

# LSB隐写+画图

原创

未完成的歌~ 于 2019-04-10 20:20:41 发布 692 收藏 1

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\\_43531669/article/details/89192331](https://blog.csdn.net/qq_43531669/article/details/89192331)

版权

关于LSB

## 一.LSB简介

LSB(英文 least significant bit)即最低有效位。LSB加密是信息隐藏中最基本的方法。由于人们识别声音或图片的能力有限，因此我们稍微改动信息的某一位是不会影响我们识别声音或图片的。

## 二.用法

通常来说LSB加密用在无损压缩的数据格式文件中，例如图像中的bmp格式和音频的wav格式。由于这两种格式未对源数据进行有损压缩，因此可以将信息隐藏起来。

### BMP文件中的使用

对于图像文件LSB的特征很明显，通常将信息隐藏在某一个颜色通道中。我们可以查看图片的每个像素点的RGB值，或者使用stegsolve工具进行查看。

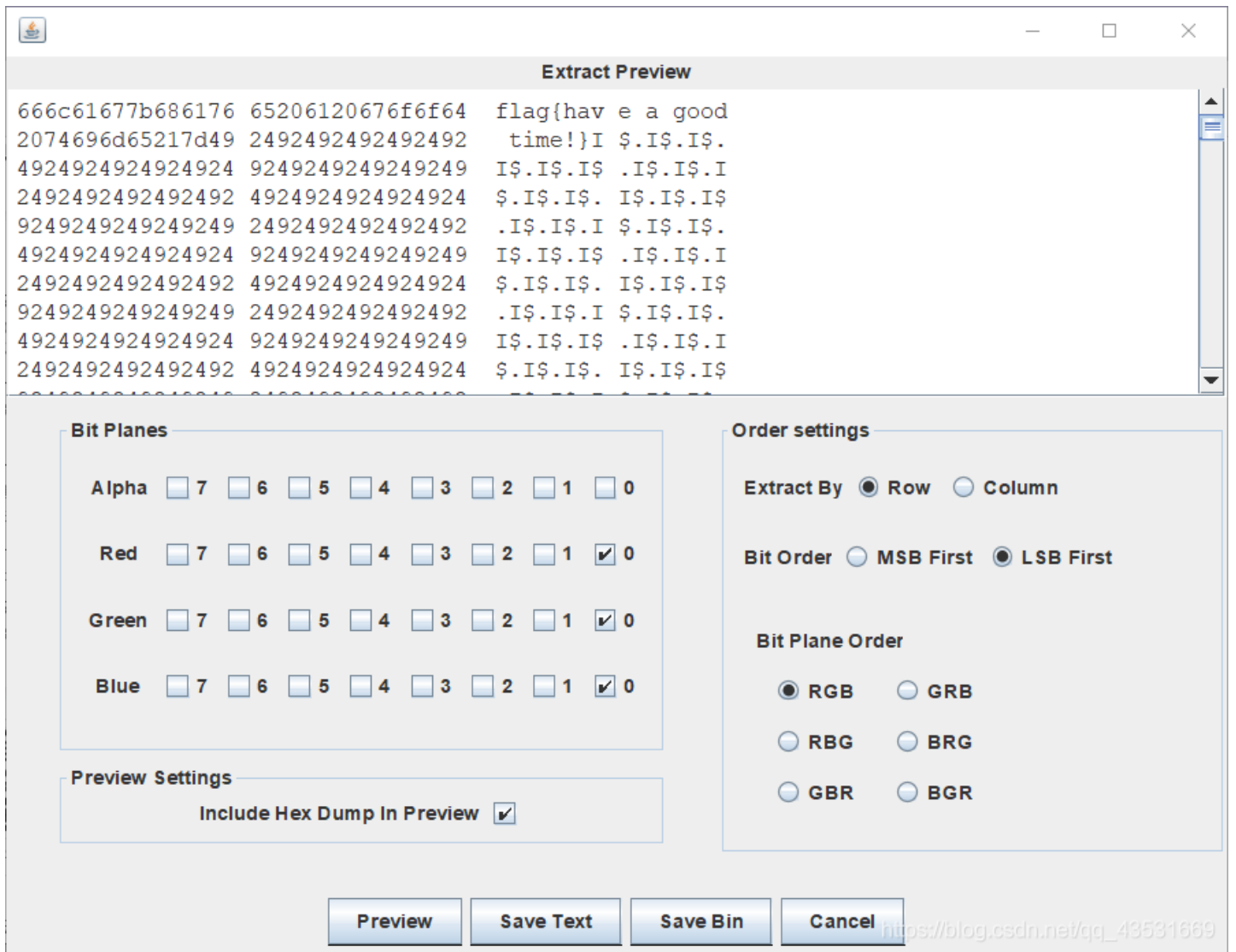
由于图像是由像素构成的，每个像素有8位(对于BMP图像来说),通常最后一位的变化，通过肉眼是无法察觉的一道LSB隐写题，打开有张图：



