Implementasi Algoritma Ecc Dan Paillier Pada Steganografi Audio Dengan Menggunakan Metode Least Significant Bit (LSB)

Agdelssa Itaqullah Putra Nalramus
Telkom University Purwokerto
Purwokerto, Jawa Tengah
agdelssa@student.telkomuniversity.ac.id

Wahyu Adi Prabowo
Telkom University Purwokerto
Purwokerto, Jawa Tengah
wahyup@telkomuniversity.ac.id

Trihastuti Yuniati
Telkom University Purwokerto
Purwokerto, Jawa Tengah
trihastutiy@telkomuniversity.ac.id

***Abstrak* — Era digital mentransformasi cara manusia berkomunikasi, berbisnis, dan menyimpan informasi. Pertukaran data digital kian masif, membawa kemudahan sekaligus memicu kecemasan terkait keamanan data. Kebocoran data dan penyalahgunaan informasi menjadi ancaman nyata yang dapat mengakibatkan kerugian finansial, reputasi, bahkan pelanggaran privasi. Internet telah menjadi saluran komunikasi yang paling disukai saat ini, di mana hampir semua dokumen seperti teks, gambar, audio, atau video, ditransmisikan melalui internet. Hal tersebut menunjukkan bahwa keamanan dalam mentransfer data melalui internet menjadi semakin penting, terutama dalam mengamankan informasi rahasia dari pihak yang tidak berwenang. Terdapat beberapa metode untuk mengamankan data pesan rahasia, seperti Teknik steganografi dan Teknik Kriptografi. Pada Teknik kriptografi terdapat banyak algoritma yang tersedia, pada penelitian ini penulis menggunakan double algoritma kriptografi, yaitu *Elliptic Curve Cryptography* (ECC) dan Paillier. Sedangkan untuk Teknik steganografi penulis menggunakan metode *Least Significant Bit* (LSB) untuk teknik penyembunyian pesan kedalam suatu media, media yang digunakan pada penelitian ini yaitu audio. Untuk mengecek keaslian data juga menggunakan Message Digest 5 (MD5).**

***Kata kunci*— Keamanan Data, *Elliptic Curve Cryptography* (ECC), Paillier, Least Significant Bit (LSB), Audio**

I. PENDAHULUAN

Era digital mentransformasi cara manusia berkomunikasi, berbisnis, dan menyimpan informasi. Pertukaran data digital kian masif, membawa kemudahan sekaligus memicu kecemasan terkait keamanan data. Kebocoran data dan penyalahgunaan informasi menjadi ancaman nyata yang dapat mengakibatkan kerugian finansial, reputasi, bahkan pelanggaran privasi. Kriptografi, metode keamanan data tradisional, telah terbukti efektif dalam melindungi data. Namun, kriptografi dapat menarik perhatian pihak yang tidak berwenang, meningkatkan risiko investigasi dan penyadapan. Steganografi, sebuah teknik menyembunyikan data rahasia di dalam media lain (carrier), menawarkan solusi keamanan yang lebih canggih. Data rahasia tersembunyi tidak terdeteksi, sehingga meningkatkan keamanan dan privasi informasi. Internet telah menjadi saluran komunikasi yang paling disukai saat ini, di mana hampir semua dokumen seperti teks, gambar, audio, atau video, ditransmisikan melalui internet. Hal ini menunjukkan bahwa keamanan dalam mentransfer data melalui internet menjadi semakin penting, terutama dalam mengamankan informasi rahasia dari pihak yang tidak berwenang. Steganografi audio berbasis LSB (Least Significant Bit) merupakan salah satu teknik yang digunakan untuk menyembunyikan data rahasia dalam file audio digital. Teknik ini memanfaatkan bit paling tidak signifikan dalam sampel audio untuk menyisipkan informasi rahasia tanpa mengubah secara signifikan kualitas audio yang terdengar. Dengan menggunakan teknik ini, informasi sensitif dapat disembunyikan secara efektif dalam file audio tanpa diketahui oleh pihak yang tidak berhak. Penyisipan LSB memanfaatkan bit paling tidak signifikan dari data host untuk menyembunyikan informasi tambahan, sehingga perubahan yang dihasilkan pada data host tidak terlalu mencolok secara visual. Dalam konteks keamanan informasi, steganografi dapat digunakan dalam berbagai aplikasi seperti komunikasi rahasia, pertukaran data sensitif, dan pengamanan informasi penting. Dengan menggunakan teknik steganografi, pengguna dapat meningkatkan keamanan data mereka dengan cara yang tidak terlihat oleh pihak yang tidak berwenang[1]. Kriptografi adalah sebuah teknik yang digunakan untuk mengamankan pesan-pesan yang dikirim melalui platform berbasis ponsel pintar. Kriptografi melibatkan proses enkripsi dan dekripsi pesan, di mana pesan asli diubah menjadi bentuk yang tidak dapat dibaca (ciphertext) sebelum dikirim, dan kemudian diubah kembali menjadi pesan asli saat diterima. Algoritma kriptografi dibagi menjadi 2 yaitu Simetris dan Asimetris. Algoritma simetris merupakan suatu algoritma yang menggunakan single key untuk melakukan enkripsi maupun dekripsi pesan atau sering disebut single key. Algoritma ini banyak digunakan untuk enkripsi dan dekripsi pesan karena kelebihannya yaitu simple dan cepat. Disisi lain kekurangan yang dimiliki algoritma ini yaitu jika key yang digunakan untuk enkripsi dan dekripsi pesan diketahui oleh pihak selain pengirim, penerima maka segala informasi didalamnya akan diketahui juga. Contoh dari algoritma ini yaitu AES, RC4, Blowfish, Rijndael.

