CTF-攻防世界-MISC-高手进阶区（001-018 持续更新中...）

原创

beglage 于 2020-08-22 20:09:09 发布 2906 收藏 10
分类专栏： CTF 文章标签： 信息安全 网络安全 CTF MISC

本文链接：https://blog.csdn.net/weixin_43252204/article/details/108173366

马上就要进军CTF了，准备把MISC，CRYPTO，WEB的题目好好刷一下．．．

目录

MISC 高手进阶区

base64÷4

下载题目附件，文本中只有一串字符串

> 666C61677B4533334237464438413342383431434139363939454444424132344236304141 7D

题目提示base64÷4=base16，尝试使用base16解码

法一：编写python脚本

```
import base64

test="666C61677B4533334237464438413342383431434139363939454444424132344236304141 7D"

flag=base64.b16decode(test)

print(flag)
```

法二：使用在线解码平台

https://www.qqxiuzi.cn/bianma/base.php?type=16

flag:

> **flag{E33B7FD8A3B841CA9699EDDBA24B60AA}**

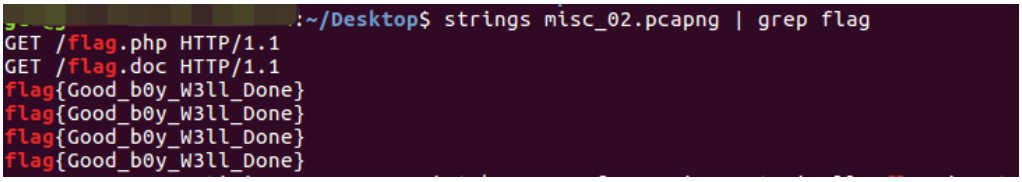
> Base16编码是一个标准的十六进制字符串，它并不包含任何控制字符，以及Base64和Base32中的“=”符号。

embarrass

下载zip附件，解压之后打开文件夹，发现文件misc_02.pcapng

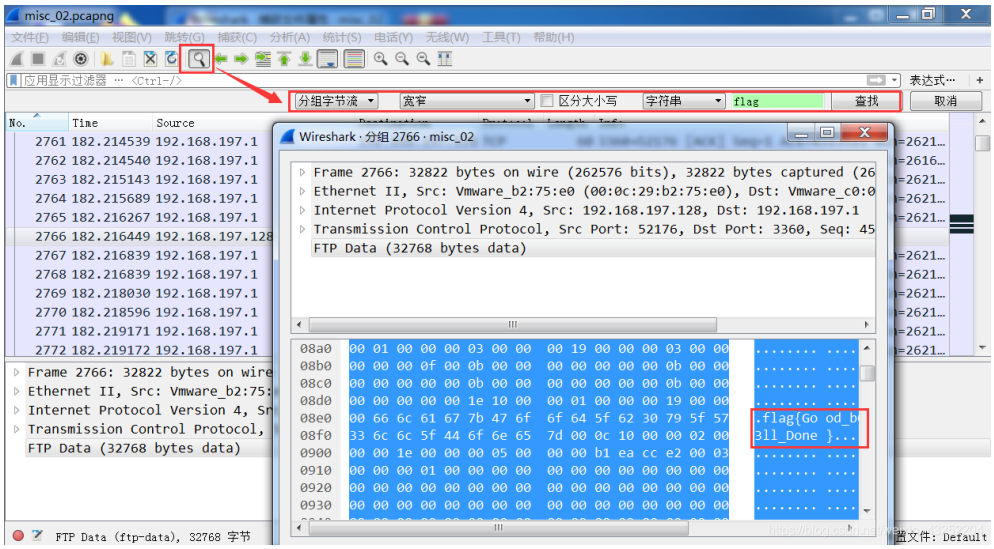
法一：使用linux系统下的string搜索命令 查找文件中flag

> strings 文件名 | grep flag



文件类型：Wireshark capture file (.pcapng)

法二：使用Wireshark 打开该文件，使用搜索功能搜索字符串flag



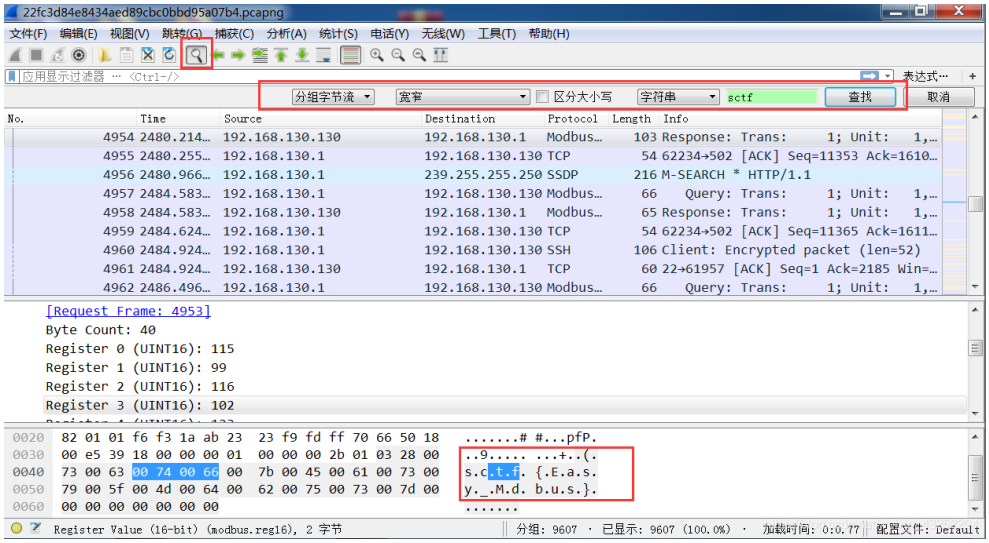
flag:

> flag{Good_b0y_W3ll_Done}

神奇的Modbus

题目描述：寻找flag,提交格式为sctf{xxx}

发现还是.pcapng的文件，继续使用Wireshark打开，搜索 sctf字符串



得到flag：sctf{Easy_Mdbus}

失败了！！！

发现题目名称是“神奇的Modbus”，加上字母o，sctf{Easy_Modbus}，尽然过了，不懂，应该是题目本身的问题

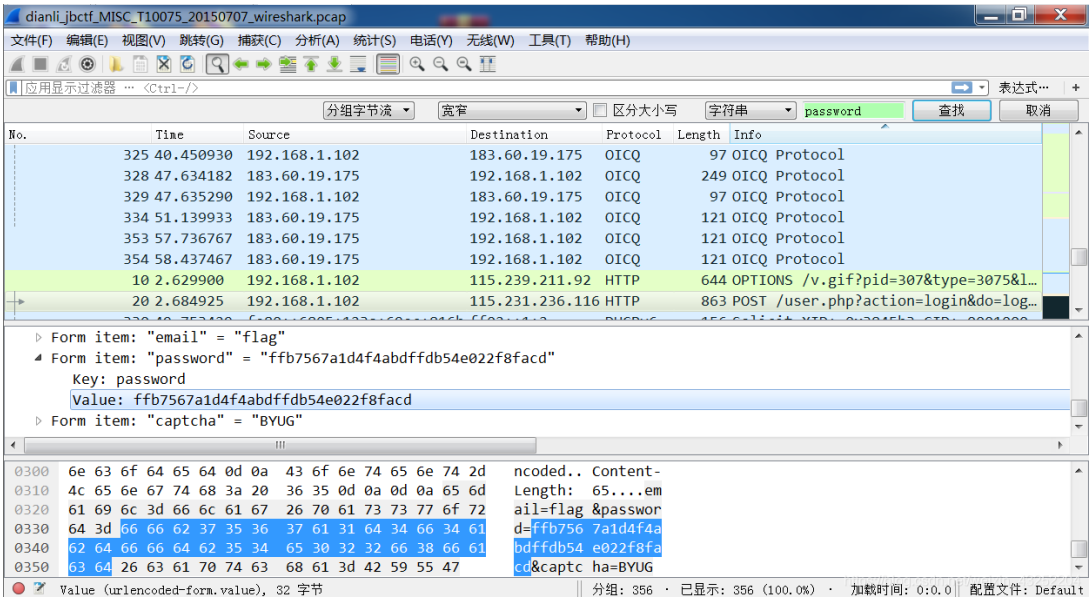
flag:

> sctf{Easy_Modbus}

wireshark-1

题目描述：黑客通过wireshark抓到管理员登陆网站的一段流量包（管理员的密码即是答案)。 flag提交形式为flag{XXXX}

继续用Wireshark打开该文件，题目给出提示管理员密码，尝试搜索字符串password



flag:

> flag{ffb7567a1d4f4abdffdb54e022f8facd}

pure_color

下载附件，得到一张图片，空白，可能使用了图片隐写

这里需要用到StegSolve工具，将隐写在图片中的内容显示出来

下载地址：

> http://www.caesum.com/handbook/Stegsolve.jar（需要java环境）

flag:

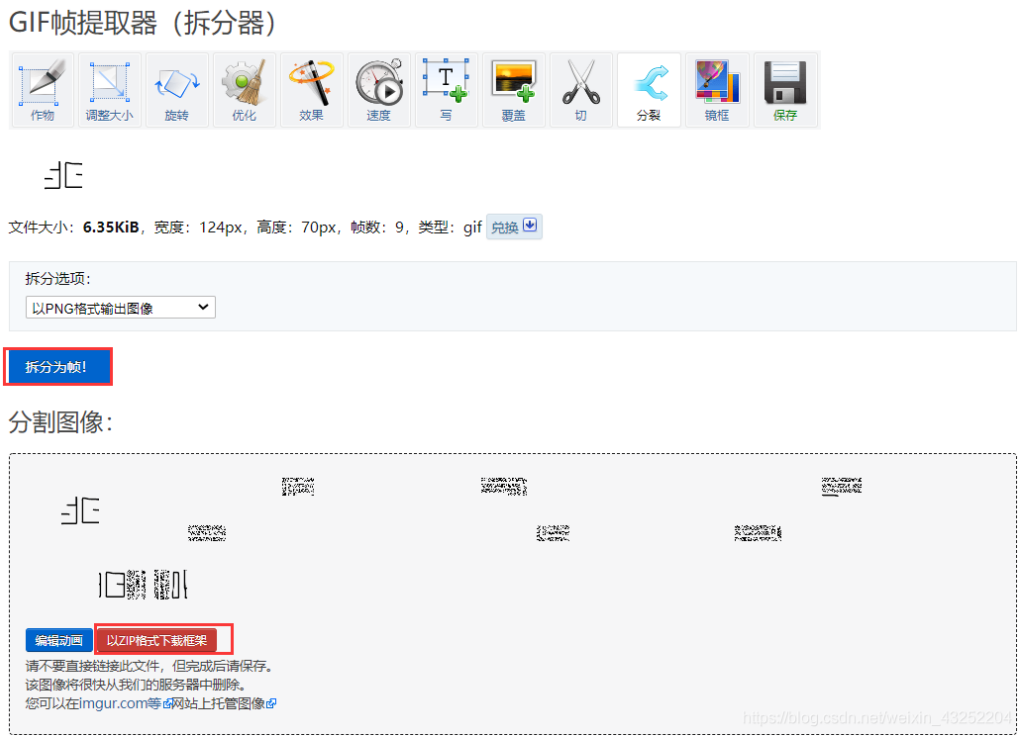
> flag{true_steganographers_doesnt_need_any_tools}

Aesop_secret

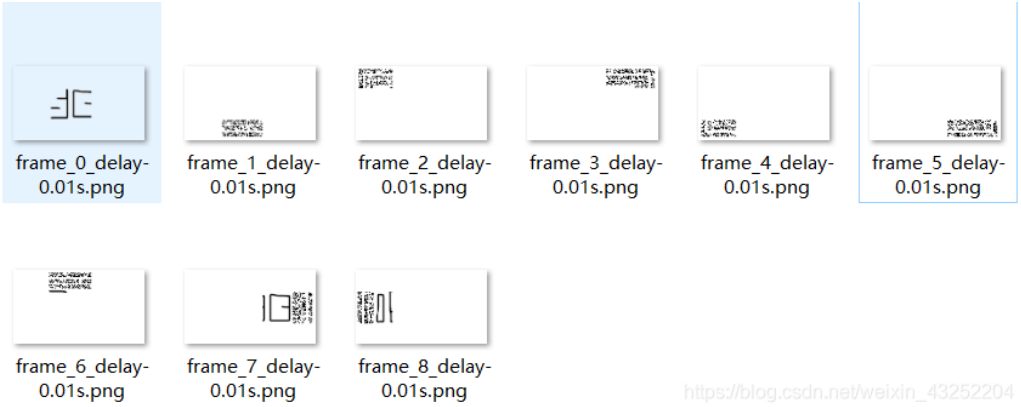
下载附件，得到一张gif图片，考虑对该gif图片进行帧拆分

用gif分解网站分解

> http://ezgif.com/split

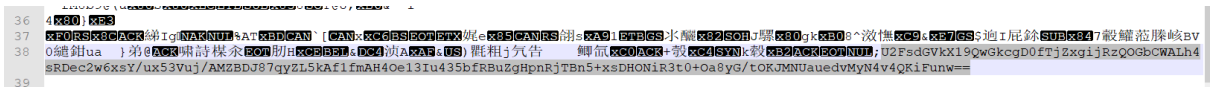


得到8张图片，明显可以拼接成一张图



使用截图和画图板进行图片拼接，得到以下图像

是四个字母**ISCC**，貌似也没有关于flag的相关信息，使用winhex或者记事本打开该gif图片，分析其源码，发现在文本的结尾有一串类似编码的字符串




```
U2FsdGVkX19QwGkcgD0fTjZxgijRzQOGbCWALh4sRDec2w6xsY/ux53Vuj/AMZBDJ87qyZL5kAf1fmAH4Oe13Iu435bfRE
```

