error/slow logs
  - System logs
- + Packetbeat monitoring:
  - ICMP, DNS, HTTP and MySQL traffic
  - TCP flows
- + Heartbeat monitoring:
  - ICMP
  - Apache, Nginx, MySQL, Kibana, Elasticsearch health checks



Heartbeat monitors all other containers via icmp. Additionally pinging NGINX, apache2, ES, and Kibana over HTTP, and MySQL over TCP.

See what’s possible:

github.com/elastic/examples/tree/master/Miscellaneous/docker/full_stack_example

References

Explore the topics covered in this presentation!

- https://www.elastic.co/blog/minimize-index-storage-size-elasticsearch-6-0
- https://www.elastic.co/blog/numeric-and-date-ranges-in-elasticsearch-just-another-brick-in-the-wall
- https://www.elastic.co/blog/index-sorting-elasticsearch-6-0
- https://www.elastic.co/guide/en/elasticsearch/reference/current/number.html#_which_type_should_i_use
- https://github.com/elastic/examples/tree/master/Miscellaneous/docker/full_stack_example

We ♥ our open-source community: www.elastic.co/community

Contact me:

@devopswannabe

www.linkedin.com/in/joshuarichau