# HGAME 2023 Week1 writeup by 1dn

## Misc

### 1.Sign In

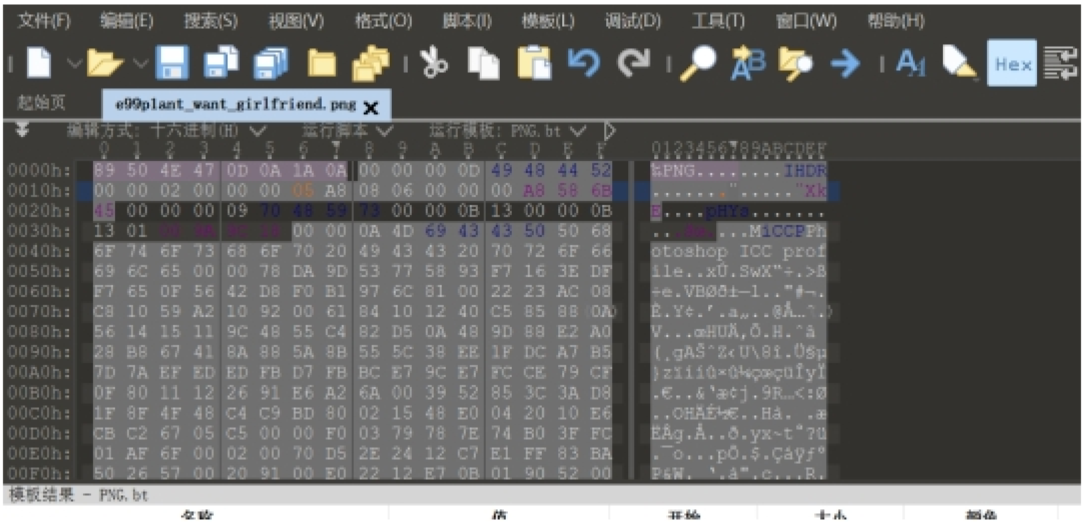
aGdhbWV7V2VsY29tZV9Ub19IR0FNRTIwMjMhfQ==

显然是base64，解密得到flag:

hgame{Welcome_To_HGAME2023!}

### 2.e99p1ant_want_girlfriend

下载的图片拖010，crc报错，改高度



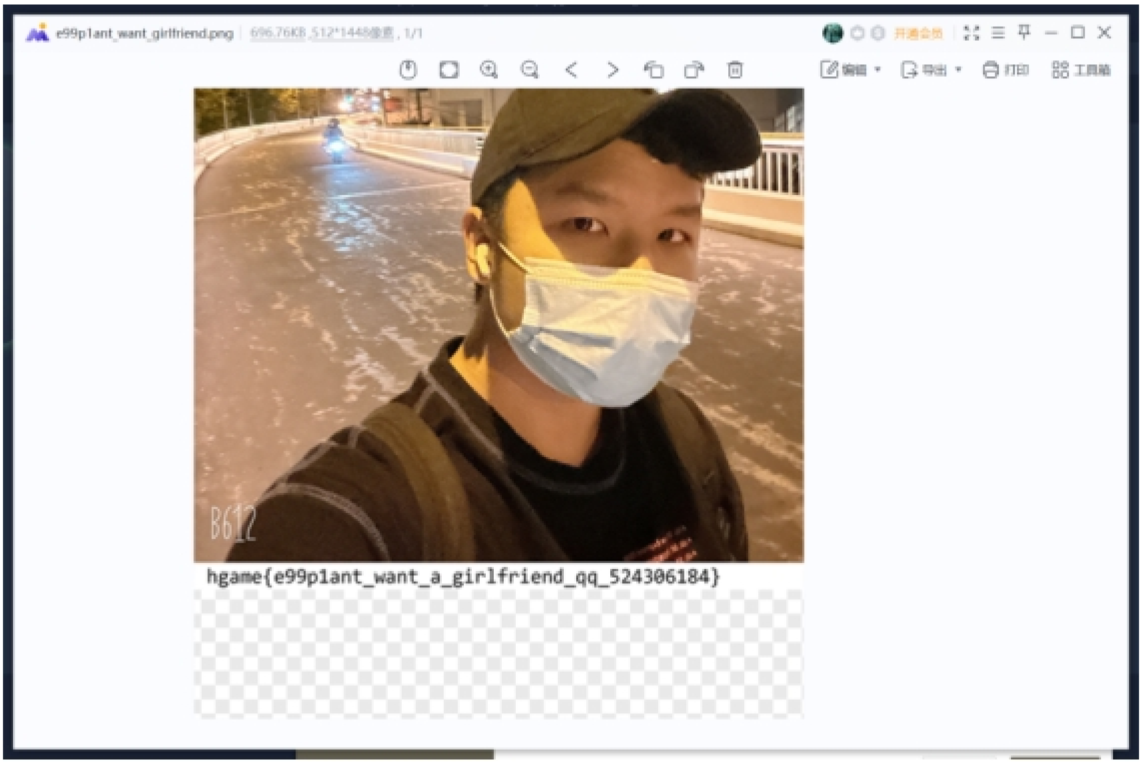
文件(F) 编辑(E) 搜索(S) 视图(V) 格式(O) 脚本(I) 模板(L) 调试(D) 工具(T) 窗口(W) 帮助(H)
Hex
起始页
e99plant_want_girlfriend.png
编辑方式: 十六进制(H)
运行脚本
运行模板: PNG.bt

```
        0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F   0123456789ABCDEF
0000h: 89 50 4E 47 0D 0A 1A 0A 00 00 00 0D 49 48 44 52  ‰PNG........IHDR
0010h: 00 00 02 00 00 00 05 A8 08 06 00 00 00 A8 58 6B  ......."......"Xk
0020h: 45 00 00 00 09 70 48 59 73 00 00 0B 13 00 00 0B  E....pHYs.......
0030h: 13 01 00 9A 9C 18 00 00 0A 4D 69 43 43 50 50 68  ...šœ....MiCCPPh
0040h: 6F 74 6F 73 68 6F 70 20 49 43 43 20 70 72 6F 66  otoshop ICC prof
0050h: 69 6C 65 00 00 78 DA 9D 53 77 58 93 F7 16 3E DF  ile..xÚ.SwX"÷.>ß
0060h: F7 65 0F 56 42 D8 F0 B1 97 6C 81 00 22 23 AC 08  ÷e.VBØð±—l.."#¬.
0070h: C8 10 59 A2 10 92 00 61 84 10 12 40 C5 85 88 0A  È.Y¢.'.a„..@Å…ˆ.
0080h: 56 14 15 11 9C 48 55 C4 82 D5 0A 48 9D 88 E2 A0  V...œHUÄ,Õ.H.ˆâ 
0090h: 28 B8 67 41 8A 88 5A 8B 55 5C 38 EE 1F DC A7 B5  (¸gAŠˆZ‹U\8î.Ü§µ
00A0h: 7D 7A EF ED ED FB D7 FB BC E7 9C E7 FC CE 79 CF  }zïííû×û¼çœçüÎyÏ
00B0h: 0F 80 11 12 26 91 E6 A2 6A 00 39 52 85 3C 3A D8  .€..&'æ¢j.9R…<:Ø
00C0h: 1F 8F 4F 48 C4 C9 BD 80 02 15 48 E0 04 20 10 E6  ..OHÄÉ½€..Hà. .æ
00D0h: CB C2 67 05 C5 00 00 F0 03 79 78 7E 74 B0 3F FC  ËÂg.Å..ð.yx~t°?ü
00E0h: 01 AF 6F 00 02 00 70 D5 2E 24 12 C7 E1 FF 83 BA  .¯o...pÕ.$.Çáÿƒº
00F0h: 50 26 57 00 20 91 00 E0 22 12 E7 0B 01 90 52 00  P&W. '.à".ç...R.
```

模板结果 - PNG.bt

名称 值 开始 大小 颜色

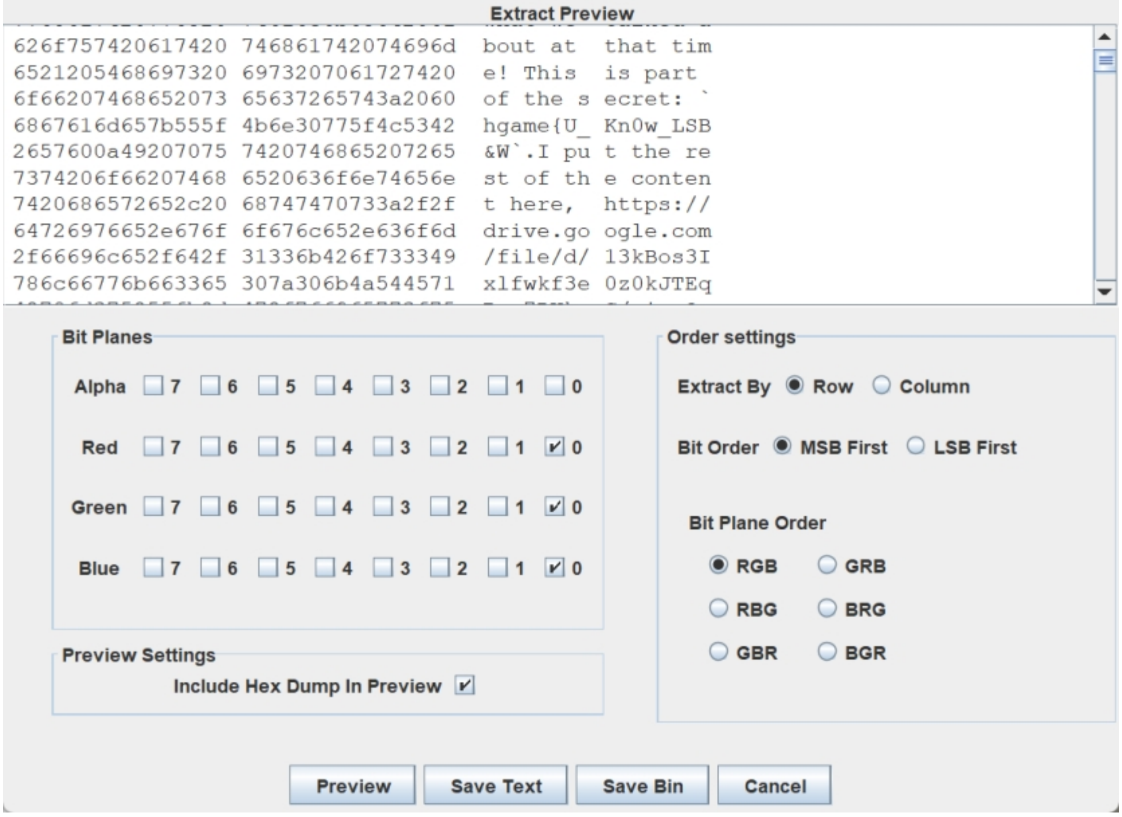
得到flag



hgame{e99p1ant_want_a_girlfriend_qq_524306184}

## 3.神秘的海报

LSB得到第一部分



Extract Preview

```
626f757420617420 746861742074696d  bout at  that tim
6521205468697320 6973207061727420  e! This  is part 
6f66207468652073 65637265743a2060  of the s ecret: `
6867616d657b555f 4b6e30775f4c5342  hgame{U_ Kn0w_LSB
2657600a49207075 742074686520726 5 &W`.I pu t the re
7374206f66207468 6520636f6e74656e  st of th e conten
742068657265 2c20 68747470733a2f2f  t here,  https://
64726976652e676f 6f676c652e636f6d  drive.go ogle.com
2f66696c652f642f 31336b426f733349  /file/d/ 13kBos3I
786c66776b663365 307a306b4a544571  xlfwkf3e 0z0kJTEq
```

