Hindawi
Mathematical Problems in Engineering
Volume 2024, Article ID 5591147, 17 pages
https://doi.org/10.1155/2024/5591147

Research Article

Blockchain-Based Electronic Voting System: Significance and Requirements

Said El Kafhali

Faculty of Sciences and Techniques, Computer, Networks, Mobility and Modeling Laboratory (IR2M), Hassan First University of Settat, Settat, Morocco

Correspondence should be addressed to Said El Kafhali; said.elkafhali@uhp.ac.ma

Received 28 August 2023; Revised 28 January 2024; Accepted 12 February 2024; Published 24 February 2024

Academic Editor: Kumarasamy Sudhakar

In a democratic regime, voting is crucial to making collective decisions. Unfortunately, although this activity has great significance and value, little effort has been made to improve the way we vote. Paper ballots are still the most used method, although this method is relatively simple, brings many inconveniences, and represents a contradiction to the modern world and its advances. This paper mostly focuses on a review study of blockchain-based voting systems. It aims at identifying the strategies and the guidelines as well as provides a comprehensive end-to-end electronic voting system based on blockchain, with the help of cryptographic techniques such as zero-knowledge proofs to improve privacy. The novelty of this paper is that we tackle the limitations of electronic voting systems found in the literature, including cost, identity management, and scalability problems. Our purpose is to provide key elements for organizations on how to design their proper electronic voting system based on blockchain technology.

1. Introduction

Go to the polls (or voting) is one of the cornerstones of modern democracy. It enables people to actively participate in decision-making by choosing their representatives among several candidates who will be mandated to act fairly act on their behalf. Even though voting is fundamental, it has changed little over time, and we still have paper-based methods as the most common voting method. The paper-based method has its pros, such as ease of use and protection of voter privacy, which explain the persistence in adoption for decades. However, most countries still rely on paper ballots to cast votes. In a paper ballot voting system, voters prove their identities with their ID cards at a voting station to get access to the voting booth; they then get to vote for their delegate in the paper ballot, fold the ballot, and put it into the ballot box. When the election operation is over, the ballot boxes are collected and then transferred to the tallying station where they are unlocked and unloaded of ballots. The ballots are then manually examined, and the votes are counted, as shown in Figure 1.

The traditional paper ballot voting method presents some advantages, mainly the facility of use, even for illiterate people, and the secrecy of the vote, since the ballot is not linked in any way to the voter. Moreover, it has many disadvantages. The current challenges and drawbacks of traditional paper-based voting systems are the following.

(1) Cost issues: Logistics expenses that include tons of papers, transportation, polling stations, and human labor.
(2) Accessibility issues: Most paper voting systems require a trip to the polling stations. This dependency can be a struggle for people living in remote areas, citizens residing abroad, or people with disabilities.
(3) Integrity issues: Since people manage the system, it is at risk of corruption and human errors. It is strongly dependent on the efficiency and trustworthiness of people.
(4) Inefficiency issues: Running a national election is a huge project, and projects at that scale tend to go

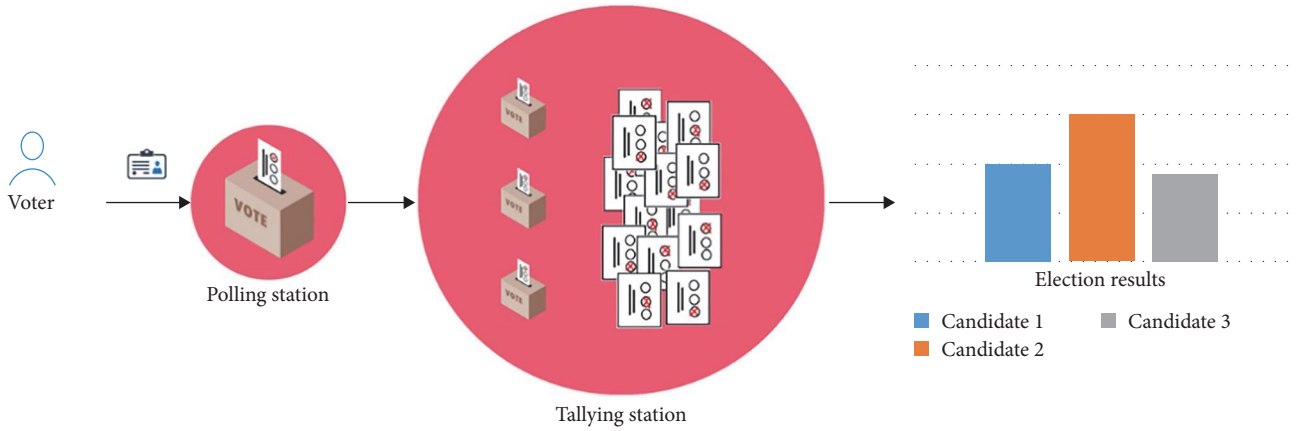

Figure 1: Traditional paper ballot voting system.

wrong. This traditional system takes a lot of time and money to implement and manage. The paper ballots are not faulted tolerant, many ballots are not valid and hence not counted, and therefore wasted.

Another voting method is electronic voting, it can be in the form of a voting machine in a process like paper voting but instead of paper, voters cast their vote via machines found in polling stations. Alternatively, it can be done online; voters cast their votes using their electronic devices. The election results are automatically counted; as a result, electronic voting is faster and more convenient than the traditional voting system. It has numerous issues. There is no guarantee that the voters' vote choice is not leaked or manipulated. Most electronic systems are black boxes and impossible to audit and are also centralized, which puts them at risk of denial of service attacks. We believe that for an electronic system online to be efficient for a large-scale election, it must possess the following properties:

(1) Security: Like any online public system, it should be immune to different cyber-attacks.
(2) Integrity: Only verified and eligible voters can vote and only once (a double spending problem in the context of blockchain), once the vote is cast, it cannot be altered.
(3) Accessibility and availability: Eligible voters should be fit to access the system remotely from anywhere and during the entire voting time.
(4) Privacy: The voter's private information should be secure and protected while voting should be done anonymously. The system of voting shall not reveal who voted for whom (ballot secrecy).
(5) Transparency: Overall system must be auditable by the public. It should not be a black box where nobody knows how to operate.
(6) End-to-end or E2E verifiable: Voters should be able to verify from end-to-end the cast (cast-as-intended) and the record (recorded-as-cast) of their vote as well as its tally (tallied-as-recorded) without being able to prove the choice to others (receipt freeness).
(7) Affordability: The cost to implement and maintain the system should be reasonable and less expensive than traditional systems.
(8) Scalability: The system must handle a large-scale election in terms of the number of participants and the response time.
(9) Coercion resistance: Voters should not be able to share or prove their vote choice with a coercer, to protect voters from blackmailing or being bought (vote buying), so the results of the election cannot be influenced unlawfully.

Blockchain systems can be the missing puzzle to solve most of these cons while maintaining maximum security. Blockchain is an open-source technology, therefore transparent and auditable [1]. This article is an extension of our previous published papers [2, 3]. In Fatrah et al.'s [2] study, we explored the feasibility of a blockchain-based voting system. Moreover, in Fatrah et al.'s [3] study, we created our first version of the system. In this paper, we are going to tackle the limitations found. In the previous articles, we assumed the existence of a user management application that the authorities use to verify the identity proof provided by the voters and also assumed that the application is secure to protect and ensure the voter data. We also found a huge scalability issue related to the cost and time, the blockchain system we designed was based on Ethereum and it turned out to be very expensive for national elections even though we had some off-chain components, also the scalability related to the time needed for the system to process all the transactions.

This article details the importance and requirements of a blockchain-based voting system capable of supporting a number of election voters simultaneously and enables voters to cast their votes at any time, from any location, and using any voting devices, including a smartphone, a SMS-based mobile phone, and web browsers. The objectives and scope of this article are highlighted as follows:

(1) We presented the challenges and drawbacks of traditional paper-based voting systems.
(2) We identified the strategies and the guidelines needed to design blockchain-based voting system.
(3) We presented a complete end-to-end electronic voting system based on blockchain.
(4) We addressed the limitations of electronic voting systems found in the literature, including cost, identity management, and scalability problems.
(5) We provided key elements via a use case design example for organizations on how to design their proper blockchain-based electronic voting system.

The remainder of this article is organized as follows. Section 2 presents the used research methodology. The state-of-the-art electronic voting is provided in Section 3. Section 4 introduces an overview of the digital identity problem of a blockchain-based voting system. Section 5 discusses the problem of scalability of blockchain-based voting systems. In Section 6, we talk about the transactions within the Ethereum blockchain. A typical blockchain-based voting system architecture is presented in Section 7. Section 8 illustrates a use case design for the blockchain-based voting system. Limitations and future research are discussed in Section 10. Finally, we conclude the article in Section 11.

2. Research Methodology

For the research methodology, we have established some questions, including where and which systems are more suitable for modern election operations? What is the traditional voting system? What is an electronic voting system? What are the added value and the key benefits of blockchain for electronic voting systems? What is a blockchain-based electronic voting system? What are the advantages, requirements, and limitations of each system? What are the challenges faced by using blockchain for electronic voting system? Following these primary defined questions, the search methodology was conducted by certain chosen key keywords, such as electronic voting, blockchain, zero-knowledge proof (ZKP), hyperledger, Ethereum blockchain, and smart contract, as well as the combination of those keywords. The search queries ran on well-known scientific databases including the Web of Science database, Scopus database, ScienceDirect database, and Google Scholar database. Following the abovementioned search process, a selection of the findings articles is used to conduct the research of this article.

3. A Survey on the Electronic Voting Systems

Electronic voting is a topic of active debate; many people, although acknowledging that paper-based voting systems are outdated and require cumbersome labor, have a hard time trusting electronic voting and the security risks it brings. However, in the era of the Internet and Web and mobile applications that boomed and became important, and now in our days, because of the recent pandemic and sanitary restrictions. COVID-19 further highlighted the need to improve the current voting system, which already changed many sectors, voting in person, and going to a polling station crowded with people is against the pandemic guidelines. In countries where the option of online voting is not provided, voters will have to choose between putting themselves in danger of exposure to the virus or staying home and not voting.

