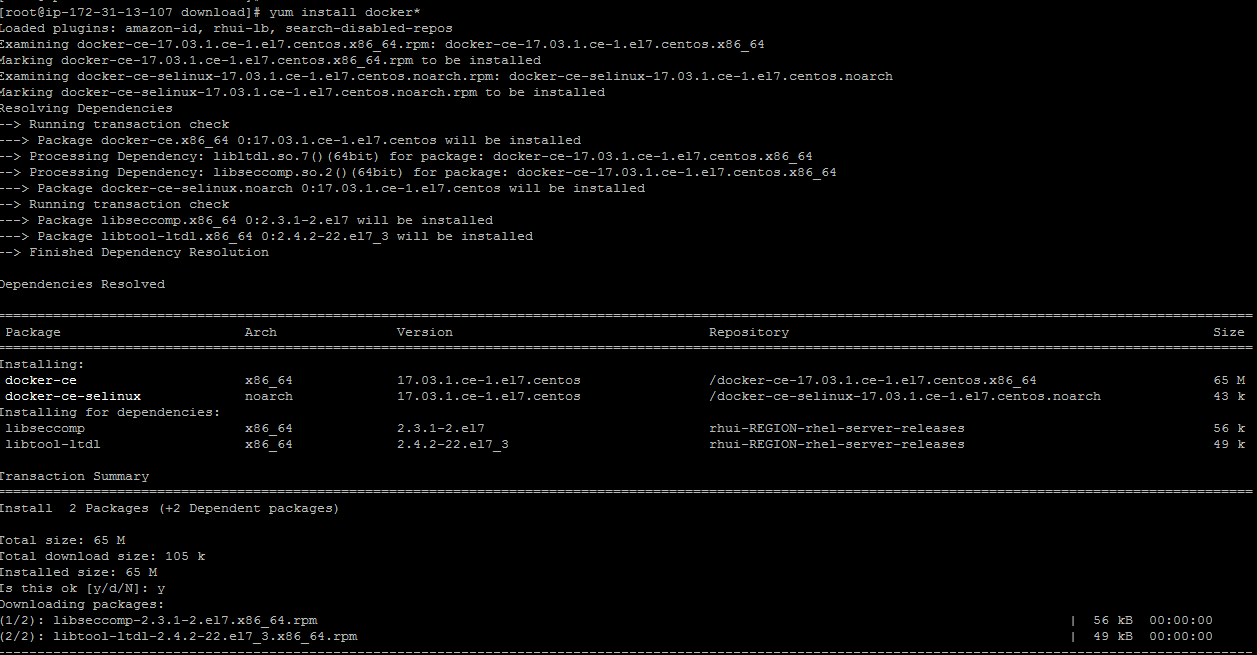
# DOCKER INSTALLATION AND BASIC OVERVIEW

