iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 11 **Issue:** VII **Month of publication:** July 2023

DOI: https://doi.org/10.22214/ijraset.2023.54573

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 11 Issue VII Jul 2023- Available at www.ijraset.com

E-Voting System using Blockchain Technology and Homomorphic Encryption

Danush Kanchi[1], Hitesh N[2], Kushal A C[3], Yuvan Yash[4], Dr. C S Jayasheela[5]
[1, 2, 3, 4]*Dept. of ISE, BIT, Bengaluru Karnataka, India*
[5]*Assistant Professor, Dept. of ISE, BIT, Bengaluru, Karnataka, India*

Abstract: ***Voting is an essential component of any democratic society but traditional systems often face obstacles like inefficiency, human error, and transparency issues thus hindering credibility. This inspired our aim to create a proof-of-concept for an e-voting system that leverages blockchain technology coupled with homomorphic encryption as means to address these impediments. In doing so, the proposed system intends to restore trust in electoral processes by improving their transparency, integrity, and efficiency. Blockchain technology provides a verifiable account and a convenient database, culminating in the smooth conduct of elections for the general public, and fostering confidence in the precision of the outcomes obtained. Homomorphic encryption secures votes utilizing encrypted data computations allaying public concerns on voter anonymity. It also affords utmost privacy security of individual votes inevitably cutting costs since elections no longer require physical ballots or polling booths.***
Keywords: E-voting system, blockchain technology and homomorphic encryption.

I. INTRODUCTION

Voting is the backbone of democracy and forms the basis on which the governance of a country rests. With the advent of technology and the increasing impact of digitalization on the youth of the country, it has become imperative to modernize the traditional voting process. The current electoral procedure has encountered a multitude of discrepancies, and it is imperative to implement technological advancements to amend the current process. Traditional voting systems, such as paper-based ballots, are prone to human errors and can be time-consuming. Electronic voting has gained attention due to its potential to reduce costs and ensure the integrity of the electoral process through privacy, security, and compliance requirements. The manual counting of ballots can be cumbersome and can lead to errors, which can affect the outcome of an election. Electronic voting systems offer a more efficient and accurate way of casting and counting votes, which can reduce the likelihood of errors and enhance the transparency of the electoral process. However, the current method of electronic voting has been proven to be unsatisfactory with respect to transparency. In addition, the deficiency of any external influence with regard to the government's execution of the vote recounting process presents a challenge in guaranteeing the authenticity of the electoral process for the voters. Blockchain technology offers a promising solution to the transparency problem in electronic voting. The utilisation of blockchain technology has the potential to furnish a lucid and verifiable documentation of the electoral process, thereby augmenting the probity and credibility of the said process. The adoption of blockchain-based electronic voting systems can ensure that voters are assured of the integrity of the electoral process and can lead to a more transparent and trustworthy democratic process. Homomorphic encryption is a cryptographic methodology that can execute computations on encrypted data without requiring decryption. It is particularly relevant in electronic voting (e-voting) technology, where privacy and security are critical. In e-voting systems that utilize blockchain technology, homomorphic encryption can be employed to ensure the confidentiality of votes while maintaining the ability to perform certain computations on the encrypted data. The combination of homomorphic encryption and blockchain technology can establish a secure and open platform for electronic voting systems to execute electoral processes.

II. RELATED WORK

Siddharth Rajput et al. [1] emphasize the significance of a decentralized and transparent record-keeping mechanism that is shared among various participants. It ensures that every transaction undergoes verification by all parties involved. Once data is recorded in the blockchain, it becomes immutable and hence, resistant to alteration. Therefore, the blockchain serves as a comprehensive transaction ledger. The review covers topics such as smart contracts, consensus algorithms, and scalability challenges encountered in blockchain networks.

Nazmun Nahar et al. [2] highlights there has been a significant surge in the demand for blockchain technology across various sectors and fields. Blockchain functions as a shared digital database within a computer network. In the context of decentralized cloud storage, data undergoes encryption using cryptographic algorithms and a user's private key. Implementing blockchain for data storage enhances system security, reduces the risk of attacks, and eliminates single points of failure.

Jilles Hasenberg et al. [3] make a valuable contribution to the discourse on electronic voting, which is often touted as a more secure option than traditional voting systems, thereby instilling greater confidence and transparency among the populace. However, several challenges exist in managing electronic voting procedures. Homomorphic encryption, a cryptographic algorithm, is foundational to many schemes utilized in electronic voting. These schemes utilize homomorphic properties and offer different encryption implementations for analysing and applying them in the context of electronic voting. Through deductive methodology, logical analytics, and exploratory research, three homomorphic encryption schemes are described and evaluated. These schemes, while employing different algorithms, maintain efficiency and do not compromise security. Furthermore, the significance of a distributed architecture is emphasized as an additional measure for handling election information. The use of homomorphic encryption in electronic voting ensures secure computation on encrypted data while preserving voter privacy.

