Module-6: Docker Networking

Demo Document - 5

edureka!

DEMO-5: Signing Images using Docker Content Trust

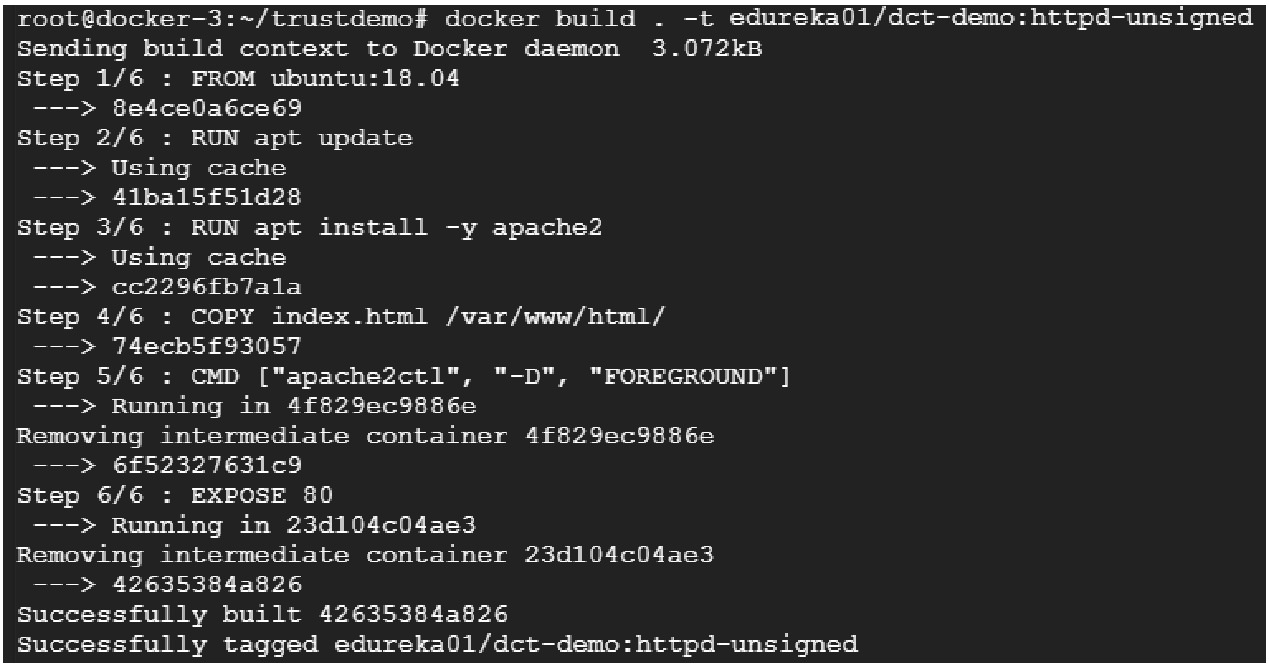
Note: All commands are executed as root.

1. Create a Dockerfile and build the image. Push the image to the repository

```
$ vi Dockerfile

$ docker build . -t <username>/<reponame>:<tagname>

$ docker push <username>/<reponame>:<tagname>
```



```
root@docker-3:~/trustdemo# docker build . -t edureka01/dct-demo:httpd-unsigned
Sending build context to Docker daemon  3.072kB
Step 1/6 : FROM ubuntu:18.04
 ---> 8e4ce0a6ce69
Step 2/6 : RUN apt update
 ---> Using cache
 ---> 41ba15f51d28
Step 3/6 : RUN apt install -y apache2
 ---> Using cache
 ---> cc2296fb7a1a
Step 4/6 : COPY index.html /var/www/html/
 ---> 74ecb5f93057
Step 5/6 : CMD ["apache2ctl", "-D", "FOREGROUND"]
 ---> Running in 4f829ec9886e
Removing intermediate container 4f829ec9886e
 ---> 6f52327631c9
Step 6/6 : EXPOSE 80
 ---> Running in 23d104c04ae3
Removing intermediate container 23d104c04ae3
 ---> 42635384a826
Successfully built 42635384a826
Successfully tagged edureka01/dct-demo:httpd-unsigned
```

