Implementasi Metode Least Significant Bit (LSB) Dengan Enkripsi Cipher Caesar Pada Steganografi Menggunakan Image Processing

Bakir[1)], Hozairi[2)]

[1,2)] Program Studi Sistem Informasi, Universitas Islam Madura
Email : [1)] bakir.madura@gmail.com, [2)] dr.hozairi@gmail.com

ABSTRAK

Pesan digital menggunakan layanan jaringan internet dapat menimbulkan bahaya dan berpotensi dimanfaatkan oleh pihak lain yang tidak bertanggungjawab. Hal ini dapat memicu pihak lain dengan mudah dapat mengambil, mendistribusikan dan mengubah atau memodifikasi isi dari pesan digital tersebut. Kondisi ini berbahaya jika pesan yang dikirim merupakan pesan yang sangat penting dan rahasia sehingga pengamanan pesan dengan berbagai metode perlu dilakukan agar pesan digital dapat terjaga keaslian dan keamanannya. Metode LSB (*Least Significant Bit*) dikombinasikan dengan algoritma *Cipher Caesar* dapat mempengaruhi pengolahan citra digital (*image processing*) dalam implementasi sistemnya sehingga dapat dimanfaatkan untuk pengamanan pesan digital. Hasil dari penelitian ini menunjukkan bahwa analisa pertama dengan objek sama dan pesan sama, analisa kedua dengan objek sama dan pesan beda, analisa ketiga dengan objek beda dan pesan sama, dan analisa keempat dengan objek beda dan pesan beda. Dari hasil analisa yang dilakukan telah terjadi perubahan beberapa nilai rata-rata yaitu warna R (*red*) pada piksel citra digital tersebut, tetapi citra yang disisipkan beberapa pesan tidak terlihat secara kasat mata, sehingga pesan teks yang telah disisipkan sulit dideteksi. Kombinasi metode *Cipher Caesar* dan *Least Significant Bit* telah berhasil diimplementasikan untuk tujuan Steganografi sebagai proses keamanan data.

Kata Kunci : *Cipher Caesar*, *Least Significant Bit* (LSB), Steganografi

1. PENDAHULUAN

Seiring berkembangnya teknologi informasi saat ini, manusia dapat berkomunikasi melalui berbagai media informasi digital. Kerahasiaan transmisi data sebagai media informasi menjadi kebutuhan untuk mengirim informasi dan menjadi hal yang sangat penting dalam kehidupan di era digital. Sebagai salah satu contoh, media internet sebagai media penghubung terluas dan sebagian besar komponen elektronik di dunia membutuhkan jaringan internet, sehingga semua orang dengan mudah bisa saling mengirim informasi melalui media layanan internet. Layanan internet memiliki banyak kelebihan dibandingkan dengan media layanan komunikasi lainnya, diantaranya adalah kecepatan (*speed*). Akan tetapi informasi yang dikirim melalui internet, tidak dapat di jamin keamanannya karena penyadapan terhadap informasi rahasia sering terjadi pada komunikasi layanan internet. Sehingga diperlukan usaha untuk menangani masalah keamanan informasi rahasia yang dikirimkan melalui layanan internet. Diantaranya menggunakan teknik kriptografi (*cryptography*). Dengan teknik kriptografi ini pesan asli (*plainteks*) yang akan dikirimkan diubah atau dienkripsi dengan suatu kunci (*key*) menjadi suatu informasi acak (*ciphertext*) yang tidak bermakna. Kunci tersebut hanya diketahui oleh pengirim dan penerima. Selanjutnya,

penerima dapat menggunakan kunci untuk mengembalikan *ciphertext* ke bentuk *plaintext* sehingga orang lain yang tidak berhak tidak dapat mengetahui isi pesan tersebut, tetapi hanya mengetahui pesan acaknya. Karena informasinya acak, maka menimbulkan kecurigaan terhadap pesan yang telah dikirim. Untuk mengatasi hal tersebut dapat digunakan teknik lainnya yaitu teknik steganografi (*steganography*).

Steganografi merupakan teknik yang dapat mengijinkan semua pengguna untuk menyembunyikan (*embedding*) suatu pesan ke dalam objek lain (Supratman et al., 2015). Contoh penerapan steganografi adalah apabila terdapat gambar yang disisipkan suatu pesan rahasia, tetapi pesan tersebut tidak terlihat pada gambar yang telah disisipkan pesan tersebut. Sedangkan bila diekstrak dengan perangkat lunak steganografi maka akan terlihat pada gambar tersebut terdapat pesan rahasia.

