

Hgame week1 writeup

web

1.谁吃了我的flag

看到这道题，点开URL，发现flag只有前面一段，后面一段并没有。

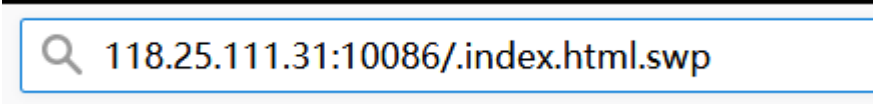
damn...hgame2019 is coming soon, but the stupid Mki haven't finished his web-challenge...

fine, nothing serious, just give you flag this time...

the flag is hgame{3eek_diScI0Sure

然后题目中说了 难道没有好好关机吗?? (没有好好关机是不是就是突然关机呢??)

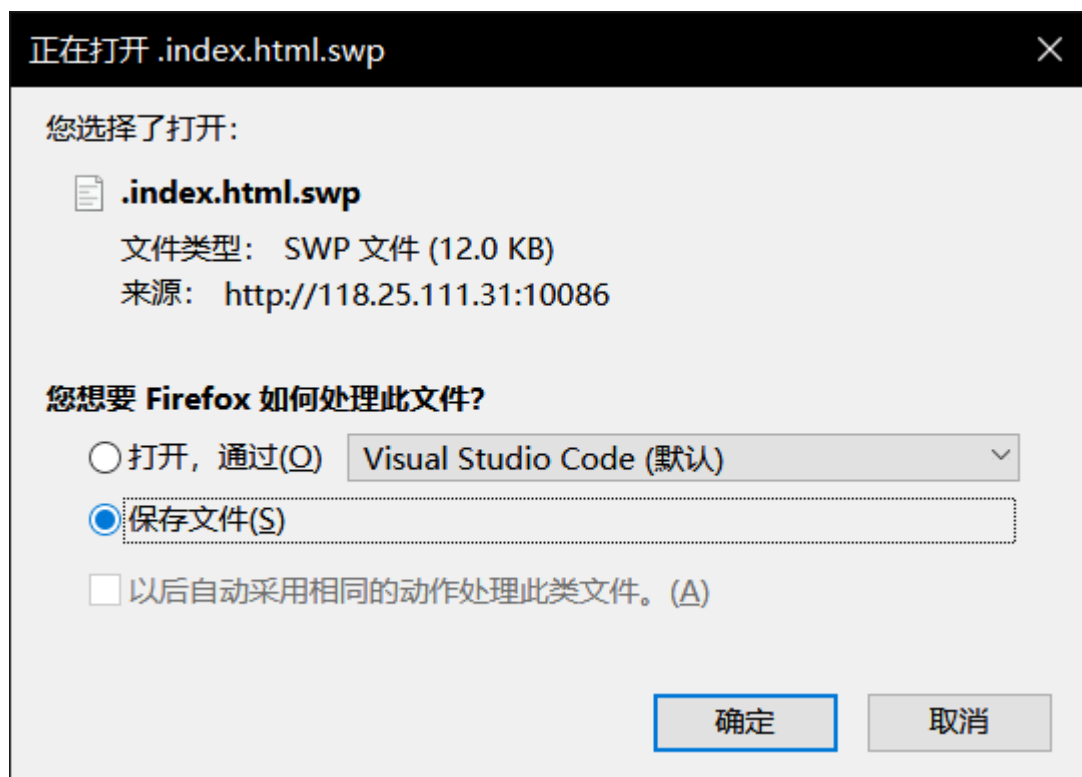
然后印象中突然关机有的会有临时交换文件，百度了下vim的临时文件是.swp，然后输入了



118.25.111.31:10086/.index.html.swp

(index前的.号第一次漏了)

进入了之后发现会弹出这个



保存打开 得到flag is `hgame{3eek_disc10Sure_fRom+wEbsit@}`

2.换头大作战

这题描述中提到了burpsuite, 所以启动burpsuite, 然后打开URL

发现

想要flag嘛:

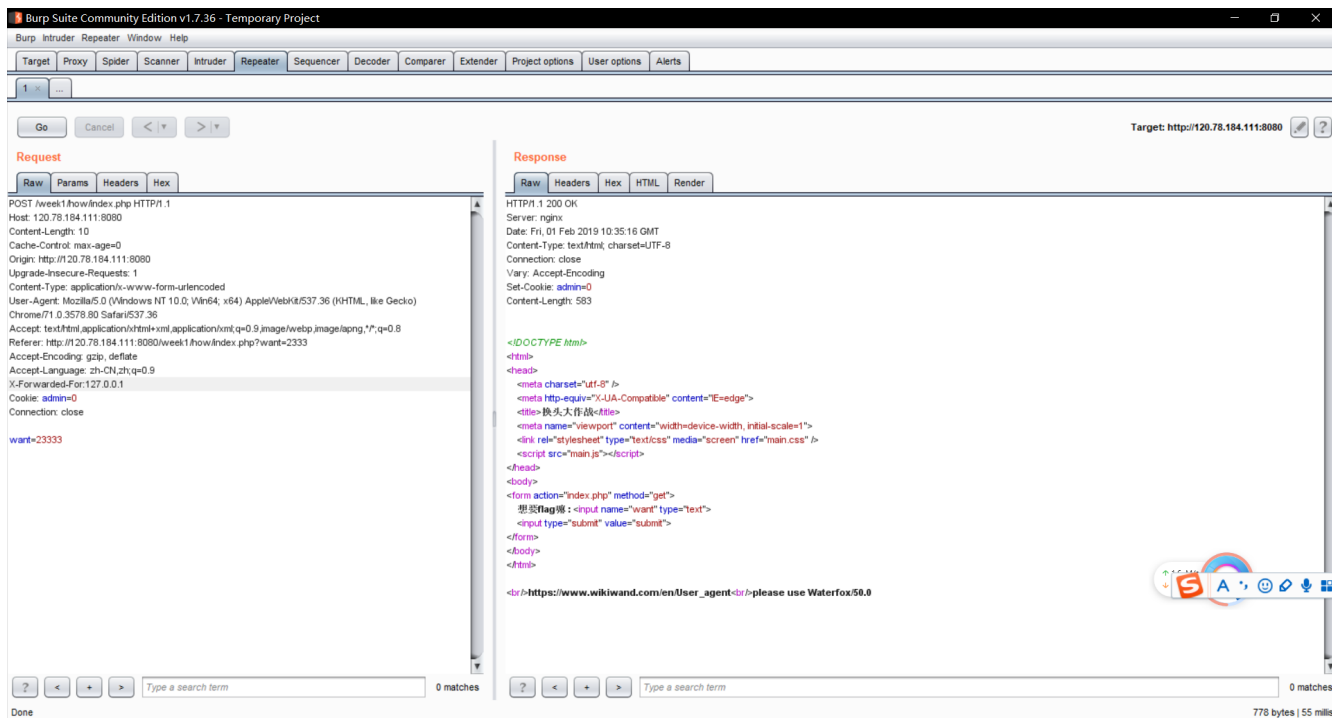
那么随便输了几个, 发现

request method is error.I think POST is better

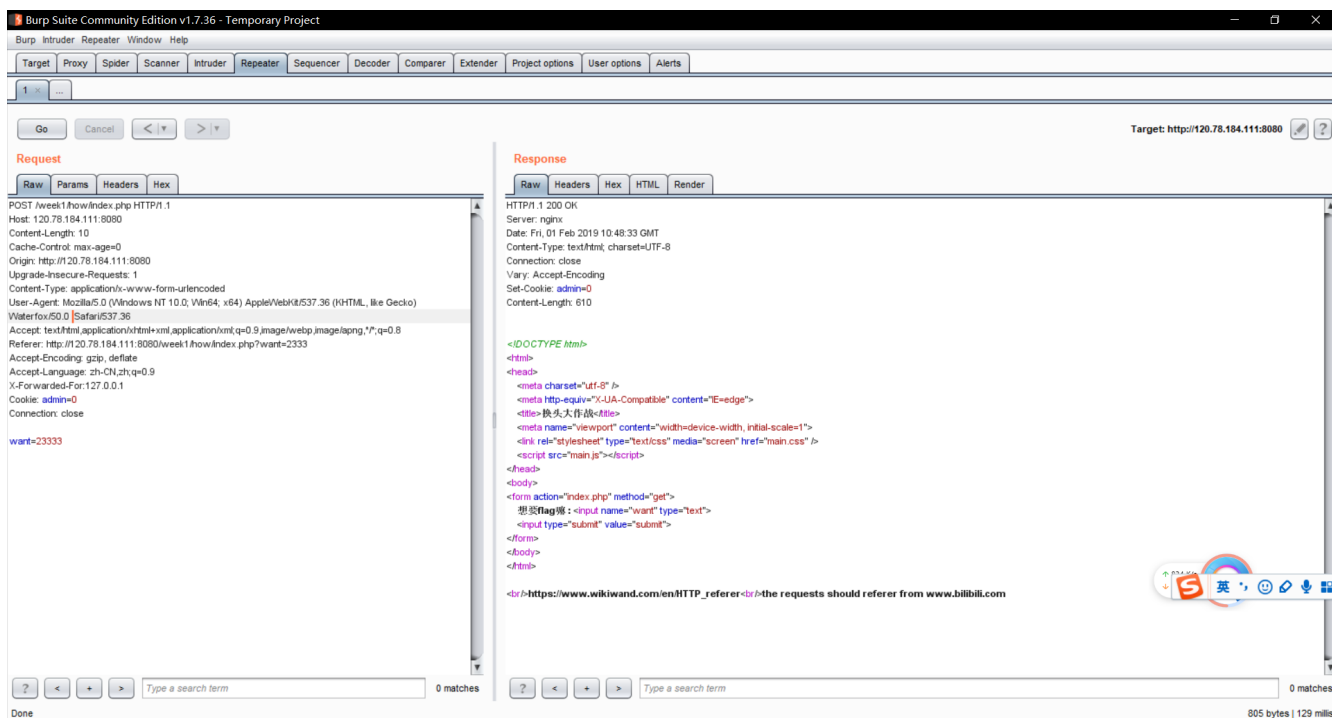
那么, 把GET请求方式改成了POST请求, 发现

<https://www.wikiwand.com/en/X-Forwarded-For>
only localhost can get flag

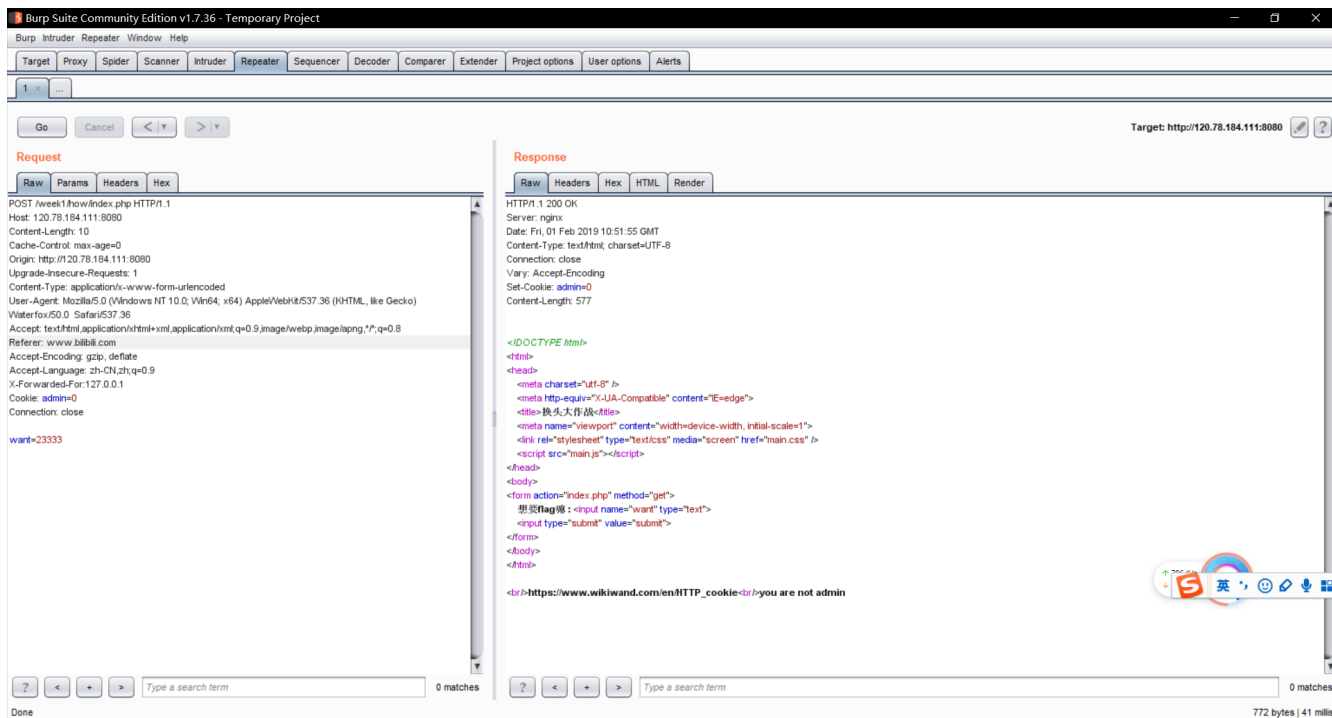
然后burp里面抓到那个POST请求的包, repeater一下, 然后在cookie上加了一行



