

## misc

---

### Hide-and-seek

打开文件，露出马脚



---

复制粘贴

moectf{Hey\_U\_ve\_f0und\_m3!}

### Misc杂项之入门指引

#### Rabbit

用010editor打开，拖到文件尾，看到在文件尾有一串类似base64的字符，但是解密之后并不是base64

```
)ùS0M..‰!-^û....  
IEND@B` , #####  
(=^_-=) ##U2FsdG  
VkX1+EP1LmNvaJK4  
Pe06nW0eLquWsUpd  
yv3fjXM2PcDBDK1X  
eKupnnWlFH..ewFE  
GmqpGyC1VdX8
```

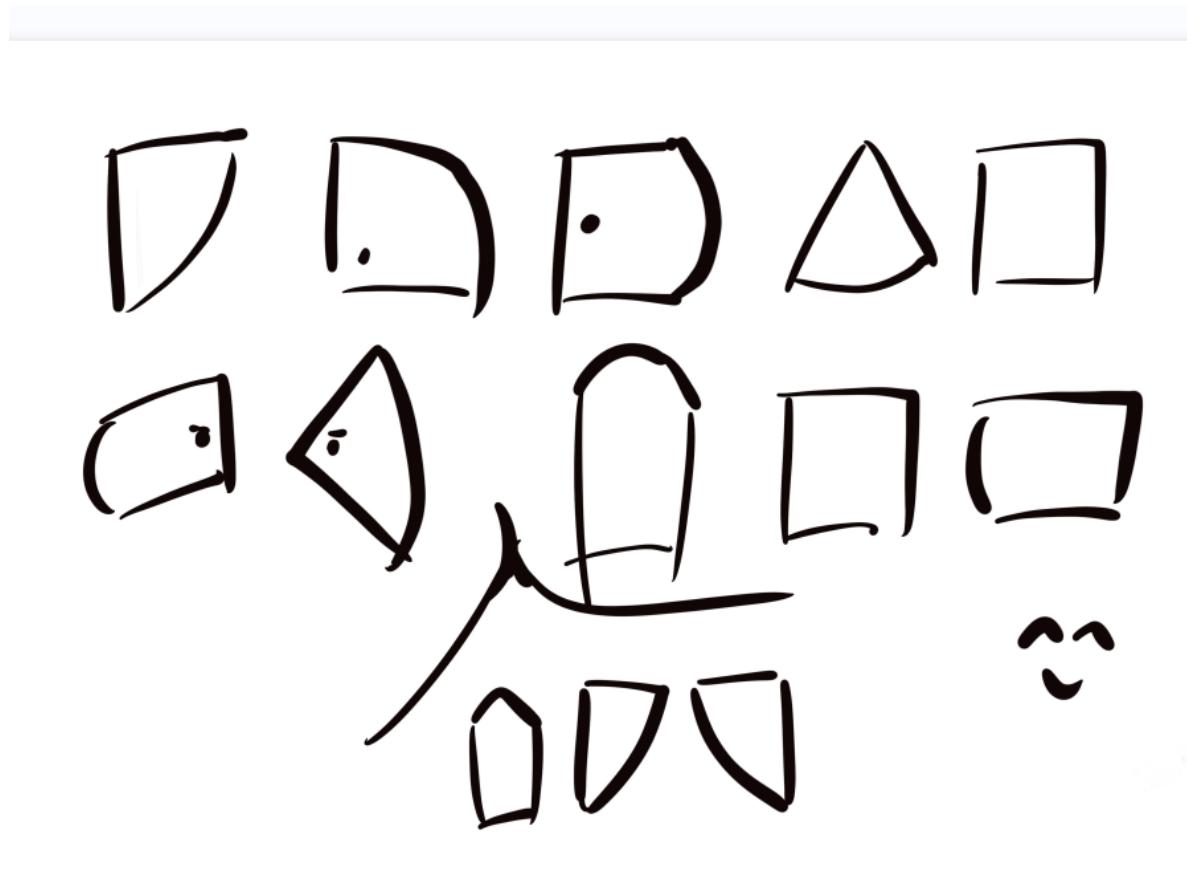
所以是其他加密，尝试性的把题目名称百度一下，发现有个Rabbit加密

加密/解密	AES加密/解密	DES加密/解密	RC4加密/解密	Rabbit加密/解密	TripleDes加密/解密	MD5加解密	Base64加解密	Hash加解密	JS 加密	JS 解密
