HUVote: Voting System for College Polls and Elections

Fatıma Betul Tan , Gizem Kaya, Mehmet Taha Usta, and Furkan Çağlayan

Hacettepe University Computer Engineering Department

July 8, 2021

Abstract

After the emergence of Blockchain technology, the other systems we used started to benefit from this technology. Of course, this effect was reflected in the paper-based voting. This article includes the processes of the e-voting system and the explanation of the proposed model that our group is working on. The model is an e-voting system using ethereum blockchain which can be used by the college in their surveys and elections.

Keywords: E-voting, Ethereum, Smart Contracts

1 Introduction

The national election is a very important topic for all countries in the world. The most important issue in the elections is security. When we look at the problems of the paper-based votes used so far, we can talk about financial problems such as the costs of ballot papers and ballot boxes, the appropriations of the officials who must stand at the beginning of each ballot and count the votes at the end of the election. In addition, the time spent on all these preparations is another problem. But the real problem is that the paper-based system does not provide complete security against manipulation of the consequences of theft or misunderstanding of votes. It was first thought to use electronic voting machines instead of this traditional method. However, with physical access to the machine later, it was realized that it was not difficult to influence the votes. This means a major security problem. For these reasons, the use of electronic machines was abandoned. In some states of Australia and Estonia, the e-voting system is used. However, it has been concluded that this system is not resistant to malware and attacks. The fact that the blockchain structure consists of data that cannot be changed interconnected by encryption has shown that the security and transparency that cannot be provided in voting systems can be ensured.

1.1 Blockchain

Blockchain is a database with protocol properties. A copy of this database exists on multiple computers. Computers are connected to each other via the P2P network. There is no center or server (decentral-

ized). The data is kept in a chain of blocks connected together. The block is produced by the process called Proof of Work (PoW) which requires a series of math and hash operations. Each block holds its own hash information as well as the hash value of the previous blog summary. This ensures that the inter-block connection cannot be changed. The blocks are confirmed by a series of operations performed by the miner. The approved block is added to the chain. The fact that no one can add or delete data in this chain, that it is cryptographically signed and verifiable by everyone, provides great assurance. In an e-voting system using Blockchain technology, the selection results can be secured by storing each piece of data irrevocably.

1.2 E-Voting

Estonia and New South Wales used the e-Voting system. However, it was discovered that a sample software for analysis systems has vulnerabilities to many attacks, such as malware, network attacks, and server attacks. The resistance of the blockchain technology to piracy with its stability property indicates that it is possible to use it to ensure the safety of the election results. The first electronic voting system was introduced by David Shaum in the early 1980s. The system used public-key encryption that was used to vote and keep voters anonymous. Blind signature theorem is used in this system. The aim is to ensure that there is no connection between voters and ballots. Scientists have shown great interest in this system and have done research on it. We can say that the e-Voting system makes it easier to vote. However, technical threats to the e-voting system have always led to doubt.

Estonian E-Voting System In 2005 Estonia became the first country to use the i-voting system in the elections. Citizens voted on the internet with an electronic national ID card. More than 30 % of citizens voted using the i-voting system.

Norwegian E-Voting System In 2011, Norway used an electronic voting system. It was originally made by Schnorr and Jakobsson in 2000 for country council elections. It was later developed by Scytl, a Spanish company. It looked like the Estonian e-Voting system. After 2014, Norway did not use the system due to security concerns. The biggest criticism of the Norwegian i-voting system was that it did not give confidence in voting in case of a cyber attack.

New South Wales E-Voting System In 2015, 280,000 citizens cast their votes in the New South Wales State elections through the e-voting system. This system was developed by Scytl Company. Unlike others, voters are presented with a gradual protocol. Citizens can check whether they can vote correctly.

e-Voting in USA e-Voting system was also used in the USA in 2000. It was an experimental study, but we can say that it is an important step in the use of an electronic voting system. In 2010, Washington D.C developed an i-voting system. He tested the reliability of this system with an untrue selection. This project was canceled and unused because some problems were encountered.

The main reason for criticizing the voting system in Estonia and Norway is that the critical parts of the code are hidden. This creates problems with transparency. An open-source voting system would be better for the election to be reliable.

