International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 07 Issue: 04 | Apr 2020 www.irjet.net p-ISSN: 2395-0072

BLOCKCHAIN BASED E-VOTING SYSTEM

Rachana H K[1], Aishwarya Bhandarkar[2], K Shobha Chandra[3], Divya M S[4], Amratha[5]

[1]*Rachana H K, Student, Malnad College of Engineering, Hassan*
[2]*Aishwarya Bhandarkar, Student, Malnad College of Engineering, Hassan*
[4]*Divya M S, Student, Malnad College of Engineering, Hassan*
[5]*Amratha, Student, Malnad College of Engineering, Hassan*
[3]*K Shobha Chandra, Dept. of Computer Science and Engineering, Malnad College of Engineering, Karnataka, India*

Abstract - *Election is a very important event in the democracy but large sections of society around the world do not trust their election system but it is a major concern to be considerate. Building a secure electronic voting system that offers the fairness and privacy, while providing the transparency, security and flexibility to the voting systems has been a challenge for very long time. The work investigates on the problems in the election voting systems and aims to propose an E-voting model which can resolve the issues such as vote rigging, hacking of the EVM (Electronic voting machine), election manipulation and polling booth capturing. Here we employ an application of Blockchain as a service to implement distributed electronic voting system. Blockchain is a constantly growing records or ledger that keeps a permanent record of all the transactions that take place, in a secure, chronological and immutable way which makes peer to peer value transfer possible. In the case of e-voting system one peer is the voter and the other is the candidate who receives the vote. By deploying blockchain technology in the decentralization of databases of a voting system can lessen the deception of database manipulation. The work to build a blockchain based e voting system will address limitations of existing systems and will also preserve participant's anonymity while still being open to public inspection.*

Key Words*: *Blockchain, Cryptocurrency, Chronological, Decentralization, Deception

1. INTRODUCTION

In every democracy, the security of an election is a matter of national security. The computer security field has for a decade studied the possibilities of electronic voting systems, with the goal of minimizing the cost of having a national election, while fulfilling and increasing the security conditions of an election. Replacing the traditional pen and paper scheme with a new election system is critical to limit fraud and having the voting process traceable and verifiable. Electronic voting machines have been viewed as flawed, by the security community, primarily based on physical security concerns. Anyone with physical access to such machine can sabotage the machine, thereby affecting all votes cast on the aforementioned machine. Blockchain-based electronic voting system could decentralize controls, making voters take over certain tasks while retaining a copy of the electoral register.

1.1 What is Blockchain and how does it work?

Blockchain is a public electronic ledger built around a P2P system that can be openly shared among disparate users to create an unchangeable record of transactions, each time-stamped and linked to the previous one. Every time a set of transactions is added, that data becomes another block in the chain (hence, the name). Every chain consists of multiple blocks and each block has three basic elements.

- The data in the block
- A 32-bit whole number called a nonce. The nonce is randomly generated when a block is created, which then generates a block header hash
- The hash is a 256-bit number wedded to the nonce. It must start with a huge number of zeroes (i.e., be extremely small)

When the first block of a chain is created, a nonce generates the cryptographic hash. The data in the block is considered signed and forever tied to the nonce and hash unless it is mined. Genesis block is the first block of the blockchain.

Genesis Block	
Previous Hash	0
Timestamp	Thu, 27 Jul 2017 02:30:00 GMT
Data	Welcome to Blockchain CLI!
Hash	0000018035a828da0…
Nonce	56551

Where each term represents;

- Index (Block #): Which block is it? (Genesis block has index 0)
- Hash: Is the block valid?

How is the hash calculated?

A hash value is a numeric value of a fixed length that uniquely identifies data. The hash is calculated by taking the index, previous block hash, timestamp, block data, and nonce as input. The SHA256 algorithm will calculate a unique hash,

given those inputs. The same inputs will always return the same hash.

