MATHunesa
Jurnal Ilmiah Matematika *Volume 13 No 02*
e-ISSN : 2716-506X | p-ISSN : 2301-9115 *Tahun 2025*

KOMBINASI ALGORITMA KRIPTOGRAFI RC6 DAN STEGANOGRAFI LSB UNTUK PENGAMANAN PESAN TEKS

Ferawati Kai
Program Studi Matematika, Jurusan Matematika, FMIPA, Universitas Negeri Gorontalo
e-mail: ferawati_s1matematika2018@mahasiswa.ung.ac.id

Agusyarif Rezka Nuha
Program Studi Matematika, Jurusan Matematika, FMIPA, Universitas Negeri Gorontalo
e-mail: agusyarif@ung.ac.id

Asriadi
Program Studi Matematika, Jurusan Matematika, FMIPA, Universitas Negeri Gorontalo
e-mail: asriadi@ung.ac.id

Hasan S. Panigoro
Program Studi Matematika, Jurusan Matematika, FMIPA, Universitas Negeri Gorontalo
e-mail: hspanigoro@ung.ac.id

Nisky Imansyah Yahya
Program Studi Matematika, Jurusan Matematika, FMIPA, Universitas Negeri Gorontalo
e-mail: nisky@ung.ac.id*

Armayani Arsal
Program Studi Matematika, Jurusan Matematika, FMIPA, Universitas Negeri Gorontalo
e-mail: armayaniarsal@ung.ac.id

Abstrak

Pesatnya perkembangan teknologi mengakibatkan munculnya berbagai jenis kejahatan teknologi. Salah satu solusi yang digunakan untuk mengamankan pesan yaitu dengan menggunakan teknik kriptografi dan steganografi. Penelitian ini mengombinasikan algoritma kriptografi *Rivest Code* 6 (RC6) dan algoritma steganografi *Least Significant Bit* (LSB) untuk meningkatkan keamanan suatu pesan. Penelitian yang dilakukan meliputi analisis perubahan suatu citra yang telah disisipkan pesan dan dikirimkan melalui aplikasi *WhatsApp*, Telegram, dan *E-mail*. Hasil akhir yang diperoleh adalah citra yang telah disisipkan pesan tidak mengalami perubahan yang signifikan. Adapun *stego image* yang dikirimkan secara langsung pada *WhatsApp* dan Telegram mengalami perubahan ukuran sehingga menyebabkan pesan di dalamnya rusak. Sedangkan, *stego image* yang dikirimkan dalam bentuk dokumen tidak mengalami perubahan ukuran dan pesannya dapat diperoleh secara utuh. Sementara itu, *stego image* yang dikirimkan pada *E-mail*, baik melalui lampiran atau secara langsung tidak mengalami perubahan ukuran, serta pesan dapat diperoleh kembali secara utuh.
Kata Kunci: Kriptografi RC6, Steganografi LSB, Keamanan Pesan Teks

Abstract

The rapid advancement of technology has led to various types of technological crimes. One solution to secure messages is by using cryptographic and steganographic techniques. This research combined the Rivest Code 6 (RC6) cryptographic algorithm and the Least Significant Bit (LSB) steganographic algorithm to enhance message security. The study analyzed changes in an image embedded with a message and sent through applications such a WhatsApp, Telegram, and E-mail. The final result show that the image embedded with the message does not undergo significant changes. However, stego images sent directly via WhatsApp and Telegram experience size alterations, which cause the embedded messages to become corrupted. Meanwhile, stego images sent as documents retain their size, allowing the message to remain intact. Additionally, stego images sent via E-mail, either as attachments or directly, do not experience size changes, and the embedded messages can be fully retrieved.
Keywords: *RC6 Cryptography, LSB Steganography, Text Message Security*

PENDAHULUAN

Komunikasi menjadi salah satu sarana yang memudahkan manusia untuk melakukan aktivitas sehari-hari dalam bertukar informasi dengan orang lain (Kuncoro & Aditama, 2019). Penggunaan teknologi saat ini telah memungkinkan komunikasi tanpa batas dan pertukaran informasi dapat dilakukan pada berbagai perangkat yang terhubung ke internet. Namun, pesatnya perkembangan teknologi mengakibatkan munculnya berbagai jenis kejahatan teknologi yang dikenal dengan penyadapan, pencurian data, dan juga perusakan data atau informasi. Sehingga, tidak menutup kemungkinan bahwa setiap pesan ataupun data yang dikirimkan oleh suatu pihak akan tersebar secara bebas kepada pihak (Rismawati & Mulya, 2019).

Melihat berbagai fenomena kejahatan teknologi yang terjadi, dibutuhkan sistem keamanan untuk menjaga kerahasiaaan informasi atau pesan yang akan dikirimkan. Oleh karena itu, suatu ilmu pengetahuan mengenai teknik mengamankan dan menyembunyikan pesan terus dikembangkan. Teknik mengamankan pesan disebut sebagai kriptografi yang dapat mengubah pesan teks menjadi pesan acak menggunakan alrogitma yang ada. Agar pesan acak tersebut tidak dicurigai oleh pihak lain, maka pesan akan disembunyikan dalam sebuah citra menggunakan teknik steganografi (Kuncoro & Aditama, 2019). Kombinasi antara kriptografi dan steganografi telah digunakan oleh beberapa peneliti untuk mengamankan pesan ataupun informasi lainnya.

Seperti pada penelitian yang dilakukan oleh Sulaiman & Isnanto (2018), digunakan kombinasi kriptografi RC4 dan steganografi LSB pada sebuah file JPEG untuk meningkatkan keamanan pesan. Penelitian ini melakukan analisis terhadap waktu proses pada berbagai jumlah karakter, dan juga menganalisis perubahan ukuran file setelah disispkan pesan. Pada penelitian Kuncoro & Aditama (2019), mereka mengombinasikan kriptografi RSA dan steganografi LSB dalam pengamanan pesan teks. Penelitian ini juga menganalisis waktu proses, nilai PSNR, dan ketahanan pesan pada *stego image*. Pada penelitian lain, Tena et al. (2019) menggunakan kombinasi kriptografi RC6 dan steganografi LSB untuk keamanan citra digital. Penelitian ini melakukan pengujian ketahanan kombinasi RC6 dan LSB terhadap *noise with salt and pepper*, waktu proses pada dua buah citra, serta presentase keberhasilan proses ektraksi dan dekripsi.

Algoritma kriptografi memiliki kunci simetri dan juga nirsimetri. Salah satu algoritma kriptografi yang menggunakan kunci simetri adalah *Rivest Code* 6 (RC6). RC6 merupakan salah satu kandidat dari AES (*Advanced Encryption Standard*) yang dikembangkan oleh Ronald L. Rivest, M.J.B. Robshaw, Y.L. Yin dan R. Sidney, kemudian diserahkan oleh Laboratorium RSA ke NIST (*National Intitute of Standards and Technology*) (Rivest et al., 1998). Kriptografi sendiri memiliki teknik enkripsi dan dekripsi pesan.

Sementara itu, steganografi memiliki metode dengan algortima yang sederhana dan mudah diimplementasikan yaitu *Least Significant Beat* (LSB). LSB melakukan konversi terhadap setiap pesan ke dalam bentuk biner sebelum akhirnya pesan tersebut akan disembunyikan pada lokasi bit terendah dalam suatu citra digital (Sulaiman & Isnanto, 2018).

Berdasarkan uraian di atas, pada penelitian ini algoritma enkripsi dan dekripsi RC6 akan dikombinasikan dengan algoritma steganografi LSB untuk menjaga keamanan dan kerahasiaan suatu pesan. Adanya kombinasi kedua algoritma tersebut bertujuan meningkatkan kerahasiaan pesan yang akan dikirimkan melalui beberapa aplikasi, sehingga tidak mudah dideteksi oleh pihak lain.

KAJIAN TEORI

KRIPTOGRAFI

Kriptografi terbagi atas dua suku kata dari Bahasa Yunani yaitu '*cryptos*' yang berarti rahasia dan '*graphien*' yang berarti tulisan. Secara harfiah kriptografi dapat diartikan sebagai tulisan rahasia (Munir, 2019). Dalam kriptografi terdapat beberapa istilah penting yang perlu diketahui, yaitu plainteks, cipherteks, enkripsi, dekripsi, dan kunci. Plainteks merupakan pesan atau informasi yang mudah dibaca dan dipahami, sedangkan cipherteks adalah pesan yang telah terenkripsi atau pesan acak yang maknanya tidak dapat dipahami. Sementara enkripsi adalah proses untuk menyandikan plainteks menjadi cipherteks, dan dekripsi digunakan untuk mengubah kembali cipherteks menjadi plainteks. Adapun kunci adalah parameter yang sering digunakan dalam setiap proses enkripsi dan dekripsi.

RIVEST CODE 6 (RC6)

Rivest Code 6 atau disebut juga RC6 merupakan algoritma yang dikembangkan dari RC5 dan telah memenuhi standar yang ditetapkan oleh NIST (Rivest et al., 1998). Berdasarkan hasil analisis pada RC5, ditemukan bahwa jumlah putaran yang terjadi tidak sepenuhnya bergantung pada data di dalam blok. Oleh karena itu, algoritma RC6 dikembangkan sebagai langkah untuk mengatasi kekurangan yang ditemukan pada RC5 (Chaniago & Manurung, 2021). Algoritma RC6 memiliki beberapa parameter yang ditentukan dengan notasi RC6-$w/r/b$. Dalam hal ini, w dinyatakan sebagai panjang kata dalam satuan bit, r sebagai jumlah putaran, dan b menyatakan ukuran atau panjang kunci.

RC6 memiliki tiga proses penyandian yaitu, pembentukan kunci internal, enkripsi, dan dekripsi.

1. Pembentukan Kunci Internal

   Pada pembentukan kunci terdapat tiga tahapan yang diantaranya adalah penempatan kunci ke dalam *array* L, inisialisasi kunci S menggunakan *magic constanta*, dan kombinasi *array* L dan S (Muharini, 2012). Penempatan kunci ke dalam *array* L terlebih dahulu akan mengkonversi kunci privat *K* dari bentuk *array of bytes* menjadi *array of words* $L[0,1,\dots,c-1]$ (Munir, 2019). Jumlah *byte* dalam satu *word* dinyatakan dengan c, di mana

   $$c = \frac{b}{u}$$

   dengan

   $$u = \frac{w}{8}$$

   Selanjutnya untuk inisialisasi kunci S menggunakan algoritma berikut:

   $$S[0] = P_w$$

   Untuk $i = 1$ sampai $(2r+3)$

   $$S[i] = S[i-1] + Q_w$$

   Tahap terakhir yaitu kombinasi *array* L dan S digunakan algoritma berikut:

   $$A = B = i = j = 0$$

   Untuk $i = 3 \times \max(c, r)$

   $$S[i] = (S[i] + A + B) \lll 3$$
   $$A = S[i]$$
   $$L[i] = (L[i] + A + B) \lll (A + B)$$
   $$B = L[i]$$
   $$i = (i+1) \bmod 2r + 3$$
   $$j = (j+1) \bmod c$$

2. Enkripsi

   Pada proses enkripsi diawali dan diakhiri dengan tahapan *whitening*, tujuannya untuk menyembunyikan proses enkripsi pada iterasi pertama serta iterasi terakhir (Mubarak, 2020). Algoritma yang digunakan yaitu:

   $$B = B + S[0]$$
   $$D = D + S[1]$$

   Untuk $i = 1$ sampai r

   $$p = \left(B(2B+1)\right) \lll \lg w$$
   $$q = \left(D(2D+1)\right) \lll \lg w$$
   $$A = \left((A \oplus p) \lll q\right) + S[2i]$$
   $$C = \left((C \oplus q) \lll p\right) + S[2i+1]$$

   $$A = A + S[2r+2]$$
   $$C = C + S[2r+3]$$

   $$A, B, C, D = B, C, D, A$$

3. Dekripsi

   Sama seperti enkripsi, proses dekripsi juga diawali dan diakhiri dengan *whitening*. Proses dekripsi memiliki langkah yang berlawanan dengan proses enkripsi (Mubarak, 2020).

   $$C = C - S[2r+3]$$
   $$A = A - S[2r+2]$$

   $$A, B, C, D = D, A, B, C$$

   Untuk $i = r$ sampai 1

   $$p = \left(B(2B+1)\right) \lll \lg w$$
   $$q = \left(D(2D+1)\right) \lll \lg w$$
   $$C = \left((C - S[2i+1]) \ggg p\right) \oplus q$$
   $$A = \left((A - S[2i]) \ggg q\right) \oplus p$$

   $$D = D - S[1]$$
   $$B = B - S[0]$$

Keterangan:

a) $A + B$ adalah operasi penjumlahan bilangan bulat dalam modulo 2^w.
b) $A - B$ adalah operasi pengurangan bilangan bulat dalam modulo 2^w.
c) $A \oplus B$ adalah operasi XOR dengan panjang w bit.
d) $A \times B$ adalah operasi perkalian bilangan bulat dalam modulo 2^w.
e) $A \lll B$ adalah rotasi A (dengan panjang w bit) secara sirkular ke kiri sejauh $\lg w$ *least significant bit* dari B.
f) $A \ggg B$ adalah rotasi A (dengan panjang w bit) secara sirkular ke kanan sejauh $\lg w$ *least significant bit* dari B.

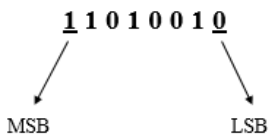
STEGANOGRAFI

Steganografi berasal dari Bahasa Yunani yaitu '*steganos*' yang berarti tersembunyi, dan '*graphein*' yang berarti tulisan. Sehingga, dapat diartikan bahwa steganografi merupakan tulisan tersembunyi (Munir, 2019). Steganografi memiliki dua properti utama, yaitu tempat atau media penampung pesan (*cover*) dan pesan rahasia (Hafis, 2019). Sama seperti kriptografi, steganografi juga memiliki beberapa istilah diantaranya adalah *embedded message* yang merupakan pesan tersembunyi baik dalam bentuk teks, gambar (citra), suara, video, dan lain sebagainya. Kemudian ada istilah *cover-object* yang merupakan tempat penampung pesan dan biasanya digunakan untuk menyembunyikan pesan rahasia. *Cover* dalam bentuk gambar disebut *cover-image*, dalam bentuk suara disebut *cover-audio*, dan seterusnya. Selanjutnya adalah *stego-object*, istilah untuk *object* yang sudah berisi pesan rahasia. Jika dalam bentuk gambar disebut *stego-image*, dalam bentuk suara disebut *stego-audio*, dan seterusnya (Munir, 2019).

LEAST SIGNIFICANT BIT (LSB)

Metode LSB berasal dari angka yang paling kurang signifikan dari jumlah bit di dalam 1 *byte*. Dengan adanya keterbatasan dalam indera penglihatan manusia, maka digunakan metode tersebut karena sulit menemukan perbedaan antara gambar asli dengan gambar yang telah disisipkan pesan rahasia (Rismawati & Mulya, 2019). Dalam LSB memiliki dua proses yaitu *embedding* atau penyisipan pesan ke dalam citra, dan *extracting* atau pengembalian pesan ke bentuk semula.

LSB merupakan bagian dari barisan data biner (basis dua) dengan nilai yang paling kurang berarti dan terletak pada bagian kanan dari susunan bit yang ada. Sedangkan kebalikan dari LSB yaitu *Most Significant Bit* (MSB) yang memiliki angka paling berarti dan terletak pada bagian kiri dari susunan bit yang ada (Hafis, 2019).



Gambar 1. MSB dan LSB

METODE PENELITIAN

Metode yang digunakan yaitu studi literatur, dengan menelusuri jurnal ilmiah, artikel, buku, dan referensi terkait dengan penelitian. Penelitian ini menggunakan *software* Matlab 2017b untuk melakukan proses pembangkitan kunci RC6, mengenkripsi pesan teks, menyisipkan pesan (*embedding*) ke dalam *cover-image*, mengekstraksi pesan yang berada dalam *stego-image*, dan mendekripsi pesan. Penelitian ini menggunakan media gambar (*cover image*) untuk menampung pesan rahasia, dan aplikasi *WhatsApp*, Telegram, dan *E-mail* untuk mengirimkan *stego-image*.

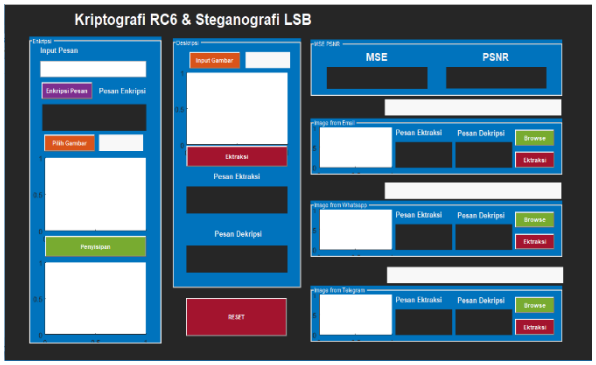
Berikut merupakan prosedur analisis yang akan digunakan dalam penelitian:

1. Tahapan awal penelitian yaitu melakukan penelusuran beberapa referensi yang berkaitan dengan penelitian.
2. Setelah melakukan penelusuran pada beberapa jurnal ilmiah, arikel ataupun buku, diperoleh bahwa sebelum melalukan enkripsi dan dekripsi RC6, akan dilakukan pembentukan kunci terlebih dahulu.
3. Kemudian, ketika kunci selesai dibentuk, akan dilakukan proses enkripsi menggunakan algoritma RC6. Setelah pesan dienkripsi, maka *ciphertext* akan disisipkan ke dalam sebuah gambar/citra menggunakan metode LSB.
4. Setelah *ciphertext* dimasukkan ke dalam *cover-image*, akan dianalisis apakah terjadi perubahan terhadap *stego-image*.
5. Selanjutnya, pesan yang berada di dalam *stego-image* akan diekstraksi menggunakan metode LSB, dan kemudian akan didekripsi menggunakan algoritma RC6 untuk mendapatkan kembali pesan asli.
6. Setelah memastikan bahwa pesan dapat diperoleh kembali, *stego-image* akan dikirimkan melalui *WhatsApp*, Telegram, dan *Email*. *Stego-image* tersebut kemudian akan diunduh dan kemudian dianalisis apakah pesan di dalamnya mengalami kerusakan atau tidak.

HASIL DAN PEMBAHASAN

GUI MATLAB YANG AKAN DIGUNAKAN

Berikut merupakan tampilan GUI Matlab yang akan digunakan untuk mengenkripsi pesan teks, menyisipkan pesan (*embedding*) ke dalam *cover-image*, mengekstraksi pesan yang berada dalam *stego-image*, dan mendekripsi pesan. Selain itu, rancangan tersebut juga digunakan untuk menganalisis pesan yang terdapat dalam *stego image* setelah dikirimkan melalui *WhatsApp*, Telegram, dan *E-mail*.

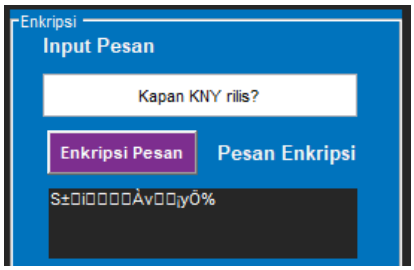


Gambar 2. Tampilan Keseluruhan GUI Matlab

PROSES ENKRIPSI-*EMBEDDING* PESAN

Langkah-langkah dalam menjalankan proses enkripsi-*embedding* yaitu sebagai berikut:

1. Saat program dijalankan, akan muncul tampilan GUI Matlab seperti pada Gambar 2.
2. Selanjutnya, masukkan pesan pada kolom yang tersedia dan klik 'Enkripsi Pesan'. Perhatikan gambar berikut.

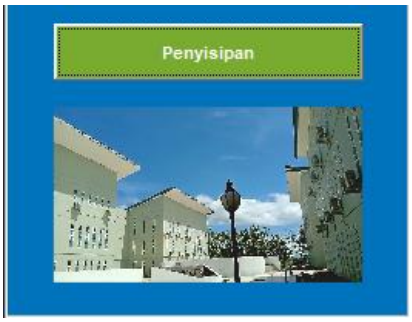


Gambar 3. Proses Enkripsi Pesan

3. Setelah mendapatkan *ciphertext*, masukkan *cover-image* yang akan digunakan untuk menyisipkan pesan.

Gambar 4. Proses Memasukkan *Cover-Image*

4. Setelah dimasukkan, klik 'penyisipan' untuk menyisipkan *ciphertext*.



Gambar 5. Proses Penyisipan *Ciphertext*

ANALISIS PERUBAHAN CITRA SETELAH DISISIPKAN PESAN

Setelah dianalisis, perbedaan *cover-image* dan *stego-image* dapat dilihat pada perubahan ukuran yang terjadi. Pada awalnya, *cover-image* berukuran 120,9 KB. Namun, begitu disisipkan *ciphertext*, *stego-image* yang dihasilkan berukuran 1,3 MB. Hal tersebut disebabkan oleh penambahan bit pada citra.

Gambar a. *Cover-Image* Gambar b. *Stego-Image*

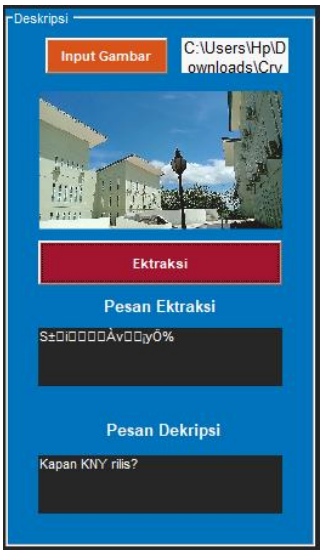
Gambar 6. Perbandingan Dua Buah Citra

Pada gambar di atas menunjukkan perbandingan antara *cover-image* dan *stego-image*. Dapat dilihat bahwa *stego-image* tidak mengalami perubahan yang signifikan meskipun telah disisipkan *chiphertext*. Selain itu, kualitas *stego-image* yang dihasilkan dapat dilihat pada nilai MSE dan PSNR-nya. Hasil perhitungan memperoleh nilai MSE sebesar 0.000147059 dan PSNR sebesar 86.4559 dB. Dalam hal ini, semakin kecil nilai MSE, maka tingkat kemiripan antara *cover-image* dan *stego-image* semakin tinggi. Sedangkan, suatu citra dengan nilai PSNR ≥ 30 memiliki kualitas yang baik, dan jika nilai PSNR < 30 maka kualitas citra telah terdegradasi secara signifikan.

PROSES EKSTRAKSI-DEKRIPSI PESAN

Setelah pesan disisipkan dengan baik, akan dilakukan ekstraksi-dekripsi untuk memastikan bahwa pesan dapat dikembalikan ke bentuk semula (*plaintext*). Langkah-langkah dalam menjalankan proses ekstraksi-dekripsi sebagai berikut:

1. Masukkan *stego-image* terlebih dahulu pada kolom yang tersedia.
2. Setelah *stego-image* dimasukkan, klik bagian 'ekstraksi' untuk mendapatkan pesan yang telah disisipkan.
3. Pesan secara otomatis akan terdekripsi dan *plaintext* pun dapat diperoleh kembali.



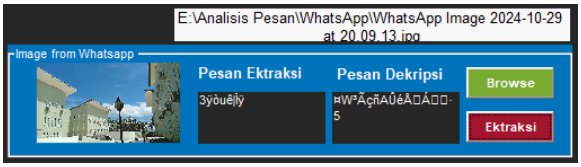
Gambar 7. Proses Ekstraksi-Dekripsi

ANALISIS PENGIRIMAN *STEGO-IMAGE*

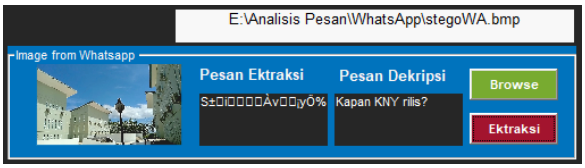
Setelah pesan pada *stego-image* dipastikan dapat diperoleh kembali secara utuh, maka pada tahap ini *stego-image* akan dikirimkan melalui *WhatsApp*, Telegram, dan *E-mail*. Kemudian, *stego-image* yang dikirimkan melalui ketiga aplikasi tersebut akan dianalisis untuk mengetahui pesan di dalamnya dapat diperoleh secara utuh atau tidak.

ANALISIS PESAN YANG DIKIRIMKAN MELALUI WHATSSAPP

Pada tahap ini, *stego-image* akan dikirimkan dalam dua cara yaitu secara langsung dan bentuk dokumen. Begitu *stego-image* dikirimkan, *stego-image* akan diunduh kembali sebelum dianalisis. Hasil unduhan menunjukkan bahwa *stego-image* yang dikirimkan secara langsung mengalami perubahan ukuran yang cukup besar. *Stego-image* yang awalnya berukuran 1,3 MB berubah menjadi 64,2 KB. Sedangkan *stego-image* yang dikirimkan dalam bentuk dokumen tidak mengalami perubahan ukuran. Selanjutnya, *stego-image* hasil unduhan akan diekstraksi untuk mendapatkan pesan di dalamnya. Perhatikan gambar berikut:



Gambar 8. Ekstraksi Pesan dari *Stego* yang Dikirimkan Langsung (*WhatsApp*)

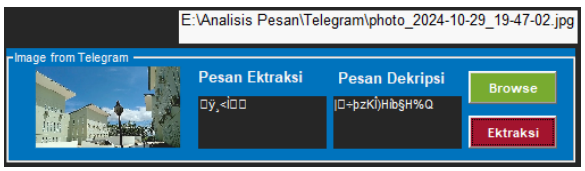


Gambar 9. Ekstraksi Pesan dari *Stego* yang Dikirimkan Bentuk Dokumen (*WhatsApp*)

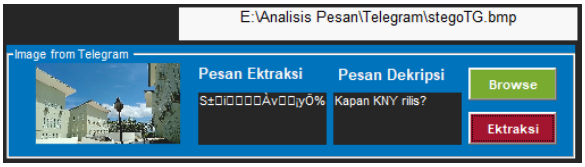
Gambar 8 merupakan ekstraksi pesan pada *stego-image* yang dikirimkan secara langsung di *WhatsApp*. Hasil ekstraksi menunjukkan bahwa pesan yang berada di dalamnya mengalami kerusakan dan tidak dapat diperoleh kembali secara utuh. Sementara itu pada Gambar 9, pesan dalam *stego-image* yang dikirim bentuk dokumen tidak mengalami kerusakan dan dapat diperoleh kembali secara utuh. Hal ini menyatakan bahwa, perubahan ukuran yang signifikan berpengaruh terhadap pesan yang berada dalam *stego-image*.

ANALISIS PESAN YANG DIKIRIMKAN MELALUI TELEGRAM

Seperti pada analisis di atas, pesan yang akan dikirimkan melalui Telegram juga diberikan perlakuan yang sama. *Stego-image* akan dikirimkan secara langsung dan bentuk dokumen. Setelah *stego-image* diunduh, diperoleh bahwa *stego-image* yang dikirimkan secara langsung mengalami perubahan ukuran yang cukup besar. *Stego-image* yang awalnya berukuran 1,3 MB berubah menjadi 87,2 KB. Sedangkan *stego-image* yang dikirimkan dalam bentuk dokumen tidak mengalami perubahan ukuran. Selanjutnya, *stego-image* hasil unduhan akan diekstraksi untuk mendapatkan pesan di dalamnya.



Gambar 10. Ekstraksi Pesan dari *Stego* yang Dikirimkan Langsung (Telegram)

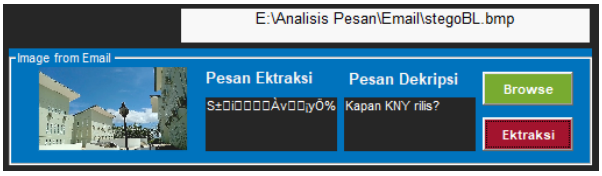


Gambar 11. Ekstraksi Pesan dari *Stego* yang Dikirimkan Bentuk Dokumen (Telegram)

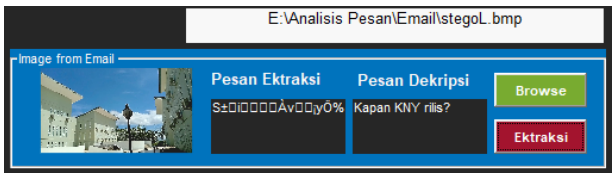
Setelah dilakukan analisis, hasil yang ditunjukkan tak jauh berbeda dengan hasil analisis pada *WhatsApp*. Pesan pada *stego-image* yang dikirimkan secara langsung dan terjadi perubahan ukuran cukup besar mengalami kerusakan dan tidak dapat diperoleh kembali. Adapun pesan yang berada dalam *stego-image* dan dikirimkan bentuk dokumen tidak mengalami kerusakan sehingga pesan dapat diperoleh secara utuh.

ANALISIS PESAN YANG DIKIRIMKAN MELALUI E-MAIL

Pengiriman melalui *E-mail* dilakukan dengan cara dilampirkan dan tidak dilampirkan. Begitu *stego-image* diunduh, baik yang dilampirkan ataupun tidak dilampirkan, tidak mengalami perubahan ukuran. Selanjutnya akan dianalisis apakah pesan di dalamnya dapat diperoleh kembali secara utuh atau tidak.



Gambar 12. Ekstraksi Pesan dari *Stego* yang Tidak Dilampirkan (*E-mail*)



Gambar 13. Ekstraksi Pesan dari *Stego* yang Dilampirkan (*E-mail*)

Hasil analisis menunjukkan bahwa pesan yang berada pada *stego-image* baik yang dilampirkan ataupun tidak dilampirkan dapat diperoleh kembali secara utuh dan tidak mengalami kerusakan.

PENUTUP

SIMPULAN

Hasil penelitian menunjukkan bahwa kombinasi algoritma kriptografi RC6 dan steganografi LSB yang dibuat pada GUI Matlab dapat dijalankan dengan baik. Berdasarkan hasil analisis, didapatkan beberapa kesimpulan yaitu:

1. Algoritma kriptografi RC6 secara efektif dapat diimplementasikan untuk mengenkripsi dan mendekripsikan pesan. Meskipun memiliki struktur yang sederhana, RC6 mampu memberikan keamanan yang kuat dalam melindungi pesan.
2. Algoritma steganografi LSB juga dapat diimplementasikan dengan baik untuk menyisipkan dan mengekstraksi pesan dalam citra digital. LSB mampu menyembunyikan pesan ke dalam citra tanpa mengubah kualitas visual citra secara signifikan.
3. *Cover-image* dan *stego-image* tidak bisa dibedakan oleh indra penglihatan manusia karena perubahannya tidak signifikan. Perubahan yang terlihat hanya terletak pada ukuran citra. Adapun *stego-image* yang dikirimkan secara langsung melalui Telegram dan *WhatsApp* mengalami perubahan ukuran citra yang sangat signifikan hingga menyebabkan kerusakan pada pesan. Sedangkan, *stego-image* yang dikirim tanpa dilampirkan pada *E-mail* tidak mengalami perubahan ukuran dan pesan dapat diperoleh kembali secara utuh. Sementara itu, *stego-image* yang dikirimkan dalam bentuk dokumen melalui Telegram dan *WhatsApp* serta dilampirkan pada *E-mail* tidak mengalami perubahan ukuran citra. Pesan yang disisipkan ke dalam citra dapat diperoleh kembali secara utuh.

SARAN

1. Pada penelitian selanjutnya dapat menggunakan aplikasi \textit{chatting} lain dan cobalah mengirimkan pesannya secara langsung untuk melihat apakah pesan itu akan rusak atau tidak.
2. Media untuk menyisipkan pesan tidak hanya citra saja. Jadi, penelitian selanjutnya dapat menggunakan media lain seperti audio dan video.

DAFTAR PUSTAKA

Chaniago, R., & Manurung, J. (2021). *Pengembangan Penyandian Dalam Aplikasi Pada Kriptografi Dengan Menggunakan Metode RC6 Berbasis Web*. *2*, 46–51.

Hafis, A. (2019). Steganografi Berbasis Citra Digital untuk Menyembunyikan Data Menggunakan Metode Least Significant Bit (LSB). *Jurnal Cendikia*, *XVII*(April), 194–198.

Kuncoro, T. R., & Aditama, R. (2019). Analisis Kombinasi Algoritma Kriptografi Rsa Dan Algoritma Steganografi Least Significant Bit (Lsb) Dalam Pengamanan Pesan Digital. *Statmat : Jurnal Statistika Dan Matematika*, *1*(2), 60–82.

Mubarak, R. (2020). Implementasi Sistem Keamanan Data Berbasis Kriptografi Rivest Code 6 , Vigenere Chipper dan Kompresi Data LZW. *Insan Pembangunan Sistem Informasi Dan Komputer (IPSIKOM)*, *6*(D).

Muharini, A. (2012). *Aplikasi Algoritma Rivest Code 6 Dalam Pengamanan Citra Digital*.

Munir, R. (2019). *Kriptografi Edisi Kedua* (2019th ed.). Informatika Bandung.

Rismawati, N., & Mulya, M. F. (2019). Analisis dan Perancangan Simulasi Enkripsi dan Dekripsi pada Algoritma Steganografi untuk Penyisipan Pesan Text pada Image menggunakan Metode Least Significant Bit (LSB) Berbasis Cryptool2. *Faktor Exacta*, *12*(2), 132. https://doi.org/10.30998/faktorexacta.v12i2.3527

Rivest, R., Robshaw, M. J. B., Sidney, R., & Yin, L. Y. (1998). *The RC6 Block Cipher*.

Sulaiman, R., & Isnanto, B. (2018). 'Peningkatan Keamanan Pesan Dengan Kriptografi RC4 dan Steganografi LSB Pada File JPEG''.' *Konferensi Nasional Sistem Informasi 2018*, 8–9.

Tena, S., Pella, S. I., Mooy, B. J., Code, R., Bit, S., & Rc, L. S. B. (2019). *Implementasi Algoritma Rivest Code 6 (RC6) dan Steganografi Least Significant Bit (LSB) untuk Keamanan Data Citra Digital*. *VIII*(2), 110–116.