MOCTF ---- MISC -----WriteUp

原创

南人旧心1906 于 2019-04-26 19:39:03 发布 350 收藏
分类专栏： ctf

本文链接：https://blog.csdn.net/qq_42133828/article/details/89557579

1.我可是黑客

打开题目是一张图片，将他保存到本地，即：

一般第一题都是是道签到题，所以使用文档打开这张图片，小编用的是nodepad++，在文档的最后面，即：

```
NUL祢眙檬NULNULACKe埒k?FSv这娭锋蓄    ?
NUL斋[厄诋蘧_SI      ETXKū椭p??n誥;嫂%"SII)STX"  ""ST
敦z   ??=NAKRS们饲DC1j禾咨赌?肠*   瀠?~淤螳鹋翩w漘?=\1
NULNULNULNULmoctf{e4sy_1ma9e_m1sc}
```

2.假装安全

同上题，先将图片保存到本地，再用nodepad++打开，即：

UL NUL NUL NUL NUL NUL NUL NUL NUL NUL NUL NUL flag.txt

可以看到，在文档的最后面有flag.txt，这说明，这个图片不是一张简单的图片，他里面包含了文件，需要分离出来。

这里我使用的是kali 里的binwalk工具进行分离

在图片所在目录下，命令框执行语句：

```
binwalk -e carefully.jpg  //分离出carefully.jpg里的文件

ls

cd _carefully.jpg.extracted/    //进入分离出来的文件

ls

cat flag.txt           //查看flag
```

```
root@kali:~/Desktop# binwalk -e carefully.jpg

DECIMAL       HEXADECIMAL     DESCRIPTION
--------------------------------------------------------------------------------
0             0x0             JPEG image data, JFIF standard 1.01
30            0x1E            TIFF image data, big-endian, offset of first ima
32255         0x7DFF          Zip archive data, at least v2.0 to extract, com
32407         0x7E97          End of Zip archive, footer length: 22

root@kali:~/Desktop# ls
1.jpg  carefully.jpg  _carefully.jpg.extracted  mount-shared-folders.sh  resta
root@kali:~/Desktop# cd _
bash: cd: _: 没有那个文件或目录
root@kali:~/Desktop# cd _carefully.jpg.extracted/
root@kali:~/Desktop/_carefully.jpg.extracted# ls
7DFF.zip  flag.txt
root@kali:~/Desktop/_carefully.jpg.extracted# cat flag.txt
mcfCrflyS1eot{@eul_ld}root@kali:~/Desktop/_carefully.jpg.extracted#
```


3.扫扫出奇迹

同上题，先将图片保存到本地。可以看到，这是一张二维码，首先的反应一般都是先扫一扫

但是，不论你是正着扫，倒着扫，还是各种姿势扫，我先你会怀疑你的手机是不是wa掉了。。。

这时候，或许就不是你的问题了，该是二维码的问题了，从二维码可以很明显的看出“黑白分明”，

那会不会他是反着来的，上工具。StegSolve

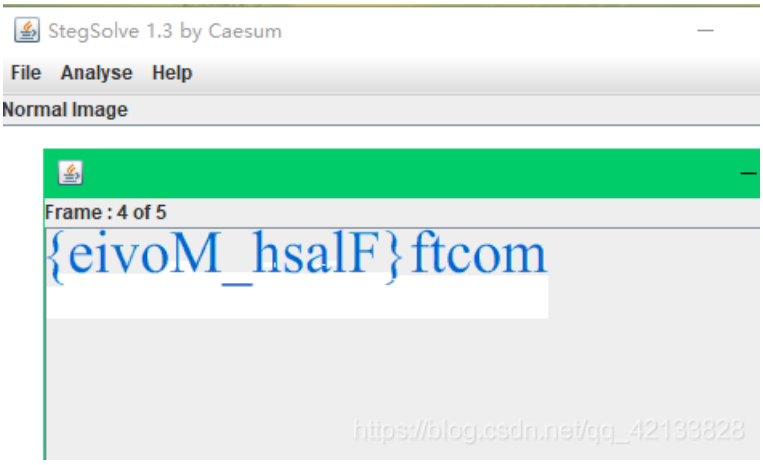
这回，在扫一扫，flag就出来了。。。。。

4.光阴似箭

同上题，先将图片保存到本地。打开图片，可以看到，有flag闪过。。。。

Where is the flag ?

