“东华杯”2021年大学生网络安全邀请赛 暨第七届上海市大学生网络安全大赛线上赛MISC-Writeup

原创

末 初 于 2021-11-01 20:49:06 发布 10078 收藏 13

分类专栏： CTF_MISC_Writeup 文章标签： 2021东华杯

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/mochu7777777/article/details/121073753
版权

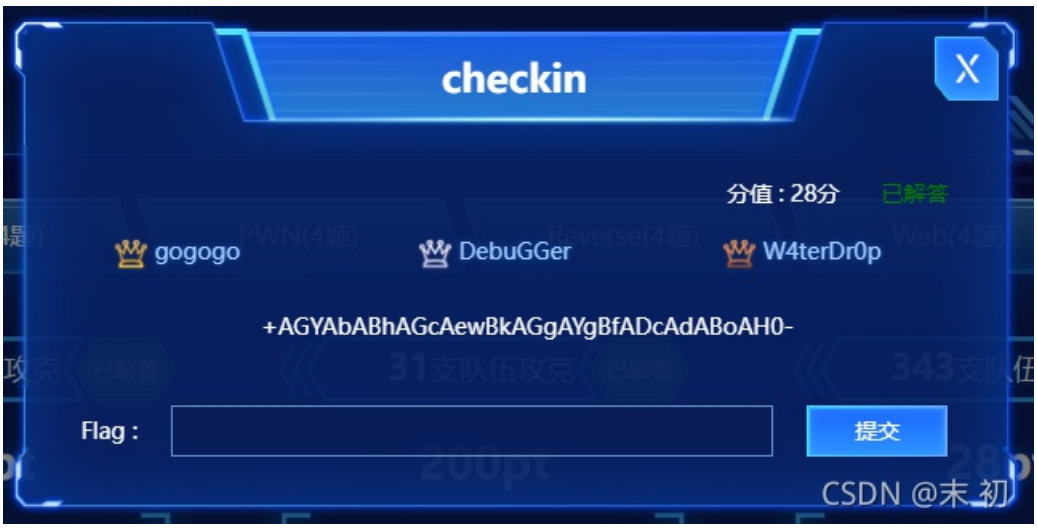
CTF_MISC_Writeup 专栏收录该内容
246 篇文章 46 订阅
订阅专栏

文章目录

checkin
project
JumpJumpTiger
where_can_find_code

```
题目附件请自取：
链接：https://pan.baidu.com/s/1T9nG-CDg_D8QYQZapuxucg
提取码：2wub
```

checkin



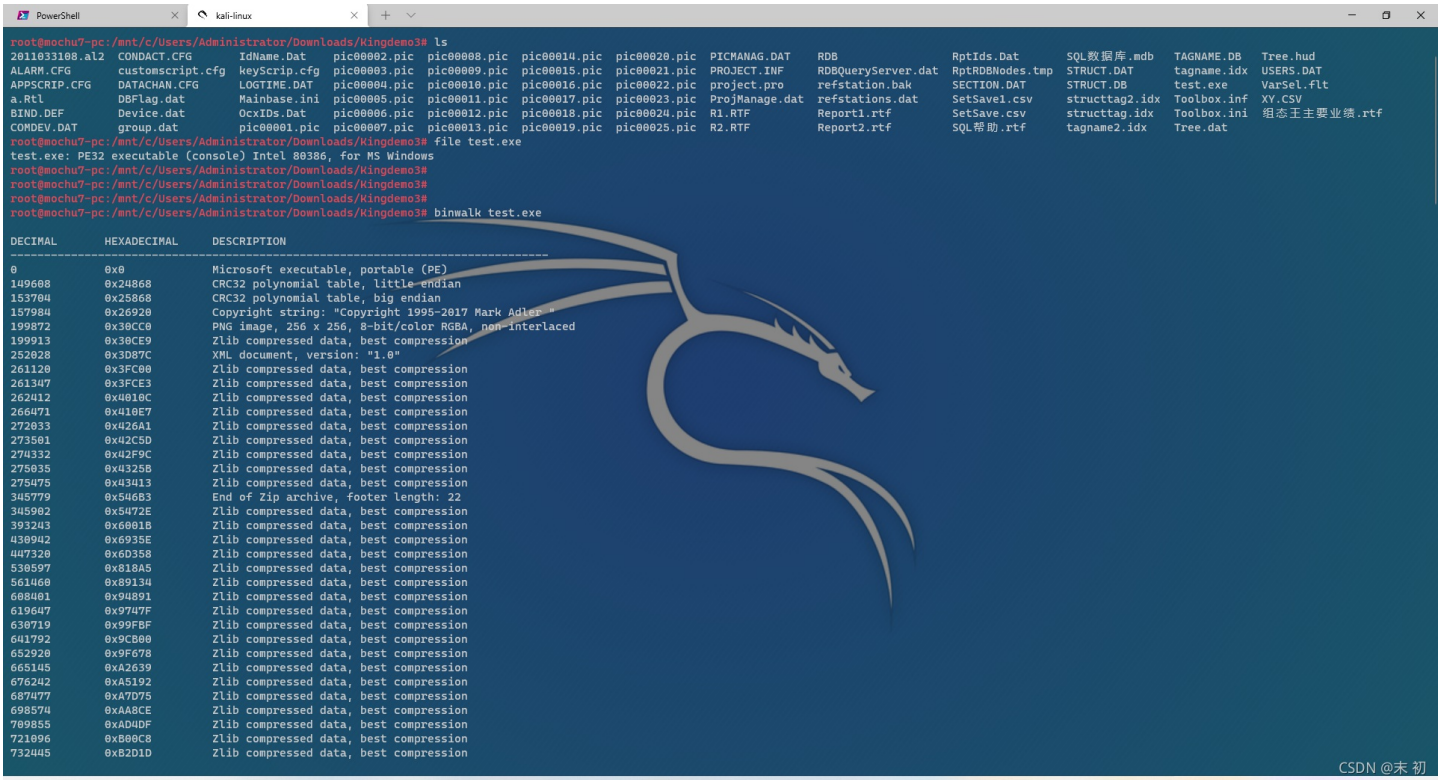

```
+AGYAbABhAGcAewBkAGgAYgBfADcAdABoAH0-
```

