



School of Computer Science, Engineering and Applications(SCSEA)
B.Tech FIY (CCSA)
Subject : Cloud Automation & Devops (P)

Name of the Student: Pratik.M.Rebari **PRN:** 20220802183

Title of Practicle : **4. Setup the Docker swarm installation (master slave architecture)**

1. Launch 4 instance with docker installation

The screenshot shows the AWS EC2 'Launch an instance' wizard. In the 'User data - optional' section, there is a large text area containing a shell script to install Docker. The script includes commands like `sudo apt-get update`, `sudo apt-get install`, and `curl` to download the Docker GPG key and repository. The 'Software Image (AMI)' is set to 'Canonical, Ubuntu, 24.04, amd64'. The 'Virtual server type (instance type)' is 't2.medium'. The 'Firewall (security group)' is 'SG_PRAFULL'. Under 'Storage (volumes)', it shows '1 volume(s) - 8 GB'. At the bottom right, there is a yellow 'Launch instance' button.

1. Connect all instance to terminal

The screenshot shows four separate terminal windows, each with a blue icon and the text 'ubuntu@ip-172-31-24-195:~'. The windows are arranged side-by-side, representing four different EC2 instances.

2. Now, Fire \$ sudo docker swarm init at 1st instance for give him leadership

```
ubuntu@ip-172-31-24-195:~$ sudo docker swarm init
Swarm initialized: current node (xr2iq9wci3jyg18orvt34m3nx) is now a manager.

To add a worker to this swarm, run the following command:

  docker swarm join --token SWMTKN-1-3qcxdb5nvejucsv2csp940en1hhc333w1xzyj35vu54qr0bf0k-7oubd4f7165zrwx417m28b3qj 172.31.24.195:2377

To add a manager to this swarm, run 'docker swarm join-token manager' and follow the instructions.
ubuntu@ip-172-31-24-195:~$
```



School of Computer Science, Engineering and Applications(SCSEA)
B.Tech FIY (CCSA)
Subject : Cloud Automation & Devops (P)

Name of the Student: Pratik.M.Rebari **PRN:** 20220802183

Title of Practicle : 4. Setup the Docker swarm installation (master slave architecture)

3. Now generate token for manager and worker with following commands

```
$ docker swarm join-token manager
```

```
$ docker swarm join-token worker
```

```
ubuntu@ip-172-31-24-195:~$ docker swarm join-token manager
To add a manager to this swarm, run the following command:
  docker swarm join --token SWMTKN-1-3qcxdb5nvejucsv2csp940en1hhc333w1xzyj35vu54qr0bf0k-5y9s
zo95v0z27sbircs2dpx4v 172.31.24.195:2377

ubuntu@ip-172-31-24-195:~$ docker swarm join-token worker
To add a worker to this swarm, run the following command:
  docker swarm join --token SWMTKN-1-3qcxdb5nvejucsv2csp940en1hhc333w1xzyj35vu54qr0bf0k-7oub
d4f7165zxrw417m28b3qj 172.31.24.195:2377

ubuntu@ip-172-31-24-195:~$ |
```

4. Now join 2nd instance as a manger with the help of token

Copy manager token and paste into 2nd instance

```
ubuntu@ip-172-31-18-27:~$ docker swarm join --token SWMTKN-1-3qcxdb5nvejucsv2csp940en1hhc333w1xzyj35vu54qr0bf0k-5y9sz095v0z27sbircs2dpx4v 172.31.24.195:2377
This node joined a swarm as a manager.
ubuntu@ip-172-31-18-27:~$ |
```

5. Join 3rd and 4th instance as a worker with the help of token

Copy worker token and paste into 3rd and 4th instance



School of Computer Science, Engineering and Applications(SCSEA)
B.Tech FIY (CCSA)
Subject : Cloud Automation & Devops (P)

Name of the Student: Pratik.M.Rebari **PRN:** 20220802183

Title of Practicle : **4. Setup the Docker swarm installation (master slave architecture)**

```
ubuntu@ip-172-31-25-4:~$ docker swarm join --token SWMTKN-1-3qcxdb5nvejucsv2csp940enlh
hc333wlxzyj35vu54qr0bf0k-7oubd4f7165zxrw417m28b3qj 172.31.24.195:2377
This node joined a swarm as a worker.
ubuntu@ip-172-31-25-4:~$ |
```

```
ubuntu@ip-172-31-20-92:~$ docker swarm join --token SWMTKN-1-3qcxdb5nvejucsv2csp940enl
hhc333wlxzyj35vu54qr0bf0k-7oubd4f7165zxrw417m28b3qj 172.31.24.195:2377
This node joined a swarm as a worker.
ubuntu@ip-172-31-20-92:~$ |
```

- Now check which node is worker and which node is manager with the help of following command . This command fire on manager node because only manager or leader can show this . -> \$ docker node ls

```
ubuntu@ip-172-31-24-195:~$ docker node ls
ID          HOSTNAME   STATUS  AVAILABILITY  MANAGER STATUS ENG
INE VERSION
z0fyvuh1hyt06dz676olyl2e8m  ip-172-31-18-27  Ready  Active        Reachable    29.
0.2
aodb8f1fbguax83e4xyxrrmq  ip-172-31-20-92  Ready  Active        Reachable    29.
0.2
xr2iq9wci3jygl8orvt34m3nx * ip-172-31-24-195  Ready  Active        Leader       29.
0.2
atiwsk98cxw05ydhitdxuac1s  ip-172-31-25-4   Ready  Active        Reachable    29.
0.2
ubuntu@ip-172-31-24-195:~$ |
```

- Now same command fire on worker node

Check worker has access or not

```
ubuntu@ip-172-31-25-4:~$ docker swarm join --token SWMTKN-1-3qcxdb5nvejucsv2csp940enlh
hc333wlxzyj35vu54qr0bf0k-7oubd4f7165zxrw417m28b3qj 172.31.24.195:2377
This node joined a swarm as a worker.
ubuntu@ip-172-31-25-4:~$ sudo docker node ls
Error response from daemon: This node is not a swarm manager. Worker nodes can't be used to view or modify cluster state. Please run this command on a manager node or promote the current node to a manager.
ubuntu@ip-172-31-25-4:~$ |
```