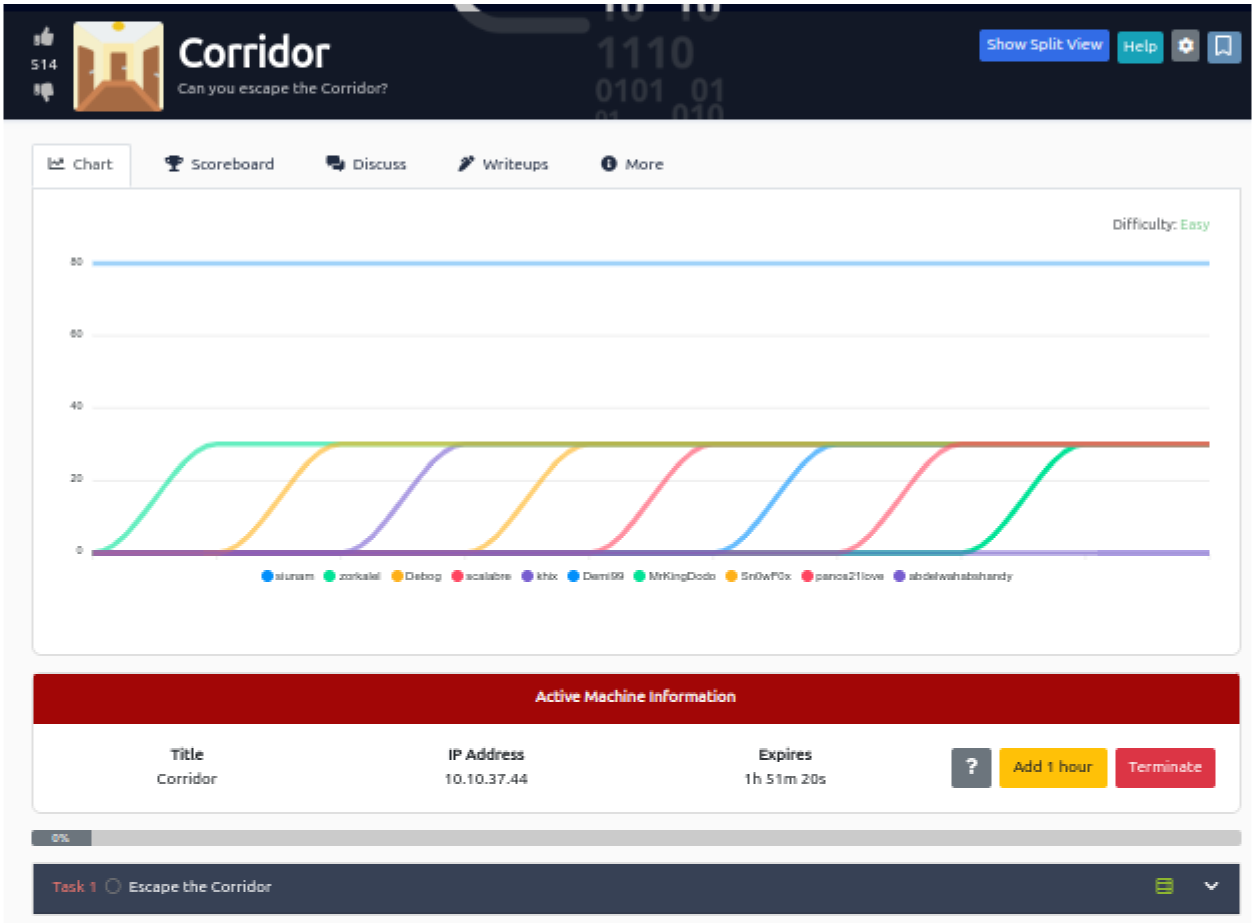


Active Machine Information

Title	IP Address	Expires
Corridor	10.10.37.44	1h 51m 20s

Task 1 ○ Escape the Corridor

Corridor

Escape the Corridor :

You have found yourself in a strange corridor. Can you find your way back to where you came?

In this challenge, you will explore potential IDOR vulnerabilities. Examine the URL endpoints you access as you navigate the website and note the hexadecimal values you find (they look an awful lot like a *hash*, don't they?). This could help you uncover website locations you were not expected to access.

