# VoteChain: A Blockchain Based E-Voting System

Archit Pandey
*Department of Computer Science and Engineering*
National Institute of Technology Karnataka,
Surathkal Mangalore, India
architpandeynitk@gmail.com

Mohit Bhasi
*Department of Computer Science and Engineering*
National Institute of Technology Karnataka,
Surathkal Mangalore, India
mohitbhasi1998@gmail.com

K. Chandrasekaran
*Department of Computer Science and Engineering*
National Institute of Technology Karnataka,
Surathkal Mangalore, India
kchnitk@ieee.org

*Abstract*— **In the past, electronic voting systems have not seen widespread adoption due to data privacy concerns. Previously proposed e-voting systems make use of a central database to store data, resulting in the servers used to store these databases being a single point of failure. These systems have also been found to be vulnerable to DoS attacks, leading to concerns over their reliability.**

**Blockchains have been used to build secure and scalable distributed systems which have shown several benefits over centralized systems. They have seen uses in sectors ranging from finance and healthcare to food and energy.**

**In this paper, we present VoteChain, a blockchain based voting system to help bring transparency and security to polls. We report on our implementation of VoteChain, as well as the results obtained in testing the system in a real-world poll which prove that such a system can be used in practice for large-scale elections.**

***Index Terms— E-Voting, Blockchain***

## I. INTRODUCTION

Since the introduction of Bitcoin [9] - an electronic currency or crypto-currency - by Satoshi Nakamoto, researchers have found ways in which the underlying technology of blockchain can be used in privacy protection and access control [14]. The novel idea behind Bitcoin is how public transactions can take place securely, without the need of a central regulatory authority, even in the presence of malicious nodes. Unlike traditional databases, data can only be inserted into or read from a blockchain. Hence data once in the blockchain cannot be deleted or modified with very high probability.

Electoral processes play a decisive role in a country's or organization's future. Currently, polls are conducted using electronic voting machines or ballot papers. Modern voting machines have many well-known hardware and software vulnerabilities that can be exploited by malicious individuals [11]. On the other hand, using ballot papers can slow the entire process of deciding the vote, because of the time and effort needed in counting the final votes. On-line voting systems that have been used in the past such as the one used in Estonia [8] have proven that e-voting can be both fast and reliable.

### A. *Our Contributions*

- We propose a blockchain based voting system called VoteChain.
- We report on an implementation and successful execu- tion of a VoteChain system in practice.

### B. *Organization*

We first study the state of the art electronic voting systems and reasons for them not being used widely. Further, we perform a literature survey for applications where blockchains have been successfully used. Finally, we propose a blockchain based voting system called VoteChain and report on its implementation and results obtained from testing.

## II. BACKGROUND

### A. *Present E-Voting Systems*

*1) Electronic Voting Machines:* Electronic Voting Machines or EVMs are the most widely adopted voting systems. Voting machines are used to store votes from each polling station, and individual voting machines are transported to central counting centers where their vote data is retrieved and counted.

*2) Estonian E-Voting System:* Estonia was one of the early adopters of e-voting systems [8]. Since 2005, voters in Estonia have been allowed to cast their vote via the Internet. The vote is cast through an application that is downloaded and installed locally on the voter's PC. The system makes use of asymmetric encryption schemes to encrypt all data transmitted between the clients and servers, while central servers owned by the government are used to store the vote data.

*3) Washington DC Internet Vote:* The Washington DC Internet Vote system was proposed in 2010 to allow absentee voters to cast their votes online. The system is based around an open-source application developed by TrustTheVote foundation using the Ruby on Rails framework. Votes are cast through PDF forms which are uploaded securely to the server. A MySQL database is used to store the votes on the

**978-1-7281-3694-3/19/$31.00 ©2019 IEEE**

server, and multiple firewalls are placed in order to reduce the size of the attack surface. Votes are moved to a non-networked computer once the elections are completed and then counted. This system didn't see mainstream use due to several security vulnerabilities found by Wolchok et. al. [12].

After examining currently used and previously proposed e-voting systems, we make the following conclusions:

- Purely electronic voting systems such as the one used in Estonia have not found mainstream success.
- Systems that depend on a central server can be vulnerable to Denial of Service attacks.
- Systems that use a combination of electronic and physical voting such as the one currently in use in India suffer from issues that affect ballot paper-based polls, namely delays in counting votes, and costs incurred for transporting voting machines to counting stations.

