Assessment

Name: Jansirani K

Emp id: 3025

DevOps OJT

**Task:**

Using ansible to deploy sample nginx/python application into kubernetes cluster.

- Build docker image for sample nginx/python and push to aws elastic container registry(ECR)      using docker cli.

- Create ansible role to deploy nginx application into kubernetes cluster.

- Create ansible playbooks to deploy nginx application.

- Container image should be pulled from aws elastic container registry(ECR).

- Outcome is an application deployed and run in k8s cluster. Able to access nginx website with url http://<ip\_address>:<port>.

- Keep the code stuff into the git repository.

- Document all the execution steps.

**Services used:**

* AWS instances
* AWS ECR
* Aws cli
* IAM
* Ansible & ansible roles
* K8s-minikube
* Docker
* Python
* Git

**1. Build docker image for sample nginx/python and push to aws elastic container registry(ECR)   using docker cli.**

Step1: Setup the docker environment, Install and configure the AWS CLI on our EC2 instance.

Step2: Build the nginx docker image with help of docker file and here I created a custom nginx [index.html].

**Dockerfile**

FROM nginx:1.10.1-alpine

COPY index.html /usr/share/nginx/html

EXPOSE 8080

CMD ["nginx", "-g", "daemon off;"]

**Index.html**

<!doctype html>

<html>

  <head>

    <title>Project!</title>  </head>

  <body>

    <h1>Hello Jansi welcome..!!! This is our Custom Nginx Home Page</h1>

    </body>

</html>

Step3: now push the docker image to aws elastic container registry(ECR)  using docker cli:

1. Retrieve an authentication token and authenticate your Docker client to your registry.  
   Use the AWS CLI:

|  |
| --- |
| aws ecr get-login-password --region us-east-1 | docker login --username AWS --password-stdin 327805045086.dkr.ecr.us-east-1.amazonaws.com |

1. Build your Docker image using the following command.

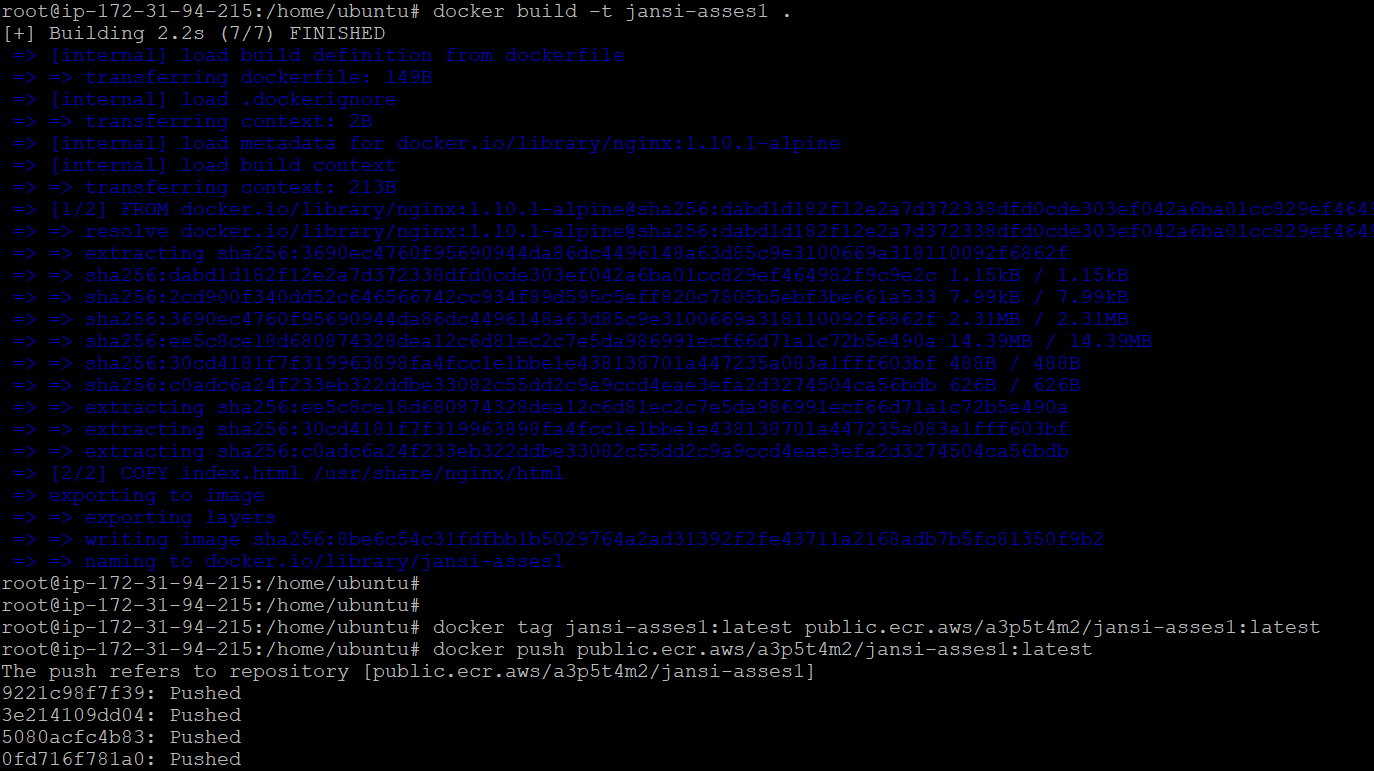
|  |
| --- |
| docker build -t jansi-asses . |

