Blockchain based electronic voting system

Group #6

Venkata Praneeth Bavirisetty

Shiva Vamsi Gudivada

Venkatesh Kallepalli

Dinesh Reddy Kavalakuntala

Table of Contents

1. Abstract

Because of the properties such as transparency, decentralization, irreversibility, nonrepudiation, etc., blockchain is not only a fundamental technology of great interest, but also has large potential when integrated into many other areas. In this report, based on the blockchain technology, we propose a decentralized voting system, without the existence of a trusted third party. The concepts of blind signature and CoinJoin are used in implementing this. Furthermore, we provide several possible extensions and improvements that meet the requirements in some specific voting scenarios.

2. Introduction

Electronic voting (e-voting), which uses electronic systems to aid casting and counting votes in an election, has been a research topic of interest for the past few decades in cryptography. In comparison with the traditional paper-based voting, remote e-voting is environmentally friendly, real-time counting and processing, less error-prone. An e-Voting system must have heightened security in order make sure it is available to voters but protected against outside influences changing votes from being cast or keep a voter's ballot from being tampered with. E-voting can be views as special cases of secure multi-party computation. Because of the properties such as transparency, decentralization, irreversibility nonrepudiation, etc., blockchain has a large potential when integrated into many areas. This report outlines our idea of how blockchain technology could be used to implement a secure digital voting system.

3. Background

3.1 Conventional voting System

The voting systems used in most countries today are inefficient and outdated. In most cases, citizens must still personally visit polling stations and complete a ballot using manual, error-prone processes. Even when voters participate, there are often questions concerning the integrity of the election process that may cause the outcome to be questioned. Without a cryptographically secure architecture that allows voters to confirm that their own vote has been accurately recorded, current voting systems fail to satisfy their primary objective of relaying people's voices accurately.

Limitations of conventional voting systems are as follows:

- Voting takes place at specified locations.
- The entire process requires a lot of man power
- Skewed results
- Susceptible to software and hardware attacks
- The process is controlled by a centralized body, election commission
- Lack of post-election auditing

3.2 Blockchain

A blockchain is an ordered, back-linked list of blocks of transactions. Blocks are linked "back," each referring to the previous block in the chain. Each block has one single "previous block hash" field referencing its single parent inside the block header and thereby affects the current block's hash. This makes the blockchain immutable, which is a key feature for security.

3.3 Blind Signature

Blind Signature scheme [4] is a protocol where the person (say X) can obtain a signature of the other person (say Y) without showing the actual message written on it. The person Y can later verify whether the signature is genuine or not. This scheme can be used in the cases where the message authors and the signer are different.

3.4 Cryptocurrency Tumbler

Cryptocurrency tumbler or cryptocurrency mixing service is a service offered to mix potentially identifiable or 'tainted' cryptocurrency funds with others, to obscure the trail back to the fund's original source. Tumblers have arisen to improve the anonymity of cryptocurrencies, usually bitcoin (hence Bitcoin mixer), since the currencies provide a public ledger of all transactions

CoinJoin is an anonymization method for bitcoin transactions based on the following idea: “When you want to make a payment, find someone else who also wants to make a payment and make a joint payment together.” When making a joint payment, there is no way to relate input and outputs in one bitcoin transaction and thus the exact direction of money movement remains unknown to third parties.

4. Voting Process

The participants of an election are:

- **Voter:** A person who has the right to vote at the election
- **Candidate:** A person who is contesting in the election
- **Election Authority:** a body charged with overseeing the implementation of election procedures

Identical to a traditional voting system, blockchain based electronic voting system consists of five distinct phases:

4.1 Configuration

Prior to an election, all the participants of the election prepare for the event. Each voter choses two pairs of keys (sk1, pk1), (sk2, pk2) for the blockchain. The first public key is used to receive a crypto coin from the election authority. The second one is blind signed by the election authority and used in a CoinJoin transaction to achieve anonymity. Voters also choose a random number as a blinding factor (r) for blind signature. Each candidate contesting in the election choses a pair of public key and secret key. The election authority generates the public key (e) and secret key (d) required for RSA blind signature. The public key (e) chosen for RSA will be revealed. The election authority also generates a list of voters and candidates.

- **List of Voters:** It contains all eligible voters for the given election. The list is open to be viewed by any third party.
- **List of Candidates:** This list outlines the individual subjects and their public keys on which voters must decide.

