BUUCTF misc 解题记录 一（超级详细）

原创

Vayn3 于 2021-03-16 23:37:03 发布

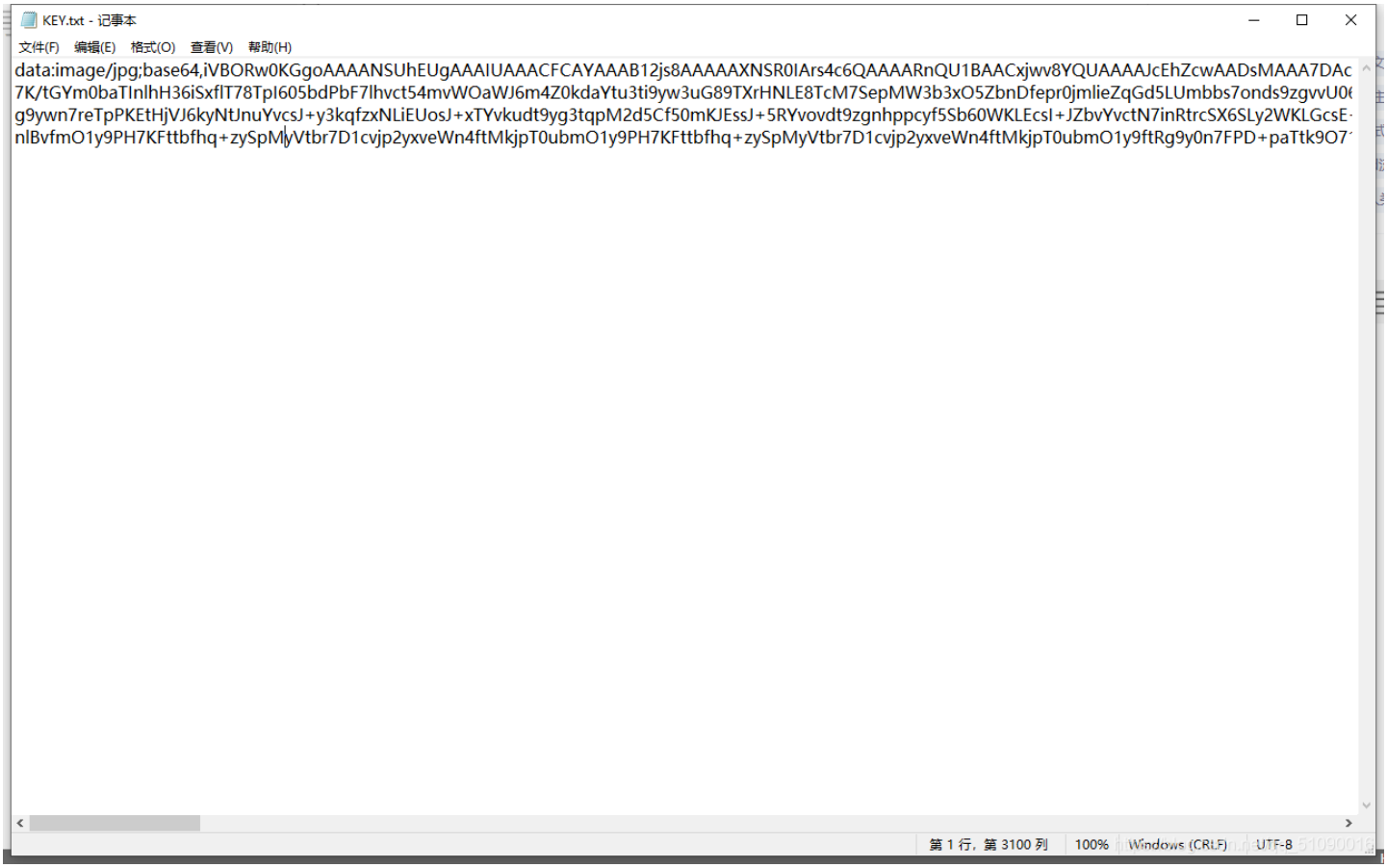
分类专栏： 笔记 文章标签： 安全

持续更新一些不会的题目，有错误的还请大佬指点

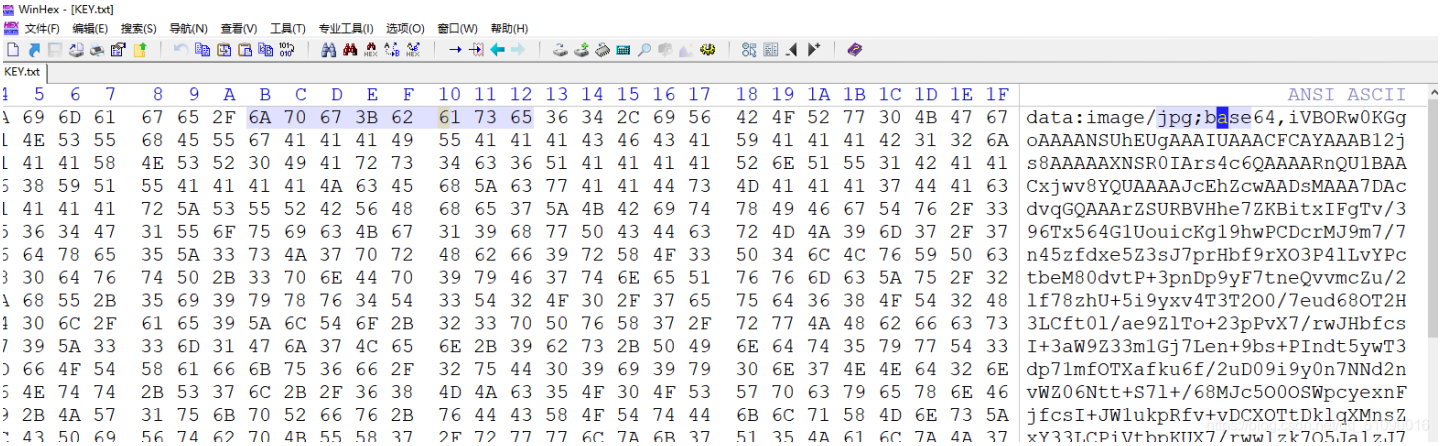
N种方法解决(运行不了的文件尝试改成txt文件)

附件下载是一个exe文件，我还运行不了。。这咋办

改成txt文件看看



好像有什么东西藏在里面 前面有个jpg 还有base，用winhex打开看看



到这没思路了，看wp

base64可以表示图片

这个让我这个菜鸡人傻了

转换网站点这

转换完还能帮你另存为，真不错

扫一下就OK了

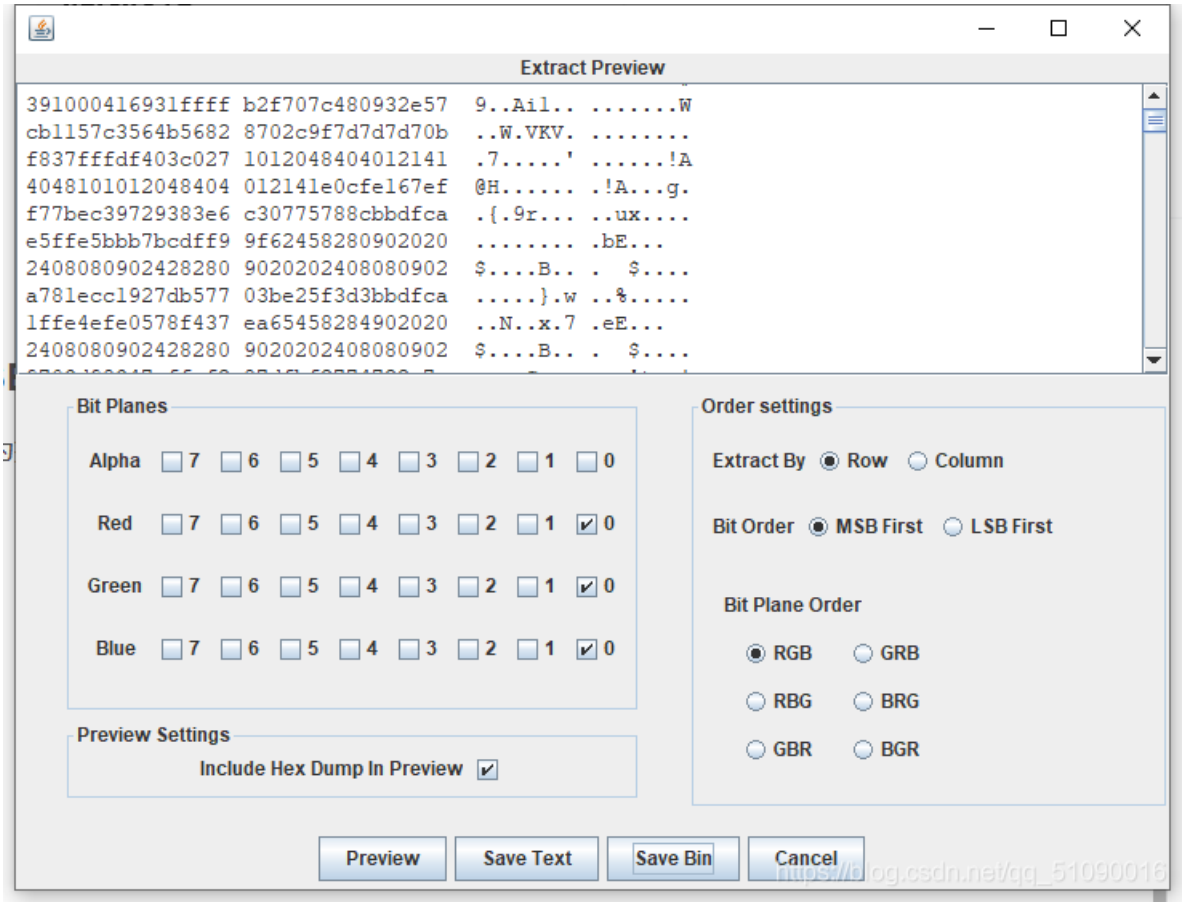
KEY{dca57f966e4e4e31fd5b15417da63269}

扫描识别

LSB（图片通道上方隐藏信息用save bin）

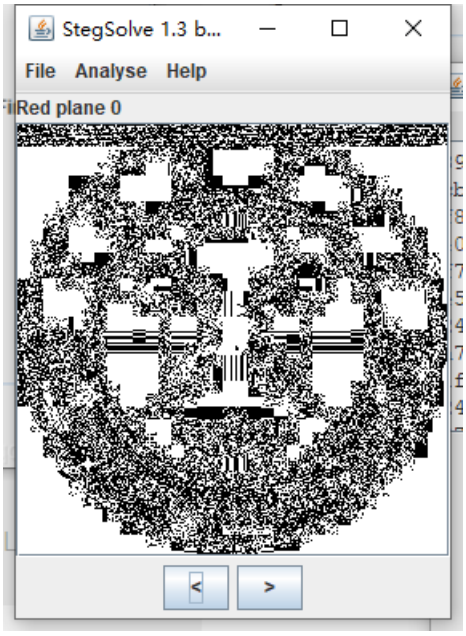
因为提示lsb，所以先用stegsolve打开看看

这样设置就行了



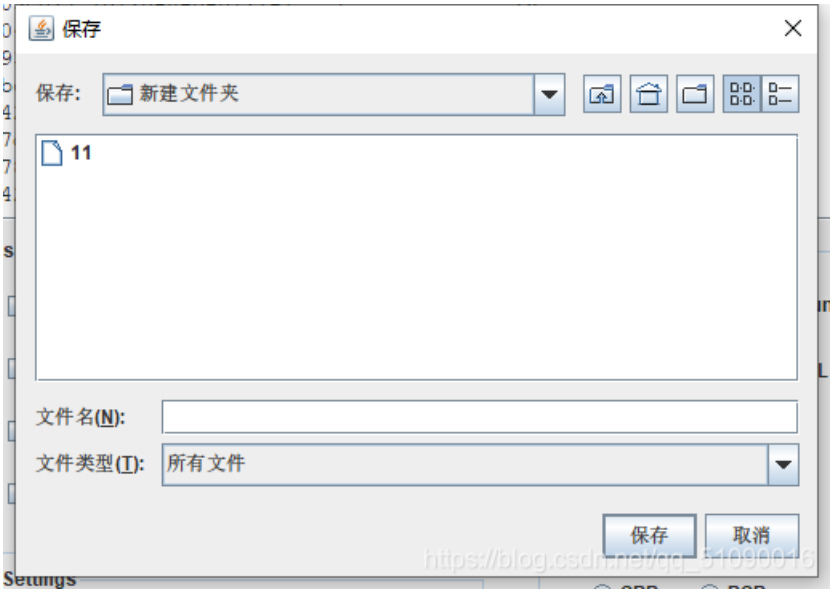
开始我以为这就完事了，傻傻的把{}里面的当成flag了

输了发现不是，然后又更换了几个选项，还是不行，只好看看wp

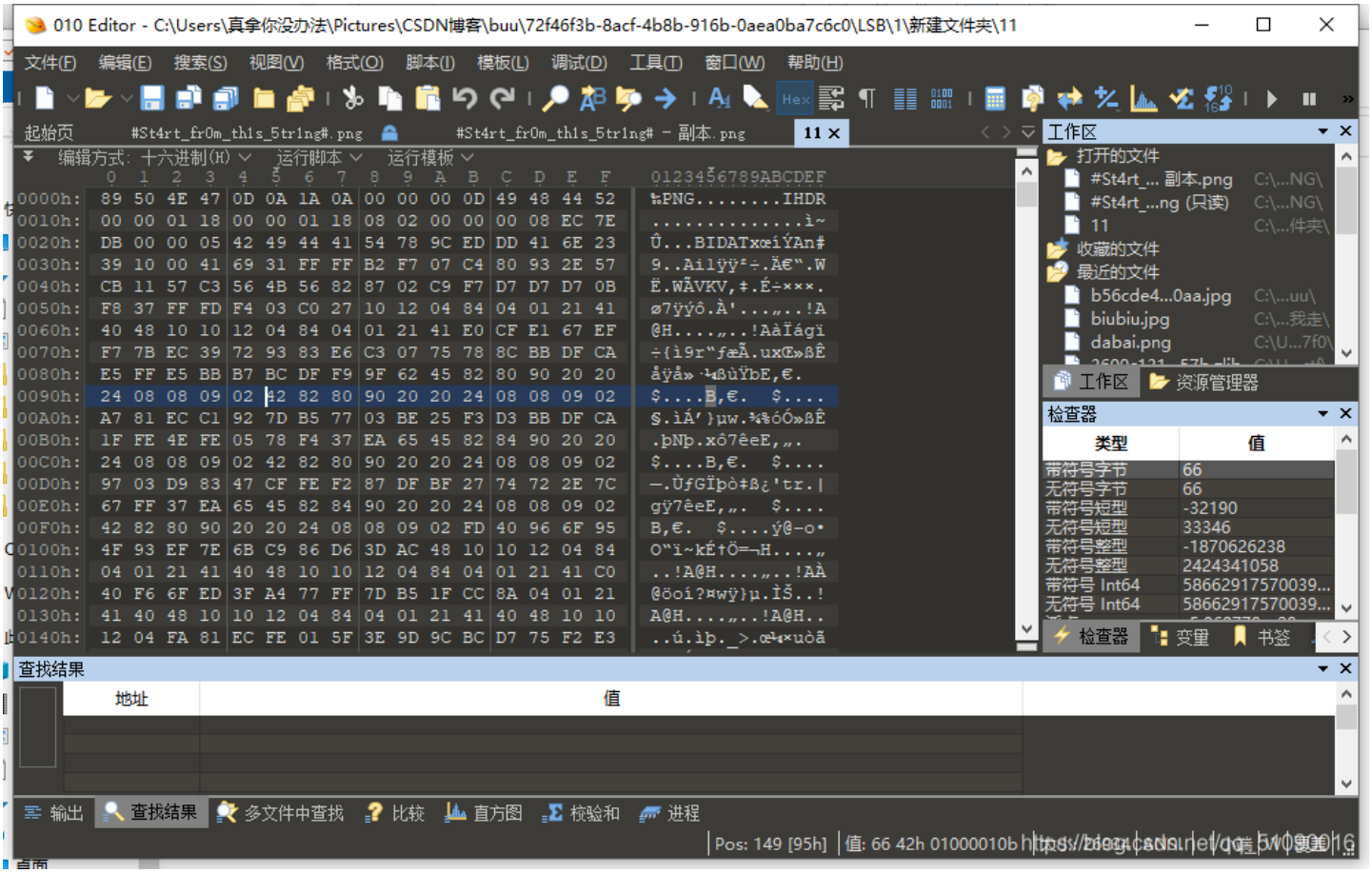


原来要查看图片通道：

发现这上面好像隐藏着东西，这样就要用到save bin的操作



保存后用哦010打开看看



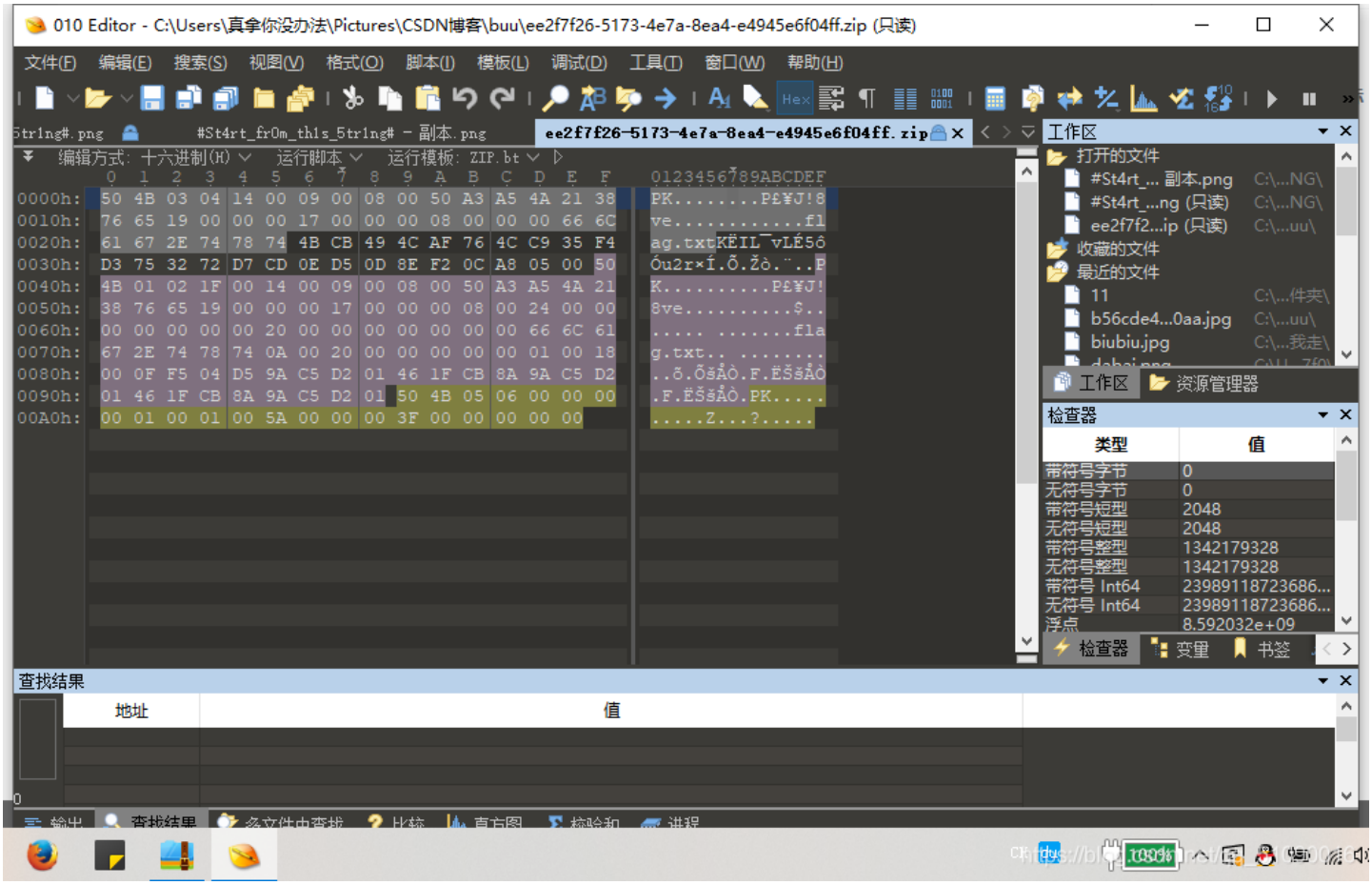
发现是png文件

这样就有了

zip伪加密

刚好借着这题把我没搞懂的伪加密弄懂

首先肯定是用010打开



我记得伪加密应该是把什么09改成00吧，既然不记得了，来系统的学习一下，转载大佬的博客

来了解一下ZIP文件的组成

一个 ZIP 文件由三个部分组成：

压缩源文件数据区+压缩源文件目录区+压缩源文件目录结束标志

a.压缩源文件数据区：

50 4B 03 04：这是头文件标记（0x04034b50）

14 00：解压文件所需 pkware 版本

00 00：全局方式位标记（有无加密，奇数加密，偶数无加密）

08 00：压缩方式

5A 7E：最后修改文件时间

F7 46：最后修改文件日期

16 B5 80 14：CRC-32校验（1480B516）

19 00 00 00：压缩后尺寸（25）

17 00 00 00：未压缩尺寸（23）

07 00：文件名长度

00 00：扩展记录长度

6B65792E7478740BCECC750E71ABCE48CDC9C95728CECC2DC849AD284DAD0500

b.压缩源文件目录区：

1

50 4B 01 02：目录中文件文件头标记(0x02014b50)

