JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

# TRANSFORMING CYBERSECURITY: ONLINE VOTING SYSTEM USING BLOCKCHIAN TECHNOLOGY

[1]Sudeep Shetty,[2]Athmaranjan K,[3]William Narvin Fernandes, [4]Archana Chandrakanth Mahale, [5]Rahul Ramdas Chinchankar

[1]Final year B.E. Student, [2] Assistant Professor, ISE, [3]Final year B.E. Student, [4]Final year B.E. Student, [5]Final year B.E. Student
[1]Department of Information Science & Engineering,
[1]Srinivas Institute of Technology, Mangaluru, India.

***Abstract*:** Elections are the most important event in a democratic country, and voting is a basic democratic activity. In addition to being more difficult to use than computerized ballots, paper balloting is prone to mistakes and misuse. Many countries use computerized voting techniques to address this. The usage of blockchain technology as a service to develop a decentralized electronic voting system architecture is assessed in this implementation paper. The goal is to provide an open and easily accessible voting system that guarantees each and every voter in the country their independence and justice. The suggested method deploys smart contracts for every voting event using the Ethereum Virtual Machine (EVM), which serves as the Blockchain runtime environment. The purpose of this decentralized blockchain-based system is to guarantee the integrity of the voting process and promote voter engagement.

**Keywords: Blockchain, Decentralized voting system, Ethereum Virtual Machine, Smart Contract**

## I. INTRODUCTION

Voting systems ensure that the electoral process is fair, transparent, and free from any form of coercion or manipulation. They also allow citizens to elect representatives who they believe will best represent their interests and make decisions on their behalf. There are various types of voting systems, such as plurality voting, proportional representation, and ranked-choice voting, each with its own special features and advantages. Voting systems are a set of procedures and rules that are used to conduct elections in a democratic society.

The government or a central authority oversees and supervises every step of the election process in a centralized voting system. Voter verification, voter registration management, paper ballot production and distribution, and vote counting are typically included in this kind of system. Every administrative task, including controlling voting at actual polling places, when voters physically cast their ballots, and the government or central authority also counts the votes. Usually, nations with powerful central governments and well-functioning election infrastructures employ this method [8].

Voting is a crucial aspect of any democratic system, as it allows citizens to express their opinions and preferences regarding the government and its policies. It is the foundation of representative democracy, ensuring that every individual has an equal opportunity to participate in the political process and have their voice heard. By voting, citizens elect representatives who reflect their interests and values, thereby shaping the direction of governance. It also serves as the most direct form of decision-making, allowing people to influence who governs them and how policies are made. Moreover, a robust voting system promotes accountability, as elected officials are more likely to fulfill their responsibilities and promises knowing they could be voted out in future elections. Additionally, voting encourages civic engagement by motivating people to stay informed about political issues and participate more actively in public life. Overall, the voting system plays a vital role in upholding democratic principles, ensuring fair representation, fostering transparency, and empowering citizens to have a meaningful say in how they are governed.

## II. METHODOLOGY

This section outlines the methodology adopted in our proposed system. The objective is to overcome the challenges previously discussed by integrating delegated voting mechanisms, blockchain technology, and supporting disciplines such as cybersecurity, privacy, and legal frameworks.

### Design Process

The proposed voting application is designed to meet essential criteria such as privacy, voter eligibility, user-friendliness, receipt-freeness, and verifiability. The system ensures secure digital voting through a web-based interface that prevents double voting by linking voter identity to a digital wallet. The administrative interface allows efficient management of voters, constituencies, and candidates. The system guarantees equal voting rights for all citizens and promotes fair competition among candidates, all while safeguarding voter privacy. After casting a vote, a cryptographic hash is generated as a receipt, which can be tracked beyond the constituency level without compromising anonymity.

**The Voting Process**

A secure cryptographic server is employed to maintain voter anonymity. Each vote is encrypted before being stored on the blockchain. Voters log in using their credentials, and upon verification, they are presented with a list of candidates, including an option to reject all. Once a vote is cast, it is validated through a mining process by a decentralized network and then recorded on the public ledger. The use of blockchain and cryptographic hashing ensures the integrity and security of each vote. Each voter's identity is represented by a unique encrypted hash, which even the system administrator cannot access, thereby preserving voter privacy [10].

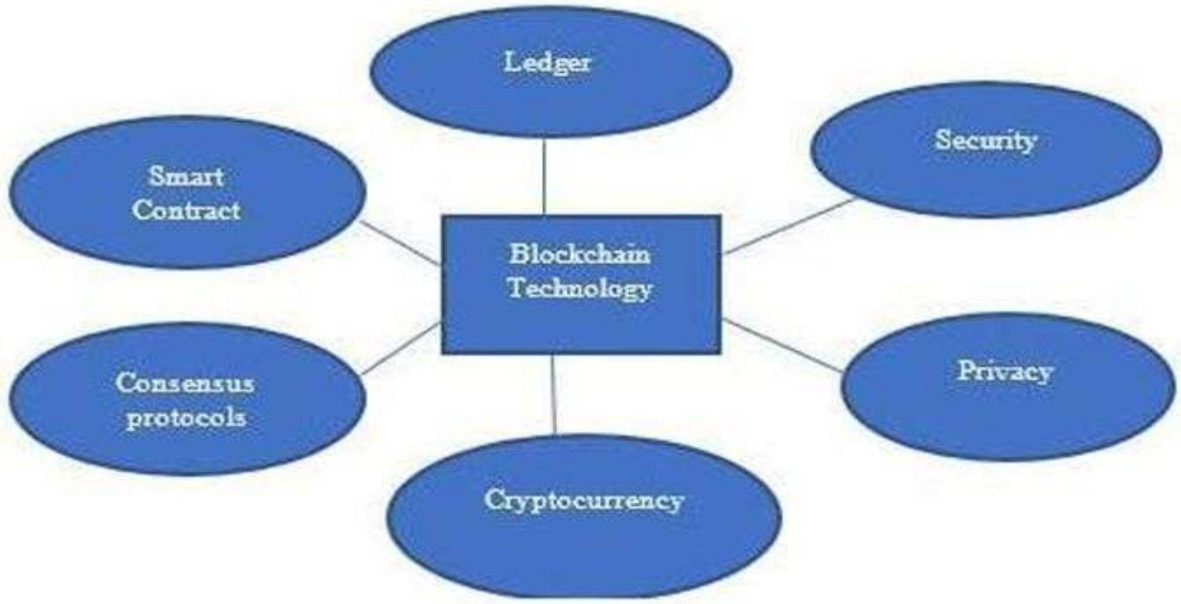

**Figure 2:** General Architecture of Blockchain Technology.

**The components on server side are:**

- **Truffle:**
  Truffle is a powerful development framework for building Ethereum-based applications using the Solidity programming language. It offers a wide range of built-in features including automated testing, client-side application development, smart contract compilation and deployment, and network management. In the proposed system, Truffle is used as the network management tool to streamline smart contract development and deployment.
- **Solidity:**
  Solidity is a high-level, contract-oriented programming language used to write smart contracts on the Ethereum blockchain. It shares similarities with JavaScript and follows an object-oriented programming approach, where contracts are structured like classes, comprising variables and functions. Solidity code is compiled into bytecode compatible with the Ethereum Virtual Machine (EVM), allowing it to run on the Ethereum blockchain.
- **Ganache:**
  Ganache is a widely-used local blockchain development tool that allows developers to simulate and test decentralized applications in a safe environment. It provides a personal Ethereum blockchain for development purposes, enabling developers to deploy contracts, test functionality, and inspect the state of the blockchain. In the context of a decentralized voting system, Ganache is used to test and validate the behavior of the system before deploying it to the live Ethereum network.
- **Node Server:**
  The proposed solution utilizes a lightweight node server, referred to as the crypto server, for cryptographic operations. This server is responsible for securely generating, storing, and managing public and private keys used for encrypting and decrypting votes. These keys are generated by the Election Commission and are essential for maintaining the confidentiality and integrity of the voting process. The crypto server is dedicated solely to cryptographic functions and key management, ensuring secure vote processing during the election.

## III. IMPLEMENTATION:

Ethereum's blockchain technology shows significant potential as a platform for electronic voting. Its smart contract functionality enables the automation of predefined actions based on the terms written into the contract, ensuring transparency and security. Ethereum supports two types of accounts: externally owned accounts (EOAs), which are controlled by users, and contract accounts (CAs), which are smart contracts deployed on the blockchain. Ether, Ethereum's native digital currency, can be held and transferred by both types of accounts. To interact with a smart contract, an EOA must first authorize it, and executing any contract function requires spending a resource called "gas." For a blockchain-based voting system to function, a web-based interface is necessary to host and facilitate the voting process. This interface forms part of a decentralized application (dApp) that can serve as an alternative to traditional voting systems. Fig. 1 illustrates the implementation details of integrating a voting system using the Ethereum blockchain.

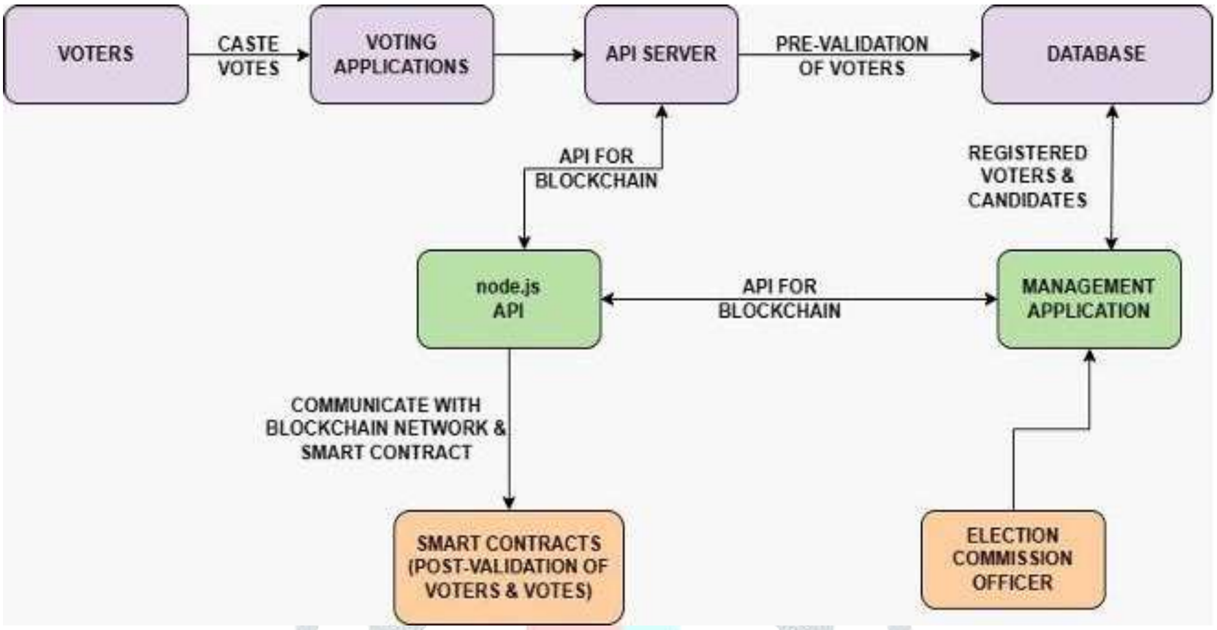
VOTERS
CASTE VOTES
VOTING APPLICATIONS
API SERVER
PRE-VALIDATION OF VOTERS
DATABASE
API FOR BLOCKCHAIN
REGISTERED VOTERS & CANDIDATES
node.js API
API FOR BLOCKCHAIN
MANAGEMENT APPLICATION
COMMUNICATE WITH BLOCKCHAIN NETWORK & SMART CONTRACT
SMART CONTRACTS (POST-VALIDATION OF VOTERS & VOTES)
ELECTION COMMISSION OFFICER

**Figure 1**: Data Flow Diagram

- **Ledger:** Blockchain operates as a distributed ledger technology, meaning that every participant in the network holds an identical copy of the record. There is no central authority or trusted third party; instead, trust is established through decentralized consensus.
- **Consensus Protocols:** Transactions must be verified by all participating nodes in the network. The process of creating a block and adding it to the ledger is decentralized and is commonly referred to as *mining*. All nodes must agree on the validity of the transactions in the newly created block before it is added to the blockchain.
- **Security:** Blockchain ensures security through the use of digital signatures and public key cryptography, which verify the authenticity and integrity of transactions and participants within the network.
- **Cryptocurrency:** A cryptocurrency is a form of digital asset that acts as a medium of exchange. It uses cryptographic techniques to secure transactions and control the creation of new units, thereby enabling secure and tamper-proof exchanges.
- **Privacy:** Blockchain can store various types of data, including sensitive information. When handling such data—such as healthcare records or citizen services—privacy regulations must be enforced to ensure proper data protection.
- **Smart Contracts:** Smart contracts are self-executing agreements with the terms directly written into code. They automatically execute and enforce obligations when predefined conditions are met. These contracts often rely on external data sources, and to ensure data integrity, cryptographic proof must be attached to prevent tampering.

**Client-SideComponents:**

Voting through Ethereum accounts is now accessible on any computer or mobile device via a user-friendly client-side interface. The interface is built using **HTML** for structure, **CSS** for styling and layout enhancements, and **React.js** for efficient data management and rendering on the client side. Communication between the client and the Ethereum blockchain is handled using **Web3.js**, a JavaScript library that facilitates interactions with smart contracts and blockchain data.One of the core components on the client side is **MetaMask**, a lightweight browser extension that functions as an Ethereum wallet. MetaMask stores users' Ether, enables them to send and receive funds, and allows interaction with decentralized applications (dApps). It is compatible with major browsers and manages users' public keys while using private keys to sign and authorize transactions securely. To date, there have been no confirmed security breaches in MetaMask resulting in stolen funds, which reflects its strong reputation for safety. MetaMask serves as a bridge between blockchain networks and web browsers, processing Web3 requests and forwarding them to the Ethereum network or a designated server.

## IV. BLOCKCHAIN TECHNOLOGIES:

Blockchain technologies A block is a part of the blockchain in which it records all the transactions and once it is completed enters into a permanent data base in the blockchain. In Blockchain, the blocks are linked one after other like a linked list. Every block consists the hash of the previous block as shown in Figure 3.

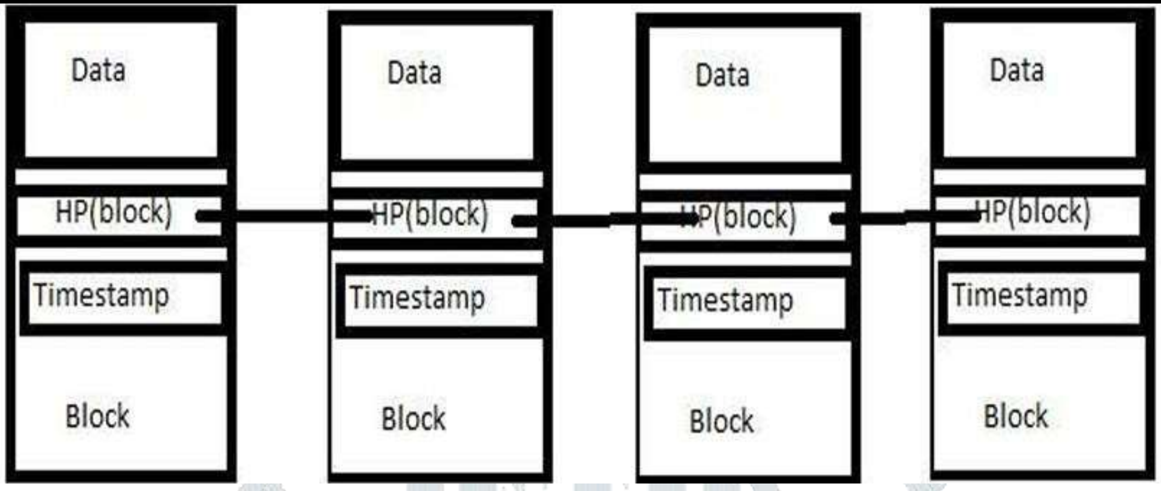

Fig 3: Blockchain as A Linked List of Blocks Connected by Hash Point-ers.

A blockchain network is made up of multiple nodes connected in a peer-to-peer structure. Users interact with the blockchain using a pair of cryptographic keys: a public key and a private key. The private key is used to sign transactions, while the public key acts as the user's identity on the network. This key pair ensures authentication, data integrity, and non-repudiation. Each node in the network verifies the validity of incoming transactions before forwarding them to other nodes. Any transaction that fails verification is automatically rejected. To maintain consistency, every blockchain network is programmed with a set of rules that define how transactions should be processed. These rules are embedded in each blockchain client to ensure that all nodes validate transactions in the same way.

**Types of Blockchain**

Blockchains are generally categorized into three types:

1. Public Blockchain
2. Private Blockchain
3. Permissioned Blockchain

In a Public Blockchain, anyone can participate in the network without needing permission, and there are no trust relationships between nodes. Once a transaction is added, it cannot be altered or removed. Common consensus algorithms used in public blockchains include Proof of Work (PoW), Proof of Stake (PoS), and Delegated Proof of Stake (DPoS).

A Private Blockchain is controlled by a single organization. Only the owner has full authority to modify data, while other nodes have restricted access. The Practical Byzantine Fault Tolerance (PBFT) algorithm is commonly used for consensus in private blockchains.

In a Permissioned Blockchain, participants must be granted access and can choose their own consensus nodes based on predefined rules. This type of blockchain is well-suited for semi-closed networks, such as those formed by a group of collaborating enterprises.

**Smart Contract Security**

Smart contracts—also known as blockchain contracts, digital contracts, or self-executing agreements—are programmable agreements that automatically execute and enforce themselves when predefined conditions are met. Once deployed on the blockchain, these contracts must comply with the fundamental rules set by the network. In some cases, smart contracts rely on external data sources, which introduces the risk of data being tampered with. To ensure the integrity of such data, cryptographic proof must be attached, verifying that the information came from a trusted source and has not been altered**.**

Contracts stored on the blockchain are both legally binding and considered "smart" due to their automated nature. These contracts are self-executing, meaning they automatically carry out the terms and conditions once predefined requirements are met, without the need for human intervention. This automation reduces the chances of errors or fraud, making them more reliable and secure. Because they are stored on the blockchain, they are also immutable and transparent, ensuring that all parties involved can trust the integrity of the contract. The combination of these features makes blockchain contracts not only efficient but also legally enforceable, as they align with the principles of traditional contract law.

**Conclusions**

Decentralized election systems present a promising alternative to traditional voting methods by enhancing transparency, security, and inclusivity through blockchain technology. These systems can empower underrepresented communities, boost voter participation, and foster greater public trust in the electoral process. However, their implementation is not without challenges. Technical complexity, scalability issues, privacy concerns, and the need for robust legal frameworks must be carefully addressed to ensure reliable and secure deployment. Despite these obstacles, the potential benefits—such as increased trust, accountability, and democratic engagement—make decentralized voting a compelling direction for future electoral reform in an evolving digital society.

## References

[1] Kosba, A., Miller, A., Shi, E., Wen, Z., & Papamanthou, C. (2016). Hawk: The blockchain model of cryptography and privacy- preserving smart contracts. In Proceedings of IEEE Symposium on Security and Privacy (S&P) (pp. 839-858).

[2] Clark, J., Essex, A., & Rainey, M. (2018). Blockchain voting systems: an analysis of security issues and their solutions. Journal of Cybersecurity, 4(1), tyx020.

[3] Rass, S., Schrajber, A., & Tauber, D. (2019). Voting with blockchain technology: An exploratory study. Government Information Quarterly, 36(4), 101389.

[4] Vora, P., & Samanta, D. (2020). A survey of blockchain-based electronic voting protocols. Journal of Ambient Intelligence and Humanized Computing, 11(7), 2765-2785.

[5] Santucci, G., & Mori, P. (2019). Blockchain and digital government: prospects for future research. Electronic Government, an International Journal, 15(2), 156-175.

[6] Kshetri, N. (2018). Blockchain's roles in meeting key supply chain management objectives. International Journal of Information Management, 39, 80-89

[7] Teague, V., & McDonald, R. (2017). Blockchain and voting systems: An analysis of security threats and solutions. Melbourne, Australia: RMIT University.

[8] Böhme, R., Christin, N., Edelman, B., & Moore, T. (2015). Bitcoin: Economics, technology, and governance. Journal of Economic Perspectives, 29(2), 213-238.

[9] Kashefizadeh, M., Kshetri, N., & Ahram, T. (2020). A blockchain-based framework for enhancing transparency and accountability in voting systems. Computers & Security, 88, 101616.

[10] Zohar, A. (2015). Bitcoin: Under the hood. Communications of the ACM, 58(9), 104-113.