2021强网杯青少赛（qwtac）楼上大佬ddw WriteUp

原创

末小心 于 2021-10-12 20:40:17 发布 104 收藏
分类专栏： CTFwp 安全工具 文章标签： 算法 网络安全

本文链接：https://blog.csdn.net/Moxin1044/article/details/120731533

楼上大佬ddw战队WRITEUP

- 战队信息

战队名称：楼上大佬ddw

战队排名：24

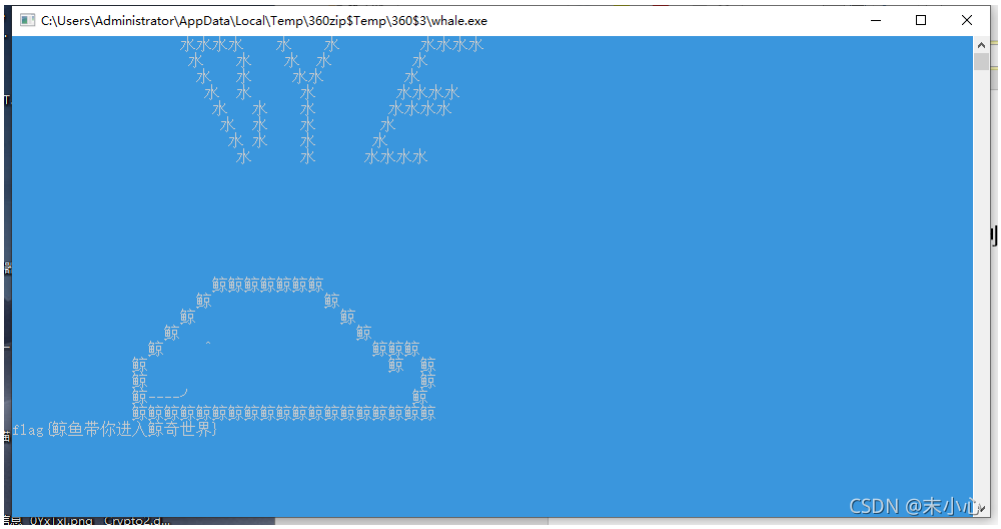
- 解题情况

- 解题过程

01 签到

操作内容：

下载附件，打开运行拿到flag



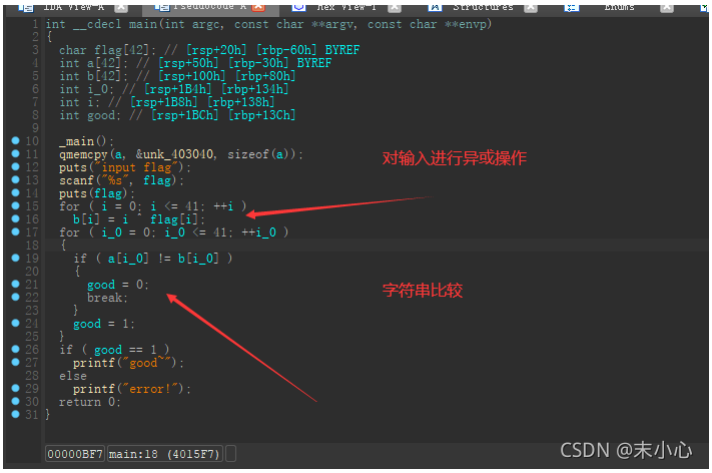
如该题使用自己编写的脚本代码请详细写出，不允许截图

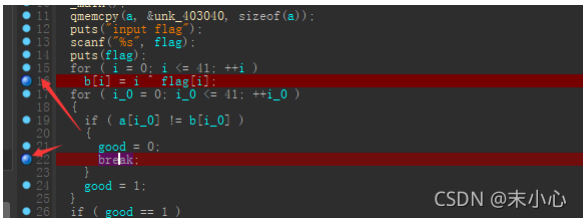
值：flag{鲸鱼带你进入鲸奇世界}

02 Lihua's for

