

Bahan kuliah IF4020 Kriptografi

Algoritma ElGamal



Oleh: Dr. Ir. Rinaldi, M.T

Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung
2025

Pendahuluan

- Algoritma Elgamal dibuat oleh Taher Elgamal (1985). Pertama kali dikemukakannya di dalam makalah berjudul "*A public key cryptosystem and a signature scheme based on discrete logarithms*", dimuat di dalam [IEEE Transactions on Information Theory](#) (Volume: 31, [Issue: 4](#), July 1985)



A PUBLIC KEY CRYPTOSYSTEM AND A SIGNATURE SCHEME BASED ON DISCRETE LOGARITHMS

TaherElGamal*

Hewlett-Packard Labs
1501 Page Mill Rd
Palo Alto CA 94301

ABSTRACT

A new signature scheme is proposed together with an implementation of the Diffie - Hellman key distribution scheme that achieves a public key cryptosystem. The security of both systems relies on the difficulty of computing discrete logarithms over finite fields.

1. INTRODUCTION

In 1976, Diffie and Hellman [3] introduced the concept of public key cryptography. Since then, several attempts have been made to find practical public key systems (see for example [6,7,9]) depending on the difficulty of solving some problems. For example, the RSA system [9] depends on the difficulty of factoring large integers. This paper presents systems that rely on the difficulty of computing logarithms over finite fields.

Section 2 shows a way to implement the public key distribution scheme introduced by Diffie and Hellman [3] to encrypt and decrypt messages. The security of this system is equivalent to that of the distribution scheme. Section 3 introduces a new digital signature

- Keamanan algoritma ini terletak pada sulitnya menghitung logaritma diskrit.
- *Persoalan logaritma diskrit*: Jika p adalah bilangan prima dan g dan y adalah sembarang bilangan bulat, carilah x sedemikian sehingga

$$g^x \equiv y \pmod{p}$$

Contoh: $7^x \equiv 15 \pmod{41}$, berapakah nilai x ?

- Sebelum membahas algoritma ElGamal lebih lanjut, maka kita perlu memahami terlebih dahulu tentang logaritma diskrit dan akar primitif dari sebuah bilangan prima.

Akar Primitif dan Logaritma Diskrit

- Jika n adalah bilangan bulat, maka a disebut **akar primitif** dari n jika perpangkatan

$$a, a^2, \dots, a^{\phi(n)} \quad (\text{dalam sebuah modulus } n)$$

menghasilkan nilai yang berbeda dan semuanya relatif prima dengan n .

- Secara khusus, jika p adalah bilangan prima, maka a disebut akar primitif dari p jika perpangkatan

$$a, a^2, \dots, a^{p-1} \quad (\text{dalam modulus } p)$$

menghasilkan nilai-nilai yang berbeda

(ingatlah dari fungsi *toitient* Euler, bahwa jika p prima maka $\phi(p) = p - 1$)

- Sebagai contoh, misalkan $p = 7$, maka $a = 3$ adalah akar primitif dari 7 karena

$$3^1 \pmod{7} = 3; \quad 3^2 \pmod{7} = 2; \quad 3^3 \pmod{7} = 6$$

$$3^4 \pmod{7} = 4; \quad 3^5 \pmod{7} = 5; \quad 3^6 \pmod{7} = 1$$

- Jadi, semua perpangkatan dari 3 menghasilkan nilai-nilai yang berbeda (3, 2, 6, 4, 5, 1), semua bilangan di dalam modulus 7 terjadi satu kali.
- Perpangkatan berikutnya, $3^7 \pmod{7}$, $3^8 \pmod{7}$, ..., akan kembali berulang menghasilkan nilai-nilai tersebut. Panjang satu siklus tidak lebih dari $7 - 1 = 6$.
- Secara umum, untuk p bilangan prima, maka panjang siklus tidak lebih dari $p - 1$.

- Perhatikan bahwa $a = 2$ bukan akar primitif dari 7 karena

$$2^1 \pmod{7} = 2; \quad 2^2 \pmod{7} = 4; \quad 2^3 \pmod{7} = 1$$

$$2^4 \pmod{7} = 2; \quad 2^5 \pmod{7} = 4; \quad 2^6 \pmod{7} = 1$$

- Nilai-nilai yang dihasilkan dari $2^1, 2^2, \dots, 2^6$ tidak semuanya berbeda dan tidak mencakup semua nilai di dalam modulus 7.
- Untuk menemukan semua akar primitif dari p , kita harus mencoba semua bilangan bulat dari 2, 3, ...

