DOCKER CASE STUDY

K Puneeth

IMT2016018

SYSTEM REQUIRMENTS:

- 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.

ALLOCATE INDEPENDENT LINUX SYSTEM FOR DIFFERENT USERS:

1. Install the required applications using the following commands
    - apt update
    - apt install vim
    - apt install gcc

2.To allocate resources for different users on the system we create a bash file with name create_containers.sh that creates a docker container for each specified name

- Touch users.txt

- Vim Users.txt

```
alpha
beta
gamma
```

- create_containers.sh

```
echo -n "Enter users file: "
read file
while read user
  do
   docker create -it --name $user <Docker Image>/bin/bash
  done <$file
```

3.Run the shell script run_containers.sh with users.txt , this creates a docker container corresponding to each user name.

4. We can start and attach multiple images to our container using another bash file let it be attach_container.sh

- attach_container.sh

```
echo -n "Enter your username: "
read name
  docker start $name
  docker attach $name
```

5.Doing this we have met the requirements of 1-5.

MONITORING PARTICIPANTS CONTAINERS:

1. To monitor the user containers create a bash file monitor_containers.sh

- Monitor_containers.sh

```
Echo - -n "Enter username of container to be monitored: "
read name
 docker logs  --f $name
```

AUTOMATE CONTAINER DELETION:

1. Create a bash file named delete_containers.sh
   - Delete_containers.sh

```
echo -n "Enter the user list file : "
read file
while read user
  do
   docker stop $user
   docker rm $user
done < $file
```

2.You can delete by using the command sh delete_containers.sh