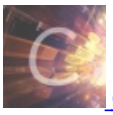
【i春秋 第二届春秋欢乐赛】crypto 密码学 rsa256

原创

Kal1 于 2020-08-16 14:32:42 发布 196 收藏 1

分类专栏： CTF刷题 杂项 文章标签： 密码学 rsa 加密解密 python 安全

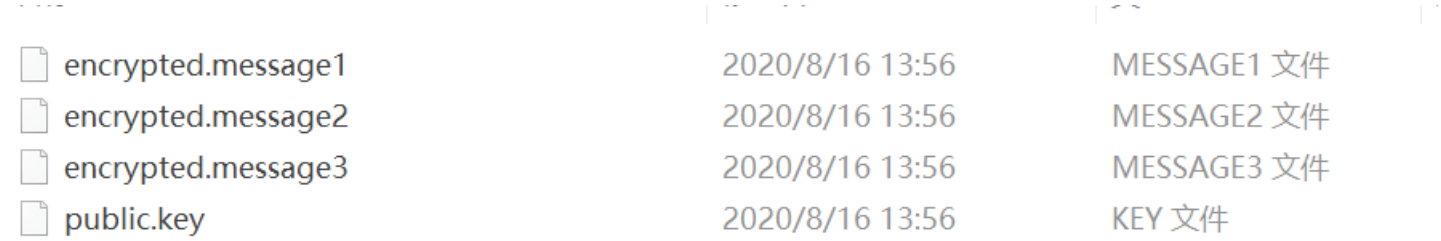
本文链接：https://blog.csdn.net/weixin_45844670/article/details/108035837



题目链接： https://www.ichunqiu.com/battalion?t=1&r=61107

下载附件，得到一个压缩包

里面有



都提示是rsa加密了，那么

```
root@kali:~/Desktop# openssl rsa -modulus -text -pubin -in public.key
RSA Public-Key: (256 bit)
Modulus:
    00:d9:9e:95:22:96:a6:d9:60:df:c2:50:4a:ba:54:
    5b:94:42:d6:0a:7b:9e:93:0a:ff:45:1c:78:ec:55:
    d5:55:eb
Exponent: 65537 (0x10001)
Modulus=D99E952296A6D960DFC2504ABA545B9442D60A7B9E930AFF451C78EC55D555EB
writing RSA key
-----BEGIN PUBLIC KEY-----
MDwwDQYJKoZIhvcNAQEBBQADKwAwKAIhANmelSKWptlg38JQSrpUW5RC1gp7npMK
/0UceOxV1VXrAgMBAAE=
-----END PUBLIC KEY-----
```


exponent是rsa中常说的的e，modulus是常说的m

下面贴一下openssl rsa 的命令

```
root@kali:~/Desktop# openssl rsa -help
Usage: rsa [options]
Valid options are:
 -help               Display this summary
 -inform format      Input format, one of DER PEM
 -outform format     Output format, one of DER PEM PVK
 -in val             Input file
 -out outfile        Output file
 -pubin              Expect a public key in input file
 -pubout             Output a public key
 -passout val        Output file pass phrase source
 -passin val         Input file pass phrase source
 -RSAPublicKey_in    Input is an RSAPublicKey
 -RSAPublicKey_out   Output is an RSAPublicKey
 -noout              Don't print key out
 -text               Print the key in text
 -modulus            Print the RSA key modulus
 -check              Verify key consistency
 -*                  Any supported cipher
 -pvk-strong         Enable 'Strong' PVK encoding level (default)
 -pvk-weak           Enable 'Weak' PVK encoding level
 -pvk-none           Don't enforce PVK encoding
 -engine val         Use engine, possibly a hardware device
```


使用python转换一下16进制

```
root@kali:~/Desktop# python
Python 2.7.18 (default, Apr 20 2020, 20:30:41)
[GCC 9.3.0] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> a=0x
KeyboardInterrupt
>>> a=0×D99E952296A6D960DFC2504ABA545B9442D60A7B9E930AFF451C78EC55D555EB
>>> print a
98432079271513130981267919056149161631892822707167177858831841699521774310891
```


用在线网址分解出p,q

Search | Sequences | Report results | Factor tables | Status | Downloads | Login

98432079271513130981267919056149161631892822707167177858831841699521774310891 Factorize! (?)

Result:

status (?)	digits	number
FF	77 (show)	9843207927...91<77> = 3028255367440967415185462127611943114 77<39> · 3250455041864363462098773013201312779 83<39>

More information

ECM

然后上脚本（python2）

```
#coding:utf-8
import gmpy
import rsa
p = 3028255367440967415185462127611943114777
q = 3250455041864363462098773013201312779837
n = 9843207927151313098126791905614916163189282270716717785883184169952177431089l
e = 65537
d = int(gmpy.invert(e , (p-1) * (q-1)))
privatekey = rsa.PrivateKey(n , e , d , p , q)      #根据已知参数，计算私钥
with open("encrypted.message1" , "rb") as f:
    print(rsa.decrypt(f.read(), privatekey).decode())       #使用私钥对密文进行解密，并打印
with open("encrypted.message2" , "rb") as f:
    print(rsa.decrypt(f.read(), privatekey).decode())       #使用私钥对密文进行解密，并打印
with open("encrypted.message3" , "rb") as f:
    print(rsa.decrypt(f.read(), privatekey).decode())       #使用私钥对密文进行解密，并打印
```

将三行合起来得到flag