International Journal of Science, Strategic Management and Technology
Volume **02** Issue **05** May-**2026** | ISSN: **3108-1762 (Online)** | Impact Factor: 3.8
An International, Peer-Reviewed, Open Access Scholarly Journal Indexed in recognized academic databases

A Secure Voting System using Blockchain Technology

Krishna Rai
Bachelor of Technology, Information Technology
Dr. A.P.J. Abdul Kalam Technical University, Lucknow, Uttar Pradesh, India Email: krishrai8765@gmail.com@gmail.com

https://doi.org/10.55041/ijsmt.v2i5.256

Cite this Article: Rai, K. (2026). A Secure Voting System using Blockchain Technology. International Journal of Science, Strategic Management and Technology, *02*(05). https://doi.org/10.55041/ijsmt.v2i5.256

License: This article is published under the Creative Commons Attribution 4.0 International License (CC BY 4.0), permitting use, distribution, and reproduction in any medium, provided the original author(s) and source are properly credited.

ABSTRACT

Secure voting remains a critical challenge in modern democracies due to concerns over transparency, tampering, and voter trust. This paper proposes a blockchain-based voting system designed to enhance the integrity, security, and reliability of electoral processes. By leveraging the decentralized and immutable nature of blockchain technology, the system ensures that each vote is securely recorded and cannot be altered once submitted. Smart contracts are utilized to automate vote validation and counting, eliminating intermediaries and reducing the risk of human error or manipulation. Cryptographic techniques preserve voter anonymity while ensuring authentication, thereby maintaining both privacy and legitimacy. The distributed ledger allows real-time verification and auditing, increasing transparency and public confidence in election outcomes. Performance analysis indicates that the system is scalable and resistant to common cyber threats such as double voting and unauthorized access.

KEYWORDS: Blockchain, E-Voting, Smart Contracts, Cryptography, Decentralized Application, Ethereum, Zero-Knowledge Proof, Election Security

1. INTRODUCTION

Voting is the cornerstone of democratic governance, enabling citizens to exercise their fundamental rights in electing representatives and shaping public policy. Traditional paper-based voting systems, while widely used, are vulnerable to a range of issues including ballot stuffing, voter impersonation, miscounting, and undue delays in result declaration.

Electronic Voting Machines (EVMs) were introduced to address some of these issues; however, they continue to raise concerns about transparency, auditability, and susceptibility to sophisticated hacking techniques. A central authority controlling the voting infrastructure introduces a single point of failure and raises questions about neutrality and trust.

Blockchain technology, originally conceived as the backbone of cryptocurrency systems, offers a decentralized, transparent, and tamper-proof ledger that is increasingly being explored across diverse application domains. Its core properties — immutability, transparency, and decentralization — make it a highly promising candidate for redesigning the election process.

This paper presents the design and implementation of a secure, decentralized voting system built on the Ethereum blockchain. The proposed system employs smart contracts for automated vote management, cryptographic techniques for voter anonymity, and a distributed ledger for public verifiability. The remainder of the paper is organized as follows: Section 2 reviews related literature, Section 3 describes the proposed methodology, Section 4 presents the results, Section 5 discusses the findings, and Section 6 concludes with directions for future work.

2. LITERATURE REVIEW

A substantial body of research has explored the intersection of blockchain technology and electronic voting. Kshetri and Voas (2018) were among the first to systematically analyze how blockchain-based e-voting could reduce electoral fraud and improve transparency [1]. Their work laid the theoretical groundwork for subsequent technical implementations.

Hjálmarsson et al. (2018) developed a blockchain voting prototype on the Ethereum platform, demonstrating the practical feasibility of smart contract-driven vote casting and counting [2]. Their system showed that automated tallying through smart contracts significantly reduces the risk of human error.

Pawlak et al. (2018) explored the use of permissioned blockchain frameworks such as Hyperledger Fabric for improved scalability and controlled voter access [3]. Their approach highlighted the trade-off between the openness of public blockchains and the performance requirements of national-scale elections.

Fusco et al. (2018) analyzed cryptographic approaches including zero-knowledge proofs to protect voter identity while ensuring vote verifiability [4]. This work demonstrated

© Author(s). This work is peer-reviewed, openly published, and permanently archived
This article is openly accessible and reusable with proper attribution.

that privacy and transparency are not mutually exclusive in a well-designed blockchain voting system.

Shahzad and Crowcroft (2019) evaluated the performance metrics of blockchain voting systems, measuring transaction latency and throughput under load [5]. Their findings indicated that with proper optimization, blockchain voting systems can meet the performance demands of large-scale elections.

Despite these advances, challenges remain. Most existing works highlight the trade-off between decentralization and scalability, the high gas costs on public Ethereum networks, and the digital accessibility gap as primary barriers to widespread adoption. This paper addresses these concerns by proposing an optimized architecture and evaluating its performance comprehensively.

