The ElGamal Cryptosystem

Gianluca Dini
Dept. of Ingegneria dell'Informazione
University of Pisa
Email: gianluca.dini@.unipi.it
Version: 2022-05-01

1

The ElGamal Cryptosystem

INTRODUCTION

2

Introduction

- Taher ElGamal, 1985
- An “extension” of Diffie-Hellman Key Exchange
- One-way function: Discrete Logarithm
- Appliable in any cyclic group where DLP and DHP are intractable
  - We consider $\mathbb{Z}_p^*$

From DHKE to ElGamal encryption

Alice **Bob**

(a) choose $d = privK_B \in \{2, \ldots, p-2\}$

(b) compute $\beta = pubK_B \equiv \alpha^d \bmod p$

< -------------- β -----------------

(c) choose $i = privK_A \in \{2, \ldots, p-2\}$

(d) compute $k_E = pubK_A \equiv \alpha^i \bmod p$

-------------------k_E-------------------- >

(e) compute $k_M \equiv \beta^i \bmod p$

(f) compute $k_M \equiv k_E^d \bmod p$

(g) Encrypt $x \in Z_p^*$

$y \equiv x \cdot k_M \bmod p$

-------------------y------------------- >

(g) decrypt $x \equiv y \cdot k_M^{-1} \bmod p$

From DHKE to ElGamal encryption

- On parameters and keys
  - Domain parameters
    - Large p and primitive element α
  - Keys
    - The public-private pair (d, β) does not change
    - The public-private pair (i, k_E) is generated for every new message
    - k_E is called *ephemeral key*
    - k_M is called the *masking key*

From DHKE to ElGamal encryption

- Intuition
  - One property of cyclic groups is that, given $k_M \in \mathbb{Z}_p^*$, every message x maps to another ciphertext if the two values are multiplied
  - If every k_M is randomly chosen from $\mathbb{Z}_p^*$ then every *y* in {1, 2, …, p – 1} is equally likely
- Remark
  - In the ElGamal encryption scheme we do not need a TTP which generates p and α

The ElGamal encryption scheme

THE ELGAMAL ENCRYPTION SCHEME

May-22 The ElGamal Cryptosystem 7

7

From DHKE to ElGamal encryption

UNIVERSITÀ DI PISA

Alice **Bob**

choose large prime p

choose primitive element α of (a subgroup of) Zp*

choose d = $privK_B \in \{2, \ldots, p-2\}$

compute $\beta = pubK_B \equiv \alpha^d \bmod p$

<-------------- $pubK_B = (p, \alpha, \beta)$ -----------------

choose i = $privK_A \in \{2, \ldots, p-2\}$

compute ephemeral key: $k_E = pubK_A \equiv \alpha^i \bmod p$

compute masking key: $k_M \equiv \beta^i \bmod p$

encrypt $x \in Z_p^*$: $y \equiv x \cdot k_M \bmod p$

-------------------(y, k_E)------------------- >

compute masking key: $k_M \equiv k_E^d \bmod p$

decrypt $x \equiv y \cdot k_M^{-1} \bmod p$

May-22 The ElGamal Cryptosystem 8

8

Proof

- Prove that $x \equiv y \cdot k_M^{-1} \bmod p$
  - Proof
    - $y \cdot k_M^{-1} \equiv (x \cdot k_M) \cdot (k_E^d)^{-1} \equiv (x \cdot (\alpha^d)^i) \cdot ((\alpha^i)^d)^{-1} \equiv$
    - $x \cdot \alpha^{d \cdot i - d \cdot i} \equiv x \bmod p$

ElGamal is probabilistic

- ElGamal encryption scheme is probabilistic
  - Encrypting two identical messages x_1 and x_2 with the same public key $pubK_B = (p, \alpha, \beta)$ results in two different ciphertext y_1 and y_2 (with high probability)
  - Masking key k_M is chosen at random for every new message
  - Brute force against x is avoided a priori

Performance issues

- Communication issues
  - Cyphertext expansion factor is 2
    - The bit size of (y, kE) is twice as the bit size of x
- Computational issues
  - Key Generation
    - Generation of large prime p (at least 1024 bits)
    - privK is generated by a RBG
    - pubK requires a modular exponentiation

May-22 The ElGamal Cryptosystem 11

11

Performance issues

- Computational issues
  - Encryption
    - Two modular exponentiations and a modular multiplication
      - Exponentiations are independent of plaintext
      - Pre-computation of k_E and k_M
  - Decryption
    - A modular exponentiation, a modular inverse and a modular multiplication
      - EEA can be used for modular inverse, or
      - We may combine exponentiation and inverse together, so we just need an exponentiation and a multiplication (➔)

May-22 The ElGamal Cryptosystem 12

12

Computational issues

- How to combine exponentiation and inverse together
  - Proof
    - Recall Fermat's Little Theorem: Let a be an integer and p be a prime, $a^{p-1} \equiv 1 \bmod p$
    - Merge the two steps of decryption: $k_M^{-1} \equiv (k_E^{d})^{-1} \equiv (k_E^{d})^{-1} k_E^{p-1} \equiv k_E^{p-d-1} \bmod p$

ElGamal Cryptosystem

SECURITY ISSUES

Security issues – passive attacks

- The ElGamal problem
  - Recovering x from (p, α , β) and (y, k_E) where $\beta \equiv \alpha^d$ mod p; $k_E = \alpha^i$ mod p, and y = x · β^i mod p
- The ElGamal Problem relies on the hardness of DHP
  - Currently there is no other known method for solving the DHP than solving the DLP
    - The adversary needs to compute Bob's secret exponent *d* or Alice's secret random exponent *i*
    - The Index-calculus method can be applied therefore |p| = 1024+

May-22 The ElGamal Cryptosystem 15

15

Security issues – active attacks

- Active attacks
  - Bob's public key must be authentic
  - Secret exponent *i* must be not reused (➔)
  - ElGamal is malleable (➔)

May-22 The ElGamal Cryptosystem 16

16

Security issues - active attacks

- On reusing the secret exponent i
  - Alice uses the same i for x1 and x2, then
    - both the masking keys and the ephemeral keys would be the same
      - $k_E = \alpha^i \equiv \text{mod } p$
      - $k_M = \beta^i \equiv \text{mod } p$
    - She transmits (y_1, k_E) and (y_2, k_E)
  - The adversary
    - Can easily identify the reuse of i
    - If (s)he can guess/know x_1, then (s)he can compute $x_2 \equiv y_2 \cdot k_M^{-1} \text{ mod } p$ with $k_M \equiv y_1 \cdot x_1^{-1} \text{ mod } p$

May-22 The ElGamal Cryptosystem 17

17

Security issues – active attacks

- On malleability
  - The adversary replaces (k_E, y) by $(k_E, s \cdot y)$
  - The receiver decrypts $x' \equiv x \cdot s \text{ mod } p$
  - Schoolbook ElGamal is often not used in practice, but some padding is introduced

May-22 The ElGamal Cryptosystem 18

18

19