Bit Planes

Alpha □ 7 □ 6 □ 5 □ 4 □ 3 □ 2 □ 1 □ 0

Red □ 7 □ 6 □ 5 □ 4 □ 3 □ 2 □ 1 ☑ 0

Green □ 7 □ 6 □ 5 □ 4 □ 3 □ 2 □ 1 ☑ 0

Blue □ 7 □ 6 □ 5 □ 4 □ 3 □ 2 □ 1 ☑ 0

Preview Settings

Include Hex Dump In Preview ☑

Order settings

Extract By ● Row ○ Column

Bit Order ● MSB First ○ LSB First

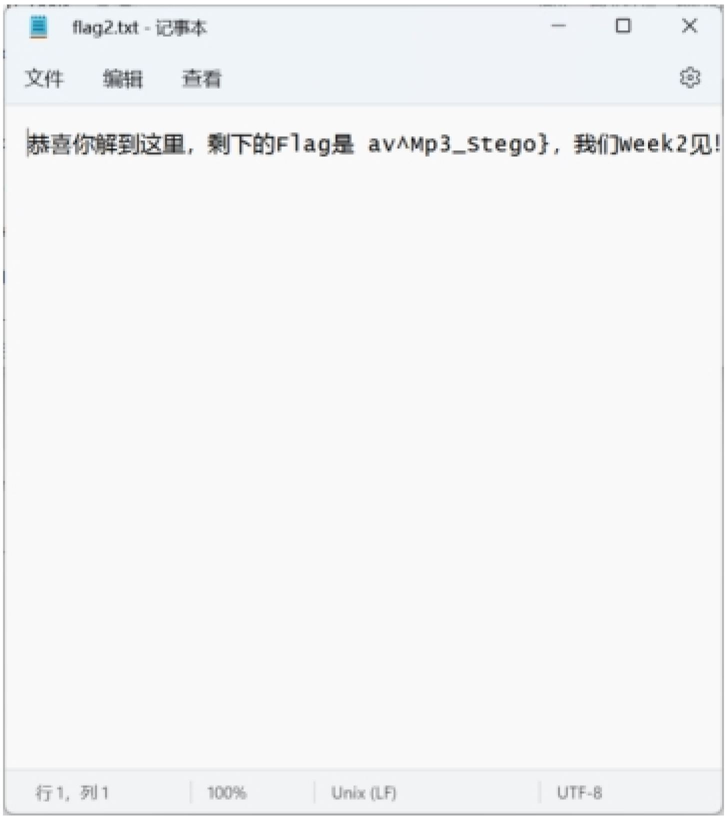
Bit Plane Order

● RGB ○ GRB

○ RBG ○ BRG

○ GBR ○ BGR

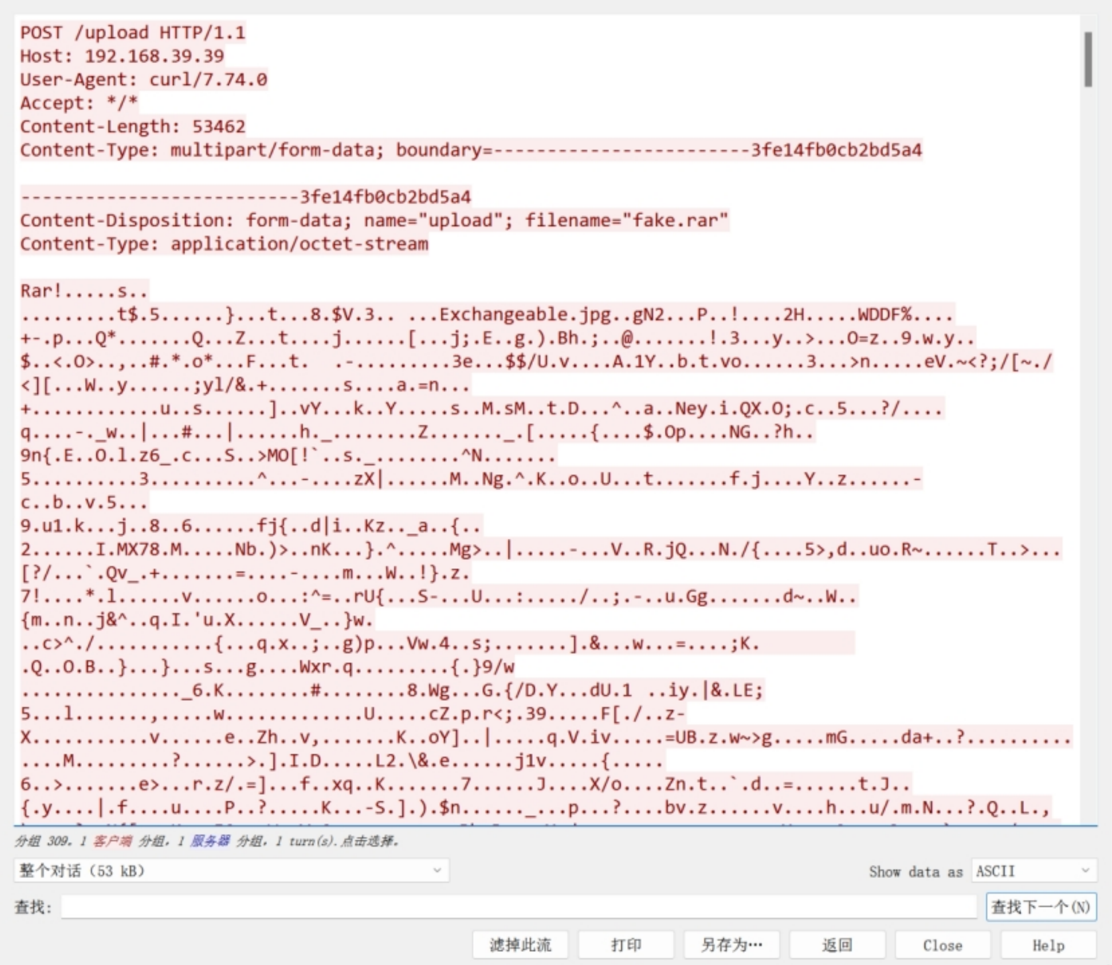
下载链接中的音频，根据提示用steghide (password is 123456)拿到另一部分



恭喜你解到这里，剩下的Flag是 av^Mp3_Stego}，我们Week2见!

## 4. Where am I

Wireshark打开附件，过滤出http协议，追踪http流，发现上传了一个rar压缩包



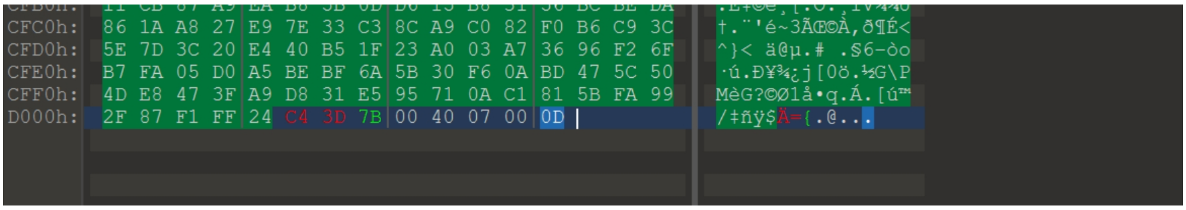
```
POST /upload HTTP/1.1
Host: 192.168.39.39
User-Agent: curl/7.74.0
Accept: */*
Content-Length: 53462
Content-Type: multipart/form-data; boundary=------------------------3fe14fb0cb2bd5a4

--------------------------3fe14fb0cb2bd5a4
Content-Disposition: form-data; name="upload"; filename="fake.rar"
Content-Type: application/octet-stream

Rar!.....s..
.........t$.5......}...t...8.$V.3.. ...Exchangeable.jpg..gN2...P..!....2H.....WDDF%....
+-.p...Q*.......Q...Z...t....j......[...j;.E..g.).Bh.;..@.......!.3...y..>...O=z..9.w.y..
$..<.O>..,..#.*.o*...F...t.  .-.........3e...$$/U.v....A.1Y..b.t.vo......3...>n.....eV.~<?;/[~./
<][...W..y......;yl/&.+.......s....a.=n...
+...........u..s......]..vY...k..Y.....s..M.sM..t.D...^..a..Ney.i.QX.O;.c..5...?/....
q....-._w..|...#...|......h._........Z......._.[.....{....$.Op....NG..?h..
9n{.E..O.l.z6_.c...S..>MO[!`..s._........^N.......
5.........3.........^...-....zX|......M..Ng.^.K..o..U...t.......f.j....Y..z......-
c..b..v.5...
9.u1.k...j..8..6......fj{..d|i..Kz.._a..{..
2......I.MX78.M.....Nb.)>..nK...}.^.....Mg>..|.....-...V..R.jQ...N./{....5>,d..uo.R~......T..>...
[?/...`.Qv_.+.......=....-....m...W..!}.z.
7!....*.l......v.......o...:^=..rU{...S-...U...:...../..;.-..u.Gg.......d~..W..
{m..n..j&^..q.I.'u.X......V_..}w.
..c>^./...........{...q.x..;..g)p...Vw.4..s;.......].&...w...=....;K.
.Q..O.B..}...}...s...g....Wxr.q.........{.}9/w
................_6.K........#........8.Wg...G.{/D.Y...dU.1  ..iy.|&.LE;
5...l.......,.....w.............U.....cZ.p.r<;.39.....F[./..z-
X..........v......e..Zh..v,.......K..oY]..|.....q.V.iv.....=UB.z.w~>g.....mG.....da+..?.........
....M.........?......>.].I.D.....L2.\&.e......j1v.....{.....
6..>.......e>...r.z/.=]...f..xq..K.......7......J....X/o....Zn.t..`.d..=......t.J..
{.y....|.f....u....P..?.....K...-S.].).$n......_...p...?....bv.z......v....h...u/.m.N...?.Q..L.,
```

分组 309。1 客户端 分组，1 服务器 分组，1 turn(s). 点击选择。

整个对话 (53 kB)

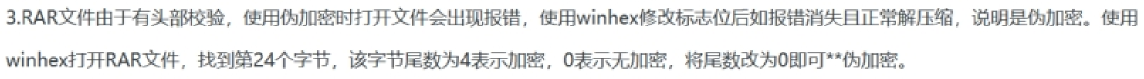
Show data as ASCII

提取出原始数据，并掐头去尾，发现打不开，会报错，拖010检查一下文件头尾，发现尾部多了0D



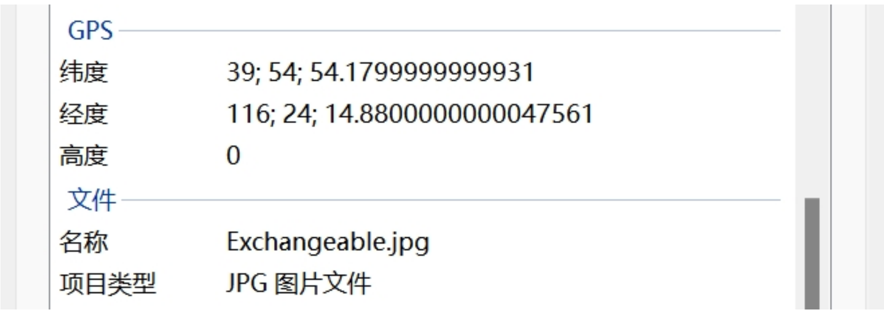
```
CFC0h:  86 1A A8 27 E9 7E 33 C3 8C A9 C0 82 F0 B6 C9 3C   †.¨'é~3ÃŒ©À‚ð¶É<
CFD0h:  5E 7D 3C 20 E4 40 B5 1F 23 A0 03 A7 36 96 F2 6F   ^}< ä@µ.# .§6–òo
CFE0h:  B7 FA 05 D0 A5 BE BF 6A 5B 30 F6 0A BD 47 5C 50   ·ú.Ð¥¾¿j[0ö.½G\P
CFF0h:  4D E8 47 3F A9 D8 31 E5 95 71 0A C1 81 5B FA 99   MèG?©Ø1å•q.Á.[ú™
D000h:  2F 87 F1 FF 24 C4 3D 7B 00 40 07 00 0D            /‡ñÿ$Ä={.@...
```

