RE 部分：

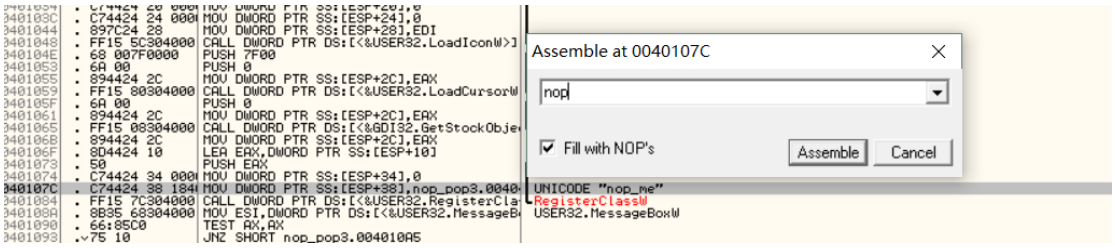
RE0

IDA 打开文件，查找字符串 flag，就找到 flag 了

Nop_pop

OD 打开 nop_pop.exe

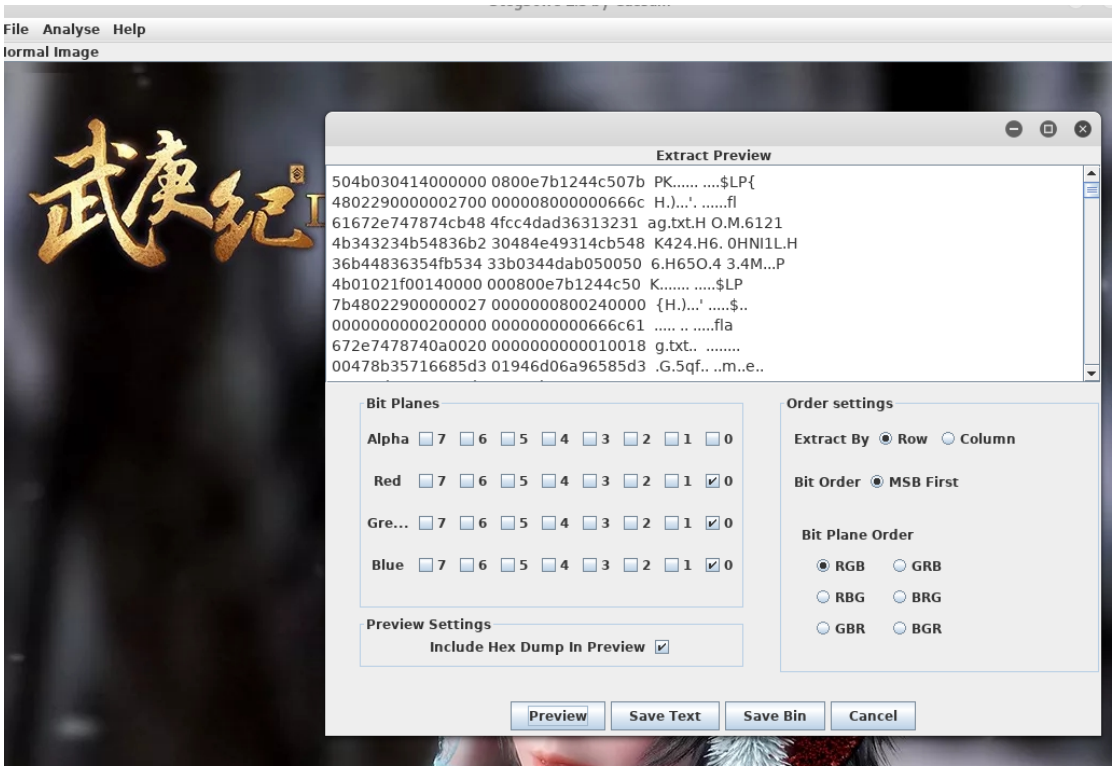
发现这里有一个 nop_me，然后就 nop 试试看，然后就有 flag 了。。。



Misc 部分：

白菜 1

用 Stegsolve 分析，在



发现了 flag.txt

使用以下脚本提取

```
from PIL import Image

im = Image.open("flag.png")
pix = im.load()
width, height = im.size

extracted_bits = []
for y in range(height):
    for x in range(width):
        r, g, b = pix[(x,y)]
        extracted_bits.append(r & 1)
        extracted_bits.append(g & 1)
        extracted_bits.append(b & 1)

extracted_byte_bits = [extracted_bits[i:i+8] for i in range(0, len(extracted_bits), 8)]
with open("extracted2.bmp", "wb") as out:
    for byte_bits in extracted_byte_bits:
        byte_str = ''.join(str(x) for x in byte_bits)
        byte = chr(int(byte_str, 2))
        out.write(byte)
```