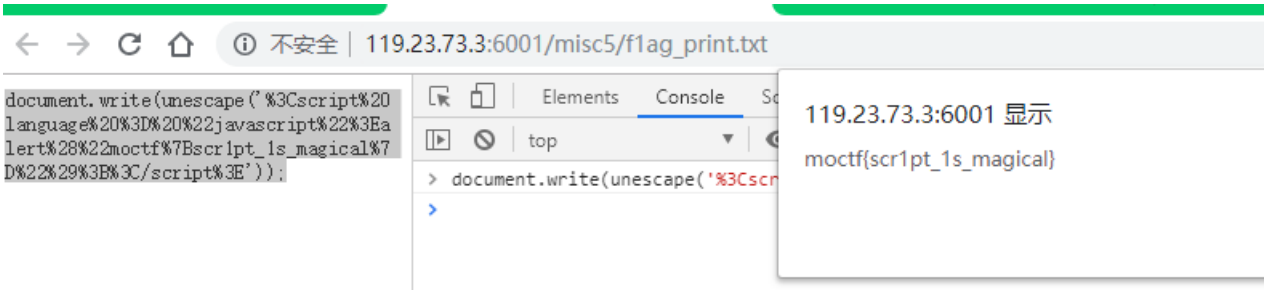
直接上工具，StegSolve，在Analyse的 Frame Brower 里，即：