### B. Related Work

Azaria et. al. have shown that using a blockchain as a scalable and secure data-store for patient-provider relationships can enable better data sharing and collaboration in health- care applications [1]. Lee et. al. have used a blockchain for building BIDaaS or ”Blockchain ID as a Service” [7] that serves as a way of generating virtual IDs for user verification of online transactions. The ID provider maintains a private blockchain while partners of the provider are given read access to the chain in order to be able to verify the user- generated IDs.

Tse et. al. have made use of a blockchain to store data such as food circulation, quantity and origin [10] in order to improve transparency in the food supply chain. Zhang et. al. have shown that the Hyperledger fabric can be used to enable secure handling of data for the Internet of Things devices [13]. Guo et. al. have found that using blockchain technology has helped banks in significantly reducing their transaction fees [5].

Hence, literature contains several applications where blockchain technology has been well adapted for different use-cases. These applications differ in the data stored inside blocks, how mining takes place and the mechanisms used to reach consensus.

We now present an e-voting system based on a blockchain that can potentially solve the issues plaguing current day e-voting systems.

## III. PROPOSED SYSTEM

We aim to use a blockchain as a distributed public ledger to store vote data. This will allow the voting system to be scalable and secure, along with bringing transparency to the entire vote process.

### A. Structure of Each Vote

We make use of a *Vote* as the atomic data stored on the blockchain, much like a *Transaction* in the Satoshi blockchain. Each vote contains the following fields:

1) Hash: Hash of the entire *Vote*
2) Timestamp: The time at which vote is cast
3) Vote data: The choice of vote
4) Voter identification: A unique ID to verify user's identity
5) One-Time Password: To verify user authenticity

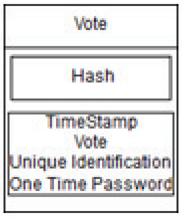

Fig. 1. Data Stored in Each Vote

Vote data is a 32 bit or 4-byte data field with one bit set, denoting the choice of candidates made by the voter. A vote data field with more than one bit set, is considered invalid. Voters are identified using their Unique ID (UID) numbers (ex: a Government issued ID such as Aadhaar in India). These UID numbers are also used to ensure that a voter does not vote twice (a problem commonly known as *double- spending*). For ensuring the correctness of UID provided, we make use of the Aadhaar infrastructure [6] in place in India. Each *Vote* is hashed in the form of a Merkle Tree, and a Merkle root is stored in a *Block*. This arrangement is similar to what is used in Bitcoin. Using a Merkle Tree structure, results in large space savings while at the same time allowing for efficient retrieval of a *Vote*.

### B. Structure of Blocks

Each block contains the following:

1) Hash Value of Previous Block's Header
2) Proof of Work (using *VoteMaker*)
3) Root of Merkle Tree of *Votes*

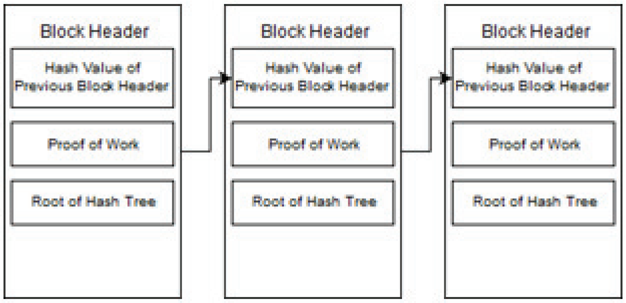

Fig. 2. Structure of each Block and the Blockchain

Our blockchain lacks monetary incentive for miners. Hence mining nodes are either run by the organization conducting the vote or by volunteers. No constraint is placed on the entities allowed to run mining nodes.

Our proposed system makes use of Proof of Work for maintaining consensus across servers. HashCash [2] is the currently the most popular algorithm for performing Proof of Work. In our system, we make use of our variant of HashCash called *VoteMaker*.

### C. VoteMaker: Proof of Work

Our Proof of Work algorithm makes use of the SHA256 hash [3] of a generated nonce value. It is based on finding partial hash collisions of the hashed nonce on all hexadecimal strings of length k that satisfy the property of a Binary Search Tree stored in-order. The choice of length *k* is used to adjust the hardness of the proof.

