End Module Assignment: Passwordless Authentication

A **Passwordless** authentication module which stands from a cyber security perspective of eliminating passwords from the authentication structure. The project is structured to serve up a Django based webserver which when browsed, presents a login page.

Brief Working of Python TOTP

Time-based OTPs depends on the device as opposed to sim-carriers or email accounts that may be accessible by unauthorised parties. They can function without an internet - connected device. They are automatically updated every 30 seconds and unlike conventional OTPs, they do not need servers to compute and store OTPs every time they are updated. Once the client and server exchange a secret key. The secret key and the current timestamp are used to produce a one-time password at regular intervals. As both the client and server possess the secret key and are time-synchronized, the client and the server may independently create OTPs. The client creates OTP at regular intervals of 30 seconds, but the server generates OTP only when it is necessary to validate the supplied OTP.

Usage

Since this is a Proof-Of-Concept [POC] code and not a fully functional application;

- Add dummy secret to google authenticator generated in the `db_connect` section, using the [+] icon in the app.
- Supply username as test1@ema.com or test2@ema.com
- Submit the `OTP` generated on Google Authenticator.
- Press authenticate

Note: ID & Secret for accounts already in the codio environment

| ID | Secret | | ---------- | ---------- | | test1@ema.com | WOG6WDMHK5OWBKDP | | test2@ema.com | LH4C4QDEZTPVLQY3 |

Setup and Install Guide

To setup the EMA project created please run through the following steps; [You can also copy paste the commands as it is for simplicity]

Copy the `EMA-Passwordless_Auth.zip` to a Ubuntu-system for installment:

`Unzip` the archive using:

```
unzip EMA-Passwordless_Auth.zip
```

`Move` into the newly EMA-Passwordless_Auth directory:

```
cd EMA-Passwordless_Auth/
```

`Install` dependancies using requirements.txt:

```
sudo pip3 install -r ./mysite/requirements.txt
```

Run the `db_install.sh` script to install postgres to the system:

```
sudo ./db_install.sh
```

Run the `db_setup.sh` script to setup the database in postgres:

```
sudo ./db_setup.sh
```

Initialize `Migration` Data:

```
python ./mysite/manage.py makemigrations
```

`Migrate` final settings

```
python ./mysite/manage.py migrate
```

`db_connect` populates the database with dummy auth data and keys [currently existing in the codio workspace]:

```
python3 ./mysite/db_connect.py
```

Start serving the site using `runserver` :

```
python ./mysite/manage.py runserver
```

For Codio you can configure the .codio file as follows Run:

```
python3 /home/codio/workspace/EMA-Passwordless_Auth/mysite/manage.py runserver
0.0.0.0:8000
```

Box URL:

```
http://{{domain8000}}/
```

Please Note:

Some things have been left untouched as this project is to showcase the applicability, feasability, and advantages of a passwordless authentication mechanism and is not meant to function or be impletemented into a `"production"` state of affairs. The items left to insecure values, insecure configurations or missing functionalities are as follows.

- Database is set with a non-secure simple password as `password`
- Django hosts the application by default at `[*]`
- The current implementation does not allow creation of users outside of `db_connect.py`
- Secret generated for Authenticator is not converted and served as a QR code.
- Generated secret length is short to keep the manual efforts minimal. [Making it Insecure by nature]

References

- Edureka Community. (2020). How to configure where to redirect after a log out in Django. Available from: https://www.edureka.co/community/81335/how-to-configure-where-to-redirect-after-a-log-out-in-django [Accessed 19 May 2022].
- Ellingwood, J. and Juell, K. (2018). How To Install the Django Web Framework on Ubuntu 18.04 | DigitalOcean. www.digitalocean.com. Available from:

https://www.digitalocean.com/community/tutorials/how-to-install-the-django-web-framework-on-ubuntu-18-04 [Accessed 19 May 2022].

- Levinas, M. (2021). How to Install and Setup PostgreSQL server on Ubuntu 20.04. Cherry Servers. Available from: https://www.cherryservers.com/blog/how-to-install-and-setup-postgresql-server-on-ubuntu-20-04 [Accessed 19 May 2022].
- Stack Overflow. (2011a). authentication - How to configure where to redirect after a log out in Django? Available from: https://stackoverflow.com/questions/5315100/how-to-configure-where-to-redirect-after-a-log-out-in-django [Accessed 20 May 2022].
- Stack Overflow. (2011b). javascript - Disable browser 'Back' button after logout? Available from: https://stackoverflow.com/questions/6923027/disable-browser-back-button-after-logout [Accessed 19 May 2022].
- Yash (2020). Understanding TOTP : In Python. Analytics Vidhya. Available from: https://medium.com/analytics-vidhya/understanding-totp-in-python-bbe994606087 [Accessed 19 May 2022].