https://jurnal.pustakagalerimandiri.co.id/index.php/pustakadata
DOI: https://doi.org/10.55382/jurnalpustakadata.v6i4.2206

Jurnal Pustaka Data
Pusat Akses Kajian Database, Analisa Teknologi, dan Arsitektur Komputer
Vol. 6 No. 4 (2026) 468– 484
E ISSN: 2809-7483

Pengembangan Sistem Warisan Digital Ringan Berbasis Blockchain dengan Kriptografi Hibrida

Beben Prihartono[1], Bambang Widodo[2], Taufik Iqbal Ramdhani[3]
[1,2,3] Teknik Informatika, Institut Teknologi Tangerang Selatan, Indonesia
[1]1003250031@students.itts.ac.id

Abstract

Digital asset inheritance presents an unresolved trifunctional security challenge: administrator-resilient confidentiality, long-term data integrity, and cryptographically verifiable temporal availability. Existing solutions—public blockchain platforms and commercial cloud digital legacy services—fail to simultaneously satisfy all three dimensions. This paper presents Catena Memoriae, a digital inheritance system prototype integrating a lightweight SQLite-backed local blockchain with AES-256-CBC/RSA-2048 hybrid encryption and a PBKDF2-HMAC-SHA256 key derivation mechanism employing 1,000,000 iterations. The implemented zero-knowledge server architecture guarantees that the server is cryptographically incapable of decrypting message payloads regardless of unrestricted database access. Empirical evaluation demonstrates AES-256 encryption of a 1 MB payload in 4.90 ms and 500-block chain verification in 3.08 ms—both well below the 100 ms human perceptual threshold. Black-box testing across 13 critical scenarios achieved 100% pass rate, and database tampering simulations validated the verify_chain() detection algorithm. The novelty lies in the cohesive integration of four mechanisms—lightweight ledger, zero-knowledge key derivation, scheduler-based time-lock, and tamper detection—within a single minimally deployable architecture, addressing a gap unserved by existing solutions.

Keywords: local blockchain, hybrid cryptography, digital inheritance, pbkdf2, zero-knowledge server

Abstrak

Manajemen warisan aset digital menghadapi tantangan pada tiga dimensi keamanan yang belum terpecahkan secara komprehensif, yaitu kerahasiaan (*confidentiality*) dari administrator sistem, integritas (*integrity*) data jangka panjang, dan ketersediaan temporal (*temporal availability*) yang terverifikasi secara kriptografis. Solusi yang ada saat ini, baik *blockchain* publik maupun platform warisan digital komersial berbasis komputasi awan (*cloud*), gagal memenuhi ketiga dimensi tersebut secara simultan. Penelitian ini mengusulkan Catena Memoriae, sebuah prototipe sistem warisan digital yang mengintegrasikan *blockchain* lokal ringan berbasis SQLite dengan enkripsi hibrida AES-256-CBC/RSA-2048 serta mekanisme derivasi kunci PBKDF2-HMAC-SHA256 sebanyak 1.000.000 iterasi. Arsitektur peladen tanpa pengetahuan (*zero-knowledge server*) yang diimplementasikan memastikan bahwa peladen secara kriptografis tidak dapat mendekripsi muatan pesan meskipun memiliki akses penuh terhadap basis data. Evaluasi empiris menunjukkan bahwa enkripsi AES-256 pada muatan 1 MB dapat diselesaikan dalam 4,90 milidetik (ms) dan verifikasi rantai 500 blok hanya memerlukan waktu 3,08 ms. Kedua waktu tersebut berada jauh di bawah ambang batas persepsi pengguna, yakni 100 ms [1]. Pengujian kotak hitam (*black-box*) terhadap 13 skenario kritis mencapai tingkat keberhasilan 100%, dan simulasi serangan perusakan data (*tampering*) memvalidasi efektivitas algoritma *verify_chain()*. Kebaruan penelitian ini terletak pada integrasi kohesif empat mekanisme, yaitu buku besar ringan (*lightweight ledger*), derivasi kunci tanpa pengetahuan (*zero-knowledge key derivation*), penguncian waktu berbasis penjadwalan (*scheduler-based time-lock*), dan deteksi perusakan (*tamper detection*). Keempat mekanisme tersebut disatukan dalam satu arsitektur yang dapat diterapkan pada infrastruktur minimal, sehingga mampu mengisi celah yang belum terlayani oleh solusi-solusi yang ada saat ini.

Submitted : 26-06-2026 | Reviewed : 30-06-2026 | Accepted : 07-08-2026

Kata kunci: blockchain lokal, kriptografi hibrida, warisan digital, pbkdf2, zero-knowledge server

1. Pendahuluan

1.1 Latar Belakang dan Masalah Penelitian

Percepatan adopsi aset digital dalam satu dekade terakhir telah mengubah cara masyarakat memandang kepemilikan informasi, dari yang sebelumnya berbasis media fisik menjadi sepenuhnya berbentuk digital dan tidak berwujud. Laporan Digital 2024 yang diterbitkan oleh We Are Social mencatat lebih dari 5,4 miliar pengguna internet aktif di seluruh dunia, dengan rata-rata setiap individu memiliki lebih dari 100 akun digital, ditambah jumlah dokumen, foto, dan aset keuangan digital yang terus bertambah [2]. Berbeda dengan aset fisik yang pewarisannya telah diatur oleh kerangka hukum yang mapan, aset digital belum memiliki mekanisme pewarisan yang baku. Kematian mendadak pemilik akun dapat menyebabkan hilangnya akses secara permanen terhadap dokumen hukum, kenangan keluarga, aset kripto, dan pesan wasiat yang hanya tersimpan dalam bentuk digital [3].

Tantangan dalam pewarisan aset digital tidak sekadar masalah penyimpanan data. Sistem warisan digital yang andal harus memenuhi tiga dimensi keamanan yang saling berkaitan secara bersamaan. Dalam literatur keamanan informasi, ketiga dimensi ini dikenal sebagai perluasan konsep segitiga CIA (*extended CIA triad*) dalam konteks komunikasi yang dititipkan kepada pihak ketiga (*escrowed communication*) [4], yaitu: (1) **kerahasiaan**, yaitu isi pesan harus terlindungi dari seluruh pihak yang tidak berwenang, termasuk administrator sistem itu sendiri; (2) **integritas**, yaitu keaslian dan kelengkapan pesan harus dapat diverifikasi secara independen kapan pun antara waktu pembuatan dan waktu pembukaan pesan; dan (3) **ketersediaan temporal**, yaitu pesan harus dapat diakses tepat pada waktu yang telah ditentukan oleh pengirim, tidak boleh lebih awal maupun lebih lambat

1.2 State of the Art: Kajian Solusi yang Ada

Kajian terhadap solusi-solusi yang ada menunjukkan bahwa tidak ada sistem yang berhasil memenuhi ketiga dimensi keamanan tersebut secara simultan. Pemahaman atas celah ini merupakan fondasi dari arsitektur yang diajukan dalam penelitian ini.

1.2.1 Platform Blockchain Publik

Blockchain, sebagaimana diperkenalkan oleh Nakamoto [5], menawarkan dua properti fundamental yang relevan untuk warisan digital: immutability melalui chained hash structure dan verifiability melalui distributed consensus [6]. Namun, penerapan blockchain publik pada domain warisan digital menghadapi konflik struktural yang mendasar.