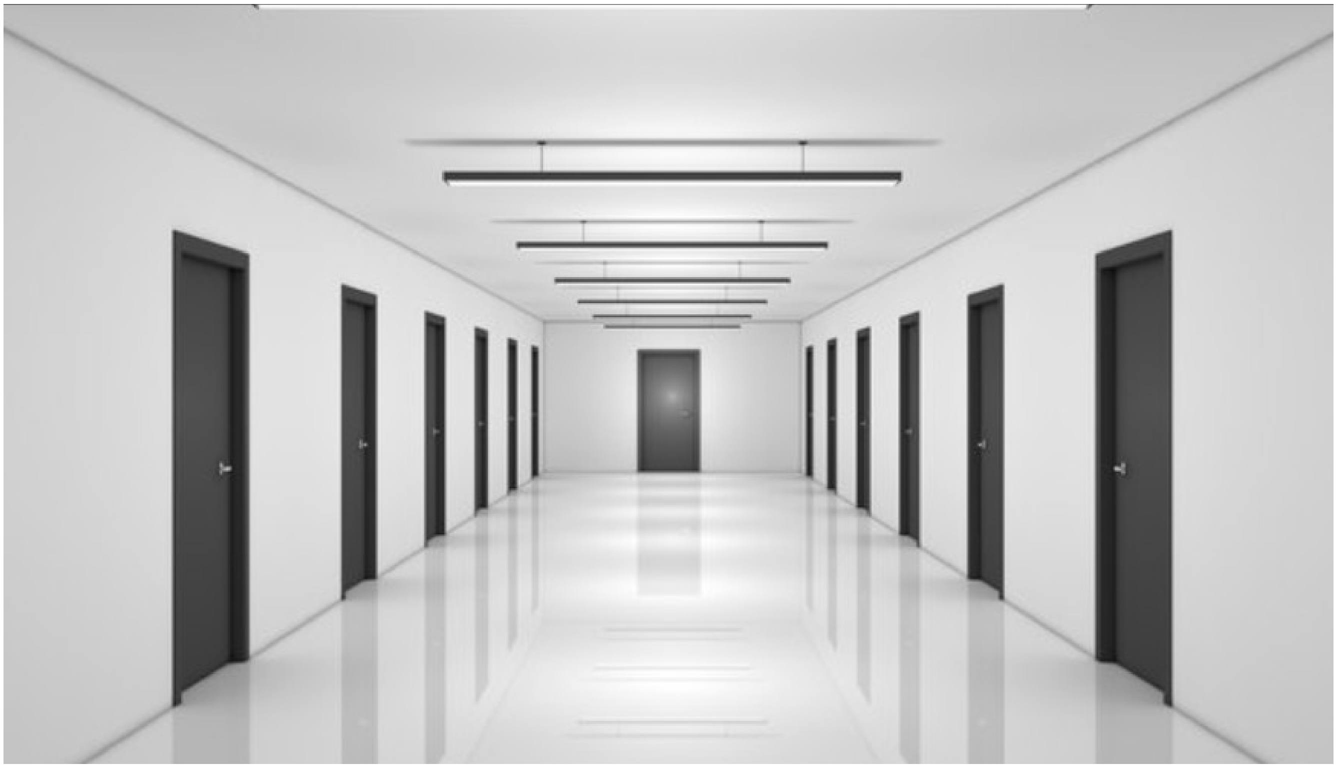
What is the flag?

Flag{################################}

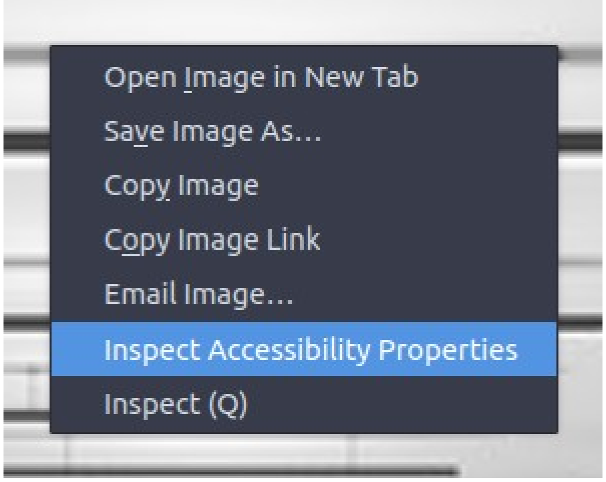
Insecure Direct Object References (IDOR) is a type of security vulnerability that occurs when an application allows an attacker to access a protected resource by modifying the value of a parameter that references the resource. For example, an IDOR vulnerability could allow an attacker to view the profile of another user by changing the user_id parameter in a URL.

