CHALLENGE : Break The Rsa

CATEGORY : Crypto

AUTHOR : 0xsakthi

LEVEL : EASY

Break The RSA

298

Megan Foxx sent a mssg regarding her age..Did you decrypt message only using public keys?

Note: the flag is seprated by two parts

Name+age enc Second one Rsa

AUTHOR - 0xsakthi

C01-Decode-...

Flag

Submit

Given Files

```
sakthi@debian:~/TamilCTF/Crypto-Writeups/Break-the-Rsa$ mv ~/Downloads/C01-Decode-The-RSA.zip .
sakthi@debian:~/TamilCTF/Crypto-Writeups/Break-the-Rsa$ ls
C01-Decode-The-RSA.zip
sakthi@debian:~/TamilCTF/Crypto-Writeups/Break-the-Rsa$ unzip C01-Decode-The-RSA.zip
Archive:  C01-Decode-The-RSA.zip
   creating: C01-Decode-The-RSA/
  inflating: C01-Decode-The-RSA/message.enc
  inflating: C01-Decode-The-RSA/message.pub
sakthi@debian:~/TamilCTF/Crypto-Writeups/Break-the-Rsa$ ls
C01-Decode-The-RSA  C01-Decode-The-RSA.zip
sakthi@debian:~/TamilCTF/Crypto-Writeups/Break-the-Rsa$ cd C01-Decode-The-RSA/
sakthi@debian:~/TamilCTF/Crypto-Writeups/Break-the-Rsa/C01-Decode-The-RSA$ ls
message.enc  message.pub
sakthi@debian:~/TamilCTF/Crypto-Writeups/Break-the-Rsa/C01-Decode-The-RSA$
```

```
sakthi@debian:~/TamilCTF/Crypto-Writeups/Break-the-Rsa/C01-Decode-The-RSA$ cat message.enc
408,217,382,380,416,613,408,162,604,9,537,146,280

C1qKLBtrUwLkebPf+JKX6ie1bKEdUGmzkYwBJWQ=
sakthi@debian:~/TamilCTF/Crypto-Writeups/Break-the-Rsa/C01-Decode-The-RSA$ cat message.pub
Can you Decode It?(35)

-----BEGING PUBLIC KEY-----
bwDVjxOXnMR1RZGjlxe1RZGMlxb1RZG0nHesRHesRceqbgy1RZ31Rub1RZ3wSZm1RJK/OdNqOdSJOdNqRd3sSseqbge1RZ31Rub1RZ3tOdGGjLfZm+1qnHesRL1u=
-----END PUBLIC KEY-----

-----BEGIN PUBLIC KEY-----
MDgwDQYJKoZIhvcNAQEBBQADJwAwJAIdDVZLl4+dIzUElY7ti3RDcyge0UGLKfHs
+oCT2M8CAwEAAQ==
-----END PUBLIC KEY-----
sakthi@debian:~/TamilCTF/Crypto-Writeups/Break-the-Rsa/C01-Decode-The-RSA$
```

- message.enc
- message.pub

Message.pub

it's (flag) separated by two parts,
so I split it into four parts(pub+cipher(enc)) for our understanding!
like,msg1.pub,msg2.pub,msg1.enc,msg2.enc

```
sakthi@debian:~/TamilCTF/Crypto-Writeups/Break-the-Rsa/C01-Decode-The-RSA$ ls
message.enc  message.pub  msg1.enc  msg1.pub  msg2.enc  msg2.pub
sakthi@debian:~/TamilCTF/Crypto-Writeups/Break-the-Rsa/C01-Decode-The-RSA$ cat message.pub
Can you Decode It?(35)

-----BEGING PUBLIC KEY-----
bwDVjxOXnMR1RZGjlxe1RZGMlxb1RZG0nHesRHesRceqbgy1RZ31Rub1RZ3wSZm1RJK/OdNqOdSJOdNqRd3sSseqbge1RZ31Rub1RZ3tOdGGjLfZm+1qnHesRL1u=
-----END PUBLIC KEY-----

-----BEGIN PUBLIC KEY-----
MDgwDQYJKoZIhvcNAQEBBQADJwAwJAIdDVZLl4+dIzUElY7ti3RDcyge0UGLKfHs
+oCT2M8CAwEAAQ==
-----END PUBLIC KEY-----
sakthi@debian:~/TamilCTF/Crypto-Writeups/Break-the-Rsa/C01-Decode-The-RSA$ cat msg1.pub
-----BEGING PUBLIC KEY-----
bwDVjxOXnMR1RZGjlxe1RZGMlxb1RZG0nHesRHesRceqbgy1RZ31Rub1RZ3wSZm1RJK/OdNqOdSJOdNqRd3sSseqbge1RZ31Rub1RZ3tOdGGjLfZm+1qnHesRL1u=
-----END PUBLIC KEY-----
sakthi@debian:~/TamilCTF/Crypto-Writeups/Break-the-Rsa/C01-Decode-The-RSA$ cat msg2.pub
-----BEGIN PUBLIC KEY-----
MDgwDQYJKoZIhvcNAQEBBQADJwAwJAIdDVZLl4+dIzUElY7ti3RDcyge0UGLKfHs
+oCT2M8CAwEAAQ==
-----END PUBLIC KEY-----
sakthi@debian:~/TamilCTF/Crypto-Writeups/Break-the-Rsa/C01-Decode-The-RSA$
```

