




第四届“安洵杯”网络安全挑战赛MISC-Writeup

原创

末初  于 2021-11-29 22:54:57 发布  4504  收藏 4

分类专栏: [CTF_MISC_Writeup](#) 文章标签: [2021安洵杯MISC](#)

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/mochu7777777/article/details/121573630>

版权



[CTF_MISC_Writeup](#) 专栏收录该内容

246 篇文章 46 订阅

订阅专栏

文章目录

[应该算是签到](#)

[CyzCC_loves_LOL](#)

[Cthulhu Mythos](#)

[lovmath](#)

题目附件请自取

链接: <https://pan.baidu.com/s/13TwadE6DenseIuRUNZlCKg>

提取码: rrpe

[应该算是签到](#)



B站搜索直接搜索这个BV号



直接页面 **Ctrl+F** 没找出来

搜索引擎找一下有没有通过API查弹幕的方法: <https://www.bilibili.com/read/cv7923601>

至此，你就可以获得视频对应的弹幕XML文件了。



弹幕文件获取 (历史弹幕) (新方法) [当前可用]

前置条件: 通过上述方式获取CID, 已经登录B站

注意: 此方式可能会出现部分无法解析的数据 (例如时间、颜色等)



【炮姐/AMV】我永远都会守护在你的身边!



以这一个视频 (BV1Js411o76u) 为例, 我们通过上面的方式, 已经获得了它的第一p的CID是1176840, 我们需要将以下链接

```
https://api.bilibili.com/x/v2/dm/web/history/seg.so?type=1&oid={cid}&date={date}
```

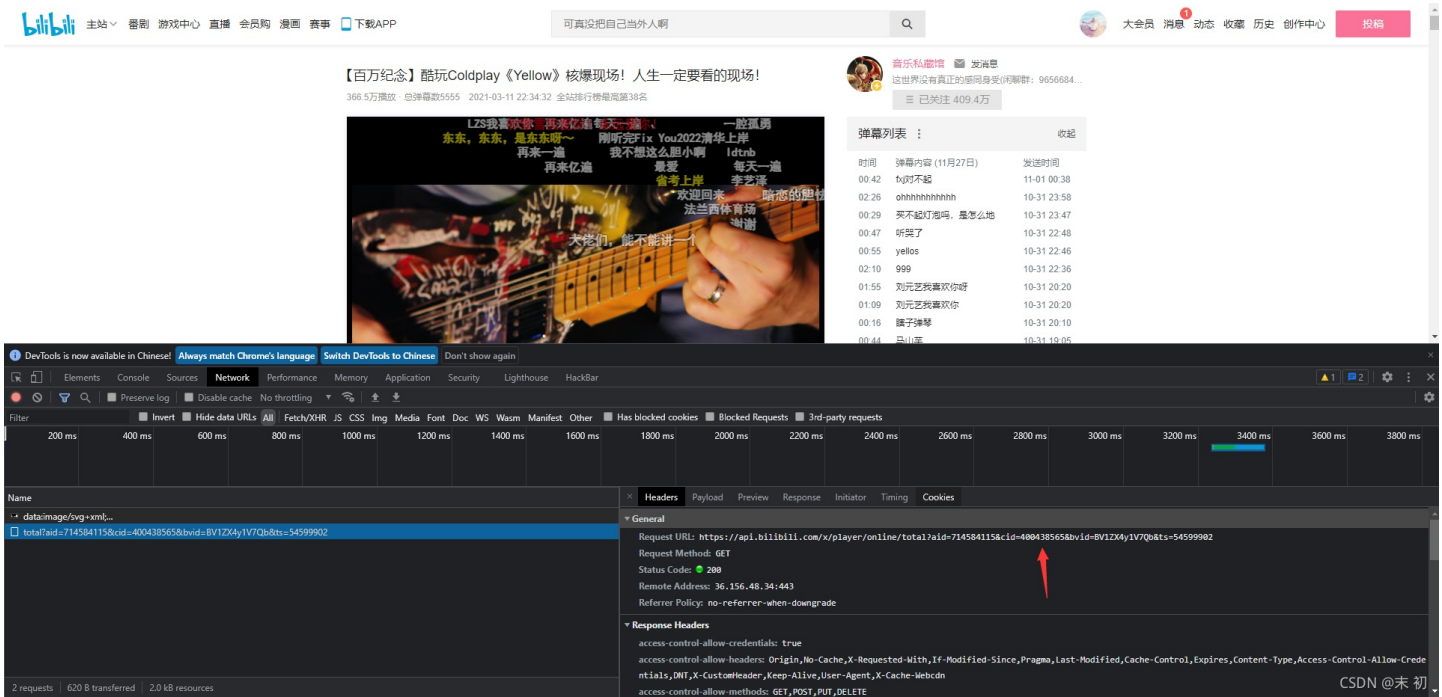
中的{cid}替换为CID, {date}替换为对应的时间, 例如, 我想要查询2013年10月26日的弹幕, 可以访问

```
https://api.bilibili.com/x/v2/dm/web/history/seg.so?type=1&oid=1176840&date=2013-10-
```

这里返回的就是那一天的历史文件了 (如果短时间内查询过多, 可能会触发风控412报错, 建议延长间隔时间, 风控后可能需要等待一段时间后可以再次获取)。通过这些历史弹幕的组合, 你就可以获得全弹幕了~

