UJICOBA SISTEM KEAMANAN INFORMASI DENGAN ALGORITMA KRIPTOGRAFI RSA DAN RSA-CRT PADA SISTEM E-MEMO BERBASIS MOBILE

Ari Muzakir
Fakultas Ilmu Komputer, Program Studi Teknik Informatika
Universitas Bina Darma
Email: arimuzakir@binadarma.ac.id

Meigi Rahman
Fakultas Ilmu Komputer, Program Studi Teknik Informatika
Universitas Bina Darma
Email: megi.mailbox@gmail.com

ABSTRAK

Kriptografi adalah suatu metode keamanan untuk melindungi suatu informasi dengan menggunakan kata-kata sandi yang hanya bisa dimengerti oleh orang yang berhak mengakses informasi tersebut. Untuk membuat pesan memo agar tidak terbaca oleh orang yang tidak memiliki hak, maka solusi yang dapat dibuat yaitu menggunakan perangkat lunak khusus yang dapat diakses menggunakan perangkat mobile dan dalam perangkat lunak tersebut di implementasikan sebuah algoritma kriptografi yaitu algoritma Algoritma *Rivest-Shamir-Adleman* (RSA). Dalam penelitian ini menggabungkan penggunaan algoritma RSA yang dimodifikasi dengan *Chinese Remainder Theorem* (CRT) untuk mempercepat proses dalam algoritma tersebut. Ujicoba dilakukan pada aplikasi e-memo yang ada di pemerintahan sebagai bagian dari informasi yang sering di pertukarkan di lembaga pemerintahan tersebut. Salah satunya adalah memo yang digunakan untuk menyampaikan pesan baik yang bersifat biasa mapun yang bersifat rahasia. Hasil dari implementasi yaitu berupa aplikasi keamanan yang menggunakan kriptografi RSA sebagai sistem keamanan berbasis mobile. Pengujian yang dilakukan yaitu proses dekripsi menggunakan algoritma RSA-CRT untuk masing-masing panjang pesan sebanyak 30, 60, dan 90 karakter untuk masing-masing pasangan nilai p dan q memiliki kecepatan rata-rata 2731 kali lebih cepat dibandingkan menggunakan algoritma RSA pada umumnya tanpa dilakukan modifikasi.

Kata kunci: teknik kriptografi, algoritma *RSA-CRT,* PHP*, jquerymobile.*

ABSTRACT

Cryptography is a security method for protecting information by using passphrases that only those who are entitled to access the information can understand. To create a memo message for unauthorized persons, a workable solution is to use special software that can be accessed using a mobile device and in that software implemented a cryptographic algorithm, the Rivest-Shamir-Adleman (RSA) Algorithm. In this study, combine the use of modified RSA algorithm with Chinese Remainder Theorem (CRT) to speed up the process in the algorithm. Tests are conducted on e-memo applications that exist in the government as part of information that is often exchanged in government agencies. One of them is a memo that is used to convey messages both of normal and secret nature. Results from the implementation in the form of a security application that uses RSA cryptography as a security system based on mobile. Tests were done of the decryption process using the algorithm RSA-CRT for each length of the message as much as 30, 60, and 90 characters for each pair of values of p and q have average speeds 2731 times faster than using the RSA algorithm in general without made modifications.

Keywords: *cryptography technique, RSA algorithm, RSA-CRT algorithm, jquery mobile.*

1. PENDAHULUAN

Dalam melaksanakan tugasnya Dinas Pendapatan, Pengelolaan Keuangan dan Aset Daerah (DPPKAD) memiliki beberapa cara dalam berkomunikasi antar pimpinan baik kepala dinas, kepala bagian dan kepala seksi yaitu menggunakan pesan singkat, telepon, surat dan memo. Salah satu bentuk komunikasi yang dipakai adalah menggunakan memo. Memo digunakan pada DPPKAD Kabupaten Musi Banyuasin untuk menyampaikan pesan baik yang bersifat biasa mapun yang bersifat rahasia. Kondisi

yang terjadi saat ini pada DPPKAD Kabupaten Musi Banyuasin dalam pemberian atau pengiriman pesan memo tidak teroganisir dengan baik, hal tersebut dapat dilihat ketika memo diberikan hanya diletakkan di atas meja penerima sehingga orang yang tidak memiliki hak untuk mengetahui isi memo dapat melihat atau membaca isi pesan tersebut dan pada akhirnya jika isi pesan tersebut bersifat penting dan rahasia maka hal tersebut sulit untuk dijaga. Aspek keamanan menjadi sangat penting untuk menjaga data atau informasi agar tidak disalahgunakan ataupun diakses secara sembarangan [2][5].

