Corridor – TryHackMe

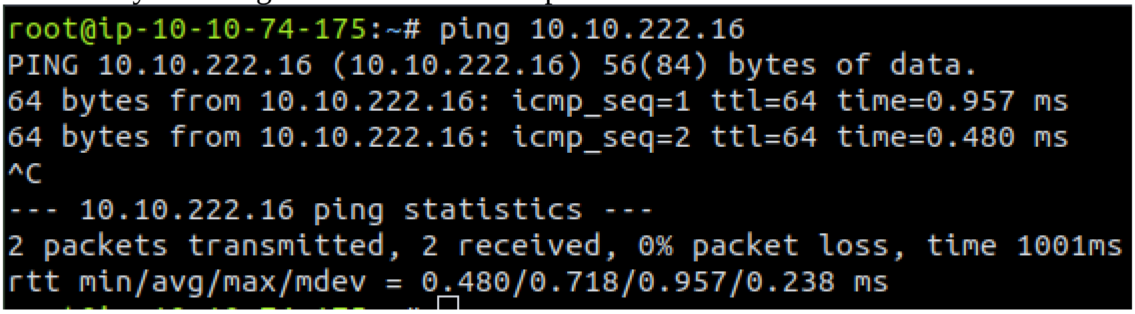
Objective: obtain the flag by exploiting an **IDOR** vulnerability.

Contents

1.Task / Steps taken... 1
2.Summary...3

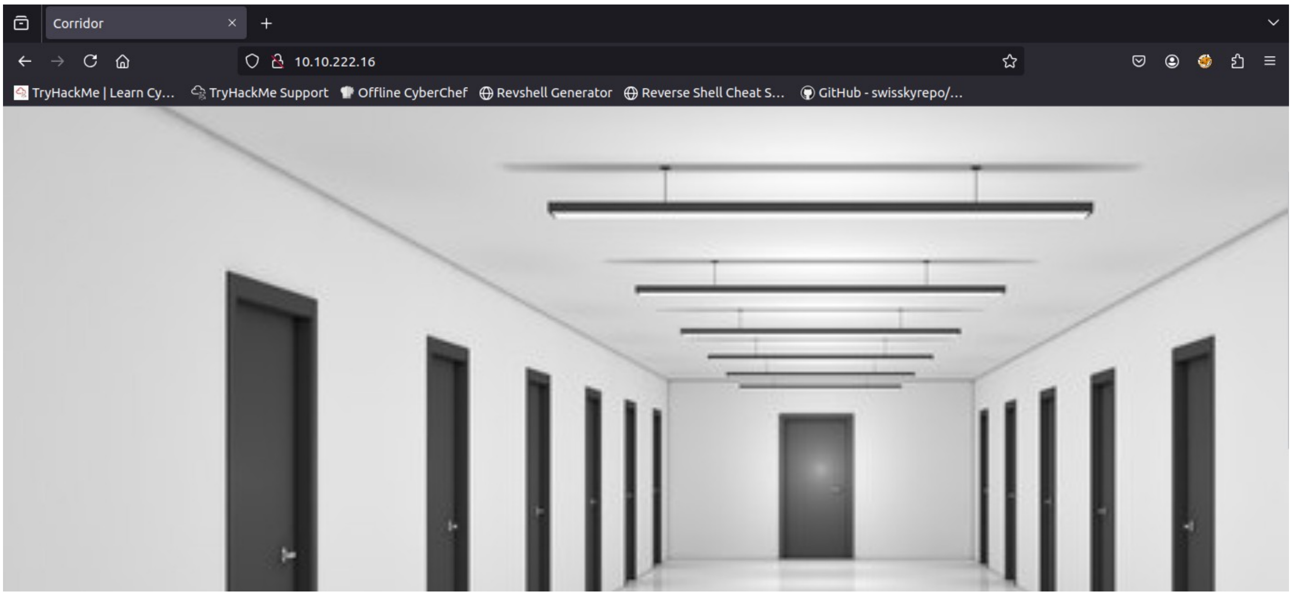
1.Task / Steps taken

We start by checking whether the host is up.

