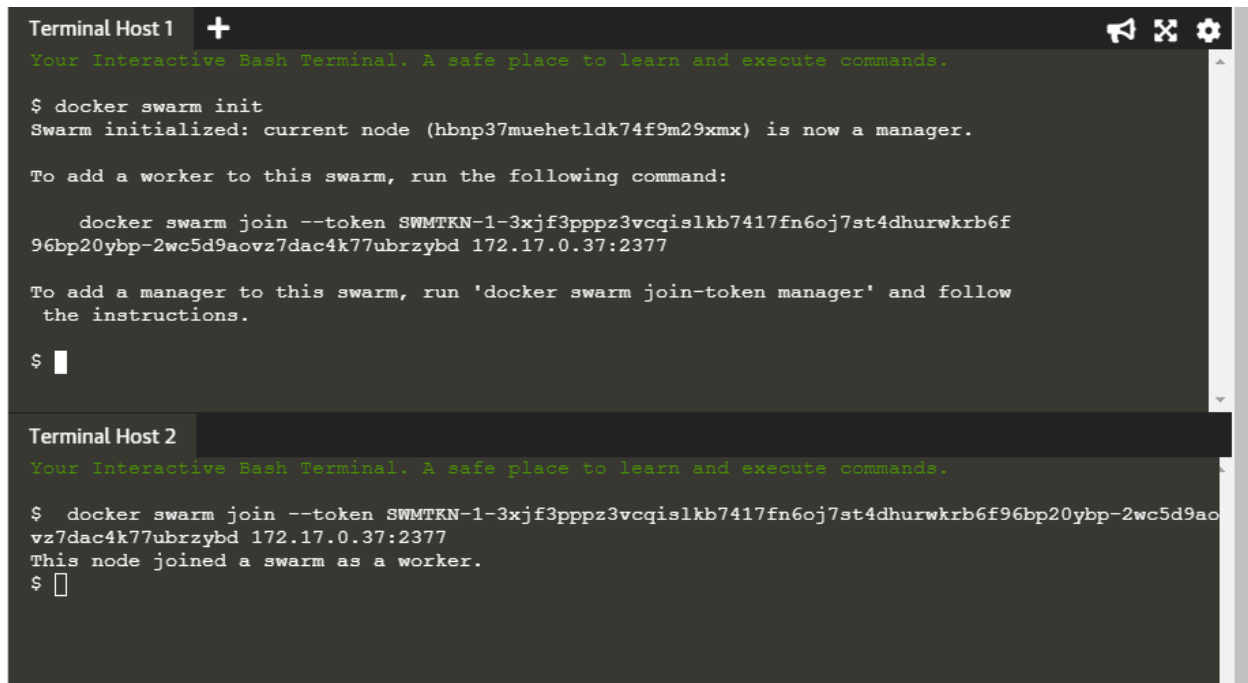


Experiment 9

Aim: Running multiple services using Docker Swarm

1) Creating manager and worker node



The image shows two terminal windows. The top window, titled 'Terminal Host 1', shows the execution of 'docker swarm init', which initializes a new Docker Swarm with the current node as the manager. It provides a long token for adding worker nodes. The bottom window, titled 'Terminal Host 2', shows the execution of 'docker swarm join' with the provided token, successfully adding Host 2 as a worker node to the swarm.

```
Terminal Host 1 +
Your Interactive Bash Terminal. A safe place to learn and execute commands.

$ docker swarm init
Swarm initialized: current node (hbnp37muehetldk74f9m29xmx) is now a manager.

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

    docker swarm join --token SWMTKN-1-3xjf3pppz3vcqislkb7417fn6oj7st4dhurwkrb6f
96bp20ybp-2wc5d9aovz7dac4k77ubrzybd 172.17.0.37:2377

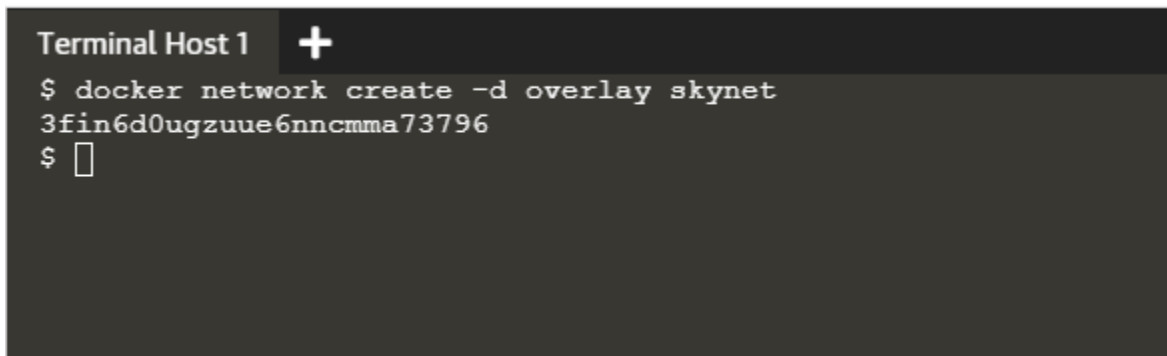
To add a manager to this swarm, run 'docker swarm join-token manager' and follow
the instructions.

$ █

Terminal Host 2
Your Interactive Bash Terminal. A safe place to learn and execute commands.

$ docker swarm join --token SWMTKN-1-3xjf3pppz3vcqislkb7417fn6oj7st4dhurwkrb6f96bp20ybp-2wc5d9aovz7dac4k77ubrzybd 172.17.0.37:2377
This node joined a swarm as a worker.
$ █
```

