ALGORITMA ASYMETRIC KEY DALAM KEAMANAN EMAIL

Andrei Dharma Kusuma/13508009

Informatika Institut Teknologi Bandung
Jalan Ganesha No 10, Bandung
e-mail: andrei_dharma@hotmail.com

ABSTRAK

Makalah ini berisi tentang bagaimana kriptografi diterapkan dalam persuratan elektronik atau lebih dikenal dengan email. Kriptografi yang akan dibahas disini pada dasarnya menggunakan fasilitas GPG yang merupakan sebuah implementasi PGP yang *free*/bebas. Disini algoritma yang digunakan dan akan dibahas adalah sebuah algoritma *asymetric key* dari banyak jenis algoritma yang digunakan dalam GnuPG atau GPG

Kata kunci: Keamanan E-mail. Asymetric key, kriptografi

1. PENDAHULUAN

1.1 Kriptografi

Kriptografi atau bahasa inggris nya cryptography, berasal dari bahasa yunani yang berarti "secret writing" atau tulisan rahasia. Definisi secara tertulisnya ialah ilmu dan seni untuk menjaga keamanan pesan.

Kriptografi ini sendiri telah ada sejak 400BC di Yunani, yaitu oleh seorang caesar bernama Chipper. Algoritma yang digunakan memang masih sangat sederhana, yaitu dengan menggeser urutan huruf sebanyak n huruf. Sebagai contoh menggeser alfabet asli sebanyak 3 huruf, akan menghasilkan deretan alfabet baru seperti ini.

pi : A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

ci : **D E F G H I J K L M N O P Q R S T U V W X Y Z A B C**

Masih banyak jenis kriptografi yang ada dan terus berkembang hingga sekarang. Dari kriptografi klasik seperti chipper, sampai kriptografi yang digunakan dalam makalah ini, yaitu *asymmetric key alghoritm,* yang akan dibahas pada subbab selanjutnya.

1.2. Asymmetric key alghoritm

Algoritma ini pada dasarnya mempunyai 2 buah kunci., yaitu sebuah kunci public dan sebuah kunci privat. Kunci-kunci ini pada dasarnya bertujuan untuk mengenkripsi dan mendekripsi *message* atau pesan yang ingin dan/atau dikirim.

Kunci public seperti dari asal kata nya sendiri merupakan sebuah kunci yang di-*publish* atau disebarkan kepada penerima pesan tersebut. Sedangkan kunci privat digunakan untuk mendekripsi file tersebut dan bersifat rahasia, yakni hanya pengirim pesan yang mengetahui nya.

Proses enkripsi dan dekripsi ini sendiri bergantung pada apa jenis algoritma yang digunakan dalam proses kriptografi ini.

2. GPG

2.1 Apa itu GPG?

GPG atau singkatan dari Gnu Privacy Guard merupakan sebuah software atau program kriptografi yang bebas/free.

GPG adalah bagian dari *Free Software Foundatin's GNU software project.*

2.2 Sejarah GPG

GnuPG pertama kali didirikan oleh Werner Koch dan versi pertama nya dikeluarkan pada tahun 1999 bulan september tanggal 7. Perdana Menteri German membiayai dokumentasi dan masuknya ke Microsoft Windows pada tahun 2000. Email enkripsi ini sendiri pertama kali dikembangkan oleh Phil Zimmerman. Versi kedua nya atau versi 2.0 di keluarkan pada 13 November 2006.

2.3 Penggunaan GPG

Walaupun program basic dari GnuPG berupa sebuah *command line interface* atau dengan kata lain hanya berupa tulisan. Namun ada juga GnuPG yang berupa *graphical user interface.*

2.4 Algoritma GPG

GPG atau GnuPG menggunakan banyak macam algoritma, seperti :

- *Block ciphers*
  - CAST5
  - Triple DES
  - AES
  - Blowfish
  - Twofish
- Asymmetric-key ciphers:
  - ElGamal
  - RSA
- Cryptographic hashes:
  - RIPEMD-160
  - MD5
  - SHA-1
  - Tiger
- Digital signatures:
  - DSA

Dalam Makalah ini akan dibahas tentang Asymmetric-key chippers, seperti ElGamal dan RSA.

3. PEMBAHASAN

3.1 Algortima ElGamal

Algoritma ElGamal atau Enkripsi ElGamal terdiri dari tiga komponen, yaitu:

1. Generator kunci
2. Algoritma enkripsi
3. Algoritma dekripsi

3.1.1 Generator Kunci

Generator Kunci merupakan sebuah tahap dimana kita membuat dan mendapat kan dua buah kunci, yaitu sebuah kunci public dan sebuah kunci privat.

Adapun algoritma yang terlibat didalam pembuatan kunci ini adalah sebagai berikut

Tahap pertama adalah pendefinisian variable-variable yang akan digunakan. Pembuatan 3 buah variable G, g, q.

Kemudian dipilih sebuah x dari himpunan {0,…,q-1}, dan dicari

$$h = g^x$$

Variable h ini disimpan dan dijadikan sebuah kunci public bersama-sama dengan variable G,q,g sehingga terdiri dari sebuah himpunan {G,q,g,h}

Kunci private yang dipunya adalah x dan harus dijaga baik-baik kerahasiaannya.

3.1.2 Algoritma Enkripsi

Untuk mengenkripsi sebuah pesan, dilakukan beberapa tahap algoritmik yang tidak terlalu rumit.

Pertama-tama dipilih sebuah variable y dari himpunan yang sudah dibuat yaitu {0,….,q-1}, lalu dicari sebuah c1 sehingga

$$c_1 = g^y$$

Setelah itu dihitung kembali sebuah variable rahasia s sehingga

$$s = h^y$$

Langkah di atas dapan dihitung terlebih dahulu.

Kemudian peng-enkripsi merubah pesan rahasia nya m menjadi elemen m' dari G

Langkah selanjutnya dalam proses enkripsi ini adalah menghitung c2 sehingga

$$c_2 = m' \cdot s$$

Kemudian Chiperteks (c_1, c_2) dikirim kepada pembuat kunci atau si pemegang kunci privat untuk di dekripsi

3.1.3 Algoritma Dekripsi

Algoritma dekripsi ini bekerja sebagai berikut, untuk mendekripsi sebuah chiperteks (c_1, c_2) dengan kunci privat x, pertama-tama dilakukan sebagai berikut

1. Pen-dekripsi menghitung
   $$s = c_1^x$$
2. kemudian dihitung
   $$m' = c_2 \cdot s^{-1}$$
   Yang kemudian dikembalikan menjadi sebuah teks awal m

Algoritma dekripsi ini menghasilkan teks yang diinginkan, karena

$$c_2 \cdot s^{-1} = m' \cdot h^y \cdot (g^{xy})^{-1} = m' \cdot g^{xy} \cdot g^{-xy} = m'.$$

3.1.4 Keamanan

Keamanan dari algoritma ElGamal tergantung pada grup G dan juga yang digunakan dalam pesan tersebut.

Jika penghitungan asumsi Diffie-Hellman atau *Computation Diffle-Hellman assumption* mendasari grup G, makan fungsi enkripsi nya adalah satu arah.

Jika keputusan asumsi Diffie-Hellman atau *Decision Diffle-Hellman assumption(DDH)* berada pada G, maka

algortima ElGamal ini merupakan sebuah *semantic security.*

Enkripsi ElGamal adalah sebuah algoritma yang lunak, lunak dalam arti tidak aman dari beberapa serangan chiperteks, sebagai contoh, diberikan sebuah enkripsi (c_1,c_2) dari beberapa (mungkin juga tidak diketahui) pesan *m,* dapat dibuat sebuah salinan enkripsi yang *valid* $(c_1,2c_2)$ dari pesan 2m.

Untuk mendapatkan keamanan chiperteks yang lebih terjamin, pola algoritma nya harus dimodifikasi. Bergantung pada modifikasi yang dilakukan, *DDH assumption* mungkin atau mungkin tidak dibutuhkan.

Skema lain yang berhubungan dengan ElGamal dimana ada keamanan terhadap serangan chiperteks juga telah diajukan. Sistem Cramer-Shoup aman dari beberapa serangan chiperteks dengan asumsi DDH bertahan untuk G. Bukti tersebut tidak menggunakan *random oracle model.* Skema lain yang diajukan adalah DHAES dimana bukti nya membutuhkan asumsi yang lebih lemah dari asumsi DDH.

3.1.5 **Efisiensi**

Enkripsi ElGamal adalah sebuah probabilitas, yang berarti sebuah teks dapat di enkripsi menjadi banyak chiperteks, dengan konsekuensi bahwa enkripsi ElGamal umumnya menghasilkan 2:1 pembengkakan ukuran dari teks biasa menjadi chiperteks.

Enkripsi dari ElGamal membutuhkan dua fungsi pangkat; bagaimanapun juga perpangkatan ini tidak bergantung dari pesan/teks dan dapat dihitung terlebih dahulu jika diperlukan.

Dekripsi dengan hanya satu perpangkatan:

Pembagian dengan s dapat dihindarkan dengan menggunakan cara lain dari metode alternative untuk dekripsi. Untuk mendekripsi sebuah chiperteks (c_1, c_2) dengan kunci privat x,
Dihitung
$$s' = c_1^{q-x}$$
s' adalah invers dari s. Hal ini terjadi karena konsekuensi dari teori Langrange, karena
$$s \cdot s' = g^x \cdot g^{q-x} = g^q = 1$$
Kemudian dihitung
$$m' = c_2 \cdot s'$$
Dimana kemudian dikembalikan lagi menjadi sebuah teks biasa m.
Proses algoritma dekripsi ini menghasilkan pesan yang diinginkan karena
$$c_2 \cdot s' = c_2 \cdot s^{-1} = m'$$

3.2 Algoritma RSA

Dalam kriptografi, RSA (yang merupakan singkatan dari Rivest, Shamir, dan Adleman yang pertama kali menjelaskan nya) adalah sebuah algoritma untuk kriptografi kunci public. Algoritma ini adalah algoritma pertama yang dikenal dan diketahui cocok untuk enkripsi, dan juga merupakan sebuah kemajuan utama yang pertama dalam kriptografi kunci public. RSA lebhi digunakan sebagai alat perdagangan elektronik dan dipercaya aman untuk implementasi *up-to-date.*

Sama seperti ElGamal, ada tiga tahap dalam algoritma seperti ini, yaitu pembuatan kunci, proses enkripsi dan proses dekripsi. Untuk yang pertama akan dibahas mengenai pembuatan kunci menggunakan metode RSA

3.2.1 Pembuatan Kunci

Sama seperti algoritma asimetri yang lainnya, algoritma RSA juga terdiri dari kunci public dan kunci privat. Dimana kunci public dapat diketahui siapa saja dan digunakan untuk mengenkripsi pesan. Pesan yang di enkripsi dengan kunci public hanya dapat di dekripsi dengan kunci privat seperti yang telah dibahas sebelum nya. Berikut adalah cara mendapatkan kunci public dan kunci privat dengan menggunakan algoritma RSA.

1. Pilih dua bilangan prima p dan q
   a. Untuk tujuan keamanan, bilangan bulat p dan q seharusnya dipilih secara tidak berpola dan dengan jumlah bit yang hamper sama.
2. Hitung n = pq
   n digunakan sebagai modulus dari kedua p dan q
3. Hitung $\varphi(p,q) = (p-1)(q-1)$.
4. Pilih sebuah bilangan e sehingga $1< e < \varphi(pq)$, dan e dan $\varphi(pq)$ saling prima
   e disebarkan sebagai kunci public
5. Cari d sehingga
   $$de \equiv 1 \pmod{\varphi(pq)}$$
   d disimpan sebagai kunci privat kita

Kunci public terdiri dari modulus n dan eksponen kan kunci privat terdiri dari modulus n dan eksponen d yang harus dirahasiakan. Ada pula algoritma lain yang menggunakan variasi-variasi lain, seperti dalam PKCS#1 v2.0 dan PKCS#1 v2.1 menggunakan fungsi
$$\lambda(n) = \operatorname{lcm}(p-1, q-1)$$
Dimana lcm adalah least common multiple / dalam bahasa Indonesia nya adalah KPK/ kelipatan persekutuan terkecil dan bukan menggunakan
$$\varphi(pq) = (p-1)(q-1)$$

3.2.2 Proses Enkripsi

Kunci Public (n,e) diberikan kepada penerima dengan tetap merahasiakan kunci privat.
Pengenkripsi kemudian mengubah pesan M menjadi bilangan 0< m < n .

Kemudian hitung chiperteks c sehingga

$$m^e \equiv c \pmod{n}.$$

Hasil c yang didapat kemudian dikirim untuk di dekripsi

3.2.3 Dekripsi

Dalam proses ini, atau dalam proses dekripsi ini, penerima dapat mengembalikan chiperteks c menjadi m menggunakan kunci privat d nya dengan perhitungan ato persamaan sebagai berikut

$$c^d \equiv m \pmod{n}.$$

Dengan didapat nya nilai m, penerima dapat mengembalikan pesan asli M dengan membalikkan *padding scheme* .

Prosedur dekripsi di atas berjalan karena

$$c^d \equiv (m^e)^d \equiv m^{ed} \pmod{n}$$

Sekarang, sejak

$$ed = 1 + k\varphi(n)$$

Dan

$$m^{ed} \equiv m^{1+k\varphi(n)} \equiv m(m^{\varphi(n)})^k \equiv m \pmod{n}$$

Kongruen terakhir secara langsung mengikuti teori Euler ketika m relative prima terhadap n. Hal ini dapat ditunjukkan bahwa persamaan tersebut memiliki semua m dengan argument kongruensi dan teori *Chinese remainder.*

Hal ini menunjukkan bahwa pesan asli telah didapatkan:

$$c^d \equiv m \pmod{n}.$$

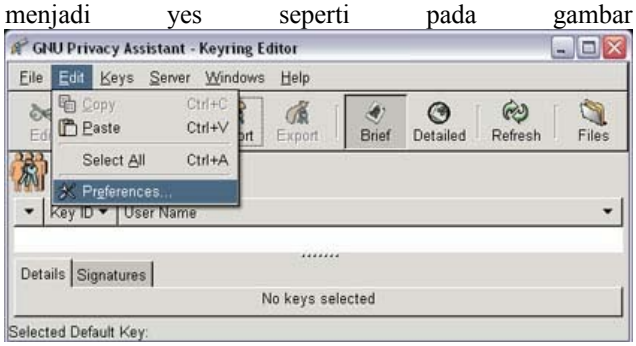
4. APLIKASI

Di sub-bab ini akan dibahas mengenai aplikasi GPG dalam bentuk sebuah software.

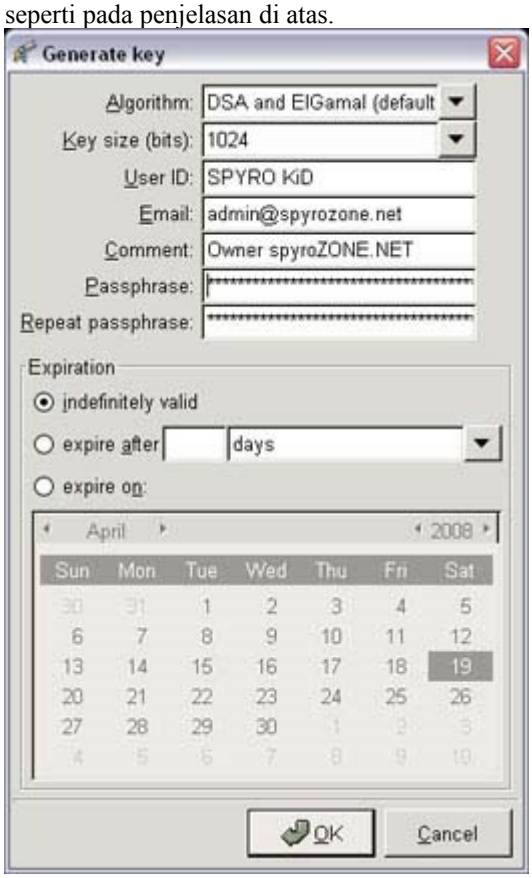
Sebelum nya aplikasi GPG ini dapat di download di situs http://gpg4win.org/ untuk OS Windows.

Setelah mendownload, Install program seperti biasa dan jalankan program tersebut.

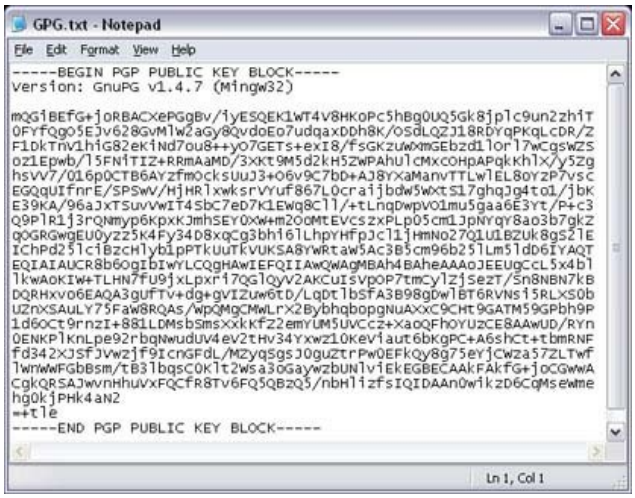
Langkah pertama yang harus dilakukan adalah membuat sebuah atau sepasang kunci public dan kunci privat. Sebelum membuat kunci public dan kunci privat, ada hal-hal terlebih dahulu yang harus dilakukan. Buka Edit >> Preferences, lalu ganti pada bagian Advance Mode menjadi yes seperti pada gambar



Kemudian pilih Key >> New Keys, generate keys dengan algoritma diset default. Dengan menset defaul maka sama saja dengan menggunakan algoritma ElGamal seperti pada penjelasan di atas.



Setelah digenereate, dan didapatkan public key, klik kunci publik baru anda tersebut dan klik tombol export, pilih direktori penyimpanan dan public key yang anda miliki akan berbentuk sebagai berikut



Jadilah kunci publik anda dan dapat disebarluaskan.

Tahapan selanjutnya adalah bagaimana cara mengenkripsi sebuah file, caranya cukup mudah dengan memanfaatkan algoritma dan program yang sudah ada. Pengguna program tinggal membuat sebuah e-mail/pesan kemudian mengenkripsi menggunakan program GPG yang telah di unduh dan penerima mengenkripsi dengan kunci privat yang ia miliki.

5. KESIMPULAN

Penggunaan Kriptografi sebagai bentuk pengamanan, terutama email sangat berperan dan sudah dapat digunakan di jama sekarang ini.

Penggunaan GPG sebagai sebuah Open PGP {Privacy Guard Protocol) sangat bervariasi hingga sekarang ini.

Pemanfaatan berbagai jenis algoritma diyakinin ampuh dalam menjaga keamanan suatu pesan atau e-mail.

REFERENSI

[1] http://en.wikipedia.org/wiki/Public-key_cryptography
[2] http://en.wikipedia.org/wiki/GNU_Privacy_Guard
[4] http://www.informatika.org/~rinaldi/Matdis/2008-2009/Teori%20Bilangan.ppt
[5] http://en.wikipedia.org/wiki/GNU_Privacy_Guard