WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Secure E-Voting System Using Blockchain Technology

Dr. C Rangaswamy[1], Gudipati Ankitha[2], Pallapu Saran Kumar[3], Machireddy Gari Gunasekhar Reddy[4]

[1]Associate Professor, Department of Electronics and communication Engineering, SJC Institute of Technology, Chikkaballapur, India crsecesait@gmail.com

[2]Department of Electronics and communication Engineering, SJC Institute of Technology, Chikkaballapur, India ankithagudipati@gmail.com

[3]Department of Electronics and communication Engineering, SJC Institute of Technology, Chikkaballapur, India Sarankumar.p0711@gmail.com

[4]Department of Electronics and communication Engineering, SJC Institute of Technology, Chikkaballapur, India gunasekhar3518@gmail.com

ABSTRACT-

To eliminate duplications and inconsistencies, electronic voting has gradually replaced paper-based and electronic machine voting. According to the historical viewpoint offered in the previous two decades, the security and privacy problems discovered over time may be the reason it hasn't been as effective. In order to guarantee the security of the data, this project offers a framework using efficient hashing techniques. In this project, the idea of block creation and block sealing is presented. The addition of block sealing makes the blockchain adaptable to the polling process' requirements. It is advised to adopt a consortium blockchain, which guarantees that a governing organisation (such as an election commission) owns the blockchain and that no unauthorised access will occur can be produced externally. The framework put out in this project examines the usefulness of hashing algorithms, the construction and sealing of blocks, the accumulation of data, and the declaration of results using an adaptable blockchain technique. This project offers an enhanced implementation of the electronic voting system and claims to understand the security and data management difficulties of blockchain.

Key words: Electronic voting, Blockchain

INTRODUCTION

A well-respected phenomenon for representing public opinion in the creation of electoral bodies is the will of the people. The parliament and college unions are two examples of these electoral bodies. Voting has become a means of expressing the will of the people when a decision needs to be taken from a range of options over time. The voting tool has contributed to increasing people's confidence in the decisions made by a majority vote. This undoubtedly contributed to the democratization of the electoral system and the importance of voting in choosing parliaments and governments. Out of a little over 200 counties in 2018, 167 have some form of democracy, whether perfect, imperfect, hybrid, etc. It is crucial that they maintain their confidence in the vote and the voting process since public faith in democracies is rising. The voting system developed as a tool to assist citizens in choosing their representatives, who then form governments, as a result of growing confidence in democratic institutions.

People's trust in the government to handle national security, domestic issues including health and education policy, international affairs, and taxation is strengthened by the power of representation.

In various parliamentary democracies, institutions like the "Election Commission" were established to improve the voting process. The institutions created voting districts, the electoral system, and voting systems in addition to setting up the procedures and laws for holding elections in order to support the holding of open, free, and fair elections. Since the inception of the voting system, the idea of secret voting has been established. Since more people are putting their faith in democratic processes, it is crucial to maintain that voter confidence doesn't slip. There have been various instances in recent years where it was recognised that the voting process was not entirely hygienic, that it had several problems, including openness and fairness, and that the public's will was not respected not shown to be properly measured and communicated in terms of government formation. These instances are prevalent in nations like Nigeria, India, Brazil, Pakistan, and Bangladesh.

LITERATURE SURVEY

FREYA SHEER HARDWICK et al. [1]: The author presents the first step of e-voting using a decentralised e-voting system with voter privacy rights, employing smart contracts and PKIs for verification and digital signatures that are extremely reliable and effective. The protocol has been created to follow basic e-voting principles, allow for a certain amount of decentralisation, and allow voters to modify or update their votes.

King-Hang Wang, Subrota K. Mondal, Ki Chan and Xiaoheng Xie [2] Their research demonstrates that electronic voting has been a field of study for more than 30 years. However, it is still some time away from being seriously applied. Providing a secure solution and winning the voters' faith in its use are the main obstacles. By examining these difficulties, we want to provide an in-depth analysis of electronic voting in this essay. We compiled the numerous security specifications listed in the literature that enable researchers to create secure systems.

We looked at a few electronic voting systems from both the literature and the real world. We also looked at several e-voting usability studies to learn how a system may be user-friendly.

