



Keamanan Data Berbasis Representasi Dimensi Partisi Graf Siklus

Amrullah, Junaidi, Ratna Yulis Tyaningsih, Nilza Humaira
Salsabila, M. Gunawan Supiarmo, Meiliza E.R.

Program Studi Pendidikan Matematika, FKIP, Universitas Mataram, Mataram

amrullah@unram.ac.id

Abstract

This study develops and evaluates the PDE-SEC (Partition Dimension--Security) prototype for securing .tex files using vertex representation codes on the cycle graph C₂₅₆. Each byte value is mapped to a graph vertex and represented using a three-class resolving partition. The representation codes are incorporated into a key-dependent permutation, followed by a SHA-256-based keystream transformation and HMAC-SHA-256 for integrity verification. Experiments were conducted on 15 .tex files using correctness, file size, overhead, encryption and decryption time, throughput, entropy, correlation, avalanche effect, and key sensitivity. All 15 files were successfully recovered. The cumulative original size of 481,582 bytes became 483,502 bytes after encryption, corresponding to 0.3987% aggregate overhead. Total encryption and decryption times were 0.457746 s and 0.752114 s, with aggregate throughputs of 1.0521 MB/s and 0.6403 MB/s, respectively. Mean ciphertext entropy was 7.9815 bit/byte, mean absolute correlation was 0.00739, and recorded key sensitivity was 50.0238%. The initial avalanche result cannot yet be treated as a final measure because the paired tests used different nonces. Overall, PDE-SEC successfully transformed and recovered the tested files with low size overhead and ciphertext statistics close to a uniform distribution; controlled avalanche testing with the same nonce is still required before stronger security conclusions can be drawn.

Keywords: partition dimension; vertex representation code; cycle graph; data security; PDE-SEC; encryption.

Abstrak

Penelitian ini mengembangkan dan mengevaluasi prototipe PDE-SEC (Partition Dimension--Security) untuk pengamanan berkas .tex menggunakan kode representasi simpul pada graf siklus C₂₅₆. Setiap nilai byte dipandang sebagai simpul graf dan direpresentasikan berdasarkan partisi pembeda tiga kelas. Kode representasi digunakan dalam pembentukan permutasi berbasis kunci, kemudian data ditransformasikan menggunakan keystream SHA-256 dan dilengkapi HMAC-SHA-256 untuk pemeriksaan integritas. Eksperimen dilakukan terhadap 15 berkas .tex dengan mengevaluasi ketepatan pemulihan, ukuran, overhead, waktu pemrosesan, throughput, entropi, korelasi, efek avalanche, dan sensitivitas kunci. Seluruh berkas berhasil didekripsi dengan benar. Ukuran kumulatif berkas asli sebesar 481.582 byte menjadi 483.502 byte setelah enkripsi, dengan overhead agregat 0,3987%. Waktu enkripsi dan dekripsi masing-masing 0,457746 detik dan 0,752114 detik, dengan throughput 1,0521 MB/s dan 0,6403 MB/s. Entropi ciphertext rata-rata mencapai 7,9815 bit/byte, korelasi absolut rata-rata 0,00739, dan sensitivitas kunci 50,0238%. Nilai avalanche awal belum dapat dijadikan ukuran final karena pengujian menggunakan nonce berbeda. Hasil menunjukkan transformasi dan pemulihan berkas berhasil dengan overhead rendah dan karakteristik statistik ciphertext mendekati distribusi seragam, tetapi pengujian avalanche terkontrol dengan nonce yang sama masih diperlukan.

Kata Kunci: dimensi partisi; kode representasi simpul; graf siklus; keamanan data; PDE-SEC; enkripsi.

1. PENDAHULUAN

Keamanan data digital merupakan kebutuhan penting dalam penyimpanan dan pertukaran informasi. Dokumen teks, termasuk dokumen yang ditulis menggunakan Editor Teks lain, dapat memuat informasi akademik, rancangan penelitian, data administratif, dan dokumen ilmiah yang perlu dilindungi dari akses tidak sah dan perubahan data. Kriptografi menyediakan mekanisme untuk mentransformasikan plaintext menjadi ciphertext menggunakan parameter rahasia sehingga isi data tidak dapat dipahami secara langsung tanpa proses dekripsi yang sesuai.

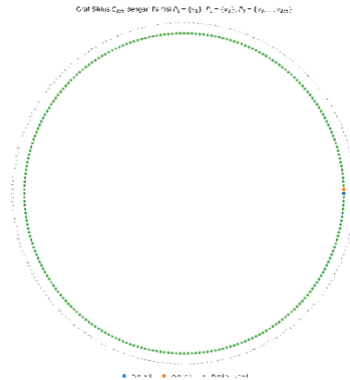
Di sisi lain, teori graf menyediakan konsep representasi yang dapat membedakan simpul berdasarkan jaraknya terhadap sejumlah himpunan. Untuk graf $G=(V,E)$ dan partisi $\Pi=\{P_1,P_2,\dots,P_k\}$, representasi simpul v didefinisikan sebagai $r(v|\Pi)=(d(v,P_1),\dots,d(v,P_k))$. Partisi disebut resolving apabila setiap dua simpul yang berbeda mempunyai representasi yang berbeda. Dimensi partisi adalah jumlah minimum kelas pada resolving partition (Chartrand et al., 2000). Kajian mutakhir menunjukkan bahwa dimensi partisi masih aktif dikaji pada berbagai keluarga dan operasi graf, termasuk graf siklus, amalgamasi, Toeplitz, serta keluarga graf dengan dimensi partisi tiga (Hasmawati et al., 2022; Luo et al., 2022; Haryeni et al., 2024; Koam et al., 2024).

Perkembangan tersebut juga menunjukkan bahwa struktur graf dapat digunakan sebagai dasar konstruksi metode keamanan data. Ni et al. (2021) mengembangkan beberapa skema enkripsi berbasis graf, sedangkan Ali et al. (2024) membahas pendekatan enkripsi menggunakan graf korona, graf bintang, dan graf bipartit. Hasil-hasil tersebut menunjukkan adanya ruang untuk menghubungkan struktur graf dengan transformasi data kriptografis. Namun, pendekatan yang menggunakan secara eksplisit kode representasi simpul dari resolving partition sebagai komponen pembentukan permutasi byte untuk berkas teks masih memerlukan kajian eksperimental.

