APLIKASI SECURE MESSAGE MENGGUNAKAN ALGORITMA CR6 BERBASIS ANDROID

Rionald Ricardo Mangundap[1], Wiwin Agus Kristiana[2]
[1]MahasiswaSistemKomputerUniversitasNarotama, Rio_Ricardo13@yahoo.com
[2]DosenFakultasIlmuKomputerUniversitasNarotama, Wiwin.Agus@narotama.ac.id

ABSTRAK

Perkembangan teknologi telekomunikasi yang pesat telah memberikan manfaat besar. Dengan adanya teknologi telekomunikasi, jarak dan waktu bukan lagi menjadi sebuah kendala dalam berkomunikasi. Salah satu hasil teknologi telekomunikasi yang sangat terkenal untuk bertukar pesan teks dengan pengguna lain adalah Short Message Service (SMS).
Media SMS tidak langsung sampai pada nomor tujuan namun melalui jaringan SMS Center. Pada SMS Center keamanan pesan rentan untuk dibaca oleh orang yang tidak bertanggung jawab. Proyek yang dibangun penulis mempunyai tujuan untuk meningkatkan keamanan pesan dengan menggunakan enkripsi, perangkat yang dibangun menggunakan metode Algoritma RC6 Berbasis Android. Aplikasi ini diharapkan dapat melindungi isi pesan singkat yang dikirim oleh pengirim ke nomor tujuan dan melindungi pesan agar tidak dibaca oleh orang yang tidak bertanggung jawab.

Kata kunci: Short Message Service, Android, Algoritma RC6.

I.PENDAHULUAN

1.1 Latar Belakang Masalah

Perkembangan teknologi telekomunikasi yang ada pada saat ini mampu menciptakan berbagai macam perangkat keras yang dapat digunakan untuk mengirim atau menerima informasi dengan cepat dan mudah. Penggunaan *handphone* sebagai *device* akses informasi telah berkembang pesat pada era ini. Terlebih lagi, banyak aplikasi *mobile* yang diciptakan, membuat informasi-informasi yang dibutuhkan mudah untuk didapat. Perangkat keras yang cukup banyak digunakan pada saat ini adalah *smartphone* android. Banyak merk dan jenis *smartphone* android beredar di pasaran.

Layanan pesan singkat menggunakan aplikasi SMS pada ponsel masih banyak digunakan, namun bukan jalur yang aman untuk pertukaran informasi. Pesan yang dikirim menggunakan aplikasi SMS bawaan ponsel masih berupa teks terbuka yang belum terproteksi, selain itu proses pengiriman SMS tidak sampai ke penerima secara langsung, akan tetapi pengiriman SMS harus melewati *Short Message Service Center* (SMSC) yang berfungsi mencatat komunikasi yang terjadi antara pengirim dan penerima.

Dengan tersimpanya SMS pada SMSC, maka seorang operator dapat memperoleh informasi atau membaca SMS di dalam SMSC tersebut, hal ini dapat dibuktikan dari beberapa kasus yang ditangani kepolisian, kejaksaan atau KPK, dimana pihak penyelidik tersebut meminta transkip SMS ke operator untuk dijadikan bahan penyelidikan di persidangan.

Dengan demikian dibutuhkan suatu metode dan aplikasi yang dapat mempertimbangkan solusi *encrypted end to end* dengan melakukan enkripsi terhadap pesan SMS. Enkripsi adalah proses mengubah suatu pesan asli yang disebut *plaintext* menjadi sebuah sandi atau kode yang tidak terbaca yang disebut *chipertext* dan tidak dapat dimengerti, untuk mengembalikan pesan ke bentuk asli seperti semula diperlukan proses yang disebut dekripsi. Enkripsi dimaksudkan untuk melindungi dan menyamarkan informasi agar tidak terlihat oleh pihak atau orang yang bukan seharusnya.

1.2 Perumusan Masalah

Rumusan masalah yang akan penulis gunakan sebagai acuan dalam penelitian ini adalah Apakah algoritma RC6 merupakan algoritma yang baik dalam menjaga keamanan sms dibandingkan RC5 dan versi sebelumnya.

1.3 Batasan Masalah

Batasan yang digunakan dalam penelitian ini adalah sebagai berikut:

1. Implementasi program pada *smartphone* android.
2. Aplikasi akan dikembangkan untuk android dengan versi minimal android 2.2 (*froyo*) dengan level API 8.
3. Aplikasi dibangun dengan target android versi 4.1 (*jelly bean*) dengan level API 18.
4. Proses enkripsi dan dekripsi pesan hanya digunakan untuk data *text* berupa huruf, angka dan *symbol*.

1.4 Tujuan Penelitian

1. Mengimplementasikan enkripsi data pada pengiriman pesan teks pada layanan SMS dengan metode RC6.
2. Membangun aplikasi yang mampu bekerja pada perangkat android, sebagai aplikasi pihak ketiga yang mampu melakukan enkripsi dan dekripsi data *text* pada layanan SMS sebelum dan sesudah dikirimkan.

1.5 Manfaat Penelitian

1. Manfaat akademis

Penelitian ini berguna untuk menambah pengetahuan serta memberikan pemahaman kepada penulis dan pembaca yang berkepentingan tentang keamanan data dan informasi.

2. Manfaat dalam implementasi atau praktik

Diharapkan pengguna dapat menggunakan hasil penelitian ini sebagai sarana untuk memberi keamanan terhadap informasi yang dikirim melalui jalur pesan singkat pada telepon selular berbasis Android.

