ISSN:0975-9646

A Literature Survey on Secure Data Sharing in Cloud Storage with Key Aggregate Cryptosystem

Prof. B. M. Kore [#1], Archana Jadhav [*2], Prof. V. V. Pottigar [#3]

[#]*Computer Science And Engineering Department, Solapur University, SKN Sinhgad College of Engineering, Korti, Pandharpur, India*

***Abstract*— Cloud Computing is a technology that uses the internet and central remote servers to maintain data and applications. Cloud computing allows consumers and businesses to use applications without installation and access their personal files at any computer with internet access. In the key aggregate cryptosystem for cloud data sharing efficient public key encryption scheme which support flexible delegation in the sense that any subset of the cipher texts is decryptable by a constant-size decryption key. The secret key holder can release a constant size aggregate key for flexible choices of cipher text in cloud storage. This paper reveals an overview and study of cryptographic techniques for securely and efficiently data sharing in cloud storage.**

***Keywords*— Cloud storage, data sharing, key aggregate encryption, cryptography.**

I. INTRODUCTION

Cloud system can be used to enable data sharing capabilities and this can proven abundant of benefits to the user. There is currently a push for IT organization to increase their data sharing efforts. In enterprise settings, there is the rise in demand for data outsourcing, which assists in the strategic management of corporate data. It is also used as a core technology behind many online services for personal applications. With current technology user can access almost all of their files or emails by mobile phone or computer from any corner of the world.

In the cloud storage efficient public key encryption scheme which support flexible delegation in the sense that any subset of the cipher texts is decryptable by a constant-size decryption key. In other words, the secret key holder can release a constant-size aggregate key for flexible choices of ciphertext set in cloud storage.

In KAC user can encrypt message not only under a public-key but also under an identifier of cipher texts called class. The ciphertexts are further categorized into different classes. The key owner holds a master-secret key called master secret key. The extracted key have can be an aggregate key which is as compact as a secret key for a single class, but aggregate the power of many such keys, i.e., the decryption power of any subset of cipher text classes.

The scheme enable a content provider to share her data in a confidential and selective way, with a fixed and small cipher text expansion, by distributing to each authorized user a single, compact, aggregate key.

Cryptography helps the data owner to share the data to in safe way. Cryptography is the practice and study of hiding information. It is the Art or Science of converting a plain intelligible data into an unintelligible data (i.e. encryption) and again retransforming that message into its original form (i.e. decryption). It provides Confidentiality, Integrity, and Accuracy.

A cryptographic solution, with proven security relied on number-theoretic assumptions is more desirable Data sharing is important functionality in cloud storage. For example bloggers can let their friends view private data or an enterprise may grant their employee access to important data. But the problem is how to effectively share encrypted data. Of course users can download the encrypted data from the storage, decrypt them and then send them to others for sharing, but it loses the value of cloud storage. So user should be able to give access rights of sharing data to others so that they can access these data from server directly.

II. LITERATURE SURVEY

Key assignment scheme aim to minimize the expense in storing and managing secret keys for general cryptographic use. Key assignment schemes most likely non-constant decryption key size, symmetric or public key for a predefined hierarchy is used. Only hash functions are used for a node to derive a descendant's key from its own key. The space complexity of the public information is the same as that of storing hierarchy and is asymptotically optimal; the private information at a node consists of a single key associated with that node and updates are handled locally in the hierarchy [2].

Presented an encryption scheme which is originally proposed for concisely transmitting large number of keys in broadcast scenario. And uses Symmetric-key encryption with Compact Key. In this paper build an efficient system that allows patients both to share partial access rights with others, and to perform searches over their records. They formalize the requirements of a Patient Controlled Encryption scheme, and give several instances, based on existing cryptographic primitives and protocols, each achieving a different set of properties.

However, it is designed for the symmetric-key setting instead. The encryptor needs to get the corresponding secret keys to encrypt data, which is not suitable for many applications. Since their method is used to generate a secret value rather than a pair of public/secret keys, it is unclear how to apply this idea for public-key encryption scheme [3]. Identity-based encryption (IBE) is a type of public-key encryption in which the public-key of a user can be set as an identity-string of the user (e.g., an email address). There is a trusted party called private key generator (PKG) in IBE which holds a master-secret key and issues a secret key to each user with respect to the user identity. The encryptor can take the public parameter and a user identity to encrypt a message. The recipient can decrypt this ciphertext by his secret key. In this scheme, key aggregation is constrained in the sense that all keys to be aggregated must come from different "identity divisions". While there are an exponential number of identities and thus secret keys, only a polynomial number of them can be aggregated. This greatly increases the costs of storing and transmitting ciphertexts, which is impractical in many situations such as shared cloud storage [4].

