Jarvis OJ BASIC部分题目writeup

原创

dr0s3 于 2019-01-20 22:47:06 发布 1007 收藏 1

分类专栏： 其他 文章标签： Jarvis BASIC writeup

本文链接：https://blog.csdn.net/qq1045553189/article/details/86516990

base64?

```
GUYDIMZVGQ2DMN3CGRQTONJXGM3TINLGG42DGMZXGM3TINLGGY4DGNBXGYZTGNLGGY3DGNBWMU3WI===
```

base64后面最多可能出现两个=号，只有base32可能出现3个等号

使用base32解码，得到 `504354467b4a7573745f743373745f683476335f66346e7d`

猜测是hex编码的字符串，使用hex解码得到FLAG

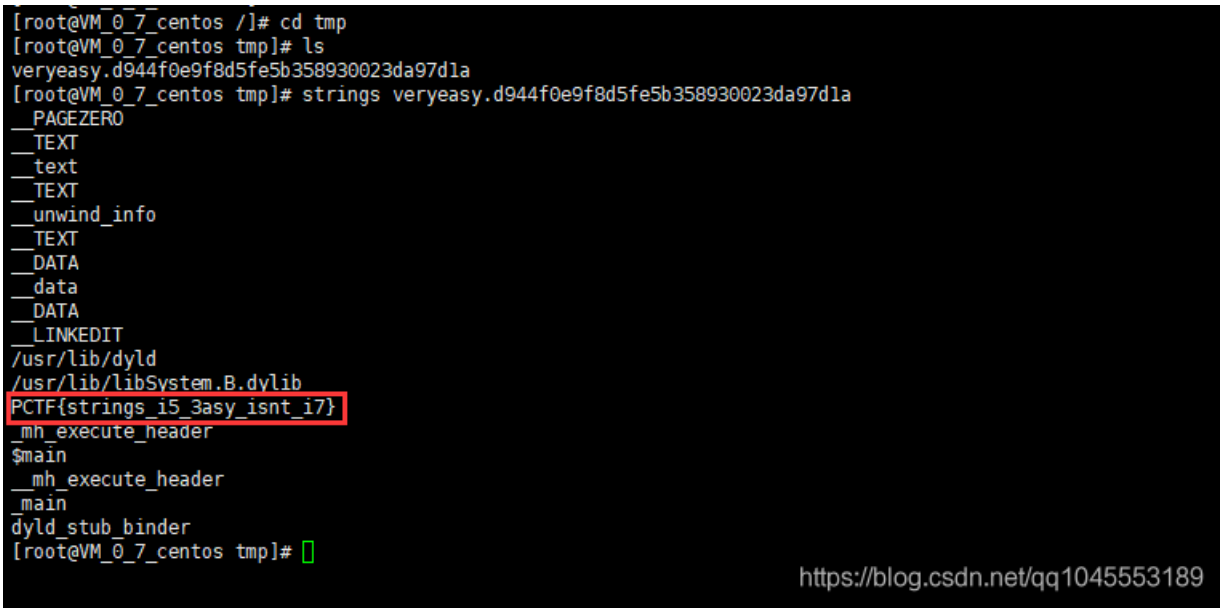
PCTF{Just_t3st_h4v3_f4n}

---

veryeasy

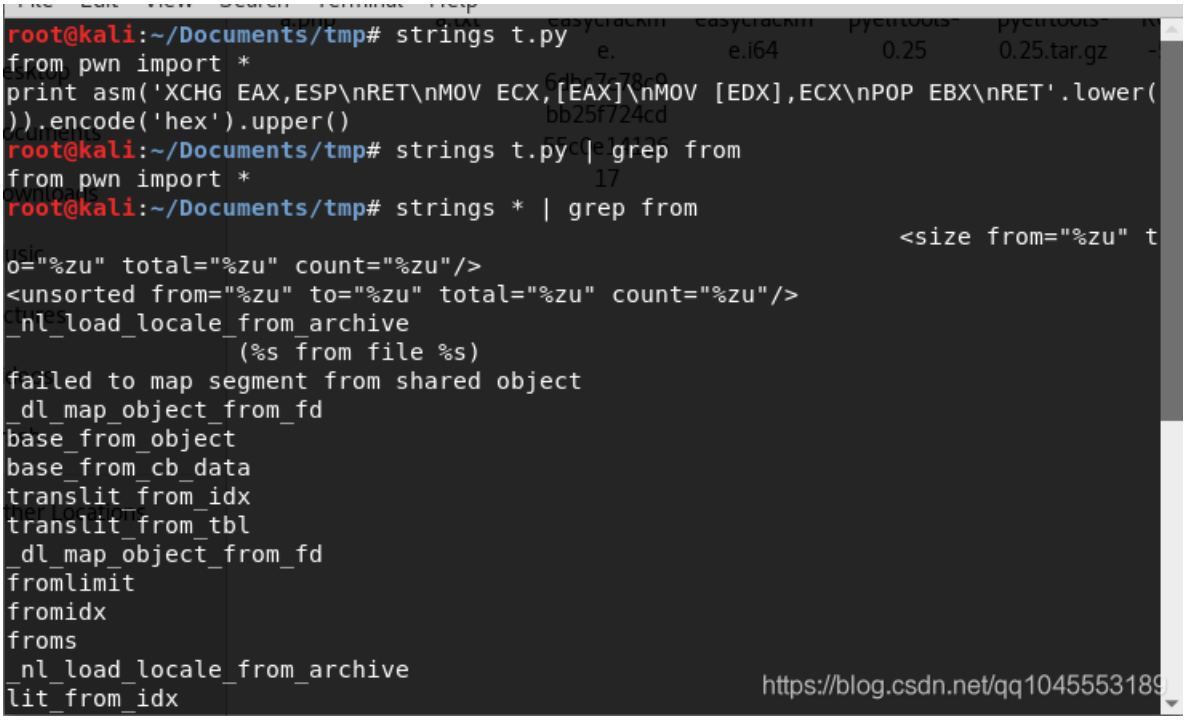
使用基本命令获取flag

使用strings命令，输出所有可以显示的字符



strings命令

strings - print the strings of printable characters in files.
打印文件中可打印的字符。可以配合管道使用。
例如：



段子

请提交其中"锟斤拷"的十六进制编码。(大写)
查看"锟斤拷"的GBK编码

手贱

