**Docker**

**Summary Contain:--**

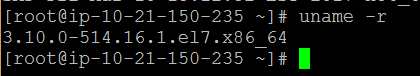
1. Prerequisites
2. Installation steps
3. Daily Administrative task

[Docker](https://github.com/docker/docker) is a tool designed to make it easier to create, deploy, and run applications by using containers. Containers allow a developer to package up an application with all of the parts it needs, such as libraries and other dependencies, and ship it all out as one package. By doing so, thanks to the container, the developer can rest assured that the application will run on any other Linux machine regardless of any customized settings that machine might have that could differ from the machine used for writing and testing the code.

**Prerequisites**

* A 64-bit installation
* Version 3.10 or higher of the Linux kernel. The latest version of the kernel available for you platform is recommended.
* iptables version 1.4 or higher
* git version 1.7 or higher

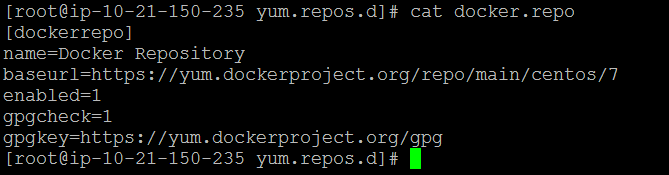
Docker requires that your kernel must be 3.10 at minimum, which Red Hat 7 runs.( Here I’m using AWS Linux instance for the installation)



**Installation steps:-**

Below are the installation steps which we need to follow ,

Update the yum repo.



1. Install the Docker package with

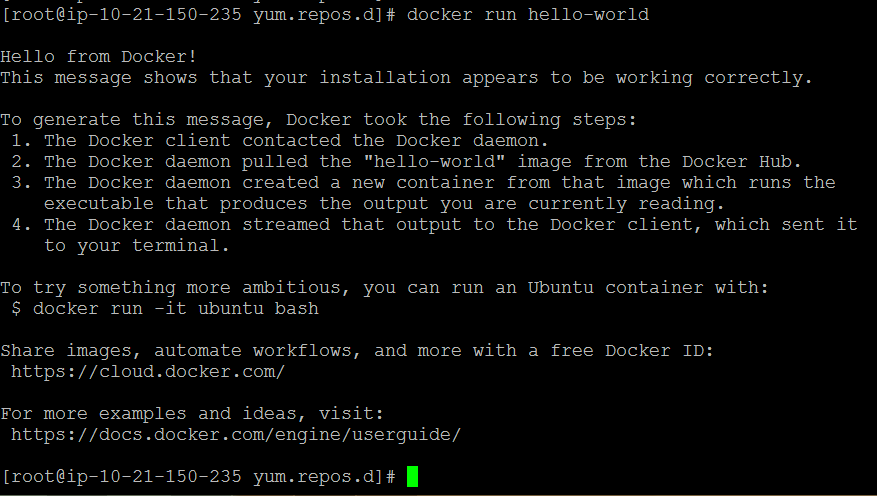
**# yum install docker-engine**

1. Start the Docker daemon.

**# service docker start**

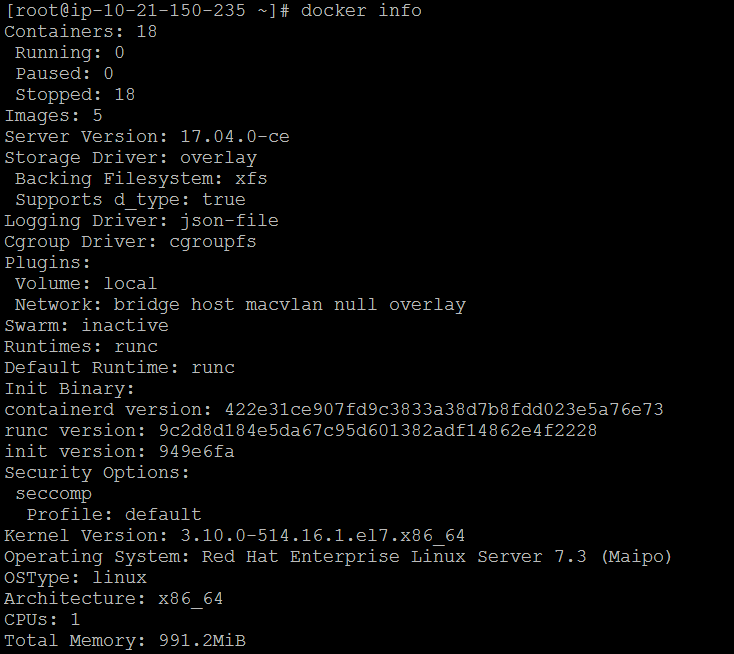
1. Check the sample image creation.