D. A. Gritzalis [3] Their research demonstrates that electronic voting, or "e-voting," is viewed as a way to deepen and improve democracy in a contemporary information society. E-voting must first adhere to the current legal and regulatory environment. Additionally, the technical implementation of e-voting should guarantee that user needs are met. As a result, this paper has two goals. First, list the general constitutional requirements that should be taken into consideration while developing an electronic voting system for general elections. This group will determine the exact design guidelines for an electronic voting system that is acceptable in court. The second step is to determine, utilising the Rational Unified Process, what a sufficient level of e-voting system security entails.

Shekhar Mishra et al. [4] Their research demonstrates that electronic voting systems are employed in general and state elections in India; however, the biggest disadvantage is the security concern. Voter impersonation occurs when a person who is not qualified to vote in an election-casts a ballot during the voting period using the name of someone who is. Due to its ease of fabrication and assembly, our idea uses the 32-bit ARM 7 processor as the host. In this procedure, the Aadhaar number and biometric data associated with it are saved in an ARM7 microprocessor, which then verifies the data on hand. This will be used to obtain the user's fingerprints.

METHODOLOGY

The system architecture is divided into three-parts, namely

1. Pre-voting phase.
2. Voting phase.
3. Post-voting phase.

Pre-voting phase:

1. In order to vote in the stated chain of networks, a user must first register with the network.
2. The user must construct an ID using the information on his official voter ID and connect to the network. At this point, the previous blocks check the new block's (user's) information to see if it matches the information in the database or not. The network permits the new block to proceed to the next round of the voting process if the validation of the block is successful.
3. The user must use Aadhar authentication to confirm his identity before moving on to the next step of the voting procedure.

Voting phase:

1. The user is permitted to vote following successful Aadhar authentication. The user's vote will be encrypted using public key encryption, and once the user has cast his vote, he cannot log in again and cast another vote on the network. This re-voting using the same ID is prohibited by smart contracts.
2. The vote cannot be manipulated. If a person wants to change their vote after casting a ballot, it is nearly impossible because doing so would require altering the entire system of blocks and requiring authentication from every node in the network.

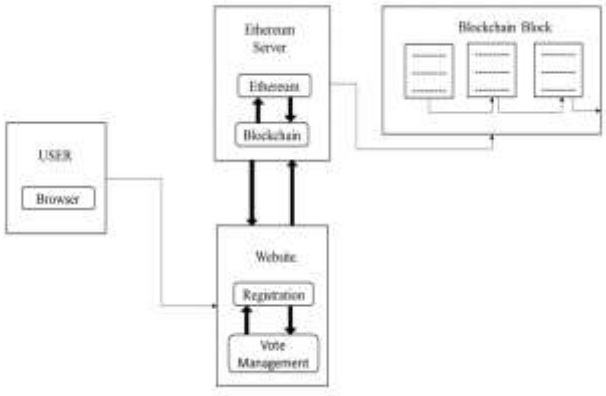

Figure: System block diagram.

Post-voting phase:

The user can use the website to see which political party won the election poll after it has been completed.

Algorithm:

a. SMTP (for sending OTP to email-id):

1. **Composition of Mail:** A By creating an electronic mail message using a Mail User Agent (MUA), a user can send emails. A programme called Mail User Agent is used to send and receive mail. The body and header are the two components of the communication. The message's primary component is its body, while the header contains details like the sender and recipient addresses. Additionally, the header contains descriptive details like the message's subject. In this instance, the message content resembles a letter, and the header resembles an envelope containing the address of the recipient.
2. **Submission of Mail:** The mail client then sends the finished email to the SMTP server using SMTP on the TCP port after finishing the email's composition.
3. **Delivery of Mail:** The recipient's username and the domain name are the two components of an email address. For instance, sarankumar.p0711@gmail.com, where "gmail.com" is the domain name and "sarankumar.p0711" is the recipient's username. Mail will be sent to the Mail Transfer Agent (MTA) if the recipient's email address's domain name differs from the sender's domain name. The MTA will locate the target domain and relay the email there. To retrieve the target domain, it looks up the MX record in the Domain Name System. The IP address and domain name of the recipient's domain are listed in the MX record. The MTA establishes a connection with the exchange server to relay the message after locating the record.
4. **Receipt and Processing of Mail:** Once the incoming message is received, the exchange server sends it to the mail delivery agent (incoming server), which archives the email and makes it available for user retrieval.
5. **Access and Retrieval of Mail:** Using MUA (Mail User Agent), it is possible to recover the email that was saved in MDA. MUA is accessible with a login and password.

