International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Hybrid Secure Algorithms and Optimal Blockchain to Ensure E-Voting Data Immutability at Cloud

Rakshitha C. M.*[1], Nirmala Hiremani[2], Nataraj K. R.[3]

Submitted: 27/04/2023 **Revised**: 28/06/2023 **Accepted**: 06/07/2023

Abstract: Electronic voting (E-voting) sometimes known as "online voting" is a quick, low-cost and secure way of voting that meets real-time data requirements. Nonetheless, with electronic systems, voter security and privacy remain significant impediments. This work focuses on presenting Blockchain-based electronic voting systems since Blockchain technology ensures data immutability and transparency in the voting process. A significant amount of voted data was saved on public cloud storage in this case. However, in this paper, a hybrid secure algorithms-based optimal Blockchain is presented to increase Blockchain performance in terms of security and cloud storage cost. First, the voting data is secured using the Advanced Encryption Standard (AES). The AES secret key is then encrypted using Extended Elliptic Curve Cryptography (EECC). Second, the encrypted data will be stored in the most appropriate blockchain blocks. For optimal block selection, the Extensible Firefly Algorithm (EFFA) is described. Finally, these blocks will be hashed using SHA-256 before being saved on the cloud server. The proposed electronic voting system achieved a high level of security while requiring less storage space, according to simulation results.

Keywords: *E-Voting, Hybrid Secure Algorithms, Immutability, Optimal Blockchain.*

1. Introduction

Cloud computing technology has advanced rapidly, and cloud services can now be used for a wide range of purposes. Cloud service providers (CSPs) typically make data processing and storage systems available by using resources that have a cost per use. The costs of establishing and maintaining systems that suit a data stakeholder's internal data demands are reduced with the help of these services. Users must therefore trust that service providers will keep their data and associated information secure while storing it. Safety and privacy are two of the most critical considerations while utilising and promoting cloud computing.

Users commonly encrypt data before saving it since it may be accessed by unauthorised persons. Several research-based solutions use authentication mechanisms and encryption algorithms to prevent unauthorised access to data, but they frequently fail to address the issue of tracking the valid changes made to the data. Data is extremely valuable in the digital age. The key responsibility is to ensure that data is unchangeable and safe from external threats. Instead of having one central body in total control, decentralisation is a good approach for distributing power and authority to the organization's limits.

Blockchain is the finest solution for decentralised data storage and maintenance right now.

Because of the openness given by blockchain technology, any nefarious or covert action on the network is impossible. Blockchain-based cloud storage generates transaction records to verify ownership and identity due to its sequential storage function, which constantly verifies transactions [1]-[3]. As a result, it produces a strong, integrated, and well-organized chain of building components.

We used electronic voting as an example of an application where maintaining and storing vote data requires a high level of security in our study. Voting is an important obligation that permits citizens to take part in democratic elections in any country. While the voting process has substantially improved in terms of security, there are still difficulties that must be addressed before it reaches the requisite degree of maturity [4]-[6].

The main problems with current voting systems include fraud, administrative errors, and most critically, a lack of transparency [7][8]. Blockchain-based voting systems, which provide end-to-end security and transparency, can address these issues [9][10]. The usage of public key cryptography, which is a feature included as standard in a blockchain to provide security, ensures the legitimacy and integrity of voting. Furthermore, the blockchain's immutability ensures that votes cast just once are final

[1] *Department of Computer Science & Engineering, Visvesvaraya Technological University, Belagavi, India*
ORCID ID : 0000-0002-8483-4103
[2] *Department of Computer Science & Engineering, Visvesvaraya Technological University, Belagavi, India*
ORCID ID : 0000-0002-9108-1445
[3] *Department of Research & Development, Visvesvaraya Technological University, Belagavi, India*
ORCID ID : 0000-0003-1669-4513
Corresponding Author Email: pr80341@gmail.com

(votes cast once cannot be cast again). With the emergence of blockchain technology, a new era of study into the development of dependable, decentralised electronic voting systems has begun.

A blockchain is a data structure created by connecting blocks in a specific way. To prevent manipulation and to store transaction information, each block has its unique hash value. The disseminated record is at the heart of it all. Each node on the blockchain maintains a backup log, and transaction information is accessible and easy to retrieve. The fact that blockchain is decentralised, transparent, and immutable (cannot be modified) may aid in the resolution of issues with computer voting systems [11]-[15].

1.1 Problem statement and contributions

Data owners can exchange and store encrypted data on cloud servers, where it can later be accessed by a large number of users. Data owners have less control over their data now that it is kept on the cloud and other systems are designed to handle it. Several research-based systems use encryption to stop unauthorized people from getting to the data, but they often forget how hard it is to keep track of changes that were made to the data legally. Blockchain is the most secure way we have right now to store private data on public cloud servers [16][17]. With this feature, the voted data can be kept safe because it can't be changed(immutable). This is due to the fact that in a cryptographic hash chain, the newly added block is connected to the previously added blocks [18]-[22].

A further difficulty is the rise in the number of Blockchain transactions, which raises the need for more storage. So, another goal of this work is to lower cloud storage costs. The contributions listed below are included in our suggested work in order to fulfil these objectives.

- In this method, hybrid secure algorithms are given alongside blockchain to assure data immutability. AES is used to encrypt the voting data. The EECC technique, which also generates difficult-to-decipher shared keys, protects the Advanced Encryption Standard's symmetric key. In EECC, a secret key is generated in addition to the pair keys.
- The EFFA method is used to find the optimal number of blocks to optimize cloud storage costs. By incorporating the oppositional based learning (OBL) technique, the performance of the FFA algorithm improves.
- The encrypted voting data and AES secret key are saved to blocks on each blockchain node, where the blocks are hashed with the SHA-256 technique. These hashed blocks are saved in the cloud server.

