Blockchain based Voting System

[1] Abhishek Gupta,
Dept. Of CS, KIET

Group of Institutions, Ghaziabad, India,

abhishek.2125cs1185@kiet.edu

[2] Abhas Chaudhari,
Dept. Of CS, KIET

Group of Institutions, Ghaziabad, India,

abhas.2125cs1016@kiet.edu

[3] Ayushi Chauhan,
Dept. Of CS, KIET

Group of Institutions, Ghaziabad, India,

ayushi.2125cs1043@kiet.edu

[4] Arambh Trayambak,
Dept. Of CS, KIET

Group of Institutions, Ghaziabad, India,

arambh.2125cs1076@kiet.edu

[5] Shreela Pareek,
Dept. Of CS, KIET

Group of Institutions, Ghaziabad, India,

shreelapareek@gmail.com

***Abstract*— Elections are fundamental to democracy, yet traditional voting systems often face issues such as fraud, inefficiency, and lack of transparency. Blockchain's immutability and decentralization offer a secure solution for internet voting. This paper introduces a blockchain-based voting system developed on Ethereum, utilizing technologies like OTP verification, token generation and validation, identity verification, and real-time vote calculation. By addressing the limitations of traditional voting and internet voting, this system demonstrates the potential of blockchain to enhance electoral security, transparency, and efficiency, paving the way for trust in democratic processes.**

***Keywords*— Blockchain, Ethereum, Immutability, Decentralization, OTP verification**

INTRODUCTION

Elections are the cornerstone of democratic governance, yet traditional voting systems face persistent challenges such as fraud, inefficiency, and a lack of transparency. These issues undermine the credibility of results and erode public trust. Historically, voting methods have evolved from hand-raising in ancient assemblies to paper ballots and electronic voting machines. Despite technological advancements, reliance on physical polling stations imposes logistical and financial burdens that discourage voter participation.

Internet-based voting has emerged as a potential solution, with countries like Estonia pioneering its use in parliamentary elections since 2007. These systems reduce costs and increase turnout but are vulnerable to cyberattacks such as DDoS, spoofing, and malware, which leave minimal traces and are difficult to mitigate. Addressing these vulnerabilities requires innovative approaches to ensure security and trustworthiness.

Blockchain technology provides a transformative solution to these challenges by leveraging decentralization, transparency, and immutability. By integrating advanced features like OTP verification, token creation and validation, identity verification, and real-time vote calculation, blockchain ensures data integrity and security. The Ethereum platform, with its smart contract capabilities, further enhances this system, enabling self-tallying and eliminating the need for intermediaries. However, designing such a system demands careful consideration of potential risks, including double voting due to consensus vulnerabilities and misuse of smart contracts for vote- buying.

This paper explores the development of a blockchain-based voting system on Ethereum, addressing these challenges and proposing robust solutions. By overcoming the limitations of traditional and internet-based voting systems, this research aims to establish a secure, transparent, and efficient framework, demonstrating blockchain's potential to revolutionize electoral processes.

PROJECT OBJECTIVE

The main objective of this project is to develop a blockchain-based voting system using Ethereum that addresses the challenges of traditional and internet- based voting methods. The specific objectives are:

1. Enhance Electoral Security: Improve the security of the voting process by utilizing Ethereum's blockchain features, ensuring tamper-proof votes and reducing vulnerabilities to fraud and cyberattacks.

2. Ensure Voter Privacy: Guarantee voter anonymity and secure storage of vote data through encryption, protecting sensitive information from unauthorized access.

3. Prevent Fraud and Double Voting: Prevent multiple votes from a single voter by using smart contracts and unique voting tokens, ensuring each voter casts only one vote.

4. Improve Transparency and Efficiency:

Increase transparency by enabling real-time vote tallying and public verification of results, while reducing the time and cost of manual counting.

5. Ensure Scalability: Design the system to handle elections of varying sizes, from local to national, using Ethereum's scalability features to maintain performance and low costs.

6. Explore Blockchain's Potential: Demonstrate how blockchain can transform electoral processes by offering a secure, transparent, and efficient alternative to current systems.

