ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

Performing Key-Aggregate Cryptosystem on Scalable Data Sharing In Cloud Storage

Chandrasekhar Patnana
M.Tech Student
Department of CSE,
Sri Venkateswara College of Engineering and Technology.

RVLS N Sastry, M.Tech
Associate Professor
Department of CSE,
Sri Venkateswara College of Engineering and Technology.

Abstract:
Cloud storage is a storage of data online in cloud which is accessible from multiple and connected resources. Cloud storage can provide good accessibility and reliability, strong protection, disaster recovery, and lowest cost. Cloud storage having important functionality i.e. securely, efficiently, flexibly sharing data with others. New public–key encryption which is called as Key-aggregate cryptosystem (KAC) is introduced. Key-aggregate cryptosystem produce constant size ciphertexts such that efficient delegation of decryption rights for any set of ciphertext are possible. Any set of secret keys can be aggregated and make them as single key, which encompasses power of all the keys being aggregated. This aggregate key can be sent to the others for decryption of ciphertext set and remaining encrypted files outside the set are remains confidential.

Keywords— ***Cloud storage, Key-aggregate cryptosystem (KAC), Ciphertext, Encryption, Decryption, secret key.***

1. INTRODUCTION

Cloud storage is nowadays very popular storage system. Cloud storage is storing of data off-site to the physical storage which is maintained by third party. Cloud storage is saving of digital data in logical pool and physical storage spans multiple servers which are manage by third party. Third party is responsible for keeping data available and accessible and physical environment should be protected and running at all time. Instead of storing data to the hard drive or any other local storage, we save data to remote storage which is accessible from anywhere and anytime. It reduces efforts of carrying physical storage to everywhere. By using cloud storage we can access information from any computer through internet which omitted limitation of accessing information from same computer where it is stored. While considering data privacy, we cannot rely on traditional technique of authentication, because unexpected privilege escalation will expose all data. Solution is to encrypt data before uploading to the server with user's own key.

Data sharing is again important functionality of cloud storage, because user can share data from anywhere and anytime to anyone. For example, organization may grant permission to access part of sensitive data to their employees. But challenging task is that how to share encrypted data. Traditional way is user can download the encrypted data from storage, decrypt that data and send it to share with others, but it loses the importance of cloud storage. Cryptography technique can be applied in a two major ways- one is symmetric key encryption and other is asymmetric key

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

encryption. In symmetric key encryption, same keys are used for encryption and decryption. By contrast, in asymmetric key encryption different keys are used, public key for encryption and private key for decryption. Using asymmetric key encryption is more flexible for our approach. This can be illustrated by following example. Suppose Alice put all data on Box.com and she does not want to expose her data to everyone. Due to data leakage possibilities she does not trust on privacy mechanism provided by Box.com, so she encrypt all data before uploading to the server. If Bob ask her to share some data then Alice use share function of Box.com. But problem now is that how to share encrypted data. There are two severe ways: 1. Alice encrypt data with single secret key and share that secret key directly with the Bob. 2. Alice can encrypt data with distinct keys and send Bob corresponding keys to Bob via secure channel. In first approach, unwanted data also get expose to the Bob, which is inadequate. In second approach, no. of keys is as many as no. of shared files, which may be hundred or thousand as well as transferring these keys require secure channel and storage space which can be expensive.

Therefore best solution to above problem is Alice encrypts data with distinct public keys, but send single decryption key of constant size to Bob. Since the decryption key should be sent via secure channel and kept secret small size is always enviable. To design an efficient public-key encryption scheme which supports flexible delegation in the sense that any subset of the (produced by the encryption scheme) is decryptable by a constant-size decryption key (generated by the owner of the master-secret key).

2. LITERATURE SURVEY

Cloud computing is visualized as architecture for succeeding generation. It has many facilities though have a risk of attacker who can access the data or leak the users identity. While setting a cloud users and service providers authentication is necessary. The issue arises whether loud service provider or user is not compromised. The data will leak if any one of them in compromised. The cloud should be simple, preserving the privacy and also maintaining user's identity.

The flexible use of cloud storage for user is a need as it is seams accessing data locally though that is present at remote side. It is important to inspect the data set on the cloud. So it is necessary to allow a public audit for integrity of outsourced data through third party auditor (TPA). TPA is also beneficial for cloud service provider. It checks the correctness of the outsourced data. TPA should be able to do public auditability, storage correctness, privacy preserving, Batch auditing with minimum communication and computation overhead.