Each voter sends his identity and an encrypted version of his second public key (pk2) to the election authority and reveals his first public key. Identity may be any valid proof issued by the government or the election authority. The public key (pk2) is modified by the voter using the RSA public key (e) and the blinding factor (r). The expression $m' \equiv m.r^e(mod\ N)$ is used for blinding the message (m) in RSA blind signature. In this scenario, pk2 is the message. The entire process is carried out online.

$$Voter\ (V) \xrightarrow[(Identıty, pk1, m')]{sends} Election\ Authority\ (EA)$$

4.2 Authorization and Anonymization

Verifying a voter is essential in establishing security within the system. Making sure that someone's identity isn't being misused for fraudulent purposes is important, especially when voting is considered, where every vote matters. The election authority verifies the identity of each voter. Once the voter's identity is confirmed as valid, the election authority sends a crypto coin to the voter's pk1. This coin is used by the voter to cast the vote. Additionally, the election authority signs the modified version of pk2 and returns it to the voter. The blind signature (S′) is computed from the expression $S' \equiv (m')^{d}(mod\ N)$, where d is the election authority's secret key for blind signature. This signature is used to distinguish between the transactions made by a voter and general audience, while counting. Only the transactions made from public keys with valid signature on them are considered as valid votes. The election authority updates the list of voters with pk1 of each voter and makes the list public.

$$Election\ Authority \xrightarrow[(S', coin)]{sends} Voter$$

The voter then unblinds the blind signature using the equation $S \equiv S'.r^{-1}(mod\ N)$, where r is the blinding factor used for blinding the message. At this stage, each valid voter has a crypto coin in pk1. But, this coin cannot be used to cast a vote, as it is not anonymous. The election authority who transferred the coin to the voters know the identity of each voter. Each public key can be linked back to voters. But, the election authority has no idea about the pk2. Moreover, each voter has a blind signature on these public keys. Every voter must transfer the coin provided by election authority from pk1 to pk2, to achieve anonymity. If a normal transaction is used to transfer this coin, it can be traced back because all the transactions in the blockchain are public. This is where the CoinJoin comes into picture.

A group of valid voters are pooled together to complete a CoinJoin transaction. The validity of a voter can be verified from the list of voters prepared by the election authority. All the coins will be shuffled by performing a CoinJoin transaction. Each CoinJoin transaction must have at least two valid voters. If not, their identity will be revealed while casting the vote. Figure4.2.1 illustrates a simple CoinJoin transaction.

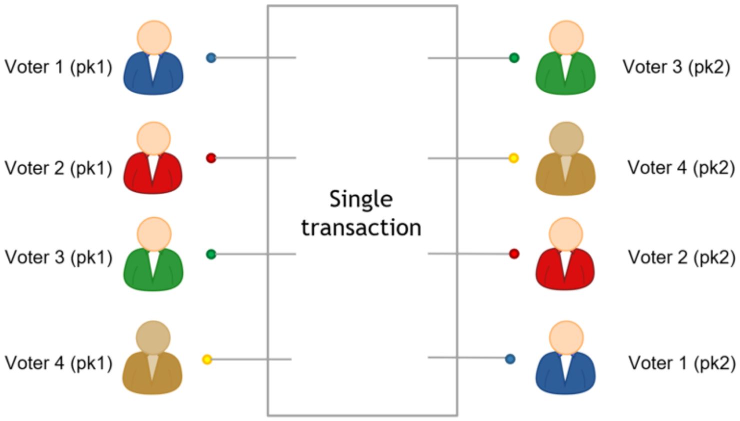

Figure 4.2.1 CoinJoin transaction

4.3 Casting

Voters should wait till every voter participates in a CoinJoin transaction before casting their vote, to achieve anonymization. On the day of election, each voter casts his vote by making a cryptocurrency transaction. This transaction should be made from voter's pk2 to the candidate's public key of their choice. This is a "Pay to PubKey hash" transaction from voters pk2 to the candidate's public key. Miners in the network verify the transactions and add them to the blockchain. The blind signature on this public key should also be included as a message in the transaction. Once the transaction is broadcasted, miners in the network will verify the transaction and add them to the blockchain. The blind signature included in the transaction will be used for tallying.

$$Voter \xrightarrow[(coin, S)]{sends} Candidate(C)$$

4.4 Counting Votes

The votes casted to a candidate can be counted by checking the transactions made to the candidate's public key. A vote casted will be valid only if the blind signature included in the transaction is valid. Anyone can verify the blind signature by calculating S^e which should be equal to m i.e., voters' pk2 in this scenario. $S^e \equiv m(mod\ N)$, where S is the unblinded signature on message m, which is also included in the transaction. After counting the votes, the election commission will officially declare the results.

4.5 Auditing

Auditing is the primary benefit of blockchain based voting system, as it achieves transparency. Anyone who doubts the results declared by the Election Authority can audit the election. Auditors can be election administrators, voters themselves or any third party. The results can be verified from the blockchain, as it is public to everyone.

