PENERAPAN KRIPTOGRAFI DENGAN MENGGUNAKAN ALGORITMA RSA UNTUK PENGAMANAN DATA BERBASIS *DESKTOP* PADA PT TRIAS MITRA JAYA MANUNGGAL

Muhamad Rizki[1)], Pipin Farida Ariyani[2)]

Teknik Informatika, Fakultas Teknologi Informasi, Universitas Budi Luhur
Jl. Raya Ciledug, Petukangan Utara, Kebayoran Lama, Jakarta Selatan 12260
E-mail : 1511501981@student.budiluhur.ac.id[1)], pipin.faridaariyani@budiluhur.ac.id[2)]

Abstrak

PT Trias Mitra Jaya Manunggal adalah sebuah kantor yang bergerak dalam bidang penyediaan jasa layanan mesin hitung uang, mesin sortir serta service dan maintenance. Dimana kantor tersebut memiliki data penting seperti data populasi mesin yang terdapat informasi tentang titik-titik bank yang menyewa mesin hitung uang, mesin sortir dan mesin deteksi valas serta harga dari penyewaan mesin tersebut. Mengingat data ini tidak boleh diketahui oleh kompetiror karena bisa merugikan pada saat melakukan tender, maka diperlukan sebuah pengamanan data agar terhindar dari pencurian data. Oleh karena itu agar pertukaran data dan informasi dapat dilakukan secara aman diperlukan sebuah aplikasi dengan metode yang dapat melindungi data dan informasi yang berada didalamnya. Metode yang dimaksud adalah kriptografi dengan algoritma RSA (Ron Rives, Adi Shamir dan Leonard Adleman). Secara umum kriptografi adalah teknik mengamankan data dengan cara mengubah atau mengolah informasi asli (plaintext) menggunakan kunci dan dengan sebuah metode enkripsi tertentu sehingga menghasilkan informasi yang baru (chipertext) yang sulit dibaca secara langsung. Agar dapat terbaca kembali, chipertext tersebut harus di dekripsi dengan kunci tertentu agar kembali menjadi plaintext. Maka dari itu dibangun suatu aplikasi yang mampu mengamankan data dengan bahasa pemrograman Java. Dari hasil pengujian yang dilakukan, aplikasi dapat merubah isi *file* asli (*plaintext*) menjadi karakter yang sulit dipahami (*chipertext*) sehingga *file* terjaga keamanannya dan kecepatan penggunaan aplikasi sangat bergantung dengan ukuran *file* yang akan dienkripsi maupun didekripsi.

Kata kunci: Kriptografi, RSA, Enkripsi, Dekripsi

1. PENDAHULUAN

Perkembangan teknologi komputerisasi dan informasi yang cepat memudahkan kita dalam mengirim, menerima dan bertukar data melalui media elektronik dengan cepat. Salah satu dampak negatif dari kemajuan teknologi informasi adalah pencurian data. PT Trias Mitra Jaya Manunggal adalah salah satu kantor penyedia jasa layanan mesin hitung uang, mesin sortir, mesin deteksi valas serta *service* dan *maintenance*. Dimana kantor tersebut memiliki data penting seperti data populasi mesin yang terdapat informasi tentang titik-titik bank yang menyewa mesin hitung uang, mesin sortir dan mesin deteksi valas serta harga dari penyewaan mesin tersebut. Mengingat data ini tidak boleh diketahui oleh kompetiror karena bisa merugikan pada saat melakukan tender, maka diperlukan sebuah pengamanan data agar terhindar dari pencurian data.

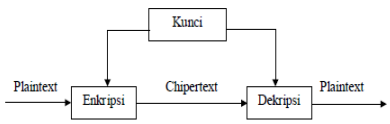
Dalam penelitian dan penyusunan ini menggunakan metode kriptografi dengan algoritma RSA (Ron Rivest, Adi Shamir dan Leonard Adleman) sebagai proses enkripsi dan dekripsi *file*. Pemilihan algoritma RSA dalam metode kriptografi dikarenakan sistem penyandian pada algoritma RSA memiliki mekanisme kerja yang baik.

