Winrm lab:

Winrm uses http or https to allow remote access to a machine. It’s a Microsoft windows admin tool, and when used correctly, lets you execute commands on windows systems.

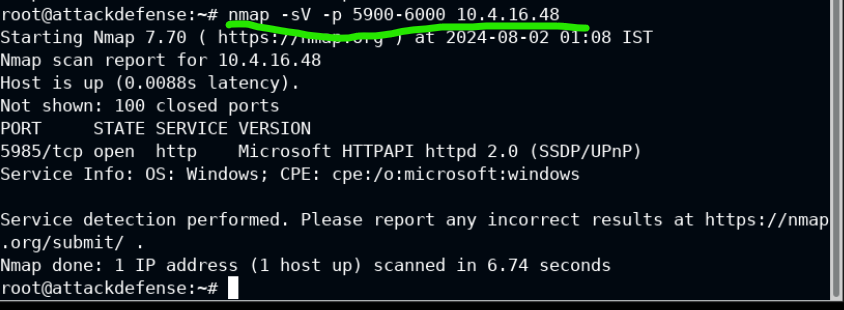
It’s TCP 5985 or for https 5986 as per the standard.

The tools “crackmapexec” as well as the script “evil-winrm.rb” are useful for taking advantage of winrm.

Starting with our nmap scan, we know an interesting service is somewhere in the port range of 5900 - 6000, outside of the normal port range (the start of this lab is black-box, so if you’re using this as a lesson plan, then just give the range to help build up skills, and don’t let them know it’s winrm as a target. Let them find that 5985 is open, then reveal that it is winrm and go over the basic info, so the students “discover” the port and make it more engaging).

So, we need to use a special scan for this range:

Command: nmap -sV -p 5900-6000 10.4.16.48

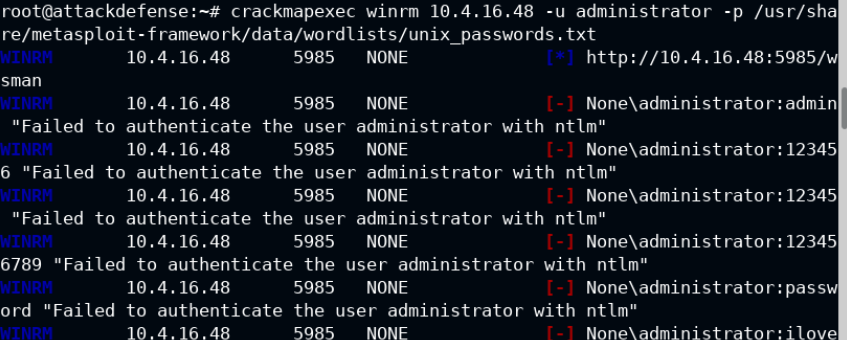
5985 is shown, halfway down, as an interesting port in response. (and our foot in the door to talking about winrm)

We then go into crackmap exe.

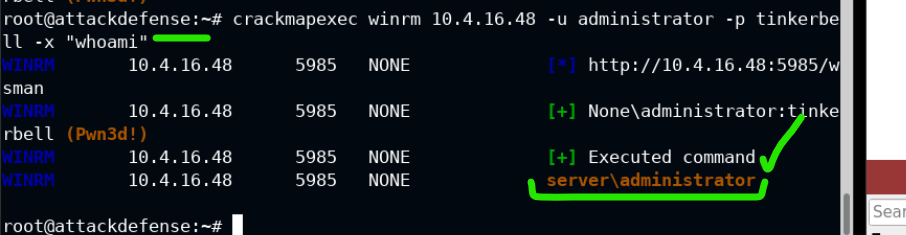
We know the administrator account exists as an assumption, so we need a password list that we feed into the brute-force. Unix\_passwords will do.

Command:

crackmapexec winrm 10.4.16.48 -u administrator -p /usr/share/metasploit-framework/data/wordlists/unix\_passwords.txt



Running the brute force returns the username of administrator attached to the password of ‘tinkerbell’



You can then run a command to -x for “execute” a command on the target. Above, the screencap just shows “whoami” but you can get more data with “systeminfo” shown in the command below

Command: crackmapexec winrm 10.4.16.48 -u administrator -p tinkerbell -x "systeminfo"

