International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 08 Issue: 06 | June 2021 www.irjet.net p-ISSN: 2395-0072

E-Voting System Using Cryptographic Hash Functions

Ms. Sneha Mishra[1], Nitya Gupta[2], Piyush Jain[3], Nishchint Gauniyal[4]

[1]*Assistant Professor, Dept. of Information Technology, Dr. Akhilesh Das Gupta Institute of Technology and Management*
[2,3,4]*UG Students, Dept. of Information Technology, Dr. Akhilesh Das Gupta Institute of Technology and Management*

Abstract - *In recent days, you might have frequently come across terms like blockchain, bitcoin, cryptocurrency, etc. Manipulation of data by any third party is not possible in the case of blockchain as data is stored in different servers globally, allowing real time entries to be viewed. This report aims to outline our proposal to rectify the problems of digital voting by using blockchain technology. Traditional elections have satisfied neither citizens nor political authorities in recent years. As it is not entirely secure, and since it is easier to attack votes, it also threatens the privacy and transparency of voters. Therefore, most countries continue to research and improve their E-voting process. Based on this, we are going ahead to make an e-voting system which will work on the fundamentals of blockchain, optimizing for the requirements and limitations identified. Using the E-Voting System, manipulation of votes is impossible as this app first verifies the details of the voter and, after a successful transaction, adds him to the blockchain in the end. Also, it is much faster, safer and gives the voting result instantly:*

An aadhar card authenticates the voter, and the user is then redirected to the dashboard. The voter's number is verified. Then after successful verification, all the credentials will be shared through SMS, which will come handy during the voting process. A voter will be eligible for voting to the parties added in the backend by the Polling Station manager. After voting, a private key of a voter is necessary to check if the voters are qualified for casting a vote or not. After the verification is done, voters are added to the end block of the blockchain. After this, nobody can edit the data once it is added to the block.

***Key Words*: Blockchain, Electronic Voting, Ballot, Database, Transaction, Nodes**.

1.INTRODUCTION

Almost all fields now have much advanced technologies, but we are still lagging when it comes to voting. We still hear comments like, and our votes are being tampered with, EVM machines were defective, options are not counted correctly, etc. Any elections in a democratic country should protect anonymity and integrity and this can be achieved with the help of blockchain. Blockchain is a technology that gained so much popularity recently with a strong encryption base that allow**s** applications to take advantage of these features to obtain elastic security solutions. Bitcoin is a distinguished application of blockchain but researchers are keen to explore the utilization of blockchain technology to facilitate applications across totally different domains investment edges like non-repudiation, integrity and confidentiality. In this paper, we tend to explore the utilization of blockchain to facilitate e-voting applications with the power to assure elector obscurity, vote integrity and end-to-end verification. We tend to believe e-voting will leverage basic blockchain options like self-cryptographic validation structure among transactions (through hashes) and public accessibility of distributed ledger of records.

Any voting system can perform the following functions :

1. Everyone eligible should have equal opportunities to vote.
2. All the votes should be counted appropriately.
3. Options should be immutable, which disallows tampering.
4. The identity of the voters during the ballots should be maintained and kept anonymous.
5. Results will be the same as the votes done by the authorized voters.

2. Methodology

2.1 Blockchain

Since the world is becoming more digital, the theory about blockchain is becoming more acceptable. In 2008, Satoshi Nakamoto invented the first cryptocurrency called bitcoin, which revolves entirely around blockchain. It may seem complex, but its core concept is quite simple. To understand it better, knowledge about databases are important as in simpler terms blockchain is the type of database.

The database is a collection of data or information that is stored electronically in either a Table or spreadsheet format. The main difference between blockchain and database is the organized data formatting.

The way data is structured significantly between a traditional database and a blockchain. Blockchain contains digital transactions that organize data into groups called blocks, each containing some data. Blocks are chained with the previous blocks after their storage capacities are filled, forming a chain of data known as a "blockchain." A database organizes data into tables, while a blockchain organizes data into linked chunks, commonly called blocks. So, we can conclude that all blockchains can be databases but all databases can't be blockchains.

We've used the fundamentals of the Merkle tree in this Application. The Merkle tree is an important part of blockchain technology. Merkle tree is a mathematical structure composed of hashes of various blocks of facts, and which presents as a precis of all digital ledgers in a block.

