2020湖湘杯-CRYPTO-LFSRXOR

原创

大熊何在 于 2020-11-05 13:21:33 发布 544 收藏 3

分类专栏： CRYPTO

本文链接：https://blog.csdn.net/zippo1234/article/details/109509826

2020HXB-CRYPTO-LFSRXOR

每天一题，只能多不能少

LFSRXOR

题目分析

可以说是被LFSR给骗了，其实跟LFSR根本没关系。

1. 异或逆向
2. 同原文不同密钥循环异或加密逆向

开始

1.题目

给了一个加密算法，两个enc在算法末端注释部分。

```
import numpy as np
from pylfsr import LFSR
from Crypto.Util.number import *
import random
import string
from secret import flag

assert flag[:6] == "DASCTF"

def xor(a,b):
    return str(chr(a^b)).encode('latin1')


def encode(content,key):
    tmp=b""
    for i in range(len(content)):
        tmp += xor(content[i],key[i%len(key)])
    return tmp

def shuffle_str(s):
    str_list = list(s)
    random.shuffle(str_list)
    return ''.join(chr(i) for i in str_list).encode('latin1')


ran_str = ''.join(chr(random.randint(1,256)) for _ in range(512)).encode()
content = ran_str+flag

L4 = LFSR(fpoly=[4,3],initstate ='random',verbose=True)
data = L4.runFullCycle()
k4 = b""
for _ in range(len(data)):
    a = b''
    for _  in range(8):
        a +=  str(L4.next()).encode()
    k4 += long_to_bytes(int(a,2))


L5 = LFSR(fpoly=[5,4,2,1],initstate ='random',verbose=True)
data = L5.runFullCycle()
k5 = b""
for _ in range(len(data)):
    a = b''
    for _  in range(8):
        a +=  str(L5.next()).encode()
    k5 += long_to_bytes(int(a,2))

k4 = shuffle_str(k4)
k5 = shuffle_str(k5)

enc2=encode(content,k5)
enc1=encode(content,k4)

print(enc1)
print(enc2)
#b'\xbb\xd3\x08\x15\xc6:\x08\xb2\xb2\x9f\xe4p\xc7\xec\x7f\xfd\xf6f\x9c\xe4\xd12\xae/\x81\xb1\x88\xab\xa5\/\xa9\x88\x14\xdf`~\xf6\xdb/\xb4\x06
```

