

makerctf_wp_[已注销]

队伍ID: [已注销]

队长qq号: 591772502

参赛队员:

- 秦智扬,学号_17180120061,qq_591772502
- 李魏,学号_17180120054,qq_455774025

P.S.都是大一

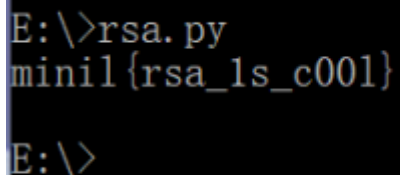
Welcome

略

Easy RSA

解题人: 李魏, 秦智扬

得到公钥模数, 网站在线工具分解一下, 得到两个大素数 p 、 q , 算出私钥, 解密。



```
E:\>rsa.py
minil{rsa_1s_c00l}
E:\>
```

Nazo

解题人: 李魏, 秦智扬

Lv3: key is where

Lv5: sos

Lv6: base64解一下, 得到qq号

Lv7: 加一下qq, 第三个验证问题

Lv8: 访问 错的是.世界

Lv9: 拉长图片

Lv10: 如果是Chrome的话, 右键点击图片, 通过Google搜索图片, 鼠标, mouse

Lv12: 这道不会, 于是直接baidu了一下“nazo.io”, 惊喜, 有个微博发了好几个题的解析

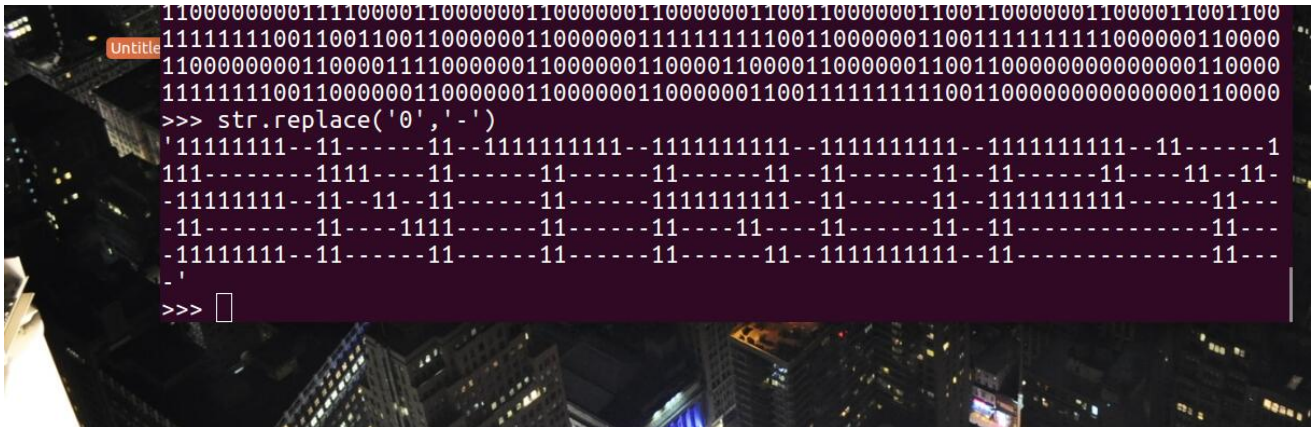
Lv13: 右键图片在新标签页中打开图片

Lv14: 同Lv12

Lv15: 用Audacity打开, 以频谱图显示, 反向, 拉宽

Lv16: 右键图片在新标签页中打开图片, F12, 把最后几行的矩形删掉

Lv17: F12, 有一些奇怪的空格, 还有全/半角之分, 但是每行等长, 猜测是字符画, 如果是FireFox的话, Ctrl+F, 半角空格, 高亮(不过一开始并没有用这种骚操作, 老老实实用python画的)