4.6 Overview of blockchain based electronic voting system

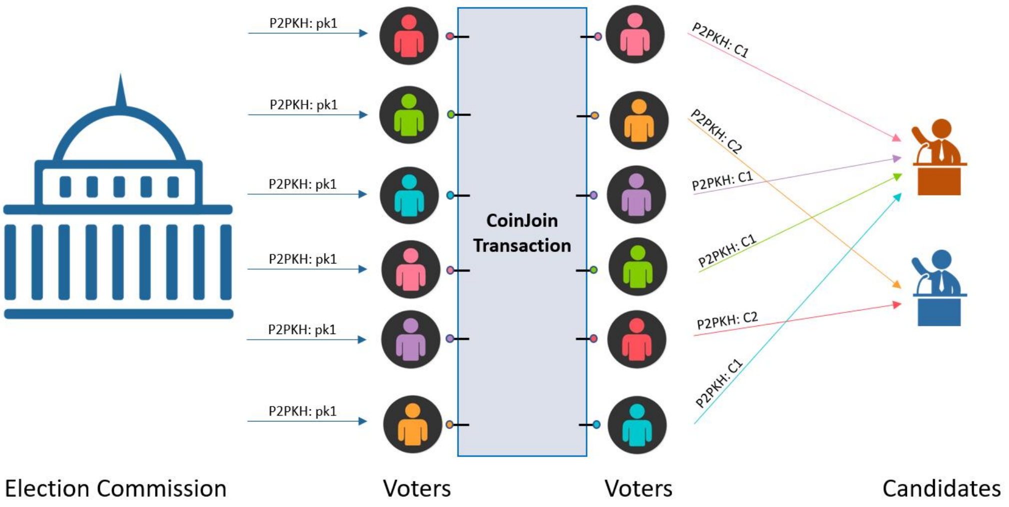

Figure 4.6.1: Overview of blockchain based electronic voting system

5. Drawbacks

- Blockchain technology is relatively new and unknown to general public
- Blockchain voting requires powerful and reliant internet infrastructure
- Blockchain scalability problem restricts this Voting System to small volumes
- A vote might be compromised if voter shares his/her secret key with a candidate
- If all the voters participating in a CoinJoin transaction vote for the same candidate, anonymity will be diminished

6. Conclusion

The proposed electronic voting system is based on the Blockchain technology. The system is decentralized and does not rely on trust. Any registered voter will have the ability to vote using any device connected to the Internet. The Blockchain will be publicly verifiable and distributed in a way that no one will be able to corrupt it. But, the blockchain scalability problem limits this voting system only to local election bodies where the number of voters is considerably small. It is practically impossible to use this system for general election in the most populous countries like China, India, USA and Brazil, unless the scalability problem is addressed.

7. Assessments

Implemented a simulation code for voting process in java, where some random voters are created for validation. The votes made by a test set of voters are recorded prior to the voting process, for verification. The results obtained from the blockchain based voting system are then compared with the previously recorded test set. programmed the simulation for different phases in the voting process and creating a CoinJoin transaction.

– *Venkata Praneeth Bavirisetty*

Implemented alternate consensus algorithm on Bearcatii blockchain, in Java programming language. In this algorithm, a transaction is accepted by a node in the network when at least one third of the nodes which it follows send the same transaction. Experimented the outcome of the algorithm when a small part of nodes in the network are malicious. Built a blockchain by arranging back-linked blocks in order using block header.

– *Shiva Vamsi Gudivada*

Explored for the various options to increase the anonymity in the entire voting process. Identified the phases in the voting process where anonymization is required, by considering the fact that there should not be any link between the voters' identity and ballots. Analyzed the various cryptocurrency mixing services and identified that CoinJoin is one of best techniques. Identified the problems associated with the other mixing services like SharedCoin and Dark Wallet.

– *Venkatesh Kallepalli*

Studied the background of Blind Signature. examined various Blind Signature techniques and arrived at the conclusion that RSA Blind Signature will be apt for this project, as the blind signature can be verified by any person. Implemented Blind Signature in Java. KeyPairGenerator is used to generate the corresponding sk and pk. The modules such as RSAPrivateKey and RSAPublicKey in java are used to retrieve the values of e, N and d. Implemented Blind Signature and verification of the signature using BigInteger.

– *Dinesh Reddy Kavalakuntala*

8. References

[1] Agora White Paper
https://www.agora.vote/Agora_Whitepaper_v0.2.pdf

[2] Mastering Bitcoin: Unlocking Digital Cryptocurrencies
Book by Andreas Antonopoulos.

[3] Follow My Vote White Paper
https://followmyvote.com/wp-content/uploads/2014/08/The-Key-To-Unlocking-The-Black-Box-Follow-My-Vote.pdf

[4] Conference paper: Blind signatures based on the discrete logarithm problem
https://link.springer.com/content/pdf/10.1007/BFb0053458.pdf

[5] Online Voting Roundtable: Electoral Futures in Canada
https://www.youtube.com/watch?v=-Uw-JnGIeIE

[6] Implementation of Blind Signature in RSA
https://www.cs.bham.ac.uk/~mdr/teaching/modules06/netsec/lectures/blind_sigs.html

[7] CoinJoin
https://en.bitcoin.it/wiki/CoinJoin

9. Appendix

[1] GitHub link to presentation

[2] GitHub link to code

[3] GitHub link to Final report