Privilege escalation in windows

Premise:

We have access to the powershell of a target system. Problem is, we’re just a low-level user called “student” and we want to get to the administrator account.

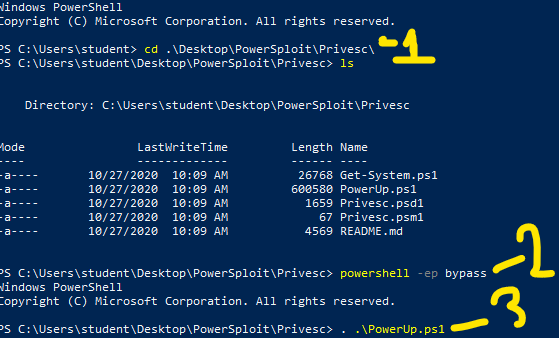
Through our shell session, a team member already uploaded PowerSploit\Privesc to the target. They’re exhausted for now, so you have to take the helm for the rest of the engagement.

First, change into the privesc directory.

The program we want to run is PowerUp.ps1. However, before we can do this, we need to invoke command: powershell -ep bypass

Otherwise, this will block the execution of the script. If you’re getting blocked in the field in a powershell environment, try invoking this powershell -ep bypass command.

Then (3) call the PowerUp program.



Run command: Invoke-PrivescAudit

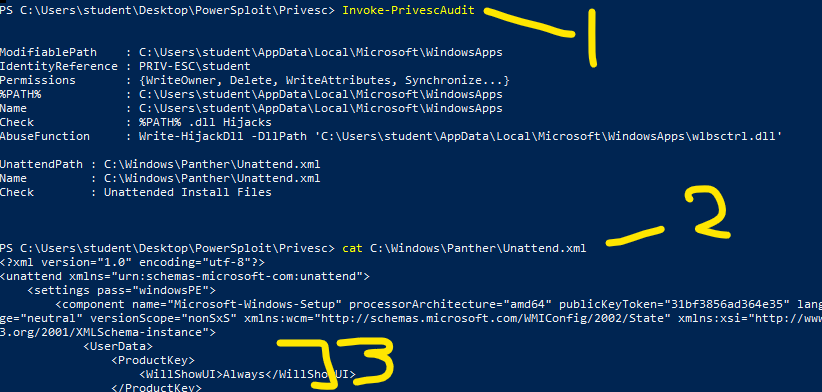
There’s a set path in windows to a possible helpful file.

C:\Windows\Panther\Unattend.xml

You’re more likely to find this on a system part of a larger network that requires a lot of frequent updates, as this allows for some automations to be executed on enabled systems. We’re going to use this to our own advantage.

Cat out the contents of C:\Windows\Panther\Unattend.xml

If that’s not there, look for C:\Windows\Panther\Autounattend.xml



There’s a lot of data output in this xml file…

Look for any kind of password. It will be between tags as shown in part 3 above.

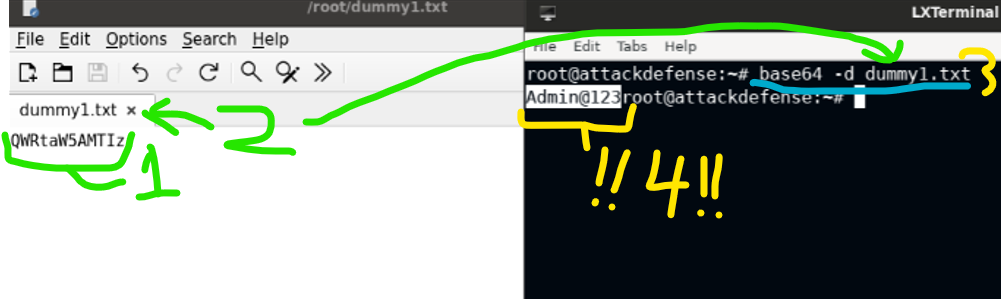
Assume, for this lab, we found the data “QWRtaW5SAMTIz”

This looks like some kind of encryption. We can use burpsuite to help us decrypt this, but since I have a hunch this looks like base64 to me. A faster way to decrypt a one-off like this is from CLI. Make a notepad doc with the text data that’s encrypted, then call command:

base64 -d my\_file.txt

-d tells the program to decrypt the contents of the file. This function is available on your ready-made Kali distro, and it a pretty common low-overhead tool.

Run it, and we get the decryption faster than those scrubs who are still waiting for burpsuite to boot up.



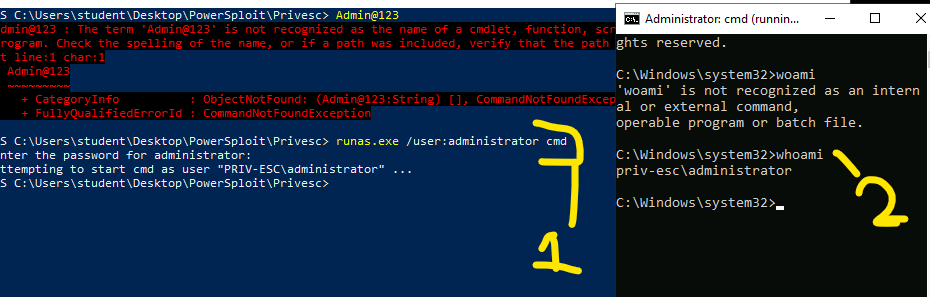
1:input text

2: save the file, here with the name dummy1.txt

3: call command: base64 -d dumy1.txt

4: results of the decryption are : Admin@123

A little hard to read on CLI, but as long as you know what to look for in output, you’re going to be fine.