Divya Rathore et al. [4] have developed a remote internet- based voting system that permits voters to participate in the electoral process from the comfort of their homes, without the need for a physical invigilator. The system leverages blockchain technology and a cryptosystem to guarantee the dependability, security, and anonymity of both the voters and their votes. Moreover, it prevents the prediction of the winning party until the results are announced, thereby enhancing citizen trust through transparency and improved voting system security. Ongoing research in this field aims to merge legal and technical aspects to create a self-sustaining proofreading system in various digital services.

Mingli Zhang et al. [5] have noted that authentication mechanisms based on trusted authorities are frequently employed in traditional systems. Although such mechanisms have proved effective, their security may not meet the demands of modern technologies. The blockchain is a centralised mechanism that requires a trusted authority for trust. The use of data encryption, timestamping, and consensus mechanisms facilitates this process. The OTP scheme is utilized within the blockchain as the OTP verifier. The security of the authentication protocol proposed is evaluated using a set of security criteria. The scheme effectively resists replay attacks, brute force attacks, and OTP forgery attacks. A comparative assessment of the proposed OTP scheme and other representative OTP schemes is conducted using designated evaluation criteria. It has been proven that the recommended plan is better than others in terms of both performance and security, as seen in the comparative experiments run on Hyperledger Fabric.

III. PROBLEM DEFINITION

The current method of physical voting is susceptible to manipulation, requires significant manual labour, and presents challenges for individuals with disabilities. This paper endeavours to address the aforementioned concerns by developing an electronic voting mechanism that incorporates blockchain technology and homomorphic encryption. The aim is to facilitate eligible voters in securely casting their votes from their computers while ensuring the transparency, integrity, and immutability of the voting process.

Through the elimination of physical ballot papers and polling booths, the proposed system seeks to improve efficiency and cost-effectiveness, all while ensuring a secure and accurate method for voters to participate in elections.

IV. OBJECTIVES

1) Develop a tamper-proof E-voting system utilizing Blockchain technology and AES with Homomorphic encryption.
2) Enable eligible voters to cast their ballots using their computers by employing a one-time password sent to their verified email ID and SMS.
3) Ensure the integrity and transparency of the voting process by securely recording all transactions in an immutable manner.
4) Improve efficiency and cost-effectiveness of the voting process by eliminating the requirement for physical ballot papers and polling booths.
5) Propose a secure, transparent, and precise system that enables voters to cast their votes while upholding the integrity of the entire election process.

V. SYSTEM ARCHITECTURE

The proposed architectural framework, as presented in Figure 5.1, comprises three indispensable modules, namely, the Election Officer, the Booth Manager, and the Voters. The Election Officer module bears the responsibility of integrating candidate details, administering booth managers, allocating booths, and validating the election outcomes. The Booth Manager module is entrusted with the task of authenticating voter information during the voting procedure. Voters are assigned a specific identification number that authorizes them to log in to the voting system using a One Time Password (OTP) and vote for their preferred candidate.

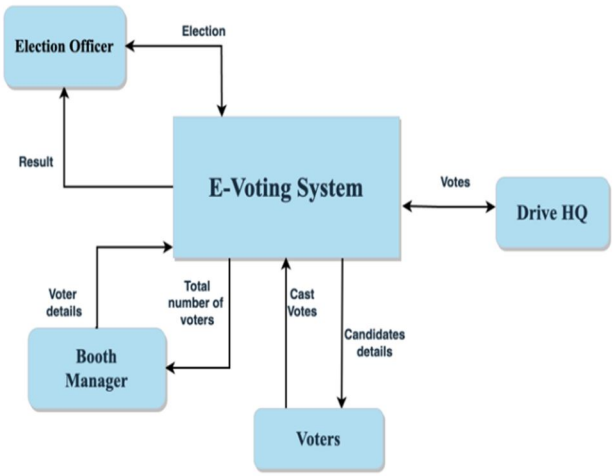


Figure 5.1 System Architecture

The election officers have the power to scrutinize, modify, add or remove various factors related to the candidates like their designation, age, political affiliation, and geographical location as well as the details of the voting booths such as their reference number, district location, and the identity of the booth manager in charge. Similarly, details pertaining to the voting booths, including their reference number, district location, and the identity of the booth manager in charge, may be viewed or modified as illustrated in the Figure 5.2. The election officer is vested with the authority and the key required to decrypt the individual votes cast for each candidate from various polling stations, and subsequently announce the winner of each electoral district.