CSDN @末初

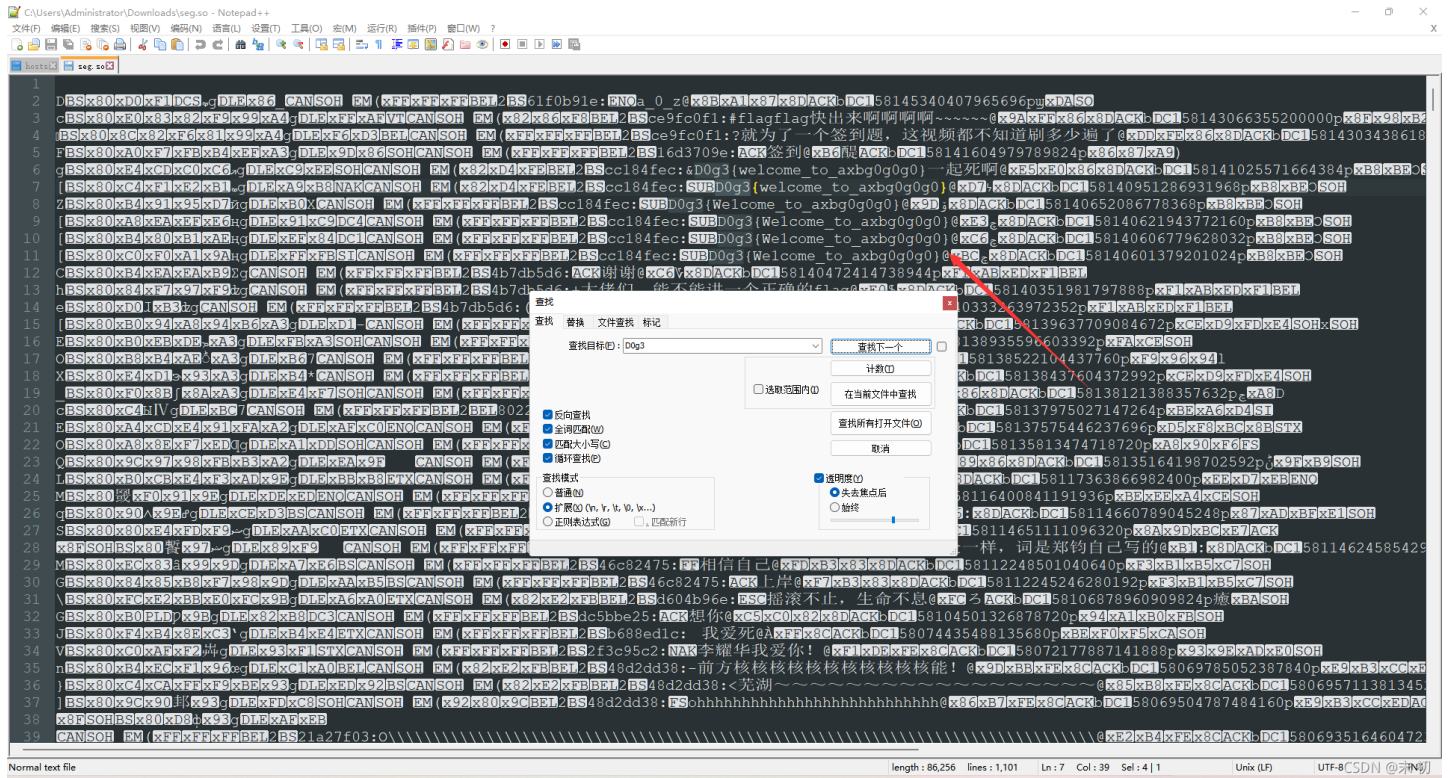
F12点击 Network, 找到这个视频的cid



从当前时间 2021-11-27 开始往前找

```
https://api.bilibili.com/x/v2/dm/web/history/seg.so?type=1&oid=400438565&date=2021-11-27
```

将历史弹幕文件下载下来，选择 UTF-8 编码，然后查找关键字即可



D0g3{We1come_to_axbg0g0}

CyzCC_loves_LOL

Challenge 7 Solves

CyzCC_loves_LOL

373

CyzCC是一个常年0-5的jinx萌妹，你能解出她留给你的题目嘛？PS：密码中的空格请换成下划线 哦 链

接<https://pan.baidu.com/s/18GxTnmcnW6Pe7CCBMv-7A> 提取码:ey96

复制这段内容后打开百度网盘手机App，操作更方便哦

Flag Submit

CSDN@末初

D0g3_LOLteampassword

```
HAI D0g3 code
I HAS A CODE ITZ "D0g3isthepAssword"
I HAS A MSG ITZ ""
I HAS A COUNTER ITZ 0
I HAS A NUM
IM IN YR LOOP UPPIN YR COUNTER WILE COUNTER SMALLR THAN LEN OF CODE
I HAS A C ITZ CODE!COUNTER
NUM R ORD OF C
NUM R SUM OF NUM AN -3
IZ NUM SMALLR THAN 65?, NUM R SUM OF NUM AN 26, KTHX
NUM R CHR OF NUM
MSG R SMOOSH MSG AN NUM
IM OUTTA YR LOOP
VISIBLE MSG
KTHXBYE
```

看不懂什么东西，猜测某种编码，搜索引擎找一下

Google

CTF IM OUTTA YR LOOP VISIBLE MSG KTHXBYE

全部 图片 新闻 视频 购物 更多 工具

找到约 144 条结果 (用时 0.87 秒)

<https://www.dcode.fr/lolcode-language> 翻译此页
LOLCODE Language - Compiler - Online Decoder, Encoder ...
Add LOLCODE Language to **your** mobile apps! ... NUM R SUM OF NUM AN 26, KTHX NUM R CHR OF NUM **MSG R SMOOSH MSG AN NUM IM OUTTA YR LOOP VISIBLE MSG...**

<https://www.dcode.fr/langage-lolcode> 翻译此页
Langage LOLCODE - Decoder, Encoder - dCode.fr
... R CHR OF NUM **MSG R SMOOSH MSG AN NUM IM OUTTA YR LOOP VISIBLE MSG KTHXBYE** ... la déclaration de variable ou les boucles de IM IN YR jusque **IM OUTTA YR** ...

<https://gist.github.com> ... 翻译此页
Hackerrank Project Euler Problem 1 - gists · GitHub
IM IN YR **loop** UPPIN YR INPUT TIL BOTH SAEM INPUT AN CHEEZEBURGER. I HAS A STUFF ... **VISIBLE** I IZ SUMMARIZE YR STUFF MKAY. **IM OUTTA YR loop. KTHXBYE** ...
缺少字词: CTF MSG

<https://www.buzzediter.com/langage-lolcode> 翻译此页
Langage LOLCODE - Interpréteur/Compilateur - Decoder, Encoder
... R CHR OF NUM **MSG R SMOOSH MSG AN NUM IM OUTTA YR LOOP VISIBLE MSG KTHXBYE** ... de capricious ou les boucles de IM IN YR jusque **IM OUTTA YR** (qui pourrait ...

CSDN @末初

- lolcode-language: <https://www.dcode.fr/lolcode-language>

解码得到 ez_misc.zip 密码: AGdJfpqebmXppt1oa



LOLCODE LANGUAGE

Informatics > Programming Language > LOLCODE Language

LOLCODE INTERPRETER

★ SOURCE CODE WRITTEN IN LOLCODE

```
NUM R SUM OF NUM AN -3
IZ NUM SMALLR THAN 65?, NUM R SUM OF NUM AN 26, KTHX
NUM R CHR OF NUM
MSG R SMOOSH MSG AN NUM
IM OUTTA YR LOOP
VISIBLE MSG
KTHXBYE
```

★ DISPLAY THE STANDARD OUTPUT (STDOUT) OF THE PROGRAM
 A JAVASCRIPT VERSION OF THE LOLCODE

See also: [Brainfuck](#)

Answers to Questions (FAQ)

Whate is LOLCODE? (Definition)

Lolcode is a programming language, it is not an encryption, it follows a syntax similar to other programming languages but is very verbose and misuses keywords that are usually replaced by symbols in programming.

Example: I HAS A VAR ITZ 0 corresponds to `VAR = 0`

How to write/encrvpt using LOLCODE?

Search for a tool

★ SEARCH A TOOL ON DCODE BY KEYWORDS:

★ BROWSE THE [FULL DCODE TOOLS' LIST](#)

Results

AGdJfpqebmXppt1oa 

LOLCODE Language - [dCode](#)

Tag(s) : Programming Language

Share



dCode and more

dCode is free and its tools are a valuable help in games, maths, geocaching, puzzles and problems to solve every day!
A suggestion ? a feedback ? a bug ? an idea ? [Write to dCode!](#)



选择语言
由 Google 翻译强力驱动

Summary

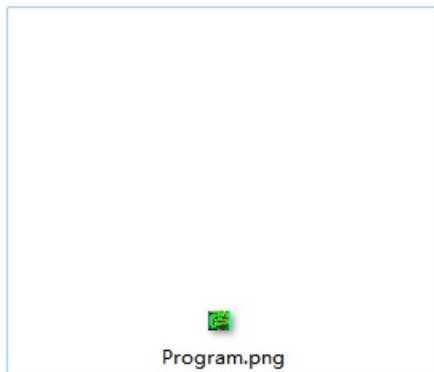
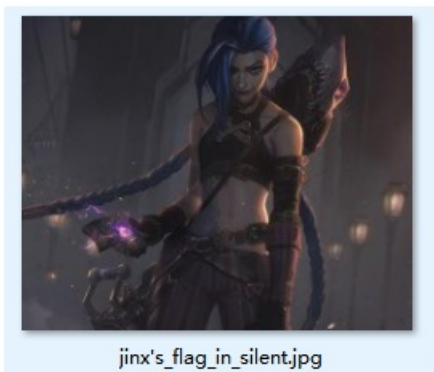
- ★ [LOLCODE Interpreter](#)
- ★ [Whate is Source code written in LOLCODE? \(Definition\)](#)
- ★ [How to write/encrypt using Source code written in LOLCODE?](#)
- ★ [How to interpret/translate/decrypt lolcode?](#)
- ★ [How to recognize lolcode?](#)

