ISSN PRINT : 2303-500X
ISSN ONLINE : 2477-3786

Aplikasi Pengamanan Informasi Menggunakan Metode *Least Significant Bit* (Lsb) dan Algoritma Kriptografi *Advanced Encryption Standard* (AES)

Dian Novianto[1)] , Yohanes Setiawan[2)]

[1)2)]Program studi Teknik Informatika, STMIK Atma Luhur
Jl. Jendral Sudirman, Selindung Baru, Kec. Gabek Pangkal pinang Kode Pos 33117
Email : diannovianto@atmaluhur.ac.id[1)], ysetiawanj@atmaluhur.ac.id[2)]

Abstract

There are several security aspects of the data that must be maintained, such as: authentication, integrity, non repudiation, authority, confidentialty, privacy and access control. One of the vulnerable parts of data security is when sending data to the destination. At the time of delivery, tapping of data can occur, so that people who are not entitled to get that information can find out. Therefore, an information needs to be secured so that only people who have access rights can know or get that information. And to maintain the confidentiality of the information, one of the ways is to insert the data into other objects, so that other people do not realize if the object contains important data or information. This hiding method is also known as steganography, and cryptography is added to strengthen the security of the data, that is a science and art to maintain the confidentiality of the message by encoding it in a form that is incomprehensible. The method used in steganography is Least Significant Bit (LSB), the algorithm used in cryptography is the Advanced Encryption Standard (AES), and the software development method used is prototype. The results of this study are all aspects of data security can be achieved, including when passing through the process of sending data through media such as the internet.

Keyword: *Steganography, Criptography, Prototype.*

Abstrak

Ada beberapa aspek keamanan data yang harus dijaga, seperti: otentikasi, integritas, non repudiation, otoritas, kerahasiaan, privasi, dan kontrol akses. Salah satu bagian yang rentan dari keamanan data adalah ketika mengirim data ke tujuan. Pada saat pengiriman, penyadapan data dapat terjadi, sehingga orang yang tidak berhak mendapatkan informasi tersebut dapat mengetahuinya. Oleh karena itu, suatu informasi perlu diamankan sehingga hanya orang yang memiliki hak akses yang dapat mengetahui atau mendapatkan informasi tersebut. Dan untuk menjaga kerahasiaan informasi, salah satu caranya adalah dengan memasukkan data ke objek lain, sehingga orang lain tidak menyadari jika objek tersebut berisi data atau informasi penting. Metode persembunyian ini juga dikenal sebagai steganografi, dan kriptografi ditambahkan untuk memperkuat keamanan data, yaitu ilmu dan seni untuk menjaga kerahasiaan pesan dengan menyandikannya dalam bentuk yang tidak dapat dipahami. Metode yang digunakan dalam steganografi adalah Least Significant Bit (LSB), algoritma yang digunakan dalam kriptografi adalah Advanced Encryption Standard (AES), dan metode pengembangan perangkat lunak yang digunakan adalah prototipe. Hasil dari penelitian ini adalah semua aspek keamanan data dapat dicapai, termasuk ketika melewati proses pengiriman data melalui media seperti internet.

Kata kunci: *Steganografi, kriptografi, Prototipe*

1. Pendahuluan

Pada era digital saat ini pengiriman data melalui media internet merupakan hal yang biasa, dimana masyarakat sudah memiliki perangkat dan akses terhadap jaringan internet yang cukup memadai. Dalam penyampaian informasi menggunakan media internet membutuhkan suatu tingkat keamanan data, agar data tidak dapat diakses oleh orang yang tidak memiliki izin sehingga kerahasiaannya dapat terjaga. Ada beberapa ancaman-ancaman yang dapat terjadi melalui jaringan internet yang harus di waspadai, hal tersebut bisa berupa interupsi, penyadapan, modifikasi maupun fabrikasi.

Oleh karena itu untuk menjamin kerahasiaan dari informasi yang akan dikirimkan melalui media seperti internet, informasi harus diletakan pada wadah pembawa yang dapat memperkuat tingkat keamanan dari informasi. Wadah dari informasi ini merupakan sebuah media yang dapat berbentuk berbagai jenis berkas, seperti berkas gambar, berkas suara, maupun berkas dokumen. Sehingga informasi tidak akan dikenali secara langsung oleh orang yang berniat untuk mendapatkan informasi tersebut tanpa hak akses.

