Perbandingan Algoritma Kriptografi Kunci Publik RSA, Rabin, dan ElGamal

Maureen Linda Caroline - 13508049
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
if18049@students.if.itb.ac.id

***Abstract*—Ketidakamanannya penyebaran kunci melalui saluran komunikasi membuat para kriptografer berpikir lebih keras untuk menemukan algoritma-algoritma baru yang dapat digunakan tanpa harus mempermasalahkan pengiriman melalui saluran komunikasi yang tidak aman tersebut. Hingga akhirnya pada tahun 1976 muncul suatu sistem kriptografi baru, yaitu kriptografi kunci publik. Hingga saat ini ada tiga algoritma kriptografi kunci publik yang sering digunakan yaitu RSA, ElGamal, dan Rabin. Ketiga algoritma ini memiliki perbedaan dalam prosesnya mulai dari pembangkitan kunci publik dan privatnya hingga proses enkripsi dan dekripsinya. Pada makalah ini, penulis akan mencoba untuk memaparkan perbedaan dari ketiga algoritma tersebut.**

***Index Terms*— Algoritma kunci publik, ElGamal, Rabin, RSA.**

I. PENDAHULUAN

Hingga akhir tahun 1970, kriptografi masih hanya mengenal satu sistem saja, yaitu sistem kriptografi kunci-simetri. Masalah terbesar dalam sistem kriptografi adalah mengenai bagaimana mengirimkan kunci rahasia kepada penerima. Pengiriman kunci rahasia pada saluran publik (telepon, pos, internet) sangat tidak aman. Oleh karena itu, kunci harus dikirim melalui saluran kedua yang benar-benar aman. Saluran kedua tersebut umunya lambat dan mahal.

Karena permasalahan inilah, muncul suatu ide kriptografi kunci-nirsimetri (*asymmetric-key cryptography*) pada tahun 1976. Namun saat itu masih belum ditemukan algoritma kriptografi kunci-nirsimetri yang sesungguhnya hingga pada akhirnya di masa sekarang ini telah banyak algoritma kriptografi kunci publik yang diantaranya adalah algoritma RSA, algoritma Rabin, dan algoritma ElGamal.

RSA adalah algoritma kunci publik yang paling popular. Algoritma RSA dibuat oleh tiga orang peneliti dari MIT (*Massachussets Institute of Technology*), yaitu Ron Rivest, Adi Shamir, dan Len Adleman pada tahun 1976. Keamanan algoritma RSA terletak pada sulitnya memfaktorkan bilangan yang besar menjadi faktor-faktor prima.

Algoritma kunci publik lainnya adalah algoritma kriptografi kunci publik Rabin. Algoritma ini merupakan varian dari RSA yang ditemukan oleh M.Rabin. Ada perbedaan yang jelas antara RSA dengan Rabin yaitu proses kerjanya.

Algoritma yang tidak kalah terkenalnya dengan RSA adalah algoritma ElGamal. Algoritma ini pada mulanya digunakan untuk digital signature, namun kemudian dimodifikasi sehingga juga bisa digunakan untuk enkripsi dan dekripsi. Keamanan algoritma ini terletak pada sulitnya menghitung logaritma diskrit.

Terlihat dari deskripsi singkat mengenai ketiga algoritma kunci publik, dapat diperkirakan bahwa akan banyak perbedaan lainnya. Oleh karena itu, pada makalah ini, penulis akan melakukan perbandingan antara algoritma RSA, algoritma Rabin, dan algoritma ElGamal.