The subsequent sections are organized as follows. Section 2 examines recent works that have focused on e-voting systems and other blockchain-based applications. Hybrid Secure Algorithms based Optimal Blockchain for E-Voting Systems is proposed in Section 3. Section 4 analyses the results.

2. Related Work

In this section, we look at recent research on E-voting systems. Because blockchain technology is transparent, the information about the candidates will be made public. Also, the feature of "pseudo-anonymity" makes members' private information public. Also, there is a chance that info could be changed by third parties. So, to solve these problems, Hai Jin et al. [23] built a way to control authority into blockchain. Also, they had made a voting system built on AMVchain, a blockchain that works well and is fully decentralized. The AMVchain is made up of a three-layer access control system. On each layer, smart contracts were used to validate and give rights. Linkable ring signature was added to the vote process to protect the privacy of the ballots. The article's results showed that the suggested system met the basic needs even when there were a lot of people using it at the same time. Sridhar Vemula, Ram Mohan Rao Kovvur, and Dyna Marneni [24] introduced CryptDB to make e-voting more reliable and secure. In the proposed electronic voting system, information about candidates, voters, and votes was encrypted and kept in a database called CryptDB. At any stage of the polling system, the suggested systems did not give any information to the thieves. So, the result of the voting process was reached by using an online framework that made data unchangeable, safe, and private. The suggested system would have kept data like vote results safe from a bad administrator. By proposing this method, the authors make sure that it is as unchangeable as possible.

P. Pandiaraja et al. [25] had suggested a protocol-based secure online e-voting system to solve security problems in e-voting systems. Their E-Voting Cloud System was made up of three steps: registering, voting, and announcing the results. Cloud software was used to check the data from the votes. The layout of cube data storage and the user-differentiated framework were the first things to be put into the protocol. Depending on the framework, the proposed protocol allows both the encrypted framework and the E-Voting Cloud System for collecting voter data. The e-voting system took less time because of the plan that was presented. Either the privacy of the voters or the way the votes are counted is very bad with the old voting methods. E-voting systems have been made as a big step toward solving these problems.

Adeel Anjum et al [26] had presented Internet e-voting system. With the dual signature, this procedure met the most important requirements, such as privacy, anonymity,

eligibility, and the ability to check. With the z3 and HLPN solver, the writers checked that the dual signature protocol worked. Because of the suggested plan, the authors were able to cut down on the time it took to run code and count votes.

Haibo Yi had suggested that blockchain technology could make e-voting safer. The writers made a synchronized structure for voting records based on distributed ledger technology. Because of this set-up, fake votes can't be made. Then, they made a structure for user credentials that was based on ECC and provided authentication and could not be disputed. In the end, they came up with a withdrawal structure that let people change their vote before a certain limit.Using the above structures as guides, the writers created blockchain technology for electronic voting. Due to the structures that were suggested, the problem of forged votes had been solved during e-voting.

Blockchain-enabled voting system, or BEV for short, is important in e-voting systems because it can't be changed. Shufan Zhang, Lili Wang, and Hu Xiong [27] came up with nine important goals that a BEV system can and must meet, such as robustness, scalability, and verifiability. But the authors knew that the traditional BEV system couldn't do any of the nine things that they wanted. So, the writers came up with a new BEV framework that they called Chaintegrity. All of the specific needs were met by this suggested framework. Also, they had suggested that counting Bloom filter and Merkle hash tree be used together to verify quickly. They also used the code-voting way to make the system more stable. By coming up with these plans, the authors were able to get low transmission overhead.

The proposed work incorporates hybrid cryptographic algorithms and an optimal block selection mechanism in a cloud server to protect the voting data in an immutable and cost effective manner. This is done to ensure high security and address performance issues with blockchain by studying all of the research work that has been done previously.

3. Proposed Methodology

To implement the electronic voting system model, the proposed proof of work consists of hybrid cryptographic algorithms, cost efficient algorithm and Hash code Generation. To ensure immutability of the voted data, different cryptographic algorithms are used, once the vote has been casted from voter.

Figure 1 depicts the overall process that would be followed by the proposed electronic voting system. Before they are allowed to vote, voters have their information checked to make sure it is accurate and that they meet the qualifications (see the figure for more information). After their eligibility to vote has been confirmed, voters are given the opportunity to cast their ballots in the election. A cloud server will be used to store the voting information for each individual voter. In order to protect the information from being altered by third parties or other entities, the data will be encrypted using the hybrid cryptography that was suggested. The Enhanced Elliptic Curve Cryptography (EECC) and the Advanced Encryption Standard are both components of this particular technique. The information about the voters is encrypted using Advanced Encryption Standard, and the encrypted data along with an AES key that has been encrypted using the EECC approach is stored on a cloud server. After then, the SHA-256 algorithm will be used to hash the transaction of encrypted voter data coming from a number of different polling stations. These newly generated hash codes are saved in blocks that have been chosen by the EFFA algorithm in order to reduce the amount of revenue spent on storage. Election officers are able to declare the results of the count because they are aware of the digital signatures of all of the voters and they have access to the encrypted data that is stored on the server.

Voter’s Registration:In this step, each candidate is signed up to vote after their status has been checked. At first, the people in charge of voting get basic information from each member, like their name, address, age, gender, place of birth, etc. The people in charge of voting check each candidate's eligibility based on the papers they have given them. If the candidate meets the requirements, he or she will be registered as a voter. This check is done based on the information they gave when they signed up.The full results of this review and the next steps for the most important finds are outside the scope of this paper. Protecting the data of voters and their ballots cast throughout the voting process is our key objective. This is done with the intention of preventing harmful attackers from gaining knowledge regarding candidate votes.

