KRESNA: Jurnal Riset dan Pengabdian Masyarakat
Volume 2, Nomor 2, November, 2022, Hal: 232-238 E-ISSN: 2809-6509

Aplikasi Steganografi Berbasis Android Menggunakan End of File dengan Enkripsi Rivest Code 4

Android-Based Steganography Application Using End of File with Rivest Code 4 Encryption

Achmad Aditya Ashadul Ushud

Fakultas Teknologi Informasi
Universitas Budi Luhur
E-mail: achmad.aditya@budiluhur.ac.id

Abstract

The research aim is to hide audio files through video media using the EOF (End of File) method and Rivest Code 4 encryption. The EOF (End of File) method is used as a message-hiding technique in steganography by inserting a message at the end of the container media file. Audio files are encrypted using Rivest Code 4 (RC4) and inserted at the end of the video file as container media. The results of this study are android-based steganography applications that can maintain the confidentiality of audio files while maintaining the quality of video files as container media. In the testing process, the resulting video file size has changed a lot. The larger the inserted audio file, the larger the resulting video file size, with an average processing time of 48 seconds.

Keywords **:** audio, encryption, steganography, video

Abstrak

Tujuan penelitian untuk menyembunyikan file audio melalui media video menggunakan metode EOF (End of File) dan enkripsi Rivest Code 4. Metode EOF (End of File) digunakan sebagai teknik penyembunyian pesan dalam steganografi dengan menyisipkan pesan pada akhir *file* media penampung. *File* audio yang dienkrip menggunakan Rivest Code 4 (RC4) akan disisipkan pada bagian akhir *file* video sebagai media penampung. Hasil penelitian ini aplikasi steganografi berbasis android yang dapat menjaga kerahasiaan *file* audio dengan tetap mempertahankan kualitas *file* video sebagai media penampung. Pada proses pengujian dihasilkan ukuran file video yang dihasilkan mengalami banyak perubahan. Semakin besar file audio yang disisipkan, semakin besar ukuran file video yang dihasilkan, dengan rata-rata waktu proses 48 detik.

Kata kunci **:** audio, enkripsi**,** steganografi, video

1. PENDAHULUAN

Teknik untuk menjaga kerahasiaan pesan tidak terbatas pada kriptografi. Metode lain adalah steganografi. Steganografi adalah teknik menyamarkan pesan rahasia di dalam pesan lain dengan cara disisipkan sehingga pesan rahasia tidak dapat dideteksi[1]. Berbeda dengan kriptografi yang menyembunyikan makna pesan namun tetap memiliki akses terhadap pesan tersebut, steganografi menjaga kerahasiaannya dengan menyembunyikan pesan [2].

Salah satu metode steganografi adalah *End of File* (EOF), dimana sebuah pesan disisipkan di akhir media file. Sebelum disisipkan pesan diubah menjadi bit biner, dan ketika citra *grayscale* digunakan, pesan diubah menjadi kode ASCII [3].

End of File (EOF) adalah pengembangan dari *Least Significant Bit* (LSB). Dalam metode ini, pesan disisipkan di akhir file. Tidak ada batasan jumlah pesan yang dimasukkan menggunakan metode ini. Namun, efek sampingnya adalah ukuran file bertambah melebihi ukuran aslinya [4].

Misalnya, citra *grayscale* menyisipkan pesan "aku". Kode ASCII untuk pesan tersebut adalah 97 107 117. Kolom citra sebelum dan setelah disisipkan pesan terlihat pada Tabel 1 dan Tabel 2 berikut:

Tabel 1: Kolom Citra Sebelum Disisipkan Pesan

123	200	120	145	190
50	130	100	80	70
30	45	123	145	60
100	122	138	182	197
300	178	90	50	30
135	70	87	67	129

Tabel 2: Kolom Citra Setelah Disisipkan Pesan

123	200	120	145	190
50	130	100	80	70
30	45	123	145	60
100	122	138	182	197
300	178	90	50	30
135	70	87	67	129
35	97	107	117	

Teknologi RC4 dikembangkan pada tahun 1987 oleh Ronald Rivest dari RSA Data Security Inc. RC4 telah meningkatkan kecepatan dan kesederhanaannya dalam mengelola banyak aplikasi, yang memudahkan penggunaan aplikasi perangkat lunak. RC4 memproses unit atau memasukkan data satu kali. Enkripsi atau dekripsi dapat dilakukan pada objek yang panjang. Algoritma ini tidak harus menunggu input data dalam jumlah tertentu.

