Rock the Blockchain Vote | Reinvent Team

Implementing the Blockchain

Reinvent Team Members: **Adrian Barberis** (abarber3@uwyo.edu), **Jagadish Bapanapally** (jbapanap@uwyo.edu), **Adam Coggeshall** (acoggesh@uwyo.edu), **Rafer Cooley** (rcooley2@uwyo.edu), **Kip DeCastro** (kdecastr@uwyo.edu), **Wyatt Emery** (wemery1@uwyo.edu), **Kyle Mitchell** (kjmitch@uwyo.edu)

Motivation/Abstract

Our goal was to develop a fully electronic voting system supported by a customized blockchain implementation which has the potential to be a much more robust application than current electronic voting solutions. Arguments against computerized voting cite inevitable software bugs and security incidents that negatively impact not just the system but the integrity of the election. A blockchain implementation created specifically to address these issues utilizes distributed ledgers and cryptographic hash functions to make the voting process verifiable while securing the integrity of both the data and the ballots of voters. The ledgers keep data in two blockchains, one for the identities of voters and the other for their votes. This separation allows the system to tally and recount the votes without tying them to any voter-identifying information, and provides verification that no voter can place more votes than they are allowed. The ledgers are distributed among a network of server nodes which will be distributed throughout the state, which enables a consensus among the nodes regarding the validity of the voting activity. Using a ledger system akin to that used in most all current cryptocurrencies allows for both an immutable vote history as well as full auditing capabilities to detect fraudulent activities.

Background

A blockchain is a list of records, represented as a growing ledger of transactions made in a system. Transactions are grouped and stored in blocks, which are then linked together using cryptographic hashes. Storing the ledger in this form creates an immutable transaction history. This history, with the distributed network a blockchain uses to decentralize authority, can enable a more secure and efficient voting method that can be computerized. Our task was to implement a customized blockchain for such a voting application.

Project Summary

Tasks

- Research existing implementations of blockchain systems and their applications
- Separate user information by implementing separate Identity and Vote blockchains
- Design the overall system and its subcomponents around these separate chains
- Build an API to connect to the UI and the hardware (systems built by other teams)
- Work with other teams to integrate development on front end and back end

Components

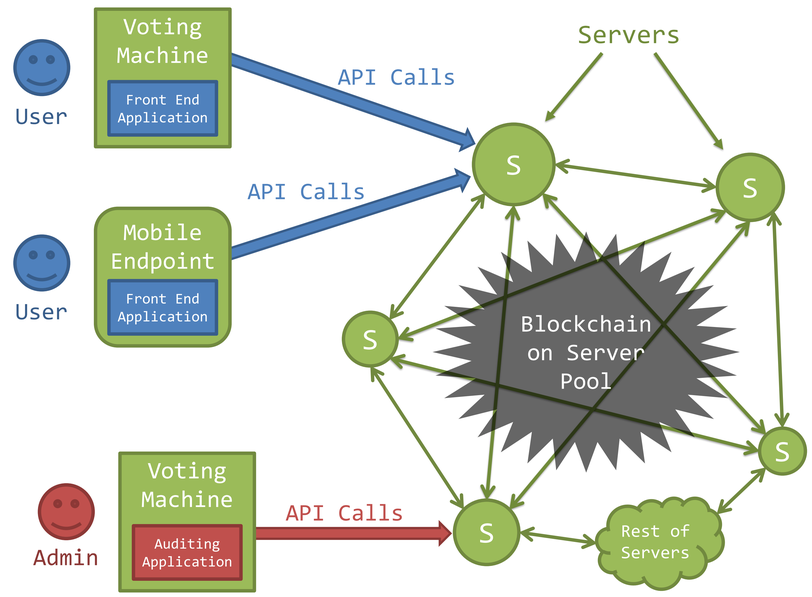
- Custom blockchain implementation, written in Python using the Flask framework
- JSON data format for messages within and with other parts of the application
- Application interface and endpoints for serving data through calls over HTTP
- Testing suite developed alongside other system components

Limitations & Challenges

Building an electronic voting system is a completely different application than the cryptocurrency ideas that originally created the blockchain mechanism. One challenge was deciding which components of the cryptocurrency model were not needed in our system, which is instead focused on the utility of its civic purpose. Additionally, a primary focus was on the security of the blockchain's distributed nature, building on that to enhance voter privacy and the efficiency of the voting system's role in an election.

Deliverables

Components of the Blockchain Voting System



The Reinvent Team's implementation covers the server software and its application programming interface. Note on this figure: The Vote chain is demonstrated; the Identity chain is similar.

Application Components

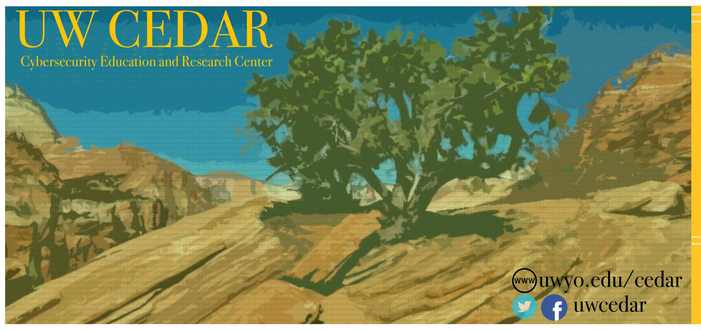
- **User:** Uses a voting machine at a voting center or a mobile endpoint application to cast votes on the current ballot
- **Front end application:** Presents the ballot to users of voting machines or a mobile endpoint and accepts their votes on the ballot
- **Voting machine/Mobile endpoint:** Runs application and makes calls to a server through the API to process a voter's information
- **Server:** Accepts votes from all voting machines at a voting center or from local mobile endpoints, validates the format of the data, pools multiple votes into a block, and broadcasts blocks to other servers in the server pool.
- **Server pool:** Network of all servers processing votes for the current election, covering all districts, voting centers, and voters. Servers receive vote blocks, compare among all recent blocks to avoid duplicating votes, and add confirmed blocks to the blockchain.
- **Blockchain:** Distributes data to all servers and tracks transactions using cryptographic signatures from blocks, accessible for audits by administrators
- **Administrator:** Proctors at a voting center, and uses a special version of the voting application to audit the process and the chain as needed

Results

The voting system and the customized blockchain implementation built to support it is functional, working as intended, and passing all tests. The front end application will make API calls to the server to process user vote information, which will then pool votes into blocks of votes. The block is broadcast to the rest of the server pool. Other servers compare the new block with other recent blocks to avoid duplicating votes, and then add it to the chain. Support modules were added to the repository, for generating sample electorate and ballot data for the system to process, and for displaying results during the process in a local web page to show successful operation. This demo setup is for testing and presentation, outside of integration with the more complete user-facing applications the UI team has built.

Conclusions & Future Work

Everyone involved in the course has gained knowledge in the mechanisms behind blockchain and the applications driving their development, and the experience working on a software development project team will be applied immediately in careers and in future studies. Future work on this project will include extensive testing and hardening of the blockchain subsystem and the voting application as a whole, and iteration on the codebase to improve and refine its operation. As the system matures as an application, members of our team and others in the course may pursue marketing the voting system as a product to government and corporate agencies, possibly with help from the Wyoming Technology Business Center.



UNIVERSITY OF WYOMING

This poster made possible by:
UWYO College of Engineering & Applied Science, UWYO Computer Science Department, and UWYO CEDAR.