

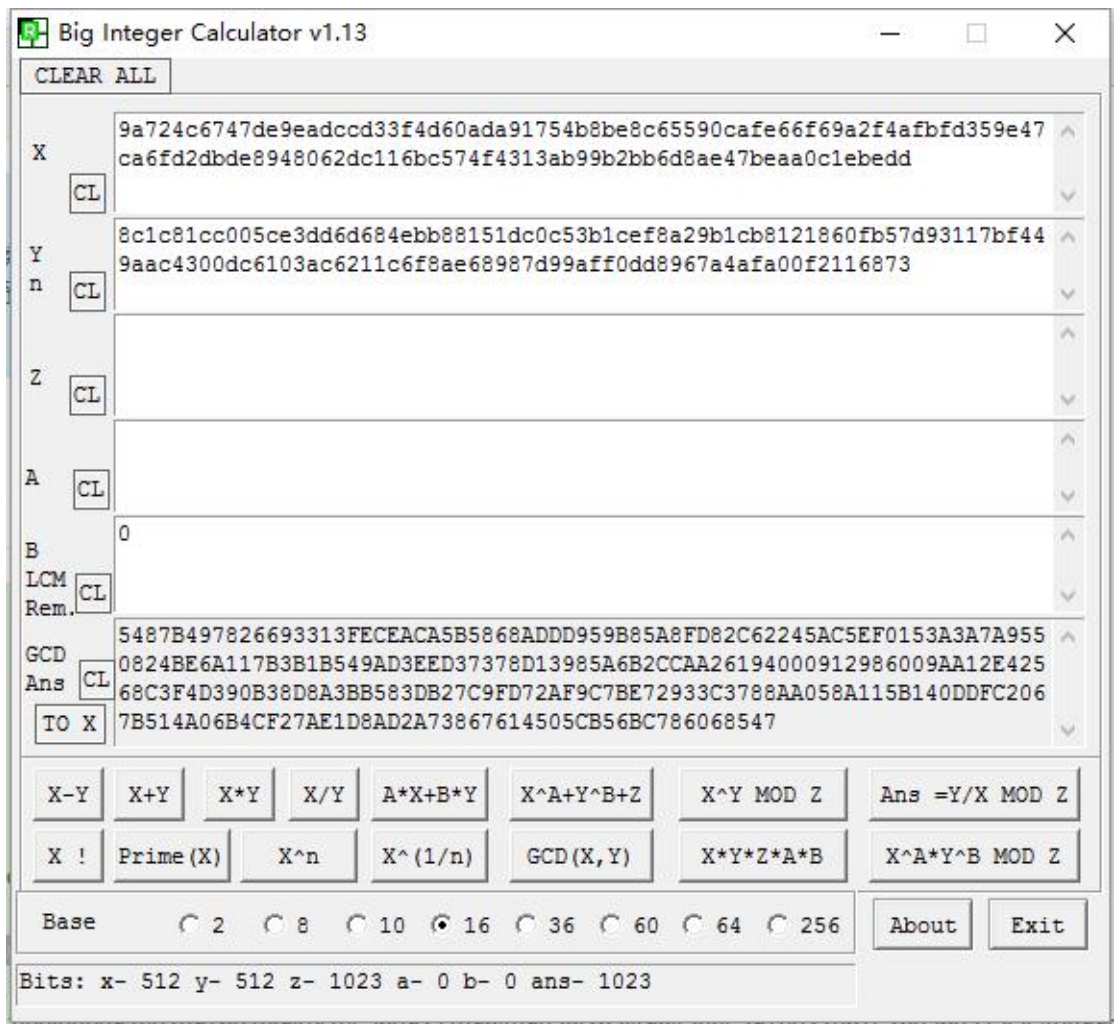
你看看，逆向多简单！

OD 载入，智能搜索一下

地址	反汇编	文本字符串
00CB1050	mov eax, dword ptr ds:[0xCC7468]	a5y!}
00CB1055	movups xmm0, dqword ptr ds:[0xCC7458]	hctf{It_ls_TOo_ea5y!}
00CB105F	mov ax, word ptr ds:[0xCC746C]	}
00CB1065	push Rei0.00CC7470	Input your flag:
00CB10C9	push Rei0.00CC7484	You Are Right!

