IMPLEMENTASI ALGORITMA KUNCI *PUBLIC* PADA ALGORITMA RSA

Hersatoto Listiyono
Fakultas Teknologi Informasi, Universitas Stikubank Semarang

Abstrak:

Ide dasar sistem kriptografi kunci *public* adalah bahwa kunci kriptografi dibuat sepasang, satu kunci untuk enkripsi dan satu kunci untuk dekripsi. Kunci untuk enkripsi bersifat *public* (tidak rahasia) – sehingga dinamakan kunci *public* (public-kunci) – sedangkan kunci dekripsi bersifat rahasia – sehingga dinamakan kunci *private* (*secret kunci*).

Hasil kajian teknik enkripsi data dengan metode algoritma simetri menunjukan bahwa kelemahan-kelemahan pada metode simetri dengan kunci tunggal bisa diselesaikan dengan menggunakan metode lain yang disebut dengan metode kunci public yang menggunakan 2 (dua) kunci yaitu kunci *public* dan kunci *private*. Hasil kajian teknik enkripsi dengan menggunakan metode kunci *public* yang pernah dilakukan menunjukan bahwa penggunaan 2 (dua) buah kunci untuk pengamanan data memberikan keamanan yang handal dan efisien.

Sistem kriptografi kunci *public* cocok untuk kelompok pengguna di lingkungan jaringan komputer. Setiap pengguna jaringan mempunyai kunci *public* dan kunci *private* yang bersesuaian.

Dari sekian banyak algoritma kriptografi kunci-publik yang pernah dibuat, algoritma yang populer adalah algoritma RSA. Langkah dalam algoritma RSA adalah membuat pasangan kunci yaitu Kunci *public* dan kunci *private*. Keamanan algoritma RSA terletak pada sulitnya memfaktorkan bilangan yang besar menjadi faktor-faktor prima. Pemfaktoran dilakukan untuk memperoleh kunci *private*. Selama pemfaktoran bilangan besar menjadi faktor-faktor prima belum ditemukan algoritma yang mangkus, maka selama itu pula keamanan algoritma RSA tetap terjamin.

Kata Kunci : enkripsi, dekripsi, kriptografi, RSA

PENDAHULUAN

Hingga pada akhir tahun 1970, hanya ada sistem kriptografi simetri. Sistem kriptografi simetri menggunakan kunci yang sama untuk enkripsi dan dekripsi, maka hal ini mensyaratkan bahwa dua pihak yang berkomunikasi saling mempercayai. Satu masalah besar dalam algoritma simetri: bagaimana mengirimkan kunci rahasia kepada penerima?. Mengirim kunci rahasia pada saluran publik (telepon, pos, dll) sangat tidak aman.

Sistem kriptografi kunci *public* ditemukan oleh Diffie dan Hellman yang mempresentasikan konsep ini pada Tahun 1976. Konsep dasar Kriptografi Kunci *Public* terletak pada pemahaman bahwa kunci-kunci selalu berpasangan yaitu kunci enkripsi dan kunci dekripsi. Pemahaman kunci enkripsi dan dekripsi sering disebut sebagai kunci *public* dan *private* . Seseorang harus memberikan kunci *public* agar pihak lain dapat meng-enkripsi sebuah pesan. Dekripsi hanya terjadi jika seseorang mempunyai kunci private. Kunci-kunci ini dipilih sedemikian sehingga secara praktek tidak mungkin menurunkan kunci rahasia dari kunci *public*.

Sistem kriptografi kunci *public* cocok untuk kelompok pengguna di lingkungan jaringan komputer. Setiap pengguna jaringan mempunyai kunci *public* dan kunci *private* yang bersesuaian. Kunci *public*, karena tidak rahasia, biasanya disimpan di dalam basis data kunci yang dapat diakses oleh pengguna lain. Jika ada pengguna yang hendak berkirim pesan ke pengguna lainnya, maka ia ia perlu mengetahui kunci *public* penerima pesan melalui basis data kunci ini lalu menggunakannya untuk mengenkripsi pesan. Hanya penerima pesan yang berhak yang dapat mendekripsi pesan karena ia mempunyai kunci rahasia. Dengan sistem kriptografi kunci *public*, tidak diperlukan pengiriman kunci rahasia melalui saluran komunikasi khusus sebagaimana pada sistem kriptografi simetri. Meskipun kunci *public* diumumkan ke setiap orang di dalam kelompok, namun kunci *public* perlu dilindungi agar otentikasinya terjamin.

Dari penjelasan di atas, kelebihan-kelebihan kriptografi kunci-*public* adalah sebagai berikut : hanya kunci *private* yang perlu dijaga kerahasiaannya oleh setiap pihak yang berkomunikasi, tidak ada kebutuhan untuk mengirim kunci *private* sebagaimana

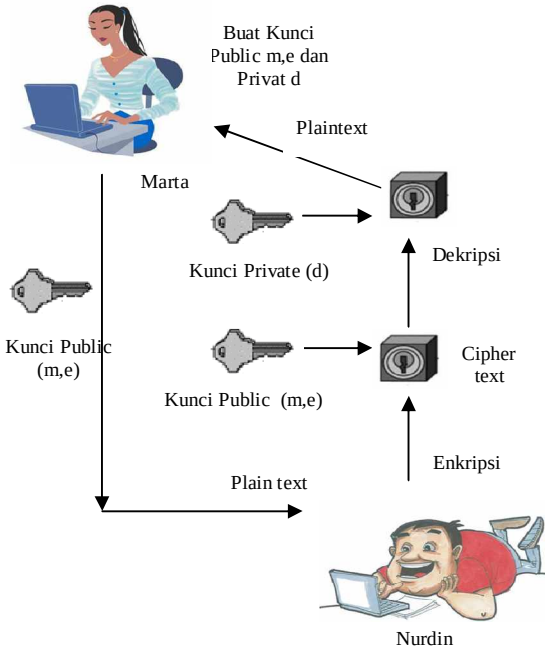
pada sistem simetri, pasangan kunci *public*/kunci *private* tidak perlu diubah, bahkan dalam periode waktu yang panjang, beberapa algoritma kunci *public* dapat digunakan untuk memberi tanda tangan digital pada pesan.

ALGORITMA RSA

Algoritma RSA adalah sebuah algoritma berdasarkan skema kriptografi kunci *public*. Nama RSA sebagai diambil dari inisial nama para penemunya: Ron Rivest, Adi Shamir, dan Leonard Adleman. RSA dibuat di MIT pada tahun 1977 dan dipatenkan oleh MIT pada tahun 1983. Setelah bulan September tahun 2000, paten tersebut berakhir, sehingga saat ini semua orang dapat menggunakannya dengan bebas.

Algoritma RSA adalah algoritma yang mudah untuk diimplementasikan dan dimengerti. Algoritma RSA adalah sebuah aplikasi dari sekian banyak teori seperti extended euclid algorithm, euler's function sampai fermat theorem.

Diagram Algoritma RSA



TAHAP-TAHAP ALGORITMA RSA

Pembangkitan Kunci pada Algoritma RSA

Misalkan Nurdin ingin mengirim sebuah pesan melalui jalur yang aman kepada Marta. Marta akan memberikan kunci *public*nya kepada Nurdin sedangkan kunci *private* akan disimpan untuk dirinya sendiri.

a. Pilih 2 bilangan prima besar seperti p,q dimana p tidak sama
dengan q.

b. Hitung m = p x q

c. Hitung n = (p-1)*(q-1)

d. Pilih **e** yang relatively prime terhadap n,
e relatively prime terhadap n artinya faktor pembagi terbesar keduanya adalah 1, secara matematis dinotasikan gcd(e,n) = 1. Untuk mencarinya dapat digunakan algoritma Euclid.

e. Hitung **d** integer sehingga e*d = 1 mod n atau (1+mn)/e.

Untuk bilangan besar, dapat digunakan algoritma extended Euclid.

f. **(m**, **e)** adalah kunci *public* yang diberikan untuk pihak lain yang ingin berkomunikasi (pihak yang akan mengirim pesan) untuk keperluan enkripsi.

g. **d** adalah kunci *private* yang harus dipegang sendiri oleh pihak yang akan menerima pesan untuk keperluan dekripsi.

Enkripsi

Misalkan Nurdin ingin mengirim sebuah pesan X kepada Marta.

a. Nurdin akan menerima atau mendapatkan kunci public dari Marta yaitu (m,e).
Dengan menggunakan kunci public (m,e) untuk mengenkripsi pesan M.
C = n^e (mod m)
C adalah ciphertext hasil enkripsi plaintext X

b. Nurdin akan mengirimkan C tersebut kepada Marta.

Dekripsi

Setelah ciphertext yang dikirim Nurdin diterima Marta, Marta akan menggunakan kunci private (m,d) untuk mendekripsi ciphertext tersebut agar dapat diketahui plaintextnya. Adapun tahapan-tahapan yang digunakan adalah sebagai berikut:

a. N = C^d (mod m)

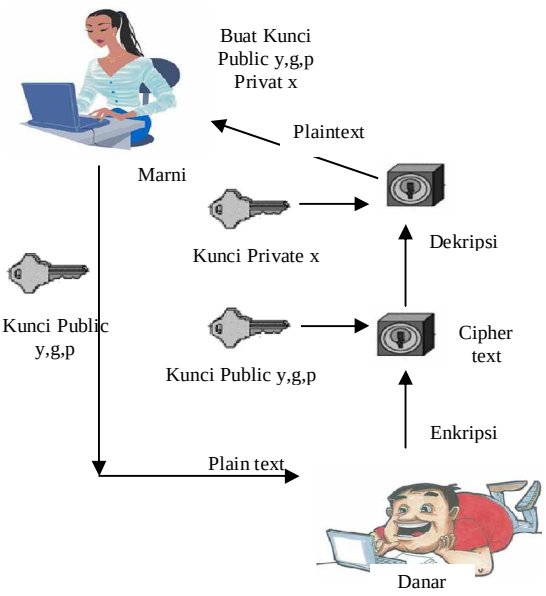
b. N adalah plaintext yang ditemukan setelah mendekripsi C dengan kunci private d dan kunci public m.

PERBANDINGAN ALGORITMA RSA DAN ELGAMAL

Algoritma Elgamal

Algoritma Elgamal juga merupakan algoritma berdasarkan konsep kunci *public* seperti halnya algoritma RSA. Algoritma Elgamal pada umumnya digunakan untuk digital signature, namun kemudian dimodifikasi sehingga juga bisa digunakan untuk enkripsi dan deskripsi. ElGamal digunakan dalam perangkat lunak sekuriti yang dikembangkan oleh GNU, program PGP, dan pada sistem sekuriti lainnya. Kekuatan algoritma ini terletak pada sulitnya menghitung algoritma diskrit.

Algoritma Elgamal tidak dipatenkan, karena algoritma ini didasarkan pada algoritma Diffie – Hellman, sehingga hak paten algoritma Diffie – Hellman juga mencakup algoritma ElGamal. Karena hak paten algoritma Diffie – Hellman berakhir pada bulan April 1997, maka algoritma ElGamal dapat diimplementasikan untuk aplikasi komersil.



Besaran-besaran dalam algoritma ElGamal

Besaran-besaran yang digunakan di dalam algoritma ElGamal:

1. Bilangan prima, p (bersifat *public*/tidak rahasia)
2. Bilangan acak, g (dimana g < p dan bersifat *public*/tidak rahasia)
3. Bilangan acak, x (x < p dan bersifat *private*/rahasia)
4. m merupakan plainteks dan bersifat *private*/rahasia.
5. a dan b merupakan pasangan cipherteks hasil enkripsi bersifat *private* tidak rahasia.

Tahap-Tahap dalam Algoritma Elgamal

Pembangkitan Pasangan Kunci

Langkah-langkah dalam membangkitkan pasangan kunci adalah sebagai berikut :

1. Pilih sembarang bilangan prima p.
2. Pilih dua buah bilangan acak, g dan x, dengan syarat $g < p$ dan $1 \leq x \leq p - 2$.
3. Hitung $y = g^x \bmod p$.

Kunci *public* adalah y, g, p sedangkan kunci *private*nya adalah x. Nilai y, g dan p tidak dirahasiakan dan dapat diumumkan kepada anggota kelompok.

Enkripsi

Langkah-langkah pada enkripsi adalah sebagai berikut :

1. Plainteks disusun menjadi blok-blok m1, m2, …, sedemikian sehingga setiap blok merepresentasikan nilai di dalam rentang 0 sampai p – 1.
2. Pilih bilangan acak k, yang dalam hal ini 0 <= k <= p – 1, sedemikian sehingga k relatif prima dengan p – 1.
3. Setiap blok m dienkripsi dengan rumus
   a = g^k mod p
   b = y^m mod p

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

Deskripsi

Untuk mendekripsi a dan b digunakan kunci rahasia, x, dan plainteks m diperoleh kembali dengan persamaan

$$1/a^x = (a^x)^{-1} = a^{p-1-x} \bmod p$$

$$m = b/ax \bmod p$$

yang berarti bahwa plainteks dapat ditemukan kembali dari pasangan cipherteks a dan b.

CONTOH KASUS

Penggunaan Algoritma RSA

Misalkan Nurdin akan mengirim pesan kepada Marta.

a. Pembangkitan Kunci

Marta membangkitkan kunci dengan langkah-langkah sebagai berikut :

1. Menentukan bilangan p, q misalkan p = 3 dan q = 11.
2. Menghitung m dan n
   m = 3 * 11 = 33
   n = (3-1) * (11-1) = 20
3. Menghitung e bilangan relatif prima terhadap n
   Misalkan jika e = 2 maka gcd(2, 20) = 2 --> bukan
   Misalkan jika e = 3 maka gcd(3, 20) = 1 --> ya, maka **e = 3**
4. Menghitung d
   Misalkan jika m = 0 maka d = (1 + 0. 20)/ 3 = 1/3 ---> bukan
   Misalkan jika m = 1 maka d = (1+(1.20)=> 21 / 3 = **7** ---> ya

Dari perhitungan diatas didapatkan kunci *Public* (33, 3) dan kunci *Private* (7), dimana kunci public (33, 3) akan dikirimkan kepada Nurdin.

b. Enkripsi

Misalkan Nurdin menginginkan untuuk mengirim pesan yaitu 20 kepada Marta

Enkripsi 20
C = 20 ^ 3 mod 33
C = 14

c. Dekripsi

Marta yang menerima ciphertext 14 akan mendekripsi dengan menggunakan kunci m dan d.

Dekripsi ciphertext angka 8
M = 14 ^ 7 (mod 33)
M = 105413504 (mod 33)
M = 20

Dekripsi yang dilakukan Marta dengan menggunakan kunci public m =33 dan kunci private = 7 menghasilkan plaintext 20 (sama seperti yang dikirim oleh Nurdin)

Penggunaan Algoritma ElGamal

a. Membuat pasangan kunci

Marni membangkitkan pasangan kunci dengan memilih p = 11, g = 7, dan x = 3. Kemudian p, g, x digunakan untuk menghitung y :

y= g^x mod p
y= 7^3 mod 11 = 343 mod 11 = 2

Jadi kunci *public*nya y = 2, g = 7, p = 11 dan kunci *private*nya x =3

b. Enkripsi

Misalkan Danar ingin mengirim plainteks m = 10 dengan ketentuan nilai m masih berada di dalam range 0 dan 11 – 1. Danar memilih bilangan acak k = 5 dengan ketentuan nilai k masih berada di dalam range 0, 11 – 1. Kemudian Danar menghitung

a = g^k mod p = 7^5 mod 11
a = 16807 mod 11 = **10**
b = y^k m mod p = (2^5) 10 mod 11
b= 320 mod 11 =**1**

Jadi, cipherteks yang dihasilkan adalah pasangan(10, 1). Danar mengirim cipherteks ini ke Marni.

c. Dekripsi

Marni mendeskripsi cipherteks dari Danar dengan melakukan cara sebagai berikut :

1/(a^x) = a^(p – 1 – x) mod p = 10^(11-1-3) mod 11
=>10000000 mod 11 = **10**
m = b/(a^x)mod p = 1 (10) mod 11 = **10**

Plainteks yang didekripsikan adalah 10, sama dengan plainteks yang dikirim oleh Danar yaitu 10.

KESIMPULAN

RSA merupakan contoh yang powerful dan cukup aman dari kriptografi Kunci *Public*. Berdasarkan matematika, proses yang digunakan berdasarkan fungsi-fungsi trap-door satu arah. Sehingga melakukan enkripsi dengan menggunakan public kunci sangat mudah bagi semua orang, namun proses dekripsinya sangat sulit, walaupun menggunakan supercomputer dan ribuan tahun, tidak dapat mendekripsi pesan tanpa mempunyai kunci *private*.

Dalam implementasinya pemilihan p dan q untuk mendapatkan p*q = M haruslah sebuah bilangan yang sangat besar sehingga sulit untuk melakukan pemfaktoran bilangan prima.

DAFTAR PUSTAKA

A. Menezes, P. Van. Oorschot, 1996, Handbook of Applied Cryptography, CRC Press, Inc.

Childs, Lindsay N. A Concrete Introduction to Higher Algebra. Undergraduate Texts in Mathematics. Springer-Verlaag: New York, 2000.

Schneier, B. Applied Cryptography, 2nd Ed. John Wiley & Sons, Inc: Canada, 1996.

Rivest R.L., Shamir A., Adleman L. "A Method for Obtaining Digital Signatures and Public-Kunci Cryptosystems. MIT: Massachusetts. 1977.

Rizaldi Munir, *Diktat Kuliah IF5054 Kriptografi*, Program Studi Teknik Informatika Institut Teknologi Bandung, 2006.

http://www.iusmentis.com/technology/encryption

http://agcrypt.wordpress.com/2008/02/25/elgamal-algorithm/

http://www.ics.uci.edu/~goodrich/teach/ics247/W03/notes/elgamal.pdf

http://people.forbes.com/profile/taher-elgamal/40757

http://www.nndb.com/people/136/000024064/

www.cacr.math.uwaterloo.ca/hac