p-ISSN: 2301-5373
e-ISSN: 2654-5101

Rancang Bangun Algoritma ChaCha20-Poly1305 Berbasis SHA-256 Untuk Pengamanan Dokumen Penting Masyarakat

I Made Chandra Widjaya[a1], I Ketut Gede Suhartana[a2], I Luh Gede Astuti[b3]

[a]Program Studi Informatika, Fakultas Matematika dan Ilmu Pengetahuan Alam
Universitas Udayana, Indonesia
[1]widjaya.2208561009@unud.ac.id
[2]ikg.suhartana@unud.ac.
[3]ig.astuti@unud.ac.

Abstract

Data security is a critical aspect in protecting sensitive information, especially in digital environments where data is vulnerable to unauthorized access. This study aims to implement the ChaCha20-Poly1305 algorithm with a SHA-256-based key for data protection, as well as to evaluate its performance and security through several testing scenarios. The method involves implementing the ChaCha20-Poly1305 algorithm using keys generated from SHA-256 hashing, followed by testing using parameters such as Avalanche Effect, image visual analysis, MSE, PSNR, and performance comparison with the AES-256-GCM algorithm. The results show that the system successfully restores data with 100% similarity after decryption. The Avalanche Effect achieves an average value of 50.038%, indicating high sensitivity to key changes. Additionally, entropy increases from 0.7159 to 7.9993 and pixel correlation decreases from 0.7914 to -0.0085, indicating a high level of randomness in the encrypted data. The MSE value of 16,715.31 and PSNR value of 6.19 dB indicate significant differences between original and encrypted data. Based on these results, it can be concluded that the ChaCha20-Poly1305 algorithm with a SHA-256 key is capable of securing data with a high level of randomness while maintaining data integrity after the decryption process.

Keywords: *ChaCha20-Poly1305, SHA-256, Encryption, Data Security, Avalanche Effect, Entropy.*

1. Pendahuluan

Dalam era kemajuan teknologi yang semakin pesat, informasi yang beredar di dunia maya menjadi semakin rentan terhadap ancaman. Keamanan data menjadi sangat penting untuk melindungi informasi sensitif, mengingat potensi dampak negatif yang bisa terjadi jika data penting disadap oleh pihak yang tidak berwenang [1]. Berbagai jenis serangan siber, seperti phishing, malware, ransomware, dan serangan Distributed Denial of Service (DDoS), menjadi ancaman nyata yang dapat merugikan individu, organisasi, bahkan negara. Oleh karena itu, penerapan langkah-langkah keamanan siber yang kuat menjadi suatu keharusan. Kriptografi menjadi salah satu solusi utama dalam menjaga kerahasiaan dan integritas data [2]. Dengan menggunakan algoritma enkripsi yang kompleks, informasi sensitif dapat diubah menjadi suatu bentukan yang tidak terbaca oleh pihak yang tidak memiliki otoritas. Enkripsi merupakan proses mengubah teks atau data menjadi format yang tidak dapat dibaca atau sulit dipahami oleh siapa pun yang tidak memiliki otorisasi [3]. Sebaliknya, dekripsi adalah proses mengembalikan *ciphertext* menjadi *plaintext* dengan menggunakan kunci dan algoritma yang sesuai.

Meskipun teknologi enkripsi telah banyak diterapkan, masih terdapat berbagai permasalahan keamanan data yang krusial di Indonesia. Berbagai surat berharga milik masyarakat, termasuk dokumen yang memiliki nilai hukum dan ekonomi, kerap menjadi sasaran penyalahgunaan,

pemalsuan, maupun manipulasi data. Berdasarkan data pada Direktori Putusan Mahkamah Agung Republik Indonesia, dalam kategori perkara "Tanah", tercatat sebanyak 378 putusan pada tahun 2023 dan 230 putusan pada tahun 2024, sehingga totalnya terdapat 608 putusan dalam rentang dua tahun tersebut [4]. Lebih lanjut, pada Mei 2021, dugaan kebocoran data menimpa BPJS Kesehatan, di mana sekitar 279 juta data peserta dilaporkan diperjualbelikan di forum peretas daring [5].

