A Systematic Review of Frameworks, Performance Metrics, and Threat Analysis in Blockchain-based E-Voting Systems

Muthu Meenakshi Ganesan
Department of Computer Science, Faculty of Science and Humanities, SRM Institute of Science and Technology
Kattankulathur, India
mg0480@srmist.edu.in

Sabeen Selvaraj
Department of Computer Science, Faculty of Science and Humanities, SRM Institute of Science and Technology
Kattankulathur, India
sabeens@srmist.edu.in

Abstract: *Blockchain technology in electronic voting systems has numerous advantages in terms of improving security and optimizing the efficiency of the system. This paper presents a comprehensive review of 53 peer-reviewed research articles on blockchain-based voting systems published between 2020 and 2025. To conduct the systematic literature study, the review methodology presented in this article describes the design frameworks implemented in existing systems and applies 12 evaluation criteria for conducting a systematic comparative analysis. The review further examines how the design of voting protocols, blockchain frameworks, and cryptographic techniques improves ballot secrecy, voter privacy, and system scalability. Based on the findings, several strategies for system implementation have been outlined, highlighting the impact of platform selection and consensus design on system performance. The study concludes by highlighting unresolved technical and operational limitations, underscoring the need for optimized, scalable, and trustworthy solutions for next-generation electronic voting systems.*

Keywords: *Blockchain, blockchain frameworks, electronic voting, consensus, cryptography, scalability, threat analysis.*

Received June 16, 2025; accepted December 18, 2025
https://doi.org/10.34028/iajit/23/4/3

1. Introduction

Blockchain is a decentralized, distributed registry of transactions recorded in a secure distributed system [11, 27, 36]. In 2008, when blockchain was applied as an enabling technology of Bitcoin, a peer-to-peer cryptocurrency, Satoshi Nakamoto established the groundwork for further investigation and future applications [43, 65]. Blockchain technology creates tamper-resistant records; hence, it is appropriate for applications where building trust in a system requires the absence of a central authority [3, 47]. The unique industries that are dependent on blockchain technology are health care systems, financial services, supply chain management, online identity management, and intellectual property systems [25, 46, 52]. Sensitive governance applications, particularly voting systems, can also be appropriate with blockchain, as it has verifiable records that cannot be altered [73].

The traditional method of voting, e.g., the use of paper ballots and their manual counting system, is subject to error and can easily be tampered with [29]. Indicatively, since 2005, Estonia has been legally permitting its electorate to vote through the Internet, thereby earning the distinction of being the longest-running nationwide Internet-based voting system [68]. Its later system, IVXV, provided a refined form of verifiability, using cryptographic proofs [30]. In Switzerland, online voting on formal referendums was simpler than the traditional method [20]. Besides all these developments, issues of centralization, cybersecurity, and data integrity also have to be considered in the case of Internet voting systems. The adoption of blockchain technology to improve the integrity and transparency of elections is a phenomenon that has begun to be discussed recently. The merits of decentralized technology, security, and reliability of blockchain-based election systems can be integrated with the effectiveness of the electronic voting process [34, 57, 71]. The blockchain-based voting system is a decentralized architecture, where votes are stored in a tamper-proof, verifiable, and anonymous manner. Despite these benefits, the barriers to practical deployments remain challenging due to the ongoing issues of scalability, voter anonymity, and implementation complexities.

The paper provides a summary of the recent developments in the use of blockchain-based voting systems. The remaining part of the study is structured in the following way: section 2 outlines the study as consisting of three phases. Section 3 presents the design strategy phase, which starts the review by specifying the review objectives. This section describes the exclusion criteria used for the selection of high-quality research articles and outlines the proposed evaluation criteria for evaluating the selected articles. Section 4 describes the procedure for choosing the relevant research articles.

Section 5 presents significant results on technical, security, and performance analyses concerning blockchain-based electronic voting systems. This section also discusses the trends, trade-offs, and optimization aspects. Finally, Section 6 summarizes the review with future research directions and applications in this area.

2. Review Methodology

In this study, the systematic review was adopted in order to examine the state-of-the-art blockchain-based e-voting systems. Figure 1 outlines how the review was conducted in three phases: design strategy, systematic exploration, and insights mapping. The review objectives, exclusion criteria, and evaluation criteria were discussed under the design strategy. Within the systematic search framework, major databases were screened to identify high-quality research articles based on predetermined exclusion criteria. The insights mapping stage involved a meticulous study of selected articles with high-quality standards to determine and categorize essential results. Also, this phase has highlighted the research gaps and challenges of the existing studies that would impact future developments in this field.

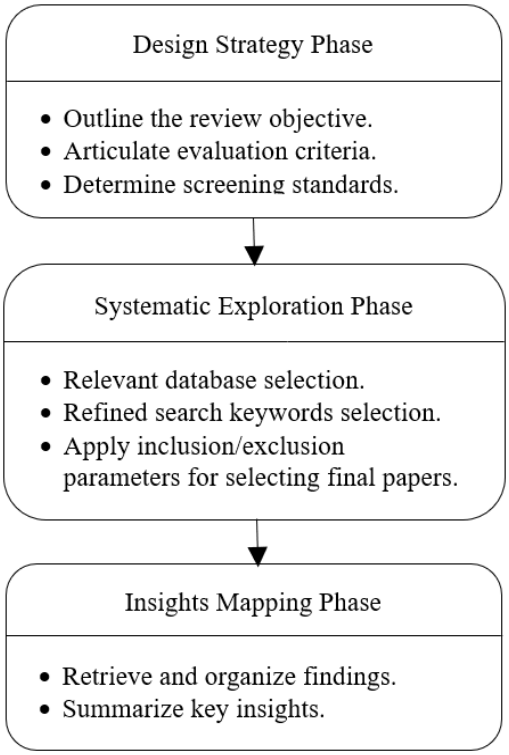

Figure 1. Review methodology.

3. Design Strategy

This phase initiates the review process by setting the review objectives, guidelines, and standards for selecting and examining relevant research papers.

- **Review Objective:** this study aims to identify the research gaps and challenges in blockchain-based voting systems that will drive the future direction of research in this domain.
- **Evaluation Criteria:** a uniform evaluation of blockchain-based voting systems was carried out through twelve Evaluation Criteria (EC1-EC12) on performance, security, and technical aspects, including the following:

  EC1: Blockchain frameworks and types.
  EC2: Consensus mechanisms.
  EC3: Off-chain storage mechanisms.
  EC4: Voter authentication methods.
  EC5: Hashing algorithms.
  EC6: Vote encryption methods.
  EC7: Anonymity and ballot privacy techniques.
  EC8: Development and verification tools.
  EC9: System deployment environments.
  EC10: Scalability approaches.
  EC11: Operational cost and system performance metrics.
  EC12: attack types and threat analysis.

- **Screening Standards:** the criteria for excluding articles in the screening process were established to ensure that the articles chosen for this study met high-quality standards. Articles published before 2020 and not written in English were excluded. This is because we focused only on recent developments in blockchain-based voting systems that are being used. Conference papers, book chapters, and review articles were excluded as the review focused only on peer-reviewed research. Articles that are not directly related to blockchain-based voting and contain only an abstract were also excluded. Finally, articles found across more than one database were eliminated to prevent duplication. These filtration methods preserve concentration, trustworthiness, and methodological coherence.

4. Systematic Exploration

In this phase, research articles were selected by applying a systematic search method, which is outlined in the following discussion.

Database selection: to identify peer-reviewed, high-quality research articles, we used renowned scientific databases, including Scopus, SpringerLink, ScienceDirect, IEEE Xplore, MDPI, and Wiley Online Library.

Search keywords: a specific set of keywords was used during the literature search to access relevant studies on blockchain-based e-voting systems. The final refined search string employed was ("Blockchain" AND ("Secure" OR "Scalable" OR "Privacy preserving" OR "Anonymous" OR "Smart contract") AND ("E-voting" OR "Voting" OR "Election System")).

Selection of final papers: in the selected databases, including Scopus (725), Springer (2,599), ScienceDirect (2,660), IEEE Xplore (499), MDPI (79),

and Wiley (1,510), a total of 8,072 research papers were retrieved. As shown in Figure 2, these 8,072 articles were retrieved from the selected databases. The selection process of final papers applied established screening standards, focusing only on research articles published in English between 2020 and 2025. Further filtering was conducted to ensure that studies were relevant to blockchain-based e-voting systems and had accessible full texts. Duplicate articles were then removed. This process resulted in a final set of 53 studies selected for detailed review and analysis. The distribution of these selected articles is illustrated in Figure 3, organized by year of publication.

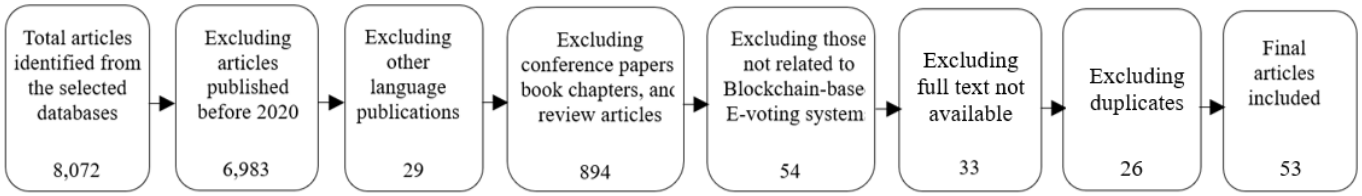

Figure 2. Flow diagram of literature screening process.

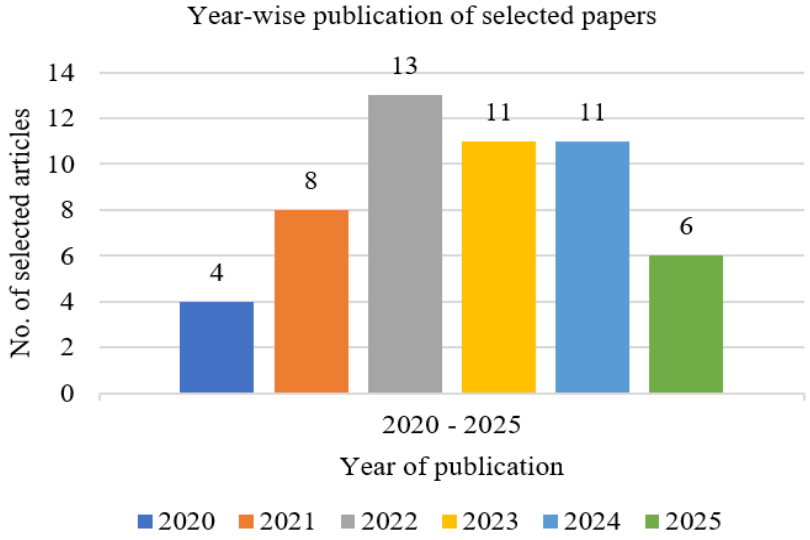

Figure 3. Year-wise publication of selected papers.

