IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

SPECIAL SECTION ON INTELLIGENT BIG DATA ANALYTICS FOR INTERNET OF THINGS, SERVICES AND PEOPLE

Received March 3, 2021, accepted March 17, 2021, date of publication March 22, 2021, date of current version March 31, 2021.

*Digital Object Identifier 10.1109/ACCESS.2021.3067958*

# A Federated Framework for Fine-Grained Cloud Access Control for Intelligent Big Data Analytic by Service Providers

**GYEONGJIN RA[1], DONGHYUN KIM[2], (Senior Member, IEEE), DAEHEE SEO[3], (Member, IEEE), AND IMYEONG LEE[1], (Member, IEEE)**

[1]Department of Software Convergence, Soonchunhyang University, Asan 336745, South Korea
[2]Department of Computer Science, Georgia State University, Atlanta, GA 30303, USA
[3]Faculty of Artificial Intelligence and Data Engineering, Sangmyung University, Seoul 03016, South Korea

Corresponding author: Imyeong Lee (imylee@sch.ac.kr)

This work was supported in part by the Ministry of Science, ICT (MSIT), South Korea, through the High-Potential Individuals Global Training Program supervised by the Institute for Information and Communications Technology Planning and Evaluation (IITP) under Grant 2020-0-01596, and in part by the Soonchunhyang University Research Fund.

**ABSTRACT** This paper proposes a novel data-owner-driven privacy-aware cloud data acquisition framework for intelligent big data analytics for service providers and users. To realize this idea, we propose three main components. The first one is a new global identity provider concept to support fine-grained access control for a federated outsourcing cloud, namely called P-FIPS (Privacy-enhanced Federated Identity Provider System), in which data owners perform identity access control with the operator of the federated outsourcing cloud so that the service providers can selectively use their encrypted data on the cloud for various purpose such as intelligent big data analytics. In P-FIPS, data owners manage the access privilege of service providers over their encrypted data on the cloud by (a) labeling the scope of use (e.g., user connection, user disconnection, user tracking) on each encrypted data on the cloud, and (b) by selectively providing the information regarding the data owners to the service provider. The label also includes the attributes related to the data owner's identity, and this allows service providers to locate the target data with the assist of cryptographic computation according to the scope of the use at the cloud outsourcing server. The second one is a new ambiguous data acquisition mechanism integrated with P-FIPS from a cloud to a service provider. The last one is the Decentralized Audit and Ordering (DAO) Chain mechanism which provides the correctness of obtained data to the service provider as well as ensures the owners that their data is being used for the approved purpose only. Most importantly, we show that our framework is much more efficient than the existing alternative in the scheme.

**INDEX TERMS** Privacy, self-sovereign, intelligent big data analytics, federated cloud, access control, outsourcing cloud, identity provider.

## I. INTRODUCTION

IoT Analytics is predicted that by 2025, more than 20 billion devices will be connected to the internet [1]. With the explosive increase of IoT devices in everyday environments, many global services are making innovative changes centered on user data by discovering hidden data insights through 'the digitization' process and intelligent big data analytic, and the speed of such changes continues increase [2]. The problem, however, is that the amount of raw data that occurs every day is truly enormous, and the most apparent problem is managing and optimizing a variety of complex systems. It might take that the centralized approach can maximize processing efficiency, but it is challenging to apply to legacy systems. Data silo causes difficulty to integrate the data needed for advanced analysis [3]. The limitations of data significantly affect the quality of the data used by artificial intelligence (AI) and the reliability of the model's predictions. New technologies such as distributed learning provide a path forward, but unfortunately, lack of transparency tends to undermine confidence in the data used for analysis [4]. In addition, although the data is composed of personal information or content produced

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

by users of the service, control over this data is monopolized by the service providers. Therefore, how to guarantee sovereignty over the data they produce to each user who produces such data and how to solve the privacy problems that may arise due to misuse of personal data is emerging as an issue.

Typically, all of the user's data collected from IoT devices run on a competent IT infrastructure (e.g., cloud center) as a digital/logical/cyber/virtual representation/replica of a physical system [5]. The cloud data center builds a cloud server in the center and processes real-time data generated by each distributed outsourcing server. Semi-trust outsourcing servers are vulnerable to malicious attacks (data learning, data inferring, forgery, man-in-the-middle attacks, etc.) [6]. Therefore, users usually perform normal encryption for privacy protection. However, after data stored and processing returning the entire ciphertext to the user would significantly increase the user's computing and network overhead, contrary to the original intent [7], [8]. Customized information can be provided to users through attribute-based cryptograms [9]–[11]. However, it can connect and infer between attributes and personal users if publishing attribute policies. This is the first challenge. It is important to consider lightweight and protect privacy for data sharing. Outsourced servers provide support for data security, and searchable encryption technology has been extensively studied with cloud adoption [12]–[14]. To privacy, outsourcing servers should have as few associations between identifiers and encrypted files as possible [15], [16]. To solve issues, several ambiguous cryptographic transmission schemes have been adopted. However, in order to provide the truth of a single version of the interaction, an additional trust manager is needed, resulting in undesirable delays and response to requests [17], [18]. Blockchain, a new decentralized database paradigm, can provide the promised value for digital artifacts and provide transparency by recording the transaction of each other without a third party [19]. However, In a practical view, it is difficult for the Peer to Peer(P2P) blockchain to support a big data environment.

For data use after data stored, users request identification and storage authorization from the service company (Server). The cloud IDP (Identity Provider) provides efficient user identification and data access control at the same time by mediating data access between repetitive users and service providers [20]. However, the traditional IDP structure remained in web-based services' client-server structure. Numerous users (Clients) provide their personal information to Internet service companies (Server) in exchange for the free use of the service, and Internet service companies exclusively manage all information such as user identifiers and data [21]. Each user receives exclusive services from their respective Service Providers (SP) in a centralized form to other companies. Each service provider individually performs all functions of identification (I: Identification), data storage (S: Storage), and service application (A: Application), which are the essential elements of the service. Therefore, users cannot recognize or control the form of corporate monopoly [22]. The recent IDP model provides horizontal trust, unlike the previous one. The blockchain-based decentralized IDP [23]–[25] specified by the World Wide Web Consortium (W3C) is called DID and provides a single truth and defines the privacy boundaries of users. First, the user makes a decision and then negotiates through communication with colleagues. However, DID [26] is just a token that temporarily grants permission for a specific task and does not contain any information about the user. User identifiers can be accumulated through the blockchain and linked to user personal information. This is the second challenge. To technically solve the monopoly of user data of a specific operator, identification, storage, and application are needed one framework with a relation [27], [28]. Table 1 summarizes the reminders that the framework should consider. Between the limited computing power of IoT and the user's data availability, the framework considers the following four perspectives (availability, efficiency, privacy, security) and sub-functions.

**TABLE 1. Considering issues to apply the data-owner-driven framework.**

| Issue | Character |
|---|---|
| Reliability | a) The truth of a single version of the database<br>b) Threat identification to problem-solving<br>c) Single point of failure |
| Privacy | a) Infer of personal information due to a shared identifier<br>b) Centralized personal data hacking<br>c) Selection of information scope<br>d) User data sovereignty with data awareness<br>e) Information balance through post-management of data |
| Security | a) Forgery and denial to conceal financial interest and adverse information<br>b) Man-in-the-middle attacks through masquerading |
| Efficiency | a) Technology integration to build trust<br>b) Redundant data storage<br>c) Encryption search<br>d) Access from anywhere, anytime |

### A. NAIVE SOLUTION

Our research goal is to allow access to privacy-enhanced data of attributes by using blockchain and cloud outsourcing. To insist on our research objectively, we introduce the privacy policy that GDPR and the global companies consider as follows [29].

(a) *Data minimization* that allows access to only minimized data using technology
(b) *Transparency and data management* that allows users to check the collected data and make their own selection

The key contribution of the proposed research work is summarized as follows.

(a) *P-FIPS*: Users control the use of data by service providers by labeling the scope of use of information (user connection, user disconnection, user tracking). Privacy labeling stores user attributes data in outsourcing servers and allows service providers to access information by calculating cryptography according to user labeling. We want to provide efficiency by applying an outsourced cloud and searchable encryption. At this time,

users lead privacy labeling and search keywords for it, thereby enhancing privacy [22].