`UTF-7` 编码

- UTF-7在线解码站：http://toolswebtop.com/text/process/decode/utf-7

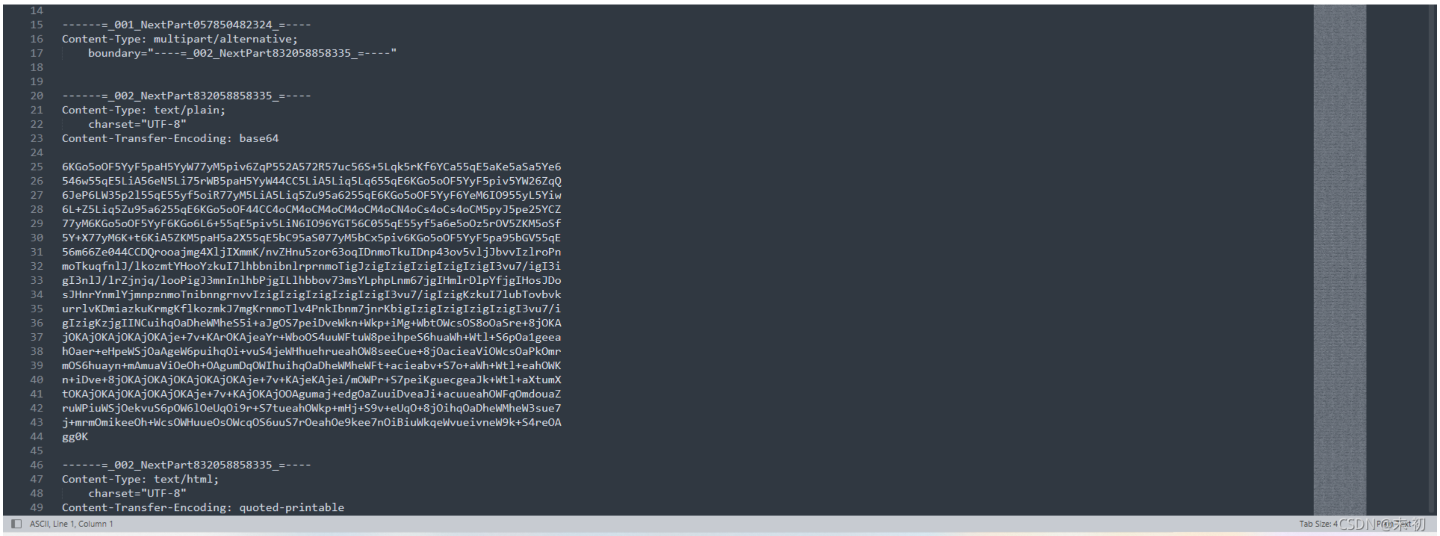
```
flag{dhb_7th}
```

project



`test.exe` 附加了很多别的数据，运行之后发现生成了一个 `problem_bak.zip`

解压 `problem_bak.zip` 得到一个邮件文件



```
------=_001_NextPart057850482324_=----
Content-Type: multipart/alternative;
    boundary="----=_002_NextPart832058858335_=----"


------=_002_NextPart832058858335_=----
Content-Type: text/plain;
    charset="UTF-8"
Content-Transfer-Encoding: base64

```

6KGo5oOF5YyF5paH5YyW77yM5piv6ZqP552A572R57uc56S+5Lqk5rKf6YCa55qE5aKe5aSa5Ye6
546w55qE5LiA56eN5Li75rWB5paH5YyW44CC5LiA5Liq5Lq655qE6KGo5oOF5YyF5piv5YW26ZqQ
6JeP6LW35p2l55qE55yf5oiR77yM5LiA5Liq5Zu95a6255qE6KGo5oOF5YyF6YeM6IO955yL5Yiw
6L+Z5Liq5Zu95a6255qE6KGo5oOF44CC4oCM4oCM4oCM4oCM4oCN4oCs4oCs4oCM5pyJ5pe25YCZ
77yM6KGo5oOF5YyF6KGo6L6+55qE5piv5LiN6IO96YGT56C055qE55yf5a6e5oOz5rOV5ZKM5oSf
5Y+X77yM6K+t6KiA5ZKM5paH5a2X55qE5bC95aS077yM5bCx5piv6KGo5oOF5YyF5pa95bGV55qE
56m66Ze044CCDQrooajmg4XljIXmmK/nvZHnu5zor63oqIDnmoTkuIDnp43ov5vljJbvvIzlroPn
moTkuqfnlJ/lkozmtYHooYzkuI7lhbbnibnlrprnmoTigJzigIzigIzigIzigIzigI3vu7/igI3i
gI3nlJ/lrZjnjq/looPigJ3mnInlhbPjgILlhbbov73msYLphpLnm67jgIHmlrDlpYfjgIHosJDo
sJHnrYnmlYjmnpznmoTnibnngrnvvIzigIzigIzigIzigIzigI3vu7/igIzigKzkuI7lubTovbvk
urrlvKDmiazkuKrmgKflkozmkJ7mgKrnmoTlv4PnkIbnm7jnrKbigIzigIzigIzigIzigI3vu7/i
gIzigKzjgIINCuihqOaDheWMheS5i+aJgOS7peiDveWkn+Wkp+iMg+WbtOWcsOS8oOaSre+8jOKA
jOKAjOKAjOKAjOKAje+7v+KArOKAjeaYr+WboOS4uuWFtuW8peihpeS6huaWh+Wtl+S6pOa1geea
hOaer+eHpeWSjOaAgeW6puihqOi+vuS4jeWHhuehrueahOW8seeCue+8jOacieaViOWcsOaPkOmr
mOS6huayn+mAmuaViOeOh+OAgumDqOWIhuihqOaDheWMheWFt+acieabv+S7o+aWh+Wtl+eahOWK
n+iDve+8jOKAjOKAjOKAjOKAjOKAje+7v+KAjeKAjei/mOWPr+S7peiKguecgeaJk+Wtl+aXtumX
tOKAjOKAjOKAjOKAjOKAje+7v+KAjOKAjOOAgumaj+edgOaZuuiDveaJi+acuueahOWFqOmdouaZ
ruWPiuWSjOekvuS6pOW6lOeUqOi9r+S7tueahOWkp+mHj+S9v+eUqO+8jOihqOaDheWMheW3sue7
j+mrmOmikeeOh+WcsOWHuueOsOWcqOS6uuS7rOeahOe9kee7nOiBiuWkqeWvueivneW9k+S4reOA
gg0K

```

------=_002_NextPart832058858335_=----
Content-Type: text/html;
    charset="UTF-8"
Content-Transfer-Encoding: quoted-printable
```

ASCII, Line 1, Column 1

复制出来Python简单处理或者直接使用在线站：https://the-x.cn/zh-cn/base64/

```python
from base64 import *

with open('jpg_base64.txt', 'r') as f:
 lines = f.readlines()
 base64_data = ''
 with open('flag.jpg', 'wb') as f1:
  for line in lines:
   base64_data += line.strip()
  f1.write(b64decode(base64_data))
```