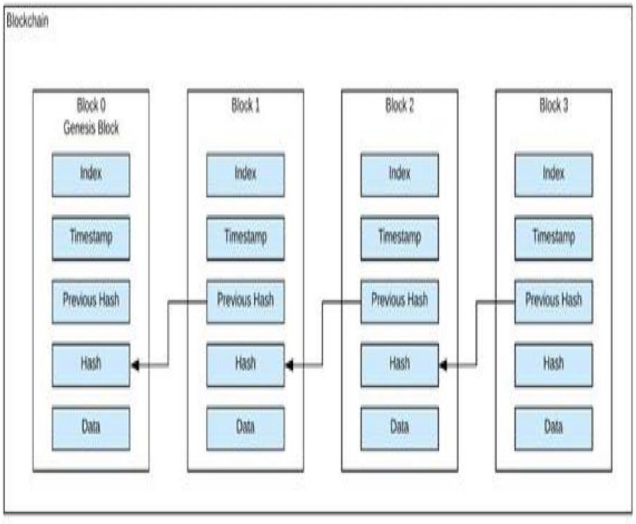
```
CryptoJS.SHA256(index + previousHash
+ timestamp + data + nonce)
```

Nonce: How many iterations did we go through before we found a valid block?

A nonce is a number used to find a valid hash. The nonce iterates until the hash is valid. In our case, a valid hash has at least four leading 0's. The process of finding a nonce that corresponds to a valid hash is mining.

```
let nonce = 0;
let hash;
let input;

while(!isValidHashDifficulty(hash)) {
nonce = nonce + 1;
input = index + previousHash + timestamp + data +
nonce;
hash = CryptoJS.SHA256(input)
}
```



1.2 Key Features of Blockchain

1. It cannot be corrupted, altering any unit of information on the blockchain would mean using a huge amount of computing power to override the entire network and even if anyone is successful in making some change, it wouldn't be possible without getting caught.
2. Blockchain technology makes replacement of trusted third parties possible as all participants have equal access to the value chain to cloud-based assets that verify each party's identity.
3. It makes material cost reduction possible through the elimination of expensive infrastructure and third-party huge fees.
4. Elimination of error handling through real-time tracking of transactions with no double spending.
5. Full automation of transactional processes, from payment through settlement.
6. Removal of documentation bottlenecks caused by duplication.
7. Transparency of data is embedded within the network as a whole, by definition it is public.
8. Blockchain technology is like the internet in that it has a built-in robustness. By storing blocks of information that are identical across its network, the blockchain cannot be controlled by any single entity.
9. Has no single point of failure because of the decentralized nature.

2. RELATED WORK

Aim was to design a secure and efficient e-voting protocol. The first thesis related to cryptographic e-voting protocol and it used an anonymous commutation channel to encrypt the ballot. The proposed a protocol requires all voters must participate and encrypt the ballot of each voter and at the end cast the ballots. The protocol encrypts the ballot by using homomorphism theorem and the government will release the tally result. The scheme used the blind signature to blind the message the voter used to vote and send it to the administrator. A lot of e-voting software had been implemented and used for the market, such as the EVOX and SENSUS.

Netvote is a decentralized blockchain-based voting network on the Ethereum blockchain. Netvote utilizes decentralized apps for the user interface of the system. The Voter dApp is used by individual voters for registration, voting and can be integrated with other devices (such as biometric readers) for voter identification. The Tally dApp is then used to tally and verify election results.

In Netvote, each election is represented by a set of smart contracts which is instantiated on the Ethereum blockchain by an election administrator through the Admin dApp. An individual voter therefore registers at the polling place and then interacts with the voter pool smart contract through the voter dApp. Each individual voter transmits a cryptographically signed vote token to the Vote Gateway for verification. The Vote Gateway then retrieves the Vote ID secret key from a vault, where the secret key is specific to each election and is destroyed when voting closes. The Vote Gateway then submits the vote payload in a blockchain transaction to the voter pool smart contract with the vote

payload mapped to the anonymous Vote ID. This Vote Gateway is used in a manner of a Zero knowledge proof to guarantee voters privacy.

Agora is an end-to-end verifiable blockchain based voting solution. This voting system is a multi-layer architecture which include the blockchain, called the Bulletin Board, which is based on the Skipchain architecture. The data on the Bulletin Board is cryptographically tied to the Bitcoin blockchain through the Cotena layer, which provides a high level of immutability and decentralization of the data mechanism. The Skipchain architecture provides a proactive Byzantine consesus mechanism. This architecture enables software clients to navigate arbitrarily long blockchain timelines both forward and backward, providing proof of transaction validity without the need for a full record of the blockchain.

