**Deploy Docker with Swarm on CentOS 7.2**

## Prerequisites

Docker requires a 64-bit installation regardless of your CentOS version. Also, your kernel must be 3.10 at minimum, which CentOS 7 runs.

To check your current kernel version, open a terminal and use uname -r to display your kernel version:

$ uname -r

3.10.0-229.el7.x86\_64

## Install Docker

## You can install using the yum package manager.

### Install with yum

Log into your machine as a user with sudo or root privileges.

Make sure your existing yum packages are up-to-date.

$ sudo yum **update**

Add the yum repo.

$ sudo tee /etc/yum.repos.d/docker.repo <<-'EOF'

[dockerrepo]

name=Docker Repository

baseurl=https://yum.dockerproject.org/repo/main/centos/$releasever/

enabled=1

gpgcheck=1

gpgkey=https://yum.dockerproject.org/gpg

EOF

### Install the Docker package.

$ sudo yum **install** docker-**engine**

**Start the Docker daemon.**

$ sudo service docker **start**

Verify docker is installed correctly by running a test image in a container.

$ sudo docker run hello-world

Unable to find image 'hello-world:latest' locally

latest: Pulling **from** hello-world

a8219747be10: Pull complete

91c95931e552: Already exists

hello-world:latest: The image you are pulling has been verified.

Verify docker version by run this command:

$ docker version

Start the Docker daemon.

$ service docker start

Stop the Docker daemon.

$ service docker stop

## Setup and configure Docker with Swarm

**Prerequisites**

* Configured 4 nodes with Docker installed. (Docker client, swarm master and two nodes)
* All the nodes should be able to talk to each other using public or private IP addresses.

Start the Docker daemon on all the nodes.

$ sudo <path to>/docker daemon -H 0.0.0.0:5555 &

Download the Docker container in the master node.

$ sudo docker -H tcp://<master\_ip>:2375 run --rm swarm create

Write down the swarm token generated by the above command as you need it for the entire cluster set up

From the Docker client node, execute the following Docker command to join it to the cluster.

$ docker -H tcp://<node\_1\_ip>:2375 run -d swarm join --addr=<node1\_ip>:2375 token://<cluster\_token>

  Verify the swarm setup on node1 using the following command.

$ sudo docker -H tcp://<node\_1\_ip>:2375 ps

Repeat the above steps for Node2, Node3… by replacing node1’s IP with node2’s IP.

Once you have joined all the nodes to the cluster, set up a swarm manager on the swarm master node using the following command.

$ sudo docker -H tcp://<master\_ip>:2375 run -d -p 5000:5000 swarm manage token://<cluster\_token>

Replace master\_ip and cluster\_token with relevant values with the token which you get.

All the components needed for the cluster set up is in place now. You can start managing your swarm cluster using swarm manager residing in the master node. To list all the nodes in the cluster, execute the following Docker command from the docker client node.

$ sudo docker -H tcp://<master\_ip>:2375 run --rm swarm list token://<cluster\_token>

Execute the following command from the client to get the information about the cluster.

$ sudo docker -H tcp://<master\_ip>:5001 info

Finally, you can test the cluster set up by deploying a container onto the cluster. Run a test busybox container for the docker client using the following command. Execute the following command from the client to get the information about the cluster.

$ sudo docker -H tcp://<master\_ip>:5001 run -dt --name swarm-test busybox /bin/sh /bin/sh

Once the container is deployed, you can list the running docker container using the following command.

$ sudo docker -H tcp://<master\_ip>:5001 ps