

# 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](#)

版权声明: 本文为博主原创文章, 遵循[CC 4.0 BY-SA](#)版权协议, 转载请附上原文出处链接和本声明。

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

版权



[CRYPTO 专栏收录该内容](#)

27 篇文章 1 订阅

订阅专栏

## 2020HXB-CRYPTO-LFSRXOR

[LFSRXOR](#)

[题目分析](#)

[开始](#)

[1.题目](#)

[2.数学理论](#)

[3.回到题目](#)

[\(1\) 源代码](#)

[\(2\) LFSR分析](#)

[\(3\) content原文分析](#)

[4.破解](#)

[5.get flag](#)

[结语](#)

[参考](#)

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'1vhb1vh31vh031vh151vh61vh081vh21vh21vh081vh41vh71vh071vhf11vh60vh01vh41vh12vh301vh81vh11vh22vh51vh091vh391vh141vhcf11vh61vh71vh41vh6
```

S!0\|xb\|xe4\|x1\|al\|xe6R\|x8e\|x84X\|x19K\|x95\|x07C\|xe8\|xb2\|`|xa9\|x80\|x15\|xec\|x8f\|x8d\|Y\|n\|K\|x85\|x99\|xb7\|`|x134\|xa9\|xb6\|x15\|xcf\|&r\|x9b\|xe1\|x99\|xe4\|3h  
 ~\|xf0\|xa9\|xa5\|x14\|xee\}\|xd19\|x14h\|x07v \*a0\|x12\|x14\|xfel\|x0f\|x05\|xdem\|x1d\|xe4s2\|x7\|xc28\|x6RR\|x8e\|xba\|xb2m\|x18M\|xf1\|xe\|f\|4\|x17\|xa8\|xb4\|x14\|x  
 c2\|x8f\|xb9Y\|K\|xa\|x06T\|!x1b\|xb\|xfdf\|x6G\|x8e\|x9a\|xe\|b\|xd9K\|x\|b\|x06N\|x9a\|x82c\|xa9\|xa0\|x14\|xed\|!x04\|x\|bd\|ml\|x13\|xe5w3B\|x7f\|xd0\|xa9\|xbf\|xb7\|x9c  
 \|xe3\|xd00\|x83K\|x86\|xab3\|x7f\|xc1\|x\|bb\|xf\|x11\|x15\|xd\|x8e\|x80Y\|x07\|xd8\|xe5\|2\|m\|xe9\|xb\|x\|ce\`|x91\|o\|x8f\|x8c\|Y\|!x81\|xe4J\|x92\|x8c\|x\|a\|7\|x16E\|x15\|xf1  
 WMY\|xb8\|x8e2y~\|xcbM\|x10\|x15\|xc7\|x1f\|WY\|x0c\|K\|x87\|x\|ce\|x5\|b\|xa8\|x83\|x14\|xec\|x6\|d\|1\|`|xc8\|x905\|xe52L\|xf1\|xba\|x\|cf\|n\|x9d\|x9d\|x\|e7\|u\|xad\|ml\|x06\|xe  
 4\|n2r\|xd8\|xba\|x\|ed\|xf6\|x7f\|x9d\|x\|d\|02m\|x12G\|x07Y\|x89\|x7f\|x\|c0\|xa8\|x4\|x15\|xe5\|x043Y\|x1e\|J\|x\|ae\|x07n\|x94\|x87\|xb\|x\|cf\|\_x8d\|x9d\|x\|d\|x14Y\|x9e\|x  
 e5b\|xd7\|x8c\|x7f\|xf7\|xa8\|x8f\|x14\|xc7\|x8f\|xb3\|x\|b6\|xf1\|x93\|xe4O\|x\|dd\|x\|c4\|x\|db\|xb\|x\|f6\|x15\|xf\|d\|`|xd1\|x18\|x\|cf\|x\|f6\|x03\|xe\|a2E\|x7f\|x\|e1\|xa9\|xa5\|x\|fe\|x9d\|x\|c9  
 \|xd1\|xd9\|x\|ee\|x05\|x06z\|x\|c8\|xb2\|x\|bb\|xe2\|xf7\{JW4\|x\|cd\|m\|x1\|al\|xe5\|U\|x8d\|x\|f\|&|x14\|x7f\|xf6\|x9d\|x\|d\|4E\|x\|bf\|x\|c3\|x\|db\|x\|e4L\|xe1\|xf7\|x\|90\|x\|bb\|x\|da\|Z\|x\|f4\|x\|9d\|x  
