**KUBERNETES**

It is container orchestration(management) tool.

alias kc=”kubectl”

**Minikubes:**

It provides the kubernetes cluster setup with one master and one worker with an abstraction layer for learning purpose.

Minimum requirement: 2CPU, 2GB RAM.

**Installation of minikubes:**

apt-get update

apt-get upgrade

apt-get install docker.io

<https://webme.ie/how-to-run-minikube-on-a-virtualbox-vm/>

CMD [“/bin/bash”, “-c”,”while true; do echo hello-world; sleep 5;done;”]

|  |  |
| --- | --- |
| **COMMANDS** | **DESCRIPTION** |
| kubectl get nodes | To get all nodes |
| which kubectl | Gives path of kubectl |
| kubectl describe node node-name | To get details of node |
| kubectl create -f pod.yml | To create manifest file(.yml file name) |
| kubectl apply -f pod.yml | To create and update manifest file |
| kubectl get pods | List of all pods |
| kubectl describe pod pod-name  **(or)**  kubectl describe pod/pod-name | To get details of pod |
| kubectl logs -f pod-name | To get logs of pod(if single container is present) |
| minikube start | To start minikube |
| kubectl logs -f pod-name -c container-name | To get logs of particular container in pod |
| kubectl delete pod pod-name  **(or)**  kubectl delete -f pod.yml | To delete pod |
| kubectl get pods -o wide pod-name | To get more details of pod (IP address, node) |
| kubectl exec -it pod-name -c container-name -- /bin*/*bash | To get into the container from particular pod |
| kubectl run pod-name --image-name=nginx --port=8080 --restart=never | Creating pod in externally(command-line) |
| kubectl get pods --show-labels | To see labels of all pods |
| kubectl get pods -l company=thbs | To get all pods with label=thbs (filtering) |
| kubectl get pods -l company=thbs, class=devops | To get all pods which holds all specified label values(AND operation) |
| kubectl get pods -l ‘class in (devops, kubernetes)’ | Returns all pods with label class=devops (or) class=kubernetes |
| kubectl get pods -l ‘class notin (devops, kubernetes)’ | Returns all pods with doesn't have label class=devops (or) class=kubernetes |
| kubectl label nodes node-name company=thbs | To give label to the node |
| kubectl scale --replicas=5 rc -l company=thbs | To scale-up/down the pods |
| kubectl delete pods --all  **(or)**  kubectl delete pod --all | To delete all pods |
| kubectl delete -f replication.yml  (or)  kubectl delete rc rc-name | To delete replication controller present in that file |
| kubectl describe rs rs-name | To get details of replica-set |
| kubectl delete rs rs-name  **(or)**  kubectl delete -f replicaset.yml | To delete replica-set |
| Kubectl scale --replicas=2 rs rs-name | To scale-up/down the pods from replica set controller |
| kubectl get rs --watch **(or)** kubectl get rs | To get replica-set list |
| kubectl describe rs my-rs | To get more details of replica-set |
| kubectl get pods pod-name -o yaml | To get yaml configuration of pod file |
| kubectl get deployment **(or)** kubectl get deployment --watch | To get list of deployment objects |
| kubectl get deployment deployment-name -o yaml | To get yaml configuration of deployment object |
| kubectl describe deployment deploy-name | To get more info of deployment object |
| kubectl scale deployment deploy-name --replicas=2 | To scale up/down pods from deployment controller |
| kubectl edit deployment deploy-name | Opens file to edit |
| kubectl set image deployment web-deploy con-name=image-name:1.7.9 --all=true | To set the image to specified version |
| kubectl rollout status deployment deploy-name | To show the status of rollback |
| kubectl rollout history deployment deploy-name | To see the history of all rollbacks |
| kubectl rollout undo deployment deploy-name | To undo previous rollback |
| kubectl rollout undo deployment deploy-name --to-revision=2 | To undo/delete particular rollback |
| Kubectl rollback pause deployment deploy-name | To pause the roll |

**Sample Manifest file: (with object as Pod)**

---

kind: Pod

apiVersion: v1

metadata:

 name: test-pod

labels:

  env: development

  name: test-pod

  class: kubernetes

  company: thbs

 annotations:

  description: This is a pod file with variable $(MYNAME)

spec:

 containers:

  - name: con1

    image: ubuntu

    command: ["/bin/bash", "-c", "while true; do echo hello!!!; sleep 5; done;"]

    env:

     - name: MYNAME

        value: THBS

  - name: con2

    image: ashwinih2611/thbs:ansible\_image

    command: ["/bin/bash", "-c", "while true; do echo hello!!!; sleep 5; done;"]

ports:

    containerPort: 80

This pod is applied to the nodes with label company=thbs

nodeSelector:

  company: thbs

  restartPolicy: Never

**Sample Manifest file: (with object as ReplicationController)**

---

kind: ReplicationController

apiVersion: v1

metadata:

 name: replica

spec:

 replicas: 2

 selector:

  company: thbs

 template:

  metadata:

   name: test

   labels:

    company: thbs

  spec:

   containers:

    - name: con1

      image: ubuntu

      command: ["/bin/bash", "-c", "while true; do echo hello-world; sleep 5; done;"]