II. LANDASAN TEORI

2.1 Android

Android adalah sebuah sistem operasi untuk perangkat *mobile* berbasis linux yang mencakup sistem operasi, *middleware* dan aplikasi. Android menyediakan *platform* terbuka bagi para pengembang untuk menciptakan aplikasi mereka. (Nazruddin Safaat H 2012. Pemrograman Aplikasi Mobile Smartphone dan Tablet PC Berbasis Android, penerbit INFORMATIKA, Bandung.).

2.2 Fitur Perangkat Lunak Android

Dalam perangkat lunak android yang paling menonjol adalah tidak diberikannya akses *root* pada perangkat android untuk mengakses partisi yang ada pada android seperti pada partisi /sistem. Dikarena untuk mencegah adanya perubahan pada partisi yang hanya bersifat *read-only* dan kemudian juga tidak diinginkanya kesalahan pengembangan pada android dan penyebaran virus dengan membuka langsung akses root tersebut, akses tersebut dapat di dapatkan dengan metoda tertentu pada perangkat android.

2.3 Algoritma Kriptografi RC6

Perkembangan algoritma kriptografi dapat kita bagi menjadi dua, yaitu:

1. Kriptografi Klasik Pada algoritma klasik, diterapkan teknik enkripsi konvensional (simetris). Algoritma ini merupakan algoritma kriptografi yang biasa digunakan orang sejak berabad-abad yang lalu.
2. Kriptografi Modern Kriptografi modern lebih menitikberatkan pada kerahasiaan kunci yang digunakan pada algoritma tersebut (oleh pemakainya) sehingga algoritma tersebut dapat saja disebarkan tanpa takut kehilangan kerahasiaan bagi para pemakainya.

Secara umum berdasarkan kesamaan kuncinya, algoritma sandi dibedakan menjadi : 1. Algoritma Simetrik (*symmetric algorithm*) adalah suatu algoritma yang menggunakan kunci enkripsi sama dengan kunci dekripsi sehingga algoritma ini disebut juga sebagai *single-key algorithm*. Beberapa algoritma kriptografi simetrik antara lain DES, Blowfish, IDEA, RC4, RC5, RC6, AES atau Rijndael, Serpent dan Twofish. 2. Algoritma Asimetrik (*asymmetric algorithm*) adalah suatu algoritma yang menggunakan kunci enkripsi tidak sama dengan kunci dekripsi. Algoritma ini menggunakan dua kunci yakni kunci publik (*public key*) dan kunci privat (*private key*). Beberapa algoritma kunci publik antara lain adalah RSA, Rabin dan ElGamal. (Kurniawan, Yusuf, Ir. 2004. Kriptografi Keamanan Internet dan Jaringan Komunikasi, penerbit INFORMATIKA, Bandung.)

2.4 Algoritma Dekripsi Kriptografi RC6

Proses dekripsi *ciphertext* pada algoritma RC6 merupakan pembalikan dari proses enkripsi. Pada proses *whitening*, bila proses enkripsi menggunakan operasi penjumlahan, maka pada proses dekripsi menggunakan operasi pengurangan. Sub kunci yang digunakan pada proses *whitening* setelah iterasi terakhir diterapkan sebelum iteasi pertama, begitu juga sebaliknya sub kunci yang diterapkan pada proses *whitening* sebelum iterasi pertama digunakan pada *whitening* setelah iterasi terkahir. Akibatnya, untuk melakukan dekripsi, hal yang harus dilakukan semata-mata hanyalah menerapkan algoritma yang sama dengan enkripsi, dengan tiap iterasi menggunakan sub kunci yang sama dengan yang digunakan pada saat enkripsi, hanya saja urutan sub kunci yang digunakan terbalik.

2.5 Keamanan Algoritma RC6

Selain algoritma RC6 ada beberapa metode enkripsi yang bisa digunakan, salah satunya metode *blowfish*, dikutip pada jurnal Reza Brianca Widodo, Makalah IF3058 Kriptografi Studi dan Perbandingan Algoritma RC6 dan Blowfish. Kedua jenis algoritma blok cipher memiliki tingkat keamanan yang tinggi jika ketentuan yang berlaku dalam mekanisme enkripsi di dalamnya terpenuhi, Algoritma RC6 memiliki keamanan yang baik walaupun memang Blowfish memiliki keunggulan dalam hal efisiensi waktu enkripsi dan dekripsi.

Dalam algoritma enkripsi, panjang kunci yang biasanya dalam ukuran bit, juga menentukan kekuatan dari enkripsi. Kunci yang lebih panjang biasanya lebih aman daripada kunci yang pendek. Jadi enkripsi dengan menggunakan kunci 128-bit lebih sukar dipecahkan dengan algoritma enkripsi yang sama tetapi memiliki kunci 56- bit. Semakin panjang sebuah kunci, semakin besar *keyspace* yang harus dijalani untuk mencari kunci dengan cara *brute force attack* atau coba-coba karena *keyspace* yang harus dilihat merupakan pangkat bilangan dari 2. Jadi kunci 128-bit memiliki *keyspace* 2128, sedangkan kunci 56-bit memiliki *keyspace* 256. Artinya semakin lama kunci baru bisa ditemukan.

