ISSN:0975-9646

Secure Multi-Cloud data sharing using Key Aggregate Cryptosystem for scalable data sharing.

Suhas Bachhav[1], Chetan Chaudhari[2], Nikhilesh Shinde[3], Poonam Kaloge[4]

[1]Computer Dept, SND COE & RC Yeola, Nashik

Abstract-**In Cloud Computing data sharing and security is important functionality. We describe public key encryption .efficient delegation of decryption which of stable size of cipher text is to possible. The new thing is that one can any aggregate set of secrete keys and make them compact as a single key. That can encompass the power of all keys aggregated. The new produced Aggregated Key can be send via email or mobile OTP or small memory storage device to the client. Ensuring the security of cloud computing is second major factor and dealing with because of service availability failure the "single cloud" providers demonstrated less famous failure and possibility malicious insiders in the single cloud. A movement towards "Multi-Clouds", In other words "Inter-Clouds" or "Cloud-Of-Clouds" as emerged recently. This works aim to reduce security risk and better flexibility and efficiency to the user.**

Index Terms- **Cloud Storage, Key Aggregate Encryption, Multi-cloud infrastructure, Data Sharing.**

I. INTRODUCTION

Now a day's cloud is gaining popularity. The demand of data outsourcing is handled and manages of corporate data. Generally many online free space providers provides Free storage space more than 15 GB take few thousands amount more than 1 TB. The main benefit that it's not only providing flexibility and scalability but it also provides maintainability and accessing easily data. Cloud service providers have most critical issues of data integrity and privacy because data not store on his own servers the privacy can be achieve by diving and encrypted data store on service providers with the respect of quality access data.

In cloud shared commuting environments, things become even worse. Different clients can be posted on separate Virtual machines but resides on a single physical machine. For the data privacy a cryptography solution are confident. In a sharable data the number theoretic assumptions is more desirable but user is not perfectly happy. As well as the technical staff of service providers may not be trusted.

Assume that Suhas puts his all documents in Google drive, and He does not want to expose to others due to data leakage possibility Suhas cannot feel protection his data. So He encrypts his data using his own keys before uploading. One day, Suhas's partners Pritam ask his to share some data for security a possible option. Suhas choose it to securely send Pritam. The secrete keys involved. Generally, there are two ways for traditional encryption paradigm:

- Suhas encrypts all files with a single encryption key and gives Pritam the corresponding secret key directly.
- Suhas encrypts files with distinct keys and sends Pritam the Corresponding secret keys.

Example, in enterprise settings, every Client can upload encrypted data on the cloud storage server without the knowledge of the company's master-secret key.[1]

Therefore, the best solution for the above problem is that Suhas encrypts files with distinct public-keys, but only sends Pritam a single (constant-size) decryption key. Since the decryption key should be directed through a secure channel and kept secret, small key size is always desirable. For example, we cannot guess great storage for decryption keys in the resource-constraint devices like smart phones, The smart cards or the wireless sensor nodes. Especially, these secret keys are usually stored in the tamper-proof memory, which is relatively costly [2].

II. LITERATURE SURVEY

A. *Study of Existing Systems/ Technologies*

I. Identity Bases Encryption (IBE): IBE is a type of a public-key encryption. User's public key (it's a set of encryption that is Identity-String). In IBE, Master secret keys are generated by the private key generator and here on the basis on user's identity secret key is provided. Sender wants to share files. So sender will encrypt the files by making use of user identity and public parameter and sends the files. By making use of his secret key Receiver will Decrypt Files. But out of key-aggregation and IBE, Random oracles assumed by only one. .From various identity key aggregation is inhibited as key to be aggregated.[3]

Advantages

- Encryption type is public-key encryption.
- This scheme has a dependable party which will grasp secret key.
- Based on the identity, secret key will be provided. Size of decryption key is Constant.

Disadvantage

- Cipher text size is non-constant.
- Cost of storing cipher text and transmitting it expensive.

II. Symmetric Key Encryption: Benaloh proposed an encryption scheme, where a huge number of keys can be sent rapidly in a broadcast consequence. The key origin is as follows. Initially choose two prime numbers p and q for a composite module. Master secret key will be chosen randomly. Dissimilar prime numbers will be allied with each class. All the prime numbers will be put for the purpose of a public System parameter. The outcome of this is a constant size key. For the purpose of symmetric-key setting, this method is designed so with corresponding secret keys sender should encrypt the files which will not be practicable[4].

Advantages

- Cipher text size is constant.
- Decryption key size is constant.
- For storing cipher text and keys it required fewer spaces.
- Construction is simple.

Disadvantage

- Both encryption and decryption is done by same key.
- Encryptor should get corresponding key to encrypt files.

III. Attribute Based Encryption (ABE): In Attribute Based Encryption method an attribute will be linked with cipher text. From master secret key, the secret key will be derived. This secret key is used to decrypt the files merely if all its associate delements go after the rules. Before Attribute Based Encryption method was introduced, the user who wanted secret key must go to third party and proving he is real by providing his identity and then he was capable to decrypt the file the secret key of user was not allowed to a single center in the ABE Scheme. Instead it was authorized by independent authorities. But still this scheme has drawback i.e. no solidity on secret key. Here in this scheme there is linear rise in key size, with the rise in attributes.