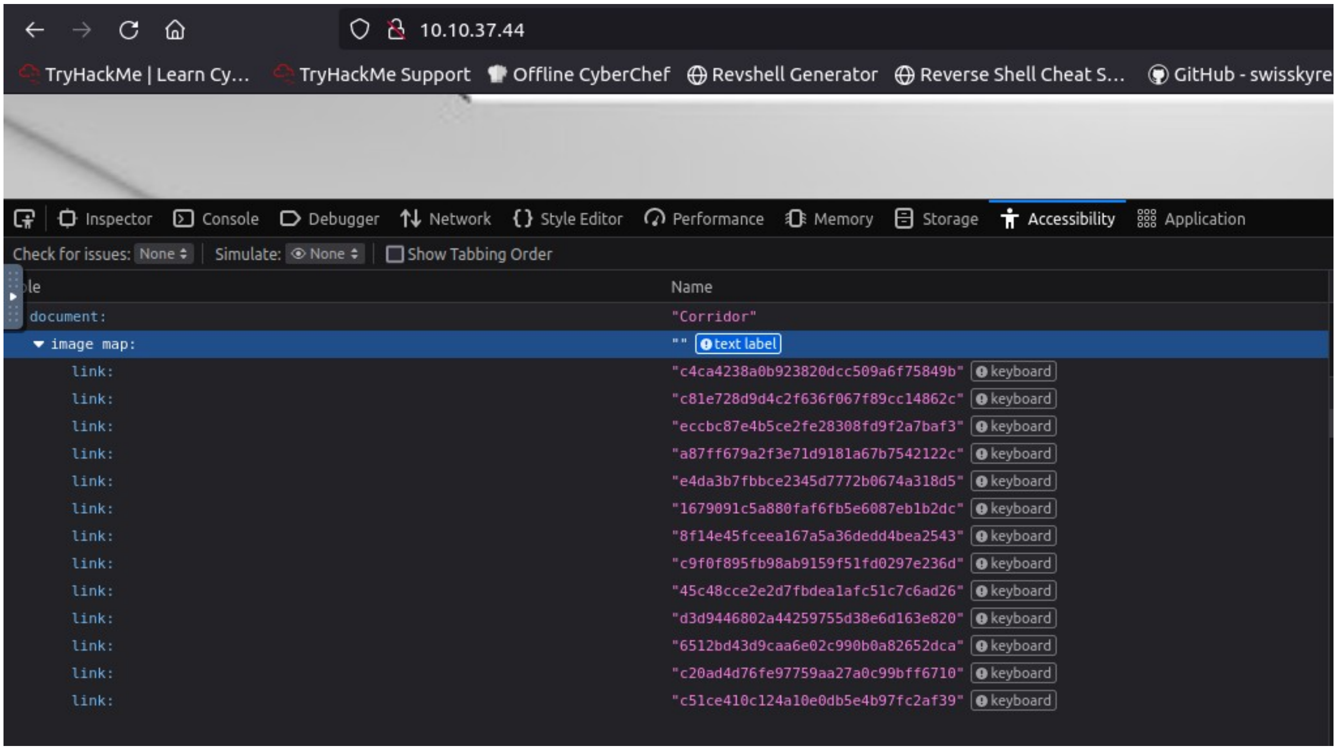
-First, we connected the machine and obtained the server's IP .

-Then I put the IP on the browser, and this was the result :



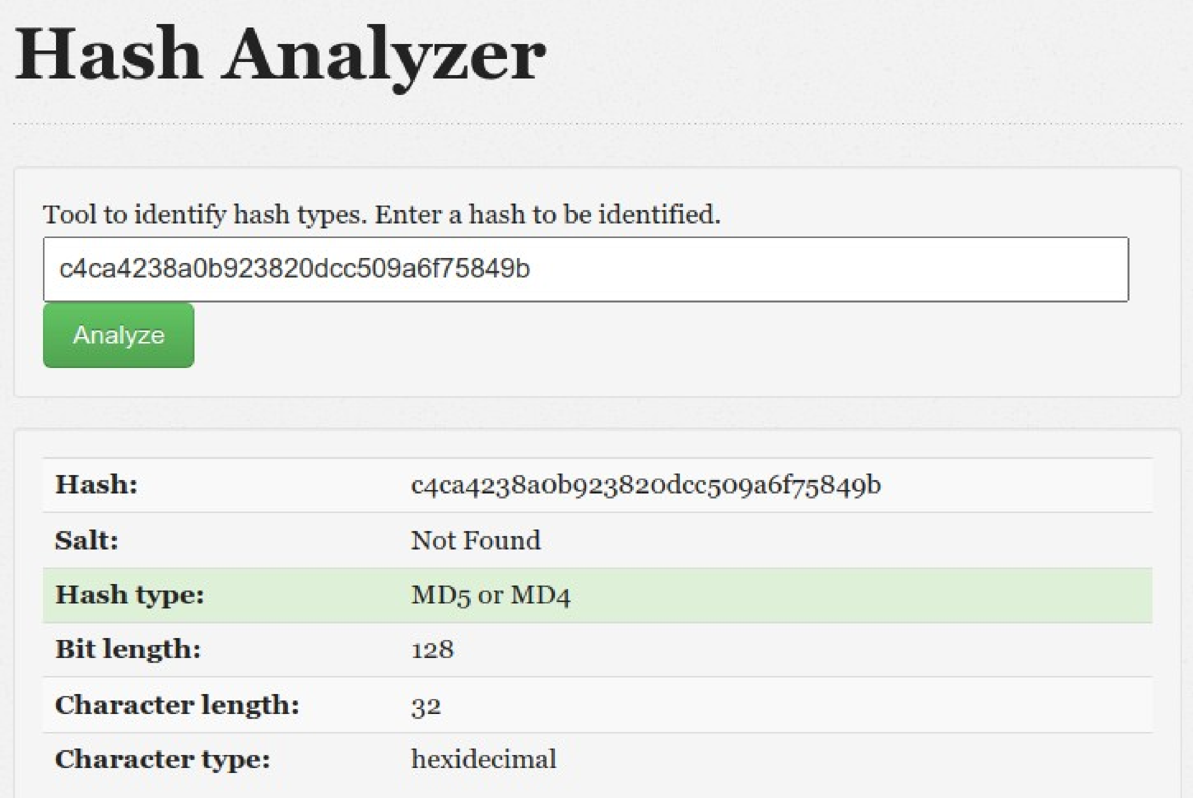
-When I was hovering over these doors with my mouse, I noticed that each door has a different path with different numbers , I had to look at the code for the page .





```
le                         Name
document:                  "Corridor"
▾ image map:               "" text label
    link:                  "c4ca4238a0b923820dcc509a6f75849b" keyboard
    link:                  "c81e728d9d4c2f636f067f89cc14862c" keyboard
    link:                  "eccbc87e4b5ce2fe28308fd9f2a7baf3" keyboard
    link:                  "a87ff679a2f3e71d9181a67b7542122c" keyboard
    link:                  "e4da3b7fbbce2345d7772b0674a318d5" keyboard
    link:                  "1679091c5a880faf6fb5e6087eb1b2dc" keyboard
    link:                  "8f14e45fceea167a5a36dedd4bea2543" keyboard
    link:                  "c9f0f895fb98ab9159f51fd0297e236d" keyboard
    link:                  "45c48cce2e2d7fbdea1afc51c7c6ad26" keyboard
    link:                  "d3d9446802a44259755d38e6d163e820" keyboard
    link:                  "6512bd43d9caa6e02c990b0a82652dca" keyboard
    link:                  "c20ad4d76fe97759aa27a0c99bff6710" keyboard
    link:                  "c51ce410c124a10e0db5e4b97fc2af39" keyboard
```