[https://blog.csdn.net/qq\\_43531669](https://blog.csdn.net/qq_43531669)

使用stegsolve工具进行查看，->分析->数据提出

Bit Order选择LSB Lirst（最低有效位），MSB是最高有效位；分析如下



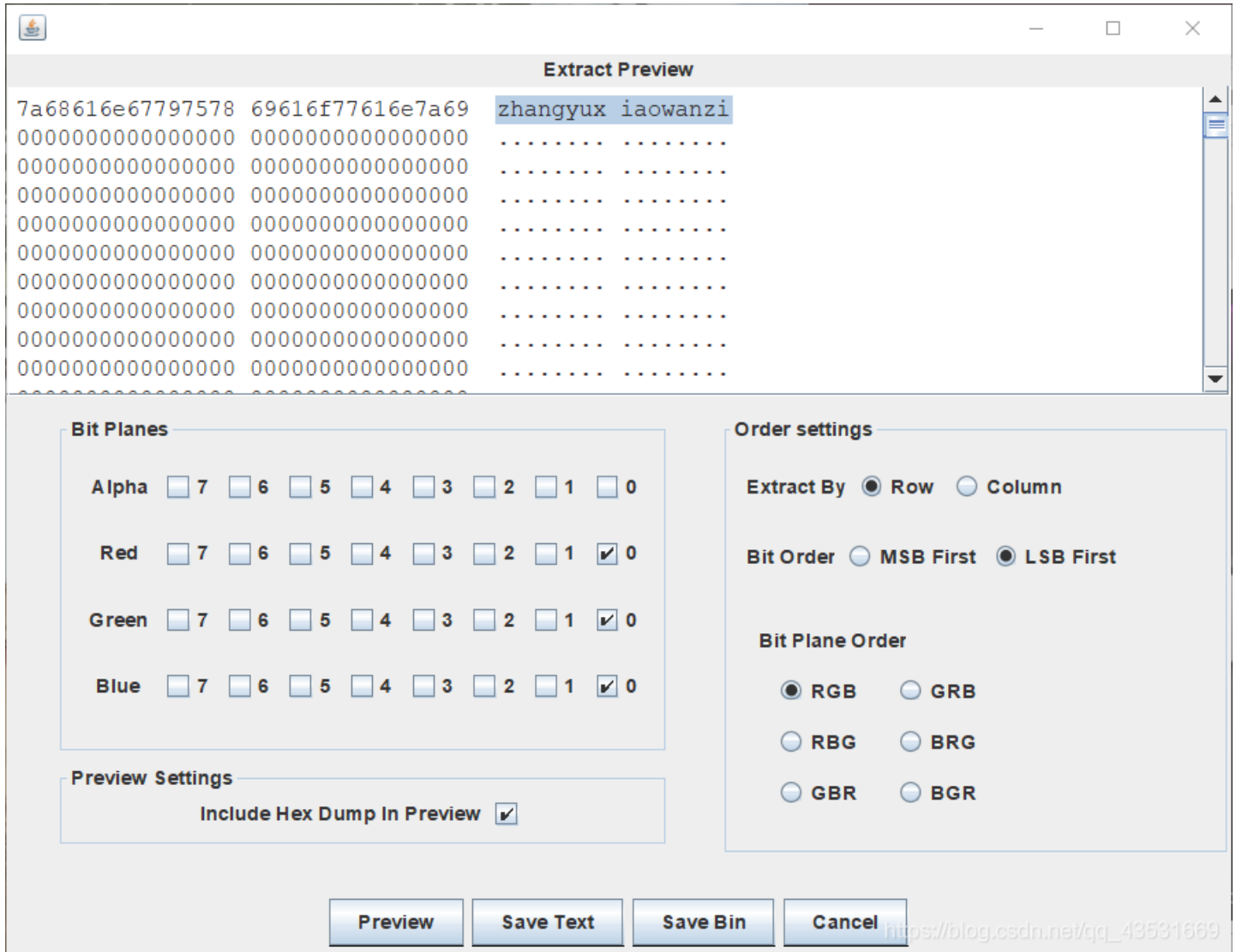
发现 flag{hav e a good time!}

LSB隐写2:

打开为:

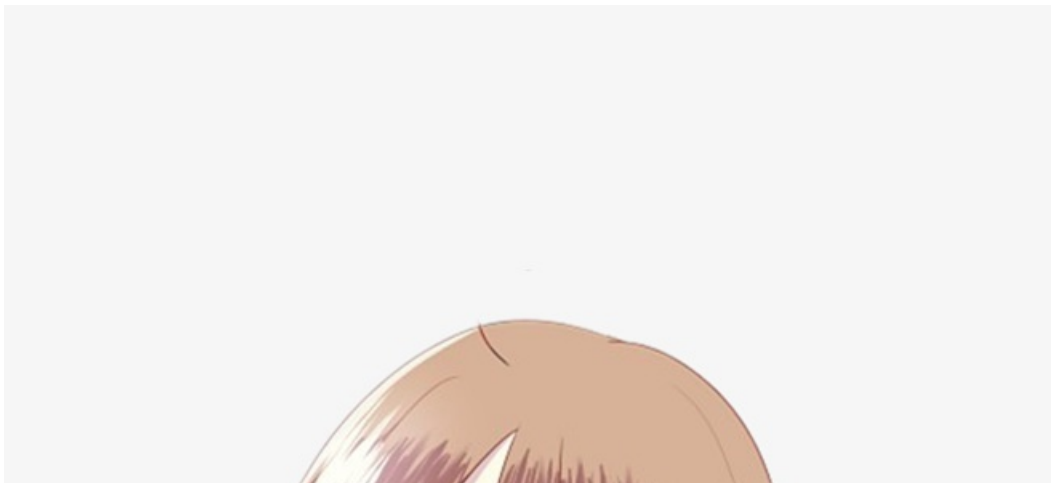


同样的套路，得到：



画图题：

打开发现有个萌妹子：





用winhex打开发现图片后面有很多的数据:

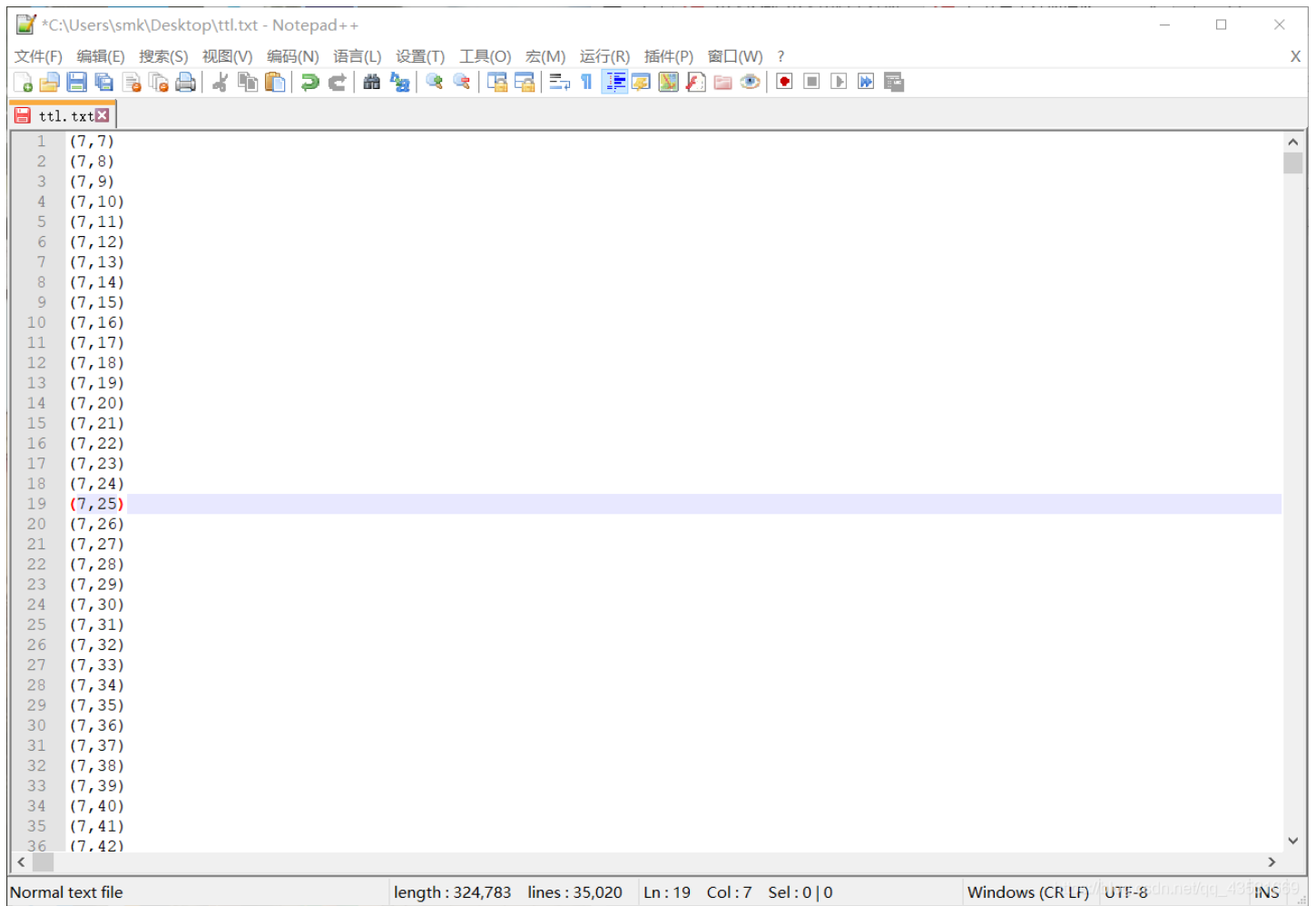
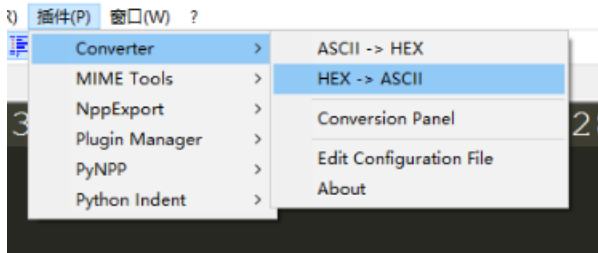
paint.png

Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	ANSI ASCII
0002DBB0	40	E5	B6	0C	48	40	02	12	90	80	04	72	22	A0	72	E7	0á¶ H0 € r" rç
0002DBC0	94	5B	DA	2A	01	09	48	40	02	12	50	B9	2D	03	12	90	"[ú* H0 P¹-
0002DBD0	80	04	24	20	81	9C	08	A8	DC	39	E5	96	B6	4A	40	02	€ \$ α ``Ü9á-¶J0
0002DBE0	12	90	80	04	54	6E	CB	80	04	24	20	01	09	48	20	27	€ TnÈ€ \$ H '
0002DBF0	02	2A	77	4E	B9	A5	AD	12	90	80	04	24	20	01	95	DB	*wN¹¶- € \$ •Û
0002DC00	32	20	01	09	48	40	02	12	C8	89	80	CA	9D	53	6E	69	2 H0 ÈÈÈÈ Sni
0002DC10	AB	04	24	20	01	09	48	40	E5	B6	0C	48	40	02	12	90	« \$ H0á¶ H0
0002DC20	80	04	72	22	A0	72	E7	94	5B	DA	2A	01	09	48	40	02	€ r" rç"[ú* H0
0002DC30	12	50	B9	2D	03	12	90	80	04	24	20	81	9C	08	A8	DC	P¹- € \$ α ``Ü
0002DC40	39	E5	96	B6	4A	40	02	12	90	80	04	54	6E	CB	80	04	9á-¶J0 € TnÈ€
0002DC50	24	20	01	09	48	20	27	02	2A	77	4E	B9	A5	AD	12	90	\$ H ' *wN¹¶-
0002DC60	80	04	24	20	01	95	DB	32	20	01	09	48	40	02	12	C8	€ \$ •Û2 H0 È
0002DC70	89	80	CA	9D	53	6E	69	AB	04	24	20	01	09	48	40	E5	ÈÈÈ Sni« \$ H0á
0002DC80	B6	0C	48	40	02	12	90	80	04	72	22	A0	72	E7	94	5B	¶ H0 € r" rç"[
0002DC90	DA	2A	01	09	48	40	02	12	50	B9	2D	03	12	90	80	04	ú* H0 P¹- €
0002DCA0	24	20	81	9C	08	A8	DC	39	E5	96	B6	4A	40	02	12	90	\$ α ``Ü9á-¶J0
0002DCB0	80	04	54	6E	CB	80	04	24	20	01	09	48	20	27	02	2A	€ TnÈ€ \$ H ' *
0002DCC0	77	4E	B9	A5	AD	12	90	80	04	24	20	01	95	DB	32	20	wN¹¶- € \$ •Û2
0002DCD0	01	09	48	40	02	12	C8	89	80	CA	9D	53	6E	69	AB	04	H0 ÈÈÈÈ Sni«
0002DCE0	24	20	01	09	48	E0	FF	01	1E	A7	A6	E0	6B	49	11	10	\$ Hàÿ \$!áKI
0002DCF0	00	00	00	00	49	45	4E	44	AE	42	60	82	32	38	33	37	IEND@B`,2837
0002DD00	32	63	33	37	32	39	30	61	32	38	33	37	32	63	33	38	2c37290a28372c38
0002DD10	32	39	30	61	32	38	33	37	32	63	33	39	32	39	30	61	290a28372c39290a
0002DD20	32	38	33	37	32	63	33	31	33	30	32	39	30	61	32	38	28372c3130290a28
0002DD30	33	37	32	63	33	31	33	31	32	39	30	61	32	38	33	37	372c3131290a2837
0002DD40	32	63	33	31	33	32	32	39	30	61	32	38	33	37	32	63	2c3132290a28372c
0002DD50	33	31	33	33	32	39	30	61	32	38	33	37	32	63	33	31	3133290a28372c31
0002DD60	33	34	32	39	30	61	32	38	33	37	32	63	33	31	33	35	34290a28372c3135
0002DD70	32	39	30	61	32	38	33	37	32	63	33	31	33	36	32	39	290a28372c313629
0002DD80	30	61	32	38	33	37	32	63	33	31	33	37	32	39	30	61	0a28372c3137290a
0002DD90	32	38	33	37	32	63	33	31	33	38	32	39	30	61	32	38	28372c3138290a28
0002DDA0	33	37	32	63	33	31	33	39	32	39	30	61	32	38	33	37	372c3139290a2837
0002ddb0	32	63	33	32	33	30	32	39	30	61	32	38	33	37	32	63	2c3230290a28372c
0002DDC0	33	32	33	31	32	39	30	61	32	38	33	37	32	63	33	32	3231290a28372c32
0002DDD0	33	32	32	39	30	61	32	38	33	37	32	63	33	32	33	33	32290a28372c3233
0002DDE0	32	39	30	61	32	38	33	37	32	63	33	32	33	34	32	39	290a28372c323429
0002DDF0	30	61	32	38	33	37	32	63	33	32	33	35	32	39	30	61	0a28372c3235290a
0002DE00	32	38	33	37	32	63	33	32	33	36	32	39	30	61	32	38	28372c3236290a28
0002DE10	33	37	32	63	33	32	33	37	32	39	30	61	32	38	33	37	372c3237290a2837

```
0002DE20 32 63 33 32 33 38 32 39 30 61 32 38 33 37 32 63 2c3238290a28372c
0002DE30 33 32 33 39 32 39 30 61 32 38 33 37 32 63 33 33 3239290a28372c33
```

