B207 Cyber Security

Passman – Your fav pw manager

Ahmed Khaled Ebrahim Mohammed

GH1029750

Prof. Sami Salamin

GitHub page:

https://github.com/Ahmed-Mohammed-x/Passman---Your-password-hero

Guide:

https://www.youtube.com/watch?v=L_l7eYsvcZw

Introduction:

This is a website which is a password manager called Passman. He was made by me -Ahmed Mohammed- as a submission to the final assessment. The task was to implement a password manager that:

- Can authenticate users and secure passwords, encrypt them, then store them in the database
- Has Password generation for the users
- Password retrieval and management by using a web interface (flask or Django)

This implementation should satisfy the requirements in the assessment brief. Below is a guide and an introduction on how to clone the files in the repository!

The Guide:

Above is the requested video that shows a demo as well as how to use the program while also highlighting different features. (if the video or link is not working, please contact me, I can provide proof that the video was submitted before the deadline)

In Summary, you clone the repository using PyCharm and then import all necessary imports. Configure the database to sql, browse passwords.db in the same main root and click apply and okay. Right click any other issues and it will auto fix.

Technical details and Choice design:

HTML

First for the layout and color, while tinkering, I liked the black and white theme as it shows a minimal and a modern design. Other html codes were made purely through learning what each command do and connecting all urls and other redirections to the right routes.

PYTHON

While the approach of dividing the files into different files according to specific functions and routes is a good idea to maintain the code, making it all in one file was easier for me as I didn't have any prior experience with flask or implementing a python code of this level. It was fun and I believe in the future, opting for the formal implementation may be the way to go as it is clearer once you understand how everything connects.

We mainly need flask for the web implementation as well as cryptography and werkzeug to help us encrypt, hash, and debugging the code.
and so Passman successfully achieve the following:

Password Encryption and hashing: The passwords of users are encrypted and hashed using the cryptography and the werkzeug library.

Key derivation: PBDKF2HMAC with SHA-256 helps derive encryption keys and protect against brute force attacks.

Database: We used sqlite and we have 2 tables Passwords and Users

Security features: Passman has protection against XSS, brute force, session management, and strong password generator

Please note: Emoji texts were used as it is my way of typing :) I understand that in production there will change to the standard messages.

FUTURE PLANS

- HTML files should be optimised as rn it feels a bit messy and more comments should be added to explain why were certain choices made despite better possible ways to achieve he same or even some cases better results.
- The Password checker function has already been added to the python app but to make the user use it, it should be called and displayed in the website.

Passman at your service ^_^