Pada intinya, keamanan suatu pesan tidak tergantung pada sulitnya algoritma tetapi pada kunci yang digunakan. Pada RC6 dengan adanya fungsi $f(x)=x(2x+1)$ yang diikuti pergeseran lima bit ke kiri dapat memberi tingkat keamanan data yang tinggi. Adanya *avalanche effect* juga memberikan cukup kesulitan kepada kriptanalis untuk melakukan serangan.

Kekurangan umum dari algoritma yang berbentuk simetris atau kunci pribadi adalah pada kunci itu sendiri. Kelemahan ini timbul jika terdapat banyak orang yang ingin saling berkomunikasi, karena setiap pasangan maupun *file* enkripsi mempunyai *key* berbeda yang harus disepakati sehingga *key* tiap orang maupun *file* harus menghafal banyak kunci dan menggunakannya dengan tepat.

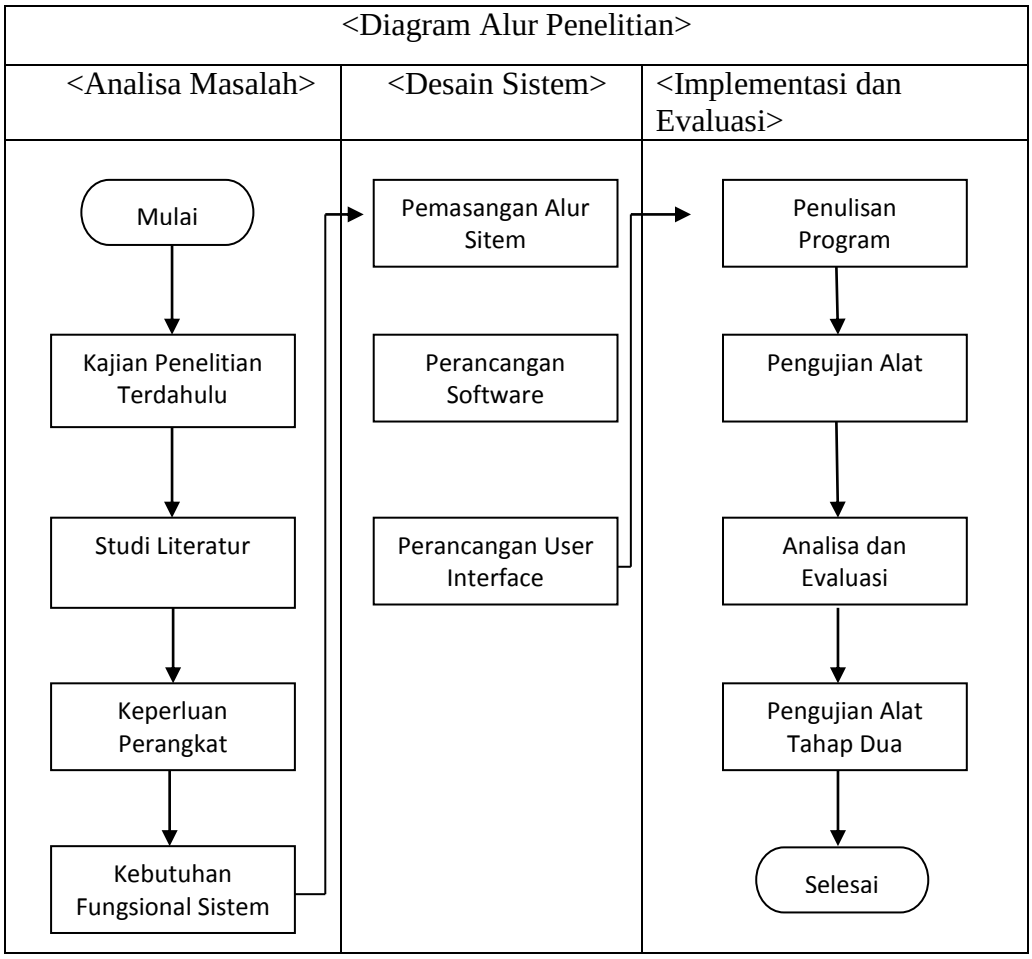
III.METODOLOGI

3.1 Metode Yang Digunakan

Pada penelitian ini, penulis menggunakan metode penelitian dan pengembangan dalam membangun sebuah aplikasi Pesan (SMS), penulis mendapatkan suatu permasalahan yang muncul di lapangan, yaitu tingkat keamanan yang kurang baik dalam pengiriman pesan antar pengguna aplikasi pesan dimana isi pesan tersebut dapat di baca oleh pihak yang tidak bertanggung jawab. Dari permasalahan tersebut, kemudian digunakan sebagai bahan untuk membangun aplikasi *Security Message* agar dapat meningkatkan keamanan dalam pengiriman pesan.

3.2 Diagram Alur Penelitian

Metode penelitian yang dipilih, digambarkan pada diagram alur (*flowchart*) gambar 3.1 sebagai sebagai alur berfikir peneliti untuk pemecahan masalah.

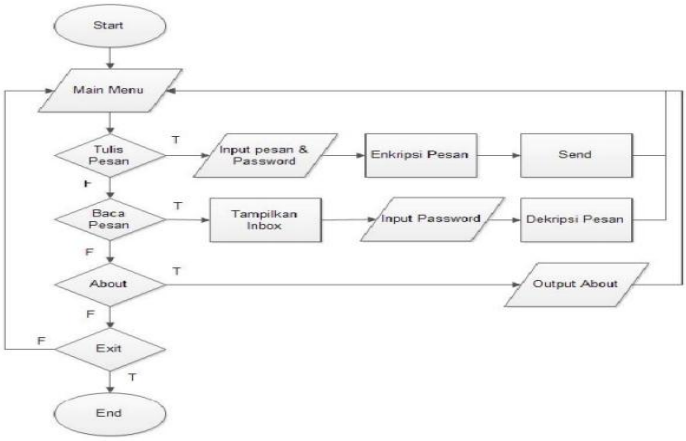


Gambar 3.1 Diagram Alur Penelitian

Diagram alur tersebut, memudahkan penulis dalam mengerjakan pembuatan aplikasi *security message*.

