

Module – 3: Docker – I Assignment – 3

You have been asked to:

- Use the saved image in the previous assignment
- Upload this image on Dockerhub
- On a separate machine pull this dockerhub image, and launch it on port 80
- Start the apache2 service
- Verify if you are able to see the apache2 service

```
$ sudo apt remove -y docker docker-engine docker.io containerd runc
$ sudo apt update
$ sudo apt install -y ca-certificates curl gnupg lsb-release
$ sudo apt autoremove -y
$ curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o /usr/share/keyrings/docker-archive-keyring.gpg
$ sudo apt update
$ sudo apt install -y docker-ce docker-ce-cli containerd.io
$ sudo docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
2db29710123e: Pull complete
Digest: sha256:97a379f4f88575512824f3b352bc03cd75e239179eea0fecc38e597b2209f49a
Status: Downloaded newer image for hello-world:latest
Hello from Docker!
This message shows that your installation appears to be working correctly.
```

01

1. Install Docker on Ubuntu using commands as shown

```
(base) hariharan@hariharan-home-pc:~$ sudo docker pull ubuntu:22.04
22.04: Pulling from library/ubuntu
c610536171e3: Pull complete
Digest: sha256:a02c32cf0c2a7e8743c74deef66637aa70e063c9bd40e9e1f8c0b3ea0750b0ba
Status: Downloaded newer image for ubuntu:22.04
docker.io/library/ubuntu:22.04
(base) hariharan@hariharan-home-pc:~$
```

02

03

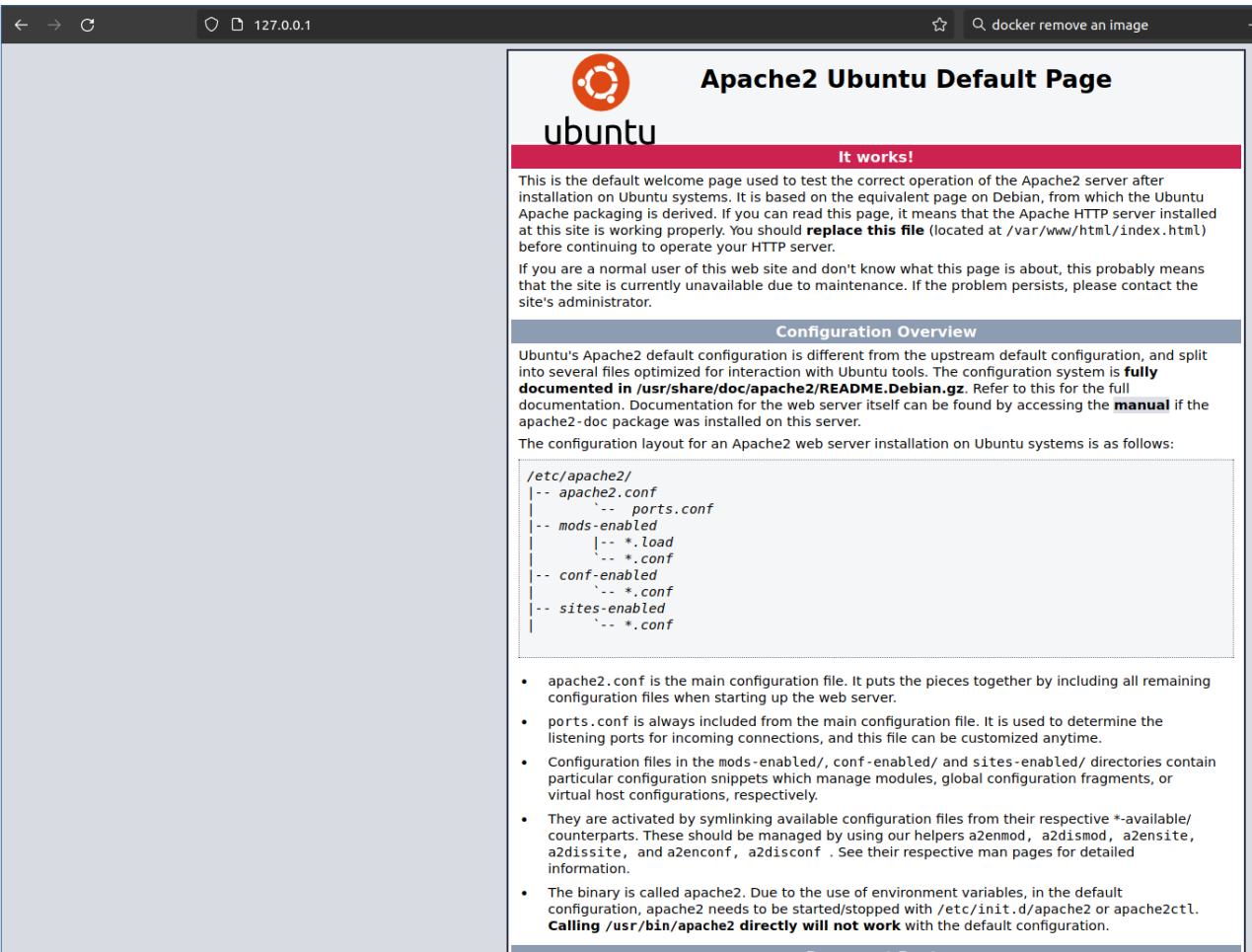
```
(base) hariharan@hariharan-home-pc:~$ sudo docker run --interactive --tty --publish 80:80 --entrypoint "/bin/bash" ubuntu:22.04
root@ef8e900c3636:/#
```

```
root@ef8e900c3636:/# apt update
Get:1 http://security.ubuntu.com/ubuntu jammy-security InRelease [90.7 kB]
Get:2 http://archive.ubuntu.com/ubuntu jammy InRelease [270 kB]
Get:3 http://archive.ubuntu.com/ubuntu jammy-updates InRelease [90.7 kB]
Get:4 http://archive.ubuntu.com/ubuntu jammy-backports InRelease [90.7 kB]
Get:5 http://archive.ubuntu.com/ubuntu jammy/universe amd64 Packages [17.2 MB]
Get:6 http://archive.ubuntu.com/ubuntu jammy/multiverse amd64 Packages [267 kB]
Get:7 http://archive.ubuntu.com/ubuntu jammy/main amd64 Packages [1805 kB]
Get:8 http://archive.ubuntu.com/ubuntu jammy/restricted amd64 Packages [129 kB]
Fetched 19.9 MB in 5s (4060 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
11 packages can be upgraded. Run 'apt list --upgradable' to see them.
root@ef8e900c3636:/# apt install -y apache2
```