Penelitian-penelitian sebelumnya telah berfokus pada keamanan data menggunakan ChaCha20-Poly1305. Beyne et al. (2025) menunjukkan bahwa ChaCha20-Poly1305 memiliki tingkat keamanan dan efisiensi yang tinggi, dengan probabilitas keberhasilan serangan yang sangat kecil yaitu sekitar 1 dari 10^{31} [6], namun kajian tersebut bersifat teoretis-kriptanalitik dan tidak mengimplementasikan sistem nyata untuk pengamanan berkas dokumen kemasyarakatan. Serrano et al. (2022) mengimplementasikan ChaCha20-Poly1305 yang kompatibel dengan TLS 1.3 dan menunjukkan peningkatan kinerja 7 kali lipat dibandingkan implementasi perangkat lunak di lingkungan RISC-V [7], namun fokus penelitian tersebut terbatas pada komunikasi jaringan berbasis perangkat keras (FPGA) dan tidak mencakup pengamanan berkas dokumen multi-format. Susanti et al. (2024) membandingkan AES-GCM dengan ChaCha20-Poly1305 dan menemukan bahwa ChaCha20-Poly1305 lebih cepat pada data berukuran besar [8], namun penelitian tersebut tidak mengintegrasikan mekanisme pembangkitan kunci kriptografis berbasis fungsi hash (SHA-256) serta tidak mengevaluasi kekuatan enkripsi melalui uji Avalanche Effect maupun analisis statistik citra seperti MSE dan PSNR. Dengan demikian, terdapat kesenjangan penelitian berupa belum adanya implementasi terpadu ChaCha20-Poly1305 yang dikombinasikan dengan kunci SHA-256 secara khusus untuk pengamanan dokumen penting masyarakat di Indonesia, sekaligus dievaluasi secara detail mencakup aspek kekuatan kriptografis, integritas data, dan perbandingan performa algoritma.

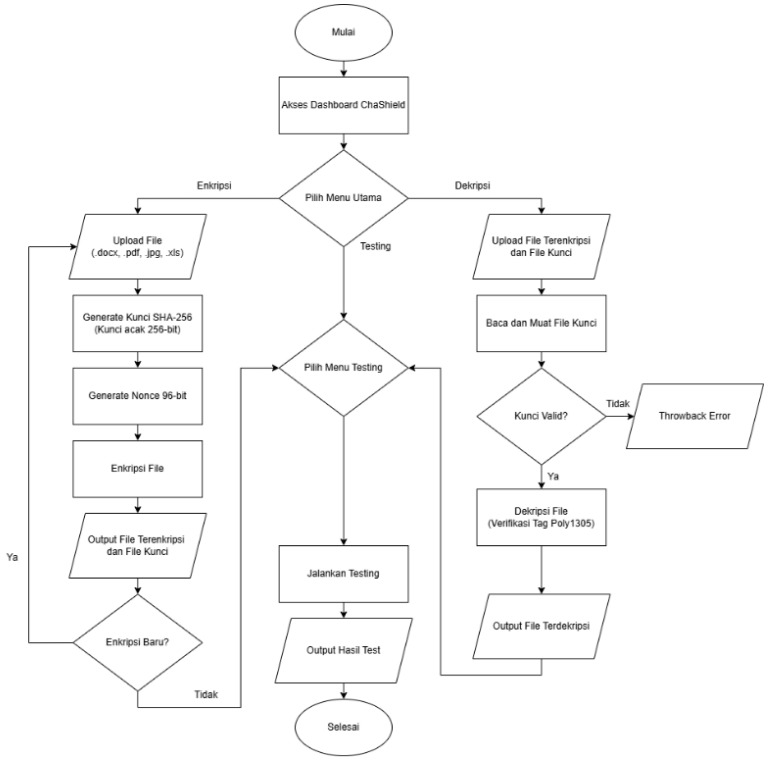
Berdasarkan kesenjangan tersebut, penelitian ini mengimplementasikan algoritma ChaCha20-Poly1305 dikombinasikan dengan kunci SHA-256 untuk mengamankan dokumen penting masyarakat dalam konteks Indonesia, sebagai upaya mengatasi keterbatasan penelitian terdahulu yang belum menghadirkan solusi berupa basis SHA-256 dengan evaluasi kriptografis menyeluruh pada berkas dokumen multi-format. Tujuan penelitian adalah untuk menganalisis langkah-langkah pengamanan data menggunakan ChaCha20-Poly1305 dengan kunci SHA-256, mengetahui persentase performa kekuatan algoritma melalui uji *Avalanche Effect* serta analisis statistik, dan mengetahui perbandingan performa waktu enkripsi dan dekripsi antara ChaCha20-Poly1305 dengan AES-256-GCM.

2. Metode Penelitian

2.1. Gambaran Umum Sistem

Sistem yang dikembangkan bernama ChaShield merupakan aplikasi berbasis *website* menggunakan bahasa pemrograman Python dengan *framework* Streamlit. Sistem terdiri dari dua kelompok menu utama, menu Algoritma yang mencakup fungsi enkripsi dan dekripsi file, serta menu Testing yang mencakup lima modul pengujian, yaitu *Visual Test, Avalanche Effect, Algorithm Comparison, Content Comparison,* dan MSE & PSNR Testing. Pemisahan antara menu operasional dan menu pengujian memungkinkan pengguna menggunakan sistem sesuai kebutuhannya. Berikut dibawah ini adalah diagram alir gambaran umum sistem.

p-ISSN: 2301-5373
e-ISSN: 2654-5101



Gambar 1. Diagram Alir Gambaran Umum Sistem

2.2. Pengumpulan Data

Data yang digunakan adalah data sekunder berupa data-data *dummy* yang mensimulasikan dokumen kemasyarakatan, seperti surat kepemilikan tanah, data BPJS, izin risiko perusahaan, dan dokumen lainnya. Jenis data yang digunakan meliputi dokumen teks (.docx, .pdf), dokumen gambar (.jpg, .png), dan dokumen spreadsheet (.xlsx, .xls). Detail representasi data dapat dilihat pada Tabel 1.