Proses menyembunyikan informasi kedalam sebuah wadah pembawa dapat dilakukan dengan menggunakan berbagai metode dan variasi. Salah satunya adalah dengan menggunakan teknik steganografi, selain itu juga dapat ditambahkan teknik untuk dilakukan pengacakan informasi dengan kunci tertentu sehingga informasi tidak diketahui artinya, teknik ini juga dikenal dengan nama kriptografi. Untuk menyelesaikan teknik steganografi ini dapat menggunakan beberapa metode, antara lain: *Least Significant Bit* (LSB), *End Of File (EOF), Discrete Cosine Transform (DCT), Discrete Wavelet Transform (DWT)* dan lain – lain, sedangkan algoritma kriptografi sendiri terdiri dari 2 jenis, yaitu asmetri kriptografi dan simetri kriptografi, perbedaan dari keduanya adalah dari kunci yang digunakan. Untuk algoritma kriptografi asimetri sering juga disebut dengan algoritma kunci publik, yaitu kunci untuk melakukan enkripsi dan dekripsi berbeda, menggunakan kunci public dan kunci privat, seperti: *Digital Signature Algorithm* (DSA), *Rivest Shamir Adleman* (RSA), *Diffie Hellman* (DH), *Quantum*, dan lain sebaganya. Sedangkan algoritma kriptografi simetri sering disebut dengan algoritma klasik karena menggunakan kunci yang sama untuk melakukan proses enkripsi dan dekripsi, seperti: *Data Encription* Standard (DES), *Advanced Encryption Standard* (AES), *International Data Encryption Algorithm* (IDEA) *One Time Pad* (OTP) dan lain sebagainya[1].

Metode yang digunakan dalam steganografi pada penelitian ini adalah *Least Significant Bit* (LSB) dan algoritma kriptografi yang digunakan adalah *simetri kriptografi yaitu Advanced Encryption Standard* (AES).
Metode LSB (*Least Significant Bit*) merupakan salah satu teknik substitusi pada steganografi. Dimana tiap bit terendah pada *byte-byte* media citra akan digantikan dengan bit-bit pesan yang akan disisipkan. Pada file citra 24 bit setiap *pixel* pada citra terdiri dari susunan tiga warna, yaitu merah, hijau dan biru (RGB) yang masing-masing disusun oleh bilangan 8 bit (1 byte) dari 0 sampai 255 atau dengan format biner 00000000 sampai 11111111[2].

Metode LSB merupakan metode steganografi yang paling sederhana dan mudah diimplementasikan. Metode ini menggunakan citra digital sebagai covertext. Pada susunan bit di dalam sebuah *byte* (1 *byte* = 8 bit), ada bit yang paling depan (*most significant bit* atau MSB) dan bit yang paling akhir (*least significant bit* atau LSB)[3].
Advanced Encryption Standard (AES) merupakan algoritma *cryptographic* yang dapat digunakan untuk mengamankan data. Algoritma AES adalah blok *chipertext* simetrik yang dapat mengenkripsi (*encipher*) dan dekripsi (*decipher*) informasi. Enkripsi merubah data yang tidak dapat lagi dibaca disebut *ciphertext*, sebaliknya dekripsi adalah merubah *ciphertext* data menjadi bentuk semula yang kita kenal sebagai *plaintext*[4].

AES ini merupakan algoritma block cipher dengan menggunakan sistem permutasi dan substitusi (*P-Box* dan *S-Box*) bukan dengan jaringan Feistel sebagaiman block cipher pada umumnya. Jenis AES terbagi 3, yaitu AES-128, AES-192, AES-256. Pengelompokkan jenis AES ini adalah berdasarkan panjang kunci yang digunakan[5].

Diperlukan pengujian dalam penelitian ini untuk mengetahui cara kerja dari teknik steganografi yang menggunakan metode LSB maupun kriptografi dengan algoritma AES, seperti proses *encoding* dari gambar yang kemudian akan disisipkan pesan rahasia yang telah terenkripsi, serta proses *decoding* gambar untuk mengeluarkan *file* rahasia yang terenkripsi untuk kemudian dilakukan proses dekripsi pada *file* rahasia nya.