(b) *DAO Chain*: Considering the Honest-but-curious model, blockchain is applied for audit without TTP. Blockchain performs as a Rewind Simulator by capturing valid information of participants and the behavior of the protocol. Objective verification provides a balance of information between users-information providers-service providers. In addition, we consider efficiency by separating the chain that records operation performance and verification information via state channel [30], [31].

(c) *Ambiguous Data Acquisition Mechanism*: Oblivious keyword search with authorization (OKSA) creates a trap door based on the keyword set, the information provider's token, and the service provider's private key so that between the user is mutual authenticated and data shared [27]. In addition, the oblivious trapdoor is to make sure that the user-connected keywords that belong to the keyword set are known but cannot be distinguished for privacy.

Our framework is not a trivial combination of blockchain technology and outsourcing cloud. It is built on a new concept of the oblivious and on-off chain. Through the concept of approval, each object authenticates the relation and preserves privacy by checking each other through information imbalance. In other words, it combines each other's information to prevent contamination of the entire system and provide forensics as an auditor of law and investigation results if necessary. Users reduce computation burden through a cooperative system and improve privacy through objective verification and data leadership. Beyond simple qualification verification, service providers can make data available, providing a concept that was not implemented in the existing blockchain.

### B. ORGANIZATION

The rest of this paper is organized as follows. Section II gives the initial knowledge of federated identity credential and searchable encryption for understanding the idea of our paper, Section III details the previous research work that has been presented preliminaries, and Section IV presents an overview of proposed framework defined objective, workflow, system requirements, Section V introduce the construction of our framework with the protocol. Then, in Section VI, We analyze our framework to system requirements. And we implement simulation, verify efficiency comparison with other schemes. Finally, Section VII concludes this research along with future directions.

## II. RELATED WORK

This chapter describes frameworks and cryptographic algorithms for identification, data processing, and storage. The following related work describes existing research to improve the consideration of cloud identity provider framework.

### A. FEDERATED IDENTITY CREDENTIAL

The IDP manages the user's personal data and identity in the IDP paradigm, where various providers handle multiple accounts and attributes. Open authentication (OAuth) works by defining a common password between the SP and the user-mediated IDP called ''Token.'' The OAuth token, on the other hand, has no privacy properties [15]. While the user facilitates the token creation, attributes such as personal data are transferred directly from the IDP to the SP once the user's token has been approved. The consumer is ''out of the loop'' at this point, and the essence of the personal data being transferred is uncertain. Traditional solutions, in short, applied a centralized identity provider, and IDP does not reject user authentication or protect against permitted denial of service attacks.

Personal data management systems [19] have been proposed with the help of blockchain technology to enable users to control their data. Blockchain-based identity management is provisioned using decentralized trust methods without a single identity provider [24]. Key pairs explicitly represent the user, and a single blockchain is maintained through agreements between verifiers and nodes that use algorithms to reach an agreement on the state of the blockchain [26]. However, user data might be more privacy issues such as correlation attacks than traditional federated IDs with OAuth because the user's attribute is distributed at blockchain [32]. Thus, privacy-enhancing techniques such as anonymous authentication credentials function as authenticating users and transferring data while ensuring privacy through encryption [32].

Many schemes for optional public credentials based on RSA or DH assumptions have emerged as a result of Chaum's blind work [34], but these schemes usually require a centralized credential provider and cannot be publicly verified [20]. After that, A healthcare chain [33] is proposed to promote data interoperability and confidentiality in health information networks. Also, [26] proposed enhanced pribacy identitification (EPID) that applied Direct Anonymous Attestation (DAA) and zero-knowledge proof(i.e., Camenisch-Lysyanskaya Signature) to the blockchain. The authors in [36] proposed to issue and manage internal and external certification separately for anonymous certification. To ensure data utilization and data confidentiality, the authors in [27] proposed data exchange on distributed storage based on OKSA in its framework instead of specific algorithms. However, existing studies, including existing DIDs, provided validation and scope proof of attributes but did not consider the attributes available. Existing schemes do not consider token distribution and transfer, or we are only considering environments tailored to Peer to Peer in Public blockchain [35]. Blockchain adopts the evolution of personal information in the concept of security tokens. Security token-based blockchain can help users address some fundamental issues related to privacy and governance and improve trust and scalability [9], [33].

Therefore, first, privacy will be largely off-chain and relies heavily on trusted central institutions to access information and keep it locally. Next, privacy solutions based on state chains can separate data into different sets and hide it across

the public network. Finally, privacy can reside directly on the chain from a more specialized security token blockchain so that owner and property information can control privacy access levels.

### B. SEARCHABLE ENCRYPTION (SE)

Cloud outsourcing server (COS) uses SE technology to offer critical information retrieval services to cloud clients while protecting their privacy. Symmetric Searchable Encryption (SSE) is more effective, but it has a more difficult secret key distribution/management process during data sharing [12]. To address key management issues, Zhang *et al.* introduced the principle of public key encryption using keyword search (PEKS) [13]. Following that, combinable multi-keyword search techniques, such as Public Key Encryption [10], [14] have been proposed to include a range of search functions. Both of the above methods, on the other hand, are honest-but-suspicious cloud environments that can't check the validity of search results. Usually, third parties have partial confidence COS, which is inadequate since they can deliberately return false search results under various synchronizations. A honest-but-curious COS, for example, can run part of the search job or reverse part of the incorrect search results to save compute and bandwidth resources. To allow authorized personal searches, the public encryption oblivious Keyword Search (PEOKS) [19] has been proposed. The $k$-out-of-$n$ unknown transmission system is the most popular oblivious transfer (OT) system type (OTkn). $S$ has $n$ messages, and $R$ is searching $k$ messages at the same time, so $S$ has no know what $R$ receives. Centered on the two-way OT protocol between the SP and the user, Ogata and Kurosawa [9]introduced the concept of ambiguous keyword search to resolve user privacy concerns in keyword searches.

## III. PRELIMINARIES

### A. BILINEAR PARINGS

It is the use of a pairing between elements of two cryptographic groups to a third group with a mapping $\mathbb{G}_1 \times \mathbb{G}_2 \rightarrow \mathbb{G}_T$. Let $\mathbb{G}_1$, $G_2$ be two additive cyclic groups of prime order $q$, and $\mathbb{G}_T$ another cyclic group of order $q$ written multiplicatively. A pairing is a map: $e : \mathbb{G}_1 \times \mathbb{G}_2 \rightarrow \mathbb{G}_T$ which satisfies the following properties:

- Bilinear: $\forall u \in \mathbb{G}_1, v \in \mathbb{G}_2$ and $a, b \in \mathbb{Z}_n : e(u^a, v^b) = e(u, v)^{ab}$.
- Nondegenerate: $e \neq 1$
- Computability: There exists an efficient algorithm to compute $e$.

### B. OBLIVIOUS KEYWORD SEARCH WITH AUTHORIZATION

In OKSA, the data user, such as a service provider (SP), produces a keyword token for any keyword in the authorized keyword set. Then the data provider, such as a cloud outsourcing server (COS), establishes the trapdoor with the obtained token, its private key, and the authorized keyword set [17].

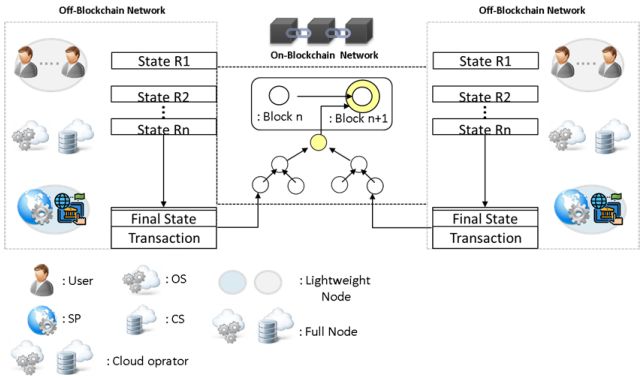



**FIGURE 1.** Architecture.

OKSA is composed of a detailed algorithm that is defined in [27].

(a) **Setup.** A public/private key is generated using the security parameter $\delta$ and the integer $n$. Negotiate a keyword set $W$ between the user and COS, where $|W| \leq n$ is the keyword.

