A Survey on Secure Data Sharing In Public Cloud Using Mediated Certificateless Encryption

Savitha N#1, Dr.S.V.M.G.Bavithiraja#2, Shanthi T#3

1PG Scholar, 2Professor, 3PG Scholar
Dept of CSE,Sri Eshwar College of Engineering,
Coimbatore, India.

savitharcp@gmail.com

Bavithiraja.svmg@sece.ac.in

shanthi21294@gmail.com

Abstract- A mediated certificateless encryption scheme without pairing operations for securely sharing sensitive information in public clouds. Mediated certificateless public key encryption (mCL-PKE) solves the key escrow problem in identity based encryption and certificate revocation problem in public key cryptography. mCL-PKE scheme is applied to construct a practical solution to the problem of sharing sensitive information in public clouds. In this system, the data owner encrypts the sensitive data using the cloud generated users public keys based on its access control policies and uploads the encrypted data to the cloud. Upon successful authorization, the cloud partially decrypts the encrypted data for the users. The users subsequently fully decrypt the partially decrypted data using their private keys. The confidentiality of the content and the keys is preserved with respect to the cloud, because the cloud cannot fully decrypt the information. To improve the efficiency of encryption at the data owner, mCL-PKE scheme is used and the overall cloud based systems evaluate its security and performance.

Keywords: Cloud Computing, Certificateless Encryption, Public Key, Private key, Access Control, Confidentiality.

INTRODUCTION

Cloud computing has been envisioned as the next-generation architecture of IT enterprise. In contrast to traditional solutions, where the IT services are under proper physical, logical and personnel controls, cloud computing moves the application software and databases to the large data centers, where the management of the data and services may not be fully trustworthy. This unique attribute, however, poses many new security challenges which have not been well understood. On cloud data storage security, which has always been an important aspect of quality of service.

To ensure the correctness of users data in the cloud, an effective and flexible distributed scheme with two salient features, opposing to its predecessors is used. By utilizing the homomorphic token with distributed verification of erasure-coded data, our scheme achieves the integration of storage correctness insurance and data error localization, i.e., the identification of misbehaving server. Unlike most prior works, the new scheme further supports secure and efficient dynamic operations on data blocks, including: data update, delete and append. Extensive security and performance analysis shows that the proposed scheme is highly efficient and resilient against Byzantine failure, malicious data modification attack, and even server colluding attacks.

From the perspective of data security, which has always been an important aspect of quality of service, Cloud Computing inevitably poses new challenging security threats for number of reasons. Firstly, traditional cryptographic primitives for the purpose of data security protection can not be directly adopted due to the users' loss control of data under Cloud Computing. Therefore, verification of correct data storage in the cloud must be conducted without explicit knowledge of the whole data. Considering various kinds of data for each user stored in the cloud and the demand of long term continuous assurance of their data safety, the problem of verifying correctness of data storage in the cloud becomes even more challenging.

Secondly, Cloud Computing is not just a third party data warehouse. The data stored in the cloud may be frequently updated by the users, including insertion, deletion, modification, appending, reordering, etc. To ensure storage correctness under dynamic data update is hence of paramount importance. These techniques, while can be useful to ensure the storage correctness without having users possessing data, cannot address all the security threats in cloud data storage, since they are all focusing on single server scenario and most of them do not consider dynamic data operations.

As an complementary approach, researchers have also proposed distributed protocols for ensuring storage correctness across multiple servers or peers. Again, none of these distributed schemes is aware of dynamic data operations. As a result, their applicability in cloud data storage can be drastically limited.

2. Identity Based Encryption

A conventional open key cryptosystem requires a trusted Certificate Authority (CA) to issue computerized testaments that tie clients to their open keys. Since the CA needs to create its own mark on every client's open key and deal with every client's declaration, the general endorsement administration is exceptionally costly and complex. To address such inadequacy, Identity-Based Public Key Cryptosystem (IBPKC) was presented. IBC depends on a trusted outsider called the Private Key Generator (PKG). Before operation can start, the PKG must create an open/private key pair and make pkPKG accessible to clients of its administrations. These keys are known as the "ace" open key and ace private key, separately.