04

2. Pull Ubuntu Docker container
3. Run it with port 80 of container mapped to port 80 of local
4. Install apache within the container

```
root@c0303ff86409:/# service apache2 status
* apache2 is not running
root@c0303ff86409:/# service apache2 start
* Starting Apache httpd web server apache2
AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 172.17.0.2. Set the 'ServerName' directive globally to suppress this message
*
root@c0303ff86409:/# service apache2 status
* apache2 is running
```



← → ↺ 127.0.0.1 ☆ 🔍 docker remove an image →

Apache2 Ubuntu Default Page

It works!

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Ubuntu systems. It is based on the equivalent page on Debian, from which the Ubuntu Apache packaging is derived. If you can read this page, it means that the Ubuntu HTTP server installed at this site is working properly. You should **replace this file** (located at `/var/www/html/index.html`) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

Configuration Overview

Ubuntu's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Ubuntu tools. The configuration system is **fully documented in `/usr/share/doc/apache2/README.Debian.gz`**. Refer to this for the full documentation. Documentation for the web server itself can be found by accessing the [manual](#) if the `apache2-doc` package was installed on this server.

The configuration layout for an Apache2 web server installation on Ubuntu systems is as follows:

```
/etc/apache2/
|-- apache2.conf
|   |-- ports.conf
|-- mods-enabled
|   |-- *.load
|   |-- *.conf
|-- conf-enabled
|   |-- *.conf
|-- sites-enabled
|   |-- *.conf
```

- `apache2.conf` is the main configuration file. It puts the pieces together by including all remaining configuration files when starting up the web server.
- `ports.conf` is always included from the main configuration file. It is used to determine the listening ports for incoming connections, and this file can be customized anytime.
- Configuration files in the `mods-enabled/`, `conf-enabled/` and `sites-enabled/` directories contain particular configuration snippets which manage modules, global configuration fragments, or virtual host configurations, respectively.
- They are activated by symlinking available configuration files from their respective `*-available/` counterparts. These should be managed by using our helpers `a2enmod`, `a2dismod`, `a2ensite`, `a2dissite`, and `a2enconf`, `a2disconf`. See their respective man pages for detailed information.
- The binary is called `apache2`. Due to the use of environment variables, in the default configuration, `apache2` needs to be started/stopped with `/etc/init.d/apache2` or `apache2ctl`. **Calling `/usr/bin/apache2` directly will not work** with the default configuration.

05

06

5. Check status of apache2 service and Start it if not already started

6. Now, from out of the container, open <http://127.0.0.1> in a browser. You should see the Apache2 default page.

This confirms that apache2 webserver is...

