IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661,p-ISSN: 2278-8727, Volume 22, Issue 4, Ser. I (Jul. – Aug. 2020), PP 52-57
www.iosrjournals.org

Secure and Decentralized Smart Elections

Rajat Gupta, Bipul Jha, Atul Kumar Shukla, Aryan Raj, and
Dr. Shabana Sultana

Department of Computer Science and Engineering at National Institute of Engineering, Mysuru, KA 570008 India.
Professor in the Department of Computer Science and Engineering at National Institute of Engineering, Mysuru, KA 570008 India.

Abstract
This main aim of this paper is to develop a Secure and decentralised smart election system using blockchain technology.
Over the last few years the traditional election processes have come under a lot of scrutiny whether they are secure or not. They are not fully secure as it is easy to attack the places where elections are being held which is called 'booth capturing'. It also threatens privacy of voters. In recent years the attack on the current election process has been widespread and there is demand to make it more secure and more transparent.
For decades Electronic voting systems have been a topic of interest, as there is a need for alternative processes to carry out an election. The main goal is to minimize the cost of conducting an election, while maintaining the election standards with respect to security and privacy, while having the voters an option of voting from any place, with any device.
However, there remain challenges to achieve widespread adoption of such systems especially with respect to protecting them from any potential faults,reducing election frauds and attacks on election results.
Index Terms: *Blockchain, Smart contract, Ethereum, Solidity, Decentralized System*

Date of Submission: 14-07-2020 Date of Acceptance: 29-07-2020

I. Introduction

Blockchain can be expressed as a distributed data structure that is replicated and allocated among the members of a network, whose main task is to make a record of every transaction carried out. Each transaction is divided into timestamped blocks and each block is described by its cryptographic hash. Each block contains the hash value of its previous block which creates a chain of blocks which are immutable. A blockchain can also be defined as a linked list, which is linked by hash pointers, where each block contains a hash pointer that stores a cryptographic hash of the header of the previous block in the chain. If the content of a block in the chain is modified, the hash pointer stored on the succeeding block will also change as well as all the subsequent blocks in the chain. Introducing blockchain technology in the voting system will prove to be a major step forward in ensuring accurate voting processes. However, the fundamental concern behind this would be to gain the trust of people and how they react to a completely new election methodology, especially given that adoption of the Blockchain technology is still at its budding stages.

Introducing blockchain is transforming the way organizations do business, and the election industry is no exception. A blockchain-based voting application is more secure because any hacker with access to the terminal will not be able to affect other nodes as if any change happens in one node, then the change will be reflected in other nodes also, so nodes can automatically know that someone has altered the data. Each voter is given a public key and a private key. So, when any voter casts their vote, the identity of the voter is safe as others can only see the public key. This also ensures that officials can count votes with absolute certainty, knowing that each ID can be accredited to one vote. It also removes the possibility of creating fake ones. The whole system revolves around smart contracts. Smart contracts are nothing but pieces of code which controls all the transactions happening in the blockchain. All the transaction records are saved in networks instead of databases which makes it incorruptible. Thanks to decentralization and encryption properties of blockchain, its database transactions are incorruptible, and each record is easily verifiable. The blockchain network cannot be influenced or taken down by a single party because it doesn't exist in one place.

II. Related Work

Blockchain-Enabled E-Voting(Kshetri, N., & Voas, J. (2018). Blockchain-Enabled E-Voting. IEEE Software) Voas tested his system in getting results for student government elections; political party events and other community elections.

A Comparative Analysis on E-Voting System Using Blockchain(Garg, K., Saraswat, P., Bisht, S., Aggarwal, S. K., Kothuri, S. K., & Gupta, S. (2019). A Comparative Analysis on E-Voting System Using Blockchain.)
This paper covers all the problems faced during implementation of the voting system based on blockchain technology. It tackles all the problems that are faced during implementation and ways to avoid it.

The first country in the world which conducted a Blockchain-based voting system was Sierra Leone. In that election votes were stored in an immutable distributed ledger which gave instant access to voting results.

III. Concept

A. ETHEREUM BLOCKCHAIN

The Ethereum blockchain is a public, and open source decentralized blockchain based computing platform and operating system which provides us the functionality of smart contracts. It is capable of supporting a modified version of consensus via transaction based on the transitions of the state. Gas is a unit used to charge for the transactions which is used to correct spam and allocate various devices in the network.

B. SMART CONTRACTS

Smart Contract is used to ease, verify, or carry out the mediation of a contract. These transactions are irreparable and traceable.Smart contracts are contracts which execute on their own, with the terms agreed by both buyer and seller being written in the form of code.

