Containers & Docker Security LAB

SUBMITTED TO

Dr. Hitesh Kumar Sharma

SUBMITTED BY

Siddharth Agarwal

500107594

R2142220663

Btech CSE DevOps B1

Lab Exercise 5- Building a Docker Image for an HTML App Using Nginx

1. Setup

You will need:

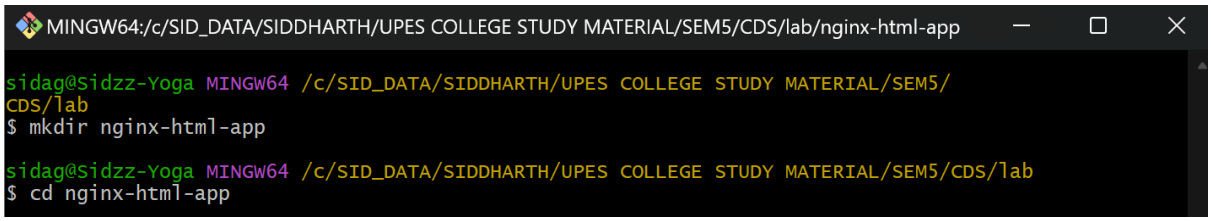
- Docker installed on your machine.
- A simple HTML file for the app.

2. Step 1: Create the HTML File

Create a directory for your HTML app and place an index.html file in it.

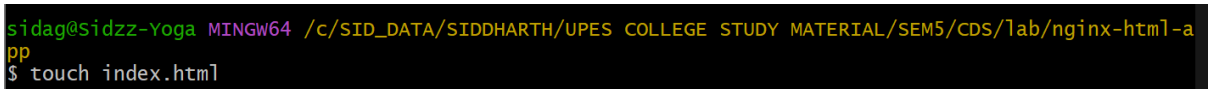
```
mkdir nginx-html-app

cd nginx-html-app
```



Inside the nginx-html-app directory, create the HTML file.

```
touch index.html
```



Edit the index.html file with the following content (or any custom HTML content you want):

```
<!DOCTYPE html>
<html>
<head>
  <title>Welcome to My Nginx HTML App</title>
</head>
<body>
```

```
    <h1>Hello, Nginx Docker!</h1>
    <p>This is a simple HTML app served by Nginx in a Docker container.</p>
</body>
</html>
```

```
sidag@Sidzz-Yoga MINGW64 /c/SID_DATA/SIDDHARTH/UPES COLLEGE STUDY MATERIAL/SEM5/CDS/lab/nginx-html-app
$ cat index.html
<!DOCTYPE html>
<html>
<head>
    <title>Welcome to My Nginx HTML App</title>
</head>
<body>
    <h1>Hello, Nginx Docker!</h1>
    <p>This is a simple HTML app served by Nginx in a Docker container.</p>
</body>
</html>
```

3. Step 2: Create a Dockerfile

In the same directory, create a Dockerfile. This file will define how to build the Docker image using Nginx as the base image.

```
touch Dockerfile
```

```
sidag@Sidzz-Yoga MINGW64 /c/SID_DATA/SIDDHARTH/UPES COLLEGE STUDY MATERIAL/SEM5/CDS/lab/nginx-html-app
$ touch Dockerfile
```

Edit the Dockerfile and add the following content:

```
FROM nginx:latest
COPY index.html /usr/share/nginx/html/
EXPOSE 80
```

```
sidag@Sidzz-Yoga MINGW64 /c/SID_DATA/SIDDHARTH/UPES COLLEGE STUDY MATERIAL/SEM5/CDS/lab/nginx-html-app
$ cat Dockerfile
FROM nginx:latest
COPY index.html /usr/share/nginx/html/
EXPOSE 80
```