2 Related Works

With the increasing potential of the decentralized interactive systems and platforms such as smart contracts that are provided by Ethereum Network[1], it is now possible to create voting-focused systems. One of such was the work[2] of Koç et al. They implemented a smart contract voting system on Solidity, which is an object-oriented language for implementing and executing smart contracts, and tested their system using the Rinkeby Network, a testing network that is used by developers to test their smart contracts. Another Ethereum implementation of voting is Ques-Chain[3]. Zhang et al. used blind signatures[4] to protect the privacy of the voters. In [5] Meeser used linkable ring signatures to sign the votes over the smart contract voting model. Apart from these, the related studies on the subject are as follows in historical order:

Helios / Zeus / Apollo: Helios Voting[1] is an open-source, web-based electronic voting system that uses homomorphic encryption to ensure ballot secrecy. Helios encrypt the sealed Voters' ballot with ciphertext. After the election ends, the System shuffles the ballots decrypts all the votes and makes the shuffle publicly accessible for interested parties to audit. Auditing allows anyone to verify that the shuffle is correct. Once a reasonable amount of time for auditing has passed, Helios decrypts the ballots and tallies the votes. Over time researchers have explored possible vulnerabilities in different versions of Helios. When searching for a solution, two different systems are created. One is Zeus, that uses Helios for counting the ballots, not for producing the election results. (uses mixnet) The other one is Apollo, which is an extension for Helios but fixing the well-known vulnerabilities. Both Systems has increased the feasibility of Helios and fixed the vulnerabilities of it.

[1] https://heliosvoting.org/

"An anonymous distributed electronic voting system using ZeroCoin"[6]: With the Zero-Knowledge-Proof feature provided by ZeroCoin, it aims to give anonymity to the voting systems made through Bitcoin. It allows us to vote online using both Zerocoin and bitcoin together. In summary, the preparation process is completed by converting bitcoins into ZeroCoin, passing through laundry, and then converting ZeroCoins back to Bitcoin. After this stage comes to the voting stage. The use of ZeroCoin has solved the privacy issues of bitcoin.

"A Smart Contract for Boardroom Voting with Maximum Voter Privacy"[7]: An implementation of a decentralized and independent internet voting protocol with blockchain and maximum voter privacy. The Open Voting Network was written as a smart contract for Ethereum. The difference between this approach and previously proposed Blockchain e-voting protocols is that it does not trust a reliable authority to protect the privacy of the voter or calculates the tally.

"Crypto-Voting, a blockchain-based e-voting system"[8]: It is a new e-voting system called crypto-voting. Shamirs' secret sharing approach is used with blockchain to form the basis of the system. Crypto-Voting aims to increase and improve voting paths and traceability without any intermediaries. The field of application is the integration of the management procedures of a selection stage and events. At the end of these studies, the system includes the following actions: installation of the system, distribution of identity information, voting, collecting ballots, counting preferences, publishing the results.

"Platform-independent Secure Blockchain-Based Voting System"[9]: In electronic voting systems, the trust of the third-party public bulletin board for the publication and audit of voting results by whole participants and the use of previously presented systems due to limitations in supported voter and candidate numbers and security frameworks, which are essentially dependent on the underlying blockchain protocol and potentially vulnerable. A platform-independent, secure and verifiable voting system that can be placed in any blockchain that supports the execution of an smart contract. While authentication is basically provided by the basic block chain platform, encryption techniques such as Paillier encryption, proof-of-knowledge and linkable ring signature are used to provide a framework independent of security and privacy features for system security and user privacy.

"Votereum: An Ethereum-Based E-Voting System"[10]: The procedure is as follows: the person who will not vote in this system is registered. The system checks whether that person is eligible and sends a token. the person to vote shall request a ballot. This will trigger a function in the Ethereum smart contract, in the end, the ballot response returns. this response is answered by voting. Voter information used by voters is stored in the central database (Government database).

3 Analysis of the Problem

Anonymity/Privacy: Votes must be secret. Entities can't link a vote with the voter

Receipt-Freeness: Voter cannot receive receipt.

Robustness/Soundness: Voters and other participants can't disrupt an election.

Verifiability: A voter can verify that own vote was really counted.

Fairness: Early results should not be obtained. Early results could influence the remaining voters

Correctness: More accurate results are obtained if participants behave as planned.

Completeness: All valid and accepted votes are considered correct

Eligibility: Only lawfull voters can vote

Unreusability: All voters can vote only one vote

3.1 Security

RSA: The encryption key is public and it is different from the decryption key which is kept secret (private). In RSA, this asymmetry is based on the practical difficulty of the factorization of the product of two large prime numbers, the "factoring problem".

Ring Signature: A ring signature is a type of digital signature that can be performed by any member of a group of users that each have keys. Therefore, a message signed with a ring signature is endorsed by someone in a particular group of people.

DDoS: The attacker must DDoS every single boot node in the private network. Implemented to each node with the Byzantine fault tolerance algorithm to find failed nodes in the system.

Authentication Vulnerability: The person who casts the vote shall have an electronic identity. To verify the system, it must enter the 6-digit PIN sent to the phone by the system.

Sybil: An individual creating a large number of nodes to interrupt network operation by intercepting or dropping messages. When our proposal is created and shared on a private network, no one has access to create a network.

4 Proposed Model

We propose a self-tallying, receipt-less, transparent and immutable voting system for small to medium-sized elections and polls.

Assumptions:(1)We assume each elector has an official password between 8-16 bytes and they are saved in a protected database. (2)Electors are able to keep their passwords private. (3)Each selection is given an ID, starting from 1 and selections count is given to the system.