3.1 Hybrid Secure Algorithms

Voting is permitted after voter registration. Voter data is safely saved on the cloud server during the voting process. An optimal blockchain based on hybrid cryptography is provided for secure data transactions. The safe exchange of vote data is discussed in the section that follows.

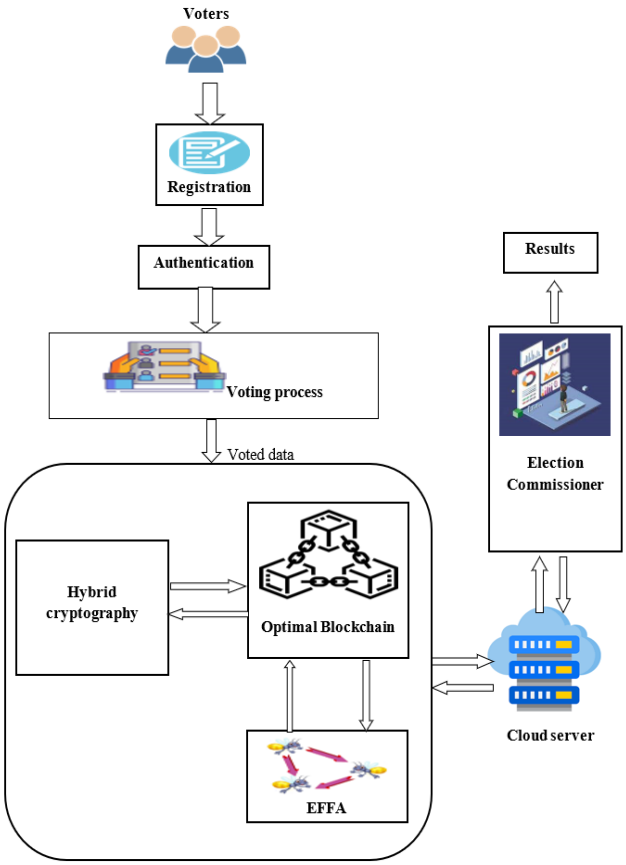

Fig.1. Overall workflow of the proposed e-voting system

The votes from various booths are encrypted using hybrid cryptography before being stored on a cloud server. This method combines the EECC and Advanced Encryption Standard algorithms. The voted data is encrypted using the Advanced Encryption Standard algorithm, and the secret key for the Advanced Encryption Standard is encrypted using the EECC algorithm. Figure 2 shows the encryption of hybrid cryptography.

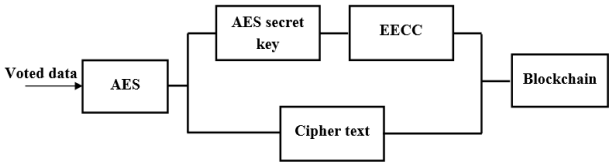

Fig.2. Encryption of voted data

Advanced Encryption Standard: First, the Advanced Encryption Standard algorithm receives the voted data as input. Using a random number generator, this approach creates round keys from the initial encryption key (the Advanced Encryption Standard secret key) (RNG). Although this technique works with a variety of key sizes, the key size used in this work is 128 bits, with 10 rounds. In addition, the block size matches the key length. After 10 rounds, the inputted vote data is encrypted.

Enhanced Elliptic Curve Cryptography(EECC) : Asymmetric or public key cryptography uses the ECC algorithm. For the purpose of data encryption and decryption, this technique generates a pair of keys, such as a public key and a private key. Although the ECC technique can raise security levels with less computer resources, it also increases the likelihood of implementation errors. Hence, the Improved ECC algorithm is introduced to raise the system's security level. Namely, a secret key is generated together with the pair of keys.

The maximum limit for the EECC algorithm is a curve with particular base points that uses a prime number function. The following is a definition of the mathematical function of ECC:

$$l^2 = m^3 + um + v \quad (1)$$

Here, the random integers are represented by u and v.

Key generation: Public (R_k), private (A_k), and secret keys(S_k) are the three types of keys that are generated in EECC. Private key (Ak) is the randomly choosen point on the elliptic curve whose value must be smaller than the large value of n on the curve.The input data is initially encrypted and created by the server which is R_k. Afterwards Akhas been used on the server side to decode the relevant data. Thirdly Sk will be generated, based on R_k , A_k and the curve's point (G). During the execution of EECC, the S_k is added to the data during encryption and subtracted during decryption from the cipher data. Moreover, A_k is picked at random from the n prime numbers. Following that, R_k is created as follows depending on Ak and G.

$$R_k = A_k * G \quad (2)$$

Afterwards, S_k will be generated by combining R_k, A_k and G.

$$S_k = R_k + A_k + G \quad (3)$$

Encryption: The Advanced Encryption Standard secret key ($AESs_k$) is modified into an affine point on the curve during this stage. Moreover, the acquired $AESs_k$ is encrypted. Two cypher messages that are defined as follows are present in the encrypted data:

$$C1 = S_k + ((K*G) \quad (4)$$

$$C2 = S_k + (AESS_k + (k*R_k)) \quad (5)$$

In this case, C1 and C2 are cipher messages, while K stands for a random number between [1, n-1].

Algorithm 1: Hybrid cryptography's encryption phase

Input: Electronicallyvoted data and $AESs_k$

Output: Encryption of Electronically voted data and $AESs_k$ in Blockchain.