5. Insights Mapping

5.1. Findings Based on Evaluation Criteria

This section provides an overview of the important findings identified and organized from the selected studies using the predetermined evaluation criteria. A comprehensive result analysis of tables and figures is used to present the findings. Each subsection focuses on individual evaluation criteria and provides an extensive evaluation and comparison, along with descriptive explanations where needed.

- **EC1: Blockchain Frameworks and Types**

Blockchain networks are commonly categorized into four types based on access and control mechanisms: public blockchains are open to everyone, allowing unrestricted participation and transaction verification. Private blockchains are limited to authorized individuals, ensuring full control over activities. In permissioned setups, only verified members are involved, operating under the guidance of a central authority. Hybrid models combine the strengths of public and private blockchains, allowing open access for certain operations while keeping sensitive information limited to selected participants.

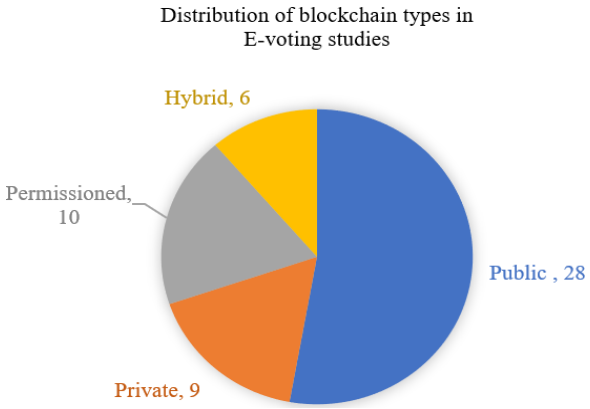

Figure 4. Distribution of blockchain types in e-voting studies.

Figure 4 shows that, among the 53 studies analyzed, 28(~53%) employed public blockchains, 9(~17%) used private blockchains, 10(~19%) adopted permissioned blockchains, and 6(~11%) implemented hybrid architectures. Table 1 summarizes the types of blockchain and frameworks used in the selected articles. Figure 5 illustrates that a substantial amount of research work has been conducted on e-voting, which was based on several key blockchain frameworks.

Table 1. Blockchain frameworks and types.

Blockchain type	Framework	Article
Public	Ethereum	[5, 6, 7, 9, 21, 22, 24, 32, 44, 45, 48, 49, 55, 56, 59, 60, 61, 62, 63, 64, 72]
	BSC	[23]
	Polygon	[31, 35]
	Algorand	[42]
	Solana	[8]
	Stellar	[10]
	Dash	[40]
Private	K-Chain	[1]
	Ethereum with PoA	[15, 53]
	Exonum	[37]
	InnoBlock	[26]
	Custom blockchain	[33, 38, 67, 69]
Permissioned	Hyperledger fabric	[14, 16, 18, 19, 50, 66]
	Custom blockchain	[2, 17, 51, 70]
Hybrid	Custom blockchain	[28, 54]
	Solana and DPLT	[13]
	Ethereum and private blockchain	[4]
	Ethereum with PBFT	[58]
	Ethereum and quorum	[12]

Ethereum was employed in 21 research studies, as shown in Table 1. This showcased the mature smart contract platform and the ease of deployment it offered. Its gas fees raised cost concerns for large-scale elections. Also, its public design can compromise the privacy of the voters. During large-scale voting, high decentralization will increase latency; however, in return, it will bring more transparency and reduce the risk of central control.

Hyperledger fabric [14, 16, 18, 19, 50, 66] was the second most used platform and was clearly recognized for its applicability in consortium-based applications. The permissioned framework further improved control and accountability, thus facilitating secure and verifiable voting systems. This may diminish voter trust; however, compared with more decentralized systems, even if the costs are reduced by avoiding transaction fees. Custom blockchains were implemented in private, permissioned, and hybrid environments. They were designed according to particular security, privacy, or performance requirements. These approaches showed how researchers were focused on tailored designs to address specific security and privacy requirements.

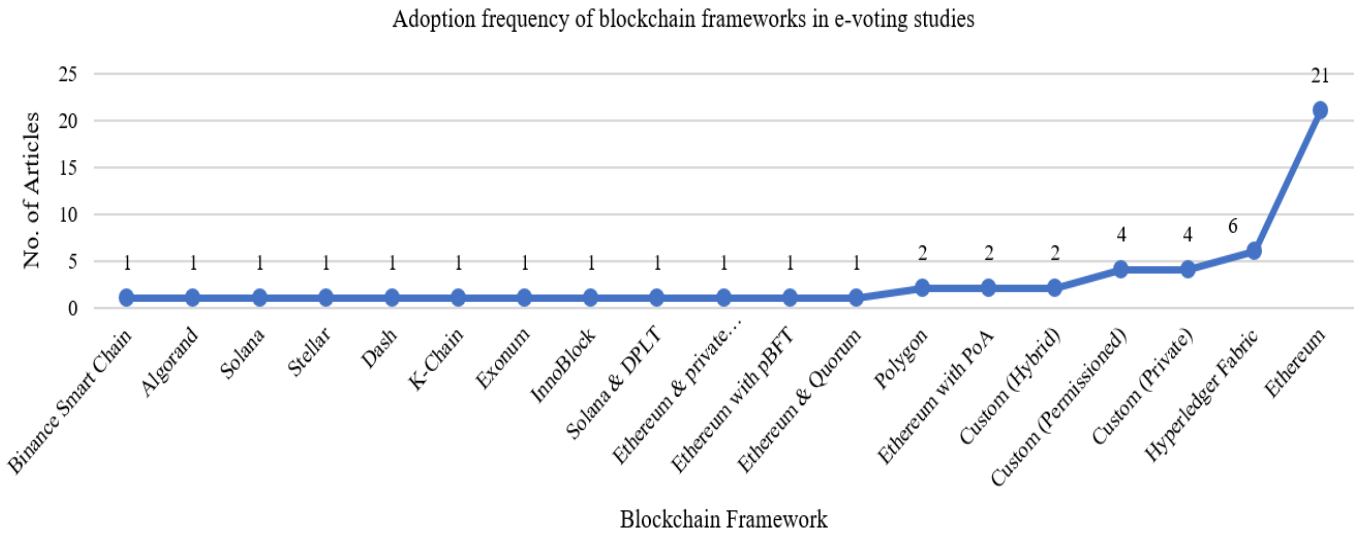

Figure 5. Adoption frequency of blockchain frameworks in e-voting studies.

Binance Smart Chain (BSC) [23], Polygon [31, 35], Algorand [42], Solana [8], Stellar [10], Dash [40], K-Chain [1], Ethereum with Proof of Authority (PoA) [15, 53], Exonum [37], InnoBlock [26], Solana with Delegated Proof of Loyalty Token (DPLT) [13], Ethereum with a private blockchain [4], Ethereum with Practical Byzantine Fault Tolerance (PBFT) [58], and Ethereum with Quorum [12] were discussed in only one or two works. This means their applications in electronic voting are still in the early stages. For instance, BSC, Polygon, Algorand, Stellar, and Dash offered faster and cheaper transaction processing than Ethereum. At the same time, the limited use of these blockchains in e-voting applications raised serious questions regarding trust and auditability. Private blockchain platforms like Exonum, Innoblock, and K-chain offer greater efficiency, but they lack transparency. Scalable blockchain frameworks such as Solana faced challenges with limited decentralization. These differences emphasized that retaining a balance among security, efficiency, and voters' trust remains an unsolved challenge.

- **EC2: Consensus Mechanisms**

The consensus mechanism ensures that nodes on a blockchain network reach an agreement on the ledger consistency without central control. Therefore, consistency and trust are guaranteed within the decentralized system. This section describes articles based on the consensus algorithms they used, as listed in Table 2. The most widely used consensus mechanism was Proof of Work (PoW), which increased resistance against tampering. However, it was unsuitable for real-world applications due to the high energy consumption and low processing speed. On the other hand, Proof of Stake (PoS) increased the efficiency level with much

lower energy consumption. but it could introduce the issue of centralization if the stakes are unevenly distributed. Raft is a lightweight and fast consensus protocol used in the permissioned setup. But it introduced robustness challenges for open and decentralized networks.

Table 2. Classification of articles based on consensus algorithms.

Consensus algorithm	Article
PoW	[1, 5, 9, 32, 40, 44, 45, 49, 55, 59, 62, 67, 70]
Raft	[14, 18, 19, 50, 66]
PoVI	[56]
PoS	[7, 21, 22, 23, 31, 60, 64, 72]
PoA	[15, 53]
Specialized consensus algorithms	[54, 69]
Modified PoS	[35]
Proof of history and PoS	[8, 13]
PoS and proof of credibility	[4]
Electoral PoS and Kafka	[42]
PBFT	[28, 58]
Modified PBFT	[33]
PoW and PoA	[24]
PBFT and Raft	[37]
PoW and PoS	[12]
FBA	[10]

PBFT offered secure, low-latency consensus and was resistant against malicious nodes, but scaled poorly due to high communication overhead [28, 58]. PoA [15, 53] supported fast block confirmation with only minimal computational resources. However, it relied on trusted validators, thus reducing the level of decentralization. Stellar's Federated Byzantine Agreement (FBA) consensus algorithm [10] offered fast, decentralized, secure, low-cost transactions without requiring heavy energy mining. In [54], an Elliptic Curve Digital Signature Algorithm (ECDSA)-based consensus mechanism was used to ensure timestamp authentication, thus providing enhanced security.

Singh and Sinha [56] introduced Proof of Voter's Identity (PoVI), a voter identity-validated consensus approach. Verma *et al*. [69] proposed the Lottery Voting-based Consensus Algorithm (LVCA) for ensuring fairness in block creation with minimal consensus latency.

Various hybrid and modified models explored in [4, 8, 12, 13, 24, 33, 35, 37, 42] investigated the combined benefits of the existing mechanisms. Technical reporting gaps were present in 10 articles that lacked evidence of any consensus procedure used. Figure 6 presents the distribution of the consensus algorithm across the reviewed literature. While PoW still predominates, there was a notable shift toward those consensus mechanisms that emphasized the efficiency, security, and scalability features, including PoS, PBFT, Raft, ECDSA-based consensus, and LVCA. These mechanisms helped to make e-voting applications more scalable and secure.