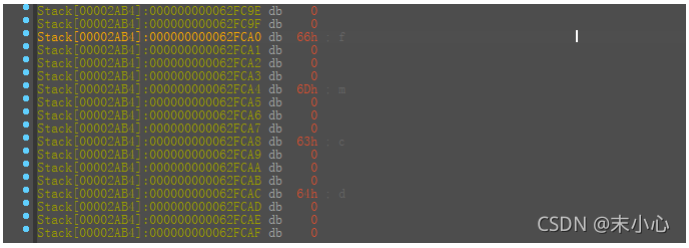
操作内容：

RE 扔进IDA

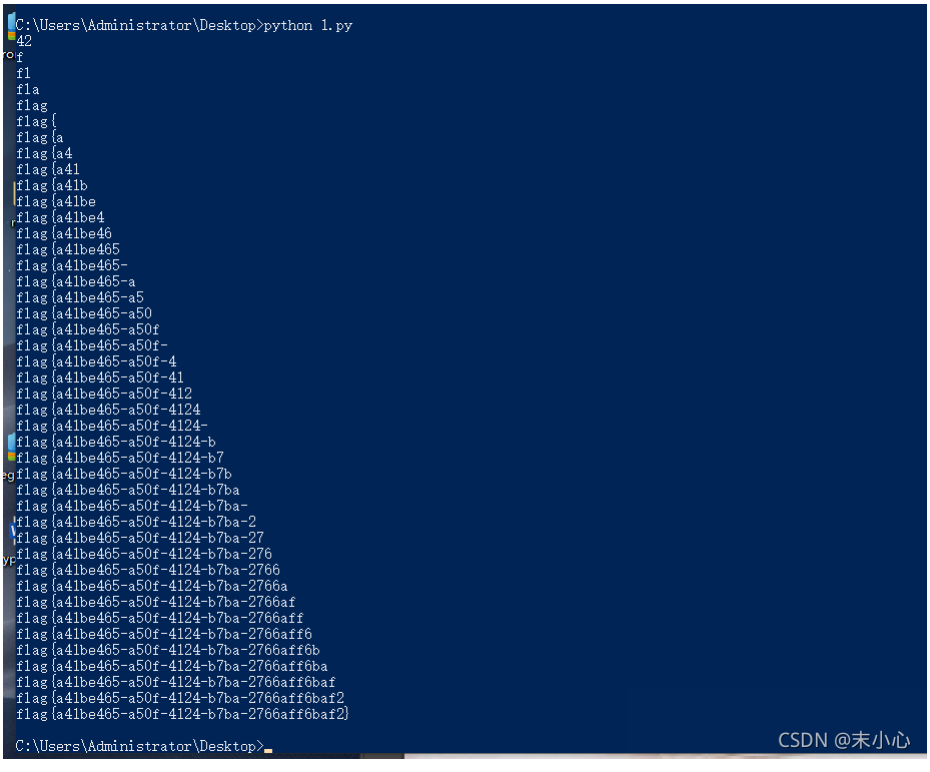




下断点，动态调试，然后取a数组的数据



写解密代码



如该题使用自己编写的脚本代码请详细写出，不允许截图

```
less=
[0x66,0x6D,0x63,0x64,0x7f,0x64,0x32,0x36,0x6a,0x6c,0x3e,0x3d,0x39,0x20,0x6f,0x3a,0x20,0x77,0x3f,0x27,0x25,0x27,0x22,0x3a,0x7a,0x2e,0x78,0x7a,0x31,0x2f,0x29,0x29,0x16,0x40,0x44,0x45,0x12,0x47,0x47,0x41,0x1a,0x54]
print(len(less))
sas=""
for i in range(len(less)):
    sas+=chr(l[i]^i)
print(sas)
```