密码学入门教室（一）

使用大数计算器，先计算模 $n=p*q$



再按照解密公式 $m=c^d \bmod n$ 计算 m

Big Integer Calculator v1.13

CLEAR ALL

X

23091e42fa7609c73f1941b320fad6d2ff6e47be588d1623f970f1fee7abd221c9834
b208f3c888902fe87ca76ec1e1363757d93c6e25c49f1c61c72b141c0b8848b54a117
427d8e30eeab89694eb5f849cafecb0e5361b9b2b0e3f89e0fdbcc66a6aad4a1a4a85

CL

Y

028b95b7e3159a851cbf537e007ae49864b7dbb93fc370a5

CL

Z

5487B497826693313FECEACA5B5868ADDD959B85A8FD82C62245AC5EF0153A3A7A955
0824BE6A117B3B1B549AD3EED37378D13985A6B2CCAA26194000912986009AA12E425
68C3F4D390B38D8A3BB583DB27C9FD72AF9C7BE72933C3788AA058A115B140DDFC206

CL

A

0

CL

B

0

CL

LCM

CL

Rem.

CL

GCD

6867616D657B7273615F31735F763372795F65347379217D

CL

Ans

CL

TO X

X-Y

X+Y

X*Y

X/Y

A*X+B*Y

X^A+Y^B+Z

X^Y MOD Z

Ans =Y/X MOD Z

X !

Prime (X)

X^n

X^(1/n)

GCD(X, Y)

X*Y*Z*A*B

X^A*Y^B MOD Z

Base

2

8

10

16

36

60

64

256

About

Exit

Bits: x- 1022 y- 186 z- 1023 a- 0 b- 0 ans- 191

再把 m 转换成字符串

6867616D657B7273615F31735F763372795F65347379217D

16进制转字符

字符转16进制


清空结果

hgame{rsa_1s_v3ry_e4sy!}

密码学入门教室（二）

简单了解凯撒加密方式后，用 c 写出简易的脚本，发现 hgame 一行，偏移量 5

```
#include <stdio.h>
#define n 36
int main(void)
{
    char ar[n];
    int i, c;
    for (i=0; i<n; i++)
        scanf("%c", &ar[i]);
    for (c=1; c<26; c++)
    {
        for (i=0; i<n; i++)
        {
            if (ar[i]+c>'z')
                ar[i]-=26;
            printf("%c", ar[i]+c);
        }
        printf("\n");
    }
    return 0;
}
```



发现 flag 大致意思为 Caesar cipher is just for fun, 此时数字 8 变成了 3，提交发现不正确。继而想到数字其他加密方式：0-a, 1-b, 2-c... 8-i。于是把 8 变成 i，更加符合 flag 意思，提交还是不对。又想到网上的人很多时候用 1 代替 i，因为长得像，所以把 i 换成 1 再提交，成功

密码学入门教室（四）

根据加密公式 $c = m^e \bmod n$ ，由于 $e=1$ ，得 $c = m \bmod n$ ，由于 n 比 c 大，所以 $m = c + k * n$ (k 为自然数)，当 $k=0$ 时， $m=c$ ，再转换成字符