尝试对其进行base-64解码，得到的结果乱码

看题目名称“Aesop_secret”猜测可能是

aes解密：https://www.sojson.com/encrypt_aes.html

密钥前面得到的ISCC，两次解密之后得到flag

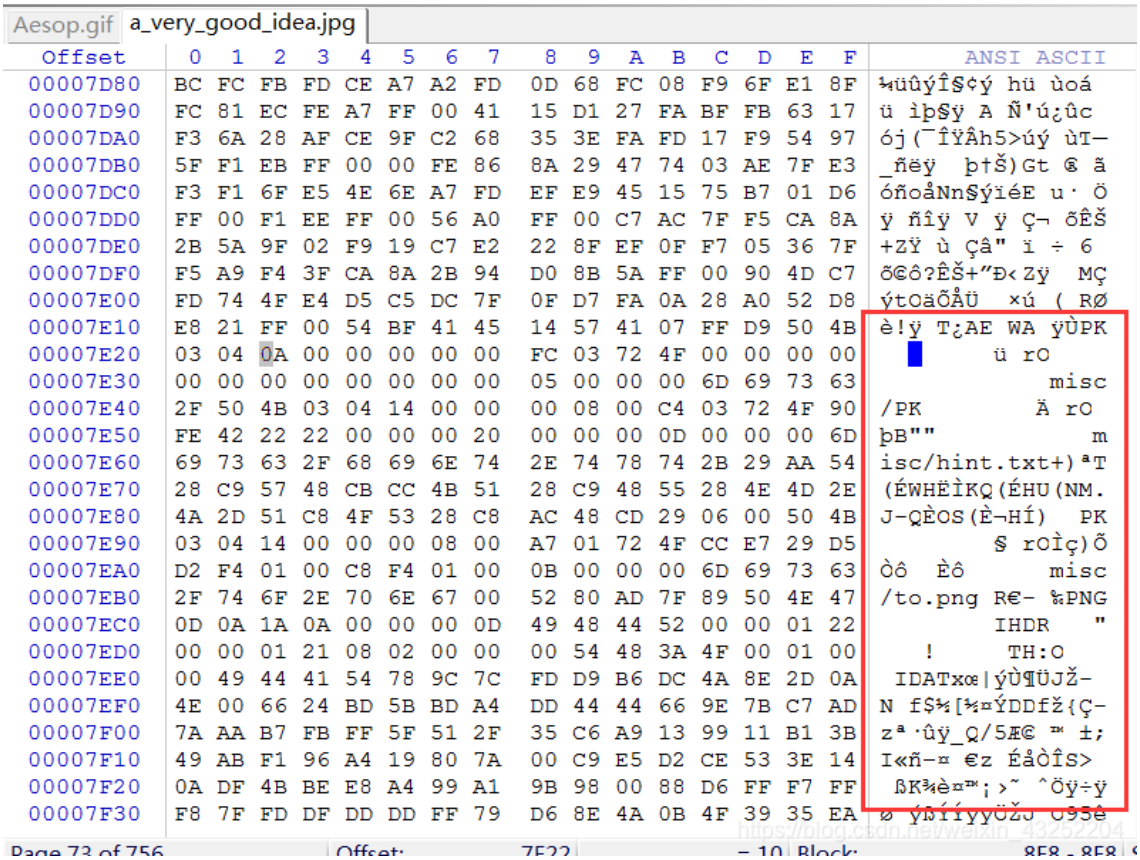
flag:

flag{DugUpADiamondADeepDarkMine}

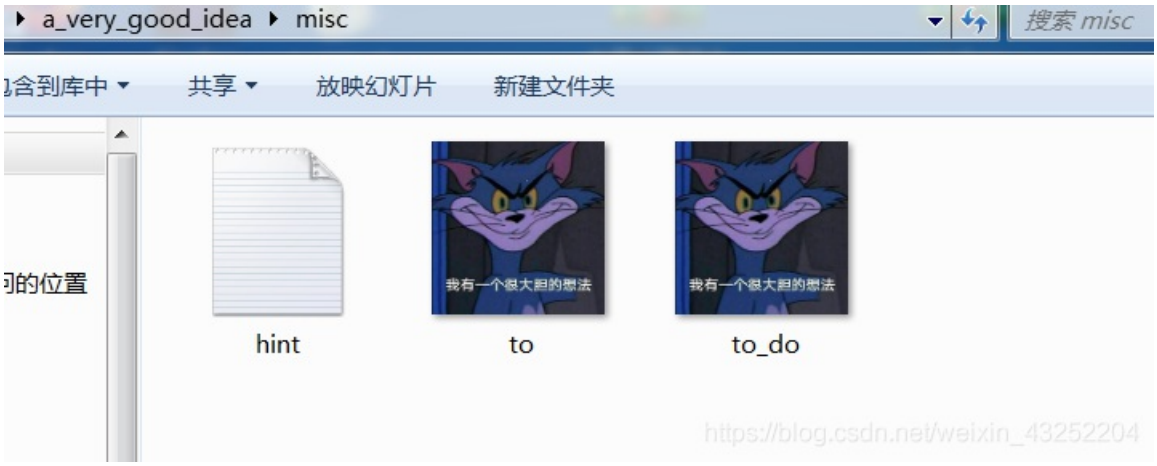
a_good_idea

下载附件，解压之后发现是一张图片

使用winhex分析该图片，发现存在压缩文件，更改后缀名为.zip



解压之后，得到以下文件，其中有一个暗时的文本文件



> 提示：try to find the secret of pixels（试着找出像素的秘密）

使用Stegsolve的combin功能分析这两张图片

打开第二张to_do.png图片，之后点击image Combiner，选择第一张图片to.png

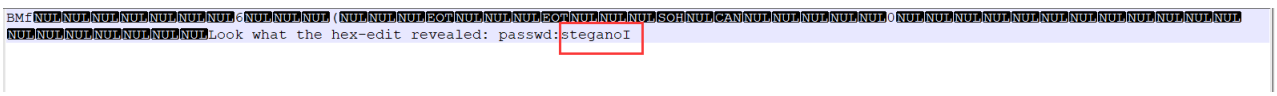
连续更换，得到两图相减的图像，之后得到一张二维码，手机扫码即可得到flag

flag:

> NCTF{m1sc_1s_very_funny!!!}

Training-Stegano-1

下载附件，得到一张图片，查看图片源码



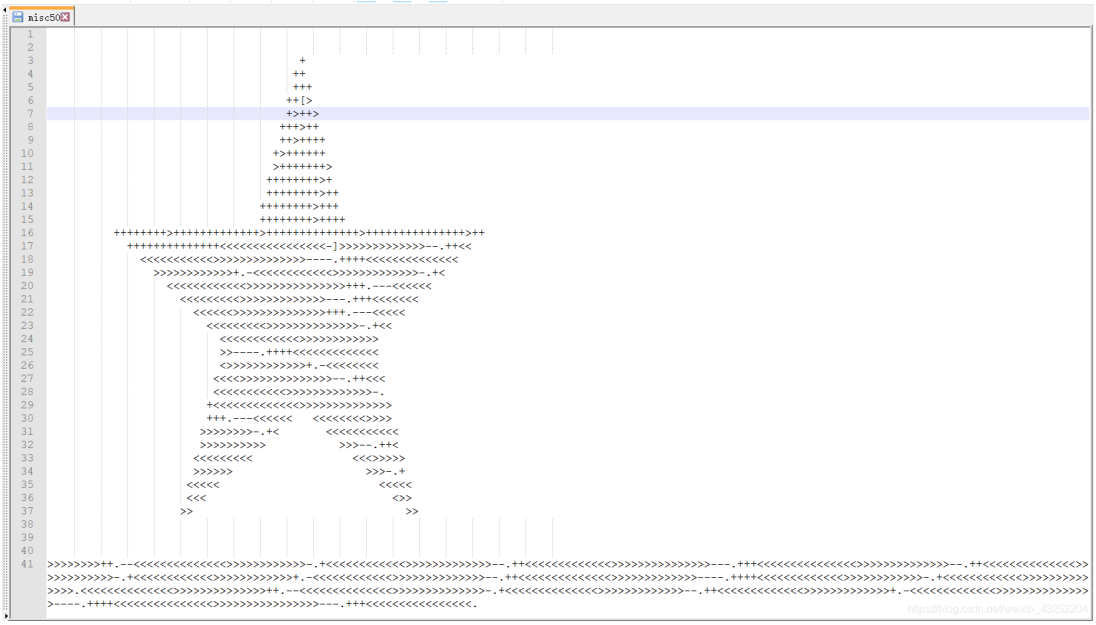
flag:

> steganol

> 有点坑，看到题目的提示还以为是图片隐写术的，用stegsolve工具跑了什么都没有，查看源码提示：Look what the hex-edit revealed:（看看十六进制编辑揭示了什么）考虑到进制转换方面的，也失败了，最后看了wp，发现flag直接是steganol，而且flag的格式也和之前的不一样

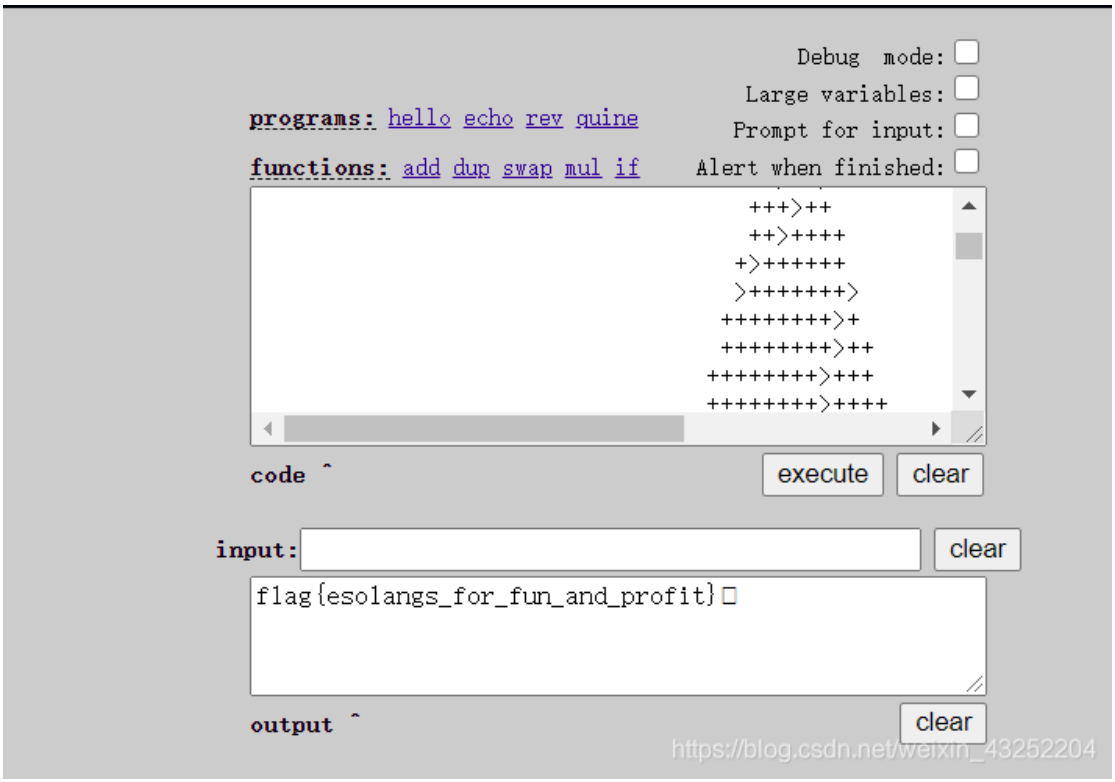
can_has_stdio?

