UNIVERSITY OF PETROLEUM & ENERGY STUDIES

Dehradun

APPLICATION CONTAINERIZATION

Name: Madhav Bhatia

Course: B. TECH CSE DevOps (2018-22)

Roll no.: R171218061

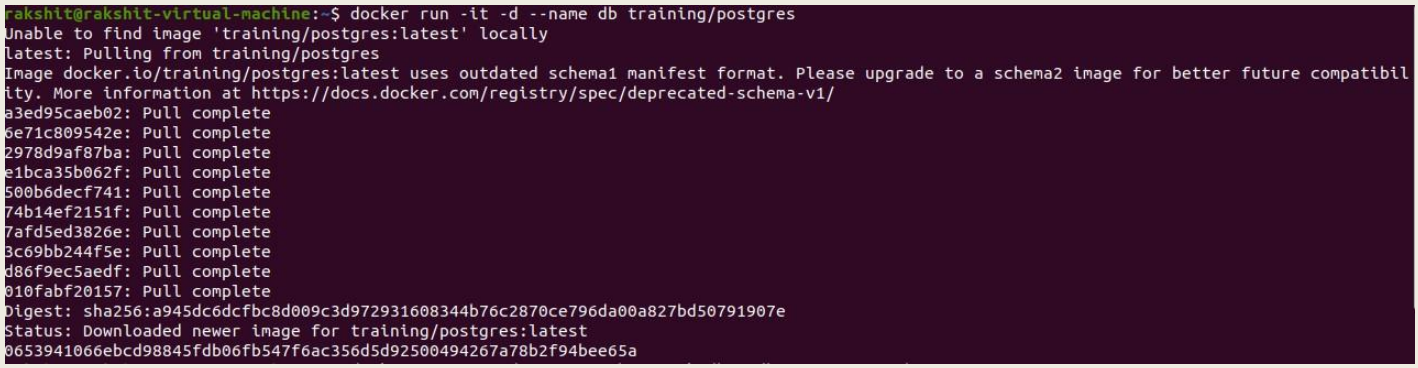
Sapid: 500068227

Experiment-6

Docker Linking

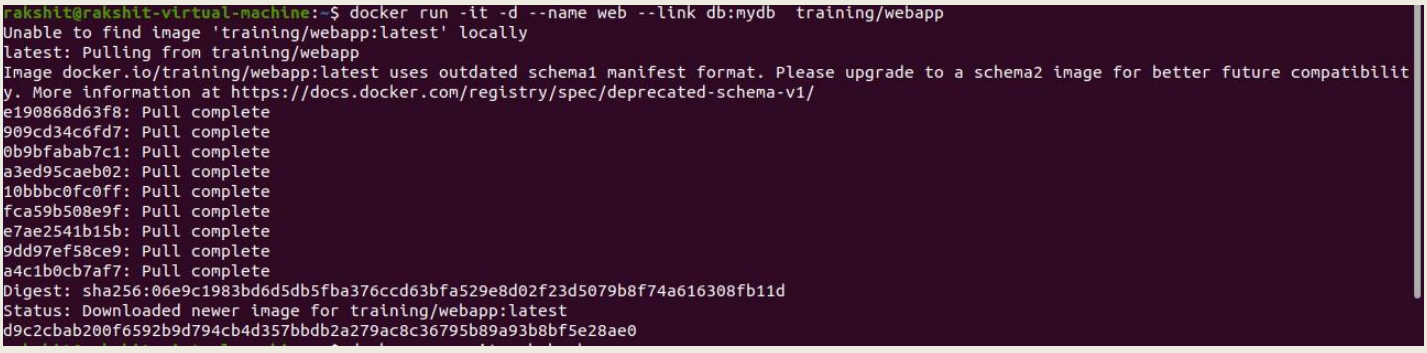
Step-1: Run a container in detached mode with name "db" from image "training/postgres"

```
$ docker run -it -d --name db training/postgres
```



Step-2: Run another container in detached mode with name "web" from image "training/webapp", link container "db" with alias "mydb" to this container and finally pass an inline command "python app.py" while running container.

```
$ docker run -it -d --name web --link db:mydb  training/webapp
```



Step-3: Take a bash terminal in "web" container and Test container linking by doing a ping to "mydb"

```
$ docker exec -it web bash
```

