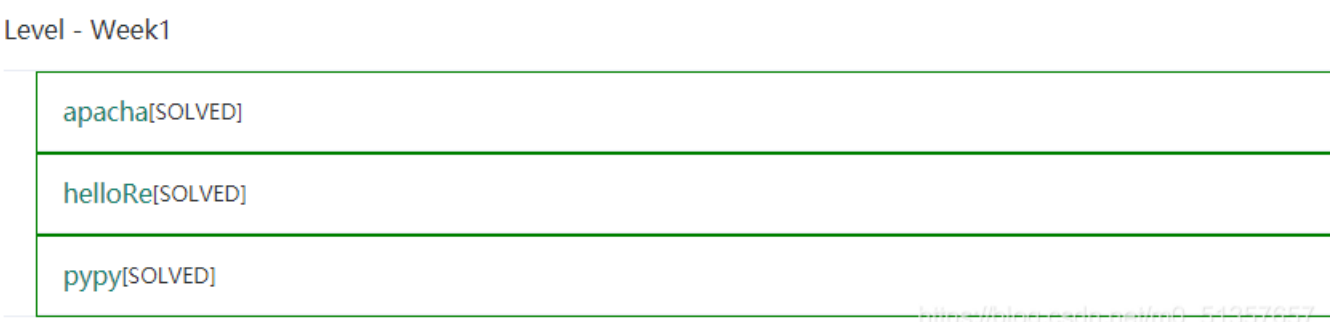
HGAME2021Week1 Writeup

原创

*^_^* 于 2021-02-07 10:53:14 发布

小白做完了re还是很开心的

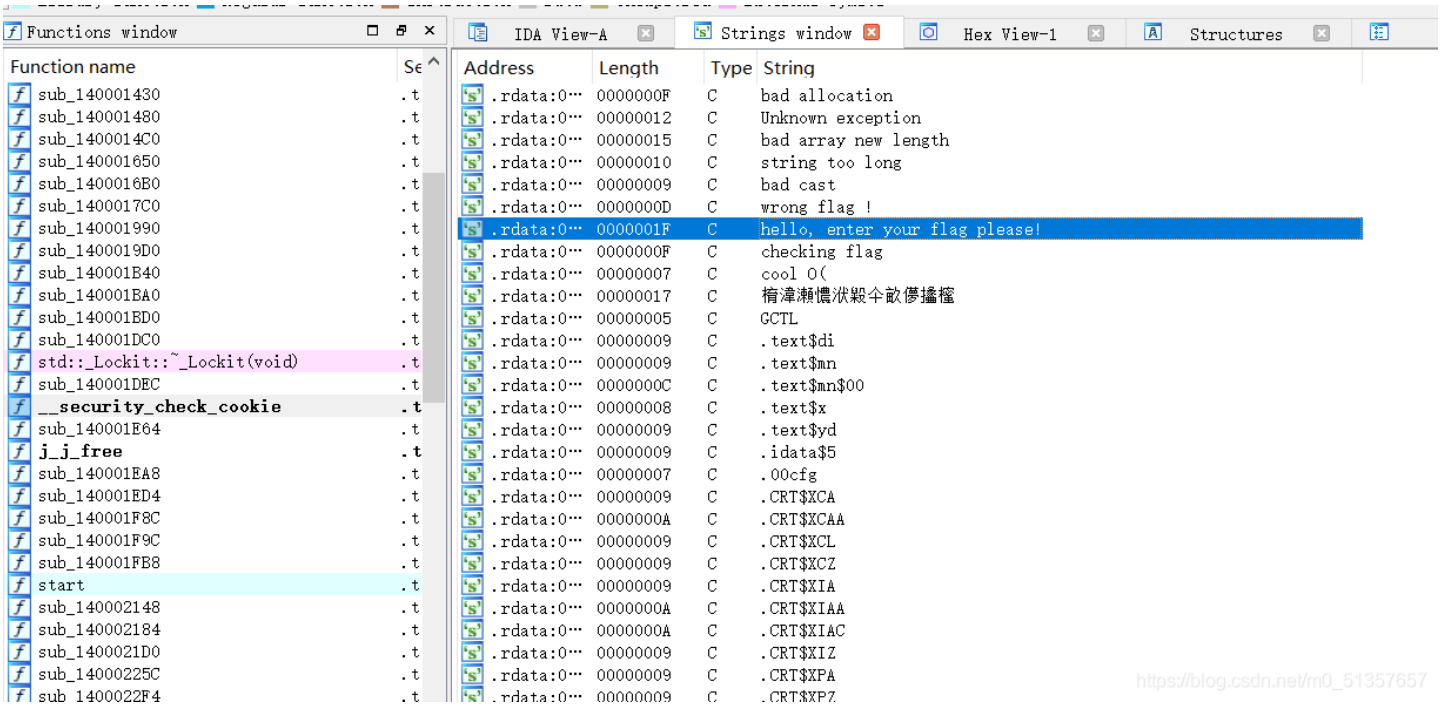


RE部分

Day 1 helloRe

第一天的第一道题~~这道题是真.签到题啊

进入ida，没有明显的main函数，F12查找字符串，看到input flag点进去



发现主函数应该是sub_1400014C0

```
v12 = 15i64;
LOBYTE(Memory) = 0;
```

```
  v1 = sub_1400017C0(std::cout, (__int64)"hello, enter your flag please!");
  v2 = (_QWORD *)std::basic_ostream<char,std::char_traits<char>>::operator<<(v1, sub_140001990);
  sub_1400017C0(v2, (__int64)"> ");
  sub_140001BD0(std::cin, &Memory);
  sub_1400017C0(std::cout, (__int64)"checking flag ");
  sub_140001290(200i64);
  if ( v11 != 22 )
LABEL_13:
    sub_140001480();                        // Wrong flag!
  v3 = v12;
  v4 = (void **)Memory;
  do                                        // Memory 是我们的输入
  {
    v5 = &Memory;
    if ( v3 >= 0x10 )
      v5 = v4;
    if ( (*((_BYTE *)v5 + v0) ^ (unsigned __int8)sub_140001430()) != asc_140003480[v0] )// 对输入异或运算后得到asc_140003480
      goto LABEL_13;
    ++v0;
  }
  while ( v0 < 22 );
  v6 = (_QWORD *)std::basic_ostream<char,std::char_traits<char>>::operator<<(std::cout, sub_140001990);
  v7 = sub_1400017C0(v6, (__int64)&unk_140003470);
  std::basic_ostream<char,std::char_traits<char>>::operator<<(v7, sub_140001990);// v7这部分的运算还挺复杂的，但感觉没有用上啊
  if ( v3 >= 0x10 )
  {
    v8 = v4;
    if ( v3 + 1 >= 0x1000 )
    {
      v4 = (void **)*(v4 - 1);
      if ( (unsigned __int64)((char *)v8 - (char *)v4 - 8) > 0x1F )
        invalid_parameter_noinfo_noreturn();
    }
    j_j_free(v4);
```



```
00140003480 asc_140003480    db '楕漳瀬慣狀縠仐畒僂擪檻',0
```

逻辑很简单，脚本也挺好写的

```
int main()
{
    char memory[30] = "";
    int v6 = 0x0FF;
    char asc_[23] = "楕漳瀬慣狀縠仐畒僂擪檻";
    for (int i = 0; i < 23; i++)
```

```
Microsoft Visual Studio 调试控制台
hgame{hello_re_player}?
C:\Users\pengzixuan\source\repos\work6\x64\Debug\work6.exe (进程 2
按任意键关闭此窗口. . .
```

```
#include<iostream>
using namespace std;

int main()
{
 char memory[30] = "";
 int v6 = 0x0FF;
 char asc_[23] = "楕漳瀬憹狀歿仐畂儚攦檶";
 for (int i = 0; i < 23; i++)
 {
  memory[i] = asc_[i] ^ (v6--);
 }
 printf("%s", memory);

}
```

Day 2 pypy

第一次做python反汇编，好在题目比较友好，没有太复杂

照着类似题的wp和dis–python字节码反汇编可以硬翻。。

下面写了注释的哦

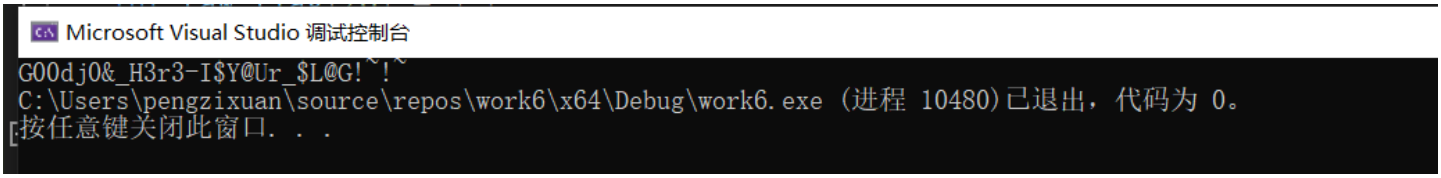
```
 0 LOAD_GLOBAL              0 (input)
             2 LOAD_CONST               1 ('give me your flag:\n')
             4 CALL_FUNCTION            1
             6 STORE_FAST               0 (raw_flag)

  5          8 LOAD_GLOBAL              1 (list)
            10 LOAD_FAST                0 (raw_flag)
            12 LOAD_CONST               2 (6)
            14 LOAD_CONST               3 (-1)
            16 BUILD_SLICE              2                    取raw_flag的第六位到倒数第一位
            18 BINARY_SUBSCR                                 所以最后要包上hgame{}才是flag
            20 CALL_FUNCTION            1
            22 STORE_FAST               1 (cipher)

  6         24 LOAD_GLOBAL              2 (len)
            26 LOAD_FAST                1 (cipher)
            28 CALL_FUNCTION            1                      length=len(cipher)
            30 STORE_FAST               2 (length)

  8         32 LOAD_GLOBAL              3 (range)
            34 LOAD_FAST                2 (length)