- Jika a adalah akar primitif dari bilangan prima p , maka untuk bilangan bulat b kita dapat menemukan pangkat x sedemikian sehingga

$$b \equiv a^x \pmod{p}, \quad 0 \leq x \leq (p - 1)$$

- Pangkat x disebut **logaritma diskrit** dari b untuk basis $a \pmod{p}$.
- Dalam **persoalan logaritma diskrit**, diberikan $b \equiv a^x \pmod{p}$, carilah x yang memenuhi kekongruenan tersebut.
- Sebagai contoh, 7 adalah akar primitif dari bilangan prima $p = 41$. Maka, carilah x sedemikian sehingga $15 \equiv 7^x \pmod{41}$, jawabannya adalah 3 karena
$$7^3 = 343 \equiv 15 \pmod{41}$$

Properti algoritma ElGamal:

1. Bilangan prima, p (tidak rahasia)
2. Bilangan acak, g ($g < p$, g adalah akar primitif dari p) (tidak rahasia)
3. Bilangan acak, x ($2 \leq x \leq p - 2$) (rahasia, **kunci privat**)
4. $y = g^x \bmod p$ (tidak rahasia, **kunci publik**)
5. m (plainteks) (rahasia)
6. a dan b (cipherteks) (tidak rahasia)

Prosedur Pembangkitan Kunci

1. Pilih sembarang bilangan prima p (p dapat di-*share* di antara anggota kelompok)
2. Pilih dua buah bilangan acak, g dan x , dengan syarat $g < p$, g akar primitif dari p , dan $2 \leq x \leq p - 2$
3. Hitung $y = g^x \bmod p$ (1)

Hasil dari algoritma ini:

- Kunci publik: tripel (y, g, p)
- Kunci privat: pasangan (x, p)

Prosedur Enkripsi

1. Susun plainteks menjadi blok-blok $m_1, m_2, \dots$, (nilai setiap blok harus berada di dalam selang $[0, p - 1]$).

2. Pilih bilangan acak k , yang dalam hal ini $1 \leq k \leq p - 1$.

3. Setiap blok m dienkripsi dengan rumus

$$a = g^k \bmod p \quad (2)$$

$$b = y^k m \bmod p \quad (3)$$

Pasangan (a, b) adalah cipherteks untuk blok pesan m . Jadi, ukuran cipherteks adalah dua kali ukuran plainteksnya.

Prosedur Dekripsi

1. Gunakan kunci privat x untuk menghitung $(a^x)^{-1} = a^{p-1-x} \bmod p$
2. Hitung plainteks m dengan persamaan:
$$m = b/a^x \bmod p = b(a^x)^{-1} \bmod p$$

Bukti bahwa pesan m dapat diungkap kembali dari pasangan cipherteks a dan b :

$$\text{Dari persamaan (2): } a = g^k \bmod p \rightarrow g^k \equiv a \pmod{p}$$

$$\text{Pangkatkan kedua ruas dengan } x : g^{xk} \equiv a^x \pmod{p}$$

$$\text{Dari persamaan (3): } b = y^k m \bmod p \rightarrow y^k m \equiv b \pmod{p}$$

$$\text{Dari persamaan (2): } y = g^x \bmod p \rightarrow g^x \equiv y \pmod{p},$$

maka

$$\begin{aligned} b/a^x &\equiv y^k m / a^x \pmod{p} \\ &\equiv g^{xk} m / g^{xk} \pmod{p} \\ &\equiv m \pmod{p} \end{aligned}$$

yang berarti bahwa plainteks m dapat diungkap kembali dari cipherteks a dan b .

Contoh 1: Bob membangkitkan kunci publik dan kunci privatnya. Alice akan mengenkripsi pesan dengan menggunakan kunci publik Bob.

(a) Pembangkitan kunci (Oleh Bob)

syarat $g < p$, g akar primitif dari 2357 dan

Misal $p = 2357$, $g = 2$, dan $x = 1751$. $1 \leq x \leq p - 2$

Hitung: $y = g^x \bmod p = 2^{1751} \bmod 2357 = 1185$

Hasil: Kunci publik: $(y = 1185, g = 2, p = 2357)$

Kunci privat: $(x = 1751, p = 2357)$.

Bob memberitahu kunci publik ini kepada Alice

(b) Enkripsi (Oleh Alice)

Misalkan pesan $m = 2035$ (nilai m masih berada di dalam selang $[0, 2357 - 1]$).

Alice memilih bilangan acak $k = 1520$ (nilai k berada di dalam selang $[0, 2357 - 1]$).

Alice melakukan enkripsi dengan kunci public Bob ($y = 1185$, $g = 2$, $p = 2357$):

$$a = g^k \bmod p = 2^{1520} \bmod 2357 = 1430$$

$$b = y^k m \bmod p = 1185^{1520} \cdot 2035 \bmod 2357 = 697$$

Jadi, cipherteks yang dihasilkan adalah (1430, 697).

Alice mengirim cipherteks ini kepada Bob.

(c) Dekripsi (Oleh Bob)

Bob melakukan dekripsi dengan kunci privatnya ($x = 1751$, $p = 2357$):

$$(a^x)^{-1} = a^{p-1-x} \bmod p = 1430^{2357-1-1751} \bmod 2357 = 1430^{605} \bmod 2357 = 872$$

$$m = b/a^x \bmod p = b \cdot (a^x)^{-1} \bmod p = 697 \cdot 872 \bmod 2357 = 2035$$

Bob mendapatkan kembali plainteks $m = 2035$ yang dikirim oleh Alice.

