Blockchain-Aided Privacy-Preserving Medical Data Sharing Scheme for E-Healthcare System

Jingwei Liu, *Member, IEEE,* Yue Fan, Rong Sun, *Member, IEEE,* Lei Liu, *Member, IEEE,* Celimuge Wu, *Senior Member, IEEE,* and Shahid Mumtaz, *Senior Member, IEEE*

***Abstract*—Due to the massive applications of Internet of Things (IoT) and the prevalence of wearable devices, e-healthcare systems are widely deployed in medical institutions. As a significant carrier of medical data, electronic medical record (EMR) is convenient to be stored and retrieved, which greatly simplifies the experience of medical treatment and cuts down the trivial work of paramedics. However, EMRs usually include much sensitive information such as patients' identification numbers or home addresses that may be easily captured by unauthorized doctors and cloud servers. Based on this concern, e-healthcare systems can make use of attribute-based encryption (ABE) to protect private information while achieving fine-grained access control of encrypted EMRs. Whereas, most ABE schemes do not support both policy hiding and keyword search. To address the above issues, we propose an inner product searchable encryption scheme with multi-keyword search (MK-IPSE) based on blockchain to provide full privacy preservation and efficient ciphertext retrieval for EMRs. Inner product encryption (IPE) can not only specify access permissions such that only users with matched attributes can get the target files, but also support access policy hiding. Besides, the proposed scheme combines searchable encryption (SE) and federated blockchain (FB) to implement efficient and stable multi-keyword search. Compared with the existing schemes, MK-IPSE shows better performance on computation and storage. Additionally, security analysis demonstrates that our scheme can resist IND-CKA and collusion attacks.**

***Index Terms*—Medical data sharing, blockchain, attribute-based encryption, inner product encryption, searchable encryption**

I. Introduction

WITH the continuous progresses of informatization in the medical industry, electronic medical records (EMRs) are highly valued and widely used for convenient and superior e-healthcare services [1]. Compared with traditional paper-based medical records, EMRs support more efficient sharing among multiple institutions, more robust traceability, securer data loss prevention and some other remarkable functions. Specifically, medical institutions can provide real-time and long-term disease information, and support profound analysis and personal therapy for patients by sharing EMRs with each other. Based on these benefits, more and more medical institutions replace outdated paper-based medical records with EMRs, which results in the exponential growth of EMRs and creates tremendous computation and storage burden for the existing hospitals. Because of the computing power of cloud services, users are more willing to outsource their medical data to the cloud, which promotes the perfection of the medical information systems.

However, in the new period of Internet of Things (IoT) and artificial intelligence (AI) closely integrated [2], [3], [4], [5], the security and privacy preservation of personal records are facing unprecedented challenges. The biggest concern is data leakage. The medical data uploaded to the cloud often contains sensitive information, such as home addresses and personal disease records that may be maliciously accessed by cloud servers or unauthorized entities. If the data gets out of hand, it reveals something personal about the patient, which seriously hinders the sharing of EMRs. In consequence, how to balance secure privacy protection and effective data sharing becomes a challenging task.

To achieve privacy preservation of EMRs, data owners usually encrypt these files before uploading to cloud servers, but there still exist other various issues. For example, it is difficult for patients and doctors to retrieve the desired medical files from large amounts of ciphertexts. A laborious method is to obtain these ciphertexts locally, decrypt them and then run queries, which require a lot of storage and computation. To avoid these issues, searchable encryption (SE) [6] was firstly presented in 2000 to retrieve encrypted files directly, in which the data owner encrypted messages and keyword indexes, and sent these ciphertexts to the cloud server. Then, according to user's trapdoors that associated with queried keywords and secret keys, cloud server can search the object ciphertext. This scheme protects the user's privacy while dramatically improving the retrieval efficiency. Consequently, more feasible and promising researches about SE were proposed to optimize the initial SE scheme in terms of security and computation performance [7], [8].

Besides efficient retrieval, the fine-grained access control of EMRs should also be considered. In traditional symmetric encryption schemes, data owners must distribute different session keys to different data users. They have to store the session key securely even though they only use it once. This kind of

Jingwei Liu and Rong Sun are with the State Key Laboratory of Integrated Services Networks, Xidian University, Xi'an 710071, China (e-mail: jwliu@mail.xidian.edu.cn, rsun@mail.xidian.edu.cn).

Yue Fan is with the Shaanxi Key Laboratory of Blockchain and Secure Computing, Xidian University, Xi'an, 710071, China (e-mail: fanyue5408@163.com).

Lei Liu is with the Guangzhou Institute of Technology, Xidian University, Guangzhou 510555, China, and also with the Key Laboratory of Embedded System and Service Computing (Tongji University), Ministry of Education, Shanghai 201804, China (e-mail: tianjiaoliulei@163.com).

Celimuge Wu is with the Department of Computer and Network Engineering, The University of Electro-Communications, Tokyo 182-8585, Japan (e-mail: celimuge@uec.ac.jp).

Shahid Mumtaz is with the Department of Applied Informatics, Silesian University of Technology, Akademicka 16, 44-100 Gliwice, Poland, and also with the Engineering Departement, Nottingham Trent University, NG1 4FQ Nottingham, U.K. (e-mail: dr.shahid.mumtaz@ieee.org).

one-to-one communication incurs complex key management. Whereas, ciphertext-policy attribute-based encryption (CP-ABE) schemes were designed for achieving efficient fine-grained access control while protecting the private information of data users [9], [10]. In CP-ABE schemes, attribute sets are embedded in private keys while message ciphertexts are labeled with the access policy that determines who can decrypt the ciphertext, which improves flexibility in setting differential access structures for diverse sets of users.

Inspired by SE and CP-ABE, scholars put forward attribute-based searchable encryption (ABSE) schemes [11], [12] to implement efficient retrieval and secure access control of encrypted data simultaneously. However, in most ABSE schemes, there is a serious flaw that the access policy shown explicitly may expose users' personal information. For example, in medical systems, unauthorized doctors and cloud servers can speculate the patient's intimate disease information ($e.g.$, $sexually\ transmitted\ diseases$, $infectious\ diseases$) according to the access policy of EMRs. Therefore, a secure ABSE scheme should support hidden access policy to avoid leaking personal information and improve the security of the system. Many schemes with hidden access policy have been proposed [13], [14]. Unfortunately, in these schemes, every attribute and keyword is encrypted separately, which leads to tremendous computation and storage cost.

Inner product encryption (IPE) is developed from ABE originally and achieves full access policy hiding in an ingenious way. It converts access policies and attribute sets into vectors for embedding in ciphertexts and secret keys respectively. The process of verifying whether the attribute sets satisfy the access structure is translated into determining whether the inner product of attribute vectors and access policy vectors is zero or not, which protects the privacy of data users [15], [16].

After increasingly profound research, people find that blockchain could be used in the medical field. For example, the EMRs stored in blockchain are immutable, transparent and traceable which meet the security and trust requirements in complex medical scenarios. In addition, blockchain can help doctors and patients manage their access privileges, and record the usage and delivery of medical data, providing reliable data tracking for users. Crucially, for burdensome e-healthcare services, blockchain can be integrated with other internet technologies (such as IoT, AI, wearable devices, and so on) to improve the adaptation and responsiveness of e-healthcare systems.

In this paper, by combining IPE and ABSE, we present a blockchain-aided medical data sharing scheme based on inner product searchable encryption with multi-keyword search (MK-IPSE). The primary contributions are shown below:

- *Data access control and efficient retrieval.* In our scheme, trusted authority (TA) generates private keys for doctors with their attributes. Patients encrypt their EMRs and keyword indexes according to the specified access policy. Only when keywords in trapdoors satisfy keywords in ciphertexts and submitted attributes pass through the access policy, can doctors get EMRs successfully.
- *Hidden access policy.* Firstly, the access policy and attributes are converted into vectors. And then, these two kinds of vectors are placed on the exponent to guarantee the security of access policy due to the discrete logarithm assumption. Therefore, by figuring up the inner product with these two kinds of vectors, decryption will recover 0 if attributes satisfy the access structure. Otherwise, it only recovers a random number but no sensitive information.
- *Blockchain-aided medical system.* The local servers in medical institutions are viewed as the trusted nodes and compose a federated blockchain. In MK-IPSE scheme, we provide the keyword retrieval in the blockchain while storing the message ciphertexts to the cloud server. This kind of collaboration pattern reduces the computation and storage cost of cloud servers. Furthermore, the inherent immutability, transparency and traceability make blockchain appropriate to ensure the security and authenticity of medical records.
- *Adaptive multi-keyword retrieval.* A doctor can submit queried keywords according to his/her interests. If the keywords satisfy $user's\ queried\ keywords \subseteq keywords\ in\ index$, it means that the corresponding ciphertext is found successfully. For example, suppose there are three EMRs that the doctor has access to, the keyword indexes of the EMRs are $\{2020, cardiopathy\}$, $\{2021, cardiopathy\}$ and $\{2022, cardiopathy\}$ respectively. If the doctor submits $\{cardiopathy\}$, s/he will get all three EMRs. If s/he submits more specific keywords such as $\{2022, cardiopathy\}$, s/he will obtain the third EMR. Obviously, the more keywords are submitted, the fewer but more accurate documents the doctor will get. So, doctors can realize adaptive multi-keyword retrieval by changing the number of keywords readily.