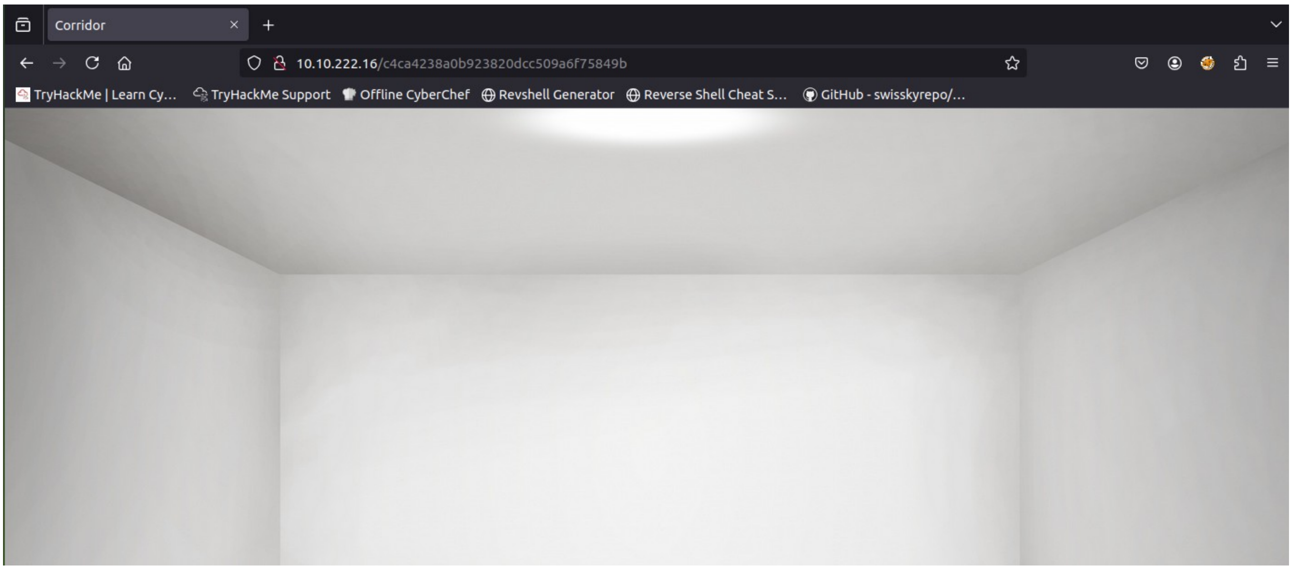


```
root@ip-10-10-74-175:~# ping 10.10.222.16
PING 10.10.222.16 (10.10.222.16) 56(84) bytes of data.
64 bytes from 10.10.222.16: icmp_seq=1 ttl=64 time=0.957 ms
64 bytes from 10.10.222.16: icmp_seq=2 ttl=64 time=0.480 ms
^C
--- 10.10.222.16 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 0.480/0.718/0.957/0.238 ms
```

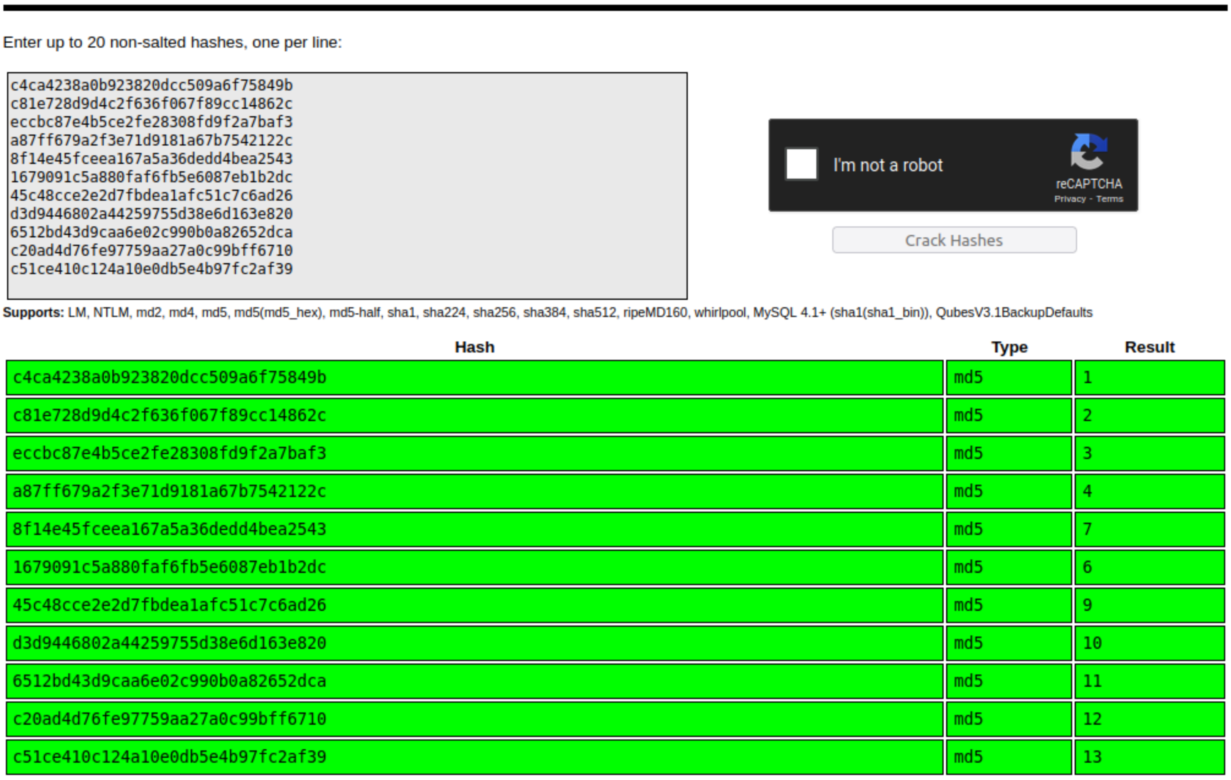
The host responds, and when we open the site we see a doors.