5.杰斯的魔法

打开题目，这道题看来并不是一道图片题，并且页面给出了一串代码，瞅着像是JS代码。。。

额。。那就尝试在console里运行，即：

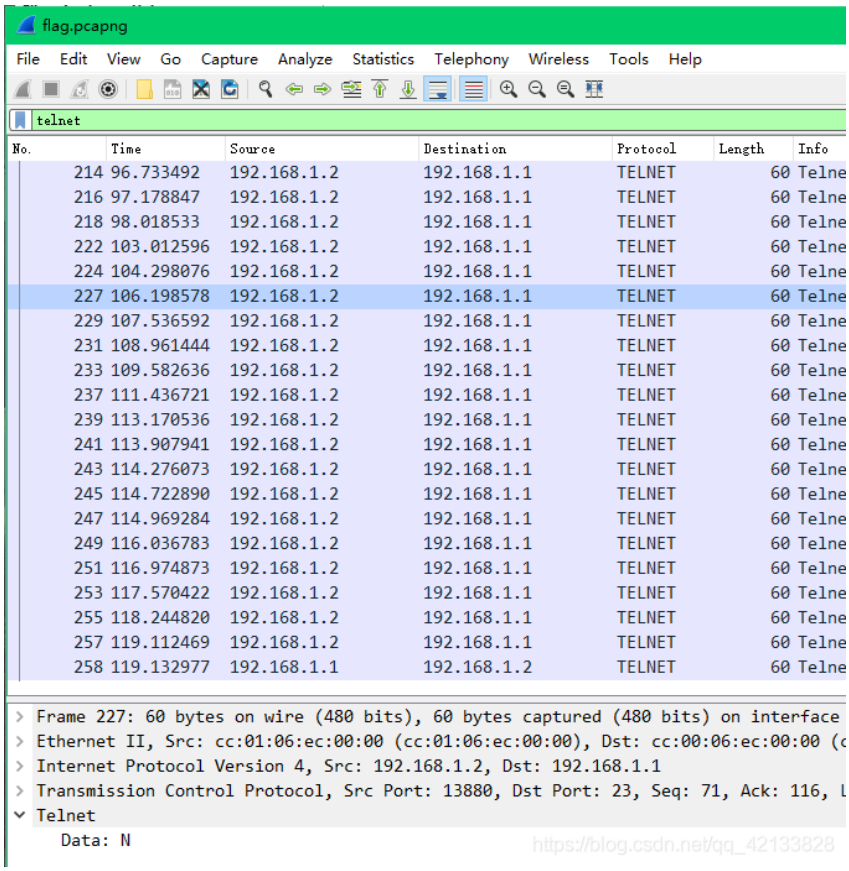


6.流量分析

下载好题目给的文件，是一个以.pcapng结尾的文件，用wireshark打开，可以看到是一道流量分析题。。。

No.	Time	Source	Destination	Protocol	Length	Info
28	55.307368	192.168.1.1	192.168.1.2	ICMP	114	Echo (ping) reply id=0x0001, s…
29	55.318549	192.168.1.2	192.168.1.1	ICMP	114	Echo (ping) request id=0x0001, s…
30	55.328988	192.168.1.1	192.168.1.2	ICMP	114	Echo (ping) reply id=0x0001, s…
31	55.339576	192.168.1.2	192.168.1.1	ICMP	114	Echo (ping) request id=0x0001, s…
32	55.340578	192.168.1.1	192.168.1.2	ICMP	114	Echo (ping) reply id=0x0001, s…
33	55.351160	192.168.1.2	192.168.1.1	ICMP	114	Echo (ping) request id=0x0001, s…
34	55.361168	192.168.1.1	192.168.1.2	ICMP	114	Echo (ping) reply id=0x0001, s…
35	59.995629	cc:00:06:ec:00:00	cc:00:06:ec:00:00	LOOP	60	Reply
36	60.913517	cc:01:06:ec:00:00	cc:01:06:ec:00:00	LOOP	60	Reply
37	62.292951	192.168.1.1	192.168.1.2	TCP	60	37465 → 23 [SYN] Seq=0 Win=4128 L…
38	62.313940	192.168.1.2	192.168.1.1	TCP	60	23 → 37465 [SYN, ACK] Seq=0 Ack=1…
39	62.334450	192.168.1.1	192.168.1.2	TCP	60	37465 → 23 [ACK] Seq=1 Ack=1 Win=…
40	62.335453	192.168.1.1	192.168.1.2	TELNET	63	Telnet Data ...
41	62.355984	192.168.1.1	192.168.1.2	TCP	60	[TCP Dup ACK 39#1] 37465 → 23 [AC…
42	62.355984	192.168.1.2	192.168.1.1	TELNET	66	Telnet Data ...
43	62.366482	192.168.1.1	192.168.1.2	TELNET	60	Telnet Data ...
44	62.366482	192.168.1.1	192.168.1.2	TELNET	60	Telnet Data ...
45	62.366482	192.168.1.1	192.168.1.2	TELNET	63	Telnet Data ...
46	62.376992	192.168.1.2	192.168.1.1	TELNET	96	Telnet Data ...

可以看到，有几种类型的协议，但是一眼望去，我想telnet这个协议，或许更具有吸引力，过滤一下协议，并且打开他的数据包，即：



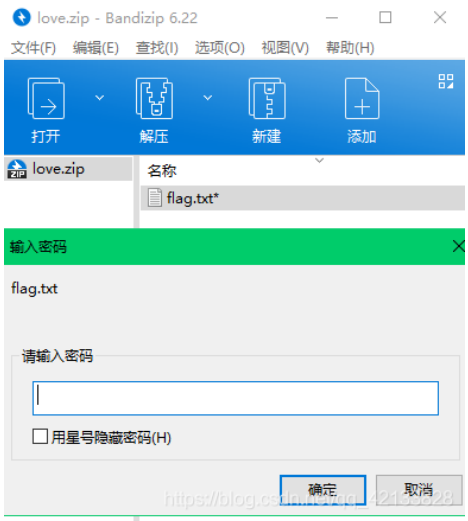
可以很清楚看到有Date这个参数，并且有值，在随意打开telnet的包几个看看，亦是如此，而且之都不一样，那么连续几个的呢？额。。小编想这里就不需要演示怎么一个个拼接成flag了吧，可以友情提示，他的flag在最后面的几个包内，别问我怎么知道的，因为小编就是一个个点出来的。。。。

