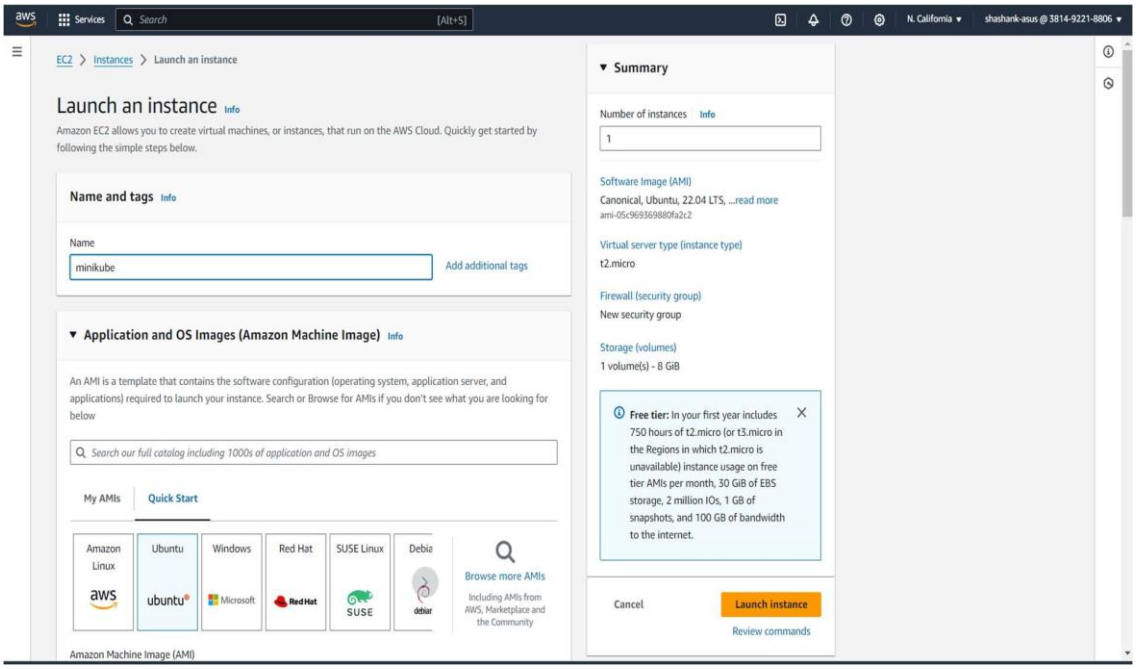
Task

Configure minikube on ubuntu

Name- tejal pawar

1. Create EC2 instance with ubuntu image use t2.medium.



2. Update your system & install docker in it.

sudo apt-get install docker-ce docker-ce-cli containerd.io docker-buildx-plugin docker-compose-plugin

3. # sudo apt update

4. Add your local user to docker group so that your local user run docker commands without sudo.

   # sudo usermod -aG docker $USER

   # newgrp docker

   # systemctl status docker

```
root@ip-172-31-5-167:~# sudo usermod -aG docker $USER
root@ip-172-31-5-167:~# newgrp docker
root@ip-172-31-5-167:~# systemctl status docker
● docker.service - Docker Application Container Engine
     Loaded: loaded (/lib/systemd/system/docker.service; enabled; vendor preset: enabled)
     Active: active (running) since Tue 2024-03-26 07:37:40 UTC; 1min 12s ago
TriggeredBy: ● docker.socket
       Docs: https://docs.docker.com
   Main PID: 3287 (dockerd)
      Tasks: 10
     Memory: 30.8M
        CPU: 340ms
     CGroup: /system.slice/docker.service
             └─3287 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerd.sock
```

5. Install minikube.
   # curl -LO
   https://storage.googleapis.com/minikube/releases/latest/minikube-linux-amd64
   # sudo install minikube-linux-amd64 /usr/local/bin/minikube

```
root@ip-172-31-5-167:~# curl -LO https://storage.googleapis.com/minikube/releases/latest/minikube-linux-amd64
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 89.3M  100 89.3M    0     0   101M      0 --:--:-- --:--:-- --:--:--  101M
root@ip-172-31-5-167:~# sudo install minikube-linux-amd64 /usr/local/bin/minikube
```