3F 00：压缩使用的 pkware 版本

14 00：解压文件所需 pkware 版本

00 00：全局方式位标记（有无加密，奇数加密，偶数无加密）

08 00：压缩方式

5A 7E：最后修改文件时间

F7 46：最后修改文件日期

16 B5 80 14：CRC-32校验（1480B516）

19 00 00 00：压缩后尺寸（25）

17 00 00 00：未压缩尺寸（23）

07 00：文件名长度

24 00：扩展字段长度

00 00：文件注释长度

00 00：磁盘开始号

00 00：内部文件属性

20 00 00 00：外部文件属性

00 00 00 00：局部头部偏移量

6B65792E7478740A002000000000000010018006558F04A1CC5D001BDEBDD3B1CC5D001BDEBDD3B1CC5D001

c.压缩源文件目录结束标志：

1

50 4B 05 06：目录结束标记

00 00：当前磁盘编号

00 00：目录区开始磁盘编号

01 00：本磁盘上纪录总数

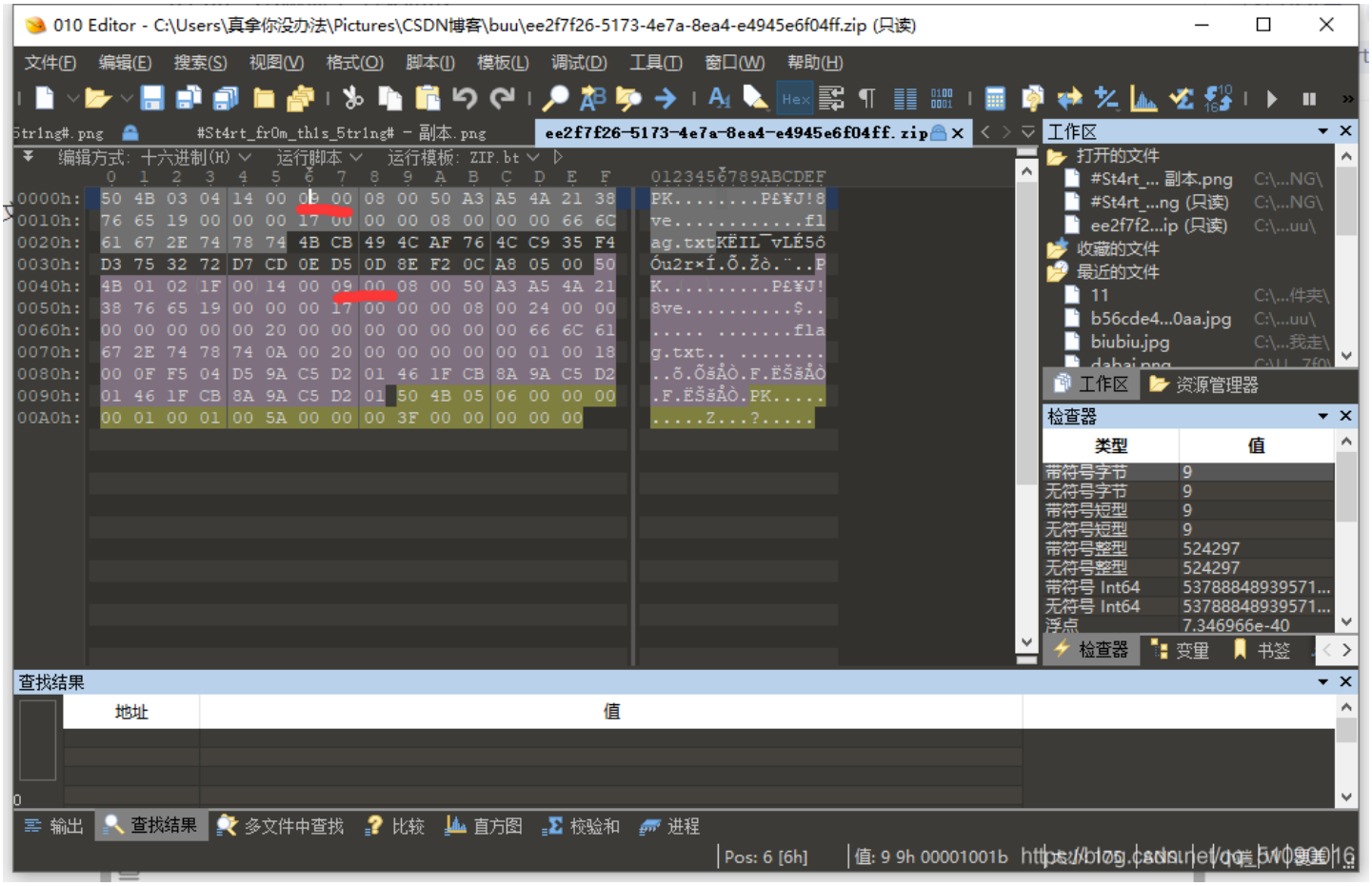
01 00：目录区中纪录总数

59 00 00 00：目录区尺寸大小

3E 00 00 00：目录区对第一张磁盘的偏移量

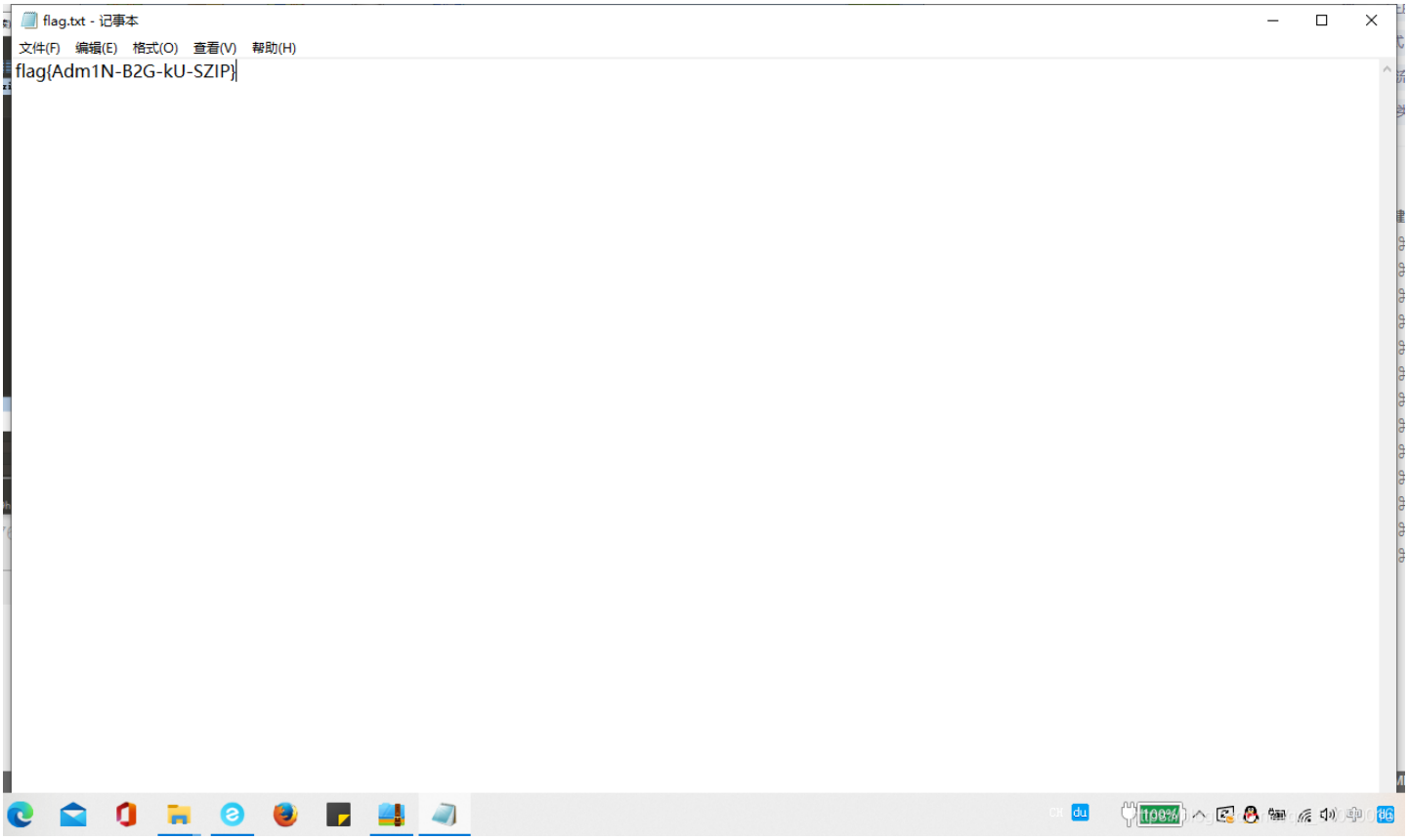
00 00 1A：ZIP 文件注释长度

也就是说，我们主要看的就是全局方式位标记，分别在这两个地方：



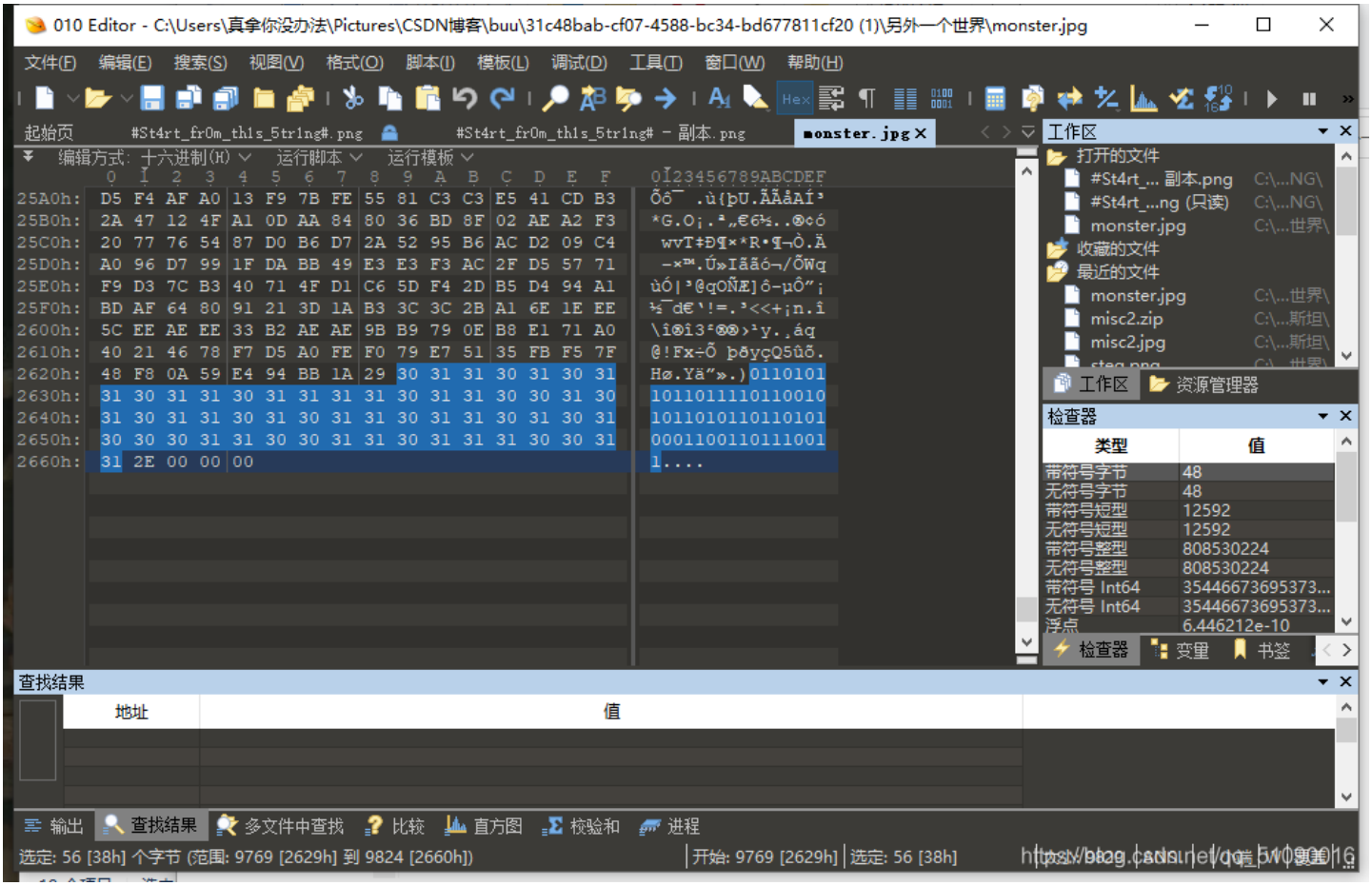
记住它就行了

把这两个都改成00



另外一个世界（二进制转字符：ASCII码）

图片用010打开看到：



这么一串二进制肯定有问题，我开始的想法是转成16进制再转字符

得到这个
然后一直不对，看wp，答案是这个

koekj3s

wp是将每八位二进制数字分开，用ASCII码得到的

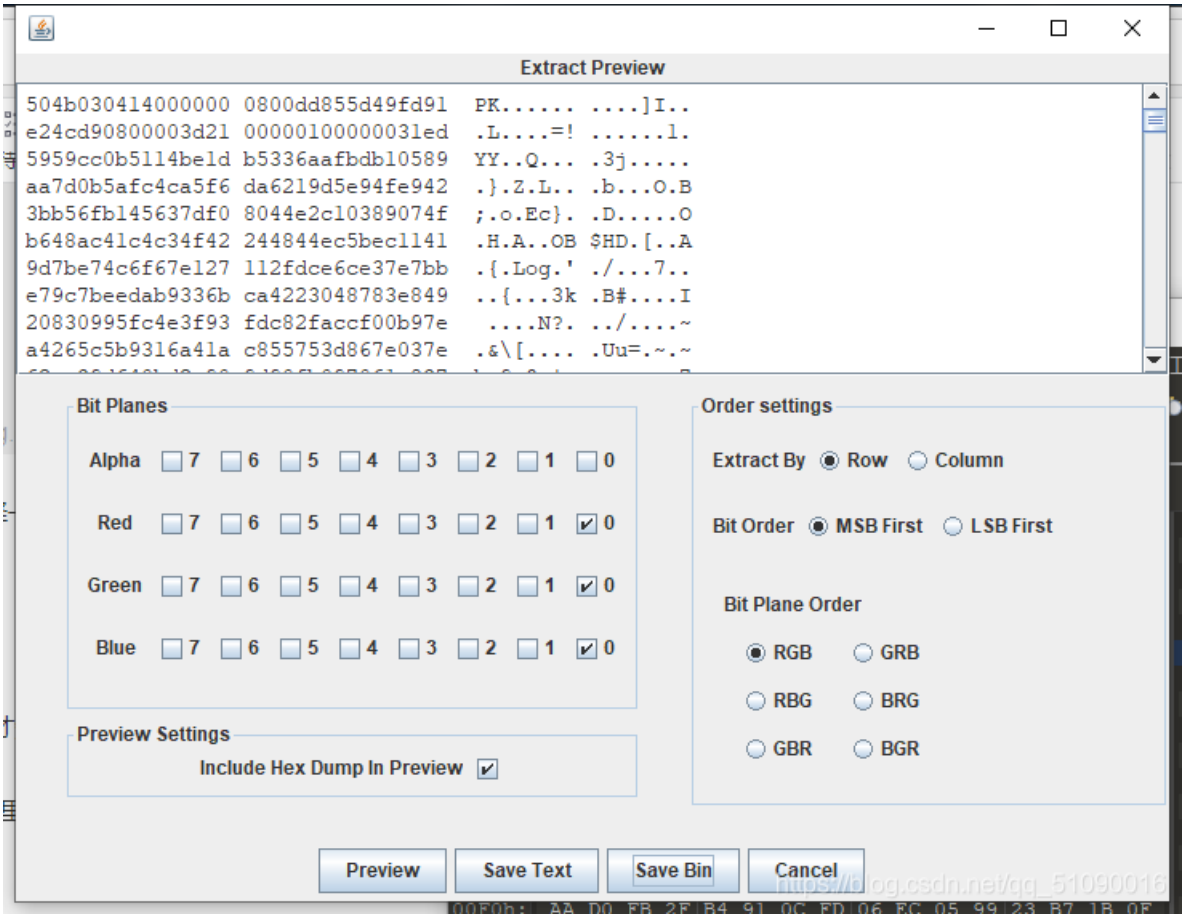
```
01101011
01101111
01100101
01101011
01101010
00110011
01110011
```

为什么就差一个字母不一样？不懂，有大佬解释一下嘛

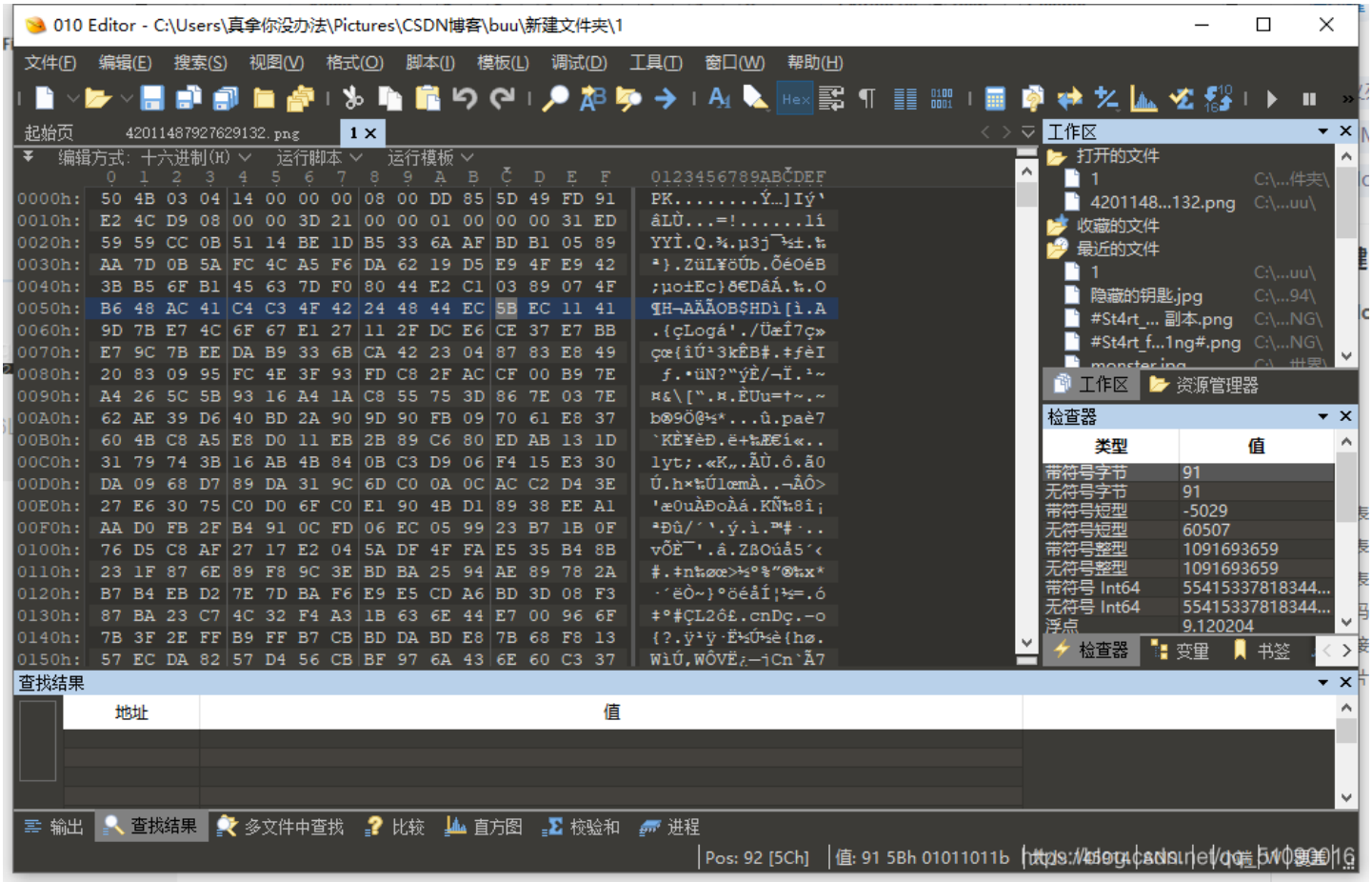
FLAG（save bin 分析文件类型）

这题是因为写了一半卡住了没思路。。看了wp才知道可惜了

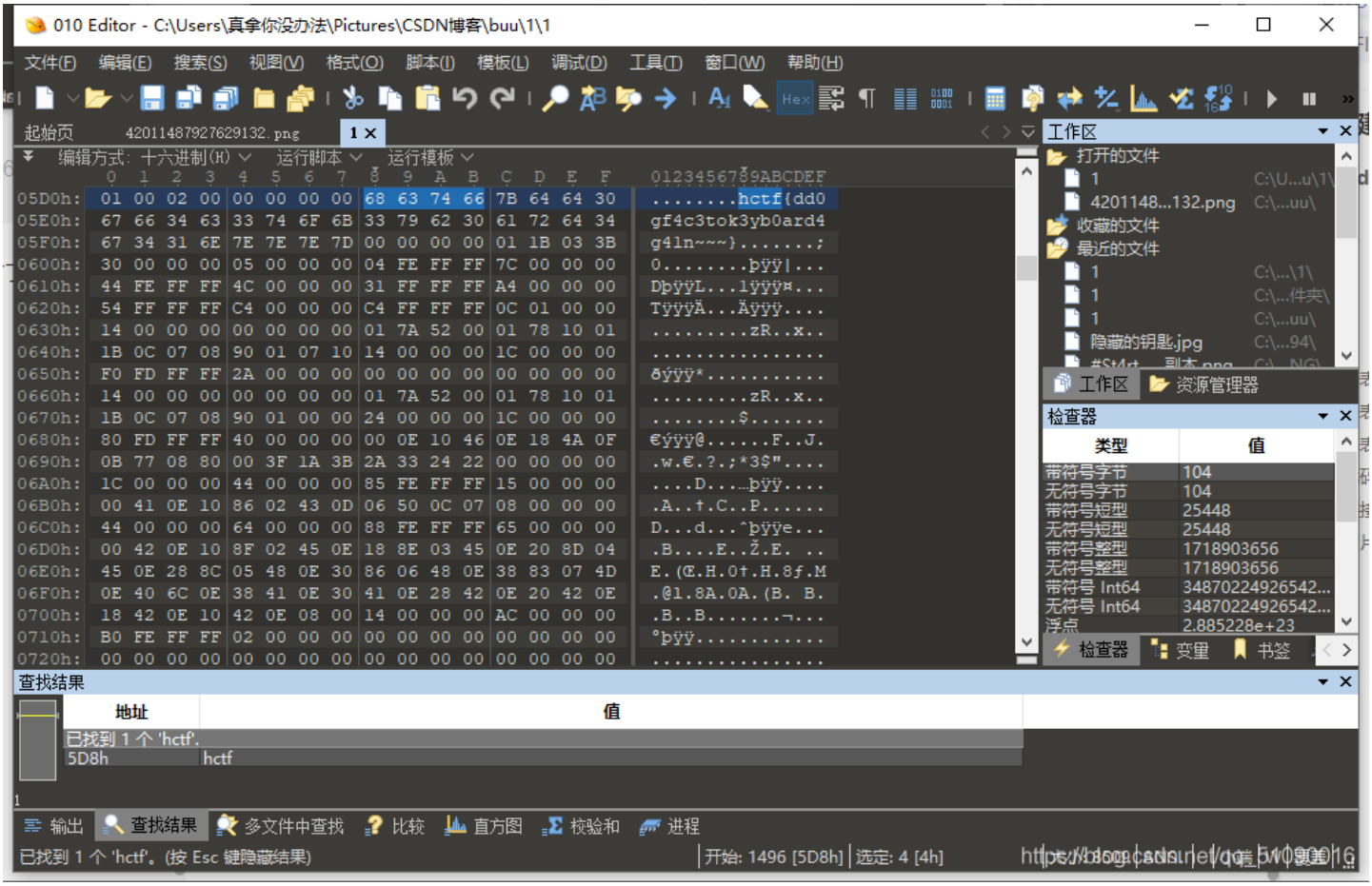
其他方法都试过了，010打开看的时候也没发现里面藏了文件，所以最后用stegsolve打开看看是不是lsb



因为前面那题save bin 的先例，所以这题我也试了试，用010打开



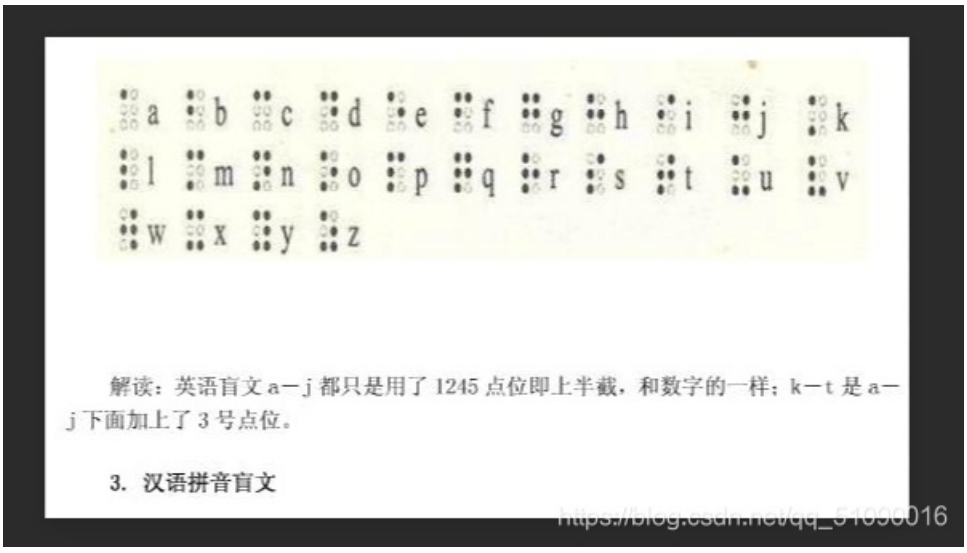
但是我不知道这就是压缩包文件，所以就没做出来，后缀改成zip，解压后用010打开就有了


010 Editor - C:\Users\真拿你没办法\Pictures\CSDN博客\buu\1\1
文件(F) 编辑(E) 搜索(S) 视图(V) 格式(O) 脚本(I) 模板(L) 调试(D) 工具(T) 窗口(W) 帮助(H)
起始页 42011487927629132.png 1
编辑方式: 十六进制(H) 运行脚本 运行模板
0 1 2 3 4 5 6 7 8 9 A B C D E F 0123456789ABCDEF
05D0h: 01 00 02 00 00 00 00 00 68 63 74 66 7B 64 64 30 ....hctf{dd0
05E0h: 67 66 34 63 33 74 6F 6B 33 79 62 30 61 72 64 34 gf4c3tok3yb0ard4
05F0h: 67 34 31 6E 7E 7E 7E 7D 00 00 00 00 01 1B 03 3B g41n~~~}.......;
0600h: 30 00 00 00 05 00 00 00 04 FE FF FF 7C 00 00 00 0........þÿÿ|...
0610h: 44 FE FF FF 4C 00 00 00 31 FF FF FF A4 00 00 00 DþÿÿL...1ÿÿÿ¤...
0620h: 54 FF FF FF C4 00 00 00 C4 FF FF FF 0C 01 00 00 TÿÿÿÄ...Äÿÿÿ....
0630h: 14 00 00 00 00 00 00 00 01 7A 52 00 01 78 10 01 ....zR..x..
0640h: 1B 0C 07 08 90 01 07 10 14 00 00 00 1C 00 00 00 ....
0650h: F0 FD FF FF 2A 00 00 00 00 00 00 00 00 00 00 00 ðýÿÿ*...........
0660h: 14 00 00 00 00 00 00 00 01 7A 52 00 01 78 10 01 ....zR..x..
0670h: 1B 0C 07 08 90 01 00 00 24 00 00 00 1C 00 00 00 ....$.......
0680h: 80 FD FF FF 40 00 00 00 00 0E 10 46 0E 18 4A 0F €ýÿÿ@......F..J.
0690h: 0B 77 08 80 00 3F 1A 3B 2A 33 24 22 00 00 00 00 .w.€.?.;*3$"....
06A0h: 1C 00 00 00 44 00 00 00 85 FE FF FF 15 00 00 00 ....D....þÿÿ....
06B0h: 00 41 0E 10 86 02 43 0D 06 50 0C 07 08 00 00 00 .A..†.C..P......
06C0h: 44 00 00 00 64 00 00 00 88 FE FF FF 65 00 00 00 D...d...ˆþÿÿe...
06D0h: 00 42 0E 10 8F 02 45 0E 18 8E 03 45 0E 20 8D 04 .B....E..Ž.E. ..
06E0h: 45 0E 28 8C 05 48 0E 30 86 06 48 0E 38 83 07 4D E.(Œ.H.0†.H.8ƒ.M
06F0h: 0E 40 6C 0E 38 41 0E 30 41 0E 28 42 0E 20 42 0E .@l.8A.0A.(B. B.
0700h: 18 42 0E 10 42 0E 08 00 14 00 00 00 AC 00 00 00 .B..B.......¬...
0710h: B0 FE FF FF 02 00 00 00 00 00 00 00 00 00 00 00 °þÿÿ............
0720h: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ....
工作区
打开的文件
1 C:\U...u\1\
4201148...132.png C:\...uu\
收藏的文件
最近的文件
1 C:\...\1\
1 C:\...件夹\
1 C:\...uu\
隐藏的钥匙.jpg C:\...94\
工作区 资源管理器
检查器
类型 值
带符号字节 104
无符号字节 104
带符号短型 25448
无符号短型 25448
带符号整型 1718903656
无符号整型 1718903656
带符号 Int64 34870224926542...
无符号 Int64 34870224926542...
浮点 2.885228e+23
检查器 变量 书签
查找结果
地址 值
已找到 1 个 'hctf'.
5D8h hctf
输出 查找结果 多文件中查找 比较 直方图 校验和 进程
已找到 1 个 'hctf'。(按 Esc 键隐藏结果)
开始: 1496 [5D8h] 选定: 4 [4h]

假如给我三天光明（盲文）

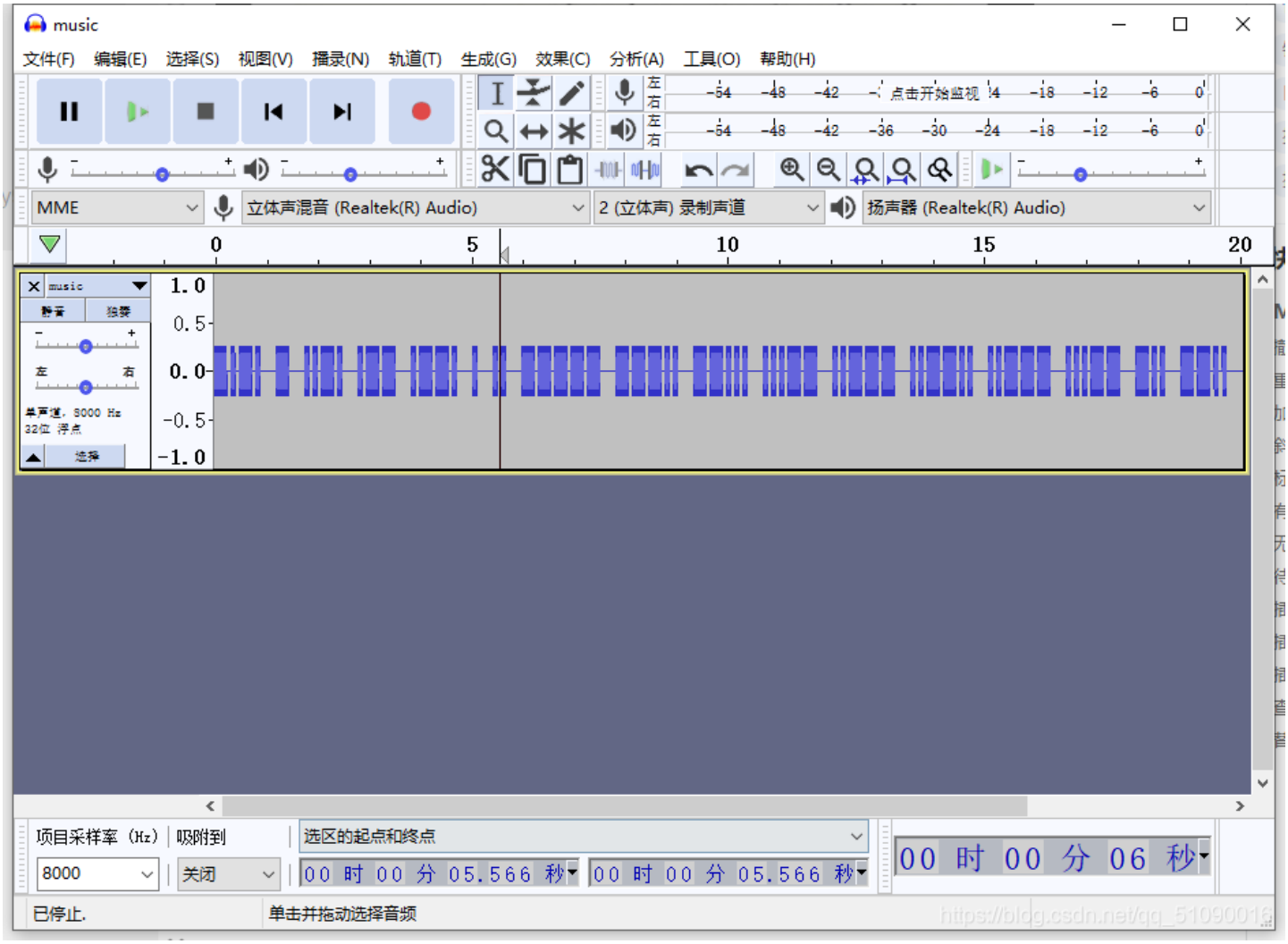
实在没想到下面这个是盲文（不应该）

搜盲文对照一下



也就是：kmdonowg

这样就能解压出音频文件了



应该是摩斯电码：-.-. - ...-. .-- .-. . ... ----- —... --... ...-- ...— ...-... ...— ...-- -... --...

按粗细分-和.

这样就得到了：

但是最后提交的时候还要改成小写，这是为什么。。

后门查杀（用d盾查杀木马）

差点以为点错地方了，还以为是web题呢

这里需要用到一个工具会比较简单：d盾

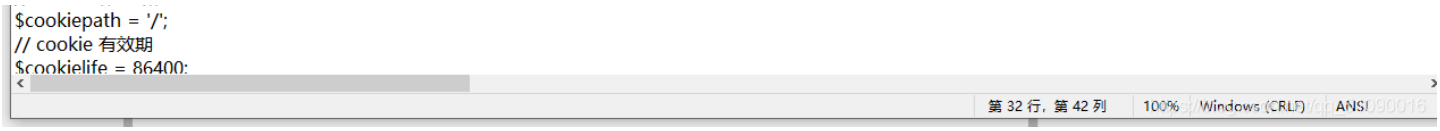
直接扫描大文件夹（文件太多了一个个看太麻烦）

webshell应该就是一句话木马吧，那查看一下下面的这个

include.php - 记事本

文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)

```
if( strpos( strtolower( $_SERVER['HTTP_USER_AGENT'] ), 'bot' ) !== false ) {
		header('HTTP/1.0 404 Not Found');
		exit;
}
ob_start();
$mtime = explode(' ', microtime());
$starttime = $mtime[1] + $mtime[0];
define('SA_ROOT', str_replace('\\', '/', dirname(__FILE__)).'/');
define('SELF', $_SERVER['PHP_SELF'] ? $_SERVER['PHP_SELF'] : $_SERVER['SCRIPT_NAME']);
define('IS_WIN', DIRECTORY_SEPARATOR == '\\');
define('IS_GPC', get_magic_quotes_gpc());
$dis_func = get_cfg_var('disable_functions');
define('IS_PHPINFO', (!eregi("phpinfo",$dis_func)) ? 1 : 0 );

if( IS_GPC ) {
		$_POST = s_array($_POST);
}
$P = $_POST;
unset($_POST);
/*====================== 程序配置 ======================*/

//echo encode_pass('angel');exit;
// 如果需要密码验证,请修改登陆密码,留空为不需要验证
$pass  = '6ac45fb83b3bc355c024f5034b947dd3'; //angel

//如您对 cookie 作用范围有特殊要求, 或登录不正常, 请修改下面变量, 否则请保持默认
// cookie 前缀
$cookiepre = '';
// cookie 作用域
$cookiedomain = '';
// cookie 作用路径
```

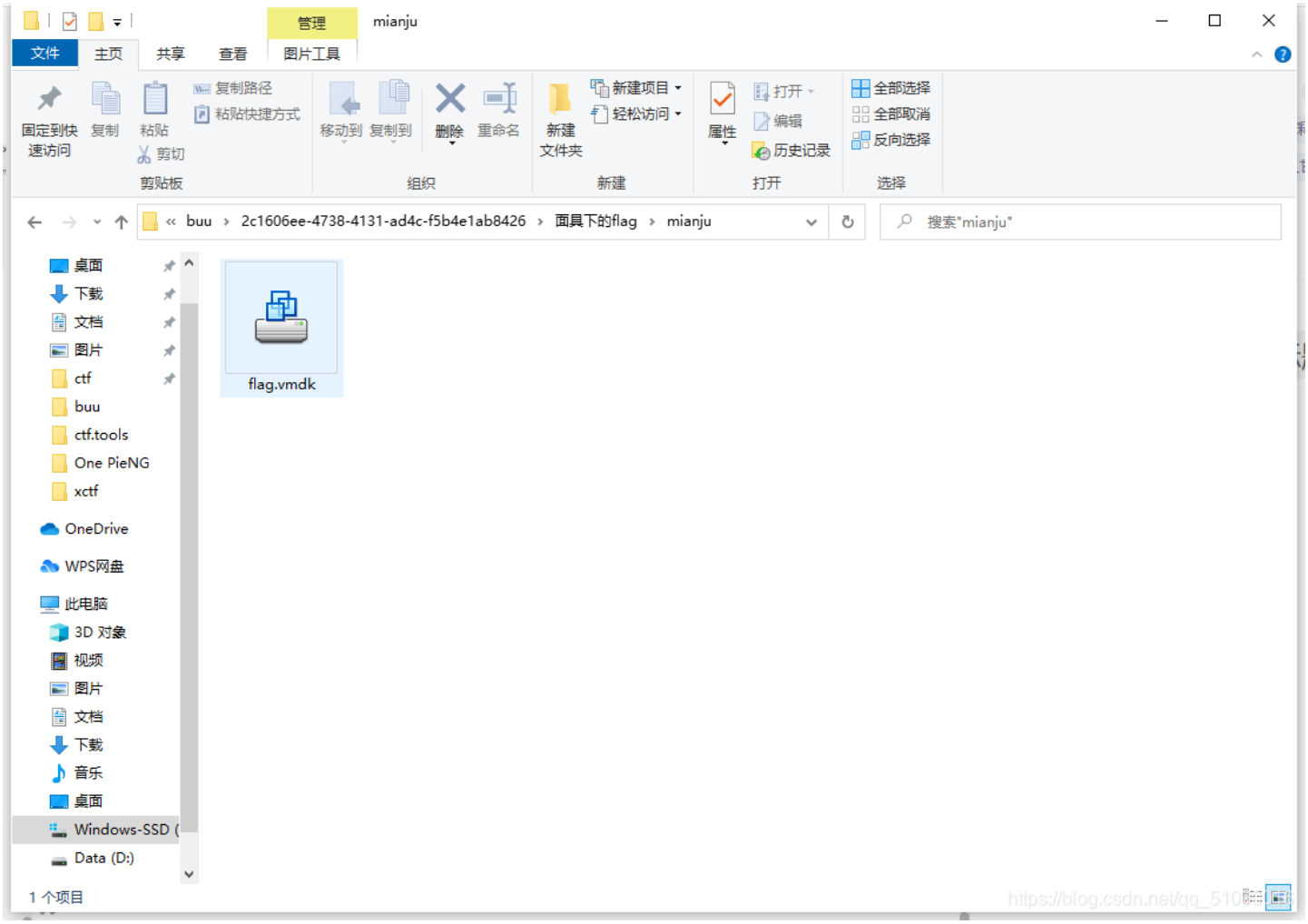


果然发现了

面具下的flag(用7z解压缩vmdk文件)