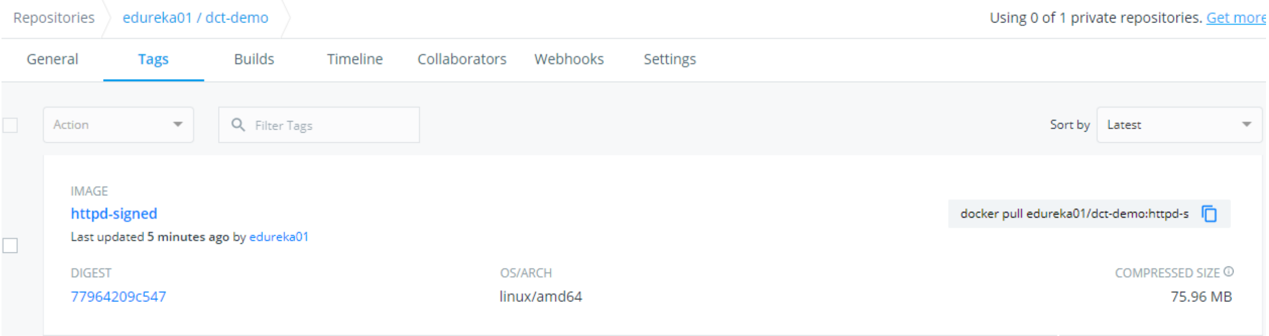


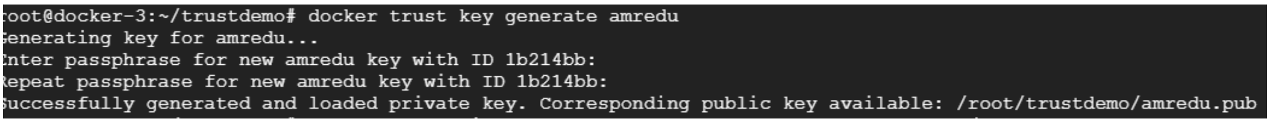
IMAGE
httpd-signed
Last updated 5 minutes ago by edureka01

docker pull edureka01/dct-demo:httpd-s

DIGEST	OS/ARCH	COMPRESSED SIZE
77964209c547	linux/amd64	75.96 MB

2. Now generate a delegation key pair using the generate command

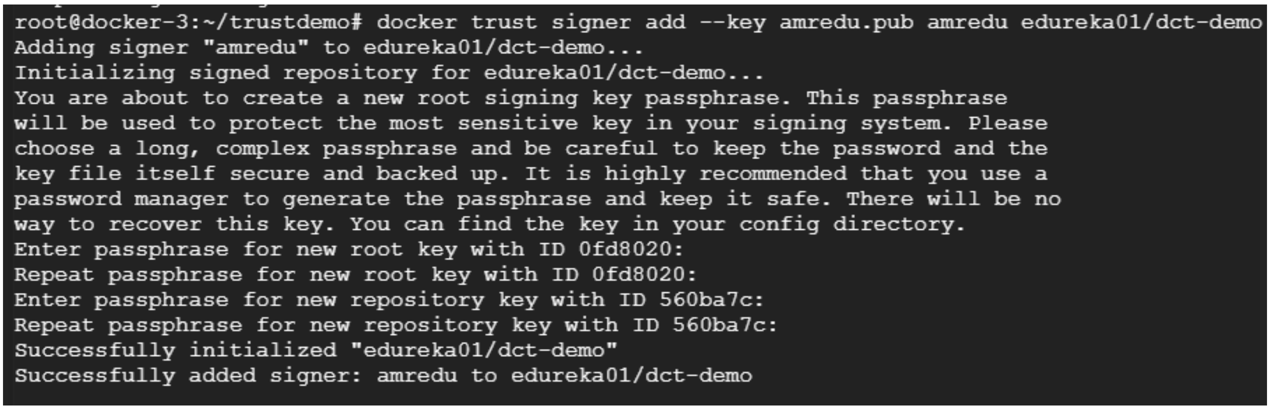
```
$ docker trust key generate <keysigner>
```



```
oot@docker-3:~/trustdemo# docker trust key generate amredu
enerating key for amredu...
nter passphrase for new amredu key with ID 1b214bb:
epeat passphrase for new amredu key with ID 1b214bb:
uccessfully generated and loaded private key. Corresponding public key available: /root/trustdemo/amredu.pub
```

