

# 攻防世界-MISC-新手题解

原创

[老大的豆豆酱](#) 于 2020-08-25 13:52:04 发布 6046 收藏 33

分类专栏: [CTF misc](#) 文章标签: [信息安全](#) [base64](#)

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/weixin\\_43486981/article/details/108216639](https://blog.csdn.net/weixin_43486981/article/details/108216639)

版权



[CTF 同时被 2 个专栏收录](#)

27 篇文章 0 订阅

订阅专栏



[misc](#)

1 篇文章 0 订阅

订阅专栏

## 文章目录

PDF

[give\\_you\\_flag](#)

[坚持60s](#)

GIF

[掀桌子](#)

[如来十三掌](#)

[stegano](#)

[SimpleRAR](#)

[base64stego](#)

[ext3](#)

[功夫再高也怕菜刀](#)

PDF

题目描述：菜猫给了菜狗一张图，说图下面什么都没有  
附件下载下来是一个pdf图片

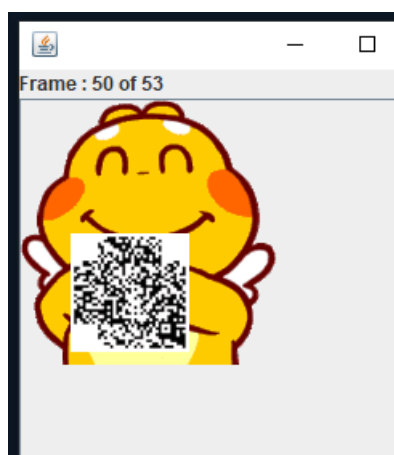


很明显是图片隐藏，根据提示应该图片下有什么东西  
把pdf放到linux中，用鼠标选中图片部分，会发现隐藏的信息显示出来了



**give\_you\_flag**

题目描述：菜狗找到了文件中的彩蛋很开心，给菜猫发了个表情包  
附件是一个gif动图，点开发现某一帧好像有个二维码  
使用stegsolve分析该gif（stegsolve中analyse下的frame browser）  
找到了有二维码的这一帧



