**Faculty of Engineering**

**Computer Engineering Department**

**Security Project report**

**Presented by**

Ahmed Emad El-din

Gaser Ashraf Fayez

1. **RSA implemetation:**

Encryption:

We have p,q,e , n=p\*q, e: public key

If we have msg less than n (msg<n) then we can encrypt msg

Cipher= msg^e mod n.

If the msg is big than or equal to n we divide the msg into blocks of size n-1 then encrypt each block.

Decryption:

We have p,q,e, n=p\*q, phi(n)=(p-1)\*(q-1) e: public key

Calc private key d : d=mod inv of e in mod phi(n)

Then msg = cipher^d mod n.

1. **RSA Test Case:**

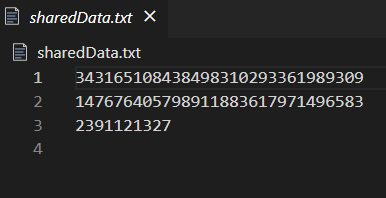
a) enter the shared data between sender and receiver in sharedData.txt

In this format

p

q

e



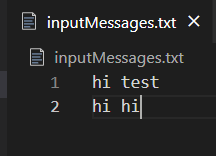
b) enter the messages you want to encrypt it and send it to receiver in inputMessages.txt

in this format

msg1

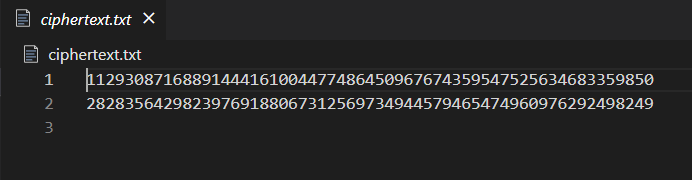
msg2

..



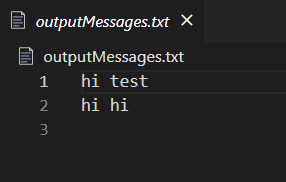
c) run this function singleTestCase in Sender.ipynb

it will generate cipher text in ciphertext.txt



d) go run Receiver.ipynb

it will decrypt the cipher text to get plaintext

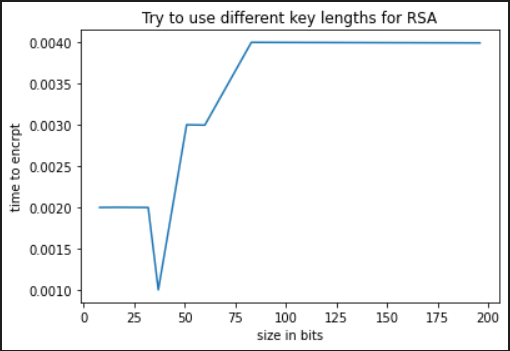


1. **RSA different key lengths:**

Go to Sender.ipynb and run this function encrptUsingDifferntNSize

In x-axis it's n in bits

In y-axis the time take to encryption using this n



1. **Brute force attack:**

We have n, public key(e)

The attack algorithm :

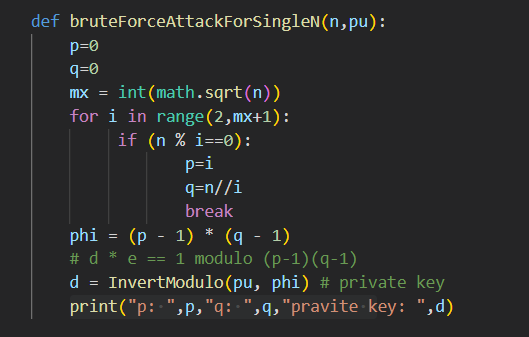
Try to factorize n to get it's primes factors p,q how?

Go throw i-> 2 to n If n is divisible by i then i is the first factor p and q=n/i

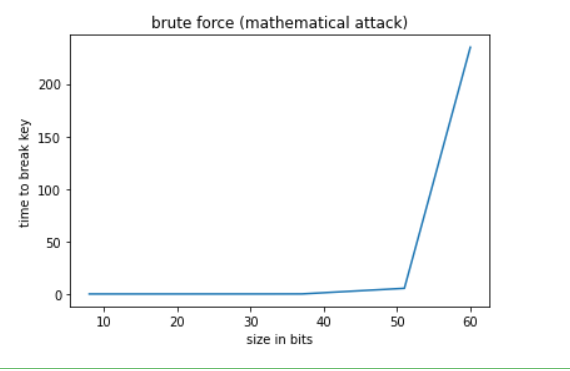
It's takes O(sqrt(n)) time and o(1) memory

Then calc phi(n)=(p-1)(q-1)

Then calc d : d = mod Inv of e in mod phi(n)

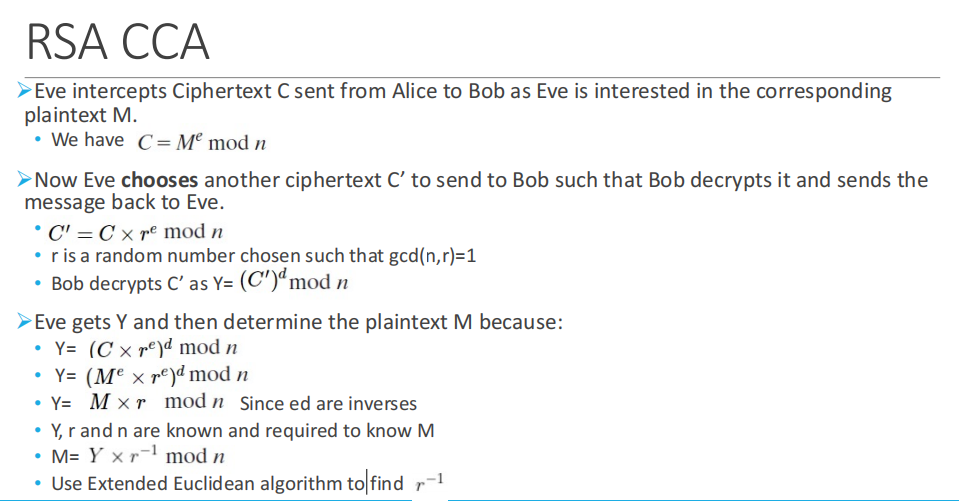


Time to break key vs key size:



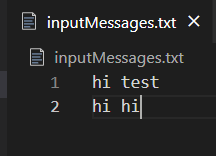
1. **Chosen Cipher Text attack:**

How it's work?



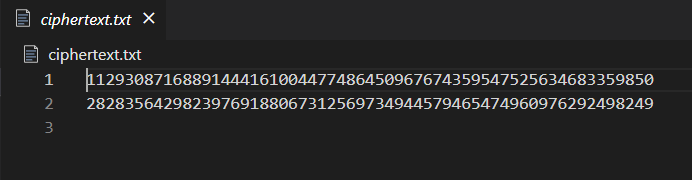
To run it:

1. Enter a messages you want to send in inputMessages.txt



b) run this function singleTestCase in Sender.ipynb

it will generate cipher text in ciphertext.txt



1. Go to CCA.ipynb and run whole file

