Blockchain Voting System

Aswathy J.S.[1], Bertila Mary[2], Maria Manu Joseph[3], Sneha A.S.[4], Ms. Nisha J.R.[5]
[1]*Trainee Engineer, Quest Global, India*
[2]*Software Engineer, Rapid Value Solutions, India*
[3]*B.Tech Graduate, India*
[4]*Systems Engineer, Infosys, India*
[5]*Assistant Professor, Marian Engineering College, India*

Abstract **- Democratic countries have been experiencing autocratic regimes which have introduced widespread terror among their people. The hazards of the current system of ballot voting are being misused by people or organizations looking to acquire power. Voter fraud, ballot stuffing and booth capturing are the fraudulent activities found in the existing voting system. The system that is being proposed solves most of the issues related to traditional ballot voting and can be implemented in the current world environment. Blockchain offers a lot of new opportunities to develop various types of digital services. While research on the topic is still ongoing, it has mostly focused on the technical and legal issues instead of taking advantage of this innovative idea and creating advanced digital services. In our project, we are going to leverage the open source Blockchain technology to develop a new electronic voting system which can be used in local or national elections. The Blockchain-based system will be secure, reliable, and anonymous, and help in increasing the number of voters and trust of people on the government.**

Index Terms **– Blockchain, e-voting system, i-Voting system, Hyperledger sawtooth, Angular, validator node, transaction processors, Consensus engine, Byzantine fault tolerance.**

I.INTRODUCTION

Democratic voting is a pivotal event in any country. Most countries use the paper based system for voting purposes. Digital voting is the use of electronic devices, such as voting machines or an internet browser, to cast votes. These are sometimes referred to as e-voting when voting using a machine in a polling station, and i-voting when using a web browser. Security of digital voting is always the biggest concern when considering the implementation of a digital voting system. With such huge decisions at stake, there shall be no doubt about the system's potential to secure data and defend against potential attacks. One way to solve security issues i s b y u s i n g the technology of blockchain. Blockchain technology arises from the underlying architectural design of the crypto currency, bitcoin. It is a distributed database where records take the structure of transactions; a block is a collection of these transactions. By using blockchain, we can devise a secure and robust system for digital. This paper outlines our ideas of how blockchain technology can be used to implement a secure digital voting system. have shown their interest in the subject, and a lot of research has been done.[2]

Estonian I-Voting System

The Estonian internet voting system formulates on the Estonian ID card. The card is a national identity document which is regular and mandatory as well as a smart card which serves as a secure remote authentication and legally binding digital signatures by using the Estonian state supported public key infrastructure. The principle of "one person, one vote" is persistent as the voter can potentially cast more than one ballot but still only a single vote.

Norwegian I-Voting System

The Norwegian government allows a few voters to cast vote from home using their own devices. The main issues with Norwegian i-voting system were coercion and the security issues with the personal devices. They dealt coercion by conducting a re-voting.

New South Wales iVote System

It had a different design than the Norwegian system. Some of the analysis found protocol flaws, including vote verification that was itself susceptible to

manipulation. It emphasizes the challenge of conducting secure elections virtually and brought warning for voters, election officials, and the e-voting research community.

II. TECHNOLOGIES USED

A. Blockchain

A blockchain is an expanding list of records, called blocks, which are chained together using cryptography. Every block comprises of a cryptographic hash of the previous block, a timestamp, and transaction data (generally illustrated as a Merkle tree). The timestamp justifies that the transaction data remained when the block was disclosed so as to get into its hash. Blocks contain the hash of the preceding block, forming a chain, with each additional block

