Indian J. Res. Found., (2019) 8, 31-38. ISSN 2454-6577

INDIAN JOURNAL OF RESEARCH FOUNDATION
www.indianjournal.net

RESEARCH ARTICLE

Secured cloud storage system based on privacy preserving weighted similarity keyword search scheme

Shaniya Shajahan, M. Biruntha[a]

Department of Computer Science and Engineering, Cape Institute of Technology, Levenjipuram, India

Received 10 March 2019; Accepted 15 April 2019
Available online 17 April 2019

Abstract

The cloud has become an important platform for data storage and processing. It centralizes essentially unlimited resources (e.g., storage capacity) and delivers elastic services to end users. In proposed method, study the problem of keyword search with access control over encrypted data in cloud computing. First propose a scalable framework where user can use his attribute values and a search query to locally derive a search capability, and a file can be retrieved only when its keywords match the query and the user's attribute values can pass the policy check. Using this framework, we propose a novel scheme called KSAC, which enables Keyword Search with Access Control over encrypted data. KSAC utilizes a recent cryptographic primitive called HPE to enforce fine-grained access control and perform multi-field query search. Meanwhile, it also supports the search capability deviation, and achieves efficient access policy update as well as keyword update without compromising data privacy. To enhance the privacy, KSAC also plants noises in the query to hide users' access privileges. Intensive evaluations on real-world dataset are conducted to validate the applicability of the proposed scheme and demonstrate its protection for user's access privilege.

However, the security of the outsourced data has become a major concern. For privacy concerns, searchable encryption, which supports searching over encrypted data, has been proposed and developed rapidly in secure Boolean search and similarity search. However, different users may have different requirements on their queries, which mean different weighted searches. This problem can be solved perfectly in the plaintext domain, but hard to be addressed over encrypted data. In this study, we use locality-sensitive hashing (LSH) and searchable symmetric encryption (SSE) to deal with a privacy preserving weighted similarity search. In the authors' scheme, data users can generate a search request and set the weight for each attribute according to their requirements. We treat the LSH values as keywords and mix them into the framework of SSE. We use homomorphic encryption to securely address the weight problem and return the top-k data without revealing any weight information of data users. Extensive experiments on actual datasets showed that the scheme is extremely effective and efficient.

Keywords

Searchable Encryption (*SE*),

Hierarchical Primitive Encryption (*HPE*),

Key Based Access Control (*KBAC*),

Advanced Encryption Standard (*AES*).

*Corresponding author Tel. +91 8607672921
E-mail : vidishabv@gmail.com

Published by GM SOFTWARE

Introduction

The cloud has become an important platform for data storage and processing. Examples of cloud services include online file storage, social networking sites, here mail, and online business applications. It centralizes essentially unlimited resources and delivers elastic services to end users without performing their own system management and upfront equipment acquisitions. However, data confidentiality protection (to hide the plaintext against the cloud server and other unauthorized users) and data access control (to grant user's access privilege) are usually required so that data owners can confidently store their data onto the cloud. Encryption is a commonly used method to preserve data confidentiality by storing cipher text in the cloud.

However, it may make traditional approaches designed for plaintext keyword search inapplicable. Aiming at enabling secure and efficient search over encrypted data, Searchable Encryption (SE) receives increasingly more attentions in recent years, in which a query is encrypted as a search capability and a cloud server will return files matching the capability without having to know the keywords both in the capability and in file's encrypted index.

However, most of existing SE schemes assume that user can access all the shared files. Such assumption does not hold in the cloud environment where users are actually granted different access permissions according to the access-control policy determined by data owners. Therefore, it is important to study how to efficiently enforce the access control policy when searching over encrypted data.

