Abdullah Magdy 203279

Khaled Essam 202715

Farouk Ayman203389

Abdullah Hisham200507

CS (401) Project

The project is a chatting app similar to WhatsApp for sending and receiving messages.

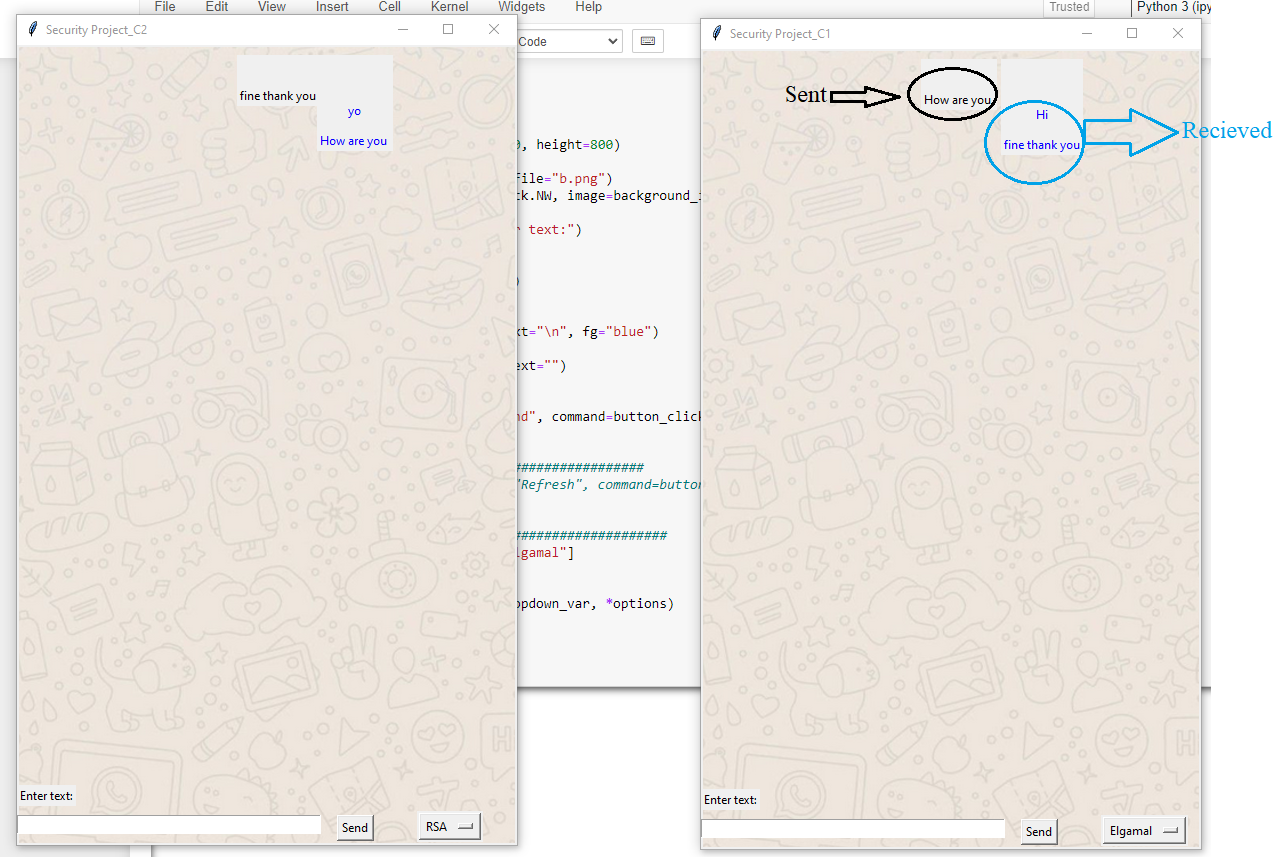
**Server**

The application uses a server as a TTP (trusted third party) and 2 clients connected to it as shown in the figure. A picture containing screenshot, design

Description automatically generated Each client is given an id in order to differentiate them from each other. Client 1 is given id 11 while Client 2 is given id 22. The server checks whether the first element from the text received is 0 which means that the client is sending a message. On the other hand if the first element is 1 that means that the client wants to receive messages.



