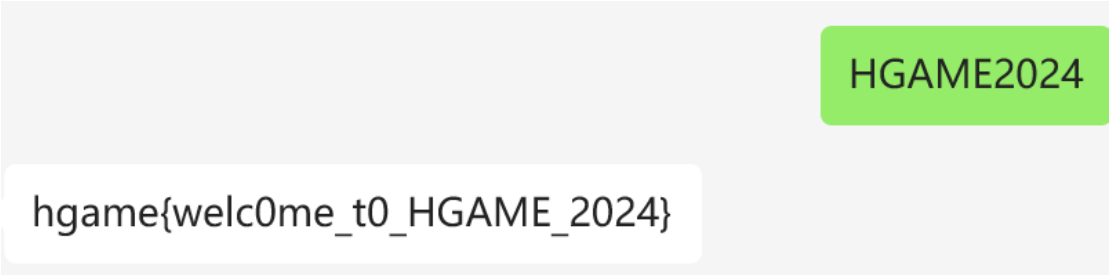
MISC

Signin

直接 word 里扯一下就好了

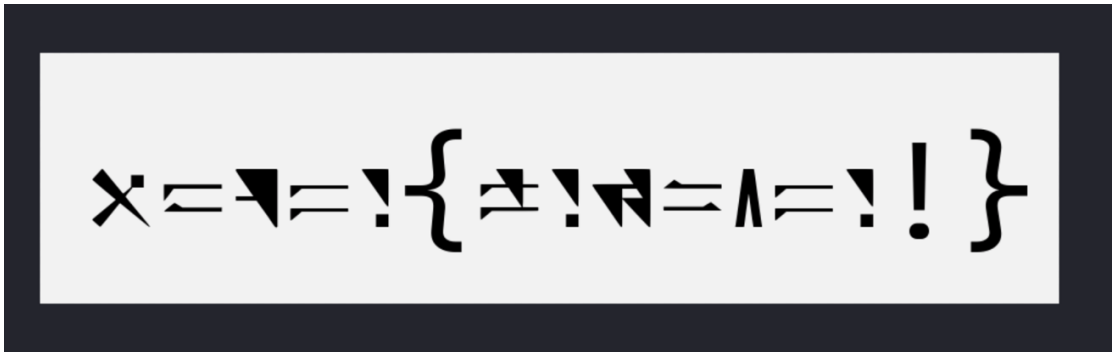
签到



添加公众号

来自星尘的问候

题目中说有六位弱加密，猜想可能是藏了压缩包，但查看内码后没有
于是考虑 steghide
Stegseek 一把梭



得到这张图片

结合题目网上找来自星尘游戏字体，对比得 flag
Hgame{welc0me!}

simple_attack

解压 src 后得到一张图和一个有密码的压缩包，观察知这张图加密压缩包内的图为同一张，想到用 archpr 明文攻击，解密成功，有如下 txt，是某 png 的内码的 base64 用 CyberChef 转换为图片