There have been a number of works on access control over encrypted data. These works can be categorized into two groups, key-based access control (KBAC) and attribute-based access control (ABAC). KBAC usually assigns each file's decryption key directly to authorized users. When a user receives increasing number of such keys accumulated, its load on the management of the keys can be too high. To reduce the load, ABAC attaches a set of attribute values to a user (or a file) and designs access policy for a file (or a user, respectively). A file can be accessed if and only if the attribute values satisfy the access policy. The access keys (e.g., the decryption keys in KBAC and the keys to represent attribute values in ABAC) are usually required to be kept secretly to prevent data security from being compromised. Therefore, the conventional way to perform encrypted search with access control is to conduct the search operations at the cloud server to take advantage of its large computation power and leave the enforcement of access control at users' machines to keep their access keys from disclosed. This separation of search and access control enforcement could lead to performance degradation, especially when users are assigned with different access permissions to search different encrypted cloud data.

Existing techniques

Conjunctive, subset, and range queries on encrypted data

The proposed method uses public-key systems that support comparison queries on encrypted data as well as more general queries such as subset queries. These systems support arbitrary conjunctive queries without leaking information on individual conjuncts. No prior Searchable Symmetric Encryption (SSE) based privacy-preserving conjunctive query processing scheme satisfies the three requirements of adaptive security, efficient query processing, and scalable index size. In this paper, we propose the first privacy preserving conjunctive query processing scheme that satisfies the above requirements.

To achieve adaptive security, we propose an Indistinguishable Bloom Filter (IBF) data structure for indexing. To achieve efficient query processing and structure indistinguishability, we propose a highly balanced binary tree data structure called Indistinguishable Binary Tree (IBtree). To optimize searching efficiency, we propose a traversal width minimization algorithm and a traversal depth minimization algorithm. To achieve scalable and compact index size, we propose an IBtree space compression algorithm to remove redundant information in IBFs. We formally prove that our scheme is adaptive secure using a random oracle model.

The key contribution of this paper is on achieving conjunctive query processing with both strong privacy guarantee and practical efficiency in terms of both speed and space.

Advantages

High accuracy

Disadvantages

Time consuming method

Preferred keyword search over encrypted data in cloud computing

Cloud computing cuts down large capital outlays in facilities purchase and eliminates complex system management for users. To protect data confidentiality in cloud utilization, sensitive data are usually stored in encrypted form, making traditional search service on plaintext inapplicable. Thus, enabling keyword search over encrypted data becomes a paramount urgency. Given massive data users with various search preferences, it becomes necessary to support preferred keyword search and output the data files in the order of the user's preference. In this paper, for the first time, we investigate the challenging problem of preferred keyword search over encrypted data (PSED).

We first establish a set of privacy requirements and utilize the appearance frequency of each keyword to serve as its "weight". A preference pre processing mechanism is then explored to ensure that the search result will faithfully respect the user's preference and the Lagrange polynomial is introduced to express the user's preference formula. We further represent keyword weights of each file by using vectors, convert the preference polynomial into the vector form, and securely calculate their inner products to quantitatively characterize the relevance measure between data files and a query. Finally, an extensive performance evaluation demonstrates the proposed scheme can achieve acceptable efficiency.

First specify a set of privacy requirements and use the appearance frequency of each keyword to a file to act as its weight. A flexible search query (e.g., the query over multiple keyword fields) is converted into polynomial form and the Lagrange polynomial is utilized to characterize the user's preference query. Then we convert the search polynomial and the preference polynomial into vector forms, and propose a secure inner product computation mechanism to capture the correlation of files to the query.

Advantages

The proposed scheme should introduce lightweight operations to the user/owner, and promise the search efficiency.

Flexible search query with preferences

Disadvantages

Difficult method

Privacy preserving EHR system using attribute-based infrastructure

Secure management of Electronic Health Records (EHR) in a distributed computing environment such as cloud computing where computing resources including storage is provided by a third party service provider is a challenging task. In this paper, we explore techniques which guarantees security and privacy of medical data stored in the cloud. We show how new primitives in attribute-based cryptography can be used to construct a secure and privacy-preserving EHR system that enables patients to share their data among healthcare providers in a flexible, dynamic and scalable manner. In this paper, consider a secure design for a patient centric EHR management system where data is saved in a storage provided by a cloud provider.

