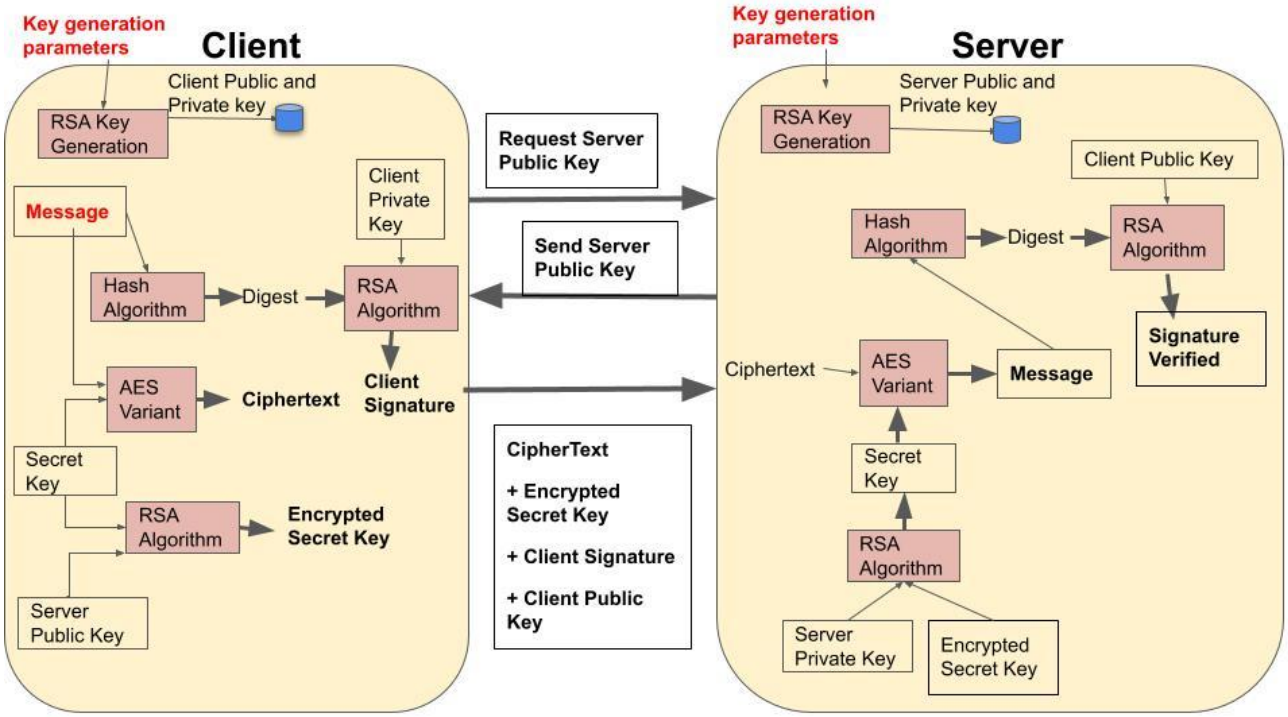
# CS3006- Network Security & Cryptography Assignment

**Submission due :- 18th October, 2021**

Create a client and server application to provide confidentiality, authentication and integrity using the concepts discussed in the course. To encrypt a message the client is using a simplified AES variant with 2 rounds. The client then sends ciphertext and key to the server. Server on receiving the encrypted message decrypts the message. The process of encryption is explained in the attached AES variant file(AES_Variant). Two examples of this AES variant are given in attached files (Example_1 and Example_2).

Implement the RSA algorithm from scratch that will be used for secret key encryption and digital signature. For Message digest creation you may use any existing implementation of hash algorithm compatible with your program. Working of the secure system will be as follows:




- **Client inputs:** Message, parameter for key generation (p, q, e) at Client end
- **Server Inputs:** parameter for key generation (p, q, e) at Server end.
- **Message Flow:**
  - Client requests for public key of server.
  - Server sends the public key.

    - Client sends Ciphertext, Encrypted secret key, Client Signature, Client public key.
- **Client side computation**:
    - Create Client signature through RSA algorithm, taking Digest from Hash algorithm and client private key as input.
    - Create Ciphertext through the AES variant, taking Message and Secret key as input.
    - Encrypt Secret key with RSA algorithm, taking Secret key and Server Public key as input.
- **Server side Computation:**
    - Decrypt Secret key using RSA algorithm
    - Decrypt ciphertext using AES variant
    - Create message digest
    - Verify Client Signature

Format for output is as follows:

## Client Side:

**Input:**

Message: <>
Secret Key: <>
Public Key parameters: <p,q,e>

**Output:**

Encrypted Secret Key: <>
Cipher text intermediate computation process:
After Pre-round transformation:
Round key K0:
After Round 1 Substitute nibbles:
After Round 1 Shift rows:
After Round 1 Mix columns:
After Round 1 Add round key:
Round key K1:
After Round 2 Substitute nibbles:
After Round 2 Shift rows:
After Round 2 Add round key:
Round Key K2:
Cipher text: <>
Digest: <>
Digital Signature: <>

## Server Side:

**Input:** Public Key parameters: <p,q,e>
**Output:**

Decrypted Secret key: <>
Decryption Intermediate process:
- After Pre-round transformation:
- Round key K2:
- After Round 1 InvShift rows::
- After Round 1 InvSubstitute nibbles:
- After Round 1 InvAdd round key:
- Round key K1:
- After Round 1 InvMix columns:
- After Round 2 InvShift rows:
- After Round 2 InvSubstitute nibbles
- After Round 2 Add round key:
- Round Key K0:

Decrypted Plaintext: <>
Message Digest: <>
Intermediate verification code: <>
Signature verified/ Signature Not Verified

Strictly follow these instructions for preparing the assignment:

1. Include comments in the code to explain the processes. It also means the author's name and roll number must be included in each code file.
2. Create a Readme file explaining all the functions and files being used.
3. Submit a zip file containing all the files for execution, as well as a readme file.
4. Include a sample output file (screenshot of the execution window) for client encryption and server decryption as well. The output should also reflect your name or roll number in it.
5. Rename the zip file to your roll number and name before uploading.