[https://blog.csdn.net/qq\\_43531669](https://blog.csdn.net/qq_43531669)

复制下来保存为txt文件用notepad++转换一下编码

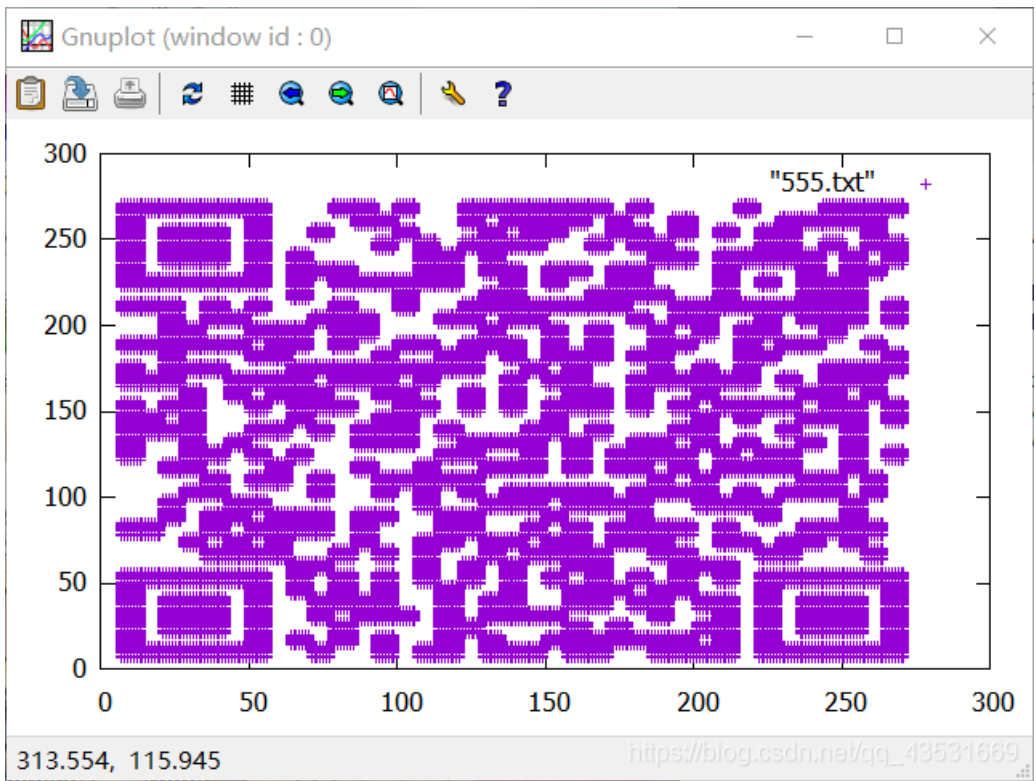


接下来就是取坐标，用Notepad++的替换功能：

```
*C:\Users\smk\Desktop\ttl.txt - Notepad++
文件(F) 编辑(E) 搜索(S) 视图(V) 编码(N) 语言(L) 设置(T) 工具(O) 宏(M) 运行(R) 插件(P) 窗口(W) ?
ttl.txt x
1 7 7
2 7 8
3 7 9
4 7 10
5 7 11
6 7 12
7 7 13
8 7 14
9 7 15
10 7 16
11 7 17
12 7 18
13 7 19
14 7 20
15 7 21
16 7 22
17 7 23
18 7 24
19 7 25
20 7 26
21 7 27
22 7 28
23 7 29
24 7 30
25 7 31
26 7 32
27 7 33
28 7 34
29 7 35
30 7 36
31 7 37
32 7 38
33 7 39
34 7 40
35 7 41
36 7 42
Normal text file length: 324,783 lines: 35,020 Ln: 23 Col: 6 Sel: 0|0 Windows (CR LF) UTF-8
```

另存为txt文件，接下来使用一个画图工具：gnuplot（百度 下载地址很多）把txt文件拖进bin目录下，然后执行命令：  
plot"文件名.txt"