 d13\|x\|b8\|m3\|x\|e2D3o~\|xf8H\|xf6U\*\|x07I\|Y\|x03K\|x\|ab\|x07~\|xa3\|x\|87\|xb\|x\|c9\|x\|f7\|sA\|Q\|x\|08Y\|x\|86\|J\|x\|86\|x07Y\|x\|e\|c\|x\|f7\|x\|bb\|x\|c6\|s\|x15\|x\|c6\|x\|f7\|x\|5\|x\|07Z\|  
 x11\|xb\|x\|c6\|T\|x15\|x\|fc\|7\|x\|d0\|x\|06\|x\|e6\|x\|f\|7\|x\|a8\|x8f\|x\|b3\|x\|b6\|xf1\|x93\|xe4O\|x\|dd\|x\|c4\|x\|db\|xb\|x\|f6\|x15\|xf\|d\|`|xd1\|x18\|x\|cf\|x\|f6\|x03\|xe\|a2E\|x7f\|x\|e1\|xa9\|xa5\|x\|fe\|x9d\|x\|c9  
 \|xd1\|xd9\|x\|ee\|x05\|x06z\|x\|c8\|xb2\|x\|bb\|xe2\|xf7\{JW4\|x\|cd\|m\|x1\|al\|xe5\|U\|x8d\|x\|f\|&|x14\|x7f\|xf6\|x9d\|x\|d\|4E\|x\|bf\|x\|c3\|x\|db\|x\|e4L\|xe1\|xf7\|x\|90\|x\|bb\|x\|da\|Z\|x\|f4\|x\|9d\|x  
 xd13\|x\|b8\|m3\|x\|e2D3o~\|xf8H\|xf6U\*\|x07I\|Y\|x03K\|x\|ab\|x07~\|xa3\|x\|87\|xb\|x\|c9\|x\|f7\|sA\|Q\|x\|08Y\|x\|86\|J\|x\|86\|x07Y\|x\|e\|c\|x\|f7\|x\|bb\|x\|c6\|s\|x15\|x\|c6\|x\|f7\|x\|5\|x\|07Z\|  
 x11\|xb\|x\|c6\|T\|x15\|x\|fc\|7\|x\|d0\|x\|06\|x\|e6\|x\|f\|7\|x\|a8\|x8f\|x\|b3\|x\|b6\|xf1\|x93\|xe4O\|x\|dd\|x\|c4\|x\|db\|xb\|x\|f6\|x15\|xf\|d\|`|xd1\|x18\|x\|cf\|x\|f6\|x03\|xe\|a2E\|x7f\|x\|e1\|xa9\|xa5\|x\|fe\|x9d\|x\|c9  
 \|xd1\|xd9\|x\|ee\|x05\|x06z\|x\|c8\|xb2\|x\|bb\|xe2\|xf7\{JW4\|x\|cd\|m\|x1\|al\|xe5\|U\|x8d\|x\|f\|&|x14\|x7f\|xf6\|x9d\|x\|d\|4E\|x\|bf\|x\|c3\|x\|db\|x\|e4L\|xe1\|xf7\|x\|90\|x\|bb\|x\|da\|Z\|x\|f4\|x\|9d\|x  
 xd6~\|x0fA\|x14\|x\|cb\|x8e\|x\|b3\|x81m9\|xe4f\|xb2\|`|x\|e1\|xb\|x\|d8s\|x\|fd\|x\|11\|x\|08W\|x\|a1\|`|x\|01\|x\|07\\_!x\|11\|xb\|x\|dd\|x\|f6\|x\|9d\|x\|f\|7\|x\|17\|x\|06\|x\|e4U\|x\|dd\|x\|f0\|  
 b\|x18\|xf\|bm\&\|x\|e4v2w\|x\|ce\|x\|ba\|x\|cb\|x\|d5\|x\|07\|x\|11QX<\|x\|bd\|x\|b22O\|x\|77\|x\|d8\|x>\|x\|c8\|x\|9c\|x\|d3\|x\|d03\|x\|9d\|x\|b5\|x\|e\|l\|x\|d72S\|x\|f2\|ry\|x\|f1\|W\|x\|9c\|x\|c89\|Y\|r\|K\|x\|8f\|x\|f7\|x  
 8a\|x\|e0\|x\|b5\|`|x\|a9\|x\|ae\|x\|b1\|x\|9d\|x\|dd\|x\|d1=\|x\|be\|x\|a3\|x\|06e!\|x\|08\|x\|ba\|x\|d2\|x\|f6\|x\|9c\|x\|f6\|x\|d0\|x\|0f#\|x\|e5o\|x\|f5\|x\|a9\|x\|99\|x\|15\|x\|e\|a6\|x\|d1:\|x\|e7\|x\|a8\|x\|e4n\|  
