Advanced Programming with Python

Authorization

Pepe García jgarciah@faculty.ie.edu

2020-04-20

Plan for today

- Review last day's homework

Plan for today

- Review last day's homework
- How do we perform authorization

Homework

Let's review last days' homework

Sessions

There's a feature of most web applications that we haven't yet discussed, sessions.

Session in the web allow to create sections of websites that are private, for which users need to authenticate in order to access.

Sessions & cookies

Sessions are hold by making HTTP use a special kind of header called cookie.

Cookies are headers that the server sends alongside the HTTP response, that the clien **will send back** in subsequent requests!

Sessions & cookies

Whiteboard

Let's whiteboard the whole flow of cookies and sessions

Using sessions in flask

sessions in flask are handled by importing the session object from flask. We can imagine the session as a dictionary to which we can add arbitrary data.

```python
from flask import session


@app.route("/")
def index():
    if "user_id" in session: # session behaves like a dictionary
        return render_template("index.html")
    else:
        return render_template("login.html")
```

Using sessions in flask

Full login example

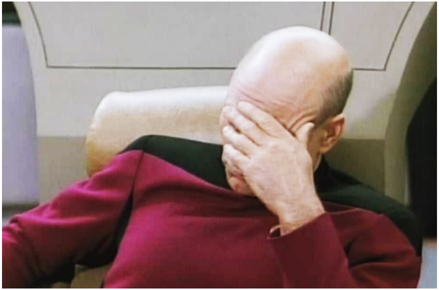
Let's go through a full login example in **`example_session.py`**

Passwords...

There's something **very wrong** about the database we're creating for paymepal...

Passwords...

There's something **very wrong** about the database we're creating for paymepal...



We're storing **PLAIN TEXT PASSWORDS**. That means that anyone with access to the DB can see the passwords right away.

Introducing hash functions

Hash functions are functions that can map arbitrarily sized data to fixed size values.

There are lots of different hash functions, but the ones we'll care about will:

Introducing hash functions

Hash functions are functions that can map arbitrarily sized data to fixed size values.

There are lots of different hash functions, but the ones we'll care about will:

- be **deterministic**

https://en.wikipedia.org/wiki/Hash_function

Introducing hash functions

Hash functions are functions that can map arbitrarily sized data to fixed size values.

There are lots of different hash functions, but the ones we'll care about will:

- be **deterministic**
- make it impossible to guess the **plaintext** given the **hash text**

https://en.wikipedia.org/wiki/Hash_function

Introducing hash functions

Hash functions are functions that can map arbitrarily sized data to fixed size values.

There are lots of different hash functions, but the ones we'll care about will:

- be **deterministic**
- make it impossible to guess the **plaintext** given the **hash text**
- avoid **collisions**

https://en.wikipedia.org/wiki/Hash_function

hash functions

Hash functions will convert **cleartext** to **hashtext**.

hash functions

Hash functions will convert **cleartext** to **hashtext**.

cleartext	hashtext
p4ssw0rd	df984bd56ad2a0df3863b6a0f5230baf520e2b24

hash functions

Hash functions will convert **cleartext** to **hashtext**.

cleartext	hashtext
p4ssw0rd	df984bd56ad2a0df3863b6a0f5230baf520e2b24
pepegar	5e1249bc5af93d7be8cb9c574bdf5b08e42ebba6

Back to passwords

The approach we'll follow to securely store passwords in our database is that we will **store their hash text** instead of their **clear text**.

Then, when checking if a user has a specific password, we'll **compare the hashed values**.

Back to passwords

The approach we'll follow to securely store passwords in our database is that we will **store their hash text** instead of their **clear text**.

Then, when checking if a user has a specific password, we'll **compare the hashed values**.

We'll use the functions `generate_password_hash` and `check_password_hash` from the `werkzeug.security` module.

Practice

Example

see `paymepal_hashed_passwords.db` and `example_hashed_passwords.py`.

Exercise

Creating shop pages

Create a new route that shows information about shops. It can receive the shop id in the url.

Exercise

Creating shop pages

Create a new route that shows information about shops. It can receive the shop id in the url.

Showing shop transactions for shop owners

make it possible that, when looking at a shop page, shop owners can see the transactions in that shop.