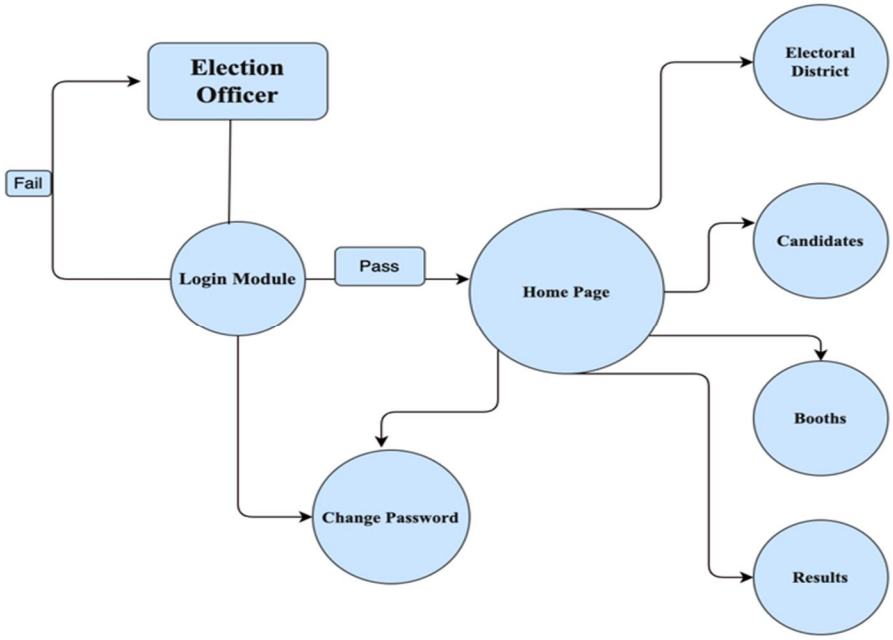


Figure 5.2 Data Flow Diagram of Election Officer

Booth managers possess booth-specific information such as the booth reference number, location and number of candidates participating in the election. They can access and manage the details of voters assigned to their booth, including adding or removing voters from the list. During the voting process, booth managers can access the overall number of votes, indicating the total number of voters who participated. However, the specific number of votes for each candidate remains encrypted, as illustrated in Figure 5.3.

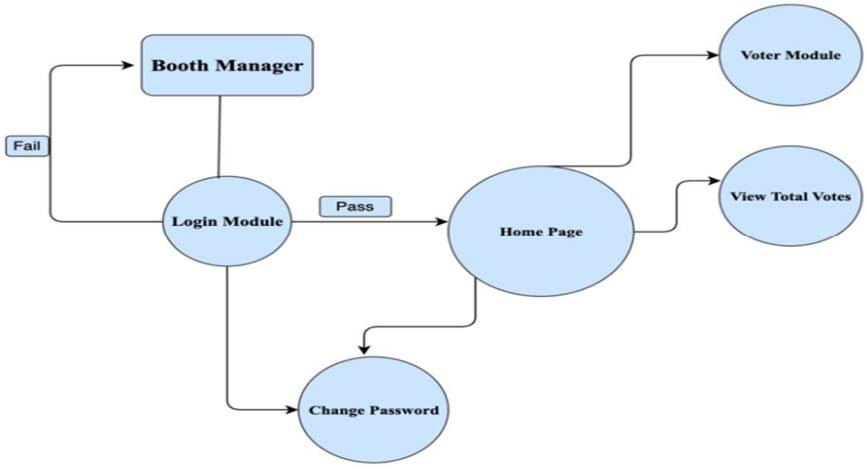


Figure 5.3 Data Flow Diagram of Booth Manager

The exhibition of voter information is dependent on their designated booth, with the mandatory validation of voter identities to ensure compliance with the relevant booth and abstention from voting.
Upon confirmation of eligibility, voters can proceed to cast the vote for their candidates of choice, with secure storage of data being implemented as depicted in Figure 5.4. vote is encrypted using AES encryption algorithm is employed, with a further encryption process through the Paillier encryption algorithm to enable mathematical operations such as addition on the encrypted data. Additionally, the system integrates blockchain technology to securely store the growing list of records, or blocks, through cryptographic links.