The Cotena schema is a tamper-resistant logging mechanism built on top of the Bitcoin blockchain. Cotena was created to leverage the data security of the Bitcoin blockchain while introducing a design that has minimal data storage requirements and reduced Bitcoin transaction costs. Agora's voting process consists of six distinct steps:

1. Configuration: Election administrators create a new election event.
2. Casting: Voters cast their encrypted ballots to Agora's network.
3. Anonymization: Agora's network anonymizes all voter ballots.
4. Decryption: Agora's network decrypts the anonymized ballots.
5. Tallying: All votes are counted.
6. Auditing: Auditors post their reviews confirming validity of the election results.

3. LITERATURE SURVEY

3.1 Anonymous voting by two-round public discussion

Proposed an addition of a self-tallying function to the 2-Round Anonymous Veto Protocol (called AV-net). The AV-net provided exceptional efficiency compared to related techniques, the paper was focused on the dining cryptographers network (DC-net) and its weaknesses and proposed the AV-net as a new way to tackle that problem. The new protocol, like the AV-net requires no trusted third party or private channel. Participants execute the protocol by sending two-round public messages, but is significantly more efficient in terms of the number of rounds, computational cost and bandwidth usage.

In general, the new protocol divided electronic voting into to two classes:

1) Decentralized elections where the protocol is essentially run by the voters.

2) Centralized elections where trusted authorities are employed to administer the process.

The protocol proposed was focused on the first class, where strong voter privacy was the primary objective which had two challenges. First challenge was that there exists no trusted third party. With a trusted third party, many security problems can be easily solved, but could lead to the 'trusted' third party to become the one who breaks the security policy. The goal therefore was to eliminate the use of a trusted third party altogether. The second challenge was that there would be no voter-to-voter private channels to ensure dispute freeness, i.e everybody could check whether all voters had followed the protocol faithfully.

3.2 A Secure and Optimally efficient Multi-Authority Election Scheme

Proposed a multi-authority secret-ballot election scheme which would guarantee privacy, universal verifiability and robustness, where voters would participate using a PC, where the main consideration is the effort required of a voter. In this model, voters cast their vote by posting ballots to a bulletin board. The bulletin board works as a broadcast channel with memory to the extent that any party can access its content but no party can erase anything from the bulletin board. The ballot does not reveal any information on the vote itself but is ensured by an accompanying proof that the ballot contains a valid vote.

The final tally, the sum of all votes, which occurs when the deadline is reached, can then be obtained and verified, by any observer, against the product of all submitted ballots. Which would ensure universal verifiability, due to the homomorphic properties of the encryption method used. While this proposal can scale up to large elections better than the previous ones, it does have limitations.

3.3 A Smart Contract for Boardroom Voting with Maximum Voter Privacy

Proposed the first implementation of a decentralized and self-tallying internet voting protocol with maximum voter privacy using the Blockchain, called The Open Vote Network (OVN). The OVN is written as a smart contract for the Ethereum blockchain. In its general idea the OVN is an implementation of the Anonymous voting by two-round public discussion we previously discussed. The creators of the OVN came to the conclusion after implementing the system, that the cost of running such system on the Ethereum blockchain was 0.73$ per voter.

The safe upper limit of voters was 50 voters, but the cost could be considered reasonable as it provided maximum voter privacy and is publicly verifiable. The limitation of number of voters was recommended because of the gas limit on the public Ethereum blockchain. By examining their research paper, the limitations of the previous protocol are

unchanged. The OVN does not provide any coercion resistance with the public verifiability in the way that the voting is conducted in an unsupervised environment, i.e. the coercer can stand over the shoulder of the voter. The OVN is also vulnerable to denial of service attack because it is implemented on the Ethereum blockchain, which has had numerous DOS attacks through its lifespan. The Ethereum blockchain could also be throttled by major traffic of transactions at the time of an election, which could delay the voting process immensely.

The implementation could therefore be optimal for small boardroom voting, with the downside of having each individual voter downloading the full Ethereum blockchain to confirm the voting protocol is being executed correctly.

3.4 Remote E-voting protocols using cryptographic tools

A trusted third party (TTP) is involved to make e-voting systems more easily implemented and controlled. However, a powerful TTP may also become the vulnerable spot of the whole system. A few efforts have been made to combine an e-voting protocol with the blockchain paradigm to design a voting protocol without a TTP, which provides anonymity and verifiability as well.