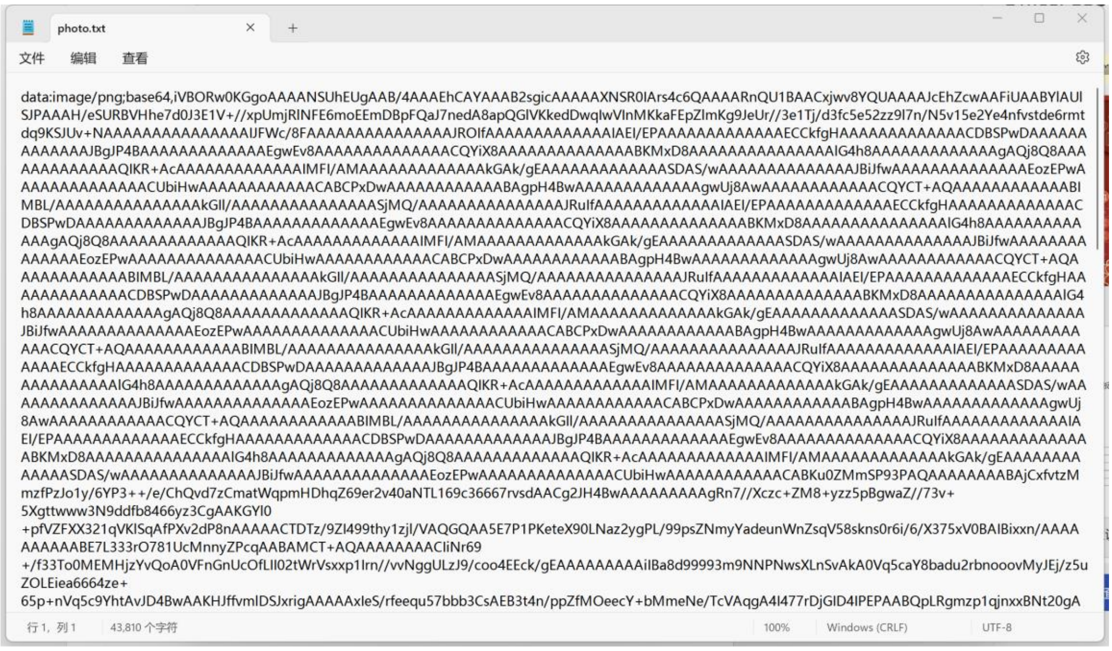
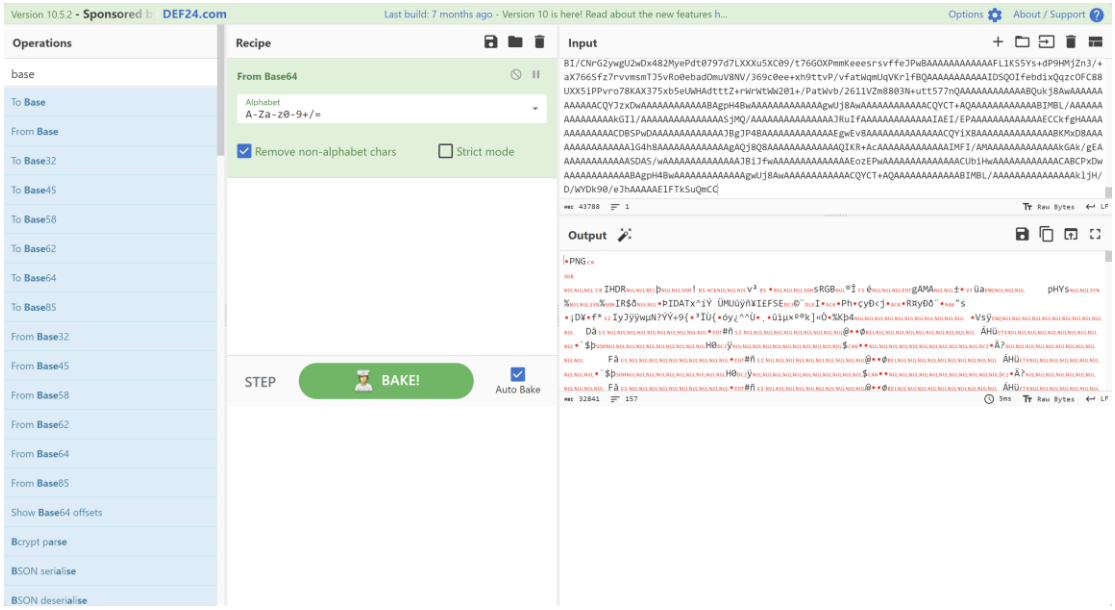


