攻防世界 Crypto best_rsa

原创

qtL0ng 于 2020-05-31 14:27:37 发布 1452 收藏 5

分类专栏： Crypto-RSA

本文链接：https://blog.csdn.net/jianpanliu/article/details/106454663

题目给了两个密文文件和两个公钥文件，首先解析一下公钥，命令：

```
openssl rsa -pubin -text -modulus -in publickey2.pem
```

由于对openssl rsa命令不熟，这里做一下记录：

用法：

```
openssl rsa [inform PEM|NET|DER] [-outform PEM|NET|DER] [-in filename] [-passin arg] [-out filename] [-passout arg]
[-sgckey] [-text] [-noout] [-modulus] [-check] [-pubin] [-pubout] [-engine id] [-des] [-des3] [-idea]
```

常用选项：

```
-in filename ：RSA密钥输入文件，默认读取的是私钥，若指定"-pubin"选项将表示读取公钥。
-out filename：RSA密钥输出文件
-text ：转换输入和输出的密钥文件格式为纯文本格式
-noout ：不输出任何密钥信息
-modulus ：打印公钥信息
-check ：检查RSA密钥是否完整未被修改过
-pubin ：从输入文件中读取公钥内容，公钥文件可以通过文件中的公钥标识符"-----BEGIN PUBLIC KEY-----“和”-----END PUBLIC KEY-----“来辨别。
-pubout ：从私钥中提取公钥，即从”-in filename"指定的私钥中提取公钥并输出
-des|-des3|-idea ：加密输出文件，使得每次读取输出文件时都需要提供密码。
```

成功解析出两个文件的公钥：

```
Exponent: 117 (0x75)
Modulus=67755F890795644EC27E68892B94042C78334C34F9A6D8B6AA488D9B424D64A8B9B2DCC91B1D098A09D7AC4F9A06A4B5267F88F896
8B4BAD29235D9A80330845F126B9A865F44C7A77DF72F763F553E99020745F40C8D97F0AB906154FBB1020B588F441F712B2377505B644FE36
A78743EE4995B42C7B17B8DF4782EBB595097EE1BE74143261893C4EE2C140DC469E32B17F8AB30E25F07164506B4E79C6B4E3AF5BEA026842
7FFB1134FB90A5122729C4EEF17B6D0B12CFBA4E7F14E27AA3C2B4F978E75163242EBD5CBD73829336F9A120E86E25D69CAE0229FDCCEB5B35
DC630187B0EEF1532EEC546F4037A6EAB0D0207199B9566011A52F8E9ACD7261
writing RSA key
-----BEGIN PUBLIC KEY-----
MIIBHzANBgkqhkiG9w0BAQEFAAOCAQwAMIIBBwKCAQBndV+JB5VkTsJ+aIkrlAQs
eDNMNPmm2LaqSI2bQk1kqLmy3MkbHQmKCdesT5oGpLUmf4j4lotLrSkjXZqAMwhF
8Sa5qGX0THp333L3Y/VT6ZAgdF9AyNl/CrkGFU+7ECC1iPRB9xKyN3UFtkT+NqeH
Q+5JlbQsexe430eC67WVCX7hvnQUMmGJPE7iwUDcRp4ysX+Ksw4l8HFkUGtOeca0
469b6gJoQn/7ETT7kKUSJynE7vF7bQsSz7pOfxTieqPCtPl451FjJC69XL1zgpM2
+aEg6G4l1pyuAin9zOtbNdxjAYew7vFTLuxUb0A3puqw0CBxmblWYBGlL46azXJh
AgF1
-----END PUBLIC KEY-----
```



```
Exponent: 65537 (0x10001)
Modulus=67755F890795644EC27E68892B94042C78334C34F9A6D8B6AA488D9B424D64A8B9B2DCC91B1D098A09D7AC4F9A06A4B5267F88F896
8B4BAD29235D9A80330845F126B9A865F44C7A77DF72F763F553E99020745F40C8D97F0AB906154FBB1020B588F441F712B2377505B644FE36
A78743EE4995B42C7B17B8DF4782EBB595097EE1BE74143261893C4EE2C140DC469E32B17F8AB30E25F07164506B4E79C6B4E3AF5BEA026842
7FFB1134FB90A5122729C4EEF17B6D0B12CFBA4E7F14E27AA3C2B4F978E75163242EBD5CBD73829336F9A120E86E25D69CAE0229FDCCEB5B35
DC630187B0EEF1532EEC546F4037A6EAB0D0207199B9566011A52F8E9ACD7261
writing RSA key
-----BEGIN PUBLIC KEY-----
MIIBITANBgkqhkiG9w0BAQEFAAOCAQ4AMIIBCQKCAQBndV+JB5VkTsJ+aIkrlAQs
eDNMNPmm2LaqSI2bQk1kqLmy3MkbHQmKCdesT5oGpLUmf4j4lotLrSkjXZqAMwhF
8Sa5qGX0THp333L3Y/VT6ZAgdF9AyNl/CrkGFU+7ECC1iPRB9xKyN3UFtkT+NqeH
Q+5JlbQsexe430eC67WVCX7hvnQUMmGJPE7iwUDcRp4ysX+Ksw4l8HFkUGtOeca0
469b6gJoQn/7ETT7kKUSJynE7vF7bQsSz7pOfxTieqPCtPl451FjJC69XL1zgpM2
+aEg6G4l1pyuAin9zOtbNdxjAYew7vFTLuxUb0A3puqw0CBxmblWYBGlL46azXJh
AgMBAAE=
-----END PUBLIC KEY-----
```


不难发现，两个文件的模数相同，想到同模攻击，通过脚本求得题解：

```python
from Crypto.PublicKey import RSA
import libnum
import gmpy2

c1=libnum.s2n(open('cipher1.txt','rb').read())
c2=libnum.s2n(open('cipher2.txt','rb').read())

pub1=RSA.importKey(open('publickey1.pem').read())
pub2=RSA.importKey(open('publickey2.pem').read())
n = pub1.n
e1= pub1.e
e2= pub2.e

s = gmpy2.gcdext(e1,e2)
s1 = s[1]
s2 = s[2]

if s1<0:
 s1 = -s1
 c1 = gmpy2.invert(c1, n)
elif s2<0:
 s2 = -s2
 c2 = gmpy2.invert(c2, n)

m = pow(c1,s1,n)*pow(c2,s2,n) % n
flag = libnum.n2s(m)
print(flag)
```