2016_CSAW_CTF_Quals_Reverse_Rock100 Writeup

原创

Flying_Fatty 于 2017-04-14 11:44:59 发布 1644 收藏
分类专栏： CTF之旅 reverse

百度杯提供了Rock题目

GitHub上有Writeup

先运行找关键点

```
1234
----------------------------------------
Quote from people's champ
----------------------------------------
*My goal was never to be the loudest or the craziest. It was to be the most enter
taining.
*Wrestling was like stand-up comedy for me.
*I like to use the hard times in the past to motivate me today.
----------------------------------------
Checking....
Too short or too long
```


1234是自己随意输入的，可以看到提示字符串：Checking和Too short or too long

说明检查函数给了我们提示字符，用IDA-string查找一下

```
.rodata:0000000000401A00 aFlag2345691236 db 'FLAG23456912365453475897834567',0
.rodata:0000000000401A00                                        ; DATA XREF: sub_4015DC+7C↑o
.rodata:0000000000401A1F aTooShortOrTooL db 'Too short or too long',0
.rodata:0000000000401A1F                                        ; DATA XREF: sub_4016BA+28↑o
.rodata:0000000000401A35 aPass           db 'Pass ',0           ; DATA XREF: sub_4017E6+58↑o
.rodata:0000000000401A3B aYouDidNotPass  db 'You did not pass ',0
.rodata:0000000000401A3B                                        ; DATA XREF: sub_4017E6:loc_40186D↑o
```


至少看到了两个关键函数：4016BA和4017E6

还有个FLAG的字符串，猜测是作为初始化的值然后需要做运算等，4015DC也是有用的

进入main中，查看函数逻辑，先是读取我们的input，存入v18变量，然后用4015DC函数对于v18做处理之后，存入v23

```
44| sub_4015DC((__int64)&v23, (const std::string *)&v18);
```

接着是4016BA的处理，因为我们需要看到Flag，所以4017E6处的判断需要成立，如下：

```
54  sub_4016BA((__int64)&v23);
55  if ( (unsigned int)sub_4017E6((__int64)&v23) == 0 )
56  {
57    LODWORD(v11) = std::operator<<<std::char_traits<char>>(&std::cout, "//////////////////////////////////");
58    std::ostream::operator<<(v11, &std::endl<char,std::char_traits<char>>);
59    LODWORD(v12) = std::operator<<<std::char_traits<char>>(&std::cout, "Do not be angry. Happy Hacking :)");
60    std::ostream::operator<<(v12, &std::endl<char,std::char_traits<char>>);
61    LODWORD(v13) = std::operator<<<std::char_traits<char>>(&std::cout, "//////////////////////////////////");
62    std::ostream::operator<<(v13, &std::endl<char,std::char_traits<char>>);
63    sub_4018D6(&v22, &v23);
64    LODWORD(v14) = std::operator<<<std::char_traits<char>>(&std::cout, "Flag{");
65    LODWORD(v15) = std::operator<<<char,std::char_traits<char>,std::allocator<char>>(v14, &v22);
66    LODWORD(v16) = std::operator<<<std::char_traits<char>>(v15, "}");
67    std::ostream::operator<<(v16, &std::endl<char,std::char_traits<char>>);
68    std::string::~string((std::string *)&v22);
69  }
```

那么进入三个函数去看功能

4015DC只给了我们初始字符串，4016BA是对我们输入的字符串进行判断和处理，先是判断长度是否为30，然后进行两次for循环都是简单xor操作

```
14  if ( std::string::length((std::string *)(a1 + 16)) != 30LL )
```

4017E6是判断初始字符串和处理后的输入字符串是否相等

只需要把过程逆着就好了，简单数学

```
string = 'FLAG234569123654534758978345 67'
flag = ''
for i in string:
 flag += chr((((ord(i) - 9) ^ 16) - 20) ^ 0x50)
print flag
```