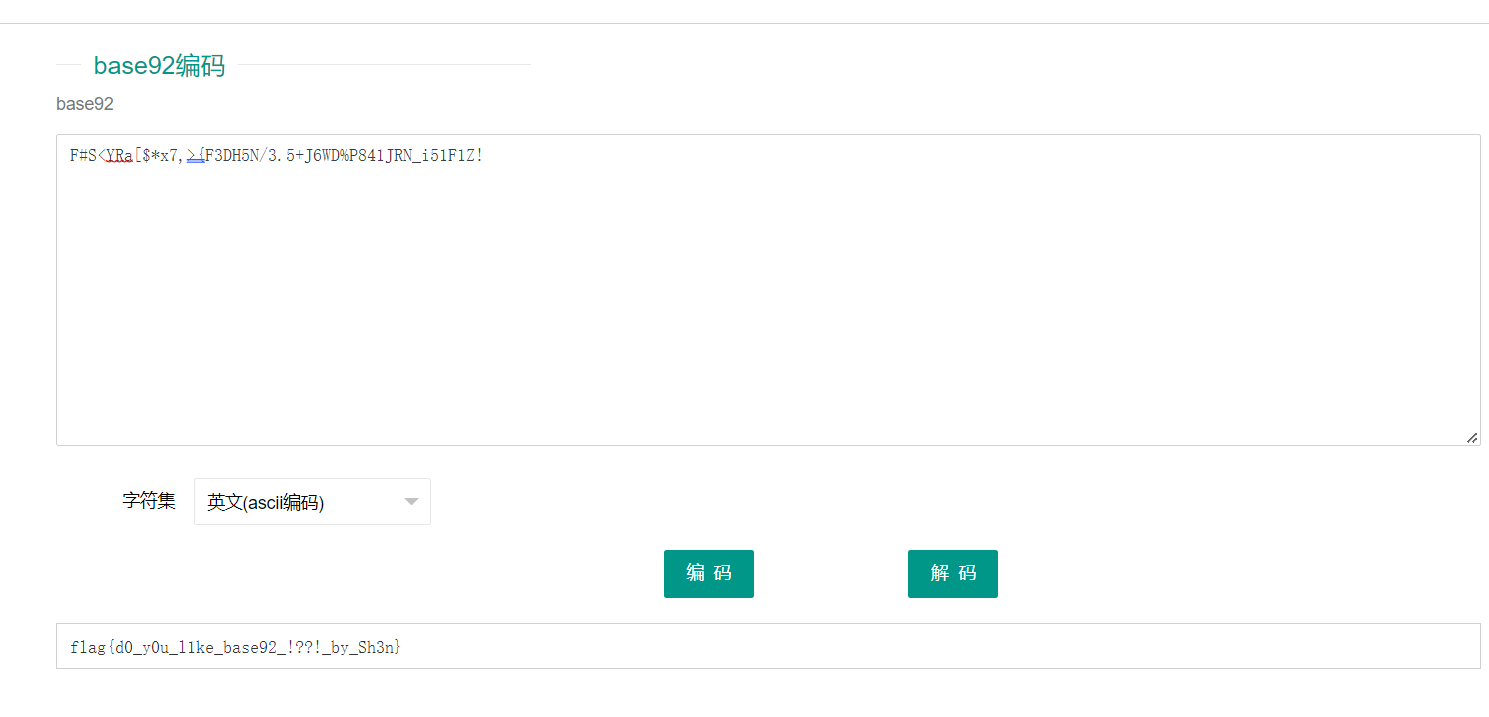
**Misc 简单的base编码**

拿到编码一直用base64解，解到最后有一串字符F#S<YRa[$\*x7,>{F3DH5N/3.5+J6WD%P841JRN\_i51F1Z!