Existing voting systems are divided into two main types: traditional methods and electronic voting methods. In traditional methods, voters mark paper ballots by hand [4] or involve mechanical lever machines [5]. Within electronic voting [6], there are many types such as punched-card [7], direct recording electronic [8], optical scanning systems [9], vote recorder [10], i-voting [11], and so forth [12]. Table 1 categorizes some existing voting systems.

The continued reliance on traditional voting systems, corruption becomes far too easy, resulting in the voice of the people not being clearly heard or completely drowned out in fraud. In 1981, Chaum [13] was the first to use cryptography to secure elections, as suggested in his famous paper on anonymous communications. He described new primitives in cryptography that can be used as building blocks in different applications, including remote electronic elections. Therefore, the time he first proposed the end-to-end verifiable voting (E2E) scheme was his votegrity scheme. Other E2E schemes later emerged from Chaum's solution, such as:

(1) Neffs Markpledge [14]. Markpledge was the first E2E voting protocol that was offered alongside voice, including the development of the other E2E schemes.
(2) Ryans Pêt à Voter [15] provides an accurate election from end-to-end with an easy and familiar voter experience. It warrants a great degree of transparency while preserving the privacy of the ballot.
(3) Helios [16], a university voting system, underwent a security scan, which revealed security vulnerabilities that could affect the election outcome. This guided the development of new versions (Helios 2.0 and Helios 3.0) to fix the vulnerabilities reported in [17].
(4) STAR-Vote (A Secure, Transparent, Auditable, and Reliable Voting System) [18] implements the homomorphic tally technique [19]. Homomorphic tally implicates changes, generally additions and multiplications, to the ciphertext, which are preserved during decryption to reveal operations that were effected on the ciphertext when retrieving the changed decrypted value.
(5) Zeus [20] and Apollo [21] are using Helios to construct their voting protocol while trying to address certain security problems inherent in the Helios voting system. For example, Apollo solves cross-site scripting (XSS), cross-site tampering, clash attacks, and clickjacking with the support of voting assistants. The XSS is in the third position of the top

Table 1: Review on the types of voting systems.

Category	Description
Paper-based	Voters usually mark their voting choices by hand on the ballot paper, and then the ballots are manually numbered. This type can be categorized into on-site and remote voting. The on-site voting category refers to the process of casting a ballot by the voter in person at a ballot box. However, the remote voting category refers to the process of casting a ballot by email or other methods
Mechanical lever	A mechanical lever voting machine was first used in the 1890s to record votes in an election operation without paper. The voter indicates his choice by pressing a lever beside the desired candidate. When the voter has finished voting, he pulls the lever another time, which increments the counters corresponding to his choice by one unit, and the machine prepares the next voter
Punched-card	Punched-card is a voting method developed in the 1960s. This method used Hollerith cards (or punched cards) where voters used a stylus to punch out chads related to their desired candidate choices. Next, the Hollerith card was introduced in a ballot box. Thereafter, a card reader counted these punched cards
Direct recording electronic (DRE)	DRE voting system was first used in the early 1990s in the United States. In this voting system, vote selections are recorded directly onto computer memory. Voters use touch screens, push buttons, or dials to interact with the DRE system. DRE presents some security and reliability issues
Optical scanning systems	An optical scanning voting system permits a voter to mark his choices directly on machine-readable ballots in voting response locations. An optical scanner and computer software and hardware are used to read marked ballots and count the results
Vote recorder	Vote recorders (or ballot-marking devices) neither classify nor store ballots, but solely let the voter record the votes on ballots that are later stored and classified elsewhere. Voters make voting choices and next, generate a human-readable ballot without recording the vote electronically
I-voting	I-voting systems are called in the literature mobile voting, remote electronic voting, or online voting systems in which ballots are transmitted and recorded over the Internet. The blockchain-based electronic voting system is a type of i-voting, which is based on the Internet and uses a network that uses blockchain to vote and count votes in a national election

web frameworks, as found by the Open Web Application Security Project in 2013 [22].

(6) Follow My Vote [23] is a framework that has a secure online blockchain voting system with the ability to audit the ballot box to see the real-time democratic development.

(7) TIVI (accessible and verifiable online voting) [24] is a remote voting platform that is considered the most advanced, secure, and universally accurate online voting solution for governments. It guarantees the end-to-end integrity of the distant voting process. TIVI was designed and developed by globally known experts in election technology, cybersecurity, information security, identity management, and verifiable cryptography.

(8) Ethereum [25] is a decentralized exchange protocol that establishes a peer-to-peer network and allows users to create smart contracts. These contracts are based on an application code to verify or enforce a mutual contract.

(9) Zcash [26] is a decentralized blockchain payment system that aims to ensure the anonymity of transactions. To expedite the transactions, Zcash implements zk-SNARKS (zero-knowledge succinct noninteractive arguments of knowledge) designed in the lib-snark library.

(10) Hyperledger [27] is an open-source distributed ledger framework and enterprise-grade codebase that aims to identify and realize a cross-industry open standard platform for distributed ledgers that can transform the way business transactions [28].

Analyzing these schemes shows how hard it is to maintain both security and transparency while achieving E2E verifiability. This leads us to think of blockchains that can help to meet this requirement. However, we believe that the blockchain represents a new solution that by its nature many security concerns [29]. There have also been numerous research about the utilization of blockchain to create decentralized electronic online voting. With blockchain gaining momentum as the decentralized trust protocol, we can imagine its utilization of it as a backbone of an electronic voting system. Blockchain will ensure a trust protocol in which voters do not have to implicitly trust the credibility of the voting system and its administration. Blockchain will ensure transparency and E2E verifiability.

The electronic voting system was and remains a hot topic in research. In this section, we give a summary of these works related to the electronic voting system. In the early 1980s, Chaum [13] introduced an electronic voting system based on the Theorem of Blind Signature; the purpose was to hide voter identity by using public-key cryptography and corrupting the link between the voter and their ballot. Many papers were published in this regard [30–32], they all used some type of cryptographic techniques to achieve secrecy in an e-voting scheme. Elgamal [30] proposed the implementation of the Diffie–Hellman key distribution scheme, in a public key cryptosystem. Articles [31] and [32] use blind signatures and digital signatures, respectively, for confidentiality and the voter's digital signature during authentication.

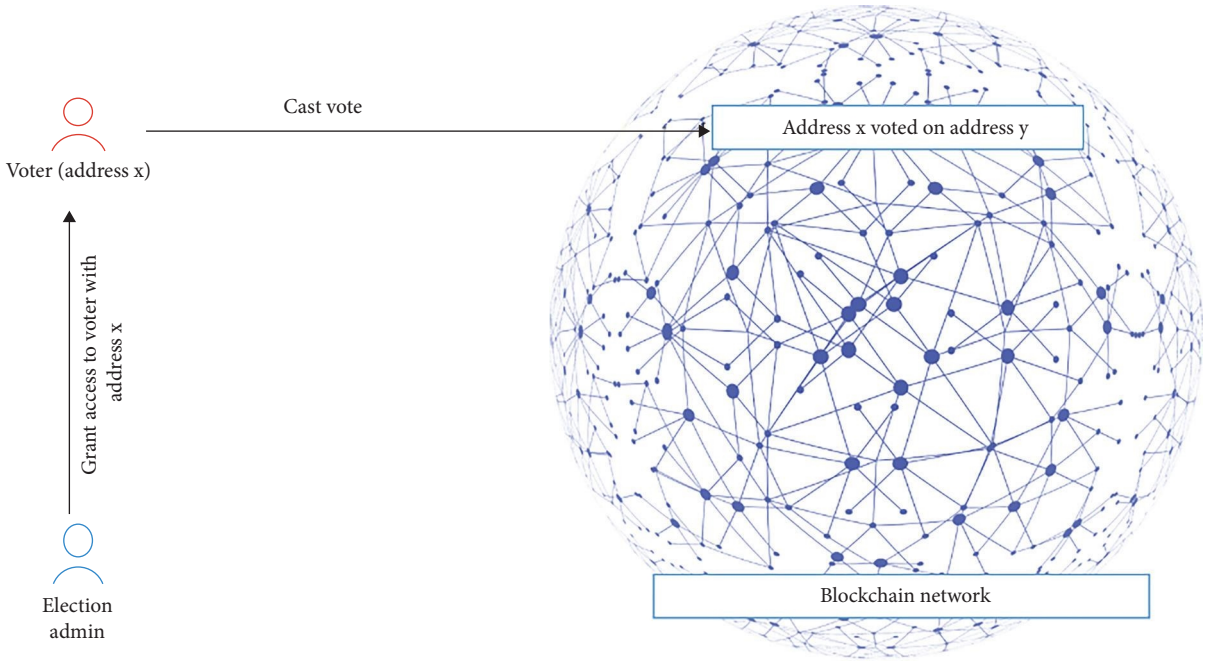

Figure 2: Voters inside the blockchain.

The first country to have a national electronic election system was Estonia in 2007 [33], the system was called i-voting and it allows citizens to cast their vote remotely via the internet, all thanks to an ID card, an electronic national identification card that enables authentication and electronic encrypted signature using both Secure Hashing Algorithms SHA1 and SHA2. The Estonian ID card also allows access to different Estonian E-services like health insurance, bank accounts, and proof of identity within the EU. The Estonia electronic system, although successful is still a black box, and it is hard to tell if they respect voter anonymity because it is not auditable by voters and it requires putting trust in the government. Norway also launched an electronic voting system project for the country council elections back in 2011, but unfortunately, the project was ceased because of some security concerns [34]. Both systems' transparency is in question, so an auditable open-source system is needed for a trusted election. Another problem is that both systems are centralized, which puts them in danger of distributed denial-of-service attacks.

Research articles have also been done on applying homomorphic encryption and the ZKP in [35–37]. In Iversen's [35] study, the author(s) proposed using interactive ZKP (IZKP) techniques to initialize voters. In Schoenmakers's [36] study, the author described a publicly verifiable secret sharing scheme with optimal running time, which can be used for an election application scheme. The purpose of ZKP is to verify the validity of the ballot without revealing the choice made by the voter. The authors in Cramer et al.'s [37] study used the factoring assumptions in their voting scheme.