2.2 ELECTION ROLES AND PROCESS

2.1 Roles

While defining an intelligent contract (intelligent contracts are essentially programs that run when predetermined requirements are met and are placed on the blockchain), we should first identify the roles during voting. Elections in our proposal allow individuals or institutions to perform several roles:

1. Election administrator- Election administrators are in charge of overseeing an election's entire lifecycle. Several reputable institutions and businesses have enrolled in this role. Election administrators define the election type, create the preceding election, configure ballots, register voters, determine the election's lifetime, and assign decentralized nodes.
2. Voters- Voters can authenticate themselves, load election ballots, vote, and validate their vote after an election is completed for elections in which they are registered.
3. District nodes- When election administrators build an election, each voting district's vote intelligent contracts are deployed in the blockchain. Every corresponding district node is granted permission to connect with their corresponding ballot smart contract when the ballot smart.

   Contracts are formed. When an individual voter votes from their corresponding smart contract, the vote data is checked by all the numbers of cluster nodes, and any ballots that they settle on are appended on blockchain after block time has passed.

1.) **Bootnodes:** Each organization that has network permissions hosts a boot node. A boot node serves in the discovery and communication of district nodes. The boot nodes do not hold any blockchain state and run on a static IP to help district nodes locate their peers more quickly.

2.3 Process

Election process includes:

1. Election establishment - Using a decentralized app, election administrators build election ballots. This decentralized app communicates with an election establishment smart contract, here the administrator specifies a candidate list and voting districts. This smart contract generates a sequence of intelligent ballot contracts. It deploys them into the blockchain, in company with a list of candidates for each voting district, with each voting district being a parameter in each smart ballot contract. Each adjacent node is granted permission to communicate with its corresponding ballot acute contract.
2. Registration of voters - The election officials handle the registering of voters. Administrators should identify a testable list of registered voters when creating an election. This entails a component for the Identity authentication program that authenticates and safely authorizes qualifying people. Using such identification services, each lodged voter should have a voter id. A wallet will be created for each registered voter. Each voter's wallet has to be distinctive for each election that the candidate is entitled to vote in.
3. Transaction - while voting, voters deal with intelligent ballot contracts according to the district assigned for them. This smart contract connects with the blockchain via the corresponding district node, which appends the vote to the blockchain if most corresponding district nodes achieve agreement. Each vote is stored on the blockchain as a transaction, and the transaction is sent to the voter for further verification. Each blockchain transaction contains details about who was voted for and where the vote took place. The corresponding ballot smart contract appends each option to the blockchain if all connected district nodes agree on the vote data

verification. When a voter casts a ballot, the weight of their wallet is reduced by one, preventing them from voting more than once per election.

4. Tallying - The election results are calculated in real-time by intelligent contracts. Each smart ballot contract keeps track of its count for each position in its storage. The final product for each smart contract is released after an election is completed.
5. Verifying - As previously stated, each voter receives his vote's transaction ID. After authenticating himself with his voter ID or Aadhar card, each voter will go to his government official and show their transaction ID. Using district node access to the blockchain, the government official uses the blockchain explorer to locate the transaction with the corresponding transaction ID on the blockchain. As a result, the voter will check his vote on the blockchain to ensure that it was counted and counted correctly.

3. PROBLEM STATEMENT

In the traditional EVM based voting system, we have faced several problems such as :

1. Much man work is required for security and verification purposes, which is not wholly reliable and always leads to several mistakes.
2. Wastage of papers and other resources.
3. Tampering of votes and machines and change in outcomes.
4. Results in traditional voting are delayed or take more time as the votes are counted manually.
5. The poll conducted in rural areas lacks several resources.

4. PROPOSED SOLUTION

This blockchain-based voting system project will develop a distributed electronic voting system and identify the technological limitations. As we know blockchain creates nodes and each node has some set of data which will help in solving problems. A set of rules (called the consensus protocol) defines the order in which nodes may take turns adding new changes to the database. Thus all nodes on the blockchain agree to the state of the database and no node has authority to forge the date. The record of the changes in the database is known as transactions. The sequence of these transactions that are added one by one in a blockchain are actually the blocks. All the blocks are ordered on the basis of the reference they have of the blocks preceding them. This originated the term "blockchain.". Votes as transactions will help us create a blockchain that will keep track of aggregation of votes. By doing all this, the final count of votes will be unbiased and will be able to satisfy everyone because verification can be done that no votes are illegitimate or removed.