2. PENELITIAN SEBELUMNYA

2.1. Definisi Kriptografi

Kata Kriptografi berasal dari bahasa Yunani,"Kriptos" berarti tersembunyi atau rahasia dan "graphien" yang berarti menulis. Dari sini dapat diartikan bahwa kriptografi adalah suatu cara atau praktek untuk memperoleh komunikasi yang aman walaupun ada pihak ketiga yang ingin berusaha mencuri.[1]

Kriptografii merupakan bidang ilmu pengetahuan yang menggunakan persamaan matematis untuk melakukan proses enkripsi (*encrypt*) dan dekripsi (*decrypt*) pesan. Teknik ini dipakai untuk mengubah pesan kedalam kode-kode tertentu sehingga informasi yang ditransmisikan melalui jaringan yang mungkin tidak aman (misalnya saja internet) tidak dapat dibaca oleh pihak manapun kecuali orang-orang yang berhak. Berikut ini adalah proses kriptografi secara umum dapat dilihat pada Gambar 1 dibawah ini:



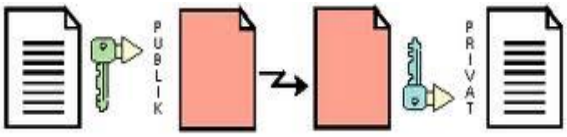
Gambar 1. Proses Kriptografi

2.2. Enkripsi dan Dekripsi

Enkripsi adalah proses yang melakukan perubahan sebuah kode yang bisa dimengerti menjadi sebuah kode yang tidak dimengerti (tidak terbaca). Enkripsi dapat diartikan sebagai sebuah kode atau *cipherteks*. Proses sebaliknya untuk mengubah *ciphertext* menjadi *plaintext* disebut dekripsi[3].

2.3. Kriptografi Asimetris

Pada pertengahan tahun 70-an Whitfield Diffiedan Martin Hellman menemukan teknik enkripsi asimetris yang merevolusi dunia kriptografi. Kunci asimetris adalah pasangan kunci-kunci kriptografi yang salah satunya diterapkan untuk proses enkripsi dan yang satu lagi untuk dekripsi. Semua orang yang menerima kunci publik dapat menggunakannya untuk mengenkripsikan suatu pesan, sedangkan hanya satu orang saja yang boleh memiliki rahasia tertentu dalam hal ini kunci private untuk melakukan pembongkaran terhadap sandi yang dikirim untuknya. Sebagai contoh jika Anto mengirim pesan kepada Badu, Anto dapat merasa yakin bahwa pesan tersebut hanya dapat dibaca oleh Budi, karena hanya Budi yang bisa melakukan dekripsi dengan kunci privatnya. Tentunya Anto harus memiliki kunci publik Budi untuk melakukan enkripsi. Anto bisa mendapatkannya dari Budi, ataupun dari pihak ketiga seperti Sari [4]. Seperti gambar 2 dibawah ini:



Gambar 2. Penggunaan Kunci Asimetris

Contoh algoritma terkenal yang menerapkan kunci Asimetris adalah algoritma RSA (merupakan singkatan penemunya yakni Rivest, Shamir dan Adleman).

2.4. Kriptografi RSA

Dari banyaknya algoritma kriptografi asimetris yang ada, algotirma yang sering dipakai adalah RSA. Algortima RSA dibuat oleh tiga orang peneliti dari MIT (Massachussets Institute of Technology) pada tahun 1976. Nama RSA merupakan singkatan dari nama tiga orang pembuatnya, yaitu Rivest, Shamir, dan Adleman. Algoritma RSA menggunakan pemfaktoran bilangan yang besar menjadi faktor-faktor prima. Pemfaktoran dilakukan untuk menemukan kunci privat [5].

