VOTING SYSTEM WITH FINGER PRINT SCANNER USING BLOCKCHAIN TECHNOLOGY

Adithya.H, Chithambara Karthikeyan.T, Harish Kumar.N, Ms. P. Shanmuga Priya Assistant Professor, Electronics and Communication, Rajalakshmi Engineering College, adithyaraven@gmail.com, karthikeyancubs@gmail.com,harishkr.1110@gmail.com,shanmugapriya.p.ece@rajalakshmi.edu.in

Abstract:

Blockchain technology is one of the emerging technologies in today's internet world. Although blockchain was originally created for bitcoins, it is found to be useful for various administrative, agricultural and other day to day services. Smart contracts are a robust set of tools which help the blockchain to provide a safer and secure environment. Smart contracts are a set of small codes of programs which help blockchain to implement the objective in a scheduled manner. Voting is one of the most important and very critical process for a country. The low voter turnouts for elections is a worrying concern for the guardians of democracy. The emergence of blockchain has provided a timely solution to help assure the voters to vote regularly and as well help hold the values of their votes high. The blockchain process is implemented in an Ethereum platform which is one of the preferred platforms due to its provisions for use of smart contracts. An ideal voting system must be one, where, votes can only be cast once and not be duplicated and most importantly the votes once cast cannot be altered, all at the expense of the protection of the identity of the voters. In this work we have employed and put into action a sample voting application as a smart contract on the Ethereum networks using solidity language and Ethereum wallets. People can vote directly from their Ethereum wallets and each transaction is carried out in agreement with all the nodes on the network. Fingerprints of voters are taken as a security for identification of voters and after matching the fingerprints, the personal records of the voter are displayed. In this paper we have attached our results for the application after testing.

*Keywords: **Blockchain, Smart contracts, Ethereum, Fingerprint, Voting.***

INTRODUCTION:

A blockchain is collection of data stored in blocks, that is managed by a group of nodes present in the network which are not owned by a single entity. The stored data cannot be altered thereafter. These blocks of data are stored securely and connected to each other like chains using cryptographic principles. Blockchain technology is the back bone for the crypto-currency systems. A Blockchain can serve as "an open and distributed ledger, that can record transactions between two parties in a verifiable and permanent way." This ledger that is shared among everyone in the network is public for all to view. This brings in transparency and trust into the system. A block is the current part of a blockchain which records some or all of the recent transactions and once completed goes into the blockchain as a permanent database. After completion of block, new blocks are generated for use. Once recorded, the data in any given block cannot be altered retroactively without the alteration of all subsequent blocks and a collusion of the network majority. Transactions once stored in the Blockchain are permanent. They cannot be hacked or manipulated. A few platforms, applications and frameworks help in creating a successful blockchain database. Chocolatey, ganache and truffle help in creating the blockchain database.

Chocolatey is a package manager for Windows. It was designed to be a decentralized framework for quickly installing applications and tools that you need. It is built on the NuGet infrastructure currently using PowerShell as its focus for delivering packages from the distros to computer. Chocolatey is a single, unified interface designed to easily work with all aspects of managing Windows software using a packaging framework that understands both versioning and dependency requirements. Truffle is a development, testing framework for blockchains using the Ethereum virtual Machine (EVM), which makes life as a developer easier. Truffle provides a built-in smart contract compilation, linking, automated contract testing, network management for deploying to any number of public and private networks. The next one is Ganache, formerly

known as TestRPC, is available as a desktop application as well as command-line tool. Ganache is a personal blockchain for Ethereum development that we can use to deploy contracts, develop our applications, and run tests. Ganache allows us to make calls to the blockchain without actually running an Ethereum node. These together help in in the deployment and successful implementation of blockchain.

As mentioned above, blockchain with its secure and distributed work can help find useful solutions to a host of issues. One such issue is voting. Voting is a fundamental part of democracy. It is every citizen's right to vote and have a crucial say in the selection of the next government. Although the present voting systems with the EVM's are in place, the reliability of which raises a few questions. A few lingering doubts for first time voters as well as other voters are A) How transparent is the voting system? B) How safe are my votes? Are they changed and duplicated? Is my vote protected? The Ideal voting system is one where votes once cast cannot be duplicated and the identity of the voters is not disclosed. Fortunately, blockchain in association with machine learning(ml) helps to achieve this. The introduction of fingerprint matching as a security has helped to replace the existing and flawed system of being marked by ink on the finger. The fingerprints are unique for each individual and it helps with the security aspects.

