Sharing Secured Data in Cloud using Aggregate Key

Aathira M[1] Sandra K[2] Sonali K[3] Sethulakshmi S[4]
[1,2,3,4]Department of Computer Science
[1,2,3,4]Nehru College of Engineering and Research Centre Thrissur, India

Abstract— Cloud storage is a model of data storage in which the digital data is stored in logical pools, the physical storage spans multiple servers and physical environment is managed by the hosting company. One of the major functions of cloud storage is sharing of data. In this paper, we use a new public key cryptosystem which use a constant size ciphertext. This can provide an efficient decryption mechanism. Any set of keys can be aggregated and can be made as compact as possible. The user who holds the key can release the constant size key that is aggregated for the different choices of ciphertexts within the cloud storage. Even though, the users can release those aggregate keys, the other files that are encrypted will be kept confidential. The aggregated key can be securely stored in the smart cards.

Key words: KCC, RO, JDBC

I. Introduction

Recently, cloud storage is much popular in the field of computer science. It is also used in online services as core technology for users personal purposes like force accounts for email, photo album, file sharing. Considering the confidentiality of data, a traditional way to provide security is to rely on the server to complaince the access control after authentication. The challenging issue in cloud storage is to how securely and efficiently we can share the encrypted the encrypted data. The users should be able to authorize the access rights of sharing the data with others. Hence, they can efficiently make use of this data directly from various servers. We solve this problem by proposing a new method known as Key Conjunction Cryptosystem (KCC). In this method, the encryption of the data is not only done by using an public key, but also by using an identifier of ciphertext known as class. This also refers that these cihpertexts are again classified into various classes. This leads to creation of master secret key, which is owned by the key owner. This key is used for the extraction of secret keys in various classes. These extracted keys are the aggregated and made it as a compact key for a single class.

II. Existing system

The existing system follows the tree structure, in which a key is provided to each node. These keys can be used to derive its children nodes. Many more advanced cryptographic scheme for key assignment were introduced either for acyclic or cyclic graphs[1][2][3]. The key is generated for the symmetric key cryptosystems in most of these schemes. The major drawback in the tree structure is that if a user wants to share a key of a particular node, he automatically grants the keys of the descendant node which is not practically acceptable.

III. Proposed system

We considered the verifiability of the cloud's transformation and provided a method to check the correctness of the transformation. However, we did not formally define verifiability. But it is not feasible to construct Abe scheme with verifiable outsourced decryption following the model defined in the existing. Moreover, the method proposed in existing relies on random oracles (RO).

IV. Implementation

The proposed system can be implemented using JAVA as front end and SQL as back end. The database is connected with the front end by JDBC. Key conjunction cryptosystem consist of a group admin and users. The key generated files are only uploaded into the cloud. The users can upload and request the files. To upload the file, the user needs to select the file and should generate key for that file, then the admin can upload the file into the cloud.

The user can request the file by selecting the filename. Then the private and public key sent via mail. Finally the user can download the file. The admin performs functions such as uploading files, verification and security. The client can perform functions such as uploading files, key generation, requesting files and downloading.

A. Key Aggregate Encryption:

Using this scheme, the key can be aggregated. The key algorithms are as follows.

The owner of the data makes the public key system via setup and generate the public and private keys by key generation phase. The files can be encrypted by the user. The owner of the data can make the decryption key via extract for decryption rights. The generated keys can be sent through mail or other secure mechanism. Finally the user can decrypt the ciphertext using the aggregated key decrypt. For the basic construction of key aggregation cryptosystem these five algorithms are needed.

B. Public Key Extension:

If the user needs to makes his ciphertexts into more classes, the user can request for more key pairs. So the new public key can be given to a new user. By public key extension also we achieve shorter key size and flexibility. The public key is extended similar to the algorithm in key aggregate encryption scheme.

V. Result and discussion

The key conjunction cryptosystem is a system in which data can be shared by using an aggregate key of constant size. Key aggregate encryption scheme is used. Thus user can download the files by using the key sent through the mail and other files will be confidential. The system provides a flexible way of secure sharing of data.

VI. Conclusion

The assurance for user data privacy is a central question in closed storage. By using more mathematical tools, cryptographic schemes are able to adapt to many different functions and involves more than one keys for a single application. Key aggregation mainly considers how to compress different secret keys for various ciphertext classes in cloud storage. Whatever may be the set of classes, we will get aggregate key of constant size. This idea is much better than hierarchical key assignment.

VII. FUTURE SCOPE

The developed system can be further enhanced by reducing the key size. Also the key can be sent to the user's personal mobile number instead of sending the key to the user's mail thereby providing easy access.

REFERENCES

[1] Y. Sun and K. J. R. Liu, "Scalable Hierarchical Access Control in Secure Group Communications," in Proceedings of the 23th IEEE International Conference on Computer Communications (INFOCOM'04). IEEE, 2004.

[2] Q. Zhang and Y. Wang, "A Centralized Key Management Scheme for Hierarchical Access Control," in Proceedings of IEEE Global Telecommunications Conference (GLOBECOM '04). IEEE, 2004, pp.2067–2071.

[3] M. J. Atallah, M. Blanton, N. Fazio, and K. B. Frikken, "Dynamic and Efficient Key Management for Access Hierarchies," ACM Transactions on Information and System Security (TISSEC), vol. 12, no. 3, 2009.