Advanced Encryption Standard with RNG

1. Use RNG to create round keys from the secret Advanced Encryption Standard secret key($AESs_k$).

2. Carry out the subsequent operation on the results of each round's voting.

i. SubBytes ii. Shift Rows iii. Mix Column iv. Round Key

3. Disregard the Mix Column operation from the previous cycle.

4. After all rounds have been completed, obtain encrypted voting data.

EECC

5. On the elliptic curve, select G.

6. Choose A_k at random from the n prime numbers.

7. Build R_k by utilising (2).

8. Build S_k by utilising (3).

9. Use (4) and (5) to encrypt Advanced Encryption Standard Secret key($AESs_k$).

10. Save the encrypted information and Advanced Encryption Standard Secret key($AESs_k$) to the blocks of the blockchain.

3.2 Optimal Blockchain

Figure 3 provides a high-level overview of the ideal or optimal blockchain, which includes the Firefly algorithm and opposition-based learning methods [24].

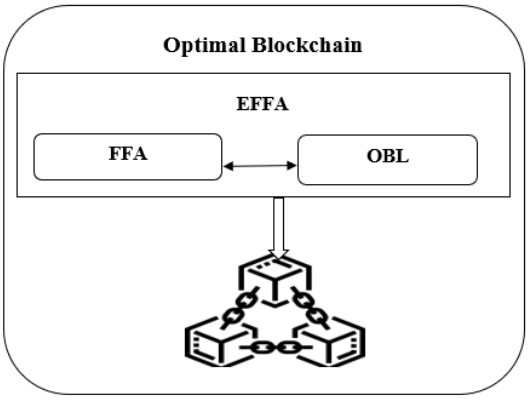

Fig. 3. Blockchain with firefly and opposition based learning

Data transactions are kept in chronological blocks that make up the blockchain. Each block contains a hash code, a collection of transactions, a time stamp, and proof of work (PoW). Also, each block is linked to the one before it via an immutable hash code. The first section is called "Genesis." The chain's maximum nodes concur on the data input. After all parties have agreed to the data, it is impossible to change or delete. These are some logical structures that the blockchain technology maintains:

$$Genesis = No\ prior Hash \quad (6)$$

$$HashFund(prior2) = PoW1 \quad (7)$$

$$HashFund(prior3) = PoW\,2 \quad (8)$$

$$PoW = HashFund(Timestamp, Value-Found, Transactionsset, Hash-\text{Previous}) \quad (9)$$

HashFunc(encryption of electronically voted data and AESsk)= Cryptographic Hash Function(10)

Moreover, the blocks are put into the cloud server. These produced hash codes can be kept in the best-chosen blocks to save on storage space. The EFFA algorithm is described for choosing the best blocks.

Extensible Firefly Algorithm(EFFA):FFA is a meta-heuristic algorithm with biological inspiration. It takes its cue from the flashing behaviour of fireflies. This algorithm's three main rules are universal, meaning that any firefly with a stronger glow will draw in more fireflies. Second, the defined fitness function is used to estimate the firefly's brightness. Finally, attractiveness and brightness are inversely correlated. When the distance grows, the appeal reduces. A firefly will move in the direction of the brighter one. Also, if there isn't a brighter one, it moves randomly. Opposition Based Learning(OBL) method is used to improve FFA performance. Opposition Based Learning(OBL) speeds up processing and improves searching capabilities. The data offloading is described in the following EFFA phases.

Initialization: Each firefly's position or the solution is set to a starting point during this phase. The solution in this work is taken into consideration to be the number of blocks in each node that will be kept on the cloud server. Particularly, $1 \le Q_m \le N$ here Qm is the number of optimum blocks, and N is the total number of blocks in a node. The following definition applies to the population.

$$F = \{f_1, f_2, \ldots\ldots, f_n\} \quad (11)$$

Here, fn indicates the location of the firefly or N blocks, i.e.

$$fn = \{b1, b2, \ldots\ldots, bn\} \quad (12)$$

Here, bn stands for a node's Nth block.

Opposition Based Learning (OBL): Then, the opposite solutions $\bar{y}$ are produced for each initialised solution [24]. The opposite approach is best described as follows:

$$\bar{y} = a + b - y \quad (13)$$

Where, y ∈ [a,b] is a real number

Fitness function: Storage costs are used to determine fitness for each solution. These are defined as:

$$Ft(i) = Mn(\text{cost}) \quad (14)$$

Cost in this context can be defined as,

$$B\text{cost} = rS * \sum_{i=1}^{m} Q_m \quad (15)$$

Here, r stands for the proportion of cloud storage to local storage, and S stands for the size of a block for each node. $\sum_{i=1}^{m} Q_m$ stands for the total size of all optimal blocks. The ideal solution or optimal blocks are those that correspond to the minimum fitness. Otherwise, the answer is revised.

Update the solution: Light's brightness is inversely related to d2. The d in this case stands for the distance from the source. As light travels through a medium with the coefficient of light absorption (λ), the brightness of light L fluctuates based on d. It can be explained as follows:

$$L(d) = L_0 e^{-\lambda d^2} \quad (16)$$

L0 indicates the brightness at the source's position in this context.

For simple computing, the definition of (16) is as follows:

$$L(d) = \frac{L_0}{1+\lambda d^2} \quad (17)$$

Moreover, firefly attractiveness is calculated as follows:

$$M(d) = \frac{M_0}{1+\lambda d^{2'}} \quad (18)$$

M0 indicates attractiveness at d=0 in this case.

The firefly at fj moves in the direction of the position fi , if the brightness of the firefly at the position fi is larger than that of the firefly at the position fj. The following is a definition of the location update:

$$f_j(t+1) = f_j(t) + M_0 e^{-\lambda d^2}\left(f_i(t) - f_j(t)\right) + \mu\nu \quad (19)$$

μ stands for the randomization parameter [$0 \le \beta \le 1$] and ν stands for random number in this context.