Menurut [5] Algoritma RSA memiliki besaran-besaran sebagai berikut:

1. p dan q adalah bilangan prima (rahasia)
2. n = p.q (tidak rahasia)
3. (n) = (p –1).(q –1) (rahasia)
4. e (kunci enkripsi) (tidak rahasia) Syarat: PBB (e, (n)) = 1
5. d (kunci dekripsi) (rahasia) d dihitung dari d e-1 mod ((n))
6. m (plainteks) (rahasia)
7. c (cipherteks) (tidak rahasia)

Pembangkitan kunci :

1. Pilih dua bilangan prima, p dan q (rahasia)
2. Hitung n = p.q. Besaran n tidak perlu dirahasiakan.
3. Hitung (n) = (p–1).(q –1).
4. Pilih sebuah bilangan bulat untuk memperoleh kunci publik, sebut namanya e, yang relatif prima terhadap (n).
5. Hitung kunci dekripsi, d,melalui ed 1 (mod m) atau d e-1 mod (n)

Hasil dari algoritma di atas:

1. Kunci publik adalah pasangan (e, n)
2. Kunci privat adalah pasangan (d, n)

2.5. Penggunaan Algoritma RSA

Contoh penggunaan algoritma RSA misalkan p= 47 dan q= 71 (keduanya prima), selanjutnya menghitung nilai n = p . q= 47 . 71 = 3337, m = (p - 1).(q - 1) = (47– 1).(71 -1) = 3220 pilih kunci e = 79, karena 79 relatif prima dengan 3220, e dan n dapat diketahui ke umum. Kemudian akan dihitung kunci dekripsi d dengan menggunakan: d= e-1mod m atau e.d (mod m)=1 sehingga akan diperoleh (e=79 dan m=3220) : 79. d mod 3220 = 1 dengan membuktikan nilai-nilai d = 1,2,3,n, diperoleh nilai kunci pribadi yang bulat dengan 1019. Kunci dekripsi ini yang akan dirahasiakan [2].

Pesan yang akan dikirim adalah M =TEGUH atau dengan decimal (kode ASCII) adalah: 8469718572, nilai ini dipecah menjadi blok-blok m. Maka blok akan terbentuk adalah: m1 = 84; m2 = 69; m3 =71; m4 = 85; m5 = 72.Sebelumnya sudah diketahui kunci publik adalah e=79 dan n=3337. Maka pesan M bisa dienkripsikan, yakni: C1 = 8479 mod 3337= 1995; C2 = 6979 mod 3337 = 1689; C3 =7179 mod 3337 = 1988; C4 = 8579 mod 3337= 3048; C5 = 7279 mod 3337 = 285.Sehingga ciphertext yang diperoleh adalah : 1995 1689 1988 3048 285 [2].

Kemudian pesan yang sudah terenkripsi tersebut dikirimkan kepada penerima pesan, yang mana sudah memiliki kunci pribadi (d,m) = (1019,3337) sehingga: m1 = 19951019 mod 3337 =84; m2 = 16891019 mod 3337 = 69; m3 =19881019 mod 3337 = 71; m4 = 30481019 mod3337 = 85; m5 = 2851019 mod 3337 = 72; Maka akan diperoleh kembali M = 8469718572, yang dalam pengkodean ASCII dapat dibaca sebagai berikut: M = TEGUH [2].

