

DOCKER SWARM PRACTICAL

{P.S- ALWAYS FIRST CREATE SWARM AND
CONNECT WITH OTHER CONTAINER THEN CREATE
NETWORK}

FOR DOCKER-CONTAINER 1 WE WILL RUN BELOW
COMMANDS

1 `docker swarm init`

2 `docker network create --driver overlay
funnet`

3 `docker network ls`

4 `docker service create --name flash --network
funnet --replicas 3 nginx`

5 `docker ps`

6 `docker inspect
flash.1.g8icfwatikvoft1lbwxvj17qo`

The screenshot shows the AWS CloudShell interface. At the top, there's a navigation bar with various AWS services like EC2, VPC, S3, IAM, CloudWatch, Aurora and RDS, Simple Notification Service, Simple Queue Service, Support, Route 53, Lambda, and CloudFront. The region is set to Asia Pacific (Mumbai). The terminal window shows a JSON configuration for a Docker network, followed by a series of commands to initialize Docker Swarm, create a network, and start services. The output shows the Docker Swarm is initialized successfully, and the network is created. The terminal prompt is root@ip-172-31-15-226:~#. A red box highlights the instance ID i-03ef90a4dfd3b9eed (ec2_docker1) in the terminal output.

```
    "NetworkID": "ac21mrpj6a1b02ac94wv122q",
    "EndpointID": "6c53b7ee8721a69e24c5d1722d2a53ec607ead04314f65bad0b24ac1ef146eaf",
    "Gateway": "",
    "IPAddress": "10.0.1.3",
    "IPPrefixLen": 24,
    "IPv6Gateway": "",
    "GlobalIPv6Address": "",
    "GlobalIPv6PrefixLen": 0,
    "DriverOpts": null,
    "DNSNames": [
      "flash.1.g8icfwatikvof1lbwxvj17go",
      "8be106943eb4"
    ]
  }
}
}
}
root@ip-172-31-15-226:~# history
1 docker swarm init
2 docker network create --driver overlay funnet
3 docker network ls
4 docker service create --name flash --network funnet --replicas 3 nginx
5 docker ps
6 docker inspect flash.1.g8icfwatikvof1lbwxvj17go
7 history
root@ip-172-31-15-226:~#
i-03ef90a4dfd3b9eed (ec2_docker1)
PublicIPs: 13.203.77.83 PrivateIPs: 172.31.15.226
```