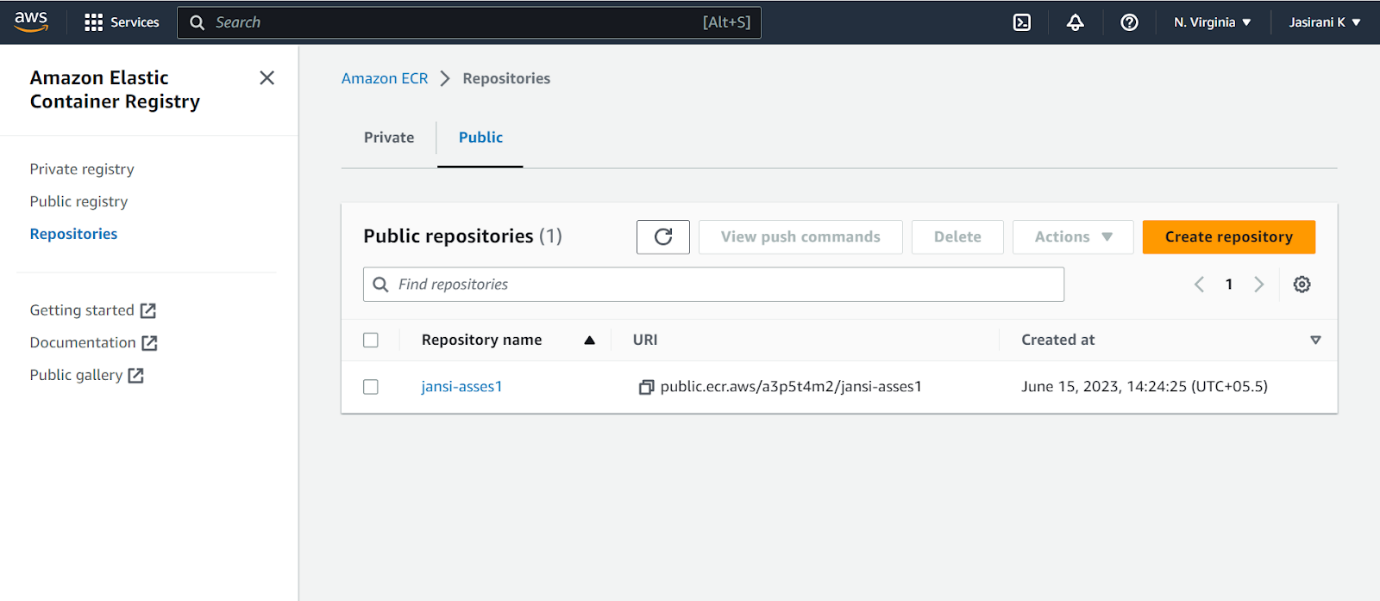
1. After the build completes, tag your image so you can push the image to this repository:

|  |
| --- |
| docker tag jansi-asses:latest 327805045086.dkr.ecr.us-east-1.amazonaws.com/jansi-asses:latest |