Other proposals were based on blockchain technology [38, 39]. Both [38] and [39] leveraged blockchain technology to create an e-voting system. The con of these proposed systems is the lack of voter privacy by binding the voter ballot with their respective identity on the blockchain. On the blockchain, every node is represented by its public address, other voters might not know the person behind that address but the committee that allowed eligible people to participate knows their corresponding address, therefore voters are not fully anonymous (Figure 2).

4. Digital Identity Problem

4.1. Using Blockchain for Internet of Things Identity Problem. The use of Internet of Things (IoT) is rapidly growing due to the convergence of different technologies, such as embedded systems, wireless communication, and machine learning. The IoT will consist of a large number of devices connected to the internet such as machines, animals, objects, or people that are connected to a network [40]. This entails the need to manage more identities and things than what a typical identity management (ID management) system is supposed to support. It is no longer only concerned with managing people and their machines, but with the management of trillions and trillions of "things" that could be connected to the Internet. The identity of things is not limited to simple attributes such as a name and a user ID. Rather, it is a collection of many attributes to describe the identity and behavior extracted from IoT data. This makes ID management very difficult, challenging, and costly for all industrial enterprises and agencies. This growing need to secure complex transactions over the Internet requires innovative ways and technologies such as blockchain, which is a distributed database that maintains a constantly growing list of transaction records hardened against tampering and revision. The blockchain is a new way to handle responsibilities related to ID management of the IoT. Therefore, the main objective is to use

blockchain to solve all problems related to IoT identity and show how the blockchain-based approach is much superior in terms of efficiency, simplicity, cost, security, performance, and effectiveness. To attain this objective, it is necessary to fully understand how blockchain works, as well as the requirements of the ID management system of the IoT.

4.2. Identity Problem for Blockchain-Based Voting System. In a public blockchain system, the concern when it comes to identity is the binding of electronic identity inside the blockchain, which consists of a pair of public and private keys, the private key generates the public key address with the real identity of the person or people behind those pairs. The protection of private keys is a crucial concern for users.

In the context of an e-voting system, we are talking about the identity of voters inside the blockchain, the challenge is that, in a voting system, the identities of people participating must be verified to make sure that only eligible voters can participate and cast their vote. Meanwhile, the vote must be done anonymously so that no one can tell who voted on whom, and the voter should be able to track their vote without being able to prove or share their choice to avoid coercion.

To mitigate this problem, we think of an off-chain voter management system that allows authorities to verify voters before allowing them to cast their votes. In the prevoting phase, the voters must first register by providing proof of identity to authorities; this verification must be done automatically, and for this reason, we chose state-of-the-art identity verification methods based on face biometrics and liveness detection. The proof of identity will be the national electronic identity card (NEIC) that contains credentials that authorities can verify and a person's face that they can match with the person requesting the registration. We chose the face as a biometric because most if not all people can get access to the digital camera found in their phones or computers. Other biometrics might require equipment that is not found in abundance.

5. Scalability Problem

5.1. Criticizing Old Architecture. In the old architecture, voters had to proof their identity to the administration, once the voter valid, he/she receives a secret phrase, the secret phrase is going to be used as proof of knowledge to allow them to voter to cast their vote, when admin finishes the verification phase they collect all the generated secret phrases and create the arithmetic circuit for the verification smart contract. The admin also has to create the candidate contract. During the voting phase, the voter provides the secret phrase to the verified contract, if the secret phrase is valid, then the voter has to change their Ethereum address to cast their vote, and they get the ID of the transaction so that they can trace their vote. When the vote is over, the voting contract tallies the votes and returns the election results. This system is secure and straightforward forward the calculation of the cost needed to deploy it on the main Ethereum network is too expensive. Moreover, to reduce the cost, some elements with high gas consumption have to be separated from the Ethereum main network (off-chain components) hence expose to vulnerability and high risk of security breaches. The other problem that comes with the usage of Ethereum is scalability; the voting system needs to be scalable and process hundreds of thousands of voters in a limited time with very low latency (Ethereum can process roughly 10–30 transactions per second, it is unable to handle the increased load of more than 1,000 transactions per second with 667.10 GB as a block size). Election day/days high flow of transactions and system should be available, fast, and scalable to handle a national population.

5.2. Hyperledger Scalability. As cited before, Ethereum is very expensive, and it goes against the affordability requirement that we cited earlier. Especially when scaled for national election use cases. Therefore, we tried to find alternatives. When looking for a better scalable blockchain that does not necessarily need broad decentralization, we found that Hyperledger is the proper alternative.

(1) Hyperledger Fabric: The latest Hyperledger Fabric scaled fabric to 20,000 transactions per second [41]. The membership service provide for Hyperledger Fabric, which is instantiated with the identity mixer identifiers works as follows [42]:
   (a) Setup: Generation of the signature key pair of the certificate authority (CA) and the public key is made available to blockchain contributors.
   (b) Enrollment: A client (or a peer) generates a privy key and establishes an enrollment certificate (ECert) request. The CA issues an ECert in the form of an identity mixer identifier. The registration certificate evenly contains the attributes disposable to the client. The ECert is stored with the corresponding identification privy key on the client side.
   (c) Signature transactions: When a peer (or a client) needs to sign a transaction, it is necessary to generate a fresh unlinkable presentation token as follows:
      (i) signs the content of the transaction,
      (ii) proves possession of a valid Cert delivered by the CA, and
      (iii) exposes the attributes needed by the access control policy for the transaction.
(2) Verifying transaction signatures: verifying the token using the public key of the CA.
(3) Hyperledger sharding: One of the well-known techniques of scaling databases is sharding and many researchers tried to apply this method to blockchain despite the differences between legacy databases and blockchain [43]. Hyperledger uses channels to scale. The sharding rules: when people believe that sharding by geographic region is the best criterion since this information is available via the voter identity card and the authorities have an estimation of the population of each region [44].

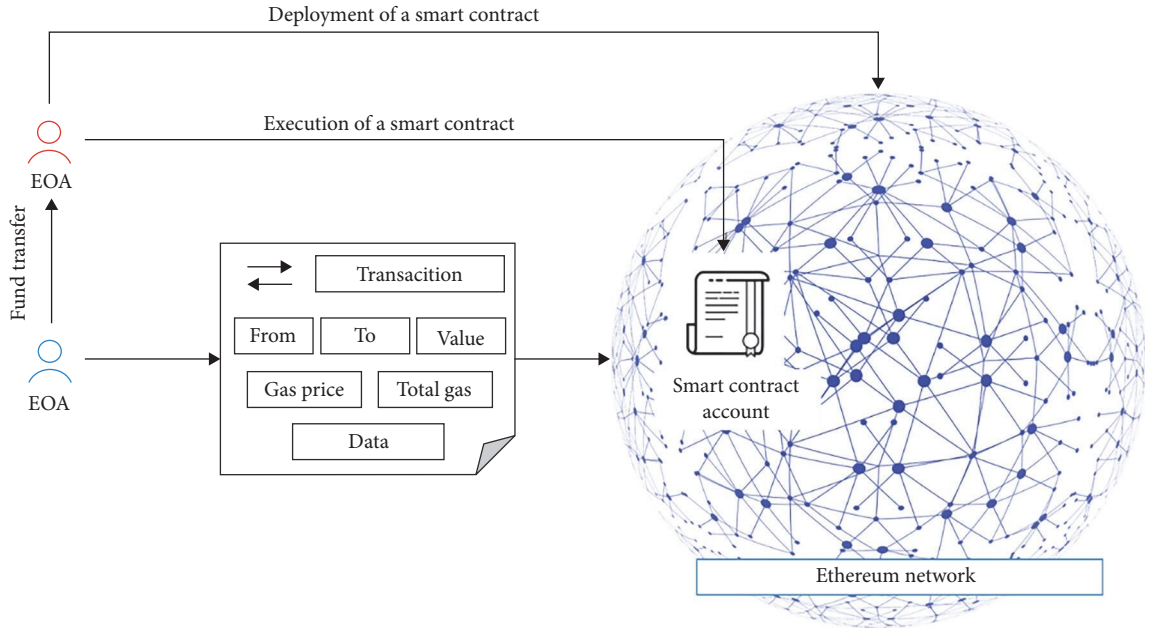

Figure 3: Transactions within the Ethereum blockchain network.

6. Transactions within the Ethereum Blockchain

In an Ethereum smart contract, agreements between contributors are written right into program code on an if-when statement. When the requirements of the if-when statements are met, the program code executes the terms of the smart contract. Contract execution begins with a transaction in which one of the contributors instructs the smart contract to do a certain task. The Ethereum node receives this transaction and then moves it onto the smart contract, indoor a virtual machine (VM). This VM is simulated in the smart contract takes the transaction as an input on a blockchain and runs it like software in which all contributors in the smart contract can watch the updates. The codes in the smart contract are distributed between all contributors, as there is no centralized authority that holds all the statement documents and controls the process. The blockchain allows various participants to agree to or do modifications to the smart contract, via their access passes. The transactions within the Ethereum blockchain network is deployed in Figure 3.

A transaction has the following parameters:

(1) From: The sender's 20-byte address (user in Ethereum network), it is the account that initiates the transaction.

(2) To: The 20-byte recipient address, it is the account that receives the transaction and it can be an externally owned account (EOA), a smart contract account, or none.

(3) Value: The total amount of Wei fund (1 ether $= 10^{18}$ Weis) to transfer to an EOA or smart contract account. Wei represents the smallest unit (denomination) of ether-the cryptocurrency coin used on the Ethereum network from which a user may make a transaction. The other way to look at it is that one wei is one quintillionth of an ether.

(4) Data/input: This is for the deployment and/or execution of contract.

(5) Gas prices: The total of Wei per gas unit.

(6) Gas limit: Every transaction has a maximum gas unit that can be spent; this is called the gas limit.

7. Blockchain-Based Voting System

A typical architecture of the blockchain-based voting system is presented in Figure 4.