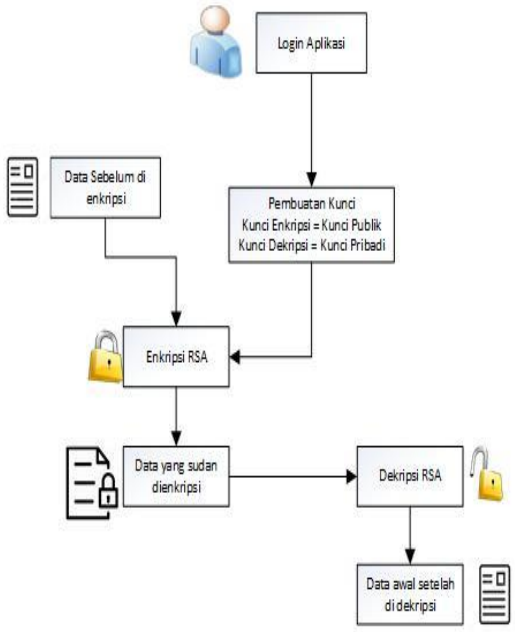
3. METODE PENELITIAN

Dalam penelitian ini, beberapa metode digunakan untuk mendapatkan informasi yang diperlukan untuk menyelesaikan masalah yang ditemui.Adapun metode-metode tersebut diantaranya sebagai berikut:

a. Studi Literatur
   Metode ini menggunakan pembelajaran dengan cara membaca, memahami dan mengumpulkan jurnal ilmiah serta referensi lain untuk mendapatkan informasi yang dibutuhkan dalam menunjang penelitian.
b. Analisa Data
   Metode ini digunakan untuk menganalisa algoritma kriptografi RSA.
c. Perancangan Sistem
   Merancang sistem aplikasi untuk menerapkan kriptografi algoritma RSA menggunakan bahasa pemrograman Java berbasis dekstop.
d. Pengujian Sistem
   Metode ini digunakan dengan cara menguji dan mengecek aplikasi.

3.1. Arsitektur Sistem

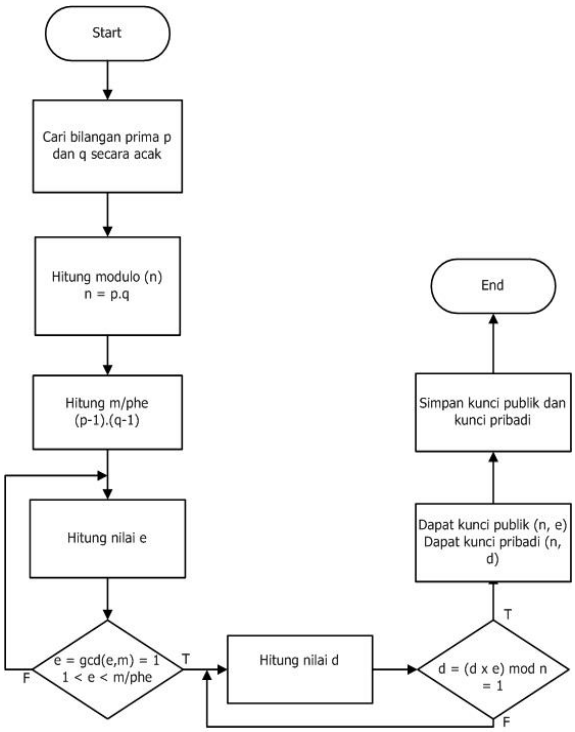
Arsitektur sistem menggambarkan garis besar proses dari sistem untuk memahami konsep aplikasi yang akan di bangun seperti gambar 3 dibawah ini :



Gambar 3. Arsitektur Sistem

3.2. Flowchart Proses Pembuatan Kunci RSA

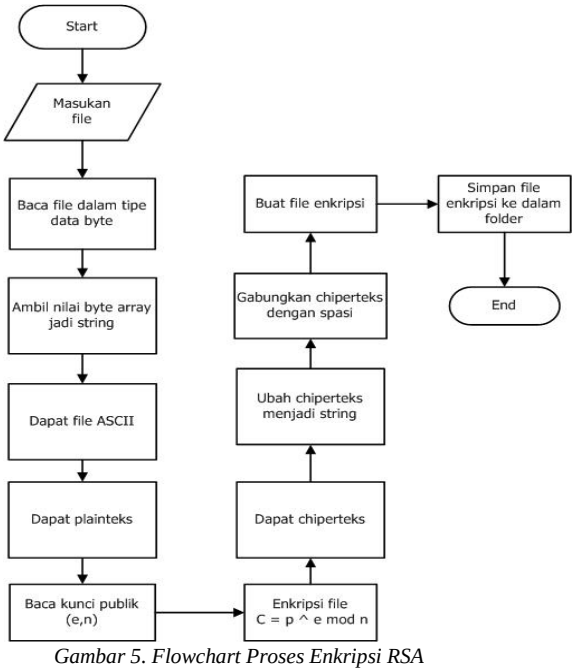
Pada proses pembuatan kunci RSA ada berberapa proses untuk mendapatkan kunci publik dan kunci pribadi. Berikut flowchart Seperti Gambar 4 dibawah ini :