(b) **Encyption.** The user encrypts ciphertext $CT_i$ for $w$ using the keyword $w$ and COS's public key and message. All ciphertext is committed to the COS by the user.

(c) **Data Request.**

(i) **Request.** The keyword token $P(w_i')$ is generated using the COS input of the allowed keyword set $W$, a given keyword $P(w_i')$, and the public key. Finally, COS uses the private key to compute the transparency signature $\sum$

(ii) **Commit.** $P(w_i')$ is sent from COS to SP. From the public/private key, keyword token, and keyword collection, $P(w_i')$ is determined. The obtained token is used to create a trapdoor for only one keyword in the allowed keyword set; the signature $\sum$ aids COS in verifying accountability.

(d) **Data Retrieval.**

(i) **Trapdoor.** The SP enters the authorized keyword collection $W$, the keyword $w'$, and the COS' public key, then generates a trapdoor $T_w'$ and sends it to the COS.

(ii) **Verification.** COS sends a trapdoor $Tw'$ to the service provider once the verification is complete.

(iii) **Data Decryption.** Message m is decrypted using an encryption $CT_i'$, a trapdoor $T_w'$, and SP's private key if $w = w'$, otherwise $\perp$.

(e) **Correctness.** If the SP obtains the message of choice after all entities obey the protocol steps above, an oblivious keyword search authorized is correct. Furthermore, the verification of accountability implies that the trapdoor was generated from a single specific keyword in the received token and that this specific keyword is in the permitted keyword set.

## IV. PROPOSED FRAMEWORK

We propose a labeled data access system with cloud outsourcing and a keyword search protocol with data linking token.

The contact costs between the information issuer service and the service provider are constant in scale. The proposed P-FIPS ensures that the service provider can produce the trapdoor data access token for any permitted keyword in the package, but cannot guess which one.

### A. MAIN SYSTEM ENTITIES

The followings are major system entities in our proposed framework.

(a) **System Manager.** System manager charges the whole system. All the users, information issuers, and service providers must register to the system management. It generates system parameters and keeps a public key for the whole client. It also generates consensus vector a for the blockchain network. Existing blockchain systems are beyond this discussion's scope to include all environmental factors such as endorse peers and block generation leaders.

(b) **User.** As the data owner and service consumer, he/she submits data through CO to service provider for a service. The user performs data labeling is by specifying the data disclosure scope. Then, they encrypt data, including index, by extracting connection keywords. They then send the index and cryptogram signature to the CO. Also, as a participant in the blockchain network, they only have the header of the block. The detailed algorithm used to encrypt each data is beyond the discussion scope so that any public key cryptographic algorithm can be applied.

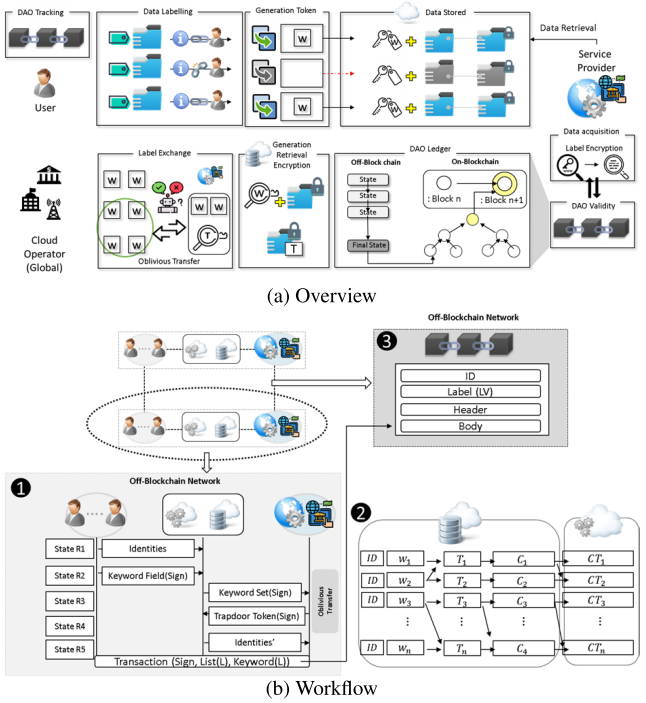
(c) **Cloud Operator (CO).** The CO with expertise and capabilities as an outsourcing server can provide data storage and resource access services to authorized cloud clients (user and service provider) through keyword authentication based on ambiguous data acquisition. It is possible to infer sensitive information available and return false retrievers with various motives. It operates on the cryptographic operation and records and shares the performance details within the blockchain network.

(d) **Resource Server.** This is the provisioning system of CO, which stores user indexes, passphrases, and data linking keywords of users. It stores the data connection passphrase generated by the outsourcing server. It returns resources as requested by the outsourcing server.

(e) **Outsourcing Server.** This is the CO's provisioning system, which performs operations to access data between users and service providers. It generates encrypted data containing the service provider's trapdoor generated from users' data linking keywords.

(f) **Service Provider (SP).** After obtaining permission from the user, it can submit a trapdoor to CO to request retrieval queries on the user's data of interest and subsequently manipulate the user's data.

The CO, like [17]–[23], is thought to be truthful yet curious. It only conducts a small percentage of retrieval operations, but it is curious about the sensitive data. It can also return false retrieve results in order to save computing




(a) Overview

(b) Workflow

**FIGURE 2. Proposed system.**

resources. The DAO, on the other hand, is decentralized and can ensure the accuracy of data retrieval performance. The approved data consumer may also retrieve queries without giving the CO any sensitive details.

### B. WORKFLOW

We make up a system with the objects mentioned above. According to the data labeling, our goal is for service provider to obtain users' data from the CO. Our framework mainly consists of the following five phases [27]:

(a) **Initialization.** In the initialization phase, the system manager's global setup and key generation are performed. The system manager generates some transaction parameters, where the public parameters are public. Whole clients generate public/secret key pairs through the public parameters and integer numbers.

(b) **Data Storage.** Users utilize encryption modules to process sensitive plaintext data before providing it to the informant. Data desired to be processed provided through linking keywords. Cryptographic data must provide both access and confidentiality through outsourcing operations by the information provider. It means that valid service providers can access the data.

(c) **Data Retrieval Request.** The service provider requests data acquisition. According to the user's labeling, we want to acquire data by creating a valid trapdoor through the connection keyword. Meanwhile, a state channel of data is created. The operation is updated to the latest state and recorded off-chain.

(d) **Data Retrieval.** The informant checks the validity of the request. It verifies that this request's trapdoor is in the

data connection keyword set and then distributes the data access token to the node service provider. The final state is distributed on the blockchain as a transaction.

(e) **Data verification.** Upon receiving the data access token, the service provider can check the blockchain's validity and perform a retrieve through the token to access the user data stored by the information provider.

### C. SYSTEM REQUIREMENTS

We adopt the semi-honest security model in our study and assume the cloud servers follow an honest but curious model, which has been widely applied in [17]. In this model, the cloud servers will honestly execute the customized protocols and capture and analyze the meaningful information of the data, query requests, and query results. To enhance the security of the system, we adopt the framework of two non-collude clouds (C1 and C2), which has also been widely adopted in recent works [17]. In practice, these non-collude clouds can be provided by competitive cloud service providers, such as Google and Amazon. Such well-known companies are highly impossible to be in collusion with each other. In addition, we assume query users are trusted and will not collude with the cloud or other users. In detail, the privacy requirements are described as follows.

(a) **Data Privacy.** User controls the scope of their data connection, and the information provider cannot know the connection keywords selected by the service provider. It provides service consumers' data and related privacy. When the service provider accesses data from the CO, it guarantees the connection keyword's authority but does not know the retrieved general password data and the associated keyword. In other words, the service provider knows nothing about the data other than the query result of the linking data. The plain text of the query results should not be learned by any party other than the query user.

(b) **Preservation of Trading Orders.** It needs to ensure that the transactions are executed in sequence. It means that each transaction block can be linked to the previous chain as an uninterrupted order.

(c) **Result Verification.** Honest-but-curious, CO can return incorrect retrieve results, compromising data security and seriously impacting the service browsing experience. It needs a result verification mechanism to ensure data retrieval accuracy.