Tabel 1. Detail Representasi Data

No	Data	Format File	Ukuran File	Kuantitas
1	KTP	.png	±30 KB	50
2	Surat Tanah	.png	±400 KB	20
3	Data BPJS	.xls	±20 KB	40
4	Data Izin Risiko & Data Satgas	.docx	±80 KB	5
5	Pertemuan1_Politik	.pdf	±2 MB	15
	TOTAL DATA			130

2.3. Pembangkit Kunci (*Generate Key*)

Proses pembangkitan kunci menggunakan mekanisme *Cryptographically Secure Pseudo Random Generator* (CSPRNG). Sistem operasi mengumpulkan *raw entropy* sebesar 256-bit melalui fungsi os.urandom(32), yang kemudian diproses melalui fungsi *hash* SHA-256 untuk menghasilkan kunci kriptografis sepanjang 32 byte (256-bit). Kunci dengan ruang kemungkinan 2^{256} kombinasi ini secara komputasional mustahil untuk ditembus. Kunci yang dihasilkan kemudian divalidasi sebelum digunakan dalam proses enkripsi dan disimpan dalam file .txt yang dapat diunduh pengguna secara terpisah dari file terenkripsi.

2.4. Pembangkitan Nonce

Number Used Once (Nonce) dibangkitkan menggunakan os.urandom(12) yang menghasilkan 12 byte acak unik setara 96-bit, sesuai dengan panjang *nonce* standar pada ChaCha20-Poly1305. Sistem memeriksa apakah *nonce* tersebut pernah digunakan bersama kunci yang sama, karena penggunaan *nonce* yang sama dua kali dapat membahayakan keamanan enkripsi secara keseluruhan. Nonce kemudian dikombinasikan dengan kunci 256-bit dan konstanta tetap ChaCha20 untuk menginisialisasi *state* matriks 4×4

2.5. Proses Enkripsi

Proses enkripsi menggunakan algoritma ChaCha20-Poly1305 yang merupakan gabungan *stream cipher* ChaCha20 untuk enkripsi dan Poly1305 untuk autentikasi. State matriks awal ChaCha20 terdiri dari 16 kata 32-bit: 4 kata konstanta tetap (0x61707865, 0x3320646e, 0x79622d32, 0x6b206574), 8 kata kunci 256-bit, 1 kata *counter* dimulai dari nol, dan 3 kata *nonce* 96-bit. Proses komputasi menjalani 20 putaran yang terdiri dari 10 *column quarter rounds* dan 10 *diagonal quarter rounds*. Setelah 20 putaran, *working state* dijumlahkan dengan *initial state* menggunakan operasi mod 2^{32} untuk menghasilkan blok *keystream* 512-bit yang di-XOR-kan dengan *plaintext*. Secara matematis, proses enkripsi dirumuskan sebagai:

$$C = ChaCha20(K, counter, nonce, P) \,||\, Poly1305(K, C) \quad (1)$$

Poly1305 kemudian menghasilkan tag autentikasi 128-bit melalui evaluasi polinomial modulus $2^{130}-5$ untuk memastikan integritas dan keaslian data.

2.6. Proses Dekripsi

Proses dekripsi merupakan kebalikan dari enkripsi. Pengguna mengunggah file terenkripsi beserta file kunci (.txt). Sistem mengekstrak *nonce* 96-bit dari bagian awal data terenkripsi, lalu merekonstruksi kunci 256-bit. Sebelum melakukan dekripsi, sistem memverifikasi tag autentikasi Poly1305. Apabila tag tidak sesuai, proses ditolak karena data dianggap telah dimodifikasi atau kunci tidak valid. Formula dekripsi adalah:

$$P = ChaCha20(K, counter, nonce, C) \;\; IF\ TAG(Valid) \quad (2)$$

2.7. Metode Pengujian dan Evaluasi

Pengujian dilakukan melalui lima skenario evaluasi. Pertama, *Visual Testing* untuk menganalisis perubahan karakteristik citra meliputi nilai Shannon Entropy, koefisien korelasi piksel, dan distribusi histogram RGB sebelum dan sesudah enkripsi. Kedua, *Avalanche Effect Testing* dengan mengenkripsi file yang sama menggunakan dua kunci yang berbeda hanya satu bit, lalu mengukur persentase bit yang berbeda pada *ciphertext*. Formula perhitungannya adalah:

$$Avalanche\ Effect\ (\%) = (Jumlah\ Bit\ Berbeda\ /\ Total\ Bit\ Output) \times 100\% \quad (3)$$