贪吃蛇

解题人: 秦智扬

下个断点调试, 吃了30个, flag就出来了



re1

解题人: 秦智扬, 李魏

提示与线代有关, ida打开, F5, 有一个29×29的矩阵, 乘上输入的列向量, 结果与一个列向量对比。那个29×29的矩阵比较特殊, 相当于单位矩阵的“0”换成“1”, “1”换成“0”, 因此可以当做一个线性方程组, 用numpy库解一下。

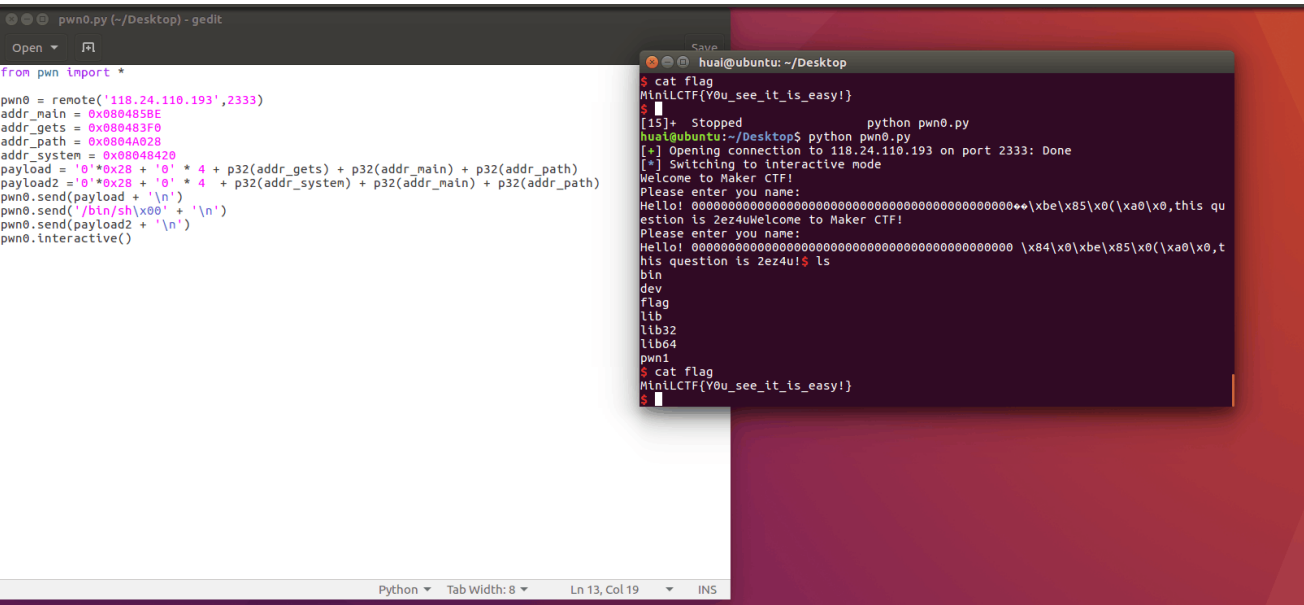
```
E:\>re.py
MiniLCTF{welcome_to_re_world}

E:\>
```