These objectives aim to create a secure, transparent, and efficient voting system that enhances trust in the electoral process.

LITRATURE REVIEW

This section provides an overview of previous research and developments in voting systems, with a focus on blockchain technology's potential to address current limitations in election processes. It explores the evolution of voting systems, their associated challenges, and how blockchain, particularly through the Ethereum platform, offers an innovative solution to enhance electoral integrity.

2.1 Challenges of Traditional Voting Systems

Traditional voting methods, such as paper ballots and electronic voting machines, have faced several significant challenges:

- Fraud: The vulnerability of physical ballots and voting machines to tampering, either by individuals or through systematic errors, undermines election integrity.

- Inefficiency: Traditional systems often involve slow processes such as manual vote counting, long waiting times, and logistical burdens associated with setting up physical polling stations, which can discourage voter participation.

- Lack of Transparency: Voters and observers frequently have limited visibility into how votes are counted and the overall election process, reducing trust in the outcome and creating opportunities for manipulation.

These issues underscore the need for more efficient and transparent solutions that can uphold the credibility of elections.

2.2 Internet Voting Systems

Internet voting has gained attention as a potential solution to many of the challenges faced by traditional voting methods, especially regarding cost, convenience, and accessibility:

- Benefits: Online voting systems offer significant advantages, including reduced operational costs, increased voter turnout, and the ability to vote remotely, making it easier for people to participate.

- Security Risks: Despite its benefits, internet voting is not without its risks. Cybersecurity threats such as Distributed Denial of Service (DDoS) attacks, malware, and identity theft pose significant challenges, as they are often difficult to detect and mitigate. Furthermore, ensuring the privacy and integrity of votes cast via the internet remains a complex problem.

These concerns highlight the necessity of finding a more secure and reliable method of online voting, especially as more governments consider transitioning to internet-based systems.

2.3 Blockchain Technology

Blockchain technology offers a promising approach to address many of the security and transparency concerns associated with both traditional and internet- based voting systems:

- Immutability: Once a vote is recorded on a blockchain, it cannot be altered or deleted, ensuring that election results remain tamper- proof.

- Decentralization: Blockchain operates on a distributed network of nodes, removing the need for a central authority that could potentially manipulate the results. This decentralization also makes the system more resilient to failures and attacks.

- Transparency: Blockchain enables a transparent voting process where every participant can verify the integrity of the election data. Voters and auditors can independently confirm vote counts in real-time, enhancing trust in the election outcome.

The combination of decentralization and immutability makes blockchain an ideal candidate for securing the voting process and ensuring that it remains transparent and trustworthy.

2.4 Ethereum and Smart Contracts in Voting

Ethereum, a leading blockchain platform, is particularly well-suited for developing blockchain-based voting systems due to its smart contract capabilities:

- Smart Contracts: Smart contracts are self-executing contracts with the terms of the agreement directly written into code. In the context of voting, smart contracts can automatically handle vote casting, validation, and result calculation, reducing human intervention and potential errors.

- Tokenization: Ethereum allows the creation of unique tokens that represent votes, ensuring that each token is valid and cannot be

duplicated or tampered with. These tokens can be used to guarantee that voters only cast one vote each, thus preventing double voting.

Smart contracts and tokenization are key components in building a secure, automated, and transparent voting system that can handle large-scale elections without the need for intermediaries.

SYSTEM DESIGN

This section outlines the design and architecture of a blockchain-based voting system developed on the Ethereum platform. The system leverages the features of blockchain technology, such as immutability, decentralization, and transparency, to create a secure, efficient, and transparent voting system. This design addresses the limitations of traditional and internet- based voting systems while ensuring voter privacy, security, and trust in election outcomes.

3.1 Blockchain Selection

The Ethereum blockchain is chosen due to its robust smart contract functionality and extensive adoption. Ethereum's decentralized nature ensures there is no central point of failure, making it highly resistant to tampering or fraud. Its ability to support smart contracts allows for the automation of voting processes, ensuring transparency, accountability, and efficiency without the need for intermediaries. The Ethereum network's security protocols and consensus mechanisms further ensure the integrity of the system.

3.2 System Components

The system consists of several components that work together to ensure a secure and transparent voting process:

1. Voter Authentication (OTP Verification)
   Voter authentication is crucial for ensuring that only authorized individuals can cast their vote. To achieve this, the system uses One- Time Password (OTP) verification. Each voter receives a unique OTP to authenticate their identity and initiate the voting process. This step ensures that the voting system is protected from unauthorized access and reduces the risk of impersonation.

2. Token Generation and Validation
   Once authenticated, each eligible voter is issued a unique voting token on the Ethereum blockchain. These tokens serve as the digital representation of a vote. They are generated through smart contracts, which ensure that each token is valid and tied to a specific voter. The system's blockchain ensures that tokens cannot be duplicated or manipulated, addressing the risk of double voting. The tokens are stored on the blockchain and cannot be altered after they have been cast.

3. Identity Verification
   Identity verification is achieved through a combination of cryptographic methods and blockchain records. A voter's identity is registered on the blockchain at the time of voter registration, and each vote cast is tied to the verified identity of the voter. This ensures that each person votes only once, eliminating the possibility of fraudulent multiple votes from the same individual.

4. Real-Time Vote Calculation
   A key feature of the blockchain voting system is the real-time calculation of votes. As votes are cast, they are recorded and tallied in real- time on the blockchain, ensuring that the results are immediately available for verification. This feature minimizes the time and cost associated with manual counting, reduces human error, and allows for quicker results dissemination.

3.3 Smart Contract Implementation

Smart contracts are at the heart of the voting system's automation and security. These self-executing contracts automatically enforce the rules and protocols of the election without requiring human intervention. The smart contract handles various tasks, including:

- Vote Casting: Once a voter is authenticated, the smart contract ensures that the vote is valid and records it on the blockchain.
- Vote Validation: Smart contracts validate the authenticity of the token associated with each vote, ensuring that it is not duplicated or tampered with.
- Vote Tallying: As votes are cast, the smart contract updates the vote tally in real-time, allowing the system to calculate results efficiently and accurately.

The smart contract system ensures that election rules are followed strictly and automatically, eliminating the need for intermediaries and minimizing the risk of errors or manipulation.

3.4 Ensuring Voter Privacy and Security

Ensuring voter privacy and security is paramount in any voting system. The blockchain-based system ensures privacy by encrypting voter identities and votes. Voter data is securely stored using cryptographic techniques, and the blockchain's decentralized nature ensures that no single entity controls or can access all voter information. Each vote is anonymous, and only the final tally is visible on the public blockchain, providing transparency without compromising individual voter privacy.

Additionally, advanced encryption and multi-factor authentication mechanisms are implemented to secure voter credentials, further reducing the risk of unauthorized access or hacking. This ensures that the

system is both secure and user-friendly, providing a seamless experience for voters while maintaining the integrity of the voting process.

3.5 Scalability and Performance

The system is designed to scale efficiently, handling elections of varying sizes—from local to national levels. Ethereum's scalability features, such as layer-2 solutions, allow the system to manage large numbers of votes without compromising performance. By utilizing off-chain storage and sidechains where appropriate, the system can manage high transaction volumes while maintaining low fees and fast processing times. This ensures that the system remains cost-effective and efficient even during large- scale elections.

3.6 Security Considerations

While blockchain technology enhances the security of the voting system, it is essential to consider potential vulnerabilities and implement countermeasures to address them:

1. Double Voting Prevention
   The use of Ethereum's consensus mechanism, combined with token validation, ensures that once a vote is cast, it is immutable and cannot be replicated. The smart contract verifies the token's authenticity, preventing multiple votes from a single voter.

2. Smart Contract Security
   Smart contracts must be thoroughly tested and audited to identify potential vulnerabilities. This process ensures that there are no coding errors or loopholes that could allow malicious actors to manipulate the voting process. Secure coding practices and external audits are essential to maintaining the integrity of the smart contract.

Cybersecurity Threats

Although blockchain offers a high level of security, the system remains vulnerable to external attacks, such as Distributed Denial of Service (DDoS) attacks and phishing attempts. To mitigate these risks, the system employs multi-layered security measures, including robust firewalls, encryption, and continuous monitoring of the network to detect and prevent attacks.