Ketiga, *Algorithm Comparison* untuk membandingkan rata-rata waktu enkripsi dan dekripsi ChaCha20-Poly1305 dengan AES-256-GCM melalui 10 iterasi per file. Keempat, *Content Comparison* untuk memverifikasi kesesuaian konten antara file asli dan file hasil dekripsi dengan cara mengekstraksi dan membandingkan teks karakter per karakter. Kelima, *MSE & PSNR Testing* untuk mengukur perbedaan kuantitatif antara citra asli dan citra terenkripsi menggunakan formula:

$$MSE = (1/n)\ \Sigma(Yi - Y'i)^2 \quad (4)$$

Berikut dibawah ini merupakan formula untuk PSNR:

$$PSNR = 10\ log_{10}(MAX^2 / MSE) \quad (5)$$

p-ISSN: 2301-5373
e-ISSN: 2654-5101

2.8. Spesifikasi Sistem

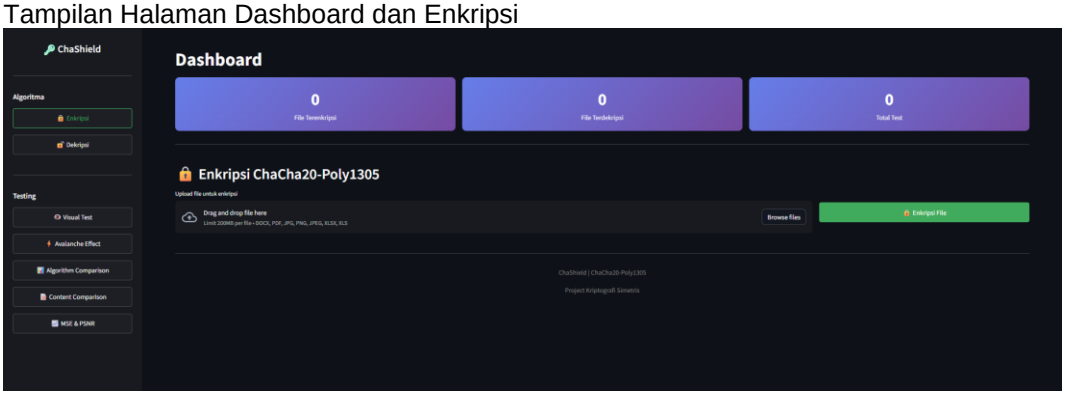
Sistem dikembangkan pada laptop Acer Swift SF-314-43 dengan prosesor AMD Ryzen 5 5500U, RAM 16 GB, penyimpanan 512 GB, dan GPU AMD Radeon Graphic. Perangkat lunak yang digunakan meliputi Windows 11, Visual Studio Code, Python, *framework* Streamlit, serta *library* Cryptography (hazmat), Pillow (PIL), NumPy, Plotly, Pandas, python-docx, PyPDF2, dan openpyxl.

3. Result and Discussion

3.1. Implementasi Sistem

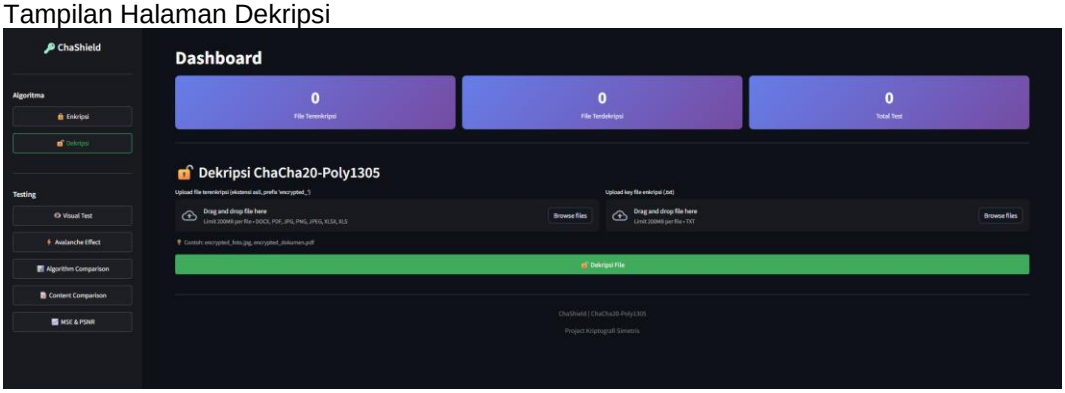
Sistem ChaShield berhasil diimplementasikan sebagai aplikasi berbasis *website* dengan antarmuka yang intuitif. Halaman enkripsi menyediakan area unggah file dengan validasi format otomatis dan menampilkan ringkasan hasil berupa ukuran file serta durasi proses. Sistem menghasilkan dua keluaran yang dapat diunduh terpisah, yaitu file terenkripsi dan file kunci berformat .txt. Halaman dekripsi menyediakan dua area unggah terpisah untuk file terenkripsi dan file kunci. Selain itu, sistem dilengkapi enam halaman pengujian yang menampilkan visualisasi hasil secara interaktif menggunakan grafik Plotly. Berikut dibawah ini merupakan gambar tampilan sistem.

a. Tampilan Halaman Dashboard dan Enkripsi



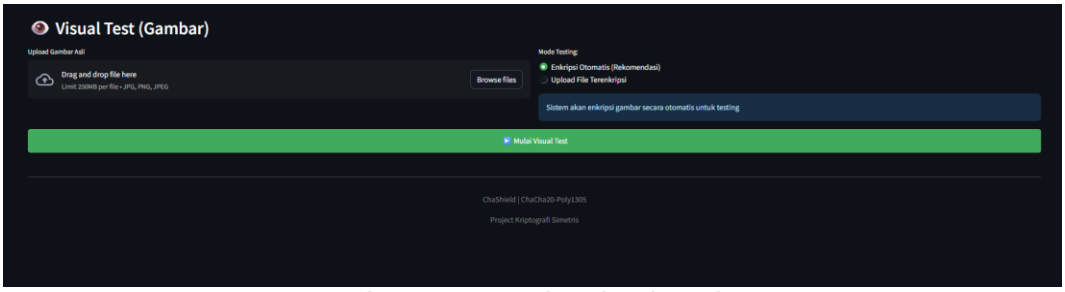
Gambar 2. Antarmuka *Dashboard* dan Enkripsi

b. Tampilan Halaman Dekripsi



Gambar 3. Antarmuka Dekripsi

c. Tampilan Halaman *Visual Testing*

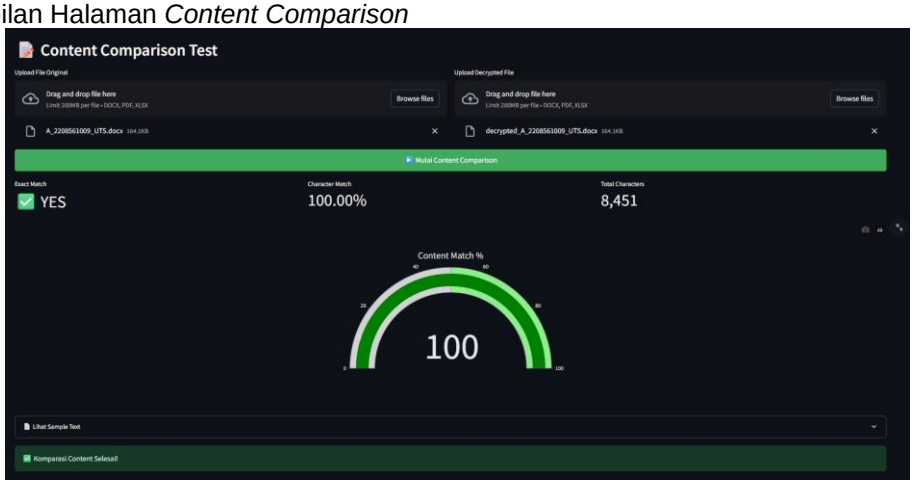


Gambar 4. Antarmuka *Visual Testing*

d. Tampilan Halaman *Algorithm Comparison*

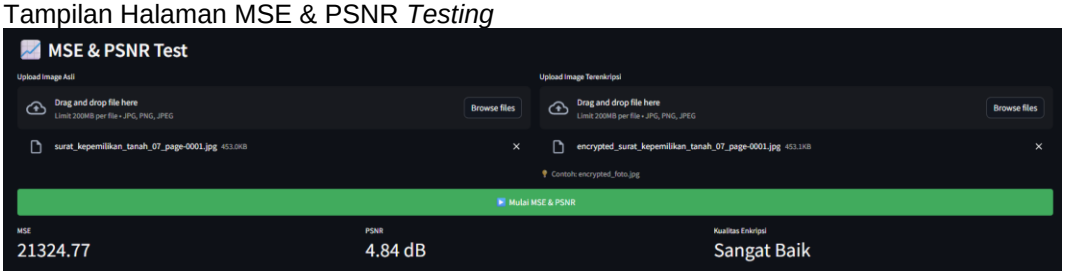
Gambar 5. Antarmuka *Algorithm Comparison*

e. Tampilan Halaman *Content Comparison*



Gambar 6. Antarmuka *Content Comparison*

f. Tampilan Halaman MSE & PSNR *Testing*



Gambar 7. Antarmuka MSE & PSNR *Testing*

p-ISSN: 2301-5373
e-ISSN: 2654-5101

3.2. Hasil Visual Testing

Pengujian visual dilakukan pada lima file gambar berekstensi .jpg (surat_kepemilikan_tanah) untuk mengevaluasi perubahan karakteristik citra sebelum dan sesudah enkripsi. Hasil pengujian ditampilkan pada Tabel 2.

Tabel 2. Hasil Uji Visual Testing

No	Nama File	Entropy Asli	Entropy Enkripsi	Selisih	Korelasi Asli	Korelasi Enkripsi
1	surat_tanah_01	0,7004	7,9988	+7,2984	0,8126	-0,0073
2	surat_tanah_02	0,7097	7,9996	+7,2899	0,7836	-0,0190
3	surat_tanah_03	0,7143	7,9988	+7,2844	0,7982	-0,0033
4	surat_tanah_04	0,7332	7,9996	+7,2664	0,7721	-0,0118
5	surat_tanah_05	0,7218	7,9996	+7,2778	0,7906	-0,0011
	Rata-rata	0,7159	7,9993	+7,2834	0,7914	-0,0085

Hasil pengujian menunjukkan peningkatan nilai entropy yang sangat signifikan setelah proses enkripsi. Rata-rata entropy dokumen asli berada pada angka 0,7159, yang kemudian melonjak menjadi rata-rata 7,9993 setelah dienkripsi, mendekati nilai ideal entropy sempurna sebesar 8,0. Peningkatan ini membuktikan bahwa data terenkripsi memiliki distribusi byte yang sangat acak dan seragam, sehingga mustahil dianalisis secara statistik. Dari sisi korelasi piksel, dokumen asli memiliki rata-rata 0,7914 yang mengindikasikan adanya hubungan linear kuat antar piksel berdekatan. Setelah enkripsi, nilai korelasi rata-rata turun drastis menjadi -0,0085, membuktikan bahwa tidak ada lagi hubungan keterkaitan antar piksel pada file output. Hasil analisis histogram RGB juga menunjukkan perubahan dari distribusi tidak merata menjadi distribusi yang sangat seragam, mengkonfirmasi tingkat keacakan tinggi pada data terenkripsi.

3.3. Hasil Avalanche Effect Testing

Pengujian *Avalanche Effect* dilakukan pada lima file dengan beragam format untuk menganalisis sensitivitas algoritma terhadap perubahan satu bit pada kunci enkripsi. Hasil pengujian ditampilkan pada Tabel 3.

Tabel 3. Hasil Uji Avalanche Effect

No	Nama File	Bit Berbeda	Total Bit	Avalanche (%)	Status
1	surat_tanah_05 (.png)	1.824.976	3.647.680	50,03%	Terpenuhi
2	DataSatgas1 (.docx)	99.227	198.304	50,04%	Terpenuhi
3	Pertemuan_7 (.pdf)	2.898.608	5.793.016	50,04%	Terpenuhi
4	DataBPJS1 (.xlsx)	79.887	159.720	50,02%	Terpenuhi
5	Ktp1_Dummy (.jpg)	1.117.354	2.232.104	50,06%	Terpenuhi
	Rata-rata			50,038%	Terpenuhi

Berdasarkan data pengujian, ChaCha20-Poly1305 menunjukkan performa yang sangat stabil dengan rata-rata nilai *Avalanche Effect* sebesar 50,038% pada seluruh format file yang diuji. Nilai ini mendekati angka ideal 50%, yang merupakan standar keamanan kriptografi, di mana perubahan satu bit saja pada kunci mampu mengubah rata-rata setengah dari total bit pada

ciphertext. Hal ini membuktikan bahwa algoritma memiliki sifat *non-linearity* yang sangat baik, sehingga sulit dieksploitasi melalui analisis diferensial maupun serangan kriptanalisis lainnya. Seluruh hasil pengujian pada semua format file memenuhi kriteria standar keamanan kriptografi.

3.4. Hasil Komparasi Algoritma

Pengujian *Avalanche Effect* dilakukan pada lima file dengan beragam format untuk menganalisis sensitivitas algoritma terhadap perubahan satu bit pada kunci enkripsi. Hasil pengujian ditampilkan pada Tabel 3.

Tabel 4. Hasil Uji Komparasi Algoritma

No	File	ChaCha Enkripsi (s)	AES Enkripsi (s)	ChaCha Dekripsi (s)	AES Dekripsi (s)
1	surat_tanah (.png)	0,000507	0,000830	0,000396	0,001058
2	DataSatgas (.docx)	0,000071	0,000327	0,000033	0,000253
3	Pertemuan_7 (.pdf)	0,000887	0,001581	0,000877	0,001709
4	DataBPJS (.xlsx)	0,000081	0,000310	0,000032	0,000280
5	Ktp1_Dummy (.jpg)	0,000249	0,000559	0,000211	0,000540
	Rata-rata	0,000359	0,000721	0,000309	0,000768

ChaCha20-Poly1305 menunjukkan performa yang lebih unggul dibandingkan AES-256-GCM pada seluruh skenario pengujian, baik dalam proses enkripsi maupun dekripsi. ChaCha20-Poly1305 mencatatkan rata-rata waktu enkripsi sebesar 0,000359 detik dan rata-rata waktu dekripsi sebesar 0,000309 detik, sementara AES-256-GCM memerlukan waktu lebih lama dengan rata-rata enkripsi 0,000721 detik dan dekripsi 0,000768 detik. Selain lebih cepat, nilai standar deviasi ChaCha20-Poly1305 secara konsisten lebih rendah pada hampir seluruh file uji, mengindikasikan kestabilan performa yang lebih baik. Keunggulan ini disebabkan karena ChaCha20-Poly1305 didesain berbasis operasi aritmetika sederhana (penjumlahan, rotasi, dan XOR) yang tidak memerlukan akselerasi perangkat keras khusus, berbeda dengan AES yang memerlukan instruksi AES-NI untuk mencapai performa optimal.

3.5. Hasil Komparasi Konten

Pengujian komparasi konten dilakukan untuk memverifikasi bahwa proses dekripsi mampu mengembalikan data ke kondisi semula tanpa perubahan isi. Pengujian dilakukan dengan mengekstraksi teks dari file original dan file hasil dekripsi, kemudian membandingkannya karakter per karakter. Hasil pengujian pada tiga file dengan format berbeda ditampilkan pada Tabel 5.

Tabel 5. Hasil Uji Komparasi Konten

No	File	Match (%)	Keterangan
1	DataSatgas1 (.docx)	100%	Valid
2	DataBPJS1_Dummy (.xlsx)	100%	Valid
3	Pertemuan7 (.pdf)	100%	Valid

p-ISSN: 2301-5373
e-ISSN: 2654-5101

Berdasarkan pengujian pada ketiga file dengan format berbeda, diperoleh tingkat kesamaan 100% antara file original dan file hasil dekripsi pada seluruh format yang diuji. Hasil ini membuktikan bahwa ChaCha20-Poly1305 bersifat *lossless*, yakni mampu mengembalikan data ke kondisi semula secara sempurna tanpa kehilangan informasi sedikit pun. Mekanisme tag autentikasi Poly1305 juga berperan penting dalam memastikan hanya data yang valid dan tidak dimodifikasi yang dapat berhasil didekripsi, sehingga integritas data terjamin sepenuhnya.

3.6. Hasil MSE dan PSNR Testing

Pengujian MSE dan PSNR dilakukan untuk mengukur secara kuantitatif tingkat perbedaan antara citra asli dan citra hasil enkripsi. Lima sampel gambar berekstensi .jpg digunakan sebagai data uji. Hasil pengujian ditampilkan pada Tabel 6.

Tabel 6. Hasil Uji MSE dan PSNR

No	File	MSE	PSNR (dB)
1	surat_tanah_05 (.jpg)	21.316,60	4,84
2	Ktp1_Dummy (.jpg)	9.277,22	8,46
3	surat_tanah_01 (.jpg)	21.332,12	4,84
4	surat_tanah_02 (.jpg)	21.288,57	4,85
5	Ktp2_Dummy (.jpg)	10.362,06	7,98
	Rata-rata	16.715,31	6,19

Nilai MSE yang sangat tinggi dan nilai PSNR yang sangat rendah menunjukkan bahwa proses enkripsi telah berhasil menghancurkan informasi visual secara total. Rata-rata MSE yang dihasilkan adalah 16.715,31, dengan nilai tertinggi mencapai 21.332,12 pada file surat_kepemilikan_tanah_01. Nilai MSE yang besar mengindikasikan selisih nilai piksel yang sangat ekstrem antara citra asli dan citra terenkripsi. Rata-rata PSNR hanya sebesar 6,19 dB, jauh di bawah ambang batas 30 dB yang secara teori sudah menunjukkan perubahan signifikan pada citra. Nilai PSNR di bawah 10 dB mengkonfirmasi bahwa data visual telah terdistorsi sepenuhnya, yang merupakan indikator keberhasilan enkripsi dalam menyembunyikan informasi visual. Nilai MSE yang tinggi pada saat enkripsi namun kembali ke nol saat dekripsi (100% *match*) juga mengkonfirmasi sifat *lossless* dari algoritma ini.

4. Kesimpulan

Berdasarkan hasil penelitian implementasi algoritma ChaCha20-Poly1305 dengan kunci berbasis SHA-256 untuk pengamanan dokumen penting masyarakat, diperoleh tiga kesimpulan utama berikut. Pertama, sistem ChaShield yang dibangun mampu melakukan proses enkripsi dan dekripsi menggunakan algoritma ChaCha20-Poly1305 dengan kunci yang dihasilkan dari hashing SHA-256. Pengujian komparasi konten pada seluruh format file yang diuji (.docx, .xlsx, .pdf) menunjukkan tingkat kesamaan 100% antara file asli dan file hasil dekripsi, membuktikan sifat *lossless* dan keandalan mekanisme autentikasi Poly1305 dalam menjaga integritas data. Kedua, algoritma ChaCha20-Poly1305 terbukti memiliki kekuatan kriptografis yang tinggi. Nilai *Avalanche Effect* rata-rata 50,038% pada seluruh format file membuktikan sensitivitas tinggi terhadap perubahan kunci. Secara statistik, enkripsi meningkatkan entropy citra dari rata-rata 0,7159 menjadi 7,9993, menurunkan korelasi piksel dari 0,7914 menjadi -0,0085, serta menghasilkan rata-rata MSE 16.715,31 dan PSNR 6,19 dB. Hasil-hasil ini secara kolektif membuktikan bahwa data terenkripsi memiliki tingkat keacakan yang sangat tinggi tanpa kemiripan visual dengan data aslinya. Ketiga, ChaCha20-Poly1305 menunjukkan performa lebih unggul dibandingkan AES-256-GCM dalam seluruh skenario pengujian. Rata-rata waktu enkripsi ChaCha20-Poly1305 sebesar 0,000359 detik dan dekripsi 0,000309 detik, lebih cepat dibandingkan AES-256-GCM dengan enkripsi 0,000721 detik dan dekripsi 0,000768 detik. Nilai

standar deviasi yang lebih rendah juga mengindikasikan kestabilan performa ChaCha20-Poly1305 yang lebih baik pada berbagai jenis dan ukuran file. Untuk pengembangan ke depan, disarankan untuk menambahkan fitur monitoring *real-time* pada proses enkripsi dan dekripsi, memperluas parameter evaluasi dengan pengukuran penggunaan memori dan konsumsi CPU, serta menguji sistem menggunakan dataset yang lebih besar dan beragam agar menghasilkan evaluasi yang lebih representatif terhadap kondisi penggunaan nyata.

Daftar Pustaka

[1] A. Susanti, B. A. Prasetiya, O. D. Pangesti, L. D. Suryawati, dan I. A. Saputro, "Perbandingan Kinerja dan Keamanan Algoritma Kriptografi Modern AES-GCM dengan CHACHA20-POLY1305," *Infomatek*, vol. 26, no. 2, hlm. 253–264, 2024.

[2] G. D. M. Zulma, H. B. Seta, dan T. Yuniati, "Implementasi Algoritma AES dan Bcrypt untuk Pengamanan File Dokumen," *Informatik: Jurnal Ilmu Komputer*, vol. 18, no. 2, hlm. 163, 2022.

[3] J. Kaur dan K. R. R. Kumar, "Analysis of Avalanche Effect in Cryptographic Algorithms," dalam *2022 10th International Conference on Reliability, Infocom Technologies and Optimization (ICRITO)*, 2022, hlm. 1–4.

[4] K. I. Masud, M. R. Hasan, Md. M. Hoque, U. D. Nath, dan Md. O. Rahman, "A New Approach of Cryptography for Data Encryption and Decryption," dalam *2022 5th International Conference on Computing and Informatics (ICCI)*, 2022, hlm. 234–239.

[5] K. Mohamed, M. N. M. Pauzi, F. H. H. M. Ali, dan S. Ariffin, "Analyse On Avalanche Effect In Cryptography Algorithm," dalam *Proc. Int. Conf.*, 2022, hlm. 610–618.

[6] Kepaniteraan Mahkamah Agung RI, "Putusan Mahkamah Agung," 2024. [Online]. Tersedia: https://putusan3.mahkamahagung.go.id. [Diakses: 15 Mar. 2026].

[7] M. Anindita, "Analisis Perbandingan Algoritma AES dan ChaCha20-Poly1305 dalam Enkripsi Konten Livestreaming," *Journal of Information Systems and Technology*, vol. 1, no. 1, hlm. 1–6, 2023.

[8] M. Azhari, D. I. Mulyana, F. J. Perwitosari, dan F. Ali, "Implementasi Pengamanan Data pada Dokumen Menggunakan Algoritma Kriptografi Advanced Encryption Standard (AES)," *Jurnal Pendidikan Sains Dan Komputer*, vol. 2, no. 01, hlm. 163–171, 2022.

[9] R. Serrano, C. Duran, M. Sarmiento, C.-K. Pham, dan T.-T. Hoang, "ChaCha20-Poly1305 Authenticated Encryption with Additional Data for Transport Layer Security 1.3," *Cryptography*, vol. 6, no. 2, hlm. 30, 2022.

[10] S. Jamil, "Review of Image Quality Assessment Methods for Compressed Images," *Journal of Imaging*, vol. 10, no. 5, 2024.

[11] S. Mashabi dan A. Kasih, "PDN Diretas, Bagaimana Nasib Data Penerima Beasiswa Di Kemendikbud?" *Kompas.com*, 30 Jun. 2024.

[12] T. Beyne, Y. L. Chen, dan M. Verbauwhede, "A Robust Variant of ChaCha20-Poly1305," *Journal of Cryptology*, vol. 1, no. 1, 2025.

[13] W. Nugroho, A. Susanto, C. Sari, E. Rachmawanto, dan M. Doheir, "A Robust and Imperceptible for Digital Image Encryption Using ChaCha20," *Jurnal Teknik Informatika (JUTIF)*, vol. 5, no. 2, hlm. 397–404, 2024.