(d) **Efficiency and Feasibility.** To quickly retrieve user data and avoid wasting bandwidth and computing resources, the service provider should be able to retrieve encrypted data and submit multiple labels.

(e) **Security Goals.** In addition, the keywords are selected from a small space and need resist the standard model's keyword guessing attacks. Moreover, our proposed framework need objectively verification the correctness of retrieved results for the honest-but-curious CO without a trusted third-party auditor.

## V. PROTOCOL

This section explains a framework constructed of P-FIPS protocol with OKSA and DAO. Privacy labeling allows the SP to retrieve user data from the user sets data linking fields. At that same time, OKSA provides negotiating data access token obliviously and retrieve user data between SP and CO [27]. DAO guarantees the record transparency of the sequential order. The algorithms in the following main focus on building index and generating each authorized so that the data linking keyword search query a service provider's token via user's labeling can be processed efficiently in the cloud operation. The DAO verify that the retrieval results are accurate. We define a function $F(centerdot)$ that maps the subscripts in the set $[1, q]$ to their corresponding subscript in the set $[1, n]$ for the sake of the following discussion. We upper mention P-FIPS consist of five-phases, and each phase has the step of algorithms which are shown as follows based [27]:

(a) **Initialization Phase.**

(i) Setup$(1^k) \rightarrow pp$: Given the security parameter $k$ and integer $n$, system manager outputs the public parameters $pp$.

(ii) KeyGen$(pp, U, O) \rightarrow \{pk_u, sk_u, pk_{co}, sk_{co}, pk_{sp}, sk_{sp}\}$: For the user $US$, service provider $SP$ and $CO$ generates the public/private key pairs $\{pk, sk\}$ respectively.

(b) **Data Stored Phase.**

(i) Label$(pp, KS, \delta, \mu) \rightarrow \{W\}$: Given the data linking keyword fields $KW$, the users output the data linking keyword set $KW'$.

(ii) Enc$(pp, KW, ID, sk_u, pk_{co}, pk_{sp}) \rightarrow \{I, ENC, OSig\}$: The user input $pp$ and the fields $KW$ output the encrypted data $ENC$, indexes $I$ and ordered signature $OSig$.

(iii) Store$(I, ENC, Sig) \rightarrow \{enc_1, enc_2, enc_3\}$: Given Sig $OSig$, and encrypted data $ENC$ the $CO$ verify the signature and store $ENC = \{enc_1, enc_2, enc_3\}$ in resource server.

(c) **Data Request Phase.** TrapGen$(pp, pk_{co}, sk_{sp}, W) \rightarrow \{T_W, Osig\}$: Given the data linking keyword set $W'$ by CO, a specific service provider generates the data access token $T_W$ and sends to CO ordered Signature $OSig$.

(d) **Data Retrieval Phase.**

(i) Retreive$(pp, T_K W, I, C, sk_{co}) \rightarrow ENC'$: Following the acquisition of the data linking token $T_K W$, the CO searches the data liking index set $I$ for a match and returns the appropriate encrypted data set $ENC'$ to the DAO chain.

(ii) BlockcGen$(pp, SK_{co}, ENC', OSig) \rightarrow Bloc$: The CO input final state and path and generate $OSig$ then record transaction and commit DAO chain.

(e) **Verification Phase.** Verify$(pp, ENC', ID', pk_{sp}) \rightarrow \{0, 1\}$: The DAO produces "1" when$C'$ passes the results

**TABLE 2.** System notations and description.

| Notation | Decription |
|---|---|
| $U$ | User |
| $SP$ | Service provider |
| $CO$ | Cloud operator |
| $pk_*, sk_*$ | Public/private key pair for * |
| $INF = \{inf_1, \ldots, inf_n\}$ | Data set for user information |
| $ID = (id_1, \ldots, id_d\}$ | Identities for INF |
| $ENC = \{ec_1, ec_2, ec_3\}$ | Encrypted data for F, $enc_1$=The Encrypted data of user information $inf$, $enc_2$=The encrypted data of the data linking keyword $kw$ which is stored resource server and retrieve outsourcing server, $enc_3$=A value of validation through the DAO chain. |
| $KW = \{kw_1, \ldots, kw_m\}$ | Data linking Keyword fields for each information |
| $Sig_{i,t}$ | Users' $U_d$' signature for $inf_i$ |
| $Sig_i$ | Ordered multi-signature for $inf_i$ |
| $OSig = (sig_i, \ldots, sig_n\}$ | Ordered Signature for $F$ |
| $I_i$ | Index for $f_i$ |
| $I = \{I_1, \ldots, I_n\}$ | Index set for $INF$ |
| $KW' = \{kw'_1, \ldots, kw'_m\}$ | Queried data linking keyword set |
| $T_{KW'}$ | Data access token for $KW'$ |
| $CT' = \{ct'_1, \ldots, ct'_n\}$ | Retrieval result |
| $ID' = (id'_1, \ldots, id'_n\}$ | Identity set for $ENC'$ |
| $\eta, \pi_\eta$ $(\eta \in (1, q])$ | Challenge information |
| $\rho, \delta$ | Proof information |

verification mechanism; otherwise, $\perp$ after obtaining the search results $ENC'$.

### A. INITIALIZATION PHASE

**Step 1.** System manager take as input a security parameter $k$, integer $n$. It choose a bilinear map system $PG = \{p, \mathbb{G}_1, \mathbb{G}_2, e\}$.

**Step 2.** Then, it selects two hash functions $h_1 : \{0, 1\}^* \rightarrow \mathbb{G}_1$, $h_2 : \{0, 1\}^* \rightarrow Z_p^*$. Finally, it sets the public parameters $pp$ as $pp = \{\mathbb{G}_1, \mathbb{G}_2, e, p, g, h_1, h_2\}$.

**Step 3.** For each user randomly selects $g, h \in \mathbb{G}$, $a, x \in Z_p$ and computes $g^a$, $h_i = h^{a_i}$ for $i = 1, 2, 3, \ldots n$. $pk_u = (g^a, h_1, h_2, \ldots, h^n)$, $sk_u = a$

**Step 4.** For $CO$, it first selects two elements $\alpha \in_R \mathbb{G}, a_s \in_R Z_p^*$ and compute $\beta = g^{a_s}$, then the issuer's public/private key pair is donated as $pk_s = (\alpha, \beta)$, $sk_s = a_s$.

**Step 5.**The SP chooses an element $y \in_R Z_p^*$ and computes $Y = g^y$ , then it sets the public/private key pair of the specific user $u$ as $pk_{sp} = Y, sk_{sp} = y$.

### B. DATA STORED PHASE

**Step 1.** The user selects a data linking keyword $kw$ where $kw \subseteq KW$ and the size of $KW$ is denoted as $(KW = k \leq n)$.

**Step 2.** The user encrypts keywords for retrieving the identities that the $CO$ will use later. The data linking keyword filed is denoted as $KW$ with the size $n$. Each data has its associated data linking keyword. Given a data $d_i \in (0, 1\}^l$ and a keyword $w_i \in KW$ the user chooses $r_i \in Z_p$ and computes the encrypted data $EN$ as

$$ENC_i = enc_{1i} = g^{ri(a+w_i)},$$
$$enc_{2i} = H\left(0, e(g, h)^{ri}\right),$$
$$enc_{3i} = H\left(1, e(g, h)^{ri} \oplus d_i\right) \quad (1)$$

**Step 3.** For each identities $f_i \in F(i \in (1, n])$ with identity $id_i$, each user $u \in U$ generates his signature $sig_{i,t} = (h_1(id_i), g^{h_2(c_i)})^{sk_u}$, and the ordered signature generated by user is set as

$$Osig = \prod_{t=1}^{d} sig_{i,t}, \text{where} t \in [1, d] \quad (2)$$

### C. DATA RETRIEVAL REQUEST PHASE

**Step 1.** The $CO$ sends each identity's index $f_i$, which is developed based on the keyword fields $KW = \{kw_1, \ldots, kw_m\}$. To begin, build a $m$-degree polynomial using the equation $F(x) = b_m x^m + b_{m-1} x^{m-1} + \ldots + b_1 x + b_0$, so that $yh_2(w_1), \ldots, yh_2(w_m)$ are the $m$ roots of the equation $F(x) = 1$. Then, select $\delta, \mu \in_R Z_p^*$ by selecting $\delta, \mu \in_R Z_p^*$ and evaluate

$$I_{i,1} = \gamma \cdot e(g, g)^{-\mu}, I_{i,2} = g^{\delta}, \quad (3)$$
$$v_{i,j} = g^{\mu \cdot b_j} (0 \leq j \leq m), \quad (4)$$
$$\text{where } \gamma = e(\beta, \alpha)^{\delta}. \quad (5)$$

**Step 2.** The $CO$ stores signature set $OSig = \{sig_i, \ldots, sig_n\}$, index set $I = \{I_i, \ldots, I_n\}$ and encrypted data $ENC$ at the outsourcing server, where $I_i = \{I_{i,1}, I_{i,2}, v_{i,0}, v_{i,1}, \ldots, v_{im}\}$.