7.女神的告白

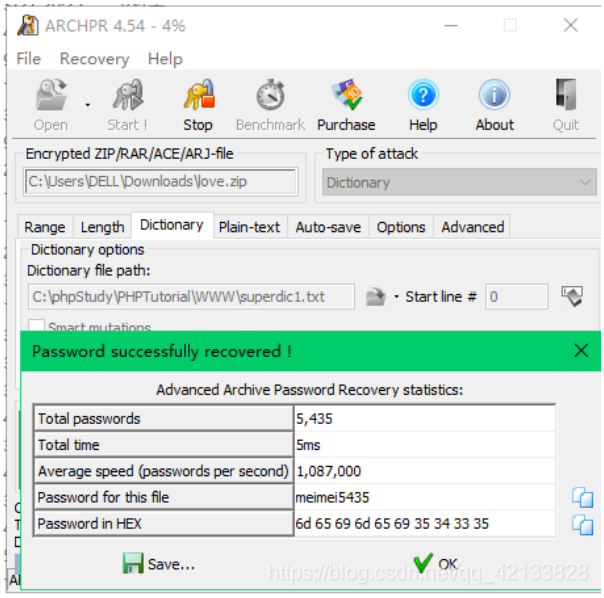
hint：李华的女神美美（meimei)给李华发了一个压缩包，却只告诉了李华压缩包密码是以她的名字开头，你能帮李华获得真爱吗？

提示的重点我想是，压缩包的解压密码开头几个是meimei

解压压缩包，即：

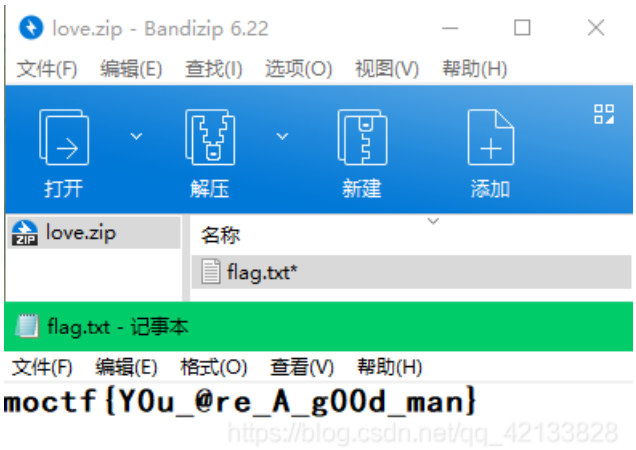


这是需要密码的，小编我是先使用一个字典生成器，先生成适应的密码字典，再用另一个工具爆破密码，话不多说，上图：



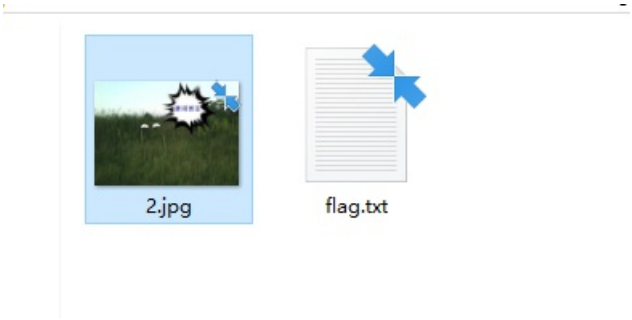
爆破的出来的密码，即：meimei5435

输入密码，解压文件得：



8.捉迷藏

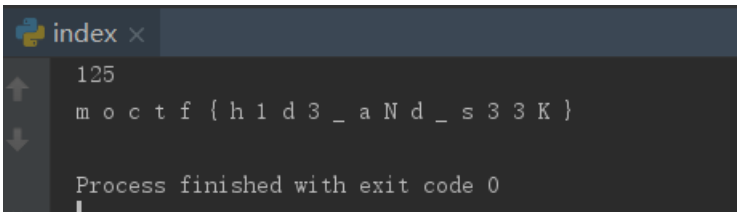
下载，并解压题目给的压缩包，可以看到：



打开flag.txt，里面有一串字符串，但是小编没解码出来，所以目标转为图片，先把图片foremat一下，并打开，即：

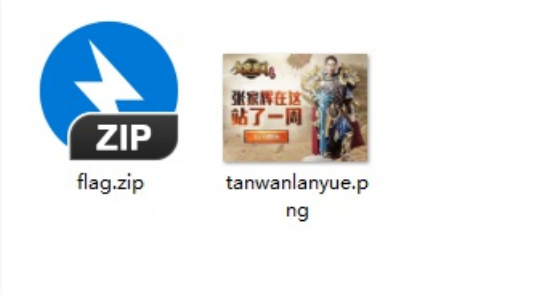
这个小编看得懂，一串ascii值，用python跑一下，即：

```
f = open('C:\\Users\DELL\\Desktop\\zhuomicang\\outfile\\zip\\flag.txt').read().split(' ')
a = []
n = 0
for l in f:
 print l
 a.append(int(l))
 n+=1
for i in range(n):
 print chr(a[i]),
```