It is necessary to keep a record of the new ledger and it can be done with the usage of the Merkel tree. Ralph Merkle first presented the Merkle tree in 1979. The Merkle tree is an important part of the blockchain. It is nothing but a complete binary tree data structure formed with the hash values of every transaction in a block. Merkle trees have been an essential key for data verification throughout the evolution of computers. It also helps to verify new transactions and their consistency. Merkle trees or hash trees are structured as binary trees where each non-leaf node hash is generated by combining the hash of the two nodes below it (i.e. Child nodes). In Merkle Tree, if any change is detected in the non-leaf nodes, then we can perform binary search throughout the tree to check which particular subtree hash has been changed. Therefore, only the root nodes of the subtrees are needed to be checked rather than checking all the nodes in the tree. Merkle Tree Structure is shown below.

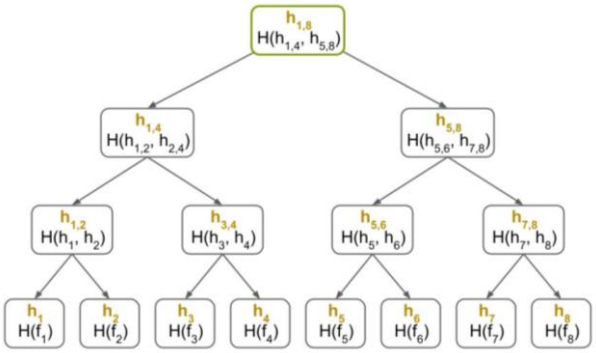

Fig 1: Merkle Tree

We can think of the Merkle tree as computing a collision-resistant hash function, denoted by MHT (Merkle Hash Tree), which takes n inputs (e.g., the n hashes generated by the details of 'n' no of voters denoted by f_1,f_2,..,f_8) and outputs the Merkle root hash. For example, we often use the following notation:

$$\square_{1,8} = \square\square\square(\square_1, \square_2, \ldots, \square_8)$$

More importantly, computing MHT (i.e., computing the Merkle Root), involves many computations of its

underlying collision-resistant hash function H. Such as the $h_{1,8}$ has a recursive structure as shown below.

$$MHT(□_1, □_2, \dots, □_8) = H(\ H(\ H(H(f_1,f_2)), H(H(f_3,f_4))),\ H(\ H(H(f_5,f_6)), H(H(f_7,f_8))))$$

Thus, from the above formula we can conclude that:

- The leaf nodes of the tree store the hash details(i.e. Details of the voters).
- The Non - leaf nodes will store the hash created by the combination of its children nodes.
- The root of the tree i.e $H_{1,8}$ is also known as Merkle root hash.

A. Algorithm:

1. First, the person will enter his/her Voter ID, and after that, an OTP will be sent on his/her registered mobile number.
2. After Verifying the OTP, the user will be redirected to the Dashboard from where the user can choose options of his/her interest.
3. If a user elects to cast a vote, then he will be directed to a window where he will be asked to choose a political party. After that, a private key will be generated and sent on the user's mobile number, which the user must enter with a private key to cast their vote.
4. In the Mine block, first, the candidate details will get verified, and then after verifying, a unique hash value will be generated, which will be used later to identify a block in the blockchain.
5. If the candidate selects the profile option, they will see their profile on the application.
6. In the results option, candidates can see the number of votes received by all the parties.