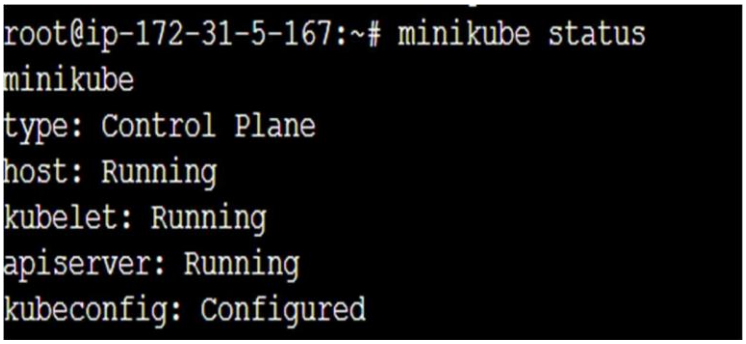
6. Check minikube version.
   # minikube version

```
root@ip-172-31-10-22:~# minikube version
minikube version: v1.32.0
commit: 8220a6eb95f0a4d75f7f2d7b14cef975f050512d
root@ip-172-31-10-22:~#
```

7. Start minikube using command. (use when you start or stop the system)
   # minikube start --driver=docker --force (with root user)
   # minikube start --driver=docker (without root user)

```
root@ip-172-31-5-167:~# minikube start --driver=docker --force
* minikube v1.32.0 on Ubuntu 22.04 (xen/amd64)
! minikube skips various validations when --force is supplied; this may lead to unexpected behavior
* Using the docker driver based on user configuration
* The "docker" driver should not be used with root privileges. If you wish to continue as root, use --force.
* If you are running minikube within a VM, consider using --driver=none:
*   https://minikube.sigs.k8s.io/docs/reference/drivers/none/
* Using Docker driver with root privileges
* Starting control plane node minikube in cluster minikube
* Pulling base image ...
* Downloading Kubernetes v1.28.3 preload ...
    > preloaded-images-k8s-v18-v1...:  403.35 MiB / 403.35 MiB  100.00% 83.40 M
    > gcr.io/k8s-minikube/kicbase...:  453.90 MiB / 453.90 MiB  100.00% 63.65 M
* Creating docker container (CPUs=2, Memory=2200MB) ...
* Preparing Kubernetes v1.28.3 on Docker 24.0.7 ...
  - Generating certificates and keys ...
  - Booting up control plane ...
  - Configuring RBAC rules ...
* Configuring bridge CNI (Container Networking Interface) ...
  - Using image gcr.io/k8s-minikube/storage-provisioner:v5
* Verifying Kubernetes components...
* Enabled addons: storage-provisioner, default-storageclass
* Done! kubectl is now configured to use "minikube" cluster and "default" namespace by default
root@ip-172-31-5-167:~# minikube status
minikube
type: Control Plane
host: Running
kubelet: Running
apiserver: Running
kubeconfig: Configured
```

8. Check minikube status.
   # minikube status



9. Install kubectl tool.
   (kubectl is a command line tool, used to interact with your Kubernetes cluster)
   # curl -LO https://storage.googleapis.com/kubernetes-release/release/`curl -s https://storage.googleapis.com/kubernetes-release/release/stable.txt`/bin/linux/amd64/kubectl

```
root@ip-172-31-5-167:~# curl -LO https://storage.googleapis.com/kubernetes-release/release/`curl -s https://storage.googleapis.com/kubernetes-release/release/stable.txt
`/bin/linux/amd64/kubectl
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 47.4M  100 47.4M    0     0  86.2M      0 --:--:-- --:--:-- --:--:-- 86.3M
```

10. Set the executable permission on it and move to /usr/local/bin and check kubectl version.

    # chmod +x kubectl

    # sudo mv kubectl /usr/local/bin

    # kubectl version -o yaml

```
root@ip-172-31-5-167:~# chmod +x kubectl
root@ip-172-31-5-167:~# sudo mv kubectl /usr/local/bin/
root@ip-172-31-5-167:~# kubectl version -o yaml
clientVersion:
  buildDate: "2024-03-15T00:08:19Z"
  compiler: gc
  gitCommit: 6813625b7cd706db5bc7388921be03071e1a492d
  gitTreeState: clean
  gitVersion: v1.29.3
  goVersion: go1.21.8
  major: "1"
  minor: "29"
  platform: linux/amd64
kustomizeVersion: v5.0.4-0.20230601165947-6ce0bf390ce3
```

11. To interact with your minikube cluster use commands.

    # kubectl get nodes

    # kubectl cluster-info

```
root@ip-172-31-5-167:~# kubectl get nodes
NAME       STATUS   ROLES           AGE   VERSION
minikube   Ready    control-plane   36s   v1.28.3
root@ip-172-31-5-167:~# kubectl cluster-info
Kubernetes control plane is running at https://192.168.49.2:8443
CoreDNS is running at https://192.168.49.2:8443/api/v1/namespaces/kube-system/services/kube-dns:dns/proxy
```