photo.txt
文件 编辑 查看

SJPAAAH/eSURBVHhe7d0J3E1V+//xpUmjRlNFE6moEEmDBpFQaJ7nedA8apQGlVKkedDwqlwVlnMKkaFEpZlmKg9JeUr//3e1Tj/d3fc5e52zz9l7n/N5v15e2Ye4nfvstde6rmt
dq9KSJUv+NAAAAAAAAAAAAAAAAAAIJFWc/8FAAAAAAAAAAAAAAAAJROIfAAAAAAAAAAAAAAAAIAEI/EPAAAAAAAAAAAAAAAAECCkfgHAAAAAAAAAAAAAAAACDBSPwDAAAAAAA
AAAAAAAJBgJP4BAAAAAAAAAAAAAAEgwEv8AAAAAAAAAAAAAAAACQYiX8AAAAAAAAAAAAAAAABKMxD8AAAAAAAAAAAAAAAAIG4h8AAAAAAAAAAAAAAgAQj8Q8AAA
AAAAAAAAAAQIKR+AcAAAAAAAAAAAAAAAIMFI/AMAAAAAAAAAAAAAAkGAk/gEAAAAAAAAAAAAAASDAS/wAAAAAAAAAAAAAAAJBiJfwAAAAAAAAAAAAAAAEozEPwA
AAAAAAAAAAAAACUbiHwAAAAAAAAAAAAAACABCPxDwAAAAAAAAAAAAAABAgpH4BwAAAAAAAAAAAAAAgwUj8AwAAAAAAAAAAAAAACQYCT+AQAAAAAAAAAAAAAABI
MBL/AAAAAAAAAAAAAAAAkGIl/AAAAAAAAAAAAAAAASjMQ/AAAAAAAAAAAAAAAJRulfAAAAAAAAAAAAAAAIAEI/EPAAAAAAAAAAAAAAAECCkfgHAAAAAAAAAAAAAAAC
DBSPwDAAAAAAAAAAAAAAJBgJP4BAAAAAAAAAAAAAAEgwEv8AAAAAAAAAAAAAAAACQYiX8AAAAAAAAAAAAAAAABKMxD8AAAAAAAAAAAAAAAAIG4h8AAAAAAAAAAA
AAAgAQj8Q8AAAAAAAAAAAAAAQIKR+AcAAAAAAAAAAAAAAIMFI/AMAAAAAAAAAAAAAAkGAk/gEAAAAAAAAAAAAAASDAS/wAAAAAAAAAAAAAAJBiJfwAAAAAAAAA
AAAAAAEozEPwAAAAAAAAAAAAAACUbiHwAAAAAAAAAAAAAACABCPxDwAAAAAAAAAAAAABAgpH4BwAAAAAAAAAAAAAAgwUj8AwAAAAAAAAAAAAAACQYCT+AQA
AAAAAAAAAAABIMBL/AAAAAAAAAAAAAAAAkGIl/AAAAAAAAAAAAAAAASjMQ/AAAAAAAAAAAAAAAJRulfAAAAAAAAAAAAAAAIAEI/EPAAAAAAAAAAAAAAAECCkfgHAA
AAAAAAAAAAACDBSPwDAAAAAAAAAAAAAAJBgJP4BAAAAAAAAAAAAAAEgwEv8AAAAAAAAAAAAAAAACQYiX8AAAAAAAAAAAAAAAABKMxD8AAAAAAAAAAAAAAAAIG4
h8AAAAAAAAAAAAAgAQj8Q8AAAAAAAAAAAAAAQIKR+AcAAAAAAAAAAAAAAIMFI/AMAAAAAAAAAAAAAAkGAk/gEAAAAAAAAAAAAAASDAS/wAAAAAAAAAAAAAA
JBiJfwAAAAAAAAAAAAAAAEozEPwAAAAAAAAAAAAAACUbiHwAAAAAAAAAAAAAACABCPxDwAAAAAAAAAAAAABAgpH4BwAAAAAAAAAAAAAAgwUj8AwAAAAAAAAAAAA
AAACQYCT+AQAAAAAAAAAAAAABIMBL/AAAAAAAAAAAAAAAAkGIl/AAAAAAAAAAAAAAAASjMQ/AAAAAAAAAAAAAAAJRulfAAAAAAAAAAAAAAAIAEI/EPAAAAAAAAAA
AAAAECCkfgHAAAAAAAAAAAAAACDBSPwDAAAAAAAAAAAAAAJBgJP4BAAAAAAAAAAAAAAEgwEv8AAAAAAAAAAAAAAAACQYiX8AAAAAAAAAAAAAAAABKMxD8AAAAAA