There are many cloud users who wants to upload there data without providing much personal details to other users. The anonymity of the user is to be preserved so that not to reveal the identity of data owner. Provable data possession (PDP) uses similar demonstrating marks to reduce computation on server, and network traffic. PDA ensures the data present on cloud which is un-trusted is original without accessing it. Security mediator (SEM) is approach allows the user to preserve the anonymity. Users are meant to upload all their data to SEM so that the SEM is not able to understand the data although it's going to generate the verification on data. As the users are signed at SEM it should not know the identity of uploader [3].

Another way for sharing encrypted data is Attribute-Based Encryption (ABE). It is likely to encrypt the data with attributes which are equivalent to users attribute rather than only encrypting each part of data. In ABE attributes description is considered as set so that only a particular key which is matched with attribute can decrypt the ciphertext. The user key and the attribute are matched if it matches it can decrypt a particular ciphertext. When there are k attributes are overlay among the ciphertext and a private key the decryption is granted.

3. KEY-AGGREGATE CRYPTOSYSTEM

In key-aggregate cryptosystem (KAC), users encrypt a message not only under a public-key, but also under an

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

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.[1] With our example, Alice can send Bob a single aggregate key through a secure e-mail. Bob can download the encrypted photos from Alice's Box.com space and then use this aggregate key to decrypt these encrypted data. The sizes of ciphertext, public-key, master-secret key and aggregate key in KAC schemes are all of constant size. The public system parameter has size linear in the number of ciphertext classes, but only a small part of it is needed each time and it can be fetched on demand from large (but non-confidential) cloud storage.

FRAMEWORK:

The data owner establishes the public system parameter through Setup and generates a public/master-secret key pair through KeyGen. Data can be encrypted via Encrypt by anyone who also decides what ciphertext class is associated with the plaintext message to be encrypted. The data owner can use the master-secret key pair to generate an aggregate decryption key for a set of ciphertext classes through Extract. The generated keys can be passed to delegatees securely through secure e-mails or secure devices Finally, any user with an aggregate key can decrypt any ciphertext provided that the ciphertext's class is contained in the aggregate key via Decrypt.

Key aggregate encryption schemes consist of five polynomial time algorithms as follows:

1. Setup (1λ, n): The data owner establishes public system parameter via Setup. On input of a security level parameter 1λ and number of ciphertext classes n , it outputs the public system parameter param

2. KeyGen: It is executed by data owner to randomly generate a public/ master-secret key pair (Pk, msk).

3. Encrypt (pk, i, m): It is executed by data owner and for message m and index i, it computes the ciphertext as C.

4. Extract (msk, S): It is executed by data owner for delegating the decrypting power for a certain set of ciphertext classes and it outputs the aggregate key for set S denoted by Ks.

5. Decrypt (Ks, S, I, C): It is executed by a delegate who received, an aggregate key Ks generated by Extract. On input Ks, set S, an index i denoting the ciphertext class ciphertext C belongs to and output is decrypted result m.

A canonical application of KAC is data sharing. The key aggregation property is especially useful when we expect delegation to be efficient and flexible. The KAC schemes enable a content provider to share her data in a confidential and selective way, with a fixed and small ciphertext expansion, by distributing to each authorized user a single and small aggregate key. Data sharing in cloud storage using KAC, illustrated in Figure 1. Suppose Alice wants to share her data m1,m2,....,mn on the server. She first performs Setup (1λ, n) to get param and execute KeyGen to get the public/master-secret key pair (pk, msk). The system parameter param and public-key pk can be made public and master-secret key msk should be kept secret by Alice. Anyone can then encrypt each mi by Ci = Encrypt (pk, i, mi). The encrypted data are uploaded to the server. With param and pk, people who cooperate with Alice can update Alice's data on the server. Once Alice is willing to share a set S of her data with a friend Bob, she can compute the aggregate key KS for Bob by performing Extract (msk, S). Since KS is just a constant size key, it is easy to be sent to Bob through a secure e-mail. After obtaining the aggregate key, Bob can download the data he is authorized to access. That is, for each i ϵ S, Bob downloads Ci from the server.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

With the aggregate key KS, Bob can decrypt each Ci by Decrypt (KS, S, i, Ci) for each i ϵ S.

