

ISCC2021 MISC WP

原创

Atkxor 于 2021-05-26 00:06:53 发布 2349 收藏 9

分类专栏: [WriteUp CTF](#) 文章标签: [ISCC](#)

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/qq_46150940/article/details/116460603

版权



[WriteUp 同时被 2 个专栏收录](#)

15 篇文章 0 订阅

订阅专栏



[CTF](#)

39 篇文章 2 订阅

订阅专栏

目录

练武

[李华的红包](#)

[Retrieve the passcode](#)

[海市蜃楼-1](#)

[区块链](#)

[美人计](#)

[我的折扣是多少](#)

[海市蜃楼-2](#)

[Hack the Victim](#)

[检查一下](#)

[变异的SM2](#)

[混乱的音频](#)

[小明的宠物兔](#)

擂台

[小明的表情包](#)

[Base小偷](#)

[真作假时假亦真](#)

来自菜鸡的复现

练武

李华的红包

题目描述：

大年初一，李华给爸爸拜年，从事计算机行业的父亲发给李华一张图片和一张银行卡。父亲告诉李华密码就藏在图片中，但是李华打开图片后却百思不得其解。你能帮助李华拿到密码吗？

下载附件，

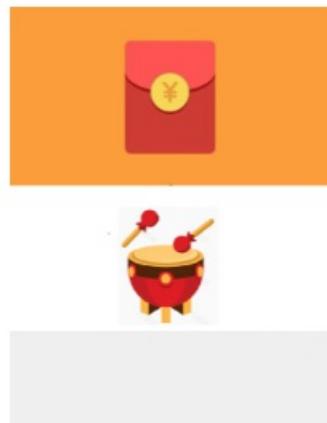


https://blog.csdn.net/qq_46150940

直接binwalk分离得到hongbao.txt

24,43,13,13,12,21,43

没啥思路，看起来图片不完全，修改图片高度，图片下面有一个鼓，联想到敲击码



https://blog.csdn.net/qq_46150940

对照敲击码表

敲击码表：

1	2	3	4	5
1 A	B C/K	D E		
2 F	G H	I J		
3 L	M N	O P		
4 Q	R S	T U		
5 V	W X	Y Z		

https://blog.csdn.net/qq_46150940

转换得到 **ISCCBFS**，有点坑，说是银行卡密码，其实并不是六位数字，最终flag为

ISCC{ISCCBFS}

Retrieve the passcode

题目描述：

Scatter说他能解开这个古怪的密码，你呢？来试试吧！Flag格式：ISCC{XXX}，XXX为小写字符串，不包括空格

xyz.txt

```
1:3:1
1.25:3:1
1.5:3:1
1.75:3:1
2:3:1
2:2.75:1
2:2.5:1
2:2.25:1
```

2:2:1
2:1.75:1
2:1.5:1
1:2.25:1
1.25:2.25:1
1.5:2.25:1
1.75:2.25:1
1:1.5:1
1.25:1.5:1
1.5:1.5:1
1.75:1.5:1
3:3:1
3.25:3:1
3.5:3:1
3.75:3:1
4:3:1
3.25:2.25:1
3.5:2.25:1
3.75:2.25:1
4:2.25:1
4:2:1
4:1.75:1
4:1.5:1
3:1.5:1
3.25:1.5:1
3.5:1.5:1
3.75:1.5:1
3:1.75:1
3:2:1
3:2.25:1
3:2.5:1
3:2.75:1
5:3:1
5.25:3:1
5.5:3:1
5.75:3:1
6:3:1
6:2.25:1
6:2:1
6:1.75:1
6:1.5:1
5.75:1.5:1
5.5:1.5:1
5.25:1.5:1
5:1.5:1
5:2.25:1
5.25:2.25:1
5.5:2.25:1
5.75:2.25:1
5:2.5:1
5:2.75:1
7:3:1
7.25:3:1
7.5:3:1
7.75:3:1
8:3:1
8:2.75:1
8:2.5:1
8:2.25:1
8:2:1

```
8:1.75:1  
8:1.5:1  
9:3:1  
9.25:3:1  
9.5:3:1  
9.75:3:1  
10:3:1  
10:2.75:1  
10:2.5:1  
10:2.25:1  
9.75:2.25:1  
9.5:2.25:1  
9.25:2.25:1  
9:2.25:1  
9:2:1  
9:1.75:1  
9:1.5:1  
9.25:1.5:1  
9.5:1.5:1  
9.75:1.5:1  
10:1.5:1  
11:3:1  
11.25:3:1  
11.5:3:1  
11.75:3:1  
12:3:1  
12:2.75:1  
12:2.5:1  
12:2.25:1  
12:2:1  
12:1.75:1  
12:1.5:1  
11.75:1.5:1  
11.5:1.5:1  
11.25:1.5:1  
11:1.5:1  
11:1.75:1  
11:2:1  
11:2.25:1  
11:2.5:1  
11:2.75:1  
11.25:2.25:1  
11.5:2.25:1  
11.75:2.25:1
```

百度找到的脚本，修改一下：

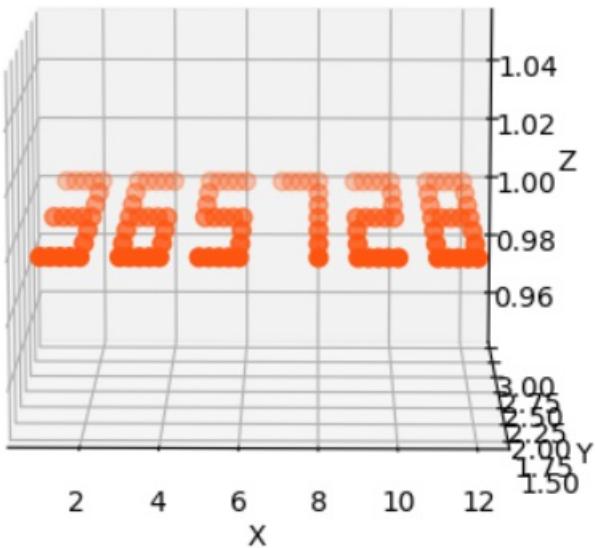
```
import matplotlib.pyplot as plt
import numpy as np
from mpl_toolkits.mplot3d import Axes3D

fig = plt.figure()
ax = fig.gca(projection="3d")

xs, ys ,zs= np.loadtxt('xyz.txt', delimiter=':', unpack=True)
ax.scatter(xs, ys, zs, zdir="z", c="#FF5511", marker="o", s=40)
ax.set(xlabel="X", ylabel="Y", zlabel="Z")

plt.show()
```

得到

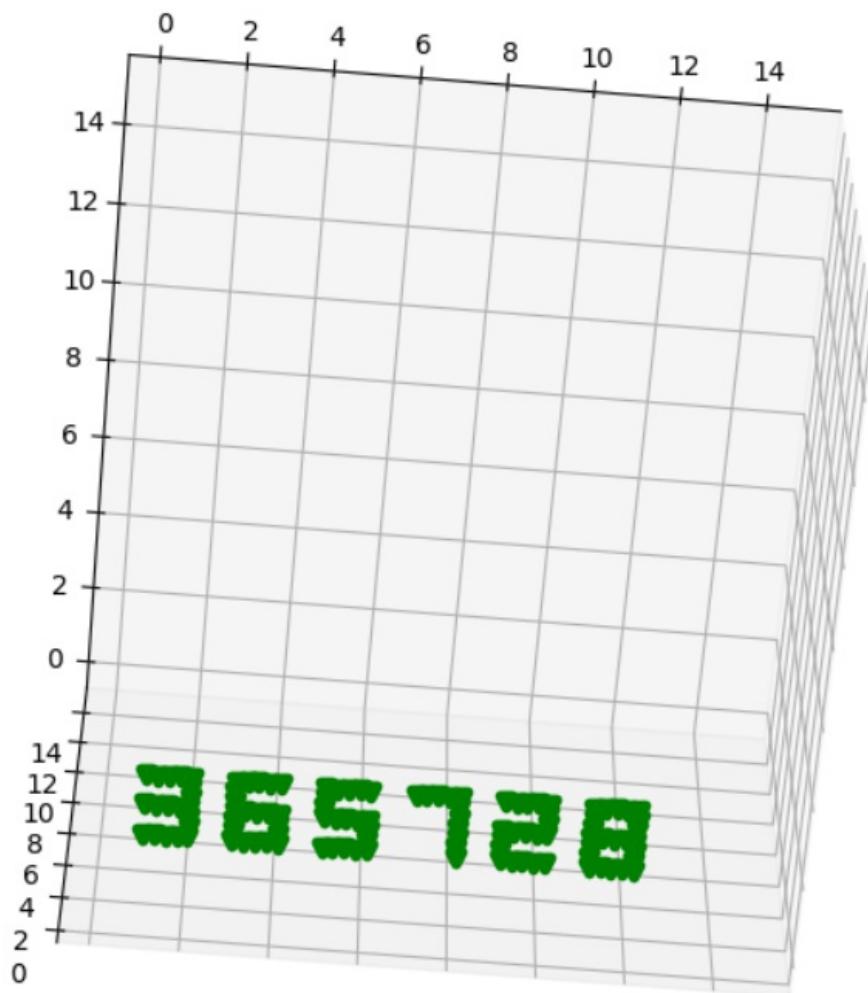


https://blog.csdn.net/qq_46150940

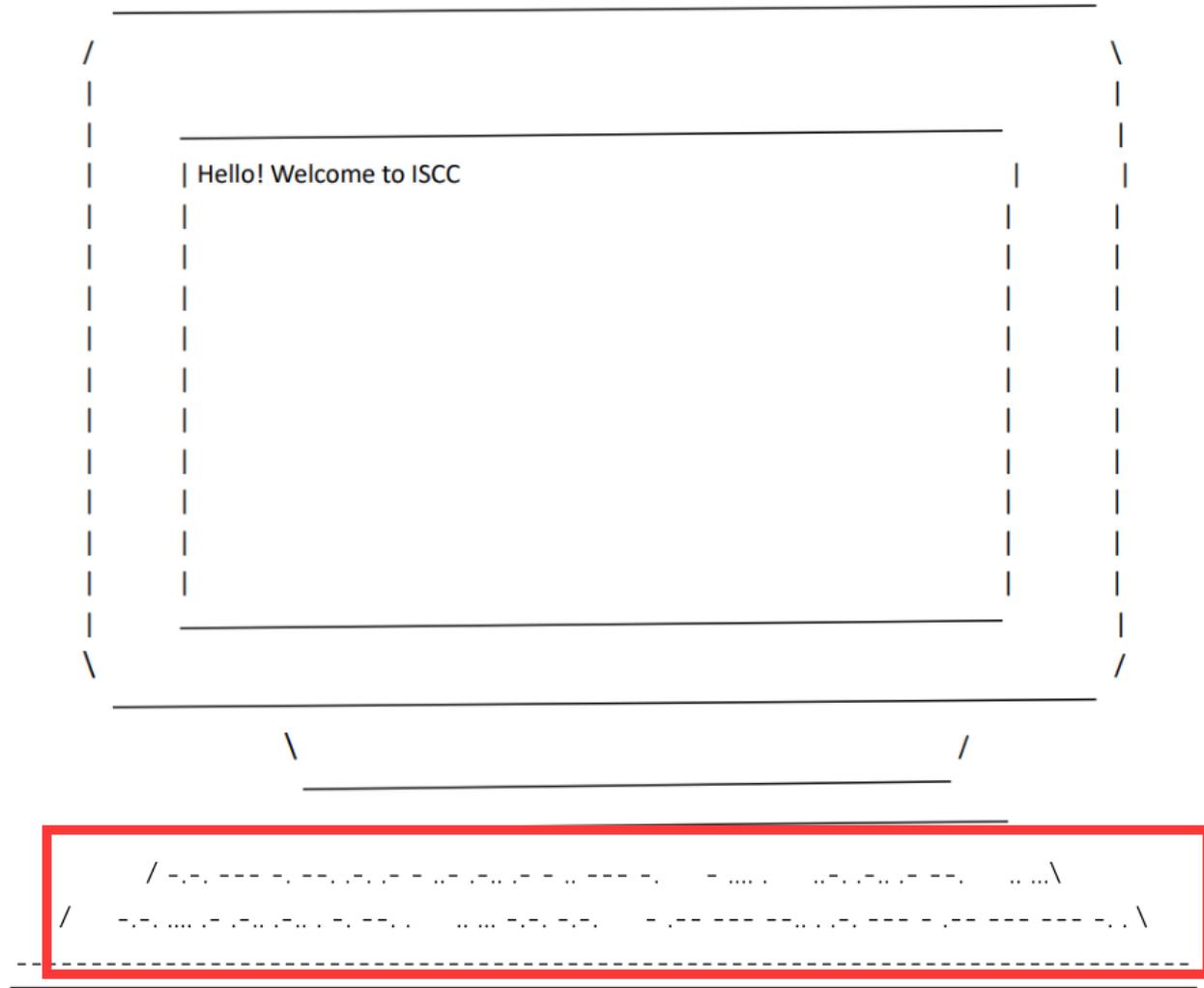
脚本2:

```
from matplotlib import pyplot as plt
from mpl_toolkits.mplot3d import Axes3D
dot1 = [[1, 3, 1], [1.25, 3, 1], [1.5, 3, 1], [1.75, 3, 1], [2, 3, 1], [2, 2.75, 1], [2, 2.5, 1], [2, 2.25, 1], [2, 2.1], [2, 1.75, 1], [2, 1.5, 1], [1, 2.25, 1], [1.25, 2.25, 1], [1.5, 2.25, 1], [1.75, 2.25, 1], [1, 1.5, 1], [1.25, 1.5, 1], [1.5, 1.5, 1], [1.75, 1.5, 1], [3, 3, 1], [3.25, 3, 1], [3.5, 3, 1], [3.75, 3, 1], [4, 3, 1], [3.25, 2.25, 1], [3.5, 2.25, 1], [3.75, 2.25, 1], [4, 2.25, 1], [4, 2, 1], [4, 1.75, 1], [4, 1.5, 1], [3, 1.5, 1], [3.25, 1.5, 1], [3.5, 1.5, 1], [3.75, 1.5, 1], [3, 1.75, 1], [3, 2, 1], [3, 2.25, 1], [3, 2.5, 1], [3, 2.75, 1], [5, 3, 1], [5.25, 3, 1], [5.5, 3, 1], [5.75, 3, 1], [6, 3, 1], [6, 2.25, 1], [6, 2, 1], [6, 1.75, 1], [6, 1.5, 1], [5.75, 1.5, 1], [5.5, 1.5, 1], [5.25, 1.5, 1], [5, 1.5, 1], [5, 2.25, 1], [5.25, 2.25, 1], [5.5, 2.25, 1], [5.75, 2.25, 1], [5, 2.5, 1], [5, 2.75, 1], [7, 3, 1], [7.25, 3, 1], [7.5, 3, 1], [7.75, 3, 1], [8, 3, 1], [8, 2.75, 1], [8, 2.5, 1], [8, 2.25, 1], [8, 2, 1], [8, 1.75, 1], [8, 1.5, 1], [9, 3, 1], [9.25, 3, 1], [9.5, 3, 1], [9.75, 3, 1], [10, 3, 1], [10, 2.75, 1], [10, 2.5, 1], [10, 2.25, 1], [9.75, 2.25, 1], [9.5, 2.25, 1], [9.25, 2.25, 1], [9, 2.25, 1], [9, 2, 1], [9, 1.75, 1], [9, 1.5, 1], [9.25, 1.5, 1], [9.5, 1.5, 1], [9.75, 1.5, 1], [11, 3, 1], [11.25, 3, 1], [11.5, 3, 1], [11.75, 3, 1], [12, 3, 1], [12, 2.75, 1], [12, 2.5, 1], [12, 2.25, 1], [12, 2, 1], [12, 1.75, 1], [12, 1.5, 1], [11.75, 1.5, 1], [11.5, 1.5, 1], [11.25, 1.5, 1], [11, 1.5, 1], [11, 1.75, 1], [11, 2, 1], [11, 2.25, 1], [11, 2.5, 1], [11, 2.75, 1], [11.25, 2.25, 1], [11.5, 2.25, 1], [11.75, 2.25, 1]] # 得到五个点
plt.figure() # 得到画面
ax1 = plt.axes(projection='3d')
ax1.set_xlim(0, 15) # X轴, 横向向右方向
ax1.set_ylim(0, 15) # Y轴, 左向与X,Z轴互为垂直
ax1.set_zlim(0, 15) # 竖向为Z轴
color1 = ['r', 'g', 'b', 'k', 'm']
marker1 = ['o', 'v', '1', 's', 'H']
i = 0
for x in dot1:
    ax1.scatter(x[0], x[1], x[2], c=color1[i],
                marker=marker1[i], linewidths=4) # 用散点函数画点
    i += 1
plt.show()
```