Faktor keamanan menjadi hal yang sulit untuk dipecahakan, dimana teknik kriptografi yang beredar saat ini sudah semakin banyak. Namun penggunaan kriptografi dalam teknologi informasi masih sangat sedikit sekali. Apalagi untuk pengguna yang belum mengetahui fungsi keamanan data sendiri [1][7]. Masing-masing jalur komunikasi memiliki kekurangan dalam hal keamanan yang menimbulkan ancaman berupa kehilangan atau kebocoran data. Sehingga, dibutuhkan sistem keamanan pada jalur komunikasi untuk menjamin kerahasian data [4][6]. Selanjutnya untuk memudahkan dalam proses informasi pesan memo, maka penggunaan teknologi mobile menjadi prioritas utama. Kemudahan penggunaan *smartphone* sudah menjadi keharusan bagi perancang maupun bagi para pembuat aplikasi atau *software* (*programmer*). Pengembangan aplikasi *mobile* harus ingat bahwa dengan tersedianya berbagai *platform smartphone* yang berbeda-beda dan membangun aplikasi untuk *smartphone* tersebut tentunya akan memakan waktu dan biaya yang mahal. Selain itu juga pada proses pengembangan tentunya juga akan menemukan berbagai bentuk permasalahan dikarenakan beda *platform* pasti beda bahasa pemrograman yang digunakan.

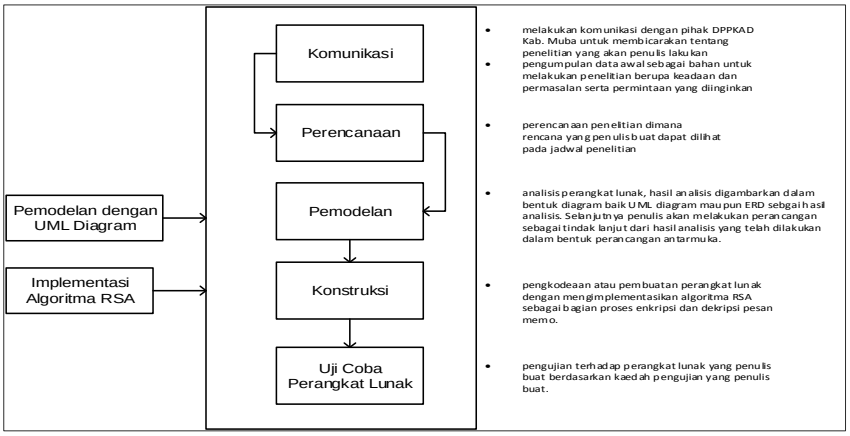
Berdasarkan uraian diatas maka solusi yang dapat dibuat yaitu menggunakan perangkat lunak khusus yang dapat diakses menggunakan perangkat *mobile*. Dimana didalam perangkat lunak tersebut pengirim dan penerima memo dapat langsung berkomunikasi dengan data yang telah dienkripsi atau diacak. Sehingga orang yang tidak berkepentingan tidak dapat membaca atau melihat pesan memo yang dikirim dengan benar. Untuk membuat pesan memo agar tidak terbaca oleh orang yang tidak memiliki hak, maka dalam perangkat lunak tersebut di implementasikan sebuah algoritma kriptografi yaitu algoritma RSA. Dalam penelitian ini akan mengimplementasikan algoritma RSA menggunakan *Chinese Remainder Theorem* (CRT) untuk mempercepat proses enkripsi dan dekripsi. CRT merupakan teorema dalam aritmatika modulo yang akan digunakan dalam melakukan operasi modulo pada proses enkripsi dan dekripsi.

2. METODOLOGI PENELITIAN

Metode yang digunakan dalam penelitian ini yaitu menggunakan metode penelitian deskriptif. Metode penelitian deskriptif adalah salah satu metode penelitan yang banyak digunakan pada penelitian yang bertujuan untuk menjelaskan suatu kejadian. Seperti yang dikemukakan oleh [6] bahwa "penelitian desktiptif adalah sebuah penelitian yang bertujuan untuk memberikan atau menjabarkan suatu keadaan atau fenomena yang terjadi saat ini dengan menggunakan prosedur ilmiah untuk menjawab masalah secara aktual". Sedangkan untuk proses pengembangan sistem menggunakan metode *web engineering* dengan beberapa tahapan yaitu komunikasi, perencanaan, pemodelan, konstruksi, dan ujicoba perangkat lunak [3].