Attribute-based encryption (ABE) allows each ciphertext to be associated with an attribute, and the master-secret key holder can extract a secret key for a policy of these attributes so that a ciphertext can be decrypted by this key if its associated attribute conforms to the policy. In a multi-authority ABE scheme, multiple attribute-authorities monitor different sets of attributes and issue corresponding decryption keys to users and encryptors can require that a user obtain keys for appropriate attributes from each authority before decrypting a message.

However, the major concern in ABE is collusion-resistance but not the compactness of secret keys. Indeed, the size of the key often increases linearly with the number of attributes it encompasses or the ciphertext-size is not constant [5].

TABLE I

Summary of Literature review		
Referred Paper	**Description**	**Conclusion**
Dynamic and Efficient Key Management for Access Hierarchies[2]	In this defined a key allocation mechanism that implements such an access graph, that is, an assignment of keys to users and objects where a user can access an object if he has a key for that object.	The number of keys increases with the number of branches. It is unlikely to come up with a hierarchy that can save the number of total keys to be granted.
Patient Controlled Encryption: Ensuring Privacy of Electronic Medical Records[3]	In this paper build an efficient system that allows patients both to share partial access rights with others, and to perform searches over their records.	The encryptor needs to get the secret keys to encrypt data which is not suitable for many applications. It is unclear how to apply this method for public key encryption scheme.
Practical Leakage-Resilient Identity-Based Encryption from Simple Assumptions[4]	Identity-based encryption (IBE) is a type of public-key encryption in which the public-key of a user can be set as an identity-string of the user (e.g., an email address).	Different secret keys have to be generated for the same identities, and as a result it is more difficult to apply leakage resilient techniques.
Improving Privacy and Security in Multi-Authority Attribute-Based Encryption[5]	In this scheme multiple attribute-authorities monitor different sets of attributes and issue corresponding decryption keys to user and encryptors can require that a user obtain keys for appropriate attributes from each authority before decrypting a message.	The size of the key often increases with the number of attributes it encompasses or the ciphertext-size is not constant.

III. PROPOSED SYSTEM

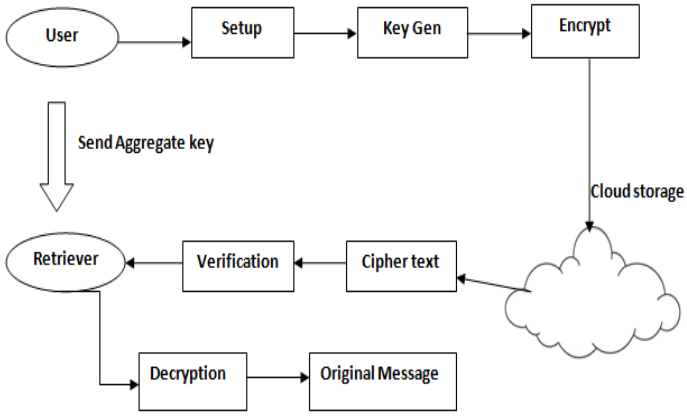

Fig. 1: System architecture

In KAC, users encrypt a message not only under a public-key, but also under an identifier of ciphertext called class. That means the ciphertexts are further categorized into different classes. The key owner holds a master-secret called master-secret key, which can be used to extract secret keys for different classes. More importantly, the extracted key have can be an aggregate key which is as compact as a secret key for a single class, but aggregates the power of many such keys, i.e., the decryption power for any subset of ciphertext classes. With our solution, user can simply send retriever a single aggregate key via a secure e-mail. Retriever can download the encrypted photos from user's Dropbox space and then use this aggregate key to decrypt these encrypted photos. The scenario is depicted in Fig. 1

A key-aggregate encryption scheme consists of five polynomial-time algorithms as follows.

Setup Phase: Executed by the data owner to setup an account on an untrusted server. On input a security level parameter and the number of ciphertext classes n (i.e., class index should be an integer bounded by 1 and n), it outputs the public system parameter param, which is omitted from the input of the other algorithms for brevity.

KeyGen Phase: Executed by the data owner to randomly generate a public/master-secret key pair (pk, msk).

Encrypt Phase (pk, i, m): Executed by anyone who wants to encrypt data. On input a public-key pk, an index i denoting the ciphertext class, and a message m, it outputs a ciphertext C.

Extract (msk, S): Executed by the data owner for delegating the decrypting power for a certain set of ciphertext classes to a delegate. On input the master secret key msk and a set S of indices corresponding to different classes, it outputs the aggregate key for set S denoted by KS.

Decrypt(KS, S, I, C): Executed by a delegate who received an aggregate key KS generated by Extract. On input KS, the set S, an index i denoting the ciphertext class the ciphertext C belongs to, and C, it outputs the decrypted result m if i belongs to S .

IV. SCOPE

1. With the advancements in Cloud computing, there is now a growing focus on implementing data sharing capabilities in the Cloud. It is also used as a core technology behind many online services for personal applications
2. On cloud anyone can share data as much they want to i.e. only selected content can be shared.
3. Cryptography helps the data owner to share the data to in safe way. So user encrypts data and uploads on server.
4. Key aggregate cryptosystem technique used for data sharing in cloud storage is more secure.
5. This technique is useful for securely, efficiently, and flexibly share data with others in cloud storage.
6. It is an efficient public-key encryption scheme which supports flexible delegation.

V. CONCLUSIONS

In this survey we studied different cryptographic techniques for data sharing security. One trivial solution to achieving secure data sharing in the cloud is for the data owner to encrypt his data before storing into the Cloud, and hence the data remain information-theoretically secure against the Cloud provider and other malicious users. With more mathematical tools, cryptographic schemes are getting more versatile and often involve multiple keys for a single application. Thus it considers how to compress secret keys in public-key cryptosystems which support delegation of secret keys for different cipher text classes in cloud storage. Our approach is more flexible than hierarchical key assignment.

REFERENCES

[1] Cheng-Kang Chu et.al, "Key-Aggregate Cryptosystem for Scalable Data Sharing in Cloud Storage", IEEE Transactions on Parallel and Distributed Systems. Volume: 25, Issue: 2. Year :2014.

[2] M. J. Atallah et.al, "Dynamic and Efficient Key Management for Access Hierarchies," ACM Transactions on Information and System Security (TISSEC), vol. 12, no. 3, 2009.

[3] J. Benaloh et.al, "Patient Controlled Encryption: Ensuring Privacy of Electronic Medical Records," in Proceedings of ACM Workshop on Cloud Computing Security (CCSW '09). ACM, 2009, pp. 103–114.

[4] S. S. M. Chow et.al, "Practical Leakage-Resilient Identity-Based Encryption from Simple Assumptions," in ACM Conference on Computer and Communications Security, 2010, pp. 152–161.

[5] M. Chase and S. S. M. Chow, "Improving Privacy and Security in Multi-Authority Attribute-Based Encryption," in ACM Conference on Computer and Communications Security, 2009, pp. 121–130.

[6] S. S. M. Chow, Y. J. He, L. C. K. Hui, and S.-M. Yiu, "SPICE – Simple Privacy-Preserving Identity-Management for Cloud Environment," in Applied Cryptography and Network Security – ACNS 2012, ser. LNCS, vol. 7341. Springer, 2012, pp. 526–543.

[7] B. Wang, S. S. M. Chow, M. Li, and H. Li, "Storing Shared Data on the Cloud via Security-Mediator," in International Conference on Distributed Computing Systems - ICDCS 2013. IEEE, 2013.

[8] L. Hardesty, "Secure computers aren't so secure," MIT press, 2009, http://www.physorg.com/news176107396.html

[9] C. Wang, S. S. M. Chow, Q. Wang, K. Ren, and W. Lou, "Privacy-Preserving Public Auditing for Secure Cloud Storage," IEEE Trans. Computers, vol. 62, no. 2, pp. 362–375, 2013.

[10] F. Guo, Y. Mu, and Z. Chen, "Identity-Based Encryption: How to Decrypt Multiple Ciphertexts Using a Single Decryption Key," in Proceedings of Pairing-Based Cryptography (Pairing '07), ser. LNCS, vol. 4575. Springer, 2007, pp. 392–406.