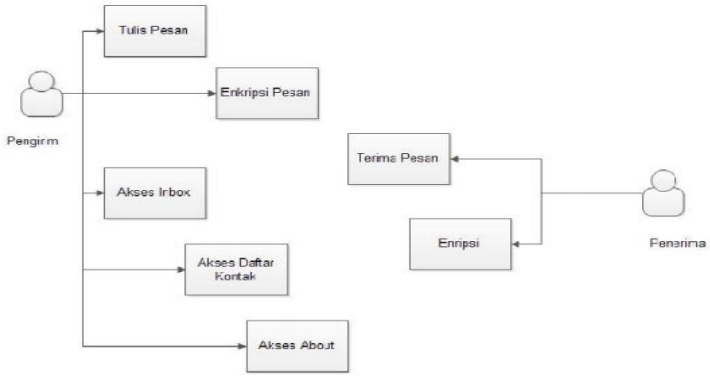
3.3 Flowchart Diagram

Pada aplikasi ini *user* akan menjalankan sebuah *user interface* yang menampilkan menu aplikasi dalam beberapa tahapan untuk membuat sebuah enkripsi SMS. Pada halaman awal atau main menu dari program *user* hanya akan melihat empat buah tombol yaitu Tulis Pesan, Baca Pesan dan *About*. Pada tombol ”Tulis Pesan” berfungsi untuk menampilkan layer baru yang berguna untuk menulis *plaintext*, applikasi akan menjalankan fungsi untuk menghasilkan sebuah *chipertext* yang akan dikirim ke penerima SMS, kemudian pada tombol ”Baca Pesan” akan mengaktifkan fungsi untuk melihat kotak masuk dari pesan yang terdapat pada inbox *smartphone user*. Pada tombol ”*About*” akan menampilkan versi dari aplikasi dan bio data dari pembuat aplikasi.



Gambar 3.2 Gambar Flowchart Program

Pada aplikasi memiliki alur dalam penggunaanya, adapun *usecase* dari aplikasi adalah sebagai berikut :



Gambar 3.3 *Usecase* aplikasi enkripsi SMS

Aktor	Deskripsi
Pengirim	Tulis Pesan - Pengirim menulis pesan pada aplikasi untuk dikirim
Pengirim	Enkripsi Pesan - Pengirim mengenkripsi pesan yang akan dikirim agar pesan teracak dan tidak dapat dibaca
Pengirim/Penerima	Akses *Inbox* - Pengirim/penerima melihat, membaca pesan di dalam

	kotak masuk - <include> Daftar Pesan *Inbox*
Penerima	Terima Pesan - Penerima mendapatkan pesan dari pengirim
Penerima	Dekripsi Pesan - Penerima mendekripsi pesan yang teracak agar dapat Terbaca
Pengirim/Penerima	Akses *About* - Pengirim/penerima mendapatkan keterangan mengenai aplikasi seperti versi dan pembuat aplikasi

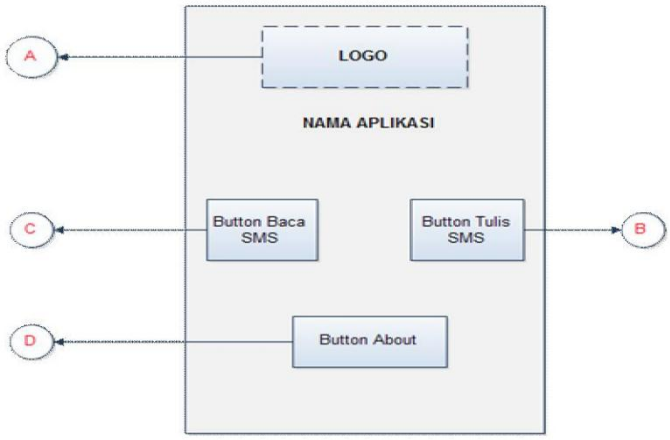
Tabel 3.1 Spesifikasi *Use Case Diagram* Sistem

3.2 Rancangan Aplikasi

Perancangan Tampilan aplikasi terdiri dari beberapa menu yaitu :

a. Rancangan Menu

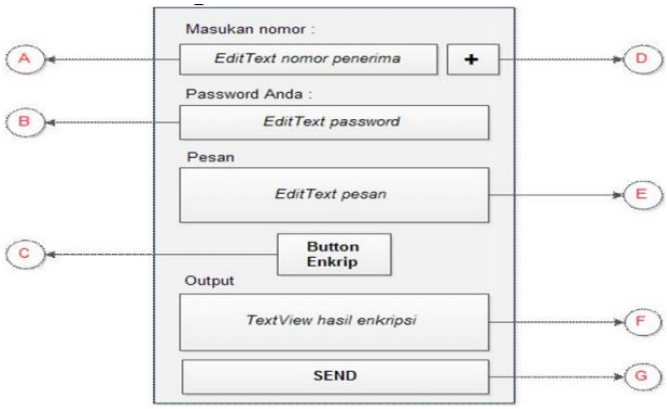
Awal Rancangan pada menu awal terdiri dari tampilan yang terdiri dari beberapa button, yaitu button *write* SMS (tulis pesan), *button read message* (baca pesan), *button about*, dan juga *button exit*.