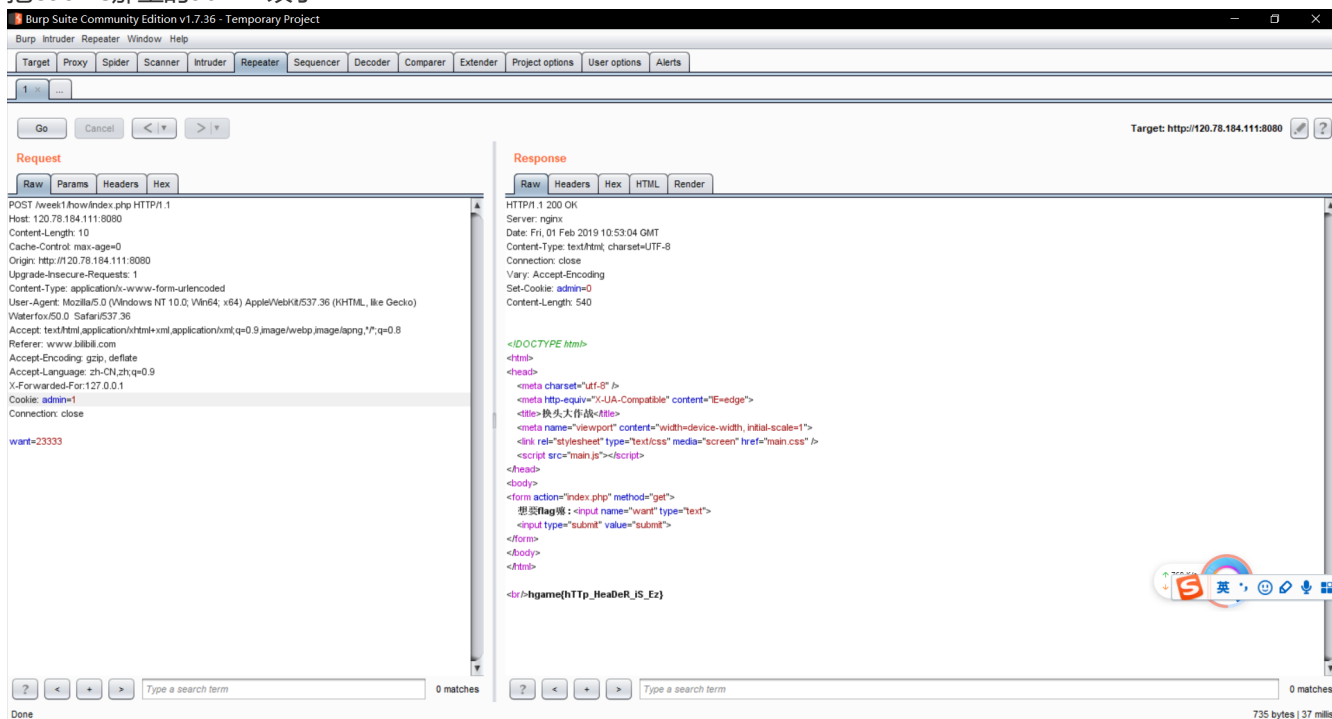
然后看到了Waterfox/50.0,百度了下发现waterfox是浏览器，那么继续



然后继续



把cookie那里的admin改了



出现了flag `hgame{hTtp_HeaDeR_iS_Ez}`

3.very easy web

进入URL发现

```
<?php
error_reporting(0);
include("flag.php");

if(strpos("vidar", $_GET['id'])!==FALSE)
    die("<p>干巴爹</p>");

$_GET['id'] = urldecode($_GET['id']);
if($_GET['id'] === "vidar")
{
    echo $flag;
}
highlight_file(__FILE__);
?>
```

好像要是某id通过URL解码===vidar，那么百度手写出了vidar的URL编码是%76%69%64%61%72

输入了

```
120.78.184.111:8080/week1/very_ez/index.php?id=%76%69%64%61%72
```

发现会变成

```
120.78.184.111:8080/week1/very_ez/index.php?id=vidar
```

那就是浏览器自己URL解码了一次，所以vidar的URL编码一次不行，那么就试试两次

```
120.78.184.111:8080/week1/very_ez/index.php?id=%2576%2569%2564%2561%2572
```

出现了flag hgame{ur1Decode_Is_GoOd}

4.can u find me ?

the gate has been hidden

can you find it? xixixi

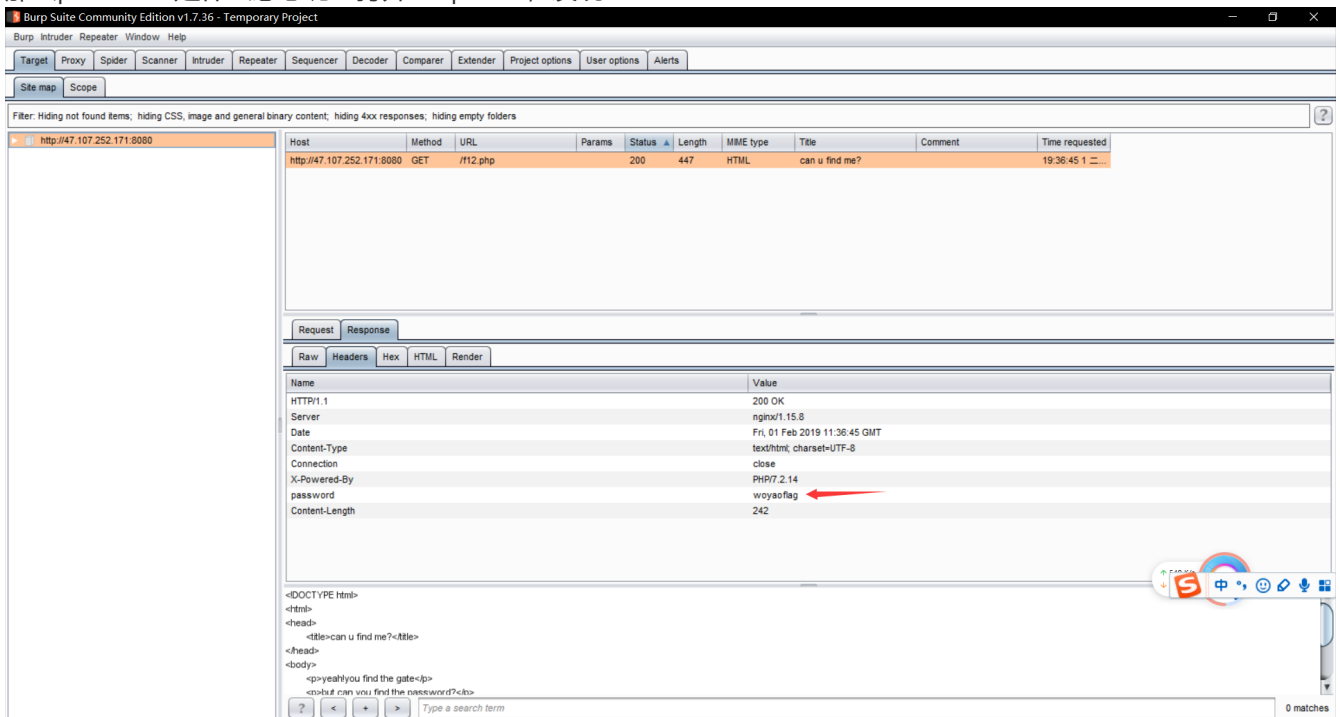
打开F12，进入f12.php，会发现post password

yeah!you find the gate

but can you find the password?

please post password to me! I will open the gate for you!

那么password是什么意思呢? 打开burpsuite, 发现:



Filter: Hiding not found items; hiding CSS, image and general binary content; hiding 4xx responses; hiding empty folders

Host	Method	URL	Params	Status	Length	MIME type	Title	Comment	Time requested
http://47.107.252.171:8080	GET	/f12.php		200	447	HTML	can u find me?		19:36:45 1 2 ...