2.1 Pola Pikir

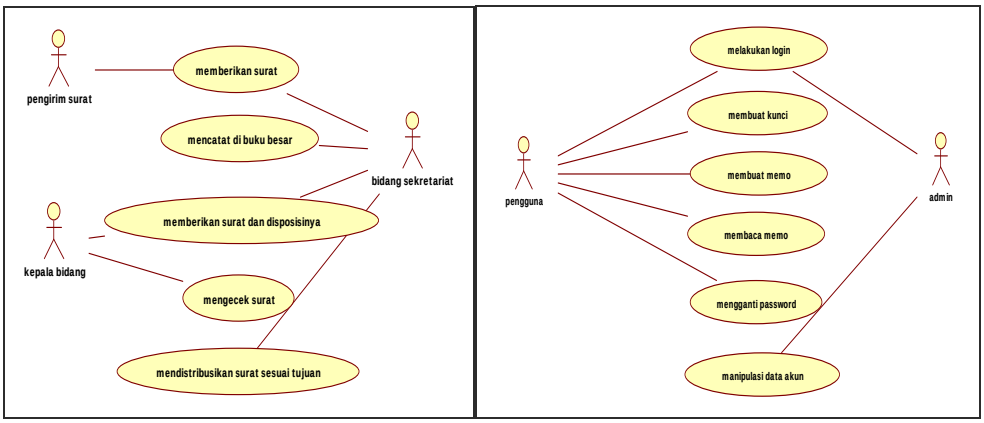
Sesuai dengan metode pengembangan sistem yang diterapkan dalam penelitian ini yaitu *web engineering*, berikut pola pikir yang dijalankan dalam penelitian ini sesuai gambar 1 berikut.



Gambar 1. Pola Pikir Yang Diterapkan Dalam Penelitian Ini

Prosedur yang diterapkan pada manajemen surat DPPKAD Kabupaten Musi Banyuasin dilakukan secara konvensional. Surat yang masuk diproses oleh bidang Sekretariat dan diteruskan ke bidang-bidang lain. Dokumentasi surat hanya berupa penulisan di buku besar, lalu dibuat didisposisi oleh Kepala Dinas berupa memo/nota dinas. Kemudian memo tersebut diteruskan ke bidang lain dan diletakkan di ruang kepala bidang sesuai tujuan.

Orang yang tidak memiliki hak untuk mengetahui isi memo dapat melihat atau membaca isi memo tersebut dan pada akhirnya jika isi memo tersebut bersifat penting dan rahasia maka hal tersebut sulit untuk dijaga kerahasiaannya. Selain itu, permasalahan lainnya yaitu ketika Kepala Dinas atau Kepala Bidang yang bersangkutan sedang tidak ada ditempat dan surat yang masuk bersifat segera dan rahasia, maka kegiatan akan tertunda. Untuk lebih jelas dari alur sistem yang ada dapat dilihat pada gambar 2 berikut.



(a) (b)

Gambar 2. Alur Sistem ((a) Alur Sistem Yang Berjalan Saat Ini, (b) Alur Sistem Yang Diusulkan Dalam Penelitian)

Berdasarkan sistem yang sudah berjalan diusulkan (gambar 2 poin b) sebuah sistem yaitu menggunakan perangkat lunak khusus yang dapat diakses menggunakan perangkat mobile. Dimana didalam perangkat lunak tersebut pengirim dan penerima memo dapat langsung berkomunikasi dengan data yang telah dienkripsi atau diacak. Sehingga orang yang tidak berkepentingan tidak dapat membaca atau melihat pesan memo yang dikirim dengan benar. Untuk membuat pesan memo agar tidak terbaca oleh orang yang tidak memiliki hak, maka dalam perangkat lunak tersebut di implementasi sebuah algoritma kriptografi yaitu algoritma RSA-CRT.