Similar pages

- ★ [Brainfuck](#)
- ★ [Javascript Keycodes](#)
- ★ [Spoon](#)
- ★ [Binaryfuck](#)
- ★ [Pikalang](#)

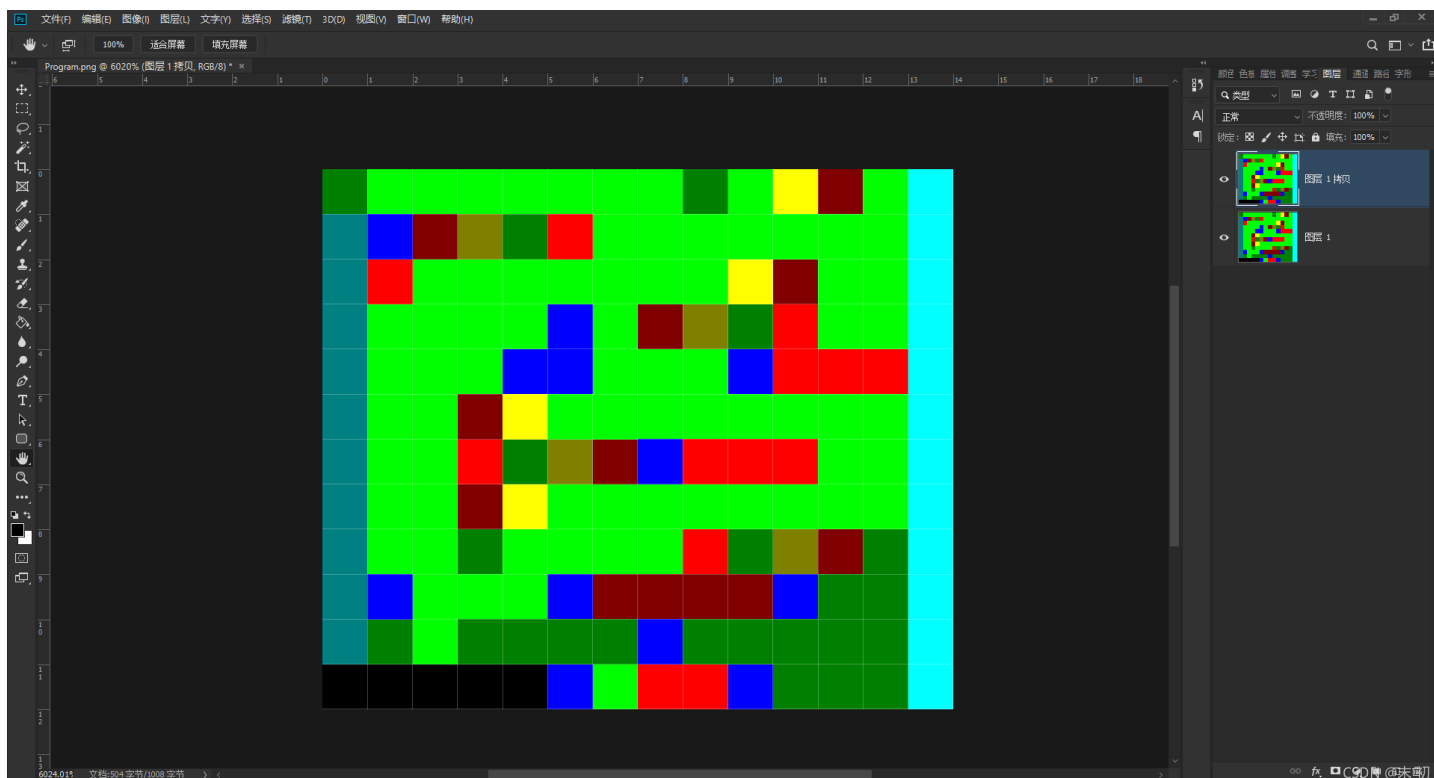
CSDN @末初

> 本地磁盘 (C:) > 用户 > Administrator > 下载 > CyzCC_loves_LOL > ez_misc



CSDN @末初

`Program.png` 根据名称提示一开始以为是 `npiet`，尝试直接编译发现不对

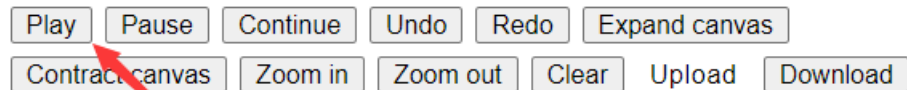


后来经过查阅资料才发现 `Brainfuck` 也有一种用像素颜色表示的语言：`Brainloller`

- <https://minond.xyz/brainloller/>

上传之后点击 `Play`，得到密码：`0MTTW CWZVN!`

Program controls



Brainloller commands



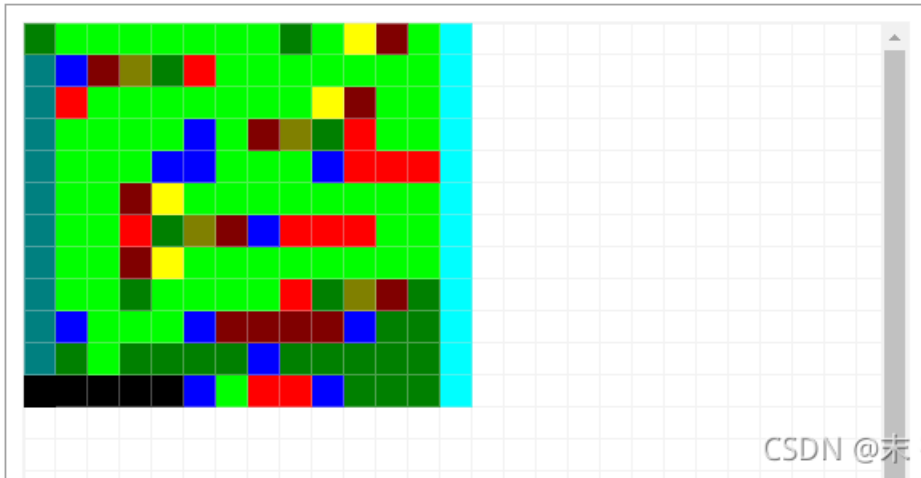
Program memory

0	33	0	67	0	0	0	0	0	0
---	----	---	----	---	---	---	---	---	---

Input

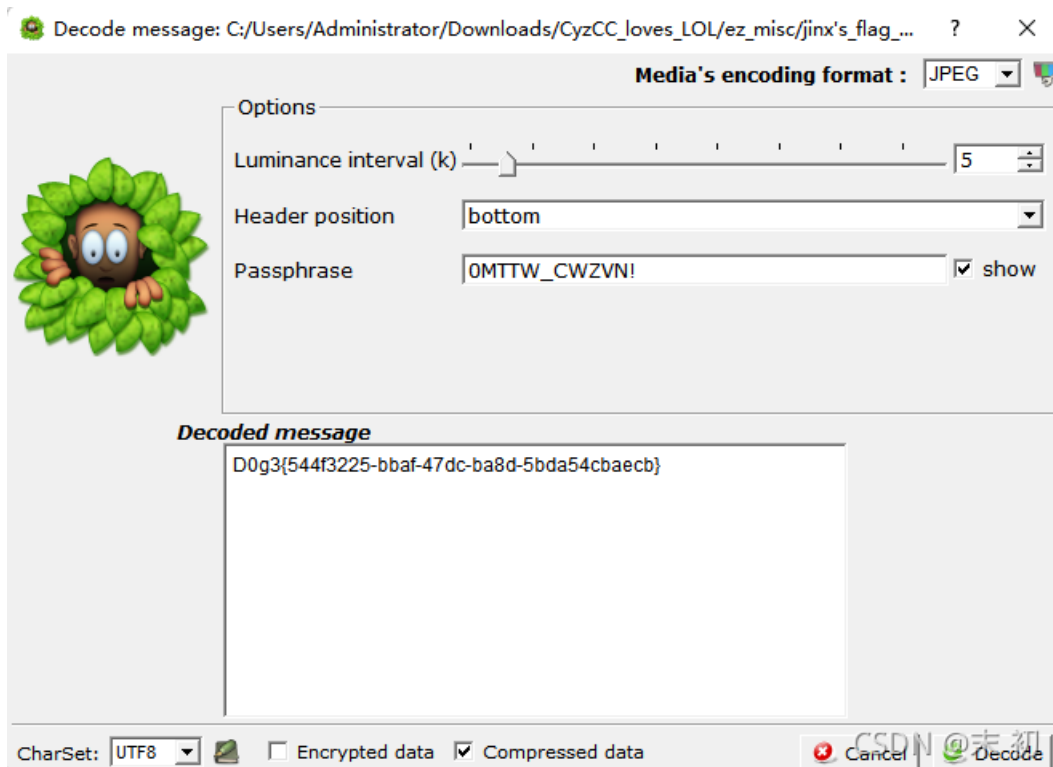
Output

0MTTW CWZVN!