删掉，发现还是打不开，依旧报错，根据文件名叫fake,猜想是伪加密



3.RAR文件由于有头部校验，使用伪加密时打开文件会出现报错，使用winhex修改标志位后如报错消失且正常解压缩，说明是伪加密。使用winhex打开RAR文件，找到第24个字节，该字节尾数为4表示加密，0表示无加密，将尾数改为0即可**伪加密。

这状况跟伪加密也确实很像（打开会报错，第24个字节尾数为4），于是将第24个字节尾数改为0，果然打开了，不过里面只有一张黑色的jpg格式的照片，最后在属性里找到 了位置（记得保留两位小数捏）



GPS

| | |
|---|---|
| 纬度 | 39; 54; 54.1799999999931 |
| 经度 | 116; 24; 14.8800000000047561 |
| 高度 | 0 |

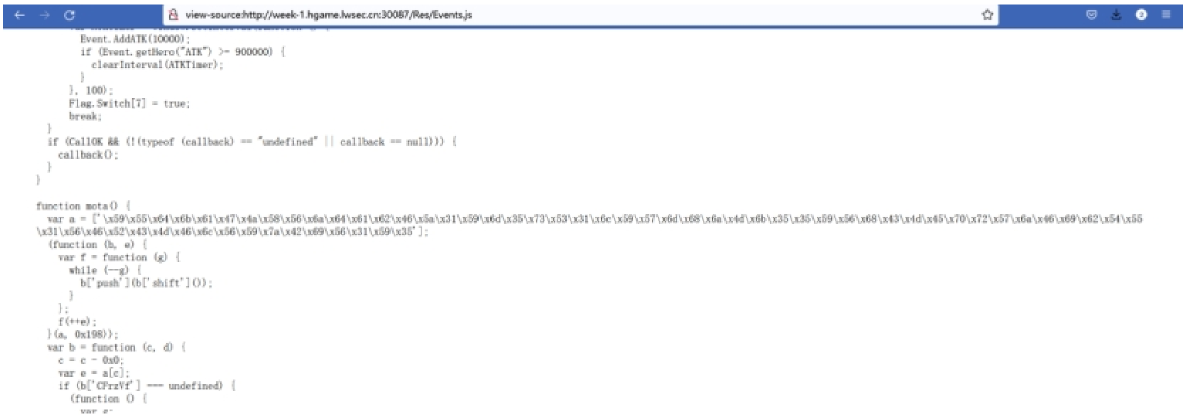
文件

| | |
|---|---|
| 名称 | Exchangeable.jpg |
| 项目类型 | JPG 图片文件 |

# Web

## 1. Classic Childhood Game

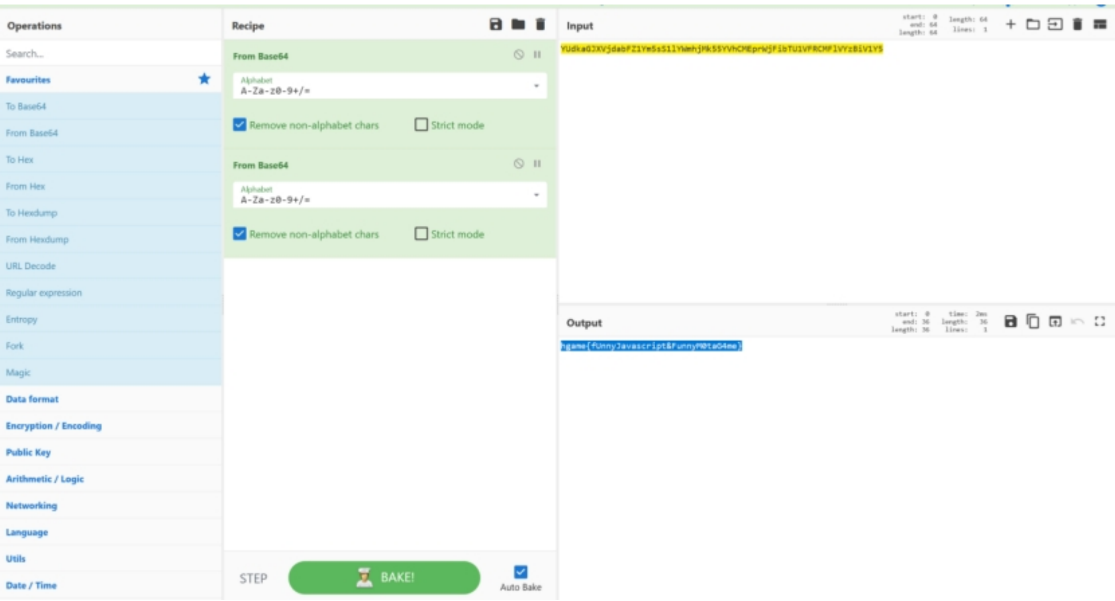
这串16进制的数有点可疑



view-source:http://week-1.hgame.lwsec.cn:30087/Res/Events.js

```
        Event.AddATK(10000);
        if (Event.getHero("ATK") >= 900000) {
          clearInterval(ATKTimer);
        }
      }, 100);
      Flag.Switch[7] = true;
      break;
  }
  if (CallOK && (!(typeof (callback) == "undefined" || callback == null))) {
    callback();
  }
}

function mota() {
  var a = ['\x59\x55\x64\x6b\x61\x47\x4a\x58\x56\x6a\x64\x61\x62\x46\x5a\x31\x59\x6d\x35\x73\x53\x31\x6c\x59\x57\x6d\x68\x6a\x4d\x6b\x35\x35\x59\x56\x68\x43\x4d\x45\x70\x72\x57\x6a\x46\x69\x62\x54\x55
\x31\x56\x46\x52\x43\x4d\x46\x6c\x56\x59\x7a\x42\x69\x56\x31\x59\x35'];
  (function (b, e) {
    var f = function (g) {
      while (--g) {
        b['push'](b['shift']());
      }
    };
    f(++e);
  }(a, 0x198));
  var b = function (c, d) {
    c = c - 0x0;
    var e = a[c];
    if (b['CPrzVf'] === undefined) {
      (function () {
```

转换一下拿去解密



From Base64
Alphabet
A-Za-z0-9+/=
☑ Remove non-alphabet chars □ Strict mode

From Base64
Alphabet
A-Za-z0-9+/=
☑ Remove non-alphabet chars □ Strict mode

Input: start: 0 end: 64 length: 64 length: 64 lines: 1

```
YUdkaGJXVjdabFZ1Ym5sS1lYWmhjMk55YVhCMEprWjFibTU1VFRCMFlVYzBiV1Y5
```

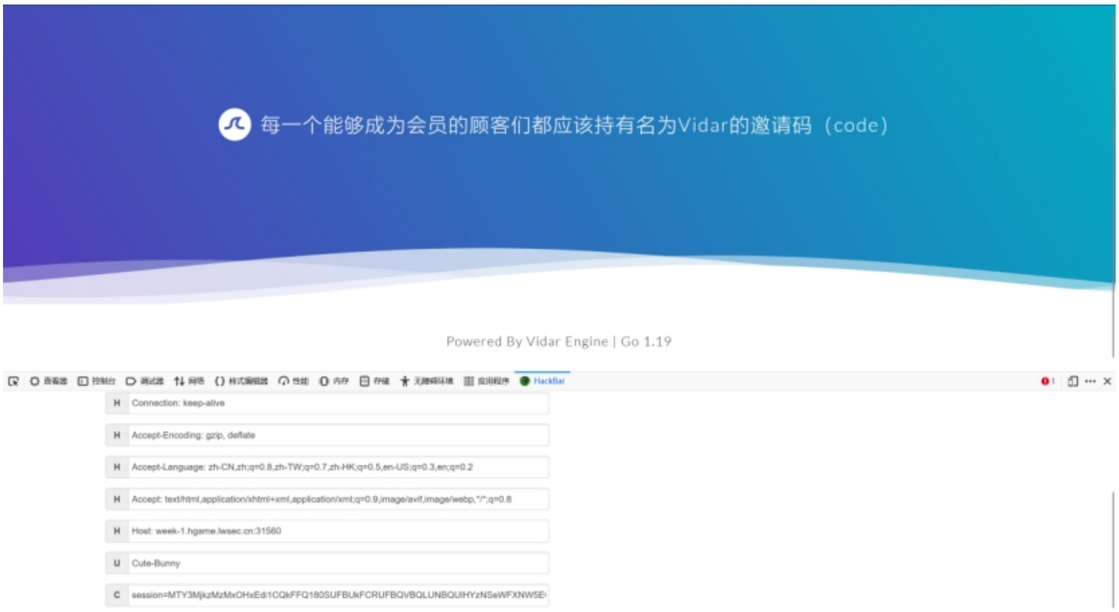
Output: start: 0 end: 36 length: 36 time: 2ms length: 36 lines: 1

```
hgame{fUnnyJavascript&FunnyM0taG4me}
```

拿到flag

## 2. Become A Member

首先把ua改成Cute-Bunny

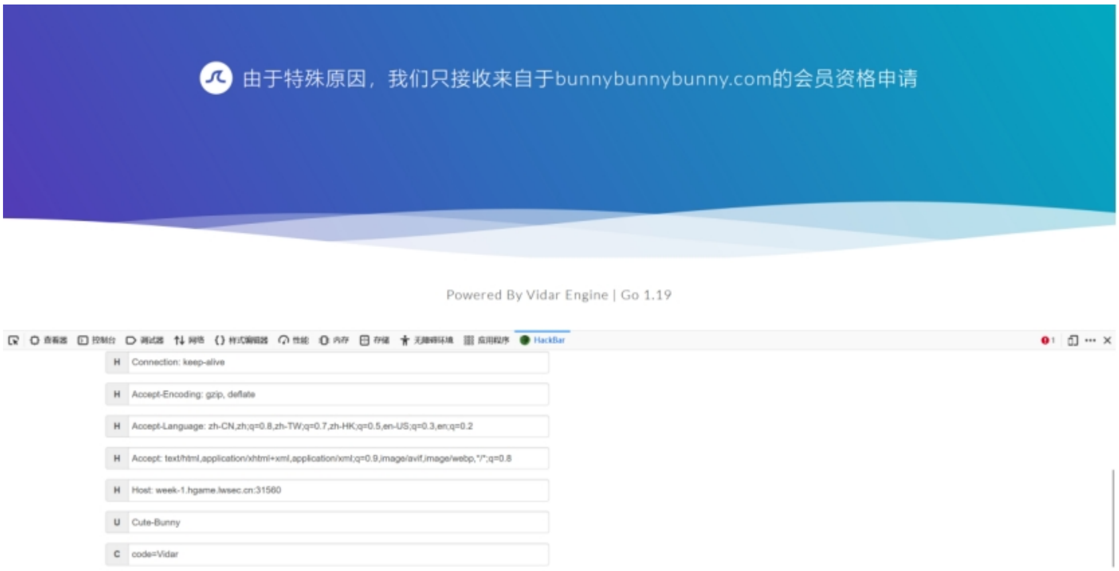


每一个能够成为会员的顾客们都应该持有名为Vidar的邀请码（code）

Powered By Vidar Engine | Go 1.19

```
H  Connection: keep-alive
H  Accept-Encoding: gzip, deflate
H  Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
H  Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
H  Host: week-1.hgame.lwsec.cn:31560
U  Cute-Bunny
C  session=MTY3MjkzMzMxOHxEd1CQkFFQ180SUFBUkFCRUFBQVBQLUNBQUlHYzNSeWFXNW5E
```