**# docker run hello-world**



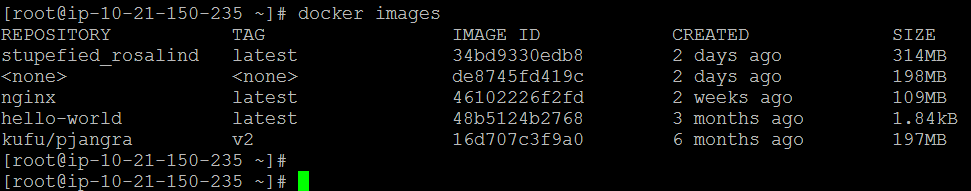
There are also some day to day use command which are helpful daily Administrative task.

**# docker info**



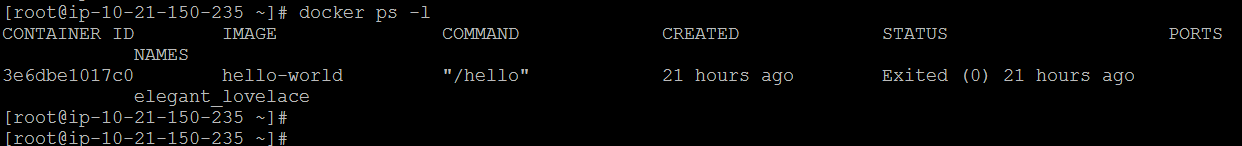
**# docker images**

Here I have created 2/3 different container images for further troubleshooting, let we have move with the nginx image.



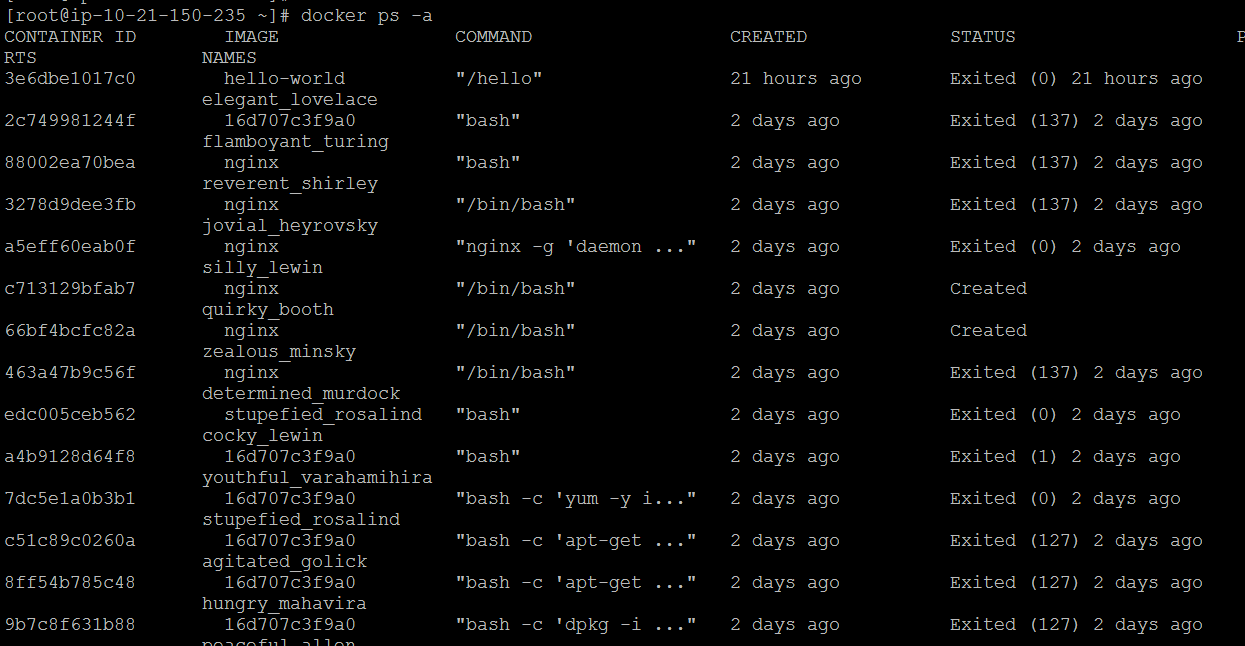
To run one of the containers again with the command that was executed to create it, first you must get the container ID (or the name automatically generated by Docker), which displays a list of the running and stopped (non-running) containers:

**# docker ps –l**



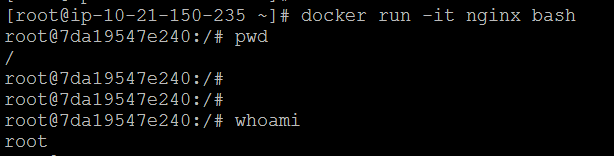
Also we can you see the available container in our environment with the below command.

**# docker ps –a**



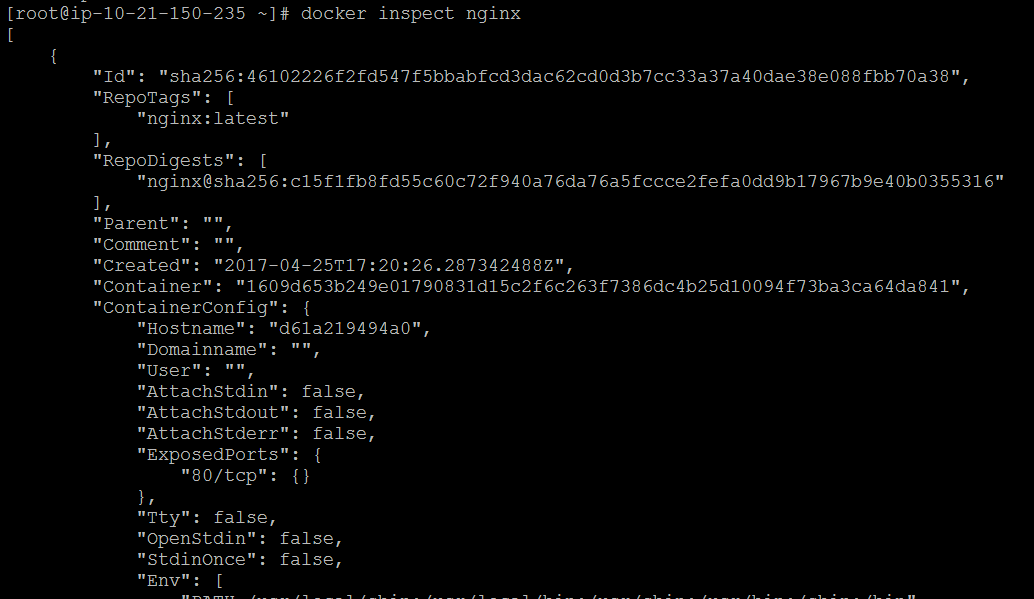
In order to interactively connect into a container shell session, and run commands as you do on any other Linux session, issue the following command:

**# docker run -it nginx bash**



Use the below command to Return low-level information on Docker objects in json format

**# docker inspect nginx**



**# docker rmi 48b5124b2768**

