Dark Web Course - Student Guide

Antonio Brandao

November 2024

DISCLAIMER:
This Course aims to inform participants about the Dark Web without promoting or facilitating illegal activities. All activities are conducted with a focus on legality and ethics. Participants are reminded to adhere to all applicable laws and regulations. The course facilitators are not responsible for misusing the information provided during the workshop.

Contents

1 Student Guide: Deploying a Flask Application Using Docker

1.1 Introduction

This guide provides a step-by-step walkthrough for deploying a docker container with a Linux machine.
It is divided into two parts:

- Part 1: Setting up the environment and deploying on Linux.
- Part 2: Setting up the environment and deploying on Windows using Docker Desktop.

2 Part 1: Deployment on Linux

Prerequisites

- A Linux server with root access.
- Basic knowledge of Linux command-line operations.
- Internet connectivity for installing packages.

2.1 Step-by-Step Linux Instructions

2.1.1 Update Package Lists and Install *tmux*

As the **root** user, update the package lists and install ***tmux***:

bash

```
apt update
apt install tmux
```

2.1.2 Start and Manage *tmux* Sessions

tmux helps manage long-running processes and maintains sessions even after disconnection.
Start a new ***tmux*** session:

bash

```
tmux
```

List all active ***tmux*** sessions:

bash

```
tmux ls
```

Attach to an existing ***tmux*** session:

bash

```
tmux a
```

2.1.3 Add New Users

Create a new user flask:

bash

```
adduser docker
```

Add the users to the sudo group:

bash

```
usermod -aG sudo docker
```

Switch to the new users:

bash

```
su - docker
```

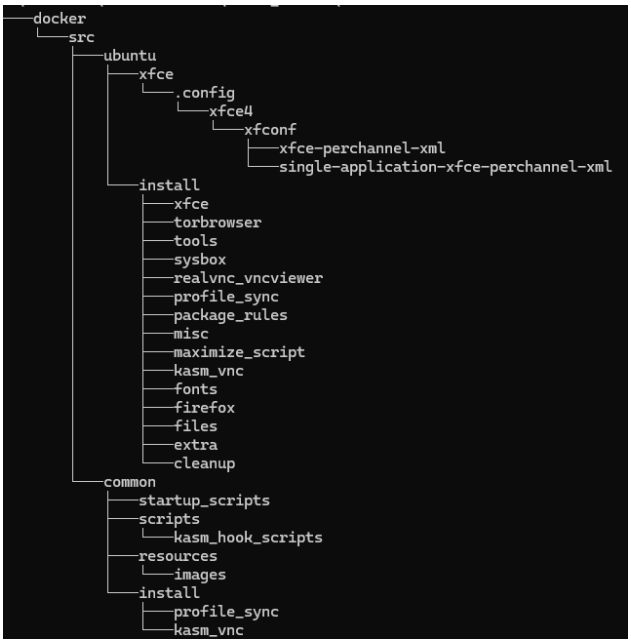
Verify user identity:

bash

```
id
```

2.1.4 Copy the "docker" Folder to the Server

Copy your local docker folder to the server using scp or any file transfer method.
Directories structure:



bash

```
sudo chown -R docker:docker /home/docker/
```

Make the docker.sh script executable and run it
The docker.sh script typically installs Docker and its dependencies.

bash

```
sudo chmod +x docker.sh
./docker.sh
```

2.1.5 Exit flask User and Set Ownership

Exit to return to the previous user:

bash

```
exit
su - docker
```

2.1.6 Build the Docker Image

Build the Docker image with a specific name and tag:

bash

```
cd /home/docker/
docker-compose build
```

2.1.7 Run the container

bash

```
docker-compose up -d
```

2.1.8 Access the container

Access the container by visiting the following address in a web browser.

Browser

```
http://IP_server:6080
```

2.1.9 Managing Containers

Stop the container

bash

```
docker stop $(docker ps -a)
```

Delete the container

bash

```
docker rm $(docker ps -a)
```

Delete the image

bash

```
docker rmi $(docker images)
```

3 Part 2: Deployment on Windows Using Docker Desktop

Prerequisites

- Windows 10 or later.
- Docker Desktop installed. Download from Docker Desktop for Windows.
- Python 3 installed. Download from Python Releases for Windows.

