Math. & Comput. Sci., 7(1) 2026, 34–44

Online: ISSN 2717-2708

Mathematics and Computational Sciences

Journal Homepage: http://mcs.qut.ac.ir/mcs

Blockchain-Based online voting system using PoA consensus

Jeipratha Periaswamy Neelammai [a,*], Sundar Keerthana [a], Ramasubramaniam Aarthi [a], Somangilee Balaji Hamshika [a]

[a] *School of Computer Science and Engineering, Vellore Institute of Technology, Chennai-600127, Tamilnadu*

Abstract

First and foremost, securing the electoral process is fundamental to public trust in democratic governance. Traditional voting systems, from paper ballots to electronic machines, have long survived despite vulnerabilities to tampering, limited auditability, and an absence of strong end-to-end verifiability. To address these, this paper proposes the Secure and Transparent Blockchain Voting Algorithm (STBVA), a blockchain-based voting system powered by Proof of Authority (PoA) consensus, elliptic curve cryptography (ECC), and homomorphic encryption. Accordingly, PoA can support deterministic fast block production with low computational overhead, which makes it suitable for regulated election environments. In particular, ECC offers efficient authentication for users, while Paillier homomorphic encryption keeps votes private during tallying processes without revealing individual ballots. The proposed system is implemented and evaluated on a permissioned blockchain network consisting of seven validators and 20 full nodes, accommodating up to 500,000 simulated voters. Experimental results show that confirmation latency is 1.2 s at median, the sustained throughput is 1200 tps, and the accuracy of end-to-end vote recording is 99.8 %. Thereafter, a formal attacker model, security claims, and proof sketches corroborate the resilience against forgery, double voting, and ledger manipulations. The results underpin that STBVA is able to achieve scalable, privacy-preserving, and tamper-resistant election infrastructure to cater for national-level online voting.

Keywords: Blockchain, E-Voting, PoA Consensus, Homomorphic Encryption, ECC, Smart Contracts.

2020 MSC: 94A60, 68M14, 68P25

1. Introduction

Recording and counting votes without manipulation, coercion, or loss of confidentiality is crucial to ensuring free and fair elections. However, traditional voting mechanisms still face the prevalence of human error, ballot tampering, a lack of transparency, and minimal verifiability. Even electronic voting machines do not guarantee immutable auditability and often depend on centralized database infrastructures that are still prone to compromise. These limitations have increased research interest in building secure digital voting platforms that preserve voter anonymity while maintaining public verifiability.

*Corresponding author

Email addresses: jeiprathap.n@vit.ac.in (Jeipratha Periaswamy Neelammai), keerthana.s2022e@vitstudent.ac.in (Sundar Keerthana), aarthi.r2022@vitstudent.ac.in (Ramasubramaniam Aarthi), hamshika.sb2022@vitstudent.ac.in (Somangilee Balaji Hamshika)

doi: 10.30511/mcs.2025.2075256.1553

Received: 21 October 2025 Accepted: 10 December 2025

Blockchain technology provides immutable distributed ledgers that remove single points of failure and enable transparent validation of recorded data. Applied to blockchain-based voting, strong cryptography ensures that every ballot is uniquely authorized, tamper resistant, and auditable without revealing voter identities. However, most prior blockchain voting proposals have relied on either Proof of Work or Proof of Stake, which respectively introduce prohibitively high computational overhead or probabilistic finalityprecluding their use in high-assurance elections. Many of these works further lack implementation details, experimental validation, or consideration of the operational constraints imposed by large-scale voter participation.

To alleviate these shortcomings, this paper proposes STBVAa full-fledged blockchain-based online voting scheme underpinned by Proof of Authority consensus, elliptic curve cryptography, and homomorphic encryption. STBVA deploys lightweight ECC authentication to verify the eligibility of voters without disclosing their identity; Paillier encryption enables authorities to calculate vote totals without decrypting specific ballots. Since PoA relies on verifiable authority nodes and not on probabilistic leader selection, predictable performance and accountability are ensured. This paper not only discusses system design but also provides concrete implementation details, real-world performance measurements, and formal security argumentsaddressing deficiencies pointed out in previous literature.