Hossain et al. [7] mengusulkan sistem smart contract Ethereum untuk manajemen warisan digital dengan mekanisme multipihak yang mencegah akses prematur. Meskipun berhasil mendemonstrasikan feasibility smart contract untuk time-lock inheritance, terdapat tiga keterbatasan kritis: pertama, seluruh metadata pesan bersifat publik on-chain—kontradiksi fundamental dengan prinsip privasi warisan digital; kedua, biaya gas Ethereum yang volatil menjadikan operasi rutin tidak dapat diprediksi secara finansial; ketiga, ketergantungan penuh pada ketersediaan jaringan Ethereum mengekspos sistem terhadap downtime, congestion, dan risiko hard fork.

Li et al. [8] mengembangkan arsitektur blockchain-IPFS untuk penyimpanan dokumen warisan terdesentralisasi, mengatasi keterbatasan on-chain storage dari pendekatan Ethereum murni. Namun, integrasi dual-infrastructure yang dipersyaratkan—node blockchain ditambah IPFS daemon—meningkatkan kompleksitas deployment dan memperkenalkan dependensi jaringan berlapis yang tidak dapat dieliminasi. Wood [9] memperkenalkan Ethereum dengan kemampuan smart contract Turing-complete, sementara Androulaki et al. [10] mempresentasikan Hyperledger Fabric sebagai alternatif permissioned untuk konteks enterprise. Meskipun demikian, infrastruktur multi-organisasi yang dipersyaratkan menjadikannya tidak feasible bagi individu maupun institusi tanpa tim IT yang memadai.

Pongnumkul et al. [11] melakukan analisis kinerja terhadap platform blockchain privat dalam berbagai beban kerja dan mengonfirmasi bahwa latensi per transaksi pada implementasi privat dapat ditekan di bawah 10 ms pada beban rendah. Temuan ini mendukung desain pilihan Catena Memoriae: dengan mengeliminasi overhead konsensus jaringan, sistem mampu mencapai waktu verifikasi rantai 3,08 ms untuk 500 blok.

DOI: https://doi.org/10.55382/jurnalpustakadata.v6i4.2206

1.2.2 Platform Digital Legacy Komersial Berbasis Cloud

Layanan digital legacy komersial—seperti Safe Beyond, Willing, dan Everplans—menyediakan mekanisme pengiriman pesan berbasis waktu dengan antarmuka yang ramah pengguna, tetapi mengadopsi model kepercayaan terpusat. Islam et al. [12] mengonfirmasi melalui systematic review bahwa mayoritas layanan cloud digital legacy tidak mengimplementasikan zero-knowledge architecture: server platform menyimpan baik data terenkripsi maupun kunci enkripsi, sehingga privasi pengguna bergantung sepenuhnya pada integritas vendor terhadap ancaman insider threat, legal subpoena, dan serangan siber eksternal.

Password manager modern seperti 1Password dan Bitwarden mengimplementasikan zero-knowledge architecture yang lebih ketat melalui paradigma client-side encryption [13], di mana kunci dekripsi tidak pernah meninggalkan perangkat pengguna. Namun, desain ini dioptimalkan untuk akses real-time, bukan untuk pewarisan temporal: tidak ada mekanisme native yang memungkinkan kunci tertentu hanya dapat diakses oleh beneficiary tertentu setelah kondisi temporal terpenuhi. Emergency access feature yang ada pada beberapa password manager bergantung pada persetujuan manual, bukan time-lock otomatis yang terverifikasi secara kriptografis. Catena Memoriae menjembatani celah ini dengan menggabungkan zero-knowledge key derivation dari paradigma password manager dengan time-lock dari paradigma digital legacy platform.

1.2.3 Kriptografi Hibrida AES-RSA dan Derivasi Kunci PBKDF2

Advanced Encryption Standard (AES) yang ditetapkan NIST [14] merupakan standar enkripsi simetris yang telah bertahan lebih dari dua dekade kriptanalisis intensif. Dalam mode Cipher Block Chaining (CBC), AES-256 menghasilkan difusi statistik yang kuat karena setiap blok ciphertext bergantung pada seluruh blok sebelumnya, dengan throughput di atas 1 GB/s pada implementasi dengan instruksi AES-NI [15]. Rivest, Shamir, dan Adleman [16] memperkenalkan kriptosistem kunci publik RSA-2048 yang hingga saat ini dianggap aman terhadap serangan faktorisasi berdasarkan rekomendasi NIST SP 800-57 [17].

Tabel 1. Pemetaan Penelitian Terkait terhadap Dimensi Keamanan Warisan Digital

PBKDF2, sebagaimana didefinisikan dalam RFC 2898 [18] dan direkomendasikan dalam NIST SP 800-132 [19], merupakan mekanisme key stretching yang dirancang untuk membuat brute-force attack secara komputasional prohibitif. OWASP merekomendasikan minimum 600.000 iterasi untuk PBKDF2-HMAC-SHA256 pada sistem produksi 2024 [20]; Catena Memoriae menggunakan 1.000.000 iterasi, melampaui rekomendasi tersebut untuk margin keamanan tambahan. Catena Memoriae mengimplementasikan varian yang tidak biasa dari paradigma hybrid encryption: kunci AES tidak diproteksi oleh RSA publik penerima, melainkan oleh kunci turunan PBKDF2 dari faktor pengetahuan penerima. Keputusan desain ini secara langsung menghasilkan properti zero-knowledge server tanpa

memerlukan infrastruktur PKI eksternal. Penelitian Juels dan Ristenpart [21] tentang honey encryption mengingatkan bahwa untuk domain input dengan entropi sangat rendah, PBKDF2 tidak memberikan jaminan keamanan yang mutlak; keterbatasan ini diakui secara eksplisit dan didokumentasikan dalam Bagian 3.8.2.

1.2.4 Time-Lock Kriptografis

Konstruksi time-lock kriptografis murni, sebagaimana diperkenalkan oleh May [22] dan diformalisasi melalui timed-release encryption, menghasilkan jaminan temporal yang tidak bergantung pada kepercayaan terhadap pihak ketiga mana pun. Boneh dan Naor [23] mengeksplorasi timed-release encryption berbasis pairing yang lebih efisien; namun, implementasinya memerlukan infrastruktur trustee atau beacon yang menghasilkan dependensi eksternal. Penelitian ini mengadopsi pendekatan pragmatis: time-lock diimplementasikan melalui APScheduler yang memantau kondisi unlock_time—sebuah trade-off yang disengaja antara kemurnian kriptografis dan keandalan sistem, serta eliminasi total dependensi terhadap infrastruktur eksternal.

1.3 Kesenjangan Penelitian

Tinjauan literatur di atas mengidentifikasi celah penelitian yang signifikan pada persimpangan tiga domain: lightweight blockchain, knowledge-based cryptography, dan verified digital inheritance. Tabel 1 merangkum posisi Catena Memoriae dalam konteks penelitian terkait berdasarkan dimensi-dimensi keamanan kunci, mempertegas bahwa tidak ada sistem dalam literatur yang diketahui secara simultan memenuhi kelima kriteria tersebut.