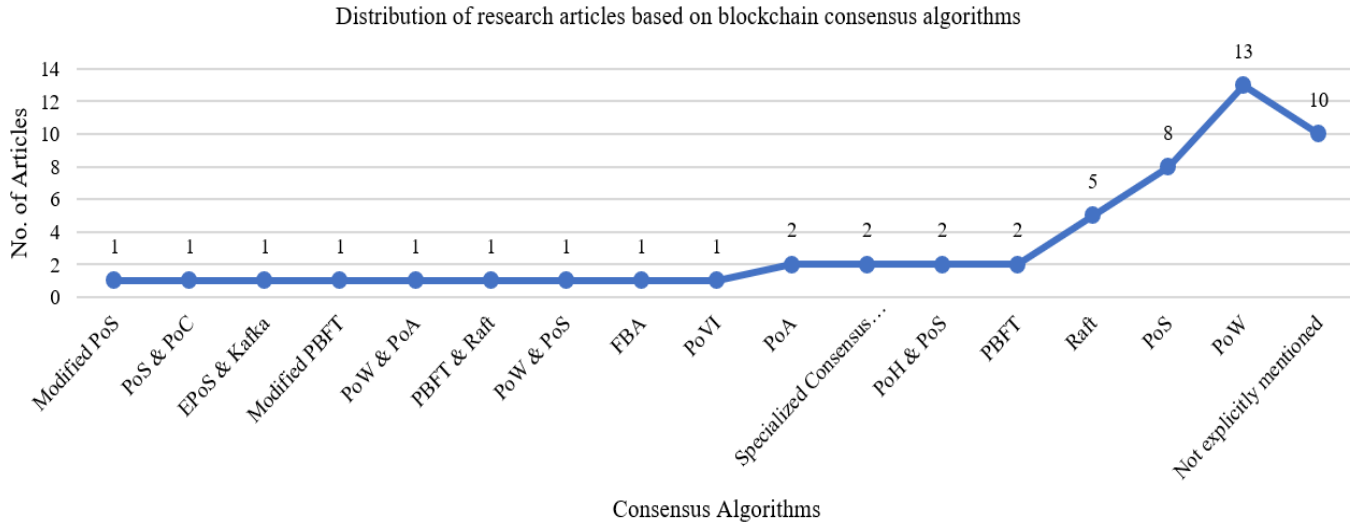

Figure 6. Article distribution by consensus algorithm.

- **EC3: Off-Chain Storage Mechanisms**

Off-chain storage was often used in blockchain-based voting systems to reduce the load on the blockchain. Such solutions, however, inherently involve some trade-off between scalability and decentralization. The different platforms used in the literature, summarized in Table 3, have different trade-offs. InterPlanetary File System (IPFS) was utilized in [5, 10, 23, 35, 58, 69] through content addressing to support decentralization and immutability, which made it suitable for maintaining transparency and voters' trust. In contrast, it struggled with persistence, relying on external services to pin, and it was much slower in terms of data retrieval. Several systems [4, 28, 44] relied on the use of cloud-based storage services. They provide high scalability, low-latency access, and are very easy to integrate with existing systems, which makes them suitable for large-scale elections. But their centralized architecture may introduce a single point of failure and lower trust than the decentralized alternatives. Similarly, both Firebase [8, 55] and MongoDB [12] offered flexible and fast access, thereby making their usage easy, especially in mobile applications. Generally speaking, IPFS is more suitable for the trusted

blockchain model, but with performance challenges, while cloud and database solutions offer efficiency at the cost of transparency. Further research might achieve a balance in these competing requirements through hybrid solutions, such as using IPFS for the critical ballots' storage and cloud storage for metadata.

Table 3. Off-chain storage platforms in e-voting.

Off-chain storage	Article
IPFS	[5, 10, 23, 35, 58, 69]
Cloud storage	[4, 28, 44]
Firebase	[8, 55]
MongoDB	[12]

- **EC4: Voter Authentication Methods**

Authentication mechanisms in blockchain-enabled e-voting systems can be broadly categorized into cryptographic verification, multi-factor authentication, and credential-based authentication. cryptographic verification methods relied on blockchain-native primitives, including wallet addresses, digital signatures, Zero-Knowledge Proofs (ZKPs), and smart contracts. These methods were used to enable decentralized and privacy-preserving voter authentication, as demonstrated in [6, 7, 14, 15, 17, 21, 22, 24, 26, 31, 32, 33, 40, 42, 45, 48, 53, 59, 61, 62]. Multi-factor authentication schemes enhanced security by combining multiple factors, including voter ID, biometrics, One-Time Passwords (OTPs), and device-specific credentials, as implemented in [8, 9,12, 19, 23, 28, 35, 37, 38, 44, 49, 50, 51, 55, 58, 60, 63, 64, 66]. Credential-based authentication employed voter IDs, passwords, encrypted tokens, or pseudonymous addresses secured with cryptographic protocols, as utilized in [1, 2, 4, 5, 10, 13, 16, 18, 54, 56, 67, 69, 70, 72]. Notably, biometric verification techniques were integrated in [19, 35, 37, 38, 44, 49, 50, 51, 55, 58, 63, 64, 66], reflecting their growing importance in ensuring secure and user-centric voter authentication.

- **EC5: Hashing Algorithms**

Hashing protects blockchain-based e-voting systems against tampering. It is also used to ensure that the ballots are not compromised while maintaining the voters' identities hidden. Another feature of hashing is used to create a unique hash for off-chain data saved in IPFS and recorded on the blockchain. This unique tamper-proof reference securely links on-chain and off-chain data, allowing integrity verification.

Keccak256 is well-known for its secure cryptographic capabilities and resistance to collisions, and has been utilized in [23, 56, 62]. SHA-256 is a widely employed cryptographic function that produces a 256-bit hash. It is reliable and collision-resistant, implemented in [1, 5, 7, 8, 9, 12, 17, 18, 23, 24, 38, 53, 54, 55, 60, 61, 66, 67, 70]. Ethash, a proof-of-work algorithm, was used by Tanwar *et al*. [62] to resist ASIC domination in Ethereum using computational and memory hardness. Cryptographic hashing functions were implemented by Rupa *et al*. [49] using the Dagger algorithm.

- **EC6: Vote Encryption Methods**

Many blockchain-based electoral systems are listed in Table 4 based on encryption techniques in order to ensure the confidentiality of votes. In [31, 53, 60], Rivest Shamir Adleman (RSA) was employed for secure key distribution and the secrecy of votes. Elliptic Curve Diffie Hellman (ECDH) and Quantum Key Distribution (QKD) were used by Mahanayak *et al*. [35] to ensure secure key transfer and protection against quantum attacks.

Table 4. Vote encryption methods.

Technique used	Article
RSA	[31, 53, 60]
ECDH and QKD	[35]
ElGamal	[17, 44, 45, 72]
Paillier	[32, 63, 67]
AES and ECDH	[2, 10]
AES	[55]
AES, RSA	[70]
VAOE	[16]
Symmetric encryption and Curve25519	[14]
Public-key cryptography	[4, 7, 28, 33, 49, 58, 64]
Implied but not specified	[8, 12, 26, 37, 51]

Symmetric encryption, especially the Advanced Encryption Standard (AES), has become a popular choice for most blockchain-based e-voting systems due to its fast and high security. In [55], AES was implemented on Ethereum for low-overhead encrypted voting with vote confidentiality. In [2, 10], hybrid methods used AES with ECDH, while another approach in [70] employed AES with RSA for the key distribution and voter privacy.

Clarke *et al*. [16] developed the Verifiable Aggregator Oblivious Encryption (VAOE) technique that leveraged homomorphic encryption based on the properties of bilinear maps. These techniques allowed aggregation and verifiable computation over encrypted data. The implementation of ElGamal-based algorithms in [17, 44, 45, 72] and Paillier encryption in [32, 63, 67] also permitted the secure counting of encrypted votes. A lightweight design was proposed by Chaisawat *et al*. [14] that combined symmetric encryption and Curve25519-based ECDH to enable efficient key exchange. Several articles [4, 7, 28, 33, 49, 58, 64] implemented public-key cryptography, but they did not reveal the details of the specific algorithm utilized. Besides, in some works [8, 12, 26, 37, 51], encryption was discussed but without mentioning the cryptographic algorithms used. Such limitations did not permit deep study, reproducibility, or comparative analysis. In general, AES was used frequently to encrypt votes, and the hybrid design with homomorphic encryption provided more security, verifiability, and scalability.

- **EC7: Anonymity and Ballot Privacy Techniques**

In this section, it is highlighted that cryptographic methods were employed to achieve voter anonymity,

making voters use voting systems without insecurity. ZKPs allow voters to verify their votes are valid without revealing their true identity or any details of their ballots. For instance, Singh and Sinha [56] used Non-Interactive Zero-Knowledge Proofs (NIZKPs) to verify votes without disclosing the identities of voters. Similarly, to prevent the possibility of multiple voting while keeping voters' identities anonymous, Sangraula and Adhikari [53] utilized Zero-Knowledge Succinct Non-Interactive Argument of Knowledge (zk-SNARK) along with nullifiers. In [45], the anonymity in the voting process was guaranteed by randomized ballots with Non-Interactive Zero-Knowledge (NIZK) proof.

In [64], the system can be utilized securely by voters with zero-knowledge proof, biometric hashing, and SoulBound tokens. Whereas in [17, 61], the identities of the voters remained unexposed by utilizing zk-SNARKs and Merkle trees, and ballots were validated with zero-knowledge proof in [18]. Similarly, in [59], cyclic group encryption was combined with NIZKP to ensure the impossibility of vote tracking. Further to mask the ballots, vote masking and Zero-Knowledge Set Membership (ZKSM) proofs have been discussed by Alshehri *et al*. [6]. The blind signature is essential in anonymous voting, whereby voters cast their ballots without their personal details being revealed to the authorities. For instance, VAOE was combined with blind signatures by Clarke *et al*. [16] to enhance privacy.

Meanwhile, blind signatures along with Elliptic Curve Cryptography (ECC)-based commitments were utilized by Hu *et al*. [24] to offer anonymity for voting. Majumder *et al*. [37] suggested an advanced approach by incorporating ZKPs, blind signatures, and ECDH for dissociating the voters from their ballots. Moreover, a blind signature mechanism that provided prevention against malicious activities was introduced by Baranski *et al*. [10], whereas an RSA blind signature with burner addresses was used in [31]. Mix networks and Oblivious Transfer (OT) are some of the other approaches adopted for privacy enhancement in e-voting systems. For example, Reversible Transaction Flooding (RTF) was employed by Abed *et al*. [1] to camouflage the legitimate votes between a set of dummy transactions, thereby concealing the source of votes. In [72], OT disrupted the direct linkage of voters to votes, which ensured that votes could not be traced back to voters' details. The exponential ElGamal encryption, along with secure deletion, was adopted by Panja *et al*. [44] to keep the anonymity of the ballot.