2.2 Analisis Proses Pada Algoritma RSA

Terdapat 3 proses yang dilakukan dalam proses algoritma ini yaitu:

a) Pembentukan kunci
Proses pembangkitan kunci pada kriptografi RSA adalah sebagai berikut :

1) Pilih dua buah bilangan prima sembarang p dan q. Jaga kerahasiaan p dan q.
2) Hitung $n = p * q$. Besaran n ini tidak perlu dirahasiakan.
3) Hitung $m = (p-1) * (q-1)$. Sekali m telah dihitung, p dan q dapat dihapus untuk mencegah diketahuinya oleh pihak lain.
4) Pilih sebuah bilangan bulat untuk kunci publik, sebut namanya e, yang relatif prima terhadap m (relatif prima berarti GCD(e, m) = 1) dengan syarat $e \neq (p-1)$, $e \neq (q-1)$, dan $e < n$.
5) Hitung kunci dekripsi, d, dengan kekongruenan $ed \equiv 1 \pmod{m}$.

b) Proses Enkripsi

Setelah proses proses pembangkitan kunci selesai, kemudian lanjut ke proses enkripsi pesan menggunakan kunci publik dari hasil pembangkitan kunci dengan menggunakan rumus (1) berikut:

$$C = Pe \bmod n \quad (1)$$

Menggunakan kunci yang diperoleh di atas kita akan mencoba untuk malakukan enkripsi pesan sederhana. Misalnya P = 48, maka akan diperoleh C seperti pada bentuk (2) berikut:

$C = Pe \bmod n = 485 \bmod 65 = 254803968 \bmod 65 = 3$ (2)

Jadi hasil enkripsi 48 menggunakan kunci yang diperoleh di atas adalah 3.

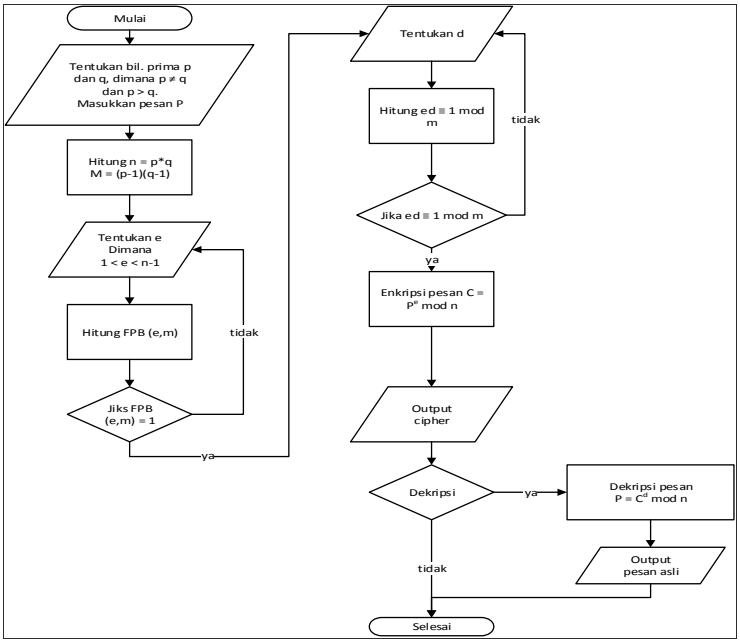
c) Proses Dekripsi
Untuk mengembalikan pesan *ciphertext* menjadi *plaintext* (pesan asli) adalah dengan menggunakan rumus (3) untuk dekripsi RSA sebagai berikut:

$$P = CD \bmod n \quad (3)$$

Dengan menggunakan pesan hasil enkripsi dan kunci yang diperoleh di atas dapat dilakukan dekripsi pesan seperti pada bentuk (4) berikut:

$$P = Cd \bmod n = 329 \bmod 65 = 68630377364883 \bmod 65 = 48 \quad (4)$$

Dari hasil dekripsi di atas dapat dibuktikan bahwa hasil enkripsi pesan dapat didekripsi kembali ke pesan asli.



Gambar 3. Flowchart Proses Algoritma RSA

2.3 Analisis Proses Pada Algoritma RSA-CRT

Terdapat 2 proses penting yang akan dibahas dalam proses algoritma ini yaitu proses pembentukan kunci dan proses dekripsi, sedangkan proses enkripsi tidak dibahas karena sama dengan proses algoritma RSA diatas.

a) Pembentukan kunci

Pembentukan kunci pada RSA-CRT berbeda dengan pembentukan kunci pada RSA standar. Pembentukan kunci pada RSA-CRT adalah sebagai berikut :

