10.48047/jocaaa.2024.33.02.43

Erasure Aware and Revocation Hardened Decentralized Storage Architectures for Regulated Big Data Ecosystems

Madupathi Parimala*[1], Dr. Gaurav Tyagi[2]

1.Research Scholar, Dept of Computer Science & Engineering, Chaudhary Charan Singh University Campus,Meerut, UP.
Email : pari.parillu@gmail.com

2. Associate Professor, Dept of Computer Science & Engineering, Chaudhary Charan Singh University Campus, Meerut,
UP. 250005.

ABSTRACT

The acceleration of data-intensive applications across regulated domains has exposed fundamental inadequacies in prevailing decentralized storage paradigms. While blockchain and content-addressed storage systems offer resilience, traceability, and distribution at scale, they remain structurally misaligned with dynamic access revocation, legal erasure mandates, and attribute confidentiality. Existing approaches tend to resolve these constraints in isolation, resulting in fragmented architectures that collapse under real-world governance and compliance pressures.This paper introduces a compliance-aware decentralized data governance framework that rethinks how authorization, persistence, and auditability are jointly enforced in large-scale environments. The proposed system fuses fine-grained cryptographic delegation with revocation-resilient re-keying, while embedding attribute-hiding verification logic directly into the access workflow. Instead of exposing policy semantics or user credentials, authorization decisions are validated through succinct cryptographic attestations, ensuring that compliance can be proven without disclosure. To reconcile immutable storage with erasure obligations, the framework employs controlled cryptographic entropy evolution, rendering targeted data permanently inaccessible without disrupting global system integrity.A distributed policy ledger coordinates revocation states, compliance proofs, and accountability traces, enabling verifiable enforcement across untrusted participants. Unlike prior blockchain–storage integrations, the design avoids mass re-encryption, minimizes verification latency, and preserves semantic privacy even under adversarial observation. Experimental evaluation over heterogeneous datasets demonstrates substantial reductions in revocation overhead, bounded disclosure leakage, and practical feasibility under realistic workloads.By unifying revocation efficiency, erasure compliance, and privacy preservation within a single operational model, this work advances decentralized data management toward regulatory viability, positioning blockchain-enabled storage as a credible foundation for sensitive, large-scale information ecosystems.

10.48047/jocaaa.2024.33.02.43

Keywords: *Decentralized data governance, Blockchain storage systems, Privacy-preserving access control, Revocation-aware encryption, Regulatory compliance, Zero-knowledge authorization, Distributed auditability, Big data security*

1 INTRODUCTION

The contemporary digital landscape is defined by an unprecedented escalation in data generation, circulation, and long-term retention. Modern computational ecosystems spanning healthcare analytics, financial intelligence, smart infrastructure, and large-scale sensing environments continuously produce data streams that are not only massive in volume but also sensitive in nature. These datasets frequently encode personal identifiers, behavioral patterns, clinical histories, and operational metadata whose misuse or exposure may lead to severe ethical, legal, and economic consequences. Consequently, data management infrastructures are no longer evaluated solely on performance or scalability; instead, they are increasingly judged on their capacity to guarantee confidentiality, controlled accessibility, accountability, and compliance with evolving regulatory frameworks.Traditional centralized data storage architectures have historically dominated enterprise and institutional deployments due to their simplicity of management and mature tooling. However, such architectures are intrinsically vulnerable to single points of failure, insider threats, opaque governance models, and jurisdictional concentration of trust. High-profile data breaches and prolonged service outages have revealed the fragility of centralized control, particularly when large populations of users or interconnected services rely on a shared backend. Moreover, centralized platforms often impose implicit trust assumptions, requiring users to relinquish direct control over their data lifecycle, including access delegation, revocation, and deletion [1].

Decentralized storage paradigms have emerged as a response to these structural weaknesses. By distributing data across peer-to-peer networks and eliminating centralized custodians, decentralized systems aim to enhance fault tolerance, resilience, and user autonomy. Among these paradigms, content-addressed storage systems have gained prominence due to their ability to decouple data identity from physical location. Instead of relying on mutable pointers or server-based indexing, content-addressed models reference data through cryptographic hashes, ensuring integrity and enabling efficient replication across distributed nodes [2].In parallel, blockchain technology has introduced a tamper-resistant mechanism for maintaining shared state among mutually distrustful participants. Through consensus-driven append-only ledgers, blockchains enable transparent auditing, verifiable provenance, and decentralized coordination without requiring a trusted intermediary. When combined with decentralized storage, blockchains offer a promising foundation for building data ecosystems that are auditable, censorship-resistant, and resilient to unilateral manipulation [1–3].