Model pengembangan perangkat lunak yang digunakan dalam penelitian ini adalah *prototype*. Model *prototype* merupakan sesuatu yang harus dievaluasi dan di modifikasi kembali, segala perubahan dapat terjadi pada saat *prototype* dibuat untuk memenuhi kebutuhan pengguna dan pada saat yang sama memungkinkan pengembang untuk lebih memahami kebutuhan pengguna secara lebih baik. Model *prototype* dimulai dengan pengumpulan kebutuhan pengguna[6].

Dalam penelitian ini *tools* pengembangan yang digunakan adalah UML. UML atau *Unified Modeling Language* merupakan perangkat lunak yang berparadigma "berorientasi objek". Pemodelan (*modeling*) sesungguhnya digunakan untuk penyederhanaan permasalahan-permasalahan yang kompleks sedemikian rupa sehingga lebih mudah dipelajari dan dipahami [7].

A. Metode Penelitian

Metode yang digunakan dalam penelitian ini menggunakan metode kualitatif dimana peneliti menjadi alat utama dalam pengumpulan data[8]. Pengumpulan data yang peneliti lakukan dengan cara mencari referensi

yang terkait dengan topic penelitian dari jurnal maupun buku. Dengan cara tersebut peneliti dapat memahami cara kerja dari metode dan algoritma yang dipakai, sehingga diharapkan dalam pengembangan perangkat lunak nantinya akan berjalan dengan baik dan lancar.
Dalam pengembangan sistem dengan model *Prototype*, model ini memiliki beberapa tahapan yang harus diikuti oleh peneliti, antara lain: Pengumpulan kebutuhan, membangun *prototyping*, evaluasi *prototyping*, mengkodekan sistem, menguji sistem dan evaluasi sistem.

Dan dalam pengembangan sistem, spesifikasi kebutuhan perangkat keras dan perangkat lunak yang peneliti gunakan dalam penelitian ini, seperti pada tabel 1 dibawah ini:

Tabel 1 Perangkat yang digunakan

Perangkat Lunak	Perangkat Keras
Windows 8.1 Pro	Laptop dengan Spesifikasi: Processor A8-4500M up to 2.80 GHz, GPU: AMD HD8750 2GB Vram, RAM 4GB, HDD 500GB
Visual Basic 6.0	
Cryptool 2	
Astah Community 7.1.0	

Tahapan yang peneliti lakukan dalam mengembangkan perangkat lunak pengamanan informasi dengan model *prototype* akan diuraikan seperti pada bagian bawah ini, antara lain :

1. Pengumpulan kebutuhan.

Pada tahap pertama ini dilakukan pengumpulan kebutuhan dengan cara mengumpulkan berbagai data yang diperlukan berupa referensi dari jurnal ilmiah maupun buku yang berkaitan dengan pembahasan yang akan peneliti lakukan, yaitu steganografi dengan metode LSB dan kriptografi dengan algoritma AES.

2. Membangun *prototyping.*

Setelah data yang dikumpulkan dirasa cukup untuk menunjang pengembangan sistem, pada tahap yang kedua peneliti melakukan proses analisis dan perancangan sistem berdasarkan hasil kesimpulan dari data yang sudah dikumpulkan. Perancangan yang dilakukan menggunakan tools UML untuk memberikan gambaran dari bentuk sistem nantinya. Terdiri dari *activity diagram, dan use case.*

3. Evaluasi *prototyping*.

Pada tahapan ketiga yang peneliti lakukan dalam evaluasi *prototyping* yaitu proses evaluasi *prototyping* yang sudah dirancang sebelumnya, apakah telah sesuai dengan kebutuhan atau belum. Jika sudah sesuai maka dilakukan proses selanjutnya. Tetapi jika belum sesuai, maka akan dilakukan pengecekan pada proses sebelumnya atau kembali ke tahap pertama.evaluasi ini dilakukan dengan cara melihat rancangan pada UML yang dibuat untuk kemudian dibandingkan dengan data yang sudah dikumpulkan pada tahapan pertama.

4. Mengkodekan sistem.

Jika pada tahapan evaluasi, desain sudah sesuai atau memenuhi kebutuhan pengguna, maka akan lanjut pada tahapan keempat, yaitu dilakukan proses pengkodean sistem dengan membuat program menggunakan bahasa pemrograman *visual studio 2010* berdasarkan rancangan yang sudah dibuat sebelumnya.

