Seclab-----WriteUp

原创

向那风 于 2018-11-10 10:54:40 发布 430 收藏
分类专栏： WriteUp 文章标签： WriteUp

密钥生成-不能再简答的RSA

在一次RSA密钥对生成中，假设p=473398607161，q=4511491，e=17 求解出d。
答案格式：flag{d}

公钥 KU	n：两素数 p 和 q 的乘积（p 和 q 必须保密） e：与（p-1）（q-1）互质
私钥 KR	d：e^{-1} (mod(p-1)(q-1)) n：
加密	$C \equiv m^e \bmod n$
解密	$m \equiv c^d \bmod n$

根据这张表，其中e^{-1}，表示 e在(p-1)(q-1)中的乘法逆元，mod 是求余运算。

解得 (p-1)(q-1) = 2135733082216268400

```
a:17
b:2135733082216268400
所求的逆元：125631357777427553
>>>
```

e-1 = ,即，125631357777427553

所以， d = 125631357777427553

也可以使用脚本跑

```
import gmpy2



p = gmpy2.mpz(473398607161)#初始化

q = gmpy2.mpz(4511491)

e = gmpy2.mpz(17)

phi_n = (p - 1) * (q - 1)

d = gmpy2.invert(e, phi_n)

print ("private key:")

print (d)
```

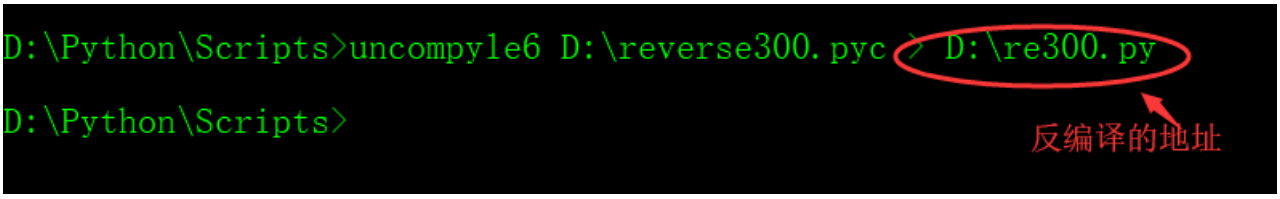
程序加密

有一个程序加密得到以下密文.....

下载后修改后缀为zip，解压里面只有一个线索文件和.pyc文件。

什么是.pyc文件，pyc和py文件的区别

使用uncompyle6模块进行python3反编译，



反编译过后是个python2脚本，研究一下代码，

```
if __name__ == '__main__':
    if len(sys.argv) < 3:
        exit(1)
    ex = 20
    for i in range(1, len(sys.argv), 2):
        a = sys.argv[i]
        b = sys.argv[i + 1]
        if a == '-t':
            ex = int(b)
        elif a == '-e':
            encoded = authcode(b, 'ENCODE', expiry=ex)
            print encoded
        elif a == '-d':
            decoded = authcode(b, 'DECODE', expiry=ex)
            print decoded
# okay decompiling D:\reverse300.pyc
```


发现关键在于 len(sys.argv)，上网查了一下，发现sys.argv表示命令行参数，

sys.argv[0]是代表当前所执行的脚本

sys.argv[1] 脚本第一个参数

所以len(sys.argv)<3 代表当前脚本的参数小于3。

若执行命令为python hello.py "111"

则len(sys.argv)==2，hello.py也是其中的一个参数

往下看可以发下，当sys.argv[1]也就是第二个参数a为 -d 时，执行decode(解码)操作，还少了一个参数，这时想起还有个线索文件hint.txt，打开是一串字符串，复制下来就是第三个参数b,代码执行

```
D:\Python\Scripts>python2 D:\re300.py -d 8d2cP9kj6Pmz5VWhEpz71IjAOL7RQZ40Ay1jbcS1LdDUwbAxog
Wh2i8wFzxxtA2je1z/axDTApIz9W5qxA==


D:\Python\Scripts>
```

什么都没有，再回到源代码，研究了一下，我最后把main函数上的判断改成这样，

```
    if operation == 'DECODE':
        return result[26:]
        #if not result[0:10].isdigit() or int(result[0:10]) == 0 or
int(result[0:10]) - int(time()) > 0:
            #if result[10:26] == md5(result[26:].encode('utf-8') +
keyb).hexdigest()[0:16]:
                #return result[26:]
#           return ''
#       else:
#           return ''
#   else:
#       return keyc + base64.b64encode(result)


if __name__ == '__main__':
    if len(sys.argv) < 3:
```


把if判断下的都注释掉，再把return result[26:]提取出来，输出结果

```
D:\Python\Scripts>python2 D:\re300.py -d 8d2cP9kj6Pmz5VWhEpz71IjA0L7RQZ4OAy1jbcS1LdDUwbAxog
Wh2i8wFzxxtA2je1z/axDTApIz9W5qxA==
BDCTF{2u0_chu_14i_d3_5hi_h3n74i}

D:\Python\Scripts>
```