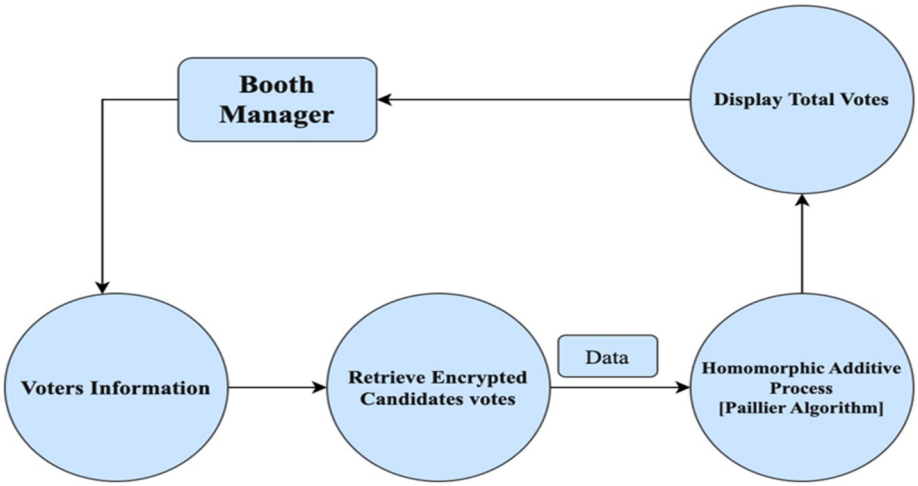


Figure 5.4 Data Flow Diagram of Viewing Votes

In the proposed e-voting system, voter details must be displayed based on their assigned booth, ensuring the validation of their identity and whether they have already cast their vote. The system conducts a prior verification to ascertain if the voter belongs to the designated booth and verifies their voting status. The encrypted vote is stored securely and added to the candidate's tally. The voting process is shown in Figure 5.5. Homomorphic encryption techniques are used to double encrypt the votes for added security.This additional layer of encryption ensures the privacy and integrity of the voting data

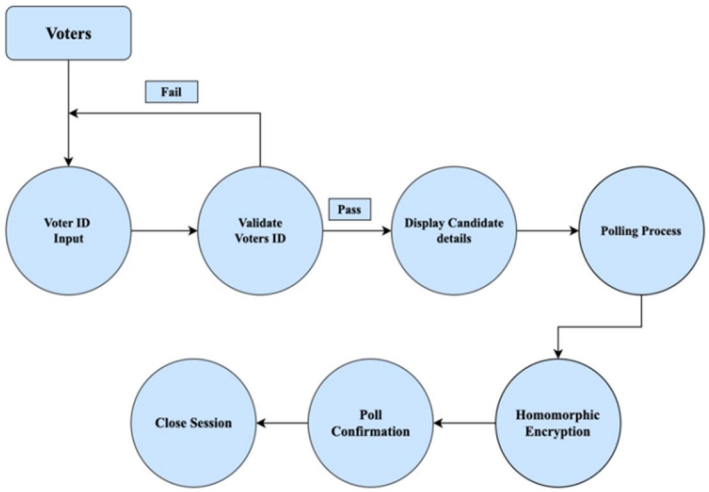


Figure 5.5 Data Flow Diagram of Voting Process

VI. METHODOLOGY

The methodology employed in the Paillier cryptosystem entails several key processes, including key generation, encryption, addition, and decryption. Key generation commences with the specification of the desired key bit length and certainty value, followed by the retrieval of large prime numbers and the calculation of the modulus and its square. Subsequently, a generator value is designated, and the lambda value is determined through the application of a formula involving the prime numbers. The adequacy of the generator is then verified. The process of encryption typically includes applying a specific formula to create ciphertext from a plaintext message and a random value. Homomorphic addition of ciphertexts is enabled through the addition process, where the decryption factor and plaintext sum are computed. Eventually, decryption recovers the initial plain text from an encrypted text. This methodological approach ensures both secure communication and computation, while simultaneously preserving the homomorphic properties intrinsic to the Paillier cryptosystem.

A. Method for Key generation

```
public void KeyGeneration (int bitLengthVal, int certainty) {
bitLength = bitLengthVal;
String a=GetProperty.getProperty("p");
String b=GetProperty.getProperty("q"); p=new BigInteger(a);
q=new BigInteger(b); System.out.println("p "+p); System.out.println("q "+q);
n = p.multiply (q);
nsquare = n.multiply (n);
System.out.println("n "+n);
g = new BigInteger ("2");
System.out.println("g "+g);
lambda= p.subtract(BigInteger.ONE).multiply(q.subtract(BigInteger.ONE)).divide(
p.subtract(BigInteger.ONE).gcd(q.subtract(BigInteger.ONE)));
System.out.println("lamda"+lambda); if(g.modPow(lambda,nsquare).subtract(BigInteger.ONE).divide(n).gcd(n).intValue() != 1) {
System.out.println("g is not good. Choose g again.");
System.exit(1);
}
}
```

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 11 Issue VII Jul 2023- Available at www.ijraset.com

This method is for generating a public-private key pair for the Paillier cryptosystem, which is a public-key cryptosystem used for encrypting and decrypting data.

It consists of following steps:

1) The method takes two input parameters: the bit length of the key (bitLengthVal) and the certainty value (certainty) used to generate the key.
2) The code retrieves two large prime numbers p and q from a property file using a utility called "GetProperty".
3) It calculates n by multiplying p and q.
4) It calculates nsquare by squaring n.
5) It sets g to the value 2.
6) It calculates lambda using the formula: lambda = (p-1)(q-1)/gcd(p-1, q-1), where gcd is the greatest common divisor function.
7) It checks if g is good for encryption by verifying that the gcd of ((g^lambda mod nsquare)- 1)/n and n is equal to 1. If it's not equal to 1, it prints an error message and exits the program.