Pengolahan citra merupakan suatu proses yang dapat dilakukan terhadap suatu gambar (*image*) sehingga dapat menghasilkan gambar lainnya yang lebih sesuai, sedangkan pemrosesan gambar berdimensi dua dapat dilakukan melalui komputer digital (Shpakov & Bogomolov, 2011). Sumber lain, pengolahan citra digital merupakan istilah yang umum untuk berbagai macam teknik untuk memanipulasi data dan memodifikasi data citra digital dengan berbagai macam cara pengelolaan sehingga menghasilkan data yang dimanfaatkan (Ari Anti, Harsa Kridalaksana, & Marisa Khairina, 2017). Sedangkan gambar yang dihasilkan dari kamera digital merupakan contoh *image* berdimensi dua yang bisa diolah dengan mudah. Sehingga pengolahan citra digital merupakan suatu proses memodifikasi, memanipulasi dan menganalisis citra digital yang berupa gambar dengan bantuan komputer. Kehadiran teknologi akan memberikan kemajuan yang luar biasa (Caesar, 2016).

Penelitian ini diharapkan dapat bermanfaat untuk membantu pengguna layanan internet yang akan mengirim dan menerima informasi atau pesan secara rahasia, tanpa diketahui orang lain atau pihak lain yang tidak bertanggung jawab.

2. TINJAUAN PUSTAKA

2.1. Pengolahan Citra

Pengolahan citra merupakan suatu proses yang dilakukan terhadap suatu gambar (termasuk gambar dua dimensi) sehingga menghasilkan gambar lain yang lebih sesuai. Pengolahan citra digital meliputi beberapa tahapan yaitu :

1. Akusisi citra
2. Peningkatan kualitas citra
3. Segmentasi citra
4. Representasi dan uraian
5. Pengenalan dan interpretasi

2.2. Least Significant Bit

LSB (*Least Significant Bit*) adalah bagian data dari sebagian data biner yang mempunyai nilai paling kecil dan letak posisinya pada barisan bit paling kanan. (Zainal, Pagar, & Bandarlampung, 2016), (Julianto & Bendi, 2016). Pada berkas file gambar yang berektensi *bitmap* 24-bit dan setiap *pixel* tersusun dari tiga warna merah, hijau dan biru (RGB). Masing-masing susunan terdapat bilangan 8 bit yang dimulai dari 0 sampai 255 yaitu menggunakan bilangan biner 00000000 sampai 11111111. Setiap *pixel* pada berkas gambar *bitmap* 24 bit dapat kita lakukan penyisipan 3 bit data.

2.3. Steganografi

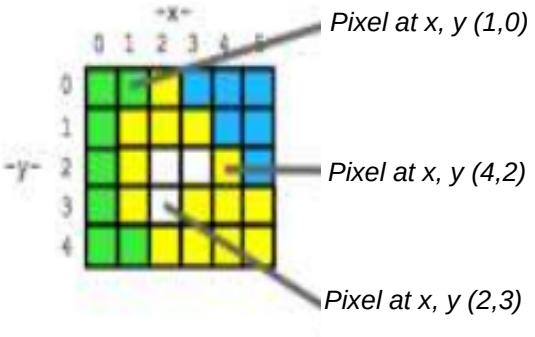
Metode Steganografi merupakan teknik untuk menyembunyikan data rahasia dan dapat menghasilkan data yang akan menampilkan data semula.

p-ISSN : 2502-5724; e-ISSN : 2541-5735

Metode ini merupakan teknik penyamaran data menggunakan media lain sebagai media penyamaran informasi rahasia tidak terlihat secara jelas dan secara acak (Supratman et al., 2015). Sebaliknya, kriptografi menyamarkan makna dari suatu pesan digital, akan tetapi tidak menyembunyikan data bahwa data tersebut ada pesan yang disamarkan.

Kata steganografi dari bahasa Yunani yaitu “steganos”, yang mempunyai arti tersembunyi atau terselubung, dan “graphein”, yang artinya menulis (Galih Fathul Rohmi, 2016). Metode steganografi menggunakan data *bit-wise* yang berfungsi menyisipkan *bit* dan *noise*. Format gambar yang paling cocok untuk cara ini adalah tipe *loss less*. Namun, cara ini sangat bergantung kepada format gambarnya (Wijaya & Prayudi, 2004).