1) Misalkan *p* dan *q* adalah dua bilangan prima yang sangat besar dengan ukuran yang hampir sama dimana $p > q$.
2) Hitung $n = p*q$ dan $m = (p-1)(q-1)$.
3) Pilih sebuah bilangan bulat untuk kunci publik, sebut namanya *e,* yang relatif prima terhadap *m* (relatif prima berarti GCD(e, m) = 1) dengan syarat $e \neq (p-1)$, $e \neq (q-1)$, dan $e < n$.
4) Hitung nilai d dengan rumus (5) berikut.

$$d = e^{-1} \bmod m = \frac{1+km}{e} \quad (5)$$

5) Hitung nilai dp dengan rumus (6) berikut.

$$dp = d \bmod (p-1) \text{ dan } dq = d \bmod (q-1) \quad (6)$$

Kunci publik adalah <*n,e*> dan kunci rahasia adalah <*p,q,dp,dq*>.

b) Proses Dekripsi pada RSA-CRT

Karena enkripsi RSA-CRT sama dengan prosedur enkripsi RSA standar, saat ini perhatian difokuskan pada dekripsi RSA-CRT. Pada deskripsi RSA standar kita mendekripsi pesan dengan menggunakan rumus $P = C^d \bmod n$, sehingga perhitungan tersebut tergantung pada nilai d dan n. Jika nilai d besar, maka perhitungannya akan lebih lama karna nilai eksponen yang besar d. Sedangkan pada RSA-CRT kita menggunakan d hanya untuk membangkitkan kunci dp dan dq dimana dp dan dq akan lebih kecil nilainya dari d karna perhitungannya menggunakan d modulus p dan q. Rumus (7) berikut digunakan untuk menghitung dp dan dq.

$$\begin{aligned} &d \bmod (p-1) = e - 1 \bmod (p-1) \\ &d \bmod (q-1) = e - 1 \bmod (q-1) \\ &dp = e - 1 \bmod (p-1) = d \bmod (p-1) \\ &dq = e - 1 \bmod (q-1) = d \bmod (q-1) \end{aligned} \quad (7)$$

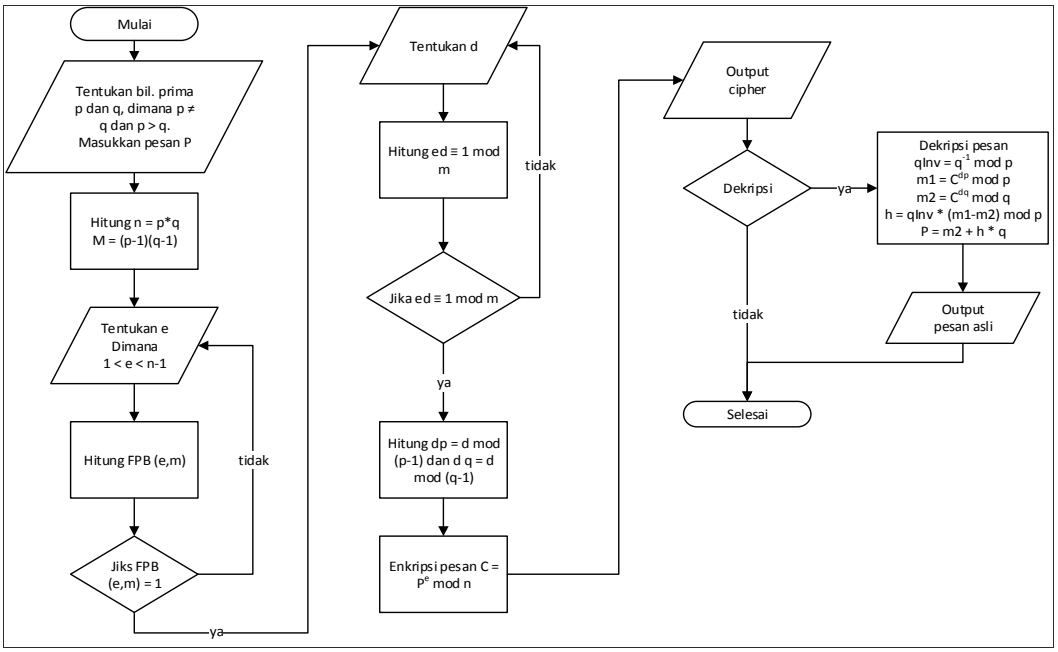
Dari hasil perhitungan di atas akan didapatkan nilai dp dan dq yang lebih kecil dari d. Selanjutnya kita menghitung representasi pesan $m1$ dan $m2$ yang akan digunakan untuk perhitungan akhir proses dekripsi menggunakan rumus (8) sebagai berikut:

$$\begin{aligned} &m1 = c^{dp} \bmod p \\ &m1 = c^{qp} \bmod p \end{aligned} \quad (8)$$

