$ ###############################################################

# WARNING!!!! #

# This is a sandbox environment. Using personal credentials #

# is HIGHLY! discouraged. Any consequences of doing so are #

# completely the user's responsibilites. #

# #

# The PWD team. #

###############################################################

[node1] (local) root@192.168.0.48 ~

$

[node1] (local) root@192.168.0.48 ~

$

[node1] (local) root@192.168.0.48 ~

$ docker swarm init

Error response from daemon: could not choose an IP address to advertise since this system has multiple addresses on different interfaces (172.18.0.3 on eth1 and 192.168.0.48 on eth0) - specify one with --advertise-addr

[node1] (local) root@192.168.0.48 ~

$ docker swarm init --advertise-addr 192.168.0.48Swarm initialized: current node (w0dq3ooo7ac1qikt6kyn6v5sk) is now a manager.

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

docker swarm join --token SWMTKN-1-40f2ighsz4zgo8hv1edgznip8i5so7th25634vyi3hzkv4rtvd-c9cmp57spnvunsvwfents266l 192.168.0.48:2377

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

[node1] (local) root@192.168.0.48 ~

$ docker node ls

ID HOSTNAME STATUS AVAILABILITY MANAGER STATUS ENGINE VERSION

w0dq3ooo7ac1qikt6kyn6v5sk \* node1 Ready Active Leader 18.06.1-ce

nyapzf23rhasgbuw48pa253s0 node2 Ready Active 18.06.1-ce

ixa4xmd0qhbycihqgxv9ccokx node3 Ready Active 18.06.1-ce

[node1] (local) root@192.168.0.48 ~

$

[node1] (local) root@192.168.0.48 ~

$

[node1] (local) root@192.168.0.48 ~

$ docker service ls

ID NAME MODE REPLICAS IMAGE PORTS

[node1] (local) root@192.168.0.48 ~

$

[node1] (local) root@192.168.0.48 ~

$

[node1] (local) root@192.168.0.48 ~

$ docker ps

CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES

[node1] (local) root@192.168.0.48 ~

$

[node1] (local) root@192.168.0.48 ~

$ docker service create --replicas 7 -p 80:80 --name myservice pradeepdocker123/myapache2:3.0

qbktq10bu4p2l1kdrh5sjtiiy

overall progress: 7 out of 7 tasks

1/7: running [==================================================>]

2/7: running [==================================================>]

3/7: running [==================================================>]

4/7: running [==================================================>]

5/7: running [==================================================>]

6/7: running [==================================================>]

7/7: running [==================================================>]

verify: Service converged

[node1] (local) root@192.168.0.48 ~

$

[node1] (local) root@192.168.0.48 ~

$ docker ps

CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES

69ea2f44cb2c pradeepdocker123/myapache2:3.0 "/bin/sh -c 'apachec…" 14 seconds ago Up 11 seconds 80/tcp myservice.1.hwesr5lakzl1v6rf8otyscnxr

191d8e09d2fc pradeepdocker123/myapache2:3.0 "/bin/sh -c 'apachec…" 14 seconds ago Up 12 seconds 80/tcp myservice.5.iavgs4coghtp6512q5oa33i5l

[node1] (local) root@192.168.0.48 ~

$

[node1] (local) root@192.168.0.48 ~

$

[node1] (local) root@192.168.0.48 ~

$

[node1] (local) root@192.168.0.48 ~

$ docker service ls

ID NAME MODE REPLICAS IMAGE PORTS

qbktq10bu4p2 myservice replicated 7/7 pradeepdocker123/myapache2:3.0 \*:80->80/tcp

[node1] (local) root@192.168.0.48 ~

$

[node1] (local) root@192.168.0.48 ~

$ docker swarm ls

Usage: docker swarm COMMAND

Manage Swarm

Commands:

ca Display and rotate the root CA

init Initialize a swarm

join Join a swarm as a node and/or manager

join-token Manage join tokens

leave Leave the swarm

unlock Unlock swarm

unlock-key Manage the unlock key

update Update the swarm

Run 'docker swarm COMMAND --help' for more information on a command.

[node1] (local) root@192.168.0.48 ~

$

[node1] (local) root@192.168.0.48 ~

$

[node1] (local) root@192.168.0.48 ~

$ docker service myservice list

Usage: docker service COMMAND

Manage services

Commands:

create Create a new service

inspect Display detailed information on one or more services

logs Fetch the logs of a service or task

ls List services

ps List the tasks of one or more services

rm Remove one or more services

rollback Revert changes to a service's configuration

scale Scale one or multiple replicated services

update Update a service

Run 'docker service COMMAND --help' for more information on a command.

[node1] (local) root@192.168.0.48 ~

$ docker service myservice ls

Usage: docker service COMMAND

Manage services