然后把cookie改成code=Vidar



由于特殊原因，我们只接收来自于bunnybunnybunny.com的会员资格申请

Powered By Vidar Engine | Go 1.19

H Connection: keep-alive

H Accept-Encoding: gzip, deflate

H Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2

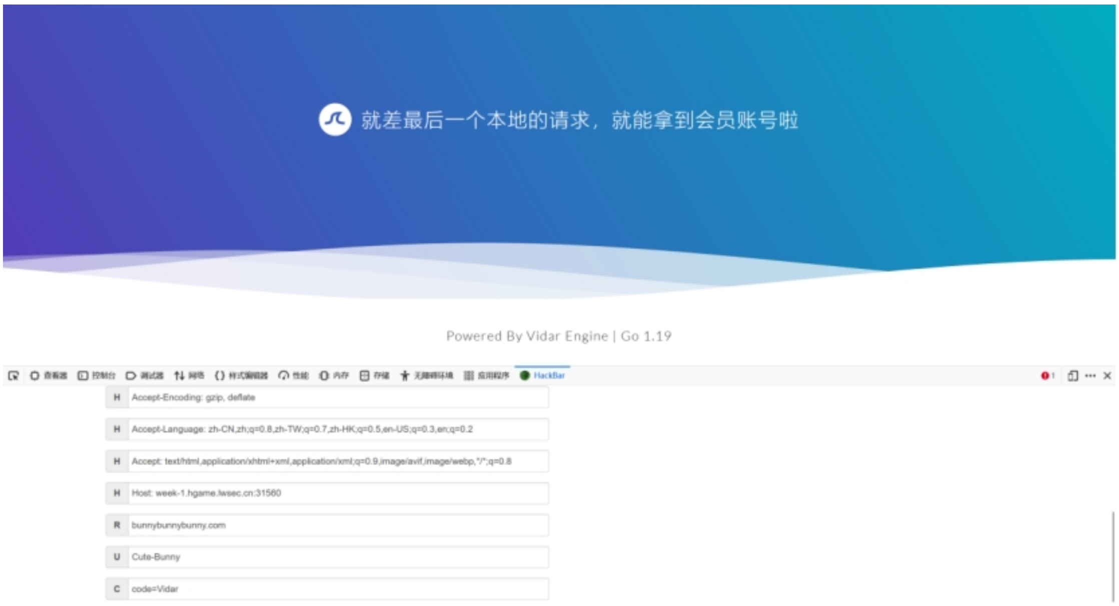
H Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8

H Host: week-1.hgame.lwsec.cn:31560

U Cute-Bunny

C code=Vidar

然后在referer输入网址



就差最后一个本地的请求，就能拿到会员账号啦

Powered By Vidar Engine | Go 1.19

H Accept-Encoding: gzip, deflate

H Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2

H Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8

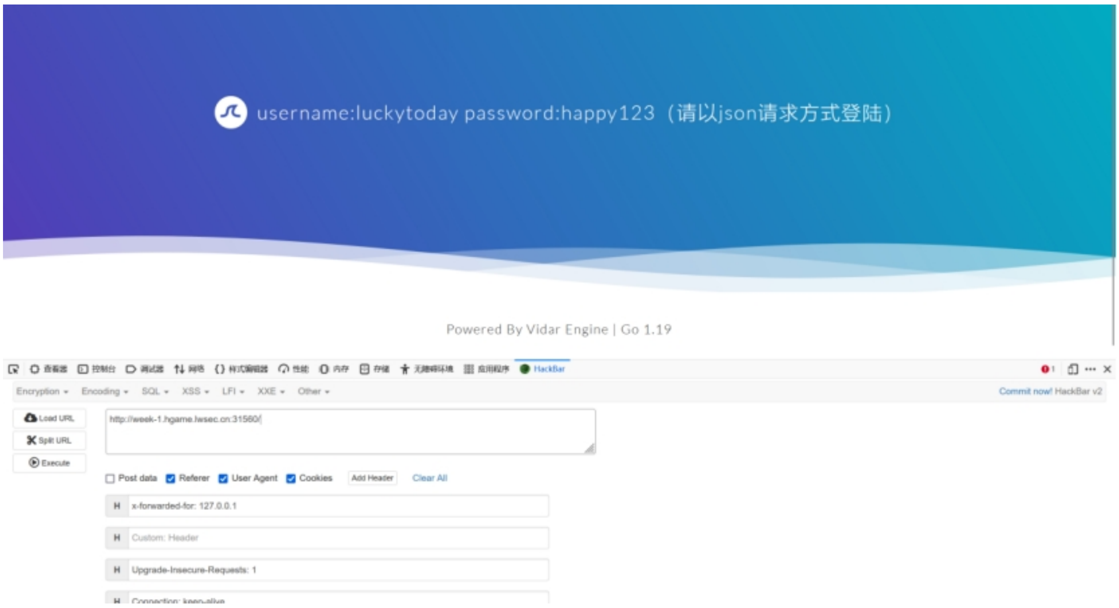
H Host: week-1.hgame.lwsec.cn:31560

R bunnybunnybunny.com

U Cute-Bunny

C code=Vidar

加一个head从本地请求，拿到账号密码



username:luckytoday password:happy123（请以json请求方式登陆）

Powered By Vidar Engine | Go 1.19

http://week-1.hgame.lwsec.cn:31560/

□ Post data ☑ Referer ☑ User Agent ☑ Cookies

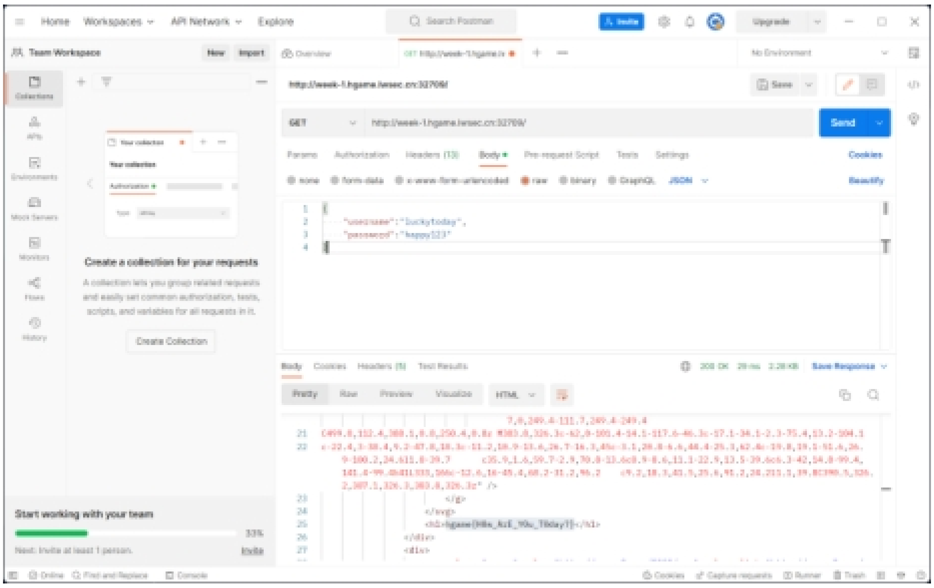
H x-forwarded-for: 127.0.0.1

H Custom: Header

H Upgrade-Insecure-Requests: 1

H Connection: keep-alive

最后一步postman做法（详情见此）

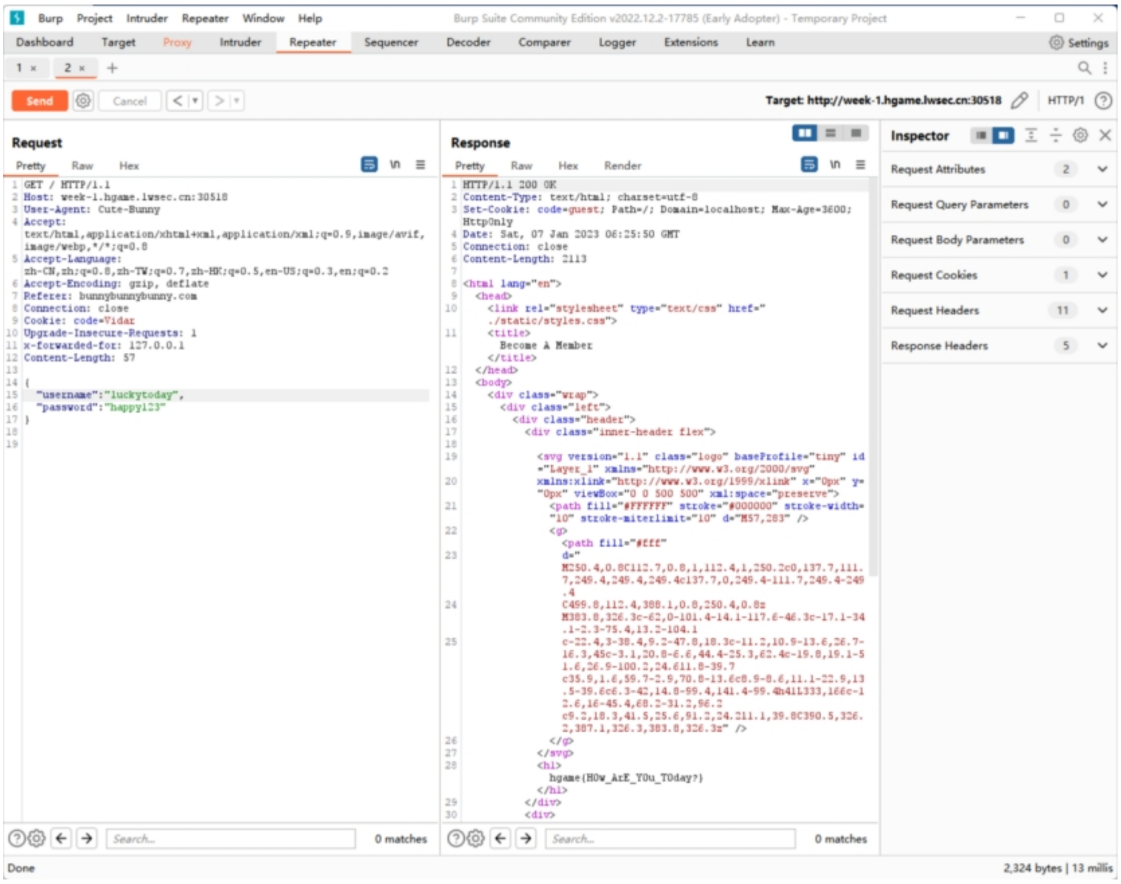


```
GET http://week-1.hgame.lwsec.cn:32709/
```

```
{
    "username":"luckytoday",
    "password":"happy123"
}
```

```
<h1>hgame{H0w_ArE_Y0u_T0day?}</h1>
```