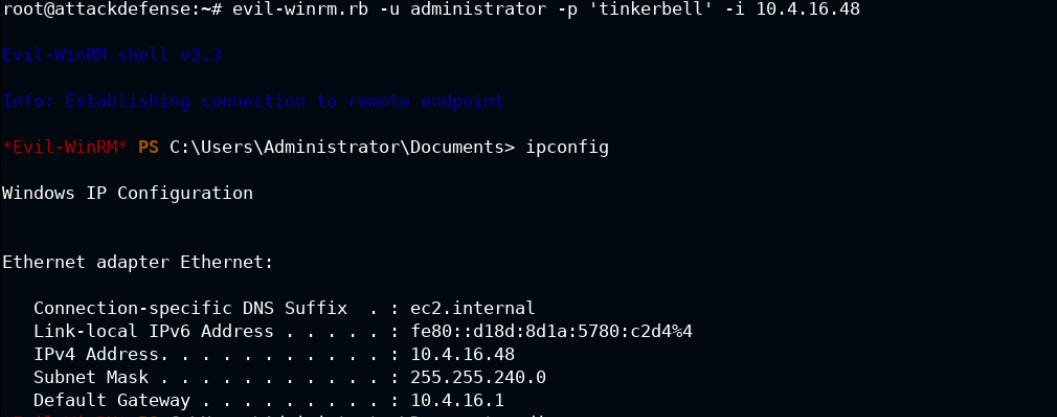
PHASE2:

Using the evil-winrm.rb script to get in

(no context exists, so the teachers got lazy here. Just use a ruby script because it exists, I guess? The reasoning for focusing on this is not clear at all)

Anyway, we can run this command, since we got the credentials for the target:

Command: evil-winrm.rb -u administrator -p ‘tinkerbell’ -i 10.4.16.48



(screencap shows the ipconfig)

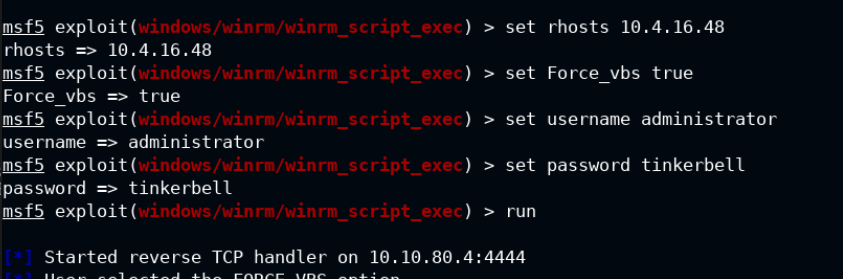
Even less context is given for the next leg of the lab.

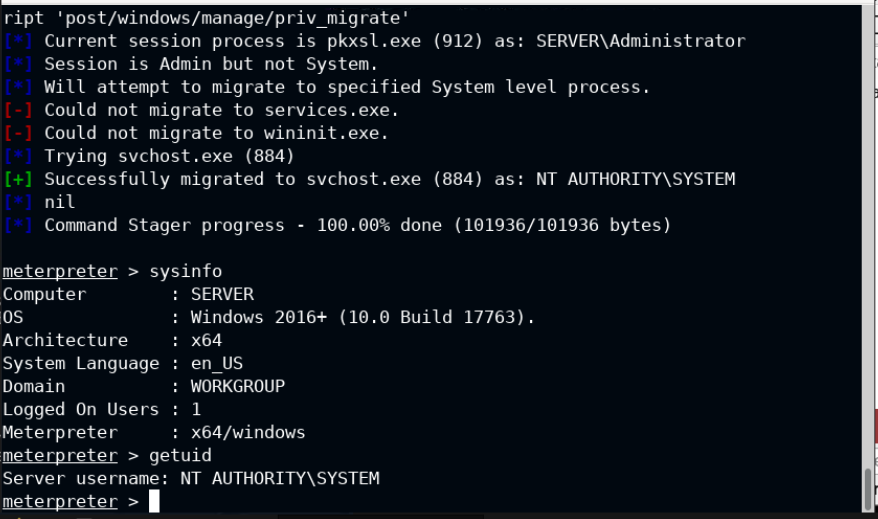
We want to use some script to get a bash shell? The instructors were not clear how this script was selected (wow, sounding like a broken record, huh?)

use exploit/windows/winrm/winrm\_script\_exec

we have to set “force\_vbs” to “true” because we want to get a shell on the target.

The other settings are shown in the screencap.





Above, we’re just typing in sysinfo to get data, and then getuid to get more data.

It’s kind of just showing that we’re in the shell.

We can also navigate to the target’s file structure if a flag existed, but there is none for this exercise.