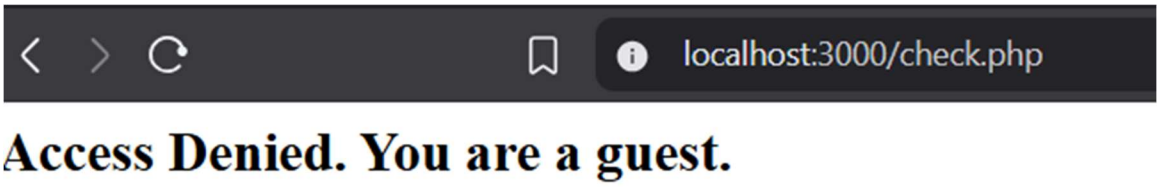
ACM

Week 3 writeup:

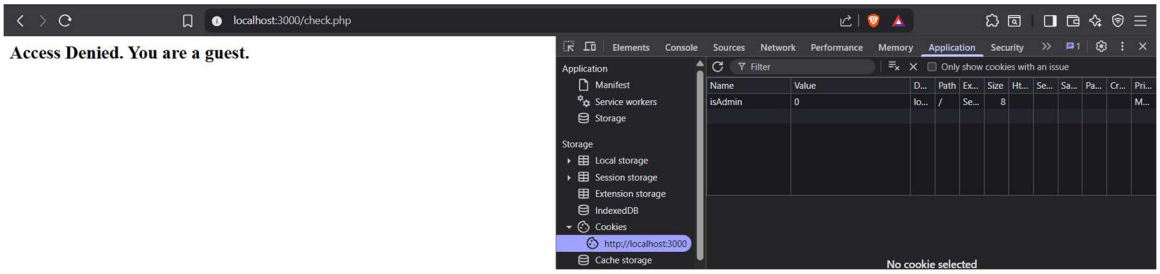
NAME: B RAJMOHITH REDDY
BRANCH: CYBER SECURITY
SIG: CYBER
TASKS WEEK 3

TASK 1:

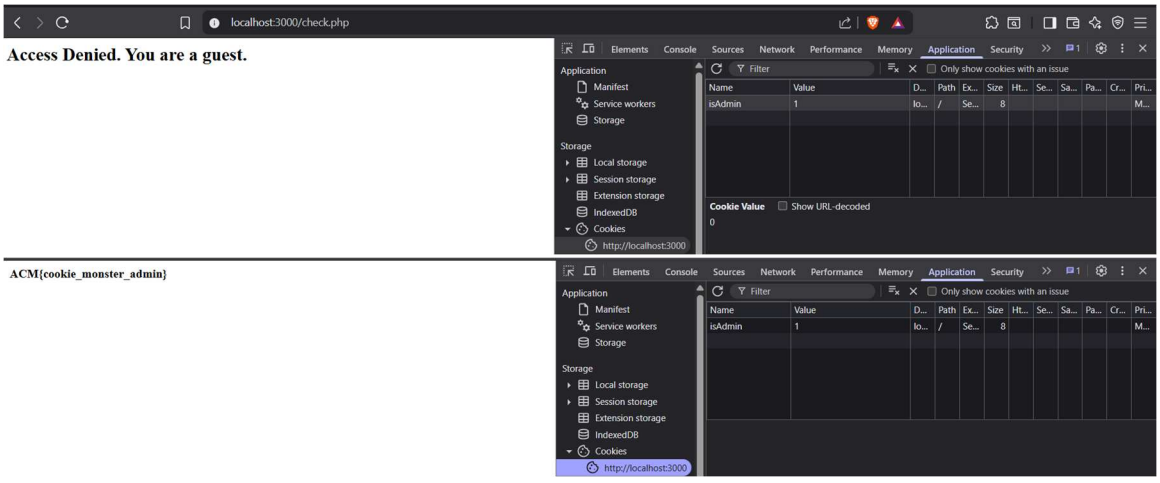
i) I have entered into to http://localhost:3000.



ii) Then on inspecting the page I found that the value of cookie for localhost is 0:



iii) Then I tried it to change it to 1 and reloaded the page so that I found the flag:



So the flag is : **ACM{COOKIE_MONSTER_ADMIN}**

Tech explanation:

iv) This is a cookie-based access control vulnerability. So in this challenge the thing is the serve gets the value of cookie as 0 which means the server decides that the user is a guest and if we change that to 1 to make the server to fell that the user is the admin and show the data. Hence, we find the data.

v) It is also some sort of cookie manipulation or tampering.