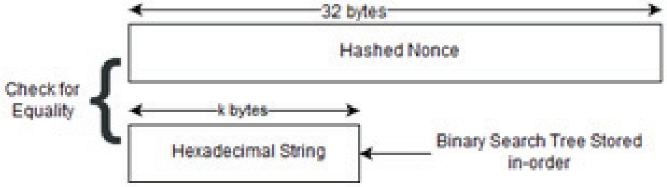

Fig. 3. *VoteMaker*: Proof of Work

Much like HashCash, for a party generating a Proof of Work using *VoteMaker*, brute-forcing the nonce values for e.g. by using a counter, is the most efficient way of arriving at a collision.

## IV. TEST IMPLEMENTATION

A test implementation of our system was developed for use in a class election with over 100 voters. The blockchain and mining servers were implemented in Python using the Flask framework. A web portal for voting was built using HTML, CSS, and PHP.

### A. Web Portal

A web portal is used for voting and viewing vote results. The web portal provides certain views for both candidates and voters. The candidate views are as follows:

1) Sign-Up: Candidate details are gathered before the voting process begins, by enabling the sign-up feature for candidates.

2) Dashboard: Candidate dashboard shows the candidate details and also information on the voter demographics for votes received by the candidate.

3) Vote Results: The results view shows live statistics from the election.

The portal does not require voters to register before voting. Voter identity is verified using a UID number, and their authenticity is verified using a One-Time Password sent to their mobile numbers. Hence, voters are given two views, as follows:

1) Cast Vote: Voter details are collected and verified before a vote can be cast.

2) Vote Results: The results view shows live statistics from the election.

Upon receiving a vote from the voter, the web-portal makes a request to a *gateway node* to the blockchain network. Beyond making the request, the vote is handled by the mining nodes in the blockchain network.

### B. Blockchain

The blockchain network consisted of three nodes and one gateway node for our tests. The gateway node serves as an access point for all requests made to and from the blockchain network. All core blockchain functionality was coded using Python and the Flask web framework. Each node in the network ran a copy of the VoteChain application.

The features included in the VoteChain application were:

1: Adding new votes
2: Adding new blocks
3: Consensus using *VoteMaker* Proof of Work

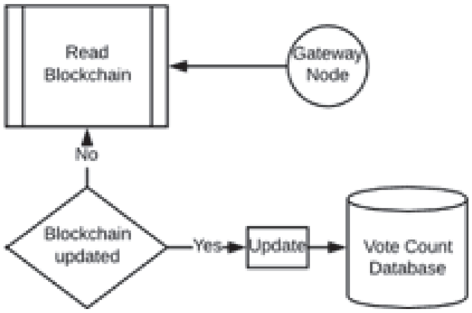

Fig. 4. Synchronous Script Running on Web Server

The web server makes use of a script called 'Read Blockchain' that is responsible for scanning the blockchain network (through the gateway node) in real-time for any changes and updating the SQL server on the web portal to reflect the changes to the blockchain. This script provides live vote data that is displayed to both voters and candidates.

### C. Results

The web-portal was hosted on the cloud with the following specifications:

- 1Ghz Processor
- 1GB DDR4 RAM
- 100GB SSD Storage

The blockchain network was run on 4 machines with the following specifications:

- Intel i7-8750H 6-core Processor
- NVIDIA GeForce 1060 with 6GB GDDR5 Memory
- 16GB DDR4 RAM
- 1TB SSD Storage

Additionally, all machines in the blockchain network were connected using 1Gbps LAN cables.

We first performed tests to find the relation between the time taken to mine per vote and the number of votes per block using the *VoteMaker* algorithm. The results obtained are shown in Fig. 5. With increasing size of the block, the time taken to transmit blocks between nodes becomes an inhibitor to further reduction in time taken to mine each vote.

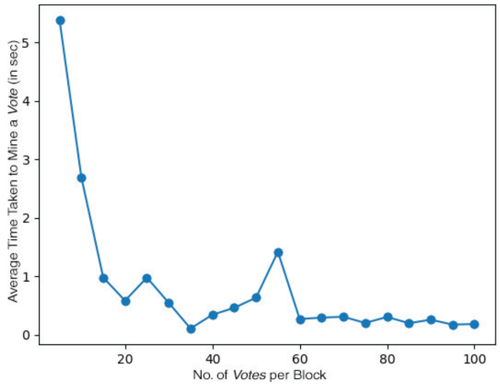

Fig. 5. Relation Between Mining Time and Number of Votes per Block

The results obtained were in line with those obtained from tests performed on Bitcoin's network [4].