再 binwalk 分离出 flag.txt

白菜 2

Winhex 打开图片，发现有一个 flag.txt

```
000FD400  00 00 00 00 20 00 00 00  00 00 00 00 66 6C 61 67                flag
000FD410  2E 74 78 74 0A 00 20 00  00 00 00 00 01 00 18 00  .txt
000FD420  74 1D D6 27 F6 85 D3 01  7E D4 60 07 F6 85 D3 01  t Ö'öIÓ ~Ô` öIÓ
000FD430  7E D4 60 07 F6 85 D3 01  50 4B 05 06 00 00 00 00  ~Ô` öIÓ PK
000FD440  01 00 01 00 5A 00 00 00  4F 00 00 00 00 00        ■   Z   O
```

把后缀改为.rar 再解压出来，得到 flag.txt

flag.png	2018/2/4 22:14	PNG 文件	1,763 KB
flag.txt	2018/1/5 15:24	文本文档	1 KB

Pacp1

Wireshark 打开数据包，查找字符串 flag，就找到了 flag

密码学

Easy caeser

使用下面的网址可以在线加解密

http://planetcalc.com/1434/

ROT12	hgame{The_qu8ck_br7wn_1x_jUmps_ovEr_a_La9y_dOg}

将数字也凯撒一下,得到
hgame{The_qu1ck_br0wn_4x_jUmps_ovEr_a_La2y_dOg}

Hill
网上找了一个 c 程序，一开始输入密钥 9 17 6 5，不成功，换个顺序再试，9 6 17 5，成功

```
======= Hill 密码 =======

请输入秘钥的值:
9 6 17 5

1. 加密  2. 解密
请选择: 2
请输入密文: phnfetzhzzwz
解密结果为:
overthehillx
```

程序如下

```
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#define MAX 60

int main()
{
	int K1[2][2] = {0}, K2[2][2] = {0};
	int Temp1[2] = {0}, Temp2[2] = {0};
	char P[MAX] = {0}, C[MAX] = {0};
	int T1[MAX] = {0}, T2[MAX] = {0};
	int len, flag=0, temp, temp1, i, j, num=0;


	printf("======= Hill 密码 =======\n\n");
	printf("请输入秘钥的值：\n");
	for(i=0; i<2; i++)
	{
		for(j=0; j<2; j++)
		{
			scanf("%d", &K1[i][j]);
		}
```

```
}

printf("\n1. 加密\t 2. 解密\n 请选择：");
scanf("%d", &num);

if(num == 1)
{
    printf("请输入明文：\n");
    scanf("%s", P);

    len = strlen(P);

    // 当长度为奇数时补齐一位
    if(len % 2 == 1)
    {
        P[len] = 'a';
        len = strlen(P);
        flag = 1;
    }

    // 将大写转成小写，并赋值给 T1 数组
    for(i=0; i<len; i++)
    {
        if(P[i] >= 'A' && P[i] <= 'Z')
        {
            P[i] = P[i] + 32;
        }

        T1[i] = P[i] - 'a';
    }


    // 得到加密后结果，存储在 T2 中
    for(i=0; i<len; i+=2)
    {
        Temp1[0] = T1[i];
        Temp1[1] = T1[i + 1];

        // Temp2 存储密文 int 值
        Temp2[0] = (Temp1[0] * K1[0][0] + Temp1[1] * K1[1][0]) % 26;
        Temp2[1] = (Temp1[0] * K1[0][1] + Temp1[1] * K1[1][1]) % 26;

        T2[i] = Temp2[0];
        T2[i + 1] = Temp2[1];
```

```
        }

        if(flag == 1)
        {
            len = len - 1;
        }

        printf("加密结果为：\n");
        for(i=0; i<len; i++)
        {
            C[i] = T2[i] + 'a';
            printf("%c ", C[i]);
        }
        printf("\n");
    }

    else if(num == 2)
    {
        printf("请输入密文：");
        scanf("%s", C);

        len = strlen(C);

        // 当长度为奇数时补齐一位
        if(len % 2 == 1)
        {
            C[len] = 'a';
            len = strlen(C);
            flag = 1;
        }


        for(i=0; i<len; i++)
        {
            if(C[i] >= 'A' && C[i] <= 'Z')
            {
                C[i] = C[i] + 32;
            }

            T2[i] = C[i] - 'a';
        }

        // 求 K 的逆
        temp = -1;
```

```
        for(i=1; temp < 0; i++)
        {
            temp = (K1[0][0] * K1[1][1] - K1[0][1] * K1[1][0]) + 26 * i;
        }

        i = 1;
        while(1)
        {
            if((temp * i) % 26 == 1)
            {
                temp1 = i;
                break;
            }
            else
            {
                i++;
            }
        }

        K2[0][0] = K1[1][1] * temp1;
        K2[0][1] = (((-1 * K1[0][1]) + 26) * temp1) % 26;
        K2[1][0] = (((-1 * K1[1][0]) + 26) * temp1) % 26;
        K2[1][1] = K1[0][0] * temp1;

//      printf(" %d %d     %d %d %d %d\n",temp, temp1, K2[0][0], K2[0][1], K2[1][0], K2[1][1]);
//      system("pause");
//      printf(" %d %d     %d %d %d %d\n",temp, temp1, K2[0][0]%26, K2[0][1]%26, K2[1][0]%26,
K2[1][1]%26);
//      system("pause");

        // 得到解密后结果，存储在 T2 中
        for(i=0; i<len; i+=2)
        {
            Temp2[0] = T2[i];
            Temp2[1] = T2[i + 1];

            // Temp1 存储明文 int 值
            Temp1[0] = (Temp2[0] * K2[0][0] + Temp2[1] * K2[1][0]) % 26;
            Temp1[1] = (Temp2[0] * K2[0][1] + Temp2[1] * K2[1][1]) % 26;

            T1[i] = Temp1[0];
            T1[i + 1] = Temp1[1];
        }
```

```
        if(flag == 1)
        {
            len = len - 1;
        }

        printf("解密结果为：\n");
        for(i=0; i<len; i++)
        {
            P[i] = T1[i] + 'a';
            printf("%c", P[i]);
        }
        printf("\n");

    }

    else
    {
        printf("error!");
        exit(0);
    }



    return 0;
}
```