-After opening the source code, I found the codes in this form, so I decided to check them , i took all this to Note and then went to search and find out what kind of hash it is , went to ***https://www.tunnelsup.com/hash-analyzer/***


Hash Analyzer

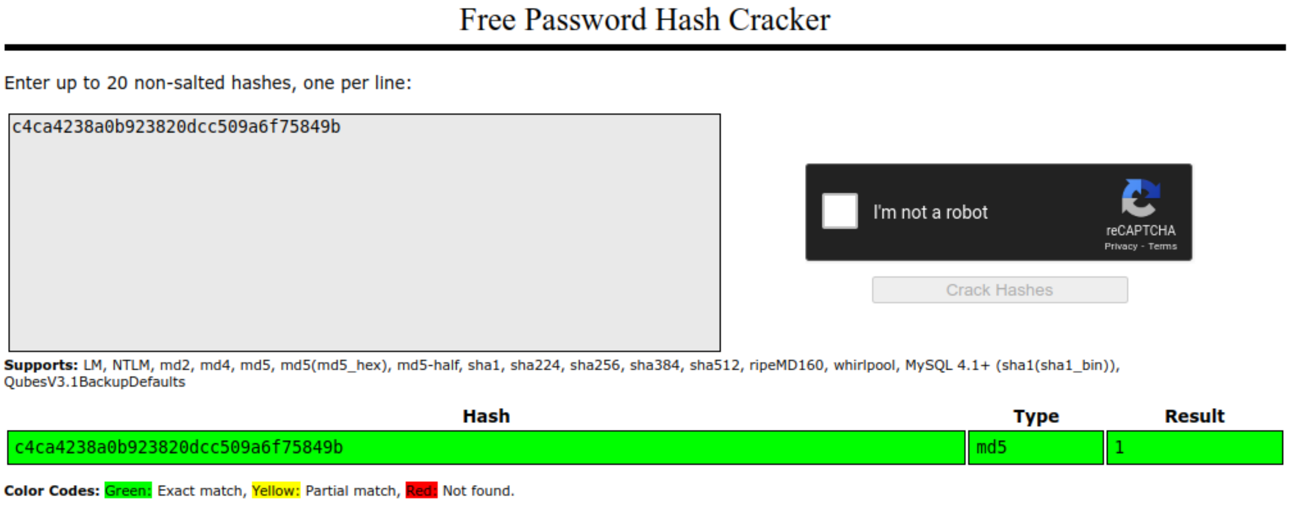
Tool to identify hash types. Enter a hash to be identified.

c4ca4238a0b923820dcc509a6f75849b

Hash:	c4ca4238a0b923820dcc509a6f75849b
Salt:	Not Found
Hash type:	MD5 or MD4
Bit length:	128
Character length:	32
Character type:	hexidecimal

I learned that the hash type (MD4 or MD5) , went to ***https://crackstation.net/***

-This was the result :



Free Password Hash Cracker

Enter up to 20 non-salted hashes, one per line:

```
c4ca4238a0b923820dcc509a6f75849b
```

