Penerapan Steganografi File Gambar Menggunakan Metode *Least Significant Bit* (LSB) dan Algoritma Kriptografi *Advanced Encryption Standard* (AES) Berbasis Android

Endar Nirmala

Teknik Informatika Universitas Pamulang
Jl. Surya Kencana No. 1 Tangerang Selatan
e-mail: endarnirmala@gmail.com

Submitted Date: March 14th, 2020 Reviewed Date: March 16th, 2020
Revised Date: March 23rd, 2020 Accepted Date: March 30th, 2020

Abstract

Sending messages in the form of image files among the public or business people using the SMS feature from time to time is increasingly easy and often done. Messages can be sent from business partners or from clients. If the file is used for business purposes, it is an important secret for the company, and very dangerous if it falls into the hands of an undue. Therefore the security and confidentiality of the files sent are important and are very necessary in the exchange of information in this case in the form of image files through internet media. The method used to overcome this problem applies a combination of Advanced Encrytion Standard (AES) Cryptographic Algorithm techniques integrated with the Least Significant Bit (LSB) method. The integration of these techniques aims to provide protection or security systems on secret file messages in the form of images sent. The results of this study are applications called "StegoKripto" which is a combination of steganography and cryptography. The application runs on the Android platform which can encrypt text messages and hide the messages in pictures in the format .jpg and .png.

Keywords: AES, LSB, Encryption, Data security, and Android.

Abstrak

Mengirim pesan dalam bentuk file gambar dikalangan masyarakat atau pembisnis menggunakan fitur SMS dari waktu ke waktu merupakan hal yang semakin mudah dan sering dilakukan. Pesan dapat dikirim dari partner bisnis atau dari client . Jika file itu digunakan untuk kepentingan bisnis maka menjadi rahasia penting bagi perusahaan tersebut, dan sangat berbahaya jika jatuh ketangan yang tidak semestinya. Oleh karena itu keamanan dan kerahasiaan file yang dikirim merupakan hal penting dan sangat diperlukan dalam pertukaran informasi dalam hal ini dalam bentuk file gambar melalui media internet. Metode yang digunakan untuk mengatasi masalah tersebut menerapkan perpaduan teknik Algoritma Kriptografi *Advanced Encrytion Standard* (AES) berintegrasi dengan metode *Least Significant Bit* (LSB). Integrasi teknik tersebut bertujuan untuk memberikan proteksi atau sistem keamanan pada pesan file rahasia dalam bentuk gambar/citra yang dikirim. Hasil penelitian ini merupakan aplikasi yang bernama "StegoKripto" yang merupakan perpaduan antar steganografi dan kriptografi. Aplikasi berjalan dengan platform Android yang dapat mengenkripsi pesan teks dan menyembunyikan pesan tersebut dalam gambar dengan format .jpg dan .png.

Kata kunci : AES, LSB, Enkripsi, Keamanan data, dan Android.

1 Pendahuluan

Keamanan komputer atau lebih dikenal *cybersecurity* merupakan perlindungan sistem komputer dan informasi terhadap bahaya pencurian atau kerusakan yang terjadi pada perangkat keras atau perangkat lunak dan informasi yang terdapat didalamnya.

Beberapa modus kejahatan terkait dengan keamanan komputer dan kerahasiaan dalam pertukaran data informasi secara elektronik atau media internet cukup banyak, diantaranya

modus tersebut adalah: *Interruption* dimana serangan dilakukan dengan merusak data, *Interception* dimana ancaman dari pihak yang tidak berhak memperoleh akses untuk mengambil data atau informasi.

Modification merupakan ancaman dari pihak lain yang berhasil mendapatkan akses untuk mengubah data dan informasi, dan *Fabrication* yaitu menyisipkan file atau record pada program aplikasi. Siapapun dapat memperoleh data/informasi yang dikirim melalui media internet jika tidak ada sistem keamanan yang terjamin.

Agar pesan atau informasi dapat terjaga dengan baik sehingga dapat terpelihara dari bermacam-macam gangguan atau bahaya yang selalu mengintai dalam pengiriman pesan maka diperlukan sistem keamanan yang benar-benar dapat dipertanggung jawabkan. Pada saat informasi diterima, keaslian informasi atau data amatlah penting baik ketika data diterima atau dikirim. Oleh sebab itu pemilik informasi akan mengalami kerugian jika informasi atau data yang dikirim jatuh kepada pihak yang tidak diinginkan. Dengan demikian teknik atau metode untuk mengamankan informasi/data sangat dibutuhkan. (Munir, 2019)

Kriptografi merupakan salah satu teknik keamanan yang dikembangkan untuk menjaga dan melindungi pesan rahasia sehingga terhindar dari pihak lain yang tidak berhak. Proses kerja kriptografi akan menyandikan pesan atau informasi sehingga tidak dapat dimengerti oleh pihak lain. Konsep utama kriptografi melakukan enkripsi dan deskripsi (Imron & Suryarini, 2014).

Algoritma kriptografi *Advanced Encryption Standard* merupakan teknik kriptografi agar data aman dan terjaga keasliannya. Blok *chpertext* simenttrik merupakan algoritma AES yang dapat mendeskripsi (*derhipr*) dan mengekskripsi (enchiper) data/informasi. Kunci kriptigrafi 128, 192 dan 256 bits digunakan agar dapat melakukan enkrip dan dekrip data/informasi pada blok 128 bits. Kelebihan pada keamanan yang dimiliki AES diantaranya karakteristik algoritma dengan implementasinya dan karakteristik algoritma serta kecepatan. (Ilyas & Widodo, 2014)

Selain kriptografi teknik lain untuk menjaga rahasia pesan adalah steganografi. Dengan teknik steganografi pesan/informasi yang akan dikirim dapat disembunyikan dalam suatu media agar informasi/pesan tidak dikenali oleh pihak lain melalui panca indra manusia.

