METODE ENKRIPSI DAN DEKRIPSI DENGAN MENGGUNAKAN ALGORITMA ELGAMAL

Mukhammad Ifanto (13508110)

Program Studi Informatika Institut Teknolgi Bandung
Jalan Ganesha 10 Bandung
e-mail: ifuntoo@yahoo.com

ABSTRAK

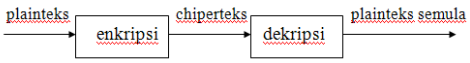
Algoritma asimetris pada kriptografi memiliki kunci enkripsi yang berbeda dengan kunci dekripsi. Algoritma ini menggunakan dua kunci yaitu kunci publik untuk mengenkripsi pesan yang telah tersamarkan dan kunci rahasia untuk mendekripsi pesan agar dapat dibaca. Algoritma ElGamal merupakan salah satu algoritma asimetris dalam kriptografi. Algoritma ini memiliki tingkat keamanan dalam pemecahan masalah logaritma diskret pada grup pergandaan bilangan bulat modulo prima. Dengan mengambil nilai bilangan prima yang besar, maka upaya untuk memecahkan pesan yang telah dienkripsi menjadi sangat sulit. Selain tingkat keamanan pada pemecahan logaritma diskret, algoritma ElGamal memiliki kelebihan dalam menghasilkan cipherteks (pesan yang telah tersamarkan) yang berbeda untuk plainteks (pesan belum disamarkan, masih dapat dibaca dengan jelas) yang sama pada proses enkripsi, tetapi ketika cipherteks di dekripsi akan menghasilkan plainteks yang sama. Kekurangan dari algoritma ini adalah panjang cipherteks yang dihasilkan dua kali panjang plainteks. Algoritma ini mempunyai kunci publik yang terdiri atas 3 buah bilangan dan kunci rahasia yang terdiri atas sebuah bilangan. Kunci publik dan kunci rahasia ini dibuat oleh penerima pesan, kemudian penerima pesan memberitahukan kunci publiknya ke pengirim pesan sehingga pengirim dapat melakukan enkripsi untuk menyamarkan pesan yang akan disampaikan. Kunci rahasia yang dibuat penerima dirahasiakan dari publik. Kunci ini digunakan untuk mendekripsikan cipherteks yang dikirim pengirim. Proses algoritma ElGamal terdiri atas 3 proses yaitu proses pembentukan kunci, proses enkripsi dan proses dekripsi. Setiap proses dalam algoritma ini menggunakan teori bilangan terutama bilangan prima dan modulo bilangan.

Kata kunci: algoritma, asimetris, ElGamal, cipherteks, plainteks, kunci publik, kunci rahasia.

1. PENDAHULUAN

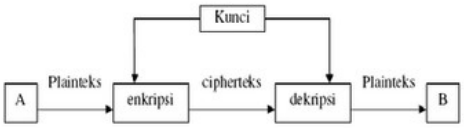
Kemajuan dan perkembangan teknologi informasi dewasa ini telah berpengaruh pada seluruh aspek kehidupan manusia, termasuk bidang komunikasi. Informasi-informasi rahasia perlu disimpan atau disampaikan melalui suatu cara tertentu agar tidak diketahui oleh pihak yang tidak dikehendaki Oleh karena itu terciptalah ilmu kriptografi.

Kriptografi merupakan ilmu sekaligus seni untuk menjaga kerahasiaan pesan dengan cara menyamarkannya menjadi bentuk tersandi yang tidak mempunyai makna. Pesan yang disamarkan (teks jelas yang dapat dimengerti) dinamakan plainteks, sedangkan pesan hasil penyamaran (teks tersandi) dinamakan chiperteks. Proses kriptografi terdiri atas enkripsi dan dekripsi. Enkripsi merupakan proses penyamaran dari plainteks ke chiperteks sedangkan dekripsi merupakan proses pembalikan dari chiperteks ke plainteks.



