**LAB 9**

**OBJECTIVE**

To implement ElGamal Encryption System

**THEORY**

ElGamal encryption is a public-key cryptosystem. It uses asymmetric key encryption for communicating between two parties and encrypting the message. This cryptosystem is based on the difficulty of finding discrete logarithm in a cyclic group.

As with Diffie-Hellman, the global elements of ElGamal are a prime number q and α, which is a primitive root of q. User A generates a private/public key pair as follows:

1. Generate a random integer XA, such that 1 < XA < q-1.

2. Compute YA = αXA mod q.

3. A’s private key is XA; A’s pubic key is {q, α, YA}.

Any user B that has access to A’s public key can encrypt a message as follows:

1. Represent the message as an integer M in the range 0 <= M <= q-1. Longer messages are sent as a sequence of blocks, with each block being an integer less than q.

2. Choose a random integer k such that 1 <= k <= q-1.

3. Compute a one-time key K = (YA)k mod q.

4. Encrypt M as the pair of integers (C1, C2) where

C1 = αk mod q

C2 = KM mod q

User A recovers the plaintext as follows:

1. Recover the key by computing K = (C1)XA mod q.

2. Compute M = (C2 K-1) mod q.

**CODE**

# Python program to implement Elgamal Encryption

import random

print("Side A")

q = int(input("Enter a prime number (q) : "))

α = int(input("Enter its primitive root (α) : "))

XA = random.randint(2,q-2)

print(f"A's private Key (XA) = {XA}")

YA = α\*\*XA % q

print(f"A's public Key (q, α, YA) = ({q}, {α}, {YA})")

# Encryption

print("\nSide B")

M = int(input("Enter the message to be sent (between 1 and q) : "))

k = random.randint(1,q-1)

K = YA\*\*k % q

print(f"One time key (K) = {K}")

C1 = α\*\*k % q

C2 = K\*M %q

print(f"Cipher Text (C1, C2) = ({C1}, {C2})")

# Inverse of K

for i in range(q+1):

if K\*i % q == 1:

K\_iv = i

break

# Decryption

print("\nSide A")

K = C1\*\*XA % q

print(f"Decrypted one time key (K) = {K}")

M = C2\*K\_iv % q

print(f"Decrypted message (M) = {M}")

**OUTPUT**

Side A

Enter a prime number (q) : 19

Enter its primitive root (α): 10

A's private Key (XA) = 11

A's public Key (q, α, YA) = (19, 10, 14)

Side B

Enter the message to be sent (between 1 and q) : 9

One time key (K) = 2

Cipher Text (C1, C2) = (13,18)

Side A

Decrypted one time key (K) = 2

Decrypted message (M) = 9

**CONCLUSION**

In this lab, we implemented the ElGamal encryption system.