Algoritma asimetris merupakan suatu algoritma kriptografi yang sering disebut public key. Algoritma ini memiliki 2 kunci yaitu public key dan private key sehingga berbeda untuk kunci untuk enkripsi dan dekripsi pesan. Kunci publik dapat dilihat untuk umum karena untuk pengirimannya tidak perlu pada saluran dengan keamanan tinggi sedangkan kunci privat dimiliki oleh masing-masing pengirim dan penerima. Kelebihan daripada algoritma asimetris ini yaitu jumlah kunci dapat ditekan untuk masing-masing penerima karena tidak perlu membuat kunci sebanyak algoritma simetris yang berbeda untuk masing-masing penerima. Contoh dari Algoritma ini yaitu ECC (Elliptic Curve Cryptography), Paillier. Penggabungan steganografi dan kriptografi secara bersamaan dapat meningkatkan pengamanan data. Metode penggabungan steganografi dan kriptografi banyak dikembangkan. Pada umumnya teknik yang digunakan yaitu dengan mengenkripsi pesan terlebih dahulu (kriptografi), kemudian menyisipkannya ke media cover (steganografi)[2]. Nilai hash adalah suatu kode alfanumerik yang dihasilkan dari suatu data tertentu. Hash digunakan untuk menghasilkan representasi digital kecil dari data yang lebih besar. Proses hash ini mengubah data menjadi kombinasi angka, huruf, atau karakter lain yang terenkripsi. MD5 merupakan singkatan dari Message Digest Algorithm 5[3]. Dari uraian diatas, penulis akan menerapkan algoritma kriptografi asimetris yaitu Algoritma ECC dan Paillier untuk enkripsi dan dekripsi pesan sehingga dapat mengetahui algoritma mana yang lebih efektif digunakan dalam upaya pengamanan pesan dalam sisipan file pada format file audio AIFF dan WAV dengan menggunakan metode LSB (Least Significant Bit), kemudian digabungkan dengan MD5 hash untuk memastikan data asli.

II. KAJIAN TEORI

A. Keamanan Data

Keamanan Data merupakan Jaringan entitas dalam berkomunikasi dapat dibuat dengan banyak cara, termasuk routing, kebijakan kontrol akses (mungkin melibatkan pelabelan), dll. Adapun meliputi bagaimana sistem diberi label, yang mampu mengekspresikan banyak jenis persyaratan keamanan, dapat dibangun untuk menetapkan entitas ke posisi yang sesuai dalam pesanan parsial jaringan. Paradigma mapan dalam keamanan data, seperti konflik, konglomerasi, agregasi, diperkenalkan dalam contoh. Ada algoritma yang efisien untuk mengimplementasikan konsep-konsep ini, mereka adalah aplikasi dari algoritma penutupan transitif dan algoritma komponen yang terhubung kuat (Logrippo, 2021)[4].

B. Steganografi

Asal kata steganografi dari bahasa Yunani *stegan* yang bermakna menyembunyikan dan *graphos* yang bermakna tulisan. Steganografi merupakan seni menyembunyikan informasi atau pesan tersembunyi (*embedded message*) pada suatu wadah penampung (*cover object*) yang dapat berupa teks, gambar, audio, video dan lain-lain. Steganografi merupakan teknik yang bertujuan menyembunyikan pesan rahasia atau tulisan rahasia sehingga informasi rahasia tersebut tidak dapat diidentifikasi oleh orang lain (pihak ketiga) dalam artian yang dapat mengetahui pesan tersebut hanya pengirim dan penerima. Tujuan awal dari teknik steganografi ini bukan untuk mengamankan pesan, yaitu agar orang lain tidak mampu mendeteksi bahwa pada suatu pesan atau informasi tersebut terdapat pesan rahasia didalamnya. Teknik steganografi ini biasanya dibuat dan diimplementasikan melalui media digital[5].

C. Kriptografi

Kriptografi (cryptography) berasal dari Bahasa Yunani: “cryptos” artinya “secret” (rahasia), sedangkan “graphein” artinya “writing” (tulisan). Jadi kriptografi adalah ilmu dan seni untuk menjaga kerahasiaan pesan dengan cara menyandikannya ke dalam bentuk yang tidak dapat dimengerti lagi maknanya. Dalam menjaga kerahasiaan data dengan kriptografi, data sederhana yang dikirim (plaintext) diubah ke dalam bentuk data sandi (ciphertext), kemudian data sandi tersebut hanya dapat dikembalikan kebentuk data sebenarnya hanya dengan menggunakan kunci (key) tertentu yang dimiliki oleh pihak yang sah saja. Tentunya hal ini menyebabkan pihak lain yang tidak memiliki kunci tersebut tidak akan dapat membaca data yang sebenarnya sehingga dengan kata lain data akan tetap terjaga kerahasiannya[6].

D. Algoritma ECC

Algoritma ECC memiliki kepanjangan dari *Elliptic Curve Cryptography* ini ditemukan pada tahun 1985 oleh Victor Miller dan Neil Koblitz sebagai mekanisme alternatif untuk implementasi kriptografi asimetris. Algoritma ECC dibuat berdasarkan logaritma diskrit yang lebih menantang pengerjaannya pada kunci yang memiliki panjang yang sama. Protocol yang dipakai adalah Suite B, yang terdiri dari Elliptic Curve Diffie Hellman (ECDH), Elliptic Curve Menezes-Qu-Vanstone (ECMQV) untuk pertukaran dan persetujuan kunci ; Elliptic Curve Digital Signature Algorithm (ECDSA) untuk digital signatures; the Advanced Encryption Standard (AES) untuk *symmetric encryption*; and the Secure Hashing Algorithm (SHA). Kelebihan dari ECC adalah sangat penting untuk NSA karena keamanan yang digunakan untuk mengamankan perangkat keras[7].

E. Algoritma Paillier

Algoritma *paillier* adalah sebuah sistem yang berbasis algoritma asimetris probalistik. Algoritma enkripsi yang digunakan adalah sebuah algoritma kriptografi kunci public. Sistem ini ditemuka oleh pascal paillier pada tahun 1999. Sistem algoritma paillier dibuat berdasarkan pemikiran bahwa untuk menghitung kelas residu yang ke-n, hal ini dikenal sebagai asumsi composite residuosity (CR). Paillier Cryptosystem Algorithm merupakan jenis kriptografi berbasis keypair, maksudnya setiap pengguna mendapatkan kunci publik dan pribadi, dan pesan yang dienkripsi dengan kunci publik mereka hanya dapat didekripsi dengan kunci pribadi mereka. Paillier Cryptosystem Algorithm tidak banyak digunakan sebagai algoritma lain seperti RSA, dan ada beberapa implementasi yang tersedia secara online. Kelebihan dari paillier cryptosystem algorithm tidak seperti banyak cryptosystem lainnya keypair, paillier cryptosystem algorithm menyediakan homomorfisme aditif. Ini berarti bahwa pesan dapat ditambahkan bersama ketika dienkripsi, dan pihak lain tidak akan mendekripsi dengan benar[8].