Multiple works in [9, 15, 21, 23, 32, 40, 42, 48, 55, 62, 67] established baseline privacy by employing persistent pseudonyms derived from standard wallet addresses. In [35, 58], decentralization and confidentiality were enhanced by using IPFS Content Identifiers (CIDs) along with encrypted vote data. A credential-based pseudonymous system with private key-based authentication was presented by Abuidris *et al*. [4]. Blockchain Transaction IDs (BCTIDs) generated from public keys in [49] and untraceable pseudonymous vote identifiers were utilized in [50] to avoid associating any vote with a particular voter. Furthermore, Ahn [5] enhanced session privacy using AES-encrypted addresses and ephemeral digital signatures, enabling voters to participate anonymously in a secure, one-time session. In [33], smart contracts enhanced voter privacy through the application of a dual blockchain. They were also designed to carefully dissociate the voted data from the identity of the voter [26].

In [7], cryptographic hashing and time-locked ballot encryption were used to ensure voters' anonymity. Faruk *et al*. [19] used dual-factor authentication, such as voter ID and biometric verification, for pseudonymous authentication. In [13], secure and authorized vote submissions were further enabled by digitally signed messages through DPLT. Anonymous voter IDs were used in [60] and, in combination with ECDSA in [54], enhanced the privacy. The IPFS-based CID hashing employed in [69] has allowed decentralized and pseudonymous authentication, and the strategies in [8, 28, 70] have aimed at securing voter data through encryption. In [38], SHA-256 hashing and biometric verifications were employed, and in [2], AES was employed with SHA-256 to enhance privacy. In [66], receipt freeness was used together with biometrics to avoid coercion and retain anonymity. In [51], privacy was improved through encryption schemes with traceability of QR codes. In the meantime, pseudonymous voting was facilitated by asymmetric JSON Web Tokens in [14]. In [22], plaintext votes were stored without identity linkages, undermining security. Homomorphic encryption with Shamir secret sharing in [63] contributed to privacy, but biometric associations were still present. Likewise, Chaabane *et al*. [12] employed SHA-256 and encryption methods, including voter ID linkages. The reviewed techniques, such as ZKPs, blind signatures, and mix networks, effectively protected voter anonymity and ballot privacy in e-voting systems.

- **EC8: Development and Verification Tools**

The development and verification tools used in the reviewed articles are presented in Table 5. They are categorized into programming languages, smart contract frameworks, testing tools, blockchain libraries, front-end and back-end platforms, databases, Software Development Kits (SDKs), and cryptographic tools. Smart contracts were mostly developed in Truffle, Hardhat, or Remix, and languages of choice for this purpose include Python, JavaScript, and Solidity. The development tools used for interaction with the blockchain include Web3.js, Ethers.js, and MetaMask. Node.js was used to build server-side services. Certain articles did not specify any development tools, as they were focused more on theoretical concepts or high-level system design rather than practical implementation.

Table 5. Summary of tools for blockchain.

Category	Tool	Articles
Programming languages	Solidity	[5, 6, 7, 9, 12, 16, 21, 22, 23, 24, 31, 32, 35, 45, 48,49, 55, 56, 58, 60, 62]
	JavaScript	[6, 7, 9, 12, 14, 18, 19, 21, 23, 24, 31, 35, 38, 45, 48, 53, 58, 62, 67]
	Python	[2, 33, 51, 69]
	Golang (Go)	[15, 16, 18, 50]
	Dart and C++	[1]
	PyTeal	[42]
	Lua	[26]
Smart contract development and deployment	Remix	[56, 59, 62, 70]
	Truffle	[6, 7, 9, 12, 32, 35, 48, 55, 56, 59, 62, 64]
	Hardhat	[23, 24, 31]
Testing and verification	Mocha	[62]
	Scyther	[37]
	Mythril	[22]
	Hyperledger caliper	[6, 14, 16, 50]
	Chaincode scanner, Gsec and Glint	[18]
Blockchain interaction and APIs	Web3.js	[5, 21, 45, 48, 56, 61, 62, 64]
	Ethers.js	[23, 24]
	Infura	[35, 48]
	MetaMask	[7, 9, 12, 21, 32, 35, 48, 49, 56, 60, 62, 64]
	Pera wallet	[42]
	Phantom wallet	[8]
	Dash core wallet	[40]
	Unspecified wallet	[4, 15, 23, 45, 53, 55]
	Geth	[15]
Front-end and UI frameworks	ReactJS	[7, 58, 60, 64]
	AngularJS	[54]
	Flutter	[1, 64]
Back-end frameworks and runtimes	Node.js/NPM	[4, 9, 12, 18, 19, 23, 28, 48, 62, 67]
	Spring boot	[54]
	NestJS	[35]
	Flask	[35]
IDE	Visual studio code	[23]
Databases	MySQL	[54]
	SQLite3	[2, 51]
	MerkleDB	[37]
Blockchain SDKs and custom tools	Hyperledger fabric SDK	[14, 18, 50, 66]
	InnoBlock SDK	[26]
Storage integration	Pinata SDK	[23]
Cryptography libraries	Crypto.js	[24]
DSL for ZK circuits	Circom	[61]

- **EC9: System Deployment Environments**

Blockchain-based e-voting deployment environments range from testnet simulations to cloud platforms and mainnets. Table 6 presents that most studies were conducted in simulation environments. Ganache was the most commonly used simulation environment, since it was very easy to use for local testing. Kovan [56], Ropsten [24, 32, 48], Sepolia [60], and Mumbai [31] were some of the public testnets being used for testing systems under realistic network conditions. Other works were deployed in alternative frameworks, such as testnets from hyperledger fabric [18, 19, 50, 66], Algorand [42], and Solana [8]. Further works have reported their deployments using AWS cloud infrastructure [4], IBM Cloud [16], and Heroku [67] to simulate real-world deployment. Among these, only two works reported real-world mainnet deployments, one used the Ethereum mainnet [21] and the other used live voting data on the Dash blockchain [40]. It indicates that only a few works in the literature reported mainnet deployments. This limitation suggested the need for further research in real-world deployment using scalable and secure implementations.

Table 6. Deployment environments used in blockchain-based voting systems.

Environment type	Deployment environments	Article
Simulation	Ganache	[6, 7, 9, 28, 35, 45, 49, 53, 55, 62, 64]
	Kovan	[56]
	Ganache and raspberry PI	[12]
	Amazon elastic compute cloud	[4]
	Sepolia	[60]
	Hyperledger fabric (local deployment)	[18, 19, 50, 66]
	Single-host docker and multi-host kubernetes	[14]
	IBM cloud's free-tier Kubernetes cluster	[16]
	Free-tier heroku cloud deployment	[67]
	BSC Testnet	[23]
	Algorand TestNet	[42]
	Ropsten	[48]
	Ropten and ganache	[32]
	Mumbai testnet	[31]
	Solana testnet	[8]
	RPC-based ethereum testnet	[5]
	Ropsten and rinkeby	[24]
	Stellar testnet	[10]
	Local blockchain simulation	[1, 2, 13, 15, 17, 22, 26, 33, 37, 38, 44, 51, 54, 58, 59, 61, 63, 69, 70, 72]
Mainnet deployment	Ethereum mainnet (IFA voting case)	[21]
	Live dash blockchain governance system with real-world voting data	[40]

- **EC10: Scalability Approaches**

Blockchain-based e-voting systems critically lack scalability in large-scale elections. Various techniques, such as cloud and edge integration, consensus optimization, hybrid architecture, and distributed storage, were used to enable scalable voting. Table 7 illustrates these approaches along with their contributions. Some promising hybrid blockchain frameworks included Solana-Firebase [8], Dual-Blockchain [33], and Quorum-Ethereum [12]. They supported parallel processing and reduced the load on nodes by segregating on-chain and off-chain activities. Scaling up the required resources to boost the performance was possible through cloud and edge deployments in IBM cloud-based hyperledger fabric [16], Firebase [55], and Edge-Kafka [42]. However, these approaches depend on centralized control. Techniques aimed at improving consensus and transaction optimization, such as PoVI [56], threshold-based block creation [1], and parallel processing [45], increased scalability at the cost of security or privacy.

Load diminishing on the blockchain was achieved through decentralized storage techniques such as IPFS in [5, 58] and Merkle trees in [17, 37]. However, ensuring integrity needs careful integration in these approaches. Nevertheless, several scalability challenges remain to be overcome, such as security compromises, high cost, and a lack of adequate testing in the real world. Scalable voting systems require researchers to focus more on the design of modular systems along with optimized consensus processes.

Table 7. Summary of scalability mechanisms used in blockchain-based e-voting systems.

Article	Scalability technique	Key contribution
[19]	Cloud infrastructure integration	Leverages cloud infrastructure to dynamically scale resources such as nodes and computational power during high-demand periods, ensuring responsiveness in voting.
[1]	Threshold-based block creation and RTF	Enhances scalability by batching transactions into fewer blocks to reduce overhead and obscures real votes through dummy transactions that flood the network without bloating the blockchain.
[56]	Energy efficient consensus	PoVI, a non-competitive and randomized validator selection method, reduces computational load and enhances scalability compared to traditional PoW consensus.
[60]	Batch processing for vote tallying	Improves scalability by retrieving encrypted votes in large batches (10,000 votes per call), avoiding Ethereum's gas limitations.
[54]	Lazy launching	Separates consensus from transaction verification and execution, reducing computational overhead and improving throughput during voting.
[53]	Threshold secret sharing	Splitting the decryption key among multiple parties enables off-chain reconstruction, reducing on-chain operations.
[23]	BSC blockchain	Its higher transaction throughput and low cost make it suitable for fast voting while maintaining Ethereum Virtual Machine (EVM) compatibility.
[69]	Trust-based miner selection and IPFS storage	Combines miner selection via trust-based lottery with IPFS for decentralized storage, distributing workloads efficiently, and reducing blockchain congestion.
[35]	Polygon, IPFS, and OrbitDB with CRDTs.	Polygon, IPFS, and OrbitDB are combined in a layered framework to enhance scalability and limit blockchain bloat.
[13]	Hybrid architecture	DPLT enables off-chain verification with Solana for scalable voting and fast, lightweight on-chain logging.
[45]	Parallel transaction processing and optimized NIZK	Splits vote operations into parallel transactions to bypass gas limits while using efficient NIZK proofs for verification.
[4]	Hybrid consensus and sharding	Achieves parallel processing with secure validator assignment for improved scalability.
[67]	Lightweight private blockchain	Eliminates gas fees and complex smart contracts, enabling higher throughput.
[2]	Distributed intermediate servers with round-robin scheduling	Horizontally scaling voting operations by distributing workload across nodes enhances scalability.
[55]	Firebase cloud integration	Offloads authentication/data storage, enabling scalable voter registration and real-time high-volume updates.
[16]	IBM cloud-based hyperledger fabric	Enables scalable e-voting through dynamic resource allocation, parallel transaction processing, and lightweight node architecture.
[42]	Edge computing with Kafka	Kafka decouples edge processing from the main blockchain, enabling efficient batch aggregation and reduced latency.
[28]	Edge server and cloud server integration	Transactions were processed locally by edge servers, while validation was delegated to the cloud, leading to greater scalability and lower latency under heavy traffic.
[72]	Optimized group encryption scheme	Minimizes system-wide computation via a group encryption scheme and ElGamal, enabling large-scale self-tallying with abort resilience.
[17]	Merkle tree for commitments	To store voter commitments, reduce validation complexity, and support 400 million votes.
[44]	Hybrid blockchain-cloud storage	Blockchain secures keys and tallies, while cloud storage handles ballots for scalability and efficiency.
[50]	Dynamic block sizing and timeout	Adjusts block size and timeout to optimize throughput.
[33]	Dual-blockchain architecture	Separates voter blockchain from ballot blockchain, reducing per-node storage overhead and improving parallel processing.
[58]	IPFS for decentralized storage	IPFS stores voting data with unique cryptographic hashes, linked to blockchain records for tamper-proof retrieval.
[8]	Hybrid architecture	The system combines Solana's scalability with Firebase authentication for voter verification.
[5]	IPFS integration	Decentralized storage of smart contract documents reduces blockchain bloat while maintaining data integrity via cryptographic hashes.
[37]	Merkle trees for lightweight verification	Enables efficient vote validation through hash trees, eliminating full ledger traversal while ensuring integrity and scalability
[6]	ZK proof optimization	The Fiat–Shamir heuristic transforms ZKSM into a non-interactive proof, reducing communication overhead and improving scalability.
[12]	Hybrid blockchain architecture	Implements scalability by separating voting operations into private Quorum blockchains and aggregating results via a public Ethereum blockchain.
[10]	Hybrid architecture	Stellar enables fast, low-cost voting, while IPFS stores voting metadata off-chain with on-chain hashes to ensure scalability

- **EC11: Operational Cost and System Performance Metrics**

Various studies have tested their implementations under different conditions to determine the feasibility and effectiveness of the voting systems. The following subsections summarize the findings on the efficiency, cost, and other important metrics of the existing systems for identifying the capabilities and limitations in this field.

Gas Cost per Vote Transaction

Cost per vote transaction refers to the cost of processing each vote on-chain. Several studies reported the cost per vote transaction, which are listed in Table 8. These values depend on the blockchain platform used, transaction complexity, and the fee structure. A direct comparison of these costs was less meaningful since some were expressed in fiat currency while others were in cryptocurrency.

Table 8. Gas cost per vote transaction.

Article	Gas cost/vote
[53]	2.63$
[21]	0.02ETH
[23]	0.13$
[7]	0.0076521ETH
[13]	0.00001$
[45]	12.39$
[67]	0.73$
[42]	0.001Algos
[49]	0.018191ETH
[31]	0.015$
[10]	0.00006XLM

Gas Units for Smart Contract Deployment

The gas units represent the resource expenditure invested in deploying the smart contracts. These values have indeed varied greatly, as shown in Table 9, for several blockchain platforms such as Ethereum [5, 7, 15, 24, 45, 53, 58], Polygon [31], and BSC [23]. Complex contracts required more gas units because these included voting or cryptographic functions. Ethereum required more gas units due to the complexity and congestion within the blockchain network. On the contrary, Polygon and BSC required fewer gas units, which means contracts were deployed in a highly effective way. Hence, intelligent selection and optimization of platforms are important to save costs during the deployment of smart contracts.

Table 9. Gas units for smart contract deployment.

Article	Gas units
[53]	7,347,623
[23]	1,905,638
[7]	(Voter) 614,032
	(Candidate) 579,917
	(Voting) 734,353
[45]	(voting) 5,498,780
	(cryptography) 3,698,878
[31]	4,209,467
[15]	2,221,279
[58]	138,758
[5]	102,418
[24]	2,353,838

Encryption Time

Encryption time refers to the time taken to securely encode a vote or block of data before it is added to the ledger or transmitted over the network. Only three works reported encryption times explicitly, as illustrated in Table 10. Vinayachandra and Krishna [70] and Tas and Tanrıover [63] measured the encryption time of each vote using hybrid Advanced Encryption Standard-Rivest Shamir Adleman (AES-RSA) and Paillier cryptosystems, respectively. On the other hand, Salman *et al*. [51] reported the encryption time of 1000 blocks using homomorphic encryption. Overall, AES-RSA and Paillier were faster in terms of encryption, whereas homomorphic encryption provided higher security at the cost of more computational overhead.

Table 10. Encryption time.

Article	Encryption Time
[70]	100ms/vote
[51]	4.55m/1000 blocks
[63]	94ms/vote

Latency and Delay Metrics

Latency is defined as the time taken by the system to react to an action, while delay refers to any waiting time within the processes that contribute to the overall latency. Various studies reported that this system metric ranges from 1.3 milliseconds in [58] to about 27 seconds in [4], as in Table 11. These variations were influenced by the effects of system architecture, consensus procedures, and network conditions.

Table 11. Latency and delay metrics.

Article	Type	Value
[69]	Transaction latency	1.9-3.3ms
	Consensus delay	100.0096ms
[4]	Hybrid consensus	27s
[28]	Transaction latency	7.55ms
	Authentication delay	1.5ms
[58]	Transaction latency	3.76ms
	Authentication delay	1.3ms
[24]	Transaction latency	9.42s
[14]	Query transaction latency	0.18s
	Invoke transaction latency	1.24s
[37]	Computation overhead	56.3264ms
[6]	Transaction latencies	~10s
	(read and write)	16-35s
[63]	Transaction latency	5774ms
[10]	Consensus delay	3-5s

Proof Generation Time

The cryptographic proofs generated by the proof generation methods were used to prove the validity of the transaction without exposing the identity information of the voters. In Table 12, a summary of proof generation durations measured across different studies is given. The time to produce proofs was lower in two studies that used zk-SNARKs with the Groth16 protocol [53, 61], but the approach suggested by Damavandi and Nogoorani [17] was based on a custom circuit, which was slower. It once again showed how the cryptographic methods and the choice of implementation influenced the system performance.

Table 12. Proof generation time.

Article	Proof generation time
[53]	~2.18s
[17]	2.55m
[61]	10,109ms

Block Creation Time

Block creation time is a metric that covers the duration of creating a new block and its inclusion in the blockchain. Block size, consensus process, network efficiency, and system design are the primary variables influencing this system metric. As shown in Table 13, the lowest block creation time range, 250-500 ms, was measured by Vinayachandra and Krishna [70], followed by Chafiq *et al*. [13].

Table 13. Block creation time.

Article	Block creation time
[69]	20s
[13]	400ms
[42]	90s
[70]	250-500ms
[15]	5s

Response Time

The time taken to handle the voting requests is the system response time. These ranged from 5.23 milliseconds for 30,000 voters by Shekhar and Yadav [54] to 12.75 minutes for 15,725 voters by Christyono *et al*. [15], as shown in Table 14. The differences reflected their system scalability and optimization techniques used.

Table 14. System response time for varying voter sizes.

Article	Response time
[54]	5.23ms/30,000 voters
[67]	0.73m/10,000 voters
[28]	7.19ms/60,000 voters
[15]	12.75m/15,725 voters
[38]	~500ms/1000 voters

Vote Casting Time

The vote-casting times in Table 15 showed that the method proposed by Umar *et al*. [67] reported the lowest vote-casting time of 0.73 minutes for 10,000 votes (~4.38 ms per vote) among all studies. In certain systems, this was much smaller using either efficient cryptography methods or optimized consensus methods.

Table 15. Time required to cast votes.

Article	Vote casting time
[53]	3.02899s/vote
[45]	~7.2ms/vote
[67]	0.73m/10,000 votes
[70]	50ms-100ms/vote
[61]	0-1.8min/vote

Vote Tallying Time

Only two articles measured the time required to count votes and publish the election results. For example, Panja *et al*. [45] reported the tallying vote time per candidate as shown in Table 16. Umar *et al*. [67] measured this system metric for a large-scale voter base of 10,000 participants.

Table 16. Vote tallying time.

Article	Vote tallying time
[45]	~5.6ms/Candidate
[67]	1.56m/10,000 voters

Vote Integrity Metrics

The vote alteration rate is the percentage of votes that can be tampered with after they have been cast. The vote alteration rates reported by some studies, as illustrated in Table 17, were low. The minimal value of 1.5% reported by Somasekhar *et al*. [58] indicated that their data integrity measures were sufficient.

Table 17. Vote integrity metrics.

Article	Vote alteration
[54]	1.92%
[28]	1.9%
[58]	1.5%

Throughput

The evaluation of throughput was done on the basis of two operational parameters: Transactions Per Second (TPS) and Votes Per Second (VPS). The number of transactions handled per second on the blockchain network is called TPS. Voting, registration, and consensus steps were in these transactions. As Figure 7 indicates, Chafiq *et al*. [13] had a high throughput, employing Solana; however, Verma *et al*. [69] had a strong blockchain infrastructure that provided decent TPS. Some studies have indicated that their throughputs were significantly lower. Their complex processing and low scalability, therefore, resulted in low performance.