B. *Method for Paillier Encryption :*

```
 public BigInteger Encryption(BigInteger m, BigInteger r)
{
return;
 g.modPow(m,nsquare).multiply(r.modPow(n, nsquare)).mod(nsquare);
}
public BigInteger Encryption(BigInteger m)
{
BigInteger r = new BigInteger(bitLength, new Random());
System.out.println("r "+r);
System.out.println("result(encry)"+g.modPow(m, nsquare).multiply(r.modPow(n, nsquare)).mod(nsquare));
Return;
 g.modPow(m,nsquare).multiply(r.modPow(n, nsquare)).mod(nsquare);
}
```

This code contains two methods that are used for encryption in the Paillier cryptosystem. The first method takes two input parameters: the plaintext message (m) and a random value (r). It returns the encrypted ciphertext by computing (g^m * r^n) mod n^2, where g, n, and nsquare are components of the public key. This formula is based on the homomorphic property of the Paillier cryptosystem, which allows for addition and multiplication of ciphertexts. The second method takes only one input parameter, which is the plaintext message (m). It generates a random value (r) of bit length specified in the key generation process, prints the value of r and the result of the encryption operation, and returns the encrypted ciphertext by computing (g^m * r^n) mod n^2 using the same formula as the first method.

The encryption operation in the Paillier cryptosystem is randomized, which means that the same plaintext message will have different ciphertexts each time it is encrypted. This randomness is introduced using a random value (r) in the encryption formula.

C. Method for Paillier Addition which is used for viewing total votes:

```
public BigInteger Addition(BigInteger em1,BigInteger em2)
{
BigInteger c = em1.multiply(em2).mod(nsquare);
BigInteger    =g.modPow(lambda,nsquare).subtract(BigInteger.ONE).divide(n).modInverse(n);
Return; c.modPow(lambda,nsquare).subtract(BigInteger.ONE).divide(n).multiply(u).mod(n);
}
```

This method is for performing homomorphic addition on two ciphertexts in the Paillier cryptosystem. Homomorphic addition allows us to add two encrypted messages without first decrypting them.

It consists of the following steps:

1) The method takes two input parameters: the ciphertexts to be added (em1 and em2).
2) It computes the product of the two ciphertexts (em1 * em2) modulo nsquare, where nsquare is the square of n (which is a component of the public key).
3) It computes u, which is equal to (g^lambda mod nsquare - 1)/n, where lambda is a component of the private key.
4) It computes the decryption factor for the sum of the ciphertexts by computing c^lambda mod nsquare - 1 / n, where c is the product of the two ciphertexts calculated in step 2. It multiplies the decryption factor with u, and takes the result modulo n, to obtain the final sum of the plaintext messages encrypted in the two ciphertexts.

The addition operation in the Paillier cryptosystem is homomorphic, which means that the sum of two encrypted messages is also an encrypted message. This method is used to display the total votes casted in a particular booth to the corresponding Booth Manager.

D. Method for Paillier Decryption

```
public BigInteger Decryption(BigInteger c) {
BigInteger u
=g.modPow(lambda,nsquare).subtract(BigInteger.ONE).divide(n).modInverse(n);
Return;
c.modPow(lambda,nsquare).subtract(BigInteger.ONE).divide(n).multiply(u).mod(n);
}
```

This method is for decrypting a ciphertext in the Paillier cryptosystem. The decryption process takes a ciphertext c as input and returns the decrypted plaintext message.

It consists of the following steps:

1) The method first computes u, which is equal to (g^lambda mod nsquare - 1)/n, where lambda is a component of the private key and n and nsquare are components of the public key.
2) It then computes the decryption factor for the ciphertext c by computing c^lambda mod nsquare - 1 / n.
3) It multiplies the decryption factor with u, and takes the result modulo n, to obtain the decrypted plaintext message. Formula used in this code to decrypt a ciphertext is based on the properties of the Paillier cryptosystem, which allows for the decryption of a ciphertext without revealing the private key.

VII. IMPLEMENTATION

The use of blockchain technology, Advanced Encryption Standard (AES), and homomorphic encryption in the implementation of an electronic voting (e-voting) system delivers a secure and transparent platform for the conduct of elections. This guarantees the integrity of the voting process while simultaneously preserving the privacy of the voters. The basic foundation of this design hinges on the incorporation of blockchain technology, which operates as a persistent and decentralized record consisting of a series of blocks. Each block contains an encrypted vote set, ensuring that once a vote is registered on the blockchain, it remains unmodifiable and impervious to tampering, thereby upholding the integrity of the election outcomes. The integration of Advanced Encryption Standard (AES) encryption into the ballot provides an additional layer of security, making it difficult for unauthorized entities to tamper with or access vote data. AES, being a resilient encryption algorithm, is employed to guarantee the secrecy and accuracy of the vote. Through the application of AES encryption to the ballot, the system incorporates an additional layer of security, rendering it arduous for unauthorized entities to manipulate or obtain access to the vote data.

