Faktor Exacta *12 (2):* 132-144, 2019
p-ISSN: 1979-276X
e- ISSN: 2502-339X
DOI : 10.30998/faktorexacta.v12i2.3527

Analisis dan Perancangan Simulasi Enkripsi dan Dekripsi pada Algoritma Steganografi untuk Penyisipan Pesan *Text* pada *Image* menggunakan Metode *Least Significant Bit* (LSB) Berbasis Cryptool2

NOFITA RISMAWATI
Program Studi Informatika
Universitas Indraprasta PGRI, Jakarta, Indonesia
Jl. Nangka No. 58 C Tanjung Barat, Jagakarsa, Jakarta 12530

MUHAMAD FEMY MULYA
Program Studi Sistem Informasi
Tanri Abeng University, Jakarta, Indonesia
Jl. Swadarma Raya No.58, Ulujami, Pesanggrahan, Jakarta 12250
Email: novi.9001@gmail.com, femy.mulya@tau.ac.id

Abstract*. Security and confidentiality of data is the central aspect of an information system so that it can only be accessed by the owner of the information or interested user. In sending data, without the existence of a technique of concealing important information, it will be very easy for people who do not have access rights/interests to know the contents of the information. Steganography is the art and science of hiding secret messages in other messages so that the contents of the secret message cannot be traced. In this study, steganography technic is implemented by using the Least Significant Bit (LSB) method, modifying less significant bits or the last bit in one byte of data by using text messages as a storage medium. The purpose of this study was to analyze and design an encryption and decryption simulation for the insertion of text messages in an image by using the steganography algorithm using the Least Significant Bit (LSB) method, so that it would secure the text messages sent by the sender to the receiver by inserting the text message into an image both in the format * .jpg, * .bmp and * .png. The software will be used to simulate steganography using free program (Open Source) Cryptool2 in order to describe the concepts of cryptography and cryptanalysis.*

Keywords: Steganography, Least Significant Bit (LSB), Cryptool2

Abstrak. Pengamanan dan kerahasiaan data merupakan aspek utama dari suatu sistem informasi, sehingga hanya bisa diakses oleh pemilik dari suatu informasi atau user yang berkepentingan saja. Dalam pengiriman data, tanpa adanya teknik penyembunyian, informasi penting akan sangat mudah bagi orang yang tidak memiliki hak akses/berkepentingan untuk mengetahui isi dari suatu informasi. Steganografi adalah seni dan ilmu menyembunyikan pesan rahasia di dalam pesan lain, sehingga isi didalam pesan rahasia tersebut tidak dapat diketahui. Pada penelitian ini menggunakan steganografi dengan metode *Least Significant Bit* (LSB) yaitu dengan memodifikasi bit yang kurang signifikan atau bit terakhir dalam satu *byte* data dengan menggunakan pesan *text* sebagai media penampung. Tujuan dari penelitian ini adalah untuk menganalisa dan merancang sebuah simulasi enkripsi dan dekripsi untuk penyisipan pesan *text* pada sebuah *image* dengan algoritma steganografi menggunakan metode *Least Significant Bit* (LSB), sehingga akan mengamankan pesan *text* yang akan dikirimkan oleh *sender* kepada *receiver* dengan menyisipkan pesan *text* tersebut kedalam sebuah *image* baik dengan format *.jpg, *.bmp maupun *.png. Perangkat lunak (*software*) yang akan digunakan untuk membuat simulasi steganografi ini menggunakan Cryptool2 yang merupakan sebuah program gratis (*Open Source*) yang digunakan untuk menggambarkan atau mendeskripsikan konsep kriptografi dan kriptoanalisis.
Kata Kunci: Steganografi, *Least Significant Bit* (LSB), Cryptool2

Faktor Exacta *12 (2):* 132-144, 2019
p-ISSN: 1979-276X
e- ISSN: 2502-339X
DOI : 10.30998/faktorexacta.v12i2.3527

PENDAHULUAN

Perkembangan teknologi data dan informasi serta komunikasi saat ini merupakan bagian penting dalam kehidupan manusia sehari-hari. Seiring perkembangan zaman, kebutuhan manusia akan informasi semakin meningkat. Ditengah-tengah perkembangan teknologi data dan informasi yang semakin maju dan berkembang, internet tidak lagi dapat menjamin untuk menyediakan data dan informasi yang cukup aman bagi penggunanya. Berbagai teknologi mesin pencari (*search-engine*) semakin berkembang ditambah dengan serangan berbagai tipe virus, penyadap, *spam* maupun *hacker/cracker* yang menjamur sehingga dapat mencuri data-data yang bersifat rahasia.

Dalam perkembangan pertukaran data dan informasi, salah satu kendala yang ditemukan adalah mengenai masalah keamanan. Adapun beberapa hal yang harus diperhatikan ketika pesan dikirimkan oleh pengirim hingga pesan bisa sampai kepada penerima adalah kerahasiaan (*confidentiality*), keutuhan (*integrity*), *authentication*, *availability*, *Access Control* dan *non-repudiation.* Keenam aspek-aspek keamanan komputer ini dapat memproteksi pesan yang dikirimkan oleh pengirim (*sender*) kepada penerima (*receiver*) agar pesan yang dikirimkan bisa sampai ke tujuan tanpa kekurangan satu bit data pun.

Confidentiality adalah menjaga informasi dari orang yang tidak berhak mengakses, lebih kearah data-data yang sifatnya pribadi (*private*), seperti e-mail, rekening tabungan, dan lainnya. *Integrity* dalam istilah keamanan informasi berarti aspek yang menjamin bahwa data tidak boleh berubah tanpa ijin pihak yang berwenang (*authorized*). A*uthentication* berarti suatu langkah untuk menentukan atau mengkonfirmasi bahwa seseorang (atau sesuatu) adalah autentik atau asli. Melakukan autentikasi terhadap sebuah objek adalah melakukan konfirmasi terhadap kebenarannya. Sedangkan melakukan autentikasi terhadap seseorang biasanya adalah untuk memverifikasi identitasnya. Pada suatu sistem komputer, autentikasi biasanya terjadi pada saat login atau permintaan akses. *Availability* merupakan upaya pencegahan ditahannya informasi atau sumber daya terkait oleh mereka yang tidak berhak. Secara umum maka makna yang dikandung adalah bahwa informasi yang tepat dapat diakses bila dibutuhkan oleh siapapun yang memiliki legitimasi untuk tujuan ini. Berkaitan dengan "*messaging system*" maka pesan itu harus dapat dibaca oleh siapapun yang dialamatkan atau yang diarahkan, sewaktu mereka ingin membacanya. *Access Control* dalam istilah keamanan informasi berarti mekanisme untuk mengatur "siapa boleh melakukan apa", "dari mana boleh ke mana', sedangkan untuk penerapannya membutuhkan klasifikasi data (*public*, *private*, *confident*, *secret*) dan berbasiskan *role* (kelompok atau group hak akses). *Non-repudiation* adalah aspek untuk menjaga agar seseorang tidak dapat menyangkal telah melakukan sebuah transaksi, sebagai contoh, seseorang yang mengirimkan email untuk memesan barang tidak dapat menyangkal bahwa dia telah mengirimkan email tersebut(Zinaly & Naghipour, 2017).