            36 LOAD_CONST               4 (2)
            38 BINARY_FLOOR_DIVIDE

            40 CALL_FUNCTION            1
            42 GET_ITER
        >>  44 FOR_ITER                54 (to 100)
            46 STORE_FAST               3 (i)

  9         48 LOAD_FAST                1 (cipher)
            50 LOAD_CONST               4 (2)
            52 LOAD_FAST                3 (i)
            54 BINARY_MULTIPLY
            56 LOAD_CONST               5 (1)
            58 BINARY_ADD
            60 BINARY_SUBSCR                                   cipher[2i+1]
```

```
             60 BINARY_SUBSCR                                     cipher[2i+1]
             62 LOAD_FAST                1 (cipher)
             64 LOAD_CONST               4 (2)
             66 LOAD_FAST                3 (i)
             68 BINARY_MULTIPLY
             70 BINARY_SUBSCR                                     cipher[2i]
             72 ROT_TWO
             74 LOAD_FAST                1 (cipher)
             76 LOAD_CONST               4 (2)
             78 LOAD_FAST                3 (i)
             80 BINARY_MULTIPLY
             82 STORE_SUBSCR
             84 LOAD_FAST                1 (cipher)
             86 LOAD_CONST               4 (2)
             88 LOAD_FAST                3 (i)
             90 BINARY_MULTIPLY
             92 LOAD_CONST               5 (1)
             94 BINARY_ADD
             96 STORE_SUBSCR                               这一段就是x,y=y,x
             98 JUMP_ABSOLUTE           44                 交换了cipher[2i+1]和cipher[2i]

12     >>   100 BUILD_LIST               0
            102 STORE_FAST               4 (res)           #创建一个list

13          104 LOAD_GLOBAL              3 (range)
            106 LOAD_FAST                2 (length)
            108 CALL_FUNCTION            1
            110 GET_ITER
       >>   112 FOR_ITER                26 (to 140)
            114 STORE_FAST               3 (i)

14          116 LOAD_FAST                4 (res)
            118 LOAD_METHOD              4 (append)
            120 LOAD_GLOBAL              5 (ord)
            122 LOAD_FAST                1 (cipher)
            124 LOAD_FAST                3 (i)
            126 BINARY_SUBSCR
            128 CALL_FUNCTION            1
            130 LOAD_FAST                3 (i)
            132 BINARY_XOR                                  res.append(ord(cipher[i])^i)
            134 CALL_METHOD              1
            136 POP_TOP
            138 JUMP_ABSOLUTE          112

15     >>   140 LOAD_GLOBAL              6 (bytes)
            142 LOAD_FAST                4 (res)
            144 CALL_FUNCTION            1
            146 LOAD_METHOD              7 (hex)
            148 CALL_METHOD              0
            150 STORE_FAST               4 (res)          把res里的数据转成16进制

16          152 LOAD_GLOBAL              8 (print)
            154 LOAD_CONST               6 ('your flag: ')
            156 LOAD_FAST                4 (res)
            158 BINARY_ADD                                              输出res
            160 CALL_FUNCTION            1
            162 POP_TOP
            164 LOAD_CONST               0 (None)
            166 RETURN_VALUE
```

```
# your flag: 30466633346f59213b4139794520572b45514d61583151576638643a
```