S!0\xbb\xe4\x1a\xe6R\x8e\x84X\x19K\x95\x07C\xe8\xb2!'\xa9\x80\x15\xec\x8f\x8dY\nK\x85\x99\xb7!\x134\xa9\xb6\x15\xcf&\r\x9b\xe1\x99\xe4]3h
~\xf0\xa9\xa5\x14\xee}\xd19l\x14h\x07v *a0\x12\x14\xfe\x0f\x05\xdem\x1d\xe4s2J\x7f\xc28\xf6RR\x8e\xba\xb2m\x18M\xf1\xef!4\x17\xa8\xb4\x14\x
c2\x8f\xb9Y:K\xaa\x06T!\x1b\xbb\xfd\xf6G\x8e\x9a\xeb\xd9K\xbb\x06N\x9a\x82c\xa9\xa0\x14\xed!\x04\xdbm\x13\xe5w3B\x7f\xd0\xa9\xbf\xb7\x9c
\xe3\xd00\x83K\x86\xab3\x7f\xc1\xbb\xfd\x11\x15\xdf\x8e\x80Y\x07\xd8\xe5]2m\xe9\xbb\xce`\x91o\x8f\x8cY!\x81\xe4J\x92\x8c\xa7T\x16E\x15\xf1
WMY(\xb8[\x8e2y~\xcbM\x10\x15\xc7\x1fWY\x0cK\x87\xce\xe5 !b\xa8\x83\x14\xec6\xd1!\xc8\x905\xe52L\xf1\xba\xcf\n\x9d\x9d\xe7u\xadm\x06\xe
4n2r\xd8\xba\xed\xf6\x7f\x9d\xd8\xd02m\x12G\x07Y\x89\x7f\xc0\xa8\xa4\x15\xe5\x043Y\x1eJ\xae\x07n\x94\x87\xbb\xcf_\x8d\x9d\xd1\x14Y,\x9e\x
e5b\xd7\x8c\x7f\xf7\xa8\x8f\x14\xc7\x8f\xb3\xb6\xf1\x93\xe4O\xdd\xc4\xdb\xba\xf6!\x15\xfd.\xd1\x18\xcf\xf6\x03\xea2E\x7f\xe1\xa9\xa5\xfe\x9d\xc9
\xd1;\xd9\xee\x05\x06z\xc8\xb2\xbb\xe2\xf7{JW4\xcdm\x1a\xe5U\x8d \x0f&\x14\x7f\xf6\x9d\xd4E\xbf\xc3\xdb\xe4L\xe1\xf7\x90\xbb\xdaZ\xf4\x9d\x
d13\xb8m3\xe2D3o~\xf8H\xf6U*\x07IY\x03K\xab\x07~\xa3\x87\xbb\xc9\xf7sAQ\x08Y6J\x86\x07Y\xec\xf7\xbb\xc6s\x15\xc6\x7fEY\x02J\x95\x07Z \
x11\xbb\xc6T\x15\xfc-\xd0\x06\xe6\x9f-\x07^ \x15\xbb\xccz\x14\xf3\x8f\x97\xd4l9t\x85\xe8\x8a\xbe\xbb\xf9\xf6f\x9d\xf2\xd19\xa2K\xb6\xcd\xcf\xf6~
\xd5\xa9\xaa\x15\xd8\x8e\xb3\x81m9\xe4f\xb2!\x1e\xba\xd8s\xfd\x11\x08W\xa1I;\x01\x07_!\x11\xbb\xdd\xf6x\x9d\xf0\x17Y\x15\xfe\x02\xc7\xa0!.W
\xa9\xa5\x8f\x9c\xe8\xd1\x12m\x04\xe5s3Q~\xdd\xa9\xa3\x15\xdb\x8f\xac\xaf\xec\xbb\x10\xde2_\xba\xba\xe8\xf6f.\x1e\xd1\x17l\x06\xe4U\xdd\xf0\
xd6~\x0fA\x14\xcb\x8e\xb0Y\x1fJ\xb2\xe4\xb3!"\xba\xfeU\x14\xedY\xd0>I-~\x06P 1\xbb\xf2\xf6waD\xd1(m\x12`\x06@\xb6~\xfa\xa9\xb1\xb0\x9d\xf
b\x18\xfbm&\xe4v2w\xce\xba\xcbo\xd5\x07\x11QX<J\xbd\xb22O\x7f\xd8x>\xc8\x9c\xd3\xd03\x9d\xb5\x1e\xd72S\xf2ry\xf1W\x9c\xc89Y\rK\x8f\xff\x
8a\xe0\xb5{\xa9\xae\xb1\x9d\xdd\xd1=\xbeK\xa3\x06e!\x08\xba\xd2\xf6j\x9c\xf6\xd0\x0fl#\xe5o\xf5\xaa~\xc2\xa9\x99\x15\xea6\xd1:\xe7\xa8\xe4n\
xbb \nV\xa9\x91\x14\xf9}\xd0!m/\xe5|2o\x81\xba\xf8\r\x14\xeb\tR\xc9\xec\xdd`\xbf\xc6\x81\xdfKXW\xb3o.%\xa9\xcd\xb9\x14\xfd\x97\x83\x8eO\n\x
03\xb6iuu\xab\x9d\xbc\x15\xf4\xc3\xd6\xc1'
