Jurnal Matematika Integratif.
Vol. 21, No. 1 (2025), pp. 41–54.

p-ISSN:1412-6184, e-ISSN:2549-903
doi:10.24198/jmi.v21.n1.62475.41-54

BASE: *Block Cipher Feistel* Berbasis *Enhanced Logistic Map*

I Ketut Yudi Sucipta, Edward Raja Parulian Lumban Tobing, dan Syauqi Akbar Al Fata

Program Studi Rekayasa Kriptografi, Politeknik Siber dan Sandi Negara
Jl. Raya Haji Usa, Putat Nutug, Ciseeng, Bogor, Jawa Barat 16120
Email: yudisucipta17@gmail.com

Abstrak

Dalam mendukung *Sustainable Development Goals* (SDGs), khususnya SDG 9 (Industri, Inovasi, dan Infrastruktur), dibutuhkan sistem transmisi data yang aman dan andal. Salah satu aspek krusial dalam mewujudkan hal ini adalah pengembangan algoritma enkripsi yang dapat menjamin kerahasiaan dan integritas data, terutama di tengah meningkatnya ancaman keamanan siber. Penelitian ini memperkenalkan BASE, sebuah algoritma enkripsi berbasis fungsi *chaos* yang dirancang untuk meningkatkan keamanan dalam sistem transmisi data guna mendukung pencapaian SDG 9. Algoritma ini dikembangkan menggunakan pendekatan desain rasional, dengan fungsi *chaos* sebagai elemen utama untuk memastikan keacakan tinggi serta ketahanan terhadap serangan kriptografi modern. Evaluasi algoritma BASE dilakukan melalui dua aspek utama, yaitu uji keacakan dan uji keamanan. Keacakan diuji menggunakan *Strict Avalanche Criterion* (SAC) *test*, yang dilakukan dalam dua skenario: (1) dengan kunci tetap dan *plaintext* acak, serta (2) dengan *plaintext* tetap dan kunci acak yang diuji dalam lima kali pengulangan untuk setiap skenario. Hasilnya menunjukkan bahwa BASE lulus uji SAC dalam sepuluh pengujian, yang mengindikasikan bahwa algoritma ini memiliki karakteristik keacakan yang baik. Sementara itu, keamanan BASE dievaluasi dengan serangan *algebraic attack*, yang bertujuan mencari persamaan matematis yang dapat merepresentasikan bit-bit *ciphertext*. Hasil pengujian menunjukkan bahwa tidak ditemukan persamaan yang dapat digunakan untuk memecahkan enkripsi, yang mengindikasikan bahwa BASE resisten terhadap serangan *algebraic attack*. BASE memiliki keacakan yang kuat serta ketahanan terhadap serangan kriptografi. Hal ini menjadikannya solusi yang mendukung aspek *confidentiality* dalam teknologi informasi, sehingga berkontribusi dalam pencapaian SDG 9.

Kata kunci: Asuransi pandemi, COVID-19, model *multiple state*, rantai Markov.

2000 Mathematics Subject Classification: 94A60.
Received: 24-03-2025, accepted: 18-05-2025.

Abstract

To support the Sustainable Development Goals (SDGs), particularly SDG 9 (Industry, Innovation, and Infrastructure), a secure and reliable data transmission system is essential. One crucial aspect of achieving this is the development of encryption algorithms that ensure data confidentiality and integrity, especially in an era where cybersecurity threats are increasing. This study introduces BASE, an encryption algorithm based on chaotic functions, designed to enhance security in data transmission systems and contribute to the realization of SDG 9. The algorithm is developed using a rational design approach, incorporating chaotic functions to ensure high randomness and resistance to modern cryptographic attacks. BASE is evaluated through two primary aspects: randomness testing and security analysis. The randomness evaluation is conducted using the Strict Avalanche Criterion (SAC) test, performed under two scenarios: (1) with a fixed key and random plaintext, and (2) with a fixed plaintext and random key, each tested five times. The results confirm that BASE passes the SAC test in all ten trials, indicating strong randomness properties. Meanwhile, the security of BASE is assessed through an algebraic attack, which attempts to derive mathematical equations representing ciphertext bits. The results show that no solvable equations were found, demonstrating that BASE is resistant to algebraic attacks. BASE exhibits strong randomness and resilience against cryptographic attacks. This makes it a suitable solution to support the aspect of confidentiality in information technology, thereby contributing to the achievement of SDG 9.

Keywords: *COVID-19, Markov chain, multiple state model, pandemic insurance.*

1. Pendahuluan

Menurut Saha *et al.* [1], *Sustainable Development Goals* (SDGs) merupakan serangkaian tujuan yang ditetapkan oleh Perserikatan Bangsa-Bangsa (PBB) untuk mencapai kehidupan yang lebih baik dan berkelanjutan bagi semua orang. Dari 17 tujuan SDGs, salah satu yang krusial dalam era digitalisasi adalah SDG 9, yaitu: Industri, Inovasi, dan Infrastruktur. Transformasi digital telah menjadi tulang punggung berbagai sektor industri dan infrastruktur, menjadikan data sebagai komoditas yang sangat berharga. Seiring dengan meningkatnya ketergantungan pada teknologi, keamanan data menjadi isu kritis yang perlu ditangani secara serius [2]. Dalam mewujudkan SDG 9, perlindungan data sangat penting untuk menjaga inovasi, melindungi hak kekayaan intelektual, serta memastikan infrastruktur digital yang kokoh [3]. Salah satu metode yang efektif untuk menjaga keamanan data adalah kriptografi, yang berperan dalam mengamankan komunikasi, transaksi, dan penyimpanan data dalam berbagai aplikasi industri, seperti *big data processing*, *Internet of Things* (IoT), dan sistem siber industri [4].

