Create a Docker Image and Deploy It to Kubernetes

DESCRIPTION

You are working as a DevOps engineer in an IT firm. You have been asked to create a Redis-based Docker image and deploy it on a Kubernetes cluster.

Background of the problem statement:

Your organization wants to use Redis in a Kubernetes cluster for data storage and caching purpose. The development team has asked you to create a Redis-based Docker image using a Dockerfile and deploy this image on a Kubernetes cluster.

You have also been asked to publish this image on your organization's Docker Hub account so that other team members can also access this image.

You must use the following:

- Docker CLI: To create the Docker image using a Dockerfile
- Docker Hub: To publish the image
- Kubectl: To deploy the image on a Kubernetes cluster

Following requirements should be met:

- Follow the above-mentioned specifications
- Make sure you create an account on Docker Hub to push the Docker image
- Document the step-by-step process involved in completing this task

Step 1. Setting Up the lab environment :

Commands:

1.sudo apt-get update

2. sudo apt-get upgrade -y

3. curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -

4. sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/ubuntu $(lsb_release -cs) stable"

#5.nstalling Docker Community edition

sudo apt-get install docker-ce -y

Step 2. Generating and Adding ssh rsa.pub keys in an authorized file

Commands:

1.ssh-keygen

2.cat .ssh/id_rsa.pub

3.touch .ssh/authorized_keys

4.sudo vi .ssh/authorized_keys

#master-172.31.31.232

ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAABAQC0WsaPaJNh4fQzCcA/anjo81lVxoADHqY6p1r72lF6Cj8L
EN5HwvHMH3yKo1WuCTn+glJUwrhdqkM5AXB7rP5nWaUlybt8X4hYTP8FTXbLeHFhl1KWvqJ487Ba
AghQubCKY3J9+btP6Utqp8Gf8Lh7Zw3wK7bAdnOHjf63HhvN/lGh3kKDCxhJVlAioBkGG5lKMpXJ9M
PNu5fYybl1sTbl7kHP32BcBWlbQXUki9JhlfMVr7eZcCZTKhPRRhokhYhSGteCiCwshXL+TPJnOZ91jV
lcFlfKE60m2V0lNz1ni+DFNO0tUPZV37/WXLDJTXqKU2lUPZWGz4cr7m1N7FbP
labsuser@ip-172-31-31-232

#worker1-172.31.22.12

ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAABAQC259f5oFtGvUUrmsVYjBjeJXmBi9daWQ5z9SYiWGq6H5
cFSmTHlKYX4jDByr8k/D1Q2Slv8w22Cdh1dsBxh3c4uEdZh6wn+heKuLokj2PKuiz39hPBwVA5FEYB8
7y5MYtkl01FN0ruAq6EJPehGG2kTfRgPk+0blqX0j2+Q7ExsyGQjAbjLSly/mhU0wY/6zYZuP2oLUQALu
15ULX6SUs3TCl35EPaszJSnj0QquVVKdT1m20sPEZiXksqfvxPl1gzRVLgA//AHJ9wE929VCbrLtMtaRV
LbcKyRlxqB/93z4F8PFvOUxekdRfLnca16CagkREnYvwfJ9APweJz6LqL labsuser@ip-172-31-22-12

#worker2-172.31.29.64

ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAABAQCzL+mBu42NSd6Nmp16WGiByu8sQ/Mi+Z2rLaeVtEbTxh
EG3Uq3jaSAlvwVHMZxd5xFi4XahJDimNTEZthmieJEZFnpRrLplWGOYQqwGVVadkCiA0EnlMSJvdiB
Woe4BMOjflr/Fwms8oiA7cVAOPw06MrKbz1StGvLJuqeo3jt4/ttSuSN7QZh85kA4wzeOLH1ACu1Bdv2
xVdurJXDCH9mkdop2R5TC8SUooOHG4ujZfOCSSwKYslRx73dVTjoULotSrlJTafbPWM60AlwS5b76n
Lgfms09vGSeSouGlOhE/xP5eslTjkQNllOEXCMJokGW6wSJFiZYrgShvp6yCRL
labsuser@ip-172-31-29-64

==========

Step 3. Setting up mirantis Kubernetes Environment -1

Commands :

1. **sudo wget https://github.com/Mirantis/launchpad/releases/latest/download/launchpad-linux-x64**
2. **mv launchpad-linux-x64 launchpad**
3. **sudo mv launchpad /usr/bin**
4. **sudo chmod 777 /usr/bin/launchpad**

#verifying Launchpad

5. **launchpad version**

Step 4. Setting up mirantis Kubernetes Environment-2

sudo vi launchpad.yml

```
---
apiVersion: launchpad.mirantis.com/mke/v1.1
kind: mke
metadata:
  name: my-mke
spec:
  hosts:
    -
      address: "172.31.29.64"
      privateInterface: eth0
      role: manager
      ssh:
        keyPath: /home/labsuser/.ssh/id_rsa
        user: labsuser
    -
      address: "172.31.22.12"
      privateInterface: eth0
      role: worker
      ssh:
        keyPath: /home/labsuser/.ssh/id_rsa
        user: labsuser
  mke:
    adminPassword: passw0rd!
    adminUsername: admin
    installFlags:
      - "--default-node-orchestrator=kubernetes"
      - "--pod-cidr 10.0.0.0/16"
      - "--force-minimums"
    version: "3.3.7"
```