The rest sections are arranged as follows. In section II, we summarize the related work about searchable encryption. The cryptographic backgrounds and preliminaries are given in section III. In section IV, we introduce the system model, formal definition, and security model of the proposed scheme. The construction of MK-IPSE is introduced specifically in section V. In section VI, we analyze the security under IND-CKA and collusion attacks, and compare the performance with relevant schemes in terms of computation and storage cost. Finally, we conclude this work in section VII.

II. Related work

Firstly, we summarize the relevant work about the EMRs sharing and searchable encryption in terms of implementing secure fine-grained access control and access policy hiding. Besides, searchable encryption schemes concerned with blockchain are also given at last.

EMRs play increasingly vital roles in our daily life due to the convenience of the online medical service [17], [18]. Therefore, how to share EMRs securely and effectively becomes a challenging research focus. Yang et al. [19] put forward a lightweight mobile health system, which achieved fine-grained access control of encrypted EMRs, the tracking of malicious users and user revocation. Xu et al. [20] utilized attribute-based encryption and searchable encryption to realize flexible search control, greatly facilitating the access and sharing of EMRs. Besides, it also provided efficient access policy assignment. In [21], the authors presented a healthcare IoT system connecting attribute-based encryption and edge computing, which supported expressive access control and lightweight decryption cost. Recently, Xu et al. [22] implemented EMRs' integrity and confidentiality with the advantages of blockchain and attribute-based encryption respectively. In addition, this scheme allowed the authority center to revoke dishonest users who revealed sensitive information deliberately.

Searchable encryption (SE) consists of symmetric searchable encryption (SSE) and public key encryption with keyword search (PEKS) that were firstly presented by Song et al. [6] and Boneh et.al [23] respectively to provide efficient keyword search of ciphertexts. There is no available information in encrypted data can be captured by the unauthorized entities unless they have permissions. SSE is mainly suitable for storing and retrieving personal data while PEKS often focuses on multi-user data sharing. Based on SSE and PEKS, a series of schemes were presented to support more comprehensive functionalities, such as single keyword search [24], [25], multi-keyword search [26], [27], data updatable [28], and so on.

Next, to implement access control in a promising way, ABSE was proposed and had been applied in many schemes. Liu et al. [29] utilized ABSE to construct verifiable search scheme, in which the trusted authority was required to help generate the search trapdoors every time, resulting in additional computation burden. For improving computation efficiency, some scholars adopted an online/offline mechanism in ABSE to mitigate constrained computing power [30], [31]. The scheme in [30] used the outsourcing ABE to greatly reduce the computation overhead. Besides, it achieved data security, unlinkability of trapdoor and search controllability at the same time. The scheme [31] made use of cloud-edge coordination for relieving the computation and storage overhead. It stored the ciphertexts in the cloud while uploading the keyword indexes to the edge node. Moreover, it could realize multi-keyword search and resist keyword guessing attacks.

However, ABSE has a server limitation that access policies may leak certain personal information of data users. In order to strengthen users' privacy, Koo et al. [32] proposed an ABSE scheme that realized access policy hiding, which supported fast ciphertext search and rich expressions of access policy. Meanwhile, this scheme also achieved the anonymity of data users. Shi et al. [33] masked the access policy by linear secret sharing, which was a lowcost and promising method to protect the access policy. However, in this scheme, data users were required to interact with trusted authorities while generating search trapdoors, causing a number of bilinear pair computation cost. After that, Wang et al. [34] designed a searchable CP-ABE scheme, which supported secure access control and hidden access policy. Besides, the scheme was multi-value-independent with constant storage overhead. Miao et al. [35] designed a CP-ABKS scheme with privacy preservation in shared multi-owner setting, which achieved secure access policy hiding and tracing of malicious data users. Moreover, it could also resist keyword guessing attacks.

Blockchain, proposed by Nakamoto in 2008 [36], is an emerging application mode of computer technology aiming at realizing Bitcoin trading at first. So far, there have been many proposals for combining blockchain with searchable encryption. Liu et al. [37] proposed a blockchain-aided ABSE scheme to achieve private key management and user revocation efficiently by replacing traditional centralized servers with decentralized blockchain systems. Yang et al. [38] presented a searchable EMR sharing scheme with blockchain. It provided efficient permission control of medical data via ABE and authentication of EMRs through attribute signature to verify the reality of EMRs, which consumed too much bandwidth and communication cost. Niu et al. [39] designed a medical data sharing scheme based on permissioned blockchain, in which the multi-keyword search was achieved by polynomial equation.

III. Preliminaries

We introduce cryptographic backgrounds as well as preliminaries concerned with MK-IPSE in this section.

A. Bilinear Map

Definition 1 (Bilinear Map): Let G_1 and G_T be multiplicative cyclic groups of prime order p, and g is a generator of group G_1. The bilinear map $e : G_1 \times G_1 \rightarrow G_T$ meets the following properties:

- Bilinearity: $e(u^a, v^b) = e(u, v)^{ab}$ for all $u, v \in G_1$ and $a, b \in Z_p$;
- Non-degeneracy: $e(g, g) \neq 1$;
- Computability: Given $u, v \in G_1$, $e(u, v)$ is efficiently computable .

B. Discrete Logarithm (DL) Assumption

Definition 2 (DL Assumption): Let G be a multiplicative cyclic group of prime order p, g is a generator of group G. Given $(g, g^a) \in G$, the DL assumption holds if there is no probabilistic polynomial time (PPT) adversary $\mathcal{A}$ in computing $a \in Z_p^*$ with a non-negligible advantage ε.

C. Decisional Bilinear Diffie-Hellman (DBDH) Assumption

Definition 3 (DBDH Assumption): Let challenger $\mathcal{C}$ randomly select $a, b, c, z \in Z_p^*$, g is a generator of group G. The DBDH assumption holds if there is no PPT algorithm $\mathcal{B}$ that can distinguish $(A = g^a, B = g^b, C = g^c, e(g, g)^{abc})$ and $(A = g^a, B = g^b, C = g^c, e(g, g)^z)$ with a non-negligible advantage ε.

D. Inner Product Encryption

We introduce the syntactic definition of IPE at first. Besides, we also illustrate how to generate the access policy vector and the attribute vector that are associated with IPE.

IPE is a prospective algorithm to check access rights of data users and support access policy hiding simultaneously. In this article, the ciphertext is connected with access policy vector $\boldsymbol{x}$ while data user's secret key is labeled with attribute vector $\boldsymbol{y}$. When attributes match the access policy, the message will be recovered with $\langle \boldsymbol{x}, \boldsymbol{y} \rangle = 0$.

Definition 4 (IPE): A standard inner product encryption scheme is an algorithm as below.

1) **Setup**$(\lambda, l) \rightarrow msk, mpk$: It takes as input a security parameter λ and the vector dimension l, and outputs master secret keys msk and master public keys mpk.
2) **KeyGen**$(\boldsymbol{y}, msk) \rightarrow sk_{\boldsymbol{y}}$: It takes as input a vector $\boldsymbol{y} = (y_1, y_2, \ldots, y_l)$ and master secret keys msk, and outputs user's secret key $sk_{\boldsymbol{y}}$.
3) **Enc**$(\boldsymbol{x}, m, mpk) \rightarrow C$: It takes as input a vector $\boldsymbol{x} = (x_1, x_2, \ldots, x_l)$, message m and master public keys mpk, outputs the ciphertext C.
4) **Dec**$(sk_{\boldsymbol{y}}, C) \rightarrow m$: It takes as input user's secret key $sk_{\mathbf{y}}$ and ciphertext C, outputs m if satisfying $\langle \boldsymbol{x}, \boldsymbol{y} \rangle = 0$.

In this paper, we construct access policy via AND-Gates, which is of significance in practical applications.

Definition 5 (Vector Generation): Given an attribute set $\boldsymbol{U} = \{U_1, \ldots, U_n\}$ of a system, where $U_i \in \boldsymbol{U}(i \in [1, n])$.

1) For each access policy that consists of an attribute subset $\boldsymbol{S}$, we translate it into a vector $\boldsymbol{x}$ with dimension $n+1$, where $\boldsymbol{S} \subseteq \boldsymbol{U}$.

   The first n components are generated as follows, where $\xleftarrow{R}$ denotes random selection and $1 \leq i \leq n$.

$$x_i = \begin{cases} r_i \xleftarrow{R} {Z_p}^*, & U_i \in \boldsymbol{S} \\ 0, & U_i \notin \boldsymbol{S} \end{cases} \tag{1}$$

   The last component x_{n+1} is generated as follows, where p is the order of Z_p.

$$x_{n+1} = -\sum_{i=1}^{n} x_i (modp) \tag{2}$$

2) For the attribute set $\boldsymbol{R}$ that the user submitted, we translate it into a vector $\boldsymbol{y}$ with dimension $n+1$, where $\boldsymbol{R} \subseteq \boldsymbol{U}$.

   The first n components are generated by the Eq.3, where $1 \leq i \leq n$.

$$y_i = \begin{cases} 1, & U_i \in \boldsymbol{R} \\ 0, & U_i \notin \boldsymbol{R} \end{cases} \tag{3}$$

   The last component $y_{n+1} = 1$.

Remarks: For $\boldsymbol{x}$, we randomly select values for every attribute in $\boldsymbol{S}$, and regard them as the first n components of $\boldsymbol{x}$. The last component is generated by summing up negative values of these random numbers. As for $\boldsymbol{y}$, we set the first n corresponding bits as 1 if $U_i \in \boldsymbol{R}$, otherwise, set 0. The last number $y_{n+1} = 1$.