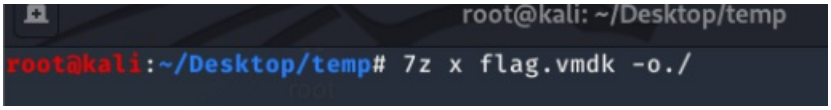
先试用010打开图片，发现里面藏着文件，然后就改成zip后缀试试

得到了这个

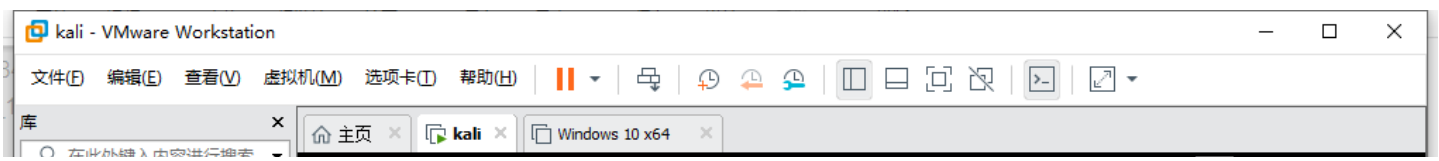


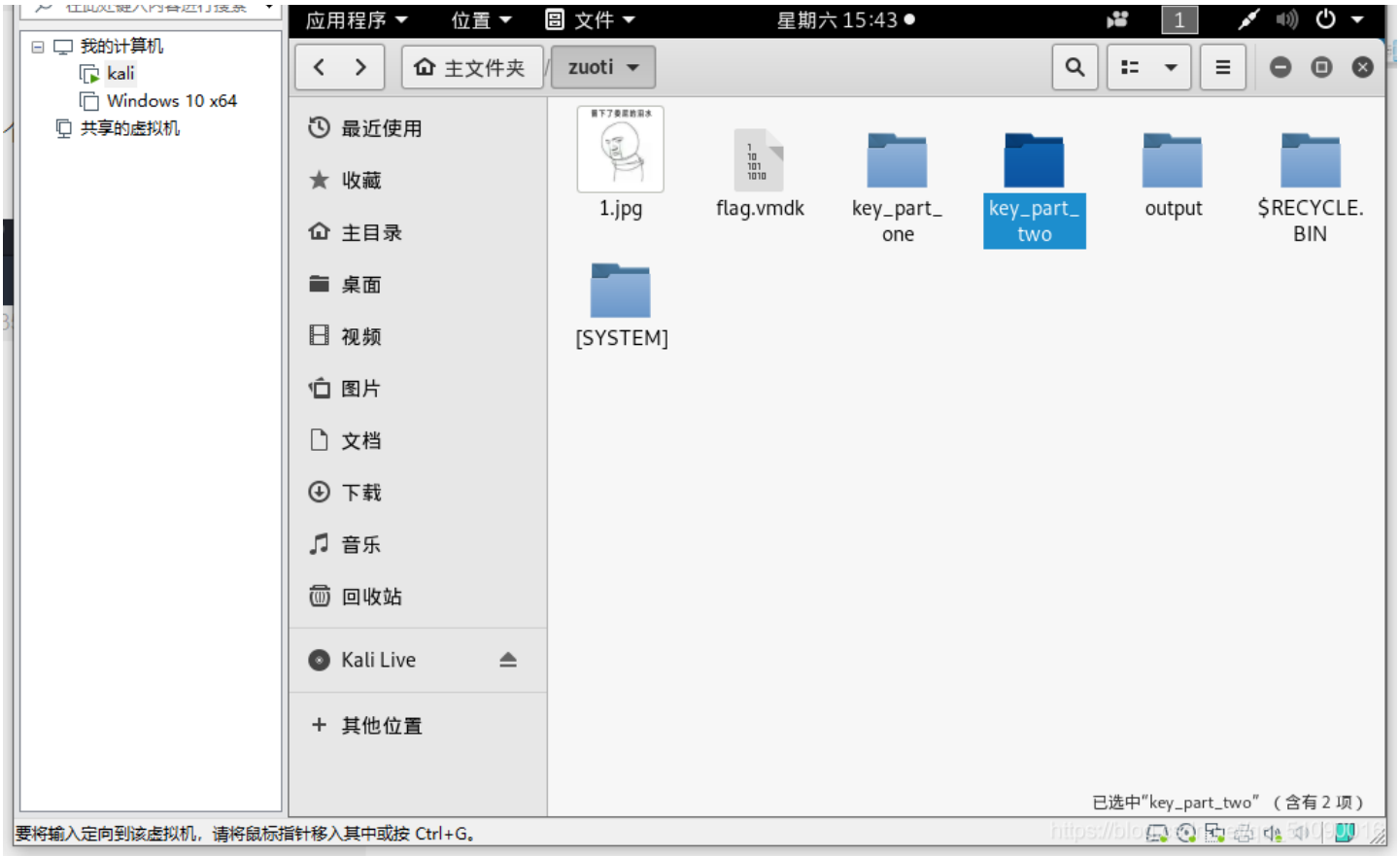
到这我就没思路了。。看wp做的后面

再kali用7z解压缩这个文件（我之前也不知道7z是什么），反正就用这个命令就能解压缩：

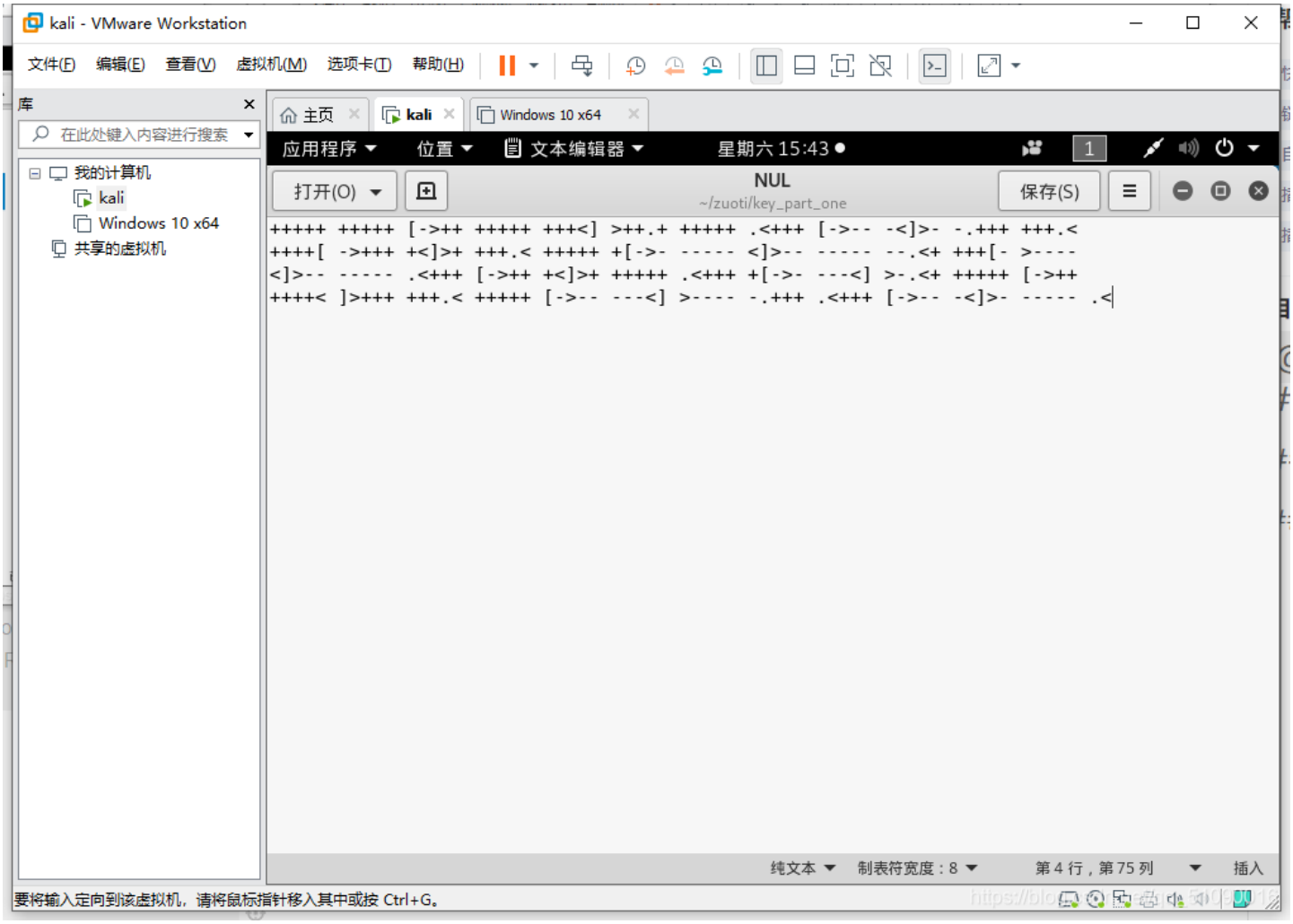


然后就得到一堆文件，每个都看看，看到两个关键文件





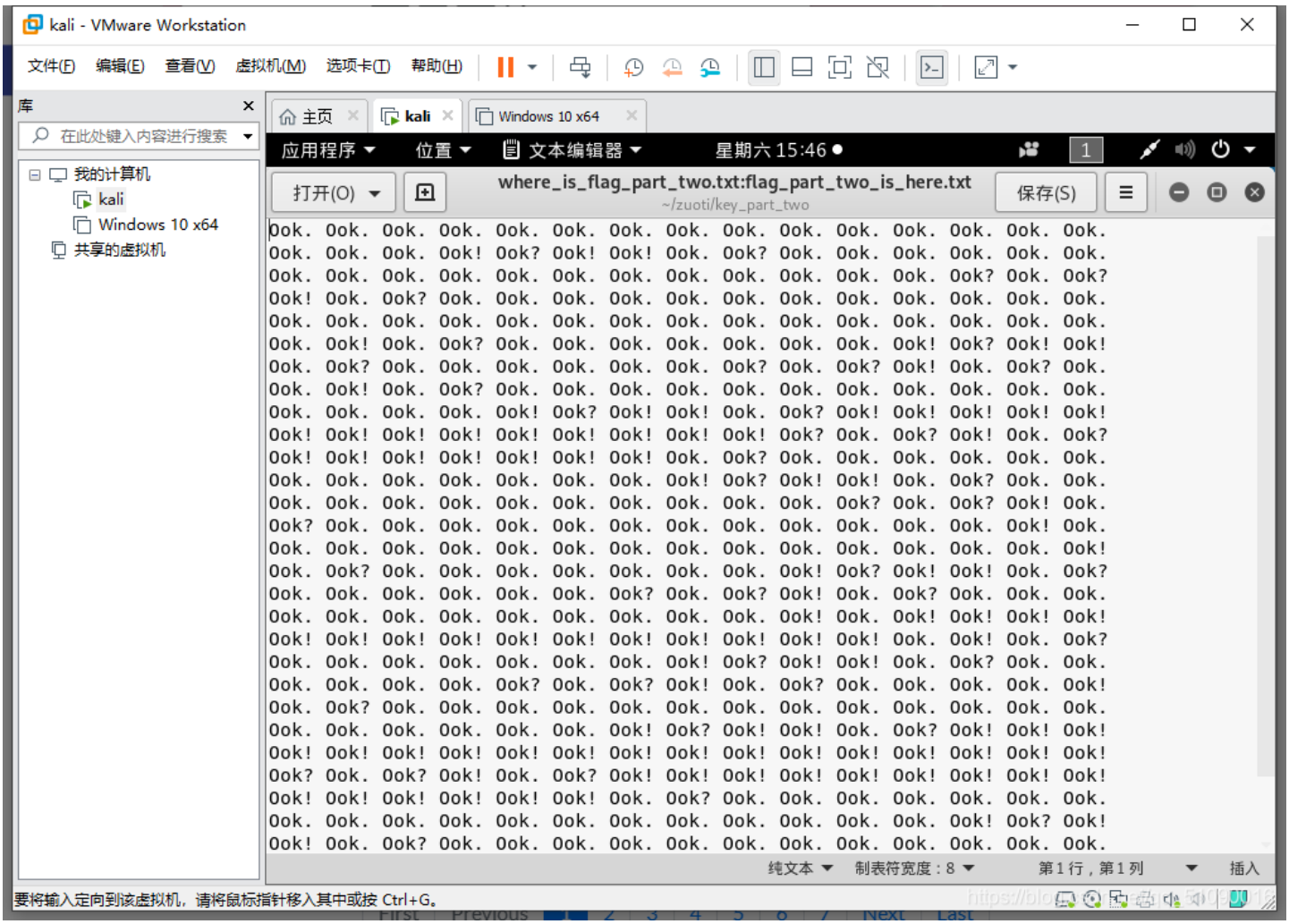
第一个长这样

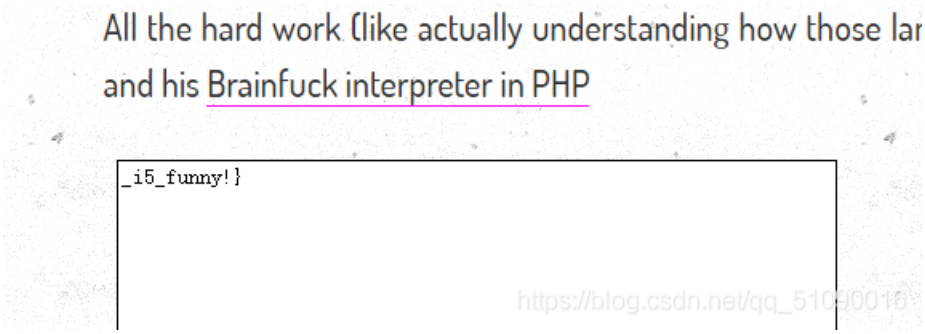


发现是brainfuck

肯定是brainfuck

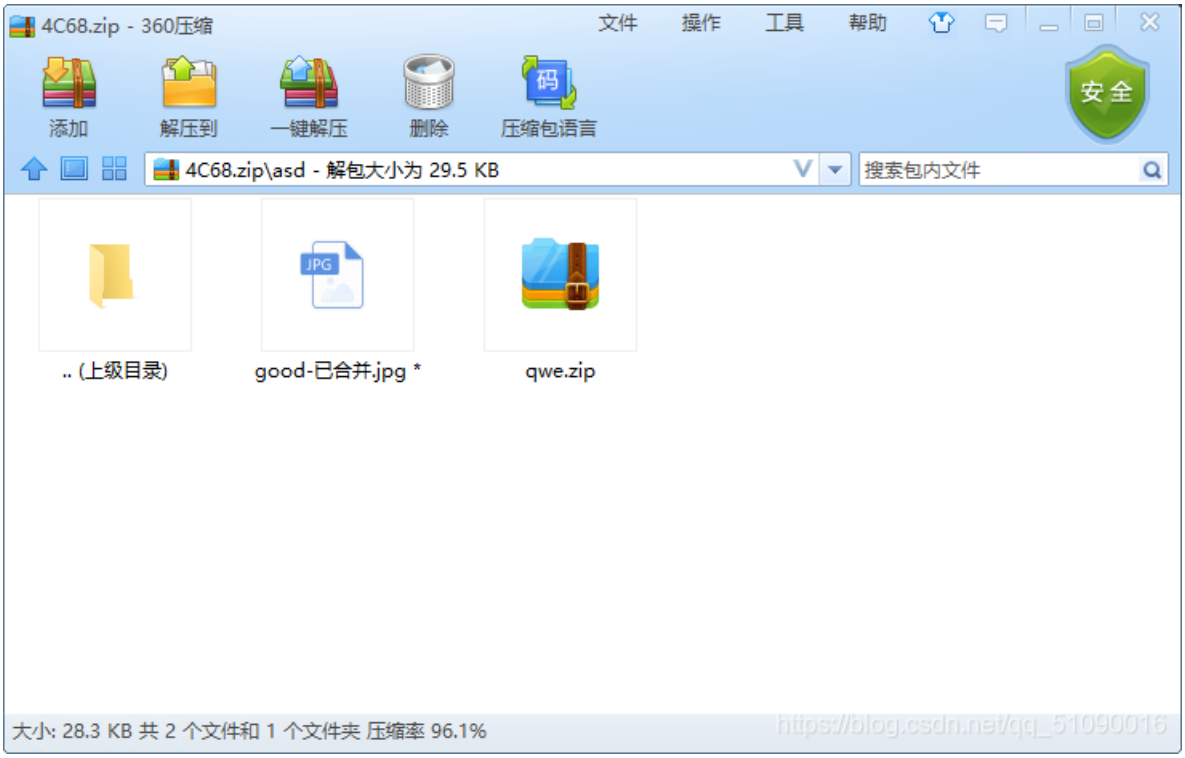
第二个长这样：





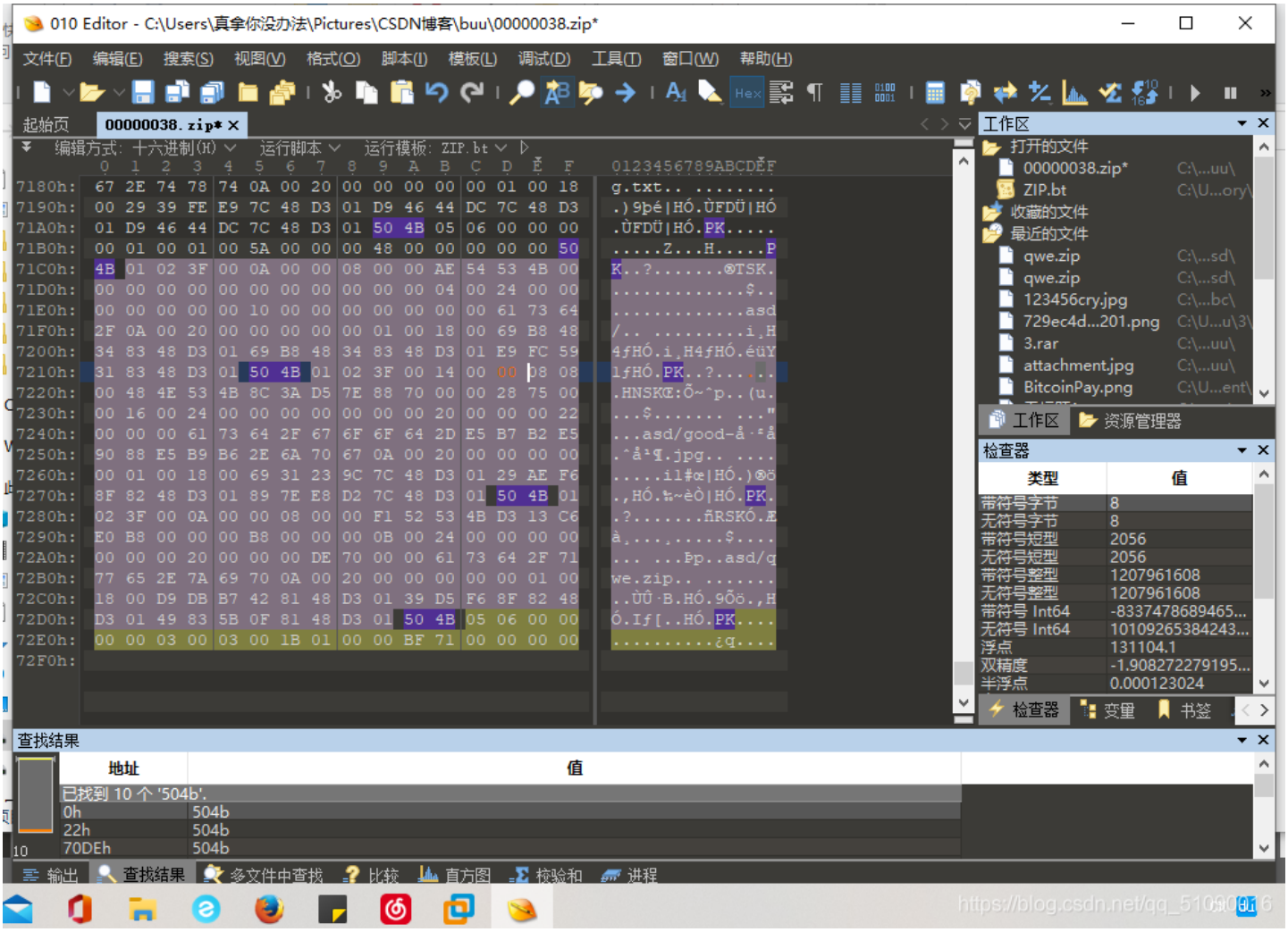
九连环（steghide）

显示010里面发现有flag。txt文件，然后使用binwalk foremos大法分离出一个压缩包和一个图片

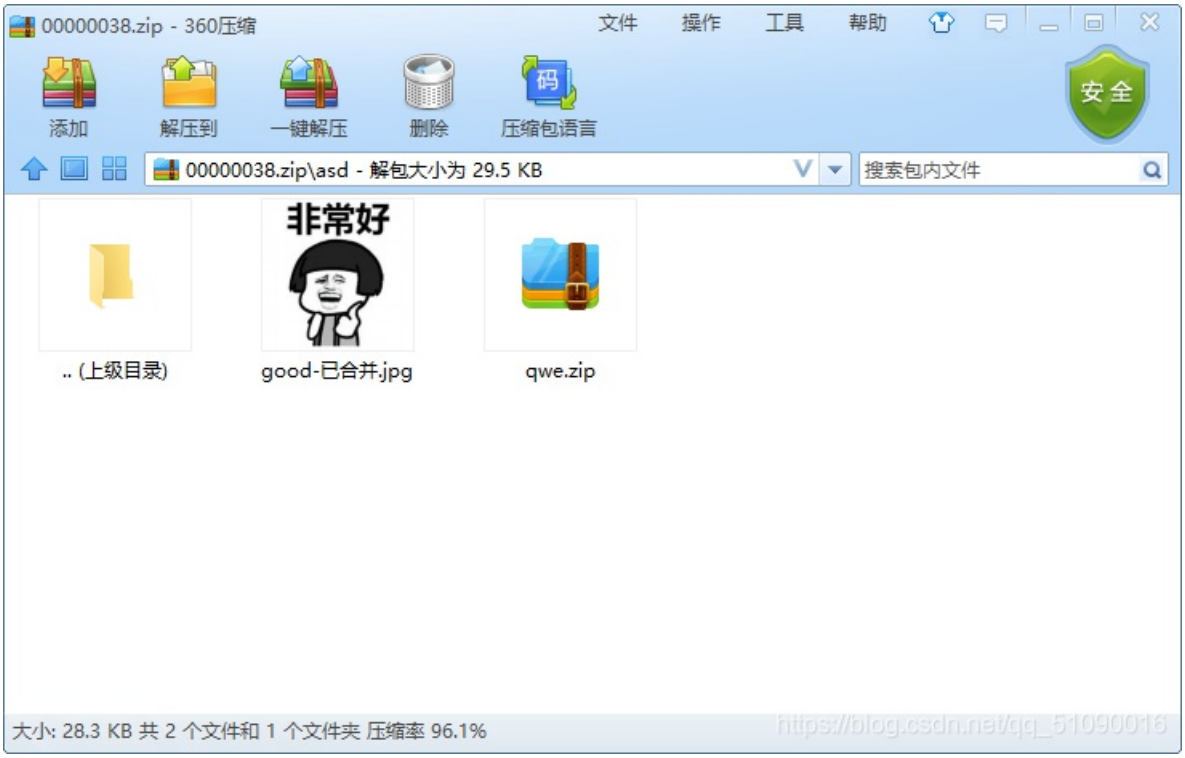


图片损坏了解压不了，压缩包点开需要密码，先看看是不是伪加密

果然，

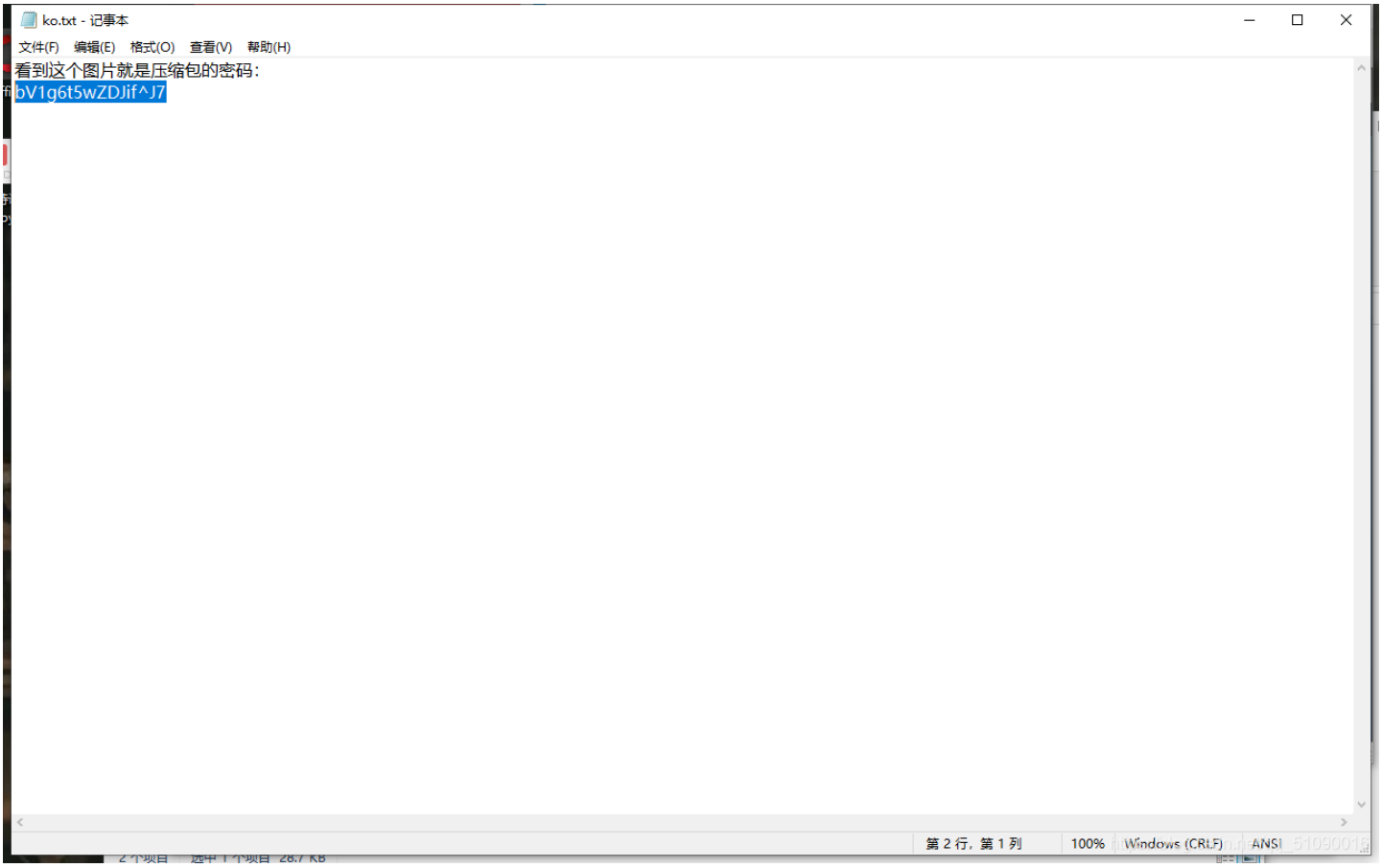

010 Editor - C:\Users\真拿你没办法\Pictures\CSDN博客\buu\00000038.zip*
文件(F) 编辑(E) 搜索(S) 视图(V) 格式(O) 脚本(I) 模板(L) 调试(D) 工具(T) 窗口(W) 帮助(H)
起始页
00000038.zip*
编辑方式: 十六进制(H) 运行脚本 运行模板: ZIP.bt
0 1 2 3 4 5 6 7 8 9 A B C D E F 0123456789ABCDEF
7180h: 67 2E 74 78 74 0A 00 20 00 00 00 00 00 01 00 18 g.txt.. ....
7190h: 00 29 39 FE E9 7C 48 D3 01 D9 46 44 DC 7C 48 D3 .)9þé|HÓ.ÙFDÜ|HÓ
71A0h: 01 D9 46 44 DC 7C 48 D3 01 50 4B 05 06 00 00 00 .ÙFDÜ|HÓ.PK.....
71B0h: 00 01 00 01 00 5A 00 00 00 48 00 00 00 00 00 50 .....Z...H.....P
71C0h: 4B 01 02 3F 00 0A 00 00 08 00 00 AE 54 53 4B 00 K..?.......®TSK.
71D0h: 00 00 00 00 00 00 00 00 00 00 00 00 04 00 24 00 ....$..
71E0h: 00 00 00 00 00 10 00 00 00 00 00 00 00 61 73 64 ....asd
71F0h: 2F 0A 00 20 00 00 00 00 00 01 00 18 00 69 B8 48 /.. ....i,H
7200h: 34 83 48 D3 01 69 B8 48 34 83 48 D3 01 E9 FC 59 4ƒHÓ.i¸H4ƒHÓ.éüY
7210h: 31 83 48 D3 01 50 4B 01 02 3F 00 14 00 00 B8 08 1ƒHÓ.PK..?....¸.
7220h: 00 48 4E 53 4B 8C 3A D5 7E 88 70 00 00 28 75 00 .HNSKŒ:Õ~ˆp..(u.
7230h: 00 16 00 24 00 00 00 00 00 00 00 20 00 00 00 22 ...$....... ...".
7240h: 00 00 00 61 73 64 2F 67 6F 6F 64 2D E5 B7 B2 E5 ...asd/good-å·²å
7250h: 90 88 E5 B9 B6 2E 6A 70 67 0A 00 20 00 00 00 00 .ˆå¹¶.jpg.. ....
7260h: 00 01 00 18 00 69 31 23 9C 7C 48 D3 01 29 AE F6 .....i1#œ|HÓ.)®ö
7270h: 8F 82 48 D3 01 89 7E E8 D2 7C 48 D3 01 50 4B 01 .‚HÓ.‰~èÒ|HÓ.PK.
7280h: 02 3F 00 0A 00 00 08 00 00 F1 52 53 4B D3 13 C6 .?.......ñRSKÓ.Æ
7290h: E0 B8 00 00 00 B8 00 00 00 0B 00 24 00 00 00 00 à¸...¸.....$....
72A0h: 00 00 00 20 00 00 00 DE 70 00 00 61 73 64 2F 71 ... ...Þp..asd/q
72B0h: 77 65 2E 7A 69 70 0A 00 20 00 00 00 00 00 01 00 we.zip.. ....
72C0h: 18 00 D9 DB B7 42 81 48 D3 01 39 D5 F6 8F 82 48 ..ÙÛ·B.HÓ.9Õö.‚H
72D0h: D3 01 49 83 5B 0F 81 48 D3 01 50 4B 05 06 00 00 Ó.Iƒ[..HÓ.PK....
72E0h: 00 00 03 00 03 00 1B 01 00 00 BF 71 00 00 00 00 ....¿q....
72F0h:
工作区
打开的文件
00000038.zip* C:\...uu\
ZIP.bt C:\U...ory\
收藏的文件
最近的文件
qwe.zip C:\...sd\
qwe.zip C:\...sd\
123456cry.jpg C:\...bc\
729ec4d...201.png C:\U...u\3\
3.rar C:\...uu\
attachment.jpg C:\...uu\
BitcoinPay.png C:\U...ent\
工作区 资源管理器
检查器
类型 值
带符号字节 8
无符号字节 8
带符号短型 2056
无符号短型 2056
带符号整型 1207961608
无符号整型 1207961608
带符号 Int64 -8337478689465...
无符号 Int64 10109265384243...
浮点 131104.1
双精度 -1.908272279195...
半浮点 0.000123024
检查器 变量 书签
查找结果
地址 值
已找到 10 个 '504b'.
0h 504b
22h 504b
70DEh 504b
输出 查找结果 多文件中查找 比较 直方图 校验和 进程

