

## DOCKERHUB AND IT'S WORKING

Prerequisite: <https://bansalkushagra.medium.com/docker-port-expose-and-publish-3470f4b49ccf>

How to push docker image in docker hub

### Step-1:

Create a container say ubuntu

Command:

```
Docker run -it ubuntu /bin/bash
```

```
ls
```

You will see many files and directory already exists. Create more files here.

Command:

```
Touch file file2 test test2 xyz
```

Also, additionally, install the apache server and files inside tmp/ directory using the command

```
apt-get install apache2 -y
```

To see an HTML webpage go to “cd var/www/ ”

Go to tmp directory

```
cd /tmp
```

```
touch file1 file2
```

```
ls
```

Now, create an image of this updated container.

Command:

```
docker commit <Container_name> <new_container_name>
```

```
docker commit practical_mclean newimageuser
```

```
docker images
```

You will see the image is created with a name newimageuser.

```
root@ip-172-31-36-185:/home/ec2-user
invoke-rc.d: policy-rc.d denied execution of start.
Processing triggers for libc-bin (2.31-0ubuntu9.1) ...
Processing triggers for ca-certificates (20210119~20.04.1) ...
Updating certificates in /etc/ssl/certs...
0 added, 0 removed; done.
Running hooks in /etc/ca-certificates/update.d...
done.
root@7c45b5f313f1:/# ls
bin  dev  file  home  lib32  libx32  mnt  proc  run  srv  test  tmp  var
boot  etc  file2  lib  lib64  media  opt  root  sbin  sys  test2  usr  xyz
root@7c45b5f313f1:/# cd /var/
root@7c45b5f313f1:/var# ls
backups  cache  lib  local  lock  log  mail  opt  run  spool  tmp  www
root@7c45b5f313f1:/var# cd ..
root@7c45b5f313f1:/# cd tmp
root@7c45b5f313f1:/tmp# touch file1 file2
root@7c45b5f313f1:/tmp# ls
file1  file2
root@7c45b5f313f1:/tmp# cd
root@7c45b5f313f1:/# cd
root@7c45b5f313f1:/# exit
exit
[root@ip-172-31-36-185 ec2-user]# docker images
REPOSITORY          TAG          IMAGE ID          CREATED          SIZE
ubuntu              latest       f63181f19b2f     4 weeks ago     72.9MB
[root@ip-172-31-36-185 ec2-user]# docker ps -a
CONTAINER ID        IMAGE          COMMAND                  CREATED          STATUS
PORTS              NAMES
7c45b5f313f1       ubuntu        "/bin/bash"           7 minutes ago   Exited (0) 16 se
conds ago          practical_mclean
[root@ip-172-31-36-185 ec2-user]# docker commit practical_mclean newimageuser
sha256:2754bbb90f885fc18dd30189b86e0923c70cb29c6708bcf21927dac4abdf8851
[root@ip-172-31-36-185 ec2-user]# docker images
REPOSITORY          TAG          IMAGE ID          CREATED          SIZE
newimageuser        latest       2754bbb90f88      5 seconds ago   213MB
ubuntu              latest       f63181f19b2f     4 weeks ago     72.9MB
[root@ip-172-31-36-185 ec2-user]#
```

```
https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-172-31-36-185 ~]# sudo su
[root@ip-172-31-36-185 ec2-user]# docker images
REPOSITORY          TAG          IMAGE ID          CREATED          SIZE
[root@ip-172-31-36-185 ec2-user]# docker ps -a
CONTAINER ID        IMAGE          COMMAND                  CREATED          STATUS
PORTS              NAMES
[root@ip-172-31-36-185 ec2-user]# docker run -it ubuntu /bin/bash
Unable to find image 'ubuntu:latest' locally
latest: Pulling from library/ubuntu
83ee3a23efb7: Pull complete
db98fc6f11f0: Pull complete
f611acd52c6c: Pull complete
Digest: sha256:703218c0465075f4425e58fac086e09e1de5c340b12976ab9eb8ad26615c3715
Status: Downloaded newer image for ubuntu:latest
root@7c45b5f313f1:/# ls
bin  dev  home  lib32  libx32  mnt  proc  run  srv  tmp  var
boot  etc  lib  lib64  media  opt  root  sbin  sys  usr
root@7c45b5f313f1:/# touch file1 file2 test test2 xyz
root@7c45b5f313f1:/# ls
bin  dev  file  home  lib32  libx32  mnt  proc  run  srv  test  tmp  var
boot  etc  file2  lib  lib64  media  opt  root  sbin  sys  test2  usr  xyz
root@7c45b5f313f1:/# yum install apache2 -y
bash: yum: command not found
root@7c45b5f313f1:/# apt-get install apache2 -y
Reading package lists... Done
Building dependency tree
Reading state information... Done
E: Unable to locate package apache2
root@7c45b5f313f1:/# aptinstall apache2 -y
bash: aptinstall: command not found
root@7c45b5f313f1:/# ap tinstall apache2 -y
bash: ap: command not found
root@7c45b5f313f1:/# apt install apache2 -y
Reading package lists... Done
Building dependency tree
Reading state information... Done
E: Unable to locate package apache2
```