moectf{We1c0m3_t0_moectf_an7_3n7oY_y0urse1f}					<input type="text" value="在此输入密钥"/> 密码是可选项，也就是可以不填。  <button>&lt; 解密</button> <button>加密 &gt;</button>					
<p>U2FsdGVkX1+EPILmNvaJk4Pe06nW0eLquWsUpdyv3fjXM2PcDBDK1eKupnnWfH ewFEGmpGyC1VdX8</p>										

解密得到flag

小纸条

打开看到图片，一脸懵



所以百度搜索图形加密

在某个网页上滑动的途中看到个 猪圈密码 和这个很像，就尝试解密，得到flag



加密的内容：

© 2024 All Rights Reserved. | [Privacy Policy](#) | [Terms of Service](#)

解密的内容：

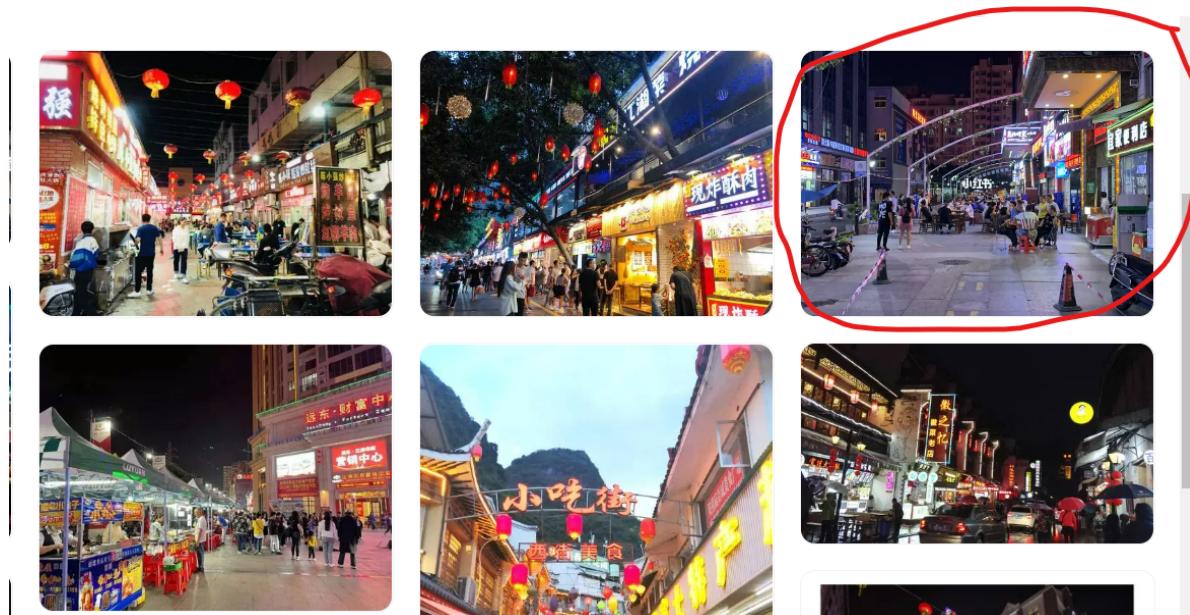
ilovemybigbed

# 寻找黑客的家

打开文件，看到两个图片



这个图片应该是更重要的，国内的地方，先百度图片搜一下



只看到有个小红书的图片，从图片可以进一个大哥的动态，但是动态里面啥都没有，甚至有人评论问大哥位置



翔子



就喜欢这种大排档再配一个喜庆照片

发布于 2020-10-06 21:05



翔子

一起来分享给朋友们看看吧:



### || 笔记评论



小红薯63068972

08-24

回复

请问这是哪里

所以还是老实百度地图找吧，搜最明显的两个店名，汉明宫足疗和宜家便利店，找到街景



位置如下

# 宜家便利店(星光城东南)

到这去

便利店

从这出发

附近

收藏

分享

扫码同步到手机

广东省深圳市龙华区清泉路9号东方会客城2栋

## Locked bass

首先要知道伪加密是什么。

zip伪加密是在文件头的加密标志位做修改，进而再打开文件时识别为加密压缩包。

0000h:	50	4B	03	04	14	00	08	00	08	00	8A	89	EB	54	53	B9	PK.....S%éTS¹
0010h:	7C	FD	D9	00	00	00	FF	00	00	00	11	00	00	00	55	6E	yÙ...y.....Un
0020h:	6C	6F	63	6B	65	64	20	62	61	73	73	2E	7A	69	70	75	locked bass.zipu
0030h:	8E	CF	0B	C1	70	18	C6	BF	F3	EB	E0	A6	24	47	07	47	ŽÍ.Áp.Æóëà!\$G.G
0040h:	69	1B	97	EF	41	4A	22	16	96	B0	39	DA	66	C9	86	03	i.-iAJ".-°9ÚFÉT.
0050h:	45	6E	4A	FE	03	42	29	27	57	94	23	F9	1F	76	E1	E0	EnJþ.B)'W"#ù.váà
0060h:	A2	98	D5	0E	FE	05	DF	6D	6E	F2	BC	3D	7D	DE	7A	DF	¢~Ö.p.ßmnò¾=}þzß
0070h:	A7	1E	9A	B2	3B	DC	C0	90	32	D1	4B	8C	4F	11	18	B4	S.š²;ÜÀ.2ÑKEO..'
0080h:	1B	F6	20	97	DB	72	87	97	EA	42	80	AB	75	BB	E1	DE	.ö -Ürt-êB€«u»áþ
0090h:	A0	A7	9D	17	CF	DD	5E	9D	6F	1F	A7	B5	B6	19	BD	96	§..ÍÝ^..o..Sµ¶.‡-
00A0h:	63	ED	B0	52	67	DB	F7	71	C9	31	50	AE	46	8A	AD	3A	cí°RgÛ÷qÉ1P@FŠ:-
00B0h:	1E	ED	0B	04	C4	6B	C9	D4	90	25	25	9C	25	53	62	55	.í..ÄkÉÔ.%%æ%SBU
00C0h:	4A	0C	79	02	56	58	32	DF	E0	64	48	E4	C8	8A	28	A4	J.y.VX2BàdHäÈŠ(¤
00D0h:	A1	C8	A7	07	8D	02	8B	C7	68	0A	B3	C5	C1	BF	1E	41	;È§...<çh.ºÅÁ¿.A
00E0h:	60	29	60	E1	B7	95	FB	7B	C2	80	1F	78	9A	C7	8B	7D	`)`á..•û{Â€.xšçç }
00F0h:	7A	C5	6E	7A	36	E3	42	14	BD	A1	3B	40	A4	29	A7	CB	zÅnz6äB.‡;@¤)SÈ
0100h:	FC	41	C3	23	4E	CC	C4	07	50	4B	01	02	3F	00	14	00	üAAÃ#NÌÄ.PK..?...
0110h:	09	00	08	00	8A	89	EB	54	53	B9	7C	FD	D9	00	00	00	....S%éTS¹ yÙ...
0120h:	FF	00	00	00	11	00	24	00	00	00	00	00	00	00	20	00	....
0130h:	00	00	00	00	00	00	55	6E	6C	6F	63	6B	65	64	20	62	.....Unlocked b
0140h:	61	73	73	2E	7A	69	70	0A	00	20	00	00	00	00	00	01	....
0150h:	00	18	00	CF	D4	E6	51	06	95	D8	01	CF	D4	E6	51	06	....ÍÔæQ..Ø.ÍÔæQ.
0160h:	95	D8	01	B3	AD	E6	51	06	95	D8	01	50	4B	05	06	00	*Ø.º-æQ..Ø.PK...
0170h:	00	00	00	01	00	01	00	63	00	00	00	08	01	00	00	00	.....c.....
0180h:	00																.