**Step 3.** The $CO$ given the authorized keyword set $KW$, a keyword $kw_i \in KW$ and the public key $PK_{sp}$, picks $s \in Z_p$ as private key $SK_sp$ and computes the token $T_{KW'}$ and the proof $\Sigma$ as

$$Q(kw_i) = h^{s \prod w_j \in W, j \neq i (a+kw_j)}, \quad (6)$$
$$\Sigma = \Sigma_1 = h^{\frac{a+kw_i}{s}}, \Sigma_2 = \Sigma_1^{a^{n-1}} \quad (7)$$

**Step 4.** Given the tuple $P(kw_i)$, $W$, $\Sigma$ and the public key $PK_{sp}$, the $SP$ checks the accountable by the following equations, if both equations hold, the user accepts the received keyword token is for the trapdoor for one keyword, and we denote is as $Q(kw_i) = 1$; otherwise, $\perp$.

$$e(\Sigma_2, h^a) = e(\Sigma_1, h^a), \quad (8)$$
$$e(h, h^{\prod kw_i \in W(a+w_i)}) = e(Q(kw_i), \Sigma_1) \quad (9)$$

**Step 5.** The $CO$ give $a_i$ and $kw$ to the $SP$ computes the trapdoor $P$ by the following equations then $CO$ returns the trapdoor $P$ to $SP$.

$$P = Q(kw_i)^{\frac{1}{\prod w_j \in KW (a+kw_j)}} \quad (10)$$

**Step 6.** Given the queried keywords set $KW' = \{kw'_r\}(r \in (1, l)$, a SP first selects two element $\theta, \eta \in_R Z_P^*$ and sets $T_{KW', O_1} = \theta$.

**Step 7.** The $SP$ computes $T_{KW', O_2}$ by following equation and sends the data access token $T_{KW'}$ to $CO$.

$$T_{KW',j} = g^{l-1 \cdot T_{KW',1} \cdot y^j \cdot \sum_{r=1}^{l} h_2(w'_r)^j} \cdot \beta^{\eta} \quad (11)$$
$$T_{KW'} = \{T_{KW',O_1}, T_{KW',O_2}, T_{KW',0}, T_{KW',m}\} \quad (12)$$

### D. DATA RETRIEVAL PHASE

**Step 1.** After gaining the data access token $T_{W'}$, the *SP* first computes $\gamma = e\left(I_{i,2},\ \alpha\right)^{a_s}$ and the *CO* returns the relevant retrieval data $CT' = \{ct'_1,\ \ldots\ , ct'_q\}$ and its corresponding identity set $ID = \{id_1,\ \ldots\ id_q\}$ to DAO; otherwise, it returns $\perp$.

**Step 2.** At the beginning, given the trapdoor $T_{W'}$ and the index $I_i$ for each record $f_i (1 \le i \le n)$, the issuer pre-processes the retrieval query with performing $m$ exponentiation operations.

**Step 3.** Afterwards, the *CO* checks whether the submitted trapdoor matches with the index $I_i$ by checking. The *CO* returns the corresponding encrypted data $enc_i$; otherwise, returns $\perp$. Finally, the *CO* returns the whole relevant encrypted data sets $ENC' = \{enc'_1,\ \ldots\ , enc'_q\}$ and its corresponding identity set $ID = \{id_1,\ \ldots\ id_q\}$ or $\perp$ to *DAO*.

**Step 4.** As the value of m is very small in practice, verify computation will not exert a heavy computational burden *DAO*. Thus, the retrieve algorithm is feasible and practical in actual scenarios.

$$I_{i,1}^{T_{W',O_1}} \cdot \prod_{j=0}^{m} e(v_{i,j},\ T_{W'}, j / T_{W',O_2}^{a_S}) = \gamma^{T_{W',O_1}} \tag{13}$$

### E. VERIFICATION PHASE

**Step 1.** After receiving the retrieval results $CT'$, the service provider first selects an element $\pi_\tau \in_R Z_p^*$ for each encrypted identities *id*

**Step 2.** The *CO* sends the challenging information $(\tau, \pi_\tau)_{\tau \in [1,q]}$ to service proivder. After gaining the challenging information, the CO first computes $\varphi$ as

$$\varphi = \sum_{\tau=1}^{q} \pi_\tau h_2(c'_\tau)\ , \tag{14}$$

$$\sigma = \prod_{\tau=1}^{q} (sig'_\tau)^{\pi_\tau} \tag{15}$$

**Step 3.** The *SP* proves information $(\varphi, \sigma)$ through *DAO*, where $sig'_\tau = \prod_{t=1}^{d} sig_{\rho(\tau),t}$. Finally, the *DAO* verifies whether Eq. (15) holds.

$$(\sigma, g) = e(\prod_{\tau=1}^{q} h_1\left(id'_\tau\right)^{\pi_\tau}) \cdot g^{\varphi},\ \prod_{t=1}^{d} PK_{sp} \tag{16}$$

**Step 4.** The above equation, $id'_\tau = id_{\rho(\tau)},\ c'_\tau = c_{\rho(\tau)}$. If holds, the *DAO* justifies that the retrieve results $C'$ are correct and sends them to the specific data user $u$; otherwise, it aborts. The detailed process of results verification can be found. At the beginning, the *DAO* interacts with the *CO* based on the challenge-response mode.

**Step 5.** Afterwards, the *DAO* computes $\prod_{t=1}^{d} pk_t$ and verifies whether the retrieve results $C'$ is correct or not. Finally, the *DAO* draws the conclusion and returns the correct retrieve results to the SP.

**Step 6.** The *SP* given $ENC_i$, $P$, $b_i$. *SP* executes the retrieval operation by

$$enc_{2i} = H\left(0, e(c1_i, P)^{\frac{1}{s}}\right). \tag{17}$$

**Step 7.** If the above equation holds, *SP* computes the decryption operation by

$$mi = enc_{3i} \oplus H\left(1, e(enc_{1i}, P)^{\frac{1}{s}}\right) \tag{18}$$

## VI. ANALYSIS

We compare our framework requirements discussed in Section IV-C. Also, We analyze the efficiency of performance evaluation of our framework and others.

### A. ANALYSIS OF SYSTEM REQUIREMENT

We compare the proposed framework security with the others according to system requirements in Section IV-C.

#### 1) PRIVACY

When a user's data labeling setting and a service provider's request for data access, if the data label is exposed, there is a problem that the user's information can be inferred. We prevent this through the data acquisition mechanism based OKSA [27] above on IND-KGA security [34].

From Eq.(1): $user \rightarrow CO$ send a encrypted data with data linking keyword.

$$\begin{aligned} ENC_i &= (enc_{1i} = g^{ri(a+kw_i)}, \\ enc_{2i} &= H(0, e(g,h)^{ri}), \\ enc_{3i} &= H(1, e(g,h)^{ri} \oplus d_i). \end{aligned}$$

From Eq.(2): $CO \rightarrow SP$ send a data linking keyword.

$$\begin{aligned} Q(w_i) &= h^{s \prod kw_j \in KW, j \neq i(a+kw_j)}, \\ \Sigma &= (\Sigma_1\ h^{\frac{a+kw_i}{s}}, \Sigma_2 = \sum_{1}^{a^{n-1}}). \end{aligned}$$

From Eq.(12): $SP \rightarrow CO$ send a Trapdoor.

$$\begin{aligned} T_{KW',j} &= g^{l-1 \cdot T_{KW',1} \cdot y^j \cdot \sum_{r=1}^{l} h_2(w'_r)^j} \cdot \beta^{\eta} \\ T_{KW'} &= \{T_{KW',O_1},\ T_{KW',O_2},\ T_{KW',0},\ T_{KW',m}\} \end{aligned}$$

For security issues, IND-KGA, as known to all, can ensure that the outsider attacker cannot infer the relationship between the target trapdoor and challenging keyword set even though it can gain other trapdoors. Our framework is secure against IND-KGA in the standard model because the DDHproblem is intractable.

We look at two challenging problems to provide the foundation for the security of OKSA, i.e., $(f, n)$-DHE Problem and $(f, q)$-MSE-DDH Problem. The $(f, n)$-DHE Problem has been based on [34], [36]

*Definition 1 ($(f, n)$-DHE Problem):* Let $G$ be a group of prime order $p$, $h \in G$ and $a \in Z_p$. Given $h, h^a, \ldots, h^{a^n}$, output $(f(x), h^{f(a)})$, where $f(x) \in Z_p[x]$ is a polynomial function with $\deg f(x) > n$ [34].

*Definition 2 ($(f, n)$-MSE-DHE Problem):* Let $PP$ be a bilinear map group system and $g_0, h_0$ be the generators of the group $G$. They assume two pairwise co-prime polynomials $f$ and $q$ with degree 1 and $n - 1$, respectively, where $n$ is an integer. Given

$g_0, g_0^a, g_0^r, h_0^{f(a)}, \ldots, h_0^{\alpha^{n-2f(a)}}, h_0^{f(a)q(\alpha)}, \ldots, h_0^{\alpha^{nf(\alpha)q(\alpha)}}$, and $Z \in G_T$, the goal is to distinguish $Z = e(g_0, h_0)^{rq(\alpha)}$ or a random group element in $G_T$ [36].

#### 2) PRESERVATION OF TRADING ORDERS

We ensure that the transactions are executed in sequence. It means that each transaction block can be linked to the previous chain as an uninterrupted order. All parameters for data retrieval are set between cloud oprator and service provider in the current transaction. Our framework's security can be directly obtained from the DAO blockchain's security and ordered signature. First, each transaction operates on a state basis and is reordered according to timestamp with a transaction ID. Also, each transaction signed the user's private key It is hard to violate this order. Second, the Unforgeability and ordering of signature [24] guarantee the impossibility to reorder the positions of blocks in the chain. Firstly, the indistinguishability of OKSA guarantees secrecy protection of $m_i$ and $w_i$ in our framework, which cannot be obtained without the corresponding secret key of the user. Secondly, the Privacy and Accountability of OKSA provides oblivious retrieval of our framework. In our framework, the CO can know the relationship $w_i \in W$ but not to know the specific label $w_i$ in an oblivious way. The CO also learns nothing about the retrieved plain-cipher data $(m_i, CT_i)$.

From Eq. (2): $User \rightarrow CO \rightarrow SP$ send a ordered signature

$$Osig = \prod_{t=1}^{d} sig_{i,t} \quad \text{where} t \in [1, d].$$

From Eq. (13): $CO \rightarrow SP$

$$I_{i,1}^{T_{KW',O_1}} \cdot \prod_{j=0}^{m} e(v_{i,j}, T_{KW'}, j/T_{KW',O_2}^{a_S}) = \gamma^{T_{KW',O_1}}.$$

#### 3) RESULT VERIFICATION

Honest-but-curious, CO can return incorrect retrieve results, compromising data security and seriously impacting the service browsing experience. It needs a result verification mechanism to ensure data retrieval accuracy. According to the proposed protocol, the cloud allows data access with service providers without relying on a third-party. We assume that the service provider specifies the label $w_i$ and retrieves the data mi associated with $w_t$ and the block $\sigma'$ is verified by more than half of the nodes

From Eq. (16): Service Provider $\rightarrow$ DAO Chain

$$e(\sigma, g) = e(\prod_{\tau=1}^{q} h_1(id'_\tau)^{\pi_\tau} \cdot g^{\varphi}, \prod_{t=1}^{d} PK_{sp}).$$

### B. ANALYSIS OF SYSTEM EFFICIENCY

We deploy a static environment composed of ten nodes without adding or revoking nodes. The block will be seen as valid once two nodes and monitor nodes approve it, respectively, via Proof-of-Authority (POA) consensus (Fig 3). We employ

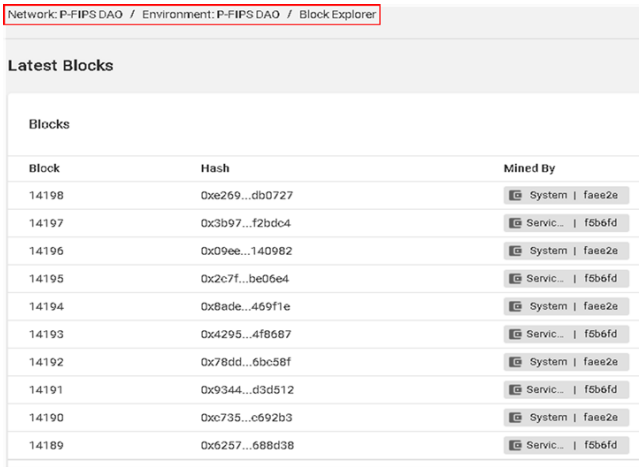



**FIGURE 3.** P-FIPS DAO chain latest blocks.

the ordered multi signature [9] and oblivious transfer keyword authority construction [9] to instantiate our protocol. We conduct the algorithm implementation on a virtual CPU 2G∼4G memory. We exploit RPC and JSON WebSocket, where the language is solidity. We select asymmetric elliptic curve $\alpha$-curve [17], where the base field size is 512-bit, and the embedding degree is 2. The $\alpha$-curve has a 160-bit group order, which means $p$ is a 256-bit length prime. The system is designed using the Ethereum Virtual Machine (EVM) based Kaleido enterprise blockchain as a service and block cloud. Also, we use the Metamask RPC App account and deploy Remix smart contract. The proposed approach is modeled by developing data processing and queries through peer collaboration between users (data owners) and COs and service providers. Here, the user uploads the encrypted data and connection attributes, while the CO generates an access passphrase through the connection attributes. Other configurations are shown in Table 3 and Table 4.

We quantify the transmission bandwidth between two nodes and the computational overhead of several steps in one transaction with ten regular messages. We compare the framework to retrieve encryption and blockchain in the following experiment.

#### 1) TRANSMISSION BANDWIDTH

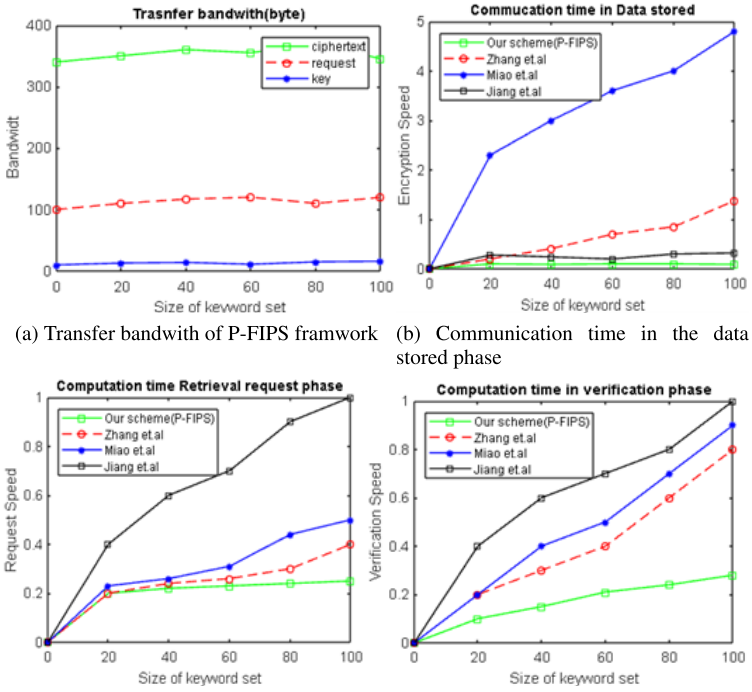
Test the bandwidth of three parameters, including cryptographic data, request, and key, and include the response bandwidth of other nodes for block acknowledgment. We choose a hash function with an output of 256 bits and plain data of the same length. Adjust the connection keyword according to the user's connection range. The experimental results are shown in Figure 4. It can be seen that each parameter is almost constant as the size of the associated keyword set increases. Therefore, the transmission bandwidth is also independent of the keyword set. It is more efficient than PEKS in [17] and multi-keyword conjunctive. We measure the computational overhead in several steps. The details and assumptions required will depend on the step.