Submitted : 26-06-2026 | Reviewed : 30-06-2026 | Accepted : 07-08-2026

Penelitian / Sistem	Zero-Know. Server	Tamper Detect.	Time-Lock Otomatis	Infra. Minimal	Open Source
Hossain et al. [7] (Ethereum)	Tidak	Ya (konsensus)	Smart contract	Tidak	Parsial
Li et al. [8] (Blockchain-IPFS)	Tidak	Ya (konsensus)	Tidak	Tidak	Parsial
Safe Beyond / Cloud	Tidak	Audit log saja	Date-triggered	Ya (SaaS)	Tidak
Legacy [12]					
1Password / Bitwarden [13]	Ya	Tidak	Tidak	Ya (SaaS)	Parsial
Catena Memoriae (Penelitian ini)	Ya	Ya (SHA-256 lokal)	Ya (APScheduler)	Ya (self-host)	Ya (MIT)

Secara spesifik, belum ada penelitian yang mengintegrasikan (1) lightweight local blockchain, (2) zero-knowledge key derivation berbasis PBKDF2, (3) scheduler-based time-lock, dan (4) tamper detection SHA-256 dalam satu arsitektur kohesif yang dapat di-deploy pada infrastruktur minimal tanpa biaya operasional eksternal. Catena Memoriae mengisi celah spesifik ini.

1.4 Kontribusi Penelitian

Penelitian ini memberikan empat kontribusi terhadap domain keamanan siber dan warisan digital, sebagaimana dirangkum dalam Tabel 2.

Tabel 2. Ringkasan Kontribusi Penelitian

No.	Kontribusi	Mekanisme Teknis	Signifikansi
C1	Arsitektur Zero-Knowledge Server	PBKDF2-HMAC-SHA256 (10^6 iterasi); kunci AES tidak pernah disimpan plaintext	Server tidak dapat mendekripsi pesan meskipun dikompromikan sepenuhnya
C2	Bukti Empiris Deteksi Tamper	SHA-256 chained ledger; verify_chain() dengan cascade detection	Validasi eksperimental 100% detection rate tanpa konsensus terdistribusi
C3	Benchmark Kriptografi Ringan (arm64)	time.perf_counter(), 50 iterasi/konfigurasi; AES + blockchain timing	Baseline empiris untuk penelitian deployment lightweight crypto
C4	Analisis Trade-off Arsitektural	Perbandingan 5 platform × 10 kriteria; STRIDE threat model	Kuantifikasi posisi lightweight local blockchain dalam ruang solusi

1.5 Kebaruan dan Ruang Lingkup

DOI: https://doi.org/10.55382/jurnalpustakadata.v6i4.2206

Kebaruan Catena Memoriae terletak bukan pada komponen-komponen individualnya—AES, RSA, PBKDF2, dan SHA-256 adalah primitif kriptografis yang telah lama dikenal—melainkan pada integrasi arsitektural koherennya dalam konteks warisan digital yang dapat di-deploy secara minimal. Secara spesifik, kombinasi zero-knowledge server melalui PBKDF2-based key derivation dengan SHA-256 chained integrity verification dan scheduler-based time-lock dalam satu Flask application merupakan konfigurasi yang belum pernah dieksplorasi dalam literatur sebelumnya.

Penelitian dibatasi pada implementasi dan evaluasi prototipe lokal berbasis Flask (Python 3.10), SQLite 3.51.1, dan PyCryptodome 3.19.0; penelitian ini tidak mencakup implementasi blockchain terdesentralisasi, protokol konsensus terdistribusi, atau integrasi jaringan publik. Sisa makalah ini disusun sebagai berikut: Bagian 2 memaparkan metodologi secara teknis; Bagian 3 menyajikan hasil pengujian beserta pembahasannya; Bagian 4 menyimpulkan penelitian.

2. Metode Penelitian

2.1 Arsitektur Sistem

Catena Memoriae mengadopsi arsitektur berorientasi layanan (Service-Oriented Architecture/SOA) berlapis yang memisahkan concern antara empat layanan inti yang berjalan dalam satu proses Flask. Pemisahan ini memiliki implikasi keamanan konkret: domain akses data setiap layanan terdefinisi secara eksplisit, sehingga kompromi yang terisolasi pada satu layanan tidak secara otomatis memberikan akses ke seluruh permukaan kriptografis sistem.

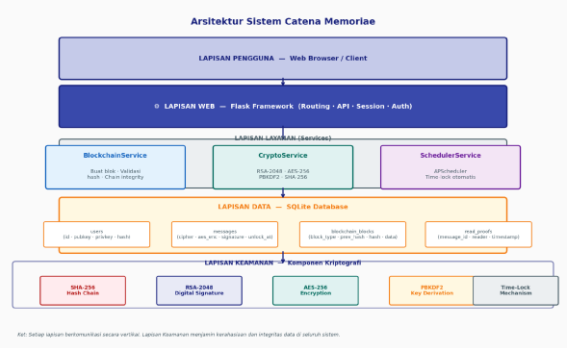
CryptoService: menangani seluruh operasi kriptografis (RSA, AES, PBKDF2, SHA-256).

BlockchainService: mengelola ledger lokal per pengguna, termasuk pembuatan dan verifikasi rantai blok.

EmailService: mengirimkan notifikasi SMTP saat kondisi time-lock terpenuhi.

SchedulerService: memantau kondisi time-lock menggunakan APScheduler dan memicu transisi status.

CryptoService tidak pernah secara langsung memanggil operasi database, memastikan bahwa kunci kriptografis yang berada di memori selama operasi derivasi PBKDF2 tidak dapat dikapitalisasi melalui serangan pada lapisan persistensi. Seluruh state persisten disimpan dalam SQLite 3.51.1 dengan mode WAL (Write-Ahead Logging) yang aktif untuk mencegah bottleneck konkurensi pada operasi baca.



Gambar 1. Diagram Arsitektur SOA Catena Memoriae: komponen Flask, CryptoService, BlockchainService, EmailService, SchedulerService.

2.2 Alur Kriptografis: Penyegelan Pesan (Message Sealing)

2.2.1 Notasi Formal

Skema kriptografis penyegelan pesan dinyatakan secara formal melalui persamaan berikut. Misalkan M adalah plaintext pesan, S adalah family secret penerima, dan TS adalah unlock timestamp.

(1) $K\_AES \leftarrow \beta(32)$ // Kunci AES-256 acak, 32 byte via CSPRNG

(2) $IV_1 \leftarrow \beta(16)$ // Inisialisasi vektor CBC, 16 byte

(3) $C\_msg = \text{AES-256-CBC}(K\_AES, IV_1, M \parallel \text{PKCS7}(M))$

(4) $\sigma \leftarrow \beta(16)$ // Salt PBKDF2, 16 byte

(5) $K\_pbk = \text{PBKDF2}(\text{HMAC-SHA256}, S, \sigma, 10^6, 32)$

(6) $IV_2 \leftarrow \beta(16)$

(7) $C\_key = \text{AES-256-CBC}(K\_pbk, IV_2, K\_AES)$

(8) $H = \text{SHA-256}(msg\_id \parallel metadata)$

(9) $B\_n.hash = \text{SHA-256}(\text{JSON-sorted}(\{uid, n, t, H, Sig, B\_\{n-1\}.hash\}))$

$\beta(n)$ menunjukkan pembangkitan n byte acak kriptografis via CSPRNG; JSON-sorted menunjukkan serialisasi JSON deterministik dengan sort_keys=True. Proteksi K_AES melalui K_pbk (persamaan 7) alih-alih kunci RSA penerima secara langsung menghasilkan properti zero-knowledge server: server tidak pernah menyimpan K_pbk

Submitted : 26-06-2026 | Reviewed : 30-06-2026 | Accepted : 07-08-2026

dalam bentuk plaintext—hanya C_key, σ, dan hitungan iterasi yang disimpan. Tanpa S yang hanya diketahui penerima, K_pbk tidak dapat direkonstruksi. Konstruksi chaining blok (persamaan 9) memastikan bahwa modifikasi apa pun pada B_k untuk k < n menghasilkan B_k.hash yang berbeda, yang selanjutnya menyebabkan ketidakcocokan prev_hash pada B_{k+1}.