Gambar 4. Flowchart Proses Pembuatan Kunci RSA

3.3. Flowchart Proses Enkripsi RSA

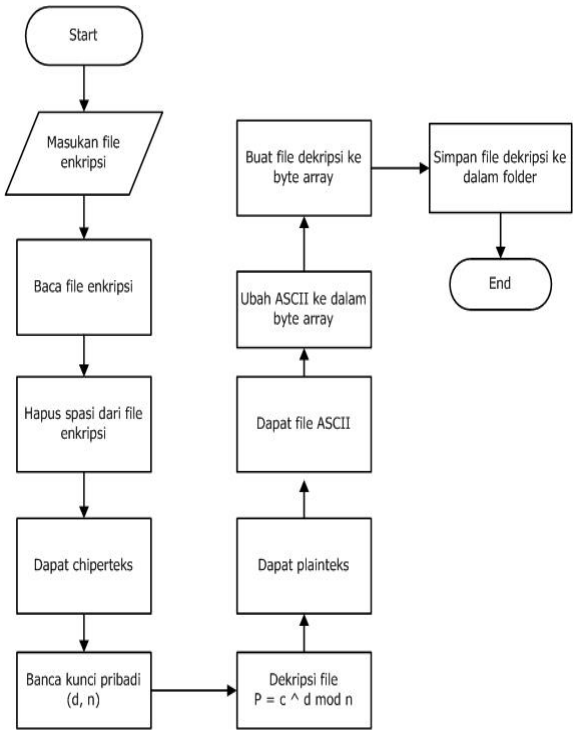
Pada proses enkripsi RSA ada berberapa proses untuk memperoleh hasil enkripsi. Berikut flowchart Seperti Gambar 5 dibawah ini :



Gambar 5. Flowchart Proses Enkripsi RSA

3.4. Flowchart Proses Dekripsi RSA

pada proses dekripsi RSA ini merupakan penjelasan gambaran alur proses dari dekripsi algoritma RSA seperti gambar 6 dibawah ini :



Gambar 6. Flowchart Proses Dekripsi RSA

4. HASIL DAN PEMBAHASAN

4.1 Tampilan Menu Utama

form menu utama terdapat tujuh *button* yang akan ditampilkan. Ketika *user* memilih salah satu *button*, *user* dapat menekan *button* sesuai dengan kebutuhan yang akan digunakan. *Form* menu utama akan tampil setelah *user* berhasil *login*. Di dalam menu utama terdapat button enkripsi, dekripsi, pembuatan kunci, laporan, info, bantuan dan keluar. Tampilan form menu utama bisa dilihat seperti pada Gambar 7.

Gambar 7 Tampilan Layar Menu Utama

4.2 Tampilan Form Pembuatan Kunci

Menu Pembuatan kunci ini digunakan untuk membuat kunci publik dan kunci pribadi.Tampilan Layar Form Pembuatan Kunci pada gambar 8 di bawah ini

Gambar 8 Tampilan Layar Form Pembuatan Kunci

4.3 Tampilan Form Enkripsi

Form enkripsi dapat digunakan oleh *user* untuk melakukan proses enkripsi file. Pertama *user* harus memilih file yang ingin dienkripsi. Setelah itu *user* memilih kunci publik yang telah dibuat sebelumnya. Selanjutnya *user* akan memilih tempat untuk penyimpanan *file*. *User* bisa memilih *button* enkripsi untuk melakukan proses enkripsi *file*. Berikut tampilan layar *form* enkripsi seperti pada Gambar 9

