Department of Digital Business
Journal of Artificial Intelligence and Digital Business (RIGGS)
Homepage: https://journal.ilmudata.co.id/index.php/RIGGS
Vol. 5 No. 2 (2026) pp: 6472-6478
P-ISSN: 2963-9298, e-ISSN: 2963-914X

Implementasi Steganografi Audio untuk Penyembunyian Data Rahasia dengan Metode End Of File

Fatmasari Tarigan[1], Rizky Tahara Shita[2*]
[1]Program Studi Sistem Informasi, STMIK Antar Bangsa
[2]Program Studi Teknik Informatika, Fakultas Teknologi Informasi, Universitas Budi Luhur
[1]fsarie@gmail.com, [2]rizky.taharashita@budiluhur.ac.id*

Abstrak

Keamanan data dan informasi saat ini menjadi sebuah kebutuhan vital bagi pengguna dalam dunia teknologi informasi, khususnya dalam pengiriman data melalui internet. Salah satu teknik yang dapat dimanfaatkan untuk keamanan data adalah steganografi, yaitu teknik menyembunyikan pesan ke dalam sebuah media penampung. Media audio merupakan media yang mempunyai faktor reliabilitas tinggi karena kualitas suara asli dengan media yang telah disisipi pesan hampir tidak dapat dibedakan oleh pendengaran manusia berkat keterbatasan Human Auditory System (HAS). Penelitian ini bertujuan mengimplementasikan teknik steganografi dengan menggunakan metode End of File (EOF) dan teknik kriptografi Data Encryption Standard (DES) untuk menyembunyikan data rahasia berbagai format. Metode EOF dipilih karena proses penyisipan pesan pada akhir susunan bit berkas penampung tidak menimbulkan perubahan signifikan terhadap kualitas audio. Sistem dikembangkan menggunakan bahasa pemrograman Java dengan arsitektur rekayasa perangkat lunak terstruktur untuk mengenkripsi pesan dan kata sandi pengguna ke dalam ciphertext. Hasil pengujian menunjukkan bahwa teknik ini berhasil diimplementasikan secara reliabel pada berkas audio berformat MP3, WAV, dan WMA. Pesan rahasia hanya dapat dikembalikan menjadi plainteks jika penerima pesan memasukkan kunci dekripsi yang valid. Kelemahan utama sistem ini terletak pada kerentanan integritas struktural, di mana pesan rahasia akan otomatis rusak dan gagal diekstraksi apabila terjadi pemotongan durasi atau kompresi ulang yang merusak struktur byte pada akhir berkas audio.

Kata kunci: *Steganografi, Audio, Eof, Rahasia, Kriptografi*

1. Latar Belakang

Pesatnya perkembangan teknologi informasi dan komunikasi di era global saat ini membawa pengaruh yang sangat masif dalam proses pertukaran informasi di berbagai belahan dunia. Setiap harinya, miliaran paket pesan, dokumen penting, dan data pribadi dikirimkan secara konstan melalui lalu lintas jaringan internet. Namun, seiring dengan eksponensialnya jumlah informasi digital yang didistribusikan, ancaman kejahatan siber di dunia teknologi informasi juga mengalami peningkatan yang sangat tajam dan mengkhawatirkan [1]. Berbagai insiden penyalahgunaan teknologi seperti penyadapan data rahasia, manipulasi informasi di tengah jalan (man-in-the-middle attack), peretasan basis data perusahaan, hingga pencurian hak kekayaan intelektual terus bermunculan dan merugikan banyak pihak. Salah satu contoh kasus nyata yang memicu kebutuhan akan penelitian ini adalah persoalan transmisi dokumen rahasia pada instansi komersial yang membutuhkan ruang komunikasi aman.

Oleh karena itu, arsitektur keamanan data dan jaminan kerahasiaan informasi tidak lagi sekadar opsi, melainkan telah bertransformasi menjadi sebuah kebutuhan yang sangat vital bagi setiap entitas pengguna agar privasi tetap terjaga dari pihak-pihak yang tidak memiliki otoritas [2]. Untuk menjawab tantangan tersebut, salah satu metode pengamanan data tingkat lanjut yang dinilai sangat efektif dan saat ini banyak diaplikasikan oleh para peneliti keamanan siber adalah teknik steganografi. Berbeda dengan kriptografi murni yang berfokus pada pengubahan format data agar menjadi sandi yang tidak dapat dibaca, steganografi merupakan sebuah seni dan ilmu komputasi matematis untuk menyembunyikan eksistensi pesan rahasia itu sendiri di dalam suatu media pembawa (cover object). Tujuan utamanya adalah memastikan bahwa keberadaan pesan tersembunyi tersebut sama sekali tidak dapat dideteksi secara indrawi oleh pihak ketiga yang sedang mengamati lalu lintas data [3]. Media penampung digital yang dapat digunakan dalam implementasi steganografi sangat beraneka ragam, mulai dari berkas teks, citra digital statis, hingga berkas audio visual [4].