PROPOSED METHODOLOGIES

The project is a voting system that leverages blockchain technology to enable transparent and immutable voting. The system consists of a front-end for user interaction, a back-end to handle the logic, a database for user management, and blockchain to record the voting process.

- Frontend Development:

Technology Stack:

HTML/CSS/JS: Used for the UI, displaying information, and handling events (such as user login, voting, etc.)

Web3.js: JavaScript library used to interact with the Ethereum blockchain.

jQuery: Used to handle DOM manipulation and AJAX requests.

Bootstrap (optional): For modal windows and styling.

- Components:

Login Page (login.js): The front-end provides a login form for users to input their credentials. Upon validation, the user is redirected to the voting page if authentication succeeds.

Voting Page (clist.js): Displays a list of candidates. Users can cast their vote by clicking on buttons that invoke the voteForCandidate function in the smart contract.

- Features:

User Authentication: Users log in by entering their username and password, which are validated on the back-end.

Display Candidates: The front-end retrieves the list of candidates from the smart contract.

Voting Interaction: Users select a candidate and submit their vote, which is sent to the blockchain.

- Backend Development:

Technology Stack:

Node.js: Used to run the back-end server. Express.js: A web framework used to handle HTTP requests.

Cookie Management: Cookies are used to track logged-in users and manage authentication.

- Components:

Authentication (/login route): Validates user credentials (username and password). A hashed password comparison is used for security.

Authentication Middleware (/auth route): Verifies the user's authentication status using cookies.

Smart Contract Interaction: The back-end interacts with the Ethereum blockchain using Web3.js to read and write data to the blockchain (for example, sending a vote).

Smart Contract Deployment: The Voting.sol contract is compiled and deployed on the Ethereum blockchain during the /info route.

- Features:

User Login: Verifies user credentials and sets an authentication cookie.

Smart Contract Integration: The server interacts with a deployed smart contract to perform voting

operations.

Redirects and Responses: Based on the user's authentication status, they are redirected to the appropriate page (either the voting page or the login page).

- Database Management:

Technology Stack:

Cookies: The system uses cookies to store the user's session and authentication token, eliminating the need for a traditional database.

In-memory Data: Candidate data is passed directly to the smart contract and stored on the blockchain, eliminating the need for storing candidate information in a relational database.

Components:

User Data: Information such as authentication status, session cookies, and user credentials are stored temporarily in cookies and managed via Express middleware.

No Traditional Database: No relational database (such as MySQL or MongoDB) is used in this project for storing votes or candidates. Instead, the blockchain is responsible for the data storage, particularly for the vote counts.

Features:

Session Management: Authentication information (like session status) is stored in cookies.

Blockchain as a Database: Candidate data and vote counts are stored in the Ethereum blockchain via the smart contract, ensuring that the data is transparent, immutable, and secure.

- Blockchain Usage:

Technology Stack:

Ethereum Blockchain: The blockchain is used to record votes in a decentralized manner, ensuring transparency and immutability.

Solidity: Smart contracts are written in Solidity to manage the election logic (candidate registration, voting, and vote counting).

Web3.js: JavaScript library to interact with the Ethereum blockchain. It allows the front-end to send transactions to the blockchain and read the contract data.

Components:

Voting Contract (Voting.sol): The smart contract handles the logic for:

Storing the list of candidates.

Counting votes for each candidate.

Validating votes to ensure users are voting for legitimate candidates.

Smart Contract Deployment: The contract is deployed on the Ethereum blockchain via Web3.js, with the contract address being used in the front-end and back-end.

Voting Mechanism: The actual voting happens on the blockchain via the voteForCandidate function. Once a vote is cast, it is recorded immutably on the blockchain.

Features:

Decentralized Voting: Votes are cast and counted in a decentralized manner on the Ethereum blockchain, ensuring that no one can tamper with the vote counts.

Transparency: Since the voting data is stored on the blockchain, it can be accessed by anyone to verify the results.

Immutability: Once a vote is recorded on the blockchain, it cannot be changed or tampered with.