其实就几行代码，因本废物只能看懂一点python但不会写，所以还是用的C++写脚本

```cpp
#include<iostream>
using namespace std;

int main()
{
    int cipher[28] = { 0x30,0x46,0x66,0x33,0x34,0x6f,0x59,0x21,0x3b,0x41,0x39,0x79,0x45,0x20,
        0x57,0x2b,0x45,0x51,0x4d,0x61,0x58,0x31,0x51,0x57,0x66,0x38,0x64,0x3a };
    int raw_flag[20] = { };
    char flag1[28] = {};
    //char temp;
    for (int i = 0; i < 28; i++)
    {
        flag1[i] = char(cipher[i] ^ i);
    }
    for (int i = 0; i < 17; i++)
    {
        swap(flag1[2 * i], flag1[2 * i + 1]);
    }
    for (int i = 0; i < 28; i++)
    {
        printf("%c", flag1[i]);
    }
    return 0;
}
```



hgame{}包上，提交~（讲真这个flag有点，，不好看，我一直以为自己哪里写错了还

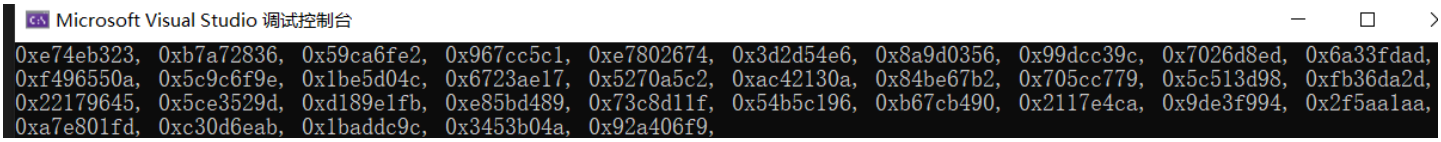
Day 3-5 apacha

压轴，对于我这种废物小白，这道题确实有一丢丢难

其实就是个xxtea加密，但也是看了好多篇wp加之超级可爱的学长的帮助下才做完的

```
do
{
  v6 -= 1640531527;
  v7 = v6 >> 2;                                 // 差值为-1640531527的hex
                                                // -1640531527=2*2147483647-1640531527=2654435769
                                                // hex(2654435769)=0x9e3779b9
  if ( a2 == 1 )
  {
    v9 = 0;
  }
  else
  {
    v8 = 0LL;
    do
    {
      v5 = a1[v8]
         + ((((v5 >> 5) ^ 4 * a1[v8 + 1]) + (16 * v5 ^ (a1[v8 + 1] >> 3))) ^ ((*(_DWORD *)(v3
                                                                                          + 4LL
                                                                                          * (((unsigned __int8)v8 ^ (unsigned __int8)v7) & 3)) ^ v5)
                                                                                 + (a1[v8 + 1] ^ v6)));
      a1[v8++] = v5;
    }                                           // (((z>>5^y<<2) + (y>>3^z<<4)) ^ ((sum^y) + (key[(p&3)^e] ^ z)))挺明显的xxtea加密
    while ( v8 != (unsigned int)(a2 - 2) + 1LL );
    v9 = a2 - 1;
  }
  result = 16 * v5 ^ (*a1 >> 3);
  v5 = *v4
     + (((*(_DWORD *)(v3 + 4LL * ((v9 ^ (unsigned __int8)v7) & 3)) ^ v5) + (*a1 ^ v6)) ^ ((4 * *a1 ^ (v5 >> 5))
                                                                                        + result));
  *v4 = v5;
}
while ( v6 != -1640531527 * (52 / a2) - 1253254570 );
return result;
```


