Q1

Document to describe my implementation for Question 1

Overview

This implementation demonstrates a secure communication mechanism using a combination of RSA and Salsa20 encryption algorithms. RSA is used for securely transmitting a symmetric key, which is then utilized by Salsa20 for encrypting and decrypting messages.

Components

1. **RSA Key Generation**: Bob generates a public-private key pair using two random prime numbers.
2. **Symmetric Key Generation**: Alice generates a 16-byte symmetric key to be used with Salsa20.
3. **RSA Encryption & Decryption**: Alice encrypts the symmetric key using Bob's public key, and Bob decrypts it using his private key.
4. **Salsa20 Encryption & Decryption**: Bob encrypts a message using the symmetric key and transmits it to Alice, who then decrypts it.

Implementation Steps

1. Generating RSA Keys

Bob generates two large prime numbers (`p` and `q`), computes `n = p * q` , and derives the public (`e`) and private (`d`) keys using the modular inverse operation.

2. Generating a Symmetric Key

Alice generates a 16-byte random key to be used for Salsa20 encryption.

3. Encrypting the Symmetric Key with RSA

Alice converts the symmetric key to an integer and encrypts it using Bob's public key with modular exponentiation.

4. Decrypting the Symmetric Key with RSA

Bob receives the encrypted symmetric key and decrypts it using his private key, restoring the original key.

5. Encrypting a Message with Salsa20

Bob encrypts a message using Salsa20 with the decrypted symmetric key. The nonce generated during encryption is stored along with the ciphertext.

6. Decrypting the Message with Salsa20

Alice decrypts the ciphertext using the same symmetric key and the transmitted nonce, recovering the original message.

Challenge Faced

If the value of p and q is too small i.e. less than 128 digits, the encryption will not be done succesfully for RSA. Because the length of the key will be more than the length of the encrypted key.

Conclusion

This implementation provides a secure way to transmit messages by combining RSA's secure key exchange with Salsa20's fast encryption. The approach ensures both security and efficiency in encrypted communications.