KRIPTOGRAFI ALGORITMA RSA UNTUK PENGAMANAN DATABASE BERBASIS JAVA DEKSTOP PADA SMA MUHAMMADIYAH 15 JAKARTA BARAT

Irwan Adji Darmawan[1)], **Mufti**[2)]
[1]Program Studi Teknik Informatika, Fakultas Teknologi Informasi, Universitas Budi Luhur
[1,2]Jl. Raya Ciledug, Petukangan Utara, Kebayoran Lama, Jakarta Selatan 12260
E-mail : Irwan.wandji@gmail.com[1)], muftyhayat@gmail.com[2)]

Abstrak

Tingginya teknologi yang digunakan untuk bertukar informasi maka semakin rentan tingkat ancaman pencurian data dan informasi. SMA Muhammadiyah 15 Jakarta Barat merupakan instansi yang memiliki database penting seperti data siswa, data nilai, data guru, data absensi yang disimpan pada media penyimpanan elektronik. Enkripsi merupakan salah satu cara yang dapat dilakukan dengan aplikasi kriptografi yang mengimplementasikan algoritma RSA (Rivest Shamir Adleman) dengan bahasa pemrograman Java berbasis Dekstop ini dapat mengamankan serta menjaga kerahasiaan database SMA Muhammadiyah 15 Jakarta Barat dari pihak yang tidak memiliki hak untuk mengetahui dan merubah isi. Setelah proses enkripsi database dilakukan maka hasil proses enkripsi database tersebut tidak dapat dibaca. Pada saat database yang sudah terenkripsi dilakukan proses dekripsi maka database yang tadinya tidak dapat dibaca akan kembali seperti semula tanpa perubahan isi.Pengujian enkripsi database dengan nama tabel "data_siswa" yang memiliki jumlah field sebanyak 20 dan terdapat 38 record, menggunakan public key "adji.pub" lalu didapatkan waktu proses selama 3.466 second, dengan status berhasil. Kemudian dilakukan pengujian dekripsi database dengan nama tabel "data_siswa" yang memiliki jumlah field sebanyak 20 dan terdapat 38 record, menggunakan private key "adji.priv" lalu didapatkan waktu proses selama 5.633 second, dengan status berhasil.

Kata kunci: RSA, Kriptografi, *Database*, *Java*, *Dekstop*.

1. PENDAHULUAN

SMA Muhammadiyah 15 Jakarta Barat adalah salah satu Sekolah Menengah Atas di kotamadya Jakarta Barat yang sudah menerapkan penyimpanan data siswa dengan database. Banyak database yang bersifat rahasia dan tidak bisa dirubah oleh pihak-pihak yang tidak berhak merubahnya. Untuk menjaga keamanan database maka dapat menggunakan kriptografi. Oleh sebab itu, pengguna database membutuhkan bantuan untuk keamanan database yang disimpannya. Penerapan kriptografi pada SMA Muhammadiyah 15 Jakarta Barat akan diutamakan bagaimana kriptografi yang dapat mengamankan database yang tersimpan menjadi aman dan hanya dapat dipergunakan oleh pihak yang berhak membuka serta menggunakannya.

Dalam melakukan penelitian ini penulis menggunakan metode kriptografi dengan algoritma RSA (Rivest Shamir Asleman) sebagai pengamanan database. Algoritma RSA (Rivest Shamir Asleman) dipilih karena memiliki mekanisme kerja yang baik dan dianggap cepat dalam eksekusi enkripsi database, sehingga data yang telah di enkripsi memiliki tingkat keamanan tinggi dari pencurian data.