Summary:

Frontend Development: HTML/CSS, JavaScript, Web3.js for interaction with the blockchain.

Backend Development: Node.js, Express.js for handling requests and interacting with the smart contract.

Database Management: Cookies for session management, and blockchain for storing votes and candidate data.

Blockchain Usage: Ethereum blockchain to store votes immutably using Solidity smart contracts.

The system combines traditional web technologies with blockchain to create a secure, transparent, and immutable voting system.

SECURITY CONSIDRATION

This section addresses the security aspects of the blockchain-based voting system, ensuring its robustness against potential vulnerabilities and threats. By leveraging Ethereum's decentralized architecture and advanced cryptographic techniques, the system aims to protect voter privacy, prevent fraud, and maintain the integrity of the election process.

4.1 Double Voting Prevention

The system implements mechanisms to prevent double voting, a critical issue in digital voting systems. By using unique voting tokens generated through smart contracts, each vote is validated and recorded on the Ethereum blockchain. The decentralized nature of blockchain ensures that once a vote is cast, it cannot be altered or duplicated. The smart contract ensures that each voter is allowed only one valid vote, reducing the risk of fraud and guaranteeing a fair voting process.

4.2 Smart Contract Security

Smart contracts, which automate the voting and validation processes, are a fundamental part of the system's security. These contracts are written in code and executed on the blockchain without human intervention. Ensuring the security of smart contracts is critical to avoid errors or manipulation. The system

undergoes extensive testing and external audits to identify vulnerabilities, ensuring that the contracts are free from flaws that could be exploited. Secure coding practices and regular code audits are essential in maintaining the integrity of the system and preventing attacks.

4.3 Protection from Cybersecurity Threats

While blockchain technology offers a high level of security, the system must still guard against potential cybersecurity threats, such as Distributed Denial of Service (DDoS) attacks and phishing. To mitigate these risks, the system integrates multiple layers of security:

- Encryption: Voter data, including identities and votes, is encrypted using state-of-the-art cryptographic methods, making it resistant to unauthorized access or tampering.
- Multi-Factor Authentication (MFA): Voter credentials are protected through multi-factor authentication, adding an extra layer of security to prevent unauthorized access.
- Continuous Monitoring: The system is continuously monitored for unusual activity, enabling quick responses to any potential attacks.

4.4 Privacy and Anonymity

Ensuring voter privacy is paramount. The blockchain- based voting system utilizes advanced encryption techniques to protect voter identities and voting choices. Voter data is stored in a decentralized manner, ensuring that no single entity has control over or access to all the information. Only the final vote tally is made public, ensuring transparency while preserving individual privacy. This maintains the trust of voters and assures them that their participation is confidential and secure.

4.5 Resilience to Network Failures

The decentralized nature of the Ethereum blockchain provides resilience against potential network failures or attacks. Unlike centralized systems, which can be taken down by targeting a single point of failure, the blockchain's distributed network ensures that the system remains operational even in the event of localized failures or attacks. This ensures continuous availability and reliability throughout the election process.

RESULT & DISSCUSSION

This section summarizes the results and their implications for the blockchain-based voting system developed on Ethereum, addressing key challenges in traditional and internet-based voting methods.

- Security and Performance

  The blockchain-based system successfully improved the security of the voting process by preventing fraud, double voting, and manipulation. Using Ethereum's decentralized architecture and smart contracts, votes were securely recorded and could not be altered once cast. OTP verification and multi-factor authentication ensured only authorized voters could participate, addressing major security concerns.

- Voter Privacy and Anonymity

  The system maintained voter privacy by encrypting identities and vote choices. Votes were cast anonymously, with only the final tally visible, ensuring transparency without compromising privacy, effectively balancing both, as intended in the project.

- Real-Time Vote Calculation and Transparency

  Smart contracts allowed for real-time vote tallying, reducing the need for manual counting and minimizing errors. The blockchain ensured immediate availability of results, fostering transparency and trust in the electoral process by enabling public verification of the vote count.

