OWASP Lab

This lab introduces students to the concept of a CTF through the form of web vulnerabilities. Most of the vulnerabilities come from the OWASP top 10. OWASP, which stands for Open Web Application Security Project, is an project dedicated to making the web more secure. There are 7 vulnerabilities in this lab based on the **OWASP top 10**. Furthermore vulnerabilities like the ones in this lab have all been found in major production websites at some point.

Setup

As seen in previous labs, this lab uses **docker**. The following steps will have the lab up and running in docker in no time.

1. In terminal/docker toolbox navigate to the directory containing `lab4.tar`.
2. Run the command `docker load < lab4.tar` to load the image into docker.
3. Run the command `docker-compose up -d` (note this must be run from the directory containing `docker-compose.yml`)
4. The lab should now be accessible from **http://localhost:1337/**

If you are unable to access the lab you may be running an older version of docker which runs in a VM with a separate IP. To get that ip run the command `docker-machine ip` and then connect to **http://<ip here>:1337/**.

When you're done the lab you will need to kill the docker containers. This can be done using: `docker compose down`. **Note:** This will wipe the database. If you don't want to wipe the database you can use `docker stop lab4_web lab4_db`.

WARNING the following commands will remove everything you have in docker. Finally when you're done with everything you can use `docker system prune -a` followed by `docker volume prune` to completely wipe the docker environment. Alternatively you can remove the images for the lab manually.

Questions

Each of the challenges in this lab have a flag associated with them. A flag is of the format flag{some_text_here}. The idea comes from the concept of a cyber security CTF in which participants hunt for vulnerabilities revealing flags. These vulnerabilities typically reflect real world security issues.

1. Register an account, you may need to use a SQL injection to help you out here. You will find a flag once your SQL injection is successful.
2. Create a link that redirects the user to **w**hen they log in. There is no flag for this challenge, just record the URL that you used.
3. Find a stored cross site scripting vulnerability, have an alert pop up on the dashboard page. Once successful a second alert will pop up with the flag.
4. A piece of information is exposed on the employee dashboard. Find this and a flag is with it.
5. Based on the piece of information exposed in question 4 are you able to find a file on the system which shouldn't be accessible?
6. So in question 5 you've found some data that shouldn't be there. See if you can figure out Fred's password. This is another form of sensitive data exposure.
7. **This one is optional and quite challenging!** Find a way to change the admin account's password, you won't be able to crack the hash... there might be another way through the messaging system. After all his assistant does automatically read the messages.

Report

For your report be sure to include screen shots of each of the flag you have found. Furthermore write a small description detailing how you were able to find that flag as well as a recommendation for the administrator on what could be done to prevent the vulnerability.