Docker

#service docker start/status/stop #docker ps -a #docker images

Pulling #docker pull Jenkins

Create 1st way #docker run -it ubuntu /bin/bash 2nd way(base-img>cont>modified-img) #docker run --name sai -itubuntu /bin/bash #echo "hello"> file1#exit to check diff btwin base & modifid img #docker diff sai create new frommodif img #docker commit sai(modified) sai(new) 3rd way //docker.file to img casei//Dockerfile(recommend) #viDockerfile FROM ubuntu WORKDIR /tmp RUN echo "hello"> /tmp/file1 RUN apt-get update ENV name shahenshaCOPY File1 /tmp ADD test.tar.gz /tmp VOLUME ["/data"] :Wq!

#docker build -t(tag) new-img .(dot-D.filepwd

Case-ii D.file=other file name #mv D.file saifile #docker build -t rahulimg . -f saifile #docker run --name rahulcont -itrahulimg /bin/bash Case-iii / Create zipfile #touch test #tar -cvf test.tar test #gzip test.tar #rm -rf test Cret img #dockerbuild -t holiimg . Cret cont #docker run -it --name holicont holiimg /bin/bash #ls #echo $name ///shahensha

Rename

#docker rename old-name/image-id new

Giving name while creating

#docker run --name sai -it ubuntu /bin/bash

To enter required container

#docker start sai #docker attach sai #docker exec -it webcont /bin/bash #exit

To delete one or all

#docker rm sai #docker rm $(docker ps -a -q)

#docker rmi saimge #docker rmi -f $(docker images -q)

To stop all running

#docker stop $(docker ps -a -q)

Volume sharing (Cont-cont mapping)

1st way by D.file: #vi Dockerfile FROM ubuntu VOLUME ["/data"] Image: #docker build -t img . Container: #dockerrun -it --name cont1 img /bin/bash #ls>> root data .. ... #cd data /#touch file #exit From dis container #docker run -it --name cont2 --privileged=true --volumes-from cont1 ubuntu /bin/bash :/#ls>> root data tmp... ... :/#cd data #ls>> file#touch filooo. #ls>>file filooo. #exit Cont1: #docker start cont1 #docker attach cont1 :/#ls>>>data. #cd data #ls>>filefilooo

2nd way by cmd #docker run -it --name cont3 -v /sai ubuntu /bin/bash #ls >>sai root boot.. ... #cd sai #touch demfileCont4: #docker run -it --name cont4 --privileged=true

--v olumes-from cont3 ubuntu /bin/bash #cd sai>>#ls >>demfile #touch samfile #ls>>demfile samfile #exit Cont3:#docker start cont3 #docker attach cont3 #cd sai #ls>>samfile demfile

Volume (Host-cont)

Host//EC2// Choose a dir under / >>>root #ls >>saifile

Cont #docker run -it --name saicont -v /root:/raj

--privileged=true ubuntu /bin/bash :/#ls>> root boot raj #cd raj #ls>>saifile #touch hostfile #exit

Host/EC2// #ls >> saifile hostfile

Port-expose //Port to port mapping 1st mapping #docker run -td --name webcont -p 80:80 ubuntu

#docker port webcont ///for port num #docker inspect saicont///for volume

#docker exec -it webcont /bin/bash :/#apt-get install apache2 -y :/#cd. /var/www/html :/#echo "hello" > index.html

:/#service apache2 start/status

Goto chrome Url///ec2public ip:port///15.2.3.196:80>> hello

2nd mapping

#docker run -td --name jenkincont -p 8080:8080 Jenkins/Jenkins

Enable 8080 port on EC2 Security grp>inbound rule>add rule 8080 no name

Goto chrome ip:port Url//1.2.3.198:8080

D.hub Creating reserved space on hub to push img

#dicker tag saiimg shahensha/newsaiimg To push #docker push shahensha/newsaiimg

To pull Start service #systemctl satrt/enable docker #dicker pull shahensha/newsaiimg #docker images #docker run -it -- name haricont shahensha/newsaiimg /bin/bash #ls>>#exit

Now delete haricont and saiimg Make img private on d.hub Then login Now try to pull img