Perbedaan kriptografi dan steganografi adalah pada kriptografi makna pesan dirahasiakan tetapi keberadaannya tetap ada, sedangkan steganografi menutupi dan merahasiakan pesan dengan menyembunyikannya. (Rahmat, 2010)

Stegokey atau kunci rahasia yang harus dimiliki oleh Steganografi ketika melakukan proses *encoding* dan *decoding.* Encoding yaitu melakukan pengkodean untuk pesan yang akan dibuat. Decoding yaitu memaknai pesan berdasarkan kode yang telah dibuat. Pengkodean disisipkan pada pesan yang dikirim dan akan diekstrak ketika diterima. Penyisipan dan ekstrak pesan dapat dilakukan hanya pada orang yang berhak saja.

Media gambar atau citra merupakan salah satu media digital yang dapat digunakan sebagai media untuk menampung dan menyembunyikan pesan ketika implementasi (*covertext*) tidak disadari keberadaannya oleh pihak lain. Gambar stego (stegotext) memiliki makna yang sama dengan bentuk aslinya dalam steganografi. Kesamaan makna tersebut secara visual sebatas kemampuan indera manusia, maksudnya mata manusia tidak dapat membedakan gambar asli yang tidak memiliki pesan dengan gambar stego. (Munir, 2019)

Teknik penyembunyian pesan, dimana pesan diubah ke bentuk biner dalam citra digital terdapat pada lokasi bit terendah dalam metode steganografi Least Significant Bit (LSB). Bit terendah akan dibaca dengan teknik pemecahan analisis dapat membongkar metode LSB jika tidak disertai dengan sistem keamanan. (Rakhmat, 2010).

Berdasarkan paparan diatas, peneliti akan mengimplementasikan kriptografi dan steganografi menggunakan algoritma *Advanced Encryption Standard* (AES) dan *Least Significant Bit* (LSB) dengan format JPG dan PNG sebagai media penampung pesan yang berjalan pada platform Mobile Android guna memberikan perlindungan keamanan data pada sebuah pesan.

2 Tinjauan Pustaka

Berikut beberapa teori terkait dengan sistem keamanan komputer:

a. Kriptografi

Kata kriptografi berasal dari bahasa Yunani, "*kryptós*" yang berarti tersembunyi dan "*gráphein*" yang berarti tulisan. Sehingga kata

kriptografi dapat diartikan berupa frase "tulisan tersembunyi". Menurut beberapa ahli ada yang mengatakan bahwa kriptografi merupakan ilmu matematika dalam hal ini transfortmasi, namun ada pula yang mengatakan bahwa kriptografi merupakan seni. Pesan yang dikirim ditransformasi untuk menyembunyikan pesan dan jika transformasinya dikembalikan menjadi bentuk yang dapat dimengerti kembali. Selain itu pesan yang disembunykan juga dapat menggunakan cara yang berbeda atau unik sehingga ketika mengirim pesan rahasia memiliki nilai seni atau estetika dengan demikian kriptografi berkembang sebagi seni untuk merahasikan pesan.

Konsep utama kriptografi adalah enkripsi dan dekripsi. Enkripsi mengubah pesan asli atau *plainteks* menjadi *chipherteks,* atau pesan yang tidak dimengerti seperti aslinya melalui proses penyandian. Deskripsi kebalikan dari enkripsi yaitu mengembalikan *chiperteks* menjadi *plainteks.* Transformasi kunci merupakan parameter yang dibutuhkan oleh deskripsi dan enkripsi. (Rakhmat, 2010).

b. *Advanced Encryption Standard* (AES)

NIST (*Nastional Institute of Standards and Technology*) menjadikan teknik enkripsi dari AES (*Advanced Encryption Standard*) sebagai standard FIPS (*Federal Information Processing Standards*). Untuk menjadi standar AES melalui beberapa tahap seleksi. Enkripsi Rjindael merupakan teknik enkripsi yang terpilih dari beberapa calon enkripsi yang dijadikan AES. Seperti DES, teknik Enkripsi Rijindael temasuk jenis block chiper.

Penggunaan substitusi (S-boxes) AES pada naskah sebagai pembeda utama antara teknik enkripsi AES dan teknik enkripsi DES. Exclusive or digunakan untuk mengoperasikan naskah pada fungsi chiper f dengan mensubstitusi s- boxes yang digunakan DES, dengan demikian DES secara langsung tidak menggunakan substitusi pada naskah. Kunci enkripsi yang digunakan AES terhadap naskah, sedangkan substitusi S-box digunakan DES hanya dalam fungsi cipher *f* yang hasilnya kemudian dioperasikan terhadap naskah menggunakan *exclusive or*, jadi DES tidak menggunakan substitusi secara langsung terhadap naskah. (Ilyas & Widodo, 2014)

Secara Garis besar Algoritma AES beroperasi pada blok 128 bit dengan kunci 128 bit sebagai berikut (di luar proses pembangkitan *round key*):

a. *AddRoundKey* : Tahap *initial round,* antara *chiper key* dan state awal (*plainteks*) melakukan XOR.
b. Putaran sebanyak *Nr* – 1 kali. Setiap putaran akan melakukan prosess :
    1. *SubBytes* : tabel substitusi (S-box) digunakan untuk subtitusi byte
    2. *ShiftRows* : Wrapping dilakukan untuk menggeser baris-baris *array state*.
    3. *MixColumns* : data pada masing-masing kolom *array state* diacak
    4. *AddRoundKey* : melakukan XOR antara *state* sekarang *round key*.
c. *Final round* : proses untuk putaran terakhir:
    1. SubBytes
    2. ShiftRows
    3. AddRoundKey