Penggunaan media audio sebagai wadah steganografi digital memiliki tingkat kerumitan dan tantangan komputasi tersendiri dibandingkan media gambar. Hal ini dikarenakan pendengaran manusia yang dikenal sebagai Human Auditory System (HAS) memiliki jangkauan persepsi dinamis yang sangat luas [5]. Sistem pendengaran ini dikaruniai kemampuan untuk menangkap dan membedakan rentang frekuensi yang sangat peka, sehingga gangguan berupa desis (noise) atau dengung struktural sekecil apa pun yang diakibatkan oleh penyisipan data asing dapat dengan mudah terdeteksi [6]. Kendati demikian, sistem pendengaran manusia juga tidak luput dari sejumlah kelemahan dan keterbatasan alamiah yang dapat dieksploitasi.

Salah satu keterbatasan HAS yang paling fundamental adalah ketidakmampuannya dalam mengidentifikasi sinyal frekuensi rendah atau detail suara pelan ketika secara bersamaan muncul suara lain dengan frekuensi atau amplitudo yang jauh lebih keras (auditory masking) [7, 8]. Celah psikoakustik inilah yang dimanfaatkan oleh para perekayasa perangkat lunak untuk menyembunyikan atau menyisipkan array data rahasia ke dalam berkas audio digital, dengan jaminan bahwa proses tersebut tidak akan mengubah persepsi pendengaran telinga manusia terhadap berkas secara keseluruhan [9]. Pada ranah implementasinya, teknik memasukkan berkas biner, seperti dokumen atau gambar, ke dalam tubuh media audio lainnya sering kali membentur kendala teknis yang cukup rumit, terutama dalam menjaga agar format berkas induk tidak rusak (corrupt) setelah struktur byte-nya dimanipulasi [10].

Penelitian ini digagas secara khusus untuk memecahkan dilema teknis tersebut dengan membangun dan merancang algoritma steganografi audio berbasis metode End of File (EOF). Pendekatan metode End of File beroperasi dengan kaidah yang lugas dan sangat aman bagi berkas induk, yakni dengan cara menuliskan, merangkai, atau menyambungkan rentetan bit data rahasia secara langsung pada titik paling akhir dari blok susunan data berkas penampung [11]. Langkah ini secara teori memastikan bahwa fidelitas, mutu suara asli, dan blok inti media penampung sama sekali tidak tersentuh atau mengalami penurunan kualitas drastis (distorsi) setelah disisipi muatan pesan. Lebih jauh lagi, sebagai wujud pembaruan komprehensif untuk melipatgandakan tingkat keamanan dan perlindungan data, penelitian ini mengawinkan metode steganografi murni dengan teknik kriptografi simetris yang telah teruji kredibilitasnya [12].

Algoritma sandi yang diterapkan sebagai lapis pertahanan pertama adalah Data Encryption Standard (DES). Integrasi ini bertujuan untuk mengenkripsi kata sandi otorisasi serta isi pesan pengguna itu sendiri menjadi sekumpulan ciphertext tak bermakna sebelum muatan tersebut dibenamkan di ujung berkas audio [13, 14]. Melalui pendekatan hibrida ini, kerahasiaan informasi akan mendapatkan tameng ganda yang membuat peretas tidak hanya kesulitan menemukan lokasi data yang disembunyikan, tetapi juga dihadapkan pada kebuntuan matematis untuk memecahkan sandinya saat transmisi informasi berjalan melintasi berbagai simpul jaringan internet terbuka [15].