在这个文件中，第一个是压缩源文件数据区的全局加密，第二个是压缩源文件目录区的全局方式标记。

无加密

压缩源文件数据区的全局加密应当为00 00

且压缩源文件目录区的全局方式位标记应当为00 00

假加密

压缩源文件数据区的全局加密应当为00 00

且压缩源文件目录区的全局方式位标记应当为09 00

真加密

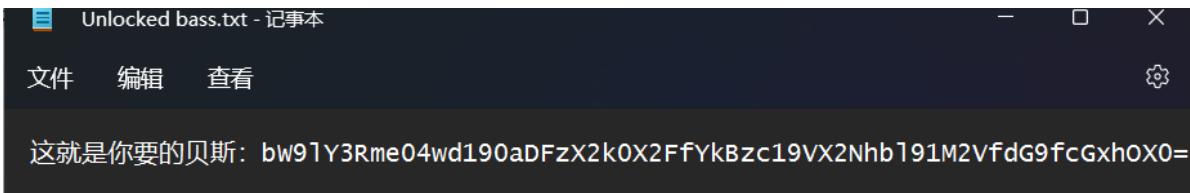
压缩源文件数据区的全局加密应当为09 00

且压缩源文件目录区的全局方式位标记应当为09 00

(09 00 和 00 00 只是表示的一个例子，只要末位是偶数就是无加密，是奇数就是有加密)

因此修改第二个位置为00 00即可

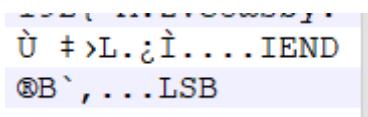
打开文件得到贝斯，解码即可



```
这就是你要的贝斯: bw91Y3Rme04wd190aDFzX2k0x2FfYkBzc19VX2Nhb191M2VfdG9fcGxh0X0=
```