Running in the container

Publishing to port 80 in the container

Container port 80 is mapped to port 80 on host machine

```
root@85a247223f14:/# exit
exit
```

```
(base) hariharan@hariharan-home-pc:~$ sudo docker ps -l
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
c0303ff86409	ubuntu:22.04	"/bin/bash"	36 minutes ago	Exited (0) 4 seconds ago		vibrant_kare

```
(base) hariharan@hariharan-home-pc:~$ sudo docker commit --message="Updated to install apache2" --author="Hariharan Narayanan" c0303ff86409 ubuntu_with_apache2
sha256:90eba8d2896876d11d4129604149db2d2adad0ff07b877998d2fcd7285089120
```

```
(base) hariharan@hariharan-home-pc:~$ sudo docker run --interactive --tty --publish 81:80 --entrypoint "/bin/bash" ubuntu_with_apache2
[sudo] password for hariharan:
root@032d6d9dcbaa:/#
```

```
root@032d6d9dcbaa:/# service apache2 status
```

```
* apache2 is not running
```

```
root@032d6d9dcbaa:/# service apache2 start
```

```
* Starting Apache httpd web server apache2
```

```
AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 172.17.0.2. Set the 'ServerName' directive globally to suppress this message
```

```
*
```

```
root@032d6d9dcbaa:/# service apache2 status
```

```
* apache2 is running
```

```
root@032d6d9dcbaa:/#
```

```
root@032d6d9dcbaa:/# apt install -y curl
```

```
Reading package lists... Done
```

```
Building dependency tree... Done
```

```
Reading state information... Done
```

```
The following NEW packages will be installed:
```

```
  curl
```

```
0 upgraded, 1 newly installed, 0 to remove and 11 not upgraded.
```

```
Need to get 194 kB of archives.
```

```
After this operation, 452 kB of additional disk space will be used.
```

```
Get:1 http://archive.ubuntu.com/ubuntu jammy/main amd64 curl amd64 7.81.0-1 [194 kB]
```

```
Fetched 194 kB in 1s (164 kB/s)
```

```
debconf: delaying package configuration, since apt-utils is not installed
```

```
Selecting previously unselected package curl.
```

```
(Reading database ... 7838 files and directories currently installed.)
```

```
Preparing to unpack .../curl_7.81.0-1_amd64.deb ...
```

```
Unpacking curl (7.81.0-1) ...
```

```
Setting up curl (7.81.0-1) ...
```

```
root@032d6d9dcbaa:/# curl localhost
```

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
```

07

08

09

10

7. Exit the container, note the container-id, and commit this container with a new name

“ubuntu_with_apache2”

