RANCANG BANGUN APLIKASI KRIPTOGRAFI PADA TEKS MENGGUNAKAN METODE REVERSE CHIPER DAN RSA BERBASIS ANDROID

Yusfrizal[1)]

Universitas Potensi Utama
Jl. K.L.Yos Sudarso Km.6,5 No.3-A Medan (20241)
E-mail : yusfrizal80@gmail.com

ABSTRACT

Quick progress also occurs in the field of cellular communication networks with the concept of open systems, making it easier for someone to enter into the network. This can cause the process of sending data to be unsafe because it can be used by other parties who are not responsible for taking data and information in the middle of the road. To achieve the goal of maintaining confidentiality, various kinds of security techniques have been developed. To protect and maintain the confidentiality of data to avoid people who are not entitled to obtain this information, namely using cryptographic methods. In terms of data security techniques, many cryptographic methods can be used. Cryptographic methods have their own techniques and methods. One of the cryptographic methods that can be used is the Reverse Cipher method. But if you only use the Reverse Cipher method, text data security is very weak. So to achieve a higher level of security this method is combined with the RSA method which uses a public key and has high security because the difficulty of factoring large numbers becomes prime factors in the RSA algorithm.

Keywords: Cryptography, Reverse Cipher, RSA

ABSTRAK

Kemajuan yang pesat juga terjadi di bidang jaringan komunikasi seluler dengan konsep open system-nya, sehingga memudahkan seseorang untuk masuk ke dalam jaringan tersebut. Hal tersebut dapat mengakibatkan proses pengiriman data menjadi tidak aman karena dapat dimanfaatkan oleh pihak lain yang tidak bertanggung jawab untuk mengambil data maupun informasi di tengah jalan. Untuk mencapai tujuan terjaganya kerahasiaan tersebut, berbagai macam teknik keamanan telah dikembangkan. Untuk melindungi dan menjaga kerahasiaan data agar terhindar dari orang yang tidak berhak mendapatkan informasi tersebut, yaitu menggunakan metode kriptografi. Dalam hal teknik pengamanan data, banyak metode kriptografi yang dapat digunakan. Metode – metode kriptografi tersebut mempunyai teknik dan cara tersendiri Salah satu metode kriptografi yang bisa digunakan adalah metode Reverse Cipher. Tetapi jika hanya menggunakan metode Reverse Cipher saja keamanan data teks sangatlah lemah. Maka untuk mencapai tingkat keamanan yang lebih tinggi metode ini dikombinasikan dengan metode RSA yang menggunakan kunci publik serta memiliki keamanan yang cukup tinggi karena sulitnya memfaktorkan bilangan yang besar menjadi faktor-faktor prima pada algoritma RSA tersebut.

Kata kunci: Kriptografi, Reverse Cipher, RSA

I. PENDAHULUAN

Penggunaan teknologi komputer dan telekomunikasi pada saat ini telah mengubah cara pandang masyarakat dalam berkomunikasi. Kemajuan yang pesat juga terjadi di bidang jaringan komunikasi seluler dengan konsep *open system*-nya, sehingga memudahkan seseorang untuk masuk ke dalam jaringan tersebut. Hal tersebut dapat mengakibatkan proses pengiriman data menjadi tidak aman karena dapat dimanfaatkan oleh pihak lain yang tidak bertanggung jawab untuk mengambil data maupun informasi di tengah jalan[1]. Masalah keamanan data merupakan suatu aspek penting dalam pengiriman data teks maupun informasi melalui jaringan seluler. Oleh karena itu, dibutuhkan suatu sistem keamanan data yang dapat menjaga kerahasiaan suatu data teks maupun informasi.

Untuk mencapai tujuan terjaganya kerahasiaan tersebut, berbagai macam teknik keamanan telah dikembangkan. Untuk melindungi dan menjaga kerahasiaan data agar terhindar dari orang yang tidak berhak mendapatkan informasi tersebut, yaitu menggunakan metode kriptografi. Kriptografi adalah suatu ilmu dan seni untuk menjaga kerahasiaan pesan dengan cara menyandikannya ke dalam bentuk yang tidak dapat dimengerti lagi maknanya[2].

