PENYEMBUNYIAN INFORMASI DENGAN METODE *CRYPTO-STEGANOGRAPHY* MENGGUNAKAN MEDIA GAMBAR BERBASIS MOBILE

Deden Pradeka[1]
Fakultas Teknologi dan Informatika, Universitas Informatika dan Bisnis Indonesia
dedenpradeka@unibi.ac.id

Abstrak

Steganografi adalah seni untuk menyembunyikan informasi atau data menggunakan media tertentu seperti gambar, suara, text, dan lainnya. Saat ini steganografi menjadi salah satu teknik yang kuat untuk menyembunyikan data pada sebuah media. Media yang populer digunakaan pada teknik steganografi adalah gambar, karena memiliki banyak informasi seperti nilai pixel, histogram, metadata, dan lainnya yang dapat dimanfaatkan untuk menyembunyikan sebuah data. Ada beberapa teknik steganografi dengan menggunakan media gambar, salah satunya adalah *Least Significant Bit* (LSB). LSB adalah teknik steganografi dengan memanfaatkan nilai pixel pada gambar, dimana setiap karakter pesan disisipkan pada nilai pixel, sehingga mengubah nilai original pixel. Dengan kata lain gambar pasti berubah setelah proses steganografi (*Noise Steganography*), dampaknya adalah menimbulkan kecurigaan bagi orang lain yang tidak berkepentingan. Pada penelitian ini mengusulkan teknik kombinasi kriptografi dan steganografi dengan menggunakan media gambar tanpa mengubah informasi pada gambar tersebut (*Noiseless Steganography*), sehingga mengurangi kecurigaan orang lain yang tidak berkepentingan.

Kata Kunci: *Steganography*, *Least Significant Bit*, *Noiseless, Noise.*

Abstract

Steganography is the art of data hiding for protecting information or data by certain media, the media can be use image, voice, text, and etcetera. Currently, steganography is one of the most powerfull technique to data hiding use a media. The media usually use image to technique steganography, because the image has lots of information like pixel value, histogram, metadata, and other can be utilized for embedding the data. There are several steganography technique by image, one of is Least Significant Bit (LSB). LSB is steganography technique by using pixel value, where each of message character embedded into pixel value, so change the original pixel. In other words, the image must change after the process of steganography (Noise Steganography), the effect is to arouse suspicion for other unauthorized people. This research proposes crypto-steganography techniques using image without changing the information in the cover (Noiseless Steganography), so can be reducing the suspicion of other unauthorized people.

Keywords: *Steganography*, *Least Significant Bit*, *Noiseless, Noise.*

1.PENDAHULUAN

1.1 Latar Belakang

Kriptografi adalah ilmu matematika untuk mengenkripsi dan dekripsi sebuah pesan atau data [4]. Kriptografi digunakan untuk melindungi informasi atau pesan yang sensitif, sehingga orang tertentu saja yang boleh mengakses informasi tersebut. Pesan atau data sedemikian rupa diubah dalam bentuk tertentu sehingga hanya pengirim dan penerima yang mengetahuinya.

Sementara kata "Steganografi" berasal dari bahasa Yunani, dimana *steganos* adalah tertutup atau rahasia, dan *graphy* adalah menulis atau menggambar, dengan demikian secara harfiah steganografi adalah tulisan rahasia atau tertutup. Steganografi merupakan teknik penyembunyian data yang bertujuan mentransmisikan pesan pada saluran tertentu [1].

Gambar digital kerap kali digunakan sebagai media untuk menyisipkan pesan pada teknik steganografi. Steganografi pada gambar adalah menyembunyikan pesan didalam gambar sedemikian rupa sehingga tidak memungkinkan "musuh" untuk mendeteksi bahwa ada pesan rahasia yang ada didalam gambar.