Further, we performed tests to measure the delay between casting of votes and votes being added to a block. For these tests the hardness factor *k* for *VoteMaker* was set to 3. Each vote cast during the test election was timed from being cast on the web portal to being reflected in the blockchain. Fig. 6. shows the time taken for each vote reaches a maximum of 90 seconds. These results are acceptable as only four machines make up the blockchain network.

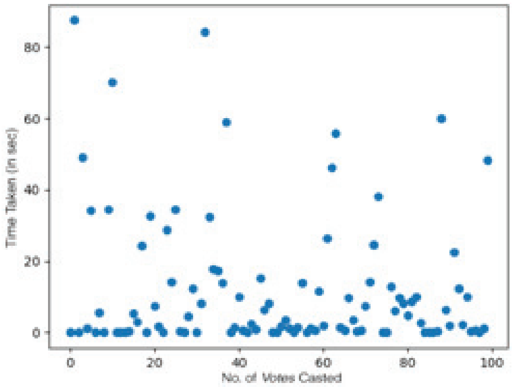

Fig. 6. Time Taken per Vote Cast

## V. CONCLUSIONS AND FURTHER WORK

We have shown a proof of concept for a blockchain based voting system which can solve the issues faced by current e-voting systems such as Denial of Service attacks and high voting times. Our test implementation has shown that such a system can be used in practice with good results.

Implementing the VoteChain system on a larger scale, with more than four machines and testing it for scalability, and making use of innovative consensus mechanisms and studying their effect on vote times are topics which can be taken up for further study.

## REFERENCES

[1] Azaria, Asaph, Ariel Ekblaw, Thiago Vieira, and Andrew Lippman. ”Medrec: Using blockchain for medical data access and permission management.” In Open and Big Data (OBD), International Conference on, pp. 25-30. IEEE, 2016.

[2] Back, Adam. ”Hashcash-a denial of service counter-measure.” (2002).

[3] Eastlake 3rd, D., and Paul Jones. US secure hash algorithm 1 (SHA1). No. RFC 3174. 2001.

[4] Gobel, J., and A. E. Krzesinski. ”Increased block size and Bitcoin blockchain dynamics.” In 2017 27th International Telecommunication Networks and Applications Conference (ITNAC), pp. 1-6. IEEE, 2017.

[5] Guo, Ye, and Chen Liang. ”Blockchain application and outlook in the banking industry.” Financial Innovation 2, no. 1 (2016): 24.

[6] Khera, Reetika. ”The UID project and welfare schemes.” Economic and Political Weekly (2011): 38-43.

[7] Lee, Jong-Hyouk. ”BIDaaS: blockchain based ID as a service.” IEEE Access 6 (2018): 2274-2278.

[8] Madise, lle, and Tarvi Martens. ”E-voting in Estonia 2005. The first practice of country-wide binding Internet voting in the world.” Electronic voting 86, no. 2006 (2006).

[9] Nakamoto, Satoshi. ”Bitcoin: A peer-to-peer electronic cash system.” (2008).

[10] Tse, Daniel, Bowen Zhang, Yuchen Yang, Chenli Cheng, and Haoran Mu. ”Blockchain application in food supply information security.” In Industrial Engineering and Engineering Management (IEEM), 2017 IEEE International Conference on, pp. 1357-1361. IEEE, 2017.

[11] Wolchok, Scott, Eric Wustrow, J. Alex Halderman, Hari K. Prasad, Arun Kankipati, Sai Krishna Sakhamuri, Vasavya Yagati, and Rop Gonggrijp. ”Security analysis of India's electronic voting machines.” In Proceedings of the 17th ACM conference on Computer and com- munications security, pp. 1-14. ACM, 2010.

[12] Wolchok, Scott, Eric Wustrow, Dawn Isabel, and J. Alex Halderman. ”Attacking the Washington, DC Internet voting system.” In Interna- tional Conference on Financial Cryptography and Data Security, pp. 114-128. Springer, Berlin, Heidelberg, 2012.

[13] Zhang, Yu, and Jiangtao Wen. ”An IoT electric business model based on the protocol of bitcoin.” In Intelligence in Next Generation Networks (ICIN), 2015 18th International Conference on, pp.184-191. IEEE, 2015.

[14] Zyskind, Guy, and Oz Nathan. ”Decentralizing privacy: Using blockchain to protect personal data.” In Security and Privacy Work- shops (SPW), 2015 IEEE, pp. 180-184. IEEE, 2015.