Request Response

Raw Headers Hex HTML Render

Name	Value
HTTP/1.1	200 OK
Server	nginx/1.15.8
Date	Fri, 01 Feb 2019 11:36:45 GMT
Content-Type	text/html; charset=UTF-8
Connection	close
X-Powered-By	PHP/7.2.14
password	woyaoflag
Content-Length	242

<!DOCTYPE html>
<html>
<head>
<title>can u find me?</title>
</head>
<body>
<p>yeah!you find the gate</p>
<p>but can you find the password?</p>

0 matches

蛤蛤蛤, woyaoflag, 然后在线POST请求

例: <http://coolaf.com/m?a=xx&b=xx>,get参数直接加在url后就行。用户本地web版本提供下载,用户可以访问本地域名, [下载链接](#)

post 参数:

参数: a=b&c=d&f=e,如果传递参数是 json,请修改高级中header为: Content-Type:application/json 显示高级功能

POST

UTF-8 --接口输出的编码

自动解压(gzip,deflate,flate)

提交

生成文档

测试示例

导出历史

清空表单

格式化Response

原始Response

Headers

分享请求

网站接入

1549021125 2019-02-01 19:38:45

复制

```

<!DOCTYPE html>
<html>
<head>
  <title>can u find me?</title>
</head>
<body>
  <p>yeah!you find the gate</p>
  <p>but can you find the password?</p>
  <p>please post password to me! I will open the gate for you!</p>
  <p>right!</p><a href='iamflag.php'> click me to get flag</a></body>
</html>

```

进入URL

47.107.252.171:8080/toofast.php

然后在burpsuite里发现flag `hgame{f12_1s_aMazIng111}`

Re

1.brainfxxker

这道题刚开始一直没读懂brainfuck的意思,后面的补充说明告诉了我们了,不执行[+.]这个部分,也就是说前一个车厢到[+.]前刚刚变成0,即第一部分为循环次数,第二部分为每次减少数量,第三部分为加减一个数(只要符合这个字符的ASCII码-1*2+3即可)

```

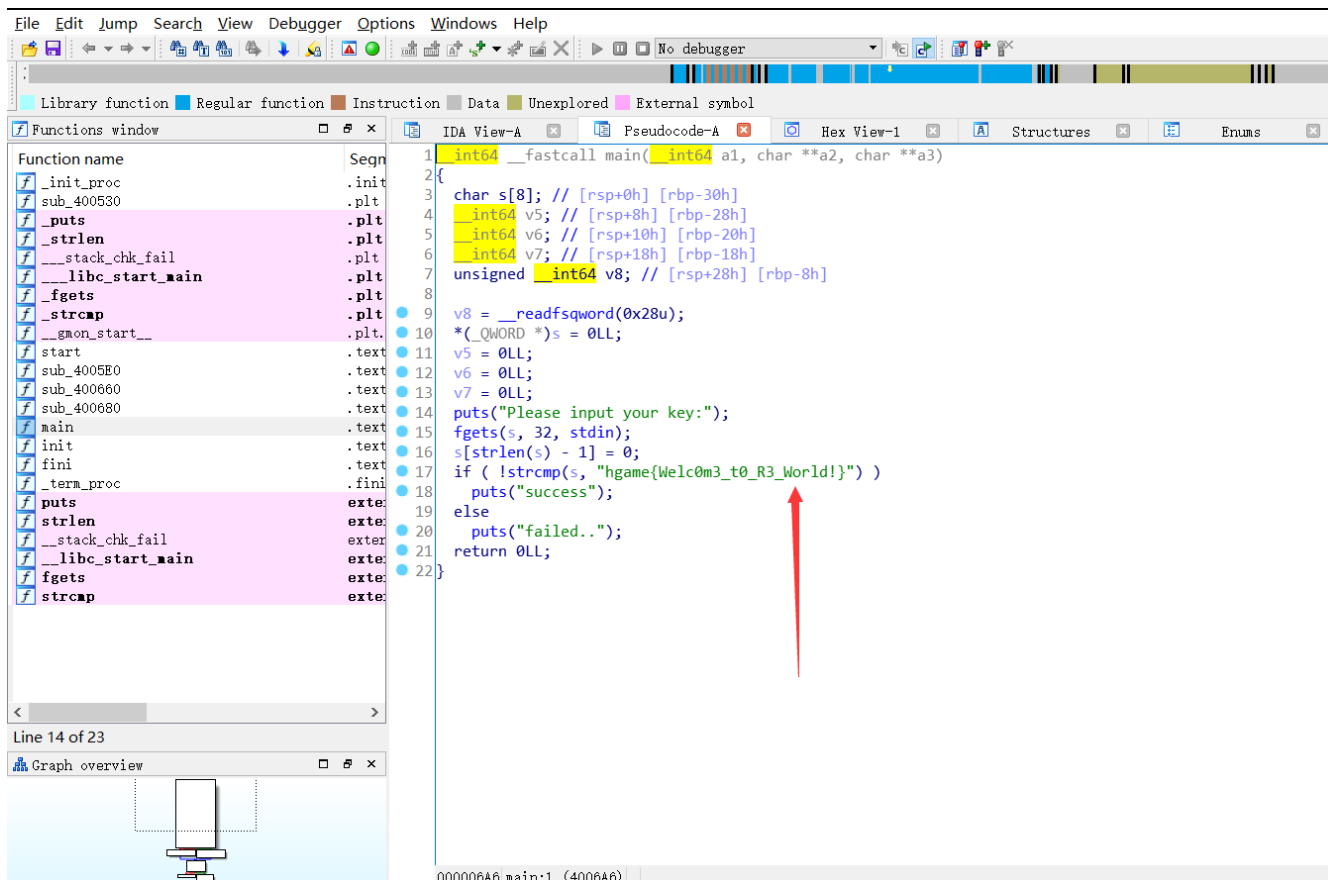
,>+++++++[<----->-]<+{+.}
  1       2       3

```

所以可以推出来flag = `hgame{br4!NfUcK}`

2.HelloRe

说实话,逆向基本不会做,这题看到,直接扔到IDA里,找到main,按F5



得到flag `hgame{we1c0m3_t0_R3_Wor1d!}`

3.Pro的Python教室（一）

这道题因为没学过Python就根据C语言语法来理解了，面向百度查函数意义

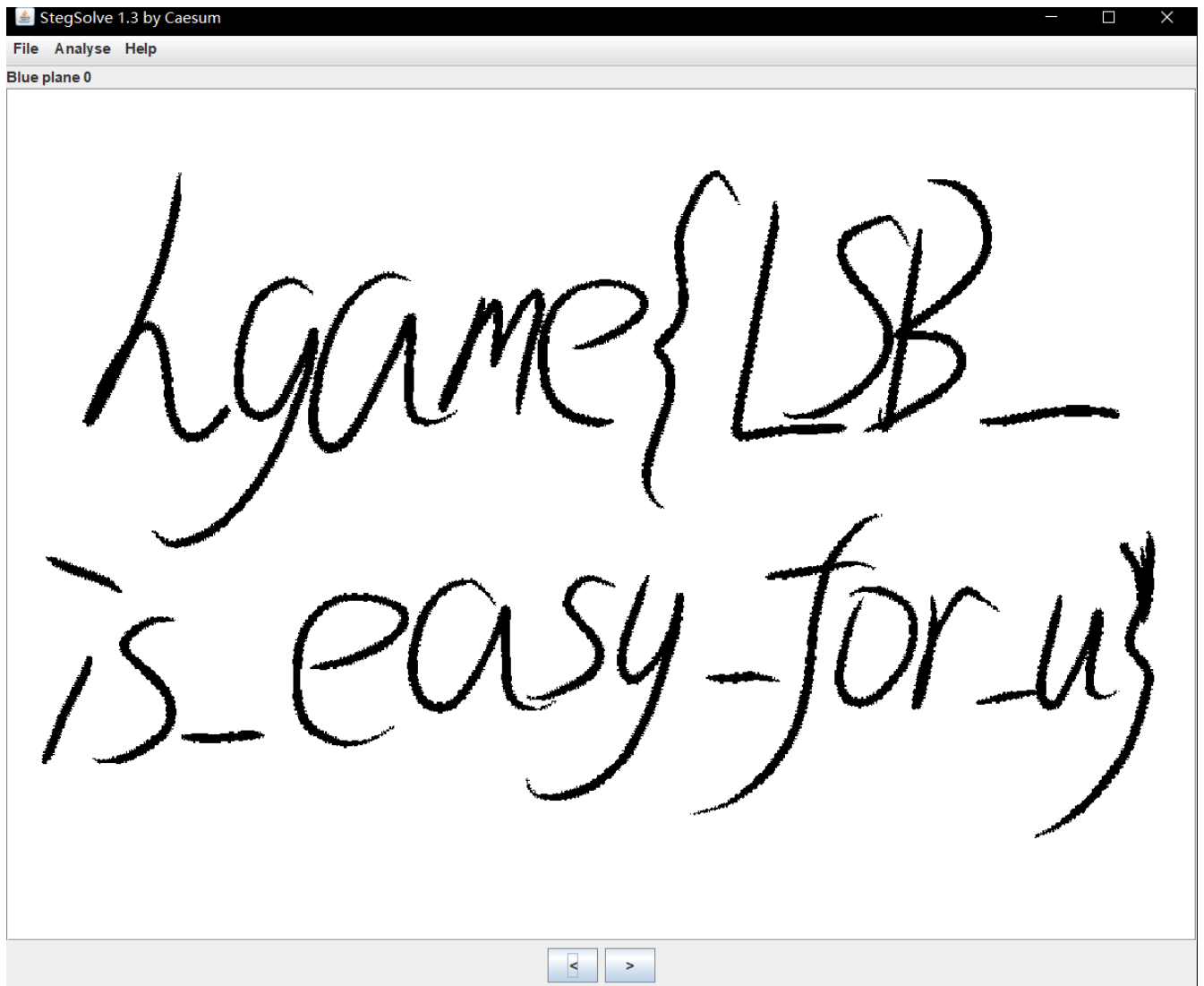
`secend = base64.b64encode(secend)` 当中出现了这个，应该是你输入的secend经过一次base64编码等于题目说的enc2,从而输入的就是enc2的base64解码。（enc3那里也一直以为是这个套路，但是总不对，后来看enc3那里本身挺像flag尾部的，就去试试了，结果对了）

得到flag `hgame{Here_1s_3asy_Pyth0n}`

MISC

1.Hidden Image in LSB

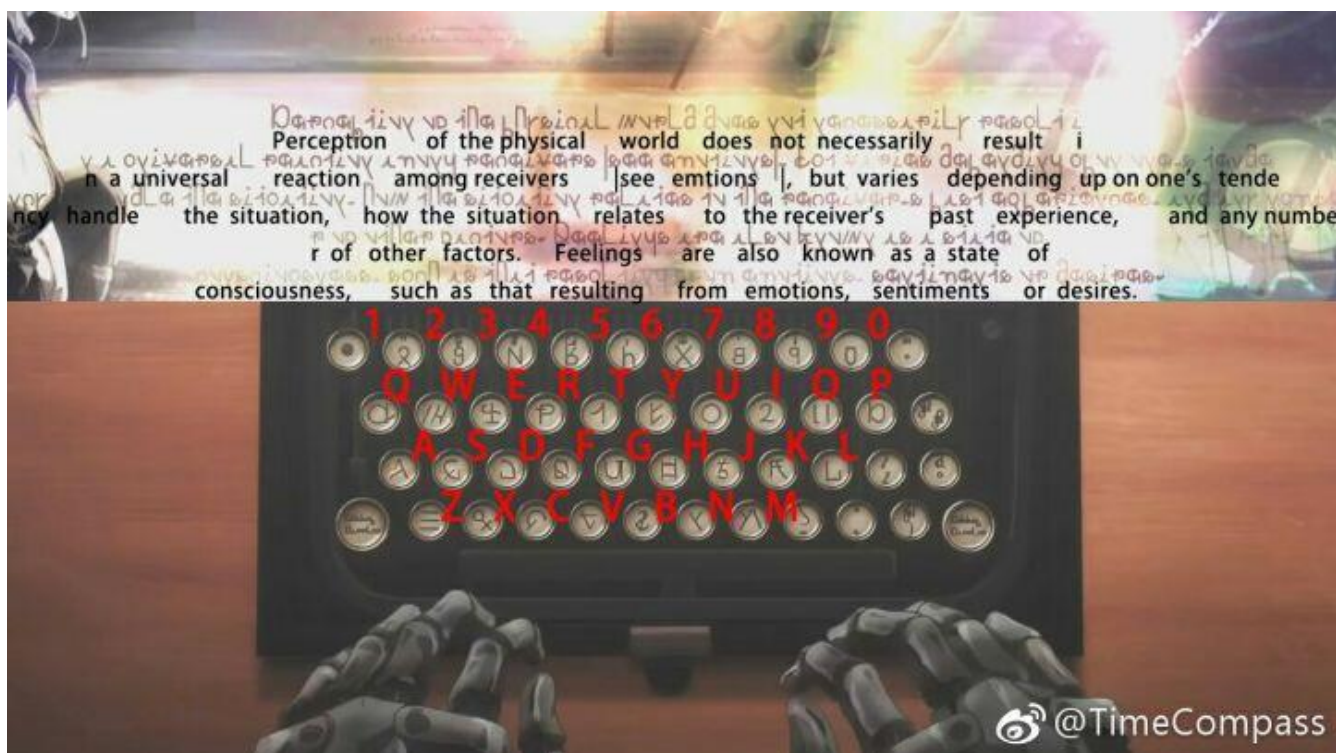
这道题后面加了hint，就简单了，下了stegsolve，稍微查了下怎么用，把压缩包解压，图片放进去，就可以得到



flag hgame{LSB_is_easy_for_u}

2.打字机

这道题有了hint之后，把图片放到谷歌的以图搜图，知道了是一部番里出现的，然后找到了这张图



然后根据这张图慢慢猜flag `hgame{My_violet_tyPewRiter}`

3.Broken Chest

这道题打开了学习资料，首先看到的有文件头标识是 `50 4B 03 04`

然后用winhex打开这个压缩包

Broken-Chest.zip																	ANSI ASCII	
Offset	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15		
00000000	4F	4B	03	04	14	00	09	00	08	00	55	BB	35	4E	CE	7C	OK	U»5Nî
00000016	B3	B0	22	00	00	00	14	00	00	00	08	00	00	00	66	6C	°"	fl
00000032	61	67	2E	74	78	74	67	49	3F	48	A0	BE	53	8B	38	E4	ag.txtgI?H	¾S<8ä
00000048	5A	42	49	02	08	5D	55	A6	4A	67	B2	B3	CE	B0	6E	C1	ZBI]U!Jg°î°nÁ	
00000064	0B	85	DC	EB	4F	91	4D	BF	50	4B	07	08	CE	7C	B3	B0	...ÜëO`M¿PK î °	
00000080	22	00	00	00	14	00	00	00	50	4B	01	02	1F	00	14	00	"	PK
00000096	09	00	08	00	55	BB	35	4E	CE	7C	B3	B0	22	00	00	00	U»5Nî °"	
00000112	14	00	00	00	08	00	24	00	00	00	00	00	00	00	20	00	\$	
00000128	00	00	00	00	00	00	66	6C	61	67	2E	74	78	74	0A	00	flag.txt	
00000144	20	00	00	00	00	00	01	00	18	00	3E	2C	76	B6	9D	B1	>,v¶ ±	
00000160	D4	01	3E	2C	76	B6	9D	B1	D4	01	1D	F1	7E	C5	9C	B1	ô >,v¶ ±ô ñ~Åæ±	
00000176	D4	01	50	4B	05	06	00	00	00	00	01	00	01	00	5A	00	ô PK	Z
00000192	00	00	58	00	00	00	10	00	53	30	6D	45	54	68	31	6E	X	S0mETHln
00000208	67	5F	55	35	65	66	75	4C									g_U5efuL	

发现头文件不对，所以修复成50

然后打开压缩包，点开flag.txt,竟然要密码???