AAAAAAAAAAIG4h8AAAAAAAAAAAAAgAQj8Q8AAAAAAAAAAAAAQIKR+AcAAAAAAAAAAAAAIMFI/AMAAAAAAAAAAAAAkGAk/gEAAAAAAAAAAAAAASDAS/wAA
AAAAAAAAAAAAJBiJfwAAAAAAAAAAAAAAAEozEPwAAAAAAAAAAAAAACUbiHwAAAAAAAAAAAAAACABCPxDwAAAAAAAAAAAAABAgpH4BwAAAAAAAAAAAAAAgwUj
8AwAAAAAAAAAAAAAACQYCT+AQAAAAAAAAAAAAABIMBL/AAAAAAAAAAAAAAAAkGIl/AAAAAAAAAAAAAAAASjMQ/AAAAAAAAAAAAAAAJRulfAAAAAAAAAAAAAAAIA
EI/EPAAAAAAAAAAAAAAAECCkfgHAAAAAAAAAAAAAACDBSPwDAAAAAAAAAAAAAAJBgJP4BAAAAAAAAAAAAAAEgwEv8AAAAAAAAAAAAAAAACQYiX8AAAAAAAAAAAAAAA
ABKMxD8AAAAAAAAAAAAAAAAIG4h8AAAAAAAAAAAAAgAQj8Q8AAAAAAAAAAAAAQIKR+AcAAAAAAAAAAAAAIMFI/AMAAAAAAAAAAAAAkGAk/gEAAAAAAAAA
AAAAASDAS/wAAAAAAAAAAAAAAJBiJfwAAAAAAAAAAAAAAEozEPwAAAAAAAAAAAAAACUbiHwAAAAAAAAAAAAAACABKu0ZMmSP93PAQAAAAAAAABAjCxfvtzM
mzfPzJo1y/6YP3++/e+x+eer++v5p7kLtyL1/Yad/6zjH4p9wAZJmpPXvzlw/a1V/9yBUCIRUBu3Svpb7Y/v1VWanfVbHYOQEu7F1GCeqdPDyb23N3A5Bq2c7/g37LRqaH8t0n2ZgwLCEmKjJ/E3SQNQrAxBqG0sDXOqBYOyRgKb9nTpdy8Xn9enGv0t1VOrAvzGIDbYXNkCoy8q2x1mg6N6L2rH5bjuZy+ofAIbL4cgJqjd58tQnExn4l4/xMcxFcD0t2ybgU4/AmSrFS/NNxnOT
mzfPzJo1y/6YP3++/e/ChQvd7zCmatWqpmDHDhqZ69er2v40aNTL169169c36667rvsdxpe+11wAAAXEWQB0IAcE3ZLyPkTOhbldXl7n/0K+LPtoXMdxIFYcSwxDSvCxx4c+ZM8+yzcz73Z/a+vvmyGBwEfVHvHyGzu//73/Y/dqgUlu0RAVqx5AAAAAAAASDAS/wAAAAAAAAAAAAAAJBiJfwAAAAAAAAAAAAAAAEozEPwAAAAAAAAAAAAAACUbiHwAAAAAAAAAAAAAACABKu0ZMmSP93PAQAAAAAAAABAjCxfvtzM
mzfPzJo1y/6YP3++/e/ChQvd7zCmatWqpmDHDhqZ69er2v40aNTL169169c36667rvsdxpe+11wAAAXEWQB0IAcE3ZLyPkTOhbldXl7n/0K+LPtoXMdxIFYcSwxDSvCxx4c+ZM8+yzcz73Z/a+vvmyGBwEfVHvHyGzu//73/Y/dqgUlu0RAVqx5AAAAAAASDAS/wAAAAAAAAAAAAAAJBiJfwAAAAAAAAAAAAAAAEozEPwAAAAAAAAAAAAACUbiHwAAAAAAAAAAAAAACABKu0ZMmSP93PAQAAAAAAAABAjCxfvtzM
mzfPzJo1y/6YP3++/e/ChQvd7zCmatWqpmDHDhqZ69er2v40aNTL169169c36667rvsdxpe+11wAAAXEWQB0IAcE3ZLyPkTOhbldXl7n/0K+LPtoXMdxIFYcSwxDSvCxx4c+ZM8+yzcz73Z/a+vvmyGBwEfVHvHyGzu//73/Y/dqgUlu0RAVqx5AAAAA

