International Journal of Research in Engineering, Science and Management
Volume-2, Issue-5, May-2019
www.ijresm.com | ISSN (Online): 2581-5792

Blockchain Voting System

K. Raghavendra[1], R. Mithuna[2], M. Manjunath[3], Hanumanth Raju[4]
[1,2,3]*Student, Dept. of Computer Science and Engg., M. S. Ramaiah Institute of Technology, Bangalore, India*
[4]*Assistant Professor, Dept. of Computer Science and Engg., M. S. Ramaiah Inst. of Tech., Bangalore, India*

***Abstract*: The decentralized web is basically based on blockchain technology that provides faster data transfer, less node failure and high security of data. Building a secure electronic voting system that offers the fairness and privacy of current voting schemes, while providing the transparency and flexibility offered by electronic systems has been a challenge for a long time. In this work-in-progress paper, we evaluate an application of blockchain as a service to implement distributed electronic voting systems. The paper proposes a novel electronic voting system based on blockchain that addresses some of the limitations in existing systems and evaluates some of the popular blockchain frameworks for the purpose of constructing a blockchain-based e-voting system. In particular, we evaluate the potential of distributed ledger technologies through the description of a case study; namely, the process of an election, and the implementation of a blockchain- based application, which improves the security and decreases the cost of hosting a nationwide election.**

***Keywords*: Blockchain, Ganache-cli.**

1. Introduction

Large sections of society today do not trust their government . This makes the election a very important event in a modern democracy. The issue with the current ballot system is that it can be easily manipulated by power hungry organizations . The proposed system looks to eliminate the aspect of trust from an election to make it more secure and transparent.

The system uses existing technology such as a client server architecture integrated with a blockchain system to ensure aspects such as transparency, security and auditability are achieved without sacrificing privacy for voters. The cost of building the system is substantially less as compared to the cost of running a ballot based system. There are substantial social benefits to using the system as well such an easier and quicker voting process which will lead to higher voter turnout. This system can be implemented for a larger number of countries as the internet penetration in the world increases. We might definitely see a future where every country has implemented a system similar to ours.

2. Background

Think of Bitcoin as the rst application and Blockchain as an Operating System. It is important to realize that Bitcoin is just one of the seven hundred applications of the underlying technology, Blockchain. The evolution of Blockchain has been disruptive. Lets take an example to prove that. We used to communicate with people verbally, which can be termed as the verbal communication era. Telecommunications enabled people to have synchronous communication with other people around the world. The internet allowed people to have scalable asynchronous communication around the world. Thus, the internet we use today is the internet of information. And the Blockchain is enabling the internet of value. The way we can transmit information through internet, we can transmit value on a peer to peer basis. This can be termed as the Blockchain era. The way currency was digitized on the Blockchain network, other important assets can be digitized on the network as well. A new global economy of immediate value of trans- fer is created, where big intermediaries no longer play a major role. This shift from traditional transactions where trust played a major role to transactions governed by complex computer codeis noticeable. One such application is the development of Ethereum public blockchain, which provides a way to execute peer to peer contracts.

A. Current Scope

The utilization of the grouping proposed in the blockchain creation process in this framework considers that in a constituent framework not required for mining as in the Bitcoin framework in light of the fact that the voter information and numbers are clear and are not permitted to choose more than once, the proposed succession guarantees that all hubs Which is legitimately associated and can maintain a strategic distance from impact in transportation. Likewise ensure all hubs that have enrolled the outcomes are incorporated into the figuring procedure. As far as expense can likewise be more effective on the grounds that it doesn't require hardware that is dependably changed in every race did. In light of the structure and the consequences of research led, it very well may be presumed that the framework is effective usefulness of chronicle the e-casting a ballot framework dependent on Blockchain innovation. A blockchain is a common, appropriated, and permanent record. The blockchain has produced serious enthusiasm for use in an assortment of enterprises and areas, going from saving money, back, and protection to medicinal services, government, retailing, and assembling. Associations are utilizing blockchains to grow new applications that are more solid and proficient. In this paper we investigate the blockchain innovation for building PC data frameworks. We first direct a methodical investigation of uses and issues identified with blockchain innovation, and after that distinguish a few issues

that require additionally look into with the end goal to be legitimately tended to. We additionally examine the potential use of blockchain in instruction.