B. FlowChat for Proposed System

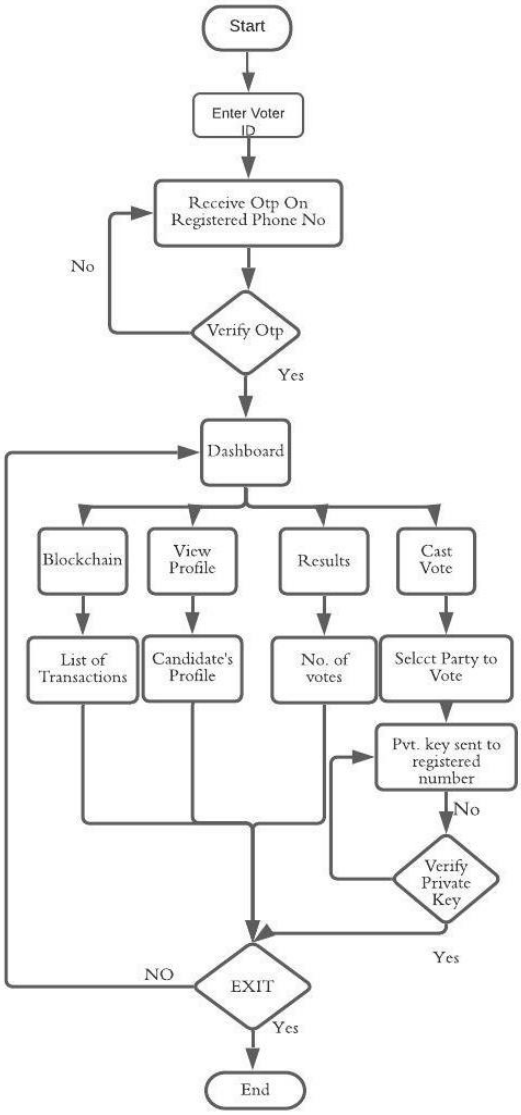

Fig 2: Flow Chart

5. RESULT AND DISCUSSION

In fig.2, E-voting login screen interface is being shown, whenever a user opens up the app. Users will be first asked to enter his/her voter ID.

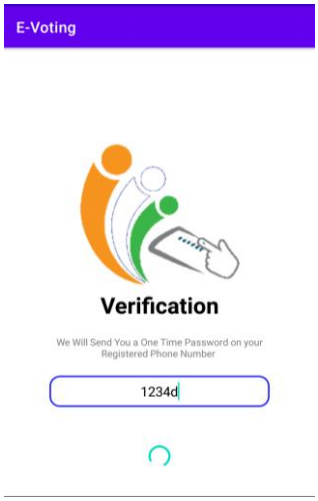

Fig 2: Interface Design - Login Screen Interface

OTP Verification

After entering the voter ID, the user will get an OTP on the registered phone number which he/she has to enter in order to move forward with the voting process, shown in fig.3.

Fig 3: Interface Design - OTP Verification Interface

Dashboard Interface

After entering a valid OTP, the candidate will be redirected to dashboard window where they will see these options:
1. Cast Vote: used for to cast a vote
2. Blockchain: used to see the block transaction
3. Candidate's Profile: to see candidate details or change lodged mobile number
4. Result of Voting: to check the election results
5. Logout, as shown in fig.4.

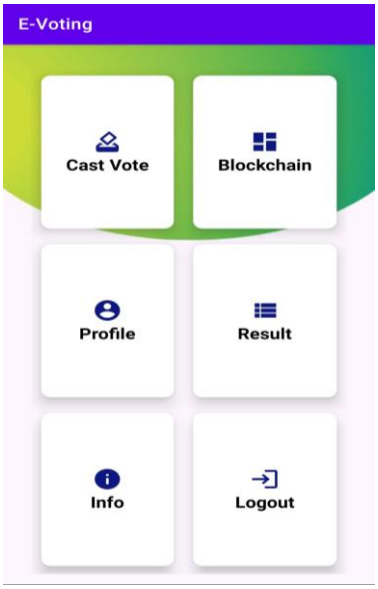

Fig 4: Interface Design - Dashboard Interface

Cast Vote

In the cast vote interface:
1. The candidate will choose the party they want to vote for.
2. After selecting the party, the candidate will receive a private encrypted key on mobile which is required in order to cast a vote, as you can see in fig.5.

Fig 5:Interface Design - Cast Vote Interface

Blockchain

In the fig.6, blockchain ledger interface is shown which contains all the transactions that were stored in the blockchain. It contains information regarding all the blocks in the blockchain, like the previous block hash value and the new merkle hash value of the block and will give the indication that the new transaction has occurred or not.

