**Distribute Your App Across a Swarm Cluster:**



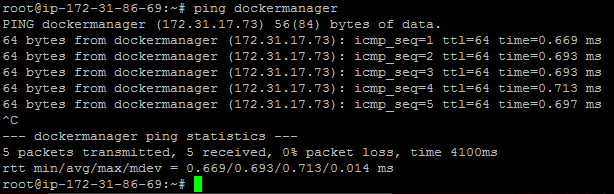
Setting up Docker swarm with multiple nodes

* Edit the /etc/hosts file across the two nodes via gedit or vim and make the following changes:

172.31.17.73dockermanager

172.31.86.69dockerworker1

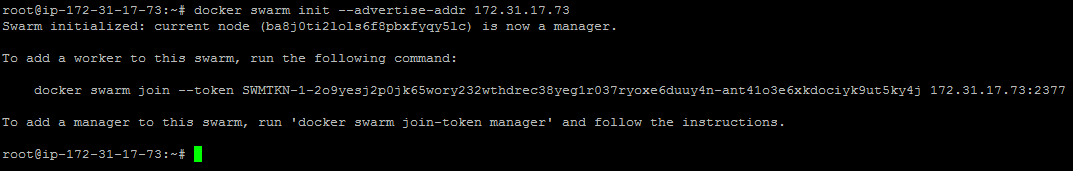
* After modifying the host file with the details mentioned above, check the connectivity with ping between all the nodes
* From Docker Manager Host instance:
* From Docker Worker Node instance:



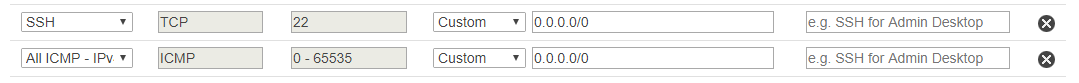
* Initialize the Docker swarm mode by running the following docker command on the dockermanager node

docker swarm init --advertise-addr<manager node IP address>

docker swarm init --advertise-addr172.31.17.73



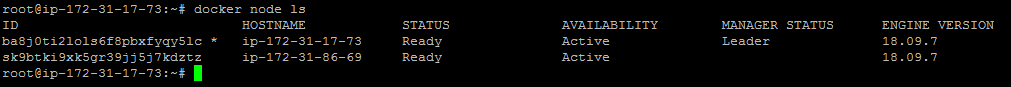
* Once the swarm cluster is initialized, allow the ports mentioned below in security groups





* While initializing the Docker swarm cluster, you will get docker swarm join command which can be executed on node manager to add node to swarm cluster
* Run the command below to see the node status

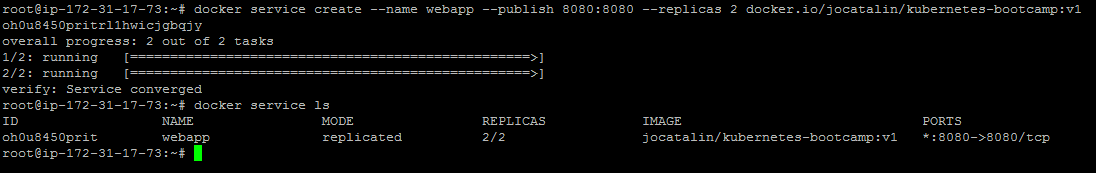
docker node ls



Step 4.5.3: Deploying a custom Docker image to a Docker swarm cluster

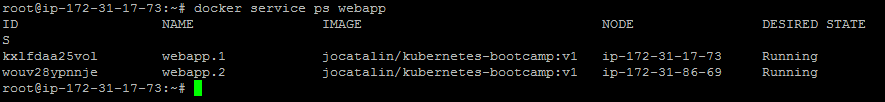
* Create service in Docker swarm cluster

docker service create --name webapp --publish 8080:8080 --replicas 2 jocatalin/kubernetes-bootcamp:v1



* You can now validate if Docker containers got deployed on both nodes or not using the command below

docker service pswebapp



**Please Note: We can validate the application using the curl command to see if the application is up and running.**