Confusion

根据题目特点，先解摩斯密码得到

MRLTK6KXNVZXQWBSNA2FSU2GGBSW45BSLAZFU6SVJBNDAZSRHU6Q====

然后尾部有====解 base64，发现行不通，再试试解 base32，得到

dW5yWmsxX2h4YSF0ent2X2ZzUHZ0fQ==

尾部有==，再解 base64，得到

unrZk1_hxa!tz{v_fsPvt}

因为 flag 是 hgame{xxxx}格式，判断应该是栅栏，解得

utnzr{Zvk_1f_shPxvat!}

然后最后再解凯撒

hgame{Mix_1s_fuCking!}

Web 部分:

Are you from Europe?

查看源码，发现有一段 js 混淆

```
,
eval(function(p,a,c,k,e,d){e=function(c){return(c<a?"":e(parseInt(c/a)))+((c=c%a)>35?String.fromCharCode(c+29):c.toString(36))};if(!''.replace(/^/,String)){while(c--)d[e(c)]=k[c]||e(c);k=[function(e){return
d[e]};e=function(){return'\\w+'};c=1;};while(c--)if(k[c])p=p.replace(new RegExp('\\b'+e(c)+'\\b','g'),k[c]);return p;}('w p()b a=1("8! k, j 8 o 1 m? ");5(!$("#4"),7()&&!$("#9"),7())2("i. ");f}b
l="",l+="",l+="",l+="",l+="y";l+="{";l+="g";l+="h";l+="3";l+="6";l+="v";l+="h";l+="0";l+="g";l+="m";l+="q";l+="6";l+="0";l+="n";l+="r";l+="1";l+="}";5(a){$("#4").d();$("#9").d();2("t. 1:
"+i)}q(2("x. "))}',35,35,'|flag|alert||serv5|if|_|html|SSR|craft5|buy|var||remove||return|T||你根本不是欧洲人|你愿意献学你全部的|欧洲人|confirm|吗||未获取|soHappy|else|E||兑换成功|N|C|function|你失去了唯一的机
会|hgame'.split('|'),0,{}))
```

http://www.bm8.com.cn/jsConfusion/在线反混淆

```
var flag = "";
flag += "";
flag += "";
flag += "";
flag += "hgame";
flag += "{";
flag += "T";
flag += "h";
flag += "3";
flag += "_";
flag += "C";
flag += "h";
flag += "0";
flag += "s";
flag += "e";
flag += "N";
flag += "_";
flag += "0";
flag += "n";
flag += "E";
flag += "!";
flag += "}";
if (buy) {
    $("#serv5").remove();
```