Clicking the door takes us to a subpage — the URL looks like some hash.



I copied those hashes and used **CrackStation** to crack them; they turned out to be sequential numbers.

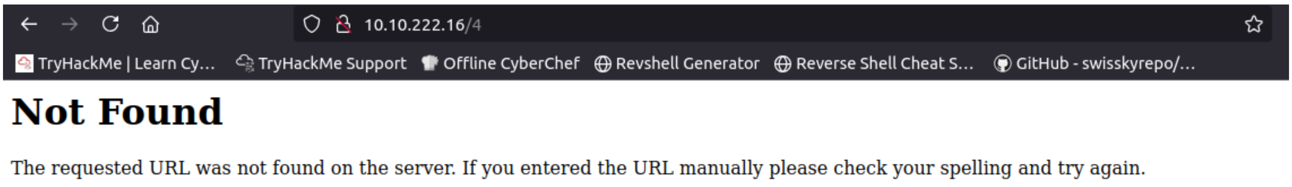

Enter up to 20 non-salted hashes, one per line:

```
c4ca4238a0b923820dcc509a6f75849b
c81e728d9d4c2f636f067f89cc14862c
eccbc87e4b5ce2fe28308fd9f2a7baf3
a87ff679a2f3e71d9181a67b7542122c
8f14e45fceea167a5a36dedd4bea2543
1679091c5a880faf6fb5e6087eb1b2dc
45c48cce2e2d7fbdea1afc51c7c6ad26
d3d9446802a44259755d38e6d163e820
6512bd43d9caa6e02c990b0a82652dca
c20ad4d76fe97759aa27a0c99bff6710
c51ce410c124a10e0db5e4b97fc2af39
```

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
8f14e45fceea167a5a36dedd4bea2543	md5	7
1679091c5a880faf6fb5e6087eb1b2dc	md5	6
45c48cce2e2d7fbdea1afc51c7c6ad26	md5	9
d3d9446802a44259755d38e6d163e820	md5	10
6512bd43d9caa6e02c990b0a82652dca	md5	11
c20ad4d76fe97759aa27a0c99bff6710	md5	12
c51ce410c124a10e0db5e4b97fc2af39	md5	13

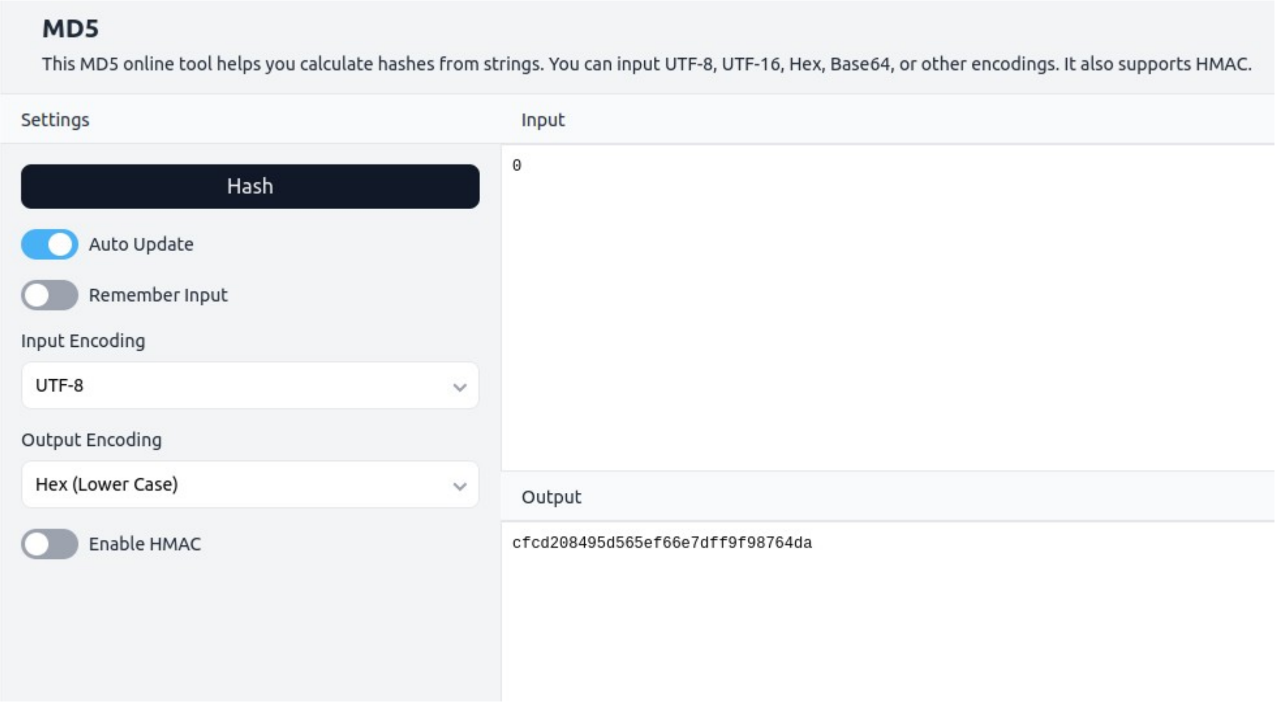
Instead of using the hashes, I tried visiting subpages 1–13 directly, but that did nothing.



