

El-Gamal Cryptosystem

Module III

Introduction

- **The *ElGamal*** encryption system is a public key cryptosystem proposed by Tahel ElGamal in 1985 that is based on the Diffie-Hellman key exchange.
- ElGamal cryptosystem steps: Generation of keys (public keys and private keys), Encryption and Decryption.

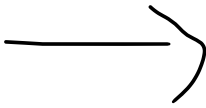
Key Exchange Algorithm

Select $p=13$, $g=2$.
 $\gcd(g, p)=1$

Select secret value d .
 $2 \leq d \leq p-2$.
 $d=3$

$e = g^d \mod p$
 $e = 2^3 \mod 13$
 $e=8$
Private key $d=3$.

Agent X



plaintext = $Y2 * (Y1^d)^{-1} \mod p$
plaintext = $7 * (11^3)^{-1} \mod 13$
plaintext = $7 * 8 \mod 13$
plaintext = $56 \mod 13 = 4$.



Spy

$[P=13, g=2]$

p is prime number, g is generator.
 g is the primitive root of p

Y_1, Y_2

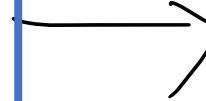
public area

Agent Y wants to send Message $M=4$ to Agent X. M should be less than p .

Select k . A random integer ($k=7$).
 $Y1 = g^k \mod p$
 $Y1 = 2^7 \mod 13$
 $Y1 = 11$.
 $Y2 = M * e^k \mod p$
 $Y2 = 4 * 8^7 \mod 13$
 $Y2 = 7$.



Agent Y



Generate Keys

- Agent X chooses.
 - I. A large prime p .
 - II. A primitive element g modulo p .
 - III. A (possibly random) integer d with $2 \leq d \leq p-2$.
 - IV. Computes $e = g^d \bmod p$.
 - V. Posts public key (p, g, e)
 - VI. Private key is d .

Encryption

1. Agent Y encrypts a short message M ($M < p$) and sends it to Agent X like this:
2. Agent Y chooses a random integer k (which he keeps secret).
3. Agent Y computes $Y1 = g^k \bmod p$ $Y2 = M * e^k \bmod p$
4. Agent Y sends his encrypted message $(Y1, Y2)$ to Agent X.

Decryption

1. When Agent X receives the encrypted message $(Y1, Y2)$, he decrypts (using private key d) by computing
2. Plaintext = $Y2 * (Y1^d)^{-1} \bmod p$.