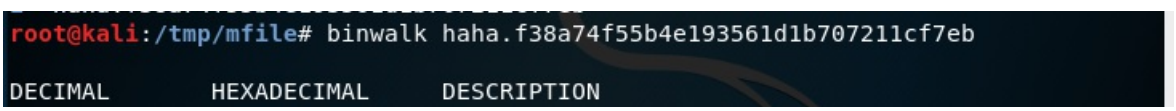
某天A君的网站被日，管理员密码被改，死活登不上，去数据库一看，啥，这密码md5不是和原来一样吗？为啥登不上咧？
d78b6f302l25cdc811adfe8d4e7c9fd34
请提交PCTF{原来的管理员密码}
md5128位，写成16进制是32个数，而不是33个数
将原来的md5减去一位（尝试每一位），使用md5解码，得到FLAG

美丽实验室logo

放到stegsolve里面浏览frame。拿到flag。

神秘的文件

使用wget命令下载
放到binwalk发现是可以挂载的文件



```
0             0x0             Linux EXT filesystem, rev 1.0, ext2 filesystem dat
a (mounted or unclean), UUID=8eecd08f-bae8-41ff-8497-8338f58af58a
35751         0x8BA7          mcrypt 2.2 encrypted data, algorithm: blowfish-448
, mode: CBC, keymode: 8bit
1058924       0x10286C        Unix path: /tmp/mfile/1/t.py
1573996       0x18046C        Unix path: /tmp/mfile/1/t.py
3146860       0x30046C        Unix path: /tmp/mfile/1/t.py
```