The contract exists across the distributed blockchain network. It only permits truthful transactions and agreements to be carried out without the need for an integral. They provide transactions which are irreparable, traceable and open to all.

The smart contract developers claim that the contracts can be written in such a way that it can be partially or fully automating, self-enforcing or both. Its main purpose is to provide security that is an upgrade to previous contract rules and to bring down other transaction costs related to it. Many types of contracts have been developed by various cryptocurrencies.

Smart contracts have the ability to automate processes and efficient operations. Nowadays it is mostly used in financial services. Although it can also be used in many business activities, voting purposes, auctions etc.

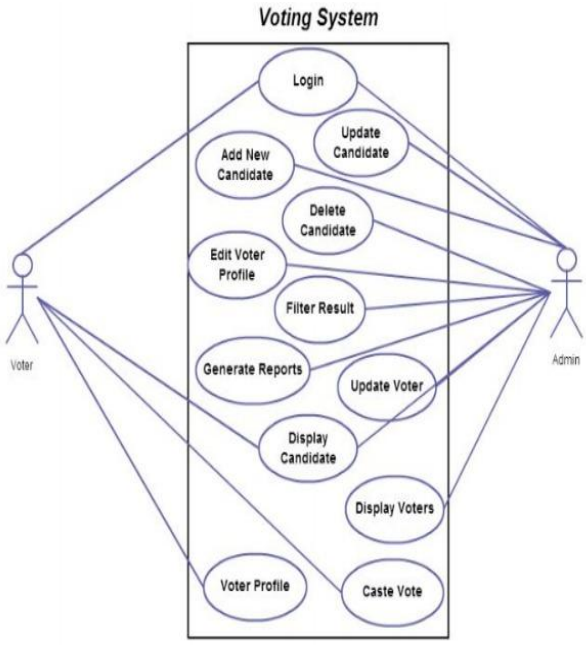

Fig. 1. Use case diagram.

In the above use case diagram, we can see that the admin has the right to add new candidates, update candidates, delete candidates, and edit voter's profiles. Apart from these it has to generate reports, filter results, display candidates, display results as well as display voters. Whereas a voter has to login into his account, has to check candidates standing in the election and has to cast his/her vote to one of the candidates standing in the election. Voters will also have access to view his/her profile.

BLOCKCHAIN FOR SMART ELECTIONS

The above section proposes a new voting system using Blockchain and the tools we need to implement our smart contracts on the Ethereum network.

A. Building Smart Contract

We have built smart contracts through Solidity language. Here we need to create an Election contract which defines the structure of the candidate to hold its name, vote count and an ID. So a candidate has an id, name and vote count as its attribute. Next we create a mapping variable to store the account address which has voted. Then we write a function to vote which makes sure that the voter hasn't voted before and validate that the candidate is legal and if this is the case then we record the vote of the voter and increment the vote count of the candidate for which a user has voted. This the core logic of the contract which enables a legal voting. Then the contract is compiled using the Ethereum virtual machine. Any node in the network can access this smart contract and initiate the voting.

B. Local Ethereum Network using Ganache

Ganache is a framework provided by truffle and is used as a local blockchain for Ethereum development by developers which is used for deploying our contracts, making applications, and carrying out tests. Ganache creates a server which is local as well as the accounts with their personal keys. Accounts provided by ganache were used to test our work. The test was to ensure that each vote is counted once, one voter can cast only once and the result is being calculated properly.

C. Deploy Contracts to Local Ethereum Network

Truffle helps us to compile our smart contracts which are built using solidity language and which has the business logic in it and deploy the contract to the local Ethereum network. These contracts are accessible to all the systems in the local network. Any nodes can perform the transaction if it satisfies the contract rules. Once the contracts are deployed in the network, it enables users to interact with it and carry on transactions to bring changes in the state of the contract.

D. Connect our web application to Smart Contracts to carry out transactions

Node js web application is used to give the GUI to communicate with our contract which has been deployed. To do this we are using the Web3 library. Web3 is an Ethereum JavaScript API which using a HTTP connection allows us to connect to local or remote Ethereum nodes. So the users have a nice interface where they can choose their desired candidate and vote for them. They can also see the current vote count of all the candidates in the election.

E. Use Metamask plugin through the browser for selecting locally available networks created by Ganache and do the voting.

Metamask helps in connecting our decentralized web application to connect to the local Ethereum network and also provides us with wallet needed for transactions. First the account is set in metamask by the accounts which are provided by the ganache which gives you ten accounts, each account has a private key which is used to set the account in the metamask. Now when you enter the web application through the web browser on your machine, the metamask account is selected and then you can make a transaction using that account. When you click on the vote button, a prompt will pop up and you need to confirm that transaction by clicking on the confirm button, this will add to the vote count of the selected candidate. And you can see the number of votes of the candidate which you have voted for being increased at the same time. This makes sure that your vote is being counted.