Not Found

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

We know the vulnerability is **IDOR**, so we can try other indices like -1 or 0, but first we must convert those numbers to their **MD5** hashes.



MD5

This MD5 online tool helps you calculate hashes from strings. You can input UTF-8, UTF-16, Hex, Base64, or other encodings. It also supports HMAC.

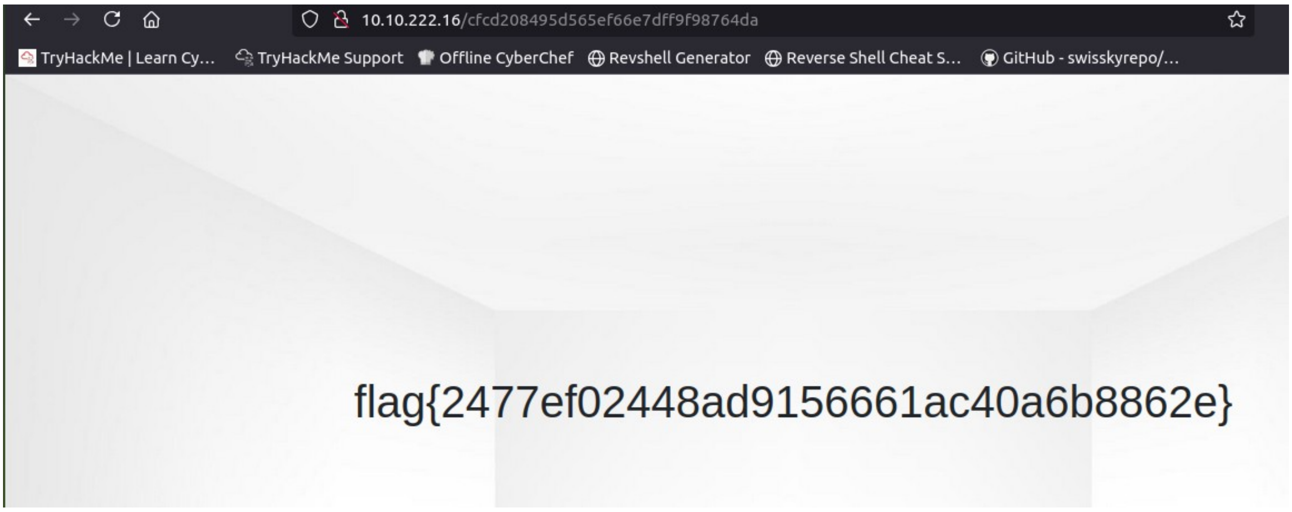
Input: 0

Input Encoding: UTF-8

Output Encoding: Hex (Lower Case)

```
cfcd208495d565ef66e7dff9f98764da
```

After visiting the subpage whose hash corresponds to the number 0, we found the flag.



10.10.222.16/cfcd208495d565ef66e7dff9f98764da

flag{2477ef02448ad9156661ac40a6b8862e}

2.Summary

This challenge is an IDOR-based web challenge: the site uses MD5-hashed numeric IDs in the URL. Cracking those hashes (e.g., with CrackStation) reveals the underlying numeric IDs; converting targeted IDs (like 0) to MD5 and visiting that hashed URL exposes the hidden flag. Key lesson: when IDs look hashed, try reversing them or enumerate nearby IDs (including negative/zero) and remember IDOR often stems from predictable ID schemes.