Termination: The method will continue until the best blocks in each node have been found. The algorithm will end once the optimal solution has been found.

SHA-256 will be used by the blockchain to hash the optimal blocks. Following that, the cloud server stores the hash codes along with their distinctive digital signature. By comparing the digital signature on the server, the counting authority will authenticate the data. The authentication, decryption of Advanced Encryption Standard(AESsk) and encrypted voting data is covered in the phase that follows.

Algorithm 2: EFFA algorithm-based optimal blockchain

Input: N blocks per node, $\lambda = 2$

Output: optimized blocks (Q_m)

1. Place N blocks or fireflies in their initial positions.

2. Determine the opposing outcome for each solution utilising (13).

3. Calculate fitness using (14).

4. If L(d) of firefly at fi> L(d) of firefly at fj

Revise the answer in accordance with (19).

5. End if

6. If the result Qm is attained, stop the algorithm.

3.3. Decryption phase

The counting officials rehash the blocks from the server because they are aware of each voter's unique digital signature. They extract the AES-256-cipher text and encrypted blocks from the blocks (voted data). The decryption phase of hybrid cryptography is shown in Figure 4. It is necessary to first decrypt Advanced Encryption Standard(AESsk) before decrypting the cypher text. Using the same EECC technique, the AESsk is decoded. As explained below, the EECC decryption function includes:

AESsk= (((C2-Ak)*C1)-Sk) (20)

According to (20), Sk is taken out of the cipher texts to get the original AESsk. The encrypted voted data is decrypted using the acquired AESsk by carrying out the operations in the order of Add round key, Inverse mix columns, Shift Rows, and Inverse Subbyte. The counting authority obtain the original voter data once all rounds have been completed. They begin counting the votes and announcing the results based on the votes received.

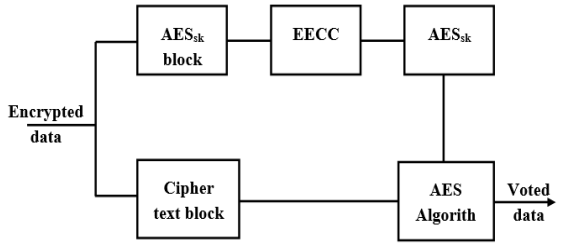

Fig. 4. Decryption phase of voted data

4. Results and Discussion

This framework was built on a computer with Windows 10 and an Intel Core i5 processor and 6GB of memory. Python is used to simulate the method that was mentioned. This simulation uses the Congressional Voting Records Data Set, which was found in the UCI machine learning library (https://archive.ics.uci.edu/ml/datasets/congressional+voting+records link).

The proposed optimal blockchain's performance is evaluated based on its encryption time, decryption time, memory usage during encryption and decryption, and security level. Comparisons are made between the performance of the proposed method and that of the ECC-AES and ECC-DES algorithms. Every cryptographic algorithm is implemented with the optimal blockchain.Figure 5 uses the optimal blockchain to show the encryption time of several hybrid cryptographic techniques. As seen in the figure, ECC-Advanced Encryption Standard has a 26% faster encryption time than ECC-Data Encryption Standard. The proposed EECC-AES encryption time, however, is reduced to 32% and 50% respectively, when compared to ECC-AES and ECC-DES. Figure 6 shows the decryption times of various hybrid cryptographic methods with the optimal blockchain. According to the figure, EECC-AES's decryption time is reduced by 34% and 48% compared to ECC-Advanced Encryption Standard and ECC- Data Encryption Standard, respectively. Compared to other hybrid cryptographic algorithms, EFFA with Improved ECC uses less memory since it selects blockchain blocks in an optimal manner. EECC-AES uses 21% and 14% less memory for encryption than ECC-AES and ECC-DES, respectively as shown in Figure 7 . Figure 8 shows the memory usage for several decryption methods. Memory utilisation for decryption in EECC-AES is reduced to 9% and 20%, respectively, compared to ECC-Advanced Encryption Standard and ECC- Data Encryption Standard.

The block selection times of several algorithms are shown in Figure 9. The block selection times of the existing cuckoo search optimisation (CSO), lion optimisation algorithm (LOA), gravitational search optimisation (GSO), and conventional FFA are compared with those of the Proposed Adaptive or extensible firefly algorithm (EFFA) . Block selection time increases as the number of iterations rises, as shown in the figure. In other words, compared to FFA, GSO, LOA, and CSO, the block selection times for EECC-AES are 83%, 92%, 94%, and 96% faster at iteration 5. Moreover, at iteration 25, the block selection time of the EECC-AES was lower than that of the FFA, GSO, LOA, and CSO by 33%, 42%, 51%, and 56%, respectively. Figure 10 shows the security level of different techniques. The security level of EECC-AES is improved to 2% and 4% than that of ECC-Advanced Encryption Standard and ECC- Data Encryption Standard correspondingly as the security strength of ECC is boosted by adding a secret key along with the pair keys. Figure 11 depicts the cost of storage for various methods. The blocks in blockchain are picked using optimisation techniques to lower the cost of cloud storage. In this method, the storage cost is decreased to 34%, 41%, 50%, and 57% less than that of FFA, GSO, LOA, and CSO, respectively, by introducing the extensible or adaptive firefly (EFFA) algorithm.