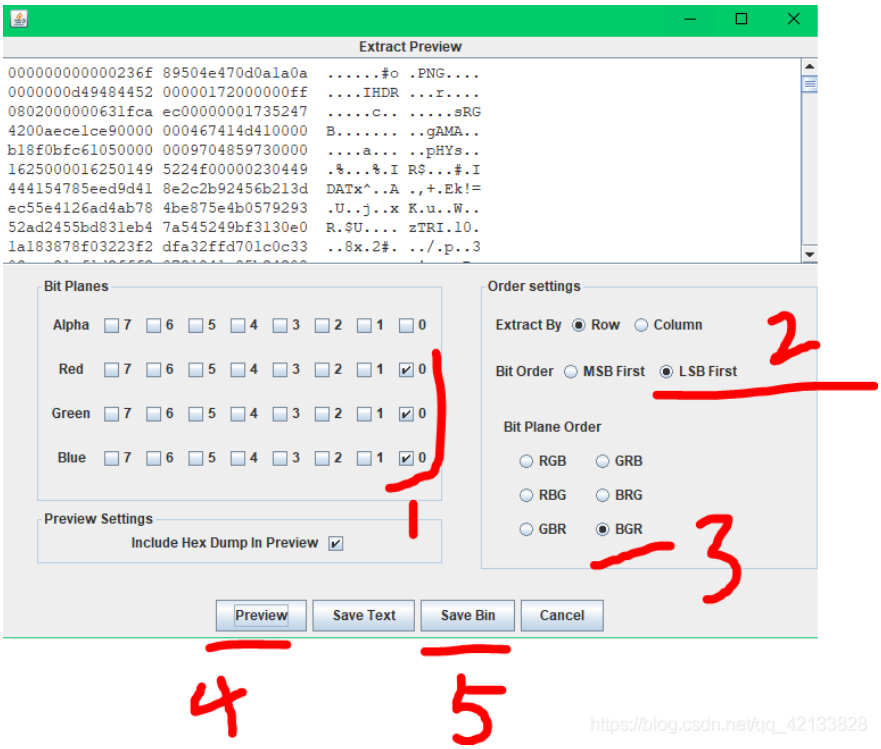


9.是兄弟就来干我

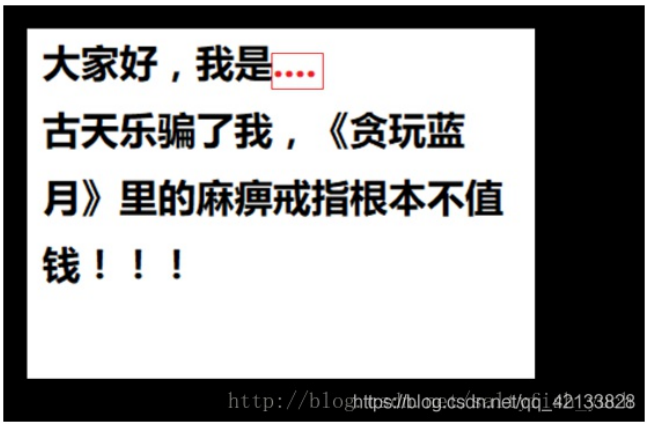
下载，并解压题目给的压缩包，可以看到：



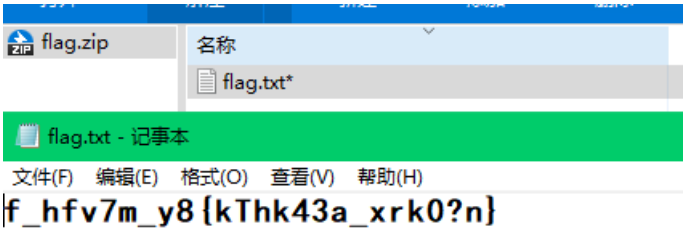
flag.zip打开里面的文件需要密码，则，先对图片分析一波，使用StegSolve里的Analyse里的Data Extract：