但是这个二维码不完整。少了三个定位符，去网上随便截了一个，用美图秀秀拼了一下图



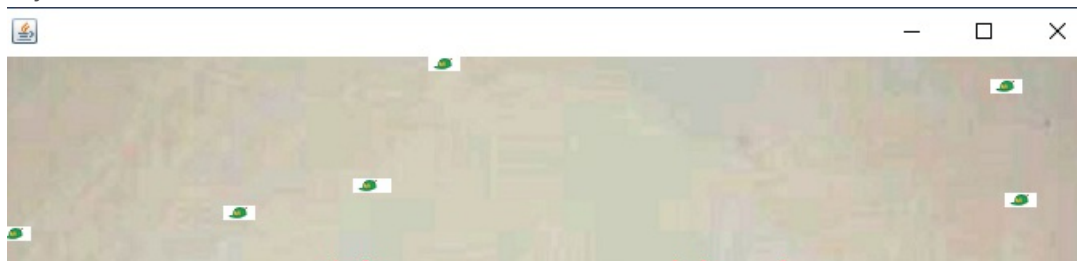
扫一下得到了flag

flag{e7d478cf6b915f50ab1277f78502a2c5}

## 坚持60s

题目描述：菜狗发现最近菜猫不爱理他，反而迷上了菜鸡

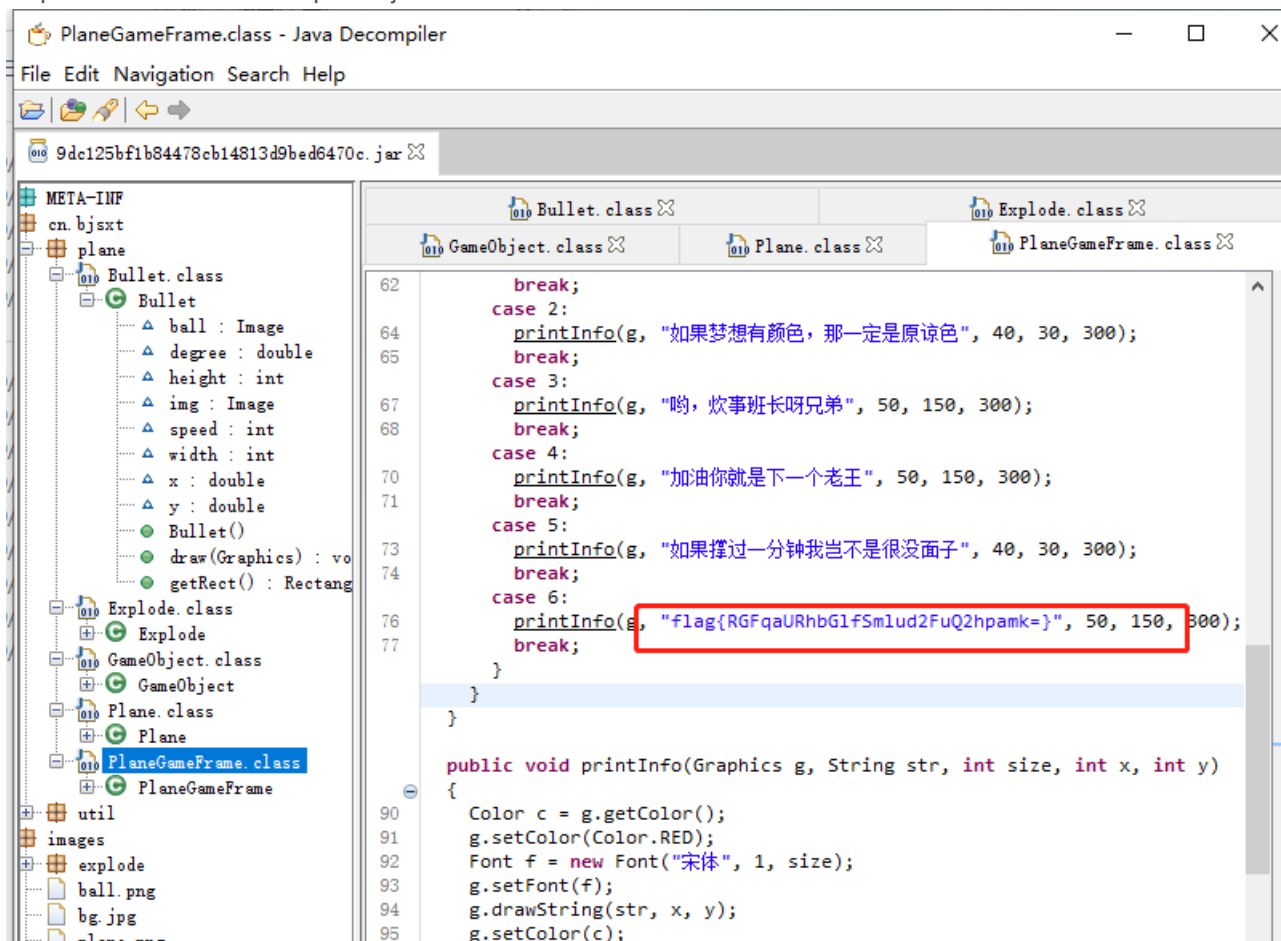
附件下载下来是个jar文件

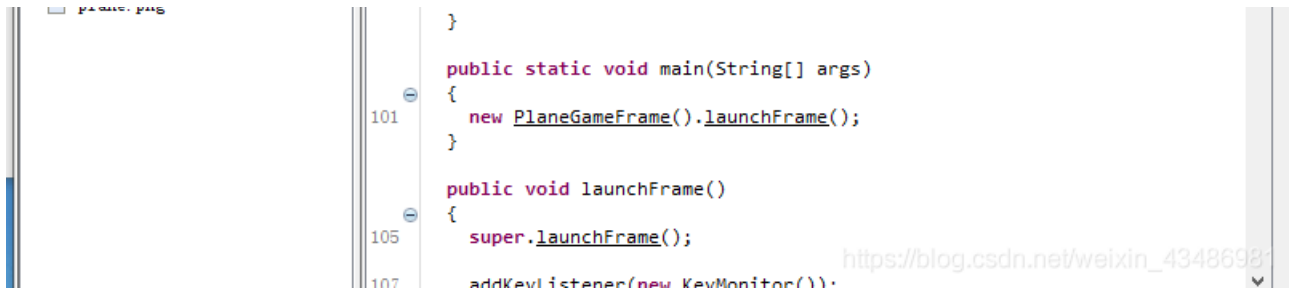




需要一个反编译器，查看小游戏的源代码。这里我用的是jd-gui。在源码中找到了flag。

flag{RGFqaURhbGlfSmlud2FuQ2hpamk=}



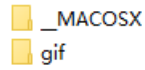


```
101 }  
102  
103 public static void main(String[] args)  
104 {  
105     new PlaneGameFrame().launchFrame();  
106 }  
107  
108 public void launchFrame()  
109 {  
110     super.launchFrame();  
111     addKeyListener(new KeyMonitor());  
112 }
```