- Scalability and Efficiency

  The system demonstrated scalability, efficiently handling elections of various sizes. Ethereum's scalability features, such as layer- 2 solutions and off-chain storage, enabled the system to process large volumes of votes with low fees and fast processing times, making it cost-effective even for large-scale elections.

- Addressing Blockchain Vulnerabilities

  The system incorporated multiple layers of security, including encryption and continuous monitoring, to protect against potential cyber threats like DDoS attacks and phishing. While secure under normal conditions, the system's resilience can be further enhanced through regular audits.

- Overall Impact

The blockchain-based voting system met the project objectives, providing a secure, transparent, and efficient alternative to traditional and internet-based voting methods. It demonstrated the potential of blockchain to improve electoral processes, increasing trust and participation while reducing fraud. This system offers a viable solution for modern elections, with potential for further refinement and scalability for larger elections.

CONCLUSION

This section summarizes the results and their implications for the blockchain-based voting system developed on Ethereum, addressing key challenges in traditional and internet-based voting methods.

- Security and Performance

  The blockchain-based system successfully improved the security of the voting process by preventing fraud, double voting, and manipulation. Using Ethereum's decentralized architecture and smart contracts, votes were securely recorded and could not be altered once cast. OTP verification and multi-factor authentication ensured only authorized voters could participate, addressing major security concerns.

- Voter Privacy and Anonymity

  The system maintained voter privacy by encrypting identities and vote choices. Votes were cast anonymously, with only the final tally visible, ensuring transparency without compromising privacy, effectively balancing both, as intended in the project.

- Real-Time Vote Calculation and Transparency

  Smart contracts allowed for real-time vote tallying, reducing the need for manual counting and minimizing errors. The blockchain ensured immediate availability of results, fostering transparency and trust in the electoral process by enabling public verification of the vote count.

- Scalability and Efficiency

  The system demonstrated scalability, efficiently handling elections of various sizes. Ethereum's scalability features, such as layer- 2 solutions and off-chain storage, enabled the system to process large volumes of votes with low fees and fast processing times, making it cost-effective even for large-scale elections.

- Addressing Blockchain Vulnerabilities

  The system incorporated multiple layers of security, including encryption and continuous monitoring, to protect against potential cyber threats like DDoS attacks and phishing. While secure under normal conditions, the system's resilience can be further enhanced through regular audits.

- Overall Impact

The blockchain-based voting system met the project objectives, providing a secure, transparent, and efficient alternative to traditional and internet-based voting methods. It demonstrated the potential of blockchain to improve electoral processes, increasing trust and participation while reducing fraud. This system offers a viable solution for modern elections, with potential for further refinement and scalability for larger elections.

REFERENCES

Anandaraj & Sakthivel. (2015). Secured Electronic Voting Machine Using Biometric.

Amold, Ed.(1999). History of voting system in California.

Buterin, V. et al. (2013). Ethereum white paper

Downs, A. (1957). An Economic Theory of Democracy, Harper and Row, N.Y., 1957.

Jane Susskind. (2017). Decrypting Democracy: Incentivizing Blockchain Voting Technology for an Improved Election System, 54 San Diego L. Rev. 785

Lelia Barlow (2003). An introduction to Electronic Voting.

National Academy of Sciences (NAS), 2005. tp://www.ncsl.org/research/elections-and- campaigns/voting-equipment.aspx

R. Hanifatunnisa & B. Rahardjo. (2017). "Blockchain based e-voting recording system design," 2017 11th International Conference on Telecommunication

Systems Services and Applications (TSSA), Lombok, 2017, pp. 1-6, doi: 10.1109/TSSA.2017.8272896

Rivest, R., Shamir, A., and Tauman, Y. (2001). How to leak a secret. Advances in Cryptology ASIACRYPT 2001, 552-565.

United States Election project (2016). 2016 November General Election Turnout Rates. From http://www.electproject.org/2016g

Valimised(2019), Voter turnout in Estonia. From https://rk2019.valimised.ee/en/voting-result/voting- result-main.html

Verified Voting. (2019) Video Shows Voting Machine Malfunctioning in Mississippi. From https://www.newsy.com/stories/video-shows- mississippi-voting-machine-malfunctioning/Newsy