Penelitian ini mengisi celah tersebut dengan memanfaatkan graf siklus C_{256} , yang secara alami mempunyai 256 simpul sehingga sesuai dengan alfabet byte 0--255. Resolving partition yang digunakan adalah $\Pi=\{P_1,P_2,P_3\}$, dengan $P_1=\{v_1\}$, $P_2=\{v_2\}$, dan $P_3=\{v_3,\dots,v_{256}\}$. Dalam penelitian ini, kode representasi tidak disimpan sebagai ciphertext dan tidak diposisikan sebagai metode kompresi. Kode tersebut digunakan sebagai komponen struktural untuk membentuk permutasi yang bergantung pada kunci. Selanjutnya, hasil substitusi ditransformasikan menggunakan keystream berbasis SHA-256 dan dilengkapi HMAC-SHA-256 untuk integritas. Pendekatan ini memisahkan fungsi representasi graf dari fungsi autentikasi dan transformasi bit.

Jarak antara dua simpul u dan v pada graf G , dinotasikan $d(u,v)$, adalah panjang lintasan terpendek yang menghubungkan keduanya. Untuk himpunan $S\subseteq V(G)$, jarak simpul v terhadap S didefinisikan sebagai $d(v,S)=\min\{d(v,x):x\in S\}$. Jika $\Pi=\{P_1,\dots,P_k\}$ merupakan partisi $V(G)$, maka $r(v|\Pi)=(d(v,P_1),\dots,d(v,P_k))$. Jika seluruh representasi berbeda, Π merupakan resolving partition dan $pd(G)$ adalah kardinalitas minimum dari resolving partition. Kajian terbaru pada amalgamasi siklus dan keluarga graf lainnya memperlihatkan bahwa struktur resolving partition dapat dikonstruksi secara eksplisit untuk memperoleh batas atau nilai dimensi partisi (Hasmawati et al., 2022; Koam et al., 2024).

Graf siklus C_n terdiri atas n simpul yang membentuk satu lingkaran. Untuk simpul yang diindeks a dan b , jaraknya memenuhi $d_{C_n}(a,b) = \min\{|a-b|, n-|a-b|\}$. Dengan memilih $n=256$, setiap kemungkinan nilai byte dapat dipetakan satu-ke-satu ke simpul graf. Konstruksi tiga kelas yang digunakan dalam penelitian ini adalah $P1=\{v1\}$, $P2=\{v2\}$, dan $P3=\{v3,...,v256\}$. Representasi contoh adalah $r(v1 | \Pi)=(0,1,1)$, $r(v2 | \Pi)=(1,0,1)$, dan $r(v3 | \Pi)=(2,1,0)$.



Gambar 1. Skema partisi pembeda tiga kelas pada graf siklus C_{256} .

Dalam PDE-SEC, kunci pengguna K terlebih dahulu diubah menjadi $K_h = \text{SHA256}(K)$. Kode representasi simpul kemudian digunakan bersama kunci untuk membentuk permutasi PK. Untuk byte plaintext M_i , hasil substitusi ditulis $X_i = \text{PK}(M_i)$. Sistem selanjutnya membangkitkan nonce acak N sepanjang 16 byte dan keystream $S_j = \text{SHA256}(K_h || N || \text{counter}_j)$. Ciphertext diperoleh melalui $C_i = X_i \text{ XOR } S_i$. Setelah ciphertext terbentuk, tag integritas dihitung dengan $\text{HMAC-SHA256}(C, K_h || N)$. Struktur ini memisahkan transformasi data, pembangkitan keystream, dan pemeriksaan integritas.

Dalam literatur, entropi, korelasi, dan avalanche sering digunakan sebagai indikator statistik untuk menilai perilaku ciphertext, walaupun metrik tersebut tidak cukup untuk membuktikan keamanan formal. Studi enkripsi multimedia mutakhir, misalnya, menggunakan entropi dan korelasi bersama pengujian avalanche dan analisis kriptografis lainnya (Ain Shams Engineering Journal, 2024; Journal of Cybersecurity and Privacy, 2024). Prinsip evaluasi yang sama digunakan dalam penelitian ini dengan penyesuaian pada data berupa berkas teks.

Tujuan penelitian adalah merancang dan mengevaluasi prototipe PDE-SEC untuk pengamanan berkas .tex. Evaluasi mencakup correctness, ukuran dan overhead, waktu komputasi, throughput, entropi, korelasi, sensitivitas kunci, dan avalanche effect. Hasil evaluasi digunakan untuk mengidentifikasi karakteristik yang mendukung rancangan serta keterbatasan metodologis yang masih perlu diperbaiki.

2. METODE PELAKSANAAN

3.1 Jenis dan Desain Penelitian

Penelitian ini merupakan penelitian pustaka dan eksperimental-komputasional. Kajian pustaka digunakan untuk mendasari konsep graf, dimensi partisi, resolving partition, representasi simpul, hash, HMAC, dan metrik evaluasi keamanan. Eksperimen digunakan

untuk mengevaluasi implementasi PDE-SEC terhadap 15 berkas .tex dengan ukuran dan karakteristik isi yang bervariasi.

3.2 Rancangan PDE-SEC

Setiap nilai byte 0-255 dipetakan ke satu simpul C256. Resolving partition tiga kelas digunakan untuk menghasilkan kode representasi setiap simpul. Kode tersebut tidak dituliskan ke dalam ciphertext, tetapi menjadi bagian dari input pembentukan permutasi berbasis kunci. Tahapan enkripsi diringkas sebagai berikut:

$$M \rightarrow r(v|\Pi) \rightarrow PK \rightarrow X \rightarrow N \rightarrow S \rightarrow C \rightarrow T$$

M menyatakan plaintext; $r(v|\Pi)$ adalah kode representasi; PK adalah permutasi berbasis kunci; X adalah hasil substitusi; N adalah nonce; S adalah keystream SHA-256; C adalah ciphertext; dan T adalah tag HMAC-SHA-256. Pada dekripsi, HMAC diverifikasi terlebih dahulu, kemudian keystream dan invers permutasi digunakan untuk memperoleh kembali plaintext.