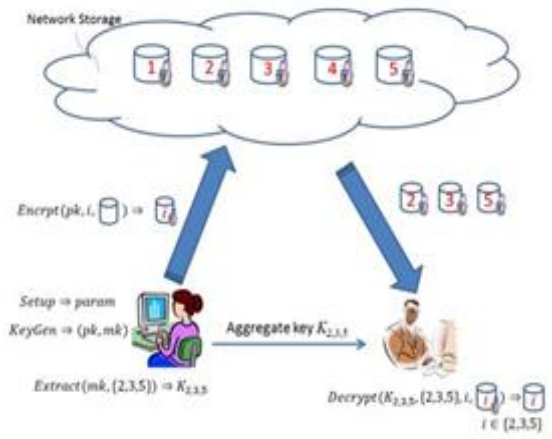

Figure. 1. Using KAC for data sharing in cloud storage.

4. Compact Key in Identity-Based Encryption

Identity based encryption scheme in is a form of the public key encryption . In this the public key of user is set as string-identity of user. In the IBE Private Key Generator which hold a master secret key and issue it to other user as per their identity. The user who encrypt the message can take public parameter and identity of user to decrypt message. The recipient decrypt ciphertext using own secret key. Guo et al. [14], [3] tried to create IBE with key aggregation . One of their technique [14] assumes random oracle but other one not [3]. Very importantly , their aggregation of key [14],[3] comes at expense of the size for both ciphertext and public parameter. This increases cost of storing and transferring ciphertext, which is not practical in some conditions. In fuzzy IBE [13], one individual secret key can decrypt siphertext under multiple identities which are close in more metric space, but not for random set of identities and it does not match with key aggregation idea.

5. Other Encryption Scheme

Attribute based Encryption allows each encrypted text to be connected with feature, and the master secret key possesor can take out secret key for a policy of this feature so that encrypted text can be decrypted by this key if it is associated attributes conforms to policy. In AB important issue is collusion-resistance not the compactness of secret keys. The range of the encrypted text is not fix. A PRE schme permit Alice to delegate to server ability to convert ciphertext encrypted under own public-key ones bob.The Proxy Reencryption PRE technique is well known to various application [17].Using PRE scheme only shift the secure key storage requirement from delegatee to proxy. Thus it is not suitable to let proxy reside in storage server. It will not suitable so each decryption needs individual interaction with proxy.

6. Construction of KAC

Boneh et al. [18] present collusion-resistant broadcast encryption scheme by using this basic scheme is designed. Their technique support fixed-size secret keys, each key only has

1. Encryption: - output ciphertext
2. Extract
3. Decrypt: - Output message

The decryption can be done more efficiently .To make extended scheme best different ciphertext classes suggested for various purposes opposite to different publickeys. This key extension approach can also view as key update process. Suppose a secret ka value is compromised then we can replace it with new key value. The less aggregate key size reduces communication overhead for transferring the new key.

CONCLUSION

Users data privacy is a central question of cloud storage. Compress secret keys in public-key cryptosystems which support delegation of secret keys for different cipher text classes in cloud storage. No matter which one among the power set of classes, the delegatee can always get an aggregate key of constant size. In cloud storage, the number of cipher texts usually grows rapidly without any restrictions. So we have to reserve enough cipher text classes for the future extension. Otherwise, we need to expand the public-key. Although the parameter can be downloaded with cipher texts, it would be better if its

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

size is independent of the maximum number of cipher text classes.

REFERENCES

[1]. 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.

[2]. 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.

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

[4]. Cheng-Kang Chu, Sherman S.M. Chow, Wen-Guey Tzeng, Jianying Zhou, and Robert H. Deng,"Key-Aggregate Cryptosystem for Scalable Data Sharing in Cloud Storage" IEEE Transactions On Parallel And Distributed System, Vol 25, No. 2 February 2014.

[5]. V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-Based Encryption for Fine-Grained Access Control of Encrypted data,"in Proceedings of the 13th ACM Conference on Computer and Communications Security (CCS '06). ACM, 2006, pp. 89–98.

[6]. 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.

[7]. D. Boneh and M. K. Franklin, "Identity-Based Encryption from the Weil Pairing," in Proceedings of Advances in Cryptology – CRYPTO '01, ser. LNCS, vol. 2139. Springer, 2001, pp. 213–229.

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