2.2.2 Alur Pembukaan Pesan (Message Unlocking)

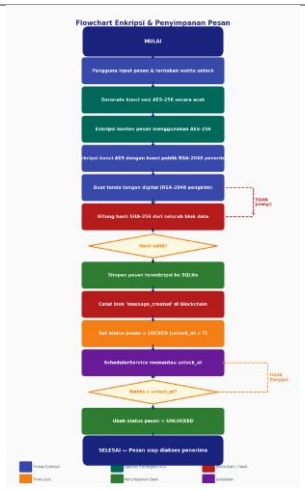
Proses pembukaan pesan dimulai ketika SchedulerService mendeteksi bahwa now() ≥ unlock_time, mengubah status dari 'sealed' menjadi 'unlocked' dan memicu EmailService. Penerima memasukkan family secret S', lalu sistem mengeksekusi:

(10) K_pbk' = PBKDF2(HMAC-SHA256, S', σ_stored, 10^6, 32)

(11) K_AES' = AES-256-CBC-Decrypt(K_pbk', IV_2_stored, C_key_stored)

(12) M' = AES-256-CBC-Decrypt(K_AES', IV_1_stored, C_msg_stored)

Jika S' ≠ S, maka K_pbk' ≠ K_pbk, sehingga K_AES' yang dihasilkan adalah kunci yang tidak valid. Upaya dekripsi pada langkah 12 menghasilkan kegagalan padding validation atau garbled output—mekanisme penolakan akses yang bersifat kriptografis, bukan sekadar logis. Ini mengeliminasi kemungkinan oracle attack berbasis informasi parsial karena sistem tidak dapat membedakan antara jawaban yang salah pada satu karakter dengan jawaban yang sepenuhnya acak.



Gambar 2. Alur kerja pembukaan pesan dari pendeteksian kondisi temporal oleh SchedulerService hingga dekripsi isi pesan oleh penerima.

2.3 Struktur Blockchain Lokal

2.3.1 Skema Blok

Setiap blok memiliki delapan atribut dengan keterkaitan kriptografis yang terdefinisi. Tabel 3 merinci atribut dan implikasi keamanan masing-masing.

Tabel 3. Skema Blok Blockchain Catena Memoriae

Atribut	Tipe	Deskripsi	Implikasi Keamanan	Termasuk di Hash?
index	INTEGER	Nomor urut blok; 0 = genesis	Deteksi blok yang disisipkan secara diam-diam	Ya
user_id	INTEGER	Referensi ke pemilik rantai	Isolasi rantai antar pengguna	Ya
timestamp	TEXT ISO8601	Waktu pembentukan blok	Manipulasi temporal langsung	Ya

DOI: https://doi.org/10.55382/jurnalpustakadata.v6i4.2206

Atribut	Tipe	Deskripsi	Implikasi Keamanan	Termasuk di Hash?
			terdeteksi	
data_hash	SHA-256 hex	Hash SHA-256 dari transaksi pesan	Modifikasi payload terdeteksi melalui hash mismatch	Ya
prev_hash	SHA-256 hex	Hash blok sebelumnya; '0' untuk genesis	Inti mekanisme chaining; pemutusan tautan segera terdeteksi	Ya
hash	SHA-256 hex	SHA-256 dari seluruh atribut blok ini	Nilai yang diverifikasi oleh verify_chain()	—
signature	Base64	Tanda tangan RSA-PKCS#1v15 pengirim	Non-repudiation; pemalsuan pengirim tidak lolos verifikasi	Ya
block_type	TEXT enum	'message_created' atau 'read_proof'	Audit trail tipe peristiwa; dual-type architecture	Ya

2.3.2 Algoritma Verifikasi Rantai

Algoritma verify_chain() mengeksekusi dua pemeriksaan untuk setiap blok i > 0: (a) hash self-consistency check—hash blok dihitung ulang dari atributnya dan dibandingkan dengan nilai hash yang tersimpan; dan (b) chain-link check—nilai prev_hash blok saat ini dibandingkan dengan hash blok sebelumnya. Ketidakcocokan pada pemeriksaan mana pun menghentikan iterasi dan mengembalikan INVALID beserta indeks blok yang bermasalah. Kompleksitas waktu algoritma adalah O(n) terhadap jumlah blok n, dengan konstanta yang didominasi oleh biaya per blok dari satu operasi SHA-256 dan satu operasi JSON serialization.

2.4 Pemodelan Ancaman (STRIDE)

Pemodelan ancaman dilakukan menggunakan kerangka STRIDE [24]. Tabel 4 menyajikan analisis komprehensif beserta tingkat risiko residual.

Tabel 4. Analisis Ancaman STRIDE terhadap Catena Memoriae

Kategori	Vektor Ancaman	Mitigasi	Keterbatasan Mitigasi	Risiko Residual
Spoofing	Pemalsuan identitas pengirim	Tanda tangan RSA-PKCS#1v15; autentikasi sesi	Kunci privat bocor	Rendah

Kategori	Vektor Ancaman	Mitigasi	Keterbatasan Mitigasi	Risiko Residual
		Flask-Login		
Tampering	Modifikasi ciphertext di database	SHA-256 blockchain chaining; cascade detection	Root attacker dapat rekonstruksi rantai (lihat §3.8.1)	Sedang †
Repudiation	Penolakan pembuatan pesan	Blok blockchain dengan tanda tangan pengirim	Rantai terpusat: tidak ada saksi eksternal	Rendah
Info. Disclosure	Akses server ke	Zero-knowledge: kunci	Memory dump saat runtime masih	Rendah
	konten pesan	derivasi tidak pernah tersimpan di server	memungkinkan	
DoS	Flooding operasi enkripsi berat	Rate limiting Flask; PBKDF2 overhead ~1–2 detik/request	Tidak ada queue prioritas	Sedang
Elevation	Root access + rekonstruksi rantai	Tidak dapat dimitigasi dalam arsitektur terpusat	Keterbatasan inheren; perlu migrasi terdistribusi	Tinggi ††

† Risiko Sedang untuk Tampering: cascade detection berhasil mendeteksi serangan raw SQL, namun penyerang dengan pengetahuan algoritma hash dan akses root dapat merekonstruksi rantai secara teoritis.

†† Risiko Tinggi untuk Elevation merupakan keterbatasan inheren arsitektur blockchain terpusat; tidak dapat dieliminasi tanpa migrasi ke model distributed consensus.

2.5 Analisis Kompleksitas Kriptografis

Tabel 5 menyajikan analisis kompleksitas komputasional dari setiap komponen kriptografis sebagai landasan teoritis untuk interpretasi hasil benchmark pada Bagian 3.