1. **Installation:**

#yum install docker\*

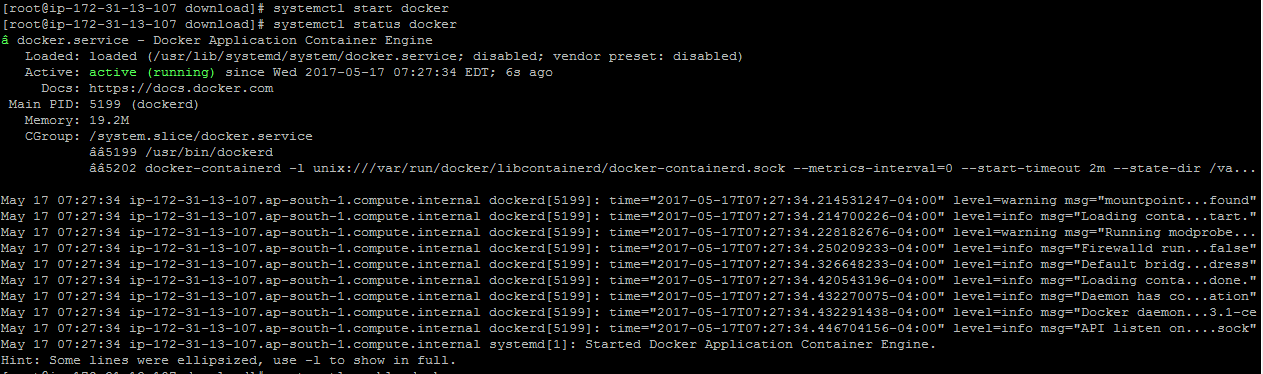


1. **Start the docker services**

# systemctl start docker

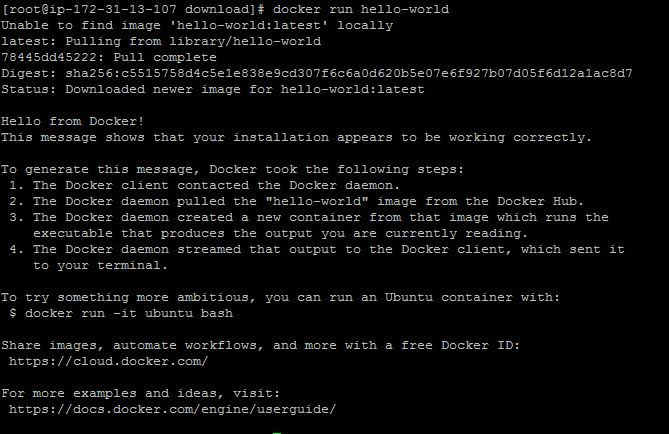
# systemctl status docker

#systemctl enable docker



1. **Start your first program**

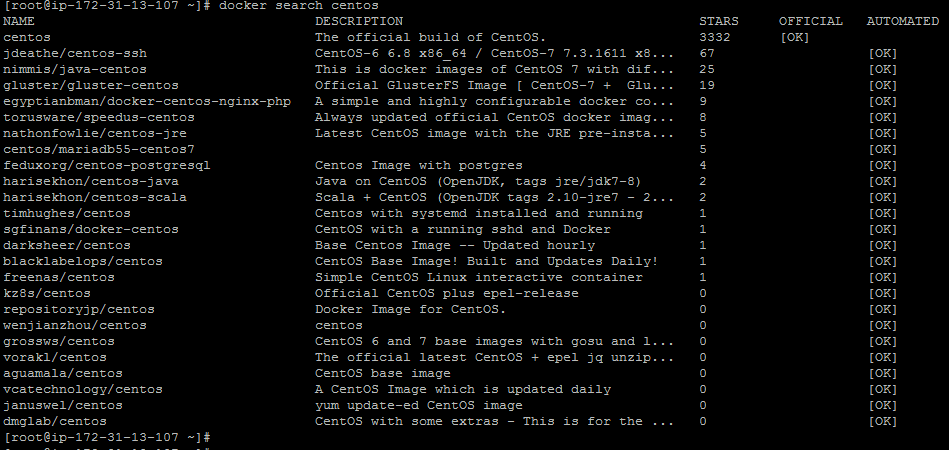
#docker run hello-world



**Note**- In order to start and run a Docker container, first an image must be downloaded from Docker Hub on your host. Docker Hub offers a great deal of free images from its repositories.

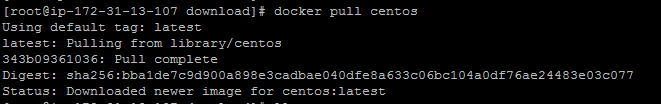
To search for a Docker image, Ubuntu for instance, issue the following command:

# docker search centos



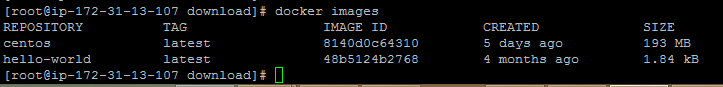
1. After you decided on what image you want to run based on your needs, download it locally by running the below command (in this case an Ubuntu image is downloaded and used):

# docker pull centos



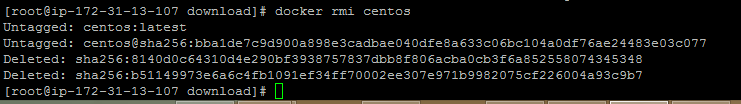
To list all the available Docker images on your host issue the following command:

# docker images



If you don’t need a Docker image anymore and you want to remove it from the host issue the following command?

# docker rmi centos

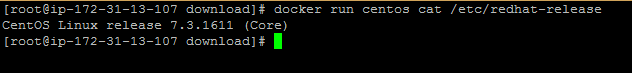


To create and run a container, you need to run a command into a downloaded image

# docker run [local image] [command to run into container]

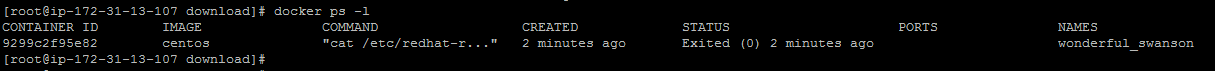
To display the distribution version file inside the container

# docker run centos cat /etc/redhat-release



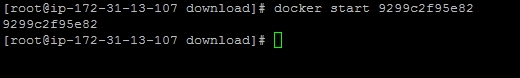
1. To run one of the containers again with the command that was executed to create it, first you must get the container ID (or the name automatically generated by Docker) by issuing the below command,

# docker ps -l



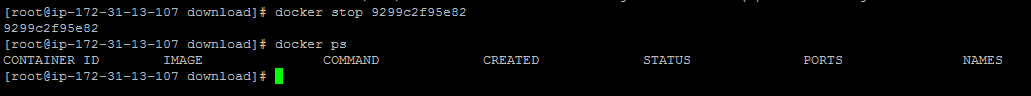
1. Once the container ID has been obtained, you can start the container again with the command that was used to create it, by issuing the following command:

# docker start 9299c2f95e82



1. In case the container is running state, you can get it’s ID by issuing *docker ps* command. To stop the running container issue docker stop command by specifying the container ID or auto-generated name.