The procedure of encryption and decoding continues as takes after:

1) Alice gets ready plaintext message M for Bob. She uses Bob's character IDBob and the PKG's open key pkPKG to encode M, getting ciphertext message C. Alice then sends C to Bob. Take note of that IDBob and pkPKG were both definitely known to Alice before starting the encryption handle, so she requires no earlier coordination or arrangement on Bob's part to encode a message for him.

2) Weave gets C from Alice. In many executions itis expected that C accompanies plaintext guidelines for reaching the PKG to get the private key required to decode it. Bounce confirms with the PKG, basically sending it adequate evidence that IDBob has a place with him, whereupon the PKG transmits Bob's private key skIDBob to him over a protected channel. On the off chance that IDBob depended on an email address, for instance, the PKG could send a nonce to this email address, the effective return of which may give an adequate level of confirmation that the proprietor of IDBob was the person who had reached the PKG. This nonce could be returned by means of an SSL hypertext interface which gave Bob a protected connection for downloading his private key. For a larger amount of affirmation, Bob could be required to present his qualifications face to face and get a smaller circle containing skIDBob.

3) Weave decodes C utilizing his private key skIDBob to recoup plaintext message M.

But said scheme suffers from the key escrow problem as the key generation server learns the private keys of all users and thus it can decrypt documents of any users hence exposing the security if attackers attack the server can get all information for decrypting document of data owner.

3. Data Dynamics For Storage Security

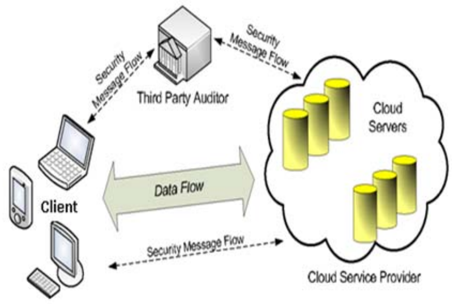
The application software and databases to the centralized large data centers, where the management of the data and services may not be fully trustworthy. This unique paradigm brings about many new security challenges, which have not been well understood. This work studies the problem of ensuring the integrity of data storage in Cloud Computing. In particular, consider the task of allowing a third party auditor (TPA), on behalf of the cloud client, to verify the integrity of the dynamic data stored in the cloud.

The introduction of TPA eliminates the involvement of the client through the auditing of whether, data stored in the cloud is indeed intact. The support for data dynamics via the most general forms of data operation, such as block modification, insertion and deletion, is also a significant step toward practicality, since services in Cloud Computing are not limited to archive or backup data only. While prior works on ensuring remote data integrity often lacks the support of either public auditability or dynamic data operations, this paper achieves both.

At first identify the difficulties and potential security problems of direct extensions with fully dynamic data updates from prior works and then show how to construct an elegant verification scheme for the seamless integration of these two salient features in our design. In particular, to achieve efficient data dynamics, improve the existing proof of storage models by manipulating block tag authentication. To support efficient handling of multiple auditing tasks, we further explore the technique of signature to extend our main result into a multi user setting, where TPA can perform multiple auditing tasks simultaneously. Extensive security and performance analysis show that the proposed schemes are highly efficient and provably secure.

4. Secure and Dependable Storage Services

Cloud storage enables users to remotely store their data and enjoy the on demand high quality cloud applications without the burden of local hardware and software management. Though the benefits are clear, such a service is also relinquishing users physical possession of their outsourced data, which inevitably poses new security risks toward the correctness of the data in cloud. In order to address this new problem and further achieve a secure and dependable cloud storage service, a flexible distributed storage integrity auditing mechanism, utilizing the homomorphic token and distributed erasure-coded data is used. The proposed design allows users to audit the cloud storage with very lightweight communication and computation cost.



The architecture of cloud data storage service

The auditing result not only ensures strong cloud storage correctness guarantee, but also simultaneously achieves fast data error localization, i.e., the identification of misbehaving server. Considering the cloud data are dynamic in nature, the proposed design further supports secure and efficient dynamic operations on outsourced data, including block modification, deletion, and append. Analysis shows the proposed scheme is highly efficient and resilient against Byzantine failure, malicious data modification attack, and even server colluding attacks.

Compared to many of its predecessors, which only provide binary results about the storage state across the distributed servers, the challenge response protocol provides the localization of data error. Unlike most prior works for ensuring remote data integrity, the new scheme supports secure and efficient dynamic operations on data blocks, including: update, delete and append. Extensive security and performance analysis shows that the proposed scheme is highly efficient and resilient against Byzantine failure, malicious data modification attack, and even server colluding attacks.

5. Provable Data Possession

With cloud storage services, it is commonplace for data to be not only stored in the cloud, but also shared across multiple users. However, public auditing for such shared data while preserving identity privacy remains tough. Privacy preserving mechanism allows public auditing on shared data stored in the cloud. In particular, ring signatures to compute the verification information needed to audit the integrity of shared data. With this mechanism, the identity of the signer on each block in shared data is kept private from a third party auditor (TPA), who is still able to verify the integrity of shared data without retrieving the entire file.

Experimental results demonstrate the effectiveness and efficiency when auditing shared data. The first provable data possession (PDP) mechanism to perform public auditing is designed to check the correctness of data stored in an un trusted server, without retrieving the entire data. It is designed to construct a public auditing mechanism for cloud data, so that during public auditing, the content of private data belonging to a personal user is not disclosed to the third party auditor.

To audit the integrity of shared data in the cloud with static groups, the group is pre-defined before shared data is created in the cloud and the membership of users in the group is not changed during data sharing. The original user is responsible for deciding who is able to share her data before outsourcing data to the cloud. Another problem is how to audit the integrity of shared data in the cloud with dynamic groups a new user can be added into the group and an existing group member can be revoked during data sharing while still preserving identity privacy.

6. Fuzzy Keyword Search

Cloud Computing becomes prevalent, more and more sensitive information are being centralized into the cloud. Although traditional searchable encryption schemes allow a user to securely search over encrypted data through keywords and selectively retrieve files of interest, these techniques support only exact keyword search. Formalize and solve the problem of effective fuzzy keyword search over encrypted cloud data while maintaining keyword privacy is done here. Fuzzy keyword search greatly enhances system usability by returning the matching files when users searching inputs exactly match the predefined keywords or the closest possible matching files based on keyword similarity semantics, when exact match fails.

To quantify keywords similarity and develop two advanced techniques on constructing fuzzy keyword sets, which achieve optimized storage and representation overheads. A brand new symbol based tree traverse searching scheme, where a multi way tree structure is built up using symbols transformed from the resulted fuzzy keyword sets. Through rigorous security analysis, a solution made is secure and privacy preserving, while correctly realizing the goal of fuzzy keyword search. Extensive experimental results demonstrate the efficiency of the proposed solution.

Efficient technique for constructing fuzzy set is based on grams. The gram of a string is a substring that can be used as a signature for efficient approximate search. While gram has been widely used for constructing inverted list for approximate string search, gram is used for the matching purpose. The fact that any primitive edit operation will affect at most one specific character of the keyword, leaving all the remaining characters untouched. In other words, the relative order of the remaining characters after the primitive operations is always kept the same as it is before the operations.

7. Generalized Inverted IndeX

Keyword search has become a ubiquitous method for users to access text data in the face of information explosion. Inverted lists are usually used to index underlying documents to retrieve documents according to a set of keywords efficiently. Since inverted lists are usually large, many compression techniques have been proposed to reduce the storage space and disk I/O time. However, these techniques usually perform decompression operations on the fly, which increases the CPU time. It presents a more efficient index structure, the Generalized Inverted IndeX (Ginix), which merges consecutive IDs in inverted lists into intervals to save storage space.

With this index structure, more efficient algorithms can be devised to perform basic keyword search operations, i.e., the union and the intersection operations, by taking the advantage of intervals. Specifically, these algorithms do not require conversions from interval lists back to ID lists. As a result, keyword search using Ginix can be more efficient than those using traditional inverted indices. The performance of Ginix is also improved by reordering the documents in datasets using two scalable algorithms. Experiments on the performance and scalability of Ginix on real datasets show that Ginix not only requires less storage space, but also improves the keyword search performance, compared with traditional inverted indexes.

Beyond explicit user input, earlier work focused on handling recency queries, which are queries that are after recent events or breaking news. The time sensitive approach processes a recency query by computing traditional topic similarity scores for each document. In contrast to traditional models, which assume a uniform prior probability of relevance for each document d in a collection, define the prior to be a function of document d's creation date. The prior probability decreases exponentially with time, and hence recent documents are ranked higher than older documents. Li and Croft's strategy is designed for queries that are after recent documents, but it does not handle other types of time-sensitive queries, such as [Madrid bombing], [Google IPO], or even that implicitly target one or more past time periods.

8. Access Control Policies

Current approaches to enforce fine-grained access control on confidential data hosted in the cloud are based on fine-grained encryption of the data. Under such approaches, data owners are in charge of encrypting the data before uploading them on the cloud and re-encrypting the data whenever user credentials or authorization policies change. Data owners thus incur high communication and computation costs.

A better approach should delegate the enforcement of fine-grained access control to the cloud, so to minimize the overhead at the data owners, while assuring data confidentiality from the cloud. An approach, based on two layers of encryption, that addresses such requirement. Under this approach, the data owner performs a coarse-grained encryption, whereas the cloud performs a fine-grained encryption on top of the owner encrypted data. A challenging issue is how to decompose access control policies (ACPs) such that the two layer encryption can be performed. To utilize an efficient group key management scheme that supports expressive ACPs. This system assures the confidentiality of the data and preserves the privacy of users from the cloud while delegating most of the access control enforcement to the cloud.

9. Third Party Auditing

Using Cloud Storage, users can remotely store their data and enjoy the on-demand high quality applications and services from a shared pool of configurable computing resources, without the burden of local data storage and maintenance. However, the fact that users no longer have physical possession of the outsourced data makes the data integrity protection in Cloud Computing a formidable task, especially for users with constrained computing resources. Moreover, users should be able to just use the cloud storage as if it is local, without worrying about the need to verify its integrity.

Enabling public audit ability for cloud storage is of critical importance so that users can resort to a third party auditor (TPA) to check the integrity of outsourced data and be worry-free. To securely introduce an effective TPA, the auditing process should bring in no new vulnerabilities towards user data privacy, and introduce no additional online burden to user. A secure cloud storage system supporting privacy-preserving public auditing. The result is extended to enable the TPA to perform audits for multiple users simultaneously and efficiently. Extensive security and performance analysis show the proposed schemes are provably secure and highly efficient.

Utilize the public key based homomorphic authenticator and uniquely integrate it with random mask technique to achieve a privacy preserving public auditing system for cloud data storage security. To support efficient handling of multiple auditing tasks, we further explore the technique of bilinear aggregate signature to extend main result into a multi-user setting, where TPA can perform multiple auditing tasks simultaneously. Extensive security and performance analysis shows the proposed schemes are provably secure and highly efficient.

10. Mediated CertificateLess Cryptosystems

Security is a serious issue in cloud computing. Encryption is the solution for the security in cloud. There are many encryption techniques. Each one has its own merits and demerits. In the case of identity based encryption it is free from security mediator, predefined keys are there, and have the problem of key escrow and certificate revocation. Then the arrival of mediated certificateless scheme eliminates the key escrow problem, and certificate revocation problem. In certificateless encryption scheme, key generation process is divided in between the user and the cloud. Data owner encrypt the data using its secret key.

Then the data owner encrypt the secret key twice. Hence formed intermediate keys. Then send this encrypted data and intermediate keys to cloud. The cloud partially decrypt the intermediate key and send partially decrypted data and encrypted data to required user. The user decrypt the partialy decrypted data. Then the user will get the required key for decryption. So the user can decrypt it completl and the data owner can send same data to multiple clients with minimum cost.

Comparison Of Different Methods

Identity based encryption provides a conventional open key cryptosystem which requires a trusted Certificate Authority (CA) to issue computerized testaments that tie clients to their open keys. To assure the Confidentiality of sensitive data stored in public clouds, the data is encrypted before uploading it to the cloud. The cloud does not know the keys used to encrypt the data, the confidentiality of the data from the cloud is assured. Many organizations are required to implement fine grained access control to the data, the encryption mechanism should be able to support fine-grained encryption based access control. A normal approach used to support fine grained encryption based access control to encrypt different sets of data items to which the same access control policy. From the above algorithms the new system is found to provide highly secure data sharing in public cloud and to ensure the confidentiality in cloud data.

Conclusion

The first mCL-PKE scheme without pairing operations and provided its formal security. mCL-PKE solves the key escrow problem and revocation problem. Using the mCL-PKE scheme as a key building block, an improved approach to securely share sensitive data in public clouds. This approach supports immediate revocation and assures the confidentiality of the data stored in an untrusted public cloud while enforcing the access control policies of the data owner. Experimental results show the efficiency of basic mCL-PKE scheme and improved approach for the public cloud. Further, for multiple users satisfying the same access control policies, improved approach performs only a single encryption of each data item and reduces the overall overhead at the data owner.

References

1. Su Peng, Fucai Zhou, Qiang Wang, Zifeng Zu, and Jian Zu, “Identity Based Public Multi-Replica Provable Data Possession,” in Proc. Of IEEE Access, November 22,2017.
2. You Zhou and Liang Min Wang, “SDS2: Secure Data-Sharing Scheme for Crowd Owners in Public Cloud Service” in Proc. Of IEEE Second International Conference on Data Science in Cyberspace, 2017.
3. Chetan Gudisagar, Bibhu Ranjan Sahoo, Sushma M, Jaidhar C D, “Secure Data Migration between Cloud Storage Systems,” in Proc. Of ICACCI IEEE,2017.
4. Xiaojie Niu, “Fine-grained Access Control Scheme Based on Cloud Storage” in Proc. Of International Conference on Computer Network, Electronic and Automation, 2017.
5. Samundiswary.S, Nilima M Dongre, “Public Auditing for shared data in cloud with safe user revocation” in Proc. Of International

Conference on Electronics, Communication and Aerospace Technology ,2017.

6. A.Praveena, Dr. S.Smys, "Ensuring Data Security in Cloud Based Social Networks" in Proc. Of International Conference on Electronics, Communication and Aerospace Technology,2017.
7. Hisham Abdalla,Xiong Hu, Abubaker Wahaballa, Nabeil Eltayieb,Mohammed Ramadan1, Qin Zhiguang, "Efficient Functional Encryption and Proxy Re-cryptography for Secure Public Cloud Data Sharing" in Proc. Of IEEE ICOACS,2016.
8. Mohamed Yoosuf and Mohamed Nabeel, "An Efficient Certificateless Encryption for Secure Data Sharing in Public Clouds," in Proc. Of IEEE Transactions on Knowledge and Data Engineering,Volume 26,2014.
9. Jueeli Dangur, S.M. Jaybhaye, "Framework for Secure Data Sharing In Dynamic Group Using Public Cloud", in Proc. Of "International Conference on Computing, Analytics and Security Trends,2016.
10. Mrs.Priyadharsini.V, Dr.S.V.M.G. Bavithiraja, M.Mohanapriya and Mr.M.Balachandar, "Performance Evaluation of 4G Uplink MAC Scheduling Algorithms – A Review", International Journal of Advanced Engineering Recent Technology,Vol 17 Issue 1, March 2017.
11. Mr.Selvakumar.K, Dr.S.V.M.G. Bavithiraja, Dr.C.Gunavathi and Mr.M.Balachandar, "Application of Cloud Computing in Aerospace and Defense", International Journal of Advanced Engineering Recent Technology,Vol 17 Issue 1.March 2017.
12. Kaitai Liang, Man Ho Au, Joseph K. Liu, Willy Susilo, Duncan S. Wong, Guomin Yang, Tran Viet Xuan Phuong, and Qi Xie "A DFA-Based Functional Proxy Re-Encryption Scheme for Secure Public Cloud Data Sharing K" in Proc.Of IEEE Transactions On Information Forensics And Security, Vol. 9,October 2014.
13. Kaiping Xue and Peilin Hong, "A Dynamic Secure Group Sharing Framework in Public Cloud Computing" in Proc. Of IEEE Transactions on Cloud Computing, 2014.
14. Shahina K M, Deepak Lal , "A Survey on Secure Data Sharing Methods in Public Cloud Computing" in Proc. Of International Journal of Science, Engineering and Technology Research, Volume 5, Issue 01, January 2016.
15. Ramyasree Nandagiri and Dinesh Chandrasehkaran, "Secure Data Sharing Using Certificate less Encryption for Providing Efficiency in Public Clouds" in Proc. Of International Research Journal of Engineering and Technology, Volume 02, Issue 09, December 2015.