RC4 menggunakan panjang kunci dari 1 byte hingga 256-byte yang digunakan untuk menginisialisasi tabel yang panjangnya 256 byte. Tabel ini digunakan untuk pseudorandom generasi berikutnya menggunakan XOR dengan plaintext untuk membuat ciphertext. Setiap elemen tabel dipermutasi setidaknya sekali. Algoritma RC4 memiliki dua fase, instalasi kunci dan enkripsi. Dalam konfigurasi kunci S-bit (S adalah panjang kunci), kunci enkripsi digunakan untuk menganalisis variabel kode menggunakan dua array, state dan kunci, dan sejumlah S hasil penggabungan. Operasi penggabungan ini dilakukan dengan pertukaran byte, fungsi modulo, dan formula lainnya. Operasi modulo adalah proses yang mengembalikan nilai sisa suatu pembagian. Variabel enkripsi dihasilkan dari pengaturan kunci menggunakan operasi XOR dengan teks biasa untuk menghasilkan teks yang sudah terenkripsi. XOR adalah operasi logika yang membandingkan dua bit biner. Jika nilainya berbeda, akan menjadi nilai 1. Jika kedua bit sama maka hasilnya adalah 0. Lalu penerima pesan akan mendekripsinya dengan operasi XOR dan satu kunci yang sama secara berurutan sehingga menghasilkan pesan dalam teks biasa [5].

Penelitian [6] menjelaskan cara menggunakan file audio format MP3 dan WAV sebagai media untuk menyisipkan pesan. Penelitian terdahulu dan sekarang memiliki persamaan pada data yang digunakan yaitu file audio, dan terdapat perbedaan pada teknik steganografi dan prosesnya. Pada penelitian sebelumnya menggunakan LSB (Least Significant Bit) dan file diimpor langsung ke format MP3 dan WAV, sedangkan pada penelitian ini, pesan terlebih dahulu dienkripsi menggunakan algoritma RC4 sebelum disisipkan.

Pada penelitian [7] telah dijelaskan bagaimana cara memasukkan pesan teks ke dalam video, perbedaannya dapat dilihat pada ukuran video yang semakin besar ketika pesan tersebut dimasukkan ke dalam video. Hasil penelitian [8] dan [9] menjelaskan bagaimana menyembunyikan pesan terenkripsi algoritma RSA ke dalam gambar sehingga keamanan pesan tersebut semakin diperkuat. Penelitian [10] bertujuan untuk mengamankan file dokumen menggunakan algoritma kriptografi RC4 dan algoritma Huffman sebagai kompresi file dengan memadukan teknik steganografi algoritma End of File (EOF). Pada penelitian ini perubahan gambar terjadi secara fisik seperti perubahan sudut, kontras, dan kedalaman warna gambar yang akan mempengaruhi isi pesan pada gambar. Untuk membuat pesan di gambar tidak terbaca dan mengubah ukuran file sehingga pesan tidak dapat dibaca.

