A Decentralized Digital Voting System Based On Blockchain Architecture

Anirudh Bajaj (B20CS005)
Mentored by : Dr. Debasis Das, Assistant Professor, CSE Department, IIT Jodhpur

Abstract

Today, a sizable portion of the population has lost faith in their government. As a result, elections are a key component of a contemporary democracy. The problem with the existing Indian electoral system is that it is susceptible to manipulation and hence not entirely trustworthy. Therefore, in order to improve the widespread usage of voting systems, cryptographic techniques are used to assure their security. However, all participants in such electronic voting systems should have faith in the public bulletin board that is maintained by the third party for reporting and auditing the vote results. Many digital solutions have recently been put up to deal with this problem.However, the supported voter and candidate numbers are limited, and the security measures rely heavily on client-server architecture, making these systems impracticable to employ. We provide a realistic, platform-independent secure distributed voting system built on top of the blockchain architecture to address all these problems. By eliminating the element of trust from an election, the suggested solution seeks to increase its security and transparency. The underlying blockchain design provides an unquestionable nature, and data security measures are strengthened by cryptographic techniques including CSPRNG, salt hashing, and proof of work. Using the method has a number of significant societal advantages as well, such as a simpler and quicker voting procedure that will increase voter turnout. There is a chance that our nation will adopt a system like ours in the future.

Index Terms

EVoting, Blockchain, Decentralized, Hashing, Ethereum, Ganache, Docker, Consensus, Proof of work, Byzantine Fault Tolerance.

I. Introduction

IN a democratic society, voting is important. The provision of a more reliable and trustworthy voting system receives considerable budgetary support from almost every local government. One of the main issues with voting systems based on the bulletin is whether the vote results that are published on the bulletin can be trusted. Blockchain offers a new paradigm for achieving public verifiability in such electronic voting systems, thanks to cryptocurrency's amazing success and expanding popularity.

A blockchain-based system lacks a reliable central coordinator and instead relies on each participating node to store the data block locally. The blockchain may be viewed as a hypothetical third party that can be trusted for accuracy and availability, presuming that the decentralised consensus mechanism is safe and that a sufficiently high percentage of blockchain network nodes are honest. Since the blockchain is immutable, any activity that modifies the data in a block infringes on the blockchain consensus rule and is thus rejected by the network.

The paper is organised as follows:

- Section II : Literature Review
- Section III: Problem Statement and proposed solution
- Section IV : Results and Analysis
- Section V : References

II. Literature Review

1) **Voting System in public bulletin:** In the following, we outline the key cryptographic techniques used in public bulletin based voting systems.
   - **Homomorphic Encryption:** This feature enables one to work on cipher-texts without unscrambling them. For a casting a ballot framework, this property enables the scrambled polls to be tallied by any outsider without releasing any data in the vote.
   - **Mix Net:** The fundamental thought of mix-net is to play out a re-encryption over an arrangement of ciphertexts and shuffle the request of those ciphertexts. Mix node just knows the node that it quickly got the message from and the prompt goal to send the shuffled messages to.

- **Zero Knowledge Proof:** Zero-knowledge evidence is frequently utilized in a casting a ballot framework to let the prover to demonstrate that the announcement is surely what it asserted without uncovering any extra knowledge of the announcement itself. In a voting system, the voter should convince the authority that his ballot is valid by proving that the ballot includes only one legitimate candidate without revealing the candidate information.

2) **Voting Systems in Blockchain:** The blockchain-based voting systems can be discussed under three broad categories as follows.
   - **Voting Systems under Cryptocurrency:** The suggested voting procedures in this are bitcoin-based. These use the lottery protocol, therefore the ballot does not need to be encrypted or decrypted. The ballots are distributed using zero-knowledge proof, and they are concealed using random numbers. Making a deposit before casting a ballot may encourage voters to follow the rules, but dishonest people might still lose their vote by declining to cast one.
   - **Voting System Using Smart Contract:** With maximal voter anonymity utilising Blockchain, it is an open voting network that uses a decentralised, self-tallying Internet voting protocol. To achieve self-tallying, they have included a smart contract as a public announcement. However, due to the number of voters being restricted, this voting mechanism can only be used to elect two candidates at a time.
   - **Voting systems using blockchain as a ballot box:** In this case, the blockchain structure really serves as the system's voting box. With the use of encryption mechanisms, the votes are saved inside the blocks. This offers a system that is more reliable and adaptable.

III. Problem Statement and Proposed Solution

Problem Statement: One of the most crucial pillars of any democracy is voting. The existing voting system has a good foundation, but it also has several flaws. One of the most prominent is a loss of movement. Voting is a very important process in which every citizen of any country or organization is expected to participate. Because voting centers are fixed at a specific station, it is extremely difficult and inconvenient for people to stand in long lines to vote.

Proposed Solution: To develop a digital decentralized fault tolerant voting application based on Blockchain architecture and consensus protocol which will be more secure, robust and more stable than the current operating voting system.