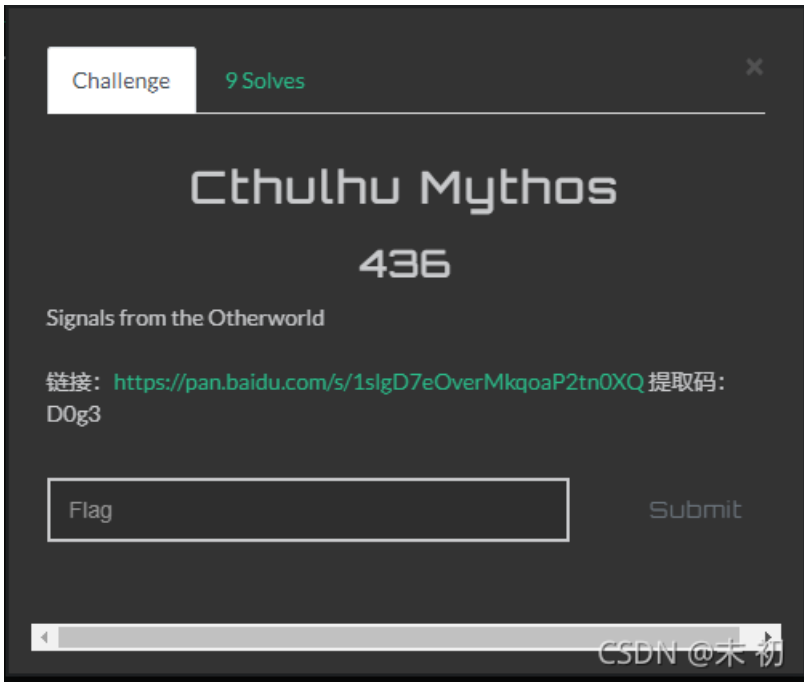
CSDN @末初

然后根据题目名称提示将密码中的空格换成下划线、以及 `jinx's_flag_in_silent.jpg` 的名称，直接尝试 `SilentEye` 解密

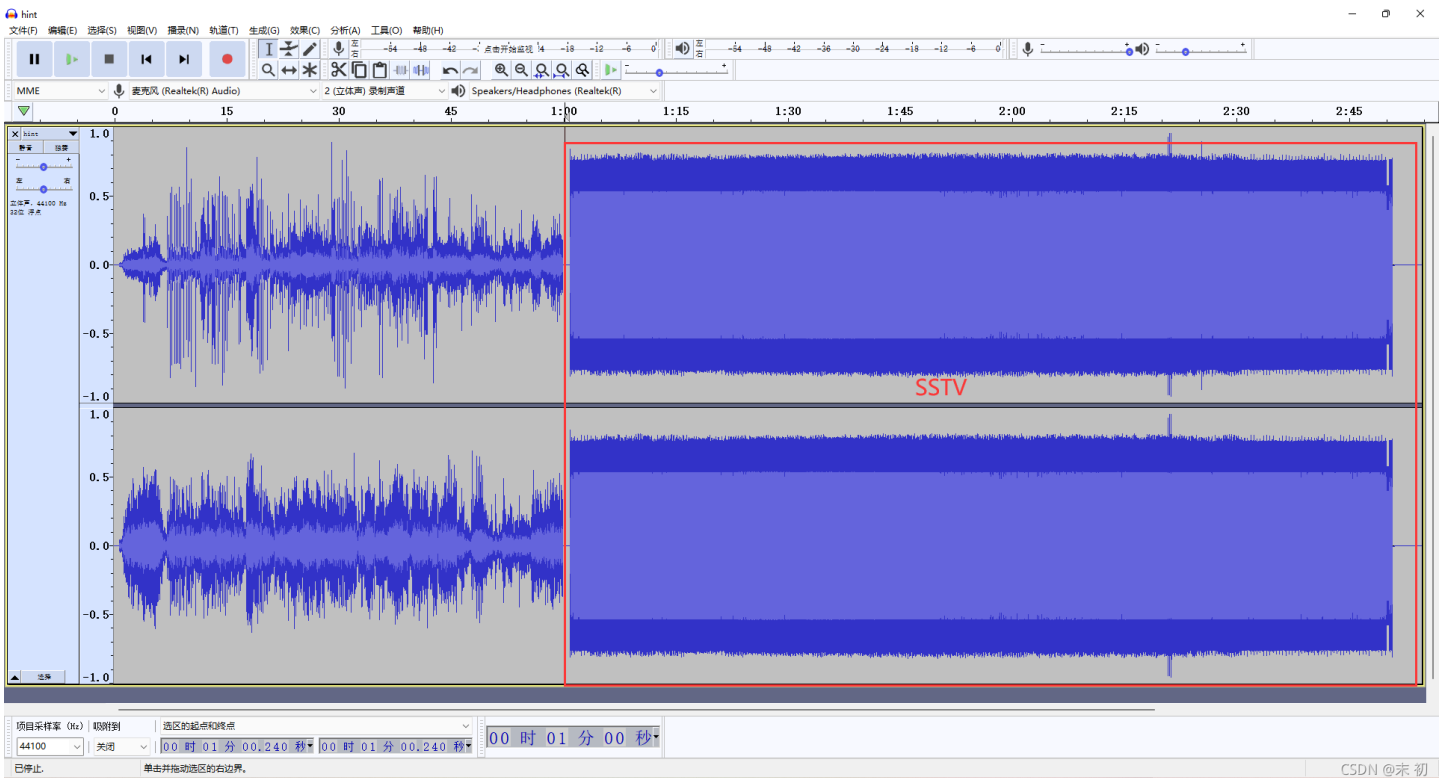


D0g3{544f3225-bbaf-47dc-ba8d-5bda54cbaecb}

Cthulhu Mythos



hint.mp3 听一下，发现前面是泰拉瑞亚的主题曲，后面部分很明显是SSTV



因为格式问题没法直接用 QSSTV，RX-SSTV 的话又比较麻烦要调整电脑录音设备，就直接用 Robot36 听吧

网上随便找个地址：<https://apkpure.com/cn/robot36-sstv-image-decoder/xdsopl.robot36>

下载好传到手机上，安装，然后听就完事了，多听几遍确认信息

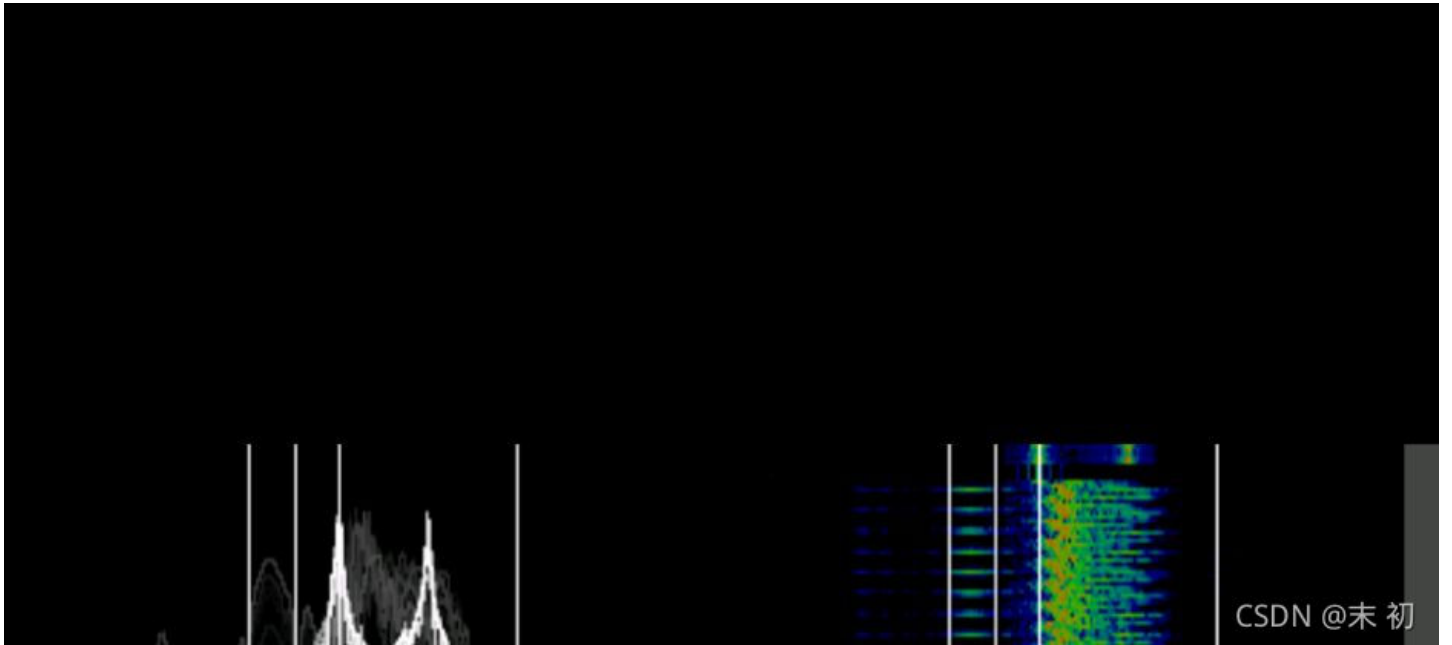




MRPVI4TZLSKKGK4TSGRZGSSYJB

pub

Eurotic Legend



CSDN @末初

```
