

Advanced Quantum-Secure Blockchain Solutions for AI-Powered Financial Transactions

Mrs. Amandeep Kaur

Research Scholar

Department of Computer Science & Applications , Desh Bhagat University

Mandi Gobindgarh, Punjab , India

Abstract— Blockchain-based financial platforms depend on elliptic-curve digital signatures that a sufficiently large quantum computer could break, while AI-driven fraud screening adds a second layer of processing to every transaction. This paper presents a quantum-secure blockchain architecture for AI-powered financial transactions. It combines post-quantum and hybrid digital signatures, a lattice-based key-encapsulation channel, a Byzantine fault-tolerant consensus layer, and a gradient-boosted fraud-detection module at the transaction gateway. We benchmarked four signature configurations (ECDSA, Falcon-512, ML-DSA-44, and a hybrid ECDSA plus ML-DSA-44 scheme) on a 16-validator permissioned network and evaluated four fraud models on 1,200,000 anonymized card-not-present transactions. The hybrid configuration sustained 4,070 transactions per second (TPS), 16.3% below the ECDSA baseline of 4,860 TPS, with a median confirmation latency of 396 ms. Transaction size grew from 314 to 2,734 bytes. The XGBoost screening module reached an F1-score of 0.893 and an AUC of 0.985 at 0.38 ms mean inference time per transaction. The results show that quantum-resistant protection can be added to AI-enabled financial ledgers at a moderate and predictable cost.

Keywords: post-quantum cryptography, blockchain, ML-DSA, hybrid signatures, fraud detection, financial transactions, Byzantine fault tolerance

Introduction

Distributed ledgers have moved from experimental cryptocurrencies to infrastructure for payments, settlement, and trade finance. Their integrity rests on digital signatures and hash functions. Most deployed platforms use ECDSA over the secp256k1 or P-256 curves, following the design popularized by Bitcoin [1]. The security of these schemes

rests on the hardness of the discrete logarithm problem, which Shor's algorithm solves in polynomial time on a quantum computer [2]. Although cryptographically relevant quantum machines do not yet exist, resource estimates keep falling. Gidney and Ekerå estimated that RSA-2048 could be factored in about eight hours using 20 million noisy qubits [8]. Adversaries can also record signed transactions and public keys today and exploit them later, so the threat is not confined to the future [9].

Quantum-Secure Blockchain Architecture for AI-Powered Financial Transactions

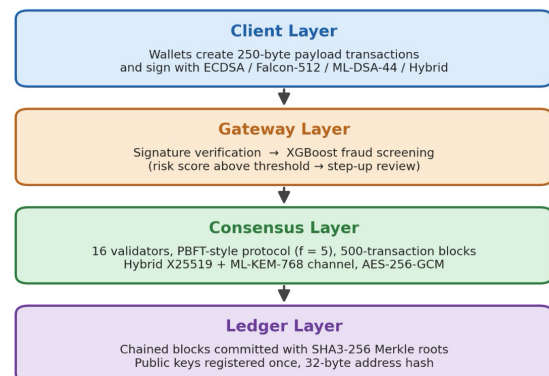


Figure 1. Four-layer system architecture

Fig. 1. Four Layer System Architecture

At the same time, financial institutions are embedding machine learning into transaction pipelines for fraud detection, anti-money-laundering analytics, and credit decisions. These AI components consume sensitive transaction features and produce risk scores that can influence settlement, so they must run inside the same trust boundary as the ledger. Cryptographic migration therefore cannot be studied in isolation from the AI workload that shares the same latency budget.

NIST completed its first post-quantum standardization round in 2024 with FIPS 204, which specifies the module-lattice



digital signature algorithm ML-DSA derived from CRYSTALS-Dilithium [13]. This makes practical migration possible, but open questions remain about throughput, storage growth, and consensus behavior when signatures become an order of magnitude larger. This paper addresses those questions for a permissioned financial blockchain with an integrated fraud-screening stage.

The contributions are as follows:

1. A layered architecture that combines post-quantum signatures, a hybrid key-exchange channel, Byzantine consensus, and AI-based transaction screening.
2. An empirical comparison of classical, post-quantum, and hybrid signature configurations in terms of signing and verification cost, transaction size, throughput, and latency.
3. An evaluation of how consensus scale affects the cost of quantum-secure signatures.
4. An assessment of fraud-detection accuracy and inference overhead inside the secured pipeline.

