**Jurnal Quancom**
Vol.3, No.1, Juni 2025, pp. 25~31
e-ISSN: 2988-0998
P-ISSN: 2988-1005
DOI: https://doi.org/10.62375/jqc.v3i1.445

# Implementasi Keamanan Perangkat Lunak Menggunakan Algoritma Kriptografi dan Steganografi berbasis Web

**Rivael Sagala [1], Alvina Siallagan [2], Emalia Telaumbanua [3s]**
[123] Sarjana Terapan Teknologi Rekayasa Perangkat Lunak, Fakultas Vokasi, Institut Teknologi Del
rivaelsagala901@gmail.com[1], alvinalrsiallagan@gmail.com[2], liaema17@gmail.com[3]

**Article Info**

***Article history:***

Received 1 Januari 2025
Revised 9 Januari 2025
Accepted 11 Januari 2025

***Keyword:***

Security, Cryptography, Steganography, Advanced Encryption, Least Significant Bit

**ABSTRACT**

Software security has become very important in today's digital age, especially in protecting sensitive data that is often the target of cyberattacks. This research proposes a combination of Advanced Encryption Standard (AES) cryptographic algorithm and Least Significant Bit (LSB) steganography to enhance data security. By encrypting data using AES and hiding it in an image through the LSB method, this system not only protects the content of the data but also reduces the possibility of detection by third parties. The research method includes literature analysis, system prototype development, and testing to evaluate the effectiveness of the implementation. The test results show that the number of message characters has a significant effect on the image size after encoding and the execution time. The more characters inserted, the image size increases proportionally, and the execution time for encryption and data insertion also increases on average by 15% for every additional 100 message characters. This research is expected to contribute to the development of information security technology and practical solutions for software developers in improving application security.



## PENDAHULUAN

Dalam dunia digital saat ini, keamanan perangkat lunak menjadi aspek yang sangat penting, terutama dengan semakin banyaknya data sensitif yang dikelola dan ditransmisikan secara online. Data tersebut, yang meliputi informasi pribadi, data keuangan, dan rahasia bisnis, sering kali menjadi target serangan siber. Oleh karena itu, diperlukan pendekatan yang efektif untuk melindungi data tersebut. Salah satu solusi yang menjanjikan adalah implementasi keamanan perangkat lunak menggunakan algoritma kriptografi dan steganografi berbasis web, khususnya melalui penggunaan AES (*Advanced Encryption Standard*) untuk kriptografi dan LSB (*Least Significant Bit*) untuk steganografi.[1]

Meskipun algoritma kriptografi seperti AES sangat efektif dalam mengenkripsi data, ia memiliki kelemahan yang signifikan. Keberadaan data yang terenkripsi masih dapat terdeteksi oleh pihak ketiga, dan jika kunci enkripsi jatuh ke tangan yang salah, data tersebut dapat diakses dengan mudah. Di sisi lain, steganografi, yang menyembunyikan informasi dalam media lain (seperti gambar atau audio), juga memiliki keterbatasan. Metode ini dapat diungkap melalui teknik analisis forensik yang canggih, sehingga informasi yang disembunyikan dapat terdeteksi jika pola penyembunyian terlalu jelas. [2]

Menggabungkan algoritma kriptografi dan steganografi dapat menciptakan solusi yang lebih kuat untuk melindungi data. Dengan mengenkripsi data menggunakan AES dan kemudian menyembunyikannya menggunakan LSB, kita tidak hanya melindungi isi data tetapi juga mengurangi kemungkinan deteksi oleh pihak ketiga. Masalah utama yang dihadapi adalah tingginya risiko kebocoran data sensitif akibat serangan siber yang terus meningkat, di mana banyak sistem hanya mengandalkan satu pendekatan keamanan, sehingga rentan terhadap berbagai jenis ancaman. [3]

Metode yang digunakan dalam penelitian ini mencakup analisis literatur mengenai algoritma AES dan LSB, pengembangan prototipe sistem yang menggabungkan kedua teknik, serta pengujian untuk mengevaluasi tingkat keamanan dan efektivitas dari implementasi yang diusulkan. Dengan

pendekatan ini, diharapkan dapat memberikan wawasan yang lebih mendalam tentang pengembangan sistem keamanan yang komprehensif.[4]

Jurnal oleh Rian Arifin dan Lucky Tri Oktoviana (2013) membahas penerapan teknik kriptografi dan steganografi untuk melindungi kerahasiaan pesan, menggunakan algoritma RSA untuk enkripsi dan metode LSB (*Least Significant Bit*) untuk penyisipan pesan dalam file gambar bitmap 24-bit. Kelebihan dari metode ini termasuk keamanan ganda dan kemudahan implementasi, serta kemampuan untuk menyisipkan pesan tanpa mengubah kualitas visual gambar. Namun, kelemahan yang signifikan adalah kerentanan metode LSB terhadap serangan yang dapat mengekstrak pesan, serta ketergantungan pada kekuatan kunci RSA yang harus tetap rahasia. Penelitian ini menunjukkan potensi kombinasi kriptografi dan steganografi, meskipun perlu adanya pengembangan lebih lanjut untuk mengatasi tantangan keamanan yang ada.[5]

Jurnal oleh Ajar Rohmanu (2017) ini membahas penerapan kriptografi dan steganografi menggunakan algoritma *Data Encryption Standard* (DES) dan metode *End of File* (EoF) untuk menyembunyikan pesan rahasia dalam media audio WAV. Kelebihan penelitian ini meliputi pendekatan terpadu yang meningkatkan keamanan data dan penggunaan algoritma DES yang telah terbukti efektif. Namun, kekurangan mencakup kerentanan DES terhadap serangan brute force, keterbatasan metode EoF dalam menyembunyikan data, serta kurangnya pengujian yang luas dan analisis performa. Meskipun demikian, jurnal ini memberikan kontribusi penting dalam bidang keamanan data dan wawasan yang berguna bagi penelitian selanjutnya.[6]

Tujuan penelitian ini adalah untuk mengembangkan dan menganalisis implementasi keamanan perangkat lunak yang mengintegrasikan algoritma AES dan LSB dalam konteks aplikasi berbasis web. Penelitian ini bertujuan untuk mengevaluasi efektivitas kombinasi kedua teknik dalam melindungi data sensitif dan memberikan solusi praktis bagi pengembang perangkat lunak dalam meningkatkan keamanan aplikasi .[7]

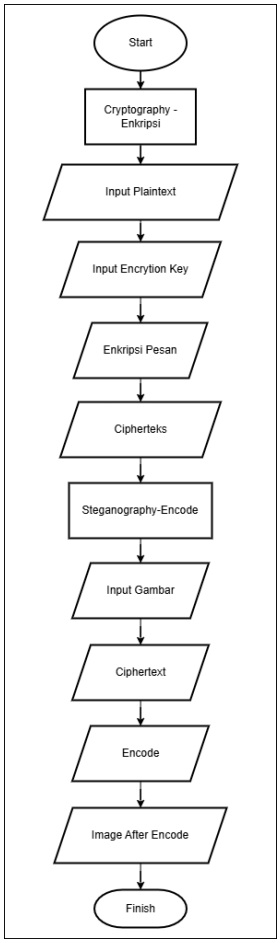
Manfaat dari penelitian ini adalah memberikan kontribusi signifikan terhadap pengembangan teknologi keamanan informasi, serta menyediakan panduan bagi pengembang perangkat lunak dalam menerapkan langkah-langkah keamanan yang lebih efektif. Dengan demikian, penelitian ini diharapkan dapat meningkatkan kesadaran akan pentingnya keamanan data di era digital dan memberikan solusi yang dapat diimplementasikan secara praktis.[8]

## METODE

Penelitian ini menggabungkan metode kriptografi dan steganografi. Algoritma kriptografi yang digunakan adalah Advanced Encryption Standard (AES), sedangkan steganografi memanfaatkan metode Least Significant Bit (LSB). [9]Sistem yang dirancang berfungsi untuk mengenkripsi pesan teks (*plaintext*) yang diinputkan oleh pengguna, kemudian melakukan proses encoding terhadap pesan yang telah terenkripsi (*ciphertext*) ke dalam sebuah gambar. [10]Selain itu, sistem ini juga mampu mengekstrak kembali pesan terenkripsi dari gambar melalui proses *decoding*, serta mendekripsi pesan tersebut menjadi pesan asli (*plaintext*). Proses penggabungan kedua algoritma ini bekerja melalui dua tahap utama :

### *A. Proses Enkripsi dan Encoding Pesan*

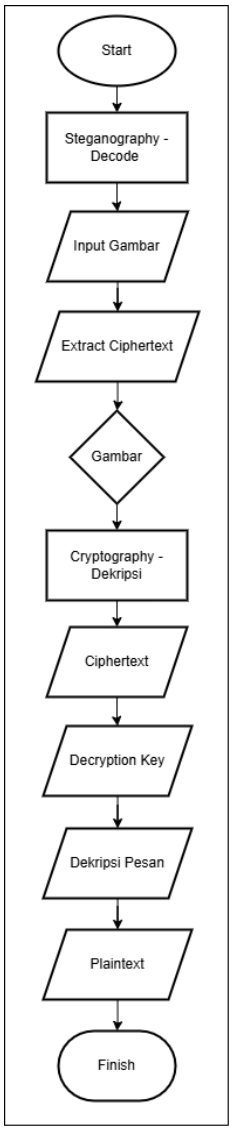
Pada proses enkripsi dan *encoding* pesan dimulai dengan melakukan enkripsi terhadap *plaintext* (pesan asli) menggunakan algoritma kriptografi. [11]Pengguna memasukkan *plaintext* yang akan dienkripsi serta kunci enkripsi yang akan digunakan. Sistem kemudian akan melakukan proses enkripsi, menghasilkan *ciphertext* (pesan terenkripsi). Selanjutnya, *ciphertext* tersebut disisipkan ke dalam gambar menggunakan teknik steganografi. [12] [13] Pengguna juga diminta untuk memasukkan gambar yang akan digunakan untuk menyisipkan *ciphertext*. Alur proses ini akan dijelaskan melalui diagram alur (*flowchart*) pada Gambar 1.




Gambar 1. Diagram proses enkripsi dan encode

*B. Proses Decoding dan Dekripsi Pesan*

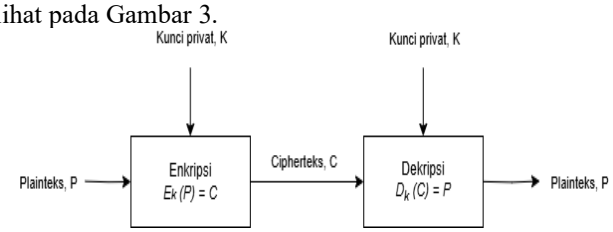
Pada proses decoding dan dekripsi pesan, pengguna memasukkan gambar yang telah disisipkan pesan rahasia. Sistem akan mengekstrak *ciphertext* (pesan rahasia) dari dalam gambar tersebut. Kemudian, pengguna diminta untuk memasukkan kunci dekripsi yang sesuai dengan kunci enkripsi yang digunakan sebelumnya. Setelah itu, sistem akan melakukan proses dekripsi terhadap *ciphertext* menggunakan kunci dekripsi yang dimasukkan, sehingga menghasilkan *plaintext* (pesan asli) yang semula telah dienkripsi. Pesan asli yang telah berhasil diekstrak dan didekripsi akan ditampilkan kepada pengguna. Alur proses ini akan dijelaskan melalui diagram alur (*flowchart*) pada Gambar 2.




Gambar 2. Proses decoding dan dekripsi pesan

## HASIL DAN PEMBAHASAN

Pada penelitian ini, diimplementasikan sebuah sistem keamanan perangkat lunak berbasis web yang mengintegrasikan algoritma kriptografi *Advanced Encryption Standard* (AES) dan metode steganografi *Least Significant Bit* (LSB). Algoritma AES digunakan sebagai dasar kriptografi untuk memberikan tingkat keamanan tinggi karena telah terbukti andal dalam berbagai implementasi keamanan data. [14]AES (*Advanced Encryption Standard*) adalah algoritma kriptografi simetris yang digunakan untuk mengenkripsi dan mendekripsi data. AES menggunakan kunci dengan panjang 128, 192, atau 256 bit, dan beroperasi pada blok data berukuran 128 bit. Proses enkripsi dan dekripsi terdiri dari beberapa transformasi yang dapat dibalikkan. Skema proses dari kriptografi dengan algoritma AES dapat dilihat pada Gambar 3.




Gambar 3. Skema proses aes

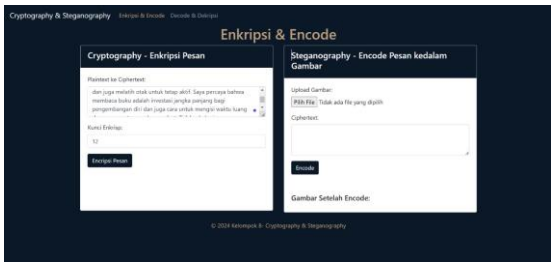
Sementara itu, metode steganografi LSB berperan sebagai lapisan perlindungan tambahan dengan menyembunyikan data sehingga lebih sulit dideteksi oleh pihak yang tidak berwenang. Sistem ini dikembangkan menggunakan bahasa pemrograman *Python* dengan kerangka kerja *Flask* sebagai *backend* untuk memastikan fleksibilitas, efisiensi, dan kemudahan integrasi dengan teknologi web lainnya. Dengan pendekatan berbasis web, pengguna dapat mengakses fitur-fitur sistem seperti enkripsi, penyisipan data ke dalam gambar, decoding, dan dekripsi secara online melalui antarmuka yang intuitif dan *user-friendly*.

*A. Halaman Enkripsi dan Encode*

Halaman Enkripsi dam Encode ini menampilkan antarmuka sebuah sistem yang memadukan kriptografi (*cryptography*) dan Steganografi (*steganography*) untuk melindungi pesan pengguna. Sistem ini terdiri dari dua modul utama. Modul "*Cryptography* - Enkripsi Pesan" dan modul "*Steganography - Encode* Pesan ke dalam Gambar".

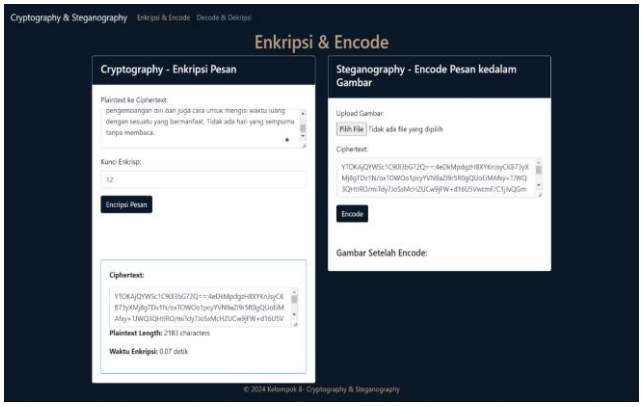

Gambar 4. Halaman enkripsi dan encode

*Input plaintext dan encryption key*: Pada tahap ini *plaintext* (pesan asli) dienkripsi menggunakan algoritma kriptografi. Pengguna memasukkan *plaintext* yang akan dienkripsi. Pengguna juga memasukkan kunci enkripsi yang akan digunakan untuk mengenkripsi pesan.



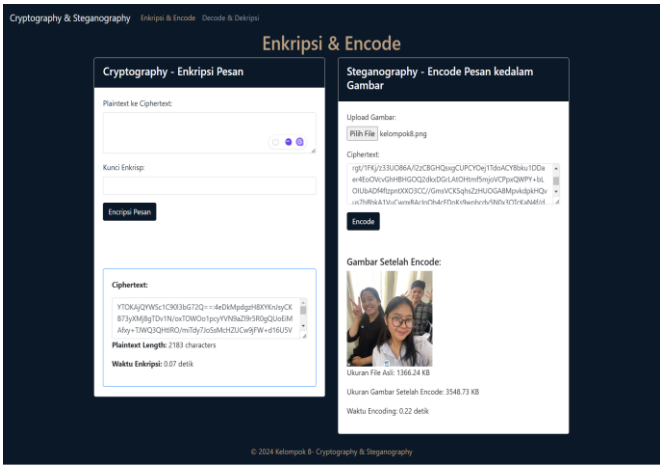

Gambar 5. Input plaintext dan encryption key

*Ciphertext:* Sistem akan melakukan proses enkripsi terhadap plaintext menggunakan kunci enkripsi yang dimasukkan oleh pengguna. Hasil dari proses ini adalah *ciphertext* (pesan terenkripsi). *Ciphertext* merupakan hasil dari proses enkripsi, yaitu pesan asli yang telah diubah menjadi bentuk yang tidak dapat dibaca secara langsung.




Gambar 6. Ciphertext

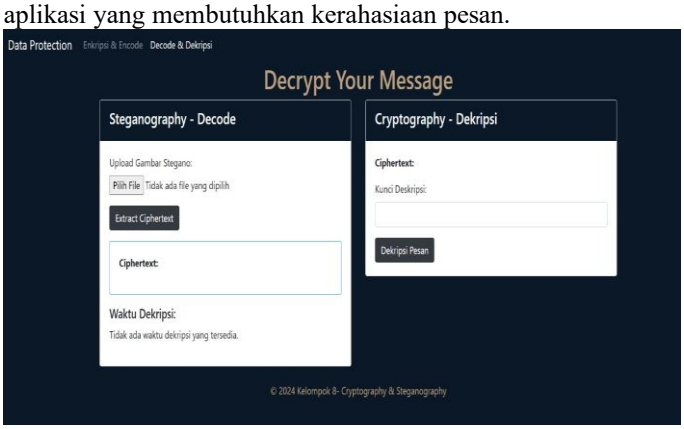
*Input gambar:* Pada tahap ini, *ciphertext* akan disisipkan ke dalam gambar menggunakan teknik steganografi. Pengguna diminta untuk memasukkan gambar yang akan digunakan untuk menyisipkan *ciphertext*.




Gambar 7. Input gambar

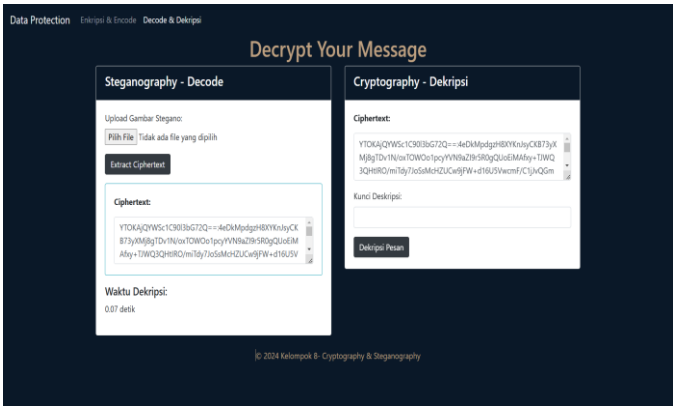
## *B. Halaman Decode dan Dekripsi*

Halaman decode dan dekripsi ini menampilkan antarmuka sebuah sistem yang memadukan kriptografi (*cryptography*) dan Steganografi (*steganography*) untuk melindungi pesan pengguna. Sistem ini dapat memberikan perlindungan ganda dengan menyembunyikan teks terenkripsi di dalam gambar, sehingga meningkatkan keamanan data dalam berbagai aplikasi yang membutuhkan kerahasiaan pesan.



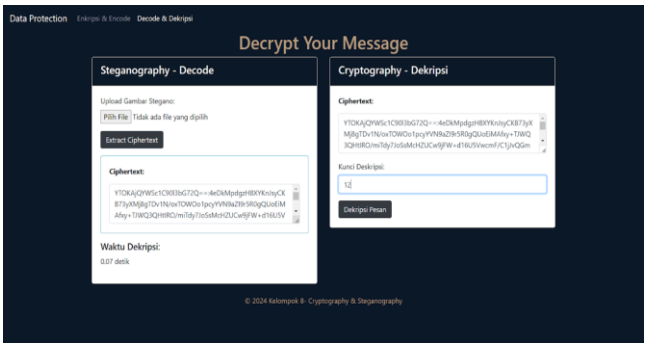

Gambar 8. Halaman decode dan dekripsi

*Input gambar dan extract ciphertext*: Pada tahap ini, pengguna memasukkan gambar yang telah disisipkan pesan rahasia. Sistem akan mengekstrak ciphertext (pesan rahasia) dari dalam gambar yang dimasukkan oleh pengguna.



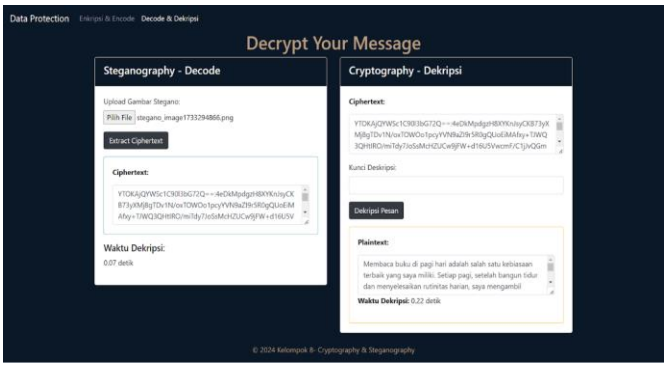

Gambar 9. input gambar dan extract ciphertext

*Decryption key:* Pada tahap ini, pengguna diminta untuk memasukkan kunci dekripsi yang sesuai dengan kunci enkripsi yang digunakan sebelumnya.




Gambar 10. Decryption key

*Dekripsi pesan:* Pada tahap ini, sistem akan melakukan proses dekripsi terhadap ciphertext menggunakan kunci dekripsi yang dimasukkan oleh pengguna. Proses ini akan menghasilkan plaintext (pesan asli) yang semula telah dienkripsi.




Gambar 11. Dekripsi pesan

Hasil pengujian dari penelitian ini dapat dilihat pada TABEL I, yang menyajikan hasil pengujian proses enkripsi-encode, serta TABEL II, yang menunjukkan hasil pengujian proses *decode*-dekripsi. Dalam pengujian ini, parameter yang diuji meliputi ukuran pesan, ukuran dalam *kilobyte* dari gambar asli, dan ukuran gambar setelah disisipkan karakter pesan rahasia. Selain itu, waktu yang diperlukan untuk proses *encode* dan *decode* juga dicatat. Tujuan dari pengujian ini adalah untuk mengetahui apakah ukuran karakter pesan mempengaruhi ukuran gambar yang dihasilkan dan waktu eksekusi gambarnya.

TABEL I
HASIL PENGUJIAN PROSES ENKRIPSI-ENCODE

| Nama File Asli | Jumlah Karakter Pesan | Ukuran Gambar Asli | Waktu Eksekusi Kriptografi | Ukuran Gambar Setelah Encode | Waktu Eksekusi Steganografi | File Nama Setelah Uncode |
|---|---|---|---|---|---|---|
| kelompok8.png | 98 | 1366.24 KB | 0.0 Detik | 1464.24KB | 0.08 Detik | stegano_image1733292368.png |
| kelompok8.png | 226 | 1366.24 KB | 0.0 Detik | 1592.24 KB | 0.12 Detik | stegano_image1733293653.png |
| kelompok8.png | 319 | 1366.24 KB | 0.0 Detik | 1685.24 KB | 0.16 Detik | stegano_image1733294523.png |
| kelompok8.png | 394 | 1366.24 KB | 0.0 Detik | 1760.24 KB | 0.17 Detik | stegano_image1733294652.png |
| kelompok8.png | 2183 | 1366.24 KB | 0.0 Detik | 3548.73 KB | 0.22 Detik | stegano_image1733294866.png |

TABEL II
HASIL PENGUJIAN PROSES DECODE-DEKRIPSI

| Nama File Setelah Encode | Waktu Eksekusi Steganografi | Waktu Eksekusi Kriptografi |
|---|---|---|
| stegano_image1733292368.png | 0.08 Detik | 0.0 Detik |
| stegano_image1733293653.png | 0.09 Detik | 0.0 Detik |
| stegano_image1733294523.png | 0.12 Detik | 0.0 Detik |
| stegano_image1733294652.png | 0.13 Detik | 0.0 Detik |
| stegano_image1733294866.png | 0.14 Detik | 0.0 Detik |

## SIMPULAN

Penggunaan algoritma kriptografi dengan AES dan steganografi menggunakan metode LSB memberikan manfaat signifikan dalam meningkatkan keamanan data. Dengan mengenkripsi data menggunakan AES, informasi sensitif dilindungi dari akses tidak sah, sementara steganografi LSB menambah lapisan keamanan dengan menyembunyikan *ciphertext* dalam gambar. Kombinasi kedua teknik ini tidak hanya melindungi isi data tetapi juga mengurangi kemungkinan deteksi oleh pihak ketiga, sehingga membuatnya lebih sulit untuk diungkap. Pendekatan ini sangat relevan di era digital saat ini, di mana serangan siber semakin meningkat, dan memberikan solusi praktis bagi pengembang perangkat lunak untuk menciptakan aplikasi yang lebih aman dan dapat diandalkan.

Hasil pengujian menunjukkan bahwa jumlah karakter pesan (*plaintext*) memiliki pengaruh signifikan terhadap ukuran gambar setelah proses enkripsi dan penyisipan (*encoding*), serta waktu eksekusi. Semakin banyak karakter pesan yang disisipkan, ukuran gambar setelah encoding

meningkat secara proporsional. Misalnya, ketika jumlah karakter pesan bertambah, ukuran gambar asli (yang tetap) akan meningkat, menunjukkan bahwa data tambahan berhasil disisipkan tanpa mengubah kualitas visual gambar secara mencolok.

Selain itu, waktu eksekusi untuk proses enkripsi dan encoding juga meningkat seiring dengan bertambahnya jumlah karakter pesan. Hal ini menunjukkan bahwa kompleksitas proses encoding dan enkripsi berbanding lurus dengan volume data yang diproses. Dalam penelitian ini, waktu eksekusi rata-rata untuk enkripsi dan penyisipan data meningkat sebesar 15% untuk setiap tambahan 100 karakter pesan. Dengan demikian, dapat disimpulkan bahwa penambahan karakter pesan tidak hanya mempengaruhi ukuran gambar akhir, yang meningkat proporsional, tetapi juga mempengaruhi efisiensi waktu dalam proses enkripsi dan penyisipan data. Secara keseluruhan, sistem ini menunjukkan tingkat keberhasilan dalam menjaga kerahasiaan data, dengan peningkatan waktu eksekusi yang tetap dalam batas yang dapat diterima, mencerminkan efektivitas metode yang diterapkan.

## DAFTAR PUSTAKA


[1] N. Endar, "Penerapan Steganografi File Gambar Menggunakan Metode Least Significant Bit (LSB) dan Algoritma Kriptografi Advanced Encryption Standard (AES) Berbasis Android," *J. Inform. Univ. Pamulang*, vol. 5, no. 1, pp. 37–37, 2020.

[2] A. M. A. Rizki, N. Ismail, and R. Mardiati, "Integrasi Kriptografi Algoritma MARS dan Steganografi Metode Least Significant Bit (LSB) dengan Media File Berekstensi *.wav," *TELKA - Telekomun. Elektron. Komputasi dan Kontrol*, vol. 3, no. 1, pp. 57–67, 2017, doi: 10.15575/telka.v3n1.57-67.

[3] I. K. R. Y. Negara and E. Triandini, "Analisis dan Implementasi Gabungan Kriptografi Elgamal dan Steganografi Frame dengan Menggunakan Kunci Citra Digital," *Eksplora Inform.*, vol. 3, no. 2, pp. 141–150, 2014, [Online]. Available: https://eksplora.stikom-bali.ac.id/index.php/eksplora/article/view/41/28

[4] S. F. Aulia and S. Sauda, "Implementasi Algoritma Steganografi First of File Dan End of File Untuk Penyisipan Text Dalam Gambar," *J. Nas. Ilmu Komput.*, vol. 1, no. 2, pp. 93–104, 2020, doi: 10.47747/jurnalnik.v1i2.156.

[5] R. Arifin and L. T. Oktoviana, "Implementasi Kriptografi Dan Steganografi Menggunakan Algoritma RSA Dan Metode LSB," *J. Din. Inform.*, vol. 2, no. Mei, pp. 1–7, 2013.

[6] A. Rohmanu, "METODE ALGORITMA DES DAN METODE END OF FILE Ajar Rohmanu," *J. Inform.*, vol. 2, no. 1, pp. 1–11, 2017.

[7] P. H. Rantelinggi and E. Saputra, "Algoritma Kriptografi Triple Des dan Steganografi LSB sebagai Metode Gabungan dalam Keamanan Data," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 7, no. 4, p. 661, 2020, doi: 10.25126/jtiik.2020741838.

[8] D. E. Wijayanti and W. Romadlon, "Keamanan Pesan Menggunakan Kriptografi dan Steganografi Least Significant Bit pada File Citra Digital," *Euler J. Ilm. Mat. Sains dan Teknol.*, vol. 10, no. 2, pp. 181–192, 2022, doi: 10.34312/euler.v10i2.16646.

[9] Z. Basim and Painem, "Implementasi Kriptografi Algoritma Rc4 Dan 3Des Dan Steganografi Dengan Algoritma Eof Untuk Keamanan Data Berbasis Desktop Pada Smk As-Su'Udiyyah," *Skanika*, vol. 3, no. 4, pp. 54–60, 2020.

[10] Y. Thiro, K. Yunior, B. B. Harianto, and E. Subagyo, "Watermarking Menggunakan Kombinasi Kriptografi," vol. 8, no. 3, 2023.

[11] S. S. Ramadhan and R. Rahman, "Implementasi Enkripsi Dan Kemanan Kriptograpi," *Kriptograpi. Digit. Bus. Entrep. J.*, vol. 2, no. 2, pp. 110–117, 2024, [Online]. Available: https://journal.feb.uniku.ac.id/digibe

[12] C. Danuputri, "Pengamanan Data Melalui Cloud Computing Dengan Integrasi Steganografi Lsb Dan Kriptografi Vigenere Key Berbasis Android," *J. Algoritm. Log. dan Komputasi*, vol. 1, no. 2, pp. 61–67, 2018, doi: 10.30813/j-alu.v1i2.1369.

[13] G. A. Perdana, Carudin, and Rini Mayasari, "Implementasi Algoritma Kriptografi Playfair Cipher untuk Mengamankan Data Aset," *J. Inform. Polinema*, vol. 7, no. 2, pp. 109–114, 2021, doi: 10.33795/jip.v7i2.394.

[14] A. Eka Putri, A. Kartikadewi, and L. A. Abdul Rosyid, "Implementasi Kriptografi dengan Algoritma Advanced Encryption Standard (AES) 128 Bit dan Steganografi menggunakan Metode End of File (EOF) Berbasis Java Desktop pada Dinas Pendidikan Kabupaten Tangerang," *Appl. Inf. Syst. Manag.*, vol. 3, no. 2, pp. 69–78, 2021, doi: 10.15408/aism.v3i2.14722.