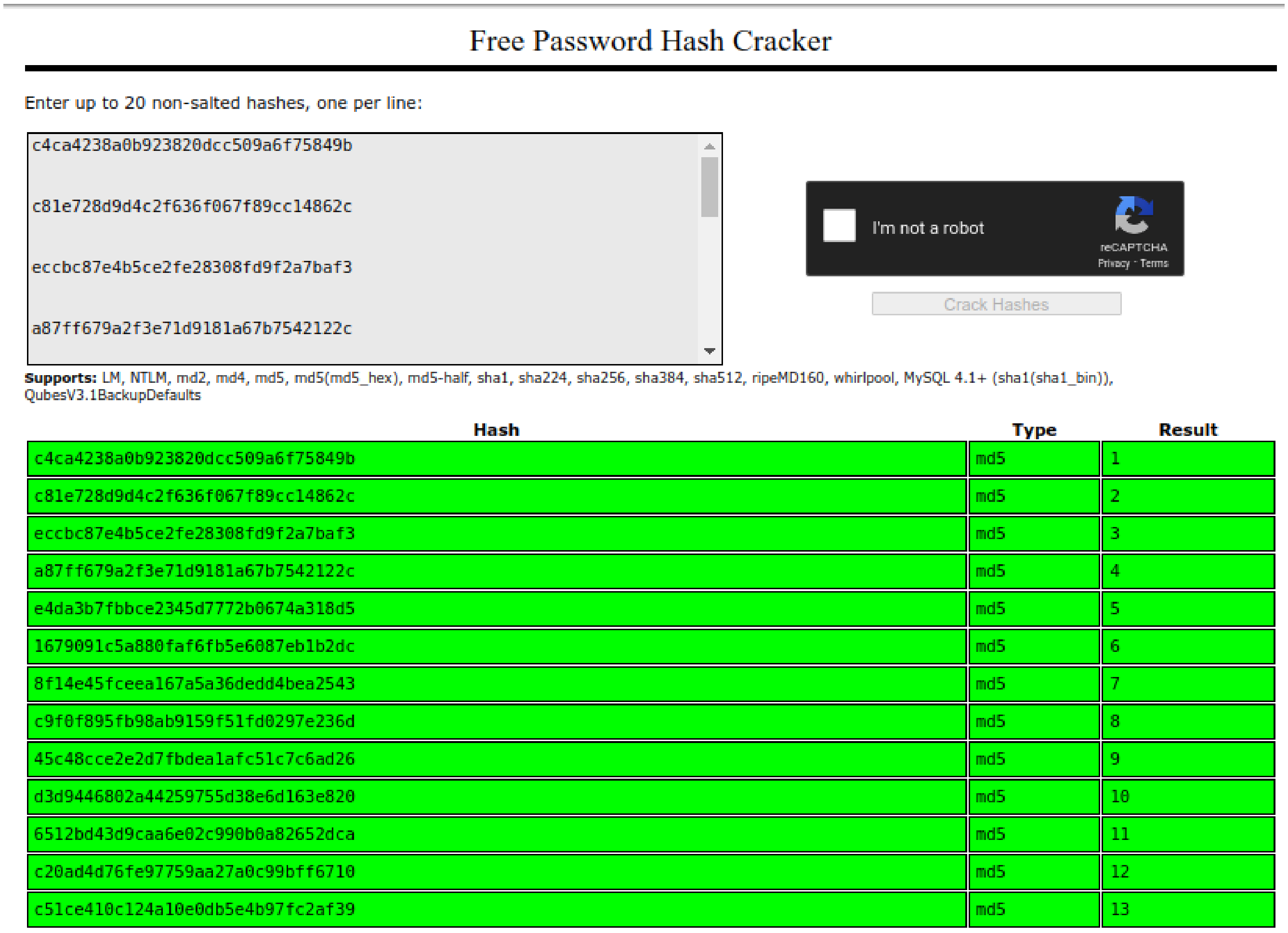
Supports: LM, NTLM, md2, md4, md5, md5(md5_hex), md5-half, sha1, sha224, sha256, sha384, sha512, ripeMD160, whirlpool, MySQL 4.1+ (sha1(sha1_bin)), QubesV3.1BackupDefaults

Hash	Type	Result
c4ca4238a0b923820dcc509a6f75849b	md5	1

Color Codes: Green: Exact match, Yellow: Partial match, Red: Not found.

-From here I was sure it was over **MD5** , But we should note something important here, the **Resalt is equal to 1 .**

-Well I have to see the result of the rest of the hashtags :



Free Password Hash Cracker

Enter up to 20 non-salted hashes, one per line:

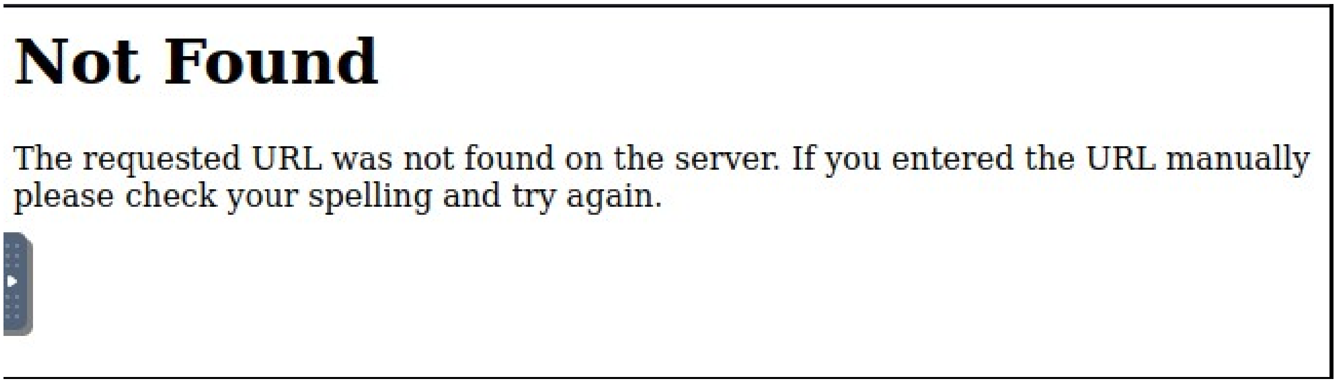
```
c4ca4238a0b923820dcc509a6f75849b

c81e728d9d4c2f636f067f89cc14862c

eccbc87e4b5ce2fe28308fd9f2a7baf3

a87ff679a2f3e71d9181a67b7542122c
```