 x\|bb\|n\|V\|x\|a9\|x\|91\|x\|14\|x\|f9\}\|xd0\|m\|x\|e5\|2o\|x\|81\|x\|ba\|x\|f8\|r\|x\|14\|x\|eb\|t\|R\|x\|c9\|x\|e\|c\|x\|dd\`|x\|b\|x\|c6\|x\|81\|x\|df\|K\|X\|W\|x\|b3o.\%x\|a9\|x\|cd\|x\|b9\|x\|14\|x\|f\|7\|x\|83\|x\|8e\|0\|n\|x  
 03\|x\|b6i\|uu\|x\|ab\|x\|9d\|x\|bc\|x\|15\|xf4\|x\|c3\|x\|d6\|x\|c1'\`|  
 #b'p\|xf\|d\|x\|f1\|f\|x\|ca\|B\|x\|a5\|x\|e6`|x\|87\|x\|a8\|x\|8c\|x\|855\|x\|92O\|8P\|x\|a5\|^`|x\|d\|8\|x\|e\|d\|x\|1\|al\|x\|88=c\|x\|e\|0\|x\|9f\|x\|e\|d\|q\|x\|f8\|x\|e\|1\%x\|7f\|x\|xd2\|x\|ba\|x\|be\|x\|03\|x\|a8\|x\|9a\|x\|9c\|x\|075\|x  
 98``x\|ca\|x\|e\|d\|x\|a4C^`|x\|c6\|j\|x\|ec\|x\|fa\|x\|10\|x\|a7\|x\|d9\|x\|01\|x\|06\|x\|87\|x\|90f\|x\|cc\|x\|f6\|x\|1b\|x\|0\|x\|de\|x\|cc\|,x\|fb\|x\|f0\|x\|c74\|x\|94\|x\|cf\|x\|8ay\|x\|d5\|x\|d2`.\@|x\|e\|d\|x\|c2\|x\|d8\|D\|Sp\|x\|f5  
 \|x\|12f\|x\|f1\|x\|f6\#`|x\|80\|x\|be\|x\|16\|x\|a8\|x\|ae\|F\|x\|d0\|x\|d1\|x\|d4\|x\|ad\|x\|b9\|x\|f7\#`|x\|16\|x\|08\|x\|b2\|x\|1a\|x\|87\|x\|8b\|x\|a0\|x\|fa\|EF\|x\|b\|x\|86\|x\|8b\|x\|8c\|x\|90\|x\|a4\|x\|d5\|x\|fb\|c\|R\|x\|e2W\|x\|9c\|  
 n5\|x\|8b\|x\|cf\|Q\`|x\|f2\|x\|16\|x\|10\|x\|b2\|x\|1a\|x\|88\|x\|8b\|x\|8c\|x\|16\|x\|eb\|p\|x\|cc\|S\|x\|d2\|x\|90\|x\|a8\|q\|x\|05\|x\|af\|q\|x\|fa\|x\|ca\|HE\`|x\|1a\|x\|ba\#\|x\|f\|d\|x\|17\|x\|b2\|L\|x\|1a\|x\|87\|x\|8a\|x\|90\|x\|c9\|D\|m\|p\|x\|ef\|x\|e\|f\|x\|f2Z\|S\|x\|00R\|x\|f\|c\|1\|x\|c1\|x\|9d\|n5\|x\|84\|x\|ce\|S\|x\|b0\|x\|a4M\\_\|x\|ff\|x\|b9\|x\|1a\|x\|8a\|x\|1d\|\|\|x\|98D\|l\|p\|x\|cb\^`|x\|dc\|v\|x\|d0\|x\|d5\|Q\|x\|e\|c\|x\|1a\|x\|fa\|x\|f\|0\|x\|91\|x\|a8\|x\|d4\|x\|8a\|x\|c\|a\|x\|9c\|x\|17\|x\|07\|x\|b2\|\_x\|f\|f\|n\|x\|8a\|x\|83\|x\|fb\|x\|c2\|x\|00\|x\|10\|x\|87\|x\|83\|x\|xae\|F\|x\|f7\#\|x\|d4\|x\|be\|`|x\|a9\|x\|8a\|\$IM\|p\|x\|14\|x\|e8\|x\|c0\|x\|a4z\|x\|d1\|x\|b2H\|x\|e\|6\|x\|8b\|x\|b0\|x\|cf\|x\|b1\|x\|01  
 <\|x\|87\|x\|88g\|x\|c2Q\|H\|x\|be\|9\|x\|a9\|x\|ad\|x\|9c\#\|4\|x\|8c\|8/\|x\|0c\|1\|x\|b3\`|x\|1b\|x\|94\|x\|01\|j\|x\|00\|x\|86\|x\|bd\|x\|d2\|x\|f6\|x\|1a\|x\|a4\|`|R\|x\|f6\?|x\|9c\|x\|08\|x\|e\|1\|x\|d4\|x\|ab\|x\|dd\|x\|8f\|x\|a4\|`|  