值：flag{a41be465-a50f-4124-b7ba-2766aff6baf2}

02 Crypto2

操作内容：

两组数中e相同，n，c不同，求出n1与n2的最大公因数即为p，之后就可以得到q和d，从而求解m，该题中的flag为两部分，要依次求出再拼接。

如该题使用自己编写的脚本代码请详细写出，不允许截图

```
import gmpy2
import binascii
e = 65537
n1 =
20663949646446787716947370247427064802032290773674573417491154934657966734874241036307633567695175131014840617208051931753476223149652427133485160771068994073566431652969243962
c1 =
20522772249591436865905796103232542494211695376973377722875606678999899690405480809231671346489821878050354380591999359607958884836644739522072985041962038305432084772291621776
n2 =
23260834024376640092536888922041147168387702014814910549469730354688848760379274203088716649609675449936234732528778557041701524981200368996310064584479657042098426164366286701
c2 =
18715009944766815149492560645051626329204114049927707292306481018724323433701970253541495090244787378826569549885480491764057526828531429033378143426272248940256432423939977805
p = gmpy2.gcd(n1,n2)
q1 = n1// p
q2 = n2//p
phi1= (p-1)*(q1-1)
phi2= (p-1)*(q2-1)
d1= gmpy2.invert(e,phi1)
d2= gmpy2.invert(e,phi2)
m1= gmpy2.powmod(c1,d1,n1)
m2= gmpy2.powmod(c2,d2,n2)
print(binascii.unhexlify(hex(m1)[2:]))
print(binascii.unhexlify(hex(m2)[2:]))
```

值：flag{afb1e6f2-9acb-efde-ad7c-246a99d8f1fd}

04 Crypto1

操作内容：

e1=49

e2=35

首先想到了共模攻击，但是不满足gcd条件(e1，e2互质)想到可以将49,35同时除以7，然后进行低加密指数攻击即可

如该题使用自己编写的脚本代码请详细写出，不允许截图

```
import gmpy2
from Crypto.Util.number import long_to_bytes
n=967226697499512129137566782343586511841340684078124704344359166031568189175458414397790319438006342500321067641548043099355576785128586300482122046964714877621607449248380107
c1=667381132234472214300097399149483032610028115530643075329267880246943198469093408069827083479046884206716564105548523407323958180070636484785930716659362778369880505261880641
c2=883309491466510425173376537408103851873616890125017927999008732799787360357906592110010479373721512194852701702296764290663273213631375027723776191071091545973355142165325998
e1=7
e2=5
def ext_euclid(a,b):
    if b == 0:
        return 1,0,a
    else:
        x,y,q = ext_euclid(b, a % b) #gcd(a,b)=gcd(b,a%b)
        x,y = y, (x-(a//b)*y)
        return x,y,q
def same_mod(n,e1,e2,c1,c2):
    s,t,q = ext_euclid(e1,e2)
    m=(gmpy2.powmod(c1,s,n)*gmpy2.powmod(c2,t,n)) % n # 大数运算
    flag = m
    # 爆破
    i = 0
    while 1:
        if gmpy2.iroot(m + i * n,7)[1]==True:
            print(long_to_bytes(gmpy2.iroot(m+i*n,7)[0]))
            break
        i += 1
    return flag
if __name__ == '__main__':
    same_mod(n,e1,e2,c1,c2)
```

