**Steps to create docker image for twin4build layer component**

**Command to pull ubuntu image**

docker pull ubuntu

**Command to run ubuntu container**

docker run -dit ubuntu

**Copy source code into ubuntu container**

docker cp Twin4build\_code\_working.zip 16fef4d9b45f:/

**Created the Dockerfile with the required dependency**

****

**Source Code**

FROM ubuntu:latest

# set a directory for the app

WORKDIR /

# install dependencies

RUN apt-get update -y

RUN apt-get upgrade -y

RUN apt-get install unzip -y

RUN apt-get install python3.10 -y

RUN apt-get install pip -y

RUN apt-get install graphviz -y

RUN pip install psycopg2-binary

COPY . .

RUN unzip Twin4build\_code\_working.zip

WORKDIR /Twin4build\_code/Twin4Build

RUN pip install -r twin4build\_req.txt

# define the port number the container should expose

#EXPOSE 8005

# run the command

#CMD ["python3", "twin4build/api/codes/ml\_layer/simulator\_api.py"]

#CMD ["python3", "twin4build/api/codes/ml\_layer/request\_to\_api.py"]

COPY entrypoint.sh .

RUN chmod +x entrypoint.sh

CMD ["./entrypoint.sh"]

**Created the *“entrypoint.sh”* to run python scripts**

****

**Source Code**

#!/bin/bash

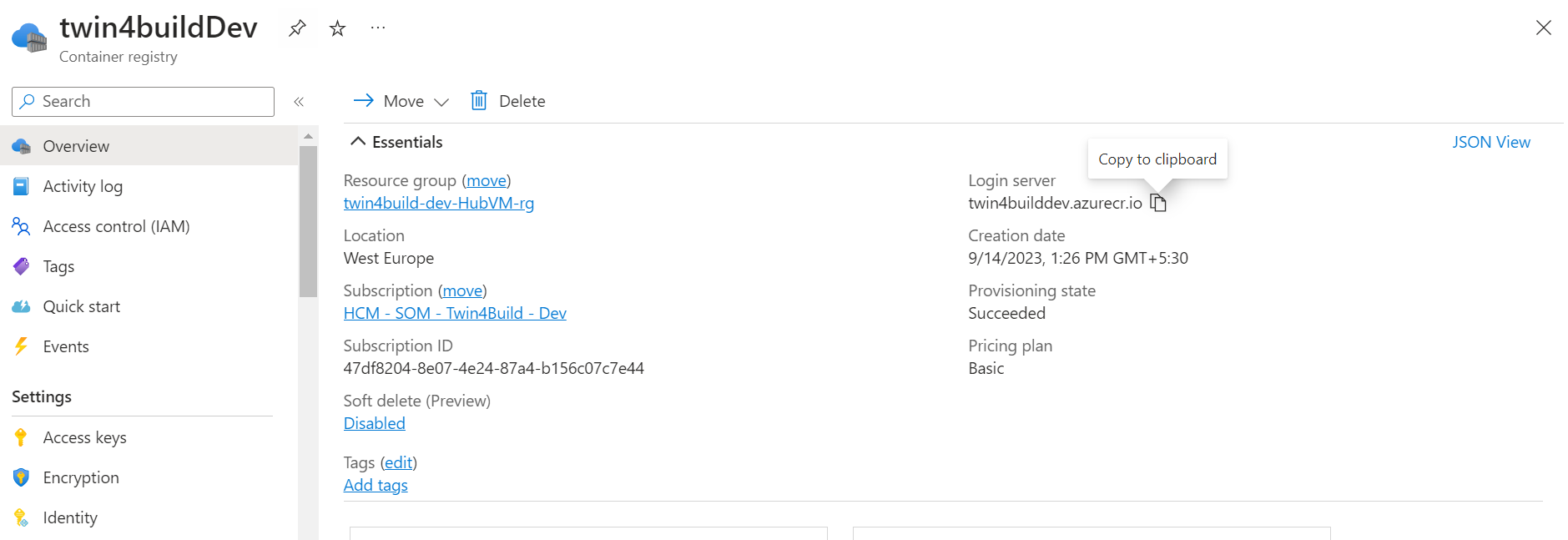
exec python3 twin4build/api/codes/ml\_layer/simulator\_api.py &