Algoritma AES memiliki 3 parameter:

a. *Plaintext* : data masukan yang terdapat pada array yang berukuran 16 byte.
b. *Ciphertext* : hasil endkripsi yang terdapat pada array yang berukuran 16 byte.
c. *Key* : kunci chipering atau chiper key yang terdapat pada array yang berukuran 16 byte.

c. Steganografi

Steganografi (*steganography*) merupakan cara menyembuyikan informasi atau data rahasia pada media digital agar keberadaaan informasi atau data rahasia tersebut tidak diketahui orang. Citra, suara (audio), teks, dan video merupakan media digital dari Steganografi digital. (Munir, 2019)

Pada steganografi terdapat beberapa istilah yang berkaitan, antara lain (Pratama dkk, 2011):

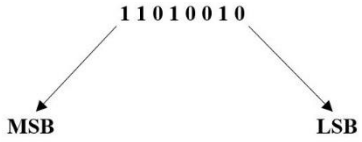
a. *Hiddenteks* atau *embedded message*, yaitu pesan yang di sembunyikan.
b. *Coverteks* yaitu *embedded message* yang disembunyikan dalam pesan
c. *Stegoteks* (*stego-object)*, yaitu *embedded message* yang terdapat pada pesan

Pesan yang disisipkan pada media coverteks disebut *encoding*, sedangkan pesan yang diekstak dari stegoteks disebut *decoding*. Proses encoding dan decoding dalam melakukan proses penyisipan dan ekstrasi pesan agar dapat digunakan pada pihak yang berhak membutuhkan kunci rahasia (*stegokey*).

d. *Least Significant Bit* (LSB)

Satuan penyimpanan data dalam komputer disebut byte (1 byte=8 bit). Byte terdiri dari

rangkaian bit dimana bit bagian kanan memiliki arti yang disebut *Most Significant Bit* atau MSB dan bit terkiri merupakan bit yang jarang memiliki arti disebut *Least Significant Bit* atau LSB. Dapat terlihat di gambar 1 yang merupakan contoh biner dengan 8 bit.



Gambar 1 *Most Significant Bit* (MSB) dan *Least Significant Bit* (LSB)

Bit LSB merupakan bit yang cocok untuk diganti. Perubahan hanya akan mengganti nilai byte satu lebih tinggi atau satu lebih rendah dari nilai sebelumnya.

Pixel merupakan kumpulan dari titik-titik yang membentuk gambar bitmap. Setiap gambar mempunyai pixel dengan nilai yang berbeda-beda.

Sebagai contoh berdasarkan gambar diatas, disisipkan karakter T dengan terlebih dahulu mengubah karakter T ke kode ASCII. T memiliki kode ASCII 84, dari nilai tersebut dikonversi ke biner menjadi 01010000.

Segmen citra sebelum disisipkan
(00100111 10101001 11101000)
(00110111 10101000 11101001)
(11001000 00100111 11101001)

Segmen citra setelah disisipkan "T"
(00100110 11101001 11001000)
(00100111 11001000 11101000)
(11001000 00100110 11101000)

Berdasarkan dari data yang disisipkan 010100000 terjadi perubahan bit dari citra sebelum disisipkan. Perubahan terjadi pada bit sisi yang paling kanan dari 8 bit yang ada. (Munir, 2019)

Informasi dengan ukuran terbatas mampu disimpan dengan metode LSB pada steganografi. Sebagai contoh data dengan ukuran 100 bit disimpan pada media citra yang berukuran citra 24-bit (R=8-bit, G=8-bit, B=8-bit), jika informasi disimpan menggunakan 1 pixel pada setiap masing-masing komponen (RGB), media citra yang dibutuhkan berukuran 34 pixel atau sama dengan 34 x 3 x 8 = 816 bit (8 kali lipat). Dengan demikian maksimum informasi yang dapat ditampung pada citra 24 bit berukuran 1/8 dari ukuran citra yang menampung.

3 Analisis dan Perancangan

Analisis dan Perancangan meliputi beberapa tahap, yaitu:

a. Metode Penelitian

Tahapan dalam metode penelitian terdiri dari beberapa tahap, dimulai dari studi pustaka atau literatur, melakukan perancangan aplikasi, kemudian membuat aplikasi, dilanjutkan dengan melakukan uji coba, dan membuat laporan.

b. Design Sistem

Teknik pengamanan pesan teks dalam penelitian ini adalah penggabungan antara kriptografi dan steganografi. *Advanced Encryption Standard* (AES) merupakan algoritma kriptografi yang digunakan, sedangkan untuk *Least Significant Bit* (LSB) merupakan steganografi yang digunakan.

Perancangan aplikasi StegoKripto ini digunakan untuk mengenkripsi sebuah pesan teks (plainteks) yang telah diinputkan, kemudian melakkan encode pesan tersebut yang telah dienkripsi (cipherteks) ke dalam sebuah gambar. Disamping itu, aplikasi ini juga dapat mengembalikan pesan yang telah terenkripsi (cipherteks) pada gambar dengan melakukan decode gambar tersebut, serta mendekripsi pesan yang telah terenkripsi (cipherteks) menjadi pesan awal (plainteks). (Yahya, 2019)