3. Now add the delegation public key to the registry

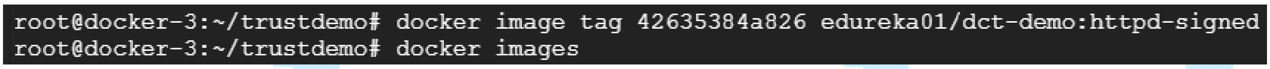
```
$ docker trust signer add --key /<path-to-key> <keysigner> <reponame>
```



```
root@docker-3:~/trustdemo# docker trust signer add --key amredu.pub amredu edureka01/dct-demo
Adding signer "amredu" to edureka01/dct-demo...
Initializing signed repository for edureka01/dct-demo...
You are about to create a new root signing key passphrase. This passphrase
will be used to protect the most sensitive key in your signing system. Please
choose a long, complex passphrase and be careful to keep the password and the
key file itself secure and backed up. It is highly recommended that you use a
password manager to generate the passphrase and keep it safe. There will be no
way to recover this key. You can find the key in your config directory.
Enter passphrase for new root key with ID 0fd8020:
Repeat passphrase for new root key with ID 0fd8020:
Enter passphrase for new repository key with ID 560ba7c:
Repeat passphrase for new repository key with ID 560ba7c:
Successfully initialized "edureka01/dct-demo"
Successfully added signer: amredu to edureka01/dct-demo
```

4. Create a signed tag for the image

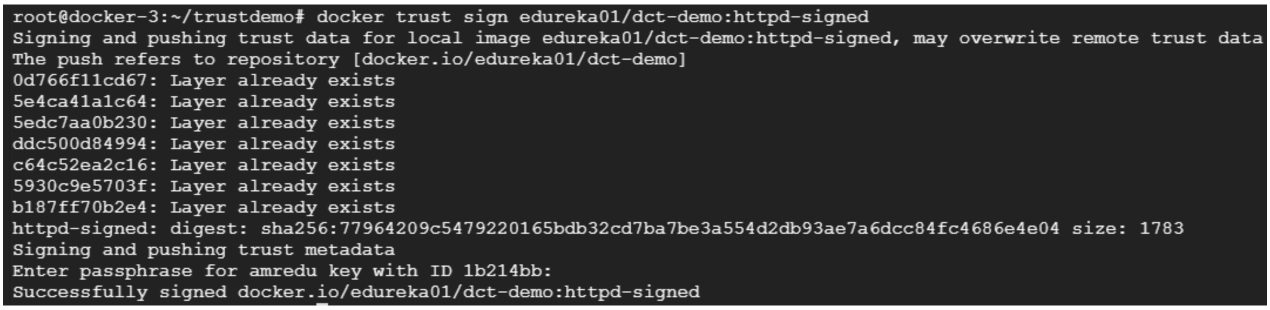
```
$ docker image tag <imageID> <username>/<reponame>:<tagname>
```



```
root@docker-3:~/trustdemo# docker image tag 42635384a826 edureka01/dct-demo:httpd-signed
root@docker-3:~/trustdemo# docker images
```

5. Now, sign the image using:

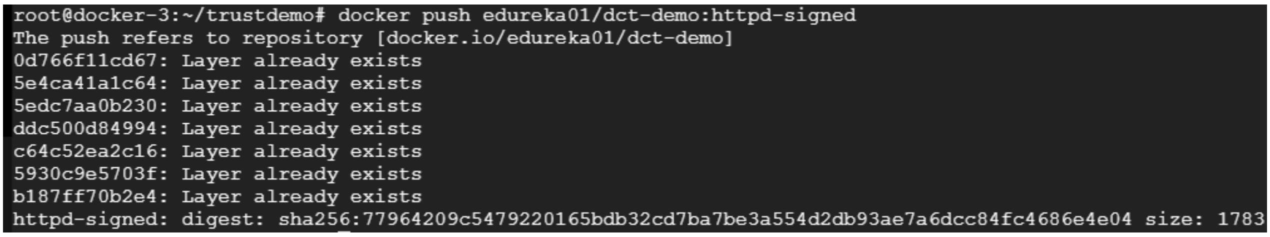
```
$ docker trust sign <username>/<reponame>:<tagname>
```



```
root@docker-3:~/trustdemo# docker trust sign edureka01/dct-demo:httpd-signed
Signing and pushing trust data for local image edureka01/dct-demo:httpd-signed, may overwrite remote trust data
The push refers to repository [docker.io/edureka01/dct-demo]
0d766f11cd67: Layer already exists
5e4ca41a1c64: Layer already exists
5edc7aa0b230: Layer already exists
ddc500d84994: Layer already exists
c64c52ea2c16: Layer already exists
5930c9e5703f: Layer already exists
b187ff70b2e4: Layer already exists
httpd-signed: digest: sha256:77964209c5479220165bdb32cd7ba7be3a554d2db93ae7a6dcc84fc4686e4e04 size: 1783
Signing and pushing trust metadata
Enter passphrase for amredu key with ID 1b214bb:
Successfully signed docker.io/edureka01/dct-demo:httpd-signed
```