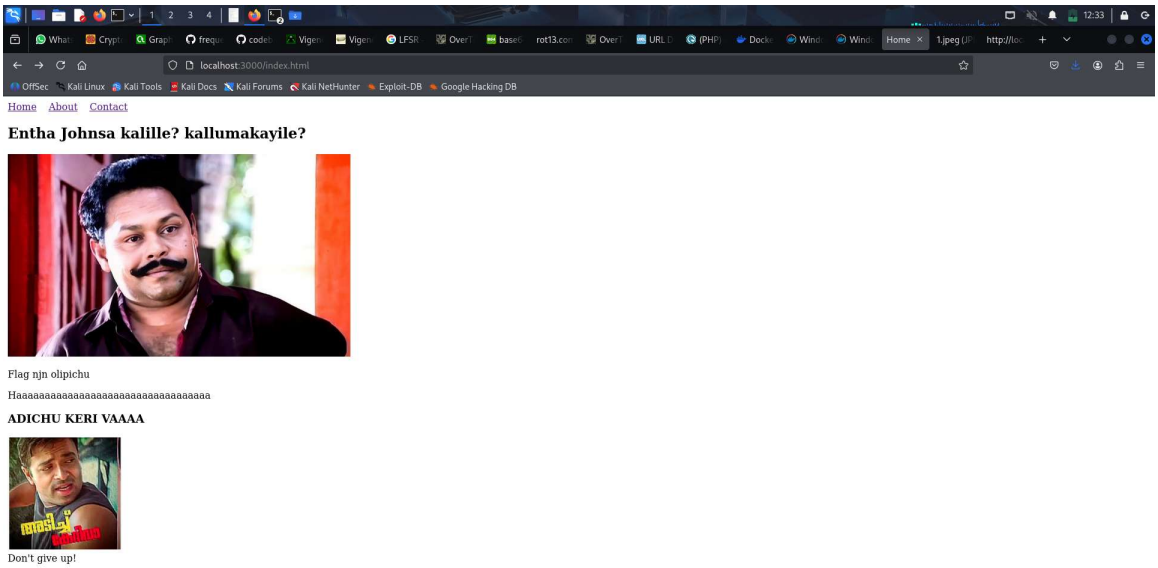
TASK 2:

i) This is a base64 flag that is found in about.html page.

ii) On inspecting the about.html page we will find a base64 encoded data that can be decoded using an online base64 decoder.

iii) The flag is “RmxhZ3tpbnNwZWN0X2ZsYWdfY2hhbmdlfQ==”.

Upon opening the local host I found this and tried to find the flag by saving the pictures that does not work.



iv) so I tried to find the hidden directories in this page using “dirb”

```
┌──(mohith㉿kali)-[~]
└─$ dirb http://localhost:3000

-----------------
DIRB v2.22
By The Dark Raver
-----------------

START_TIME: Sun Aug  3 12:35:18 2025
URL_BASE: http://localhost:3000/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt

-----------------

GENERATED WORDS: 4612

---- Scanning URL: http://localhost:3000/ ----
+ http://localhost:3000/index.html (CODE:200|SIZE:1390)
+ http://localhost:3000/readme (CODE:200|SIZE:64)

-----------------
END_TIME: Sun Aug  3 12:35:24 2025
DOWNLOADED: 4612 - FOUND: 2
```

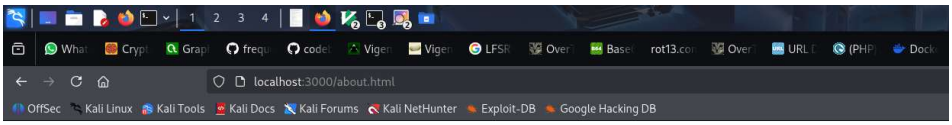
v) here I found that there is a readme thing so I tried going to I using http://localhost:3000/readme.

