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

Block Chain Assisted Document Updation using Positional Index Altering Scheme in Cloud Server: Asymmetric Searchable Encryption Approach

Beena G Pillai[1] , Dayanand Lal N[2]

Submitted: 27/01/2024 **Revised**: 05/03/2024 **Accepted**: 13/03/2024

Abstract: Users have the option to share or use their data on the cloud. Cloud computing provides a great deal of convenience for users. However, it also brings up a number of security issues. Data owners may be unable to fully trust cloud servers and lose control over their data, which is one of the key security risks of cloud computing. One of the main issues with the existing system is security when changing files. Ensuring security during updates and encrypting the submitted material is the most significant problem. This paper presents an index-altering approach in Cloud Server that uses a Positional Index to facilitate blockchain-assisted document updating. In this situation, use a smart contract to utilize trustworthy cloud computing and Blockchain technologies. The work's implementation of forward and backward privacy and document updating is its most crucial component. It also fends off both active and passive assaults. The Diffie–Hellman (DH) key exchange protocol, the Positional Index Altering Scheme, the Fuzzy duo Trapdoor key search, the Homographic Asymmetric El-Gamal encryption technique, and the Blockchain smart contract are the security features of the suggested scheme. In a decentralized search system, the entire evaluation is validated.

Keywords: *Asymmetric Encryption, Blockchain, Ethereum, positional index-altering scheme, Smart contract.*

1. Introduction

A blockchain is capable of accurately recording every transaction in a decentralized network. It consists of several blocks linked together by referencing the previous block. The two primary components of a block are a block body and a block header [1]. The block header stores the block's attributes, such as its predecessor's date and hash value. Each full node maintains a copy of the ledger on the blockchain network, and different consensus techniques are used to ensure its consistency. The Bitcoin system is the first blockchainbased application to be used. By maintaining a distributed ledger, the Bitcoin system establishes a decentralised, transparent, and fault-tolerant transaction paradigm that satisfies the specifications of an entirely new cryptocurrency architecture [2].

Businesses have recently entered the cloud computing era, where it is now conceivable to turn almost anything into an online service (XaaS) [3]. Strangely enough, this isn't the case with the e-voting services yet, mostly because previous instances of the usage of e-voting in political elections have highlighted important crucial flaws, including low transparency and susceptibility. However, according to our argument, there are several situations in everyday life when an electronic vote may be successfully used. Elections in private or limited organizations are only a few examples. Consider the election of the executive officers of a firm, where each shareholder is granted one vote multiplied by the number of chosen officers for each share they own [4]. The rigidity of the rules for such voting systems may reduce their inclusiveness in some circumstances (the voting task is carried out with a smaller pool of voters available) or have an impact on how the organization functions, as meeting all the requirements for a valid vote may delay the appointment of officers. Due to their widespread use, electronic voting systems can drastically lower the expenses associated with their deployment and verification, making voting easier and more motivating. Additionally, Blockchain technology enables a comprehensive solution for all security and dependability demands imposed by voting processes[5]. Blockchain, in comparison, is a static, monolithic technology solution heavily dependent on the specific business case being addressed in the given state of the art.

The development of decentralized applications (dApps), a hybrid of classic cloud applications and new blockchain-enabled applications, is one specific application of the push towards decentralization (i.e., the smart contract)[6]. "Blockchain-based cloud applications" to make them sound more understandable (BCP for short). This article interchangeably refers to decentralized and cloud applications built on blockchains because they both refer to the same idea.

A blockchain provides computation power and a permanent location for data storage in cloud computing. This

[1] *GITAM UNIVERSITY, Begaluru, Karataka*
ORCID ID : https://orcid.org/0000-0001-7040-0083
[2] *GITAM UNIVERSITY, Begaluru, Karataka*
ORCID ID : https://orcid.org/0000-0003-3485-9481
Corresponding Author Email: bpillai@gitam.edu

capability's core is smart contracts, often known as programmable transaction scripts[7]. To accomplish flexible processing logic, first-generation blockchains like Bitcoin, by design, employ a straightforward transaction script (also known as smart contracts). Its application outside of payment settlement is limited because it needs Turing-complete capacity and the Unspent transaction output account model, even if this permits nontrivial transaction settlement logic such as escrow services, micropayment channels, and private transactions. As a result, the Ethereum blockchain was suggested as a solution, and it later developed into the standard design for open, decentralized cloud computing systems[8]. In contrast to Bitcoin, Ethereum takes a simpler method of holding transactional entities, simulating each transacting party as an individual account. More specifically, there are two sorts of accounts in Ethereum: smart contracts and externally owned accounts (or EOAs). An integer identifier with a length of 160 bits that is used to identify each account uniquely can be obtained by using their respective addresses. Ethereum, every account is kept in a state database with its distinct state preserved, and it is directly maintained on the blockchain technology[9]. The state of an account consists of four fields: Nonce, which is used to avoid replay attacks; balance, which represents the amount of Ether (or ETH), the native coin of Ethereum; storage Root, It is arranged as a Merkle tree and represents account-owned storage data; codeHash, which stands for self-governance code. In this instance, storageRoot and codeHash are essential to smart contracts.

