Docker Assignment-3

Multi-Node Docker Swarm Cluster for High Availability

Step 1: Initialize Docker Swarm, Check the cluster

Step 2: Create a Docker Compose File

```
master@master-vm:-$ docker swarm init --advertise-addr 192.168.219.137
Swarm initialized: current node (m5n7ln8kilmism69uo4mqvgzv) is now a manager.

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

docker swarm join --token SWMTKN-1-1xz5sjyweo4dinhrhwjfjulgxef83lx1277uyaa3ky3ejrnw3k-a9px9h85pid31d5957g7pmtj7 192.168.219.137:

2377

To add a manager to this swarm, run 'docker swarm join-token manager' and follow the instructions.

master@master-vm:-$ docker node ls

Do HOSTNAME STATUS AVAILABILITY MANAGER STATUS ENGINE VERSION

msnlnskilmism69uo4mqvgzv * master-vm Ready Active Leader 28.0.1

42vd78y6a4rh2ne7n2ie13yhd node1-vm Ready Active 26.1.3

pnp1a2fuitpikr5dkvu9au2y7 node2-vm Ready Active 26.1.3

master@master-vm:-$ ls

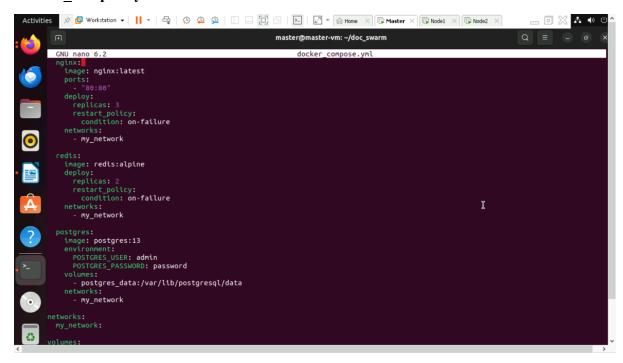
master@master-vm:-$ mkdir doc_swarm

mkdir: cannot create directory 'doc_swarm': File exists

master@master-vm:-$ mkdir doc_swarm

master@master-vm:-$ cidoc_swarm$ nano docker_compose.yml
```

Docker compose.yml file



Step 3: Deploy the Application on Swarm

```
master@master-vm:-/doc_swarn$ docker stack deploy -c docker_compose.yml myapp

$ince --detach=false was not specified, tasks will be created in the background.

In a future release, --detach=false will become the default.

Creating network myapp_my_network

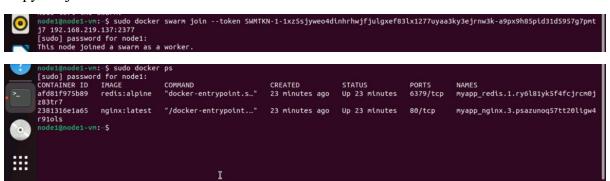
Creating service myapp_nginx

Creating service myapp_redis

Creating service myapp_postgres
```

Check service status

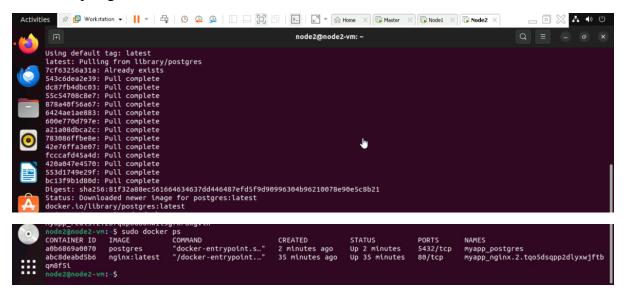
Copy the join token and add worker node-1



Copy the join token and add worker node-2

```
node2@node2-vm:-$ sudo docker swarm join --token SWMTKN-1-1xz5sjyweo4dinhrhwjfjulgxef83lx1277uyaa3ky3ejrnw3k-a9px9h85pid31d5957g7pmt j 192.168.219.137:2377 [sudo] password for node2: This node joined a swarm as a worker.
```

Pulled the postgres



See the list of services running in multiple nodes