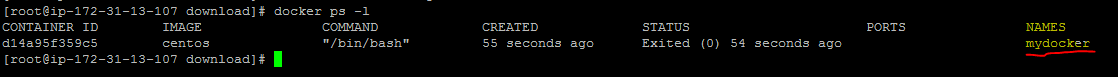
# docker stop 9299c2f95e82



1. The container ID would be to allocate a unique name for every container you create by using the --name option on command line

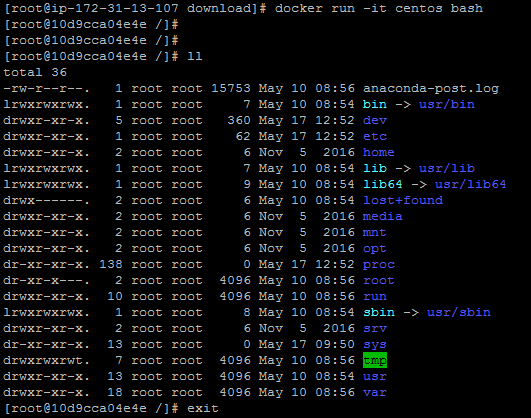
# docker run --name mydocker centos





1. In order to interactively connect into a container shell session, and run commands as you do on any other Linux session, issue the following command:

# docker run -it centos bash



# DOCKER SWARM CLUSTER

**Installation and H/W requirement:-**

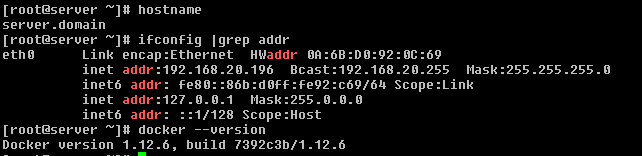
1. Install the Docker version >=1.12 via rpm or via source code on all the nodes.
2. Docker swarm is in built in Docker version >=1.12 so no other package is required.
3. Make sure you have at least 2GB of memory and 2 CPUs for smooth functioning.
4. All the nodes must be accessible on network and connected with each other via network.

**Docker repo access:**

1. Please create account on Docker hub [**https://hub.docker.com/login/**](https://hub.docker.com/login/) **.** It’s free and useful when you will try to download and upload Docker image across the nodes.

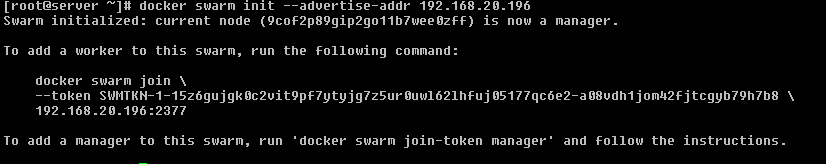
**Set up:**

1. **On server /manager:-**



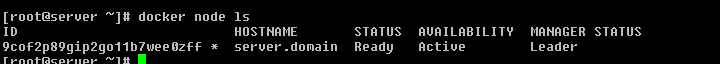
1. Initialize the docker swarm by below command

#docker swarm init --advertise-addr 192.168.20.196



**Note** –This will generate a token to add worker to this cluster.

1. Check the status of cluster, as we can see only one node which is a manager. We need to add some worker node in this cluster.

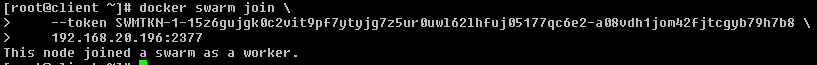


1. **On Worker node:**
2. Run the token as shown in the above process.

# docker swarm join \

> --token SWMTKN-1-15z6gujgk0c2vit9pf7ytyjg7z5ur0uwl62lhfuj05177qc6e2-a08vdh1jom42fjtcgyb79h7b8 \

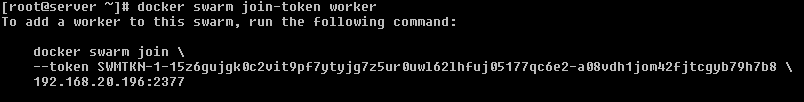
> 192.168.20.196:2377



**Note** – In case you forget the token to add worker you can run below command on server/manager to retrieve the token.

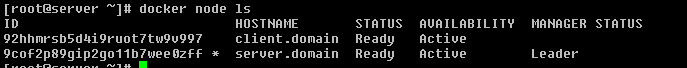
#docker swarm join-token worker # For worker node

#docker swarm join-token manager # For manager node



1. **Commands:**
2. To check the cluster status and node info.

#docker ps –l



1. To promote or demote a node in cluster.

# docker node promote client.domain



1. To demote a node in cluster.

#docker node demote client.domain

1. Login to Docker hub for images.

#docker login -u <username>

1. Download the images form the Docker hub.

#docker pull kufu/pjangra:v2