E. Blockchain Technology

Blockchain is a decentralized distributed ledger in which the verified transactions and events are stored and linked chronologically. The trusted and semi-trusted nodes participating in the blockchain system jointly maintain a growing chain that will gradually form an efficient and complete transaction system through consensus protocols. Because of the cryptographic researches applied in blockchains such as hash, digital certificate and signature, each transaction recorded by the node can not be retroactively altered. Therefore, blockchain inherently possesses the advantages of decentralization, trustworthiness, traceability and unforgeability, which provides secure and stable transactions for data users.

After continuous development, blockchain has evolved many branches. According to the degree of decentralization, blockchain is classified into three categories:

1) Public blockchain: In public blockchains, any nodes can get access to the blockchain network, send transactions, and validate blocks. Typically, this kind of network tends to give incentives to encourage users who generate and validate the blocks. The most well-known products of public blockchain are Bitcoin and Ethereum.
2) Federated blockchain: It operates under multiple authorities instead of a highly trusted node like the private blockchain. The authority nodes are pre-selected from all the organizations in the network. In federated blockchains, it is not necessary to pursue credibility. Therefore, there are many entities can join the blockchain network, which brings more frequent business and creates a vibrant industrial chain.
3) Private blockchain: It is maintained by a private organization or individual who regulates the permissions for mining process and consensus mechanism. Only people who have been granted privilege can participate in the network and execute the transaction procedure. Private blockchain is best suitable for managing the data in private institutions.

In order to share EMRs in different medical institutions, our scheme incorporates inner product searchable encryption into federated blockchain, in which the hospital local servers serve as consensus nodes to retrieve data. The traditional EMRs sharing schemes based on searchable encryption take advantage of centralized cloud server, which is liable to form data islands. It is only convenient for hospitals to operate EMRs managed by themselves, hindering the interoperate and sharing of EMRs with each other. Besides, the outsourced cloud servers are usually not completely trusted, they may return ciphertexts randomly or output the results dealt with previously to economize the computational resource. Fortunately, the inherent openness, immutability and consensus trust mechanism make federated blockchain appropriate to solve above problems and ensure the security and authenticity of retrieval process.

IV. System Framework

We describe the system model, formal definition of MK-IPSE and security model respectively in this part.

A. System Model

All participants involved in our system mainly include: trusted authority, patients, doctors, blockchain and the cloud server. Their tasks are introduced respectively in detail below.

- *Trusted Authority (TA).* TA is a third party supposed to be totally trusted. It generates master secret keys and master public keys, manages doctors' attribute sets and constructs secret keys for them.

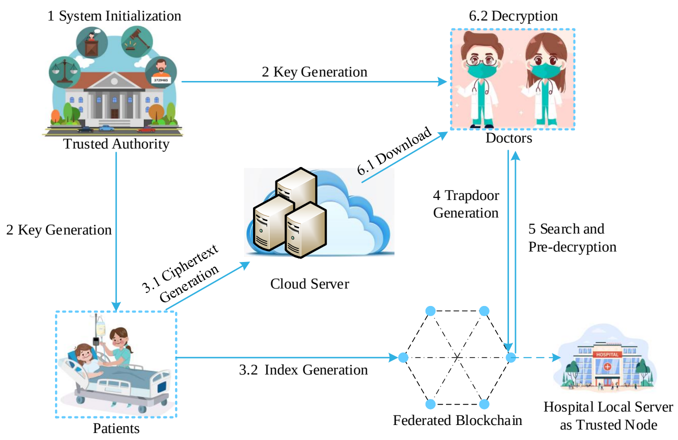


Fig. 1. System model.

- *Patients*. EMRs usually contain sensitive information. Therefore, patients would like to share these records in a secure and efficient way. To this end, patients encrypt EMRs with defined access rights and store ciphertexts in the cloud server.
- *Doctors*. Doctors request their secret keys associated with attributes from TA. In addition, doctors can retrieve the ciphertexts with their interests if the queried keywords match the keyword indexes and their attributes pass through the access structure.
- *Cloud Server (CS)*. CS is responsible for storing the encrypted EMRs. After receiving the encrypted EMR from a patient, CS generates its uniform resource location (URL) which is used to access and download the related ciphertext, and returns it to the patient. A doctor can obtain the corresponding ciphertext from CS when s/he submits the URL.
- *Federated Blockchain (FB)*. FB consists of hospital local servers (HLS) that serve as consensus nodes. In FB, the keyword indexes and the URL of ciphertexts are stored and locked. If a doctor submits his/her trapdoors related to the queried keywords and attributes, FB calls smart contract to search the corresponding ciphertext.

MK-IPSE realizes secure access permission control and efficient retrieval of encrypted EMRs. The system model is shown intuitively as depicted in Fig. 1.

In our scheme, TA is viewed as fully trusted. It is responsible for the registration and certification of every patient and doctor. Patients are viewed as trusted. They will encrypt EMRs with defined access rights and store ciphertexts in the cloud server honestly following the established mechanism. Doctors are untrusted as data users. They may attempt to access the unauthorized EMRs and even conspire with others whose attributes do not satisfy the access policy to get patients' sensitive information. CS is honest but curious who stores the ciphertexts rigorously but tries to infer the medical records with some potential passive aggressions. The federated blockchain in our system uses practical Byzantine fault tolerance (PBFT) as the consensus protocol. Therefore, if there are N consensus nodes in the blockchain, the number of malicious nodes that may tamper with information f should satisfy $3f+1 \leq N$, which indicates that the authorized doctor can get the correct results even if there are faulty nodes and malicious nodes. The rest nodes are honest but curious, that is to say, they follow the smart contract to execute ciphertext search honestly but want to learn the private information from what they store and compute.

B. Formal Definition of MK-IPSE

The proposed MK-IPSE is a tuple of PPT algorithms as follows.

1) **Setup**$(1^{\lambda}) \rightarrow (msk, pp)$: Given the security parameter λ and system attribute set $\boldsymbol{U}$, TA outputs master secret keys msk and public parameters pp.
2) **KeyGen**$(msk, pp, \boldsymbol{R}) \rightarrow (SK, \boldsymbol{y})$: Given master secret keys msk, public parameters pp, and the doctor's attribute set $\boldsymbol{R}$, TA outputs the doctor's secret key SK and attribute vector $\boldsymbol{y}$.
3) **Enc**$(pp, W, m, P) \rightarrow (CT_m, CT_{\omega})$: Given public parameters pp, keyword index set W, message m and access policy P, the patient firstly generates vector $\boldsymbol{x}$ by access policy P, and then runs the algorithm to output the ciphertexts $CT = \{CT_m, CT_{\omega}\}$ of message m and keyword index set W. Finally, the patient sends CT_m to the CS, and uploads CT_{ω} and URL to FB.
4) **Trapdoor**$(SK, W') \rightarrow TK$: The doctor generates search trapdoor TK by his/her secret key SK and the queried keyword set W'. Then the doctor sends TK to FB.
5) **Search**$(CT, TK) \rightarrow (CT_m, B)$: Given the ciphertext CT_{ω} and search trapdoor TK, HLS computes the intermediate parameter B to verify the doctor's attributes. Then, if the queried keyword set W' matches the index keyword set W, HLS will find the URL of ciphertext CT_m on FB and return the URL and B to the doctor.
6) **Dec**$(CT_m, B, SK) \rightarrow m)$: The doctor can download the corresponding CT_m with URL. Given the message ciphertext CT_m and the intermediate parameter B, the doctor can decrypt the ciphertext CT_m and output the message m successfully by his/her secret key SK.

C. Security Model

MK-IPSE includes five entities, namely trusted authority, patients, doctors in medical institutions, blockchain and the cloud server. Here, the trusted authority is fully trusted while doctors are semi-trusted. The blockchain and cloud server are honest but curious. In this case, the proposed scheme can realize the indistinguishability under chosen keyword attack. The security model of MK-IPSE is described through an interactive game between an attacker $\mathcal{A}$ and a challenger $\mathcal{C}$. The game is shown concretely as follows:

Definition 6 (The security of indistinguishability under chosen keyword attack IND-CKA): It is claimed that MK-IPSE can resist IND-CKA if there is no probabilistic polynomial time adversary $\mathcal{A}$ that wins the game with a non-negligible advantage ε.

- ***Setup:*** The setup algorithm is executed by $\mathcal{C}$. $\mathcal{C}$ generates master secret key msk and public parameters pp, and sends pp to $\mathcal{A}$.

- ***Phase1:*** $\mathcal{A}$ issues a series of secret key queries that concerned with attribute sets $\boldsymbol{R}_1, \boldsymbol{R}_2, \ldots, \boldsymbol{R}_n$. According to these attribute sets, $\mathcal{C}$ generates the corresponding attribute vectors $\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_n$, calls the **KeyGen** algorithm to construct secret keys $SK_1, SK_2, \ldots, SK_n$, and then sends these secret keys to $\mathcal{A}$.
- ***Challenge:*** $\mathcal{A}$ selects two keyword sets W_0, W_1 and a challenging access policy P^* that all attribute sets in ***Phase1*** do not satisfy. Then, $\mathcal{A}$ sends W_0, W_1 and P^* to $\mathcal{C}$. $\mathcal{C}$ randomly selects an element $b \in \{0, 1\}$, executes **Enc** algorithm to generate ciphertext CT of message m and index keyword set W_b, and returns CT to $\mathcal{A}$.
- ***Phase2:*** Similar to ***Phase1***, $\mathcal{A}$ continues to choose some attribute sets $\boldsymbol{R}_{n+1}, \boldsymbol{R}_{n+2}, \ldots, \boldsymbol{R}_q$ and request for the secret keys of these attribute sets, where q is the number of secret key queries and the attribute sets $\boldsymbol{R}_{n+1}, \boldsymbol{R}_{n+2}, \ldots, \boldsymbol{R}_q$ still do not satisfy the access policy P^*.
- ***Guess:*** $\mathcal{A}$ outputs the guess $b' \in \{0, 1\}$. It will win the game if $b = b'$.