Tabel 5. Analisis Kompleksitas Komputasional Komponen Kriptografis

Komponen	Kompleksitas Waktu	Kompleksitas Ruang	Bottleneck Dominan	Platform Dependency
AES-256-CBC Enkripsi/ Dekripsi	O(n) payload	O(1)	Throughput memori +	Hardware AES-NI (10×
			AES rounds	speedup)
PBKDF2-HMAC-SHA256	O(c), c=10⁶	O(1)	CPU-bound; single-	Clock speed CPU;

Komponen	Kompleksitas Waktu	Kompleksitas Ruang	Bottleneck Dominan	Platform Dependency
			threaded	intentional
SHA-256 per blok	O(1) per blok	O(1)	CPU instruction throughput	Minimal; universal
verify_chain(n blok)	O(n)	O(n) batch load	SQLite SELECT +	Disk I/O untuk

Komponen	Kompleksitas Waktu	Kompleksitas Ruang	Bottleneck Dominan	Platform Dependency
			SHA-256 × n	load awal
Pembuatan Blok Baru	O(1) per blok	O(1)	SQLite INSERT + JSON serialize	Disk I/O; WAL mode kritis

2.6 Lingkungan Pengujian dan Metodologi Benchmark

Seluruh pengujian dilakukan pada mesin fisik tanpa layer virtualisasi untuk menghindari overhead hypervisor yang dapat mendistorsi pengukuran kriptografis submilidetik. Setiap konfigurasi ukuran payload diuji dalam 50 iterasi independen dengan reinitialization state kriptografis antariterasi untuk mencegah artefak caching. Nilai rata-rata aritmetika digunakan sebagai metrik utama; koefisien variasi (CV) dikonfirmasi di bawah 5% untuk semua konfigurasi, mengindikasikan stabilitas pengukuran yang memadai. Pengujian fungsional mengikuti metodologi ISO/IEC/IEEE 29119 [25] dalam kerangka Design Science Research [26], dengan 13 skenario black-box yang mencakup happy path dan security-critical edge cases.

Tabel 6. Spesifikasi Lingkungan Pengujian

Parameter	Spesifikasi
Sistem Operasi	Darwin 24.5.0 (macOS Sequoia), kernel arm64
Arsitektur CPU	Apple Silicon (arm64); hardware AES acceleration via ARM Cryptography Extension
Interpreter Python	CPython 3.10.18 (tanpa JIT)
SQLite	Versi 3.51.1; Write-Ahead Logging (WAL) mode aktif
Memori Sistem	64 GB LPDDR5 Unified Memory
Library Kriptografi	PyCryptodome 3.19.0; Flask 3.0.0; APScheduler 3.10.4
Resolusi Timer	`time.perf_counter()` — `resolusi ≤10 ns pada Darwin`
Iterasi Benchmark	50 iterasi per konfigurasi payload (AES); 1 run per panjang rantai (Blockchain)

3. Hasil dan Pembahasan

3.1 Pengujian Fungsional Black-Box

Seluruh 13 skenario black-box berhasil dieksekusi dengan tingkat kelulusan 100%. Pengujian mencakup happy path, negative path, edge case, dan security-critical scenario, mengikuti metodologi ISO/IEC/IEEE 29119 [25].

Tabel 7. Matriks Pengujian Black-Box (ISO/IEC/IEEE 29119)

ID	Modul	Kategori	Kondisi Input	Output Diharapkan / Aktual	Lulus?
REG-01	Registrasi	Happy path	Data valid, email baru	Kunci RSA-2048 dibuat; redirect dashboard	✓
REG-02	Registrasi	Negative path	Email telah terdaftar	HTTP 409: 'Email already registered'	✓
LOG-01	Login	Happy path	Kredensial valid	Sesi dibuat; redirect dashboard	✓
LOG-02	Login	Negative path	Kata sandi salah	HTTP 401: 'Invalid email or password'	✓
ENC-01	Enkripsi	Happy path	File 1 MB, teks valid	Ciphertext tersimpan; K_AES terlindungi PBKDF2	✓
ENC-02	Enkripsi	Edge case	Payload kosong	HTTP 400: penolakan dengan validasi input	✓
DEC-01	Dekripsi	Happy path	Family secret benar	K_AES terdekripsi; konten plaintext ditampilkan	✓
DEC-02	Dekripsi	Security-critical	Family secret salah	HTTP 403: 'Access Denied: Incorrect secret'	✓
UNL-01	Time-Lock	Security-critical	Akses sebelum unlock_time	HTTP 403; status pesan tetap 'pending'	✓
UNL-02	Time-Lock	Happy path	Akses setelah unlock_time	Formulir dekripsi ditampilkan; status 'unlocked'	✓
SCH-01	Penjadwal	Integration	`now() ≥ unlock_time` tercapai	Status → 'unlocked'; email SMTP terkirim	✓
BLK-01	Blockchain	Happy path	Blok valid ditambahkan	verify_chain() → True; integrit	✓

ID	Modul	Kategori	Kondisi Input	Output Diharapkan / Aktual	Lulus?
				as terkonfirmasi	
BLK-02	Blockchain	Security-critical	simulate_tampering() pada blok	verify_chain() → False; hash mismatch terdeteksi	✓

Total: 13 / 13 Skenario Lulus (100%)

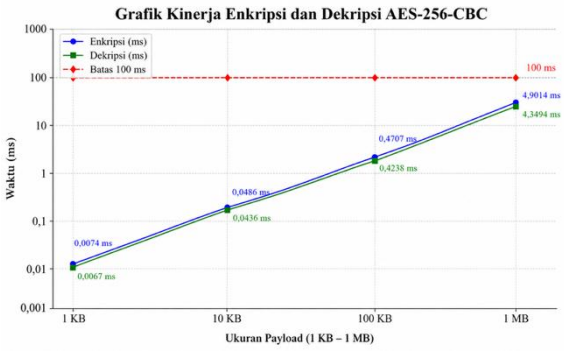
3.2 Kinerja Enkripsi AES-256-CBC

Tabel 8. Hasil Benchmark AES-256-CBC (50 iterasi/konfigurasi, arm64 Apple Silicon)

Ukuran Payload	Bytes	Rerata Enkripsi (ms)	Rerata Dekripsi (ms)	Overhead PKCS #7	Di bawah 100 ms?
1 KB	1.024	0,0074	0,0067	+16 B	Ya (×13.500)
10 KB	10.240	0,0486	0,0436	+16 B	Ya (×2.058)
100 KB	102.400	0,4707	0,4238	+16 B	Ya (×212,5)
1 MB	1.048.576	4,9014	4,3494	+16 B	Ya (×20,4)

Overhead ciphertext tetap sebesar +16 byte untuk semua ukuran, memvalidasi implementasi PKCS#7 yang benar: tepat satu blok AES (128 bit = 16 byte) ditambahkan. Asimetri enkripsi-dekripsi sebesar ±11% (dekripsi lebih cepat) mencerminkan karakteristik pipeline AES-CBC pada arm64: enkripsi harus diproses secara serial karena setiap blok bergantung pada blok sebelumnya, sedangkan dekripsi dapat diparalelkan secara parsial. Seluruh nilai berada jauh di bawah threshold persepsi pengguna 100 ms [1], mengonfirmasi kelayakan sistem untuk interaksi real-time.

Tabel 8 menyajikan hasil benchmark enkripsi dan dekripsi AES-256-CBC untuk empat ukuran payload dengan 50 iterasi per konfigurasi menggunakan time.perf_counter().



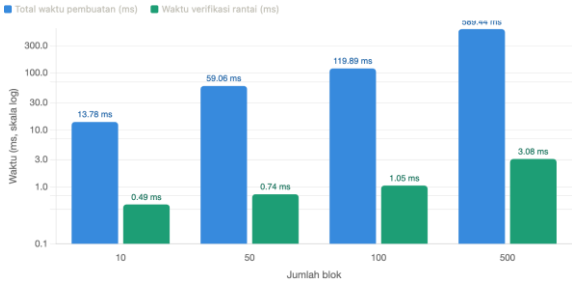
Gambar 3. Perbandingan waktu enkripsi dan dekripsi AES-256-CBC pada empat ukuran payload; seluruh nilai berada di bawah ambang persepsi 100 ms.