值：

flag{8ac9f9e3-82ba-ff7e-ac7b-235a02d891ef}

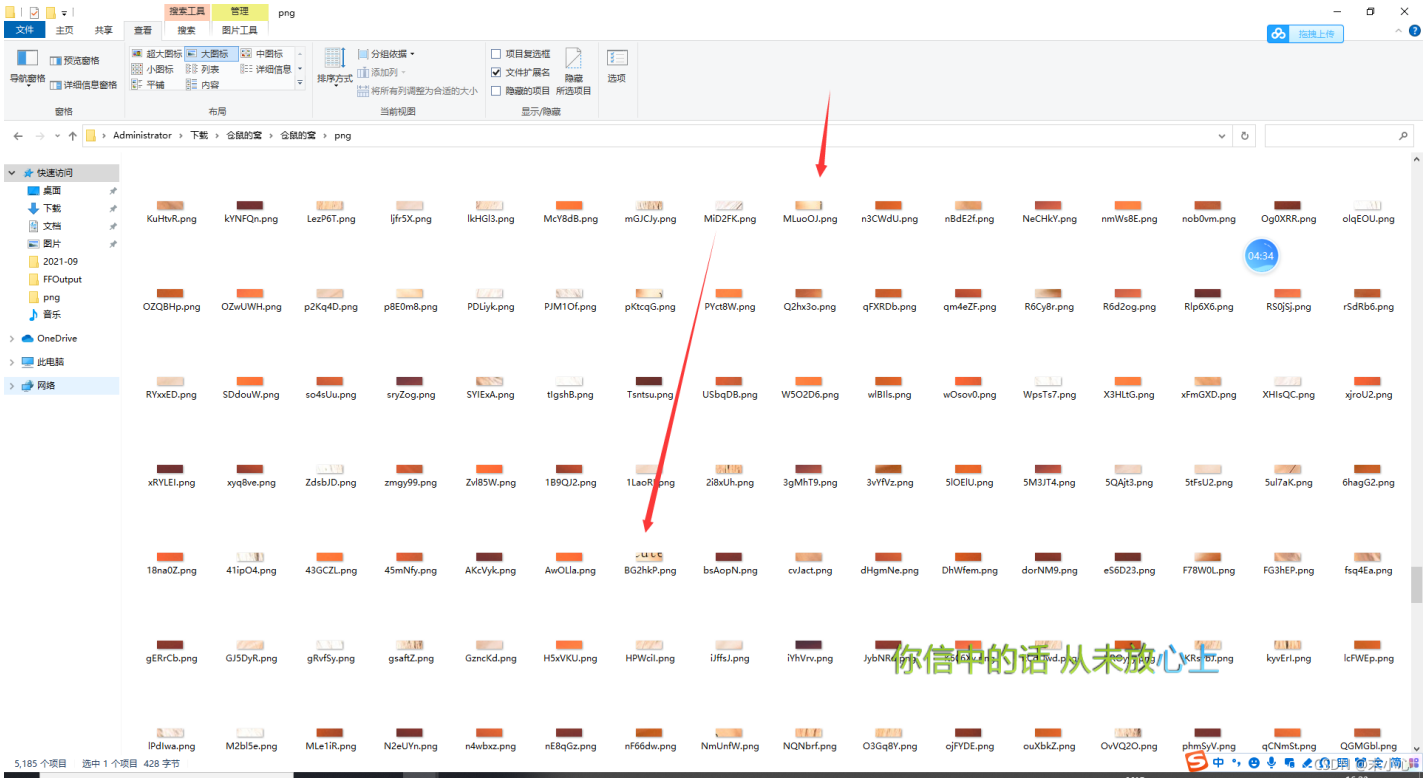
06 拼图

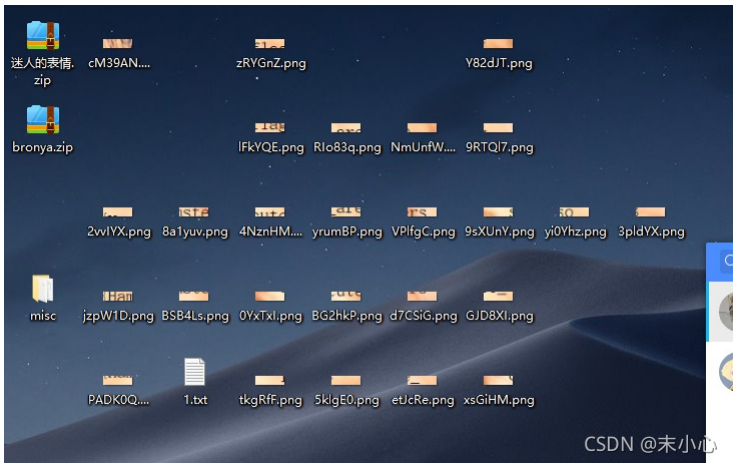
操作内容：

拿到附件先解压，原图先不用看，直接看分割后的。

因为分割图片，所以我们可以通过时间排列 （分割文件的保存的时间排序。）

可以看到分割的顺序是从左到右依次分割。整理含字符串的图片。





整理出来有如下片段:

能拼成这样子，提交该flag: flag{Hamsters_are_so_cute} 都提示不对。

发现了这个东西，可能是个感叹号什么的？还是i呢？我想了一下，按照英语语法，i有点不合适。

可能是感叹号，那就交了试试flag{Hamsters_are_so_cute!} 发现不对

但是我发现文件中还有很多没有拼接，比如

试试多加几个!吧，提交发现3个!就对了得到flag

flag值：flag{Hamsters_are_so_cute!!!}