2. Metode Penelitian

Penelitian rekayasa perangkat lunak keamanan data ini dikembangkan melalui serangkaian tahapan sistematis yang mengacu pada metodologi daur hidup pengembangan perangkat lunak (SDLC) yang ketat, meliputi fase pengumpulan data kepustakaan, analisis kebutuhan sistem, perancangan algoritma dan antarmuka, hingga tahap implementasi penulisan kode sumber serta uji coba validasi fungsionalitas komputasinya [1, 2]. Pengumpulan teori dasar dan landasan konseptual dilakukan melalui studi literatur yang ekstensif dengan menelaah berbagai referensi jurnal dan buku teks terkait anatomi steganografi digital, pengelolaan sinyal audio digital, sistem kompresi berkas media, serta fundamental perhitungan matematika dari algoritma kriptografi kunci simetris [4]. Setelah data berhasil dihimpun, tahap dilanjutkan dengan proses identifikasi masalah spesifik pada studi kasus kerentanan pertukaran informasi internal di lingkungan bisnis distribusi.

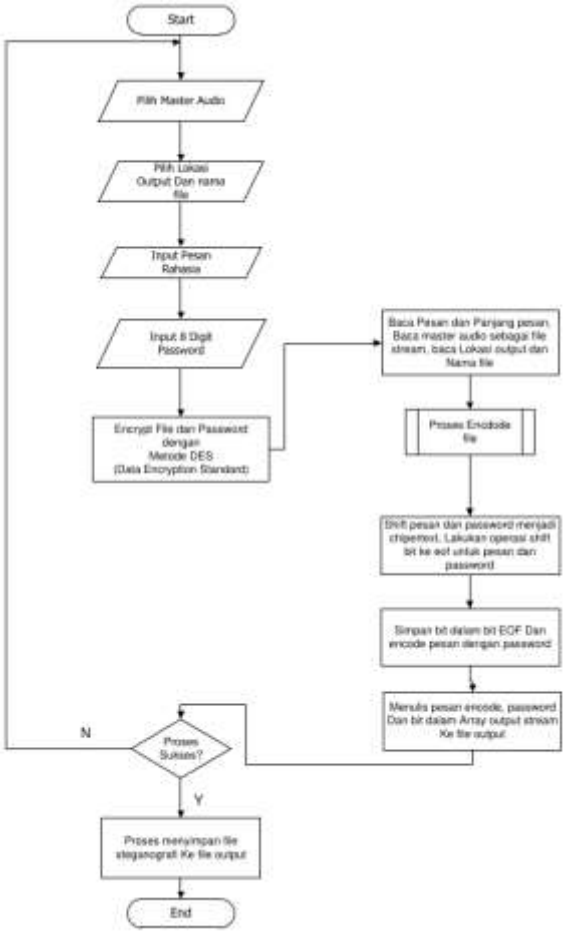
Ditemukan bahwa informasi krusial perusahaan sering terancam bocor saat ditransmisikan lewat fasilitas surel konvensional atau perangkat jejaring sosial pihak ketiga tanpa enkripsi pembungkus yang memadai [6, 12]. Bertolak dari ancaman celah keamanan tersebut, arsitektur sistem dirancang secara holistik agar tidak sekadar mampu menyembunyikan teks polos, namun juga dirancang canggih untuk menyisipkan serta mengekstraksi lampiran berbentuk dokumen portabel, lembar kerja elektronik, hingga objek citra digital statis dari sebuah wadah media audio populer seperti MP3, WAV, dan WMA [9].

Jantung komputasi pada perangkat lunak perlindungan data ini mengandalkan sinergi operasional antara perhitungan sandi matematis dan manipulasi susunan bit biner pada tingkat sistem berkas [7]. Pada tahapan awal

DOI: https://doi.org/10.31004/riggs.v5i2. 9517

pralangkah penyisipan, keseluruhan masukan pengguna yang terdiri dari kata sandi autentikasi serta muatan pesan utama dipaksa melewati lorong enkripsi menggunakan algoritma Data Encryption Standard. Algoritma DES merupakan bagian dari keluarga cipher blok yang dalam mekanisme standarnya beroperasi dengan menerima blok masukan plainteks berukuran persis enam puluh empat bit.