F. Audio AIFF

Format AIFF dengan seri Apple Macintosh yang berbasis pada Motorola 68K (dan, kemudian, PowerPC), dan format WAVE dengan IBM PC dan seri prosesor Intel. Spesifikasi AIFF secara eksplisit membahas kemungkinan aliran multi-

saluran, mengutip contoh yang menggunakan tiga, empat, dan enam saluran. Masalahnya di sini adalah ambiguitas, karena dua contoh empat saluran alternatif diberikan, tanpa informasi yang jelas. Ambiguitas menandakan tabrakan antara dua atau lebih makna yang ada, sedangkan ketidaklengkapan menandakan tidak adanya makna apa pun. Penghapusan ambiguitas adalah salah satu masalah utama dalam mendesain format file, jika data ingin independen dari intervensi pengguna untuk rendering atau pemrosesan [10].

G. Audio WAV

WAV adalah singkatan dari istilah dalam bahasa Inggris waveform audio format merupakan standar format berkas audio yang dikembangkan oleh Microsoft dan IBM. WAV merupakan varian dari format bitstream RIFF dan mirip dengan format IFF dan AIFF yang digunakan komputer Amiga dan Macintosh. WAV maupun AIFF kompatibel dengan sistem operasi Windows dan Macintosh. Walaupun WAV dapat menampung audio dalam bentuk terkompresi, umumnya format WAV merupakan audio yang tidak terkompres[11].

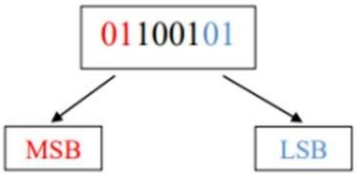
H. Message Digest 5 (MD5)

MD5 adalah salah satu dari sekian banyak metode hash untuk enkripis, dan merupakan proyek lanjutan dari MD4, Professor Ronald Rivest adalah orang yang mengembangkan metode MD5 pada tahun 1991. MD5 ini merupakan versi pembaruan dari MD4 dikarenakan algoritma MD4 dirasa sudah mudah untuk ditebak kuncinya dalam kata lain sudah mendapat pola pasti dari MD4 yang menyebabkan adanya serangan yang melehmahkan algoritma MD4. Proses kerja MD5 adalah memproses informasi asli pada satuan blok-blok input sebesar 512 bit yang diproses secara berulang-ulang. Secara garis besar proses MD5 dijabarkan sebagai berikut :

a. Menambahkan bit pengganjal (padding bits).
b. Menambahkan nilai Panjang informasi semula.
c. Inisialisasi penyangga (buffer) MD.
d. Pengolahan pesan dalam blok berukuran 512[12].

I. Least Significant Bit (LSB)

Least Significant Bit (LSB) merupakan teknik steganografi yang sering digunakan.



Gambar 1. Metode LSB

Angka 0 yang berada di depan disebut *Most Significant Bit* (MSB) maka bit LSB pada biner tersebut yaitu angka 1 yang paling kanan atau paling belakang. Ketika bit paling akhir LSB disisipi atau diubah dengan 0 hal tersebut tidak akan mempengaruhi tampilan warna secara jelas (tidak terlihat jelas perbedaannya). Namun jika bit yang disisipi dengan bit yang berbeda maka akan terlihat perbedaan pada citra. *Least Significant Bit* (LSB) merupakan metode steganografi yang populer dan sering digunakan. Pada metode ini pesan akan disisipkan dengan cara mengganti bit terkecil (terakhir) dari pixel citra dengan bit pesan, karena tidak akan memberikan pengaruh atau perubahan yang signifikan terhadap citra digital[13].

J. Mean Square Error (MSE)

Mean Square Error (MSE) merupakan tingkat keakuratan suatu model prediksi, nilainya merepresentasikan rata-rata kesalahan antara hasil prediksi dengan nilai sebenarnya dengan memperhitungkan kuadrat bias. Sehingga MSE berguna dalam membandingkan audio asli dan audio hasil penyisipan dengan memeriksa selisih nilai keduanya yang dapat dirumuskan pada persamaan sebagai berikut :

$$\text{MSE} = \frac{1}{N}\sum_{i}^{N}(x_i - y_i)^2 \quad (1)$$

dimana :

x = original audio signal
y = stego audio signal
N = jumlah signal sample

MSE mewakilkan perbedaan antara audio asli dengan audio hasil stego-audio. Semakin mirip kedua audio, maka semakin kecil pula nilai MSE[14].

K. Peak Signal to Noise Ratio (PNSR)

Peak Signal to Noise Ratio merupakan nilai perbandingan antara nilai maksimum pada audio hasil pengolahan dengan kuantitas gangguan atau disebut juga noise, yang dinyatakan dalam satuan desibel (dB). Setelah nilai MSE didapatkan, PSNR dapat dihitung dengan persamaan :

$$\text{PSNR} = 10log_{10}(\frac{R^2}{MSE}) \quad (2)$$

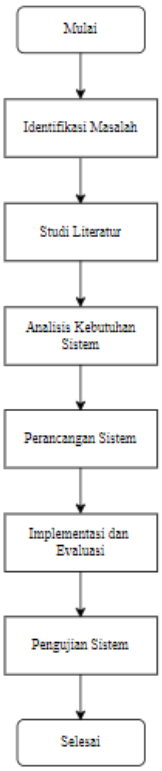
dimana :

R = nilai puncak signal

PSNR adalah perbandingan dari nilai puncak signal audio dengan nilai MSE. Ketika kedua file yang sama dibandingkan, akan menghasilkan nilai MSE=0, sehingga nilai PSNR menunjukkan nilai tak hingga. Untuk itu, semakin kecil nilai MSE semakin tinggi PSNR dari audio tersebut.Semakin tinggi nilai PSNR, semakin kecil noisedari audio tersebut, sehingga kualitas audio semakin baik[15]

III. METODE

Pada penelitian ini, metode yang digunakan untuk pengamanan data teks melibatkan dua teknik utama yaitu Kriptografi ECC (*Elliptic Curve Cryptography*) dan Paillier dengan Steganografi LSB (*Least Significant Bit*). Berikut adalah langkah-langkah yang diterapkan dalam penelitian ini.



Gambar 2 Diagram Alir Penelitian

Pada tahap ini peneliti mengidentifikasi permasalahan yang muncul sehingga dapat memberikan solusi yang bermanfaat untuk menyelesaikan permasalahan tersebut. Merancang latar belakang perlunya pemecahan suatu masalah, merumuskan masalah, mencari tujuan pemecahan masalah, dan mencapai manfaat berdasarkan tujuan penelitian yang dicapai.

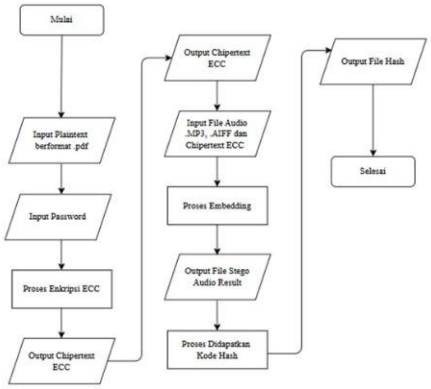
A. Studi Literatur

Langkah selanjutnya adalah melakukan studi literatur. Pada tahap ini peneliti melakukan pengumpulan bahan referensi tentang pengamanan data, dari buku-buku, jurnal dan penelitian tugas akhir sebelumnya yang dapat membantu peneliti untuk memecahkan masalah.

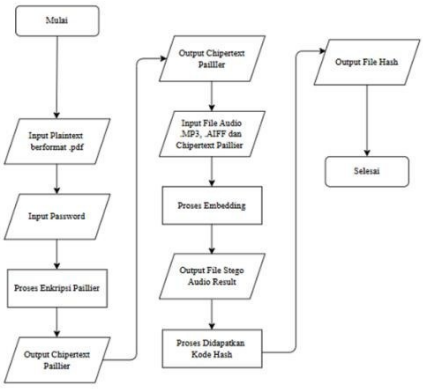
B. Analisa Kebutuhan Sistem

Pada Analisis kebutuhan pada sistem yang akan dirancang penulis yaitu analisis kebutuhan masukan (input), analisis kebutuhan proses dan analisis kebutuhan hasil (output).

C. Peracangan Sistem

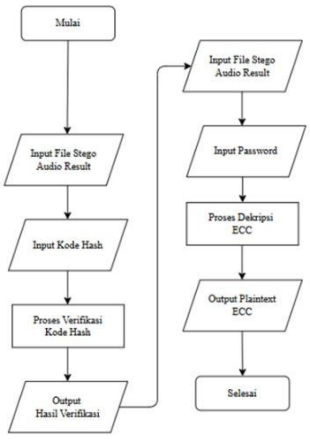


Gambar 4 Tahap Enkripsi pada sistem dengan algoritma ECC

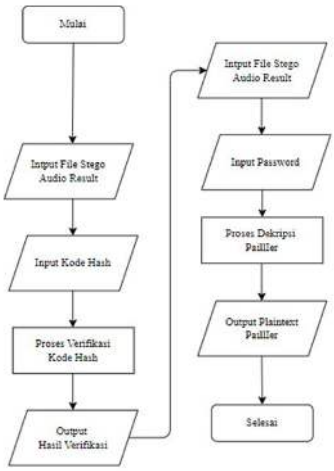


Gambar 3 Tahap Enkripsi pada sistem dengan algoritma Paillier

Pada gambar 3 dan 4 menjelaskan tahapan alur proses enkripsi dengan menggunakan algoritma yang berbeda yaitu ECC (gambar3) dan Paillier (gambar 4). Secara garis besar, proses yang akan dilakukan sistem pada tahap enkripsi yaitu menginputkan plainteks, input kunci dimana pada tahap ini menggunakan kunci publik, proses enkripsi masing-masing algoritma, output dari hasil proses enkripsi (*chiperteks*), input file media penampung atau *stego object* berupa file audio berformat (.AIFF dan .WAV), proses embedding atau penyisipan file, lalu dengan output file audio berformat (.AIFF dan .WAV) yang sudah disisipi psan rahasia atau *chiperteks*.



Gambar 5 Tahap Dekripsi pada sistem dengan algoritma ECC



Gambar 6 Tahap Dekripsi pada sistem dengan algoritma Paillier

Pada gambar 5 dan 6 menjelaskan tahapan alur proses dekripsi dengan menggunakan algoritma yang berbeda yaitu ECC (gambar 5) dan Paillier (gambar 6). Proses yang akan dijalankan oleh sistem pada tahap ini yaitu proses ekstraksi file stego audio (file yang telah disisipi pesan rahasia) lalu kemudian menggunakan kunci yaitu *private key* untuk melanjutkan proses dekripsi dengan masing-masing algoritma. Kemudian setelah dilakukan proses dekripsi berhasil maka akan menghasilkan plainteks atau pesan asli dengan format awal (.PDF).

D. Implementasi dan Evaluasi

Pada tahap implementasi melakukan langkah-langkah dari tahap sebelumnya yaitu perancangan sistem. Jika terdapat kesalahan atau kekurangan sistem dapat dilakukan perbaikan dengan melakukan evaluasi sistem.

E. Pengujian

Pada tahap pengujian sistem dilakukan untuk membuktikan jika aplikasi yang dibangun sudah memenuhi kebutuhan. Proses steganografi dan kriptografi menunjukkan pengujian ini. Tujuan dari kedua prosedur adalah untuk meningkatkan keamanan proses pengiriman pesan. Proses pertama melibatkan pengujian kriptografi menggunakan dua algoritma yang berbeda untuk mengenkripsi pesan dan menghasilkan pesan rahasia atau chiperteks. Proses kedua melibatkan pengujian steganografi menggunakan metode LSB (*Least Significant Bit*) untuk memasukkan chiperteks yang dihasilkan dari proses enkripsi ke dalam file stego audio (.AIFF dan .WAV). Setelah itu, pengujian kembali akan dilakukan, yaitu ekstraksi algoritma kriptografi menggunakan dua algoritma ECC dan Paillier, untuk melihat apakah hasil dekripsi akan sesuai dengan pesan asli atau plainteksnya. Kemudian menguji integritas data dengan hash MD5 untuk memastikan bahwa data atau pesan rahasia yang diuji tetap akurat dan asli.

IV. HASIL DAN PEMBAHASAN

Pada penelitian ini, pengujian terbagi menjadi tahapan yaitu : enkripsi, embed, ekstrak, dekripsi, hash dan pengujian perhitungan MSE dan PNSR. Berikut pembahasannya :

A. Pengujian Enkripsi ECC

Tabel 1 Perbandingan Enkripsi ECC

Analisis Perbandingan Enkripsi ECC				
No	Nama File PDF	Besaran File Sebelum dienkripsi (KB)	Besaran File Sesudah dienkripsi (KB)	Lama Waktu Mengenkripsi (detik)
1	SD01	7	7	5,73
2	SD02	41	41	15,35
3	SD03	542	542	5,76
4	SD04	1.724	1.724	7,47
5	SD05	2.173	2.173	9,31
6	SD06	3.699	3.699	9,55

Untuk hasil pengujian terdapat pada tabel 1, yang dimana ukuran file sebelum dan sesudah proses enkripsi tetap sama. Hal ini menunjukkan bahwa metode ECC yang digunakan dalam penelitian ini tidak menambah atau mengurangi ukuran file, sehingga dapat disimpulkan jika algoritma ECC bekerja dengan efisien dalam hal manajemen ukuran file.

B. Pengujian Enkripsi Paillier

Tabel 2 Perbandingan Enkripsi Paillier

Analisis Perbandingan Enkripsi Paillier				
No.	Nama File PDF	Besaran File Sebelum dienkripsi (KB)	Besaran File Sesudah dienkripsi (KB)	Lama Waktu Mengenkripsi (detik)
1	SD01	7	160	64,61
2	SD02	41	1.001	280,33
3	SD03	542	13.365	3610,77
4	SD04	1.724	42.571	10235,15
5	SD05	2.173	53.655	14871,43
6	SD06	3.699	91.350	21823,91

Untuk hasil pengujian terdapat pada tabel 2, yang dimana ukuran file mengalami perubahan signifikan setelah proses enkripsi. Hal ini menunjukkan bahwa metode Paillier yang digunakan dalam penelitian ini menyebabkan peningkatan ukuran file secara drastis. Sebagai contoh, file SD01 yang awalnya hanya berukuran 7 KB menjadi 160 KB setelah dienkripsi, sedangkan file SD06 mengalami pertambahan dari 3.699 KB menjadi 91.350 KB. Semua besaran file hasil enkripsi hampir naik 24,69 kali lipat dari besaran file awal.

C. Pengujian Embed Chipertext ECC Steganografi WAV

Tabel 3 Perbandingan Embed Chipertext ECC Steganografi WAV

Analisis Perbandingan Chipertext ECC Steganografi WAV (embed)						
No.	Nama File PDF	Besaran File PDF Sebelum diembedd (KB)	Nama File Audio WAV	Besaran File WAV Sebelum diembedd (KB)	Besaran File WAV Sesudah diembedd (KB)	Lama Waktu Mengembed (detik)
1	SD01	7	SA01	590	3.250	10,10
			SA02	1.616	8.906	9,02
			SA03	3.398	16.306	11,32
			SA04	4.995	23.972	10,97
			SA05	7.548	36.230	11,12
			SA06	10.741	51.555	13,28
2	SD02	41	SA01	590	3.250	16,26
			SA02	1.616	8.906	11,24
			SA03	3.398	16.306	12,53
			SA04	4.995	23.972	12,26
			SA05	7.548	36.230	14,31
			SA06	10.741	51.555	15,94
3	SD03	542	SA01	590	Gagal	Gagal
			SA02	1.616	Gagal	Gagal
			SA03	3.398	Gagal	Gagal
			SA04	4.995	23.972	53,16
			SA05	7.548	36.230	52,33
			SA06	10.741	51.555	53,82

Untuk hasil pengujian terdapat pada Tabel 3, yang dimana proses embedding ciphertext hasil enkripsi ECC ke dalam file audio WAV menunjukkan hasil yang bervariasi. Ukuran file ciphertext hasil enkripsi ECC yang diambil dari file PDF berbeda-beda, mulai dari 7 KB hingga 3.699 KB. Proses embedding dilakukan ke berbagai file audio WAV dengan ukuran yang juga bervariasi.

Hal ini menunjukkan bahwa keberhasilan embedding tidak hanya bergantung pada ukuran file audio, tetapi juga pada struktur data ciphertext dan batas kapasitas efektif metode steganografi yang digunakan.

D. Pengujian Embed Chipertext Paillier Steganografi WAV

Tabel 4 Perbandingan Embed Chipertext Paillier Steganografi WAV

Analisis Perbandingan Chipertext Paillier Steganografi WAV (embed)						
No.	Nama File PDF	Besaran File PDF Sebelum diembed	Nama File Audio WAV	Besaran File WAV Sebelum diembed	Besaran File WAV Sesudah diembed	Lama Waktu Mengembed (detik)

		d (KB)		d (KB)	d (KB)	
1	SD01	160	SA01	590	Gagal	Gagal
			SA02	1.616	8.906	19,92
			SA03	3.398	16.306	21,32
			SA04	4.995	23.972	21,96
			SA05	7.548	36.230	25,06
			SA06	10.741	51.555	22,42
2	SD02	1.001	SA01	590	Gagal	Gagal
			SA02	1.616	Gagal	Gagal
			SA03	3.398	Gagal	Gagal
			SA04	4.995	Gagal	Gagal
			SA05	7.548	Gagal	Gagal
			SA06	10.741	51.555	104,54

Untuk hasil pengujian terdapat pada Tabel 4, yang dimana proses penyisipan file PDF ke dalam file audio WAV menggunakan metode Paillier Steganografi menunjukkan bahwa embedding hanya berhasil dilakukan pada file PDF berukuran kecil. Contohnya, file SD01 dengan ukuran 160 KB berhasil di-embed ke dalam beberapa file audio seperti SA02 hingga SA06, dan file SD02 di-embed hanya 1 file audio yaitu SA06 seperti ukuran file WAV setelah embedding bertambah sesuai dengan ukuran ciphertext, yang menandakan proses embedding berhasil dilakukan. Selisih file pdf (SD01) dan file audio (SA01) pada embed pertama yang gagal adalah 3,69 kali lipat. Hal ini menunjukkan bahwa metode Paillier Steganografi WAV masih dapat bekerja dengan baik ketika ukuran file yang di-embed relatif kecil.

E. Pengujian Ekstrak Chipertext ECC Steganografi WAV

Tabel 5 Perbandingan Ekstrak Chipertext ECC Steganografi WAV

Analisis Perbandingan Chipertext ECC Steganografi WAV (ekstrak)				
No.	Nama File Audio WAV	Besaran File WAV Sebelum diekstrak (KB)	Besaran File WAV Sesudah diekstrak (KB)	Lama Waktu Mengekstrak (detik)
1	stego_SA01	3.250	1.625	16,66
	stego_SA02	8.906	4.453	12,30
	stego_SA03	16.306	8.153	18,86
	stego_SA04	23.972	11.986	30,74
	stego_SA05	36.230	18.115	38,17
	stego_SA06	51.555	25.778	53,06
2	stego_SA01	3.250	1.625	10,28
	stego_SA02	8.906	4.453	11,36
	stego_SA03	16.306	8.153	19,58
	stego_SA04	23.972	11.986	29,69
	stego_SA05	36.230	18.115	38,23
	stego_SA06	51.555	25.778	56,57
3	–	–	–	–
	–	–	–	–
	–	–	–	–
	stego_SA04	3.250	11.986	28,66
	stego_SA05	8.906	18.115	34,81
	stego_SA06	16.306	25.778	47,75

Untuk hasil pengujian terdapat pada Tabel 5, terlihat bahwa ukuran file WAV setelah proses ekstraksi selalu lebih kecil dibandingkan ukuran file sebelum diekstraksi. Contohnya, file stego_SA01 yang semula berukuran 3.250 KB menjadi 1.625 KB setelah diekstrak, dan hal ini konsisten terjadi pada file-file lainnya seperti stego_SA04 (23.972 KB menjadi 11.986 KB) dan stego_SA06 (51.555 KB menjadi 25.778 KB). Hal ini menandakan bahwa proses ekstraksi berhasil dilakukan dan data tersembunyi berhasil diambil dari file audio, meskipun ukuran file hasil ekstraksi menunjukkan bahwa proses ini tidak mengembalikan seluruh ukuran file awal.

F. Pengujian Ekstrak Chipertext Paillier Steganografi WAV

Tabel 6 Perbandingan Ekstrak Chipertext Paillier Steganografi WAV

Analisis Perbandingan Chipertext Paillier Steganografi WAV (ekstrak)				
No.	Nama File Audio WAV	Besaran File WAV Sebelum diekstrak (KB)	Besaran File WAV Sesudah diekstrak (KB)	Lama Waktu Mengekstrak (detik)
1	stego_SA01	–	–	–
	stego_SA02	8.906	4.453	18,54
	stego_SA03	16.306	8.153	21,22
	stego_SA04	23.972	11.986	28,51
	stego_SA05	36.230	18.115	39,37
	stego_SA06	51.555	25.778	52,11
2	stego_SA01	–	–	–
	stego_SA02	–	–	–
	stego_SA03	–	–	–
	stego_SA04	–	–	–
	stego_SA05	–	–	–
	stego_SA06	51.555	25.778	55,17

Untuk hasil pengujian terdapat pada Tabel 6, terlihat bahwa ukuran file WAV setelah proses ekstraksi mengalami penurunan dari ukuran aslinya sebelum diekstrak. Sebagai contoh, file stego_SA02 awalnya memiliki ukuran 8.906 KB dan setelah diekstrak menjadi 4.453 KB. Hal serupa juga terlihat pada file stego_SA06 yang dari 51.555 KB menjadi 25.778 KB setelah proses ekstraksi. Hal ini menunjukkan bahwa proses ekstraksi dengan algoritma Paillier dalam steganografi WAV menghasilkan file yang lebih kecil, yang kemungkinan disebabkan oleh penghilangan bagian data tersembunyi dalam file audio stego.

G. Pengujian Dekripsi ECC Steganografi WAV

Tabel 7 Perbandingan Dekripsi ECC Steganografi WAV

Analisis Perbandingan Dekripsi ECC Pada Steganografi WAV					
No.	Nama Folder	Nama File Extracted PDF	Besaran Extracted PDF Sebelum didekripsi (KB)	Besaran Extracted PDF Sesudah didekripsi (KB)	Lama Waktu Mendekripsi (detik)
1	SD01	pdfextracted_stego_SA01.wav.wav	7	7	4,20
		pdfextracted_stego_SA02.wav.wav	7	7	,50

		pdfextracted_stego_SA03.wav.wav	7	7	3,78
		pdfextracted_stego_SA04.wav.wav	7	7	2,99
		pdfextracted_stego_SA05.wav.wav	7	7	0,32
		pdfextracted_stego_SA06.wav.wav	7	7	,33
2	SD02	pdfextracted_stego_SA01.wav.wav	41	41	2,83
		pdfextracted_stego_SA02.wav.wav	41	41	1,60
		pdfextracted_stego_SA03.wav.wav	41	41	1,56
		pdfextracted_stego_SA04.wav.wav	41	41	,90
		pdfextracted_stego_SA05.wav.wav	41	41	,78
		pdfextracted_stego_SA06.wav.wav	41	41	1,30
3	SD03	–	–	–	
		–	–	–	
		–	–	–	
		pdfextracted_stego_SA04.wav.wav	542	542	1,10
		pdfextracted_stego_SA05.wav.wav	542	542	0,51
		pdfextracted_stego_SA06.wav.wav	542	542	0,38

Untuk hasil pengujian terdapat pada Tabel 11, terlihat bahwa ukuran file PDF hasil ekstraksi dari file steganografi WAV pada folder SD01 hingga SD03 tidak mengalami perubahan ukuran, yaitu tetap sebesar 7 KB untuk folder SD01, 41KB untuk folder SD02, dan 542KB untuk folder SD03 sebelum dan sesudah proses ekstraksi. Hal ini menunjukkan bahwa proses dekripsi menggunakan algoritma ECC pada steganografi WAV tidak mempengaruhi ukuran file hasil ekstraksi.

H. Pengujian Dekripsi Paillier Steganografi WAV

Tabel 8 Perbandingan Dekripsi Paillier Steganografi WAV

Analisis Perbandingan Dekripsi Paillier Pada Steganografi WAV					
No.	Nama Folder	Nama File Extracted PDF	Besaran Extracted PDF Sebelum didekripsi (KB)	Besaran Extracted PDF Sesudah didekripsi (KB)	Lama Waktu Mendekripsi (detik)
1	SD01	–	–	–	–
		pdfextracted_stego_SA02.wav.wav	160	7	19,43
		pdfextracted_stego_SA03.wav.wav	160	7	19,35
		pdfextracted_stego_SA04.wav.wav	160	7	24,50
		pdfextracted_stego_SA05.wav.wav	160	7	18,11
		pdfextracted_stego_SA06.wav.wav	160	7	19,01
2	SD02	–	–	–	–
		–	–	–	–
		–	–	–	–
		–	–	–	–
		–	–	–	–
		pdfextracted_stego_SA06.wav.wav	1.001	41	73,18

Untuk hasil pengujian terdapat pada Tabel 12, terlihat bahwa ukuran file PDF setelah proses ekstraksi mengalami penurunan dari ukuran aslinya sebelum diekstrak. Sebagai contoh, file pdfextracted_stego_SA02.wav.wav awalnya memiliki ukuran 160 KB dan setelah diekstraksi menjadi 7 KB. Sedangkan pada folder SD02, file pdfextracted_stego_SA06.wav.wav mengalami penurunan ukuran dari 1.001 KB menjadi 41 KB. Hal ini menunjukkan bahwa proses ekstraksi algoritma Paillier dalam steganografi WAV menghasilkan file yang lebih kecil.

I. Pengujian Hash MD5

Tabel 9 Perbandingan Hash MD5 Pada Chipertext PDF

Analisis Perbandingan Hash MD5 Pada Chipertext PDF		
No.	Nama Chipertext File PDF	Hasil Hash MD5
1	encryptedecc_SD01	5136951458f754b7d2cd25250d8cc616
	encryptedpaillier_SD01	a394af2ce37a0b1d0f8d902292345d64
2	encryptedecc_SD02	5dfc1f5ceacbcad2ec143341b2545476
	encryptedpaillier_SD02	3aa0859914a5f53f8950200567ff5e18
3	encryptedecc_SD03	547fa97b6c3482398dea7dd95052ec3b
	encryptedpaillier_SD03	85f645cdd64ac4714cbcf70de5694f3c
4	encryptedecc_SD04	f486c64e31149e40309249264a4ec429
	encryptedpaillier_SD04	c31145ec50b037b3e6accaf9a9966d7f
5	encryptedecc_SD05	73f6de7ef56f7e81d0e25acd386d3872
	encryptedpaillier_SD05	9c0303970f5666b9fa335a494a70a91c
6	encryptedecc_SD06	9d22c6e10b86b5d4e95525116cd97316
	encryptedpaillier_SD06	422a3d83846ecfbdf23338ddd057e12c

Untuk hasil pengujian terdapat pada Tabel 15, terlihat bahwa proses enkripsi file PDF menggunakan dua algoritma, yaitu ECC dan Paillier, berhasil dilakukan sehingga menghasilkan file encryptedecc_SD01 hingga encryptedpaillier_SD06. Hal ini ditunjukkan oleh keberadaan nilai hash MD5 yang unik pada setiap file *ciphertext*, yang menandakan bahwa setiap file hasil enkripsi memiliki keunikan tersendiri dan tidak mengalami duplikasi atau kegagalan proses enkripsi.

J. Perhitungan Embed Chipertext ECC Steganografi WAV

Tabel 10 Analisis Perhitungan Chipertext ECC Steganografi WAV

Analisis Perhitungan Chipertext ECC Steganografi WAV (embed)						
No.	Nama File PDF	Nama File Audio WAV	Besaran File WAV Sebelum diembedd (KB)	Besaran File WAV Sesudah diembedd (KB)	Perhitungan MSE	Perhitungan PNSR (dB)
1	SD01	SA01	590	3.250	0.0000000001	99.64478
		SA02	1.616	8.906	0.0000000000	96.40522
		SA03	3.398	16.306	0.0000000000	100.10528
		SA04	4.995	23.972	0.0000000000	101.85845
		SA05	7.548	36.230	0.0000000000	102.62905

		SA06	10.741	51.555	0.0000000014	89.21608
2	SD02	SA01	590	3.250	0.0000000002	94.45595
		SA02	1.616	8.906	0.0000000001	93.08563
		SA03	3.398	16.306	0.0000000001	97.80812
		SA04	4.995	23.972	0.0000000001	100.12234
		SA05	7.548	36.230	0.0000000001	101.36294
		SA06	10.741	51.555	0.0000000014	89.18269
3	SD03	SA01	590	Gagal	–	–
		SA02	1.616	Gagal	–	–
		SA03	3.398	Gagal	–	–
		SA04	4.995	23.972	0.0000000004	92.43691
		SA05	7.548	36.230	0.0000000003	94.63529
		SA06	10.741	51.555	0.0000000015	88.74668

Untuk hasil perhitungan terdapat pada Tabel 16, terlihat bahwa proses penyisipan ciphertext ke dalam file audio WAV menggunakan algoritma ECC berhasil dilakukan pada folder SD01 hingga SD03. Hal ini ditunjukkan oleh ukuran file WAV yang mengalami kenaikan setelah proses embedding, serta nilai MSE yang sangat kecil (hampir nol) dan PSNR yang tinggi di atas 88 dB, menandakan bahwa kualitas audio tetap terjaga.

K. Perhitungan Embed Chipertext Pailllier Steganografi WAV

Tabel 11 Analisis Perhitungan Chipertext Paillier Steganografi WAV

Analisis Perhitungan Chipertext Paillier Steganografi WAV (embed)						
No.	Nama File PDF	Nama File Audio WAV	Besaran File WAV Sebelum diembedd (KB)	Besaran File WAV Sesudah diembedd (KB)	Perhitungan MSE	Perhitungan PNSR (dB)
1	SD01	SA01	590	Gagal	–	–
		SA02	1.616	8.906	0.0000000003	88.51810
		SA03	3.398	16.306	0.0000000002	93.96389
		SA04	4.995	23.972	0.0000000001	96.78709
		SA05	7.548	36.230	0.0000000001	98.62943
		SA06	10.741	51.555	0.0000000014	89.07530
2	SD02	SA01	590	Gagal	–	–
		SA02	1.616	Gagal	–	–
		SA03	3.398	Gagal	–	–
		SA04	4.995	Gagal	–	–
		SA05	7.548	Gagal	–	–
		SA06	10.741	51.555	0.0000000016	88.38203

Untuk hasil perhitungan terdapat pada Tabel 17, terlihat bahwa proses penyisipan ciphertext ke dalam file audio WAV menggunakan algoritma Paillier berhasil dilakukan pada sebagian file dalam folder SD01 dan SD02. Hal ini ditunjukkan oleh file-file seperti SA02, SA03, SA04, dan SA06 dalam folder SD01, serta SA06 dalam folder SD02, yang mengalami kenaikan ukuran file setelah proses embedding. Nilai MSE yang sangat kecil dan PSNR di atas 88 dB menunjukkan bahwa kualitas audio masih terjaga dengan baik, menandakan keberhasilan embedding meskipun dengan algoritma Paillier yang memiliki kompleksitas lebih tinggi.

V. KESIMPULAN

Berdasarkan hasil pengujian dan analisa sistem dari penelitian yang telah dilakukan, terdapat beberapa kesimpulan bahwa algoritma ECC lebih efisien dari segi kecepatan pengujian dan besarnya data file pdf dibanding algoritma Paillier. dimana pada tahap enkripsi untuk algoritma Paillier kecepatan enkripsi lebih lama sekitar 10 kali lipat dan data file pdf yang dihasilkan sekitar 22 kali lipat lebih besar dibandingkan algoritma ECC. Sehingga bisa disimpulkan bahwa algoritma ECC lebih efisien daripada algoritma Paillier.

REFERENSI

[1] K. Bansal, A. Agrawal, dan N. Bansal, “bit (LSB) Embedding Approach,” no. Icoei, hal. 64–69, 2020.

[2] A. Amir Alkodri, S. Supardi, R. Maulana, dan L. Fahreni, “Implementation Of Rivest Chiper 6 and Blowfish Algorithm for Mobile-Based Message Cryptography,” *J. Sisfotek Glob.*, vol. 12, no. 2, hal. 87, 2022,

[3] M. A. Kustian, “Analisis Forensik Penggunaan Fungsi Hash Dalam Menentukan Keaslian Video, Metadata Image Dan Magic Number File,” *J. Sains, Nalar, dan Apl. Teknol. Inf.*, vol. 2, no. 2, hal. 10–16, 2023,

[4] S.- Sallu dan Q. Qammaddin, “Keamanan Data Pembelajaran Online Jaringan Komputer Di Perguruan Tinggi,” *Instruksional*, vol. 2, no. 1, hal. 35, 2020,

[5] A. P. Ratnasari dan F. A. Dwiyanto, “Metode Steganografi Citra Digital,” *Sains, Apl. Komputasi dan Teknol. Inf.*, vol. 2, no. 2, hal. 52, 2020,

[6] S. D. Nurcahya, “Implementasi Aplikasi Kriptografi Metode Kode Geser Berbasis Java,” *J. Nas. Komputasi dan Teknol. Inf.*, vol. 5, no. 4, hal. 694–697, 2022,

[7] Masita, “Perbandingan Algoritma Ecdh Dan Algoritma Ecc Dalam Mengamankan Pesan Gambar,” *J. Inf. dan Teknol. Ilm.*, vol. 8, no. 1, hal. 25–29, 2020.

[8] M. F. Mulya, N. Rismawati, dan D. Trisanto, “Analisis Dan Perancangan Simulasi Algoritma Paillier Cryptosystem Pada Pesan Text Dengan Presentation Format Binary, Octal, Hexadecimal dan Base64,” *Fakt. Exacta*, vol. 13, no. 4, hal. 208, 2021,

[9] R. Siburian, L. Lindawati, dan A. Aryanti, “Implementasi Steganografi Audio Mp3 Dan Wav Untuk File Pdf Pada Smartphone Android Dengan Menggunakan Metode Lsb (Least Significant BIT),” *Sntibd*, hal. 400–404, 2017,

[10] R. W. Dobson, “Developments in Audio File

Formats," *Int. Comput. Music Conf. ICMC Proc.*, 2000.

[11] U. Suwardoyo, "Aplikasi Steganografi Menggunakan Metode LSB Pada Media Audio," *Proceeding KONIK (Konferensi Nas. Ilmu ...*, hal. 508–512, 2021,

[12] M. R. Zayana, I. Fitri, F. Fauziah, dan A. Gunaryarti, "Penerapan Message Diggest Algorithm MD5 untuk Pengamanan Data Karyawan PT. Swifect Berbasis Desktop," *J. JTIK (Jurnal Teknol. Inf. dan Komunikasi)*, vol. 6, no. 3, hal. 386–394, 2022,

[13] S. Supardi, A. A. Alkodri, dan B. Isnanto, "Teknik Steganografi Penyembunyian Pesan Text Rahasia Pada Citra Digital Dengan Metode Least Significant Bit," *J. Sisfotek Glob.*, vol. 11, no. 1, hal. 70, 2021,

[14] R. Rivaldo, H. Handrizal, dan H. Herriyance, "Pengamanan Pesan Menggunakan Metode MLSB PRNG dan Kompresi File dengan Algoritma RLE pada File Audio," *J. Sist. Inf. Bisnis*, vol. 11, no. 1, hal. 1–8, 2021,

[15] A. D. Hendrata dan A. Prihanto, "Analisis Kualitas Suara Stego Audio Penyisipan Informasi Tersembunyi dengan Metode Least Significant Bit," *J. Informatics Comput. Sci.*, vol. 2, no. 03, hal. 178–184, 2021,