CSDN @末 初

得到一张丑不拉几的图片，尝试了很多jpg隐写，最后发现是`OurSecret`隐写

但是没有密码，继续分析；在邮件里还有一些数据，一段Base64、一段Quoted-Printable编码

Decoded:

```
<style>body { line-height: 1.5; }body { font-size: 14px; font-family: "Microsoft YaHei UI"; color: rgb(0, 0, 0); line-height: 1.5; }body { font-size: 14px; font-family: "Microsoft YaHei UI"; color: rgb(0, 0, 0); line-height: 1.5; }</style></head>
<body>
<div><div>表情包文化，是随着网络社交沟通的增多出现的一种主流文化。一个人的表情包是其隐藏
```

Encoded: (permalink)

```
=E8=BD=AF=E4=BB=B6=E7=9A=84=E5=A4=A7=E9=87=8F=E4=BD=BF=E7=94=A8=EF=BC=8C=
```

```
=E8=A1=A8=E6=83=85=E5=8C=85=E5=B7=B2=E7=BB=8F=E9=AB=98=E9=A2=91=E7=8E=87=
=E5=9C=B0=E5=87=BA=E7=8E=B0=E5=9C=A8=E4=BA=BA=E4=BB=AC=E7=9A=84=E7=BD=91=
=E7=BB=9C=E8=81=8A=E5=A4=A9=E5=AF=B9=E8=AF=9D=E5=BD=93=E4=B8=AD=E3=80=82</=
div></div><img src=3D"cid:_Foxmail.1@e1a633df-2fe4-f85c-7270-ca78308ac1c5"=
 border=3D"0"></body></html>
```

CSDN @末 初

没有密码线索，都是文本，试一下零宽度字符隐写

- https://330k.github.io/misc_tools/unicode_steganography.html

Unicode Steganography with Zero-Width Characters

This is plain text steganography with zero-width characters of Unicode.
Zero-width characters is inserted within the words.

JavaScript library is below.
http://330k.github.io/misc_tools/unicode_steganography.js

Text in Text Steganography Sample

Original Text: Clear (length: 307)

表情包文化，是随着网络社交沟通的增多出现的一种主流文化。一个人的表情包是其隐蔽起来的真我，一个国家的表情包里能看到这个国家的表情。有时候，表情包表达的是不能道破的真实想法和感受，语言和文字的尽头，就是表情包施展的空间。
表情包是网络语言的一种进化，它的产生和流行与其特定的“生存环境”有关。其追求醒目、新奇、谐谑等效果的特点，与年轻人张扬个性和搞怪的心理相符。
表情包之所以能够大范围地传播，是因为其弥补了文字交流的枯燥和态度表达不准确的弱点，有效地提高了沟通效率。部分表情包具有替代文字的功能，还可以节省打字时间。随着智能手机的全面普及和社交应用软件的大量使用，表情包已经高频率地出现在人们的网络聊天对话当中。

Hidden Text: Clear (length: 7)

hurryup

Encode »

« Decode

Steganography Text: Clear (length: 363)

表情包文化，是随着网络社交沟通的增多出现的一种主流文化。一个人的表情包是其隐蔽起来的真我，一个国家的表情包里能看到这个国家的表情。有时候，表情包表达的是不能道破的真实想法和感受，语言和文字的尽头，就是表情包施展的空间。
表情包是网络语言的一种进化，它的产生和流行与其特定的“生存环境”有关。其追求醒目、新奇、谐谑等效果的特点，与年轻人张扬个性和搞怪的心理相符。
表情包之所以能够大范围地传播，是因为其弥补了文字交流的枯燥和态度表达不准确的弱点，有效地提高了沟通效率。部分表情包具有替代文字的功能，还可以节省打字时间。随着智能手机的全面普及和社交应用软件的大量使用，表情包已经高频率地出现在人们的网络聊天对话当中。

Download Stego Text as File

CSDN @末 初

Unicode Steganography with Zero-Width Characters

This is plain text steganography with zero-width characters of Unicode.
Zero-width characters is inserted within the words.

JavaScript library is below.
http://330k.github.io/misc_tools/unicode_steganography.js

Text in Text Steganography Sample

Original Text: Clear (length: 768)

<html><head><meta http-equiv="content-type" content="text/html; charset=UTF-8"><style>body { line-height: 1.5; }body { font-size: 14px; font-family: "Microsoft YaHei UI"; color: rgb(0, 0, 0); line-height: 1.5; }body { font-size: 14px; font-family: "Microsoft YaHei UI"; color: rgb(0, 0, 0); line-height: 1.5; }</style></head><body>
表情包文化，是随着网络社交沟通的增多出现的一种主流文化。一个人的表情包是其隐蔽起来的真我，一个国家的表情包里能看到这个国家的表情。有时候，表情包表达的是不能道破的真实想法和感受，语言和文字的尽头，就是表情包施展的空间。 表情包是网络语言的一种进化，它的产生和流行与其特定的“生存环境”有关。其追求醒目、新奇、谐谑等效果的特点，与年轻人张扬个性和搞怪的心理相符。 表情包之所以能够大范围地传播，是因为其弥补了文字交流的枯燥和态度表达不准确的弱点，有效地提高了沟通效率。部分表情包具有替代文字的功能，还可以节省打字时间。随着智能手机的全面普及和社交应用软件的大量使用，表情包已经高频率地出现在人们的网络聊天对话当中。 <img src="cid:_Foxmail.1@e1a633df-2fe4-f85c-7270-ca78308ac1c5" border="0"></body></html>

Hidden Text: Clear (length: 7)

hurryup

Encode »

« Decode

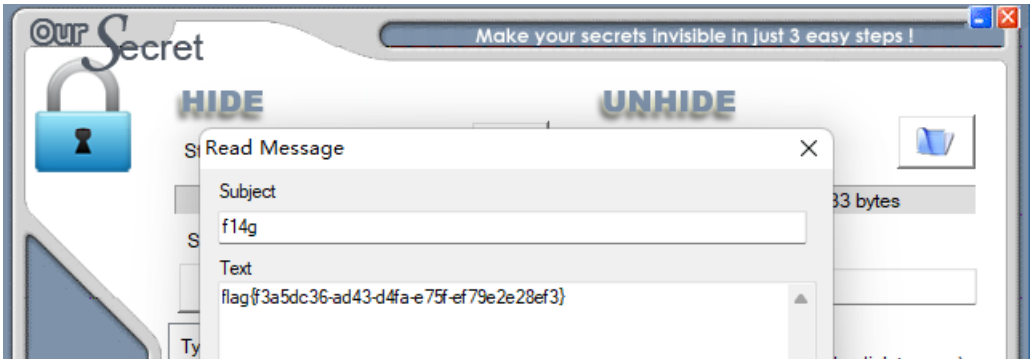
Steganography Text: Clear (length: 824)