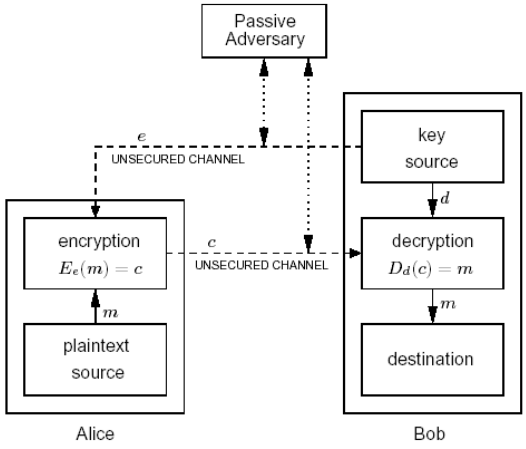
II. DASAR TEORI

2.1. Algoritma kunci publik

Pada kriptografi kunci publik, masing-masing pengirim memiliki sepasang kunci:

1. Kunci publik: untuk mengenkripsi pesan
   $E_e(m) = c$
2. Kunci privat: untuk mendekripsi pesan.
   $D_d(c) = m$

Kunci enkripsi atau kunci publik dapat dikirim melalui saluran yang tidak perlu aman. Saluran yang tidak aman ini mungkin sama dengan saluran yang digunakan untuk mengirimkan chiperteks.



Gambar 1. Contoh enkripsi dan dekripsi kunci publik

Dua keuntungan kriptografi kunci publik adalah tidak diperlukan pengiriman kunci rahasia dan jumlah kunci dapat ditekan.

Pembangkitan sepasang kunci pada kriptografi kunci publik didasarkan pada persoalan *integer* klasik sebagai berikut:

1. Pemfaktoran

   Diberikan bilangan bulat *n*. Faktorkan *n* menjadi faktor primanya.

   Contoh: $10 = 2 * 5$

   $60 = 2 * 2 * 3 * 5$

   $252601 = 41 * 61 * 101$

   $2^{13} - 1 = 3391 * 23279 * 65993 * 1868569 * 1066818132868207$

   Semakin besar *n*, semakin sulit memfaktorkan (butuh waktu sangat lama). Algoritma yang menggunakan prinsip ini: *RSA*

2. Logaritma diskrit

   Temukan *x* sedemikian sehingga $a^x \equiv b \pmod n$ → sulit dihitung.

   Contoh: jika $3^x \equiv 15 \pmod{17}$ maka x = 6

   Semakin besar *a*, *b*, dan *n* semakin sulit memfaktorkan (butuh waktu lama). Algoritma yang menggunakan prinsip ini: ElGamal, *DSA*

   Catatan: Persoalan logaritma diskrit adalah kebalikan dari persoalan perpangkatan modular:

   $a^x \bmod n$ → mudah dihitung

2.2. Algoritma RSA

Algoritma RSA ditemukan oleh tiga peneliti dari MIT (*Massachussets Institute of Technology*), yaitu Ron Rivest, Adi Shamir, dan Len Adleman, pada tahun 1976. Keamanan algoritma RSA terletak pada sulitnya memfaktorkan bilangan yang besar menjadi faktor-faktor prima.

a. Pembangkitan kunci

   Langkah-langkah dalam pembangkitan kunci:

   1. Pilih dua bilangan prima, *p* dan *q* (rahasia)
   2. Hitung $n = pq$
   3. H itung $\phi(n) = (p - 1)(q - 1)$
   4. Pilih sebuah bilangan bulat *e* untuk kunci publik, sebut, *e* relatif prima terhadap $\phi(n)$
   5. Hitung kunci dekripsi, d, dengan persamaan $ed \equiv 1 \pmod{\phi(n)}$ atau $d \equiv e^{-1} \bmod (\phi(n))$

   Hasil dari algoritma di atas:

   - Kunci publik adalah pasangan (*e*, *n*)
   - Kunci privat adalah pasangan (*d*, *n*)

b. Metode enkripsi

   Langkah-langkah dalam mengenkripsi pesan:

   1. Nyatakan pesan menjadi blok-blok plainteks: m_1, m_2, m_3, … (syarat: $0 < m_i < n - 1$)
   2. Hitung blok cipherteks c_i untuk blok plainteks p_i dengan persamaan

      $c_i = m_i^e \bmod n$

      yang dalam hal ini, *e* adalah kunci publik

c. Metode dekripsi

   Proses dekripsi dilakukan dengan menggunakan persamaan

   $m_i = c_i^d \bmod n$,

   yang dalam hal ini, *d* adalah kunci privat.