5. Menguji sistem.

Pada tahapan kelima yaitu pengujian sistem, aplikasi yang sudah dibangun akan diuji dengan menggunakan metode *Black Box* untuk mengetahui fungsi dari sistem apakah sudah berjalan dengan baik atau masih terdapat kekurangan/kesalahan. Selain pengujian fungsi, dilakukan pengujian data dengan melihat data dari berkas yang dijadikan sampel, untuk melihat ada tidaknya perubahan kualitas dari informasi.

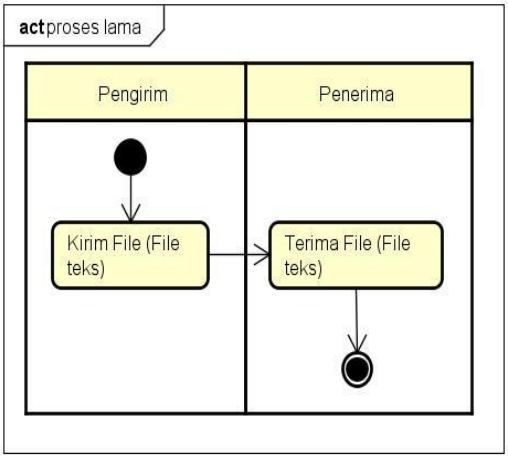
6. Evaluasi sistem.

Setelah tahapan pengujian selesai, tahapan selanjutnya adalah evaluasi sistem. Dimana pada tahapan evaluasi sistem ini jika terdapat perubahan yang diminta oleh penguji program, maka peneliti akan kembali pada tahapan keempat yaitu memperbaiki pengkodean sistem sesuai dengan permintaan penguji.

7. Menggunakan sistem.

Pada tahapan terakhir model pengembangan *prototype*, sistem yang sudah dievaluasi dan sudah dinyatakan lolos oleh penguji program, maka aplikasi pengamanan informasi sudah siap untuk diimplementasikan atau digunakan.

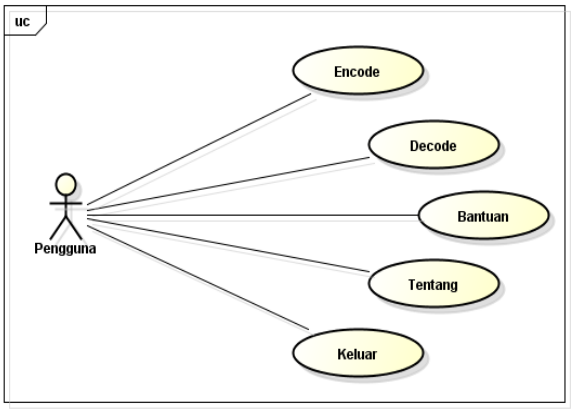
Dari hasil analisa data pada tahapan pertama model *prototype*, maka didapatlah hasil dalam bentuk diagram dengan menggunakan UML sebagai tool pengembangan perangkat lunak dimana dalam penggambaran ini peneliti menggunakan *astah community*, hal ini dimaksudkan untuk mempermudah dalam penggambaran proses bisnis yang terjadi. Seperti pada gambar 1 *activity diagram* sistem berjalan dibawah ini:



Gambar 1. *Acitivity Diagram* Sistem Berjalan

ISSN PRINT : 2303-500X
ISSN ONLINE : 2477-3786

Dari gambar diatas diketahui bahwa proses yang banyak terjadi dari hasil analisa adalah, orang langsung mengirimkan berkas/*file* secara langsung ke tujuan tanpa adanya tambahan usaha untuk mengamankan informasi. Dari hasil tersebut dirancanglah *prototype* berupa *use case diagram* usulan pada sistem yang baru, ditunjukkan seperti gambar 2 dibawah ini:



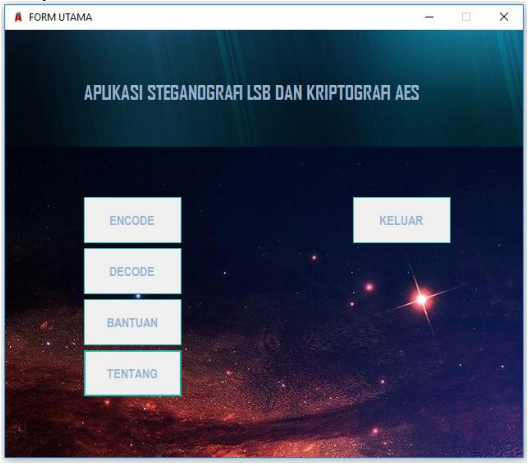
Gambar 2. *Use case usulan sistem*

Use case diagram diatas merupakan rancangan untuk sistem yang akan dibangun, untuk mengetahui fungsi apa saja yang tersedia di aplikasi. *Use case diagram* diatas dibuat berdasarkan hasil analisa dari kebutuhan pengguna. Dimana kebutuhan pengguna disini adalah penyisipan berkas dan enkripsi, serta dekripsi dan ekstraksi berkas, dengan tambahan menu bantuan untuk cara penggunaan dari aplikasi, menu tentang untuk mengetahui fungsi dari aplikasi, dan menu keluar untuk menutup aplikasi.

2. Hasil dan Pembahasan

Pada bagian ini akan dibahas hasil dari perancangan dalam bentuk jadi berupa tangkapan layar dari aplikasi dan hasil pengujian aplikasi.

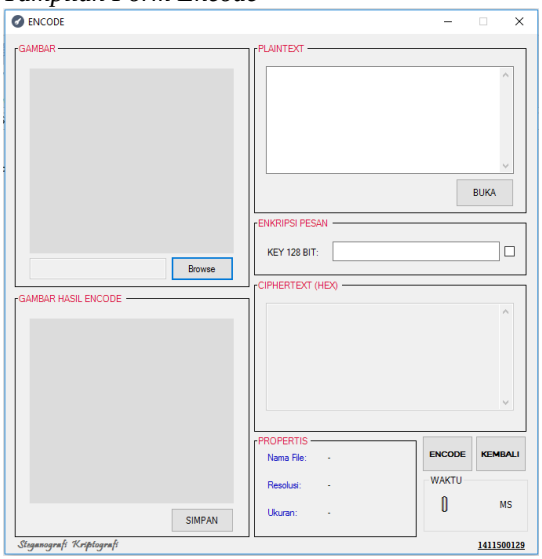
A. Tampilan Form Utama



Gambar 3. *Form* Utama

Dimana pada tampilan utama, terdiri dari lima menu, yaitu *encode* yang berfungsi untuk melakukan penyisipan dan enkripsi data atau informasi kedalam gambar, menu *decode* berfungsi untuk mengeluarkan data atau informasi rahasia dari gambar, menu bantuan untuk mengetahui cara penggunaan dari aplikasi, menu tentang untuk mengetahui fungsi dari aplikasi, dan menu keluar untuk menutup aplikasi.

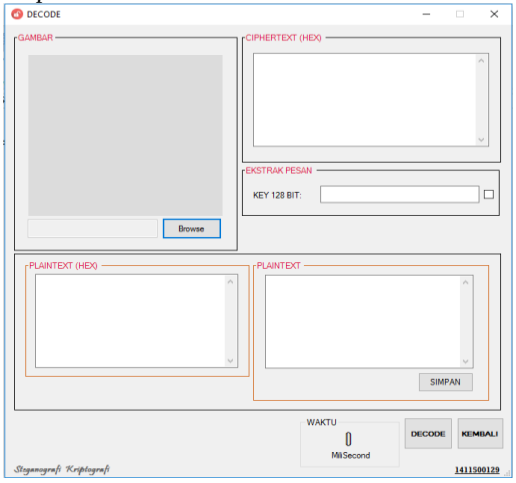
B. Tampilan Form Encode



Gambar 4. *Form Encode*

Pada gambar 4 diatas, pengguna dapat memasukkan data atau informasi dalam bentuk huruf yang ingin di sembunyikan kedalam sebuah gambar, selain itu juga pengguna diharuskan memasukkan kunci kriptografi AES sepanjang 128 bit, dimana waktu berfungsi menampilkan jumlah waktu yang dibutuhkan untuk memproses penyisipan data, dimana proses pertama dilakukan enkripsi menggunakan kunci tertentu, dan hasilnya akan berupa *chipertext hexadecimal*, setelah itu baru kemudian *chipertext* akan disisipkan kedalam gambar sebagai wadah informasi yang ingin di amankan.