Gambar 9 Tampilan Layar Form Enkripsi

4.4 Tampilan Form Dekripsi

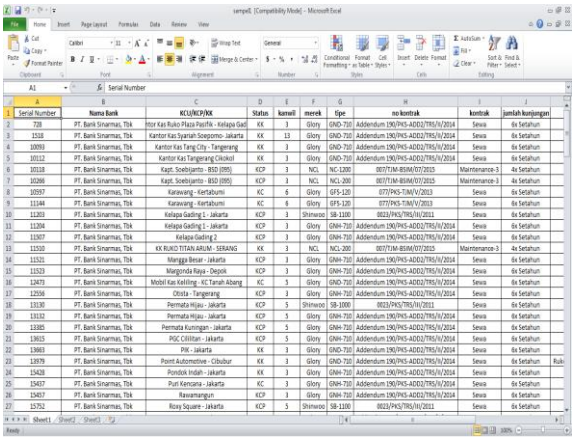
Form dekripsi *file* dipakai oleh *user* untuk mengembalikan hasil *file* enkripsi. Terdapat button dekrip, Bersihkan Field dan selesai yang mempunyai fungsi masing-masing. Pertama *user* akan memilih *file* yang sudah terenkripsi. Selanjutnya memilih kunci pribadi yang telah dibuat, lalu memilih tempat untuk penyimpanan *file* hasil dekripsi. Tampilan layar *form* dekripsi seperti pada Gambar 10.

Gambar 10 Tampilan Layar Form Dekripsi

4.5 Uji Coba Aplikasi

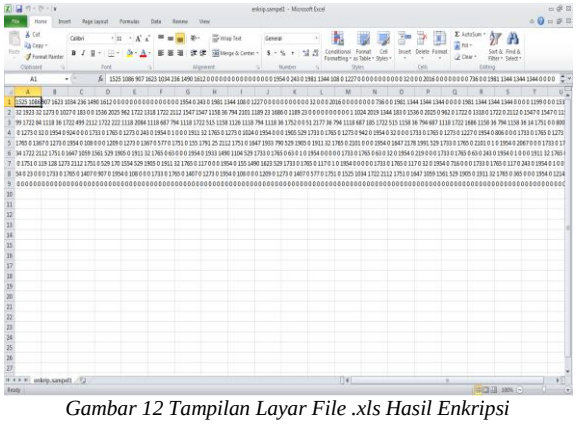
Gambar di bawah ini adalah perbandingan *file* .xls sebelu melakukan enkripsi dan hasil sesudah di melakukan enkripsi. Pada gambar 11 adalah tampilan layar *file* .xls sebelum melakukan proses enkripsi.

el



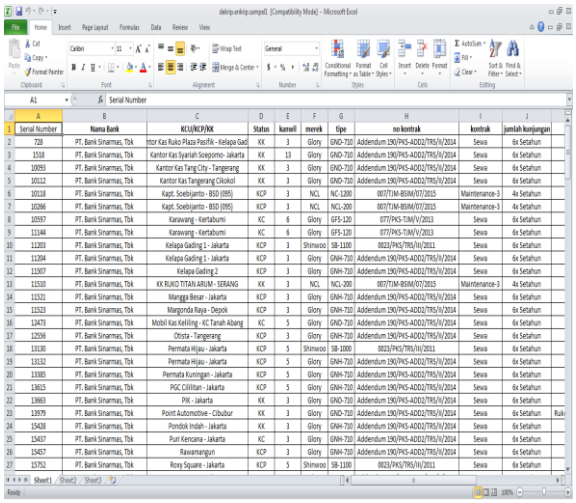
Gambar 11 Tampilan Layar File .xls Sebelum Enkripsi

Setelah *user* melakukan proses enkripsi maka isi *file* .xls akan menjadi angka-angka yang sulit dibaca isinya. Berikut adalah tampilan layar *file* .xls hasil proses setelah enkripsi seperti Gambar 12.