The utilisation of homomorphic encryption serves to preserve voter confidentiality by allowing mathematical operations to be executed on encrypted data without the need for decryption. This cryptographic technique facilitates mathematical operations to be performed on encrypted data sans the prerequisite of decryption.When applied to the context of an e-voting system, this characteristic permits the amalgamation of encrypted votes on the blockchain while upholding the secrecy of personal votes. The encrypted vote, which utilizes a combination of AES and homomorphic encryption, is subsequently transformed into a block. This strategy provides an assurance that the votes will be kept confidential and safe, since the voters' actual preferences are not exposed. The commencement of the electoral process involves the initiation of the election by an Election officer, thereby granting eligible voters access to the system's voting interface. The legitimacy of voters is verified by utilizing an authentication process that incorporates a One-Time Password (OTP) mechanism. Upon successfully accessing the voting interface, the voters are prompted to provide their credentials, which typically comprise a unique identification number.

Subsequently, the system ascertains the authenticity of these initial credentials and proceeds to generate a distinctive OTP, which is transmitted to the registered mobile number and email address linked with the voter. This verification process ensures that only individuals possessing valid credentials and access to the registered contact information are eligible to proceed with voting.

The voter receives the OTP on their mobile device and in their email inbox, subsequently inserting it into the specified field within the voting interface. The system then verifies the entered

OTP by comparing it against the OTP generated and sent earlier. The voter is authorized to vote if the OTPs match, indicating a successful authentication process.

Booth managers play a crucial role in managing the voting process. Their responsibility encompasses the addition of voters to their respective booths, as well as the ability to monitor the total votes cast within their designated areas, as depicted in Figure 7.1. This empowers them to ensure that the voting process is conducted with the utmost accuracy and legitimacy. However, due to the encryption and privacy-preserving nature of the system, booth managers are not privy to specific vote details or knowledge of which candidate garnered more votes.

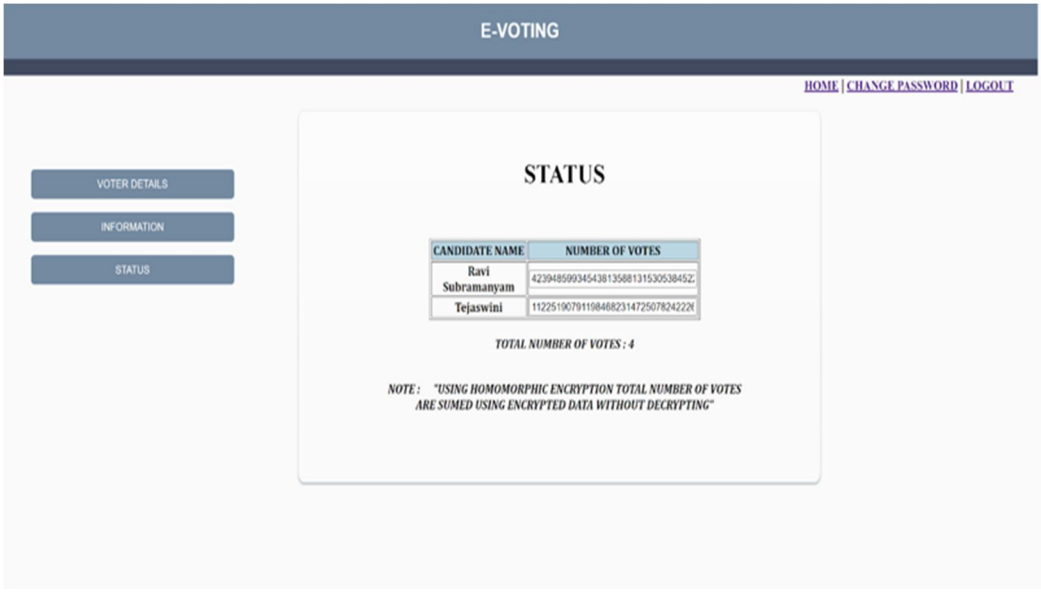


Figure 6.1 Voting status of particular booth

Upon the conclusion of the designated voting period, the election officer finalizes the election process. It is at this juncture that the encrypted votes stored within the blockchain are readily retrievable. The election officer is vested with the authority to access the blockchain and retrieve the encrypted votes for further processing, including decryption and tallying of the results, as illustrated in Figure 7.2.