**GUI**

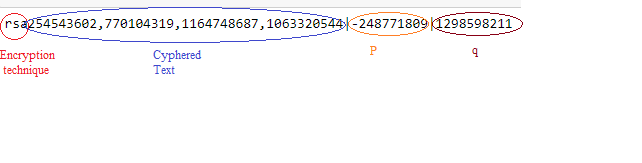


The application uses Tkinter for creating GUI. The text sent is shown in black, while text received is shown in blue. The user has the freedom to choose one of 4 encryption techniques in order to encrypt his text. A screen shot of a screen

Description automatically generated with low confidence

**Encryption Algorithms**

Each client contains encryption and decryption functions in order to communicate with each other.



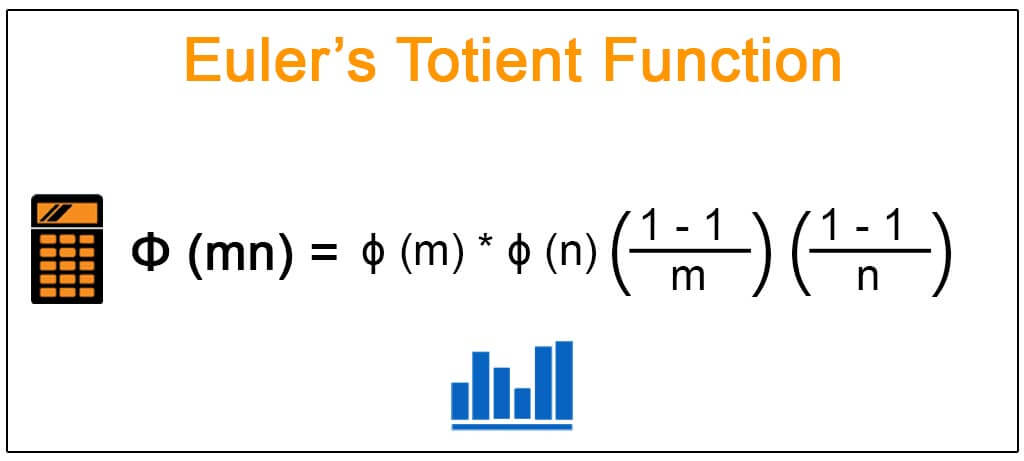
The cyphered text is sent as follows first is the encryption technique used, followed by cyphered text, after that p, and then q. and all of them are concatenated by |.

**RSA**

A picture containing text, screenshot, diagram, line

Description automatically generated

Encryption:

In the RSA algorithm, the totient function is used to select two large prime numbers, which are then multiplied together to create a modulus. The totient of the modulus is then calculated and used as part of the encryption and decryption process. The security of the RSA algorithm relies on the fact that it is difficult to factor large numbers into their prime factors, making it difficult for an attacker to calculate the totient function and break the encryption.

The first step is generating random p and q numbers. Then checking if they are both prime if not, we choose another number.

Ꝋn is calculated by multiplying (p-1)\*(q-1)

N is calculated by multiplying p\*q

E is calculated by choosing an integer between (1, Ꝋn-1) and then checking gcd with Ꝋn to ensure that e and Ꝋn are coprime.

The text is then converted to ascii to encrypt it.

Each character is encrypted using a^b mod m

Decryption:

The text is decrypted using the equation a^b mod m

**DES**

A picture containing screenshot, text, diagram, design

Description automatically generated

Des Contain on two stages. First stage responsible on Key to do permutation choice for keys to create 16 key. Second stage responsible on plaintext for encrypt plain text using initial permutation, EP, xor, S-Box , swap and ipInverse Permutation choice 1 turn into 64 bit to 56 bit and Permutation choice 2 turn into 56 bit to 48 bit.