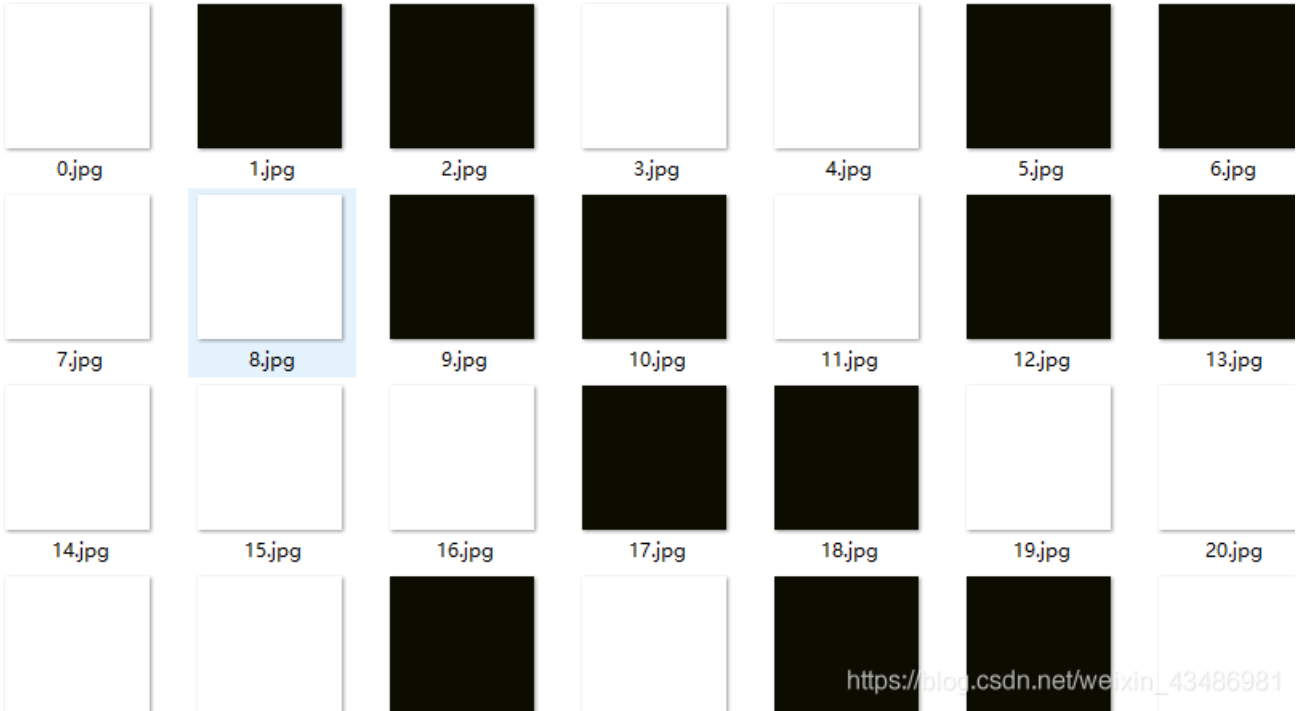
得到的flag是加密的，需要使用base64解密  
flag{DajiDali\_JinwanChiji}

**GIF**

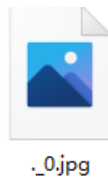
题目描述：菜狗截获了一张菜鸡发给菜猫的动态图，却发现另有玄机



gif文件夹下有104个这样的黑白图片



\_\_MACOSX/gif中有一个ipg图片，但是打不开



黑白图片和二进制中的1和0很像，于是黑图片为1，白图片为0，得到一串二进制数，将二进制转为字符，就能得到flag。  
二进制转文本在线工具

```
01100110011011000110000101100111011110110100011001110101010011100101111101100111011010010100011001111101
```

转换后的文本:

```
flag{FuN_giF}
```

[https://blog.csdn.net/weixin\\_43486981](https://blog.csdn.net/weixin_43486981)

## 掀桌子

题目描述：菜狗截获了一份报文如下

```
c8e9aca0c6f2e5f3e8c4efe7a1a0d4e8e5a0e6ece1e7a0e9f3baa0e8eafae3f9e4eafae2eae4e3eaebfabe3f5e7e9f3e4e3e8eaf9eaf3e2e4e6f2, 生气地掀翻了桌子(ノ°□°)ノ┌┴┴
```



题解:

获取了一个字符串, 一看只有0-9, a-f, 好是16进制啊。将16进制两两一组转成10进制。

两个一组发现最大的数是250, 最小的数是160。而ascii的可显示字符从32开始, ascii的最大为128, 猜想减去128, 刚好最小的到32。

```
1 a = 'c8e9aca0c6f2e5f3e8c4efe7a1a0d4e8e5a0e6ece1e7a0e9f3baa0e8eafae3:
```

```
1 import re
2 a = re.findall('.{2}', a)
3
4 a = [int(i, 16) for i in a]
5 print(max(a), ' ', min(a))
```

