IMPLEMENTASI TEKNIK KRIPTOGRAFI RSA UNTUK PENGAMANAN DATA PENGIRIMAN SMS

Hardiana Riski Riswanto[1)], Khamaida Safinah[2)], Ainafatul Nur Muslikah[3)], Khadijah Fahmi Hayati Holle[4)]

[1] Fakultas Sains dan Teknologi, UIN Maulana Malik Ibrahim Malang
email: dianarisky44@gmail.com
[2] Fakultas Sains dan Teknologi, UIN Maulana Malik Ibrahim Malang
email: khamaidasafinah99@gmail.com
[3] Fakultas Sains dan Teknologi, UIN Maulana Malik Ibrahim Malang
email: ainafatul20@gmail.com
[4] Fakultas Sains dan Teknologi, UIN Maulana Malik Ibrahim Malang
email: khadijah.holle@uin-malang.ac.id

Abstract

Message sending is one activity that is often used by everyone. However, security in this message delivery system needs to be wary of spying or message piracy during the process of sending messages. Surely someone who sent the message does not know if someone's personal message has been stolen. With this initiative builds a security message using cryptographic RSA algorithm where the message sender or recipient of the message can send the message safely without being known to the message hijacker or spy. Cryptography that uses the RSA algorithm to secure messages. This RSA algorithm message will be decrypted with the public key and to encrypt the message. This application was built on the Android platform because the dominant person has an Android smartphone with a system that runs the length of the message character does not affect the speed at the time of sending the message to the recipient, and there is no limit on the length of the message character during the encryption process so that any length of the massage character can be encrypted well.

Keywords: security,android,cryptograph, RSA, SMS

1. PENDAHULUAN

Banyaknya fitur dan aplikasi yang ada pada android SMS (Short Message Service) sering digunakan untuk berkomunikasi. SMS (Short Message Service) sendiri merupakan sistem komunikasi tanpa kabel (nirkabel). SMS pertama kali muncul di Eropa bersamaan dengan sebuah teknologi komunikasi wireless yang memiliki banyak pengguna, yaitu Global Sistem for Mobile Comunication (GSM). SMS memiliki mekanisme kerja sistem dengan melakukan pengiriman short message dari satu terminal ke terminal yang lain. sistem pengiriman datanya dalam paket bandwidh yang kecil, dengan hal ini pengiriman suatu data yang singkat atau pendek dapat dilakukan dengan efisiensi yang tinggi[1].

Banyak terjadi kasus penyadapan pesan, contoh kasus yang terkenal pada tahun 1867 terdapat laporan perkara oleh Sebuah saham wall streetyang bekerjasama dengan Western Union agar melakukan penyadapan ke operator telegrap yang dikirim ke orang dan kemudian pesan telegrap itu diganti dengan pesan atau teks yang palsu[2].

Penggunaan SMS dipilih karena praktis dan selain itu biaya pengirimnya juga murah (di Indonesia, pengiriman SMS memakan biaya Rp. 250,- sampai Rp. 350,- dan range harga tersebut tergantung dari setiap operatornya). Dari faktor tersebut SMS banyak digunakan dalam hal pengiriman pesan. Akan tetapi pengiriman pesan melalui SMS keamanannya tidak terjaga kerahasiaannya. Untuk itu diperlukan suatu sistem yang dapat menjaga kerahasiaan pesan pada SMS. Sehingga dapat dibuat suatu software yang memiliki fungsi untuk enkripsi dan juga sebaliknya yaitu dekripsi pada pesan yang sudah dibuat. Proses enkripsi adalah suatu proses untuk mengamankan informasi, informasi tersebut tidak dapat dibaca tanpa pengetahuan khusus. Proses dekripsi adalah kebalikan dari proses enkripsi yaitu upaya pengolahan data menjadi suatu yang akan diutarakan secara jelas dan tepat agar pesan yang di kirim dapat dimengerti oleh orang yang dituju. Untuk melakukan proses enkripsi dan dekripsi pada pesan ini menggunakan

teknik kriptografi yang menggunakan algoritma RSA[3].

Kriptografi merupakan ilmu dan juga seni yang berguna untuk menjaga suatu keamanan pesan dengan menggunakan teknik atau algoritma matematika. Dalam menjaga keamanan data menggunakan kriptografi, data sederhana yang dikirim diubah kedalam bentuk sandi, laluu data sandi hanya bisa dibaca atau dikembalikan ke data yang sebenarnya hanya dengan menggunakan kunci(key) tertentu yang dimiliki oleh pihak tertentu [4].

