**Step 1**: Create one Master and One Worker Instance

copy key (.pem file) connect it to mobaXterm

**step 2**: install docker on both instances

do docker login to connect it with docker hub

1. set up the repo

> sudo apt-get update

> sudo apt-get install ca-certificates curl gnupg lsb-release

2. Add Docker’s official GPG key:

>curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o /usr/share/keyrings/docker-archive-keyring.gpg

> echo "deb [arch=$(dpkg --print-architecture) signed-by=/usr/share/keyrings/docker-archive-keyring.gpg] https://download.docker.com/linux/ubuntu $(lsb\_release -cs) stable" | sudo tee /etc/apt/sources.list.d/docker.list /dev/null

3. Install Docker

> sudo apt-get update

> sudo apt-get install docker-ce docker-ce-cli containerd.io

it will prompt do you want to continue type Y

Step 4: if docker installed chek versions

docker -v or docker --version

sudo docker info

**step 3**: set up hosts config file

>sudo vi /etc/hosts

>press i to insert

type privateIpOfMatser master

privateIpOfWorker worker1

esc

:wq!

enter

**step 4**: check file updated or not by executing ping command

ping master

ctrl + c (to stop)

ping worker1

ctrl + c (to stop)

**step** 5: init swarm

(if previously already initiated a swarm them first realese your node from that swarm by using below command >docker swarm leave --force)

> docker swarm init --advertise-addr 172.31.83.197 (Ip address of master node)

you will get one link that execute on worker node to join this swarm

(docker swarm join --token SWMTKN-1-692gbv2hdus4uu14cqt2ay8jm4ygs26yhw8c3a9oidxy402eib-0nq3zmrv5bdjnnx22sh7xgjjk 172.31.83.197:2377)

> docker swarm join --token (token number)

(In your machine you will get diffrent link)

>docker node ls (execute on master console)

**Step 6**: create service with 2 replicas

>docker service create --name cluserservice --publish 8085:8085 --replicas 2 vaishnavidonade/myimage

>docker service ps cluserservice

you can check 2 services are running

1 is on master node

1 is on worker node