这样就能正常解压了

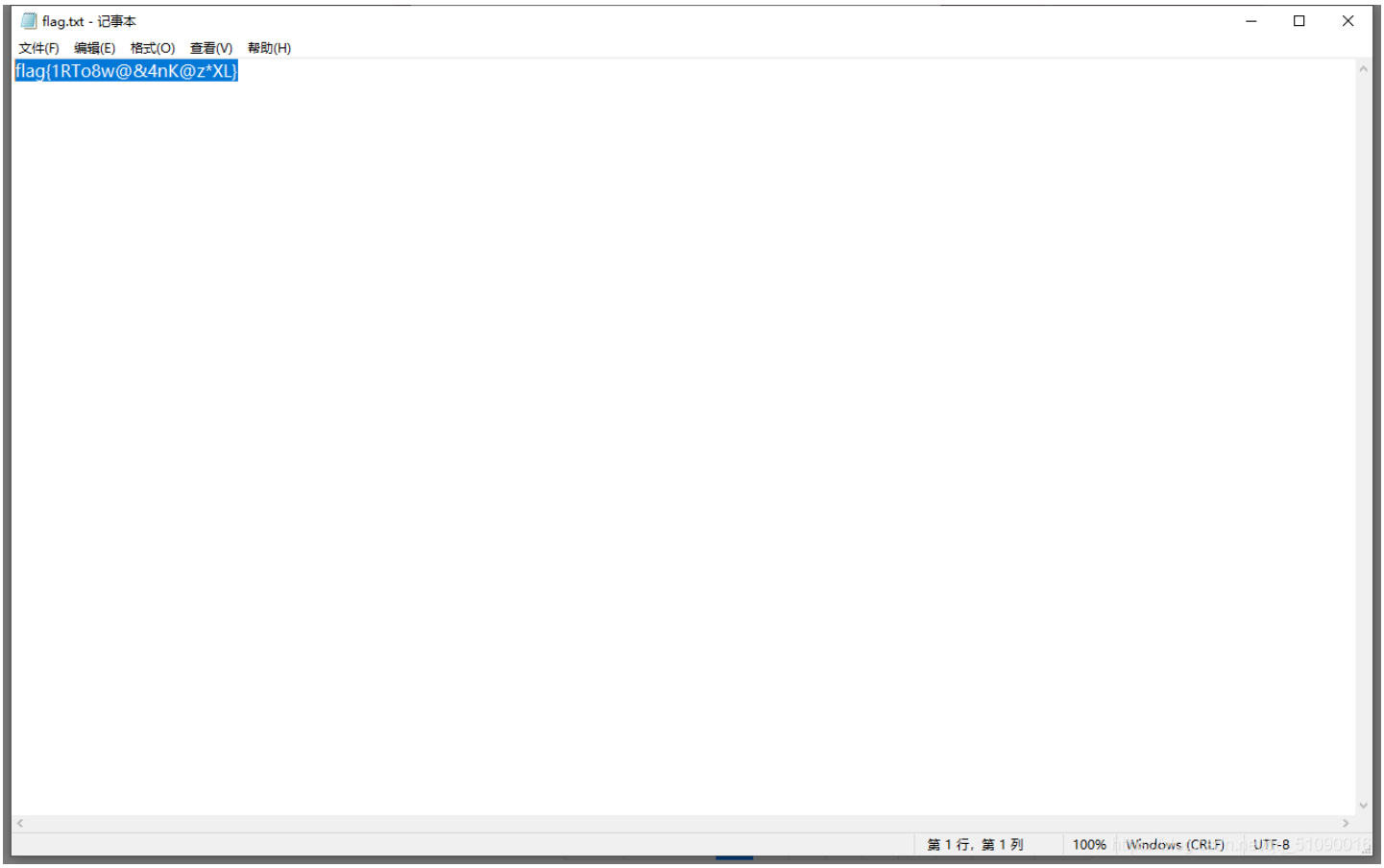


但是另一个压缩包也需要密码，这肯定不是伪加密了，关键应该在图片上：这里涉及到steghide的使用

使用`steghide extract -sf good.jpg`，空密码即可

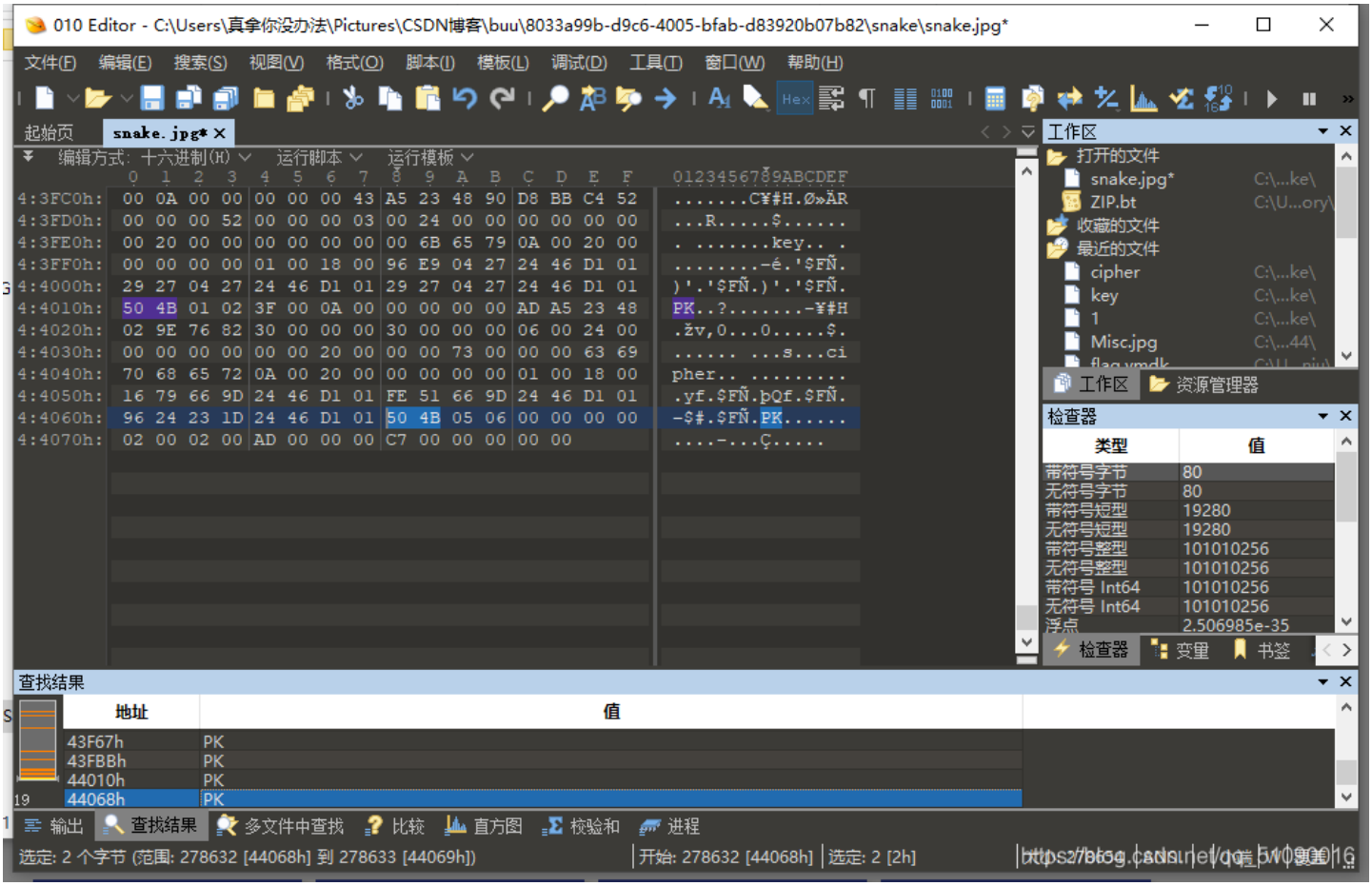


解压得到的txt文件

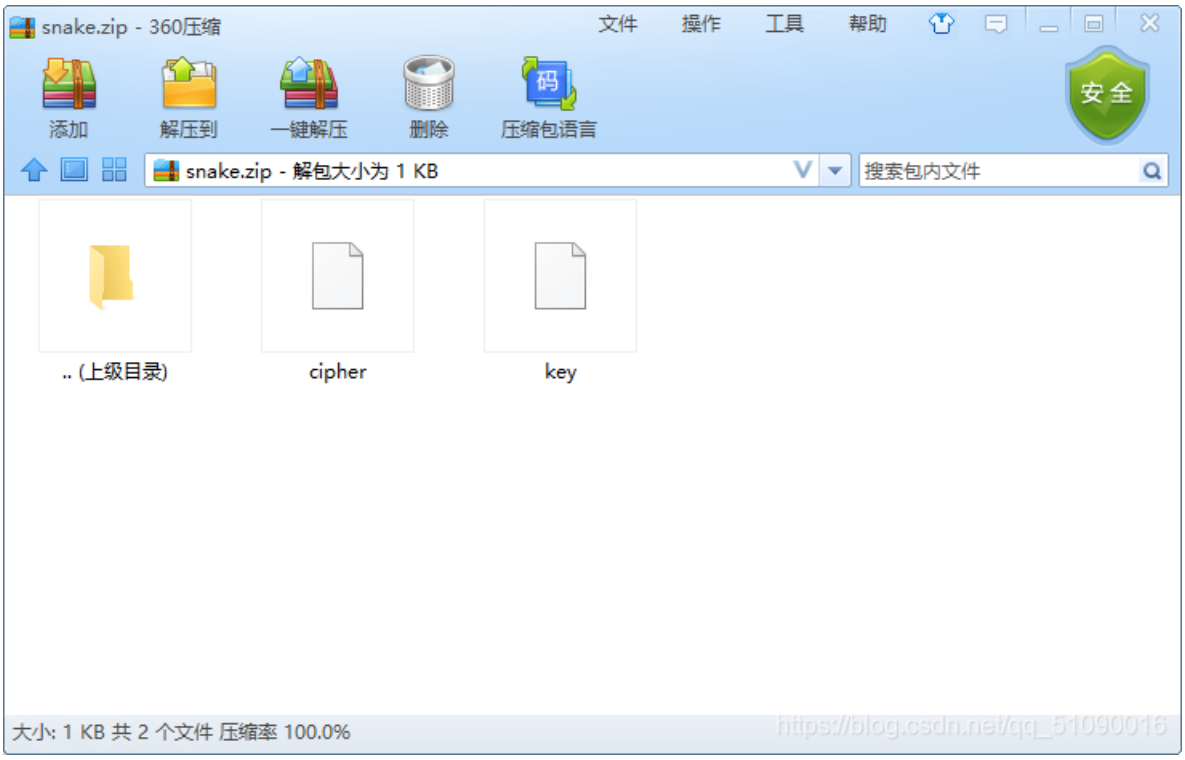


snake（serpent解密）

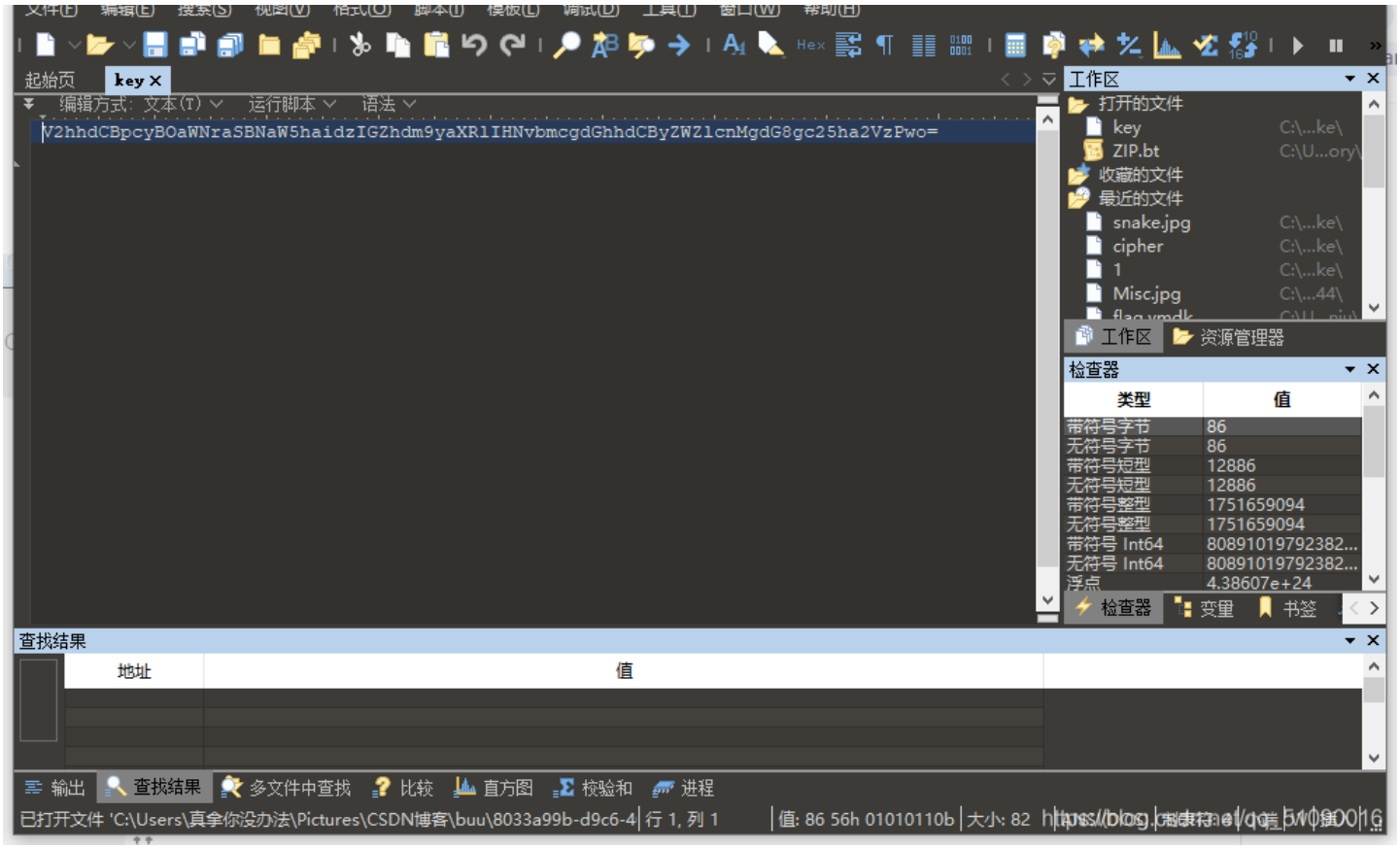
附件用010打开看看，搜索下flag txt pk等等



里面应该是有压缩文件，改成zip看看

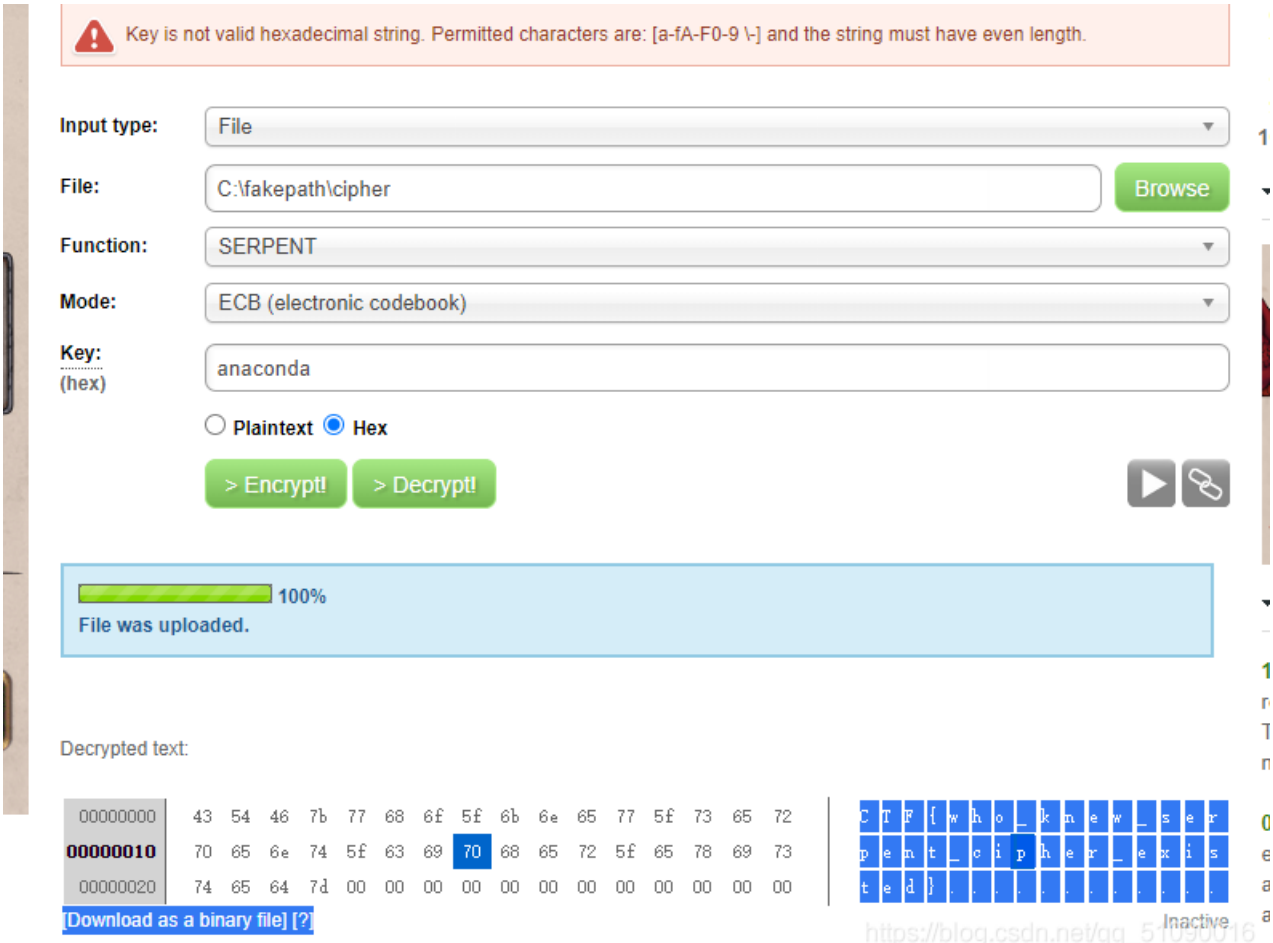


果然有，先看key



base64解密：

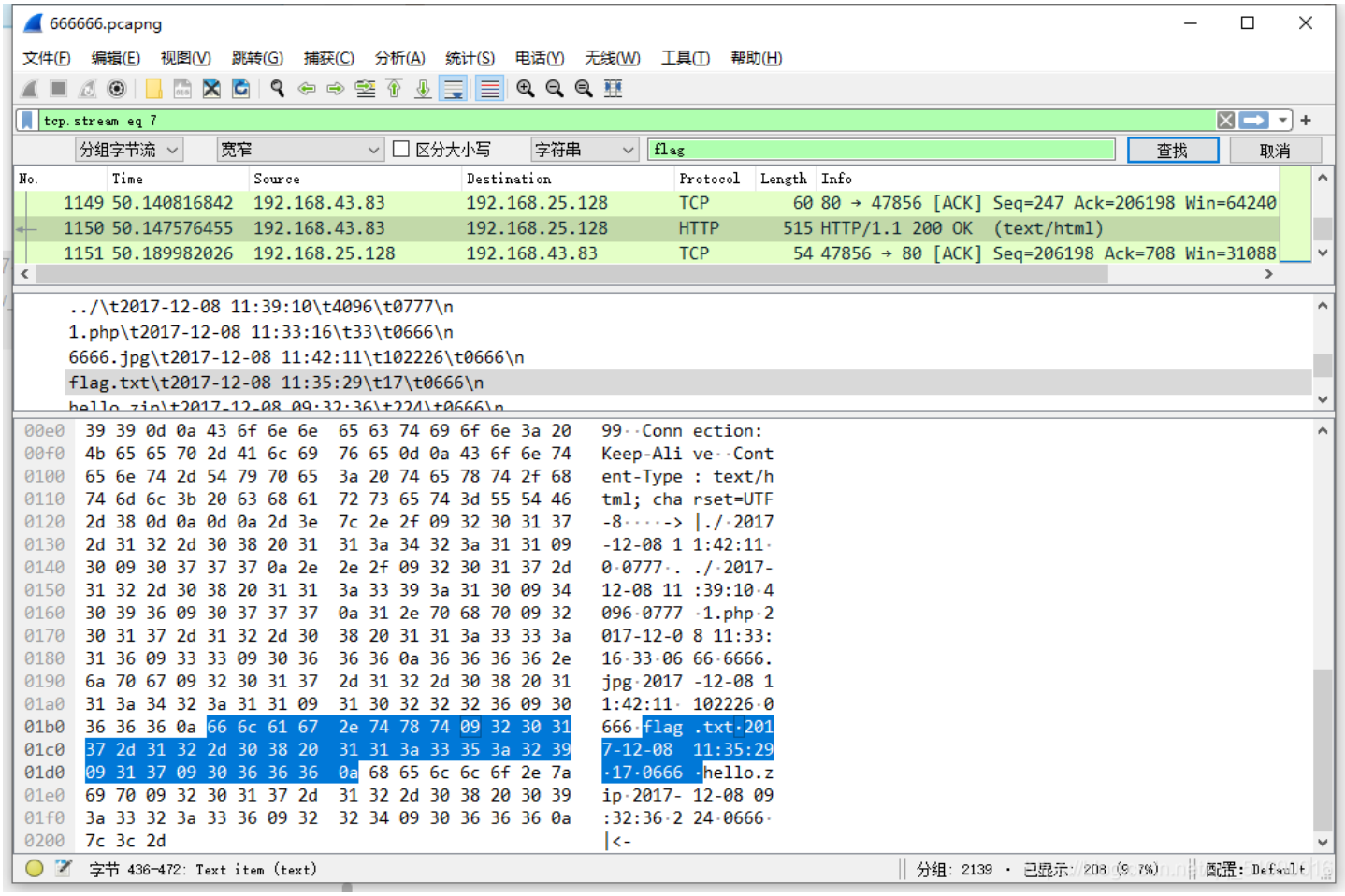
这我哪知道，百度一下是anaconda，然后就是解密了，是什么解密呢？题目是snake，解密应该和这个有关，搜了一下才知道还有serpent解密，网址贴这