We assume the cloud provides a reliable storage for data but the stored data can be seen (and copied) by the cloud provider. This means that it is the responsibility of the user to provide mechanisms that ensure security and privacy of their information. Users store their data in encrypted form in the cloud and grant access to portion of the data in accordance with the requesters' identity information. The storage provider will not be able to see data, or associated metadata, therefore confidentiality and privacy of data will be guaranteed. The scheme presented can also be applied to other general security-sensitive database applications.

Advantages

Attribute-based cryptographic primitives provide flexible policies which can be used to build secure infrastructure for designing privacy preserving electronic health record system.

Disadvantages

High computational cost

2.4 Public Key Encryption with keyword Search

This paper presents a public key encryption with keyword search (PEKS) and gave two constructions. Constructing a PEKS is related to Identity Based Encryption (IBE), though PEKS seems to be harder to construct. The proposed method gives the concept of a public key encryption with keyword search (PEKS) and gave two constructions.

Constructing a PEKS is related to Identity Based Encryption (IBE), though PEKS seems to be harder to construct. We showed that PEKS implies Identity Based Encryption, but the converse is currently an open problem. Our constructions for PEKS are based on recent IBE constructions. We are able to prove security by exploiting extra properties of these schemes. The public key encryption with keyword search (PEKS) scheme, proposed by Boneh, Di Crescenzo, Ostrovsky and Persiano, enables one to search for encrypted keywords without compromising the security of the original data.

In this paper, we address two important issues of a PEKS scheme, "removing secure channel" and "refreshing keywords", which have not been considered in Boneh et al.'s paper. We point out the inefficiency of the original PEKS scheme due to the use of the secure channel. We resolve this problem by constructing an efficient PEKS scheme that removes a secure channel. We then argue that care must be taken when keywords are used frequently in the PEKS scheme as this situation might contradict the security of PEKS.

ADVANTAGES:

Improve security

DISADVANTAGES:

Low accuracy

Architectures for an Event notification service scalable to wide-area networks

A wide range of software systems are designed to operate in a reactive manner. In such systems, the high-level control flow is not explicitly programmed; instead it is driven by the occurrence of *events*. These systems realize their functionality by performing some actions in response to events, possibly using the information associated with the stimulating events. Examples of reactive systems are integrated development environments, work-flow and process analysis systems, graphical user interfaces, network management systems, software deployment systems and security monitors.

The general case of "on-line" input, the concept of *event* is a good modeling and design abstraction. Similarly, the same abstraction is useful for those components that, although not necessarily functioning in an asynchronous way, are integrated by means of some communication mechanisms that introduce a synchronicity in their interactions. The other benefit of adopting an event-based style is that it requires only a loose coupling for the integration of heterogeneous components. Components do not need to export interfaces to be accessed by other components. Components can request some services without addressing a specific server component and, to a certain extent, components can interoperate even if they have been designed and developed separately without any mutual knowledge.

The idea of integrating software components by means of a common event service seems to be very promising especially for those distributed applications that are deployed on a wide-area network such as the Internet. For one thing, the vast number of available information sources offers a great deal of opportunities for the development of new applications. New classes of wide-scale event-driven applications can be devised including stock market analysis tools, efficient news and mailing systems, data mining tools, and indexing tools. Also, many existing applications that are already designed to exploit event based infrastructures can be proficiently integrated at a much higher scale thanks to the "global" connectivity provided by the network. For example, work-flow systems can be federated for companies that have multiple distributed branches or even across corporate boundaries, or else software deployment systems can connect software producers and consumers through the Internet. In general, the synchronicity, the heterogeneity, and the high degree of loose coupling that characterize wide-area networks suggest that a wide-scale event service would be a good integration infrastructure for existing systems and for new applications.

Disadvantages:

Redundant connections are that special algorithms must be implemented to avoid cycles and to choose the best paths.

Messages will carry a time-to- live counter, and routes will be established according to minimal spanning trees.