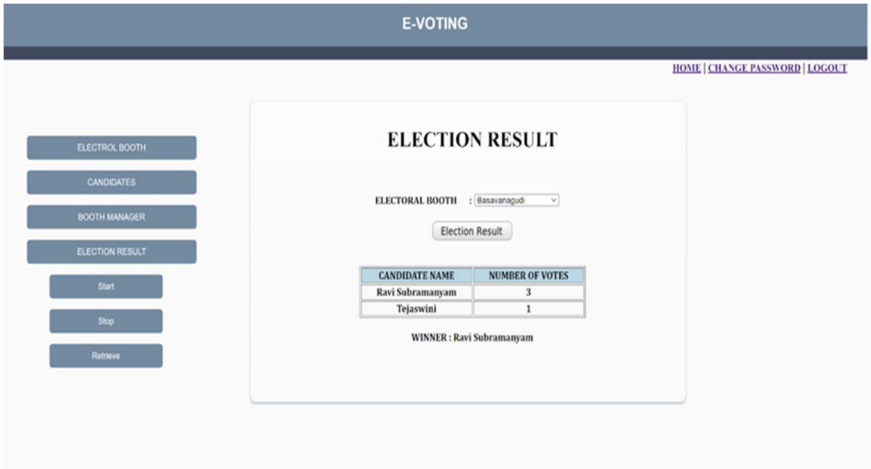


Figure 6.2 Election result

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 11 Issue VII Jul 2023- Available at www.ijraset.com

VIII. EVALUATION AND RESULT

The provided graphs serve to demonstrate a comparative analysis of the encryption and decryption times of three notable encryption methodologies, namely RSA, ElGamal, and Paillier. The abscissa of said analysis denotes the quantity of blocks, while the ordinate indicates the duration of the execution in seconds.

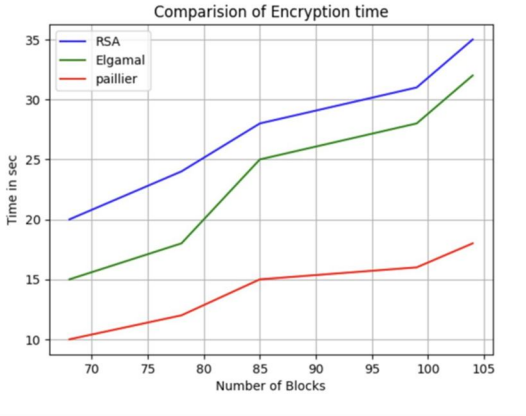


Figure 8.1 Encryption time

After analysing the graph that depicts the encryption time in Figure 8.1, it is evident that the Paillier encryption algorithm consistently outperforms the other two methods concerning encryption time. As the number of blocks increases, the Paillier encryption method exhibits exceptional efficiency, with encryption time remaining relatively stable and minimal. This efficiency is notably advantageous in scenarios where numerous blocks need to be encrypted quickly and securely.

On the other hand, the ElGamal encryption algorithm displays marginally higher encryption times than the Paillier method, but still outperforms RSA. As the number of blocks increases, the encryption time for ElGamal gradually increases, indicating a linear relationship. Although not as efficient as the Paillier method, ElGamal encryption remains a viable option for scenarios involving a moderate number of blocks.

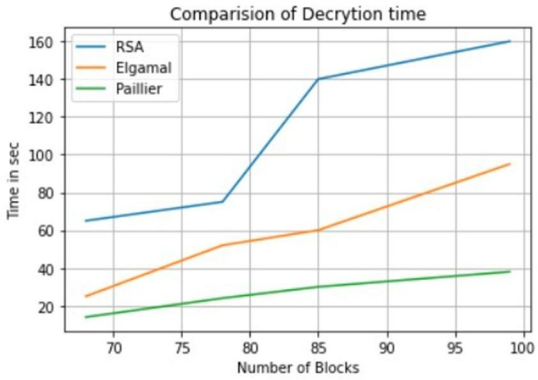


Figure 8.2 Decryption time

In relation to the decryption time, as presented in Figure 8.2, the Paillier encryption algorithm once again emerges as the most efficient. The decryption time for Paillier remains consistently low for all numbers of blocks, thus highlighting its suitability for prompt and secure decryption processes. ElGamal demonstrates similar performance, with decryption time slightly higher than Paillier, yet significantly lower than RSA. In contrast, RSA encryption exhibits the highest encryption and decryption times among the three algorithms, particularly for a larger number of blocks. As the number of blocks increases, RSA encryption and decryption times experience a significant increase, rendering it less favourable for applications that require fast processing or real-time operations.