F. Epoch timer

Epoch timer is used to set the end time for the election after which the contract would not allow any voter to vote in the election. This timer is defined in the contract, and in the function for voting we have added a requirement that the time at which the voter votes should be lesser than the end time defined in the contract. By this way we restrict the time duration during which voting can be done by the voters. A timeout function is also

defined which calls the display result function after the set time which displays the result of election to the users.

These are the list of important elements used to create this tool:
1. Chrome extension: Metamask
2. Framework: Truffle
3. Server Environment: NodeJS
4. Library: Web3.js
5. Ethereum Blockchain Simulator: Ganache

PROTOTYPE

Smart Contracts are written in solidity which has the logic that voters can vote only for one candidate, and for only one time within a given timeframe. Then these contracts are compiled and deployed to the local Ethereum network where they are compiled using EVM (Ethereum virtual machine). Simple node js application is made which provides us the interface for the voters to vote and it connects to the contracts.

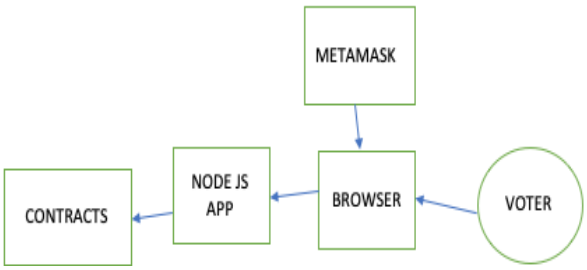

Fig. 2. System Flow

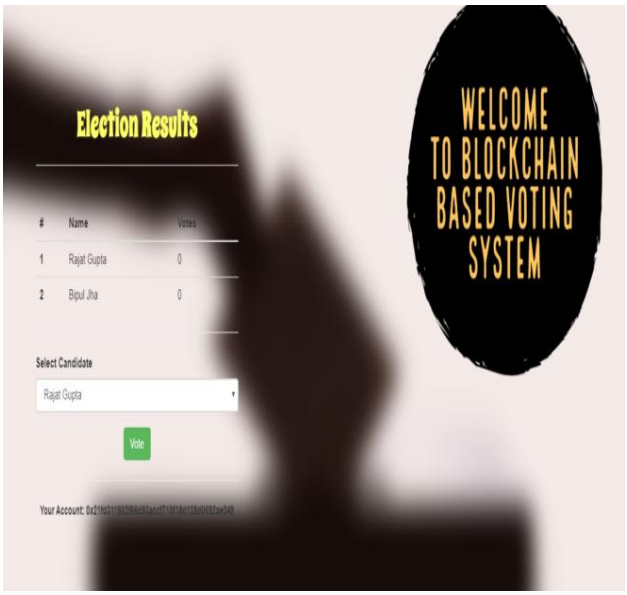

Fig. 3. Voting Page.

The voters can open this web app to vote for their favorable candidates. Voters can select the candidates using the dropdown. The public key of the voter is displayed below the vote button.

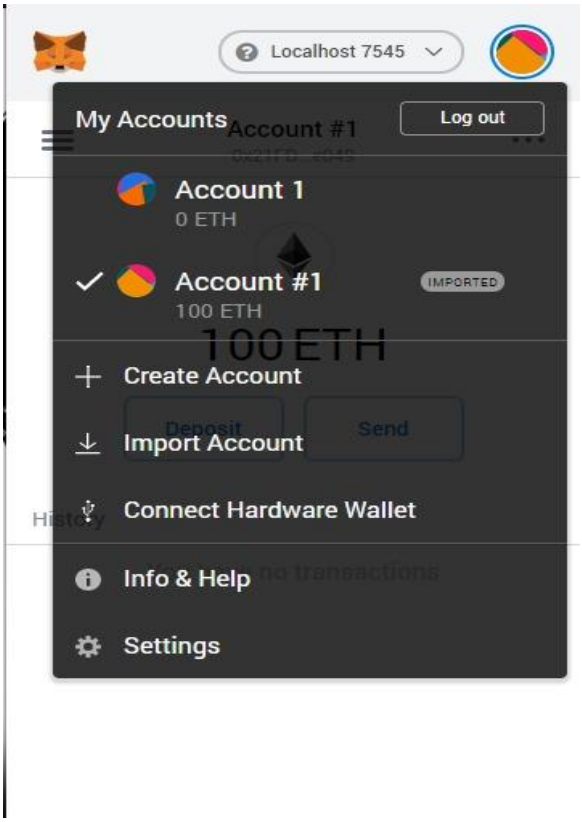

Fig. 4. Metamask extension.