launchpad register

```
> Name Tushar
> Email tusharkuchekar2014@gmail.com
> Company iamtushar
> I agree to Mirantis Launchpad Software Evaluation License Agreement
        https://github.com/Mirantis/launchpad/blob/master/LICENSE Yes
INFO Registration completed!


============================
```

launchpad apply

```
INFO Note: The configuration has been migrated from a previous version
INFO       to see the migrated configuration use: launchpad describe config


                ..,,,,,,..
          .:i1fCG0088@@@@@880GCLt;,              .,,::::::,,...
       ,;tC0@@@@@@@@@@@@@@@@@@@@@@@@@@@0:,     .,:ii111i;:,,..
     ,;1ttt1;;::::::;itfCG8@@@@@@@@@@i @@@@0fi1t111i;,.
   .,.               .:1L0@@   @8GCft111ii1;
                        :f0CLft1i;i1tL . @8Cti:.             .,:,.
                     .:;i1111i;itC;  @@@@@@@@@@@@80GCLftt11ttfLLLf1:.
                .,:;ii1111i:,.    , G8@@@@@@@@@@@@@@@@@@@@@@@@@0Lt;,
          ...,,::;;;;;::,.              ,;itfLCGGG0GGGCLft1;:.


   ;1:      i1, .1, .11111i:      .1i     :1;     ,1, i11111111: ;i   ;1111;
   G@GC:  1G0@i ;@1 ;@t:::;G0.   .0G8f    L@GC:   i@i :;;;@G;;;, C@ .80i:,:;
   C8 10CGC::@i :@i :@f:;;;CG.  .0G ,@L   f@.iGL, ;@;     @L     L@. tLft1;.
   G8   1;  ;@i ;@i :@L11C@t   ,08fffL@L  L@.  10fi@;    .@L     L@.    .:t@1
   C0       ;@i :@i :@i   ;Gf..0C     ,8L f@.   .f0@;    .8L     L8  fft11fG;
   ..       .   .   ..     ,..,        , ..      ..      ..     ..  .,:::,

   Mirantis Launchpad (c) 2021 Mirantis, Inc.                        v1.3.1

INFO ==> Running phase: Check For Upgrades
WARN a newer version of MKE is available: 3.5.1 (installing 3.3.7)
INFO ==> Running phase: Open Remote Connection
INFO [ssh] 172.31.29.64:22: testing connection
INFO [ssh] 172.31.22.12:22: testing connection
INFO ==> Running phase: Detect host operating systems
INFO [ssh] 172.31.29.64:22: is running Ubuntu 18.04.4 LTS
INFO [ssh] 172.31.22.12:22: is running Ubuntu 18.04.4 LTS
INFO ==> Running phase: Gather Facts
INFO [ssh] 172.31.29.64:22: gathering host facts
INFO [ssh] 172.31.22.12:22: gathering host facts
```

```
INFO [ssh] 172.31.29.64:22: mirantis container runtime not installed
INFO [ssh] 172.31.22.12:22: mirantis container runtime not installed
INFO [ssh] 172.31.29.64:22: using 172.31.29.64 as private IP
INFO [ssh] 172.31.29.64:22: is running "Ubuntu 18.04.4 LTS"
INFO [ssh] 172.31.29.64:22: internal address: 172.31.29.64
INFO [ssh] 172.31.29.64:22: gathered all facts
INFO [ssh] 172.31.22.12:22: using 172.31.22.12 as private IP
INFO [ssh] 172.31.22.12:22: is running "Ubuntu 18.04.4 LTS"
INFO [ssh] 172.31.22.12:22: internal address: 172.31.22.12
INFO [ssh] 172.31.22.12:22: gathered all facts
INFO ==> Running phase: Validate Facts
WARN [ssh] 172.31.29.64:22: added manager node's public address to mke installFlag SANs:
        --san=172.31.29.64
INFO ==> Running phase: Validate Hosts
INFO validating hostname uniqueness
INFO ==> Running phase: Download Mirantis Container Runtime installer
INFO downloading container runtime install script from https://get.mirantis.com/
INFO ==> Running phase: Prepare hosts
INFO [ssh] 172.31.29.64:22: installing base packages
INFO [ssh] 172.31.22.12:22: installing base packages
INFO [ssh] 172.31.22.12:22: base packages installed
INFO [ssh] 172.31.29.64:22: base packages installed
WARN [ssh] 172.31.22.12:22: added the current user to the 'docker' group
INFO [ssh] 172.31.22.12:22: reconnecting
INFO [ssh] 172.31.22.12:22: waiting for reconnection
WARN [ssh] 172.31.29.64:22: added the current user to the 'docker' group
INFO [ssh] 172.31.29.64:22: reconnecting
INFO [ssh] 172.31.29.64:22: waiting for reconnection
INFO ==> Running phase: Install Mirantis Container Runtime on the hosts
INFO [ssh] 172.31.29.64:22: installing container runtime (20.10.0)
INFO [ssh] 172.31.22.12:22: installing container runtime (20.10.0)
INFO [ssh] 172.31.29.64:22: running installer
INFO [ssh] 172.31.22.12:22: running installer
INFO [ssh] 172.31.22.12:22: mirantis container runtime version 20.10.0 installed
INFO [ssh] 172.31.29.64:22: mirantis container runtime version 20.10.0 installed
INFO ==> Running phase: Pull MKE images
INFO [ssh] 172.31.29.64:22: pulling image docker.io/mirantis/ucp:3.3.7
INFO [ssh] 172.31.29.64:22: pulling linux images
INFO [ssh] 172.31.29.64:22: pulling image mirantis/ucp-auth-store:3.3.7
INFO [ssh] 172.31.29.64:22: pulling image mirantis/ucp-calico-kube-controllers:3.3.7
INFO [ssh] 172.31.29.64:22: pulling image mirantis/ucp-calico-cni:3.3.7
INFO [ssh] 172.31.29.64:22: pulling image mirantis/ucp-agent:3.3.7
INFO [ssh] 172.31.29.64:22: pulling image mirantis/ucp-auth:3.3.7
INFO [ssh] 172.31.29.64:22: pulling image mirantis/ucp-azure-ip-allocator:3.3.7
INFO [ssh] 172.31.29.64:22: pulling image mirantis/ucp-calico-node:3.3.7
INFO [ssh] 172.31.29.64:22: pulling image mirantis/ucp-cfssl:3.3.7
INFO [ssh] 172.31.29.64:22: pulling image mirantis/ucp-compose:3.3.7
INFO [ssh] 172.31.29.64:22: pulling image mirantis/ucp-containerd-shim-process:3.3.7
INFO [ssh] 172.31.29.64:22: pulling image mirantis/ucp-controller:3.3.7
INFO [ssh] 172.31.29.64:22: pulling image mirantis/ucp-coredns:3.3.7
INFO [ssh] 172.31.29.64:22: pulling image mirantis/ucp-csi-attacher:3.3.7
INFO [ssh] 172.31.29.64:22: pulling image mirantis/ucp-csi-liveness-probe:3.3.7
INFO [ssh] 172.31.29.64:22: pulling image mirantis/ucp-csi-node-driver-registrar:3.3.7
```