Command: runas.exe /user:administrator cmd

This will prompt us for the pass, but we got the credentials now as Admin@123, so we can login.

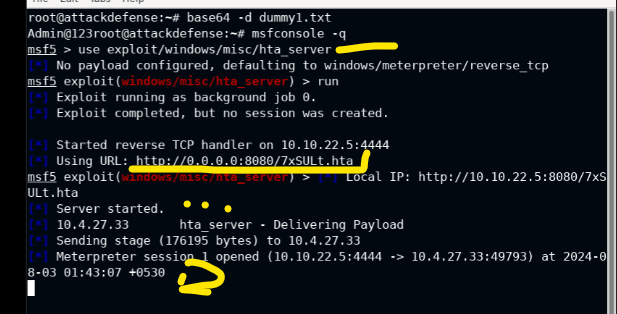
We’re calling the “cmd” keyword to pop open a command prompt in the target system, running it AS the administrator account.

In 2, you see the whoami command returning that we are the administrator now

Now, let’s pop open Metasploit with msfconsole -q

This is an HTA server exploit we want to use, and it’s defaults will get us the reverse TCP session we want.

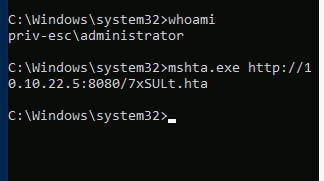
The session will give us some kind of code at the end of an HTTP url string. We’re interested in the /7xSULt.hta string. (Might just be coincidence, but that random string reminds me of my ex-wife for some reason. Anyway, moving on)



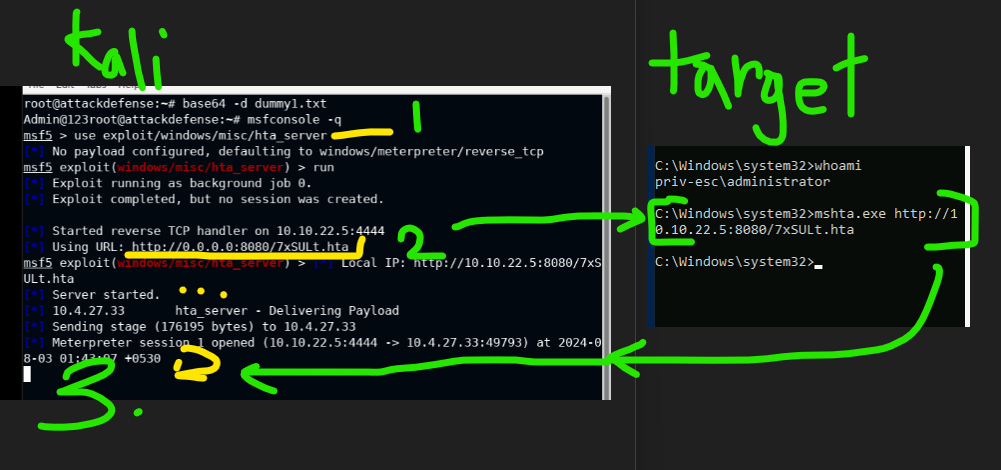
We see the server above is started, giving us a meterpreter session.

We can then move to the cmd window we got open as the administrator and invoke command:

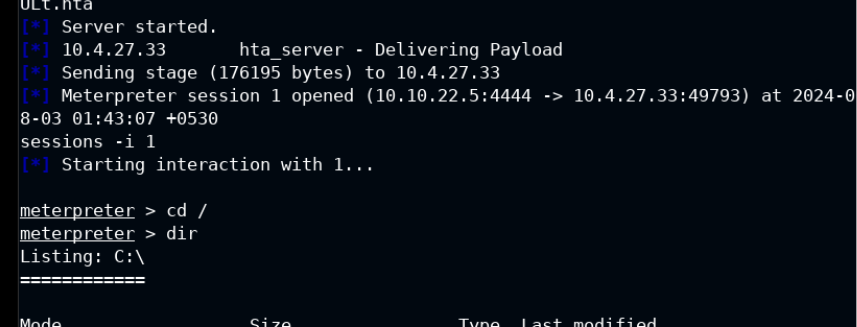
Mshta.exe and use the url for the reverse TCP session we just set up.



(A clearer view of the 2 windows simultaneously, where Kali is out attack machine on the left, and the victim or target is the cmd window on the right.)



With the meterpreter session open, cd into the main drive.



We want to get to the Admin’s desktop to capture the flag there. Navigate to it and cat out the file to get the flag.

