Logging in Kubernetes with Fluentd sidecar containers

InfraCoders VI

Bostjan Bozic, A1 Telekom Austria AG

Agenda

- Overview
- Fluentd
- Architecture
- Demo

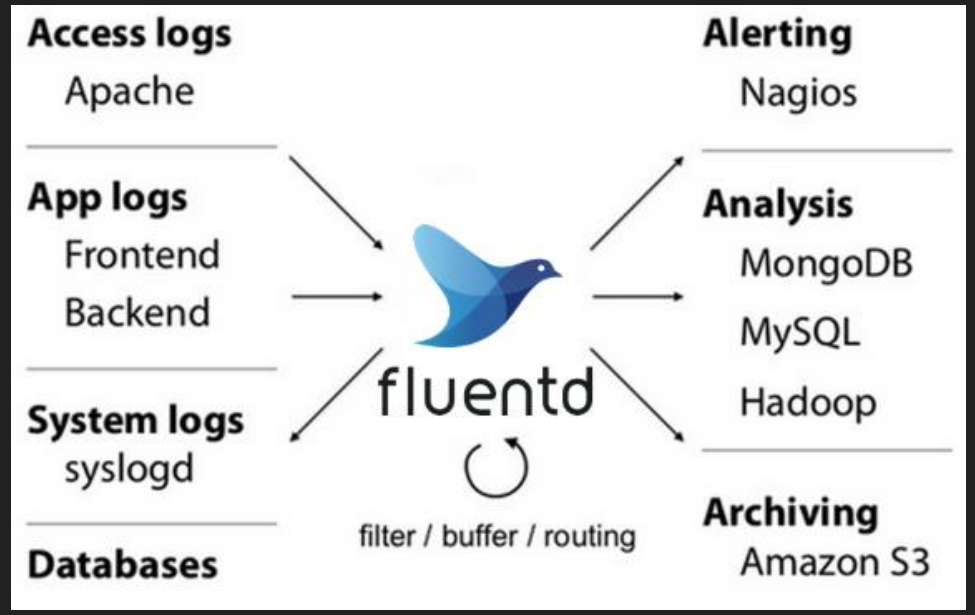
Overview

- Pod logs accessible via `kubectl logs <podname>`
- What happends if pod crashes or becomes inaccessible?

- Solution – permanent log storage:
  - Store log files in centralized location (e.g. S3)
  - Stream logs to [distributed] database (e.g. Elasticsearch, MongoDB)

Fluentd

- Opensource data collector for unified logging layer
- Unified logging with JSON
- Pluggable architecture
- Minimum resources required
- Built-in reliability

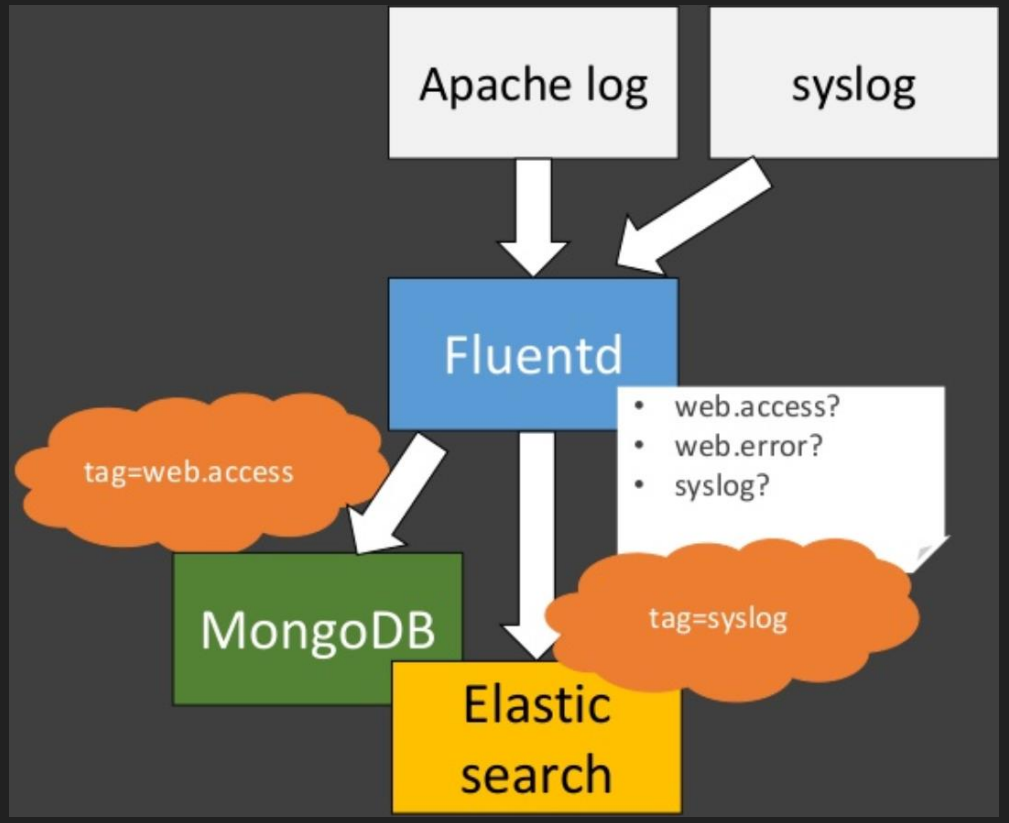


Fluentd Plugins

- Input plugins
- Filter plugins
- Output plugins
- Parser plugins (nested in input plugins)
- Formatter plugins (nested in output plugins)
- Buffer plugins (nested in output plugins)
- Storage plugins (nested in input, filter and output plugins)