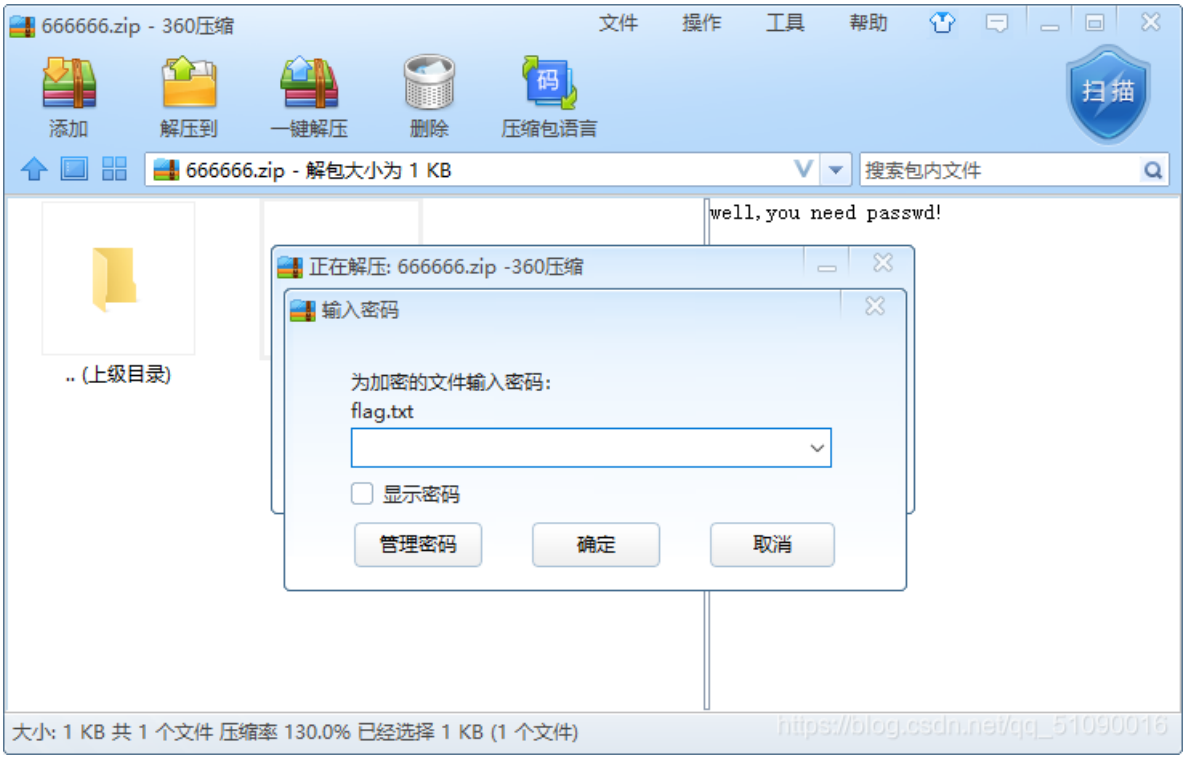


这样就有了

菜刀666（流量分析）

首先是很常规的套路，用winshark打开，搜索flag，发现有个txt文件藏在里面

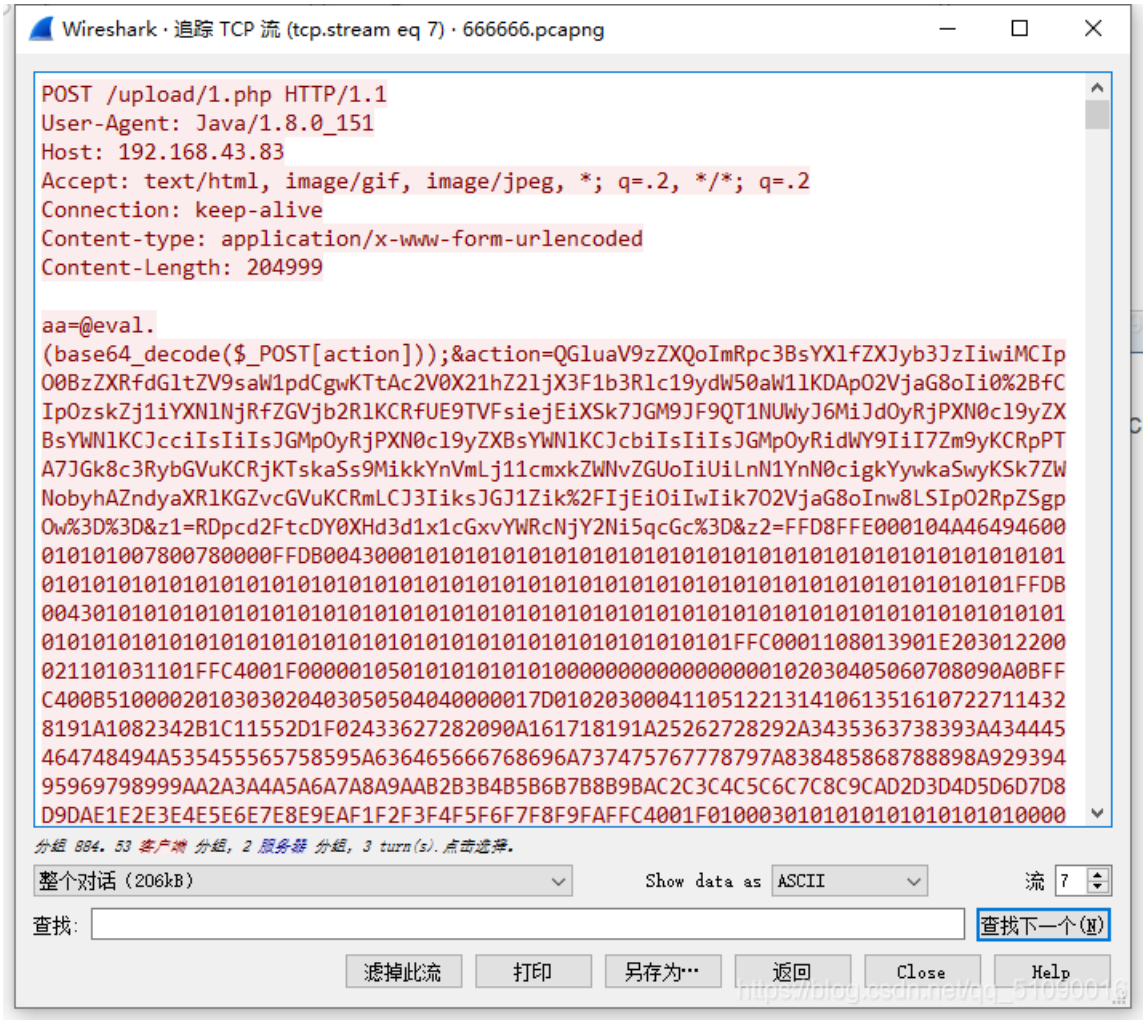




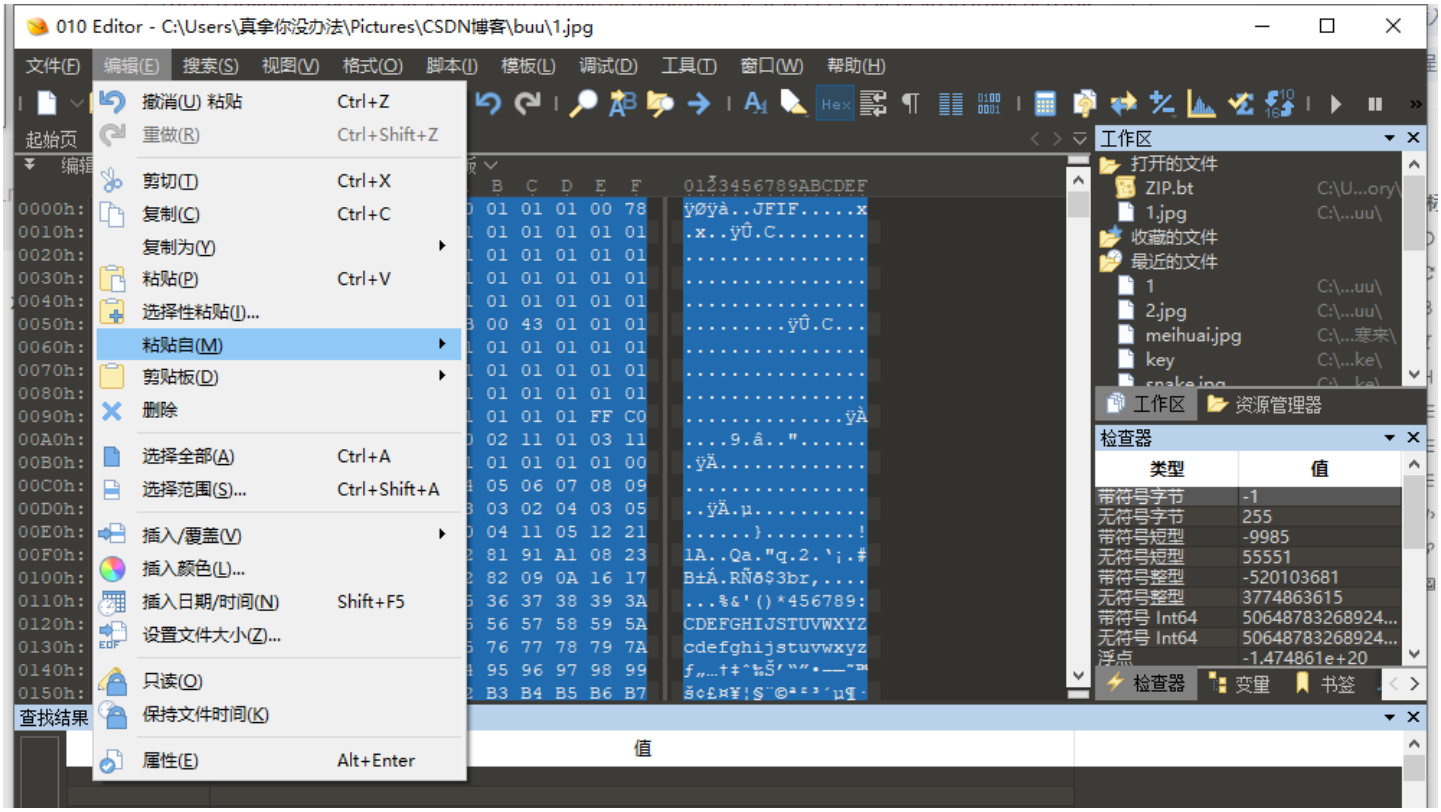
后缀改成zip打开看看

需要密码，这时候就要想到从本身的文件入手了，搜素一下password。。。没有，那tcp追踪流看看

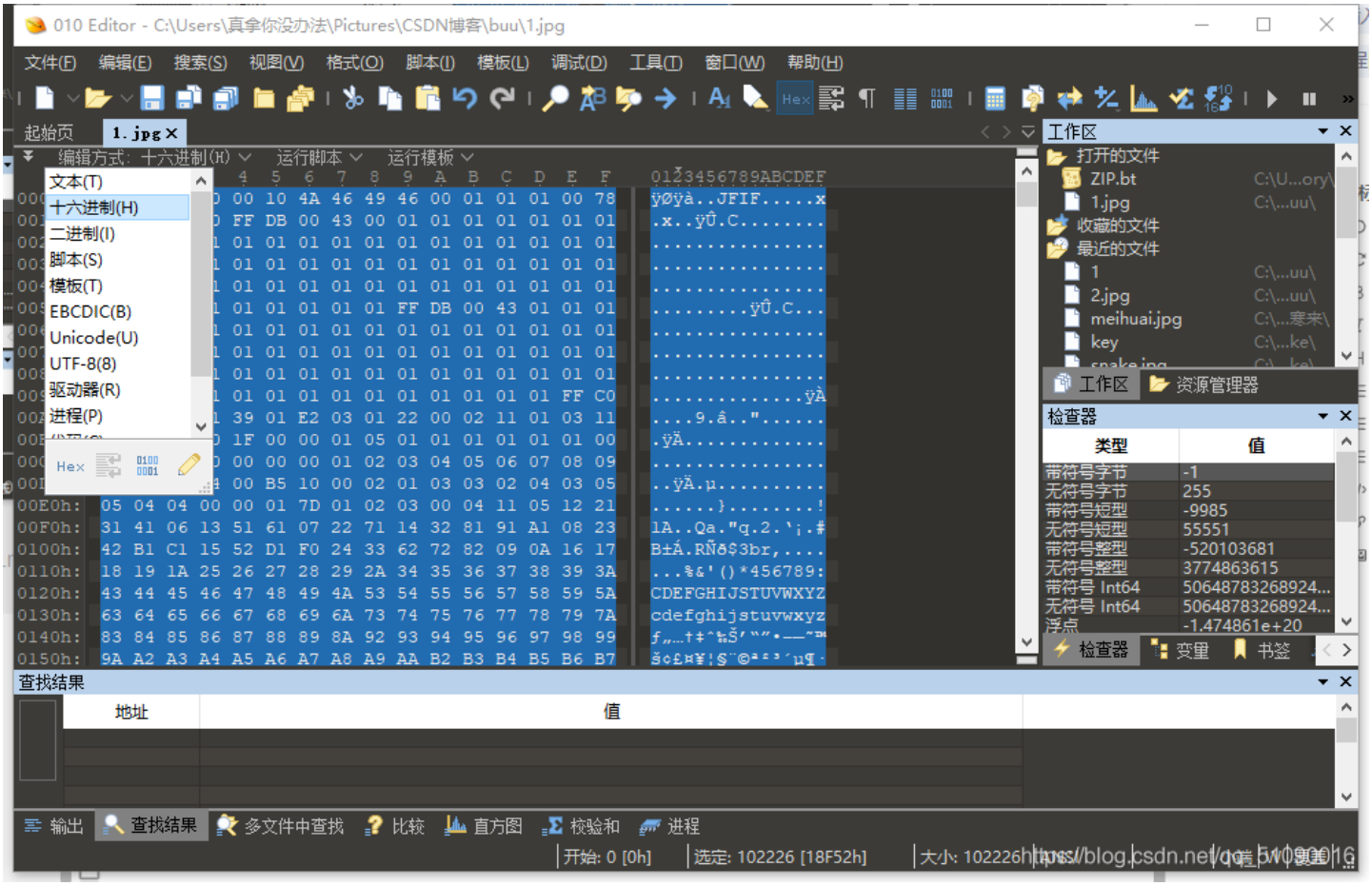
看到7流的时候发现一串16进制字符



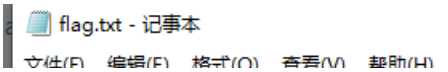
又看到有个z1 z2, z1我还以为是base，结果不是，先看z2吧 FF D8开头FF D9结尾，判断为jpg图片，将这些十六进制复制出来，粘贴到010里面，注意具体操作如图，要粘贴自16进制文件



文件类型也是16进制：



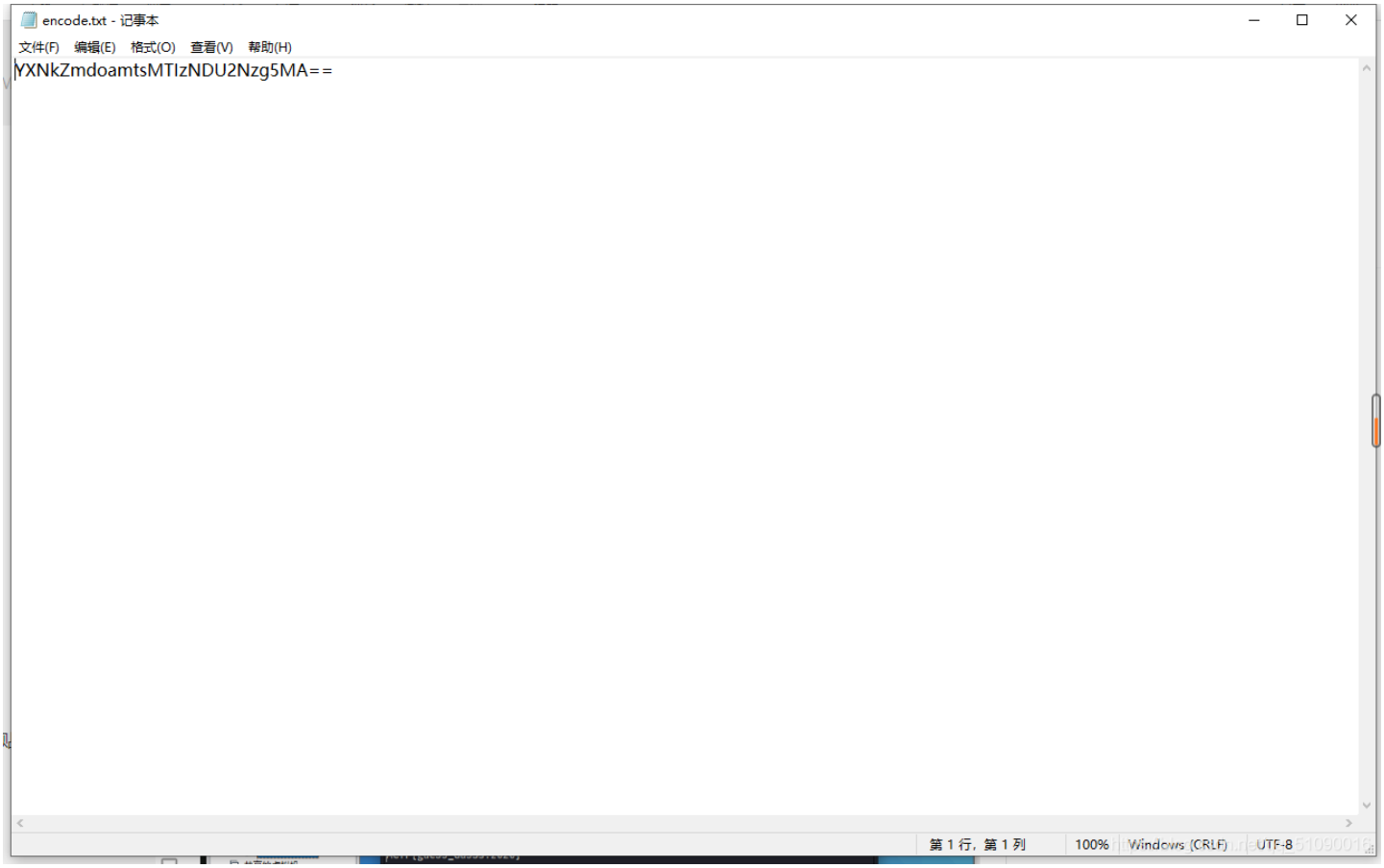
密码有了



flag{3OpWdJ-JP6FzK-koCMAK-VkfWBq-75Un2z}

[SWPU2019]神奇的二维码（binwalk -e分离）

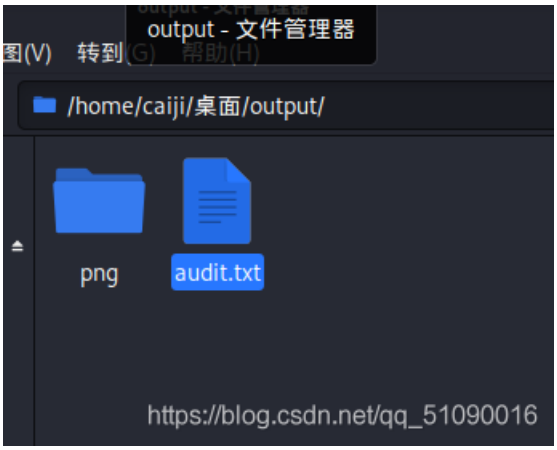
正常010打开，发现有rar文件，先试试改成rar文件，解压缩得到



解码得到

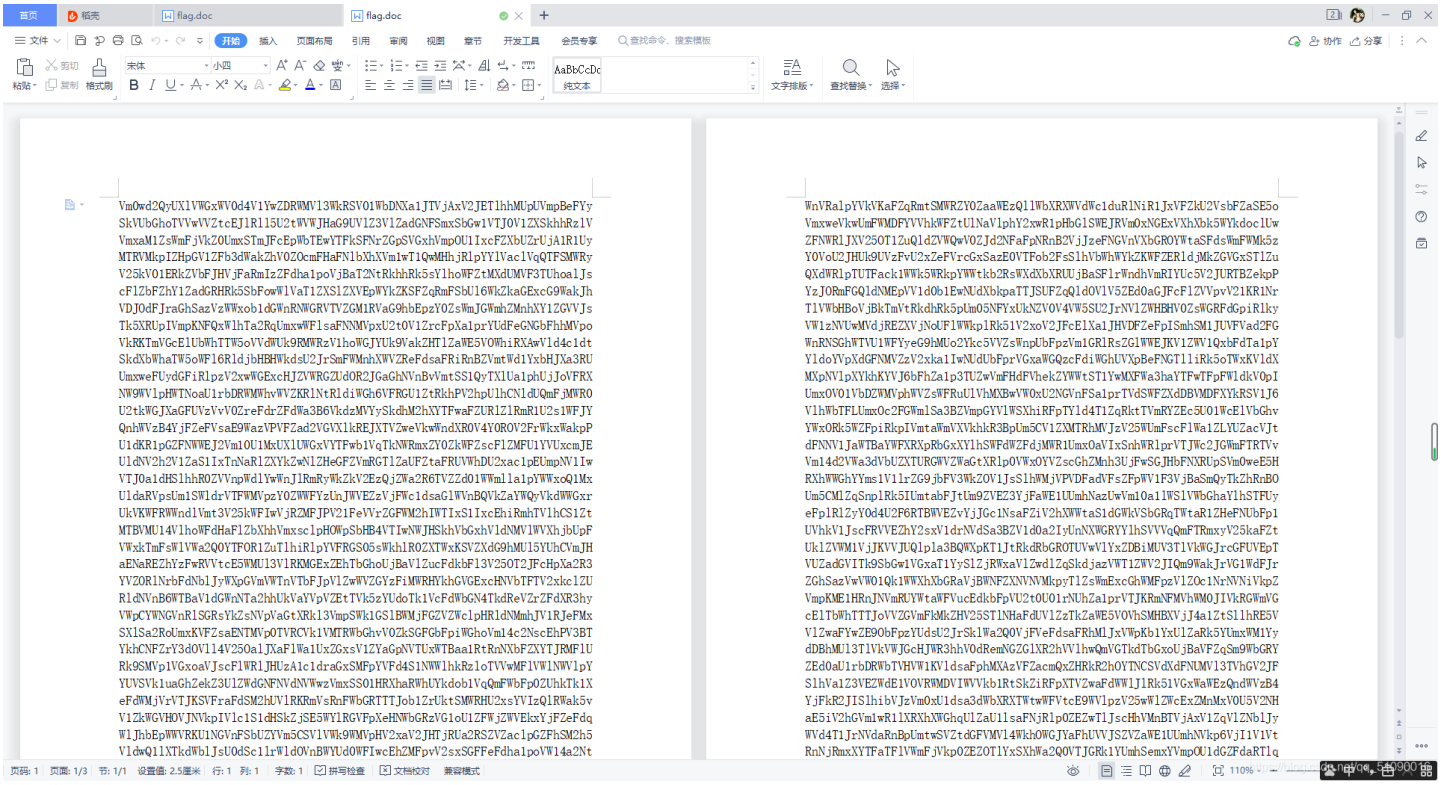
还以为这就是flag。。没思路了，直接看wp

放到虚拟机里binwalk一下 再foremost一下

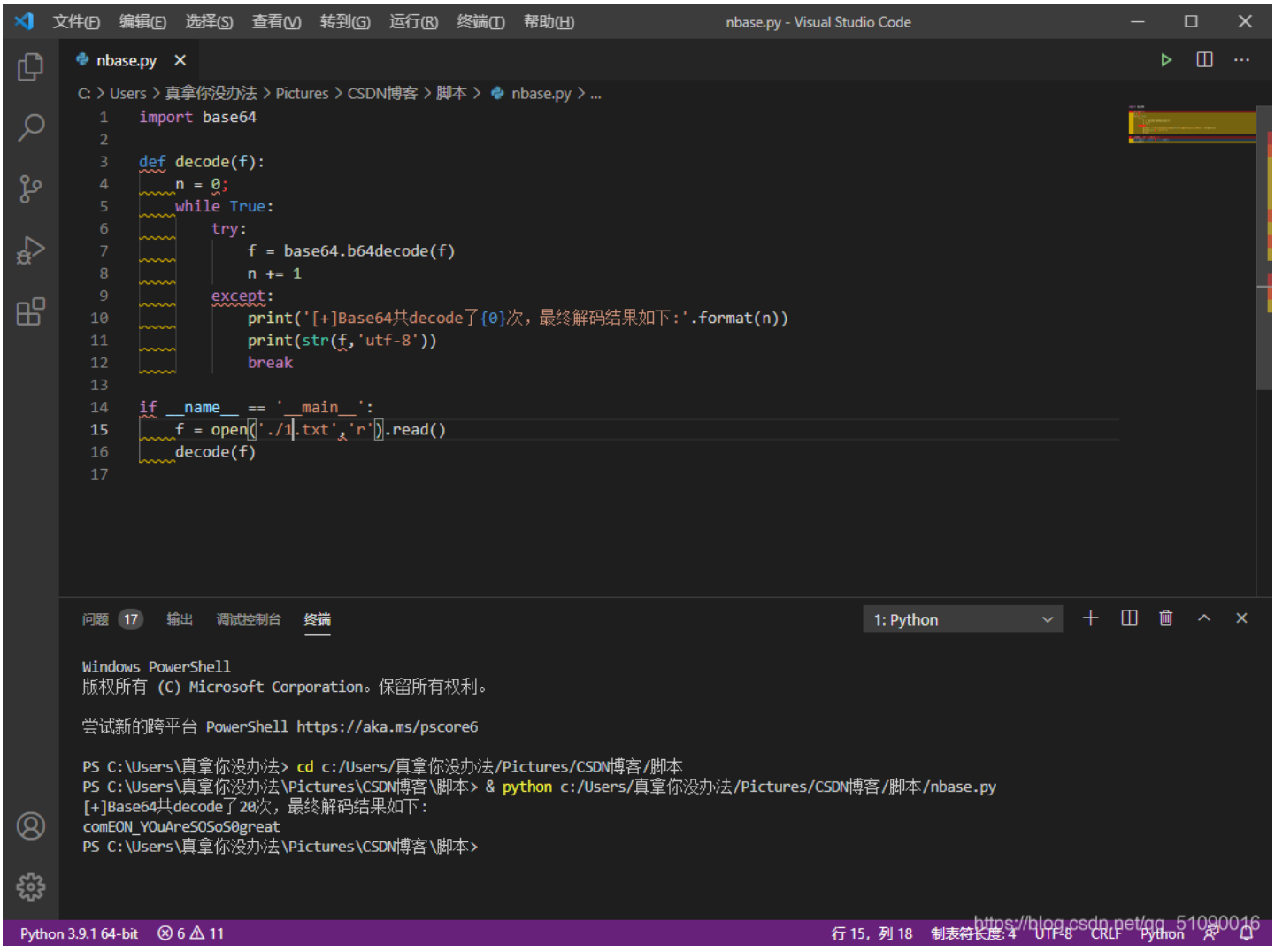


用binwalk -e试一下

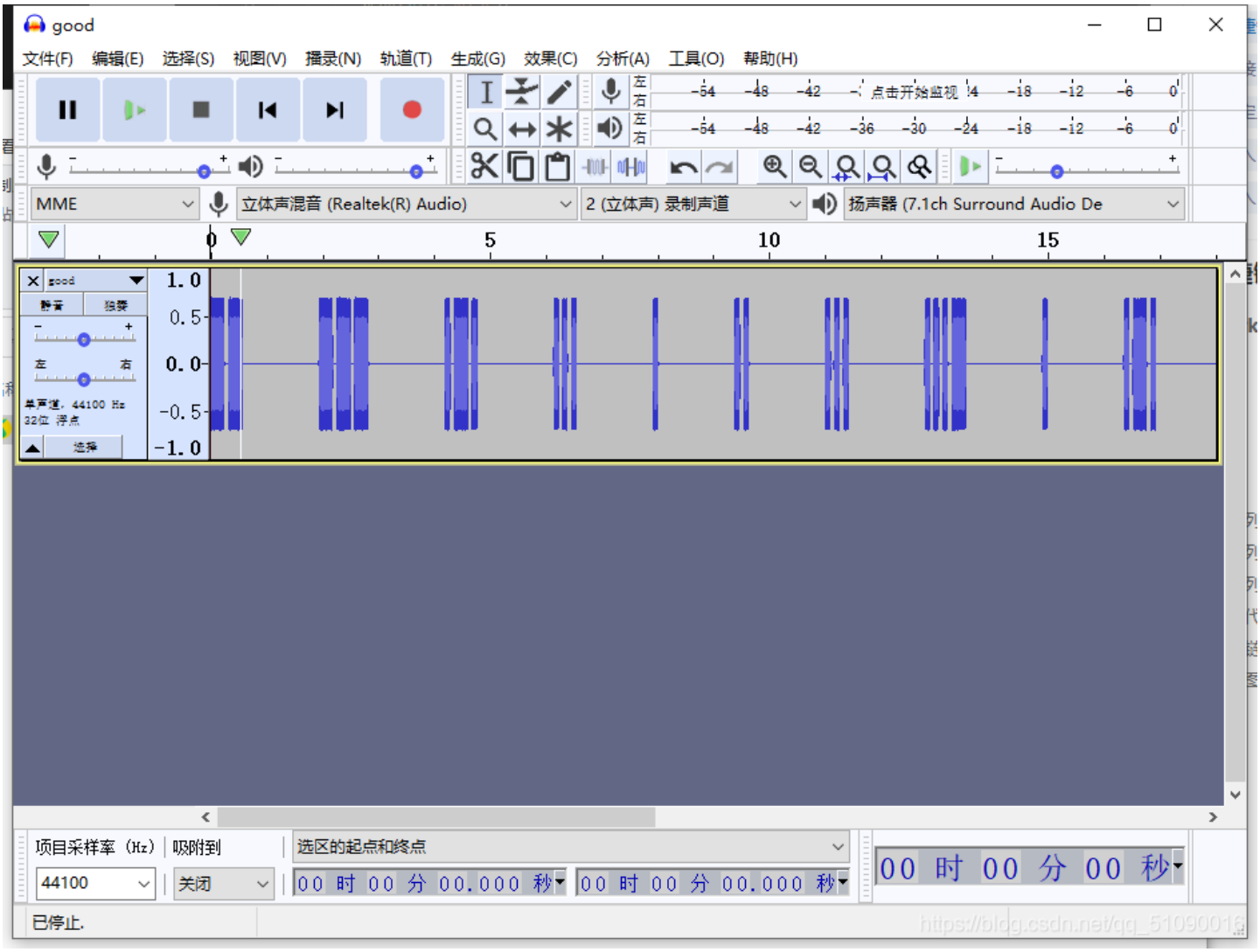
分离出了四个压缩包，发现有一个需要密码。。试了下发现其实asdfghjkl1234567890是看看flag在不在里面–.rar的密码
其实这里面什么都没有，就是单纯的图片，什么都没有隐写…浪费我时间…



一大串。。应该被编码了很多次，用脚本



解压得到音频文件，使用Audacity打开



摩斯电码

– — .-. … . … … …- . .-. -.-- …- . .-. -.-- . .- … -.–

英文字母:

MORSEISVERYVERYEASY

转换为摩斯电码 清除 生成摩斯代码的分隔方式: 空格分隔 单斜杠/分隔

摩斯电码: (格式要求：可用空格或单斜杠/来分隔摩斯电码，但只可用一种，不可混用)

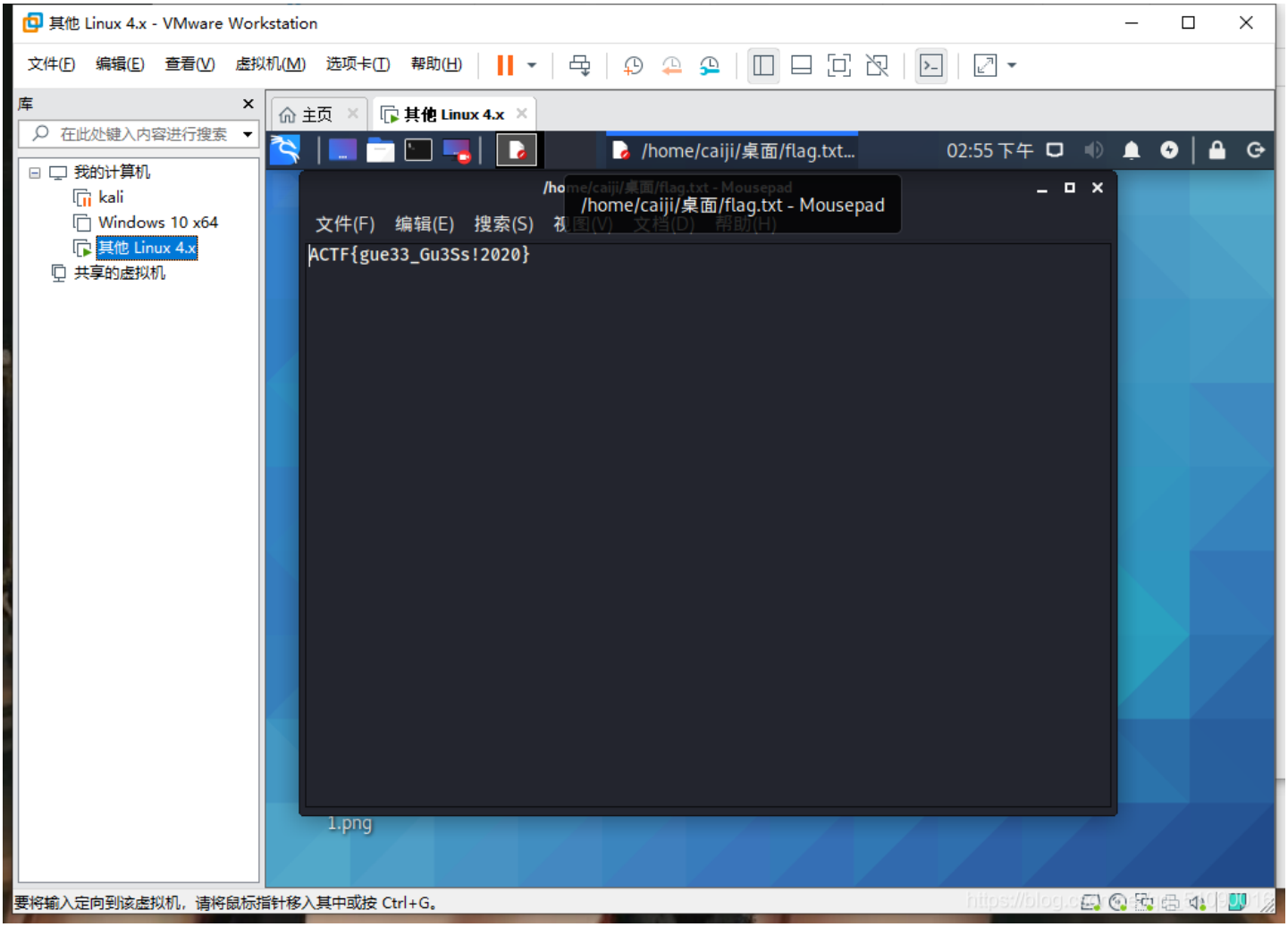
-- --- .-. … . .. … …- . .-. -.-- …- . .-. -.-- . .- … -.--

[ACTF新生赛2020]outguess（outguess的用法）

解压后发现一堆东西，重点应该还是在图片上，根据提示猜测是outguess的使用，用法贴这里

kali下面输入outguess -k ‘abc’ -r mmm.jpg flag.txt

得到flag.txt

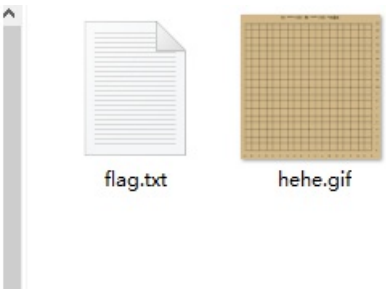


谁赢了比赛？

正常流程，丢进kali

```
root@kali:~/Desktop# binwalk who_won_the_game.png

DECIMAL       HEXADECIMAL     DESCRIPTION
--------------------------------------------------------------------------------
0             0x0             PNG image, 10000 x 10000, 8-bit
erlaced
1430236       0x15D2DC        RAR archive data, version 4.x,
MAIN_HEAD
```

