

Docker Case Study

Soumith Kumar Dacheppalli
IMT2016110

Allocating Containers To Users

```
soumith@soumith-PC: ~/Desktop/Software_Engineering/docker_assn
soumith@soumith-PC:~/Desktop/Software_Engineering/docker_assn$ cat users.txt
user1
user2
user3
user4
user5
soumith@soumith-PC:~/Desktop/Software_Engineering/docker_assn$ ./create_containers.sh
root@soumith-PC: ~/Desktop/Software_Engineering/docker_assn
root@soumith-PC:~/Desktop/Software_Engineering/docker_assn# ./create_containers.sh
Enter name of file with usernames: users.txt
d1211c5d12120775ff355983f62d0e4b220d579507c1f477d5c9f54312154444
7e7176851733ab3283d7615c8ae732606025e7b127629a541a861662fd3cd61a
a4786a8781fa435cf2cd3b4dbce3a372da552b3d0329af600fb126837d2bc57c
da3dcee112debe5d9294262e68f37b1fc984b2111c0d5f7d82b8d843ade8152c
0ede396f8090c751854d1efb5c467b2aa79ea187e95cbb26870efd41426e181f
root@soumith-PC:~/Desktop/Software_Engineering/docker_assn#
```

Created Containers

```
root@soumith-PC:~/Desktop/Software_Engineering/docker_assn# docker ps -a
CONTAINER ID        IMAGE               COMMAND             CREATED
STATUS            PORTS              NAMES
0ede396f8090       ubuntu             "/bin/bash"        About a minute ago
Created                                     user5
da3dcee112de       ubuntu             "/bin/bash"        About a minute ago
Created                                     user4
a4786a8781fa       ubuntu             "/bin/bash"        About a minute ago
Created                                     user3
7e7176851733       ubuntu             "/bin/bash"        About a minute ago
Created                                     user2
d1211c5d1212       ubuntu             "/bin/bash"        About a minute ago
Created                                     user1
```

For a user to access a container

```
soumith@soumith-PC: ~/Desktop/Software_Engineering/docker_assn
echo -n "Enter your username: "
read name
docker start $name
docker attach $name
~
~

root@soumith-PC: ~/Desktop/Software_Engineering/docker_assn
root@soumith-PC:~/Desktop/Software_Engineering/docker_assn# ./useContainer.sh
Enter your username: user1
user1
root@d1211c5d1212:/# ls
bin  dev  home  lib64  mnt  proc  run  srv  tmp  var
boot  etc  lib  media  opt  root  sbin  sys  usr
root@d1211c5d1212:/# cd usr
root@d1211c5d1212:/usr# ls
bin  games  include  lib  local  sbin  share  src
root@d1211c5d1212:/usr# exit
exit
root@soumith-PC:~/Desktop/Software_Engineering/docker_assn#
```

Monitoring The Containers

```
soumith@soumith-PC: ~/Desktop/Software_Engineering/docker_assn
echo -n "Enter username of container to be monitored: "
read name
docker logs -f $name
~
~

root@soumith-PC: ~/Desktop/Software_Engineering/docker_assn
root@soumith-PC:~/Desktop/Software_Engineering/docker_assn# ./monitorContainers.sh
sh
Enter username of container to be monitored: 
```

Output

```
root@d1211c5d1212:/# exit
exit
root@d1211c5d1212:/# ls
bin  dev  home  lib64  mnt  proc  run  srv  tmp  var
boot  etc  lib  media  opt  root  sbin  sys  usr
root@d1211c5d1212:/# cd usr
root@d1211c5d1212:/usr# ls
bin  games  include  lib  local  sbin  share  src
root@d1211c5d1212:/usr# exit
exit
```

```

echo -n "Enter 'y' to delete the containers of all usernames, else enter 'n': "
read option
if [ "$option" == "n" ]
then
    echo -n "Enter the usernames you want to delete, enter 'exit' when done deleting: "
    while read user
    do
        if [ "$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 rm $user
    done < $file
fi

```

```
root@soumith-PC:~/Desktop/Software_Engineering/docker_assn#
```

It can be seen that the container for user1 has been deleted