Voters send their personal data to administrators for verification via their devices, which we assume to be secure. The interaction between voters and system administrators is off-chain, which means that it is not part of the blockchain system. When a voter's identity is confirmed, the administrators issue the tokens that let voters to cast their vote into the blockchain. Eligible voters receive one token in their blockchain application that acts as an electronic wallet; it is also the interface to interact with the blockchain to vote and audit. The token can only be used once and cannot be transferred or sold between wallets. When a voter wants to fill out his ballot to cast the vote, the application generates a zero-knowledge set membership proof (ZKSMP) code to prove the validity of the choice made without having to reveal it; the vote must be within a list of candidates predefined by the administrators. So, both the token and the code will be used to validate the ballot; this will eliminate the risk of Sybil attacks. All voters have to cast their ballot within a period previously configured by the administrators. Proof of Authority validators act like miners in the Bitcoin blockchain system. They validate transactions and add them to the blockchain over the voting phase.

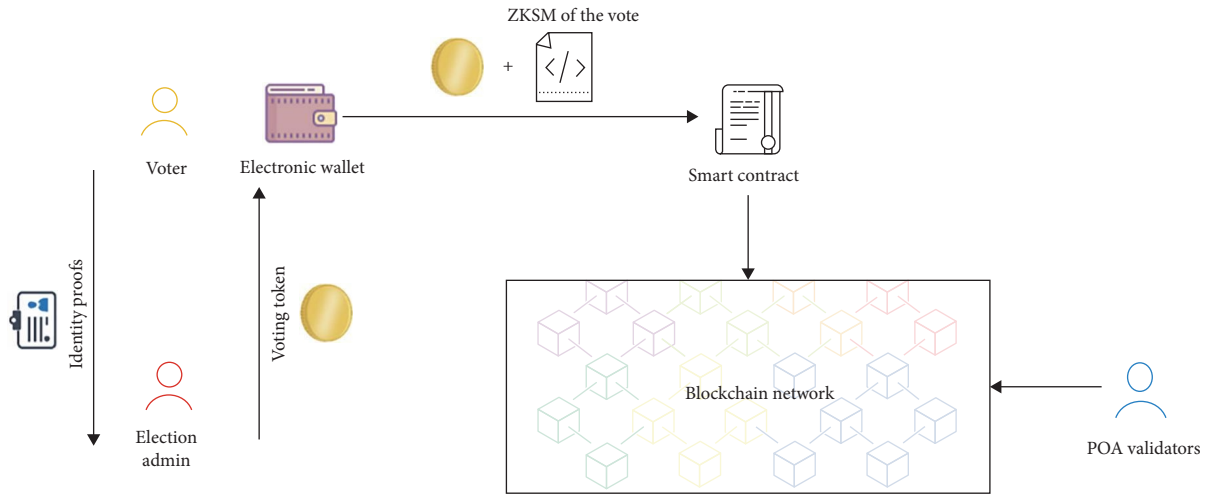

Figure 4: Typical scheme of blockchain-based voting system.

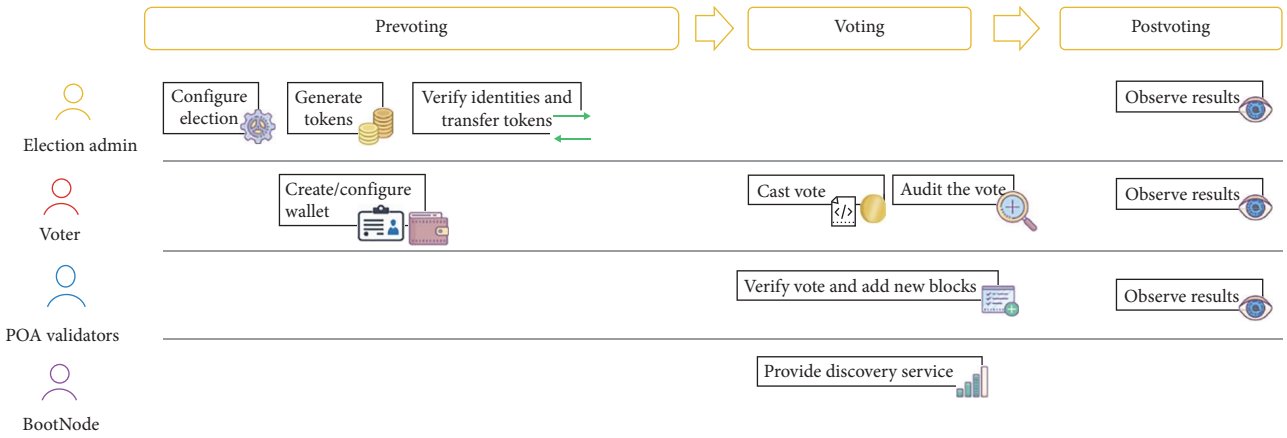

Figure 5: Workflow and roles.

BootNode (discovery service) is hosted by members with granted access to the system, and they help other nodes with discovery and enable the ease of connectivity by providing a static IP or data API endpoint that contains a set of connection information, as illustrated in Figure 5.

Once the transaction is added to the blockchain, it cannot be removed to alter, and since a voter's identity is represented by an address, voters can track their ballot and make certain it was tallied.

7.1. System Requirements. There exist some significant requirements for an electronic voting system on a national scale to be powerful, and we mention:

(1) Integrity: Eligible voters approved by the organizational committee are allowed to participate, and once votes are cast into the system, they cannot be altered or deleted.
(2) Accessibility and availability: The system must be accessible remotely throughout the electoral period.
(3) Privacy: Voter privacy should be respected; the choice of the voter should remain secret.
(4) Transparency: Everyone should be able to audit the system and know how it works, also voters can follow up on their vote and make sure it is cast and counted.
(5) Affordability: The system should be interestingly less expensive to implement and maintain than the traditional paper ballot system.

7.2. Steps of the Electronic Voting Process. Figure 6 shows the flow graph for the overall electronic voting process, from the initialization of smart contracts to the announcement of the voting results.

The overall voting process contains five essential phases which are presented as follows:

(1) Smart contracts initialization: The smart contracts are created and initialized with the list of candidates, list of voters, and all the voting rules initially established [45]. Any future modification should be made as maintained by these initial smart contracts.
(2) Voters identification: The identification of voters is necessary to assure that votes are not extorted, sold,

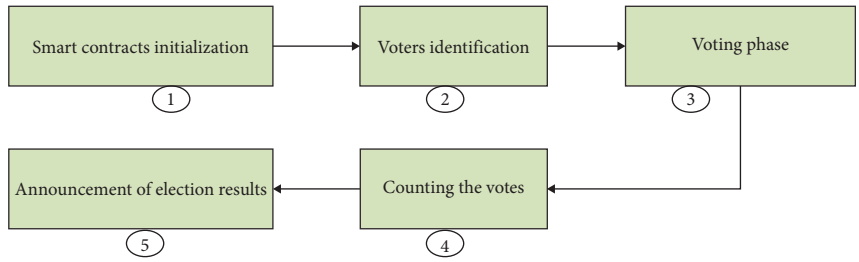

Figure 6: Flow graph for the overall electronic voting process.

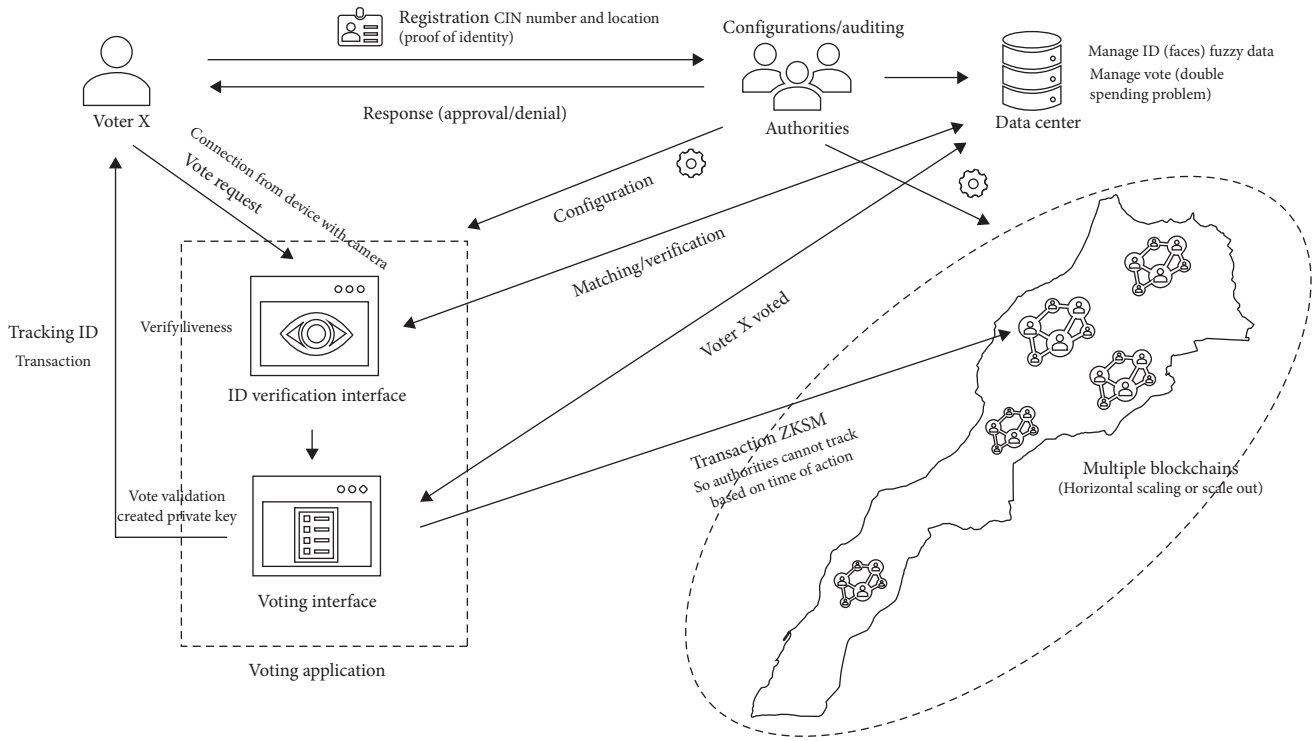

Figure 7: Blockchain-based voting system design.

or stolen. This is very important to decrease the impact of confidentiality loss and to guarantee that the person voting is who they say they are. The voter can identify himself due to some authentication mechanisms such as a scan copy of an ID document (Passport or ID card) [46], phone number [47], biometric authentication [48], and validity of credentials (public/private key) [49].

