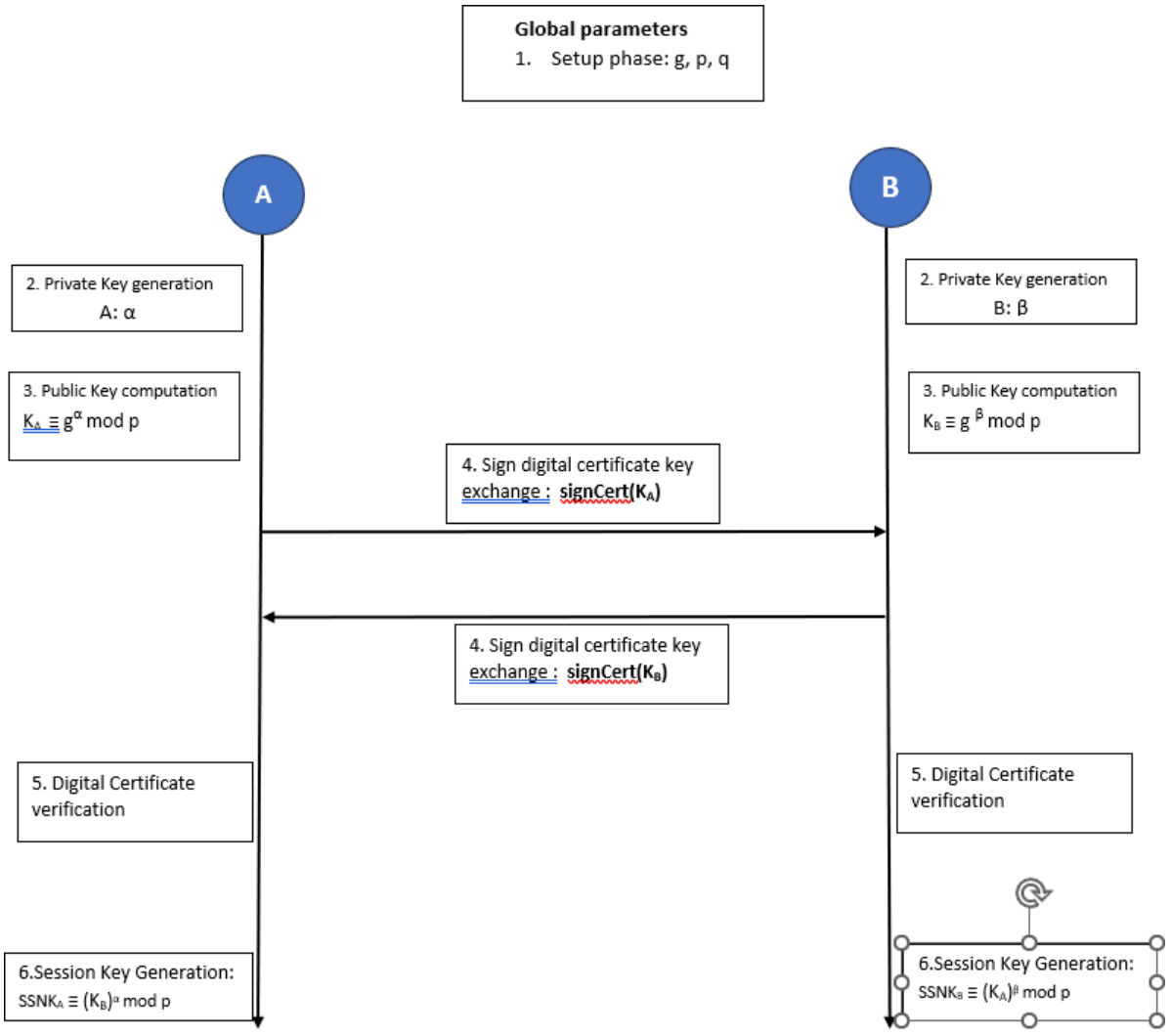
Cyber Security

Lab Assignment- 6

Objective: Implement the Fixed Diffie-Hellman key exchange protocol to securely establish a shared secret session key between two parties over an insecure communication channel.

Note: Use only the Integer class do not use any other libraries function from Crypto++.



Process phase implementation.

1. Setup Phase
2. Private Key Generation Phase
3. Public Key Generation Phase
4. Certificate Generation Phase
5. Certificate Verification Phase
6. Session Key Generation and Verification Using md5sum.

1. **Setup**
   Objective:
   1. Generate the large prime number p.
   2. Generate a generator 'g' of a subgroup of Z_P^* whose order must be another large prime number q.
   3. **Note** that the sizes of 'p' and 'q' must be given as input as command-line argument (for example 1024 & 160)
   4. Store the generated parameters **(g, p, q)** in the binary file named **'params.bin'**

   Note:
   1. $Z_q^* \subset Z_p^*$.
   2. $g \in Z_p^*$.

   Workflow:

2. **Private Key Generation:**
   Objective: Generate the private keys α and β for Alice and Bob respectively .

   Note: $\alpha, \beta \in Z_q^*$

Implementation Process:

1. Read the global parameters (g,p,q) from the binary file **'params.bin'** .
2. Generate the private keys α and β using only Integer Class function do not use any other Crypto++ library functions.
3. Save the private keys in the separate binary files named '**privatekeyA.bin**' and '**privatekeyB.bin**' , respectively.