运行得到



密码口令是 **365728**，解压压缩包，得到pdf文档



摩斯密码解密

CONGRATULATIONTHEFLAGICHALLENGEISCCCTWOZEROOTWOONE

根据提示最终flag为

ISCC{congratulationtheflagischallengeiscctwozerotwoone}

海市蜃楼-1

题目描述： 或许你看到的只是海市蜃楼...

下载附件，是个docx文档，看到了压缩包文件头PK

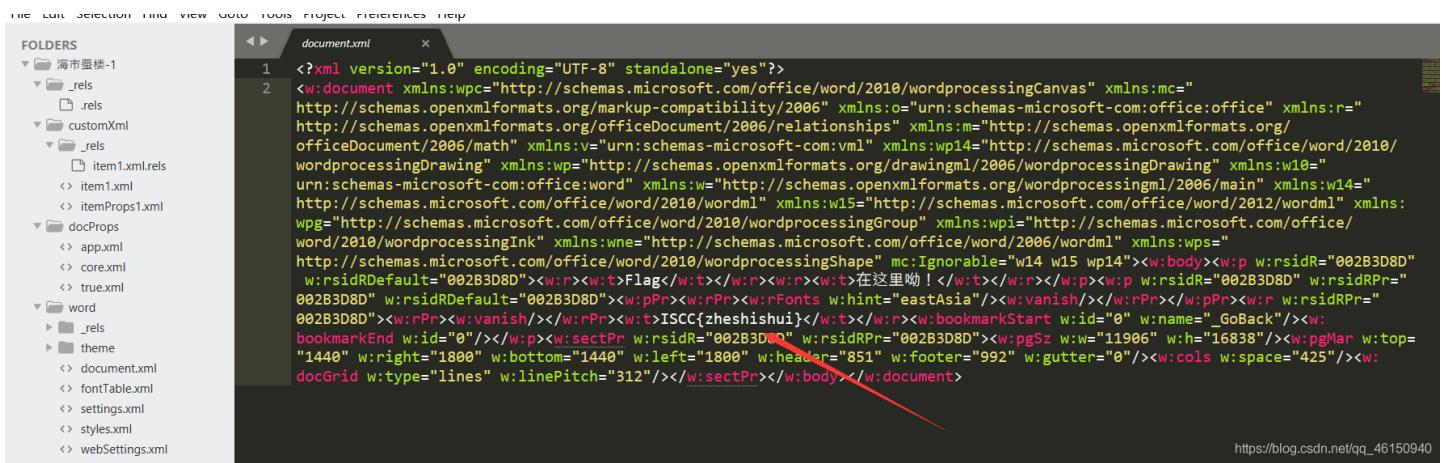
The image shows a Microsoft Word document with the file header "PK" visible at the top. Below the header, there is some Chinese text, likely the content of the compressed file's header.

```
PK
***** 热 eR*** **** * ****
* * 海市蜃楼-1/up * @c 潘娴峰競鋒寫七-1/PK
***** a 賽 R*** **** * **** * * 海市蜃樓-1/海市蜃樓-1/up#* 1 m≈捣甯傷潘
好?1/潘娴峰競鋒寫七-1/PK
```

***** ?

https://blog.csdn.net/qq_46150940

修改后缀为zip，解压，在document.xml文件中找到flag

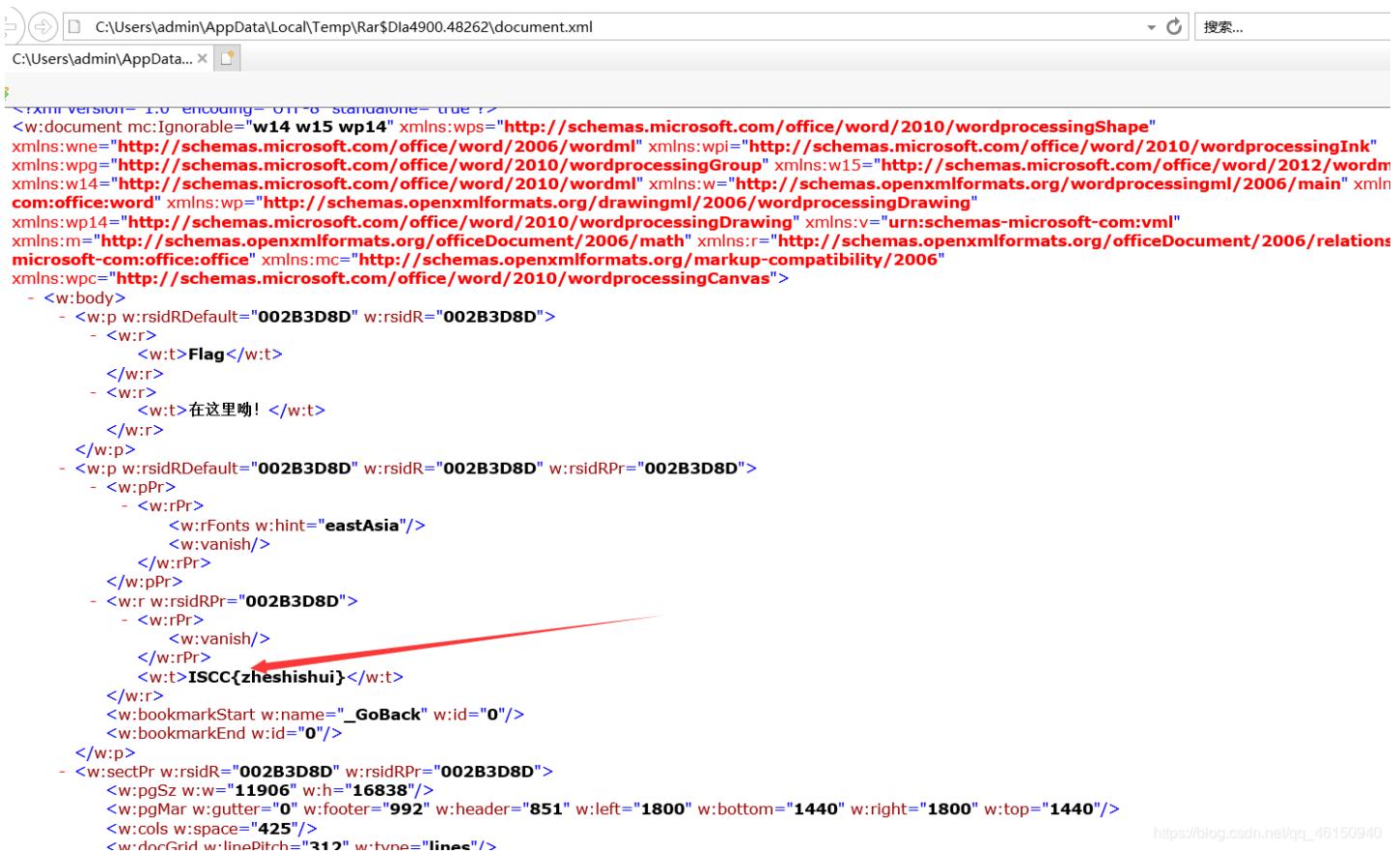


```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<w:document mc:Ignorable="w14 w15 wp14" xmlns:wps="http://schemas.microsoft.com/office/word/2010/wordprocessingShape"
  xmlns:wne="http://schemas.microsoft.com/office/word/2006/wordml" xmlns:wpi="http://schemas.microsoft.com/office/word/2010/wordprocessingInk"
  xmlns:wp="http://schemas.microsoft.com/office/word/2010/wordprocessingGroup" xmlns:w15="http://schemas.microsoft.com/office/word/2012/wordml"
  xmlns:w14="http://schemas.microsoft.com/office/word/2010/wordml" xmlns:w="http://schemas.openxmlformats.org/wordprocessingml/2006/main"
  xmlns:mc="http://schemas.openxmlformats.org/officeDocument/2006/math" xmlns:r="http://schemas.openxmlformats.org/markup-compatibility/2006"
  xmlns:office="urn:schemas-microsoft-com:office:office" xmlns:vm1="http://schemas.microsoft.com/office/word/2010/wordprocessingDrawing"
  xmlns:wp14="http://schemas.openxmlformats.org/drawingml/2006/wordprocessingDrawing" xmlns:w10="urn:schemas-microsoft-com:office:word"
  xmlns:ws="http://schemas.microsoft.com/office/word/2010/wordml" xmlns:w15="http://schemas.microsoft.com/office/word/2006/main"
  xmlns:w14="http://schemas.microsoft.com/office/word/2010/wordml" xmlns:wp="http://schemas.openxmlformats.org/wordprocessingml/2006/main"
  xmlns:wps="http://schemas.microsoft.com/office/word/2010/wordprocessingGroup" xmlns:wpi="http://schemas.microsoft.com/office/word/2010/wordprocessingInk"
  xmlns:wne="http://schemas.microsoft.com/office/word/2006/wordml" xmlns:wps="http://schemas.microsoft.com/office/word/2010/wordprocessingShape" mc:Ignorable="w14 w15 wp14"><w:body><w:p wrsidR="002B3D8D" wrsidRDefault="002B3D8D"><w:t>Flag</w:t></w:p><w:p wrsidR="002B3D8D" wrsidRPr="002B3D8D" wrsidRDefault="002B3D8D"><w:pPr><w:rFonts w:hint="eastAsia"/><w:vanish/></w:pPr><w:r w:rsidR="002B3D8D" w:rsidRPr="002B3D8D" w:rsidRDefault="002B3D8D"><w:rPr><w:vanish/></w:rPr><w:t>ISCC{zheshishui}</w:t></w:r><w:bookmarkStart w:id="0" w:name="_GoBack"/><w:bookmarkEnd w:id="0"/><w:pgSz w:w="11906" w:h="16838"/><w:pgMar w:top="1440" w:right="1800" w:bottom="1440" w:left="1800" w:header="851" w:footer="992" w:gutter="0"/><w:cols w:space="425"/><w:docGrid w:type="lines" w:linePitch="312"/></w:sectPr></w:body></w:document>
```