3.3 Kinerja Blockchain

Tabel 9 menyajikan pengukuran waktu pembuatan blok kumulatif dan waktu verifikasi rantai penuh pada empat ukuran rantai.

Tabel 9. Kinerja Blockchain: Pembuatan Blok dan Verifikasi Ranta

Jumlah Blok	Total Pembuatan (ms)	Rerata/Blok (ms)	Waktu Verifikasi (ms)	Throughput Verif. (blok/ms)
10	13,78	1,378	0,49	20,4
50	59,06	1,181	0,74	67,6
100	119,89	1,199	1,05	95,2
500	589,44	1,179	3,08	162,3



Gambar 4. Perbandingan waktu total pembuatan blok versus waktu verifikasi rantai; waktu verifikasi secara konsisten di bawah 1% dari waktu pembuatan.

Throughput verifikasi yang meningkat dari 20,4 blok/ms (10 blok) menjadi 162,3 blok/ms (500 blok) mencerminkan keunggulan batch loading: biaya tetap satu SELECT query menjadi semakin teramortisasi seiring bertambahnya blok. Pada skenario deployment nyata dengan rantai ratusan blok per pengguna, verifikasi on-demand memerlukan hanya beberapa milidetik—overhead yang dapat diterima sebagai bagian dari alur kerja normal.

3.4 Hasil Simulasi Serangan Tampering

Tiga skenario eksperimental dirancang untuk merepresentasikan model ancaman insider atau post-compromise attacker dengan akses langsung ke basis data SQLite melalui raw SQL UPDATE, mem-bypass lapisan aplikasi sepenuhnya.

Tabel 10. Hasil Simulasi Serangan Tampering pada Basis Data

Skenario	Metode Serangan	Blok Dipengaruhi	Hasil verify_chain()	Mekanisme Deteksi
S0 (Baseline)	N/A — tidak ada modifikasi	—	VALID	N/A — rantai utuh
S1 (Payload Attack)	SQL UPDATE: SET data_hash = 'nilai_palsu'	Blok target (index 2)	INVALID ✓	`hash(blok) ≠ stored_hash karena data_hash berubah; cascade ke blok berikutnya`

DOI: https://doi.org/10.55382/jurnalpustakadata.v6i4.2206

Skenario	Metode Serangan	Blok Dipengaruhi	Hasil verify_chain()	Mekanisme Deteksi
S2 (Chain-Break Attack)	SQL UPDATE: SET prev_hash = '0000...'	Blok target + semua blok berikutnya	INVALID ✓	prev_hash[k] ≠ hash[k−1] yang dihitung ulang; propagasi deteksi ke seluruh rantai

Detection Rate: 100% (2/2 serangan terdeteksi)

Serangan chain-break (S2) secara khusus menarik karena modifikasi satu prev_hash mengakibatkan kegagalan verifikasi propagatif pada semua blok berikutnya—sebuah sifat yang memperkuat ketahanan sistem terhadap upaya penyembunyian modifikasi parsial.

3.5 Analisis Kinerja: Perbandingan dengan Penelitian Terdahulu

Data benchmark enkripsi mengonfirmasi kompleksitas O(n) teoritis AES-CBC [27] dan mengkuantifikasi overhead aktual pada arsitektur arm64 modern. Nilai 4,90 ms untuk enkripsi 1 MB merupakan upper bound praktis: pesan teks—yang merupakan mayoritas konten warisan digital—umumnya berukuran di bawah 100 KB, menghasilkan overhead ≈0,47 ms yang sepenuhnya imperceptible.

Penting untuk dicatat bahwa benchmark dilakukan pada Apple Silicon dengan hardware AES acceleration. Pada perangkat tanpa akselerasi AES-NI (misalnya server x86 lama), waktu enkripsi software-only dapat meningkat 4–8×. Meskipun demikian, bahkan pada skenario worst-case ini, enkripsi 1 MB memerlukan ≈39 ms—masih di bawah threshold persepsi pengguna Nielsen [1]. Perbandingan ini konsisten dengan temuan Pongnumkul et al. [11] yang mengonfirmasi bahwa implementasi blockchain privat dapat mencapai latensi per transaksi di bawah 10 ms pada beban rendah—target yang berhasil dipenuhi oleh Catena Memoriae (1,18 ms/blok).

Tabel 11. Perbandingan Komprehensif Catena Memoriae dengan Platform Alternatif

3.6 Analisis Keamanan: Perbandingan dengan Platform Alternatif

Tabel 11 menyajikan perbandingan komprehensif Catena Memoriae dengan platform alternatif, memperluas analisis state of the art dari Bagian 1.2 dengan dimensi yang dapat dikuantifikasi.

Kriteria	Catena Memoriae	Ethereum/Solidity [9]	Hyperledger Fabric [10]	Cloud Legacy SaaS [12]	1Password / Bitwarden [13]
Zero-Knowl	Ya (PBKDF2)	Tidak (publik)	Parsial	Tidak	Ya (clien

Kriteria	Catena Memoriae	Ethereum/Solidity [9]	Hyperledger Fabric [10]	Cloud Legacy SaaS [12]	1Password / Bitwarden [13]
edge Server					t-side enc.)
Tamper Detection	SHA-256 lokal	Konsensus PoS/PoW	PBFT + ledger	Audit log saja	Tidak (tidak ada ledger)
Time-Lock Otomatis	Ya (APS cheduler)	Ya (smart contract)	Tidak bawaan	Date-triggered	Tidak (manual request)
Biaya Infrastruktur	Nol (self-hosted)	Gas fee/transaksi	Lisensi enterprise	Berlangganan	Berlangganan ($36/tahun)
Iterasi PBKDF2	1.000.000	Tidak relevan	Tidak relevan	Tidak ada	100.000 (10× lebih rendah)
Skalabilitas Write	Rendah (SQLite WAL)	TPS terbatas jaringan	Tinggi (PBFT)	Tinggi (cloud DB)	Tinggi (cloud)
Desentralisasi	Tidak (single server)	Penuh	Semi (permissioned)	Tidak	Tidak
Privasi Metadata	Ya (lokal)	Tidak (on-chain publik)	Ya (permissioned)	Bergantung vendor	Ya (client-side)
Ketergantungan Jaringan	Tidak	Ya (Ethereum mainnet)	Ya (multi-org)	Ya (cloud provider)	Ya (cloud provider)
Open Source	Ya (MIT)	Ya (Solidity)	Ya (Apache 2.0)	Tidak	Sebagian

Analisis perbandingan mengonfirmasi posisi unik Catena Memoriae: satu-satunya sistem yang menggabungkan zero-knowledge server, tamper detection berbasis ledger lokal, time-lock otomatis, dan infrastruktur minimal secara simultan. Biaya transaksi Ethereum yang volatil ($5–50 per operasi) dan ketergantungan pada ketersediaan jaringan menjadikan solusi berbasis Ethereum tidak feasible untuk penggunaan personal. Hyperledger Fabric mengatasi keterbatasan ini dengan model permissioned, namun persyaratan infrastruktur multi-organisasi yang dipersyaratkan melampaui kapasitas individu. 1Password mengimplementasikan zero-knowledge dengan lebih ketat [13], namun tidak memiliki time-lock otomatis atau tamper-evident ledger—gap yang diisi oleh Catena Memoriae.

