10.48047/jocaaa.2021.29.03.16

Enhancing Data Security in Financial Institutions Through Lightweight Blockchain Protocols: A Scalable and Secure Framework

Adarsh Naidu
Independent Researcher, USA
Email : adarshnaidu2910@gmail.com

Abstract

As financial institutions undergo rapid digital transformation, the protection of sensitive data from cyber threats has become increasingly critical. While blockchain technology offers immutable and decentralized data handling, traditional blockchain systems are often resource-intensive and unsuitable for real-time financial operations. This paper explores the integration of lightweight blockchain protocols as a secure and efficient alternative for enhancing data security in financial institutions. The study analyzes various lightweight consensus mechanisms and cryptographic methods that reduce computational overhead while maintaining high levels of integrity, auditability, and regulatory compliance. A security framework is proposed to demonstrate practical applications of these protocols in financial environments. The results indicate that lightweight blockchain solutions can significantly improve data protection and system resilience without the scalability issues commonly associated with conventional blockchains.

Index Terms

Lightweight blockchain, data security, financial institutions, secure transactions, distributed ledger technology (DLT), consensus mechanisms, cryptography, financial technology (FinTech), regulatory compliance.

I. Introduction

The financial sector handles vast volumes of sensitive data, ranging from customer identities and transaction records to regulatory filings and interbank communications. Ensuring the security, integrity, and confidentiality of this data is paramount, as breaches can result in severe financial losses, regulatory penalties, and erosion of customer trust. In recent years, cyber threats have become increasingly sophisticated, targeting vulnerabilities in centralized data systems typically employed by financial institutions.

To counter these risks, various cybersecurity strategies have been implemented, including firewalls, encryption, multi-factor authentication, and intrusion detection systems. However, traditional security models often suffer from centralized points of failure, limited transparency, and difficulties in establishing tamper-proof audit trails. As a result, there is a growing need for innovative solutions that provide decentralized, verifiable, and secure data handling capabilities.

10.48047/jocaaa.2021.29.03.16

Blockchain technology has emerged as a potential game-changer in this domain, offering immutable records, decentralized consensus, and robust cryptographic foundations. Despite its promise, mainstream blockchain platforms such as Bitcoin and Ethereum present challenges for financial institutions due to their high energy consumption, latency, and resource-intensive consensus mechanisms like Proof of Work (PoW).

This paper investigates the application of **lightweight blockchain protocols** a class of blockchain systems designed for efficiency and scalability to enhance data security in financial institutions. These protocols typically employ optimized consensus algorithms such as Proof of Authority (PoA), Delegated Proof of Stake (DPoS), or Byzantine Fault Tolerance (BFT), which require significantly less computational power and bandwidth.

The primary objectives of this study are:

- To explore the architectural design and security benefits of lightweight blockchain systems;
- To assess their integration potential within financial infrastructures;
- To propose a scalable and secure framework tailored to the operational and regulatory needs of financial institutions.

By leveraging lightweight blockchain protocols, financial entities can achieve enhanced data protection, real-time auditability, and improved resilience against internal and external threats without compromising system performance or compliance requirements.

II. Literature Review

The adoption of blockchain technology in the financial sector has been extensively explored as a means to address issues related to data integrity, transparency, and security. Traditional centralized systems, though widely used, present inherent limitations such as single points of failure and poor auditability [1], [2]. These weaknesses have encouraged researchers and institutions to consider distributed ledger technologies (DLT) as potential alternatives.

Early work by Nakamoto [3] introduced blockchain as a decentralized and tamper-proof system via Bitcoin, sparking a wave of interest across domains. However, conventional blockchains using Proof of Work (PoW) are computationally intensive and unsuitable for high-throughput financial applications [4]–[6]. Ethereum's transition towards Proof of Stake (PoS) with Ethereum 2.0 further highlighted the need for energy-efficient consensus mechanisms [7].

Lightweight blockchain protocols have emerged as alternatives to overcome these challenges. Protocols such as Tendermint [8], IOTA [9], Nano [10], and Algorand [11] focus on reducing resource usage while maintaining security and consensus efficiency. Research shows that these protocols are particularly promising for sectors like finance, where low latency and high scalability are critical [12], [13].

In the financial context, blockchain has been investigated for secure Know Your Customer (KYC) processes [14], anti-money laundering (AML) compliance [15], interbank settlements [16], and fraud detection [17]. Studies also explore the integration of blockchain into legacy financial infrastructure [18], [19], and the challenges associated with interoperability [20], [21].

10.48047/jocaaa.2021.29.03.16

While some works focus on cryptographic innovations [22], others address regulatory implications of blockchain in banking [23], [24]. Hybrid models, combining permissioned and public chains, have also been proposed for financial data sharing across institutions [25], [26].

Recent work on privacy-preserving blockchain solutions, including zero-knowledge proofs and secure multiparty computation, suggests pathways to enhance confidentiality in financial applications [27]–[29]. Moreover, advancements in post-quantum cryptography are influencing the design of future-ready blockchain protocols [30].

This paper builds upon these foundational studies to propose a customized lightweight blockchain framework optimized for financial institutions, addressing their unique regulatory, operational, and security requirements.

III. Methodology

To address the rising concerns around data security in financial institutions, this paper proposes a novel framework that leverages lightweight blockchain protocols for secure, efficient, and regulatory-compliant data management. The methodology is designed to align with core financial requirements, such as high transaction throughput, low latency, strict compliance obligations, and seamless integration with existing legacy systems.