pwn0

解题人：李魏

很简单的一道栈溢出，并且没有开启PIE，直接输入/bin/sh然后调用system函数就可以了



我的世界

略

Get_flag

解题人：秦智扬，李魏

用工具逆向一下（我用的是一个叫APKtool+的东西，可以在手机上逆apk），没学过java，看了一下Encrypt.java，感觉是RSA，里面的公钥模数太大，找的网站分解不了，又去翻了翻各个文件夹，找到一个key.txt，里面是私钥。解密就好。

```
E:\>1.py
MiniLCTF{Th_is_a_mobile_flag}

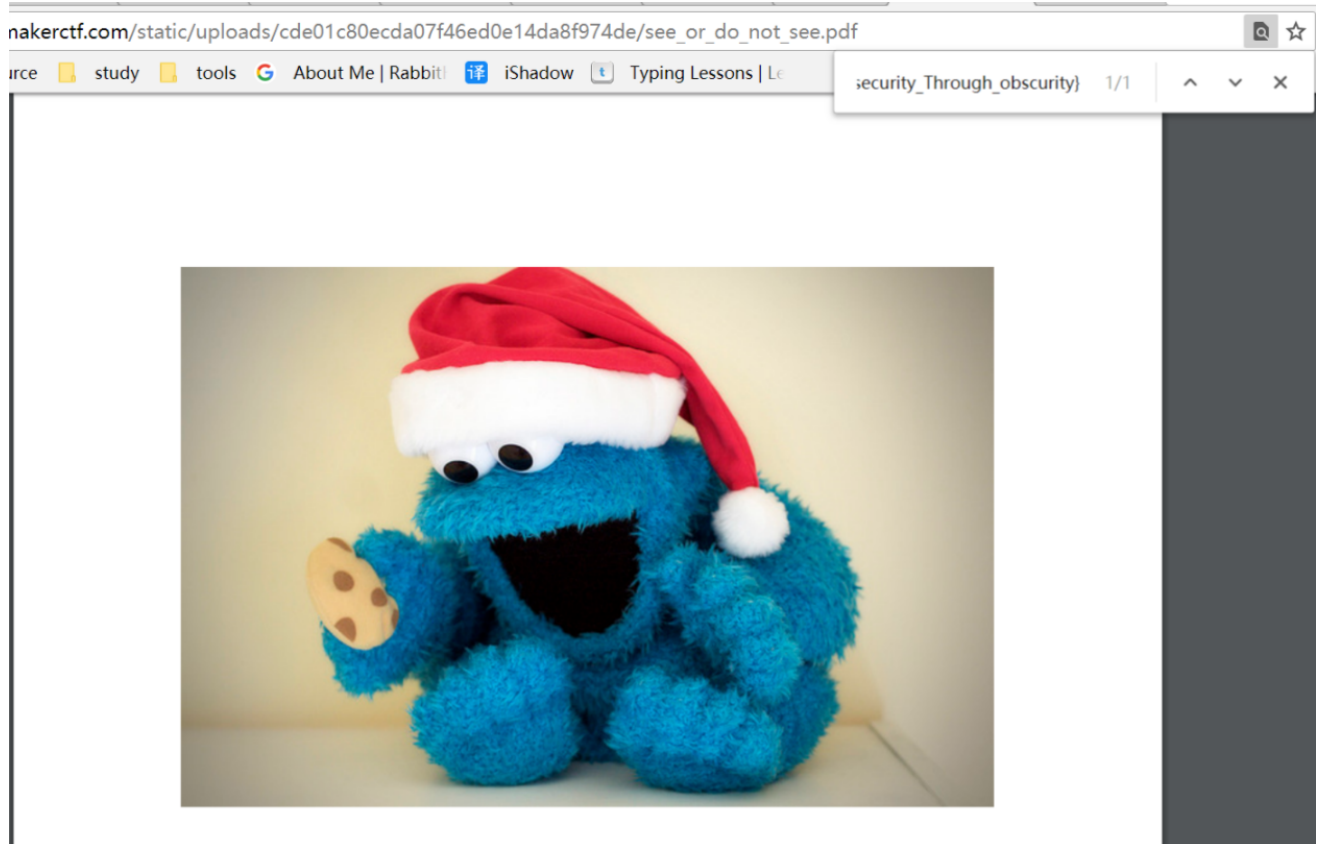
E:\>
```

see or do not see

解题人：秦智扬

手上没有什么可以打开pdf的东西，于是直接浏览器打开，Ctrl+F，手动爆破一下

(文本框太小，显示不全，就只放后半部分了)