b. SHA-256 Bit Encryption Algorithm:

1. The SHA-256 algorithm is a variant of the SHA-2 (secure hash algorithm 2) method, which was developed by the National Security Agency in 2001 to replace the SHA-1 algorithm. A 256-bit value is produced by the patented cryptographic hash algorithm SHA-256.
2. In Data is changed into a secure format through encryption so that it cannot be read unless the recipient has a key. The data can be as big as you like when it's encrypted, and it's frequently the same size as unencrypted data. Contrarily, in hashing, data of any size is converted to data of a specific size. For instance, SHA-256 hashing would reduce a 512-bit string of data to a 256-bit string.
3. The hashed data is altered during cryptographic hashing such that it is utterly unintelligible. The above-mentioned 256-bit hash cannot easily be changed back to its 512-bit original state. Verifying the content of data that needs to be kept secret is the most frequent justification. Hashing, for instance, is used to check the validity of secure messages and files. A secure file's hash code can be made available to the public so that users who download it can verify it is the real deal without the file's contents being made known. Similar to how digital signatures are verified, hashes are used.

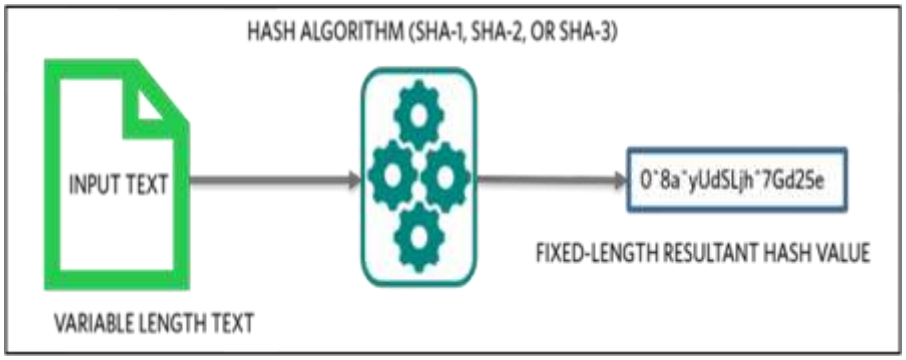

Figure: SHA-256 Bit Algorithm Working.

4. A crucial use for cryptographic hashing is password verification. A plain-text document containing user passwords is a tragedy waiting to happen; any hacker who gains access to the document will find a wealth of unprotected credentials. Because of this, it is safer to keep password hash values on hand. The hash value generated from a user's password is then calculated and compared to the table. A valid password can be used to gain access if it matches one of the hashes previously saved.

c. Merkel Hash

1. To determine if a transaction is included in the block, a Merkle tree adds up all of the transactions in a block and creates a digital fingerprint of the full set of operations.
2. To create Merkel trees, pairs of nodes are continuously hashed until only one hash—the Merkel root—remains. Using transaction IDs, which are hashes of the individual transactions, they are constructed from the ground up. Every leaf node is a hash of transactional data, while every non-leaf is a hash of its previous hash.

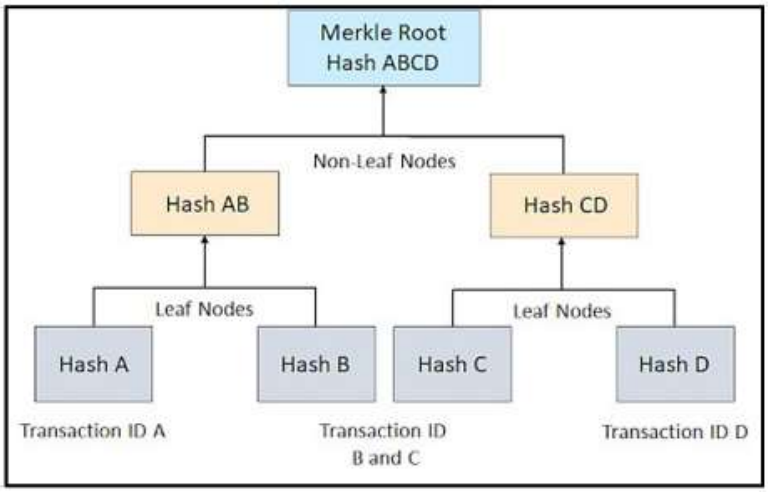

Figure: Merkel Hash Working.