2. Contributions

An overview of the principal contributions of this work is provided below:

- To secure the privacy of the data, the locally trained model's weights were encrypted using a homomorphic encryption method.
- To Implement a Positional Index, Altering Scheme is utilized via blockchain and encryption, which protects the index and encrypted files from leaking sensitive information.

3. Outline

The Organization of the article is as follows.

Section 4 Contains the Related Work, Section 5 Contains the Background of the research work, and Section 6 Contains the Results and Analysis. Finally, Section 7 is the conclusion of this article.

4. Related Work

Wenzheng Zhang [10] proposes a cryptographic method that lets senders and recipients search encrypted keywords together. It is referred to as PEBKS, or public-key encryption with bidirectional keyword search. A PEBKS scheme's formal specifications and indistinguishable security model reflect the situation. A PEBKS system whose security is dependent on the random oracle model's solution to the well-known hard problem of bilinear Diffie-Hellman. A formal security notion to prevent adaptive chosen keyword attacks was proposed, together with the PEBKS scheme concept.

Fahimeh Zare [11] proposed a new type of asymmetric searchable encryption is called secure public key searchable encryption (SSAE). They claimed that their strategy is resistant to counterfeit attempts. Demonstrating a forgery attack on this technique, it demonstrates the inadequacy of SSAE's security. Modify the SSAE after that to fend against this forgery attempt.

Jianyi Zhang [12] proposed a revolutionary technique that makes it possible to have an inverted index structure, do advanced searches, and dynamically update the information. The results of a thorough examination and numerous experiments show that the procedure is both effective and secure. In the MDO architecture, there is an issue with effective and verifiable security keyword search. H.S. Rhee et al. [13] the SAE method achieves security for both the index and the trapdoor. Deterministic encryption and free search with logarithmic time pairing. The secure index and trapdoor are accessible to the adversaries, who can also use the search algorithm.

Siyi Lv [14] propose that FFSSE, offering the best performance in the literature is a recently designed flexible forward safe SSE algorithm. regarding speedy token creation, speedy search processes, and O(1) update complexity In this example, it also permits add and remove operations. It uses a novel "key-based blocks chain" technique that guarantees forward privacy directly on index tree structures, including key-value structures, by utilising symmetric cryptographic primitives. Ming Zeng [15] proposed An innovative searchable asymmetric encryption strategy is created to facilitate in a multi-client paradigm, sub-linear boolean searches are made on encrypted data and is derived from the significant finding that numerous clients continually contribute to and search the cloud-based outsourced database. Public key searchable encryption and symmetric searchable encryption are combined for the aim of establishing the system, and after that, a novel secure inverted index is designed. An extensive system security study is also part of the simulation-based security definition. Every client has a public key and secret key pair in order to nontrivially organise the outsourced database.

Yu Wei [16] proposed a symmetric encryption primitive that can be used to create the keyed-block chain in a forward secure SSE method that enables both add and delete operations simultaneously. To reduce client-side storage, it

uses the one-way permutation function. SSE can safeguard the confidentiality of the data content to a certain extent, but because it uses deterministic encryption, it is simple to detect leaks such as repetitive searches and other intrusions on the cloud server. Due to forward privacy, a malicious server cannot tell whether a recently added document matches earlier search criteria. Forward secure properties are inherent to the keyed-block chain. Therefore, it would be interesting to explore new applications that seek forward security using the keyed-block chain technique. Making the keyed-block chain into a multi-level linked list that resembles a tree structure has advantages. Baodong Qin[17] proposed a PAEKS (Public-key Encryption with Keyword Search) security model that takes into account both keyword guessing and specific multi-ciphertext threats and shows that it is secure in light of the new PAEKS security model. This scheme uses an identity-based key exchange protocol to ease the administration of the data sender's keys. This novel security architecture, known as multi-ciphertext, uses public-key authentication with encryption and keyword search. The concept of multi-ciphertext indistinguishability represents a real-world method of relating two encrypted data.

Zehong Chen [18] proposed an innovative multi-user Boolean keyword search method (MBKSS) is used to quickly find results for Boolean queries while preventing user-data owner query interactions. A new homomorphic cryptosystem with partial decryption, this method might serve as the basis for the creation of a fast ranking search protocol (FRSP). J. Baek et al.[19], In the cloud, conjunction and disjunction can be allowed simultaneously inside each keyword field to the public key encryption with conjunctive and disjunctive keyword search (PECDK) technology. Prime-order bilinear groups serve as the basis for it, and its entire security can be demonstrated using the standard model. It is completely secure in the classic way and is built in prime-order bilinear groups.

5. Background

5.1. Blockchain-enabled Searchable Encryption

Recent research has concentrated on fixing current blockchain-based mechanisms. Blockchain-based encrypted keyword search was the subject of a study completed by Cai et al.[20] By integrating encryption with keyword search and employing a distributed hash table technique, the researchers discovered the problem of hostile nodes potentially manipulating search results. As the majority of nodes use a selfdetermining method, the suggested remedy might locate and get rid of malicious nodes. J.W. Byun et al.[21] suggested a SEPSE, a blockchain-assisted PKE, to protect against Keyword Guessing Attacks (KGAs). This paper proposed several strategies, such as regular key renewal, screening key encryption, and key request monitoring, to reduce the probability that KGA will succeed. The work created a key aggregation searchable encryption technique that is resistant to CPA in order to address the key leaking issue; some methods used broadcasted transactions to help verify the search result. For instance, Searchchain was one of the techniques along this technical path. D. Boneh[22] et.al, it was added to the Obvious Keyword Search with Authentication (OKSA) system in order to provide private user key encryption. The novel OKSA approach addressed the traditional Oblivious Keyword Search (OKS) constraints by providing keyword search authorization. It was proposed that Searchchain might be used to enhance privacy-preserving when users' access authentication was verified by CSPs using a predefined term.

A blockchain-based time commitment system using several types of transactions was proposed by Y. Zhang et al[23]. This concept will penalize dishonest parties with bitcoin compensation without dependable third parties (TTP). B.J. Wang et al. [24] proposed two-sided verification in a searchable encryption scheme in subsequent work. Malicious service providers and owners of data may both face sanctions. The search results were checked by the root of a Merkle tree that the authors built using ciphertext leaves. The payment fairness was founded on time commitment, just like Bpay. Data integrity checks were handled using the incremental hashing approach known as multi-set hashing. In this paradigm, there were two different kinds of participants[25]. Client Peers initially acted as the data owner while storage peers supplied the services. Client Peers asked storage peers for an authenticated cypher text search. Dynamic updates and optimised storage overhead were also features of this strategy.

Although verification systems could yield sound search results, miners could still forego confirming complicated transactions to concentrate on extremely profitable mining activity. The Verifier's Dilemma is the term used to describe the phenomena[26]. Authors investigated the use of smart contracts in their work to offer soundness keyword searches without requiring a laborious data owner verification process. The search algorithm could be integrated into smart contracts, ensuring accurate results only when the blockchain contract was executed correctly.

There was no longer a requirement for the tedious procedure of checking the data that was searched. Additionally, to reduce the computational complexity, encrypted indexes were stored by the author. The gas cost was also decreased by packing. Additionally, this strategy uses smart contracting to implement equitable payment. Fair payment practices could reward the righteous and deter the dishonest. Fairness was guaranteed in both single-user and multi-user settings via time commitment. For instance, Zhang et al. [12] implemented a fair payout to promote ethical behavior

during the SSE process. Blockchain was used to keep the user's file index, but public clouds were used for file storage. The smart contract also included a time obligation for fairness. When determining fairness, both single-user and multi-user settings were taken into account The work was expanded upon in a later study by Chen et al.[18] into a situation when different health agents asked questions during the exchange of electronic health records.

In contrast to Hu's work, Sophisticated logic expressions were used to generate and store the EHR index on a blockchain. Through this effort, the owner of the data could fully control who may access it. Zhang et al.[10] presented a searchable public keyword encryption system that handled medical data in the context of sharing private health information.

5.2. Methodology

- **Data Query Algorithm:** A significant amount of data (designated as D) is split into several data files (designated as D1, D2, D3,...) and placed on a cloud server to increase query efficiency. The information is encrypted and kept on a cloud server for data privacy. Since the cloud server allows for the manipulation of encrypted data, blockchain technology is connected with it. Every encrypted document is verified by the consensus process and registered on the blockchain. Before the document is encrypted, searching for keywords W1, W2, and W3 is challenging. Document D is used to identify Wm, and Index Table I is created. A trapdoor will be created, delivered to the cloud server, and sent to the smart contract if the user wants to search for a particular term. The consensus process is used to categorize and validate incoming requests, and the smart contract is automatically carried out. The client receives the results of the search after it is finished, and they can download the necessary data and information about trapdoors that are kept in the blockchain ledger.
- **Homographic Encryption:** A homomorphic cryptosystem for a given message space M is a quadruple (K, E, D, A) of time-based algorithms predicted probabilistically and satisfying the necessary conditions.

– *Key Generation (K)* K stands for the key space, and a key pair (ke, kd)= kϵK.The K element has a significant impact on calculation algorithms

– *Encryption (E):* is the process of using key 'ke' on a message mϵM to create a ciphertext c in cipher-space C, where cϵC

– *Decryption (D):* involves creating mϵM by using the key 'kd' on an encrypted message c; Homomorphic Property (A): is a scheme where m1, m2ϵM holds only when m3 = m1m˙ 2; that is, c1, c2ϵC must yield a third element, c3ϵC.

Mathematical processes like multiplication, summation, and logic XOR operations could be supported by homomorphic encryption. However, systems that enable both types of computations are referred to as Fully Homomorphic Encryption (FHE) systems.

The majority of consumers use the cloud through public cloud services. Blockchain and cloud integration provides a solution to the secrecy issue. A likely answer is homomorphic encryption (HE), which, when stored in the cloud, encrypts client data so that it can be partially altered without decrypting it. Multiplicative Homomorphic Encryption: It's important to remember that different academics describe separate cryptographic techniques when talking about HE. At the same time, the ElGamal cryptographic system, which utilises the asymmetric public-key encryption technique, has also been dependent on a multiplicative homomorphism. With this method, the key in a cyclic group is provided in order of a given generator index. The generator (T), exponent product (L), and function of a cyclic group (G) comprise the public key. An order function (d) is the exponent of the ElGamal encryption.

Additive Homomorphic Encryption: An additive homomorphism, an asymmetric probabilistic encryption model with characteristics identical to the ElGamal model, is computed based on the Parlier encryption model. To encrypt and decode messages, this architecture uses separate private and public keys. Additionally, the encryption model uses two randomly chosen prime integers to compute the Greatest Common Divisor (GCD).

- **Evaluation:** The encrypted image undergoes homomorphic procedures to generate a new encrypted image, while the decryption process offers comparable capabilities. The matching pixels of the two encrypted images undergo this homomorphic procedure. Assume that cp2 = s2, t2 and cp1 = s1, t1. These two encrypted images undergo a simple addition procedure that yields the new ciphertexts, s3 and t3. Upon decrypting a fresh ciphertext, cp3, the message bit msg3, which is identical to msg1 + msg2, will be obtained. proposed a homomorphic encryption approach to maintain privacy by encrypting and decrypting the model's gradients. Secured is the local model that both encrypts and decrypts the gradients using the homomorphic encryption technique.
- **Passive leakage attack in asymmetric searchable encryption system architecture:** In asymmetric searchable encryption (ASE) system architecture, a passive leakage attack is a security risk in which a malicious party with access to the encrypted data and potentially some auxiliary data or metadata tries to learn more about the plaintext content of the encrypted documents without actually decrypting them. The

context of public key searchable encryption, passive leakage attacks refer to a type of security risk where an adversary attempts to learn more about the plaintext content of encrypted documents without decrypting them. Siyu Xiao et.al [27] an encryption method known as asymmetric searchable encryption enables users to search over encrypted data without disclosing the content of the data or the search query. It is very helpful when data privacy is essential, such as cloud storage or secure information retrieval systems. In ASE, passive leakage attacks can take many different shapes, but they frequently entail deducing information by examining the patterns and metadata linked to the encrypted data.

5.3. ASE System Architecture

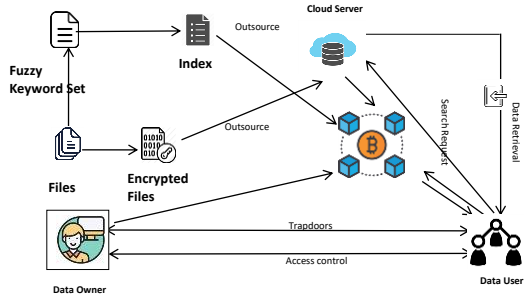

Fig. 1. Blockchain-based Asymmetric Searchable Encryption Architecture

- **Data Owner:** Sensitive data must be encrypted, and access control must be managed by the owner. The Data Owner uses blockchain to maintain a distributed ledger of access permissions and records access regulations. On the blockchain, access policies are specified in smart contracts to guarantee secure data sharing.
- **Blockchain Network:** Nodes that validate and store transactions and smart contracts make up the blockchain network. Smart contracts specify how encrypted data can be accessed and enforce access policies. The blockchain ledger keeps track of access histories and control procedures, enabling transparency and auditability.
- **Untrusted Server:** The encrypted data is stored on the Untrusted Server, which also answers to search requests made by Authorised Users. It cannot read encrypted data because it lacks access to the decryption keys. The Untrusted Server talks with the blockchain network to confirm user access permissions.
- **Authorized Users:** Authorized Users are people or organizations that want to search the encrypted data for specific information. Users send search requests to the Untrusted Server, which handles the requests by the access control regulations set down in the blockchain.

5.4. Obstacles and Assaults that ASE

- **Eavesdropping and Data Exposure :** Traditional search systems are susceptible to eavesdropping because they send data and requests unencrypted over a network. ASE encrypts the data to prevent data exposure due to network eavesdropping, rendering it unreadable to unauthorized parties.
- **Server Side Attacks:** Data is frequently kept on unreliable servers, such as those operated by cloud storage companies. These servers might be compromised if they aren't properly protected, resulting in data breaches. ASE enables data owners to safely store their data on untrusted servers without disclosing the plaintext to the server.
- **Key Leakage Hierarchy:** Let K = K pub, K sym, K Priv, K root, K master, K hsm, K escrow represent a set of cryptographic keys, where

  K pub : PublicKeys

  K sym : SymmetricEncryptionKeys

  K priv : P rivateKeys(AsymmetricEncryptionKeys)

  K root : RootKeys(KeyDerivationKeys)

  Kmaster : MasterKeys

  K hsm : HardwareSecurityModule(HSM)Keys

  K escrow : EncryptionKeyEscrowKeys
- **Implications of Key Leakage:** The effects of key leakage can be discussed using conditional statements. For example, if a key with sensitivity level s is broken into, the following could happen to keys with sensitivity levels higher than s: If S(K i) = sandK i is compromised, then for all K jwhereS(K j) > s, K j may be at risk. In mathematical notation: K i, K jєK : [S(K i) = sKˆ i compromised] → [S(K j) s → K j at risk]
- **Key Recovery**: A mathematical model can be used to specify the circumstances and procedures for key recovery for encryption key escrow keys. This may entail mathematical formulas and cryptographic methods connected to key recovery activities. The particular effects of key leakage might differ significantly depending on the system's architecture and use cases since real-world key management systems use a variety of cryptographic algorithms, security guidelines, and access control techniques.

5.5. Security of Asymmetric Searchable Encryption

Table I represents the Security Measures Against Server Side Attacks. An asymmetric searchable encryption (ASE) encryption promises to protect data privacy while enabling secure search functionality over encrypted data.

Mathematical formulas and official security definitions are often used to analyze the security of ASE schemes.

- **Sematic Security**: The security of encryption schemes, such as ASE, is frequently defined as semantic security, a key concept in cryptographic security. Pr[Enc(pk, m1) = c] Pr[Enc(pk,m2) = c] Where: • Enc(pk, m)represents the encryption of plaintext m under the public key pk. • C represents the resulting ciphertext. • Pr denotes probability. • M1 and m2 are two plaintexts of the same length.

Table 1. Security Measures against Server side Attacks

Security Measure	Traditional Server-Side Security	Blockchain-Based Security
Data Integrity	Hash Functions (e.g., SHA-256)	Hash Functions (e.g., SHA-256) and Immutable Blockchain.
Authentication	Usernames and Passwords.	Digital Signatures and Decentralized Identity.
Access Control	Role-Based Access Control (RBAC).	Smart Contracts and Access Tokens.
Auditing and Traceability	Logging and Event Monitoring.	Transparent Blockchain Ledger.
Decentralization	Centralized Server Infrastructure.	Decentralized Blockchain Network.
Resilience to Attacks	Vulnerable to Single Point of Failure.	Distributed Nodes, Resilient to Attacks.

- **Trapdoor Function:** In ASE, the idea of trapdoor functions is used to enable searching over encrypted data. Authorized users can produce trapdoor keys using these mathematical operations, allowing them to conduct searches. To prevent unauthorized access, the security of these trapdoor features is essential. Based on the particular ASE scheme, the mathematical characteristics of trapdoor functions may change.
- **Defending Against Passive Leakage Attacks**: Table 3 represents the attack resistance of different techniques. ASE systems make use of a variety of cryptographic and privacy-preserving measures to ward off passive leakage attacks. To ensure that search queries and results do not reveal sensitive information, these strategies include query obfuscation, noise addition to search queries, and using cryptographic primitives like homomorphic encryption. Additionally, secure key management procedures are crucial to prevent unauthorized access to the decryption keys. It's important to remember that passive leakage attacks might be difficult to prevent completely, and the success of the defense mechanisms depends on the particular ASE scheme and the design decisions made during its development. To improve defense against such attacks, researchers are constantly creating more effective and safe ASE approaches. L = (LSetup, LSearch, LUpdate) is how further represent the leakage function. Let simulator S and adversary A be components of the ASE scheme = (Setup, Search, Update). The next two games are described. Real A: The scheme is faithfully carriedout. The adversary A watches the actual transcript of the scheme and outputs a bit with the values 0 and 1.

Table 2. Comparison of Computation Overhead

Strategies	Index File Size Cost	Index Cost with Encryption	Trapdoor Generation cost	Search Cost
Bonesh[22]	$(\|\|G_1\| + \|Z_q^*\|)*N_w$	$(T_h + 2T_e + T_p) * N_w$	$(T_h + T_e) * Q_w$	$(T_h + T_e) * Q_w * N_w$
Lu[28]	$(\|G_1\| + \|Z_q^*\|)*N_w$	$D(5T_h + 4T_e + 2T_p) * N_w$	$(4T_h + T_m + 4T_e) * Q_w$	$(T_h + T_e) * Q_w * N_w$
Sultan[29]	$N_w * (\|G_1\| + 2\|G_2\| + v +$	$(v + 2N_w$	$N_attr * (\|G_1\| + 2\|G_2\| +$	$[(3—G_1\| + 4T_p) +$

	$Z_q^*\|$)	$+1)\|G_1\|+2\|G_2\|vT_p$	$2T_p$)	$(N_attr\|G_1\|+2T_p)]$*Q_w * N_w
ASE	$(\|G_1\|+\|G_3\|+\|Z_q^*\|)*N_w$	$(T_h+4T_s+T_m+2T_p)*N_w$	$(T_h+4T_s+T_m+2T_p)*Q_w$	$(2T_h+2T_e+2T_e)*Q_w*N_w$

Table 3. Resistance of Attack – Comparative Study

Model	Resists Chosen Keyword Attack	Resists Keyword Guessing Attack	Authenticated Index	Index resists Statistical Attack	Trapdoor resists Statistical Attack	Resists Collusion
Bonesh[22]	✓	X	X	✓	X	X
Lu[28]	✓	✓	✓	✓	X	X
Sultan[29]	✓	✓	✓	✓	✓	X
ASE	✓	✓	✓	✓	✓	✓

6. Result and Analysis

The construction and application of Block Chain Assisted Fuzzy Keyword Search Based on Homographic Asymmetric El-Gamal Encryption Approach in Cloud

Server are highlighted in this section's discussion of system implementation. Precision is a statistic for the precision of search results produced by the ASE system in response to user queries in the context of blockchain-based asymmetric searchable encryption (ASE). Precision (P) is calculated as (Total Number of Retrieved Documents / Relevant Documents Retrieved). The number of documents in the search results that are truly pertinent to the user's query is indicated by the phrase "A number of Relevant Documents Retrieved." Overall Count of Documents Obtained. The effectiveness of suggested schemes is influenced by adding fuzzy keywords and grouping documents. P k = Kk, where K is the number of actual top-k records that the server actually sent back to the data consumer, specifies the precision. They demonstrated the close relationship cbetween standard deviation and secrecy while also demonstrating how increasing standard deviation reduces accuracy. They found a clear connectionbetween standard deviation and secrecy, but they also showed that precision decreases as standard deviation increases. The Homographic Asymmetric El-Gamal (HAEE) encryption technique serves as the foundation for the suggested approach. HAEE's accuracy has a standard deviation of 0.03.

Table 4. Time Cost for Index Generation

The number of retrieved documents	50	60	70	80	90	100	110	120
ASE	95.87	97.94	95.99	96.5	98.56	96.18	96.11	98.51
DMKRS	95.3	95.9	95	95.2	96	95.1	95	97.2
EDMRS	93.03	93.7	94.7	93	92.2	91.1	94	92.4

Table 5. Comparison of Computation Time to Generate Index

Model	100	200	300	400	500	600	700	800	900	1000
ASE	5	15	25	28	45	48	50	60	68	75
Sultan [29]	20	30	40	50	60	70	80	90	105	125

Lu [28]	25	35	60	70	80	95	105	130	150	170
Bonesh [22]	25	50	75	95	105	135	165	185	205	230

Table 6. Trapdoor Generation Computation Time Comparison

Model	100	200	300	400	500	600	700	800	900	1000
ASE	5	15	30	40	50	60	75	90	100	110
Sultan [29]	50	75	100	125	150	175	200	225	250	275
Lu [28]	25	50	75	90	110	135	150	175	190	210
Bonesh [22]	55	80	105	125	145	170	195	220	245	270

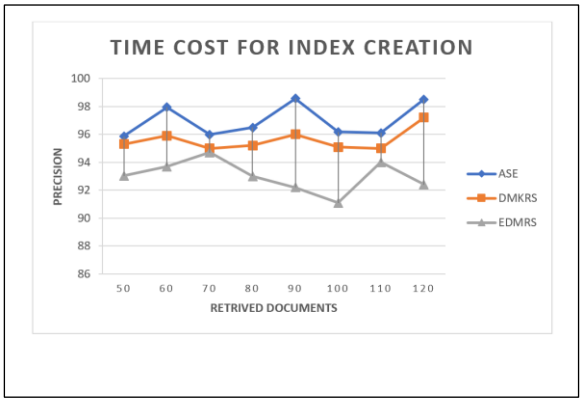

Fig. 2. Time Cost for Index Generation

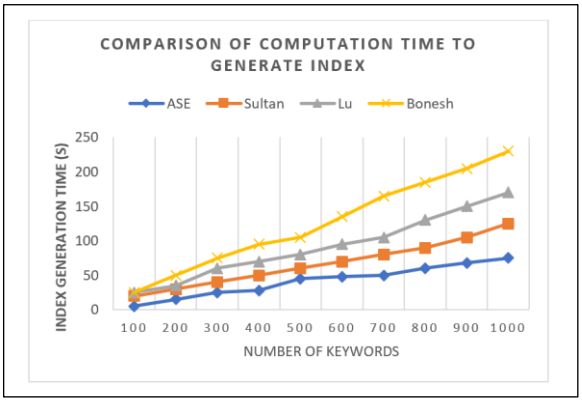

Fig. 3. Comparison of Computation time to Generate Index

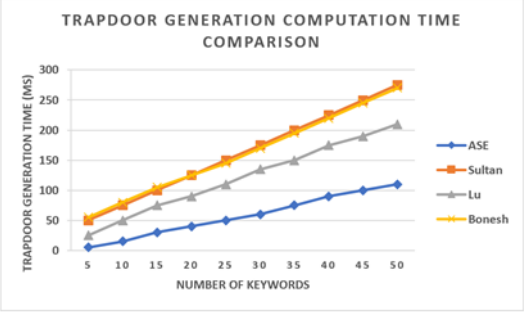

Fig. 4. Trapdoor Generation Computation time Comparison

7. Conclusion

Cloud computing provides a great deal of convenience for users. Data owners may be unable to fully trust cloud servers and lose control over their data, which is one of the key security risks of cloud computing. To maintain security, a blockchain smart contract is used in the network, ensuring that every peer-to-peer node has a copy of the most recent blockchain ledger. A unique positional index-altering scheme is used via blockchain and encryption to update files by modifying, inserting, and deleting operations for file mod- ification and insertion. This prevents sensitive information from leaking out of the encrypted files and index. Blockchain technology is used to deliver search results. The creation of theinverted indices by the suggested effort also made the passive and active attacks easier. In order to prevent data leaks and update documents, the suggested effort offers both forward andbackward privacy. Smart contracts assist in removing irrelevant results from the search process, and the suggested system's overall evaluation is effective.

Acknowledgements

This research was supported/partially supported by Research Centre, GITAM University, Bengaluru, who provided insight and expertise that greatly assisted the research, although they may not agree with all of the interpretations/conclusions of this paper.

Author contributions

Beena G Pillai: Writing – review & editing – original draft, Methodology, Formal analysis, Investigation. **Dayanand Lal N:** Supervision, Validation, Resources.

Conflicts of interest

The Authors declare that they have no conflict of interest. They are not received any research grants from any Company or Agencies.

References

[1] C. Liu et al., "Search pattern leakage in searchable

encryption: Attacks and new construction," Information Sciences, vol. 265, pp. 176–188, 2014.

[2] K. Gai, K.-K. R. Choo, and L. Zhu, "Blockchain-enabled reengineering of cloud datacenters," IEEE Cloud Computing, vol. 5, no. 6, pp. 21–25, 2018.

[3] M. Andoni et al., "Blockchain technology in the energy sector: A systematic review of challenges and opportunities," Renewable and sustainable energy reviews, vol. 100, pp. 143–174, 2019.

[4] P. Chaudhari and M. L. Das, "Privacy preserving searchable encryption with fine-grained access control," IEEE Transactions on Cloud Comput- ing, vol. 9, no. 2, pp. 753–762, 2019.

[5] H. Li et al., "Blockchain-based searchable symmetric encryption scheme," Computers & Electrical Engineering, vol. 73, pp. 32–45, 2019.

[6] S. Wang, Y. Zhang, and Y. Zhang, "A blockchain-based framework for data sharing with fine-grained access control in decentralized storage systems," Ieee Access, vol. 6, pp. 38 437–38 450, 2018.

[7] M. Asif et al., "Blockchain-based authentication and trust management mechanism for smart cities," Sensors, vol. 22, no. 7, p. 2604, 2022.

[8] J. Niu et al., "Blockchain-based anti-key-leakage key aggregation searchable encryption for iot," IEEE Internet of Things Journal, vol. 7, no. 2, pp. 1502–1518, 2019.

[9] P. Jiang et al., "Searchain: Blockchain-based private keyword search in decentralized storage," Future Generation Computer Systems, vol. 107, pp. 781–792, 2020.

[10] W. Zhang et al., "Public-key encryption with bidirectional keyword search and its application to encrypted emails," Computer Standards & Interfaces, vol. 78, p. 103542, 2021.

[11] F. Zare and H. Mala, "Cryptanalysis of an asymmetric searchable encryption scheme," in 2017 14th International ISC (Iranian Society of Cryptology) Conference on Information Security and Cryptology (ISCISC). IEEE, 2017, pp. 82–85.

[12] J. Zhang et al., "Efficient and provable security searchable asymmetric encryption in the cloud," IEEE Access, vol. 6, pp. 68 384–68 393, 2018.

[13] H. S. Rhee et al., "Trapdoor security in a searchable public-key encryption scheme with a designated tester," Journal of Systems and Software, vol. 83, no. 5, pp. 763–771, 2010.

[14] S. Lv et al., "Forward secure searchable encryption using key-based blocks chain technique," in International Conference on Algorithms and Architectures for Parallel Processing. Springer, 2018, pp. 85–97.

[15] M. Zeng et al., "A searchable asymmetric encryption scheme with support for boolean queries for cloud applications," The Computer Journal, vol. 62, no. 4, pp. 563–578, 2019.

[16] Y. Wei et al., "Fsse: Forward secure searchable encryption with keyed- block chains," Information Sciences, vol. 500, pp. 113–126, 2019.

[17] B. Qin et al., "Public-key authenticated encryption with keyword search revisited: Security model and constructions," Information Sciences, vol. 516, pp. 515–528, 2020.

[18] Z. Chen et al., "Multi-user boolean searchable encryption supporting fast ranking in mobile clouds," Computer Communications, vol. 164, pp. 100–113, 2020.

[19] J. Baek, R. Safavi-Naini, and W. Susilo, "Public key encryption with keyword search revisited," in Computational Science and Its Applications–ICCSA 2008: International Conference, Perugia, Italy, June 30–July 3, 2008, Proceedings, Part I 8. Springer, 2008, pp. 1249– 1259.

[20] C. Cai, X. Yuan, and C. Wang, "Hardening distributed and encrypted keyword search via blockchain," in 2017 IEEE Symposium on Privacy- Aware Computing (PAC). IEEE, 2017, pp. 119–128.

[21] J. W. Byun et al., "Off-line keyword guessing attacks on recent keyword search schemes over encrypted data," in Workshop on secure data management. Springer, 2006, pp. 75–83.

[22] D. Boneh et al., "Public key encryption with keyword search," in Advances in Cryptology-EUROCRYPT 2004: International Conference on the Theory and Applications of Cryptographic Techniques, Interlaken, Switzerland, May 2-6, 2004. Proceedings 23. Springer, 2004, pp. 506– 522.

[23] Y. Zhang et al., "Outsourcing service fair payment based on blockchain and its applications in cloud computing," IEEE Transactions on Services Computing, vol. 14, no. 4, pp. 1152–1166, 2018.

[24] W. BingJian, C. TzungHer, and J. FuhGwo, "Security improvement against malicious server's attack for a dpeks scheme," Int J Inf Educ Technol, vol. 1, no. 4, pp. 350–353, 2011.

[25] Q. Tang and L. Chen, "Public-key encryption with registered keyword search," in European public key infrastructure workshop. Springer, 2009, pp. 163–178.

[26] Q. Huang and H. Li, “An efficient public-key searchable encryption scheme secure against inside keyword guessing attacks,” Information Sciences, vol. 403, pp. 1–14, 2017.

[27] S. Xiao et al., “Asymmetric searchable encryption from inner product encryption,” in Advances on P2P, Parallel, Grid, Cloud and Internet Computing: Proceedings of the 11th International Conference on P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC–2016) Novem- ber 5–7, 2016, Soonchunhyang University, Asan, Korea. Springer, 2017, pp. 123–132.

[28] Y. Lu, J. Li, and Y. Zhang, “Secure channel free certificate-based searchable encryption withstanding outside and inside keyword guessing attacks,” IEEE Transactions on Services Computing, vol. 14, no. 6, pp. 2041–2054, 2019.

[29] N. H. Sultan et al., “Authorized keyword search over outsourced encrypted data in cloud environment,” IEEE Transactions on Cloud Computing, vol. 10, no. 1, pp. 216–233, 2019.