EasyCrack

解题人：秦智扬

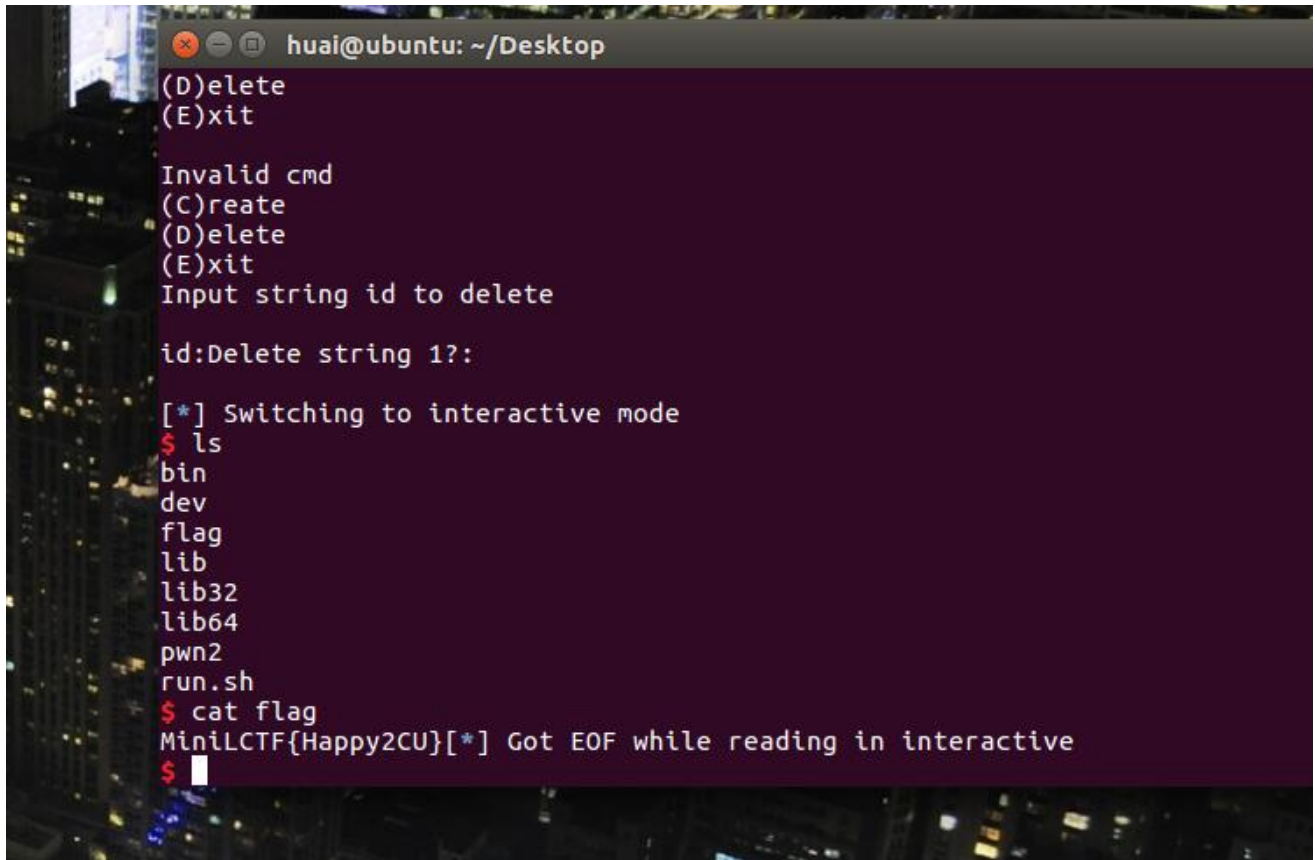
两个线程，变换两次，最后和一个数组比较。第二个线程比较简单，第一个线程比较复杂，逆一下第二个线程的算法，得到一个字符串，计做string2，当第一个线程得到的结果（计做string1）等于string2时，则成功。输入的值不同，string1不同，但输入的值部分正确，string1也部分符合string2。于是把线程一的函数复现一下（ida，F5，复制粘贴，再稍稍处理一下），暴力破解。

```
E:\>gcc easy.c
E:\>a
MiniLCTF{Base64_1s_Essential}
E:\>
```