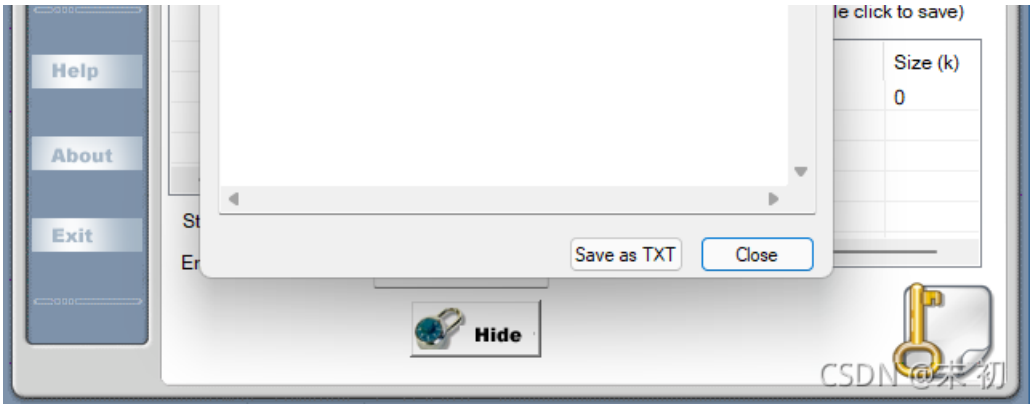
<html><head><meta http-equiv="content-type" content="text/html; charset=UTF-8"><style>body { line-height: 1.5; }body { font-size: 14px; font-family: "Microsoft YaHei UI"; color: rgb(0, 0, 0); line-height: 1.5; }body { font-size: 14px; font-family: "Microsoft YaHei UI"; color: rgb(0, 0, 0); line-height: 1.5; }</style></head><body>
表情包文化，是随着网络社交沟通的增多出现的一种主流文化。一个人的表情包是其隐蔽起来的真我，一个国家的表情包里能看到这个国家的表情。有时候，表情包表达的是不能道破的真实想法和感受，语言和文字的尽头，就是表情包施展的空间。 表情包是网络语言的一种进化，它的产生和流行与其特定的“生存环境”有关。其追求醒目、新奇、谐谑等效果的特点，与年轻人张扬个性和搞怪的心理相符。 表情包之所以能够大范围地传播，是因为其弥补了文字交流的枯燥和态度表达不准确的弱点，有效地提高了沟通效率。部分表情包具有替代文字的功能，还可以节省打字时间。随着智能手机的全面普及和社交应用软件的大量使用，表情包已经高频率地出现在人们的网络聊天对话当中。 <img src="cid:_Foxmail.1@e1a633df-2fe4-f85c-7270-ca78308ac1c5" border="0"></body></html>

Download Stego Text as File

CSDN @末 初

两段话都隐写了密码






```
flag{f3a5dc36-ad43-d4fa-e75f-ef79e2e28ef3}
```

JumpJumpTiger

```
root@mochu7-pc:/mnt/c/Users/Administrator/Downloads# file jump.exe
jump.exe: PE32+ executable (console) x86-64, for MS Windows
root@mochu7-pc:/mnt/c/Users/Administrator/Downloads#
```

丢进 `ida64` 分析， `main` 函数是个提示

```
int __cdecl main(int argc, const char **argv, const char **envp)
{
  int v4[201]; // [rsp+20h] [rbp-60h]
  int v5; // [rsp+344h] [rbp+2C4h]
  int v6; // [rsp+348h] [rbp+2C8h]
  int i; // [rsp+34Ch] [rbp+2CCh]

  _main();
  printf("This is your hint!!!");
  v6 = 0;
  v5 = 0;
  for ( i = 0; i <= 99; ++i )
  {
    if ( (i & 1) != 0 )
      v4[v5++] = i;
    else
      v4[v6++ + 100] = i;
  }
  return 0;
}
```

CSDN @末 初

大概就是个这么的意思，根据奇偶位分数据

process.py

```python
v4 = '/i9VjB/O4RAwA0QKSGkgZoJARAgAAABNASQUEhAEAUQgAABAABA4DA/A2AwABQD4A'
v5 = ''
v6 = ''
for i in range(len(v4)):
    if (i & 1) != 0:
        v5 += v4[i]
    else:
        v6 += v4[i]

print(v5)
print(v6)
```

PowerShell / kali-linux

```
PS C:\Users\Administrator\Downloads> python .\process.py
iVBORw0KGgoAAAANSUhEUgAAB4AAAAQ4
/9j/4AAQSkZJRgABAQEAAQABAAD/2wBDA
PS C:\Users\Administrator\Downloads>
```

