Review on Decentralised Application Using Block Chain

Anukool Srivastava and Shubham Verma

Abstract--- *Elections are the prime attribute in the fate of a country. Current Election Systems are a Client-Server Model based on a Centralised Database. The problem resides with a Central Authority where data could be changed or manipulated by the administrator without the knowledge of other. The survey proposed is on decentralised systems implemented using Ethereum Blockchain which enables full distribution of data along with security. A comparison between centralised and decentralised systems are used to draw out the advantage of using Blockchain as the distributed system. Migrating the code and data on a blockchain ensures the proper security of the system and any changes that need to be done in it should require a Transaction which will be reflected to each node connected to the blockchain.*

Keywords--- *Blockchain, Security, e-Voting, Database.*

I. INTRODUCTION

Greater the power, greater the control. Technology has become a helping hand for humans in recent times. The use of technology has spread across various domains. Applications in recent fields are based on Centralised Systems in which there exists a Central Point of Connection for all operations related to the application. However there are a few demerits that cannot be solved by it. Hence a concept of Decentralised System was introduced to overcome the drawbacks of the Centralised System.

Elections are one of the most important events in a country's governance. Its fairness is of prime importance. It has a great power in determining the future of the country. Elections are conducted in various forms across the globe. It includes Ballot Papers, E-Voting Machines and Online Voting Systems. Security and privacy of the voter should be assured in these systems. Each registered voter is required to verify his/her identity by providing a valid Voter ID to the Election Committee which validates them. After verification, the user is allowed to cast a vote on the Ballot Paper or Voting machine.

It often consumes a lot of time to cast a vote by a voter and declaration of its results. Time may vary from 2-4 days. Ballot papers lacked vote security since a vote could easily be forged for another person in the room.

Vote tampering has become an issue recently in E-Voting machines putting it's credibility under radar.

These drawbacks occur due to Security Issues of the Centralised System. A Decentralised System holds various advantages such as Voter ID protection, theoretical impossible hacking, voter authentication, their confidentiality and data security.

Anukool Srivastava, School of Computing, SRM Institute of Science and Technology, Kattankulathur, Kanchipuram, Tamil Nadu, India. E-mail: anukoolsriv@gmail.com

Shubham Verma, School of Computing, SRM Institute of Science and Technology, Kattankulathur, Kanchipuram, Tamil Nadu, India. E-mail: shubham96verma@gmail.com

DOI: 10.37200/IJPR/V24I1/PR200258
Received: 20 Nov 2019 | Revised: 24 Dec 2019 | Accepted: 08 Jan 2020

BlockChain is a method of applying the concept of decentralised system. It is a distributed, unchangeable and transparent ledger technology that eliminates the use of typical databases. It can be one step solution to various problems in voting systems. It consists of several verified blocks of data chained behind one another, hence the name. Each consecutive block is formed by a process of cryptographic functions that create hash values linking each block to its next. The entire ledger is distributed to the participating nodes in the network. Hence a risk of vote tampering will be conceptually impossible since there is no single point of access.

Here we analyse various risks in securities of centralised systems and provide a concept of a BlockChain system for implementation in Elections.

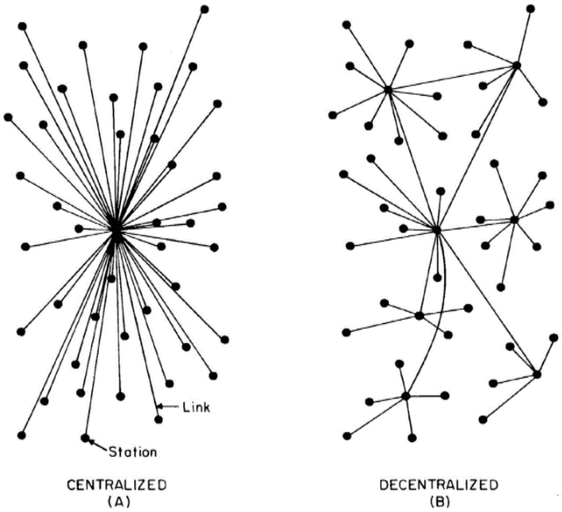
II. COMPARISON

A. Centralised System

This system is based on a conventional Client-Server Model that is based on a central authority. A centralised system runs on the concept of a star topology in which there is a major hub known as server which is used as a medium to communicate with the other peers in the network. This central hub encapsulates the data and functioning of whole system allowing only the necessary information to be accessed by the peers known as clients. Data access, encapsulation and maintenance plays a major role in a centralised system which is the sole advantage of having a Client-Server model.