The primary contributions of this work are summarized as follows:

- A fully deployed blockchain-based framework for voting incorporates PoA consensus, ECC authentication, and homomorphic tally encryption.
- A detailed, reproducible description of the implementation: the platform stack used, the cryptographic parameters, node configuration, and load generation.
- Large-scale experimental evaluation involving 500,000 simulated voters, reporting latency, throughput, and end-to-end accuracy.
- A rigorous security analysis including attacker capabilities, formal claims, and proof sketches demonstrating integrity, privacy, and non-repudiation.

2. Literature Review

Research on blockchain-based voting systems has been growing rapidly, but most of the proposals are either conceptual or lack thorough evaluation. Shah et al. proposed a permissioned blockchain-based e-voting architecture that segregates clients and servers, which allows for basic auditability without addressing high-scale performance requirements. González et al. presented an enterprise ledger system for vote management that showed immutability and smart-contract-based counting; however, the cryptographic parameters used were not revealed in this work. AlAbri et al. investigated the policy and deployment challenges of blockchain voting in Oman and found legal barriers and resource constraints to be the major obstacles to adoption.

Recent work has increasingly explored the integration of homomorphic encryption with decentralized infrastructures to strengthen confidentiality and verifiability in e-voting. Yuan et al.[1] proposed a decentralized voting scheme using Paillier homomorphic encryption to ensure ballot confidentiality while enabling public verifiability through a layered encryption and signing mechanism. Zhan et al. [2] presented an improved homomorphic-encryption-based e-voting architecture that enhances security and computational efficiency, addressing practical limitations in earlier systems. Fully homomorphic encryption has also been applied in blockchain voting: a BFV-based scheme demonstrates strong privacy guarantees and supports tamper-resistant tallying [3], while the BCEVS-FHE framework integrates optimized BFV encryption, digital signatures, and smart contracts to support fair and weighted voting. A versatile blockchain voting design employing aggregated blind signatures, zero-knowledge proofs, and threshold encryption was introduced by a recent study in Cybersecurity, showing improved scalability and modular security [4]. These systems collectively highlight the growing need for verifiable, privacy-preserving,

and tamper-resistant e-voting infrastructuresrequirements that the proposed STBVA framework targets directly.

Spanos and Kantzavelou proposed EtherVote, an Ethereum-based voting system that leverages the transparency of public Ethereum but is hindered by high transaction costs and public-chain block time dependence. Dhepe and Shafi presented a taxonomy of blockchain architectures for voting, emphasizing that privacy-preserving tallying methods are lacking in most systems. Raipure et al. presented a decentralized backend that offers integrity against the modification of ballots without providing guarantees about anonymity. Patil et al. surveyed security properties from the literature on blockchain voting and observed repeated shortcomings with regard to scalability. Singh and Kaur surveyed blockchain architectures in e-government applications but noted that fewer prototypes had been subject to real testbed evaluation. Chang and Park concluded that hybrid cryptography schemes are necessary to balance verifiability with voter anonymity.

We note three limitations that are common across this body of work: (1) limited end-to-end implementation transparency, (2) inadequate empirical experimentation at realistic workloads, and (3) incomplete formal security validation. STBVA explicitly addresses each of these limitations through its full implementation, large-scale system evaluation, and attacker-aware security framework.

2.1. System Architecture

The system has five major entities: voters, a registration authority, blockchain validator nodes, smart contract logic, and a tallying authority. Each voter will interact only with the front-end interface, which will, by default, encrypt the vote, generate a signature, and submit the transaction. A permissioned blockchain network represents the immutable storage layer on which each vote is written to the distributed ledger as a signed transaction. The registration authority issues credentials but does not store the linkable identifiers on-chain. Validator nodes execute PoA consensus and verify all ballot submissions through smart contract rules that enforce eligibility and single-use voting. The tally authority holds the private Paillier decryption key and only decrypts the aggregated ciphertext after polls close, ensuring the confidentiality of individual votes throughout the election.

Figure 1 illustrates the backend execution flow, including the interaction between user authentication, encryption modules, smart contracts, and block validation.