Aspek utama dalam keamanan data adalah kerahasiaan (*confidentiality*), yang memastikan informasi hanya dapat diakses oleh pihak yang berwenang. Menezes *et al.* [5] menjelaskan bahwa kriptografi mencakup berbagai elemen penting dalam keamanan informasi, termasuk *confidentiality, data integrity, entity authentication*, dan *data origin authentication.* Salah satu teknik yang paling umum digunakan adalah enkripsi, dengan AES-256 (Advanced Encryption Standard 256-bit) sebagai salah satu algoritma *block cipher* yang paling kuat

saat ini [5]. Bahkan, layanan pesan terenkripsi seperti WhatsApp dan Telegram *Secret Chat* menggunakan AES untuk melindungi komunikasi pengguna.

Namun, meskipun AES-256 telah menjadi standar global, berbagai serangan terhadap algoritma ini telah dikembangkan.*Related Key-Boomerang Attack*, yang pertama kali diperkenalkan oleh Biryukov dan Khovratovich pada 2009, telah digunakan untuk mengeksploitasi kelemahan AES-256 [6]. Kemudian, pada tahun 2022, Jian Guo, Ling Song, dan Haoyang Wang berhasil menyempurnakan serangan ini, memungkinkan pemulihan kunci dengan reduksi kompleksitas waktu dan data hingga 28 kali [6]. Kondisi ini memunculkan kebutuhan mendesak akan pengembangan algoritma enkripsi alternatif yang lebih tangguh dalam menghadapi evolusi teknik kriptanalisis modern. Berangkat dari tantangan tersebut, penelitian ini mengusulkan BASE, algoritima *block cipher* yang memanfaatkan sifat dinamis fungsi *chaos* untuk meningkatkan keacakan dan ketahanan struktural.

Rancangan algoritma BASE bertujuan untuk mengoptimalkan integrasi fungsi *chaos* dalam struktur enkripsi guna mencapai tingkat keacakan dan keamanan yang melebihi algoritma konvensional. Evaluasi menyeluruh akan dilakukan melalui serangkaian pengujian, meliputi Strict Avalanche Criterion (SAC) *test* dan analisis ketahanan terhadap serangan kriptografi. Keunggulan pendekatan ini diharapkan tidak hanya memberikan solusi teknis dalam bidang kriptografi, tetapi juga berkontribusi pada penguatan infrastruktur digital yang berkelanjutan sesuai dengan prinsip SDGs.

Signifikansi penelitian ini terletak pada dua aspek utama. Pertama, sebagai upaya meningkatkan kesadaran akan pentingnya proteksi data di era transformasi digital yang semakin masif. Kedua, temuan penelitian ini diharapkan dapat menjadi landasan bagi inovasi algoritma enkripsi masa depan yang lebih adaptif terhadap perkembangan ancaman siber. Dengan demikian, pengembangan algoritma BASE tidak hanya menjawab kebutuhan praktis di bidang keamanan data, tetapi juga mendukung terciptanya ekosistem digital yang lebih aman dan berkelanjutan.

2. Landasan Teori

2.1. **Block Cipher.** *Block cipher* merupakan sebuah fungsi dalam kriptografi yang memetakan n-bit blok *plaintext* ke n-bit blok *ciphertext*. Secara garis besar, *block cipher* harus memenuhi sifat konfusi dan difusi. Konfusi berarti setiap bit *ciphertext* harus independen pada beberapa bagian dari kunci yang menyamarkan keterkaitan antara keduanya. Sedangkan difusi berfokus pada *avalanche effect*, yaitu modifikasi atau perubahan kecil pada *plaintext* harus menyebar ke seluruh *ciphertext* [5].

2.2. ***Enhanced Logistic Map* (ELM).** Dalam kriptografi, c*haotic maps* sering digunakan dalam enkripsi gambar, fungsi *hash*, dan algoritma *watermarking* [7]. Chaotic maps satu dimensi (1D), seperti logistic map, memiliki keunggulan dalam efisiensi komputasi namun terbatas dalam rentang *chaos* dan tingkat keacakan [7]. *Logistic map* secara matematis dinyatakan sebagai:

$$\gamma_{n+1} = f_L(\eta, \gamma_n) = \eta \cdot \gamma_n \cdot (1 - \gamma_n) \tag{1}$$

dengan γ_n sebagai variabel *state* dalam interval [0,1] dan η sebagai parameter sistem $f_L(\eta, \gamma_n)$ sebagai fungsi iterasi. Namun, penggunaan langsung *logistic map* dalam kriptografi dapat menyebabkan masalah keamanan [8]. Untuk mengatasi hal ini, Enhanced Logistic Map (ELM) dikembangkan dengan pendekatan *chatification*, yang memodifikasi fungsi *chaos* dasar guna meningkatkan karakteristik *chaotic*-nya.

ELM didefinisikan sebagai:

$$\gamma_{n+1} = f_{ELM}(\eta, \gamma_n, 10) = \frac{2^{10}}{2^{f_L(\eta, \gamma_n)}}. \tag{2}$$

Studi oleh Masri *et al.* [9] merekomendasikan ELM untuk digunakan dalam kriptografi primitif karena sifat *chaotic* yang lebih baik dibandingkan metode konvensional.

2.3. **Uji SAC.** SAC adalah suatu sifat yang dimiliki oleh suatu fungsi sedemikian hingga apabila terdapat perubahan satu bit *input* akan menyebabkan perubahan bit-bit *output* dengan probabilitas 0.5. Menurut Sulak *et al.* [10], SAC *test* diusulkan dengan tujuan untuk menentukan apakah suatu fungsi memenuhi sifat SAC atau tidak. Langkah awal dalam SAC *test* adalah membuat matriks SAC. Tahapan-tahapan dalam pembuatan matriks SAC ditunjukkan pada Algoritma 1.

