Docker Case Study:-

Automate Infrastructure Allocation for Learning & Development

Requirements-

- 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.

Allocation of Linux systems for users:-

Containers are created and allocated to each user automatically by executing the shell script *createcontainers.sh.*

1)Fill the entries in Users.txt with usernames .

```
Users.txt

    User_1
    User_2
    User_3
    User_4
```

2)Run the shell script createcontainers.sh. This creates a docker container corresponding to each username from users.txt.

- createcontainers.sh.

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

3)The user can then start using the allocated container by running the u*secontainers.sh* script.

- usecontainers.sh
- echo -n "Enter Username: "

```
read name
docker start $name
docker attach $name
```

Monitoring The Containers

- Container can be monitored by executing the monitorcontainers.sh
- monitorcontainers.sh

```
echo -n "Enter container’s username that need to be monitored"
read name
docker logs -f $name
```

Deleting The Containers

- Deletion of containers can be automated by using shell script *deletetecontainers.sh* script.
- Deletion of single container can be done by using shell script *deletetecontainers.sh -x* script.
-
- deletecontainers.sh

```
echo -n "Enter file name which has list of usernames:  "
read file
while read user
   do
     docker stop $user
     docker rm $user
   done < $file
```

Siddarth Reddy Desu

(IMT2016037)