## Step-2: login into the docker hub account

Command:

Docker login

Add credentials when asked carefully.

```
[root@ip-172-31-36-185 ec2-user]# docker login
Login with your Docker ID to push and pull images from Docker Hub. If you don't have a Docker ID
, head over to https://hub.docker.com to create one.
Username: kush8755
Password:
Error response from daemon: Get https://registry-1.docker.io/v2/: unauthorized: incorrect userna
me or password
[root@ip-172-31-36-185 ec2-user]# docker login
Login with your Docker ID to push and pull images from Docker Hub. If you don't have a Docker ID
, head over to https://hub.docker.com to create one.
Username: kush8755
Password:
WARNING! Your password will be stored unencrypted in /root/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store

Login Succeeded
[root@ip-172-31-36-185 ec2-user]#
```

### Step-3:

Now, push it into the docker hub using the following command

Command:

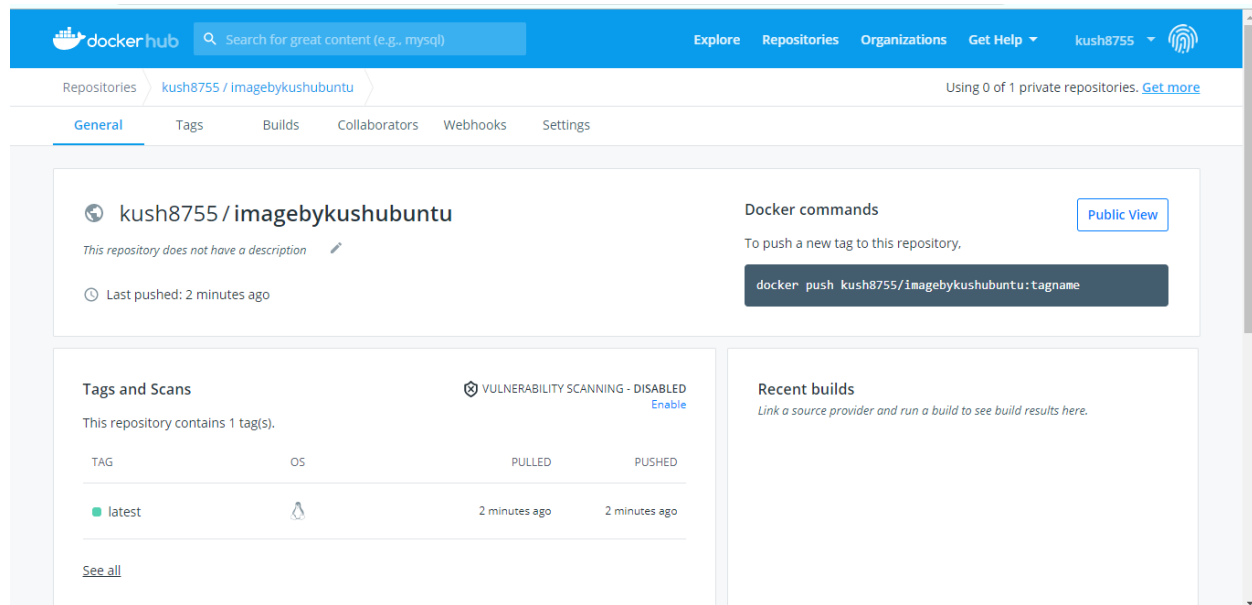
Docker tag <image\_name\_to\_be\_pushed> <New\_i mage\_name>