得到 flag

special number

$pattern = '/^(?=.*[0-9].*)(?=.*[a-zA-Z].*).{7,}$/ ';
这里正则匹配要求最少有 7 个字符，然后还要有数字和字母

```
include_once("flag.php");
if(isset($_GET['key'])){
    $pattern = '/^(?=.*[0-9].*)(?=.*[a-zA-Z].*).{7,}$/ ';
    $key = $_GET['key'];
    if(preg_match($pattern,$key)===0){
        echo "格式错误";
    }else{
        $lock="******************";
        $b = json_decode($key);
        if($b==$lock)
            echo $flag;
        else
            echo "this is no special number";
    }
}
```

这里做判断的时候，用的双等于，不会判断类型，如果是”asd”和数字 0 比，那么判断结果为 true。json_decode($key)，json_decode 可以直接解析字符串，数字，数组，对象，true，null，如果 key 是数字，那么函数返回也是数字。考虑到 json 没有其他进制表示法，用科学记数法表示 0.0000E-15，即可获得 flag

can u find me?

only robot know where is the flag

于是，想到有个 robots.txt 用来控制搜索引擎收录，访问后

```
User-agent: *
Disallow: /flaaaaaaaag.php
```

再访问这个 url，显示

only admin can get flag

修改 cookie 为 admin，再访问一次，得到 flag

tell me what you want

根据一步步的提示构造请求，使用 burpsuit 改包，即可获取 flag

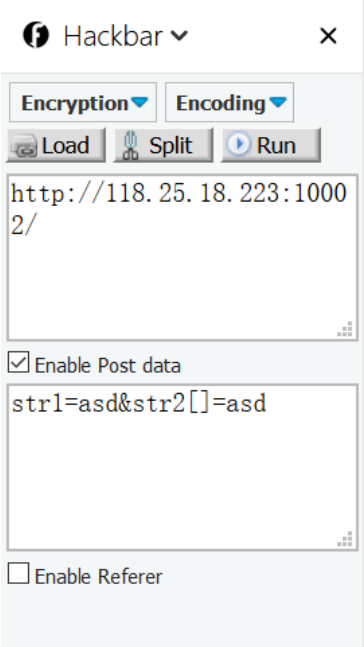
```
POST /index.php?want=flag HTTP/1.1
Host: 123.206.203.108:10001
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:59.0) Gecko/20100101 Icefox/57.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
content-type: application/x-www-form-urlencoded
cache: no-cache
origin: moz-extension://cac673b8-d81b-4b31-b35a-63bc0a2cd3f6
Content-Length: 9
Cookie: isadmin=1
Connection: close
X-Forwarded-For:127.0.0.1
referer: www.google.com

want=flag POST /index.php?want=flag HTTP/1.1
Host: 123.206.203.108:10001
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:59.0) Gecko/20100101 Icefox/57.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
content-type: application/x-www-form-urlencoded
cache: no-cache
```

origin: moz-extension://cac673b8-d81b-4b31-b35a-63bc0a2cd3f6
Content-Length: 9
Cookie: isadmin=1
Connection: close
X-Forwarded-For:127.0.0.1
referer: www.google.com

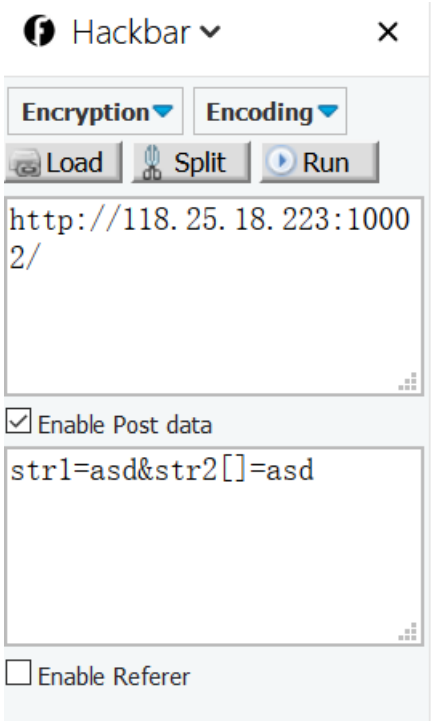
want=flag

我们不一样

==弱比较，数组和字符串用 strcmp($_POST['str1'],$_POST['str2'])
比较会判断相等，post 以下数据



flag is:hgame{g3t_f14g_is_so0000_ez}


Hackbar
Encryption
Encoding
Load
Split
Run
http://118.25.18.223:10002/
Enable Post data
str1=asd&str2[]=asd
Enable Referer