

WEB.40 从 0 开始 LFI 之 0

```
src="show.php?file=.
```

+ 题目 hint 得到 flag

← → ↻ | 119.29.138.57:12000/show.php?file=../flag.php

```
hctf{Include_i5_s0_d4ngerous}
```

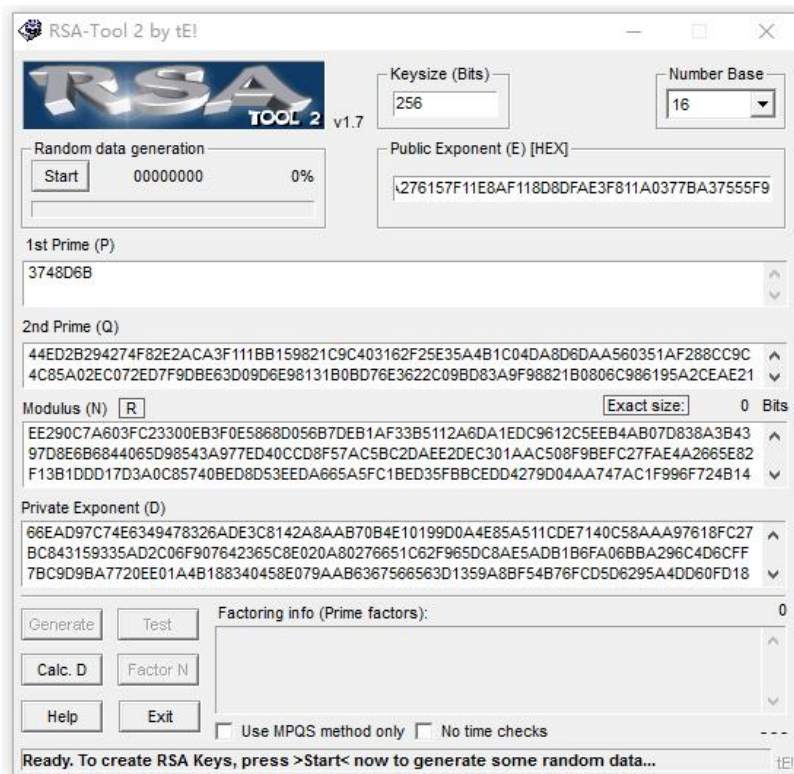
RE.48 re 从零开始的逆向之旅 : Gold Miner

.swf 文件 用 shankejingling 查看

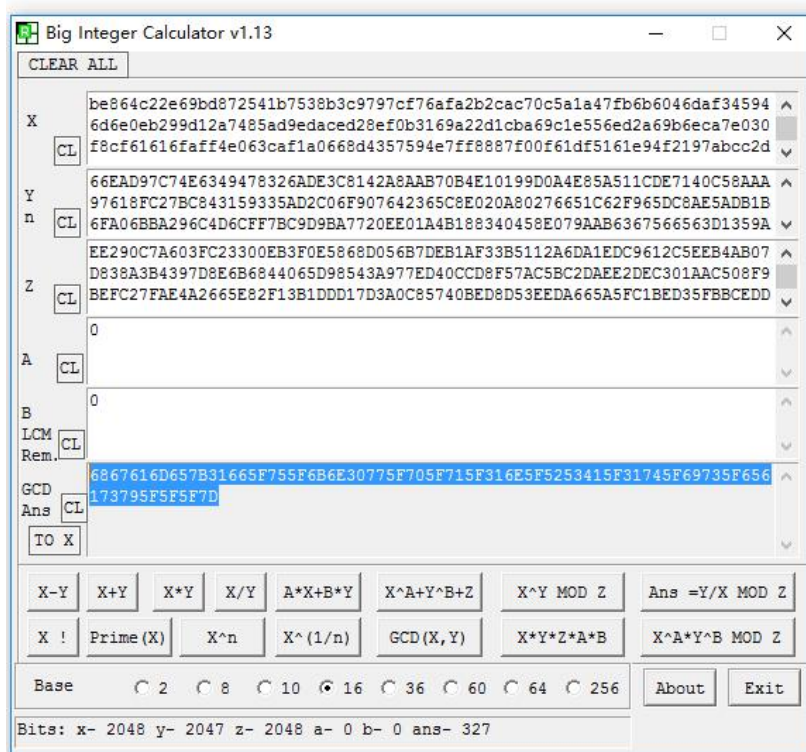
在 main 里面发现字符串

```
32 {
33     gotoAndPlay(_currentframe - 3);
34 } // end if
35
36 // [Action in Frame 10]
37 stop ();
38
39 // [Action in Frame 15]
40 goal = 375;
41 goalAddOn = goalAddOn + 275;
42 goal = goal + goalAddon;
43 goalDis = "$" + goal;
44 levelDis = 1;
45 level = 1;
46
47 // [Action in Frame 48]
48 gotoAndStop("L" + level + "_" + (random(3) + 1));
49 strength = 0;
50
51 // [Action in Frame 52]
52 if (levelDis <= 9)
53 {
54     goalAddOn = goalAddOn + 270;
55 } // end if
56 goal = goal + goalAddon;
57 if (level == 5)
58 {
59     goalDis = "$" + "hctf{Give_ME_Gold_Please}";
60 }
61 else
62 {
63     goalDis = "$" + goal;
64 } // end else if
65 levelDis = levelDis + 1;
66 level = level + 1;
67 if (total < level)
68 {
69     level = 4;
70 } // end if
71
72 // [Action in Frame 86]
73 gotoAndStop("L" + level + "_" + (random(3) + 1));
74
75 // [Action in Frame 90]
76 if (score >= goal)
77 {
```