exec python3 twin4build/api/codes/ml\_layer/request\_to\_api.py

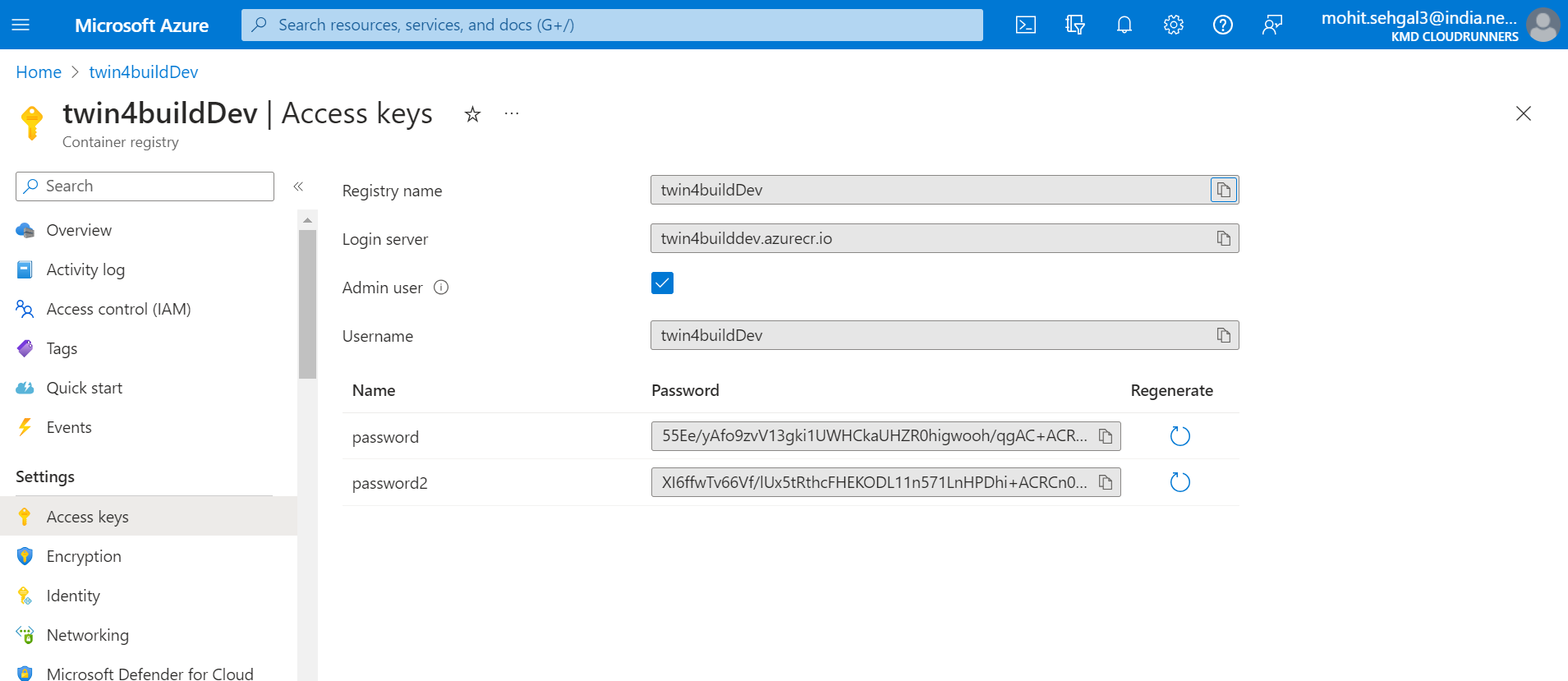
**Command to build docker image**

**[anurag.mishra1@nci-ip-d-kmd01 Working\_Code]$** docker build -t "twin4build:latest" .

**Create container register on Azure cloud console (To store build images)**



**Create access keys for registry “twin4buildDev”**



<https://learn.microsoft.com/en-us/azure/container-registry/container-registry-get-started-portal?tabs=azure-cli>

**Steps to install azure CLI**

**[anurag.mishra1@nci-ip-d-kmd01 Working\_Code]$** az login --tenant 1aaaea9d-df3e-4ce7-a55d-43de56e79442

To sign in, use a web browser to open the page https://microsoft.com/devicelogin and enter the code L49H6ABJ7 to authenticate.