然后用了神器ARCHPR破解密码，但没有出来。

然后重新看了一下压缩包里的内容



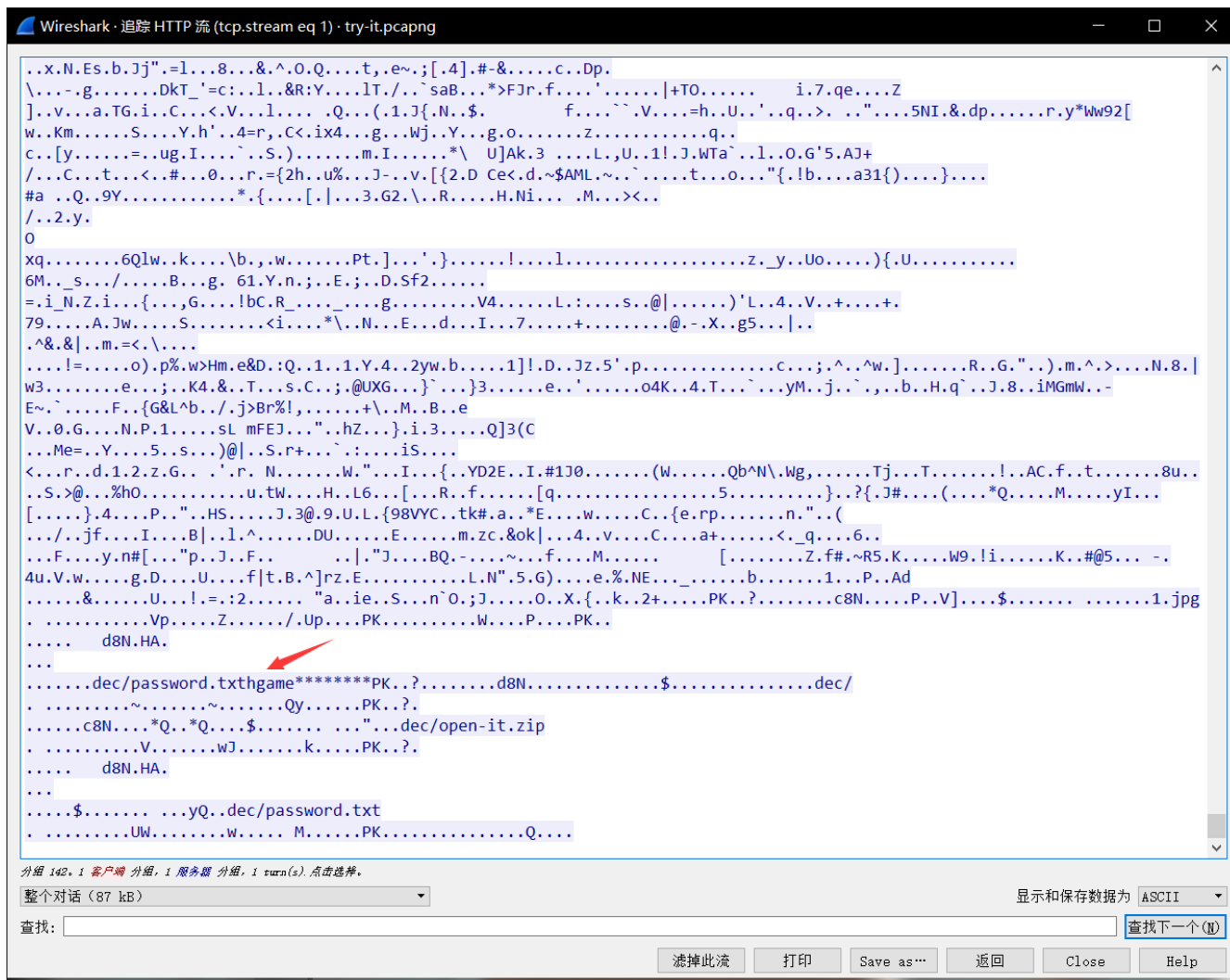
发现 这个在这里不知道什么用，就当做密码试试吧，emmm然后对了

得到flag `hgame{Cra2y_D1aM0nd}`

4.无字天书

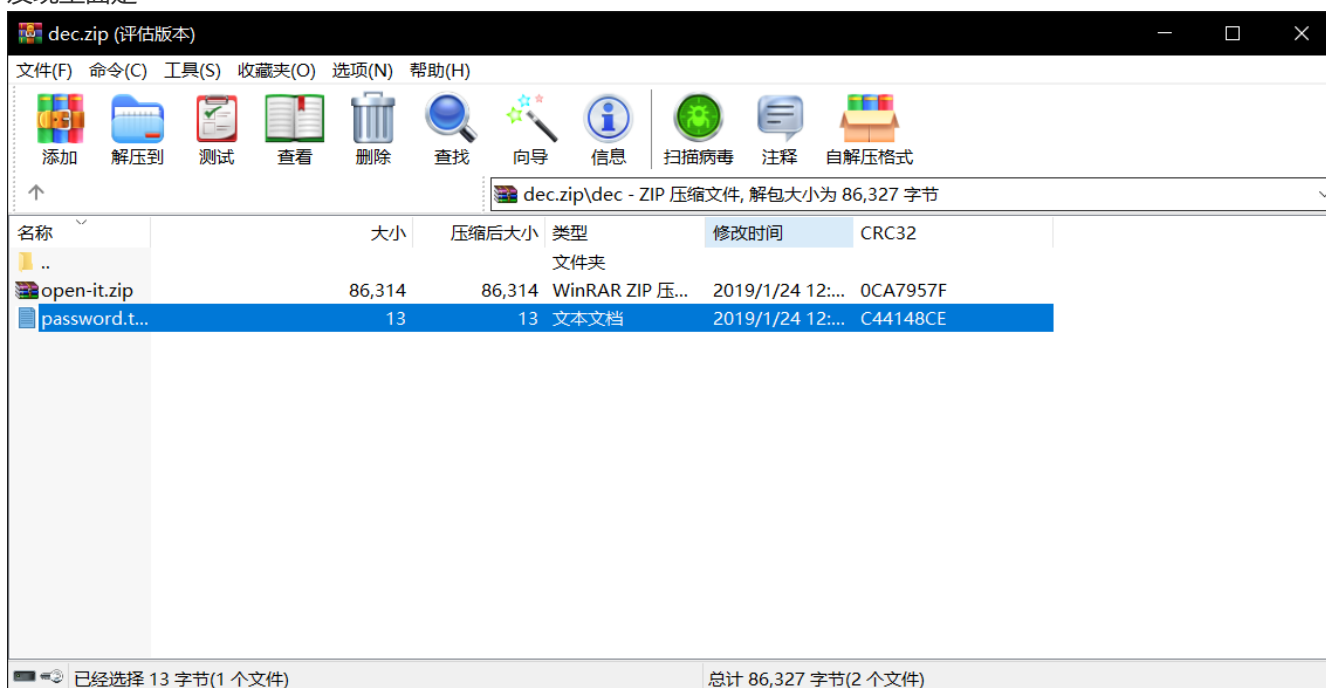
这道题到网上百度了下pcapng，出现的是流量分析题，那么打开wireshark，显示过滤器写上http