Dari beberapa aspek masalah keamanan komputer di atas, dengan demikian dibutuhkan suatu metode untuk pengamanan pesan *text* yang dinamakan steganografi (*steganography*). Steganografi merupakan seni dan ilmu menyembunyikan pesan ke dalam pesan lainnya sedemikian rupa dengan demikian orang lain tidak akan menyadari bahwa di dalam pesan tersebut terkandung sesuatu.

Dari uraian yang telah diberikan, pada penelitian ini akan dijawab permasalahan bagaimana pemanfaatan metode steganografi untuk menjaga integritas dan keamanan data dan informasi yang saat ini sudah semakin berkembang, serta bagaimana cara untuk menyisipkan dan mengekstrak pesan dari dalam media gambar (*image*) sesuai dengan pesan yang disisipkan.

Tujuan dari penelitian ini adalah untuk menganalisa dan merancang sebuah simulasi enkripsi dan dekripsi untuk penyisipan pesan *text* pada sebuah *image* dengan algoritma steganografi dengan menggunakan metode *Least Significant Bit* (LSB), sehingga akan mengamankan pesan *text* yang akan dikirimkan oleh *sender* kepada *receiver* dengan menyisipkan pesan *text* tersebut kedalam sebuah *image* baik dengan format *.jpg, *.bmp

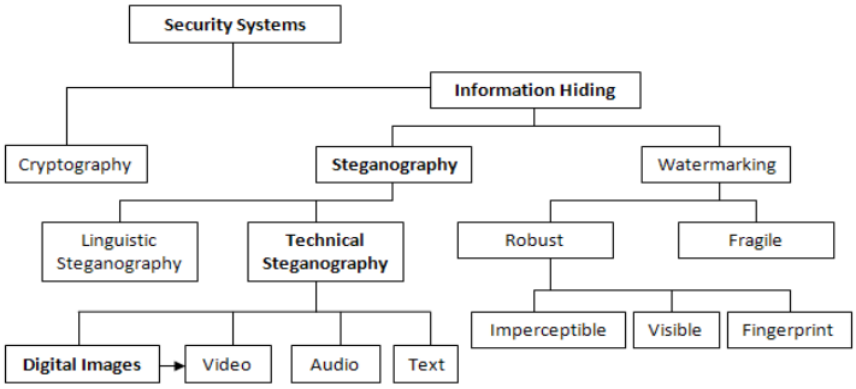
Faktor Exacta *12 (2):* 132-144, 2019
p-ISSN: 1979-276X
e- ISSN: 2502-339X
DOI : 10.30998/faktorexacta.v12i2.3527

maupun *.png. Perangkat lunak yang digunakan untuk membuat simulasi steganografi ini menggunakan Cryptool2.

Pengertian Steganografi

Steganografi adalah suatu ilmu dan seni untuk menyembunyikan pesan. Steganografi membutuhkan dua properti yaitu tempat menyembunyikan pesan dan pesan yang akan disembunyikan. Media penampung yang umum digunakan adalah gambar, suara, video ataupun *text*. Pesan yang disembunyikan dapat berupa sebuah artikel, gambar, daftar barang, kode program ataupun pesan lain(Lestari, Nurmaesa, & Mariana, 2017).

Tujuan dari Steganografi ini sendiri berguna untuk merahasiakan atau menyembunyikan sebuah pesan atau informasi agar tidak diketahui orang lain. Steganografi membutuhkan dua properti seperti: suatu wadah penampung dan data-data rahasia yang ingin disembunyikan. Steganografi digital menggunakan media digital sebagai wadah penampung, misalnya citra, suara, teks, dan video. Data rahasia yang disembunyikan juga dapat berupa citra, suara, teks, atau video. Dalam prakteknya, kebanyakan pesan yang akan disembunyikan, yaitu dengan membuat sedikit perubahan pada data digital lain yang isinya tidak menarik perhatian dari penyerang potensial, misalnya saja sebuah objek gambar yang terlihat tidak berbahaya, akan tetapi didalamnya terdapat informasi yang cukup penting.



Gambar 1. Fokus disiplin ilmu penyisipan(Hussein, Abbass, Naji, Al-augby, & Lafta, 2018)

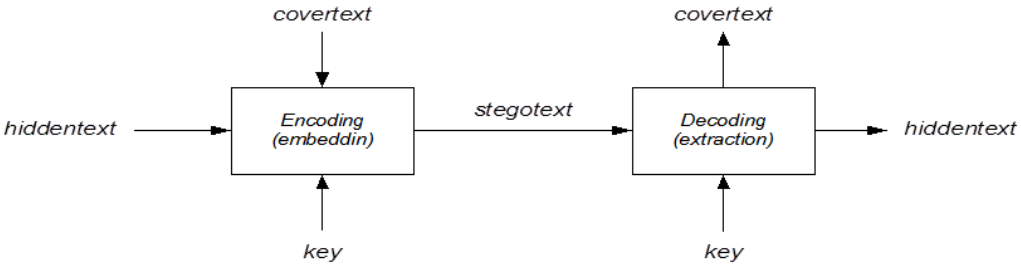
Adapun beberapa jenis citra yang bisa disisipikan pesan dalam steganografi adalah sebagai berikut(Widyawan, 2018):