https://blog.csdn.net/qq_46150940

方法二：

使用winrar在文件中搜索ISCC字符串



```
<?xml version="1.0" encoding="UTF-8" standalone="true "?>
<w:document mc:Ignorable="w14 w15 wp14" xmlns:wps="http://schemas.microsoft.com/office/word/2010/wordprocessingShape"
  xmlns:wne="http://schemas.microsoft.com/office/word/2006/wordml" xmlns:wpi="http://schemas.microsoft.com/office/word/2010/wordprocessingInk"
  xmlns:wp="http://schemas.microsoft.com/office/word/2010/wordprocessingGroup" xmlns:w15="http://schemas.microsoft.com/office/word/2012/wordml"
  xmlns:w14="http://schemas.microsoft.com/office/word/2010/wordml" xmlns:w="http://schemas.openxmlformats.org/wordprocessingml/2006/main"
  xmlns:mc="http://schemas.openxmlformats.org/officeDocument/2006/math" xmlns:r="http://schemas.openxmlformats.org/markup-compatibility/2006"
  xmlns:office="urn:schemas-microsoft-com:office:office" xmlns:vm1="http://schemas.microsoft.com/office/word/2010/wordprocessingDrawing"
  xmlns:wp14="http://schemas.openxmlformats.org/drawingml/2006/wordprocessingDrawing" xmlns:w10="urn:schemas-microsoft-com:office:word"
  xmlns:ws="http://schemas.microsoft.com/office/word/2010/wordml" xmlns:w15="http://schemas.microsoft.com/office/word/2006/main"
  xmlns:w14="http://schemas.microsoft.com/office/word/2010/wordml" xmlns:wp="http://schemas.openxmlformats.org/wordprocessingml/2006/main"
  xmlns:wps="http://schemas.microsoft.com/office/word/2010/wordprocessingGroup" xmlns:wpi="http://schemas.microsoft.com/office/word/2010/wordprocessingInk"
  xmlns:wne="http://schemas.microsoft.com/office/word/2006/wordml" xmlns:wps="http://schemas.microsoft.com/office/word/2010/wordprocessingShape" mc:Ignorable="w14 w15 wp14"><w:body><w:p wrsidR="002B3D8D" wrsidRDefault="002B3D8D"><w:t>Flag</w:t></w:p><w:p wrsidR="002B3D8D" wrsidRPr="002B3D8D" wrsidRDefault="002B3D8D"><w:pPr><w:rFonts w:hint="eastAsia"/><w:vanish/></w:pPr><w:r w:rsidR="002B3D8D" w:rsidRPr="002B3D8D" w:rsidRDefault="002B3D8D"><w:rPr><w:vanish/></w:rPr><w:t>ISCC{zheshishui}</w:t></w:r><w:bookmarkStart w:id="0" w:name="_GoBack"/><w:bookmarkEnd w:id="0"/><w:pgSz w:w="11906" w:h="16838"/><w:pgMar w:top="1440" w:right="1800" w:bottom="1440" w:left="1800" w:header="851" w:footer="992" w:gutter="0"/><w:cols w:space="425"/><w:docGrid w:type="lines" w:linePitch="312"/></w:sectPr></w:body></w:document>
```

https://blog.csdn.net/qq_46150940

区块链

参考: <https://terminalcats.com/0x41414141-ctf-blockchain-sanity-check-400/>

```
pragma solidity ^0.7.0; // 指定所需的编译器版本
// SPDX-License-Identifier: UNLICENSED

contract look_look {
    function sloc111() public pure returns (string memory){ //public智能合约外部和内部都可使用的方法
        return "flag{}";
    }
}
```

合约地址: 0x0ed72dfd4c63dd97df8fec07e5a6bba466c6adf5

需要挂代理, 访问<https://rinkeby.etherscan.io/>

第一次做智能合约方面的题, 输入合约地址, 然后点击Contract

The screenshot shows the Etherscan interface for the Rinkeby Testnet Network. At the top, there's a search bar with the placeholder "Search by Address / Txn Hash / Block / Token / Ens". Below it, a navigation bar includes "All Filters", "Home", "Blockchain", "Tokens", "Misc", and "Rinkeby". A red arrow points from the search bar towards the "Contract" tab in the navigation bar. The main content area has two sections: "Contract Overview" on the left and "More Info" on the right. In the "Contract Overview" section, it says "Balance: 0 Ether". In the "More Info" section, it lists "My Name Tag: Not Available" and "Contract Creator: 0x3598f8763bd3af6d6a... at txn 0xb091eb263e976c30f4...". Below these sections, there are tabs for "Transactions", "Internal Txns", "Contract" (which is highlighted with a red arrow), and "Events". A note at the bottom encourages users to verify their contracts: "Are you the contract creator? Verify and Publish your contract source code today!".

点击Decompile ByteCode, 到这就卡住了

The screenshot shows the Etherscan bytecode decompiler for the same contract. At the top, it displays the URL "rinkeby.etherscan.io/bytecode-decompiler?a=0x0ed72dfd4c63dd97df8fec07e5a6bba466c6adf5". Below the URL, it says "Not available or unverified.". The main content area shows the decompiled bytecode, which is extremely long and complex, starting with "0x60806040523480156100105760008fd5b506004361061002b5760003560e01c8063582840f214610030575b600080fd5b6100386100b3565b604051808060200182810382528381815181526020019150805190602001908083836005b8381101". A red arrow points from the "Decompile ByteCode" button at the top right down to the bottom of the page where the decompiled code is listed. The decompiled code is a series of assembly-like instructions, mostly zeros and ones, representing the contract's logic.

第一个二维码：

U2FsdGVkX1/Ka+sScszwQkwh0+VLiJwV/6IFg5W+TfNHGxG2qZsIr2iwMwb9X9Iu
3GuGwmPOtO27z8vNppD2D50fwsD+8VWhdtW9J4cewYivH/Z/7GoUvcJXJMrvf+vu
+CBqWDGp6HWd0e5whGhuzlK0ZtBcDzdPDSIHA7+GuUlifp8PcFCtJPgiuk143REE
+pKFiSJXo1XLR1vJCdGY9w5mXFbiWPrb2U7r/v5noP8=

第二个二维码：

U2FsdGVkX19e0Y/pDh8+vPAcvfkLi1XLUneVzjLLOMu153sKK8UpobdCoiPIv4KE

图片属性给的提示，AES加密，密钥是ISCC2021

Try AES, and you will get the flag. ISCC2021

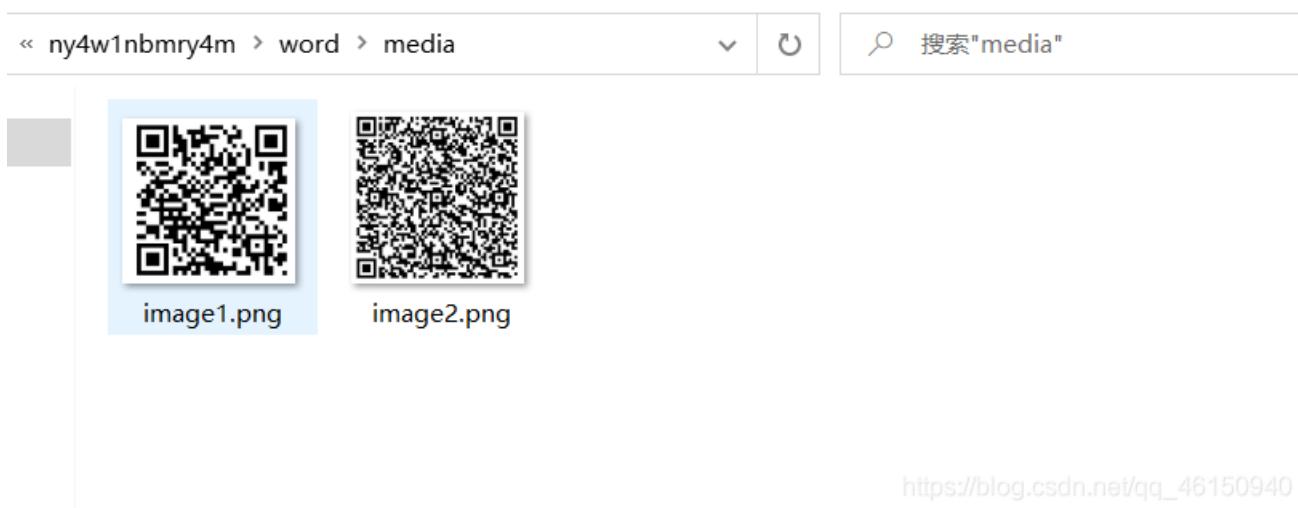
对第一个二维码内容，解了三次AES加密得到

y0u_h@ve_fall1en_int0_tHe_tr@p_0f_tHe_be@uty_!

第二个二维码内容，进行AES解密发现不行，然后尝试DES解密结果成功了，得到flag

ISCC{be@uty_1ike\$_Y0u_2021ISCC}

再复现的时候，附件中就剩一张二维码了，把docx后缀转换成zip解压，出题人变狗了，把第二张二维码藏在这了，和上面的方法一样，得到flag



我的折扣是多少

题目描述：

小c同学去参加音乐会，在官网买票时发现了有提示消息，提供给的有“give_me_discount”的压缩包，好奇的小c下载下来，但却无从下手，为了节省零花钱，你能帮帮他吗？

命令行运行give.exe，得到

pass1{\u006b\u0072\u0077}

把中间的进行unicode转码得到： pass1{krw}

010打开me.zip，在末尾发现Base编码

启动 me.zip x

编辑为: Hex 运行脚本 Run Template: ZIP.b1 ↻

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	0123456789ABCDEF
0000h:	50	4B	03	04	14	00	09	00	08	00	5C	99	71	52	D3	97	PK.....\m{qRÓ-
0010h:	BD	30	1E	00	00	00	10	00	00	00	0F	00	00	00	77	68	\20.....wh
0020h:	61	74	27	73	5F	6D	65	61	6E	2E	74	78	74	4C	C0	64	at's mean.txtLÀd
0030h:	A8	3E	C5	5E	4C	A8	9B	57	0A	E1	CD	6C	56	D2	E8	87	">À^L">W.áílVÒè‡
0040h:	E4	2B	33	24	06	B1	EE	22	7F	62	2E	50	4B	01	02	14	ä+3\$.±î".b.PK...
0050h:	00	14	00	09	00	08	00	5C	99	71	52	D3	97	BD	30	1E\m{qRÓ-\20.
0060h:	00	00	00	10	00	00	00	0F	00	24	00	00	00	00	00	00\$.....
0070h:	00	20	00	00	00	00	00	00	00	77	68	61	74	27	73	5Fwhat's_
0080h:	6D	65	61	6E	2E	74	78	74	0A	00	20	00	00	00	00	00	mean.txt..
0090h:	01	00	18	00	0F	63	63	33	1E	1B	D7	01	0F	63	63	33cc3..x..cc3
00A0h:	1E	1B	D7	01	85	DA	F4	9A	0C	1B	D7	01	50	4B	05	06	..x...Úôš..x.PK..
00B0h:	00	00	00	00	01	00	01	00	61	00	00	00	4B	00	00	00a...K...
00C0h:	00	00	63	47	46	7A	63	7A	4A	37	5A	32	4E	6A	4E	6A	.cGFzczJ7Z2NjNj
00D0h:	59	32	66	51	3D	3D										Y2fQ==	

https://blog.csdn.net/qq_46150940

base64解码得到: pass2{gcc666}

所以压缩包口令为krwgcc666，解压得到

ew91Zm91bmRtZT8=

base64解码

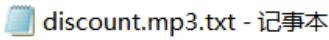
eW91Zm91bmRtZT8=

youfoundme?

https://blog.csdn.net/qq_46150940

最后使用MP3Stego

decode -X -P youfoundme? discount.mp3



文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)

ISCC{LFXXK4TENFZWG33VNZ2DELRRGU=====}

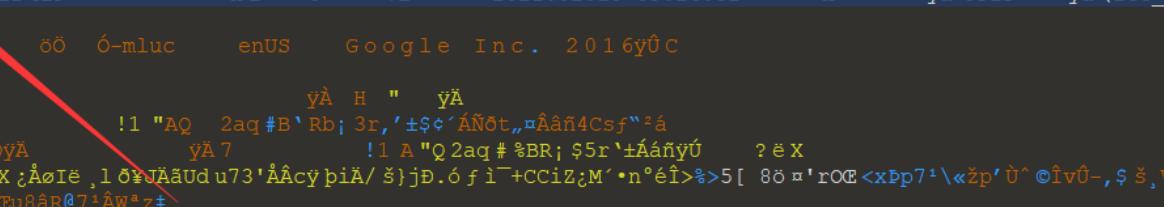
base32解码，最终flag为

ISCC{Yourdiscount2.15}

上海市蜃楼-2

foremost分离图片，得到加密的压缩包

由于这是组合题，所以要在海市蜃楼-1中去找线索。最终发现true.xml有猫腻，原来是jpg文件



启动 app.xml core.xml true.xml* x

编辑为: XML 运行脚本 运行模板

0 10 20 30 40 50 60 70 80 90 100

1 yðýá „Exif MM *

2 Y Đ

3 [Xíz öö Ó-mluc enUS Google Inc. 2016yÚC

4

5 yÚC yÀ H " yÁ

6 yÁT !1 "AQ 2aq #B' Rb; 3r,'±\$¢' Áñt,,¤Áññ4Csf" ^á

7 %&c'f'sÓyÄ yÁ7 !1 A "Q 2aq # %BR; \$5r'±ÁñyÚ ? eX

8 \õlÁér,X ãòíE , l õyñääud u73 'ÁAcý piÄ/ š)jD.ó f i-+CCiZ;M' •n°éf>%>5[8ö n'rOe <xþp7^1\«žp' Ú^ @ívÛ-, \$ š, vÛs| s~sn2,

