攻防世界 crypto-Normal_RSA

原创

faithzyA 于 2021-04-21 15:52:30 发布 117 收藏 1

分类专栏： 攻防世界WP CRYPTO

本文链接：https://blog.csdn.net/xyx2019i/article/details/115943324

1 篇文章 0 订阅

订阅专栏

1 篇文章 0 订阅

订阅专栏

攻防世界 crypto-Normal_RSA

下载附件解压打开如下：

flag.enc ：后缀enc，分析是一个通过openssl加密后生成的文件

pubkey.pem ：公钥信息文件

将解压后的附件丢进kali中，使用openssl工具

```
进入openssl
查看信息
rsa -pubin -text -modulus -in warmup -in pubkey.pem
```

```
└─$ openssl
OpenSSL> rsa -pubin -text -modulus -in warmup -in pubkey.pem
RSA Public-Key: (256 bit)
Modulus:
    00:c2:63:6a:e5:c3:d8:e4:3f:fb:97:ab:09:02:8f:
    1a:ac:6c:0b:f6:cd:3d:70:eb:ca:28:1b:ff:e9:7f:
    be:30:dd
Exponent: 65537 (0x10001)
Modulus=C2636AE5C3D8E43FFB97AB09028F1AAC6C0BF6CD3D70EBCA281BFFE97FBE30DD
writing RSA key
-----BEGIN PUBLIC KEY-----
MDwwDQYJKoZIhvcNAQEBBQADKwAwKAIhAMJjauXD2OQ/+5erCQKPGqxsC/bNPXDr
yigb/+l/vjDdAgMBAAE=
-----END PUBLIC KEY-----
OpenSSL>
```


Exponent：指的是RSA中的e

Modulus：指的是N，即pq相乘，由上图可得

e=65537

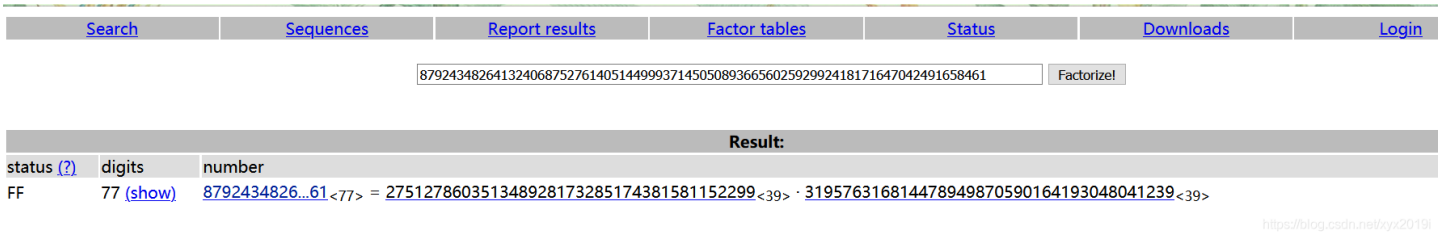
Modulus=C2636AE5C3D8E43FFB97AB09028F1AAC6C0BF6CD3D70EBCA281BFFE97FBE30DD

将获得的Modulus进行进制转换

16进制转为10进制：

87924348264132406875276140514499937145050893665602592992418171647042491658461

将10进制进行质因数分解:（在线网站http://www.factordb.com）



p=275127860351348928173285174381581152299

q=319576316814478949870590164193048041239

e=65537

使用rsatool工具生成prikey.pem私钥文件

```
安装rsatool工具：
git clone https://github.com/ius/rsatool.git
cd rsatool/  //进入这个目录
python setup.py install
安装gmpy模块：
python3 -m pip install gmpy
生成private.pem私钥文件：
python3 rsatool.py -f PEM -o prikey.pem -p 275127860351348928173285174381581152299 -q 319576316814478949870590164193048041239 -e 65537
```

```
└─$ python3 rsatool.py -f PEM -o private.pem -p 27512786035134892817328517438158
1152299 -q 31957631681447894987059016419304804123 9 -e 65537

Using (p, q) to initialise RSA instance

n =
c2636ae5c3d8e43ffb97ab09028f1aac6c0bf6cd3d70ebca281bffe97fbe30dd

e = 65537 (0x10001)

d =
1806799bd44ce649122b78b43060c786f8b77fb1593e0842da063ba0d8728bf1

p = 27512786035134892817328517438158115229 9 (0xcefbb2cf7e18a98ebedc36e3e7c3b02b)

q = 31957631681447894987059016419304804123 9 (0xf06c28e91c8922b9c236e23560c09717)

Saving PEM as private.pem
```


使用私钥解密flag

openssl rsautl -decrypt -in flag.enc -inkey prikey.pem

```
└─$ openssl rsautl -decrypt -in flag.enc -inkey prikey.pem
PCTF{256b_i5_m3dium}
```