1. JPG / JPEG (*Joint Photographic Experts Assemble*) merupakan jenis data yang dikembangkan oleh *Joint Photographic Experts Assemble* (JPEG) yang merupakan standar baku untuk para fotografer profesional.
2. PNG (*Portable Network Graphic*) merupakan jenis data yang dikembangkan sebagai alternatif lain untuk format GIF, dengan menggunakan paten LZW– algoritma kompresi. PNG juga merupakan format gambar yang sangat baik untuk tampilan grafis di internet, karena sudah mendukung transparansi didalam sebuah *browser* dan memiliki ciri tersendiri yang tidak bisa diberikan oleh format GIF atau bahkan format gambar dalam bentuk JPG. Adapun kelebihan dari file PNG yaitu, adanya unsur warna transparan dan *alpha*. Unsur warna *alpha* memungkinkan sebuah objek gambar menjadi transparan, akan tetapi gambar tersebut masih dapat dilihat oleh mata seperti, samar-samar atau bening.
3. BMP (*Bitmap*). merupakan representasi dari suatu citra (*image*) grafis yang terdiri dari susunan titik (*pixel*) yang disimpan pada memori komputer. Nilai untuk setiap titik selalu diawali oleh satu bit data (untuk citra dengan warna hitam putih) atau lebih (untuk citra yang berwarna). Kerapatan titik-titik pada suatu citra tersebut

Faktor Exacta *12 (2):* 132-144, 2019
p-ISSN: 1979-276X
e- ISSN: 2502-339X
DOI : 10.30998/faktorexacta.v12i2.3527

dinamakan *resolusi*, yang menunjukkan seberapa tajam gambar ini ditampilkan, ditunjukkan dalam bentuk total/jumlah *pixel* baris dan kolom (contoh: 1024×768)(Reddy Ch & Ramani, n.d.).

Properti Steganografi



Gambar 2. Properti Steganografi(Anti, Kridalaksana, & Khairina, 2017)

1. *Embedded message* (*hiddentext)*: pesan yang akan disembunyikan.
2. *Cover-object* (*covertext):* pesan yang akan digunakan untuk menyembunyikan *hiddentext*.
3. *Stego-object* (*stegotext):* pesan yang telah berisi pesan *hiddentext*.
4. *Stego-key*: kunci yang digunakan untuk melakukan penyisipan pesan dan mengekstraksi pesan dari *Stego-object* (*stegotext)*.

Kriteria Steganografi

Kriteria steganografi yang harus diperhatikan dalam menyembunyikan sebuah data, image, teks dan suara antara lain(Utami, n.d.):

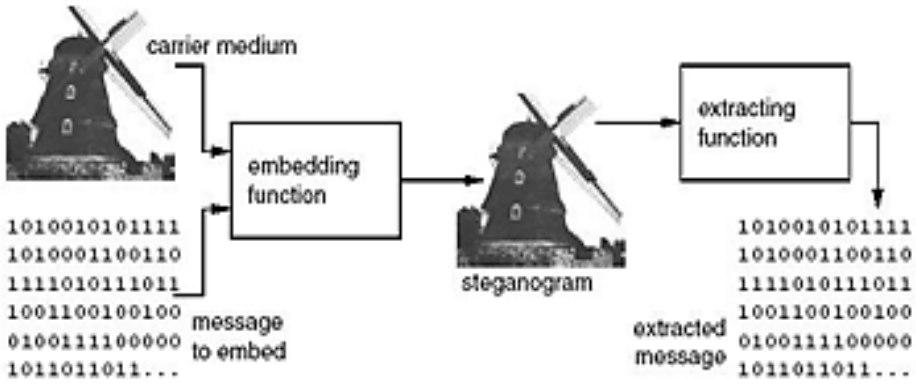
1. *Fidelity*. Kualitas mutu penampung tidak jauh berubah dengan yang aslinya. Apabila dilakukan penambahan data rahasia, maka hasil steganografi masih terlihat dengan baik. Pengamat tidak akan mengetahui kalau di dalam citra tersebut terdapat suatu data rahasia.
2. *Robustness*. Data yang akan disembunyikan harus dapat bertahan terhadap manipulasi yang dilakukan terhadap penampung (seperti perubahan terhadap kontras, penajaman, pemampatan, rotasi, perbesaran gambar, pemotongan gambar (cropping), enkripsi dan lainnya). Bila pada citra/gambar dilakukan suatu operasi pengolahan, maka data yang disembunyikan tidak mengalami kerusakan.
3. *Recovery*. Data yang akan disembunyikan harus mampu untuk diungkapkan kembali (*recovery*), dimana tujuan steganografi adalah data *hidding*, maka apabila sewaktu-waktu data rahasia yang terdapat didalam penampung dibutuhkan, maka dapat diambil kembali untuk dipergunakan lebih lanjut.

Metode *Least Significant Bit* (LSB)

Salah satu metode yang umumnya digunakan dalam teknik steganografi pada tipe berkas audio dan gambar adalah *least significant bit* atau biasa disebut juga dengan *low bit encoding*(Darwis, 2015). Metode ini berasal dari angka yang paling kurang signifikan dari jumlah bit dalam 1 *byte*. Pengubahan *Least Significant Bit* (LSB) pada citra/gambar yang terkompresi sangat sulit diketahui secara kasat mata, sehingga metode ini termasuk kategori sangat baik dan banyak digunakan. Metode ini memanfaatkan ketidakmampuan indera penglihatan manusia dalam menemukan perbedaan antara gambar yang asli dengan gambar yang sudah dimasukkan/disisipkan pesan rahasia. Pada Gambar 3 ditunjukan bahwa medium pembawa yang disisipkan pesan dengan menggunakan suatu fungsi penyisipan,

Faktor Exacta *12 (2):* 132-144, 2019
p-ISSN: 1979-276X
e- ISSN: 2502-339X
DOI : 10.30998/faktorexacta.v12i2.3527

dalam hal ini LSB , menghasilkan *Stego-Image* yang tidak mengalami perubahan yang *significant* dari gambar aslinya.



Gambar 3. Penyisipan pesan pada gambar(Fuad, -, & Ir. Endang Setyati, 2011)

Untuk menjelaskan metode LSB ini, maka digunakan citra digital sebagai *Image-Object*. Setiap *Pixel* pada citra digital memiliki ukuran 1 sampai 3 byte. Pada susunan bit didalam byte (*1 byte = 8 bit*), terdapat bit yang kurang berarti *Least Significant Bit* (LSB). Misalnya pada byte 00110011, maka bit LSB-nya adalah bit yang terletak paling kanan yaitu 1. Dengan demikian untuk melakukan penyisipan pesan terhadap citra, maka bit paling cocok untuk diubah dengan bit pesan adalah bit LSB, karena pengubahan bit pada citra/gambar hanya akan merubah nilai byte-nya menjadi satu lebih tinggi atau satu lebih rendah.