分解得到 两个素数 p, q 再放入 RSA-Tool 计算 d



//这里智商下线，没注意进位转换导致拖了很久
得到 d 利用公式 $c^d \bmod N = m$



16 进制转字符串
得到 flag

2. 欧几里得法

参考链接:

<http://wenku.baidu.com/link?url=DVHtPgH1BiaEPwzHC7VcaC5dMjuzCdsKq8XLJgpcokvSJVW5MSmzzOkxj8u5YCdIhwYtJpJlNMxiexXp2YvSL2QQfXUuxr0P5tV10xLp-aq>

$26 == 7 * 3 + 5$ 2 1 3 个数为奇

$7 == 5 * 1 + 2$ 1 2 3 11

$5 == 2 * 2 + 1$

$2 == 1 * 0 + 1$

则 逆元 = $26 - ((2*1+1)(3)+2) == 15$

得到解密矩阵 225 -255

 -60 75

与密文 矩阵乘法

矩阵A:	矩阵B:	第1列	第2列
9 19	225 -255	885.0000	-870.0000
2 16	-60 75	-510.0000	690.0000
7 11		915.0000	-960.0000
5 16		165.0000	-75.0000
4 0		900.0000	-1020.0000
2 12		-270.0000	390.0000
13 16		1965.0000	-2115.0000
2 20		-750.0000	990.0000
21 19		3585.0000	-3930.0000
23 16		4215.0000	-4665.0000
14 18		2070.0000	-2220.0000
6 6		990.0000	-1080.0000
12 13		1920.0000	-2085.0000
19 13		3495.0000	-3870.0000
6 8		870.0000	-930.0000
16 22		2280.0000	-2430.0000

又写了个由数字转为字母的脚本

```
hill.c  dehill.c
1  #include <stdio.h>
2  int main(void){
3      int num[8];
4      char str[8];
5      int i;
6
7      printf("Encrypt: ");
8      for(i=0; i<8; i++)
9          scanf("%d",&num[i]);
10     for(i=0; i<8; i++)
11     {
12         printf("%d", num[i] % 26);
13         str[i]=num[i]+'0'+49;
14         printf("%c ",str[i]);
15     }
16
17     return 0;
18 }
```

发现爆炸...尴尬 但又不知道哪里的问题

正确的姿势: <http://www.practicalcryptography.com/ciphers/hill-cipher/>

Plaintext

haohaoxuexiandainihuiqiuniyuanma

key = 5 17 4 15

v Encrypt v

^ Decrypt ^

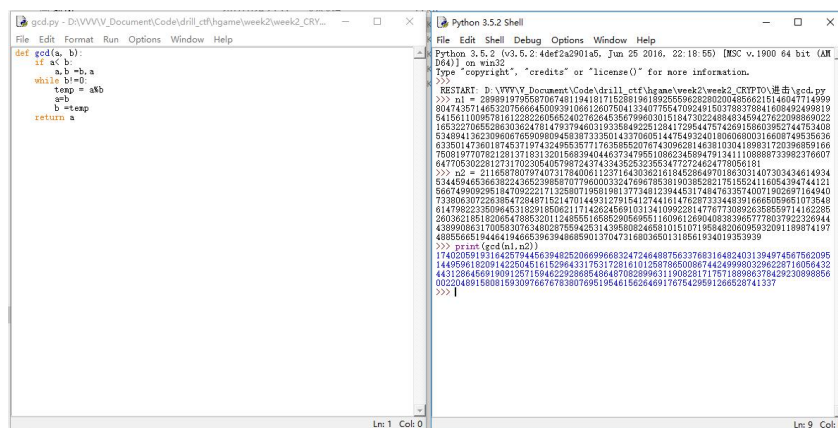
Ciphertext

lchfecncvxoqmtgqtqlqamautqsgnnlw

拿到 flag

CRYPTO.55 进击的 Crypto[0]