$m1$ dan $m2$ dari hasil perhitungan di atas kita subsitusikan ke dalam rumus Garner's (9) untuk menghitung solusi akhir dekripsi sebagai berikut:

$$\begin{aligned} &qInv = \left(\frac{1}{q}\right) mod\ p = 1 \div \frac{kp}{q} \\ &h = qInv(m1 - m2) \bmod p \\ &m = m2 + h.q \end{aligned} \quad (9)$$

Hasil perhitungan m di atas adalah solusi akhir dari proses dekripsi dimana m akan bernilai sama dengan pesan sebelum di enkripsi. Berikut seperti pada gambar 4 *flowchart* algoritma RSA-CRT.



Gambar 4. Flowchart Proses Algoritma RSA-CRT

3. HASIL DAN PEMBAHASAN

Hasil dari penelitian yang telah dilakukan berupa sistem pengirim pesan memo yang ditambahkan algoritma kriptografi RSA-CRT sebagai pengaman isi memo. Sistem ini sendiri memiliki menu-menu yaitu menu untuk membuaat kunci publik dan kunci privat, menu untuk membuat dan mengirim memo, serta menu untuk membaca memo yang telah dienkripsi. Sistem juga telah dilakukan pengujian

menggunakan metode *blackbox testing*. Dimana dari hasil pengujian mengunjukkan semua fungsi dapat berjalan dengan baik sesuai dengan fungsinya. Dengan demikian dapat disimpulkan bahwa sistem ini telah sesuai dengan yang diharapkan. Berikut pada gambar 5 hasil aplikasi keamanan e-memo.

Gambar 5. Hasil Aplikasi Keamanan Sistem Informasi E-Memo

Pada aplikasi yang dibuat terdapat menu untuk menentukan kunci, fungsinya untuk membuat kunci publik dan kunci privat karena RSA merupakan algoritma kunci asimetris sehingga menggunakan kunci publik. Sebelum menggunakan aplikasi, pengguna wajib menentukan nilai p dan nilai q agar dapat digunakan pada proses enkripsi dan dekripsi, hasil pembuatan kunci dan data akan disimpan ke *database* berupa data *chipertext*. Untuk membuat memo, pengguna dapat memilih tujuan pengirim dari daftar yang sudah dan juga pengguna harus memasukkan kunci publik yaitu nilai n dan nilai e dari data yang telah dimasukkan sebelumnya. Pada gambar 6 berikut memperlihatkan *interface* dari proses *input* memo.

Gambar 6. *Interface* Proses Enkripsi Pada Perangkat *Mobile*

Pada sistem keamanan informasi ini, satu kunci hanya dapat digunakan untuk 1 memo. Jika akan mengirimkan memo baru, maka pengguna diwajibkan untuk mengganti kunci yang lainnya. Selanjutnya, untuk menguji sistem telah berjalan sesuai yang diharapkan atau tidak, maka perlu dilakukan pengujian sederhana menggunakan metode *blackbox*. Metode ini menguji kemampuan program dalam menjalankan instruksi yang kita harapkan berdasarkan masukan, proses dan keluaran. Pengujian digunakan untuk menemukan kesalahan–kesalahan dan memastikan bahwa input yang dibatasi akan memberikan hasil aktual yang sesuai dengan hasil yang dibutuhkan. Pada tabel 1 sampai tabel 4 berikut memperlihatkan hasil dari pengujian fungsionalitas dari aplikasi keamanan sistem informasi e-memo, dimana hasil pengujian memperlihatkan berhasil atau tidaknya sistem dalam melakukan perintah login, *generate key*, *encrypt, decript*.