使用mount命令挂载

```
root@kali:/tmp/mfile# mkdir 1
root@kali:/tmp/mfile# cd 1
root@kali:/tmp/mfile/1# cd ..
root@kali:/tmp/mfile# mount haha.f38a74f55b4e193561d1b707211cf7eb ./1
root@kali:/tmp/mfile# cd 1
root@kali:/tmp/mfile/1# ls
0    113  129  144  16   175  190  205  220  236  251  39  54  7   85
1    114  13   145  160  176  191  206  221  237  252  4   55  70  86
10   115  130  146  161  177  192  207  222  238  253  40  56  71  87
100  116  131  147  162  178  193  208  223  239  26   41  57  72  88
101  117  132  148  163  179  194  209  224  24   27   42  58  73  89
102  118  133  149  164  18   195  21   225  240  28   43  59  74  9
103  119  134  15   165  180  196  210  226  241  29   44  6   75  90
104  12   135  150  166  181  197  211  227  242  3    45  60  76  91
105  120  136  151  167  182  198  212  228  243  30   46  61  77  92
106  121  137  152  168  183  199  213  229  244  31   47  62  78  93
107  122  138  153  169  184  2    214  23   245  32   48  63  79  94
108  123  139  154  17   185  20   215  230  246  33   49  64  8   95
109  124  14   155  170  186  200  216  231  247  34   5   65  80  96
11   125  140  156  171  187  201  217  232  248  35   50  66  81  97
110  126  141  157  172  188  202  218  233  249  36   51  67  82  98
111  127  142  158  173  189  203  219  234  25   37   52  68  83  99
112  128  143  159  174  19   204  22   235  250  38   53  69  84  lost+found
```


发现有很多文件，且都只有一个字母，写一个脚本把这些文件拼接起来即可

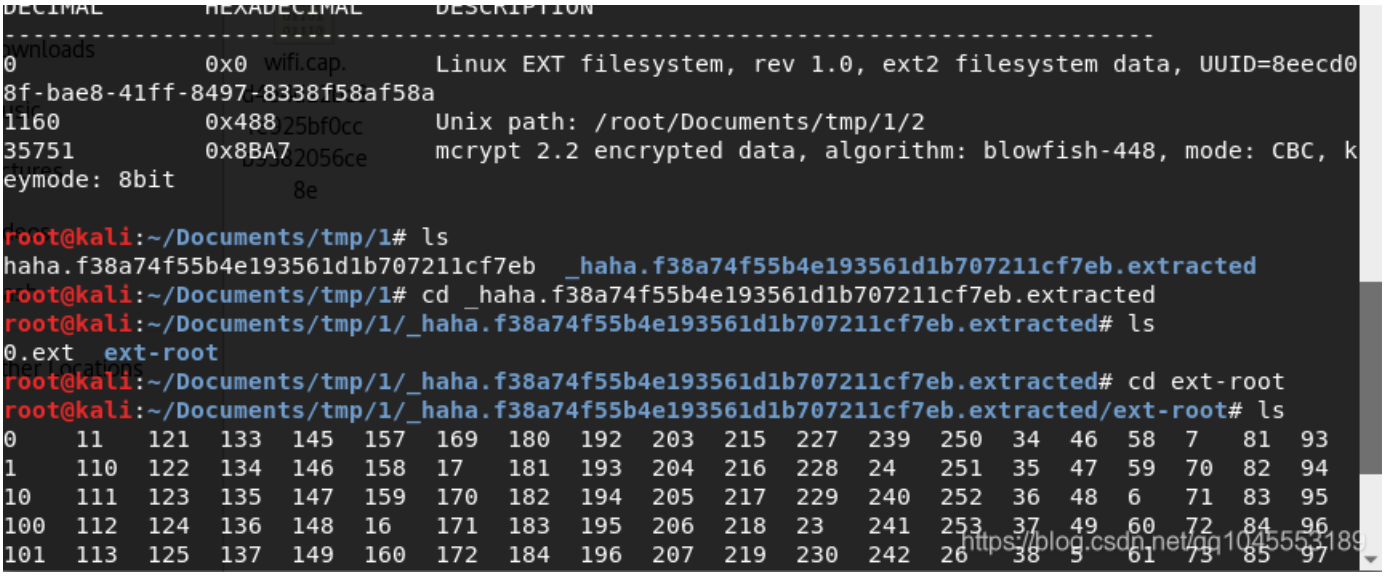
```
File  Edit  View  Search  Terminal  Help
out =''
for i in range(254):
    out += open(str(i),'r').read()
print(out)
```