We define the advantage of $\mathcal{A}$ winning the game as follows:

$$Adv_{\mathcal{A}} = |\Pr[b' = b] - 1/2|.$$

V. The Proposed Scheme

In this part, we specifically introduce the construction of MK-IPSE that supports secure data access control, hidden access policy, along with efficient and adaptive multi-keyword retrieval. MK-IPSE is composed by the following stages: system initialization, secret key generation, ciphertext generation, trapdoor generation, ciphertext search and pre-decryption, and decryption. In addition, the main steps of MK-IPSE are shown in Fig. 2.

A. Construction of MK-IPSE System

1) **Setup**(1^λ). The algorithm is executed by TA that sets the security parameter λ and chooses the system attribute set $\boldsymbol{U} = (U_1, \ldots, U_n)$. The consensus nodes are $\{CN_1, CN_2, \ldots, CN_N\}$. Let G_1, G_T be multiplicative cyclic groups of prime order p, g is a generator of group G_1 and e denotes the bilinear map $e : G_1 \times G_1 \rightarrow G_T$. TA selects a hash function $H : \{0, 1\}^* \rightarrow Z_p^*$ and random elements $\alpha, \beta, s_1, s_2, \ldots, s_{n+1} \in Z_p^*$, computes $h_i = g^{s_i} (i \in [1, n+1])$, $h = g^\beta$, and generates master secret key msk and public parameter pp.

$$msk = (\boldsymbol{s} = (s_1, s_2, \ldots, s_{n+1}), \alpha, \beta);$$
$$pp = (G_1, G_T, p, g, e, e(g, g)^\alpha, H, h, h_i, \boldsymbol{U}).$$

2) **KeyGen**$(msk, pp, \boldsymbol{R})$. Given a doctor's attribute set $\boldsymbol{R}$, where $\boldsymbol{R} \subseteq \boldsymbol{U}$, TA generates the attribute vector $\boldsymbol{y} = (y_1, y_2, \ldots, y_{n+1})$, randomly selects $\theta, \mu \in Z_p^*$ and computes $sk_1 = g^{\frac{\alpha+\mu}{\beta}}, sk_2 = g^{\frac{1}{\beta}}, sk_3 = g^\theta, sk_{\boldsymbol{y}} = g^{\frac{\mu - \theta\langle \boldsymbol{s}, \boldsymbol{y} \rangle}{\beta}}$. Finally, TA returns $SK = \{sk_1, sk_2, sk_3, sk_{\boldsymbol{y}}\}$ and $\boldsymbol{y}$ to the doctor.

3) **Enc**(pp, W, m, P). This algorithm is run by the patient who wants to share his/her medical records with access control.

   - Firstly, the patient generates the access policy vector $\boldsymbol{x} = (x_1, x_2, \ldots, x_{n+1})$ according to the access policy P, where the access policy P is composed of an attribute subset $\boldsymbol{S}$ and $\boldsymbol{S} \subseteq \boldsymbol{U}$.
   - Secondly, the patient extracts the keyword index $W = (w_1, \ldots, w_d)$ from his/her EMR m. Then, s/he chooses a random element $t \in Z_p^*$, and encrypts the EMR m and the keyword index set W as follows:

   $$ct^* = h^t,\ ct_m = m \cdot e(g, g)^{\alpha t},$$
   $$ct_{w_j} = e(g, g)^{\alpha t} \cdot e(g^{H(w_j)}, g^t)(j \in [1, d]),$$
   $$ct_i = h_i^t \cdot g^{x_i} (i \in [1, n+1]).$$

   - Finally, the patient sends CT_m to the CS and gets the returned URL, and then uploads CT_ω and the corresponding URL to FB, where

   $$CT_m = \{ct^*, ct_m\},$$
   $$CT_\omega = \{ct^*, ct_{w_j}(j \in [1, d]), ct_i(i \in [1, n+1])\}.$$

4) **Trapdoor**(SK, W'). According to SK and the queried keyword set $W' = (w'_1, ..., w'_{d'})$, the doctor computes as follows:

$$T_0 = {sk_2}^{\sum_{j=1}^{d'} H(w'_j)} = g^{\frac{\sum_{j=1}^{d'} H(w'_j)}{\beta}},$$
$$T^* = {sk_1}^{d'} \cdot T_0 = g^{\frac{\alpha d' + \mu d' + \sum_{j=1}^{d'} H(w'_j)}{\beta}},$$
$$T_i = {sk_3}^{y_i} = g^{\theta y_i} (i \in [1, n+1]).$$

Then the doctor sends the trapdoor $TK = \{T^*, T_i(i \in [1, n+1]), sk_{\boldsymbol{y}}, d'\}$ to FB.

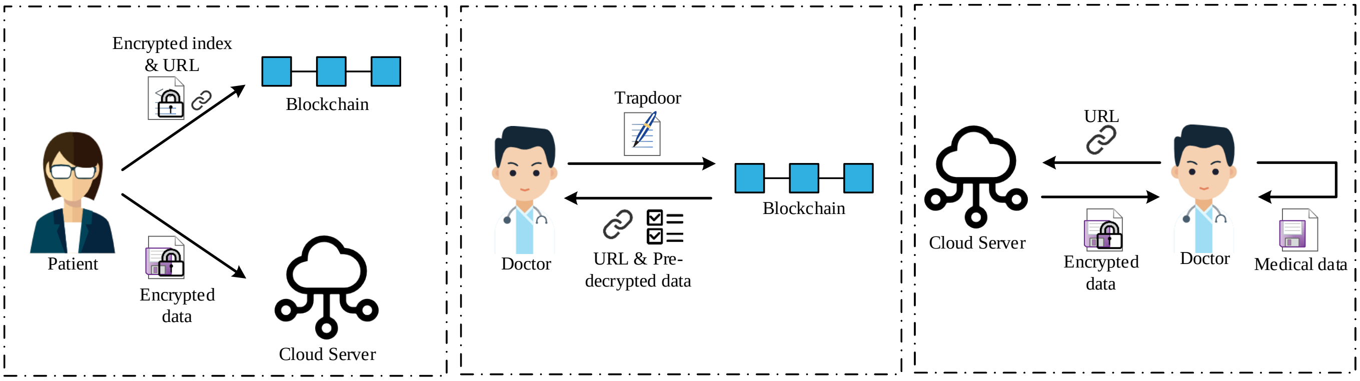
5) **Search**(CT_ω, TK). Once receiving the query request from a doctor, each node on FB automatically executes the predefined algorithm in the smart contract to search the target ciphertext of EMRs and pre-decrypt the ciphertext, which greatly reduces the computation burden of the doctor. The details of the smart contract are shown in Algorithm 1.

Algorithm 1 Ciphertext Retrieval and Pre-decryption

```
Input: $[\{CT_{w_1}, URL_1\}, \{CT_{w_2}, URL_2\}, \{CT_{w_3}, URL_3\}, \ldots]$,
    $[TK]$
$CN_i$ computes $B$ by executing attribute-policy matching
for i=1 to $C_d^{d'}$ do
    $CN_i$ selects $d'$ keywords from keyword index
    $CN_i$ computes $D$ and $D'$ with keyword query
    if $D = D'$ then
        $result = \{B, URL\}$
        break
    end if
end for
```

The specific processes of attribute-policy matching and keyword query are described below.

- *Attribute-Policy Matching:* the consensus node computes the intermediate parameter B according to CT_ω and TK that submitted by the doctor as



(a) **Encryption** stage (b) **Search** stage (c) **Decryption** stage

Fig. 2. Main steps of MK-IPSE.

follows.

$$\begin{aligned} B =& e(ct^*, sk_{\boldsymbol{y}}) \cdot \prod_{i=1}^{n+1} e(T_i, ct_i) \\ =& e(h^t, g^{\frac{\mu-\theta\langle \boldsymbol{s},\boldsymbol{y}\rangle}{\beta}}) \cdot \prod_{i=1}^{n+1} e(g^{\theta y_i}, h_i^t \cdot g^{x_i}) \\ =& e(g,g)^{t\mu - t\theta\langle \boldsymbol{s},\boldsymbol{y}\rangle} \cdot \prod_{i=1}^{n+1} e(g,g)^{\theta t s_i y_i + \theta x_i y_i} \\ =& e(g,g)^{t\mu - t\theta\langle \boldsymbol{s},\boldsymbol{y}\rangle} \cdot e(g,g)^{\theta t\langle \boldsymbol{s},\boldsymbol{y}\rangle + \theta\langle \boldsymbol{x},\boldsymbol{y}\rangle} \\ =& e(g,g)^{t\mu + \theta\langle \boldsymbol{x},\boldsymbol{y}\rangle} \end{aligned}$$

If the doctor's attributes satisfy access rights, $B = e(g,g)^{\mu t}$ because $\langle \boldsymbol{x}, \boldsymbol{y}\rangle = 0$.