Dalam hal teknik pengamanan data, banyak metode kriptografi yang dapat digunakan. Metode – metode kriptografi tersebut mempunyai teknik dan cara tersendiri. Langkah – langkah pengerjaan setiap metode pun berbeda – beda, baik dari segi panjang maupun kerumitan. Salah satu metode kriptografi yang bisa digunakan adalah metode Reverse Cipher. Tetapi jika hanya menggunakan metode Reverse Cipher saja keamanan data teks sangatlah lemah. Maka untuk mencapai tingkat keamanan yang lebih tinggi metode ini dikombinasikan dengan metode RSA yang menggunakan kunci publik serta memiliki keamanan yang cukup tinggi karena sulitnya memfaktorkan bilangan yang besar menjadi faktor-faktor prima pada algoritma RSA tersebut.

Berdasarkan uraian di atas, dilakukan penelitian yang lebih mendalam mengenai kombinasi antara metode kriptografi Reverse Cipher dan RSA yang diterapkan pada keamanan data teks untuk mencapai tingkat keamanan yang tinggi.

2. METODOLOGI

Dengan banyaknya pertukaran informasi yang terjadi, maka banyak pula orang-orang yang menginginkan informasi-informasi tersebut untuk kepentingan pribadi maupun kepentingan kelompok. Hal tersebut dapat terjadi apabila tidak adanya tingkat keamanan yang tinggi dari pertukaran informasi-informasi yang penting. Berdasarkan masalah tersebut, maka sistem yang akan dibangun adalah sistem keamanan data yang berfungsi untuk mengamankan informasi berupa data teks dari para penyadap yang tidak bertanggung jawab. Maka diperlukan pengamanan data teks, dalam hal ini pengamanan data teks dilakukan dengan menggunakan kombinasi metode Reverse Cipher dan RSA sebagai proses enkrip dan dekrip.

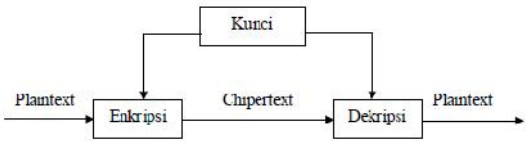
2.1 Kriptografi

Kriptografi dapat didefinisikan sebagai seni maupun ilmu yang menghasilkan pesan yang rahasia. Sebuah pesan asli yang disebut sebagai *plaintext* disandikan menjadi pesan yang tersandi yang disebut sebagai *ciphertext* melalui proses enkripsi dan *ciphertext* dipulihkan menjadi *plaintext* kembali melalui proses dekripsi. Kriptografi memiliki beragam algoritma yang telah banyak digunakan sebagai keamanan untuk informasi. Algoritma kriptografi dikelompokkan ke dalam dua jenis yaitu algoritma kriptografi klasik dan algoritma kriptografi modern. Dalam pengoperasiannya, algoritma kriptografi klasik bekerja menggunakan mode karakter sedangkan algoritma

kriptografi modern bekerja menggunakan mode *bit* [3].

Masalah keamanan merupakan salah satu aspek penting dari sebuah sistem informasi. Salah satu hal penting dalam komunikasi menggunakan komputer dan dalam jaringan komputer untuk menjamin keamanan pesan, data ataupun informasi adalah enkripsi. Enkripsi dapat diartikan sebagai sebuah proses yang dilakukan untuk mengubah pesan asli menjadi pesan yang tersandikan. Informasi yang asli disebut sebagai plaintext, dan bentuk yang sudah dienkripsi disebut sebagai ciphertext. Pesan ciphertext berisi seluruh informasi dari pesan plaintext, tetapi tidak dalam format yang dapat dibaca oleh manusia ataupun komputer tanpa menggunakan mekanisme yang tepat untuk melakukan dekripsi [4].

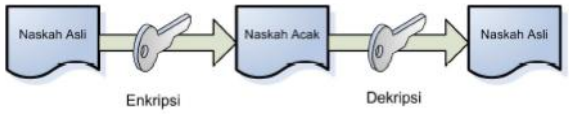
Kriptografi memiliki dua konsep utama, yaitu enkripsi (*encryption*) dan dekripsi (*decryption*). Enkripsi adalah proses penyandian plainteks menjadi cipherteks, sedangkan dekripsi adalah proses mengembalikan cipherteks menjadi plainteks semula. Enkripsi dan dekripsi membutuhkan kunci sebagai parameter yang digunakan untuk transformasi [4].



Gambar 1. Skema Enkripsi dan Dekripsi

Kriptografi *Type Symmentric Key*

Algoritma kriptografi klasik memiliki ciri di antaranya berbasis karakter dan menggunakan kunci simetri. Dalam kriptografi klasik, teknik enkripsi yang digunakan adalah enkripsi simetris dimana kunci dekripsi sama dengan kunci enkripsi seperti dapat dilihat pada Gambar 2 [5].



Gambar 2. Proses Enkripsi dan Dekripsi

Salah satu algoritma klasik adalah *Caesar Cipher*. Dalam kriptografi klasik, secara umum dapat dikelompokkan dalam dua model yaitu menggunakan tekik substitusi dan transposisi. Teknik substitusi dilakukan dengan mengganti salah satu karakter yang ada dalam sebuah teks menggunakan karakter yang lain. Teknik yang termasuk dalam kategori substitusi adalah kriptografi *Caesar* [5].

Algoritma kriptografi modern merupakan suatu perbaikan yang mengacu pada kriptografi klasik. Algoritma ini menggunakan pengolahan simbol biner yang dibentuk dari kode ASCII (*American Standard Code for Information Interchange*) karena berjalan mengikuti operasi komputer digital, sehingga membutuhkan pengetahuan dasar matematika untuk menguasainya. Algoritma ini memiliki tingkat kesulitan yang kompleks yang menyebabkan kriptanalis sangat sulit memecahkan *ciphertext* tanpa mengetahui kuncinya. Adapun jenis kunci dalam kriptografi modern terdiri dari 3 yaitu: simetri, asimetri, dan hibrida. Pada kriptografi modern terdapat berbagai macam algoritma yang dimaksudkan untuk mengamankan informasi yang dikirim melalui jaringan komputer. Contoh kriptografi modern yaitu MD5, RC4, AES dan lain-lain [6].

Berikut akan dijelaskan jenis – jenis kunci dalam kriptografi modern adalah sebagai berikut [6]:

1. Algoritma Simetris, adalah algoritma yang menggunakan kunci yang sama untuk enkripsi dan dekripsinya. Algoritma kriprografi simetris sering disebut algoritma kunci rahasia, algoritma kunci tunggal, atau algoritma satu kunci, dan mengharuskan pengirim dan penerima menyetujui suatu kunci tertentu. Kelebihan dari algoritma kriprografi simetris adalah

waktu proses untuk enkripsi dan dekripsi relatif cepat. Hal ini disebabkan efesiensi yang terjadi pada pembangkit kunci. Karena prosesnya relatif cepat maka algoritma ini tepat untuk digunakan pada sistem komunikasi digital secara *real time* seperti GSM. Aplikasi dari algoritma simetris digunakan oleh beberapa algoritma seperti Data Encryption Standard (DES), Advance Encryption Standard (AES), International Data Encryption Algoritma (IDEA), A5, dan lain – lain.

2. Algoritma Asimetris, adalah pasangan kunci kriptografi yang salah satunya digunakan untuk proses enkripsi dan satu lagi lagi deskripsi. Semua orang yang mendapatkan kunci publik dapat menggunakannya untuk mengenkripsi suatu pesan, sedangkan hanya satu orang saja yang memiliki rahasia itu, yang dalam hal ini kunci rahasia, untuk melakukan pembongkaran terhadap kode yang dikirim untuknya. Contoh algoritma terkenal yang menggunakan kunci asimetris adalah RSA (merupakan singkatan dari nama penemunya, yakni Rivest, Shamir dan Adleman).
3. Algoritma Hibrida, adalah algoritma yang memanfaatkan dua tingkatan kunci, yaitu kunci rahasia (simetri) – yang disebut juga *session key* (kunci sesi) untuk enkripsi data dan pasangan kunci rahasia – kunci publik untuk pemberian tanda tangan digital serta melindungi kunci simetri.