pwn1

解题人：李魏

pwn1是一道堆利用题，大概看了UAF的利用，然后还要leak出程序加载地址和libc加载地址，在加上gadgets就可以利用ROP引导程序了。



```
huai@ubuntu: ~/Desktop
(D)delete
(E)xit

Invalid cmd
(C)reate
(D)delete
(E)xit
Input string id to delete

id>Delete string 1?:

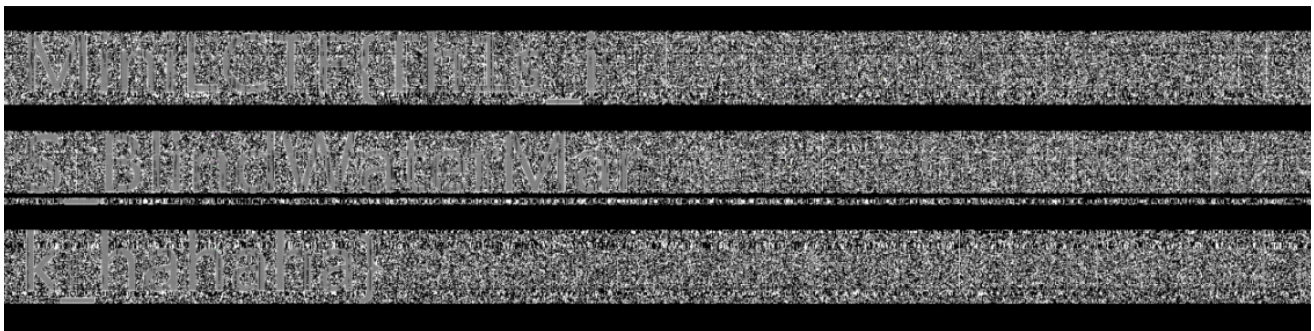
[*] Switching to interactive mode
$ ls
bin
dev
flag
lib
lib32
lib64
pwn2
run.sh
$ cat flag
MiniLCTF{Happy2CU}[*] Got EOF while reading in interactive
$
```

Moe

解题人：李魏

用WinHex看一下，发现有两个IHDR，是两张图，分离一下。

然后百度了各种方法去尝试，最后发现是盲水印。



Crypto

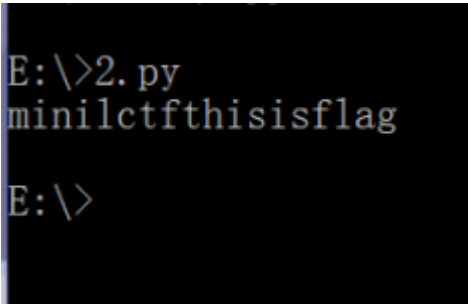
解题人：秦智扬

r是10000000以内随机数

ciphertext = 2^r%p + "=" + h^r%p*plaintext

解密原理是什么？看不懂。

根据ciphertext的前一部分爆破出r，再由后一部分算出plaint，转成36进制



pcap

解题人：秦智扬

跟踪TCP流，有用的信息：一段密文，加密了23轮，一个十六进制的pyc。处理一下得到pyc文件，反编译得到py。看看py，有三种加密方式，rot13、rot3、base64，并且rot3只对小写字母加密。每次用什么方法加密是随机的，但是会把所用加密方法的索引值放到密文前面，1：rot13，2：base64，3：rot3。那么一次次解密就好。。。

base64 解码/编码

字符编码 图片编码

请输入要进行编码或解码的字符：

TW1uaUx idGZ7MVRhQ0hsX3NvX0N1N2V9

编码

解码



解码结果以16进制显示

复制

清空

Base64编码或解码结果：

MiniLctf{1TaCH1_so_Cu7e}

调查问卷

略