CSDN @末 初

在程序中还发现了大量字符串

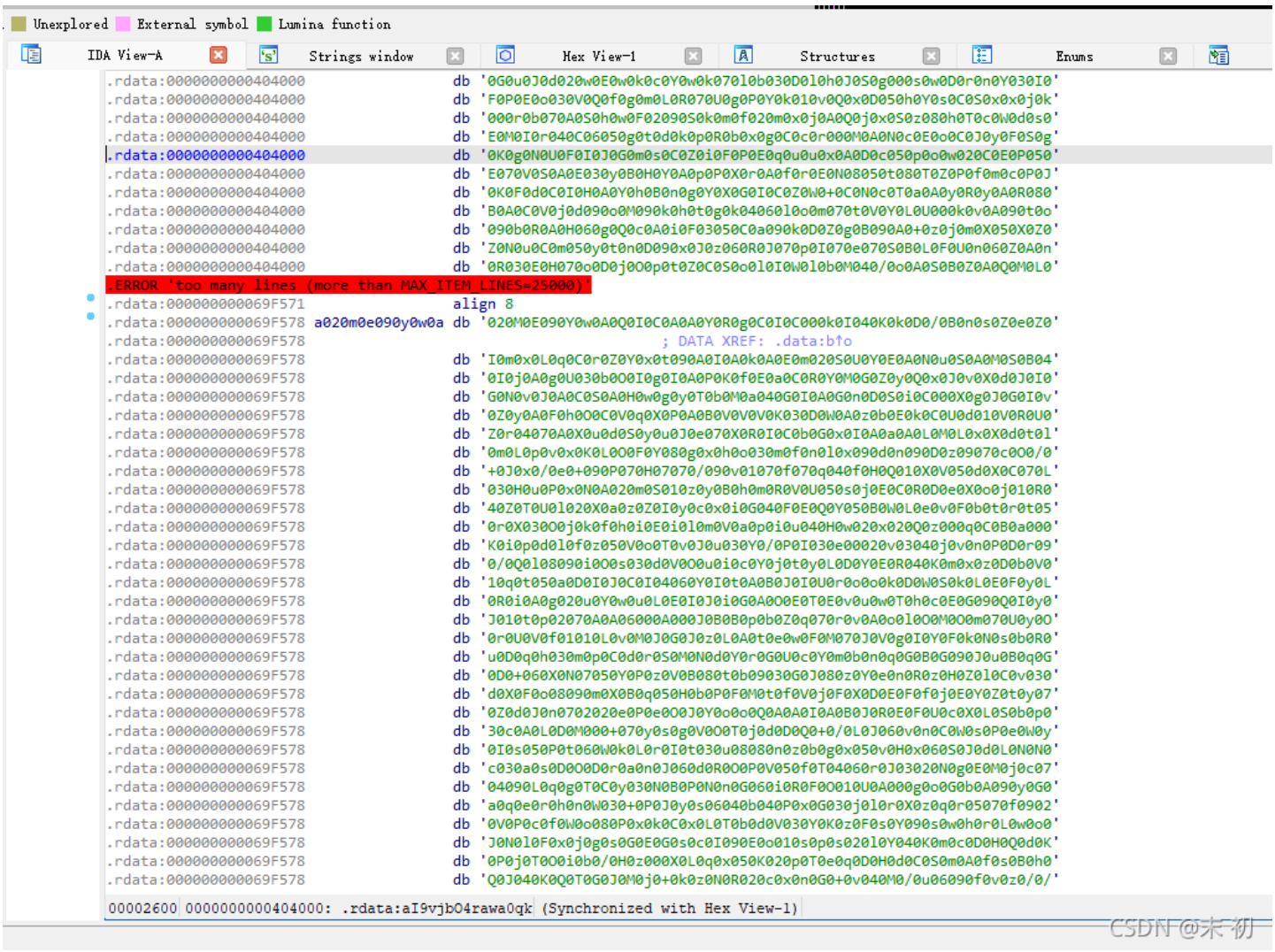
IDA View-A | Strings window | Pseudocode-A | Hex View-1 | Structures

```
.rdata:0000000000404000                 assume cs:_rdata
.rdata:0000000000404000                 ;org 404000h
.rdata:0000000000404000 aI9vjbO4rawa0qk db '/i9VjB/O4RAwA0QKSGkgZoJARAgAAABNASQUEhAEAUQgAABAABA4DA/A2AwABQD4A'
.rdata:0000000000404000                                         ; DATA XREF: .data:a1o
.rdata:0000000000404000                 db 'CAAUIDABAAAQBEnAswVUYEUBAAAQAFgBAQEUlGEBQwVwRI4BAwWcTHBBQwZ8YLsC5'
.rdata:0000000000404000                 db 'w5kaMcE1Q88/SsEuhCEcPAEURvEOTfFFhdwUXiERx8QBaaFaRqEtRBGsCtE7YNG+h'
.rdata:0000000000404000                 db 'I08dZHIxx8xfIE6xtcbiSJ3CvIrevJ/B/wWe/H9xA7f/Q2NwkBnDbAJQJUJFsBXQ9'
.rdata:0000000000404000                 db 'ccG1BMw74aIBCtAr4veLFQBxEKU/HShZ4ee3HChm4xefHYhk4CeAH/hN42eqHrhT4'
.rdata:0000000000404000                 db '4eqHkh54wekHbhY4GeKHRh14QeMHKhr44eGHUhP4leIH1hX4beTH9hh4eezHuh74b'
.rdata:0000000000404000                 db 'e/H/hP4MePHSha4/eXHKhz4me2H/h77j/Vw9ApAyRzCuA5QK4uB645ALDEAWR9E0A'
.rdata:0000000000404000                 db 'fAkhmExBLAsxtE0B2/j87QYAjH8QcAOA6AJgPIHDoAIQ0EnBbA/A3ARA7A5ARAXAH'
.rdata:0000000000404000                 db 'ALAgAZAUAAQUIcAtABwkQJFJBjgSc7IbCTfO/aEbApEGU3QeAcQEAcBdA1wxMoDEA'
.rdata:0000000000404000                 db 'MwkMNDtAUwGMkDXAmwLElC5DtwEE1COAHAmM1RFBbBLI3hZBRTmF7BVBqipJFRtBo'
.rdata:0000000000404000                 db 'xGN1h6M2ngGWBkF2ChO6hrCLESKARZFfVbKRxNFkiNQQzWwadAEul0YlncISXbNRE'
.rdata:0000000000404000                 db 'fMom2J8+mHUxbgthWhWTTY63RPtwg/a82QaAAGzweEuAcAEw7EiBuASQ9ELBuAtAp'
.rdata:0000000000404000                 db 'A0AZA2A0ADArANAnA1AAA5ANEqCWAdwLQYFoBkgvfy/mxiA0A63rEVQuA0CCAbgYI'
.rdata:0000000000404000                 db 'GC2AQgFItDAAEAoI+B2B/ACIZBoAxABsiA3A6QTIkRhAFyLEkSbMaQWRxBfB1V9Ed'
.rdata:0000000000404000                 db 'TlIomeEyUeM3nREeGWIP0yK6BqkVakElV4M77FEFkZU2s/Hcw9FRtuFkDBYbuJH2/'
.rdata:0000000000404000                 db 's2JgEAAMSAjw7EZAAAzhkEsDaEQQHAL/VAdPjoemvW5JNKhVuyCn7aaNuCmPd2s4a'
.rdata:0000000000404000                 db 'OcGNgsC1Lui9OuazAkDmEb4GJKZMwnNuHXLticOFK9sIXcek7285HJJ45JpNcnkm3'
.rdata:0000000000404000                 db 'zRfXbE1MYyIMzXUc9SnyvPPYforc+SyW6qzZIDyd9j0dgICyJfLFUHFE+pkOOhSEb'
.rdata:0000000000404000                 db 'ITQ2EActiv5xxoQmy6SQwPR2y37reYcldp6Tzkon0NAPZW/PUyUTwZIBGgVrV5oME'