INFO [ssh] 172.31.29.64:22: pulling image mirantis/ucp-csi-provisioner:3.3.7
INFO [ssh] 172.31.29.64:22: pulling image mirantis/ucp-csi-resizer:3.3.7
INFO [ssh] 172.31.29.64:22: pulling image mirantis/ucp-csi-snapshotter:3.3.7
INFO [ssh] 172.31.29.64:22: pulling image mirantis/ucp-dsinfo:3.3.7
INFO [ssh] 172.31.29.64:22: pulling image mirantis/ucp-etcd:3.3.7
INFO [ssh] 172.31.29.64:22: pulling image mirantis/ucp-hyperkube:3.3.7
INFO [ssh] 172.31.29.64:22: pulling image mirantis/ucp-interlock-config:3.3.7
INFO [ssh] 172.31.29.64:22: pulling image mirantis/ucp-interlock-extension:3.3.7
INFO [ssh] 172.31.29.64:22: pulling image mirantis/ucp-interlock-proxy:3.3.7
INFO [ssh] 172.31.29.64:22: pulling image mirantis/ucp-interlock:3.3.7
INFO [ssh] 172.31.29.64:22: pulling image mirantis/ucp-istio-mixer:3.3.7
INFO [ssh] 172.31.29.64:22: pulling image mirantis/ucp-istio-node-agent-k8s:3.3.7
INFO [ssh] 172.31.29.64:22: pulling image mirantis/ucp-istio-pilot:3.3.7
INFO [ssh] 172.31.29.64:22: pulling image mirantis/ucp-istio-proxyv2:3.3.7
INFO [ssh] 172.31.29.64:22: pulling image mirantis/ucp-kube-compose-api:3.3.7
INFO [ssh] 172.31.29.64:22: pulling image mirantis/ucp-kube-compose:3.3.7
INFO [ssh] 172.31.29.64:22: pulling image mirantis/ucp-kube-gmsa-webhook:3.3.7
INFO [ssh] 172.31.29.64:22: pulling image mirantis/ucp-metrics:3.3.7
INFO [ssh] 172.31.29.64:22: pulling image mirantis/ucp-nvidia-device-plugin:3.3.7
INFO [ssh] 172.31.29.64:22: pulling image mirantis/ucp-openstack-ccm:3.3.7
INFO [ssh] 172.31.29.64:22: pulling image mirantis/ucp-openstack-cinder-csi-plugin:3.3.7
INFO [ssh] 172.31.29.64:22: pulling image mirantis/ucp-pause:3.3.7
INFO [ssh] 172.31.29.64:22: pulling image mirantis/ucp-secureoverlay-agent:3.3.7
INFO [ssh] 172.31.29.64:22: pulling image mirantis/ucp-secureoverlay-mgr:3.3.7
INFO [ssh] 172.31.29.64:22: pulling image mirantis/ucp-swarm:3.3.7
INFO [ssh] 172.31.29.64:22: pulling image mirantis/ucp:3.3.7
INFO [ssh] 172.31.29.64:22: already exists: mirantis/ucp:3.3.7
INFO ==> Running phase: Initialize Swarm
INFO [ssh] 172.31.29.64:22: initializing swarm
INFO [ssh] 172.31.29.64:22: swarm initialized successfully
INFO ==> Running phase: Install MKE components
INFO [ssh] 172.31.29.64:22: time="2022-01-12T10:19:52Z" level=info msg="Your Docker daemon version 20.10.0, build 1e08f21 (5.3.0-1030-aws) is compatible with UCP 3.3.7 (0c9bfa3)"
INFO [ssh] 172.31.29.64:22: time="2022-01-12T10:19:52Z" level=warning msg="Your system does not have available disk space. UCP requires a minimum of 25.00 GB, but you only have 10.97 GB"
INFO [ssh] 172.31.29.64:22: time="2022-01-12T10:19:54Z" level=warning msg="None of the Subject Alternative Names we'll be using in the UCP certificates [\"172.31.29.64\" \"ip-172-31-29-64\"] contain a domain component. Your generated certs may fail TLS validation unless you only use one of these shortnames or IP addresses to connect. You can use the --san flag to add more aliases"
INFO [ssh] 172.31.29.64:22: time="2022-01-12T10:19:54Z" level=info msg="Checking required ports for connectivity"
INFO [ssh] 172.31.29.64:22: time="2022-01-12T10:20:03Z" level=info msg="Checking required container images"
INFO [ssh] 172.31.29.64:22: time="2022-01-12T10:20:03Z" level=warning msg="Possible conflict between Kubernetes service CIDR range 10.96.0.0/16 and default address pool for Swarm overlay networks 10.0.0.0/8"
INFO [ssh] 172.31.29.64:22: time="2022-01-12T10:20:03Z" level=info msg="disabling checks which rely on detecting which (if any) cloud provider the cluster is currently running on"
INFO [ssh] 172.31.29.64:22: time="2022-01-12T10:20:03Z" level=info msg="Running install agent container ..."