2.3. Algoritma ElGamal

Algoritma ElGamal dibuat oleh Taher Elgamal pada tahun 1985. Keamanan algoritma ini terletak pada sulitnya menghitung logaritma diskrit.

Masalah logaritma diskrit adalah jika *p* adalah bilangan prima dan g dan y adalah sebarang bilangan bulat, carilah *x* sedemikian sehingga

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

a. Pembangkitan kunci

   Langkah-langkah dalam 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 $g < p$ dan $1 \le x \le p - 2$
   3. Hitung $y = g^x \bmod p$.

   Hasil dari algoritma ini:

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

b. Metode enkripsi

   Langkah-langkah dalam mengenkripsi pesan:

   1. Susun plainteks menjadi blok-blok m_1, m_2, …, (nilai setiap blok di dalam selang $[0, p - 1]$.
   2. Pilih bilangan acak *k*, yang dalam hal ini $1 \le k \le p - 2$.
   3. Setiap blok *m* dienkripsi dengan rumus

      $a = g^k \bmod p$

      $b = y^k m \bmod p$

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

c. Metode dekripsi

   Langkah-langkah dalam mendekripsi pesan:

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

2.4. Algoritma Rabin

Algoritma Rabin pertama kali diperkenalkan pada tahun 1979 oleh Michael O. Rabin. Algoritma Rabin merupakan salah satu sistem kriptografi asimetris yang kemampuan sekuritasnya dibuktikan secara matematik mengingat metode pemfaktoran bilangan secara cepat sampai saat ini belum terpecahkan.

a. Pembangkitan kunci

   Sama seperti sistem kriptografi asimetri lainnya, Rabin juga menggunakan sistem kunci publik dan kunci privat. Kunci publik nantinya akan digunakan

pada prises enkripsi dan dapat diketahui oleh semua pihak (tidak rahasia), sementara kunci privat digunakan oleh poha penerima pesan untuk dekripsi dan bersifat rahasia.

Algoritma pembangkitan kuncinya adalah sebagai berikut:

1. Pilih dua buah bilangan prima besar sebarang yang saling berbeda (p dan q).
2. Hitung n = p.q
   n adalah kunci publik. Bilangan prima p dan q adalah kunci privat.

Untuk mengenkripsi pesan hanya dibutuhkan kunci publik n, sedangkan untuk dekripsi, dibutuhkan bilangan p dan q sebagai kunci privat.

b. Metode enkripsi

Teknik Rabin merupakan algoritma kriptografi kunci publik, maka semuaorang dapat melakukan enkripsi dengan satu kunci publik tertentu, namum proses dekripsi hanya dapat dilakukan dengan menggunakan kunci privat oleh orang yang bersangkutan.

Proses enkripsi pada teknik Rabin sangat sederhana. Proses enkripsi tersebut dapat dituliskan dengan rumus berikut:

$$C = P_2 \bmod n$$

Keterangan: *C* : Cipherteks
P : Plainteks
n : kunci publik

Proses enkripsi yang sederhana ini menyebabkan proses enkripsi teknik Rabin ini dapat dilakukan dengan waktu yang relative singkat karena tidak memiliki proses yang rumit. Kesederhanaan ini merupakan keuntungan yang dimiliki oleh teknik Rabin untuk menghadapi keterbatasan *resource* yang ada pada media kriptografi. Misalnya ada smart card yang memiliki memori terbatas dan membutuhkan waktu proses CPU yang singkat.

c. Metode dekripsi

Proses dekripsi pada teknik Rabin dilakukan dengan menggunakan sebuah rumus sederhana, namun membutuhkan teorema *Chinese remainder*. Teorema ini digunakan untuk mendapatkan plainteks yang benar. Namun yang menjadi poin penting dari teknik ini adalah teknik Rabin tidak menghasilkan jawaban plainteks tunggal. Jawaban yang dihasilkan pada teknik Rabin ini terdiri dari 4 kemungkinan jawaban, tidak menghasilkan satu jawaban yang pasti.