 4\|\_x\|ca\@|x\|e\|d\|x\|e\|86\|x\|f7\|x\|9c\|x\|018\|x\|04\|x\|c3\|x\|90\|x\|a8\|x\|aa\|x\|0c\|x\|de\|x\|f2\|x\|ba\?\|x\|f4\|x\|d3\|x\|ce\|\|^`|x\|f\|e\|x\|16\|x\|0c\|Y\|/|x\|e\|d\|x\|e\|9\|x\|a5\|x\|018o,\|g\|x\|db\|x\|f7\|x\|12\|  
 x\|dag\|x\|b6=\|x\|fa\|x\|cc\|H\|g\|k\|x\|cf\|H\|x\|b\|f\|x\|18\|x\|9e\|x\|bd\|x\|b3\|x\|8f\|n\|S\|H\|k\|x\|0e\|x\|d3\|x\|a6\|x\|e\|1\|x\|15=\|x\|16\}\|R\|x\|b3\|x\|a8\|x\|82\|x\|9b\|x\|0b4\|x\|a9\|x\|cf\|`|x\|c2\|x\|a4\|v\|x\|e8\|x\|93\|x\|1a\|x  
 83\|x\|8a\|x\|97j\|t\|x\|82\|x\|88\|x\|86\|x\|80f\|xf6\^`|x\|a2\|x\|d5\|x\|be\|x\|08\|x\|9a\|x\|98\|x\|9c\#\|x\|f8\|\|\|x\|ce\|v\|x\|a7\|x\|5\|x\|ae\&\|t\|x\|e\|c\|x\|fb\|x\|d9\|x\|02D\|n\|p\|x\|e\|8C\|x\|f0\|U\|R\|4\|x\|87a\|x\|f\|0\|\_`|x\|d\|4\|x\|a\|x\|b6\|x\|D\|p\|x\|f50m\|  
 xe\|1\|x\|16\|x\|0e\}\|Z\|R\|x\|c4\|x\|f\|2H\@|x\|d4\|x\|28\|l\|c\|x\|17\|&\|x\|07\|x\|c8\|x\|da\~|x\|8b\|x\|88\|x\|86DS\|x\|e\|b\|x\|87\|x\|87f\|x\|d\|a\|x\|f73\|r\|x\|ca\|S\|x\|d9\|x\|fa\|x\|f\|a\`|`|x\|d5\|x\|889\^`|x\|97\|x\|ae\|F\|x\|t\|6\|x\|1a\|x\|92\|N\|x\|d8\\*E\|r\|c3\|x\|16\|x\|e0\}\|x\|91\|x\|ba\|\_Q\|x\|83\|x\|00\>;|x\|ff5\|x\|82\|x\|ce\|X\^`|x\|d7\|x\|17\|x\|08P\|x\|ae\|x\|1a\|x\|b\|x\|1\|x\|8a\|x\|8f\|x\|c9\|E\|p\|x\|a7\|x\|86\|x\|86g\|x\|f6m\|o\|x\|b\|f\|x\|c\|x\|a9\|  
 xa\|1\|x\|9c\|+|x\|c9\|x\|1e\|x\|cf\|\#\|x\|f\|c\|x\|92\^`|x\|c1\|x\|b\|x\|8\|x\|1b\|x\|ad\|x\|8a\|x\|9e\|x\|ce\|E\|u\|x\|b\|x\|8\\$|x\|e0\|x\|b\|x\|90\|x\|87\|/|x\|0f\\$|x\|cab\|]x\|d2\|x\|aa\|u\|x\|cf\|h\^`|x\|f\|c\|x\|a2\|\_x\|dd\|y\|<|C\|x\|05k\|x\|18\|x\|0\|x\|1aw\|x\|1e\|x\|9c\|A\|x\|f6\|x\|0f\|x\|80w\|x\|83\|x\|ae\|x\|b\|x\|8\|x\|9d\|x\|0e\|x\|d\|x\|d\|x\|4\|x\|af\|9H\||x\|a\|f\|9\|e\|y\|x\|ef\|x\|1b\|x\|b\|4\|x\|d9\|D\|d\|x\|a2\|x\|87\|x\|a7\|x\|7f\|x\|c6\|x\|6\|n\|x\|o\|x\|c\|4\|R\|d\|7\|x\|f\|a\|x\|e\|4\|H\|c\|x\|d4\|x\|a\|x\|9a\|x\|0c\|f\|x\|f\|a\|x\|c5\|x\|9  
 \|xf5\|>\|x\|15\|x\|c71\|x\|06\|x\|8d\|x\|ac\|x\|19\|x\|a0\|t\|x\|0e\|x\|e\|9\|x\|c6\%\|4\|x\|9d\|x\|80U\|x\|e\|3\|x\|fd\|x\|8d\|x\|ee\|x\|17.+\|x\|9b\|x\|b\|3\|x\|f\|0\|x\|83w\|x\|16\|x\|d\|9\`|`|

## 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。



## 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]