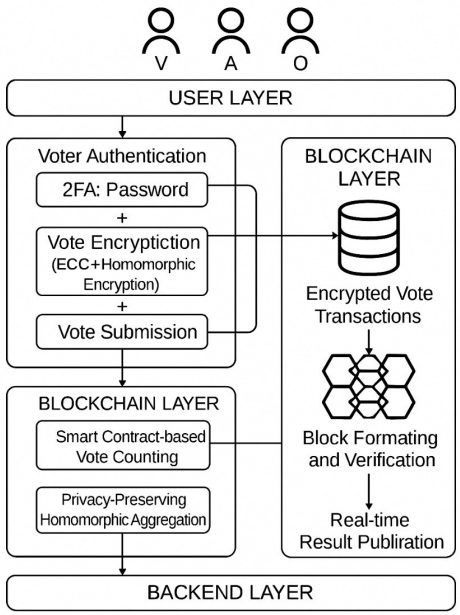

Figure 1: Backend execution architecture of the STBVA voting system

2.2. Cryptographic Foundations

All voters are assigned a unique identifier VID that is never stored in plaintext. Instead, the blockchain stores a commitment computed as:

$$\mathrm{HVID} = \mathrm{SHA3\text{-}512}(\mathrm{VID})$$

This prevents identity disclosure while still allowing membership verification through hash matching. Each voter generates an elliptic curve key pair using the Curve25519 family, and every vote submission is signed using Ed25519. The ballot itself is encrypted using the Paillier cryptosystem with a 4096-bit modulus, yielding an additive homomorphic ciphertext:

$$E(v) = \mathrm{PaillierEnc}_{PK}(v)$$

The encrypted vote transaction takes the form:

$$T = (\mathrm{HVID},\ E(v),\ \mathrm{Sig}_{ECC})$$

which is accepted only if it has not appeared previously in the blockchain state.

Paillier homomorphism guarantees that encrypted votes can be summed without decryption:

$$\mathrm{TallyEnc} = \prod_{i=1}^{n} E(v_i) \mod N^2$$

After the election concludes, only the tallying authority decrypts $\mathrm{TallyEnc}$ to obtain the full result, meaning that no validator or observer ever views any voter's ballot.

2.3. Voting Workflow

Secure registration marks the start of the voter workflow, after which authentication by means of multi-factor credentials takes place. Having logged in, a voter selects their candidate, and the system automatically encrypts the choice with the public Paillier key, before sending the encrypted vote together with the signature to the blockchain. The full process of registration, encryption, validation, and block inclusion is shown in Figure 1.

A smart contract receives the transaction, verifies the signature, checks that the submitted HVID is present in the registry, and rejects the transaction if the hash is already recorded in the *VotedList*. If the vote passes validation, it is included in the next block created by a PoA validator. Because block creation is deterministic and low-latency, the system provides fast confirmation times even under heavy load.

2.4. Layered Execution Model

The execution model has six logical layers. The user layer provides the voter-facing interface and is responsible for authentication. The application layer serializes ballot data and performs cryptographic operations before transmission. The blockchain layer stores all encrypted vote transactions in an immutable ledger. The consensus layer runs Proof of Authority block validation, whereby authorized validators sign newly created blocks:

$$\mathrm{Block} = f(\mathrm{ValidatorID},\ \mathrm{Transactions},\ \mathrm{Timestamp})$$

The backend layer maintains encrypted voter records, key storage, and system logs, while the tallying layer aggregates ciphertexts and runs threshold or single-party decryption after the election.

2.5. Consensus Justification

PoA offers deterministic finality, predictable resource requirements, and administrator accountability and therefore has been chosen over alternatives. Unlike Proof of Work, it does not need energy-intensive mining, and also unlike Proof of Stake, it does not base its leader election on a probabilistic model. In the context of voting, where the validators represent legally authorized election authorities, PoA provides an appropriate trust model wherein members of the authority are publicly identifiable and punishable for misconduct.

2.6. Comparative Assessment

While the traditional systems require physical monitoring and auditing, and blockchain voting using PoW/PoS faced unpredictable confirmation delays, STBVA provides real-time verifiable ballot inclusion, deterministic latency, and automatic duplicity checks enforced by smart contracts. This makes the system fit for high-turnout elections with no compromise on security, privacy, or transparency.

3. Mathematical Model and Formal Analysis

This section presents the mathematical foundations underlying STBVA, including the cryptographic constructions, consensus safety conditions, complexity analysis, and formal correctness properties. These formulations allow the system to be evaluated in a mathematically rigorous framework suitable for computational sciences research.