And then run

```
# ping mydb
```

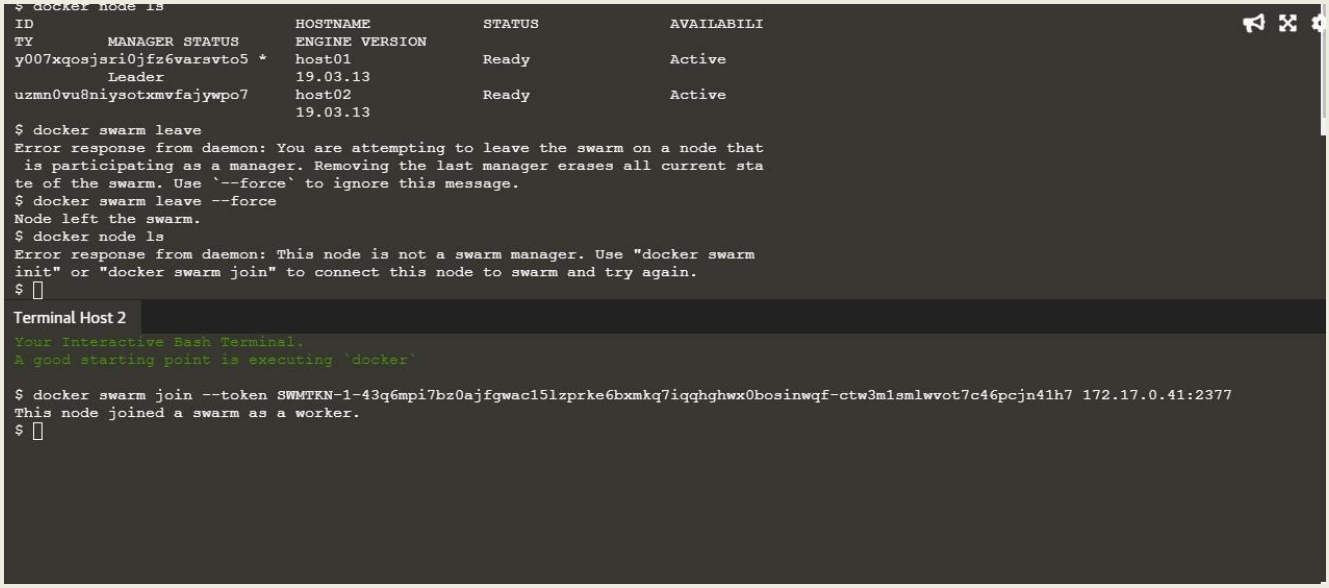
```
rakshit@rakshit-virtual-machine:~$ docker exec -it web bash
root@d9c2cbab200f:/opt/webapp# ping mycb
ping: unknown host mycb
root@d9c2cbab200f:/opt/webapp# ping mydb
PING mydb (172.17.0.2) 56(84) bytes of data.
64 bytes from mydb (172.17.0.2): icmp_seq=1 ttl=64 time=1.32 ms
64 bytes from mydb (172.17.0.2): icmp_seq=2 ttl=64 time=0.195 ms
64 bytes from mydb (172.17.0.2): icmp_seq=3 ttl=64 time=0.088 ms
64 bytes from mydb (172.17.0.2): icmp_seq=4 ttl=64 time=0.071 ms
64 bytes from mydb (172.17.0.2): icmp_seq=5 ttl=64 time=0.096 ms
64 bytes from mydb (172.17.0.2): icmp_seq=6 ttl=64 time=0.079 ms
64 bytes from mydb (172.17.0.2): icmp_seq=7 ttl=64 time=0.097 ms
64 bytes from mydb (172.17.0.2): icmp_seq=8 ttl=64 time=0.260 ms
64 bytes from mydb (172.17.0.2): icmp_seq=9 ttl=64 time=0.090 ms
64 bytes from mydb (172.17.0.2): icmp_seq=10 ttl=64 time=0.096 ms
64 bytes from mydb (172.17.0.2): icmp_seq=11 ttl=64 time=0.089 ms
^C
--- mydb ping statistics ---
11 packets transmitted, 11 received, 0% packet loss, time 10318ms
rtt min/avg/max/mdev = 0.071/0.226/1.327/0.352 ms
root@d9c2cbab200f:/opt/webapp#
```

Docker Swarm

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

    docker swarm join --token SWMTKN-1-43q6mpi7bz0ajfgwac15lzprke6bxmkq7iqqhghwx
0bosinwqf-ctw3m1smlwvot7c46pcjn41h7 172.17.0.41:2377

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

$ docker node ls
ID                            HOSTNAME            STATUS              AVAILABILI
TY        MANAGER STATUS      ENGINE VERSION
y007xqosjsri0jfz6varsvto5 *   host01              Ready               Active
          Leader              19.03.13
uzmn0vu8niysotxmvfajywpo7     host02              Ready               Active
                              19.03.13
$
```

Terminal Host 2

```
Your Interactive Bash Terminal.
A good starting point is executing `docker`

$ docker swarm join --token SWMTKN-1-43q6mpi7bz0ajfgwac15lzprke6bxmkq7iqqhghwx0bosinwqf-ctw3m1smlwvot7c46pcjn41h7 172.17.0.41:2377
This node joined a swarm as a worker.
$
```


$ docker node ls
ID HOSTNAME STATUS AVAILABILI
TY MANAGER STATUS ENGINE VERSION
y007xqosjsri0jfz6varsvto5 * host01 Ready Active
Leader 19.03.13
uzmn0vu8niysotxmvfajywpo7 host02 Ready Active
19.03.13
$ docker swarm leave
Error response from daemon: You are attempting to leave the swarm on a node that
is participating as a manager. Removing the last manager erases all current sta
te of the swarm. Use `--force` to ignore this message.
$ docker swarm leave --force
Node left the swarm.
$ docker node ls
Error response from daemon: This node is not a swarm manager. Use "docker swarm
init" or "docker swarm join" to connect this node to swarm and try again.
$
Terminal Host 2
Your Interactive Bash Terminal.
A good starting point is executing `docker`
$ docker swarm join --token SWMTKN-1-43q6mpi7bz0ajfgwac15lzprke6bxmkq7iqqhghwx0bosinwqf-ctw3m1sm1wvot7c46pcjn41h7 172.17.0.41:2377
This node joined a swarm as a worker.
$