1. Run the following command to push this image to your newly created AWS repository:

|  |
| --- |
| docker push public.dkr.ecr.us-east-1.amazonaws.com/jansi-asses:latest |





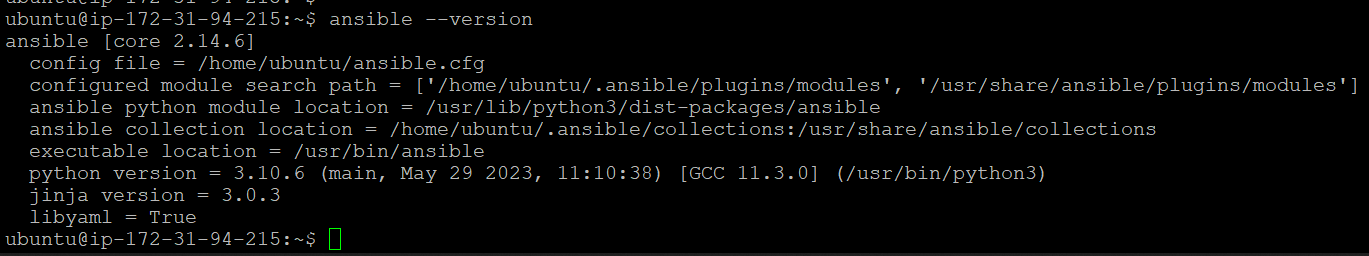
Step4: successfully push the image to AWS ECR using docker cli.

**2.  Create ansible role to deploy nginx application into kubernetes cluster**

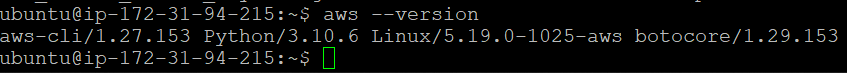
Step1: Install and setup the ansible and minikube in ec2 ubuntu instance.

Below screenshots are installed software’s on my instance

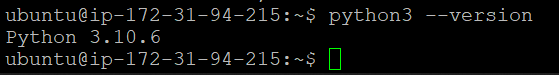
Ansible:



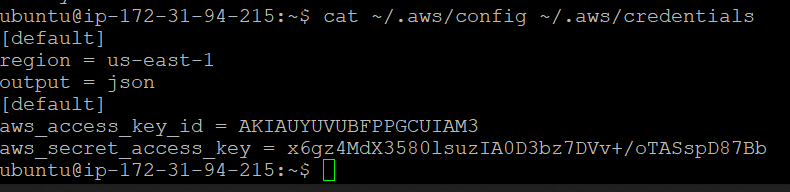
AWS CLI:



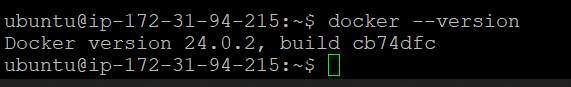
Python:



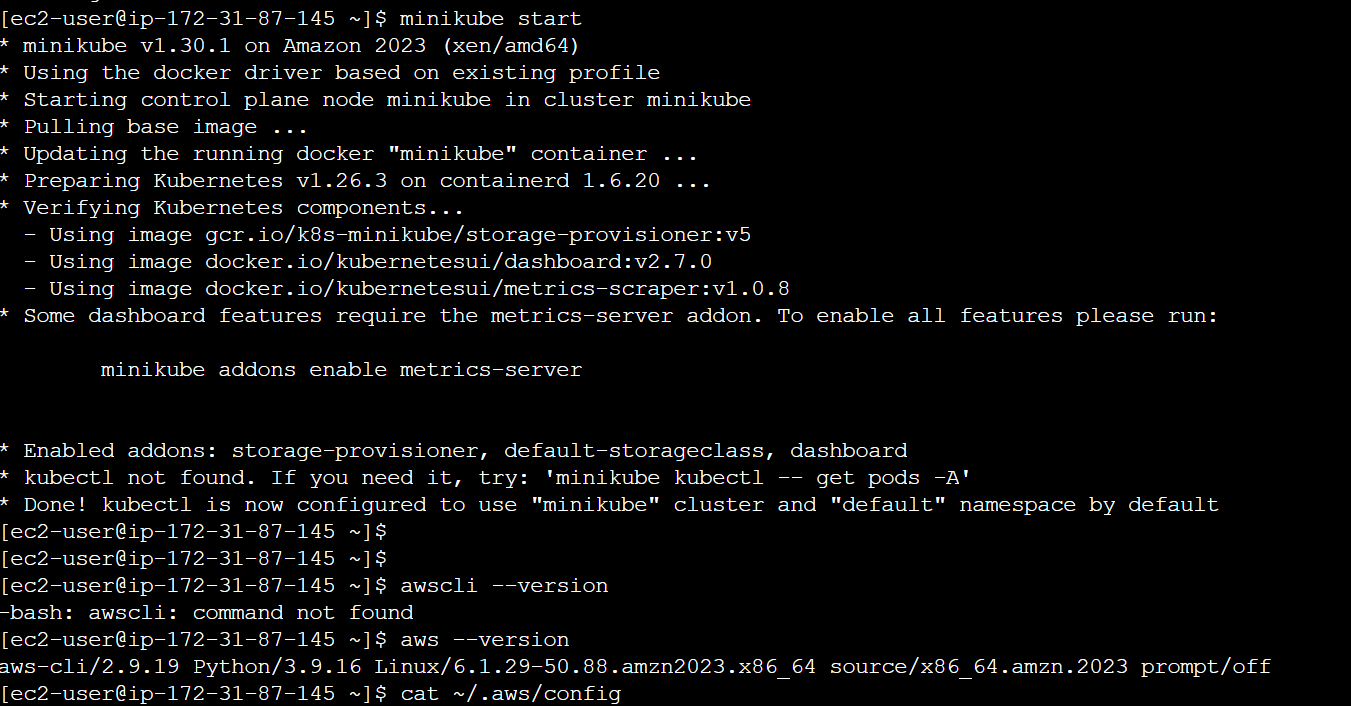
AWS Credential:



Docker:

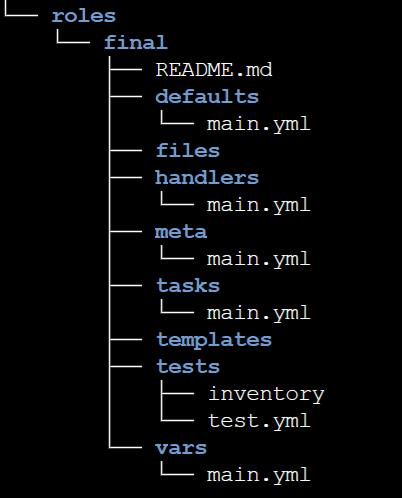


Minikube:



Step2: create ansible role here my role name is roles.

|  |
| --- |
| ansible-galaxy init roles |



Step3: Inside the ansible role directory add the file - tasks /main.yml.