Dari beberapa algoritma kriptografi yang sangat populer dan sering digunakan, algoritma RSA ini memiliki keamanan yang terletak pada pemfaktoran bilangan prima yang cukup besar dan sulit. Pemfaktoran bilangan prima yang cukup besar inilah yang membuat algoritma RSA belum dapat dipecahkan dengan algoritma yang lainnya sehingga sangat dijamin keamanannya. Oleh karena itu, dalam penelitian yang judul "Implementasi Teknik Kriptografi RSA untuk Pengamanan Data Pengiriman SMS" menggunakan algoritma kriptografi RSA untuk memberi keamanan pada pengiriman pesan di SMS.

2. METODE PENELITIAN

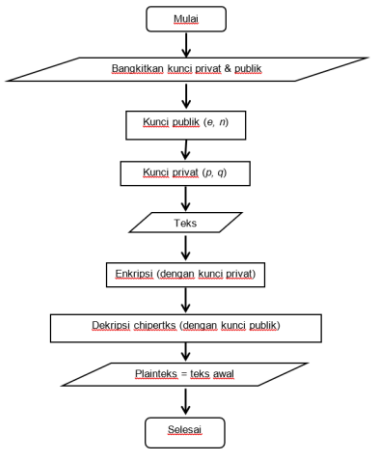
Metode untuk implementasi sistem pada penelitian ini adalah algoritma kriptografi RSA. Algoritma kriptografi RSA merupakan algoritma kriptografi asimetris. Algoritma asimetris ini sering disebut dengan algoritma kunci publik, yang artinya kata yang digunakan untuk melakukan enkripsi dan dekripsi berbeda[5]. RSA merupakan kependekan dari nama ketiga penemu algoritma ini yaitu Ron Rivest, Adi Sahmir, Leonard Adleman . Algoritma RSA ditemukan pada tahun 1977 oleh ketiga penemu tersebut. RSA adalah algoritma yang menggunakan perhitungan matematika yang rumit, yang memiliki 2 kunci yaitu kunci publik dan kunci privat. Kunci publik disini merupakan kunci yang boleh diketahui siapa saja yang digunakan untuk proses enkripsi. Sedangkan kunci privat adalah kunci yang hanya boleh diketahui oleh pihak-pihak tertentu saja yang digunakan untuk proses dekripsi. Hal tersebut membuat algoritma ini sulit ditembus oleh hacker. Hal lain yang menjadi terjaminnya keamanan algoritma ini adalah sulitnya pemfaktoran bilangan besar pada algoritma RSA[6].

Adapun langkah-langkah untuk proses mendapatkan kunci publik dan kunci privat adalah sebagai berikut:

a. Pilih 2 buah bilangan prima, *p* & *q*.
b. Hitung $r = p \times q$.....(1).
c. Sebaiknya $p \neq q$, sebab jika $p = q$ maka $r = p^2$ sehingga p dapat diperoleh dengan menarik akar pangkat dari *r*.
d. Hitung $\emptyset r = (p-1)(q-1)$ ..... (2).
e. Bangkitkan kunci rahasia dengan menggunakan $SK \cdot PK = 1 \pmod{(\emptyset r)}$.
f. $SK \cdot PK = 1 \pmod{(\emptyset r)}$ ekuivalen dengan $SK \cdot PK = 1 + \text{mod } (\emptyset r)$ sehingga dapat dihitung dengan $SK = \frac{1+m\emptyset(r)}{PK}$.....(3). [7]

Pada proses enkripsi, plainteks diubah ke dalam bentuk bilangan dengan menggunakan kode ASCII dalm bilangan decimal, sehingga plaintks m dinyatakan menjadi blok-blok x_1, x_2, x_3..... dalam [o, n-1]. Kemudian dienkripsi menggunakan rumus: $y_1 = x_i^{PK}$ mod r. Dalam proses dekripsinya, cipherteks didekripsi kembali blok c_i menggunakan rumus $x_1 = y_i^{SK}$ mod r, lalu dibuah kembali ke bentuk huruf dengan kode ASCII hasil dekripsi[8].