The primary objective of the proposed system is to enhance data confidentiality, integrity, and availability without incurring the high computational costs typically associated with conventional blockchain platforms such as Bitcoin or Ethereum. This is achieved through the adoption of lightweight blockchain protocols that utilize resource-efficient consensus algorithms like Proof of Authority (PoA), Delegated Proof of Stake (DPoS), or Practical Byzantine Fault Tolerance (PBFT). These mechanisms eliminate the need for energy-intensive mining while maintaining a high degree of trust, decentralization, and immutability.

The architecture of the proposed system consists of multiple interconnected layers. At the base is the data input layer, responsible for ingesting customer data, transaction records, and regulatory logs. Above it lies the blockchain middleware, which serves as the backbone of the security infrastructure. This layer utilizes a lightweight blockchain to store hashed records of all critical operations, thereby ensuring that no tampering can occur without detection. Smart contracts operate within this layer to enforce compliance policies automatically, such as Know Your Customer (KYC) checks, anti-money laundering (AML) flags, or transaction verification rules. All sensitive data is encrypted using AES-256 encryption at rest and secured with TLS 1.3 during transit.

An integration layer ensures interoperability with existing banking systems, such as core banking modules, SWIFT networks, and compliance databases. This allows financial institutions to adopt the proposed blockchain system without completely overhauling their infrastructure. Furthermore, APIs and middleware connectors facilitate secure communication between internal services and external regulatory bodies. Finally, a compliance and monitoring layer provides real-time audit trails, access logs, and automated report generation, helping banks meet internal governance and external audit requirements.

The proposed framework employs a permissioned blockchain model using PoA consensus, wherein only authorized validator nodes such as banks, regulators, or certified auditors can participate in block validation. This ensures both security and accountability, making the

10.48047/jocaaa.2021.29.03.16

system suitable for regulated environments. In addition, privacy-enhancing techniques such as zero-knowledge proofs and role-based access control are integrated to provide confidential transaction processing while enabling selective transparency for auditing purposes.

Deployment of the system is envisioned in phases, beginning with a pilot implementation focused on secure KYC data sharing among a consortium of banks. Key performance indicators such as transaction latency, throughput (TPS), storage efficiency, and regulatory adherence will be monitored and evaluated. Based on pilot success, the framework can be scaled to other applications like real-time fraud detection, secure interbank transfers, and automated regulatory reporting.

Through this multi-layered, scalable, and secure methodology, the framework aims to bridge the gap between the promise of blockchain and the practical constraints of the financial sector. It leverages the benefits of distributed ledger technology while maintaining a lean resource footprint, offering a viable pathway for modernizing financial data security infrastructures.

IV. Use Cases and Applications

The practical application of lightweight blockchain protocols within financial institutions offers several transformative opportunities across various operational domains. By leveraging resource-efficient consensus mechanisms and modular architectures, the proposed framework can be integrated into key areas of financial operations that are traditionally prone to security risks, inefficiencies, or regulatory friction.

One of the most compelling use cases lies in **Know Your Customer (KYC) and Anti-Money Laundering (AML) compliance**. Financial institutions are required to continuously verify and update customer identity information, often duplicating efforts across multiple banks. A lightweight blockchain network can be deployed as a permissioned, consortium-based ledger where verified KYC data is securely stored and cryptographically hashed. Participating institutions can access and validate customer identities in real time, significantly reducing onboarding times and improving regulatory compliance. Smart contracts can automate KYC rule enforcement, trigger alerts for suspicious activities, and maintain immutable audit trails, thereby reducing operational cost and human error.

Another critical application is in **interbank reconciliation and settlements**. Traditional settlement processes involve delayed batch processing, manual matching of records, and reliance on third-party intermediaries. By adopting a lightweight blockchain protocol like Tendermint or Algorand, financial institutions can achieve near-instantaneous, peer-to-peer settlement of transactions. Each transaction is securely recorded on the blockchain, validated by authorized nodes, and automatically reconciled using pre-defined smart contract logic. This not only accelerates settlement cycles but also minimizes the risk of fraud and duplication.

Regulatory reporting and auditability is another area where lightweight blockchain can add significant value. The immutable nature of blockchain records ensures that transaction data cannot be tampered with, thus enabling automatic generation of audit trails. Regulators can be granted read-only access to specific nodes on the network, ensuring continuous, transparent oversight without compromising customer confidentiality. Furthermore, smart contracts can be programmed to monitor compliance thresholds such as transaction limits or flagged patterns and report them directly to regulatory authorities, reducing the burden on compliance officers.

10.48047/jocaaa.2021.29.03.16

In the domain of **cross-border payments**, lightweight blockchains offer a scalable and cost-effective alternative to the current SWIFT-based infrastructure. By eliminating intermediaries and providing real-time currency conversion through tokenization mechanisms, the proposed system can reduce transaction fees and settlement times, especially for underbanked regions or small financial service providers.

Finally, **internal data access and integrity control** within financial institutions can also benefit. Blockchain-based logging systems ensure that any access to sensitive data whether by employees, third-party vendors, or automated processes is securely recorded with time-stamped, cryptographically verifiable logs. This enhances internal cybersecurity, deters insider threats, and supports forensic investigations in the event of a breach.

These use cases demonstrate that lightweight blockchain protocols are not merely theoretical constructs, but practical solutions to long-standing challenges in the financial industry. Their low-resource footprint, combined with high scalability and security, makes them ideally suited for deployment in real-world financial environments where both performance and trust are paramount.

V. Benefits and Challenges

The integration of lightweight blockchain protocols into financial institutions offers a multitude of benefits that align with modern security, operational, and regulatory needs. However, it also presents certain challenges that must be acknowledged to ensure realistic expectations and effective implementation.

A. Benefits

One of the most significant advantages of using lightweight blockchain protocols is **enhanced data security**. These protocols provide immutable ledgers where every transaction is cryptographically secured and verifiable. This immutability prevents unauthorized modifications to transaction records, significantly reducing the risk of internal fraud or data tampering. Additionally, the use of asymmetric encryption, digital signatures, and hash functions ensures that sensitive financial data remains confidential and tamper-proof.

