[RE]lab1B&lab1A's writeup&脚本；使堆栈平衡的另一种方法

原创

PepperYouth 于 2017-12-03 09:48:03 发布 242 收藏

分类专栏： RE 文章标签： RE-CTF

本文链接：https://blog.csdn.net/PepperYouth/article/details/78699903

lab1B

main：

1.读取输入（数字密码，password）

2.将 password 与数字 322424845 传入test函数

test：

3.将 322424845 - password 的结果（result）传入decrypt，如果 result 的范围不在1~21以内，则传入一个随机数

decrypt：

4.将字符串中的每一个字符与result异或，若得到的字符串结果为”Congratulations!”，则成功，否则失败。

```
>>> print(322424845-(ord('Q')^ord('C')))
322424827
```

---

lab1A

main:

1.读取输入字符串（username），以及数字（serial）

2.将username，serial传入函数auth，若auth的返回值为0，则成功

auth：

3.若auth的返回值要为0，则需要满足：

a.username的长度length大于5

b.ptrace(0, 0, 1, 0)!=-1(这个ptrace函数似乎与进程跟踪有关，百度了一下，依然不是很懂)

c.username中的每一个字符的ascii值大于31

满足上述条件，执行的代码块是：

```
v4 = (username[3] ^ 0x1337) + 6221293;
for ( i = 0; i < length; ++i )
    v4 += (v4 ^ username[i]) % 0x539;
result = serial != v4;
// serial != v4 的值应为0，则v4就等于serial
```

用python实现一下：

```
    def fun(username):
        length = len(username)
        v4 = (ord(username[3]) ^ 0x1337) + 6221293
        for i in range(length):
            v4 += (v4 ^ ord(username[i])) % 0x539
        return v4
    print(fun("Pepper"))
```

使堆栈平衡的另一种方法

既然知道了是“add esp 4”使堆栈不平衡的，

那么在汇编代码段中选中4，

转到Hex-View窗口，F2，把4改成0，再次F2保存编辑结果，

同样也是可行的。