Bp做法

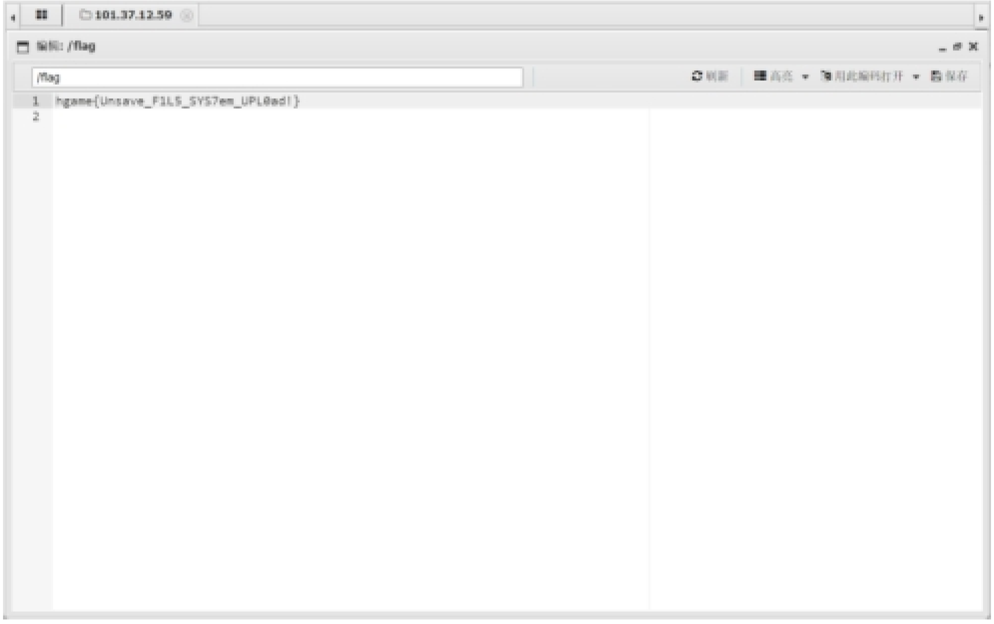

Burp Suite Community Edition v2022.12.2-17785 (Early Adopter) - Temporary Project
Burp Project Intruder Repeater Window Help
Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Logger Extensions Learn
Settings
Send
Cancel
Target: http://week-1.hgame.lwsec.cn:30518
HTTP/1
Request
Pretty Raw Hex
Response
Pretty Raw Hex Render
Inspector
Request Attributes 2
Request Query Parameters 0
Request Body Parameters 0
Request Cookies 1
Request Headers 11
Response Headers 5
Search...
0 matches
Done
2,324 bytes | 13 millis

```
GET / HTTP/1.1
Host: week-1.hgame.lwsec.cn:30518
User-Agent: Cute-Bunny
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,
image/webp,*/*;q=0.8
Accept-Language:
zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
Accept-Encoding: gzip, deflate
Referer: bunnybunnybunny.com
Connection: close
Cookie: code=Vidar
Upgrade-Insecure-Requests: 1
x-forwarded-for: 127.0.0.1
Content-Length: 57

{
  "username":"luckytoday",
  "password":"happy123"
}
```

```
HTTP/1.1 200 OK
Content-Type: text/html; charset=utf-8
Set-Cookie: code=guest; Path=/; Domain=localhost; Max-Age=3600;
HttpOnly
Date: Sat, 07 Jan 2023 06:25:50 GMT
Connection: close
Content-Length: 2113

<html lang="en">
  <head>
    <link rel="stylesheet" type="text/css" href="
    ./static/styles.css">
    <title>
      Become A Member
    </title>
  </head>
  <body>
    <div class="wrap">
      <div class="left">
        <div class="header">
          <div class="inner-header flex">

            <svg version="1.1" class="logo" baseProfile="tiny" id
            ="Layer_1" xmlns="http://www.w3.org/2000/svg"
            xmlns:xlink="http://www.w3.org/1999/xlink" x="0px" y=
            "0px" viewBox="0 0 500 500" xml:space="preserve">
              <path fill="#FFFFFF" stroke="#000000" stroke-width=
              "10" stroke-miterlimit="10" d="M57,283" />
              <g>
                <path fill="#fff"
                d="
                M250.4,0.8C112.7,0.8,1,112.4,1,250.2c0,137.7,111.
                7,249.4,249.4,249.4c137.7,0,249.4-111.7,249.4-249
                .4
                C499.8,112.4,388.1,0.8,250.4,0.8z
                M383.8,326.3c-62,0-101.4-14.1-117.6-46.3c-17.1-34
                .1-2.3-75.4,13.2-104.1
                c-22.4,3-38.4,9.2-47.8,18.3c-11.2,10.9-13.6,26.7-
                16.3,45c-3.1,20.8-6.6,44.4-25.3,62.4c-19.8,19.1-5
                1.6,26.9-100.2,24.611.8-39.7
                c35.9,1.6,59.7-2.9,70.8-13.6c8.9-8.6,11.1-22.9,13
                .5-39.6c6.3-42,14.8-99.4,141.4-99.4h41L333,166c-1
                2.6,16-45.4,68.2-31.2,96.2
                c9.2,18.3,41.5,25.6,91.2,24.211.1,39.8C390.5,326.
                2,387.1,326.3,383.8,326.3z" />
              </g>
            </svg>
            <h1>
              hgame{H0w_ArE_Y0u_T0day?}
            </h1>
          </div>
          <div>
```

## 3. Show Me Your Beauty

文件上传题，想复杂了，把所有想到的方法都试了一遍都没做出来，最后试着把含有一句话木马的文件后缀改成png提交，然后bp抓包，把提交的文件后缀改成Php发送(改成php是传不上去的)，然后用蚁剑连接拿到flag

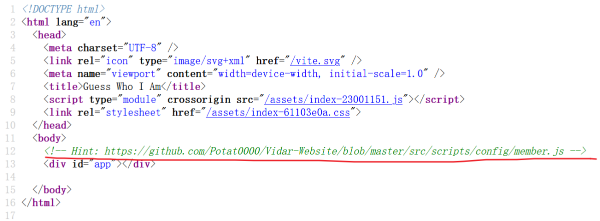


/flag

```
hgame{Unsave_F1L5_SYS7em_UPL0ad!}
```

## 4.Guess Who I Am

源码里有学长学姐的信息

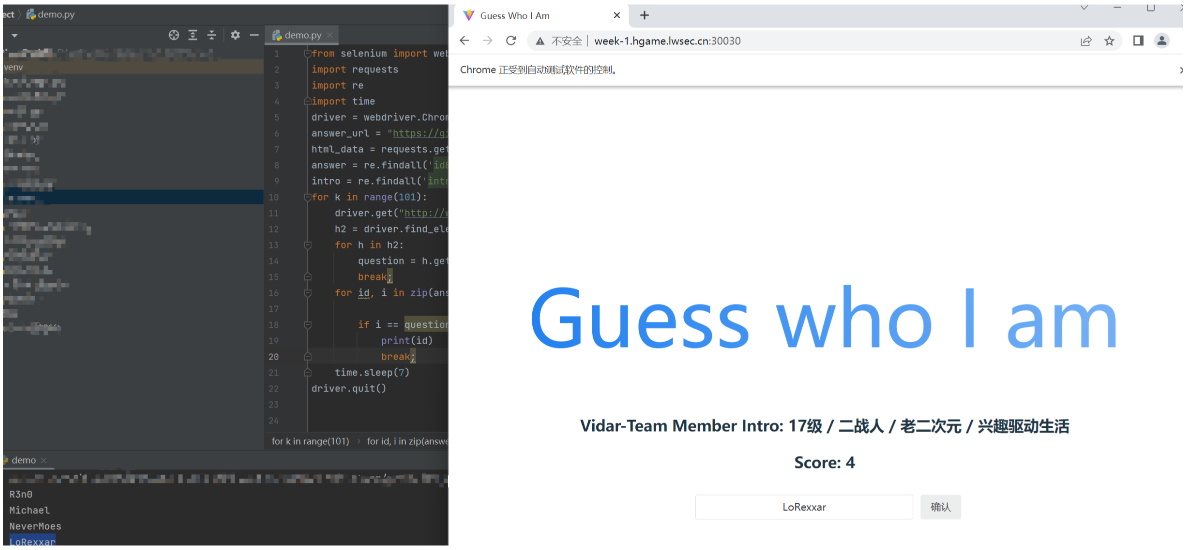


```html
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <link rel="icon" type="image/svg+xml" href="/vite.svg" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Guess Who I Am</title>
    <script type="module" crossorigin src="/assets/index-23001151.js"></script>
    <link rel="stylesheet" href="/assets/index-61103e0a.css">
  </head>
  <body>
    <!-- Hint: https://github.com/Potat0000/Vidar-Website/blob/master/src/scripts/config/member.js -->
    <div id="app"></div>

  </body>
</html>
```

然后写个脚本

```python
from selenium import webdriver
import requests
import re
import time
driver = webdriver.Chrome()
answer_url = "https://github.com/Potat0000/Vidar-Website/blob/master/src/scripts/config/member.js"
html_data = requests.get(url=answer_url).text
answer = re.findall('id&quot;</span>: <span class=pl-s>&quot;(.*?)&quot', html_data)
intro = re.findall('intro&quot;</span>: <span class=pl-s>&quot;(.*?)&quot', html_data)
for k in range(101):
    driver.get("http://week-1.hgame.lwsec.cn:30030/")
    h2 = driver.find_elements_by_css_selector('div.card h2')
    for h in h2:
        question = h.get_attribute('value')
        break;
    for id, i in zip(answer, intro):

        if i == question:
            print(id)
            break;
    time.sleep(7)
driver.quit()
```

emmm,之前没学过python，临时学的，上面的脚本并不能完全自动答题，运行以后会把答案输出在窗口，只能手动输入答案提交



[redacted]

```
from selenium import web
import requests
import re
import time
driver = webdriver.Chrom
answer_url = "https://gi
html_data = requests.get
answer = re.findall('id&
intro = re.findall('intr
for k in range(101):
    driver.get("http://w
    h2 = driver.find_ele
    for h in h2:
        question = h.get
        break;
    for id, i in zip(ans

        if i == question
            print(id)
            break;
    time.sleep(7)
driver.quit()
```