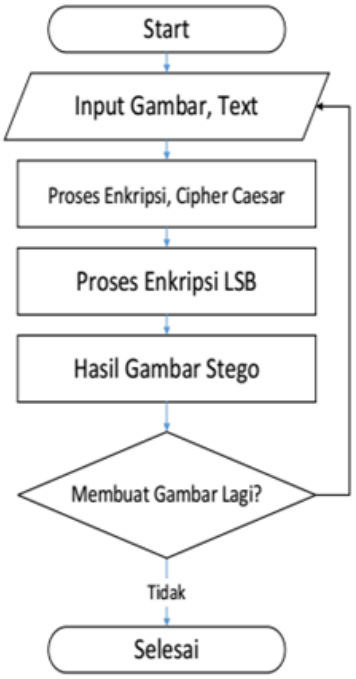
Pengolahan citra merupakan fungsi dua dimensi dapat dinyatakan sebagai fungsi kontinyu dari data intensitas cahaya dengan dua dimensi, f (x,y). Dimana x dan y menyatakan kordinat ruang, dan nilai f pada suatu koordinat (x,y) menyatakan kecerahan dan informasi warna citra. Secara matematis fungsi intensitas, f (x,y) adalah sebagai berikut (Krisnawati, 2008).



Gambar 1. Image Element

3. METODE PENELITIAN

Proses perancangan algoritma untuk steganografi dengan metode *Least Significant Bit* (LSB) mengikuti alur pada Gambar 2.



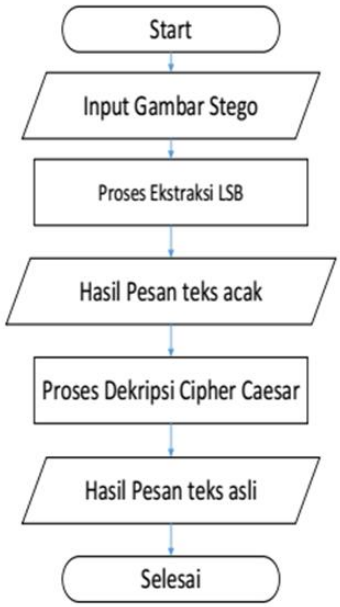
Gambar 2. Alur Image Processing

Sistem kerja pada *image processing* ini dapat dijelaskan berdasarkan Gambar 2 berikut ini.

1. User akan mengambil atau memasukkan (*input*) gambar yang akan dijadikan media untuk menyisipkan pesan.
2. User harus membuat teks yang akan disisipkan dalam gambar yang sebelumnya diinputkan.
3. User akan mengenkripsi pesan tersebut dengan metode *Chiper Caesar*.
4. User akan menyisipkan pesan teks yang sudah di enkripsi ke dalam gambar dengan metode *Least Significant Bit* (LSB).
5. Aplikasi menampilkan gambar yang telah disisipkan pesan teks.

Gambar hasil penyisipan pesan teks diatas, selanjutnya disebut dengan gambar stego (*Steganography*).

Gambar 3 menunjukkan *flowchart* untuk mengembalikan pesan teks yang disisipkan, sehingga menghasilkan pesan teks dari gambar.



Gambar 3. Hasil Pengolahan Data

Sistem kerja pengolahan data pada proses enkripsi yaitu :

1. User akan mengambil atau memasukkan (*input*) gambar steganografi.
2. Proses ekstraksi dengan metode *Least Significant Bit* (LSB)
3. Proses dekripsi *Chiper Caesar*.

Selanjutnya setelah proses dekripsi selesai, maka aplikasi akan menampilkan pesan teks yang disisipkan pada gambar steganografi dengan proses enkripsi.

4. HASIL DAN PEMBAHASAN

Pengujian data yang digunakan untuk menganalisa sebanyak empat skenario. Pertama, objek sama dan pesan sama. Kedua, objek sama dan pesan beda. Ketiga, objek beda dan pesan sama. Keempat, objek beda dan pesan beda. Dari empat hasil analisa yang telah dilakukan terjadi perubahan nilai rata-rata warna R (*red*) pada piksel citra tersebut. Berbeda dengan keadaan citra yang telah disisipi pesan tidak terlihat secara kasat mata, sehingga pesan teks yang telah disisipkan sulit dideteksi. Hasil penelitian ini sangat bermanfaat bagi para pengguna yang ingin mengirimkan pesan digital agar pesan yang akan disampaikan tetap aman dan terjaga keasliannya.

Simulasi analisa juga dilakukan untuk menguji aplikasi steganografi dengan metode *Least Significant Bit* (LSB) ditunjukkan dalam Gambar 4.