2.2 Android

Android adalah sistem operasi (*Operating System*) yang umumnya digunakan pada perangkat dengan navigasi *full touch screen* yang biasa dimiliki oleh *smartphone* dan komputer tablet. Android sudah diambil alih oleh perusahaan *Google Inc.* yang telah membelinya pada tahun 2005 dari *Android Inc. Google* menyediakan *software/tools* yang dikembangkan khusus untuk dijadikan alat pengembang aplikasi android [7].

Pada tanggal 23 september 2008, sistem operasi *Android* versi 1.0 resmi diluncurkan. Sekitar sebulan berikutnya, pada tanggal 22 Oktober 2008, *smartphone* pertama yang menjalankan Android 1.0 itu, yaitu HTC Dream, diluncurkan ke pasar. Pada tanggal 9 februari, Android versi 1.1 diluncurkan untuk memperbaiki bug dari versi sebelumnya dan menambah fitur yang tersedia. Setelah versi 1.1, rilis *Android* berikutnya menggunakan nama makanan manis dengan urutan *alfabetis*, dimulai dengan 1.5 *Cupcake* yang diluncurkan pada tanggal 30 April 2009. Rilis-rilis *Android* selanjutnya, yaitu *Donut*, *clair*, *Froyo*, dan *Gingerbread* semua dibuat untuk smartphone. Namun, *Apple* meluncurkan *iPad* pada tahun 2010 dan meningkatkan ketertarikan masyarakat luas kepada *computer tablet*. Beberapa pengembang *Android* mencoba mengembangkan *tablet Android* untuk menyaingi *iPad*, seperti *Samsung Galaxy Tab* yang menggunakan Gingerbread yang dikustomisasi. *Google* dan OHA pun bergerak dengan melakukan pengembangan *Android* versi baru yang lebih optimal untuk tablet. Pada tanggal 22 februari 2011, android Honeycomb diluncurkan ke pasar dan pada tanggal 24 februari 2011, tablet pertama yang menggunakan *honeycomb*, yaitu *Motorola Xoom*, diluncurkan ke pasar. Pada tanggal 19 Oktober 2011, *Android* meluncurkan *Ice Cream Sandwich* versi ini dapat bekerja secara optimal baik di *smartphone* maupun di *tablet*. Rilis *android* berikutnya, yaitu *Jelly Bean*, bertujuan untuk semakin meningkatkan apa yang sudah tersedia di *Ice Cream Sandwich*, dengan memperbaiki *bug-bug* dn menambahkan fitur-fitur. Pada tanggal 3 september 2013, diumumkan versi *Android* selanjutnya adalah *Android* 4.4 Kit Kat. Android sudah mendapatkan izin dari *Nastle* dan *Hershey* selaku pemilik

dagang dagang Kit Kat. Sebelum pengumuman ini, banyak yang berspekulasi bahwa versi *Android* berikutnya akan diberi nomor 5.0 dengan nama *Key Lime Pie* brikut adalah tabel untuk semua sistem operasi *Android* yang sudah diluncurkan sampai sekarang. Saat buku ini ditulis, sistem operasi *Android* yang terbaru adalah *Android 6.0 Marshmallow* [7].

2.3 Proses Enkripsi dan Dekripsi Metode Reverse Cipher

Contoh kriptografi klasik yang menggunakan transposisi yaitu mengganti satu huruf dengan huruf lain. Reverse Chiper adalah salah satu contoh yang paling sederhana dari kriptografi transposisi yaitu mengubah suatu kalimat dengan menuliskan setiap kata secara terbalik[8]. Adapun contoh kriptografi Reverse Chiper adalah :

Plaintext:	IBU AKAN DATANG BESOK PAGI
Ciphertext:	UBI NAKA GNATAD KOSEB IGAP

Gambar 3. Reverse Chiper

2.4 Proses Enkripsi dan Dekripsi Metode RSA

Dalam algoritma kriptografi, proses enkripsi diterapkan untuk mengamankan data. Dengan enkripsi data tidak dapat terbaca karena teks asli atau *plaintext* telah diubah ke teks yang tak terbaca atau disebut *ciphertext*. Ada banyak algoritma kriptografi yang dapat digunakan. Berdasarkan sifat kuncinya algoritma kriptografi dibagi menjadi dua yaitu simetris yang hanya memakai satu kunci rahasia dan asimetris (*public key algorithm*) yang memakai sepasang kunci publik dan kunci rahasia [9].