#b'p\xfd\x1ff\xcaB\xa5\xe6`\x87\xa8\x8ci\x855\x92O8P\xa5}^\xd8\xed\x1a\x88=c\xe0\x9f\xedq\xf8\xe1%\x7fX\xd2\xba\xbe\x03\xa8\x9a\x9c\x075\x
98"\xca\xed\xa4C^\xc6.j\xec\xfa\x10\xa7\xd9\x01\x06\x87\x90f\xcc\xf6\x1b\x0c\xde\xcc,\xfb\xf0\xc74\x94\xcfj\x8ay\xd5\xd2`.@\xed\xc2\xd8!DSp\xf5
\x12f\xf1\xf6#\x80\xbe\x16\xa8\xaeF\xd0\xd1\xd4\xad\xb9\xf7#\x16\x08\xb2[\x1a\x87\x8b\xa0\xfaEF\xbf\x86\x8b\x8c\x90\xa4\xd5\xfbcR\xe2W\x9c\
n5\x8b\xcfQ"\xf2\x16\x10\xb2l\x1a\x88\x8b\x8cj\x16\xebp\xccS\xd2\x90\xa8|q\x05\xafq\xfa\xcaHE{\x1a\xba#\xfd\x17/\xb2L\x1a\x87\x8a\x90\xc9Dm
p\xef\x0ef\xf2Z|S\x00R\xfc\x1c\x9d\n5\x84\xceS\xb0\xa4M_\xff\xb9\x1a\x8a\x1d\\\x98D\\p\xcb*f\xdcV\xd0\xd5Q\xec\x1a\xfa\xf0\x91\xa8\xd4\x8a\xc
a\x9c-\x17\x07\xb2_\xff\n\x8a\x83\xfb\xc2\x00\x10\x87\x83\xaeF\xf7#\xd4\xbe\'\xa9\x8a$IMp\x14\xe8\xc0\xa4z\xd1\xb2H\xe6e\x8b\xb0\xcf\xb1\x01
<\x87\x88g\xc2Q|H\xbe9\xa9\xad\x9c#4\x8cl8l\x0c\x17$\xb3}\x1b\x94\x01:j7\x00;\x86\xbd\xd2i\xf6\x1a\xa4\'R\xf6?\x9c\x08\xe1\xd4\xab\xdd\x8f\xa
4[_\xca/@\xed\xe86\xf7\x9c\x018i\x04\xc3\x90\xa8\xaa\x0c\xde\xf2\xa8\xba?\xf4\xd39\xce\\"\xfe\x16\x0cY/]\xed\xe9l\xce\xa5\x018o,g\xdb\xf7\x12\
xdag\xb6=\xfa\xccHgk\xcfH\xbf\x18\x9e\xbd\xb3u\x8f\n$Hk\x0e\xd3\xa6i\xe1\x15=\x16}R]\xb3\xa8\x82\x9b\x0b4\x9a\xcf{\xc2\xa4V\xe8:\x93\x1a\x
83\x8a\x97j\t\x82\x88\x86\x80f\xf6*\xa2\xd5\xbe\x08\xa9\x98\x9c#\xf8\\\xceV\xa7\xa5L\xae&/t\xec\xfb\xd9\x02Dnp\xe8Cf\xf0U}R4\x87a\xfb\xf0I_\x
d4\xaa\xb4"\xca\x16\x18>/j}\t\x03\xc1\x84\x00!\x86\x93g\xed\xf7\x1d\xc3\xbf\x01c\x06KI[\xd5\x929g\xa4t\x87\xb2\\\x1b\x8d\x0b\xd9\x0bDp\xf5om\
xe1\x16\x0e}|ZR\xc4\xfb\xf2H@\xd4\xa28\\c\x17&\x07\xc8\xda~\x8b\x88\x86DS\xeb\x87\x87f\xda\xf73\r\xcaS\xd9\xfa\xfa\`\xd5\x889^R\x97\xaeF\x
f6\x1a\x92N\xd8*Er\xc3\x16\xe0)\x91\xba|_Q\x83\x00>;\xff5\x82\xceX"\xd7\x17\x08P\xae\x1a\xb1\x8a\x8f\xc9Ep\xa7\x86\x86g\xf6m|o\xbf\x1c\xa9\
xa1\x9c+\xc9\x1e\xcfl#\xfc\x92^\xc1\xb8\x1b\xad\x8a\x9e\xceEu\xb8$\xe0\x0b\x90\x87}[\x0fS\xcab]\xd2\xaaU\xcfh"\xfc\xa2_\xdd/y<C\x05k\x18\x0
0\x1aw\x1e\x9cA\xf6\x0f\x80w\x83\xae\xb8\x9d\x0e\xdc\xd4\xaf9H\\\xaf\x9ey\xef\x1b\xb4.\xd99Dd\xa2\x87\xa7f\xc6\xf6\n\x0c\xc4R\xd7\xfa\xe4Hc\
xd4\xa78Jc\x9c^\xca.u\xed\xfcak&\x8b\x92\x87\x88\xee\x90\x83\x90\x0c\xd9R\xcd\x08\x9c04\xb1\xceC"\xea\xe9^\xe3\xd4\x1a\x9a\x0c[\xfa\xc5\x9
7\xf5>\x15\xc71\x06\x8d\xac\x19\xa0\t\x0e/\xe9\xc6%4\x9d\x80U\xe3\xfdF\x8d\xee\x17.+\x9b\xb3\xf0\x83w\x16\xd9'

