IMPLEMENTASI STEGANOGRAFI DENGAN METODE LEAST SIGNIFICANT BIT (LSB) DAN KRIPTOGRAFI ADVANCED ENCRYPTION STANDART (AES)

E.Setyabudi, R.Renaldy dan M.D.Febriansyah[3]
[1,2,3]*Jurusan Informatika, Fakultas Teknik dan Informatika, Universitas PGRI Semarang*
Gedung B Lantai 3, Kampus 1 Jl. Sidodadi Timur 24, Semarang
E-mail : 22670144@upgris.ac.id[1], ramadhanrenaldy@upgris.ac.id[2], 22670145@upgris.ac.id[3]

Abstrak

Keamanan informasi merupakan kebutuhan penting di era digital, terutama dalam melindungi data sensitif dari ancaman akses tidak sah. Penelitian ini menggabungkan teknik steganografi metode Least Significant Bit (LSB) dengan algoritma kriptografi Advanced Encryption Standard (AES) untuk menyediakan perlindungan ganda terhadap data digital. Steganografi LSB menyembunyikan pesan pada piksel gambar dengan memodifikasi bit paling tidak signifikan, sementara AES mengenkripsi pesan untuk meningkatkan keamanan. Penelitian ini melibatkan empat tahapan utama: enkripsi pesan, penyisipan pesan ke dalam gambar (embedding), ekstraksi pesan, dan dekripsi pesan. Implementasi menunjukkan bahwa metode ini dapat menjaga integritas pesan yang disisipkan tanpa perubahan signifikan pada gambar asli. Namun, pesan dapat rusak jika citra diubah formatnya, dipangkas, atau diubah ukurannya. Kombinasi teknik ini memberikan solusi yang andal untuk menjaga kerahasiaan informasi digital dalam berbagai aplikasi.

***Kata Kunci*:** Keamanan informasi, steganografi, Least Significant Bit (LSB), Advanced Encryption Standard (AES), enkripsi, penyisipan pesan, kerahasiaan data.

I. PENDAHULUAN

Pada era industri 4.0, kebutuhan manusia akan informasi semakin pesat setiap harinya. Keamanan informasi menjadi semakin penting di era digital saat ini, terutama dalam menjaga kerahasiaan data sensitif. Dengan meningkatnya penggunaan internet dan pertukaran data secara daring, muncul kebutuhan untuk melindungi informasi dari penyadapan atau pencurian data oleh pihak yang tidak berwenang.

Alternatif keamanan yang digunakan untuk menangani kerahasiaan informasi biasa dikenal dengan istilah kriptografi. Kriptografi berfokus pada penyandian pesan sehingga hanya pihak yang berwenang yang dapat memahami informasi yang tersimpan. Enkripsi yang kuat dapat meminimalisir terjadinya kebocoran data. Namun, meskipun pesan telah terenkripsi, jejak dari pesan tersebut tetap terlihat. Hal ini menjadi celah bagi pihak lain untuk mendeteksi adanya pesan rahasia. Untuk menghindari kecurigaan tersebut, kriptografi dapat dikombinasikan dengan teknik steganografi. Steganografi menyembunyikan eksistensi pesan di dalam media pembawa (cover media) sehingga pihak lain tidak mengetahui keberadaan pesan tersebut. Dengan menggabungkan kedua metode ini, diharapkan pesan tidak hanya terlindungi secara kriptografi tetapi juga tersamarkan keberadaannya.

Salah satu teknik steganografi yang populer adalah Least Significant Bit (LSB). Teknik ini bekerja dengan mengganti bit paling rendah atau bit paling kanan pada data piksel yang menyusun file gambar digital. Sementara itu, metode Advanced Encryption Standard (AES) adalah algoritma kriptografi simetris yang kuat dan cepat, yang banyak digunakan dalam berbagai aplikasi keamanan informasi. Pengamanan menggunakan algoritma AES memungkinkan informasi digital tidak terbaca oleh pihak tidak berwenang (Jayana, M. A., 2022). Kombinasi antara steganografi LSB dan kriptografi AES menawarkan lapisan keamanan ganda, di mana pesan tidak hanya dienkripsi tetapi juga disembunyikan.

Penelitian ini menitikberatkan pada pengembangan teknik steganografi berbasis Least Significant Bit (LSB). Metode ini digunakan untuk menyisipkan informasi secara tersembunyi dalam gambar digital dengan mengubah bit-bit yang paling tidak signifikan pada piksel gambar. Untuk meningkatkan keamanan dan kerahasiaan data yang disisipkan, penelitian ini juga mengintegrasikan algoritma enkripsi Advanced Encryption Standard (AES). Kombinasi kedua teknik ini diharapkan mampu menyediakan lapisan perlindungan ganda, yaitu dengan menyembunyikan data secara visual sekaligus mengenkripsi kontennya untuk mencegah akses tidak sah. Implementasi ini dirancang untuk menghadirkan solusi yang lebih andal dalam menjaga kerahasiaan informasi digital.

Tujuan penelitian ini dirumuskan sebagai berikut:

1. Mengimplementasikan steganografi dengan metode Least Significant Bit (LSB) untuk menyisipkan informasi secara tersembunyi dalam gambar digital.
2. Mengintegrasikan algoritma kriptografi Advanced Encryption Standard (AES) untuk memberikan lapisan keamanan tambahan pada pesan yang disisipkan.
3. Mengembangkan aplikasi steganografi yang memanfaatkan metode LSB dan enkripsi AES untuk meningkatkan keamanan informasi digital.

Sebagai bagian dari evaluasi, penelitian ini bertujuan untuk menganalisis efektivitas metode yang diusulkan dalam menyembunyikan informasi dan mencegah deteksi oleh pihak ketiga. Penelitian ini juga bertujuan untuk mengukur performa metode dalam melindungi informasi yang disisipkan, dengan harapan dapat memberikan solusi yang lebih andal dalam menjaga kerahasiaan informasi digital.

II. METODOLOGI PENELITIAN

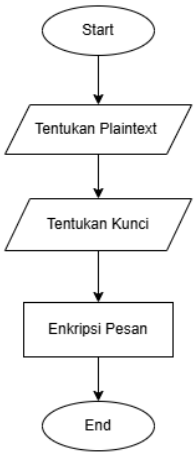
Metode yang akan digunakan dalam penelitian ini terdiri dari empat bagian utama, yaitu enkripsi, embedding, ekstraksi dan dekripsi. Proses ini mencakup langkah-langkah untuk mengenkripsi pesan, menyisipkan pesan ke dalam gambar, mengekstraksi pesan yang disisipkan dan mendekripsi pesan.

A. Enkripsi

Enkripsi adalah metode keamanan yang mengacak data sehingga hanya dapat dibaca dengan menggunakan kunci.

1. Memilih dan menentukan pesan teks (plaintext).
2. Memilih dan menentukan kunci (key).
3. Melakukan enkripsi pada pesan dan kunci menggunakan metode AES.

E-ISSN: 2985-3532

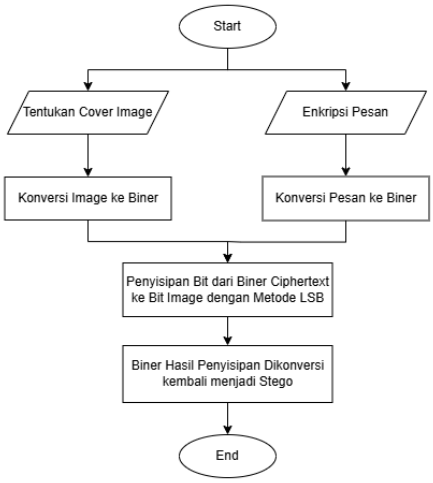


Gambar 1. Proses enkripsi

B. Embedding

Embedding adalah skema penyisipan pesan rahasia ke dalam cover-file. File yang disisipkan maupun yang disisipi pesan dapat berupa citra (gambar), audio, video, teks dan lain-lain.

1. Konversi gambar menjadi biner.
2. Ciphertext selanjutnya dikonversi menjadi biner.
3. Selanjutnya melakukan penyisipan biner ciphertext pada biner gambar dengan metode LSB, setiap bit dari ciphertext disisipkan pada bit terakhir dari gambar.



Gambar 2. Proses embedding

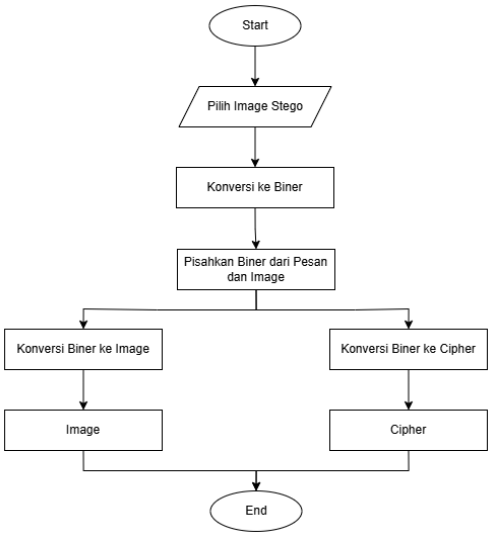
C. Ekstraksi

Merupakan proses mengembalikan dan memisahkan pesan dari cover menjadi bentuk semula.

1. Pilih dan tentukan gambar (stego image).
2. Lalu gambar stego dikonversi menjadi biner.

E-ISSN: 2985-3532

3. Melakukan pemisahan biner ciphertext dan biner gambar.
4. Biner gambar diubah menjadi desimal kemudian dipetakan menjadi gambar.

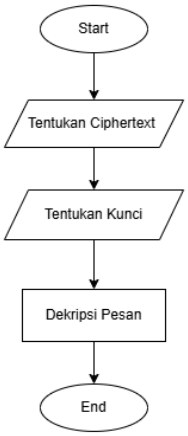


Gambar 3. Proses ekstraksi

D. Dekripsi

Dekripsi adalah proses mengubah data terenkripsi kembali ke bentuk yang dapat dibaca.

1. Memilih dan menentukan pesan teks (Ciphertext).
2. Menginputkan kunci yang telah dibuat sebelumnya kemudian diproses dengan perintah dekripsi.



Gambar 4. Proses dekripsi

E. Pengujian Kualitas Hasil Menggunakan PSNR (Peak Signal-to-Noise Ratio)

PSNR adalah parameter yang digunakan untuk mengukur kualitas gambar hasil steganografi dengan membandingkan gambar asli (cover image) dan gambar hasil

E-ISSN: 2985-3532

penyisipan pesan (stego image). Nilai PSNR diukur dalam desibel (dB) dan memberikan indikasi tingkat perbedaan antara kedua gambar tersebut.

Langkah-langkah pengujian kualitas menggunakan PSNR adalah sebagai berikut:

1. Menghitung Mean Squared Error (MSE) antara gambar asli dan gambar stego. MSE adalah rata-rata kuadrat selisih intensitas piksel antara dua gambar.
2. Menggunakan nilai MSE untuk menghitung PSNR dengan rumus:
3. Di mana adalah nilai maksimum intensitas piksel (misalnya 255 untuk gambar 8-bit).

$$PSNR = 10 \times \log_{10}\left(\frac{\text{MAX}^2}{MSE}\right) \quad (1)$$

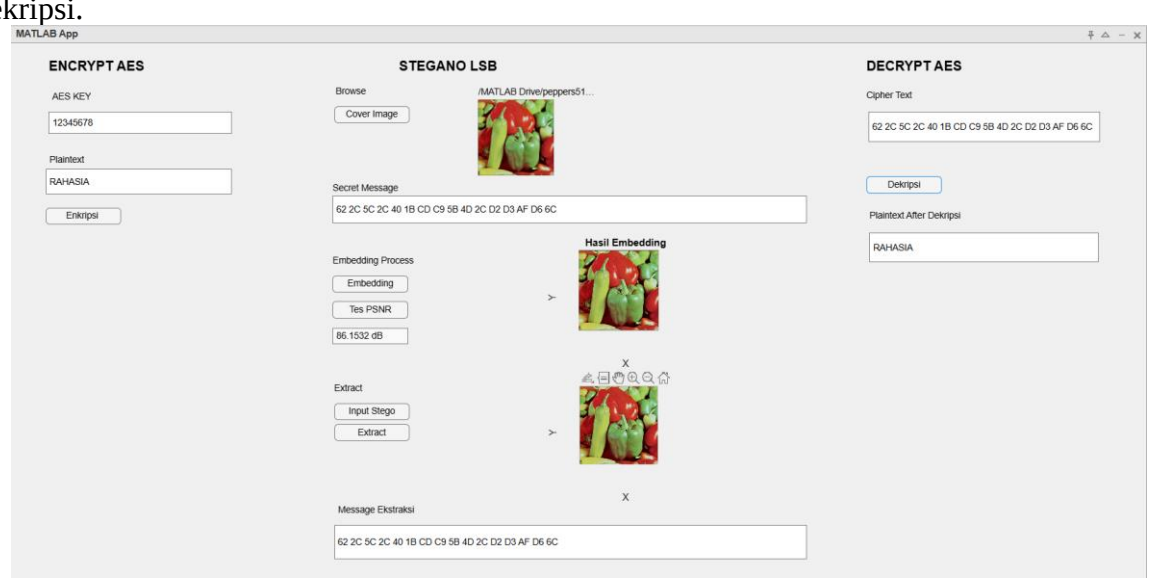
Di mana :

- **MAX** adalah nilai maksimum piksel dari gambar (misalnya, 255 untuk gambar 8-bit per channel).
- **MSE** (Mean Squared Error) adalah rata-rata kesalahan kuadrat antara gambar asli dan gambar hasil kompresi. MSE dihitung dengan rumus:

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(I_{\text{original}}(i) - I_{\text{compressed}}(i))^2 \quad (2).$$

III. HASIL DAN PEMBAHASAN

Pada proses implementasi steganografi LSB terdapat empat tahapan yang dilakukan yaitu enkripsi, embedding (penyisipan pesan), ekstraksi (memisahkan pesan dengan cover) dan dekripsi pesan, berikut merupakan proses enkripsi, embedding citra, ekstraksi citra dan dekripsi.



Gambar 5. aplikasi stegano pada matlab

Pada gambar di atas merupakan hasil implementasi dari kriptografi metode AES-128 bit dengan Steganografi metode Least Significant Bit.

A. Enkripsi

Proses pengamanan data pada algoritma AES bergantung pada ukuran kunci yang dipilih. Penelitian ini menggunakan ukuran kunci sebesar 128 bit. Sehingga jumlah roundyang dilakukan sebanyak 10 round sebagai berikut. Plaintext **"RAHASIA"**, Cipherkey **"12345678".**

Dari plain text dan chipper text tersebut, maka langkah awal adalah memasukkan ke dalam kolom 4×4 sebagai berikut.

Plaintext

R	A	H	A
S	I	A	Z
Z	Z	Z	Z
Z	Z	Z	Z

Cipherkey

1	2	3	4
5	6	7	8
0	0	0	0
0	0	0	0

Kemudian keduanya dikonversikan ke dalam bilangan heksadesimal menggunakan tabel ASCII, dan hasil konversi sebagai berikut

Hasil Konversi Plaintext

4B	4A	4B	4B
45	4C	4F	4D
50	4F	4B	30
36	50	49	46

Hasil Konversi Chiperkey

32	30	31	39
32	30	32	33
30	30	30	30
30	30	30	30

Selanjutnya dilakukan konversi ke dalam bilangan biner, dan hasil konversi sebagai berikut.

Hasil Biner Plain Text

01001011	01001010	01001011	01001011
01000101	01001100	01001111	01001101
01010000	01001111	01001011	00110000
00110110	01010000	01001001	01000110

Konversi Biner Chiper Key

00110010	00110000	00110001	00111001
00110010	00110000	00110010	00110011
00110000	00110000	00110000	00110000
001100	001100	001100	001100

E-ISSN: 2985-3532

00	00	00	00

Initial Round

Adapun hasil dan proses add round keydari tahap ini sebagai berikut.

Add Round Key

Tahap pertama ini sebagai berikut.

4B XOR 32 = 01001011 XOR 00110010 = 01111001

45 XOR 32 = 01000101 XOR 00110010 = 01110111

50 XOR 30 = 01010000 XOR 00110000 = 01100000

36 XOR 30 = 00110110 XOR 00110000 = 00000110

Dengan cara yang sama, maka didapatkan hasil biner sebagai berikut.

01001011	01001010	01001011	01001011
01000101	01001100	01001111	01001101
01010000	01001111	01001011	00110000
00110110	01010000	01001001	01000110

Dari hasil biner tersebut, dilakukan konversi ke bentuk heksadesimal sehingga hasil pada tahap initial round sebagai berikut.

79	73	7F	7B
73	75	74	79
6E	72	7F	78
61	77	79	78

Round 0

Pada round ini hasil akhir initialround akan dilanjutkan ke tahap selanjutnya sesuai dengan algoritma yang ada yakni sub bytes, shift rows, mix coloum, dan add round key. Hasil dari tiap tahap sebagai berikut.

Sub Bytes

Tahap ini, melakukan transformasi dari hasil yang didapatkan dengan tabel S-Box.

Sub Bytes ->

79	73	7F	7B
73	75	74	79
6E	72	7F	78
61	77	79	78

79	73	7F	7B
73	75	74	79
6E	72	7F	78
61	77	79	78

Shift Rows

Tahap ini melakukan pergeseran pada tiap kolom sesuai dengan ketentuan yang ada. Sehingga hasil dari tahap ini sebagai berikut.

Shift Rows ->

B6	8F	D2	21
8F	9D	92	B6
9F	40	D2	BC
EF	F5	B6	BC

B6	8F	D2	21
9D	92	B6	8F
D2	BC	9F	40
BC	EF	F5	B6

E-ISSN: 2985-3532

Mix Coloumn

Tahap ini akan dilakukan perkalian dari hasil akhir tahap sebelumnya dengan blok matriks polinomial tetap. Sebagai contoh perhitungan, berikut contoh perhitungan pada bit pada baris 1 kolom 1 (S10,0)

Maka hasil dari mix coloumn sebagai berikut

5F	B9	4D	E0
CD	33	70	4A
AC	12	D6	CB
80	37	2E	90

Add Round Key

Tahap akhirpada round 0 ini adalah melakukan add round key dari hasil mix coloumn dengan hasil pada initial round. Sehingga hasil pada tahap ini sebagai berikut.

5F	B9	4D	E0
CD	33	70	4A
AC	12	D6	CB
80	37	2E	90

XOR

79	73	7F	7B
55	3B	4E	3D
5E	52	49	48
49	53	49	49

Hasil dari proses XOR di atas sebagai berikut

26	C3	37	92
BA	4F	0D	34
CC	6D	AD	CB
86	57	57	E6

B. Embedding

1. Tentukan citra yang dijadikan sebagai penampung (cover) Citra penampung yang dijadikan cover image yaitu pappers.jpg dengan dimensi 4x4x3 size 102 bytes.

Gambar 6. cover image pappers.jpg

2. Baca nilai desimal citra cover, Resolusi citra yang dijadikan cover pesan yaitu citra RGB dengan ukuran 4x4x3, sehingga terdapat 48 nilai desimal, terdapat 16 bit pada masingmasing warna (red, green dan blue).

Tabel 1. Pixel RGBA

E-ISSN: 2985-3532

Red
179 61 40 255 168 64 44 255 133 43 30 255 131 58 35 255
Green
145 185 102 255 149 180 101 255 135 172 91 255 154 174 89 255
Blue
212 213 217 255 212 219 223 255 210 215 217 255 216 224 225 255

3. Konversi nilai desimal ke biner.

Tabel 2. Konversi desimal ke biner

Red
10110011 00111101 00101000 11111111 10101000 01000000 00101100 11111111 10000101 00101011 00011110 11111111 10000011 00111010 00100011 11111111
Green
10010001 10111001 01100110 11111111 10010101 10110100 01100101 11111111 10000111 10101100 01011011 11111111 10011010 10101110 01011001 11111111
Blue
11010100 11010101 11011001 11111111 11010100 11011011 11011111 11111111 11010010 11010111 11011001 11111111 11011000 11100000 11100001 11111111

4. Ciphertext dikonversi ke dalam biner, dan selanjutnya akan disisipkan pada citra cover.

Tabel 3. Biner ciphertext

01110010 01100001 01101000 01100001 01110011 01101001 01100001

5. Proses penukaran bit (steganografi dengan metode LSB), yaitu menyisipkan pesan pada citra cover, dapat dilakukan apabila jumlah biner pesan dapat

E-ISSN: 2985-3532

ditampung pada citra cover berdasarkan kriteria perhitungan jumlah pixel dibagi 8 bit.

Ciphertext akan disisipkan pada biner pixel citra cover berdasarkan metode LSB menggunakan perulangan pada baris dan kolom, yaitu masing-masing bit pada ciphertext disisipkan secara bergiliran pada pixel warna red, green dan blue. Biner terakhir cover diganti dengan biner ciphertext 0 atau 1. Berikut merupakan hasil penyisipan ciphertext pada citra cover:

Tabel 4. Penyisipan biner ciphertext pada citra

Red
10110010 00111101 00101001 11111110 10101000 01000001 00101100 11111111 10000101 00101011 00011111 11111111 10000011 00111010 00100010 11111111
Green
10010000 10111000 01100110 11111111 10010101 10110100 01100100 11111111 10000111 10101101 01011010 11111111 10011010 10101111 01011000 11111111
Blue
11010100 11010100 11011000 11111111 11010100 11011011 11011111 11111111 11010010 11010111 11011001 11111111 11011000 11100001 11100000 11111111

6. Ciphertext akan disisipkan pada biner pixel citra cover, berdasarkan metode LSB yaitu masing-masing bit pada ciphertext disisipkan pada bit terakhir citra cover.

7. Hasil dilakukan penyisipan, hasil dari nilai biner cover baru di konversi kembali ke bentuk bilangan desimal dan kemudian dipetakan menjadi citra baru yang disebut stego image.

Gambar 6. stego image

C. Ekstraksi

Proses ekstraksi merupakan kebalikan dari proses embedding, dimana citra yang berisi pesan akan dipisahkan kembali, yang terdiri atas cover image (media penampung), kunci, dan plaintext (pesan).

E-ISSN: 2985-3532

1. Masukkan/pilih citra yang telah disisipkan pesan teks (stego image).

Gambar 7. stego image

2. Baca nilai pixel stego image selanjutnya konversi ke biner.

tabel 5. Konversi desimal ke biner

Red			
178	61	41	254
168	65	44	255
133	43	31	255
131	58	34	255
Green			
144	184	102	255
149	180	100	255
135	173	90	255
154	175	88	255
Blue			
212	212	216	255
212	219	223	255
210	215	217	255
216	225	224	255

tabel 6. Pixel stego image

Red
10110010 00111101 00101001 11111110
10101000 01000001 00101100 11111111
10000101 00101011 00011111 11111111
10000011 00111010 00100010 11111111
Green
10010000 10111000 01100110 11111111
10010101 10110100 01100100 11111111
10000111 10101101 01011010 11111111
10011010 10101111 01011000 11111111
Blue
11010100 11010100 11011000 11111111
11010100 11011011 11011111 11111111

E-ISSN: 2985-3532

11010010 11010111 11011001 11111111 11011000 11100001 11100000 11111111

3. Ambil bit LSB dari setiap elemen pixel dimulai dari bit ke-9 hingga sejumlah perkalian kunci dengan 8 bit lalu tambahkan dengan 8 bit kunci LSB, kemudian kelompokkan nilai bit-bit LSB menjadi 8 kelompok, selanjutnya konversi ke bilangan desimal.

tabel 7 . Pixel awal cover

Red
10110011 00111101 00101000 11111111 10101000 01000000 00101100 11111111 10000101 00101011 00011110 11111111 10000011 00111010 00100011 11111111
Green
10010001 10111001 01100110 11111111 10010101 10110100 01100101 11111111 10000111 10101100 01011011 11111111 10011010 10101110 01011001 11111111
Blue
11010100 11010101 11011001 11111111 11010100 11011011 11011111 11111111 11010010 11010111 11011001 11111111 11011000 11100000 11100001 11111111

4. Setelah diperoleh bilangan desimal dari biner pengelompokkan, konversi ke karakter, karakter yang dihasilkan tersebutlah yang menjadi pesan yang telah disembunyikan sebelumnya.

Gambar 6. cover image pappers.jpg

Tabel 3. Biner ciphertext

01110010	01100001	01101000	01100001	01110011	01101001	01100001
114	97	104	97	115	105	97

D. Dekripsi

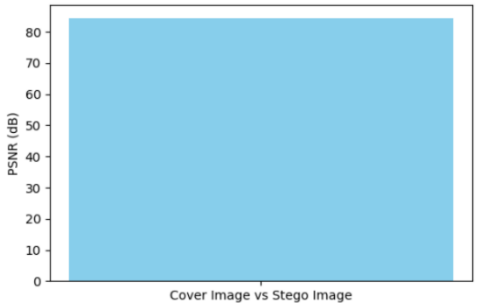
E-ISSN: 2985-3532

Proses mengubah informasi yang telah dienkripsi menjadi teks biasa atau plaintext, kebalikannya dari enkripsi. langkah langkahnya seperti dibawah ini.

1. **AddRoundKey**: XOR blok ciphertext dengan kunci putaran.
2. **Inverse ShiftRows**: Kebalikan dari ShiftRows, baris-blok diputar dalam arah yang berlawanan.
3. **Inverse SubBytes**: Penggantian byte menggunakan S-box terbalik (Inverse S-box).
4. **Inverse MixColumns**: Proses mencampur kolom data (pada semua putaran kecuali yang terakhir).
5. **Repeat**: Langkah-langkah ini diulang sebanyak jumlah putaran, kecuali pada putaran terakhir yang hanya melibatkan **AddRoundKey**, **Inverse SubBytes**, dan **Inverse ShiftRows**.

E. Pengujian PSNR

Pengujian PNSR (Peak Signal to Noise Ratio) digunakan untuk mengukur kualitas citra yang dihasilkan. Metode PNSR adalah ukuran perbandingan antara nilai piksel cover image dengan nilai piksel pada citra stego yang dihasilkan.



Gambar 7. Hasil pengecekan PSNR dengan nilai **86.1532 dB.**

IV. KESIMPULAN

Penelitian ini berhasil mengimplementasikan metode steganografi berbasis Least Significant Bit (LSB) yang terintegrasi dengan algoritma enkripsi Advanced Encryption Standard (AES) untuk meningkatkan keamanan informasi digital. Proses enkripsi AES memastikan bahwa data yang disisipkan tidak dapat dibaca oleh pihak tidak berwenang, sementara teknik LSB menyembunyikan keberadaan data dalam gambar digital. Pengujian dilakukan menggunakan parameter Peak Signal-to-Noise Ratio (PSNR) untuk mengevaluasi kualitas gambar setelah penyisipan data. Hasil pengujian menunjukkan bahwa metode ini mampu mempertahankan kualitas visual gambar yang baik, dengan nilai PSNR yang sesuai standar sehingga sulit dideteksi oleh pihak ketiga. Implementasi algoritma dilakukan menggunakan MATLAB, yang mempermudah integrasi antara proses enkripsi dan steganografi serta memberikan platform yang efektif untuk analisis dan visualisasi hasil. Metode yang dikembangkan dalam penelitian ini memberikan solusi yang andal untuk menyembunyikan informasi secara aman dan efisien dalam lingkungan digital, sekaligus membuktikan bahwa kombinasi AES dan LSB dapat meningkatkan keamanan data tanpa

E-ISSN: 2985-3532

mengorbankan kualitas media pembawa.Kesimpulan ditulis dengan singkat pada hasil penelitian.

(Azhari et al., 2022; Cristy & Riandari, 2021; Jayana et al., n.d.; Satya & Dwiatma, 2021; Wachid Hidayatulloh et al., 2023)

VI. REFERENSI

[1] Azhari, M., Perwitosari, J., & Ali, F. (2022). Implementasi Pengamanan Data pada Dokumen Menggunakan Algoritma Kriptografi Advanced Encryption Standard (AES). *Jurnal Pendidikan Sains Dan Komputer*, *2*(1), 2809–476. https://doi.org/10.47709/jpsk.v2i1.1390

[2] Cristy, N., & Riandari, F. (2021). *Niolinda Cristy 1 , Fristi Riandari 2 [Implementasi Metode Advanced Encryption Standard (AES 128 Bit) Untuk Mengamankan Data Keuangan*. *4*(2), 75.

[3] Jayana, M. A., Rafael, D., & Rahman, A. A. (n.d.). *IMPLEMENTASI PENGAMANAN DATA PENGARSIPAN DENGAN METODE ALGORITMA KRIPTOGRAFI AES STUDI KASUS PADA BANK BJB KCP PASTEUR BANDUNG*.

[4] Satya, R., & Dwiatma, T. G. (2021). *ANALISA STEGANOGRAFI DENGAN METODE BPCS (Bit-Plane Complexity Segmentation) DAN LSB (Least Significant Bit)PADA PENGOLAHAN CITRA*.

[5] Wachid Hidayatulloh, N., Tahir, M., Amalia, H., Afdlolul Basyar, N., Prianggara, A. F., & Yasin, M. (2023). Mengenal Advance Encrytion Standard (AES) Sebagai Algoritma Kriptografi Dalam Mengamankan Data. *Digital Transformation Technology (Digitech) | e*, *3*(1). https://doi.org/10.47709/digitech.v3i1.2293

E-ISSN: 2985-3532