Dibawah ini merupakan flowchart algoritma RSA:

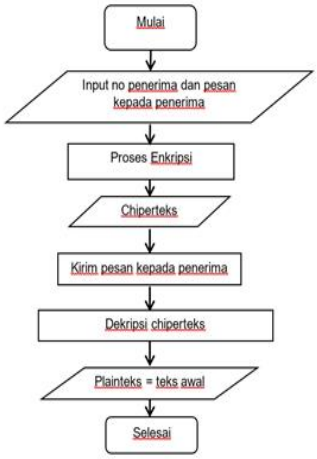


Gambar 1. Flowchart Algoritma RSA
Sumber : [2]

Dari gambar diatas dapat disimpulkan bahwa pada algoritma RSA awalnya teks

diamankan dengan proses enkripsi yang akan berubah menjadi susunan huruf/angka acak sehingga tidak dapat diakses oleh siapapun kemudian agar teks tersebut dapat dibaca oleh orang yang dituju, maka dilakukan proses dekripsi RSA kembali.

Alur dari perancangan sistem ini adalah, pertama pengirim harus mengisikan nomer penerima agar bisa dikirim ke sms penerima, setelah itu pengirim mengisikan pesan yang ingin disampaikan kepada pengirim kemudian dilakukan proses enkripsi pesan sehingga dididapatkan chiperteks dari pesan tersebut selanjutnya dilakukan proses pengiriman kepada penerima.Dan pengirim dapat mengetahui seberapa cepat pesan dapat dikim ke penerima. Setelah penerima mendapatkan chiperteks pesan, selanjutnya pesan di dekripsi sehingga pesan yang dikirm oleh pengirim dapat terbaca. Representasi dari alur perancangan sistem ini terdapat pada gambar flowchart sistem dibawah ini:



Gambar 2. Flowchart Sistem

3. HASIL DAN PEMBAHASAN

Dalam memahami algoritma RSA dilakukan perhitungan manual terlebih dahulu agar mengetahui dengan jelas implementasi perhitungan algoritma RSA secara manual, yang kemudian diimplementasikan pada sistem yang dibuat. Adapun implementasi perhitungan secara manual dan implementasi pada sistem algoritma RSA adalah sebagai berikut:

a. Perhitungan Manual Algoritma RSA

1) Menentukan nilai *p* dan *q*
   Pada penentan nilai *p* dan *q* ini, nilai *p* dan *q* harus bilangan prima, contohnya *p* = 13 sedangkan nilai *q* = 31.
2) Hitung kunci publik dengan cara menghitung modulus *r* seperti pada persamaan 1 dan 2 yaitu:
   $n = p \times q$
   $= 13 \times 31$
   $= 403$
   kemudian,
   $\emptyset(r) = (p-1)(q-1)$.
   $= (13-1)(31-1)$
   $= 360$
   Setelah mendapatkan kunci publik, kemudian menentukan nilai *PK* yang merupakan faktorial dari $\emptyset(r)$, faktorial dari $\emptyset(r)$ adalah 7.
3) Nilai *PK* sudah ditentukan, setelah itu menentukan nilai *SK* dengan rumus seperti pada persamaan 3 yaitu *SK* = $\frac{1+m\emptyset(r)}{PK} = \frac{1+m360}{7} = 103$
4) Setelah semua parameter telah berhasil dihitung dilakukan proses enkripsi, seperti dengan plainteks = *"BELAJAR"* dengan kode ASCII =

Tabel 1. Tabel ASCII kata belajar

B	E	L	A	J	A	R
66	69	76	65	74	65	82

Sehingga didapat enkripsinya dengan cara kode ASCII dipangkatkan dengan *PK* kemudian dikalikan dengan modulo *n*, seperti dibawah ini:

B = 66^7 mod 403 = 326
E = 69^7 mod 403 = 121
L = 76^7 mod 403 = 236
A = 65^7 mod 403 = 234
J = 74^7 mod 403 = 334
A = 65^7 mod 403 = 234
R = 82^7 mod 403 = 173

5) Dalam proses dekripsi juga memiliki cara yang sama dengan enkripsi, namun ada sedikit pebedaan dari rumusnya, yaitu $Chiperteks^{KS} \ mod \ n$.