binwalk -e 分离获得压缩包，爆破得到

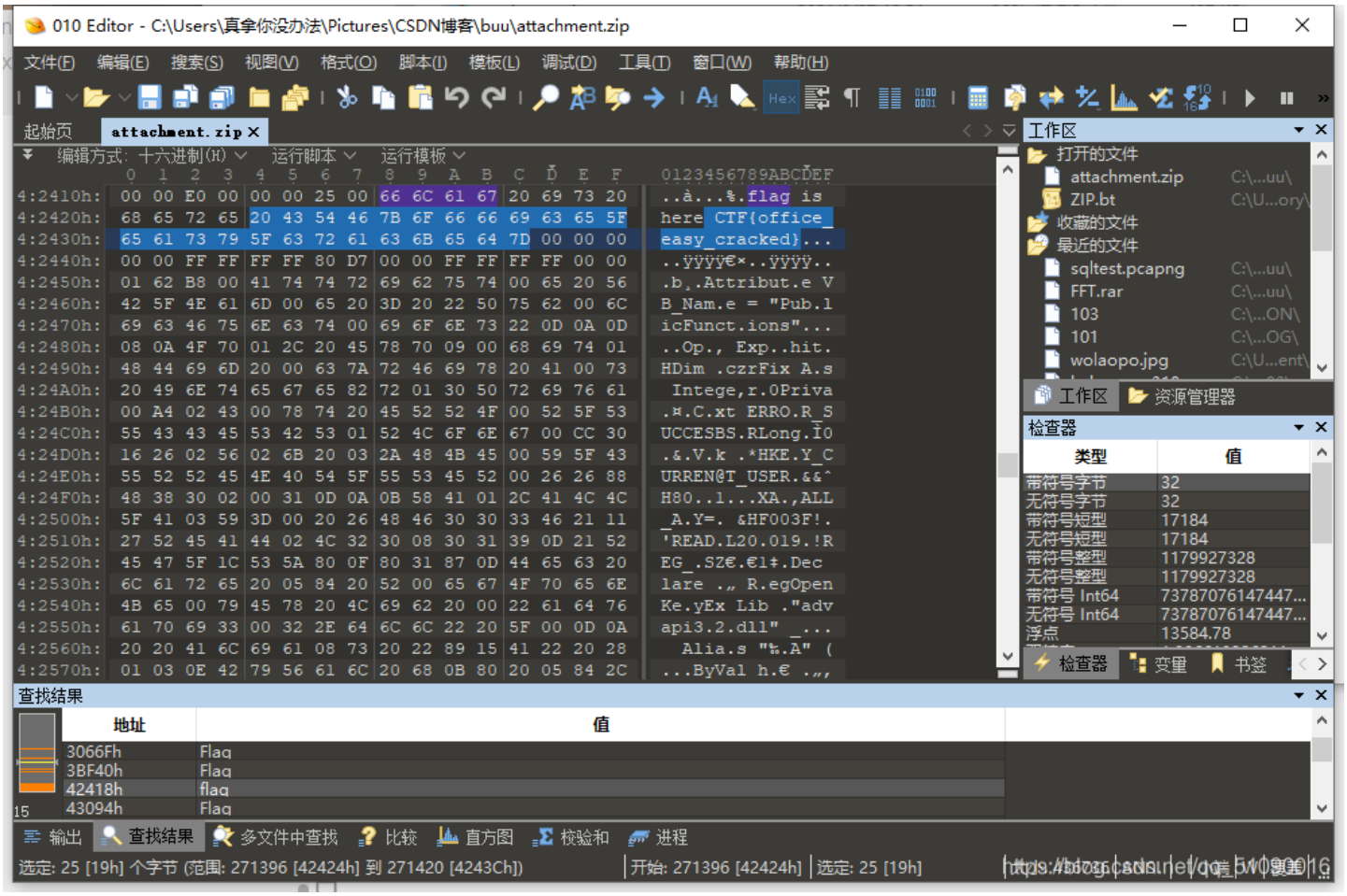


flag.txt假的。。。
分离gif吧：一直看到三百多张发现了点东西

stegsolve打开。。换通道，终于有了

[HBNIS2018]excel破解（修改后缀）

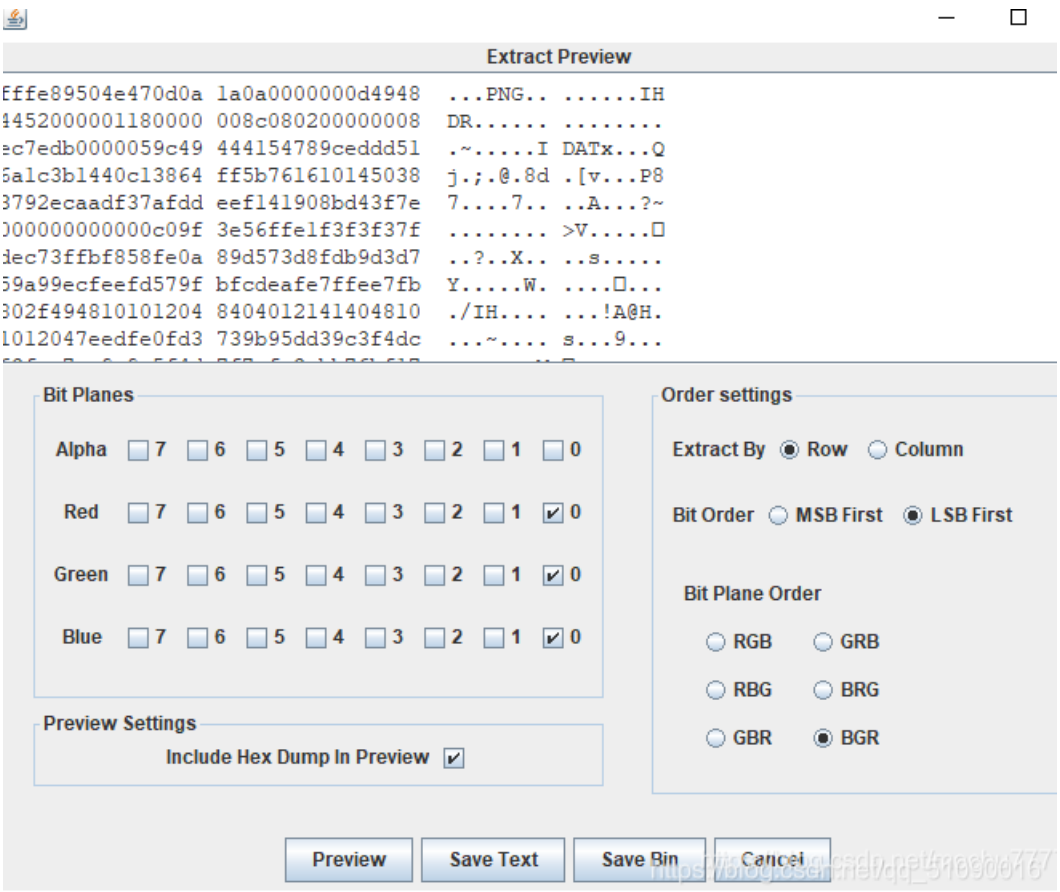
修改zip后缀，然后用010打开。。。就有了？？



喵喵喵（奇怪的lsb NTFS文件流隐写）

这题感觉好难

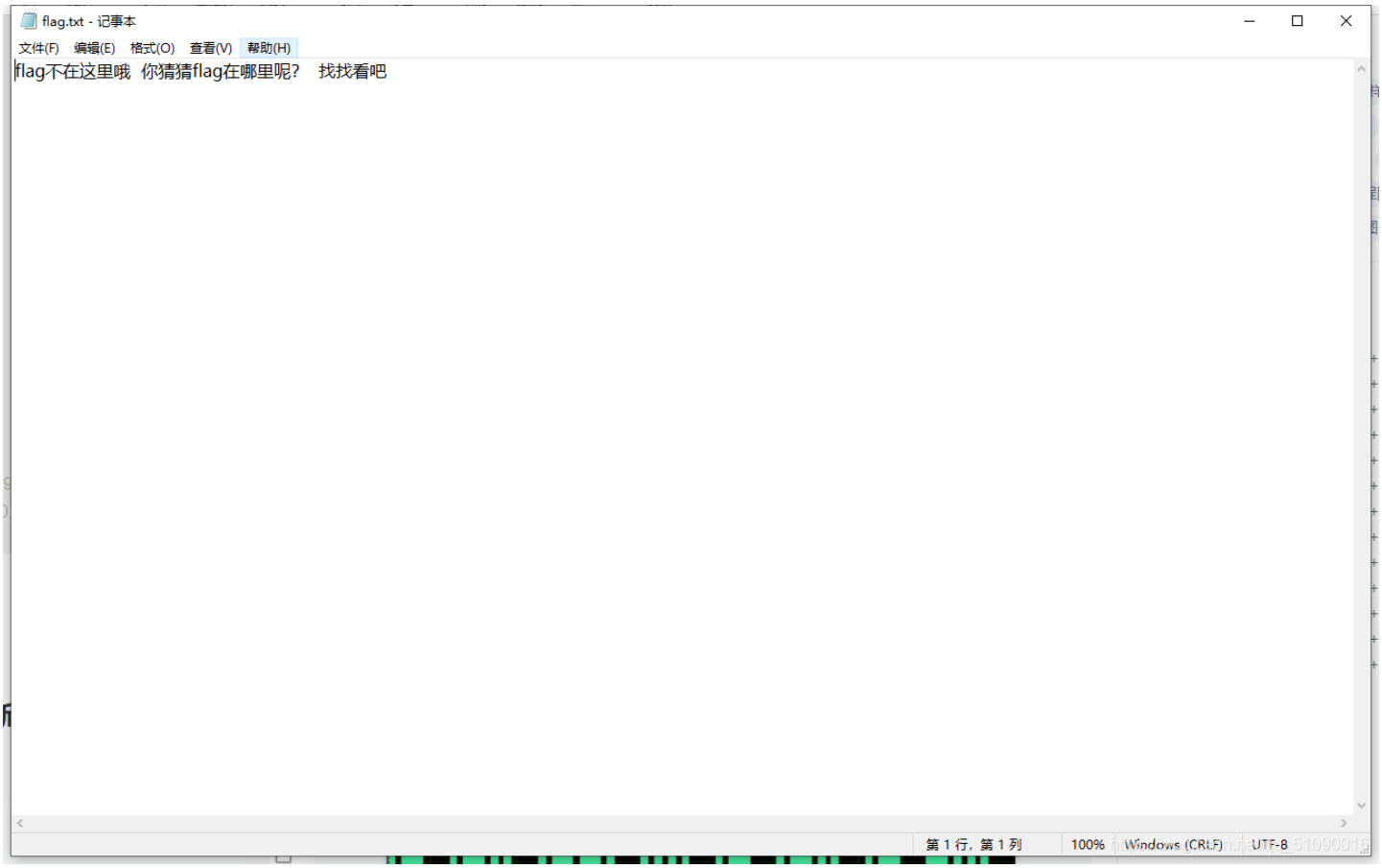
题目提示有二维码，但是010打开并没有发现隐藏压缩包，应该想到lsb隐写
但是谁能想到要设置成这样才行？？



另存为png文件发现打不开。。改一下文件头就好了

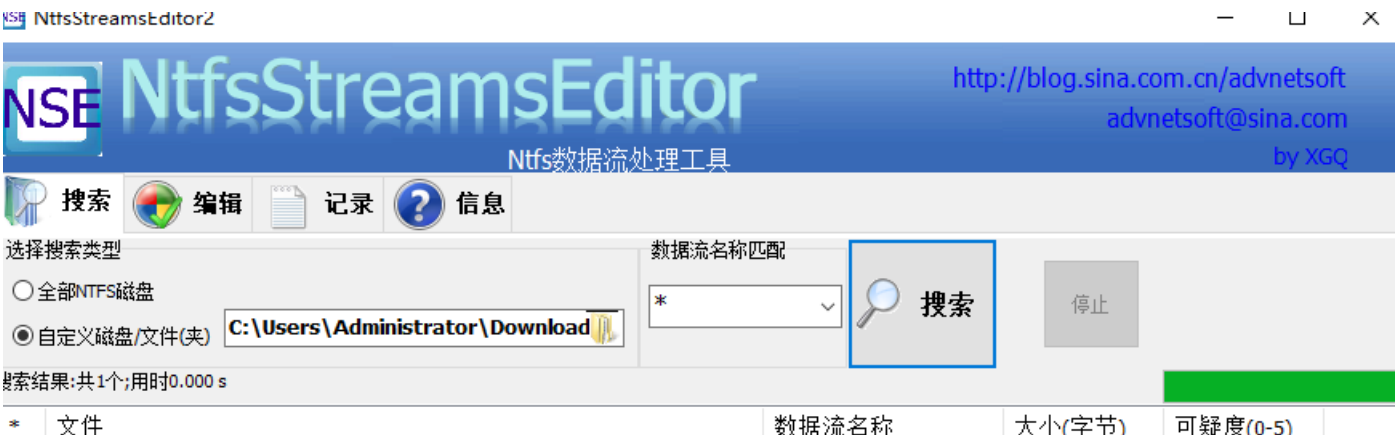
这明显是高的问题了，加个高吧

扫了之后是个网盘地址。。下载flag.rar打开



？？？看wp

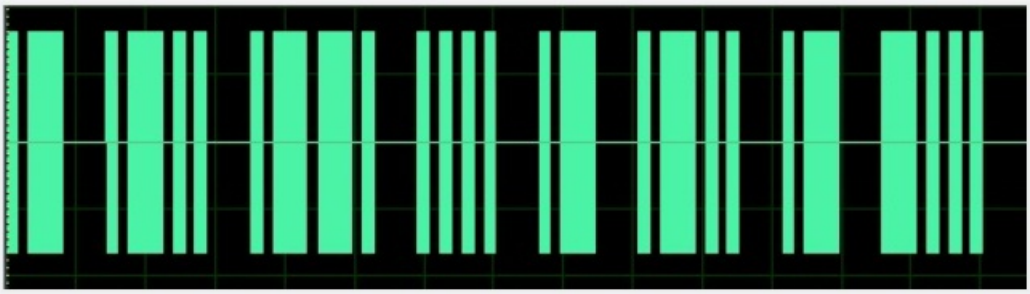
这里猜测有NTFS文件流隐写，将flag.txt解压到一个新建的文件夹内，利用NtfsStreamsEditor



☐	C:\Users\Administrator\Downloads\新建文件夹\flag.txt:flag.pyc	flag.pyc	755	1

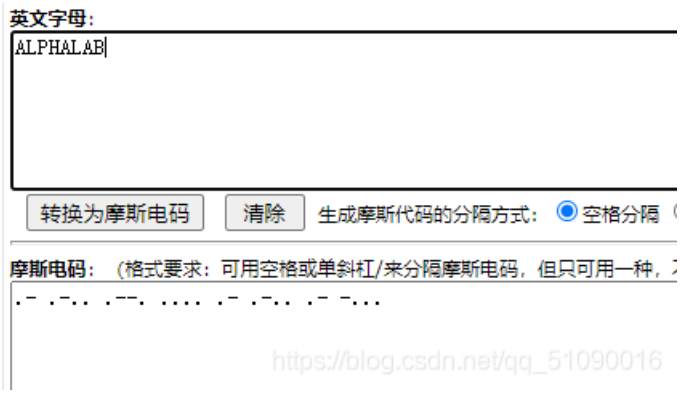
导出就有了

[HBNIS2018]来题中等的吧（条形码表示摩斯电码）



开始我以为是条形码，扫了半天没扫出啥。。。看wp，原来是摩斯电码，依次为:

```
.- .-.. .--. .... .- .-.. .- -...
```



改成小写字母就行了

弱口令（这也能藏莫斯代码？ 没遇到过的lsb隐写）

先是在解压缩的右边有提示，复制粘贴到vs code，会发现隐藏着摩斯电码，翻译完是

```
HELL0FORUM
```

这样就能解压图片了，然后是奇怪的lsb隐写，需要用到cloacked-pixel。。但是我的脚本一直报错，很烦

[SWPU2019]你有没有好好看网课？（敲击码）

先是爆破zip3

得到：

从小 5 就 20 列文虎克，

我每年的 7 月 11 日的生日愿望就是拥有一个

提示 520 711

使用Kinovea打开影流之主.mp4

敲击码(Tap code)是一种以非常简单的方式对文本信息进行编码的方法。因该编码对信息通过使用一系列的点击声音来编码而命名，

敲击码是基于5×5方格波利比奥斯方阵来实现的，不同点是是用K字母被整合到C中。

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
```

Bash

在第7.36s发现第二段信息：dXBfdXBfdXA=

base解码 `up_up_up`

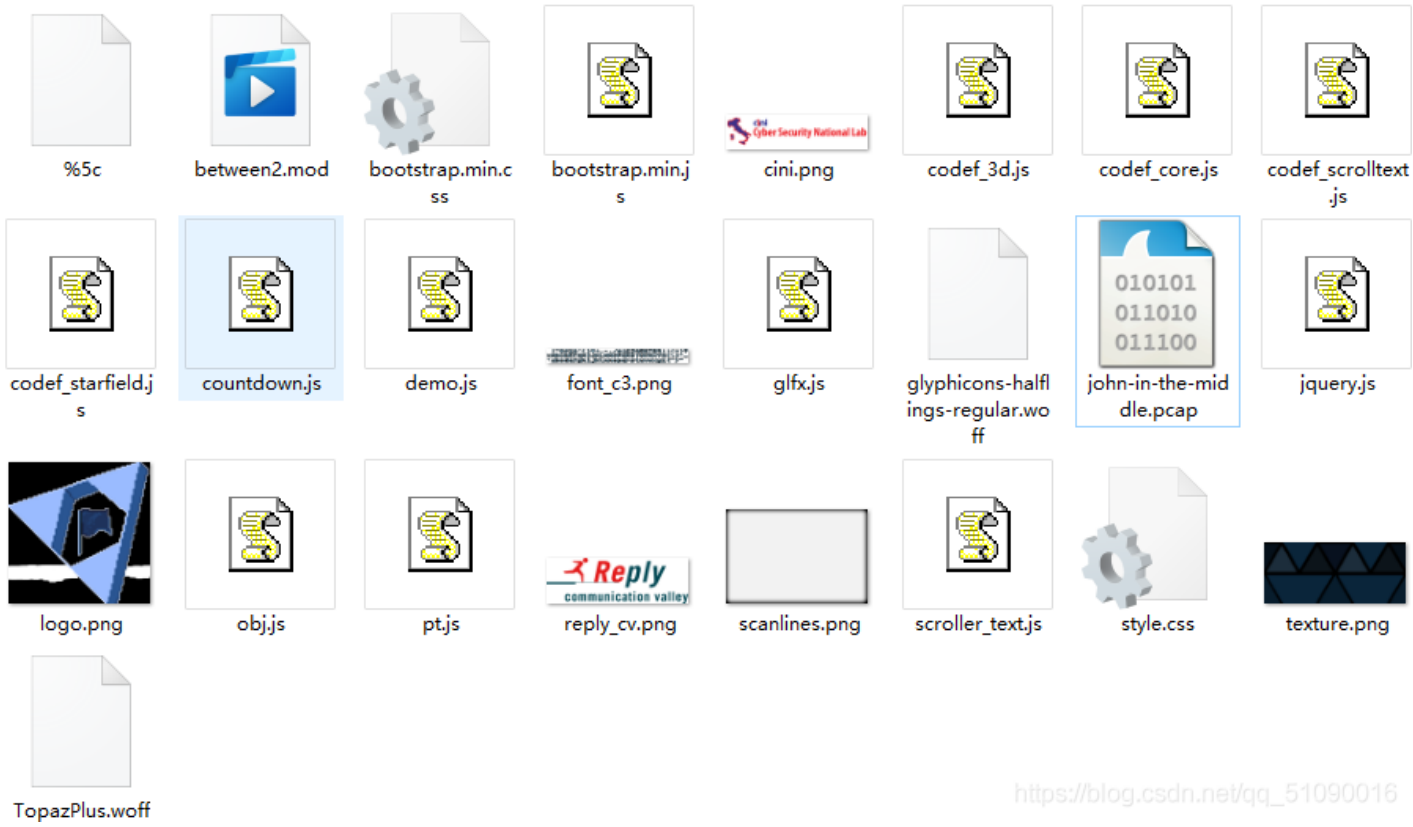
合起来：`wllmup_up_up`

使用这个密码解压flag2.zip，解压得到Real flag.jpg，使用010 Ediotr打开搜索关键词ctf即可得到flag

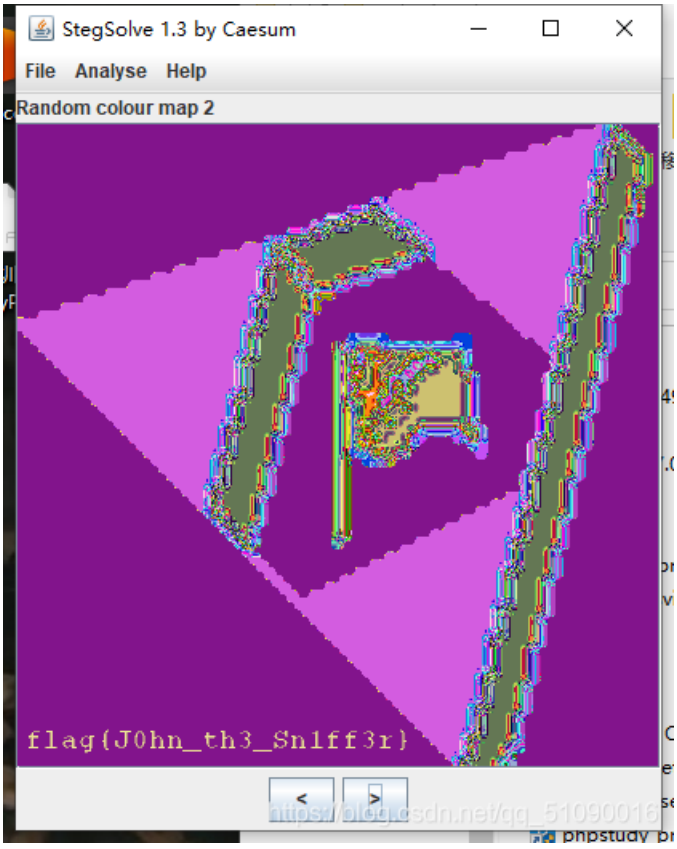
john-in-the-middle(导出http)

wireshark打开文件后，查看数据量并没有线索，然后我就麻了，看wp

需要导出http，具体步骤：文件 →导出对象→http →save all



得到了好多东西，从6张png下手，一个个在stegsolve中打开，终于在logo.png里面找到了（好狗）



低个头（键盘加密？？）

attachment.txt - 记事本

文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)

题目：低个头

描述：EWAZX RTY TGB IJN IO KL 请破解该密文 f

lag格式：XXX 明文

提交：直接提交明文（大写）

用键盘连线组成字母

EWAZX	构成C
RTY+TGB	构成T
IJN+IO+KL	构成F

那就是 `flag{CTF}`

zip

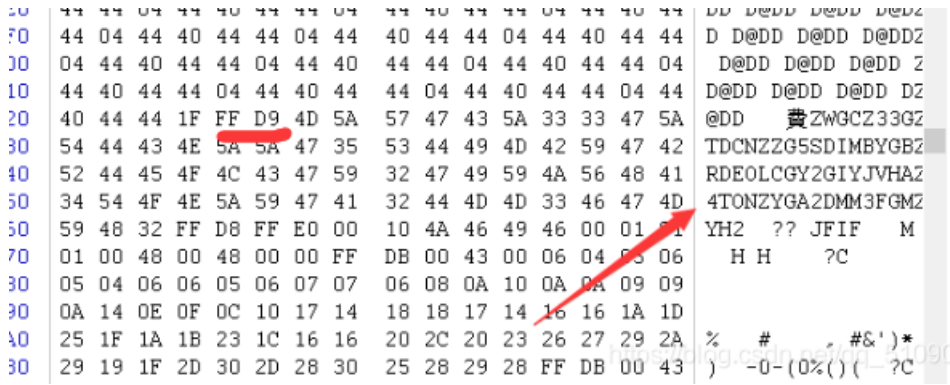
参考大佬的wp： https://blog.csdn.net/destiny1507/article/details/102133268/

不过那个暴破脚本为什么运行的那么慢。。。害的我以为出错了试了好多次

我吃三明治(两张图拼接之间隐藏信息)

binwalk的时候发现这张图片是由两张图片拼接起来的：

```
root@mochu7:~/Desktop# ls
firefox-esr.desktop  flag.jpg
root@mochu7:~/Desktop# binwalk flag.jpg

