IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 15 August 2023, accepted 3 September 2023, date of publication 6 September 2023,
date of current version 15 September 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3312609

RESEARCH ARTICLE

Enhancing Secure Communication in the Cloud Through Blockchain Assisted-CP-DABE

G. SUCHARITHA[1], VEDULA SITHARAMULU[2], SACHI NANDAN MOHANTY[3], (Senior Member, IEEE), ANJANNA MATTA[4], AND DEEPA JOSE[5], (Senior Member, IEEE)

[1]Department of Electronics and Communication Engineering, Institute of Aeronautical Engineering, Dundigal, Hyderabad, Telangana 500043, India
[2]Department of Computer Science and Engineering, GITAM School of Technology, GITAM (Deemed-to-be-University), Hyderabad, Telangana 530045, India
[3]Department of Computer Science and Engineering, Vardhman College of Engineering (Autonomous), Hyderabad, Telangana 501218, India
[4]Department of Mathematics, Faculty of Science and Technology (IcfaiTech), ICFAI Foundation for Higher Education, Hyderabad 502300, India
[5]Department of Electronics and Communication Engineering, KCG College of Technology, Chennai 600097, India

Corresponding author: Anjanna Matta (anjireddyiith@ifheindia.org)

This work was supported by the ICFAI Foundation for Higher Education, Hyderabad, India.

ABSTRACT The use of encryption is essential to protect sensitive data, but it often poses challenges when it comes to locating and retrieving information without decryption. Searchable encryption provides an effective mechanism that achieves secure search over encrypted data. In this paper a new approach to address the fine-grained search and to protect sensitive data, Blockchain Assisted ciphertext policy decentralized attribute-based encryption (BA-CP-DABE) in cloud has been developed. The CP-DABE is employed to manage data access, secure key generation, while the immutability of blockchain ensures the confidentiality of ciphertext. By leveraging searchable encryption, it becomes possible to securely search encrypted data stored on the blockchain. Keywords are encrypted using attribute-based encryption and stored on a remote server, along with the corresponding ciphertext in the blockchain. One of the significant challenges in this approach is the assumptions-based technique i.e., bilinear mapping, which involves keyword ciphertext and trapdoor security. However, through extensive numerical experiments, the system has demonstrated its ability to generate key and trapdoor structures, as well as effectively find keywords within the encrypted data.

INDEX TERMS CP-attributed based searchable encryption, blockchain, cloud, data ciphertext, fine-grained data access, keyword ciphertext.

I. INTRODUCTION

With the rise of cloud storage technology, it has become common for companies to utilize cloud service providers for their data storage needs [1]. However, as cloud servers cannot be fully trusted, encrypting data files before storing them on the cloud is necessary [2]. Unfortunately, this can result in inefficiencies when searching for data, as multiple files need to be downloaded and decrypted locally, which consumes a significant amount of network bandwidth. To tackle this problem, searchable encryption (SE) was introduced as a solution that allows for efficient and secure data search while maintaining data privacy [3]. By utilizing searchable encryption (SE), users can conduct efficient searches for specific keywords within encrypted data files, which helps to reduce the amount of communication and computation required [4]. Numerous SE schemes currently in use are based on essential public SE.

In their research, Gao et al. [5] explored the use of essential public SE in managing medical information, specifically developing a basic SE application for mobile medical systems. However, the researchers identified offline keyword guessing attacks as a potential security risk, leading them to propose a more secure public key SE scheme under the random oracle model [6]. Despite the advancements made, numerous public key-based SE approaches remain constrained by the requirement for one-to-one encryption and decryption, mandating knowledge of the recipient's identity for each encryption instance. This constraint hinders the effective control of access to encrypted cloud-stored data, neglecting the user's search capabilities and rendering practical applications inconvenient. Searchable encryption

The associate editor coordinating the review of this manuscript and approving it for publication was Jun Wang.

emerges as a cryptographic solution that empowers users to explore specific information within encrypted data, upholding data privacy and confidentiality. This technique proves particularly valuable in scenarios necessitating secure data storage and transmission, while retaining the ability to conduct searches or queries [7]. There are several different types of searchable encryption techniques, including:

Symmetric Searchable Encryption: This type of searchable encryption uses symmetric key encryption techniques to encrypt the data and allows authorized users to search the data using certain keywords or search terms. The search is performed on the encrypted data, and the results are returned in an encrypted form. Chai et al. [8] recommended a new approach called verifiable SSE scheme to offer verifiable searchability in addition to the data privacy.

Public Key Searchable Encryption: In this technique, public key encryption is used to encrypt the data and a public key is used to encrypt the search terms. The encrypted search terms are sent to the server, which uses the private key to search the encrypted data and return the results in an encrypted form. Li et al. [9] used keyword-searchable attribute-based encryption for efficient bigdata management of electronic medical records.

Searchable encryption has a variety of applications, including secure cloud storage, secure messaging, and secure search engines. By allowing users to search for specific data within encrypted data without compromising the security or privacy of the data, searchable encryption enables organizations to store and transmit sensitive information securely while still allowing authorized users to access and search the data as needed.

The SE schemes mentioned earlier enable data users to conduct searches using any keyword to retrieve encrypted data from the server containing their desired keywords. However, to design a SE scheme that authorizes keyword searches, researchers must incorporate attribute-based encryption technology, as data owners cannot effectively enforce access controls on outsourced data. The concept of Attribute-based encryption technology, which utilizes fuzzy identification to implement fine-grained data access control, was initially introduced in [10].

All of the aforementioned SE schemes provide data users with limitless search capabilities, allowing them to use any keyword to request encrypted material from the server that contains their desired keywords. In order to create a SE scheme with keyword search authorization, researchers need attribute-based encryption technology since data owners are unable to impose effective access controls to outsourced data information. The notion of attribute-based encryption, which implements the fine-grained access control of data through the mechanism of fuzzy identification, is originally proposed in [11]. It is a new form of cryptographic primitive. In [12] the authors have presented a CP-ABE cryptosystem designed to facilitate efficient and secure operations in a cloud environment. Their proposed system supports access policy complication and outsourced decryption. The cryptosystem employs a linear secret sharing (LSS) scheme [13], which enhances the access policy's expressiveness by supporting any monotonic access structures. Furthermore, the system uses a prime-order bilinear group and a matrix-based LSS scheme to improve its computational efficiency. To enable precise control over data access, the author of [14] proposed an attribute-based encryption system that integrates attributes into keys. Gupta et al. [15] subsequently introduced a fine-grained data SE approach, aimed at striking a balance between data outsourcing security and user experience. The author also highlighted its potential application in a secure mobile cloud environment. Embedding attributes in the attribute-based encryption approach is crucial for ensuring effective communication [16]. However, utilizing this approach to send trapdoors requires a secure communication channel, resulting in increased communication costs. In the context of semi-honest cloud storage, the authors of [17] and [18] proposed an attribute-based encryption method that enables a more comprehensive and adaptable access control approach by incorporating attribute-based encryption into critical processes.

According to the literature [19], an attribute-based SE system can be designed where the cloud server performs complex computing tasks, reducing the user's computational burden and enhancing flexibility when modifying the access policy. As most attribute-based SE schemes use cloud storage, concerns related to data security and privacy protection are increasingly prevalent. While cloud servers offer users access to convenient and extensive data storage services, the complexity of their security situation undermines customer confidence, as unauthorized individuals can potentially access the servers and data protection cannot be guaranteed.