再把dword_5020里的数据转换成16进制数



```
void hex()
{
    uint32_t v[] = { 35, 179,  78, 231,  54,  40, 167, 183, 226, 111,
  202,  89, 193, 197, 124, 150, 116,  38, 128, 231,
  230,  84,  45,  61,  86,   3, 157, 138, 156, 195,
  220, 153, 237, 216,  38, 112, 173, 253,  51, 106,
   10,  85, 150, 244, 158, 111, 156,  92,  76, 208,
  229,  27,  23, 174,  35, 103, 194, 165, 112,  82,
   10,  19,  66, 172, 178, 103, 190, 132, 121, 199,
   92, 112, 152,  61,  81,  92,  45, 218,  54, 251,
   69, 150,  23,  34, 157,  82, 227,  92, 251, 225,
  137, 209, 137, 212,  91, 232,  31, 209, 200, 115,
  150, 193, 181,  84, 144, 180, 124, 182, 202, 228,
   23,  33, 148, 249, 227, 157, 170, 161,  90,  47,
  253,   1, 232, 167, 171, 110,  13, 195, 156, 220,
  173,  27,  74, 176,  83,  52, 249,   6, 164, 146, };
    for (int i = 1; i <= 35; i++)
    {
        printf("0x");
        for (int j = i * 4 - 1; j > (i - 1) * 4 - 1; j--)
            if (v[j] < 16)
            {
                printf("0%x", v[j]);
            }
            else printf("%x", v[j]);
        printf(", ");
    }

}
```

对了，密钥的位置传入的是&v6，所以密钥就是1，2，3，4

```
  v6 = 1;
  v7 = 2;
  v8 = 3;
  v9 = 4;
  sub_11AA(a1, a2, a3);
  __printf_chk(1LL, "Please input: ");
  __isoc99_scanf("%35s", v10);
  if ( (unsigned int)strlen(v10) != 35 )
  {
    puts("wrong length!");
    exit(0);
  }
  v3 = malloc(0x8CuLL);
  v4 = 0LL;
  do
  {
    v3[v4] = v10[v4];
    ++v4;
  }
  while ( v4 != 35 );
  sub_1447(v3, 35, (__int64)&v6);
```


上网搜解密脚本套用即可

```
#include <stdio.h>
#include <stdint.h>
#include<iostream>
#define DELTA 0x9e3779b9
#define MX (((z>>5^y<<2) + (y>>3^z<<4)) ^ ((sum^y) + (key[(p&3)^e] ^ z)))
```

```cpp
using namespace std;

uint32_t v3[] = { 0xe74eb323, 0xb7a72836, 0x59ca6fe2, 0x967cc5c1, 0xe7802674,
0x3d2d54e6, 0x8a9d0356,0x99dcc39c, 0x7026d8ed, 0x6a33fdad, 0xf496550a, 0x5c9c6f9e,
0x1be5d04c, 0x6723ae17, 0x5270a5c2, 0xac42130a,
0x84be67b2, 0x705cc779, 0x5c513d98, 0xfb36da2d, 0x22179645, 0x5ce3529d, 0xd189e1fb,
0xe85bd489, 0x73c8d11f,
0x54b5c196, 0xb67cb490, 0x2117e4ca, 0x9de3f994, 0x2f5aa1aa, 0xa7e801fd, 0xc30d6eab,
0x1baddc9c, 0x3453b04a, 0x92a406f9, };


void btea(uint32_t* v, int n, uint32_t const key[4])
{
    uint32_t y, z, sum;
    unsigned p, rounds, e;
    if (n > 1)            /* Coding Part */
    {
        rounds = 3;
        sum = 0;
        z = v[n - 1];
        do
        {
            sum += DELTA;
            e = (sum >> 2) & 3;
            for (p = 0; p < n - 1; p++)
            {
                y = v[p + 1];
                z = v[p] += MX;
            }
            y = v[0];
            z = v[n - 1] += MX;
        } while (--rounds);
    }
    else if (n < -1)      /* Decoding Part */
    {
        n = -n;
        rounds = 6 + 52 / n;
        sum = rounds * DELTA;
        y = v[0];
        do
        {
            e = (sum >> 2) & 3;
            for (p = n - 1; p > 0; p--)
            {
                z = v[p - 1];
                y = v[p] -= MX;
            }
            z = v[n - 1];
            y = v[0] -= MX;
            sum -= DELTA;
        } while (--rounds);
    }
}



int main()
{
```

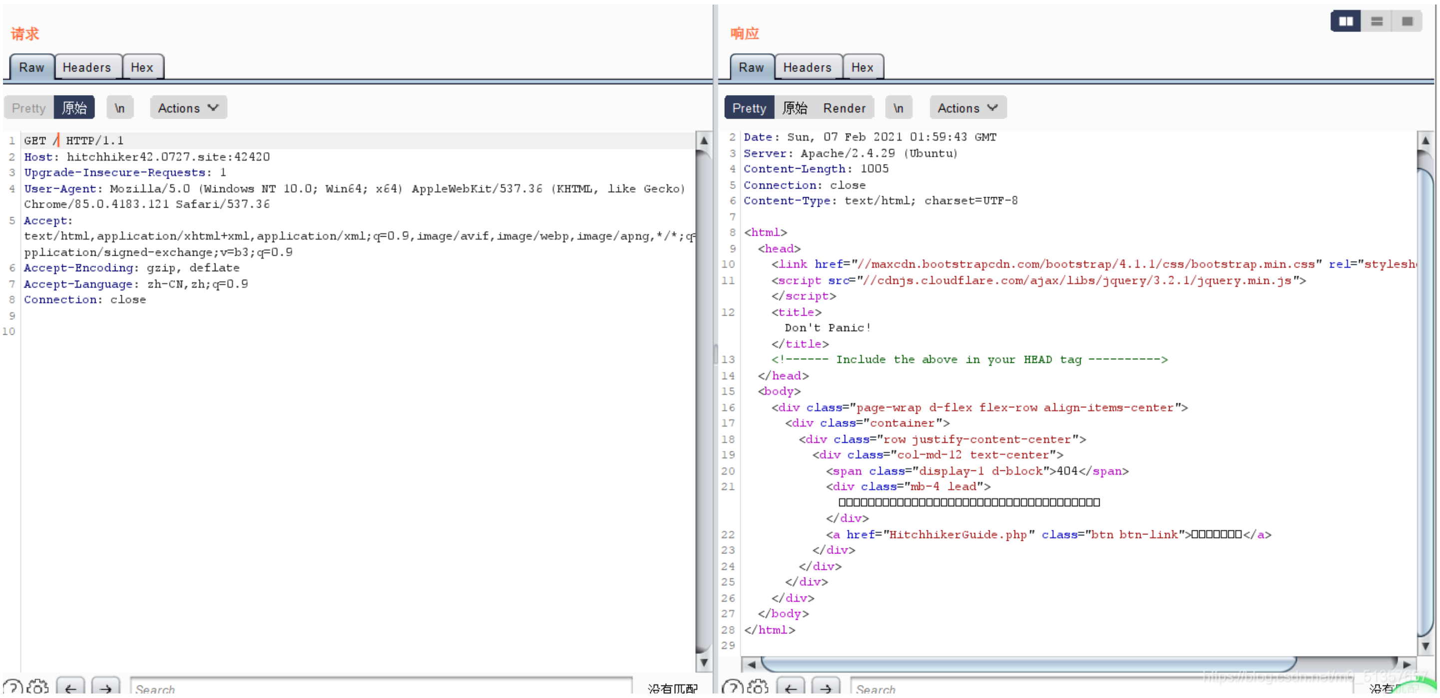
```
    uint32_t const k[4] = { 1,2,3,4 };

    int n = 35; //n的绝对值表示v的长度，取正表示加密，取负表示解密
    // v为要加密的数据是两个32位无符号整数
    // k为加密解密密钥，为4个32位无符号整数，即密钥长度为128位
    //printf("加密前原始数据：%u %u\n", v[0], v[1]);
    //btea(v, n, k);
   // printf("加密后的数据：%u %u\n", v[0], v[1]);
   btea(v3, -n, k);
   for (int i = 0; i < 35; i++)
   {
       printf("%c", v3[i]);
    }

    return 0;
}
```

```
Microsoft Visual Studio 调试控制台
hgame{100ks_like_y0u_f0Und_th3_t34}
C:\Users\pengzixuan\source\repos\Hackgame\Debug\Hackgame.exe (进程 10280)已退出，代码
按任意键关闭此窗口. . .
```

呜呜呜week2就不会做了，我还是太废物了。。。

WEB

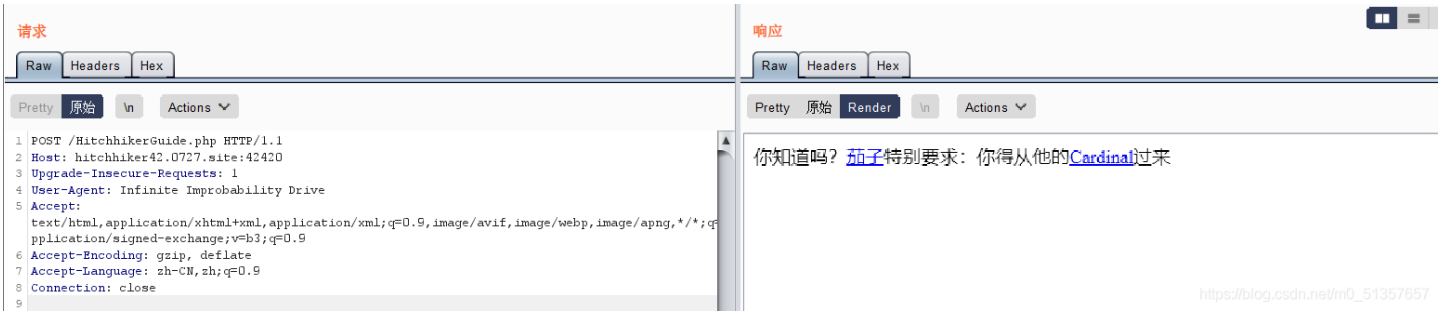
只做了常规套娃的签到题



GET不行。。改成POST。。。之后就全程跟着提示安静套娃

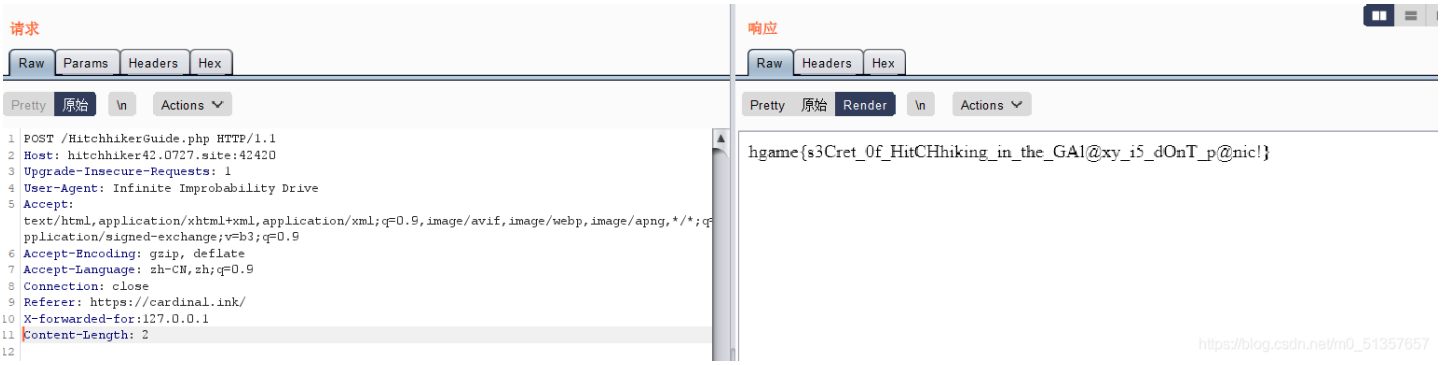
改一下UA

加个Referer头



加个XFF:127.0.0.1

搞定。。



唉，本来还想做一道合成大西瓜的题。。。一看全是js我就怂了

CRYPTO

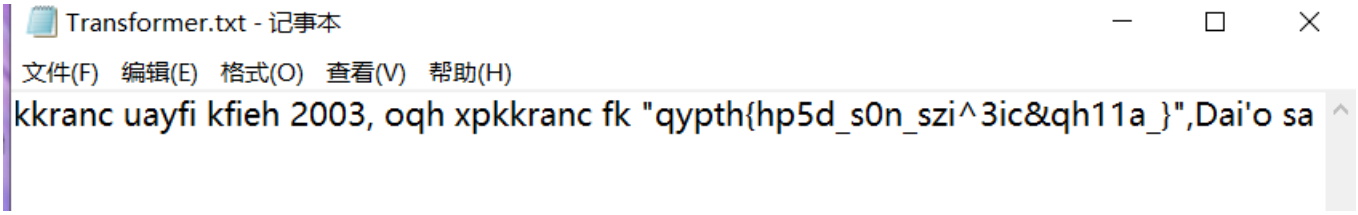
Transformer[SOLVED]

Description

所有人都已做好准备,月黑之时即将来临,为了击毁最后的主控能量柱,打开通往芝加哥的升降桥迫在眉睫 看守升降桥的控制员已经失踪,唯有在控制台的小房间留下来的小纸条,似乎是控制员防止自己老了把密码忘记而写下的,但似乎都是奇怪的字母组合,唯一有价值的线索是垃圾桶里的两堆被碎纸机粉碎的碎纸,随便查看几张,似乎是两份文件,并且其中一份和小纸条上的字母规律有点相像 附件md5:0340142700c8f63546368fa14fd6fb24

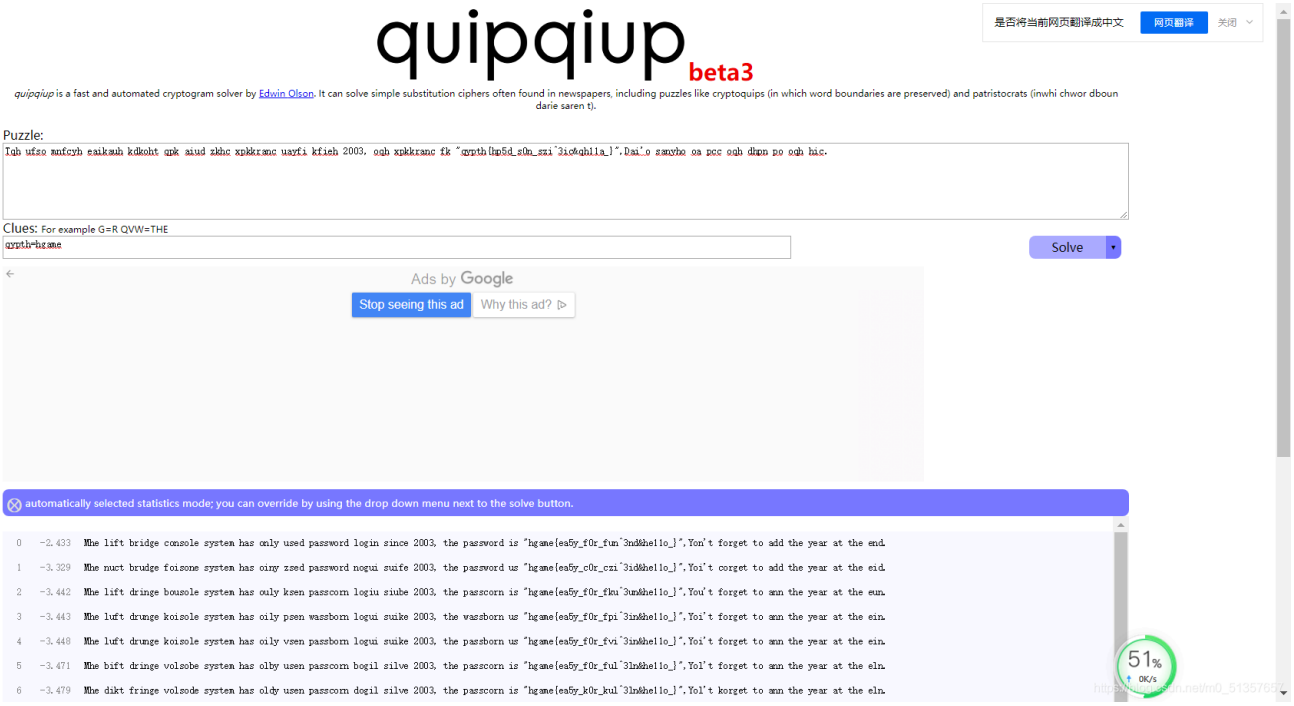
Challenge Address https://1.oss.hgame2021.vidar.club/Transformer.zip

题目里的那串md5想不通什么意思，而且md5也不好解，先看txt



乱中有序哈哈哈哈，替换加密，已知qypyh=hgame可以直接爆破

解密网站链接



据提示要在最后加年份。。试了2003。。不行。。头秃。。

灵光一现。。试个2021。。。成功！

打开文件，摩斯电码，找个在线网站解一下

```
86/109/108/110/90/87/53/108/99/109/85/116/84/71/108/114/97/84/112/57/86/109/116/116/100/107/112/105/73/84/70/89/100/69/70/52/90/83/70/111/99/69/48/120/101/48/48
/114/79/88/104/120/101/110/74/85/84/86/57/79/97/110/53/106/85/109/99/48/101/65/61/61
```

盲猜是ASCII码，VS里跑一下是一串base64（等号结尾很明显）

据提示维吉尼亚密码，密钥Liki。。。
}KccnYt!1NlPpu!zeE1{C+9pfrhLB_Fz~uGy4n

然后我就不会了，下面是神仙操作：

栅栏 6 ：}!!Ch~K1z+LucNe9BGclEp_ynP1fF4Yp{rzntu

rot13：}!!Pu~X1m+YhpAr9OTpyRc_laC1sS4Lc{emagh

reverse：hgame{cL4Ss1Cal_cRypTO9rAphY+m1X~uP!!}

（问神仙，神仙说他是猜的。。）

MISC

Base全家福[SOLVED]

Description

新年即将来临之际，Base家族也团聚了，他们用他们特有的打招呼方式向你问了个好，你知道他们在说什么吗?

R1k0RE1OWldHRTNFSU5SVkc1QkRLTlpXR1VaVENOUlRHTVlETVJCV0dVMlVNTlpVR01ZREtSUlVIQTJET01aVUdSQ0RHTVpWSVlaVEVNWlFHTVpER01KWEIRPT09PT09

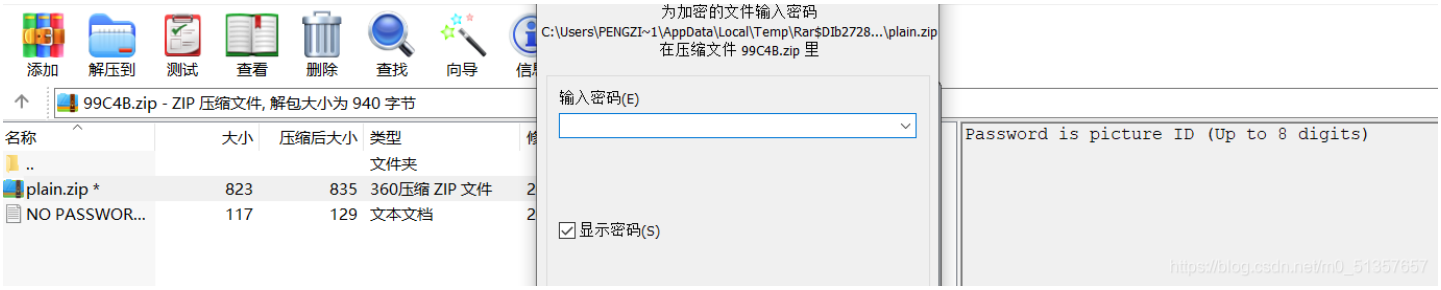
真.签到题，base64，base32，base16一通解就行

不起眼压缩包的养成的方法[SOLVED]