INFO [ssh] 172.31.29.64:22: time="2022-01-12T10:20:04Z" level=info msg="Loading install configuration"
INFO [ssh] 172.31.29.64:22: time="2022-01-12T10:20:04Z" level=info msg="Running Installation Steps"
INFO [ssh] 172.31.29.64:22: time="2022-01-12T10:20:04Z" level=info msg="Step 1 of 39: [Setup Internal Cluster CA]"
INFO [ssh] 172.31.29.64:22: time="2022-01-12T10:20:07Z" level=info msg="Step 2 of 39: [Setup Internal Client CA]"
INFO [ssh] 172.31.29.64:22: time="2022-01-12T10:20:07Z" level=info msg="Step 3 of 39: [Initialize etcd Cluster]"
INFO [ssh] 172.31.29.64:22: time="2022-01-12T10:20:13Z" level=info msg="Step 4 of 39: [Set Initial Config in etcd]"
INFO [ssh] 172.31.29.64:22: time="2022-01-12T10:20:13Z" level=info msg="Step 5 of 39: [Deploy RethinkDB Server]"
INFO [ssh] 172.31.29.64:22: time="2022-01-12T10:20:15Z" level=info msg="Step 6 of 39: [Initialize RethinkDB Tables]"
INFO [ssh] 172.31.29.64:22: time="2022-01-12T10:20:17Z" level=info msg="Step 7 of 39: [Create Auth Service Encryption Key Secret]"
INFO [ssh] 172.31.29.64:22: time="2022-01-12T10:20:17Z" level=info msg="Step 8 of 39: [Deploy Auth API Server]"
INFO [ssh] 172.31.29.64:22: time="2022-01-12T10:20:20Z" level=info msg="Step 9 of 39: [Setup Auth Configuration]"
INFO [ssh] 172.31.29.64:22: time="2022-01-12T10:20:21Z" level=info msg="Step 10 of 39: [Deploy Auth Worker Server]"
INFO [ssh] 172.31.29.64:22: time="2022-01-12T10:20:22Z" level=info msg="Step 11 of 39: [Deploy MKE Proxy Server]"
INFO [ssh] 172.31.29.64:22: time="2022-01-12T10:20:23Z" level=info msg="Step 12 of 39: [Initialize Swarm v1 Node Inventory]"
INFO [ssh] 172.31.29.64:22: time="2022-01-12T10:20:23Z" level=info msg="Step 13 of 39: [Deploy Swarm v1 Manager Server]"
INFO [ssh] 172.31.29.64:22: time="2022-01-12T10:20:24Z" level=info msg="Step 14 of 39: [Deploy Internal Cluster CA Server]"
INFO [ssh] 172.31.29.64:22: time="2022-01-12T10:20:25Z" level=info msg="Step 15 of 39: [Deploy Internal Client CA Server]"
INFO [ssh] 172.31.29.64:22: time="2022-01-12T10:20:26Z" level=info msg="Step 16 of 39: [Deploy MKE Controller Server]"
INFO [ssh] 172.31.29.64:22: time="2022-01-12T10:20:30Z" level=info msg="Step 17 of 39: [Deploy Kubernetes API Server]"
INFO [ssh] 172.31.29.64:22: time="2022-01-12T10:20:39Z" level=info msg="Step 18 of 39: [Deploy Kubernetes Controller Manager]"
INFO [ssh] 172.31.29.64:22: time="2022-01-12T10:20:43Z" level=info msg="Step 19 of 39: [Deploy Kubernetes Scheduler]"
INFO [ssh] 172.31.29.64:22: time="2022-01-12T10:20:47Z" level=info msg="Step 20 of 39: [Deploy Kubelet]"
INFO [ssh] 172.31.29.64:22: time="2022-01-12T10:21:26Z" level=info msg="Step 21 of 39: [Deploy Kubernetes Proxy]"
INFO [ssh] 172.31.29.64:22: time="2022-01-12T10:21:26Z" level=info msg="Step 22 of 39: [Wait for Healthy MKE Controller and Kubernetes API]"
INFO [ssh] 172.31.29.64:22: time="2022-01-12T10:21:27Z" level=info msg="Step 23 of 39: [Create Kubernetes Pod Security Policies]"
INFO [ssh] 172.31.29.64:22: time="2022-01-12T10:21:29Z" level=info msg="Step 24 of 39: [Install default storage class based on cloudprovider (for deprecated InTree providers)]"