250 160

```
1 a = [chr(i-128) for i in a]
2 s = "".join(a)
3 print(s)
```

Hi, FreshDog! The flag is: hjzcydjzbdcjzkcugisdchjyjsbdf

[https://blog.csdn.net/weixin\\_43486981](https://blog.csdn.net/weixin_43486981)

```
a = 'c8e9aca0c6f2e5f3e8c4efe7a1a0d4e8e5a0e6ece1e7a0e9f3baa0e8eafae3f9e4eafae2eae4e3eaebfaebe3f5e7e9f3e4e3e8eaf9eaf3e2e4e6f2'

import re
a = re.findall('.{2}', a)

a = [int(i, 16) for i in a]
print(max(a), ' ', min(a))

a = [chr(i-128) for i in a]
s = "".join(a)
print(s)
```

flag为: flag{hjzcydjzbdcjzkcugisdchjyjsbdf}

如来十三掌

附件下载是个docx文档，打开啥也看不懂

夜哆悉諳多苦奢陀奢諦冥神哆盧穆幡三侄三即諸諳即冥迦冥隸數顛耶迦奢若吉怯陀諳怖奢智侄諸若奢數苦奢集遠俱老竟寫明奢若梵等盧幡豆蒙密離怯婆幡礙他哆提哆多鉢以南哆心曰姪罰蒙呐神。舍切真怯勝呐得俱沙罰娑是怯遠得呐數罰輸哆遠薩得槃漫夢盧幡亦醯呐娑幡瑟輸諳尼摩罰薩冥大倒參夢侄阿心罰等奢大度地冥殿幡沙蘇輸奢恐豆侄得罰提哆伽諳沙楞鉢三死怯摩大蘇者數一遮

看别人的教程说是佛语，使用在线工具把佛语转成字符串

MzkuM3gvMUAwnzuvn3cgozMIMTuvqzAenJchMUAeqzWenzEmLJW9

## 与佛论禅

MzkuM3gvMUAwnzuvn3cgozMIMTuvqzAenJchMUAeqzWenzEmLJW9

听佛说宇宙的真谛

参悟佛所言的真意

普度众生

心不动，万物皆不动

佛曰：夜哆悉諳多苦奢陀奢諦冥神哆盧穆幡三侄三即諸諳即冥迦冥隸數顛耶迦奢若吉怯陀諳怖奢智侄諸若奢數苦奢集遠俱老竟寫明奢若梵等盧幡豆蒙密離怯婆幡礙他哆提哆多鉢以南哆心曰姪罰蒙呐神。舍切真怯勝呐得俱沙罰娑是怯遠得呐數罰輸哆遠薩得槃漫夢盧幡亦醯呐娑幡瑟輸諳尼摩罰薩冥大倒參夢侄阿心罰等奢大度地冥殿幡沙蘇輸奢恐豆侄得罰提哆伽諳沙楞鉢三死怯摩大蘇者數一遮

[https://blog.csdn.net/weixin\\_43486981](https://blog.csdn.net/weixin_43486981)

这个“十三”对应的是另一种叫做“ROT13”的编码。ROT13编码后得到一串代码。ROT13编码在线工具

ZmxhZ3tiZHNjamhia3ptbmZyZGhidmNraWpuZHNrdmJramRzYWJ9

再使用base64解码一次得到flag

flag{bdscjhbkmzfrdhbvcikjndskvbkjdsab}

## stegano

题目描述：菜狗收到了图后很开心，玩起了pdf 提交格式为flag{xxx}，解密字符需小写

附件下载了一个pdf，打开是一篇英文文章。

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Cras faucibus odio ut metus vulputate, id laoreet magna volutpat. Integer nec enim vel arcu porttitor egestas. Vestibulum suscipit lorem sed sem faucibus rutrum. Nunc diam orci, convallis vitae auctor vehicula, interdum ut mi. Maecenas nec urna at dolor mattis dictum sit amet at orci. Mauris condimentum adipiscing erat nec feugiat. Curabitur scelerisque varius ligula, iaculis adipiscing dui. Duis eget ullamcorper arcu. In facilisis et tortor commodo aliquam. Nulla feugiat, sem eu molestie bibendum, leo nisi porttitor massa, id accumsan sapien libero id tellus. In enim lacus, sollicitudin a felis quis, blandit porta ipsum. Donec sed nibh egestas, tristique mauris eu, rutrum justo. Nulla facilisi. Duis gravida semper dui laoreet vulputate. Aenean quis tempor orci. Cras placerat lectus nulla, eu bibendum metus interdum in. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Cras faucibus odio ut metus vulputate, id laoreet magna volutpat. Integer nec enim vel arcu porttitor egestas. Vestibulum suscipit lorem sed sem faucibus rutrum. Nunc diam orci, convallis vitae auctor vehicula, interdum ut mi. Maecenas nec urna at dolor mattis dictum sit amet at orci. Mauris condimentum adipiscing erat nec feugiat. Curabitur scelerisque varius ligula, iaculis adipiscing dui. Duis eget ullamcorper arcu. In facilisis et tortor commodo aliquam. Nulla feugiat, sem eu molestie bibendum, leo nisi porttitor massa, id accumsan sapien libero id tellus. In enim lacus, sollicitudin a felis quis, blandit porta ipsum. Donec sed nibh egestas, tristique mauris eu, rutrum justo. Nulla facilisi. Duis gravida semper dui laoreet vulputate. Aenean quis tempor orci. Cras placerat lectus nulla, eu bibendum metus interdum in. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Cras faucibus odio ut metus vulputate, id laoreet magna volutpat. Integer nec enim vel arcu porttitor egestas. Vestibulum suscipit lorem sed sem faucibus rutrum. Nunc diam orci, convallis vitae auctor vehicula, interdum ut mi. Maecenas nec urna at dolor mattis dictum sit amet at orci. Mauris condimentum adipiscing erat nec feugiat. Curabitur scelerisque varius ligula, iaculis adipiscing dui. Duis eget