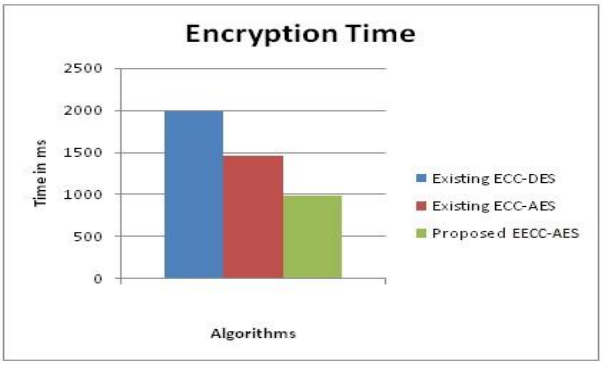

Fig. 5. Study of encryption time

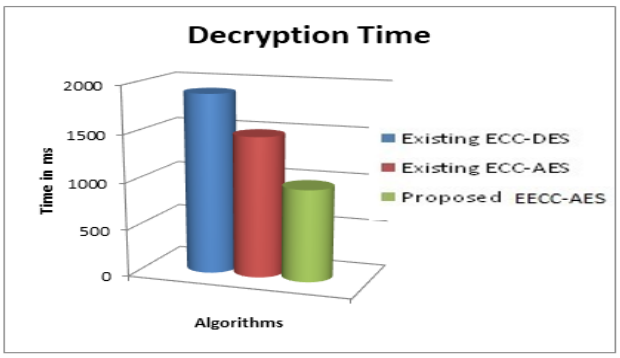

Fig. 6. Study of decryption time

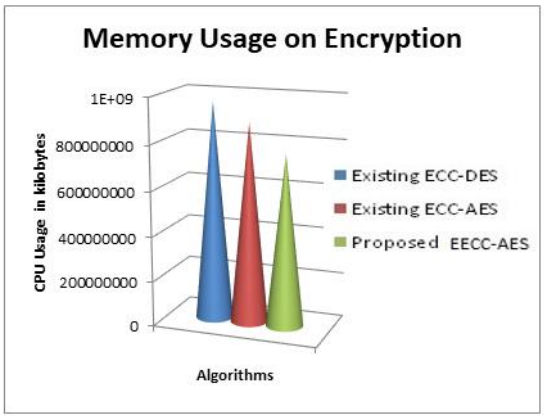

Fig. 7. Study of memory usage on encryption

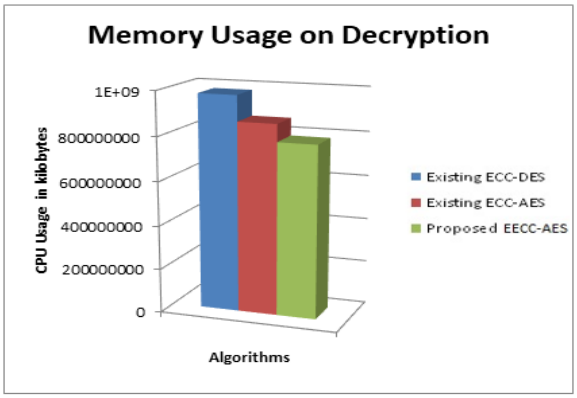

Fig. 8. Study of memory usage on decryption

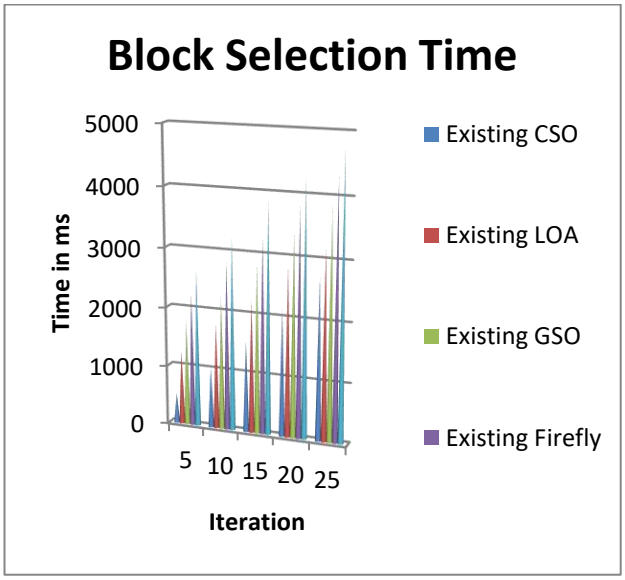

Fig. 9. Study of block selection time

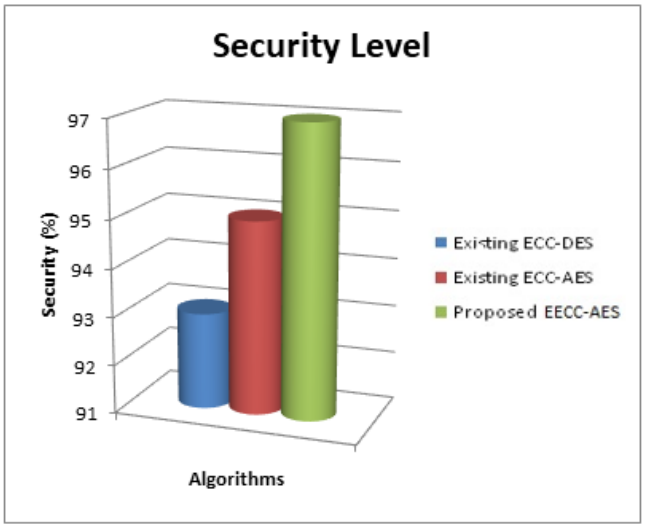

Fig. 10. Study of security level

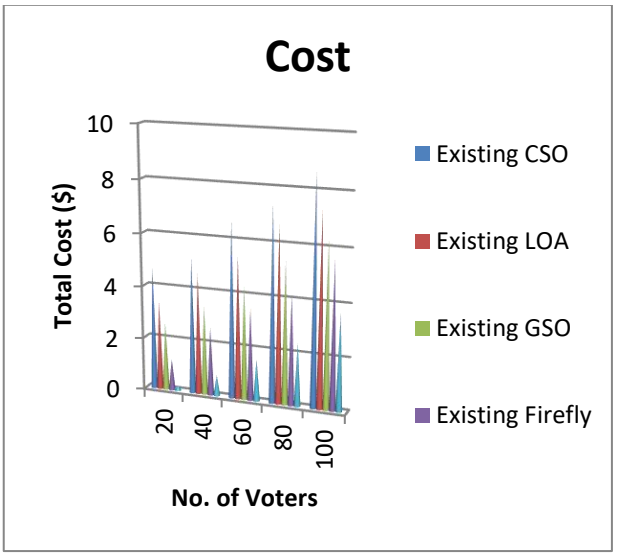

Fig. 11. Study of storage cost

5. Conclusion

This work presents Hybrid Secure Algorithms Based On Blockchain to assure the data immutability of voters in Electronic Voting System. The votes of each voter have been encrypted using the Advanced Encryption Standard algorithm to improve privacy preservation. The secret key for the Advanced Encryption Standard has also been encrypted with the EECC technique to increase security. The Advanced Encryption Standard secret key and encrypted data were in the optimal blocks that the EFFA algorithm had chosen. These blocks were then hashed using SHA-256 and stored on the cloud server. Finally, the voting data has been acquired or decrypted by the counting authority. The votes have been tallied, and the results are now known. According to simulation results, the proposed voting method achieved storage cost of 57%, block selection time of 56%, and security level of 4%.

Acknowledgements

We would like to convey our heartfelt gratitude to the All India Council for Technical Education (AICTE) for providing a Doctoral Fellowship to support our research.

Author contributions

Rakshitha C M: Conceptualization, Methodology, Software, Writing-Original draft.

Nirmala Hiremani: Related work, preparation, Software, Validation.

Nataraj K R: Visualization, Investigation, Writing-Reviewing and Editing.

Conflicts of interest

The authors declare no conflicts of interest.

References

[1] Xu, Mengtian & Feng, Guorui & Ren, Yanli & Zhang, Xinpeng. (2020). “On Cloud Storage Optimization of Blockchain With a Clustering-Based Genetic Algorithm”. IEEE Internet of Things Journal. PP. 1-1. 10.1109/JIOT.2020.2993030

[2] Lakhan, Abdullah & Mohammed, Mazin & Ibrahim, Dheyaa & Abdulkareem, Karrar. (2021). “Bio-Inspired Robotics Enabled Schemes in Blockchain-Fog-Cloud Assisted IoMT Environment”. Journal of King Saud University - Computer and Information Sciences.DOI: https://doi.org/10.1016/j.jksuci.2021.11.009.

[3] Swarna Priya R.M., Sweta Bhattacharya, Praveen Kumar Reddy Maddikunta, Siva Rama Krishnan Somayaji, Kuruva Lakshmanna, Rajesh Kaluri, Aseel Hussien, Thippa Reddy Gadekallu, “Load balancing of energy cloud using wind driven and firefly algorithms in internet of everything”. Journal of Parallel and Distributed Computing, Volume 142, 2020, Pages 16-26, ISSN 0743-7315 DOI:https://doi.org/10.1016/j.jpdc.2020.02.010

[4] Jagtap, A. M., Vishakha Kesarkar, and Anagha Supekar. "Electronic voting system using biometrics, raspberry pi and TFT module." In 2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI), pp. 977-982. IEEE, 2019. DOI: 10.1109/ICOEI.2019.8862671

[5] Govindaraj, Ramya, and P. Kumaresan. "Online voting system using cloud." In 2020 International Conference on Emerging Trends in Information Technology and Engineering (ic-ETITE), pp. 1-4. IEEE, 2020. DOI: 10.1109/ic-ETITE47903.2020.245

[6] Prabhu, S. Ganesh, A. Nizarahammed, S. Prabu, S. Raghul, R. R. Thirrunavukkarasu, and P. Jayarajan. "Smart Online Voting System." In 2021 7th International Conference on Advanced Computing and Communication Systems (ICACCS), vol. 1, pp. 632-634. IEEE, 2021. DOI: 10.1109/ICACCS51430.2021.9441818

[7] S. Drakshayani, U. Vijayalakshmi, S. R. Sri, A. Srivani and A. Vyshnavi, "Online Voting System Using Blockchain," 2022 International Conference on Electronics and Renewable Systems (ICEARS),Tuticorin, India, pp. 886-891, 2022, DOI: 10.1109/ICEARS53579.2022.9752044.

[8] A. Mendon, B. Manoj Votavat and A. Lodh, "Blockchain based e-voting system," 2022 Third International Conference on Intelligent Computing Instrumentation and Control Technologies(ICICICT), Kannur, India, pp. 1785-1792, 2022, DOI: 10.1109/ICICICT54557.2022.9918010.

[9] M. N. Birje, R. H. Goudar, C. M. Rakshitha and M. T. Tapale, "A Review on Layered Architecture and Application domains of Blockchain Technology," 2022 International Conference on Electrical, Computer and Energy Technologies (ICECET), Prague, Czech Republic, 2022, pp. 1-5, DOI:10.1109/ICECET55527.2022.9872729.

