Report Lab 2 SEC

Author

```
Alban Favre
```

Default account

The code comes with a pre-filled database.
The default user is `fav@alb.an` and it's password is `90p5saLU+`.
It's qr code is

```
https://chart.googleapis.com/chart?
chs=200x200&chld=M|0&cht=qr&chl=otpauth%3A%2F%2Ftotp%2Fsecure%2520auth%2520dot%
2520com%3Fsecret%3DGYHVYI7YY5T4OC4FPQA7M6UXUCUMRSBD%26issuer%3D2FA
```

Those default values are not mandatory, and completely deleting the fake database file won't be a problem.

Design

Database

The database will be simulated with multiple .txt files, notably `cooldatabase.txt`, which stores passwords hash (salt is in password hash) and user-names, `tokendatabase.txt` which stores tokens.

If those files don't already exist, there should be no problems, and the app will just create them.
Deleting them while using the app will probably have a lot of unexpected behavior, don't do that.

Email

The email box will be simulated in `coolmailbox.txt`. It should be verified for the registration process and the password forgotten process.
The QR code link for Google authenticator will also be sent there.
Each new mail erase the last one. But the application is linear and that shouldn't be a problem.

Registration

To register the user needs to ask a registration token, which will be sent via mail.
On the email they can find the next steps.
If the email address is already in use, the mail will say so.
A token is only valid 15 minutes from its creation date.

Once the token entered, the user will be asked to create a password.
Once the process of registration is done, an email will be sent to the user, the email contains the link to the mandatory qr code.

If the registration process fails, the token will have been consumed, and the user will need to ask a new one.
But this will only happen if there are unexpected things happening in the database, which should never happen.

Login

Login works as expected, even if the code is pure spaghetti.
Depending on the `is_2fa_active` bool stored in db. The 2fa token might or not be asked

Forgotten password

Ask for a password reset token that will be sent by mail, if an incorrect email is given, the token will still be created and stored, but won't be sent.
Once the token entered, the user can change their password. This password follow the same rule as normal password.
if 2fa is active, a Google authenticator token will be asked first.

If the pass reset process fails, the token will have been consumed, and the user will need to ask a new one.
But this will only happen if there are unexpected things happening in the database, which should never happen.

Changing 2fa state

This operation will always ask for the same thing as login would. Even if the user is already logged in.
2fa state can be changed as much as the user likes, but the associated qr code will never change.

Passwords

The password strength is verified with `zxcvbn` as `passablepasswords` is deprecated.
The password hash and salt are done with sodium oxyde default primitives. Salt is stored in with the hash.

Token

Token are generated with 256 random bytes from sodium oxyde methods, it might be too much.
But I heard about birthday paradoxes, and I can only picture birthday clowns that are unbounded by time or space, and that scares me.

2FA

2FA is done with Google authenticator, a fake mail with the link to the qr code is sent in `coolmailbox.txt`.
To deactivate 2FA for testing, it can be directly changed in `cooldatabase.txt`, change `true` to `false` for `is_2fa_active`
the secret are kept in clear in database.

The app lets a user deactivate the 2fa verification for itself

Tests

I'm sorry but I cannot do them, there are other lab where I'm late and my code is spaghetti.