III. RESEARCH

A. Existing Electronic Voting Systems

Since the system was first introduced, many scholars

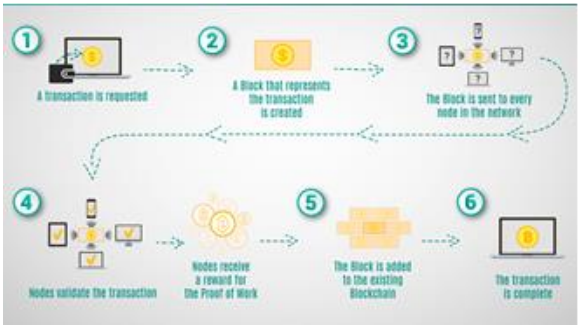

Fig.1. Working of Blockchain system[5]

strengthening the ones before it. Blockchains are usually managed by a peer-to-peer network to utilize as a publicly distributed ledger, where nodes mutually abide by a protocol to communicate and verify new blocks. Although blockchain records are not immutable as forks are viable, blockchains may be considered secure by design and illustrate a distributed computing system with strong Byzantine fault tolerance.

Bitcoin is considered the first application of the Blockchain concept to create a currency that could be exchanged over the Internet relying only on cryptography to secure the transactions.

Types of blockchain are: [5]

Public Blockchain

Anybody can participate in public Blockchain network. Examples are Bitcoin and Etherium. Public blockchain is permission-less.

Private Blockchain

It is similar to public blockchain but with limited access. Only parties who are given access can participate. Here, data is confidential. Example is Hyperledger Fabric.

Consortium Blockchain

It is an extension of private blockchain. Consortium blockchain is formed by a consortium of individuals or groups. Example is Quorum

Hybrid Blockchain

The hybrid blockchain is a combination of the public and private blockchain. This means that some process is kept private and others public.

B. Node.js

Node.js is an open-source, cross-platform, back-end JavaScript runtime environment that runs on the V8 engine and executes JavaScript code outside a web browser. Node.js permits developers to use JavaScript to write command line tools and server-side scripting— running scripts server-side to produce dynamic web page content before the page is driven to the user's web browser.

Though .js is the standard filename extension for JavaScript code, the name "Node.js" doesn't mention a particular file in this context and it's only the name of the product. Node.js has an event-driven architecture capable of asynchronous I/O. These design choices aims to hone throughput and scalability in web applications with much input or output operations, as well as for real-time Web applications.

C. Angular

Angular is a very powerful JavaScript-based front-end web framework. It is applied in Single Page Application projects. It is an application design framework and development platform for creating efficient and sophisticated single-page apps. It is completely extendible and works well with other libraries. AngularJS' data binding and dependency injection eliminate much of the code which would have been coded manually.

D. Hyperledger Sawtooth

Hyperledger Sawtooth is an enterprise remedy for developing, deploying, and executing distributed ledgers which are also called blockchains. It provides a highly modular and adaptable platform for implementing transaction-based updates to shared state between untrusted parties organized by consensus algorithms.

IV. PROPOSED SYSTEM

The proposed system includes four main requirements listed below: [1]

Authentication

Only people who have already registered can cast a vote. Our system does the registration process by using Aadhar id and a customized password from the user. Registration usually requires verification of certain information and documents to comply with current laws, which cannot be done online in a secure manner. Therefore, the system should be able to verify voter's identity against a formerly verified database, and then they can cast their vote only once.

Anonymity

The e-Voting system should not let any links between voters' identities and ballots. The voter must remain unidentified during and after the election.

Accuracy

Votes must be precise; every vote should be counted, and can't be changed, duplicated or removed.

Verifiability

The system should be verifiable to make sure all votes are counted correctly. Beside the main requirement, our solution supports mobility, flexibility, and efficiency.

A user interface for casting vote is created to examine the real world possibilities of our solutions by being in our limits. The user interface consists of two phases:

Uploader-client

The role of the uploader client is to upload the details of voter and candidate. The uploader client will be uploading the details before election. Here the addition of data on the database is done using a private key which will be unique for each candidate and each voter.[3] The private key is hardcoded with the public which means the private key data will be embedded with some function which does the corresponding action. This is done because in transactions related to bitcoin, we have to make the sender anonymous in blockchain and so the private key remains secret for each sender. In case of our public, it might not be effective if they are not well aware about the system. Also, since it is hard-coded they represent unchanging pieces of information. For the candidates, the candidate name, representational sign and the election area are given as the inputs. The Election area can be selected from the drop down list. The addition of candidate details takes place whenever we conduct an election. For voters, the details of the voters, specific to the respective voting stations are uploaded. The name, id, password and polling station are given as the inputs. The polling station can be selected from the dropdown list which is unique to each voter. The addition of voter details takes place when a person initially casts a vote.