2.数学理论

假装自己真的知道，嗯。我真（wan）的（quan）知（bu）道（dong）：
首先是最基础的。

$$a \oplus b = c$$

$$a \oplus c = b$$

然后是稍微复杂的。

3.回到题目

（1）源代码

加密脚本简单说起来就是通过LFSR算法产生了两个key。其中一个是L4，另一个是L5。
两个key经过打乱顺序后用于encode。encode其实就是循环异或而已。
encode的原文是一个随机序列+flag。
所以当时比赛的时候的想法是首先去爆LFSR的状态得到L4和L5。然后再去爆随机的问题。想也知道以我的水平怎么可能写得出来。。。虽然现在这个WP我也还是写不出来。

（2）LFSR分析

试试自己生产一个看看，分析一下生成的key的情况。

```
>>> from pylfsr import LFSR
>>> L4 = LFSR(fpoly=[4,3],initstate ='random',verbose=True)
>>> L4.info()
4 bit LFSR with feedback polynomial  x^4 + x^3 + 1
Expected Period (if polynomial is primitive) =  15
Current :
 State       :  [0 1 0 1]
 Count       :  0
 Output bit  :  -1
 feedback bit :  -1
 >>> L5 = LFSR(fpoly=[5,4,2,1],initstate ='random',verbose=True)
>>> L5.info
<bound method LFSR.info of <pylfsr.pylfsr.LFSR object at 0x000002A373695EB0>>
>>> L5.info()
5 bit LFSR with feedback polynomial  x^5 + x^4 + x^2 + x^1 + 1
Expected Period (if polynomial is primitive) =  31
Current :
 State       :  [1 1 1 1 0]
 Count       :  0
 Output bit  :  -1
 feedback bit :  -1
```