得到flag

```
root@kali:/tmp/mfile/1# python t.py
Haha ext2 file system is easy, and I know you can easily decompress of it and fi
nd the content in it.But the content is spilted in pieces can you make the piece
s together. Now this is the flag PCTF{P13c3_7oghter_i7}. The rest is up to you.
Cheer up, boy.
```

这里也可以不使用mount挂载，而使用binwalk -e命令分离其中的文件

```
File  Edit  View  Search  Terminal  Help
root@kali:~/Documents/tmp/1# ls
haha.f38a74f55b4e193561d1b707211cf7eb
root@kali:~/Documents/tmp/1# binwalk -e haha.f38a74f55b4e193561d1b707211cf7eb
```

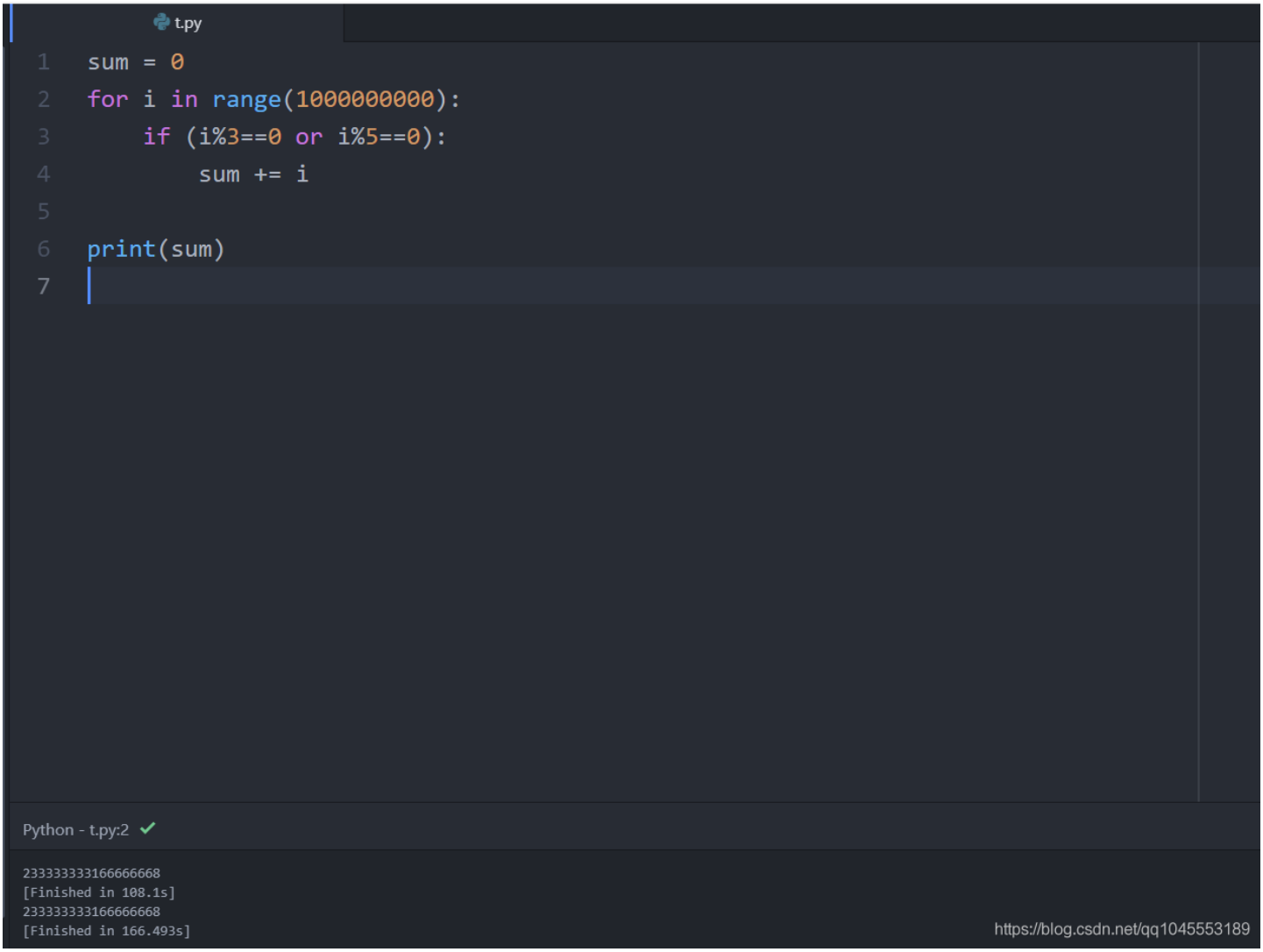


binwalk

做misc题的时候，一般都需要从某个网址下载一个文件，然后开始分析这个文件，最终从文件中得到flag。而查看下载的文件中是否包含其他文件的时候一般都需要binwalk这个神器。

binwalk参考

公倍数



veryeasyRSA

```
import libnum

p = 348758394758943758923795872389234625477 7
q = 876786784356893476598347658437657838 9
e = 65537
phin = (p - 1) * (q - 1)
d = libnum.invmod(e, phin)
print(d)
# d=191785687961555604236759757741428291538278837090277177233630776062607174343 69
```

kali安装gmpy2教程

Easy RSA

```python
import libnum

c = 0xdc2eeeb2782c
n = 322831561921859
e = 23
q = 13574881
p = 23781539

d = libnum.invmod(e, (p - 1) * (q - 1))
m = pow(c, d, n)
print(libnum.n2s(m))
```

爱吃培根的出题人

密文：

```
bacoN is one of aMerICa'S sWEethEartS. it's A dARlinG, SuCCulEnt fOoD tHAt PaIRs FlawLE
```

去掉无关符号：

```
bacoNisoneofaMerICaSsWEethEartSitsAdARlinGSuCCulEntfOoDtHAtPaIRsFlawLE
```

大小写分别代表b和a：

```
aaaabaaaaaaaabaabbababbaaabaaabaaababbaaabbabbaabaaababababbababbabaaabb
```