9 ^' ãÖc‰NEü@R@? Áwaz#

10 FCQ Úyø<' >fÛçlYøôàa W7j Š,É Ù...4 ÁxZ o °iñu ·) ñ >,-kwj <@< ðy^a-RB ^éyq -^G?) š

11 pö@7Ùr³xñö+ " õäøø Sö ' É /iÜGž ybR >Mz\x^--tå dâ3#59 \$) ' ¥() qe YM'; ' Ñt, ä + èÚf \$øçm% -^äe>+uÃiCr3 «qÀYøµ€R ^é(C2ç‰;

12 HÃzvgÖzFrÉ; <sy 2 Knçj j Y8Èx ^#8i-oö ' Ášç' ^é "íf>îQYd7±_ òuÃi yðfå9 9%šoÉ°Mž Bø žò>ôpÈ Yéö ^23A åž I é "i.8yžö' >t1

13 Y ^-< \nóuáÖèéOaö - Y ' è Ys yðvàla^uIT ^çùVey GSÁ^ "ew GÁé^ } b&U^VB %^4ö È!hP>{ , ýlÄyCwR5; , ^æ^Lý^vP> pëR] ^?L] ç^f

14 Awö;çé- ' { õqé} | õéYjd^, , ËJIAfèYf^@:dþ+^m' É^*vOdþ

15 má, ' ouím ' òaÈrC, ¶(oýç#^5 %, " ßy òx \i òøpù÷äTeç) S'užä|söEi# Z 1¶(díqExKf-i,) in^Jg^rý ^" ð òAs. éd^2YJb (ÂA<a

16 sñ#)) %4öf " # õIr@y¶=Gx" 2â2Wf-^, oåT ' hñwl'L-B^!š, ^o... ÁòU: ' sùâšçç ^z-Kûâ.oó > á-Ù: bÉ! ' ïK~iö = ÚFë*^" ^á^ç ôUi^y

17 [*< zm, ÕE^ž@i^t^mñe^ ^úz^yûs:sA>- Ù. [Ñçv" ð^øtòÜL^ &<XçR' A< lcf : 4RDhn%u oé=S@æV-çâ úZ[Cedç Ny y-Ù' x, ç 2â2#H

18 t^, Á3BÝfio^Mô" y 8 èm KíÜÛ\g^g;

19 Nñæün Á' sô !

20 ÇEYšiÙ^7W#^ | šc‰qÅ

21 ÇKåöDfæ Ýék=ö, TÝ úúY ^

22 S: ÁK, ¶a aó ŠÄçÛáR" ¶BâAsÖicûnÙ/Irwpi- | [b^m" " ÚÉq¾iÂ•n" t>ùä _Zí+&+-, eK‰ | <ùù, ò^ó^ *e/+r-ë } žó' ^ò ; RWA; xb=mp^kî ŽE

23 O- òQF5, ÁY s" " ïkäñi! < y ðo ùž ^ o^ < V Jw 4fý ÁF ÒR,, ¶úy 1<äiešE' Úù": Èz; u/ ÷ üsøx- ^nc- ^çœÄ...ü*ü^u^A E e - ^@?# N©fUFEæöi y

24 Kjo- ìm f rÈg óvL ži3Nu #í^s ^SB^ñ' ^k' M í hýAœJTU, úty þ^+voz f Yùt^ yÁY, Eymšûí 7cæÜÓ Cmb! I# ùââb ^çœpb" nò ~ á- L~i< õa

25 Yoñ^sfañù, 2N\$ f^ | ãAíç? ^y ÚççKíYÚ ÜÜš ñu Rn, " ò-^o^6»ea https://blog.csdn.net/qq_46150940

修改为jpg





https://blog.csdn.net/qq_46150940

海市蜃楼-1的flag也有猫腻，结合flag和图片，最终得到压缩包的口令为zhongnanshan，解压海市蜃楼-2的压缩包



https://blog.csdn.net/qq_46150940

扫码得到一个网址，内容为 `ZWFzeQ==`，base解码后为 `easy`

<https://ctewm.com/aTvQcE/kva11w>

继续往下走，foremost分离图片得到压缩包，口令为easy

名称	修改日期	类型	大小
这才是真的.txt	2021/3/5 20:10	文本文件	1 KB

Hack the Victim

题目

Victim为含有漏洞的智能合约，在 Rinkeby 测试网络的合约地址为：

0x68D28fE315E6A344029D42915Fbc7af4261AB833

接口为：

```
contract Victim {  
function withdraw() public returns (string memory ){  
return "ISCC{xxxxx}";  
}  
}
```

请编写攻击合约，实现对 Victim 的攻击，获取 flag。

合约地址

<https://rinkeby.etherscan.io/address/0x68D28fE315E6A344029D42915Fbc7af4261AB833>

16进制转字符串

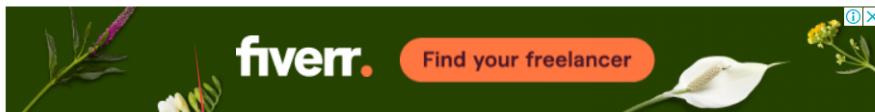
16进制到文本字符串

加密或解密字符串长度不可以超过10M

当前长度: 3654

3

16进制转字符 字符转16进制 测试用例 清空结果 复制结果



- 1 Ⓛ·a·W·6d·a·bW·5·c·h珍a·gW將·a·僑·x·a·m·a·SW殃·a·T·□·a·JW·@盈□□□·>C·V·@Q·..□a·h·FV·@Q·..□a·堵LVI·@Q·..□a·=·□·a·W·@盈□□□·>C·TV·@Q ..□a·[a·IV·@QBC..`·豐[墨].·□AP·h·轉.

2 ...QP[PQ□a·a·..□·a·엠·@盈□□□·>C·..□·R·`·dV·[·T·曠·R·`·`·dV[·`·涤□□□□·P·TTa·W·@Q·..`·畢j·R·□have executed the withdrawl·`

3 `·T·a·TW·d·T·a·JW·`·SV·`·T·`·3s□□□盈□□□·P·T·a·..□·看·T·`·3s□□□盈□□□·P·T·a·W·`·3s□□□盈□□□·P·`·..□·吠·3s□□□·@QA·`·@Q·ZdeP=✓·ēP·`·?·=·e·^cV·`·AP·

4 咻□溼地錢\$鵠·@Q□□□hs□□□·`·嗣Q·P@Q3·`·`·3s□□□盈□□□·P·T·`·□□□hs□□□·`·`·T·`·□□□hs□□□·`·`·T·a·iW·@Q·..`·畢j·`·`·Ed to withdraw·`·..□·`·@Q·Rt·..`·ISCC(h@ve fun~Re-EntRan(y).....△`·`·□□□□□□·`·`·TK·Ir1X 戢刀鋒)口囉·[4dsolcC...2

也可以直接点击**Decompile ByteCode**进行反编译

An Ethereum Virtual Machine (EVM) decompiler for extracting information from Runtime bytecode and presenting it in a more human-readable form. Useful for debugging smart contracts where the original source code is not available or unverified.

Attribution: This decompiler uses the [Panoramix decompiler](#) created by [@Tomasz Kolinko](#)

Decompile Bytecode

ByteCode Decomilation Result:

```
39 def _fallback() payable: # default function
40     revert
41
42 def withdraw() payable:
43     if balances[caller]:
44         revert with 0, 'you have executed the withdrawal'
45     if success > 10:
46         if success >= 100:
47             stor4 = 2
48         else:
49             stor4 = 3
50     if unknown0568e65e[caller] == stor4:
51         success++
52     if unknown0568e65e[caller] < stor4:
53         unknown0568e65e[caller]++
54     call caller with:
55         value stor3 wei
56         gas gas_remaining wei
57         log 0xae00e6674: caller, stor3, b'01(ext_call.success)'
58 balances[caller] = stor3 * unknown0568e65e[caller]
59 if balances[caller] < stor3:
60     revert with 0, 'failed to withdraw'
61 return 'ISCC{@ve_fun-Re-EntRan(y)'
```

检查一下

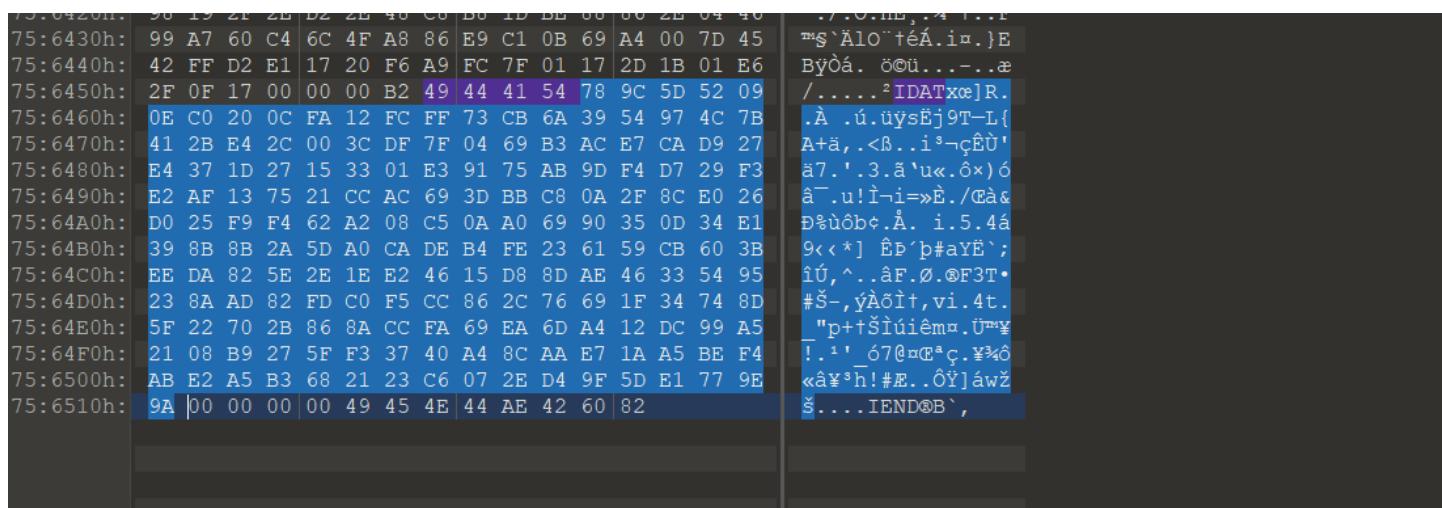
题目描述：你真的了解png文件的格式吗？

用pngcheck检查一下图片，发现最后一个IDAT块异常

```
C:\Windows\System32\cmd.exe
chunk IDAT at offset 0x742bba, length 8192
chunk IDAT at offset 0x744bc6, length 8192
chunk IDAT at offset 0x746bd2, length 8192
chunk IDAT at offset 0x748bde, length 8192
chunk IDAT at offset 0x74abea, length 8192
chunk IDAT at offset 0x74cbf6, length 8192
chunk IDAT at offset 0x74ec02, length 8192
chunk IDAT at offset 0x750c0e, length 8192
chunk IDAT at offset 0x752c1a, length 8192
chunk IDAT at offset 0x754c26, length 6181 ←
chunk IDAT at offset 0x756457, length 178
chunk IEND at offset 0x756515, length 0
No errors detected in 221B.png (942 chunks, 79.0% compression).
D:\CTF\MISC\隐写\pngcheck-3.0.2-win32>
```

https://blog.csdn.net/qq_46150940

010复制出来



75:6420h: 90 19 21 2E B2 2E 40 C0 B6 1D BE 00 00 2E 04 40
75:6430h: 99 A7 60 C4 6C 4F A8 86 E9 C1 0B 69 A4 00 7D 45
75:6440h: 42 FF D2 E1 17 20 F6 A9 FC 7F 01 17 2D 1B 01 E6
75:6450h: 2F 0F 17 00 00 00 B2 49 44 41 54 78 9C 5D 52 09
75:6460h: 0E C0 20 0C FA 12 FC FF 73 CB 6A 39 54 97 4C 7B
75:6470h: 41 2B E4 2C 00 3C DF 7F 04 69 B3 AC E7 CA D9 27
75:6480h: E4 37 1D 27 15 33 01 E3 91 75 AB 9D F4 D7 29 F3
75:6490h: E2 AF 13 75 21 CC AC 69 3D BB C8 0A 2F 8C E0 26
75:64A0h: D0 25 F9 F4 62 A2 08 C5 0A A0 69 90 35 0D 34 E1
75:64B0h: 39 8B 8B 2A 5D A0 CA DE B4 FE 23 61 59 CB 60 3B
75:64C0h: EE DA 82 5E 2E 1E E2 46 15 D8 8D AE 46 33 54 95
75:64D0h: 23 8A AD 82 FD C0 F5 CC 86 2C 76 69 1F 34 74 8D
75:64E0h: 5F 22 70 2B 86 8A CC FA 69 EA 6D A4 12 DC 99 A5
75:64F0h: 21 08 B9 27 5F F3 37 40 A4 8C AA E7 1A A5 BE F4
75:6500h: AB E2 A5 B3 68 21 23 C6 07 2E D4 9F 5D E1 77 9E
75:6510h: 9A |00 00 00|00 49 45 4E|44 AE 42 60|82