Gambar 3.4 Layout tampilan awal program

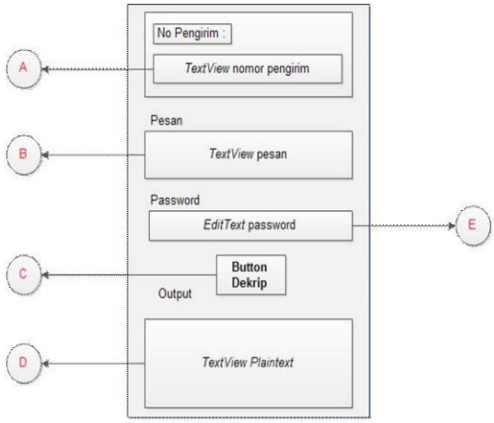
b. Rancangan Menu Tulis SMS

Pada menu ini menampilkan *user interface* untuk melakukan enkripsi data pada pesan yang akan dikirimkan oleh *user*. Akan terdapat beberapa tombol pada menu ini. Yaitunya terdapat tombol enkripsi SMS, tambah kontak, dan juga adanya *edit text* untuk memasukan pesan dan kemudian *text view* untuk melihat *chipertext*, dan juga *edit text* untuk memasukan *password*.



Gambar 3.5 Layout tampilan Tulis SMS

C. Rancangan Menu Baca SMS



Gambar 3.6 Layout menu Baca SMS

3.3 Analisa Penerapan Algoritma RC6 dalam Enkripsi SMS

Algoritma RC6 merupakan algoritma sederhana, fungsi yang digunakan merupakan fungsi yang sederhana dan hanya mengandalkan prinsip teknik cipher berulang (*iterated cipher)* untuk keamanan. Tampilan hasil enkripsi dan data hasil enkripsi yang diterima setiap karakternya memiliki panjang 8 bit, sedangkan sebagian telepon seluler hanya dapat menampilkan karakter dengan panjang 7 bit. Dengan demikian dalam penerapan algoritma RC6 pada SMS karakter-karakter yang akan dienkripsi diubah kedalam nilai ASCII, dimana nilai karakter dalam table ASCII ditambah dalam karakter special adalah 0 sampai 244, artinya satu karater ASCII akan diwakili oleh 8 bit, dimana 28 = 256. Sehingga, dalam 1 blok plainteks (32 bit) akan menyimpan 4 karakter dan setiap kali iterasi, maka akan diambil 16 karakter plainteks.

Apabila panjang plainteks atau panjang kunci kurang dari 16 karakter, maka akan dilakukan *padding*, yaitu dengan menambahkan bit ”0” (nol) di akhir teks, sehingga panjang teks mencukupi 16 karakter. Layar pada sebagagian besar telepon selular hanya dapat menampilkan karakter dengan panjang 7 bit dan pesan yang telah terenkripsi akan berbentuk *binary*, sehingga layar tidak akan menampilkan dengan semestinya. Oleh karena itu, pada aplikasi yang akan dibangun, untuk menampilkan pesan yang telah terenkripsi, ditambahkan informasi karakter yang terdapat pesan tersebut dengan format heksadesimal agar dapat ditampilkan dilayar dan informasinya lebih terbaca.

Algoritma RC6 yang akan digunakan dalam aplikasi enkripsi SMS yang akan dibangun dengan W sebesar 32 bit, R sebsar 20 kali putaran dan panjang kunci beragam lebih dari 1 karakter (8 bit). Langkah-langkah algoritma RC6 dalam pelaksanaan Proyek Madya ini akan dikelompokkan kedalam beberapa bagian, yaitu

1. Pembangkit Subkunci
   Kunci dari pengguna ini akan dimasukkan oleh pengguna pada saat akan melakukan proses enkripsi dan dekripsi. Kunci ini memiliki tipe data *string* dan memiliki panjang 16 byte (16 karakter).
2. Baca masukkan untuk proses enkripsi
   Yang dilakukan pada tahapan ini adalah membaca teks yang menjadi masukan pada proses enkripsi, yaitu *field* dari aplikasi enkripsi SMS. Pada proses enkripsi pesan, *field*-nya adalah isi pesan.
3. Enkripsi meliputi *whitening* awal, iterasi dan *whitening* akhir.
4. Baca masukkan untuk Proses Dekripsi
   Yang dilakukan pada tahapan ini adalah membaca teks yang menjadi masukkan pada proses dekripsi, yaitu *record* dari hasil pesan yang telah dienkripsi pada pengirim dan menjadi field pesan pada penerima.
5. Dekripsi merupakan kebalikan dari proses enkripsi.
   Langkah-langkah diatas akan dijelaskan dalam algoritma-algoritma berikut:

1. Algoritma Pembangkit Sub Kunci

```
Kamus

Type Word32 : 32 bit (tipe data 32 bit)
Kunci : String { kunci yang dimasukkan oleh pengguna}
I,  j, c, s, v : integer
A :integer
B : Integer
S : array [0..43] of word 32
L : array [0..43] of word 32

Function
ROTL (X:Word32; y: integer) – Word 32 {fungsi untuk merotasi bit
sebanyak variable kedua}

Algoritma
Input (kunci)
S(0) – b7e15163
For I – 1 to 43 do
S[i] – s[i-1] + 9e3779b9
Endfor
A – B – I – j – 0
V – 44
If {c>v} then
v – c
v v*3

For s – 1 to v do
A – S[i] – ROTL ((S[i] + A + B). 3
S – L[j] = ROTL (L[j] + A + B, A + B)
I – (i+1) mod 44
J – (j+1) mod c
Endfor
```