Proposed System Architecture:

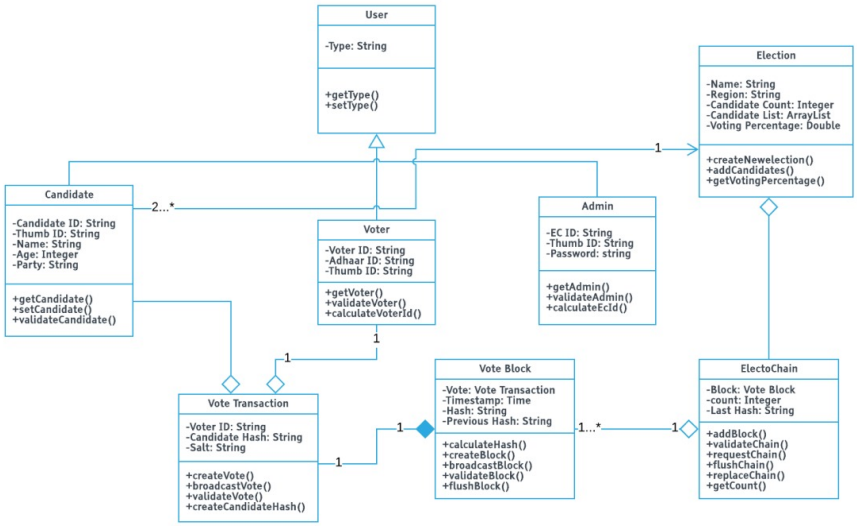

Fig. 1: Class Diagram of the Proposed System

1) **Voter Authentication:** Aadhaar API services are used for voter authentication. The system authenticates users using two separate Aadhaar APIs. The voter must first submit his or her thumb print, which the Aadhaar Thumb Impression API verifies, and then present their Aadhaar ID, which is then verified by the Aadhaar eKYC API. Here, the Aadhaar API also verifies all of the voter's information, including their age and electoral district, before the system uses the SHA-256 hashing technique to generate a special, secure voter ID.
2) **Vote Casting and Transaction Broadcasting:** After successful authentication, the voter must select any candidate from the candidate list and cast their ballot. A vote-transaction is formed at the peer where the vote was cast when a voter chooses one of the candidates. The following fields make up this vote-transaction:
   - *Transaction ID:* A 256-bit unique ID to identify the transaction.
   - *Voter ID:* A 256-bit secure hash ID of the voter created at the time of authentication.

- *Salt:* A random string of characters generated dynamically which is appended on candidate ID.
- *Candidate Hash:* Every candidate is assigned with a unique candidate ID by Election Commission of India. A salt generated is appended on this candidate ID and this complete string (ID + salt) is hashed using SHA-256 algorithm. This unique hash acts as candidate Hash.

This generated vote-transaction is broadcasted to every peer in the distributed peer to peer network. At the time of broadcasting, the vote-transaction is encrypted using SSL encryption technique. The transaction is decrypted when received by each peer.

3) **Proof of work and Block Broadcasting:** Each peer begins the proof of work procedure after receiving the broadcasted transaction. The main goal of this operation is to build a block for the vote-transaction. There is only one vote transaction per block. Here, each peer will attempt to solve a mathematical riddle including cryptography that varies in difficulty. A hash ID is generated for the block by the peer who solves this challenge first. The fields in this block are as follows:
   - *Time Stamp:* It is an instance of time at which the block is created.
   - *Transaction:* This is the vote-transaction created earlier.
   - *Hash ID:* This is a secure 256-bit hash ID created by a proof of work having difficulty count 4 and a random nonce string.
   - *Previous Hash ID:* This is a secure 256-bit hash ID of the previous block in the blockchain.

   After the block is created, the peer who has created the block again broadcasts it to every other peer in the distributed P2P network. After receiving this broadcasted block every peer stops their proof of work process and start the next process of block validation.
4) **Block Validation and Consensus Protocol:** Each peer begins the block validation procedure after receiving the block. Every peer here verifies that the prior hash ID of each block coincides with the previous hash ID in their local copy of the blockchain, starting with the new block and going all the way back to the genesis block. The block validation procedure has been successful for that peer if all the matches are accurate, at which point the peer broadcasts a success packet into the P2P network. The block validation at that peer fails if any of the matches turn out to be false, and that peer subsequently broadcasts a failure packet into the P2P network. The new block will only be added to the blockchain copy at each peer by using the consensus protocol if the majority of peers are successful. The new block is deleted and flushed out of the system if the majority of peers fail to validate it. Therefore, each time rather than relying on only one server, we are taking the majority by taking into account each peer, greatly enhancing the reliability. The voter is also given a second chance to cast his ballot. The voter receives the confirmation if the block is successfully uploaded to each peer's copy of the blockchain.

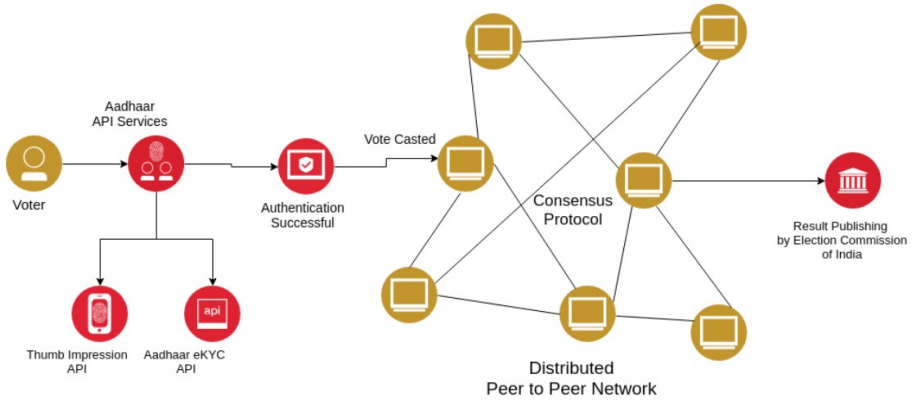

Fig. 2: Architecture Control Flow of the Proposed System

5) **Byzantine Fault Tolerance:** Everytime after the consensus protocol, the process of byzantine fault tolerance is carried out. Here, if the blockchain copy at any peer is corrupted or disturbed or tampered, then that peer requests the copy of blockchain from its nearest peer holding valid blockchain copy, and replaces its own blockchain copy by the new one.
6) **Vote Calculation and Result Publishing:** This phase of system is reserved only for the officials of Election Commission of India. Here the officials are authenticated by using their thumb impression and election-commission ID. At the time of vote calculations, the transactions in every block is traced out. The salt from each transaction is appended to the each candidate ID and hash is generated by SHA-256. If the newly created hash and the candidate Hash from the transaction matches then the vote is casted for that candidate for which we got the successful hash-match and we increment its vote-count by 1. In this way result is calculated and can be published in a very short time. Also as every peer holds the copy of same blockchain, hence the result can be calculated at any peer. Also the total number of blocks gives the total number of votes casted, hence the percentage of voting can be easily recorded.

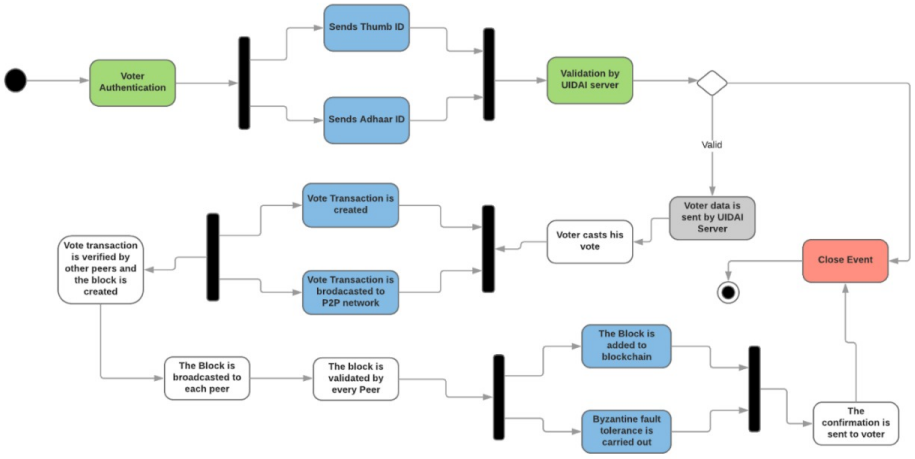

Fig. 3: Operational Flow of the Proposed System

IV. Results and Analysis

The system proposed in this paper eradicates most of the issues faced by the current voting system in India. The issues overcame by the proposed systems are as follows:

- Privacy of the voter is completely maintained as the voter's details are never kept at any peer they are directly verified at UIDAI Aadhaar API servers.
- As every piece of data is stored using hashes and salts generated by SHA-256, the anonymity of the voter and also the data security is maintained at its peak.
- Authentication is carried out by thumb impression and Aadhaar ID with secure UIDAI servers hence, fake voting and double voting is avoided.
- The time and expenses required to carry out election is remarkably lesser than current voting system.
- The confirmation of the successful vote cast is provided to the voter.
- The results can be published in a very short time.

References

[1] Peer to peer network https://www.computerworld.com/article/2588287/networking/networking-peer-to-peer-network.html
[2] Consensus Protocol https://blockgeeks.com/guides/blockchain-consensus/
[3] Zhao, Z., Chan, T.H.H.: How to vote privately using bitcoin. In: International Conference on Information and Communications Security. pp. 82–96. Springer (2015)
[4] SHA-256 https://www.mycryptopedia.com/sha-256-related-bitcoin/
[5] SSL Encryption: http://info.ssl.com/article.aspx?id=10241
[6] Byzantine Fault Tolerance https://medium.com/loom-network/understanding-blockchain-fundamentals-part-1-byzantine-fault-tolerance-245f46fe8419
[7] Platform-independent Secure Blockchain-Based Voting System, Monash University Australia CSIRO,Australia,The Hong Kong Polytechnic University https://eprint.iacr.org/2018/657.pdf