JITET (Jurnal Informatika dan Teknik Elektro Terapan)
Vol. 14 No. 1, pISSN: 2303-0577 eISSN: 2830-7062 http://dx.doi.org/10.23960/jitet.v14i1.8521

IMPLEMENTASI KOMBINASI *FERNET ENCRYPTION* DAN LSB *STEGANOGRAPHY* PADA SISTEM KEAMANAN INFORMASI BERBASIS CITRA

Jadiaman Parhusip[1], Muhammad Arifin Ilham[2*], Rinaldi Rizwar[3], Arya Bima Mohammad Heriansyah[4], Raydamar Rizkyaka Riyadi[5]

[1,2,3,4,5] Teknik Informatika, Universitas Palangka Raya; Jl. Yos Sudarso, Palangka, Kec. Jekan Raya, Kota Palangka Raya, Kalimantan Tengah 74874; 0536-3227111

Keywords:
cryptography;
steganography;
Fernet encryption;
LSB;
information security.

Corespondent Email:
muhammadarifinilham@mhs.eng.ac.id

Abstrak: Keamanan informasi menjadi aspek penting dalam era pertukaran data digital yang rentan terhadap penyalahgunaan. Kombinasi kriptografi dan steganografi dipilih karena mampu memberikan perlindungan ganda, yaitu menjaga kerahasiaan isi dan keberadaan pesan. Penelitian ini mengimplementasikan algoritma Fernet Encryption sebagai metode enkripsi dan teknik Least Significant Bit (LSB) untuk penyisipan pesan ke dalam citra digital. Proses implementasi dilakukan menggunakan Python dengan pustaka cryptography, Pillow, dan NumPy. Pada penelitian ini, media penampung berupa citra acak (random noise) yang dihasilkan secara khusus untuk memastikan kapasitas embedding mencukupi, sehingga perubahan visual tidak menjadi faktor utama. Hasil pengujian menunjukkan bahwa token terenkripsi dapat disisipkan dan diekstraksi kembali secara utuh, serta berhasil didekripsi menggunakan kunci yang benar. Temuan ini membuktikan bahwa kombinasi Fernet dan LSB dapat digunakan sebagai bukti konsep mekanisme keamanan berlapis yang efektif untuk penyimpanan pesan rahasia pada media citra digital, dan berpotensi dikembangkan lebih lanjut pada penerapan dengan citra alami untuk penggunaan praktis.

Abstract: *Information security is an essential aspect in the digital era, where data exchange is highly vulnerable to misuse. The combination of cryptography and steganography provides dual protection by securing message confidentiality and concealing its presence. This study implements the Fernet Encryption algorithm for data encryption and the Least Significant Bit (LSB) technique for embedding encrypted messages into digital images. The system is developed using Python with the cryptography, Pillow, and NumPy libraries. In this study, the cover medium is a randomly generated noise image, chosen to ensure sufficient embedding capacity, making visual distortion irrelevant to the analysis. Experimental results show that the encrypted token can be successfully embedded, extracted, and decrypted using the correct key. These findings demonstrate that the combination of Fernet and LSB serves as an effective proof-of-concept for layered security mechanisms in digital image media, with potential for further development using natural images for practical steganography applications.*

1. PENDAHULUAN

Perkembangan teknologi digital membuat pertukaran informasi berlangsung semakin cepat melalui berbagai media elektronik. Situasi ini turut meningkatkan risiko kebocoran data, pencurian informasi, dan manipulasi pesan oleh

pihak tidak berwenang. Keamanan informasi menjadi aspek penting karena tanpa mekanisme perlindungan yang memadai, data yang dikirimkan dapat dengan mudah disalahgunakan dan merugikan pemilik data [1]. Hal ini menegaskan perlunya sistem yang mampu menjaga kerahasiaan, integritas, dan keaslian data agar hanya dapat diakses oleh pihak yang berhak.

Salah satu pendekatan yang banyak digunakan untuk melindungi informasi adalah steganography, yaitu teknik menyembunyikan pesan rahasia ke dalam media digital seperti citra, sehingga perubahan yang terjadi tidak tampak secara visual. Di antara berbagai teknik steganography, metode Least Significant Bit (LSB) dikenal luas karena mampu menyisipkan pesan dengan hanya memodifikasi bit paling rendah pada piksel. Meskipun efisien, metode ini tetap memiliki kelemahan, terutama terhadap serangan analisis pola apabila proses penyisipannya dapat diidentifikasi [2].

Sejumlah penelitian sebelumnya telah mencoba meningkatkan keamanan steganography LSB dengan menambahkan lapisan kriptografi. Fadel et al. menunjukkan bahwa LSB yang digunakan secara tunggal rentan terhadap deteksi oleh algoritma steganalisis yang lebih canggih, sehingga membutuhkan penguatan tambahan [3]. Penelitian yang dilakukan oleh Sari, C. A. & Sari, W. S., 2022 mengintegrasikan teknik enkripsi sebelum penyisipan pada citra digital, dan hasilnya kualitas visual citra stegano tetap tinggi berdasarkan nilai PSNR. Penelitian lain oleh Sidiq, R. F., Gunadhi Rahayu, R. & Supriatna, A. D., 2023 menambahkan proses enkripsi guna meningkatkan kerahasiaan melalui keamanan berlapis [4]. Namun, penelitian-penelitian tersebut belum membahas secara komparatif efisiensi algoritma kriptografi maupun kompleksitas pengelolaan kunci, sehingga belum memberikan gambaran menyeluruh mengenai alternatif kriptografi yang lebih ringan dan praktis untuk mendukung steganography berbasis citra.

Pendekatan lain dilakukan oleh Set et al., yang menerapkan enkripsi AES-256 sebelum proses penyisipan menggunakan LSB, dan menghasilkan perlindungan berlapis dengan nilai PSNR di atas 58 dB [5]. Penelitian Alanzy et al. juga mengusulkan metode hibrida menggunakan AES dan Blowfish, yang meningkatkan tingkat keamanan tanpa menurunkan kualitas visual citra [6]. Sementara itu, Ahmed et al. mendemonstrasikan bahwa kombinasi Fernet dan LSB mampu menghasilkan sistem steganography yang aman, mudah diimplementasikan, dan tahan terhadap gangguan [7]. Pendekatan ini menggunakan dua lapis enkripsi sebelum penyimpanan data sehingga memberikan tingkat keamanan lebih baik dibandingkan metode tunggal [8].

Fernet sendiri merupakan algoritma yang menggabungkan enkripsi simetris dan autentikasi, sehingga ciphertext yang dihasilkan tidak hanya bersifat rahasia tetapi juga dapat diverifikasi integritasnya sebelum didekripsi. Keunggulan ini membuatnya relevan sebagai bagian dari sistem keamanan yang menuntut kerahasiaan sekaligus validasi data.

Berdasarkan kondisi tersebut, penggunaan kombinasi Fernet encryption dan LSB steganography menjadi relevan karena menyediakan mekanisme keamanan berlapis, mempertahankan kualitas citra penampung, serta memiliki kebutuhan komputasi yang relatif ringan. Penelitian ini menawarkan kebaruan berupa penerapan skema gabungan tersebut dengan fokus pada efektivitas serta performa sistem dalam proses penyisipan, ekstraksi, dan dekripsi pesan [1][2]. Tujuannya adalah memastikan bahwa ciphertext dapat disisipkan, diambil kembali, dan didekripsi secara utuh menggunakan kunci yang tepat [4], sehingga memberikan kontribusi yang lebih komprehensif terhadap pengembangan metode keamanan informasi berbasis citra digital yang aman, efisien, dan terukur.

2. TINJAUAN PUSTAKA

2.1 Enkripsi dan Deskripsi

Enkripsi merupakan proses melindungi data dengan menyamarkannya atau mengubahnya menjadi format yang tidak dapat dibaca maupun dipahami. Sementara itu, dekripsi adalah proses mengembalikan data yang sudah terenkripsi ke bentuk aslinya agar dapat dibaca dan dimengerti lagi[10].

2.2 Kriptografi

Kriptografi adalah ilmu yang mempelajari cara menjaga keamanan data atau pesan selama pengiriman, agar tetap terlindungi dari gangguan pihak ketiga. Teknik kriptografi

dimanfaatkan untuk mengubah pesan rahasia ke bentuk tersandi dengan memanfaatkan metode operasi cipher block chaining[12]. Pada penelitian ini digunakan kriptografi simetris, dimana proses enkripsi dan dekripsi menggunakan kunci yang sama.

2.3 Advanced Encryption Standard

Advanced Encryption Standard (AES) merupakan algoritma enkripsi yang banyak digunakan pada berbagai sistem keamanan modern berkat tingkat proteksi yang kuat dan kinerja yang efisien. Pada penerapannya, AES dapat digunakan untuk mengamankan berbagai jenis data, termasuk citra digital berformat PNG, yang sering dijadikan media penyimpanan dan pengolahan informasi [11].

2.4 Fernet Encryption

Metode enkripsi Fernet, atau algoritma Fernet, bekerja dengan konsep yang serupa dengan AES. Fernet mendukung pergantian atau rotasi kunci yang dihasilkan oleh MultiFernet saat proses penyandian atau enkripsi dilakukan[1]. Fernet mengimplementasikan AES (Advanced Encryption Standard) dengan panjang kunci 128 bit dalam mode *Cipher Block Chaining* (CBC), di mana setiap blok 29 plaintext dienkripsi menggunakan kunci yang sama dan hasil enkripsi dari blok sebelumnya, untuk meningkatkan ketahanan terhadap serangan analisis [13].

Kelebihan Fernet:

a) Aman karena menggabungkan enkripsi dan autentikasi pesan.
b) Format ciphertext sudah terstandarisasi Base64.
c) Mudah diimplementasikan dengan satu kunci.
d) Struktur ciphertext Fernet:

version || timestamp || IV || ciphertext || HMAC

2.5 Steganography

Steganography merupakan teknik menyembunyikan pesan di dalam suatu media penyisipan pesan atau cover image didalam sebuah objek penutup seperti gambar, video, atau audio sehingga keberadaan pesan rahasia yang disisipkan tidak dapat dilihat secara langsung [9]. Tujuan utama steganography adalah menyamarkan keberadaan pesan, bukan sekadar mengacak isinya[14].

Menurut **Subramanian dkk. (2021)**, steganography citra merupakan teknik untuk menyembunyikan data rahasia baik berupa teks, gambar, atau video ke dalam sebuah citra penampung (cover image), sehingga keberadaan data tersembunyi tidak dapat dideteksi secara kasat mata[15].

2.6 Metode Least Significant Bit (LSB)

Selain Steganography, metode yang paling sering dipakai untuk menyembunyikan informasi adalah teknik Least Significant Bit (LSB). Metode LSB ini menggunakan citra digital sebagai covertext. LSB merupakan teknik steganography yang menyisipkan bit pesan ke dalam bit paling tidak signifikan pada nilai piksel citra. Contoh penyisipan LSB pada satu komponen warna: [1][2]

Nilai piksel	: 10110110
Bit pesan	: 1
Hasil	: 10110111

Perubahan hanya pada bit terakhir, sehingga perubahan warna hampir tidak terlihat oleh mata manusia

3. METODE PENELITIAN

3.1 Desain Penelitian (Research Design)

Penelitian ini menggunakan pendekatan eksperimental terapan (*applied experimental research*) dengan fokus pada pengembangan sistem keamanan informasi berbasis citra menggunakan kombinasi dua metode, yaitu kriptografi Fernet dan steganography *Least Significant Bit* (LSB).

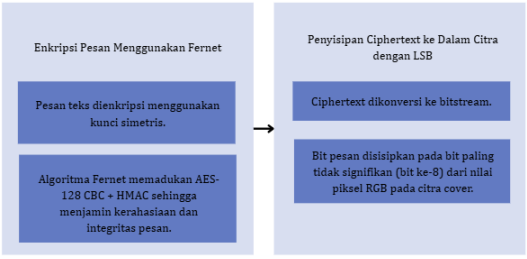
Eksperimen dilakukan untuk membuktikan tingkat keamanan dan efektivitas kombinasi kedua metode tersebut dalam menjaga kerahasiaan pesan teks yang disisipkan pada gambar digital.

3.2 Arsitektur Sistem (System Architecture)

Arsitektur sistem yang dikembangkan terbagi menjadi dua komponen utama, yaitu:

a) Modul Enkripsi (*Encryption Module*): berfungsi untuk mengenkripsi pesan teks menggunakan algoritma Fernet, yang memanfaatkan AES-128 dalam mode CBC dan HMAC-SHA256 untuk menjamin integritas data[16].
b) Modul Steganography (Steganography Module): berfungsi untuk melindungi kerahasiaan pesan[12], dengan

menyisipkan ciphertext hasil enkripsi ke dalam citra (cover image) dengan teknik *Least Significant Bit* (LSB) pada komponen warna RGB.



3.3 Implementasi Sistem

Implementasi dilakukan menggunakan bahasa pemrograman Python 3.11 dengan pustaka utama berikut:

a) Cryptography.fernet untuk proses enkripsi dan dekripsi.
b) Pillow (PIL) untuk pengolahan citra digital (load, modify, dan save image pixel).
c) NumPy untuk manipulasi data piksel dalam format array.

Langkah-langkah implementasi:

3.3.1. Proses Enkripsi dan Penyisipan (Embedding)

Algotitma Embedding:

Input : Citra asli (cover image), Pesan teks rahasia M
Output : Citra stegano yang berisi pesan terenkripsi
1. Generate kunci Fernet (FKey)
2. Encrypt(M, FKey) → C
3. Convert C menjadi representasi bitstream B
4. Buka citra dan konversi ke array piksel
5. Untuk setiap bit pada B:
   - Ambil piksel berikutnya
   - Ganti bit LSB dari tiap komponen warna dengan bit dari B.
6. Simpan citra hasil penyisipan sebagai stegano image.

3.3.2. Proses Ekstraksi dan Deskripsi (Decoding)

Input : Citra stegano, Kunci Fernet (FKey)
Output : Pesan asli M
1. Buka citra stegano dan ekstrak bit LSB dari piksel
2. Gabungkan bit menjadi ciphertext C
3. Decrypt(C, FKey) → M
4. Tampilkan pesan M

3.4 Metode Pengujian dan Pengambilan Data

Pengujian dalam penelitian ini dilakukan untuk mengevaluasi kualitas citra stegano dan memastikan pesan yang disisipkan dapat diekstraksi serta didekripsi kembali secara utuh. Pengujian dilakukan melalui dua tahap, yaitu uji kualitas visual dan uji keberhasilan dekripsi pesan.

3.4.1. Pengambilan Data Citra Uji

Data citra yang digunakan diambil dari dataset citra digital bebas hak cipta berformat PNG dan BMP dengan ukuran 512×512 piksel. Pemilihan format ini didasarkan pada sifatnya yang lossless, sehingga tidak terjadi penurunan kualitas gambar akibat kompresi. Jumlah citra uji dapat terdiri dari 5–10 gambar dengan variasi tingkat kecerahan dan komposisi warna.

3.4.2. Pengujian Proses Enkripsi dan Steganography

a) Pesan teks dengan panjang variatif (misalnya 50, 100, 150 karakter) dienkripsi menggunakan algoritma Fernet.
b) Ciphertext hasil enkripsi kemudian disisipkan ke dalam citra menggunakan metode LSB.
c) Citra stegano yang dihasilkan kemudian diekstraksi kembali untuk memperoleh ciphertext, lalu dilakukan proses dekripsi menggunakan kunci yang sama.

3.4.3. Uji Keberhasilan

Keberhasilan penyembunyian pesan dikatakan valid apabila:

$$M_{asli} = M_{hasil\ deskripsi}$$

Tidak boleh terdapat perubahan karakter pada pesan yang diperoleh kembali.

4. HASIL DAN PEMBAHASAN

Hasil eksperimen pada implementasi kombinasi enkripsi simetris menggunakan pustaka Fernet dan teknik steganography LSB (*Least Significant Bit*) untuk menyembunyikan berkas teks ke dalam citra digital. Eksperimen dilakukan pada lingkungan Google Colab dengan bahasa pemrograman Python, pustaka cryptography untuk Fernet, serta PIL dan

numpy untuk manipulasi citra. Deskripsi alur eksperimen: teks asli dienkripsi terlebih dahulu menggunakan kunci Fernet yang dihasilkan secara acak, kemudian token terenkripsi (dengan header panjang 4 bytes) dikonversi menjadi representasi bit dan di-embed ke LSB kanal RGB pada citra buatan (random noise) yang ukurannya dihitung sesuai kebutuhan kapasitas.

4.1 Hasil Eksperimen

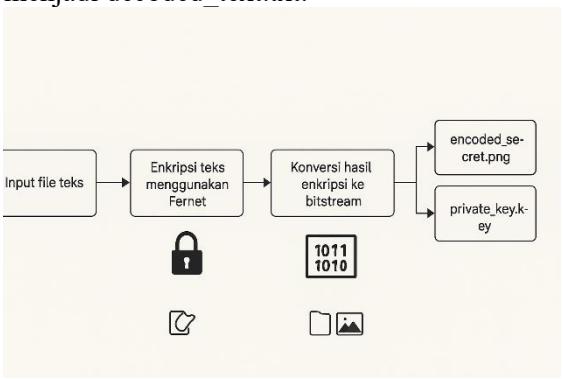
Hasil percobaan yang diperoleh dari skrip eksekusi ditampilkan pada Tabel 1.2 dan dijabarkan dalam paragraf berikut.

Tabel 1.2 Hasil Eksperimen

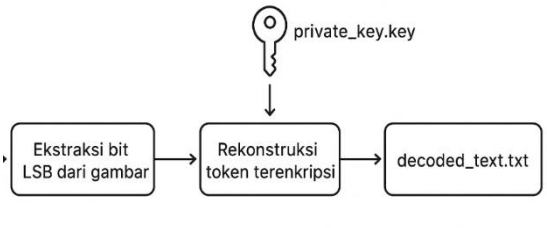
PARAMETER	HASIL
Panjang token terenkripsi (bytes)	3.108
Panjang payload (header 4 bytes + token) (bytes)	3.112
Panjang payload (bits)	24.896
Pixel minimum yang dibutuhkan (ceil(bits/3))	8.299
Dimensi citra yang digenerasi (lebar × tinggi)	92 × 92 piksel
Total kapasitas embedding (bits)	25.392
Kapasitas embedding (bytes)	3.174
Berkas yang dihasilkan pada proses encode	encoded_secret.png, private_key.key
Hasil dekripsi pada proses decode	Berhasil: decoded_text.txt dipulihkan dan diunduh

Perhitungan di atas dilakukan sebagai berikut. Token terenkripsi memiliki panjang 3.108 bytes; payload yang disisipkan terdiri dari header 4 bytes untuk menyimpan panjang token, sehingga total payload menjadi 3.112 bytes. Total bit payload adalah 3.112 × 8 = 24.896 bit. Karena setiap piksel RGB menyediakan 3 bit (satu bit per kanal R, G, B) untuk teknik LSB yang digunakan, jumlah piksel minimum yang diperlukan adalah ceil(24.896 / 3) = 8.299 piksel. Untuk mempermudah pemrosesan dan menjaga bentuk citra persegi, dimensi citra dipilih sebagai lebar = tinggi = ceil(sqrt(8.299)) = 92, sehingga citra berukuran 92 × 92 piksel (total 8.464 piksel) yang menyediakan kapasitas 8.464 × 3 = 25.392 bit (setara 3.174 bytes) untuk embedding. Kapasitas ini lebih besar daripada ukuran payload sehingga embedding dapat dilakukan tanpa overflow.

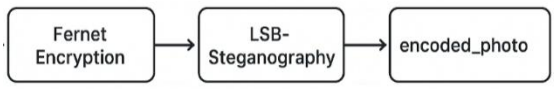
Proses enkripsi menghasilkan token terenkripsi sebesar 3.108 bytes yang disimpan sebagai token biner sebelum disisipkan. Proses encoding menulis bit payload ke LSB setiap kanal warna pada tiap piksel berurutan hingga seluruh payload selesai disisipkan. Citra keluaran disimpan sebagai encoded_secret.png dan kunci Fernet diekspor ke berkas private_key.key. Pada proses dekripsi, skrip membaca ulang bit LSB dari citra, merekonstruksi header 4 bytes pertama untuk menentukan panjang token, mengekstrak token sesuai panjang tersebut, lalu mendekripsi token menggunakan kunci Fernet yang diunggah pengguna. Pada percobaan ini proses dekripsi berhasil dan teks asli berhasil direkonstruksi menjadi decoded_text.txt.



Gambar 1.2 Diagram alur proses enkripsi dan penyisipan pesan



Gambar 1.3 Diagram alur proses dekripsi dan rekonstruksi token



Gambar 1.4 Perbandingan citra sebelum dan sesudah penyisipan pesan rahasia

4.2 Pembahasan

Lapisan pertama keamanan dalam sistem ini adalah kriptografi. Implementasi ini tidak menggunakan algoritma enkripsi tunggal seperti AES murni, melainkan memilih Fernet, sebuah resep kriptografi simetris terotentikasi tingkat tinggi. Pilihan ini memiliki implikasi keamanan yang signifikan:

1) Keamanan Terpadu: Fernet secara otomatis menyediakan enkripsi terotentikasi. Ini berarti Fernet tidak hanya memberikan kerahasiaan (menggunakan AES-128 dalam mode CBC) tetapi juga menjamin integritas dan otentikasi (menggunakan HMAC-SHA256).
2) Struktur Token: Hasil dari log (Panjang token terenkripsi: 3108 bytes) merujuk pada "token" Fernet. Token ini bukan ciphertext murni, melainkan sebuah paket data terstruktur yang berisi: Versi (1 byte), Stempel Waktu (8 byte), IV (16 byte), Ciphertext (data terenkripsi), dan Tanda Tangan HMAC (32 byte). Ini berarti terdapat overhead tetap sebesar 57 byte per token untuk fitur keamanan tambahan ini.
3) Mekanisme Pertahanan: Fitur paling kritis dari Fernet adalah validasi HMAC. Dalam Program 2, blok try...except InvalidToken: berfungsi sebagai fitur keamanan inti. Error InvalidToken akan muncul tidak hanya jika kunci salah, tetapi juga jika payload token yang diekstraksi dari gambar telah rusak atau diubah (misalnya, akibat kompresi lossy atau modifikasi LSB yang disengaja). Ini memberikan lapisan integritas data yang kuat pada lapisan steganography yang mendasarinya.

Lapisan kedua adalah steganography, yang diimplementasikan menggunakan metode Least Significant Bit (LSB) 1-bit per saluran warna. Analisis kode Program menunjukkan implementasi mekanisme LSB yang secara teknis fundamental.

4.2.1. Proses Penyisipan (Encoding)

Logika penyisipan data (img_array[i, j, k] & 254) | bit adalah operasi bitwise dua langkah yang presisi:

1) **Pembersihan LSB (... & 254)**: Angka 254 dalam biner 8-bit adalah 11111110. Melakukan operasi bitwise AND (&) antara nilai piksel asli dengan 11111110 akan secara paksa mengatur LSB (bit terakhir) ke 0 sambil mempertahankan 7 bit lainnya.
2) **Pengaturan LSB (... | bit)**: Variabel bit (yang bernilai 0 atau 1) kemudian di-OR-kan. Karena LSB dari langkah sebelumnya dijamin 0, operasi bitwise OR (|) ini secara efektif mengatur LSB ke nilai bit data rahasia.

4.2.2. Proses Ekstraksi (Decoding)

Logika ekstraksi arr[i, j, k] & 1 adalah kebalikan yang efisien:

1) **Isolasi LSB (... & 1)**: Angka 1 dalam biner 8-bit adalah 00000001. Melakukan *bitwise AND* (&) antara nilai piksel *stego* dengan 1 akan mengabaikan 7 bit pertama dan *hanya* mengembalikan nilai LSB (0 atau 1), yang merupakan bit data rahasia yang disembunyikan.

4.3 Pembahasan "Jembatan" Sistem: Manajemen Payload

Komponen paling kritis dari implementasi gabungan ini adalah "jembatan" logis yang menghubungkan lapisan kriptografi (token Fernet) dan lapisan steganography (aliran bit LSB).

Tantangan utamanya adalah: decoder harus tahu kapan harus berhenti membaca bit. Jika decoder membaca terlalu sedikit bit, token Fernet akan terpotong. Jika membaca terlalu banyak, token akan tercampur dengan data LSB "sampah". Kedua skenario ini akan menyebabkan kegagalan validasi HMAC dan memicu error InvalidToken.[10]

Implementasi ini memecahkan masalah tersebut menggunakan solusi header dengan panjang tetap, yang merupakan praktik standar dalam protokol data.[2]

4.3.1 Saat Encoding (Program 1):

a) Sistem menghitung panjang token (misal, 3108 bytes).

b) Sistem membuat header 4-byte (32-bit) yang merepresentasikan angka 3108 (header = token_len.to_bytes(4,...)).
c) Sistem menggabungkan keduanya: payload = header + token.
d) Seluruh 3112 byte payload inilah yang diubah menjadi 24.896 bits dan disisipkan ke dalam citra.

4.3.2 Saat Decoding (Program 2):

a) Sistem mengekstraksi 32 bit pertama (bitstring[:32]).
b) Sistem mengonversi 32 bit ini kembali menjadi integer (token_length = int.from_bytes(...)), yang menghasilkan nilai 3108.
c) Sistem kemudian tahu persis berapa banyak bit lagi yang harus dibaca: 3108×8 bits.
d) Sistem secara presisi "mengiris" sisa aliran bit (bitstring[32:32 + token_length * 8]), memastikan bahwa token_bytes yang direkonstruksi bersih, berukuran tepat, dan siap untuk didekripsi oleh Fernet.

Pilihan *header* 4-*byte* (32-*bit*) juga menunjukkan skalabilitas desain. Ini memungkinkan sistem untuk menangani *payload* rahasia dengan ukuran maksimum $2^{32}-1$ *bytes* (sekitar 4.29 GB), yang hanya dibatasi oleh kapasitas citra *cover*.

4.4 Analisis Kritis Media Sampul

Satu temuan penting dari analisis kode Program 1 adalah pilihan media sampul. Implementasi ini tidak menggunakan gambar yang ada, melainkan membuat media sampul baru menggunakan img_array = np.random.randint(0, 256,...).

Meskipun fungsional untuk membuktikan mekanisme LSB, pendekatan ini secara fundamental melemahkan tujuan steganography. Tujuan steganography adalah covertness (terselubung) atau bersembunyi di depan mata (hiding in plain sight).16 Sebuah file gambar yang seluruhnya terdiri dari noise acak murni memiliki entropi maksimum dan merupakan anomali statistik yang sangat mencolok. File semacam itu akan segera menarik kecurigaan dalam analisis forensik, yang bertentangan dengan tujuan untuk tidak terdeteksi.

Oleh karena itu, hasil penelitian ini menunjukkan bahwa sementara mekanisme kriptografi (Fernet) dan mekanisme manajemen payload (Header 4-byte) diimplementasikan dengan kuat dan canggih, pemilihan media sampul (citra acak) dalam eksperimen ini berfungsi sebagai proof-of-concept teknis, bukan sebagai aplikasi steganography yang aman secara praktis.

4.5 Implikasi dan Saran untuk Penelitian Lanjutan

Pendekatan enkripsi-then-steganography yang digunakan di sini memperlihatkan bukti konsep yang kuat: data terenkripsi secara kriptografis terlebih dahulu sehingga meskipun terdeteksi keberadaannya, isi pesan tetap terlindungi. Untuk meningkatkan kualitas kontribusi ilmiah, disarankan melakukan pengujian terkontrol yang meliputi uji deteksi steganalisis (mis. menggunakan fitur statistik dan model pembelajaran mesin), evaluasi ketahanan terhadap manipulasi (crop, resize, recompression), dan eksperimen embedding pada citra alami dengan berbagai tingkat kompleksitas tekstur. Benchmark kinerja juga dapat ditambahkan untuk melaporkan waktu enkripsi, waktu embedding, serta konsumsi memori pada berbagai ukuran payload.

4.6 Lampiran: Ringkasan Keluaran Program

Program enkode menghasilkan dua berkas utama yaitu encoded_secret.png (citra dengan pesan tersembunyi) dan private_key.key (kunci Fernet). Panjang token terenkripsi yang tercatat pada percobaan ini adalah 3.108 bytes. Proses decode pada percobaan berhasil merekonstruksi teks asli dan menyimpannya sebagai decoded_text.txt.

5. KESIMPULAN

1) Penelitian ini berhasil mengimplementasikan kombinasi algoritma Fernet Encryption dan metode Least Significant Bit (LSB) Steganography untuk menyisipkan pesan terenkripsi ke dalam citra digital. Hasil pengujian menunjukkan bahwa ciphertext dapat disisipkan, diekstraksi, dan didekripsi kembali secara utuh menggunakan kunci yang benar.
2) Kelebihan sistem terletak pada lapisan keamanan ganda yang diperoleh melalui enkripsi dan steganography sekaligus.

Penggunaan Fernet memberikan perlindungan kerahasiaan serta integritas pesan melalui mekanisme enkripsi terotentikasi, sementara LSB memungkinkan penyisipan data secara sederhana dan efisien pada tingkat bit.

3) Kekurangan sistem terlihat pada penggunaan citra acak (random noise) sebagai media penampung, yang meskipun efektif untuk pembuktian konsep (proof-of-concept), namun tidak optimal untuk skenario steganography praktis karena tidak memiliki aspek covertness atau kemampuan menyamar dalam citra alami.
4) Sistem ini menunjukkan bahwa kombinasi Fernet dan LSB dapat menjadi pendekatan yang efektif dan mudah diimplementasikan sebagai mekanisme penyimpanan pesan rahasia berbasis citra digital, terutama dalam konteks eksperimen teknis.
5) Pengembangan lebih lanjut dapat diarahkan pada penggunaan citra alami sebagai media cover, pengujian ketahanan terhadap berbagai serangan steganalisis, analisis kualitas citra seperti PSNR/SSIM, serta evaluasi performa sistem pada berbagai ukuran payload dan variasi format citra.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada semua pihak yang telah memberikan dukungan, bantuan, dan fasilitas selama proses penelitian ini berlangsung, sehingga penelitian ini dapat diselesaikan dengan baik.

DAFTAR PUSTAKA

[1] G. Ragasiwi, T. F. N. S, and V. A. L, "Pemanfaatan Metode Least Significant Bit dan Kriptografi Fernet dalam Steganografi," pp. 461–465, 2024.
[2] C. A. Sari and W. S. Sari, "Kombinasi Least Significant Bit (LSB-1) Dan Rivest Shamir Adleman (RSA) Dalam Kriptografi Citra Warna," vol. 13, no. 1, pp. 45–58, 2022.
[3] R. D. Saputra, R. N. Putra, Y. Fatma, T. Informatika, I. Komputer, and U. M. Riau, "Jurnal Computer Science and Information Technology (CoSciTech)," vol. 5, no. 1, pp. 36–41, 2024.
[4] R. F. Sidiq, R. Erwin, G. Rahayu, and A. D. Supriatna, "Implementasi Kriptografi Advanced Encryption Standard dan Least Significant Bit untuk Keamanan Pesan Email dalam Gambar," pp. 305–315.
[5] F. Maria, C. B. Set, C. M. N. Bana, M. A. Anunut, D. Costa, and Y. Niis, "Penerapan Steganografi LSB dan Enkripsi AES untuk Keamanan Data Rahasia pada Gambar Digital," vol. 3, no. 7, pp. 999–1006, 2025.
[6] M. Alanzy, R. Alomrani, B. Alqarni, and S. Almutairi, "applied sciences Image Steganography Using LSB and Hybrid Encryption Algorithms," 2023.
[7] S. Ahmed, N. Podile, M. H. Kodali, V. Nalajala, Y. Sri, and M. Kukunuri, "Secure and Robust Data Hiding in RGB Images using Steganography," vol. 16, no. 2, pp. 1–13, 2025.
[8] M. Rachna and M. Rachna, "Enhancing Security of Cloud Data through Encryption with AES and Fernet Algorithm through Convolutional-Neural-Networks (CNN)," vol. 8, no. 4, pp. 288–299, 2021, doi: 10.22247/ijcna/2021/209697.
[9] I. Science, C. Science, C. Science, V. S. A. Group, and A. E. Standard, "Securing Medical Images using Encryption and LSB Steganography".
[10] S. N. Nugraha and K. J. Timur, "Penerapan algoritma kriptografi elgamal pada aplikasi pengamanan pesan berbasis website," vol. 12, no. 3, 2024.
[11] M. A. Firdaus, A. Rahmatulloh, F. Teknik, and U. Siliwangi, "IMPLEMENTASI STEGANOGRAFI CITRA DIGITAL LSB MENGGUNAKAN ENKRIPSI AES-256 DAN EMBEDDING," vol. 13, no. 1, 2025.
[12] M. Afsari, D. I. Mulyana, A. Damaiyanti, and N. Sa, "Jurnal Pendidikan Sains dan Komputer Implementasi Mode Operasi Kombinasi Cipher Block Chaining dan Metode LSB-1 Pada Pengamanan Data text Jurnal Pendidikan Sains dan Komputer," vol. 2, no. 1, pp. 70–82, 2022.
[13] D. A. N. Transfer, L. Restnet, and M. N. E. T. V, "Komparasi algoritma," 2024.
[14] I. Pujianto and D. Darwis, "UJI KETAHANAN CITRA DIGITAL TERHADAP MANIPULASI," vol. 2, no. 1, pp. 16–27, 2021.
[15] N. Subramanian and O. Elharrouss, "Image Steganography : A Review of the Recent Advances," vol. 9, pp. 23409–23423, 2021, doi: 10.1109/ACCESS.2021.3053998.
[16] E. Clarita, T. Tunas, M. M. Seran, and V. V. Yaved, "Enkripsi End-to-End pada Aplikasi WhatsApp Menggunakan Metode," vol. 3, no. 7, pp. 927–933, 2025.