First part

can you decode it?(35) what is mean by 35?....

```
sakthi@debian:~/TamilCTF/Crypto-Writeups/Break-the-Rsa/C01-Decode-The-RSA$ cat message.pub
Can you Decode It?(35)

-----BEGING PUBLIC KEY-----
bwDVjxOXnMR1RZGjlxe1RZGMlxb1RZG0nHesRHesRceqbgy1RZ31Rub1RZ3wSZm1RJK/OdNqOdSJOdNqRd3sSseqbge1RZ31Rub1RZ3tOdGGjLfZm+1qnHesRL1u=
-----END PUBLIC KEY-----

-----BEGIN PUBLIC KEY-----
MDgwDQYJKoZIhvcNAQEBBQADJwAwJAIdDVZLl4+dIzUElY7ti3RDcyge0UGLKfHs
+oCT2M8CAwEAAQ==
-----END PUBLIC KEY-----
sakthi@debian:~/TamilCTF/Crypto-Writeups/Break-the-Rsa/C01-Decode-The-RSA$
```

```
Rsa keys start with BEGIN not BEGING
so first pub key its not a valid pub key
```

i confirmed this to I run OpenSSL command to extract modules in the public key it throws the error
:)

```
openssl rsa -pubin -inform PEM -text -noout -in msg1.pub
```

```
sakthi@debian:~/TamilCTF/Crypto-Writeups/Break-the-Rsa/C01-Decode-The-RSA$ openssl rsa -pubin -inform PEM -text -noout -in msg1.pub
unable to load Public Key
140681717146944:error:0909006C:PEM routines:get_name:no start line:../crypto/pem/pem_lib.c:745:Expecting: PUBLIC KEY
sakthi@debian:~/TamilCTF/Crypto-Writeups/Break-the-Rsa/C01-Decode-The-RSA$
```

lit bite osint need lol(or just read the description)
Megan folks age? 35
Name+Age
Got an encoding type
MEGAN 35
msg1.pub encrypted by Megan35

Megan35-encoding-decoding-site

MEGAN-35 Encrypt or Decrypt message
This code has been used to encrypt and decrypt ASCII text.

Type or paste in the text you want to encrypt or decrypt

bwDVjxOXnMR1RZGjlxe1RZGMlxb1RZG0nHesRHesRceqbgy1RZ31Rub1RZ3wSZm1RJK/OdNqOdSJOdNqRd3sSseqbge1RZ31Rub1RZ3tOdGGjLfZm+1qnHesRL1u=

encrypt decrypt clear

After decryption

MEGAN-35 Encrypt or Decrypt message
This code has been used to encrypt and decrypt ASCII text.

