Sane Fluentd for multi-tenant Kubernetes clusters

kubernetes

What is a multi-tenant Kubernetes?

A broad term but often amounts to:

- ”Soft” tenancy model: groups within an organization sharing the same cluster
- RBAC enabled
- Namespace as the tenant boundary

Why is logging hard?

- Logs don’t have K8S metadata attached by default
- Logs are just files in /var/log/containers
- There are non-K8S logs that are still releveant (journald)
- You want to route logs from different apps/namespaces to different sinks (Splunk, ELK, Loggly, AMQP, etc.)
- You don’t want containers bind-mount the host filesystem
- Configuring log aggregation is error prone

Fluentd helps but is not enough

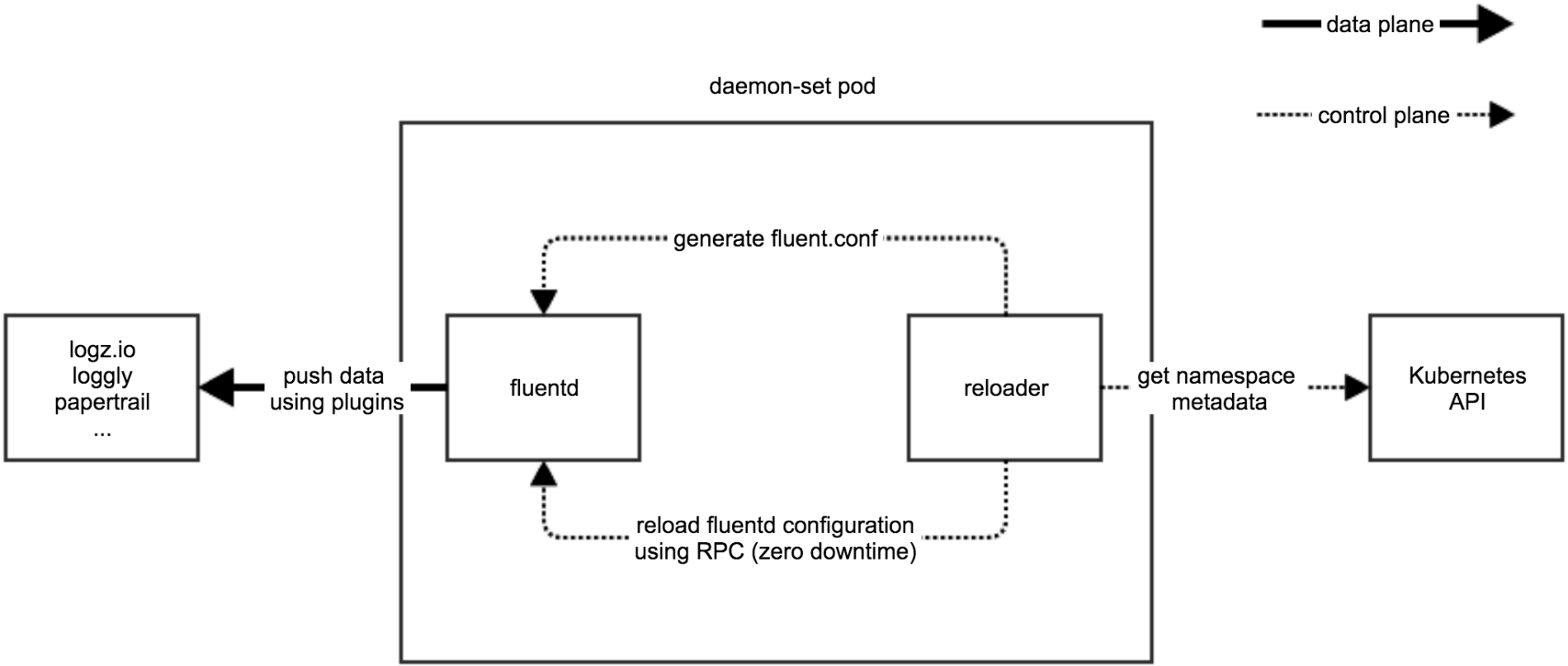
- Collects K8S metadata
- Input plugins that read from files
- Many output plugins already available (most of them work)