Advantages

- Encryption type is public key encryption.
- Cipher text size is constant.

Disadvantage

- Decryption key size is non-constant.
- Requires more space to store keys.
- The size of Decryption key rises linearly.
- Managing keys is expensive.

IV. Attribute-Based Encryption for Fine-Grained Access Control of Encrypted Data(2006): Each cipher text to be associated with an attribute allowed by ABE(Attribute-based encryption), and the master-secret key holder can extract a secret key for a policy of these attributes so that a cipher text can be decrypted by this key if its related attribute follows to the policy. For example By using secret key for the policy (2v3v6v8), one can decrypt cipher text tagged with class number 2, 3, 6, or 8Collusion resistance but not the compactness of secret keys is the major concern in ABE. Indeed, the size of the key often rises linearly with the number of attributes it encompasses, or the cipher text-size is not constant.

V. Chosen-Cipher text Secure Proxy Re-Encryption (2007): To delegate the decryption power of some cipher texts without transferring the secret key to the representative, a useful primitive is proxy re-encryption (PRE). A PRE scheme allows sender to representative to the server (proxy) the ability to convert the cipher texts encrypted under her public-key into ones for receiver. PRE is well known to have numerous applications including cryptographic system. Nevertheless, sender has to trust the proxy that it only converts cipher texts according to her in structure, which is what we want to avoid at the first place. Even poorer, if the proxy conspires with receiver, some form of sender's secret key can be recovered which decrypt senders (convertible) cipher texts without receivers can further help. That also means that the transformation key of proxy should be well protected. It moves the secure key storage requirement from the delegate to the proxy using PRE. Therefore, unwanted to let the proxy occur in the storage server. That will also be problematic since each decryption needs separate interaction with the proxy.

VI. Dynamic and Efficient Key Management for Access Hierarchies (2007): We start by discussing the most relevant study in the literature of security/cryptography. The cryptographic key assignment schemes aim to minimize the expense in storing and handling secret keys for purpose of general cryptographic use. Utilizing a tree structure, A key Using KAC for data sharing in cloud storage[1]. We call this as master-secret key to avoid confusion with the delegated key we will explain later. For ease, now we omit the presence of a decryption algorithm for the original data owner using the master-secret key. In our specific structures, we will show how the knowledge of the master- secret key allows a faster decryption than using Extract followed by decryption. For a given branch can be used to derive the keys of it's Just, granting the parent key indirectly permits all the keys of its descendant nodes. Proposed a method to generate a tree hierarchy of symmetric-keys by using repeated evaluations of pseudo random function/block-cipher on a fixed secret. The idea can be generalized from a tree to a graph. More progressive cryptographic key assignment schemes sup- port access policy that can be modeled by an acyclic graph or a cyclic graph. Most of these schemes produce keys for symmetric-key cryptosystems, even though the key derivations may require modular arithmetic as used in public-key cryptosystems, which are generally more expensive than symmetric-key operations such as pseudorandom function. We take the tree structure as an example. Suhas can first categorize the cipher- text classes agreeing to their subjects like Each node in the tree signifies a secret key, while the leaf nodes represents the keys for separate cipher text classes. Filled circles represent circles circumvented by dotted lines the keys for the classes to be delegated and represent the keys to be granted. Note that each key of the non-leaf node can develop the keys of its successor nodes. If Suhas wants to share all the files in the individual category, she only requires allowancing the key for the node 6 personal, which automatically grants the delegatee the keys of all the successor nodes (photo, music). This is the ideal case, where most classes to be shared fit to the same branch and therefore a parent key of them is sufficient[5].

VII. Fuzzy Identity-Based Encryption. Theory and Applications of Cryptographic Techniques (2005): 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). Private key is a trusted party generator in IBE which holds a master-secret key and is- sues a secret key to each user with respect to the identity of the user[6]. The encryptor can take the public parameter and a user identity to encrypt a message. by his secret key the receiver can decrypt this cipher text. One of their schemes assumes random oracles but alternative does not. In their systems, key aggregation is constrained in the sense that all keys to be aggregated must come from dissimilar identity partitions. While there are an exponential number of identities and thus secret keys, single polynomial number of them can be aggregated. Most importantly, their key aggregation comes at the expense of on sizes for both cipher texts and the public parameter, where is the number of secret keys which can be aggregated into a constant size one. These significant lyrics the costs of storing and transmitting cipher texts, which is impractical in various circumstances such as the shared cloud storage. By means of we stated, our schemes feature constant cipher text size, and their security grips in the typical model. In fuzzy IBE, one single compact secret key can decrypt cipher texts encrypted under several individualities which are near in a certain metric space, but not for an subjective set of identities and, therefore, it does not match with our idea of key aggregation.