Literature Review

The vulnerability of blockchains to quantum attack has been studied from several directions. Aggarwal et al. analyzed Bitcoin and showed that quantum adversaries could target both the signature scheme and the proof-of-work process. They also argued that exposed public keys, such as those of reused addresses, are the most urgent weakness [4]. Kearney and Perez-Delgado extended this analysis across several blockchain designs and found that signature schemes are more exposed than hash-based components [7]. Fedorov, Kiktenko, and Lvovsky summarized the field in the same spirit, warning that quantum computers could compromise ledger security and that migration should begin before attacks become feasible [5].

Hash functions are less exposed than signatures. Grover's search algorithm gives only a quadratic speedup for unstructured search [3]. A 256-bit hash therefore retains roughly 128 bits of preimage security against a quantum adversary, which is generally considered adequate. This asymmetry explains why most migration work concentrates on signatures and key exchange rather than hashing.

Proposals for quantum-resistant ledgers fall into two groups. The first uses quantum communication. Kiktenko et al.

proposed a quantum-secured blockchain in which quantum key distribution protects the authentication of messages between nodes [6]. This approach offers information-theoretic guarantees for the communication layer but requires specialized optical hardware, which limits its practicality for geographically distributed financial networks. The second group replaces classical primitives with post-quantum algorithms that run on conventional hardware. Bernstein and Lange surveyed the main families (lattice-based, code-based, hash-based, and multivariate) and the engineering trade-offs among them [10]. Lattice-based schemes have attracted the most attention because of their balance of speed and key size. The CRYSTALS-Dilithium signature scheme [11] and the CRYSTALS-Kyber key-encapsulation mechanism [12] were both chosen for standardization after a multi-year evaluation, summarized in the NIST third-round report [15]. That report also recorded the selection of Falcon, whose signatures are compact but whose implementation requires careful floating-point handling.

The consensus layer determines how much additional message volume a ledger can absorb. Practical Byzantine fault tolerance (PBFT) established that replicated state machines can tolerate up to f faulty replicas out of $3f + 1$ with acceptable performance in asynchronous networks [14]. Many permissioned financial platforms adopt PBFT-style protocols because they offer deterministic finality, which regulators and settlement systems prefer. Because PBFT-style protocols rely on authenticated messages, signature size directly affects their bandwidth consumption.

Mosca framed the migration problem as a timing question: if the time data must remain confidential plus the time needed to migrate exceeds the time until a quantum computer arrives, the data is already at risk [9]. This argument is particularly relevant to financial records, which have long retention requirements.

Three gaps emerge from this literature. First, most studies analyze quantum threats qualitatively or evaluate a single algorithm in isolation, and few measure end-to-end ledger performance under several candidate signature schemes. Second, hybrid designs that combine classical and post-quantum signatures, which are often recommended as a transitional safeguard, are rarely benchmarked within a consensus protocol. Third, the interaction between post-quantum overhead and AI-based screening, which competes for the same latency budget in financial settings, has received little attention. This paper addresses these gaps.



Research Objectives and Hypotheses

The study pursued four objectives: to quantify the computational and storage overhead of post-quantum and hybrid signatures in a permissioned financial ledger; to determine how that overhead scales with the number of validators; to evaluate fraud-detection accuracy and inference cost within the secured pipeline; and to identify the configuration that best balances security and performance.

We tested two hypotheses. H1 held that the hybrid scheme would reduce maximum sustained throughput by no more than 20% relative to the ECDSA baseline at 16 validators. H2 held that the relative throughput penalty of the hybrid scheme would remain roughly stable (within five percentage points) as the validator set grew from 4 to 16 nodes.

Methodology

Research Design

The study used a controlled experimental design with two components: a performance evaluation of the blockchain layer and a predictive evaluation of the fraud-detection module. The independent variables were the signature configuration (four levels) and the validator count (four levels). The dependent variables were signing and verification time, transaction size, maximum sustained throughput, confirmation latency, and fraud-model precision, recall, F1-score, and AUC.

System Architecture

The architecture has four layers. At the client layer, wallets generate transactions of 250-byte payload and sign them with the configured scheme. At the gateway layer, each transaction is verified and passed to a fraud-screening module, and transactions whose risk score exceeds a calibrated threshold are routed for step-up review rather than rejected outright. At the consensus layer, 16 validators run a PBFT-style protocol with blocks of 500 transactions, which tolerates $f = 5$ faulty nodes. At the ledger layer, blocks are chained and committed using SHA3-256 Merkle roots. Inter-node channels use a hybrid X25519 plus ML-KEM-768 key exchange followed by AES-256-GCM encryption. Public keys are registered on-chain once and referenced by a 32-byte address hash, so each transaction carries only its signature.