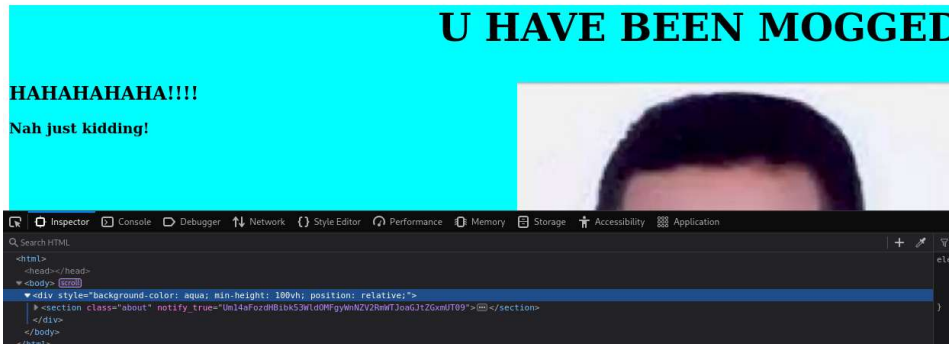
```
File  Edit  Tools  Syntax  Buffers  Window  Help

main index.html page

in about page
inspect > base64 coded flag
~
~
```

vi) this gave me an idea to get the base64 encoded data as below by navigating to about page and inspecting as below.





Tech explanation:

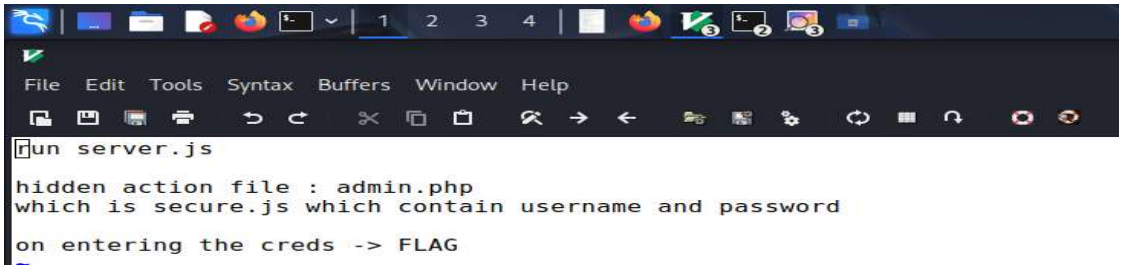
vi) This technique is a encoded data hiding in the coded scripts and the flag is hidden most commonly base64, ROT13, hex and URL encoded form.

vii) This is just the analysis of the web page scripts.

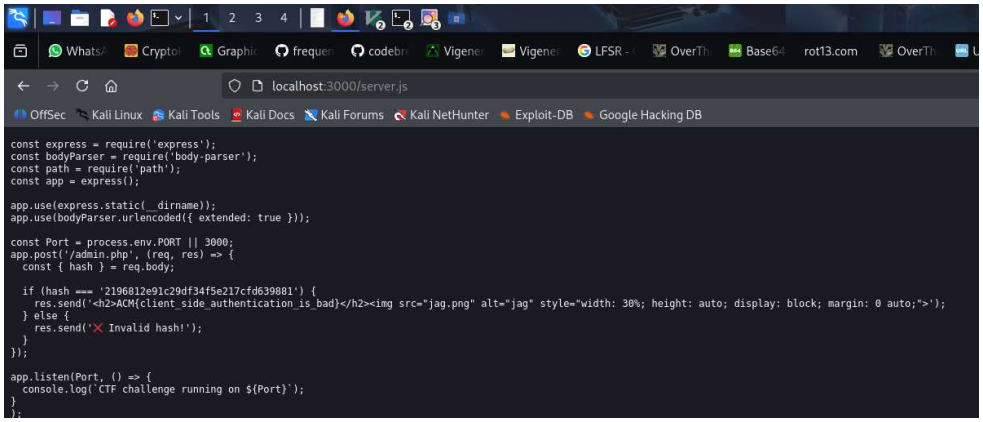
TASK 3:

i) I tried to find the hints in this. So, I tried to open “readme” and I used http://localhost:3000/readme.

ii) Then I found the following details

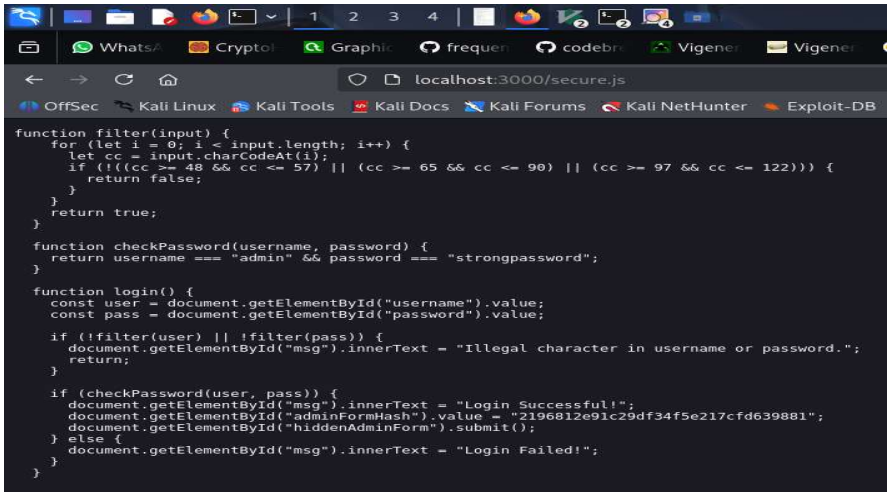


iii) so I modified the url to http://localhost:3000/server.js then that has the following data.



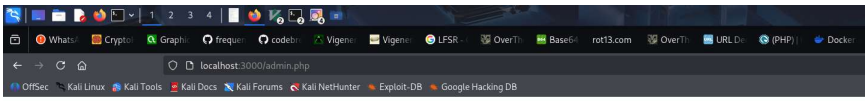
But this confused me so I re-checked the data given in readme again so I found that we can also try http://localhost:3000/secure.js.

iv) In secure.js I found the following data



V) In that I found some data like “username === "admin" && password === "strongpassword";

vi) So I tried that credentials and found the following



ACM{client_side_authentication_is_bad}

vii) The flag is **“ACM {client_side_authentication_is_bad}”.**

Tech explanation:

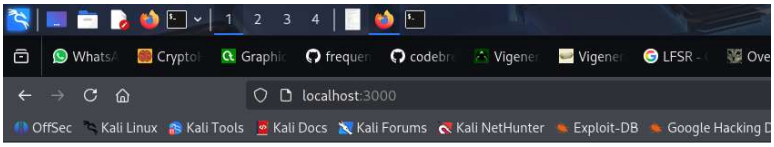
i) This is a kind of static code analysis

ii) This is a kind of static code analysis and authentication bypass with disclosure credentials.

iii) Even this is some sort of credentials harvesting where we will find the credentials in .js scripts.

Task 4:

i) In this task we will find a login page like shown below

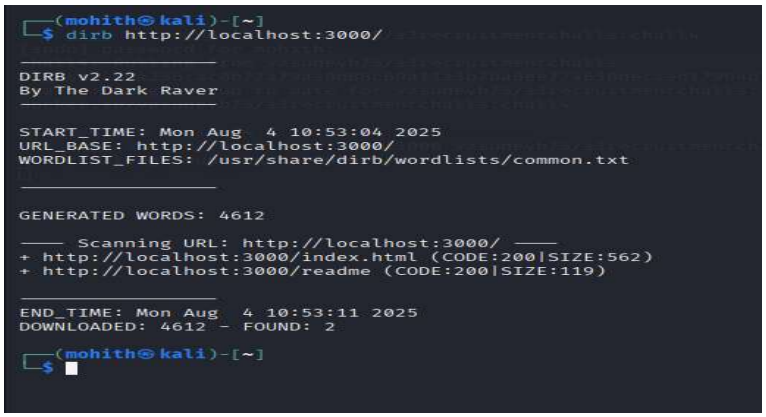


Login

Username Password Submit

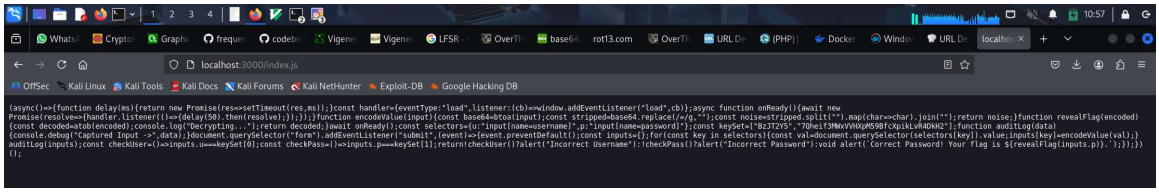
Use Coupon Code: 58

ii) To find the hidden files/directories in this we will use “dirb”



iii) Tried to open readme file and found the following details. These details helped me to find the flag.

IV) Upon navigating to index.js I found some information shown in the picture.



v) This gave me a key set and that are the login credentials but that data is base 58 encoded form.

vi) The login credentials are : “username: admin” , “password: FLAG {base64_is_outdated}”.

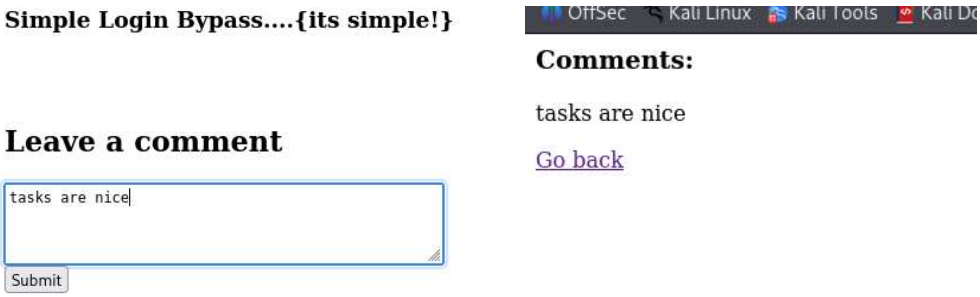
vii) And here that is mentioned as that the “flag is password” so the required flag is “FLAG {base64_is_outdated}”.

Tech explanation:

viii) This is some kind of web exploitation or a type of information disclosure where we will find the required things in .js file which is hidden.

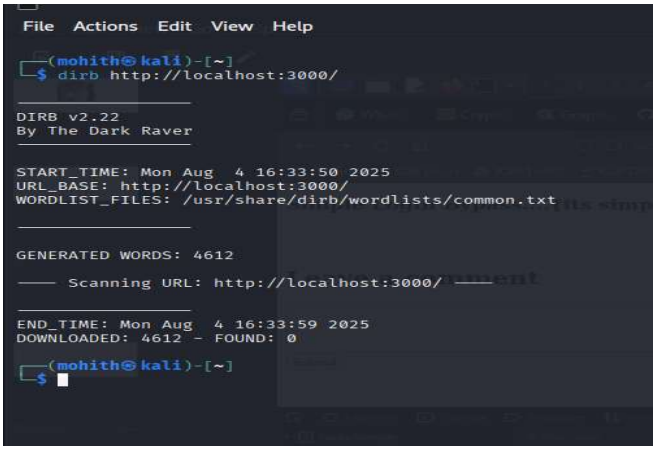
Task 5:

i) Upon opening this I found something like leave a comment box and tried to give some input to find any clue.



ii) I have tried to find the hidden files/directories.

iii) I used “dirb” to find those files but found no details. That just ran a simple search and gave the following output.



iv) Then I tried to inspect the page and analyzed the cookie data Which has the flag data encoded in URL form as following “FLAG%7Bstolen_cookie_flag%7D”

v) So, the decoded data is “FLAG {stolen_cookie_flag}”.

Tech explanation:

i) This is a cookie manipulation or cookie tampering or also some sort of cookie-based access control which hides the flag in cookie data.