Docker tag newimageuser kush8755/imagebykushubuntu

Docker push kush8755/imagebykushubuntu

```
https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-172-31-36-185 ~]$ sudo su
[root@ip-172-31-36-185 ec2-user]# docker images
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
newimageuser         latest             2754bbb90f88       14 minutes ago     213MB
ubuntu               latest             f63181f19b2f       4 weeks ago        72.9MB
[root@ip-172-31-36-185 ec2-user]# docker ps -a
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS
7c45b5f313f1       ubuntu             "/bin/bash"        22 minutes ago     Exited (0) 15 mi
nutes ago
practical_mclean
[root@ip-172-31-36-185 ec2-user]# docker tag newimageuser kush8755/imagebykushubuntu
[root@ip-172-31-36-185 ec2-user]# docker tag newimageuser kush8755/imagebykushubuntu
[root@ip-172-31-36-185 ec2-user]# docker tag ubuntu kush8755/imagebykushubuntu
[root@ip-172-31-36-185 ec2-user]# ls
[root@ip-172-31-36-185 ec2-user]# docker login
Authenticating with existing credentials...
WARNING! Your password will be stored unencrypted in /root/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store

Login Succeeded
[root@ip-172-31-36-185 ec2-user]# docker tag newimageuser kush8755/imagebykushubuntu
[root@ip-172-31-36-185 ec2-user]#
[root@ip-172-31-36-185 ec2-user]# docker puch kush8755/imagebykushubuntu
docker: 'puch' is not a docker command.
See 'docker --help'
[root@ip-172-31-36-185 ec2-user]# docker push kush8755/imagebykushubuntu
The push refers to repository [docker.io/kush8755/imagebykushubuntu]
4ec026c52b6d: Pushed
02473afd360b: Mounted from library/ubuntu
dbf2c0f42a39: Mounted from library/ubuntu
9f32931c9d28: Mounted from library/ubuntu
latest: digest: sha256:3a600137ade84cb61f8eecd98460f8df73d3030d59938b4abfa0af371cd9e5 size: 11
55
[root@ip-172-31-36-185 ec2-user]# docker ps -a
```



**Step-4:** To verify the software we installed, The files we created will come pre-install when another user uses our image.

Create an AWS ec2 instance [see the previous blog how to create]

Command-1:

sudo su

Redirect to root directory

Command-2:

Yum update -y

Update the Linux machine

Command-3:

Yum install docker -y

Install docker on Linux machine

Command-4:

Service docker start

Start the Docker Daemon

Command-5:

docker images

Show the image available

Command-6:

docker ps -a

Show containers status

Command-7:

docker pull kush8755/imagebykushubuntu

Pull the image of the name “imagebykushubuntu” from the docker hub of username “kush8755”.

Command-8:

docker images

Shows the image available, now it shows the above image we pull.

Command-9: docker run -it --name kushcontainer kush8755/imagebykushubuntu /bin/bash

Create a container name kushcontainer using the image we pull.

Command-10:

cd tmp

ls

It will verify that folder and software conf files are pre-installed in the new user account

Command-11:

```
cd var/www/
```

ls

It will verify that folder and software conf files are pre-installed in the new user account.

```
root@1a8c157004b6: /var/www
Installed:
  docker.x86_64 0:19.03.13ce-1.amzn2

Dependency Installed:
  containerd.x86_64 0:1.4.1-2.amzn2    libgroup.x86_64 0:0.41-21.amzn2    pigz.x86_64 0:2.3.4-1.amzn2.0.1    runc.x86_64 0:1.0.0-0.1.20200826.gitff819c7.amzn2

Complete!
[root@ip-172-31-35-131 ec2-user]# service docker start
Redirecting to /bin/systemctl start docker.service
[root@ip-172-31-35-131 ec2-user]# docker images
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
[root@ip-172-31-35-131 ec2-user]# docker ps -a
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS              NAMES
[root@ip-172-31-35-131 ec2-user]# docker push kush8755/imagebykushubuntu
The push refers to repository [docker.io/kush8755/imagebykushubuntu]
An image does not exist locally with the tag: kush8755/imagebykushubuntu
[root@ip-172-31-35-131 ec2-user]# docker push kush8755/imagebykushubuntu:tagname
The push refers to repository [docker.io/kush8755/imagebykushubuntu]
An image does not exist locally with the tag: kush8755/imagebykushubuntu
[root@ip-172-31-35-131 ec2-user]# docker pull kush8755/imagebykushubuntu
Using default tag: latest
latest: Pulling from kush8755/imagebykushubuntu
83ee3a23efb7: Full complete
db99f6c6f11f0: Full complete
f611a6d52c6c: Full complete
3d9b46f623e4: Full complete
Digest: sha256:3a600137ade84cb6b1f8eecd98460f8df73d3030d59938b4abfa0af371cd9e5
Status: Downloaded newer image for kush8755/imagebykushubuntu:latest
docker.io/kush8755/imagebykushubuntu:latest
[root@ip-172-31-35-131 ec2-user]# docker images
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
kush8755/imagebykushubuntu  latest             2754bbb90f88        33 minutes ago     213MB
[root@ip-172-31-35-131 ec2-user]# docker run -it --name kushcontainer kush8755/imagebykushubuntu /bin/bash
root@1a8c157004b6:/# ls
bin  boot  dev  etc  file  file2  home  lib  lib32  lib64  libx32  media  mnt  opt  proc  root  run  sbin  srv  sys  test  test2  tmp  usr  var  xyz
root@1a8c157004b6:/# cd tmp
root@1a8c157004b6:/tmp# ls
file1  file2
root@1a8c157004b6:/tmp# cd /var
root@1a8c157004b6:/var# cd www
root@1a8c157004b6:/var/www# ls
html
root@1a8c157004b6:/var/www#
```