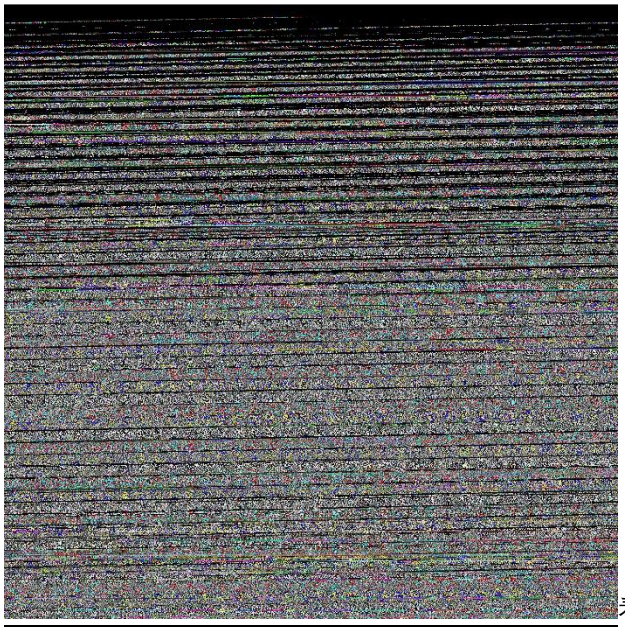
行 1，列 1　43,810 个字符　100%　Windows (CRLF)　UTF-8