d. Functional modules:

The functional modules of the electronic voting using blockchain are:

1. **Input Aadhar number:** Voters must use their Aadhar number to register their votes prior to the elections in order to cast ballots. Voting is restricted to registered candidates only. Voters must enter their Aadhar numbers as part of the voting process.
2. **OTP Validation:** After providing a valid Aadhar number, the entry is validated with an OTP that is delivered to the candidate's email address provided at registration.
3. **Private key authentication:** After the OTP validation process has been successfully completed, the candidate can now see the party names and symbols along with the nota. When a user clicks on the party for which they intend to cast a ballot, a private key authentication prompt appears, requiring them to enter the private key they got from their registered email address. Private key creation is accomplished through hashing.
4. **Casting vote:** After the OTP validation process has been successfully completed, the candidate can now see the party names and symbols along with the nota. When a user clicks on the party for which they intend to cast a ballot, a private key authentication prompt appears, requiring them to enter the private key they got from their registered email address. The vote will be recorded upon the private key's successful validation, and voters can view the results of the election after casting their ballots.

TESTING AND RESULTS

a. Unit Testing:

1. Tests that confirm the functionality of a particular area of code, typically at the function level, are referred to as unit tests, sometimes known as component tests. This occurs typically at the class level in an object-oriented system, and the constructors and destructors are covered by the bare minimum unit tests. In order to lower the risks, expenses, and duration of software development, unit testing is a software development method that comprises the coordinated use of a wide range of defect prevention and detection strategies. The functions that were tested at the time of programming are displayed in the following unit testing table. All the modules that were tested are listed in the first column, and the test results are listed in the second. The results of the tests show whether the functions are producing the correct results for the specified inputs.
2. Tests for Function Name The user's ability to cast his vote through the website indicates that the results of feeding the legitimately registered Aadhar number and authorized user using blockchain hash techniques were successful.

Function name	Test results
Feed enrolled Aadhar number	Tested for different input of Aadhar numbers verification
Verify the details	Authorizing only valid users to vote
Display result	Output is to cast vote only once by a single user

Table: Function Name and Test Results.

b. Integrating testing:

1. Any sort of software testing that aims to validate the interfaces between components in comparison to a software design is referred to as integration testing. Software components can either be put together incrementally or all at once ("big bang"). The former is typically seen as a better practice because it makes it possible to identify and address interface problems more rapidly.
2. Integration testing looks for flaws in how integrated components (modules) interact and communicate with one another. Up until the software functions as a system, ever-larger groupings of tested software components that match components of the architectural design are merged and tested.

c. Validation Testing:

1. After integration testing, the software is finished and put together as a package. Interfacing mistakes have been found and fixed. Numerous definitions exist for validation testing; in this instance, the testing verifies that the software performs as the client may reasonably anticipate.
2. Verification and validation (V&V) is the process of ensuring that a software system complies with specifications and serves the intended purpose in the fields of software project management, software testing, and software engineering. Another name for it is software quality control.

d. User Acceptance Testing:

Performance 1. The user is the star of an acceptance test's performance. User motivation and expertise are essential for the system to operate well.

2. The newly constructed system performed as expected during the tests mentioned above. The following test case design was used to execute all of the aforementioned testing methodologies.

1. Unit Test Results:

Input Aadhar test case:

Table: input Aadhar test case.

Test case	1
Name of the test	Input Aadhar
Input	Valid unique ID
Expected output	Input Aadhar feed by the user
Actual output	Valid Aadhar number is accepted as enrolled in the database
Result	Successful

Email OTP authentication test case:

Table: Email OTP authentication test case.

Test case	1
Name of the test	Email OTP authentication
Input	Valid/ enrolled email-id
Expected output	Obtain OTP to the registered email
Actual output	Receiving unique OTP from the enrolled email ID
Result	Successful

Private key verification Test case:
Table: Private key verification Test case.