PROPOSED SYSTEM AND ITS ADVANTAGES

The main motivating factor behind this system was to provide a clean, trustable and safe voting environment for voters with the help of blockchain. The continued trend of low voter turnouts has been a huge cause for concern for the governments. Blockchain with its immutable data provides that security for votes. As mentioned above Blockchain, is a field on the rise and many organizations have recognized and started to use it. If the system is made more trustworthy and reliable, then it will help to make the politicians more accountable. This is one step towards true democracy.

The proposed voting system is one where, the fingerprint, which is unique for every individual is taken as input and stored in the database. All the other details such as name, age, address is also taken as input and stored in the database. When a voter comes to cast his vote, the voter is required to place his fingerprint for authentication. The fingerprint taken currently is looked up for a match with the existing fingerprints stored in the database. Upon match of the fingerprints, the details such as name, age, address of that particular voter is displayed on the monitor. In the next step, the voter has to choose the candidate he/she wants to vote for and all that is left is to cast the vote. Once the whole voting process is completed successfully, the results which were counted real-time will be displayed at the end.

This voting system also helps to cut down on the expenses that the election commission has to spend on significantly. The man power required to handle this system is significantly less than the existing system. A political party in Denmark has used voting with blockchain for its internal voting and the resource link has been shared https://followmyvote.com/danish-political-party-unleashes-blockchain-voting/. Current voting system do not instill the confidence in voters to come out and cast their votes. So how does blockchain differ from the existing systems? Every record on a blockchain appears on a peer-to-peer, distributed ledger, meaning that everyone on the network can view and verify records. For election purposes, each vote would connect to an individual voter, and any discrepancies would be solved by simply reviewing the ledger — reducing voter fraud. Further, voters' identities would be completely anonymous. Though individual votes would be publicly available, voters would be masked behind an encrypted key of random numbers and letters. This would ensure privacy and security — much more than traditional ballot boxes — and reduce the possibility of voter suppression: if you can't identify voters, you can't target them. A blockchain is also immutable: that is, every record on the ledger is permanent and unalterable. That would make auditing easier, tampering almost impossible and lost or missing votes a thing of the past. Every ballot would be final and guaranteed. But perhaps the important security advantage of the blockchain is decentralization. Because the ledger is distributed across a public network, it has no single storage base. For hackers to compromise the network, they would need to hack a majority of the "blocks" and complete the hack before new blocks are created, like an extremely complex game of whack-a-mole. This makes hacking any blockchain hard. For a large blockchain — such as an election blockchain — it would be even harder.

IMPLEMENTATION:

The necessary applications, frameworks and platforms are downloaded and installed for working on the Ethereum Blockchain. Initially a directory is created. The truffle framework has an inbuilt package known as pet-shop, which is extracted or unboxed in the newly created directory. This package consists of the framework to work on the contracts, migrations, src and test folders. Two solidity files are created 1) Election. Sol – for the main code 2) migration. Sol – for migration. Smart contract is defined in the contracts folder. To check for the working of smart contract, a smoke test is done. The election.sol which contains the main code, consists of the voters and candidates which are declared as private and public respectively. Addcandidate () is created which is used to add candidates. The voting ballot is a counter algorithm where the votes that are cast are counted dynamically. The migration.sol file ensures the eligibility and validity of the voters. The truffle.js file should always be in the text format. Before migrating the smart contracts, ensure ganache, the client-side application, is kept running. The network ID, port number and IP should match with that of Ganache's. In the contracts sub-folder under the build folder, holds the code for the development of the user interface. Windows PowerShell is the suitable command-line for use. Switch Windows power Shell to NPM (Node Package Manager) mode. Run the Ethereum server using the testRPC. To migrate the contract, run the command truffle migrate –reset. In case to verify the address of the node, switch to truffle console. Once the migration is done, run the command run npm dev. Web browser pops up after this command.

The fp-sensor is interfaced with the Arduino Uno R3 ATmega328P Microcontroller. The 4 pins from the finger-print sensor, namely VCC, TX, RX, Ground are connected to the respective pins in the Arduino Microcontroller, the VCC is the 5V supply to the fingerprint sensor from the Arduino Microcontroller, TX stands for transmitter and RX stands for receiver pins and Gnd stands for Ground. The finger-print sensor is capable of storing up to 128 finger-print templates. The Microcontroller-fingerprint sensor interface has two processes, i.e., Storing a new finger-print(template) with a unique ID on the finger-print sensor and checking for any matching fingerprint that's already been stored in the finger print sensor and if yes then display the Finger-print template id along with the confidence level of the matched finger print template with the input finger-print.