保存图片，并修改一下图片的值，删除头部多余的东西，打开图片，即：

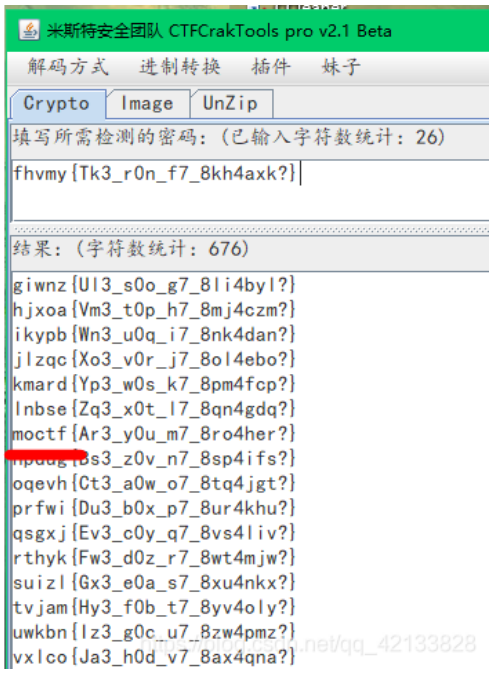


红色得地方，有提示 zhazhahui，我想这应该是密码吧，解密得：



栅栏密码解密得：

凯撒密码解密得：



10.百变flag

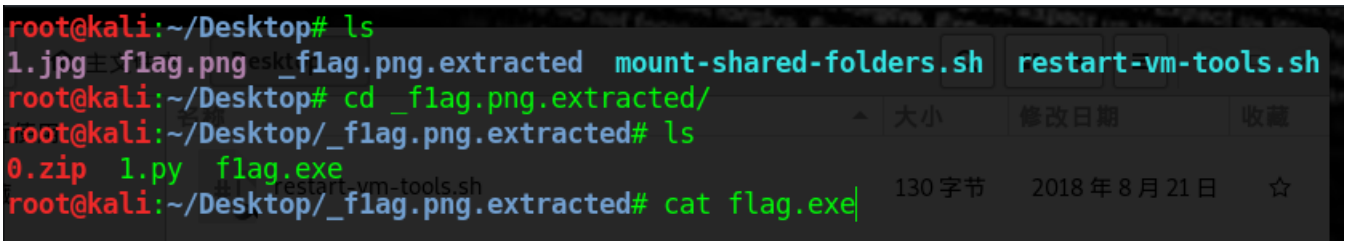
打开链接，发现只有一个小正方框，但是也不妨碍他是照片的事实，在离小方框远一点的黑色地方，点击保存图片。