Proposed a voting protocol, which introduces a reward/penalty scheme for correct or incorrect behaviors of voters. Later proposed an e-voting protocol, which involves a TTP into blockchain to preserve voters' choices. This protocol divides the organizer of elections into two different parts - the Authentication Server (AS) and the Token Distribution Server (TDS), to protect voters' privacy. However, there remain some problems in this protocol, for example, it is difficult to inspect these two parts' behaviors, and it limits the extension of the voting scheme

3.5 Estonian I-Voting System

Estonia proposed first where the citizens were able to cast their vote using only the Internet and an electronic national identification card. The ID card used in the elections was designed to run on an integrated circuit, a chip Java chip platform and protected with 2048 bit PIN. The card is able to create signatures using SHA1/SHA2. The card is easily usable for authentication, encryption, and signatures. The voter has to download the voting application, authenticate using the electronic ID, the list of the candidates will be displayed and a vote could be cast. The vote will be encrypted using the election's public key and signed with the voter private key. As soon as the vote is cast it will be sent to a vote storage server controlled by the Estonian government. Voters could vote multiple times, and only the last vote will be considered valid. This is done to prevent vote buying.

3.6 New South Wales iVote System:

All eligible citizens placed their vote using iVote system in the New South Wales State election. Vote was developed by *Scytl* as wellbut. To cast a vote, citizens have to undergo four steps, which of two are optional:

1. The voter has to register with authorities, receive a voter ID and choose a six digit PIN.
2. The voter logins in the system using his ID and PIN, cast a vote, then receives a 12-digit receipt number as a confirmation.
3. The voter enters his ID, PIN, and receipt number to verify that his vote went through. This step is optional.
4. After the election is over, the voter is still can use his 12-digit receipt to check if his vote was included in the final count. If the vote was not counted a reason will be displayed. This is an optional step as well.

4. PROPOSED METHOD

Blockchain based E-voting system

The proposed E-voting system will be designed to support a voting application in the real-world environment taking into account specific requirements such as privacy, eligibility, convenience, receipt freeness and verifiability. The blockchain is an open and distributed ledger. It uses an append-only data structure, meaning new transactions and data can be added on to a blockchain, but past data cannot be erased. The first transaction added to the block will be a special transaction that represents the first candidate. When this transaction is created it will include the candidate's name and will serve as the foundation block, with every vote for that specific candidate placed on top of it. Unlike the other transactions, the foundation will not count as a vote, and it will only contain the name of the candidate. Every time a person votes, the transaction gets will be recorded and the blockchain will be updated. To ensure that the system is secure, the block will contain the previous voter's information. If any of the blocks were compromised, then it would be easy to find out since all blocks are connected to each other. The blockchain is decentralized and cannot be corrupted, no single point of failure exists. The blockchain is where the actual voting takes place. The user's vote gets sent to one of the nodes on the system, and the node then adds the vote to the blockchain. The voting system will have a node in each district to ensure the system is decentralized.

Technology stack:

The technologies that are used in building the project are explained in this section.

1. Ethereum

Ethereum is an open source, public, blockchain-based distributed computing platform and operating system featuring smart contract (scripting) functionality. Ethereum blockchain allows us to execute code with the Ethereum Virtual Machine (EVM) on the blockchain with the use of smart contract.

2. Smart contracts

Smart contracts are where the business logic of our application lives and one can code decentralized portion the application. A "smart contract" is simply a piece of code that is running on Ethereum. It's called a "contract" because code that runs on Ethereum can control valuable things like ether or other digital assets. Smart contracts are in charge of reading and writing data to the blockchain. Smart contacts are written in Solidity programming language.

3. Node Package Manager (NPM)

The first requirement is Node Package Manager, or NPM, which comes with Node.js. Node.js is an open source server environment which uses JavaScript on the server

4. Truffle Framework

The next dependency is the Truffle Framework, which allows us to build decentralized applications on the Ethereum blockchain. It provides a set of tools that will allow one to write smart contacts with the Solidity programming language. It is also used to test smart contracts and deploy them to the blockchain. It also provides a place to develop our client-side application.