得到 flag

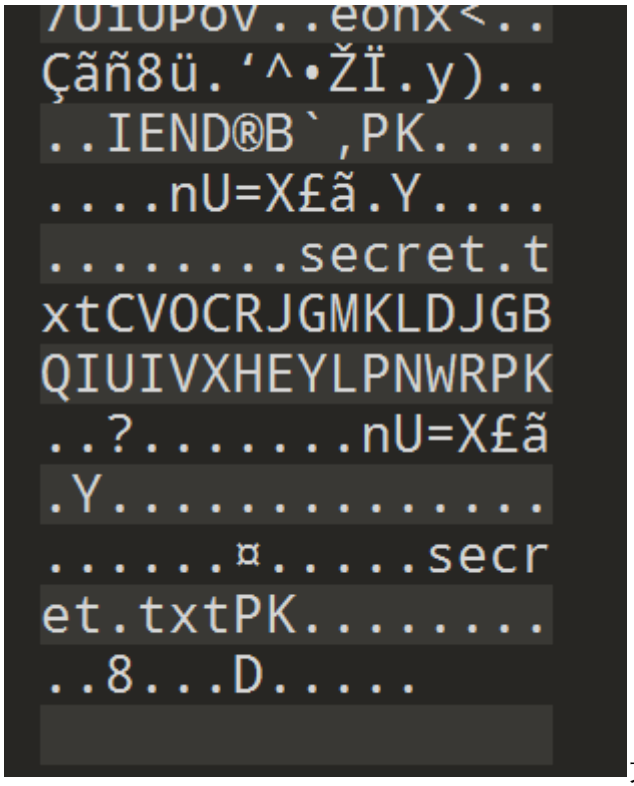
```
hgame{s1mple_attack_for_zip}
```

希儿希儿希尔

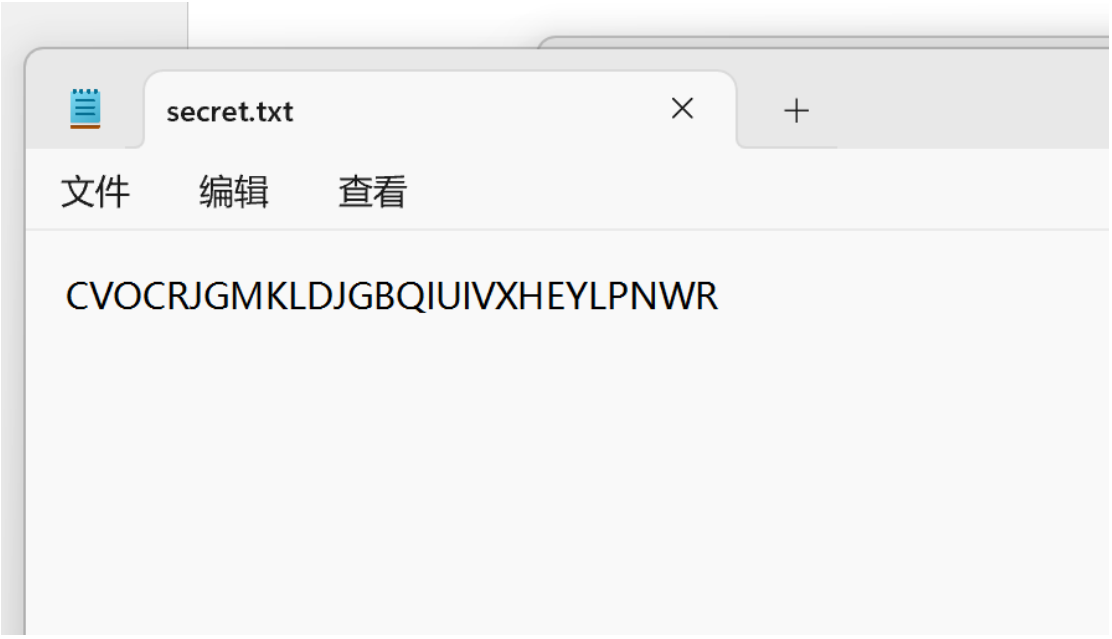


看着感觉是宽高被改，用脚本 crc 校验后

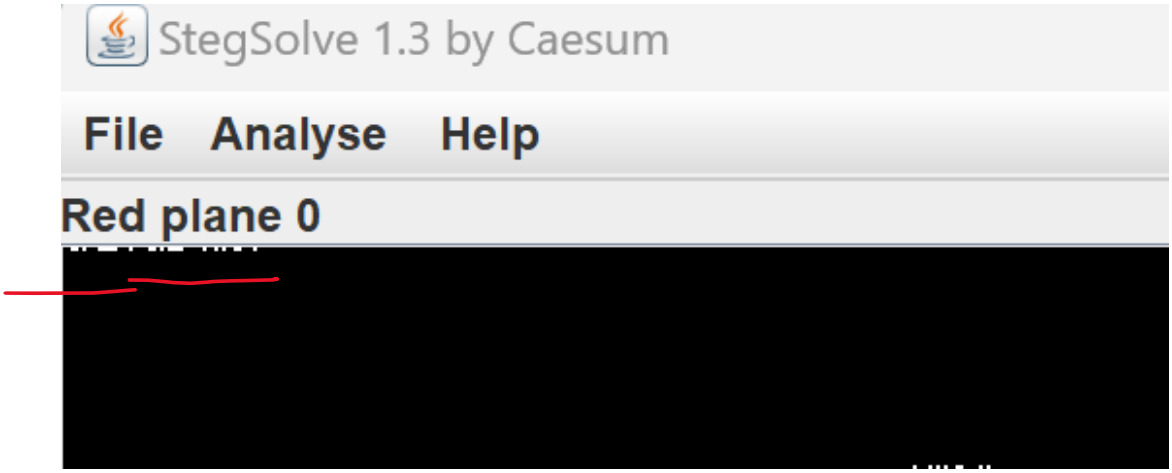
改回正常宽高



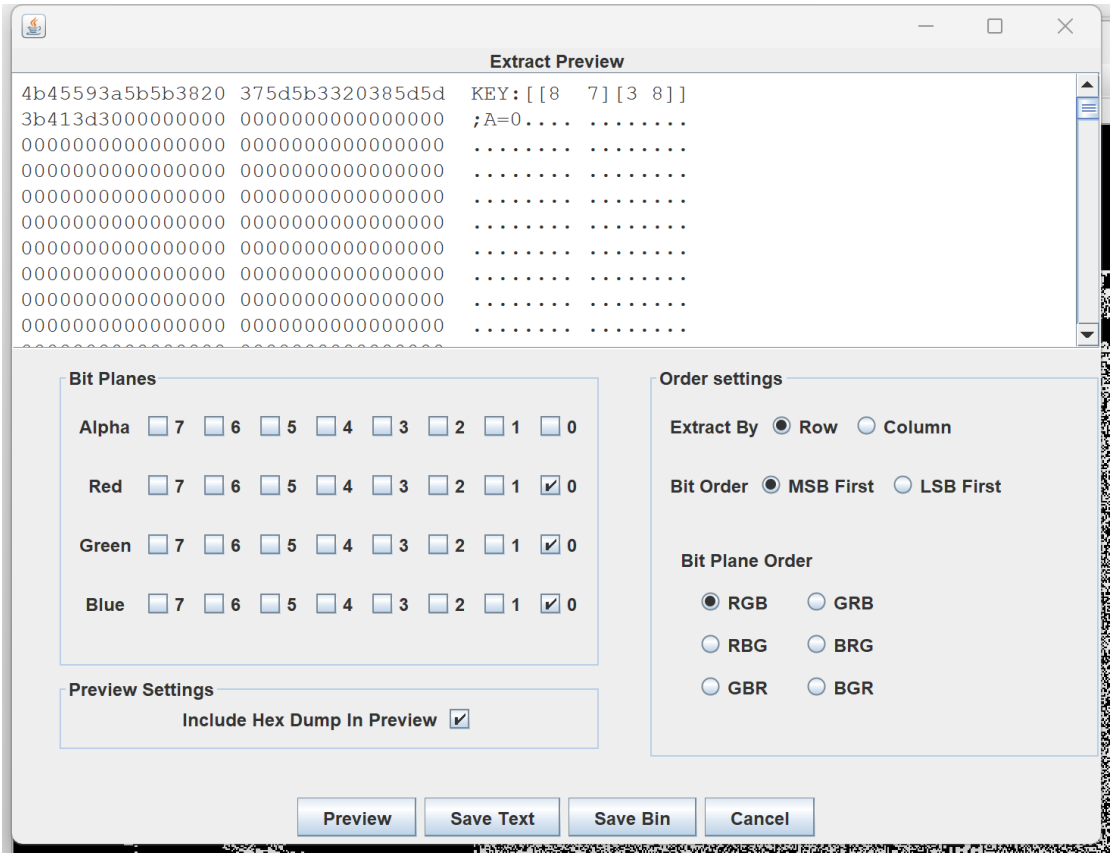
文件尾有一压缩包，解压得 txt



上交后发现不对；再读题发现应该是 hill（希尔密码）于是要找加密矩阵
用 stegsolve 观察图片



发现隐写痕迹



找到加密矩阵

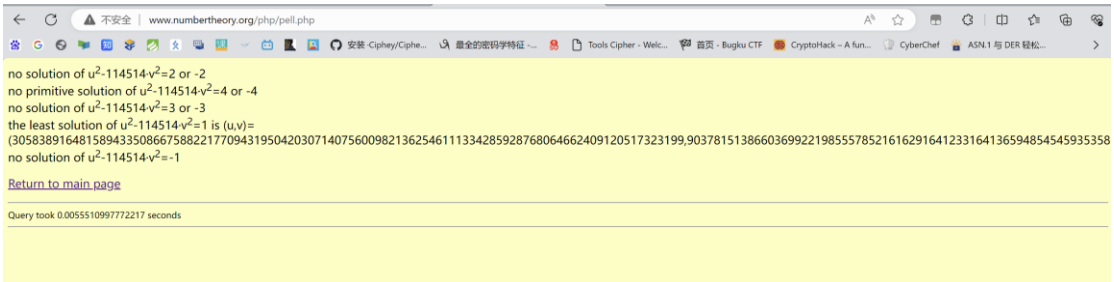
在线解密

CRYPTO

ezMath

```
D = 114514
assert x**2 - D * y**2 == 1
```

是 pell 方程，暴力求解未成功



在线网站求解得 y

```
from Crypto.Util.number import *
from Crypto.Cipher import AES
from math import *
def pad(x):
    return x+b'\x00'*(16-len(x)%16)
def decrypt(KEY):
    cipher=AES.new(KEY,AES.MODE_ECB)
```

```
    decrypted=cipher.decrypt(enc)
    return decrypted
D = 114514
y=903781513866036992219855578521616291641233164136594854545935358689571770257604962653352777910868
0
enc=b"\xce\xf1\x94\x84\xe9m\x88\x04\xcb\x9ad\x9e\x08b\xbf\x8b\xd3\r\xe2\x81\x17g\x9c\xd7\x10\x19\x1a\xa6\xc3\x9d\xde\xe7\xe0h\xed/\x00\x95tz)1\\\t8:\xb1,U\xfe\xdec\xf2h\xab`\xe5'\x93\xf8\xde\xb2\x9a\x9a"

x1=int(sqrt(y**2*D+1))
key=pad(long_to_bytes(y))[:16]
flag=decrypt(key)
print(f'flag={flag}')
#flag=b'hgame{G0od!_Yo3_k1ow_C0ntinued_Fra3ti0ns!!!!!!!}\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00'
```

ez_RSA

```
leak1=pow(p,q,n)
leak2=pow(q,p,n)
```

那么我们直接将 leak1、2 作为 p、q 求解即可

```
from Crypto.Util.number import *