**tasks /main.yml**

**---**

# tasks file for final

- name: Deploy NGINX Deployment

kubernetes.core.k8s:

state: present

definition:

apiVersion: apps/v1

kind: Deployment

metadata:

namespace: default

name: nginx-app

labels:

app: nginx-app

spec:

replicas: 2

selector:

matchLabels:

app: nginx-app

template:

metadata:

labels:

app: nginx-app

spec:

containers:

- name: nginx-app

image: public.ecr.aws/a3p5t4m2/jansi-asses1:latest

imagePullPolicy: Always

ports:

- containerPort: 80

imagePullSecrets:

- name: regcred

- name: Create NGINX Service

kubernetes.core.k8s:

state: present

definition:

apiVersion: v1

kind: Service

metadata:

namespace: default

name: nginx-app

labels:

name: nginx-app

spec:

selector:

app: nginx-app

type: NodePort

ports:

- protocol: TCP

port: 8080

nodePort: 30010

targetPort: 80

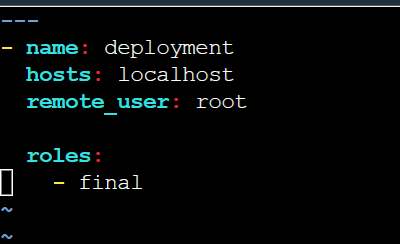




**3. Create ansible playbooks to deploy nginx application**

Step1**:** Creating ansible playbook for deploying the nginx.

playbook name- **playbook.yml**



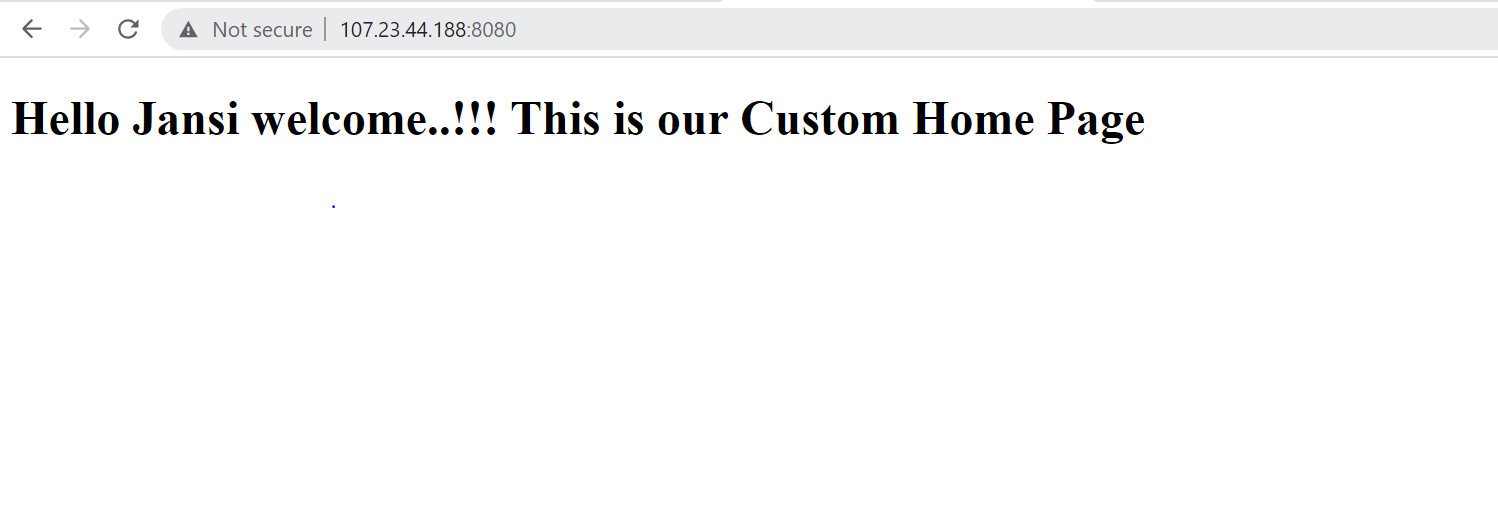
**4. Container image should be pulled from aws elastic container registry(ECR)**

In an Ansiblerole I have a tasks directory inside that has a main.yml file here. added the code for pulling the image from the AWS ECR and also given the credentials.

Finally run the ansible playbook

|  |
| --- |
| ansible-playbook playbook.yml |

**Output:**



**5. Keep the code stuff into the git repository**

https://github.com/johnkarthik142/Final-asses-ansible.git