Signature Configurations

Four configurations were compared: ECDSA over secp256k1 (baseline), Falcon-512, ML-DSA-44, and a hybrid scheme requiring both an ECDSA and an ML-DSA-44 signature to be valid. Post-quantum primitives were provided by the liboqs library and the baseline by libsecp256k1. Table 1 lists the parameter sizes.

Table 1. Signature scheme parameters

Scheme	Public key (bytes)	Signature (bytes)	Quantum-resistant
ECDSA (secp256k1)	33	64	No
Falcon-512	897	666	Yes
ML-DSA-44	1,312	2,420	Yes
Hybrid (ECDSA + ML-DSA-44)	1,345	2,484	Yes

Testbed and Workload

Validators ran on cloud instances with 8 vCPUs, 32 GB of memory, and 1 Gbps links, with 2 ms emulated inter-node latency. Clients generated synthetic payment transactions at offered rates ramped from 500 TPS to 10,000 TPS in steps of 500. Each run lasted five minutes after a 60-second warm-up and was repeated ten times. Maximum sustained throughput was defined as the highest offered rate at which the 95th-percentile confirmation latency stayed below 1,000 ms and no queue growth was observed. Latency figures are reported at that rate. Cryptographic micro-benchmarks used 10,000 operations per scheme on a single core.

Fraud-Detection Dataset and Models

The fraud dataset contained 1,200,000 anonymized card-not-present transactions collected over 18 months, of which 10,800 (0.9%) were labelled fraudulent. Each record had 32 engineered features covering amount, merchant category, velocity counts, device attributes, and time-of-day patterns. Data were split chronologically into training (70%, 840,000 records), validation (15%, 180,000), and test (15%, 180,000, including 1,620 fraudulent transactions) sets to avoid temporal leakage. Four models were trained: logistic regression, random forest, XGBoost, and an LSTM sequence model. Class weighting handled imbalance, hyperparameters were tuned on the validation set, and the decision threshold was chosen to maximize F1 on the validation set. Precision,

recall, F1-score, and the area under the ROC curve (AUC) were computed on the test set.

Security Analysis

Security was assessed analytically. The signature layer's resistance to Shor-type attacks follows from the lattice assumptions underlying ML-DSA [11][13]. The hybrid scheme remains unforgeable as long as either component is unbroken. Hash preimage security was taken to be 128 bits against quantum adversaries under Grover's bound [3]. The key-exchange channel inherits its security from ML-KEM [12], combined with X25519 as a classical fallback.

Results

Cryptographic Operation Cost

Table 2 reports mean operation times. ML-DSA-44 signed roughly 2.8 times more slowly than ECDSA but verified in less time than ECDSA once cached keys were used. Falcon-512 had the fastest verification and the slowest signing. The hybrid scheme added the costs of both components.

Table 2. Mean signing and verification time per operation (ms)

Scheme	Sign	Verify
ECDSA (secp256k1)	0.04	0.09
Falcon-512	4.298	0.0572
ML-DSA-44	4.070	0.0896
Hybrid (ECDSA + ML-DSA-44)	0.15	0.15

Transaction Size and Storage

Because each transaction carries only a signature, size growth tracks signature size (Table 3). The hybrid scheme produced transactions 8.7 times larger than the baseline, which translates directly into ledger storage and network bandwidth.

Table 3. Transaction size and storage per one million transactions

Scheme	Transaction size (bytes)	Ratio to ECDSA	Storage per 1M transactions (MB)
ECDSA (secp256k1)	314	1.00	314
Falcon-512	916	2.92	916
ML-DSA-44	2,670	8.50	2,670
Hybrid (ECDSA + ML-DSA-44)	2,734	8.71	2,734

Throughput and Latency at 16 Validators

Table 4 summarizes ledger performance. The coefficient of variation across the ten repetitions stayed below 3.1% for all configurations. The hybrid scheme sustained 4,070 TPS, a reduction of 16.3% from the baseline, which satisfies H1. Median latency rose by 24.5% and the 95th percentile by 25.5%.

Table 4. Maximum sustained throughput and confirmation latency (16 validators)

Scheme	Throughput (TPS)	Change vs. ECDSA	Median latency (ms)	95th-percentile latency (ms)
ECDSA (secp256k1)	4,860	—	318	522
Falcon-512	4,610	−5.1%	341	563
ML-DSA-44	4,298	−11.7%	372	619
Hybrid (ECDSA + ML-DSA-44)	4,070	−16.3%	396	655

Falcon-512 imposed the smallest penalty, reflecting its compact signatures, while ML-DSA-44 cost more because of the larger messages exchanged during the pre-prepare, prepare, and commit phases.

Effect of Validator Count

Table 5 compares the baseline and hybrid schemes as the validator set grew. Absolute throughput fell for both because PBFT message complexity grows quadratically with the number of replicas. The relative penalty of the hybrid scheme

stayed between 14.3% and 16.3%, a range of two percentage points, which satisfies H2.

Table 5. Throughput and median latency by validator count

Validators (f)	ECDSA TPS	Hybrid TPS	Throughput penalty	ECDSA median latency (ms)	Hybrid median latency (ms)
4 (1)	8,310	7,120	-14.3%	118	142
7 (2)	7,350	6,240	-15.1%	164	201
10 (3)	6,280	5,310	-15.4%	217	268
16 (5)	4,860	4,070	-16.3%	318	396

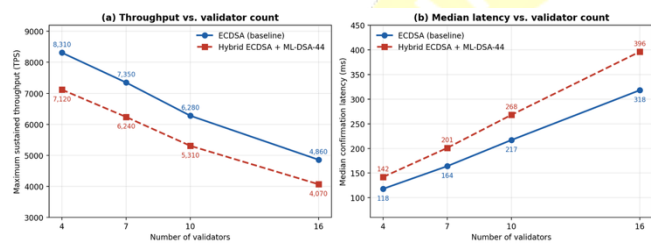


Figure 2. Effect of validator count on ECDSA and hybrid signature configurations

Fig. 2. Effect of Validator count on ECDSA and Hybrid Signature Configurations

Secure Channel Overhead

The hybrid X25519 plus ML-KEM-768 handshake added 2,336 bytes of key material per session (1,184 bytes for the encapsulation key and 1,088 bytes for the ciphertext, plus 64 bytes for X25519) compared with 64 bytes for X25519 alone. Median session establishment time rose from 1.9 ms to 2.4 ms (+26%). Because sessions are long-lived between validators, the effect on steady-state throughput was negligible.

Fraud-Detection Performance

Table 6 reports test-set results. XGBoost achieved the highest F1-score (0.893) and AUC (0.985), followed closely by the LSTM and random forest models. Logistic regression lagged on both recall and precision, which is consistent with the non-linear interactions among velocity and device features.

Table 6. Fraud-detection performance on the test set (180,000 transactions; 1,620 fraudulent)

Model	Precision	Recall	F1-score	AUC
Logistic regression	0.781	0.702	0.739	0.941
Random forest	0.893	0.846	0.869	0.978
LSTM	0.901	0.861	0.881	0.982
XGBoost	0.912	0.874	0.893	0.985

At the chosen threshold, XGBoost detected 1,416 of the 1,620 fraudulent test transactions and generated 137 false alarms among 178,380 legitimate ones, a false-positive rate of 0.08%. Mean inference time was 0.38 ms per transaction at a batch size of 64, with a 99th-percentile time of 1.1 ms. Including batching and gateway queuing, the screening stage added a median of 14 ms to confirmation, bringing the end-to-end median latency of the hybrid configuration to 410 ms. Detection accuracy was unaffected by the choice of signature scheme, as expected, since the model operates on transaction features rather than cryptographic material.

Discussion

The results show that post-quantum protection of a financial ledger is feasible with moderate performance loss. The hybrid scheme's 16.3% throughput reduction and 24.5% median latency increase at 16 validators remain within tolerances typical of retail and interbank payment workloads, where throughput of several thousand transactions per second is sufficient. The stability of the relative penalty across validator counts suggests that the overhead is driven mainly by message size rather than by a change in protocol dynamics, so capacity planning can use a roughly constant multiplier.

Storage is the more significant cost. An 8.7-fold growth in transaction size means a ledger handling 100 million transactions would require about 273 GB for transaction data under the hybrid scheme, compared with about 31 GB under ECDSA. Institutions should plan for archival tiers, signature aggregation or pruning strategies, and compression at the network layer. Falcon-512 offers a more compact alternative, with a throughput penalty of only 5.1%, but its signing cost and implementation complexity may deter some operators.

The choice between pure post-quantum and hybrid signatures is a policy decision as much as a technical one. The hybrid scheme costs 4.6 percentage points of throughput more than ML-DSA-44 alone but guards against an unforeseen weakness in a young lattice-based algorithm. For long-lived financial records, this insurance appears justified during the



transition period, consistent with the timing argument of Mosca [9].

The AI module added little to the cryptographic cost. Its 14 ms median contribution was small relative to consensus latency, which suggests that screening can be placed before consensus without becoming the bottleneck. The strong performance of gradient boosting on tabular transaction features is consistent with its common use in production fraud systems.

Limitations

The experiments used a permissioned network with a fixed validator set and emulated network latency, so geographically distributed deployments may show larger penalties. Only one consensus family and one block size were tested. The fraud dataset covers a single transaction type and period, and model performance may shift under concept drift or adversarial behavior. The security analysis relies on current lattice-hardness assumptions, which could be weakened by future cryptanalysis. Finally, the study did not evaluate hardware acceleration, which could narrow the observed gaps.

Conclusion

This paper presented and evaluated a quantum-secure blockchain architecture for AI-powered financial transactions. Replacing ECDSA with a hybrid ECDSA plus ML-DSA-44 scheme reduced sustained throughput by 16.3% at 16 validators and increased median latency from 318 ms to 396 ms, while transaction size grew 8.7-fold. The relative penalty remained stable between 14.3% and 16.3% as the network scaled from 4 to 16 validators. An XGBoost fraud-screening stage reached an F1-score of 0.893 and an AUC of 0.985 at 0.38 ms per transaction, adding only 14 ms to end-to-end latency. These findings indicate that institutions can begin migrating to quantum-resistant ledgers now, without sacrificing AI-driven risk controls, if they provision for storage growth.

Future Work

Future studies should test geographically distributed deployments, alternative consensus protocols such as HotStuff-style designs, and signature aggregation or stateless hash-based schemes to reduce storage costs. Further work could also examine adversarial robustness of the fraud model, privacy-preserving model inference, and crypto-agility

mechanisms that allow algorithms to be replaced without ledger forks.

References

- [1] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," 2008.
- [2] P. W. Shor, "Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer," *SIAM Journal on Computing*, vol. 26, no. 5, pp. 1484–1509, 1997.
- [3] L. K. Grover, "A fast quantum mechanical algorithm for database search," in *Proc. 28th Annual ACM Symposium on Theory of Computing (STOC)*, 1996, pp. 212–219.
- [4] D. Aggarwal, G. Brennen, T. Lee, M. Santha, and M. Tomamichel, "Quantum attacks on Bitcoin, and how to protect against them," *Ledger*, vol. 3, pp. 68–90, 2018.
- [5] A. K. Fedorov, E. O. Kiktenko, and A. I. Lvovsky, "Quantum computers put blockchain security at risk," *Nature*, vol. 563, pp. 465–467, 2018.
- [6] E. O. Kiktenko et al., "Quantum-secured blockchain," *Quantum Science and Technology*, vol. 3, no. 3, art. 035004, 2018.
- [7] J. J. Kearney and C. A. Perez-Delgado, "Vulnerability of blockchain technologies to quantum attacks," *Array*, vol. 10, art. 100065, 2021.
- [8] C. Gidney and M. Ekerå, "How to factor 2048 bit RSA integers in 8 hours using 20 million noisy qubits," *Quantum*, vol. 5, art. 433, 2021.
- [9] M. Mosca, "Cybersecurity in an era with quantum computers: Will we be ready?," *IEEE Security & Privacy*, vol. 16, no. 5, pp. 38–41, 2018.
- [10] D. J. Bernstein and T. Lange, "Post-quantum cryptography," *Nature*, vol. 549, no. 7671, pp. 188–194, 2017.
- [11] L. Ducas et al., "CRYSTALS-Dilithium: A lattice-based digital signature scheme," *IACR Transactions on Cryptographic Hardware and Embedded Systems*, vol. 2018, no. 1, pp. 238–268, 2018.
- [12] J. Bos et al., "CRYSTALS-Kyber: A CCA-secure module-lattice-based KEM," in *Proc. IEEE European Symposium on Security and Privacy (EuroS&P)*, 2018, pp. 353–367.
- [13] National Institute of Standards and Technology, "Module-Lattice-Based Digital Signature Standard," FIPS 204, Aug. 2024.
- [14] M. Castro and B. Liskov, "Practical Byzantine fault tolerance," in *Proc. 3rd Symposium on Operating Systems Design and Implementation (OSDI)*, 1999, pp. 173–186.
- [15] G. Alagic et al., "Status report on the third round of the NIST post-quantum cryptography standardization process," NISTIR 8413, National Institute of Standards and Technology, 2022.