>>> from base64 import *  
>>> b32decode('MRPVI4TZL5K GK4TSGRZGSYJBPU====')  
b'd_Try_Terr4ria!}'
```

.WLD 文件扩展名

▶ 3个文件类型 使用 .wld 文件扩展名。

档案类型1

Terraria世界档案



开发人员 重新逻辑
声望 ★★★★★ 4.3 (47 投票)
类别 游戏档案
格式 二进制

什么是WLD文件?

Terraria使用的游戏文件，这是2D动作冒险和沙盒构建游戏；保存一个包含地形，水，怪物，物品和其他物体的世界；用于单人或多人游戏，可以由Terraria专用服务器加载，该服务器允许玩家通过网络托管Terraria世界。

更多信息

世界文件可以是预先安装的游戏地图，也可以是用户创建的自定义地图。WLD文件保存到Windows中的以下目录：`[用户] \ Documents \ My Games \ Terraria \ Worlds \`

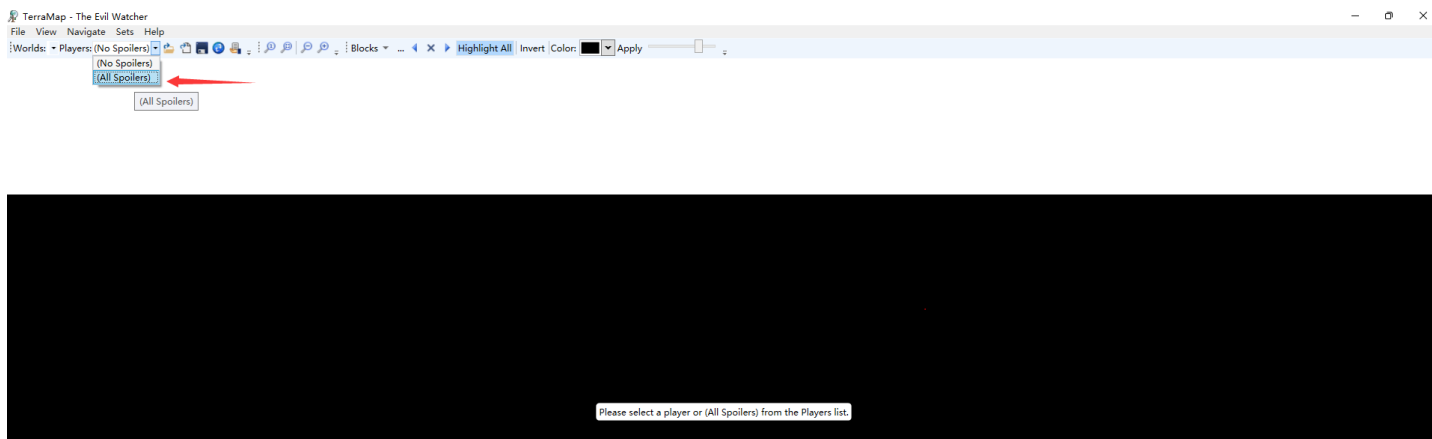
可以使用Terraria地图编辑器（TEdit）编辑WLD文件。可以使用MoreTerra（Terraria World Viewer）在没有Terraria游戏的情况下查看它们。