3. Implementation

A. *Tools Introduction*

A tool is an object used to extend the ability of an individual to modify features of the surrounding environment. Although many animals use simple tools, only human beings, whose use of stone tools dates back hundreds of millennia, use tools to make other tools. The set of tools needed to perform different tasks that are part of the same activity is called gear or equipment.

While one may apply the term tool loosely to many things that are means to an end (e.g., a fork), strictly speaking an object is a tool only if, besides being constructed to be held, it is also made of a material that allows its user to apply to it various degrees of force. If repeated use wears part of the tool down (like a knife blade), it may be possible to restore it; if it wears the tool out or breaks it, the tool must be replaced. Thus tool falls under the taxonomic category implement, and is on the same taxonomic rank as instrument, utensil, device,

B. *Technology Introduction*

Technology is the collection of techniques, skills, methods, and processes used in the production of goods or services or in the accomplishment of objectives, such as scientific investigation. Technology can be the knowledge of techniques, processes, and the like, or it can be embedded in machines to allow for operation without detailed knowledge of their workings. Systems (e. g. machines) applying technology by taking an input, changing it according to the system's use, and then producing an outcome are referred to as technology systems or technological systems.

The simplest form of technology is the development and use of basic tools. The prehistoric discovery of how to control fire and the later Neolithic Revolution increased the available sources of food, and the invention of the wheel helped humans to travel in and control their environment. Developments in historic times, including the printing press, the telephone, and the Internet, have lessened physical barriers to communication and allowed humans to interact freely on a global scale.

Overall view of the project in terms of implementation

C. *Blockchain DApps*

As from the on top of rationalization it would be thought of as that it's not necessary for a dApp to be blockchain or crypto based mostly, it may be p2p affiliation however victimization the blockchain technology as its own benefits

For AN application to be thought of a dApp within the context of Blockchain, it should meet the subsequent criteria:

1. Application should be utterly ASCII text file
2. It should operate autonomously, and with no entity dominant the bulk of its tokens. the appliance might adapt its protocol in response to projected enhancements and market feedback, however the accord of its users should decide all changes.
3. Application's information and records of operation should be cryptographically hold on must be cryptographically hold on in an exceedingly public, suburbanised blockchain so as to avoid any central points of failure.
4. Application should use a cryptologic token (Bitcoin or a token native to its system) that is important for access to the appliance and any contribution important from (miners/farmers) ought to be rewarded with the application's tokens.
5. Application should generate tokens according to a regular cryptologic formula acting as a symptom of the worth, nodes square measure causative to the appliance (Bitcoin uses the Proof of labor Algorithm).

4. DAPP development process

The subsequent is that the common procedure for launching a d-apps,

A. *Whitepaper*

A whitepaper is printed describing the dApp and its options. This whitepaper will define the thought for dApp development however additionally entail an operating paradigm.

B. *Explanation of Algorithm and how it is been implemented*

Information about the implementation of Modules

Preceding the introduction to our voting system, it merits mentioning that the Ethereum protocol utilized as part of our system has not been modified in any way. Our system, BlockVote, uses existing functionality and features provided by Ethereum to provide the ability for creating and voting on ballots. Our implementation consists of three smart contracts coded in Ethereum's Solidity language, two scripts written in JavaScript, and one HTML page. BlockVote is an open source project and the entirety of the code isavailable for public use 6. We assume the administrator, creators, and voters have the MetaMask plugin downloaded in their browser or running an Ethereum node to create and manage Ethereum accounts as well as interact with our system. We utilize Ethereum's Web3 framework internally, this allows our users to easily manage signed transactions and interactions with the Ethereum blockchain. Using MetaMask and Web3 eliminates the need for users to download full or evenpartial Ethereum blockchains on their local machines in order to broadcast transactions. The only action required of users when registering, voting, or creating ballots is to use their passwords to unlock their Ethereum accounts in the MetaMask plugin and securely interact with the blockchain. If the user decides not to utilize the Metamask plugin then they are responsible for running a node on their local machine and syncing it with the blockchain to interact with our system using Web3.