The primary purpose for the inclusion of finger print sensor is voter authentication, since our application is decentralized, we need to connect our Microcontroller to the internet, for the purpose of which we are using an Ethernet Shield (W5100). The match of any fingerprint with the fingerprint templates will be notified to the PHP page along with the template id through the Ethernet Shield. The PHP page acquires user credentials through the template id from the phpMyAdmin database. phpMyAdmin is a free software tool written in PHP, intended to handle the administration of MySQL over the Web. phpMyAdmin supports a wide range of operations on MySQL, frequently used operations (managing databases, tables, columns, relations, indexes, users, permissions, etc.) can be performed via the user interface, while we still have the ability to directly execute any SQL statement. The phpMyAdmin database runs on the local server created by the XAMPP Software, this local server works on desktop or laptop, XAMPP software package contains Apache distributions for Apache server, MariaDB, PHP, and Perl. And it is basically a local host or a local server. XAMPP server software gives suitable environment for testing MYSQL, PHP, Apache and Perl projects on the local computer. The php page's source code will be stored in the same folder as the XAMPP in the Laptop or desktop which is out Local server.

```
contract Election {// Model a
Candidate
    struct Candidate {
        uint id;
        string name;
        uint voteCount;
    }
    //mapping (address => uint)
public balances;
    // Store accounts that have
voted
```

```
    mapping(address => bool) public
voters;
    // Store Candidates
    // Fetch Candidate
    mapping(uint => Candidate)
public candidates;
    // Store Candidates Count
    // voted event
    event votedEvent (
        uint indexed _candidateId
    );
    uint public candidatesCount = 0;
    function addCandidate (string
memory cname) private {
        candidatesCount++;
        candidates[candidatesCount]
= Candidate(candidatesCount, cname,
0); //Add Candidate parameters
    }
    constructor() public {
        addCandidate("Candidate 1");
//Add Candidate 1 to vote
        addCandidate("Candidate 2");
//Add Candidate 2 to vote
    }
    function vote (uint
_candidateId) public {

require(!voters[msg.sender]);
// require that they haven't voted
before
        require(_candidateId > 0 &&
_candidateId <= candidatesCount); //
require a valid candidate
        voters[msg.sender] = true;
// record that voter has voted

candidates[_candidateId].voteCount
++;  // update candidate vote Count
        emit
votedEvent(_candidateId);   //
trigger voted event
    }
}
```

The above code is a solidity file named as Elections.sol which acts as a vote counter and adds the number of candidates by a method called addCandidate dynamically. It ensures one vote per voter and deploys a dynamic json file which is extracted from Truffle framework (petshop).

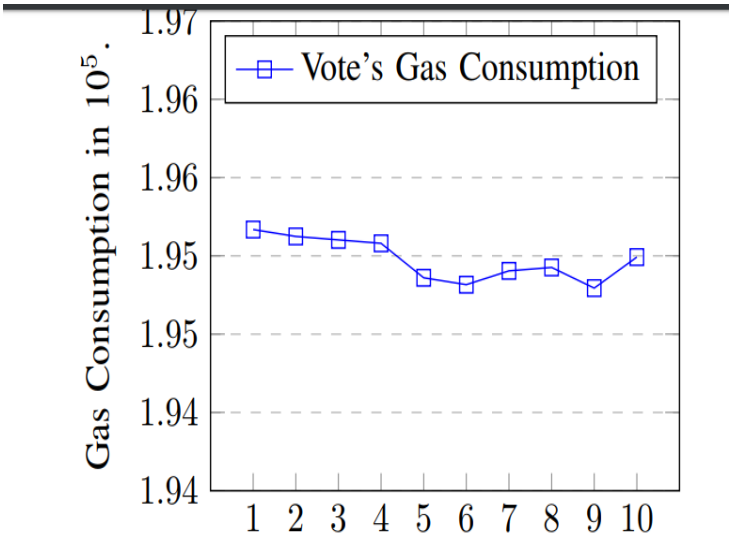

Fig. 3: Individual Initial Vote's GAS usage (x 10^5)