FOR 2ND CONTAINER(EC2 INSTANCE)-

1 apt update

2 apt install iputils-ping

3 ping 10.0.1.3

4 history

root@ad835f6d522d:/# exit

exit

root@ip-172-31-12-176:~# history

1 docker swarm join --token SWMTKN-1-2yu56t657oewb67c4t6je36v94pcu9srkpp6tzdhvjc

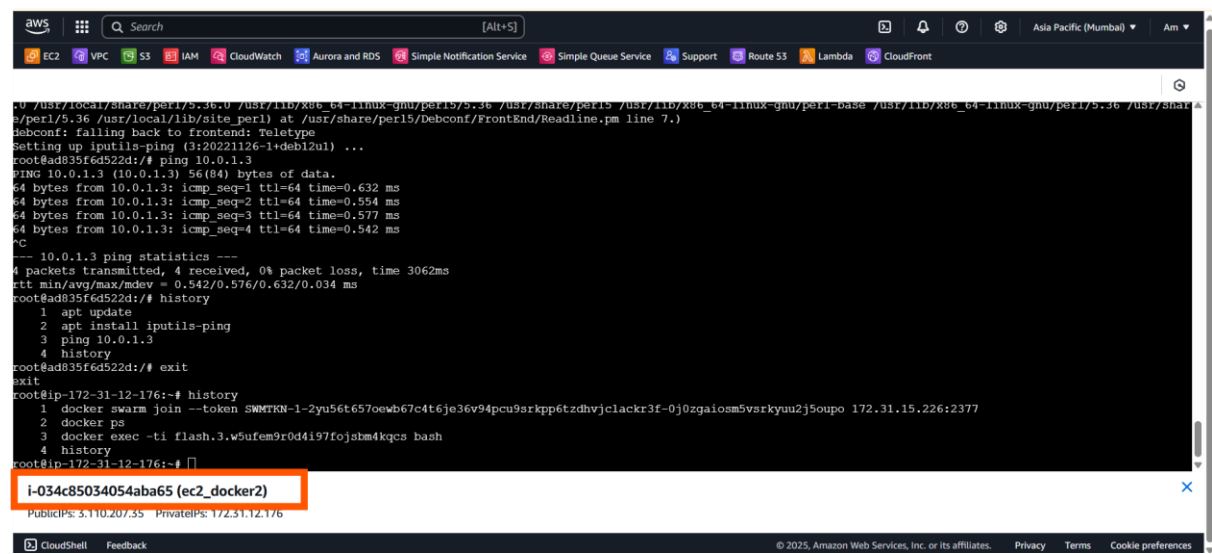
lackr3f-0j0zgaiosm5vsrkyuu2j5oupo

172.31.15.226:2377

2 docker ps

3 docker exec -ti

flash.3.w5ufem9r0d4i97fojsbm4kqcs bash



```
root@ip-172-31-12-176:~# history
1 apt update
2 apt install iputils-ping
3 ping 10.0.1.3
4 history
root@ip-172-31-12-176:~# exit
exit
root@ip-172-31-12-176:~# history
1 docker swarm join --token SWMTKN-1-2yu56t657oewb67c4t6je36v94pcu9srkpp6tzhvjclackr3f-0j0zgaiosm5vsrkyuu2j5oupo 172.31.15.226:2377
2 docker ps
3 docker exec -ti flash.3.w5ufem9r0d4i97fojsbm4kqcs bash
4 history
root@ip-172-31-12-176:~#
```

FOR CONNECTING INSTANCE TO
INSTANCE(CONTAINER TO CONTAINER)-

1 apt update

2 apt install iputils-ping

3 ping 10.0.1.3

4 history

5 exit

aws

Search

[Alt+5]

Asia Pacific (Mumbai)

Am

EC2

VPC

S3

IAM

CloudWatch

Aurora and RDS

Simple Notification Service

Simple Queue Service

Support

Route 53

Lambda

CloudFront

64 bytes from 10.0.1.3: icmp_seq=3 ttl=64 time=0.577 ms

64 bytes from 10.0.1.3: icmp_seq=4 ttl=64 time=0.542 ms

^C

--- 10.0.1.3 ping statistics ---

4 packets transmitted, 4 received, 0% packet loss, time 3062ms

rtt min/avg/max/mdev = 0.542/0.576/0.632/0.034 ms

root@ad835fed522d:~# history

1 apt update

2 apt install iputils-ping

3 ping 10.0.1.3

4 history

root@ad835fed522d:~# exit

exit

root@ip-172-31-12-176:~# history

1 docker swarm join --token SWMTKN-1-2yu56t657oewb67c4t6je36v94pcu9srkpp6t2dhvjclackr3f-0j0zgaio5m5vskryu2j5oupo 172.31.15.226:2377

2 docker ps

3 docker exec -ti flash.3.w5ufem9r0d4i97fojsbm4kqcs bash

4 history

root@ip-172-31-12-176:~# docker exec -ti flash.3.w5ufem9r0d4i97fojsbm4kqcs bash

root@ad835fed522d:~# history

1 apt update

2 apt install iputils-ping

3 ping 10.0.1.3

4 history

5 exit

6 history

root@ad835fed522d:~#

i-034c85034054aba65 (ec2_docker2)

PublicIPs: 3.110.207.35 PrivateIPs: 172.31.12.176

CloudShell

Feedback

© 2025, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)