CSDN @末初

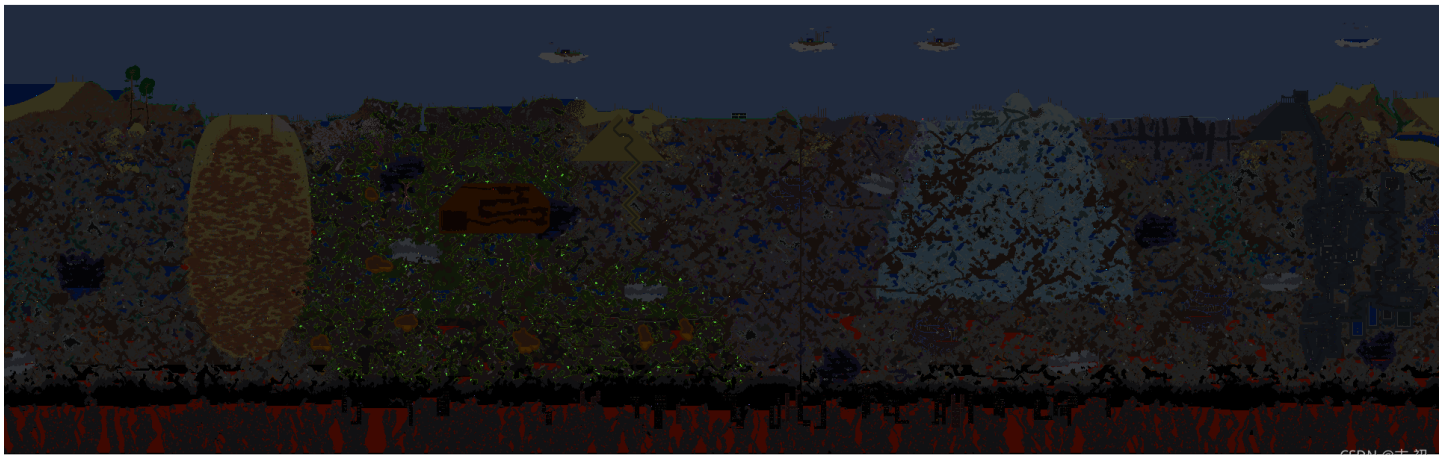
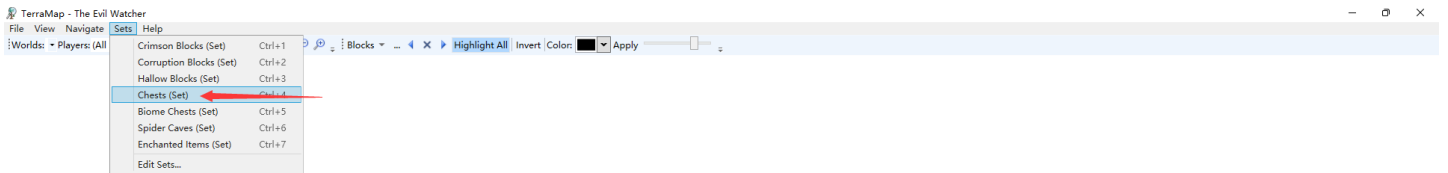
关于泰拉瑞亚地图编辑器

- <https://www.bilibili.com/read/cv275739?from=search>
- <https://www.binaryconstruct.com/downloads/>
- <https://www.bilibili.com/video/BV1Za4y1a7uN>
- <https://m33.wiki/extension/wld.html>

使用 TerraMap 打开 The Evil Watcher.wld ; Players->All Spoilers

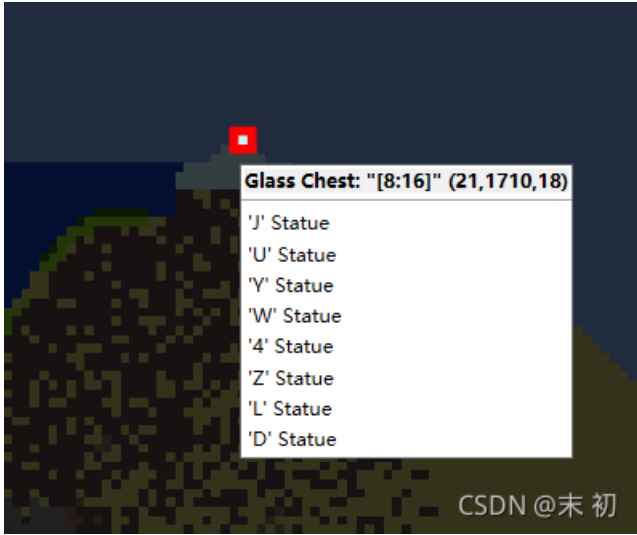


Sets->Chests 找宝箱



在地面上找到四个 `Class Chest`

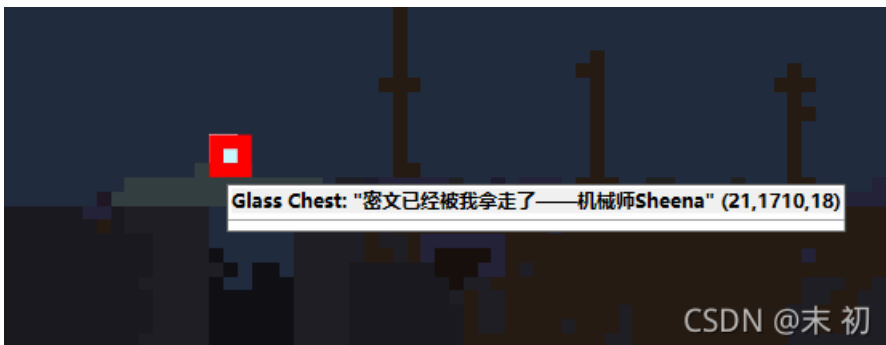




JUYW4ZLD



KI2GM5C



总共四部分，前三部分为: IQYGOM33JUYW4ZLDKI2GM5C

Base32

IQYGOM33JUYW4ZLDKI2GM5C

编码 解码 清空

D0g3{M1necR4ft@

CSDN @末初

第四部分可以使用 TEdit 打开 The Evil Watcher.wld



在这个位置找到第四部分



7I4YF6QLO

最终得到的base32为: IQYGOM33JUYW4ZLDKI2GM5C7I4YF6QLO

IQYGOM33JUYW4ZLDKI2GM5C7I4YF6QLO

编码

解码

清空

D0g3{M1necR4ft_G0_An

CSDN @末初

最终flag

D0g3{M1necR4ft_G0_And_Try_Terr4ria!}

lovemath



The screenshot shows a challenge window titled "lovemath" with a difficulty level of "496". The text "数字你是如此美丽,甚至能画出自己。" is displayed. Below this, there is a "View Hint" button, a download icon with the text "lovemath...", a "Flag" input field, and a "Submit" button. The interface is dark-themed. A watermark "CSDN @末初" is visible at the bottom right of the screenshot.

hint: not blindwater but you can search it

crc32爆破

lovemath.zip - Bandizip (Standard)

文件(F) 编辑(E) 查找(I) 选项(O) 视图(V) 工具(T) 帮助(H)

打开 解压 新建 添加 删除 测试 扫描 查看 代码页

名称	压缩后大小	原始大小	类型	循环冗余检验 (CRC)	修改日期
flag_05.txt*	18	6	TXT 文件	d9e12803	2021/11/22 23:23:23
flag_04.txt*	18	6	TXT 文件	2d2c423c	2021/11/22 22:25:05
flag_03.txt*	18	6	TXT 文件	2a75b14e	2021/11/22 22:24:52
flag_02.txt*	18	6	TXT 文件	f81abecd	2021/11/22 19:26:04
flag_01.txt*	18	6	TXT 文件	a430239a	2021/11/22 19:25:49
flag.zip*	516,577	516,565	ZIP 压缩文件	c38199da	2021/11/22 23:23:49