按照第二种方式解密得：baconisnotfood

取证

百度“内存取证”，即volatility

熟悉的声音

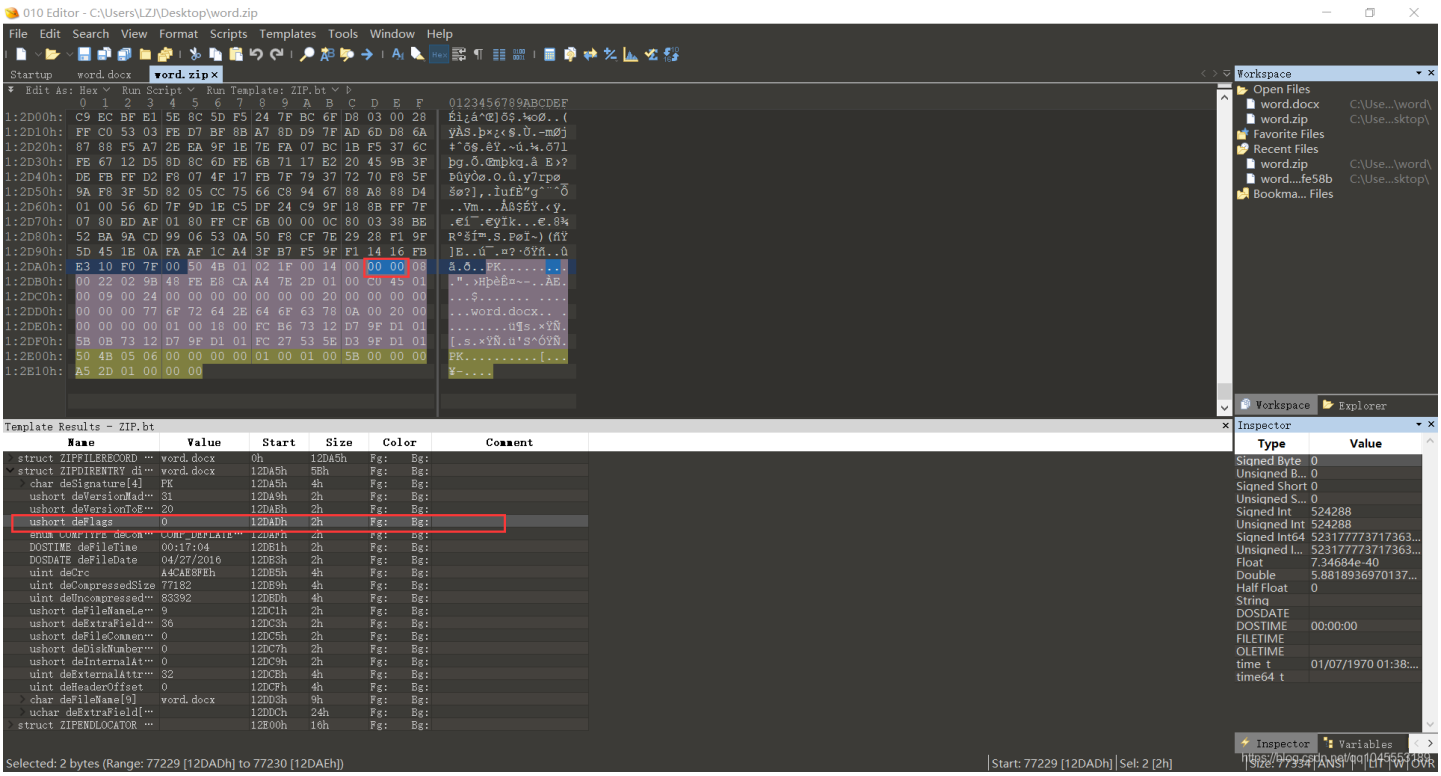
只有X、Y空格三种字符，猜测是摩斯密码

```
XYYY YXXX XYXX XXY XYY X XYY YX YYXX
.--- -... .-.. ..- .-- . .-- -. --..
```

解密得`JBLUWEWNZ`
交上去不对，凯撒一下，得到flag：`PHRACKCTF`

Help!!

使用010Editor打开压缩包word.zip，其中deFlags字段标记是否加密。当该字段为偶数时含义是未加密，当该字段是奇数时，含义是已加密。打开word.zip后发现该字段是09，将其修改为00，即可正常解压。



解压后打开word.docx，并没有看到flag

但是发现word文件首为PK开头，那说明是就一个压缩文件包了，尝试将该word文档的后缀改为.zip，然后用解压缩软件居然果然能够正常解压缩。

在 word\media 目录下发现flag

参考：

Help!!

ZIP伪加密

ZIP文件头协议

Word文件格式分析

ROPGadget

```
from pwn import *
print asm('XCHG EAX,ESP\nRET\nMOV ECX,[EAX]\nMOV [EDX],ECX\nPOP EBX\nRET'.lower()).encode('hex').upper()
```

Kali安装pwntools

---

a piece of cake

quipquip传送门

解得：

```
the word ro?ot can refer to ?oth physical ro?ots and virtual software agents, ?ut the latter are usually
referred to as ?ots. there is no consensus on which machines ?ualify as ro?ots ?ut there is general
agreement among e?perts, and the pu?lic, that ro?ots tend to do some or all of the following: accept
electronic programming, process data or physical perceptions electronically, operate autonomously to
some degree, move around, operate physical parts of itself or physical processes, sense and
manipulate their environment, and e?hi?it intelligent ?ehavior - especially ?ehavior which mimics
humans or other animals. flag is su?stitutepassisveryeasyyougotit. closely related to the concept of a
ro?ot is the field of synthetic ?iology, which studies entities whose nature is more compara?le to ?
eings than to machines.
```

