Accessing the Kubernetes Logs

Piotr Gaczkowski
IT CONSULTANT

@doomhammerng doomhammer.info

Overview

Different types of logs that Kubernetes presents

Log management

Log rotation

Kubernetes Architecture

Nodes

Pods

Containers

Control Plane

Log Rotation

Saves disk space

Keeps logs small and viewable

Archives logs

Complies with retention policy

Deletes old logs to prevent overflow

Logs Presented by Nodes

`kubectl describe node`

System component logs
(`journalctl`, `syslog`, etc.)

kubectl describe node

```
Name:               pool-pluralsight-fbo6
Roles:              <none>
Labels:             beta.kubernetes.io/arch=amd64
                    beta.kubernetes.io/instance-type=s-1vcpu-2gb
                    beta.kubernetes.io/os=linux
Annotations:        io.cilium.network.ipv4-cilium-host: 10.244.0.107
                    io.cilium.network.ipv4-health-ip: 10.244.0.57
                    io.cilium.network.ipv4-pod-cidr: 10.244.0.0/25
                    node.alpha.kubernetes.io/ttl: 0
```

Logs Presented by Pods and Containers

`kubectl logs`

`kubectl describe pod`

kubectl logs Arguments


```
kubectl logs --previous [-p]
kubectl logs --container [-c]
kubectl logs --label [-l]
kubectl logs --all-containers
```

Examples for `kubectl logs`

```
# Find all pods with a given label, select the first one and show its logs

kubectl logs $(kubectl get pod -l app=run-controller -o
jsonpath="{.items[0].metadata.name}") run-controller


# Show logs of all the pods with a given label

kubectl logs -l app=run-controller
```

kubectl logs Arguments


```
kubectl logs --since=1h

kubectl logs --since-time= “2014-06-07T23:20:50.52Z”

kubectl logs --tail

kubectl logs --follow [-f]
```

`kubectl logs` for Objects Other Than Pods

Examples from kubectl man page

```
# Return snapshot logs from first container of a job named hello
kubectl logs job/hello

# Return snapshot logs from container nginx-1 of a deployment named nginx
kubectl logs deployment/nginx -c nginx-1
```

Logs Presented by the Control Plane

`kubectl get events`

System component logs (`journalctl`, `syslog`, etc.)

Service provider logs

Filtering kubectl get events Results

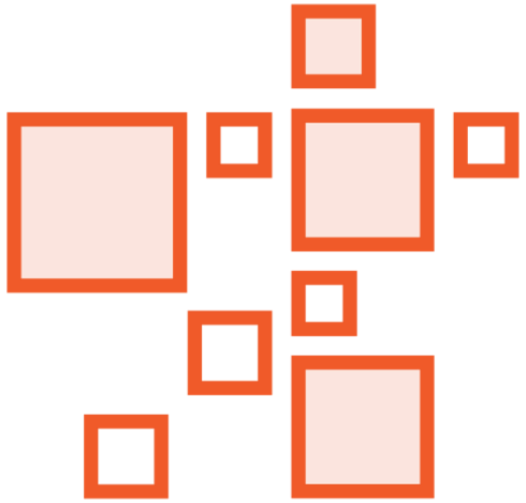
```
kubectl get events --field-selector type=Warning

kubectl get events --field-selector involvedObject.kind!=Pod

kubectl get events --field-selector
involvedObject.kind=Node,involvedObject.name=minikube

kubectl get events --field-selector type!=Normal
```

kubectl top



`kubectl top pod`

`kubectl top node`

Examples for kubectl top

```
# Show the resource utilization of all the pods matching a label

kubectl top pod -l app=run-controller


# Find all nodes, select the first one and show its utilization

kubectl top node $(kubectl get nodes -o
jsonpath="{.items[0].metadata.name}")
```

kubectl exec

Allows execution of arbitrary commands inside running containers

Can be used to call regular UNIX commands available inside the container (`top`, `ps`, `tail`, ...)

Allows interactive operation (`exec -ti`)

Examples for kubectl exec

```
# Find all pods with a given label, select the first one and

# run top in its default container

kubectl exec -ti $(kubectl get pod -l app=workout-gateway -o
jsonpath="{.items[0].metadata.name}") top


# Find all pods with a given label, select the first one and

# list the files in /app

kubectl exec $(kubectl get pod -l app=workout-gateway -o
jsonpath="{.items[0].metadata.name}") -- ls -l /app
```

Log Management

Node logging agent

Sidecar container with the logging agent

Exposing logs directly from the application

Demo

Read the Carved Rock Fitness application logs

See how log filtering works

Debug common problems with applications on Kubernetes

Summary

Kubernetes presents different types of logs that can aid in debugging

Several of `kubectl` subcommands help by providing relevant information

Navigating all the information manually is rather tedious