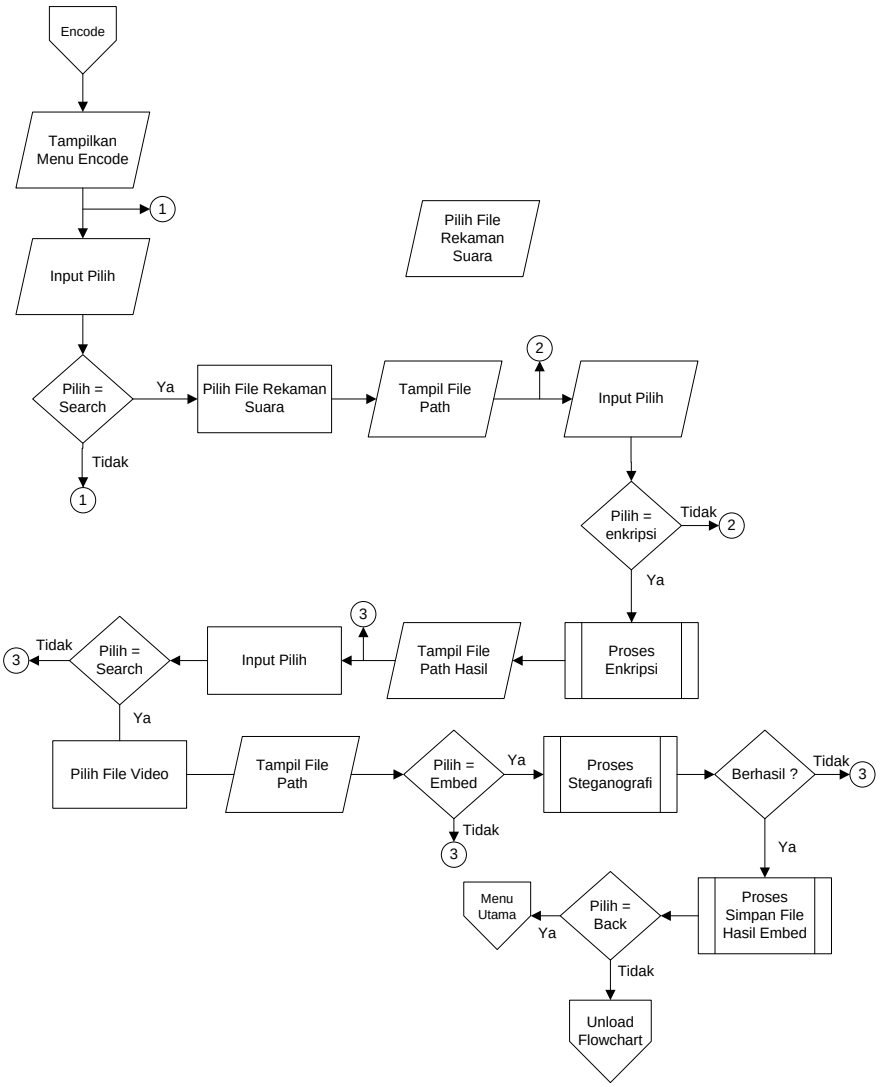
2. METODE PENELITIAN

2.1 Analisis

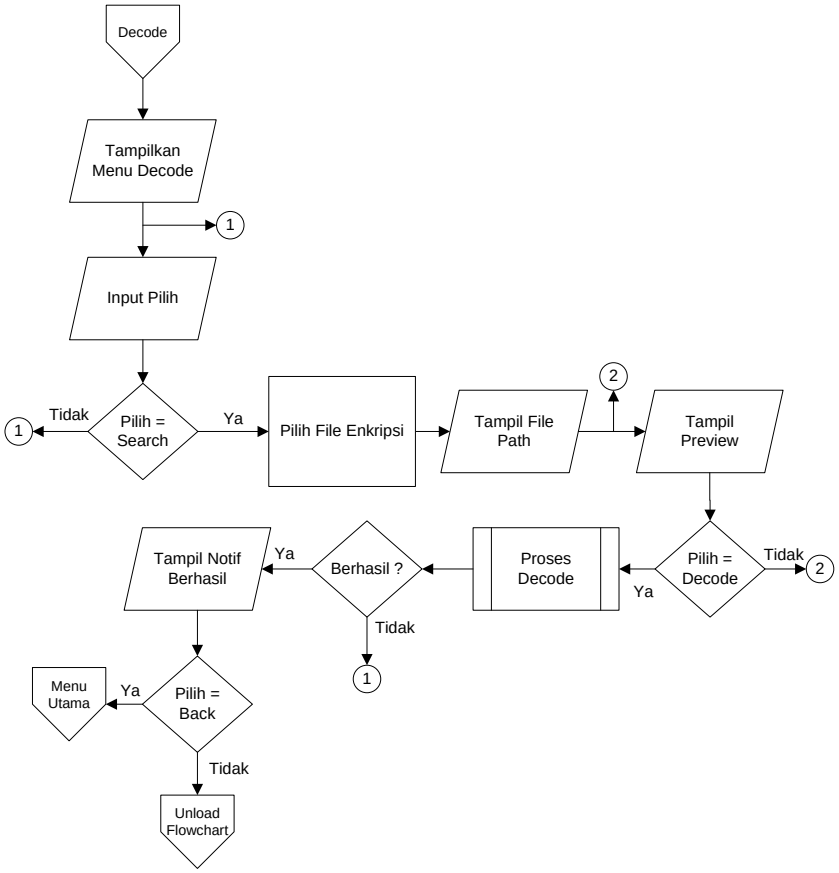
Membuat konsep rancangan aplikasi steganografi yang memiliki fungsi untuk menyisipkan dan mengekstraksi pesan. Terdapat dua proses dalam implementasi steganografi ini yaitu proses enkripsi dan dekripsi. Aplikasi mengenkripsi file audio dan menyisipkannya ke dalam video. Dekripsi adalah proses ekstraksi untuk mengeluarkan file audio.

2.2 Perancangan Aplikasi

Aplikasi yang dirancang adalah steganografi berbasis android yang mempunyai beberapa buah menu, yaitu menu encode, decode, about, help, dan tombol exit. Aplikasi ini akan digunakan untuk mengenkripsi sekaligus melakukan proses *embed* (steganografi). Menu encode adalah menu untuk mengenkripsi file audio kemudian menyisipkannya ke dalam file video. Untuk menu decode adalah menu untuk mendekripsi atau mengeluarkan kembali file audio yang telah dienkripsi. Menu help adalah menu yang berisi tentang cara penggunaan dan pemakaian aplikasi, sedangkan tombol exit digunakan untuk keluar dari aplikasi. Diagram alir untuk menu encode dapat dilihat pada Gambar 1 dan untuk menu decode dapat dilihat pada Gambar 2.



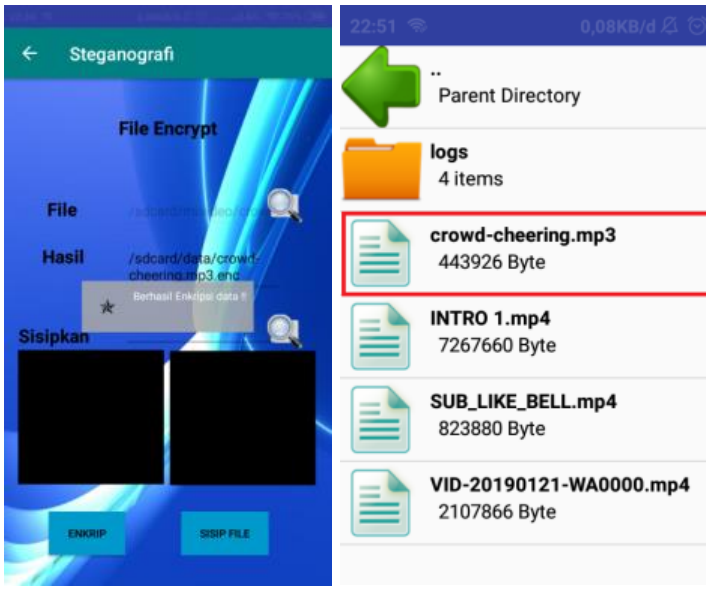
Gambar 1: Flowchart Menu Encode



Gambar 2: Flowchart Menu Decode

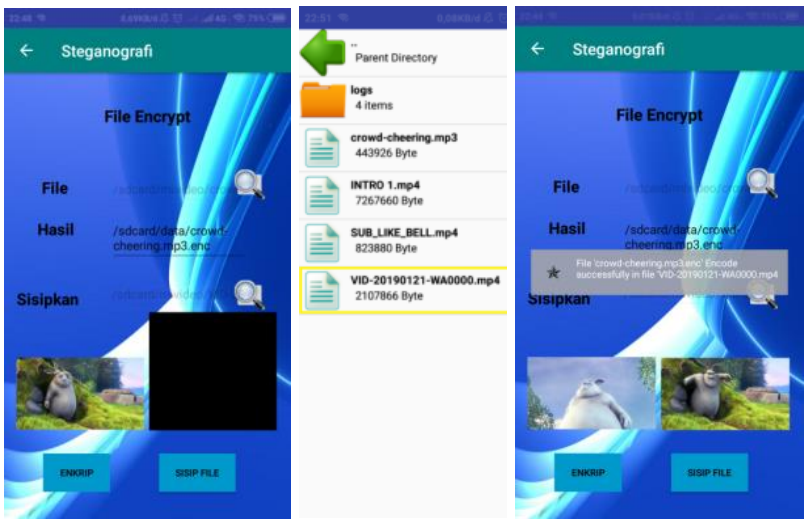
3. HASIL DAN PEMBAHASAN

Pada Gambar 3 merupakan tampilan layar proses memilih file audio yang akan dienkripsi menggunakan algoritma RC4. Tampilan layar memperlihatkan file MP3 yang akan dijadikan sebagai pesan yang akan disisipkan.



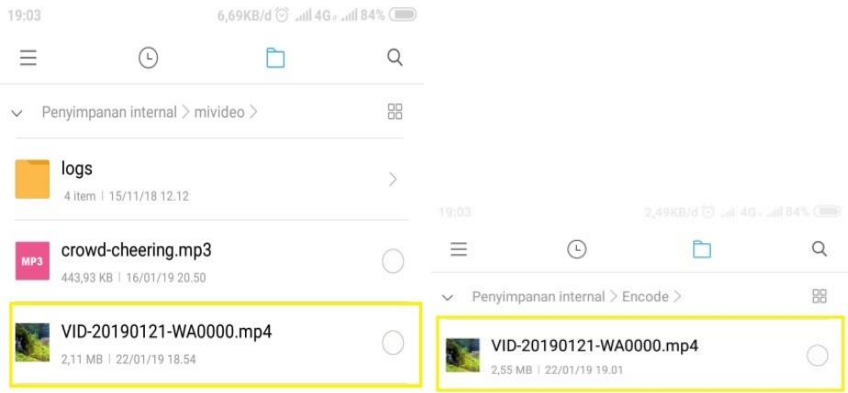
Gambar 3: Tampilan Layar Pemilihan File Audio

Setelah berhasil melakukan proses enkripsi terhadap file audio, langkah berikutnya menentukan file video sebagai media penampung pesan terenkripsi seperti diperlihatkan pada Gambar 4.



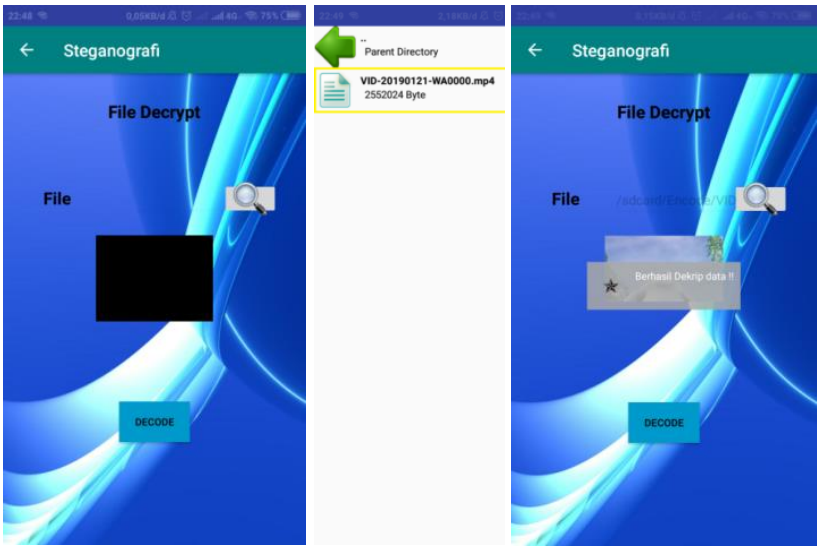
Gambar 4: Tampilan Layar Pemilihan File Video

Pada Gambar 4 memperlihatkan proses penyisipan pesan file MP3 dengan teknik steganografi EOF ke dalam file video dengan format MP4. Setelah proses penyisipan, ukuran file video yang dihasilkan mengalami banyak perubahan. Semakin besar file audio yang disisipkan, semakin besar ukuran file video yang dihasilkan. Perubahan ukuran file video dapat dilihat pada Gambar 5.



Gambar5: Perubahan Ukuran File Video

Setelah melakukan enkripsi dan penyisipan file, selanjutnya melakukan proses dekripsi untuk mengekstraksi file audio yang tersimpan dalam file video seperti digambarkan pada Gambar 6.



Gambar: Tampilan Layar Menu Decode

Dari Gambar 6 dijelaskan bahwa proses dimulai dengan memilih file MP4 yang telah berisi pesan enkripsi. Setelah file dipilih maka dilakukan proses dekripsi. Proses dekripsi ini akan mengembalikan file audio asli.

Tahap selanjutnya adalah pengujian yang digunakan untuk membandingkan ukuran video sebelum dan sesudah penggunaan steganografi tergantung pada besar file audio yang disertakan. Pengujian juga dilakukan untuk melihat waktu yang diperlukan dalam penyisipan file audio. Hasil pengujian tersebut disajikan pada Tabel 3.

Tabel 3: Pengujian Berdasarkan Ukuran File dan Waktu Proses EOF

No.	File Video	Ukuran Video Sebelum	File Audio	Ukuran Audio	Ukuran Video Sesudah	Waktu Proses (menit:detik:milidetik)
1.	VID-01.mp4	2,11 MB	AUD-01.mp3	443 KB	2,55 MB	00:51:40
2.	VID-02.mp4	1,69 MB	AUD-02.mp3	705 KB	2,10 MB	00:47:34
3.	VID-03.mp4	2,91 MB	AUD-03.mp3	512 KB	3,22 MB	01:46:28
4.	VID-04.mp4	1,86 MB	AUD-04.mp3	360 KB	2,07 MB	00:22:09
5.	VID-05.mp4	1,55 MB	AUD-05.mp3	230 KB	1,71 MB	00:12:27

4. KESIMPULAN DAN SARAN

Kesimpulan yang diperoleh dari penelitian ini antara lain:

a. Aplikasi steganografi berbasis android dengan metode EOF dapat menjaga kerahasiaan file audio sehingga menarik untuk diterapkan.
b. Implementasi algoritma RC4 pada proses enkripsi file audio dapat berjalan dengan baik.
c. Aplikasi dapat mengembalikan pesan (file audio) yang disisipkan pada file video secara utuh tanpa mengalami perubahan sedikitpun.
d. Ukuran file audio yang terenkripsi mengubah ukuran file video.

Adapun saran agar aplikasi ini dapat dikembangkan ke tahap selanjutnya, antara lain:

a. Meningkatkan kinerja proses enkripsi dan penyisipan file sehingga tidak hanya file MP3 namun file audio lainnya seperti WAV atau MIDI.
b. Menambahkan algoritma kompresi untuk mengubah ukuran file hasil enkripsi menjadi lebih kecil.

DAFTAR PUSTAKA

[1] T. Taburet, P. Bas, W. Sawaya, and J. Fridrich, “Natural steganography in JPEG domain with a linear development pipeline”, *IEEE Transactions on Information Forensics and Security*, vol. 16, pp. 173-186, 2020.

[2] S. R. Widianto, "Desain dan Analisa Algoritma Steganografi dengan Metode Spread Spectrum Berbasis PCMK (Permutasi Chaotic Multiputaran Mengecil dan Membesar) Menggunakan Matlab", *Jurnal Elektra*, vol. 3, no. 1, pp. 37-46, 2018.

[3] A. E. Putri, A. Kartikadewi, dan L. A. A. Rosyid, "Implementasi Kriptografi Dengan Algoritma Advanced Encryption Standard (AES) 128 Bit Dan Steganografi Menggunakan Metode End of File (EOF) Berbasis Java Desktop Pada Dinas Pendidikan Kabupaten Tangerang", *Applied Information Systems and Management (AISM)*, vol. 3, no. 2, pp. 69-78, 2020.

[4] A. Rohmanu, "Implementasi Kriptografi dan Steganografi Dengan Metode Algoritma DES dan Metode End of File", *Jurnal Informatika SIMANTIK*, vol. 1, no. 2, pp 1-11, 2017.

[5] A. E. Setiawan, A. Pasaribu, dan R. C. Pratama, "Penerapan Steganografi Pada Citra Digital Menggunakan Metode Least Significant Bit (LSB) Kombinasi RC4 Berbasis Mobile Android", *Aisyah Journal of Informatics and Electrical Engineering*, vol. 2, no. 1, pp. 18-28, 2020.

[6] R. Siburian, L. Lindawati, A. Aryanti, "Implementasi Steganografi Audio MP3 dan WAV untuk File PDF pada SmartPhone Android dengan Menggunakan Metote LSB (Least Significant Bit)", *Seminar Nasional Teknologi Informasi, Bisnis, dan Desain*, STMIK – Politeknik PalComTech, 12 Juli 2017.

[7] U. A. Anti, A. H. Kridalaksana, dan D. M. Khairina, "Steganografi pada Video Menggunakan Metode Least Significant Bit (LSB) dan End of File (EOF)", *Informatika Mulawarman: Jurnal Ilmiah Ilmu Komputer*, vol. 12, no. 2, pp. 104-111, 2017.

[8] M. H. N. Aini, "Implementasi Steganografi Penyembunyian Pesan pada Gambar Menggunakan Metode EOF dengan Enkripsi RSA Berbasis Android", *Ubiquitous: Computers and its Applications Journal*, vol. 2, no. 2, pp. 131-136, 2019.

[9] Usanto, "Aplikasi Enkripsi dengan Algoritma Rivest Shami Aldeman (RSA) dan Parity Bit Coding untuk File Multimedia", *Jurnal Elektro & Informatika Swadharma (JEIS)*, vol. 2, no. 2, pp. 17-28, 2022.

[10] Y. Septianto, G. Barovih, Pujiono, "Implementasi Multi Algoritma pada Aplikasi Enkripsi dalam Mengamankan File", *TEKNOMATIKA*, vol. 12, no. 1, pp. 1-12, 2022.