A brief description of all the user parts of Block Vote

follows:

- Administrator is responsible for deploying the initial Registrar and Creator smart contracts. The 6https://goo.gl/nqBpzM administrator also has the ability to grant or revoke ballot creation permission for registered voters/ creators.
- Voter registers in our system with a valid student/ employee ID and e-mail address to vote on given ballot ID numbers.
- Creator is a voter with ballot creation permission. A brief description of the front/back-end pages implemented in Block Vote follows:
- VoteUI.html page is the user interface for our users. This page allows users to enter necessary information for each of the different use cases. Once the user enters the necessary information, the corresponding click buttons will invoke functions in App.js.
- VotingApp.js gathers information from VoteUI. html and interacts with Crypto.js and the Ethereum Blockchain. For each corresponding request from VoteUI.html, it utilizes eth.calls,
- Crypto.js server calls, and Ethereum transactions to verify, encrypt/decrypt votes, and store ballot/ vote information.
- Crypto.js acts as a cryptographic server. All votes are encrypted, homomorphically added, and decrypted using the Paillier homomorphic encryption system key pair in this server. A brief overview of the smart contracts implemented in Block Vote follows:
- Registrar.sol acts as the record and gate keeper. It keeps track of all registered voters and creators, ballot IDs, voting contract addresses, and whitelisted e-mail domains. As we can see in Figure 1, information regarding the voter and different ballots are linked together in the contract. This allows the contract to perform voter verification, permission modification, and Voting.sol address retrieval. The owner of this contract is the administrator.
- Creator.sol acts as a spawner for different Voting. sol contracts. The Creator defines the voting contract's details when filling out the required information in VoteUI.html. The owner of this contract is the administrator.

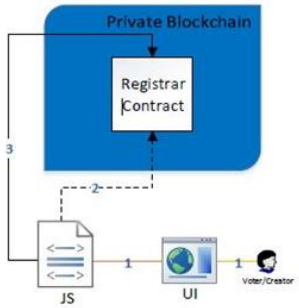

Fig. 1. Registrar contract

- Voting.sol acts as a virtual ballot and regulates the voting on the ballot. Another set of voter verification, that includes vote attempts and ballot time limit, is also conducted in this contract. As we can see in Figure 1, ballot title and the choice encrypted votes are also stored here so that we can retrieve at later stages. The owner of this contract is the contract's creator.

C. Initial Setup

The administrator is responsible for the initial deployment of both the Registrar and Creator contracts to activate the system and enable users to start registering, voting, and creating new voting contracts. When deploying the Registrar Contract, the administrator is also responsible for whitelisting a set of e-mail domains that are allowed to register to be part of the voting system.

D. Register Voter

BroncoVote was created for a university setting. Therefore, anyone with a student/employee ID number and an e-mail with the whitelisted domain is allowed to register as a voter. When the voter completes the ID and e-mail field in VoteUI.html, then the information is sent to VotingApp.js. As we can see in Figure 2, the VotingApp.js makes eth.calls to the registrar contract to verify the domain provided is part of the whitelist and if the user has previously registered If those checks are passed, then VotingApp.js sends a transaction to the registrar contract to store the new voter information, including the voter's ID, Ethereum address, and e-mail. It links the user's Ethereum address and e-mail address so that they cannot double register. Individuals can also request access to create ballots during the registration process; these requests are planned to be manually processed by the administrator but currently are granted automatically.

E. Create Ballot

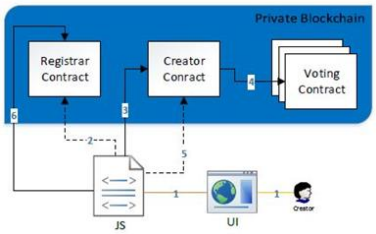

Fig. 2. Creator contract

If the user has permissions to create a ballot, the user is able to spawn a new voting contract by entering them required information in VoteUI.html. In order to create a ballot, the creator must provide their registered e-mail address then decide whether to create an election or poll, determine the title of the ballot, voting options, and number of votes allowed per voter. During this process, the creator can also elect to have a whitelisted ballot. If a whitelisted ballot is chosen, the creator enters the list of e-mail addresses allowed to vote on their ballot.