6867616d657b7273615f31735f737469316c5f653473795f6e6f77217d

16进制转字符

字符转16进制

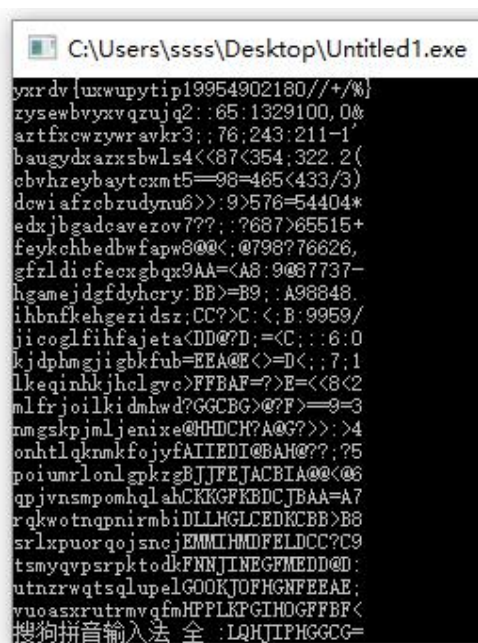
清空结果

hgame{rsa_1s_still_e4sy_now!}

密码学教室番外篇

用之前写的脚本发现了 hgame 一行，英文已解密

```
1  #include <stdio.h>
2  #define n 31
3  int main(void)
4  {
5      char ar[n];
6      int i,c;
7      for (i=0;i<n;i++)
8          scanf("%c",&ar[i]);
9      for (c=1;c<26;c++)
10     {
11         for(i=0;i<n;i++)
12         {
13             if (ar[i]+c>'z')
14                 ar[i]-=26;
15             printf("%c",ar[i]+c);
16         }
17         printf("\n");
18     }
19     return 0;
20 }
21
```



```
C:\Users\ssss\Desktop\Untitled1.exe
vxrdv{uxwupytip19954902180//+/%}
rysewbvxyvqrujq2::65:1329100,0&
artfxcwzywrvkr3;;76:243:211-1'
baugydxarxsbwls4<<87<354:322.2(
cbvhzeybaytcxmt5==98=465<433/3)
dowiafzcbzudynu6>>:9>576=54404*
edxjbgadcaverov7???:?687>65515+
feykchbedbwfapw8@@<:@798?76626,
gflzldicfecxgbqx9AA=<A8:9@87737-
hgamejdgdycry:BB>=B9::A98848.
ihbnfkehgezidsz;CC?>C:<B:9959/
jicoglfihfajeta<DD@?D;=<C::6:0
kjdpmgjigbkfub=EEA@E<>=D<;7:1
lkeqinhkjholgyo>FFBAF=?>E=<8<2
mlfrjoilkidmhwD?GGCBG>@?F>=9=3
nmgskpjmljenixa@HHDCH?A@G?>>:4
onhtlqknmkfojyfAIIEDI@BAH@???:?5
poiwmrlonl gpkzgBJJFEJACBIA@@<@6
qpjvnsmpomhqlahCKKGFKBDCJBAA=A7
rqlwotnqpnirmbidLLHGLCEDKCB>B8
srlxpuorqojsncjEMMIHMDFELDCC?C9
tsmyqvpsrpktodkFNNJINEGFMEDD@D:
utnzwqtsqlupe1GOOKJDFHGNFEEAE:
vuoaasxrutrmvgfmHPFLKPGITHOGFFBF<
搜狗拼音输入法 全 :LQHJIPHGCG=
```

计算出偏移量是 17，再处理数字，写出脚本


```
#include <stdio.h>
#define n 11
int main(void)
{
    char ar[n];
    int i;
    for (i=0;i<n;i++)
        scanf("%c",&ar[i]);
    for(i=0;i<n;i++)
    {
        while(ar[i]-17<'0')
            ar[i]+=10;
        printf("%c",ar[i]-17);
    }
    printf("\n");
    return 0;
}
```

其余字符不变，组成 flag

Explorer 的图库之一

下载之后不知道是什么格式，先拖入 winhex 看一下，发现了 flag

Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
00000000	FF	D8	FF	E0	00	10	4A	46	49	46	00	01	01	00	00	01	ÿøÿà JFIF
00000010	00	01	00	00	FF	E1	00	A6	45	78	69	66	00	00	49	49	ÿá ;Exif II
00000020	2A	00	08	68	63	74	66	7B	32	65	33	65	33	7D	00	00	* hctf{2e3e3}