2. LANDASAN TEORI

2.1. Definisi Sistem Keamanan Data

Sistem merupakan seperangkat elemen yang membentuk kumpulan atau prosedur-prosedur atau bagan-bagan pengolahan yang mencari suatu tujuan bagian atau tujuan bersama dengan mengoprasikan data atau barang pada waktu rujukan tertentu untuk menghasilkan informasi atau barang. Keamanan adalah suasana aman, ketentraman, dan ketenangan (Salim, 2002). Data adalah keterangan-keterangan atau fakta-fakta yang dikumpulkan dari suatu populasi atau bagian populasi yang akan digunakan untuk menerangkan ciri-ciri populasi yang bersangkutan (Lungan, 2006). Sistem keamanan data adalah sistem yang dibuat untuk mencegah atau melindungi data yang ada dari kemungkinan pencurian data dari pihak yang tidak memiliki hak.

2.2. Definisi Kriptografi

Menurut Prayudi (2005), berdasarkan etimologi kata kriptografi (*Cryptography*) mmerupakan bahasa dari bahasa Yunani, pertama *kryptos* yang artinya yang tersembunyi dan kedua *graphein* yang artinya tulisan. Pada awalnya kriptografi diketahui sebagai ilmu untuk menyembunyikan pesan, namun dengan zaman yang berkembang dengan pesat saat ini arti kriptografi berkembang menjadi ilmu tentang teknis matematis yang digunakan untuk menyelesaikan persoalan keamanan berupa privasi dan otentikasi.

2.3. Jenis Kriptografi Berdasarkan Kunci

2.3.1 Algoritma Simetris

Algoritma ini disebut simetris karena memikiki *key* yang sama dalam proses enkripsi dan dekripsi maka algoritma ini juga disebut algoritma kunci tunggal atau algoritma satu kunci.

Kelebihan algoritma simetris :
1) Ukuran kunci relative lebih pendek.
2) Proses enkripsi dan dekripsi kriptografi simetris membutuhkan waktu yang terbilang singkat.
3) Otentikasi pengiriman pesan langsung dari *ciphertext* yang diterima, karena kunci hanya diketahui oleh pengirim dan penerima saja.

Kekurangan algoritma simetris:
1) Kunci harus diubah setiap melakukan pertukaran informasi. Apabila kunci tersebut lupa atau hilang, maka pesan tersebut terkunci secara permanen.
2) Kunci simetris harus dikirim melalui saluran komunikasi yang aman, dan kedua pihak yang berkomunikasi harus menjaga kerahasiaan kunci.

2.3.2 Algoritma Asimetris

Algoritma ini disebut asimetris karena kunci yang digunakan untuk enkripsi berbeda dengan kunci yang digunakan untuk dekripsi. Kunci yang digunakan untuk enkripsi adalah kunci publik atau *public key* sehingga algoritma ini juga disebut dengan algoritma kunci *public*. Sedangkan kunci untuk dekripsi menggunakan kunci rahasia atau *private key* (Prayudi, 2005)

Kelebihan algoritma asimetris:
1) Pasangan *private key* dan *public key* tidak perlu dirubah dalam jangka waktu yang lama.
2) Hanya *private key* yang perlu dijaga kerahasiaannya oleh setiap pihak yang bertukar informasi. Tidak ada kebutuhan mengirim *private key* sebagaimana kunci simetri.

Kelemahan algoritma asimetris:
1) Proses enkripsi dan dekripsi yang tebilang lebih lama dari algoritma simetris, karena menggunakan bilangan yang cukup besar dan operasi bilangan yang besar
2) Ukuran *ciphertext* lebih besar dari *plaintext*.
3) Ukuran kunci lebih besar daripada ukuran kunci simeris

2.4. Algoritma RSA

Algoritma RSA dikenalkan oleh tiga orang peneliti yaitu Ron Rivest, Adi Shamir, dan Len Adleman, pada tahun 1976. RSA (Rivest Shamir Adleman) adalah teknik kriptografi yang memanfaatkan 2 bilangan prima. Dari kedua bilangan prima yang di olah maka dapat diperoleh sebuah *Public Key* dan *Private Key*. Public key digunakan untuk menenkripsi sedangkan Private Key digunakan sebagai kunci untuk membuka atau mengembalikan kebentuk semula informasi yang sudah di enkripsi.

