**KUBERNETES TASK 5**

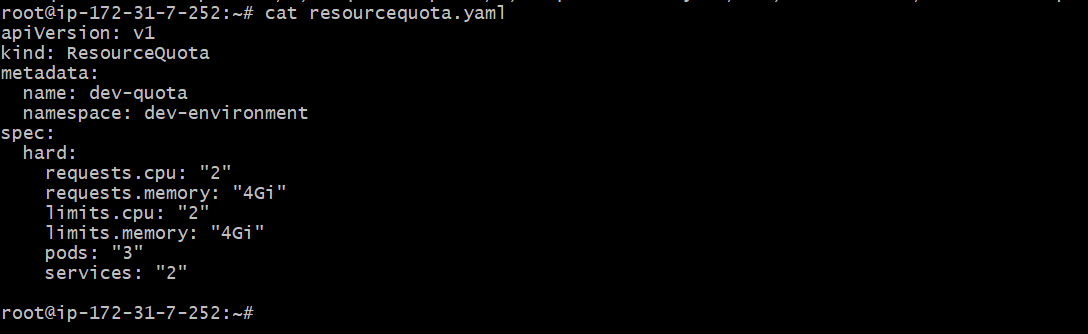
1. Create a namespace dev-environment and apply a resource-based quota that restricts the number of pods to 3 and services to 2.

To create name spaces use this command

kubectl create name space dev-environment

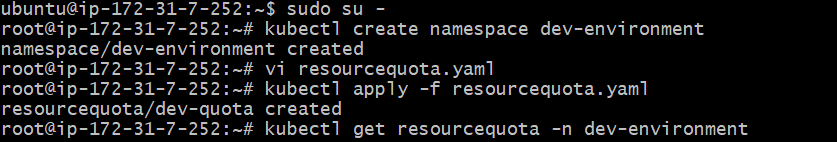
after created

use vi resourcequota.yaml



Now we need to apply the yaml file use this command

kubectl apply -f resourcequota.yaml



For checking the resources

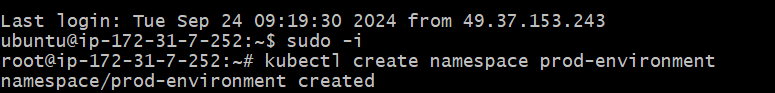
kubectl get resourcesquota -n dev-environment



1. Create a pod in the prod-environment namespace with 0.2 CPU and 200Mi memory requests, and 0.5 CPU and 500Mi memory limits.

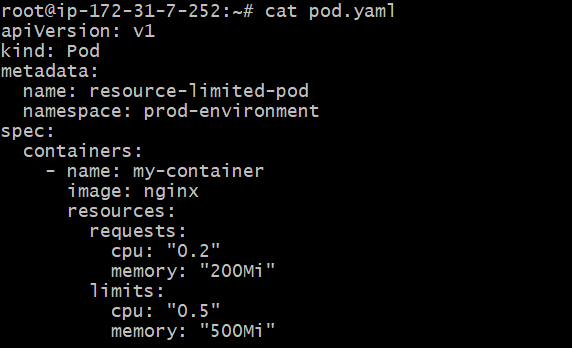
To create pod in prod-env

Kubectl create namespace prod-environment



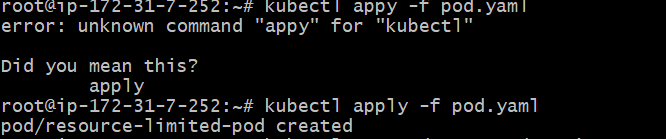
Create yaml file using vi command

vi pod.yaml



after that apply the yaml file using

kubectl apply -f pod.yaml



to check the running pods in namesspaces

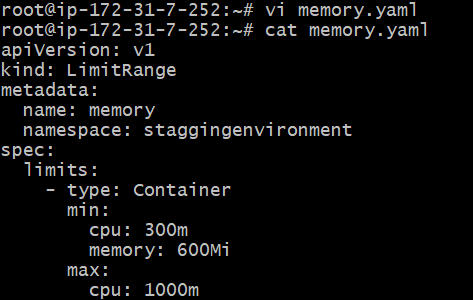
kubectl get pods -n prod-environment



1. In the staging-environment namespace, set a LimitRange that assigns default CPU and memory limits (300m CPU, 600Mi memory) and applies a minimum and maximum CPU.