```
INFO [ssh] 172.31.29.64:22: time="2022-01-12T10:21:29Z" level=info msg="Step 25 of 39:
      [Install Kubernetes CNI Plugin]"
INFO [ssh] 172.31.29.64:22: time="2022-01-12T10:21:49Z" level=info msg="Step 26 of 39:
      [Install CoreDNS]"
INFO [ssh] 172.31.29.64:22: time="2022-01-12T10:21:52Z" level=info msg="Step 27 of 39:
      [Install Cloud Controller Manager based on cloudprovider]"
INFO [ssh] 172.31.29.64:22: time="2022-01-12T10:21:52Z" level=info msg="Step 28 of 39:
      [Install Container Storage Interface Driver based on cloudprovider]"
INFO [ssh] 172.31.29.64:22: time="2022-01-12T10:21:52Z" level=info msg="Step 29 of 39:
      [Install Istio Ingress]"
INFO [ssh] 172.31.29.64:22: time="2022-01-12T10:22:08Z" level=info msg="Step 30 of 39:
      [Create MKE Controller Kubernetes Service Endpoints]"
INFO [ssh] 172.31.29.64:22: time="2022-01-12T10:22:10Z" level=info msg="Step 31 of 39:
      [Install Metrics Plugin]"
INFO [ssh] 172.31.29.64:22: time="2022-01-12T10:22:11Z" level=info msg="Step 32 of 39: [Install
      Kubernetes Compose Plugin]"
INFO [ssh] 172.31.29.64:22: time="2022-01-12T10:22:20Z" level=info msg="Step 33 of 39:
      [Deploy Manager Node Agent Service]"
INFO [ssh] 172.31.29.64:22: time="2022-01-12T10:22:20Z" level=info msg="Step 34 of 39:
      [Deploy Worker Node Agent Service]"
INFO [ssh] 172.31.29.64:22: time="2022-01-12T10:22:20Z" level=info msg="Step 35 of 39:
      [Deploy Windows Worker Node Agent Service]"
INFO [ssh] 172.31.29.64:22: time="2022-01-12T10:22:20Z" level=info msg="Step 36 of 39:
      [Deploy Cluster Agent Service]"
INFO [ssh] 172.31.29.64:22: time="2022-01-12T10:22:20Z" level=info msg="Step 37 of 39: [Set
      License]"
INFO [ssh] 172.31.29.64:22: time="2022-01-12T10:22:20Z" level=info msg="Step 38 of 39: [Set
      Registry CA Certificates]"
INFO [ssh] 172.31.29.64:22: time="2022-01-12T10:22:20Z" level=info msg="Step 39 of 39: [Wait
      for All Nodes to be Ready]"
INFO [ssh] 172.31.29.64:22: time="2022-01-12T10:22:25Z" level=info msg="All Installation Steps
      Completed"
INFO ==> Running phase: Upgrade MKE components
INFO [ssh] 172.31.29.64:22: cluster already at version 3.3.7, not running upgrade
INFO ==> Running phase: Join managers to swarm
INFO [ssh] 172.31.29.64:22: already a swarm node
INFO ==> Running phase: Join workers
INFO [ssh] 172.31.22.12:22: joined succesfully
INFO ==> Running phase: Validating MKE Health
INFO [ssh] 172.31.29.64:22: waiting for MKE to become healthy
INFO ==> Running phase: Label nodes
INFO [ssh] 172.31.29.64:22: labeling node
INFO [ssh] 172.31.22.12:22: labeling node
INFO ==> Running phase: Close Connection
INFO [ssh] 172.31.22.12:22: disconnected
INFO [ssh] 172.31.29.64:22: disconnected
INFO ==> Running phase: MKE cluster info
INFO Cluster is now configured.
INFO MKE cluster admin UI: https://172.31.29.64/
INFO You can download the admin client bundle with the command 'launchpad client-config'
================================================================================
      ====================
```