This tool allows the voters to connect to the Ethereum network to cast their vote. Here the private key of the voter can be added to create the account for voting. The voter can cast their vote within a timeframe, if he/she fails to cast the vote within that timeframe and tries to vote then this tool gives an error message indicating "insufficient funds".

IV. Result And Discussions

So, this application helps in maintaining the immutability and security aspect. Any valid user having the private key can vote for only one candidate, only once from any browser using this decentralized application. This helps the user to vote from anywhere and it is not possible to tamper the results as the values are not stored in the central database. Every node in the network has the result stored in it. Also, the timer concept implemented in the smart contract helps close the voting process and declare the winner. The timer concept uses an epoch time system which we can manually set in our codebase, and depending on the scale of the election we can set it for any duration. This application needs to be implemented on a bigger scale because according to a recent report published by Election commission of India, around 300 million voters were not able to vote as they were not present at the place where they were registered as voters. So, if this work is implemented it will reduce government's expenditure as well as voters' time. To make this work successful we need to make sure that voters are fully aware of this technology and it is the duty of the government to make the voters have all the answers to their query and explain the user interface. To make it more scalable, the government can think of making both web as well as mobile apps.

V. Conclusion

This proposed work takes benefit of the transparency of smart contracts to help all voters to join in both the recording and validation of ballots. It helps in enhancing the voters' confidence and reduces the waste of election resources. We have shown that through blockchain technology we can ensure a secure and a transparent election. It would be safe to say that blockchain in voting processes has the potential to rebuild a robust democracy. At the same moment, the different considerations that come along with featuring a new technology in matters of utmost public and governmental concern need to be taken care of. Rather than this, educating the masses and also the public officials on this new technology is highly imperative, so as to mitigate the doubt that inevitably follows an innovation.

References

[1]. Blockchain-based e-Vote-as-a-Service, Emanuele Bellini1 IEEE Member, Paolo Ceravolo2 IEEE Member, Ernesto Damiani IEEE Member,2019 IEEE 12th International Conference on Cloud Computing (CLOUD)

[2]. Blockchain-Based E-Voting System, Friðrik Þ. Hjálmarsson, Gunnlaugur K. Hreiðarsson, Mohammad Hamdaqa, Gísli Hjálmtýsson School of Computer Science Reykjavik University, Iceland,2018 IEEE 11th International Conference on Cloud Computing.

[3]. Securing E-Voting Based On Blockchain in P2P Network, Haibo Yi, EURASIP Journal on Wireless Communications and Networking 2019, Springer.

[4]. Secure Electronic Voting System Using Blockchain Technology , Ashish Singh, Kakali Chatterjee, International Conference on Computing, Power and Communication Technologies 2018, IEEE.

[5]. Votereum: An Ethereum-Based E-Voting System, Linh Vo-Cao- Thuy, Khoi Cao-Minh , Chuong Dang-Le-Bao, Tuan A. Nguyen, IEEE-RIVF International Conference on Computing and Communication Technologies 2019.

[6]. Reinforcing the links of the blockchain, Angelos Stavrou, Professor, Computer Science Department, George Mason University, 2017, IEEE..

[7]. A Proposal of Blockchain-Based Electronic Voting System, Cosmos Krisna Adiputra, Rikard Hjort, Hiroyuki Sato, Second World Conference on Smart Trends in Systems, Security and Sustainability 2018, IEEE.

[8]. VoteChain: A Blockchain Based E-Voting System, Archit Pandey, Mohit Bhasi, K. Chandrasekaran, Global Conference for Advancement in Technology (GCAT) 2019, IEEE.

[9]. The Basics of Bitcoins and Blockchains by Antony Lewis.

[10]. Mastering Bitcoin: Programming the Open Blockchain (Second Edition) by Andreas Antonopoulos.

[11]. Solidity Official Docs Website - https://solidity.readthedocs.io/en/v0.6.6/

[12]. Node.js Official Docs Website - https://nodejs.org/en/docs/

[13]. Epoch Official Website - https://www.epochconverter.com/

[14]. Truffle Official Website - https://www.trufflesuite.com/ganache

[15]. Ethereum Official Website - https://ethereum.org/

Rajat Gupta, et. al. “Secure and Decentralized Smart Elections.” *IOSR Journal of Computer Engineering (IOSR-JCE)*, 22(4), 2020, pp. 52-57.