2) Creating the network



The image shows a terminal window titled 'Terminal Host 1' where the command 'docker network create -d overlay skynet' is executed. The command is successful, and the prompt returns.

```
Terminal Host 1 +
$ docker network create -d overlay skynet
3fin6d0ugzuue6nncmma73796
$ █
```

3) Deploying the service and running docker service ls

```
Terminal Host 1 +
$ docker network create -d overlay skynet
3fin6d0ugzuue6nnmma73796
$ docker service create --name http --network skynet --replicas 2 -p 80:80 katac
erverocker-http-s
xfb81jfdike9qtr0w3x0qqrk7
overall progress: 2 out of 2 tasks
1/2: running
2/2: running
verify: Service converged
$
```

4) Running docker ps on both the terminal

```
Terminal Host 1 +
$ docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED
1acc096c4314      katacoda/docker-http-server:latest "/app"             31
seconds ago        Up 29 seconds      80/tcp             http.1.ikz56jx4id1hdexx
0lhkwr3qa
$
```

```
Terminal Host 2
Your Interactive Bash Terminal. A safe place to learn and execute commands.

$ docker swarm join --token SWMTKN-1-3xjf3pppz3vcqislkb7417fn6oj7st4dhurwkrb6f96bp20ybp-2wc5d9aovz7dac4k77ubrzyb172.17.0.37:2377
This node joined a swarm as a worker.
$ docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED
STATUS            PORTS              NAMES
be77f9811b72      katacoda/docker-http-server:latest "/app"             36 seconds ago
Up 34 seconds      80/tcp             http.2.17eapf7oin99rvlyzz61os9fq
$
```

5) Running docker service ps http

```
Terminal Host 1 +
$ docker service ps http
ID            NAME          IMAGE              ERROR             NOD
E            DESIRED STATE  CURRENT STATE      ERROR             NOD
PORTS
ikz56jx4idlh  http.1        katacoda/docker-http-server:latest  hos
t01           Running       Running about a minute ago
17eapf7oin99  http.2        katacoda/docker-http-server:latest  hos
t02           Running       Running about a minute ago
$
```

6) Running docker service inspect

```
Terminal Host 1 +
$ docker service inspect --pretty http

ID:          xfb81jfdike9qtr0w3x0qqrk7
Name:        http
Service Mode: Replicated
  Replicas:  2
Placement:
UpdateConfig:
  Parallelism: 1
  On failure:  pause
  Monitoring Period: 5s
  Max failure ratio: 0
  Update order: stop-first
RollbackConfig:
  Parallelism: 1
  On failure:  pause
```

7) Scaling up

```
Terminal Host 1 +
$ docker service scale http=5
http scaled to 5
overall progress: 5 out of 5 tasks
1/5: running
2/5: running
3/5: running
4/5: running
5/5: running
verify: Service converged
$
```

8) Checking with docker ps on both the terminal

```
Terminal Host 1 +
$ docker ps
CONTAINER ID        IMAGE               COMMAND                  CREATED
d3a79c917efc       katacoda/docker-http-server:latest "/app"                  38
seconds ago        Up 37 seconds      80/tcp                  http.3.6od5heponz6j87su
uekvepox4
329f0b4b19c0       katacoda/docker-http-server:latest "/app"                  38
seconds ago        Up 37 seconds      80/tcp                  http.5.t7y41u2od6j8uw7
rw3uuf4hr
1acc096c4314       katacoda/docker-http-server:latest "/app"                  2 m
inutes ago         Up 2 minutes       80/tcp                  http.1.ikz56jx4id1hdexx
0lhkwr3qa
$
```

```
Terminal Host 2
$ docker ps
CONTAINER ID        IMAGE               COMMAND                  CREATED
f722cd76d8f6       katacoda/docker-http-server:latest "/app"                  43 seconds ago
Up 42 seconds      80/tcp                  http.4.i8r2mh0s0mp0uq2vtlwgp56z
be77f9811b72       katacoda/docker-http-server:latest "/app"                  2 minutes ago
Up 2 minutes       80/tcp                  http.2.17eapf7oin99rvlyzz61os9fq
$
```