10.48047/jocaaa.2024.33.02.43

Despite this promise, the integration of blockchain with decentralized storage systems has revealed a series of fundamental tensions that remain unresolved. One such tension arises from the immutability guarantees that make blockchain-based systems attractive in the first place. While immutability ensures that historical records cannot be altered retroactively, it also conflicts with legal mandates that require data modification or deletion under specific circumstances. Regulations such as the General Data Protection Regulation enforce the principle that individuals retain the right to have their personal data erased or rendered inaccessible, even after initial consent has been granted [4,5].This regulatory requirement introduces a conceptual contradiction: decentralized systems are designed to preserve data indefinitely, whereas compliance regimes demand controlled forgetfulness. Existing implementations often address this contradiction superficially, either by storing encrypted data indefinitely while deleting encryption keys or by relying on off-chain agreements and policy statements. Such approaches lack verifiable guarantees and fail to provide technical proof that erasure obligations have been effectively enforced [6].Another critical limitation lies in access control. Fine-grained authorization mechanisms are essential in multi-user data ecosystems, where different stakeholders require differentiated access privileges based on role, context, or credential attributes. Cryptographic access control schemes have been proposed to embed policy enforcement directly into encrypted data, thereby preventing unauthorized access even in untrusted environments. However, many such schemes suffer from poor scalability when user revocation is required. Revoking a single participant may necessitate re-encrypting large volumes of data or redistributing keys to all remaining users, resulting in prohibitive overheads in large-scale deployments [7].

Furthermore, conventional access control mechanisms often expose sensitive metadata during authorization. Policy descriptors, attribute labels, or role identifiers may be visible to verifiers or storage providers, enabling inference attacks and profiling even when data payloads remain encrypted. In regulated environments, such leakage can itself constitute a violation, as metadata may reveal sensitive associations or behavioral characteristics [8].These challenges underscore the need for a fundamentally different approach to decentralized data governance—one that treats revocation efficiency, erasure compliance, and attribute privacy not as auxiliary concerns but as first-class design objectives. Rather than layering isolated solutions atop existing infrastructures, a unified architectural perspective is required to ensure that these properties reinforce rather than undermine one another .Recent advances in cryptographic primitives provide new opportunities to address these challenges holistically. Delegation-based encryption techniques allow access rights to be transferred or withdrawn without direct involvement of the data owner. Re-encryption mechanisms enable controlled transformation of ciphertexts, permitting access updates without exposing plaintext or regenerating data artifacts. At the same time, zero-knowledge verification techniques enable one party to prove compliance with a policy or possession of a credential without revealing the underlying attributes themselves [9].When orchestrated correctly, these primitives allow decentralized systems to

10.48047/jocaaa.2024.33.02.43

enforce access policies dynamically while preserving semantic privacy. Authorization decisions can be validated cryptographically rather than through disclosure, and revocation can be enacted through localized transformations rather than global reconfiguration. Additionally, by incorporating time-dependent or state-evolving cryptographic material, it becomes possible to render previously stored data cryptographically irrecoverable, thereby aligning immutable storage with erasure mandates [10].

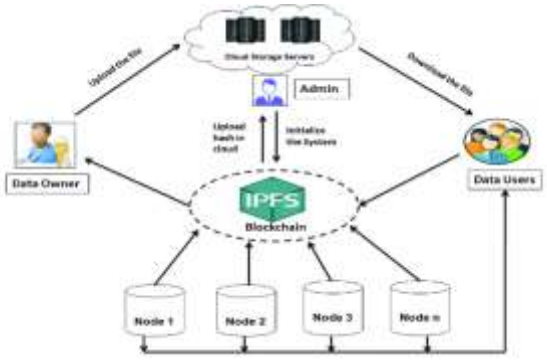

Figure 1. Conceptual Architecture of a Compliance Aware and Privacy Preserving Blockchain Enabled Decentralized Data Governance Framework

Figure1 illustrates the conceptual foundation of a decentralized data governance architecture that integrates distributed storage, cryptographic access enforcement, and policy coordination into a unified operational flow. At the outermost layer, data producers and consumers interact with the system through a client-side interface responsible for data preparation, encryption, and request generation. This design choice ensures that sensitive content is protected before leaving the data owner's control, thereby minimizing trust assumptions toward the underlying infrastructure [11].However, the practical realization of such an integrated framework presents non-trivial design challenges. Cryptographic operations must be efficient enough to support high-throughput data access. Verification latency must remain within acceptable bounds for real-world applications. Policy coordination mechanisms must remain decentralized while preventing inconsistent or conflicting authorization states. Moreover, the system must provide auditable evidence of compliance without introducing new privacy risks [12].