下载附件，解压之后用文本打开文件



wp得知是brainfuck语言，使用在线语言转换得到flag

http://esoteric.sange.fi/brainfuck/impl/interp/i.html



flag:

> flag{esolangs_for_fun_and_profit}

János-the-Ripper

使用winhex打开，发现明显是一个压缩文件，修改后缀名，加压需要密码，使用ziperello工具破解

Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	ANSI ASCII
00000000	50	4B	03	04	14	00	03	00	08	00	0E	A2	77	44	44	D4	PK ¢wDDÔ
00000010	88	77	27	00	00	00	19	00	00	00	08	00	00	00	66	6C	ˆw' fl
00000020	61	67	2E	74	78	74	00	10	01	4B	93	FF	03	EE	9C	FA	ag.txt K"ÿ îœú
00000030	D3	12	83	A1	57	88	57	8C	BF	41	AA	41	87	16	F6	85	Ó ƒ¡WˆWŒ¿AªA‡ ö…
00000040	FE	40	02	DA	73	CA	1F	AC	16	97	89	44	3A	50	4B	01	þ@ ÚsÊ ¬ —‰D:PK
00000050	02	14	00	14	00	03	00	08	00	0E	A2	77	44	44	D4	88	¢wDDÔˆ
00000060	77	27	00	00	00	19	00	00	00	08	00	00	00	00	00	00	w'
00000070	00	01	00	20	00	00	00	00	00	00	00	66	6C	61	67	2E	flag.
00000080	74	78	74	50	4B	05	06	00	00	00	00	01	00	01	00	36	txtPK 6
00000090	00	00	00	4D	00	00	00	00	00								M

> **ziperello下载：**
>
> 链接：https://pan.baidu.com/s/1KW6UlUglDqKG-e5zu162Xg
>
> 提取码：oqyy

flag:

> flag{ev3n::y0u::bru7us?!}

Test-flag-please-ignore

常规操作之后，得到misc10文件，打开发现字符串

> 666c61677b68656c6c6f5f776f726c647d

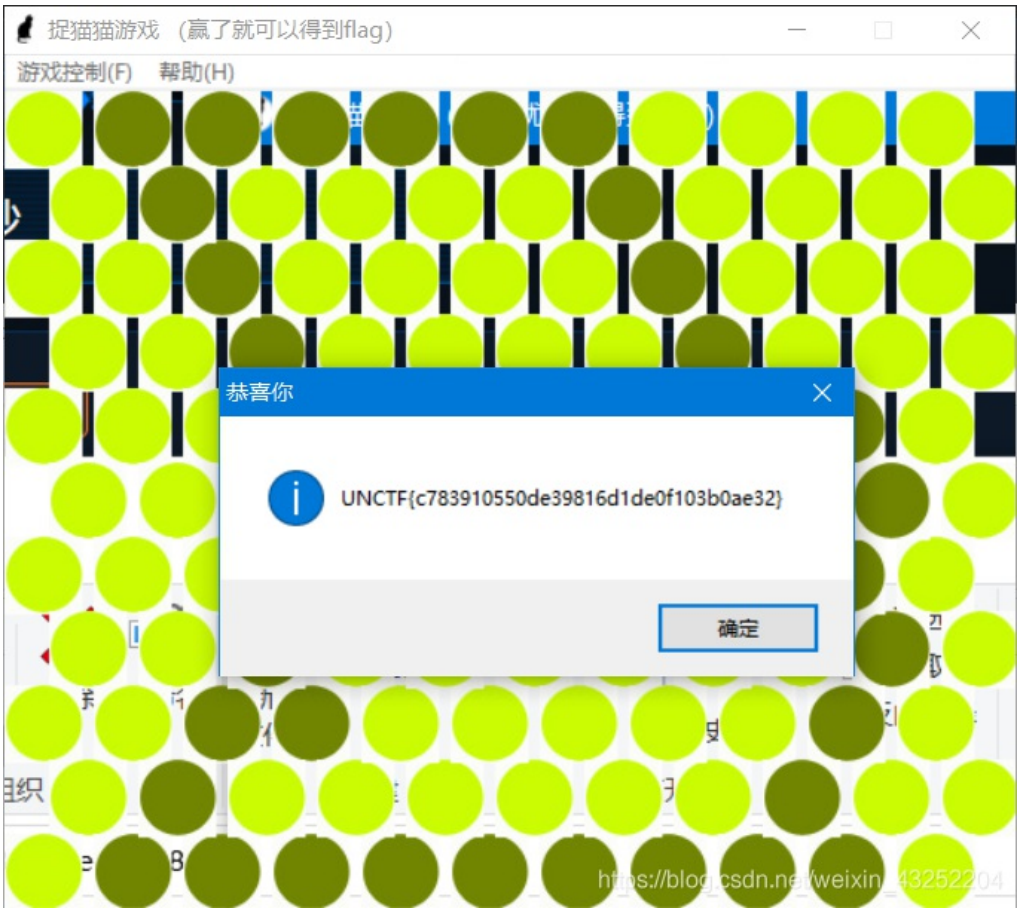
看到字符串没有f以后的，应该就是16进制了，使用base16解码得到flag

flag:

> flag{hello_world}

快乐游戏题

常规操作之后是一个，通关游戏即可

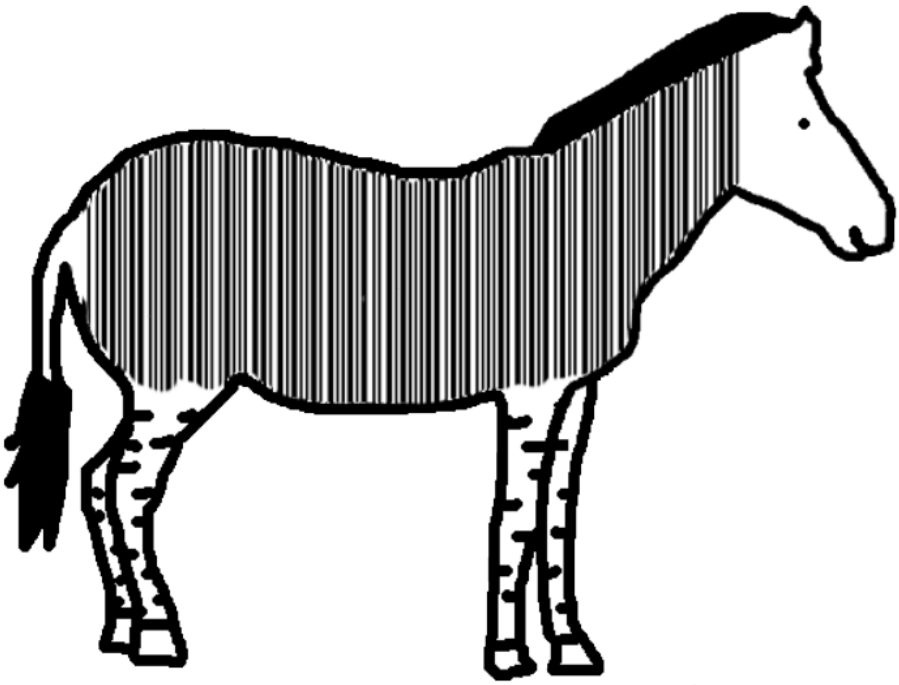


flag:

> UNCTF{c783910550de39816d1de0f103b0ae32}

Banmabanma

常规操作之后，得到一张斑马的图片



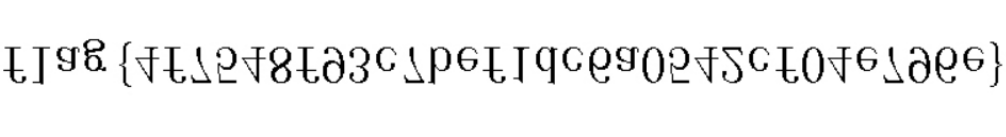
查看wp说斑马的纹路是条形码，直接手机扫码得到flag

flag:

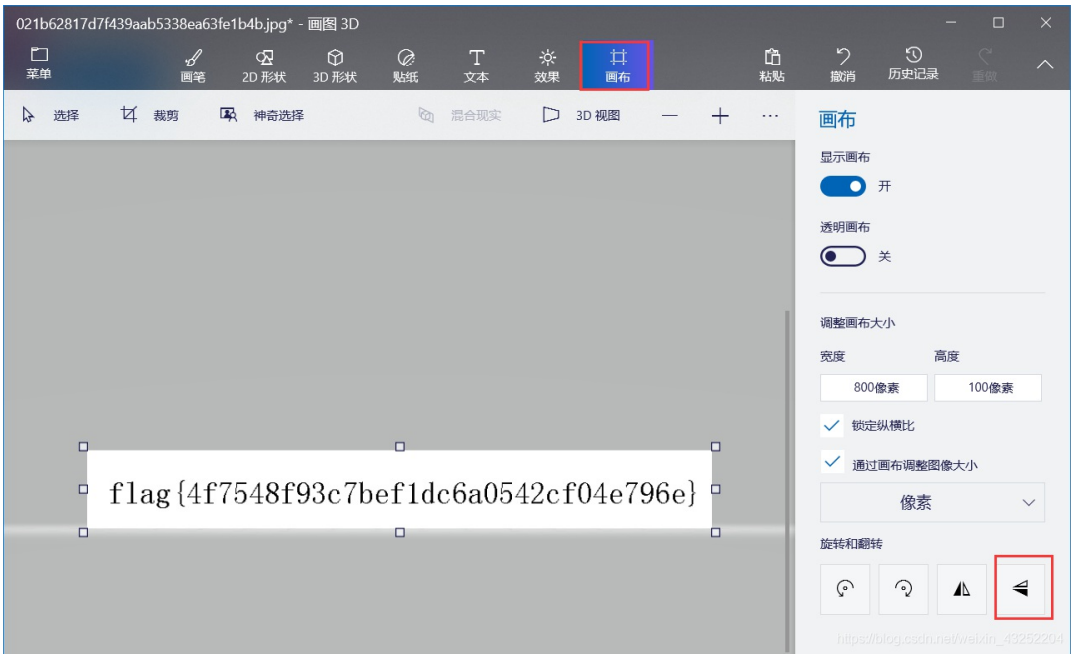
> flag{TENSHINE}

reverseMe

常规操作之后，得到一张图片



使用3D编辑，对图片进行垂直翻转，得到flag(或者拿一面镜子，更省事)



flag:

> flag{4f7548f93c7bef1dc6a0542cf04e796e}

stage1

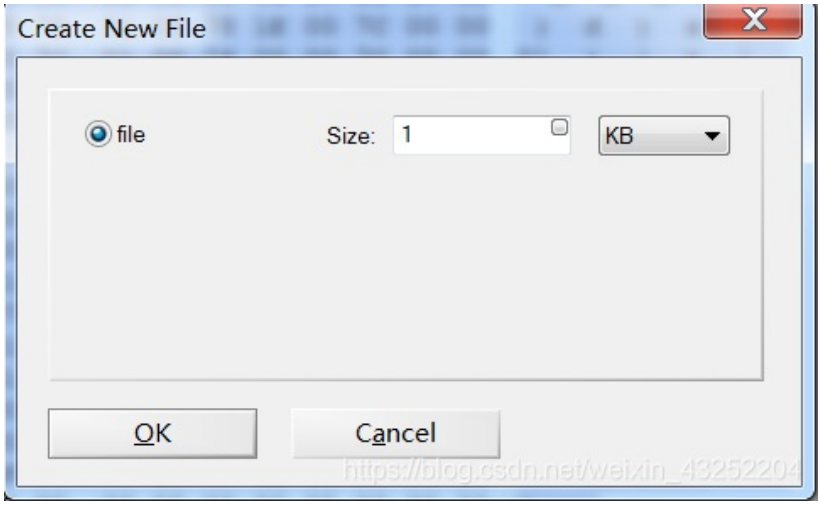
常规操作之后

手机扫描二维码得到

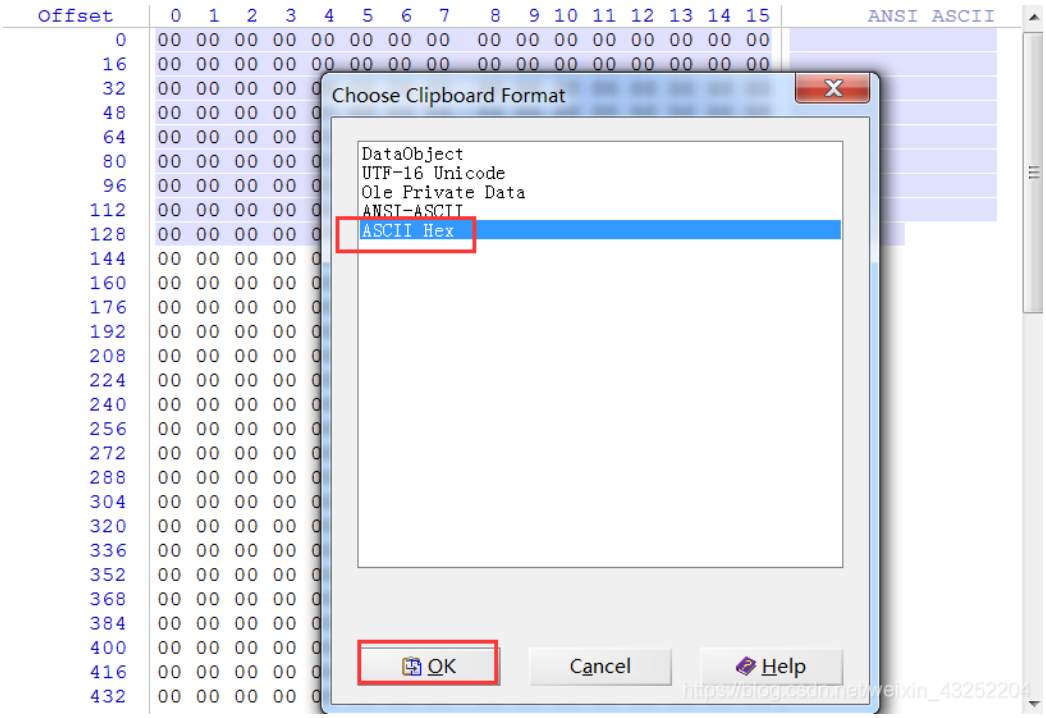
03F30D0AB6266A576300000000000000000010000004000000730D0000006400008400005A0000640100532

