Certificate Validation Using Blockchain

Mayuri Nagare[1], Digvijay Dakhore[2], Aadi Mahalle[3], Mrs. Soudamini Somvanshi[4], Mrs. Supriya Sathe[5]
[1,2,3,4,5]*dept of Computer Engineering, D.Y. Patil College of Engineering, Pune India*

***Abstract*—The project proposes a decentralized blockchain framework that ensures the integrity, authenticity, and efficiency of digital certificates. Leveraging the Ethereum blockchain and smart contracts, the system facilitates tamper-proof storage of certificate hashes and incorporates a consortium-based validation mechanism. A novel aspect is the implementation of a hash-mapped Bloom Filter which significantly reduces verification time and computational overhead by performing preliminary checks on certificate existence. The system integrates stakeholders including issuers, validators, students, and verifiers, and uses cryptographic hashing (SHA-256/Keccak-256) and RSA digital signatures to provide a functional, low-cost, and trusted solution for global certificate authentication.**

***Index Terms*—Blockchain, Certificate Validation, Ethereum, Smart Contracts, Bloom Filter, Cryptographic Hashing, RSA Digital Signature, Decentralized Systems**

I. INTRODUCTION

In an era where academic and professional credentials are increasingly targeted for forgery, traditional paper-based and centralized digital certificate verification systems reveal significant vulnerabilities. These conventional methods are prone to tampering, depend heavily on centralized authorities, and suffer from inefficiencies due to manual verification processes. Such weaknesses not only lead to financial and productivity losses but also erode trust within educational and hiring ecosystems. The persistent challenges of certificate fraud, delayed verification, and lack of transparency underscore the urgent need for a secure, decentralized, and efficient solution.

This study proposes a decentralized blockchain-based framework designed to ensure the integrity, authenticity, and efficiency of digital certificates. Leveraging Ethereum smart contracts and a consortium-based validation mechanism, the system enables tamper-proof storage of certificate hashes on a public blockchain while utilizing a Bloom Filter to optimize search performance. By cryptographically hashing each certificate, signing it with RSA, and storing only the hash on-chain with the actual document maintained off-chain the system achieves an optimal balance between transparency and storage efficiency.

The proposed system integrates multiple stakeholders' issuers, validators, students, and verifiers into a unified architecture were only trusted institutions, approved through a rigorous vote-based onboarding process, can issue certified documents. Experimental evaluation on the Ethereum Sepolia testnet confirms the system's cost-effectiveness and scalability, demonstrating significantly reduced gas fees for certificate issuance compared to existing models.

II. LITERATURE REVIEW

Several studies have explored the application of blockchain for secure certificate management. Turkanovic et al. (2018) proposed EduCTX, a blockchain-based platform for managing academic credits across institutions. While EduCTX demonstrated the feasibility of decentralized academic record-keeping, it focused primarily on credit transfer rather than granular certificate-level verification, leaving a gap in authenticating individual diplomas and degrees. Similarly, Ghani et al. (2022) implemented a permissioned blockchain using Hyperledger Fabric for student certificate management. Their system emphasized secure access control and auditability but incurred high setup costs and faced scalability limitations, making it less viable for large-scale, multi-institutional adoption.

In the domain of verification efficiency, Garba et al. (2021) introduced LightLedger, which integrated Bloom Filters with blockchain to accelerate certificate lookup times. Their approach significantly reduced query latency for invalid certificates and incorporated

Zero-Knowledge Proofs (ZKPs) for privacy preservation. However, Light Ledger relied on external browser plugins and employed random validator selection, which could compromise reliability and user accessibility. Meanwhile, Rahman et al. (2023) addressed the challenge of correcting issued certificates without violating blockchain immutability through a dual-chain architecture. While innovative in supporting controlled updates, their model was not designed for decentralized multi-institutional consensus, limiting its applicability in heterogeneous educational ecosystems.

Privacy-focused approaches have also been investigated. Pericas-Gornals et al. (2022) developed a blockchain-based management system for digital COVID-19 certificates using proxy re-encryption to enhance data privacy. Although effective for health credentials, their system was niche-specific and lacked mechanisms for universal certificate verification. Adja et al. (2021) proposed a blockchain-enabled Public Key Infrastructure (PKI) for certificate revocation, improving the management of invalidated credentials but introducing complexity in consensus integration and higher latency. Recent works have begun integrating hybrid models to balance transparency and control. Priyadarshini et al. (2025) presented an integrated blockchain system combining Bloom Filters, RSA signing, and a voting mechanism for issuer validation. Their solution achieved low transaction costs and efficient lookup, yet incurred moderate setup expenses for new institutions. Similarly, Rahardja et al. (2021) advocated for a globalized, immutable digital certificate framework using blockchain, though their model lacked a dynamic trust mechanism and validation voting, reducing its adaptability in evolving networks.