If the creator chooses to not make a whitelisted ballot, everyone with an e-mail address that has the whitelisted domain will be allowed to vote. Lastly, the creator sets the end date and time.

After submitting this information, VotingApp.js utilizes three eth.calls to verify the Creator and they are condensed into one step in Figure 3. VotingApp.js sends the first two eth.calls to the Registrar Contract to verify the creator by checking if their e-mail address is registered and if the request originates from the registered Ethereum address. If those two checks are passed, then VotingApp.js sends the third eth.call to determine if the user has permission to create a ballot.

Afterwards, if it was determined the user was allowed to create the ballot, VotingApp.js gathers the input data along with a randomly generated ballot ID number and sends a transaction to the Creator Contract with a request to create a new Voting Contract with the provided information. Once the new Voting Contract has been deployed, the contract's address is returned to the Creator Contract. VotingApp.js then sends another eth.call to the Creator Contract to retrieve the new Voting contract address and sends it as a transaction to the Registrar Contract to store the new ballot ID and contract address. The ballot ID is then displayed afterwards and the creator must write down this ballot ID and pass it along to all the voters in order to let voters vote on the ballot.

F. Load Ballot

Using the ballot ID provided by the Creator of the Voting Contract, a voter can check the results or vote on the ballot, provided the voting period has not passed. Once the voter enters the ballot ID in VoteUI.html, VotingApp.js sends an eth.call to the Registrar Contract to determine the validity of the ballot ID. If the ballot ID is valid, the voting options, title, and encrypted vote count for each choice if the voting period has ended unless it is a poll. If the ballot type was a poll then the results are displayed live. Before the vote count can be displayed, there is another step involved, as step 4, that involves sending the encrypted vote count to the Crypto.js server so that we can display the tallied vote for each choice on VoteUI.html.

G. Vote

Once the ballot has been loaded, the user can vote for a particular choice on the ballot with his/her registered e-mail address. When the voter clicks vote, VotingApp. js receives the information and sends eth.calls to the Registrar Contract to verify the voter, it checks the voter registration and Ethereum address. If the voter is verified, an eth.call is sent to the Voting Contract to check whether the ballot is whitelisted or not.

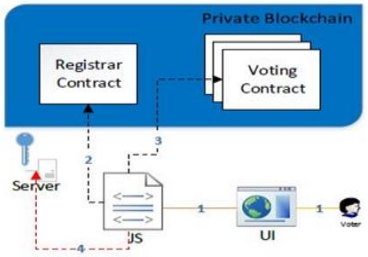

Fig. 3. Voting contract

5. Conclusions

E-voting, as discussed in the paper, is a potential solution to the lack of interest in voting amongst the young tech savvy population. For e-voting to become more open, transparent,

and independently auditable, a potential solution would be base it on blockchain technology. This paper explores the potential of the blockchain technology and its usefulness in the e-voting scheme. The paper proposes an e-voting scheme, which is then implemented. The implementation and related performance measurements are given in the paper along with the challenges presented by the blockchain platform to develop a complex application like e-voting. The paper highlights some shortcomings and presents two potential paths forward to improve the underlying platform (blockchain technology) to support e-voting and other similar applications.

Blockchain technology has a lot of promise; however, in its current state it might not reach its full potential. There needs to be concerted effort in the core blockchain technology research to improve is features and support for complex applications that can execute within the blockchain network.

References

[1] Gallup, "Trust in Government," Gallup, 30 September 2015. Available: http://www.gallup.com/poll/5392/trust-government.aspx.
[2] Wikipedia, "List of controversial elections," 20 September 2016. Available: https://en.wikipedia.org/wiki/List_of_controversial_elections.
[3] R. Skudnov, "Bitcoin Clients," Turku University of Applied Sciences, Turku, 2012.
[4] Affectiva, "Affective Product Overview," 15 January 2016.
[5] P. Noizat, "Blockchain Electronic Vote," in handbook of digital Currency, Paris, Elsevier Inc., 2015, pp. 453-461.