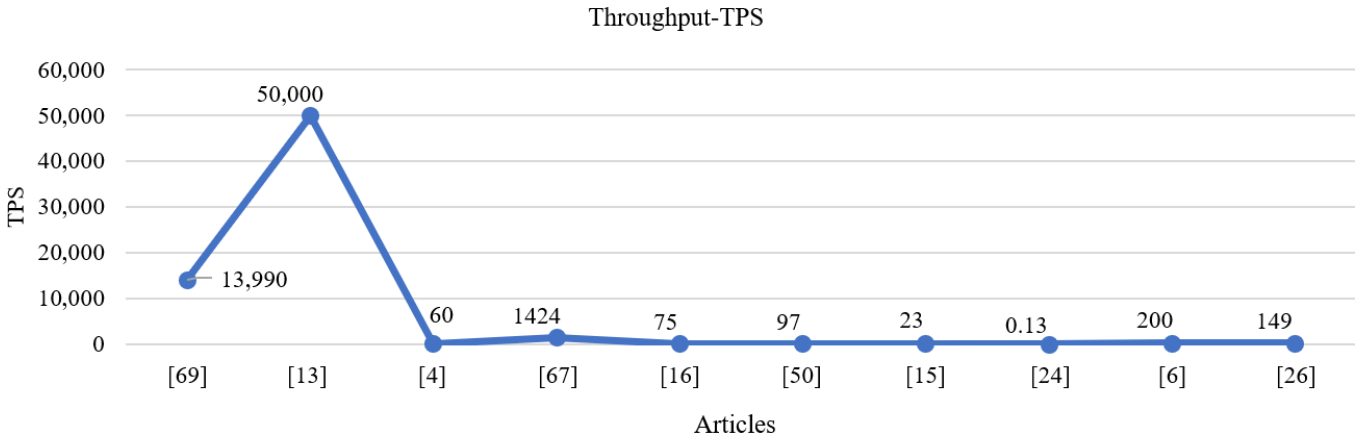

Figure 7. Throughput analysis (TPS).

VPS, on the other hand, measures the number of ballots that are registered every second. Figure 8 summarizes the VPS outcomes in different articles. Vinayachandra and Krishna [70] recorded the vote recording rate of 200 VPS, which was indicated as good performance. Singh *et al*. [55] and Baranski *et al*. [10] showed lower throughputs, and they represent moderate performances. In addition to the above-mentioned performance metrics, various other works also mentioned some other key system attributes, such as biometric accuracy, authentication performance, computation complexity, and cryptographic methods. The performance of the authentication schemes was determined by Faruk *et al*. [19] showed that fingerprint recognition was better than face recognition. False acceptance and false rejection rates of fingerprint

recognition were 1.2 and 2.5, respectively, as opposed to 2.5 and 3.0 in face recognition. Fingerprint verification took an average of 1.8 seconds, and face recognition took an average of 3.1 seconds. A biometric system presented by Faruk *et al*. [19] reported an accuracy of 87% in a combination of fingerprint and facial recognition, whereas a face recognition accuracy of 99% reported by Salman *et al*. [51].

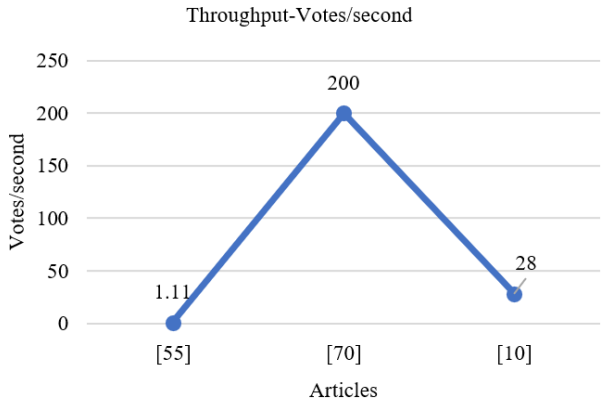

Figure 8. Throughput analysis (VPS).

In [45], the computational complexity of the decentralized voting counting protocol was measured to be $O(nk^2)$, where n is the number of electors, and k denotes the number of candidates. As given by Hu *et al*. [24], the ECC-based blind signature algorithm produced signatures for 50 voters within two seconds in local testing, ensuring its efficiency. zk-SNARK proofs and Merkle trees were integrated by Damavandi and Nogoorani [17] to verify voter commitments and ballot integrity. Tang *et al*. [61] used a 20-level Merkle tree structure to handle up to 1,048,576 voter identities for anonymous authentication. These performance metrics indicate that an optimal balance among security, scalability, and performance should be sought in practical applications.

- **EC12: Attack Types and Threat Analysis**

Several blockchain-based e-voting studies investigated various attack vectors that violate the integrity, confidentiality, and availability of the system. These attacks are summarized in Table 18. Most researchers explored different types of cyberattacks, such as Denial of Services (DoS), Distributed Denial of Services (DDoS), Sybil, spoofing, and vote manipulation. DoS/DDoS and coercion attacks were discussed most, since these two types of attack directly impact the participation of voters and the reliability of the system.

Estonia's i-Voting elections (2005-2015) were a real-world example of remote Internet voting [68]. They granted voters using personal computers the facility to cast ballots from unsupervised environments. Such unsupervised environments created coercion and external influence threats in remote voting systems. Replay attacks compromise vote integrity, which Mestel *et al*. [39] investigated in-depth. In [59], proof-of-concept prototypes of Ethereum-based voting smart contracts emphasized the validation of timestamps and nonces to prevent replay attacks. According to the Mueller Report by the US Department of Justice, military intelligence agents from Russia conducted spear-phishing attacks during the 2016 US presidential election [41].

Table 18. Reported attacks in blockchain-based e-voting studies.

Attack type	Nature and description of attacks	Articles addressed
Sybil attack	Multiple fake identities are created to manipulate or disrupt the network.	[16, 18, 37]
Spoofing attack	A fake identity or biometric is used to impersonate a legitimate voter.	[18, 19, 60]
Snooping attack	Eavesdropping on sensitive voting data in transit.	[1]
Coercion attack	Voters are forced or influenced to vote in a particular way.	[1, 7, 10, 17, 35, 44, 66]
51% attack	The majority control of the network is used to alter the blockchain.	[2, 4, 24, 28, 37, 60]
MITM attack	Interception and tampering with communication between users.	[18, 35, 37, 60]
DoS/DDoS attack	Overloading the system with traffic to block legitimate access.	[2, 16, 18, 24, 28, 31, 37, 38, 60]
Vote alteration attack	Legitimate votes are modified, deleted, or replaced.	[28, 54, 58]
Quantum attack	Breaking encryption using quantum computing.	[35]
Replay attack	Resending old, valid messages to trick the system.	[2, 37, 45]
SQL injection attack	Exploiting input fields to access or alter backend databases.	[2]
Byzantine attack	Nodes behave maliciously or inconsistently to disrupt consensus.	[42]
DAO attack	Logical flaws in smart contracts are used to hijack voting operations.	[37]
Phishing attack	Fake websites or messages are used to steal voter credentials.	[18, 37]
Eclipse attack	Isolating a node to feed it false or limited information.	[18]
Wormhole attack	Relaying blockchain data to unauthorized locations outside the network.	[18]

These attacks targeted local election officials and the private voting software company to steal the voters' data and login credentials. These incidents have shown that electronic voting systems can be vulnerable to various cyberattacks in digital environments. Therefore, any future blockchain-based voting systems should be designed with context-specific threat models, anti-replay validation, robust voter authentication, and coercion-resistant design in order to ensure security against such threats. Other emerging threats, such as quantum and wormhole attacks, have also been explored. Advanced attacks, including Eclipse and Decentralized Autonomous Organization (DAO)-style governance vulnerabilities, however, remain comparatively understudied.

5.2. Summary of Key Insights

This section provides an overview of recent developments in blockchain-based e-voting, including the trends and further actions that need to be taken.

Among the platforms used publicly, Ethereum is flexible due to its mature functions for smart contracts. Hyperledger fabric dominates in permissioned environments more than custom blockchain solutions, which are suitable for specific security or privacy needs. PoS, PBFT, and Raft are proven to be efficient alternatives to energy-intensive PoW. Hybrid consensus algorithms, such as Electoral PoS and LVCA, are also drawing attention as they balance decentralization, security, and scalability.

Off-chain storage is often combined with blockchain-based voting mechanisms to reduce latency and enhance scalability. Cloud storage offers scalability but introduces trust issues, and IPFS ensures immutability but is slow; hybrid approaches can be feasibly applied in future implementations that integrate both components to combine the advantages of these two approaches. Different research studies have been proposed based on biometrics, digital signatures, and multi-factor authentication as the core of voter identification. While biometric authentication improves the security of the authentication process, it raises various privacy, usability, and accuracy problems. Digital signatures and ZKPs ensure the authenticity and anonymity of voters.

Hashing and encryption are two critical elements of any secure electoral system. SHA-256 and Keccak256 are the widely used hash functions for data integrity. Encryption ensures the confidentiality of ballots using AES, RSA, and homomorphic cryptographic algorithms. Asymmetric and symmetric methods are also used in most articles, but without the details of their cryptographic techniques, therefore makes reproducibility challenging. Blind signatures, mix-nets, and ZKPs are utilized to enable voters' privacy and to maintain the secrecy of ballots. Techniques like zk-SNARKs and NIZK proofs also contribute to the elegant verification of votes without compromising voters' privacy. However, achieving a balance among privacy, auditability, and system performance remains a challenge.

On the development side, Solidity, Truffle, Hardhat, and Hyperledger caliper are some of the tools widely used in implementing and testing the voting system. Most of the systems are deployed in test environments such as Ganache, Ropsten, and other testnets. Only a few studies report on actual mainnet deployment, which indicates a gap between simulation and real-world applications.

While Polygon and Solana are able to process transactions quickly, they typically minimize a bit of decentralization. Other performance-improving initiatives that are underway include sharding, off-chain storage, and hybrid cloud-blockchain models. Still, many systems have high transaction costs and performance limits, particularly those deployed on Ethereum.

Existing systems address the most common risks, such as Sybil attacks, spoofing, and vote manipulation, with security as the highest priority. DoS attacks are examined in numerous studies. Emerging threats, such as wormholes and quantum-based attacks, are starting to be addressed in the literature. Advanced vectors, such as Eclipse and DAO attacks, are not handled by many studies.

6. Conclusions

Generally, blockchain-based e-voting systems have the potential to enhance integrity, transparency, and voter trust in the electoral process. Nevertheless, a number of obstacles, including the high operational cost, poor scalability, and usability concerns, obstruct their implementation in practice. Future research efforts must focus on improving privacy-sensitive algorithms and consensus protocols that are highly scalable. Concretely, more severe security threats, including next-generation quantum attacks, network-level eclipse attacks, and DAO-type governance attacks, need to be studied and addressed in subsequent research. Real-world testing and deployments would be necessary to test the usability and efficiency of systems in real-time applications. The electoral systems built on blockchain must address both technical and ethical issues to demonstrate security, efficiency, and scalability in large-scale applications.