III. METHODOLOGY

The proposed system adopts a layered, decentralized architecture integrating blockchain technology, cryptographic hashing, and probabilistic data structures to achieve secure, efficient, and tamper-proof certificate validation. The methodology is structured across four core layers: Frontend, Application, Blockchain, and Storage. The following subsections detail the technical models, algorithms, and processes employed.

A. System Architecture

Frontend Layer: A React.js-based web interface provides role-specific dashboards for institutions, students, and verifiers. MetaMask integration enables decentralized authentication and transaction signing.

Application Layer: Implements business logic using Node.js and Web3.js, handles cryptographic hashing (SHA-256), RSA signing, and communicates with the blockchain and IPFS.

Blockchain Layer: Ethereum smart contracts (written in Solidity) manage certificate hashes, issuer validation via voting, and access control. The Sepolia testnet is used for deployment and testing.

Storage Layer: IPFS (InterPlanetary File System) stores actual certificate files off-chain, referenced via Content Identifiers (CIDs) linked to on-chain hashes.

B. Cryptographic Models

Hash Mapping Model: Each certificate is converted into a unique fixed-length identifier using cryptographic hashing to ensure immutability and enable fast lookup via Bloom Filters. Let c be the certificate file, H(c) its SHA-256 hash, and CH the integer representation of H(c). The index for Bloom Filter mapping is computed as: FN = CH mod TI, where TI is the total number of Bloom Filter bit-array indices.

RSA Digital Signature Model: To ensure issuer authenticity and non-repudiation, each certificate hash is signed by the issuing institution using RSA. Let M = H(c) be the certificate hash. The signature S is generated using the issuer's private key d and modulus n: S = M^d mod n. Verification uses the issuer's public key e: M' = S^e mod n. If M' = M, the signature is valid.

Bloom Filter Optimization Model: A Bloom Filter is employed to reduce unnecessary on-chain queries during verification. Given n (certificates inserted), m (size of bit array), k (number of hash functions), and p (false positive probability), the optimal parameters are: m = −(n ln p) / (ln 2)^2 and k_opt = (m/n) ln 2. The false positive rate is approximated by: p ≈ (1 − e^(−kn/m))^k.

C. Core Algorithms

Certificate Issuance Algorithm: (1) Generate H(c) = SHA-256(c). (2) Upload c to IPFS, receive CID. (3) Issue RSA signature S for H(c). (4) Invoke smart contract add Certificate(H(c), CID, S). (5) Update

Bloom Filter with H(c). Output: Transaction receipt and on-chain hash record.

Certificate Verification Algorithm: (1) Compute H(c'). (2) Query Bloom Filter if false, certificate is invalid (terminate); if true, proceed. (3) Call smart contract verify Certificate(H(c')). (4) Compare stored hash with H(c'). Output: Valid/Invalid result.

Issuer Onboarding (Voting Mechanism) Algorithm: (1) Existing validators vote via Vote () smart contract function. (2) Unanimous approval required for onboarding. (3) If approved, institute is added to trusted validator list. Output: Updated validator consortium.

Certificate Correction Algorithm: (1) Verify original certificate's existence via Bloom Filter and blockchain. (2) Mark original hash as invalid in revocation list. (3) Issue new certificate c_new, hash and store on-chain. (4) Update Bloom Filter with new hash. Output: Corrected certificate record and updated audit trail.

D. Security and Cost Models

Security Analysis: 51% Attack Resistance is achieved through Ethereum's Proof-of-Stake consensus and a validator consortium, which increases attack cost and complexity. Sybil Attack Prevention is ensured by allowing each institution only a single validated identity requiring unanimous voting. Immutability and Integrity are guaranteed by cryptographic hashing and blockchain immutability, which prevent tampering. Privacy Preservation is maintained by storing only hashes on-chain while actual certificates are encrypted and stored off-chain.

Gas Cost Model: Transaction costs on Ethereum are calculated as: Cost (USD) = (EP × TGC × GP) / 10^9, where EP is the Ether price (USD/ETH), TGC is total gas consumed, and GP is gas price in gwei.

E. Implementation and Testing Framework

Smart contracts were developed in Solidity and deployed on the Sepolia testnet. Unit tests were conducted using Mocha/Chai, static analysis was performed with Slither, and end-to-end validation was carried out. Performance metrics measured include gas consumption, verification latency, and Bloom Filter false positive rate. Tools used include Remix IDE, MetaMask, Web3.js, IPFS CLI, and React.js.

IV. RESULTS

The proposed blockchain-based certificate validation system was experimentally evaluated on the Ethereum Sepolia testnet. The results demonstrate significant improvements in verification efficiency, cost-effectiveness, and security compared to existing solutions.

A. System Performance and Efficiency

Verification Time Reduction: The integration of a Bloom Filter reduced the average certificate search time by 87% in cases where certificates were invalid or non-existent. For valid certificates, verification remained under 2 seconds, including on-chain confirmation.

Gas Cost Analysis: Smart contract transactions were optimized for gas efficiency. The cost of adding a certificate was 4.57 USD, compared to 14.32 USD (Pericas-Gornals et al., 2022) and 8.24 USD (Elva Leka et al., 2021). Adding a new institution incurred 8.97 USD, slightly higher than some systems but justified by the robust voting-based validation mechanism.

Bloom Filter Optimization: With a 1 MB bit array and optimal hash functions (k = 5), the false positive rate remained below 0.5% for up to 50,000 certificates. This ensured reliable pre-screening without significant compromise in accuracy.

B. Security Validation

Smart Contract Auditing: Static analysis using Slither detected zero high or medium-risk vulnerabilities. One low-risk issue (unused variable) was identified and resolved prior to deployment.

Attack Resistance: 51% Attack Mitigation was achieved through the consortium-based validator model and Ethereum's PoS consensus, which increased the computational and economic cost of an attack. Sybil Attack Prevention was enforced via strict single-identity registration and unanimous voting, preventing malicious multi-identity infiltration. Immutability Assurance was confirmed as all certificate hashes remained tamper-proof once recorded on-chain, with no unauthorized modifications detected during testing.

V. CONCLUSION

The proliferation of digital credentials and the persistent vulnerabilities of traditional verification systems have created a critical need for secure, transparent, and efficient solutions. This study presented a blockchain-based framework for certificate validation that effectively addresses these challenges by leveraging decentralized ledger technology, cryptographic assurance, and optimized data structures.

The proposed system successfully integrates Ethereum smart contracts, a consortium-based validator network, and a Bloom Filter optimization layer to create a tamper-proof, scalable, and cost-effective credentialing ecosystem. By storing only cryptographic hashes on-chain and utilizing IPFS for off-chain document storage, the system balances transparency with efficiency while maintaining data integrity. The inclusion of a voting mechanism for issuer validation ensures that only trusted institutions can issue certificates, significantly reducing the risk of fraud. Experimental results on the Sepolia testnet confirm the system's practical viability, with reduced transaction costs, accelerated verification times, and robust security against common blockchain attacks such as Sybil and 51% assaults.

This research demonstrates that blockchain technology can fundamentally transform how academic and professional credentials are issued, shared, and verified. The proposed framework not only eliminates forgery and centralized points of failure but also introduces a flexible correction mechanism that respects blockchain immutability.

REFERENCES

[1] R. Pericas-Gornals, M. Mut-Puigserver, and M. M. Payeras-Capella, “Highly private blockchain-based management system for digital COVID-19 certificates,” International Journal of Information Security, vol. 21, no. 5, pp. 1069–1090, Oct. 2022.

[2] M. Turkanovic, M. Holbl, K. Kosic, M. Hericko, and A. Kamisalic, “EduCTX: A blockchain-based higher education credit platform,” IEEE Access, vol. 6, pp. 5112–5127, 2018.

[3] A. Garba, Z. Chen, Z. Guan, and G. Srivastava, “LightLedger: A novel blockchain-based domain certificate authentication and validation scheme,” IEEE Transactions on Network Science and Engineering, vol. 8, no. 2, pp. 1698–1710, Apr. 2021.

[4] M. M. Rahman, M. T. K. Tonmoy, S. R. Shihab, and R. Farhana, “Blockchain-based certificate authentication system with enabling correction,” arXiv preprint arXiv:2302.03877, 2023.

[5] U. Rahardja, A. N. Hidayanto, P. O. H. Putra, and M. Hardini, “Immutable ubiquitous digital certificate authentication using blockchain protocol,” Journal of Applied Research and Technology, vol. 19, no. 4, pp. 308–321, Aug. 2021.

[6] S. Mondal, A. Panja, and S. Karforma, “An efficient e-certificate management system in e-learning using blockchain,” Scientific Culture, vol. 89, nos. 3–4, pp. 1–5, Apr. 2023.

[7] R. F. Ghani, A. A. Salman, A. B. Khudhair, and L. Aljobouri, “Blockchain-based student certificate management and system sharing using Hyperledger Fabric platform,” Periodicals of Engineering and Natural Sciences, vol. 10, no. 2, pp. 207–218, Apr. 2022.

[8] Y. C. Elloh Adja, B. Hammi, A. Serhrouchni, and S. Zeadally, “A blockchain-based certificate revocation management and status verification system,” Computers & Security, vol. 104, Art. no. 102209, May 2021.

[9] S. Zhang, Y. Cao, Z. Ning, F. Xue, D. Cao, and Y. Yang, “A heterogeneous IoT node authentication scheme based on hybrid blockchain and trust value,” KSII Transactions on Internet and Information Systems, vol. 14, no. 9, pp. 3615–3638, 2020.

[10] R. Priyadarshini, R. Pandey, K. C. Ankit, D. Bhandari, B. Khadka, R. K. Barik, and M. J. Saikia, “A faster, integrated, and trusted certificate authentication and issuer validation system based on blockchain,” IEEE Access, vol. 13, pp. 27035–27049, 2025.