Test case	1
Name of the test	Private key verification
Input	Valid/ enrolled email ID
Expected output	Obtaining unique hash value
Actual output	Receiving unique hash value blockchain from the enrolled email ID
Result	Successful

2. RESULTS

Figure: Private key sent to email-id

Figure: Signing of Vote Successfully

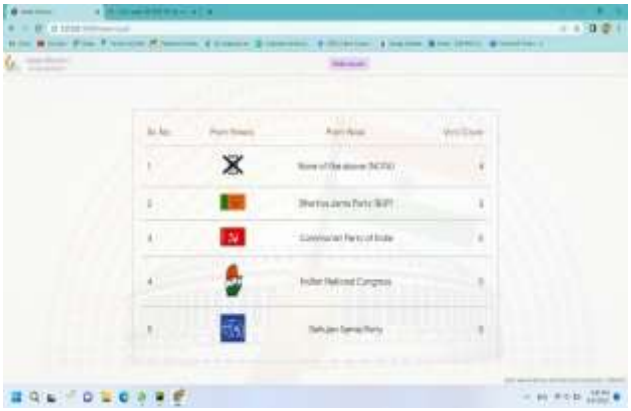

Figure: Vote Count

CONCLUSION

The suggested framework fully secures the e-voting system and uses the Ethereum blockchain and smart contracts to further fortify the system's security. The use of blockchain technology eliminates vote tampering and offers voters' ballots privacy and integrity. With the use of multiple security algorithms like SHA-256, Merkel hash, and SMTP prototype, smart contracts ensure that each voter can cast a single vote using their unique identification number (Aadhar number). This increases the security of the system. As a result, the voter has the ability to cast their ballot from anywhere they may be, providing the system with high security standards as well as easy and simple methods of voting.

REFERENCES

[1] Freya Sheer Hardwick, Apostolos Gioulis, Raja Naeem Akram, and Konstantinos Markantonakis, E-Voting with Blockchain: An E-Voting Protocol with Decentralization and Voter Privacy, Vol 7, Issue 6, 2018, pp. 1561- 1567.

[2] King-Hang Wang, Subrota K. Mondal, Ki Chan, Xiaoheng Xie, A review of contemporary e-voting: Requirements technology systems and usability, Ubiquitous International vol 1, Issue 1, 2017, pp. 31-47.

[3] D. A. Gritzalis, Principles and requirements for a secure e-voting system, *publication history*, vol 21, Issue 6, 2002, pp. 539-556.

[4] Shekhar Mishra, Y. Roja Peter, Zaheed Ahmed Khan and M. Renuka, Electronic Voting Machine using Biometric Finger Print with Aadhar Card Authentication, International Journal of Engineering Science and Computing, Vol 7, Issue 3, 2017, pp. 5897- 5899.

[5] Amna Qureshi, David Megias and Helena Rifa-Pous, SeVEP: Secure and Verifiable Electronic Polling System, Vol 7, Issue 6, 2019, pp. 19266- 19290.

[6] RIFA HANIFATUNISA and BUDI RAHARDJO: Block Chain based E-voting Recording System design, 11th International Conference on Telecommunication Systems Services and Applications (TSSA), Vol. 4, Issue 4, 2017, pp. 662–673.

[7] T. A. T. Nguyen and T. K. Dang: Enhanced security in internet voting protocol using blind signature and dynamic ballots, Electronic Commerce Research, Vol. 13, Issue 3, 2013, pp. 257–272.

[8] C. Porkodi, R. Arumuganathan, and K. Vidya, “Multi-authority electronic voting scheme based on elliptic curves,” International Journal of Network Security, Vol. 12, Issue 2, 2011, pp. 84–91.

[9] M. A. Smith, S. S. Monfort, and E. J. Blumberg: Improving voter experience through user testing and iterative design, Journal of Usability Studies, Vol. 10, Issue 4, 2015, pp. 116–128.

[10] Ayed, Ahmed Ben: A conceptual secure blockchain-based electronic voting system." International Journal of Network Security & Its Applications 9, Vol. 3, Issue 6, 2017, pp. 01-09.