References

[1] Abed H., Al-Zoubi O., Alayan H., and Alshboul M., "Towards Maintaining Confidentiality and Anonymity in Secure Blockchain-based E-Voting," *Cluster Computing*, vol. 27, no. 4, pp. 4635-4657, 2023. https://doi.org/10.1007/s10586-023-04194-5

[2] Abo-Akleek F., Mowafi M., Taqieddin E., and Shatnawi A., "Leveraging Blockchain for Robust and Transparent E-Voting Systems," *Cyber Security and Applications*, vol. 3, pp. 1-9, 2025. https://doi.org/10.1016/j.csa.2025.100086

[3] Abrar I. and Sheikh J., "Current Trends of Blockchain Technology: Architecture, Applications, Challenges, and Opportunities," *Discover Internet of Things*, vol. 4, no. 1, pp. 1-17, 2024. DOI: 10.1007/s43926-024-00058-5

[4] Abuidris Y., Kumar R., Yang T., and Onginjo J., "Secure Large-Scale E-Voting System Based on Blockchain Contract Using a Hybrid Consensus Model Combined with Sharding," *ETRI Journal*, vol. 43, no. 2, pp. 357-370, 2020. https://doi.org/10.4218/etrij.2019-0362

[5] Ahn B., "Implementation and Early Adoption of an Ethereum-based Electronic Voting System for the Prevention of Fraudulent Voting," *Sustainability*, vol. 14, no. 5, pp. 1-16, 2022. https://doi.org/10.3390/su14052917

[6] Alshehri A., Baza M., Srivastava G., Rajeh W., and et al., "Privacy-Preserving E-Voting System Supporting Score Voting Using Blockchain," *Applied Sciences*, vol. 13, no. 2, pp. 1-18, 2023, https://doi.org/10.3390/app13021096

[7] Alvi S., Nasir Uddin M., Islam L., and Ahamed S., "DVTChain: A Blockchain-Based Decentralized Mechanism to Ensure the Security of Digital Voting System Voting System," *Journal of King Saud University-Computer and Information Sciences*, vol. 34, no. 9, pp. 6855-6871, 2022. https://doi.org/10.1016/j.jksuci.2022.06.014

[8] Balaji S., Vishagasudan S., and Ezhilarasi K., "Online Voting System Using Block Chain Technology," *International Journal of Health Sciences*, vol. 6, no. S4, pp. 3978-3987, 2022. https://doi.org/10.53730/ijhs.v6nS4.10425

[9] Baothman F., Saeedi K., Aljuhani K., Alkatheri S., and et al., "Computational Intelligence Approach for Municipal Council Elections Using Blockchain," *Intelligent Automation and Soft Computing*, vol. 27, no. 3, pp. 625-639, 2021. https://doi.org/10.32604/iasc.2021.014827

[10] Baranski S., Szymanski J., Sobecki A., Gil D., and Mora H., "Practical I-Voting on Stellar Blockchain," *Applied Sciences*, vol. 10, no. 21, pp. 1-22, 2020. https://doi.org/10.3390/app10217606

[11] Bhutta M., Khwaja A., Nadeem A., Ahmad H., and et al., "A Survey on Blockchain Technology: Evolution, Architecture and Security," *IEEE Access*, vol. 9, pp. 61048-61073, 2021. https://ieeexplore.ieee.org/document/9402747

[12] Chaabane F., Ktari J., Frikha T., and Hamam H., "Low Power Blockchained E-Vote Platform for University Environment," *Future Internet*, vol. 14, no. 9, pp. 1-17, 2022. https://doi.org/10.3390/fi14090269

[13] Chafiq T., Azmi R., and Mohammed O., "Blockchain-based Electronic Voting Systems: A Case Study in Morocco," *International Journal of Intelligent Networks*, vol. 5, pp. 38-48, 2024. https://doi.org/10.1016/j.ijin.2024.01.004

[14] Chaisawat S. and Vorakulpipat C., "Towards Achieving Personal Privacy Protection and Data Security on Integrated E-Voting Model of Blockchain and Message Queue," *Security and Communication Networks*, vol. 2021, no. 1, pp. 1-14, 2021. https://doi.org/10.1155/2021/8338616

[15] Christyono B., Widjaja M., and Wicaksana A., "Go-Ethereum for Electronic Voting System Using Clique as Proof-of-Authority," *TELKOMNIKA (Telecommunication, Computing, Electronics and Control)*, vol. 19, no. 5, pp. 1565-1572, 2021. http://doi.org/10.12928/telkomnika.v19i5.20415

[16] Clarke R., McGuire L., Baza M., Rasheed A., and Alsabaan M., "Online Voting Scheme Using IBM Cloud-Based Hyperledger Fabric with Privacy-Preservation," *Applied Sciences*, vol. 13, no. 13, pp. 1-19, 2023. https://www.mdpi.com/2076-3417/13/13/7905

[17] Damavandi S. and Nogoorani S., "An Electronic Voting Scheme Based on Blockchain Technology and Zero-Knowledge Proofs," *ISeCure*, vol. 14, no. 3, pp. 123-133, 2022. https://doi.org/10.22042/isecure.2022.14.3.13

[18] Daramola O. and Thebus D., "Architecture-Centric Evaluation of Blockchain-Based Smart Contract E-Voting for National Elections," *Informatics*, vol. 7, no. 2, pp. 1-22, 2020. https://doi.org/10.3390/informatics7020016

[19] Faruk M., Alam F., Islam M., and Rahman A., "Transforming Online Voting: A Novel System Utilizing Blockchain and Biometric Verification for Enhanced Security, Privacy, and Transparency," *Cluster Computing*, vol. 27, no. 4, pp. 4015-4034, 2024. https://doi.org/10.1007/s10586-023-04261-x

[20] Germann M., "Making Votes Count with Internet Voting," *Political Behavior*, vol. 43, no. 4, pp. 1511-1533, 2021. https://doi.org/10.1007/s11109-020-09598-2

[21] Ghazinoory S., Mardani A., Maddah-Ali M., and Montazer G., "A Blockchain-Powered E-Cognocracy Model for Democratic Decision Making," *Information Systems and e-Business Management*, vol. 22, no. 2, pp. 209-246, 2024. https://doi.org/10.1007/s10257-023-00663-x

[22] Granata D., Rak M., Palmiero P., and Pastena A., "A Methodology for Vulnerability Assessment and Threat Modelling of an E-Voting Platform Based on Ethereum Blockchain," *IEEE Access*, vol. 12, pp. 176598-176614, 2024. https://ieeexplore.ieee.org/document/10750196

[23] Haldar P., Roy R., and Biswas U., "An Approach Towards Implementing Online Voting System Framework Using Blockchain Technology and Smart Contract," *Wireless Personal Communications*, vol. 138, no. 4, pp. 2699-2732, 2024. https://doi.org/10.1007/s11277-024-11622-1

[24] Hu H., Ou J., Qian B., Luo Y., and et al., "A Practical Anonymous Voting Scheme Based on Blockchain for Internet of Energy," *Security and Communication Networks*, vol. 2022, pp. 1-15, 2022. https://doi.org/10.1155/2022/4436824

[25] Huang J. and Yan X., "Blockchain Adoption Strategy in E-Commerce Supply Chain with Competing Retailers," *Journal of Systems Science and Systems Engineering*, pp. 1-27, 2025. https://doi.org/10.1007/s11518-025-5649-9

[26] Ihm Y. and Kim S., "Development of a Blockchain-based Online Secret Electronic Voting System," *IEICE Transactions on Information and Systems*, vol. E105.D, no. 8, pp. 1361-1372, 2022. https://doi.org/10.1587/transinf.2021EDK0005

[27] Jain A., Gupta N., and Gupta B., "A Survey on

Scalable Consensus Algorithms for Blockchain Technology," *Cyber Security and Applications*, vol. 3, pp. 1-16, 2025. DOI: 10.1016/j.csa.2024.100065

[28] Jayakumari B., Sheeba S., Eapen M., Anbarasi J., and et al., "E-Voting System Using Cloud-Based Hybrid Blockchain Technology," *Journal of Safety Science and Resilience*, vol. 5, no. 1, pp. 102-109, 2024. https://doi.org/10.1016/j.jnlssr.2024.01.002

[29] Kafhali S., "Blockchain-based Electronic Voting System: Significance and Requirements," *Mathematical Problems in Engineering*, vol. 2024, no. 1, pp. 1-17, 2024. https://doi.org/10.1155/2024/5591147

[30] Kraavi T. and Willemson J., "Proving Vote Correctness in the IVXV Internet Voting System," *Scientific Reports*, vol. 15, no. 1, pp. 1-14, 2025. https://doi.org/10.1038/s41598-025-16764-1

[31] Larriba A. and Lopez D., "A Solidity Implementation of TAVS," *Frontiers in Blockchain*, vol. 6, pp. 1-10, 2023. https://doi.org/10.3389/fbloc.2023.1105119

[32] Lee C., Neo H., and Teo C., "Secure E-Voting System based on Blockchain Technology," *Journal of System and Management Sciences*, vol. 12, no. 5, pp. 121-138, 2022. DOI:10.33168/JSMS.2022.0508

[33] Leune K. and Punjwani J., "Enhancing Electronic Voting with a Dual-Blockchain Architecture," *LEDGER*, vol. 6, pp. 42-57, 2021. https://doi.org/10.5195/ledger.2021.199

[34] Mahalakshmi M., Bhatnagar V., and Pandita K., "Decentralizing Voting: Block Chain Based E-Voting System Using Ethereum Smart Contracts," *in Proceedings of the International Conference on Intelligent Systems for Cybersecurity*, Gurugram, pp. 1-7, 2024. https://ieeexplore.ieee.org/document/10581197

[35] Mahanayak S., Nikhita B., and Bilgaiyan S., "Enhancing E-Voting Security with Quantum-Resistant Encryption: A Blockchain-Based Approach Utilizing Elliptic Curve Diffie-Hellman and Decentralized Storage," *SN Computer Science*, vol. 4, no. 5, pp. 1-17, 2023. https://doi.org/10.1007/s42979-023-02041-3

[36] Mahmud M., Sohan M., Reno S., Sikder M., and Hossain F., "Advancements in Scalability of Blockchain Infrastructure Through IPFS and Dual Blockchain Methodology," *The Journal of Supercomputing*, vol. 80, no. 6, pp. 8383-8405, 2024. DOI: 10.1007/s11227-023-05734-x

[37] Majumder S., Ray S., Sadhukhan D., Dasgupta M., and et al., "ECC-EXONUM-eVOTING: A Novel Signature-Based E-Voting Scheme Using Blockchain and Zero Knowledge Property," *IEEE Open Journal of the Communications Society*, vol. 5, pp. 583-598, 2023. https://ieeexplore.ieee.org/document/10376464

[38] Malkawi M., Bani Yassein M., and Bataineh A., "Blockchain Based Voting System for Jordan Parliament Elections," *International Journal of Electrical and Computer Engineering*, vol. 11, no. 5, pp. 4325-4335, 2021. https://ijece.iaescore.com/index.php/IJECE/article/view/23605

[39] Mestel D., Muller J., and Reisert P., "How Efficient are Replay Attacks against Vote Privacy? A Formal Quantitative Analysis," *in Proceedings of the IEEE 35th Computer Security Foundations Symposium*, Haifa, pp. 179-194, 2022. DOI: 10.1109/CSF54842.2022.9979167

[40] Mosley L., Pham H., Guo X., Bansal Y., and et al., "Towards a Systematic Understanding of Blockchain Governance in Proposal Voting: A Dash Case Study," *Blockchain Research and Applications*, vol. 3, no. 3, pp. 1-11, 2022. https://doi.org/10.1016/j.bcra.2022.100085

[41] Mueller R., Report on the Investigation into Russian Interference in the 2016 Presidential Election: Volume I.U.S. Department of Justice, https://www.justice.gov/archives/sco/file/1373816/download, Last Visited, 2025.

