Docker Swarm

It is a technique to create and maintain a cluster of docker Engines.

Any service deployed in any node in a cluster can be accessed by the other nodes in the same cluster.

Worker1

Worker2

Worker3

Manager

Manager: is the one how manages the cluster. Here the manager will initialize the swarm and executes the services on the nodes. Here the manager make sure at all the applications are running on the nodes.

Workers: are nothing but nodes where the services are executed. In fact manager is also one node.

Features of Docker Swarm.

1. High Availability of services
2. Auto Load Balacing.
3. Decentralized access
4. Easy to scale up deployments

Requisites

For this we should have 3 machines which are in same network and each machine should connect to each other.

And Docker engine version 1.12 or more than that should be installed in each machine.

All the nodes within the cluster must have same docker version.

1. To initialize the docker swarm cluster

$ docker swarm init --advertise-addr <manager ip address> where this command is runned that machine will act as manager.

Now copy the token and run the token command on the nodes to join the node to the swarm.

To view the nodes run the command

$ docker node ls

Deploying the services on the nodes

$ docker service create --name "<name>" -p <hostport>:<containerport> <image-name>

The above command create a service only in manager.

$ docker service create --name "<name>" -p port:port --mode global <image-name>

The above command create a services in all the nodes within the swarm

$ docker service create --name "<name>" -p port:port --replicas 2 <image-name>

The above command creates a nodes depends upon the replicas and add the services on the nodes.

To remove the service

$ docker service rm <app-name>

To check the nodes connected

$ docker node ls

To check the services command

$ docker service ls / docker ps

Command to verify in which node the app is running

$ docker service ps <appname>

List the services in the node

$ docker node ps <node-name>

Scaling up of the services

$ docker service scale <app-name>=5

Scaling down the services

$ docker service scale <app-name>=3

Update command to drain or activate any node

$ docker node update --availability drain manager-1 <node> (once any node is drained the service is automatically deployed to active node)

$ docker node update --availability active manager-1 <node>

To remove or leave a node from the swarm

$ docker swarm leave –force <node> (this command leaves the node from the swarm but will display in the node list)

$ docker node rm <node> (this command is used to remove the node from the node list)

To add new node to master.

$ docker swarm join-token worker

To deploy docker-compose file in single network.

$ docker stack deploy -c docker-compose.yml test(test=stack name).

To remove docker stack

$ docker stack rm test