6. Push the image to the repository

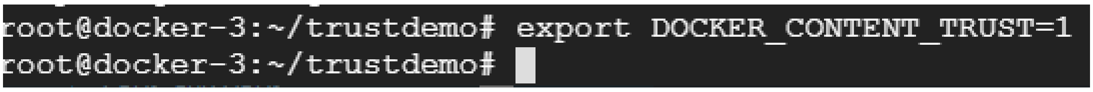
```
$ docker push <username>/<reponame>:<tagname>
```



```
root@docker-3:~/trustdemo# docker push edureka01/dct-demo:httpd-signed
The push refers to repository [docker.io/edureka01/dct-demo]
0d766f11cd67: Layer already exists
5e4ca41a1c64: Layer already exists
5edc7aa0b230: Layer already exists
ddc500d84994: Layer already exists
c64c52ea2c16: Layer already exists
5930c9e5703f: Layer already exists
b187ff70b2e4: Layer already exists
httpd-signed: digest: sha256:77964209c5479220165bdb32cd7ba7be3a554d2db93ae7a6dcc84fc4686e4e04 size: 1783
```

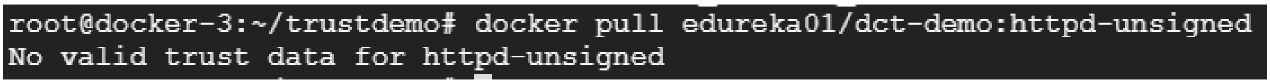
7. Enable DCT on the host for the session by setting the environment variable. This should be done for every new session

```
$ export DOCKER_CONTENT_TRUST=1
```



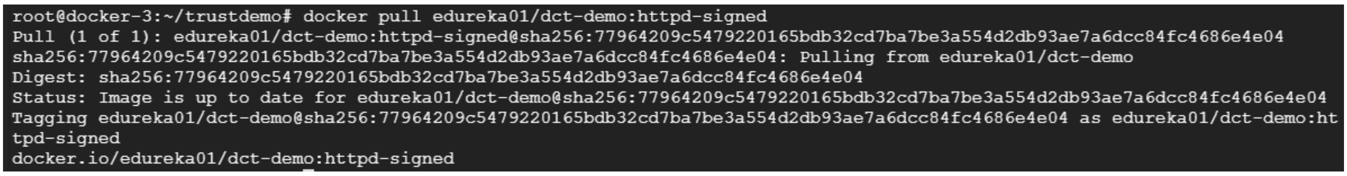
```
root@docker-3:~/trustdemo# export DOCKER_CONTENT_TRUST=1
root@docker-3:~/trustdemo# 
```

8. Now if you try to pull unsigned image from the repo you will get:



```
root@docker-3:~/trustdemo# docker pull edureka01/dct-demo:httpd-unsigned
No valid trust data for httpd-unsigned
```

But you will be able to pull signed images



```
root@docker-3:~/trustdemo# docker pull edureka01/dct-demo:httpd-signed
Pull (1 of 1): edureka01/dct-demo:httpd-signed@sha256:77964209c5479220165bdb32cd7ba7be3a554d2db93ae7a6dcc84fc4686e4e04
sha256:77964209c5479220165bdb32cd7ba7be3a554d2db93ae7a6dcc84fc4686e4e04: Pulling from edureka01/dct-demo
Digest: sha256:77964209c5479220165bdb32cd7ba7be3a554d2db93ae7a6dcc84fc4686e4e04
Status: Image is up to date for edureka01/dct-demo@sha256:77964209c5479220165bdb32cd7ba7be3a554d2db93ae7a6dcc84fc4686e4e04
Tagging edureka01/dct-demo@sha256:77964209c5479220165bdb32cd7ba7be3a554d2db93ae7a6dcc84fc4686e4e04 as edureka01/dct-demo:ht
tpd-signed
docker.io/edureka01/dct-demo:httpd-signed
```