Firstly create a yaml file using vi

Vi memory.yaml



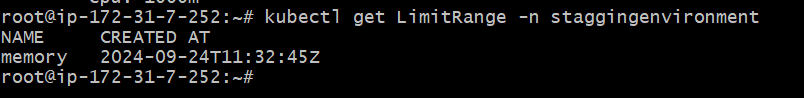
Apply the ns

Kubectl apply -f memory.yaml



To check the created limit memory range

Kubectl get LimitRange -n staggingenvironment



4) Create a pod and a NodePort service in the default namespace, then create another pod in the test namespace and communicate between them using Service DNS.

Create namespace

Kubectl create ns test

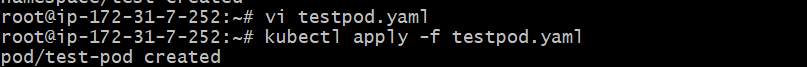


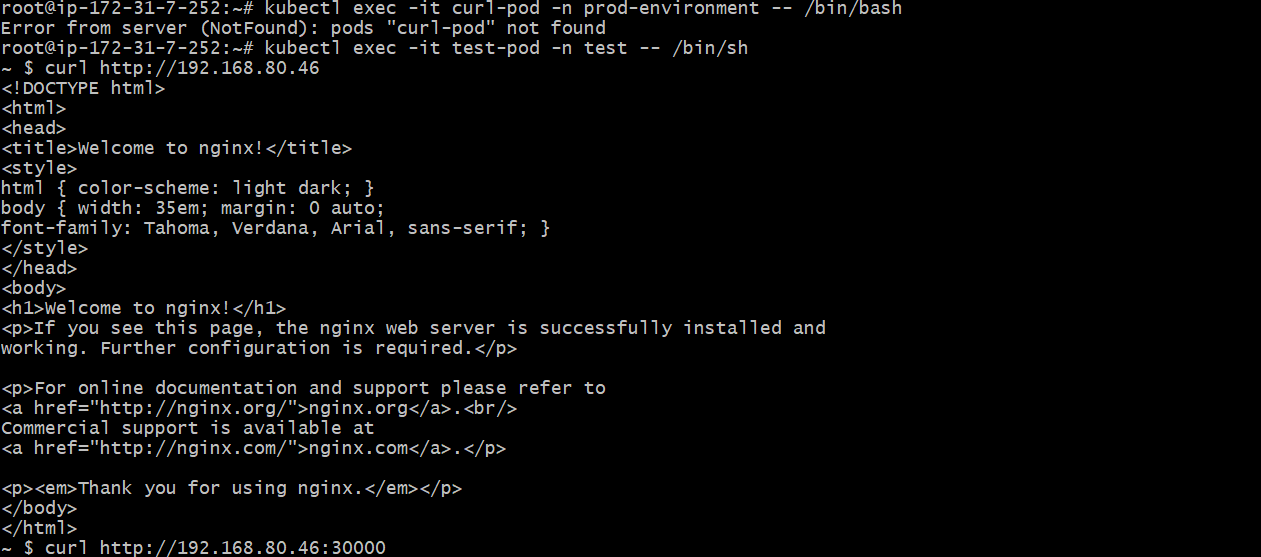
Create yaml file using vi

Vi test-pod.yaml

And apply the yaml file using

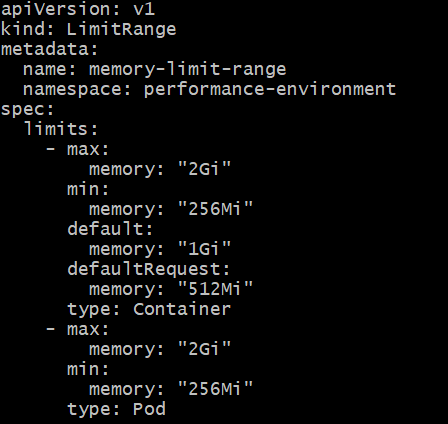
Kubectl apply -f test-pod.yaml





5) Apply a LimitRange with a max limit/request ratio of 2 for memory in the performance-environment namespace, and test by creating a pod with mismatched resource requests and limits.

Create name space and using yaml file apply the range with limits



Using apply command we can mismatched the range

kubectl apply -f range.yaml -n performane-environment