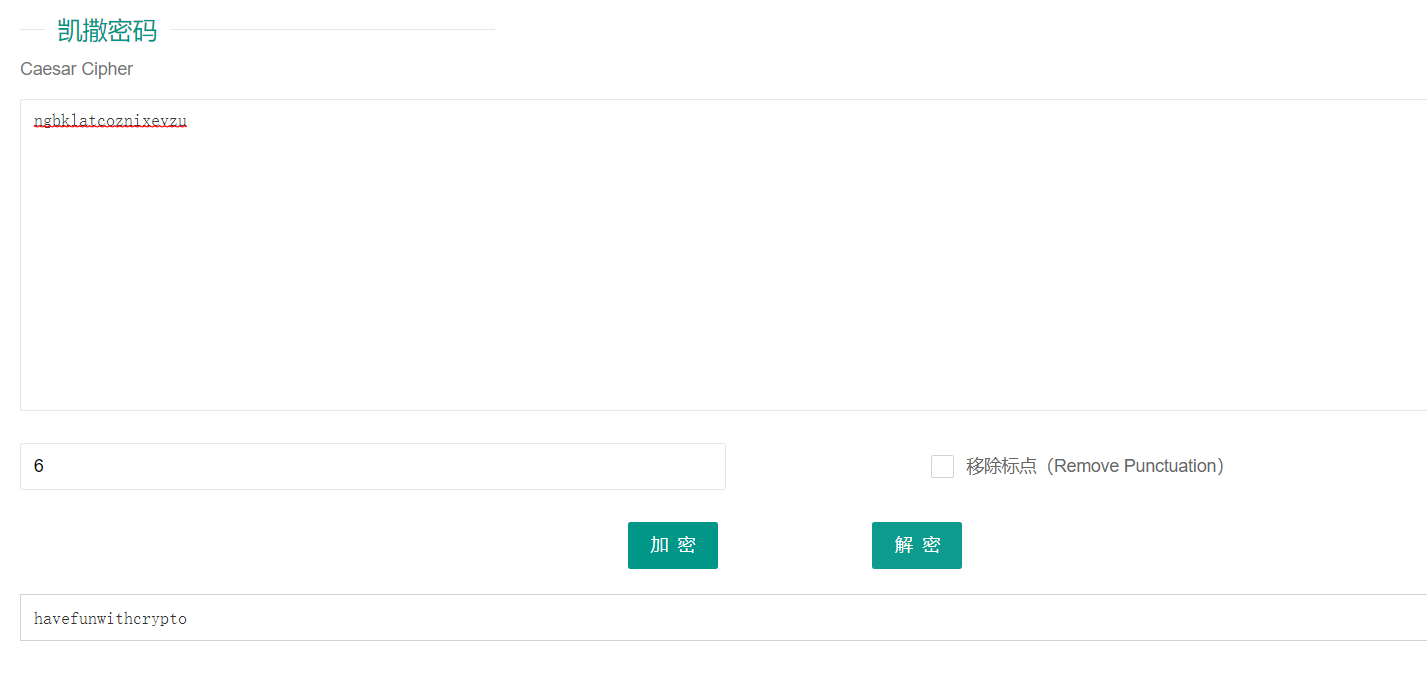
这个拿base92解



**Misc凯撒大帝在培根里藏了什么**

拿到一串AB的编码，用培根密码解得到ngbklatcoznixevzu

用凯撒解密，密钥是6，



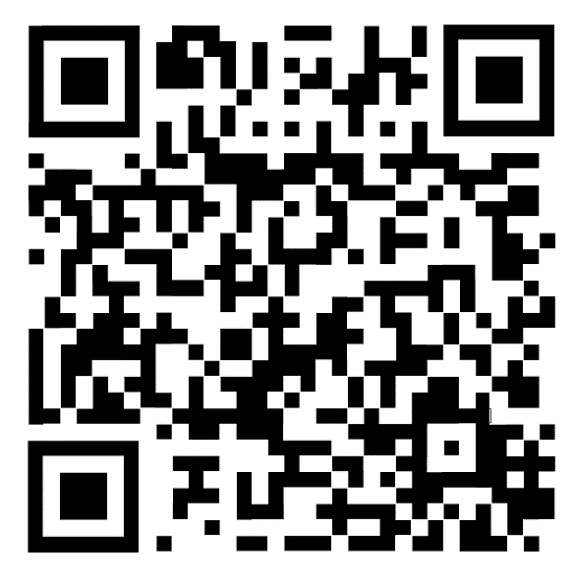
flag{havefunwithcrypto}

**Misc QR1**

给了一个二维码的点阵图，手动拼画。

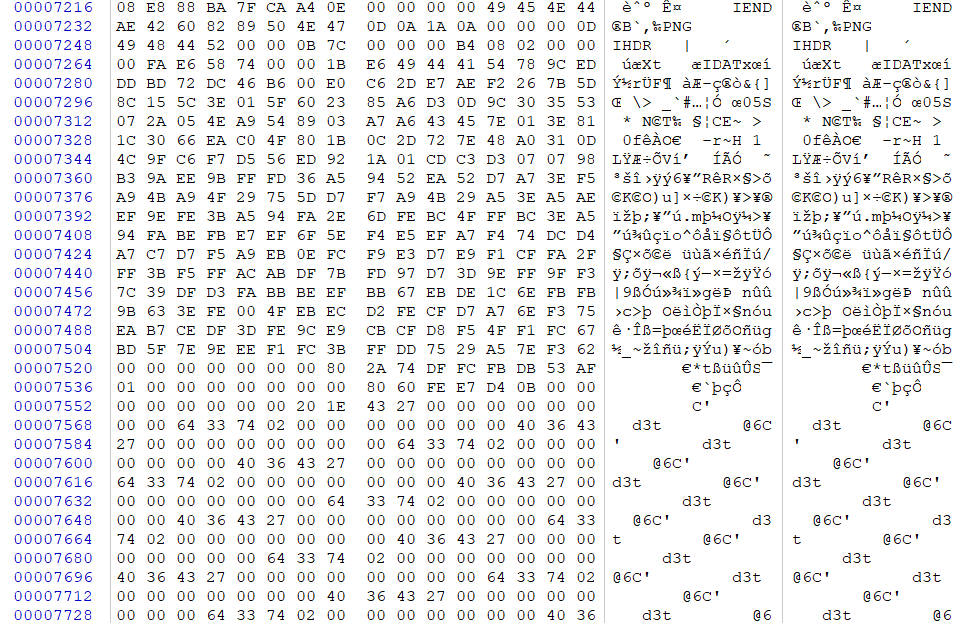
直接上图

flag{AHA\_U\_Kn0w\_QR\_c0d3\_312468ed-ea59-4fe9-9cd2-b5e9d8b39498}



**Misc BotW**

题目给了一个弓箭的图片，首先丢尽winhex里面查看，然后用敏锐的眼睛发现



有另一个png，然后把数据复制下来保存为png

图片为



应该是某种奇怪的文字，与题目有关，查了一下Botw是个游戏，方向大概就是没错了。



对着翻译得到结果：

YOU GET THE FLAG! FLAG IS CANZELDAGETDUPLEXBOW.

flag{CANZELDAGETDUPLEXBOW}

**Crypto 初识RSA**

题目给出了pq=p(q-1)，qp=q(p-1)，两式相减得到p和q的差，pq-qp=p-q=x，又知道p\*q=n，解一元二次方程，得到p

x = 12939817320805560266981907982643983551550650701532023767279797375464886357450821336932430580340492716021165527290784243425867512346446496505607272539456363654607452963763831335832900475965743147706780868073560077699512439860241198179356729243686532528394746382137060414657776631623019985510554481583115935078

n = 10466186506773626671397261081802640650185744558208505628349249045496105597268556020207175016523119333667851114848452038431498926527983706092607207796937431312520131882751891731564121558651246025754915145600686076505962750195353958781726515647847167067621799990588328894365930423844435964506372428647802381074584935050067254029262890188260006596141011807724688556673520261743199388391094490191001701011230322653422314758778116196105077883955436582364267530633358016652912054880813710531145973799193443828969535902856467548523653920307742364119002349899553478815101092655897400295925170383678499125295006364960124859003

b = x

t = b\*b+4\*n

l = 0

r = n

mid = 0

while l+1<r:

    mid = (l+r)>>1