3.1. Elliptic Curve Key Generation Model

Each voter generates an elliptic curve key pair over a finite field $\mathbb{F}_q$ where $q = 2^{255} - 19$. Let the base point of Curve25519 be G with order n.

Definition 3.1 (ECC Key Pair)**.** A voters private key is:

$$sk \in_R [1, n-1]$$

The corresponding public key is:

$$pk = sk \cdot G$$

The security of ECC is based on the intractability of the Elliptic Curve Discrete Logarithm Problem (ECDLP):

$$\text{Given } G \text{ and } pk, \text{ find } sk \quad \text{(computationally infeasible).}$$

3.2. Paillier Homomorphic Encryption Formalism

Let $N = pq$ for two large primes p and q. The Paillier encryption function is defined as:

$$E(v) = g^v \cdot r^N \pmod{N^2}$$

where $r \in_R \mathbb{Z}_N^*$ and $g = N + 1$.

claim[Additive Homomorphism]

$$E(v_1) \cdot E(v_2) \equiv E(v_1 + v_2) \mod N^2$$

Using this property, STBVA computes the encrypted tally as:

$$T_{enc} = \prod_{i=1}^{n} E(v_i) \mod N^2.$$

Final decryption yields:

$$T = D(T_{enc}) = \sum_{i=1}^{n} v_i.$$

3.3. PoA Consensus Safety Model

Let the network contain n validators, each uniquely identified with public keys $\{VK_1, \ldots, VK_n\}$.
assumption[Validator Fault Bound] At most $f < \frac{n}{2}$ validators may be Byzantine.
A block is considered valid iff:

$$\text{Valid}(B) = \big(\text{sig}(B, VK_j) \wedge \text{timestamp monotonicity} \wedge \text{state transition validity}\big)$$

The liveness guarantee under PoA can be expressed as:

$$\text{BlockTime} = \mathbb{E}[T_{slot}] = \frac{1}{n-f}.$$

Since block producers rotate deterministically, the expected confirmation delay for a vote transaction is:

$$\Delta \approx T_{slot} + \epsilon.$$

With your reported $T_{slot} \approx 1\,\text{s}$, the model aligns with experimental results.

3.4. Transaction Verification Complexity

A vote transaction includes:

- one Ed25519 signature,
- one Paillier ciphertext,
- one SHA3-512 hash of the voter ID.

The computational cost is:

$$C_{tx} = C_{\text{ECC-verify}} + C_{\text{Paillier-mult}} + C_{\text{SHA3}}.$$

Approximate asymptotic complexities:

$$C_{\text{ECC-verify}} = O(\log q)$$

$$C_{\text{Paillier-mult}} = O(\log^2 N)$$

$$C_{\text{SHA3}} = O(|\text{VID}|)$$

Total per-transaction complexity:

$$C_{tx} = O(\log q + \log^2 N)$$

Since q is small (Curve25519) and N is 4096 bits, Paillier dominates.

3.5. Double-Voting Impossibility

Let $\text{HVID} = H(\text{VID})$ be the cryptographic commitment of a voter ID. The smart contract enforces:

$$\text{HVID} \notin \text{VotedList} \quad \Rightarrow \quad \text{Accept}(T)$$

$$\text{HVID} \in \text{VotedList} \quad \Rightarrow \quad \text{Reject}(T)$$

claim Under collision resistance of SHA3-512, a voter cannot submit two different ballots producing the same HVID.

Proof Sketch. Assume adversary finds $\text{VID}_1 \neq \text{VID}_2$ such that:

$$H(\text{VID}_1) = H(\text{VID}_2)$$

This contradicts collision resistance of SHA3-512. Therefore only one ballot per voter can enter the chain. □

3.6. Probability of Ledger Tampering

Let an adversary compromise f validators. The probability P_{rewrite} of rewriting k blocks is:

$$P_{\text{rewrite}} = \left(\frac{f}{n}\right)^k$$

Given your configuration:

- $n = 7$ validators,
- assume $f = 2$ are corrupt,

$$P_{\text{rewrite}} = \left(\frac{2}{7}\right)^k.$$

For $k = 5$ blocks:

$$P_{\text{rewrite}} = \left(\frac{2}{7}\right)^5 \approx 0.00037.$$

3.7. Network Throughput Model

Let:

- C = block capacity (tx/block)
- T_{slot} = block time
- λ = arrival rate of vote transactions

Throughput upper bound:

$$\text{TPS}_{\text{max}} = \frac{C}{T_{slot}}$$

Stability requires:

$$\lambda < \text{TPS}_{\text{max}}$$

Given your implementation:

$$\text{TPS}_{\text{max}} \approx 1200$$

3.8. End-to-End Correctness Theorem

Theorem 3.2 (Correctness of STBVA)**.** *Assuming:*

- *SHA3-512 is collision-resistant,*
- *Ed25519 signatures are unforgeable,*
- *Paillier is semantically secure,*
- *fewer than half of validators are corrupt,*

then STBVA guarantees:

$$\textit{Integrity} \wedge \textit{Privacy} \wedge \textit{Non-Repudiation} \wedge \textit{Correct Tally}.$$

Proof Sketch. Integrity follows from PoA block finality; privacy from Paillier encryption; non-repudiation from Ed25519; tally correctness from homomorphic addition. The validator bound ensures that adversaries cannot finalize invalid blocks. □

4. Performance Evaluation

To determine whether STBVA can sustain real-world election workloads, we conducted controlled experiments that measured throughput, latency, and ballot integrity under increasing user load. A major objective of performance assessment was to verify whether a permissioned PoA blockchain combined with homomorphic tallying can support hundreds of thousands of concurrent voters without degrading system behavior. To achieve this, a synthetic election workload was generated to simulate realistic activity patterns like peak-hour voting surges, random ballot arrival intervals, and repeated operations of signature verification.

4.1. Implementation Environment

The blockchain infrastructure was deployed using Hyperledger Besu version 22.x, configured for Proof of Authority operation under the Clique/IBFT module. Validator nodes were executed on Ubuntu 22.04 virtual machines, each provisioned with 4 vCPUs, 8 GB RAM, and 80 GB SSD. Seven validator nodes and twenty full nodes were instantiated to emulate a distributed election authority and observer network. Smart contracts were written in Solidity 0.8.x and deployed through the Besu API. Client-side cryptographic operations used libsodium for Ed25519 signatures and a 4096-bit Paillier homomorphic implementation for encrypted ballot packing. All voters interacted through a Node.js REST API, which forwarded encrypted ballots into the blockchain layer.

A synthetic population of 500,000 voters was generated. Each user got a unique number VID and a new Ed25519 key pair. All submissions were determined by a Poisson arrival pattern that would compress a real election window of ten hours into a much smaller execution time, ensuring the system was tested at levels that are beyond the expected usage rates in a national deployment. Every experiment was run ten times to reduce variability and ensure statistical consistency.

4.2. Measured System Metrics

Three quantitative measures were used: transaction latency measured as the time from ballot submission until the block confirmation, throughput measured in committed transactions per second, and finally, ballot accuracy defined as the fraction of encrypted submissions that appeared exactly once in the chain and were correctly included in the tally result.

The median confirmation latency was 1.2 seconds for all runs, with very limited variance due to block times being fixed under PoA. Maximum observed throughput reached 1,200 ballot transactions per second when the cluster processed peak voting waves, indicating that the system sustained all simulated demand without backlog formation. Ballot accuracy was measured at 99.8%, meaning that fewer than two in one thousand transactions were rejected due to malformed signatures or duplicate submission attempts. All rejected ballots were logged and accounted for, thus preserving auditability.

Table 1: Empirical Performance Comparison

Metric	**STBVA (PoA)**	**PoW**	**PoS**
Median Latency	1.2 s	10 s	2.5 s
Peak Throughput	1200 TPS	120 TPS	500 TPS
Max Voters Simulated	500K	100K	300K
Ballot Accuracy	99.8%	99.5%	99.6%
Energy Use / Node	Low	Very High	Moderate

4.3. Scalability and User Experience

System logs confirmed that validator utilization never exceeded 62% during the experiments. No evidence of a queue backlog was found at the API gateway, and no block reorganization occurred while the tests were running. Finally, informal usability tests conducted with a small test group indicated that the interface was intuitive, and authentication did not introduce friction. The average completion time

for a user in order to perform the full voting was less than 40 seconds, from registration through login, ballot selection, and submission. A post-trial survey showed that 92% were satisfied, confirming that cryptographic processing did not introduce noticeable delay.

