ELGAMAL CRYPTOGRAPHIC SYSTEM

- In 1984, T. Elgamal announced a public-key scheme based on discrete logarithms, closely related to the Diffie-Hellman technique.
- The ElGamal cryptosystem is used in some form in a number of standards including the digital signature standard (DSS), and the S/MIME e-mail standard
- 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 X_A, such that $1<X_A<q-1$.

2. Compute $Y^A= \alpha^{XA}$ mod q.

3. A's private key is X_A; A's pubic key is $\{q,\alpha,Y_A\}$.

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

1. Represent the message M as an integer 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 $K= (Y_A)^k \bmod q$

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

$C1= \alpha^k \bmod q$; $c2=KM \bmod q$

User A recovers the plaintext as follows:

1. Recover the key by computing .

2. Compute $M=(C_2K^{-1}) \bmod q$

Global Public Elements	
q	prime number
α	$\alpha < q$ and α a primitive root of q

Key Generation by Alice	
Select private X_A	$X_A < q - 1$
Calculate Y_A	$Y_A = \alpha^{XA} \bmod q$
Public key	$PU = \{q, \alpha, Y_A\}$
Private key	X_A

Encryption by Bob with Alice's Public Key	
Plaintext:	$M < q$
Select random integer k	$k < q$
Calculate K	$K = (Y_A)^k \bmod q$
Calculate C_1	$C_1 = \alpha^k \bmod q$
Calculate C_2	$C_2 = KM \bmod q$
Ciphertext:	(C_1, C_2)

Decryption by Alice with Alice's Private Key	
Ciphertext:	(C_1, C_2)
Calculate K	$K = (C_1)^{XA} \bmod q$
Plaintext:	$M = (C_2 K^{-1}) \bmod q$

Reference :William Stallings, Cryptography and Network Security: Principles and Practice, PHI 3rd Edition, 2006

- Let us demonstrate why the ElGamal scheme works. First, we show how is recovered by the decryption process:

$K = (Y_A)^k \bmod q$	K is defined during the encryption process
$K = (\alpha^{X_A} \bmod q)^k \bmod q$	substitute using $Y_A = \alpha^{X_A} \bmod q$
$K = \alpha^{kX_A} \bmod q$	by the rules of modular arithmetic
$K = (C_1)^{X_A} \bmod q$	substitute using $C_1 = \alpha^k \bmod q$

Next, using K, we recover the plaintext as

$$C_2 = KM \bmod q$$

$$(C_2 K^{-1}) \bmod q = KMK^{-1} \bmod q = M \bmod q = M$$

1. Bob generates a random integer k.

2. Bob generates a one-time key K using Alice's public-key components YA, q, and k.

3. Bob encrypts k using the public-key component α, yielding C1, C2 provides sufficient information for Alice to recover K.

4. Bob encrypts the plaintext message using K.

5. Alice recovers K from C1 using her private key.

6. Alice uses $K^{-1}$to recover the plaintext message from C2.

Thus, K functions as a one-time key, used to encrypt and decrypt the message.