此处使用的是kali完成的操作的。在window上有一些伪加密得文件打不开，在linux却可以直接解压。。。

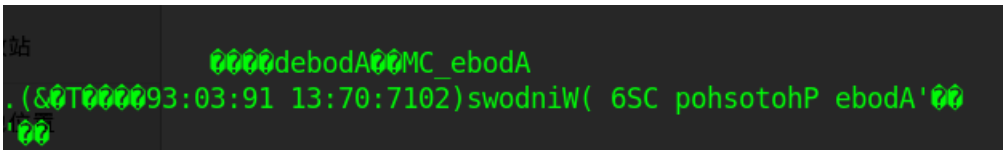
先在图片所在得目录下执行以下语句：

binwalk -e f1ag.png

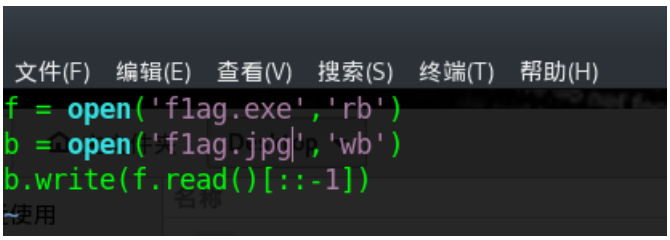
然后：



打开flag..exe文件查看：



如果你细心一点，你会发现他里面的东西倒过来了。所以，我们需要把她正回来，python代码如下：



运行后，是一张图片，打开，得：

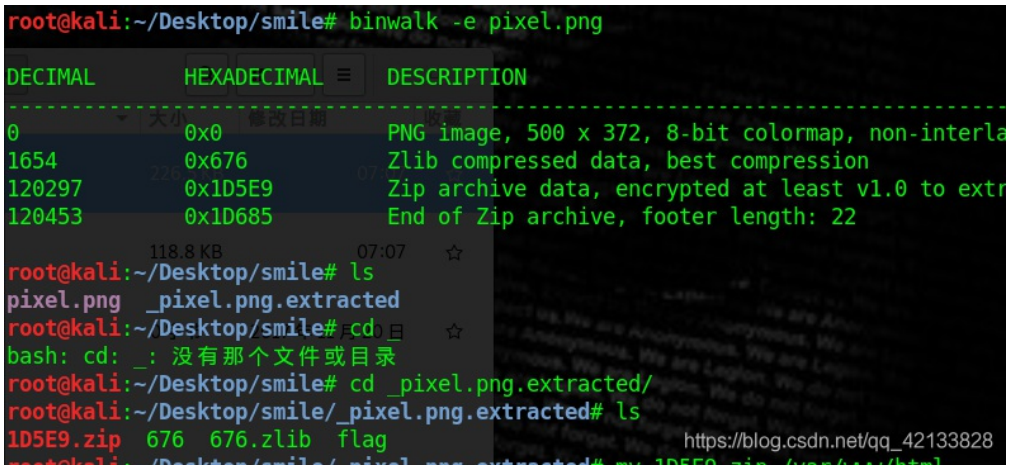
11.蒙娜丽萨的微笑

下载图片，使用StegSolve对图片进行分析，发现无果，在对图片修改大小，即：

`c2ltbGVpc2ludGVyaW5n`

base64解下得到：`simleisintering`

解出来一串字符，但不知道有啥用，那么再对图片分析一波

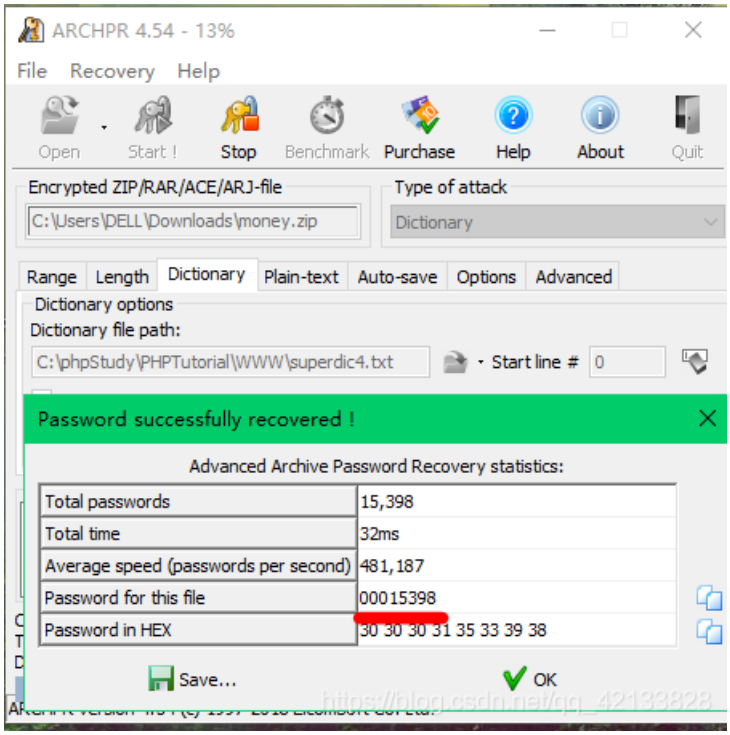


打开zip文件，输入刚才得到的密码，即：

12.李华的双十一

下载，并打开压缩包，会发现里面得两个文件都需要密码解密，意味着没有任何线索来获取密码

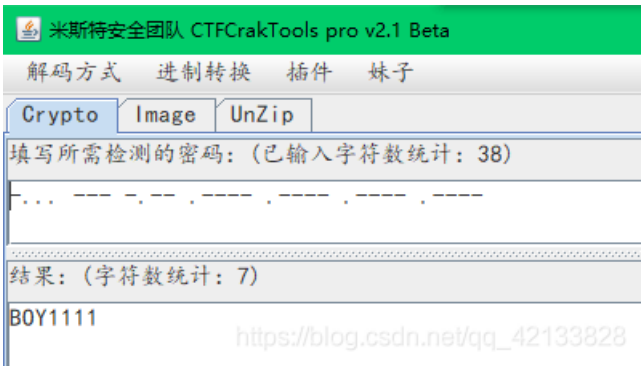
那么就只能粗暴一点，直接爆破出他的密码，即：



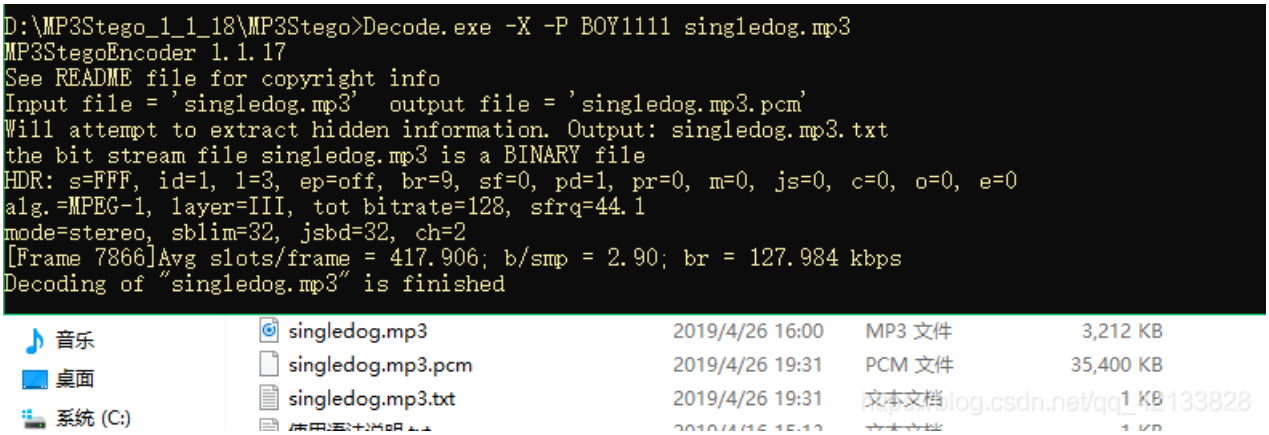
解密money.zip，并打开里面得文件，在文件的最底部，即：

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

摩斯密码解密得：