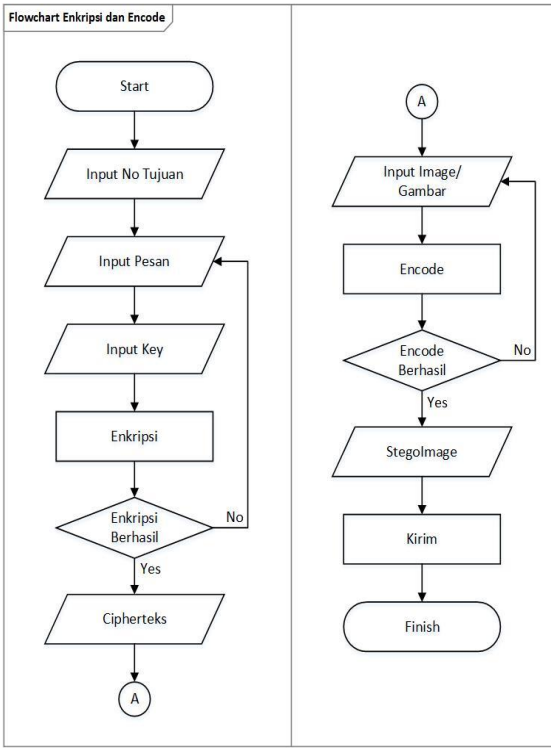
Berikut tahapan proses yang akan dipaparkan :

1. Analisis Alur Algoritma

Cara kerja dari penggabungan kedua algoritma tersebut adalah:

a. Proses Enkripsi dan Encode Pesan

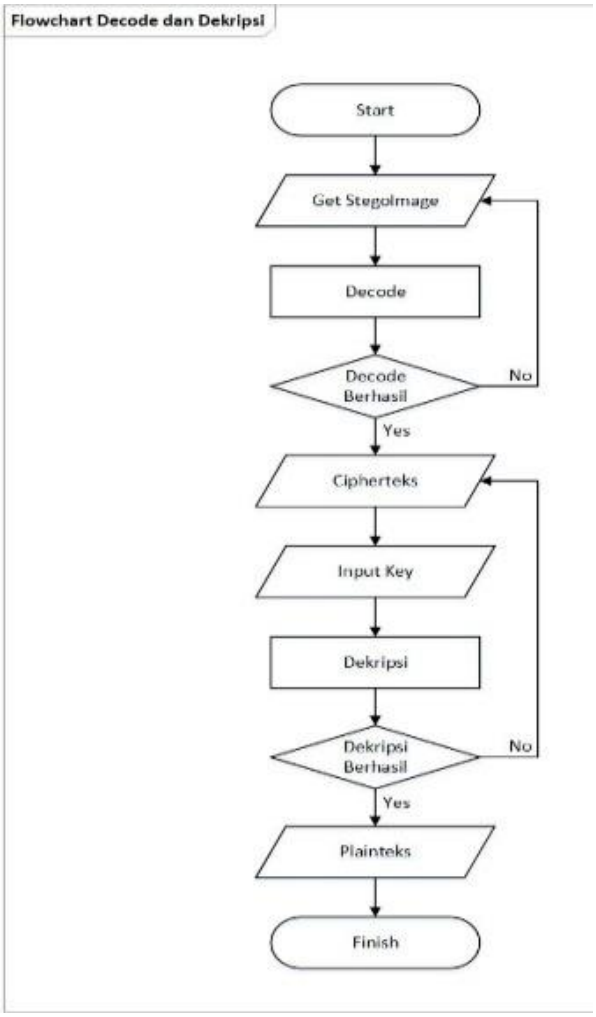
Plainteks sebuah pesan teks dienkripsi menghasilkan cipherteks. Cipherteks hasil enkripsi kemudian disisipkan ke dalam sebuah gambar (encode) dan menghasilkan gambar yang berisi cipherteks (Stegoimage). Alur *flowchart* pada proses ini akan dideskripsikan sebagai berikut:



Gambar 2. *Flowchart* Enkripsi dan Encode

b. Proses Decode dan Dekripsi Pesan

Proses decode dilakukan pada gambar yang telah berisi pesan cipherteks (stegoimage) untuk mendapatkan cipherteks (pesan rahasia). Setelah cipherteks berhasil didapatkan kemudian dilakukan proses dekripsi pesan atau mengubah cipherteks menjadi plainteks. Alur (*flowchart*) pada proses ini akan dideskripsikan sebagai berikut:



Gambar 3. *Flowchart* Dekripsi dan Decode

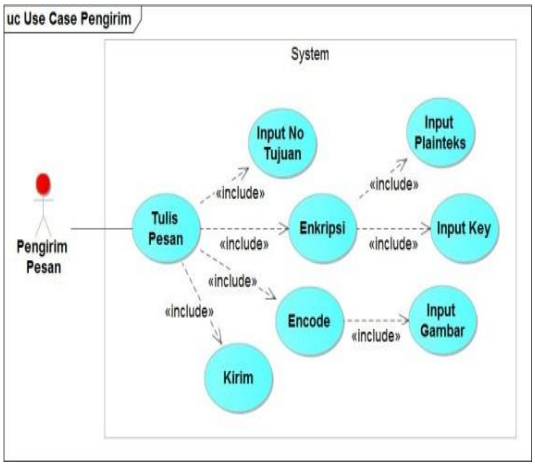
2. Perancangan Aplikasi

Perancangan aplikasi dimulai dengan pembuatan *use case diagram*, *sequence diagram* dan dilanjutkan pembuatan *user interface*. Berikut tahapan perancangan aplikasi :

a. *Use Case* Diagram

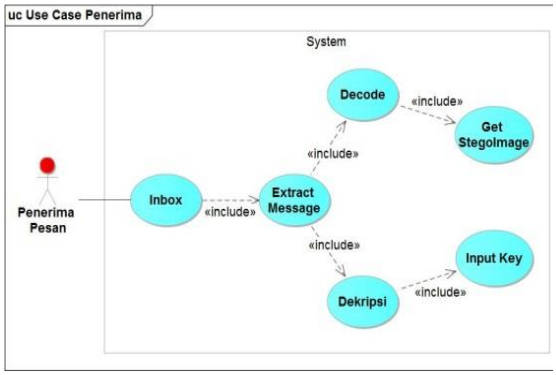
Use Case Diagram dapat menjelaskan atau mendeskripsikan interaksi antara user dengan sistem serta memberikan narasi tentang apa yang sistem lakukan. (Nugroho, 2010) Pada bagian ini penulis akan membuat *Use Case* Diagram untuk pembuatan aplikasi, dengan membagi use case menjadi 2 bagian, yaitu:

1. *Use Case* Diagram Pengirim Pesan



Gambar 4. Use Case Diagram Pengirim Pesan

2. *Use Case* Diagram Penerima Pesan

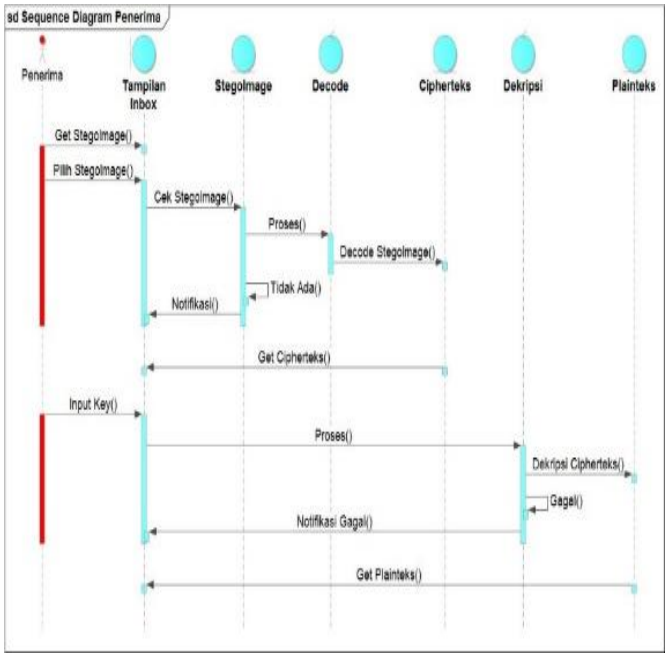


Gambar 5 Use Case Diagram Penerima Pesan

b. Sequence Diagram

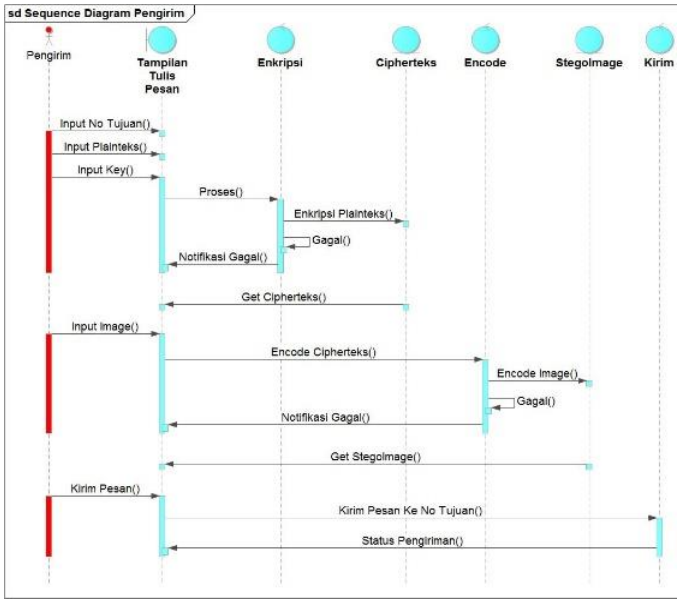
Dalam pembuatan aplikasi ini, Sequence Diagram yang diusulkan dibagi menjadi 2 bagian, yaitu:

1. Sequence Diagram Pengiriman Pesan



Gambar 6. Sequence Diagram Pengirim Pesan

2. Sequence Diagram Penerimaan Pesan



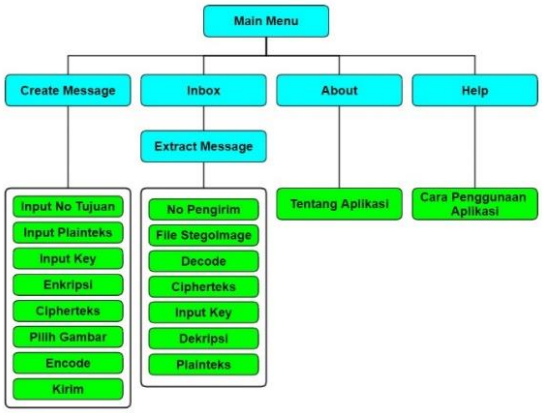
Gambar 7. Sequence Diagram Penerima Pesan

c. Perancangan *User Interface*

Form akan dikelompokkan berdasarkan proses aplikasi untuk mempermudah interasksi sisten dan user. Form tersebut terdiri dari *Main Menu*, *Form Create Message*, *Form Inbox*, *Form About*, dan *Form Help*. Berikut rancangan desain antarmuka pemakaian (*user interface*)

1. Hierarki Menu

Berikut adalah desain struktur dari aplikasi:



Gambar 8. Desain Struktur Aplikasi

2. Rancangan Tampilan Form *Main Menu*

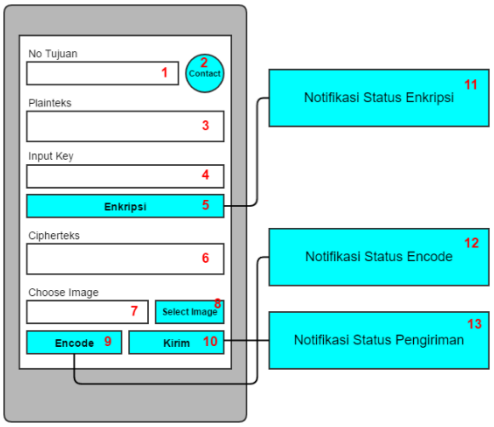
Berikut ini adalah rancangan desain antarmuka pada form *Main Menu*:

Gambar 9. Form *Main Menu*

Form main menu terdiri dari dari 4 menu utama yaitu *Create Message*, *Inbox*, *About*, dan *Help*.