[

{

"cloudName": "AzureCloud",

"id": "d24812e1-9f18-43fa-a52d-f21b33eb89b6",

"isDefault": true,

"name": "HCM - SOM - OS2IoT - Dev",

"state": "Enabled",

"tenantId": "1aaaea9d-df3e-4ce7-a55d-43de56e79442",

"user": {

"name": "mohit.sehgal3@india.nec.com",

"type": "user"

}

},

{

"cloudName": "AzureCloud",

"id": "47df8204-8e07-4e24-87a4-b156c07c7e44",

"isDefault": false,

**"name": "HCM - SOM - Twin4Build - Dev",**

"state": "Enabled",

"tenantId": "1aaaea9d-df3e-4ce7-a55d-43de56e79442",

"user": {

"name": "mohit.sehgal3@india.nec.com",

"type": "user"

}

},

{

"cloudName": "AzureCloud",

"id": "3db72739-f311-4c16-8c3a-c5aa204d0da3",

"isDefault": false,

"name": "HCM - SOM - DOLL - Prod",

"state": "Enabled",

"tenantId": "1aaaea9d-df3e-4ce7-a55d-43de56e79442",

"user": {

"name": "mohit.sehgal3@india.nec.com",

"type": "user"

}

},

{

"cloudName": "AzureCloud",

"id": "662fdea5-8111-4542-b0ae-ad0bb981351e",

"isDefault": false,

"name": "HCM - SOM - OS2IoT - Prod",

"state": "Enabled",

"tenantId": "1aaaea9d-df3e-4ce7-a55d-43de56e79442",

"user": {

"name": "mohit.sehgal3@india.nec.com",

"type": "user"

}

}

]

**[anurag.mishra1@nci-ip-d-kmd01 Working\_Code]$** az account set --subscription "HCM - SOM - Twin4Build - Dev"

**Login to created container registry named “twin4builddev”**

**[anurag.mishra1@nci-ip-d-kmd01 Working\_Code]$** az acr login --name twin4builddev

Login Succeeded

**Checked the docker images that we want to tag and push to above container registry**

**[anurag.mishra1@nci-ip-d-kmd01 Working\_Code]$** docker images

REPOSITORY TAG IMAGE ID CREATED SIZE

**twin4build latest 6e80eadd5700 2 hours ago 8.98GB**

**Command to tag image to ACR**

**[anurag.mishra1@nci-ip-d-kmd01 Working\_Code]$** docker tag twin4build twin4builddev.azurecr.io/twin4build:v1

**Command to verify tag image**

**[anurag.mishra1@nci-ip-d-kmd01 Working\_Code]$** docker images

REPOSITORY TAG IMAGE ID CREATED SIZE

twin4build latest 6e80eadd5700 2 hours ago 8.98GB

**twin4builddev.azurecr.io/twin4build v1 6e80eadd5700 2 hours ago 8.98GB**

**Command to push image to ACR**

**[anurag.mishra1@nci-ip-d-kmd01 Working\_Code]$** docker push twin4builddev.azurecr.io/twin4build:v1

The push refers to repository [twin4builddev.azurecr.io/twin4build]