Pada tahun 1977, Ronald L. Rivest, Adi Shamir, dan Leonard M. Adleman merumuskan algoritma praktis yang mengimplementasikan sistem kriptografi kunci publik yang disebut dengan sistem kriptografi RSA. Sepasang kunci yang dipakai pada kedua proses ini adalah kunci publik (*e,n*) sebagai kunci enkripsi dan kunci privat *d* sebagai sebagai kunci dekripsi dimana *e, d* dan *n* adalah bilangan bulat positif. Algoritma RSA adalah sebuah *block cipher algorithm* (algoritma yang bekerja per blok data) yang mengelompokkan *plaintext* menjadi blok-blok terlebih dahulu sebelum dilakukan enkripsi hingga menjadi ciphertext [10].

RSA adalah metode yang menggunakan perhitungan matematika yang rumit dan disertai dengan kunci pengaman awal (dengan *private key* maupun dengan *public key*) sehingga amat sulit untuk ditembus oleh *hacker*. Adapun prinsip pengamanan metode ini adalah bagaimana sistem dapat mengamankan proses penyimpanan dan pengiriman dokumen. Mula-mula dokumen dalam bentuk teks dienkripsi dengan metode RSA. Sehingga dokumen tidak dapat dibaca oleh siapapun, karena teks telah berubah menjadi susunan huruf yang teracak. Dokumen yang susunan hurufnya telah teracak tersebut jika ingin dibaca oleh pemilik dokumen, maka dokumen tersebut harus dibuka dengan dekripsi RSA kembali [10].

Secara garis besar, proses kriptografi pada algoritma RSA terdiri dari 3 tahapan yaitu :

a. Pembangkitan Kunci, untuk membangkitkan kedua kunci, dipilih dua buah bilangan prima yang sangat besar *p* dan *q*. Untuk mendapatkan keamanan yang maksimum, dipilih dua bilangan p dan q yang besar. Kemudian dihitung :

$$n = p \cdot q \quad (1)$$

Kemudian dihitung :

$$\varphi = (p-1)(q-1) \quad (2)$$

Lalu dipilih kunci enkripsi *e* secara acak, sedemikian sehingga *e* dan *(p-1)(q-1)* relatif prima. Artinya *e* dan *φ* tidak memiliki faktor persekutuan

bersama. Kemudian dengan algoritma Euclidean yang diperluas, dihitung kunci dekripsi *d*, sehingga :

$$ed = 1 \bmod (p-1)(q-1) \quad (3)$$

atau

$$ed - 1 = k\,(p-1)(q-1) \quad (4)$$

Dimana *k* merupakan konstanta *integer*. Perhatikan bahwa *d* dan *n* juga relatif prima. Bilangan *e* dan *n* merupakan kunci publik, sedangkan *d* kunci privat. Dua bilangan prima *p* dan *q* tidak diperlukan lagi. Namun *p* dan *q* kadang diperlukan untuk mempercepat perhitungan dekripsi.

b. Proses Enkripsi, untuk mengenkripsi pesan *m*, terlebih dahulu pesan dibagi ke dalam blok-blok numerik yang lebih kecil dari *n* (dengan data biner, dipilih pangkat terbesar dari 2 yang kurang dari *n*). Jadi jika *p* dan *q* bilangan prima 100 digit, maka *n* akan memiliki sekitar 200 buah digit dari setiap blok pesan *m*, seharusnya kurang dari 200 digit panjangnya. Pesan yang terenkripsi *(c)*, akan tersusun dari blok-blok *(ci)* yang hampir sama panjangnya. Rumus enkripsinya adalah:

$$c_i = m_i^{\,e} \,.\, mod\ n \quad (5)$$

c. Proses Dekripsi, setelah menerima pesan yang sudah terenkripsi maka penerima pesan akan melakukan proses dekripsi pesan dengan cara :

$$c_i = m_i^{\,d} \,.mod\ n \quad (6)$$

Besaran-besaran yang digunakan pada algoritma RSA :

1. *p* dan *q* bilangan prima (rahasia)
2. $r = p \,.\, q$ (tidak rahasia)
3. $¤\,(r) = (p-1)(q-1)$ (rahasia)
4. *PK* (kunci enkripsi) (tidak rahasia)
5. *SK* (kunci dekripsi) (rahasia)
6. *X* (plainteks) (rahasia)
7. *Y* (cipherteks) (tidak rahasia)

3. HASIL DAN PEMBAHASAN

Dalam penelitian ini, penulis mencoba untuk membuat suatu aplikasi pengamanan data teks menggunakan kombinasi metode Reverse Cipher dan RSA. Dengan memanfaatkan kombinasi metode ini, diharapkan dapat dikembangkan suatu aplikasi pengamanan data teks yang memungkinkan pengguna untuk mengenkrip data teks dengan metode Reverse Cipher, lalu dienkrip lagi menggunakan metode RSA dan dapat melakukan dekripsi terhadap data teks terenkripsi tersebut. Aplikasi pengamanan data teks ini akan dibangun berbasis *mobile* pada *platform* Android.

Metode Reverse Cipher hanya mengubah atau membalikkan posisi huruf atau plainteks. Lalu dilakukan proses enkrip kembali dengan metode RSA. Untuk penerapan RSA dapat dilihat dari proses berikut ini.

Misalkan $p = 47$ dan $q = 71$ (keduanya prima).

a. Selanjutnya, hitung nilai :

$$r = p \cdot q = 3337$$

dan

$$¤(r) = (p-1)(q-1) = 3220$$

b. Pilih kunci publik *SK* = 79, karena 79 relatif prima dengan 3220. *PK* dan *r* dapat dipublikasikan ke umum.

c. Selanjutnya akan dihitung kunci dekripsi *SK* seperti yang dituliskan pada langkah instruksi 5 dengan menggunakan persamaan :

$$SK = \frac{1 + (m \times 3220)}{79} \quad (7)$$

d. Dengan mencoba nilai-nilai m = 1, 2, 3, …, diperoleh nilai SK yang bulat adalah 1019. Ini adalah kunci dekripsi yang harus dirahasiakan.

Enkripsi :

- Plainteks disusun menjadi blok-blok x_1, x_2, …, sedemikian sehingga setiap blok merepresentasikan nilai di dalam rentang 0 sampai r – 1.
- Setiap blok x_i dienkripsi menjadi blok y_i dengan rumus :

$$y_i = x_i^{PK} \bmod r \quad (8)$$

Dekripsi :

- Setiap blok cipherteks y_i didekripsi kembali menjadi blok x_i dengan rumus:

$$x_i = y_i^{SK} \bmod r \quad (9)$$

Misalkan plainteks yang akan dienkripsikan adalah X = HARI INI, atau dalam sistem desimal (pengkodean ASCII) adalah 7265827332737873.

a. Pecah X menjadi blok yang lebih kecil, misalnya X dipecah menjadi enam blok yang berukuran 3 digit :
   x_1 = 726 x_4 = 273
   x_2 = 582 x_5 = 787
   x_3 = 733 x_6 = 003
   Nilai-nilai x_i ini masih terletak di dalam rentang 0 sampai 3337 – 1 (agar transformasi menjadi satu-ke-satu).
b. Blok-blok plainteks dienkripsikan sebagai berikut:
   $726^{79} \bmod 3337 = 215 = y_1$
   $582^{79} \bmod 3337 = 776 = y_2$
   $733^{79} \bmod 3337 = 1743 = y_3$
   $273^{79} \bmod 3337 = 933 = y_4$
   $787^{79} \bmod 3337 = 1731 = y_5$
   $003^{79} \bmod 3337 = 158 = y_6$
   Jadi, cipherteks yang dihasilkan adalah Y = 215 776 1743 933 1731 158
c. Dekripsi dilakukan dengan menggunakan kunci rahasia :
   SK = 1019
d. Blok-blok cipherteks didekripsikan sebagai berikut:
   $215^{1019} \bmod 3337 = 726 = x_1$
   $776^{1019} \bmod 3337 = 582 = x_2$
   $1743^{1019} \bmod 3337 = 733 = x_3$
   …
e. Blok plainteks yang lain dikembalikan dengan cara yang serupa. Akhirnya kita memperoleh kembali plainteks semula
   P = 7265827332737873
   yang dalam karakter ASCII adalah :
   P = HARI INI.

Aplikasi ini dibuat dengan bahasa pemograman Ec*lipse for Java Developers* dapat dijalankan dengan komputer yang berbasiskan *windows* ataupun *handphone android*. Ada beberapa cara untuk menjalankannya aplikasi ini yang akan dijelaskan sebagai berikut :

1. Menjalankan Melalui Komputer, untuk menjalankan aplikasi ini melalui komputer, dilakukan dengan cara :
   a. Instal *Software Eclipse Galileo* dan perangkat lainnya untuk menjalankan program *Java Android Mobile*.
   b. Kemudian jalankan program enkripsi dan dekripsi ini dengan mengklik kanan folder *project* yang telah dibuat sebelumnya, lalu klik *Run as Android Application*.
   c. Maka akan ditampilkan hasil dari *load project* tersebut.
2. Menjalankan Melalui Android, untuk menjalankan aplikasi ini melalui android dilakukan dengan cara :
   a. *Copy*-kan *file .apk* dari *folder bin* hasil *run* dari *project* yang di komputer, bisa menggunakan media *bluetooth* ataupun media kabel data.
   b. Lakukan penginstalan sebelum menjalankan program, konfigurasi penginstalan akan menyesuaikan dengan android yang digunakan.

Program aplikasi enkripsi dan dekripsi ini memiliki kelebihan dan kekurangan pada implementasinya di

lingkungan *user*. Kelebihan dan kekurangan pada aplikasi dijelaskan di bawah ini.

Kelebihan pada perancangan aplikasi enkripsi dan dekripsi ini di antaranya yaitu:

1. Tampilan yang sederhana sehingga udah untuk menggunakan enkripsi dan dekripsi ini.
2. Membutuhkan spesifikasi *hardware* dan *software* yang rendah.
3. Mudah instalasi *file* ke dalam *handphone android*.

Kekurangan atau kelemahan pada perancangan aplikasi enkripsi dan dekripsi ini di antaranya yaitu :

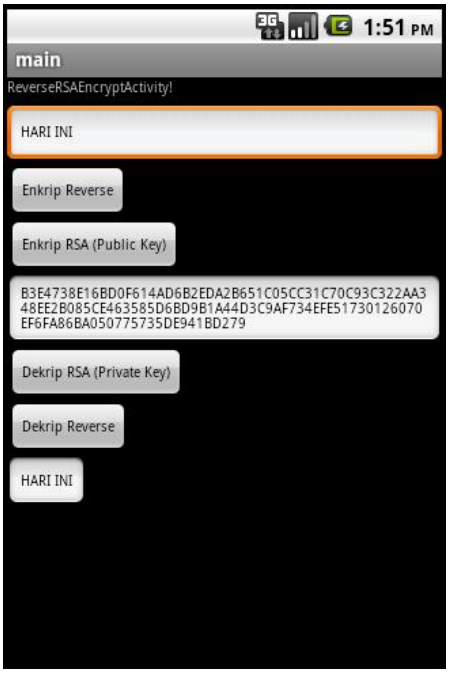
1. Program aplikasi ini hanya bisa dijalankan pada sistem operasi *android*.
2. Hanya dapat dijalankan secara *stand alone*, tidak berbentuk aplikasi sistem jaringan.