Voter-client

The role of voter client part is to do the voting. This client part carries major role on the day of election. Initially, the authentication process will be taking place by checking the voter's id and password. If the id is not valid, it will pop up an error message showing invalid id. If the password does not match, it will show an authentication error. If user id and password matches, the voter will be navigated to the next page. In voting page, the data is fetched from the state database which is there in the validator part of hyperledger sawtooth. The voter can vote only for one candidate. All the buttons will be disabled after voting, so that the voter will not be able to change the casted vote. The data is directly sent to the validator and the voting status of voter will be changed to true. By default, it is always false. If the voting status is not false then initially after authentication, the voter will not be directed to vote casting page. When the voter reaches the vote casting page and done with voting, it will be checked if the voter has done the voting properly or not. If voting is done, then voter will be redirected to the login page or else to the voting page.

V. WORKFLOW

Initially, the voting system is intended to do the voter registration. The issue with the current voter registration is that it lacks immutability as well as authenticity. So, the main idea of having blockchain based voting system is to set up an immutable and tamper proof voting solution. Here, we are using a private blockchain solution called Hyperledger sawtooth. The reason, why we are not using the conventional or famous blockchains like Etherium, Corda or Tron is that, those are public blockchains up to an extent. Hence, even though pseudo- anonymity of an entity is kept in Etherium based blockchain solutions, we can never assure that these solutions are completely anonymous. Anonymity is something that should be kept at the core in a solution, like that of a voting system. So, the mentioned solution is using a Hyperledger sawtooth based implementation.

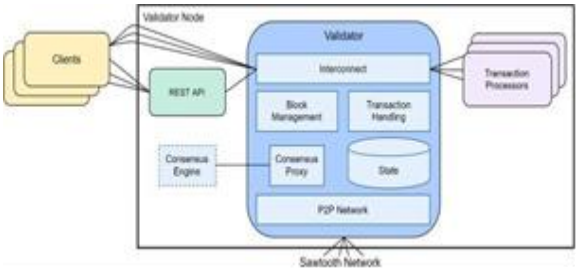

Fig.2. Sawtooth Architecture[6]

Hyperledger sawtooth is a blockchain solution developed by Intel and it falls under the family of Hyperledger ecosystem that is by Linux foundation. In this Hyperledger sawtooth implementation, we are creating a client for the voter registration and voting as well as for the transaction processor to handle the voting. The architectural components of a Hyperledger Sawtooth are Clients, Rest API, Transaction processors, Consensus engine and Validator.[6] Here, we are considering validator as a black box. Validator handles the validation of transaction which is received from the client as well as transferring transaction to the transaction processor. Also, validator is an entity that communicates with the network. The primary components of the validator node are an interconnection between transaction processor and clients rest API servers, a state database, the consensus engine and the transaction-based other entities.

Initially, there is a form for user registration. This form reads the Aadhar id and the customized password by the user. Hyperledger sawtooth gives us the freedom to setup the address in the developer's required way. A transaction hash is created for the purpose. A bulk of transaction is taken and it is encoded later. The transactions are bugged into batches. Batch is the fundamental and atomic unit of sawtooth based transaction entity. This transaction is later committed through rest api servers through validator node. And validator node takes this transaction into the transaction processor. The transaction processor validates use-cases based on the required system. Then, it check whether the voter has registered or voted previously i.e., the status of voting. Once the transaction processor completes validating the data, it will be returned back to the state database of validator. Likewise, it will be propagated throughout the blockchain network through gossip protocol. The gossip protocol is a protocol followed by sawtooth system for propagation and it confirms that all the data about transactions are delivered to all nodes on the network.

Fig.3. Workflow

The proposed system is using an angular based system for client application and docker containerization for setting up the transaction processes and entities. The system makes use of nginx servers for front end to interact with blockchain based back end.