Gambar 1. Diagram proses kriptografi

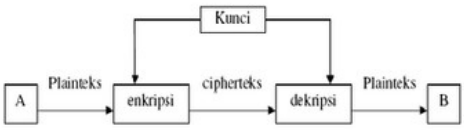
Kriptografi yang pertama kali dibuat menggunakan algoritma simetris yang disebut juga algoritma kunci rahasia atau sandi kunci rahasia. Algoritma ini mempunyai kunci enkripsi yang sama dengan kunci dekripsi.



Gambar 2. Diagram algoritma simetris

Keamanan dari algoritma ini tergantung dari penerima dan pengirim yang menyimpan rahasia tersebut. Namun algoritma ini tidak efisien bila digunakan untuk berkomunikasi dengan banyak orang. Oleh karena itu dibuat algoritma yang mana kunci enkripsinya tidak dirahasiakan sehingga dapat dilihat oleh siapa saja (kunci

publik). Sedangkan kunci dekripsinya dirahasiakan dan hanya orang-orang tertentu saja yang boleh mengetahuinya (kunci rahasia). Algoritma ini dinamakan algoritma asimetris karena kunci enkripsinya yang berbeda dengan kunci dekripsinya.



Gambar 3. Diagram algoritma asimetris

Kunci ini tidak dirahasiakan sehingga dapat dilihat oleh siapa saja. Sedangkan kunci rahasia adalah kunci yang dirahasiakan dan hanya orang-orang tertentu saja yang boleh mengetahuinya. Keuntungan utama dari algoritma ini adalah memberikan jaminan keamanan kepada siapa saja yang melakukan pertukaran informasi meskipun di antara mereka tidak ada kesepakatan mengenai keamanan pesan terlebih dahulu maupun saling tidak mengenal satu sama lainnya.

Salah satu algoritma asimetris adalah Algoritma ElGamal. Algoritma ini dikembangkan pertama kali oleh Taher ElGamal. Algoritma ini diaplikasikan pada PGP dan GnuPG yang dapat digunakan untuk pengamanan e-mail dan tanda tangan digital. Algoritma ini juga diadopsi dalam pembentukan algoritma asimetris lainnya yang dikenal dengan nama DSA (Digital Signature Algorithm).

Pada makalah ini akan dibahas, bagaimana melakukan proses enkripsi dan dekripsi dengan menggunakan algoritma ElGamal meliputi konsep matematis yang melandasinya dan proses penyandiannya.

2. ALGORITMA ELGAMAL

Algoritma ElGamal merupakan algoritma dalam kriptografi yang termasuk dalam kategori algoritma asimetris. Keamanan algoritma ElGamal terletak pada kesulitan penghitungan logaritma diskret pada bilangan modulo prima yang besar sehingga upaya untuk menyelesaikan masalah logaritma ini menjadi sangat sukar.

Algoritma ElGamal mempunyai kunci publik berupa tiga pasang bilangan dan kunci rahasia berupa satu bilangan. Algoritma ini mempunyai kerugian pada cipherteksnya yang mempunyai panjang dua kali lipat dari plainteksnya. Akan tetapi, algoritma ini mempunyai kelebihan pada enkripsi. Untuk plainteks yang sama, algoritma ini memberikan cipherteks yang berbeda (dengan kepastian yang dekat) setiap kali plainteks di enkripsi.

Algoritma ElGamal terdiri dari tiga proses, yaitu proses pembentukan kunci, proses enkripsi dan proses dekripsi. Algoritma ini merupakan cipher blok, yaitu melakukan proses enkripsi pada blok-blok plainteks dan menghasilkan blok-blok cipherteks yang kemudian dilakukan proses dekripsi dan hasilnya digabungkan.

2.1 Proses Pembentukan Kunci

Pembentukan kunci terdiri atas pembentukan kunci publik dan kunci rahasia. Pada proses ini dibutuhkan sebuah bilangan prima p yang digunakan untuk membentuk grup Zp*, elemen primitif α dan sembarang $a \in \{0,1,..., p-2\}$.

Kunci publik algoritma ElGamal terdiri atas pasangan 3 bilangan (p,α ,β) di mana

$$\beta = \alpha^a \bmod p \qquad (1)$$

Sedangkan kunci rahasianya adalah bilangan a tersebut. Proses pembentukan kunci untuk algoritma ElGamal terdiri atas:

a Penentuan bilangan prima aman yang bernilai besar
b Penentuan elemen primitif
c Pembentukan kunci berdasarkan bilangan prima aman dan elemen primitif

2.1.1 Penentuan bilangan prima aman besar

Tujuan penentuan bilangan prima aman ini adalah untuk mempermudah dalam penentuan elemen primitif. Digunakan bilangan prima p sehingga

$$p = 2.q + 1 \qquad (2)$$

dengan q adalah bilangan prima sehingga nilai minimal p adalah 5 dan q adalah 2. Bilangan prima p tersebut disebut sebagai bilangan prima aman.

Langkah penentuan bilangan prima tersebut dinyatakan sebagai berikut:

a Tentukan bilangan prima $p \geq 5$
b Hitung q dengan "persamaan (2)"
c Jika q merupakan bilangan prima, maka p merupakan bilangan prima aman.
d Jika q bukan merupakan bilangan prima, maka p bukan merupakan bilangan prima aman.

Untuk menguji keprimaan suatu bilangan, digunakan suatu metode yang disebut **Teorema Fermat**.

Teorema Fermat

Jika x adalah bilangan prima dan y adalah bilangan bulat yang tidak habis dibagi dengan x, yaitu PBB(y,x) = 1, maka

$$y^{x-1} \equiv 1 \ (\bmod p) \qquad (3)$$

Contoh pengujian keprimaan suatu bilangan.

Diuji apakah 17 dan 21 bilangan prima atau bukan. Kemudian diambil nilai a = 2 karena PBB(17,2) = 1 dan PBB(21,2) = 1. Untuk 17,

$$2^{17-1} = 65536 \equiv 1 \ (\bmod 17)$$

Karena 17 habis membagi 65536 – 1 = 65535 (65535 ÷ 17 = 3855).
Untuk 21,

$2^{21-1} = 1048576 \equiv\backslash 1 \pmod{21}$

Karena 21 tidak habis membagi 1048576 – 1 = 1048576.

Akan tetapi, teorema ini memiliki kekurangan karena terdapat bilangan komposit *n* sedemikian sehingga $2^{n-1} \equiv 1 \pmod{n}$. Bilangan itu disebut bilangan **prima semu** (*pseudoprimes*). Mislanya komposit 341 (yaitu 341 = 11. 31) adalah bilangan prima semu menurut teoreme Fermat,

$2^{340} \equiv 1 \pmod{340}$

Namun, bilangan prima semu ini relatif jarang terdapat.

2.1.2 Penentuan elemen primitif

Teorema:
"Suatu elemen yang membangun Z_p * disebut *elemen primitif (primitive root) mod p*."
"Bila $\alpha^2 \bmod p \neq 1$ dan $\alpha^q \bmod p \neq 1$. Jika keduanya dipenuhi, maka α adalah elemen primitif dari Z_p *."

Langkah penentuan elemen primitif tersebut dapat dinyatakan sebagai berikut:

a Tentukan bilangan prima $p \geq 5$ dan $\alpha \in Z_p$ *
b Hitung *q* dengan "persamaan (2)"
c Hitung $\alpha^2 \bmod p$ dan $\alpha^q \bmod p$.
d Jika $\alpha^2 \bmod p = 1$ atau $\alpha^q \bmod p = 1$, maka α bukan merupakan elemen primitif.
e Jika $\alpha^2 \bmod p \neq 1$ dan $\alpha^q \bmod p \neq 1$, maka α bukan merupakan elemen primitif.

2.1.3 Pembentukan kunci berdasarkan bilangan prima aman dan elemen primitif

Setelah bilangan prima aman dan elemen primitif diperoleh, kunci publik dan kunci rahasia untuk algoritma ElGamal dapat dibentuk. Algoritma ElGamal dalam prosesnya menggunakan bilangan bulat untuk perhitungan. Oleh karena itu, pesan yang terkandung dalam plainteks harus dalam bentuk bilangan bulat.

Untuk memenuhi persyaratan tersebut, digunakan kode ASCII *(American Standard for Information Interchange)* yang merupakan representasi numeric dari karakter-karakter yang digunakan pada komputer, serta mempunyai nilai minimal 0 dan maksimal 255.

Selanjutnya, dengan kondisi-kondisi tersebut, pembentukan kunci dapat dibentuk dengan mengacu pada langkah berikut:

a Tentukan bilangan prima $p \geq 5$ dan $\alpha \in Z_p$ *
b Pilih $a \in \{0,1,..., p-2\}$ sembarang.
c Hitung nilai β dengan rumus

$$\beta = \alpha^a \bmod p \quad (4)$$

Diperoleh kunci publik (p,α,β) yang dapat dipublikasikan serta nilai kunci rahasia *a* yang dirahasiakan nilainya.

Pihak yang membuat kunci publik dan kunci rahasia merupakan pihak penerima pesan. Sedangkan pihak pengirim hanya mengetahui kunci publik dari penerima untuk mengenkripsi pesan yang akan dikirim.

2.2 Proses Enkripsi

Proses enkripsi menggunakan kunci publik (p,α,β) dan sebuah bilangan integer acak *k* ($k \in \{0,1,..., p-1\}$) yang dijaga kerahasiaannya oleh penerima yang mengenkripsi pesan. Untuk setiap karakter dalam pesan dienkripsi dengan menggunakan bilangan k yang berbeda-beda. Satu karakter yang direpresentasikan dengan menggunakan bilangan bulat ASCII akan menghasilkan kode dalam bentuk blok yang terdiri atas dua nilai (*r*, *t*).

Langkah proses enkripsi:

a Ambil sebuah karakter dalam pesan yang akan dienkripsi dan transformasi karakter tersebut ke dalam kode ASCII sehingga diperoleh bilangan bulat *M*.
b Hitung nilai *r* dan *t* dengan persamaan berikut:
$r = \alpha^k \pmod{p}$ (5)
$t = \beta^k M \pmod{p}$ (6)
c Diperoleh cipherteks untuk karakter *M* tersebut dalam blok (*r, t*)
d Lakukan proses di atas untuk seluruh karakter dalam pesan termasuk karakter spasi.

2.2 Proses Dekripsi

Dekripsi dari cipherteks ke plainteks menggunakan kunci rahasia *a* yang disimpan kerahasiaanya oleh penerima pesan.

Teorema:
Diberikan (p,α,β) sebagai kunci public dan *a* sebagai kunci rahasia pada algoritma ElGamal. Jika diberikan cipherteks (*r, t*), maka

$M = t\,(r^a)^{-1} \bmod p$ (7)
dengan *M* adalah plainteks.
Di mana nilai
$(r^a)^{-1} = r^{-a} = r^{p-1-a} \bmod p.$ (8)

Langkah proses dekripsi:

a Ambil sebuah blok cipherteks dari pesan yang telah dienkripsikan pengirim.
b Dengan menggunakan *a* yang dirahasiakan oleh penerima, hitung nilai plainteks dengan menggunakan "persamaan (7)" dan "persamaan (8)".

3. PENERAPAN ALGORITMA ELGAMAL

Sebagai ilustrasi dari algoritma ElGamal tersebut, algoritma tersebut diaplikasikan pada sebuah kasus.

Misal kasus tersebut:

Alice ingin saling berkomunikasi dengan Bob. Suatu hari ia ingin mengirimkan sebuah pesan rahasia untuk Bob agar tidak diketahui orang lain. Alice menggunakan algoritma ElGamal untuk menyamarkan pesan yang akan disampaikan. Pesan tersebut, "SELAMAT PAGI".

Ditunjukkan proses penyamaran pesan tersebut melalui langkah-langkah yang telah disebutkan.

3. 1. PEMBENTUKAN KUNCI OLEH PENERIMA PESAN

Pembentukan kunci publik dan kunci rahasia dilakukan oleh penerima pesan yaitu Alice. Langkah-langkah yang dilakukan Alice dalam pembentukan kunci:

a Menentukan bilangan prima aman *p*, disarankan ambil nilai *p* yang besar. Diambil nilai $p = 107$.

b Alice mengecek apakah *p* termasuk bilangan prima aman atau tidak dengan mencari nilai *q* berdasarkan "persamaan (2)". Diperoleh nilai $q = 53$ yang juga merupakan bilangan prima. Oleh karena itu, Alice menyimpulkan *p* merupakan bilangan prima aman.

c Selanjutnya Alice mencari elemen primitif α dari Z_p^*.

d Alice membuat tabel perhitungan untuk beberapa nilai α untuk mengecek apakah nilai tersebut termasuk elemen primitif atau tidak.

Tabel 1. Perhitungan $\alpha^2 \bmod p$ dan $\alpha^q \bmod p$

α	2	3	4	5	6
$\alpha^2 \bmod p$	4	9	16	25	36
$\alpha^q \bmod p$	106	1	1	106	106

e Dari beberapa nilai tersebut Alice mendapatkan beberapa nilai α dari Z_p^* yaitu 2, 5 dan 6. Alice memilih nilai α yang dipakai sebagai elemen primitif adalah $\alpha = 2$.

f Kemudian Alice menentukan kunci rahasia *a* (di mana $a \in \{0,1,..., p-2\}$) dengan nilai 63 sehingga sekarang Alice mempunyai nilai $(p, \alpha, a) = (107, 2, 63)$.

g Dengan nilai *a* yang sudah diketahui, Alice mencari nilai β dengan "persamaan (4)".
$\beta = \alpha^a \bmod p$
$\beta = 2^{63} \bmod 107$
$\beta = 46$.

h Alice mendapatkan kunci publik $(p, \alpha, \beta) = (107, 2, 46)$ dan kunci rahasia $a = 63$. Kunci publik tersebut diberitahukan ke Bob untuk mengenkripsi pesan yang akan dikirim Bob ke Alice dan kunci rahasia dijaga keamanannya oleh Alice.

3. 2. ENKRIPSI PESAN OLEH PENGIRIM

Bob menerima kunci publik dari Alice $(p, \alpha, \beta) = (107, 2, 46)$. Dengan kunci publik tersebu, Bob mengenkripsi pesan "SELAMAT PAGI" untuk dikirimkan ke Alice. Langkah-langkah yang dilakukan Bob dalam mengenkripsi pesan tersebut:

a Pertama Bob mengonversi pesan tersebut dalam kode ASCII.

Tabel 2. Konversi karakter ke kode ASCII

i	karakter	Plainteks M_i	ASCII
1	S	M_1	83
2	E	M_2	69
3	L	M_3	76
4	A	M_4	65
5	M	M_5	77
6	A	M_6	65
7	T	M_7	84
8	<spasi>	M_8	32
9	P	M_9	80
10	A	M_{10}	65
11	G	M_{11}	71
12	I	M_{12}	73

b Kemudian Bob menentukan bilangan acak *k* $(k \in \{0,1,..., 106\})$ yang dijaga kerahasiaannya untuk setiap plainteks *M* dan mengenkripsi plainteks tersebut dengan menghitung nilai *r* dan *t* dengan "persamaan (5)" dan "persamaan (6)".

Tabel 3. Enkripsi plainteks ke cipherteks

i	M_i	k_i	$r = 2^k \pmod{107}$	$t = 46^k M_i \pmod{107}$
1	83	57	91	21
2	69	43	7	78
3	76	65	77	82
4	65	88	89	66
5	77	34	9	98
6	65	46	56	93
7	84	47	5	4
8	32	76	85	22
9	80	87	98	83
10	65	69	55	23
11	71	41	82	11
12	73	35	18	23

c Berdasarkan tabel tersebut, Bob memperoleh cipherteks (r_i, t_i), $i = 1, 2, \ldots, 12$ sebagai berikut:
(91, 21)
(7, 78)
(77, 82)
(89, 66)

(9, 98)
(56, 93)
(5, 4)
(85, 22)
(98, 83)
(55, 23)
(82, 11)
(18, 23

d Selanjutnya cipherteks tersebut dikirimkan ke Alice.

Tampak bahwa dengan menggunakan bilangan integer acak yang berbeda akan menghasilkan cipherteks yang berbeda pula. Namun, ketika cipherteks tersebut di dekripsi, akan diperoleh plainteks yang sama.

3. 2. DEKRIPSI PESAN OLEH PENERIMA

Alice memperoleh pesan yang telah disamarkan dari Bob. Karena Alice yang memegang kunci rahasia ($a = 63$) dari enkripsi tersebut, Alice dapat mendekrip pesan tersebut agar dapat dibaca.

Dengan menggunakan "persamaan (7)" dan "persamaan (8)", Alice melakukan perhitungan untuk tiap blok cipherteks.

Tabel 3. Dekripsi cipherteks ke plainteks

i	r	t	$r^{43} \bmod 107$	$M_i = t\, r^{43} \bmod 107$	Karakter M_i
1	91	21	60	83	S
2	7	78	5	69	E
3	77	82	74	76	L
4	89	66	48	65	A
5	9	98	39	77	M
6	56	93	3	65	A
7	5	4	21	84	T
8	85	22	89	32	<spasi>
9	98	83	68	80	P
10	55	23	54	65	A
11	82	11	94	71	G
12	18	23	59	73	I

Berdasarkan tabel tersebut, Alice memperoleh plainteks dari pendekripsian cipherteks yang diberikan Bob. Pesan yang hendak disampaikan Bob: "SELAMAT PAGI".

IV. KESIMPULAN

a Algoritma asimetris memiliki kunci enkripsi yang berbeda dengan kunci dekripsi. Kunci untuk enkripsi disebut kunci publik dan kunci untuk dekripsi disebut kunci rahasia.

b Algoritma ElGamal merupakan salah satu algoritma asimetris dalam kriptografi. Algoritma ini memiliki kunci publik yang terdiri atas 3 bilangan dan kunci rahasia yang terdiri atas sebuah bilangan.

c Tingkat keamanan algoritma ini didasarkan pada kesulitan pemecahan masalah logaritma diskret pada penggandaan bilangan bulat modula prima yang besar.

d Cipherteks yang dihasilkan dari plainteks dengan menggunakan algoritma ElGamal dapat berbeda-beda karena adanya penggunaan bilangan acak pada pengenkripsian plainteks. Akan tetapi, ketika didekripsikan, plainteks yang dihasilkan sama.

e Penggunaan blok-blok cipherteks pada algoritma ElGamal menyebabkan panjang cipherteks menjadi dua kali panjang dari plainteks.

REFERENSI

[1] Munir, Rinaldi. "Struktur Diskrit", Informatika, 2003.

[2] http://sandi.math.web.id , Akses : 3:04, 19 Desember 2009

[3] http://en.wikipedia.org/wiki/ElGamal_encryption , Akses: 3:02, 19 Desember 2009.

[4] www.informatics.indiana.edu/markus/i400/lecture7.ppt , Akses: 3:10, 19 Desember 2009.

[5] www.math.uic.edu/~leon/mcs425-s08/handouts/el-gamal.pdf , Akses: 3:11, 19 Desember 2009