DECIMAL       HEXADECIMAL     DESCRIPTION
--------------------------------------------------------------------------------
0             0x0             JPEG image data, JFIF standard 1.01
37475         0x9263          JPEG image data, JFIF standard 1.01
```



找到中间点：搜索jpg文件尾ffd9

看到一串base加密，base64不行，后面又没等号，试试base32：

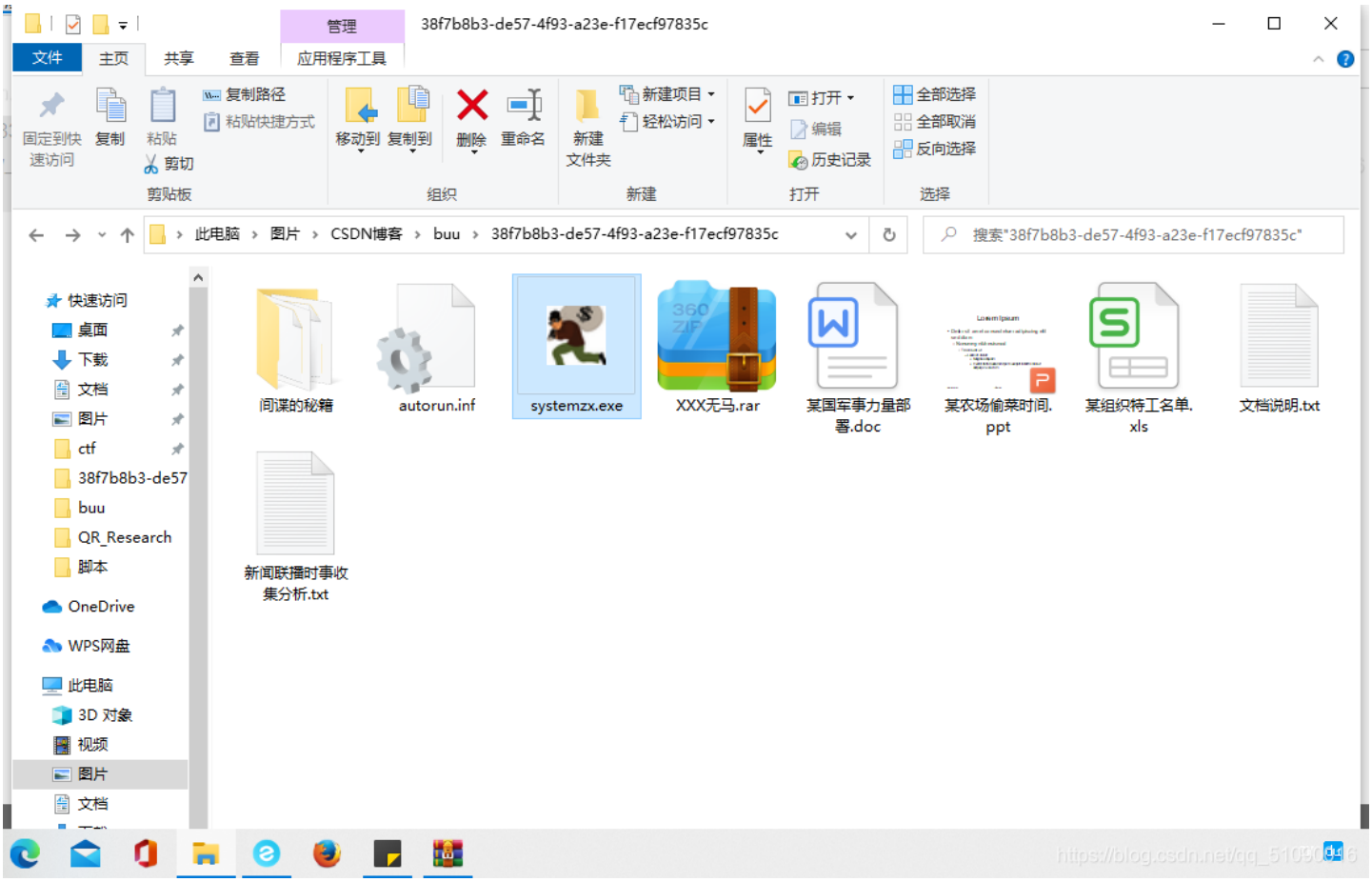
MZWGCZ33GZTDCNZZG5SDIMBYGBRDEOLCGY2GIYJVHA4TONZYGA2DMM3FGMYH2

编码 解码 清空

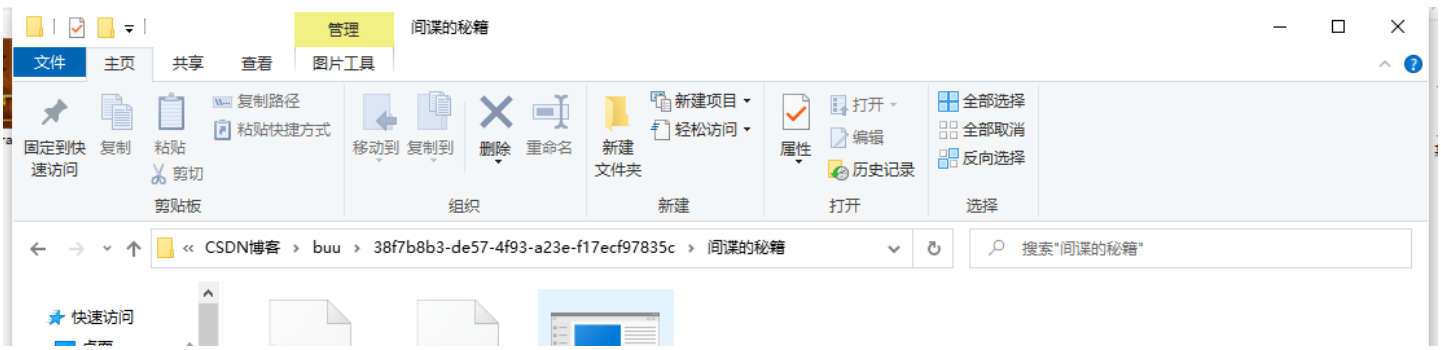
flag{6f1797d4080b29b64da5897780463e30}

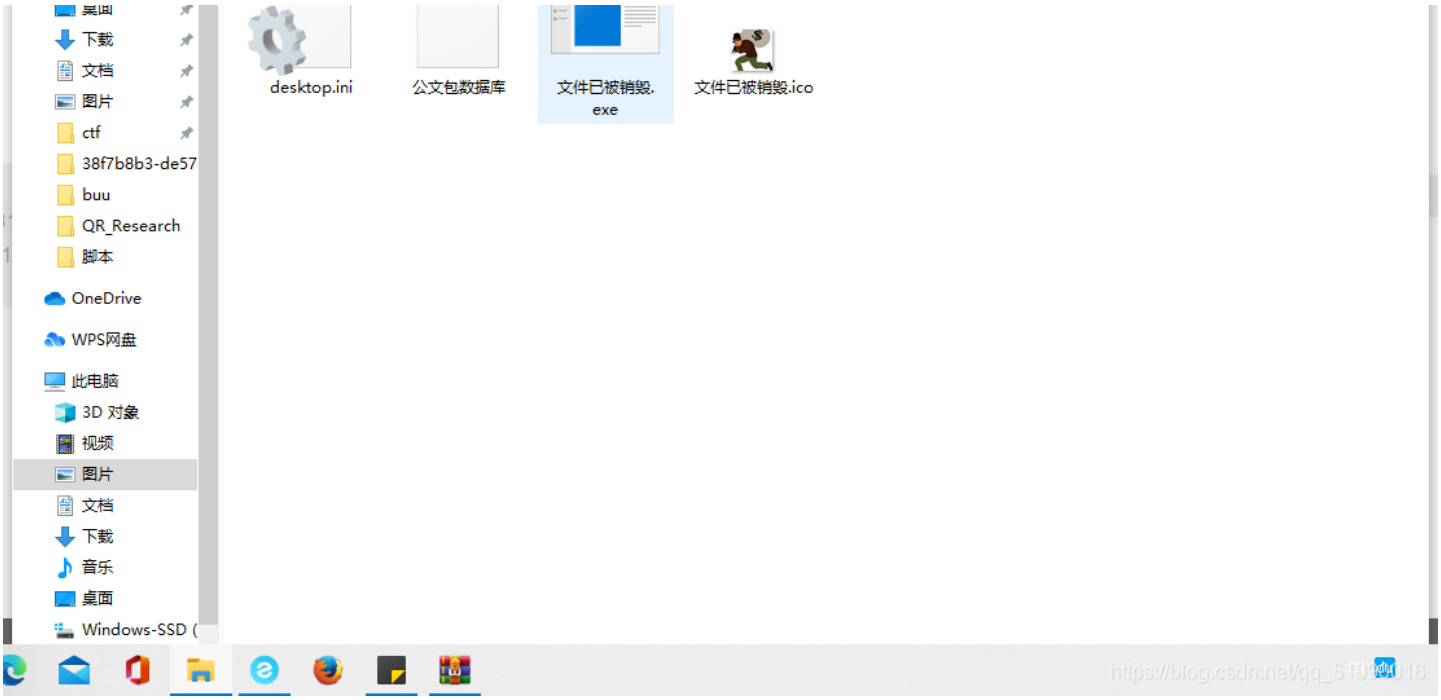
间谍启示录（奇怪的题目）

解压后看到一堆文件，只有这个exe有点东西



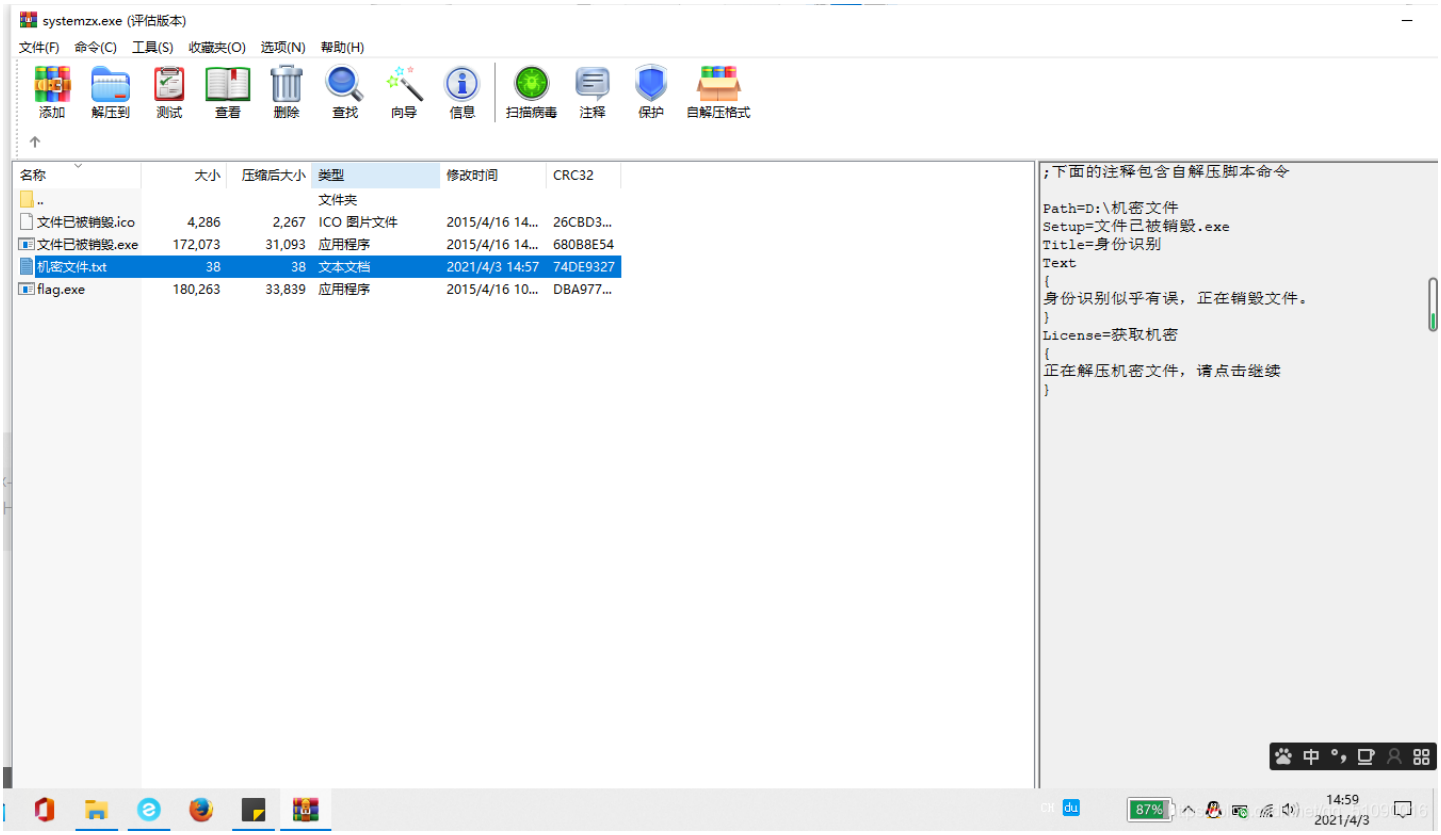
直接点开就被销毁了





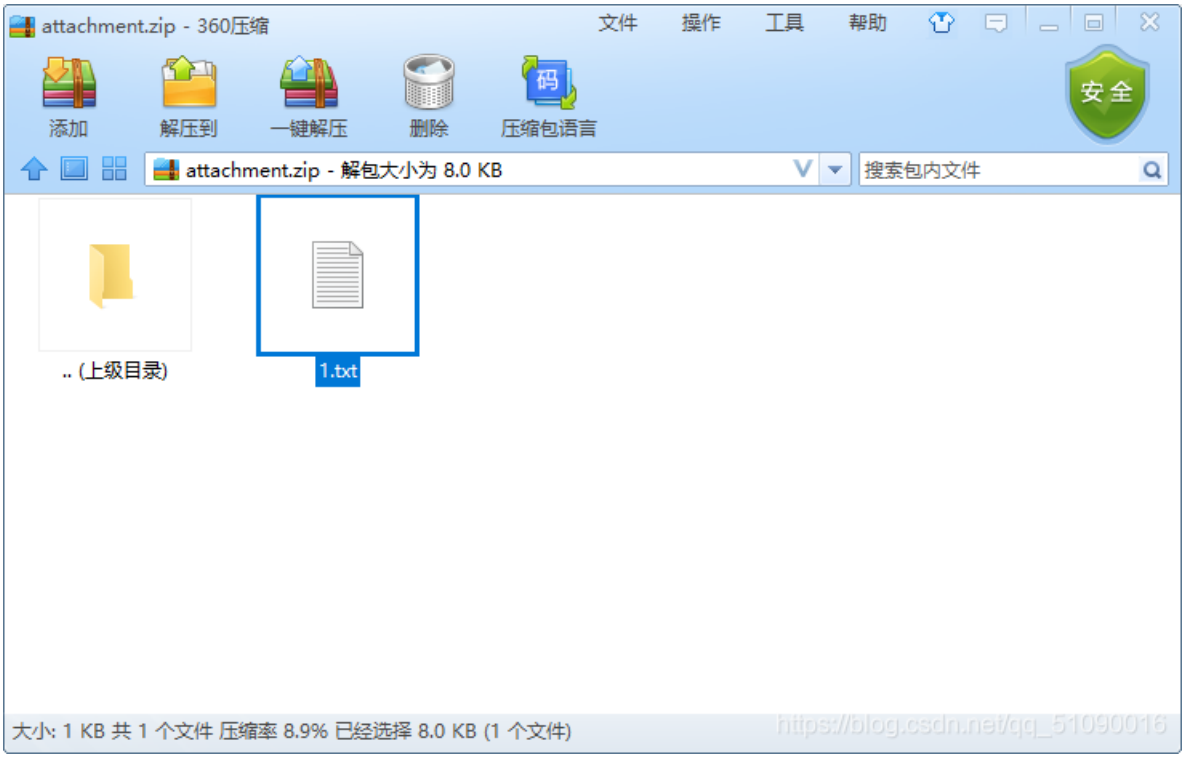
想到解压或者foremost也行

点那个flag.exe就会生成机密文件



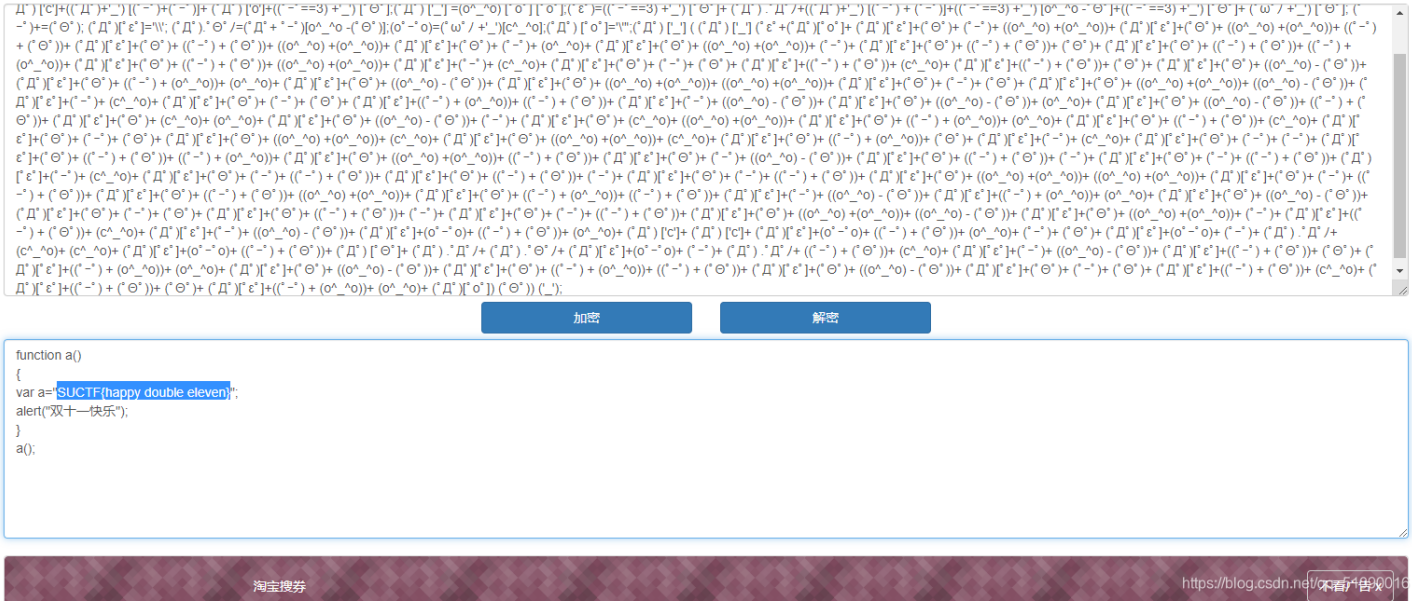
[SUCTF2018]single dog(AAEncode解密)

用010打开发现里面有其他文件，改成zip后缀打开



里面是一大段表情。。。这就涉及到我的知识盲区了，百度吧

AAEncode加密/解密

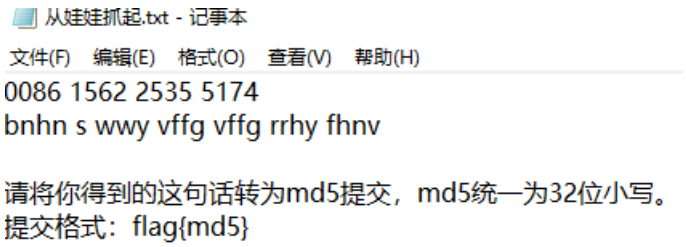


[安洵杯 2019]吹着贝斯扫二维码

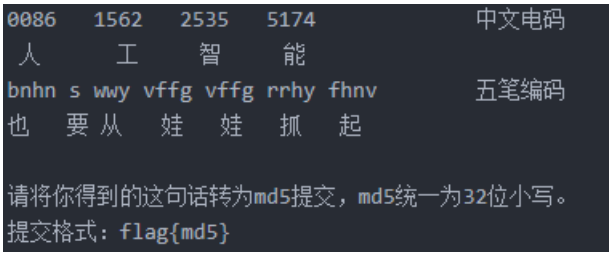
这题需要用到改后缀的脚本，但我运行一直出错啊啊啊啊啊啊。。。

从娃娃抓起（中文电码）

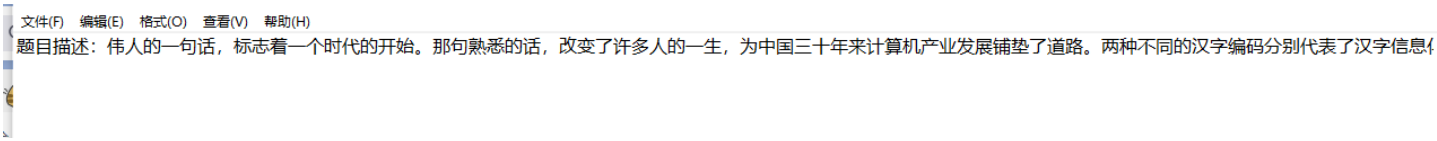
两个txt文件，第一个：



中文电码和五笔编码



第二个是提示



md5加密：

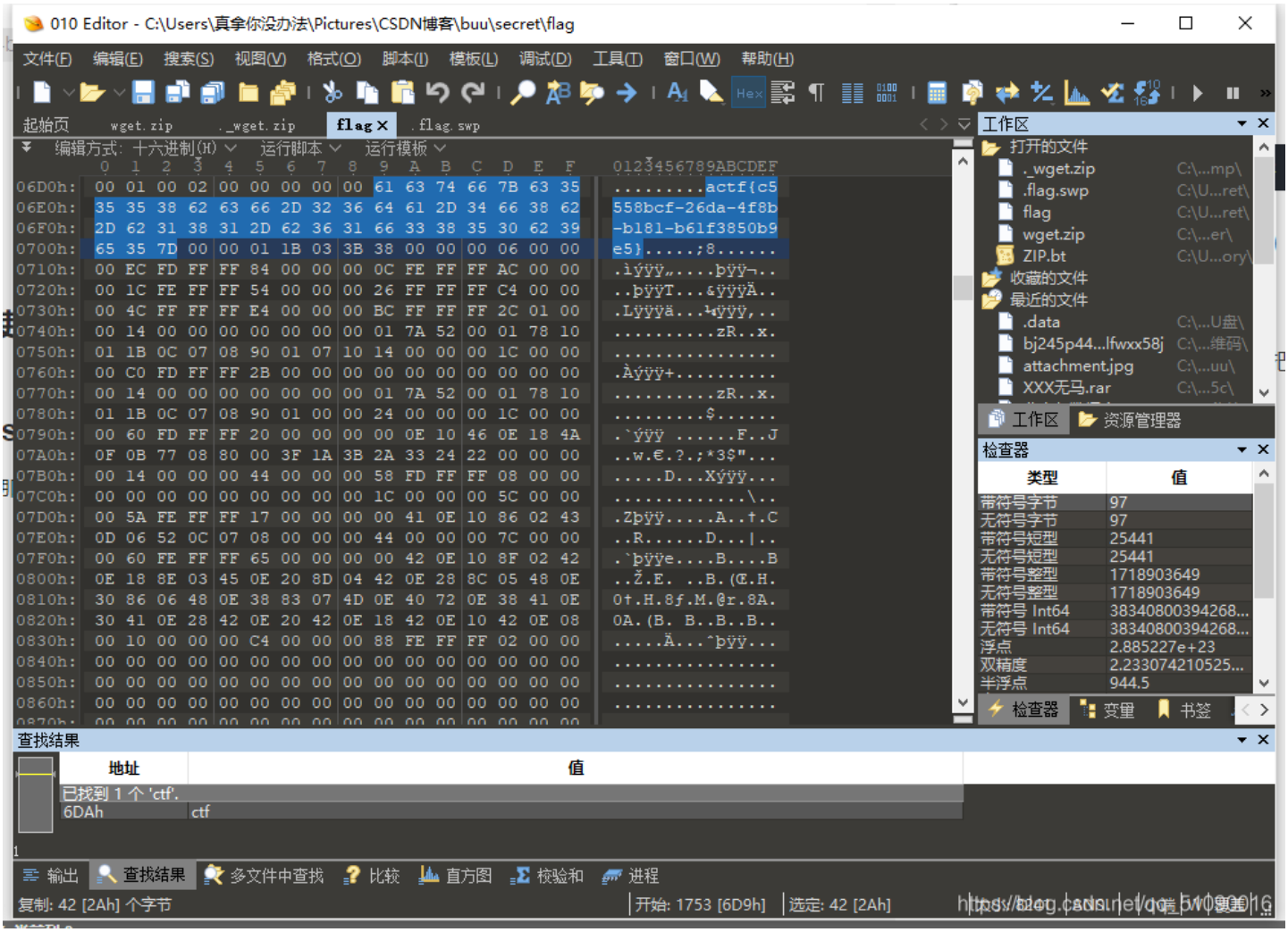
```
3b4b5dccd2c008fe7e2664bd1bc19292
```

小易的U盘（ida的使用）

这个 本来ida一直出错。。百度知道ida安装路径不能有中文，改了之后就好了

[ACTF新生赛2020]swp（导出http）

这题主要是根据swp的提示想到把那个pcapng文件导出http，获得了一堆文件，里面有一个压缩包，解压后搜索ctf就有了



百里挑一

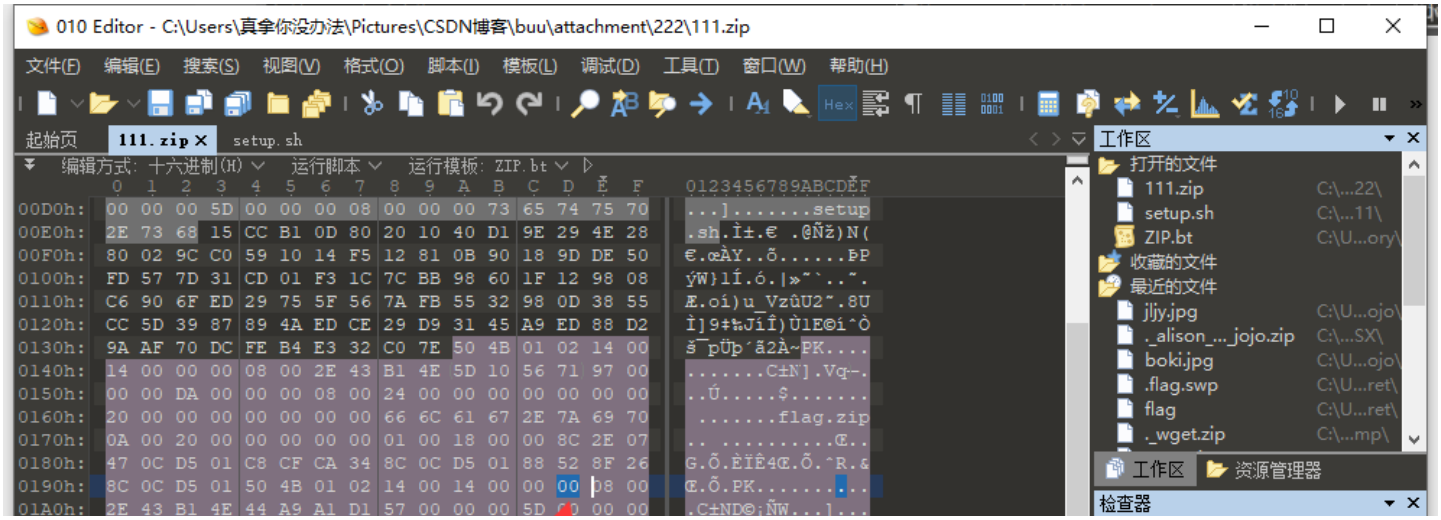
[WUSTCTF2020]alison_likes_jojo（outguess隐写）

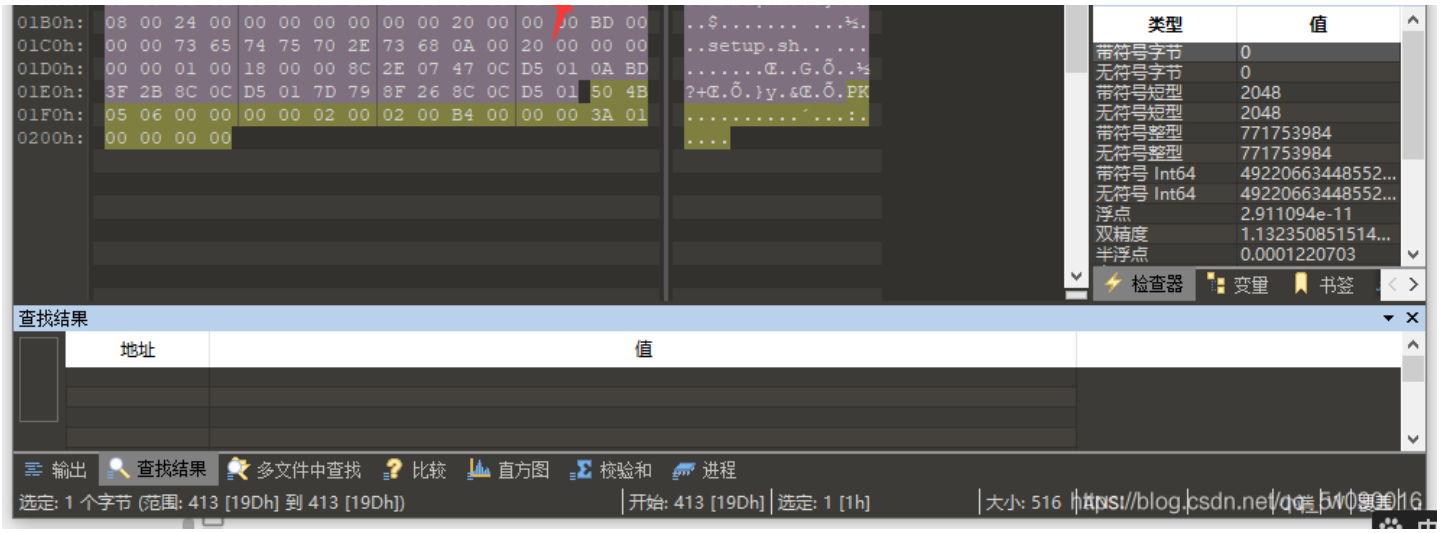
这一题只做了一半，outguess遇到的太少了，还是太菜了

[GUET-CTF2019]zips（伪加密 python时间戳？）

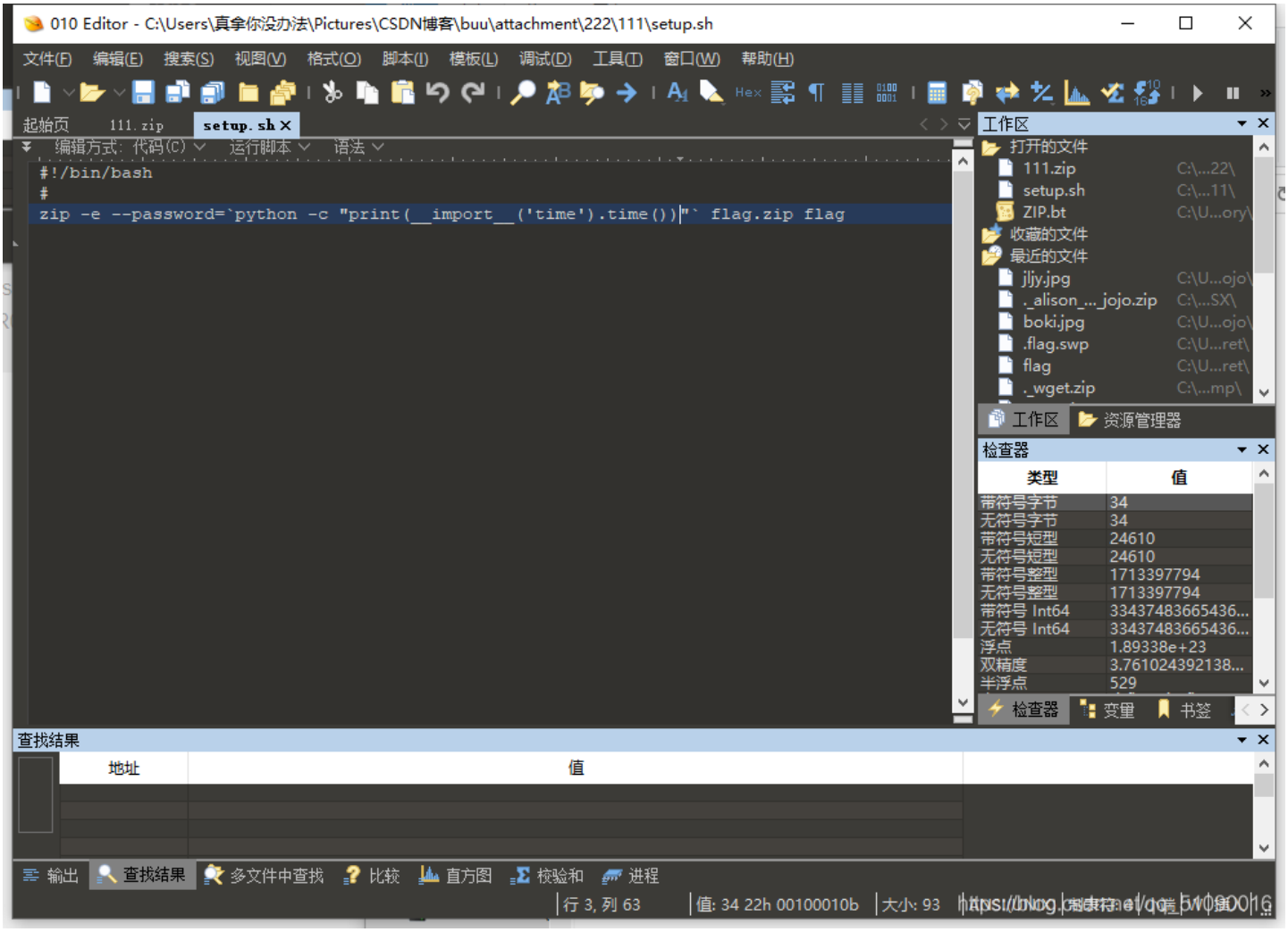
zip伪加密还是没搞透，先是第一层的爆破，得到密码723456

然后是第二层的伪加密：把这个位置的09改成00就行了





解压得到



这是要执行这段代码么

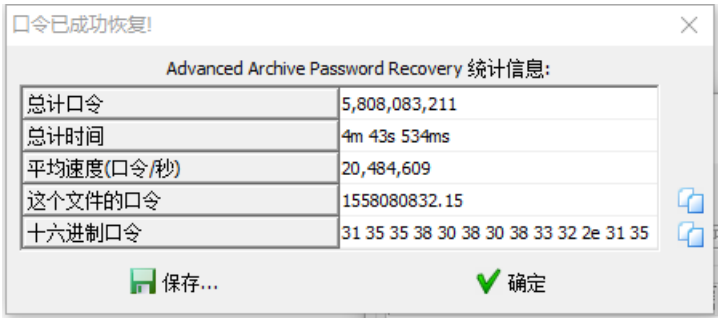
```
1617716051.4345222
```

执行完了。。怎么不是密码,看 WP吧

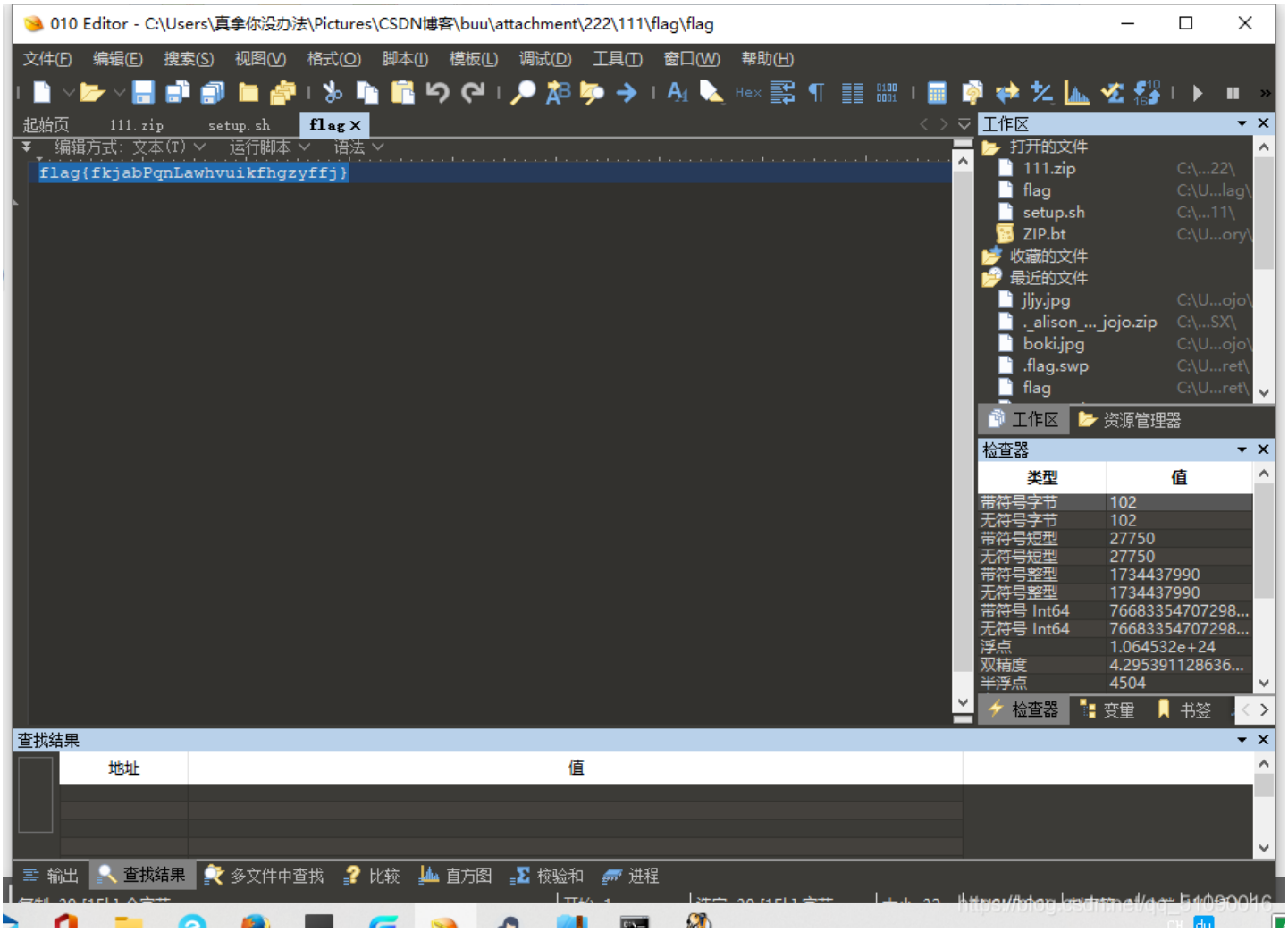
因为时间戳和出题当时不太一样所以往前推，利用ARCHPR设置掩码15???.?? 掩码符号为**?**

不太懂这个意思

算了，照着wp做



得到了这个，解压

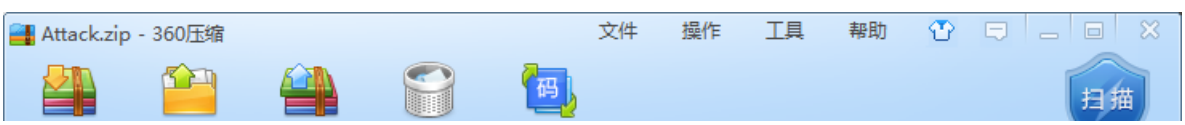


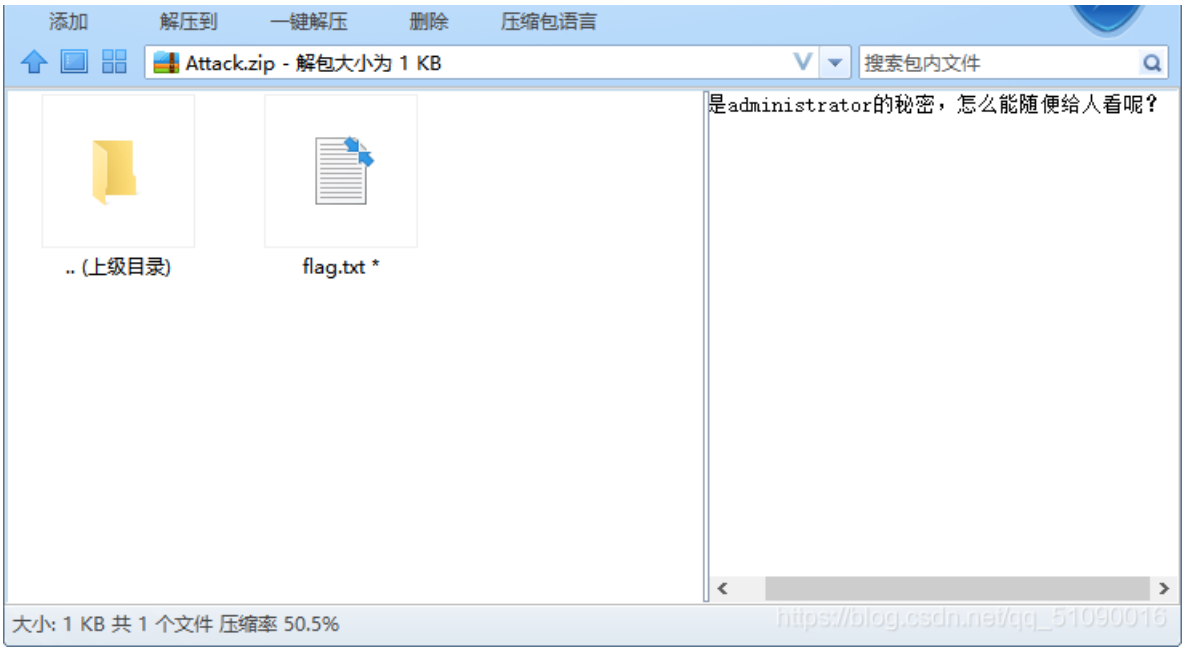
ok了，主要就是那个时间戳不知道啥意思

[安洵杯 2019]Attack（mimikatz的使用）

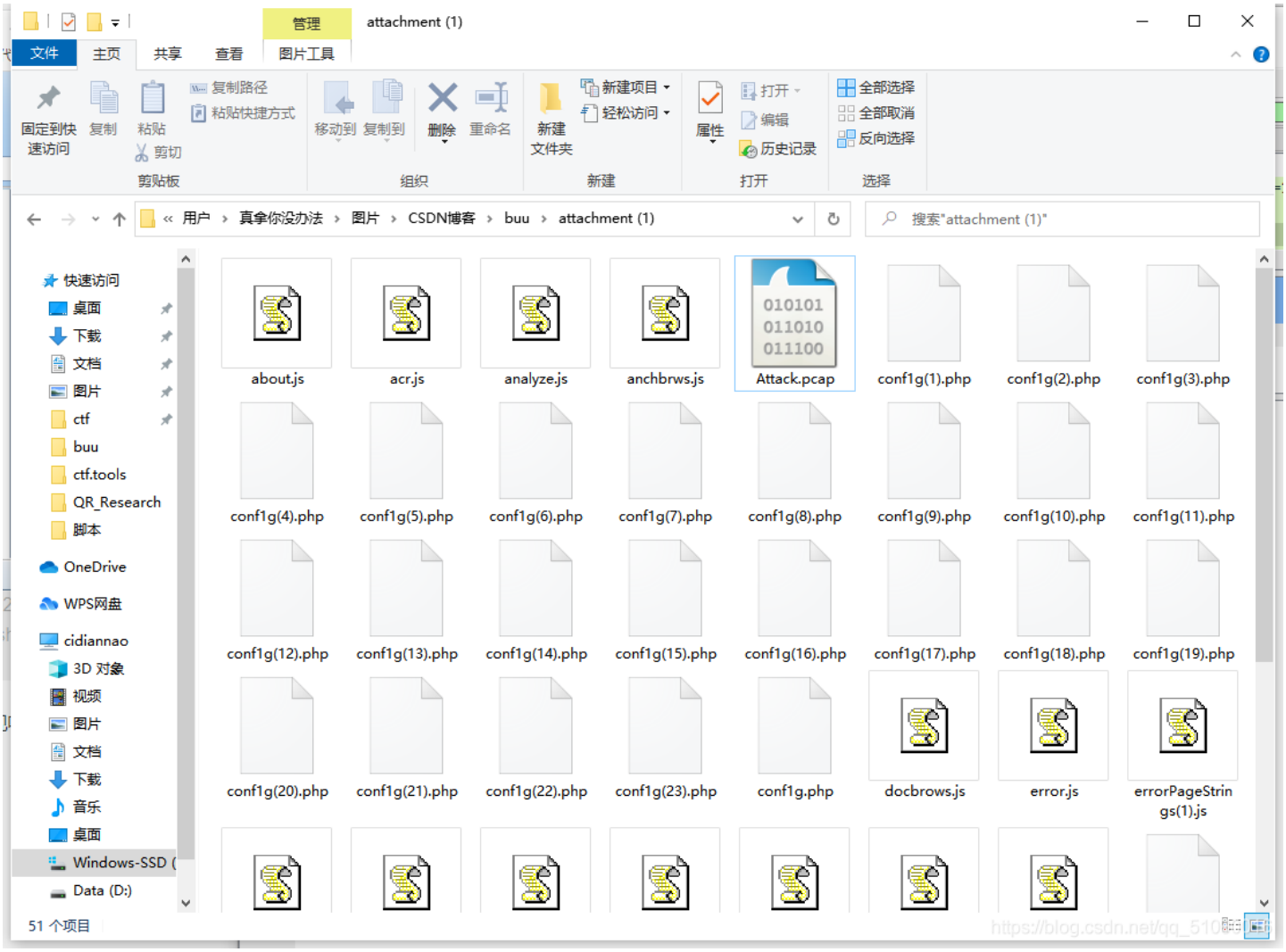
又是一道涉及知识盲区的题

先是用010打开看了看，发现里面藏了txt文件，懒得开虚拟机，改成zip后缀试试





密码肯定要在原文件里搞到了，我先是看tcp流。。真的没发现啥东西，再导出http



这一堆实在让我不知道干啥。。看wp吧

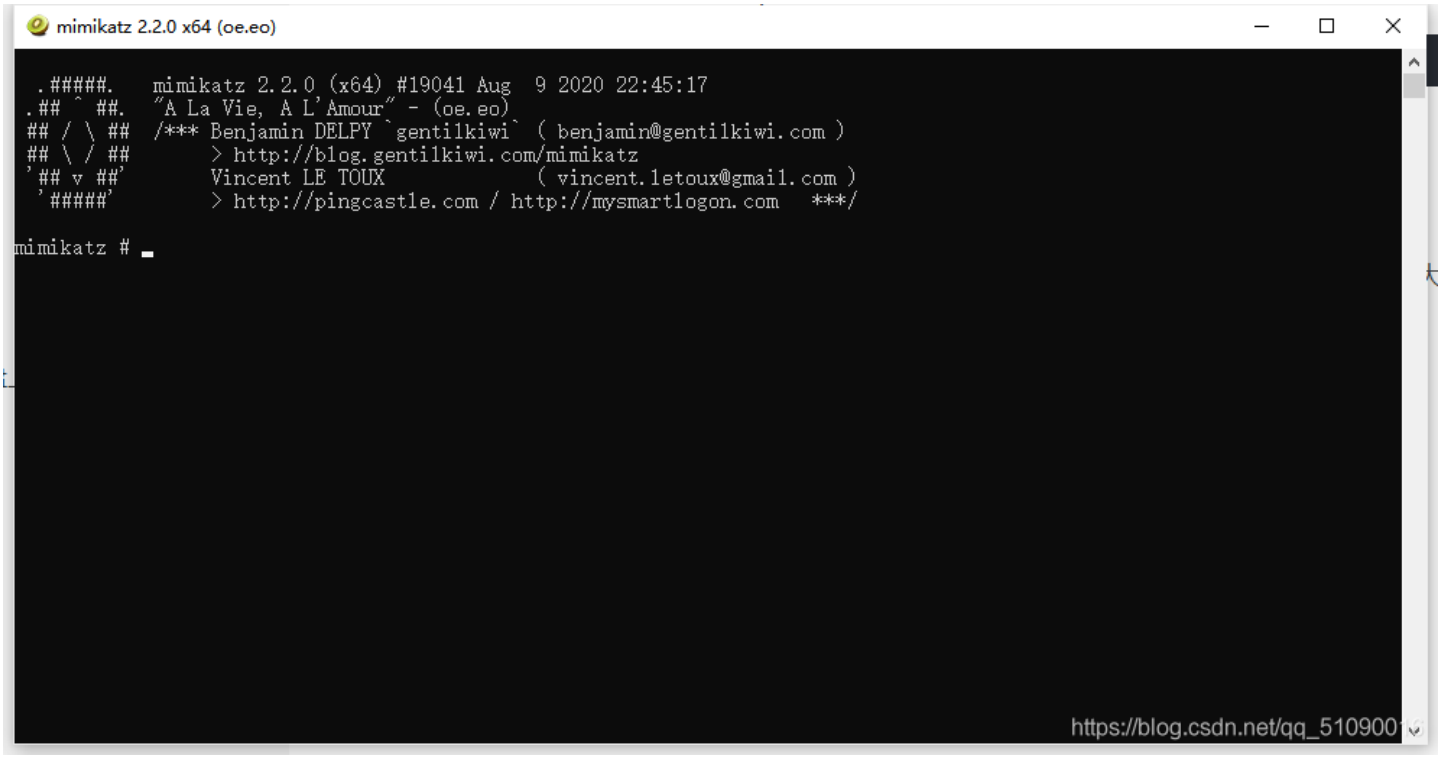
```
lsass.exe是一个系统重要进程，用于微软Windows系统的安全机制。它用于本地安全和登陆策略。
```

这里利用mimikatz获取明文密码，保存lsass下来放入mimikatz同目录下（64位）
用x64位右键管理员运行打开
mimikatz（内网渗透工具，可在lsass.exe进程中获取windows的账号明文密码）

这里的minikatz我在github上下载的没有exe文件。。。可能是被直接删掉了，最后从下面这个大佬的网盘上下的

https://blog.csdn.net/weixin_45663905/article/details/108013149

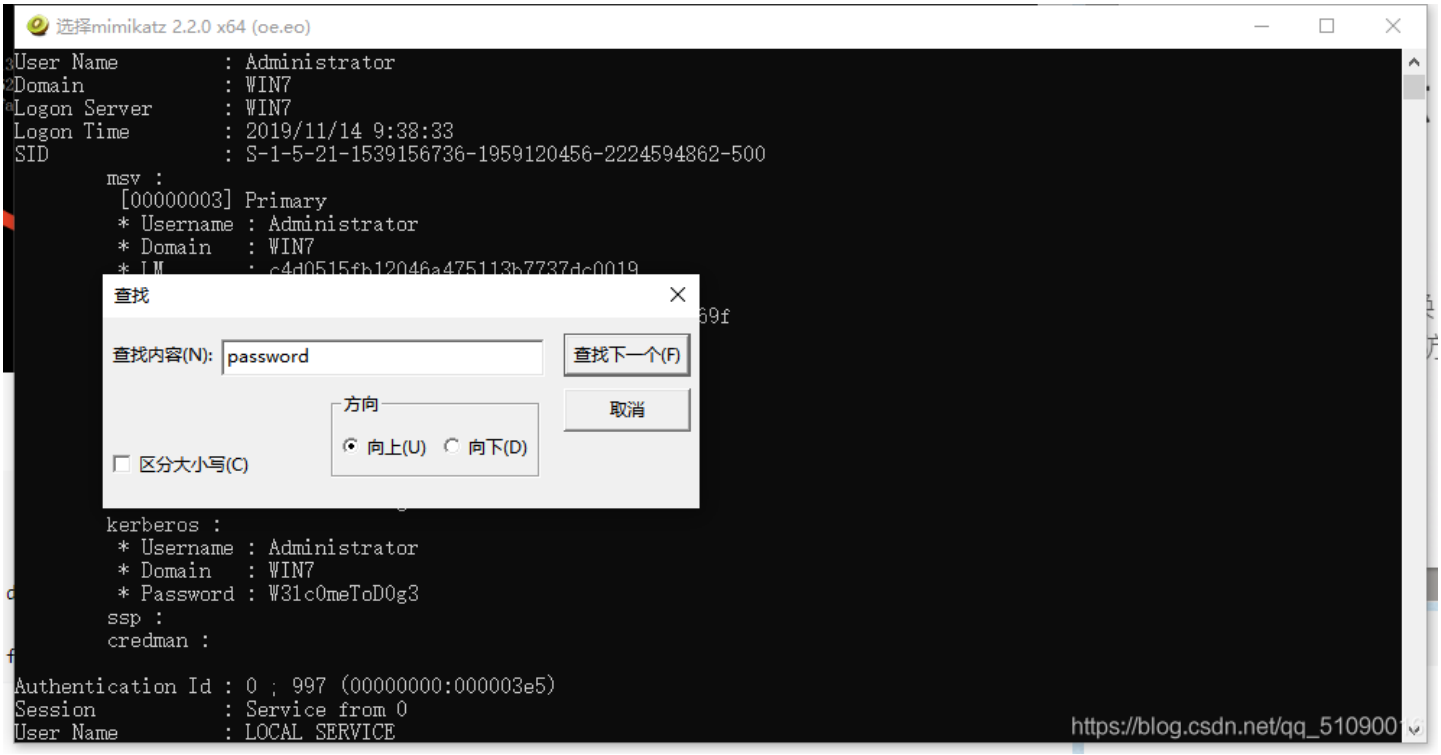
用管理员权限才能打开：

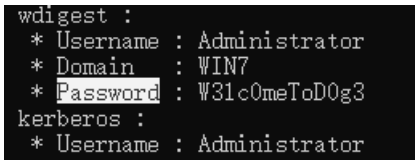


具体步骤：