Universal cross - cloud communication

Integration of applications, data-centers, and programming abstractions in the cloud-of-clouds poses many challenges to system engineers. Different cloud providers offer different communication abstractions and applications exhibit different communication patterns. By abstracting from hardware addresses and lowered - level communication, the publish/subscribe paradigm seems like an adequate abstraction for supporting communication

across clouds, as it supports many-to-many communication between publishers and subscribers, of which one-to-one or one-to-many can be veered as special cases. In particular, content-based publish/subscribe (CPS) systems provide an expressive abstraction that matches here all with the key-value pair model of many established cloud storage and computing systems, and decentralized overlay-based CPS implementations scale up here all. However, CPS systems perform poorly at small scale, e.g., one-to-one or one-to-many communication. This holds especially for multi-send scenarios which here refer to as entourages that may range from a channel between a publisher and a single subscriber to a broadcast between a publisher and a handful of subscribers. These scenarios are common in cloud computing, where cheap hardware is exploited for parallelism (efficiency) and redundancy (fault-tolerance). With CPS, multi-send messages go over several hops before their destinations are even identified via predicate matching, resulting in increased latency, especially when destinations are located in different data-centers or zones. Topic-based publish/subscribe (TPS) systems support communication at small scale more efficiently, but still route messages over multiple hops and inversely lack the flexibility of CPS systems. In this, CPS protocols for cloud-of-clouds communication that can dynamically identify entourages of publishers and corresponding subscribers. The CPS protocols dynamically connect the publishers with their entourages through transmit messages from a publisher to its corresponding subscribers with low latency. This experiments show that our protocols make CPS abstraction viable and beneficial for many applications. To introduce a CPS system named Atmosphere that leverages out CPS protocols and illustrate how Atmosphere has allohered us to implement, with little effort, versions of the popular HDFS and Zoo Keeper systems which operate efficiently across data-centers.

Disadvantages:

There is no standardize security making it more granular.

There is no universal service catalog (at the federal government level) built to support portability

There are no updated procurement processes and policies to enable migration to and between clouds.

Building a reliable and high - performance content - based publish/subscribe system

Provisioning reliability in a high -performance content – based publish/subscribe system is a challenging problem. The inherent complexity of content-based routing makes message loss detection and recovery, and network state recovery extremely complicated. Existing proposals either try to reduce the complexity of handling failures in traditional network architecture, which only partially address the problem, or rely on robust network architectures that can gracefully tolerate failures, but perform less efficiently than the traditional architectures. In this, a hybrid network architecture for reliable and high-performance content-based publish/subscribe. Two overlay networks, a high-performance one with moderate fault tolerance and a highly-robust one with sufficient performance, work together to guarantee the performance of normal operations and reliability in the presence of failures. The design exploits the fact that, in a high-performance content-based publish/subscribe system, subscriptions are broadcast to all brokers, to facilitate efficient backup routing when failures occur, which incurs a minimal overhead. Per-hop reliability is used to gracefully detect and recover lost messages that are caused by transit errors. Two backup routing methods based on DHT routing are proposed. Extensive simulation experiments are conducted. The results demonstrate the superior performance of our system compared to other state-of-the-art proposals.

Disadvantage:

Inflexible Semantic coupling.

Message Delivery.

Efficient event routing in content-based publish - subscribe service networks

Efficient event delivery in a content-based publish/subscribe system has been a challenging problem. Existing group communication solutions, such as IP multicast or application-level multicast techniques, are not readily applicable due to the highly heterogeneous communication pattern in such systems. First explore the design space of event routing strategies for content-based publish/subscribe systems. Two major existing approaches are studied: filter-hosed approach, which performs content-based filtering on intermediate routing servers to dynamically guide routing decisions, and multicast-based approach, which delivers events through a few high-quality multicast groups that are pre-constructed to approximately match user interests. These approaches have different trade-offs in the routing quality achieved and the implementation cost and system load generated. To present a new routing scheme called Kyra that carefully balance

these trade-offs. Kyra combines the advantages of content-based filtering and event-space partitioning in the existing approaches to achieve better overall routing efficiency. To use detailed simulations to evaluate Kyra and compare it with existing approaches. The results demonstrate the effectiveness of Kyra in achieving high network efficiency, reducing implementation cost and balancing system load across the publish - subscribe service network.

Disadvantage:

Topic-based model is the very limited expressiveness it offers to subscribers.

Matching algorithms for XML-based language requires heavier processing.

Existing system

A cloud data sharing system consisting of four entities, i.e., data owners, authority, data users and cloud server. Data owners create data files, design the encrypted indices containing both keywords and access policy for each file, and upload the encrypted files along with the indices to the cloud server. Authority is responsible to authenticate user's identity. It issues a set of keys as a credential to represent user's attribute values. Data user generates a search capability according to his credential and a search query, and submits it to the cloud server for file retrieval. The cloud server stores the encrypted data and performs search when receiving search capabilities from users. Access control is a security technique that can be used to regulate who or what can view or use resources in a computing environment.

The typical participants of a secure search system in the cloud involve the cloud server, the data owner ,and the data user , as shown in Figure . The data owner outsources the encrypted dataset and the corresponding secure indexes to the cloud server, where data can be encrypted using any secure encryption technique, such as Advanced Encryption Standard (AES), while the secure index is generated by some particular search-enabled encryption techniques. When a data user wants to query the outsourced dataset hosted on the cloud server, he/she first either generates a trapdoor with the keyword of interest (applied to most PKC-based search schemes), or requests such trapdoor by sending a set of intended keywords to the data owner (in the case of SKC-based search schemes). In the latter case, upon receiving the trapdoor generation request, the data owner constructs the trapdoor, and return it to the user. Then the data user submits the trapdoor to the cloud server. The cloud server will execute the search program with the trapdoor as the input, the search results will be sent back to the user. Note that here we assume there is pre-existing security context between each user and the data owner thus authentication between user and data owner is already in place. The trapdoors can be requested and returned through a secure channel. The management of the decryption keys of the returned files is an orthogonal problem and has been studied separately.Search can be based on certain search criteria and the results be ranked based on certain ranking criteria.

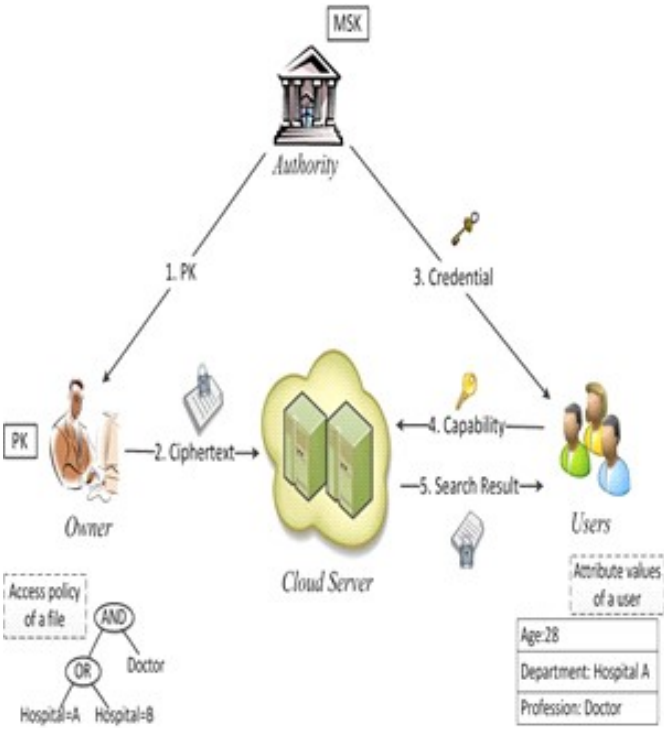

Fig 1 System architecture

Proposed techniques

The proposed method introduces a keyword search with access control in encrypted cloud data, where the data are tied with access policies and keywords, and allow the searches by multiple users whose access privileges are specified. First, we propose a scalable framework that integrates multi-field keyword search with fine-grained access control. In the framework, every user authenticated by an authority obtains a set of keys called credential to represent his attribute values. Each file stored in the cloud is attached with an encrypted index to label the keywords and specify the access policy.

Each user can use his credential and a search query to locally generate a search capability, and submit it to the cloud server who then performs search and access control

in an interleaving manner. Second, to enable such a framework, we make a novel use of Hierarchical Predicate Encryption (HPE), to realize the derivation of search capability from credential and a search query. Based on HPE, we propose our scheme named as KSAC, which enables the service of both the query search and access control over multiple fields.

There are three entities in a traditional SE system model. However, to perform the secure ranked search operation in our scheme, added another server in our SSE system, as shown in Fig. 2. The system model consists of data owner, data user, cloud server A, and cloud server B. Considering a realistic scenario, the data owner intends to outsource the data to the cloud server so that he can enjoy the high quality service provided by the cloud in hosting his data. Since cloud servers are always employed in an untrusted environment provided by third-party service providers, the data owner must encrypt his or her data to avoid data leakage and unauthorized access. To make the query more convenient and efficient, the data owner will build an encrypted index and outsource it to the cloud server along with the encrypted data. When a data user wants to perform a search operation, first, he or she will generate a trapdoor. Usually, a data user has some requirements about the attributes in the query to get a more accurate result. The data user may consider that one of the attributes is the most important or some attributes are more important than others. So, the user will assign the weights to the trapdoor and send it to the cloud server. The weight information related to the trapdoor should be encrypted to prevent information leakage. As long as the cloud server receives the trapdoor, it will perform the match algorithm and compute the relevant scores, which are encrypted against the relevance information leakage. However, encrypted relevant scores cannot be compared. Hence, add another cloud server to decrypt and rank the relevant scores. Finally, the cloud server would return the most k relevant data records as a result of the query.

Index privacy: Two aspects of index privacy should be preserved, i.e. (i) the cloud server cannot learn the content of the index because the index directly reflects the content of the data records and (ii) the cloud server cannot deduce any content of the data records by analyzing the encrypted index.

Trapdoor privacy: The trapdoor is generated by a searched data item and can require the cloud server to perform a search operation. The trapdoor will preserve the data users' query information against the cloud server. In addition, the cloud server cannot tell whether a query is a duplicate of an earlier query because it will not receive two identical trapdoors even if two queries are the same.

Relevance score privacy: The relevance score is used to measure the similarity between the search request and data records stored in the cloud. Given the relevance scores, the cloud server that stores the original dataset will learn nothing about the relevance information. The cloud server that owns the secret key can decrypt the scores, but it cannot obtain the original data records.

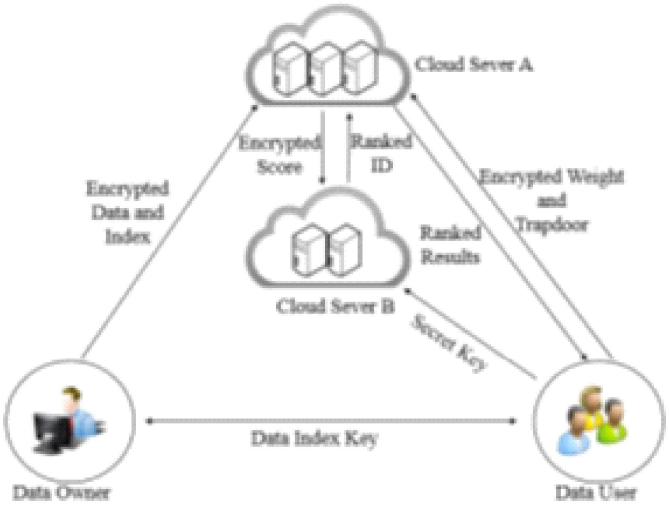

Fig 2 System model

Implementation

Data owner

The data owner is the entity which generates and encrypts the data and uploads them to the cloud server. It can be either an organization or an individual. To use the service, the data owner uses its application which consists of a data processor for uploading new contents to the cloud. It encrypts the data and metadata with a cryptographic scheme that enables searching capability.

User

This entity is also a subscriber to the cloud storage which sends encrypted queries to the cloud service provider to search for a specific encrypted data. There may be more than one data user in the system and in some scenario, the data owner and the data user might be the same entity.

Authority

Authority get login by using their username and password. The authority can view users and activate and deactivate users. The authority also view owner details and send response

Cloud server A

This entity provides the data storage and retrieval service to the subscribers. The cloud service provider consists of cloud data server and cloud service manager. The first entity is used to store the outsourced encrypted data whereas the latter one is used for data management in the cloud. Upon receiving the encrypted search queries from the data user, the cloud service provider tests on the encrypted queries and encrypted metadata in the cloud storage. The encrypted data that satisfies the search criteria is retrieved and sent back to the data owner upon completion of the test. The cloud service provider should not learn any information from the operation.

Cloud server B

The cloud servers in an untrusted cloud environment can be attacked by outside adversaries who can obtain all of the stored data. The two cloud servers are 'curious but honest', which means that they are curious about obtaining the content of the encrypted data, but they follow the designed protocol exactly. Also, we assume that cloud server A will not collude with cloud server B and that cloud server B will not give incorrect answers to cloud server A.

Conclusion

In this paper, we propose a scalable framework that allows users to locally derive the search capability by utilizing both their credentials and a search query. We then utilize HPE to realize this framework and present KSAC. KSAC realizes the fine-grained access control and multi-field keyword search, enables efficient update of both access policy and keywords, and protects user's access privacy.

We explored the problem of SE in the untrusted cloud computing environment. Different from prior works, our scheme enables a secure and convenient weighted similarity search. Our design starts with two building blocks, LSH and SSE. Since our target was a reasonable requirement, we transformed a traditional LSH-based index. To achieve secure relevance scores computing and comparing, we identified another cloud server and used the Paillier cryptosystem. By adapting the security framework of SSE, we carefully identified any information leakage and proved the security of our scheme.

In our future work, we plan to improve our scheme to support similarity search for multiple data owners.

References

[1] J. Shu, Z. Shen, W. Xue. "Shield: A stackable secure storage system for file sharing in public storage." *J. Parallel Distr. Com.*, 74(9), 2872–2883, 2014.

[2] M. A Tinghuai, Z. Jinjuan, T.Meili, T. Yuan, A. Abdullah, A. Mznah, L. Sungyoung, "Social network and tag sources based augmenting collaborative recommender system." *IEICE Trans. Inform. Systems*, 98(4), 902–910, 2015.

[3] Y. Ren, J. Shen, J. Wang, J. Han, S. Lee. 'Mutual verifiable provable data auditing in public cloud storage." *J. Internet Technol*., 16(2), 318, 2015.

[4] P. Eugster, J. Stephen, "Universal cross-cloud communication," *IEEE T. Cloud. Comput.,* 2014.

[5] J. Shu, Z. Shen, W. Xue, Y. Fu. "Secure storage system and key technologies." *In Design Automation Conference 2013, 18th Asia and South Pacific*, 376–383, 2013.

[6] Y. Chang, M. Mitzenmacher. "Privacy preserving keyword searches on remote encrypted data." *In Applied Cryptography and Network Security*, 2005.

[7] D. Boneh, G. D. Crescenzo, R. Ostrovsky, G. Persiano., "Public key encryption with keyword search." *In Proc. Eurocrypt*, 506–522, 2004.

[8] E. Shi, J. Bethencourt, T. Chan, D. Song, A. Perrig. "Multi-dimensional range query over encrypted data." *In Proc. of IEEE Symposium on Security and Privacy*., 2007.

[9] C. Wang, N. Cao, J. Li, K. Ren, W. Lou. "Secure ranked keyword search over encrypted cloud data." *In Proc. IEEE ICDCS,* 2010.

[10] D. Boneh, B. Waters. "Conjunctive, subset, and range queries on encrypted data. *In Proc. TCC*, 2007.

[11] C. Dong, G. Russello, N. Dulay. "Shared and searchable encrypted data for untrusted servers." *J. Comp. Sec.*, 19(3), 367–397, 2011.

[12] J. Li, Q. Wang, C.Wang, N. Cao, K. Ren, W. Lou. "Fuzzy keyword search over encrypted data in cloud computing." *In. Proc. IEEE INFOCOM,* 2010.