3.3 Data dan Implementasi

Data penelitian terdiri atas 15 berkas .tex. Implementasi dilakukan menggunakan PHP. Program menyediakan fungsi enkripsi, dekripsi, serta pengujian keamanan dan kinerja. Header format .pde-sec berukuran tetap 128 byte dan memuat informasi format, versi, parameter graf, ukuran asli, nonce, serta tag integritas.

3.4 Metrik Evaluasi

Correctness diuji dengan membandingkan SHA-256 plaintext asli dan SHA-256 file hasil dekripsi. Overhead dihitung sebagai selisih ukuran file terenkripsi terhadap ukuran asli. Throughput dihitung dari ukuran plaintext dibagi waktu pemrosesan. Entropi Shannon dihitung pada alfabet byte dengan maksimum teoritis 8 bit/byte. Korelasi dihitung menggunakan koefisien Pearson. Sensitivitas kunci dihitung dari Hamming distance dua ciphertext yang dienkripsi dengan kunci berbeda. Avalanche effect dihitung dari persentase bit ciphertext yang berbeda ketika satu bit plaintext diubah.

Untuk pengujian avalanche yang valid, nonce harus dikendalikan sama pada dua enkripsi sehingga hanya plaintext yang berubah. Prinsip yang sama berlaku untuk uji sensitivitas kunci: plaintext dan nonce dipertahankan, sedangkan kunci diubah. Ketentuan ini penting karena nonce acak merupakan input langsung ke pembangkitan keystream.

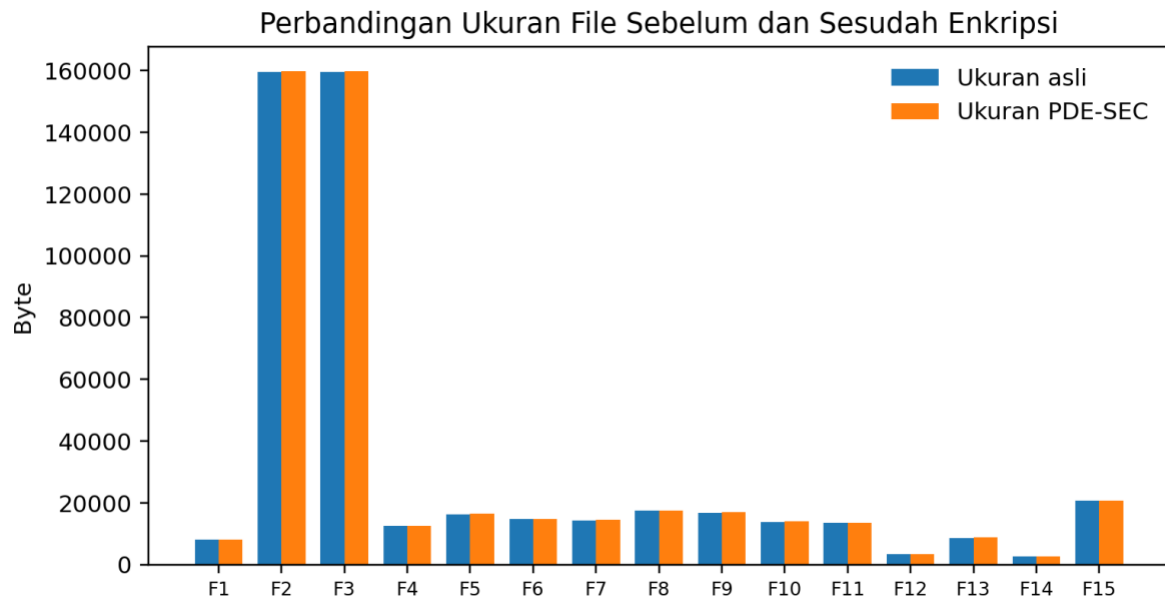
3. HASIL DAN PEMBAHASAN

4.1 Correctness dan Ukuran File

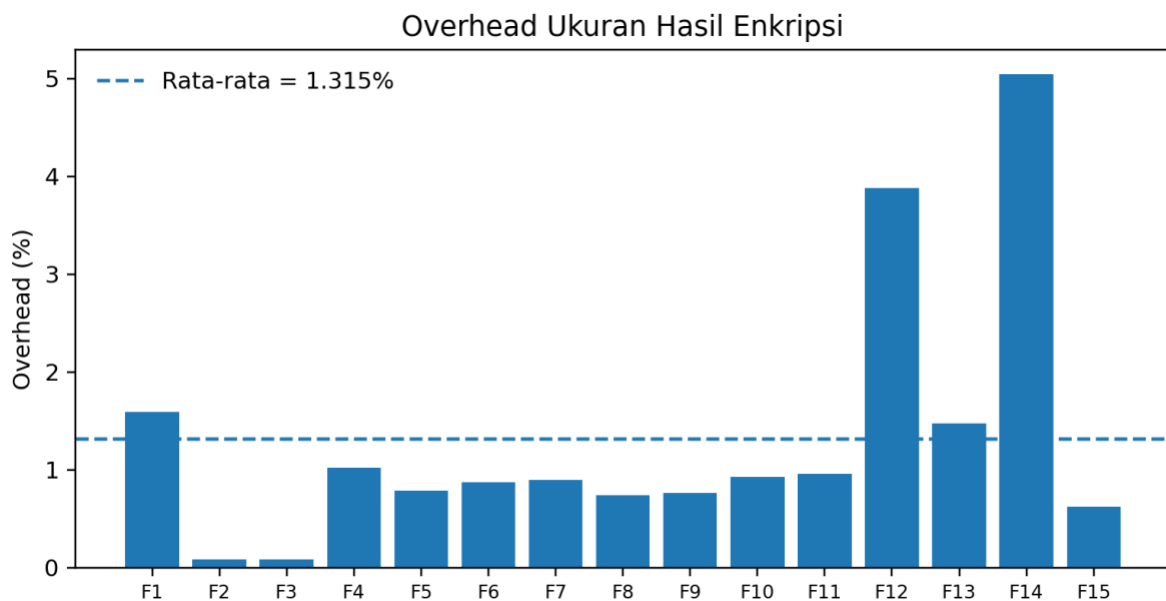
Seluruh 15 berkas berhasil dienkripsi dan didekripsi dengan benar. Nilai correctness adalah 15/15 atau 100%. Hasil tersebut menunjukkan bahwa, pada data uji, proses invers permutasi, pembangkitan kembali keystream, dan verifikasi integritas dapat mengembalikan file ke bentuk semula. Correctness merupakan indikator keberhasilan fungsional dan tidak dapat dipersamakan dengan bukti keamanan terhadap penyerang.