    if (mid\*mid<=t):

        l=mid

    else:

        r=mid

if (l\*l==t):

    print("YES")

    q=(-b+l)//2

    print('q=',q)

    print('p=',x+q)

解一元二次方程，python代码

带入p，q，解rsa，得到m

#python3

import gmpy2 as gp

import binascii

p =  gp.mpz(96038853037549564062764850442359894131887890858233489654286089903584525731057752315254523619433946100262433785995278342631827079147892495663178701536519700102765260114095129653169819831610648580293168637786843648315377042167032673045632934737617102728113023442076467143323716796385555641601627531295223680679)

q =  gp.mpz(108978670358355124329746758425003877683438541559765513421565887279049412088508573652186954199774438816283599313286062586057694591494338992168785974075976063757372713077858960989002720307576391727999949505860403726014889482027273871224989663981303635256507769824213527557981493428008575627112182012878339615757)

e =  gp.mpz(65537)

c =  gp.mpz(8722269075970644434253339592758512788160408912707387632591552130175707843950684315083250494010055435391879036285103810263591951437829414438640307561645721347859659807138051841516634704123100270651976676182059252251162982609391666023674158274992400910869692389001622774140191223807887675081808561012755545464977015973615407965906513878979919700065923364884766974187303774330319143647840846354404070430118235352622445115153298578370521811697710289716188726587743282814946239856766713516166990341116198180068191759095913957606379780234116317390622824096667107736103270907349927467971817639795094030622157581511033950777)

n = p\*q

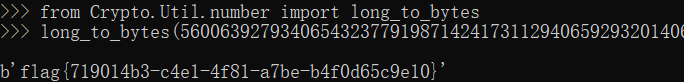
phi = (p-1) \* (q-1)

d = gp.invert(e, phi)

m = pow(c, d, n)

print(m)