Algorithm 1 Matriks SAC

Input: Fungsi f yang akan dievaluasi, 2^{20} buah pesan input
Output: Matriks M berukuran $m \times n$
Prosedur:

(1) Inisialisasi Matriks $M = \{m^{ij}\}_{i=1,2,3,\ldots,m;j=1,2,3,\ldots,n}$ dengan $m^{ij} = 0$.
(2) Untuk $a = 1$ sampai 2^{20}:
  (a) Hitung $Y_a = f(X_a)$.
  (b) Untuk $i = 1$ sampai m:
    (i) Lakukan perubahan bit ke-i pada X_a sehingga didapatkan X_a^i.
    (ii) Hitung $Y_a^i = f(X_a^i)$.
    (iii) Hitung $Z_a^i = (Z_a^{i1}, Z_a^{i2}, Z_a^{i3}, \ldots, Z_a^{in}) = Y_a \oplus Y_a^i$, dengan Z_a^{ij} merupakan bit ke-j dari Z_a^i.
(3) Bentuk matriks $Z_a = \{Z_a^{ij}\}_{i=1,2,3,\ldots,m;j=1,2,3,\ldots,n}$.
(4) Perbarui matriks M dengan $M = M \oplus Z_a$, yaitu $m^{ij} = m^{ij} \oplus Z_a^{ij}$.

Hasil akhir matriks SAC akan dievaluasi menggunakan metode *chi-square goodness of fit test*. Nilai dari setiap entri matriks M didistribusikan ke dalam kelas-kelas yang telah ditentukan pada Tabel 1, lalu dihitung frekuensinya, yang disebut nilai observasi (o_i). Selanjutnya, dilakukan perbandingan antara o_i dengan nilai harapan (e_i) yang dihitung secara empiris menggunakan *chi-square goodness of fit test* dengan derajat kebebasan $df = 4$ dan taraf signifikansi $\alpha = 0.01$. Evaluasi pertama berhasil jika *p_value* dari *chi-square goodness of fit test* lebih besar dari $\alpha = 0.01$.

Tabel 1. Kelas uji SAC untuk 2^{20} percobaan

Kelas	Rentang	Probabilitas
1	0 – 523857	0.200224
2	523858 – 524158	0.199937
3	524159 – 524417	0.199677
4	524418 – 524718	0.199937
5	524719 – 1048576	0.200224

2.4. **Serangan Aljabar.**

Definisi 2.1. *Serangan aljabar (algebraic attack) merupakan pendekatan kriptanalisis yang memanfaatkan representasi matematis dari struktur internal sebuah block cipher dalam bentuk sistem persamaan aljabar, biasanya polinomial atas field hingga seperti* $GF(2)$.

Tujuan utama serangan ini adalah untuk memecahkan kunci enkripsi atau merekonstruksi *plaintext* dengan menyelesaikan sistem persamaan tersebut menggunakan teknik komputasi aljabar [12]. *Block cipher*, seperti AES atau DES, yang mengandalkan kombinasi operasi linear

(misalnya, permutasi dan XOR) serta non-linear (misalnya, S-box), menjadi target analisis karena operasi-operasi ini dapat dimodelkan sebagai hubungan polinomial.

Dalam merancang algoritma *block cipher*, kami mengacu pada analisis aljabar mendalam yang dilakukan terhadap algoritma LCB (*Light Cipher Block*) dan variannya yang ditingkatkan (*improved* LCB), sebagaimana diuraikan dalam penelitian oleh Sucipta *et al.* [13]. Studi tersebut menunjukkan bahwa LCB asli memiliki kerentanan terhadap serangan aljabar, terutama karena sifat aljabar liniernya dan kurangnya kompleksitas dalam skema enkripsi, yang memudahkan eksploitasi oleh penyerang. Sebaliknya, versi *improved* LCB yang diperkenalkan oleh Chan *et al.* Temuan ini menjadi landasan penting bagi pendekatan kami dalam menciptakan algoritma *block cipher* yang lebih aman, dengan fokus pada penghindaran struktur aljabar linier yang sederhana dan peningkatan kompleksitas untuk menangkal serangan aljabar yang potensial. Temuan ini menjadi landasan penting bagi pendekatan kami dalam menciptakan algoritma *block cipher* yang lebih aman, dengan fokus pada penghindaran struktur aljabar linier yang sederhana dan peningkatan kompleksitas untuk menangkal serangan aljabar yang potensial.

3. Metode Penelitian

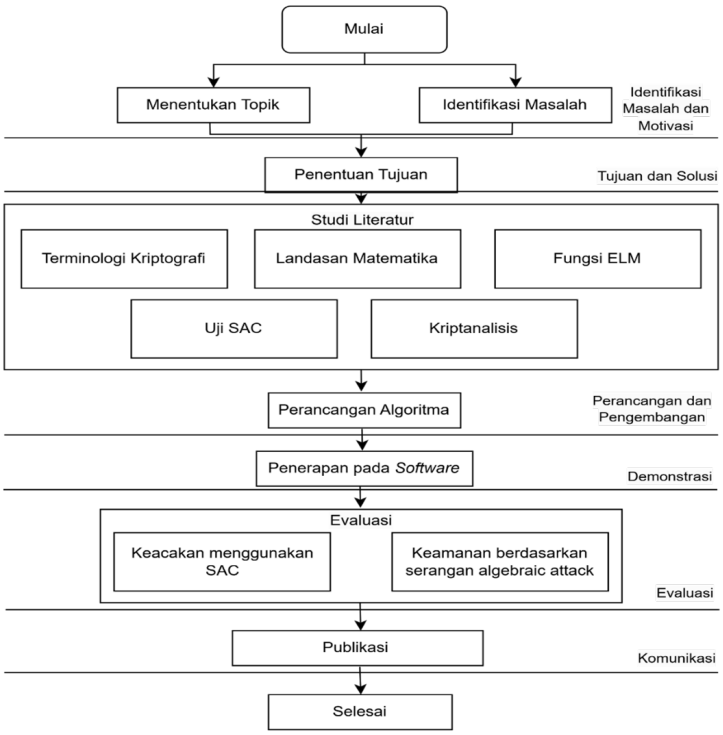
3.1. **Jenis Penelitian.** Penelitian ini bersifat eksperimental dan kuantitatif. Penelitian eksperimental dilakukan untuk merancang algoritma *block cipher* yang diusulkan dengan pendekatan desain rasional, sementara penelitian kuantitatif dilakukan untuk mengevaluasi desain dengan pengujian hipotesis guna memperoleh kesimpulan.

3.2. **Desain Penelitian.** Desain penelitian ini menggunakan metode *Design Science Research Methodology* (DSRM) yang terdiri dari enam tahapan, yaitu Identifikasi Masalah dan Motivasi, Tujuan dari Solusi, Perancangan dan Pengembangan, Demonstrasi, Evaluasi, dan Komunikasi, seperti yang dijelaskan oleh Peffers *et al.* [14] dan diselaraskan serta diilustrasikan pada Gambar 1.

Identifikasi masalah dan motivasi dilakukan melalui penentuan topik penelitian serta analisis mendalam terhadap tantangan kriptografi kontemporer, seperti yang diuraikan pada pendahuluan. Berdasarkan identifikasi masalah tersebut, kemudian dirumuskan tujuan solusi yang menjadi landasan pengembangan algoritma BASE. Tahap perancangan dan pengembangan meliputi studi literatur komprehensif mengenai terminologi kriptografi, landasan matematika, fungsi ELM, teknik kriptanalisis, dan uji SAC, yang menjadi dasar untuk merancang algoritma BASE dengan pendekatan desain rasional. Proses demonstrasi dilakukan melalui simulasi implementasi algoritma dalam perangkat lunak untuk memvalidasi konsep rancangan yang dikembangkan. Selanjutnya, evaluasi kuantitatif dilakukan untuk mengukur performa algoritma, mencakup pengujian keacakan menggunakan SAC dan analisis ketahanan terhadap serangan aljabar (*algebraic attack*). Hasil penelitian ini kemudian dikomunikasikan melalui publikasi ilmiah dan presentasi di forum akademik untuk mendapatkan masukan dan validasi dari komunitas peneliti terkait. Keseluruhan tahapan ini dirancang untuk memastikan bahwa pengembangan algoritma BASE memenuhi standar akademik dan kebutuhan praktis di bidang keamanan data.

4. Hasil dan Pembahasan

4.1. **Deskripsi Algoritma BASE.** Desain Algoritma BASE merupakan salah satu keluarga dari *block cipher* yang memiliki 128-Bit *input plaintext* menghasilkan 128-bit *output ciphertext* dalam 10 *round* enkripsi dengan panjang kunci 128-bit yang kemudian dijadwalkan menjadi 10 *subkey* dengan panjang masing-masing 64-bit yang digunakan pada setiap *round*, fungsi F yang digunakan menerapkan fungsi ELM yang merupakan salah satu fungsi *chaos*. ELM berguna untuk menambah sifat *chaotic* pada algoritma BASE sehingga keacakan dari *ciphertext* yang akan dihasilkan meningkat.



Gambar 1. Skema Penyelarasan DSM

4.1.1. *Penjadwalan Kunci.* Penjadwalan kunci BASE dilakukan dengan melibatkan fungsi ELM di dalamnya, hal ini dimaksudkan untuk menambah sifat *chaotic* dari variabel-variabel kuncinya. Algoritma dari penjadwalan kunci pada BASE dijelaskan pada algoritma 2.

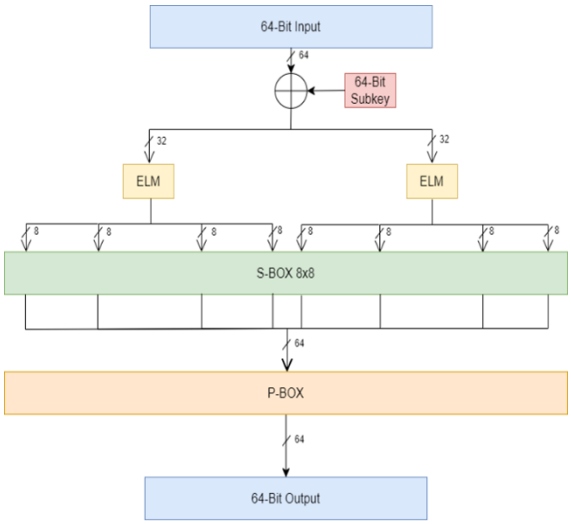
Algorithm 2 Key Scheduling BASE

Input: Kunci 128-bit `key`
Output: Subkey K_1 sampai K_{10}
Prosedur:

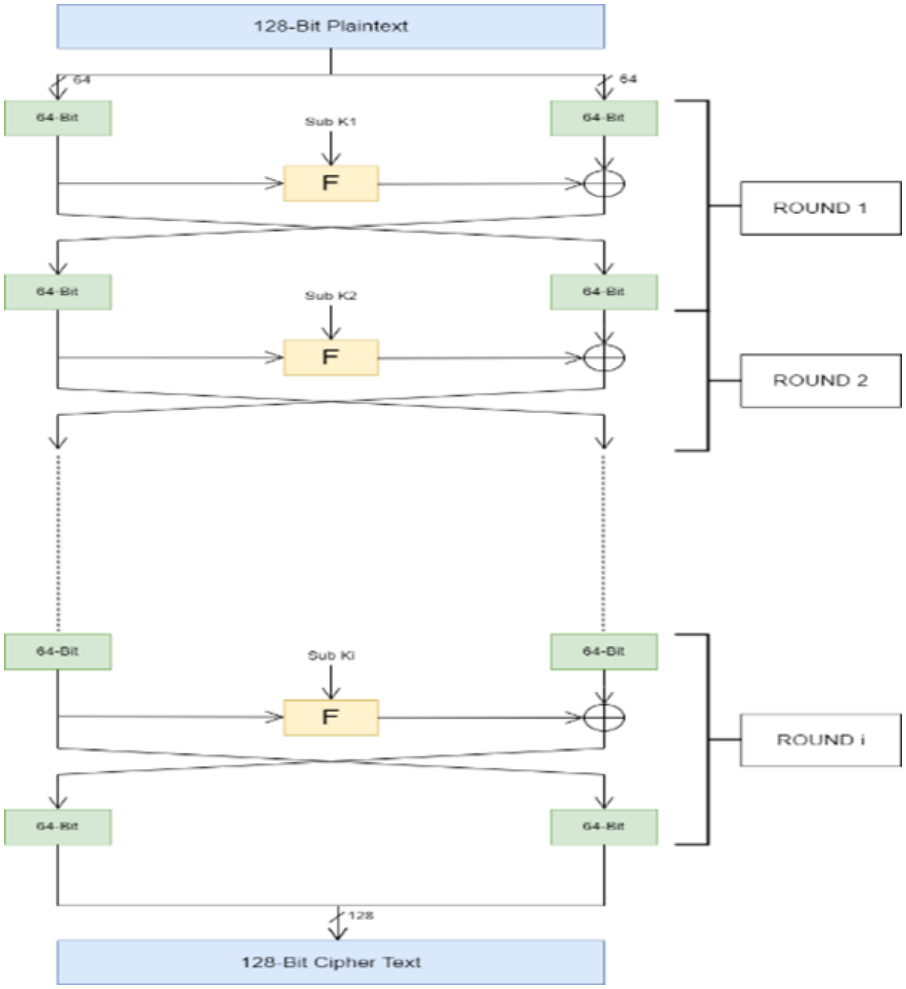
(1) Bagi kunci `key` menjadi dua bagian:
  (a) $K_1 = \texttt{key}[0:64]$
  (b) $K_2 = \texttt{key}[64:128]$
(2) Inisialisasi daftar subkey dengan K_1 dan K_2: `subkeys` $\leftarrow [K_1, K_2]$.
(3) Untuk $i = 2$ hingga 9:
  (a) Ambil 32-bit LSB dari K_{i-2}: $a = K_{i-2}[32:64]$.
  (b) Ambil 32-bit MSB dari K_{i-1}: $b = K_{i-1}[0:32]$.
  (c) Hitung K_i sebagai:
    (i) $K_i = \texttt{ELM}(a) \parallel b$.
  (d) Tambahkan K_i ke dalam daftar `subkeys`.
(4) Kembalikan daftar `subkeys` yang berisi K_1 hingga K_{10}.

4.1.2. *Skema Enkripsi dan Dekripsi.* Desain skema enkripsi dan fungsi F pada algoritma BASE yang mengenkripsi 128-bit *input* menjadi 128-bit *output*. fungsi F dan Skema enkripsi yang didesain diilustrasikan pada Gambar 2 dan Gambar 3 secara berturut-turut.

Pada proses dekripsi akan dilakukan secara invers atau terbalik dimulai dari *round*-n hingga *round*-1 begitu pula dengan *subkey* yang digunakan merupakan *subkey* ke-n, $n-1$,



GAMBAR 2. Fungsi F algoritma BASE



GAMBAR 3. Skema Enkripsi algoritma BASE

seterusnya hingga *subkey* ke-1, namun tetap menggunakan fungsi F yang sama. *P-box* yang digunakan pada fungsi F adalah permutasi bit pada PRESENT yang dijelaskan kembali oleh Mohanapriya dan Kumar [15]. Hal ini dikarenakan permutasinya memiliki sifat difusi yang baik. *S-box* yang digunakan diadopsi dari Qazy *et al.*[16] terlihat pada Gambar 4.

254	92	73	87	156	100	236	205	23	237	14	197	177	227	148	215
46	122	57	68	97	232	172	228	70	184	35	229	139	225	82	150
69	127	63	41	18	51	226	250	167	58	245	212	115	132	24	126
93	119	66	5	204	183	143	45	8	220	199	34	86	251	106	249
135	105	28	235	19	209	160	75	0	129	146	198	223	186	114	59
53	113	83	88	246	103	118	166	142	9	196	29	230	216	241	182
16	239	33	96	32	149	181	47	15	91	131	231	224	222	253	171
99	233	252	49	218	158	214	54	134	168	169	195	109	101	22	42
153	244	202	189	175	36	3	138	147	213	44	173	136	107	111	163
94	108	7	4	95	133	39	20	117	110	174	25	164	242	170	55
40	76	243	162	161	178	176	38	145	194	141	207	2	152	217	203
77	60	165	71	208	43	81	192	238	116	154	30	48	137	144	50
102	191	17	188	187	201	10	123	130	78	84	255	72	157	62	179
234	185	211	98	190	21	90	6	26	155	79	31	12	27	61	120
210	124	67	74	80	56	248	247	65	140	85	13	64	200	52	11
37	221	121	159	151	112	125	89	180	206	193	240	219	128	104	1

Gambar 4. S-box pada Algoritma BASE [16]

S-box ini dipilih karena memenuhi persyaratan *S-box* sebagai pemetaan bit yang *nonlinear*. Selain itu *S-box* ini dibangun berdasarkan *chaotic map* memiliki sifat *chaos*. Sedangkan fungsi ELM diadopsi dan dimodifikasi berdasarkan Masri *et al.* [9]. Fungsi ELM ini dipilih dikarenakan telah ditunjukkan bahwa ELM memiliki sifat *chaotic* serta konfusi dan difusi yang lebih baik daripada fungsi *chaos* 1 dimensi sehingga dapat diterapkan sebagai komponen dari primitif kriptografi [9]. Fungsi ELM yang dimodifikasi dijelaskan pada Algoritma 3.

Algorithm 3 ELM

Input: 32-bit `x`
Output: 32-bit `y`
Prosedur:

(1) Bagi `x` menjadi tiga bagian:
  (a) x_l = 12-bit pertama dari `x`
  (b) x_m = 16-bit kedua dari `x`
  (c) x_n = 4-bit ketiga dari `x`
(2) Hitung parameter:
  (a) $\gamma_0 = x_l \cdot \frac{1}{2^{12}}$
  (b) $\eta = \left(x_m \cdot \frac{2}{2^{16}}\right) + 2$
  (c) $k = \left(x_l \cdot \frac{1}{2^4}\right) + 10.01$
  (d) $n = \lfloor 6 \cdot \gamma_0 \rfloor$
(3) Gunakan parameter γ_0, η, dan k dalam persamaan (2).
(4) Hitung nilai:
  (a) $w_1 = \text{binary32}(\eta_{n+1} \cdot 10^{10})$
  (b) $w_2 = \text{binary32}(\eta_{n+2})$
  berdasarkan aturan konversi IEEE-754 *Floating Point Converter* tahun 2024.
(5) Hitung nilai keluaran:
  (a) $y = (w_1 \ll 17) \oplus w_2$

4.2. **Uji Keacakan dan Keamanan pada Algoritma BASE.**

4.2.1. *SAC Test.* Untuk melakukan enkripsi menggunakan BASE, diperlukan dua parameter utama, yaitu *plaintext* 128-bit dan kunci 128-bit. Berdasarkan hal ini, pengujian *Strict Avalanche Criterion* (SAC) pada algoritma BASE dilakukan dalam dua skenario utama:

(1) Skenario pertama: Kunci tetap, sementara *plaintext* diubah secara acak.
(2) Skenario kedua: *Plaintext* tetap, sementara kunci diubah secara acak.

Setiap skenario diuji sebanyak lima kali untuk memastikan bahwa hasilnya bukan kebetulan, sehingga total terdapat sepuluh pengujian SAC pada BASE. Uji keacakan ini bertujuan untuk mengevaluasi sejauh mana perubahan kecil pada *input* (*plaintext* atau kunci) dapat menyebabkan perubahan signifikan pada *output* (*ciphertext*), sesuai dengan prinsip *avalanche effect* dalam kriptografi.

Untuk mengimplementasikan pengujian ini, Algoritma 1 digunakan sebagai dasar dalam pembuatan source code Python yang menjalankan SAC *test* terhadap BASE. Hasil dari sepuluh kali pengujian SAC dapat dilihat pada Tabel 8. Langkah-langkah SAC *test* pada algoritma BASE dijelaskan secara rinci sebagai berikut:

(1) Membangkitkan sampel M_a pesan input acak sebanyak 2^{20}, seperti yang ditampilkan pada Tabel 2

Tabel 2. Sampel 2^{20}

a	X_a
1	697109D5E7338F0A3E34F22E541BC6EC
2	ED403B1C7233F9EE5449C0A489F0B888
3	A07E68130B23953C89061B06B498C983
⋮	⋮
2^{20}	57C0FEA20C89CB3A5C2AE3257B419494

(2) Kemudian pilih salah satu pesan acak, misal X_1 kemudian hitung *Ciphertext*-nya. $C_1 = BASE(M_1) = BASE(697109D5E7338F0A3E34F22E541BC6EC)$
$= 9754736A82B0AD8A7B0D8246F0DC253D$
(3) Kemudian untuk setiap posisi bit ke-i dari M_1, lakukan *flipping bit* dan simpan sebagai M_1^i yang kemudian hitung *ciphertext*-nya dan simpan sebagai C_1^i dengan $1 \leq i \leq 64$. Nilai pesan dan *ciphertext* dari *flipping bit* ditampilkan pada Tabel 3
(4) Melakukan operasi XOR antara *ciphertext* pesan asli M_i dengan*ciphertext* pesan hasil *flipping bit* M_1^i sehingga diperoleh $Z_1^i = M_1 \oplus M_1^i$. Himpunan nilai Z_1 dapat dilihat pada Tabel 4.
(5) Lakukan tahap kedua sampai keempat untuk $2^{20} - 1$ sampel pesan lainnya
(6) Mengelompokkan setiap entri matriks SAC akhir ke dalam kelas i berdasarkan rentangnya dan menghitung frekuensi observasi yang dinotasikan sebagai o_i dengan $i = 1, 2, \ldots, 5$. Frekuensi observasi o_i hasil SAC test pada pengujian pertama algoritma Atakee diperlihatkan pada tabel 5
(7) Mengevaluasi nilai matriks menggunakan chisquare goodness of fit test dengan $\alpha = 0.01$ dan $df = 4$. Frekuensi ekspektasi e_i dengan cara $e_i = v \cdot p_i$ dengan $v = n \times m = n \cdot m = nm$ dan p_i merupakan probabilitas ekspektasi pada kelas ke-i yang telah disajikan pada Tabel 6
(8) Dari nilai-nilai e_i yang diperoleh akan dihitung nilai *chi-square* χ^2. Tabel 7 menunjukkan perhitungan nilai *chi-square* χ^2 pada SAC *test*

Tabel 3. Nilai Hash setelah Bit Flip

a	$M_a^{'i}$	$C_a^{'i}$
1	697109D5E7338F0A3E34F22E541BC6ED	9754736A82B0AD8A7B0D8246F0DC253D
2	697109D5E7338F0A3E34F22E541BC6EE	9EC6631163239C76B088F92C72FF8398
3	697109D5E7338F0A3E34F22E541BC6EF	ABF65972E5D6C15B91E6672CC304658F
⋮	⋮	⋮
2^{20}	E97109D5E7338F0A3E34F22E541BC6EC	C5793F4EADB2FF84F759134719DAD1E1

Tabel 4. XOR Pesan awal dan Pesan Bitflip

a	M_a^i	C_a^i	$Z_a^i = C_a^i \oplus C_a$
1	697109 . . . 1BC6ED	975473 . . . DC253D	FE257A . . . C7E3D0
2	697109 . . . 1BC6EE	9EC663 . . . 2FF8398	F7B76A . . . 6E44574
3	697109 . . . 1BC6EF	ABF659 . . . 04658F	C28750 . . . 71FA363
⋮	⋮	⋮	⋮
2^{20}	E97109D . . . 1BC6EC	C5793F . . . DAD1E1	AC0836 . . . DC1170D

Tabel 5. Jumlah dalam kelas interval

Kelas	Rentang	o_i
1	0 - 523857	3292
2	523858 - 524158	3270
3	524159 - 524417	3221
4	524418 - 524718	3224
5	524719 - 1048576	3377

Langkah-langkah di atas dilakukan sebanyak 10 kali untuk memastikan hasil dari SAC *test* bukan suatu kebetulan. Hasil uji sepuluh kali SAC uji SAC pada BASE dapat dilihat pada Tabel 8. Dari hasil yang ditampilkan pada Tabel 8, dapat dilihat bahwa semua nilai p_{value} lebih besar dari $\alpha = 0.01$. Hal ini menunjukkan bahwa BASE berhasil memenuhi kriteria uji SAC, yang mengindikasikan bahwa algoritma ini memiliki karakteristik keacakan yang baik. Dengan demikian, BASE menunjukkan ketahanan yang baik terhadap kriptanalisis berbasis *avalanche effect*, sehingga dapat digunakan sebagai alternatif algoritma block cipher yang kuat dalam menjaga keamanan data.

Langkah-langkah di atas dilakukan sebanyak 10 kali untuk memastikan hasil dari SAC *test* bukan suatu kebetulan. Hasil uji sepuluh kali SAC uji SAC pada BASE dapat dilihat pada

Tabel 6. Tabel Probabilitas Kelas Interval

Kelas	Rentang	Probabilitas
1	0 - 523857	0.200224
2	523858 - 524158	0.199937
3	524159 - 524417	0.199677
4	524418 - 524718	0.199937
5	524719 - 1048576	0.200224

Tabel 7. Tabel Perhitungan *P Value*

i	e_i	o_i	$o_i - e_i$	$(o_i - e_i)^2$	$(o_i - e_i)^2/e_i$
1	3280.470016	3292	-11.5300	132.9405	0.0405
2	3275.767808	3270	5.7678	33.2676	0.0102
3	3276.259328	3221	55.2593	3053.5933	0.9320
4	3275.767808	3224	51.7678	2679.9059	0.8181
5	3280.470016	3377	-96.5300	9318.0378	2.8405
χ^2	4.6413				
p_{value}	0.3261				

Tabel 8. **Hasil Uji SAC pada BASE**

No	**Kunci**	***Plaintext***	p_{value}	**Status Uji**
1	0x123A4F30295FB1B10D542D8574DC8E88		0.573554	lulus
2	0x5EC06D9E63459ADC8BB7C5FB8AA8B30E		0.178359	lulus
3	0xB6CFD8EDD40A9773289A76B3FA275C26	*Random*	0.078367	lulus
4	0x28064F9CDC61D673387B5E295288BCEA		0.147724	lulus
5	0xCF58AB8BFFB3EA6CB17D56E61BBD762C		0.322079	lulus
6		0x810035D10D0E3181A33D7A143CE78EA4	0.508941	lulus
7		0x8D0F87FC6A432748217584C24376B82E	0.889677	lulus
8	*Random*	0xBF6BE0310B39FED751D767C1DDDDFBC6	0.923908	lulus
9		0x4FDDA037AB2B7C61048BA580CAF96AED	0.150585	lulus
10		0x3551B9E1834674BD74C620748BA1624C	0.809141	lulus

Tabel 8. Dari hasil yang ditampilkan pada Tabel 8, dapat dilihat bahwa semua nilai p_{value} lebih besar dari $\alpha = 0.01$. Hal ini menunjukkan bahwa BASE berhasil memenuhi kriteria uji SAC, yang mengindikasikan bahwa algoritma ini memiliki karakteristik keacakan yang baik. Dengan demikian, BASE menunjukkan ketahanan yang baik terhadap kriptanalisis berbasis *avalanche effect*, sehingga dapat digunakan sebagai alternatif algoritma block cipher yang kuat dalam menjaga keamanan data.

4.2.2. *Serangan Aljabar.* Dalam penelitian ini, dilakukan analisis aljabar terhadap metode enkripsi BASE untuk mengevaluasi ketahanannya terhadap *algebraic attack*. Langkah pertama

adalah merepresentasikan setiap output dari komponen enkripsi BASE sebagai persamaan matematis, dengan *input* komponen sebagai variabel. Pendekatan ini bertujuan untuk menyusun sistem persamaan yang menggambarkan hubungan antara *ciphertext* dan bit-bit kunci, sehingga memungkinkan analisis lebih lanjut terhadap kemungkinan pemulihan kunci.

Pencarian persamaan dilakukan dengan menggunakan Python, dieksekusi pada perangkat dengan RAM 8 GB dan prosesor Intel Core i5. Dengan spesifikasi ini, proses komputasi berhasil dilakukan hingga *round* ke-3, sebelum akhirnya dihentikan karena keterbatasan memori. Tingkat kompleksitas yang meningkat pada setiap *round* menunjukkan bahwa persamaan yang terbentuk semakin rumit, dengan jumlah monomial yang bertambah signifikan serta derajat polinomial yang semakin tinggi.

Meskipun hanya tersedia 128 persamaan, kompleksitas penyelesaiannya sangat tinggi, terbukti dari kegagalan menemukan solusi pada *round* ke-3. Ketidakterpecahan sistem persamaan ini mengindikasikan bahwa BASE memiliki ketahanan yang kuat terhadap serangan aljabar, karena peningkatan jumlah *round* semakin memperumit struktur persamaan yang harus dipecahkan. Dengan demikian, dapat disimpulkan bahwa BASE resisten terhadap *algebraic attack*.

5. Simpulan

BASE adalah algoritma *block cipher* yang mengenkripsi 128-bit *plaintext* menggunakan parameter kunci simetris 128-bit. Algoritma ini dirancang secara rasional dengan struktur Feistel dan memanfaatkan fungsi *chaos* sebagai komponen utama. Evaluasi dilakukan melalui pengujian keacakan dengan uji SAC sebanyak sepuluh kali serta analisis terhadap potensi serangan *algebraic attack*. Hasil menunjukkan bahwa BASE lulus seluruh pengujian SAC, mengindikasikan properti keacakan yang baik. Selain itu, serangan *algebraic attack* menghasilkan sistem persamaan berdimensi kompleks dengan banyak monomial dan derajat tinggi, sehingga menunjukkan resistansi terhadap serangan tersebut. Keberhasilan BASE dalam memenuhi aspek keacakan dan ketahanan terhadap kriptanalisis dasar menjadikannya kandidat potensial untuk diterapkan dalam *environment* yang menuntut keamanan tinggi. Potensi pengembangan lebih lanjut mencakup optimalisasi performa, penerapan pada sistem dengan keterbatasan sumber daya seperti IoT, serta pengujian terhadap vektor serangan lanjutan. Kontribusi utama dari penelitian ini adalah rancangan algoritma yang sederhana namun kuat, serta bukti awal ketahanannya, yang dapat menjadi dasar bagi riset dan pengembangan sistem kriptografi selanjutnya.

Ucapan Terima Kasih

Dengan hormat, penulis menyampaikan terima kasih kepada Ibu Bety Hayat Susanti atas bimbingan, arahan, dan dukungan yang sangat berarti selama proses penelitian dan penulisan paper ini. Ucapan terima kasih juga disampaikan kepada Politeknik Siber dan Sandi Negara atas fasilitas dan dukungan akademik yang telah diberikan. Semoga kontribusi seluruh pihak mendapat balasan yang setimpal dan semoga karya ini bermanfaat bagi pengembangan ilmu kriptografi.

Daftar Pustaka

[1] P. Saha, H. M. Belal, dan S. Talapatra, "Driving Toward Sustainable Development Goals (SDGs) in the Ready-Made Garments (RMGs) Sector: The Role of Digital Capabilities and Operational Transparency," *IEEE Trans. Eng. Manage.*, vol. 71, hlm. 14071–14082, 2024. doi: 10.1109/TEM.2024.3439290.

[2] G. Dede, A. M. Petsa, S. Kavalaris, E. Serrelis, S. Evangelatos, dan I. Oikonomidis, "Cybersecurity as a Contributor to United Nations SDGs towards enhancing the ESG Reporting," dalam *Proc. 2024*

5th Int. Conf. Electron. Eng., Inf. Technol. & Educ. (EEITE), Chania, Greece, 2024, hlm. 1–6. doi: 10.1109/EEITE61750.2024.10654435.

[3] I. Pandey, S. Kumari, dan A. Vij, "Artificial Intelligence and Regulatory Framework for Sustainable Plastic Management: An Analytical Study towards Achieving SDGs," dalam *Proc. 2023 Int. Conf. Commun., Security and Artif. Intell. (ICCSAI)*, Greater Noida, India, 2023, hlm. 592–597. doi: 10.1109/ICCSAI59793.2023.10421531.

[4] C.-H. Lin et al., "Intelligent Symmetric Cryptography With Chaotic Map and Quantum Based Key Generator for Medical Images Infosecurity," *IEEE Access*, vol. 9, hlm. 118624–118639, 2021. doi: 10.1109/ACCESS.2021.3107608.

[5] J. Menezes, P. C. Van Oorschot, dan S. A. Vanstone, *Handbook of Applied Cryptography*. Boca Raton, FL, USA: CRC Press, 2018.

[6] J. Guo, L. Song, dan H. Wang, "Key Structures: Improved Related-Key Boomerang Attack Against the Full AES-256," dalam *Lecture Notes in Computer Science, vol. 13494*. Cham: Springer, 2022, hlm. 1–31. doi: 10.1007/978-3-031-22301-31.

[7] M. Alawida, J. S. Teh, A. Mehmood, A. Shoufan, dan W. H. Alshoura, "A chaos-based block cipher based on an enhanced logistic map and simultaneous confusion-diffusion operations," *J. King Saud Univ. - Comput. Inf. Sci.*, vol. 34, no. 10, hlm. 8136–8151, Nov. 2022. doi: 10.1016/j.jksuci.2022.07.025.

[8] J. S. Teh dan A. Samsudin, "A chaos-based authenticated cipher with associated data," *Security and Communication Networks*, vol. 2017, 2017. doi: 10.1155/2017/9040518.

[9] I. H. Masri dan B. H. Susanti, "General Chaos Implementation as a Construction Element of Primitive Cryptography," dalam *Proc. 2023 IEEE Int. Conf. Ind. 4.0, Artif. Intell., and Commun. Technol. (IAICT)*, Bali, Indonesia, 2023, hlm. 168–174. doi: 10.1109/IAICT59002.2023.10205692.

[10] A. Doganaksoy, B. Ege, O. Kocak, dan F. Sulak, "Cryptographic Randomness Testing of Block Ciphers and Hash Functions," *IACR Cryptology ePrint Archive*, hlm. 564, 2010.

[11] M. S. Jawed dan M. Sajid, "Cryptanalysis of Lightweight Block Ciphers using Metaheuristic Algorithms in Cloud of Things (CoT)," dalam *Proc. 2022 Int. Conf. Data Anal. for Bus. and Ind. (ICDABI)*, Sakhir, Bahrain, 2022, hlm. 165–169. doi: 10.1109/ICDABI56818.2022.10041583.

[12] A. Al-Sabaawi, "Cryptanalysis of Stream Cipher: Method Implementation," dalam *Proc. 2021 IEEE Asia-Pacific Conf. Comput. Sci. and Data Eng. (CSDE)*, Brisbane, Australia, 2021, hlm. 1–4. doi: 10.1109/CSDE53843.2021.9718432.

[13] I. K. Y. Sucipta, B. H. Susanti, dan S. Siswantyo, "Algebraic Attack on LCB and Improved LCB," *2024 1st International Conference on Cyber Security and Computing (CyberComp)*, vol. 1, hlm. 150–154, 2024. doi: 10.1109/CyberComp60759.2024.10913608.

[14] K. Peffers, T. Tuunanen, M. A. Rothenberger, dan S. Chatterjee, "A design science research methodology for information systems research," *J. Manage. Inf. Syst.*, vol. 24, no. 3, hlm. 45–77, 2007. doi: 10.2753/MIS0742-1222240302.

[15] M. R dan N. K. V., "Optimized Implementation of S-box and Inverse S-box for PRESENT Lightweight Block Cipher," dalam *Proc. 2023 2nd Int. Conf. Vision Towards Emerging Trends in Commun. and Netw. Technol. (ViTECoN)*, Vellore, India, 2023, hlm. 1–5. doi: 10.1109/ViTECoN58111.2023.10156932.

[16] A. S. Qazi, A. H. Zahid, A. Baz, F. Arslan, M. Ali, dan J. Ali, "Innovative Transformation of S-Box Through Chaotic Map Using a Pragmatic Approach," *IEEE Access*, vol. 12, hlm. 42725–42736, 2024. doi: 10.1109/ACCESS.2024.3378731.