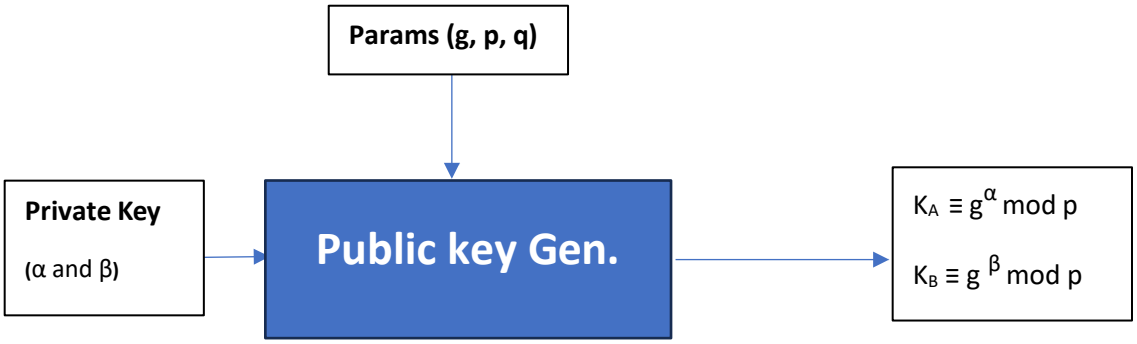
Workflow:

3. Public Key Generation Phase:

Objective :

1. Generate the public keys K_A and K_B for parties A and B, respectively.
2. Execute the programs twice for both the parties and save their computed keys in separate binary files named '**publicKeyA.bin**' and '**publicKeyB.bin**' , respectively.

Workflow:



Note: Use only Integer Class functions for (* mod p) operations for key calculations.

4. **Certificate Generation Phase:**

   Objective: Share the Diffie-Hellman public key K_A and K_B by signing with their certificate. (**Refer to assignment 5**).

5. **Certificate Verification Phase:**

   Objective: Verify the Diffie-Hellman public key certificate to ensure the authenticity of the parties, respectively. **(Refer to the assignment 5).**

6. **Session Key Generation and Verification Using md5sum.**

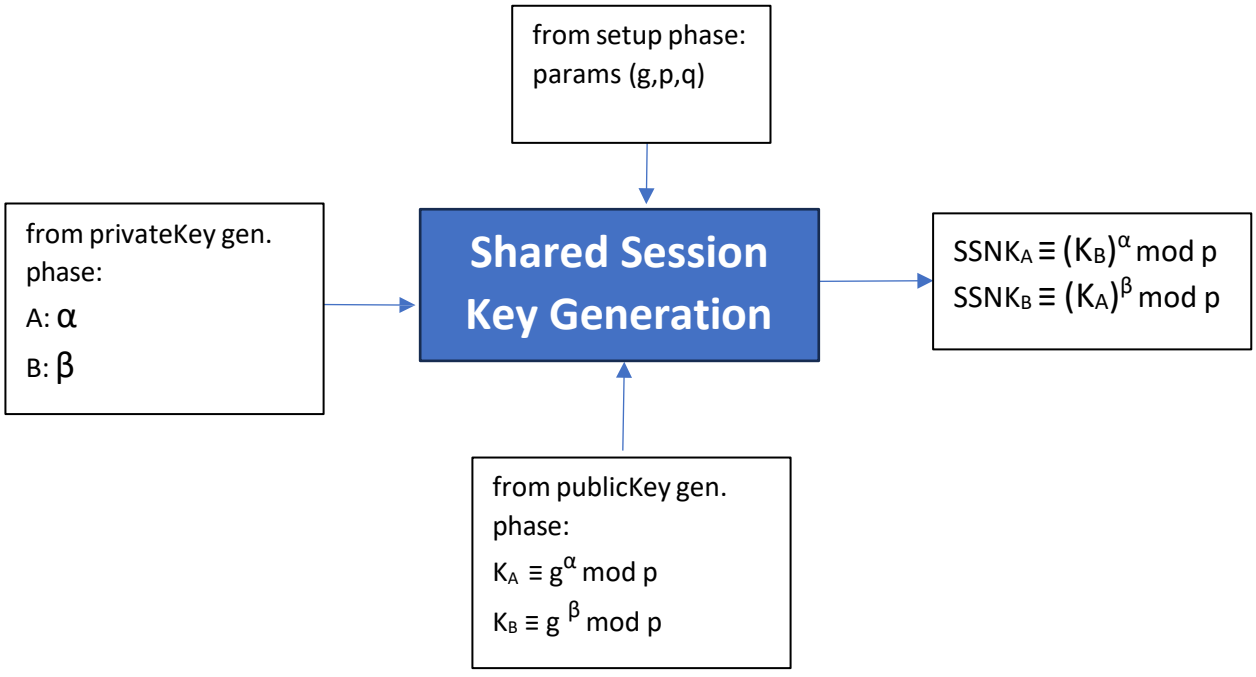
   1. Objective: Session Key Generation

   1. Generate the shared secret key of both the parties.
   2. Execute the programs twice for both the parties and save their computed keys in separate binary files named '**$SSNK_A$.bin**' and '**$SSNK_B$.bin**' , respectively.

   Computation:

   a. Session key generation for party A:
      a. $SSNK_A \equiv (K_B)^{\alpha} \bmod p$
   b. Session key generation for party B:
      a. $SSNK_B \equiv (K_A)^{\beta} \bmod p$

   Workflow:



2. **Objective : Session Key Verification**

Verify the shared session key generated for both the parties by computing their md5sum respectively.

Computation:

1. On the command line write the following command for hash.
    - md5sum <filename>