查找结果	
地址	值
7468D2h	IDAT
748BDEh	IDAT
74ABEAh	IDAT
74CBF6h	IDAT
74EC02h	IDAT
750C0Eh	IDAT
752C1Ah	IDAT
754C26h	IDAT
756457h	IDAT
939	

https://blog.csdn.net/qq_46150940

使用脚本解压zlib

```

import zlib
s = ''
78 9C 5D 52 09 0E C0 20 0C FA 12 FC FF 73 CB 6A
39 54 97 4C 7B 41 2B E4 2C 00 3C DF 7F 04 69 B3
AC E7 CA D9 27 E4 37 1D 27 15 33 01 E3 91 75 AB
9D F4 D7 29 F3 E2 AF 13 75 21 CC AC 69 3D BB C8
0A 2F 8C E0 26 D0 25 F9 F4 62 A2 08 C5 0A A0 69
90 35 0D 34 E1 39 8B 8B 2A 5D A0 CA DE B4 FE 23
61 59 CB 60 3B EE DA 82 5E 2E 1E E2 46 15 D8 8D
AE 46 33 54 95 23 8A AD 82 FD C0 F5 CC 86 2C 76
69 1F 34 74 8D 5F 22 70 2B 86 8A CC FA 69 EA 6D
A4 12 DC 99 A5 21 08 B9 27 5F F3 37 40 A4 8C AA
E7 1A A5 BE F4 AB E2 A5 B3 68 21 23 C6 07 2E D4
9F 5D E1 77 9E 9A
...
s = s.replace(' ', '').replace('\n', '')
b = bytes.fromhex(s)
flag = zlib.decompress(b)
print(flag)

```

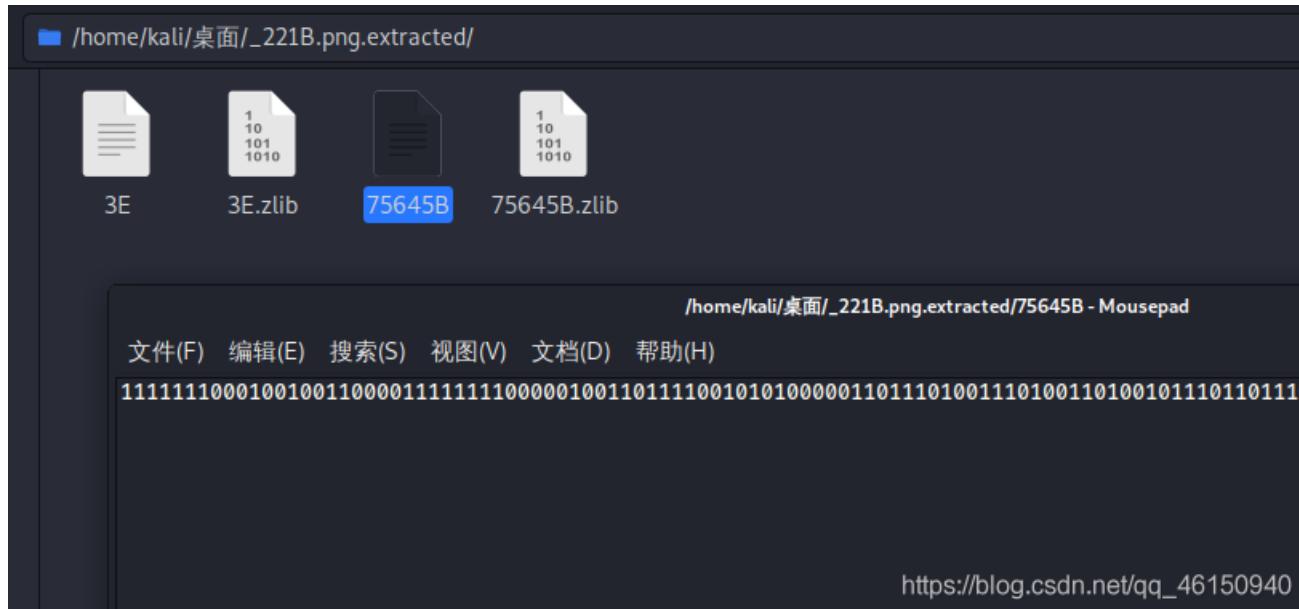
得到

```

1111110001001001100001111111000001001101110010101000001101110100111010010111011011101000000111011110101
110110111010010111011100010111011000010111011100010010000011111111010101010101011111000000000111001001101
00000000100101101110000101010100001101000011101011011100101000001110010010001111011100100011000010110110
01111110010101101011100001000111101001100010010101110000111111111100101001101010100111110010110001100
11011111011100011001111001000111100111000010001101000011010010000010110010101011101000110110100111000111010
0111101111100010100110101101001010100011110100000000111011011010001011011111100001111000010101101010000
0101011110010101000101101011101001010011001011110010101110101101011110000011010011011010010101000010100111011
0000010000111010000110010010111111010010010000011101001110100101110101011110000011010011011010010101000010100111011

```

直接binwalk图片，也能得到同样的结果



一共841位，猜测是29*29的二维码，直接使用脚本

扫码得到flag

变异的SM2

附件

```
#!/usr/bin/python3

# server.py
from gmssl import func, sm2
# from flag import FLAG
FLAG = '{testFlag}'

sm2p256v1_ecc_table = {
    'n': 'FFFFFFFFFFFFFFFFFFFFFF7203DF6B21C6052B53BBF40939D54123',
    'p': 'FFFFFFFFFFFFFFFFFFFFFF7203DF6B21C6052B53BBF40939D54123',
    'g': '32c4ae2c1f1981195f9904466a39c9948fe30bbff2660be1715a4589334c74c7' +
          'bc3736a2f4f6779c59bdcee36b692153d0a9877cc62a474002df32e52139f0a0',
    'a': 'FFFFFFFFFFFFFFFFFFFFFF7203DF6B21C6052B53BBF40939D54123',
    'b': '28E9FA9E9D9F5E344D5A9E4BCF6509A7F39789F515AB8F92DDBCBD414D940E93'
}

n = 'FFFFFFFFFFFFFFFFFFFFFF7203DF6B21C6052B53BBF40939D54123'
G = '32c4ae2c1f1981195f9904466a39c9948fe30bbff2660be1715a4589334c74c7' \
    'bc3736a2f4f6779c59bdcee36b692153d0a9877cc62a474002df32e52139f0a0'

def sign(tsm2):
    data = func.random_hex(len(n))
    k1_str = func.random_hex(len(n))
    print(tsm2.send_p1(data, k1_str))
    backdoor = input('backdoor:').strip()
    result = tsm2.output_p1(k1_str, backdoor)
    print(result)

def verify(tsm2):
    message = input('msg:').strip().encode().strip(b'\x00')
    sign = input('sign:').strip().encode().strip(b'\x00')
    check = tsm2.verify(sign, message)
    if check is True and message == b'Hello, Welcome to ISCC2021!':
        print(FLAG)
```

```

    print(FLAGS)
else:
    print(check)

class TSM2(object):
    def __init__(self, sk):
        ecc_table = sm2p256v1_ecc_table
        self.ecc_table = ecc_table
        self.n = int(ecc_table['n'], 16)
        self.para_len = len(ecc_table['n'])
        self.ecc_a3 = (int(ecc_table['a'], base=16) + 3) % int(ecc_table['p'], base=16)

        self.sk = int(sk, 16)
        self.pk = self._kg(self.sk, ecc_table['g'])

        self.sks = int(func.random_hex(self.para_len), 16)
        self.pks = pow((self.sk + 1) * self.sks, self.n - 2, self.n) % self.n

    def send_p1(self, data, k1_str):
        e = int(data, 16)
        k1 = int(k1_str, 16)
        k1 = k1 % self.n
        R1 = self._kg(k1, self.ecc_table['g'])
        return '%064x%0128s' % (e, R1)

    def output_p1(self, k1_str, r_s2_s3):
        r = int(r_s2_s3[0:self.para_len], 16)
        s2 = int(r_s2_s3[self.para_len:2 * self.para_len], 16)
        s3 = int(r_s2_s3[2 * self.para_len:], 16)

        k1 = int(k1_str, 16)
        d1 = self.sks
        s = (d1 * k1 * s2 + d1 * s3 - r) % self.n
        if s == 0 or s == (self.n - r):
            return None
        return '%064x%064x' % (r, s)

    def verify(self, Sign, data):
        r = int(Sign[0:self.para_len], 16)
        s = int(Sign[self.para_len:2 * self.para_len], 16)
        e = int(data.hex(), 16)
        t = (r + s) % self.n
        if t == 0:
            return 0

        P1 = self._kg(s, self.ecc_table['g'])
        P2 = self._kg(t, self.pk)

        if P1 == P2:
            P1 = '%s%s' % (P1, 1)
            P1 = self._double_point(P1)
        else:
            P1 = '%s%s' % (P1, 1)
            P1 = self._add_point(P1, P2)
            P1 = self._convert_jacb_to_nor(P1)

        x = int(P1[0:self.para_len], 16)
        return r == ((e + x) % self.n)

    def _kg(self, k, Point):

```

```

if (k % self.n) == 0:
    return '0' * 128
Point = '%s%s' % (Point, '1')
mask_str = '8'
for i in range(self.para_len - 1):
    mask_str += '0'
mask = int(mask_str, 16)
Temp = Point
flag = False
for n in range(self.para_len * 4):
    if flag:
        Temp = self._double_point(Temp)
    if (k & mask) != 0:
        if flag:
            Temp = self._add_point(Temp, Point)
        else:
            flag = True
            Temp = Point
    k = k << 1
return self._convert_jacb_to_nor(Temp)

def _double_point(self, Point):
    l = len(Point)
    len_2 = 2 * self.para_len
    if l < self.para_len * 2:
        return None
    else:
        x1 = int(Point[0:self.para_len], 16)
        y1 = int(Point[self.para_len:len_2], 16)
        if l == len_2:
            z1 = 1
        else:
            z1 = int(Point[len_2:], 16)

        T6 = (z1 * z1) % int(self.ecc_table['p'], base=16)
        T2 = (y1 * y1) % int(self.ecc_table['p'], base=16)
        T3 = (x1 + T6) % int(self.ecc_table['p'], base=16)
        T4 = (x1 - T6) % int(self.ecc_table['p'], base=16)
        T1 = (T3 * T4) % int(self.ecc_table['p'], base=16)
        T3 = (y1 * z1) % int(self.ecc_table['p'], base=16)
        T4 = (T2 * 8) % int(self.ecc_table['p'], base=16)
        T5 = (x1 * T4) % int(self.ecc_table['p'], base=16)
        T1 = (T1 * 3) % int(self.ecc_table['p'], base=16)
        T6 = (T6 * T6) % int(self.ecc_table['p'], base=16)
        T6 = (self.ecc_a3 * T6) % int(self.ecc_table['p'], base=16)
        T1 = (T1 + T6) % int(self.ecc_table['p'], base=16)
        z3 = (T3 + T3) % int(self.ecc_table['p'], base=16)
        T3 = (T1 * T1) % int(self.ecc_table['p'], base=16)
        T2 = (T2 * T4) % int(self.ecc_table['p'], base=16)
        x3 = (T3 - T5) % int(self.ecc_table['p'], base=16)

        if (T5 % 2) == 1:
            T4 = (T5 + ((T5 + int(self.ecc_table['p'], base=16)) >> 1) - T3) % int(self.ecc_table['p'], base=16)
        else:
            T4 = (T5 + (T5 >> 1) - T3) % int(self.ecc_table['p'], base=16)

        T1 = (T1 * T4) % int(self.ecc_table['p'], base=16)
        y3 = (T1 - T2) % int(self.ecc_table['p'], base=16)