容易猜到?是b

Shellcode

文件中都是可见字符串，可能是经过处理得shellcode。

```
int main(void)
{
    int x;
    char a[] = "PYIIIIIIIIIIIIIIII7QZjAXP0A0AkAAQ2AB2BB0BBABXP8ABuJIYIhkmKzyCDq4l4FQyBlrRWEahI1tLKT16Pnk1ftLnkPv
wlnkW6fhNkan5pNkgF6XPOR8T5HsCivaN19okQSPlKRLvD6DNk3uelNkpTthRXuQ9znk2jEHLK1Ja0FaXkhcTtBink4tlKUQhnvQYotqo0ylnLMT
O0SDEWZahOtMwqhG8kXteksLwTdh1e8aLKsja4uQ8kavLKdLrklK0ZeL7qjKLKUTLKuQM8k9bdvDeL1qiSnR5XVIXTOyjENikrphNnrnVnhlBrzH
ooKOYoyok93u7tOKCNyHzBBSnguLgTcbyxlNKOYoYoMYaUTHphRL2LupQQ0htsFRTn541x3E2Se5T26PyKK8QLTddJlIZFBvyoSeUTLIkrv0oKy8
ORpMmlk7Gl6DBrm8Soyoioyoaxr0qh0XwP1xu1Qw1upBbHrmrED3T34qiKOxQLTdEZOyZCaxQmRxgPUp0hpnPn4srRe8BDSo2PT7axqOCWROpoph
SYpnSo04u83K72Peu70hBpCsqDpF4qHIMXpLQ429k98aEaJr1BF3Ca3bIozp01IPf0Yof5GxAA";

    asm("jmp %0;"
        : "=a"(x)
        : "0"(a));
}
```

运行即可得到flag
参考：
171115 杂项-可见字符组成的Shellcode

Shellcode

Shellcode实际上就是汇编对应的机器码
但是由于机器码大部分都是不可见字符，所以无法直接显示出来

本题直接给了大段的乱码可见字符串，再结合机器码不可见，首先就想到了Base64编码–它的出现就是为了将不可见字符全部转为可见字符嘛

然而解b64发现不仅长度不符合，而且在最后添上等号以后解出来的值转汇编也并没有意义
Alpha2这个工程是专门将Shellcode编码成可见字符串的（甚至仅有字母和数字）

