Treasure Hunt with Public Key Cryptography

1 Intro – RSA

RSA is one of the widely used public key cryptosystem in real world. It's composed of three algorithms: key generation (Gen), encryption (Enc), and decryption (Dec). In RSA, the public key is a pair of integers (e,N), and the private key is an integer d.

Gen The key pair is generated by the following steps:

1. Choose two distinct big prime numbers with the same bit size, say p and q.
2. Let N=p∗q, and φ(N)=(p−1) ∗ (q−1).
3. Pick up an integer e, such that 1 < e < φ(N) and gcd(e,φ(N)) = 1.
4. Let d ≡ e^{-1} mod φ(N). (I see this as (d*e) mod φ(N) = 1)
5. Return (N,e) as public key, and d as private key.

Enc To encrypt integer *m* with public key (N,e), the cipher integer c=m^e mod N.

Dec To decrypt cipher integer c with private key d, the plain integer m=c^d mod N.

In this treasure hunt you will be aiming to decrypt two different numbers, which form into two keys which can unlock your treasure! This treasure can be redeemed on Steam! If at any time you find this to be too difficult, you can find your key in "youreapussy.txt"

2 First Half – Factors Known

The point of this exercise is to let you do a decryption given the factors. Just to give you a bit more background, this situation isn't realistic, as normally your primes numbers would be very VERY large. So large that using a factoring algorithm to find the primes would take many years to complete. You are given:

- p = 283
- q = 311
- e = 271
- d = 3871

You can use whatever way to calculate your values, but I find python to be quite easy. On your mac, open terminal and simply type python and hit enter. Now you can type in any valid python syntax and do calculations. You can find an easy reference guide online. The most important is that modulo (MOD) is the % symbol and to do x^y you do pow(x,y). Do a decryption of the Encrypted message 8866. This decrypted message is the first key of two needed for your treasure. (If while doing this python gives you a number with an ‘L’ at the end just ignore the ‘L’ and use the number part).

3 Second Half – The Weak Key Problem

There actually is a weakness in public key cryptography, and it comes when two public keys share a prime factor. When two numbers share a factor, that factor can be found very easily doing a simple greatest common denominator algorithm. This is a problem because if an attacker has the prime factors for a public key he can find the private key. For this I will be doing almost all the work for you, but if you are curious you can go through and look at the python code to see how it works.

You have two public keys available to you and you know that both share the same e = 1249:

- Target public key: 274217
- Some other public key: 318517

You need to find the common factor between these public keys so that you can get both factors of the target public key. To do this you can use a very simple python function gcd(arg1,arg2). Before you use this function, you need to type “from fractions import gcd” and press enter.

Once you have your two factors you need to find the private key *d*. This is a difficult bit to do without using an algorithm, so I have done this for you using a python script. Open a fresh Terminal window and make sure to be in the same directory as the folder I gave you. You can do this by typing “cd directory” and pressing enter (and of course fill directory with whatever the directory of the folder is). Once you get there you can type “python getd.py” and you will be asked for your factors along with e and you will get your private key! Use this as you did with the first half and decrypt your second key code from the encrypted message 194312.

4 Treasure Retrieval

Now that you have both keys, you should open a terminal window to the folder’s directory and type “python getTreasure.py” and you will be asked for your two keys. If successful you will be given your steam code. Good Job! Enjoy!