

MINIKUBE INSTALLATION

Steps to install

Step 1:

1. Create one instance
 - Select ubuntu flavor
 - Select instance type as t2 medium
 - Finally launch the instances
2. Connect the instance to the server
3. Switch to root user by the command **sudo-i**
4. Update the server by the command **apt update -y**

Step 2:

1. Installation of **docker** by the commands

sudo apt install curl wget apt-transport-https -y
sudo curl -fsSL https://get.docker.com -o get-docker.sh
chmod 777 get-docker.sh
sh get-docker.sh

2. Perform these commands one by one in terminal to install the docker services

Step 3:

1. Installation of **minikube** by the commands

sudo curl -LO https://storage.googleapis.com/minikube/releases/latest/minikube-linux-amd64
sudo mv minikube-linux-amd64 /usr/local/bin/minikube

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

sudo minikube version

2. Perform these commands one by one in terminal to install the minikube services

Step 4:

1. Installation of **kubectl** by the commands

sudo curl -LO "https://dl.k8s.io/release/\$(curl -L -s https://dl.k8s.io/release/stable.txt)/bin/linux/amd64/kubectl"
--

sudo curl -LO "https://dl.k8s.io/\$(curl -L -s https://dl.k8s.io/release/stable.txt)/bin/linux/amd64/kubectl.sha256"

udo echo "\$(cat kubectl.sha256) kubectl" sha256sum --check
--

sudo install -o root -g root -m 0755 kubectl /usr/local/bin/kubectl
--

sudo install -o root -g root -m 0755 kubectl /usr/local/bin/kubectl
--

sudo kubectl version --client --output=yaml
--

sudo minikube start --driver=docker --force
--

2. Perform these commands one by one in terminal to install the kubectl services

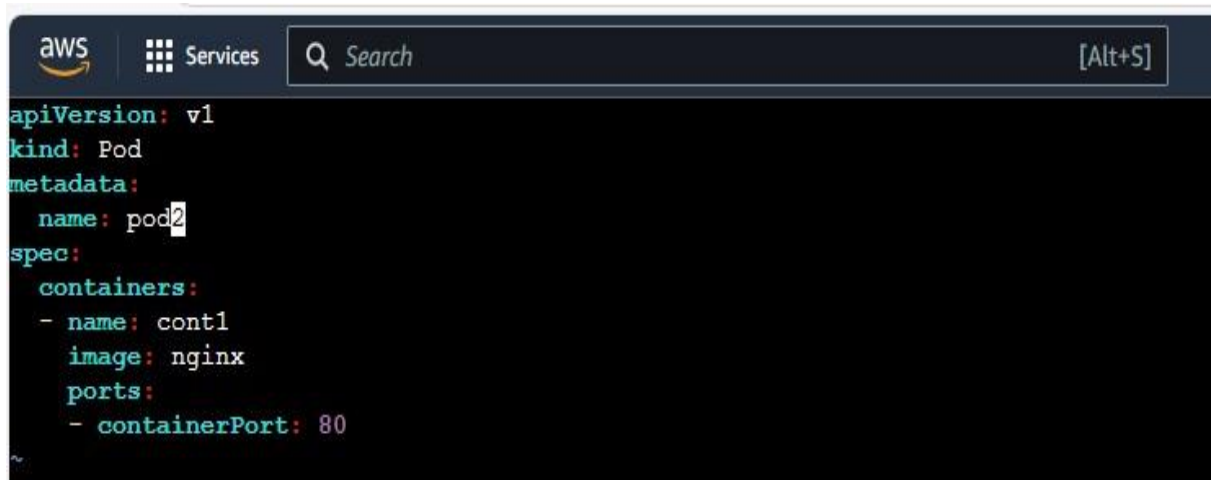
Step 5:

1. Create the pod by the command

kubectl run pod-name --image=image-name
--

2. Create the yaml file by the command **vi file-name.yaml**

3. Write the script in the yaml editor as shown in below figure



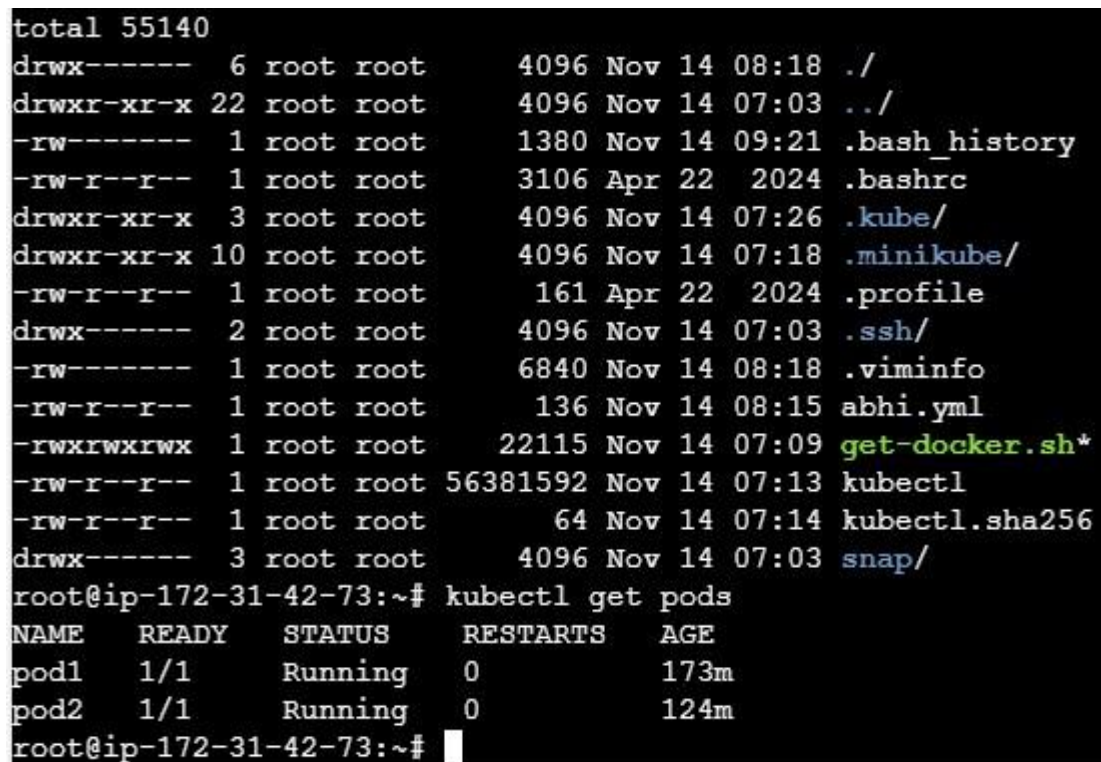
The screenshot shows the AWS IAM console with a search bar and a [Alt+S] button. Below the header, a YAML script is displayed in a dark-themed editor. The script defines a Pod with the following details:

```
apiVersion: v1
kind: Pod
metadata:
  name: pod2
spec:
  containers:
  - name: cont1
    image: nginx
    ports:
    - containerPort: 80
```

4. After writing the script in editor save the editor

5. **Kubectl create -f filename.yml** - by this command if pod is created then the work is done

6. **Kubectl get pods** – it is used to see the created pods as shown in below figure



The screenshot shows a terminal window with the following output:

```
total 55140
drwx----- 6 root root    4096 Nov 14 08:18 ./
drwxr-xr-x 22 root root    4096 Nov 14 07:03 ../
-rw----- 1 root root    1380 Nov 14 09:21 .bash_history
-rw-r--r-- 1 root root    3106 Apr 22 2024 .bashrc
drwxr-xr-x 3 root root    4096 Nov 14 07:26 .kube/
drwxr-xr-x 10 root root    4096 Nov 14 07:18 .minikube/
-rw-r--r-- 1 root root     161 Apr 22 2024 .profile
drwx----- 2 root root    4096 Nov 14 07:03 .ssh/
-rw----- 1 root root    6840 Nov 14 08:18 .viminfo
-rw-r--r-- 1 root root     136 Nov 14 08:15 abhi.yml
-rwxrwxrwx 1 root root   22115 Nov 14 07:09 get-docker.sh*
-rw-r--r-- 1 root root 56381592 Nov 14 07:13 kubectl
-rw-r--r-- 1 root root      64 Nov 14 07:14 kubectl.sha256
drwx----- 3 root root    4096 Nov 14 07:03 snap/

root@ip-172-31-42-73:~# kubectl get pods
NAME    READY   STATUS    RESTARTS   AGE
pod1    1/1     Running   0           173m
pod2    1/1     Running   0           124m
root@ip-172-31-42-73:~#
```