CSDN @末初

```
-----Filename CRC Info-----
```

```
[+] flag.zip: 0xc38199da  
[+] flag_01.txt: 0xa430239a  
[+] flag_02.txt: 0xf81abecd  
[+] flag_03.txt: 0x2a75b14e  
[+] flag_04.txt: 0x2d2c423c  
[+] flag_05.txt: 0xd9e12803  
-----
```

```
PS D:\Tools\Misc\crc32> python .\crc32.py reverse 0xa430239a
```

```
4 bytes: {0x56, 0x34, 0xbc, 0x00}  
verification checksum: 0xa430239a (OK)  
alternative: 3RAsk0 (OK)  
alternative: 5jYl3N (OK)  
alternative: ANz4c9 (OK)  
alternative: DViXxW (OK)  
alternative: EJg5bZ (OK)  
alternative: JE94EM (OK)  
alternative: JYvhDY (OK)  
alternative: O1YuZg (OK)  
alternative: R3ix8v (OK)  
alternative: _lAJB8 (OK)  
alternative: dYZaCR (OK)  
alternative: mOBoUw (OK)  
alternative: pMrb7f (OK)  
alternative: qImCE3 (OK)  
alternative: sq6Mub (OK)  
alternative: th1s_I (OK)  
alternative: uhpBDP (OK)
```

```
PS D:\Tools\Misc\crc32> python .\crc32.py reverse 0xf81abecd
```

```
4 bytes: {0xf6, 0x44, 0x6a, 0xcc}  
verification checksum: 0xf81abecd (OK)  
alternative: 5XyM2J (OK)  
alternative: 9kccWY (OK)  
alternative: DdIyyS (OK)  
alternative: MrQwov (OK)  
alternative: ONTi6k (OK)  
alternative: RLddTz (OK)  
alternative: s_Y0ur (OK)  
alternative: uZPcET (OK)
```

```
PS D:\Tools\Misc\crc32> python .\crc32.py reverse 0x2a75b14e
```

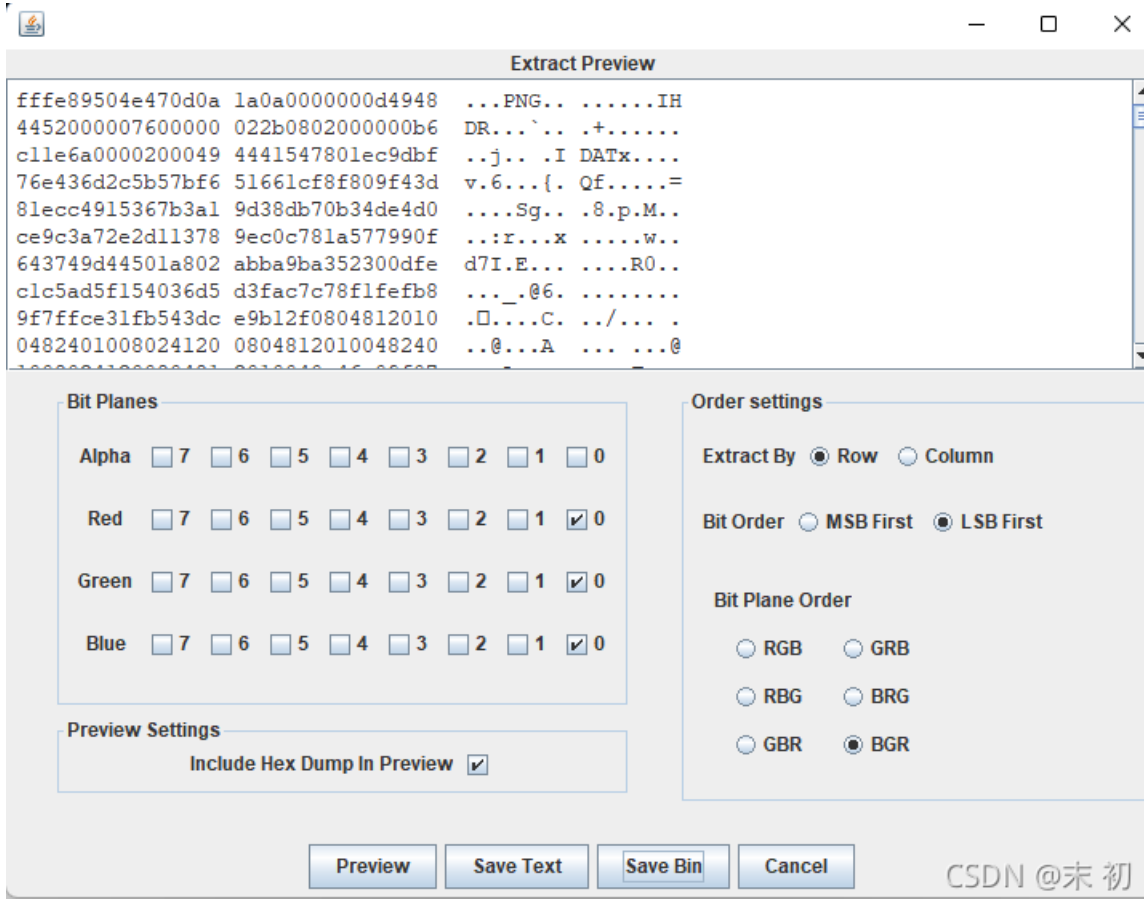
```
4 bytes: {0x39, 0x11, 0xcc, 0x5b}  
verification checksum: 0x2a75b14e (OK)
```

```
alternative: 0njyYo (OK)
alternative: 4Wf40T (OK)
alternative: 7wmGsD (OK)
alternative: 8x3FtS (OK)
alternative: 9xrwOJ (OK)
alternative: Cn_SKk (OK)
alternative: KdI0GC (OK)
alternative: R_upLi (OK)
alternative: S3G1S4 (OK)
alternative: S_4AWp (OK)
alternative: UFrNfB (OK)
alternative: W7ZmRW (OK)
alternative: _pa33w (OK)
alternative: caljpn (OK)
alternative: d5Fi7M (OK)
alternative: dxkTZE (OK)
alternative: jwtdfK (OK)
alternative: ln2kWy (OK)
alternative: rPFIwL (OK)
alternative: w8iTIR (OK)
alternative: x77UNE (OK)
PS D:\Tools\Misc\crc32> python .\crc32.py reverse 0x2d2c423c
4 bytes: {0x78, 0x6b, 0xc3, 0x45}
verification checksum: 0x2d2c423c (OK)
alternative: 0rd_We (OK)
alternative: 1nj2Mh (OK)
alternative: BSNT74 (OK)
alternative: CrQuEa (OK)
alternative: DkVKoJ (OK)
alternative: FWSU6W (OK)
alternative: P2Suvv (OK)
alternative: Sbdw06 (OK)
alternative: Wfyv1U (OK)
alternative: ZIm5NK (OK)
alternative: dderTO (OK)
alternative: jkzBhA (OK)
alternative: rLHoyf (OK)
alternative: uUOQSM (OK)
PS D:\Tools\Misc\crc32> python .\crc32.py reverse 0xd9e12803
4 bytes: {0x9f, 0x48, 0x0c, 0x36}
verification checksum: 0xd9e12803 (OK)
alternative: 1c0m3e (OK)
alternative: 2_tBqa (OK)
alternative: 3_5sjx (OK)
alternative: 98DoSQ (OK)
alternative: A_Ahce (OK)
alternative: FFFVIN (OK)
alternative: J8qEAU (OK)
alternative: K80tZL (OK)
alternative: NLP5Ef (OK)
alternative: PnkKdg (OK)
alternative: Rs0ET6 (OK)
alternative: WwluNL (OK)
alternative: YxsErB (OK)
alternative: ZD7j0F (OK)
alternative: ZXx61R (OK)
alternative: _a5JcP (OK)
alternative: fIuorK (OK)
alternative: hFj_NE (OK)
alternative: izd2TH (OK)
```