- No dynamic configuration reloading
- Single monolithic config file
- No log splitting based on K8S metadata
- No “battery included” image with all required plug-ins
- A single Fluentd administrator is assumed
- Need to be careful with .pos and buffer files

kube-fluentd-operator (KFO) provides what's missing

- A single Daemonset to deploy
- No CRDs – can run on very old versions of K8S
- It runs a stock image with a curated list of plug-ins
- A sidecar "reloader" (in GO + 4% Ruby) image which compiles a configuration file for Fluentd
- No unnecessary reloads: only if the checksum of the parsed tree has changed
- This file is built by combining segments defined in the namespaces
- Every namespace config is validated in isolation: namespaces with valid configuration are not impacted
- Attach cluster-level metadata (region, cluster-name, department, etc.)

What's inside the DaemonSet?



Example 1:

```
apiVersion: v1
kind: ConfigMap
metadata:
  name: fluentd-config
data:
  fluent.conf: |
    <match **>
      @type elastisearch
      # configuration omitted
    </match>
```

- Create ConfigMap fluentd-config
- Namespaces without such a ConfigMap are ignored
- Provide the usual Fluentd configuration in the ”fluent.conf” item
- This example will send all logs from this namespace to some ELK

Example 2:

```
apiVersion: v1
kind: ConfigMap
metadata:
  name: fluentd-config
data:
  fluent.conf: |
    <match $labels(app=consul, release=my-consul)>
      @type null
    </match>
    <match **>
      @type elastisearch
      # configuration omitted
    </match>
```

- This will send all logs from all pods to ELK but will first discard all logs with labels app=consul and release=my-consul
- $labels() is a macro
- ** means all logs from this namesapce

KFO processes your fluent.conf's

- ** syntax is expanded to the current namespace (the one hosting the ConfigMap)
- **$thisns** gets expanded the current namespace (rarely needed thoug)
- **$labels()** is macro that translates to about 30 lines of Fluentd configure
- **@pipeline** is rewritten to @pipeline-{hash} so that labels from different namespaces don't collide
- **buffer_path** parameters get rewritten too to avoid collisions
- **mounted-file** is a also a macro which lets you export logs from the containers filesystem
- Finally, the processed file is tested in isolation
- Only **kube-system** namespace is left untouched

KFO’s limitations

- Cannot declare arbitrary `<source>` directives (well, the K in KFO)
- Cannot change the tag of a log: it is bound forever to kube.{namespace}.{podid}
- Validation depends on the plug-in: if a plug-in cannot detect an error, neither can KFO
- Doesn’t support multiple tags, for example `<filter a b>`

What needs to be improved

- Multiple tags `<filter a b>`
- Reload configuration on event, not polling
- Write status annotation only once

DEMO

Agenda

- Setup
  - A single Kubernetes cluster
  - Two namespaces: **london** and **paris**
  - Two apps: “**cat**” and “**dog**” logging a “sound” in French and English to stdout and to a file
- Configuration
  - Collect logs from **london** to loggly.com and **paris** to ~~papertrail.com~~
  - Collect logs from all “**cat**” to loggly and all “**dog”** ~~to papertrail~~
  - Collect file output to ~~humio.com~~

Deploy KFO

```
# link to the Github release
CHART_URL='https://github.com/vmware/kube-fluentd-operator/releases/download/v1.7.0/
log-router-0.2.5.tgz'

helm install --name kfo ${CHART_URL} \
   --set rbac.create=true \
   --set image.tag=v1.7.0 \
   --set image.repository=jvassev/kube-fluentd-operator \
   --set meta.key=csp \
    --set meta.values.region=eu-west-2 \
   --set meta.values.cluster=demo
```

There is more…

- “Sharing” logs between namespaces: for example access logs from a shared ingress controller
- Multi-line aggregator implemented as a filter
- Systemd journals end up with a `systemd.{unit}` tag ;can get the from the kube-system namespace, for example `systemd.kubelet`
- Templates & images are easily customizable
- K8S logs are parsed: `I0725 19:05:31.119316 1 kernel_monitor.go:93] …`

Resources

- README: https://github.com/vmware/kube-fluentd-operator
- Today's demo: under folder meetup-2018-11-22
- Get involved!

Q & A

kubernetes