d290ef21adf3: Pushed

36dad0ff8056: Pushed

819d5e1ad0ba: Pushed

5f70bf18a086: Pushed

5a6e7eb22ced: Pushed

5fe899201eb7: Pushed

141cc37a0ca7: Pushed

6e46651cfdde: Pushed

7fb1d74280c0: Pushed

0fe9cbc7b1a0: Pushed

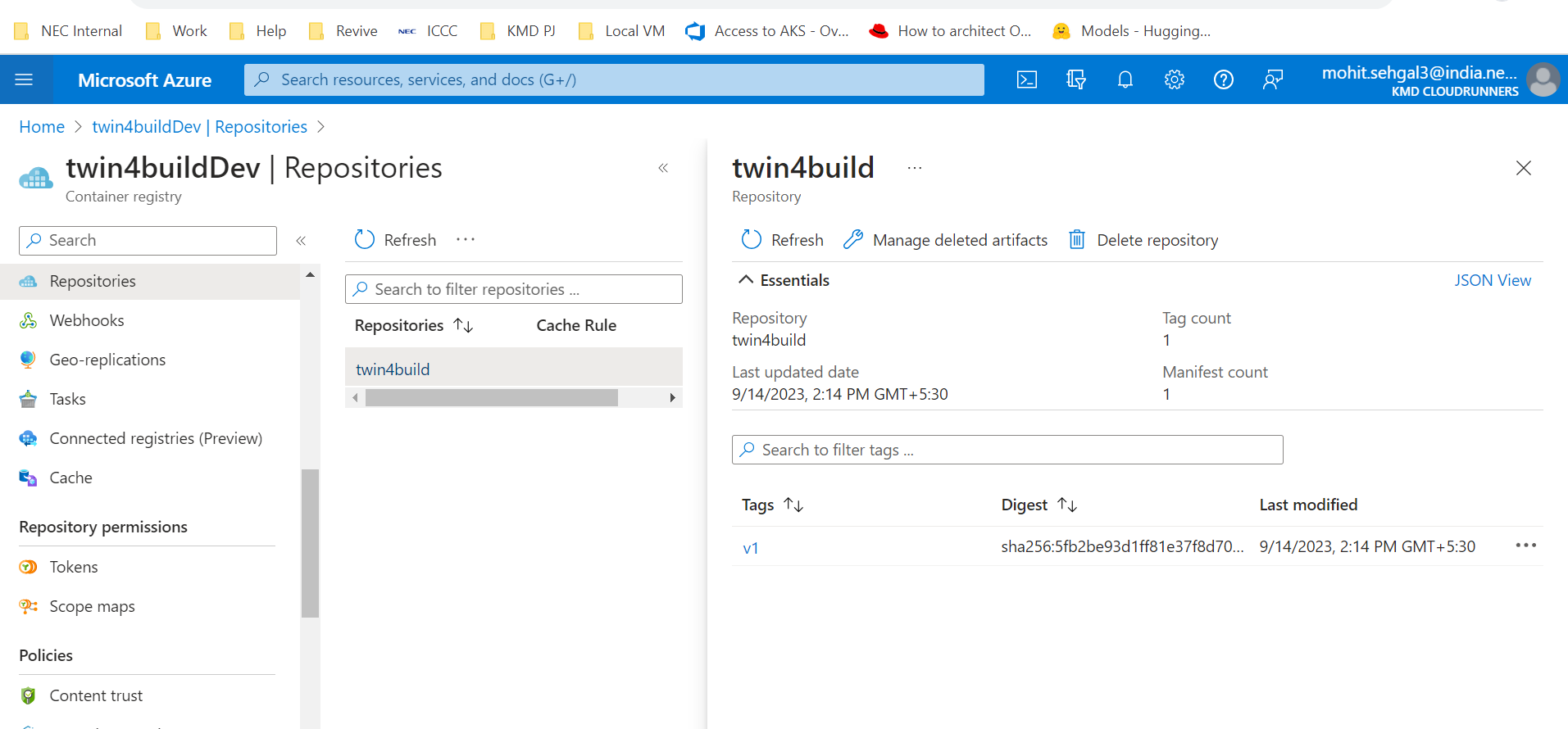
7687d1bd7b94: Pushed

d94ee032e905: Pushed

ba26f8986bc4: Pushed

dc0585a4b8b7: Pushed

v1: digest: sha256:5fb2be93d1ff81e37f8d702c0f449761eb8d13fcb0a3ed839fb6915ecc8f6202 size: 3270



**Command to save docker image (6e80eadd5700:image id)**

docker save 6e80eadd5700 > twin4build\_image.tar

**Note:** You can give any relevant name **twin4build\_image.tar,** Image id can vary **6e80eadd5700**

**Create directory on twin4build-dev environment AKS cluster**

mkdir twin4build\_aiml

**Execute the below command on twin4build-dev environment aks cluster**

kubectl create secret docker-registry twin4build-aiml-secret \  
    --namespace twin4vuild \  
    --docker-server=twin4builddev.azurecr.io \  
    --docker-username=twin4buildDev \  
    --docker-password=55Ee/yAfo9zvV13gki1UWHCkaUHZR0higwooh/qgAC+ACRBv5mkE

**Create a file name replicaset.yaml with the below details so we have deployed twin4build-sdu image as *“ReplicaSet”* type**

**Source Code**

apiVersion: apps/v1

kind: ReplicaSet

metadata:

name: twin4build-sdu

labels:

app: twin4build-sdu

namespace: twin4build

spec:

selector:

matchLabels:

app: twin4build-sdu

replicas: 1

template:

metadata:

name: twin4build-sdu

labels:

app: twin4build-sdu

namespace: twin4build

spec:

containers:

- name: twin4build-sdu

image: twin4builddev.azurecr.io/twin4build:v1

ports:

- containerPort: 8005

imagePullSecrets:

- name: twin4build-aiml-secret

**Check the file replicaset.yaml**

**mohit.sehgal3@india.nec.com@twin4build-dev-jumphost:~/twin4build\_aiml$ ll**

total 12

drwxrwxr-x 2 mohit.sehgal3@india.nec.com mohit.sehgal3@india.nec.com 4096 Sep 14 09:53 ./

drwxr-xr-x 6 mohit.sehgal3@india.nec.com mohit.sehgal3@india.nec.com 4096 Sep 14 08:23 ../

**-rw-r--r-- 1 root root 897 Sep 14 09:53 replicaset.yaml**

**Command to deploy replicaset.yaml(Final Step)**

kubectl create -f replicaset.yaml

You will see pod **“twin4build-sdu-rxgh9”** running succcesffully, which means dpeloyment is succceful and **twin4build-sdu** app service is up and runining as a **conatiner**.