a) Pengujian *login*

Tabel 1. Pengujian *login*

No	*Data Masukkan*	*Hasil yang diharapkan*	*Hasil pengujian*	*Keterangan*
1	*Input username dan password secara benar*	*Ketika data login dimasukkan dan tombol login diklik maka dilakukan proses validasi data login. Apabila valid maka pengguna bisa mengakses halaman utama.*	*Pengguna dapat login ke dalam sistem dan mengakses menu utama*	***Berhasil*** */ Tidak*
2	*Input username dan password yang salah*	*Sistem menampilkan pesan kesalahan karena data tidak valid "Username dan Password tidak benar"*	*Sistem menampilkan pesan kesalahan "Username dan Password salah"*	***Berhasil*** */ Tidak*

b) Pengujian *generate key*

Tabel 2. Pengujian *generate key*

No	*Data Masukkan*	*Hasil yang diharapkan*	*Hasil pengujian*	*Keterangan*
1	*Mengisi nilai p dan nilai q*	*Setelah nilai p dan q dimasukkan dan tombol lihat hasil ditekan. Maka sistem akan menampilkan hasil berupa kunci privat dan publik.*	*Sistem menampilkan kunci privat dan publik*	***Berhasil** / Tidak*
2	*Pengguna menekan tombol simpan untuk menyimpan kuncih*	*Sistem menyimpan kunci ke database dan menampilkan pesan kunci berhasil disimpan*	*sistem menampilkan pesan "kunci berhasil dibuat" dan selanjutnya menyimpan kunci ke dalam database*	***Berhasil** / Tidak*

c) Pengujian *encrypt*

Tabel 3. Pengujian *encrypt*

No	*Data Masukkan*	*Hasil yang diharapkan*	*Hasil pengujian*	*Keterangan*
1	*Pengguna memilih tujuan pengiriman memasukkan kunci publik (nilai n dan nilai e) dan menekan tombol kirim*	*Sistem menyimpan data ke database dan menampilkan pesan berhasil*	*Sistem menyimpan memo ke database dan menampilkan pesan "Memo berhasil dibuat"*	***Berhasil** / Tidak*

d) Pengujian *decrypt*

Tabel 4. Pengujian *decrypt*

No	*Data Masukkan*	*Hasil yang diharapkan*	*Hasil pengujian*	*Keterangan*
1	*Pengguna memilih judul memo selanjutnya memasukkan kunci privat (nilai p, q dan nilai d) dan menekan tombol dekripsi*	*Sistem akan menampilkan pesan yang sudah didekripsi*	*Sistem menampilkan pesan yang sudah didekripsi*	***Berhasil** / Tidak*

Dari hasil pengujian yang terlihat pada tabel maka dapat disimpulkan bahwa semua fungsionalitas sistem telah berjalan dengan benar sesuai harapan. Selain itu dilakukan pengujian algoritma kriptografi RSA-CRT pada aplikasi e-memo dengan membandingkan kecepatan dekripsi antara algoritma kriptografi RSA dengan algoritma kriptografi RSA-CRT melalui perangkat mobile. Pada tabel 5 dan 6 berikut menunjukkan beberapa hasil ujicoba data menggunakan 2 algoritma RSA tersebut dari sisi kecepatan performa.

Tabel 5. Perbandingan algoritma RSA dan RSA-CRT

Nilai p	*Nilai q*	*Karakter Pesan*	*Dekripsi RSA (s)*	*Dekripsi RSA-CRT (s)*
431	*311*	*30*	*1.438530921936*	*0.00250005722045*
431	*311*	*60*	*3.8675880432129*	*0.007000923156738*
431	*311*	*90*	*4.5791049003601*	*0.007499933242797*
983	*967*	*30*	*7.8121421337128*	*0.007499933242797*
983	*967*	*60*	*18.669845104218*	*0.01700091361999*
983	*967*	*90*	*36.184340000153*	*0.0880048274993*
2671	*1213*	*30*	*36.897782802582*	*0.01500010490417*
2671	*1213*	*60*	*74.975780010223*	*0.02400112152099*
2671	*1213*	*90*	*136.16322994232*	*0.05400300025939*
5413	*3121*	*30*	*561.57491111755*	*0.08900499343872*
5413	*3121*	*60*	*1035.5359208584*	*0.1420009136*
5413	*3121*	*90*	*1288.3005919456*	*0.1900010108947*

