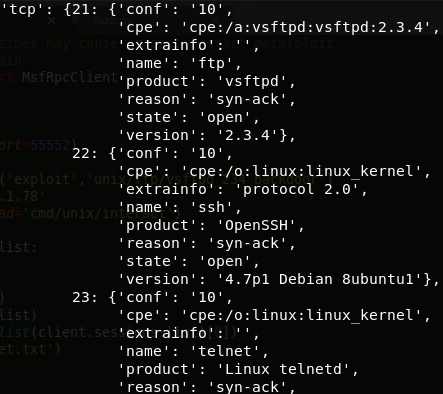
Penetration Testing Lab

In this lab you will be exploiting a vulnerable target machine using Python scripts. The machine is located at 10.50.XX.78 where XX is your group number. All tasks must be executed remotely meaning you can not log onto the target machine. It is highly advised that you test your exploits from the Python shell before transferring them to script form. You will be turning in your working Python scripts along with your lab report. You must explain how each script works in your report.

Part 1 intelligence gathering:

Write a script that will scan and print all the open ports. Be sure to also include the services they are running. Additionally, your script must be able to identify and print the OS type that the target machine is running. You can use whatever libraries you want. You can to do extra intelligence gathering outside of what is required from scripts if you need to. Add screenshots of your results to your report. Save your script as reconnaissance.py.



*Port scan results*

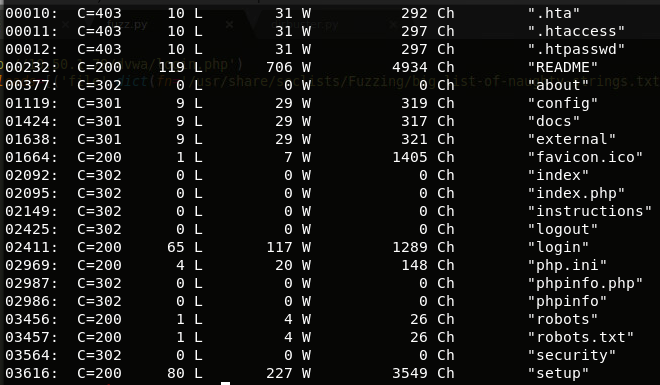
* What OS is the target machine running? Linux
* Is port 5050 open? No
* Is port 8180 open? No
* What service is port 23? Linux telnetd
* What service is port 139? Samba

Part 2 Dirbuster

The target machine is running a wide variety of vulnerable web applications. Choose one and write a script to crawl any possible common files it might have. You will want to download a wordlist for this activity:

* <https://github.com/danielmiessler/SecLists>
* <https://www.netsparker.com/blog/web-security/svn-digger-better-lists-for-forced-browsing/>

You might want to filter out certain responses so your output is readable (e.g. 404 responses). Include screenshots of your results. List any noteworthy results. Save your python script as dirbuster.py

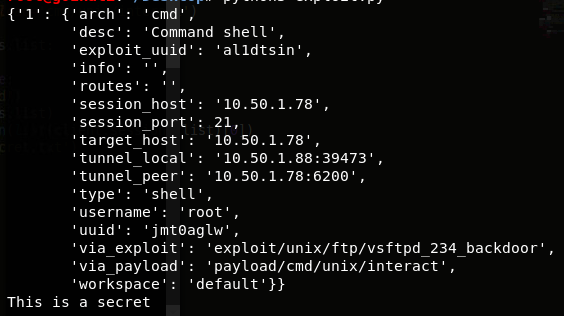


*Dirbuster script interesting results*

* How would you know if your script browsed to a valid file? It should return a 200/300 level or similar http code response.

Part 3 Exploit

There is a secret file on the machine called secret.txt. Write a script that exploits the machine and prints the contents of the file. It is highly advised that you start testing with the Metasploit console and Python console before transferring your code over to a script. Include screenshots of your results and explain what methods/exploits you used to hack into the machine. Save your script as exploit.py.



*Exploit results and contents of secret.txt*

* What does the secret file say? This is a secret
* Where was it located? /root

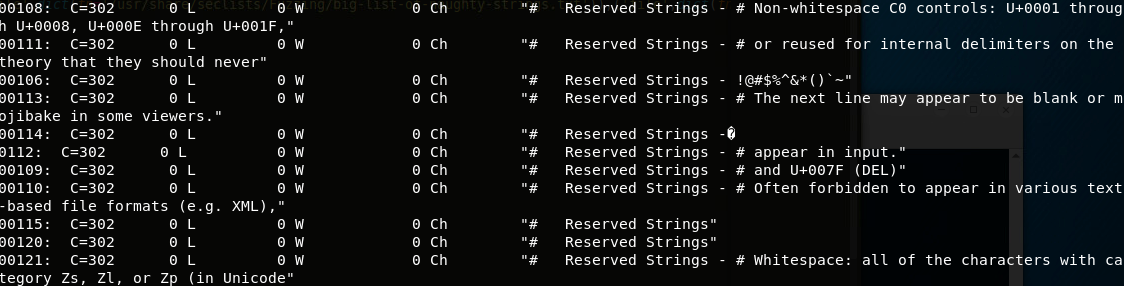
Part 4 Wfuzz

The target machine is running several vulnerable web applications. Select one of the web applications and use wfuzz to conduct a fuzz attack on one of its forms (this means you are fuzzing a POST request). The attack does not have to be successful. Use a wordlist file as your fuzzing material. Refer to

* <https://github.com/fuzzdb-project/fuzzdb>
* <https://github.com/danielmiessler/SecLists>

Include any noteworthy results in your report including whether or not you were successful. You might also want to filter out certain results so that your output is readable (e.g. 302 responses). Include screenshots of your results. Save your script as fuzz.py

\*\*Note make sure to include the Login button name and value in your fuzzed request if one is included\*\*



*Fuzzing Results*

* List one way you would know your attack was successful? It returns a 200 level http code response