M= 56006392793406543237791987142417311294065929320140616629889216443512551772328106378173698666811961469



Long\_to\_bytes将m转成字符串

**Crypto Babyrsa**

题目要完成三个任务：1、爆破验证sha256；2、解出简单的rsa，n很小；3、解出30次n很大得rsa。

由于有时间限制，所以必须用pwntool。

1、爆破验证sha256：直接枚举前4位，与给定的sha256值比对

2、对n直接分解，直接找n的因数

3、题目给出的p，q相近，因为q是p的下一个质数，所以可以对n开方然后找下一个质数。得到p、q。

得到p，q直接套上rsa解密模板

**Exp.py**

import imp

from os import remove

import babyrsa2

import gmpy2 as gp

import binascii

from pwn import \*

from hashlib import sha256

import string

from Crypto.Util.number import \*

sh = remote("43.248.98.206",10098)

st=string.ascii\_letters+string.digits

line=sh.recvline(keepends=False)

print(line.decode())

#line=str(line,encoding='utf-8')

line=line.decode()

s=line[14:26]

#t='sha256(XXXX + Sp3yZHH6cSbb) == 21c9b17f54a0c10bf83665b04e3e654baeba28ce28543991faef8406aadc00c2'

shaa=line[-64:]

out = ''

line=sh.recvline(keepends=True)

print(line.decode())

for i in st:

    for j in st:

        for k in st:

            for l in st:

                t=i+j+k+l+s

                sha=sha256(t.encode()).hexdigest()

                if (sha == shaa):

                    out = i+j+k+l

out+='\n'

sh.send(out.encode())

print(out)

input=sh.recvuntil("n=".encode())

print(input.decode())

n = int(sh.recvline(keepends=False).decode(),16)

print(n)

input=sh.recvuntil('e='.encode())

print(input.decode())

e = int(sh.recvline(keepends=False).decode(),16)

print(e)

input=sh.recvuntil('c='.encode())

print(input.decode())

c = int(sh.recvline(keepends=False).decode(),16)

print(c)

p=0

q=0

for i in range(2,n//2):

    if n%i==0:

        q=i

        p=n//i

        break

def rsa(p,q,e,c,n):

    p =  gp.mpz(p)

    q =  gp.mpz(q)

    e =  gp.mpz(e)

    c =  gp.mpz(c)

    n = p\*q

    phi = (p-1) \* (q-1)

    d = gp.invert(e, phi)

    m = pow(c, d, n)

    return(hex(m))

m=rsa(p,q,e,c,n)

out=str(m)+'\n'

sh.send(out.encode())

print(out)

for i in range(30):

    input=sh.recvuntil("n=".encode())

    print(input.decode())

    n = int(sh.recvline(keepends=False).decode(),16)

    print(n)

    input=sh.recvuntil('e='.encode())

    print(input.decode())

    e = int(sh.recvline(keepends=False).decode(),16)

    print(e)

    input=sh.recvuntil('c='.encode())

    print(input.decode())

    c = int(sh.recvline(keepends=False).decode(),16)

    print(c)

    temp=gp.iroot(n,2)[0]

    p=gp.next\_prime(temp)

    q=n//p

    m=rsa(p,q,e,c,n)

    out=str(m)+'\n'

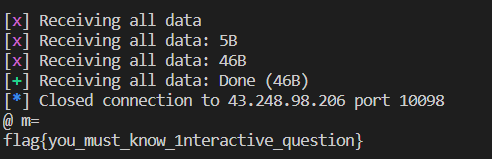
    sh.sendline(out.encode())

    print(out)

line=sh.recvall()

print(line.decode())

**flag{you\_must\_know\_1nteractive\_question}**



**easyEZbaby\_app**

安卓逆向

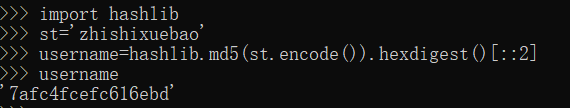
dex2jar反编译apk

用ju-gui-windows查看jar文件

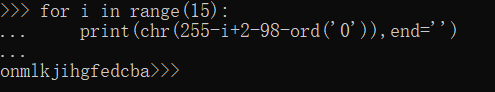
发现flag是username+password



逆向username



逆向password



username=7afc4fcefc616ebd

password=onmlkjihgfedcba

flag{7afc4fcefc616ebdonmlkjihgfedcba}

**简单的LFSR**

flag{48eea154e05fefe7a657680a080ada3b}

已知给定明文，求lfsr的反馈系数，实际上明文长度是3m，所以取前2m，解线性方程组。

即A是一个128\*128的矩阵，A[i][j]=S[i+j]，S是给定的明文，B是一个向量，B[i]=S[i+128]

丢到sagemath解AX=B，X=A.solve\_right(B)。得到一串01序列

按照题目意思，把01串取md5即为flag。