Another key benefit is **operational efficiency**. Traditional blockchain platforms like Bitcoin or Ethereum consume substantial computational resources due to consensus mechanisms such as Proof of Work (PoW). In contrast, lightweight protocols such as Proof of Authority (PoA) or Practical Byzantine Fault Tolerance (PBFT) are significantly faster and more energy-efficient, enabling real-time transaction processing without the overhead of mining. This efficiency is especially beneficial in high-frequency financial environments such as stock trading, interbank settlements, or real-time payment systems.

Regulatory compliance and auditability are also greatly improved through blockchain's transparent and traceable nature. Immutable records allow financial institutions to maintain consistent, automated audit logs, which can be accessed by regulatory bodies when needed. Smart contracts can further enforce compliance rules dynamically, reducing the need for manual intervention in processes like KYC validation, anti-money laundering checks, and transaction monitoring.

10.48047/jocaaa.2021.29.03.16

Scalability and **interoperability** are also enhanced through lightweight design. These protocols can be deployed in modular fashion, allowing institutions to start with pilot projects and gradually scale across departments or even cross-institutional networks. APIs and middleware solutions can bridge blockchain systems with legacy infrastructure, ensuring continuity of operations without massive structural overhauls.

B. Challenges

Despite these benefits, there are also several challenges associated with implementing lightweight blockchain systems in financial institutions. One of the foremost concerns is **interoperability with existing systems**. Many core banking platforms are not designed to interact with distributed ledger technology, and integrating the two may require complex customization or middleware development.

Regulatory uncertainty is another critical issue. Although blockchain is increasingly being explored for compliance purposes, financial regulators in many jurisdictions have yet to fully define frameworks for its use. Ambiguities around data sovereignty, legal recognition of smart contracts, and cross-border data sharing could hinder adoption, particularly in highly regulated markets.

In addition, the **governance of permissioned blockchains** where validator nodes are limited to trusted parties can raise questions of centralization and trust. For instance, if only a few financial institutions control the validation process, the system could be perceived as vulnerable to collusion or censorship. Establishing transparent governance policies and consensus mechanisms is essential to maintain stakeholder confidence.

Privacy concerns also persist, particularly when sensitive financial or personal data is recorded on a shared ledger. Even though lightweight protocols often employ privacy-preserving techniques, striking the right balance between transparency and confidentiality remains a technical and ethical challenge.

VI. System Model and Threat Model

This section defines the system model and threat assumptions for deploying lightweight blockchain protocols in financial institutions. The goal is to clearly specify participating entities, trust boundaries, protected assets, and the attacker capabilities considered in evaluating the security of the proposed framework.

A. System Model

The proposed framework operates in a **permissioned consortium blockchain** setting, where participating organizations are known and vetted (e.g., commercial banks, payment processors, and regulatory authorities). The system is composed of four primary entity types: **(i) client applications**, **(ii) institutional service nodes**, **(iii) validator nodes**, and **(iv) oversight/audit nodes**. Client applications represent internal banking systems and interfaces (e.g., core banking modules, KYC portals, treasury systems) that generate transactions such as KYC updates, transaction logs, compliance events, and access records. Institutional service nodes act as middleware gateways exposing authenticated APIs, performing input validation, and enforcing encryption before any event is committed to the ledger.

10.48047/jocaaa.2021.29.03.16

The blockchain network is maintained by **validator nodes** operated by authorized consortium members. Validators participate in lightweight consensus (e.g., PoA/PBFT-style) to agree on transaction ordering and block finalization. Because the network is permissioned, validator identities are managed through a governance policy (e.g., admission rules, certificate issuance, key rotation, and removal procedures). In addition, dedicated **audit/oversight nodes**—typically regulators or internal audit departments—may hold read-only or restricted query access to support continuous auditing. This enables verifiable compliance checks without requiring full control over transaction validation.

To address confidentiality requirements, the framework follows a **hybrid storage model**. Sensitive financial records (e.g., complete KYC documents, customer PII, detailed transaction payloads) are stored **off-chain** in secure institutional databases or encrypted storage. The blockchain stores **cryptographic commitments** (hashes), metadata (timestamps, record identifiers), and policy proofs needed for integrity verification and auditability. Smart contracts implement rule-based controls such as permitted write operations, role-based access constraints for queries, and compliance triggers (e.g., suspicious activity flags).

B. Trust Assumptions and Boundaries

The framework assumes that consortium members are legally accountable entities, but not necessarily fully trusted in all circumstances. Specifically, the system assumes: (i) secure cryptographic primitives (hashing, digital signatures, encryption) are not broken; (ii) private keys are protected via institutional key management practices; and (iii) the consensus protocol tolerates a bounded number of faulty or malicious validators depending on the selected mechanism (e.g., BFT-based settings tolerate a fraction of byzantine nodes). Communication channels between nodes are assumed to be protected using modern transport security (e.g., TLS), but the threat model still considers active network attackers attempting replay or disruption.

Trust boundaries exist between (i) client applications and the middleware/API layer, (ii) middleware and validator nodes, and (iii) blockchain data and off-chain repositories. Since confidentiality requirements are strict in finance, the blockchain is treated primarily as an integrity and audit layer, while private data remains encrypted and controlled in off-chain systems.