有很多 n, 同时 e 均为 65537, 考虑公素因子
利用 gcd 算法

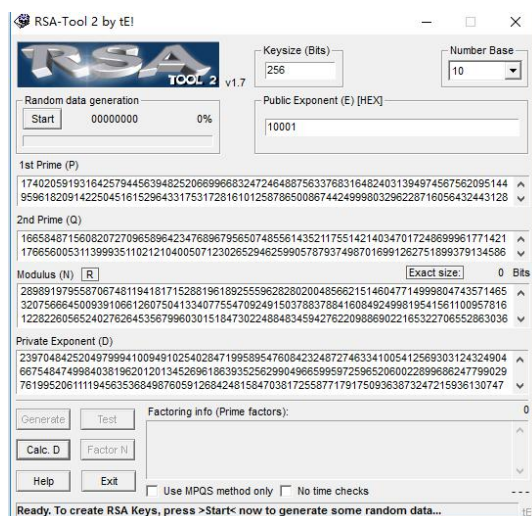


```
def gcd(a, b):
    if a < b:
        a, b = b, a
    while b != 0:
        a, b = b, a % b
    return a

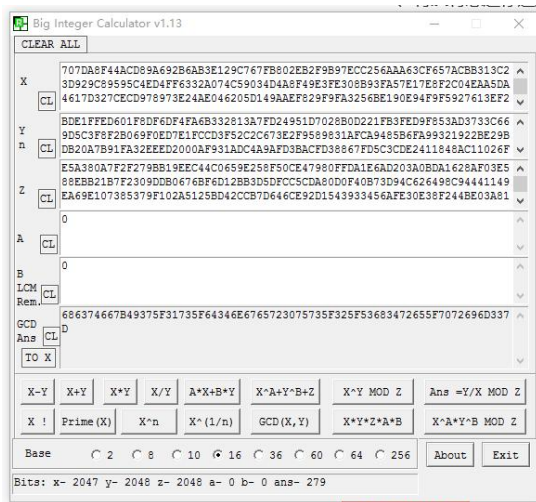
n1 = 289891979587067481194181711208196189295962828020048566215146047714999
804743571465207566450093910661200750413340775547092491503788378841606492499819
5415611100878161235220056524076584635679960310151947302248846348542762309889022
16532270655286303624781479379460319338049225128417285447574269158803952744753408
534994136230806765909090459373350143770605144754532401006060031660874953536
6336014738019745371074324955357717638852076743096281483810304189831720396859166
7081977078212813716313201588394044637247951086234689479134110886733982376607
64770530281273170230640979875474324852523534775724654778966181
>>> n2 = 21165878076740731784006112371643836216184528648701863031407303434614934
534494655683824365236857077860003247698785381003853817518204111054394144121
56674990929518470922217132580719681981377348123944531748476335740071902697164940
732086307239395472940715214701449212791541274416147830732244091666569651073540
614798223590964531829185082117426245691031341099228147767730892636859714162285
28036218618206547883201124955516895296699511609612096408383965777603792232944
43899063170853076348929759425214396588485810151771984623093962091189874197
488566519446419465396394868590137047316803650131856193401935393
>>> print(gcd(n1, n2))
1740205919316425794456394825206699683247246488756377683164824031394974567562095144
1449591820914225045161529643317531728161012587865008674424999803296228716056432443128
44312864589190912671594622928685486487082899631100828171757188986378429230898856
002204891580818930976676783807696195461562646917675429591268528741337
>>>
```

得到 p

由 $N/p == q$ 求出 q 后 利用 Rsatools 得到 d



//没看到 E 强制 hex, 果然没眼睛



和进阶五一样 16 转字符 得到 flag

PENTEST.45 我是最简单的渗透题

hint 告诉我们找万能密码 首先/index.php 发现可行 是 php 找 php 的万能密码 度娘不行谷歌去了



Questions Jobs Documentation

what's the meaning of 'admin' OR 1=1 — '

Emacs or Vim?

[Tell us](#)

The following query return all the passwords in the table tbl_user but I can not understand why this is happening.

```
SELECT password FROM tbl_users WHERE name = 'admin' OR 1=1 --'
```

Please help me to understand this part of the query: 'admin' OR 1=1 --'

Can you introduce other threats like this (website, book, etc)?

php mysql security sql-injection cracking

share improve this question

edited Jul 21 '14 at 14:25 asked Jul 19 '14 at 18:36

SilverlightFox 16.5k ● 5 ● 33 ● 70

Daniyal 20 ● 6

3 That is a classical SQL injection. It returns the users where the name is 'admin' OR where 1=1 (since 1 is indeed 1 every record will match). The -- just makes sure everything after it is seen as a comment — PeeHaa Jul 19 '14 at 18:39

You can remove 1=1 as it does nothing — Gadgetster Jul 19 '14 at 18:39

@Gadgetster what do you mean? — PeeHaa Jul 19 '14 at 18:40

2 That's the entire point @Gadgetster. It certainly does something. — PeeHaa Jul 19 '14 at 18:42

2 @Gadgetster it does quite the opposite of nothing — andrew Jul 19 '14 at 18:43

username password flag

123 321 hgame{sqli____very_interesting_233333}

成功 拿到 flag