A picture containing text, screenshot, font

Description automatically generated

A screenshot of a computer program

Description automatically generated with low confidence

A screen shot of a computer program

Description automatically generated with low confidence

A picture containing text, screenshot, software

Description automatically generated

we should do padding if plaintext is less than 64 bit. if plaintext more than 64 bit , we should take first 64 bit and take the rest bits in next steps

A screen shot of a computer program

Description automatically generated with low confidence

A screen shot of a computer program

Description automatically generated with low confidence A screen shot of a computer screen

Description automatically generated with low confidence A picture containing text, screenshot, font

Description automatically generated

**AES**

-AES, short for Simplified Advanced Encryption Standard, is a type of encryption algorithm that is designed to be simpler and use fewer bits than the Advanced Encryption Standard (AES). To encrypt data using S-AES, 16 binary numbers are input along with a generated key, which is used to calculate K0, K1, and K2. K0 is the initial key received, and w2 is obtained by XORing w0 with '10000000' and SubNib(RotNib(W1)), where RotNib and SubNib are functions that swap and substitute 4 binary numbers, respectively. W3 is then calculated by XORing W1 and W2, and W4 is calculated by XORing w2 with 00110000 and SubNib(RotNib(W3)). Finally, W5 is obtained by XORing W3 and W4.

To calculate K0, K1, and K2, the resulting W0, W1, W2, W3, W4, and W5 are concatenated together in pairs. Specifically, K0 is obtained by concatenating W0 and W1, K1 is obtained by concatenating W2 and W3, and K2 is obtained by concatenating W4 and W5. It is important to note that the '+' sign here is not an addition operator, but simply indicates that the numbers are concatenated together.

The S-AES encryption process involves multiple rounds. In each round, K0 is XORed with the plaintext received, which is then subnibed and shift rowed. In the shift row step, the 2nd four binary number is switched with the 4th four binary number. The result is then added to mix column, where mix column has its own multiplication and addition matrix. After completing the first round, the process is repeated for the subsequent rounds, except without the mix column step.

In decryption firstly Input the ciphertext and the generated key. Then Calculate K0, K1, and K2 using the same process as encryption. Then Obtain W0 and W1 by concatenating the first 8 binary digits of K0, and W2 and W3 by concatenating the first 8 binary digits of K1, and W4 and W5 by concatenating the first 8 binary digits of K2. Then Calculate W3 by XORing the ciphertext with W2. Then Calculate W4 by XORing the result from step 4 with '00110000' XOR SubNib(RotNib(W3)). Then Calculate W5 by XORing the result from step 4 with W4. Then Calculate W1 by XORing the result from step 5 with '10000000' XOR SubNib(RotNib(W0)). Then Calculate W0 by XORing the result from step 5 with W1. Then Obtain the plaintext by concatenating W0, W1, W2, W3, W4, and W5 in that order. Then going backward of encryption to decrypt but just the mix column matrix of 1 4 4 1 will be changed to 9 2 2 9

It's important to note that the decryption process is essentially the reverse of the encryption process. The only difference is that the order in which the calculations are performed is reversed, and the inverse of the substitution and mixing functions is used.

A picture containing text, screenshot, rectangle, number

Description automatically generated

A picture containing text, screenshot, line

Description automatically generatedA screenshot of a computer

Description automatically generated with low confidence

**ELGAMAL**

El-Gamal encryption is a public-key cryptosystem that was first proposed by Taher El-Gamal in 1985. It is based on the Diffie-Hellman key exchange algorithm, which allows two parties to establish a shared secret key over an insecure communication channel. In El-Gamal encryption, the sender generates a random session key and encrypts the message with it using the recipient's public key. The recipient can then decrypt the message using their private key. El-Gamal encryption is secure against attacks from classical computers, but it can be vulnerable to attacks from quantum computers.

A picture containing text, screenshot, line, font

Description automatically generated

A screenshot of a computer

Description automatically generated

Function generating the prng key using the random value q. while making sure that the GCD in the correct range.

A picture containing text, screenshot, software, design

Description automatically generated

The 2 functions representing the encrypt and decrypts. Creating encryption list representing each letter in the plain text in its corresponding in ascii number then calculating the cipher text using plaintext and the private key. Then the results are returned with its corresponding public key to be shared.

A screen shot of a computer program

Description automatically generated with low confidence

Example for the run and the code calls. First we generate the q and g randomly to be able create the session and exchange the corresponding public key for description then we generate the main private key for encryption.

A screen shot of a computer

Description automatically generated with medium confidence