The Elgamal cryptosystem is a widely used public-key cryptosystem. It is named after its inventor Taher Elgamal. The Elgamal cryptosystem has become popular due to its strong security properties and many applications.

History

The Elgamal cryptosystem was invented by Taher Elgamal in 1985 while he was a graduate student at Stanford University. The system was developed based on the Diffie-Hellman key exchange. Since its invention, the Elgamal cryptosystem is still used in digital signatures and key exchange.

Encryption Process

The Elgamal cryptosystem uses a key pair for encryption and decryption. The key pair consists of a public key and a private key to encrypt data and is extremely similar to Diffie-Hellman. The Elgamal cryptosystem involves the following steps:

Key Generation: A user selects a large prime number p and a primitive root g of p. The user selects a random number x, computes y = g^x mod p, and publishes (p, g, y) as the public key. The user keeps x as the private key.

Encryption: A user who wants to send a message to the owner of the public key selects a random number k and computes the ciphertext (a, b) as follows: a = g^k mod p and b = y^k \* M mod p, where M is the plaintext message.

Decryption: The owner of the private key computes the plaintext message as follows: M = b \* a^(-x) mod p, where a^(-x) is the modular inverse of a modulo p.

An example of the encryption process follows:

>Suppose Clark wants to send a message to Drew.

>Drew generates his public and private keys as follows:

Key Generation

-Drew selects a prime number p = 29 and a primitive root g = 2 of p.

-Drew selects a random number x = 5,

and computes,

y = g^x mod p = 2^5 mod 29 = 3,

and publishes,

(p, g, y) = (29, 2, 3) as his public key.

and keeps,

keeps x = 5 as his private key.

>Clark wants to send a message M = 13 to Drew. He encrypts the message as follows:

Encryption:

-Clark selects a random number k = 7

and computes the ciphertext (a, b),

a = g^k mod p = 2^7 mod 29 = 4

b = y^k \* M mod p = 3^7 \* 13 mod 29 = 24.

and sends the ciphertext,

(a, b) = (4, 24) to Drew.

>Drew can now decrypt the message as follows:

Decryption

-Drews private key is 5.

Drew computes the plaintext message for M.

M = b \* a^(-x) mod p = 24 \* 4^(-5) mod 29 = 13.

Security

If an attacker can compute the private key x from the public key (p, g, y), then they can easily decrypt any ciphertext. However, no efficient algorithm is known for computing discrete logarithms yet, so the system is believed to be secure. The use of larger prime numbers and longer key increases the security.