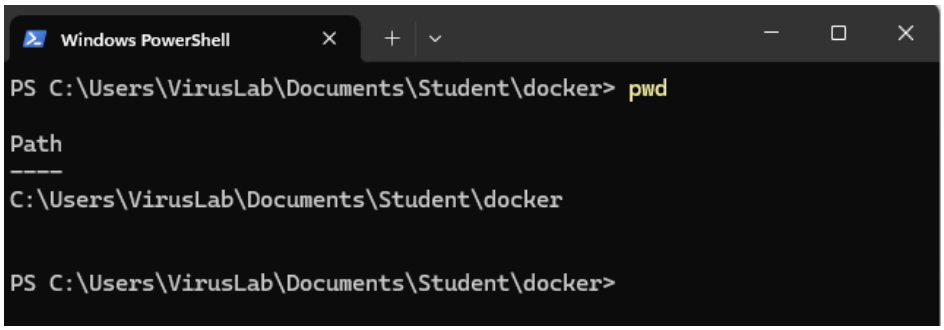
3.1 Step-by-Step Windows Instructions

3.1.1 Install Docker Desktop, Git, and Python

- **Install Docker Desktop:** Download and install Docker Desktop from the official website (https://www.docker.com/products/docker-desktop)
- **Install Python:** Download and install Python 3 (https://www.python.org/downloads/windows/)
- **Install Git for Windows:** Need to clone repositories (https://git-scm.com/download/win)

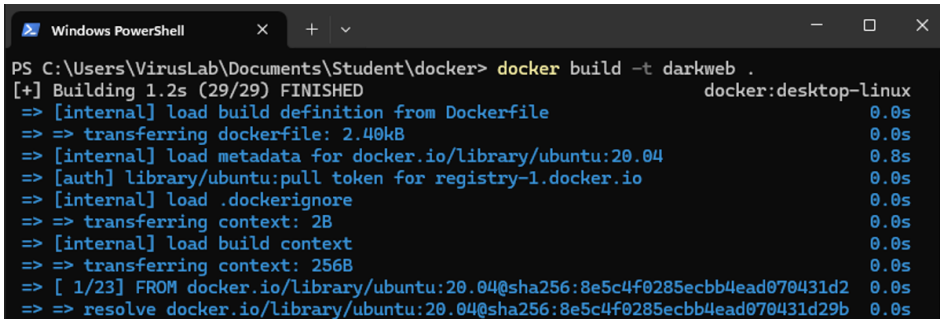
3.1.2 Build the Docker Image using Terminal

Open Command Prompt or PowerShell and navigate to the docker directory. And do the command to build the image.