2.1 Problem Statement

Traditional voting systems, whether paper-based or electronic, suffer from critical vulnerabilities including vote tampering, lack of transparency, voter impersonation, and single points of failure. Current electronic systems rely on centralized databases that can be compromised, manipulated, or rendered inaccessible through targeted attacks.

There is a pressing need for a secure, transparent, and decentralized voting platform that: (a) guarantees voter anonymity, (b) prevents double voting, (c) ensures result integrity, (d) provides real-time public auditability, and (e) operates without reliance on a central authority. This paper addresses all five requirements through a blockchain-based architecture.

3. METHODOLOGY

The proposed methodology involves the design and development of a decentralized application (DApp) on the Ethereum blockchain. The system is structured into three primary phases: voter registration and authentication, vote casting and recording, and result tallying and audit.

3.1 System Architecture

The overall system architecture consists of four principal components: a React.js front-end interface, a MetaMask-based authentication layer, Solidity smart contracts deployed on the Ethereum network, and an IPFS-based distributed storage layer for voter registration data. Each component is designed to be modular, enabling independent upgrades without affecting the overall system integrity.

3.2 Voter Registration and Authentication

Voter registration is conducted through a government-verified identity system. Each eligible voter is assigned a unique Ethereum wallet address, which serves as their cryptographic identity. MetaMask is used to manage key pairs and sign transactions, ensuring that only authenticated voters can interact with the voting smart contract.

The registration data, including hashed voter credentials, is stored on IPFS to avoid overloading the blockchain while maintaining decentralized access. The Ethereum smart contract stores only the hash reference, preserving both privacy and integrity.

3.3 Smart Contract Design

The smart contract, written in Solidity, governs three core processes: voter eligibility verification, vote casting, and automated tallying. Key features of the smart contract include:

- Mapping of each voter address to a boolean flag preventing duplicate votes.
- A candidate registry storing candidate identifiers and their respective vote counts.
- A voting window defined by start and end timestamps, enforced on-chain.
- An event emission mechanism that logs each vote transaction for public audit.
- An owner-only function to finalize results after the voting window closes.

3.4 Vote Casting and Privacy

When a voter casts a ballot, their vote is encrypted using their private key and broadcast as a transaction to the Ethereum network. Zero-knowledge proof (ZKP) techniques are employed to allow the smart contract to verify that a vote is valid — i.e., cast by an eligible voter for an existing candidate — without revealing the voter's identity or choice to any external observer.

The ZKP implementation is based on zk-SNARKs (Succinct Non-Interactive Arguments of Knowledge), which generate compact proofs that can be efficiently verified on-chain without exposing the underlying data. This ensures that voter anonymity is maintained even under adversarial scrutiny of the public ledger.

3.5 Technology Stack

- **Blockchain Platform:** Ethereum (Goerli Testnet for evaluation)
- **Smart Contract Language:** Solidity v0.8.x
- **Front-End:** React.js with Web3.js for blockchain interaction
- **Wallet / Authentication:** MetaMask browser extension
- **Distributed Storage:** IPFS via Infura gateway
- **Privacy Layer:** zk-SNARKs via snarkjs library
- **Development Tools:** Hardhat, Remix IDE, Ganache

© Author(s). This work is peer-reviewed, openly published, and permanently archived
This article is openly accessible and reusable with proper attribution.

4. RESULTS

The proposed system was evaluated on the Ethereum Goerli testnet across multiple performance and security dimensions. A series of controlled experiments were conducted with simulated voter populations ranging from 100 to 10,000 users.

4.1 Performance Metrics

The system achieved an average transaction throughput of 150 transactions per second (TPS) under normal network conditions, which is sufficient for constituency-level elections. The average vote confirmation latency was measured at 3.2 seconds, well within the acceptable threshold for interactive user experience.

Scalability tests demonstrated linear performance growth up to 10,000 concurrent voters, with no significant degradation in throughput or latency. Beyond this scale, the use of Layer-2 solutions such as Polygon or Optimism is recommended to maintain performance.

4.2 Security Evaluation

Security testing was conducted across three threat models: Sybil attacks, replay attacks, and man-in-the-middle (MITM) attacks. The results were as follows:

- **Vote Integrity:** 100% of test votes were recorded without modification, confirming blockchain immutability.
- **Double Voting Prevention:** All 500 attempted duplicate votes were successfully rejected by the smart contract.
- **Sybil Attack Resistance:** The one-wallet-one-vote constraint prevented all simulated Sybil attack attempts.
- **Replay Attack Resistance:** Transaction nonces and block timestamps prevented all replay attempts.
- **Anonymity:** ZKP verification confirmed that no voter identity could be inferred from on-chain data.

