Crypto2

by scs / ℂ

Tags: rsa-crypto

Rating:

Task: What did he said? / Decrypt RSA

You have two RSA private keys in files "recovered_1.key" + "recovered_2.key"and require to decrypt file "encrypt.txt":

```
$ cat recovered_1.key
-----BEGIN PUBLIC KEY-----
MCwwDQYJKoZIhvcNAQEBBQADGwAwGAIRAMlciLTeSYml/7kmx5RUToUCAwEAAQ==
-----END PUBLIC KEY-----
$ cat recovered_2.key
-----BEGIN PUBLIC KEY-----
MCwwDQYJKoZIhvcNAQEBBQADGwAwGAIRAOeiuMWobft9fGsyIB23Q4sCAwEAAQ==
-----END PUBLIC KEY-----
$ xxd encrypt.txt
00000000: 4802 4d24 1ab7 70c7 1123 de23 34bd cb34  H.M$..p..#.#4..4
```

Details:

https://s3.amazonaws.com/hackim18/crypto/rsa/What+did+he+said.pdf

https://s3.amazonaws.com/hackim18/crypto/rsa/What_did_he_said.zip

How to:

1) Get modulus:

$ openssl rsa -in recovered1.key -text -inform PEM -pubin

```
Public-Key: (128 bit)
Modulus:
    00:c9:5c:88:b4:de:49:89:a5:ff:b9:26:c7:94:54:
    4e:85
Exponent: 65537 (0x10001)
writing RSA key
-----BEGIN PUBLIC KEY-----
MCwwDQYJKoZIhvcNAQEBBQADGwAwGAIRAMlciLTeSYml/7kmx5RUToUCAwEAAQ==
-----END PUBLIC KEY-----
```

$ openssl rsa -in recovered2.key -text -inform PEM -pubin

```
Public-Key: (128 bit)
Modulus:
    00:e7:a2:b8:c5:a8:6d:fb:7d:7c:6b:32:20:1d:b7:
    43:8b
Exponent: 65537 (0x10001)
writing RSA key
-----BEGIN PUBLIC KEY-----
MCwwDQYJKoZIhvcNAQEBBQADGwAwGAIRAOeiuMWobft9fGsyIB23Q4sCAwEAAQ==
-----END PUBLIC KEY-----
```

2) Convert to hex:

$ python -c "print int('00c95c88b4de4989a5ffb926c794544e85',16)"

```
267655291201323217581766648921840701061
```

$ python -c "print int('00e7a2b8c5a86dfb7d7c6b32201db7438b',16)"

```
307896566740839738127153373769666872203
```

3) Looking for prime:

$ lynx --dump http://www.factordb.com
/index.php?query=267655291201323217581766648921840701061 | head | tail -n 2

```
FF 39 [10](show) [11]267655291201323217581766648921840701061[<39>] =
   [12]14673311234908966559[<20>] · [13]18240960538242393179[<20>]
```

$ lynx --dump http://www.factordb.com
/index.php?query=307896566740839738127153373769666872203 | head | tail -n 2

```
   FF 39 [10](show) [11]307896566740839738127153373769666872203[<39>] =
   [12]16879405341365159057[<20>] · [13]18240960538242393179[<20>]
```

4) Generate private keys:

$ rsatool.py -p 14673311234908966559 -q 18240960538242393179 -o 1.key

```
Using (p, q) to initialise RSA instance
n = 267655291201323217581766648921840701061 (0xc95c88b4de4989a5ffb926c794544e85)
e = 65537 (0x10001)
d = 17220326462156939542468163758601226905 3 (0x818d247361d4e4569ff75ccb4350e9fd)
p = 14673311234908966559 (0xcba212d35b7f4e9f)
q = 18240960538242393179 (0xfd24e8f6fbdb245b)
Saving PEM as 1.key
```

$ rsatool.py -p 16879405341365159057 -q 18240960538242393179 -o 2.key

```
Using (p, q) to initialise RSA instance
n = 307896566740839738127153373769666872203 (0xe7a2b8c5a86dfb7d7c6b32201db7438b)
e = 65537 (0x10001)
d = 14696931958059858593974500794703336598 5 (0x6e9142e7bebd3904c59f0edc03304de1)
```

```
p = 16879405341365159057 (0xea3fb1ba1fe6c491)
q = 18240960538242393179 (0xfd24e8f6fbdb245b)
Saving PEM as 2.key
```

5) Decrypt message:

$ openssl rsautl -decrypt -raw -inkey 1.key -in encrypt.txt

```
{binary_output_here}
```

$ openssl rsautl -decrypt -raw -inkey 2.key -in encrypt.txt

```
BabaSaidJaiJugad
```

6) Flag is

```
hackim18{'BabaSaidJaiJugad'}
```

Comments