Ukuran kumulatif plaintext sebesar 481.582 byte menjadi 483.502 byte setelah enkripsi. Selisih 1.920 byte menghasilkan overhead agregat 0,3987%. Rata-rata overhead aritmetika

per file adalah 1,3150%. Perbedaan kedua nilai terjadi karena overhead tetap 128 byte per file mempunyai proporsi lebih besar pada file kecil. Misalnya, file sekitar 159 KB hanya memiliki overhead sekitar 0,08%, sedangkan file sekitar 2,5 KB mencapai sekitar 5,04%.



Gambar 2. Perbandingan ukuran plaintext dan berkas PDE-SEC.



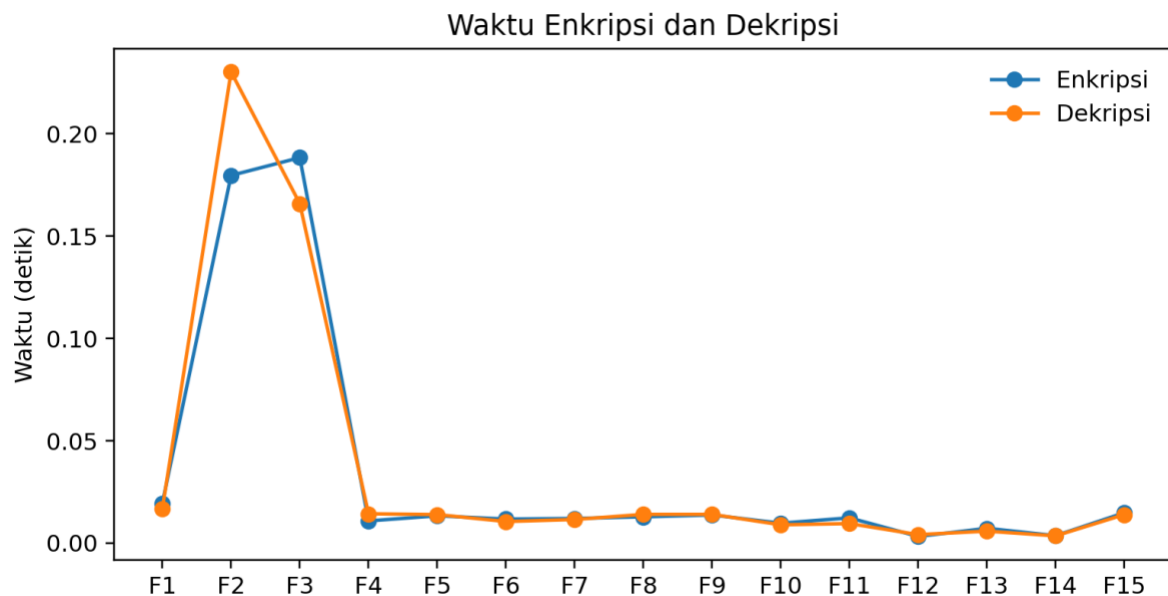
Gambar 3. Overhead ukuran akibat format PDE-SEC.

Pola ini merupakan konsekuensi langsung dari desain format kontainer. Overhead bukan menunjukkan kegagalan enkripsi atau kompresi, tetapi biaya penyimpanan metadata yang diperlukan untuk dekripsi dan autentikasi. Dalam sistem enkripsi yang membawa nonce/IV dan tag autentikasi, tambahan ukuran memang merupakan bagian dari desain. Karena PDE-

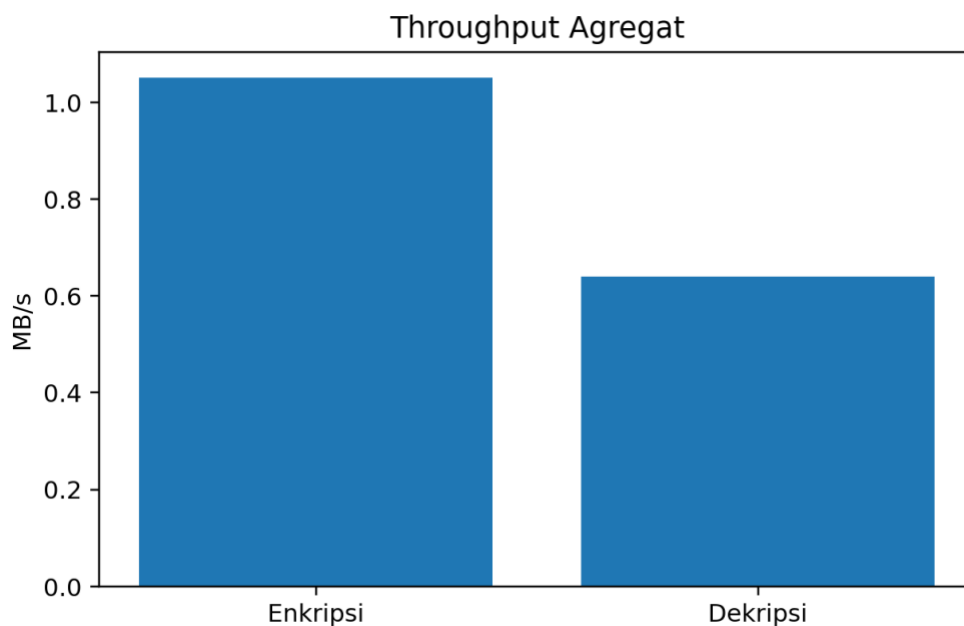
SEC tidak bertujuan melakukan kompresi, kenaikan ukuran merupakan konsekuensi yang dapat diprediksi.

4.2 Waktu Pemrosesan dan Throughput

Total waktu enkripsi adalah 0,457746 detik, sedangkan dekripsi 0,752114 detik. Throughput agregat mencapai 1,0521 MB/s untuk enkripsi dan 0,6403 MB/s untuk dekripsi. Korelasi antara ukuran plaintext dan waktu enkripsi adalah 0,9516, sedangkan korelasi antara ukuran plaintext dan waktu dekripsi adalah 0,9947. Kedua nilai menunjukkan bahwa waktu pemrosesan meningkat seiring bertambahnya jumlah byte.



Gambar 4. Waktu enkripsi dan dekripsi setiap berkas.

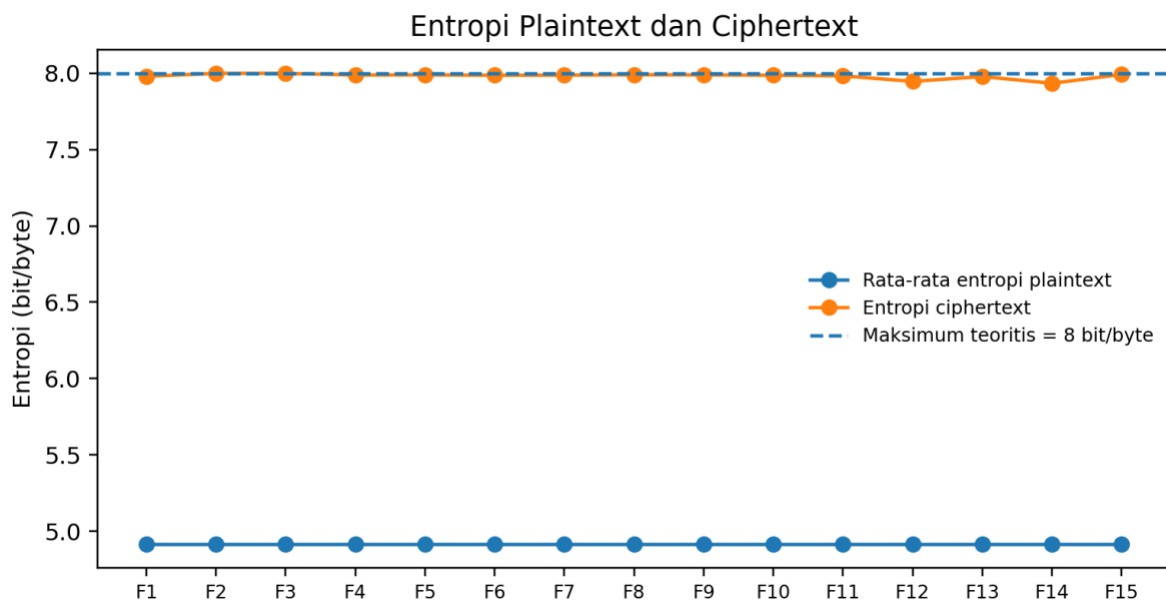


Gambar 5. Throughput agregat proses enkripsi dan dekripsi.

Hubungan tersebut dapat dipahami melalui bentuk sederhana $T(N) \approx T_0 + cN$, dengan N sebagai jumlah byte, T_0 sebagai biaya tetap inisialisasi, dan cN sebagai biaya yang meningkat mengikuti data. Dekripsi sedikit lebih mahal pada eksperimen ini karena mencakup verifikasi HMAC, pembacaan metadata, pembentukan kembali keystream, dan invers permutasi. Namun, karena hanya 15 file yang diuji dan waktu absolut sangat kecil, hasil throughput harus dipandang sebagai throughput pada lingkungan eksperimen, bukan karakteristik universal.

4.3 Entropi Ciphertext

Rata-rata entropi plaintext adalah 4,9164 bit/byte, sedangkan rata-rata entropi ciphertext mencapai 7,9815 bit/byte dengan rentang 7,9338-7,9989 bit/byte. Nilai tersebut sangat dekat dengan maksimum teoritis 8 bit/byte. Kenaikan entropi menunjukkan bahwa transformasi PDE-SEC membuat distribusi byte ciphertext jauh lebih mendekati seragam dibandingkan plaintext.



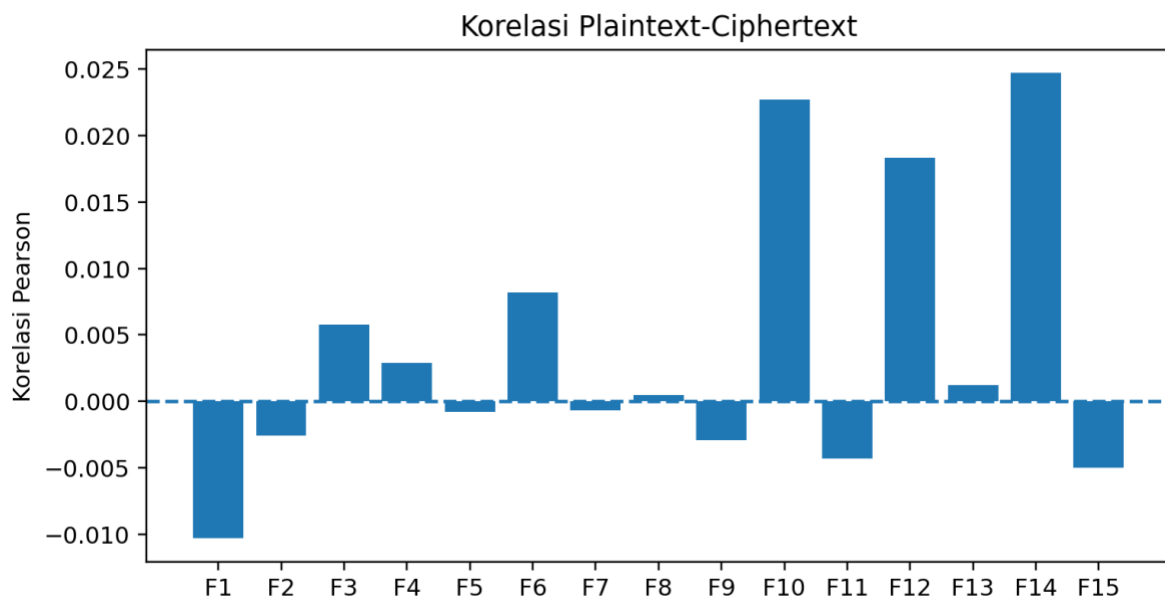
Gambar 6. Perbandingan entropi plaintext dan ciphertext.

Hasil ini sejalan dengan penelitian enkripsi multimedia yang menggunakan entropi sebagai indikator statistik dan melaporkan nilai ciphertext mendekati 8 bit/byte. Namun, kesamaan indikator tidak berarti kesamaan tingkat keamanan. Entropi hanya menggambarkan distribusi frekuensi simbol dan tidak menguji seluruh kemungkinan serangan. Kajian evaluasi enkripsi juga menekankan perlunya kombinasi metrik dan analisis kriptografis yang lebih mendalam.

4.4 Korelasi Plaintext-Ciphertext

Korelasi Pearson berada pada rentang -0,010283 sampai 0,024671, dengan rata-rata 0,003841 dan rata-rata nilai absolut 0,007391. Nilai yang sangat dekat dengan nol

menunjukkan bahwa hubungan linear byte-ke-byte antara plaintext dan ciphertext rendah pada data uji.



Gambar 7. Korelasi Pearson antara plaintext dan ciphertext.

Secara algoritmik, hasil tersebut dapat dijelaskan oleh dua lapisan transformasi. Pertama, permutasi berbasis kunci mengubah pemetaan byte. Kedua, XOR dengan keystream SHA-256 yang bergantung pada hash kunci, nonce, dan counter mengurangi pola statistik yang masih tersisa. Oleh karena itu, karakteristik ciphertext merupakan hasil gabungan kode representasi graf dan lapisan transformasi kriptografis berikutnya, bukan akibat dimensi partisi semata. Hasil korelasi mendekati nol konsisten dengan literatur enkripsi yang menggunakan rendahnya korelasi sebagai indikator statistik ciphertext, tetapi bukan bukti keamanan formal.

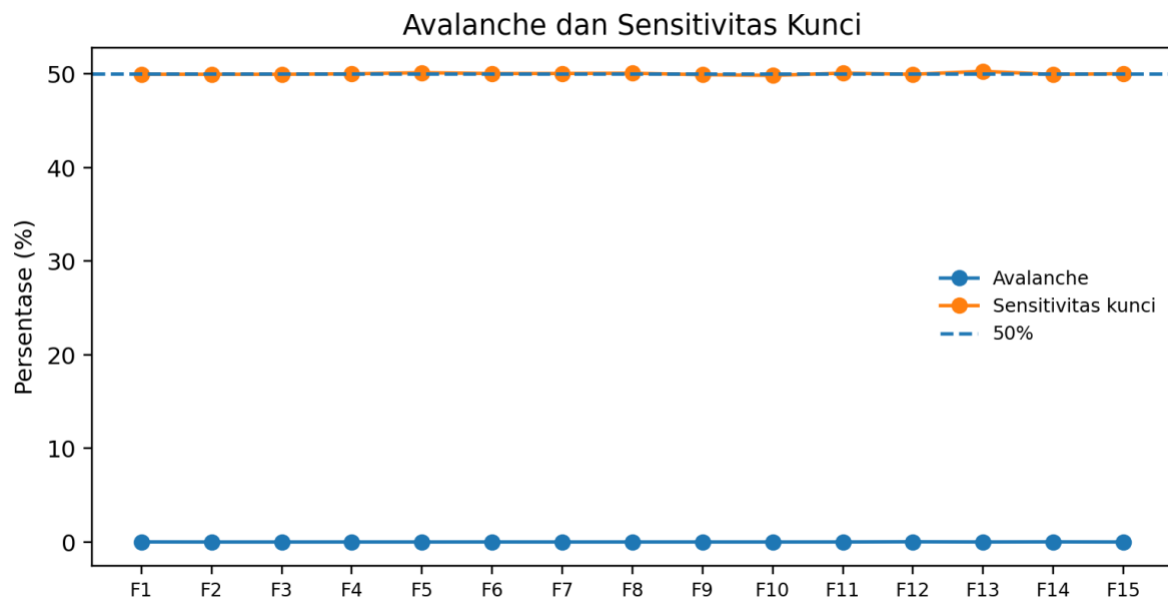
4.5 Sensitivitas Kunci dan Avalanche Effect

Rata-rata sensitivitas kunci yang tercatat adalah 50,0238%, dengan rentang 49,8538%--50,2822%. Secara statistik, nilai yang mendekati 50% menunjukkan bahwa dua ciphertext yang dibandingkan mempunyai perbedaan bit yang luas. Literatur evaluasi kriptografi juga menggunakan pengujian perubahan kunci sebagai salah satu cara mengevaluasi sensitivitas sistem.

Namun, interpretasi hasil tersebut harus dibatasi. Pada eksperimen awal, ciphertext pembandingan dihasilkan melalui pemanggilan enkripsi yang masing-masing membangkitkan nonce acak baru. Karena itu, perbedaan $C1=E(M,K1,N1)$ dan $C2=E(M,K2,N2)$ dipengaruhi oleh perubahan kunci sekaligus perubahan nonce. Dengan demikian, 50,0238% merupakan hasil observasi program dan belum merupakan pengukuran murni sensitivitas terhadap kunci.

Masalah yang lebih jelas terjadi pada avalanche effect. Nilai rata-rata yang tercatat hanya 0,005442% dan maksimum 0,022748%. Nilai tersebut tidak boleh langsung ditafsirkan

sebagai kegagalan difusi. Pasangan plaintext dan plaintext yang diubah satu bit juga dienkripsi dengan nonce yang berbeda. Bentuk pengujian yang lebih tepat adalah $C1=E(M,K,N)$ dan $C2=E(M',K,N)$, dengan M dan M' hanya berbeda satu bit sedangkan K dan N tetap. Setelah pengujian tersebut dilakukan, hasil avalanche baru dapat dibandingkan secara bermakna dengan perilaku sekitar 50% yang diharapkan pada keluaran yang menunjukkan difusi kuat.



Gambar 8. Avalanche effect dan sensitivitas kunci pada pengujian awal.

Temuan ini merupakan aspek penting dari penelitian karena menunjukkan bahwa evaluasi keamanan harus mengendalikan variabel eksperimen. Literatur mutakhir tentang evaluasi enkripsi juga menggunakan avalanche, entropy, dan correlation secara bersama-sama, tetapi metrik statistik tetap tidak menggantikan analisis kriptografi yang lebih kuat. Dengan demikian, hasil avalanche awal ditempatkan sebagai hasil sementara yang memerlukan pengujian ulang.

4.6 Ringkasan Hasil 15 Berkas

File	Asli	PDE	Overhead (%)	Enkripsi (s)	Dekripsi (s)	Entropi C	Korelasi	Avalanche (%)	Key Sens. (%)	Correct
F1	8,053	8,181	1.589	0.0194	0.0168	7.9793	-0.0103	0.0093	49.9721	TRUE
F2	159,585	159,713	0.080	0.1796	0.2302	7.9988	-0.0026	0.0005	49.9690	TRUE
F3	159,587	159,715	0.080	0.1883	0.1658	7.9989	0.0058	0.0005	49.9704	TRUE
F4	12,511	12,639	1.023	0.0107	0.0142	7.9878	0.0029	0.0030	50.0190	TRUE
F5	16,336	16,464	0.784	0.0132	0.0138	7.9887	-0.0008	0.0023	50.1362	TRUE
F6	14,684	14,812	0.872	0.0116	0.0104	7.9862	0.0082	0.0051	50.0468	TRUE
F7	14,269	14,397	0.897	0.0119	0.0114	7.9869	-0.0007	0.0026	50.0526	TRUE

F8	17,357	17,485	0.737	0.0127	0.0139	7.9894	0.0005	0.0036	50.0792	TRUE
F9	16,851	16,979	0.760	0.0136	0.0139	7.9886	-0.0029	0.0037	49.9310	TRUE
F10	13,762	13,890	0.930	0.0096	0.0088	7.9863	0.0227	0.0027	49.8538	TRUE
F11	13,395	13,523	0.956	0.0122	0.0094	7.9823	-0.0043	0.0047	50.0887	TRUE
F12	3,297	3,425	3.882	0.0031	0.0040	7.9462	0.0183	0.0227	49.9697	TRUE
F13	8,683	8,811	1.474	0.0071	0.0057	7.9778	0.0012	0.0043	50.2822	TRUE
F14	2,539	2,667	5.041	0.0035	0.0034	7.9338	0.0247	0.0148	49.9557	TRUE
F15	20,673	20,801	0.619	0.0147	0.0138	7.9917	-0.0050	0.0018	50.0314	TRUE

Tabel 1. Ringkasan ukuran, waktu, dan indikator keamanan untuk 15 berkas.

4.7 Implikasi dan Keterbatasan

Secara teoritis, hasil penelitian menunjukkan bahwa kode representasi dari resolving partition dapat diposisikan sebagai komponen struktural dalam transformasi data tanpa harus menyimpan tuple representasi secara literal. Secara penerapan, pendekatan ini memungkinkan graf digunakan untuk membentuk pemetaan byte yang bergantung pada kunci. Namun, kontribusi tersebut tidak boleh diartikan bahwa dimensi partisi sendiri merupakan algoritma kriptografi lengkap. Keamanan ciphertext dalam PDE-SEC merupakan hasil gabungan permutasi, keystream, nonce, XOR, dan HMAC.

Keterbatasan utama penelitian adalah jumlah data uji yang masih terbatas pada 15 berkas, pengukuran waktu yang sangat singkat, dan belum dilakukannya pengujian kriptanalisis yang lebih luas. Selain itu, pengujian avalanche dan sensitivitas kunci pada tahap awal belum sepenuhnya terkontrol karena nonce berbeda. Keterbatasan ini harus dipertimbangkan sebelum hasil statistik digunakan untuk menyatakan tingkat keamanan.

4. SIMPULAN

PDE-SEC menunjukkan bahwa kode representasi simpul yang dibangun dari partisi pembeda pada graf siklus C256 dapat dimanfaatkan sebagai komponen struktural dalam pembentukan transformasi data berbasis kunci. Pada 15 berkas .tex yang diuji, seluruh proses enkripsi dan dekripsi berhasil dengan correctness 100%. Total ukuran data meningkat dari 481.582 menjadi 483.502 byte dengan overhead agregat 0,3987%, sehingga biaya tambahan format relatif kecil terutama pada berkas berukuran besar.

Ciphertext menghasilkan entropi rata-rata 7,9815 bit/byte dan korelasi absolut rata-rata 0,00739, sedangkan sensitivitas kunci yang tercatat sebesar 50,0238%. Waktu total enkripsi dan dekripsi masing-masing 0,457746 detik dan 0,752114 detik, dengan throughput agregat 1,0521 MB/s dan 0,6403 MB/s. Hasil tersebut menunjukkan keberhasilan fungsional dan karakteristik statistik ciphertext yang baik pada data uji, tetapi indikator tersebut tidak dengan sendirinya merupakan bukti keamanan kriptografi formal.

Nilai avalanche 0,0054% belum dapat dijadikan kesimpulan akhir karena pasangan pengujian menggunakan nonce yang berbeda. Oleh sebab itu, penelitian lanjutan perlu menggunakan nonce yang sama pada eksperimen avalanche dan key sensitivity yang

terkontrol, memperluas jumlah dan variasi data, melakukan pengujian known-plaintext dan chosen-plaintext, menganalisis ruang kunci, serta membandingkan prototipe dengan algoritma kriptografi standar. Dengan tahapan tersebut, posisi kontribusi PDE-SEC dapat dievaluasi secara lebih kuat dan terukur.

5. UCAPAN TERIMA KASIH

Penulis menyampaikan terima kasih kepada Program Studi Pendidikan Matematika, FKIP, Universitas Mataram, atas dukungan lingkungan akademik yang mendukung pelaksanaan penelitian ini. Ucapan terima kasih juga disampaikan kepada pihak-pihak yang membantu penyediaan data dan pengujian implementasi PDE-SEC.

6. REKOMENDASI

Penelitian selanjutnya direkomendasikan untuk memperbaiki modul pengujian sehingga nonce dapat diberikan secara eksplisit pada eksperimen terkontrol, sedangkan pada penggunaan operasional nonce tetap dibangkitkan secara aman dan unik. Dataset perlu diperluas dengan variasi ukuran dan karakteristik file. Pengujian keamanan juga perlu mencakup serangan known-plaintext, chosen-plaintext, analisis brute-force terhadap ruang kunci, pengujian integritas terhadap ciphertext yang dimodifikasi, serta perbandingan performa dan keamanan dengan konstruksi standar. Hambatan utama yang perlu diperhatikan adalah membedakan secara tegas indikator statistik dari bukti keamanan kriptografi formal.

7. REFERENSI

- Ali, N. A., Sadiqa, A., Shahzad, M. A., Qureshi, M. I., Siddiqui, H. M. A., Abdallah, S. A. O. O., & Abd El-Gawaad, N. S. (2024). Secure communication in the digital age: A new paradigm with graph-based encryption algorithms. *Frontiers in Computer Science*, 6, 1454094. <https://doi.org/10.3389/fcomp.2024.1454094>
- Amrullah, A., Hayati, L., & Junaidi, J. (2025). Partition dimension of bridge graphs between complete and star graphs. *International Journal of Computing Science and Applied Mathematics*, 11(2), 48–52. <https://doi.org/10.12962/j24775401.ijcsam.v11i2.8363>
- Baskoro, E. T., & Haryeni, D. O. (2020). All graphs of order $n \geq 11$ and diameter 2 with partition dimension $n - 3$. *Heliyon*, 6(4), e03694. <https://doi.org/10.1016/j.heliyon.2020.e03694>
- Chartrand, G., Salehi, E., & Zhang, P. (2000). The partition dimension of a graph. *Aequationes Mathematicae*, 59, 45–54.
- Daming, A. S., & Yuliani. (2024). Dimensi partisi graf hasil amalgamasi sisi graf roda dengan graf bintang. *Euler: Jurnal Ilmiah Matematika, Sains dan Teknologi*, 12(2).
- Fakhira, L. A., Hadi, N. W., Asmiati, A., & Nurvazly, D. E. (2025). The partition dimension of origami graphs and its barbell. *Indonesian Journal of Combinatorics*, 9(2). <https://doi.org/10.19184/ijc.2025.9.2.4>
- Hasibuan, I. M., & Abidin, W. (2025). On the partition dimension of a subdivision of complete graph and complete bipartite graph. *Jurnal Matematika UNAND*, 14(3), 216–224. <https://doi.org/10.25077/jmua.14.3.216-224.2025>
- Hasmawati, H., Hinding, N., Nurwahyu, B., Daming, A. S., & Amir, A. K. (2022). The partition dimension of the vertex amalgamation of some cycles. *Heliyon*, 8(6), e09596. <https://doi.org/10.1016/j.heliyon.2022.e09596>

- Haryeni, D. O., Baskoro, E. T., & Saputro, S. W. (2024). Family of graphs with partition dimension three. *Indonesian Journal of Combinatorics*, 8(2), 64–75. <https://doi.org/10.19184/ijc.2024.8.2.1>
- Koam, A. N. A., Ahmad, A., Alatawi, M. S., Khalil, A., Azeem, M., & Alsinai, A. (2024). On the constant partition dimension of some generalized families of Toeplitz graph. *Journal of Mathematics*, 2024, 4721104. <https://doi.org/10.1155/2024/4721104>
- Luo, R., Khalil, A., Ahmad, A., Azeem, M., Ibragimov, G., & Nadeem, M. F. (2022). Computing the partition dimension of certain families of Toeplitz graph. *Frontiers in Computational Neuroscience*, 16, 959105. <https://doi.org/10.3389/fncom.2022.959105>
- Ni, B., Qazi, R., Rehman, S. U., & Farid, G. (2021). Some graph-based encryption schemes. *Journal of Mathematics*, 2021, 6614172. <https://doi.org/10.1155/2021/6614172>
- Abdel-Kader, R. F., et al. (2024). Image encryption algorithms: A survey of design and evaluation metrics. *Journal of Cybersecurity and Privacy*, 4(1), 126–152. <https://doi.org/10.3390/jcp4010007>
- Design and optimization of nonlinear component of block cipher: Applications to multimedia security. (2024). *Ain Shams Engineering Journal*, 15(3), 102507. <https://doi.org/10.1016/j.asej.2023.102507>
- Lightweight image encryption using a novel chaotic technique for the safe Internet of Things. (2024). *International Journal of Computational Intelligence Systems*. <https://doi.org/10.1007/s44196-024-00535-3>
- Sine-Polynomial Chaotic Map (SPCM): A decent cryptographic solution for image encryption in wireless sensor networks. (2025). *International Journal of Computational Intelligence Systems*.
- Enhanced lightweight and compromised-resilient image encryption for resource constrained environments. (2025). *PLOS ONE*. <https://doi.org/10.1371/journal.pone.0320046>
- National Institute of Standards and Technology. (2000). Report on the development of the Advanced Encryption Standard (AES) (NISTIR 6390). U.S. Department of Commerce.
- National Institute of Standards and Technology. (2019). Recommendation for block cipher modes of operation: Galois/Counter Mode (GCM) and GMAC (NIST SP 800-38D). U.S. Department of Commerce.
- Nir, Y., & Langley, A. (2018). ChaCha20 and Poly1305 for IETF protocols (RFC 8439). Internet Engineering Task Force. <https://doi.org/10.17487/RFC8439>
- Krawczyk, H., Bellare, M., & Canetti, R. (1997). HMAC: Keyed-hashing for message authentication (RFC 2104). Internet Engineering Task Force. <https://doi.org/10.17487/RFC2104>