也就是L4生成15位的key。L5生成31位的key。你再打乱位数又不会变，而且加密过程中也没有再打乱。

（3）content原文分析

content是一串随机+flag，所以可以猜到最后一位必定是 *}*

也就是说enc的最后一个字节与 *}* 异或后就可以得到key中的一位。

我们用L5生产的k5来分析。

enc的长度是810。

k5的长度是31。

$810/31 = 4$

所以可知enc的最后一个字节是由k5的第4个字节异或得到的，于是可得到$k5[3]$

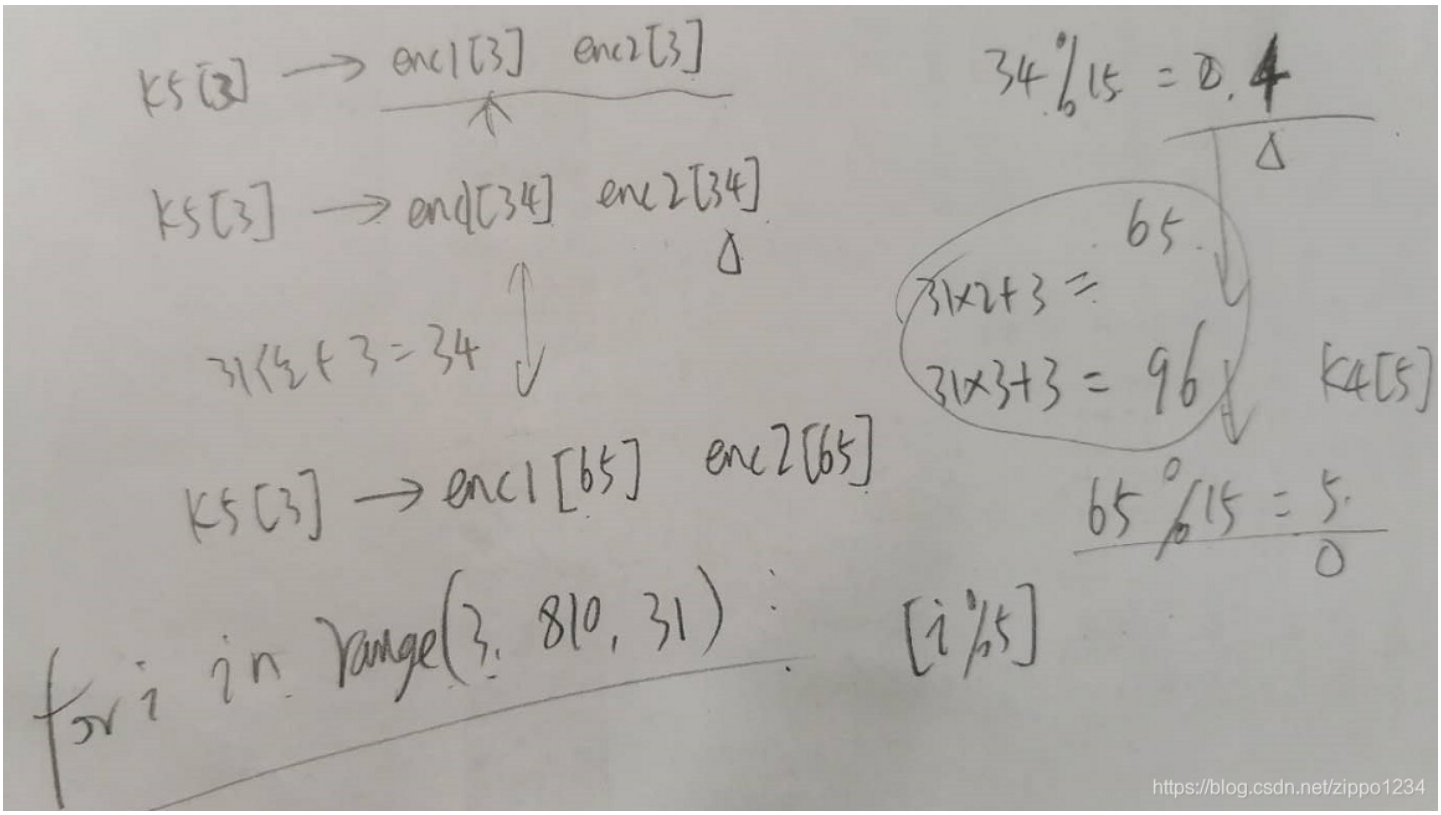
```
k5[3] = enc2[809]^ord('}')
```

同时根据上面数学理论中

```
将明文使用两个密钥加密得到两个密文，根据异或关系，可以从一个密钥得到另一个密钥。
```

的原理，可以得到。

```
k4[3] = k5[3] ^ enc1[3] ^ enc2[3]
k4[4] = k5[3] ^ enc1[34] ^ enc2[34]
k4[5] = k5[3] ^ enc1[65] ^ enc2[65]
```