B. Decentralised System

It is a system where there is no single point of control. Lack of single authority makes the system fair and more secure. Each component in the system is equally responsible for contributing to the global aim of the application. They are linked to the idea of self organisation. Further privacy is ensured by Decentralized platforms compared to a Centralised system. As every node connected to the blockchain contains the same data, information tapping and duplication becomes extremely hard and it prevents a single point of failure which is a common issue in a Centralised System.



DOI: 10.37200/IJPR/V24I1/PR200258
Received: 20 Nov 2019 | Revised: 24 Dec 2019 | Accepted: 08 Jan 2020

III. BLOCKCHAIN

A. Preface

In case of a regular web application, a centralized server is used to connect to the application. While the application runs on the centralized server, the entire data lies in a centralized database. Each communication with its data requires a mandatory connection to its centralized server. If a voting application is to be built on the web, there would be a few problems.

The problem lies in the database. It is prone to unauthorized alteration. Data could be manipulated multiple times.

Also, the application code is prone to changes that can't be traced easily and is difficult to revert back. A decentralised system provides security to most of the disadvantages of a centralised system listed above.

Blockchain is an implementation of a distributed peer to peer decentralised system. Its a network and a database all in one. Its network of systems, termed as nodes that share the code and data across the entire network. Any device connected to the blockchain will be able to communicate with other nodes in the network. A copy of all the code and data will be stored in each node connected to the blockchain. Hence they become a group of computers or nodes that communicate with each other without any central entity.

Every transaction in a blockchain network is stored in a block and connected together to create a public ledger thus implementing a blockchain.

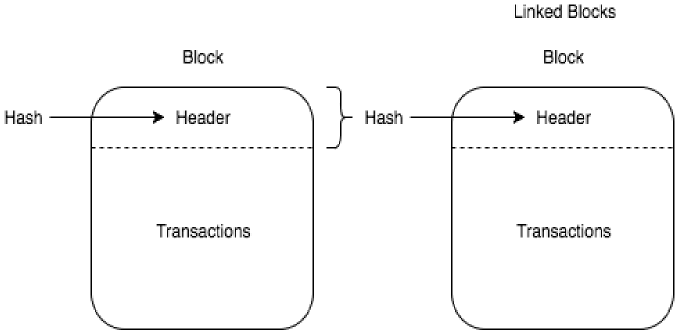
Every record in the blockchain is validated by a consensus algorithm and secured by cryptographic hashing called Proof Of Work. The nodes connected to the blockchain ensure that there is an exact similar copy distributed throughout the network. An account with a wallet address is needed by a user along with some Ether (Ethereum's cryptocurrency).

In case of an Election Application, once the user connects to the network, he will be allowed to cast his vote. In return for the vote, a small fee called "gas" is paid to record the transaction on the blockchain. For every transaction, few selected nodes or miners contend against each other on the network for the transaction. Ethereum blockchain allows the execution of code with the EVM (Ethereum Virtual Machine) on the blockchain with methods known as Smart Contracts, which contain the application logic. The application becomes decentralised at this point. Smart contracts are used to read and write data on the blockchain and execute the business logic of the application.

A contract oriented language called Solidity is used to write the Smart contacts. It is a full fledged programming language that allows us to perform tasks in a way similar to Javascript with a different behaviour. Smart contracts are identical to web microservices.

If the database layer can be represented using the public ledger of the blockchain then the entire business logic is represented by the smart contracts. They are called smart contracts because they represent an agreement.

B. Architecture



- **Hashing:** Its a cryptographic algorithm that converts any length string into a fixed size string called a hash.
- **Header:** It is a unique identifier for a particular block in the BlockChain.
- **Transaction:** small unit of task that is stored in public records. It contains all the required components to identify a unique transaction.
- **Block:** An integral component of BlockChain consisting of a header and a transaction record.

IV. APPLICATION ANALYSIS

Literature survey presented here discusses the various applications of BlockChain for elections:

A. Layout

The first phase of application development accompasses the architecture of the application. This involves the creation of data structures that holds voter information. Creation of architecture includes the links between the voter and the casting of votes. The structure should be designed in a way such that the vote is cast only once. The linking between the transaction should be done properly. There should be a proper structure that creates vote records of the votes cast to the candidate structure.