4.3 Comparison with Existing Systems

Compared to existing EVM-based systems, the proposed blockchain approach offers superior transparency and public verifiability. Unlike centralized electronic systems, results on the blockchain can be independently audited by any observer without requiring trust in a central authority. In comparison to prior blockchain voting proposals [2][5], the integration of ZKP provides a stronger anonymity guarantee while maintaining full on-chain verifiability.

5. DISCUSSION

The results demonstrate that blockchain technology provides a robust foundation for secure, transparent, and tamper-proof elections. The smart contract architecture effectively automates vote validation, eliminating human intervention and the associated risks of bias or error.

The ZKP mechanism successfully decouples voter identity from their vote choice, satisfying both the privacy and auditability requirements that traditional systems struggle to reconcile simultaneously. The IPFS-based registration storage reduces on-chain data load while preserving the decentralized nature of the system.

However, several challenges warrant further consideration. The gas costs associated with Ethereum public network transactions present a financial barrier for large-scale national deployments. Migrating to a permissioned blockchain such as Hyperledger Fabric, or adopting Layer-2 scaling solutions, could substantially reduce operational costs.

The digital divide — encompassing limited internet access, lack of hardware, and low digital literacy among certain voter demographics — remains the most significant non-technical obstacle to inclusive deployment. Addressing this requires complementary policy and infrastructure investments beyond the technical solution itself.

From an academic perspective, this work contributes to the growing body of evidence that blockchain and zero-knowledge cryptography, when carefully combined, can satisfy the five core requirements of a trustworthy election system: security, anonymity, verifiability, fairness, and eligibility.

6. CONCLUSION

This paper successfully demonstrates that a secure, transparent, and decentralized voting system can be built on blockchain technology. The proposed system leverages Ethereum smart contracts, zk-SNARK cryptography, and IPFS distributed storage to ensure vote integrity, voter anonymity, and public verifiability. Experimental results confirm scalability up to 10,000 concurrent voters, resistance to known cyber attacks, and sub-5-second transaction confirmation times.

Future work will explore: (1) integration with national biometric identity systems such as Aadhaar for stronger voter authentication; (2) migration to Layer-2 Ethereum or Hyperledger Fabric to reduce transaction costs; (3) mobile application development to improve accessibility; (4) adoption of post-quantum cryptography to protect against emerging quantum computing threats; and (5) pilot deployment in institutional elections to gather real-world empirical data.

ACKNOWLEDGMENT

The author would like to express sincere gratitude to Ms. Nidhi Chauhan (Assistant Professor, Department of Information Technology) for her invaluable guidance and

© Author(s). This work is peer-reviewed, openly published, and permanently archived
This article is openly accessible and reusable with proper attribution.

continuous support throughout this research. The author also acknowledges the resources provided by NIET Greater Noida and Dr. A.P.J. Abdul Kalam Technical University, Lucknow.

REFERENCES

[1] N. Kshetri and J. Voas, "Blockchain-Enabled E-Voting," IEEE Software, vol. 35, no. 4, pp. 95–99, 2018.

[2] F. Hjálmarsson, G. K. Hreiðarsson, M. Hamdaqa, and G. Hjálmtýsson, "Blockchain-Based E-Voting System," Proc. IEEE 11th Int. Conf. Cloud Computing, pp. 983–986, 2018.

[3] M. Pawlak, J. Guziur, and A. Poniszewska-Mara■da, "Voting Process with Blockchain Technology: Auditable Blockchain Voting System," IFIP Int. Conf. Computer Information Systems, 2018.

[4] F. Fusco, M. I. Lunesu, F. E. Pani, and A. Pinna, "Crypto-voting, a Blockchain Based e-Voting System," Proc. KMIS 2018, pp. 1–8, 2018.

[5] B. Shahzad and J. Crowcroft, "Trustworthy Electronic Voting Using Adjusted Blockchain Technology," IEEE Access, vol. 7, pp. 24477–24488, 2019.

[6] S. Nakamoto, "Bitcoin: A Peer-to-Peer Electronic Cash System," Bitcoin.org, 2008.

[7] V. Buterin, "A Next-Generation Smart Contract and Decentralized Application Platform," Ethereum White Paper, 2014.

[8] E. Ben-Sasson et al., "Succinct Non-Interactive Zero Knowledge for a von Neumann Architecture," Proc. USENIX Security Symposium, pp. 781–796, 2014.

[9] A. Biryukov, D. Khovratovich, and I. Pustogarov, "Deanonymisation of Clients in Bitcoin P2P Network," Proc. ACM CCS, pp. 15–29, 2014.

[10] R. C. Merkle, "A Digital Signature Based on a Conventional Encryption Function," Proc. CRYPTO 1987, Lecture Notes in Computer Science, vol. 293, pp. 369–378, 1988.

© Author(s). This work is peer-reviewed, openly published, and permanently archived
This article is openly accessible and reusable with proper attribution.