```

```

        form = '%%0%dx' % self.para_len
        form = form * 3
        return form % (x3, y3, z3)

def _add_point(self, P1, P2):
    if P1 == '0' * 128:
        return '%s%s' % (P2, '1')
    if P2 == '0' * 128:
        return '%s%s' % (P1, '1')
    len_2 = 2 * self.para_len
    l1 = len(P1)
    l2 = len(P2)
    if (l1 < len_2) or (l2 < len_2):
        return None
    else:
        X1 = int(P1[0:self.para_len], 16)
        Y1 = int(P1[self.para_len:len_2], 16)
        if l1 == len_2:
            Z1 = 1
        else:
            Z1 = int(P1[len_2:], 16)
        x2 = int(P2[0:self.para_len], 16)
        y2 = int(P2[self.para_len:len_2], 16)

        T1 = (Z1 * Z1) % int(self.ecc_table['p'], base=16)
        T2 = (y2 * Z1) % int(self.ecc_table['p'], base=16)
        T3 = (x2 * T1) % int(self.ecc_table['p'], base=16)
        T1 = (T1 * T2) % int(self.ecc_table['p'], base=16)
        T2 = (T3 - X1) % int(self.ecc_table['p'], base=16)
        T3 = (T3 + X1) % int(self.ecc_table['p'], base=16)
        T4 = (T2 * T2) % int(self.ecc_table['p'], base=16)
        T1 = (T1 - Y1) % int(self.ecc_table['p'], base=16)
        Z3 = (Z1 * T2) % int(self.ecc_table['p'], base=16)
        T2 = (T2 * T4) % int(self.ecc_table['p'], base=16)
        T3 = (T3 * T4) % int(self.ecc_table['p'], base=16)
        T5 = (T1 * T1) % int(self.ecc_table['p'], base=16)
        T4 = (X1 * T4) % int(self.ecc_table['p'], base=16)
        X3 = (T5 - T3) % int(self.ecc_table['p'], base=16)
        T2 = (Y1 * T2) % int(self.ecc_table['p'], base=16)
        T3 = (T4 - X3) % int(self.ecc_table['p'], base=16)
        T1 = (T1 * T3) % int(self.ecc_table['p'], base=16)
        Y3 = (T1 - T2) % int(self.ecc_table['p'], base=16)

        form = '%%0%dx' % self.para_len
        form = form * 3
        return form % (X3, Y3, Z3)

def _convert_jacb_to_nor(self, Point):
    len_2 = 2 * self.para_len
    x = int(Point[0:self.para_len], 16)
    y = int(Point[self.para_len:len_2], 16)
    z = int(Point[len_2:], 16)
    z_inv = pow(z, int(self.ecc_table['p'], base=16) - 2, int(self.ecc_table['p'], base=16))
    z_invSquar = (z_inv * z_inv) % int(self.ecc_table['p'], base=16)
    z_invQube = (z_invSquar * z_inv) % int(self.ecc_table['p'], base=16)
    x_new = (x * z_invSquar) % int(self.ecc_table['p'], base=16)
    y_new = (y * z_invQube) % int(self.ecc_table['p'], base=16)
    z_new = (z * z_inv) % int(self.ecc_table['p'], base=16)
    if z_new == 1:

```

```

        form = '%%0%dx' % self.para_len
        form = form * 2
        return form % (x_new, y_new)
    else:
        return None

if __name__ == '__main__':
    sk = func.random_hex(len(sm2p256v1_ecc_table['n']))
    tsm2 = TSM2(sk)
    print('pk:%s' % tsm2.pk)
    print('pks:%064x' % tsm2.pks)
    for i in range(10):
        op = input('op: ').strip()
        if op == 'sign':
            sign(tsm2)
        elif op == 'verify':
            verify(tsm2)
        else:
            print("""sign: sign message
verify: verify message""")

```

暂时没做出来

混乱的音频

题目描述:

小明正在整理包含bed、bird、cat、dog、down、eight的音频文件数据，可小明不小心把分类好的一部分数据集弄混了。文件名非常混乱，无法判别出那些文件属于哪个单词类别。碰巧小红来问小明考试复习资料的密码。小明说：“我记得这个密码都是大写的英文字母，并且密码和我的爱好息息相关，具体的我也记不清了。你要是能帮我把这些音频文件分好类，你应该就会发现密码。”小红了解到小明是个计算机专业的学生，平时经常搜集一些与信息技术有关的新闻资讯。他经常参加ctf比赛，对密码学十分熟悉，对数字非常敏感。为了小红能得到考试复习资料，你能帮助小红将混乱的音频文件分好类并获取密码吗？

提示：比例 数据下载链接：<https://pan.baidu.com/s/1r8C1FByHpgNZJsaUkjASOw> 提取码：p1yh

小明的宠物兔

题目描述:

小明的宠物兔总是发出一些神秘的声音，小明很想知道兔兔在干什么，你能帮他翻译一下吗？

一张图片rabbit.txt，提示碰撞，应该是CRC32碰撞，foremost分离出压缩包

flag.txt内容，加salt，应该是rabbit加密

U2FdGVkX18kNy7R1BvcV9WJsqa+oxvdd0Ir86U2cU2996N61tZi7VV0aw==

CRC32碰撞，得到加密压缩包key.zip内容，

```
└─(kali㉿kali)-[~/桌面/Python/CRC32]
└$ python3 crc32.py reverse 0x3dacac6b
4 bytes: {0x47, 0x18, 0x87, 0xce}
verification checksum: 0x3dacac6b (OK)
5 bytes: (0_0) (OK)
5 bytes: DCr4m (OK)
6 bytes: 1QhloU (OK)
6 bytes: 3mmr6H (OK)
6 bytes: 49Gqqk (OK)
6 bytes: 5Uumn6 (OK)
6 bytes: 7ips7+ (OK)
6 bytes: 8Gpbyp (OK)
6 bytes: 9G1Sbi (OK)
6 bytes: EH3zxWz (OK)
6 bytes: F93jxv (OK)
6 bytes: I6mk_a (OK)
6 bytes: J+wTt) (OK)
6 bytes: K+6eo0 (OK)
6 bytes: KGEHkt (OK)
6 bytes: N/jUuJ (OK)
6 bytes: O/+dnS (OK)
6 bytes: O3d8oG (OK)
6 bytes: TX.K94 (OK)
6 bytes: Uy1jKa (OK)
6 bytes: XJju5k (OK)
6 bytes: YJ+D.r (OK)
6 bytes: Zvoklv (OK)
6 bytes: a3H1hL (OK)
6 bytes: dG(pwf (OK)
6 bytes: e7U0i/ (OK)
6 bytes: fgb2/o (OK)
6 bytes: g6AbXj (OK)
6 bytes: kHvqPq (OK)
6 bytes: kT9-Qe (OK)
6 bytes: lQq0zz (OK)
6 bytes: vwW0Z8 (OK)
```

5字节，最终密钥为 **(0_0)**，Rabbit解密得到flag

在线Rabbit算法加密解密工具

U2FsdGVkX18kNy7RIBvcV9WJsqa+oxvdd0lr86U2cU2996N6ltZi7VV0aw==

(0_0)

Rabbit加密 Rabbit解密 清空输入框 复制结果文本

ISCC{u_really_know_rabbits}

https://blog.csdn.net/qq_46150940

擂台

小明的表情包

放假期间小红被亲戚叫去帮店里帮忙，店里忙极了导致小红没有时间写代码。小红苦恼极了，她突然想起来小明有一张非常适合描述她此时心情的表情包。于是，小红让小明把表情包分享给她。小明说如果你记得我的出生的日月年，我就交给你。小明的生日年份隐藏在这串凯撒密码“AVARGRRAAVARGL AVAR”中，你能帮小红得到小明的表情包吗？

编写脚本进行凯撒密码爆破：

```
s = "AVARGRRAAVARGLAVAR"

def kaisa(k):
    t = ""
    for c in s:
        if 'a' <= c <= 'z':
            t += chr(ord('a') + ((ord(c) - ord('a')) + int(k)) % 26)
        elif 'A' <= c <= 'Z':
            t += chr(ord('A') + ((ord(c) - ord('A')) + int(k)) % 26)
        else:
            t += c
    print(t)

for i in range(0, 26):
    kaisa(i)
```

看到有含义的英文：NINETEEN NINETY NINE

KaisaCrack (1) ×

FAFWLWWF FAFWLQ FAFW
GBGXMXHG GBGXMR GBGX

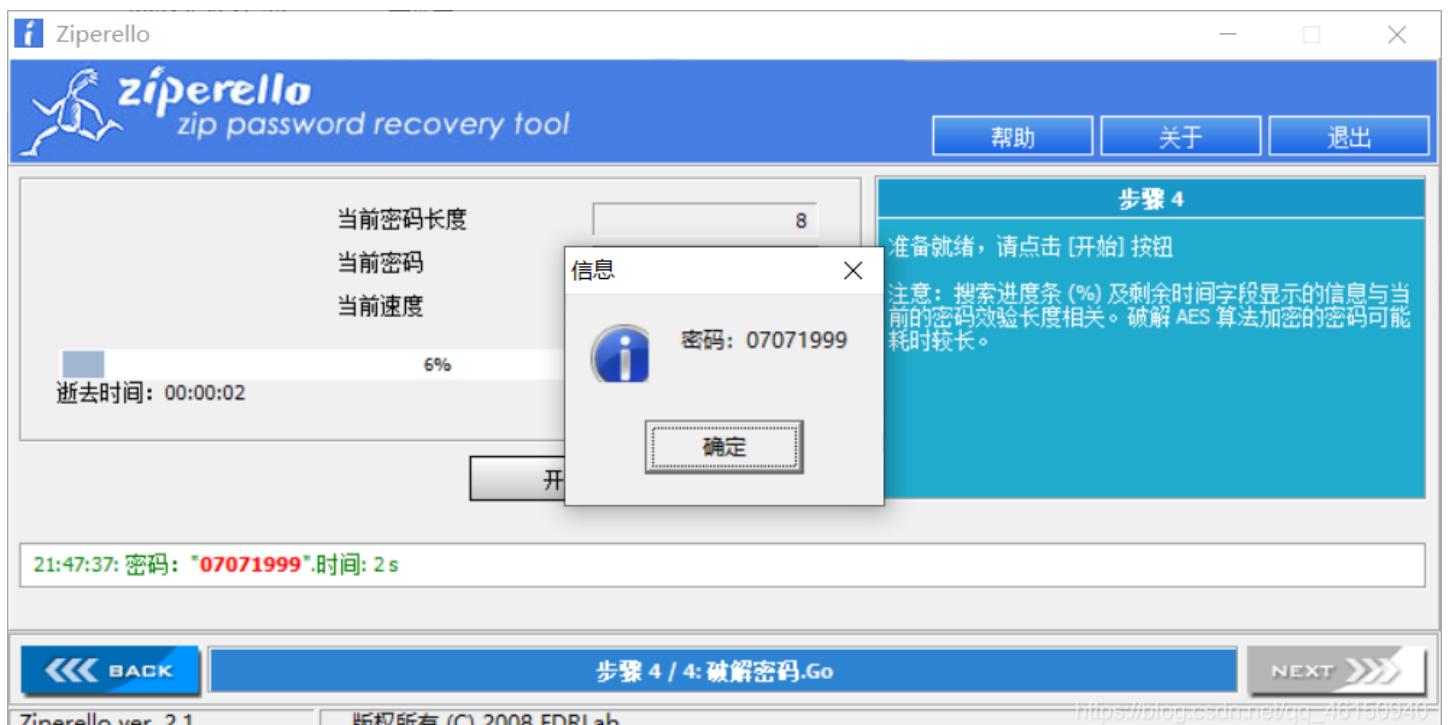
```

HCHNYYYH HCHYNS HCHY
IDIZOZZI IDIZOT IDIZ
JEJAPAAJ JEJAPU JEJA
KFKBQBBK KFKBQV KFKB
LGLCRCCL LGLCRW LGLC
MHMDSDDM MHMDSX MHMD
NINETEEN NINETY NINE ←
OJOFUFFO OJOFUZ OJOF
PKPGVGPP PKPGVA PKPG
QLOUWWUQ QLOUWP QLOU

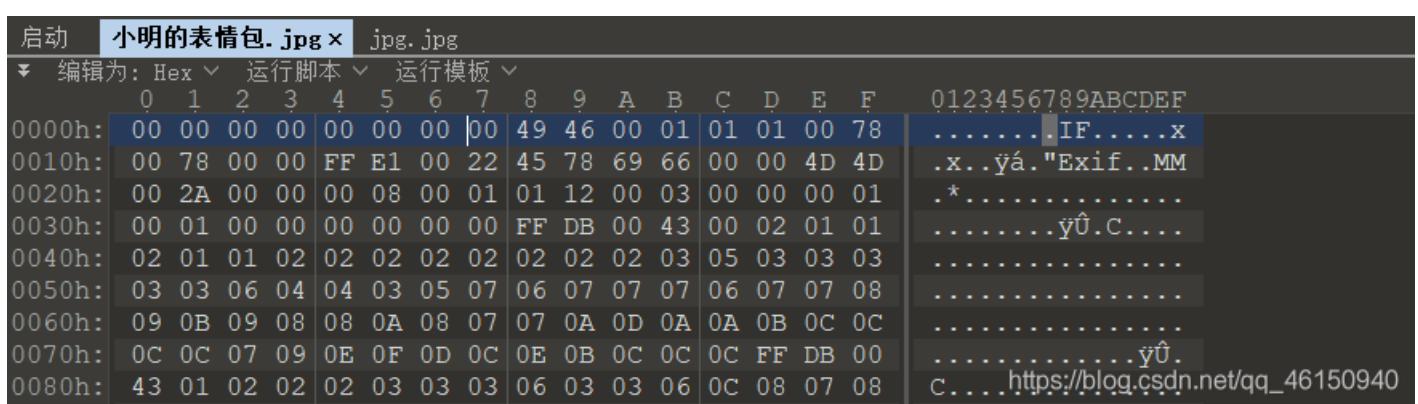
```

https://blog.csdn.net/qq_46150940

猜测是901909，然而并不是压缩包口令，根据题目提示密码是生年份，直接爆破8位数字密码



得到口令07071999，解压压缩包，得到的jpg文件无法打开，拖入010 editor中，与正常的jpg文件比较，发现缺少JPG文件头



将前八位改为FF D8 FF E0 00 10 4A 46，保存得到flag

求求你们放我 回去写代码吧



ISCC{Nyuuuiitt}

https://blog.csdn.net/qq_46150940

Base小偷

被凯撒小猫偷走的等号1/3

Flag格式: flag{XXX}

密文

trefy2k2ov2lig2gqd2eqakoxjqcw4lzttnfli

使用上面的凯撒密码爆破脚本，修改一下脚本，写入crack.txt文件中：

```
s = "trefy2k2ov2lig2gqd2eqakoxjqcw4lzttnfli"
f = open("crack.txt", "w")
def kaisa(k):
    t = ""
    for c in s:
        if 'a' <= c <= 'z':
            t += chr(ord('a') + ((ord(c) - ord('a')) + int(k)) % 26)
        elif 'A' <= c <= 'Z':
            t += chr(ord('A') + ((ord(c) - ord('A')) + int(k)) % 26)
        else:
            t += c
    print(t)
    f.write(t+"\n")

for i in range(0,26):
    kaisa(i)
```

得到crack.txt

```
trefy2k2ov2lig2gqd2eqakoxjqcw4lzttnfli  
usfgz2l2pw2mjh2hre2frblpykrdx4mauogmj  
vtgha2m2qx2nki2isf2gscmqzlsey4nbvphnk  
wuhib2n2ry2olj2jtg2htdnramtfz4ocwqiol  
xviijc2o2sz2pmk2kuh2iueosbnuga4pdxrjpm  
ywjkd2p2ta2qn121vi2jvfptcovhb4qeyskqn  
zxkle2q2ub2rom2mwj2kwgqudpwi4rfzt1ro  
aylmf2r2vc2spn2nxk21xhrveqxd4sgaumsp  
bzmnng2s2wd2tqo2oy12myiswfrke4thbvntq  
canoh2t2xe2urp2pz2nzjtxgszf4uicwour  
dbopi2u2yf2vsq2qan2oakuyhtamg4vjdxpvs  
ecpqj2v2zg2wtr2rbo2pblvziubnh4wkeyqwt  
fdqrk2w2ah2xus2scp2qcmwajvcoi4xlfzrxu  
gersl2x2bi2yvt2tdq2rdnxbkwdpj4ymgasqv  
hfstm2y2cj2znu2uer2seoyclxeqk4zhbtzw  
igtun2z2dk2axv2vfs2tfpzdmmyfrl4aoicuax  
jhuvo2a2el2byw2wgt2ugqaenzgsm4bpjdvby  
kiwp2b2fm2czx2xhu2vhrlbfoahnt4cqkewcz  
ljwxq2c2gn2day2yiv2wiscgpbiuo4dr1fxda  
mkxyr2d2ho2ebz2zjw2xjtdhqcjvp4esmgyebr  
nlyzs2e2ip2fca2akx2ykueirdkwq4ftnhzfc  
omzat2f2jq2gdb2bly2z1vfjselxr4guoiagd  
pnabu2g2kr2hec2cmz2amwgktfmys4hvpjbhe  
qobcv2h21s2ifd2dna2bnxhlugnzt4iwqkcif  
rpcdw2i2mt2jge2eob2coyimvhoau4jxrldjg  
sqdex2j2nu2khf2fpc2dpzjnwigbv4kysmekh
```

猜测是Base32编码，得到密文每行37字符，而Base32按5比特切分的二进制数据必须是40比特的倍数，需要在每行末尾追三个==，并且小写转换为大写

```
# coding: UTF-8  
import base64  
  
ff = open('output.txt','w')  
with open('crack.txt','r') as f:  
    lines = f.readlines()  
    for line in lines:  
        line_n = line.replace('\n','')  
        line_n += '==' #行末尾加上"=="，同时加上"\n"换行符  
        line_new = line_n.upper() #小写转换为大写  
        ff.write(line_new) #写入一个新文件中  
ff.close()  
  
with open('output.txt','r') as f:  
  
    list = f.read().splitlines() #存入列表中  
    i = 0  
    ls2 = [str(i) for i in list] #转换为字符串类型  
    while i < len(list):  
        print(base64.b32decode(ls2[i]))  
        i += 1
```

运行脚本，发现一段base64编码

```
zhuijia x
D:\Code\Pycharm\PyCharm2021\work\venv\Scripts\python.exe D:/Code/Pycharm/PyCharm2021/work/ISCC/Base\偷\zhuijia.py
b'\x9cH\izUt\xb4\x1bF\x80\xf4H\x01N\xba'+qy\x9bJ\xb4'
b'\xa4\x8a\xe9z}\xb4\xc4\x9f6\x894\x85o\xc2\xa2;\xf1\x80\xa3\x8c\xc4'
b'\xac\xccpi\x9a\x85\xf4\xd5#H\x91ti\t\x90\xca\xe4Lq\xa1\xab\xce\xd5'
b'\xb5\x0e\x80\xe9\xba\x8e4\xe5\xa7I\x99\xb4y\x8d\xb1\x03&\\xf1\xc2\xb4\x10\xe5'
b'\xbdP\x91i\xda\x96t\xf6+J\xa1\xf4\x8a\x11\xd2\x0bh\q\xe3\xbcR\xf6'
b'\xc5\x92\xa1\xe9\xfa\x985\x06\xafK\xaa4\x9a\x95\xf3\x13\xAAP\xf2\x04\xc4\x95\x06'
b'\xcd\xd4\xb2j\x1a\xa0u\x173L\xb2t\xab\x1a\x14\x1b\xec\x81%\xcc\xd7\x17'
b"\x06\x16\xc2\xea:\xa8\xb5'\xb7M\xba\xb4\xbb\x9e5$.x91\xf2F\x05\x19"
b'\x0e\xd3jZ\xb0\xf58;N\xc2\xf4\xcc"p\xa2r\g\r[8'
b'\x10\x1a\xe3\xez\xb95H\xbf0\xcb4\xdc\xa6w\xb2\xb2\xf2\x88\x15\x9dH'
b'\x18\\xf4j\x9a\xc1uYCP\x03t\xe0*x98<\xc0\xc3r\x9a9\x1d\xdfY'
b'\x9f\x04\xea\xba\xc9\xb5i\x7Q\x0b\xb4\xf0\xae\xb9E\x02\xd3\xf2\xca&!i'
b'(\xe1\x15j\xda\x01\xf5zKR\x13\xf5\x012\xc0MD\xe4r\xeb.cz'
b'1%\xea\xfa\n5\x8a\xcfS\x1c5\x11\xb6\xe1U\x86\xf4\xf3\x0c0%\x8a'
b'9e6k\x1a\x12u\x9bST$u";\x02]\xc9\x05s-8g\x9b'
b'A\x7F\xeb:\x1a\xb4\x0b\xd7U,\xb52\xbf#\f\x0b\x15\xf0\xe0@\xa8\x0b'
b'I\xe9Wh\x1a"\xf4\x1c[V\xf5C@\\x04nM&p/H\xea\x1c'
b'R+g\xe8:+4,\xdfW=5\$xc4%p\x0f6\xf0PQ,,'
b'ZmhZ3t0cXEudHFxQGpqYn0' ←
```

base64解码得到flag

flag{tqq.tqq@jjb}

真作假时假亦真

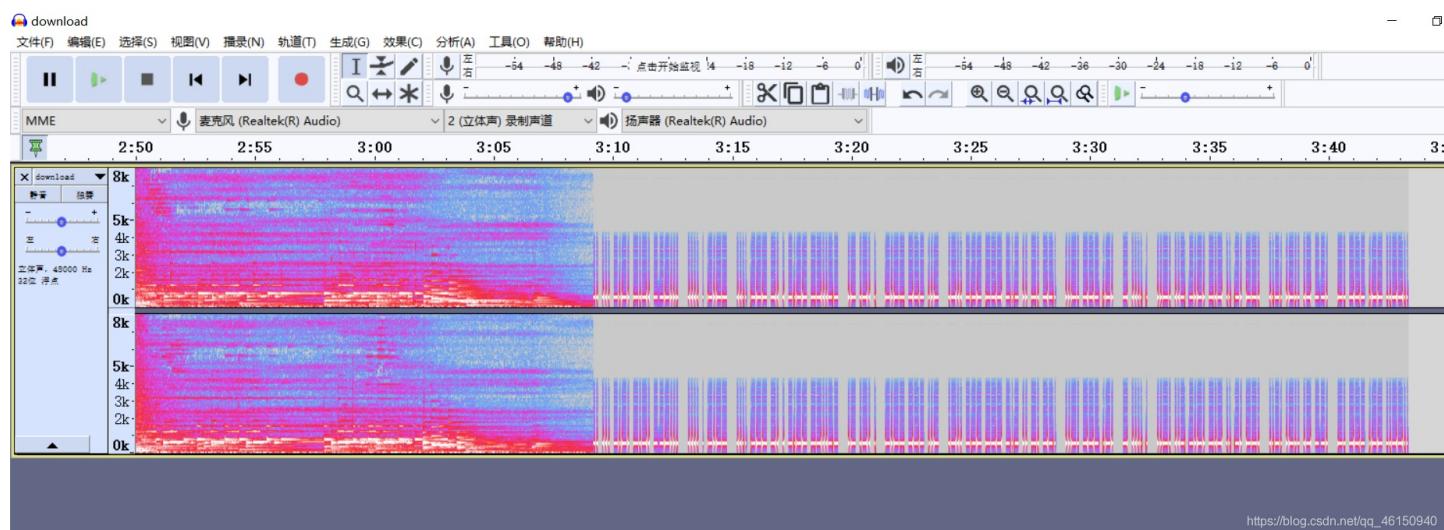
浅韵姐姐第一次参加ISCC，但没有很多CTF工具，为此她很苦恼。

你能切身体会一下吗？

附件下载链接: https://pan.baidu.com/s/1jroyNjtnCwPXxt6A8_0pAw

提取码: a0oh

下载附件，得到download.wav，拖入Audacity查看频谱图



弄下来



摩斯密码解密

ISCCHECHONEYYOULOOKINGFORTHE FUCKING FLAG NOW?

不是flag, foremost分离文件得到一张png图片, exiftool查看图片信息, 得到QQ号

```
Kali㉿Kali: ~/桌面/output/png
└─$ exiftool 00083771.png
ExifTool Version Number      : 12.16
File Name                   : 00083771.png
Directory                   : .
File Size                   : 288 Kib
File Modification Date/Time : 2021:05:08 08:46:57-04:00
File Access Date/Time       : 2021:05:08 08:47:00-04:00
File Inode Change Date/Time: 2021:05:08 08:46:57-04:00
File Permissions            : rw-r--r--
File Type                   : PNG
File Type Extension         : png
MIME Type                   : image/png
Image Width                 : 1242
Image Height                : 1704
Bit Depth                   : 8
Color Type                  : RGB with Alpha
Compression                 : Deflate/Inflate
Filter                      : Adaptive
Interlace                   : Noninterlaced
Pixels Per Unit X          : 2874
Pixels Per Unit Y          : 2874
Pixel Units                 : meters
XMP Toolkit                 : Adobe XMP Core 6.0-c002 79.164488, 2020/07/10-22:06:53
Creator Tool                : Adobe Photoshop 22.0 (Windows)
Create Date                 : 2021:05:03 00:55:27+08:00
Modify Date                 : 2021:05:03 00:59:09+08:00
Metadata Date               : 2021:05:03 00:59:09+08:00
Format                      : image/png
Color Mode                  : RGB
ICC Profile Name           : sRGB IEC61966-2.1
Instance ID                 : xmp.iid:08dbb1cf-06a6-6147-a19e-3c30c111887d
Document ID                 : adobe:docid:photoshop:0cf82e22-bc60-394b-9808-1ec5719e29bc
Original Document ID        : xmp.did:94a58657-d199-9a45-bb13-6a6d55604521
Text Layer Name              : 求④wang諸榮燿排位表 Q(-)5.肆5.5.23)2寸
Text Layer Text              : 求④wang諸榮燿排位表 Q(-)5.肆5.5.23)2寸
History Action               : created, saved
History Instance ID          : xmp.iid:94a58657-d199-9a45-bb13-6a6d55604521, xmp.iid:08dbb1cf-06a6-6147-a19e-3c30c111887d
History When                : 2021:05:03 00:55:27+08:00, 2021:05:03 00:59:09+08:00
https://blog.csdn.net/qq_46150940
```