Berikut adalah *pseudo-code* algoritma proses dekripsi teknik kriptografi Rabin:

```
Dekripsi(p,q,C)
{
    a1 ← +(C(p+1)/4) mod p
    a2 ← -(C(p+1)/4) mod p
    b1 ← +(C(q+1)/4) mod q
    b2 ← -(C(q+1)/4) mod q

    // Chinese_Rem adalah fungsi
    yang
    // memanggil fungsi untuk
    Chinese
    // Remainder

    P1 ← Chinese_Rem(a1,b1,p,q)
    P2 ← Chinese_Rem(a1,b2,p,q)
    P3 ← Chinese_Rem(a2,b1,p,q)
    P4 ← Chinese_Rem(a2,b2,p,q)

    return P1,P2,P3,P4
}
```

Teknik Rabin selalu menghasilkan empat kemungkinan hasil, yang diberikan semuanya kepada orang yang melakukan dekripsi terhadap pesan rahasia. Kemudian orang tersebut harus dapat menentukan mana pesan yang sebenarnya dari keempat hasil dekripsi tersebut. Walaupun menghasilkan empat pesan berbeda pada akhirnya, namun penerima pesan dapat memilih pesan yang benar dengan tidak terlalu sulit, karena pesan yang benar seharusnya akan terlihat jelas dibandingkan dengan ketiga hasil dekripsi yang lain.

III. KELEBIHAN DAN KEKURANGAN

1. RSA

Kekuatan algoritma RSA terletak pada tingkat kesulitan dalam memfaktorkan bilangan menjadi faktor-faktor prima, yang dalam hal ini $n = a \times b$. Sekalipun n berhasil difaktorkan menjadi a dan b, maka $\phi(n) = (a - 1)\times(b - 1)$ dapat dihitung. Selanjutnya, karena kunci enkripsi e diumumkan (tidak rahasia), maka kunci dekripsi d dapat dihitung dari persamaan $ed \equiv 1 \pmod{n}$.

Dengan adanya hal ini, maka penemu algoritma ini menyarankan nilai a dan b panjangnya lebih dari 100 digit. Dengan demikian hasil kali $n = a \times b$ akan berukuran lebih dari 200 digit. Dengan digit angka yang sebesar ini, maka usaha untk mencari faktornya membutuhkan waktu komputasi selama 4 miliar tahun dengan asumsi bahwa algoritma pemfaktoran yang digunakan adalah algoritma yang tercepat saat ini dan komputer yang dipakai mempunyai kecepatan 1 milidetik.

Akan tetapi disamping kelebihannya itu, RSA memiliki juga beberapa kekurangan. Algoritma RSA lebih lambat daripada algoritma Rabin ataupun algoritma kriptografi kunci simetri seperti DES dan AES. Dalam prakteknya juga, RSA tidak digunakan untuk mengenkripsi pesan, tetapi mengenkripsi kunci simetri dengan kunci publik penerima pesan.

Serangan terhadap RSA:

1. *Man-in-the-middle attack*

Pihak “di tengah” berlaku sebagai salah satu pihak yang berkomunikasi.

Tujuan: memperoleh pesan rahasia

2. *Chosen-Plaintext Attack*

Tujuan: mempelajari isi pesan.

2. ElGamal

Hingga saat ini belum ada yang berhasil memecahkan algoritma ElGamal. Karena kekomplesitasan algoritma ini, maka penyerangan yang dilakukan dari segala sisi tidak mampu menembus pertahanan algoritma ElGamal ini.

Kelebihan dari algoritma ini adalah pembangkitan kunci yang menggunakan logaritma diskrit dan metode enkripsi dekripsi yang menggunakan proses komputasi yang besar sehingga hasil enkripsinya berukuran dua kali dari ukuran semula. Tetapi kekurangan dari algoritma ini adalah membutuhkan resource yang baik dan processor yang mampu untuk melakukan komputasi yang besar.

3. Rabin

Algoritma Rabin merupakan algoritma varian RSA. Fungsi dasar algoritma Rabin mirip dengan fungsi dasar dari algoritma RSA. Hanya saja komputasi algoritma Rabin lebih sederhana dibandingkan algoritma RSA. Karena kesederhaan komputasinya inilah maka serangan terhadap algoritma inipun lebih banyak, antara lain:

- *Factorization Attack:* Serangan yang dilakukan dengan proses pemfaktoran
- *Chosen-Ciphertext Attack:* Penyerang dapat memilih suatu cipherteks y dan mengkonstruksi plainteks yang terkait sehingga ia bisa menguraikan proses yang terjadi dalam sistem kriptografi tersebut
- *Chosen-Plaintext Attack:* Penyerang dapat memilih suatu plainteks x dan mengkonstruksi cipherteks yang terkait sehingga ia bisa menguraikan proses yang terjadi dalam sistem kriptografi tersebut
- *Encryption Exponent Attack*
- *Decryption Exponent Attack*
- *Plaintext Attack*
- *Modulus Attack*
- *Implementation Attack*

Teknik kriptografi Rabin aman dalam menghadapi serangan-serangan yang bersifat oasif seperti serangan faktroisasi, terlebih lagi jika menggunakan ukuran kunci yang besar. Tetapi teknik ini tidak aman jika mendapat serangan seperti *chosen-ciphertext attack* dan *man-in-the-middle attack.*

IV. ANALISIS

Algoritma rabin merupakan varian dari RSA, sehingga memiliki fungsi dasar yang cukup mirip. Maka analisis perbandingan antara RSA, Rabin, dan ElGamal dilihat dari berbagai aspek yang ada adalah sebagai berikut:

a. Kerumitan proses pembangkitan kunci

Dilihat dari sisi pembangkitan kuncinya, pembangkitan kunci yang paling sulit dari ketiga algoritma tersebut adalah ElGamal karena menggunakan proses logaritma diskrit. Komputasi logaritma diskrit jauh lebih lama dibandingkan dengan pemfaktoran biasa.

Sementara pada RSA pembangkitan kunci sedikit lebih rumit jika dibandingkan dengan Rabin. Pembangkitan kunci pada RSA menggunakan teknik pemfaktoran dimana membutuhkan 3 nilai, yaitu p, q, dan N.

Sedangkan proses pembangkitan kunci yang terlemah adalah Rabin. Pembangkitan kunci pada algoritma Rabin hanya menggunakan 2 nilai saja, yaitu p dan q yang jauh lebih mudah ditebak kuncinya dibandingkan dengan ElGamal atau RSA.

b. Kecepatan proses pembangkitan kunci

Keceptan proses pembangkitan kunci berhubungan erat dengan seberapa rumit pembangkitan kunci. Semakin rumit proses pembangkitan kuncinya, maka akan semakin lama proses komputasinya. Jadi bila dilihat dari aspek kecepatan proses pembangkitan kunci, dari ketiga algoritma tersebut, yang tercepat dalam pembangkitan kunci adalah algoritma Rabin. Algoritma RSA sedikit lebih lama dibandingkan algoritma Rabin dan yang terlama dalam komputasi pembangkitan kunci adalah algoritma Elgamal karena menggunakana proses logaritma diskrit dalam melakukan pembangkitan kunci.

c. Penyimpanan kunci publik dan privat

Kunci publik yang dihasilkan oleh algoritma Elgamal terdiri dari tiga nilai, algoritma RSA terdiri dari dua nila dan algoritma Rabin terdiri dari satu nilai. Kunci privatnya ketiganya sama-sama terdiri dari dua nilai. Melihat banyaknya nilai yang dihasilkan, maka penyimpanan kunci publik dan privat paling dibutuhkan oleh algoritma ElGamal walaupun pada kenyataannya ketiga algoritma membutuhkan penyimpanan kunci publik dan privat karena angka yang dibentuk dari pembangkitan kunci bukanlah ukuran angka yang biasa saja tetapi mungkin angka tersebut merupakan angka 256 bit atau 512 bit sehingga tidak mungkin orang menghapalkannya sekalipun kunci yang dihasilkan hanya satu nilai.

d. Proses enkripsi

Proses enkripsi ElGamal, RSA dan Rabin masing-masing memiliki keunggulan. ElGamal memiliki proses enkripsi yang paling rumit dibandingkan dengan dua algoritma yang lain. Pada proses enkripsi ElGamal, komputasi yang dilakukan

cukup rumit dan membutuhkan *resource* komputer yang besar. Dengan kerumitan seperti ini, maka tingkat keamannya lebih tinggi juga dibandingkan dua algoritma yang lain.

Jika dibandingkan, proses enkripsi teknik Rabin merupakan proses yang paling sederhana dari ketiganya. Kecepatan proses enkripsi RSA jelas jauh lebih baik dibandingkan dengan RSA dan ElGamal karena menggunakan komputasi yang sederhana. Selain itu juga untuk *resource* yang terbatas, Rabin lebih unggul.

Jadi jika diperingkatkan antara ketiga algoritma ini, mka algoritma ElGamal akan menempati peringkat pertama karena waktu komputasinya yang lebih lama dibandingkan dengan dua algoritma lainnya. Algoritma RSA akan menempati peringkat kedua karena waktu komputasinya sedikit lebih lama dibandingkan Rabin walaupun tidak selama algoritma ElGamal. Dan yang menduduki peringkat terakhir adalah Rabin karena proses enkripsinya yang sederhana sehingga waktu komputasi yang dibutuhkannya jauh lebih sedikit dibandingkan dengan dua algoritma yang lainnya.

e. Hasil enkripsi

Hasil dari enkripsi dari algoritma ElGamal menjadi dua kali ukuran plainteksnya karena pada proses enkripsi ElGamal, blok yang sama dilakukan enkripsi sebanyak dua kali dengan proses yang berbeda dan hasilnya dijadikan cipherteks.

Hasil enkripsi RSA dan teknik Rabin sama-sama baik dan memiliki tingkat keamanan yang tinggi walaupun tidak setinggi hasil enkripsi algoritma ElGamal karena proses yang rumit yang ada dalam proses enkripsinya. Hasil enkripsi dari kedua teknik tersebut tidak dapat dibandingkan mana yang lebih baik karena sama-sama merupakan hasil enkripsi.

f. Proses dekripsi

Dari aspek proses dekripsi, tidak jauh berbeda dengna proses enkripsi. Proses dekripsi pada ElGamal membutuhkan waktu yang lebih lama dibandingkan dengan RSA maupun Rabin karena kompleksitas proses dekripsinya yang rumit. Dibutuhkan dua kali komputasi karena ukuran chiperteks yang lebih besar dibandingkan plainteksnya.

Proses dekripsi pada RSA membutuhkan waktu sedikit lebih lama dari pada teknik Rabin. Namun teknik Rabin memiliki kekurangan pada proses dekripsinya, yaitu melakukan proses yang dapat dikatakan kurang efektif karena melakukan proses *Chinese remainder* sebanyak empat kali, sementara hasil dekripsi yang benar hanya satu dan penerima harus menentukan sendiri mana yang benar.

g. Hasil dekripsi

RSA dan ElGamal memiliki hasil dekripsi yang sangat akurat dengan plainteksnya. Kemungkinan kesalahan sangat kecil. Namun berbeda dengan teknik Rabin, seperti yang telah dijelaskan dalam makalah ini bahwa teknik Rabin menghasilkan empat kemungkinan plainteks dimana penerima pesan harus dapat menentukan sendiri plainteks sebenarnya diantara keempat kemungkinan yang dihasilkan proses dekripsi.

h. Keamanan teknik kriptografi

Tingkat keamanan ElGamal sangatlah tinggi mengingat tingkat kerumitan proses pembangkitan kunci, proses enkripsi, dan proses dekripsinya. Hingga saat ini belum ada kriptanalis yang berhasil memecahkan chiperteks dengan teknik ElGamal.