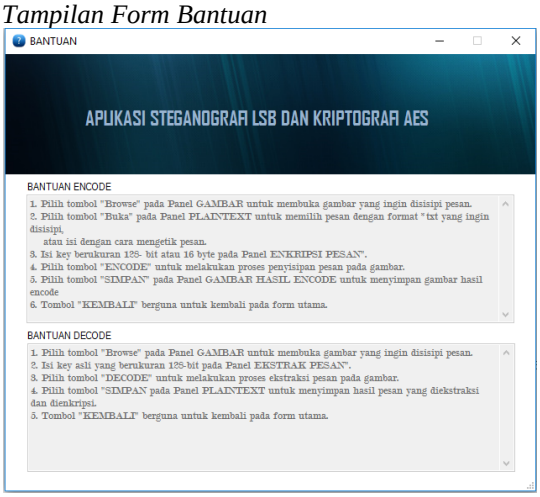
c. Tampilan Form Decode



Gambar 5. *Form Decode*

Selanjutnya, pada bagian *decode*, pengguna yang mendapatkan data atau informasi yang tersimpan didalam gambar dapat mengeluarkan informasi tersebut dengan mencari gambar melalui menu browse, dan secara otomatis akan muncul pesan dalam bentuk *chipertext hexadecimal* untuk menerjemahkan, pengguna diharuskan memasukkan kunci yang sama pada saat enkripsi sepanjang 128 bit, maka pesan akan secara otomatis dikembalikan kedalam bentuk semula atau *plaintext* dalam dua jenis, yaitu *plaintext hexadecimal* dan *plaintext* biasa.

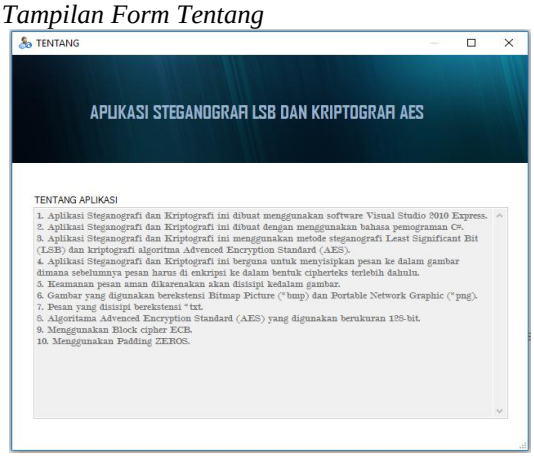
D. Tampilan Form Bantuan



Gambar 6. *Form* Bantuan

Pada form bantuan, pengguna dapat membaca langkah – langkah yang harus dilakukan ketika menggunakan aplikasi pengamanan data ini.

E. Tampilan Form Tentang



Gambar 7. *Form* Tentang

Pada *form* tentang, pengguna dapat membaca fungsi dari aplikasi yang dibuat, metode pengamanan dan algoritma yang digunakan.

a. Pengujian

Pada pengujian yang dilakukan pada aplikasi steganografi dan kriptografi ini, ada dua hal yang diujikan yaitu pengujian fungsi menggunakan metode *blackbox* dan pengujian data berupa pengujian *fidelity* dengan hasil, antara lain:

1. Pengujian fungsi dengan metode *blackbox*

a. Pengujian pertama dilakukan pada menu *encode*, dimana hasil dari pengujian dapat dilihat pada tabel 2 dibawah ini:

Tabel 2. Pengujian fungsi *encode*

No	Kasus Uji	Skenario	Hasil yang Diharapkan	Keterangan
1	Melakukan proses *encode*.	Tidak memasukkan pesan, tidak memasukkan gambar, dan tidak memasukkan *key*. Lalu menekan tombol "ENCODE".	Aplikasi tidak akan melakukan proses *encode* dan akan menampilkan *message* "Data tidak ada boleh yang kosong!"	Berhasil
2	Melakukan proses *encode*.	Memasukkan gambar(*bmp, png*), tapi tidak memasukkan pesan, dan tidak memasukkan *key*. Lalu menekan tombol "ENCODE".	Aplikasi tidak akan melakukan proses *encode* dan akan menampilkan *message* "Pesan yang ingin anda sembunyikan tidak boleh yang kosong!"	Berhasil
3	Melakukan proses *encode*.	Memasukkan gambar(*bmp,png*), memasukkan pesan, dan tidak memasukkan *key*. Lalu menekan tombol "ENCODE".	Aplikasi tidak akan melakukan proses *encode* dan akan menampilkan *message* "Key tidak boleh kosong!"	Berhasil
4	Melakukan proses *encode*.	Memasukkan gambar(*bmp, png*), memasukkan pesan, dan memasukkan *key*. Lalu menekan tombol "ENCODE".	Berhasil mengenkripsi pesan dan menampilkan hasil *ciphertext* (hex), berhasil menampilkan gambar hasil *encode* di panel "GAMBAR HASIL Proses *encode* berjalan lancar.	Berhasil
5	Menampilkan halaman *browse* memilih gambar(*bmp,*	Pengguna memilih tombol "Browse"	Aplikasi menampilkan halaman *browse* untuk memilih gambar (*bmp, png*) yang ada pada media penyimpanan	Berhasil
6	Menampilkan halaman *browse* memilih pesan(*txt*)	Pengguna memilih tombol "BUKA"	Aplikasi menampilkan halaman browse untuk memilih pesan (*txt*) yang ada pada media penyimpanan	Berhasil
7	Berganti tampilan *image view*	Melakukan proses pilih gambar(*bmp,png*) dan menekan tombol "Open"	*Image view* berhasil berganti, dan menampilkan propertis gambar	Berhasil
8	Menampilkan pesan(*txt*) yang dipilih.	Melakukan proses pilih pesan (*txt*) dan menekan tombol "Open"	Pesan teks berhasil ditampilkan.	Berhasil
9	Menyimpan gambar.	Melakukan proses menyimpan gambar dengan menekan tombol "SIMPAN"	Aplikasi berhasil menyimpan gambar(*bmp,png*)	Berhasil
10	Kembali ke menu utama	Pengguna memilih tombol "KEMBALI"	Aplikasi melakukan proses kembali ke menu utama	Berhasil

b. Pengujian kedua dilakukan pada menu *decode*, dimana hasil dari pengujian dapat dilihat pada tabel 3 dibawah ini:

ISSN PRINT : 2303-500X
ISSN ONLINE : 2477-3786

Tabel 3. Pengujian fungsi *decode*

No	Kasus Uji	Skenario	Hasil yang Diharapkan	Keterangan
1	Memilih gambar hasil *decode*	Pengguna memilih tombol "Browse"	Aplikasi menampilkan halaman browse untuk memilih gambar (*bmp, png*) yang ada pada media penyimpanan	Berhasil
2	Melakukan proses *decode*	Memasukkan gambar yang belum pernah disisipi pesan atau yang belum pernah di-*encode* terlebih dahulu. Lalu menekan tombol "DECODE"	Aplikasi akan menampilkan pesan "ÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿ ÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿ ÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿ ÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿ ÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿ ÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿ ÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿ ÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿ..... ÿÿÿÿÿ?" pada *textbox*	Berhasil
3	Melakukan proses *decode*	Memasukkan gambar (*bmp, png*) yang belum disisip pesan dan mengisi *key* yang salah. Lalu menekan tombol	Aplikasi tidak akan melakukan proses *decode* dan akan menampilkan *message* "Tidak ada	Berhasil
4	Melakukan proses *decode*.	Memasukkan gambar(*bmp, png*) yang sudah disisip pesan, tanpa mengisi *key* . Lalu menekan tombol "DECODE".	Aplikasi akan melakukan proses *decode*. Tapi, hasil dari *plaintext* dan *plaintext* (*hex*) tidak keluar, hanya *ciphertext* saja yang ditampilkan	Berhasil
5	Melakukan proses *decode*.	Memasukkan gambar(*bmp, png*) yang sudah disisipi pesan, dan mengisi *key* yang benar. Lalu menekan tombol "DECODE".	Aplikasi akan melakukan proses *decode* dan akan menampilkan hasil *ciphertext* pesan yang disisip dan menampilkan	Berhasil
6	Menyimpan Pesan.	Melakukan proses menyimpan pesan dengan menekan tombol "SIMPAN"	Aplikasi berhasil menyimpan pesan.	Berhasil
7	Kembali ke menu utama	Pengguna memilih tombol "KEMBALI"	Aplikasi melakukan proses kembali ke menu utama	Berhasil

Dari hasil pengujian dengan menggunakan metode *blackbox*, kesimpulan yang didapat semua fungsi berjalan dengan baik tanpa adanya kesalahan pada kode program.