Supports: LM, NTLM, md2, md4, md5, md5(md5_hex), md5-half, sha1, sha224, sha256, sha384, sha512, ripeMD160, whirlpool, MySQL 4.1+ (sha1(sha1_bin)), QubesV3.1BackupDefaults

Hash	Type	Result
c4ca4238a0b923820dcc509a6f75849b	md5	1
c81e728d9d4c2f636f067f89cc14862c	md5	2
eccbc87e4b5ce2fe28308fd9f2a7baf3	md5	3
a87ff679a2f3e71d9181a67b7542122c	md5	4
e4da3b7fbbce2345d7772b0674a318d5	md5	5
1679091c5a880faf6fb5e6087eb1b2dc	md5	6
8f14e45fceea167a5a36dedd4bea2543	md5	7
c9f0f895fb98ab9159f51fd0297e236d	md5	8
45c48cce2e2d7fbdea1afc51c7c6ad26	md5	9
d3d9446802a44259755d38e6d163e820	md5	10
6512bd43d9caa6e02c990b0a82652dca	md5	11
c20ad4d76fe97759aa27a0c99bff6710	md5	12
c51ce410c124a10e0db5e4b97fc2af39	md5	13

-Well, the results here are sequential in numbers from 1 to 13 by the number of doors ,

We felt that since this room is under the name of the IDOR vulnerability, I decided to put the numbers one behind the second behind the IP in the URL :



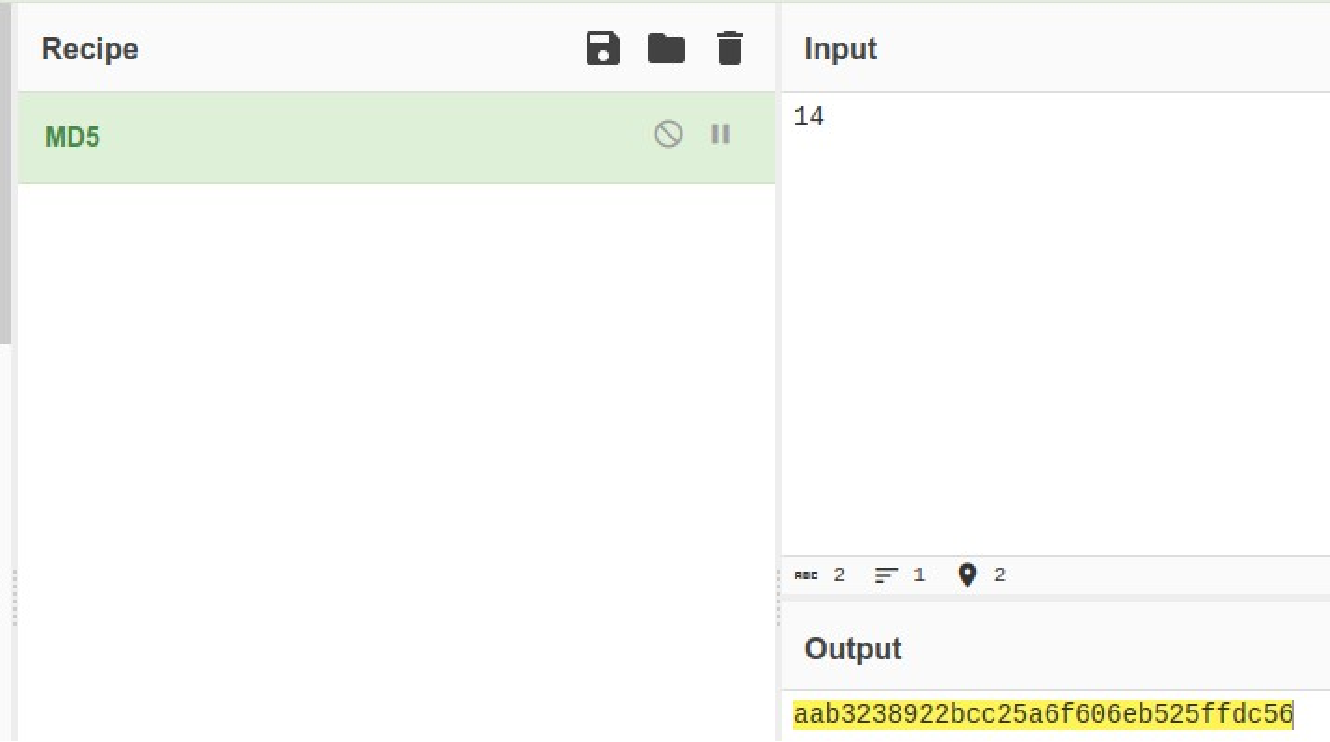
Not Found

The requested URL was not found on the server. If you entered the URL manually please check your spelling and try again.

-Well,it seems to be something else, but do not forget that it is the IDOR vulnerability,So we will try numbers before 1 and after 13 ,Well it didn't work .