Sebagai contoh, urutan bit berikut ini menggambarkan 3 *Pixel* pada *Cover-Image* 24 bit(Utami, n.d.).

(01010110	10111001	10000110)
(10001001	10001010	00010011)
(01011110	01111000	10101010)

Pesan yang akan disisipkan pada sebuah citra/gambar adalah karakter "**M**", yang nilai binernya adalah **10010011**, maka yang aka dihasilkan *Stego-Image* dengan urutan bit sebagai berikut:

(0101011**1**	1011100**0**	1000011**0**)
(1000100**1**	1000101**0**	0001001**0**)
(0101111**1**	0111100**1**	10101010)

Perubahan yang tidak *significant* ini tidak dapat ditangkap oleh indera penglihatan manusia (jika media wadah berupa gambar, audio dan video).

Dalam contoh diatas penggantian *pixel* tak *significant* dilakukan secara terurut. Penggantian *pixel* tak *significant* juga bisa dilakukan secara tidak terurut, bahkan hal seperti ini bisa lebih meningkatkan keamanan sebuah data.

Pada gambar Bitmap 24-bit , tiap *pixel*-nya terdapat 24-bit kandungan warna atau 8-bit untuk masing-masing warna dasar (R, G dan B), dengan besaran nilai kandungan antara 0 (00000000) samapai dengan 255 (11111111) untuk setiap warna. perubahan LSB ini pada gambar jenis ini hanya akan merubah 1 nilai dari 256 nilai sehingga gambar hasil Steganografi akan sulit dibedakan dengan gambar aslinya(Gunawan, n.d.).

Steganografi dengan menggunakan metode LSB hanya mampu untuk menyimpan informasi dengan ukuran yang relatif sangat terbatas. Sebagai contoh: suatu citra 24-bit

Faktor Exacta *12 (2):* 132-144, 2019
p-ISSN: 1979-276X
e- ISSN: 2502-339X
DOI : 10.30998/faktorexacta.v12i2.3527

(R=8, G=8, B=8) digunakan sebagai tempat untuk menyimpan data berukuran 100-bit, jika masing-masing komponen warnanya (RGB) menggunakan satu *pixel* untuk menyimpan informasi/pesan rahasia tersebut, maka setiap *pixel*-nya menyimpan 3-bit informasi, dengan demikian setidaknya dibutuhkan citra/gambar wadah berukuran 34 *pixel*. Jadi suatu citra/gambar 24-bit jika digunakan untuk menyimpan/menyisipkan informasi rahasia, maka hanya dapat menampung informasi berukuran 1/8 dari ukuran citra/gambar penampung tersebut.

Kapasitas gambar maksimum pesan yang dapat ditampung adalah panjang gambar x lebar gambar x 3 bit. Sebagai contoh, Desktop umum berukuran 1024 *pixel* x 768 *pixel*. Jadi ukuran pesan maksimum pada gambar dengan ukuran tersebut adalah 2.359.296 bit, atau sebanyak 294.912 karakter (1 karakter = 1 byte atau 8 bit). Selanjutnya jika file lebih besar dari *Image* maka akan manfaatkan metode *Resize Image* dan Kompres pada file. untuk menutupi kelemahan yang dimiliki oleh Metode *Least Significant Bit*.

CrypTool2

CrypTool2 merupakan sebuah tool yang digunakan untuk menggambarkan atau mendeskripsikan konsep kriptografi dan kriptoanalisis. Secara umum aplikasi Cryptool2 ini mendukung dua algoritma kriptografi, yaitu algoritma kriptografi *modern* dan algoritma kriptografi *classic*. CrypTool2 adalah sebuah program gratis (*Open Source*) yang dikembangkan oleh University of Kassel (Jerman)("CodeSaya | Yuk belajar Kriptografi atau Enkripsi lebih mudah dengan software "CrypTool"," n.d.).

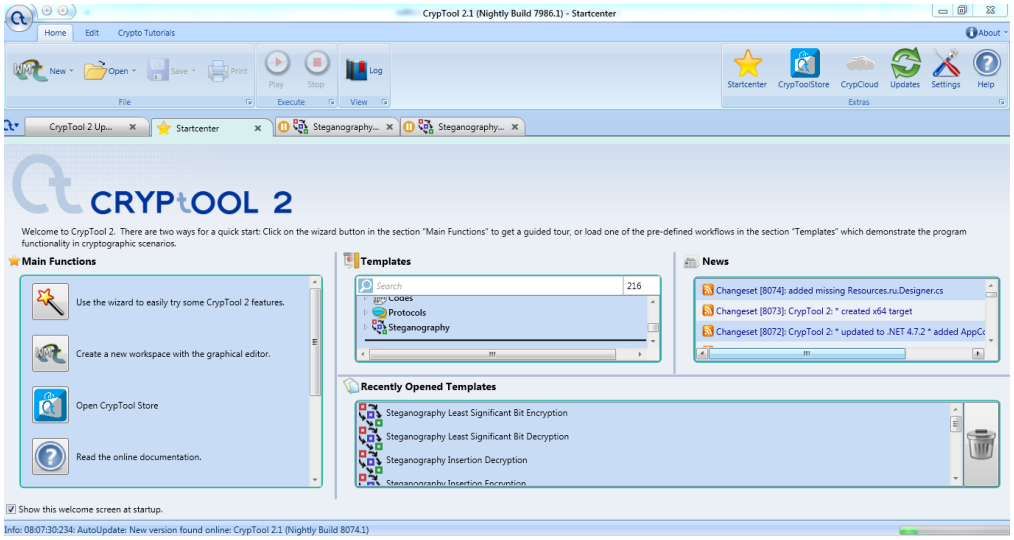
Perangkat lunak (*software*) ini sangat bermanfaat bagi orang yang sedang mempelajari atau belajar mengenai kriptografi. Akan tetapi bagi pengguna umum maupun praktisi komputer, *software* ini amat sangat bermanfaat sekali terutama bagi yang ingin memperdalam pengetahuan tentang algoritma kriptografi untuk proses enkripsi dan dekripsi. Adapun contoh algoritma yang dapat diselesaikan dengan *software* ini antara lain seperti : Caesar, Vigenere, MD5, TEA, AES, DES, RSA serta lebih dari 300 algoritma lainnya. CrypTool2 juga menyediakan antarmuka (GUI) pengguna grafis untuk pemrograman visual.