唉，差点没被这道题套娃套死

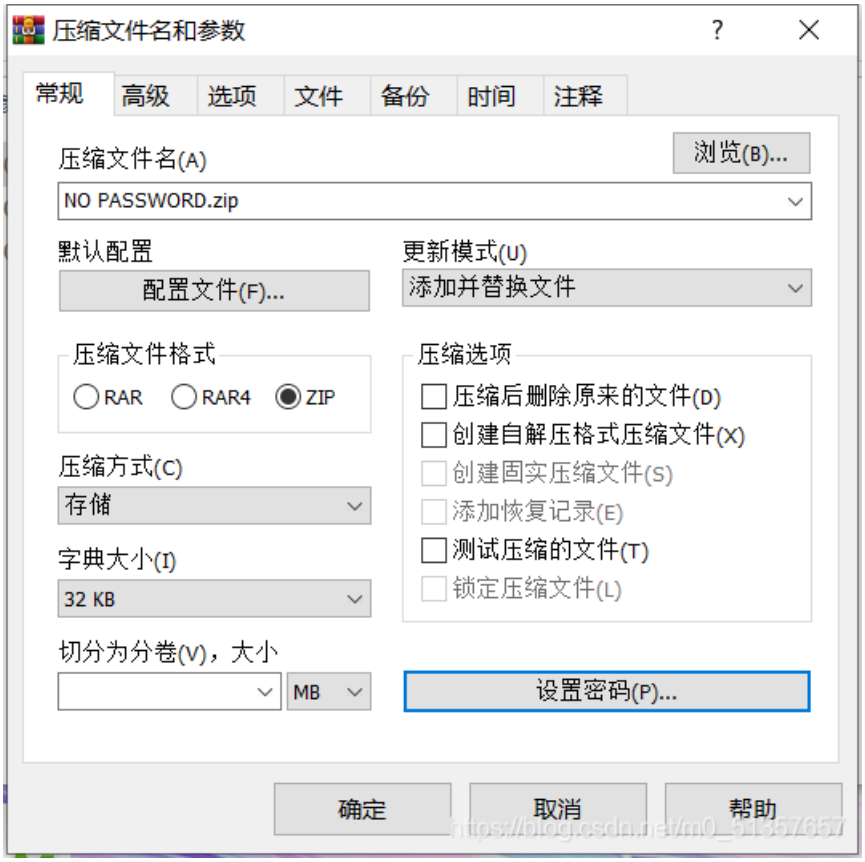
给了张图片，据提示应该是藏了压缩包，binwalk分离

得到加密压缩包，提示密码是八位数字，暴力破解即可



又来一层加密，很明显的明文攻击，试了好久不行，根据提示storage

在给明文压缩时选择压缩方式为存储，密码为zip传统加密即可



又来加密压缩包。。。看了一眼不是伪加密.。。。求助学长，被秒解，原来旁边这一串是html编码。。。。是我见识太少，意识太差了呜呜呜

Offset(h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	对应文本
00000000	50	4B	03	04	14	00	09	00	00	00	F3	AB	3D	52	43	97	PK.....ó«=RC—
00000010	03	00	F0	00	00	00	F0	00	00	00	08	00	00	00	66	6C	..ð...ð.....fl
00000020	61	67	2E	74	78	74	26	23	78	36	38	3B	26	23	78	36	ag.txth&#x6
00000030	37	3B	26	23	78	36	31	3B	26	23	78	36	44	3B	26	23	7;am&#
00000040	78	36	35	3B	26	23	78	37	42	3B	26	23	78	33	32	3B	x65;{2
00000050	26	23	78	34	39	3B	26	23	78	35	30	3B	26	23	78	35	IP&#x5
00000060	46	3B	26	23	78	36	39	3B	26	23	78	37	33	3B	26	23	F;is&#
00000070	78	35	46	3B	26	23	78	35	35	3B	26	23	78	37	33	3B	x5F;Us
00000080	26	23	78	36	35	3B	26	23	78	36	36	3B	26	23	78	37	ef&#x7
00000090	35	3B	26	23	78	33	31	3B	26	23	78	35	46	3B	26	23	5;1_&#
000000A0	78	36	31	3B	26	23	78	36	45	3B	26	23	78	36	34	3B	x61;nd
000000B0	26	23	78	35	46	3B	26	23	78	34	44	3B	26	23	78	36	_M&#x6
000000C0	35	3B	26	23	78	33	39	3B	26	23	78	37	35	3B	26	23	5;9u&#
000000D0	78	36	44	3B	26	23	78	36	39	3B	26	23	78	35	46	3B	x6D;i_
000000E0	26	23	78	36	39	3B	26	23	78	33	35	3B	26	23	78	35	i5&#x5
000000F0	46	3B	26	23	78	35	37	3B	26	23	78	33	30	3B	26	23	F;W0&#
00000100	78	37	32	3B	26	23	78	33	31	3B	26	23	78	36	34	3B	x72;1d
00000110	26	23	78	37	44	3B	50	4B	01	02	14	00	14	00	01	00	}PK.....
00000120	00	00	F3	AB	3D	52	43	97	03	00	F0	00	00	00	F0	00	..ó«=RC—..ð...ð.
00000130	00	00	08	00	24	00	00	00	00	00	00	00	20	00	00	00	....$..... ...
00000140	00	00	00	00	66	6C	61	67	2E	74	78	74	0A	00	20	00	....flag.txt.. .
00000150	00	00	00	00	01	00	18	00	13	33	1B	11	43	F6	D6	01	.........3..CöÖ.
00000160	CA	4C	45	30	43	F6	D6	01	EE	BA	BE	6B	7D	F5	D6	01	ÊLE0CöÖ.î°¾k}õÖ.
00000170	50	4B	05	06	00	00	00	00	01	00	01	00	5A	00	00	00	PK.....Z...
00000180	16	01	00	00	00	00											.....

PWN

。。。除了最最最基础的栈溢出能对着wp做其他我都不会，于是签到题就跪了
。。。pwn太难了呜呜呜。。。