- *Keywords Query:* the consensus node checks if the queried keyword set W' matches the keyword index set W or not. The trapdoor contains d' keywords and the keyword index has d keywords, where $d' < d$. Randomly select d' keywords from the keyword index, compute $D = \frac{e(ct^*,T^*)}{B^{d'}}$, $D' = \prod_{j=1}^{d'} ct_{w_j}$, and verify whether the equation $D = D'$ holds.

$$\begin{aligned} D =& \frac{e(ct^*, T^*)}{B^{d'}} = \frac{e(g^{\beta t}, g^{\frac{\alpha d' + \mu d' + \sum_{j=1}^{d'} H(w'_j)}{\beta}})}{e(g,g)^{\mu t d'}} \\ =& e(g,g)^{\alpha t d' + \sum_{j=1}^{d'} H(w'_j) t}, \\ D' =& \prod_{j=1}^{d'} ct_{w_j} = \prod_{j=1}^{d'} e\left(g^{H(w_j)}, g^t\right) \cdot e\left(g,g\right)^{\alpha t} \\ =& e(g,g)^{\alpha t d' + \sum_{j=1}^{d'} H(w_j) t}. \end{aligned}$$

The equation mostly matches $C_d^{d'} = \frac{d\times(d-1)\times\ldots\times(d-d'+1)}{d'!}$ times. The ciphertext search will be successful if D is equal to D'. It means that d' keywords in trapdoor are the same as d' keywords in index and the doctor's attributes satisfy the access structure at the same time.

Then each node on FB performs consensus protocol with computation results. Finally, return $\{B, URL\}$ to the doctor.

6) **Dec**(CT_m, B, SK). After receiving the URL, the doctor first needs to download the corresponding ciphertext CT_m from CS.
Therefore, with the ciphertext CT_m and the intermediate parameter B, the doctor decrypts the ciphertext correctly with his/her secret key SK as follows.

$$\begin{aligned} \frac{ct_m \cdot B}{e(ct^*, sk_1)} =& \frac{m \cdot e\left(g,g\right)^{\alpha t} \cdot e\left(g,g\right)^{\mu t}}{e\left(g^{\beta t}, g^{\frac{\alpha+\mu}{\beta}}\right)} \\ =& \frac{m \cdot e\left(g,g\right)^{\alpha t} \cdot e\left(g,g\right)^{\mu t}}{e\left(g,g\right)^{\alpha t + \mu t}} \\ =& m. \end{aligned}$$

VI. Security and Performance Analysis

We demonstrate the security of MK-IPSE, and compare the performance in terms of computation and storage cost in this section.

A. Security Analysis

Theorem 1: The security of MK-IPSE can be reduced to the DBDH assumption. A PPT adversary $\mathcal{A}$ can recover files from ciphertexts on the condition that there is a simulator $\mathcal{B}$ can solve the DBDH problem with a non-negligible advantage ε.

Proof: The challenger $\mathcal{C}$ generates a bilinear map $e : G_1 \times G_1 \rightarrow G_T$, where G_1, G_T are two multiplicative cyclic groups of prime order p and g is the generator of group G_1. $\mathcal{C}$ randomly selects a bit $\sigma \in \{0,1\}$ and sends a tetrad (A,B,C,Z) to simulator $\mathcal{B}$. When $\sigma = 0$, $(A,B,C,Z) = (g^a, g^b, g^c, e(g,g)^{abc})$. When $\sigma = 1$, $(A,B,C,Z) = (g^a, g^b, g^c, e(g,g)^z)$, where $a,b,c,z \in Z_p^*$. Define system attribute set $\boldsymbol{U}$, set the number of attributes is n and the vector dimension is $l = n+1$. The simulator $\mathcal{B}$ interacts with the adversary $\mathcal{A}$ as follows.

- ***Setup:*** $\mathcal{B}$ firstly computes $Y = e(A,B) = e(g,g)^{ab}$, selects random elements $s_1, s_2, \ldots, s_l \in Z_p^*$ and then sets $h_i = g^{s_i} (i = 1,2,\ldots,l)$. Finally, $\mathcal{B}$ sends the public parameters $\{Y, h_i(i=1,2,\ldots,l)\}$ to $\mathcal{A}$.
- ***Phase1:***

1) *Secret key and keyword index queries:* $\mathcal{A}$ chooses a series of attribute sets $\{U_1, U_2, \ldots, U_n\}$ and asks for the corresponding secret keys $SK_1, SK_2, \ldots, SK_n$ from $\mathcal{B}$. Besides, on input a number of keyword sets $W_1, W_2, \ldots, W_m$, $\mathcal{A}$ obtains the keyword index ciphertext $CT_{W_i} = \{CT_{\omega_1}, CT_{\omega_2}, \ldots, CT_{\omega_d}\}$, where $i \in [1, m]$.
2) *Trapdoor queries:* $\mathcal{A}$ picks a queried keyword set W_i' and a secret key SK_i, calls the ***Trapdoor***(SK_i, W') algorithm and then generates the trapdoor TK_i, where the secret key SK_i is connected with the attribute set U_i. By running the ***Search***(CT_W, TK) algorithm with the trapdoor TK_i and the keyword index ciphertext CT_{W_i}, $\mathcal{A}$ can receive the queried ciphertext successfully if and only if $D = D'$.

- ***Challenge:*** $\mathcal{A}$ selects an access policy vector $\boldsymbol{x}^*$ and two keyword sets W_0, W_1 that it wants to challenge, and sends $(W_0, W_1, \boldsymbol{x}^*)$ to $\mathcal{B}$. However, all attribute sets $U_1, U_2, \ldots, U_n$ mentioned in ***Phase1*** do not satisfy the access policy P^*, i.e., for $\forall i \in [1, n], \langle \boldsymbol{x}^*, \boldsymbol{y}_i \rangle \neq 0$. $\mathcal{B}$ randomly selects a bit $\tau \in \{0, 1\}$ and a random element t, and encrypts the keyword set W_τ. The ciphertext is
$$CT = \{ct^* = h^t, CT_{W_\tau} = \{e(g^{H(\omega_j)t}, g) \cdot Z(j \in [1, d])\}, ct_i = h_i^t \cdot g^{x_i}(i \in [1, l])\}.$$
If $\sigma = 0$, $Z = e(g, g)^{abc}$. Set $c = t$, then the ciphertext is a valid ciphertext set as follows.
$$CT = \{ct^* = h^t, CT_{W_\tau} = \{e(g^{H(\omega_j)t}, g) \cdot e(g, g)^{abt}(j \in [1, d])\}, ct_i = h_i^t \cdot g^{x_i}(i = 1, 2, \ldots, l)\}.$$
If $\sigma = 1$, $Z = e(g, g)^z$. The ciphertext is shown below.
$$CT = \{ct^* = h^t, CT_{W_\tau} = \{e(g^{H(\omega_j)t}, g) \cdot e(g, g)^z\}(j \in [1, d])\}, ct_i = h_i^t \cdot g^{x_i}(i = 1, 2, \ldots, l)\}.$$
Here, z is selected from Z_p^* randomly, which results in CT_{W_τ} is a random collection in group G_T. So $\mathcal{A}$ can not decrypt CT correctly in this condition.
- ***Phase2:*** Similar to ***Phase1***, $\mathcal{A}$ continues to construct the attribute sets $U_{n+1}, U_{n+2}, \ldots, U_q$ and requests secret keys about these attribute sets from $\mathcal{B}$, where q is the number of secret key queries and all these attribute sets do not satisfy the access policy P^*. At the same time, $\mathcal{A}$ inputs the keyword sets $W_{m+1}, W_{m+2}, \ldots, W_\eta$ to obtain corresponding keyword index ciphertexts $CT_{W_{m+1}}, CT_{W_{m+2}}, \ldots, CT_{W_\eta}$.
- ***Guess:*** $\mathcal{A}$ submits the guess τ' to $\mathcal{B}$. If $\tau' = \tau$, it indicates that $\mathcal{B}$ outputs $\sigma' = 0$ and $Z = e(g, g)^{abc}$; If $\tau' \neq \tau$, it means that $\mathcal{B}$ outputs $\sigma' = 1$ and $Z = e(g, g)^z$.
  1) *Case 1*: σ=0. $\mathcal{A}$ decrypts the ciphertext with a non-negligible advantage ε, so $\Pr[\tau' = \tau | \sigma = 0] = 1/2+\varepsilon$. Then, the advantage that $\mathcal{B}$ can solve DBDH problem is $\Pr[\sigma' = \sigma | \sigma = 0] = 1/2 + \varepsilon$.
  2) *Case 2*: σ=1. $\mathcal{A}$ can not get any information about τ, so $\Pr[\tau' \neq \tau | \sigma = 1] = 1/2$. Then, the advantage that $\mathcal{B}$ can solve DBDH problem is $\Pr[\sigma' = \sigma | \sigma = 1] = 1/2$.

Finally, the advantage that $\mathcal{B}$ can solve DBDH problem is $1/2Pr[\sigma' = \sigma | \sigma = 0] + 1/2Pr[\sigma' = \sigma | \sigma = 1] - 1/2 = 1/2(1/2 + \varepsilon) + (1/2)(1/2) - 1/2 = (1/2)\varepsilon$.

However, there is no efficient method to break DBDH assumption, so there is no PPT adversary $\mathcal{A}$ can recover files from ciphertexts and our scheme achieves IND-CKA security.

Theorem 2: MK-IPSE scheme can resist collusion attack.

Proof: It is assumed that multiple doctors whose attribute sets do not pass through the access structure in EMRs tend to unite their secret keys to obtain the access of certain EMRs. However, it does not work in MK-IPSE. Suppose Doctor $\mathbb{A}$ and Doctor $\mathbb{B}$ who respectively have the attribute vectors $\boldsymbol{y}_1$and $\boldsymbol{y}_2$ gain the corresponding secret keys from TA:

$$\{sk_1^1 = g^{\frac{\alpha+\mu_1}{\beta}}, sk_2^1 = g^{\frac{1}{\beta}}, sk_3^1 = g^{\theta_1}, sk_{\boldsymbol{y}_1} = g^{\frac{\mu_1 - \theta_1 \langle s, \boldsymbol{y}_1 \rangle}{\beta}}\}$$

$$\{sk_1^2 = g^{\frac{\alpha+\mu_2}{\beta}}, sk_2^2 = g^{\frac{1}{\beta}}, sk_3^2 = g^{\theta_2}, sk_{\boldsymbol{y}_2} = g^{\frac{\mu_2 - \theta_2 \langle s, \boldsymbol{y}_2 \rangle}{\beta}}\}$$

We know that $\langle \boldsymbol{x}, \boldsymbol{y}_1 \rangle \neq 0$ and $\langle \boldsymbol{x}, \boldsymbol{y}_2 \rangle \neq 0$, so Doctor $\mathbb{A}$ and Doctor $\mathbb{B}$ would like to unite their attributes and get a legal attribute vector $\boldsymbol{y}$ that satisfies $\langle \boldsymbol{x}, \boldsymbol{y} \rangle = 0$. However, because of the random elements μ and θ, they can not decrypt ciphertexts of EMRs successfully by uniting their attributes or inferring the correct secret keys. Hence, our scheme resists collusion attack securely.

B. Performance Analysis

1) Functional Comparison: In this part, we compare the computation and storage cost with the existing schemes [26] and [27] which both support multi-keyword search and have similar system models to ours. The scheme in [26] uses LSSS matrix to determine whether user's attribute sets match access rights. Besides, It utilizes fast and efficient 0-1 coding theory to encrypt the keywords, greatly reducing the computation overhead and providing different kinds of search methods such as Boolean search and rang search, which is the first application in SE. The scheme in [27] takes advantage of access tree and key blinding technique to achieve access control mechanism. In addition, it proposes a two-phase search algorithm to achieve stable and secure multi-keyword retrieval. The scheme [27] does not involve decryption operations, so we will not consider the performance analysis in this stage. In MK-IPSE, through the inner product of attribute vector as well as access policy vector, FB determines if the requester is permitted to get the shared data. Moreover, based on FB, the proposed scheme achieves the adaptive, secure and stable multi-keyword search by simply changing the number of keywords. Finally, we summarize the functional features in TABLE I.

2) Theoretical Analysis: In order to show the analysis results intuitively, we list the related notations and their corresponding definitions in TABLE II.

In TABLE III, we show the computation cost of ABSE-ERM in [26], ABKRS-KGA in [27], MK-IPSE. For the convenience of theoretical analysis, we mainly consider the imperative cryptographic operations such as encryption, generation of trapdoors and search. In encryption phase, the number of encrypted keyword indexes in scheme [26] and [27] is $3W$, and that in MK-IPSE is W. Besides, the computation consumption of attribute ciphertexts is similar in above three schemes, so our solution spends the lowest computation resources in general. While generating the trapdoors, MK-IPSE

TABLE I
Functional Comparison

Scheme	Immutability	Single point of failure resistance	Multi-keyword search	Access policy hiding	Access structure
ABSE-ERM [26]	×	×	√	×	Access tree
ABKRS-KGA [27]	×	×	√	×	Access tree/LSSS
MK-IPSE(ours)	√	√	√	√	Inner product encryption

TABLE II
Notations for Theoretical Analysis

Notation	Definition	Notation	Definition
G_i	Exponential operation in group (i=1,T)	W	Number of keywords in index
P	Bilinear pairing operation	M	Number of keywords in trapdoor
N	Number of attributes in ciphertext	t	Average number of keywords match
S	Number of attributes in secret key	$\|*\|$	Bit-Length of element in $*$

involves only the sum of all queried keywords' hash instead of all keywords' hash like [26] and [27]. So the computation overhead in our scheme is a constant while that grows linearly with keywords increase in [26] and [27]. Therefore, MK-IPSE has advantages in the stage of trapdoor generation. In ciphertext search stage, MK-IPSE needs to check the user's attributes and queried keywords respectively, and compute the equation $D = D'$ repeatedly for adaptive search, so the computation overhead in MK-IPSE is a little more than that in [26]. In addition, the scheme in [27] pays out the most tremendous computation consumption for stable and secure multi-keyword search.

TABLE IV shows the storage cost concerned with secret keys, ciphertexts and trapdoors in [26], [27] and our scheme. In the stage of secret key generation, it is obvious that MK-IPSE has better performance compared with the other two. MK-IPSE diverts attribute set into a vector and places the attribute vector on the exponent, resulting in the storage cost of secret keys is a constant $4|G_1|$ and does not increase with the growing of attributes which is much lower than $(1 + 2S)|G_1|$ and $(2M + S + 1)|G_1|$ respectively in [26] and [27]. While generating the ciphertexts, [26] and [27] execute more keyword encryption operations and generate more ciphertexts than MK-IPSE, and similarly, their storage overhead is also higher than ours. In trapdoor generation stage, MK-IPSE just stores the attribute vector and the sum of queried keywords' hash values while [26] and [27] need to store all attribute values and keywords' hash values, so the storage source consumption of trapdoors in our scheme is lower than [26] and [27].

3) Simulation Analysis: In this part, in order to graphically show the efficiency analysis of the schemes, we give the simulation results of MK-IPSE, ABSE-ERM in [26] and ABKRS-KGA in [27]. The experimental simulation is implemented by JAVA language, using the JPBC library and choosing the type-A elliptic curve group. The simulation result is the average of 1000 times. The configuration of the computer is an Intel(R) Xeon(R) CPU E3-1230 V2, @3.30GHz, and 16.0 GB RAM. For a detailed comparison, we set $N = S = \{5, 10, 15, 20, 25, 30\}, W = 20, M = 5$. Besides, for the uniformity of comparison, the units of computation time and storage space are ms and KB respectively.

Fig. 3 shows the computation cost comparisons in different stages. It is apparent that all the computation overhead increases linearly with the number of attributes. In encryption phase, since LSSS matrix in [26] and access tree encryption in [27] have complicated constructions while the procedure of inner production encryption in MK-IPSE is simple, the computation time in [26] and [27] is more than that in our scheme as shown in Fig. 3a. Moreover, the **Enc** algorithm of MK-IPSE achieves securer data sharing by providing hidden access policy while [26] and [27] can not work it. Therefore, our solution not only spends less computation overhead, but also achieves more comprehensive functionality. In Fig. 3b, it is obvious that the computation overhead in MK-IPSE is the lowest, because the trapdoors for all keywords are generated at once instead of calculating separately for each keyword like [26] and [27]. Besides, it is clearly that the computation overhead in scheme [26] grows fastest as the number of attributes increases. As shown in Fig. 3c, the **Search** algorithm in scheme [26] is the most efficient, because it adopts fast 0-1 code theory to simplify the search of keywords. In scheme [27], the author makes use of complex and tedious two-phase search system for stable and secure keywords retrieval, which guarantees the privacy in search but results in the hugest computation overhead. Furthermore, in order to realize adaptive multi-keyword search, MK-IPSE matches queried keywords and keyword indexes repeatedly so that it spends more computation resources than [26]. However, as attributes' number grows, the spent of ciphertext search in our solution will be lower than [26].

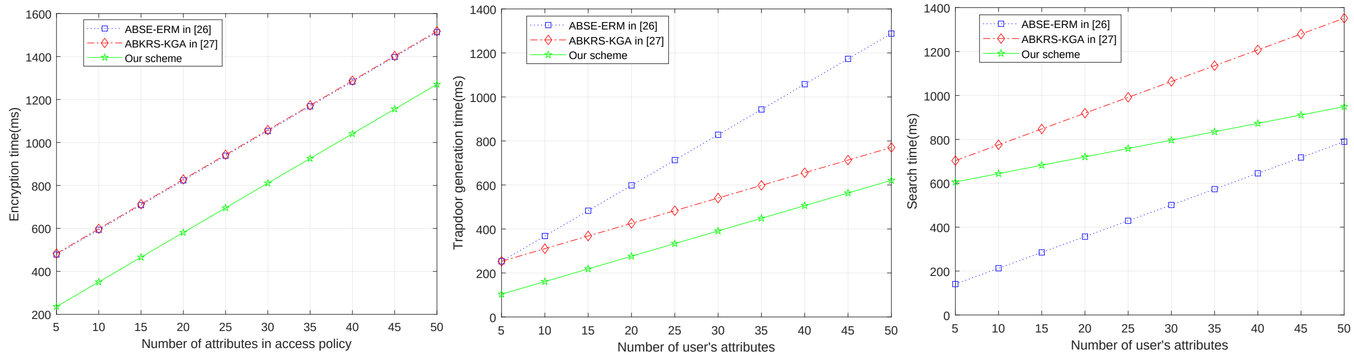
In Fig. 4, the storage consumption in [26], [27] and MK-IPSE is compared mainly involving secret keys, ciphertexts and trapdoors. It is apparent that MK-IPSE always has the lowest storage burden, especially in the stage of secret keys storage, as shown in Fig. 4a. In MK-IPSE, the secret keys are generated with the inner product of attribute vector which is a constant from $Z_p{}^*$, therefore, the storage consumption of secret keys is a constant correspondingly. Different from MK-IPSE, [26] and [27] generate secret keys for each of the user's attributes, so the storage consumption increases linearly with the growing of attributes. In addition, the storage overhead of ciphertexts and trapdoors is shown in Fig. 4b and Fig. 4c respectively which is closely related to the above computation

TABLE III
The Comparison of Computation Cost

Scheme	Encryption	Generation of trapdoors	Search
ABSE-ERM [26]	$(1+2N+3W)G_1+P$	$(2+2S+2M)G_1$	$(2N+2M)P+(1+N)G_T$
ABKRS-KGA [27]	$(2+2N+3W)G_1$	$(2+S+3M)G_1$	$(2S+2M+W)P+(4W+3)G_1+(S+1)G_T$
MK-IPSE(ours)	$(2N+4)G_1+(W+1)P$	$(N+4)G_1$	$(N+3)P+G_1+(N+tM+2)G_T$

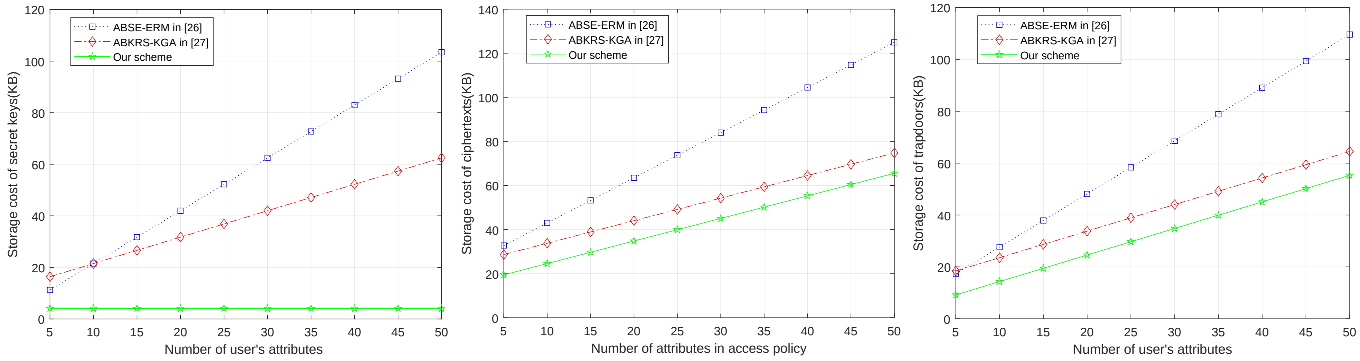
TABLE IV
The Comparison of Storage Cost

Scheme	Secret key	Ciphertext	Trapdoor
ABSE-ERM [26]	$(1+2S)\|G_1\|$	$(1+2N+2W)\|G_1\|+\|P\|$	$(2+2S+M)\|G_1\|$
ABKRS-KGA [27]	$(2M+S+1)\|G_1\|$	$(2+N+2W)\|G_1\|+\|P\|$	$(2+S+2M)\|G_1\|$
MK-IPSE(ours)	$4\|G_1\|$	$(3+N)\|G_1\|+(W+1)\|P\|$	$(N+4)\|G_1\|$



(a) **Enc** algorithm (b) **Trapdoor** algorithm (c) **Search** algorithm

Fig. 3. Computation cost in different algorithms.



(a) **KeyGen** algorithm (b) **Enc** algorithm (c) **Trapdoor** algorithm

Fig. 4. Storage cost in different algorithms.

overhead analysis of encryption and trapdoor generation, so we omit the corresponding analysis contents.

To sum up, the proposed scheme shows good performance in computation and storage overhead compared with [26] and [27]. Besides, our scheme achieves comprehensive functionalities and the privacy preservation of data efficiently. Therefore, MK-IPSE scheme can be fully utilized in practical applications.

VII. Conclusion

The privacy protection and efficient retrieval of medical data are imperative requirements in e-healthcare system. However, as the gradual development of computer techniques, these demands become troublesome to achieve. Based on these concerns, we designed a blockchain-aided medical data resource sharing scheme based on inner product searchable encryption with multi-keyword search (MK-IPSE). It supported secure access permission control as well as hidden policy, accelerating the improvement of medical system and enhancing the security of medical information. Besides, the proposed scheme achieved adaptive multi-keyword search securely and stably with the inherent features in the blockchain. In addition, the security analysis proved MK-IPSE could resist IND-CKA and collusion attacks, which met the requirement of privacy preservation in e-healthcare system. By comparing with the relevant researches, MK-IPSE showed better performance on computation and storage consumption.

Acknowledgments

This work is supported by the Key Research and Development Program of Shaanxi (No. 2020ZDLGY05-04, No. 2021ZDLGY05-03, No. 2023-ZDLGY-34) and Key-Area Research and Development Program of Guangdong Province (No. 2020B0101090001). Besides, this research is supported in part by the ROIS NII Open Collaborative Research 23S0601, and in part by JSPS KAKENHI grant number 21H03424.

References

[1] X. Zhou, X. Ye, K. I.-K. Wang, W. Liang, N. K. C. Nair, S. Shimizu, Z. Yan, and Q. Jin, "Hierarchical federated learning with social context clustering-based participant selection for internet of medical things applications," *IEEE Transactions on Computational Social Systems (Early Access)*, pp. 1–10, 2023.

[2] M. K. Afzal, Y. B. Zikria, S. Mumtaz, A. Rayes, A. Al-Dulaimi, and M. Guizani, "Unlocking 5G spectrum potential for intelligent IoT: Opportunities, challenges, and solutions," *IEEE Communications Magazine*, vol. 56, no. 10, pp. 92–93, 2018.

[3] B. Ji, Y. Wang, K. Song, C. Li, H. Wen, V. G. Menon, and S. Mumtaz, "A survey of computational intelligence for 6G: Key technologies, applications and trends," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 10, pp. 7145–7154, 2021.

[4] X. Zhou, W. Liang, K. I.-K. Wang, and L. T. Yang, "Deep correlation mining based on hierarchical hybrid networks for heterogeneous big data recommendations," *IEEE Transactions on Computational Social Systems*, vol. 8, no. 1, pp. 171–178, 2021.

[5] X. Zhou, Y. Hu, J. Wu, W. Liang, J. Ma, and Q. Jin, "Distribution bias aware collaborative generative adversarial network for imbalanced deep learning in industrial IoT," *IEEE Transactions on Industrial Informatics*, vol. 19, no. 1, pp. 570–580, 2023.

[6] D. X. Song, D. Wagner, and A. Perrig, "Practical techniques for searches on encrypted data," in *Proc. of 2000 IEEE Symposium on Security and Privacy*, 2000, pp. 44–55.

[7] J. Ning, J. Xu, K. Liang, F. Zhang, and E.-C. Chang, "Passive attacks against searchable encryption," *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 3, pp. 789–802, 2019.

[8] 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, 2021.

[9] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute-based encryption," in *Proc. of 2007 IEEE Symposium on Security and Privacy*, 2007, pp. 321–334.

[10] J. Ning, Z. Cao, X. Dong, K. Liang, H. Ma, and L. Wei, "Auditable σ-time outsourced attribute-based encryption for access control in cloud computing," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 1, pp. 94–105, 2018.

[11] Y. Miao, J. Ma, X. Liu, J. Weng, H. Li, and H. Li, "Lightweight fine-grained search over encrypted data in fog computing," *IEEE Transactions on Services Computing*, vol. 12, no. 5, pp. 772–785, 2019.

[12] Y. Miao, J. Ma, X. Liu, X. Li, Q. Jiang, and J. Zhang, "Attribute-based keyword search over hierarchical data in cloud computing," *IEEE Transactions on Services Computing*, vol. 13, no. 6, pp. 985–998, 2020.

[13] T. V. X. Phuong, G. Yang, and W. Susilo, "Hidden ciphertext policy attribute-based encryption under standard assumptions," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 1, pp. 35–45, 2016.

[14] K. Yang, Q. Han, H. Li, K. Zheng, Z. Su, and X. Shen, "An efficient and fine-grained big data access control scheme with privacy-preserving policy," *IEEE Internet of Things Journal*, vol. 4, no. 2, pp. 563–571, 2017.

[15] K. Jonathan, S. Amit, and W. Brent, "Predicate encryption supporting disjunctions, polynomial equations, and inner products," in *Proc. of Eurocrypt on Advances in Cryptology*, 2008, pp. 146–162.

[16] H. Wee, "Attribute-hiding predicate encryption in bilinear groups, revisited," in *Proc. of Theory of Cryptography*, 2017, pp. 206–233.

[17] X. Zhou, W. Liang, K. I.-K. Wang, and S. Shimizu, "Multi-modality behavioral influence analysis for personalized recommendations in health social media environment," *IEEE Transactions on Computational Social Systems*, vol. 6, no. 5, pp. 888–897, 2019.

[18] X. Zhou, Y. Li, and W. Liang, "CNN-RNN based intelligent recommendation for online medical pre-diagnosis support," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 18, no. 3, pp. 912–921, 2021.

[19] Y. Yang, X. Liu, R. H. Deng, and Y. Li, "Lightweight sharable and traceable secure mobile health system," *IEEE Transactions on Dependable and Secure Computing*, vol. 17, no. 1, pp. 78–91, 2020.

[20] L. Xu, X. Chen, F. Zhang, W. Li, H.-T. Wu, S. Tang, and Y. Xiang, "ASBKS: Towards attribute set based keyword search over encrypted personal health records," *IEEE Transactions on Dependable and Secure Computing*, vol. 18, no. 6, pp. 2941–2952, 2021.

[21] S. Xu, Y. Li, R. H. Deng, Y. Zhang, X. Luo, and X. Liu, "Lightweight and expressive fine-grained access control for healthcare Internet-of-Things," *IEEE Transactions on Cloud Computing*, vol. 10, no. 1, pp. 474–490, 2022.

[22] S. Xu, J. Ning, Y. Li, Y. Zhang, G. Xu, X. Huang, and R. H. Deng, "A secure EMR sharing system with tamper resistance and expressive access control," *IEEE Transactions on Dependable and Secure Computing*, vol. 20, no. 1, pp. 53–67, 2023.

[23] D. Boneh, G. D. Crescenzo, R. Ostrovsky, and G. Persiano, "Public key encryption with keyword search," in *Proc. of Eurocrypt on Theory and Application of Cryptographic Techniques*, 2004, pp. 506–522.

[24] H. Li, D. Liu, Y. Dai, and T. H. Luan, "Engineering searchable encryption of mobile cloud networks: when QoE meets QoP," *IEEE Wireless Communications*, vol. 22, no. 4, pp. 74–80, 2015.

[25] R. Chen, Y. Mu, G. Yang, F. Guo, X. Huang, X. Wang, and Y. Wang, "Server-aided public key encryption with keyword search," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 12, pp. 2833–2842, 2016.

[26] Y. Li, H. Wang, S. Wang, and Y. Ding, "Attribute-based searchable encryption scheme supporting efficient range search in cloud computing," in *Proc. of 2021 IEEE Conference on Dependable and Secure Computing*, 2021, pp. 1–8.

[27] Y. Chen, W. Li, F. Gao, Q. Wen, H. Zhang, and H. Wang, "Practical attribute-based multi-keyword ranked search scheme in cloud computing," *IEEE Transactions on Services Computing*, vol. 15, no. 2, pp. 724–735, 2022.

[28] L. Cao, Y. Kang, Q. Wu, R. Wu, X. Guo, and T. Feng, "Searchable encryption cloud storage with dynamic data update to support efficient policy hiding," *China Communications*, vol. 17, no. 6, pp. 153–163, 2020.

[29] P. Liu, J. Wang, H. Ma, and H. Nie, "Efficient verifiable public key encryption with keyword search based on KP-ABE," in *Proc. of 2014 Ninth International Conference on Broadband and Wireless Computing, Communication and Applications*, 2014, pp. 584–589.

[30] J. Cui, H. Zhou, Y. Xu, and H. Zhong, "OOABKS: Online/offline attribute-based encryption for keyword search in mobile cloud," *Information Sciences*, vol. 489, pp. 63–77, 2019.

[31] J. Liu, Y. Li, R. Sun, Q. Pei, N. Zhang, M. Dong, and V. C. M. Leung, "EMK-ABSE: Efficient multikeyword attribute-based searchable encryption scheme through cloud-edge coordination," *IEEE Internet of Things Journal*, vol. 9, no. 19, pp. 18 650–18 662, 2022.

[32] D. Koo, J. Hur, and H. Yoon, "Secure and efficient data retrieval over encrypted data using attribute-based encryption in cloud storage," *Computers and Electrical Engineering*, vol. 39, pp. 34–46, 2013.

[33] J. Shi, J. Lai, Y. Li, R. H. Deng, and J. Weng, "Authorized keyword search on encrypted data," in *Proc. of European Symposium on Research in Computer Security*, 2014, pp. 419–435.

[34] H. Wang, X. Dong, and Z. Cao, "Multi-value-independent ciphertext-policy attribute based encryption with fast keyword search," *IEEE Transactions on Services Computing*, vol. 13, no. 6, pp. 1142–1151, 2020.

[35] Y. Miao, X. Liu, K.-K. R. Choo, R. H. Deng, J. Li, H. Li, and J. Ma, "Privacy-preserving attribute-based keyword search in shared multi-owner setting," *IEEE Transactions on Dependable and Secure Computing*, vol. 18, no. 3, pp. 1080–1094, 2021.

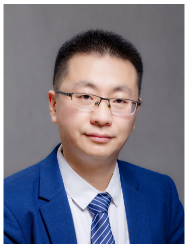
[36] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," [online] Available: http://www.bitcoin.org/bitcoin.pdf, 2008.

[37] 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 of Things Journal*, vol. 7, no. 9, pp. 7851–7867, 2020.

[38] X. Yang, T. Li, R. Liu, and M. Wang, "Blockchain-based secure and searchable EHR sharing scheme," in *Proc. of 2019 4th International*

Conference on Mechanical, Control and Computer Engineering, 2019, pp. 822–825.

[39] S. Niu, L. Chen, J. Wang, and F. Yu, “Electronic health record sharing scheme with searchable attribute-based encryption on blockchain,” *IEEE Access*, vol. 8, pp. 7195–7204, 2020.



Jingwei Liu (Member, IEEE) received the B.S. degree majoring in applied mathematics, and the M.S. and Ph.D. degrees majoring in communication and information systems from Xidian University, Xi’an, China, in 2001, 2004, and 2007, respectively, where he is currently with the School of Telecommunications Engineering. He has published more than 70 papers in journals and conference proceedings and authored or coauthored two books. His research interests include big data security and privacy preservation, cloud and edge computing, and cryptography. He is a member of the Chinese Association for Cryptologic Research.

Yue Fan is currently pursuing the M.S. degree in Communication Engineering with Xidian University in 2021. Her research interests include searchable encryption, attribute-based access control, information security and privacy preservation.

Rong Sun (Member, IEEE) received her B.E. degree in telecommunications engineering, her M.E. degree in communications and information systems, and her Ph.D. degree in communications and information systems from Xidian University, Xi’an, China in 1998, 2001, and 2008, respectively. She is a member of IEEE and IEICE. She is now with the School of Telecommunications Engineering, Xidian University. Her research interests include wireless communications, channel coding design, and information theory.

Lei Liu (Member, IEEE) received the B.Eng. degree in electronic information engineering from Zhengzhou University, Zhengzhou, China, in 2010, and the M.Sc. and Ph.D. degrees in communication and information systems from Xidian University, Xi’an, China, in 2013 and 2019, respectively. From 2013 to 2015, he worked with a subsidiary of China Electronics Corporation. From 2018 to 2019, he was supported by the China Scholarship Council to be a Visiting Ph.D. Student with the University of Oslo, Oslo, Norway. From 2020 to 2022, he was a Lecturer with the School of Telecommunications Engineering, Xidian University. He is currently an Associate Professor with the Guangzhou Institute of Technology, Xidian University. His research interests include vehicular ad hoc networks, intelligent transportation, edge intelligence, and distributed computing.

Celimuge Wu (Senior Member, IEEE) received the Ph.D. degree from The University of Electro-Communications, Japan, in 2010. He has been an Associate Professor with The University of Electro-Communications since 2015. His research interests include vehicular networks, edge computing, the IoT, intelligent transport systems, and application of machine learning in wireless networking and computing.

He was a recipient of the 2021 IEEE Communications Society Outstanding Paper Award, the 2021 IEEE INTERNET OF THINGS JOURNAL Best Paper Award, and the IEEE Computer Society 2019 Best Paper Award Runner-Up. He is the Chair of IEEE TCGCC Special Interest Group on Green Internet of Vehicles and IEEE TCBD Special Interest Group on Big Data with Computational Intelligence. He serves as an Associate Editor for IEEE TRANSACTIONS ON NETWORK SCIENCE AND ENGINEERING, IEEE TRANSACTIONS ON GREEN COMMUNICATIONS AND NETWORKING, IEEE OPEN JOURNAL OF THE COMPUTER SOCIETY, Wireless Networks, and IEICE Transactions on Communications. He has also been a Guest Editor of IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS, IEEE TRANSACTIONS ON EMERGING TOPICS IN COMPUTATIONAL INTELLIGENCE, and IEEE Computational Intelligence Magazine.

Shahid Mumtaz (Senior Member, IEEE) received the master’s degree from Blekinge Institute of Technology, Karlskrona, Sweden, in 2005, and the Ph.D. degree from the University of Aveiro, Aveiro, Portugal, in 2011.

He has authored four technical books, 12 book chapters, more than 300 technical papers, more than 200 IEEE Journals/transactions, and more than 100 conferences. Most of his publication is in the field of wireless communication.

Dr. Mumtaz is the recipient of the NSFC Researcher Fund for Young Scientist in 2017 from China and the IEEE ComSoC Young Researcher Award in 2020. He was awarded an Alain Bensoussan Fellowship in 2012. He received two IEEE Best Paper Awards in the area of mobile communications. He is the Founder and the EiC of IET Journal of Quantum Communication, a Vice Chair of Europe/Africa Region—IEEE ComSoc: Green Communications & Computing society and the IEEE standard on P1932.1: Standard for Licensed/Unlicensed Spectrum Interoperability in Wireless Mobile Networks. He is serving as the Scientific Expert and the Evaluator for various Research Funding Agencies. He is an IET Fellow, the IEEE ComSoc Lecturer, and an ACM Distinguished Speaker.