for k in range(101) 〉 for id, i in zip(answe

[redacted]

```
R3n0
Michael
NeverMoes
LoRexxar
```

不安全 | week-1.hgame.lwsec.cn:30030

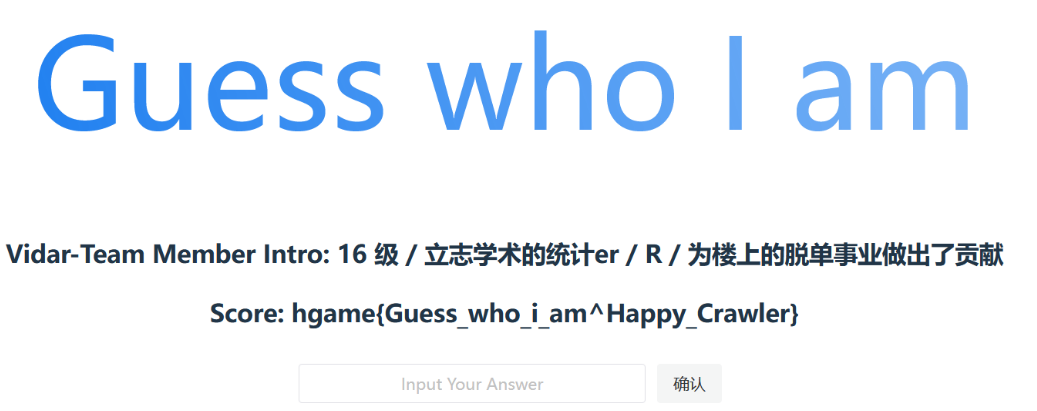
Chrome 正受到自动测试软件的控制。

Guess who I am

Vidar-Team Member Intro: 17级 / 二战人 / 老二次元 / 兴趣驱动生活

Score: 4

LoRexxar



Guess who I am

Vidar-Team Member Intro: 16 级 / 立志学术的统计er / R / 为楼上的脱单事业做出了贡献

Score: hgame{Guess_who_i_am^Happy_Crawler}

## Crypto

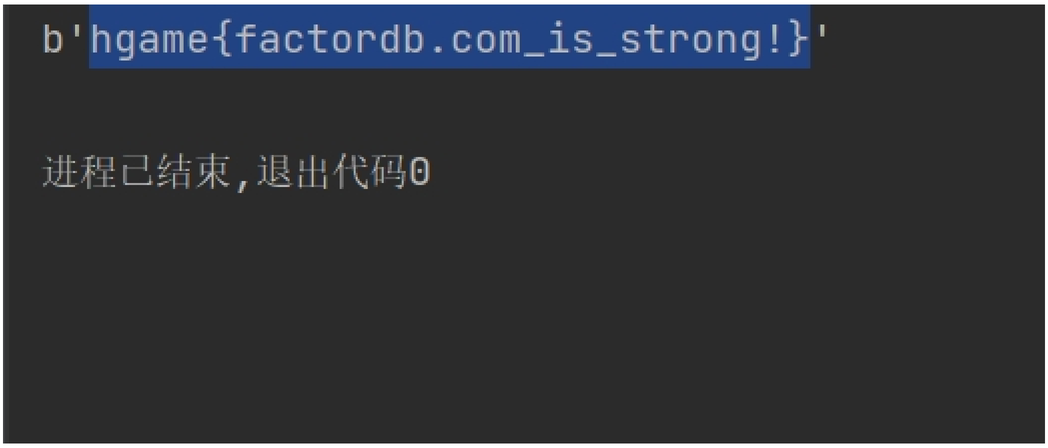
### 1.RSA

到factordb.com分离出两个素数pq，然后写脚本

```
from Crypto.Util.number import *

p=11239134987804993586763559028187245057652550219515201768644770733869088185320740938450178816138394844329723311433549899499795775655921261664087997097294813

q=12022912661420941592569751731802639375088427463430162252113082619617837010913002515450223656942836378041122163833359097910935638423464006252814266959128953
```

```
c=11067479267401774824323235118589601966043471834200168690652778987626497632868613410197212549393843499278700291556250047548069329736086768100009272558328461635354342238848920811454500713860654367804079865183602743338328217708103415158993502429201720720905682925015221918351840036487110955982567927350227495558

n=13512713834829975737419644706264085841692035009832009999311594971905135421354559664321673955545394619607811083472637547598179122306945136402418195281805680208956706492651029412459417447812321651660036833476384920694294282471153133423910680745408638921113915302366226612593748166952077187935508999767112502078

i = (p-1)*(q-1)

e = 65537

d = inverse(e,i)

print(long_to_bytes(pow(c, d, n)))
```

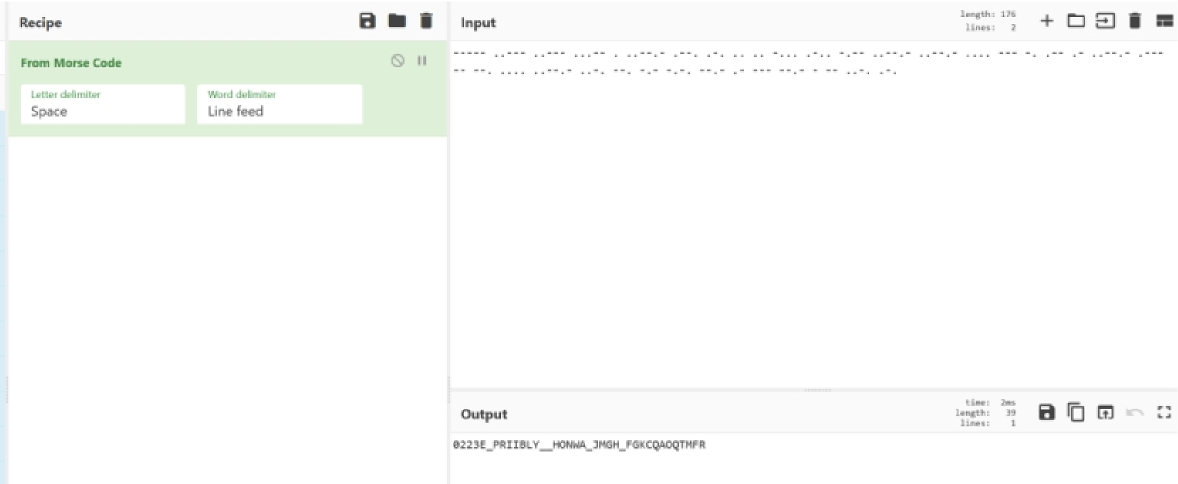
运行得到flag



```
b'hgame{factordb.com_is_strong!}'

进程已结束,退出代码0
```

## 2.神秘的电话

附件里一个音频一个文本，音频拖到Audacity拿到摩斯密码，解一下



From Morse Code

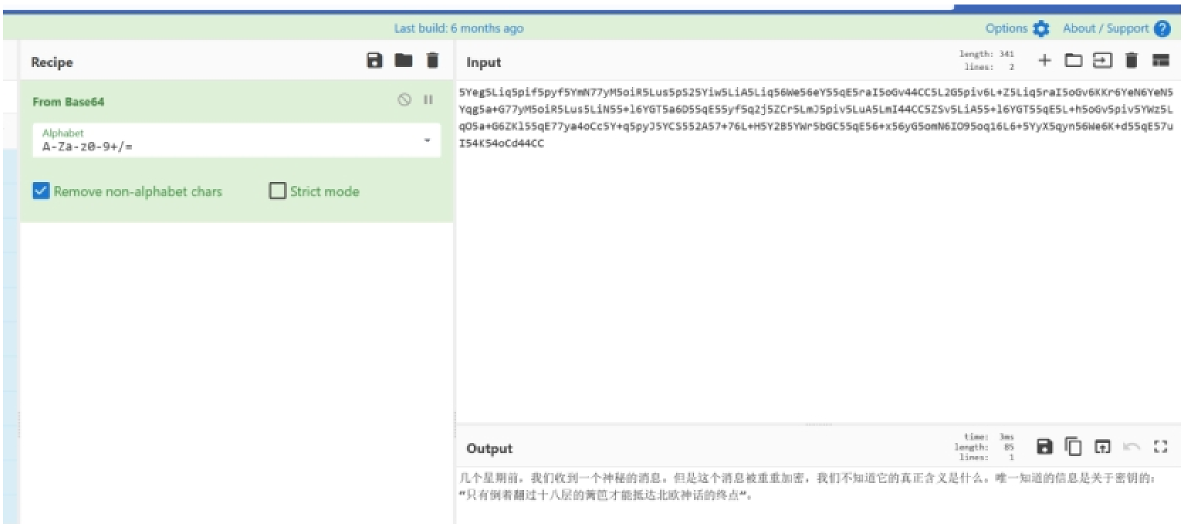
Letter delimiter: Space

Word delimiter: Line feed

```
----- ..--- ..--- ...-- . ..--.- .--. .-. .. .. -... .-.. -.-- ..--.- ..--.- .... --- -. .-- .- ..--.- .---
-- --. .... ..--.- ..-. --. -.- -.-. --.- .- --- --.- - -- ..-. .-.
```

```
0223E_PRIIBLY__HONWA_JMGH_FGKCQAOQTMFR
```

然后文本也解个密



From Base64

Alphabet
A-Za-z0-9+/=

☑ Remove non-alphabet chars　□ Strict mode

length: 341
lines: 2

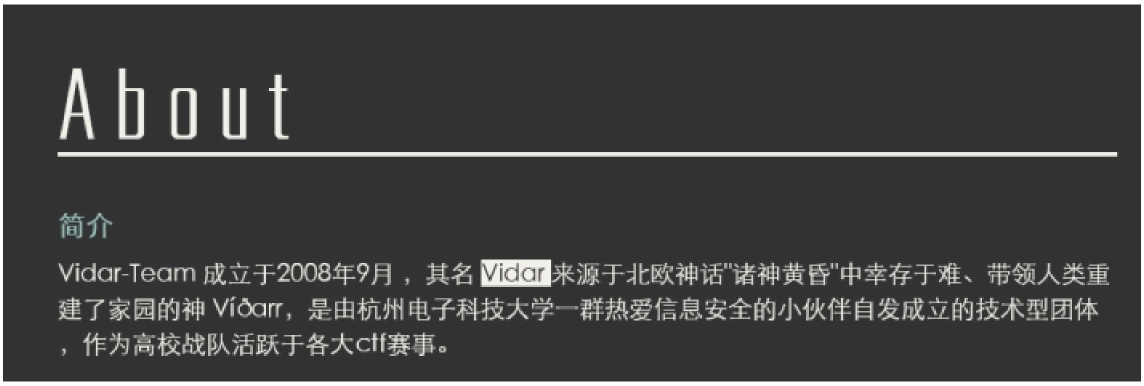
```
5Yeg5Liq5pif5pyf5YmN77yM5oiR5Lus5pS25Yiw5LiA5Liq56We56eY55qE5raI5oGv44CC5L2G5piv6L+Z5Liq5raI5oGv6KKr6YeN6YeN5
Yqg5a+G77yM5oiR5Lus5LiN55+l6YGT5a6D55qE55yf5q2j5ZCr5LmJ5piv5LuA5LmI44CC5ZSv5LiA55+l6YGT55qE5L+h5oGv5piv5YWz5L
qO5a+G6ZKl55qE77ya4oCc5Y+q5pyJ5YCS552A57+76L+H5Y2B5YWr5bGC55qE56+x56yG5omN6IO95oq16L6+5YyX5qyn56We6K+d55qE57u
I54K54oCd44CC
```

time: 3ms
length: 85
lines: 1

```
几个星期前，我们收到一个神秘的消息。但是这个消息被重重加密，我们不知道它的真正含义是什么。唯一知道的信息是关于密钥的：
“只有倒着翻过十八层的篱笆才能抵达北欧神话的终点”。
```

根据意思将前面的密文倒过来再进行key=18的栅栏密码解密，本来以为就结束了，然而并没有，还有一个古典密码，根据学长的提示，北欧神话的终点为密钥。

emmm,北欧神话的终点也就是诸神的黄昏，结果就傻傻地用诸神的黄昏的英文试了一圈古典密码，发现都不对，最后在vidar官网看到：



About

简介

Vidar-Team 成立于2008年9月，其名 Vidar 来源于北欧神话"诸神黄昏"中幸存于难、带领人类重建了家园的神 Víðarr，是由杭州电子科技大学一群热爱信息安全的小伙伴自发成立的技术型团体，作为高校战队活跃于各大ctf赛事。