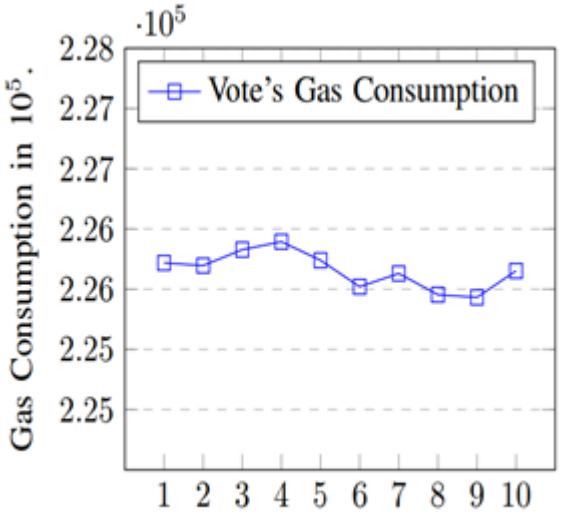

Fig. 5: Individual Altering Vote's GAS usage (x 10^5)

CONCLUSION

By building this proposed smart contract of ours, we have succeeded in moving e-voting to the blockchain platform and we addressed some of the fundamental issues that legacy e-voting systems have, by using the power of the Ethereum network and the blockchain structure. As a result of our trials, the concept of blockchain and the security methodology which it uses, namely immutable hash chains, has become adaptable to polls and elections. This achievement may even pave the way for other blockchain applications that have impact on every aspect of human life. At this point, Ethereum and the smart contracts, which made one of the most revolutionary breakthroughs since the blockchain itself, helped to overturn the limited perception of blockchain as a cryptocurrency (coin), and turned it into a broader solution-base for many Internet-related issues of the modern world, and may enable the global use of blockchain. E-voting is still a controversial topic within both political and scientific circles. Despite the existence of a few very good examples, most of which are still in use; many more attempts were either failed to provide the security and privacy features of a traditional election or have serious usability and scalability issues. On the contrary, blockchain-based e-voting solutions, including the one we have implemented using the smart contracts and the Ethereum network, address (or may address with relevant modifications) almost all of the security concerns, like privacy of voters, integrity, verification and non-repudiation of votes, and transparency of counting. Yet, there are also some properties that cannot be addressed solely using the blockchain, for example authentication of voters (on the personal level, not on the account level) requires additional mechanisms to be integrated, such as use of biometric factors. The prominence of distributed systems stands out especially when considering the mitigation of the risk that storing the registrations at a central location (office). This can always somehow allow officials to have the opportunity to physically access to the vote records, which could lead to corruptions and cheatings by the authorities. Additionally, in today's connected world, with the concept of the Internet of Things (IoT), expectedly, many non-computer devices will gain access to the Internet. While we are still working on a mobile phone application as a supportive extension to our work to widen the usability; It is important to note that, apart from phones and tablets; air conditioning devices. cars, chairs, clothes, refrigerators, televisions, and many other everyday objects are/will be able to directly reach to the internet. In terms of blockchain, it won't be difficult to build such distributed systems when there is such a large network and a reserve processing power. Moreover, if all these devices work together as a grid to shorten the validation period of transactions in a blockchain, we will be able to do most of our online transactions securely, reliably, and effectively, not only in theory but also in practice.

REFERENCES

[1] Kumaresan Mudliar (2018) proposed a comprehensive integration of national identity with blockchain technology, where the national identity of an individual (Aadhar) currently stored in centralized applications, can be with the help of blockchain technology stored in decentralized applications. The scheme also proposed future applications of blockchain with national identity in digital voting, finance, banking, digital healthcare etc.

[2] Rifa Hanifatunnisa proposed a blockchain based e-recording voting system design, where the votes in the voting system is stored in decentralized system with the database spread across multiple users.

[3]Ali Kaan Koç, proposed a e-voting system using Ethereum blockchain, where the voting is carried out in a decentralized system and the voting can be carried out in android devices.

[4] Silvia Bartolucci (2018), proposed a secret share based voting on the blockchain (SHARVOT), where the system uses Shamir's Secret Sharing to enable on-chain, i.e. within the transactions script, votes submission and winning candidate determination. The protocol is also using a shuffling technique, Circle Shuffle, to de-link voters from their submissions.

[5] SungHyun Na (2018), proposed a web-based nominal group technique decision making tool using blockchain, where the system proposes to provide complete anonymity of the identity of the voters in the nominal group discussion technique.

[6] Building blockchain Projects authored by Narayan Prusty, gives a deep insight and detailed explanation on how to build and develop real-time practical DApps using Ethereum and Javascripts.