B. Comparison of Existing System with Proposed System

Methods	**Existing System**	**Proposed System**
Technique	1) Key Attributed Based Encryption. 2) Multi Identity Single Key Decryption	Key Aggregate Encryption
Key	Symmetric	Asymmetric Key
Size of the Decryption Key	Constant Size decryption Key	Constant size Decryption Key
Relation between Classes	Required	Not Required

Table. Comparative Study on Existing vs. Proposed System

III. Proposed System

The design of The proposed system with an efficient public-key encryption. In this any number of subsection of the cipher text can be decrypted by using the decryption key. The problem is solved by the overview of key aggregate cryptosystem. In key Aggregate cryptosystem user will encrypt message not just in a public key but also beneath an identifier. These cipher texts are more categorized into classes. The owner will have the main secret key. The secret keys are extracted from the main secret key, these secrets keys are used to encrypt the files. The extracted key can be aggregate key which is as compact single key. By this explanation, sender shares the single aggregate key by means of a safe channel say email. The receiver downloads the encrypted files from senders drop box and then decrypts those files with single aggregate key. This state is shown in below Figure.

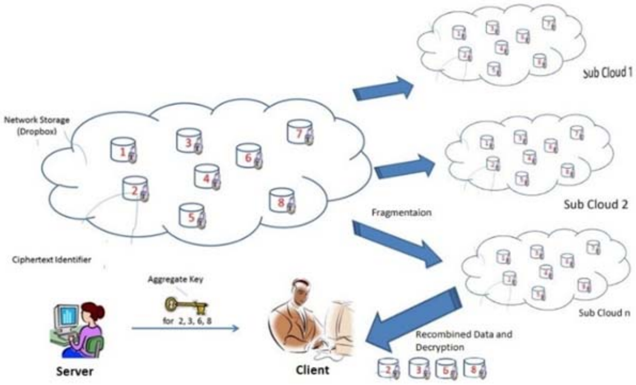

Fig. System Architecture

IV. Conclusion

In this work we have reviewed three authentication techniques: Attribute based encryption (ABE), Identity Based Encryption (IBE) and Key Aggregate Cryptosystem (KAC). The major concern in ABE is collusion resistance but not compression of secret keys. Certainly, the cipher text-size is not constant. In IBE, random set of individualities are not match with our design of key aggregation. Key Aggregate Cryptosystem defends user's data privacy by compressing the secret key in public key cryptosystem which supports delegation of secret key for dissimilar cipher text classes. For upcoming extension it is required to reserve sufficient cipher texts classes because in cloud cipher texts grows rapidly and the limitation is that predefined bound of the number of maximum cipher text classes. To share data flexibly is vital thing in cloud computing. Users favor to upload there data on cloud and among dissimilar users. Outsourcing of data to server may lead to leak the private data of user to everyone. Encryption is a one solution which provides to share designated data with wanted candidate. Sharing of decryption keys in secure way plays significant part. Public-key cryptosystems offers allocation of secret keys for dissimilar cipher text classes in cloud storage.

References

[1] Cheng-Kang Chu, Sherman S. M. Chow, Wen-Guey Tzeng, Jianying Zhou, Robert H. Deng, "Key-Aggregate Cryptosystem for Scalable Data Sharing in Cloud Storage", IEEE Transactions on Parallel & Distributed Systems, vol.25, no. 2, pp. 468-477, Feb. 2014, doi:10.1109/TPDS.2013.112

[2] Sagar Patil, Kirti Korabu, Novel Dynamic Key Aggregate Cryptosystem for Cloud Data Sharing, IRJET Vol.2 Issue 05, August 2015

[3] D. Boneh, R. Canetti, S. Halevi, and J. Katz, "Chosen-Ciphertext Security from Identity-Based Encryption," *SIAM J. Computing,* vol. 36, no. 5, pp. 1301-1328, 2007.

[4] J. Benaloh, "Key Compression and Its Application to Digital Fingerprinting," technical report, Microsoft Research, 2009.

[5] W.-G. Tzeng, "A Time-Bound Cryptographic Key Assignment Scheme for Access Control in a Hierarchy," *IEEE Trans. Knowledge and Data Eng.*, vol. 14, no. 1, pp. 182-188, Jan./Feb. 2002.

[6] A. Sahai and B. Waters, "Fuzzy Identity-Based Encryption," *Proc. 22nd Int'l Conf. Theory and Applications of Cryptographic Techniques (EUROCRYPT '05),* vol. 3494, pp. 457-473, 2005.

AUTHORS

Suhas S. Bachhav: Student of BE in Computer Engineering in SND Collage of Engineering and research Center, Babulgaon, Yeola, Dist. Nashik

Chetan R. Chaudhari: Student of BE in Computer Engineering in SND Collage of Engineering and research Center, Babulgaon, Yeola, Dist. Nashik

Nikhilesh V. Shinde: Student of BE inComputer Engineering in SND Collage of Engineering and research Center, Babulgaon, Yeola, Dist. Nashik

Poonam S. Kaloge: Student of BE in Computer Engineering in SND Collage of Engineering and research Center, Babulgaon, Yeola, Dist. Nashik