2. Algoritma Baca File Masukan Proses Enkripsi

```
Prosedur Baca_Masukan_proses_Enkripsi

{Input : Field masukan belum dibaca}
{Output : Field masukan dibaca per 16 karakter dan ditampung dalam
buffer. Pada proses ini pesan, filed nya adalah isi pesan }

Kamus
Field_masukan ; string
Buff : array [0..15] of char
i : integer

algoritma
Input (field_masukan)
i – 0
while (I <=15) and not (EOF) do
Read (field_masukan, Buff [i])
Endwhile
```

3. 1 Prosedur Whitening_awal

```
{input : blok kedua dan keempat belum dijumlahkan dengan sub kunci}
{output : blok kedua dan keempat yang telah dijumlahkan dengan sub
kunci}
Kamus
Type word32 : 32 bit (tipe data sebesar 32 bit)
I : word32 array [0..3] (blok enkripsi/planteks)
E : Array [0..43] of word 32 (sub kunci)

Algoritma
K[1] – X[1] + S[0]
K[3] – X[3] + S[1]
```

3.2 Algoritma Iterasi

```
Prosedur Iterasi
{input : keempat blok setelah whitening awal belum diproses}
{Output : keempat blok yang telah diproses dan saling dipertukarkan}

Kamus
Type word32 : 32 bit {tipe data sebesar 32 bit}
X : word array [0..3] {blok enkripsi/planteks}

Function
ROTL(X : Word32; Y : integer} – word32
{merotasi bit kekiri sebanyak variable kedua}

Temp : word32
U, t : word32
I : integer

Algoritma
For I – 1 to 20 do
t – ROTL ( (X[1]*(2*X[1]+1)), 5)
u – ROTL ( (X[3]*(2*X[3]+1)), 5)
X[0] – (ROTL ( (X[0] XOR t), u)) + S[2*i}
X[2] – (ROTL ( (X[2] XOR u), t)) + S[2*I + 1]
Temp – X[0]
X[1]  – X[1]
X[2]  – X[2]
X[3]  – Temp
End for
```

3.3 Algoritma Whitening Akhir

Prosedur *Whitening_akhir*
{input : blok pertama dan ketiga belum dijumlahkan dengan sub kunci}
Output : blok pertama dan ketiga yang telah dijumlahkan dengan sub kunci}

Kamus
Type word32 : 32 bit (tipe data sebesar 32 bit)
X : word32 array [0..3] blok enkripsi/planteks
S : Array [0..43] of word 32 (sub kunci)

Algoritma
X[0] – X[0] + S[42]
X[2] – X[0] + S[43]

4. Algoritma Baca File Masukan Proses Dekripsi

Prosedur Baca_File_Masukan_Proses_Dekripsi
{input : Field masukan berupa chiperteks}
(output : Field pada isi pesan yang berupa chiperteks dibaca per 16 karakter dan ditampung dalam buffer}

Kamus
Field_masukan : string
Buff : array [0..15]
i : integer

Algoritma
Input (field_masukan)
i – 0
while (i <= 15) and not (field_masukan.EOF) do
Read (isi_kolom, Buff [i])
Endwhile

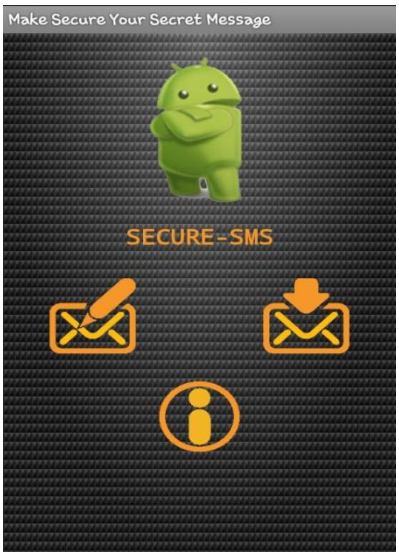
5. Algoritma Dekripsi

```
Prosedur Dekripsi
{input : keempat blok belum diproses}

{output : keempat blok yang telah diproses dans aling dipertukarkan}

Kamus
Type word32 : 32 bit {tipe data sebesar 32 bit}
X : word32 array [0..3] {blok dekripsi/ciperteks}

Function
ROTL (X:word32; Y:integer) – word32 {merotasi bit kekiri sebanyak variable
kedua}

Temp : word32
u, t :word32
I : integer

Algoritma
X[2] – X[2] – S[43]
X[0] – X[0] – S[42]
For I – 20 down to 1 do
Temp – X[3]
X[3] – X[2]
X[2] – X[1]
X[1] – X[0]
u – ROTL ( (X[3]*(2*X(3)+1)), 5)
t – ROTL ( (X[1]*(2*X[1]+1}}, 5)
X[2] – (ROTL (X[2] – S(2*1+1]), t) XOR u)
X[0] – (ROTL (X[0] – S[2*i]), u) XOR t)
End for

X[3] – X[3] – S[1]
X[1] – X[1] – S[0]
```

IV. PEMBAHASAN DAN ANALISA

4.1. Tampilan Menu Utama

Pada menu utama terdapat 3 tombol yang memiliki fungsi masing-masing yaitu, tombol Tulis pesan untuk menuju *activity* tulis pesan. Tombol kotak masuk berfungsi menuju *activity Inbox*. Sedangkan tombol *About* berfungsi untuk idenditas pembuat. Tampilanya adalah sebagai berikut:

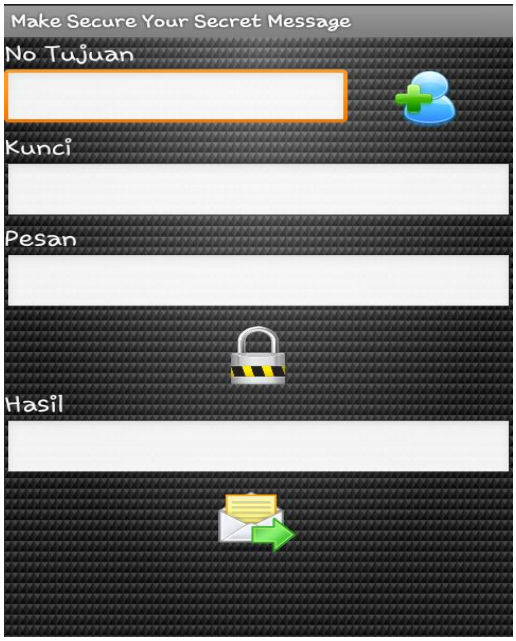


Gambar 4.1 Tampilan Menu Utama

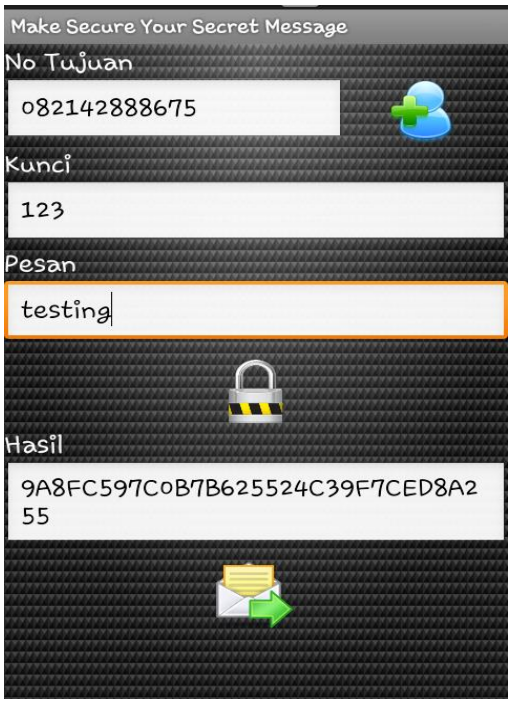
4.2. Form Tulis Pesan

Pada tab Tulis Pesan ini pengguna diharapkan untuk memasukan beberapa data di sub-sub menu yang ada,berikut sub menu dan fungsinya

1. No Tujuan, berfugsi untuk menginput nomor HP penerima.
2. Kunci, berfungsi untuk memprotect isi pesan.
3. Pesan, merupakan kolom yg digunakan untuk mengetik kata-kata yang ingin disampaikan.
4. Tombol enkripsi , berfungsi untuk mengenkripsi isi dari pesan yang sudah di ketik
5. Hasil, berfungsi untuk menampilkan kata-kata atau kode-kode yang sudah di enkripsi.
6. Tombol untuk mengirim pesan
7. Tombol untuk menampilkan kontak-kontak nomor HP yang sudah tersimpan.



Gambar 4.2 Form Tulis Pesan



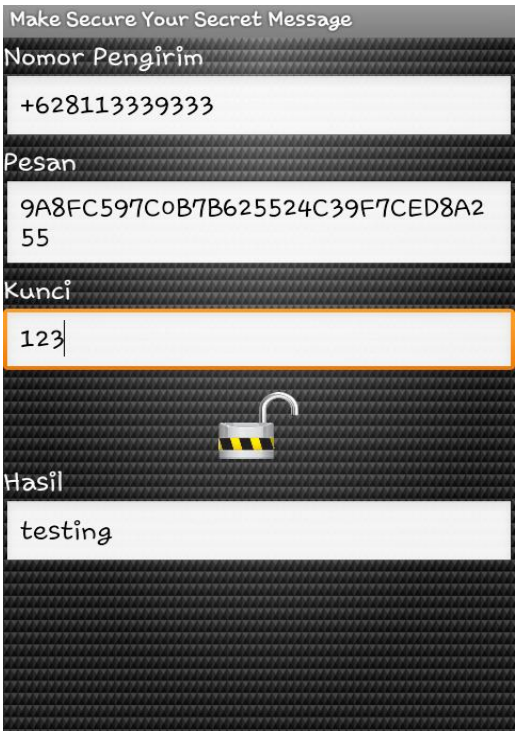
Gambar 4.3 Form Tulis Pesan yang sudah diisi

4.3. Form Baca Pesan

Pada tab Baca Pesan ini pengguna akan menerima pesan yg dalam kondisi masih terenkripsi,tampilan Baca Pesan adalah sebagai berikut:

1. Nomor pengirim, terisi otomatis nomor pengirim.
2. Pesan, ini pesan yang di kirim oleh pengirim dan masih dalam bentuk terenkripsi.
3. Kunci, merupakan kode sandi dari isi pesan yang akan di dekripsi,kode kunci penerima harus sama dengan kode kunci yang sudah di buat oleh pengirim.
4. tombol  untuk mendekripsi pesan yang diterima.
5. Hasil, merupakan isi pesan yang sudah di dekripsi.

Berikut Gambar 4.3 adalah tampilannya.



Gambar 4.4 Tampilan Form Baca Pesan

Tabel 4.1 Contoh Hasil Enkripsi dan Dekripsi sms

Contoh Enkripsi dekripsi	
Pesan	**Hasil**
1234567	AA10F25B57B9D3B9E9447AD59C301C54
!@#$/&	783F5B22613A6FE8BAD0EE50172E8F98
Terima Kembali	3FA63E674384BFD778FA5F2FD5D20F3F
1234567890123456	B65ABA956FDB71FBE7237C09FF5B14FEB 101449F18DOEA7366CEAE6F2473A4A9
ABCDE	26B357033546E2CEE8ED53D0E799ED33
abcde	20ACF7F082E311A6D33E1EC2CG0EA542
ABCDEFGHIJKLM NOP	52D19CE6D058E82ACAFE652E1C750F9B6 0790DC362925915415150C52B03565B
abcdefghijklmnop	19CE95936DE80AEF6818A1E92FDF16EEB 101449F18D0EA7366CEAE6F2473A4A9
ABCDefgh	40DE36A5FBEE6B49B8DEE383B675DBC3
abcdEFGH	705C4CBEC04904A9B886AD078AA29F40
Total 1234	544ADB5C359F76616003053463799D54
1234 Total	A28D9730EB94657E9B73F66A6CDC7E0F
Kode !@#$/	D24584835C081E4493498A37F8D4CC88
!@#$/ Kode	369D6943DA8A2885B62A56BFDB77250B
Kembali Rp. 5,000	706B7F5FC98D6B1F0FEB2C7E47CD613306 A8F92E6D9CC00ED1ED9C28822AF6B2

V. KESIMPULAN DAN REKOMENDASI

1. Penerapan algoritma kunci privat untuk enkripsi SMS pada telepon selular dapat meningkatkan keamanan. Pesan yang terenkripsi tidak akan dapat dibaca jika tidak didekripsi dengan menggunakan kunci yang benar, sehingga orang yang tidak mengetahui kunci yang sebenenarnya tidak dapat membaca pesan yang dikirimkan.
2. Algoritma RC6 dapat diimplementasikan dengan baik untuk melakukan enkripsi SMS yang bekerja pada jaringan GSM dengan mengirimkan pesan yang berbentuk binary.
3. Kekurangan dari implementasi algoritma RC6 untuk enkripsi SMS adalah pesan yang dikirimkan menjadi lebih besar karena harus bekerja pada 8 bit dan dibutuhkan *padding* untuk memenuhi panjang blok.
4. Semakin besar jumlah rotasi pada algoritma RC6, maka tingkat keamanan akan semakin baik, namun waktu yang diperlukan untuk melakukan enkripsi dan dekripsi akan semakin besar.
5. Pesan yang dikirim hanya data text berupa huruf, angka dan *symbol* sehingga untuk pengiriman gambar belum bisa dilakukan.
6. Versi android yang tidak kompatibel dengan aplikasi ini adalah versi *Jelly Bean* 4.2 keatas.

DAFTAR PUSTAKA

Ariyus, Dony. 2008, Pengantar Ilmu KRIPTOGRAFI, Penerbit ANDI, Yogyakarta.

Kurniawan, Yusuf, Ir. 2004. Kriptografi Keamanan Internet dan Jaringan Komunikasi, penerbit INFORMATIKA, Bandung.

Muhammad Ridho, Rika Fitriana, Sepri Hardianti, Suhendi, PERBANDINGAN ALGORTIMA RC4, RC5 DAN RC6, Tugas Jurnal Keamanan Komputer Oktober 2010, Jurusan Teknik Informatika Fakultas Sains dan Teknologi UIN Suska Riau.

Nazruddin Safaat H 2012. Pemrograman Aplikasi Mobile Smartphone dan Tablet PC Berbasis Android, penerbit INFORMATIKA, Bandung.

Nazruddin Safaat H 2013. Aplikasi Berbasis Android : Berbagai Implementasi dan Pengembangan Aplikasi Mobile Berbasis Android, penerbit INFORMATIKA, Bandung.

Reza Brianca Widodo, Makalah IF3058 Kriptografi Studi dan Perbandingan Algoritma RC6 dan Blowfish. Program Studi Teknik Informatika, Sekolah Teknik Elektro dan Informatika, Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia

Sunardi, Hari Murti, Hersatoto Listiyono, Aplikasi SMS Getway, Jurnal Teknologi Informasi DINAMIK Volume XIV, No.1, Januari 2009, Fakultas Teknologi Informasi, Universitas Stikubank Semarang.

Supardi, Yuniar.2008. Sistem Informasi Penjualan dengan JAVA. Jakarta:Elex Media Komputindo, Jakarta.

Yudi Prayudi dan Idham Halik, STUDI DAN ANALISIS ALGORITMA *RIVEST CODE 6* (RC6) DALAM ENKRIPSI/DEKRIPSI DATA, Seminar Nasional Aplikasi Teknologi Informasi, Juni 2005, Jurusan Teknik Informatika, Fakultas Teknologi Industri, Universitas Islam Indonesia Yogyakarta.