B = 326^{103} mod 403 = 66
E = 121^{103} mod 403 = 69
L = 236^{103} mod 403 = 76
A = 234^{103} mod 403 = 65
J = 334^{103} mod 403 = 74
A = 234^{103} mod 403 = 65
R = 173^{103} mod 403 = 82

Dari hasil proses dekripsi membuktikan bahwa nilai-nilai pada karakter kembali ke kode ASCII nya masing-masing tiap huruf.

b. Implementasi Algoritma RSA

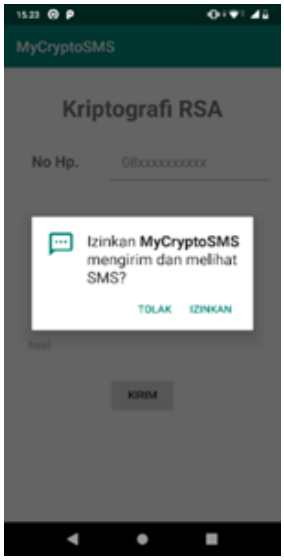
Implementasi algoritma kriptogrfi RSA disini menggunakan pemrograman berbasis android dengan *software* android studio, dan dengan bahasa promrograman java, berikut ini adalah tampilan pada sistem:

1) Tampilan Awal

Gambar 3. Tampilan Awal Sistem

Pada halaman ini terdapat 2 pilihan yaitu kirim pesan sms yang diperuntukkan untuk *sender*, dan baca pesan sms yang diperuntkkan untuk *receiver*.

2) Tampilan Tulis SMS

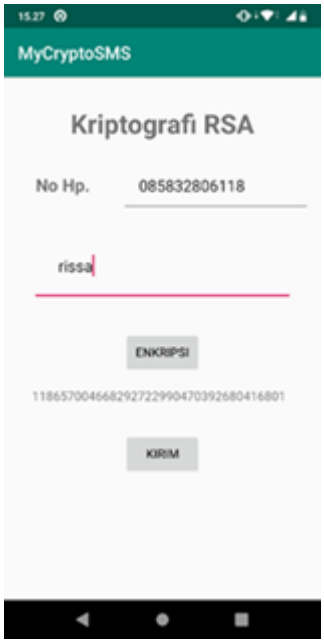


Gambar 4. Input Nomer *Handphone Receiver*

Notifikasi *permission* pada gambar 4 muncul untuk meminta izin fungsi internal yang terdapat di android. Fungsi internal yang digunakan adalah fungsi SMS.

Kemudian akan muncul tampilan seperti pada gambar 5.

- No Hp adalah nomer HP yang dituju untuk mengirim pesan.
- Terdapat pesan yang akan di enkripsi
- Button Enkripsi digunakan untuk mengenkripsi pesan yang ditulis
- Button Kirim digunakan untuk mengirim pesan yang telah di enkripsi dengan menggunakan fungsi SMS yang ada di HP

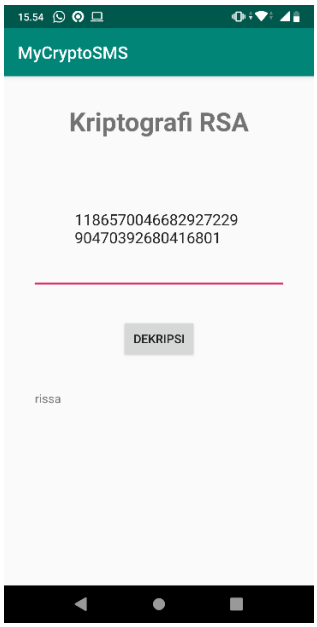


Gambar 5. Tampilan Input Pesan

Pada gambar 5 dapat dijelaskan bahwa ketika pengirim menuliskan pesan dan akan mengirimnya kepada penerima pengirim harus melakukan proses enkripsi pada pesan terlebih dahulu dengan cara mengmencet tombol enkripsi. Agar pesan yang dikirim tidak dapat dibaca oleh orang lain apalagi oleh orang yang tidak bertanggung jawab. Setelah dilakukan proses enkripsi dengan memencet tombol enkripsi barulah pengirim dapat mengirim pesan rahasia tersebut kepada penerima yang ingin dituju. Setelah penerima memperoleh chiperteks pesan dari pengirim, untuk melihat pesan apa yang dikirim oleh

pengirim, penerima harus memiliki aplikasi ini, agar dapat mendekripsi chiperteks menjadi plainteks awal sehingga dapat melihat isi pada pesan dibalik chiperteks tersebut.

3) Tampilan Baca Pesan



Gambar 6. Tampilan Dekripsi Pesan.

c. Pengujian Sistem

Pengujian pada sistem ini bertujuan untuk mengetahui kecepatan sistem dalam mengirim pesan hasil enkripsi melalui SMS dan panjang karakter pesan maksimal yang dapat dikirim oleh sistem. Pengujian dilakukan dengan cara menginputkan pesan yang akan dikirim setelah itu dilakukan proses enkripsi dengan menekan tombol enkripsi pada aplikasi dan menekan tombol kirim untuk mengirim pesan tersebut kepada penerima.

