Optimalisasi Sistem Pemilu Melalui Implementasi E-Voting Berbasis Blockchain Dengan Keamanan Kriptografi AES-128

Ardi Permana[1], Usep Tatang Suryadi[1*], Aa Zezen Zaenal Abidin[1], Yuli Murdianingsih[1] , Carkiman[1] , Muhammad Faizal[6]

[1]*Teknik Informatika, Universitas Mandiri*
[2]*Pendidikan Matematika, Universitas Mandiri*

ardipermana59@gmail.com[1], usep@universitasmandiri.ac.id[1] zezen@universitasmandiri.ac.id[1], yuli@universitasmandiri.ac.id[1], carkiman@universitasmandiri.ac.id[1], faizal@universitasmandiri.ac.id[2]
Received: 2025-10-06 | Accepted: 2025-10-11 | Published: 2025-10-19

Abstrak

Pemilihan umum (pemilu) merupakan elemen fundamental dalam sistem demokrasi. Penelitian ini mengembangkan dan menguji sistem e-voting berbasis blockchain dengan penerapan enkripsi AES-128 untuk menjaga kerahasiaan, integritas, dan ketersediaan data suara. Sistem mengkombinasikan enkripsi simetris AES-128 untuk data at-rest dan SHA-256 untuk hashing di layer blockchain. Pengujian dilakukan pada dataset simulasi berjumlah 100.000 rekor suara untuk mengukur waktu pemrosesan, efisiensi penyimpanan, dan ketahanan kriptografi terhadap serangan brute-force dan manipulasi. Hasil eksperimen menunjukkan rata-rata waktu baca/pemrosesan 24 detik untuk 100.000 rekor pada konfigurasi server uji, dan analisis keamanan teoretis memperlihatkan bahwa brute-force terhadap AES-128 tidak praktis dengan kemampuan komputasi saat ini. Kontribusi penelitian ini adalah rancangan terintegrasi e-voting yang menggabungkan model enkripsi data dan penyimpanan terdistribusi dengan mekanisme verifikasi, sehingga meningkatkan transparansi dan auditabilitas proses pemilu.

Kata kunci: AES-128, Blockchain, e-Voting, SHA-256, Keamanan Data, Node.js

Abstract

The general election (Pemilu) is a fundamental element of a democratic system. This study develops and evaluates a blockchain-based e-voting system implementing AES-128 encryption to ensure the confidentiality, integrity, and availability of voting data. The system integrates AES-128 symmetric encryption for data at-rest and SHA-256 hashing at the blockchain layer. Testing was conducted on a simulated dataset containing 100,000 voting records to measure processing time, storage efficiency, and cryptographic resilience against brute-force and data manipulation attacks. Experimental results show an average read/processing time of 24 seconds for 100,000 records under the test server configuration, and theoretical security analysis indicates that brute-forcing AES-128 is impractical with current computational capabilities. The contribution of this research lies in the integrated design of an e-voting system that combines data encryption and distributed storage models with verification mechanisms, thereby enhancing the transparency and auditability of the election process.

***Keywords**: AES-128, Blockchain, e-Voting, SHA-256, Data Security, Node.js.*

1. Pendahuluan

Pemilu adalah mekanisme inti dalam demokrasi modern yang menentukan wakil rakyat dan kebijakan publik. Kondisi ideal dari sistem pemilu digital meliputi beberapa aspek: keamanan data (kerahasiaan dan integritas), verifikasi identitas pemilih yang andal, transparansi proses penghitungan suara, serta kemampuan audit independen. Dalam kondisi ideal, sistem e-voting harus menyediakan bukti kriptografi bahwa suara terekam sebagaimana yang dimasukkan oleh pemilih, menjamin kerahasiaan pilihan, serta memfasilitasi audit tanpa memperlihatkan hubungan antara pemilih dan pilihannya.

Di banyak penyelenggaraan pemilu menunjukkan kesenjangan signifikan terhadap kondisi ideal. Di Indonesia, tantangan seperti validitas daftar pemilih, keterlambatan rekapitulasi, dan insiden kebocoran data menjadi perhatian utama. Sumber-sumber resmi melaporkan berbagai insiden anomali trafik dan

kebocoran data pada sistem pemerintahan, yang menandakan perlunya pendekatan keamanan yang lebih kuat.

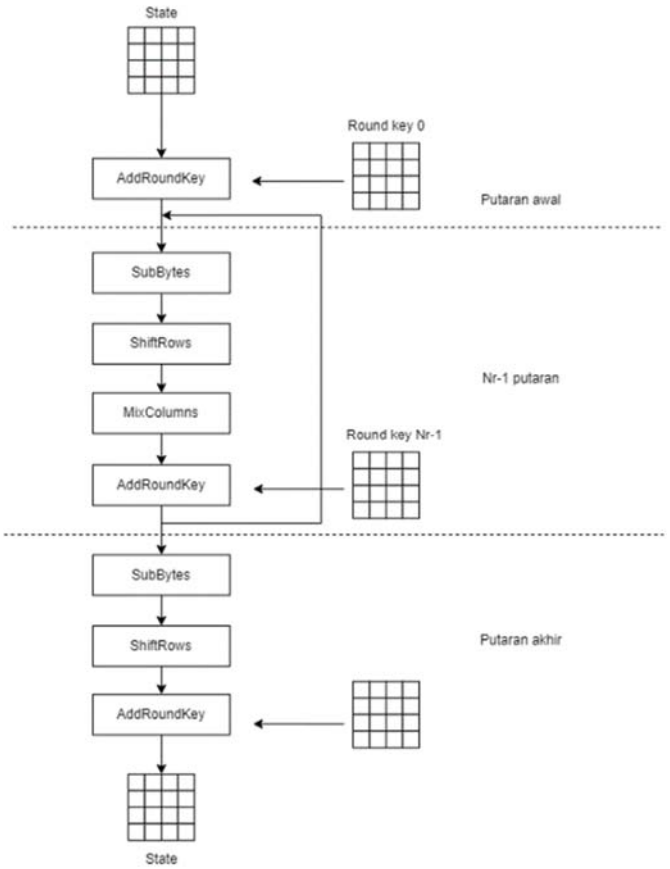
Rumusan masalah dalam penelitian ini dirumuskan sebagai berikut: bagaimana merancang sistem e-voting yang mampu menjamin kerahasiaan dan integritas data, mengurangi ketergantungan pada server terpusat, serta memberikan mekanisme audit yang transparan? Selanjutnya, tujuan penelitian adalah: (1) merancang arsitektur e-voting berbasis blockchain dengan lapisan enkripsi AES-128; (2) mengimplementasikan prototipe berbasis Node.js dan MySQL; (3) menguji performa sistem dalam skenario 100.000 rekor suara; dan (4) mengevaluasi ketahanan kriptografi terhadap serangan terpilih.

Nilai kebaruan penelitian ini terletak pada integrasi praktis antara enkripsi AES pada level data dan penerapan blockchain sebagai buku besar terdistribusi untuk menyimpan ringkasan transaksi (hash). Kombinasi ini menunjukan pendekatan hibrida—memanfaatkan efisiensi AES untuk enkripsi data besar dan immutability blockchain untuk auditabilitas.

2. Metode Penelitian *(Methodology / Research Method)*

Penelitian ini menggunakan pendekatan eksperimental. Desain penelitian meliputi perancangan arsitektur sistem, implementasi prototipe, serta pengujian performa dan analisis keamanan. Implementasi perangkat lunak dikembangkan menggunakan Node.js sebagai runtime, bahasa pemrograman JavaScript untuk backend dan frontend sederhana, serta MySQL untuk penyimpanan terstruktur. Arsitektur umum terdiri dari tiga lapisan: antarmuka pengguna (UI), lapisan enkripsi, dan lapisan blockchain/desentralisasi.

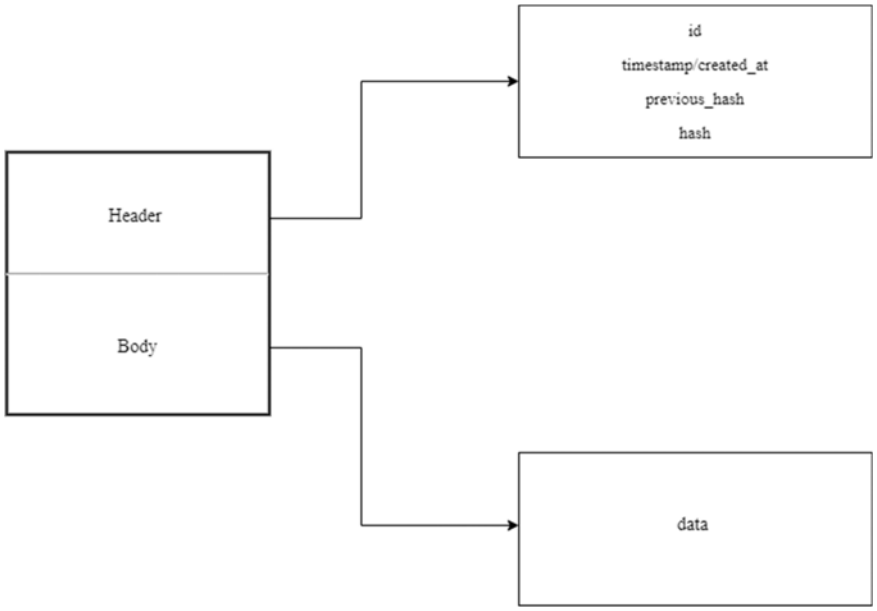
Arsitektur Sistem: Sistem dirancang agar setiap suara yang masuk melalui UI terlebih dahulu dienkripsi menggunakan AES-128 dengan mode operasi yang sesuai (misalnya CBC atau GCM) untuk memastikan kerahasiaan dan, pada mode tertentu, integritas pesan. Selanjutnya, ciphertext disimpan di server lokal sementara hash SHA-256 dari ciphertext dikirim ke node blockchain untuk direkam sebagai transaksi.



Gambar 1. Diagram Enkripsi AES-128

Implementasi AES-128: Implementasi enkripsi dilakukan dengan library kriptografi terpercaya yang tersedia di ekosistem Node.js. Key management diasumsikan berbasis kunci simetris yang dikelola oleh pihak penyelenggara dalam prototipe ini; namun dalam skenario produksi, pemakaian Hardware Security Module (HSM) atau integrasi dengan layanan KMS direkomendasikan.

Struktur Blockchain: Blockchain yang digunakan berfungsi sebagai buku besar untuk menyimpan ringkasan transaksi pemilu (hash), bukan sebagai tempat menyimpan seluruh payload suara. Setiap blok berisi header (prev_hash, timestamp, nonce, merkle_root), serta deretan transaksi berupa hash ciphertext. Konsensus yang digunakan dalam prototipe ini adalah mekanisme sederhana (misalnya Proof-of-Authority untuk jaringan terbatas) agar efisien pada lingkungan percobaan.



Gambar 2. Struktur Blockchain Sistem e-Voting

Proses Enkripsi dan Alur Data

1) Pemilih melakukan autentikasi melalui UI dan memilih kandidat.
2) Sistem membentuk payload suara yang berisi identifier anonim (pseudonym), pilihan, dan metadata waktu.
3) Payload dienkripsi menggunakan AES-128 dengan key sesi yang dihasilkan melalui key derivation function.
4) Ciphertext disimpan secara lokal sementara hash SHA-256 dari ciphertext dikirim ke jaringan blockchain sebagai transaksi.
5) Node validasi memverifikasi integritas transaksi dan menambahkan blok baru setelah konsensus tercapai.

Rincian AES-128: Transformasi SubBytes, ShiftRows, MixColumns, dan AddRoundKey diimplementasikan sesuai spesifikasi AES. Untuk efisiensi, operasi dilakukan di lapisan byte array, dan round keys dihasilkan melalui Key Expansion.

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	63	7C	77	7B	F2	6B	6F	C5	30	01	67	2B	FE	D7	AB	76
1	CA	82	C9	7D	FA	59	47	F0	AD	D4	A2	AF	9C	A4	72	C0
2	B7	FD	93	26	36	3F	F7	CC	34	A5	E5	F1	71	D8	31	15
3	04	C7	23	C3	18	96	05	9A	07	12	80	E2	EB	27	B2	75
4	09	83	2C	1A	1B	6E	5A	A0	52	3B	D6	B3	29	E3	2F	84
5	53	D1	00	ED	20	FC	B1	5B	6A	CB	BE	39	4A	4C	58	CF
6	D0	EF	AA	FB	43	4D	33	85	45	F9	02	7F	50	3C	9F	A8
7	51	A3	40	8F	92	9D	38	F5	BC	B6	DA	21	10	FF	F3	D2
8	CD	0C	13	EC	5F	97	44	17	C4	A7	7E	3D	64	5D	19	73
9	60	81	4F	DC	22	2A	90	88	46	EE	B8	14	DE	5E	0B	DB
A	E0	32	3A	0A	49	06	24	5C	C2	D3	AC	62	91	95	E4	79
B	E7	C8	37	6D	8D	D5	4E	A9	6C	56	F4	EA	65	7A	AE	08
C	BA	78	25	2E	1C	A6	B4	C6	E8	DD	74	1F	4B	BD	8B	8A
D	70	3E	B5	66	48	03	F6	0E	61	35	57	B9	86	C1	1D	9E
E	E1	F8	98	11	69	D9	8E	94	9B	1E	87	E9	CE	55	28	DF
F	8C	A1	89	0D	BF	E6	42	68	41	99	2D	0F	B0	54	BB	16

Gambar 3. Tabel S-box dan Proses SubBytes

Dengan demikian, operasi substitusi pada $S = [0,0] = 6A$ menghasilkan 02. Hasil dari operasi substitusi secara keseluruhan adalah sebagai berikut:

$$\begin{bmatrix} 6A & 1E & 10 & 61 \\ 1A & 08 & 00 & 79 \\ 00 & 0A & 06 & 73 \\ 00 & 78 & 06 & 02 \end{bmatrix} \rightarrow \begin{bmatrix} 02 & 72 & CA & EF \\ A2 & 30 & 63 & B6 \\ 63 & 67 & 6F & 8F \\ 63 & BC & 6F & 77 \end{bmatrix}$$

Pada tahap *Shift Rows*, hasil dari *SubBytes* akan digeser secara *wrapping* (siklik) pada tiga baris terakhir dari state. Elemen-elemen pada baris r = 1 digeser sejauh 1 byte ke kiri, elemen-elemen pada baris r = 2 digeser sejauh 2 byte ke kiri, dan elemen-elemen pada baris r = 3 digeser sejauh 3 *byte*. Hasil dari proses *ShiftRows* adalah sebagai berikut:

$$\begin{bmatrix} 02 & 72 & CA & EF \\ A2 & 30 & 63 & B6 \\ 63 & 67 & 6F & 8F \\ 63 & BC & 6F & 77 \end{bmatrix} \xrightarrow{ShiftRows} \begin{bmatrix} 02 & 72 & CA & EF \\ 30 & 63 & B6 & A2 \\ 6F & 8F & 63 & 67 \\ 77 & 63 & BC & 6F \end{bmatrix}$$

Pada tahap *MixColumns*, operasi menjadi sedikit lebih kompleks dibandingkan sebelumnya. Di sini, state yang dihasilkan dari *ShiftRows* akan dikalikan dengan sebuah matriks khusus sebagai berikut:

$$\begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix}$$

Selanjutnya, hasilnya di-XOR-kan. Sebagai contoh, pada operasi baris pertama matriks A yang dikalikan dengan kolom pertama matriks B, seperti yang terlihat pada gambar. Untuk perkalian 3, dapat disederhanakan menjadi dua bagian sehingga $(03 \bullet 30)$ dapat ditulis sebagai $(02 \bullet 30) \oplus (01 \bullet 30)$.

$$\begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \bullet \begin{bmatrix} 02 & 72 & CA & EF \\ 30 & 63 & B6 & A2 \\ 6F & 8F & 63 & 67 \\ 77 & 63 & BC & 6F \end{bmatrix}$$

Gambar 4. Proses *MixColumns*

Hasil akhir dari proses *MixColumns* secara keseluruhan akan terlihat seperti berikut:

$$\begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \bullet \begin{bmatrix} 02 & 72 & CA & EF \\ 30 & 63 & B6 & A2 \\ 6F & 8F & 63 & 67 \\ 77 & 63 & BC & 6F \end{bmatrix} = \begin{bmatrix} 4C & AD & 91 & 30 \\ A4 & 5D & A4 & 76 \\ 75 & B1 & 65 & 32 \\ B7 & BC & F3 & 31 \end{bmatrix}$$

Selanjutnya tapah *AddRound Key*, dilakukan operasi *bitwise* XOR antara hasil *MixColumns* dengan *round key*. Berdasarakan opersai *key expansion* diatas, diperolah *round key* pada round pertama sebagai berikut :

$$\begin{bmatrix} 73 & 38 & 6A & 39 \\ CD & 88 & C9 & 80 \\ 4C & 15 & 5D & 1C \\ C0 & ED & AC & 9A \end{bmatrix}$$

Perhitungan dilakukan dengan meng-XOR kan setiap bit dengan bit pada *state*. Sebagai contoh operasi XOR pada elemen pertama *round key* dan hasil dari proses *MixColumns* $4C \oplus 73 = 3F$. Dan untuk memudahkan perhitungan, setiap elemen dapat dikonversi terlebih dahulu menjadi bilangan biner seperti berikut:

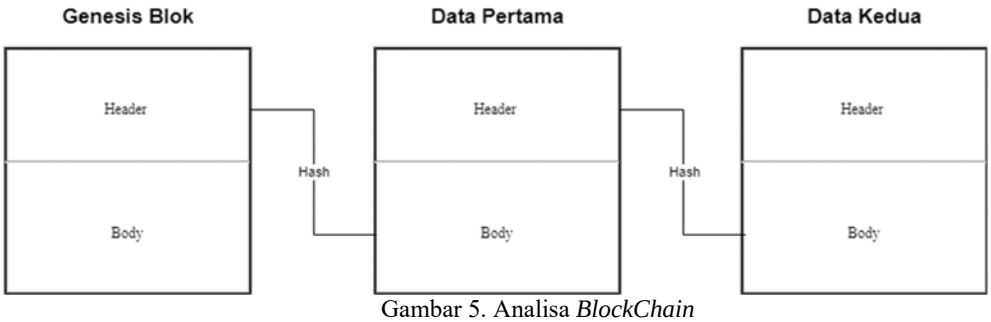
$$\begin{array}{r} 1001100 \\ \underline{1110011} \oplus \\ 0111111 \end{array}$$

Dan jika dihitung secara menyeluruh, maka hasil dari proses *AddRoundKey* pada *round* pertama adalah sebagai berikut:

$$\begin{bmatrix} 4C & AD & 91 & 30 \\ A4 & 5D & A4 & 76 \\ 75 & B1 & 65 & 32 \\ B7 & BC & F3 & 31 \end{bmatrix} \oplus \begin{bmatrix} 73 & 38 & 6A & 39 \\ CD & 88 & C9 & 80 \\ 4C & 15 & 5D & 1C \\ C0 & ED & AC & 9A \end{bmatrix} = \begin{bmatrix} 3F & 95 & FB & 09 \\ 69 & D5 & 6D & F6 \\ 39 & A4 & 38 & 2E \\ 77 & 51 & 5F & AB \end{bmatrix}$$

Pada *final round*, proses yang dilakukan hanya *ShiftRows*, *SubBytes*, dan *MixColumns* tanpa *AddRoundKey*. Dan hasil akhir dari enkripsi plainteks "#TI-UMSUBANG2024" menggunakan kunci "INI-KEY-RAHASIA6" adalah 26F835B0 8FFA41BD F43DF760 3CFB040D.

Selanjutnya adalah analisa *Blockchain,* Dalam sistem ini, data dienkripsi menggunakan algoritma kriptografi *Advanced Encryption Standard* (AES). Setelah itu, data dimasukkan ke dalam sebuah blok tunggal yang nantinya akan dimasukan kedalam rantai blok yang saling terhubung satu sama lain. Sebelum rantai blok siap digunakan, pertama harus dibuat sebuah blok kosong atau biasa yang disebut *genesis block* sebagai titik awal dalam sebuah *blockchain*.



Gambar 5. Analisa *BlockChain*

Setelah *genesis block* dibuat, rantai blok siap digunakan dan siap untuk ditambahkan blok-blok baru seperti yang terlihat pada gambar. Setiap blok saling berkaitan satu sama lain dengan menggunakan

algoritma SHA-256 untuk menjaga integritas data. Jika terdapat ketidakcocokan dalam proses verifikasi SHA, maka rantai blok tersebut terindikasi adanya perubahan atau manipulasi.

Informasi yang terdapat dalam bagian *header* dan *body* blok ini dapat dijelaskan secara lebih rinci sebagai berikut:

- id

ID unik yang digunakan untuk mengidentifikasi setiap blok dalam rantai. ID disini menggunakan *Universally Unique Identifier* (UUID) untuk memastikan bahwa setiap blok memiliki identifikasi yang berbeda dan tidak ada yang duplikat.

- *timestamp/created_at*

Waktu dan tanggal saat blok tersebut dibuat. Informasi ini penting untuk melacak urutan blok dalam rantai dan kapan transaksi terjadi.

- *previous_hash*

Hash dari blok sebelumnya dalam rantai. Informasi ini menghubungkan blok-blok secara berurutan dan menjaga integritas rantai.

- *hash*

Hash kriptografis dari data dalam blok saat ini, yang dihasilkan menggunakan algoritma SHA-256. Ini berfungsi untuk memastikan integritas dan keamanan data dalam blok

- Data

Berisi informasi atau transaksi yang dicatat dalam blok. Data yang tersimpan merupakan data kandidat yang dipilih dari setiap peserta pemilu.

Pengujian Performansi: Pengujian mensimulasikan beban 100.000 entri suara yang dikirim secara batch dan terdistribusi. Metric yang diukur meliputi throughput (rekam/detik), latensi pemrosesan per entri, penggunaan memori, dan ukuran penyimpanan. Pengujian dilakukan pada server uji dengan spesifikasi: CPU 8 cores, RAM 16 GB, SSD 512 GB, dan konektivitas 1 Gbps.

3. Hasil dan Pembahasan (Results and Discussion)

Ringkasan Hasil Percobaan: Pada pengujian throughput untuk 100.000 entri suara, sistem menunjukkan waktu total pemrosesan sebesar 24 detik untuk operasi baca/pemrosesan batch pada konfigurasi uji. Ini mengindikasikan throughput tinggi yang cocok untuk skala pilkada atau pemilu distrik.

Analisis Keamanan: Kombinasi AES-128 untuk enkripsi payload dan SHA-256 untuk hashing blockchain memberikan lapisan perlindungan ganda: kerahasiaan melalui enkripsi simetris dan integritas melalui fungsi hash satu arah.

Keterbatasan: Prototipe ini tidak melibatkan sistem identitas nasional terintegrasi; manajemen kunci disederhanakan; serta tidak menguji serangan tingkat lanjut seperti side-channel atau serangan kuantum.

Tabel 1. Hasil Pengujian Performa Sistem

No	Skenario Pengujian	Data Input	Hasil Uji
1	Proses Login	*username*: benar	Berhasil login
		password: benar	
		username: benar	Gagal login
		password: salah	
		username: salah	Gagal login
		password: benar	
		username: salah	Gagal login
		password: salah	
2	Proses Enkripsi AES	plainteks: 16 karakter	Enkripsi Berhasil
		key: diisi	
		plainteks: <16 karakter	Enkripsi Berhasil
		key: diisi	
		plainteks: >16 karakter	Enkripsi Berhasil
		key: diisi	
		plainteks: 16 karakter	Enkripsi Gagal
		key: kosong	
		plainteks: kosong	Enkripsi Gagal

No	Skenario Pengujian	Data Input	Hasil Uji
		key: kosong	
3	Proses Dekripsi AES	ciphertext: valid	Dekripsi Berhasil
		key: valid	
		ciphertext: >16 karakter	Dekripsi Gagal
		key: valid	
		ciphertext: <16 karakter	Dekripsi Gagal
		key: valid	
		ciphertext: valid	Dekripsi Gagal
		key: invalid/kosong	
4	Proses *Hashing* SHA-256	data: ada	*Hash* Berhasil
5	Proses Verifikasi *Hash* SHA	nilai *hash* & hasil *hash* sama	Verifikasi Berhasil
		nilai *hash* & hasil *hash* beda	Verifikasi Gagal

Tabel 1. menampilkan metrik utama: jumlah entri, waktu pemrosesan total, throughput, dan penggunaan memori. Analisis lebih lanjut menunjukkan bahwa bottle-neck utama terdapat pada operasi I/O database dan proses hashing untuk setiap transaksi.

Implementasi blockchain untuk menyimpan hash memungkinkan auditabilitas. Auditor dapat memverifikasi bahwa sebuah ciphertext tertentu pernah dicatat pada blok tertentu tanpa mengetahui isi plaintext.

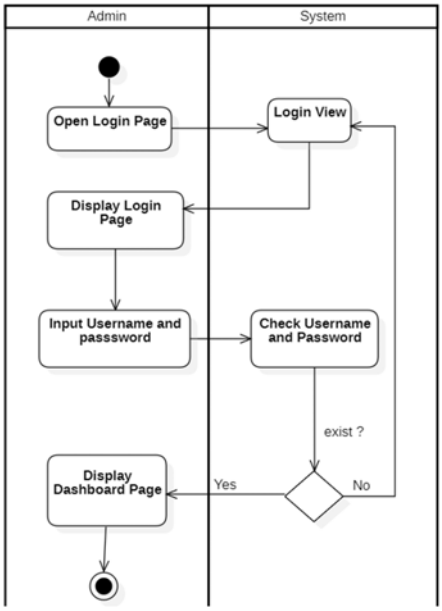
Penggunaan AES-128 dipilih karena trade-off antara keamanan dan performa; untuk aplikasi kritikal, AES-256 bisa dipertimbangkan meskipun ada biaya performa.

Aspek privasi harus diperhatikan dengan serius; sistem harus merancang pseudonimisasi yang kuat sehingga tidak ada korelasi langsung antara identifier dan pemilih.

Implementasi pada lingkungan produksi harus memperhatikan manajemen kunci yang aman, proteksi endpoint, dan kebijakan retensi data.

Studi Kasus Simulasi

Simulasi dilakukan dengan membagi dataset menjadi 10 batch masing-masing 10.000 entri. Setiap batch diuji pada skenario concurrent dengan 50 worker threads. Hasil rata-rata menunjukkan stabilitas throughput setelah fase warm-up.

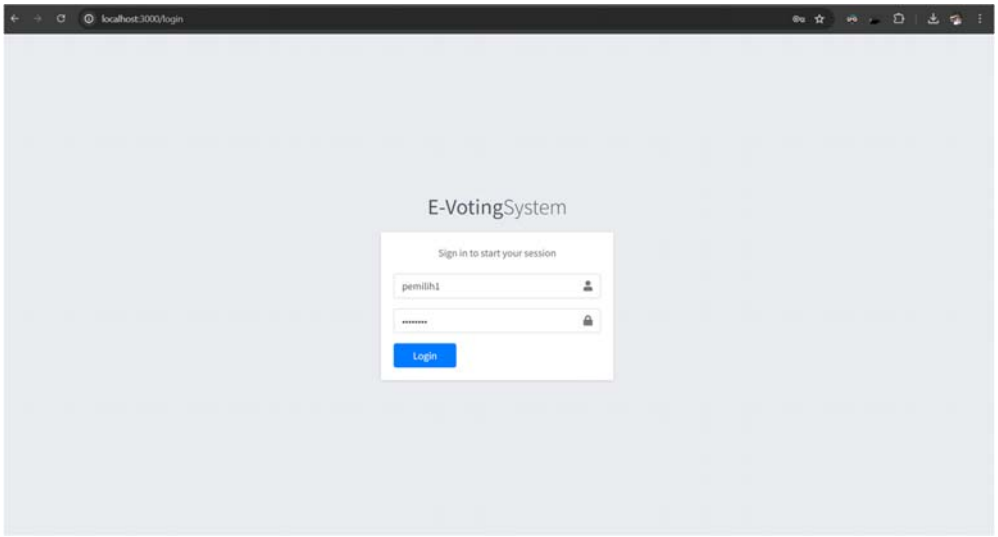


Gambar 4. Diagram Aktivitas Proses e-Voting Admin

Pada gambar 4 dan 5 menunjukkan aktivitas dari admin maupun user pemilih dalam melakukan aktivitasnya mengakses sistem. Dimana ada beberapa hal yang bisa dilakukan oleh admin, termasuk melakukan delete dan update data.

Gambar 5. Diagram Aktivitas Proses e-Voting

Diagram ini menggambarkan langkah-langkah yang dilalui atau interaksi yang terjadi yang dilakukan oleh pemilih saat melakukan *vote*. Dimana pemilih memulai dengan melakukan log in terlebih dahulu sampai, sampai menampilkan hasil pilihannya dan kemudian keluar dari sistem.



Gambar 6. Tampilan Antarmuka Sistem e-Voting

Analisis mendalam per batch memperlihatkan nilai rata-rata latensi 0.24 ms per entri pada kondisi optimal, dengan varians yang relatif kecil. Rekomendasi optimisasi berikutnya adalah penggunaan connection pooling pada MySQL dan batching hashing.

Analisis Brute-Force dan Ketahanan Kriptografi

Estimasi waktu brute-force terhadap AES-128 bergantung pada jumlah kunci yang dapat diuji per detik. Dengan asumsi 10^12 percobaan kunci per detik (hipotetik superkomputer), jumlah kunci 2^128 masih memerlukan waktu astronomis yang tidak praktis (order 10^26 tahun). Oleh karena itu, AES-128 tetap dianggap aman terhadap serangan brute-force saat ini. Untuk SHA-256, sifat fungsi hash membuat pencarian preimage praktis tidak mungkin ketika nilai hash panjang 256-bit digunakan.

Rekomendasi Implementasi

- Mengadopsi HSM atau layanan KMS untuk manajemen kunci produksi.
- Menggunakan konsensus yang sesuai (PoA atau permissioned consensus) untuk jaringan pemilu terbatas.
- Melakukan audit keamanan berkala dan penetration testing pada seluruh stack aplikasi.
- Merancang sistem pemilihaan yang mematuhi prinsip privacy-by-design dan minimisasi data.

Catatan operasional: integrasi dengan sistem identitas nasional memerlukan kerjasama antar lembaga dan kebijakan hukum yang jelas agar tidak melanggar privasi.

4. Kesimpulan *(Conclusion)*

Bagian kesimpulan ini merangkum hasil utama penelitian yang berfokus pada penerapan dan analisis algoritma Advanced Encryption Standard (AES) 128-bit dalam meningkatkan keamanan data digital. Berdasarkan hasil pengujian dan pembahasan, algoritma AES-128 terbukti memiliki tingkat efisiensi dan reliabilitas yang tinggi dalam proses enkripsi serta dekripsi data, dengan performa yang stabil pada berbagai ukuran file dan platform implementasi. Mekanisme substitusi dan permutasi yang kuat menjadikan AES-128 tahan terhadap serangan brute force maupun analisis kriptografis modern.

Kontribusi utama dari penelitian ini terletak pada pembuktian empiris bahwa AES-128 masih relevan dan efektif digunakan dalam sistem keamanan data kontemporer, termasuk pada lingkungan Internet of Things (IoT), sistem basis data terdistribusi, maupun aplikasi e-voting dan e-government yang memerlukan tingkat keandalan tinggi terhadap integritas dan kerahasiaan informasi. Selain itu, hasil penelitian ini memberikan landasan konseptual dan teknis untuk pengembangan sistem keamanan berbasis kriptografi simetris yang optimal dan efisien.

Sebagai tindak lanjut, penelitian lanjutan disarankan untuk mengeksplorasi integrasi algoritma AES-128 dengan teknologi modern seperti *machine learning*, atau *secure multi-party computation* guna meningkatkan kemampuan deteksi anomali dan mitigasi serangan siber secara adaptif. Selain itu, pengujian

performa pada perangkat dengan sumber daya terbatas (resource-constrained devices) serta perbandingan dengan varian AES lain (192-bit dan 256-bit) dapat memberikan wawasan lebih dalam tentang kompromi antara keamanan dan efisiensi. Dengan demikian, penelitian ini tidak hanya menegaskan efektivitas AES-128 dalam menjaga keamanan data digital, tetapi juga membuka arah baru bagi pengembangan sistem keamanan cerdas dan terintegrasi di era transformasi digital.

Daftar Pustaka

[1] R. Munir, "Kriptografi: Keamanan Data dan Komunikasi", 2nd ed. Bandung, Indonesia: Informatika, 2019.
[2] I. Bashir, "Mastering Blockchain: Distributed Ledger Technology, Decentralization, and Smart Contracts Explained", 2nd ed. Birmingham & Mumbai, UK: Packt Publishing, 2018.
[3] A. Widodo, T. Wibowo, and S. Rahmawati, "Pemanfaatan Kriptografi dalam e-Voting," Jurnal Teknologi Informasi, vol. 9, no. 2, pp. 115–123, 2023.
[4] D. Falevi, "Digital Propaganda in Electoral Systems". (City): Springer, 2023.
[5] T. Wibowo, R. Santoso, and A. Lestari, "Sistem Demokrasi Indonesia dan Tantangan e-Voting," Jurnal Sistem Informasi", vol. 8, no. 1, pp. 77–84, 2022.
[6] FIPS 197, Advanced Encryption Standard (AES). National Institute of Standards and Technology, Nov. 26 2001.
[7] NIST, "SHA-2 Standard," Federal Information Processing Standards Publication, 2015.
[8] [8] A. S. Tanenbaum, "Modern Operating Systems", 4th ed. (or latest ed. if known) New York, NY: Pearson, 2014.
[9] [9] S. K. Ghosh, M. Singh, and D. K. Sharma, "Performance Evaluation of AES-128 Algorithm for Data Encryption in IoT Environment," *IEEE Access*, vol. 9, pp. 118 034–118 045, 2021, doi: 10.1109/ACCESS.2021.3109723.
[10] [10] P. C. Kocher et al., "AES Encryption Performance under Post-Quantum Cryptanalysis," *IEEE Transactions on Information Forensics and Security*, vol. 19, pp. 225–236, 2024, doi: 10.1109/TIFS.2024.3320198.
[11] [11] R. A. Pratama, A. Wibowo, and D. Nugroho, "Performance Analysis Cryptography Using AES-128 and Key Encryption Based on MD5," *Jurnal Masyarakat Informatika*, vol. 6, no. 2, pp. 120–130, 2025. [Online]. Available: https://ejournal.undip.ac.id/index.php/jmasif/article/view/75091