**TABLE 3.** Implementation environment of our framework.

| Provider | Language | Consensus Algorithm | Cryptography Algorithm | Chain | PKI |
|---|---|---|---|---|---|
| Go Ethereum (Geth) | Solidity | POA-2 user nodes + monitor node | Prime-256 Elliptic Curve | On-Off hybrid chain | Full PKI Trust chain |

**TABLE 4.** Character of DAO chain each node.

| Name | Node ID | Node Size | Blockchain Node ID | Consensus role |
|---|---|---|---|---|
| User | k0sz1w9xcw | 2GB memory, 1vCPU | f52178373318...3094f34cdf7d | Non-signer |
| CO | k0rjfe9jxr | 4GB memory, 2vCPU | 4793a617b3f7...268621fd2a66 | Signer |
| Service provider | k0tqb7qhpb | 4GB memory, 2vCPU | 223fce74f9f2...ff453538314d | Non-signer |




(a) Transfer bandwith of P-FIPS framwork (b) Communication time in the data stored phase

(c) Computation time in the request and retrieval phase (d) Computation time in the verification phase

**FIGURE 4.** Performance comparison between our framework and other schemes.

### 2) DATA STORED PHASE

Test the calculation time to generate cryptographic data. Encryption for each message is an independent process, so 20 measurements are run for each of the ten pre-set messages. Figure 4b shows the data encryption speed for the data labeling size. It can see that the computation time of the data storage step is independent of the size of the data label.

### 3) DATA REQUEST AND RETRIEVAL PHASE

We test the computation time of the data retrieval request and data retrieval. It time includes keyword and token transfer to commit data searchable encryption, and retrieve one message. The computational work should primarily contribute to data retrieval and decryption. The results are shown in Figure 4c. When the keyword set's size increases, there is no explicit change that occurs when retrieval for messages. It is aggregated into a protocol with an independent retrieve algorithm for a set of keywords as state blockchain.

### 4) VERIFICATION PHASE

Measures the rate of verification, including block approval, request responsibility, and access creation. In the experiment, we assume block validation. Our experiment is created by combining two nodes and one system manager through POA consensus. However, the approach in [17] requires majority consent, and the approach in [33] is valid as long as six nodes approve it. Figure 4d shows the validation rate for the size of a keyword set. It shows that [27] increases the computation time linearly as more keywords are included in the keyword set.

## VII. CONCLUSION

Due to IoT and intelligent big data recent development, global companies provide user-centric services. However, the global services' legacy system is challenging to integrate the data required for high-quality analysis. Therefore, It is hard for accurate response and expecting big data processing reliability. In addition, the absence of an integrated framework leads to the loss of user data sovereignty and misuse of personal data. For data processing, cloud data centers have centrally built cloud servers, each distributed outsourcing server provides support for data security, and has been providing efficient data processing through searchable encryption technology. Among them, Cloud IDP

provides authorization to storage through identification. However, the existing identification and access control structure remained in the client-server structure of web-based service. Also, providing consistent effectiveness for interactions requires a separate manager, resulting in undesirable delays and responses to requests. Blockchain, a new decentralized database paradigm, can realize the promised value for digital artifacts and provide transparency by recording each other's performance without a third party. The blockchain-based decentralized IDP [10] specified by the World Wide Web Consortium (W3C) is called DID and provides a single truth and defines users' privacy boundaries. However, DID [11] is just a token that temporarily grants permission for a specific task and does not contain any information about the user. User identifiers can be accumulated through the blockchain and linked to user personal information. Unfortunately, from a practical point of view, it is difficult for the Peer blockchain to support a big data environment. We still face availability, efficiency, and trust, and security such as privacy.

This paper proposed a new data user-centric, privacy-aware cloud data sharing framework for users and service providers. It is a new global identity provider concept that supports granular access control to a federated outsourcing cloud called P-FIPS (Privacy-enhanced Federated Identity Provider System) where data owners perform identity access control with operators. To efficiently provide encrypted data, the user the scope of use for each labeled data in the cloud (e.g., user connection, user disconnection, user Tracking) to manage service providers' access searchable encrypted data in the cloud. The service provider computes data tokens within the labeled keyword scope and locates data via a cloud server. Also, the DAO chain mechanism provides that the service provider correctness for the received data and ensures that the data is used only for authorized purposes by the user. As a result, we satisfied the existing scheme's security requirements thorough security analysis. Simultaneously, a simple simulation implementation demonstrated that the overhead does not increase regardless of the number of data labels. We plan to develop a framework that operates lightly from the number of objects in the future.

## REFERENCES

[1] Y. Ren, T. Wang, S. Zhang, and J. Zhang, "An intelligent big data collection technology based on micro mobile data centers for crowdsensing vehicular sensor network," *Pers. Ubiquitous Comput.*, pp. 1–17, Aug. 2020.

[2] M. G. Sarowar, M. S. Kamal, and N. Dey, "Internet of Things and its impacts in computing intelligence: A comprehensive review–IoT application for big data," in *Big Data Analytics for Smart and Connected Cities*. 2019, pp. 103–136.

[3] L. Guo, H. Xie, and Y. Li, "Data encryption based blockchain and privacy preserving mechanisms towards big data," *J. Vis. Commun. Image Represent.*, vol. 70, Jul. 2020, Art. no. 102741.

[4] W. Hummer, V. Muthusamy, T. Rausch, P. Dube, K. El Maghraoui, A. Murthi, and P. Oum, "ModelOps: Cloud-based lifecycle management for reliable and trusted AI," in *Proc. IEEE Int. Conf. Cloud Eng. (IC2E)*, Jun. 2019, pp. 113–120.

[5] L. Yu, L. Chen, Z. Cai, H. Shen, Y. Liang, and Y. Pan, "Stochastic load balancing for virtual resource management in datacenters," *IEEE Trans. Cloud Comput.*, vol. 8, no. 2, pp. 459–472, Apr./Jun. 2016.

[6] L. Yu and Z. Cai, "Dynamic scaling of virtual clusters with bandwidth guarantee in cloud datacenters," in *Proc. IEEE 35th Annu. IEEE Int. Conf. Comput. Commun. (INFOCOM)*, Apr. 2016, pp. 1–9.

[7] L. Yu, H. Shen, Z. Cai, L. Liu, and C. Pu, "Towards bandwidth guarantee for virtual clusters under demand uncertainty in multi-tenant clouds," *IEEE Trans. Parallel Distrib. Syst.*, vol. 29, no. 2, pp. 450–465, Feb. 2018.

[8] A. Alabdulatif, I. Khalil, and X. Yi, "Towards secure big data analytic for cloud-enabled applications with fully homomorphic encryption," *J. Parallel Distrib. Comput.*, vol. 137, pp. 192–204, Mar. 2020.

[9] S. M. Pournaghi, M. Bayat, and Y. Farjami, "MedSBA: A novel and secure scheme to share medical data based on blockchain technology and attribute-based encryption," *J. Ambient Intell. Humanized Comput.*, vol. 11, pp. 4613–4641, Jan. 2020.

[10] H. Li, L. Pei, D. Liao, S. Chen, M. Zhang, and D. Xu, "FADB: A fine-grained access control scheme for VANET data based on blockchain," *IEEE Access*, vol. 8, pp. 85190–85203, 2020.

[11] Y. Rahulamathavan, R. C.-W. Phan, M. Rajarajan, S. Misra, and A. Kondoz, "Privacy-preserving blockchain based IoT ecosystem using attribute-based encryption," in *Proc. IEEE Int. Conf. Adv. Netw. Telecommun. Syst. (ANTS)*, Dec. 2017, pp. 1–6.