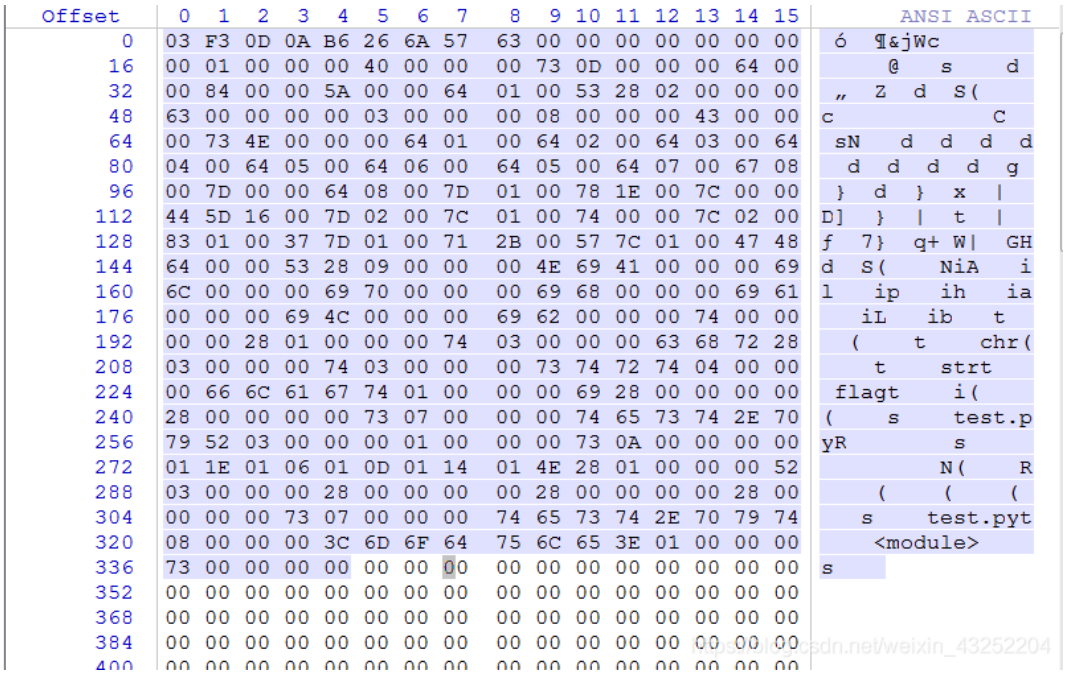
将其在winhex中打开

新建文件



将前面得到的字符串复制，ctrl+v 弹框选择





Offset	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	ANSI ASCII
0	03	F3	0D	0A	B6	26	6A	57	63	00	00	00	00	00	00	00	ó ¶&jWc
16	00	01	00	00	00	40	00	00	00	73	0D	00	00	00	64	00	@ s d
32	00	84	00	00	5A	00	00	64	01	00	53	28	02	00	00	00	„ Z d S(
48	63	00	00	00	00	03	00	00	00	08	00	00	00	43	00	00	c C
64	00	73	4E	00	00	00	64	01	00	64	02	00	64	03	00	64	sN d d d d
80	04	00	64	05	00	64	06	00	64	05	00	64	07	00	67	08	d d d d g
96	00	7D	00	00	64	08	00	7D	01	00	78	1E	00	7C	00	00	} d } x |
112	44	5D	16	00	7D	02	00	7C	01	00	74	00	00	7C	02	00	D] } | t |
128	83	01	00	37	7D	01	00	71	2B	00	57	7C	01	00	47	48	ƒ 7} q+ W| GH
144	64	00	00	53	28	09	00	00	00	4E	69	41	00	00	00	69	d S(NiA i
160	6C	00	00	00	69	70	00	00	00	69	68	00	00	00	69	61	l ip ih ia
176	00	00	00	69	4C	00	00	00	69	62	00	00	00	74	00	00	iL ib t
192	00	00	28	01	00	00	00	74	03	00	00	00	63	68	72	28	(t chr(
208	03	00	00	00	74	03	00	00	00	73	74	72	74	04	00	00	t strt
224	00	66	6C	61	67	74	01	00	00	00	69	28	00	00	00	00	flagt i(
240	28	00	00	00	00	73	07	00	00	00	74	65	73	74	2E	70	(s test.p
256	79	52	03	00	00	00	01	00	00	00	73	0A	00	00	00	00	yR s
272	01	1E	01	06	01	0D	01	14	01	4E	28	01	00	00	00	52	N(R
288	03	00	00	00	28	00	00	00	00	28	00	00	00	00	28	00	(((
304	00	00	00	73	07	00	00	00	74	65	73	74	2E	70	79	74	s test.pyt
320	08	00	00	00	3C	6D	6F	64	75	6C	65	3E	01	00	00	00	<module>
336	73	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	s
352	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
368	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
384	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
400	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	

观察发现为python编译之后的代码，保存为.pyc文件，反编译

python反编译在线工具：https://tool.lu/pyc/

反编译之后得到python源码，修饰之后运行

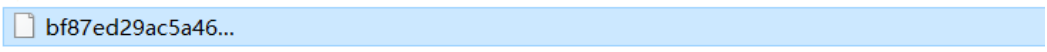
```
def flag():

    str = [65,108,112,104,97,76,97,98]

    flag = ''

    for i in str:

        flag += chr(i)

    print(flag)

flag()
```

flag:

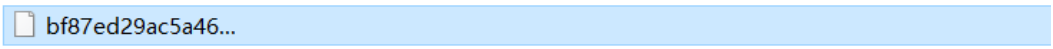
> AlphaLab

Hear-with-your-Eyes

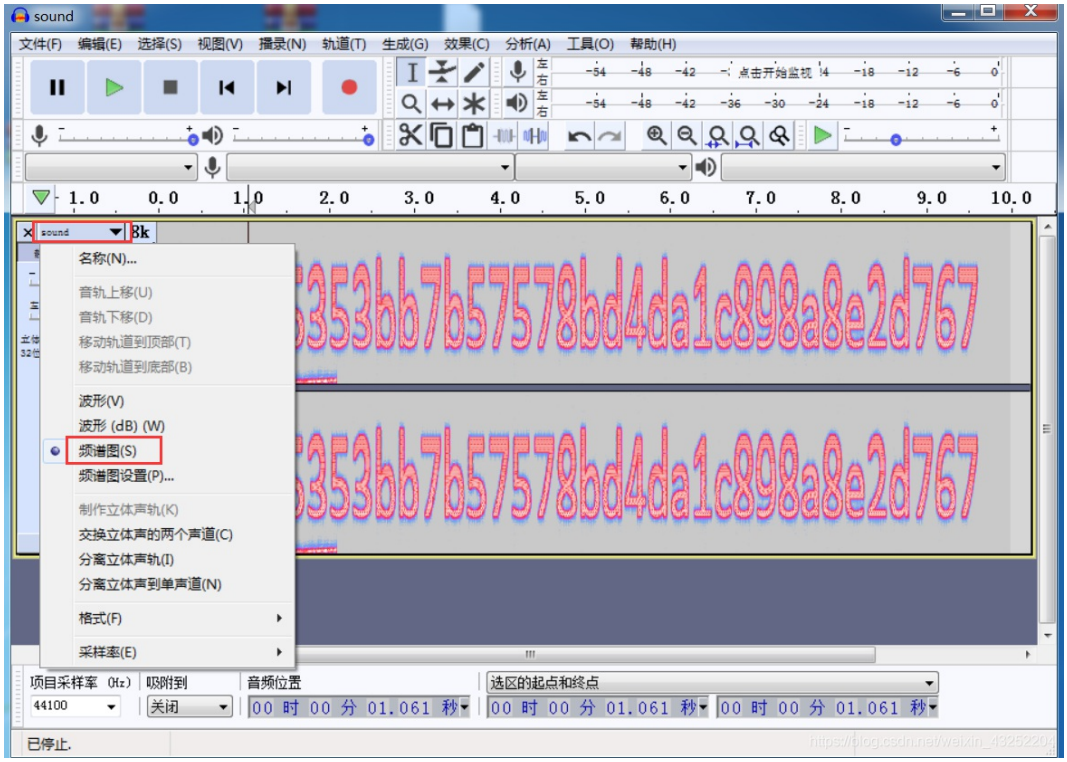
常规操作之后得到的文件



修改后缀名为.zip继续解压，得到文件



使用Audacity(https://www.onlinedown.net/soft/46359.htm)软件打开该sound.wav文件，查看频谱图，即可得到flag



flag:

> e5353bb7b57578bd4da1c898a8e2d767

What-is-this

常规操作之后，得到文件

修改后缀名.zip，继续解压

Offset	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	ANSI ASCII
0	03	F3	0D	0A	B6	26	6A	57	63	00	00	00	00	00	00	00	ó ¶&jWc
16	00	01	00	00	00	40	00	00	00	73	0D	00	00	00	64	00	@ s d
32	00	84	00	00	5A	00	00	64	01	00	53	28	02	00	00	00	„ Z d S(
48	63	00	00	00	00	03	00	00	00	08	00	00	00	43	00	00	c C
64	00	73	4E	00	00	00	64	01	00	64	02	00	64	03	00	64	sN d d d d
80	04	00	64	05	00	64	06	00	64	05	00	64	07	00	67	08	d d d d g
96	00	7D	00	00	64	08	00	7D	01	00	78	1E	00	7C	00	00	} d } x |
112	44	5D	16	00	7D	02	00	7C	01	00	74	00	00	7C	02	00	D] } | t |
128	83	01	00	37	7D	01	00	71	2B	00	57	7C	01	00	47	48	f 7} q+ W| GH
144	64	00	00	53	28	09	00	00	00	4E	69	41	00	00	00	69	d S(NiA i
160	6C	00	00	00	69	70	00	00	00	69	68	00	00	00	69	61	l ip ih ia
176	00	00	00	69	4C	00	00	00	69	62	00	00	00	74	00	00	iL ib t
192	00	00	28	01	00	00	00	74	03	00	00	00	63	68	72	28	(t chr(
208	03	00	00	00	74	03	00	00	00	73	74	72	74	04	00	00	t strt
224	00	66	6C	61	67	74	01	00	00	00	69	28	00	00	00	00	flagt i(
240	28	00	00	00	00	73	07	00	00	00	74	65	73	74	2E	70	(s test.p
256	79	52	03	00	00	00	01	00	00	00	73	0A	00	00	00	00	yR s
272	01	1E	01	06	01	0D	01	14	01	4E	28	01	00	00	00	52	N(R
288	03	00	00	00	28	00	00	00	00	28	00	00	00	00	28	00	(((
304	00	00	00	73	07	00	00	00	74	65	73	74	2E	70	79	74	s test.pyt
320	08	00	00	00	3C	6D	6F	64	75	6C	65	3E	01	00	00	00	<module>
336	73	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	s
352	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
368	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
384	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
400	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	

使用stegsolve工具的combiner功能将两张图片组合，即可得到flag

flag:

> AZADI TOWER

MISCall

常规操作得到文件，对其加.zip后缀连续两次进行解压，得到文件ctf

flag.txt文件中提示我们：Nothing to see here, moving along…（继续向前走）

发现有.git文件，接下里就要用到git的相关知识了

在linux下打开该ctf文件夹

```
~/Desktop/ctf$ ll
total 16
drwxrwxrwx 3 gcd gcd 4096 7月  25  2014 ./
drwxr-xr-x 5 gcd gcd 4096 8月  22 19:18 ../
-rwxrw-rw- 1 gcd gcd   37 7月  25  2014 flag.txt*
drwxrwxrwx 8 gcd gcd 4096 7月  25  2014 .git/
```

git log 查看git记录

```
~/Desktop/ctf$ git log
commit bea99b953bef6cc2f98ab59b10822bc42afe5abc
Author: Linus Torvalds <torvalds@klaava.Helsinki.Fi>
Date:   Thu Jul 24 21:16:59 2014 +0200

    Initial commit
```

git stash show 校验列表中存储的文件

```
~/Desktop/ctf$ git stash show
 flag.txt | 25 +++++++++++++++++++++++-
 s.py     |  4 ++++
 2 files changed, 28 insertions(+), 1 deletion(-)
```

rm flag.txt 为避免冲突，删除flag.txt文件

git stash apply 重新进行存储，复原文件

```
~/Desktop/ctf$ git stash apply
On branch master
Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

        new file:   s.py

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

        modified:   flag.txt
```


查看新的flag.txt文件，没有发现flag

查看s.py文件，发现是对 flag.txt 文件内容的读取并进行sha1 加密，同时在开头加上几个字母，怀疑与 flag 有关

```
~/Desktop/ctf$ cat s.py
#!/usr/bin/env python
from hashlib import sha1
with open("flag.txt", "rb") as fd:
    print "NCN" + sha1(fd.read()).hexdigest()
```

运行s.py，得到flag

```
~/Desktop/ctf$ python s.py
NCN4dd992213ae6b76f27d7340f0dde1222888df4d3
```

flag:

NCN4dd992213ae6b76f27d7340f0dde1222888df4d3