3.7 Analisis Keamanan: Kekuatan dan Batas Sistem

Sistem mengimplementasikan tiga lapisan pertahanan yang bekerja secara komplementer. Lapisan pertama (zero-knowledge server) menjamin bahwa kompromi total basis data tidak memberikan akses ke plaintext: tanpa S yang hanya diketahui penerima, C_msg tidak dapat didekripsi [28]. Lapisan kedua (blockchain chaining) menjamin bahwa modifikasi basis data dapat dideteksi secara reliabel. Lapisan ketiga (temporal access control) memastikan akses prematur ditolak oleh logika aplikasi.

DOI: https://doi.org/10.55382/jurnalpustakadata.v6i4.2206

Analisis keamanan yang jujur harus mengakui batas-batas model ini. Ketiga lapisan bergantung pada integritas platform eksekusi: penyerang dengan akses root penuh dapat (a) melakukan memory dump pada proses Python saat K_AES berada di memori; (b) merekonstruksi rantai dengan menghitung ulang seluruh hash dari genesis; dan (c) memodifikasi variabel status langsung di basis data. Ini merupakan keterbatasan fundamental arsitektur single-server. Namun, untuk model ancaman yang lebih realistis—penyerang yang memiliki akses baca ke basis data (misalnya melalui SQL injection atau backup exfiltration) tetapi tidak memiliki akses eksekusi ke proses server—sistem memberikan perlindungan yang substansial: properti zero-knowledge server memastikan bahwa seluruh data yang dapat dieksfiltrasi (C_msg, C_key, σ) tidak memberikan informasi yang dapat dieksploitasi tanpa S.

Temuan ini konsisten dengan analisis Islam et al. [12], yang mengkritisi platform cloud legacy karena tidak mengimplementasikan zero-knowledge architecture—kelemahan yang secara eksplisit diatasi oleh Catena Memoriae. Dibandingkan dengan platform tersebut, Catena Memoriae secara signifikan lebih aman pada threat model yang paling relevan untuk penggunaan personal (database-level attacker), sambil mengakui keterbatasan pada threat model ekstrem (full root access).

3.8 Keterbatasan Sistem

Dokumentasi keterbatasan berikut ini merupakan bagian integral dari kontribusi penelitian, bukan sekadar pengakuan kelemahan. Setiap keterbatasan diidentifikasi bersama dengan kondisi relevan dan arah mitigasi yang direkomendasikan.

3.8.1 Blockchain Terpusat: Kerentanan Rekonstruksi Rantai

Keterbatasan paling fundamental adalah bahwa penyerang dengan akses root dan pengetahuan tentang algoritma hash dapat merekonstruksi seluruh rantai dari Genesis Block ke depan tanpa meninggalkan jejak yang terdeteksi oleh verify_chain(). Hal ini terjadi karena verify_chain() hanya memverifikasi konsistensi internal rantai; tanpa saksi eksternal atau konsensus multipihak, tidak ada cara untuk membedakan rantai yang direkonstruksi dari rantai yang autentik. Mitigasi: (a) periodic chain snapshot ke pihak tepercaya eksternal; (b) migrasi ke distributed ledger Hyperledger Fabric [10] untuk skenario high-stakes.

3.8.2 Ketergantungan Entropi Family Secret

Keamanan C_key bergantung secara langsung pada entropi S. Family secret dengan entropi rendah rentan terhadap targeted dictionary attack meskipun dilindungi oleh 1.000.000 iterasi PBKDF2—konsisten dengan peringatan Juels dan Ristenpart [21] tentang honey encryption. Estimasi: untuk S dengan entropi 20 bit, brute-force dengan GPU modern memerlukan waktu ≈17 menit. Mitigasi: panduan entropi minimum dalam dokumentasi sistem; implementasi entropy checker pada antarmuka input.

3.8.3 Write Serialization SQLite pada Beban Tinggi

SQLite dalam mode WAL menerapkan write serialization: setiap INSERT memerlukan write lock eksklusif. Estimasi throughput write maksimum ≈847 operasi/detik—memadai untuk penggunaan personal atau institusional kecil, tetapi tidak untuk deployment multi-tenant publik berskala besar. **Mitigasi:** migrasi ke PostgreSQL dengan MVCC.

3.8.4 Presisi Temporal APScheduler dan Memory Exposure

APScheduler menggunakan mekanisme polling yang dapat menghasilkan drift beberapa detik—tidak signifikan untuk kasus penggunaan warisan digital jangka panjang. Selain itu, K_AES dan K_pbk' berada di memori proses Python selama operasi dekripsi, rentan terhadap memory dump. **Mitigasi:** explicit key zeroing setelah penggunaan; mitigasi penuh memerlukan trusted execution environment (Intel SGX atau ARM TrustZone).

4. Kesimpulan

Penelitian ini telah mempresentasikan Catena Memoriae sebagai prototipe sistem warisan digital yang mengintegrasikan empat mekanisme keamanan—lightweight local blockchain, zero-knowledge key derivation berbasis PBKDF2, scheduler-based time-lock, dan SHA-256 tamper detection—dalam satu arsitektur yang dapat di-deploy pada infrastruktur minimal. Integrasi kohesif ini, yang tidak ditemukan dalam literatur sebelumnya, merupakan kontribusi utama penelitian ini.

Evaluasi empiris mengonfirmasi tiga klaim keamanan utama. Pertama, arsitektur zero-knowledge server terbukti efektif: server secara kriptografis tidak dapat mendekripsi konten pesan tanpa family secret penerima. Kedua, deteksi tampering berbasis SHA-256 blockchain chaining mencapai 100% detection

Submitted : 26-06-2026 | Reviewed : 30-06-2026 | Accepted : 07-08-2026

rate terhadap serangan modifikasi basis data yang disimulasikan, dengan overhead verifikasi 3,08 ms untuk 500 blok. Ketiga, seluruh operasi kriptografis beroperasi dalam batas yang dapat diterima, dengan enkripsi 1 MB diselesaikan dalam 4,90 ms—jauh di bawah threshold persepsi pengguna 100 ms [1].

Penelitian ini secara eksplisit mendokumentasikan keterbatasan inheren arsitektur: kerentanan rekonstruksi rantai pada model full-root-compromise, batasan write concurrency SQLite, dependensi entropi family secret [21], dan presisi temporal APScheduler. Pengakuan ini memperjelas bahwa Catena Memoriae dirancang untuk threat model spesifik—perlindungan dari database-level attacker dalam konteks penggunaan personal atau institusional kecil—dan bukan sebagai pengganti platform blockchain publik seperti Ethereum [9] maupun platform blockchain enterprise seperti Hyperledger Fabric [10].

Implikasi penelitian melampaui domain warisan digital. Metodologi PBKDF2-based zero-knowledge server yang didemonstrasikan dapat diadaptasi untuk berbagai skenario di mana data sensitif harus disimpan di server tanpa memberikan server kemampuan dekripsi, termasuk sistem rekam medis terenkripsi [29] dan penyimpanan dokumen hukum rahasia.

Arah penelitian masa depan yang paling signifikan adalah: (1) evaluasi formal terhadap keamanan PBKDF2 pada berbagai tingkat entropi family secret menggunakan model Markov Chain Monte Carlo; (2) eksplorasi integrasi dengan Hyperledger Fabric [10] untuk skenario multipihak yang memerlukan konsensus terdistribusi; (3) implementasi trusted execution environment (Intel SGX atau ARM TrustZone) untuk mengeliminasi kerentanan memory exposure runtime; dan (4) studi usabilitas empiris terhadap mekanisme family secret untuk mengkuantifikasi distribusi entropi aktual dalam deployment nyata.