[12] Y. Zhang, R. H. Deng, J. Shu, K. Yang, and D. Zheng, "TKSE: Trustworthy keyword search over encrypted data with two-side verifiability via blockchain," *IEEE Access*, vol. 6, pp. 31077–31087, 2018.

[13] X. Zhang, C. Xu, H. Wang, Y. Zhang, and S. Wang, "FS-PEKS: Lattice-based forward secure public-key encryption with keyword search for cloud-assisted industrial Internet of Things," *IEEE Trans. Dependable Secure Comput.*, early access, Apr. 30, 2019, doi: 10.1109/TDSC.2019.2914117.

[14] H. Anada, A. Kanaoka, N. Matsuzaki, and Y. Watanabe, "Key-updatable public-key encryption with keyword search (or: How to realize PEKS with efficient key updates for IoT environments)," *Int. J. Inf. Secur.*, vol. 19, no. 1, pp. 15–38, Feb. 2020.

[15] A. Majumder, S. Nath, A. Bhattacharjee, and R. Choudhury, "Trust relationship establishment among multiple cloud service provider," in *Cloud Security: Concepts, Methodologies, Tools, and Applications*. Hershey, PA, USA: IGI Global, 2019, pp. 1548–1576.

[16] S. Alansari, F. Paci, A. Margheri, and V. Sassone, "Privacy-preserving access control in cloud federations," in *Proc. IEEE 10th Int. Conf. Cloud Comput. (CLOUD)*, Jun. 2017, pp. 757–760.

[17] Y. Miao, J. Ma, X. Liu, Q. Jiang, J. Zhang, L. Shen, and Z. Liu, "VCKSM: Verifiable conjunctive keyword search over mobile E-health cloud in shared multi-owner settings," *Pervas. Mobile Comput.*, vol. 40, pp. 205–219, Sep. 2017.

[18] M. K. Khani and M. Noroozian, "Analyzing the effective factors on internal audit quality of health insurance organization of Iran," *Int. J. Academic Res. Accounting, Finance Manage. Sci.*, vol. 8, no. 1, pp. 19–25, Feb. 2018.

[19] T. Gorski, J. Bednarski, and Z. Chaczko, "Blockchain-based renewable energy exchange management system," in *Proc. 26th Int. Conf. Syst. Eng. (ICSEng)*, Dec. 2018, pp. 1–6.

[20] S. Y. Lim, M. M. Kiah, and T. F. Ang, "Security issues and future challenges of cloud service authentication," *Acta Polytechnica Hungarica*, vol. 14, no. 2, pp. 69–89, 2017.

[21] M. S. Ferdous, F. Chowdhury, and M. O. Alassafi, "In search of self-sovereign identity leveraging blockchain technology," *IEEE Access*, vol. 7, pp. 103059–103079, 2019.

[22] B. Houtan, A. S. Hafid, and D. Makrakis, "A survey on blockchain-based self-sovereign patient identity in healthcare," *IEEE Access*, vol. 8, pp. 90478–90494, 2020.

[23] C. Lin, D. He, X. Huang, K.-K.-R. Choo, and A. V. Vasilakos, "BSeIn: A blockchain-based secure mutual authentication with fine-grained access control system for industry 4.0," *J. Netw. Comput. Appl.*, vol. 116, pp. 42–52, Aug. 2018.

[24] S. Wang, Y. Zhang, and Y. Zhang, "A blockchain-based framework for data sharing with fine-grained access control in decentralized storage systems," *IEEE Access*, vol. 6, pp. 38437–38450, 2018.

[25] V. Malamas, P. Kotzanikolaou, T. K. Dasaklis, and M. Burmester, "A hierarchical multi blockchain for fine grained access to medical data," *IEEE Access*, vol. 8, pp. 134393–134412, 2020.

[26] P. Dunphy and F. A. P. Petitcolas, "A first look at identity management schemes on the blockchain," *IEEE Secur. Privacy*, vol. 16, no. 4, pp. 20–29, Jul./Aug. 2018.

[27] P. Jiang, F. Guo, K. Liang, J. Lai, and Q. Wen, "Searchain: Blockchain-based private keyword search in decentralized storage," *Future Gener. Comput. Syst.*, vol. 107, pp. 781–792, Jun. 2020.

[28] J. Alsayed Kassem, S. Sayeed, H. Marco-Gisbert, Z. Pervez, and K. Dahal, "DNS-IdM: A blockchain identity management system to secure personal data sharing in a network," *Appl. Sci.*, vol. 9, no. 15, p. 2953, Jul. 2019.
[29] Apple Implements Privacy. *Nutrition Label*. For Apps. [Online]. Available: https://www.mondaq.com/unitedstates/privacy-protection/1015152/apple-implements-privacy-nutrition-label-for-apps
[30] G.-J. Ra and I.-Y. Lee, "A study on hybrid blockchain-based XGS (XOR global State) injection technology for efficient contents modification and deletion," in *Proc. 6th Int. Conf. Softw. Defined Syst. (SDS)*, Jun. 2019, pp. 300–305.
[31] S. Dziembowski, S. Faust, and K. Hostáková, "General state channel networks," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, Oct. 2018, pp. 949–966.
[32] 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.*, early access, Jun. 17, 2020, doi: 10.1109/TCC.2019.2923222.
[33] A. Zhang and X. Lin, "Towards secure and privacy-preserving data sharing in E-health systems via consortium blockchain," *J. Med. Syst.*, vol. 42, no. 8, pp. 1–18, Aug. 2018.
[34] R. R. Farashahi, B. Schoenmakers, and A. Sidorenko, "Efficient pseudorandom generators based on the DDH assumption," in *Proc. Int. Workshop Public Key Cryptogr.* Berlin, Germany: Springer, 2007, pp. 426–441.
[35] X. Yang and W. Li, "A zero-knowledge-proof-based digital identity management scheme in blockchain," *Comput. Secur.*, vol. 99, Dec. 2020, Art. no. 102050.
[36] A. R. Pedrosa and G. Pau, "ChargeltUp: On blockchain-based technologies for autonomous vehicles," in *Proc. 1st Workshop Cryptocurrencies Blockchains Distrib. Syst.*, 2018, pp. 87–92.
[37] A. W. Dent and S. D. Galbraith, "Hidden pairings and trapdoor DDH groups," in *Proc. Int. Algorithmic Number Theory Symp.* Berlin, Germany: Springer, 2006, pp. 436–451.

**GYEONGJIN RA** received the B.S. and M.S. degrees in computer software and engineering from Soonchunhyang University, Asan, South Korea, in February 2016 and February 2018, respectively, where she is currently pursuing the Ph.D. degree with the Department of Software Convergence. Her research interests include security and privacy protection, applied cryptography, and blockchain.

**DONGHYUN KIM** (Senior Member, IEEE) received the B.S. degree in electronic and computer engineering, and the M.S. degree in computer science and engineering from Hanyang University, Ansan, South Korea, in February 2003 and February 2005, respectively, and the Ph.D. degree in computer science from the University of Texas at Dallas, Richardson, TX, USA, in May 2010. He is currently an Assistant Professor with the Department of Computer Science, Georgia State University (GSU), Atlanta, GA, USA. He is a Senior Member of ACM. He has served as a TPC co-chair for several international conferences, most recently IPCCC 2020 and COCOON 2020.

**DAEHEE SEO** (Member, IEEE) received the B.S. degree in electronic and electrical engineering from Dongshin University, Naju, South Korea, in February 2001, and the M.S. degree in computer science and engineering and the Ph.D. degree in computer science from Soonchunhyang University, Choognam, South Korea, in February 2003 and February 2006, respectively. He is currently an Assistant Professor with the Faculty of Artificial Intelligence and Data Engineering, SangMyung University (SMU), Seoul, South Korea.

**IMYEONG LEE** (Member, IEEE) received the B.S. degree in electronic engineering from Hongik University, Seoul, in 1981, and the M.S. and Ph.D. degrees in information and communication engineering from Osaka University, Osaka, Japan, in 1986 and 1989, respectively. He is currently a Professor with the Department of Computer Software Engineering, Soonchunhyang University (SCH), Asan, South Korea. His research interests include information security, cryptographic protocol, information theory, and data communication.