Tabel 2. Proses Pengujian

Percobaan ke-	Data Pengujian			Enkripsi	Status
	Jumlah Karakter sebelum dienkripsi	Jumlah Karakter setelah dienkripsi	Waktu Pengiriman		
1.	10	72	0.02 detik	Berhasil	SMS terkirim
2.	15	108	0.064 detik	Berhasil	SMS terkirim
3.	20	151	0.042 detik	Berhasil	SMS terkirim
4.	30	189	0.045 detik	Berhasil	SMS tidak terkirim
5.	40	288	0.019 detik	Berhasil	SMS tidak terkirim
6.	50	308	0.108 detik	Berhasil	SMS tidak terkirim

Dari pengujian tersebut dapat disimpulkan bahwa sistem berjalan dengan baik, panjang karakter pesan tidak mempengaruhi kecepatan pada waktu pengiriman pesan pada penerima, dan tidak ada batasan panjang karakter pesan pada saat proses enkripsi sehingga berapapun panjang karakter pesan dapat di enkripsi dengan baik . Namun, jumlah karakter pada pesan sebelum dienkripsi terbatas sampai 20-25 karakter saja, hal itu disebabkan karena adanya perubahan panjang karakter pesan setelah dienkripsi sehingga melebihi batas panjang karakter yang ada pada SMS yaitu ≤ 160 karakter.

4. KESIMPULAN

Berdasarkan hasil penelitian yang telah dilakukan maka dapat disimpulkan bahwa pada SMS dibutuhkan keamanan untuk mengamankan data atau pesan rahasia agar tidak disalah gunakan oleh orang yang tidak bertanggungg jawab. Dengan adanya implementasi algoritma kriptografi RSA ini hal itu dapat teratasi. Terbukti dengan keberhasilan sistem dalam enkripsi pada pesan SMS yang dikirm, jumlah karakter pada pesan juga tidak mempengaruhi kecepatan waktu pengiriman pesan. Begitu juga dengan dekripsi pesan chiperteks, pengembalian pesan ke plainteks awal berhasil dilakukan dengan baik. Namun, terbatasnya jumlah karakter pada pengiriman pesan SMS yaitu sejumlah ≤ 160, membuat pesan sebelum dienkripsi menjadi terbatas karena pesan yang sudah dienkripsi telah melalui proses perhitungan algoritma kriptografi RSA sehingga jumlah panjang karekter pesan berubah. Oleh karena itu, penelitian ini dapat dikembangkan dengan adanya pengiriman SMS terpisah apabila panjang karakter pesan setelah dienkripsi lebih dari 160 karakter.

5. REFERENSI

[1] S. T. Riyanto, "Aplikasi Enkripsi Dan Dekripsi SMS Menggunakan Kriptografi Kunci Publik Dengan Algoritma RSA," *eprint UPN "Veteran" Yogyakarta.*, 2012.

[2] A. R.; D. Alvianto, "Pengaman Pengiriman Pesan Via SMS dengan Algoritma RSA Berbasis Android," *J. SAINS dan SENI ITS*, vol. 4, no. 1, pp. 1–5, 2015.

[3] S. Deris, *Sistem Keamanan Komputer*.

Jakarta: PT. Alex Media Komputindo, 2005.

[4] M. Harun, *Kriptografi untuk Keamanan Data*. Yogyakarta: Deepublish, 2018.

[5] D. Ariyus, "Pengantar Ilmu Kriptografi Teori, Analisis dan Implementasi.," *Journal of Chemical Information and Modeling*. pp. 1–438, 2008, doi: 10.1017/CBO9781107415324.004.

[6] R. Sahara, H. Prastiawan, and A. Rohman, "Implementasi Keamanan SMS Dengan Algoritma RSA Pada Smartphone Android," *J. Ilm. FIFO*, vol. 9, no. 2, p. 118, 2017, doi: 10.22441/fifo.2017.v9i2.004.

[7] I. J. Dewanto, V. Yanto, J. A. Utara, T. Tomang-Kebon, and J. Jakarta, "Pembuatan Aplikasi SMS Kriptografi RSA Dengan Android," 2013.

[8] A. Ginting, R. R. Isnanto, and I. P. Windasari, "Implementasi Algoritma Kriptografi RSA untuk Enkripsi dan Dekripsi Email," *J. Teknol. dan Sist. Komput.*, vol. 3, no. 2, p. 253, 2015, doi: 10.14710/jtsiskom.3.2.2015.253-258.