3. Rancangan Tampilan Form *Create Message*

Berikut ini adalah rancangan desain antarmuka pada form *Create Message*:

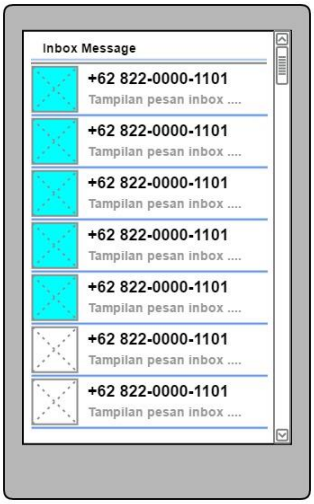


Gambar 10. Form *Create Message*

Pada desain tampilan form *create message* memperlihatkan bagaimana plainteks dienkripsi dan cipherteks melakukan encode ke dalam gambar penampung. Setelah proses enkripsi dan encode berhasil, akan menghasilkan file StegoImage yang siap dikirim ke no tujuan.

4. Rancangan Tampilan Form *Inbox*

Berikut ini adalah rancangan desain antarmuka pada form *Inbox*:



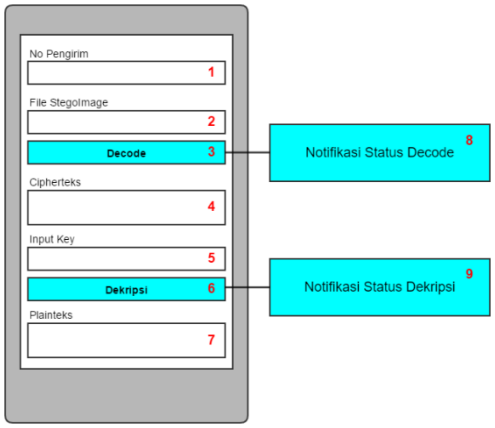
Gambar 11. Form *Inbox*

Pada desain tampilan form *inbox* memperlihatkan list keseluruhan pesan yang

diterima oleh user penerima pesan. Pada tampilan form ini user diminta untuk memilih pesan. Setelah user memilih pesan akan tampil form *extract message*.

5. Rancangan Tampilan Form *Extract Message*

Berikut ini adalah rancangan desain antarmuka pada form *Extract Message*:



Gambar 12. Form *Extract Message*

Pada design tampilan form *extract message* memperlihatkan bagaimana pesan StegoImage dari pengirim didecode (pengambilan cipherteks dari StegoImage) dan didekripsi (merubah cipherteks menjadi plainteks). Pada form inilah terdapat proses decode dan enkripsi guna mendapatkan pesan awal/asli (plainteks).

6. Rancangan Tampilan Form *About*

Berikut ini adalah rancangan desain antarmuka pada form *About*:

Gambar 13. Form *About*

Pada desain tampilan form *about* berisi deskripsi tentang aplikasi.

7. Rancangan Tampilan Form *Help*

Berikut ini adalah rancangan desain antarmuka pada form *Help*:

Gambar 14. Form *Help*

Pada desain tampilan form help berisi tentang tata cara penggunaan aplikasi. Setelah membaca isi halaman tekan tombol *closed*.

4 Implementasi dan Pengujian

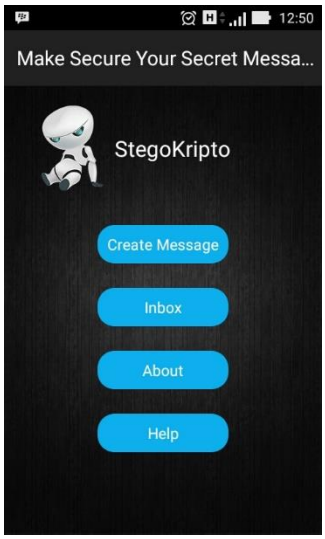
Pada bagian ini akan dilakukan implementasi dan pengujian dari aplikasi yang dibuat.

a. Implementasi Antarmuka (*User Interface*)

Agar antar User dan sistem dapat berinteraksi dengan baik diperlukan interface yang user friendly sehingga memudahkan penggunda. Untuk itu interface berdasarkan prosesnya dikelompokkan menjadi : pengiriman pesan dan penerimaan pesan yaitu enkripsi dan encode untuk pengiriman pesan kemuadian decode dan dekripsi untuk penerimaan pesan. Tampilan awal meliputi: menu utama (*main menu*), *create message*, *inbox*, *extract message*, *about* dan *help*. Berikut implementasi hasil rancangan *interface:*

1. Form Menu Utama (*Main Menu*)

Halaman ini menampilkan menu utama yang terdiri dari menu *Create Message*, *Inbox*, *About*, dan *Help*.

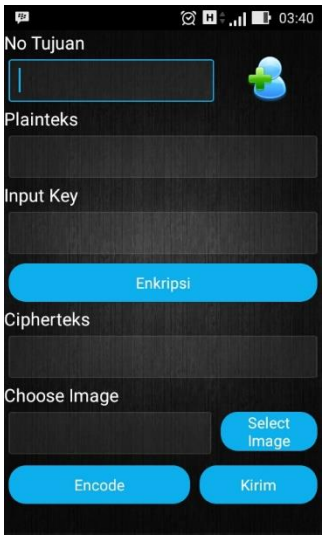


Gambar 15. Form *Main Menu*

Pada form *Main Menu* tersebut, user dapat memilih *Create Message* untuk membuat pesan baru, *inbox* untuk melihat pesan masuk dan melakukan proses pengekstrakan pesan, dan *About* yang berisi deskripsi tentang aplikasi.