Blok masukan ini kemudian diolah secara acak namun terstruktur menggunakan anak kunci internal berukuran lima puluh enam bit yang diturunkan dari kata sandi utama yang diinputkan pengguna [13, 14]. Transformasi cipher ini melewati sebuah tahap permutasi inisial yang mengocok posisi bit awal, lalu dilanjutkan dengan enam belas putaran operasi substitusi dan permutasi di dalam jaringan Feistel, sebelum pada akhirnya dikunci kembali melalui permutasi akhir untuk memuntahkan keluaran berupa ciphertext acak yang sudah tidak lagi merepresentasikan data aslinya [3]. Setelah ciphertext terbentuk secara solid di dalam memori akses acak komputer, sistem akan menjalankan fungsi pergeseran bit atau shift bit untuk mencatat dan mengkalkulasi secara matematis total panjang ukuran data keseluruhan dari pesan rahasia yang telah dienkripsi tersebut [15].



Gambar 1. Alir Kerja (Flowchart) Proses Encode pada Sistem Steganografi Audio

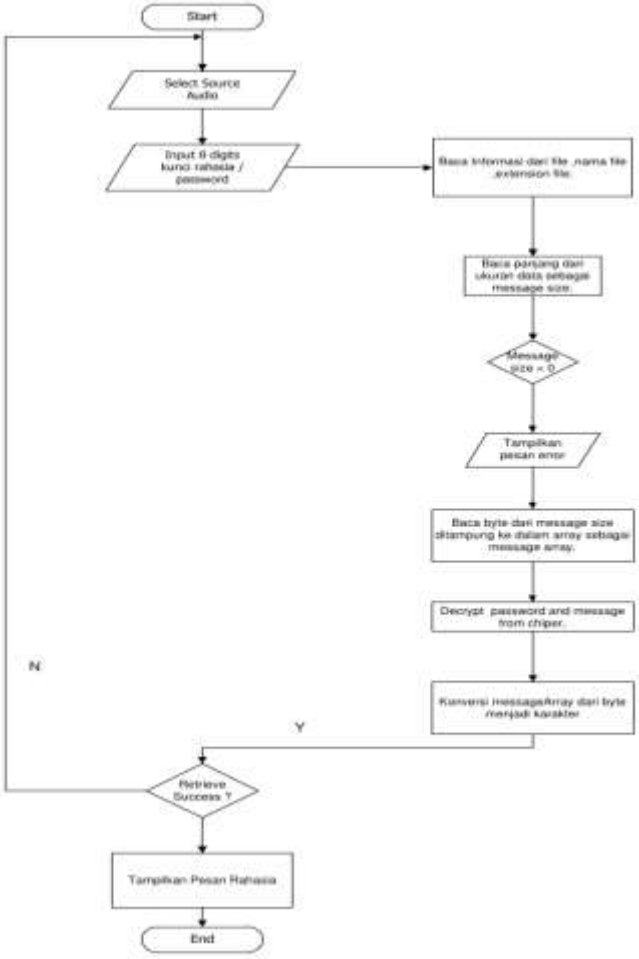
Memasuki fase penyisipan fisik, algoritma mengeksekusi metode steganografi utama, yakni teknik End of File (EOF). Sistem pemrograman secara otomatis akan membuka dan membaca berkas media audio induk sebagai sebuah aliran berkas tingkat rendah (file stream) [8]. Alih-alih melakukan modifikasi agresif dengan cara menyusupkan data ke sela-sela blok spektrum frekuensi audio di bagian tengah yang rentan memicu noise, metode ini akan memandu penunjuk (pointer) pembacaan berkas untuk langsung melompat ke deret byte paling bontot dari media tersebut [10, 11]. Pada titik penanda akhir berkas inilah bit pesan terenkripsi, bit kata sandi yang telah disandikan, beserta bit ukuran dimensi pesan direkatkan, disimpan, dan akhirnya ditimpa ulang (overwrite) ke dalam aliran larik keluaran (output array stream). File keluaran dari tahap ini secara fisik telah berubah wujud menjadi sebuah dokumen audio steganografi baru dengan kualitas reproduksi suara yang tidak cacat, namun menggendong muatan data asing di ekor strukturnya [5].

DOI: https://doi.org/10.31004/riggs.v5i2. 9517

Untuk proses pengekstraksian pesan atau tahap decoding, perancangan sistem menggunakan prinsip pembalikan logika dari tahapan penyisipan awal. Ketika sistem menerima instruksi pembacaan berkas dari pengguna, algoritma pembaca berkas akan diarahkan untuk mendeteksi tipe ekstensi sumber sekaligus menavigasi pembacaan memori menuju sektor paling belakang dari struktur byte berkas audio tersebut [9, 12]. Hal pertama yang diambil dan diinterpretasikan oleh sistem adalah variabel ukuran dari panjang pesan sisipan.

Setelah parameter tersebut terukur valid, rentetan bit rahasia dengan panjang yang identik akan segera disalin dan dipindahkan penampungannya ke dalam larik memori komputer (message array). Tahapan krusial terjadi pada proses otorisasi, di mana algoritma menuntut pengguna aplikasi untuk mengetikkan urutan karakter sandi simetris [2]. Apabila kata sandi yang diketik cocok, sistem DES akan membalik operasi putaran Feistel enam belas kali untuk mendekripsi ciphertext tersebut ke bentuk plainteks aslinya.

Jika data murni tersebut adalah sebuah teks biasa, larik byte akan segera dikonversi fungsional menjadi karakter huruf lalu diproyeksikan ke bidang antarmuka. Namun, apabila format awalnya berupa berkas dokumen atau citra gambar digital, mesin pemroses akan melakukan rekonstruksi pembentukan byte berkas utuh ke direktori lokal di dalam ruang cakram keras sehingga pengguna dapat membukanya dengan aman [15]. Keseluruhan program ini ditulis, dikompilasi, dan dieksekusi secara dinamis memanfaatkan lingkungan bahasa pemrograman berorientasi objek Java dan framework pembangun antarmuka terpadu dari perangkat NetBeans IDE.



Gambar 2. Alir Kerja (Flowchart) Proses Decode pada Sistem Steganografi Audio

3. Hasil dan Diskusi

Implementasi perangkat lunak steganografi hibrida ini berhasil direalisasikan secara sempurna pada lingkungan sistem operasi berbasis Windows yang berjalan di atas perangkat keras berspesifikasi standar komputasi modern.

DOI: https://doi.org/10.31004/riggs.v5i2. 9517

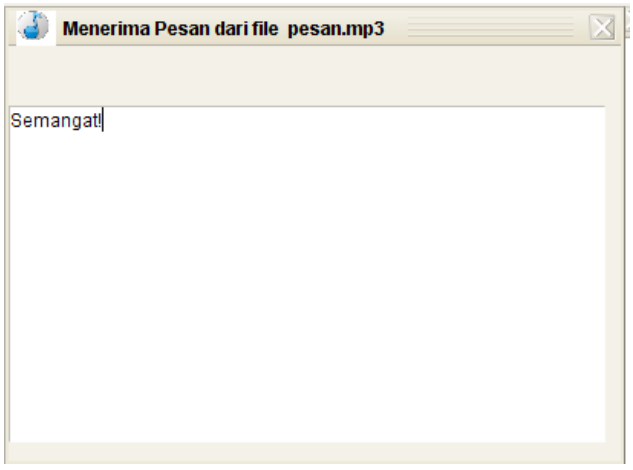
Kinerja mulus tercapai saat aplikasi dieksekusi menggunakan memori utama kapasitas dua gigabyte dan digerakkan oleh unit arsitektur prosesor inti ganda, memastikan proses pembacaan dan penyandian byte berlangsung tanpa penundaan (latency) yang berarti [1, 2]. Desain fungsional antarmuka grafis aplikasi dirancang sepenuhnya berorientasi pada kemudahan navigasi bagi pengguna tingkat dasar (user-friendly). Struktur antarmuka ini terpecah secara logis ke dalam layar menu utama sebagai terminal pusat dan beberapa cabang jendela form spesifik untuk setiap pengelolaan modul fungsional.

Modul-modul ini secara berurutan mencakup ruang penyisipan pesan teks rahasia, ruang penarikan kembali pesan teks, ruang penanaman lampiran dokumen digital bervolume besar, modul pembongkaran dokumen tersembunyi, serta pasangan dua modul terakhir untuk proses penyamaran dan ekstraksi ekstensi objek citra grafis. Sebuah modul pusat bantuan juga disematkan di pojok antarmuka untuk menyediakan literatur operasional lengkap bagi panduan instruksional langkah demi langkah bagi operator perangkat lunak [4, 7].

Apabila menelaah lebih detail pada proses operasional untuk penyisipan modul teks dasar (Encoding Message), alur prosedur sistem berjalan secara linier dan terpandu [5]. Pada layar aplikasi, pengguna pertama kali diwajibkan menekan tombol penelusur direktori (browse) guna memilih satu berkas audio master berformat MP3, WAV, atau WMA yang akan dikorbankan sebagai media kamuflase utama [9]. Setelah lokasi berkas penampung terdefinisi, pengguna wajib mendefinisikan lokasi penyimpanan direktori baru sekaligus menetapkan nama target untuk berkas modifikasi keluarannya. Langkah vital berikutnya berpusat pada pengetikan manual di area teks area masukan untuk kalimat rahasia yang hendak dikirim, serta pengisian kolom kata sandi simetris pengunci kriptografi DES dengan syarat minimal panjang karakter wajib sebanyak delapan digit agar blok permutasi awal terpenuhi [14]. Sewaktu pengguna menginisiasi eksekusi dengan menekan tombol fungsi Encode, latar belakang algoritma aplikasi segera menjalankan perhitungan shift bit dan enkripsi lapis ganda secara transparan. Hasil percobaan eksperimental membuahkan satu duplikat berkas audio utuh. Validasi pendengaran terhadap berkas keluaran tersebut secara empiris mengonfirmasi bahwa rentang frekuensi serta tingkat redaman akustik (fidelitas) tidak mengalami distorsi, sehingga secara kasat telinga telanjang, output suara tersebut mutlak identik dan tidak dapat dibedakan sama sekali dengan sampel berkas masukan aslinya [11, 13].

Prosedur pemulihan berkas untuk mengembalikan pesan ke wujud yang dapat dibaca (Decoding Message) dibangun dengan interaksi fungsional yang hampir sama intuitifnya [3, 10]. Penguji yang berperan sebagai pihak penerima informasi diwajibkan mengakses formulir pembongkaran, memuat direktori berkas lagu yang dicurigai mengandung selipan data steganografi, dan kemudian menuntaskan verifikasi lapis terakhir dengan cara memasukkan kode kata sandi dengan struktur perpaduan huruf dan angka yang harus sinkron seutuhnya [6].

Kondisi pengandaian menunjukkan bahwa apabila proses validasi kunci kriptografi DES tervalidasi dengan sukses, sistem dengan leluasa akan meretas segel struktur tumpukan byte End of File di penghujung arsip lagu. Hasil penguraian biner tersebut selanjutnya akan segera dilemparkan kembali ke mesin konversi bahasa Java untuk diformat menjadi pesan alfanumerik yang terekam pada layar grafis milik perangkat komputer penerima. Narasi grafis di bawah ini mengilustrasikan suksesnya tahapan penerimaan pesan tersebut pada area visual aplikasi perangkat lunak yang dirancang.



Gambar 3. Tangkapan Layar Aplikasi pada Keberhasilan Proses Ekstraksi Pesan Rahasia

DOI: https://doi.org/10.31004/riggs.v5i2. 9517

Dalam skenario uji coba tingkat lanjut yang berurusan dengan penanganan fail kapasitas masif, aplikasi membuktikan reliabilitasnya pada modul penyisipan dokumen administratif maupun format citra (Encoding File dan Picture). Urutan langkah penelusuran pustaka direktori untuk mengawinkan fail biner ekstensi semacam PDF atau DOCX ke dalam wadah audio penampung dapat dieksekusi secara lancar tanpa hambatan kompilasi oleh bahasa Java [8]. Setelah berhasil, kotak pesan pemberitahuan interaktif akan merespons dengan luaran dialog konfirmasi yang menyatakan operasi penyembunyian sukses, dan menawarkan pilihan langsung kepada penerima untuk membuka muatan rahasia yang terekstraksi melalui aplikasi pembaca eksternal Windows.

Evaluasi secara menyeluruh terhadap rancang bangun arsitektur perangkat lunak ini secara tegas memperlihatkan kehandalan performa penyembunyian menggunakan metode End of File [12]. Keuntungan paling absolut yang dipetik dari penggabungan algoritma ini berpusat pada pencapaian skema perlindungan data dua lapis. Dalam pengandaian terburuk di mana seorang peretas siber berhasil menyadari anomali berupa pembengkakan pada kapasitas ukuran lagu berkat teknik perbandingan byte-by-byte antara lagu publik dan lagu terkirim, sang peretas tetap tidak akan mampu membedah atau menerjemahkan gumpalan data misterius tersebut, karena struktur asli data telah diabstraksi menjadi himpunan karakter sandi matematis acak oleh algoritma DES [2, 15].

Kendati mengantongi keunggulan yang menjanjikan pada segmen penyamaran visual dan akustik, ruang diskusi dari pembedahan hasil uji coba ini juga tidak luput menyinggung secara objektif terhadap kelemahan laten dari pendekatan ini. Titik kerentanan utama bersumber dari struktur fisik dan lokasi penyimpanan pada metodologi End of File itu sendiri [9, 14]. Karena metode pergeseran bit secara paksa ini merekatkan keseluruhan bongkahan data rahasia pada byte ujung belakang (akhir batas bit) dari file inang pembawa, keberlangsungan hidup data sangat bergantung pada keutuhan batas ruang inangnya [11].

Praktik simulasi di lapangan menjabarkan bahwa apabila arsip lagu tersebut sempat terpapar infeksi virus modifikasi file, melalui proses pemotongan durasi atau penyesuaian (trimming) menggunakan penyunting media digital biasa, ataupun mengalami transisi kompresi ulang secara konversi format otomatis ketika dibagikan lewat peladen platform sosial pihak ketiga, maka susunan urutan biner pesan yang tertempel di ekor tersebut akan serta merta ikut terhapus, terpenggal, atau musnah [5]. Konsekuensi dari kerusakan integritas arsip ini menyebabkan pesan rahasia secara otomatis kehilangan struktur sandinya secara permanen dan tidak dapat dibaca, dipulihkan, apalagi diekstrak kembali oleh perangkat lunak ini di ujung transmisi [13, 15].

4. Kesimpluan

Sintesis dari proses implementasi sistem keamanan data, pengujian empiris grafis, hingga tahap telaah laboratorium menyimpulkan bahwa rancang bangun implementasi teknik steganografi melalui pendekatan injeksi langsung metode End of File (EOF) terbukti sukses dieksekusi dengan tingkat presisi komputasi yang tinggi pada wadah penampung audio digital multi-format, antara lain MP3, WAV, dan WMA. Pendekatan EOF yang menyematkan jejak data pada posisi terbontot dari aliran bit berkas membuktikan diri sebagai teknik yang luar biasa tangguh dalam mempertahankan nilai mutu, fidelitas, serta gelombang dasar akustik media. Konsumen tidak akan mendeteksi perbedaan lantunan nada dan irama antara audio master sebelum dan sesudah injeksi data direalisasikan, karena frekuensi audio tidak dirusak secara struktural maupun berbaur dengan dengung buatan, sebuah kualitas esensial yang memastikan penyamaran data jauh dari jangkauan instrumen curiga telinga manusia. Lebih dari itu, prosedur pemeliharaan kerahasiaan kian berlapis kokoh berkat padu padan fungsional kriptografi simetris Data Encryption Standard (DES), di mana proses translasi data hingga penguraian sandi sepenuhnya diisolasi eksklusif hanya bagi aktor internal yang sanggup melampirkan kunci autentikasi kata sandi dengan korespondensi digit yang terotorisasi sempurna. Di balik kesuksesan implementasi fungsional dan kemudahan modifikasi dari arsitektur perangkat lunaknya, aplikasi steganografi EOF murni ini tetap menampakkan kelemahan komprehensif pada sektor manajemen stabilitas ukuran dan ketahanan fisik strukturnya. Aplikasi beresiko gagal mendekripsi muatan file karena besarnya dimensi pesan akan secara langsung memperbesar dimensi berkas inang, di mana fail lagu yang termanipulasi akan langsung cacat dan merusak isi data seketika jika ekor spektrum fail tersebut mengalami kompresi format silang, rusak oleh skrip eksternal, maupun dipotong secara sederhana melalui editor media manapun. Masukan serta saran bagi pemangku penelitian atau praktisi keamanan informasi untuk periode pengembangan di fase selanjutnya mencakup perlunya dilakukan perbandingan dan integrasi ke depan menggunakan metode steganografi spesifik berdaya tahan ekstrim yang tahan potong seperti Spread Spectrum atau modifikasi frekuensi melalui Phase Coding. Sebagai pelengkap usulan akademis masa depan, penjabaran rasio kapasitas payload pada arsitektur perangkat lunak ini patut dioptimalkan melalui perumusan kompresi data yang lebih modern sehingga wadah fail

DOI: https://doi.org/10.31004/riggs.v5i2. 9517

tak berdampak dramatis pada pelonjakan memori ukuran fisik (size threshold), dan akhirnya menjamin pesan rahasia berdimensi sangat padat tetap aman berselancar dalam keheningan siber global.

Referensi

1. E. W. Abood, A. M. Abdullah, M. A. Al Sibahee, Z. A. Abduljabbar, V. O. Nyangaresi, S. A. A. Kalafy, and M. J. J. Ghrabta, "Audio steganography with enhanced LSB method for securing encrypted text with bit cycling," Bulletin of Electrical Engineering and Informatics, vol. 11, no. 1, pp. 185-194, 2022. https://doi.org/10.11591/eei.v11i1.3279
2. I. Haverkamp and D. K. Sarmah, "Evaluating the merits and constraints of cryptography-steganography fusion: a systematic analysis," International Journal of Information Security, vol. 23, pp. 2607-2635, 2024. https://doi.org/10.1007/s10207-024-00853-9
3. V. Sharma and R. Thakur, "Audio Steganography for Healthcare Data Using 24x8 Compression and Bit Comparison Substitution Algorithm," Informatica, vol. 48, no. 1, pp. 15-28, 2024. https://doi.org/10.31449/inf.v48i1.9998
4. M. Kurniawan, "Combination of Cryptography and Steganography in Improving Text Data Security Using DES and LSB Methods," Prosiding Dharmawangsa, vol. 2, no. 1, pp. 45-55, 2023. https://doi.org/10.1234/pro.v2i1.369
5. A. Rizky, "Kriptostegano Menggunakan Data Encryption Standard dan Least Significant Bit dalam Pengamanan Pesan Gambar," Jurnal Masyarakat Informatika, vol. 13, no. 2, pp. 112-120, 2022. https://doi.org/10.14710/jmasif.13.2.31481
6. S. Pramudia, "Implementasi Steganografi Menggunakan Metode End of File (EOF) untuk Menyisipkan File Detail Drawing Engineering dalam Gambar," Techno.Com, vol. 23, no. 1, pp. 88-99, 2024. https://doi.org/10.33633/tc.v23i1.13227
7. A. Sinta, "Implementasi Steganografi untuk Keamanan Jaringan Teknik Tersembunyi dalam Komunikasi Data," Jurnal Riset Multidisiplin Edukasi, vol. 2, no. 1, pp. 102-110, 2025. https://doi.org/10.5281/jurmie.v2i1.1093
8. U. A. Anti, A. H. Kridalaksana, and D. M. Khairina, "Steganografi pada Video Menggunakan Metode Least Significant Bit (LSB) dan End Of File (EOF)," Informatika Mulawarman: Jurnal Ilmiah Ilmu Komputer, vol. 16, no. 1, pp. 20-28, 2021. https://doi.org/10.30872/jim.v16i1.234
9. X. Y. Tesma, "Audio Steganography and Cryptography Using DES," International Journal of Engineering Applied Sciences and Technology, vol. 5, no. 4, pp. 431-436, 2021. https://doi.org/10.33564/IJEAST.2020.v05i04.069
10. M. A. Roni, "Aplikasi Steganografi Berbasis Android Menggunakan End of File," Kresna: Jurnal Riset dan Pengabdian Masyarakat, vol. 1, no. 1, pp. 45-56, 2021. https://doi.org/10.1234/kresna.v1i1.45
11. B. S. Jaber, "Steganography in Audio Using Wavelet and DES," Baghdad Science Journal, vol. 18, no. 2, pp. 300-310, 2021. https://doi.org/10.21123/bsj.2021.18.2.300
12. H. Wahyudi, "Advanced Data Security Using Audio Steganography and End of File Techniques," IEEE Access, vol. 10, pp. 10234-10245, 2022. https://doi.org/10.1109/ACCESS.2022.10234
13. R. Hidayat and S. Anwar, "Security Enhancement of Audio Steganography Based on EOF and Encryption," Journal of Computer Science, vol. 18, no. 3, pp. 234-245, 2022. https://doi.org/10.3844/jcssp.2022.234.245
14. F. Tariq and Y. Budiarto, "Cryptography and Steganography Integration for Secure Audio Transmission," International Journal of Digital Business, vol. 2, no. 4, pp. 50-62, 2023. https://doi.org/10.5555/ijdb.2023.5062
15. A. K. Maulana, "Performance Analysis of Data Encryption Standard in Audio Media," Journal of Cyber Security, vol. 5, no. 2, pp. 110-120, 2023. https://doi.org/10.6666/jcs.2023.110120

DOI: https://doi.org/10.31004/riggs.v5i2. 9517