8. Run the new container “ubuntu_with_apache2” with publishing the container’s port 80 as port 81 on the host machine

9. Start apache2 service in the container and verify that it is running

10. Verify that port 80 within container opens the default apache2 webpage



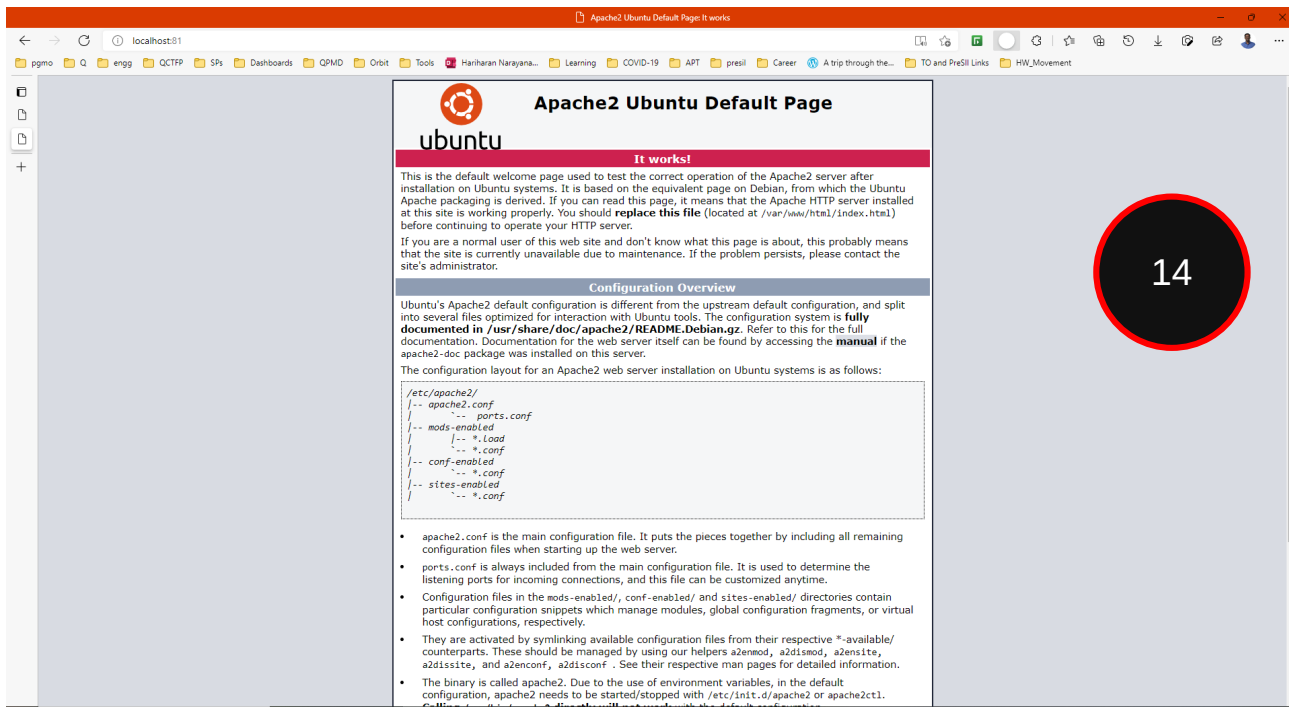
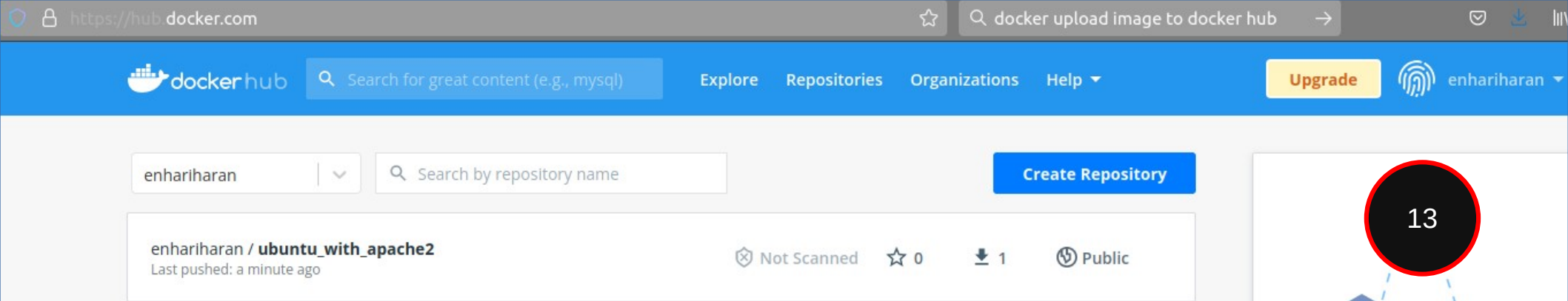
11

```
(base) hariharan@hariharan-home-pc:~$ export DOCKER_ID_USER="enhariharan"
(base) hariharan@hariharan-home-pc:~$ sudo 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
(base) hariharan@hariharan-home-pc:~$ sudo docker ps -l
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS          NAMES
b817296c2b88   ubuntu_with_apache2  "/bin/bash"             10 minutes ago Exited (0) 6 minutes ago           wonderful_boyd
(base) hariharan@hariharan-home-pc:~$ sudo docker tag ubuntu_with_apache2 $DOCKER_ID_USER/ubuntu_with_apache2
(base) hariharan@hariharan-home-pc:~$ sudo docker push enhariharan/ubuntu_with_apache2
Using default tag: latest
The push refers to repository [docker.io/enhariharan/ubuntu_with_apache2]
40745734d150: Pushed
a8f850575cf1: Mounted from library/ubuntu
latest: digest: sha256:78aca28a520a06145eff423a3513861c9c071697c6cbfc6fc3390eaecccc14fa1 size: 741
(base) hariharan@hariharan-home-pc:~$
```

12

11. Open <http://localhost:81> on host machine and verify that the default apache2 web page hosted within the container started in 10 is served in the host machine throughport 81
12. Tag this image, login into dockerhub, and upload this image to dockerhub



13. Login into dockerhub using the browser and verify that image is uploaded correctly

14. I opened my work laptop running Windows 11, downloaded Docker Desktop, logged in into Dockerhub using Docker Desktop, pulled this image in (13) and ran it with port 80 of container mapped to port 81 of host. Finally, opened the browser to <http://localhost:81> to verify that default Apache page is shown