The utilization of blockchain technology has created new opportunities to tackle data access and sharing challenges by providing a secure and unrestricted way of accessing and sharing data [15]. In their work, the authors in [20] emphasized the significance of storing data on the public chain and subsequently introduced a novel data deletion scheme based on blockchain technology. The proposed scheme enables data owners to verify the deletion result, thereby enhancing the transparency of the deletion operation, irrespective of how poorly the cloud server behaves. Zhang et al. [10] proposed a SE scheme that utilizes blockchain technology to ensure fairness and minimize computational overhead for users. To prevent unauthorized access to encrypted data, Liu et al. [21] proposed a trusted SE strategy based on cloud storage that is targeted towards criminal users and cloud service providers. The proposed strategy relies heavily on attribute-based encryption, specifically encryption that incorporates attributes into ciphertext. While blockchain technology can help ensure the integrity and immutability of policy-related information, access control systems in distributed networks often leak sensitive data information.

Tahir et al. [22] suggested a traceable, efficient, and privacy-preserving attribute-based searchable encryption technique in the blockchain in order to address the effectiveness of attribute encryption, privacy leakage, and critical abuse. Blockchain technology is used by the system to guarantee the immutability and integrity of data. Jiang et al. [23] addresses the challenge of conducting efficient multi-keyword searches over encrypted data stored on the blockchain while preserving user privacy. It proposes a solution that combines symmetric searchable encryption (SSE) and blockchain technology. Yan et al. [24] proposes a blockchain-enabled searchable encryption scheme that incorporates fair payment mechanisms. The main objective of the scheme is to provide secure and efficient keyword search over encrypted data while ensuring fairness in terms of payment for resource utilization. In [26] the authors presented a framework that combines attribute-based access control (ABAC) and blockchain-enabled searchable encryption to provide a flexible and privacy-preserving solution for multi-user search scenarios. In [27], the study delves into the development of a secure cloud storage framework, leveraging blockchain technology for access control mechanisms. By utilizing blockchain's inherent security features, the research aims to enhance the confidentiality and integrity of data stored in cloud environments while providing robust access control capabilities. This work contributes to strengthening the security landscape of cloud storage systems. Guo et al. [31] tried to explore a blockchain-driven ABE scheme, incorporating multi-authority functionality, tailored for on-demand medical services within telemedicine systems. The focus lies on achieving flexibility and efficiency in the secure exchange of medical data, thereby contributing to the advancement of secure telemedicine practices. Liu et al. [33] explored the integration of blockchain technology to enhance comprehensive key management within the context of Ciphertext-Policy Attribute-Based Encryption (CP-ABE) for data stored in cloud environments. By leveraging blockchain's capabilities, the research aims to address key management challenges and reinforce the security of cloud-stored data under the CP-ABE framework. This work contributes to advancing the management and protection of sensitive data in cloud-based systems.

In this paper, we have developed a novel distributed data-sharing scheme by combining blockchain technology and attribute-based searchable encryption technology. Our scheme is designed to achieve fine-grained searchable access to encrypted cloud data while considering factors such as low computational cost, policy privacy, attribute revocation, and dynamic authorization. By integrating blockchain and attribute-based searchable encryption technologies, we have overcome the challenges of fine-grained access control in a distributed environment. Our scheme provides a secure and efficient solution for data sharing while preserving the privacy of the data owner and allowing for dynamic authorization and revocation of access based on attributes.

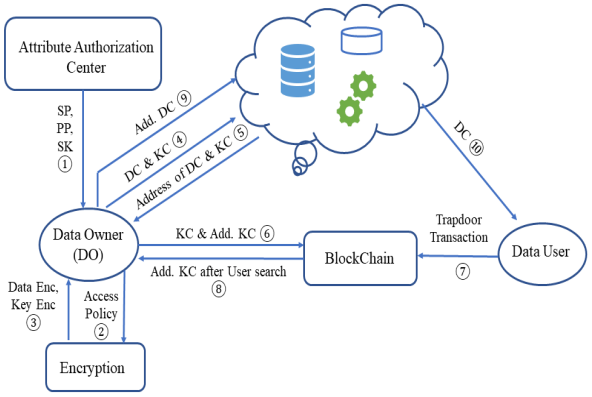


FIGURE 1. Framework of system model.

In summary, our study's main contributions are the development of a distributed data-sharing scheme that achieves fine-grained searchable access to encrypted cloud data, addresses the challenges of fine-grained access control in a distributed environment, and enables dynamic authorization and revocation of access based on attributes. The scheme provides a secure and efficient solution for data sharing while preserving the data owner's control over access to their data.

The organization of the paper as follows, section II gives about system and security model, section III deals about the proposed system, performance analysis in section IV and the conclusion in sectionV.

II. SYSTEM AND SECURITY MODEL

In this section, we mainly introduce the scheme's system model, formal definition and security model.

System Model

This research puts into practice fine-grained access control for encrypted data using cloud-based blockchain technology. On the cloud server, data files and encrypted keywords are first saved. Additionally, the blockchain stores the encrypted keywords' storage addresses on the cloud server. Data owner, various data users, cloud server, trusted attribute authorization center, and blockchain are the five entities that make up the system (consortium chain). In Figure 1, the system model is displayed.

1. **Decentralized Attribute Authorization Center:**It is a cryptographic technique that combines the principles of Attribute-Based Encryption (ABE) with decentralization, allowing for secure data sharing and access control in distributed and collaborative environments. In DABE, data is encrypted based on attributes, and decryption is granted to users who possess the required set of attributes. However, unlike traditional ABE, which relies on a centralized authority for attribute management and key distribution, DABE distributes these responsibilities across multiple nodes or authorities. DABE systems often involve multiple attribute authorities, each responsible for managing specific attributes. Users can possess attributes issued by

different authorities, and the combination of attributes determines access to encrypted data. The decentralized nature of DABE enhances privacy, scalability, and fault tolerance, making it suitable for scenarios where multiple organizations or entities need to collaboratively share and access encrypted data without relying on a single central authority.

2. **Data owner**: The data owner extracts the keyword set from the data file according to the agreed rules, encrypts the keywords with the access policy defined by himself, and finally uploads the data file ciphertext and keyword ciphertext to the cloud server. After receiving the ciphertext, the cloud server stores it and returns the storage address to the data owner. Next, the data owner establishes the reverse index relationship between the data file's ciphertext and the keyword's ciphertext in the cloud server's storage address. Finally, the data owner embeds the keyword ciphertext and its storage address into a constructed transaction and uploads it to the blockchain to form and broadcast the new block. Other data users on the blockchain are responsible for the new partnership.
3. **Cloud server:** Data storage services are offered by cloud servers. The cloud server saves the data file ciphertext and keyword ciphertext that the data owner has submitted and then sends the storage address back to them. When the keyword search is successful, the data owner uses the address provided by the blockchain to locally check the index relationship between the ciphertext of the data file and the ciphertext of the keyword. The cloud server will then receive a request, perform a search using the data file's ciphertext, and return the user's data file's ciphertext.
4. **Blockchain:** Within the blockchain, nodes provide data search capabilities. Initiating the process, the data owner generates a transaction embedding both the transaction itself and the encrypted keyword, along with its designated address, into the blockchain. As the broadcasted block reaches more users of the blockchain data, it attains verification. Executing the search algorithm, a blockchain node, driven by the incentive mechanism, aims to secure the reward by facilitating a user's trapdoor upload as a transaction. If the search proves fruitful, the node furnishes the storage address for the encrypted keyword to the data owner; otherwise, a failure signal is returned.
5. **Data users:** Users create search trapdoors using their private keys and desired keywords, upload the trapdoors to the blockchain as transactions, and the blockchain's nodes carry out searches using the transactions. If the search is successful, the blockchain node gives the data owner and the keyword ciphertext storage address. The data owner then uses the index relationship to identify the data file's ciphertext address and gives it to the cloud server. Locate the encrypted data file next, and then give the user access to the data file's ciphertext.