(3) Voting phase: Over this phase, the voter chooses candidates according to the voting rules initially established in the smart contracts. The ballot is then hashed using a hash function or encrypted using an encryption algorithm such as ZKP [50], SHA-256 [51], homomorphism encryption [52], Blind Signature and Ring Signature [53], and Ethereum-specific Hash function [54]. After that, the hashed or encrypted ballot is however added to the blockchain.

(4) Counting the votes: In this phase, some audits take place to guarantee that no fraudulence has been committed [55]. To do this, when the close of the election is declared, it becomes not possible to add votes. If the counting of the votes occurs in parallel with the voting phase, it is required that the present count is not observable to anyone to avoid influencing other voters who have not yet voted.

(5) Announcement of election results: Finally, the voting results are communicated and made available to all via a confident and secure channel.

8. Proposed System Design Use Case

8.1. System Design Description. Figure 7 presents a blockchain-based voting system design solution. The solution is for what kind of elections (government elections in which only one candidate can represent one party). We suppose that people (voters) have some IT (phone/computer) knowledge to be able to use the system. The voter requests registrations from the authorities. Upon verification by the authorities, the voter

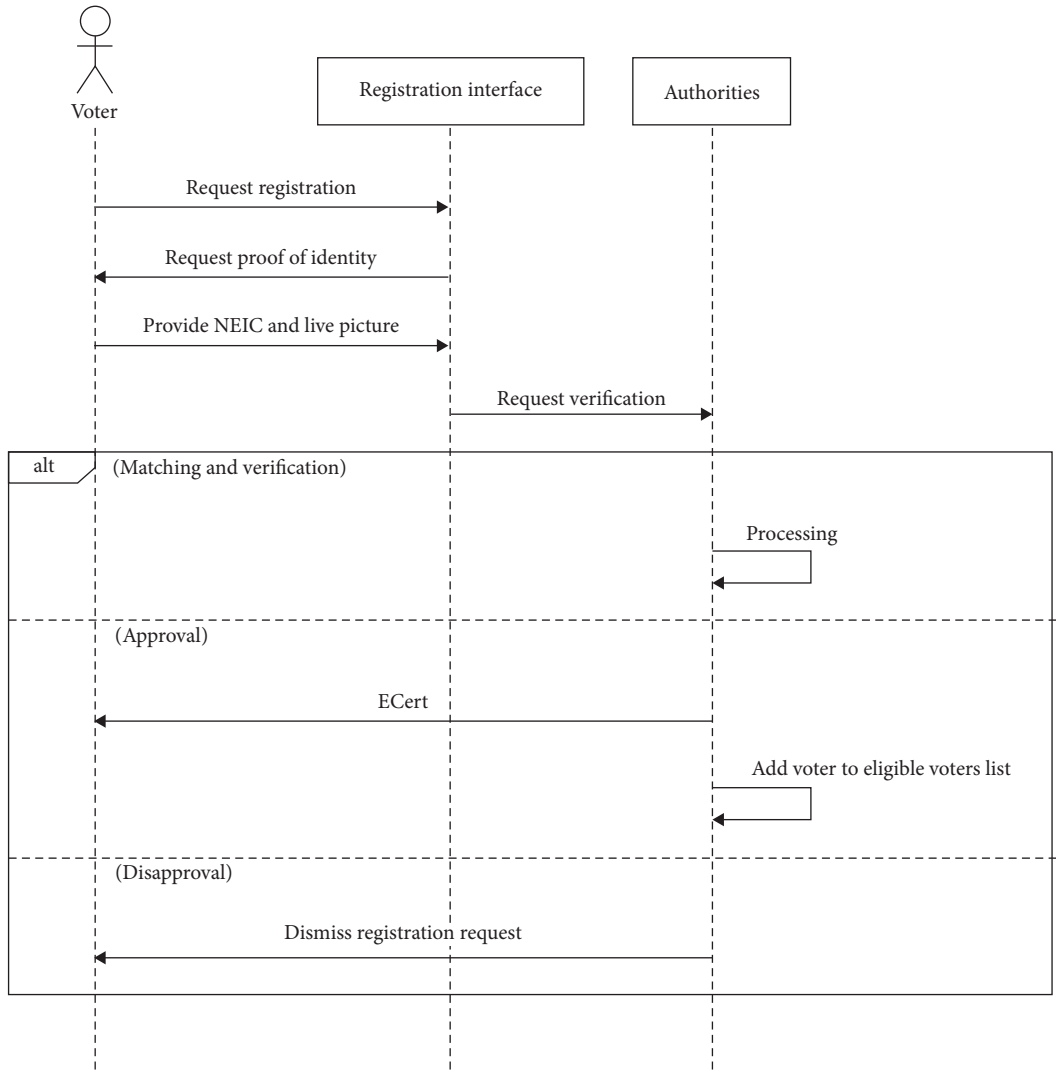

Figure 8: Registration of a voter.

receives approval from the authorities. The voter communicates with the voting application to request the Ecert and vote via the ID verification interface. Then, the voting application communicates with the Data Center to verify the legibility of the voter. In the case of acceptance, the voter selects a candidate and validates their choice via the voting interface. This later communicates with the Data Center to register the voting action and casts voting with the distributed ledger. After that, the voting application provides the tracking ID to the voter. Finally, the distributed ledger informs the voter.

8.2. User Authentication and Vote Casting. Two diagrams are presented in the following to demonstrate the sequence of operations of a voter for authentication and vote casting. The sequence diagram presented in Figure 8 shows the sequence of operations for a voter to be registered with authorities. The voter requests registrations from the registration interface. This later requests the voter's proof of identity. The voter provides the NEIC and life picture as proof of identity. Upon successful verification at the registration interface from authorities, the voter is added to the eligible voter's list.

The sequence diagram presented in Figure 9 shows the sequence of the authentication and voting operations.

Upon successful acceptance of the voter by the authorities and added to the eligible voter's list, the voter communicates with the voting interface to request the Ecert and voting. Then, the voting interface provides the candidate's list or rejects the voting request. In the case of voting acceptance, the voter selects a candidate and validates their choice. The voting interface registers the voting action with the authorities and casts voting with the distributed ledger. Finally, the distributed ledger informs the voter.

8.3. Cryptographic Tokens. Cryptographic tokens represent some particular digital asset of utility, they are issued by an organization and built on top of a blockchain, and they are alternatives to cryptocurrencies for certain tasks like representing an amount number of loyalty points, or rights to execute some actions. Ethereum is a blockchain platform that facilitates the creation of tokens thanks to smart contract. The majority of issued tokens are ERC-20, which is a technical standard for implementing tokens with smart

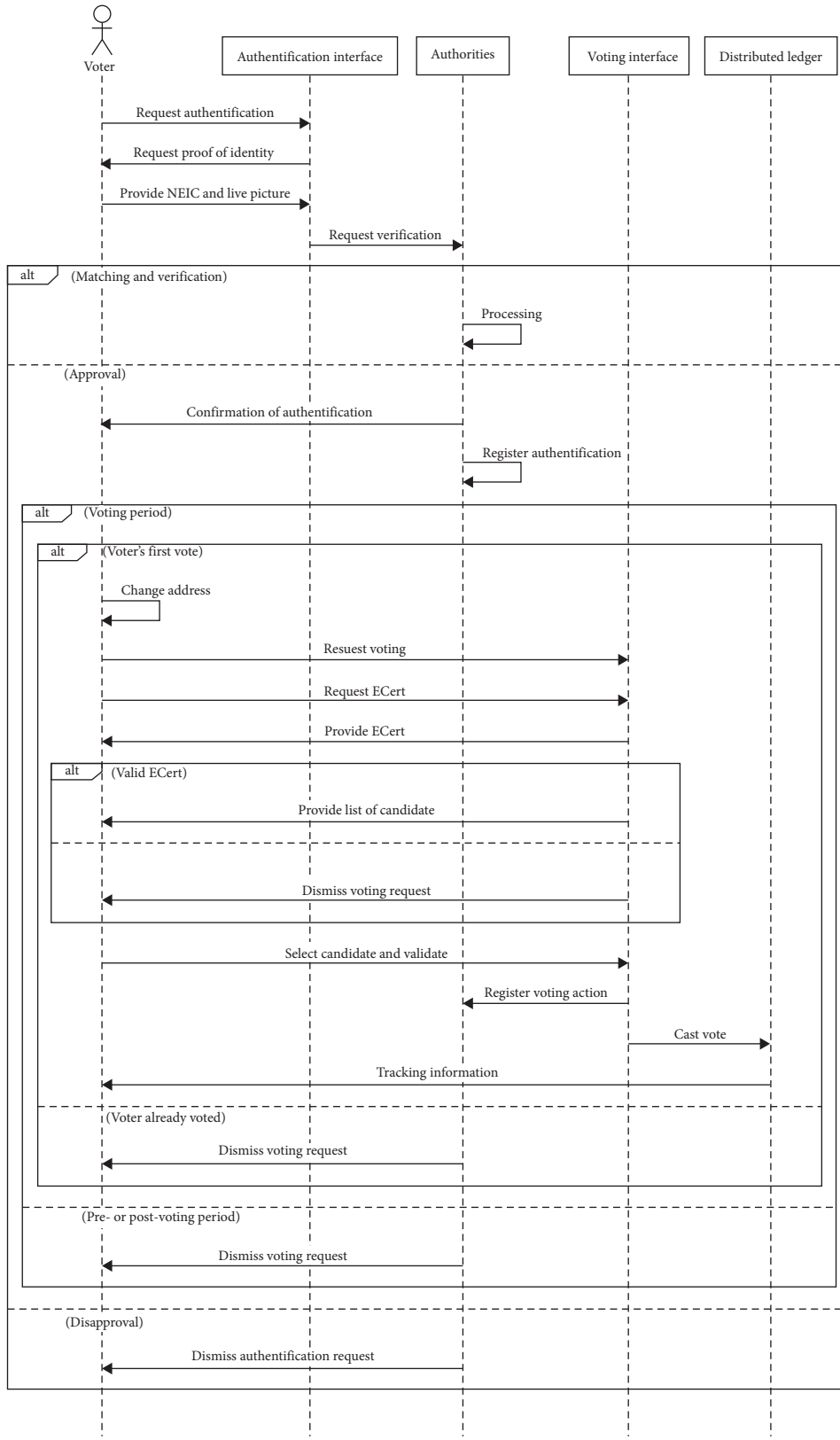

Figure 9: Authentication and casting vote.

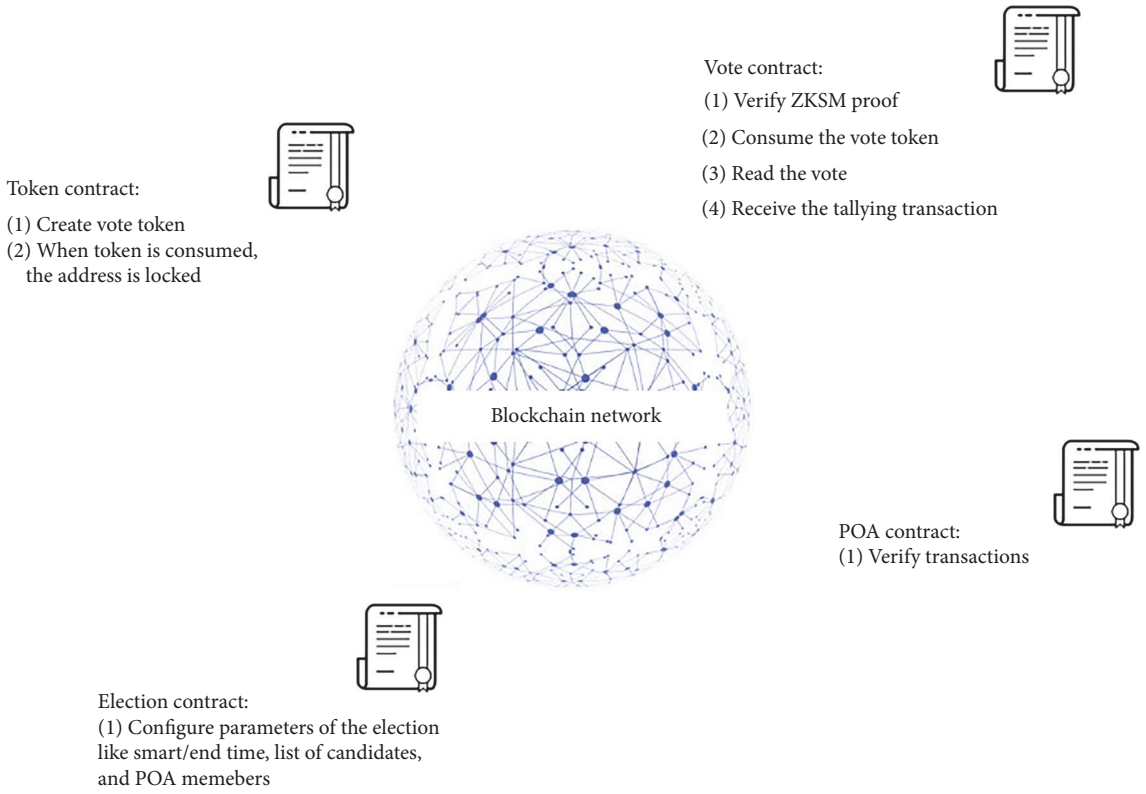

Figure 10: Main smart contracts of the system.

contract on Ethereum. In the proposed system, the token of voting will let participants to vote while disable the double-voting (same participant voting more than once), it will be delivered by the election committee and then distributed among the eligible voters. The voting tokens give the voters permission to vote and it must be consumed once they cast their votes, it also limit voters to sale or transfer it between accounts.

8.4. Anonymity and Verifiability Paradigm. In this part, we are going to discuss how we are going to protect the vote choice from being visible while also verifying the validity of the vote. To do so, we are going to rely on ZKP and more specifically the zero-knowledge-set-membership since the voter is going to be choosing among a list of candidates or parties. Two aspects of privacy can be done in Hyperledger Fabric using ZKPs. The first is the anonymous client authentication with Identity Mixer, and the second is the privacy-preserving exchange of assets with Zero-Knowledge Asset Transfer. In our case, we only want the transactions to be anonymous while keeping the values (how much each candidate acquired of vote tokens) visible and auditable.

More specifically for a blockchain-based voting system, it is the difficulty of proving that ballot contains a reliable vote choice that be included in the election candidates list without revealing those choices to the public, this will enhance voter privacy and anonymity. The ZKP is frequently used in many voting systems to demonstrate that the declaration is indeed what it claims without revealing any further details regarding the declaration itself [56]. The voter should convince the national authority of the validity of their ballot by showing that it contains only one legitimate candidate, without revealing their vote.

9. Implementing Blockchain in the Electronic Voting System

9.1. Technical Aspects. The system logic is composed of different smart contracts (Figure 10), these contracts have different functionalities.

The token contract is based on the ERC20 token, an Ethereum standard for fungible token, i.e., indistinguishable tokens. ERC20 tokens can be used as a voting permit. They were previously used in the Proof of Stake protocol to allow voting on finalization of blocks in the Casper protocol [57]. The voting token main goal is to solve the double-vote problem by only allowing eligible voters to participate, i.e., voting token allows its holder to vote, but the same token should not be used to vote again. While the blockchain keeps track of transactions history, the Ethereum virtual machine does not allow smart contracts to read state of past blocks. Accordingly, the voting token must remember its balances at a certain point in time. We cloned the MiniMe token contract [58] that uses the basic concepts of checkpoints that allow to retrieve the historical balance of the token using binary search on the array of checkpoints (Figure 11).

MiniMe token contract has been used extensively by many projects in the Ethereum space because it comes with many other features such as create and destroy tokens,

```
/// @dev `getValueAt` retrieves the number of tokens at a given block number
/// @param checkpoints The history of values being queried
/// @param _block The block number to retrieve the value at
/// @return The number of tokens being queried
function getValueAt(Checkpoint[] storage checkpoints, uint _block
) constant internal returns (uint) {
    if (checkpoints.length == 0) return 0;

    // Shortcut for the actual value
    if (_block >= checkpoints[checkpoints.length-1].fromBlock)
        return checkpoints[checkpoints.length-1].value;
    if (_block < checkpoints[0].fromBlock) return 0;

    // Binary search of the value in the array
    uint min = 0;
    uint max = checkpoints.length-1;
    while (max > min) {
        uint mid = (max + min + 1)/ 2;
        if (checkpoints[mid].fromBlock<=_block) {
            min = mid;
        } else {
            max = mid-1;
        }
    }
    return checkpoints[min].value;
}
```

Figure 11: MiniMe retrieving balance history from checkpoint array.

```
/// @notice Generates `_amount` tokens that are assigned to `_owner`
/// @param _owner The address that will be assigned the new tokens
/// @param _amount The quantity of tokens generated
/// @return True if the tokens are generated correctly
function generateTokens(address _owner, uint _amount
) public onlyController returns (bool) {
    .....
}


/// @notice Burns `_amount` tokens from `_owner`
/// @param _owner The address that will lose the tokens
/// @param _amount The quantity of tokens to burn
/// @return True if the tokens are burned correctly
function destroyTokens(address _owner, uint _amount
) onlyController public returns (bool) {
    .....
}

/// @notice Enables token holders to transfer their tokens freely if true
/// @param _transfersEnabled True if transfers are allowed in the clone
function enableTransfers(bool _transfersEnabled) public onlyController {
    .....
}
```

Figure 12: MiniMe main functions.

black and white listing addresses, and stop (freeze) and start transfer (Figure 12).

The purpose of using MiniMe contract is to get many features such as create and destroy tokens, black and white listing addresses, and stop (freeze) and start transfer:

function generate Tokens (address _holder, uint _value) only Controller

function destroy Tokens (address _holder, uint _value) only Controller

function enable Transfers (bool _transfersEnabled) only Controller

Another key contract is the vote contract in which we implement a ZKSMP validator of the proof. It was based on Fabrice Boudot paper [59] and the code from GitHub [60] based on the Byzantium precompiles. The validation of the membership proof calls for a precompiled contract written in the Golang Ethereum native language, because otherwise, it will be too computationally expensive to run on the Ethereum virtual machine:

Table 2: Comparison of traditional voting, electronic voting, and blockchain-based e-voting.

Voting systems	Advantages	Disadvantages
Traditional voting	-Presents high familiarity among voters.	-Not easy to access for voters from remote areas. -Costly. -Vulnerable to tampering of voting results or electoral fraud. -Prone to voter threat and intimidation. -Subject to human error.
Electronic voting	-Reusable for various elections. -Cheaper in long.	-Prone to hacks. -Less transparent than the blockchain-based e-voting system. -Less reliable. -Prone to rejection by technology-agnostic users.
Blockchain-based e-voting	-Potentiality extra secure, more transparent, and private. -Offers verifiable, secure, and auditable options in the electoral process. -Scalable to a large-scale election in terms of the number of participants and the response time. -Presents immutable and fixed records. -Faster voting ballot count.	-Prone to rejection by technology-agnostic users.

function validate (uint lower, uint upper, bytes commitment, bytes proof) view returns (bool)

The contracts in the blockchain-based voting system were coded in solidity and were deployed and tested on the Ethereum test-network, the contracts implementation description with different tools (truffle, metamask, etc.) that were used can be found on github [61]. The code is just a proof-of-concept project and therefore cannot be used in production.

9.2. Implementation Tools. To implement a blockchain-based electronic voting system, the following tools can be used:

(1) Go-Ethereum: Go-Ethereum (aka Geth) is a tool to implement Ethereum blockchain built using Go programming language to run smart contracts and decentralized applications. Go-Ethereum has a decentralized machine based on consensus algorithms such as Proof of Work [62], Proof of Activit [63], or Proof of Stake [64].

(2) Ganache is an isolated personal blockchain for testing smart contracts and developing distributed applications on the Ethereum blockchain [65].

(3) Truffle is a development environment that includes a collection of implements for developing decentralized applications in the Ethereum blockchain [66].

(4) MetaMask is a browser-based wallet cryptocurrency software used to interact with the Ethereum network. It is used to manage transactions, keys, and user accounts in Ethereum blockchain networks [67].

(5) Hyperledger Avalon is a popular Hyperledger development tool [68] that addresses major issues for blockchain-based electronic voting system such as confidentiality, integrity, and scalability by incorporating ZKP, trusted execution environments [69], and multiparty compute [70]. There are other tools for Hyperledger development such as Caliper, Cello, Explorer, Cactus, and so forth [71].

10. Discussion, Limitations, and Future Work

Traditional electronic voting systems habitually rely on centralized systems, which can give occasion for vulnerability issues such as electoral fraud or tampering the voting results. Blockchain provides a good solution to these vulnerabilities and issues related to other electronic voting systems and traditional voting systems. It can create a transparent and tamper-proof platform for different entities to conduct e-voting operations. Blockchain-based solutions offer verifiable, secure, and auditable options in the electoral process via the integration of protocols and cryptographic techniques. Moreover, blockchain-based electronic voting can optimize the time and cost related to other voting systems.

A comparison of traditional, electronic, and blockchain-based e-voting systems is summarized and presented in Table 2.

Although blockchain technology's success has become conspicuous, many users still do not understand it. Hence, it restricts its power to emerge and be used in different areas other than cryptocurrencies. Binding the physical and digital identities of its users is another issue facing blockchain technology. Indeed, blockchain technology cannot manage the users' identities outside the blockchain which calls for a third party to do the management work.

However, this search has some limitations and issues that offer opportunities for future work. First, the issue of accessibility of blockchain-based e-voting to all eligible voters. This should be more important when considering voters who are not accustomed to accessing the Internet, people with restricted access to new technology, or people with disabilities. Moreover, the designed electronic voting system should be more accessible and friendly for acceptance by all categories of voters. Second, is the issue of voter registration and authentication. In various blockchain-based e-voting

systems, this issue is discussed. It would be interesting to discuss biometrics, the IoT, and other secure and practical ways of voter registration and authentication in blockchain-based e-voting systems. Third, we have the issue of scalability of the e-voting system. A blockchain-based e-voting system with a small number of voters is less costly than a system with a large number of voters, which results in a longer transaction confirmation time. We have discussed this point in the article, for example, by using Hyperledger sharding that divides the whole network of a blockchain into various smaller networks to avoid longer transaction confirmation time, but scalability always remains an important key element to explore in the cost analysis of blockchain-based e-voting systems. Finally, it would also be of interest to explore another important issue such as interoperability with other existing systems, equal access to the system by all categories of voters, and the trust of the voters compared to the traditional and other electronic voting systems.

11. Conclusion

In this article, first, we presented some important research schemes and solutions in the field of electronic voting. Then, we presented the techniques that they used to address different security issues and also discussed the emergence of blockchain solutions. However, we presented the significance and requirements of the blockchain-based e-voting system. The main objective is to present more transparency in the electoral process system, ensure voters' privacy, and authorize anyone to audit the electoral system. Therefore, it will increase the number of voting participants and confidence among the people. The system is based on blockchain technology, which brings all its security features. Therefore, we presented all requirements processes, leading to more reliable, cost-effective, and strong results in the implementation and management of the electronic voting system and the improvement of security levels and voter confidence. We also included some limitations that will be addressed in future research papers.

Data Availability

No underlying data were collected or produced in this study.

Conflicts of Interest

The author declares that there is no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] M.-V. Vladucu, Z. Dong, J. Medina, and R. Rojas-Cessa, "E-voting meets blockchain: a survey," *IEEE Access*, vol. 11, pp. 23293–23308, 2023.

[2] A. Fatrah, S. El Kafhali, A. Haqiq, and K. Salah, "Proof of concept blockchain-based voting system," in *Proceedings of the 4th International Conference on Big Data and Internet of Things*, pp. 1–5, Association for Computing Machinery, 2019, October.

[3] A. Fatrah, S. El Kafhali, K. Salah, and A. Haqiq, "Transparent blockchain-based voting system: guide to massive deployments," in *Proceedings of the International Conference on Advanced Intelligent Systems and Informatics 2020. AISI 2020*, A. E. Hassanien, A. Slowik, V. Snášel, H. El-Deeb, and F. M. Tolba, Eds., vol. 1261 of *Advances in Intelligent Systems and Computing*, pp. 237–246, Springer, Cham, 2021.

[4] R. Krimmer and M. Volkamer, "Bits or paper? Comparing remote electronic voting to postal voting," pp. 225–232, 2005, In EGOV (Workshops and Posters).

[5] D. W. Jones, "The evaluation of voting technology," in *Secure Electronic Voting*, D. A. Gritzalis, Ed., vol. 7 of *Advances in Information Security*, pp. 3–16, Springer, Boston, 2003.

[6] R. Krimmer and J. B. I Esteve, *Electronic Voting. In Routledge Handbook of Election Law*, pp. 60–72, Routledge, 2022.

[7] B. Vignesh, P. P. Sricharan, S. Shankrith Chokkalingam, J. Bhuvana, and B. Bharathi, "E-biometric voting machine," in *Futuristic Communication and Network Technologies. VICFCNT 2020*, A. Sivasubramanian, P. N. Shastry, and P. C. Hong, Eds., vol. 792 of *Lecture Notes in Electrical Engineering*, pp. 505–516, Springer, Singapore, 2022.

[8] M. Dunn and L. Merkle, "Overview of software security issues in direct-recording electronic voting machines," in *Proceedings of the ICCWS. 2018 13th International Conference on Cyber Warfare and Security*, pp. 8-9, Academic Conferences & Publishing International Ltd (ACPIL), Washington, DC, USA, 2018, March.

[9] T. Antonyan, S. Davtyan, S. Kentros et al., "State-wide elections, optical scan voting systems, and the pursuit of integrity," *IEEE Transactions on Information Forensics and Security*, vol. 4, no. 4, pp. 597–610, 2009.

[10] A. W. Appel, R. A. DeMillo, and P. B. Stark, "Ballot-marking devices cannot ensure the will of the voters," *Election Law Journal: Rules, Politics, and Policy*, vol. 19, no. 3, pp. 432–450, 2020.

[11] S. J. Turnbull-Dugarte and D. Devine, "Support for digitising the ballot box: a systematic review of i-voting pilots and a conjoint experiment," *Electoral Studies*, vol. 86, Article ID 102679, 2023.

[12] M. H. Berenjestanaki, H. R. Barzegar, N. El Ioini, and C. Pahl, "Blockchain-based e-voting systems: a technology review," *Electronics*, vol. 13, no. 1, Article ID 17, 2024.

[13] D. L. Chaum, "Untraceable electronic mail, return addresses, and digital pseudonyms," *Communications of the ACM*, vol. 24, no. 2, pp. 84–90, 1981.

[14] C. A. Neff, "Practical high certainty intent verification for encrypted votes," 2004.

[15] P. Y. A. Ryan, D. Bismark, J. Heather, S. Schneider, and Z. Xia, "Prêt à voter: a voter-verifiable voting system," *IEEE Transactions on Information Forensics and Security*, vol. 4, no. 4, pp. 662–673, 2009.

[16] B. Adida, "Helios: web-based open-audit voting," *USENIX Security Symposium*, vol. 17, pp. 335–348, 2008.

[17] S. Estehghari and Y. Desmedt, "Exploiting the client vulnerabilities in Internet e-voting systems: hacking helios 2.0 as an example," in *Proceedings of the 2010 international conference on Electronic voting technology/workshop on trustworthy elections*, pp. 1–9, Association for Computing Machinery, August 2010.

[18] S. Bell, J. Benaloh, M. D. Byrne et al., "STAR-Vote: a secure, transparent, auditable, and reliable voting system," *USENIX*

Journal of Election Technology and Systems (JETS), vol. 1, no. 1, pp. 18–37, 2013.

[19] J. D. Cohen and M. J. Fischer, "A robust and verifiable cryptographically secure election scheme," in *26th Annual Symposium on Foundations of Computer Science (sfcs 1985)*, pp. 372–382, IEEE, Portland, OR, USA, October 1985.

[20] G. Tsoukalas, K. Papadimitriou, P. Louridas, and P. Tsanakas, "From helios to zeus," *USENIX Journal of Election Technology and Systems (JETS)*, vol. 1, no. 1, pp. 1–17, 2013.

[21] D. Gaweł, M. Kosarzecki, P. L. Vora, H. Wu, and F. Zagórski, "Apollo-end-to-end verifiable Internet voting with recovery from vote manipulation," in *Electronic Voting. E-Vote-ID 2016*, R. Krimmer, Ed., vol. 10141 of *Lecture Notes in Computer Science*, pp. 125–143, Springer, Cham, 2017.

[22] D. Wichers, *Owasp Top-10 2013*, OWASP Foundation, 2013.

[23] F. M. Vote, "The secure mobile voting platform of the future-follow my vote," 2020, https://followmyvote.com/.

[24] TIVI, "Tivi–verificable voting: accesssible, anytime, anwhere," Tech. Rep, 2017, https://tivi.io.

[25] G. Wood, "Ethereum: a secure decentralised generalised transaction ledger," *Ethereum Project Yellow Paper*, vol. 151, pp. 1–32, 2014.

[26] D. E. Hopwood, S. Bowe, T. Hornby, and N. Wilcox, *Zcash protocol specification*, GitHub, San Francisco, CA, USA, 2020, https://github.com/zcash/zips/blob/main/protocol/protocol.pdf.

[27] V. Dhillon, D. Metcalf, and M. Hooper, "The hyperledger project," in *Blockchain Enabled Applications*, pp. 139–149, Apress, Berkeley, CA, 2017.

[28] S. Aggarwal and N. Kumar, "Hyperledger," *Advances in Computers*, vol. 121, pp. 323–343, Elsevier, 2021.

[29] S. Nakamoto, "Bitcoin: a peer-to-peer electronic cash system," *Decentralized Business Review*, vol. 21260, 2008.

