Minikube Tutorials

1. Install a hypervisor (either virtual or kvm)
   1. For VirtualBox: https://websiteforstudents.com/installing-virtualbox-5-2-ubuntu-17-04-17-10/
2. Install Minikube and kubectl
   1. curl -Lo minikube https://storage.googleapis.com/minikube/releases/v0.25.2/minikube-linux-amd64 && chmod +x minikube && sudo mv minikube /usr/local/bin/
   2. curl -LO https://storage.googleapis.com/kubernetes-release/release/$(curl -s <https://storage.googleapis.com/kubernetes-release/release/stable.txt>)/bin/linux/amd64/kubectl
   3. chmod +x ./kubectl
   4. sudo mv ./kubectl /usr/local/bin/kubectl
3. Start Minikube
   1. minikube start
      1. In case minikube failed to start, run the following commands
         1. **Uninstall/delete all minikube related files**
         2. **minikube delete**
         3. **rm /usr/local/minikube**
         4. **rm -rf ~/.minikube**
4. Check the status of cluster using below command:

**Command: kubectl cluster-info**

**Output: root@aashish-ThinkPad-E450:/home/aashish/Desktop/Opcito/kubernetes# kubectl cluster-info**

**Kubernetes master is running at https://192.168.42.159:8443**

**To further debug and diagnose cluster problems, use 'kubectl cluster-info dump'.**

1. Check the status of nodes using below command:

**Command: kubectl get nodes**

**root@aashish-ThinkPad-E450:/home/aashish/Desktop/Opcito/kubernetes# kubectl get nodes**

**NAME STATUS ROLES AGE VERSION**

**minikube Ready <none> 1d v1.9.4**

1. Write **nginx-deployment.yaml** file for deployment

**nginx-deployment.yaml**

**apiVersion: apps/v1**

**kind: Deployment**

**metadata:**

**name: nginx-deployment #deployment name**

**labels:**

**app: nginx**

**spec:**

**replicas: 3 # creates three replicated Pods**

**selector: # it defines how the Deployment finds which Pods to manage. In this case, we simply select on one label defined in the Pod template**

**matchLabels:**

**app: nginx**

**template:**

**metadata:**

**labels:**

**app: nginx**

**spec: # indicates that the Pods run one container, nginx, which runs the nginx image 1.7.9**

**containers:**

**- name: nginx**

**image: nginx:1.7.9**

**ports:**

**- containerPort: 80 # deployment port 80**

1. Create deployment using the .yaml file, use the below command:

**Command: kubectl create -f nginx-deployment.yaml --record**

**Output:**

**root@aashish-ThinkPad-E450:/home/aashish/Desktop/Opcito/kubernetes# kubectl create -f nginx-deployment.yaml --record**

**deployment "nginx-deployment" created**

1. Use the below command to check deployment, pods and replica-sets:

**Command: kubectl get deployments**

**Outputs: root@aashish-ThinkPad-E450:/home/aashish/Desktop/Opcito/kubernetes# kubectl get deployments**

**NAME DESIRED CURRENT UP-TO-DATE AVAILABLE AGE**

**nginx-deployment 3 3 3 3 1m**

**Command: kubectl get pods**

**root@aashish-ThinkPad-E450:/home/aashish/Desktop/Opcito/kubernetes# kubectl get pods**

**NAME READY STATUS RESTARTS AGE**

**nginx-deployment-6c54bd5869-c2wq9 1/1 Running 0 3m**

**nginx-deployment-6c54bd5869-kmx58 1/1 Running 0 3m**

**nginx-deployment-6c54bd5869-kqbgr 1/1 Running 0 3m**

**Command: kubectl get pods --show-labels**

**Check the labels attached on each pod**

**Output:**

**root@aashish-ThinkPad-E450:/home/aashish/Desktop/Opcito/kubernetes# kubectl get pods --show-labels**

**NAME READY STATUS RESTARTS AGE LABELS**

**nginx-deployment-6c54bd5869-c2wq9 1/1 Running 0 7m app=nginx,pod-template-hash=2710681425**

**nginx-deployment-6c54bd5869-kmx58 1/1 Running 0 7m app=nginx,pod-template-hash=2710681425**

**nginx-deployment-6c54bd5869-kqbgr 1/1 Running 0 7m app=nginx,pod-template-hash=2710681425**

**Command: kubectl get rs**

**Check the replica sets**

**Output:**

**root@aashish-ThinkPad-E450:/home/aashish/Desktop/Opcito/kubernetes# kubectl get rs**

**NAME DESIRED CURRENT READY AGE**

**nginx-deployment-6c54bd5869 3 3 3 8m**

1. Create a new service and expose it to external traffic we’ll use the expose command with NodePort as parameter (minikube does not support the LoadBalancer option yet)

**Command: kubectl expose deployment nginx-deployment --type=NodePort**

**Output:root@aashish-ThinkPad-E450:/home/aashish/Desktop/Opcito/kubernetes# kubectl expose deployment nginx-deployment --type=NodePort**

**service "nginx-deployment" exposed**

**Check the services:**

**Command: kubectl get services**

**Output:root@aashish-ThinkPad-E450:/home/aashish/Desktop/Opcito/kubernetes# kubectl get services**

**NAME TYPE CLUSTER-IP EXTERNAL-IP PORT(S) AGE**

**kubernetes ClusterIP 10.96.0.1 <none> 443/TCP 3h**

**nginx-deployment NodePort 10.106.110.197 <none> 80:32189/TCP 10s**

**(Note: minikube does not provide external ip, you need to configure ingress for that, but you can still see the content in browser using minikube ip )**

**Command: minikube ip**

**Copy the minikube ip in browser, you see the nginx server on browser**

**192.168.42.159:32189**

**OR**

**Command: minikube service nginx-deployment**

**The above command will directly open the service in the browser.**

1. To view the minikube dashboard in browser

**minikube dashboard**

1. To delete all the service, deployment and pods, use the below command

**kubectl delete pod “pod\_name”**

**kubectl delete service “service\_name”**

**kubectl delete deployment “service\_name”**

Note: User Minikube Docker Daemon to check minikube docker image

eval $(minikube docker-env)

eval $(minikube docker-env -u)