Explorer 的图库之二

binwalk 先跑一下，看下里面都藏着什么

```
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)
root@OldDog:~# binwalk 1
DECIMAL      0x0      0x0      0x0      0x0
HEXADECIMAL  0x0      0x0      0x0      0x0
DESCRIPTION  JPEG image data, JFIF standard 1.01
45654        0xB256   gzip compressed data, from Unix, last modified: 2017-01-15 08:19:26
45801        0xB2E9   PNG image, 1500 x 1072, 8-bit/color RGB, non-interlaced
45842        0xB312   Zlib compressed data, default compression
```

把文件名加上.jpg，打开



没发现什么特别，接着把 gz 文件 dd 出来，再解压文件，看到 flag

```
root@OldDog:~# dd if=1 of=2.gz skip=45654 bs=1
记录了1428939+0 的读入
记录了1428939+0 的写出
1428939 bytes (1.4 MB, 1.4 MiB) copied, 12.996 s, 110 kB/s
root@OldDog:~# gunzip --stdout 2.gz
1.txt0000766000175000017500000000003113036630207012250 0ustar  lorexxarlorexxarhctf{niz
hldao_tuzh0ngm4}

gzip: 2.gz: decompression OK, trailing garbage ignored
```

Explorer 的图库之三

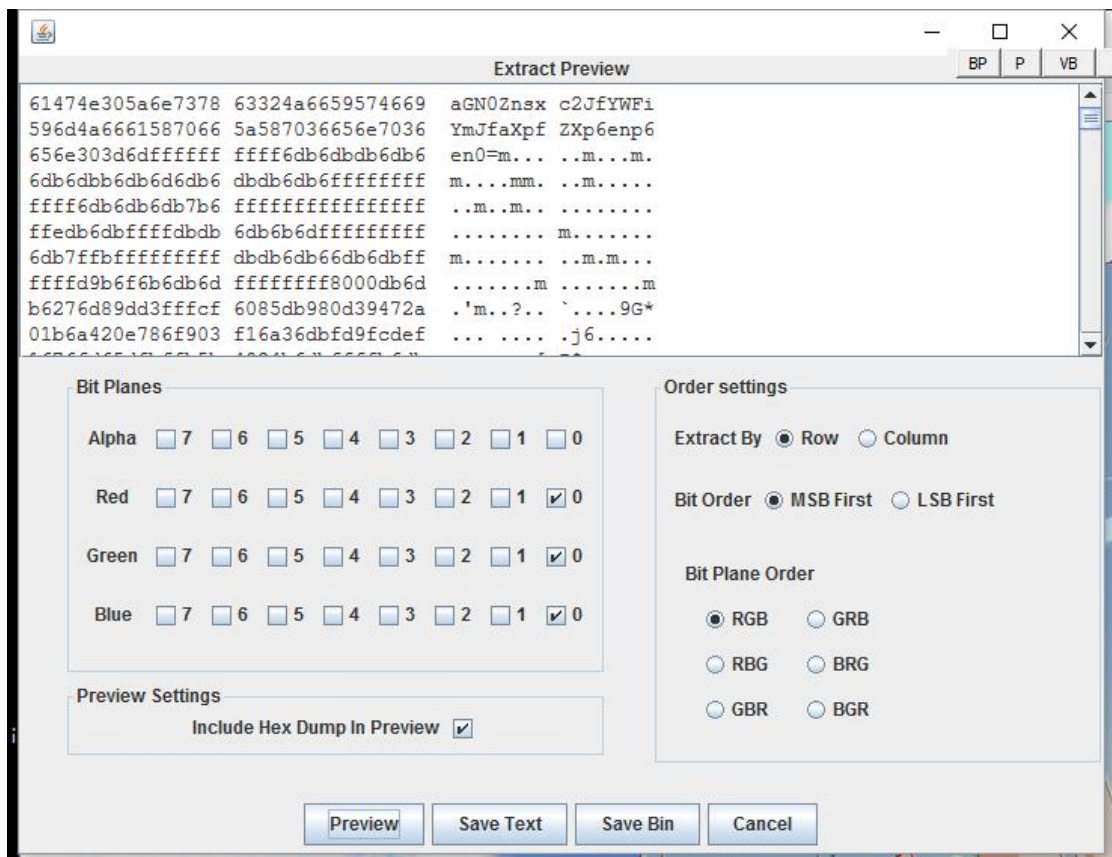
继续 dd 出 png 文件

```
root@OldDog:~# dd if=1 of=3.png skip=45801 bs=1
记录了1428792+0 的读入
记录了1428792+0 的写出
1428792 bytes (1.4 MB, 1.4 MiB) copied, 15.6899 s, 91.1 kB/s
```