2. Pengujian *fidelity*

Tujuan dari pengujian ini adalah untuk mengetahui perubahan ukuran gambar dengan 2 (dua) jenis ekstensi berkas gambar yang digunakan sebagai wadah dari data atau informasi, yaitu .png dan .bmp.

Tabel 4. Pengujian *fidelity*

Nama File	Dimensi (Ukuran)	Panjang pesan	Ekstensi gambar	Ukuran gambar
Kupukupu .png	768x512	250 karakter	.png	843 Kb
	(599 kb)	1000 karakter	.png	845 Kb
		2000 karakter	.png	847 Kb
		250 karakter	.bmp	1.12 Mb
		1000 karakter	.bmp	1.12 Mb
		2000 karakter	.bmp	1.12 Mb

Kesimpulan dari tabel diatas, terlihat gambar berekstensi .png terjadi perbedaan ukuran saat terjadi penyisipan pesan dengan jumlah karakter yang berbeda. Sedangkan gambar yang berekstensi .bmp terlihat bahwa tidak ada perbedaan ukuran gambar meskipun disisipkan pesan dengan jumlah yang berbeda, hanya saja ukuran gambar dengan ekstensi .bmp lebih besar dari .png.

3. Kesimpulan

1. Penerapan *steganografi* dengan metode *Least Significant Bit* (LSB) dapat dikombinasikan bersama kriptografi AES. Hal ini dapat dibuktikan dengan data yang berhasil di enkripsi dan disisipkan kedalam gambar tanpa mengurangi kualitas gambar/citra asli secara signifikan, sehingga tidak dapat dibedakan dengan gambar/citra dari hasil penyisipan pesan.
2. Ukuran gambar dengan ekstensi .bmp lebih konsisten terhadap penyisipan informasi meskpun jumlah karakter atau huruf yang disisipkan berbeda jumlahnya.
3. Penerapan metode steganografi LSB dan algoritma kriptografi AES dapat diimplementasikan dalam sebuah aplikasi berbasis desktop. Sehingga pesan yang akan dikirimkan akan terlindungi dengan aman. Karena hanya orang yang berkepentingan yang dapat mengetahui pesan tersebut.

Daftar Pustaka

Ariyus, Dony. 2008. Pengantar Ilmu Kriptografi: Teori Analisis & Implementasi. Penerbit Andi: Yogyakarta.

Syaiful Anwar, 2017, Implementasi Pengamanan Data Dan Informasi Dengan Metode Steganografi LSB Dan Algoritma Kriptografi AES, Universitas Budi Luhur, Jakarta Selatan.

Gede Wisnu Bhaudhayana dan I Made Widiartha, 2015, Implementasi Algoritma Kriptografi AES 256 Dan Metode Steganografi LSB Pada Gambar Bitmap, Universitas Udayana, Denpasar.

Ami Aisiah Ibrahim, 2017, Perancangan Pengamanan Data Menggunakan Algoritma AES (Advanced Encyption Standard), STMIK Antar Bangsa, Tanggerang.

Indra Suryanto, Drs. Cucu Suhery, M.A., dan Yulrio Brianorman, S.Si., M.T., 2017, Pengembangan Aplikasi Chat Messenger Dengan Metode Advanced Encryption Standard (AES) Pada Smartphone, Universitas Tanjungpura, Pontianak.

Chandra, A., Raharjo, W. S., dan Tampubolon, J. K., 2015, Implementasi steganografi dengan metode end of file pada teks yang terenkripsi menggunakan block cipher rivest code-6 ke dalam citra, Jurnal Teknik Universitas Kristen Duta Wacana, No.1, Vol.11,:https://labti.ukdw.ac.id/ojs/index.php/informatika/article/view/

ISSN PRINT : 2303-500X
ISSN ONLINE : 2477-3786

Roger, S. Pressman, Ph.D., 2012, Rekayasa Perangkat Lunak (Pendekatan Praktisi), Ed.7, diterjemahkan oleh Andi, Yogyakarta.

Novianto, Dian. 2017. Optimasi Waktu Query Dan Filtering Nama Domain Pada Dns Server Lokal Menggunakan Bind 9. Jurnal Ilmiah Informatika Global Vol. 8 No.1 : http://ejournal.uigm.ac.id/index.php/IG/article/view/320