43486981



用谷歌浏览器打开pdf文件，CTRL+A -> CTRL+C，粘贴到txt文件中。发现开头有一段AB编码，看起来像摩尔斯点码

文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)

NoFlagHere! NoFlagHere! NoFlagHere!

XX  
XXXX

XX  
XXXXXXXXXX Close - but still not here !

BABA BBB BA BBA ABA AB B AAB ABAA AB B AA BBB BA AAA BBAABB  
AABA ABAA AB BBA BBBAAA ABBBB BA AAAB ABBBB AAAAA ABBBB  
BAAA ABAA AAABB BB AAABB AAAAA AAAAA AAAAB BBA AAABB  
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Cras faucibus odio ut  
metus vulputate, id laoreet magna

[https://blog.csdn.net/weixin\\_43486981](https://blog.csdn.net/weixin_43486981)

把A换成.,B换成\_得到一个串

```
1 a = 'BABA BBB BA BBA ABA AB B AAB ABAA AB B AA BBB BA AAA BBAABB AABA AB BBA BBB'
```

```
1 a = a.replace("A",'.')
2 a = a.replace("B","_")
3 a = a.replace(" ",'/')
4 a
```

'-.\_./---/./--./.-./-/-/..-./.../-/-/.../---/./.../---..--/..-./.-./-/-/---.../.-  
---/./...-/..----/...../..----/-.../..-...--/--/...--/...../...../.....-/--/...--'

[https://blog.csdn.net/weixin\\_43486981](https://blog.csdn.net/weixin_43486981)

解密摩斯密码得到flag

CONGRATULATIONS,FLAG:1NV151BL3M3554G3

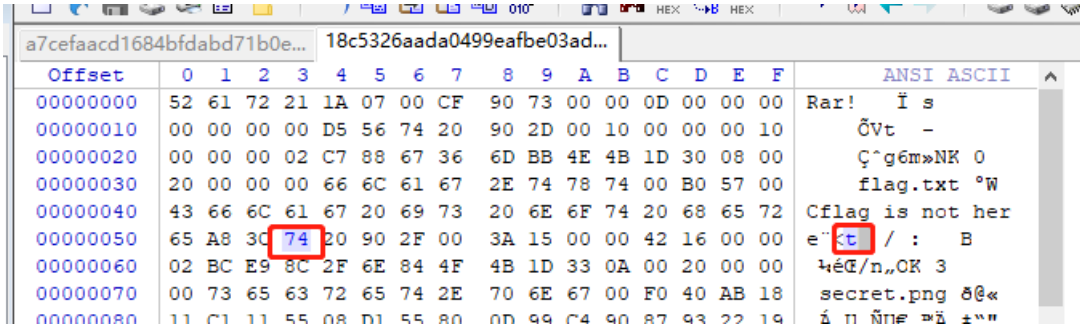
SimpleRAR

题目描述：菜狗最近学会了拼图，这是他刚拼好的，可是却搞错了一块(ps:双图层)

附件下载解码压缩包出错

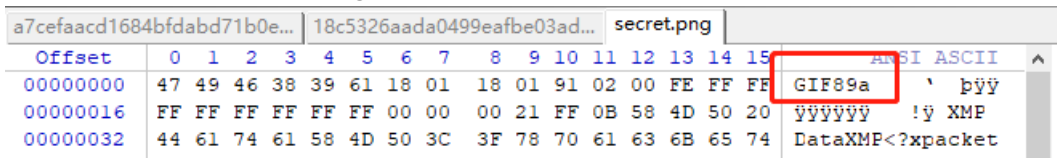


用winhex打开，rar对png的文件类型编码是74，就在flag.txt文件结束，原来是7A，改成了74.