Tabel 6. Perbandingan algoritma RSA dan RSA-CRT dari segi kecepatan

Nilai p	*Nilai q*	*Karakter Pesan*	*Dekripsi RSA (s)*	*Dekripsi RSA-CRT (s)*	*Kecepatan (RSA / RSA-CRT)*
431	*311*	*30*	*1.438530921936*	*0.002500057220459*	*575 kali lebih cepat*
431	*311*	*60*	*3.8675880432129*	*0.0070009231567383*	*552 kali lebih cepat*
431	*311*	*90*	*4.5791049003601*	*0.0074999332427979*	*611 kali lebih cepat*
983	*967*	*30*	*7.8121421337128*	*0.0074999332427979*	*1042 kali lebih cepat*
983	*967*	*60*	*18.669845104218*	*0.017000913619995*	*1098 kali lebih cepat*
983	*967*	*90*	*36.184340000153*	*0.08800482749939*	*411 kali lebih cepat*
2671	*1213*	*30*	*36.897782802582*	*0.015000104904175*	*2460 kali lebih cepat*
2671	*1213*	*60*	*74.975780010223*	*0.024001121520996*	*3124 kali lebih cepat*
2671	*1213*	*90*	*136.16322994232*	*0.054003000259399*	*2521 kali lebih cepat*
5413	*3121*	*30*	*561.57491111755*	*0.089004993438721*	*6309 kali lebih cepat*
5413	*3121*	*60*	*1035.5359208584*	*0.14200091362*	*7292 kali lebih cepat*
5413	*3121*	*90*	*1288.3005919456*	*0.19000101089478*	*6780 kali lebih cepat*
	Rata-Rata				***2731 kali lebih cepat***

Dari pengujian algoritma kriptografi RSA dan algoritma RSA-CRT dapat dilihat semakin panjang kunci p, q dan pesan maka proses dekripsi memo akan semakin lama juga. Kecepatan yang diperolah rata-rata lebih cepat ketika menggunakan dekripsi RSA-CRT dibandingkan menggunakan dekripsi RSA. Dengan demikian dapat disimpulkan bahwa CRT dapat meningkatkan kecepatan proses dekripsi RSA.

4. KESIMPULAN

Berdasarkan hasil penelitian dan pembahasan dapat ditarik kesimpulan yaitu algoritma kriptografi RSA dapat diaplikasikan pada aplikasi e-memo untuk mengamankan informasi pesan. Algoritma RSA yang sudah dimodifikasi dengan RSA-CRT memiliki keuntungan dalam kecepatan proses bila dibandingkan dengan algoritma RSA standar, hal ini tentunya memberikan performa yang lebih baik tentunya karena ujicoba dilakukan pada perangkat *mobile/smartphone* yang secara komputasi memiliki keterbatasan performa dibandingkan perangkat komputer desktop. Proses dekripsi menggunakan algoritma RSA-CRT untuk masing-masing panjang pesan sebanyak 30, 60, dan 90 karakter untuk masing-masing pasangan nilai p dan q memiliki kecepatan rata-rata 2731 kali lebih cepat dibandingkan menggunakan algoritma RSA standar.

DAFTAR PUSTAKA

[1] Muzakir, Ari. (2014). *"Prototype Model Keamanan Data Menggunakan Kriptografi Data Encryption Standar (DES) Dengan Mode Operasi Chiper Block Chaining (CBC)"*. Seminar Nasional Inovasi dan Tren(SNIT). Jakarta.

[2] Kharisma, R. S., & Rachman, M. A. F. (2017). *"Pembuatan Aplikasi Notes Menggunakan Algoritma Kriptografi Polyalphabetic Substitution Cipher Kombinasi Kode Ascii Dan Operasi Xor Berbasis Android"*. *Jurnal Teknologi Informasi Respati*, *12*(2).

[3] Pressman, Roger S. (2012). *Rekayasa Perangkat Lunak.* Yogyakarta. Penerbit Andi.

[4] Rakhim, R. (2010). "*Keamanan Web Service Menggunakan Token "*.Tesis S2 Magister Ilmu Komputer. Universitas Gadjah Mada. Yogyakarta.

[5] Rifai, R. Y., Christyono, Y., & Santoso, I. (2016). *"Implementasi Algoritma Kriptografi Rivest Code 4, Rivest Shamir Adleman, Dan Metode Steganografi Untuk Pengamanan Pesan Rahasia Pada Berkas Teks Digital"*. *Transient*, *5*(1), 86-91.

[6] Sugiyono. (2011). *Metode Penelitian Kuantitatif, Kualitatif dan R&D*. Bandung. Alfabeta.

[7] Sari, DR., Kunang, YN., dan Muzakir, A. (2015). *"Sistem Keamanan SSO Berbasis SAML pada Jalur Komunikasi dengan Menggunakan XML Encryption"*. Student Colloquium Sistem Informasi & Teknik Informatika (SC-SITI). Palembang