-But the hash result was from 1 to 13, so we will make a hash MD5 For number 14 At

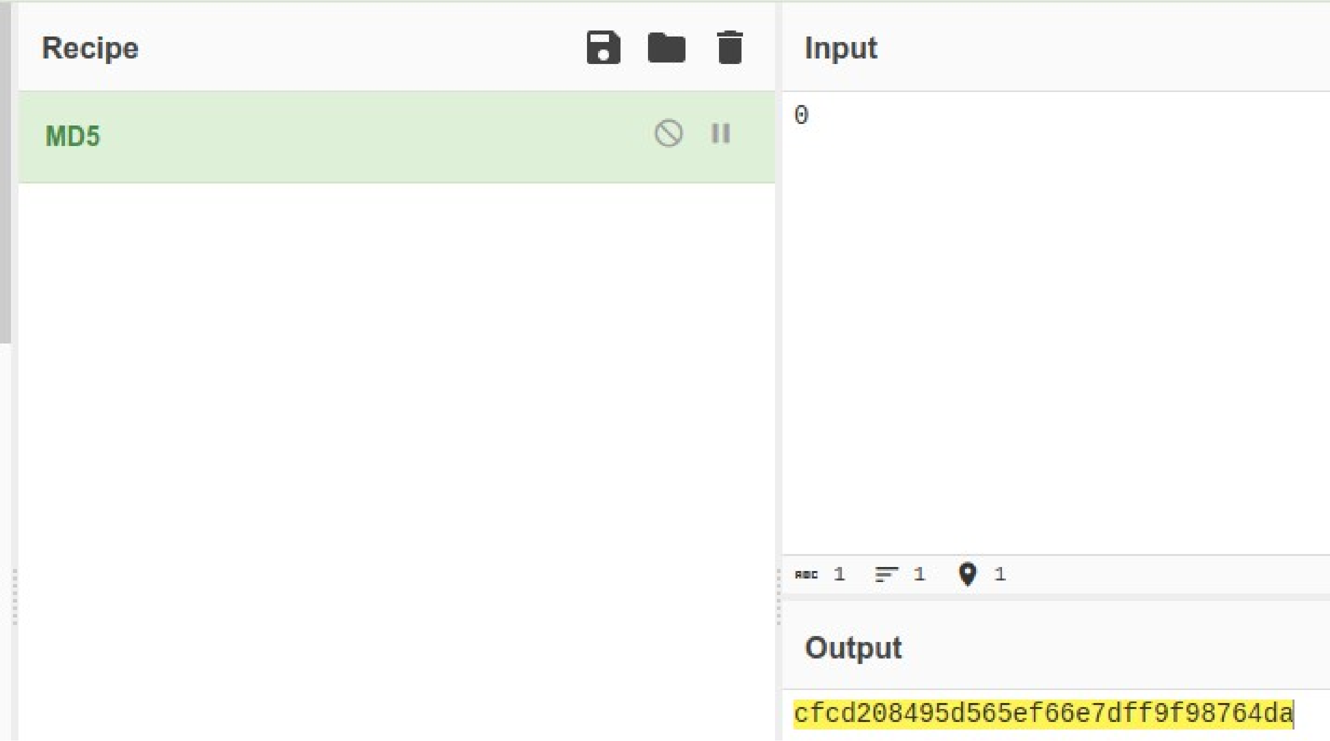
https://gchq.github.io/CyberChef/



```
14
```

```
aab3238922bcc25a6f606eb525ffdc56
```

-Then I tried it again by hashing it to the URL until I found the flag , But it didn't work



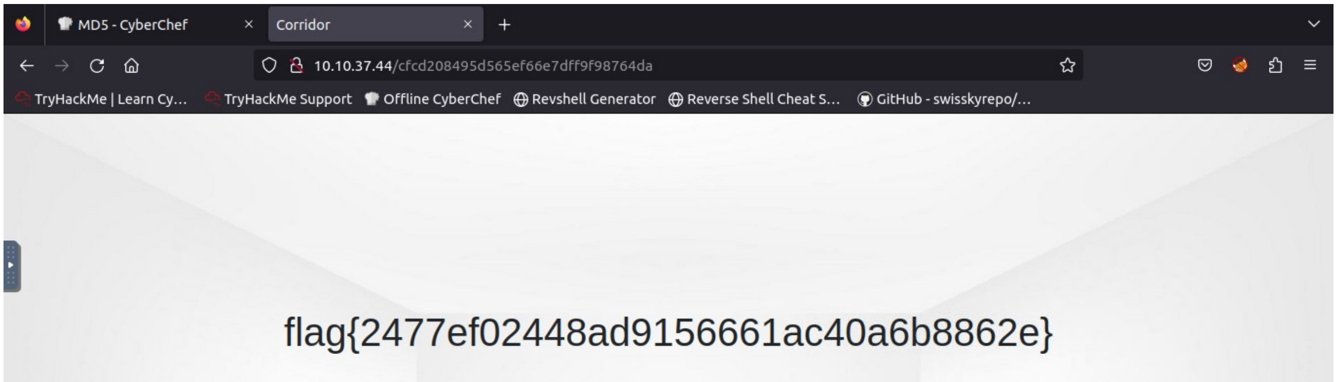
```
0
```

```
cfcd208495d565ef66e7dff9f98764da
```

-Then bring the hash of number 0 (cfcd208495d565ef66e7dff9f98764da) .