Fluentd Configuration

- Consists of following directives:
  - Source
  - Match
  - Filter
  - System
  - Label
  - @include



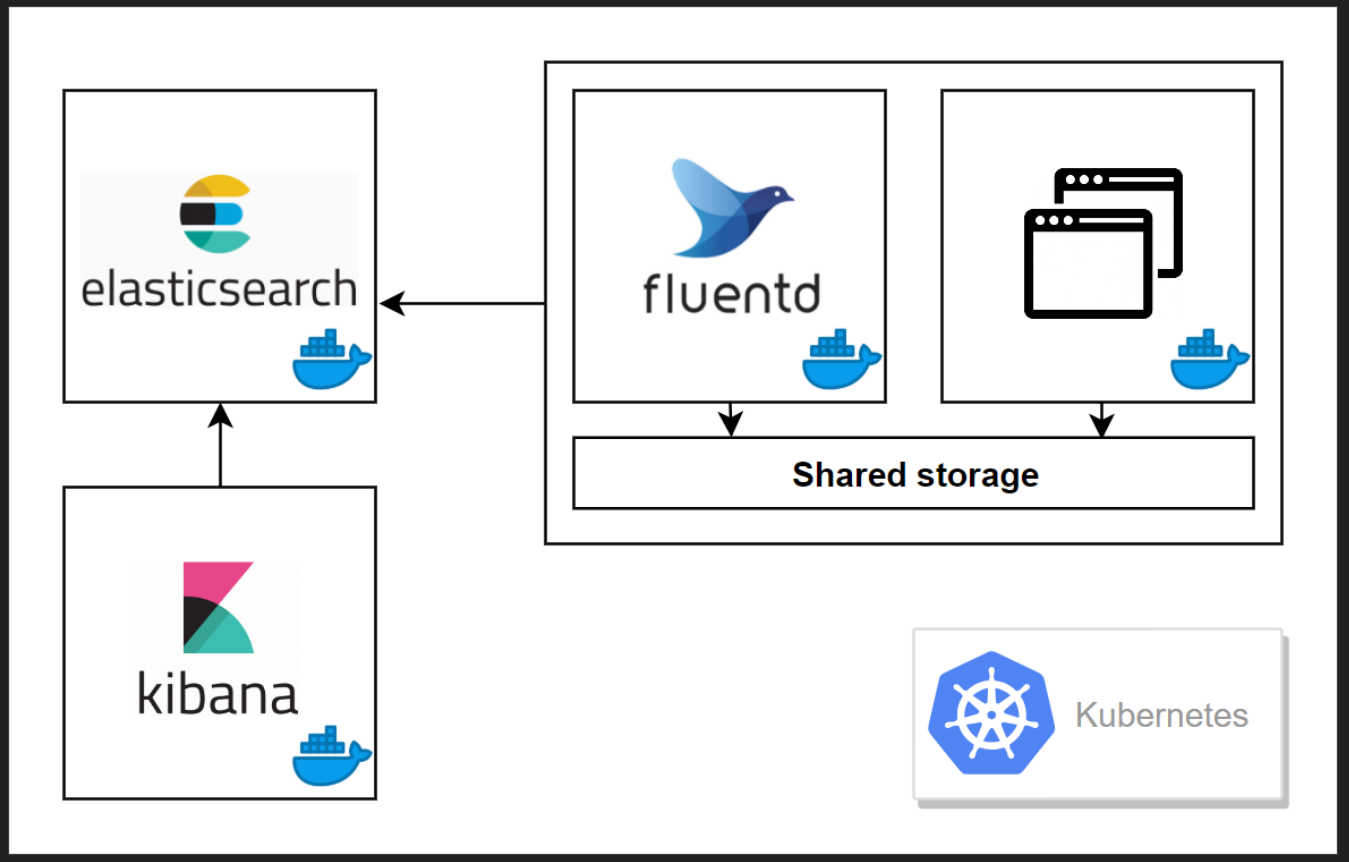
Fluentd Configuration

- 2 sources - `forward` and `dstat` type
- filter applied only to `forward` type source
- `forward` logs persisted as file
- `dstat` type source skips to `label` part and persists data in Elasticsearch

```
<source>
  @type forward
  tag app.infracoders
</source>

<source>
  @type dstat
  @label @METRICS
</source>

<filter app.**>
  @type record_transformer
  <record>
    host_param "#{Socket.gethostname}"
  </record>
</filter>

<match app.**>
  @type file
  path /var/log/fluent/infracoders
</match>

<label @METRICS>
  <match **>
    @type elasticsearch
    host elasticsearch
    port 9200
    scheme http
  </match>
</label>
```

Architecture

- Running application within k8s pod
- Fluentd log collector
- Elasticsearch for data storage
- Kibana for visualization



Demo

Q&A

Thank you

https://github.com/BostjanBozic/infracoders.git