Create 3 servers in AWS .

FOR - ANSIBLE , JENKINS – t2.micro – UBUNTU / LINUX .

MINICUBE single node server FOR WEBAPP – t2.medium – UBUNTU / LINUX

**Install Jenkins in Jenkins server .**

1 sudo yum upgrade -y

2 sudo apt-get update -y

4 java --version

6 sudo apt install openjdk-11-jre

7 java --version

8 sudo apt install wget

10 wget -q -O - https://pkg.jenkins.io/debian-stable/jenkins.io.key |sudo gpg --dearmor -o /usr/share/keyrings/jenkins.gpg

11 sudo sh -c 'echo deb [signed-by=/usr/share/keyrings/jenkins.gpg] http://pkg.jenkins.io/debian-stable binary/ > /etc/apt/sources.list.d/jenkins.list'

12 sudo apt update

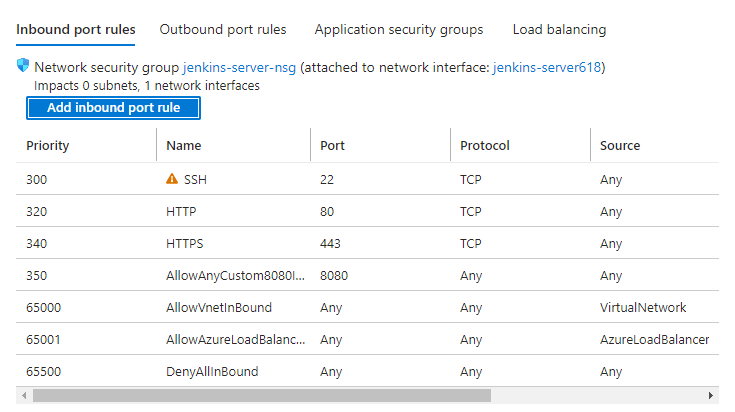
13 sudo apt install jenkins

14 sudo systemctl start jenkins.service

15 sudo systemctl status jenkins

16 telnet localhost 8080

Also add inbound rule for port 8080 in security group .



Jenkins will be accessible .

**INSTALL ansible in ansible server and docker also .**

1 sudo apt update && sudo apt upgrade -Y

2 sudo apt install ansible

3 ansible --version

4 sudo apt install docker.io

sudo usermod -a -G docker appadmin

sudo reboot now

**KUBERNETES CLUSTER WEBAPP:**

Install docker and minikube in server as kubernetes uses docker .

Installing docker :

3 sudo apt-get update && sudo apt-get upgrade

4 sudo apt install docker.io

6 sudo snap install docker

9 docker --version

10 systemctl start docker

11 docker ps -a

12 sudo systemctl status docker

13 ls -ltr

19 docker run hello-world

20 sudo usermod -a -G docker appadmin

22 systemctl reload

23 sudo reboot now

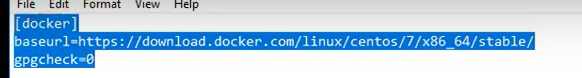
25 docker run hello-world

sudo docker images – list of images

sudo docker ps -a : list of containers

We can also add docker repo in default repo list of server to install docker .

Vi /etc/yum.repos.d/docker.repo



Yum repolist – to update repo list .

Yum install docker-ce – to install docker using repo we added in default repo list .

Installing minikube :

Minikube for Kubernetes Development.

What is minikube ? Minikube is a tool for running a Kubernetes cluster locally on your computer for development and testing purposes.It is designed to make it easy for developers to run and experiment with Kubernetes features without needing

access to a full-scale, production-grade Kubernetes cluster.

Minikube works by creating a singlenode Kubernetes cluster inside a virtual machine on your local machine.

It provides a simple commandline interface for starting and stopping the cluster, and for interacting with the

 Kubernetes API and running applications on the cluster.

With Minikube, developers can test their applications in a Kubernetes environment, experiment with Kubernetes

features like service discovery and load balancing, and gain experience with Kubernetes concepts and terminology. It is a popular tool among developers and DevOps engineers who work with Kubernetes

Install MINIKUBE in kubernetes server :

Download the latest Minikube binary with the command:

wget https://storage.googleapis.com/minikube/releases/latest/minikube-linux-amd64

Copy the file to the /usr/local/bin directory with the command:

sudo cp minikube-linux-amd64 /usr/local/bin/minikube

Give the Minikube executable the proper permissions with:

sudo chmod +x /usr/local/bin/minikube

Next, we need to install the kubectl command line utility. Download the binary executable file with:

curl -LO https://storage.googleapis.com/kubernetes-release/release/`curl -s [https://storage.googleapis.com/kubernetes-release/release/stable.txt`/bin/linux/amd64/kubectl](https://storage.googleapis.com/kubernetes-release/release/stable.txt%60/bin/linux/amd64/kubectl)

Give the new file the executable permission with:

chmod +x kubectl

Move the file into /usr/local/bin with the command:

sudo mv kubectl /usr/local/bin/

You can now start Minikube with the command:

minikube start --driver=docker

After the command completes, you can verify it’s running properly with the command:

minikube status

Now we can work with kubernetes as minikube and kubectl is installed .

**PLUGIN REQUIRED :** ssh agent