打开，没什么特别



用 stegsolve 打开，看一下 lsb



由于第一段数据 aGN0Znsx c2JfYWFiYmJfaXpf ZXp6enp6 是由等号结尾且其他均有大小写英文和数字组成，符合 base64 加密特征，对其进行解密，得到 flag

aGN0Znsx c2JfYWFiYmJfaXpf ZXp6enp6en0=

BASE64加密

BASE64解密

清空结果

UTF-8▼

```
hctf{1sb_aabbbb_iz_ezzzzzzz}
```

explore 的奇怪番外 1

根据提示模仿网上以及参考源码写出 socket，拿到 flag

```
socket.py - C:\Users\ssss\Desktop\socket.py (2.7.13)
File Edit Format Run Options Window Help
import socket
sock=socket.socket(socket.AF_INET,socket.SOCK_STREAM)
sock.connect(('121.42.25.113',20000))
r=sock.recv(1024)
print r
while 1:
    r=sock.recv(1024)
    print r
    if '?' in r:
        sock.send('yes\n')
    else:
        sock.send('ready\n')
        r=sock.recv(1024)
        print r
        break
sock.close()
```

```
ready?:  
ready?:  
ready?:  
ready?:  
ready?:  
ready?:  
ready?:  
ready?:  
ready?:  
ready?:  
ready?:  
ready?:  
ready?:  
so you are ready to get flag  
now just say ready one times  
:  
htcf{pwnt0ols 1s gr3aT}
```