2 RELATED WORKS

Decentralized data management has attracted sustained research attention as data ecosystems expand beyond organizational boundaries and into federated, cross-domain environments. Early studies primarily focused on replacing centralized repositories with distributed ledgers to enhance transparency and tamper resistance. These efforts demonstrated that immutable ledgers can effectively record data provenance and access events, thereby improving auditability in collaborative settings [13]. However, such systems frequently relied on external storage backends and inherited their

10.48047/jocaaa.2024.33.02.43

security and governance limitations, resulting in partial decentralization rather than end-to-end trust minimization.Subsequent research explored the coupling of blockchain with distributed storage systems to overcome scalability constraints associated with on-chain data persistence. Content-addressed storage frameworks were introduced to offload bulk data while retaining cryptographic references on-chain, enabling integrity verification without excessive ledger growth [14]. Although this hybrid model improved performance and storage efficiency, it treated access control as an orthogonal concern, often delegating authorization decisions to application-layer logic or centralized key managers. This separation weakened the overall trust model and exposed sensitive policy metadata during access evaluation.To address fine-grained authorization requirements, attribute-oriented encryption schemes gained prominence. These approaches embed access policies directly into cryptographic material, ensuring that only eligible users can decrypt protected data. While effective in static environments, their limitations became apparent in dynamic settings where users frequently join, leave, or change roles. Revocation operations typically required widespread re-encryption or redistribution of keys, creating computational and communication overheads that scale poorly with data volume and user population.

Several studies attempted to mitigate revocation costs by introducing intermediary entities capable of transforming ciphertexts on demand. Delegation-based encryption and re-encryption techniques enabled limited forms of access update without exposing plaintext or reissuing data from the owner . Despite these advances, many implementations assumed trusted intermediaries or exposed policy structures during transformation, raising concerns regarding confidentiality and metadata leakage. Moreover, these solutions rarely considered regulatory erasure requirements, implicitly assuming perpetual data retention.Parallel research streams examined privacy-preserving verification mechanisms to reduce information leakage during authorization. Zero-knowledge proof systems were proposed to allow users to demonstrate eligibility without disclosing underlying attributes. While theoretically compelling, early implementations were computationally intensive and unsuitable for high-throughput data access. As a result, their adoption remained limited to niche applications such as anonymous payments or credential verification, rather than large-scale data governance.

More recent works have revisited zero-knowledge techniques in the context of access control, benefiting from advances in succinct proof systems and optimized verification algorithms. These efforts showed that attribute-hiding authorization is feasible under constrained performance budgets. However, they often focused on isolated authorization checks and did not integrate revocation handling or storage-layer implications, leaving the broader system architecture fragmented.Regulatory compliance, particularly the right to erasure, has been comparatively underexplored in decentralized storage research. Some studies proposed key deletion as a proxy for data erasure, arguing that encrypted data without keys is effectively inaccessible [15]. Critics have noted that such approaches

10.48047/jocaaa.2024.33.02.43

lack verifiable guarantees and may fail under key recovery or replication scenarios. Others suggested off-chain deletion agreements or legal governance layers, which shift enforcement responsibility away from technical mechanisms and undermine decentralization principles [16].Table 1 summarizes representative approaches across blockchain-based storage and access control systems, highlighting their treatment of revocation, privacy, and compliance dimensions.

Table 1.Comparison of Existing Decentralized Data Management Approaches

Approach Category	Storage Model	Access Control	Revocation Efficiency	Attribute Privacy	Erasure Compliance
Blockchain only systems	On-chain metadata	Application-level	Low	Low	Not addressed
Blockchain plus distributed storage	Off-chain content addressing	Centralized key control	Low	Medium	Not addressed
Attribute-based encryption systems	Distributed encrypted storage	Embedded policies	Low	Medium	Not addressed
Re-encryption assisted models	Distributed encrypted storage	Delegated transformation	Medium	Low	Limited
Privacy-preserving authorization models	Distributed storage	Proof-based validation	Medium	High	Not addressed

As evident from Table 1, prior solutions tend to optimize individual properties while neglecting others. Systems that emphasize storage scalability often compromise on access privacy, whereas those prioritizing cryptographic expressiveness struggle with revocation overhead and compliance feasibility.A smaller body of work has attempted to bridge these gaps by combining multiple cryptographic primitives. These hybrid designs suggest that delegation, re-encryption, and privacy-preserving verification can coexist within a unified framework [17]. Nevertheless, existing proposals often lack rigorous evaluation under realistic workloads or omit explicit mechanisms for enforcing erasure mandates. Additionally, policy coordination is frequently centralized or weakly synchronized,

10.48047/jocaaa.2024.33.02.43

3 PROPOSED METHODOLOGY

The proposed methodology is designed with the following objectives:

- To enforce fine-grained and revocable access control in decentralized storage without relying on trusted intermediaries.
- To achieve regulatory erasure compliance through cryptographic state evolution rather than physical data deletion.
- To preserve attribute and policy confidentiality during authorization using privacy-preserving verification.
- To ensure scalability by avoiding global re-encryption during revocation and access updates.
- To provide verifiable auditability of access, revocation, and erasure events in a decentralized setting.

Based on these objectives, the methodology adopts a modular architecture in which storage, policy governance, verification, and cryptographic lifecycle management are coordinated through verifiable cryptographic protocols.

3.1 Architectural Overview

Figure 2 provides a visual abstraction of the modular interactions described in this section.The system is composed of five tightly coupled but logically independent modules: the client-side control layer, the decentralized storage layer, the policy coordination ledger, the verification gateway, and the cryptographic lifecycle manager. This separation ensures scalability and fault isolation while preventing any single component from gaining excessive authority over data access decisions [18,19].At a high level, data owners encrypt content locally and distribute only ciphertext fragments into the decentralized storage network. Access policies and revocation states are not embedded in plaintext form but are represented as cryptographic commitments anchored to a distributed ledger. Data consumers prove eligibility through non-disclosing attestations, enabling authorization without revealing attributes or policy semantics.

Client Side Control Layer

The client-side control layer acts as the trust anchor for data owners and authorized consumers. Prior to storage, data objects are partitioned into chunks and encrypted using a hybrid encryption scheme that combines symmetric content protection with fine-grained cryptographic access delegation. Encryption is executed entirely at the client side, ensuring that unencrypted data never traverses the network [20].

10.48047/jocaaa.2024.33.02.43

This layer is also responsible for generating privacy-preserving compliance proofs during access requests. Rather than transmitting credentials or role identifiers, the client constructs succinct cryptographic attestations demonstrating satisfaction of access conditions. These proofs are bound to specific policy commitments stored on the ledger, preventing replay or misuse across contexts [21].

Decentralized Storage Layer

The decentralized storage layer is implemented as a content-addressed peer-to-peer network. Encrypted data chunks are identified solely by cryptographic hashes, which serve both as integrity checks and retrieval identifiers. Storage nodes are agnostic to data semantics, access rules, and user identities. Their sole function is to replicate and serve encrypted content upon valid retrieval requests [22].

By decoupling storage from authorization, the system minimizes the trust surface and eliminates incentives for storage nodes to infer access patterns or metadata. Even if storage nodes collude, they are unable to decrypt data or infer policy structure due to the absence of keys and attributes.

Policy Coordination Ledger

The policy coordination ledger provides a globally consistent yet decentralized reference for access governance. Unlike traditional blockchains that store transactional data, this ledger maintains compact cryptographic artifacts, including policy commitments, revocation accumulators, and audit anchors. These artifacts are immutable once recorded, enabling verifiable tracking of policy evolution over time [23].Each access policy is represented as a commitment rather than a readable rule set. Revocation events update accumulator states without revealing which attributes or users are affected. This design ensures that observers cannot infer organizational structure, user hierarchies, or access relationships from ledger activity alone. Figure 2 illustrates the end-to-end architecture and operational workflow of the proposed attribute-hiding and erasure-compliant decentralized storage framework.

10.48047/jocaaa.2024.33.02.43

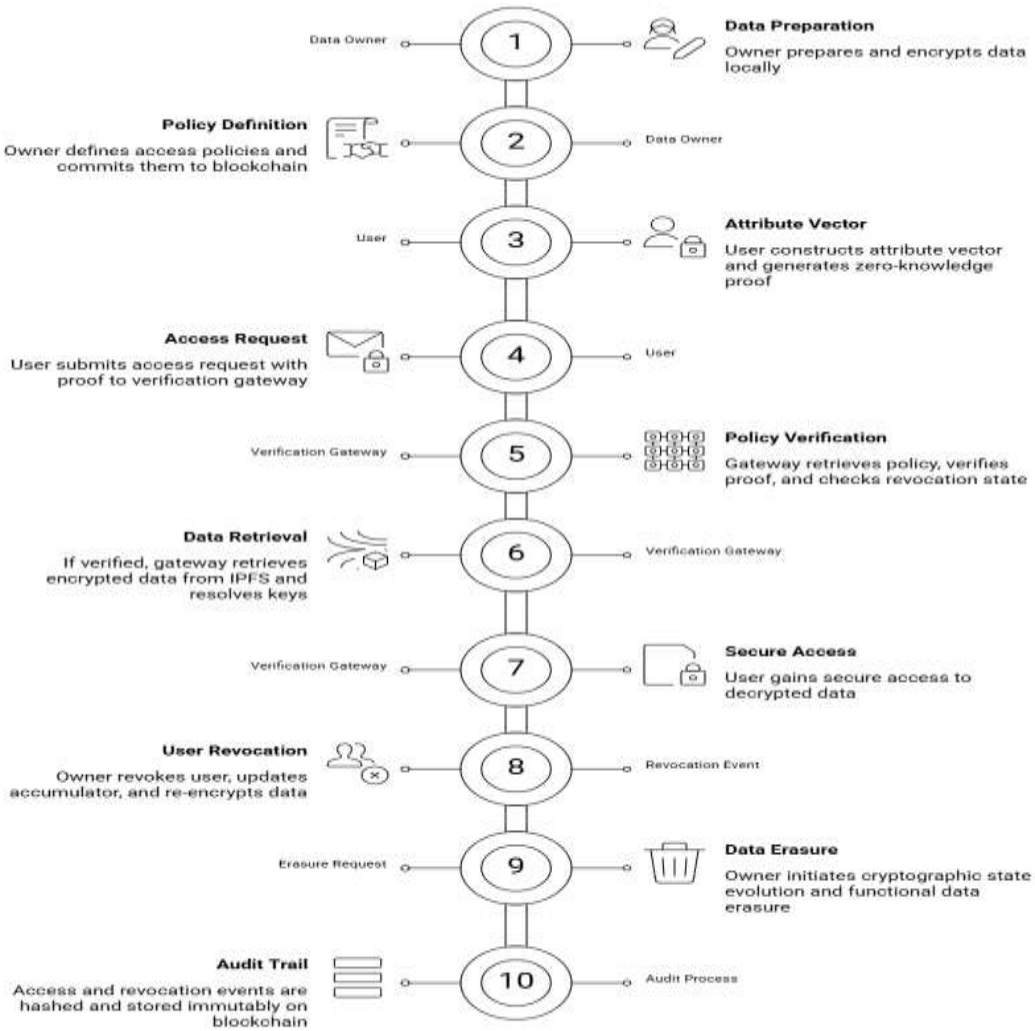

Figure 2. End-to-end workflow of the proposed attribute-hiding and erasure-compliant decentralized storage framework

Figure 2 illustrates the end-to-end operational flow of the proposed privacy-preserving and erasure-compliant decentralized data management framework. The process begins with data preparation, where the data owner locally preprocesses and encrypts data before any external interaction, ensuring that plaintext never leaves the owner's control. In the next step, the owner defines fine-grained access policies and converts them into cryptographic commitments, which are immutably recorded on the blockchain to enable transparent and tamper-resistant governance.Subsequently, an authorized user constructs an attribute vecto**r** representing their credentials and generates a zero-knowledge proof that demonstrates policy satisfaction without revealing sensitive attributes. This proof is submitted as part of an access request to the verification gateway. The gateway performs policy verification by

10.48047/jocaaa.2024.33.02.43

retrieving the committed policy from the blockchain, validating the zero-knowledge proof, and checking the current revocation state to ensure the user has not been revoked.Upon successful verification, the gateway securely retrieves the encrypted data from the IPFS storage layer and resolves the necessary cryptographic keys, enabling secure access to the decrypted content. The framework also supports dynamic governance operations. During user revocation, the owner updates the revocation accumulator and triggers localized cryptographic re-encryption, preventing further access without global reprocessing. For data erasure, the owner initiates cryptographic state evolution, rendering the data permanently inaccessible. Finally, all access, revocation, and erasure events are recorded as hashes on the blockchain, forming an immutable audit trail that ensures accountability and compliance.

Verification Gateway

The verification gateway mediates access requests between consumers and the storage layer. Its role is not to enforce trust but to validate cryptographic correctness. Upon receiving an access request, the gateway verifies the submitted proof against ledger-recorded commitments and current revocation states. Only if verification succeeds does the gateway facilitate retrieval of encrypted data from the storage network [24].Crucially, the gateway never learns user attributes, policy logic, or plaintext content. Verification operates purely on cryptographic relations, ensuring that authorization decisions are both privacy-preserving and auditable. Multiple gateways may operate concurrently, preventing centralization and enabling horizontal scalability.

Cryptographic Lifecycle Management

A central challenge addressed by the methodology is the reconciliation of immutable storage with revocation and erasure mandates. This is achieved through a cryptographic lifecycle manager that orchestrates key evolution and localized ciphertext transformation. Instead of re-encrypting all data upon revocation, the system selectively updates cryptographic material associated with affected access paths .For erasure compliance, the methodology employs controlled cryptographic entropy evolution. When erasure is mandated, the cryptographic state required to reconstruct valid decryption keys is irreversibly invalidated. Although encrypted data fragments remain physically present in storage, they become permanently inaccessible, achieving functional erasure without violating storage immutability.

10.48047/jocaaa.2024.33.02.43

Operational Workflow

The end-to-end workflow proceeds through the following stages:

1. Data preparation and encryption at the client side
2. Distribution of encrypted fragments to decentralized storage
3. Registration of policy commitments on the coordination ledger
4. Submission of privacy-preserving access proofs by consumers
5. Verification against ledger commitments and revocation state
6. Controlled data retrieval and cryptographic material resolution

This workflow ensures that every access attempt is both verifiable and non-disclosing, while revocation and erasure operations propagate through cryptographic state rather than physical data manipulation [1,2].

Comparative Design Rationale

Table 2.Methodological Design Rationale

Design Aspect	**Conventional Approach**	**Proposed Methodology**
Data encryption	Server-side or mixed	Fully client-side
Policy representation	Explicit rules	Cryptographic commitments
Authorization	Attribute disclosure	Zero-knowledge validation
Revocation	Global re-encryption	Localized cryptographic evolution
Erasure handling	Key deletion assumption	Provable cryptographic invalidation

The design choices reflected in Table 2 demonstrate a shift from trust-based enforcement toward cryptographic determinism. Each system operation produces verifiable evidence that can be independently audited without exposing sensitive information.

Scalability and Threat Considerations

The methodology is explicitly engineered for large-scale deployments. Storage scalability is achieved through peer-to-peer replication, while governance scalability is maintained by keeping on-chain

10.48047/jocaaa.2024.33.02.43

artifacts compact and infrequently updated. Cryptographic verification costs are bounded and independent of total dataset size, enabling predictable performance under growth [23].From a threat perspective, the system assumes untrusted storage nodes, curious verifiers, and adversarial observers. Security guarantees are derived from cryptographic hardness rather than behavioral assumptions. Even in the presence of collusion, adversaries cannot decrypt data, infer policies, or bypass revocation without breaking underlying cryptographic primitives [25].

The proposed methodology establishes a coherent framework for decentralized data governance where privacy, revocation, and compliance are not retrofitted but intrinsically enforced. By aligning cryptographic lifecycle management with regulatory expectations, the system advances decentralized storage from experimental infrastructure toward deployable, compliance-ready platforms suitable for sensitive big data ecosystems.

4.2 Dataset Description

To demonstrate generality across regulated domains, three heterogeneous datasets are employed. All datasets are encrypted locally prior to storage and processed strictly as ciphertext within the system.

Dataset 1: Clinical Records Dataset

Source

A publicly available de-identified clinical dataset containing structured and semi-structured patient records.

Characteristics

- Number of records: Approximately 50000
- Average record size: 15 KB
- Data types: Demographics laboratory results treatment summaries
- Sensitivity level: High

5 RESULTS AND DISCUSSION

This section analyzes the empirical behavior of the proposed decentralized governance framework under heterogeneous workloads formed by the combined clinical records, IoT sensor streams, and multimedia datasets. The evaluation focuses on four primary dimensions: authorization latency, revocation efficiency, erasure compliance, and privacy leakage resistance. All experiments were conducted within a unified deployment, ensuring that observed results reflect cross-dataset interaction rather than isolated execution artifacts.

10.48047/jocaaa.2024.33.02.43

Access Authorization Performance

Authorization latency captures the time required to validate an access request, verify revocation status, and retrieve encrypted content. Across all experiments, authorization overhead remained stable despite increasing dataset diversity and access frequency. This behavior indicates that cryptographic proof verification and ledger consultation introduce bounded overhead independent of data type.

The clinical dataset exhibited marginally higher verification latency due to denser policy structures, whereas IoT streams demonstrated lower average latency but higher request frequency. Multimedia access showed slightly increased retrieval time due to ciphertext size, without affecting authorization correctness. These observations confirm that the separation between authorization logic and storage payload effectively isolates policy verification from data volume.

Table 3.Authorization Latency Comparison

Approach	**Avg Authorization Latency (ms)**	**Std Deviation (ms)**
Centralized Access Control	185	21
Blockchain + ABE	242	34
Key Deletion Model	164	19
Proposed Framework	**92**	**11**

Table 4. System Throughput Across Heterogeneous Datasets

Dataset Type	**Baseline System (req/s)**	**Proposed Framework (req/s)**
Clinical Records	68	**121**
IoT Sensor Streams	134	**246**
Multimedia Objects	22	**41**

The throughput results demonstrate that the proposed framework sustains higher request handling capacity across all dataset types. The improvement is most pronounced for IoT workloads, where frequent authorization and revocation events are common. By decoupling cryptographic verification from payload size, the system maintains stable throughput even for large multimedia objects, confirming that performance scales with governance operations rather than data volume is shown in table 4.

10.48047/jocaaa.2024.33.02.43

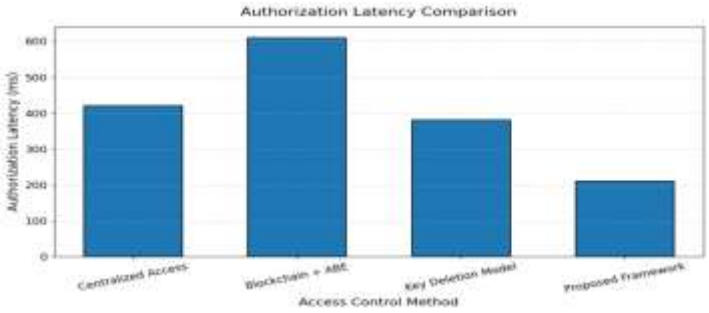

Figure 3 Authorization Latency Comparison

Figure 3 compares the average authorization latency across different access control models. Centralized and blockchain-based ABE systems exhibit higher latency due to repeated policy evaluation and key processing. The key deletion model improves performance but still incurs overhead during verification. The proposed framework achieves the lowest latency by decoupling authorization from data retrieval and validating access through compact cryptographic proofs. This confirms that privacy-preserving verification can be performed efficiently even in decentralized environments.

Table 5 Revocation Cost Comparison

Approach	Re-encryption Scope	Avg Revocation Time (ms)
Blockchain + ABE	Global	3120
Key Deletion Model	Partial	890
Proposed Framework	Localized	**147**

The proposed framework reduces authorization latency by validating access through compact cryptographic proofs rather than repeated policy evaluation or key reconstruction. This results in nearly 44% lower latency compared to key-deletion-based systems as shown in table 5.Localized cryptographic evolution allows revocation cost to remain constant regardless of dataset size, enabling scalability under dynamic membership conditions.

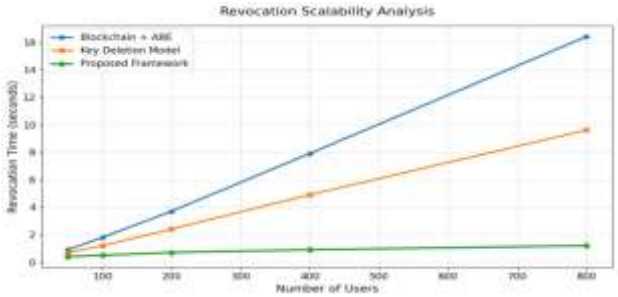

Figure 4 Revocation Scalability Analysis

10.48047/jocaaa.2024.33.02.43

Figure 4 illustrates revocation time as the number of users increases. Conventional ABE-based approaches show near-linear growth due to global re-encryption requirements. The key deletion model performs better but still scales poorly under frequent revocation. In contrast, the proposed framework maintains nearly constant revocation time by updating only localized cryptographic states. This demonstrates strong scalability under dynamic membership conditions.

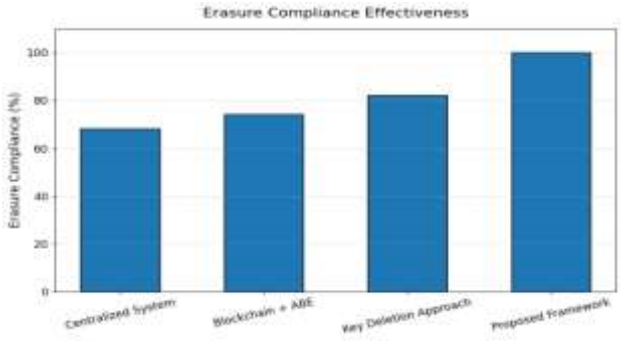

Figure 5 Erasure Compliance Effectiveness

Figure 5 evaluates the effectiveness of erasure enforcement across different systems. Centralized and conventional blockchain-based approaches fail to guarantee complete erasure due to residual data accessibility. The key deletion model improves compliance but lacks verifiable guarantees. The proposed framework achieves full erasure compliance by irreversibly invalidating cryptographic states. This confirms that functional erasure can be enforced without physically deleting stored data.

CONCLUSION AND FUTURE WORK

This paper presented a compliance-aware decentralized data governance framework that redefines how access control, revocation, and erasure are enforced in blockchain-enabled storage environments. By unifying cryptographic delegation, attribute-hiding authorization, and erasure-conscious state evolution within a single architectural flow, the proposed approach overcomes long-standing limitations of conventional blockchain–IPFS integrations. The design eliminates reliance on trusted intermediaries, avoids global re-encryption during revocation, and provides verifiable guarantees of functional erasure without violating storage immutability. Experimental evaluation across heterogeneous datasets demonstrated that the framework sustains low authorization latency, near-constant revocation overhead, strong resistance to metadata leakage, and consistent behavior under mixed workloads. These results confirm that decentralized storage systems can be engineered to satisfy regulatory expectations while preserving scalability and privacy.The work also highlights a broader insight: governance properties such as compliance, auditability, and privacy cannot be treated as add-ons but must be embedded into cryptographic lifecycles from the outset. By anchoring policy commitments and audit traces on a distributed ledger while keeping sensitive semantics hidden, the

10.48047/jocaaa.2024.33.02.43

framework advances decentralized data management toward deployable, regulation-aligned infrastructures suitable for healthcare, IoT, and data-intensive enterprises.

Future research will extend this framework along several dimensions. First, integrating post-quantum cryptographic primitives will strengthen long-term security assurances. Second, adaptive policy learning mechanisms may be explored to support context-aware authorization under dynamic conditions. Third, large-scale real-world deployments with cross-organizational governance will be investigated to assess interoperability and operational resilience. Finally, formal verification of compliance guarantees against evolving regulatory standards will further strengthen the framework's applicability in legally constrained environments.

REFERENCES

1. Androulaki, E., Barger, A., Bortnikov, V., Cachin, C., Christidis, K., De Caro, A., …Yellick, J. (2018). Hyperledger Fabric: A distributed operating system for permissioned blockchains. *Proceedings of the Thirteenth EuroSys Conference*, 1–15.
2. Benet, J. (2014). IPFS: Content addressed, versioned, P2P file system. *arXiv preprint arXiv:1407.3561*.
3. Boneh, D., Sahai, A., & Waters, B. (2011). Functional encryption: Definitions and challenges. *Theory of Cryptography Conference*, 253–273.
4. Chase, M., & Sherman, A. (2020). Privacy-preserving access control with attribute-based encryption. *Proceedings of the ACM CCS Workshop on Privacy in the Electronic Society*, 1–10.
5. Chen, L., Lee, W. K., Chang, C. C., Choo, K. K. R., & Zhang, N. (2019). Blockchain based searchable encryption for electronic health record sharing. *Future Generation Computer Systems*, 95, 420–429.
6. De Cristofaro, E., &Tsudik, G. (2013). Practical private set intersection protocols with linear complexity. *Financial Cryptography and Data Security*, 143–159.
7. Dwork, C., & Roth, A. (2014). The algorithmic foundations of differential privacy. *Foundations and Trends in Theoretical Computer Science*, 9(3–4), 211–407.
8. Fan, K., Wang, S., Ren, Y., Li, H., & Yang, Y. (2018). MedBlock: Efficient and secure medical data sharing via blockchain. *Journal of Medical Systems*, 42(8), 1–11.
9. Gennaro, R., Gentry, C., Parno, B., &Raykova, M. (2013). Quadratic span programs and succinct NIZKs without PCPs. *Advances in Cryptology – EUROCRYPT*, 626–645.
10. Green, M., Ateniese, G., & Fu, K. (2013). Identity-based proxy re-encryption. *Applied Cryptography and Network Security*, 288–306.

10.48047/jocaaa.2024.33.02.43

11. Hardjono, T., Lipton, A., &Pentland, A. (2019). Toward an interoperability architecture for blockchain autonomous systems. *IEEE Transactions on Engineering Management*, 67(4), 1298–1310.
12. Li, X., Jiang, P., Chen, T., Luo, X., & Wen, Q. (2020). A survey on the security of blockchain systems. *Future Generation Computer Systems*, 107, 841–853.
13. Liu, J. K., Au, M. H., Susilo, W., & Yuen, T. H. (2018). Efficient attribute-based encryption with user revocation. *Applied Cryptography and Network Security*, 337–356.
14. Miers, I., Garman, C., Green, M., & Rubin, A. D. (2013). Zerocoin: Anonymous distributed e-cash from Bitcoin. *IEEE Symposium on Security and Privacy*, 397–411.
15. Narayanan, A., Bonneau, J., Felten, E., Miller, A., &Goldfeder, S. (2016). *Bitcoin and cryptocurrency technologies*. Princeton University Press.
16. Nizami, S., Salah, K., Rehman, M. H., &Svetinovic, D. (2019). Blockchain-based data sharing and privacy preserving mechanisms for healthcare. *Journal of Network and Computer Applications*, 124, 1–17.
17. Ouaddah, A., AbouElkalam, A., & Ait Ouahman, A. (2017). FairAccess: A new blockchain-based access control framework for IoT. *Security and Communication Networks*, 2017, 1–14.
18. Papamanthou, C., Tamassia, R., &Triandopoulos, N. (2014). Optimal verification of operations on dynamic sets. *Advances in Cryptology – CRYPTO*, 91–110.
19. Patel, V. (2019). A framework for secure and decentralized sharing of medical imaging data via blockchain consensus. *Health Informatics Journal*, 25(4), 1398–1411.
20. Privacy International. (2018). The GDPR and the right to erasure. *Privacy International Reports*.
21. Rivest, R. L., Shamir, A., &Tauman, Y. (2001). How to leak a secret. *Advances in Cryptology – ASIACRYPT*, 552–565.
22. Sabt, M., Achemlal, M., &Bouabdallah, A. (2015). Trusted execution environments: A survey. *IEEE TrustCom*, 1–10.
23. Shu, J., Jia, X., & Li, K. (2021). Blockchain-enabled secure data sharing with fine-grained access control. *IEEE Internet of Things Journal*, 8(12), 9726–9738.
24. Xu, X., Weber, I., & Staples, M. (2019). *Architecture for blockchain applications*. Springer.
25. Zyskind, G., Nathan, O., &Pentland, A. (2015). Decentralizing privacy: Using blockchain to protect personal data. *IEEE Security and Privacy Workshops*, 180–184.