**Create a Docker Image and Deploy It to Kubernetes**

First create a Docker file to build an image to create a runtime container. Then push it to Docker Hub for deploy/share to Kubernetes or any other environment. Container isolated and comprised with application and its libraries, dependencies. So, it works in the same way where ever it deploys.

**Step1:** Create dockerfile

**Vi Dockerfile**

|  |
| --- |
| FROM ubuntu:18.04    RUN apt-get update && \      apt-get install -y redis-server && \      apt install redis-tools && \      apt-get clean  EXPOSE 6379  CMD ["redis-server", "--protected-mode no"] |

With the base image Ubuntu, install Redis server and expose this container on port 6379.

**Step2:** Built image

|  |
| --- |
| $docker build -t redis-server . |

**Step3:** Run container.

|  |
| --- |
| $docker run -d -p 6379:6379 –name redis redis-server |

**Step4:** Verify

|  |
| --- |
| $redis-cli ping |

Returns : PONG

**Step5:** Docker login to push image to docker hub account

Create an docker hub account. And run following commands

|  |
| --- |
| $docker login  $ docker tag redis-server:latest ashwini22595/redis-repo:v1  $docker push ashwini22595/redis-repo:v1  Return success |

**Step6:** configure Kubernetes cluster.

Install kubeadm, kubectl, docker on Master Node and Worker Nodes

Then initialize Kubernetes cluster by running kubeadm init command on Master. And create an flannel/weave network on master. Then join command on worker nodes.

|  |
| --- |
| $sudo kubeadm init --pod-network-cidr=[10.244.0.0/16](http://10.244.0.0/16)  $kubectl apply -f <https://github.com/flannel-io/flannel/releases/latest/download/kube-flannel.yml> |

**Step6:** create vi pod.yaml

Configuration file to create pod, here we created an container with an redis image which is available in docker hub.

|  |
| --- |
| apiVersion: v1  kind: Pod  metadata:    name: redis    labels:      store: data  spec:    containers:    - name: mycontainer  #Pull image from docker hub account repository      image: ashwini22595/redis-repo:v1      ports:      - containerPort: 6379 |

**Step7: Verify**

$kubectl get pods

$kubectl describe pod redis

$kubectl exec -it redis bash

$redis-cli ping

Returns: PONG

**Screenshots:**

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated