Docker Case Study

Problem : Automate Infra allocation for L&D

Requirements:-

1. Dynamic Allocation of Linux systems for users
2. Each user should have independent Linux System
3. Specific training environment should be created in Container
4. User should not allow to access other containers/images
5. User should not allow to access docker command
6. Monitor participants containers
7. Debug/live demo for the participants if they have any doubts/bug in running applications.
8. 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 "Rohith" 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`

    ```
    Rohith
    Srujan
    Siddu
    Puneeth
    ```

    - `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:

Nomula Rohith Yogi
(IMT2016072)