q=14912717007361127196818257675129033155901844180572531042609541283758922767075754074392986585365039983910283843150720074472493965946320015801246967697998769641905090084279822566586181233111363289243874272420291641606026658159016906386768829928898573410412763223217565735269789838344132347745065817972772890866
9
p=11612299271467091538130991696749043648902000117288064416717991546702179489292797727208059664178556911913425903752238833519804315220615025910348557455881642474020473621555193348258394195999462535658120105453452939578174433863102142370317114645666343295584359854812259330878224522079201871650853849740257670946
1
c=10529481867532520034258056773864074017027019578041866245400647840230251661652999709715919620810933437191661180003295923273655675729588558899592524235622728816065501918076120812236580344991140980991532347991252705288633014913479970610056845543523591324177567061948922552275235486615514913932125436543991642607028689762693617305246716492783116813070355512606971626645594961850567586340389705821314842096465631886812281289843132258131809773797777049358789182212570606252509790830994263132020094153646296793522975632191912463919898988349282284972919932761952603379733234575351624039162440021940592552768579639977713099971

n=p*q
phi=(p-1)*(q-1)
```

```python
e=0x10001
d=inverse(e,phi)
print(long_to_bytes(pow(c,d,n)))
#b'hgame{F3rmat_l1tt1e_the0rem_is_th3_bas1s}'
```