解压得到了一个什么也没有的图片。

用winhex打开该图片看看，发现图片文件格式为gif



把图片后缀改为gif，用stegsolve打开看看，发现半截二维码，他的另一半应该在另一半图片里



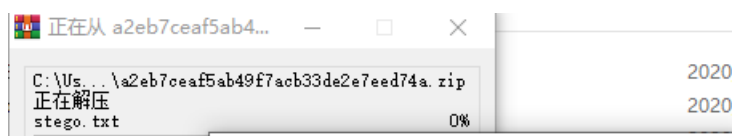
另一半我也不知道在哪 哭了

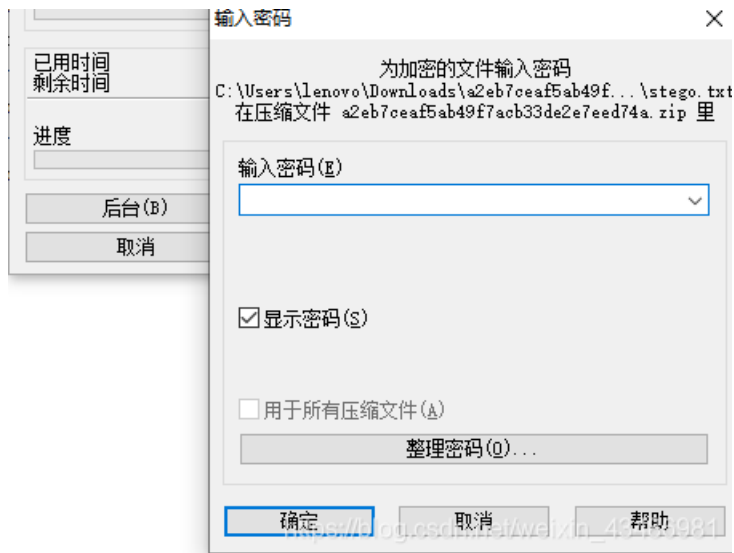
看别人的教程，找到的另一半是残缺的，需要拼图，拼好之后扫描就得到了flag

## base64stego

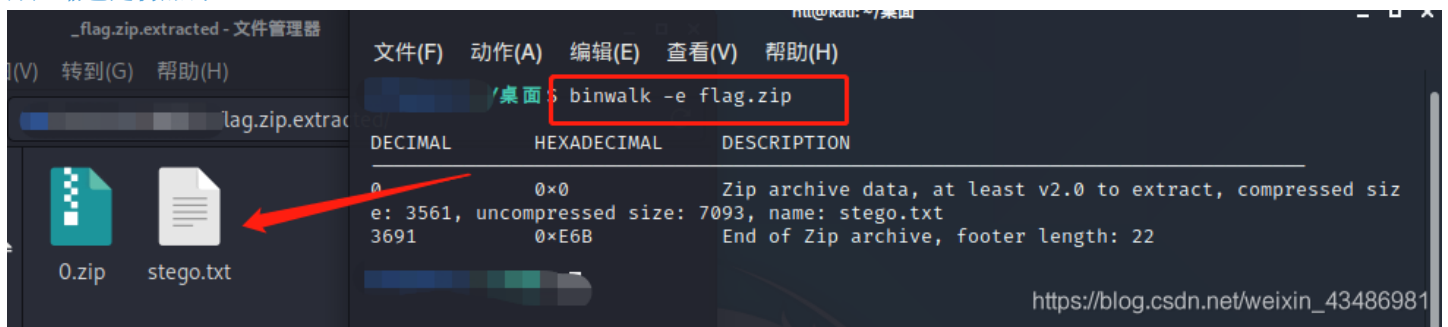
题目描述：菜狗经过几天的学习，终于发现了如来十三掌最后一步的精髓

附件下载后解压需要密码：

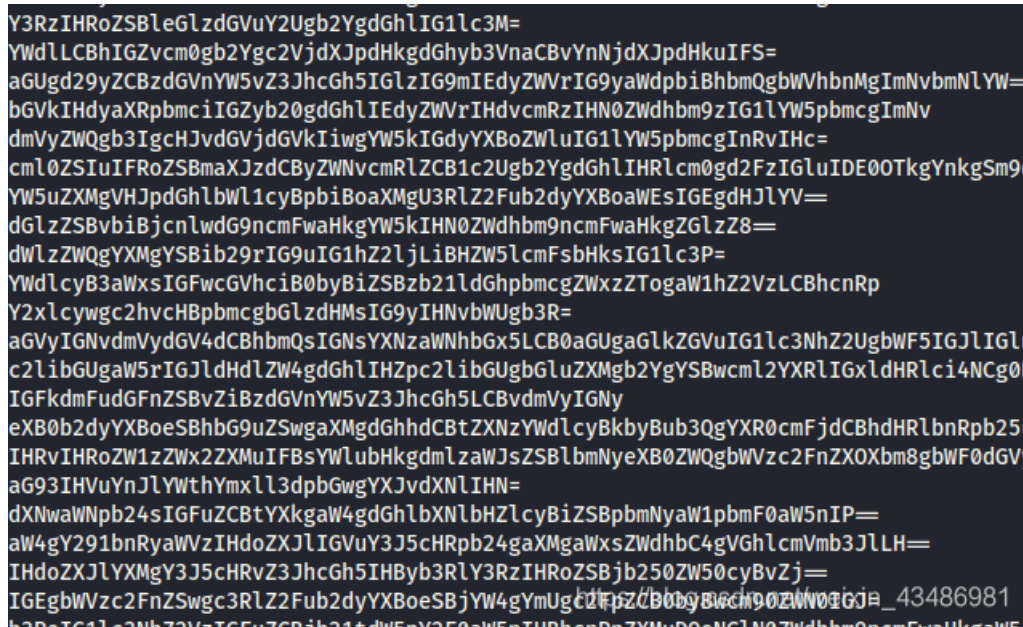




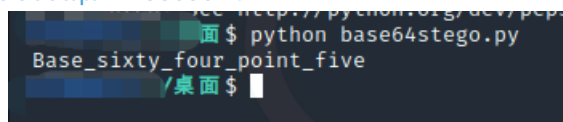
把文件复制到linux中，使用binwalk工具，`binwalk -e flag.zip`  
原压缩包是伪加密。



得到一个stego.txt文件，打开是一堆字符串



判断是使用的Base64隐写，直接使用py代码解密，得到flag: flag{Base\_sixty\_four\_point\_five}  
py代码: <https://www.cnblogs.com/sesefadou/p/11788090.html>



ext3

题目描述：今天是菜狗的生日，他收到了一个linux系统光盘  
在linux中使用 `file` 命令查看文件，发现是ext3类型

```
h11@kali:~/桌面$ file f1fc23f5c743425d9e0073887c846d23
f1fc23f5c743425d9e0073887c846d23: Linux rev 1.0 ext3 filesystem data, UUID=cf6d7bff-c377-403f-84ae-956ce3c99aaa
h11@kali:~/桌面$
```

ext3就是Linux的一个文件系统，strings查看一下有没有flag这样的字符串

```
strings f1fc23f5c743425d9e0073887c846d23 | grep flag
```

```
~/桌面$ strings f1fc23f5c743425d9e0073887c846d23 | grep flag
.flag.txt.swp
Flag.txtt.swx
~root/Desktop/file/07avZhikgKgbF/flag.txt
.flag.txt.swp
flag.txtt.swx
.flag.txt.swp
Flag.txtt.swx
h11@kali:~/桌面$
```

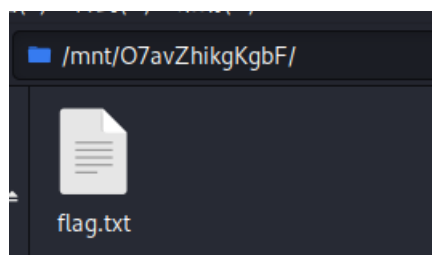
发现一个flag.txt文件

flag应该就在这个flag.txt中了，把这个文件系统挂载到Linux上

```
mount 文件名 /mnt
```

```
h11@kali:~/桌面$ sudo mount f1fc23f5c743425d9e0073887c846d23 /mnt
[sudo] h11 的密码:
```

挂上去之后看一下/mnt/下的文件，用命令 `ls -al /mnt/`，可以看到上面strings查找到的O7avZhikgKgbF，flag.txt就在这个目录里，



得到文件内容为：ZmxhZ3tzYWpiY2lienNrampjbmJoc2J2Y2pianN6Y3N6Ymt6an0=，base64解码即可。

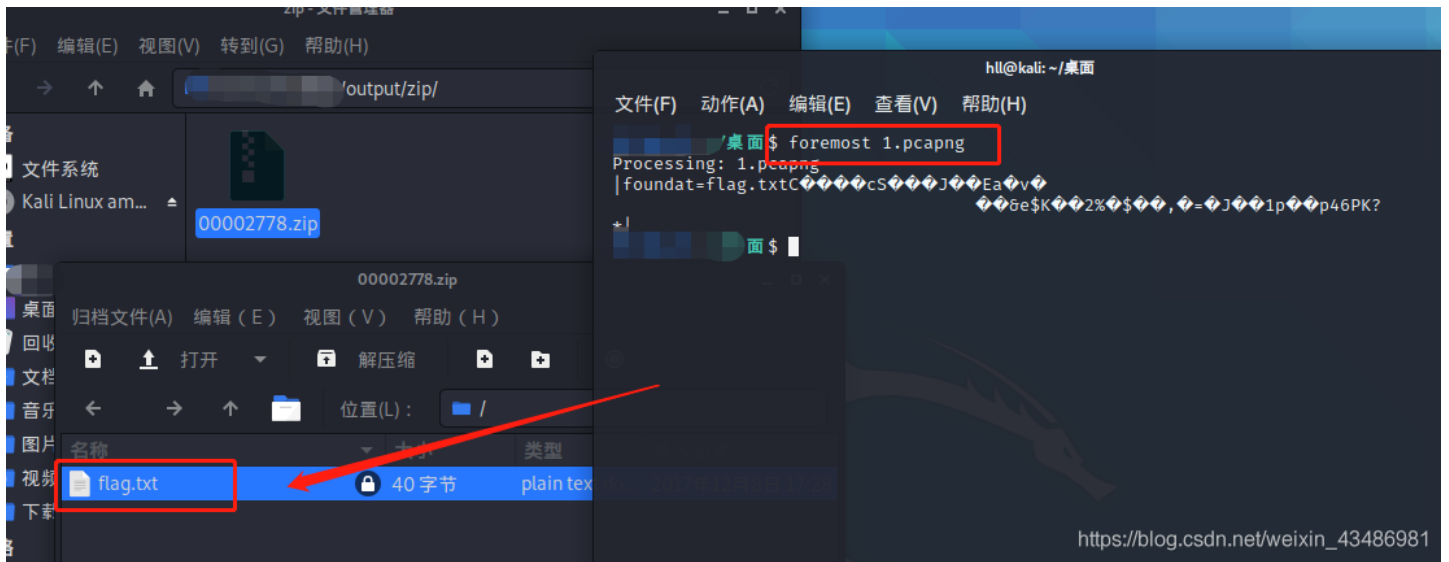
## 功夫再高也怕菜刀

附件是一个pcapng格式的文件，是wireshark抓包的文件格式。

使用foremost可以分离出一个有密码的压缩包。里面就是flag文件。我们接下来要做的就是寻找密码

kali中已经预安装了foremost，使用命令apt-get install foremost进行安装，即可使用

```
hll@kali: ~$ sudo apt-get install foremost
[sudo] hll 的密码:
正在读取软件包列表 ... 完成
正在分析软件包的依赖关系树
正在读取状态信息 ... 完成
下列【新】软件包将被安装：
  foremost
升级了 0 个软件包，新安装了 1 个软件包，要卸载 0 个软件包，有 0 个软件包需要下载 42.1 kB 的归档。
解压缩后会消耗 103 kB 的额外空间。
获取:1 http://mirrors.neusoft.edu.cn/kali kali-rolling/main amd64 foremost 1.5.7-9+b1 [42.1 kB]
已下载 42.1 kB，耗时 1秒 (37.7 kB/s)
正在选中未选择的软件包 foremost。
(正在读取数据库 ... 系统当前共安装有 288321 个文件和目录。)
准备解压 ... /foremost_1.5.7-9+b1_amd64.deb ...
正在解压 foremost (1.5.7-9+b1) ...
正在设置 foremost (1.5.7-9+b1) ...
正在处理用于 man-db (2.9.3-2) 的触发器 ...
正在处理用于 kali-menu (2020.3.2) 的触发器 ...
```



使用foremost可以分离出一个有密码的压缩包。里面就是flag文件。我们接下来要做的就是寻找密码





使用wireshake打开该pcapng文件，分组字节流查找flag关键字

The screenshot displays the Wireshark interface with the following details:

- Packet List:** Packet 1314, Time 65.547155778, Source 192.168.43.83, Destination 192.168.25.128, Protocol HTTP, Length 515. Details: HTTP/1.1 200 OK (text/html).
- Packet Details:** Internet Protocol Version 4, Src: 192.168.43.83, Dst: 192.168.25.128; Transmission Control Protocol, Src Port: 80, Dst Port: 47862, Seq: 2377, Ack: 1921, Len: 461; Hypertext Transfer Protocol; Line-based text data: text/html (7 lines).

```
->|./\t2017-12-08 11:42:11\t0\t0777\n
..\t2017-12-08 11:39:10\t4096\t0777\n
1.php\t2017-12-08 11:33:16\t33\t0666\n
6666.jpg\t2017-12-08 11:42:11\t102226\t0666\n
flag.txt\t2017-12-08 11:35:29\t17\t0666\n
```
- Packet Bytes:** Hexadecimal and ASCII representation of the packet data. A red box highlights the ASCII sequence 'flag.txt' at offset 00b0.

右键该分组追踪TCP流，  
看到内容像是16进制数据  
翻到最后发现一个jpg文件

The screenshot shows the raw data of the selected packet in hexadecimal and ASCII. A red box highlights the ASCII sequence 'flag.txt' at offset 00b0.

在追踪到第7个流时，观察十六进制头，有FFD8表示，说明是jpg文件的开头

jpg格式是以：FFD8FF开头，以FFD9结尾的







[创作打卡挑战赛](#) >

[赢取流量/现金/CSDN周边激励大奖](#)