Skema algoritma kunci *public* sandi RSA terdiri dari tiga proses, yaitu proses pembentukan kunci, proses enkripsi dan proses dekripsi. Sebelumnya diberikan terlebih dahulu beberapa konsep perhitungan matematis yang digunakan RSA

Pada dasarnya pada RSA terdapat 3 proses, pertama yaitu proses pembentukan kunci atau bisa disebut dengan Generate Key, kedua proses enkripsi sebagai proses pengamanan data datau informasi dan ketiga proses dekripsi sebagai proses pengembalian data dan informasi yang sudah di enkripsi kebentuk semula (RSA and *Public Key Criptography*, 2003).

2.4.1 Algoritma Pembangkitan Pasangan Kunci

Untuk pembangkitan pasangan kunci RSA, maka digunakan algoritma dengan alur sebagai berikut:
1) Memilih dua bilangan prima sembarang yang kemudian diberi nama, pada kali ini penulis memberikan nama p dan q. Untuk p dan q diharapkan untuk menjaga kerahasiaannya.
2) Kemudian hitung besaran nilai n yang didapat dari p x q
3) Menghitung m = (p-1)(q-1).
4) Memilih nilai e (kunci publik) yang relatif prima terhadap m.
5) Relatif prima terhadap m artinya faktor pembagi keduanya yang didapat adalah 1, atau dapat disebut dengan cara gcd (e,m) = 1. Mencari bilangan yang relatif prima dapat menggunakan algoritma *Euclid*.
6) Menghitung d (kunci pribadi), untuk mencari nilai d secara matematis (e x d) mod m = 1. Dapat juga menggunakan algoritma *Extended Euclid*.

Maka hasil dari algoritma tersebut diperoleh:
1) Kunci publik adalah pasangan (e,n).
2) Kunci pribadi adalah pasangan (d,n).

2.4.2 Enkripsi Pesan

1) Menggunakan kunci publik (e,n).
2) *Plaintext* M dibagi-bagi menjadi blok-blok m_1, m_2, m_3,… dengan tujuan untuk mempermudah
3) Setiap blok m_i di enkripsikan menjadi c_i, dengan rumus $c_i = m_i^e \bmod n$.

2.4.3 Dekripsi Pesan

1) Menggunakan kunci pribadi (d,n).
2) Pilih *ciphertext* C.

3) Setiap blok c_i di dekripsikan dalam blok m_i, dengan menggunakan rumus $m_i = c_i^d \bmod n$.

2.4.4 Penggunaan Algoritma RSA

Berikut ini merupakan contoh penggunaan algoritma RSA :

Misalkan p=13 dan q=31 (harus bilangan prima), kemudian hitung nilai n = p x q = 13 x 31 = 403, m = (p-1)(q-1) = (13-1)(31-1) = 360

Pilih kunci e = 7, karena 7 relatif prima dengan 360, e dan n dapat di publikasikan ke umum.

Selanjutnya menghitung kunci dekripsi d menggunakan rumus (e x d) mod m = 1, sehingga menjadi (7 x d) mod 360 = 1. Setelah melakukan percobaan nilai-nilai d = 1,2,3,4… maka diperoleh nilai kunci pribadi yaitu 103. Ini merupakan kunci dekripsi yang harus dirahasiakan.

Pesan yang akan dikirim adalah M = IRWAN atau dalam *decimal* (kode ASCII) adalah : 7382876578, nilai tersebut dipecah menjadi blok-blok m. maka blok-blok yang akan terbentuk menjadi:

$m_1 = 73$; $m_2 = 82$; $m_3 = 87$; $m_4 = 65$; $m_5 = 78$;

sebelumnya telah diketahui bahwa kunci publik adalah e = 7 dan n = 403. Maka pesan M dapat dienkripsikan menjadi :

$c_1 = 73^7 \bmod 403 = 44$
$c_2 = 82^7 \bmod 403 = 173$
$c_3 = 87^7 \bmod 403 = 87$
$c_4 = 65^7 \bmod 403 = 234$
$c_5 = 78^7 \bmod 403 = 39$

sehingga *ciphertext* yang dihasilkan adalah : 44 173 87 234 39.

Selanjutnya pesan yang terenkripsi tersebut dikirim kepada penerima pesan dan si penerima pesan sudah memiliki kunci pribadi d = 103 dan n = 403 sehingga:

$m_1 = 44^{103} \bmod 403 = 73$
$m_2 = 173^{103} \bmod 403 = 82$
$m_3 = 87^{103} \bmod 403 = 87$
$m_4 = 234^{103} \bmod 403 = 65$
$m_5 = 39^{103} \bmod 403 = 78$

maka dapat dihasilkan kembali M = 7382876578, lalu dalam pengkodean ASCII dapat dibaca sebagai berikut : M = IRWAN

3. METODOLOGI PENELITIAN

Untuk penelitian ini penulis menggunakan beberapa metode untuk mendapatkan informasi yang dibutuhkan dalam penelitian ini serta metode penelitian juga dapat digunakan untuk mengatasi masalah yang ditemukan pada tempat penelitian. Adapun beberapa metode sebagai berikut dibawah ini :

3.1. Studi Literatur

Dalam Metode Studi Literatur penulis menggunakannya untuk memahami beberapa referensi jurnal, makalah ataupun buku yang bertujuan untuk mendapatkan informasi sebagai pendukung penelitian

3.2. Analisis Data

Menganalisis Algoritma kriptografi yang digunakan yaitu algoritma RSA (Rivest Shamir Asleman), serta teknik-teknik yang digunakan.

3.3. Perancangan Sistem

Merancang sistem aplikasi untuk mengimplementasikan kriptografi algoritma RSA (Rivest Shamir Asleman) menggunakan bahasa pemrograman *java* dengan berbasis *desktop*.

3.4. Pengujian Sistem

Metode ini digunakan untuk memperoleh hasil yang didapat dari serangkaian pengujian dan jalan program yang nantinya akan dituangkan pada bagian tabel pengujian dan kesimpulan.

4. HASIL DAN PEMBAHASAN

4.1 Tampilan Layar

Pada bagian ini akan dijelaskan mengenai tampilan layar pada aplikasi mulai dari tampilan awal dijalankan sampai dengan selesai dijalankan. Berikut ini juga akan diberikan gambar beserta penjelasan mengenai tampilan-tampilan yang ada ada aplikasi pengamanan *database* yang telah dibuat

4.1.1 Tampilan Layar Form Login

Layar ini muncul pada pertama kali aplikasi dijalankan yang mengharuskan *user* memasukan *username* dan *password* yang telah ditentukan.

Gambar 4.1 Tampilan Form Login

4.1.2 Tampilan Layar Form Menu Utama

Tampilan layar *Form* Menu Utama pada *form* ini berisi beberapa menu seperti *Generate key*, *Encryption*, *Decryption*, *Help*, Profile, dan tak lupa tombol Exit untuk keluar dari aplikasi.

Gambar 4.2 Tampilan Layar Form Menu Utama

4.1.3 Tampilan Layar Form Generate Key

Tampilan layar *Form Generate Key* berfungsi membangkitkan sepasang *Public key* dan *Private key* seperti pada gambar 4.5 berikut ini :

Gambar 4.3 Tampilan Layar Form Generate Key

4.1.4 Tampilan Layar Form Encryption Database

Pada rancangan Layar *Form Encryption Database*, berfungsi untuk melakukan enkripsi *database* seperti pada gambar 4.4 berikut :

Gambar 4.4 Tampilan Layar Form Encryption Database

4.1.5 Tampilan Layar Form Decryption Database

Pada rancangan Layar *Form Decryption Database*, berfungsi untuk melakukan dekripsi

Gambar 4.5 Tampilan Layar Form Decryption Database

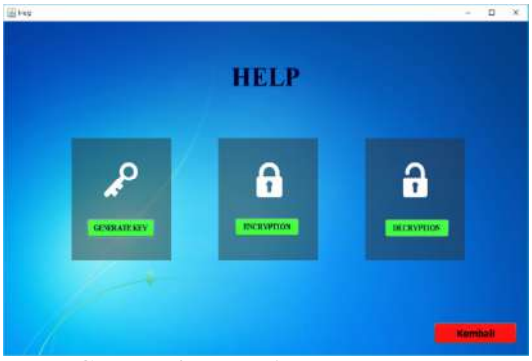
4.1.6 Tampilan Layar Form Profile

Tampilan layar *form* profile berfungsi agar *user* dapat melihat dan mengetahui informasi tentang penulis

Gambar 4.6 Tampilan Layar Form Profile

4.1.7 Tampilan Layar Form Help

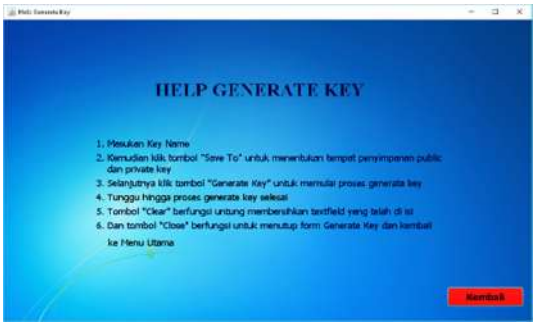
Pada rancangan menu *Form Help* ini, berfungsi agar *user* dapat mengetahui informasi yang bisa digunakan untuk menggunakan aplikasi ini.



Gambar 4.7 Tampilan Layar Form Help

4.1.8 Tampilan Layar Form Help Generate Key

Pada rancangan menu *Form Help Generate key* ini, berfungsi agar *user* dapat mengetahui informasi bantuan



Gambar 4.8 Tampilan Layar Form Help Generate Key

4.1.9 Tampilan Layar Form Help Encryption Database

Pada rancangan menu *Form Help Encryption Database* ini, berfungsi agar *user* dapat mengetahui informasi bantuan atau tata cara *user* menggunakan menu *Encryption Database*.

Gambar 4.9 Tampilan Layar Form Help Encryption Database

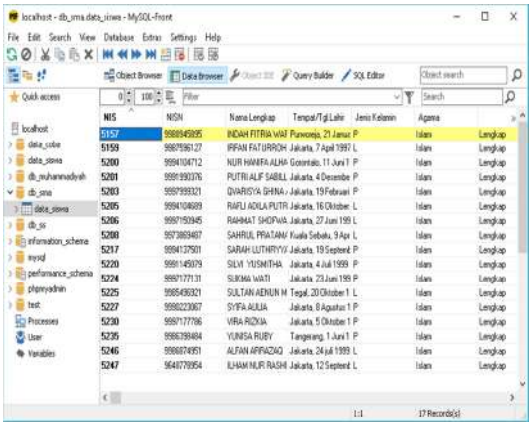
4.1.10 Tampilan Layar Form Help Decryption Database

Pada rancangan menu *Form Help Decryption Database* ini, berfungsi agar *user* dapat mengetahui informasi bantuan atau tata cara *user* menggunakan menu *Decryption Database*.

Gambar 4.10 Tampilan Layar Form Help Decryption Database

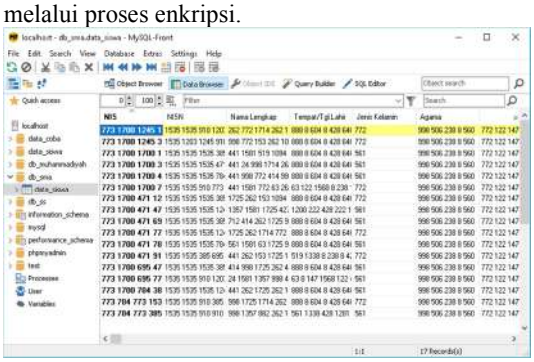
4.2 Pengujian Program

4.2.1 Proses Pengamanan Database (Enkripsi)



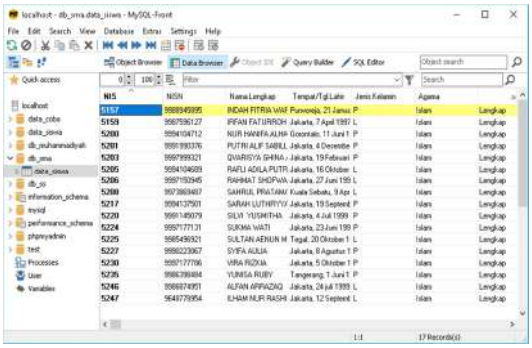
Gambar 4.11 Tampilan Tabel Database Asli

. Berikut ini pada gambar 4.23 merupakan tampilan *database* dengan isi acak yang berhasil melalui proses enkripsi.



Gambar 4.12 Tampilan Tabel Database Hasil Enkripsi

4.2.2 Proses Pengembalian Isi Database (Dekripsi)



Gambar 4.24 Tampilan Tabel Database Hasil Dekripsi

5. KESIMPULAN

Berdasarkan hasil analisa serta serangkaian uji coba yang telah dilakukan terhadap permasalahan dari aplikasi yang telah dibuat, maka dapat di ambil kesimpulan maupun saran yang diperlukan untuk melakukan pengembangan aplikasi yang lebih baik lagi kedepannya.

1) Aplikasi ini dapat merubah *database* asli (*plaintext*) menjadi karakter yang tidak dapat dipahami (*ciphertext*).
2) Aplikasi ini dapat mengembalikan isi *database* yang telah dienkripsi menjadi seperti semula secara utuh tanpa adanya perubahan isi *database*.
3) Satu *public key* dan *private key* dapat digunakan lebih dari satu kali untuk proses enkripsi dan dekripsi
4) Waktu yang diperlukan untuk proses enkripsi dan dekripsi berbanding lurus dengan ukuran *database* yang diproses (semakin kecil ukuran *database* yang akan diproses, maka semakin cepat proses enkripsi dan dekripsi lalu sebaliknya semakin besar ukuran *database* yang akan diproses, maka semakin lama proses enkripsi dan dekripsi).
5) Dengan menggunakan metode algoritma kriptografi RSA (*Rivest Shamir Adleman*) mampu meningkatkan keamanan data instansi yang bersifat rahasia sehingga dapat menurunkan tingkat pencurian data.

6. DAFTAR PUSTAKA

[1] Robert G Murdick, dkk. 1991. Sistem Informasi Untuk Manajemen Modern. Jakarta: Erlangga.

[2] Salim, Peter dan Yenny Salim. 2002, Kamus Bahasa Indonesia Kontemporer. Jakarta: Modern English Press.

[3] Lungan, R. Aplikasi Statistika dan Hitung Peluang, Yogyakarta : Graha Ilmu. (2006).

[4] Prayudi 2005, Studi dan Analisis Algoritma Rivest Code 6 (RC6) Dalam Enkripsi/Dekripsi Data. Seminar Nasional Aplikasi Teknologi Informasi 2005, Yogyakarta.

[5] Sadikin, R., 2012. Kriptografi untuk keamanan jaringan.

[6] Diffie, W.,&Hellman, M. New Direction In Cryptography. IEEE transactions on information Theory, 1976.

[7] Diny Ariyus, Pengantar Ilmu kriptografi teori analisis dan implementasi. Yogyakarta(2008) CV Andi Offset.

[8] Wico Chandra.2010, Kriptografi Dan Algoritma RSA, Bandung : Institut Teknologi Bandung. Makalah II2092 Probabilitas dan Statistik.

[9] RSA and *Public Key Criptography*, 2003. International workshop on practice and theory in public key cryptography proceding.

[10] Yuhefizard, S.kom. 2008, Database Management Menggunakan Microsoft Access 2003, Jakarta: PT Elex Media Komputindo.

[11] Abdul Kadir. 2002, Konsep & Tuntutan Praktis Basis Data, Yogyakarta: Penerbit Andi.