ez_PRNG

是 LFSR

看博客[深入分析 CTF 中的 LFSR 类题目（一）-安全客 - 安全资讯平台]后写出解密代码

```python
ot=['1111110110111011110000101011010001000111111001111110100101000011
110111111100010000111110110111100001001000101101011110111100010010100
0000111111011011101010110101110000000111100001000111011110110110001 00
101100110100101110001010001101101110000001000100011110010101001011011 0
11110111001101100101111101101010101100001101100011101101111110011010 10
11110010110011000101101001010111001110100110011100001111011100000 1101
1100000011111000001000001011111000101101110011100110100000110111101 10
011000001101011111111010110011010111010101001000010011110110011110 110
10101111011101001101001011011111101001110100011010111110111100011 0011
11111001011000010010010010110101010111001010100110101010101111011101 0
0111011100001001011110101101011111110001111111110010000000000111001110 0
1000010111111101001110110001010011010011100100100011000110000011 01000
1110100100001011011111010110000000101000001110001011001010010001000011
00000010001001001001011101001111111101110010010010010111111110011100 00
1111101100011110011111001010010011000 10',
'00100000000001010111100001100011101111110111100010010011101010111001 01
10011001011110101100011101010000000110000011000000000110000000110101111 1
11011100100110111011010000100011111000111001000101001110010110010001 0
0011001010101111001110100001111110110101100001111000110101111100011 01
1100001100011001110010010110011110000010010010111100101110111000101 10
1111111101101010001011101100001001010111011010000011010000001000101010
000101111010010000110000000000111010010101010111101101011111101100100 01
0100010001100110010101011011000101001000101011011101101111111010111001
11001101111111111101001110111101001001111001111111101001100111111110110 0
0100011110001011100010111100001101101111111011101011101001110000111000
0101011011110001100101101001101011100011010110011010001110110101110 10
0011101100010011011000110011010101011001001101111000011111010011110 11
1000010001000011110001011100001000001000111111011010000100011011010 01
0011011001011011101001111110101111000001110101010011010101111000011 01
0111011101101011011000001000011000 1',
'111011011001000101110011111011111011100111110101001100111110010000 10
00111001101011010100010111110101110101111010111100101100010011001001 0
11101000101011000110111000010000101001000100111010110001010000111110 1
10111000011001100010001101000010001111111110000010111100010010100000 00
010010010011011100001001110011100010010110101111110101111011011010011
```

```
10111010111110110011001000010001010100010010110110101011100000101111
00100110011110001001001111100101111001111011011010111001001111010001 1
00110001100001100000110000011111010100101111000000101011111010000111 1
10000101111100010000010010111010110100101010100111110010101110001106
01001011000101010100110110001011000001000111001111001110011100011015
01010111010011010000001100001011000011101101000000011111000101111101 0
11110011000011011000100100110111010011001111101100101100011000101001 1
10101111001000010110010111101110110010101101000000101001011000000001 1
10001110000100000001001111100011010011000000011011101111101001111110 0
01011101100000010001001010011000001',
'00011010101010101000010010011000100001010101000010100010001000111011
00110001001100001001110000110100010101111010110111001101011011101110 0
00011001000100100101000011011010001110010010100111000100101011011106
11100100111110111001010010111010100000100111110101110010010110100001 0
00010010001101111001110100010001011101100111011101011101100100101011 0
10101000101001000101110011011111110110011111111100000000011100000010 0
11000110001000110101010001011000010101000110000101001110101010111011 0
10010111011001010011100010101001100110000110101100010000100110101110 1
00001101001011011110011001100110010101101001010101111101101111000001 1
10100011111101110000000000111011011101000011001010010111001110111000 1
00111011110100101000100011011101100011111000101110110110111111001111 0
00000011100011000010000101001011001101110101000010101001000100110010 0
00101001111100101000001011011010011110001101000001101111010100101001 1
00010100000111000011110101010100011011001110001011110111010111011010 1
01101100000110000001010010101111011']
mask='10001001000010000100010010001001'
for i in range(4):
    R = ''
    key=ot[i][:32]
    for j in range(32):
        output = '?' + key[:31]
        ans = int(key[-1])^int(output[-1])^int(output[-
4])^int(output[-8])^int(output[-11])^int(output[-15])^int(output[-
20])^int(output[-25])^int(output[-28])
        R += str(ans)
        key = str(ans) + key[:31]
    R = format(int(R[::-1], 2), 'x')
    flag = R
    print(flag,end='')
#fbbbee823f434f919337907880e4191a
```

再根据 uuid 的规则还原

hgame{fbbbee82-3f43-4f91-9337-907880e4191a}

奇怪的图片

观察加密代码知题目为在一张图上依次写下 flag 的每一位，并输出每次写后与同一张图片的异或结果，那么我们只要把两两图片异或，如果结果中只剩下一个字符，则这两张图是相邻生成的。（只要找到两大括号内就行，hgame 已知）

依次读取

```
print("}c308_be71_fda1{"[::-1])
#{1adf_17eb_803c}
```

flag 即为 hgame{1adf_17eb_803c}