.rdata:0000000000404000                 db 'X3PQvz7hshxz8RfTXF2y1G/if5Xhr+1azcnUHyNZsOQ9PkQDNkAaH1MeDjOJSd2ju'
.rdata:0000000000404000                 db 'k6CjXrIS8T0b4kquLYFBP7oKfMgtMDP0emt2Egob0YT8E9RL5A4tkK5crkN5lJE4I'
.rdata:0000000000404000                 db 'k4kO2HxPP46qAYC9TjxZaGLKtzmOHKmq2QThtyKMts/uTeBC5otQe2dDi2W5x5Mq1'
.rdata:0000000000404000                 db 'hbBHc9B6xO5doMGFnlRpC3M/tlrcldw5N2QAxU2mG2C5ZU/0bCiXiNgdGlAmJ4bdn'
.rdata:0000000000404000                 db 'ttwRLVagkI3yfS3S+R/etK5adVYMGkaYatUq8ZyZbcx/9FSUqAWN6ov8yZl5Ro+AA'
.rdata:0000000000404000                 db 'AJYN5ulSygHDi6pWf8Yq/PQC46mJMyxTM5RH84lAN+92Aii3MFpcOiD6ZfiNL02za'
.rdata:0000000000404000                 db 'emoLN3UQFi6QRj8IQcwZBwpd66Cu3QN0eCWdUxf3kbqY7lrQmOhimIgi7Za5uH/xg'
.rdata:0000000000404000                 db 'SEfsQGNVAEVTje22MjZqong+KKg2s2VmMInpnWvoWXgckLgtyNgu5PMoPENiA16fA'
.rdata:0000000000404000                 db 'fZl3+YNxa7+j+zXQOT7LgQs6NkppXDMc/5XCbgzhEdhkqQme6zQxtQ4MOmtGPJZib'
.rdata:0000000000404000                 db 'gOCvb5dryANrA+BUl1HzKlfwiXg6AShNOJRWhD45KeB2Muk5hvIE79Znd3sf5Mqwh'
.rdata:0000000000404000                 db 'ODzzcj5FKg0E5ecGfpMGXiNtehlvXdKrFLnCve8a8B5kb2by+BkJsSVyXHOa9goAO'
.rdata:0000000000404000                 db 'Q1HXASrixxQfBVZ1PXar4ORyyeFcA7CJmV3aLmklo8F+bzJEHzcbyXR+M9Bv2ppNp'
.rdata:0000000000404000                 db 'cW1F+jX77YsvnP3m2lVL5QKwzM28z4fYXntNsSZlAYlR00uweSSI0yaRKl83qqI7L'
.rdata:0000000000404000                 db 'N2GPK2D1yeRQlghszzS9EiGuU1jVbZ/SA2H4HQKsdkqmAZBuH2G7MdhSS+fdQD1v2'
.rdata:0000000000404000                 db 'PTfIvu6c5cX1PJ+QkFLG+esMcUwCMZpH2E6Mb4d5VFg58o3Aocl2+ht7eDZCt1vSO'
.rdata:0000000000404000                 db '69EKyILeN/9Nsqey503NaSK233Y1q8IXYXtc5rUayntvVsYyq+Kb9R2RXIBB2tqSJ'
.rdata:0000000000404000                 db 'TOzi6o2oJD7uXi6h54VakmZMqi7prI8kkxMySIk0E3dsbVQhDu9weWrGGS2XgP7md'
.rdata:0000000000404000                 db 'pma0KnjU7fBbFmiQ2Fi2Kt46l3StKtKu7eucY6J7lpy4/SNJJq95lgxb6bF5XTOSF'
.rdata:0000000000404000                 db 'Wa/QTwWS0z0BlUM/3Vpi7goLqAxPfseJlHK6msr0eB1YBX6vnlQ5Nl0inh2s2uGP4'
.rdata:0000000000404000                 db 'YrtyrTGbFQTNksanEu1J7vKnYX26I1sutSy7cf1uS+cZR0dZBStH5pkLsynmhJOnC'
.rdata:0000000000404000                 db 'rquuoJDLXkoL6H2Akqm8btkQe0khp8LbsBbGlatKJJfuoPhjp4mp4ed9tDQWAg8Se'
.rdata:0000000000404000                 db 'Yb5YUFjztjamA3cdrED+DjEDuYVIwQlFVJENQntd6JLLKaZ/Ivlrk7KuU+uzh7pf0'
00002600 0000000000404000: .rdata:aI9vjbO4rawa0qk (Synchronized with Hex View-1)
```