解压包里面还有一个mp3的文件，又得到密码，那么使用工具MP3Stego，运行以下命令，即：



打开singledog.txt，得一串base64的密文，解密得：

13.李华的疑惑

下载，并打开压缩包，可以看到一个password.txt，以及一个flag.zip（需密码）

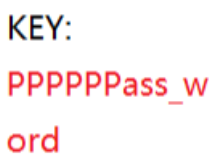
打开password.txt是一串字符串，使用python读取，每一行有三个字符，一共有22500行

说实话，小编一开始还是不知道的，去网上搜索才造，原来这是RGB的图像

所以，我们需要还原它，即：

```
from PIL import Image
x = 150
y = 150
im = Image.new("RGB",(x,y))
f = open('C:\\Users\\DELL\\Desktop\\yihuo\\password.txt')
for i in range(0,x):
 for j in range(0,j):
  l = f.readline()
  rgb = l.split(',')
  im.putpixel((i,j),(int(rgb[0]),int(rgb[1]),int(rgb[2])))
im.show()
```

得到的图像是这样的：



通过得到的key，去解压flag.zip,可得一串字符：

U2FsdGVkX18R9EylBVacP/j0XpClSh9nZth6TFwoh5GUv0edeVp3ZV9gXVqd/rlH66OlZgSHn2Mock4hcdqFEg=

进行AES解码（https://www.sojson.com/encrypt_aes.html），得：

moctf{D0_You_1ik3_tO_pAinH_wi4h_pi8e1}