

# **IBM DBA USECASE 1- DOCKER HIGH AVAILABILITY**

Pruthvi Raj Reddy Chukkannagari-16200936

Naga Mounika Dandamudi-16208510

Suhas Sai Raparathi-16208254

Sravanti Ravi-16183040

## Use Case:

Docker Problem 1: Demonstrate how Docker can be leveraged to create a highly available Web service. In production environments, companies set up redundant nodes so that a backup node can take over when the active node fails. Create a Docker image and run multiple container instances of that image. Investigate how your solution enables (a) failover to the backup node; and (b) failback to the formerly active node. Describe how your solution implements these functions.

### 1. Creating the docker machines

```
docker-machine create -d virtualmachine manager1
```

```
docker-machine create -d virtualmachine secondary
```

```
docker-machine create -d virtualmachine node1
```

```
docker-machine create -d virtualmachine node2
```

```
pruthvirajreddy@PRUTHVIRAJREDDY MINGW64 ~ <master>
$ docker-machine ls
NAME      ACTIVE   DRIVER      STATE     URL                  SWARM
DOCKER    ERRORS
default   -        virtualbox   Running   tcp://192.168.99.103:2376
v1.10.3
manager1  *        virtualbox   Running   tcp://192.168.99.100:2376
v1.10.3
node1     -        virtualbox   Running   tcp://192.168.99.101:2376
v1.10.3
node2     -        virtualbox   Running   tcp://192.168.99.102:2376
v1.10.3
secondary -        virtualbox   Running   tcp://192.168.99.106:2376
v1.10.3
```

### 2. Docker swarm discovery using console

```
eval $(docker-machine env manager1)
```

```
docker run --restart=unless-stopped -d -p 8500:8500 -h consul1 progrium/consul -server -bootstrap
```

```
docker ps
```

```
pruthvirajreddy@PRUTHUIRAJREDDY MINGW64 ~ (master)
$ docker -H=tcp://192.168.99.101:2376 run -d swarm join --advertise=192.168.99.101:2376 consul://192.168.99.100:8500/
Unable to find image 'swarm:latest' locally
latest: Pulling from library/swarm
25da0aa87182: Pull complete
45707a9f4c2b: Pull complete
7f0c09406c8f: Pull complete
a3ed95caeb02: Pull complete
Digest: sha256:5f2b4066b2f7e97a326a8bfcfa623be26ce45c26ffa18ea63f01de045d2238f3
Status: Downloaded newer image for swarm:latest
8f892cee9295c67a6afdfb31062c2158eef68db76cdbac0e551fe6bea4697c5a
```

### 3. Deploying swarm manager/agent

docker run --restart=unless-stopped -d -p 3376:2376 swarm manage  
consul://182.162.99.100:8500

docker ps

### 4. Joining the nodes into the cluster

docker -H=tcp://192.168.99.101:2376 run -d swarm join --  
advertise=192.168.99.101:2376 consul://192.168.99.100:8500/

docker -H=tcp://192.168.99.102:2376 run -d swarm join --  
advertise=192.168.99.102:2376 consul://192.168.99.100:8500/

docker ps

```
pruthvirajreddy@PRUTHUIRAJREDDY MINGW64 ~ (master)
$ docker -H=tcp://192.168.99.102:2376 run -d swarm join --advertise=192.168.99.102:2376 consul://192.168.99.100:8500/
Unable to find image 'swarm:latest' locally
latest: Pulling from library/swarm
25da0aa87182: Pull complete
45707a9f4c2b: Pull complete
7f0c09406c8f: Pull complete
a3ed95caeb02: Pull complete
Digest: sha256:5f2b4066b2f7e97a326a8bfcfa623be26ce45c26ffa18ea63f01de045d2238f3
Status: Downloaded newer image for swarm:latest
1c66bcb29c52507d606f13bed09ebb079d999dfa6e81632c6e9f7c9f1a683708
```

```
pruthvirajreddy@PRUTHUIRAJREDDY MINGW64 ~ (master)
$ docker ps
CONTAINER ID        IMAGE               COMMAND                  CREATED
STATUS            PORTS              NAMES
0f6e27ce7439       swarm              "/swarm manage consul"  5 hours ago
Up 5 hours        2375/tcp, 0.0.0.0:3376->2376/tcp   admiring_keller
6071ce1d4eab       progridium/consul  "/bin/start -server -"  5 hours ago
Up 5 hours        53/tcp, 53/udp, 8300-8302/tcp, 8400/tcp, 8301-8302/udp, 0.0.0.0:8500->8500/tcp   amazing_tesla
```

## 5. Creating primary server and its replica

docker stop 0f6e && docker rm

docker run --restart=unless-stopped -d -p 3375:2376 swarm manage --replication --advertise 192.168.99.100:3376 consul://192.168.99.100:8500

docker logs 4f73

```
pruthvirajreddy@PRUTHUIRAJREDDY MINGW64 ~ <master>
$ docker stop 0f6e && docker rm 0f6e
0f6e
0f6e
```

```
pruthvirajreddy@PRUTHUIRAJREDDY MINGW64 ~ <master>
$ docker run --restart=unless-stopped -d -p 3375:2376 swarm manage --replication --advertise 192.168.99.100:3376 consul://192.168.99.100:8500
4f7342702a58ea72f915179ae10ff535714da38e1da4dd19799dcb28c7b26a68
```

```
pruthvirajreddy@PRUTHUIRAJREDDY MINGW64 ~ <master>
$ docker logs 4f73
time="2016-04-11T02:07:38Z" level=info msg="Initializing discovery without TLS"
time="2016-04-11T02:07:38Z" level=info msg="Listening for HTTP" addr=":2375" proto=tcp
time="2016-04-11T02:07:38Z" level=info msg="Leader Election: Cluster leadership lost"
time="2016-04-11T02:07:38Z" level=info msg="Leader Election: Cluster leadership acquired"
```

## 6. Creating secondary server with swarm manager

eval \$(docker-machine env secondary)

docker run --restart=unless-stopped -d -p 3376:2376 swarm manage --replication --advertise 192.168.99.106:3376 consul://192.168.99.100:8500

docker logs f5b3

```
pruthvirajreddy@PRUTHUIRAJREDDY MINGW64 ~ <master>
$ eval $(docker-machine env secondary)

pruthvirajreddy@PRUTHUIRAJREDDY MINGW64 ~ <master>
$ docker run --restart=unless-stopped -d -p 3376:2376 swarm manage --replication --advertise 192.168.99.106:3376 consul://192.168.99.100:8500
Unable to find image 'swarm:latest' locally
latest: Pulling from library/swarm
25da0aa87182: Pull complete
45707a9f4c2b: Pull complete
7f0c09406c8f: Pull complete
a3ed95cae02: Pull complete
Digest: sha256:5f2b4066b2f7e97a326a8bfcfa623be26ce45c26ffa18ea63f01de045d2238f3
Status: Downloaded newer image for swarm:latest
f5b3f4e7ee49c80e086ccd98d6714c08cf4134276121c84dd7791cf6c02c0322
```

```
pruthvirajreddy@PRUTHUIRAJREDDY MINGW64 ~ <master>
$ docker logs f5b3
time="2016-04-11T02:21:48Z" level=info msg="Initializing discovery without TLS"
time="2016-04-11T02:21:48Z" level=info msg="Listening for HTTP" addr=":2375" proto=tcp
time="2016-04-11T02:21:48Z" level=info msg="Leader Election: Cluster leadership lost"
time="2016-04-11T02:21:48Z" level=info msg="New leader elected: 192.168.99.100:3376"
```

## 7. Testing the server availability by failing the primary

```
eval $(docker-machine env manager1)
```

```
docker ps
```

```
docker stop 4f73
```

```
eval $(docker-machine env secondary)
```

```
docker logs f5b3
```

```
pruthvirajreddy@PRUTHUIRAJREDDY MINGW64 ~ <master>
$ eval $(docker-machine env manager1)

pruthvirajreddy@PRUTHUIRAJREDDY MINGW64 ~ <master>
$ docker ps
CONTAINER ID        IMAGE               COMMAND
STATUS            PORTS              NAMES
4f7342702a58       swarm              "/swarm manage --repl" 23 minutes ago
Up 24 minutes      2375/tcp, 0.0.0.0:3375->2376/tcp awesome_mahavira
3e96cb3bf2ae       swarm              "/swarm manage consul" 25 minutes ago
Up 25 minutes      2375/tcp, 0.0.0.0:3376->2376/tcp gloomy_wilson
6071ce1d4eab       progrim/consul     "/bin/start -server -" 5 hours ago
Up 5 hours         53/tcp, 53/udp, 8300-8302/tcp, 8400/tcp, 8301-8302/udp, 0.0.0.0:8500->8500/tcp amazing_tesla

pruthvirajreddy@PRUTHUIRAJREDDY MINGW64 ~ <master>
$ docker stop 4f73
4f73
```

```
pruthvirajreddy@PRUTHUIRAJREDDY MINGW64 ~ <master>
$ eval $(docker-machine env secondary)

pruthvirajreddy@PRUTHUIRAJREDDY MINGW64 ~ <master>
$ docker logs f5b3
time="2016-04-11T02:21:48Z" level=info msg="Initializing discovery without TLS"
time="2016-04-11T02:21:48Z" level=info msg="Listening for HTTP" addr=":2375" proto=tcp
time="2016-04-11T02:21:48Z" level=info msg="Leader Election: Cluster leadership lost"
time="2016-04-11T02:21:48Z" level=info msg="New leader elected: 192.168.99.100:3376"
time="2016-04-11T02:33:31Z" level=info msg="Leader Election: Cluster leadership acquired"
```

**Conclusion:** Requests to the **manager** are rerouted to **secondary** when there is failure of manager.