Contoh 2: Alice membangkitkan pasangan kuncinya. Bob akan mengirim pesan dengan menggunakan kunci publik Alice.

(a) Pembangkitan kunci (oleh Alice)

Alice memilih bilangan prima $p = 2273$, akar primitif dari p yaitu $g = 3$, dan $x = 243$.

Alice kemudian menghitung:

$$y = g^x \bmod p = 3^{243} \bmod 2273 = 461$$

Jadi,

kunci privat Alice: $(x = 243, p = 2273)$

kunci publik Alice: $(y = 461, g = 3, p = 2273)$.

(b) Enkripsi (oleh Bob)

Misalkan Bob ingin mengirim plainteks 'HALO' kepada Alice.

Misalkan $A = 00$, $B = 01$, ..., $Z = 25$, maka pesan m dikodekan ke dalam *integer* adalah

$$m = 07001114$$

Bob memecah m menjadi blok yang lebih kecil, misalkan m dipecah menjadi blok-blok sepanjang 4 angka:

$$m_1 = 0700 \quad \text{dan} \quad m_2 = 1114$$

Nilai-nilai m_i ini masih terletak di dalam selang $[0, 2273 - 1]$ agar transformasi menjadi satu-ke-satu.

(i) Enkripsi $m_1 = 0700$

Bob memilih bilangan acak $k = 1463$ (nilai k masih berada di dalam selang $[0, 2273 - 1]$).

Bob mengenkripsi pesan dengan menggunakan kunci publik Alice ($y = 461, g = 3, p = 2273$):

$$a = g^k \bmod p = 3^{1463} \bmod 2273 = 1439$$

$$b = y^k m_1 \bmod p = 461^{1463} \cdot 700 \bmod 2273 = 74$$

Jadi, cipherteks yang dihasilkan untuk m_1 adalah $c_1 = (1439, 74)$.

(ii) Enkripsi $m_2 = 1114$

Bob memilih bilangan acak $k = 2001$ (nilai k masih berada di dalam selang $[0, 2273 - 1]$).

Bob mengenkripsi pesan dengan menggunakan kunci publik Alice ($y = 461, g = 3, p = 2273$):

$$a = g^k \bmod p = 3^{2001} \bmod 2273 = 1220$$

$$b = y^k m_2 \bmod p = 461^{2001} \cdot 1114 \bmod 2273 = 1682$$

Jadi, cipherteks yang dihasilkan untuk m_2 adalah $c_2 = (1220, 1682)$.

Bob mengirim cipherteks $(1439, 74)$ dan $(1220, 1682)$ kepada Alice.

(c) Dekripsi (oleh Alice)

Alice mendekripsi cipherteks dari Bob dengan kunci privatnya ($x = 243$, $p = 2273$):

(i) Dekripsi $c_1 = (1439, 74)$

$$(a^x)^{-1} = a^{p-1-x} \bmod p = 1439^{2273-1-243} \bmod 2273 = 1439^{2029} \bmod 2273 = 1791$$

$$m_1 = b/a^x \bmod p = b(a^x)^{-1} \bmod p = 74 \cdot 1791 \bmod 2273 = 700 = \text{0700}$$

(ii) Dekripsi $c_2 = (1220, 1682)$

$$(a^x)^{-1} = a^{p-1-x} \bmod p = 1220^{2029} \bmod 2273 = 1125$$

$$m_2 = b/a^x \bmod p = 1682(a^x)^{-1} \bmod p = 1682 \cdot 1125 \bmod 2273 = \text{1114}$$

Plainteks yang didekripsi adalah $m_1m_2 = 07001114$, yang kalau dikodekan menjadi teks adalah empat digit adalah “HALO”, sama dengan plainteks yang dikirim oleh Bob.

- Demo online enkripsi ElGamal: <https://www.debjitbiswas.com/elgamal/>

The screenshot shows a web browser window with the address bar displaying <https://www.debjitbiswas.com/elgamal/>. The website is titled "ElGamal Encryption Playground" and has links for "About" and "Github".

Alice's Machine:

- Enter a prime p :** Input field contains 2273. Button: Get Generators.
- Choose g :** Dropdown menu shows 3.
- Enter Private Key x :** Input field contains 243. Button: Get Random Key.
- Calculation:** h is calculated as $h = g^x \bmod p$. Button: Generate & Publish Public Key →.
- Decryption:** "You received a message from Bob." Button: Decrypt. Result: **Bob's decrypted message: 700**.
- Formula:** Message is decrypted using $s = c_1^x \bmod p$ and $m = c_2 \cdot s^{-1} \bmod p$.

Bob's Machine:

- Private Key for Encryption r :** Input field contains 1463. Button: Get Random Key.
- Message m :** Input field contains 700.
- Calculation:** Encrypted message is calculated as $(c_1, c_2) = (g^r \bmod p, (h^r m) \bmod p)$. Button: Encrypt & Send.

Central Cloud:

- Alice's Public Key: $p: 2273$ $g: 3$ $h: 461$
- Bob's encrypted message: $c1: 1439, c2: 74$

Footer: Support this site by trying the Brave browser.