Pada Gambar 4 terdapat enam fungsi tombol yang digunakan dalam proses pengolahan data yaitu :

1. Tombol ***Buka Gambar***, berfungsi untuk mengambil gambar dan menampilkannya pada panel Gambar Asli. Jjumlah komposisi warna *red* (R) akan tampil di bagian kanan aplikasi.
2. Tombol ***Tampilkan Pesan***, berfungsi untuk mengambil teks dan menampilkannya pada list pesan 1.
3. Tombol ***Encrypt***, berfungsi untuk mengacak pesan asli dengan Kunci Chiper dan menampilkan pesan acak pada list pesan 2.
4. Tombol ***Proses STG***, berfungsi untuk menyisipkan teks yang telah di acak terhadap citra. Secara otomatis jumlah komposisi warna red (R) akan tampil di bagian kanan aplikasi.

p-ISSN : 2502-5724; e-ISSN : 2541-5735

5. Tombol ***Open Stego***, berfungsi untuk membuka gambar stego (citra yang sudah tersisipi pesan).
6. Tombol ***Ekstrak***, berfungsi untuk memisahkan pesan acak dari citra, dan pesan tersebut akan tampil pada list pesan 3.
7. Tombol ***Decrypt***, berfungsi untuk mengembalikan pesan acak ke pesan asli dan akan tampil pada list pesan 4.
8. Tombol ***Reset***, berfungsi untuk mengembalikan program ke status awal jika terjadi kesalahan prosedur penggunaan atau ingin melakukan pengujian berikutnya.
9. Tombol ***Keluar***, berfungsi untuk mengakhiri program.

Analisa ini bertujuan untuk menguji aplikasi, apakah dapat berjalan dengan baik. Hasil pengujian ditunjukkan dalam Tabel 1, Tabel 2, Tabel 3, dan Tabel 4.

Gambar 4. Proses *Image Processing*

Tabel 1. Objek Sama dan Pesan Sama

No	File Citra	Size (byte)	Screet File	Size Pesan (byte)	Output	Size (byte)	Karakter input dan output	Nilai Rata Warna Sebelum dan Sesudah
1	main.bmp	480,05	Pesan 3	68	Hasil 1	480,05	68/68	31457690 / 31457687
2	main.bmp	480,05	Pesan 3	68	Hasil 2	480,05	68/68	31457690 / 31457657
3	main.bmp	480,05	Pesan 3	68	Hasil 3	480,05	68/68	31457690 / 31457687
4	main.bmp	480,05	Pesan 3	68	Hasil 4	480,05	68/68	31457690 / 31457657
5	main.bmp	480,05	Pesan 3	68	Hasil 5	480,05	68/68	31457690 / 31457677

Tabel 2. Objek Sama dan Pesan Berbeda

No	File Citra	Size (byte)	Screet File	Size Pesan (byte)	Output	Size (byte)	Karakter input dan output	Nilai Rata Warna Sebelum dan Sesudah
1	main.bmp	66,806	Pesan 1	2	Hasil 1	66,806	2/2	2835907/ 2835914
2	main.bmp	66,806	Pesan 2	32	Hasil 2	66,806	32/32	2835907/ 2836033
3	main.bmp	66,806	Pesan 3	42	Hasil 3	66,806	42/42	2835907/ 2836068
4	main.bmp	66,806	Pesan 4	63	Hasil 4	66,806	63/63	2835907/ 2836145
5	main.bmp	66,806	Pesan 5	83	Hasil 5	66,806	83/83	2835907/ 2836232

Tabel 3. Objek Berbeda & Pesan Sama

No	File Citra	Size (byte)	Screet File	Size Pesan (byte)	Output	Size (byte)	Karakter input dan output	Nilai Rata Warna Sebelum dan Sesudah
1	main.bmp	67,854	Pesan 1	189	Hasil 1	67,854	189/189	4066717/ 4046823
2	main.bmp	188,054	Pesan 2	189	Hasil 2	188,054	189/189	8599094/ 8599735
3	main.bmp	368,254	Pesan 3	189	Hasil 3	368,254	189/189	14935652/ 14935742
4	main.bmp	608,454	Pesan 4	189	Hasil 4	608,454	189/189	38784330/ 38784449
5	main.bmp	750,054	Pesan 5	189	Hasil 5	750,054	189/189	52988190/ 52988677

