Kubernetes Cmd

Kubernetes Tips

Installation

GPT CheatSheet

Helm Charts

- Delete/Create Configmap
    - kubectl delete configmap partner-api-sms-configmap -n phcore
    - kubectl create configmap partner-api-sms-configmap -n phcore --from-file=./conf
        - Inside this conf folder, at least should be an application.yaml file if it’s a springboot app for example
        - application.yaml must be the same as the application.yaml of the application project repository
- Apply a deployment (Navigate to deployment.yaml directory inside lens terminal or powershell)
    - kubectl apply -f deployment.yaml
        - When new docker image is generated, just run the above command, or if it's a new deployment as well.
            - No need for restarting the pod
- Restart deployment if deployment.yaml is not changed (and config map is changed for example)
    - kubectl rollout restart deployment partner-api-sms -n phcore
- Permanently delete a deployment (so the pod won’t create another one when just doing “delete pod”) - Useful when after deleting the respective service, we want to delete permanently the pod

    - kubectl delete deployment <deployment_name> -n <namespace>
        - Eg. : kubectl delete deployment external-api -n local-namespace
- Create new namespace: kubectl create namespace local-namespace
    - kubectl get namespaces (to see what namespaces are there)
- Force delete a PV (The pvc must be deleted first)
    - kubectl delete pv docker-registry-pv --grace-period=0 --force -n [Namespace_Name]
        - PV - Persistent Volume
        - PVC - Persistent Volume Claim
- Force Delete PVC
    - kubectl delete pvc nexus-pvc --grace-period=0 --force -n local-namespace

//////////////////////////

/////////////////////////////

- kubectx - alias to get contexts - needs to be installed
- kubens - alias to get namespaces - needs to be installed

Find namespaces, pods etc:

- kubectl get namespaces; kubectl get deployments etc...
- kubectl delete pods [POD_NAME]
- kubectl describe pod [POD_NAME] - Example see logs of pod
- Send jar or other file to the container
    - kubectl cp [LOCAL FILE PATH] [POD NAME]:[PATH INSIDE CONTAINER] Reference: https://howchoo.com/kubernetes/kubectl-cp-copy-files-to-from-pods

- Example:
    - kubectl cp ~/Desktop/svn_repos/galactus-storm/trunk/target/galactus-storm.jar workflow-engine-mbc-jazz-5b7d48b575-gbmbr:/opt/timwe/app/drum_galactus_storm/lib/galactus-storm.jar

- Grabbing files from INSIDE THE POD
- kubectl cp POD_NAME:FILE_PATH_INSIDE_CONTAINER LOCAL_DESTIONATION_FILE_PATH
  Reference:
  https://howchoo.com/kubernetes/kubectl-cp-copy-files-to-from-pods
- Example:
    - kubectl cp workflow-engine-mbc-jazz-5b7d48b575-gbmbr:/opt/timwe/app/drum_galactus_storm/lib/galactus-storm-1.0.1-SNAPSHOT.jar ~/Desktop/galactus-storm-1.0.1-SNAPSHOT.jar
      kubectl cp workflow-engine-mbc-jazz-5b7d48b575-gbmbr:/opt/timwe/log/drum_galactus_storm/galactus.log ~/Desktop/galactus.log

- -- Matheus Notes ---

The --watch flag keeps on listening for changes in the command run
Create a pod
Command syntax:
kubectl [name_of_pod] --image=[docker_image]
kubectl run my-pod --image=hello-world
Watch pods creations and changes of states:
kubectl get pods --watch
Describe pod
kubectl describe pod [podname]
Run a component file (pod, ConfigMap, service, etc) in a declarative way
kubectl apply -f [.yaml, .json]
Finding information about a Node
kubectl get nodes -o wide
Listing the ReplicaSets
kubectl get rs
Scale down a deployments file

```
kubectl scale deploy my-awesome-deployment --replicas=0
```

Checking the history of a deployment

```
kubectl rollout history deployment [deployment name]
```

Defining a semantic message to the latest change applied to a deployment file

```
kubectl annotate deployment [deployment name] kubernetes.io/change-cause="desired message"
```

Rolling back to a specific deployment version

```
kubectl rollout undo deployment [deployment name] --to-revision=[revision number]
```

Getting a resource from all Namespaces in the cluster

```
kubectl get [RESOURCE] --all-namespaces
```

Get YAML used to create resources

```
kubectl get [RESOURCE] [RESOURCE NAME] -o yaml
```

Sending files to Pod

```
kubectl cp [LOCAL FILE PATH] [POD NAME]:[PATH INSIDE CONTAINER]
```

Reference:
https://howchoo.com/kubernetes/kubectl-cp-copy-files-to-from-pods

Example:

```
kubectl cp ~/Desktop/svn_repos/galactus-storm/trunk/target/galactus-storm.jar workflow-engine-mbc-jazz-5b7d48b575-gbmbr:/opt/timwe/app/drum_galactus_storm/lib/galactus-storm.jar
```

Grabbing files from INSIDE THE POD

```
kubectl cp POD_NAME:FILE_PATH_INSIDE_CONTAINER LOCAL_DESTIONATION_FILE_PATH
```

Reference:
https://howchoo.com/kubernetes/kubectl-cp-copy-files-to-from-pods

Example:

```
kubectl cp workflow-engine-mbc-jazz-5b7d48b575-gbmbr:/opt/timwe/app/drum_galactus_storm/lib/galactus-storm-1.0.1-SNAPSHOT.jar ~/Desktop/galactus-storm-1.0.1-SNAPSHOT.jar
kubectl cp workflow-engine-mbc-jazz-5b7d48b575-gbmbr:/opt/timwe/log/drum_galactus_storm/galactus.log ~/Desktop/galactus.log
```

Binding a ClusterIP service port to a localhost port

```
kubectl port-forward services/[SERVICE NAME] [LOCAL_PORT]:[SERVICE_PORT]
```

Example:

```
kubectl port-forward services/workflow-engine 4567:4567
```

Deleting all evicted pods

1. kubectl get pod | grep Evicted | awk '{print $1}' | xargs kubectl delete pod

Cheat sheet:

https://kubernetes.io/pt-br/docs/reference/kubectl/cheatsheet/