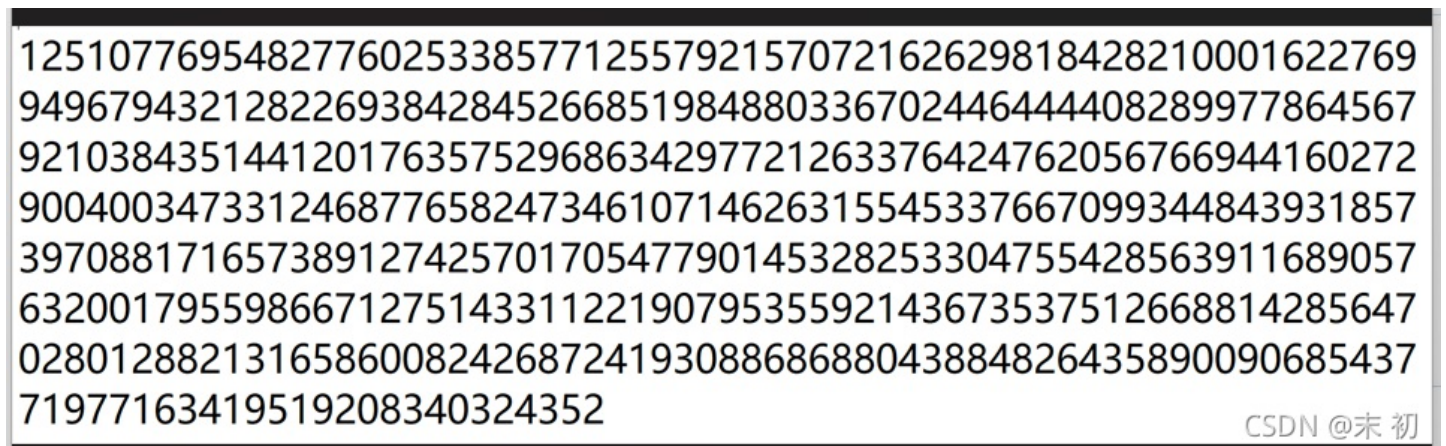
```
alternative: 1202m (OK)
alternative: o_madn (OK)
alternative: paXr_b (OK)
alternative: wx_LuI (OK)
PS D:\Tools\Misc\crc32>
```

得到密码: `this_Is_Y0ur_pa33w0rd_We1c0m3e`

`blind.png` 存在LSB隐写PNG内容



保存下来使用010Editor将前面的几个干扰字节去掉，得到图片



CSDN @末初

图片OCR: https://www.onlineocr.net/zh_hant/

```
1251077695482776025338577125579215707216262981842821000162276994967943212822693842845266851984880336702446444408
2899778645679210384351441201763575296863429772126337642476205676694416027290040034733124687765824734610714626315
5453376670993448439318573970881716573891274257017054779014532825330475542856391168905763200179559866712751433112
2190795355921436735375126688142856470280128821316586008242687241930886868804388482643589009068543771977163419519
208340324352
```

根据题目给出的提示 **画出自己**；需要用到一种叫 **塔珀自指公式(Tupper's self-referential formula)** 的公式

- <https://zh.wikipedia.org/wiki/%E5%A1%94%E7%8F%80%E8%87%AA%E6%8C%87%E5%85%AC%E5%BC%8F>

脚本参考：<https://www.cnblogs.com/1024th/p/14418846.html>

把 **k** 的值换成上面的数字即可

```

"""
Plot Tupper's self-referential formula
"""
import textwrap
import matplotlib.pyplot as plt

K = 125107769548277602533857712557921570721626298184282100016227699496794321282269384284526685198488033670244644
4408289977864567921038435144120176357529686342977212633764247620567669441602729004003473312468776582473461071462
6315545337667099344843931857397088171657389127425701705477901453282533047554285639116890576320017955986671275143
3112219079535592143673537512668814285647028012882131658600824268724193088686880438848264358900906854377197716341
9519208340324352

H = 17
W = 106

if __name__ == "__main__":
    plt.figure(figsize=(6.8, 4), dpi=600)
    plt.axis("scaled")

    K_ = K//17
    for x in range(W):
        for y in range(H):
            if K_ & 1:
                plt.bar(x+0.5, bottom=y, height=1,
                        width=1, linewidth=0, color="black")
            K_ >>= 1

    plt.figtext(0.5, 0.8, r"\frac{1}{2}\left\lfloor \operatorname{mod}\left(\left\lfloor \frac{y}{d}\right\rfloor\right.\right.\left.\left.\right\rfloor 2^{-\left\lfloor x\right\rfloor}\operatorname{mod}\left(\left\lfloor y\right\rfloor, d\right), 2\right)\right\rfloor" % (H, H, H), ha="center", va="bottom", fontsize=18)

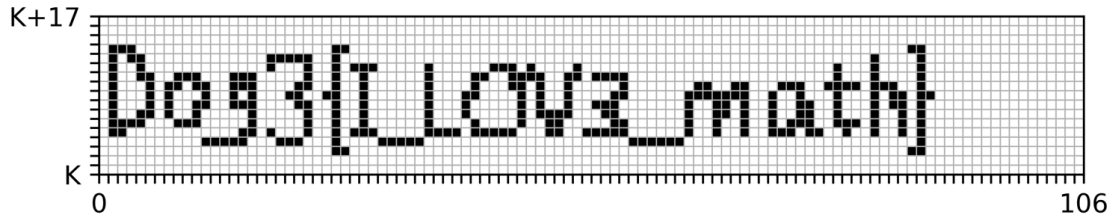
    plt.subplots_adjust(top=0.8, bottom=0.5)
    K_str = textwrap.wrap(str(K), 68)
    K_str[0] = f"K={K_str[0]}"
    for i in range(1, len(K_str)):
        K_str[i] = f" {K_str[i]}".ljust(70)
    K_str = "\n".join(K_str)
    plt.figtext(0.5, 0.45, K_str, fontfamily="monospace", ha="center", va="top")

    plt.xlim((0, W))
    plt.ylim((0, H))
    xticks = list(range(0, W+1))
    xlabels = ["" for i in xticks]
    xlabels[0] = "0"
    xlabels[-1] = str(W)
    plt.xticks(xticks, xlabels)
    yticks = list(range(0, H+1))
    ylabels = ["" for i in yticks]
    ylabels[0] = "K"
    ylabels[-1] = f"K+{H}"
    plt.yticks(yticks, ylabels)
    plt.grid(b=True, linewidth=0.5)

    # plt.show()
    plt.savefig("Tupper-plot.png")
    # plt.savefig(fname="name", format="svg")

```

$$\frac{1}{2} < \lfloor \text{mod}(\lfloor \frac{y}{17} \rfloor 2^{-17\lfloor x \rfloor} - \text{mod}(\lfloor y \rfloor, 17), 2) \rfloor$$



K=12510776954827760253385771255792157072162629818428210001622769949679
 43212822693842845266851984880336702446444408289977864567921038435144
 12017635752968634297721263376424762056766944160272900400347331246877
 65824734610714626315545337667099344843931857397088171657389127425701
 70547790145328253304755428563911689057632001795598667127514331122190
 79535592143673537512668814285647028012882131658600824268724193088686
 8804388482643589009068543771977163419519208340324352

D0g3{I_Lov3_math}