4.破解

那么就可以用这个宝贵的k5[3]来得到整个k4。
k4有了。不就是无脑还原明文了？

```python
#!python3
# -*- coding: utf-8 -*-
# @Time : 2020/11/5 11:48
# @Author : A.James
# @FileName: lfsr-exp-1.py
#fork from:https://www.cnblogs.com/lnjoy/p/LFSRXOR.html
import string

chars = string.ascii_letters+string.digits+'{}'
enc1 = b'\xbb\xd3\x08\x15\xc6:\x08\xb2\xb2\x9f\xe4p\xc7\xec\x7f\xfd)\xf6f\x9c\xe4\xd12\xaeJ\x81\xb1\x88\xab\xa5V\xa9\x88\x14\xdf`~\xf6\xdbJ\xb4\x06S!0\xbb\xe4\x1a\xe6R\x8e\x84X\x19K\x95\x07C\xe8\xb2\'\xa9\x80\x15\xec\x8f\x8dY\nK\x85\x99\xb7!\x134\xa9\xb6\x15\xcf&\r\x9b\xe1\x99\xe4]3h~\xf0\xa9\xa5\x14\xee}\xd19l\x14h\x07v *a0\x12\x14\xfe\x0f\x05\xdem\x1d\xe4s2J\x7f\xc28\xf6RR\x8e\xba\xb2m\x18M\xf1\xef!4\x17\xa8\xb4\x14\xc2\x8f\xb9Y:K\xaa\x06T!\x1b\xbb\xfd\xf6Gv\x8e\x9a\xeb\xd9K\xbb\x06N\x9a\x82c\xa9\xa0\x14\xed!\x04\xdbm\x13\xe5w3B\x7f\xd0\xa9\xbf\xb7\x9c\xe3\xd00\x83K\x86\xab3\x7f\xc1\xbb\xfd\x11\x15\xdf\x8e\x80Y\x07\xd8\xe5]2m\xe9\xbb\xce`\x91o\x8f\x8cY!\x81\xe4J\x92\x8c\xa7T\x16E\x15\xf1WMY(\xb8[\x8e2y~\xcbM\x10\x15\xc7\x1fWY\x0cK\x87\xce\xe5 !b\xa8\x83\x14\xec6\xd1!\xc8\x905\xe52L\xf1\xba\xcf\n\x9d\x9d\xe7u\xadm\x06\xe4n2r\xd8\xba\xed\xf6\x7f\x9d\xd8\xd02m\x12G\x07Y\x89\x7f\xc0\xa8\xa4\x15\xe5\x043Y\x1eJ\xae\x07n\x94\x87\xbb\xcf_\x8d\x9d\xd1\x14Y,\x9e\xe5b\xd7\x8c\x7f\xf7\xa8\x8f\x14\xc7\x8f\xb3\xb6\xf1\x93\xe4O\xdd\xc4\xdb\xba\xf6!\x15\xfd.\xd1\x18\xcf\xf6\x03\xea2E\x7f\xe1\xa9\xa5\xfe\x9d\xc9\xd1;\xd9\xee\x05\x06z\xc8\xb2\xbb\xe2\xf7{JW4\xcdm\x1a\xe5U\x8d \x0f&\x14\x7f\xf6\x9d\xd4E\xbf\xc3\xdb\xe4L\xe1\xf7\x90\xbb\xdaZ\xf4\x9d\xd13\xb8m3\xe2D3o~\xf8H\xf6U*\x07IY\x03K\xab\x07~\xa3\x87\xbb\xc9\xf7sAQ\x08Y6J\x86\x07Y\xec\xf7\xbb\xc6s\x15\xc6\x7fEY\x02J\x95\x07Z \x11\xbb\xc6T\x15\xfc-\xd0\x06\xe6\x9f-\x07^ \x15\xbb\xccz\x14\xf3\x8f\x97\xd4I9t\x85\xe8\x8a\xbe\xbb\xf9\xf6f\x9d\xf2\xd19\xa2K\xb6\xcd\xcf\xf6~\xd5\xa9\xaa\x15\xd8\x8e\xb3\x81m9\xe4f\xb2!\x1e\xba\xd8s\xfd\x11\x08W\xa1I;\x01\x07_!\x11\xbb\xdd\xf6x\x9d\xf0\x17Y\x15\xfe\x02\xc7\xa0!.W\xa9\xa5\x8f\x9c\xe8\xd1\x12m\x04\xe5s3Q~\xdd\xa9\xa3\x15\xdb\x8f\xac\xaf\xec\xbb\x10\xde2_\xba\xba\xe8\xf6f.\x1e\xd1\x17I\x06\xe4U\xdd\xf0\xd6~\x0fA\x14\xcb\x8e\xb0Y\x1fJ\xb2\xe4\xb3!"\xba\xfeU\x14\xedY\xd0>I-~\x06P 1\xbb\xf2\xf6waD\xd1(m\x12`\x06@\xb6~\xfa\xa9\xb1\xb0\x9d\xfb\x18\xfbm&\xe4v2w\xce\xba\xcbo\xd5\x07\x11QX<J\xbd\xb22O\x7f\xd8x>\xc8\x9c\xd3\xd03\x9d\xb5\x1e\xd72S\xf2ry\xf1W\x9c\xc89Y\rK\x8f\xff\x8a\xe0\xb5{\xa9\xae\xb1\x9d\xdd\xd1=\xbeK\xa3\x06e!\x08\xba\xd2\xf6j\x9c\xf6\xd0\x0fl#\xe5o\xf5\xaa~\xc2\xa9\x99\x15\xea6\xd1:\xe7\xa8\xe4n\xbb \nV\xa9\x91\x14\xf9}\xd0!m/\xe5|2o\x81\xba\xf8\r\x14\xeb\tR\xc9\xec\xdd`\xbf\xc6\x81\xdfKXW\xb3o.%\xa9\xcd\xb9\x14\xfd\x97\x83\x8eO\n\x03\xb6iuu\xab\x9d\xbc\x15\xf4\xc3\xd6\xc1'