Type or paste in the text you want to encrypt or decrypt

Congrats You Got it !
n = 667
d = 1027
e = 3
decrypt itD□□

encrypt decrypt clear

so now we have n,d and e values its enough to decrypt the
simply, use RsaCtfTool.py to decrypt

msg1.enc

```
sakthi@debian:~/TamilCTF/Crypto-Writeups/Break-the-Rsa/C01-Decode-The-RSA$ cat msg1.enc
408,217,382,380,416,613,408,162,604,9,537,146,280
sakthi@debian:~/TamilCTF/Crypto-Writeups/Break-the-Rsa/C01-Decode-The-RSA$
```

408

```
sakthi@debian:~/hacking/ctf/RsaCtfTool$ python3 RsaCtfTool.py -n 667 -e 3 --uncipher 408
private argument is not set, the private key will not be displayed, even if recovered.

[*] Testing key /tmp/tmpynoi98rc.
[*] Performing fibonacci_gcd attack on /tmp/tmpynoi98rc.
  0%|
[*] Attack success with fibonacci_gcd method !

Results for /tmp/tmpynoi98rc:

Unciphered data :
HEX : 0x0054
INT (big endian) : 84
INT (little endian) : 21504
utf-8 : T
utf-16 : 䀀T
STR : b'\x00T'
```

T *is a starting letter of our flag*
just do same steps for other ciphers(217,382,380,416,613,408,162,604,9,537,146,280)

finally the first part of flag, look like this

First Part Of Flag

```
TamilCTF{y0u_
```

Second Part

msg2.pub

```
-----BEGIN PUBLIC KEY-----
MDgwDQYJKoZIhvcNAQEBBQADJwAwJAIdDVZLl4+dIzUElY7ti3RDcyge0UGLKfHs
+oCT2M8CAwEAAQ==
-----END PUBLIC KEY-----
```

in this public key, we can extract the private key then decode msg2.enc that's it(because its low-bit)

first, we need to find the modulus of this pub key

```
openssl rsa -pubin -inform PEM -text -noout -in msg2.pub
```

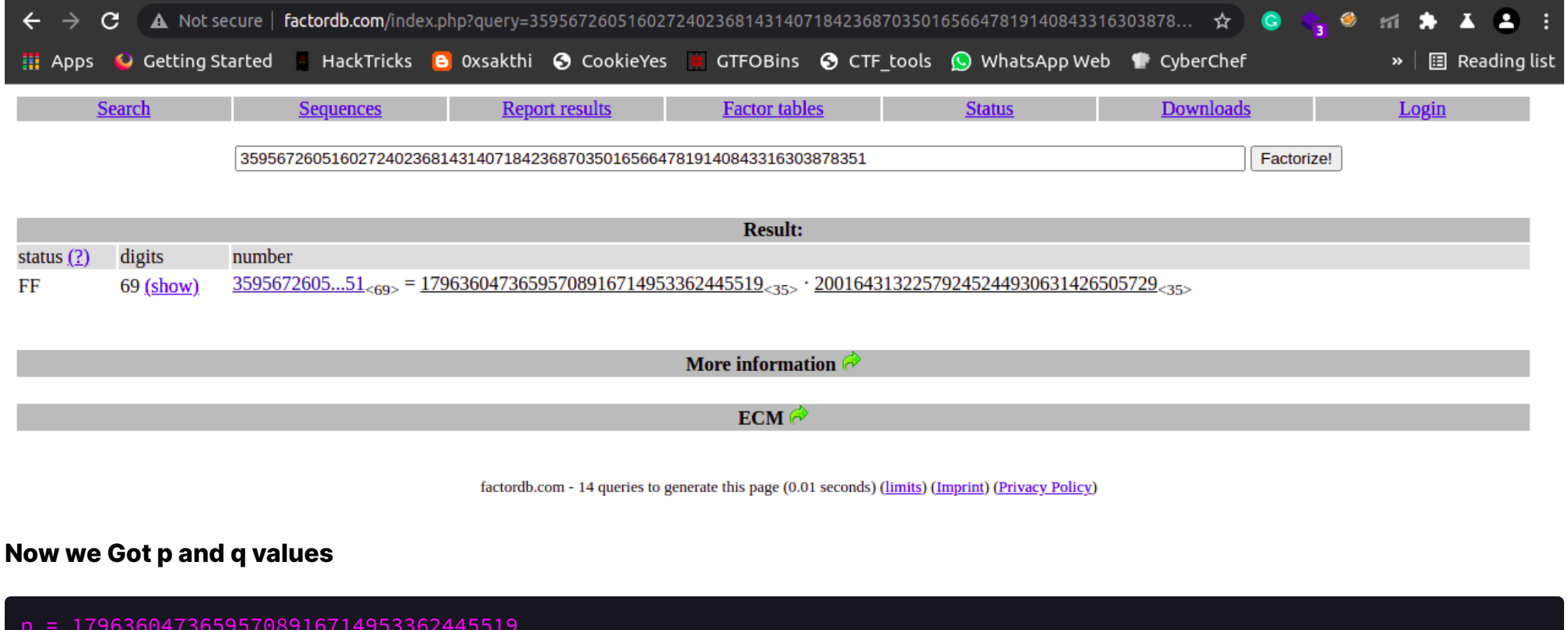
```
sakthi@debian:~/TamilCTF/Crypto-Writeups/Break-the-Rsa/C01-Decode-The-RSA$ openssl rsa -pubin -inform PEM -text -noout -in msg2.pub
RSA Public-Key: (228 bit)
Modulus:
    0d:56:4b:97:8f:9d:23:35:04:95:8e:ed:8b:74:43:
    73:28:1e:d1:41:8b:29:f1:ec:fa:80:93:d8:cf
Exponent: 65537 (0x10001)
sakthi@debian:~/TamilCTF/Crypto-Writeups/Break-the-Rsa/C01-Decode-The-RSA$
```

```
    0d:56:4b:97:8f:9d:23:35:04:95:8e:ed:8b:74:43:
    73:28:1e:d1:41:8b:29:f1:ec:fa:80:93:d8:cf

remove the :(colon)

0d564b978f9d233504958eed8b744373281ed1418b29f1ecfa8093d8cf

let's decode the hex via python3
```

```
sakthi@debian:~/TamilCTF/Crypto-Writeups/Break-the-Rsa/C01-Decode-The-RSA$ python3
Python 3.8.10 (default, Jun  2 2021, 10:49:15)
[GCC 9.4.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> 0x0d564b978f9d233504958eed8b744373281ed1418b29f1ecfa8093d8cf
359567260516027240236814314071842368703501656647819140843316303878351
>>>
```