5. Ganache

Ganache, is a local in-memory blockchain. It will give us 10 external accounts with addresses on our local Ethereum blockchain where each account is preloaded with 100 fake ether.

6. Metamask

Metamask is an extension for Google Chrome .In order to use the blockchain, one must connect to it (as blockchain is a network). Metamask is a special browser extension in order to use the Ethereum block chain. One can connect to the local Ethereum blockchain with their personal account, and interact with our smart contract.

7. Sublime text

Sublime Text is a text editor, and can download "Ethereum" package which provides nice syntax highlighting for Solidity.

5. EXPECTED OUTCOMES

- As there are some limitations in existing systems, our proposed system will provide some benefits over the existing system that will help much to provide efficiency in the e-voting system.
- Every eligible voter will be able to cast vote from anywhere. An eligible voter will not be able to vote twice.
- The system will reduce the use of papers and EVM machines. So it will gradually reduce the cost of elections. The system will provide a high level of security.
- The system will reduce the risk of hacking and vote tampering. The system will ensure more voter participation.
- It will help in saving the time which can be invested in other works.

6. CONCLUSION

The E-voting system is been used in many vacillating forms since 1970s. With the remarkable growth in the field of Blockchain technology, a number of initiatives have shown that using blockchain can aid to an effective solution to e-voting system, which provide authenticity, integrity, verifiability, anonymity and general consensus from every voter. This work has presented one such effort which leverages benefits of Blockchain such as decentralization, immutability, security and transparency.

REFERENCES

[1] S. Mori, C.Y. Suen and K. Kamamoto, "Historical review of OCR research and development," *Proc. of IEEE*, vol. 80, pp. 1029-1058, July 1992.

[2] J.Pradeep, E.Srinivasan, S.Himavathi, "Neural Network based Handwritten Character Recognition system without feature extraction". International Conference on Computer, Communication and Electrical Technology – ICCCET 2011, 18th & 19th March, 2011

[3] N. Arica and F. Yarman-Vural, "An Overview of Character Recognition Focused on Off-line Handwriting", *IEEE Transactions on Systems, Man, And Cybernetics*, Part C: Applications and Reviews, Vol.31 (2), pp. 216- 233. 2001.

[4] V.K. Govindan and A.P. Shivaprasad, "Character Recognition – A review," *Pattern Recognition,* Vol. 23, no. 7, pp. 671- 683, 1990.

[5] Cheriet, M., Kharma, N., Liu, C.-L., Suen, C.: Character Recognition Systems: a guide for students and practioners. Wiley-Interscience, Hoboken (2007)

[6] MATHCrossRefGoogle Scholar Senior, A.W., Robinson, A.J.: An Off-Line Cursive Handwriting Recognition System. IEEE Transactions on pattern analysis and machine intelligence 20, 309–320 (1996)

[7] CrossRefGoogle Scholar Haralick, R.M.: Statistical and structural approaches to Texture. IEEE Proceedings 67, 786–804 (1979)
[8] CrossRefGoogle Scholar Sabeenian, R.S., Palanisamy, V.: Rotation Invariant Texture Characterization and Classification using Radon and Wavelet Transform. Published in the International Journal of Computational Intelligence and Health Care Informatics 1(2), 95–100 (2008)
[9] P.C. Woodland, J.J. Odell, V.V. Valtchev, and S.J. Young, "Large Vocabulary Continuous Speech Recognition Using HTK," Proc. IEEE Int'l Conf. Acoustics, Speech, and Signal Processing, vol. 2, pp. 125–128, Apr. 1994.
[10] J.B. Bellegarda, D. Nahamoo, K.S. Nathan, and E.J. Bellegarda, "Supervised Hidden Markov Modeling for On-Line Handwriting Recognition," Int'l Conf. Acoustics, Speech and Signal Processing, vol. 5, pp. 149–152, 1994.
[11] H. Bourlard and N. Morgan, Connectionist Speech Recognition: A Hybrid Approach. Kluwer, 1993.
[12] M. Schenkel, I. Guyon, and D. Henderson, "On-Line Cursive Script Recognition Using Time Delay Neural Networks and Hidden Markov Models," Int'l Conf. Acoustics, Speech and Signal Processing, vol. 2, pp. 637–640, 1994.