**Docker Case Study**

**Problem : Automate Infra allocation for L&D**

**Requirements:-**

1.

2. 3. 4. 5. 6. 7. 8.

Dynamic Allocation of Linux systems for users

Each user should have independent Linux System

Specific training environment should be created in Container

User should not allow to access other containers/images

User should not allow to access docker command

Monitor participants containers

Debug/live demo for the participants if they have any doubts/bug in running applications. Automate container creation and deletion.

**Creating the container image:-**

1. Create a new container from a base image

sudo docker create it name docker\_list ubuntu /bin/bash

2. Start the container

sudo docker start docker\_contain

3. Attach to the container

sudo docker attach docker\_contain

4. Install packages required

apt update

apt install vim apt install gcc

5. Create questions.txt, instructions.txt and save them.

touch questions.txt

touch instructions.txt

6. Commit the container

docker commit a "Manideep" 37f609ba3b38 docker\_contain\_image

Now our training container image is ready.

**Allocating Containers To Users:-**

1. The shell script create\_Containers.sh will automatically create a docker container

for every user.

• users.txt

Manideep

Baswanth

Srikar

• create\_Containers.sh

echo n "Enter name of file with usernames: "

read file

while read user

do

docker create it name $user docker\_contain\_image /bin/bash

done < $file

2. Fill the entries in users.txt with usernames and run the shell script

create\_Containers.sh. This creates a docker container corresponding to each username from users.txt.

3. The user can then start using the allocated container by running the

use\_Containers.sh script.

• use\_Containers.sh

echo n "Enter your username: "

read name

docker start $name

docker attach $name

**Monitoring The Containers:-**

1. To monitor the containers, use the monitor\_Containers.sh script.

• monitor\_Containers.sh

echo n "Enter username of container to be monitored: "

read name

docker logs f $name

**Automating deletion of the containers:-**

1. Automate the deletion using the delete\_Containers.sh script.

• delete\_Containers.sh

echo n "Do you wish to delete containers of all usernames? If yes

enter 'Y' , else enter 'N': "

read option

if [ "$option" == "N" ]

then

echo n "Give the usernames you want to delete and enter 'exit'

at the end: "

while read user

do

if [ "$user" != "exit" ]

then

docker rm $user

else

break

fi

done

else

echo n "Enter name of file containing usernames: "

read file

while read user

do

docker stop $user

docker rm $user

done < $file

fi

2. You can either delete all users or user by name using sh delete\_Containers.sh x .

BY:-

Manideep Nizam (IMT2016026)