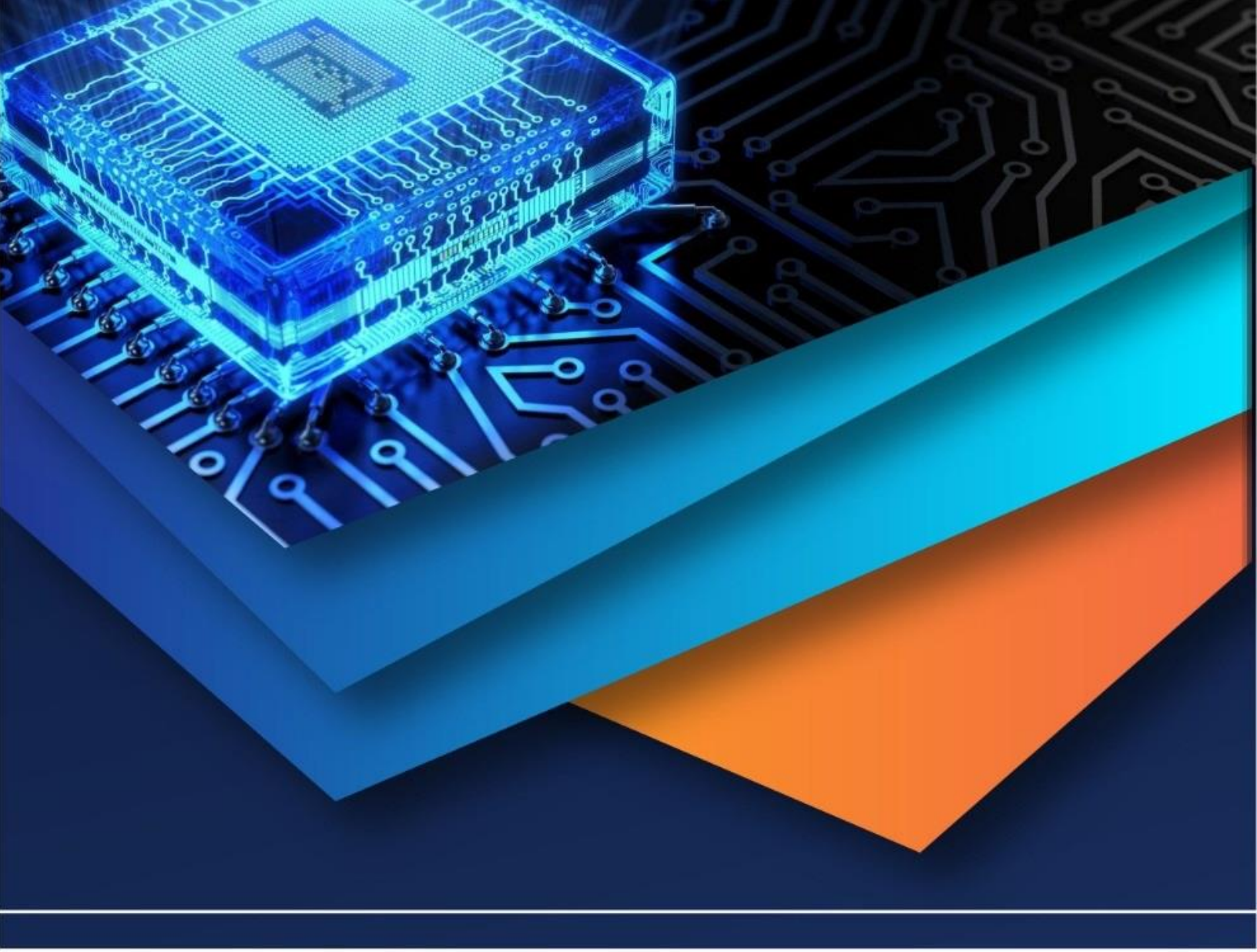
The results of the graph indicate that the Paillier encryption algorithm is the most suitable option for achieving efficient encryption and decryption processes.

IX. CONCLUSION

This advanced approach instils trust and confidence in the electoral process, mitigating concerns regarding tampering and violations of privacy. Through the amalgamation of these technologies, a sturdy and impermeable environment is established, which guarantees the integrity, authenticity, and confidentiality of electoral outcomes. This advanced approach instils trust and confidence in the electoral process, thereby mitigating concerns regarding tampering and violations of privacy. The combination of blockchain technology and homomorphic encryption allows for a secure and private analysis of results while simultaneously maintaining voter privacy and system transparency. Statistical calculations can be performed with precision without revealing individual voting preferences. The combination of AES, homomorphic encryption, and blockchain technology in e-voting systems establishes a robust and impenetrable environment, which guarantees the confidentiality, authenticity, and integrity of electoral outcomes, thereby instilling trust and confidence in the electoral process. Despite the barriers, the ongoing creation and enhancement of scalable and secure implementations will have a meaningful impact on the widespread adoption of blockchain-based e-voting systems, thereby fostering trust, integrity, and impartiality in democratic processes.

REFERENCES

[1] “Blockchain Technology and Cryptocurrencies” by Siddharth Rajput, Archana Singh, Smiti Khurana, Tushar Bansal, Sanyukta Shreshtha IEEE 2019
[2] “Application of Blockchain for the Security of Decentralized Cloud Computing” by Nazmun Nahar, Farah Hasin and Kazi Abu Taher IEEE 2021
[3] “A Homomorphic Encryption Approach in a Voting System in a Distributed Architecture” by Segundo Moisés Toapanta Toapanta , Luis José Chávez Chalén , Javier Gonzalo Ortiz Rojas , Luis Enrique Mafla Gallegos 2020 IEEE
[4] “Secure Remote E-Voting using Blockchain” by Divya Rathore, Virender Ranga IEEE 2021
[5] “A Blockchain-Based Authentication Method with One-Time Password” by Mingli Zhang, Liming Wang, Jing Yang IEEE 2019



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY