Analisa Algoritma Digital Signature Cramer-Shoup

Simon H S - 13504056 [1)]

1) Program Studi Teknik Informatika, Sekolah Teknik Elektro dan Informatika, ITB Bandung 40135,
email: if14056@students.if.itb.ac.id

Abstract *– Makalah ini membahas mengenai algoritma digital signature Cramer-Shoup yang merupakan algoritma hasil pengembangan dari El-Gamal dan menggunakan asumsi Strong RSA. Pada makalah ini juga dibahas mengenai cara kerja Cramer-Shoup cryptosystem, analisa keamanan dari algoritma tersebut dan perbandingan algoritma Cramer-Shoup terhadap algoritma digital signature yang sudah cukup terkenal yaitu RSA.*

Kata Kunci*: Algoritma Cramer-Shoup, Digital Signature*

1. PENDAHULUAN

1.1 Kriptografi kunci public

Proses pengamanan pesan melalui teknik kriptografi mempunyai sejarah yang sangat panjang, dimana Julius Caesar diyakini sebagai orang yang berpengaruh besar di dalamnya sebagai penemu salah satu sistem kriptografi paling tua, yang dulunya digunakan untuk mengirimkan pesan kepada jendral-jendralnya.

Sejak saat itu, dunia kriptografi pun mengalami perkembangan yang cukup pesat. Namun, menurut sejarah, terdapat satu permasalahan besar yang cukup menghalangi perkembangan kriptografi[2]. Masalah tersebut adalah manajemen kunci (*key management*).

Dalam dunia kriptografi, kunci merupakan sekumpulan nilai yang dapat digunakan oleh suatu algoritma untuk menyembunyikan informasi, melindungi informasi tersebut, dan juga membatasi orang-orang yang dapat menerima informasi tersebut, yaitu orang-orang yang mempunyai kunci. Untuk itu, terminologi manajemen kunci merupakan suatu metoda penyebaran kunci yang aman kepada orang-orang yang tepat kapanpun dan dimanapun orang tersebut berada.

Masalah ini kemudian dipecahkan oleh Diffie dan Heliman dengan mengusulkan kriptografi nirsimetri (*asymmetric cryptosystem*) yang memungkinkan pengguna berkomunikasi secara aman tanpa perlu berbagi kunci rahasia [1]. Kriptografi nirsimetri ini dikenal juga dengan nama kriptografi kunci publik (*public key cryptography*).

Kriptografi kunci pubik kemudian berkembang menjadi besar dan menjadi revolusi baru dalam bidang kriptografi. Tidak seperti kriptografi kunci simeteri yang didasarkan pada substitusi dan permutasi, kriptografi kunci publik menggunakan fungsi matematika sebagai dasarnya. Jika kekuatan kriptografi kunci simetri terletak pada panjang kuncinya yang membutuhkan usaha sangat besar dalam menemukan kunci, maka kriptografi kunci publik kekuatannya terletak pada sulitnya memecahkan permasalahan matematis seperti pemfaktoran dan logaritma diskrit. Algoritma kunci publik dapat diaplikasikan menjadi 3 kategori yang salah satunya adalah tanda tangan digital.

1.2 Tanda tangan digital

Sejak berabad-abad yang lalu, tanda-tangan digunakan untuk membuktikan otentikasi dari suatu dokumen kertas (cth. surat, piagam, ijazah, buku, dan sebagainya). Fungsi tanda-tangan pada dokumen kertas ini juga diterapkan untuk otentikasi pada data digital seperti pesan yang dikirim melalui saluran komunikasi ataupun dokumen elektronis yang disimpan di dalam memori komputer. Tanda tangan pada data digital ini disebut sebagai tanda tangan digital (*digital signature*).

Salah satu perbedaan yang cukup besar antara tanda tangan digital dengan tanda tangan pada dokumen kertas adalah pada dokumen digital, tanda tangan yang dimaksud adalah suatu nilai kriptografis yang bergantung dari isi pesan dan pengirim pesan,sedangkan pada tanda tangan pada dokumen kertas bergantung hanya kepada pengirimnya saja (dapat dilihat dari bentuk tanda tangan yang selalu sama apapun isi dokumennya).

Sebuah tanda tangan digital, harus memenuhi beberapa properti berikut ini:

a. Kebenaran
   Tanda tangan yang telah di-*sign* oleh pengirimnya haruslah dapat di-*verify* kembali.
b. Tidak dapat dipalsukan
   Sebuah tanda tangan tidak boleh bisa dipalsukan. Seorang penyerang dapat menerima banyak dokumen yang ditanda-tangani dan apabila ia berhasil membuat sebuah dokumen dan sebuah string yang lolos verifikasi, maka algoritma tersebut akan dinyatakan tidak aman.

2. ALGORITMA CRAMER-SHOUP

Algoritma Cramer-Shoup merupakan sebuah algoritma kunci publik, yang dikembangkan oleh Ronald Cramer dan Victor Shoup pada tahun 1998. Dirancang sebagai algoritma untuk pembuatan atanda tangan digital, algoritma ini merupakan algoritma efisien pertama yang terbukti tahan terhadap *adaptive chosen chipertext attack* dengan menggunakan asumsi RSA kuat (*strong RSA assumption*) [3]. Algoritma ini merupakan pengembangan dari algoritma ElGamal dan juga menggunakan Diffie-Hellman Decision Problem.

2.1 Asumsi RSA

Permasalahan RSA adalah sebagai berikut:

Misalnya terdapat bilangan n yang merupakan hasil pembangkitan dari RSA, eksponen r dan bilangan random $z \in Z_n^*$, temukanlah $y \in Z_n^*$ sedemikian sehingga $y^r = z$.

Asumsi RSA adalah asumsi dimana permasalahan ini sulit atau tidak mungkin dipecahkan.

Sedangkan asumsi RSA kuat adalah sebagai berikut:

Misalnya terdapat bilangan *n* yang merupakan hasil pembangkitan dari RSA, temukan bilangan $r > 1$ dan $y \in Z_n^*$ sedemikian sehingga $y^r = z$.

Perhatikan bahwa pada asumsi RSA biasa, *r* tidak bergantung pada *z*, sedangkan pada asumsi RSA kuat *r* dapat bergantung pada *z*. Sama seperti asumsi RSA, asumsi RSA kuat adalah dimana permasalahan di atas adalah sulit atau tidak mungkin dipecahkan.

2.2 Fungsi Hash bebas Collition

Sebuah fungsi hash dikatakan sebagai fungsi hash yang tahan Collition apabila nilai hash dibangun secara random, tidak akan ada duah buah nilai *x* dan *y* sehingga *H(x) = H(y)*. Apabila hal tersebut terjadi (terdapat 2 buah nilai yang menghasilkan sebuah nilai hash yang sama) maka fungsi hash tersebut dikatakan berasarl dari keluarga hash *target collition.*

2.3 Diffie-Hellman Decission Problem

Protokol ini dikembangkan oleh Diffie dan Hellman pada tahun 1976. Keamanan algoritma ini berdasarkan fakta sulitnya menghitung logaritma diskrit.

Awalnya Alice dan Bob menyepakati bilangan prima yang besar yaitu *n* dan *g*, sedemikian sehingga *g*< *n*. Bilangan *n* dan *g* ini tidak perlu dirahasiakan.

Bahkan Alice dan Bob dapat membicarakannya melalui saluran telepon yang tidak aman sekalipun

Terdapat beberapa formulasi yang equivalen dalam Algoritma Diffie-Hallman. Dalam algoritma ini, formulasi yang digunakan adalah sebagai berikut:

Terdapat *G* yang merupakan bilangan prima besar berorde *q*, dan terdapat 2 buah distribusi sebagai berikut:

a. Distribusi R terhadap pasangan 4 bilangan acak $(g_1, g_2, u_1, u_2) \epsilon\, G^4$
b. Distribusi D terhadap pasangan 4 bilangan acak $(g_1, g_2, u_1, u_2) \epsilon\, G^4$dan $u_1 = g_1^r$ dan $u_2 = g_2^r$

Algoritma yang dapat menyelesaikan permasalahan Diffie-Helman merupakan suatu tes statistik yang dapat secara efektif menyelesaikan kedua buah distribusi tersebut, menghasilkan keluaran 0 atau 1, dan tidak boleh terdapat perbedaan bilangan yang berarti antara (a) menghasilkan bilangan 1 dengan memasukkan bilangan hasil keluaran distribusi R atau menghasilkan bilangan 1 dengan memasukkan bilangan hasil keluaran distribusi D.
Permasalahan Diffie-Hellman akan sangat sulit dipecahkan apabila tidak terdapat tes statistikal terhadap tes polinomial.

2.4 *Adaptive Chosen Chipertext Attack*

Adaptive Chosen Chipertext Attack atau sering disingkat dengan CCA2 merupakan bentuk yang lebih dinamis dari *chosen chipertext attack* dimana penyerang mengirimkan sejumlah *chipertext* untuk didekripsi, dan kemudian menggunakan hasil dekripsi tersebut untuk memilih *subsequent* dari *chipertext*.

Tujuan dari serangan ini adalah untuk mengetahui informasi yang tersimpan di dalam *chipertext* sedikit demi sedikit, ataupun untuk mencari tahu kunci yang digunakan di dalam dekripsi tersebut. Dalam kriptografi kunci publik, serangan ini biasanya dapat diaplikasikan apabila algoritma tersebut mempunyai atribut *chipertext malleability*. Atribut ini menyatakan bahwa *chipertext* tersebut dapat diubah dengan suatu cara sehingga ketika dilakukan dekripsi *chipertext* tersebut mempunyai efek yang bisa diprediksi.

Serangan ini dulunya hanya dianggap sebagai masalah yang ada di level teoritis sampai pada tahun 1998 dimana Daniel Bleichenbacher, seorang ilmuwan yang berasal dari Bell Laboratories, melakukan demonstrasi serangan terhadap protokol SSL yang waktu itu digunakan oleh ribuan website di dunia. Dia melakukan serangan dengan mengirimkan beberapa

juta chipertext percobaan.

Sebuah Algoritma kriptografi dinyatakan aman terhadap serangan *Adaptive Chosen Chipertext Attack* apabila hasil yang didapat dari setiap serangan polinomial berbasis waktu dapat diabaikan (sama seperti fungsi dalam parameter keamanan).

2.5 Skema Dasar

Skema Cramer-Shoup diparameterisasi oleh 2 parameter keamanan, *l* dan *l'*, dimana *l* + 1 < l'.
Salah satu pilihan yang masuk akal adalah *l* = 160 dan *l'* = 512[4]. Skema ini juga menggunakan fungsi hash yang tahan terhadap *collition H* dengan hasil fungsi adalah bilangan posifif lebih kecil dari 2^l. Fungsi yang direkomendasikan adalah SHA-1[4].
Untuk bilangan positif *n*, QR_n adalah bagian dari Z_n^* dipangkat 2.

Skema Cramer-Shoup terdiri dari 3 algoritma:

a. Pembangkit kunci
   Untuk membangkitkan kunci, pilih dua buah bilangan random *l'* bit *p* dan *q*, dimana

   $$p = 2p' + 1 \text{ dan } q = 2q' + 1$$

   dengan *p'* dan *q'* adalah bilangan prima.

   Kemudian hitung bilangan *n* dimana

   $$n = pq$$

   Lalu, pilih lagi dua buah bilangan $h, x\ \varepsilon\ QR_n$ dan sebuah bilangan random prima *e'* sebesar $(l + 1)$-bit.

   Maka kunci publiknya adalah: (n, h, x, e')
   Dan kunci privatnya adalah: (p, q)

b. Pembangkit tanda tangan
   Untuk menandatangani dokumen m (berbentuk bit string), pilih bilangan prima *e* dimana $e \neq e'$ sebesar $(l+1)$ bit, dan bilangan random $y' \epsilon\ QR_n$ sehingga

   $$y^e = xh^{H(x')} \text{ dan } (y')^{e'} = x'h^{H(m)}.$$

   Perhatikan bahwa *y* dapat dihitung dengan mencari faktor dari *n* dalam kunci privat.

   Maka tanda tangannya adalah: (e, y, y').

c. Verifikasi tanda tangan
   Untuk memverifikasi tanda tangan di dalam dokumen *m*, pertama kali dilakukan adalah dengan memeriksa bilangan *e* adalah bilangan (*l*+1)-bit yang berbeda dengan *e'*.

   Setelah memastkan kedua bilangan tidak sama, hitunglah

   $$x' = (y')^{e'} h^{-H(m)}.$$

   Setelah nilai *x'* didapatkan, periksalah nilai *x*

   $$x = y^e h^{-H(x')}$$

3. ANALISIS ALGORITMA CRAMER SHOUP

3.1 Perbandingan Cramer-Shoup dengan RSA

RSA adalah algoritma kunci publik yang masih banyak sekali digunakan. Dalam prosesnya, RSA menggunakan 2 buah bilangan prima yang sangat besar sebagai dasar dalam algoritmanya. Faktor keamanan dari RSA sendiri, seperti yang disebutkan di atas merupakan sulitnya mencari pemfaktoran dari bilangan yang sangat besar. Namun kemudian ditemukan bahwa RSA lemah terhadap serangan *adaptive chosen chipertext* dikarenakan algoritma ini sangat tergantung pada 2 buah bilangan saja.

Cramer-Shoup, di lain pihak, mempunyai keunggulan dalam hal ini. Cramer-Shoup merupakan algoritma yang dibangun untuk dapat bertahan dari serangan tersebut. Dengan adanya fungsi yang tidak hanya tergantung pada modulo, yaitu fungsi $y^e = xh^{H(x')}$ dan $(y')^{e'} = x'h^{H(m)}$, kemungkinan serangan dapat dihilangkan.

Namun, menurut referensi [5], algoritma ini mempunai beberapa kekurangan antara lain:

a. menghasilkan *chipertext* dengan panjang sampai 4 kali lipat dari *plain text* (akibat menggunakan 2 buah bilangan besar sepanjang *plain text*)
b. membutuhkan waktu 2 kali lipat dari algoritma El Gamal yang merupakan asalnya.

3.3 Analisa Keamanan Cramer-Shoup

Algoritma Cramer-Shoup adalah algoritma yang aman terhadap serangan *adaptive chosen message attack* dengan asumsi permasalahan Diffie-Hellman sulit dalam grup G dan asumsi bahwa fungsi *H* bebas dari *collition*.

Untuk membuktikannya, diasumsikan terdapat serangan yang dapat menembus algoritma dan fungsi hash yang ada berasal dari keluarga hash 1 arah, dan menunjukkan bagaimana menggunakan serangan tersebut untuk membangun tes statistikal dalam permasalahan Diffie-Helman.
Untuk tes statistikal, diberikan (g_1, g_2, u_1, u_2) yang

berasal dari distribusi R dan D. Dalam *high level*, konstruksinya berjalan dengan mensimulasikan hubungan kedua distribusi dan mengandung pandangan penyerang dalam sistem, dan bit b yang tersembunyi bahkan dari mata penyerang.

Simulator kemudian diberikan masukan sebagai berikut: (g_1, g_2, u_1, u_2). Simulator kemudian akan menjalankan algoritma pembangkit kunci dengan menggunakan nilai (g_1, g_2) dan memilih :

$$(x_1, x_2, y_1, y_2, z_1, z_2 \epsilon\, Z_q)$$

lalu menghitung

$$c = g_1^{x1} g_2^{x2}\ , d = g_1^{x1} g_2^{x2}, h = g_1^{x1} g_2^{x2}$$

Simulator juga kemudian memilih fungsi H secara random. Kunci publik yang diketahui penyerang adalah (g_1, g_2, c, d, h, H), sementara simulator mengetahui $(x_1, x_2, y_1, y_2, z_1, z_2 \epsilon\, Z_q)$.

Simulator kemudian menjawab query dekripsi seperti serangan sebenarnya, kecuali simulator menghitung

$$m = e/(u_1^{z1} u_2^{z2})$$

Lalu simulasi enkripsinya adalah:
Diberikan m_0, m_1, simulator memilih bilangan random $b \epsilon \{0,1\}$ dan menghitung

$$e = u_1^{z1} u_2^{z2} m_b, \alpha = H(u_1, u_2, e), v = u_1^{x1+y1\alpha} u_2^{x2+y2\alpha}$$

Dan menghasilkan:

$$(u_1, u_2, e, v)$$

Dengan demikian, kita dapat melihat bahwa ketika input dari simulator dating dari D, output dari algoritma adalah chipertext yang sempurna. Namun, ketika input dari simulator berasal dari R, output dari algoritma tidak akan layak dalam kasus $log_{g1} u_1 \neq log_{g2} u_2$. Dengan demikian, maka akan terdapat bukti penyusupan.

4. KESIMPULAN

Algoritma Cramer-Shoup merupakan algoritma yang terbukti aman terhadap serangan *Adaptive Chosen Chipertext Attack* sehingga disebut paling aman dalam dunia kriptografi kunci publik. Namun, algoritma ini mempunyai kelemahan yang diantaranya dari segi hasil enkripsi yang menghasilkan chipertext yang jauh lebih besar dan juga waktu enkripsi yang lebih lama. Algoritma ini masih tergolong muda (dipublikasikan pada tahun 1998) dan penelitian terhadapnya masih berjalan.

DAFTAR REFERENSI

[1] Munir, Rinaldi, *Diktat Kuliah IF5054 Kriptografi*, Sekolah Teknik Elektro dan Informatika Institut Teknologi Bandung, 2007

[2] www.entrust.com/resources/pdf/cryptointro.pdf, diakses pada tanggal 10 Januari 2008 pukul 21.00 WIB

[3] http://cryptodox.com/Cramer-Shoup_cryptosystem, diakses pada tanggal 11 Januari 2008 pukul 21.00

[4] http://www.zurich.ibm.com/security/ace/sig.pdf, diakses pada tanggal 11 Januari 2008 pukul 21.00

[5] http://www.schneier.com/crypto-gram-9809.html, diakses pada tanggal 11 Januari 2008 pukul 23.00