```
//提升权限
privilege::debug
//载入dmp文件
sekurlsa::minidump lsass.dmp
//读取登陆密码
sekurlsa::logonpasswords full
```

搜索一下password

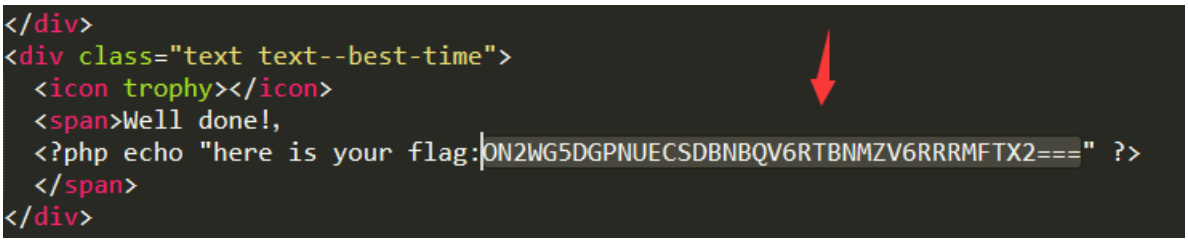




有了
输入密码解压就行了

[SUCTF 2019]Game

先是用vs code打开发现一串base32



得到 `suctf{hAHaha_Fak3_F1ag}`
假的flag。。。

再看另一张图片，LSB发现一串base64

这串base64解码后头部是Salted，应该是AES或者3DES

[WUSTCTF2020]girlfriend（DTMF拨号音 手机键盘密码）

先是把音频用audacity打开，结果并不是摩斯电码，麻了，不知道啥东西，看wp

猜测DTMF拨号音识别，有个程序可以识别一下dtmf2num.exe

下载地址

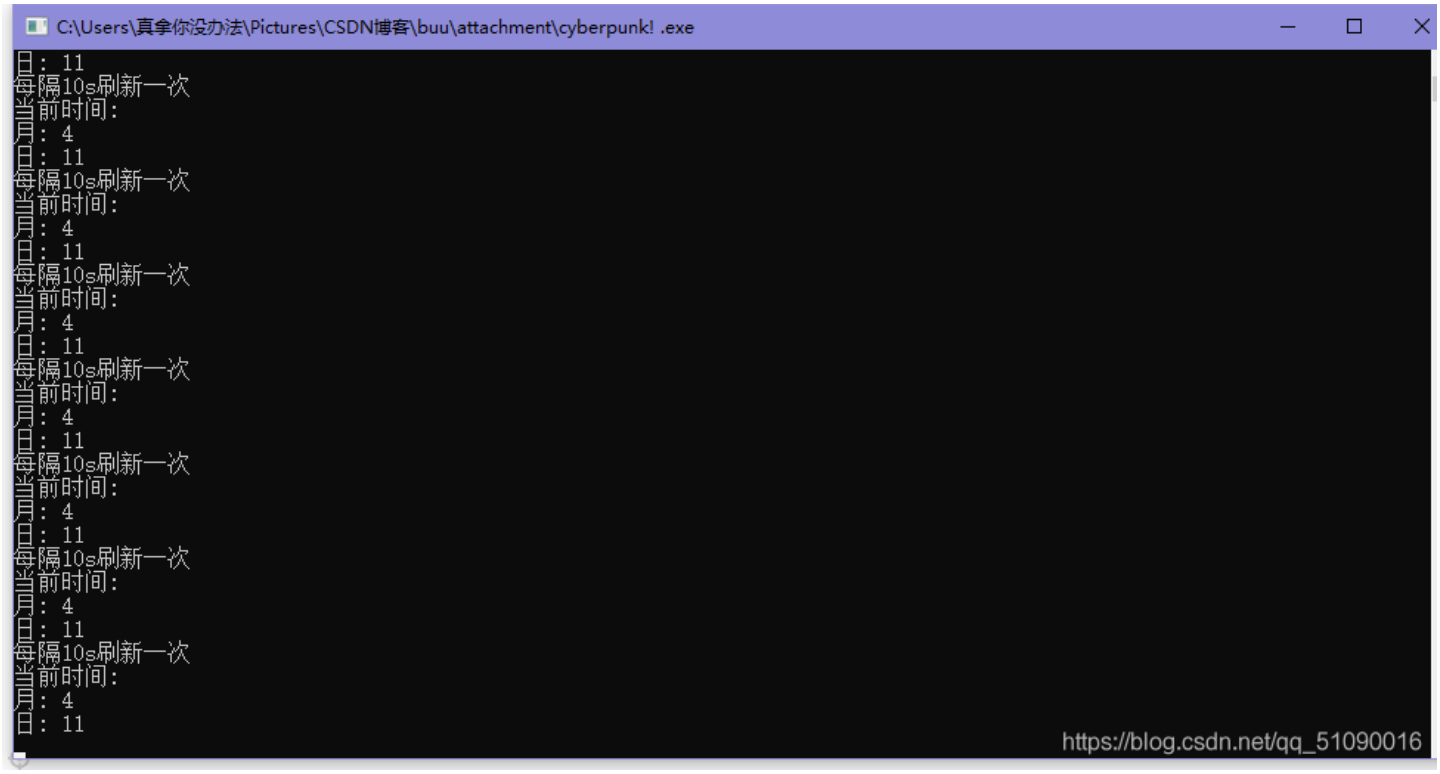
放在同一个文件夹下打开cmd

然后就是手机键盘密码

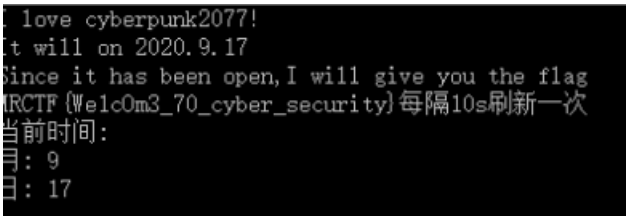

```
999     --->   y
666     --->   o
88      --->   u
2       --->   a
777     --->   r
33      --->   e
6       --->   m
999     --->   y
4       --->   g
4444    --->   i
777     --->   r
555     --->   l
333     --->   f
777     --->   r
444     --->   i
33      --->   e
66      --->   n
3       --->   d
7777    --->   s

youaremygirlfriends
```

[MRCTF2020]CyberPunk

又是一道脑洞题



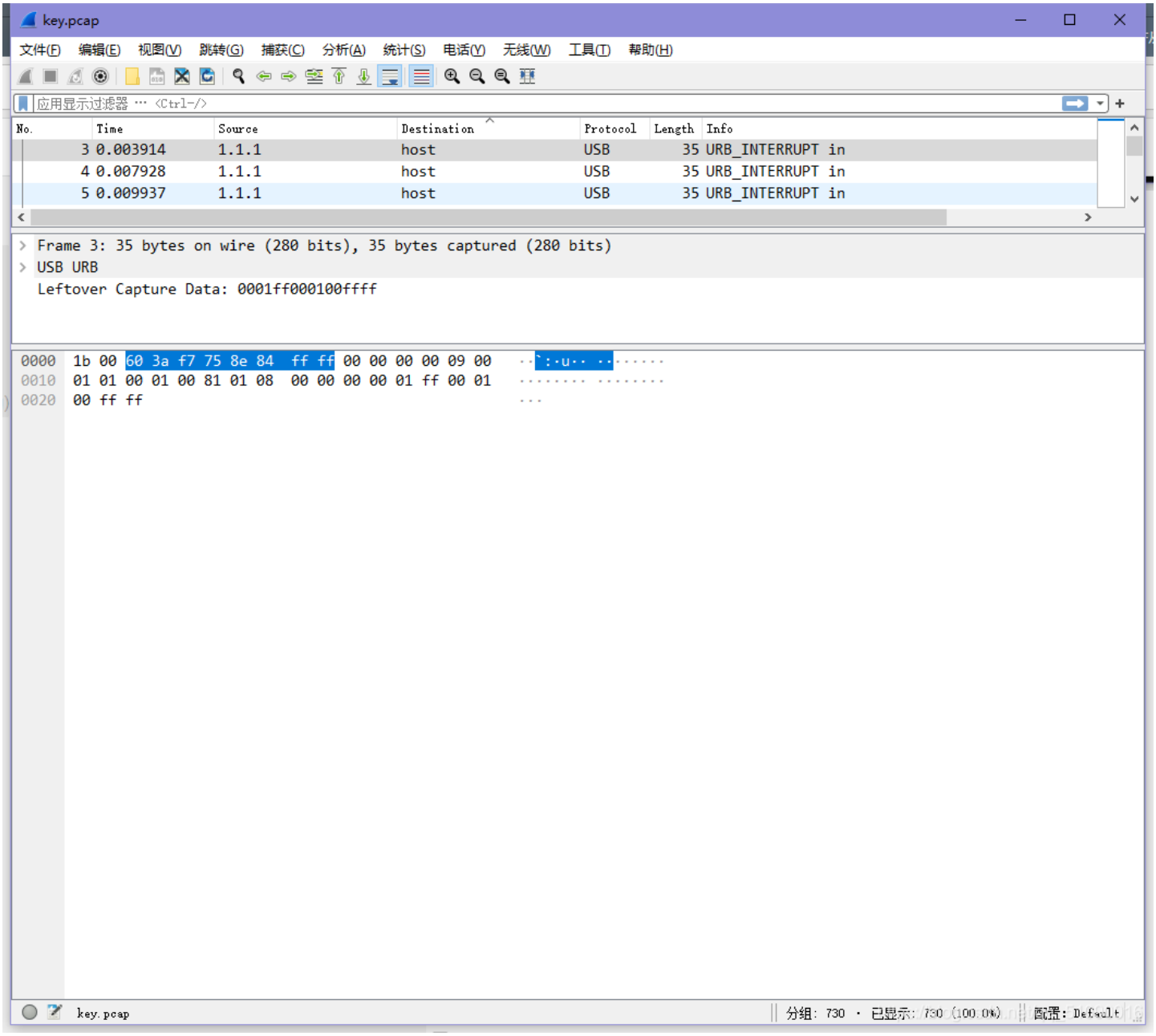
需要我们把电脑时间调到2020年9月17日就有了



USB（USB数据包 维吉尼亚解密）

010打开后发现里面有很多rar，改成rar后缀,解压得到一个pcapng文件

根据wp，pcapng文件里面是USB数据包：



直接使用UsbKeyboardDataHacker脚本提取内容

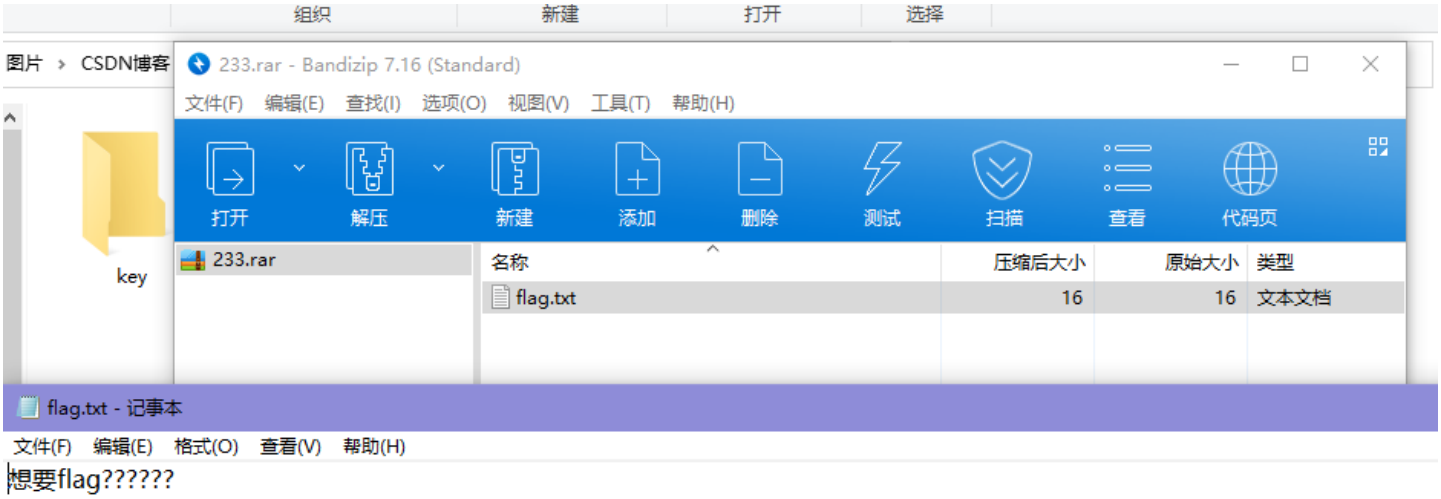
```
https://github.com/WangYihang/UsbKeyboardDataHacker
```

下载后放到kali里，把pcap文件和脚本放在一个文件夹，用命令行打开，然后输入python UsbKeyboardDataHacker.py key.pcap

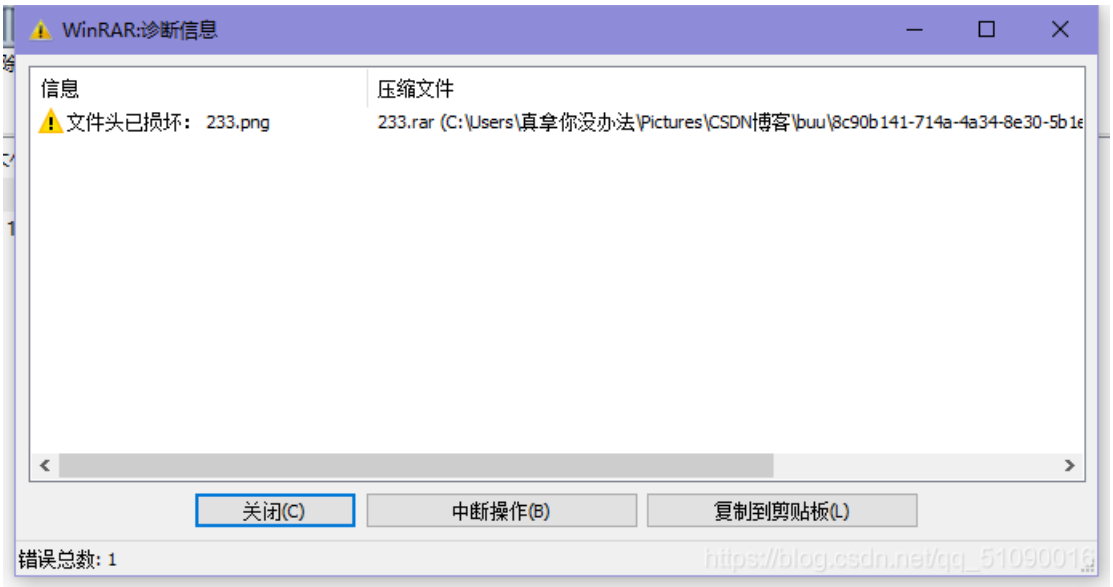
```
acker.py key.pcap
[-] Unknow Key : 01
[-] Unknow Key : 01
[+] Found : aababacbbdbdccccdcdcdbbcccbcbbcbbaababaaaaaaaaaaaaaaaaaakey{xin
an}
```

```
key{xinan}
```

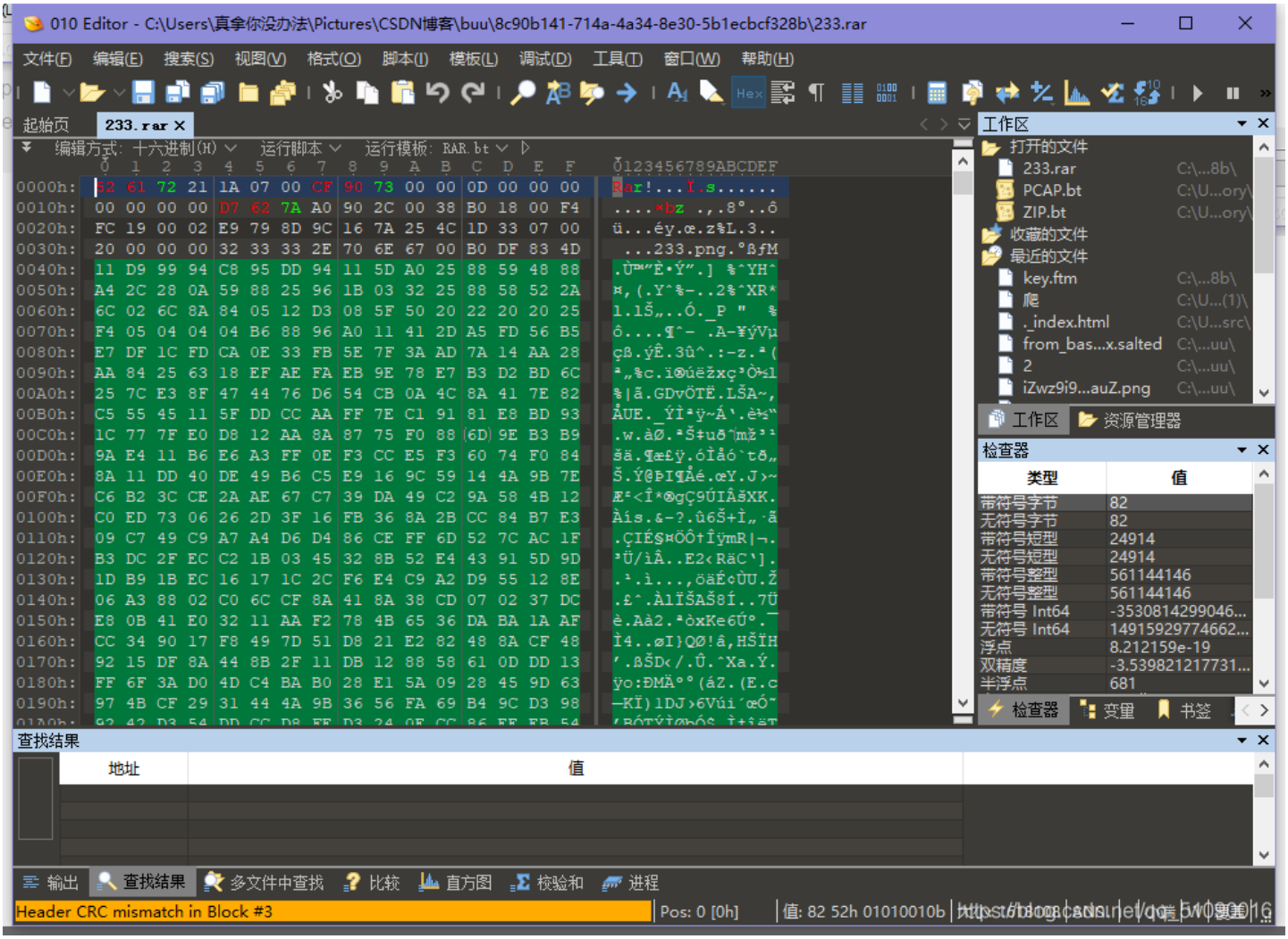
再回过头看另一个rar文件，里面的txt文件显然并没有flag



用WinRAR打开显示文件头损坏



010打开看看



crc错误，而且里面有个png图片，修复一下将70A0改为74A0

再用stegsolve打开图片，在一个通道里发现了二维码

得到`ci{v3erf_0tygidv2_fc0}`

维吉尼亚解密

前面没有flag，但整个字符串里可以拼到，应该是栅栏密码，再进行栅栏解密

fa{i3eei_011gvgn2_sc0}

每组字数 2 加密 解密

flag{vig3ne2e_is_c00l}