Tingkat keamana kedua teknik yang lain, yaitu Rabin dan RSA itu relatif sama karena keduanya mengandalkan kekuatan sulitnya memfaktorkan bilangan yang berukuran besar. Disamping itu, mengingat bahwa teknik Rabin merupakan varian dari teknik RSA maka tingkat keamanan keduanya relatif sama.

V. KESIMPULAN

Dilihat dari berbagai aspek yang dianalisis, maka dapat disimpulkan beberapa hal antara lain:

- Teknik kriptografi algoritma kunci-publik bergantung pada pembangkitan kunci. Pembangkitan kunci yang paling kompleks adalah algoritma ElGamal dan yang paling sederhana serta cepat adalah algoritma Rabin. Semakin kompleks kunci yang dibangkitkan, maka akan semakin kuat algoritma tersebut tetapi lebih memakan *resource* yang lebih banyak
- Hasil dekripsi cipherteks pada teknik Rabin menghasilkan empat kemungkinan plainteks dimana penerima harus menentukan sendiri plainteks yang benar. Penerima akan semakin sulit menentukan plainteks hasil dekripsi mana yang benar jika plainteks awalnya berupa angka.
- Teknik kriptografi Rabin merupakan varian dari RSA sehingga tingkat keamanan keduanya relative sama dengan mengandalkan kekuatan sulitnya memfaktorkan bilangan yang besar.
- Teknik kriptografi Rabin lebih cocok digunakan pada *resource* yang sederhana dengan *processor* yang tidak dapat melakukan komputasi yang kompleks dan rumit seperti *smartcard*. Sedangkan teknik kriptografi ElGamal merupakan kebalikan dari teknik kriptografi Rabin yang hanya dapat digunakan pada *resource* yang sudah lebih baik dengan *processor* yang dapat melakukan komputasi

yang rumit dan kompleks.

- Karena adanya disambiguitas secara tidak langsung (pada hasil dekripsi) hal ini menambah biaya komputasi dan hal tersebut menyebabkan Rabin dianggap kurang efisien dibandingkan RSA dan ElGamal sehingga algoritma Rabin jarang digunakan.
- Diantara ketiga algoritma tersebut, algoritma RSA merupakan algoritma yang tidak terlalu sederhana tetapi juga tidak terlalu rumit sehingga algoritma RSA merupakan algoritma yang paling pas jika hendak mengimplementasikan algoritma kriptografi kunci publik.

REFERENCES

[1] Munir, Rinaldi, *Kriptografi Kunci Publik*, Program Studi Teknik Informatika.
Waktu akses: 26 April 2011 pukul 18.00.

[2] Munir, Rinaldi, *Algoritma RSA*, Program Studi Teknik Informatika.
Waktu akses : 8 Mei 2011 pukul 19.00.

[3] Munir, Rinaldi, *Algoritma ElGamal*, Program Studi Teknik Informatika.
Waktu akses : 8 Mei 2011 pukul 20.38.

[4] http://en.wikipedia.org/wiki/Public-key_cryptography
Waktu akses: 26 April 2011 pukul 18.15.

[5] http://en.wikipedia.org/wiki/RSA
Waktu akses: 26 April 2011 pukul 18.31.

[6] http://en.wikipedia.org/wiki/ElGamal_encryption
Waktu akses: 26 April 2011 pukul 18.50.

[7] http://en.wikipedia.org/wiki/Rabin_cryptosystem
Waktu akses: 26 April 2011 pukul 19.10.

[8] Pinkas, Benny, *Rabin's Encryption Systems, Digital Signature*, 2005.
Waktu akses : 8 Mei 2011 pukul 23.38.

[9] Rădulescu, Mihnea, *Public-Key Cryptography : the RSA and the Rabin Cryptosystems*, 2008.
Waktu akses : 9 Mei 2011 pukul 00.20.

[10] http://sandi.math.web.id/download/paper/cryptoclub-serangan_terhadap_kriptografi.pdf
Waktu akses : 9 Mei 2011 pukul 08.37.

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 9 Mei 2011

ttd

Maureen Linda Caroline
13508049