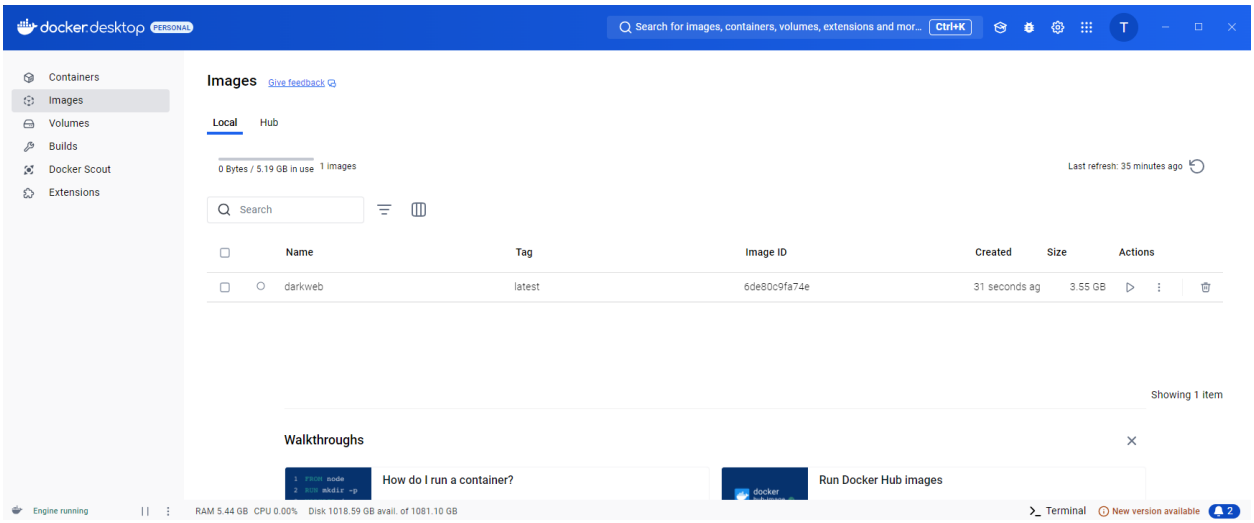


bash

```
docker build -t darkweb .
```



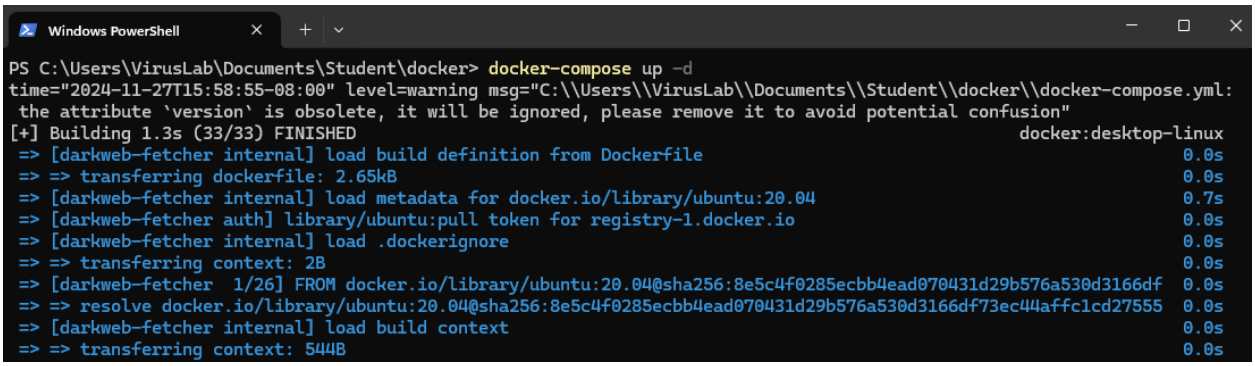
3.1.3 Open Docker Desktop to manage the Image



