DLL Attacks   
(Red)

Graphical user interface, text, application

Description automatically generated

Summary

Attack Explains:

Windows operation system has implemented the shared library concept in which is DLL (dynamic-link libraries). It has the vulnerabilities of injection injecting malicious code into an application by exploiting the way some Windows applications.

this attack can evade process-based defenses as well as possibly elevate privileges. 

We are going to modify the 7z.dll in target’s system and to add local administration users to have a persistence user in the system.

Target Address: 172.26.15.94

Target System: Windows 10 Target file: 7z.dll

Account Compromised: brailee.ogden (windows account). Password: ‘Winter2022’

Account Created: francis.Net Password: ‘Difficult3’

Tools: OpenVpn, Evil-winrm, Invoke-Snow.ps1, Get-exports.py, Mingw-w64.

Download: 7z.dll

Connect to vpn and pivot to the target 172.26.14.94 windows 10 using evil-winrm.rb, and implemented the scripts and exefiles by using ‘-s’ and ‘-e ‘

Execute Invoke-Snow.ps1 to enable download function.

Text

Description automatically generated

Go to the “c:\ Program Files\7-zip” where 7z.dll file is located

Dir to see the target file 7z.dll

Graphical user interface, text

Description automatically generated

Execute “download” command to download 7z.dll to downloads directory in Evil-winrm.

Graphical user interface, text, application

Description automatically generated

Create Malicious DLL

Download DLLExample from github and put it in the evil-winrm for easy access.

After downloading, go to the ProxyDLLExample to fine the script we need : get\_exports.py

And copy that to downloads directory.

A screenshot of a computer

Description automatically generated with medium confidence

Inside of the downloads directory, verify 7z.dll, get\_exports.py are properly located at the same folder.

Use “./get\_exports.py –target 7z.dll” to export the malicious python script to 7z.dll.

Text

Description automatically generated

Copy the proxy.c, proxy.def, proxy.o from ProxyDLLExample and Modify the file.

change mirror()

add first line of system (“….”); to add user and password

add second line of system (“…. Administrators ”); to add the user to administractor groupf

Text

Description automatically generated

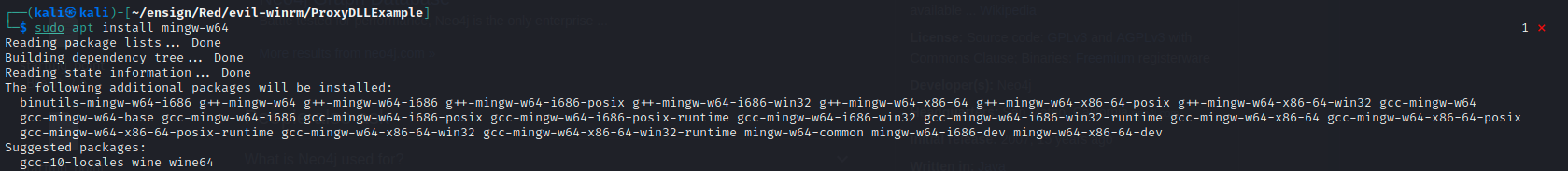
Try to execute x86\_6*5-w64-mingw32 command to compile files.*

*If it does not work, it mingw-w64 tool needs to be download*

Text

Description automatically generated

Download mingw-w64 tool using “apt install”



After downloading mingw-w64 compile tool, execute the two compile commands below.

This will create malicious dll file “proxy.dll” from the prerequisite files: proxy.c, proxy.def proxy.o

When the compile is done, verify the proxy.dll is created.

A screenshot of a computer

Description automatically generated

Upload and Execute Malicious DLL

Go back to evil-winrm on windows 10. And upload the proxy.dll

Save the 7z.dll to original.dll so the original DLL file will not be lost.

Then Remove the 7z.dll and move the proxy.dll to “c:\Program Files\7-zip\”

Text

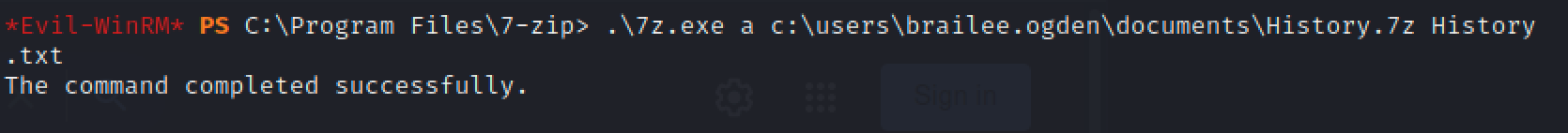
Description automatically generated

Then rename proxy.dll to 7z.dll by using “mv” command

A screenshot of a computer

Description automatically generated with medium confidence

Execute “.\7z.exe a c:\users\brailee.ogden\documents\History.7z History.txt” to created elevated users. And it shows “the command completed successfully.”



Execute “net users” to verify francis.Net is created which I added in proxy.c reverseshell function. This is that I can have persistence access to Host by using the windows account I created.

Text

Description automatically generated