[10] Imran Bashir (2018) Blockchain 101 Mastering Blockchain-Distributed ledger technology, Decentralization and Smart contracts explained-2018 2nd Edition Expert Insight

[11] G. Rathee, R. Iqbal, O. Waqar and A. Bashir, "On the Design and Implementation of a Blockchain Enabled Electronic voting Application within IoT-Oriented Smart Cities", IEEE Access, vol. 9, pp. 34165-34176, 2021. DOI: 10.1109/ACCESS.2021.3061411

[12] H. Li, Y. Li, Y. Yu, B. Wang and K. Chen, "A Blockchain-Based Traceable Self-Tallying Electronic voting Protocol in AI Era", IEEE Transactions on Network Science and Engineering, vol. 8, no. 2, pp. 1019-1032, 2021. DOI:http://dx.doi.org/10.1109/TNSE.2020.3011928

[13] S. Panja and B. Roy, "A secure end-to-end verifiable Electronic voting system using blockchain and cloud server", Journal of Information Security and Applications, vol. 59, p. 102815, 2021. DOI: 10.1016/j.jisa.2021.102815

[14] K. Khan, J. Arshad and M. Khan, "Investigating performance constraints for blockchain based secure Electronic voting system", Future Generation Computer Systems, vol. 105, pp. 13-26, 2020. DOI: https://doi.org/10.1016/j.future.2019.11.005

[15] M. Pawlak and A. Poniszewska-Marańda, "Trends in blockchain-based electronic voting systems", Information Processing & Management, vol. 58, no. 4, p. 102595, 2021. DOI: 10.3390/s22197585

[16] Haiyang Yu , Qi Hu, Zhen Yang , and Huan Liu , (2021) “Efficient Continuous Big Data Integrity Checking for Decentralized Storage” IEEE TRANSACTIONS ON NETWORK SCIENCE AND ENGINEERING 2021 Vol8, pp. 1659-1673 DOI: 10.1109/TNSE.2021.3068261

[17] E. B. Sifah, Q. Xia, K. O. -B. O. Agyekum, H. Xia, A. Smahi and J. Gao, "A Blockchain Approach to Ensuring Provenance to Outsourced Cloud Data in a Sharing Ecosystem," in IEEE Systems Journal, vol. 16, no. 1, pp. 1673-1684 March 2022, DOI: 10.1109/JSYST.2021.3068224.

[18] Meng Shen, Gaopeng Gou, Qi Xuan, “Security and privacy of blockchain”. Blockchain: Research and Applications, Volume 4, Issue 1, 2023, 100130, ISSN 2096-7209, pp. 1-92023https://doi.org/10.1016/j.bcra

[19] Sudeep tanwar “Basics of Cryptographic Primitives for Blockchain Development”. Studies in Autonomic, Data-driven and Industrial Computing, Springer Chapter 2022, pp 83–111.

[20] Vittorio Capocasale, Danilo Gotta, Guido Perboli, “Comparative analysis of permissioned blockchain frameworks for industrial applications”. Blockchain: Research and Applications, Volume 4, Issue 1, 2023, 100113, ISSN 2096-7209, pp. 1-13. DOI: https://doi.org/10.1016/j.bcra.

[21] Sudeep tanwar “Mining Procedure in Distributed Consensus”. Studies in Autonomic, Data-driven and Industrial Computing, Springer Chapter 2022, pp 191–209.

[22] Shadab Siddiqui, Manuj Darbari, and Diwakar Yagyasen. 2020. “Enhancing the Capability of Load Management Techniques in Cloud Using H_FAC Algorithm Optimization”. Int. J. e-Collab. 16, 2020, 65–81. DOI: https://doi.org/10.4018/IJeC.2020040105

[23] C. Li, J. Xiao, X. Dai and H. Jin, "AMVchain: authority management mechanism on blockchain-based voting systems", Peer-to-Peer Networking and Applications, vol. 14, no. 5, pp. 2801-2812, 2021. DOI:https://doi.org/10.1007/s12083-021-01100-x

[24] S. Vemula, R. Kovvur and D. Marneni, "Secure Electronic voting System Implementation Using CryptDB", SN Computer Science, vol. 2, no. 3, 2021. DOI: 10.1007/s42979-021-00613-9

[25] A. Shankar, P. Pandiaraja, K. Sumathi, T. Stephan and P. Sharma, "Privacy preserving Electronic voting cloud system based on ID based encryption", Peer-to-Peer Networking and Applications, vol. 14, no. 4, pp. 2399-2409, 2021. DOI: 10.1007/s12083-020-00977-4

[26] M. Saqib et al., "Anonymous and formally verified dual signature based online Electronic voting protocol", Cluster Computing, vol. 22, no. 1, pp. 1703-1716, 2018. DOI: 10.1007/s10586-018-2162-7

[27] S. Zhang, L. Wang and H. Xiong, "Chaintegrity: blockchain-enabled large-scale Electronic voting system with robustness and universal verifiability", International Journal of Information Security, vol. 19, no. 3, pp. 323-341, 2020. DOI: https://doi.org/10.1007/s10207-019-00465-8

[28] Ana Oliveira, Yosef Ben-David, Susan Smit, Elena Popova, Milica Milić. Machine Learning for Forecasting and Predictive Modeling in Decision Science. Kuwait Journal of Machine Learning, 2(3). Retrieved from http://kuwaitjournals.com/index.php/kjml/article/view/199

[29] Singh, H., Ahamad, S., Naidu, G. T., Arangi, V., Koujalagi, A., & Dhabliya, D. (2022). Application of machine learning in the classification of data over social media platform. Paper presented at the PDGC 2022 - 2022 7th International Conference on Parallel, Distributed and Grid Computing, 669-674. doi:10.1109/PDGC56933.2022.10053121 Retrieved from www.scopus.com

[30] Joseph Miller, Peter Thomas, Maria Hernandez, Juan González, Carlos Rodríguez. Exploring Deep Learning Models for Decision Science Applications. Kuwait Journal of Machine Learning, 2(3). Retrieved from http://kuwaitjournals.com/index.php/kjml/article/view/198