**Chi Seng 1002853**

**Challenge Description**

## The story

Caesar wants to send Rivest a message, which is broken up into 2 parts. It is encrypted with Rivest’s 2 public keys. Caesar then delivers the 2 ciphertexts to Rivest.

## Challenge Statement:

You as an attacker does a Man-in-the-Middle (MitM) attack and obtains both Caesar’s ciphertexts and Rivest’s public keys. Using the python script provided, or by coming up with your own python script, your task is to figure out the message, which is a readable text. This message is the flag.

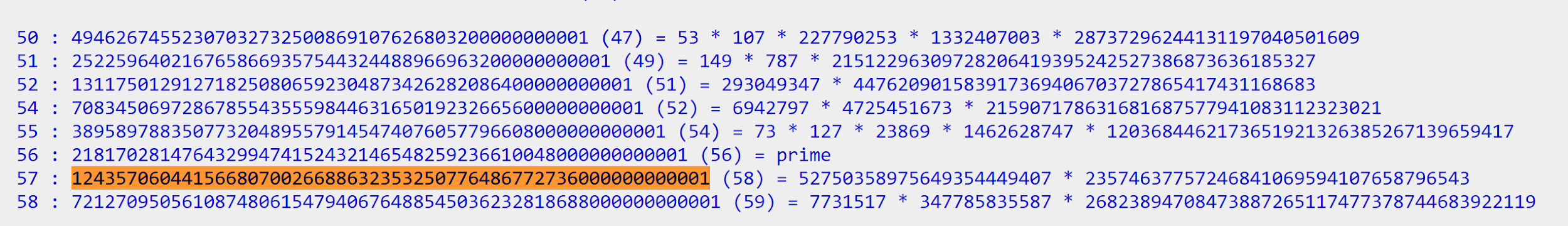
Solution

We are given two ciphertexts, both encrypted with either one of the public keys that are also given.

Retrieving n from the public key 1 shows: 131175012912718250806592304873426282086400000000001

And 2:

1243570604415668070026688632353250776486772736000000000001



Dropping the public keys into google, we can get a factorization which is:

p1: 293049347

q1: 447620901583917369406703727865417431168683

p2: 52750358975649354449407

q2: 23574637757246841069594107658796543

From this we can retrieve d through inverse modulus of phi and e.

At first I thought that ciphertext 1 will be decrypted with pubkey 1 and ciphertext 2 with pubkey2. However, hint from Derrick not to limit myself with the challenge gave me the idea to swap the keys around. This gave us a decrypted ciphertext ["6","G","9","n","F","H","G","7","R","\_","\\","U","R","%","$","#","#","[","p"]

Thinking that this might be a shift cipher, I used an online shift cipher solver which results in:

CTF{SUTD\_lib\_2100h}

From this challenge, I learned to be observant about N values of public keys and how composite keys like these can be easily factorized. Also learned how to think out of the box in solving problems.