[42] Mwansa P. and Kabaso B., "Improving Election Integrity: Blockchain and Byzantine Generals Problem Theory in Vote Systems," *Electronics*, vol. 13, no. 10, pp. 1-27, 2024. https://doi.org/10.3390/electronics13101853

[43] Nakamoto S., "Bitcoin: A Peer-to-Peer Electronic Cash System," *SSRN*, pp. 1-9, 2008. https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3440802

[44] Panja S. and Roy B., "A Secure End-to-End Verifiable E-Voting System Using Blockchain and Cloud Server," *Journal of Information Security and Applications*, vol. 59, pp. 102815, 2021. https://doi.org/10.1016/j.jisa.2021.102815

[45] Panja S., Bag S., Hao F., and Roy B., "A Smart Contract System for Decentralized Borda Count Voting," *IEEE Transactions on Engineering Management*, vol. 67, no. 4, pp. 1323-1339, 2020. https://ieeexplore.ieee.org/document/9089037

[46] Qizi L. and Kamalovich B., "Blockchain Technology Usage on Intellectual Property Rights," *International Journal for the Semiotics of Law-Revue Internationale De Sémiotique Juridique*, vol. 38, pp. 363-380, 2025. https://doi.org/10.1007/s11196-024-10224-1

[47] Rajasekaran A., Azees M., and Al-Turjman F., "A Comprehensive Survey on Blockchain Technology," *Sustainable Energy Technologies and Assessments*, vol. 52, pp. 102039, 2022. https://doi.org/10.1007/s43926-024-00058-5

[48] Razali M., Jamal A., Abdullah F., Zakaria M., and et al., "E-Voting on Ethereum Blockchain," *Journal of Advanced Research in Applied Sciences*

and Engineering Technology, vol. 50, no. 2, pp. 186-194, 2025. https://doi.org/10.37934/araset.50.2.186194

[49] Rupa C., Kumari J., Gadekallu T., and Iwendi C., "Distributed-Ledger-Based Blockchain Technology for Reliable Electronic Voting System with Statistical Analysis," *Electronics*, vol. 11, no. 20, pp. 1-13, 2022. https://doi.org/10.3390/electronics11203308

[50] Saeed S., Hadi S., and Hamad A., "Iraqi Paradigm E-Voting System Based on Hyperledger Fabric Blockchain Platform," *Ingénierie Des Systèmes D Information*, vol. 27, no. 5, pp. 737-745, 2022. https://doi.org/10.18280/isi.270506

[51] Salman S., Al-Janabi N., and Sagheer A., "Valid Blockchain-Based E-Voting Using Elliptic Curve and Homomorphic Encryption," *International Journal of Interactive Mobile Technologies*, vol. 16, no. 20, pp. 79-97, 2022. https://doi.org/10.3991/ijim.v16i20.33173

[52] Sammeta N. and Parthiban L., "Blockchain-based Scalable and Secure EHR Data Sharing using Proxy Re-Encryption," *The International Arab Journal of Information Technology*, vol. 20, no. 5, pp. 702-710, 2023. https://doi.org/10.34028/iajit/20/5/2

[53] Sangraula P. and Adhikari N., "Zero Knowledge Proof on Top of Blockchain for Anonymous and Verifiable E-Voting System," *Journal of the Institution of Engineers (India) Series B*, vol. 106, pp. 1861-1872, 2025. https://doi.org/10.1007/s40031-025-01198-0

[54] Shekhar C. and Yadav R., "An Innovative and Secured Electronic Voting System Based on Elliptic Curve Signing Approach (ECDSA) and Digital Signatures," *International Journal of Information Technology*, vol. 17, no. 5, pp. 2679-2684, 2025. https://doi.org/10.1007/s41870-025-02405-3

[55] Singh A., Ganesh A., Patil R., Kumar S., and et al., "Secure Voting Website Using Ethereum and Smart Contracts," *Applied System Innovation*, vol. 6, no. 4, pp. 1-22, 2023. https://doi.org/10.3390/asi6040070

[56] Singh H. and Sinha A., "A Blockchain Framework for E-Voting," *Multimedia Tools and Applications*, vol. 83, no. 20, pp. 58875-58889, 2023. https://doi.org/10.1007/s11042-023-17837-x

[57] Singh S., Singh A., Verma S., and Dwivedi R., "Designing a Blockchain-Enabled Methodology for Secure Online Voting System," *in Proceedings of the International Conference on Intelligent Data Communication Technologies and Internet of Things*, Bengaluru, pp. 178-184, 2023. https://ieeexplore.ieee.org/document/10053410

[58] Somasekhar G., Jinka S., Kanekal C., and Marouthu A., "Digital Voting with Blockchain using Interplanetary File System and Practical Byzantine Fault Tolerance," *Engineering Technology and Applied Science Research*, vol. 14, no. 6, pp. 19009-19015, 2024. https://doi.org/10.48084/etasr.8440

[59] Song J., Moon S., and Jang J., "A Scalable Implementation of Anonymous Voting over Ethereum Blockchain," *Sensors*, vol. 21, no. 12, pp. 1-19, 2021. https://doi.org/10.3390/s21123958

[60] Spanos A. and Kantzavelou I., "EtherVote: A Secure Smart Contract-based E-Voting System," *Wireless Networks*, vol. 31, no. 2, pp. 1279-1299, 2024. https://doi.org/10.1007/s11276-024-03818-x

[61] Tang W., Yang W., Tian X., and Yuan S., "Distributed Anonymous E-Voting Method Based on Smart Contract Authentication," *Electronics*, vol. 12, no. 9, pp. 1-17, 2023. https://doi.org/10.3390/electronics12091968

[62] Tanwar S., Gupta N., Kumar P., and Hu Y., "Implementation of Blockchain-based E-Voting System," *Multimedia Tools and Applications*, vol. 83, no. 1, pp. 1449-1480, 2023. https://dl.acm.org/doi/10.1007/s11042-023-15401-1

[63] Tas R. and Tanrıover O., "A Manipulation Prevention Model for Blockchain-Based E-Voting Systems," *Security and Communication Networks*, vol. 2021, no. 1, pp. 1-16, 2021. https://doi.org/10.1155/2021/6673691

[64] Thanga Revathi S., Gayathri A., and Sathya A., "Decentralized E-Voting and Governance System Using Blockchain," *Journal of Information Systems Engineering and Management*, vol. 10, no. 2s, pp. 61-68, 2025. https://doi.org/10.52783/jisem.v10i2s.201

[65] Tripathi G., Abdul Ahad M., and Casalino G., "A Comprehensive Review of Blockchain Technology: Underlying Principles and Historical Background with Future Challenges," *Decision Analytics Journal*, vol. 9, pp. 1-21, 2023. https://doi.org/10.1016/j.dajour.2023.100344

[66] Tu S., Hsu C., and You B., "An On-Site Electronic Voting System Using Blockchain and Biometrics," *The International Arab Journal of Information Technology*, vol. 20, no. 5, pp. 808-818, 2023. https://doi.org/10.34028/iajit/20/5/13

[67] Umar B., Olaniyi O., Olajide D., and Dogo E., "Paillier Cryptosystem Based ChainNode for Secure Electronic Voting," *Frontiers in Blockchain*, vol. 5, pp. 1-11, 2022. https://doi.org/10.3389/fbloc.2022.927013

[68] Vassil K., Solvak M., Vinkel P., Trechsel A., and Alvarez R., "The Diffusion of Internet Voting. Usage Patterns of Internet Voting in Estonia Between 2005 and 2015," *Government Information Quarterly*, vol. 33, no. 3, pp. 453-459, 2016. https://doi.org/10.1016/j.giq.2016.06.007

[69] Verma S., Chandra G., and Yadav D., "LVCA: An Efficient Voting-based Consensus Algorithm in

Private Blockchain for Enhancing Data Security," *Peer-to-Peer Networking and Applications*, vol. 18, no. 2, 2025. https://doi.org/10.1007/s12083-024-01808-6

[70] Vinayachandra. and Krishna P., "Blockchain Based Cryptographic Algorithm for Data Protection in Electronic Voting System," *EAI Endorsed Transactions on Internet of Things*, vol. 11, pp. 1-12, 2025. https://doi.org/10.4108/eetiot.7680

[71] Vladucu M., Dong Z., Medina J., and Rojas-Cessa R., "E-Voting Meets Blockchain: A Survey," *IEEE Access*, vol. 11, pp. 23293-23308, 2023. https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=10061373

[72] Yao J., Yang B., Wang T., and Zhang W., "A Distributed Self-Tallying Electronic Voting System Using the Smart Contract," *Chinese Journal of Electronics*, vol. 33, no. 4, pp. 1063-1076, 2024. https://ieeexplore.ieee.org/document/10606192

[73] Zhu G., He D., An H., Luo M., and Peng C., "The Governance Technology for Blockchain Systems: A Survey," *Frontiers of Computer Science*, vol. 18, no. 2, pp. 1-17, 2023. https://doi.org/10.1007/s11704-023-3113-x

Muthu Meenakshi Ganesan received her B.Sc. in Computer Science from Madurai Kamaraj University in 2004 and her MCA degree from Alagappa University in 2007. She earned her M.Phil. degree in Computer Science from Mother Teresa Women's University in 2012. She is pursuing her Ph.D. in Computer Science at the SRM Institute of Science and Technology in Kattankulathur, Tamil Nadu, India. Her research interests include Cloud Computing, Cryptography, and Blockchain.

Sabeen Selvaraj is presently working in the Department of Computer Science, Faculty of Science and Humanities, SRM Institute of Science and Technology, Kattankulathur. He completed his Ph.D. in Computer Science at Anna University in 2012, MCA in 2002, and M.Tech. in Computer Science and Engineering in 2015. He has published several papers in international journals. He is a life member of ISTE, and his areas of interest are Data Mining, IoT, Machine Learning, Cloud Computing, Cryptography, and Blockchain.