## Nyanyanya!

图片隐写先看看哪里呢，先放010editor里面吧。拖到文件尾，看到提示LSB



对于LSB隐写我并不是很清楚，此时就要借助ctfwiki来了解一些知识。

LSB ↴

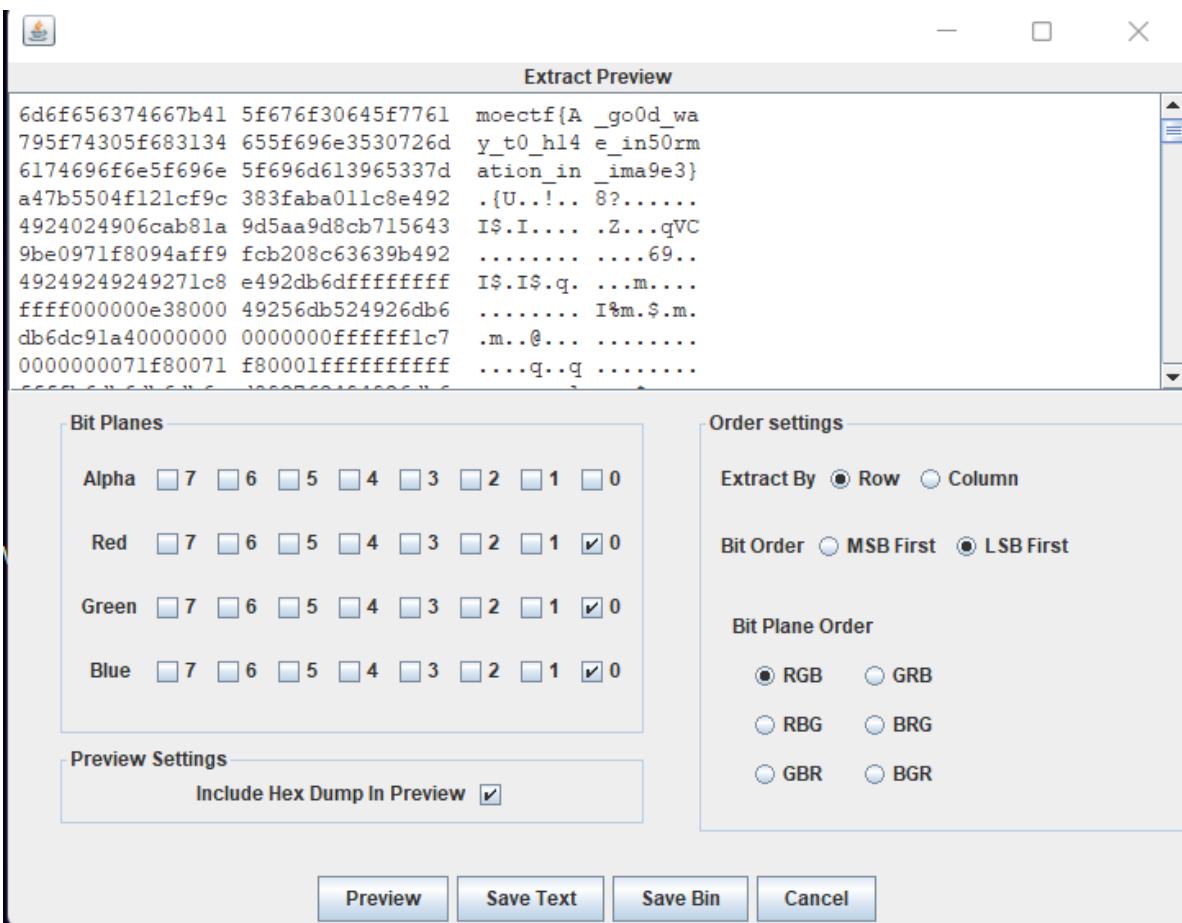
LSB 全称 Least Significant Bit，最低有效位。PNG 文件中的图像像数一般是由 RGB 三原色（红绿蓝）组成，每一种颜色占用 8 位，取值范围为 `0x00` 至 `0xFF`，即有 256 种颜色，一共包含了 256 的 3 次方的颜色，即 16777216 种颜色。

而人类的眼睛可以区分约 1000 万种不同的颜色，意味着人类的眼睛无法区分余下的颜色大约有 6777216 种。

LSB 隐写就是修改 RGB 颜色分量的最低二进制位（LSB），每个颜色会有 8 bit，LSB 隐写就是修改了像数中的最低的 1 bit，而人类的眼睛不会注意到这前后的变化，每个像素可以携带 3 比特的信息。