A structure for verification and updation of voter and candidate records should be properly designed[2]. There should be linking between the voters of a particular constituency to the candidate of that constituency.

Creating layers of architecture for various levels of users such as Admin, voter, District Node, Boot Node will help in developing modular architecture. The Admin should have the authority to the overall network application from creation of election to the deployment of result[6].

Voter should have the only authority to cast and view the vote result. He could given the authority to edit his vote till the end of voting period if required[1]

District Node can have the access to verification of the votes and observation of results.

Boot Node should be carefully designed to establish the network.

DOI: 10.37200/IJPR/V24I1/PR200258
Received: 20 Nov 2019 | Revised: 24 Dec 2019 | Accepted: 08 Jan 2020

B. Technology Used for Blockchain Implementation

The backend of the application is coded in Solidity. It is a Contract Oriented Programming Language that consists of Smart Contracts and their functionalities. Each smart contract is analogical to a Method in any Object Oriented Programming Language.

It is designed to run on the Ethereum Virtual Machine. Solidity has established itself as the primary language on multiple platforms such as Ethereum, Morax and hyperledger. SWIFT has also used Solidity to implement "Proof Of Concept" running on "Burrow".

Solidity is a programming language designed for developing smart contracts that run on the EVM. It is executable on the Ethereum Virtual Machine. Self enforcing business logic can be implemented with the functionalities of Solidity. These functionalities are embedded in smart contracts that form the heart of the application. They make the contract non repudiable.

It has a Migration directory and a Contracts Directory. All the contracts need a migration module to get uploaded to the blockchain. The migration directory stores all the files consisting of these modules. The Contracts Directory stores all the contracts that the application requires and stores them in a separately.

Frontend is made using conventional languages like HTML, CSS, Javascript and PHP.

The Voter is given a user Interface to interact with the Blockchain to login, choose the respective candidate to vote and to cast the vote. These votes are stored on the blockchain and are hidden from everyone. And the user is logged out once he casts the vote.

There are several dependencies that are used to make a local blockchain within a system and to provide a fake ethereum account to the users through which they can login to the blockchain.

C. Strategies Used in Blockchain Applications

The voting process is the protocol that is used while an election process runs.

SHA-256 algorithm to encrypt the voter information in the BlockChain using asymmetric cryptography[1]. Private keys have been used to cast the vote[2]. The votes were secured using hash function that linked each node to its previous node.

Records of its users in the public ledger and revoting of a candidate is defended using Proof Of Work algorithm[3].

Verification of voters has been done using district nodes that were independent for each district. The votes were monitored using independent bodies called miners[4][5].

An implementation called CoinJoin is also used to ensure confidential votes and ensure voter privacy[7]. Ethereum and Solidity were used to make backend of the application that were deployed on the ethereumnetwork[8].

RCoin was used to introduce a degree of centralisation between two components of decentralised system to give a sense of control to the network[11]. Ethereum Virtual Machine(EVM) was utilised to provide a way to make smart contracts and then are deployed to the BlockChain[12].

DOI: 10.37200/IJPR/V24I1/PR200258
Received: 20 Nov 2019 | Revised: 24 Dec 2019 | Accepted: 08 Jan 2020

Table I provides a transaction table that records every transaction made on the network Application. It can be used for tallying results and ensuring whether the voter has cast his vote or not. One record in the table uniquely identifies the transaction.

Table 1: Transaction Table

Hash	Block	Age	From	To	Value	Trans. Fee
0xdead...	1337	33 sec ago	0xsdgs...	0xqfgwl...	10	0.012
0xqwfg..	1338	12 sec ago	0xqqw12….	Token	0.5	0.00006

The voter identifies himself in the network before casting his vote[6].

A local node is setup to communicate with other nodes under the constituency node[5].

In a situation, vote is cast using zero knowledge proof i.e. either yes or no. This will ensure the voter has only two options[4].

The block consisting of the vote and the candidate information is added in the BlockChain and incase of concusses, the longest chain rule is applied to proceed with[1].

The votes care concurrently calculated for each candidate and the corresponding BlockChain is added with incremented nodes.