[30] T. Elgamal, "A public key cryptosystem and a signature scheme based on discrete logarithms," *IEEE Transactions on Information Theory*, vol. 31, no. 4, pp. 469–472, 1985.

[31] S. Ibrahim, M. Kamat, M. Salleh, and S. R. A. Aziz, "Secure e-voting with blind signature," in *4th National Conference of Telecommunication Technology, 2003. NCTT. 2003 Proceedings*, pp. 193–197, IEEE, January 2003.

[32] J. K. Jan, Y. Y. Chen, and Y. Lin, "The design of protocol for e-voting on the Internet," in *Proceedings IEEE 35th Annual 2001 International Carnahan Conference on Security Technology (Cat. No.01CH37186)*, pp. 180–189, IEEE, London, UK, October 2001.

[33] P. Ehin, M. Solvak, J. Willemson, and P. Vinkel, "Internet voting in Estonia 2005–2019: evidence from eleven elections," *Government Information Quarterly*, vol. 39, no. 4, Article ID 101718, 2022.

[34] Ministry of Local Government and Modernisation, "Internet voting pilot to be discontinued," 2014, https://www.regjeringen.no/en/aktuelt/Internetvoting-pilot-to-bediscontinued/id764300/.

[35] K. R. Iversen, "The application of cryptographic zero-knowledge techniques in computerized secret ballot election schemes," *Doktor ingenir-avhandling*, vol. 15, 1991.

[36] B. Schoenmakers, "A simple publicly verifiable secret sharing scheme and its application to electronic voting," in *Advances in Cryptology—CRYPTO' 99. CRYPTO 1999*, M. Wiener, Ed., vol. 1666 of *Lecture Notes in Computer Science*, pp. 148–164, Springer, Berlin, Heidelberg, 1999.

[37] R. Cramer, R. Gennaro, and B. Schoenmakers, "A secure and optimally efficient multi-authority election scheme," *European Transactions on Telecommunications*, vol. 8, no. 5, pp. 481–490, 1997.

[38] A. B. Ayed, "A conceptual secure blockchain-based electronic voting system," *International Journal of Network Security & Its Applications (IJNSA)*, vol. 9, no. 3, pp. 1–9, 2017.

[39] F. Hjálmarsson, G. K. Hreiðarsson, M. Hamdaqa, and G. Hjálmtýsson, "Blockchain-based e-voting system," in *2018 IEEE 11th International Conference on Cloud Computing (CLOUD)*, pp. 983–986, IEEE, San Francisco, CA, USA, July 2018.

[40] S. El Kafhali, C. Chahir, M. Hanini, and K. Salah, "Architecture to manage Internet of Things data using blockchain and fog computing," in *Proceedings of the 4th International Conference on Big Data and Internet of Things*, pp. 1–8, Association for Computing Machinery, October 2019.

[41] C. Gorenflo, S. Lee, L. Golab, and S. Keshav, "FastFabric: scaling hyperledger fabric to 20,000 transactions per second," *International Journal of Network Management*, vol. 30, no. 5, Article ID e2099, 2020.

[42] "Hyperledger Membership Service Provider (MSP) implementation with identity mixer," 2023, https://hyperledger-fabric.readthedocs.io/en/release-1.1/idemix.html.

[43] H. Dang, T. T. A. Dinh, D. Loghin, E.-C. Chang, Q. Lin, and B. C. Ooi, "Towards scaling blockchain systems via sharding," in *Proceedings of the 2019 International Conference on Management of Data*, pp. 123–140, Association for Computing Machinery, June 2019.

[44] E. Androulaki, C. Cachin, A. De Caro, and E. Kokoris-Kogias, "Channels: horizontal scaling and confidentiality on permissioned blockchains," in *Computer Security. ESORICS 2018*, J. Lopez, J. Zhou, and M. Soriano, Eds., vol. 11098 of *Lecture Notes in Computer Science*, pp. 111–131, Springer, Cham, 2018.

[45] X. Yang, X. Yi, S. Nepal, A. Kelarev, and F. Han, "Blockchain voting: publicly verifiable online voting protocol without trusted tallying authorities," *Future Generation Computer Systems*, vol. 112, pp. 859–874, 2020.

[46] M. Chaieb and S. Yousfi, "LOKI vote: a blockchain-based coercion resistant e-voting protocol," in *Information Systems. EMCIS 2020*, M. Themistocleous, M. Papadaki, and M. M. Kamal, Eds., vol. 402 of *Lecture Notes in Business Information Processing*, pp. 151–168, Springer, Cham, 2020.

[47] D. Khoury, E. F. Kfoury, A. Kassem, and H. Harb, "Decentralized voting platform based on Ethereum blockchain," in *2018 IEEE International Multidisciplinary Conference on Engineering Technology (IMCET)*, pp. 1–6, IEEE, Beirut, Lebanon, November 2018.

[48] K. M. Khan, J. Arshad, and M. M. Khan, "Secure digital voting system based on blockchain technology," *International Journal of Electronic Government Research (IJEGR)*, vol. 14, no. 1, pp. 53–62, 2018.

[49] H. Yi, "Securing e-voting based on blockchain in P2P network," *EURASIP Journal on Wireless Communications and Networking*, vol. 2019, no. 1, Article ID 137, 2019.

[50] T. Dimitriou, "Efficient, coercion-free and universally verifiable blockchain-based voting," *Computer Networks*, vol. 174, Article ID 107234, 2020.

[51] B. Shahzad and J. Crowcroft, "Trustworthy electronic voting using adjusted blockchain technology," *IEEE Access*, vol. 7, pp. 24477–24488, 2019.

[52] S. Zhang, L. Wang, and H. Xiong, "Chaintegrity: blockchain-enabled large-scale e-voting system with robustness and universal verifiability," *International Journal of Information Security*, vol. 19, no. 3, pp. 323–341, 2020.

[53] A. M. Larriba, A. Cerdà i Cucó, J. M. Sempere, and D. López, "Distributed trust, a blockchain election scheme," *Informatica*, vol. 32, no. 2, pp. 321–355, 2021.

[54] P. McCorry, S. F. Shahandashti, and F. Hao, "A smart contract for boardroom voting with maximum voter privacy," in *Financial Cryptography and Data Security: 21st International Conference, FC 2017*, pp. 357–375, Association for Computing Machinery, Sliema, Malta, April 2017.

[55] G. Srivastava, A. D. Dwivedi, and R. Singh, "Crypto-democracy: a decentralized voting scheme using blockchain technology," *Proceedings of the 15th International Joint Conference on e-Business and Telecommunications*, vol. 2, pp. 508–513, 2018.

[56] A. Benabdallah, A. Audras, L. Coudert, N. El Madhoun, and M. Badra, "Analysis of blockchain solutions for e-voting: a systematic literature review," *IEEE Access*, vol. 10, pp. 70746–70759, 2022.

[57] V. Buterin and V. Griffith, "Casper the friendly finality gadget," arXiv preprint arXiv, 2017.

[58] B. Jordi, "MiniMe token," 2016, https://github.com/Giveth/minime.

[59] F. Boudot, "Efficient proofs that a committed number lies in an interval," in *Advances in Cryptology—EUROCRYPT 2000. EUROCRYPT 2000*, B. Preneel, Ed., vol. 1807 of *Lecture Notes in Computer Science*, pp. 431–444, Springer, Berlin, Heidelberg, 2000.

[60] M. Eduardo and R. Peter, "Zero knowledge range proof," 2017, https://github.com/bgrieder/zkrangeproof.

[61] A. Fatrah, "Blockchain-based voting system," 2019, https://github.com/aiichaa/votingSystem.

[62] S. Zhang and J.-H. Lee, "Analysis of the main consensus protocols of blockchain," *ICT Express*, vol. 6, no. 2, pp. 93–97, 2020.

[63] I. Bentov, C. Lee, A. Mizrahi, and M. Rosenfeld, "Proof of activity: extending bitcoin's proof of work via proof of stake [Extended Abstract]y," *ACM SIGMETRICS Performance Evaluation Review*, vol. 42, no. 3, pp. 34–37, 2014.

[64] S. King and S. Nadal, "Ppcoin: peer-to-peer crypto-currency with proof-of-stake," vol. 19, no. 1, pp. 1–6, 2012, Self-Published Paper.

[65] S. Khan, A. Arshad, G. Mushtaq, A. Khalique, and T. Husein, "Implementation of decentralized blockchain e-voting," *EAI Endorsed Transactions on Smart Cities*, vol. 4, no. 10, Article ID 164859, 2020.

[66] V. Anilkumar, J. A. Joji, A. Afzal, and R. Sheik, "Blockchain simulation and development platforms: survey, issues and challenges," in *2019 International Conference on Intelligent Computing and Control Systems (ICCS)*, pp. 935–939, IEEE, Madurai, India, 2019, May.

[67] W. M. Lee, "Using the MetaMask crypto-wallet," in *Beginning Ethereum Smart Contracts Programming: With Examples in Python, Solidity, and JavaScript*, pp. 111–144, Apress, Berkeley, CA, 2023.

[68] S. Van Hijfte, "Hyperledger and DAGs," in *Blockchain Platforms*, Synthesis Lectures on Computer Science, pp. 191–207, Springer, Cham, 2020.

[69] G. Ayoade, V. Karande, L. Khan, and K. Hamlen, "Decentralized IoT data management using blockchain and trusted execution environment," in *2018 IEEE International Conference on Information Reuse and Integration (IRI)*, pp. 15–22, IEEE, Salt Lake City, UT, USA, 2018, July.

[70] H. Zhong, Y. Sang, Y. Zhang, and Z. Xi, "Secure multi-party computation on blockchain: an overview," in *Parallel Architectures, Algorithms and Programming. PAAP 2019*, H. Shen and Y. Sang, Eds., vol. 1163 of *Communications in Computer and Information Science*, pp. 452–460, Springer, Singapore, 2020.

[71] S. Sah, B. Surendiran, R. Dhanalakshmi, and N. Arulmurugaselvi, "A survey on hyperledger frameworks, tools, and applications," in *Internet of Things, Artificial Intelligence and Blockchain Technology*, R. Kumar, Y. Wang, T. Poongodi, and A. L. Imoize, Eds., pp. 25–43, Springer, Cham, 2021.