The ElGamal Cryptosystem

Gianluca Dini
Dept. of Ingegneria dell'Informazione
University of Pisa
Email: gianluca.dini@.unipi.it
Version: 2024-04-04

1

The ElGamal Cryptosystem

INTRODUCTION

Apr-24 The ElGamal Cryptosystem 2

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 $\mathbb{Z}_p^*$

Apr-24 The ElGamal Cryptosystem 3

3

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 a new $i \in \{2, \ldots, p-2\}$		
(d) compute $k_E \equiv \alpha^i \bmod p$ (*ephemeral key*)		
	-------------------k_E-------------------- >	
(e) compute $k_M \equiv \beta^i \bmod p$ (*masking key*)		(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 $x \equiv y \cdot k_M^{-1} \bmod p$

Apr-24 The ElGamal Cryptosystem 4

4

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 y 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

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 a new $i \in \{2, \ldots, p-2\}$		
compute *ephemeral key*: $k_E \equiv \alpha^i \bmod p$		
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 $x \equiv y \cdot k_M^{-1} \bmod p$

Consistency

- Consistency proof consists in proving that:
  $x \equiv y \cdot k_M^{-1} \bmod p$
  1. $y \cdot k_M^{-1} \equiv (x \cdot k_M) \cdot (k_E^{d})^{-1} \equiv (x \cdot (\alpha^d)^i) \cdot ((\alpha^i)^d)^{-1} \equiv$
  2. $x \cdot \alpha^{d \cdot i - d \cdot i} \equiv x \bmod p$

Apr-24 The ElGamal Cryptosystem 9

9

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

Apr-24 The ElGamal Cryptosystem 10

10

Performance issues

- Communication issues
  - Cyphertext expansion factor is 2
    - The bit size of (y, k_E) 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

Apr-24 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 (➔)

Apr-24 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
      - $k_M^{-1} \equiv (k_E^{d})^{-1} \equiv (k_E^{d})^{-1}\, k_E^{p-1} \equiv k_E^{p-d-1} \bmod p$

Apr-24 The ElGamal Cryptosystem 13

13

ElGamal Cryptosystem

SECURITY ISSUES

Apr-24 The ElGamal Cryptosystem 14

14

Security issues – passive attacks

- The ElGamal problem
  - Recovering x from (p, α, β) and (y, k_E) where $\beta \equiv \alpha^d \bmod p$; $k_E = \alpha^i \bmod p$, and $y = x \cdot \beta^i \bmod p$
- 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 ➔ $|p| = 1024+$

Apr-24 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 (➔)

Apr-24 The ElGamal Cryptosystem 16

16

Security issues - active attacks

- On reusing the secret exponent i
  - Alice uses the same i for x_1 and x_2, then
    - both the masking keys and the ephemeral keys would be the same
      - $k_E = \alpha^i \equiv \bmod p$
      - $k_M = \beta^i \equiv \bmod p$
    - 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 $x_2 \equiv y_2 \cdot k_M^{-1} \bmod p$ with $k_M \equiv y_1 \cdot x_1^{-1} \bmod p$

17

Security issues – active attacks

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

18

19