然后追踪流—http流，然后慢慢看，发现了这个



那么导出吧，导出了一个dec.zip

发现里面是



这个open-it.zip里面的图有密码，password里面是hgame加8个*

那么用掩码试试?? 先用数字吧



然后打开图片，是一个小姐姐

那么解压，用binwalk扫一下

```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows [版本 10.0.17134.523]
(c) 2018 Microsoft Corporation。保留所有权利。

C:\Users\lenovo>cd C:\Python\Scripts
C:\Python\Scripts>python binwalk 1.jpg

DECIMAL      HEXADECIMAL  DESCRIPTION
-----
0            0x0          JPEG image data, JFIF standard 1.01
79837       0x137DD      Zip archive data, at least v2.0 to extract, compressed size: 9447, uncompressed size: 1217
8, name: 1.docx
89408       0x15D40      End of Zip archive

C:\Python\Scripts>_
```

发现是docx文件，然后解压

```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows [版本 10.0.17134.523]
(c) 2018 Microsoft Corporation。保留所有权利。

C:\Users\lenovo>cd C:\Python\Scripts

C:\Python\Scripts>python binwalk 1.jpg

DECIMAL      HEXADECIMAL    DESCRIPTION
-----
0             0x0            JPEG image data, JFIF standard 1.01
79837        0x137DD        Zip archive data, at least v2.0 to extract, compressed size: 9447, uncompressed size: 12178, name: 1.docx
89408        0x15D40        End of Zip archive

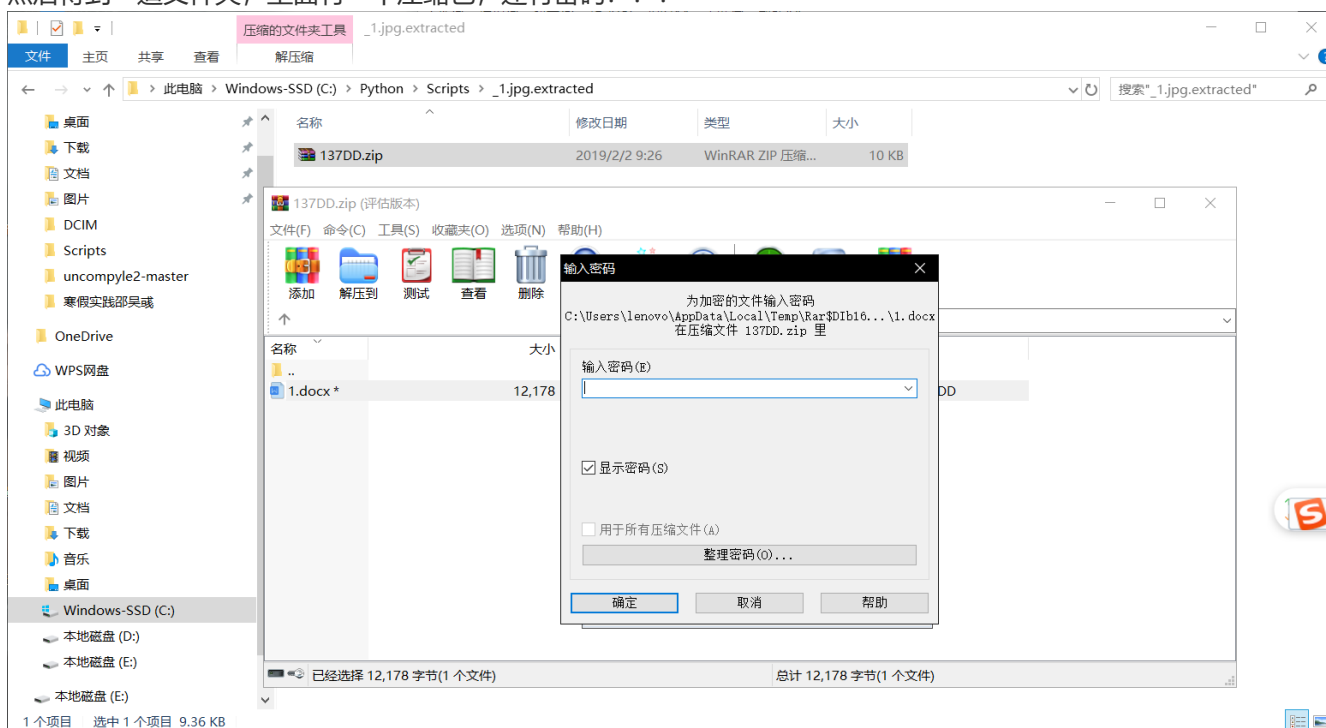
C:\Python\Scripts>python binwalk -e 1.jpg

DECIMAL      HEXADECIMAL    DESCRIPTION
-----
0             0x0            JPEG image data, JFIF standard 1.01
79837        0x137DD        Zip archive data, at least v2.0 to extract, compressed size: 9447, uncompressed size: 12178, name: 1.docx
89408        0x15D40        End of Zip archive

WARNING: Extractor.execute failed to run external extractor '7z x -y '%e' -p ''': [WinError 2] 系统找不到指定的文件。

C:\Python\Scripts>_
```

然后得到一道文件夹，里面有一个压缩包，还有密码？？？



先试试修复，修复成功后打开1.docx啥都没有，真正的无字天书

然后百度发现了这个[文件docx](#) 发现 docx文件也可以改后缀zip，然后里面会有很多xml文件，那么试试，

然后一个个翻过去，发现了

```
C:\Users\lenovo\AppData\Local\Temp\lra$Dla1076.3385\document.xml
<w:body>
  <w:p w:rsidRDefault="00A54AE2" w:rsidRPr="004066DE" w:rsidR="005C0554" w:14:textId="5B397D1E" w:14:paraId="21366764">
    <w:pPr>
      <w:rPr>
        <w:vanish/>
      </w:rPr>
    </w:pPr>
    <w:bookmarkStart w:name="_GoBack" w:id="0"/>
    <w:r w:rsidRPr="004066DE">
      <w:rPr>
        <w:vanish/>
      </w:rPr>
      <w:t>hgame</w:t>
    </w:r>
    <w:r w:rsidRPr="004066DE">
      <w:rPr>
        <w:Fonts w:hint="eastAsia"/>
        <w:vanish/>
      </w:rPr>
      <w:t>{</w:t>
    </w:r>
    <w:r w:rsidRPr="004066DE">
      <w:rPr>
        <w:vanish/>
      </w:rPr>
      <w:t>59d28413e36019861498e823f3f41406</w:t>
    </w:r>
    <w:r w:rsidRPr="004066DE">
      <w:rPr>
        <w:Fonts w:hint="eastAsia"/>
        <w:vanish/>
      </w:rPr>
      <w:t>}</w:t>
    </w:r>
    <w:bookmarkEnd w:id="0"/>
  </w:p>
  <w:sectPr w:rsidRPr="004066DE" w:rsidR="005C0554">
    <w:pgSz w:w="11906" w:h="16838"/>
    <w:pgMar w:gutter="0" w:footer="992" w:header="851" w:left="1800" w:bottom="1440" w:right="1800" w:top="1440"/>
    <w:cols w:space="425"/>
    <w:docGrid w:linePitch="312" w:type="lines"/>
  </w:sectPr>
</w:body>
</w:document>
```