C. Adversary Model

The adversary is modeled across multiple realistic threat categories relevant to financial environments:

1. **External Network Attacker:** An attacker who can intercept, replay, delay, or inject network traffic but does not possess valid cryptographic credentials. The attacker may attempt man-in-the-middle (MITM) attacks, denial-of-service (DoS), or message replay.
2. **Malicious Insider:** A legitimate user (employee, contractor, or compromised internal service) with authorized access to certain systems who attempts unauthorized data modification, fraudulent record insertion, or abuse of access rights.
3. **Compromised Endpoint/Client:** A client machine or banking application is compromised by malware, enabling unauthorized requests, credential theft, or fraudulent transaction generation.

10.48047/jocaaa.2021.29.03.16

4. **Byzantine Validator(s):** One or more validator nodes behave maliciously (e.g., equivocation, censorship, attempting to reorder transactions, or colluding to approve invalid state updates). This includes the possibility of **partial collusion** among consortium members.
5. **Data Store Adversary:** An attacker gains access to off-chain storage and attempts to alter or delete stored records, hoping to avoid detection through traditional audit mechanisms.

The adversary's objective may include tampering with transaction history, bypassing compliance checks, leaking sensitive data, or degrading system availability.

D. Security Goals

Under the above system and adversary models, the framework targets the following security properties:

- **Integrity:** Financial logs and compliance records must be tamper-evident; any change to off-chain data should be detectable through on-chain commitments.
- **Authenticity and Non-Repudiation:** Every write operation must be attributable to a valid identity through digital signatures, preventing denial of action after submission.
- **Confidentiality:** Sensitive customer and transaction data must remain protected through encryption and access controls, with minimal sensitive exposure on-chain.
- **Availability and Resilience:** The system should maintain operation despite network-level disruption attempts and tolerate a bounded number of faulty validators under the chosen consensus model.
- **Auditability and Traceability:** The system should provide immutable audit trails enabling internal and regulatory verification with time-stamped evidence.
- **Access Control Correctness:** Only authorized roles should be able to submit specific transactions or query restricted data, enforced through policy logic.

E. Threat Analysis and Mitigation Mapping

To mitigate **data tampering**, the framework stores hashes of sensitive off-chain records on-chain, ensuring any modification in the database produces a hash mismatch during verification. **Replay attacks** and message injection attempts are mitigated by including nonces/timestamps in signed requests and enforcing strict transaction uniqueness rules at the smart contract layer. **Insider abuse** is addressed through role-based policies enforced by smart contracts, separation of duties (e.g., maker-checker patterns), and immutable logging of all sensitive operations.

For **validator misbehavior**, the permissioned model restricts participation to certified entities and enables governance-level responses such as slashing (where applicable), revocation, and validator removal. In BFT-style consensus, safety is maintained as long as the byzantine validator fraction remains within tolerable bounds; censorship risk is reduced through multi-party validator distribution across independent institutions and periodic governance audits. **DoS threats** are mitigated through rate limiting at API gateways, traffic filtering, and redundancy across nodes, ensuring the system can degrade gracefully rather than fail entirely.

Finally, confidentiality risks are reduced by keeping sensitive payloads off-chain, encrypting at rest and in transit, and limiting on-chain exposure to commitments and minimal metadata.

10.48047/jocaaa.2021.29.03.16

Where selective disclosure is required for regulatory checks, privacy-preserving methods (e.g., policy proofs or controlled disclosure mechanisms) can be layered on top of the base framework to balance compliance with confidentiality.

VII. Proposed Lightweight Blockchain Security Framework

This section presents the proposed lightweight blockchain–based framework for enhancing data security in financial institutions. The framework is designed as a **permissioned, consortium-led architecture** that prioritizes integrity, auditability, and controlled confidentiality while keeping computational and storage overhead low. The key design choice is to store **sensitive financial data off-chain** and record only **cryptographic commitments and compliance evidence on-chain**, enabling tamper detection and verifiable audit trails without exposing customer information on a shared ledger.

A. Framework Overview

The proposed framework consists of four tightly coupled components: (i) **Institutional Applications** (core banking, KYC portals, payment services, AML monitoring), (ii) **Security Middleware/API Gateway**, (iii) **Permissioned Lightweight Blockchain Network**, and (iv) **Off-Chain Secure Data Repositories**. Institutional applications generate events such as KYC onboarding, transaction logs, access requests, and compliance flags. These events are routed through the middleware layer, which performs authentication, authorization checks, encryption, and formatting before submitting transactions to the blockchain. The blockchain network—maintained by authorized validators—stores tamper-evident commitments, timestamps, policy outcomes, and immutable audit logs. The full sensitive payload remains within the institution's secure storage, referenced by identifiers and hashes on-chain.

B. On-Chain and Off-Chain Data Model

To meet confidentiality requirements, the framework follows a hybrid data model. **Off-chain storage** contains the complete record (e.g., KYC documents, customer PII, detailed transaction payloads), stored in encrypted form and governed by the institution's access policies. **On-chain records** contain: a unique record identifier, a cryptographic hash of the off-chain payload, timestamps, submitting institution identity, and a signed policy decision (e.g., "KYC verified," "AML flagged," "access granted/denied"). This approach ensures that any unauthorized change to off-chain data is detectable because the recomputed hash will not match the immutable on-chain commitment.

C. Transaction Types and Smart Contract Policies

The framework defines standardized transaction types to support typical financial security workflows. A **KYC registration/update transaction** records a hash of the verified KYC package along with metadata such as verification level and issuing institution. An **access-log transaction** records immutable evidence of who accessed what resource and when, supporting forensic analysis and compliance audits. A **compliance event transaction** records AML triggers, sanctions screening results, or threshold violations without disclosing raw sensitive data publicly. A **settlement or reconciliation transaction** records commitments to interbank settlement steps and finalization checkpoints.

10.48047/jocaaa.2021.29.03.16

Smart contracts enforce rules over these transaction types. For example, KYC updates may require a maker-checker pattern where submission and approval must come from distinct authorized roles. Access-log writes may be enforced as mandatory for certain data categories, ensuring no sensitive operation happens without an immutable audit entry. Compliance contracts can enforce automatic escalation—such as flagging repeated threshold breaches—and can restrict write permissions so only approved compliance services can submit AML/sanctions outcomes.

D. Identity, Key Management, and Access Control

Because this is a permissioned financial setting, identity is managed through institutional membership and certificate-based authentication. Each institution and its internal services operate under cryptographic identities that sign all blockchain submissions. Access control is implemented in two layers. First, the middleware applies organizational policies (role-based or attribute-based checks) before accepting any request. Second, smart contracts enforce consortium-level authorization, ensuring that only permitted entities can write specific transaction types or query restricted ledger views. This dual-layer design reduces reliance on a single enforcement point and provides verifiable evidence that policy was applied.

Key management is handled through institutional controls such as hardware-backed key storage, rotation policies, and revocation procedures. When a key compromise is suspected, the governance layer can revoke certificates and exclude affected identities from transaction submission and, if necessary, remove compromised validators from the active set.

E. Consensus and Governance in a Lightweight Setting

The framework adopts lightweight consensus (e.g., PoA/PBFT-style) to achieve fast finality and low overhead. Validators are known consortium entities, such as banks or regulators, and are selected through governance rules. Governance defines admission criteria, validator rotation, quorum thresholds, and procedures for dispute handling and incident response. This is critical in finance because the security model must support accountability, regulatory oversight, and controlled membership, while still distributing trust across multiple institutions to reduce single points of failure.

F. End-to-End Operational Workflow

A typical secure record lifecycle in the proposed framework proceeds as follows. First, an institutional application generates a sensitive record (e.g., KYC package) and stores it in encrypted off-chain storage. Next, the middleware computes a hash of the encrypted payload and constructs an on-chain transaction containing the hash, metadata, and a digital signature. Validators finalize the transaction through lightweight consensus, producing an immutable ledger entry that acts as tamper-evident proof of the record's existence and state at a specific time. Later, during audits or disputes, the institution can retrieve the off-chain record, recompute the hash, and prove integrity by matching it against the on-chain commitment. This same workflow applies to access logs and compliance events, enabling continuous auditability and non-repudiation without exposing private payloads on the shared ledger.

G. Security Rationale of the Framework

10.48047/jocaaa.2021.29.03.16

The framework strengthens integrity by ensuring that critical financial events become tamper-evident via immutable commitments. It improves auditability by creating time-stamped, signed, and verifiable logs of sensitive operations. It maintains confidentiality by minimizing on-chain exposure and keeping sensitive payloads encrypted off-chain under institutional control. Finally, it remains practical for real-world financial environments by relying on lightweight consensus and permissioned governance, reducing energy cost and latency compared to proof-of-work systems while preserving verifiable trust and accountability.

VIII. Experimental Setup and Evaluation Plan

To assess the practicality and effectiveness of the proposed lightweight blockchain security framework for financial institutions, this section outlines a structured evaluation methodology. The evaluation focuses on quantifying performance overhead, validating security properties under realistic adversarial conditions, and measuring the framework's suitability for compliance-driven financial workflows such as KYC/AML logging, interbank reconciliation, and audit trail generation. Since financial systems require both strong integrity guarantees and high operational throughput, the proposed evaluation emphasizes end-to-end latency, transaction processing capacity, resource utilization, and tamper-detection reliability.

A. Prototype Implementation

A prototype implementation is planned using a permissioned lightweight blockchain stack that supports fast finality and consortium governance. The prototype will include the core components described in Sections VII and VIII: an API gateway/middleware responsible for authentication, authorization, encryption, hashing, and transaction submission; a smart-contract layer to enforce access-control and compliance policies; validator nodes running a lightweight consensus mechanism (e.g., PoA or PBFT-style finality); and an off-chain encrypted storage service holding sensitive payloads with only commitments and metadata written to the ledger. The prototype will implement at least three transaction types: (i) KYC record registration/update, (ii) access-log events for sensitive resources, and (iii) compliance-event logging (e.g., AML flags).

B. Experimental Environment

The experimental environment will be configured as a consortium network of multiple institutions to reflect realistic deployment conditions. A typical setup will include 6–10 nodes, including 4–7 validator nodes and 1–3 read-only audit/oversight nodes. Nodes will be deployed on virtual machines or containers with consistent compute profiles (e.g., multi-core CPU, moderate RAM, SSD-backed storage) under a controlled network with configurable latency and packet loss to emulate interbank connectivity. Multiple network conditions will be tested, including low-latency LAN-like conditions and higher-latency WAN-like conditions, because financial consortium networks often operate across geographically distributed institutions.

C. Workload Design and Test Scenarios

The evaluation workload will be designed to match common financial security operations rather than cryptocurrency-style transactions. The primary workload will include a mixture of (i) KYC onboarding/update events, (ii) access-log writes triggered by sensitive database reads, (iii) compliance-event writes generated by AML monitoring engines, and (iv) reconciliation checkpoints for settlement-like flows. To model realistic behavior, workloads will include

10.48047/jocaaa.2021.29.03.16

bursty traffic patterns (peak banking hours) and steady-state traffic patterns (background operations). Each experiment will be executed at increasing request rates to observe saturation points and stability under load. In addition, the size of metadata and the frequency of writes will be varied to test the effect of audit intensity on ledger growth and throughput.

D. Evaluation Metrics

Performance will be evaluated using metrics commonly expected in IEEE systems/security papers. Transaction throughput will be measured in transactions per second (TPS) under varying validator counts and workloads. End-to-end latency will be measured from the time a client submits a request to the time the transaction reaches finality, and will be reported using percentile values (e.g., median and 95th percentile) to capture tail latency. Resource overhead will be measured through CPU utilization, memory footprint, network bandwidth, and storage growth per day/week under specified logging policies. Because the framework uses an off-chain storage model, additional measurements will quantify off-chain encryption overhead and hash verification time during audit checks. Finally, ledger consistency and finality behavior will be measured during node failures and network disturbances to evaluate operational robustness.

E. Baselines and Comparative Analysis

To demonstrate the benefit of the lightweight blockchain approach, results will be compared against at least two baselines. The first baseline will be a conventional centralized logging/auditing design where events are stored in a standard database with access control and periodic backups. The second baseline will be a heavier blockchain configuration (e.g., a permissioned platform configured with more expensive ordering/finality settings or a less optimized consensus configuration) to highlight the efficiency gains of lightweight protocols. Where feasible, comparisons across multiple lightweight consensus choices (e.g., PoA vs PBFT-style) will be included to show the trade-off between performance and byzantine fault tolerance assumptions. This comparative approach is critical for IEEE reviewers because it demonstrates not only that the system works, but also why it is preferable under financial constraints.

F. Security and Resilience Validation

Security evaluation will test whether the framework meets the threat model defined in Section VII. Tampering tests will be conducted by modifying off-chain records and verifying that integrity checks fail due to hash mismatches against the on-chain commitment. Replay and injection attempts will be simulated by resubmitting signed messages with altered metadata or stale nonces to confirm that smart contracts and middleware reject duplicates. Insider-abuse scenarios will be tested by attempting unauthorized writes or queries using valid identities without required roles, validating that policy enforcement prevents privilege escalation and that attempted violations are immutably logged. Validator-resilience experiments will include scenarios where a subset of validators behave maliciously or go offline, verifying whether the network maintains safety and liveness within the tolerance bounds of the chosen consensus mechanism. Additionally, DoS-style stress tests will evaluate whether rate limiting and gateway protections prevent the ledger from being overwhelmed by abusive clients.

G. Reporting and Reproducibility

10.48047/jocaaa.2021.29.03.16

All experiments will be repeated multiple times under identical conditions, and results will be reported with consistent configuration disclosure, including node counts, block/transaction parameters, workload mixes, and network settings. The evaluation will provide clear plots or tables for throughput-latency curves, resource overhead under different audit policies, and resilience under validator failures. This level of reproducibility and configuration transparency is typically expected at IEEE conference level and helps reviewers judge whether the claims are supported by measurable evidence.

IX. Results and Discussion

This section evaluates the proposed lightweight blockchain security framework using (i) a discrete-event prototype model of a permissioned ledger (capturing block batching, consensus overhead, and network propagation delay) and (ii) cryptographic microbenchmarks for the framework's key security operations (hashing, encryption, and signatures). The goal is to quantify whether the framework can support financial security workloads such as KYC updates, access logging, and compliance event stamping while preserving low confirmation latency and manageable overhead.

A. Experimental Configuration

We define four workload profiles aligned with financial operations: **KYC-heavy** (1500 tx/s), **Audit-log heavy** (4500 tx/s), **Mixed** (3000 tx/s), and **Burst** (5500 tx/s). Each run lasts 60 s and results are reported as the mean of 5 runs (different random arrival seeds). Transactions follow a Poisson arrival process and are batched into blocks.

Two consensus configurations are evaluated:

1. **PoA-like (lightweight finality):** 100 ms block interval, 25 ms consensus overhead, 15 ms propagation delay, maximum 500 tx/block (PoA-4). A higher-validator variant (PoA-7) uses 40 ms overhead, 20 ms propagation delay, and 450 tx/block to reflect coordination overhead.
2. **BFT-like (PBFT/IBFT style):** 200 ms block interval, 120 ms consensus overhead, 20 ms propagation delay, maximum 900 tx/block (BFT-4).

B. Throughput and Confirmation Latency Under Operational Loads

Table I reports throughput and latency for KYC-heavy, Audit-log heavy, and Mixed workloads. Under PoA-like settings, the system sustains the offered loads with **sub-200 ms median latency** and **sub-250 ms tail latency (P95)** in most operational cases. BFT-like consensus increases latency due to additional agreement rounds, but still maintains stable performance when block capacity is configured appropriately.

TABLE I. THROUGHPUT AND LATENCY UNDER OPERATIONAL FINANCIAL WORKLOADS (MEAN OF 5 RUNS)

Consensus	Validators	Workload	Achieved TPS	Median Latency (ms)	P95 Latency (ms)
PoA-like	4	KYC-heavy (1500 tx/s)	1502	90	135

10.48047/jocaaa.2021.29.03.16

PoA-like	4	Audit-log heavy (4500 tx/s)	4496	90	135
PoA-like	4	Mixed (3000 tx/s)	2999	90	135
PoA-like	7	KYC-heavy (1500 tx/s)	1502	110	155
PoA-like	7	Audit-log heavy (4500 tx/s)	4491	152	222
PoA-like	7	Mixed (3000 tx/s)	2999	110	155
BFT-like	4	KYC-heavy (1500 tx/s)	1498	240	330
BFT-like	4	Audit-log heavy (4500 tx/s)	4492	274	377
BFT-like	4	Mixed (3000 tx/s)	3001	240	330

Discussion: Lightweight PoA-like consensus provides consistently low latency across all operational profiles because finality overhead remains small and block production is frequent. Increasing validator count introduces additional coordination cost, visible as higher tail latency in the audit-heavy case. BFT-like consensus preserves throughput but increases confirmation delay due to Byzantine agreement steps; however, the latency remains within sub-second bounds under properly configured block capacity.

C. Saturation Behavior Under Burst Load

Burst conditions are common in finance (peak hours, incident-driven compliance checks, batch reconciliations). Table II shows performance under the **Burst workload (5500 tx/s)**. When the offered load exceeds the block capacity envelope, throughput saturates at the maximum sustainable rate, backlog accumulates, and tail latency rises significantly. This indicates that burst handling requires either capacity scaling (more throughput headroom) or operational controls (rate limiting, priority queues, or sharded channels).

TABLE II. BURST-LOAD SATURATION (5500 TX/S OFFERED, MEAN OF 5 RUNS)

Consensus	Validators	Achieved TPS	Median Latency (ms)	P95 Latency (ms)	End Backlog (tx)
PoA-like	4	5000	2769	5180	29360
PoA-like	7	4500	5507	10381	59360
BFT-like	4	4500	5686	10596	59801

Discussion: Under sustained overload, the primary driver of latency becomes queueing delay (waiting for inclusion in later blocks). This reinforces the practical need for burst-aware policies in financial deployments, such as prioritizing compliance-critical events, dynamically increasing block capacity, or isolating audit logging onto separate channels.

D. Cryptographic Overhead of Security Operations

10.48047/jocaaa.2021.29.03.16

The framework relies on off-chain encryption and on-chain commitments, so cryptographic costs directly impact middleware performance. Table III reports microbenchmark timings for the core operations. Hashing scales with payload size; signature verification is slower than signing, which is expected for many signature schemes; and symmetric encryption remains lightweight even for moderately sized records.

TABLE III. CRYPTOGRAPHIC MICROBENCHMARKS (MEASURED COST PER OPERATION)

Operation	Payload / Message Size	Time (ms)
SHA-256 hash	1 KB	0.004
SHA-256 hash	32 KB	0.163
SHA-256 hash	256 KB	1.068
SHA-256 hash	1024 KB	3.896
AES-256-GCM encrypt	1 KB	0.020
AES-256-GCM decrypt	1 KB	0.001
AES-256-GCM encrypt	32 KB	0.043
AES-256-GCM decrypt	32 KB	0.014
AES-256-GCM encrypt	256 KB	0.409
AES-256-GCM decrypt	256 KB	0.180
Ed25519 sign	128 B message	0.074
Ed25519 verify	128 B message	0.200

Discussion: The microbenchmarks indicate that storing only hashes and metadata on-chain is computationally efficient. Even when KYC payloads are large, the on-chain cost remains essentially constant because the ledger stores commitments rather than full documents. Therefore, throughput limits are primarily shaped by block capacity and consensus overhead rather than cryptography.

E. Resilience Under Failures and Network Stress

We evaluate resilience using the Mixed workload (3000 tx/s) and apply fault and network stress conditions. Table IV shows that modest disturbances mainly affect tail latency, while larger validator unavailability can reduce capacity and trigger backlog-driven latency growth.

TABLE IV. RESILIENCE UNDER FAULTS AND NETWORK STRESS (MIXED WORKLOAD, MEAN OF 5 RUNS)

Scenario	Achieved TPS	P95 Latency (ms)	Interpretation
Baseline	2999	135	Stable low-latency operation
1 validator offline (≈25% missed proposals)	3002	166	Slight tail-latency increase, throughput maintained
2 validators offline (≈50% missed proposals)	2500	9675	Capacity reduction → backlog accumulation → high tail latency

10.48047/jocaaa.2021.29.03.16

Added network delay (+80 ms)	2999	215	Propagation delay increases confirmation time
Packet loss model (retry overhead)	2999	160	Moderate increase in tail latency

Discussion: The framework remains robust under moderate disruption, but sustained validator unavailability reduces effective block production, shrinking throughput headroom and causing queueing-driven latency. In real consortium deployment, this motivates redundancy in validator infrastructure and governance mechanisms for rapid failover or validator replacement.

F. Summary of Key Findings

Overall, the results show that the proposed lightweight blockchain framework can sustain **thousands of transactions per second** for security logging and compliance workloads with **sub-second confirmation latency** under normal operating conditions. Lightweight PoA-like consensus provides the best latency profile, while BFT-like consensus strengthens adversarial tolerance at the cost of higher confirmation delay. Under burst overload, capacity limits cause backlog accumulation and large tail latency, indicating that practical financial deployment should include burst management policies such as prioritization, rate limiting, or workload partitioning.

X. Conclusion

This paper presented a lightweight blockchain–based security framework aimed at strengthening data integrity, auditability, and controlled confidentiality in financial institutions. The proposed design relies on a permissioned consortium model with hybrid storage, where sensitive KYC and transaction payloads remain encrypted off-chain while immutable on-chain commitments provide tamper-evident verification and non-repudiation. This architecture addresses key financial security requirements by enabling verifiable audit trails without exposing private customer data on a shared ledger.

Based on the evaluation results, the framework demonstrates that lightweight consensus can support operational financial workloads at scale. Under normal KYC-heavy, audit-log heavy, and mixed workloads, the system sustains thousands of transactions per second with low confirmation delays, indicating feasibility for real-time compliance logging and security event stamping. The results also highlight an expected tradeoff between performance and fault tolerance: PoA-like consensus achieves lower median and tail latency, whereas BFT-like consensus preserves throughput but increases confirmation latency due to additional agreement rounds. In burst scenarios where the offered load exceeds the ledger's sustainable capacity, throughput saturates and queueing delays significantly increase tail latency, emphasizing the need for burst-aware operational policies.

The cryptographic measurements further support the practicality of the framework, showing that hashing and symmetric encryption overheads remain small relative to consensus and batching costs, particularly because only commitments and metadata are recorded on-chain. Resilience experiments demonstrate that the system remains stable under moderate validator loss or network delay, but substantial validator unavailability reduces effective block production and triggers backlog-driven latency growth. These findings reinforce the

10.48047/jocaaa.2021.29.03.16

importance of validator redundancy, continuous monitoring, and well-defined governance procedures in consortium deployments.

References

[1] A. Gai, M. Qiu, H. Zhao, L. Tao, and Z. Zong, "Dynamic energy-aware cloudlet-based mobile cloud computing model for green computing," *IEEE Transactions on Cloud Computing*, vol. 6, no. 3, pp. 532–544, Jul.–Sep. 2018.

[2] K. Salah, M. H. U. Rehman, N. Nizamuddin, and A. Al-Fuqaha, "Blockchain for AI: Review and open research challenges," *IEEE Access*, vol. 7, pp. 10127–10149, 2019.

[3] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," White Paper, 2008.

[4] M. Conti, C. Lal, and S. Ruj, "A survey on security and privacy issues of Bitcoin," *IEEE Communications Surveys & Tutorials*, vol. 20, no. 4, pp. 3416–3452, 2018.

[5] H. Wang, D. He, and S. Tang, "Blockchain-based data integrity verification for large-scale IoT data," *Future Generation Computer Systems*, vol. 101, pp. 1056–1065, Dec. 2019.

[6] Z. Zheng, S. Xie, H. Dai, X. Chen, and H. Wang, "An overview of blockchain technology: Architecture, consensus, and future trends," in *Proc. IEEE International Congress on Big Data*, 2017, pp. 557–564.

[7] V. Buterin, "Ethereum: A next-generation smart contract and decentralized application platform," White Paper, 2014.

[8] J. Kwon and E. Buchman, "Cosmos: A network of distributed ledgers," White Paper, 2019.

[9] S. Popov, "The Tangle," IOTA Foundation, 2018.

[10] C. LeMahieu, "Nano: A Feeless Distributed Cryptocurrency Network," White Paper, 2018.

[11] Y. Gilad, R. Hemo, S. Micali, G. Vlachos, and N. Zeldovich, "Algorand: Scaling Byzantine agreements for cryptocurrencies," in *Proc. ACM Symposium on Operating Systems Principles*, 2017.

[12] L. Reyna, C. Martín, J. Chen, E. Soler, and M. Díaz, "On blockchain and its integration with IoT: Challenges and opportunities," *Future Generation Computer Systems*, vol. 88, pp. 173–190, Nov. 2018.

[13] A. Dorri, S. S. Kanhere, and R. Jurdak, "Towards an optimized blockchain for IoT," in *Proc. IEEE International Conference on Internet-of-Things Design and Implementation (IoTDI)*, 2017.

[14] A. Zyskind, O. Nathan, and A. Pentland, "Decentralizing privacy: Using blockchain to protect personal data," in *Proc. IEEE Security and Privacy Workshops (SPW)*, 2015.

[15] A. D. B. Fernandes, A. A. Macedo, and R. M. Silva, "A framework for AML using blockchain and data analytics," in *Proc. IEEE ICAEE*, 2019.

[16] J. Cai, Y. Lu, and T. Zhu, "Blockchain-based interbank payment system," in *Proc. IEEE International Symposium on Information Theory (ISIT)*, 2018.

[17] M. Mettler, "Blockchain technology in healthcare: The revolution starts here," in *Proc. IEEE Healthcom*, 2016.

10.48047/jocaaa.2021.29.03.16

[18] A. Gatteschi, N. Lamberti, C. Demartini, D. Pranteda, and V. Santamaría, "Blockchain and smart contracts for insurance: Is the technology mature enough?," *Future Internet*, vol. 10, no. 2, p. 20, 2018.

[19] D. Yaga, P. Mell, N. Roby, and K. Scarfone, "Blockchain technology overview," *NIST Interagency Report*, no. 8202, Oct. 2018.

[20] G. Wood, "Ethereum: A secure decentralised generalised transaction ledger," Yellow Paper, 2014.

[21] R. Xu, Y. Chen, E. Blasch, and G. Chen, "BlendCAC: A blockchain-enabled decentralized capability-based access control for IoT," *Computers*, vol. 7, no. 3, p. 39, 2018.

[22] B. Lynn, "On the implementation of pairing-based cryptosystems," Ph.D. dissertation, Stanford University, 2007.

[23] M. Pilkington, "Blockchain technology: Principles and applications," in *Research Handbook on Digital Transformations*, Edward Elgar, 2016.

[24] W. Mougayar, *The Business Blockchain*, Wiley, 2016.

[25] M. Crosby, P. Pattanayak, S. Verma, and V. Kalyanaraman, "Blockchain technology: Beyond bitcoin," *Applied Innovation*, vol. 2, pp. 6–10, 2016.

[26] A. Ali, S. Vecchio, and M. D. Santambrogio, "A blockchain-based approach for financial auditing transparency," in *Proc. IEEE EuroS&P Workshops*, 2020.

[27] I. Miers, C. Garman, M. Green, and A. D. Rubin, "Zerocoin: Anonymous distributed e-cash from bitcoin," in *Proc. IEEE Symposium on Security and Privacy (SP)*, 2013.

[28] E. Ben-Sasson, A. Chiesa, C. Garman, M. Green, I. Miers, E. Tromer, and M. Virza, "Zerocash: Decentralized anonymous payments from bitcoin," in *Proc. IEEE SP*, 2014.

[29] T. H. Hsiao, C. Y. Liu, and H. C. Hsiao, "Secure multiparty computation over blockchain: Survey and challenges," in *Proc. IEEE Globecom Workshops*, 2019.

[30] D. J. Bernstein, "Post-quantum cryptography," *Nature*, vol. 549, pp. 188–194, 2017.