然后最后试出来是维吉尼亚密码

hgame{welcome_to_hgame2023_and_enjoy_hacking}

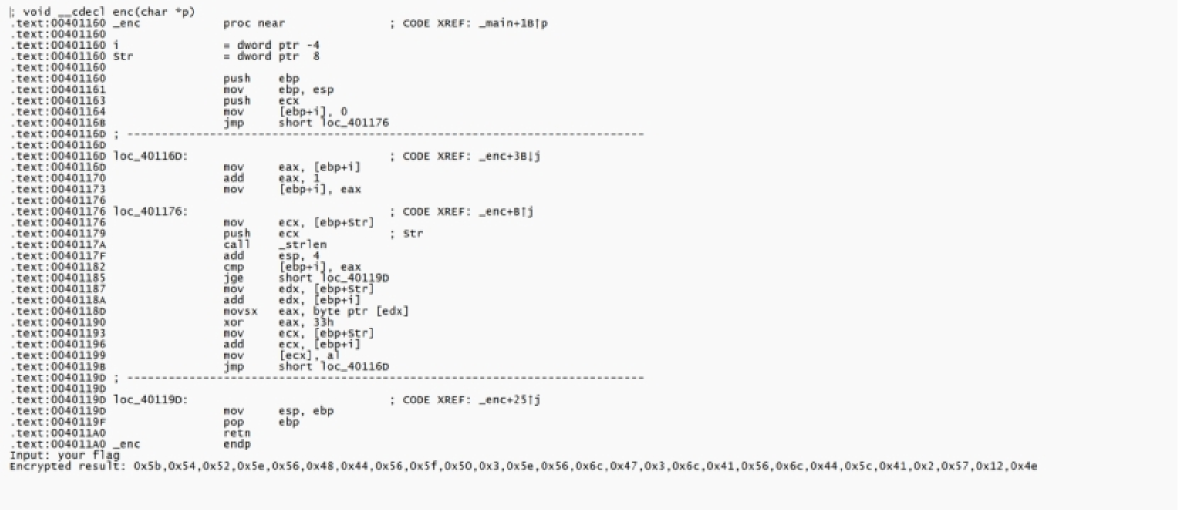
# Reverse

## 1.test your IDA

是签到题捏，用ida打开看一下就能找到flag

## 2.easyasm

这是一个txt附件

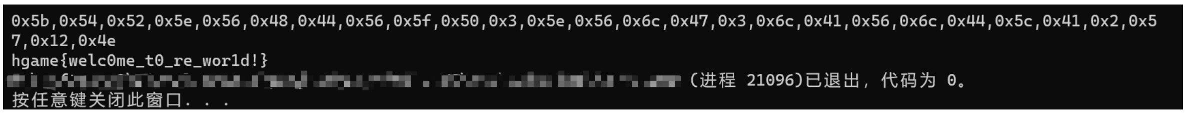


```
; void __cdecl enc(char *p)
.text:00401160 _enc            proc near               ; CODE XREF: _main+1B↑p
.text:00401160
.text:00401160 i               = dword ptr -4
.text:00401160 Str             = dword ptr  8
.text:00401160
.text:00401160                 push    ebp
.text:00401161                 mov     ebp, esp
.text:00401163                 push    ecx
.text:00401164                 mov     [ebp+i], 0
.text:0040116B                 jmp     short loc_401176
.text:0040116D ; ---------------------------------------------------------------------------
.text:0040116D
.text:0040116D loc_40116D:                             ; CODE XREF: _enc+3B↓j
.text:0040116D                 mov     eax, [ebp+i]
.text:00401170                 add     eax, 1
.text:00401173                 mov     [ebp+i], eax
.text:00401176
.text:00401176 loc_401176:                             ; CODE XREF: _enc+B↑j
.text:00401176                 mov     ecx, [ebp+Str]
.text:00401179                 push    ecx             ; Str
.text:0040117A                 call    _strlen
.text:0040117F                 add     esp, 4
.text:00401182                 cmp     [ebp+i], eax
.text:00401185                 jge     short loc_40119D
.text:00401187                 mov     edx, [ebp+Str]
.text:0040118A                 add     edx, [ebp+i]
.text:0040118D                 movsx   eax, byte ptr [edx]
.text:00401190                 xor     eax, 33h
.text:00401193                 mov     ecx, [ebp+Str]
.text:00401196                 add     ecx, [ebp+i]
.text:00401199                 mov     [ecx], al
.text:0040119B                 jmp     short loc_40116D
.text:0040119D ; ---------------------------------------------------------------------------
.text:0040119D
.text:0040119D loc_40119D:                             ; CODE XREF: _enc+25↑j
.text:0040119D                 mov     esp, ebp
.text:0040119F                 pop     ebp
.text:004011A0                 retn
.text:004011A0 _enc            endp
Input: your flag
Encrypted result: 0x5b,0x54,0x52,0x5e,0x56,0x48,0x44,0x56,0x5f,0x50,0x3,0x5e,0x56,0x6c,0x47,0x3,0x6c,0x41,0x56,0x6c,0x44,0x5c,0x41,0x2,0x57,0x12,0x4e
```

没学过汇编看不懂捏，不过看到xor推测是将原数据与33h进行了异或，得到了下面的16进制数据，只要再异或一遍即可获得原数据

将数据异或出来，然后转成字符串输出（还真是这样捏，菜鸡只能靠猜）

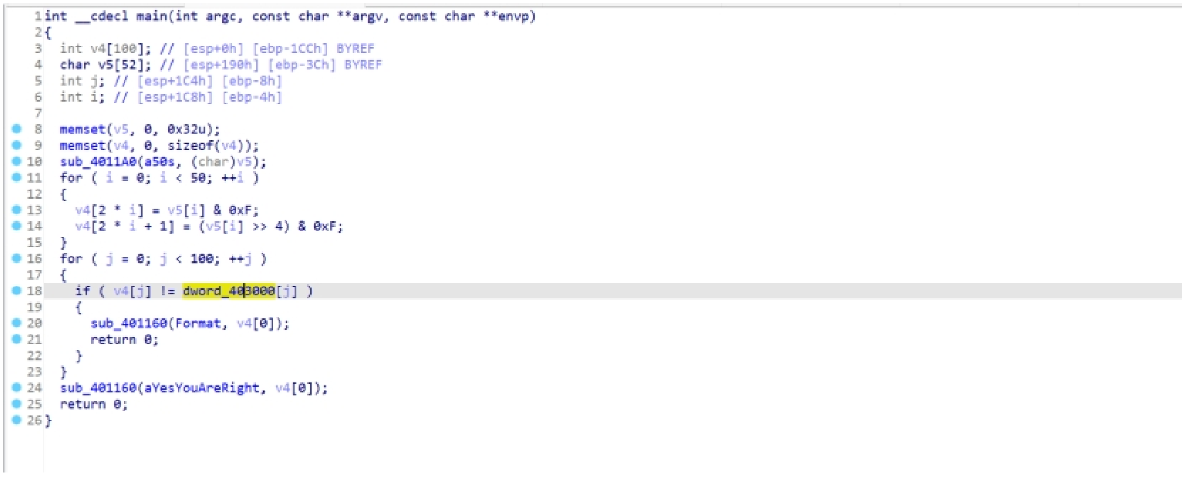
```c
#include<stdio.h>
int main() {
    int a[27], i = 0;
    for (i = 0; i < 27; i++) {
        scanf("%x,", a + i);
    }
    for (i = 0; i < 27; i++) {
        a[i] ^= 0x33;
        printf("%c", a[i]);
    }
    return 0;
}
```



```
0x5b,0x54,0x52,0x5e,0x56,0x48,0x44,0x56,0x5f,0x50,0x3,0x5e,0x56,0x6c,0x47,0x3,0x6c,0x41,0x56,0x6c,0x44,0x5c,0x41,0x2,0x5
7,0x12,0x4e
hgame{welc0me_t0_re_wor1d!}
[redacted] (进程 21096)已退出，代码为 0。
按任意键关闭此窗口. . .
```

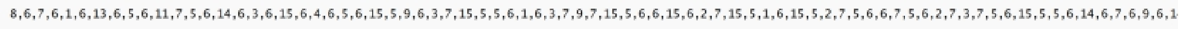
## 3.Encode

拖ida，F5反汇编



```c
int __cdecl main(int argc, const char **argv, const char **envp)
{
  int v4[100]; // [esp+0h] [ebp-1CCh] BYREF
  char v5[52]; // [esp+190h] [ebp-3Ch] BYREF
  int j; // [esp+1C4h] [ebp-8h]
  int i; // [esp+1C8h] [ebp-4h]

  memset(v5, 0, 0x32u);
  memset(v4, 0, sizeof(v4));
  sub_4011A0(a50s, (char)v5);
  for ( i = 0; i < 50; ++i )
  {
    v4[2 * i] = v5[i] & 0xF;
    v4[2 * i + 1] = (v5[i] >> 4) & 0xF;
  }
  for ( j = 0; j < 100; ++j )
  {
    if ( v4[j] != dword_403000[j] )
    {
      sub_401160(Format, v4[0]);
      return 0;
    }
  }
  sub_401160(aYesYouAreRight, v4[0]);
  return 0;
}
```

需要输入数据给V5,然后V4通过V5加密得到的数据需与dword_403000相同，查看dword_403000的数据并提取出来



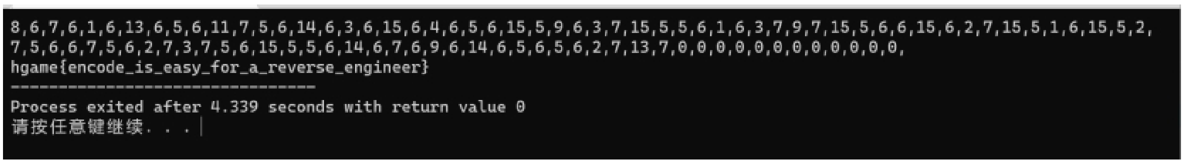
```
8,6,7,6,1,6,13,6,5,6,11,7,5,6,14,6,3,6,15,6,4,6,5,6,15,5,9,6,3,7,15,5,5,6,1,6,3,7,9,7,15,5,6,6,15,6,2,7,15,5,1,6,15,5,2,7,5,6,6,7,5,6,2,7,3,7,5,6,15,5,5,6,14,6,7,6,9,6,1
```

V4下标为奇数的存的是V5对应的那个数据的前4位，V4下标为偶数的存的是V5对应那个数据的后4位

写个脚本

```
#include<stdio.h>
int main(){
    int a[100],i=0,ch=0;

    for(i=0;i<100;i++){
        scanf("%d,",a+i);
    }
    for(i=0;i<50;i++){
        ch=a[2*i+1]*16+a[2*i];
        printf("%c",ch);
    }

    return 0;
}
```