并且在该介绍下方就有stegsolve的推荐，以及LSB的相关例题。

在本题中，图片的alpha通道中始终没有任何信息，则尝试对另外三个通道的最低位进行提取得到flag。



## What do you recognize

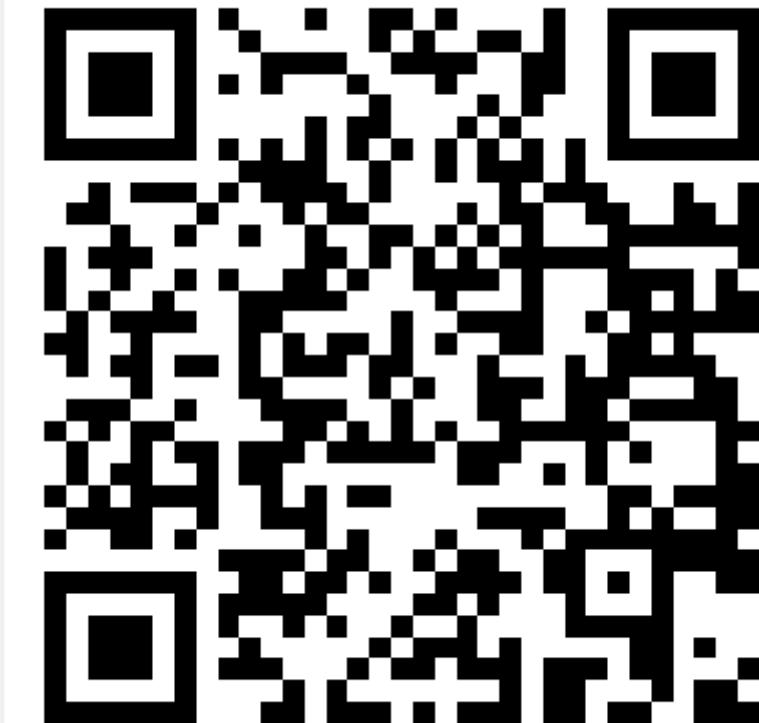
拿到文件发现没有后缀，放入010editor中，文件名中似乎有个‘而导致不能打开，删除’，重新打开

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	0123456789ABCDEF
0000h:	89	50	4E	6C	0D	0A	1A	0A	00	00	00	0D	49	48	44	52	%PN1.....IHDR
0010h:	00	00	01	90	00	00	01	90	08	02	00	00	00	0F	DD	A1	.....Y;
0020h:	9B	00	00	00	09	70	48	59	73	00	00	0E	C4	00	00	0E	>....pHYS...Ä...
0030h:	C4	01	95	2B	0E	1B	00	00	08	A9	49	44	41	54	78	9C	Ä.+....©IDATxœ
0040h:	ED	DD	31	72	E3	48	10	00	41	F2	42	FF	FF	F2	9E	B9	iÝlräH..AòBÿyòž¹
0050h:	9E	B0	11	A3	99	9E	A2	32	7D	91	20	40	56	C0	40	AB	ž°.f™ž¢2} ` @VÀ@«
0060h:	DF	7F	FE	FC	79	01	14	FC	37	7D	00	00	FF	4A	B0	80	ß.büy..ü7}..ýJº€

文件头有点问题，改成PNG的文件头就行了。

	0123456789ABCDEF
2	%PNG.....IHDR
1	.....Y;

然后修改后缀为png，得到flag的二维码



## A band

打开文件，看到一堆二进制数，题目描述让我们一层一层还原，所以先还原成十六进制数。（带懒狗只能找在线工具。）

再把下面这堆十六进制处理(去掉标点, 前缀)后转成字符串

这时可以看到转换后的字符最大都只是f，所以这些还是十六进制字符串，再次转换

字符编码:   十六进制带 \x 前缀  十六进制大写

得到一堆颜文字（有的网站解出来颜文字不太对），想到颜文字加密。

This\_is\_a\_small\_bass\_KRUGS427MJQXG427ONSWK3LT52G6X3CMVPWI2LGMZSXZLOORPWF4TPNPVXI2DFL5YHEZLWNFXXK427N5XGXZXGI3EEND2GRKHQRSBMJSHI5SWIF2WM2LQMJJWVCS2FMVCUJYLHOB4WQ3YNYE2RLDEM4EMWBRY3VIMTSKBG2TBZOYLEE6DTJJVQVTGVGBHEYTEKY4EQWLPIJDVQY3PGJUFKQSMKBGUKZDWPJJTSYLKX43EOWLBKA2XANDDLA3FEMSXKA4GWYY=

戳我加密 ↓

戳我解密 ↑↑

帮助 ??

只有大写字母和数字2-7，是base32，解码得到

KRUGS427MjqXG4270NSWK3LTl52G6X3CMWPi2LGMZSXeZL0ORPW4TPNVPXi2DFL5YheZLWNFXXk427N5XGXZXGi3EEND2GRKHQRSBMjSHi5SWiF2WM2L  
QMjWVCS2FMVCUiYLHOB4WQ3VNvYE2RLDME4EMWBRYN3VIMTSKBBG2TBZ0VLEE6DTJJV0QtVGZBHEYTEKY4EqWLPIJDVQY3PGJUfKQSMKBGUKZDWPFJJTSY  
LXK43EoWLbKA2XANDLAA3EfMSXKA4GWYY=

编码 解码 清空

This\_bass\_seems\_to\_be\_different\_from\_the\_previous\_one\_726B4z4TxFAbdtvVAufipbmQKEeDdagpygCxmpMEca8FX1n7T2rPB  
mL9uVbxjsJSWBu6BrbdV8HYoBGXco2hUBLPMEdyzS9awW6GYaP5p4cX6R2WP8kc

这个经过尝试得到是base58，再解码得到

726B4z4TxFAbdtvVAufipbmQKEeEDagpygCxmpMEca8FX1n7T2rPBmL9uVBxsJSWBu6BrbdV8HYoBGXco2hUBLPMEdvzS9awW6GYaP5p4cX6R2WP8kc

编码Base58>

解码Base58>

转换后：

The\_last\_step\_should\_be\_familiar\_to\_you\_bW9lY3Rme1doeV9zMf9tYW55XzFuc3RydW1lbnRzP30=

最后这个，最熟悉的当然就是base64了，解码得到flag

bW9lY3Rme1doeV9zMf9tYW55XzFuc3RydW1lbnRzP30=

编码Base64>

解码Base64>

转换后：

moectf{Why\_s0\_many\_1nstruments?}

## 想听点啥？

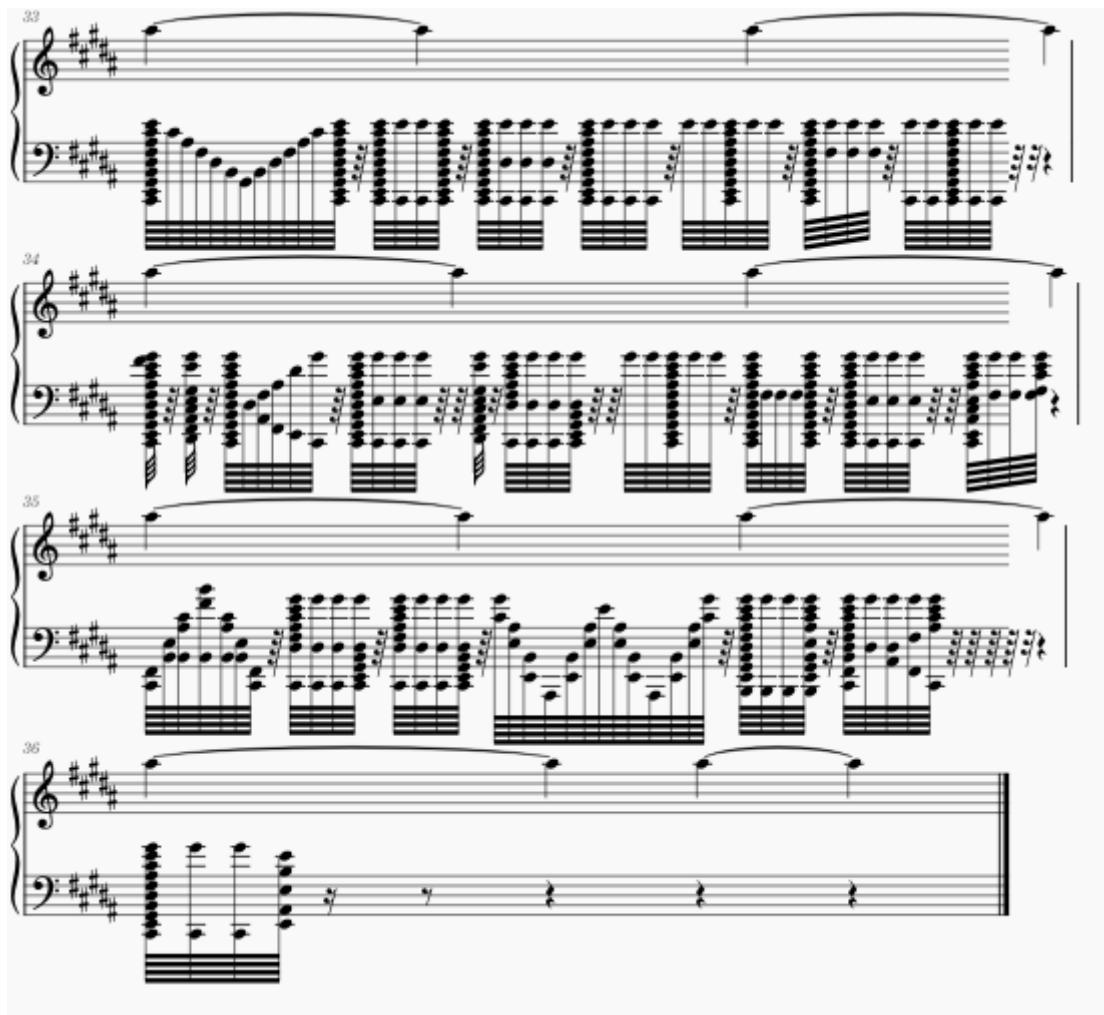
打开文件，一个MP3文件，一个mscz文件，还有一个加密的压缩包。

 whatdoyouliketohear.mscz	32,230	32,394	Compressed MuseSc...	2022/7/24 15:32:34
 whatdoyouliketohear.mp3	934,039	960,887	MP3 文件	2022/7/24 15:34:36
 mister.7z	463	463	7Z 压缩文件	2022/8/18 10:02:18

先听听MP3吧，音乐多美。

然后打开mscz文件，要下载musescore才能查看。

在乐谱中找到了压缩包的密码。



打开压缩包，得到flag.txt和qaq.py。flag是用qaq加密过的。已知密文，算法就是flag的相邻两位异或得到对应位，爆破得到flag。前面加上m

```

f1ag = 'moectf{xxxxxxxxxxxxxxxxxxxxx}'
enc_text = b'\x02\n\x06\x17\x12\x1d,6\x0f\x1a+,C]\x08:2\x02\x1dA12\x18\x06BR\\B'
for i in range(len(f1ag)-1):
    for j in range(30, 130):
        if j ^ enc_text[i] == ord(f1ag[i]):
            print(chr(j), end="")
            flag = list(f1ag)
            flag[i+1] = chr(j)
            flag = ''.join(flag)
            break

```

| oectf {Want\_s0me\_mor3\_mus1c?}

## zip套娃

用ARCHPR根据提示爆破就可以得到对应密码

最后一层是伪加密，改一下标志位就能打开了

## cccccrc

在文档受复杂密码保护，但内部文件极小（只有几个字节）时，可直接用crc碰撞得到明文

随便找个脚本都能解

```
请输入压缩包名字:  
ReadZip >>> C:\Users\王林博\Desktop\hhhh\moe2022\misc\cccccrc.zip  
+-----遍历指定压缩包的CRC值-----+  
[OK] 1.txt: 0x67b2d3df  
[OK] 2.txt: 0x628abed2  
[OK] 3.txt: 0x6b073427  
[OK] 4.txt: 0x8c8da10  
+-----对输出的CRC值进行碰撞-----+  
[Success] 0x67b2d3df: moec  
[Success] 0x628abed2: tf{q  
[Success] 0x6b073427: wq_c  
[Success] 0x8c8da10: rc!}  
+-----CRC碰撞结束! ! ! -----+  
读取成功, 导出CRC列表为: [0x67b2d3df, 0x628abed2, 0x6b073427, 0x8c8da10]  
CRC碰撞成功, 结果为: moectf{qwq_crc!}
```