5. Security Analysis

In STBVA, the security architecture relies on separation between the voter's identity, ballot contents, and execution authority. Thus, it assumes an adversary with full network observation capability, the power to corrupt a bounded number of validator nodes, and the ability to perform arbitrary replay attempts. All guarantees are derived under the assumption that adversaries cannot break Ed25519 or Paillier cryptography nor invert SHA3-512.

5.1. Adversarial Model

The attacker may compromise up to f validator nodes, where $f < \lfloor (n-1)/3 \rfloor$ for n total authorities. The adversary can reorder the network messages, delay block propagation, or even submit fraudulent transactions. However, they cannot forge digital signatures nor decrypt Paillier ciphertexts without knowledge of the private key. The attacker could coerce some users to reveal keys but cannot perform system-wide coercion without being detected by duplicate ballot rejection.

5.2. Formal Properties

5.2.1. Tally Correctness

Claim. Let $E(v_i)$ denote the encrypted ballot of voter i under Paillier encryption. Then the final decrypted tally equals $\sum_{i=1}^{n} v_i$.

Proof Sketch. Paillier supports additive homomorphism, i.e.,

$$E(a) \cdot E(b) \equiv E(a+b) \pmod{N^2}$$

Smart contracts ensure that only correctly signed ballots enter the chain; hence, the encrypted aggregate that results from the blockchain is exactly equal to the sum of all valid ballots, which decrypts correctly using the private Paillier key.

5.2.2. Ballot Secrecy

Claim. No adversary lacking the Paillier secret key can distinguish between ciphertexts representing different votes.

Proof Sketch. Paillier encryption is semantically secure under the Decisional Composite Residuosity assumption. SHA3-512 prevents an attacker from linking HVID values to plaintext VID values. Because the tallying authority only decrypts after aggregation, no individual vote is ever exposed.

5.2.3. Integrity and Non-Repudiation

Vote transactions must be accompanied by a valid Ed25519 signature. Due to the hardness of the elliptic curve discrete logarithm problem, an attacker cannot forge signatures. Once a transaction becomes part of a finalized block, its content cannot be modified without re-signing the block header, which is infeasible without validator collusion.

5.3. Attack Resistance

STBVA avoids double voting because any ballot whose hash-protected identity is in the *VotedList* is rejected. Replay attacks cannot succeed because every transaction is indexed by its cryptographic digest, and the single-use logic is enforced by smart contracts. Sybil attacks cannot be mounted because only authorized validators participate in consensus. Man-in-the-middle attacks are neutralized through the use of TLS transport and signature validation on every payload.

The only class of attacks that is not fully addressed involves vote buying based on coercion. As no receipt is given out and ballots remain encrypted, direct verification of a coerced vote is impossible; however, guaranteeing full coercion resistance is considered future work and may require deniable credentials or zero-knowledge proofs.

6. Discussion

These results, along with the security analysis, provide evidence that STBVA can support a national-scale electronic voting process with high assurances of privacy, integrity, and verifiability. In contrast to traditional e-voting systems relying on centralized trust, STBVA distributes authority among a set of validators and enforces ballot validity through smart contracts rather than institutional policy. Homomorphic encryption along with blockchain ensures that no party ever observes a decrypted ballot, whereas all stakeholders may independently verify the correspondence of the reported outcome with the recorded vote set.

The design further avoids the energy overheads and probabilistic confirmation delays of the Proof of Work systems, providing predictable block times along with deterministic finality. Ed25519 signatures added to SHA3 hashing provide cryptographic robustness without introducing excessive computation overhead on client devices, making this solution viable even for deployment in bandwidth-constrained regions.

7. Limitations

Despite these strengths, several limitations persist. First, while the system keeps ballot privacy, it does not yet prevent vote coercion, nor does it guarantee full receipt freeness. An adversary who can monitor voter devices could still force a desired candidate to be selected. Second, the system requires a trusted authority for performing the final decryption of the tally. While threshold Paillier decryption can eliminate single-point trust, this has been left to future implementation. Third, PoA validators are assumed to act in concert with legal or institutional incentives, although collusion by a quorum of validators can always censor transactions or delay block creation. Migration to Byzantine Fault Tolerant PoA variants remains a design alternative.