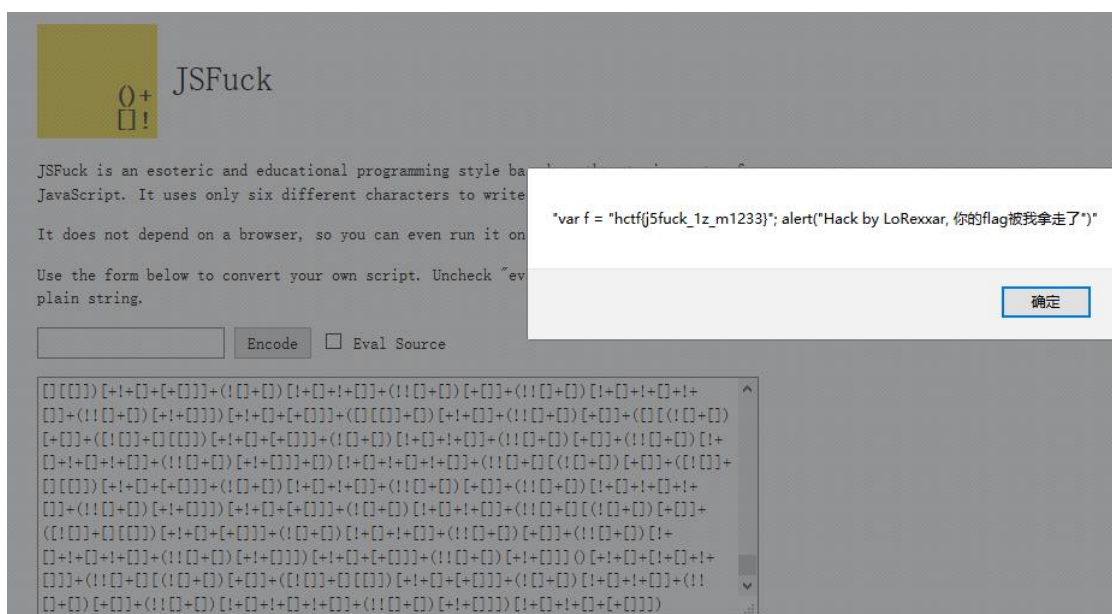
lightless 的渗透教室入门篇（一）

在 linux 下发送 `curl -d "hacker=HelloPost" 115.28.78.16:13333/pentest/01/?hacker=HelloGet`
得到 flag


```
root@OldDog: ~
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)
6/rfc2616-sec9.html">RFC2616-sec9-method - 晦涩复杂，选择阅读</a></li>
<li>扩展阅读：《HTTP权威指南》，经典必读书目，
ops曾经有提到过。</li>
<li>扩展阅读：《图解HTTP》，多了几张图，小书
可以参考看看。</li>
</p>
</div>
<p>
<h2>题目内容：</h2>
<li>向本页面同时发送GET和POST请求；</li>
<li>GET请求内容为hacker=HelloGet</li>
<li>POST请求内容为hacker=HelloPost</li>
<li>如果你不知道如何发送POST请求，方法一：学习
url命令。方法二：学习burp工具。方法三：学习Chrome/Firefox上的开发者工具或各种浏览器插件。</li>
</p>
hctf{PostAndGetIsSoEasy_comeon!}
</div>
</body>
</html>
root@OldDog:~#
```

这 TM 是啥

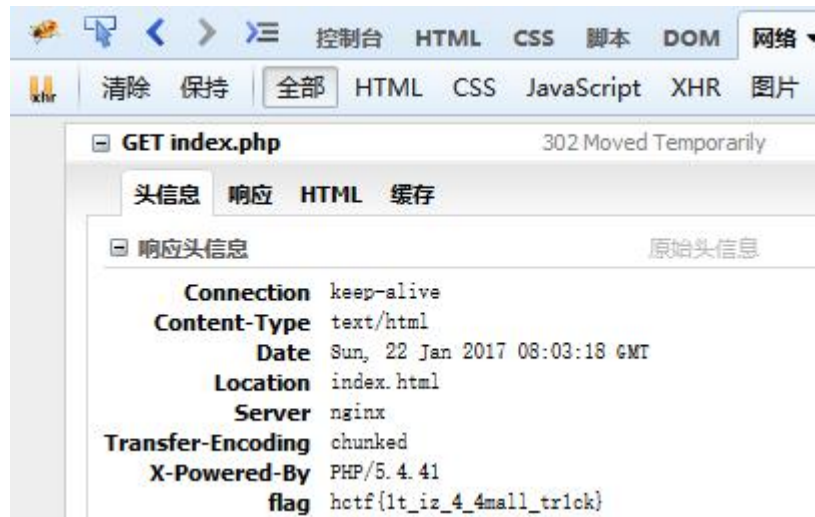
查看源代码，发现一段 jsfuck 加密的代码，找到了 jsfuck 的官网，jsfuck 的格式是 `[["filter"]["constructor"] (CODE)()`，于是把倒数第二个 `()` 中的内容拿出来放在 jsfuck 官网里当成字符串运行，得到 flag



我是谁我在哪？？？

题目里给的网址是这个 `http://115.28.78.16:13333/web/web2/index.php`，复制到浏览器打开变

成了 `115.28.78.16:13333/web/web2/index.html`，用 firebug 在响应头里找到了 flag



神奇的数字

实说实说，这道题的分数是捡来的，因为查到了一模一样的题目的 wp





掰指头算一下，这里过滤的空白字符和之前跳过的空白字符有什么区别？

少了一个"/f"，嘿嘿。

于是我们可以引入/f（也就是%0c）在数字前面，来绕过最后那个is_palindrome_number函数，而对于前面的数字判断，因为intval和is_numeric都会忽略这个字符，所以不会影响。

最后通过payload: http://f2ed13418097d206c.jie.sangebaimao.com/?number=%0c121 拿到第二个flag:



把%0c121 post 出去找到 flag（由于比赛完才写的 wp，主机好像关了页面弹不出来了，故没有截图）



问过土土 dalao 百度出来 wp 了能不能交 flag，得到肯定答复后交了 flag