A. SECURITY MODEL

Keyword ciphertext indistinguishability security and trapdoor indistinguishability security of the scheme under chosen-plaintext attack is defined by probabilistic polynomial time game between attacker A and challenger B.

Game 1: Keyword ciphertext indistinguishability

The initial phase B runs the system to establish the algorithm output public parameters; A defines a challenge access tree T.

Stage 1: At this stage, A adaptively performs the following query of polynomial bounded degree.

Key extraction challenge. A adaptively asks B for the private key corresponding to the R_1, R_2,..,R_n attribute sets.

Keyword ciphertext query. A adaptively asks B for the ciphertext corresponding to l_1, l_2,..,l_m. During this process, none of the private keys that are asked for satisfies the access tree U.

Challenge: A submits two challenge keywords, x_0 and x_1, to B.

B randomly selects $\mu \in \{0,1\}$, encrypts x_μ to obtain the keyword ciphertext $J_{x\mu}$, and returns it to A.

Stage 2: A continues to initiate a series of queries corresponding to the attribute sets R_{q+1}, R_{q+2},..as in phase 1 and requires that none of the private keys obtained by the question satisfies the access tree T.

Guess. Finally, A outputs $\mu' \in \{0,1\}$, if $\mu' = \mu$, then A wins game 1.

A's an advantage in successfully winning this game is defined as

$$Adv_A^{DJQ}(\lambda) = |Qr\left[\mu' = \mu\right] - \frac{1}{2}|$$

If $Adv_A^{DJQ}(\lambda)$ is negligible for attacker A in probabilistic polynomial time, the scheme is said to satisfy the indistinguishability of the key-ciphertext security.

Game 2: Trapdoor Indistinguishability.

Suppose A is a polynomial-time attacker trying to break the indistinguishable trapdoor security. Then, challenger B solves the DDH problem by establishing an algorithm, and B obtains the instance $F = (H_1, H_2, f, q, h, b, c, h^{bc})$.

The initial phase: B Run the system to establish the algorithm to output the public parameters.

Stage 1: At this stage, A adaptively performs the following query of polynomial bounded degree.

Key extraction challenge. B runs the key generation algorithm to calculate RL_U and returns the essential RL_U to A.

Trapdoor inquiry: Given a keyword ω, compute the corresponding trapdoor $T\omega$ and return it to A.

Challenge: A submits two challenge keywords, ω_0 and ω_1, to B. B randomly selects $\mu \in \{0,1\}$ and uses $\omega\mu$ to get the trapdoor $U_{\omega\mu}$ and returns it to A.

Stage 2: A Continue to initiate a series of queries as in Phase 1, but cannot ask for information about the challenge keyword.

Guess. Finally, A outputs $\mu\prime \in\{0,1\}$, if $\mu\prime = \mu$, then A wins game 2.

A's an advantage in successfully winning this game is defined as

$$Adv_A^{USB}(\lambda) = |Qr\left[\mu' = \mu\right] - \frac{1}{2}|$$

If $Adv_A^{USB}(\lambda)$ is negligible for attacker A in probabilistic polynomial time, the scheme is said to satisfy trapdoor indistinguishability security.

B. ACCESS STRUCTURE AND ACCESS TREE

Access Structure: Let $\{p_1, p_2, \ldots p_n\}$ be a set of parties, a collection $A \subseteq 2^{\{p_1, p_2, \ldots p_n\}}$ is monotone if $\forall B, C$:*if*$B \epsilon A$ and $B \subseteq C$ then $C \epsilon A$. An access structure is a collection A of non-empty subsets $\{p_1, p_2 \ldots p_n\}$,i.e. $A \subseteq 2^{\{p_1, p_2, \ldots p_n\}} \setminus \{\emptyset\}$, the sets in A are called the authorised sets, and the sets not in A are called non-authorised sets.

Access Tree: Consider a tree denoted as T, which serves as a representation of an access control policy. Within this tree structure, non-leaf nodes hold the role of threshold gates. These gates are characterized by their associated children nodes and a specific threshold value. On the other hand, leaf nodes in the tree correspond to attributes. To describe the gates further, we utilize two parameters: n_x and m_x. Here, $n_x$represents the count of children that a given node 'x' has, while mx signifies the threshold value of that node

There exist three distinct scenarios for the value of m_x when considering a non-leaf node 'x':

- If m_x equals 1, it indicates that the node 'x' functions as an OR gate.
- When n_x is equal to mx, this implies that the node 'x' operates as an AND gate.
- In cases where $1 < m_x < n_x$, the node 'x' takes on the role of a threshold gate.

It's important to note that in the context of leaf nodes, mx is always set to 1. This delineates the structure and behavior of the access control policy represented by the tree T.

Various notations pertaining to the access tree are established as follows:

- To indicate the parent of a node 'x', we use the notation parent(x).
- When referring to a leaf node 'x', the attribute linked with that leaf node is represented as attr(x).
- The label of a node 'x' is denoted by index(x).
- When considering a node 'y' with a total of 'c' children, these child nodes are sequentially numbered from 1 to 'c'.

III. PROPOSED WORK

The blockchain's cloud-assisted attribute-based searchable encryption scheme is divided into five stages: system establishment, data encryption, blockchain security and data search.

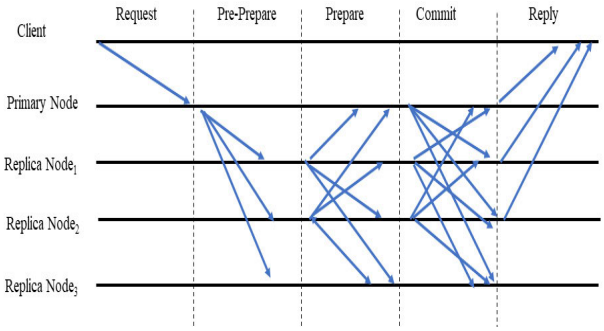


FIGURE 2. Practical byzantine fault tolerance (PBFT) protocol.