4. Step 3: Build the Docker Image

Now that you have the Dockerfile and index.html, it's time to build the Docker image. Run the following command to build the image, giving it a tag (e.g., nginx-html-app):

```
docker build -t nginx-html-app .
```

```
sidag@Sidzz-Yoga MINGW64 /c/SID_DATA/SIDDHARTH/UPES COLLEGE STUDY MATERIAL/SEM5/CDS/la
b/nginx-html-app
$ docker build -t nginx-html-app .
[+] Building 0.3s (7/7) FINISHED                                  docker:desktop-linux
 => [internal] load build definition from Dockerfile                               0.0s
 => => transferring dockerfile: 106B                                               0.0s
 => [internal] load metadata for docker.io/library/nginx:latest                    0.0s
 => [internal] load .dockerignore                                                  0.0s
 => => transferring context: 2B                                                    0.0s
 => [internal] load build context                                                  0.0s
 => => transferring context: 267B                                                  0.0s
 => [1/2] FROM docker.io/library/nginx:latest                                      0.1s
 => [2/2] COPY index.html /usr/share/nginx/html/                                   0.0s
 => exporting to image                                                             0.1s
 => => exporting layers                                                            0.0s
 => => writing image sha256:24d5057379f7a9a94497c0f528a91073b077aecb210109ff1c1    0.0s
 => => naming to docker.io/library/nginx-html-app                                  0.0s

View build details: docker-desktop://dashboard/build/desktop-linux/desktop-linux/2rdsu
z6551ld77h6sjm7km82f

What's next:
    View a summary of image vulnerabilities and recommendations → docker scout quickvi
ew
```

Docker will use the Nginx base image, copy your index.html into the appropriate directory, and build the image.

5. Step 4: Run the Docker Container

After building the image, you can run the container with the following command:

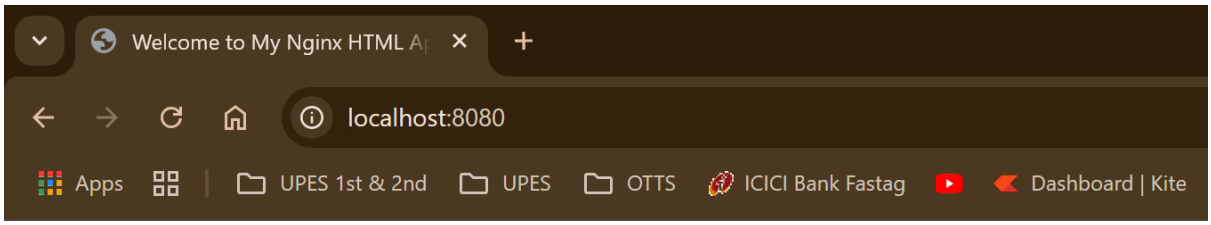
```
docker run -d -p 8080:80 nginx-html-app
```

```
sidag@Sidzz-Yoga MINGW64 /c/SID_DATA/SIDDHARTH/UPES COLLEGE STUDY MATERIAL/SEM5/CDS/la
b/nginx-html-app
$ docker run -d -p 8080:80 nginx-html-app
ae5833b12f296925fb2e46b2cbd7d187ecf57959353bd6f999cd853d57d790a7
```

This command runs the container in detached mode (-d) and maps port 8080 on your host machine to port 80 inside the container, where Nginx is serving your HTML app.

6. Step 5: Verify

Open a browser and go to http://localhost:8080. You should see your HTML page with the message “Hello, Nginx Docker!”.



Hello, Nginx Docker!

This is a simple HTML app served by Nginx in a Docker container.

7. Step 6: Stop and Remove the Container

Once you're done, you can stop and remove the container:

```
docker ps  # to see running containers

docker stop <container-id>

docker rm <container-id>
```

```
sidag@Sidzz-Yoga MINGW64 /c/SID_DATA/SIDDHARTH/UPES COLLEGE STUDY MATERIAL/SEM5/CDS/la
b/nginx-html-app
$ docker ps
CONTAINER ID   IMAGE            COMMAND                  CREATED         STATUS
  PORTS                  NAMES
ae5833b12f29   nginx-html-app   "/docker-entrypoint.…"   2 minutes ago   Up 2 minutes
  0.0.0.0:8080->80/tcp   recursing_roentgen

sidag@Sidzz-Yoga MINGW64 /c/SID_DATA/SIDDHARTH/UPES COLLEGE STUDY MATERIAL/SEM5/CDS/la
b/nginx-html-app
$ docker stop ae5833b12f29
ae5833b12f29

sidag@Sidzz-Yoga MINGW64 /c/SID_DATA/SIDDHARTH/UPES COLLEGE STUDY MATERIAL/SEM5/CDS/lab/nginx-html-a
$ docker rm ae5833b12f29
ae5833b12f29

sidag@Sidzz-Yoga MINGW64 /c/SID_DATA/SIDDHARTH/UPES COLLEGE STUDY MATERIAL/SEM5/CDS/lab/nginx-html-a
pp
$ docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES

sidag@Sidzz-Yoga MINGW64 /c/SID_DATA/SIDDHARTH/UPES COLLEGE STUDY MATERIAL/SEM5/CDS/lab/nginx-html-a
pp
$
```