enc2 = b'p\xfd\x1ff\xcaB\xa5\xe6`\x87\xa8\x8ci\x855\x92O8P\xa5}^\xd8\xed\x1a\x88=c\xe0\x9f\xedq\xf8\xe1%\x7fX\xd2\xba\xbe\x03\xa8\x9a\x9c\x075\x98"\xca\xed\xa4C^\xc6.j\xec\xfa\x10\xa7\xd9\x01\x06\x87\x90f\xcc\xf6\x1b\x0c\xde\xcc,\xfb\xf0\xc74\x94\xcfj\x8ay\xd5\xd2`.@\xed\xc2\xd8!DSp\xf5\x12f\xf1\xf6#\x80\xbe\x16\xa8\xaeF\xd0\xd1\xd4\xad\xb9\xf7#\x16\x08\xb2[\x1a\x87\x8b\xa0\xfaEF\xbf\x86\x8b\x8c\x90\xa4\xd5\xfbcR\xe2W\x9c\n5\x8b\xcfQ"\xf2\x16\x10\xb2l\x1a\x88\x8b\x8cj\x16\xebp\xccS\xd2\x90\xa8|q\x05\xafq\xfa\xcaHE{\x1a\xba#\xfd\x17/\xb2L\x1a\x87\x8a\x90\xc9Dmp\xef\x0ef\xf2Z|S\x00R\xfc\x1c\x9d\n5\x84\xceS\xb0\xa4M_\xff\xb9\x1a\x8a\x1d\\\x98D\\p\xcb*f\xdcV\xd0\xd5Q\xec\x1a\xfa\xf0\x91\xa8\xd4\x8a\xca\x9c-\x17\x07\xb2_\xff\n\x8a\x83\xfb\xc2\x00\x10\x87\x83\xaeF\xf7#\xd4\xbe\'\xa9\x8a$lMp\x14\xe8\xc0\xa4z\xd1\xb2H\xe6e\x8b\xb0\xcf\xb1\x01<\x87\x88g\xc2Q|H\xbe9\xa9\xad\x9c#4\x8cl8l\x0c\x17$\xb3}\x1b\x94\x01:j7\x00;\x86\xbd\xd2i\xf6\x1a\xa4\'R\xf6?\x9c\x08\xe1\xd4\xab\xdd\x8f\xa4[_\xca/@\xed\xe86\xf7\x9c\x018i\x04\xc3\x90\xa8\xaa\x0c\xde\xf2\xa8\xba?\xf4\xd39\xce\\"\xfe\x16\x0cY/]\xed\xe9l\xce\xa5\x018o,g\xdb\xf7\x12\xdag\xb6=\xfa\xccHgk\xcfH\xbf\x18\x9e\xbd\xb3u\x8f\n$Hk\x0e\xd3\xa6i\xe1\x15=\x16}R]\xb3\xa8\x82\x9b\x0b4\x9a\xcf{\xc2\xa4V\xe8:\x93\x1a\x83\x8a\x97j\t\x82\x88\x86\x80f\xf6*\xa2\xd5\xbe\x08\xa9\x98\x9c#\xf8\\\xceV\xa7\xa5L\xae&/t\xec\xfb\xd9\x02Dnp\xe8Cf\xf0U}R4\x87a\xfb\xf0l_\xd4\xaa\xb4"\xca\x16\x18>/i}\t\x03\xc1\x84\x00!\x86\x93g\xed\xf7\x1d\xc3\xbf\x01c\x06Kl[\xd5\x929g\xa4t\x87\xb2\\\x1b\x8d\x0b\xd9\x0bDp\xf5om\xe1\x16\x0e}|ZR\xc4\xfb\xf2H@\xd4\xa28\\c\x17&\x07\xc8\xda~\x8b\x88\x86DS\xeb\x87\x87f\xda\xf73\r\xcaS\xd9\xfa\xfaI`\xd5\x889^R\x97\xaeF\xf6\x1a\x92N\xd8*Er\xc3\x16\xe0)\x91\xba|_Q\x83\x00>;\xff5\x82\xceX"\xd7\x17\x08P\xae\x1a\xb1\x8a\x8f\xc9Ep\xa7\x86\x86g\xf6m|o\xbf\x1c\xa9\xa1\x9c+\xc9\x1e\xcfl#\xfc\x92^\xc1\xb8\x1b\xad\x8a\x9e\xceEu\xb8$\xe0\x0b\x90\x87}[\x0fS\xcab]\xd2\xaaU\xcfh"\xfc\xa2_\xdd/y<C\x05k\x18\x00\x1aw\x1e\x9cA\xf6\x0f\x80w\x83\xae\xb8\x9d\x0e\xdc\xd4\xaf9H\\\xaf\x9ey\xef\x1b\xb4.\xd99Dd\xa2\x87\xa7f\xc6\xf6\n\x0c\xc4R\xd7\xfa\xe4Hc\xd4\xa78Jc\x9c^\xca.u\xed\xfcak&\x8b\x92\x87\x88\xee\x90\x83\x90\x0c\xd9R\xcd\x08\x9c04\xb1\xceC"\xea\xe9^\xe3\xd4\x1a\x9a\x0c[\xfa\xc5\x97\xf5>\x15\xc71\x06\x8d\xac\x19\xa0\t\x0el\xe9\xc6%4\x9d\x80U\xe3\xfdF\x8d\xee\x17.+\x9b\xb3\xf0\x83w\x16\xd9'

k4 = [0]*15
k5 = [0]*31
k5[3] = ord('}') ^ enc2[809]
for i in range(3, 810, 31):
    k4[i%15] = k5[3] ^ enc1[i] ^ enc2[i]

for i in range(810):
    if chr(k4[i%15]^enc1[i]) in chars:
        print(chr(k4[i%15]^enc1[i]), end='')
    else:
        print('|', end='')
```