得到flag hgame{59d28413e36019861498e823f3f41406}

CRYPTO

1.Mix

看到题目这一大串...想到摩斯密码

手动翻译得到744B735F6D6F7944716B7B62516634306570，然后先试了base16，然后看了去年的week1，发现有题是用了栅栏密码和凯撒密码的，那么试试

栅栏密码

tKs moyDqk{bQf40ep

输入每栏的字符数(100内的整数且必须是字符总数的因数)

加密↓

暴力解密↓

2字一栏: tsmYq{Q4eK_oDkbf0p
3字一栏: t_ykQ0KmD{fesoqb4p
6字一栏: tyQKDfsq4_k0m{eobp
9字一栏: tkK{sb_Qmfo4y0Deqp

简介: 请百度“栅栏密码”, 了解详情。(注意提示)

返回

凯撒加密与解密

加密

这里输入需要加密的英文

输入偏移量(整数)

加密

这里是加密后的密码

解密

tsmva {Q4eK_oDkbf0p

解密 使用英文字典智能分析

第2次解密:srlxp{p4dj_ncjae0o
第3次解密:rqkwo{o4ci_mbizd0n
第4次解密:qpjvn{n4bh_lahyc0m
第5次解密:poiun{m4ag_kzgx0l
第6次解密:orhtl{l4zf_jyfw0k
第7次解密:nmgsk{k4ye_ixevz0j
第8次解密:mlfrj{j4xd_hwduy0i
第9次解密:lkeqi{i4wc_gvctx0h
第10次解密:kjdph{h4vb_fubsw0g
第11次解密:jicog{g4ua_etarv0f
第12次解密:ihbnf{f4tz_dszqu0e
第13次解密:hgame{e4sy_crypt0d
第14次解密:gfzld{d4rx_bqxos0c
第15次解密:feykc{c4qw_apwnr0b
第16次解密:edxjb{b4pv_zovmq0a
第17次解密:dcwia{a4ou_ynulp0z
第18次解密:cbvhz{z4nt_xmtko0y
第19次解密:baugy{y4ms_wlsjn0x
第20次解密:aztfx{x4lr_vkrim0w

然后大小写转化下，得到flag hgame{E4sY_cRypt0d}

2.Base全家

这道题在线解码跟我说太多了，那就自己写吧

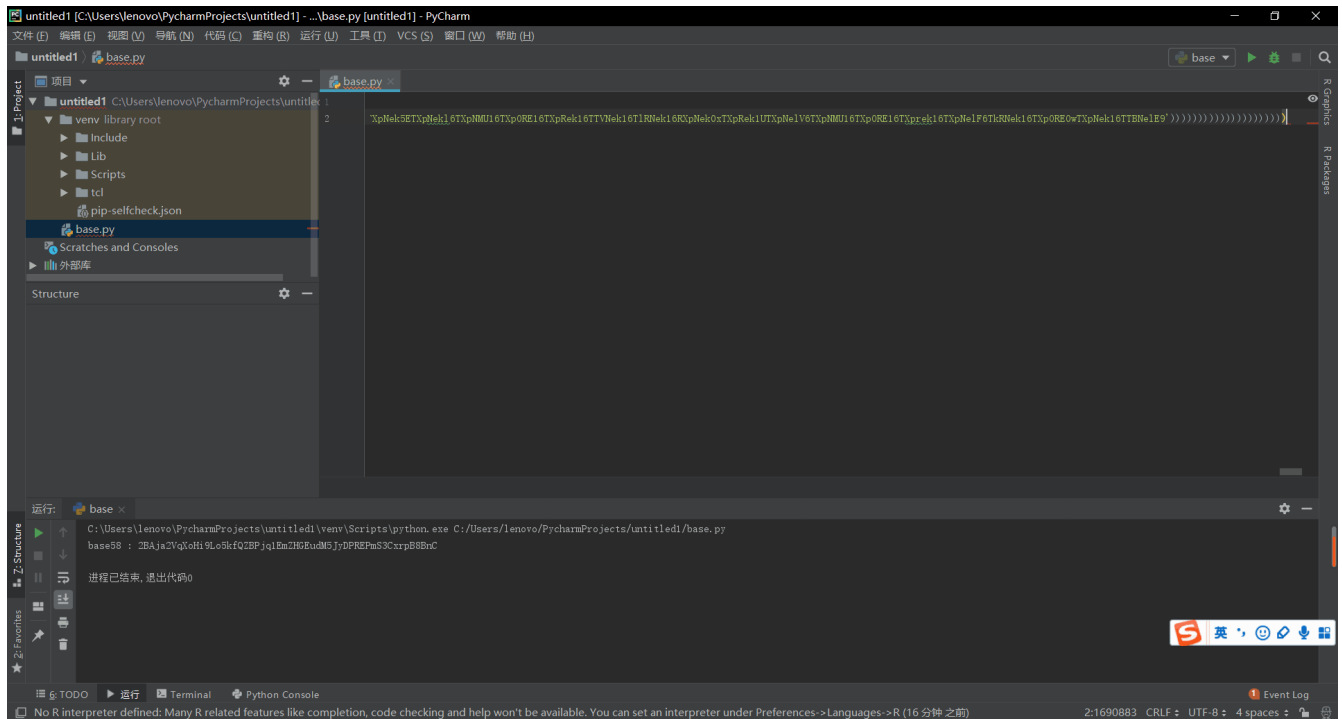
去百度了下Python脚本

```
import base64

print (base64.b64decode(''))
```

差不多就是这样，然后base64 base32 base16疯狂试

最后会出现



一段base58代码，找到一个base58在线解码

Base58 Encode, Decode, and Validate

<div><div>Bitcoin Address Validator</div><div>Input:</div><div><input type="text"/></div><div>Check if address is valid.</div><div><input type="text"/></div></div>
<div><div>Bitcoin Address Base58 Decoder</div><div>Input:</div><div><input type="text" value="2BAja2VqXoHi9Lo5kfQZBFjq1EmZHGEudM5JyDPREpms3Cxrpb8BnC"/></div><div>Decode address to hex.</div><div><input type="text" value="6867616d657b343063613738636465313434353864613639373036366562346"/></div></div>
<div><div>Bitcoin Address Base58 Encoder</div><div>Input:</div><div><input type="text"/></div><div>Encode address from hex.</div><div><input type="text"/></div></div>

The code for these procedures can be found here: [Base58 Validator/Decoder/Encoder in Perl](#).

看着像base16，base16解码，得到flag `hgame{40ca78cde14458da697066eb4cc7daf6}`