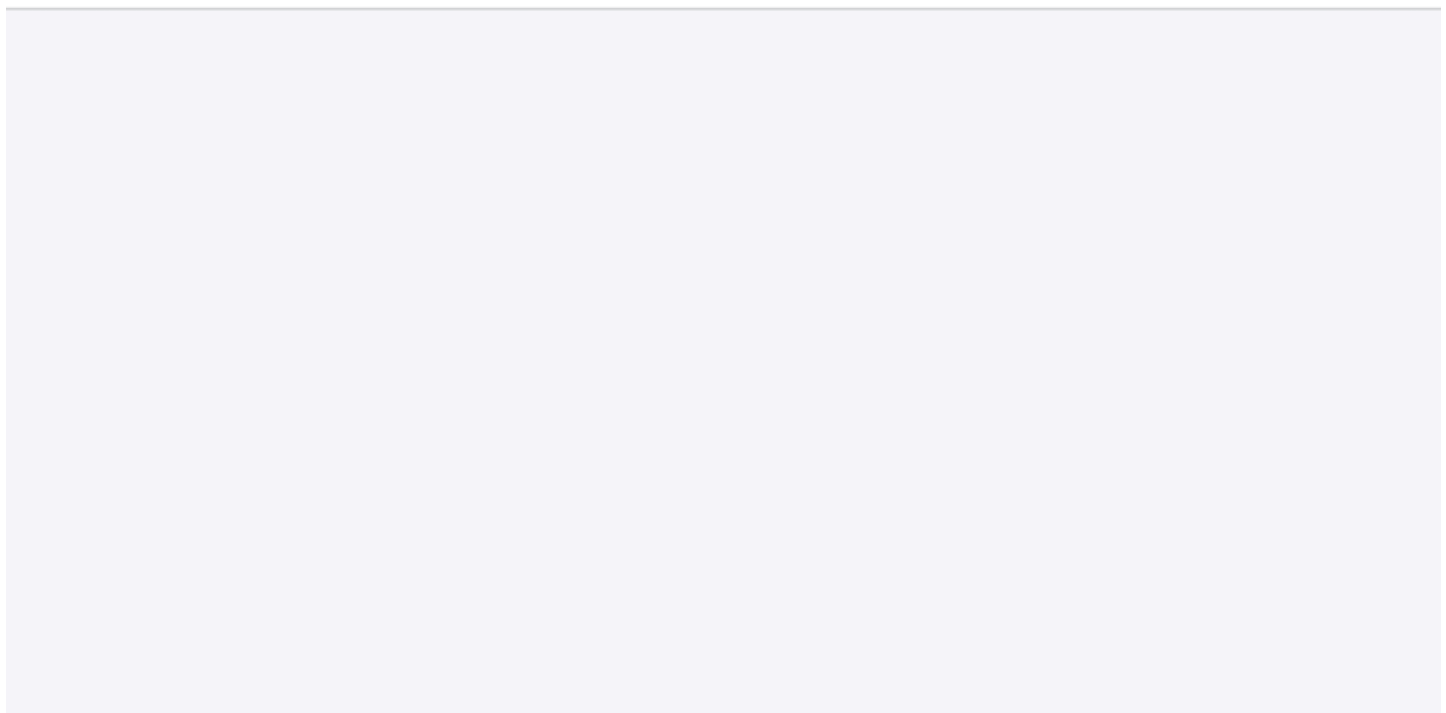
```
gnuplot
File Plot Expressions Functions General Axes Chart Styles 3D Help
GNU PLOT
Version 5.2 patchlevel 6 last modified 2019-01-01
Copyright (C) 1986-1993, 1998, 2004, 2007-2018
Thomas Williams, Colin Kelley and many others
gnuplot home: http://www.gnuplot.info
faq, bugs, etc: type "help FAQ"
immediate help: type "help" (plot window: hit 'h')
Terminal type is now 'wxt'
gnuplot> plot"555.txt"
gnuplot>
```



回车：  
扫描二维码得到flag:



## < 扫描结果



已识别到二维码内容

flag{40fc0a979f759c8892f4  
dc045e28b820}

复制内容

扫码内容非支付宝提供，请谨慎使用