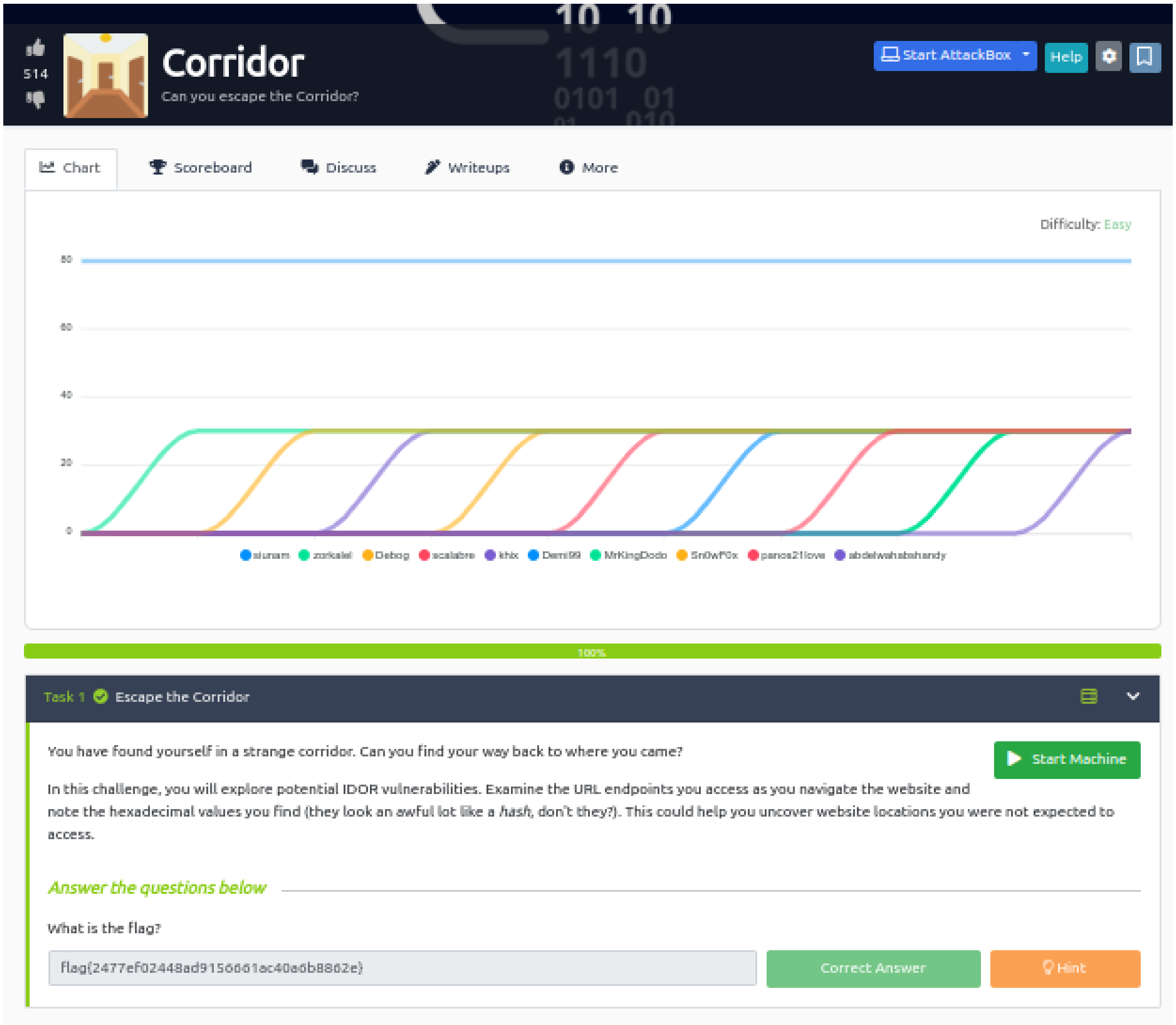
-Then try the hash of number 0 with the URL .

-Hey, it worked .



flag{2477ef02448ad9156661ac40a6b8862e}

Flag: **flag{2477ef02448ad9156661ac40a6b8862e}**



Task 1 Escape the Corridor

You have found yourself in a strange corridor. Can you find your way back to where you came?

In this challenge, you will explore potential IDOR vulnerabilities. Examine the URL endpoints you access as you navigate the website and note the hexadecimal values you find (they look an awful lot like a *hash*, don't they?). This could help you uncover website locations you were not expected to access.

Answer the questions below

What is the flag?

flag{2477ef02448ad9156661ac40a6b8862e}

Here are some additional tips for preventing IDOR vulnerabilities:

•Use parameterized queries instead of direct object references in database queries.

•Use input validation to prevent attackers from injecting malicious code into parameters.

•Use strong authentication and authorization mechanisms to protect resources.

•Regularly scan your applications for IDOR vulnerabilities.

BY : Abdelwahab_Ahmed_Shandy
AS_Cyber