CTFwp3- crypto-pyc文件

原创

Mi piace 于 2020-12-27 00:39:47 发布 121 收藏

分类专栏： CTFwp

本文链接：https://blog.csdn.net/qq_52196579/article/details/111771460

ctf- pyc文件

ORZ (*_*)

1，pyc文件简介

pyc是一种二进制文件，是由py文件经过编译后，生成的文件，是一种byte code，py文件变成pyc文件后，加载的速度有所提高，因为py文件是可以直接看到源码的，如果不希望泄露源代码就需要编译为pyc后再发布。当然，pyc文件也是可以反编译的，不同版本编译后的pyc文件是不同的。

python可以用 $uncompyle6$ 库来进行反编译pyc文件。

具体操作是，pip install uncompyle6 后，在cmd里pyc文件的位置处，用uncompyle6 -o. file.pyc，就可以得到反编译的python文件了。

2，题目

攻防世界的crypto题目，主要就是了解一下pyc和简单的代码反写。

py文件打开后是

```
import base64
def encode1(ans):
    s = ''
    for i in ans:
        x = ord(i) ^ 36
        x = x + 25
        s += chr(x)
    return s
def encode2(ans):
    s = ''
    for i in ans:
        x = ord(i) + 36
        x = x ^ 36
        s += chr(x)
    return s
def encode3(ans):
    return base64.b32encode(ans)
flag = ' '
print('Please Input your flag:')
flag = input()
final = 'UC7KOWVXWVNKNIC2XCXKHKK2W5NLBKNOUOSK3LNNVWW3E==='
if encode3(encode2(encode1(flag))) == final:
    print('correct')
else:
    print ('wrong')
```

这就很明显了，只需要照着那一串base64和前面的俩加密写回去，

```
import base64
str='UC7KOWVXWVNKNIC2XCXKHKK2W5NLBKNOUOSK3LNNVWW3E==='
s1=base64.b32decode(str).decode('ISO-8859-1')
m=''
flag=''
for i in s1:
   m+=chr((ord(i)^36)-36)
for i in m:
    flag+=chr((ord(i)-25)^36)
print(flag)
```

其中的关键问题就是那个’ISO-8859-1’ 编码的问题，emm,是看了wp在stackflow上找到的解决方案。

运行得到flag。