Pada aplikasi ini penulis melakukan pengujian yang dilakukan adalah pengujian fungsionalitas dari sistem, apakah sistem berfungsi dengan hasil yang diinginkan atau tidak. Pada aplikasi enkripsi dan dekripsi ini, pengujian merujuk pada fungsi-fungsi yang dimiliki sistem, kemudian membandingkan hasil keluaran dengan hasil yang diharapkan. Bila hasil yang diharapkan sesuai dengan hasil pengujian, berarti perangkat lunak sesuai dengan desain yang telah ditentukan sebelumnya. Bila belum sesuai maka perlu dilakukan pengecekan lebih lanjut dan perbaikan. Adapun uji coba sistem yang telah dilakukan dapat dilihat pada tabel IV.1 sebagai berikut :

Tabel 1. Uji Coba Sistem

No	Nama Proses	Prosedur Pengujian	Hal Yang Diharapkan	Hasi Pengujian
1.	Tombol Enkrip dan Dekrip	Menekan tombol Enkrip dan Dekrip	Menampilkan menu Enkripsi dan Dekripsi	Sukses
3.	Tombol Keterangan	Menekan tombol Keterangan	Menampilkan menu Keterangan	Sukses
4.	Tombol Keluar	Menekan tombol Keluar	Keluar dari aplikasi	Sukses

Berikut tampilan dari aplikasi yang dirancang :



Gambar 1. Tampilan Aplikasi

4. KESIMPULAN

Setelah dilakukan analisa dan implementasi pembahasan maka penelitian ini menyimpulkan beberapan yaitu :

1. Teks diamankan dengan menggunakan kombinasi metode kriptografi Reverse Cipher dan RSA.
2. Proses enkripsi diawali dengan mengenkrip plainteks dengan metode Reverse Cipher, lalu dienkrip lagi menggunakan metode RSA yang menghasilkan cipherteks.
3. Proses dekripsi diawali dengan proses dekrip cipherteks menggunakan metode RSA, lalu didekrip kembali menggunakan Reverse Cipher, dan dihasilkan plainteks.
4. Aplikasi ini hanya mengenkrip dan mendekrip teks.

5. SARAN

Adapun saran dari penelitian ini adalah sebagai berikut :

1. Diharapkan pengembangan aplikasi menggunakan kombinasi metode – metode yang lain.

2. Diharapkan pengembangan aplikasi pada *platform* yang berbeda.

REFERENSI

[1]. Siti Zulfah. 2015. *Pengaruh Perkembangan Teknologi Informasi Lingkungan (Studi Kasus Kelurahan Siti Rejo I Medan)*. Medan

[2]. Nandar Pabokory Fresly, dkk. 2015. *Implementasi Kriptografi Pengamanan Data pada Pesan Teks, Isi File Dokumen, dan File Dokumen Menggunakan Algoritma Advanced Encryption Standard*. Samarinda

[3]. Eka Putra Muhammad. 2017. *Perancangan Aplikasi Pengamanan Informasi Teks Dengan Menggunakan Algoritma Kriptografi Alpha-Qwerty Reverse*. Palembang

[4]. Riski Alvianto Andi dan Darmaji. 2018. *Pengaman Pengiriman Pesan Via SMS dengan Algoritma RSA Berbasis Android*. Surabaya

[5]. Miftakul Amin Muhammad. 2016. *Implementasi Kriptografi Klasik pada Komunikasi Berbasis Teks*. Palembang

[6]. Sumandri. 2017. *Studi Model Algoritma Kriptografi Klasik dan Modern*. Yogyakarta

[7]. Aditya Permana Angga. 2018. *Penerapan Kriptografi Pada Teks Pesan dengan Menggunakan Metode Vigenere Cipher Berbasis Android*. 2018

[8]. Uzzin Nadhori Isbat, dkk. 2015. *Pembuatan Perangkat Lunak Media Pembelajaran Kriptografi Klasik*. Surabaya

[9]. Sulaiman Rahmat dan Vebu Marina. 2018. *Peningkatan Keamanan Pesan Berbasis Android Menggunakan Algoritma Kriptografi RSA*. Pangkal Pinang

[10]. Riad Sahara, dkk. 2017. *Implementasi Keamanan SMS dengan Algoritma RSA pada Smartphone Android*. Jakarta Barat

[11]. Fauzi, Achmad, dkk. 2017. *Analisa Penerapan Algoritma Rivest Shamir Adlemen (RSA) Pada Kerahasiaan Data Teks*. *SNIKOM 2017*