Pada penelitian ini, menggunakan *tools* simulasi CrypTool2, karena *tools* ini bersifat *open source* (gratis) serta memiliki *user interface* yang mudah dipahami dan dioperasikan, selain itu *tools* ini juga memiliki kelebihan untuk mengkombinasikan beberapa algoritma kriptografi menjadi satu kesatuan yang utuh.

Fitur-Fitur CrypTool2("Belajar Kriptografi/Enkripsi lebih mudah dg program gratis CrypTool – ebsoft," n.d.)

1. Memberikan antarmuka (GUI) untuk berbagai algoritma kriptografi, termasuk visualisasi yang bisa diatur untuk setiap parameternya.
2. Tampilan GUI *software* menggunakan antaramuka seperti Office 2007.
3. Menyediakan berbagai algoritma Cipher klasik, seperti Caesar, Enigma, Hill Cipher, M209, Solitaire, Substitution, XOR, Trasnposition dan lain sebagainya.
4. Menyediakan berbagai algoritma Cipher modern baik Algoritma kunci Simetris (misalnya AES, DES, HC128, Camellia, RC2, RC4, Salsa20, SDES, TEA, Twofish, dan sebagainya) ataupun Algoritma Kunci Asimetris (DGK, Paillier, RSA, RSA Key Generator).
5. Mendukung beberapa metode untuk simulasi Stenografi dengan lengkap, seperti penyisipan, LSB dan Permutasi.

Faktor Exacta *12 (2):* 132-144, 2019
p-ISSN: 1979-276X
e- ISSN: 2502-339X
DOI : 10.30998/faktorexacta.v12i2.3527



Gambar 4. Tampilan antarmuka CrypTool2

METODE

Untuk menganalisa dan merancang simulasi enkripsi dan dekripsi pada algoritma steganografi untuk penyisipan pesan *text* pada sebuah *image* menggunakan metode *Least Significant Bit* (LSB) dalam melakukan penelitian ini, jenis penelitian yang digunakan adalah penelitian dengan studi literatur dan metode kuantitatif.

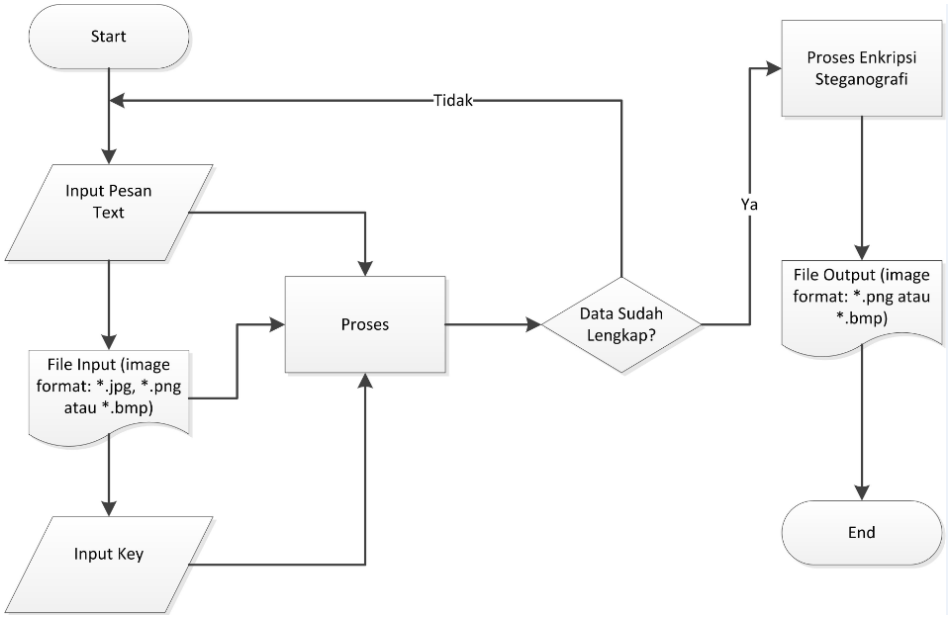
Pada studi literatur dilakukan studi pustaka yang membahas teknik penyembunyian (*embedding* dan *extraction*) dengan algoritma metode *Least Significant Bit* (LSB). Kemudian pada metode kuantitatif dilakukan menggunakan metode penelitian eksperimental, yaitu dengan melakukan eksperimen terhadap variabel-variabel *input* untuk menganalisis *output* yang dihasilkan. Penelitian Eksperimental merupakan bentuk penelitian dimana peneliti (eksperimenter) dengan sengaja melakukan uji coba terhadap objek yang terdapat pada perangkat lunak (*software*) simulasi, selanjutnya dilakukan pengamatan dan pencatatan hasil uji coba yang dilakukan, kemudian melihat hubungan diberikan dan reaksi yang muncul dari Proses. Adapun dua proses yang akan dilakukan untuk simulasi steganografi ini, yaitu proses enkripsi dan dekripsi. Enkripsi adalah proses penyisipan pesan *text* kedalam *image* dan dekripsi adalah proses ekstraksi untuk mengeluarkan pesan *text* dari *image*.

HASIL DAN PEMBAHASAN

Analisis Alur Kerja Simulasi Steganografi dengan Metode LSB

Analisis ini dilakukan guna memahami alur kerja simulasi steganografi untuk proses enkripsi dan dekripsi pesan *text* yang akan disisipkan pada sebuah objek gambar (*image*) dengan metode *Least Significant Bit* (LSB). Berikut adalah *flowchart* untuk simulasi enkripsi steganografi LSB untuk pesan text yang disisipkan pada sebuah gambar (*image*).

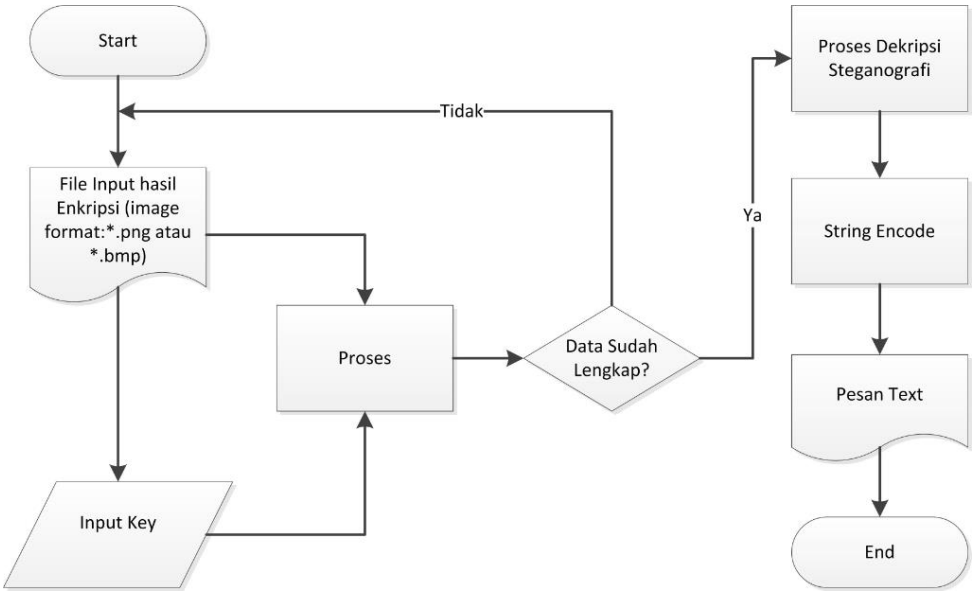
Faktor Exacta *12 (2):* 132-144, 2019
p-ISSN: 1979-276X
e- ISSN: 2502-339X
DOI : 10.30998/faktorexacta.v12i2.3527



Gambar 5. *Flowchart* simulasi enkripsi steganografi dengan metode LSB

Pada *flowchart* simulasi enkripsi steganografi dengan metode LSB terlihat pada gambar 5, langkah pertama untuk melakukan proses enkripsi steganografi dengan metode LSB dibutuhkan inputan berupa pesan *text*, file input *image* dalam format: *.jpg, *.png atau *.bmp, serta *input key* (ini harus diingat, karena akan dipakai untuk proses dekripsi *image* ke pesan *text*). Kemudian jika semua data telah selesai diinput, proses selanjutnya akan dilakukan perhitungan enkripsi steganografi dengan menggunakan metode LSB, lalu akan dihasilkan output berupa gambar (*image*) dengan format: *.png atau *.bmp.

Berikut adalah *flowchart* untuk simulasi dekripsi steganografi dengan metode LSB untuk gambar (*image*) yang akan di tampilkan isi pesan *text*-nya.



Gambar 6. *Flowchart* simulasi dekripsi steganografi dengan metode LSB

Pada *flowchart* simulasi dekripsi steganografi dengan metode LSB terlihat pada gambar 6, menjelaskan bagaimana proses dekripsi sebuah gambar (*image*) supaya bisa menampilan

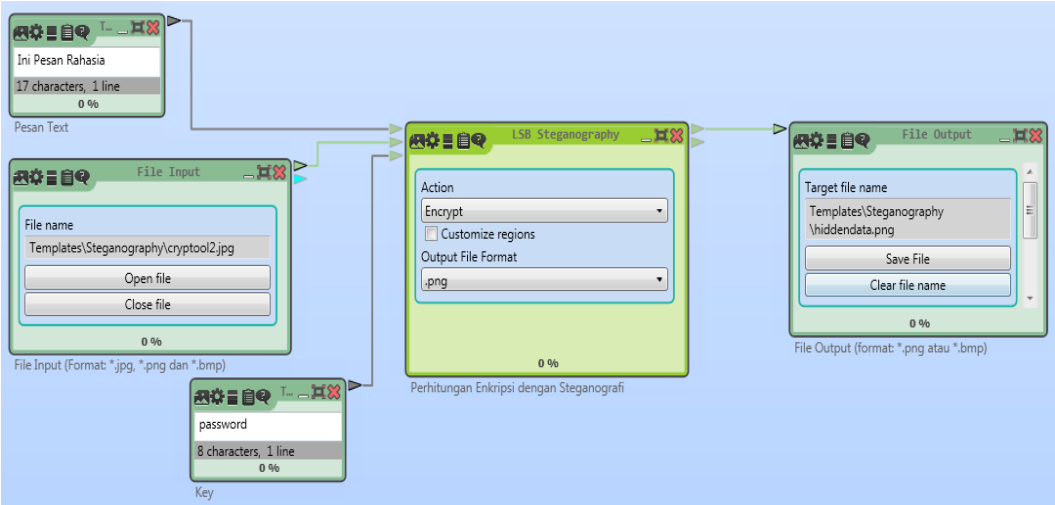
Faktor Exacta *12 (2):* 132-144, 2019
p-ISSN: 1979-276X
e- ISSN: 2502-339X
DOI : 10.30998/faktorexacta.v12i2.3527

isi pesan *text*-nya yang tersembunyi pada gambar (*image*). Langkah pertama masukkan *file input* berupa gambar (*image*) hasil output dari enkripsi steganografi LSB, kemudian masukkan pula *input key* yang dimasukkan sama persis saat proses enkripsi steganografi dengan metode LSB. Selanjutnya jika semua data telah selesai diinput, maka proses berikutnya akan dilakukan perhitungan dekripsi steganografi dengan metode LSB dan perhitungan string encode (merubah biner kedalam bentuk ascii), kemudian akan dihasilkan output berupa pesan *text*.

Simulasi Enkripsi Steganografi dengan metode LSB berbasis Cryptool2

Untuk membuat simulasi enkripsi steganografi dengan metode LSB berbasis cryptool2 kita harus membuat design seperti pada gambar 7 (desain mengikuti alur *flowchart* yang sudah dibuat). Adapun beberapa *Properties*/*Tools* yang dibutuhkan antara lain sebagai berikut:

1. *Text input* sebanyak 2 (dua) buah, yang satu digunakan untuk pesan *text* yang akan disisipkan ke objek gambar (*image*). Kemudian yang kedua digunakan untuk *input key* (kunci steganografi).
2. Satu buah *File Input* yang akan digunakan untuk menginput sebuah gambar (*image*) dalam format: *.jpg, *.png dan *.bmp.
3. Satu buah *LSB Steganography*, kemudian kita pilih actionnya untuk *Encrypt* (enkripsi) dan kita pilih juga output file format: *.png (support format output: *.png dan *.bmp).
4. Satu buah *File Output* yang akan digunakan untuk menyimpan file gambar hasil enkripsi steganografi dalam format: *.png dan *.bmp.



Gambar 7. Simulasi Enkripsi Steganografi dengan metode LSB berbasis Cryptool2

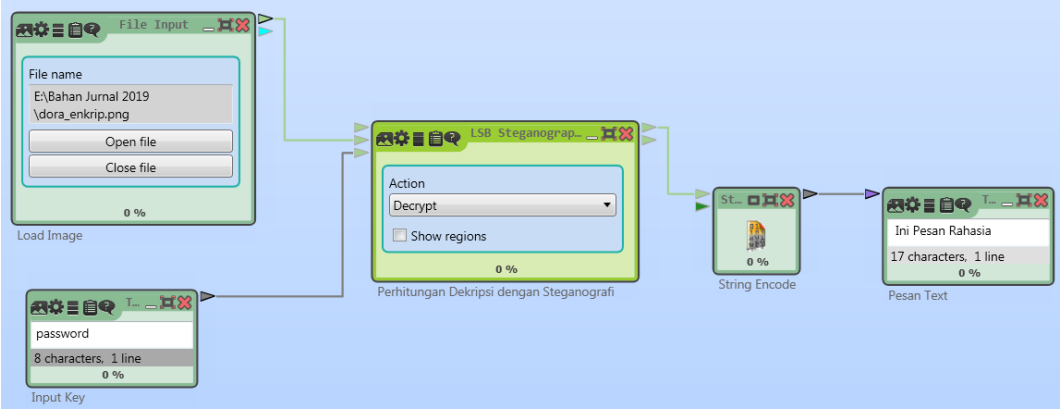
Simulasi Dekripsi Steganografi dengan metode LSB berbasis Cryptool2

Untuk membuat simulasi dekripsi steganografi dengan metode LSB berbasis cryptool2 kita harus membuat design seperti pada gambar 8. Adapun beberapa *Properties*/*Tools* yang dibutuhkan antara lain sebagai berikut:

1. Satu buah *Text input*, yang akan digunakan untuk *input key* (kunci steganografi).
2. Satu buah *File Input* yang akan digunakan untuk menginput sebuah gambar (*image*) yang telah disisipkan pesan *text*, dengan format: *.png dan *.bmp.

Faktor Exacta *12 (2):* 132-144, 2019
p-ISSN: 1979-276X
e- ISSN: 2502-339X
DOI : 10.30998/faktorexacta.v12i2.3527

3. Satu buah *LSB Steganography*, kemudian kita pilih actionnya untuk *Decrypt* (dekripsi) dan kita pilih juga output file format: *.png (support format output: *.png dan *.bmp).
4. Satu buah *String Encode* yang digunakan untuk merubah pesan biner menjadi *text*.
5. Satu buah *Text Output* yang akan digunakan untuk menampilkan isi pesan *text* dari proses perhitungan dekripsi steganografi.



Gambar 8. Simulasi Dekripsi Steganografi dengan metode LSB berbasis Cryptool2

Uji Coba Enkripsi dan Dekripsi Steganografi dengan Metode LSB

Pada penelitian ini akan dilakukan pengujian terhadap simulasi enkripsi dan dekripsi steganografi dengan metode LSB menggunakan perangkat lunak cryptool2. Adapun perbandingan hasil gambar (*image*) original dan hasil gambar (*image*) setelah disisipkan pesan *text* dengan metode LSB sebagai berikut:

Tabel 1. Hasil Uji Coba Simulasi Enkripsi Steganografi dengan Metode LSB

		
Gambar/Image	Original	Hasil Enkripsi
Pesan Text	-	Ini Pesan Rahasia
Key (Kunci)	-	password
Format	*.png	*.png
Resolusi Gambar	820 × 557 pixels	820 × 557 pixels
Ukuran File	97 KB	128 KB
Waktu Proses	0,061 second	

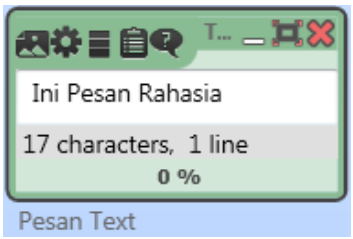
Dari hasil uji coba simulasi proses enkripsi steganografi dengan metode LSB berbasis Cryptool2 yang terlihat pada tabel 1, terdapat beberapa perbedaan antara gambar (*image*) original dan hasil enkripsi dengan format *.png yang dihasilkan tidak bisa dibedakan oleh indera penglihatan manusia. Kemudian untuk ukuran file *.png juga mengalami perubahan dari 97KB (untuk *image* original) menjadi 128KB (untuk *image* hasil enkripsi) atau mengalami peningkatan ukuran file sebesar 31,96%. Lalu waktu proses yang dibutuhkan untuk menyisipkan pesan *text* dari gambar (*image*) original menjadi gambar (*image*) hasil

Faktor Exacta *12 (2):* 132-144, 2019
p-ISSN: 1979-276X
e- ISSN: 2502-339X
DOI : 10.30998/faktorexacta.v12i2.3527

enkripsi dibutuhkan waktu 0,061 *second* untuk resolusi gambar (*image*) 820 *pixels* untuk *width* dan 557 *pixels* untuk *height*.

Kemudian pada uji coba berikutnya akan dilakukan uji coba simulasi proses dekripsi steganografi dengan metode LSB berbasis Cryptool2 yang terlihat pada tabel 2. Dari hasil uji coba pada tabel 2 terlihat bahwa isi pesan *text* berupa "*Ini Pesan Rahasia*" berhasil ditampilkan isi pesannya oleh perangkat lunak Cryptool2 dengan lengkap, serta waktu proses selama 0,031 *second* jauh lebih cepat dari simulasi proses enkripsi steganografi.

Tabel 2. Hasil Uji Coba Simulasi Dekripsi Steganografi dengan Metode LSB

		
Gambar/Image	Hasil Enkripsi	-
Pesan Text	Ini Pesan Rahasia (*Hidden*)	Ini Pesan Rahasia
Key (Kunci)	password	-
Format	*.png	-
Resolusi Gambar	820 × 557 pixels	-
Ukuran File	128 KB	-
Waktu Proses	0,031 second	

Selanjutnya akan dilakukan uji coba sampling sebanyak 10 kali uji coba untuk masing-masing proses enkripsi dan dekripsi steganografi dengan metode LSB yang memiliki format gambar (*image*) *.png, lalu untuk sampling yang digunakan adalah objek gambar (*image*) berukuran foto, adapun hasil uji cobanya terlihat pada tabel 3 sebagai berikut:

Tabel 3. Hasil Uji Coba Simulasi Enkripsi dan Dekripsi Steganografi dengan Metode LSB untuk Resolusi Gambar Ukuran Foto dengan format *.png

No.	Gambar / *Image*		Enkripsi Steganografi dengan LSB				Dekripsi Steganografi dengan LSB	
				Ukuran File (KB)				
	Ukuran Foto	Resolusi (Pixels)	Key (Kunci)	Original	Hasil Enkripsi	Waktu Proses (second)	Pesan Text	Waktu Proses (second)
1	3R	1051×1500	password	110	141	0,082	Ini Pesan Rahasia1	0,041
2	4R	1205×1795		121	152	0,132	Ini Pesan Rahasia2	0,066
3	5R	1500×2102		143	174	0,971	Ini Pesan Rahasia3	0,486
4	6R	1795×2551		165	196	2,421	Ini Pesan Rahasia4	1,211
5	8R	2398×3000		188	219	5,221	Ini Pesan Rahasia5	2,611
6	S8R	2398×3602		193	224	6,301	Ini Pesan Rahasia6	3,151
7	10R	3000×3602		252	283	8,764	Ini Pesan Rahasia7	4,382
8	S10R	3000x4500		273	304	10,104	Ini Pesan Rahasia8	5,052
9	11R	3300×4200		297	328	15,897	Ini Pesan Rahasia9	7,949
10	S11R	3300×5100		319	350	17,116	Ini Pesan Rahasia10	8,558

Faktor Exacta *12 (2):* 132-144, 2019
p-ISSN: 1979-276X
e- ISSN: 2502-339X
DOI : 10.30998/faktorexacta.v12i2.3527

PENUTUP

Simpulan

Berdasarkan hasil uji coba simulasi Enkripsi dan Dekripsi Steganografi dengan Metode LSB yang dilakukan sebanyak 10 sampling dengan objek gambar (*image*) berukuran foto yang terdapat pada tabel 3, didapat beberapa simpulan sebagai berikut:

1. Gambar (*image*) yang dihasilkan pada proses enkripsi steganografi dengan metode LSB untuk 10 sampling, dihasilkan gambar (*image*) yang tidak bisa dibedakan oleh indera penglihatan manusia.
2. Ukuran file gambar (KB) yang dihasilkan pada proses enkripsi steganografi dengan metode LSB untuk 10 sampling, dihasilkan persentase peningkatan rata-rata 31,87%, akan tetapi tergantung dari panjang pesan *text* dan kunci (*key*) yang disisipkan (semakin panjang pesan *text* dan *key* yang disisipkan, maka akan semakin besar ukuran file gambar yang dihasilkan).
3. Semakin tinggi resolusi gambar (*pixels*) yang akan di proses untuk simulasi enkripsi dan dekripsi steganografi dengan metode LSB, maka akan semakin lama waktu prosesnya (dalam *second*).
4. Lama waktu proses (*second*) dekripsi steganografi dengan metode LSB lebih singkat dari pada waktu proses enkripsi, terbukti dari hasil uji coba untuk lama waktu proses dekripsi sama dengan 50% dari lama waktu proses enkripsi.
5. Perangkat lunak simulasi Cryptool2 terbukti sangat *powerfull* dalam melakukan proses enkripsi dan dekripsi steganografi dengan metode LSB, terbukti dari hasil uji coba dengan 10 sampling, semua data berhasil di enkripsi dan di dekripsi dengan algoritma steganografi menggunakan metode *Least Significant Bit* (LSB) dengan sangat akurat.

Saran

Berdasarkan hasil dari penelitian yang telah dilakukan, berikut ini beberapa saran untuk pengembangan simulasi dengan perangkat lunak Cryptool2, antara lain:

1. Pada proses enkripsi steganografi dengan metode LSB menggunakan perangkat lunak Cryptool2, diharapkan kedepannya tools simulasi ini mendukung output gambar atau image dalam format *.jpg.
2. Penelitian dalam bentuk simulasi ini diharapkan dapat terus dikembangkan dengan menggabungkan metode steganografi lainnya seperti DCT, *spread spectrum & echo hiding*, atau bahkan bisa di kombinasikan dengan algoritma kriptografi seperti: RSA, AES, TEA dan lainnya.

DAFTAR PUSTAKA

Anti, U. A., Kridalaksana, A. H., & Khairina, D. M. (2017). Steganografi Pada Video Menggunakan Metode Least Significant Bit (LSB) Dan End Of File (EOF). *Informatika Mulawarman : Jurnal Ilmiah Ilmu Komputer*, *12*(2), 104. http://doi.org/10.30872/jim.v12i2.658

Belajar Kriptografi/Enkripsi lebih mudah dg program gratis CrypTool – ebsoft. (n.d.).

CodeSaya | Yuk belajar Kriptografi atau Enkripsi lebih mudah dengan software "CrypTool" (n.d.).

Darwis, D. (2015). Implementasi Steganografi pada Berkas Audio Wav untuk Penyisipan Pesan Gambar Menggunakan Metode Low Bit Coding. *EXPERT*, *5*(1).

Fuad, N., -, S., & Ir. Endang Setyati, M. (2011). Teknik Stenografi dengan Menggunakan Metode Visual Attacks dan Statistical Attacks. *Jurnal Ilmiah Teknologi Informasi Asia*, *5*(2), 28–36.

Gunawan, P. (n.d.). *Studi dan Analisis Mengenai Teknik Steganalisis Terhadap Pengubahan LSB Pada Gambar: Enhanced LSB dan Chi-square*.

Hussein, H. L., Abbass, A. A., Naji, S. A., Al-augby, S., & Lafta, J. H. (2018). Hiding text in gray image using mapping technique. *Journal of Physics: Conference Series*, *1003*, 012032. http://doi.org/10.1088/1742-6596/1003/1/012032

Lestari, T., Nurmaesa, N., & Mariana, A. R. (2017). Aplikasi Steganografi Untuk Menyisipkan Pesan Dalam Media Image. *JURNAL SISFOTEK GLOBAL*, *7*(2).

Reddy Ch, R., & Ramani, R. A. (n.d.). *The Process of Encoding and Decoding of Image Steganography using LSB Algorithm*.

Utami, E. (n.d.). PENDEKATAN METODE LEAST BIT MODIFICATION UNTUK MERANCANG APLIKASI STEGANOGRAPHY PADA FILE AUDIO DIGITAL TIDAK TERKOMPRESI.

Widyawan, T. I. (2018). PENGAMANAN PESAN STEGANOGRAFI DENGAN METODE LSB BERLAPIS ENKRIPSI. *JIK: Jurnal Ilmu Komputer*, *3*(01).

Zinaly, E., & Naghipour, A. (2017). Audio Steganography to Protect the Confidential Information: A Survey. *International Journal of Computer Applications*, *169*(1), 22–29. http://doi.org/10.5120/ijca2017914561