The flow of the system can be explained as: The values are read from the fields and passed in through a function that sends data by specifying the corresponding action. The payload is set with the passed values which are encoded. The payload can be understood as: when data is sent over the Internet, each unit transmitted includes header information and the actual data being sent. The header identifies the source and destination of the packet, while the actual data is

referred to as the payload. Then, address has to be created. Since in blockchain, each node is identified by a unique address that is generated through hashing. The Address for voter is set according to the voter id. The Address for the candidate is set according to the area of the candidate. The address is sent to Transcation Header. Then, the transaction is signed with Transcation Header byte i.e., the transaction is signed by the signer's private key and the resulting signature is stored in the header. After that, the batch is passed with the public key. The batcher public key must match with the public key used to sign the batch in which contains the current transaction. Then it is sent to the Rest API server of the validator node and finally it will be passed on to the network.

VI. RESULTS

The interface was tested with few users.

The interface directs in the following manner: voter registration and candidate registration will be done prior to the day of election. When the interface is launched on the day of election, at first, the user will be viewing a voter login page in which the required credentials are entered. Then, if they are valid, he will be guided to the voting page. After casting the vote, he will return to the home page.

The user remains anonymous. Because, the data is encrypted before appending to blockchain network.

It was noted that for smooth user experience, proper internet connectivity is mandatory.

The interface was easy to use even for common people.

Anyone can use it from anywhere.

Based on the test run, the following factors were observed:

Features	High	Medium	Low
Security	✔		
Accuracy	✔		
Anonymity	✔		
Verifiability	✔		
Performance		✔	
Interface		✔	
Ease of use	✔		

Fig 4. Analysis Table

VII. LIMITATIONS

The proposed system uses a customized password from user for registration and logging in. If someone can obtain the password and Aadhar id from the voter, prior to vote casting, then the system fails to provide a safe and secure authentication step. Initially, a thought of facial recognition or iris recognition came up. But, due to the possibility of unavailability in required devices among common people, diminished such thoughts from implementation.

Also, if any mistake happens from the side of voter while voting for the candidate, then the system remains helpless. Because, a vote once casted is immutable.

VIII. CONCLUSION

The traditional voting system has a lot of issues that leads to widespread issues in a democratic country. It is essential for a democratic country to have a voting system with least barriers. The proposed system not only handles voter privacy and audit ability but also provides a transparent system for verification of the election. The main factor is remote vote casting is possible which avoids the issue of returning to hometown just for casting a vote. Also, it proves to be economically feasible compared to traditional voting systems. This solution can be implemented with existing infrastructure owned by a nation. It is obvious, that the mentioned system has defects in the authentication phase. But, a voting system based on private blockchain can initiate a lot of changes. Keeping all these facts, the proposed system is a comprehensive solution that satisfies most of the requirements during voting.

ACKNOWLEDGEMENT

We would like to express our sincere gratitude to Dr.Ruby Abraham, Principal, Marian Engineering College and Dr.Jayaprakash Pavithran for the help and facilities rendered towards the completion of this project. We wish to place on records our ardent and earnest gratitude to our project guide Prof.Nisha J.R., Assistant Professor, Department of Computer Science and Engineering, Marian Engineering College for her consistent guidance and support.

REFERENCES

[1] Blockchain Architecture Basics: Components, Structure, Benefits & Creation", Anastasiia Lastovetska, 2021.
[2] Architecture of Hyperledger Sawtooth: A comprehensive Overview", Toshendra Kumar Sharma, 2020.
[3] Hyperledger Sawtooth for Application Developers",learnthings.online, 2019.
[4] "Secure Digital Voting System based on Blockchain Technology", Kashif Mehboob Khan, Junaid Arshad and Muhammad Mubashir Khan, 2018.
[5] "A Conceptual Secure Blockchain-based Electronic voting system", Ahmed Ben Ayed, 2017 International Journal of network security and its applications(IJNSA) Vol.9, No.3, DOI:10.5121/ijnsa.2017.9301.
[6] "A public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms", T. ElGamal, IEEE Trans. Info. Theory. Vol. 31. (1985), pp. 469-472.