1. Form *Create Message*

Tampilan berikut merupakan tampilan form *Create Message*:

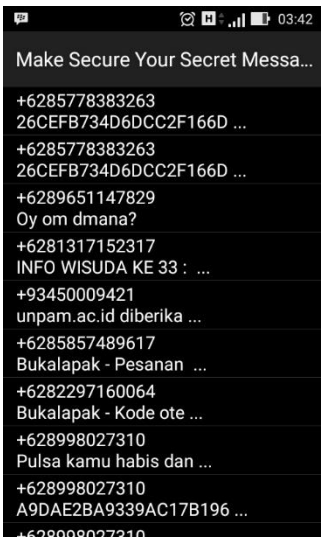


Gambar 16. Form *Create Message*

Tampilan form *Create Message* ditampilkan melalui *Main Menu* > *Create Message*. Pada tampilan *Create Message* ini plainteks diproses menjadi cipherteks dan StegoImage, kemudian pesan yang telah menjadi file StegoImage dikirim ke no tujuan. Pada tampilan form diatas user diminta untuk memasukkan no tujuan, pesan teks dan key kemudian user diminta untuk mengklik tombol Enkripsi untuk mengenkripsi pesan. Apabila pesan berhasil dienkripsi sistem akan menampilkan cipherteks (pesan hasil enkripsi). Selanjutnya user diminta untuk memilih file gambar untuk penampung cipherteks (pesan yang telah terenkripsi), dan user diminta untuk mengklik tombol Encode. Apabila proses encode berhasil, sistem akan menampilkan notifikasi proses encode berhasil dan pesan pun siap untuk dikirim ke no tujuan.

2. Form *Inbox*

Tambilan berikut tampilan dari form *Inbox*:



Gambar 17. Form *Inbox*

Tampilan menu *Inbox* ditampilkan melalui *Main Menu* > *Inbox*. Form ini hanya menampilkan list pesan yang berisi semua pesan yang diterima oleh user penerima pesan. Di tahap ini user yang menerima pesan dari pengirim diminta untuk memilih pesan, yang nantinya akan menampilkan form *Extract Message*, di mana form ini berisi pesan StegoImage dari pengirim, proses decode (pengambilan cipherteks dari StegoImage) dan proses dekripsi (mengubah cipherteks menjadi plainteks).

3. Form *Extract Message*

Tampilan berikut merupakan tampilan form *Extract Message*:

Gambar 18. Form *Extract Message*

Tampilan menu *Extract Message* ditampilkan setelah user penerima pesan memilih pesan pada form *inbox* (melalui *Main Menu* > *Inbox* > *Pilih Pesan*). Pada tampilan form *Extract Message*, user yang mendapat kiriman pesan berupa file StegoImage membuka menu *inbox*. Kemudian memilih pesan StegoImage yang hendak didecode dan didekripsi. Pesan yang telah dipilih kemudian didecode. Bila proses decode berhasil maka Cipherteks (Pesan terenkripsi) akan muncul pada *textarea* Cipherteks. Selanjutnya user diminta untuk menginputkan kunci (*key*) lalu klik tombol dekripsi. Bila proses dekripsi berhasil maka akan muncul teks asli/awal (Plainteks) pada *textarea* plainteks.

4. Form *About*

Berikut ini adalah tampilan dari form *About*:

Gambar 19. Form *About*

Menu tampilan *About* ditampilkan melalui *Main Menu* > *About*. Form *About* ini hanya terdapat deskripsi tentang aplikasi.

b. Pengujian Sistem

Pengujian software diperlukan untuk memastikan software yang dikembangkan sesuai dengan persyaratan (*requirement*) dan tidak ada cacat. Cacat software dapat menyebabkan business process tidak didukung oleh software yang dikembangkan, dan perlu perbaikan atau pengerjaan ulang jika banyak cacat (Saifudin & Wahono, 2015).

Pada pengujian aplikasi digunakan Teknik black box yang memiliki arti bahwa pengujian yang dilakukan hanya mengambil hasil eksekusi melalui data uji dan memeriksa fungsional dari perangkat lunak (Krismadi, et al., 2019). Pengujian dilakukan dengan memberikan masukan dan membandingkan hasilnya apakah sudah sesuai harapan atau tidak. Jika belum sesuai maka dilakukan perbaikan sampai sesuai dengan persyaratan (*requirement*).

Bagian ini merupakan pengujian sistem yang meliputi**:**

1. Data Pengujian

Pengujian dilakukan dengan menyisipkan pesan 46 karakter. Ukuran pixel 277x277 (ukuran:4.77 Kb) dengan type file PNG sebagai media penampung pesan.

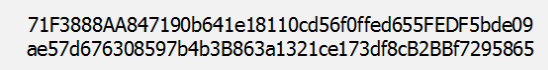
2. Proses dan Hasil Pengujian

Masuk ke menu *Create Message*, kemudian melakukan proses *generate key* dengan merandom 128 bit sesuai dengan tipe enkripsinya. Pada Class Java.security, SecureRandom.Class menggunakan SHA1PRNG agar menghasilkan bilangan random. Dengan menggunakan key yang telah *di-generate* mengubah plainteks dalam bentuk byte pada proses enkripsi, dilanjutkan dengan menggunakan mode ENCRYPT_MODE dan AES yang berasal dari pemanggilan *cipher instance*..

Byte dari hasil plainteks diubah ke dalam bentuk bilangan hexadesimal dan kemudian dikonversi ke dalam bentuk *string*. Hasil dari konversi tersebut menghasilkan pesan acak berupa hexadesimal dalam bentuk string. yang disebut dengan cipherteks. Sebagai contoh plainteks (46 karakter) dan cipherteks.

Ini adalah contoh implementasi kriptografi AES

Gambar 20. Plainteks (pesan awal/asli)



Gambar 21. Cipherteks (hasil dari enkripsi)

Setelah proses enkripsi selesai, dilanjutkan dengan proses encode cipherteks. Pada gambar yang dijadikan integer array satu dimensi pixel diambil pada awal proses encode, kemudian melalui proses konversi menjadi *byte* array. Penyisipan pesan pada gambar dilakukan dengan mengkonversi chipherteks ke dalam byte dalam bentuk string, ini dilakukan untuk mengubah masing-masing *byte* Red, Green, dan Blue. Melalui operasi shift pesan disisipkan pada masing-masing RGB pada bit terkahir pada.

Proses decode dilakukan dengan mengubah ke dalam byte array menjadikan ke dalam array integer satu dimensi dengan mengambil ukuran pixel pada gambar. Pesan yang terdapat pada gambar diambil, dengan menggeser masing-masing *byte*. Proses dekripsi dilanjutkan setelah isi pesan diperoleh, dalam bentuk pesan acak (hexadicemal) yang dihasilkan.

Konversi pesan dan key kedalam byte sebagai awal proses dekripsi, kemudian untuk mendapatkan plaintext harus melakukan *gerate key*. Agar pesan dapat dibaca pesan diubah ke dalam bentuk string.

5 Penutup

Berdasarkan hasil penelitian yang telah dipaparkan diperoleh kesimpulan dan saran sebagai berikut:

a. Kesimpulan

Berdasarkan hasil penelitian dan uraian-uraian serta pengujian pada aplikasi yang telah dijelaskan pada paparan sebelumnya, maka peneliti menarik kesimpulan, bahwa:

1. Algoritma kriptografi Advanced Encryption Standard (AES) dapat memberikan proteksi keamanan pada teks pesan, dengan cara mengenkripsi plainteks pesan yang dapat dipahami oleh pihak yang tidak diinginkan menjadi cipherteks yang sulit untuk dipahami sehingga dapat memberikan rasa aman bagi siapapun yang membutuhkan proteksi keamanan pada pesan teks.
2. Berdasarkan penerapannya, penggunaan algoritma kriptografi Advanced Encryption Standard (AES) pada pesan teks memakan waktu proses enkripsi dan dekripsi yang tergolong singkat sehingga sangat efisien untuk digunakan.
3. Algoritma S*teganografi Least Significant Bit* (LSB) dapat memberikan proteksi keamanan tambahan pada teks pesan, yaitu dengan menyembunyikan cipherteks (plainteks hasil enkripsi kriptografi AES) ke dalam sebuah gambar (image). Image hasil proses sisipan (encode) menghasilkan file StegoImage. StegoImage ini cukup baik, yakni secara kasat mata tidak terdapat perbedaan antara image asli dengan StegoImage (gambar yang telah disisipi pesan).
4. Penggabungan algoritma kriptografi Advanced Encryption Standard (AES) dan algoritma steganografi Least Significant Bit (LSB) menghasilkan proteksi yang sangat baik. Pesan terlindungi dengan algoritma kriptografi dan tersembunyi dalam sebuah gambar.
5. Algoritma kriptografi Advanced Encryption Standard (AES) dan algoritma steganografi Least Significant Bit (LSB) dapat digabungkan dan diterapkan pada perangkat mobile android.

b. Saran

Dalam penerapan algoritma kriptografi dan steganografi, serta pembuatan aplikasi StegoKripto ini masih terdapat banyak kekurangan, sehingga perlu diadakan penelitian dan pengembangan lebih lanjut tentang keamanan data teks pesan. Agar penelitian selanjutnya lebih baik maka disarankan:

1. Pembuatan aplikasi dapat menggabungkan algoritma kriptografi dan steganografi yang dapat diterapkan pada pesan file seperti file dokumen (.doc), file excel (.xls), file autocad (.dwf), dan lain sebagainya.
2. Media untuk penampung steganografi tidak hanya file gambar namun juga dapat diterapkan dengan media lainnya seperti audio atau video.
3. Disarankan agar pembuatan aplikasi yang dapat dijalankan pada platform lainnya, seperti Linux, iOS, atau Windows Phone.

Daftar Pustaka

Ilyas, I. A., & Widodo, S. (2014). Kriptografi File Menggunakan Metode AES Dual Password. *Prosiding Seminar Ilmiah Nasional Komputer dan Sistem Intelijen (KOMMIT), Vol 8.* Depok: Universitas Gunadarma.

Krismadi, A., Lestari, A. F., Pitriyah, A., Mardangga, I. W., Astuti, M., & Saifudin, A. (2019). Pengujian Black Box berbasis Equivalence Partitions pada Aplikasi Seleksi Promosi Kenaikan Jabatan. *Jurnal Teknologi Sistem Informasi dan Aplikasi*, *2*(4), 155-161.

Munir, R. (2019). *Kriptografi, Edisi kedua.* Bandung: Informatika.

Nugroho, A. (2010). *Rekayasa Perangkat Lunak Berorientasi Objek dengan Metode USDP*. Yogyakarta: Andi.

Rahmat, B. (2010). Steganografi Menggunakan Metode Least Significant Bit dengan Kombinasi Algoritma Kriptografi Vigenere dan RC4. *Jurnal Dinamika Informatika, Vol 5 No 2*.

Saifudin, A., & Wahono, R. S. (2015). Penerapan Teknik Ensemble untuk Menangani Ketidakseimbangan Kelas pada Prediksi Cacat Software. *Journal of Software Engineering*, *1*(1), 28-37.

Yahya, A. (2019). *Steganography Techniques for Digital Images 1st editon.* London: Springer.