

other kubectl commands

- kubectl get nodes → List of nodes [registered]
- kubectl describe nodes → Description of nodes

notes:

POLES: Just a label

node-vol.kubernetes.io/...

↳ go through the Description of a node!

|| Taints ||

↳ You attach but not able

|| Automatically added ||

LABELS: Foundamental k8s
object concepts

key = value couple

that kubernetes wait for it's purpose (Processing)

Taints: Perform scheduled operations based on labels
choices

→ No schedule: if that label is present no schedule can be performed // no eviction will take place

→ No execute: Pods are evicted immediately UNLESS they are specifically instructed to tolerate the taint

→ Prefer no schedule: this to avoid scheduling here

useful when the node is under pressure!

Kublet can automatically add status label indicating the status of a node AND add taint to act as consequence

eg: node.kubernetes.io/memory-pressure

try to
run more stress
on the node?

check
kubectl
behaviour!!

common kubectl commands & concepts

(11)

namespaces: Folder to organize cluster object & organize compartmentalization

contexts: a way to change Default namespace (& others variables..)

k config set-context my-context --namespace=my-stuff

k config use-context my-context

k get pods -o jsonpath --template={.status.PodIP} }
 k get pods, services }
 }
 resource name } listed in the current namespace

k describe ~~pod~~ <resource-name> <name> } Detailed H2 Description

k explain pods } Description of every field of the object
 Pod.Spec

k apply -f x.YAML → apply transform yaml to objects in kub.

k delete -f x.YAML → remove the resources → Looks@ their name

k label pods bar color=red ⇒ add color label

k label pods bar color- ⇒ remove color key from labels

Debug

k logs pod-name --container container-name. } Print Logs

k exec -it pod-name --container container-name --bash } Drop you in bash

k attach -it pod-name --container container-name } last resource, allow you to give std input to the running Application!
 k port-forward <Pod-name> 8080:80

SERVICING A NODE

k cordon node → Mark as un-schedulable } -l flag / example of label usage to issue a command!
 k drain node → remove All apps
 k uncordon



K9S AN ALTERNATIVE TOOL TO MANAGE YOUR CLUSTER

an alternative view to the cluster now

- Pods → ~~ssh~~ spawn a bash in a container
- Services Describe a pod / Edit a pod...
- namespaces

Boys AFTER RBAC:
write cluster managing!

we have a few PODS, but how do they manage communication?