CSDN @末 初

`iVB...`开头的很明显是PNG图片的base64数据开头，`/9j/...`开头的很明显是`JPG`的base64数据开头；从0开始奇数位是`PNG`数据，偶数位是`JPG`数据

另外除了这部分base64数据之外，还发现了夹杂着`0`的base64数据



先把base64数据提取出来，数据太长了，不便复制，使用脚本来提取比较方便，Python简单处理即可

```
from base64 import *

start_opt = 0x2600
middle_opt = 0x10DF00
end_opt = 0x7D4660
png_data = ''
jpg_data = ''

with open('jump.exe', 'rb') as f:
 f.seek(start_opt)
 part1_base64 = f.read(middle_opt - start_opt).decode()
 for i in range(len(part1_base64)):
  if i % 2 == 0:
   jpg_data += part1_base64[i]
  else:
   png_data += part1_base64[i]

with open('flag.jpg', 'wb') as f1:
 f1.write(b64decode(jpg_data))

with open('flag.png', 'wb') as f2:
 f2.write(b64decode(png_data))
```

得到两张图，但是`png`的图片数据并不完整；猜测另一部分的png图片的base64数据即为之前参杂0的数据

flag.jpg

flag.png CSDN @末 初

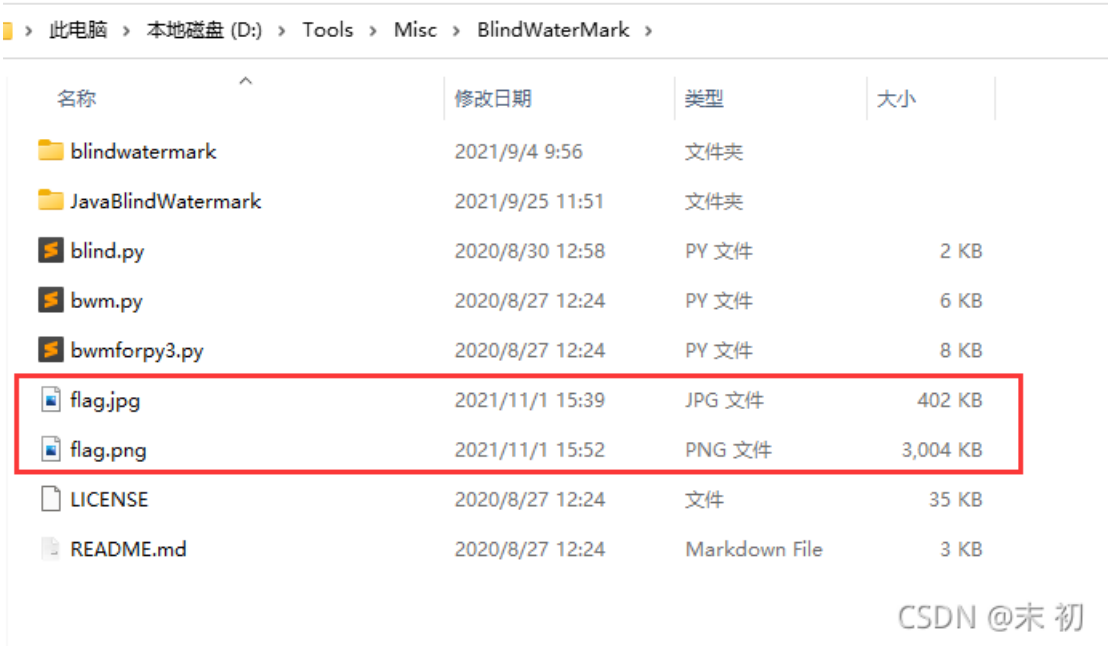
继续使用Python简单处理

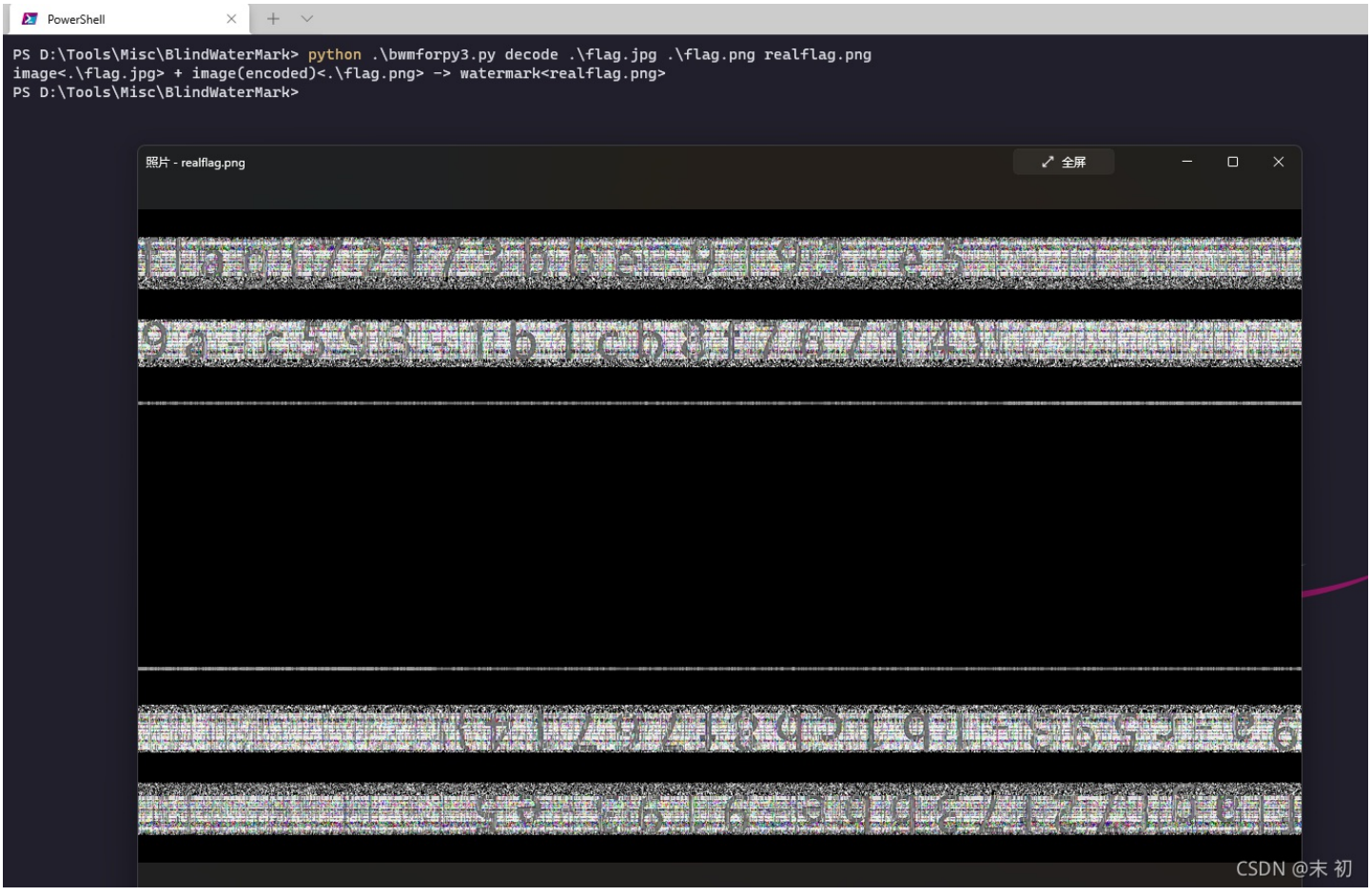
```
from base64 import *

start_opt = 0x2600
middle_opt = 0x10DF00
end_opt = 0x7D4660
png_data = ''

with open('jump.exe', 'rb') as f:
 f.seek(start_opt)
 part1_base64 = f.read(middle_opt - start_opt).decode()
 for i in range(len(part1_base64)):
  if i % 2 == 0:
   pass
  else:
   png_data += part1_base64[i]
 f.seek(middle_opt)
 part2_base64 = f.read(end_opt - middle_opt).decode()
 for i1 in range(len(part2_base64)):
  if i1 % 2 == 0:
   pass
  else:
   png_data += part2_base64[i1]

with open('flag.png', 'wb') as f2:
 f2.write(b64decode(png_data))
```

flag.png

flag.jpg CSDN @末 初

盲水印





看不清的话用 `stegsolve` 调整一下

```
flag{72f73bbe-9193-e59a-c593-1b1cb8f76714}
```

where_can_find_code

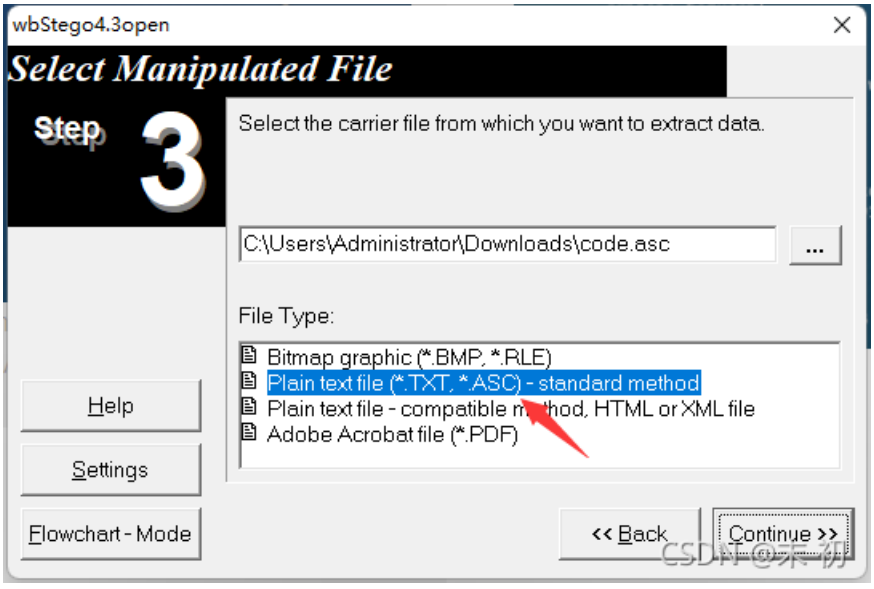
```
format("Translate the letter J into I");
dpeb{e58ca5e2-2c51-4eef-5f5e-33539364deoa}
```