Token Generation: To vote, each elector must be verified first, using their passwords. This stage is done via the token generation. When an elector requests a token generation, the system concatenates the password of the elector and the Unix timestamp(this is saved to a temporary database), creating a 30 bytes long token after adding 4 zeros to the end.

Verification: We said passwords are available via some kind of protected database. Using the same procedure above, the system fetches the password from the database and verifies if the password is valid. If it is, vote selections become visible to the elector.

Voting: After the elector votes for a selection, the ID of the selection is added to the token and it gets encrypted with SHA256 algorithm in order to hide the vote on Ethereum blockchain. The hash output is saved to the contract for tallying phase.

Tallying: After the voting time has ended, anyone can request a tallying which makes the system self-tallying and verifiable by third parties. If a tallying call has been made, the system fetches the electors' passwords from the system and performs hash comparisons for each one of them for each possible selection. If a hash match is found, vote count for the selection is incremented by 1.

5 Implementation Details

We implemented HUVote with Solidity programming languages just as in [2, 3]. Then using web3.js[2], Ethereum JS API and Python we optimized the code. The code[3] is open to access and licensed under GNU General Public license v3.0[4]

System has these actors: **Moderator**, **Proposal** and **Elector**. A proposal is simply an election that was proposed by a moderator or an elector. If a proposal was proposed by a moderator, then the proposal is *official*. If it was proposed by an elector, then the proposal is just a basic poll that can be used among student groups unofficially.

Election. If an instructor wants to start an official election, they have to run the contract with their previously authorized Ethereum wallets. If their address is confirmed, they become moderators and can call the necessary functions to create an official

[2]https://github.com/ethereum/web3.js/

[3]https://github.com/furkancaglayan/voting-system-for-college-polls-and-elections

[4]https://www.gnu.org/licenses/gpl-3.0.html

election.

Voting. System guarantees an elector can vote only once. But in official elections, the moderator has the ability to reset a voter's status *once*. So if an elector wants to revert their vote with an acceptable reason, the moderator can make that happen, again, only once.

6 Limitations

Given the recent advances in the hardware technologies, SHA256 might be insufficient to hide the password of an elector. To overcome this problem, passwords can be rearranged as 32 characters and SHA512 can be used as a hash function.

The current system requires a database that contains instructors' or possible moderators' wallet addresses. This centralization is obviously open to abuse. One of the future works might be to overcome this problem, maybe with a consensus among electors.

Another centralization problem is giving too much control to instructors or moderators over the system. For practical concerns, we wanted the moderators the set the time limit, participator count and the ability revert the elector's vote but an ideal voting system must be completely decentralized and transparent while keeping the elector credentials private and secure.

7 Conclusions and Future Works

As a result, the use of Blockchain technology in the e-voting system due to its general nature will facilitate the process for all actors in the voting process. Although there are peers in the world, this new technology will become more widely available in other countries. In our proposed model, we used blockchain Ethereum technology. We think it is an idea that is open to development and suitable for its versatile use.

References

[1] G. Wood *et al.*, "Ethereum: A secure decentralised generalised transaction ledger," *Ethereum project yellow paper*, vol. 151, no. 2014, pp. 1–32, 2014.

[2] A. Koç, E. Yavuz, U. Çabuk, and G. Dalkılıç, "Towards secure e-voting using ethereum blockchain," 03 2018.

[3] Q. Zhang, B. Xu, H. Jing, and Z. Zheng, "Queschain: an ethereum based e-voting system," *arXiv preprint arXiv:1905.05041*, 2019.

[4] D. Chaum, "Blind signatures for untraceable payments," in *Advances in cryptology*, pp. 199–203, Springer, 1983.

[5] F. L. Meeser, "Decentralized, transparent, trustless voting on the ethereum blockchain," 2017.

[6] Y. Takabatake, D. Kotani, and Y. Okabe, "An anonymous distributed electronic voting system using zerocoin," 2016.

[7] P. McCorry, S. Shahandashti, and F. Hao, "A smart contract for boardroom voting with maximum voter privacy," 01 2017.

[8] F. Fusco, M. I. Lunesu, F. E. Pani, and A. Pinna, "Crypto-voting, a blockchain based e-voting system.," in *KMIS*, pp. 221–225, 2018.

[9] B. Yu, J. K. Liu, A. Sakzad, S. Nepal, R. Steinfeld, P. Rimba, and M. H. Au, "Platform-independent secure blockchain-based voting

system," in *International Conference on Information Security*, pp. 369–386, Springer, 2018.

[10] L. Vo-Cao-Thuy, K. Cao-Minh, C. Dang-Le-Bao, and T. A. Nguyen, "Votereum: An ethereum-based e-voting system," in *2019 IEEE-RIVF International Conference on Computing and Communication Technologies (RIVF)*, pp. 1–6, IEEE, 2019.