```

$k_5[3] \rightarrow \underbrace{enc1[3]}_{\Delta} \underbrace{enc2[3]}_{\Delta}$

$k_5[3] \rightarrow \underbrace{enc1[34]}_{\Delta} \underbrace{enc2[34]}_{\Delta}$

$$31 \times 2 + 3 = 65$$

$k_5[3] \rightarrow enc1[65] \quad enc2[65]$

$\text{for } i \text{ in range}(3, 810, 31) : \quad [i \% 5]$

$$34 \% 15 = 4$$

$$\begin{aligned} 31 \times 2 + 3 &= 65 \\ 31 \times 3 + 3 &= 96 \end{aligned}$$

$k_4[5]$

$$65 \% 15 = 5$$

<https://blog.csdn.net/zippo1234>

#### 4. 破解

那么就可以用这个宝贵的 $k_5[3]$ 来得到整个 $k_4$ 。

$k_4$ 有了。不就是无脑还原明文了？



```
|||w|||6||B||||uy|||mlt||||1|||||F||L||k||N||||0||N|||BD||||S||||k5||o|||||qv|||x|||A|||||o||}|||Q|||U|||UF1||||cn||}p|||1}
J|||||RD|||||K||g|||M||J||f7|||d|||||a||k|||||v|||jZ|||||n|||||g|||c|||||vg|||||W|||||J||V||R||||o|||||tJ|||A|||||K||F||2V|||||a|||||
O|||N||RA|h|||||C|||FOED|||||w||Q|||X|||||5C26|||||pS|||||jdt|||||B|||||X|||||F||T||g|||a|||r|||Z||Y||B|||v|||||28||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](#)