kali-linux

```
<main>Where can find code?</main>
<style>
  @font-face {
      src: url("https://www.axis-praxis.org/fonts/webfonts/MetaVariableDemo-Set.woff2")
      format("woff2");
      font-family: "Meta";
      font-style: normal;
      font-weight: normal;
  }

  body {
      box-sizing: border-box;
      margin: 0;
      padding: 0;
      display: flex;
      justify-content: space-evenly;
      align-items: center;
      background-color: #8357eb;
      width: 100vw;
      height: 100vh;
  }

  main {
      transition: all 0.5s;
      -webkit-text-stroke: 4px #d6f4f4;
      font-variation-settings: "wght" 900, "ital" 1;
        font-size: 15rem;
        text-align: center;
        color: transparent;
        font-family: "Meta", sans-serif;
        text-shadow: 10px 10px 0px #07bccc,
        15px 15px 0px #e601c0,
        20px 20px 0px #e9019a,
        25px 25px 0px #f40468,
        45px 45px 10px #482896;
        cursor: pointer;
    }

    main:hover {
        font-variation-settings: "wght" 100, "ital" 0;
        text-shadow: none;
    }
</style>
<font>
format("Translate the letter J into I");
dpeb{e58ca5e2-2c51-4eef-5f5e-33539364deoa}
</font>
~
~
~
"code.asc" [noeol][dos] 47L, 1247B
```

CSDN @末 初

wbstego隐写，无密码



CSDN @末 初

得到云影密码

```
20810842042108421
```

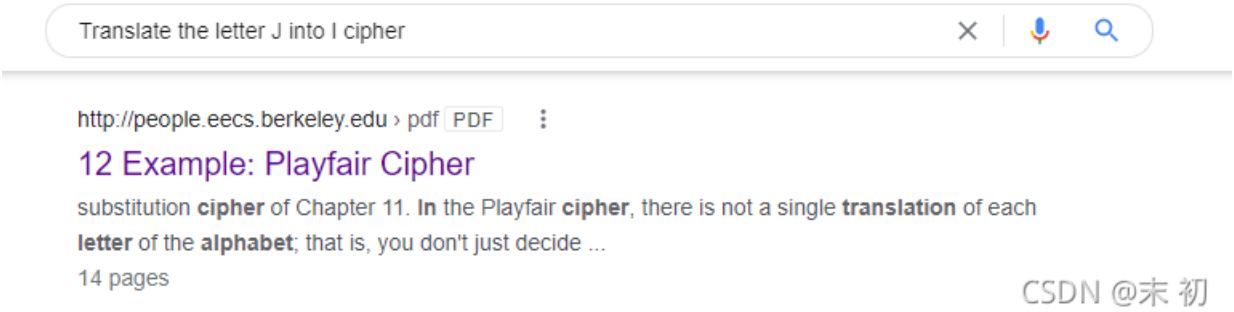
Python简单处理转换即可

```
code = '20810842042108421'
code_list = code.split('0')
alphabet = 'ABCDEFGHIJKLMNOPQRSTUVWXYZ'
text = ''
for nums in code_list:
 plus_num = 0
 for n in nums:
  plus_num += int(n)
 text += alphabet[plus_num - 1]
print('[+]: {}'.format(text))
```

```
PS C:\Users\Administrator\Downloads> python code.py
[+]: BINGO
```

接下来就是利用一条这条语句

```
Translate the letter J into I
```



12 Example: Playfair Cipher

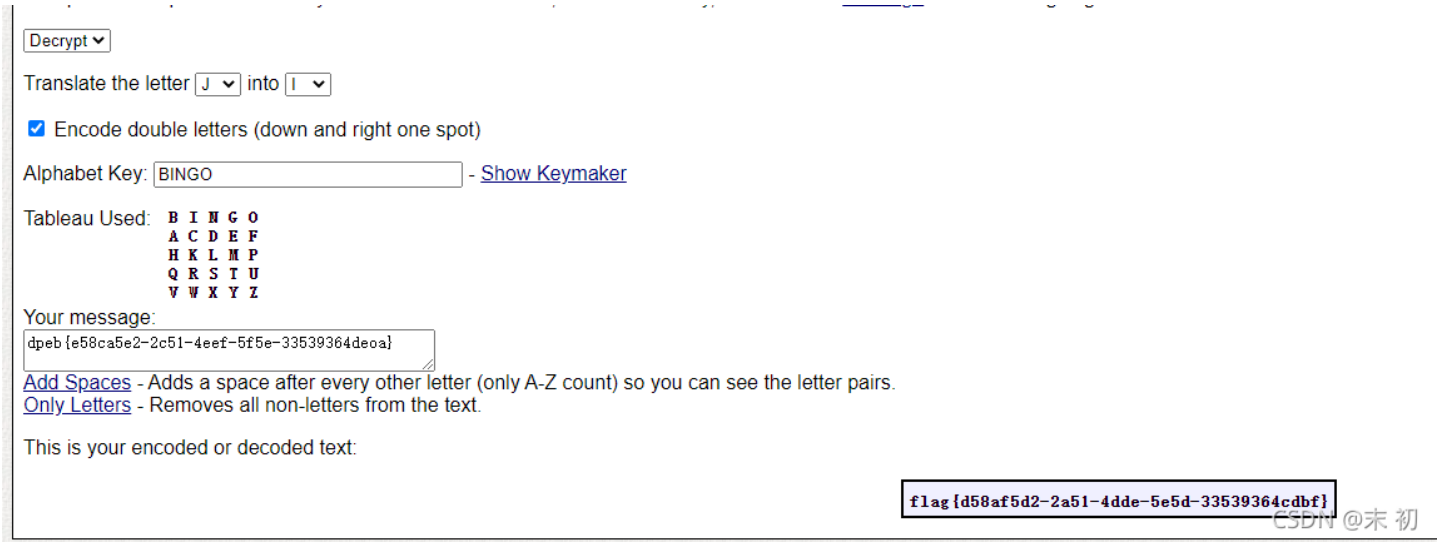
Program file for this chapter: `playfair`

This project investigates a cipher that is somewhat more complicated than the simple substitution cipher of Chapter 11. In the Playfair cipher, there is not a single translation of each letter of the alphabet; that is, you don't just decide that every B will be turned into an F. Instead, *pairs* of letters are translated into other pairs of letters.

Here is how it works. To start, pick a *keyword* that does not contain any letter more than once. For example, I'll pick the word `keyword`. Now write the letters of that word in the first squares of a five by five matrix:

CSDN @末 初

- http://rumkin.com/tools/cipher/playfair.php




```
flag{d58af5d2-2a51-4dde-5e5d-33539364cdbf}
```