-.-字符串

摩斯解码即可得到flag
摩斯电码转换器

德军的密码

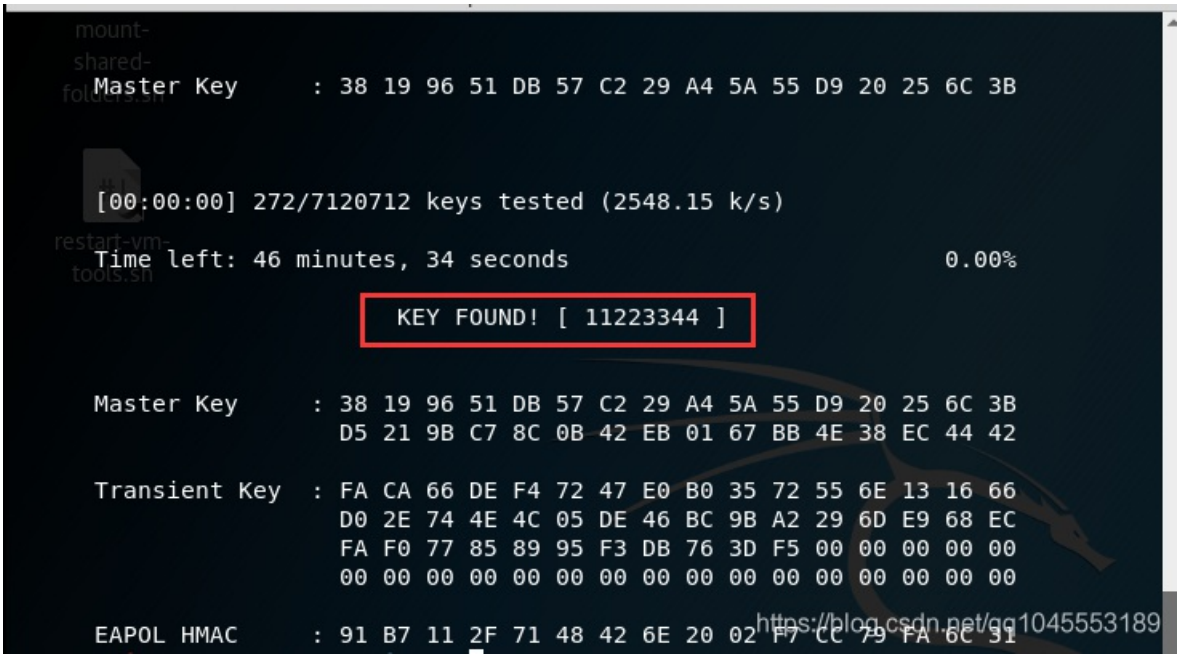
密钥也是12位大写字母
密文一共84位，7位对应一个明文字符
通过异或得到明文

```
a = '000000000000000000000000000000000000000000000000000000000000000010111000011000100000010100000001'
b = 'WELCOMETOCFF'
c = ''

for i in range(12):
    d=int(a[i*7:(i+1)*7],2)
    e=ord(b[i])
    c+=chr(d^e)

print(c)
```

握手包

使用Kali中的aircrack-ng爆破密码即可
输入命令：

```
aircrack-ng -w /usr/share/wordlists/rockyou.txt ~/Documents/tmp/wifi.cap.d4e4d22bc8fe925bf0ccb9382056ce8e
```



得到密码：`11223344`

Baby's Crack

解压后得到两个文件，一个加密程序和一个flag加密得到的密文。将加密程序丢到IDA中，可以看到关键的加密代码如下：

```
31  LODWORD(v6) = fopen(*(_QWORD *)&argc, argv, "rb+", v19[1]);
32  v16 = v6;
33  if ( v6 )
34  {
35    LODWORD(v7) = fopen(*(_QWORD *)&argc, argv, "wb+", "tmp");
36    v15 = v7;
37    while ( feof(*(_QWORD *)&argc, argv, v8, v16) == 0 )
38    {
39      v17 = fgetc(*(_QWORD *)&argc, argv, v9, v16);
40      if ( v17 != -1 && v17 )
41      {
42        if ( v17 > 47 && v17 <= 96 )
43        {
44          v17 += 53;
45        }
46        else if ( v17 <= 46 )
47        {
48          v17 += v17 % 11;
49        }
50        else
51        {
52          v17 -= v17 % 61;
53        }
54        fputc(*(_QWORD *)&argc, argv, v15, (unsigned int)v17);
55      }
56    }
57    fclose(*(_QWORD *)&argc, argv, v9, v15);
58    fclose(*(_QWORD *)&argc, argv, v10, v16);
59    sprintf(*(_QWORD *)&argc, argv, "del %s", &v14, v19[1]);
00000C2B 53
```


暴力破解得到flag：

```
f = open("flag.enc", "rb+")
cypher = f.read()
flag = ''
print(cypher)

for i in range(len(cypher)):
    for j in range(256):
        t = j
        if t > 47 and t <= 96 :
            t += 53
        elif t <= 46 :
            t += t % 11
        else :
            t -= t % 61
        if t == cypher[i] :
            flag += chr(j)

print(flag)
print(bytearray.fromhex(flag))
f.close()
```

运行结果：

```
b'jeihjiiklwjnk{ljj{kflghhj{ilk{k{kij{ihlgkfkhkwhhjgly'
504354467B596F755F6172335F476F6F645F437261636B33527D
bytearray(b'PCTF{You_ar3_Good_Crack3R}')
```