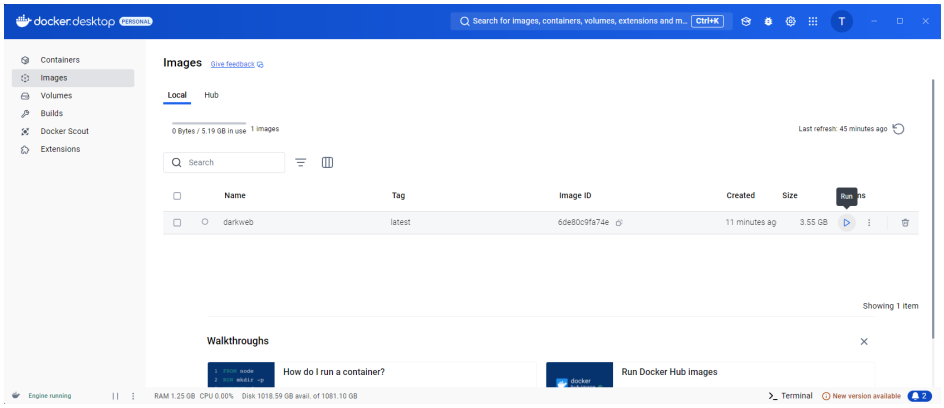
3.2 Run the Docker Container

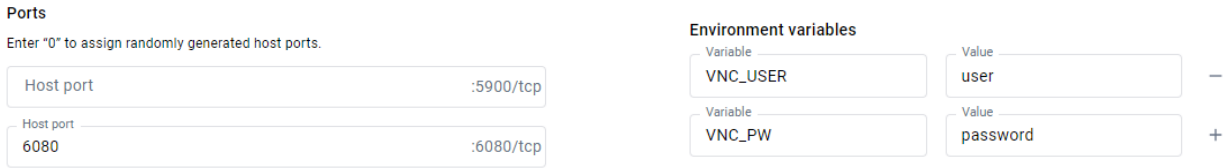
- Using the Terminal:

```bash
docker-compose up
```

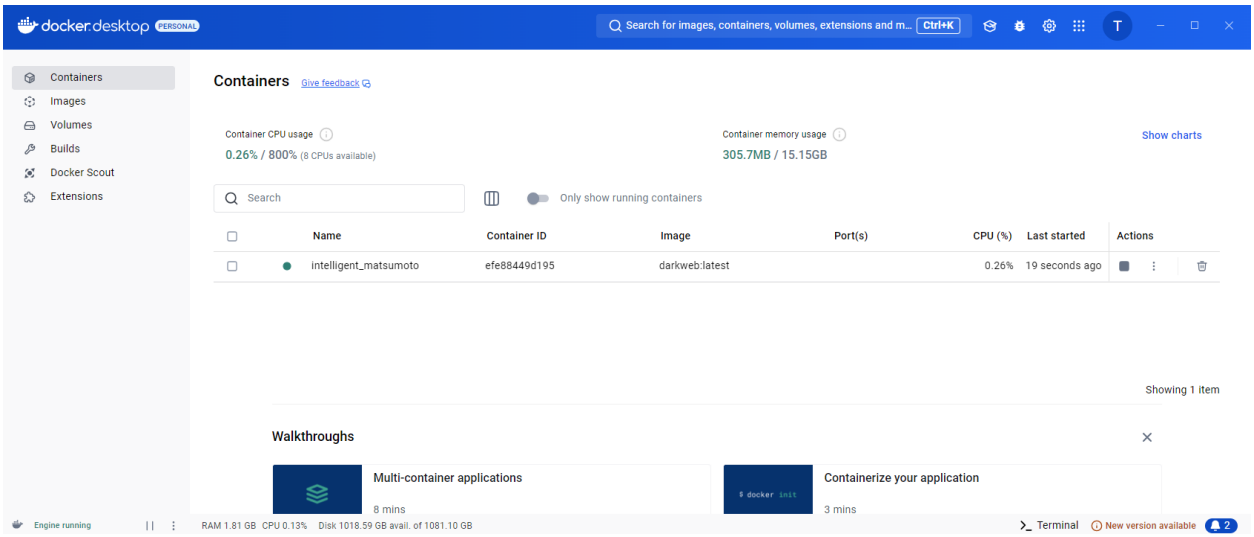


- **Using Docker Desktop:** Go to ”Images” on the left side. You can find the image you just created. Click on ”Run”. Expand the ”Optional settings” and add ”6080” to ”Ports” in ”Environment variables” add ”VNC_USER” in variables and ”user” as a Value, do again by clicking on the plus icon ”+” and add uma more, ”VNC_PW” and ”password” and, finally, click on ”Run”





- **Checking the Container running:** Go to ”Containers” on the left side. You can find the container information in this panel



4 Accessing the Container

4.1 Using the Browser (Chrome) to access the container

Open a web browser and navigate to:

Browser

```
https://localhost:6080
```

NOTE: Because you are using a Docker container on your computer, the https with a self-signed certificate will show you an alert. The alert is related to the self-signed certificate, and there is no problem with the container.
Follow the step to access the container (Chrome Browser):

- In the alert "Your connection is not private" click on "Advanced"
- The click on "Proceed to localhost (unsafe)"
- Now you have the access to the container