Tabel 4. Objek Berbeda & Pesan Beda

No	File Citra	Size (byte)	Screet File	Size Pesan (byte)	Output	Size (byte)	Karakter input dan output	Nilai Rata Warna Sebelum dan Sesudah
1	main.bmp	2.430.056	Pesan 1	214	Hasil 1	2.430.056	210/210	136858978/ 136859074
2	main.bmp	3.000.054	Pesan 2	255	Hasil 2	3.000.054	249/249	160208248/ 160208390
3	main.bmp	6.750.054	Pesan 3	426	Hasil 3	6.750.054	417/208	357662711/ 357662921
4	main.bmp	9.720.054	Pesan 4	461	Hasil 4	9.720.054	450/225	444737907/ 444738173
5	main.bmp	12.000.054	Pesan 5	514	Hasil 5	12.000.054	500/125	557818116/ 557818263

Dari hasil pengujian aplikasi dapat dilihat bahwa pesan maksimal yang dapat di ekstrak adalah 249 karakter. Jika pesan lebih maka proses ekstrak tidak berjalan dengan baik, walaupun dengan ukuran objek yang besar.

p-ISSN : 2502-5724; e-ISSN : 2541-5735

5. KESIMPULAN DAN SARAN

Berdasarkan hasil analisa dan pengujian sistem dapat diambil empat kesimpulan sebagai berikut.

1. Dari analisa penyisipan pesan yang telah dilakukan, terjadi perubahan nilai rata-rata R pada pixel. Pertama, nilai R sebelum disisipi pesan 160208248, setelah disisipi pesan menjadi 160208370. Kedua, nilai R sebelum disisipi pesan 38784330, setelah disisipi pesan menjadi 38784449. Ketiga, nilai R sebelum disisipi pesan 136858978, setelah disisipi pesan menjadi 136859074. Perubahan nilai rata-rata R ini membuktikan bahwa pesan telah berhasil disisipkan.
2. Citra yang bisa dijadikan objek stego hanya citra yang berekstensi .bmp (*bitmap*).
3. Pesan yang dapat diekstrak maksimal 249 karakter, apabila lebih maka aplikasi tidak dapat mengektrak dengan sempurna.
4. Kombinasi metode (LSB) dan (CC) mampu menjaga keamanan pesan, karena pesan tidak merubah keadaan citra.

DAFTAR PUSTAKA

Ari Anti, U., Harsa Kridalaksana, A., & Marisa Khairina, D. (2017). Steganografi Pada Video Menggunakan Metode Least Significant Bit (LSB) Dan End of File (EOF). Jurnal Informatika Mulawarman, 12(2), 104–111.

Caesar, A. (2016). Perancangan Aplikasi Kriptografi Berlapis Menggunakan. In Seminar Nasional Teknologi Informasi Dan Multimedia 2016 (Vol. 6, pp. 6–7).

Galih Fathul Rohmi, E. I. (2016). Implementasi Algoritma Chiper Caesar Untuk Enkripsi dan Dekripsi Pada Tabel Ascii Menggunakan Bahasa Java, (May).

Julianto, Y., & Bendi, K. J. (2016). Sistem Steganografi Dengan Metode Least Significant Bit (Lsb) Teracak. Flash, 2(2), 116–123.

Krisnawati. (2008). Metode Least Significant Bit (Lsb) Dan End of File (Eof). In Seminar Nasional Informatika (semnasIF) (Vol. 24 Mei 200, pp. 39–44).

Shpakov, O. N., & Bogomolov, G. V. (2011). Pengolahan Citra Digital Untuk Mendeteksi Obyek Menggunakan Pengolahan Warna Model Normalisasi RGB. In Semantik 2011 (Vol. 1, pp. 329–332). https://doi.org/10.1016/S0166-1116(08)71924-1

Supratman, S. G., Studi, P., Ilmu, M., Luhur, U. B., Utara, P., & Selatan, J. (2015). Steganografi dengan menggunakan metode lsb dan algoritma hill cipher. Jurnal Buffer Informatika, 1(1), 38–45.

Wijaya, E. S., & Prayudi, Y. (2004). Konsep Hidden Message Menggunakan Teknik Steganografi Dynamic Cell Spreading. Media Informatika, 2(1), 23–38. https://doi.org/10.20885/informatika.vol2.iss1.art3

Zainal, J., Pagar, A., & Bandarlampung, N. K. (2016). Implementasi Teknik Steganografi Least Significant Bit (Lsb) Dan Kompresi Untuk Pengamanan Data Pengiriman Surat Elektronik. Teknoinfo, 10(2), 1–7.