Fig 6: Interface Design - Blockchain Ledger Interface

Candidate's Profile

In the fig.7, it is shown how:
1. The candidate profile will look after getting all the data from the database provided by the Election Commission.
2. In this, the user can change their registered mobile number if they wish to.

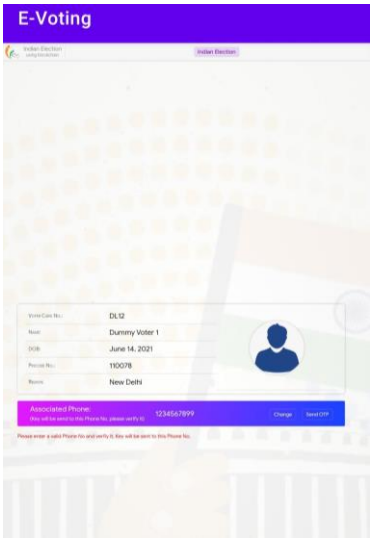

Fig 7: Interface Design - Candidate's Profile Interface

6. CONCLUSIONS

In this thesis, we presented a Blockchain-based secure E-voting framework that empowers the decentralized data set to project voting in an advanced manner. We showed how blockchain innovation can be used for voting. It can settle the security, decency, and trust issues, plus its offering to decrease the boundaries of E-voting frameworks. The Blockchain technology will be transparently specific and conveyed so no one will be proficient in ruining it. Voting with our application permits each elector to vote in a democraatic area based on their personal preference while ensuring that every citizen's vote is checked from the right region, which might build citizen turnout.

7. FUTURE SCOPE

This paper has shown how blockchain technology can solve the security, fairness, and trust issues people have regarding voting by offering them to register their votes online while guaranteeing voters' privacy. Technology, we use here, is so secure that nobody can or will corrupt it or change the results. However, this application can be extended to give more efficient performance and could enhance its privacy by making it more secure. As we know, research in any technology, be it Blockchain or Machine Learning, is not restricted to the topics mentioned in this thesis. So there is always some possibility for future research. Some ideas for future research could be directed as: The work can be extended by adding biometric or duo factor authentication through which we can corroborate that the voter is not being coerced to vote. By this duo authentication service, candidates shall be prompted a second time to recheck their choice before the vote is sent; this will permit us to eradicate accidental voting also..

REFERENCES

[1] G. K. H. M. H. G. H. Friðrik Þ. Hjálmarssonm ``Blockchain-Based E-Voting System,'' Institute of Electrical and Electronics Engineers, 2018.

[2] F. J. M. S. M. A. A. M. R. M. M. A. K. Md. Razu Ahmed, "The Future Of Electronic Voting System Using Blockchain," INTERNATIONAL JOURNAL OF SCIENTIFIC & TECHNOLOGY RESEARCH, vol. VOLUME 9, no. ISSUE 02, 2020.

[3] C. N. S. F. Z. Brihat Sharma, "merkle-tree based Approach for Ensuring Integrity of Electronic Medical Records," IEEE, 2018.

[4] D. B. T. B. M. J. S. B. Urmil Bharti, "]Android Based e-Voting Mobile App Using Google Firebase as BaaS," Shaheed Rajguru College of Applied Sciences for Women, 2020.

[5]G. k, "ADVANCED E-VOTING APPLICATION USING ANDROID PLATFORM," International Journal of Computer-Aided technologies (IJCAx) , 31 july 2020.

[6] F. Schär, "Decentralized Finance: On Blockchain- and Smart Contract-Based Financial Markets," https://research.stlouisfed.org/, vol. 103, no. 2, second quarter, 2021.

[7] A.-L. S. &. A. C. White, "Strategic targeting: authoritarian capacity, state dependent populations, and electoral manipulation," Journal of Elections, Public Opinion and Parties, vol. 21, no. 2, 24 march 2020.

[8] O. K. Y. DEBASHISH DAS2, "BLOCK CHAIN TECHNOLOGY FOR ELECTRONIC VOTING," Journal of Critical Reviews, vol. 7, no. 3, 2020.

[9] J. A. M. M. K. Kashif Mehboob Khan, "Secure Digital Voting System based on Blockchain Technology," University of Engineering and Technology, Pakistan 2 University of West London.