 ISSN:2302-8149
e-ISSN:2540-9719

Implementasi Algoritma *Advanced Encryption Standard* (AES) dan Teknik Steganografi *Bit Plane Complexity Segmentation* (BPCS) dalam Eskalasi Keamanan File Teks

Implementation of the Advanced Encryption Standard (AES) Algorithm and Bit Plane Complexity Segmentation (BPCS) Steganography Technique for Enhancing Text File Security

[1]Afthar Kautsar*, [2]Muhammad Ikhsan
[1,2]Program Studi Ilmu Komputer, Fakultas Sains dan Teknologi, Universitas Islam Negeri Sumatera Utara, Medan, Sumatera Utara, Indonesia
*e-mail: jjtaryoung21@gmail.com

(***received*:** 22 February 2025, ***revised*:** 25 February 2025, ***accepted*:** 26 February 2025)

Abstrak

Di era digital saat ini, keamanan data menjadi aspek yang sangat penting mengingat banyaknya informasi yang dikirim dan disimpan secara elektronik. *Advanced Encryption Standard* (AES) merupakan salah satu algoritma enkripsi yang banyak digunakan karena tingkat keamanannya yang tinggi. Namun, data yang hanya dienkripsi masih berisiko terdeteksi, sehingga teknik steganografi *Bit-Plane Complexity Segmentation* (BPCS) digunakan untuk menyembunyikan data dalam gambar digital agar lebih sulit diidentifikasi. Penelitian ini bertujuan untuk meningkatkan keamanan data dengan menggabungkan enkripsi AES dan steganografi BPCS. Proses dimulai dengan mengenkripsi data teks menggunakan AES-128 agar informasi tidak dapat diakses tanpa kunci yang valid. Selanjutnya, *ciphertext* yang dihasilkan disisipkan ke dalam citra digital menggunakan teknik BPCS, yang memilih bit-*plane* dengan kompleksitas tinggi untuk penyisipan tanpa mengubah tampilan visual gambar secara signifikan. Pengujian dilakukan dengan mengukur keakuratan ekstraksi data serta dampak terhadap kualitas gambar. Hasil penelitian menunjukkan bahwa metode ini dapat menjaga keamanan dan kerahasiaan data dengan baik. Data yang telah disisipkan dalam gambar dapat diekstraksi kembali dengan tingkat keberhasilan 100%, sementara kualitas gambar tetap terjaga tanpa perubahan mencolok. Dengan demikian, kombinasi AES dan BPCS memberikan lapisan perlindungan ganda: enkripsi memastikan data tidak dapat dibaca, sementara steganografi menyembunyikan keberadaan data dari pihak yang tidak berwenang. Metode ini dapat diterapkan dalam berbagai skenario keamanan data, baik untuk keperluan pribadi maupun organisasi.

Kata kunci: keamanan data, enkripsi AES, steganografi BPCS, bit-*plane complexity segmentation*

Abstract

In today's digital era, data security has become a crucial aspect due to the vast amount of information being transmitted and stored electronically. The Advanced Encryption Standard (AES) is one of the most widely used encryption algorithms because of its high level of security. However, data that is merely encrypted still risks detection; therefore, the Bit-Plane Complexity Segmentation (BPCS) steganography technique is employed to hide data within digital images, making it more difficult to identify. This study aims to enhance data security by combining AES encryption and BPCS steganography. The process begins by encrypting text data using AES-128, ensuring that the information cannot be accessed without a valid key. Next, the resulting ciphertext is embedded into a digital image using the BPCS technique, which selects bit-planes with high complexity for insertion without significantly altering the visual appearance of the image. Testing is conducted by measuring the accuracy of data extraction and the impact on image quality. The results indicate that this method effectively maintains data security and confidentiality. The data embedded in the image can be extracted with a success rate of 100%, while the image quality remains intact without noticeable changes. Thus, the combination of AES and BPCS provides a dual layer of protection: encryption ensures that the data cannot be read, while steganography conceals the existence of the data from

 ISSN:2302-8149
e-ISSN:2540-9719

unauthorized parties. This method can be applied in various data security scenarios, both for personal and organizational purposes.

Keywords: *data security, AES encryption, BPCS steganography, bit-plane complexity segmentation*

1 Pendahuluan

Keamanan data telah menjadi aspek yang krusial di era digital, di mana jumlah data yang dikirim dan disimpan secara elektronik terus meningkat. Ancaman terhadap keamanan informasi, seperti pencurian data dan serangan siber, terus berkembang dengan metode yang semakin canggih. Salah satu tantangan utama dalam keamanan data adalah menjaga kerahasiaan informasi dari akses yang tidak sah. Oleh karena itu, diperlukan solusi yang tidak hanya mengandalkan enkripsi tetapi juga menyembunyikan keberadaan data agar lebih sulit dideteksi.

Salah satu metode kriptografi yang banyak digunakan adalah *Advanced Encryption Standard* (AES). Algoritma ini telah menjadi standar enkripsi global karena tingkat keamanannya yang tinggi serta efisiensinya dalam proses enkripsi dan dekripsi [1]. AES mengubah data menjadi bentuk yang tidak dapat dibaca tanpa kunci yang sesuai, sehingga sangat efektif dalam menjaga kerahasiaan informasi. Namun, data yang telah dienkripsi masih dapat menarik perhatian pihak yang tidak berwenang, sehingga diperlukan metode tambahan untuk menyembunyikan keberadaannya.

Selain enkripsi, steganografi adalah teknik yang umum digunakan untuk meningkatkan keamanan data. Steganografi merupakan metode untuk menyembunyikan informasi di dalam media lain, seperti gambar atau audio, sehingga informasi yang disisipkan tidak terdeteksi oleh pihak yang tidak berwenang [2]. Salah satu teknik steganografi yang efektif adalah *Bit-Plane Complexity Segmentation* (BPCS), yang dapat menyisipkan data ke dalam gambar digital dengan memodifikasi bit-plane yang memiliki kompleksitas tinggi. Metode ini memungkinkan penyisipan informasi tanpa mengubah tampilan visual gambar secara signifikan [3].

AES (*Advanced Encryption Standard*) adalah metode kriptografi yang banyak digunakan karena tingkat keamanannya yang tinggi dan efisiensinya. Metode ini mengubah data ke dalam format yang tidak dapat dibaca, sehingga memastikan kerahasiaan informasi. Namun, data yang telah dienkripsi masih dapat menarik perhatian, sehingga diperlukan metode tambahan untuk menyembunyikan keberadaannya [4].

Steganografi adalah teknik yang digunakan untuk meningkatkan keamanan data dengan menyisipkan informasi ke dalam media lain, seperti gambar, sehingga tidak terdeteksi. *Bit-Plane Complexity Segmentation* (BPCS) merupakan metode steganografi yang efektif dengan memodifikasi bit-plane kompleks dalam gambar digital untuk menyisipkan data tanpa menyebabkan perubahan visual yang signifikan [5].

Dengan menggabungkan enkripsi AES dan steganografi BPCS, dapat dicapai sistem keamanan data yang lebih kuat. Enkripsi AES mengubah data menjadi format yang tidak dapat dibaca, sementara steganografi BPCS menyembunyikan data yang telah dienkripsi di dalam gambar digital untuk mencegah pendeteksian. Kombinasi ini diharapkan dapat meningkatkan perlindungan data secara lebih aman dan efisien. Namun, hingga saat ini, masih sedikit penelitian yang secara langsung mengintegrasikan kedua metode tersebut dalam satu sistem.

Beberapa penelitian sebelumnya telah meneliti penggunaan enkripsi dan steganografi dalam keamanan data. Penelitian yang dilakukan Olivia *et al*, [6] dalam jurnal *Implementation of Cryptography in Data Security Using the Advanced Encryption Standard* (AES) *Algorithm* membahas penggunaan algoritma AES untuk meningkatkan keamanan data dalam sistem informasi yang kompleks. Hasil penelitian menunjukkan bahwa AES mampu mengubah data menjadi bentuk yang tidak dapat dibaca tanpa kunci yang sesuai, menjadikannya pilihan yang efektif dalam menjaga kerahasiaan informasi. Namun, penelitian ini hanya berfokus pada implementasi AES tanpa mengintegrasikannya dengan metode perlindungan tambahan seperti steganografi.

Sementara itu, Widyanarko [7] dalam jurnal *Implementation of Steganography Using the Bit-Plane Complexity Segmentation (BPCS) Method for Compressed Image Documents* meneliti penggunaan metode BPCS untuk menyembunyikan informasi rahasia dalam gambar digital terkompresi. Penelitian ini mengembangkan perangkat lunak *Secret-Postcard* untuk menyisipkan pesan ke dalam gambar berformat JPEG, GIF, dan PNG. Namun, penelitian ini tidak mengombinasikan BPCS dengan enkripsi sebelumnya, sehingga data yang disisipkan tetap dalam bentuk aslinya dan berisiko jika ditemukan oleh pihak yang tidak berwenang.

 ISSN:2302-8149
e-ISSN:2540-9719

Selain itu, penelitian oleh Hakim dan Baihaqi [8] dalam jurnal *Implementation of a Web-Based Steganography Application Using the LSB and BPCS Algorithms* mengembangkan aplikasi berbasis web yang mengintegrasikan algoritma *Least Significant Bit* (LSB) dan BPCS untuk menyisipkan informasi dalam gambar digital. Aplikasi ini dirancang agar mudah diakses melalui browser dengan antarmuka yang dibangun menggunakan HTML, CSS, dan JavaScript. Namun, penelitian ini lebih berfokus pada aspek implementasi sistem dan tidak secara khusus membahas keamanan data sebelum disisipkan ke dalam gambar.

Sementara itu, Zai dan Mesran [9] dalam jurnal *Analysis of Digital Image Processing in RGB Images* meneliti berbagai teknik pemrosesan gambar, termasuk peningkatan kualitas visual dan segmentasi gambar. Hasil penelitian menunjukkan bahwa metode pemrosesan gambar yang diterapkan dapat meningkatkan akurasi dalam analisis gambar dan mempertahankan kualitas informasi visual. Namun, penelitian ini tidak berfokus pada aspek keamanan data melalui enkripsi atau steganografi.

Selanjutnya, Cisco dan Tracer [10] dalam jurnal *Implementation of the Advanced Encryption Standard (AES) Algorithm in Digital Data Security* meneliti efisiensi enkripsi dan dekripsi AES dalam berbagai skenario penggunaan. Hasil penelitian menunjukkan bahwa AES dapat mengenkripsi dan melindungi data dengan tingkat keamanan tinggi, sehingga informasi menjadi lebih sulit diakses oleh pihak yang tidak berwenang. Namun, penelitian ini tidak mempertimbangkan ancaman yang timbul akibat data yang telah dienkripsi masih dapat menarik perhatian dari pihak yang tidak berwenang.

Berdasarkan penelitian terkait, ditemukan bahwa masih sedikit penelitian yang mengintegrasikan enkripsi AES dengan metode steganografi BPCS. Sebagian besar penelitian hanya menggunakan enkripsi atau steganografi secara terpisah, atau mengombinasikan AES dengan teknik steganografi lain yang kurang optimal dalam menyembunyikan data berukuran besar. Oleh karena itu, penelitian ini bertujuan untuk menjembatani kesenjangan tersebut dengan mengembangkan sistem yang mengintegrasikan kedua metode secara optimal.

Penelitian ini bertujuan untuk mengembangkan sistem keamanan data berbasis kombinasi enkripsi AES dan steganografi BPCS, menganalisis efektivitas metode ini dalam menyembunyikan data terenkripsi tanpa mengurangi kualitas gambar, serta mengevaluasi kinerja enkripsi dan penyisipan data dari segi kecepatan, kapasitas, dan tingkat keamanan. Dengan adanya penelitian ini, diharapkan dapat diperoleh metode perlindungan data yang lebih aman, efisien, dan sulit dideteksi. Implementasi sistem ini diharapkan dapat diterapkan dalam berbagai bidang, seperti komunikasi digital, penyimpanan data rahasia, dan sistem keamanan informasi guna meningkatkan perlindungan terhadap data sensitif.

2 Tinjauan Literatur

Berbagai penelitian telah dilakukan terkait kombinasi algoritma kriptografi dan steganografi untuk meningkatkan keamanan data. Salah satu studi mengimplementasikan AES untuk enkripsi data sebelum disembunyikan dalam gambar menggunakan steganografi LSB yang dimodifikasi. Penelitian ini mengusulkan metode penyisipan bit secara diagonal dalam matriks piksel guna meningkatkan keamanan tanpa mengubah tampilan visual secara signifikan. Hasil penelitian menunjukkan bahwa modifikasi ini efektif dalam menyembunyikan ciphertext dan memastikan bahwa pesan dapat diekstrak kembali dengan baik [11]. Meskipun metode ini cukup aman, penggunaan LSB tetap memiliki kelemahan karena masih rentan terhadap analisis statistik dan serangan deteksi steganalisis, sehingga diperlukan pendekatan yang lebih kompleks seperti BPCS untuk meningkatkan keamanan data.

Penelitian lain menggunakan steganografi BPCS untuk menyembunyikan teks terenkripsi menggunakan RSA ke dalam citra digital. RSA digunakan untuk mengenkripsi pesan sebelum disisipkan, meningkatkan keamanan informasi yang dikirim melalui jaringan internet. Hasil penelitian ini menunjukkan bahwa kombinasi RSA dan BPCS membuat pesan lebih sulit dideteksi secara visual, memberikan perlindungan lebih baik terhadap penyadapan [12]. Namun, dalam konteks efisiensi, RSA memiliki kelemahan dalam hal kecepatan proses enkripsi dan dekripsi dibandingkan dengan AES, terutama ketika digunakan untuk file teks yang lebih besar. Selain itu, penelitian ini tidak

 ISSN:2302-8149
e-ISSN:2540-9719

membahas secara mendalam bagaimana kompleksitas bit-plane dapat dioptimalkan dalam proses penyisipan, yang menjadi celah untuk eksplorasi lebih lanjut.

Studi lainnya menggabungkan AES-256 dengan steganografi LSB dalam gambar bitmap untuk menjaga keamanan pesan. AES-256 digunakan untuk mengenkripsi data sebelum disisipkan menggunakan LSB, sementara SHA-1 digunakan untuk memastikan integritas data, dan PSNR (*Peak Signal-to-Noise Ratio*) digunakan untuk mengukur kualitas gambar setelah penyisipan. Hasil pengujian menunjukkan bahwa rata-rata PSNR sebesar 44,14 dB, yang menandakan bahwa perubahan visual pada gambar setelah penyisipan sangat minim [13]. Meskipun pendekatan ini berhasil dalam menjaga keamanan dan kualitas gambar, metode ini tetap menggunakan LSB, yang masih memiliki kelemahan dari sisi ketahanan terhadap serangan analisis spektral. Oleh karena itu, metode steganografi yang lebih kompleks, seperti BPCS, dapat menjadi alternatif yang lebih baik untuk mengatasi keterbatasan ini.

Selain dalam gambar, penelitian lain mengimplementasikan steganografi BPCS pada file audio untuk menyembunyikan pesan rahasia. Teknik ini memungkinkan penyisipan pesan dalam bit-plane file audio tanpa menurunkan kualitas suara yang signifikan. Pesan yang akan disisipkan sebelumnya dienkripsi menggunakan Bit-*Plane,* yang bertujuan untuk meningkatkan keamanan dan mengurangi kemungkinan deteksi. Hasil penelitian menunjukkan bahwa metode ini efektif dalam melindungi informasi rahasia, tetapi penerapannya lebih difokuskan pada media audio dibandingkan dengan file teks [14]. Hal ini menunjukkan bahwa masih terdapat ruang eksplorasi lebih lanjut dalam penerapan steganografi BPCS untuk penyisipan teks yang dienkripsi menggunakan AES.

Dari sisi implementasi sistem keamanan file digital, penelitian lain telah mengembangkan aplikasi berbasis web menggunakan PHP dan MySQL yang mengimplementasikan AES untuk mengenkripsi berbagai jenis file, termasuk gambar, dokumen Word, PDF, Excel, dan PowerPoint. Sistem ini memungkinkan pengguna untuk mengunggah file yang akan dienkripsi dan menyimpannya dalam database. Hasil penelitian menunjukkan bahwa sistem ini efektif dalam melindungi file pribadi dari akses tidak sah [15]. Namun, penelitian ini tidak membahas teknik steganografi sebagai lapisan tambahan untuk menyembunyikan ciphertext, sehingga masih terbuka peluang untuk menggabungkan AES dengan steganografi dalam satu sistem.

Berdasarkan tinjauan literatur tersebut, masih terdapat celah penelitian yang dapat dikembangkan, terutama dalam integrasi AES dan steganografi BPCS secara langsung dalam satu system serta optimasi proses penyisipan berdasarkan kompleksitas bit-plane. Penelitian sebelumnya belum banyak membahas bagaimana kompleksitas bit-plane dapat digunakan sebagai parameter utama dalam pemilihan lokasi penyisipan pada teknik BPCS, yang dapat berpengaruh terhadap keamanan dan efisiensi penyisipan pesan. Oleh karena itu, penelitian ini berfokus pada penerapan AES dan steganografi BPCS dalam eskalasi keamanan file teks, dengan mempertimbangkan kompleksitas bit-*plane* sebagai faktor utama dalam penyisipan data terenkripsi. Dengan pendekatan ini, diharapkan keamanan informasi dapat ditingkatkan, baik dari segi enkripsi maupun dari sisi penyembunyian data, sehingga lebih sulit dideteksi dan dieksploitasi oleh pihak yang tidak berwenang.

3 Metode Penelitian

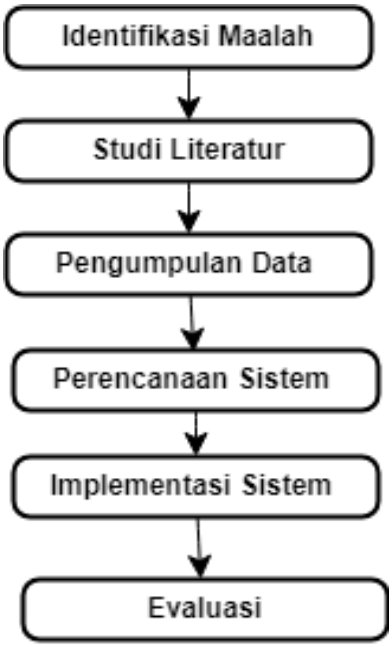
Metode penelitian yang digunakan dalam penelitian ini bertujuan untuk memberikan gambaran yang jelas tentang cara pengumpulan, analisis, dan interpretasi data yang relevan dengan masalah yang diteliti. Berikut rincian metode penelitian yang dilakukan dalam penelitian ini.

3.1 Kerangka Penelitian

Penelitian ini dilakukan secara sistematis melalui beberapa tahapan utama untuk mencapai tujuan utamanya, yaitu mengamankan file teks dengan mengombinasikan enkripsi AES dan steganografi BPCS dalam lingkungan Google Colab. Tahapan penelitian meliputi identifikasi masalah, studi literatur, pengumpulan data, perancangan sistem, implementasi, serta evaluasi dan pengujian. Tahap identifikasi masalah bertujuan untuk memahami tantangan utama dalam meningkatkan keamanan file teks, terutama dalam mengurangi risiko pendeteksian oleh pihak yang tidak berwenang.Studi literatur dilakukan untuk memperoleh pemahaman menyeluruh tentang cara kerja enkripsi AES dan steganografi BPCS, termasuk keunggulan dan keterbatasannya. Selanjutnya, tahap pengumpulan data merujuk pada berbagai sumber akademik untuk mendapatkan parameter keamanan yang sesuai, karakteristik BPCS, serta metode evaluasi yang tepat. Pada tahap perancangan sistem, dikembangkan

 ISSN:2302-8149
e-ISSN:2540-9719

model integrasi antara enkripsi AES dan steganografi BPCS dalam Google Colab menggunakan Python. Perancangan ini mencakup pemilihan pustaka pemrograman, pembuatan algoritma enkripsi dan dekripsi AES, serta perancangan metode penyisipan dan ekstraksi pesan menggunakan BPCS. Tahap implementasi dilakukan dengan membangun sistem secara menyeluruh, termasuk pengujian dengan berbagai ukuran file teks dan jenis citra digital untuk memastikan efektivitas metode yang diusulkan. Tahap terakhir adalah evaluasi dan pengujian, yang bertujuan untuk menilai kinerja sistem dalam berbagai aspek, seperti keakuratan enkripsi dan dekripsi, kualitas citra setelah penyisipan pesan, serta ketahanan terhadap analisis steganografi. Melalui tahapan penelitian ini, sistem yang dikembangkan diharapkan dapat secara optimal meningkatkan keamanan data dengan mengombinasikan enkripsi AES dan steganografi BPCS ditampilkan dalam Gambar 1.



Gambar 1. Kerangka penelitian

3.2 Proses Enkripsi dan Dekripsi AES

Proses enkripsi dan dekripsi AES (*Advanced Encryption Standard*) adalah metode kriptografi simetris yang digunakan untuk mengamankan data.

Proses Enkripsi AES sebagai berikut:

1. Mulai: yaitu menunjukkan titik awal program.
2. *Input Plaintext* dan *Key*: yaitu memasukkan *plaintext* (teks asli yang akan dienkripsi) dan *key* (kunci enkripsi).
3. Proses *Key Expansion*: yaitu proses kunci yang dimasukkan diperluas menjadi beberapa *round key* yang akan digunakan pada setiap ronde enkripsi.
4. *AddRoundKey* (*Plaintext* XOR dengan *Round Key* pertama): yaitu *plaintext* digabungkan dengan *round key* pertama menggunakan operasi XOR untuk memulai proses enkripsi.
5. Ronde 1-9 (*SubBytes, ShiftRows, MixColumns, AddRoundKey*): yaitu setiap ronde terdiri dari empat tahap berikut:
   a. *SubByte*: Setiap *byte* dalam blok *plaintext* digantikan dengan nilai dari tabel substitusi (*S-Box*), yang bertujuan menambah keamanan dengan meningkatkan keacakan.
   b. *ShiftRows*: Setiap baris *byte* digeser ke kiri dengan jumlah langkah yang berbeda.
   c. *MixColumns*: Setiap kolom pada blok *plaintext* dicampurkan menggunakan operasi matriks untuk menyebarkan data lebih merata.
   d. *AddRoundKey*: Blok data digabungkan dengan *round key* ronde menggunakan XOR.
6. Ronde ke-10 (*SubBytes, ShiftRows, AddRoundKey*): yaitu ronde terakhir memiliki tiga tahap yang sama dengan ronde sebelumnya, tetapi *mixcolumns* tidak diterapkan. Ini dilakukan agar struktur *cipher* final memiliki keunikan tambahan.

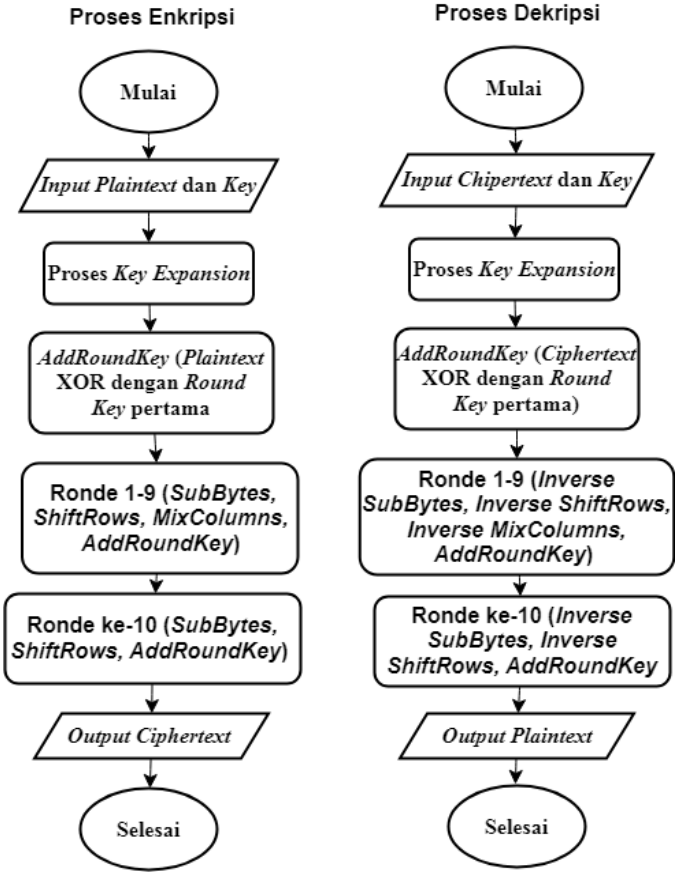
 ISSN:2302-8149
e-ISSN:2540-9719

7. *Output Ciphertext*: yaitu hasil akhir dimana teks asli kini menjadi *ciphertext* (teks terenkripsi).
8. Selesai: yaitu titik akhir dari *flowchart* yang menandakan proses enkripsi selesai.

Proses Dekripsi AES sebagai berikut:

1. Mulai: yaitu menunjukkan titik awal program.
2. Input *Ciphertext* dan *Key*: yaitu memasukkan *ciphertext* (teks terenkripsi yang akan didekripsi) dan key yang sama dengan yang digunakan saat enkripsi.
3. Proses *Key Expansion*: yaitu proses kunci yang dimasukkan diperluas menjadi beberapa round key yang akan digunakan pada setiap ronde dekripsi.
4. *AddRoundKey* (*Ciphertext* XOR dengan *Round Key* pertama): yaitu *chiphertext* digabungkan dengan *round key* terakhir (kebalikan dari enkripsi yang dimulai dengan *round key* pertama) menggunakan operasi XOR untuk memulai proses dekripsi.
5. Ronde 1-9 (*Inverse SubBytes*, *Inverse ShiftRows*, *Inverse MixColumns*, *AddRoundKey*): yaitu setiap ronde terdiri dari empat tahap berikut:
    a. *Inverse SubBytes*: Setiap *byte* dalam blok *ciphertext* digantikan dengan nilai kebalikannya dari tabel substitusi (*Inverse S-Box*).
    b. *Inverse ShiftRows*: Setiap baris *byte* digeser ke kanan (kebalikan dari enkripsi) untuk mengembalikan posisi *byte*.
    c. *Inverse MixColumns*: Setiap kolom pada blok *ciphertext* dicampurkan dengan operasi matriks terbalik untuk mengembalikan penyebaran data.
    d. *AddRoundKey*: Blok data digabungkan dengan *round key* ronde menggunakan XOR.
6. Ronde ke-10 (*Inverse SubBytes*, *Inverse ShiftRows*, *AddRoundKey*): yaitu ronde terakhir memiliki tiga tahap yang sama dengan ronde sebelumnya, tetapi *inverse mixcolumns* tidak diterapkan. Ini mencerminkan struktur yang sama pada proses enkripsi.
7. *Output Plaintext*: yaitu hasil akhir dimana *ciphertext* kini dikembalikan menjadi *plaintext* (teks asli).
8. Selesai: yaitu titik akhir dari *flowchart* yang menandakan proses dekripsi selesai.

Flowchart Proses Enkripsi dan Dekripsi AES dalam penelitian ini ditampilkan dalam Gambar 2.



Gambar 2. *Flowchart* proses enkripsi dan dekripsi AES

 ISSN:2302-8149
e-ISSN:2540-9719

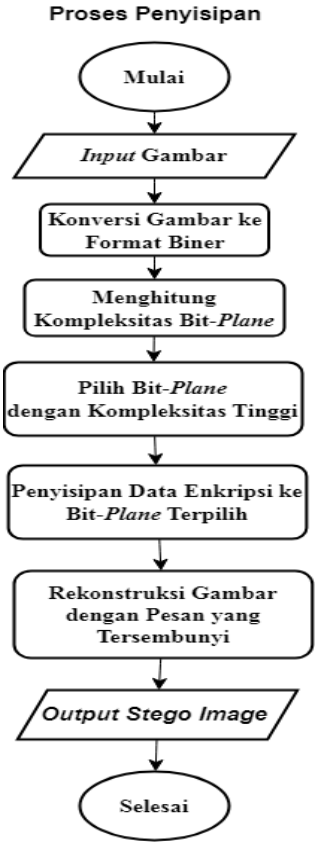
3.3 Proses Penyisipan dengan Steganografi BPCS

Proses penyisipan dengan steganografi BPCS (*Bit-Plane Complexity Segmentation*) adalah teknik yang digunakan untuk menyembunyikan informasi dalam citra digital tanpa merusak kualitas visualnya.

Proses Penyisipan dengan BPCS sebagai berikut:

1. Mulai: yaitu menunjukkan titik awal program.
2. *Input* Gambar: yaitu langkah untuk memasukkan gambar dan pesan teks yang telah dienkripsi dengan AES. Gambar akan digunakan sebagai media untuk menyisipkan pesan teks terenkripsi menggunakan teknik steganografi BPCS.
3. Konversi Gambar ke Format Biner: yaitu untuk mengonversi gambar penampung menjadi bentuk biner, yaitu bit-plane, sehingga dapat dianalisis kompleksitasnya dan digunakan untuk menyisipkan data.
4. Menghitung Kompleksitas Bit-*Plane*: yaitu tahap untuk menghitung kompleksitas dari setiap bit-*plane* gambar untuk menentukan bit-*plane* mana yang cukup kompleks dan dapat digunakan untuk penyisipan pesan tanpa terlihat oleh pengamat biasa.
5. Pilih Bit-*Plane* dengan Kompleksitas Tinggi: yaitu hasil perhitungan kompleksitas, bit-*plane* yang memiliki kompleksitas tinggi akan dipilih. Bit-*plane* yang lebih kompleks lebih mampu menyembunyikan data tanpa terdeteksi karena sudah memiliki pola yang tidak beraturan.
6. Penyisipan Data Enkripsi ke Bit-*Plane* Terpilih: yaitu menyisipkan data terenkripsi ke dalam bit-*plane* yang telah dipilih.
7. Rekonstruksi Gambar dengan Pesan yang Tersembunyi: yaitu setelah penyisipan data selesai, gambar dikonstruksi kembali menjadi file gambar yang mengandung pesan terenkripsi.
8. *Output Stego Image*: yaitu hasil akhir dari proses, berupa gambar *stego* yang berisi pesan tersembunyi. Gambar ini terlihat seperti gambar biasa namun telah mengandung data terenkripsi.
9. Selesai: yaitu titik akhir dari *flowchart* yang menandakan bahwa proses sudah selesai.

Flowchart Proses Penyisipan dengan Steganografi BPCS dalam penelitian ini ditampilkan dalam Gambar 3.



Gambar 3. *Flowchart* proses penyisipan dengan steganografi BPCS

 ISSN:2302-8149
e-ISSN:2540-9719

4 Hasil dan Pembahasan

Pada tahap ini akan dijelaskan hasil penelitian yang didapatkan, untuk lebih rincinya sebagai berikut:

4.1 Pengujian Program

Penelitian ini melakukan pengujian menggunakan Google Colab dengan proses enkripsi dan dekripsi data menggunakan algoritma AES, serta penyisipan hasil enkripsi ke dalam file gambar menggunakan teknik steganografi BPCS. File yang diuji mencakup format .txt dan .pdf.

4.2 Proses Enkripsi

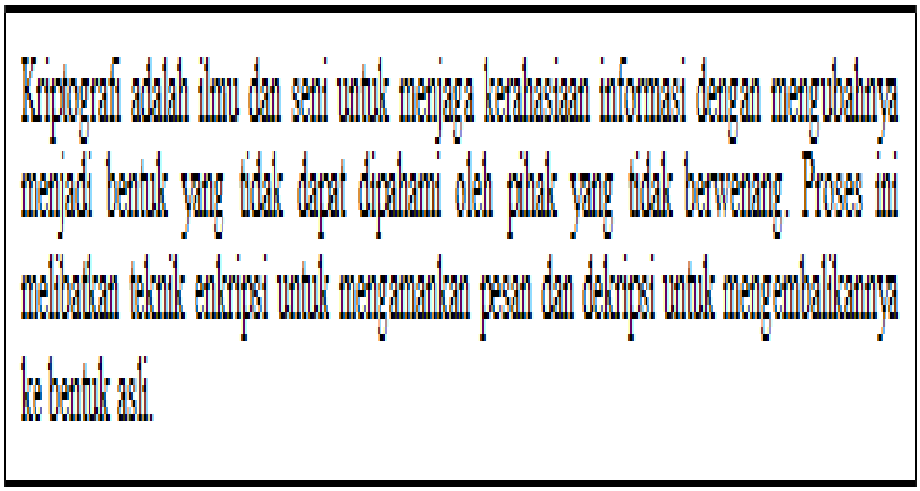
Proses enkripsi dilakukan untuk menjaga keamanan data. Tiga jenis file yang dienkripsi adalah file berformat TXT dan PDF. Hasil enkripsi setiap file menghasilkan *ciphertext* yang berbeda, meskipun menggunakan kunci yang sama. Berikut adalah penjelasan proses enkripsi untuk masing-masing file:

4.2.1 File TXT

Input:

File teks Kriptografi.txt dengan isi sebagai berikut.

Plaintext ditampilkan dalam Gambar 4.

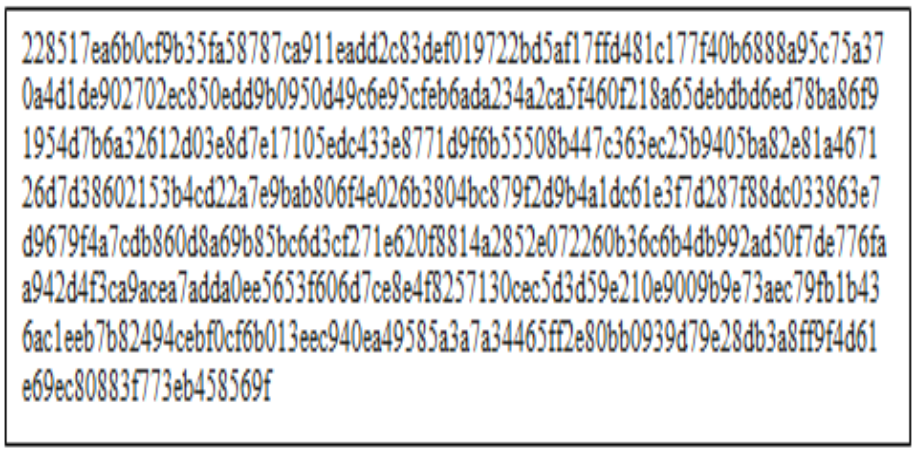


Gambar 4. *Plaintext* file TXT

Key:

AYOSEMANGATAFTAR

Chipertext ditampilkan dalam Gambar 5.



Gambar 5. *Chipertext* file TXT

4.2.2 File PDF

Input:

File teks Surat Lamaran.pdf dengan isi sebagai berikut:

Plaintext ditampilkan dalam Gambar 6.

 ISSN:2302-8149
e-ISSN:2540-9719

Medan, 11 September 2023

Perihal: Lamaran Asisten Praktikum
Lampiran: 1 (satu) berkas
Kepada Yth.
Kepala Laboratorium FST UIN SU
Di Universitas Islam Negeri Sumatera Utara Medan
Assalamualaikum warahmatullahi wabarakatuh,
Dengan hormat, saya Afthar Kautsar (NIM 0701212239), mahasiswa Fakultas Sains dan Teknologi Jurusan Ilmu Komputer, bermaksud mengajukan lamaran sebagai Asisten Laboratorium FST UIN SU. Saya memiliki minat besar untuk berkontribusi dalam pengembangan kemampuan mahasiswa di bidang ilmu komputer. Sebagai bahan pertimbangan, saya lampirkan KHS Semester 3 dan 4 serta pas foto 4x6. Besar harapan saya agar dapat diberikan kesempatan menjadi bagian dari Laboratorium FST UIN SU.
Atas perhatian dan kesempatan yang diberikan, saya ucapkan terima kasih.
Wassalamualaikum warahmatullahi wabarakatuh.
Hormat saya,
Afthar Kautsar

Gambar 6. *Plaintext* file PDF

Key:
AYOSEMANGATAFTAR
Chipertext ditampilkan dalam Gambar 7.

31c961b3a44a2d77d22ae180523c6f032c955e449aca53a5f296c415cd438b5341c50b3f8ea8cee
bb111d2619bbb196822fd383f0cae2ae7bd37aa083ac6387603a78fbdcc085f8c2b128fb8a554e3
ac178f66b322f387e6033aef782613e3020f841b6e855b7ec6559b97430696f275dbc459876f3db
a69c12f232aea0f33b33a491551bbdd4a9281119a12b22acb4aac9dff7a8c01586654b7dbc84d76
aa2a3a0c7a2bcee7f22852aa99a574a89ef51893b0f9c808f635a6921d32f609f35edf0cd3e26dbea
d2d9f7ec78bdd5e30e417f574f974e53eebe1bfdb2755da1ec2d53a16cb2cb0cfe9d5ac464efc4ae
17160e863418d26c62d084cc5be7ab0e2ddd19cd1f8906405db0653bb32e11175f3f31e0088b7b
978b17c83b9f53a7b628554fe28be651c61478ccc8e6503de968ce1ada697e6083cdf921f41053a
377d3df3684d537d2266d2871f54f159900ddf40276aa162e0a988e5c86b0d68b50cd45d90457b
90b606480b10f8ea45a730e412281e01405c7ef5534853df8c16e791e8ee84d3523fffae84434d4e
18980a6add79add1c95b8bd58ccd4fdb8fbcafe04e3d8bbb3da9b9cfe295ba7c2236814d9f1f8d0b
091d715e29409f0f72498a910a477020ec12fe80e7f3f4a7c72ab1446dc7692b9efe95559b0d9f49
244f79cc419e15b3fb35d4c4c8a54e078f4e01e6ec25f08bd6fb4dc8d539abeb0df908c17b215ea6
15f9bae2b599a9b0de5ff82e4b1eb21f0546b92f6fe70c192ef0a872ee77c8590b878715fde7a3ba
649aa2e2b56161a231ca36ae3843f615de84a12bb1ee0bcb6a32f567ca8990b1f5f77fb6d23afb3a
4ede7f7e8ef4a7933410dfefd2c1a8c524586f3d643ad2b4fabcbe64ab8b965c538e1b05dcb0d7f2
119df53affd59f193a3a9faf0aab4408406462db71aead44bdb6a36cc330017882f9211ff48f7c580
5dc588ec731e4fe3742cbcf12ce4a07f3a64c22ca16b5e87f024a2f5b44a221c5b5f115c0b281b9d
e2d7d378ea58aa9ef0d23e5498c5ed254945d4f82f59f7064dba99c2c09fd31e0c5d2978ae807b4
b1a8285a29dc789f935dd6cccacc640b78206c7690af8cbc7811602364825a6bb138d7ac65a1fb3
9c096dd6d7a3b1038aa2d1484ef81667e0c10f4534fb2bfb1e7a19f20dfd7151e237ff9560d9af66
cef778e2c6fc3ae37acbcacb19eeb3ab11df2110d16065cb6d9a1c7649d912c7e5e6208b2953d36
2f59b5830ae3224582be75482dbf5050d80aadf8a8e60a2817f5abc679b270bf39adfac72f63209d
e0173350a653a0375c912fba9f47637f8357139c6ec25f08bd6fb4dc8d539abeb0df908c17b215ea
615f9bae2b599a9b0de5ff82e4b1eb2d26c62d084cc5be7ab0e2ddd19cd1f8906405db0653bb32e
11175f3f31e0088b7b978b17c83b9f53a7b628554fe28be651c61478ccc8e6503de968ce1ada697
e6083cdf921f41053a377d3df3684d537d2266d2871f54f159900ddf40276aa162e0a988e5c86b0

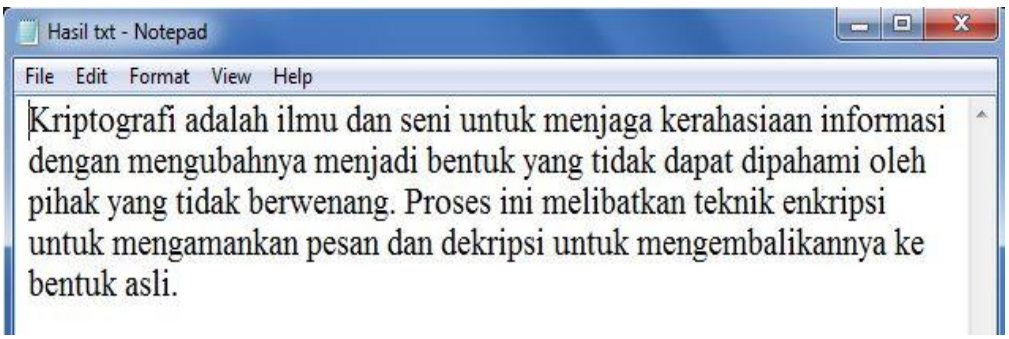
Gambar 7. *Chipertext* file PDF

4.3 Proses Dekripsi

File yang telah dienkripsi dapat didekripsi kembali ke bentuk semula, yang diproses melalui tahapan dekripsi menggunakan Google Colab. Berikut adalah hasil dekripsi untuk masing-masing format file:

4.3.1 File TXT

Hasil dekripsi file TXT menampilkan konten teks asli yang sebelumnya telah dienkripsi. Proses ini memastikan tidak ada perubahan pada isi file ditampilkan dalam Gambar 8.

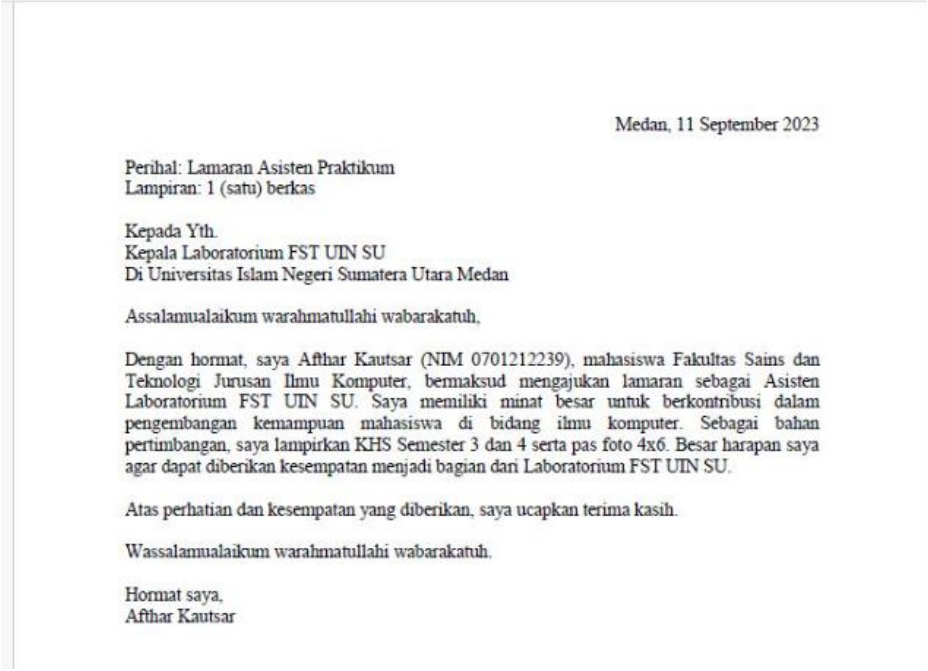


Gambar 8. Hasil dekripsi file TXT

 ISSN:2302-8149
e-ISSN:2540-9719

4.3.2 File PDF

Proses dekripsi untuk file PDF berhasil mengembalikan dokumen ke bentuk aslinya tanpa perubahan pada struktur atau isi dokumen ditampilkan dalam Gambar 9.



Medan, 11 September 2023

Perihal: Lamaran Asisten Praktikum
Lampiran: 1 (satu) berkas

Kepada Yth.
Kepala Laboratorium FST UIN SU
Di Universitas Islam Negeri Sumatera Utara Medan

Assalamualaikum warahmatullahi wabarakatuh,

Dengan hormat, saya Afthar Kautsar (NIM 0701212239), mahasiswa Fakultas Sains dan Teknologi Jurusan Ilmu Komputer, bermaksud mengajukan lamaran sebagai Asisten Laboratorium FST UIN SU. Saya memiliki minat besar untuk berkontribusi dalam pengembangan kemampuan mahasiswa di bidang ilmu komputer. Sebagai bahan pertimbangan, saya lampirkan KHS Semester 3 dan 4 serta pas foto 4x6. Besar harapan saya agar dapat diberikan kesempatan menjadi bagian dari Laboratorium FST UIN SU.

Atas perhatian dan kesempatan yang diberikan, saya ucapkan terima kasih.

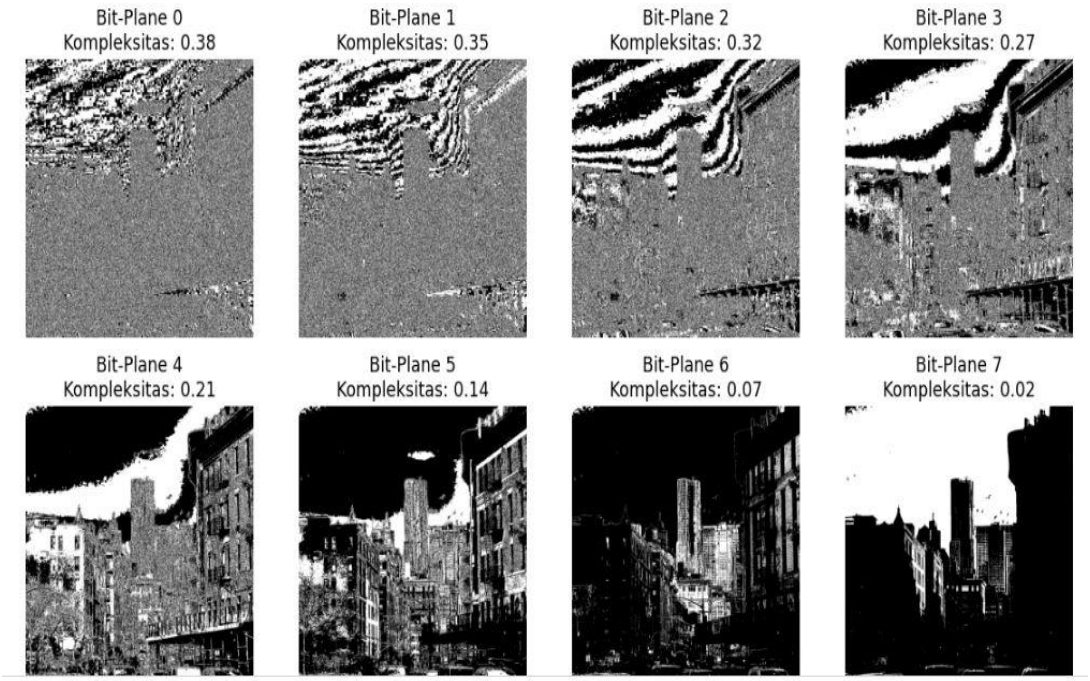
Wassalamualaikum warahmatullahi wabarakatuh.

Hormat saya,
Afthar Kautsar

Gambar 9. Hasil dekripsi file PDF

4.4 Analisis Kompleksitas Bit-*Plane*

Proses pengukuran kompleksitas bit-*plane* menggunakan metode transisi horizontal dan vertikal menghasilkan nilai kompleksitas ditampilkan dalam Gambar 10.



Gambar 10. Tampilan Kompleksitas

Kompleksitas Bit-Plane merujuk pada tingkat kerumitan atau variasi yang ada dalam bit-plane dari sebuah gambar digital. Sebuah gambar digital, yang terdiri dari piksel-piksel dengan intensitas warna tertentu, dapat dipecah menjadi beberapa lapisan bit (bit-plane) berdasarkan urutan bit dalam representasi biner setiap pikselnya. Setiap bit-plane mewakili satu bit dari data biner piksel (misalnya, bit terendah atau bit tertinggi). Kompleksitas Bit-Plane dijelaskan dalam Tabel 1.

 ISSN:2302-8149
e-ISSN:2540-9719

Tabel 1. Kompleksitas bit-*plane*

Bit-Plane	Transisi Horizontal	Transisi Vertikal	Total Kompleksitas
0	441868	447017	0.38
1	410054	412952	0.35
2	371995	367152	0.32
3	320394	304647	0.27
4	253073	231442	0.21
5	168459	156464	0.14
6	82111	77242	0.07
7	24384	17248	0.02

4.5 Proses Penyisipan Pesan

Proses penyisipan pesan dilakukan dengan menyisipkan file ke dalam gambar menggunakan teknik steganografi BPCS. File dengan format yang berbeda (TXT dan PDF) disisipkan ke dalam gambar yang sama. Pemilihan bit-plane dilakukan berdasarkan nilai kompleksitas tertinggi.

4.5.1 Penyisipan File TXT

Penyisipan file TXT berhasil dilakukan dengan panjang pesan biner 2224 bit pada bit-plane 0 yang memiliki kompleksitas 0.38. Berikut adalah hasil ditampilkan dalam Gambar 11.

Gambar sebelum penyisipan:
Ukuran file: 188 KB

Gambar setelah penyisipan:
Ukuran file: 1,35 MB

Gambar 11. Tampilan penyisipan file TXT

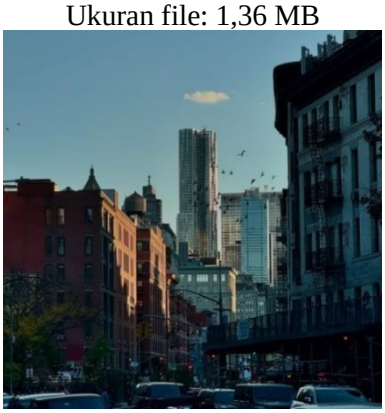
Perbandingan kedua gambar tersebut menunjukkan tidak adanya perubahan signifikan antara gambar sebelum dan sesudah penyisipan pesan.

4.5.2 Penyisipan File PDF

Penyisipan file PDF berhasil dilakukan dengan panjang pesan biner 39104 bit pada bit-plane 0 yang memiliki kompleksitas 0.38. Berikut adalah hasil ditampilkan dalam Gambar 12.

Gambar sebelum penyisipan:
Ukuran file: 188 KB

Gambar setelah penyisipan:
Ukuran file: 1,36 MB



Gambar 12. Tampilan penyisipan file PDF

Perbandingan kedua gambar tersebut menunjukkan tidak adanya perubahan signifikan antara gambar sebelum dan sesudah penyisipan pesan.

 ISSN:2302-8149
e-ISSN:2540-9719

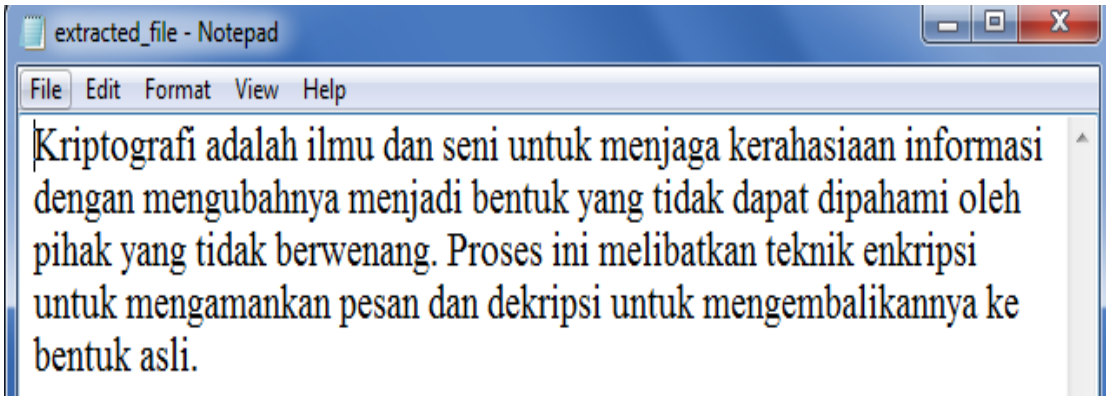
4.6 Ekstraksi Pesan

Ekstraksi pesan dilakukan untuk mengambil kembali file yang sebelumnya telah disisipkan ke dalam gambar menggunakan teknik steganografi BPCS. Proses ini bertujuan memastikan file yang tersembunyi dapat diambil kembali dengan utuh tanpa ada kerusakan atau kehilangan data. Berikut adalah hasil ekstraksi untuk setiap format file yang diuji dalam penelitian ini:

4.6.1 Ekstraksi File TXT

Ekstraksi pesan berhasil dilakukan, dan file TXT yang telah disisipkan ke dalam gambar berhasil dikembalikan tanpa perubahan.

a. Nama file: *extracted_file.txt*
b. Ukuran file: 0.27 KB
c. Isi file ditampilkan dalam Gambar 13.

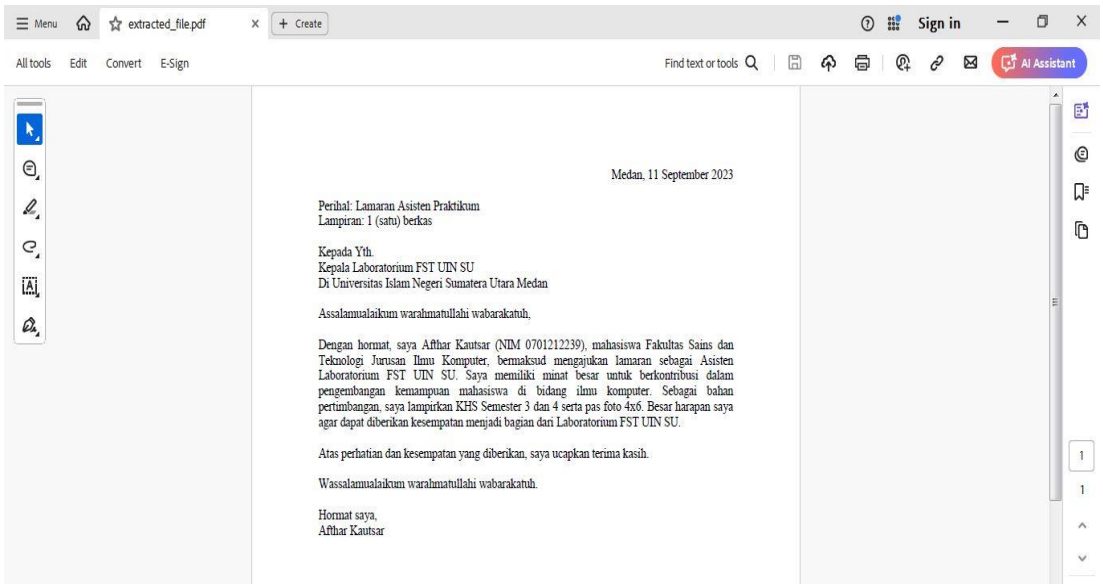


Gambar 13. Hasil penyisipan file TXT

4.6.2 Ekstraksi File PDF

Proses ekstraksi file PDF berhasil dilakukan dengan mengembalikan isi dokumen secara lengkap.

a. Nama file: *extracted_file.pdf*
b. Ukuran file: 4.77 KB
c. Isi file ditampilkan dalam Gambar 14.



Gambar 14. Hasil penyisipan file PDF

Pengujian yang dilakukan menggunakan Google Colab membuktikan bahwa platform ini dapat digunakan secara efisien dalam mengimplementasikan algoritma enkripsi dan steganografi. Eksekusi program berjalan lancar dan memungkinkan pemrosesan yang lebih kompleks dengan dukungan komputasi berbasis cloud. Dengan demikian, penelitian ini berhasil membuktikan bahwa kombinasi AES dan BPCS dapat digunakan sebagai metode yang efektif dalam meningkatkan keamanan file

 ISSN:2302-8149
e-ISSN:2540-9719

teks, terutama dalam konteks perlindungan informasi digital. Metode ini tidak hanya memastikan keamanan data melalui enkripsi yang kuat, tetapi juga memungkinkan penyimpanan tersembunyi dalam gambar tanpa terdeteksi oleh teknik steganalisis konvensional.

5 Kesimpulan

Berdasarkan hasil penelitian dan pembahasan, maka dapat diambil kesimpulan bahwa penelitian ini berhasil mengimplementasikan algoritma AES dan teknik steganografi BPCS dalam mengamankan file teks berformat .txt dan .pdf menggunakan Google Colab. Hasil enkripsi dengan AES memastikan keamanan data, sementara penyisipan ciphertext ke dalam citra menggunakan BPCS memungkinkan penyembunyian tanpa perubahan visual yang signifikan. Kompleksitas bit-plane menjadi faktor utama dalam menentukan lokasi penyisipan, meningkatkan ketahanan terhadap analisis steganografi. Proses ekstraksi berhasil mengembalikan file tanpa kehilangan data, membuktikan efektivitas metode ini dalam menjaga kerahasiaan informasi.

Referensi

[1] V. B. S. Gajare, S. Charles, S. Wagh, "*A Tactic for Encrypting Stego-Image using AES and BPCS Algorithm*," *Int. J. Comput. Sci. Mob. Comput.*, vol. 4, no. 3, pp. 396–405, 2015.

[2] A. F. Mahmood, N. A. Kanai, and S. S. Mohmmad, "*BPCS Steganography for Data Security using FPGA Implementation*," *Int. Conf. Commun. Signal Process.*, vol. 19, no. 4, pp. 14–23, 2012.

[3] M. A. Y. A. Tauhid, Maisha Tasnim, Saima Arifin Noor, Nuruzzaman Faruqui, "*A Secure Image Steganography using Advanced Encryption Standard and Discrete Cosine Transform*," *J. Inf. Secur.*, vol. 10, no. 3, 2019.

[4] M. Salim, "Penerapan Enkripsi untuk Keamanan Data di Era Digital," *J. Teknol. Informasi*, vol. 4, no. 2, pp. 45–59, 2022.

[5] D. Solichin, A., & Wulandari, "*Bit-Plane Complexity Segmentation (BPCS)* untuk Steganografi," *J. Teknol. Inf.*, vol. 4, no. 2, pp. 99–111, 2015.

[6] B. Olivia *et al.*, "Implementasi Kriptografi pada Keamanan Data menggunakan *Algoritma Advance Encryption Standard (AES) Cryptographic Implementation in Data Security using Advanced Encryption Standard (AES) Algorithm*," *J. Simantec*, vol. 11, no. 2, pp. 167–174, 2023.

[7] A. Widyanarko, "Implementasi Steganografi dengan Metode *Bit-Plane Complexity Segmentation (BPCS)* untuk Dokumen Citra Terkompresi," *Transform*, pp. 1–6, 2013.

[8] A. R. Hakim and W. M. Baihaqi, "Implementasi Aplikasi Steganografi berbasis *Web* menggunakan Algoritma LSB dan BPCS," vol. 12, no. 2, pp. 50–58, 2023.

[9] R. A. Zay and M. Mesran, "Analisa Metode MGR untuk mendeteksi Keaslian Citra Digital," *TIN Terap. Inform. Nusant.*, vol. 2, no. 8, pp. 492–500, 2022, doi: 10.47065/tin.v2i8.1029.

[10] M. Cisco, and P. Tracer, "Implementasi Algoritma *Advanced Encryption Standard (AES)* untuk Pengamanan Citra Digital," vol. 8, no. 1, pp. 1–10, 2024.

[11] S. Anwar, "Implementasi Pengamanan Data dan Informasi dengan Metode Steganografi LSB dan Algoritma Kriptografi AES," *J. Format*, vol. 6, no. 1, pp. 65–74, 2017.

[12] Y. Fatma, H. Mukhtar, and M. Taufik, "Implementasi Steganografi pada Teks Terenkripsi dengan Algoritma RSA menggunakan Metode BPCS," *J. Fasilkom*, vol. 7, no. 2, pp. 260–265, 2018, doi: 10.37859/jf.v7i2.783.

[13] I. M. W. Gede Wisnu Bhaudhayana1, "Implementasi Algoritma Kriptografi AES 256 dan Metode Steganografi LSB pada Gambar Bitmap," *J. Ilm. Ilmu Komput. Univrsitas Undayana*, vol. 8, no. 2, 2015.

[14] I. S. Rangkuti and E. R. Siagian, "Implementasi Penyembunyian Pesan pada Audio dengan Metode *Bit-Plane Complexity Segmentation (BPCS)*," *Jurikom (Jurnal Ris. Komputer)*, vol. 7, no. 2, p. 285, 2020, doi: 10.30865/jurikom.v7i2.2088.

[15] S. Saripa, "Implementasi Sistem Keamanan File menggunakan Algoritma AES untuk mengamankan File Pribadi: Implementasi Sistem Keamanan File menggunakan Algoritma AES untuk mengamankan File Pribadi," *Progress. Information, Secur. Comput. Embed. Syst.*, vol. 1, no. 2, pp. 138–148, 2023.