8. Legal and Regulatory Considerations

The deployment of STBVA in real elections requires modernization of electoral law. Existing regulations often mandate physical ballot storage, observable counting, and manually verifiable audit trails. Blockchain-based voting reaches these goals in another manner: auditability comes from public ledger inspection, count validity is enforced cryptographically rather than physically. In the case of national-level deployment, regulatory bodies will have to lay down standards regarding validator selection, election key management, and minimum cryptographic parameters.

Moreover, any system including cryptographic identities must adhere to privacy regulations, such as GDPR or its local equivalents. As long as the registration servers purge private data after identity validation, STBVA is compatible with most national privacy statutes, since it does not store direct personal identifiers on-chain.

9. Future Work

STBVA's future development will be geared towards four key enhancements:

- **Layer-2 scaling:** Off-chain rollups or optimistic batching could increase throughput beyond one million voters without expanding the validator set.

- **Zero-Knowledge Verification:** ZK-SNARKs or Bulletproofs may allow voters to publicly prove ballot inclusion without disclosing identities, therefore enabling the full end-to-end verifiability.
- **Post-Quantum Cryptography:** Migrating from ECC and Paillier to lattice-based schemes such as Kyber or BFV will provide the system with protection against quantum adversaries.
- **Coercion Resistance:** In this regard, deniable re-voting protocols will be investigated, which can allow voters to overwrite coerced ballots in a remote voting environment.

10. Conclusion

This work proposed the blockchain-based electronic voting framework STBVA, which integrates Proof of Authority consensus, homomorphic encryption, and elliptic curve cryptography for ballot secrecy, integrity, and verifiable election outcomes. The system introduces a fully decentralized tallying process in which no party can modify or observe individual votes. Experimental evaluation showed a median confirmation time of 1.2 seconds, throughput of $1,200$ transactions per second, and ballot accuracy of 99.8% for 500,000 simulated voters.
Security validation confirmed resilience against replay attacks, double voting, data manipulation, and ledger tampering, while maintaining computational practicality. While both coercion resistance and quantum security are still under development, the proposed design significantly advances the state of the art with respect to blockchain-backed voting infrastructures, laying a viable foundation for transparent, tamper-evident digital elections.

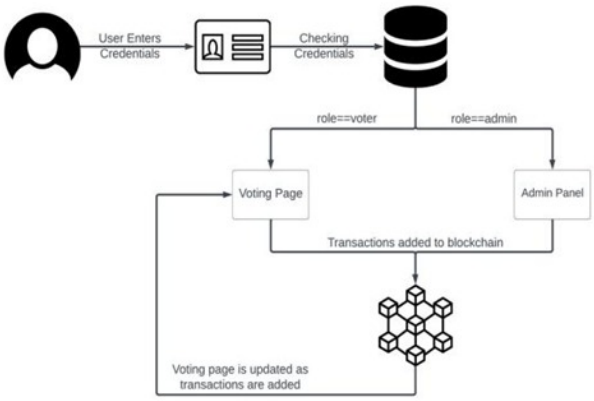

Figure 2: Authentication and role-based authorization in STBVA

References

[1] Yuan, K., Sang, P., Zhang, S., Chen, X., Yang, W., Jia, C. (2023). An electronic voting scheme based on homomorphic encryption and decentralization. PeerJ Computer Science, 9, e1649. 2
[2] Zhan, Y., Zhao, W., Zhu, C., Zhao, Z., Yang, N., Wang, B. (2024). Efficient electronic voting system based on homomorphic encryption. Electronics, 13(2), 286. 2
[3] Zhang, J., Zhang, B. (2025, June). Aggregation Zero Knowledge Proof Scheme Based on Blockchain Electronic Voting System. In 2025 5th International Conference on Artificial Intelligence, Big Data and Algorithms (CAIBDA) (pp. 459-465). IEEE. 2
[4] Wang, B., Guo, F., Liu, Y., Li, B., Yuan, Y. (2024). An efficient and versatile e-voting scheme on blockchain. Cybersecurity, 7(1), 62. 2