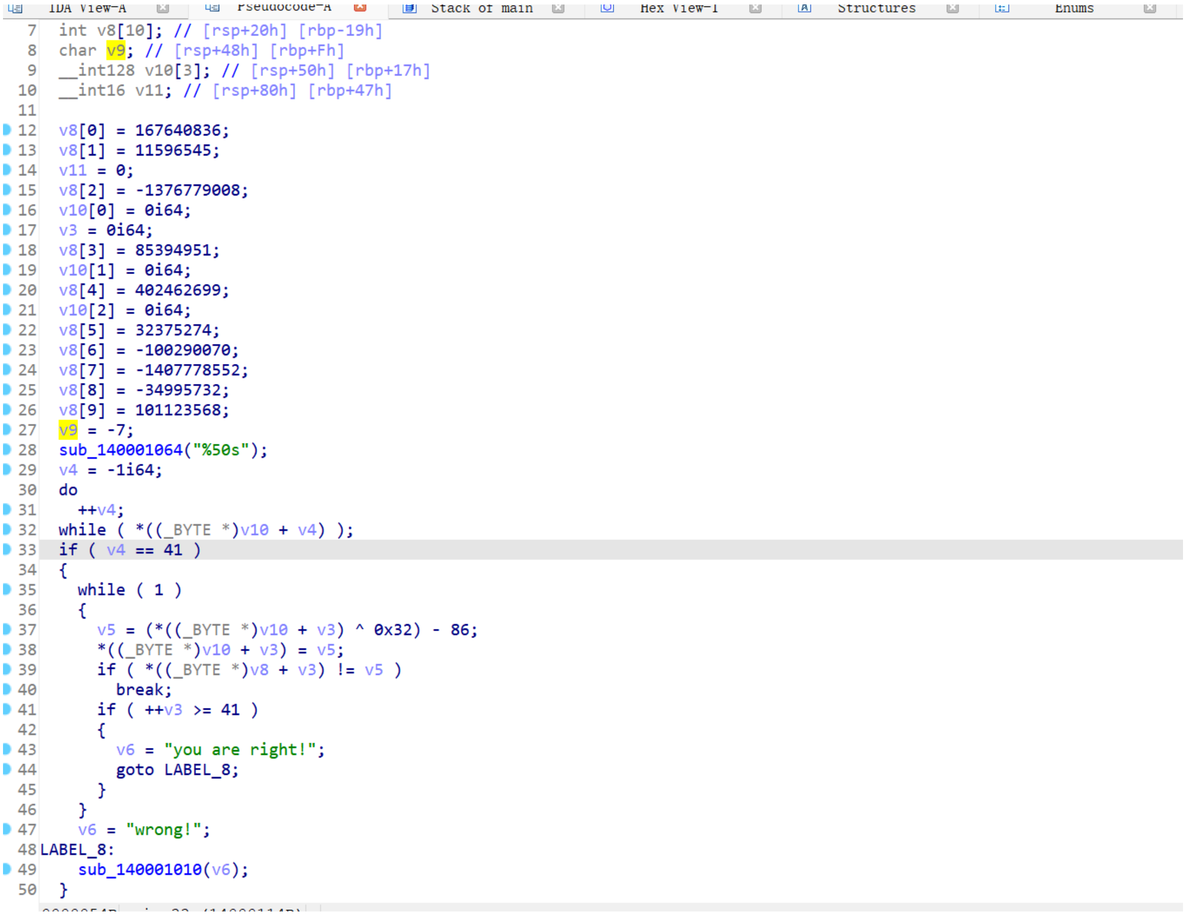
将前面提取的数据输入，运行得到flag



```
8,6,7,6,1,6,13,6,5,6,11,7,5,6,14,6,3,6,15,6,4,6,5,6,15,5,9,6,3,7,15,5,5,6,1,6,3,7,9,7,15,5,6,6,15,6,2,7,15,5,1,6,15,5,2,
7,5,6,6,7,5,6,2,7,3,7,5,6,15,5,5,6,14,6,7,6,9,6,14,6,5,6,5,6,2,7,13,7,0,0,0,0,0,0,0,0,0,0,0,0,
hgame{encode_is_easy_for_a_reverse_engineer}
--------------------------------
Process exited after 4.339 seconds with return value 0
请按任意键继续. . . |
```

## 4.easyenc

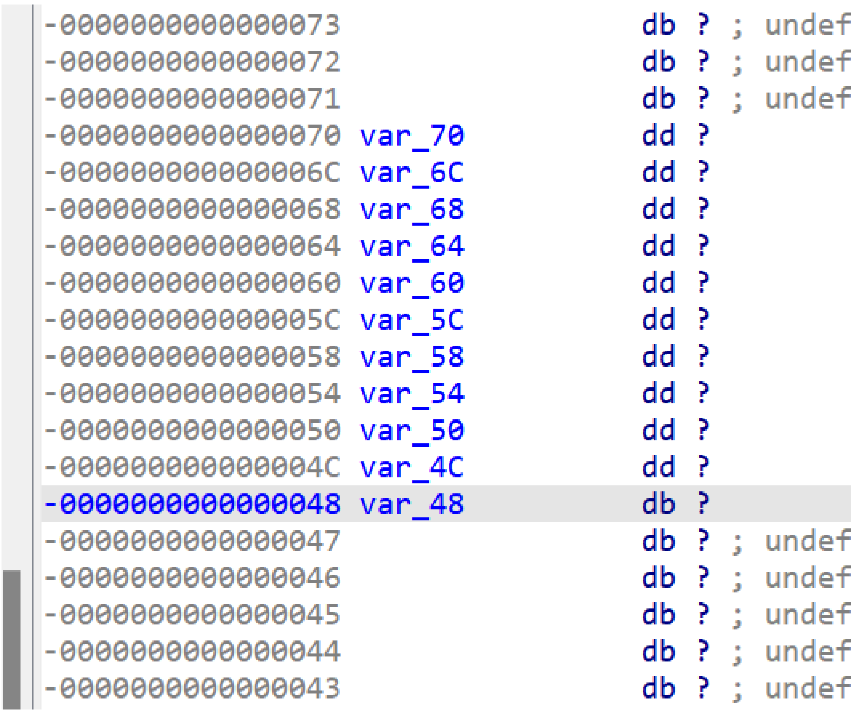
ida里f5反汇编



```
  int v8[10]; // [rsp+20h] [rbp-19h]
  char v9; // [rsp+48h] [rbp+Fh]
  __int128 v10[3]; // [rsp+50h] [rbp+17h]
  __int16 v11; // [rsp+80h] [rbp+47h]

  v8[0] = 167640836;
  v8[1] = 11596545;
  v11 = 0;
  v8[2] = -1376779008;
  v10[0] = 0i64;
  v3 = 0i64;
  v8[3] = 85394951;
  v10[1] = 0i64;
  v8[4] = 402462699;
  v10[2] = 0i64;
  v8[5] = 32375274;
  v8[6] = -100290070;
  v8[7] = -1407778552;
  v8[8] = -34995732;
  v8[9] = 101123568;
  v9 = -7;
  sub_140001064("%50s");
  v4 = -1i64;
  do
    ++v4;
  while ( *((_BYTE *)v10 + v4) );
  if ( v4 == 41 )
  {
    while ( 1 )
    {
      v5 = (*((_BYTE *)v10 + v3) ^ 0x32) - 86;
      *((_BYTE *)v10 + v3) = v5;
      if ( *((_BYTE *)v8 + v3) != v5 )
        break;
      if ( ++v3 >= 41 )
      {
        v6 = "you are right!";
        goto LABEL_8;
      }
    }
    v6 = "wrong!";
LABEL_8:
    sub_140001010(v6);
  }
```

byte就是unsigned char

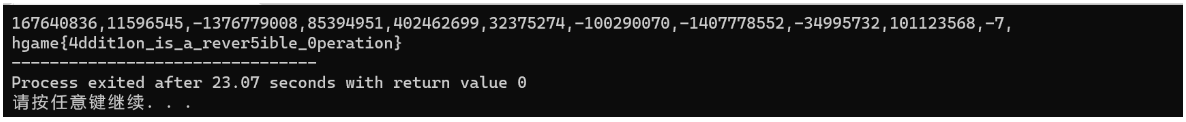
现在需要通过已知的数据求出v10(41字节)，而v8的一个元素为4字节，总共10个元素，这里有40个字节，还差一个，双击v9查看一下



```
-0000000000000073                  db ? ; undef
-0000000000000072                  db ? ; undef
-0000000000000071                  db ? ; undef
-0000000000000070 var_70           dd ?
-000000000000006C var_6C           dd ?
-0000000000000068 var_68           dd ?
-0000000000000064 var_64           dd ?
-0000000000000060 var_60           dd ?
-000000000000005C var_5C           dd ?
-0000000000000058 var_58           dd ?
-0000000000000054 var_54           dd ?
-0000000000000050 var_50           dd ?
-000000000000004C var_4C           dd ?
-0000000000000048 var_48           db ?
-0000000000000047                  db ? ; undef
-0000000000000046                  db ? ; undef
-0000000000000045                  db ? ; undef
-0000000000000044                  db ? ; undef
-0000000000000043                  db ? ; undef
```

发现，v9存在v8[10]的后面，正好1字节，接下来就可以写脚本跑出来了

```
#include<stdio.h>
int main()
{
    int a[100],i=0,count=0,j=0,n=0;
    unsigned int number=0;
    for(i=0;i<11;i++){
        scanf("%d,",&number);
        n=4;
        for(j=count;n--;j++,count++){
            a[j]=number%256;
            number/=256;
            if(i==10){
                n=0;
            }
        }
    }
        for(i=0;i<count;i++){
        a[i]=(a[i]+86)%256;
        a[i]^=50;
        printf("%c",a[i]);
        }
    return 0;
}
```

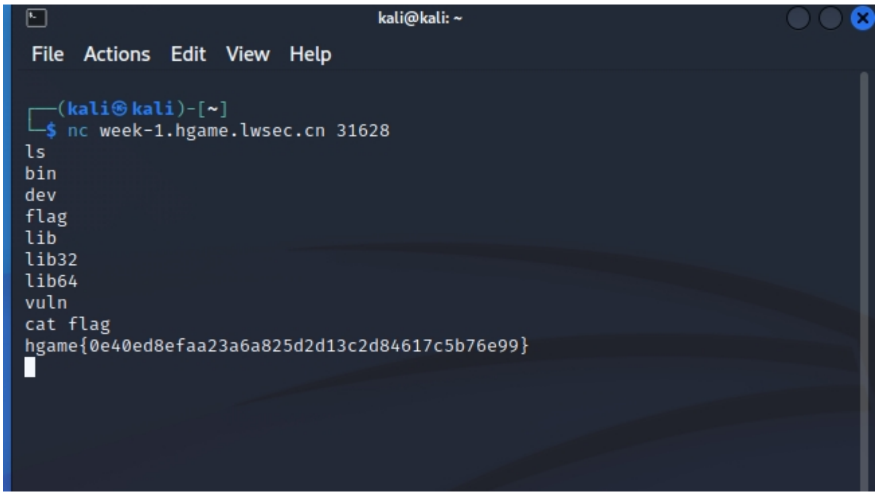


```
167640836,11596545,-1376779008,85394951,402462699,32375274,-100290070,-1407778552,-34995732,101123568,-7,
hgame{4ddit1on_is_a_rever5ible_0peration}
--------------------------------
Process exited after 23.07 seconds with return value 0
请按任意键继续. . .
```

# Pwn

## 1. test_nc

来签个到，先用nc命令连接，然后ls看看有啥，最后cat flag



```
┌──(kali㉿kali)-[~]
└─$ nc week-1.hgame.lwsec.cn 31628
ls
bin
dev
flag
lib
lib32
lib64
vuln
cat flag
hgame{0e40ed8efaa23a6a825d2d13c2d84617c5b76e99}
```