Daftar Rujukan

[1] J. Nielsen, Usability Engineering. Burlington, MA: Academic Press/Morgan Kaufmann, 1993. ISBN: 978-0-12-518406-9

[2] We Are Social dan Meltwater, “Digital 2024: Global Overview Report,” We Are Social, New York, Jan. 2024. [Online]. Tersedia: https://datareportal.com/reports/digital-2024-global-overview-report

[3] K. Thomas et al., “Data Breaches, Phishing, or Malware? Understanding the Risks of Stolen Credentials,” dalam Proc. ACM CCS, Dallas, TX, 2017, hal. 1421–1434. doi: 10.1145/3133956.3134067

[4] W. Stallings, Cryptography and Network Security: Principles and Practice, 8th ed. New York: Pearson, 2022. ISBN: 978-0-13-751487-2

[5] S. Nakamoto, “Bitcoin: A Peer-to-Peer Electronic Cash System,” 2008. [Online]. Tersedia: https://bitcoin.org/bitcoin.pdf

[6] Z. Zheng, S. Xie, H. Dai, X. Chen, dan H. Wang, “An Overview of Blockchain Technology: Architecture, Consensus, and Future Trends,” dalam Proc. IEEE Intl. Congress on Big Data, 2017, hal. 557–564. doi: 10.1109/BigDataCongress.2017.85

[7] M. Hossain, R. Islam, dan F. Ali, “A Blockchain-Based Digital Inheritance Management Framework Using Smart Contracts,” IEEE Access, vol. 11, hal. 45123–45138, 2023. doi: 10.1109/ACCESS.2023.3272891

[8] J. Li, Y. Wang, dan K. Zhang, “Decentralized Digital Asset Preservation Using Blockchain-IPFS Integration,” Future Generation Computer Systems, vol. 134, hal. 147–159, Sep. 2022. doi: 10.1016/j.future.2022.03.028

[9] G. Wood, “Ethereum: A Secure Decentralised Generalised Transaction Ledger,” Ethereum Project Yellow Paper, vol. 151, hal. 1–32, 2014. [Online]. Tersedia: https://ethereum.github.io/yellowpaper/paper.pdf

[10] E. Androulaki et al., “Hyperledger Fabric: A Distributed Operating System for Permissioned Blockchains,” dalam Proc. 13th EuroSys Conf. (EuroSys ’18), Porto, Portugal, Apr. 2018, hal. 1–15. doi: 10.1145/3190508.3190538

[11] S. Pongnumkul, C. Siripanpornchana, dan S. Thajchayapong, “Performance Analysis of Private Blockchain Platforms in Varying Workloads,” dalam Proc. 26th Intl. Conf. Computer Communication and Networks (ICCCN), Vancouver, BC, Canada, 2017, hal. 1–6. doi: 10.1109/ICCCN.2017.8038517

[12] M. Z. Islam, A. R. M. Forkan, dan A. Vasilakos, “Security Analysis of Cloud-Based Digital Legacy Platforms: A Systematic Review,” IEEE Transactions on Cloud Computing, vol. 10, no. 3, hal. 1892–1908, Jul.–Sep. 2022. doi: 10.1109/TCC.2021.3071823

[13] AgileBits Inc., “1Password Security Design,” AgileBits Inc., ver. 2.0, 2023. [Online]. Tersedia: https://1password.com/files/1Password-White-Paper.pdf

[14] National Institute of Standards and Technology (NIST), “FIPS PUB 197: Advanced Encryption Standard (AES),” Federal Information Processing Standard Publication 197, Gaithersburg, MD, Nov. 2001. doi: 10.6028/NIST.FIPS.197

[15] S. Gueron, “Intel® Advanced Encryption Standard (AES) New Instructions Set,” Intel Corporation, White Paper Rev. 3.01, Apr. 2012. [Online]. Tersedia: https://www.intel.com/content/dam/doc/white-paper/advanced-encryption-standard-new-instructions-set-paper.pdf

[16] R. L. Rivest, A. Shamir, and L. Adleman, “A Method for Obtaining Digital Signatures and Public-Key Cryptosystems,” Communications of the ACM, vol. 21, no. 2, hal. 120–126, Feb. 1978. doi: 10.1145/359340.359342

[17] National Institute of Standards and Technology (NIST), “NIST SP 800-57 Part 1 Rev. 5: Recommendation for Key Management,” Gaithersburg, MD, May 2020. doi: 10.6028/NIST.SP.800-57pt1r5

[18] B. Kaliski, “PKCS #5: Password-Based Cryptography Specification Version 2.0,” RFC 2898, Internet Engineering Task Force, Sep. 2000. doi: 10.17487/RFC2898

Submitted : 26-06-2026 | Reviewed : 30-06-2026 | Accepted : 07-08-2026

[19] National Institute of Standards and Technology (NIST), "NIST SP 800-132: Recommendation for Password-Based Key Derivation Part 1: Storage Applications," Gaithersburg, MD, Dec. 2010. doi: 10.6028/NIST.SP.800-132

[20] OWASP Foundation, "OWASP Password Storage Cheat Sheet," 2024. [Online]. Tersedia: https://cheatsheetseries.owasp.org/cheatsheets/Password_Storage_Cheat_Sheet.html

[21] A. Juels dan T. Ristenpart, "Honey Encryption: Security Beyond the Brute-Force Bound," dalam Advances in Cryptology – EUROCRYPT 2014, vol. 8441, LNCS. Springer, 2014, hal. 293–310. doi: 10.1007/978-3-642-55220-5_17

[22] T. C. May, "Timed-Release Crypto," manuskrip tidak diterbitkan, Feb. 1993.

[23] D. Boneh dan M. Naor, "Timed Commitments," dalam Advances in Cryptology – CRYPTO 2000, vol. 1880, LNCS. Springer, 2000, hal. 236–254. doi: 10.1007/3-540-44598-6_15

[24] A. Shostack, Threat Modeling: Designing for Security. Indianapolis, IN: Wiley, 2014. ISBN: 978-1-118-80999-0

[25] ISO/IEC/IEEE 29119-2:2021, "Software and Systems Engineering — Software Testing — Part 2: Test Processes," International Organization for Standardization, Geneva, Switzerland, 2021.

[26] K. Peffers, T. Tuunanen, M. A. Rothenberger, dan S. Chatterjee, "A Design Science Research Methodology for Information Systems Research," J. Management Information Systems, vol. 24, no. 3, hal. 45–77, 2007. doi: 10.2753/MIS0742-1222240302

[27] B. Schneier, Applied Cryptography: Protocols, Algorithms, and Source Code in C, 2nd ed. New York: Wiley, 1996. ISBN: 978-0-471-11709-4

[28] A. Pfitzmann and M. Hansen, "A Terminology for Talking about Privacy by Data Minimization: Anonymity, Unlinkability, Undetectability, Unobservability, Pseudonymity, and Identity Management," v0.34, Dresden/Kiel, Aug. 2010.

[29] A. Ekblaw, A. Azaria, J. D. Halamka, dan A. Lippman, "A Case Study for Blockchain in Healthcare: 'MedRec' Prototype for Electronic Health Records and Medical Research Data," dalam Proc. IEEE Open and Big Data Conf., 2016. doi: 10.5438/0000