Docker Basics

Docker hub is a registry where we can find the images and we can share with others and we can do some automatic image processing. Image means it’s a file which have multiple layers (OS or application layers) where we can create a container out of it.

Without any image we cannot create a docker container.

Two types of images: Public Image and Private Image

We can publish docker image on hub.docker.com (i.e. On docker website)

If we want to use public images, it is better to use official public images because unofficial public images might have malicious or harmful code.

Whenever we launch a container with an image it will install specific OS or anything according to the image.

In a git hub we store a code same way in a docker we can store images.

**LINKS TO FOLLOW:** <https://www.youtube.com/watch?v=h9f1kEteklY>

**NOTE: JUST FOLLOW DOCKER COMMAND SECTION ONLY**

Docker Commands

|  |  |
| --- | --- |
| To enable Docker | sudo systemctl enable docker |
| To start docker | sudo systemctl start docker  service docker start |
| To run docker Images with sudo and create a container | sudo docker run ImageName  Ex. sudo docker run hello-world |
| Without using sudo we need to add user into the docker group than we can run docker image on normal user  Once we add user to docker group we need to restart terminal (i.e. log out and log in with that user) | Add user to the docker group,  sudo usermod -a -G docker UserName  Ex. sudo usermod -a -G ans\_admin  sudo chmod 777 /var/run/docker.sock  Run command without sudo  docker run ImageName  Ex. docker run hello-world |
| To see what containers are running | docker ps OR  docker ps -a |
| To check the status of docker whether it is active (running) or not | service docker status |
| To see the images into local repository  Local repository means within our docker server | docker images  docker image ls |
| Whenever we download or pull any docker image from the docker hub it will download latest one by default.  But to download specific version of OS and software of an image than we need to use OS or software name with “:” | docker pull ubuntu  (To install ubuntu on server)  It will download latest one  docker pull ubuntu:16.04  (It will download ubuntu 16.04 version) |
| To create a container of the image | docker run -d --name ContainerName ImageName  d= detach mode  (It is only for latest version of an image)  Ex. docker run -d --name ubuntu\_latest ubuntu  docker run -d --name ContainerName ImageName:VersionNumber  (It is for specific version)  Ex. docker run -d --name ubuntu\_version ubuntu:16.04 |
| To see the containers | docker ps OR  docker ps -a |
| To customize an image | docker tag ImageName:TagName DockerUserName/NewImageName  Ex. docker tag hello-world:latest harshlad726/harsh\_hello\_world\_image |
| To push docker image to the docker hub OR  Push image on to the remote repository on hub.docker.com  But before we push to the docker image to the docker hub we need to provide a [credentials](http://54.234.136.192:8080/credentials)  Without credentials it won’t push images on to the docker hub | To give [credentials](http://54.234.136.192:8080/credentials) we need to log in to the docker hub from the Linux terminal  **Command:**  **docker login**  After command we need to give our Docker Hub account Username and Password  Now, we can push images to the docker hub  **Command:**  **docker push DockerUserName/ImageName**  **Ex. docker push harshlad726/harsh\_image** |
| To remove images from local repository  **NOTE: IF WE USE COMMAND WITHOUT IMAGE, IT WILL LOOK FOR THE CONTAINER TO REMOVE.** | docker image rm -f ImageName OR  docker image rm -f ImageName:VersionNumber(TAG NAME)  f=forcefully  Ex. docker image rm -f harshlad726/harsh\_image  To remove more than one image at a time  docker image rm -f ImageName1 ImageName2 ….  docker image rm -f hello-world harshlad726/harsh\_image |
| To remove the containers from the docker | docker rm -f ContainerName  Ex. docker rm -f 90bcf6b5e9e5  **NOTE: FIRST WE NEED TO REMOVE CONTAINERS AND THAN WE CAN REMOVE IMAGES** |
| Copy file into docker container | docker cp FIleNameWhichWantToCopy ContainerID:/Path  docker cp httpd.yml 0f6bccea81ae:/temp |
| To check the file is copied or not on container | docker exex -it ContainerID /AnyPathOfContainer  docker exec -it 0f6bccea81ae /bin/bash |

Install Docker And Configure

|  |  |
| --- | --- |
| Update yum package | yum update -y |
| To install Docker on EC2 instance | sudo yum install -y docker OR  sudo yum install docker |
| To check docker is installed or not OR  Check docker version | docker version |
| Create Docker Image | |
| Go to ECS from the Services | A screenshot of a cell phone  Description automatically generated |
| Click on “Get Started” | A screenshot of a cell phone  Description automatically generated |
| Click on Sample-app | A screenshot of a cell phone  Description automatically generated |
| Scroll down and  Click on “Next” |  |
| Select “Application Load Balancer” from  “Load Balancer Type”  and click “Next” | A screenshot of a cell phone  Description automatically generated |
| Give the cluster name (TestCluster) and press “Next”  Name contains only Alphabets and Numbers  Should not contains White Speces, Special Characters($@%$&) | A screenshot of a social media post  Description automatically generated |
| Press Create and wait for creation process done | A screenshot of a social media post  Description automatically generated |
| Once process done Click on “View Service” | A screenshot of a social media post  Description automatically generated |
| Created Image | A screenshot of a cell phone  Description automatically generated |
| To Check Docker Image is Running | |
| Go to Details and click on the Link | A screenshot of a cell phone  Description automatically generated |
| Click on the link of the Load balancer | A screenshot of a cell phone  Description automatically generated |
| Copy the DNS name link and pest it into the browser address bar | A picture containing screenshot  Description automatically generated |
| Container working properly | A screenshot of a cell phone  Description automatically generated |

Delete Docker Container and Cluster

|  |  |
| --- | --- |
| Go to ECS | A screenshot of a cell phone  Description automatically generated |
| Click on “ClusterName”  Ex. [TestCluster **>**](https://console.aws.amazon.com/ecs/home?region=us-east-1#/clusters/TestCluster) | A screenshot of a cell phone  Description automatically generated |
| Select Service and  Press “Update Cluster” | A screenshot of a cell phone  Description automatically generated |
| Set the “Number of tasks” value as “0”  And press “Next Step”  Again press “Next Step”  Again press “Next Step” | A screenshot of a cell phone  Description automatically generated |
| Press “Update Service” | A screenshot of a social media post  Description automatically generated |
| Click on “Service Name”  Ex.  [sample-app-service](https://console.aws.amazon.com/ecs/home?region=us-east-1#/clusters/TestCluster/services/sample-app-service/deployments) | A screenshot of a cell phone  Description automatically generated |
| Click “Delete” | A screenshot of a social media post  Description automatically generated |
| Type “delete me” into the textbox and press “Delete” | A screenshot of a cell phone  Description automatically generated |
| Press “Delete Cluster”  We can see into the “Services”,  there are not any services available | A screenshot of a social media post  Description automatically generated |
| Type “delete me” and press “Delete” | A screenshot of a cell phone  Description automatically generated |
| Cluster Deleting in Process | A screenshot of a social media post  Description automatically generated |