here modulo(n)
359567260516027240236814314071842368703501656647819140843316303878351

lets factorize this using [factordb]



Now we Got p and q values

```
p = 179636047365957089167149533624455 19
q = 200164313225792452449306314265057 29
```

so now write a simple python script to decrypt the msg2.enc
msg2.enc

```
C1qKLBtrUwLkebPf+JKX6ie1bKEdUGmzkYwBJWQ=
```

flag.py

```
from libnum import invmod
from Crypto.Util.number import *
import base64

p = 179636047365957089167149533624455 19
q = 200164313225792452449306314265057 29

n = p*q

a = bytes_to_long(base64.b64decode('C1qKLBtrUwLkebPf+JKX6ie1bKEdUGmzkYwBJWQ='))

phi = (p-1)*(q-1)
e = 65537

d = invmod(e, phi)

pt = pow(a,d,n)


print(long_to_bytes(pt))
```

```
sakthi@debian:~/TamilCTF/Crypto-Writeups/Break-the-Rsa/C01-Decode-The-RSA$ python3 flag2.py
b'\x02\x90\xa9\x14\x93l\xe2\x9f\x8a?-\xa1\xf4\x01b\xbbD\xa8\x00br34k3d}\n'
sakthi@debian:~/TamilCTF/Crypto-Writeups/Break-the-Rsa/C01-Decode-The-RSA$
```

Second Part Of Flag

```
br34k3d}
```

another method for obtaining a private key in the second part is, use rsa tool(not we previously used) just pass the p and q value to get the **private key**

Once you Get private key use OpenSSL To Decrypt

```
sakthi@debian:~/TamilCTF/Crypto-Writeups/Break-the-Rsa/C01-Decode-The-RSA$ cat msg2.enc
C1qKLBtrUwLkebPf+JKX6ie1bKEdUGmzkYwBJWQ=
sakthi@debian:~/TamilCTF/Crypto-Writeups/Break-the-Rsa/C01-Decode-The-RSA$ cat msg2.enc |base64 -d > secondpart
sakthi@debian:~/TamilCTF/Crypto-Writeups/Break-the-Rsa/C01-Decode-The-RSA$ openssl rsautl -inkey secondpart.key --decrypt -in secondpart
br34k3d}
sakthi@debian:~/TamilCTF/Crypto-Writeups/Break-the-Rsa/C01-Decode-The-RSA$
```

thats it ..

Final Flag

```
TamilCTF{y0u_br34k3d}
```