Gambar 12 Tampilan Layar File .xls Hasil Enkripsi

Setelah melakukan proses dekripsi, maka isi dari *file* .xls kembali bisa dibuka seperti awal secara utuh yang telah melakukan proses enkripsi sebelumnya pada *file* tersebut . Berikut ini adalah tampilan layar *file* .xls setelah melakukan proses dekripsi seperti Gambar 13.



Gambar 13 Tampilan Layar File .xls Hasil Dekripsi

4.6 Tabel Pengujian

Dalam pengujian akan dibahas mengenai perbandingan antara proses enkripsi dan dekripsi *file*. *File* yang diuji meliputi *file* yang berformat *.xlsx, .xls, .docx, .doc, .pptx. Pengujiannya yaitu antara lain ukuran *file* sebelum dienkripsi, waktu proses pada saatyenkripsi, hingga hasil yang dicapai dalam proses enkripsi. Berikut adalah tabel pengujian proses enkripsi file seperti pada tabel 1.

Tabel 1 Tabel Pengujian Proses Enkripsi *File*

Nama File	**Ukuran File**	**Waktu Enkripsi (Detik)**	**Hasil Enkripsi**
sampel1.xlsx	31	0.275	File tidak dapat dibuka
sampel1.xls	85	0.422	File teracak menjadi angka
sampel2.doc	274	0.632	File tidak dapat dibuka
sampel2.docx	394	0.824	File teracak menjadi angka
sampel3.pptx	82	0.354	File tidak bisa dibuka

5. KESIMPULAN DAN SARAN

5.1 Kesimpulan

Dari hasil pengujian dan analisa yang telah dilakukan dapat disimpulkan bahwa:

a. Aplikasi ini dapat merubah isi *file* asli (*plaintext*) menjadi karakter yang sulit dipahami (*chipertext*).
b. Aplikasi ini dapat mengembalikan isi *file* seperti awal secara utuh yang telah dilakukan proses enkripsi sebelumnya.
c. Kecepatan penggunaan aplikasi sangat bergantung dengan ukuran *file* yang akan dienkripsi maupun didekripsi.

5.2 Saran

Selain dari beberapa kesimpulan, dapat pula diajukan beberapa saran yang bisa dijadikan pertimbangan dalam pengembangan aplikasi, diantaranya:

a. Waktu proses enkripsi dan dekripsi diharpakan bisa lebih cepat.
b. Sebaiknya nilai p dan nilai q bisa diatur dengan angka yang besar, sehingga nilai d semakin sulit ditebak.

6. DAFTAR PUSTAKA

[1] Apdillah, Dicky. Dan Swanda, Heru. "Penerapan Kriptografi RSA Dalam Mengamankan File Berbasis Teks PHP". Jurnal Teknologi Informasi, vol. 2, no. 1, pp. 45-52, Jun. 2018

[2] Hariyanto., Nugraha, F.R., Lukman, S., Irawati D.R. "Aplikasi Enkripsi dan Dekripsi Pada Soal Ujian Menggunakan RSA Berbasis Java Desktop". Jurnal Ilmiah Komputasi, vol. 17, no. 3, pp 229-237, Sept. 2018

[3] Simargolang, M.Y. "Implementasi Kriptografi RSA Dengan PHP". Jurnal Teknologi Informasi vol 1, no. 1, pp 1-10, Jul. 2017.

[4] Basri. "Kriptografi Simetris Dan Asimetris Dalam perspektif Keamanan Data Dan Kompleksitas Komputasi. Jurnal Ilmiah Ilmu Komputer", vol 2 , no 2, pp 17-23, Sept. 2016.

[5] Arief Muhammad. "Kriptografi RSA Pada Aplikasi File Transfer Client-Server Based". Jurnal Ilmiah Teknologi Informasi Terapan, vol 1, no 3, pp 45-5, Aug. 2015.