D. Outcome

S.no	Paper	Output Received
1.	A Conceptual Secure BlockChain- Based Electronic Voting System	SHA-256 used to secure ID of the user
2	Block Chain Based E-Voting Recording System Design	Verification Of Users asymmetric cryptography
3	A Block Chain implemented Voting System	Data Block are linked to each other using Hashing algorithm
4	A Smart Contract for Boardroom Voting with Maximum Voter Privacy	Avoidance of revoting using Proof Of Work.
5	Blockchain Based E-Voting System	Smart Contracts created using solidity for each district
6	Digital Voting with the use of Blockchain Technology	Use of QR code for identification of voter
7	Introduction to Security and Privacy on Blockchain	Use of CoinJoin for multiple payments and confidential transactions.
8	Smart Contracts: Security Patterns in Ethereum Ecosystem and Blockchain.	Implementation of Smart Contracts and solidity for deployment of application securely on Blockchain
9	Blockchain and Smart Contract for Insurance : Is the technology mature enough?	Survey of application of smart contracts to build a public ledger application
10	Transferring data through an Ethereum Blockchain using Transactions.	Analysis of creation and deployment of applications on Ethereum platform
11	Centrally Banked Cryptocurrencies	Use of RS Coin for introducing a central authority for the decentralised components of the application.
12	Blockchain based Smart Contracts : A Systematic Mapping Study	A study of mapping of smart contracts in an Ethereum network.

DOI: 10.37200/IJPR/V24I1/PR200258
Received: 20 Nov 2019 | Revised: 24 Dec 2019 | Accepted: 08 Jan 2020

V. CONCLUSION

This paper presents outcomes of various studies and approaches used to develop a decentralised application using BlockChain. The use of BlockChain is to add securities to the network and ensure maximum voter privacy. Various methods have been applied to make the network more secure and robust. Numerous algorithms have been applied to ensure secure count of voting and reduce vote tampering. It can be concluded that BlockChain has advantages that can be used to develop secure applications such as banking and voting applications. This platform can be used to secure trust in the voter. Furthermore, BlockChain has been verified by these application and outcomes that it is hack free and an immutable network. A vote cannot be tampered with once it is cast. If tampered, it can be easily tracked down.

REFERENCES

[1] Ahmed Ben Ayed, “A Conceptual Secure Block Chain- Based Electronic Voting System.” *International Journal of Network Security & Its Applications (IJNSA)* Vol.9, No.3, May 2017.

[2] Rifa Hanifatunnisa, Budi Rahardjo, “Block Chain Based E-Voting Recording System Design.”*11th International Conference on Telecommunication Systems Services and Applications (TSSA),* 2017.

[3] Francesca Caiazzo, “A Block-Chain Implemented Voting System-The Benefits and Risks of Block-Chain Voting.”

[4] Patrick Mc Corry, Siamak F. Shahandashti, Feng Hao, “A Smart Contract For BoardRoom Voting With Maximum Voter Privacy,” *Financial Cryptography and Data Security Conference,* January 2017.

[5] Andrew Barnes, Christopher Brake, Thomas Perry, “Digital Voting with the use of Blockchain Technology.”

[6] Friðrik Þ. Hjálmarsson, Gunnlaugur K. Hreiðarsson, “BlockChain Based E-Voting System,”

[7] Harry Halpin, “Introduction to Security and Privacy on the Blockchain,” *2017 IEEE European Symposium on Security and Privacy Workshops (EuroS& PW).*

[8] Maximilian Wöhrer, Uwe Zdun. “Smart Contracts: Security Patterns in the Ethereum Ecosystem and Solidity”

[9] Valentina Gatteschi, FabrizioLamberti, Claudio Demartini, Chiara Pranteda, Víctor Santamaría, “Blockchain and Smart Contract for Insurance: Is the technology mature enough?” *Future Internet, February* 2018.

[10] DNAtix Development Team, “Transferring Data through an Ethereum Blockchain using Transactions.” *DNAtix DevOps Whitepaper,* 2018.

[11] George Danezis, Sarah Meiklejohn, “Centrally Banked Cryptocurrencies”

[12] MaherAlharby, Aad van Moorsel, “Block Chain Based Smart Contracts: A Systematic Mapping Study.” *Fourth International Conference on Computer Science and Information Technology (CSIT-2017)*

DOI: 10.37200/IJPR/V24I1/PR200258
Received: 20 Nov 2019 | Revised: 24 Dec 2019 | Accepted: 08 Jan 2020