Stage 1: Decentralized ABE

In our proposed decentralized Attribute-Based Encryption (ABE) model for enhancing authorization in a distributed environment, we establish a network of autonomous authorization nodes to collectively manage access control and key retrieval. Each authorization node is responsible for a distinct attribute subset, ensuring no single node holds complete authority. A consensus mechanism, based on Practical Byzantine Fault Tolerance (PBFT), is deployed to facilitate collaborative decision-making among nodes for granting access [32]. The nodes collectively verify user attributes and determine key retrieval. To enhance security, keys are fragmented and distributed among nodes using Shamir's Secret Sharing. Load balancing ensures equitable distribution of authorization requests. Regular monitoring and auditing mechanisms are implemented to track node behavior. This model, detailed in our research article, advances decentralized authorization, enhancing fault tolerance, security, and efficient access management in distributed ABE systems.

Practical Byzantine Fault Tolerance (PBFT) operates through three key stages as shown in Figure 2, to ensure consensus in distributed systems. In the Pre-prepare stage, the primary node proposes a transaction and broadcasts it to the network. In the Prepare stage, other nodes acknowledge the proposal's validity and agreement, signaling their intention to commit. Once a certain threshold of prepared messages is reached, the network moves to the Commit stage, during which nodes broadcast their commitment to the proposed transaction. If enough nodes confirm the commitment, consensus is achieved, and the transaction is finalized. PBFT's stages facilitate a robust and efficient consensus protocol, even in the presence of malicious nodes or network failures.

Stage 2: System Establishment.

This stage is divided into two steps: system initialization and key generation.

System initialization (SetUp): In this process, the attribute authorization center executes the algorithm to initialize the system. Input the security parameter (SP) λ, output the system's public parameters (PP) and the data owner's key SK.

1. Generate a bilinear map, e: $H_1 \times H_1 \rightarrow H_2$, where H_1 and H_2 is cyclic multiplicative.

2. Hash functions I. $\{0,1\}^* \rightarrow A_a^*$, I_1:$\{0,1\}^* \rightarrow I_1$.
3. Define the Lagrange coefficient. $\Delta_{i,R}(x) = \prod_{j\in R, j\neq i} \frac{x-j}{i-j}$ R represents a set, i,j$\in A_a^*$.
4. Randomly select α, $\beta \in A_a^*$, and calculate h^{α}, h^{β}, $e(h, h)^{\alpha}$. Return QQ = $\{H^1, H^2, e, h, I, I_1\}$, RL = $\{e(h, h)^{\alpha}, h^{\beta}\}$.

Key Generation (KeyGen): During this procedure, the attribute authorization center runs the algorithm to produce the user's private key for its attribute set R_{uid}.

1) Randomly select s$\in A_q^*$, and calculate

$$RL_{u1} = h^{\frac{\alpha+s}{\beta}}, RL_{u2} = h^{\frac{1}{\beta}}, RL_{u3} = h^s$$

2) For $\forall$ att $\in R_{uid}$, randomly select $s_a \in A_q^*$ and calculate:

$$RL_{ua} = RL_{u3} \times I_1(att)^{s_a} = h^s \times I_1(att)^{s_a}$$

Finally, the user's key

$RL_U = \{RL_{u1}, RL_{u2}, RL_{u3}, \forall att \epsilon R_{uid} : RL_{ua}, RL'_{ua}\}$ is obtained, and RL_U is returned to the user.

Stage 3: Data Encryption

Keyword Encryption (Encrypt): At this stage, the data owner invokes this algorithm to encrypt all keywords, each corresponding to the access tree defining the keyword search authority.

1) Randomly select r$\in A_q^*$ as the secret value, and calculate

$$D_x = e\left(h^{I(x),r}, h\right) e(h, h)^{\alpha r} \text{ and } D'_x = g^{\beta r}$$

2) First, execute the secret sharing algorithm [30] for each node x in the access tree U (including the leaf node t) from the root node t.

To start, choose a polynomial p_x. The specific steps are:

1. For each node in T, make the degree e_x of the polynomial p_x be the node's threshold value l_x -1, that is, $e_x = l_x$-1.
2. Starting from the root node t, define $p_t(0)$=r, and then randomly select e_t points of the polynomial p_t to complete the definition of t_t. For other nodes x, define $p_x(0) = p_{parent(x)}(index(x))$, and randomly select e_x points to complete the definition of p_x.
3. Let X be the set of leaf nodes in U, for the node $\forall x \in X$ in the set X, calculate $D_x = h_{px(0)}$, $D'_x = I_1(attr(x))^{px(0)}$.

Stage 4: Blockchain Security

Finally, the encrypted keyword is J_w =$\{Dw, D'_w, \forall x \in X: D_x, D'_x\}$. The data owner sends the encrypted data file F and the encrypted keyword Jw to the cloud, and the cloud returns the storage address. The data owner will have Jw, and the storage address of Jw. Through marketing Ux, Jw embeds the transaction Ux, signs it, and broadcasts it to the whole blockchain system, and miners record the confirmed transaction on the blockchain. Table 2 depicts the data structure of blockchain. It is made up of a block header and a trade. The block header has the following information: block identifier, block size, Hash, and the date of the preceding block; transactions include the following information: block producer (DO) identity ID_{DO}, block producer's signature DO and Iw, and Address. The transaction $U_x$comprises J_w = (J_w, Address of J_w).

TABLE 1. Notations used in complexity analysis.

Notations	Descriptions
U_p	Time of Pairing Operation
U_e	Time of Exponentiation Operation
T_m	Time of Multiplication Operation
U_h	Time of Hash Operation
T	Time of Multiplication and Inverse Element Operation
M	Least Attribute set satisfying an Access Tree
|S|	Attribute Set of a user
|X|	Leaf node set of an access tree
|N|	Minimum Attribute Set that Satisfy the Access Tree

Stage 5: Data Search: This stage includes two steps: trapdoor generation (Trapdoor) and keyword search (Search).

Trapdoor generation: In this process, the user uses his essential SKU and the keyword ω to be queried to generate the trapdoor U_ω.

1) Randomly select s1 $\in A_{qa}^*$ and calculate

$$U_1 = RL_{u1} X RL_{u2}^{I(\omega)+s1} = h^{\frac{\alpha+s+(\omega)+s1}{\beta}}$$

2) For $\forall att \epsilon R_{uid}$, calculate $U_a = RL_{ua} X h^{r1} = h^{r+r1} XI(att)^{ra}$ and $U'_a = RL'_{ua} A = \frac{e(D_w, U_1)}{G_t}$

Therefore, the trapdoor generated by the keyword ω to be queried is $U_\omega = \{U_1, \forall att \in R_{uid} : U_a, U'_a\}$. Embed the trapdoor $U_\omega$into the transaction, U_y, sign it and broadcast it to the entire blockchain system in the form of the transaction U_y, and the miners record the verified transaction $U_y = U_\omega$ on the blockchain.

A. SECURITY ANALYSIS

In the keyword search stage, according to the trapdoor information, U_ω submitted by the user, the node on the blockchain (also called the searcher P) executes the algorithm to search for the keyword ciphertext. During the whole search process, helpful information about data files and keywords to be searched will not be leaked to the blockchain and cloud servers. The user constructs a transaction $U_y$that contains his trapdoor information. The nodes on the blockchain calculate the central part of the transaction g according to the transaction U_y, embed the searched Iw into the transaction g, and sign it to the whole blockchain network. Then, broadcast the transaction and get the reward in trade U_y simultaneously d. When the transaction g does not appear on the blockchain, the user can choose to construct a new transaction to recover the reward in the previous transaction, U_y.

The nodes on the blockchain verify whether the equation A = D_w holds, where $A = \frac{e(D'_w, U_1)}{G_t}$,

If the equation is established, the search is successful, indicating that the user's attribute set R_{uid} satisfies the access tree embedded in J_w and w and ω are consistent; at this time,

blockchain will store the J_w and address of J_w. It is returned to the data owner. If the equation does not hold, the search fails. There are two situations in which the search fails: the user's attribute set R_{uid} does not satisfy the access tree embedded in J_w, and the algorithm terminates; that is, the user does not have the search authority for the keyword w, or the user has the search authority for the keyword w, but the search found that w and ω are not the same.

x means to visit the node in the tree U, the algorithm runs:

1) If node x is a leaf node, let att=attr(x), that is, att represents the attribute associated with the leaf node x.
   i. If $att \epsilon R_{uid}$, calculate:

$$G_x = \frac{e(U_a, D_x)}{e(U`_a, D`_x)} = \frac{e(h^{s+s_1} \times I_1(att)^{s_a}, h^{p_x(0)})}{e(h^{s_a} \times I_1(att(x))^{p_x(0)})}$$
$$= \frac{e\left(h^{s+s_1}, h^{p_x(0)}\right) e(I_1(att)^{s_a}, h^{p_x(0)})}{e(h^{s_a}, I_1\,(att\,(x))^{p_x(0)})}$$
$$= e(h, h)^{(s+s_1)p_x(0)}$$

   ii. If att $\notin R_{uid}$, define $G_x = \perp$.
2) If node x is a non-leaf node, for all child nodes z of node x, the result after executing the algorithm is denoted as G_z, and all values of $G_z \neq \perp$ are reserved in the set V_x.

a) If $|V_x| <$kx, it means that the attribute set of the child node of node x does not meet the threshold value of this node, then terminate and output $\perp$.

b) If $|V_x| \geq$kx, it means that the attribute set of the child node of node x satisfies the threshold value of this node, then randomly select l_x values of G_z from the set V_x, and calculate the G_x value in combination with the Lagrange coefficient:

$$G_x = \prod_{z \in U_x} G_z^{\Delta_i, R_x(0)} = \prod_{z \in U_x} (e(h, h)^{(s+s_1)p_z(0)})^{\Delta_i, R_x(0)}$$
$$= \prod_{z \in U_x} (e(h, h)^{(s+s_1)} r_{Parent(z)}\,(index\,(z)))^{\Delta_i, R_z(0)}$$
$$= \prod_{z \in U_x} e(h, h)^{(s+s_1)q_x(i)\Delta_i, R_x(0)} = e(h, h)^{(s+s_1)q_x, R_x(0}$$

where i = index (z), $R_x = \{\forall z \in V_x$: index (z)$\}$, Δ_i, R_x represents the Lagrange coefficient.

3) If the user's attribute set R_{uid} satisfies the access tree U, the execution result of the algorithm is expressed as

$$G_t = e(h, h)^{(s+s_1)q_x, R_x(0)} = e(h, h)^{(s+s_1)r}$$

Proof of correctness: Calculate $A = \frac{e(D`_w, U_1)}{G_t}$, verify whether $A = D_w$ is established and if so, return 1

$$D_w = e\left(h^{G(w)r}, h\right) e(h, h)^{\alpha r}$$
$$B = \frac{e(D`_w, U_1)}{G_t} = \frac{e(h^{\beta r}, h^{\frac{\alpha+s+G(\omega)+r_1}{\beta}})}{e(h, h)^{(s+s_1)r}}$$
$$= \frac{e\left(h^r, h^{s+s_1}\right) e(h^r, h^{\alpha+s+G(\omega)+s_1})}{e(h, h)^{(s+s_1)r}}$$
$$= e\left(h^r, h^{\alpha+G(\omega)}\right) = e(h, h)^{\alpha r} e(h^{eG(\omega)}, h)$$

TABLE 2. Data Structure of blockchain.

Chunky			Transaction slip		
Block ID block size	Front Block Hash	Tim esta mp	Block Producer Identity	Previous block signature	Tran sacti on
ID	size	has h	ID_{DO}	δDO	Ux, Uy

Once the data owner receives the storage address of J_w and J_w, the data owner retrieves the required DC address of cloud storage using relative index scheme G. Then G returns the address of DC to cloud server to retrieve and return the corresponding encrypted data file to DU.

IV. PERFORMANCE ANALYSIS

A. COMPARISON OF FUNCTIONAL CHARACTERISTICS

This paper makes a functional comparison with attribute-based encryption schemes [25], [26], [27], [28] in recent years. The access control strategy mainly includes two kinds of access trees and a linear secret sharing scheme. The comparison results are shown in Table2. In addition, Table 3 shows that the proposed method has certain advantages in functional characteristics.

B. THEORETICAL ANALYSIS

In Table 4, U_p represents the time of pairing operation, U_e represents the time of exponentiation operation, Tm represents the time of multiplication operation, and U_h represents the time of Hash operation. Finally, T represents the time of multiplication and inverse element operation.

1) Comparing the amount of calculation.
   In Tables 3 and 4, |S|, |X|, and |N| represent the attribute set of a user, the leaf node set of an access tree and the minimum attribute set that satisfies the access tree, respectively.
2) Comparison of storage capacity
   In Table 5, we use $|H_1|$, $|H_2|$, and $|A_q^*|$ to denote the lengths of elements in H_1, H_2, and A_q^*, respectively.

C. NUMERICAL SIMULATION

We use the bilinear pairing package (pairing-based cryptography library) [30] under the Linux operating system to program in C language and run on a 2.9GHz CPU, 8GBRAM PC. (The elliptic curve base field used is 512b, and the bilinear pairing parameter type is Type-A. The experimental results are shown in Figure 3 with Table 6:

The efficiency of the suggested system is greater than that of the literature [25] and [28] in the key generation stage, trapdoor generation stage, and search stage, it can be

TABLE 3. Comparison of functional characteristics.

Program	Access Control Policy	Searchable	Privacy Protection	Blockchain Technology
Reference [25] scheme	access tree	×	×	×
Reference [27] scheme	Linear Secret Sharing	√	×	×
Reference [28] scheme	access tree	√	√	×
Reference [29] scheme	access tree	√	√	×
Presented Scheme	access tree	√	√	√

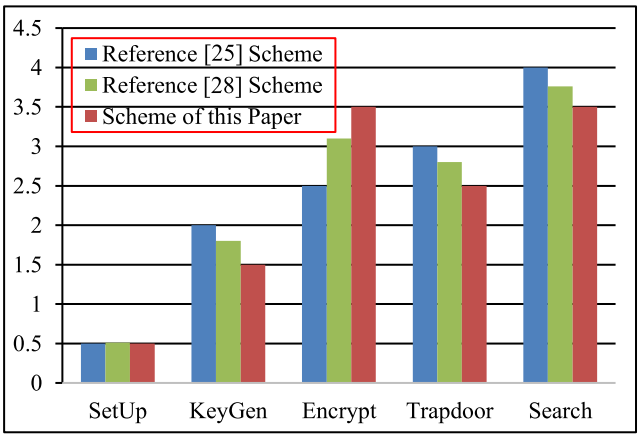


FIGURE 3. The time cost of the algorithm (fix the number of keywords and attributes to 500 and 10) in terms of key generation, trapdoor generation and search stage.

inferred from Figure 3. In terms of system establishment, the two are essentially identical. The efficiency of the suggested approach in the key generation stage, trapdoor generation stage, and search stage is greater than that in the literature, as can be observed from Figure 4 with Tables 7.

The Encrypt algorithm generates an encrypted representation of an indexed keyword, embedding it within an access tree composed of |X| leaf nodes. In Figure 5(a), the time

TABLE 4. Comparison of the calculation amount.

Algorithm	Reference [25] scheme	Reference [28] Scheme	Scheme of this paper
SetUp	$3U_e$	U_p+U_e	U_p+3U_e
Key Gen	$(3\|T\|+1)U_e+(\|T\|+2)U_m+\|T\|U_h+U_{inv}$	$(\|T\|+1)U_e+(\|T\|+2)U_m+2\|T\|)U_h+U_{inv}$	$(2\|T\|+1)U_e+(\|T\|+1)U_m+\|T\|U_h+U_{inv}$
Encrypt	$(2\|V\|+4)U_e+2U_m+(\|V\|+1)U_h$	$(\|V\|+4)U_e+2U_m+(\|V\|+1)U_h$	$U_p+(2\|V\|+3)U_e+3U_m+(\|V\|+1)U_h$
Trapdoor	$(2\|T\|+4)U_e+U_m+U_h$	$(2\|T\|+3)U_e+U_m+(1+\|T\|)U_h$	$(\|T\|+1)U_e+(\|T\|+1)U_m+U_h$
Search	$(2\|M\|+3)U_p+\|M\|U_e+(\|M\|+2)U_m+2U_{inv}$	$(\|M\|+1)U_p+\|M\|U_e+\|M\|U_m+U_{inv}$	$(2\|M\|+1)U_p+\|M\|U_e+(\|M\|+3)U_m+\|M\|U_{inv}$

TABLE 5. Comparison of storage costs.

Algorithm	Reference [25] scheme	Reference [28] Scheme	Scheme of this paper
SetUp	$4\|H_1\|+3\|A\|$	$3\|H_1\|+3\|H_2\|$	$\|H_1\|+\|H_2\|+\|Z*\|$
KeyGen	$(2\|R\|+1)\|H_1\|$	$2(\|R\|+1)\|H_1\|$	$(2\|T\|+2)\|H_1\|$
Encrypt	$(2\|U\|+3)\|H_1\|$	$(\|U\|+3)\|H1\|$	$(2\|U\|+1)\|H_1\|+\|H_2\|$
Trapdoor	$(2\|R\|+3)\|H_1\|$	$(\|R\|+2)\|H_1\|$	$(2\|T\|+1)\|H_1\|$
Search	$(2\|N\|+1)\|H1\|+\|H2\|$	$(2\|M\|+1)\|H_1\|+\|H_2\|$	$(2\|M\|+3)\|H1\|$

required for encrypting a single index keyword in Proposed scheme and [25], [28] demonstrates a linear correlation with

TABLE 6. The time cost of the algorithm (fix the number of keywords and attributes to 500 and 10).

Algorithm	Reference [25] scheme	Reference [28] Scheme	Scheme of this paper
SetUp	0.5	0.51	0.5
KeyGen	2	1.8	1.5
Encrypt	2.5	3.1	3.5
Trapdoor	3	2.8	2.5
Search	4	3.76	3.5

TABLE 7. Comparison of the algorithm running time (fix the number of keywords to 500).

S.No	Reference [25] Scheme	Reference [28] Scheme	Scheme of this Paper
SetUp	0.4	0.42	0.44
KeyGen	0.6	0.64	0.64
Encrypt	1	1.2	1.4
Trapdoor	1.4	1.52	1.8
Search	1.6	1.71	2

TABLE 8. Comparison of the algorithm running time (fix the number of keywords to 500).

Algorithm	Reference [25] Scheme	Reference [28] Scheme	Scheme of this paper
SetUp	3.7	2.8	1.7
KeyGen	3.7	2.4	1.7
Encrypt	3.7	2	2
Trapdoor	4	2.5	2.2
Search	4	2.7	2.2

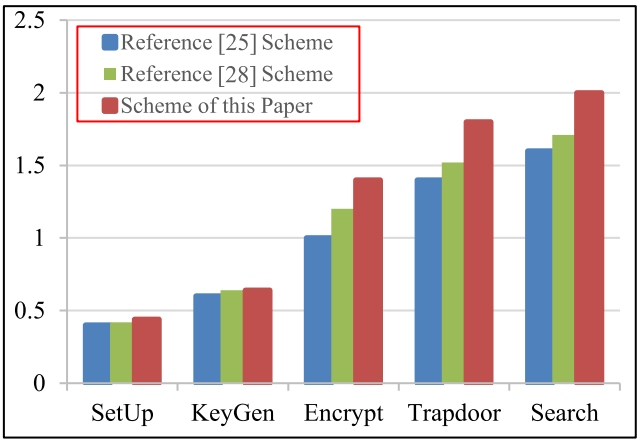
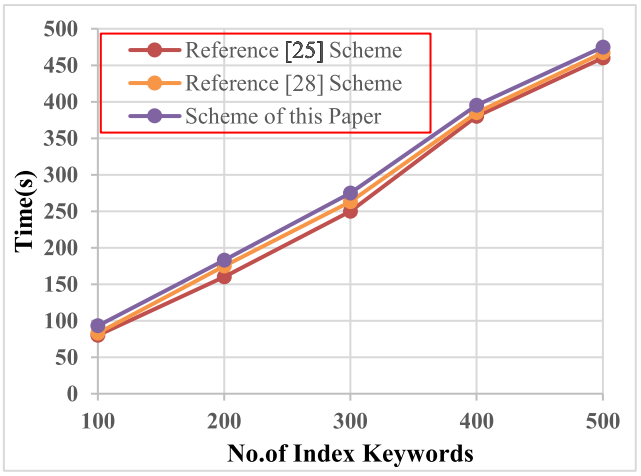
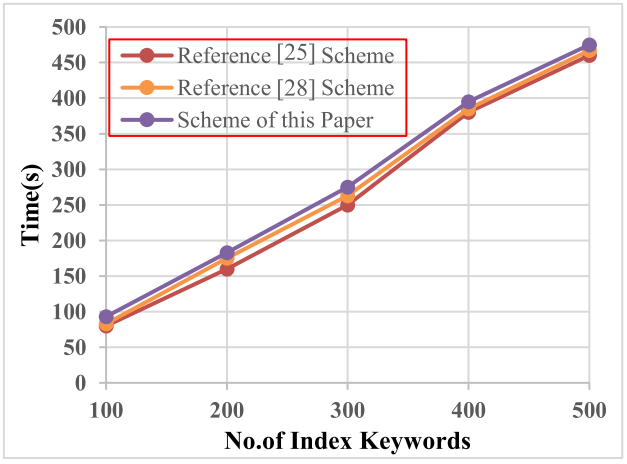


FIGURE 4. Comparison of the algorithm running time (fix the number of keywords to 500).



(a) Time cost of encrypting one index keyword for different number of leaf nodes.



(b) Time cost of encrypting index keywords for different number of index keywords with fixed number of leaf nodes x=8 and fixed number of data files n= 1800.

FIGURE 5. Time cost of index keyword encryptio n.

the quantity of leaf nodes present in the access tree. Moving to Figure 5(b), the complete time expenditure for constructing a secure index in both schemes displays a linear relationship with the count of index keywords extracted from data files. Interestingly, this relationship remains consistent regardless

of the size of the data file collection, provided that the number of leaf nodes in the access trees is held constant.

V. CONCLUSION

This paper proposes a cloud-assisted attribute-based searchable encryption scheme on the blockchain. The system in this paper uses attribute-based encryption technology to enable data owners to perform fine-grained search authorization for data users. Use searchable encryption technology to complete the search of keywords on the blockchain and realize the secure access of data users to encrypted data. During the process, no vital information about keywords and data files will be leaked to the cloud server. We give detailed correctness proofs, performance analyses and security proofs. Numerical experiment results show that the proposed scheme has high efficiency. In future work, we consider combining proxy re-encryption technology to apply it in electronic medical record data sharing to realize data sharing with third-party data users.

REFERENCES

[1] F. Nzanywayingoma and Y. Yang, "Efficient resource management techniques in cloud computing environment: A review and discussion," *Int. J. Comput. Appl.*, vol. 41, no. 3, pp. 165–182, May 2019.

[2] M. Li, S. Yu, N. Cao, and W. Lou, "Authorized private keyword search over encrypted data in cloud computing," in *Proc. 31st Int. Conf. Distrib. Comput. Syst.*, Jun. 2011, pp. 383–392.

[3] C. Bösch, P. Hartel, W. Jonker, and A. Peter, "A survey of provably secure searchable encryption," *ACM Comput. Surv.*, vol. 47, no. 2, pp. 1–51, Jan. 2015.

[4] J. Sun, D. Chen, N. Zhang, G. Xu, M. Tang, X. Nie, and M. Cao, "A privacy-aware and traceable fine-grained data delivery system in cloud-assisted healthcare IIoT," *IEEE Internet Things J.*, vol. 8, no. 12, pp. 10034–10046, Jun. 2021.

[5] J. Gao, H. Yu, X. Zhu, and X. Li, "Blockchain-based digital rights management scheme via multiauthority ciphertext-policy attribute-based encryption and proxy re-encryption," *IEEE Syst. J.*, vol. 15, no. 4, pp. 5233–5244, Dec. 2021.

[6] Y. Hei, J. Liu, H. Feng, D. Li, Y. Liu, and Q. Wu, "Making MA-ABE fully accountable: A blockchain-based approach for secure digital right management," *Comput. Netw.*, vol. 191, May 2021, Art. no. 108029.

[7] H. Yin, J. Zhang, Y. Xiong, L. Ou, F. Li, S. Liao, and K. Li, "CP-ABSE: A ciphertext-policy attribute-based searchable encryption scheme," *IEEE Access*, vol. 7, pp. 5682–5694, 2019.

[8] Q. Chai and G. Gong, "Verifiable symmetric searchable encryption for semi-honest-but-curious cloud servers," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2012, pp. 917–922.

[9] C. Li, M. Dong, J. Li, G. Xu, X.-B. Chen, W. Liu, and K. Ota, "Efficient medical big data management with keyword-searchable encryption in healthchain," *IEEE Syst. J.*, vol. 16, no. 4, pp. 5521–5532, Dec. 2022.

[10] Z. Zhang, J. Zhang, Y. Yuan, and Z. Li, "An expressive fully policy-hidden ciphertext policy attribute-based encryption scheme with credible verification based on blockchain," *IEEE Internet Things J.*, vol. 9, no. 11, pp. 8681–8692, Jun. 2022.

[11] Y. He, H. Wang, Y. Li, K. Huang, V. C. M. Leung, F. R. Yu, and Z. Ming, "An efficient ciphertext-policy attribute-based encryption scheme supporting collaborative decryption with blockchain," *IEEE Internet Things J.*, vol. 9, no. 4, pp. 2722–2733, Feb. 2022.

[12] K. Routray, K. Sethi, B. Mishra, P. Bera, and D. Jena, "CP-ABE with hidden access policy and outsourced decryption for cloud-based EHR applications," in *Information and Communication Technology for Intelligent Systems*, vol. 2. Singapore: Springer, 2021.

[13] S. Mashhadi, "Secure publicly verifiable and proactive secret sharing schemes with general access structure," *Inf. Sci.*, vol. 378, pp. 99–108, Feb. 2017.

[14] P.-C. Chen, T.-H. Kuo, and J.-L. Wu, "A study of the applicability of ideal lattice-based fully homomorphic encryption scheme to Ethereum blockchain," *IEEE Syst. J.*, vol. 15, no. 2, pp. 1528–1539, Jun. 2021.

[15] B. B. Gupta, K.-C. Li, V. C. M. Leung, K. E. Psannis, and S. Yamaguchi, "Blockchain-assisted secure fine-grained searchable encryption for a cloud-based healthcare cyber-physical system," *IEEE/CAA J. Autom. Sinica*, vol. 8, no. 12, pp. 1877–1890, Dec. 2021.

[16] M. Chen, "Discussion on implementation and application of RSA algorithm in smart card operating system," in *Proc. Int. Conf. High Perform. Comput. Commun. (HPCCE)*, Feb. 2022, pp. 1–6.

[17] S. Yaji, K. Bangera, and B. Neelima, "Privacy preserving in blockchain based on partial homomorphic encryption system for ai applications," in *Proc. IEEE 25th Int. Conf. High Perform. Comput. Workshops (HiPCW)*, Dec. 2018, pp. 81–85.

[18] P. P. Nayudu and K. R. Sekhar, "Accountable specific attribute-based encryption scheme for cloud access control," *Int. J. Syst. Assurance Eng. Manage.*, vol. 2022, pp. 1–10, Jul. 2022.

[19] Y. Yang, M. Hu, Y. Cheng, X. Liu, and W. Ma, "Keyword searchable encryption scheme based on blockchain in cloud environment," in *Proc. 3rd Int. Conf. Smart BlockChain (SmartBlock)*, Oct. 2020, pp. 1–4.

[20] Y. Zhang, C. Xu, J. Ni, H. Li, and X. S. Shen, "Blockchain-assisted public-key encryption with keyword search against keyword guessing attacks for cloud storage," *IEEE Trans. Cloud Comput.*, vol. 9, no. 4, pp. 1335–1348, Oct. 2021.

[21] S. Liu, J. Yu, Y. Xiao, Z. Wan, S. Wang, and B. Yan, "BC-SABE: Blockchain-aided searchable attribute-based encryption for cloud-IoT," *IEEE Internet Things J.*, vol. 7, no. 9, pp. 7851–7867, Sep. 2020.

[22] S. Tahir and M. Rajarajan, "Privacy-preserving searchable encryption framework for permissioned blockchain networks," in *Proc. IEEE Int. Conf. Internet Things (iThings) IEEE Green Comput. Commun. (GreenCom) IEEE Cyber, Phys. Social Comput. (CPSCom) IEEE Smart Data (SmartData)*, Jul. 2018, pp. 1628–1633.

[23] S. Jiang, J. Cao, J. A. McCann, Y. Yang, Y. Liu, X. Wang, and Y. Deng, "Privacy-preserving and efficient multi-keyword search over encrypted data on blockchain," in *Proc. IEEE Int. Conf. Blockchain (Blockchain)*, Jul. 2019, pp. 405–410.

[24] X. Yan, X. Yuan, Q. Ye, and Y. Tang, "Blockchain-based searchable encryption scheme with fair payment," *IEEE Access*, vol. 8, pp. 109687–109706, 2020.

[25] B. Chen, D. He, N. Kumar, H. Wang, and K. R. Choo, "A blockchain-based proxy re-encryption with equality test for vehicular communication systems," *IEEE Trans. Netw. Sci. Eng.*, vol. 8, no. 3, pp. 2048–2059, Jul. 2021.

[26] J. Han, Z. Li, J. Liu, H. Wang, M. Xian, Y. Zhang, and Y. Chen, "Attribute-based access control meets blockchain-enabled searchable encryption: A flexible and privacy-preserving framework for multi-user search," *Electronics*, vol. 11, no. 16, p. 2536, Aug. 2022.

[27] S. Wang, X. Wang, and Y. Zhang, "A secure cloud storage framework with access control based on blockchain," *IEEE Access*, vol. 7, pp. 112713–112725, 2019.

[28] R. Awadallah, A. Samsudin, J. S. Teh, and M. Almazrooie, "An integrated architecture for maintaining security in cloud computing based on blockchain," *IEEE Access*, vol. 9, pp. 69513–69526, 2021.

[29] M. Whaiduzzaman, Md. J. N. Mahi, A. Barros, Md. I. Khalil, C. Fidge, and R. Buyya, "BFIM: Performance measurement of a blockchain based hierarchical tree layered fog-IoT microservice architecture," *IEEE Access*, vol. 9, pp. 106655–106674, 2021.

[30] Y. Sun, X. Li, F. Lv, and B. Hu, "Research on logistics information blockchain data query algorithm based on searchable encryption," *IEEE Access*, vol. 9, pp. 20968–20976, 2021.

[31] R. Guo, H. Shi, D. Zheng, C. Jing, C. Zhuang, and Z. Wang, "Flexible and efficient blockchain-based ABE scheme with multi-authority for medical on demand in telemedicine system," *IEEE Access*, vol. 7, pp. 88012–88025, 2019.

[32] S. Alqahtani and M. Demirbas, "Bottlenecks in blockchain consensus protocols," in *Proc. IEEE Int. Conf. Omni-Layer Intell. Syst. (COINS)*, Aug. 2021, pp. 1–8.

[33] S. Liu, J. Yu, L. Chen, and B. Chai, "Blockchain-assisted comprehensive key management in CP-ABE for cloud-stored data," *IEEE Trans. Netw. Service Manage.*, vol 20, no. 2, pp. 1745–1758, Jun. 2023.

G. SUCHARITHA is currently an Associate Professor with the Department of Electronics and Communication Engineering, Institute of Aeronautical Engineering, Hyderabad. She has more than 15 years of teaching experience. She is an expert of teaching in image processing, artificial intelligence, machine and deep learning, computer organization and architecture, microprocessors, and the Internet of Things. She has published more than ten research articles Scopus and SCIE journals, five patents, six book chapters, and one edited book. Her research interests include biomedical image processing, image retrieval, machine learning, deep learning and information security. She received the CIEMA (Springer Conference) ''Outstanding Thesis and Dissertation Award-2022'' in March, 2022.

VEDULA SITHARAMULU received the M.Tech. and Ph.D. degrees. He is currently with the Department of Computer Science and Engineering, School of Technology, GITAM (Deemed to be University), Hyderabad, Telangana, India. He has cumulative triumph record of 22 years of teaching and academic experience in various top ranked engineering colleges across the country. His research interests targeted in Publishing five Patents and published more than 20 journal articles in Scopus, SCI, WOS, and international journals. His research interests include data mining, machine learning, artificial intelligence, and networks. He was awarded the prestigious ''Excellence in Research Award'' and the ''Best Senior Faculty Awards'' during his tenure in academics. Being an academician, he always strives hard to deliver quality teaching practices to his students. At the outset, he has guided many more scholars and still guiding his researchers. He is a driving force for students to build their skill sets and produce good technocrats to the nation.

SACHI NANDAN MOHANTY (Senior Member, IEEE) received the Ph.D. degree from IIT Kanpur, in 2019, and the Ph.D. degree from IIT Kharagpur, India, in 2015, with MHRD scholarship from the Government of India. He has guided nine Ph.D. scholars. He has authored/edited 32 books, published by IEEE-Wiley, Springer, Wiley, CRC Press, NOVA, and DeGruyter. He has published 120 international journals of international repute. His research interests include data mining, big data analysis, cognitive science, fuzzy decision making, brain–computer interface, cognition, and computational intelligence. He received four best paper awards during the Ph.D. at IIT Kharagpur from the International Conference, Beijing, China, and the other at International Conference on Soft Computing Applications organized by IIT Rookee, in 2013. He was awarded the Best Thesis Award first prize by the Computer Society of India, in 2015. He has been elected as a fellow of the Institute of Engineers, European Alliance Innovation (EAI), and Springer, and a Senior Member of IEEE Computer Society Hyderabad Chapter. He is a Reviewer of *Robotics and Autonomous Systems* journal (Elsevier), *Computational and Structural Biotechnology Journal* (Elsevier), *Artificial Intelligence Review* (Springer), and *Spatial Information Research* (Springer).

ANJANNA MATTA received the M.Tech. degree (IMSC) from IIT Madras, India, and the M.Sc. degree in mathematics from NIT Warangal, India. He is currently an Assistant Professor with the Department of Mathematics, Faculty of Science and Technology (ICFAITECH), IFHE, Hyderabad. He has published several articles in various international journals. He is the author of few books and book chapters. He was awarded the Ph.D. from IIT Hyderabad, India.

DEEPA JOSE (Senior Member, IEEE) is currently the Head Research of the KCG College of Technology, Chennai, and a Professor with the Department of Electronics and Communication. She is also a Life-Time Member of IEI. She has more than 16 years of teaching experience. She is an approved Supervisor of Anna University, produced one Ph.D. student and currently guiding eight Ph.D. students. She has Indian patent grant and one international patent grant. She has published more than 60 research papers in journals and international conferences in India and abroad. Her research interests include VLSI for wireless communication, deep learning, biomedical signal processing, IOT for healthcare, GIS initiatives, soft computing, and AI. She has received various funds from AICTE, DST, ICMR, and IEEE. She has won best project awards at IET Fourth Edition of IET CLN Industry Institution Summit 4.0, the Best Project Award at Grand Asia Challenge from La Trobe University, Australia, and the IEEE Yesist 12 (Youth Endeavours for Social Innovation Using Sustainable Technology). She has received external funded projects from AICTE, DST, and IEEE worth U.S. 47.15 lakhs. She was a recipient of the Best Academic Practitioner Award from the IET Chennai. She is the IEEE Women in Engineering Chair.