Ada beberapa teknik steganografi dengan menggunakan media gambar, salah satunya adalah Least Significant Bit (LSB). LSB adalah adalah teknik steganografi yang sering digunakan pada media gambar, teknik ini memanfaatkan spatial domain pada gambar dengan mengubah setiap karakter pesan dan pixel kedalam bentuk biner dan menyisipkan nilai biner pesan tersebut kedalam nilai biner pixel pada gambar [2]. Dengan kata lain nilai gambar pasti berubah setelah proses steganografi. Seperti contoh: huruf "a" sebagai karakter pesan dan 250, dan seterusnya adalah nilai pixel pada gambar. Huruf "a" diubah dalam desimal (ASCII) adalah 97, kemudian 97 diubah dalam bentuk biner (8 bit) adalah 01100001, dan 250 adalah 11111010. Setelah itu angka "1" pada bagian terakhir biner "a" disisipkan pada biner terakhir pada pixel 250, sehingga nilai biner berubah menjadi 11111011 (251). Kemudian lakukan hal yang serupa terhadap biner "a" selanjutnya pada biner pixel selanjutnya.

Jika melihat prosedur dan teknik LSB dalam menyisipkan data pada cover maka gambar setelah proses steganografi pasti berubah dan kapasitas gambar dalam menampung pesan sedikit.

Pada penelitian ini maka diusulkan teknik noiseless steganography dengan menggunakan media gambar, yaitu gambar hasil proses steganografi tidak ada perubahan, kemudian kapasitas gambar untuk menampung karakter pesan dapat meningkat.

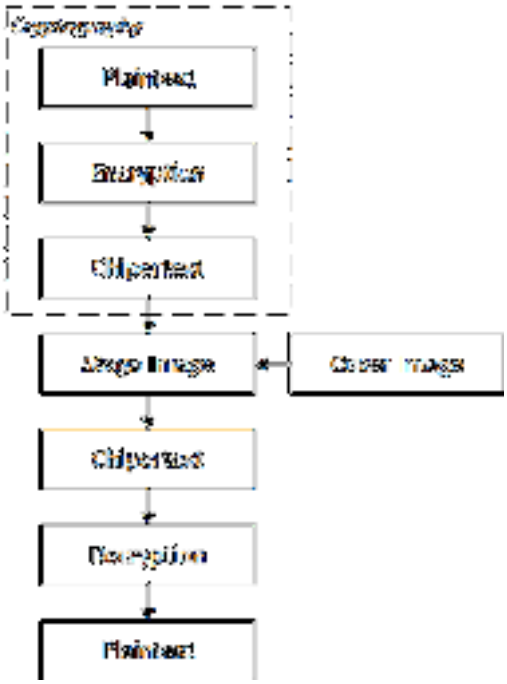
2. TINJAUAN PUSTAKA

2.1 Kombinasi Kriptografi dan Stegnografi

Kriptografi dan steganografi merupakan teknik yang berbeda. Kriptografi adalah teknik keamanan dengan megubah data menjadi bentuk kode tertentu sehingga hanya pengirim dan penerima yang bisa membukanya. Sementara Steganografi adalah teknik keamanan dengan menyisipkan pesan kedalam sebuah media atau *cover,* sehingga secara kasat mata orang lain tidak mencurigainya.

Kedua teknik tersebut memiliki kelemahan dan keunggulan, pada perkembangan teknologi keamanan komputer pada saat ini, banyak penelitian mengembangkan metode kombinasi antara kripto dan stego. Tujuannya adalah meningkatkan kapasitas dan keamanan data yang disematkan.

Pada gambar 1.1 menjelaskan proses kombinasi kripto dan stego.



Gambar 2.1 Kombinasi kriptografi dan steganografi [3].

Pada gambar 2.1 menjelaskan kombinasi kriptografi dan steganografi, dimana pesan atau

plaintext dienkripsi terlebih dahulu menggunakan teknik kriptografi, kemudian setelah pesan di enkripsi atau disebut dengan *chipertext* maka selanjutnya masuk keproses steganografi, dimana chipertext disisipkan kedalam cover gambar.

Dapat dilihat dari proses diatas maka pesan mengalami enkripsi sebanyak dua kali, yaitu menggunakan teknik kriptografi dan steganografi. Tujuannya adalah untuk meningkatkan keamanan ganda, sehingga orang yang tidak berkepentingan tidak bisa mendapatkan pesan original dengan mudah.

Teknik kombinasi antara kriptografi dan steganografi biasanya menggunakan *secrete key* dan *public key* untuk mengakses pesan dari pengirim ke penerima, jadi tidak cukup menggunakan media saja [3]. Ide ini menjadikan teknik keamanan data menjadi variatif dan baik.

2.2 Least Significant Bit (LSB)

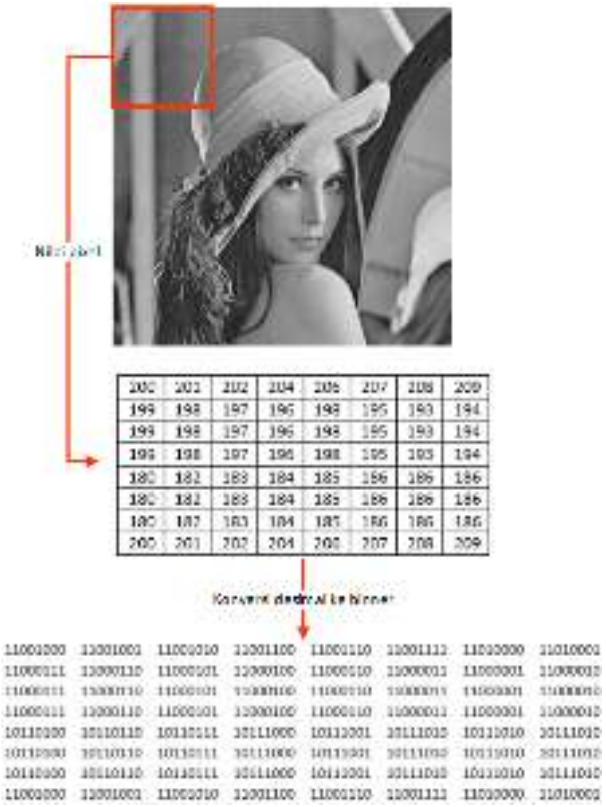
Least Significant Bit adalah teknik steganografi yang umum digunakan untuk menyisipkan data kedalam gambar. Pada dasarnya metode LSB bekerja dengan menukarkan sebuah karakter pada pesan yang disisipkan kedalam pixel gambar, sehingga sedemikan rupa pixel mengalami perubahan. Walaupun mengalami perubahan secara kasat mata orang lain tidak mengetahuinya, karena teknik ini memanfaatkan keterbatasan mata manusia yang tidak peka dengan perubahan kecil pada warna gambar. Tapi pada sebuah komputer perubahan gambar tersebut sangat mudah diketahui, dan sedikit banyak menimbulkan kecurigaan bagi orang yang tidak berkepentingan.

Metode LSB dapat bekerja dibentuk gambar seperti jpg, png, bmp, dan tiff baik menggunakan warna (3 dimensi) atau *greyscale* (2 dimansi). Pada gambar 2.2 menunjukkan proses encoder pada sisi pengirim.

Gambar 2.2 Proses enkripsi pada metode LSB

Dibawah ini dijelaskan langkah-langkah proses enkripsi dan dari LSB:

1. Diketahui pengirim menulis plaintext “MAJU” yang dikirim ke penerima.
2. Kemudian kata “MAJU” dikonversi dari ASCII menjadi bilangan binner (8 bit), maka didapat 01001101,01000001,01001010,01010101.
3. Pengirim menggunakan cover lenna.png 32x32 pixel, dimana setiap pixel gambar (8x8) dikonversi ke binner.



Gambar 2.3 Konversi pixel ke binner

4. Langkah selanjutnya adalah proses enkripsi atau menyisipkan setiap bit nilai biner chipertext (01001101, 01000001, 01001010, 01010101) ke nilai binner pada cover pada bagian akhir saja.

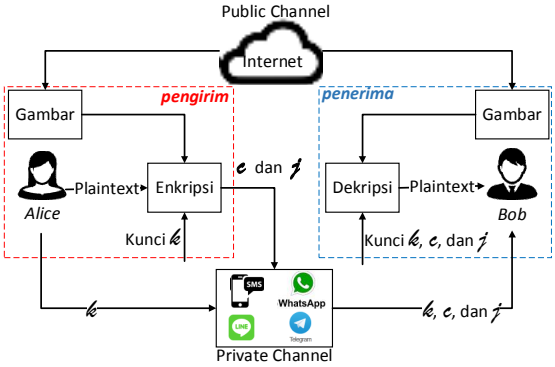
Tabel.2.1 Proses penyisipan dengan LSB

Binner pixel	110 010 00	110 010 01	110 010 10	110 011 00	110 011 10	110 011 11	110 100 00	110 100 01
Binner pesan	01001101							
Hasil penyisipan	110 010 00	110 010 01	110 010 10	110 011 00	110 011 11	110 011 11	110 100 00	110 100 01

5. Setelah semua bit sudah disisipkan, kemudian cover dikirim kepada penerima, dimana nilai pixel pasti berubah.

3. METODE PENELITIAN

Penelitian yang dilakukan adalah dengan mengkombinasi metode Caesar cipher dan algoritma modern Fisher-Yates. Algoritma Fisher-Yates digunakan untuk mengacak table hash berdasarkan kunci yang telah disepakati antara pengirim dan penerima.



Gambar 3.1 Desain Proses

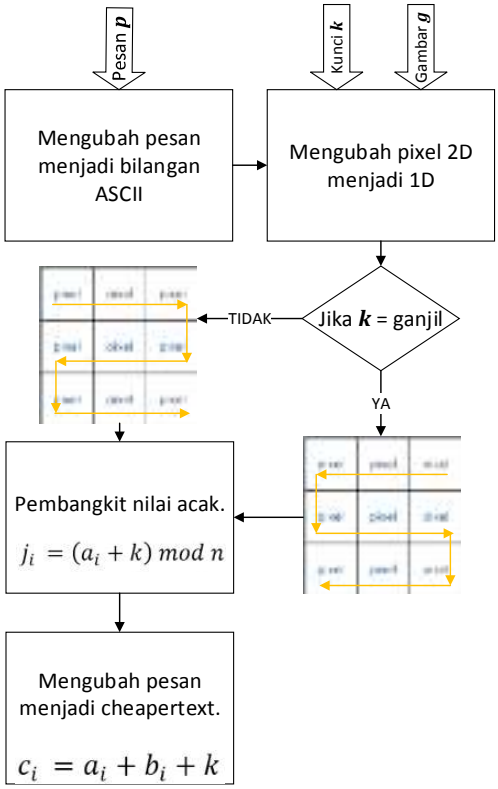
Pada gambar 3.1 adalah skenario dari data komunikasi, dimana Alice sebagai pengirim dan Bob sebagai penerima. Pada desain proses menggunakan dua channel untuk mengirimkan informasi, yang pertama private channel digunakan sebagai saluran pengiriman kunci *k, c*, dan *i* dari pengirim ke penerima. Pada jalur komunikasi private channel dapat menggunakan sms, whatsApp, line, telegram dan lainnya. Kedua adalah public channel, digunakan sebagai saluran untuk mengambil gambar (di internet) dimana sebelumnya sudah disepakati antara pengirim dan penerima gambar yang digunakan.

Skenarionya adalah Alice ingin mengirim plaintext "MAJU" kepada Bob, sebelumnya Alice dan Bob sudah sepakat menggunakan gambar lenna.png untuk proses komunikasi ini, gambar diambil dari internet. Kemudian Alice menggunakan kunci $k = 5$ yang didapat secara acak. Kemudian dari gambar dan kunci tersebut dilakukan proses enkripsi sehingga menghasilkan *chippertext* (*c*) dan kunci *i*.

Dari hasil tersebut Alice mengirim *k, c,* dan *i* kepada Bob melalui private channel, sehingga Bob dapat mendekripsi kembali dan mendapatkan pesan aslinya.

3.1 Enkripsi

Pada bagian 3.1 ini dijelaskan proses enkripsi dari metode yang diusulkan, dimana pesan diubah menjadi chippertext menggunakan crypto-steganography dan gambar sebagai media penutupnya,



Gambar 3.1 Desain Proses Enkripsi

Berikut ini adalah penjelasan dari langkah-langkah enkripsi yang dilakukan pada gambar 3.1:

1. Langkah pertama adalah pesan diubah dalam bentuk bilangan ASCII. Dalam contoh ini pesan yang dikirim adalah "MAJU", dari kata tersebut maka setelah diubah kedalam bentuk bilangan ASCII menjadi M=77, A=65, J=74, U=85. *a* = (77, 65, 74, 85).
2. Langkah kedua adalah mengubah gambar grayscale menjadi nilai pixel 2 dimensi, kemudian nilai pixel 2 dimensi diubah menjadi 1 dimensi dengan ketentuan sebagai berikut; jika nilai kunci *k* adalah ganjil maka urutan perubahan pixel dari kanan kekiri jika genap dari kiri kekanan. Kemudian nilai 1 dimensi disimpan dalam variabel *g* dengan tipe array. Contoh kunci kunci *k*

yang digunakan adalah 5 maka nilainya adalah ganjil. Selanjutnya terdapat sebuah pixel dengan nilai 3x3 seperti gambar 3.2. Maka variabel g = 20, 208, 105, 117, 50, 61, 32, 70, 254.

20	208	105
117	50	61
32	70	254

Gambar 3.2 Pixel

3. Langkah ketiga adalah memilih nilai pixel yang digunakan dengan cara diacak menggunakan persamaan 3.1, dimana j adalah variabel array, variabel a = 77, 65, 74, 85 didapat dari langkah pertama, kunci k = 5, dan variable n adalah banyaknya nilai dari dimensi yaitu 9. Contoh pada kasus ini adalah:

$j_1 = (77 + 5) \bmod 9 = 1$
$j_2 = (65 + 5) \bmod 9 = 7$
$j_3 = (74 + 5) \bmod 9 = 7$
$j_4 = (85 + 5) \bmod 9 = 0$

Maka dari perhitungan diatas didapat index yang digunakan dari nilai array variabel g = 20, 208, 105, 117, 50, 61, 32, 70, 254 adalah b = 208, 70, 70, dan 20.

$$j_i = (a_i + k) \bmod n \qquad (3.1)$$

4. Langkah keempat adalah mengubah pesan menjadi cheapertext dengan persamaan 3.2. dimana sudah diketahui nilai a = 77, 65, 74, 85, b = 208, 70, 70, 20, dan kunci k = 5. Contoh pada kasus ini adalah:

$c_1 = 77 + 208 + 5 = 290$
$c_2 = 65 + 70 + 5 = 140$
$c_3 = 74 + 70 + 5 = 149$
$c_4 = 85 + 20 + 5 = 110$

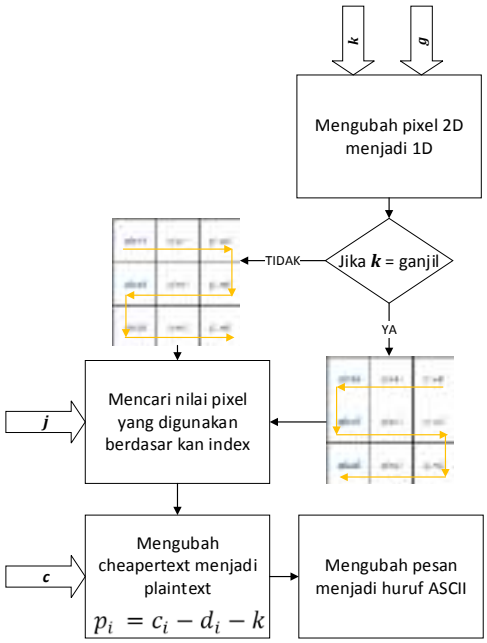
$$c_i = a_i + b_i + k \qquad (3.2)$$

Maka kunci yang dikirim melalui public chanel adalah g (gambar grayscale lena.jpg), sedangkan yang melalui private chanel adalah k = 5, c = 290, 140, 149, 110, dan j = 1, 7, 7, 0..

3.2 Dekripsi

Pada bagian ini dijelaskan proses dekripsi, yaitu proses pengembalian nilai di sisi penerima dari cheapertext ke plaintext atau pesan asli.

Pada skenario sebelumnya penerima sudah sepakat dengan pengirim menggunakan gambar yang ada di internet, sehingga pada proses dekripsi ini penerima bisa mendownload gambar tersebut. Kemudian pada jalur private penerima sudah mendapatkan kunci yang di janjikan oleh pengirim yaitu k = 5, c = 290, 140, 149, 110, dan j = 1, 7 ,7 , 0. Pada gambar 3.2 menjelaskan alur proses dekripsi pada metode yang diusulkan.



Gambar 3.2 Desain Proses Dekripsi

Dibawah ini dijelaskan langkah-langkah dekripsi:

1. Langkah pertama adalah mengubah gambar grayscale menjadi nilai pixel 2 dimensi, kemudian nilai pixel 2 dimensi diubah menjadi 1 dimensi dengan ketentuan seperti di langkah 2 proses enkripsi. Maka berdasarkan contoh kasus pada gambar 3.2 maka variabel g = 20, 208, 105, 117, 50, 61, 32, 70, 254.
2. Langkah kedua adalah mencari nilai pixel yang digunakan berdasarkan index yang sudah diketahui, variabel indexnya adalah j = 1, 7, 7, 0, maka d = 208, 70, 70, dan 20.

Variabel *d* bertipe array untuk menampung nilai pixel yang digunakan.

3. Langkah ketiga adalah mengubah variabel *c* = 290, 140, 149, 110 menjadi angka plaintext menggunakan persamaan 3.3.

$$p_i = c_i - d_i - k \qquad (3.3)$$

p_1 = 290 - 208 - 5 = 77
p_2 = 140 - 70 - 5 = 65
p_3 = 149 - 70 - 5 = 74
p_4 = 110 - 20 - 5 = 85

Berdasarkan contoh kasus yang dilakukan nilai *p* = 77, 65, 74, 85.

4. Langkah keempat adalah mengubah variabel *p* kebentuk huruf berdasarkan ASCII. Maka didapat 77=M, 65=A, 74=J, 85=U, pesan yang didapat adalah "MAJU".

4. HASIL DAN PEMBAHASAN

Berdasarkan hasil Penelitian yang dilakukan metode sebelumnya Least Significant Bit (LSB) lebih sedikit menampung karakter pesan dibandingkan metode yang diusulkan, karena satu karakter LSB membutuhkan pixel 8x8 untuk menyisipkan karakter tersebut. Tabel 4.1 menunjukan hasil perbandingan jumlah karakter yang disisipkan dalam sebuah gambar antara metode sebelumnya dan yang diusulkan.

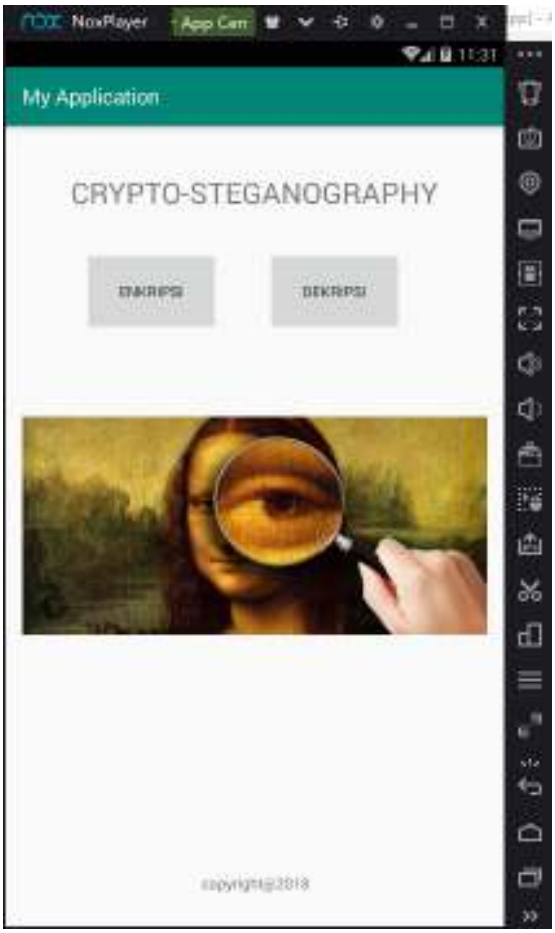
Tabel 4.1. Perbandingan kapasitas karakter yang dibisa ditampung

no	gambar	ukuran	jumlah karakter	
			sebelumnya	diusulkan
1		128x128	2048	>2048
2		32x32	128	>128
3		16x16	32	>32

Jika dilihat dari tabel 4.1 dapat diketahui LSB memiliki batasan karakter yang dapat disisipkan kedalam sebuah gambar, karena satu karakter membutuhkan 8 pixel untuk menyisipkan angka binner. Sedangkan metode yang diusulkan dapat menyisipka karakter sesuai dengan jumlah karakter pada pesan, karena 1 karakter dapat menggunakan pixel yang sama.

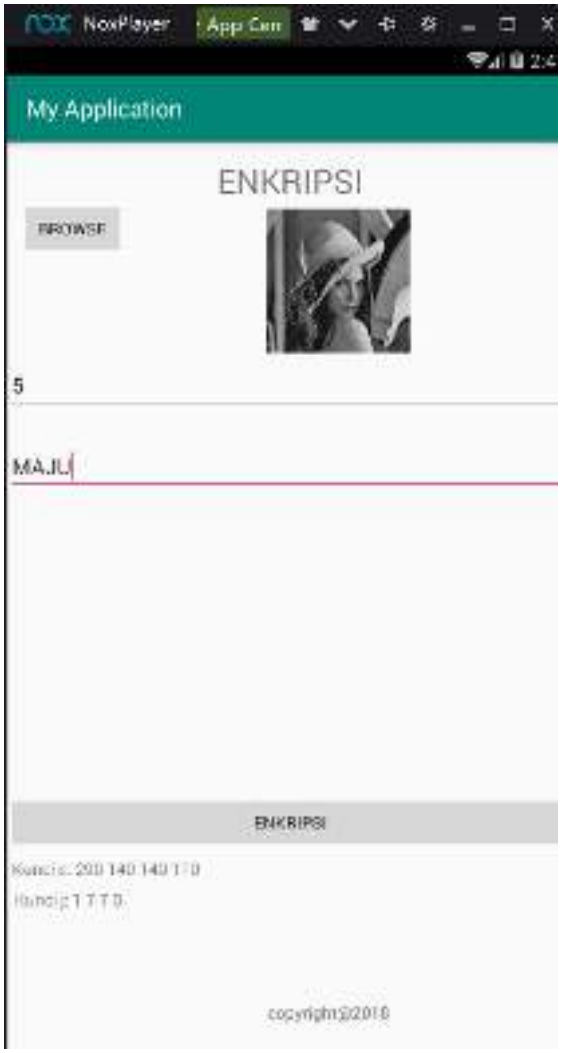
4.1 Tampilan Pengguna Mobile

Pada bagian ini dilaskan hasil tampilan enkripsi dan dekripsi pada Android menggunakan metode yang diusulkan. Pada gambar 4.1 menjelaskan tampilan awal.



Gambar 4.1 Tampilan Utama Aplikasi

Kemudian pada gambar 4.2 memperlihatkan tampilan untuk proses enkripsi untuk sisi pengirim.



Gambar 4.2 Tampilan Enkripsi

Kemudian pada gambar 4.3 menggambarkan tampilan dekripsi untuk sisi penerima.

Untuk membangun aplikasi ini, penulis menggunakan software Android Studio, dan Java. Kemudian hardware yang digunakan Personal komputer dengan spesifikasi; processor intel Core-i5-3470 dengan kecepatan 3.20 GHz, RAM 4.00 GB, dan Operasi sistemyang digunakan windows 10 (64-bit).

Gambar 4.3 Tampilan Dekripsi

5. KESIMPULAN

Berdasarkan hasil penelitian dan masalah yang diangkat dari metode sebelumnya, maka metode yang diusulkan dengan menggunakan kombinasi kriptografi dan steganografi maka gambar tidak mengalami perubahan pada pixel, dan jumlah karakter pesan dapat menampung lebih banyak dibandingkan metode sebelumnya.

Adapun demikian kelemahan yang terdapat pada metode yang diusulkan adalah besarnya ukuran kunci yang dikirim dari pengirim ke penerima.

6. REFERENSI

[1] S Gupta, G Gujral and N Anggarwal, *Enhanced Least Significant Bit algorithm for Image Steganography*, IJCEM, Vol.15 Issue 4, July 2012.

[2] Champakamala .B.S, Padmini.K, Radhika .D. K, *Least Significant Bit algorithm for image steganography*, IJACT,

[3] A Joseph dan V Sundaram, *Cryptography and Steganography – A Survey*. IJCTA, 2(3):626-630, February 2011.

[4] University of Michigan, *An Introduction to Cryptography,* 1990-1999

[5] S Gupta, A Goyal, B Bhushan, *Information Hiding Using Least Significant Bit Steganography and Cryptography*, I.J. Modern Education and Computer Science, 27-34, 2012

[6] Prof.S.V.Kamble, Prof. B.G.Warvante, *A Review on Novel Image Steganography Techniques*, IOSR-JCE,