查看QQ空间, 在留言板发现假flag



加好友问题 ISCC{e@\$Y_socia1_engineering} 也是假flag

空间里唯一的一张图片





https://blog.csdn.net/qq_46150940

我当时做的时候掉坑里去了，直接下载图片，010 editor什么也看不到，思路就断了。

启动 psc.jpg x

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	0123456789ABCDEF
1490h:	F5	A2	37	60	10	19	B3	D4	48	71	E1	17	A9	37	C8	D3	Öç7..³Ôhqá.©7ÈÓ
14A0h:	C4	24	A4	EA	0C	63	37	9B	A7	57	F3	73	1C	56	2D	2D	Àşmê.c7>SWós.V--
14B0h:	4D	0C	58	36	B1	67	D8	F2	A5	BB	CA	F6	AD	72	46	98	M.X6±gØð¥»Èö-rF~
14C0h:	D3	67	A2	C5	CF	EF	32	4A	CA	DE	74	1A	F2	75	D6	50	ÓgcÄi2JÉpt.òuÖP
14D0h:	03	7E	DA	B6	9C	6B	2A	CA	6D	D0	DC	57	67	E3	38	7B	.~Ú¶ek+ÈmÐÜwgä8{
14E0h:	53	2A	CA	73	5C	CE	EE	FF	00	37	DA	8D	D5	A5	5A	69	S*Ès,ïiy.7Ù.Ö¥zi
14F0h:	4B	48	4C	E4	F3	A4	B9	90	E4	B3	5A	69	D9	36	D6	05	KHLäóñ¹.ä³ziÙ6Ö.
1500h:	CD	77	7E	0F	D5	6F	EC	CB	AF	90	D1	39	1D	94	FB	6C	íw~.ÖoiÈ~.Ñ9."ùl
1510h:	67	36	CB	2E	ED	31	56	BD	2D	8D	B3	5D	11	AA	BB	1E	g6È.i1Vñ~-.³].²».
1520h:	67	35	FE	67	C2	12	C3	1A	6F	BB	D2	57	C6	59	6C	B4	g5pgÄ.Ä.o»ÖWEYL'
1530h:	C5	68	BF	F8	7F	A0	0B	98	00	00	88	78	1E	45	36	83	Àhçø. .~..^x.E6f
1540h:	12	B2	95	09	5C	2A	[E7] 08	EB	AA	B9	BD	C5	AD	F3	D2	.².·.*(ç).éë¹¹Å-ôò	
1550h:	D4	A6	D0	E4	82	DD	6E	12	7E	B9	FE	F0	00	1A	F1	52	Ó!Ðä,Yñ.~¹þð..ñR
1560h:	E5	19	F5	92	E9	FF	00	68	C8	75	37	5F	51	75	7D	CF	å.ö'éy.hÈu7_Qu}í
1570h:	CC	60	00	3A	21	6E	20	CF	65	1E	E7	E6	3A	AC	8D	4A	í`.:!n Íe.çæ:~.J
1580h:	DF	70	00	1C	93	8A	24	F0	6F	D0	71	B1	9F	98	00	0E	ßp..“ŠšðoÐq±Ý”..
1590h:	CD	74	3E	83	99	2B	71	4B	F5	8B	E2	00	01	CB	0A	75	ít>j™+qKð<å..É.u
15A0h:	0A	33	69	C3	41	FB	C8	54	EF	93	08	FA	C9	58	00	0A	.3iÄAÙÉTI".úEX..
15B0h:	0A	52	94	EF	19	99	9A	8F	CC	C7	A0	DE	94	F6	C6	B7	.R"i.™š.Íç P"ÖE·
15C0h:	D6	AF	ED	98	00	0B	C6	40	CA	A2	42	64	F9	86	A7	77	Ö"í..Æ@È«Bdùt\$w
15D0h:	FB	7E	62	C4	52	A4	12	B8	92	F2	C9	47	E7	B8	00	0A	ú-bÄR¤. 'ðÉGç,..
15E0h:	06	66	66	66	67	B9	9F	89	8A	E9	7D	E4	B3	CB	E6	28	.ffffg¹Ý%Sé)å³Èæ(
15F0h:	DB	33	DF	87	70	00	1D	63	BC	F3	67	EA	97	C3	B9	8E	Ù3ß+p..c³ogè-Ä¹ž
1600h:	AF	29	5C	47	D4	F7	33	EA	67	E6	00	03	BA	5D	7D	45	¬) \Gð=3ègæ..°])E
1610h:	B7	3D	CD	BF	EA	31	27	3B	0B	C3	85	2B	23	7B	BC	46	-íçé1';.Ä...+#({F
1620h:	4A	FA	80	00	96	90	70	2D	39	81	35	33	E2	E1	B5	0C	Jú€.-.p-9.53ååu.
1630h:	BE	C6	DC	2E	22	13	64	63	3C	49	70	96	E7	D5	47	E6	¾EÙ.".dc<Ip-çÖGæ
1640h:	00	03	90	00	01	FF	D9									ýÙ

https://blog.csdn.net/qq_46150940

用原图地址链接下载图片，能看到一个蓝奏云链接

启动 psc.jpg x

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	0123456789ABCDEF
2F00h:	66	AF	1F	B8	97	13	ED	BC	4B	B7	3A	92	B8	4E	26	BC	f~,-.í³K::',N&¹
2F10h:	B2	AB	47	C9	6E	23	F7	81	E2	36	BC	6A	68	B1	11	3C	²«GEn#÷.å6½jh±.<
2F20h:	84	17	D0	E7	36	49	43	68	8F	1C	B4	86	AA	AA	27	89	..Ðç6ICh.. '†²a¹%‰
2F30h:	DD	E6	C5	57	B4	5A	ED	B4	A9	A6	27	6F	F1	99	C8	CA	ÝæÅW'Zí'©!'oñ™ÈÈ
2F40h:	88	63	A8	27	DE	2B	2F	EB	57	14	DE	ED	0A	C6	85	25	^c' P+/ëW.Pí.E..%
2F50h:	38	36	2E	4C	86	2D	2A	C0	99	10	25	E9	FB	B3	9E	D7	86.L†-*À³%.%éù³ž×
2F60h:	F7	01	03	5F	F8	7B	3F	10	15	32	16	BB	5C	AF	47	F1	÷...ø{?..2..»\^-Gñ
2F70h:	56	8A	A0	15	8A	61	85	93	C1	1B	5D	97	2E	99	31	D3	VŠ .Ša...“Á.]~-.™1ó
2F80h:	21	AA	92	FC	B8	E6	5A	87	7E	A7	1F	D4	C0	D0	38	1B	!²' ü,æZ‡~\$.ÖÄD8.
2F90h:	3F	F4	68	F5	25	57	5A	FC	3A	18	44	A5	CA	B0	ED	E3	?ôhð%WZü:.D¥È°íã
2FA0h:	4F	06	76	17	2E	8E	E8	2C	83	2C	1C	A3	5B	24	0A	47	O.v..Žè,f,.f[§.G
2FB0h:	BC	FE	68	6C	20	3A	4F	F2	C2	9A	F2	B3	FB	22	19	32	¾phl :OðÄšð³û".2
2FC0h:	34	59	A2	B8	0D	8E	68	BD	0B	80	26	10	FF	00	BD	AC	4Yç, .žh¾.È.&.ý.¾-
2FD0h:	6E	C7	A9	C1	29	E4	E4	B2	B5	E6	4D	CB	27	28	23	28	nç@Á)ää²µæMÉ'(#+
2FE0h:	25	4A	5E	61	F2	79	36	E7	8E	A4	48	E1	D1	25	5E	96	%J^aðy6çŽ¤HáÑ%^-
2FF0h:	B1	3B	4C	B0	8A	86	6B	E6	CF	E2	2F	32	F6	1B	4A	62	±;L°Štkaëlä/2ö.Jb
3000h:	30	94	8E	A8	67	40	B9	78	8A	13	32	63	74	21	E6	0B	0"ž`g@¹xS.2ct!æ.
3010h:	D9	9A	1B	66	6B	1F	0C	E9	4B	86	ED	40	D5	73	22	9A	Üš.fk..éKtí@Ös"š
3020h:	69	7F	A2	54	A8	95	7B	C6	CF	F5	7A	04	76	6E	EA	AE	i.cT''•{Eiõz.vnè®
3030h:	32	C6	01	42	5D	E7	F1	2B	F2	31	B5	27	6E	46	76	D7	2Æ.B]çñ+ò1µ'nFvx
3040h:	53	FA	60	9E	60	0F	CB	2F	87	4E	29	81	6E	20	17	EB	Sú ž'.È./+N).n..é
3050h:	E9	A3	1D	51	16	51	84	CB	EB	12	B0	09	FD	F1	61	4B	éf.Q.Q,,Èë..º.yñaK

3060h:	EF 1B 56 A7	3B 88 EF 51	D6 D3 25 21	0B 2A EE 4A	i.V\$;^iQÖÖ%!*iJ
3070h:	3F D6 5C 12	D7 78 D0 BB	E6 A2 E7 15	82 A5 E7 5C	?Ö\.xxĐ»æ¢ç.,¥ç\
3080h:	33 39 8E 70	A0 2A BC 54	B5 B6 DB 79	87 FD D4 27	39žp *‡Tp¶Ùy†vJ'
3090h:	BB 79 94 17	DA 7B 02 BE	7A 7F FF D9	0D 0A 68 74	>y''.Ú{.%z.yÙ..ht
30A0h:	74 70 73 3A	2F 2F 77 77	61 2E 6C 61	6E 7A 6F 75	tps://wwa.lanzou
30B0h:	73 2E 63 6F	6D 2F 69 4E	56 62 73 6F	71 72 7A 38	s.com/iNVbsoqrz8
30C0h:	66 0D 0A 4D	54 56 44 51	77 3D 3D		f..MTVDQw==

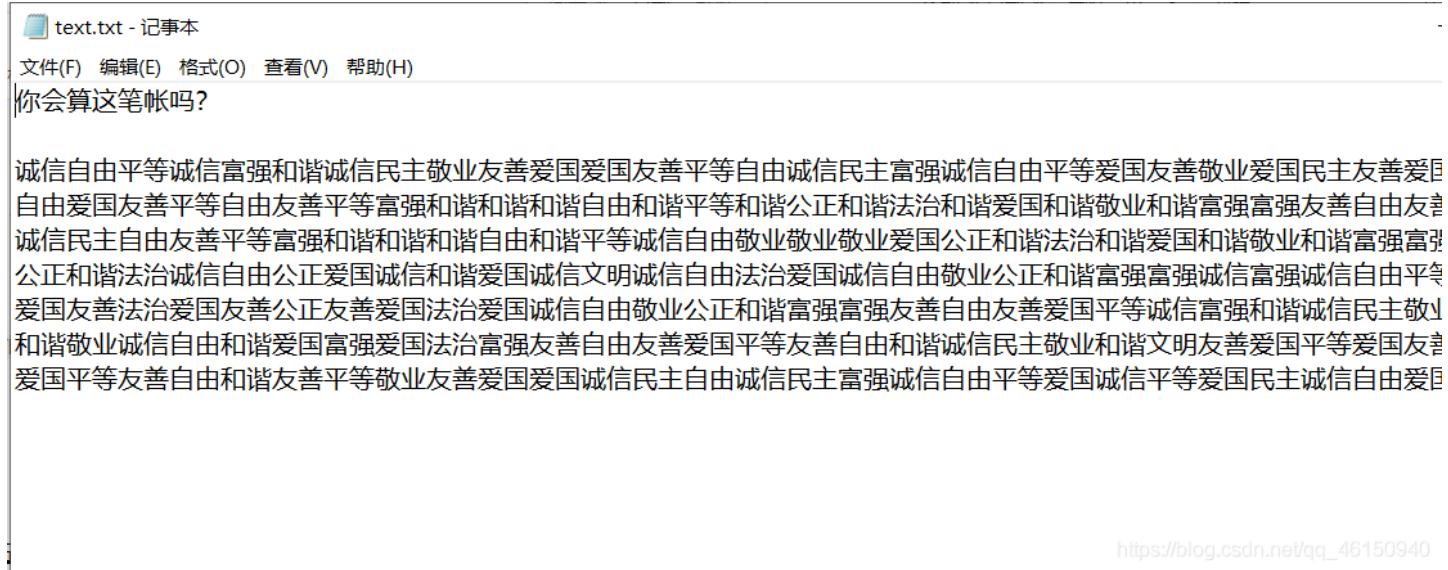
https://blog.csdn.net/qq_46150940

提取出来

<https://wwa.lanzous.com/iNVbsoqrz8f>

MTVDQw==

提取码是 **MTVDQw==** , Base64解码得到 **15CC** , 下载附件



https://blog.csdn.net/qq_46150940

使用在线网站进行社会主义核心价值观解密得到

汉字转换成1，数字换成0。

1111111011101100100110011
1000001010010010110110110
1011101001111101110100010
1011101000110111100000000
1011101010011010111100010
1000001010100101100010110
11111110001111001010100010
0000000011011110100010000
110100110000010011111010
00101000011001111010101010
00111010000111110101101110
1100100011000001000000110
10101111000101001101100000
10010100110010010101101010
10100110010110010000100110
1100000001110110100101110
11001110011011011001011110
00000000010010100000000000
11111110101010101011111110
10000010111010001010000010
10111010000000011010111010
10111010110100011010111010
1000001010000100101010000010
11111110100101110011111110

25x25,之前做过一样的，直接使用脚本转换成二维码

```
import PIL.Image
MAX = 25
pic = PIL.Image.new("RGB", (MAX, MAX))
str = '1111111011101001100111000001010010010110110101110100111101110100010101110100011011110000000010111
010100110101111100011000001010100101100010110111111000111100101010001000000001101111010001000011010011000001001
111110100010100001100111101010100111101000111110101101111001000110000010000011010101111000101001101100001001
010011001001010110101101001011001000010011110000000111011010010111011001101101100101111000000001001010
00000000011111110101010111111100000101110100010100000110111010000000011010111011101010000010101011101101
110101101000110101110100000101000010010100001111111010010111001111111'
i=0
for y in range (0,MAX):
    for x in range (0,MAX):
        if(str[i] == '1'):
            pic.putpixel([x,y],(0,0,0))
        else:
            pic.putpixel([x,y],(255,255,255))
        i = i+1
pic.show()
pic.save("flag.png")
```

扫描二维码得到flag