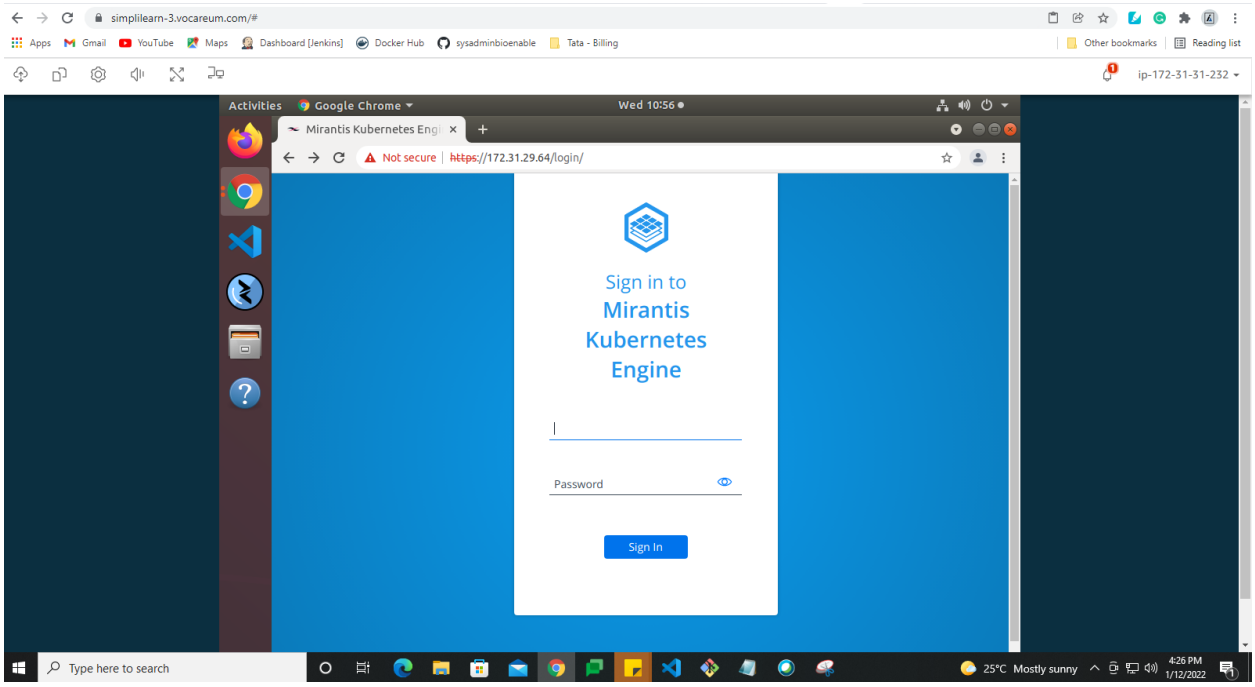
launchpad client-config

```
INFO Note: The configuration has been migrated from a previous version
INFO       to see the migrated configuration use: launchpad describe config
INFO ==> Running phase: Open Remote Connection
INFO [ssh] 172.31.29.64:22: testing connection
INFO ==> Running phase: Detect host operating systems
INFO [ssh] 172.31.29.64:22: is running Ubuntu 18.04.4 LTS
INFO ==> Running phase: Gather Facts
INFO [ssh] 172.31.29.64:22: gathering host facts
INFO [ssh] 172.31.29.64:22: is running mirantis container runtime version 20.10.0
INFO [ssh] 172.31.29.64:22: using 172.31.29.64 as private IP
INFO [ssh] 172.31.29.64:22: is running "Ubuntu 18.04.4 LTS"
INFO [ssh] 172.31.29.64:22: internal address: 172.31.29.64
INFO [ssh] 172.31.29.64:22: gathered all facts
INFO [ssh] 172.31.29.64:22: MKE has version 3.3.7
INFO ==> Running phase: Validate Hosts
INFO validating hostname uniqueness
INFO ==> Running phase: Download Client Bundle
INFO Successfully wrote client bundle to
         /home/labsuser/.mirantis-launchpad/cluster/my-mke/bundle/admin
INFO ==> Running phase: Close Connection
INFO [ssh] 172.31.29.64:22: disconnected
```

Now unzip ucp-bundle-admin

```
unzip ucp-bundle-admin.zip
cd ucp-bundle-admin
eval "$(<env.sh)"
Cluster "ucp_172.31.29.64:6443_admin" set.
User "ucp_172.31.29.64:6443_admin" set.
Context "ucp_172.31.29.64:6443_admin" created.
```

Snapshot :



Step 5: Docker CLI: To create the Redis Based Docker image using a Dockerfile

Commands :

1. mkdir redis

2. cd redis

3. touch Dockerfile

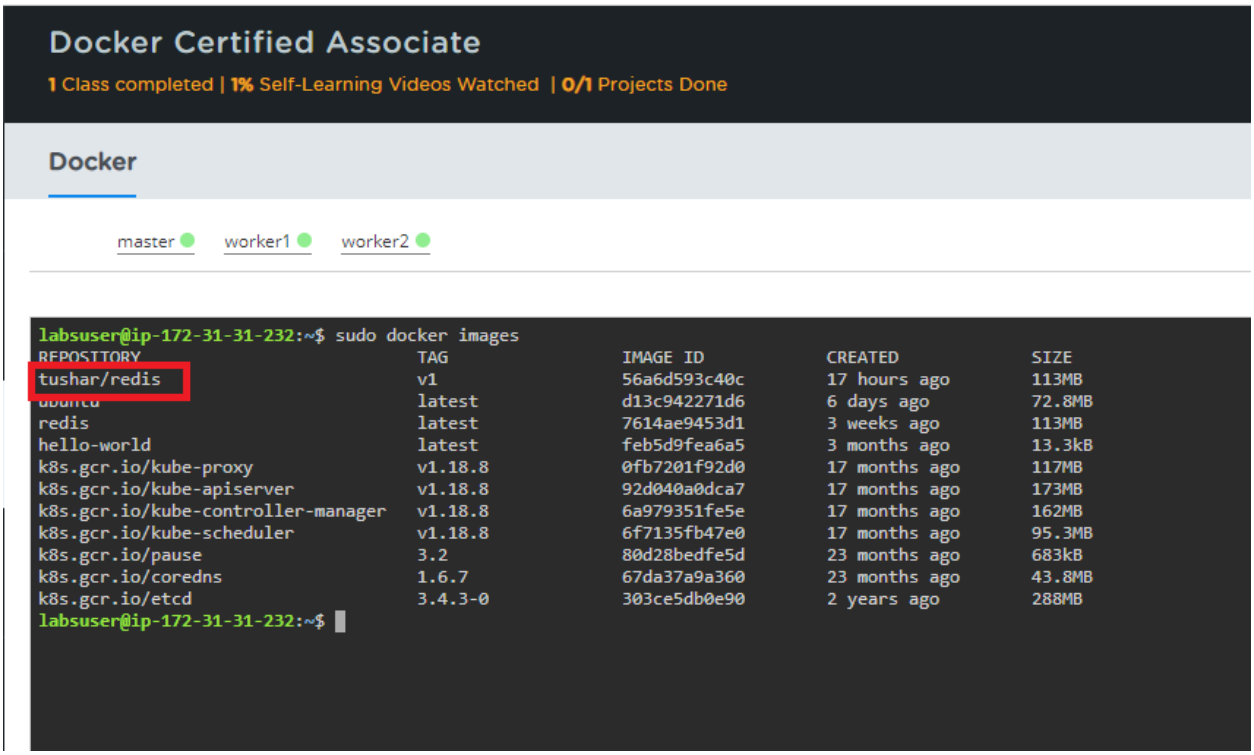
4 .vi Dockerfile

```
FROM redis

COPY redis.conf /usr/local/etc/redis/redis.conf

CMD [ "redis-server", "/usr/local/etc/redis/redis.conf" ]
```

#Build the redis Image

5. docker build -t tushar/redis:v1 .

```
Sending build context to Docker daemon  128.5kB
Step 1/4 : FROM redis:latest
latest: Pulling from library/redis
a2abf6c4d29d: Pull complete
c7a4e4382001: Pull complete
4044b9ba67c9: Pull complete
c8388a79482f: Pull complete
413c8bb60be2: Pull complete
1abfd3011519: Pull complete
Digest: sha256:db485f2e245b5b3329fdc7eff4eb00f913e09d8feb9ca720788059fdc2ed8339
Status: Downloaded newer image for redis:latest
 ---> 7614ae9453d1
Step 2/4 : COPY redis.conf /usr/local/etc/redis/redis.conf
 ---> 9d1c42397248
Step 3/4 : CMD [ "redis-server", "/usr/local/etc/redis/redis.conf" ]
 ---> Running in 30e39eafc908
Removing intermediate container 30e39eafc908
 ---> 2f450fd703aa
Step 4/4 : EXPOSE 6379
 ---> Running in 21f2ecf17168
Removing intermediate container 21f2ecf17168
 ---> 56a6d593c40c
Successfully built 56a6d593c40c
Successfully tagged tushar/redis:v1
```



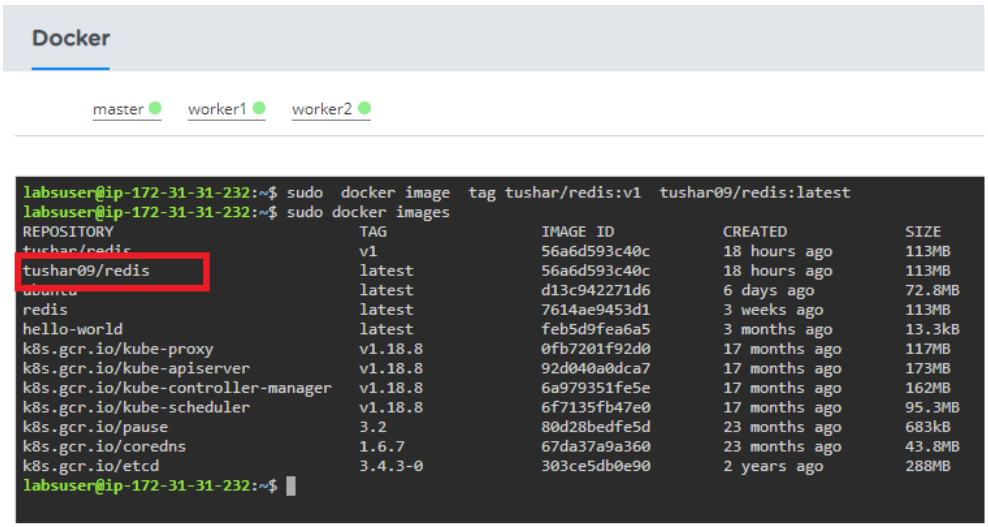
Step 6: Docker Hub: To publish the image

Create your Docker Hub account and publish the image in the docker hub

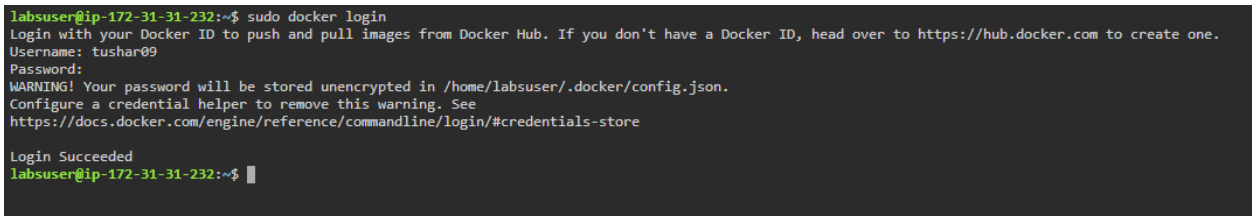
Commands:

Tag The image to push into DockerHub

sudo docker image tag tushar/redis:v1 tushar09/redis:latest



Docker Login:



Push The image into DockerHub :

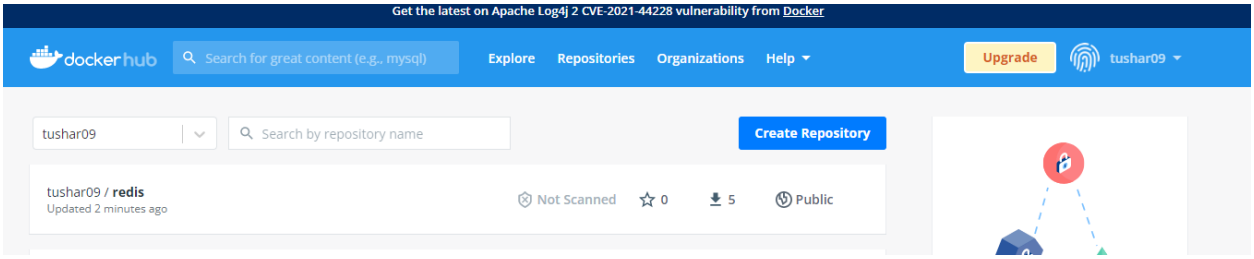
Command:

sudo docker image push tushar09/redis:latest

Snapshot:

```
labsuser@ip-172-31-31-232:~$ sudo docker image push tushar09/redis:latest
The push refers to repository [docker.io/tushar09/redis]
69bc31728c19: Pushed
8e5669d83291: Mounted from library/redis
9975392591f2: Mounted from library/redis
529cdb636f61: Mounted from library/redis
4b8e2801e0f9: Mounted from library/redis
9b24afeb7c2f: Mounted from library/redis
2edcec3590a4: Mounted from library/redis
latest: digest: sha256:19f4d2e6460c879334963391428ad5187527c082ea717a7e401f1d3b6c8defad size: 1782
labsuser@ip-172-31-31-232:~$
```

Image is successfully Published in Docker Hub



Step 7: Kubectl: To deploy the image on a Kubernetes cluster

Commands :

kubectl create deployment redis-image --image=tushar09/redis:latest
kubectl get pods
kubectl scale deployment redis-image --replicas=3
kubectl get pods -o wide
kubectl get service

```
labsuser@ip-172-31-31-232:~$ kubectl create deployment redis-image  --image=tushar09/redis:latest
deployment.apps/redis-image created
labsuser@ip-172-31-31-232:~$ kubectl get pods
NAME                           READY   STATUS    RESTARTS   AGE
nginx-68d7859cc6-9bqkx         1/1     Running   1          19h
redis-image-6cdf7977fc-zlfhk   1/1     Running   0          15s
labsuser@ip-172-31-31-232:~$
```

Scale the deployment in a cluster

```
labsuser@ip-172-31-31-232:~$ kubectl scale deployment redis-image --replicas=3
deployment.apps/redis-image scaled
labsuser@ip-172-31-31-232:~$ kubectl get pods
NAME                           READY   STATUS              RESTARTS   AGE
nginx-68d7859cc6-9bqkx         1/1     Running             1          19h
redis-image-6cdf7977fc-fvj29   1/1     Running             0          4s
redis-image-6cdf7977fc-xfqdz   0/1     ContainerCreating   0          4s
redis-image-6cdf7977fc-zlfhk   1/1     Running             0          115s
labsuser@ip-172-31-31-232:~$ kubectl get pods
NAME                           READY   STATUS    RESTARTS   AGE
nginx-68d7859cc6-9bqkx         1/1     Running   1          19h
redis-image-6cdf7977fc-fvj29   1/1     Running   0          10s
redis-image-6cdf7977fc-xfqdz   1/1     Running   0          10s
redis-image-6cdf7977fc-zlfhk   1/1     Running   0          2m1s
```

Pods and Service details

```
labsuser@ip-172-31-31-232:~$ kubectl get pods -o wide
NAME                           READY   STATUS    RESTARTS   AGE     IP            NODE              NOMINATED NODE   READINESS GATES
nginx-68d7859cc6-9bqkx         1/1     Running   1          19h     10.0.187.8    ip-172-31-22-12   <none>           <none>
redis-image-6cdf7977fc-fvj29   1/1     Running   0          3m57s   10.0.187.10   ip-172-31-22-12   <none>           <none>
redis-image-6cdf7977fc-xfqdz   1/1     Running   0          3m57s   10.0.187.11   ip-172-31-22-12   <none>           <none>
redis-image-6cdf7977fc-zlfhk   1/1     Running   0          5m48s   10.0.187.9    ip-172-31-22-12   <none>           <none>
labsuser@ip-172-31-31-232:~$ kubectl get service
NAME          TYPE        CLUSTER-IP     EXTERNAL-IP   PORT(S)    AGE
kubernetes    ClusterIP   10.96.0.1      <none>        443/TCP    20h
redis-image   ClusterIP   10.96.72.253   <none>        6379/TCP   18h
labsuser@ip-172-31-31-232:~$
```