得到：

```
|||||w|||||6|||B|||||||||uy|||||||mlt|||||||||1|||||||||||||||||||||F||L||||k||N|||||||||0||||N|||||||B|D|||||||S|||||||k5|||o|||||||||||||||||qv|||x||||||A|||||||||||||||o|}|||||||||Q||||U||UF1|||||||cn||}p|||||1}
J|||||||RD|||||||||||{||g|||||M||J||f7|||||d|||||||||a||k|||||||l|||||||v||||jZ|||||||n|||||||||||||g||||c|||||||||||||||vg|||||||||||W|||||||||J||||V|IR|||||||o|||||||||||||tJ|||||||A|||||||||||||K||F||2V|||||||||a|||||l|||||||||
O||||N||RA|h|||||||||||||||||||||||||||C|||F|OED|||||||||||||||w|Q||||||X|||||||||||||||5C26|||||||pS|||||||j|dt|||||||||B|||||||||||X|||||||||||||F||T||||g|||a|||||||r||Z|Y|B||||v|||||||||||2|8|||N|||||||||||{|||||f|||||
|||||||||||||||||H|||||||{||H||J|||||||0|||||||||||8||DASCTF{7cc33bd1c63b029fa27a6a78f1253024}
```

这里为了方便显示，作者把非可见字符都用“|”来替换了，便于查找flag。

5.get flag

```
DASCTF{7cc33bd1c63b029fa27a6a78f1253024}
```

结语

深入分析代码，才能找到突破口。所以前提就是读懂加密过程并掌握一定的数学理论知识（问题就是没有这些能力。。。）

参考

2020湖湘杯-CRYPTO-LFSRXOR WriteUp