

Secure – Scalable Data Sharing In Cloud Environment

Mr. Balasundaram.B^{*1},

*M.E Student, Department of Computer Science and Engineering, Dhanalakshmi
Srinivasan college of Engineering and Technology.*

Email ID: bala_sundaram@outlook.com

Mrs. M. Thamizharasi^{*2} M. Tech.,

*Assistant Professor, Department of Computer Science and Engineering,
Dhanalakshmi Srinivasan college of Engineering and Technology.*

Email ID: m.thamizharasi@gmail.com

Abstract

Using cloud storage, users will remotely store their information and share their information through cloud service, while not the burden of native information storage and maintenance. This paper provides high security, with efficiency and flexibly share information in cloud atmosphere with new MES theme which verifies outsourced secret writing. Describe the new public key secret writing that produces constant-size cipher-text and mixture key and the information and key have incorporated along in painting secret writing. The secret is managed with the assistance of xml format. This tends to thought of the verifiability of the cloud transformation and provided a replacement technique to visualize the correctness of the transformation. The MES schemes with verifiable outsourced secret writing following the model outlined within the existing. It conjointly focuses on CP-ABE with verifiable outsourced secret writing.

Key words: Multi-Encryption Standard, Iconic encryption, data sharing, cloud storage.

Introduction

Cloud Computing has been envisioned as the next-generation architecture of IT environment, due to its long list of unprecedented Advantages in the IT history: ubiquitous network access, location independent resource pooling, on-demand self-service, rapid resource elasticity and usage-based pricing. As a disruptive technology with profound implications and Cloud Computing is transforming the very nature of how businesses used in IT Companies. One fundamental aspect of this paradigm shifting is that data is being centralized or outsourced into the Cloud. From users

perspective, storing data remotely into the cloud in a flexible on-demand manner brings appealing benefits: universal data access with independent geographical locations, avoidance of capital expenditure on, software's, hardware's and personnel maintenance, etc While these Advantages of using clouds are assured, due to the opaqueness of the Cloud—as separate administrative entities and the internal operation details of cloud service providers (CSP) may not be known by cloud users—data outsourcing is also relinquishing user's ultimate control over the fate of their data. As a result, the cloud storage provides security for data sharing in this project. The encryption techniques used in this project are AES, DES, MD5 and key management is main concept in this project. How they have sharing the key to another user or organization.

Related Work

Considering data privacy, a traditional way to ensure it is to rely on the server to enforce the access control after authentication, which means any unexpected privilege escalation will expose all data. In a shared-tenancy cloud computing environment, things become even worse. Regarding availability of files, there are a series of cryptographic schemes which go as far as allowing a third-party auditor to check the availability of files on behalf of the data owner without leaking anything about the data, or without compromising the data owner's anonymity. Likewise, cloud users probably will not hold the strong belief that the cloud server is doing a good job in terms of confidentiality. A cryptographic solution, with proven security relied on number-theoretic assumptions is more desirable, whenever the user is not perfectly happy with trusting the security of the VM or the honesty of the technical staff.

Hierarchical Access Control

We start by discussing the most relevant study in the literature of security. Cryptographic key assignment aim to minimize the expense in storing and managing secret keys for general cryptographic use. Utilizing a tree structure, a key for a given branch can be used to derive the keys of its descendant nodes. Just granting the parent key implicitly grants all the keys of its descendant nodes. They proposed a method to generate a tree hierarchy of symmetric-keys by using repeated evaluations of pseudorandom function/block cipher on a fixed secret. The concept can be generalized from a tree to a graph. More advanced cryptographic key assignment schemes support 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 Pseudo random function.

Key-Aggregate Encryption

A key-aggregate encryption have five polynomial-time algorithms as follows.

The data owner establishes the public system parameter via Setup and generates a public/master-secret key air Key Gen. Messages can be encrypted via Encrypt by

anyone who also decides what cipher-text class is associated with the plain-text message to be encrypted. The data owner use the master-secret to generate an aggregate decryption key for a set of cipher-text classes as Extract. The generated keys can be passed to delegates securely. Then any user with an aggregate key can decrypt any cipher-text provided that the cipher text's class is contained in the aggregate key Decrypt.

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. There is a trusted party called private key generator in IBE which holds a master-secret key and issues a secret key to each user with respect to the user identity. The encrypter can take the public parameter and a user identity to encrypt a message. The recipient can decrypt this cipher-text by his secret key.

One of their schemes assumes random oracles but another does not. In their schemes, 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. Most importantly, their key-aggregation comes at the expense of sizes for both cipher-texts and the public parameter, where n is the number of secret keys which can be aggregated into a constant size one. This greatly increases the costs of storing and transmitting cipher-texts, which is impractical in many situations such as shared cloud storage. As we mentioned, our schemes feature constant cipher-text size, and their security holds in the standard model. In fuzzy IBE. one single compact secret key can decrypt cipher-texts encrypted under many identities which are close in a certain metric space, but not for an arbitrary set of identities and, therefore, it does not match with our idea of key aggregation.

Proposed System

To design an efficient public-key encryption scheme this supports flexible delegation in the sense that any subset of the cipher-texts (produced by the encryption scheme) is decryptable by a constant-size decryption key (generated by the owner of the master-secret key). To recover this problem by introducing a special type of public-key encryption this called as key-aggregate cryptosystem (KAC). In KAC, users encrypt a message not only under a public-key, but also under an identifier of cipher-text.class. That means the cipher-texts are further categorized into different class. The key owner holds a master-secret key and master secret key used to extract secret keys for different class files. More important is to extract key can be an aggregate key which is as compact to a secret key for decrypt a single class, but aggregates the power of many keys, i.e., the decryption power for any subset of cipher-text class.

Techniques Used For Encryption

Multi-Encryption Standard (MES):

Multiple encryptions is the process of encrypting an already encrypted message one or more times, either using the same or a different algorithm. It is also known as cascade encryption, cascade ciphering, multiple encryption, and super encipherment.

Two encryption algorithms are

- Advanced Encryption Standard
- Data Encryption Standard

Advanced Encryption Standard (AES):

AES comprises three block ciphers, AES-128, AES-192 and AES-256. Each cipher encrypts and decrypts data in blocks of 128 bits using cryptographic keys of 128-bits, 192-bites and 256-bits, respectively. Symmetric or secret-key ciphers use the same key for encrypting and decrypting, so both the sender and the receiver must know and use the same secret key.

The encryption process uses a set of specially derived keys called round keys. These are applied along with other operations, on an array of data that holds exactly one block of data to be encrypted.

The steps to encrypt a 128-bit block:

1. Derive the set of round keys from the cipher key.
2. Initialize the state array with the block data (plaintext).
3. Add the initial round key to the starting state array.
4. Perform nine rounds of state manipulation.
5. Copy the final state array out as the encrypted data (ciphertext).

The round function consists of four transformation stages used in AES.

- a) Sub-Bytes()
- b) Shift Rows()
- c) Mix Columns()
- d) Add Round Key()

a. Sub-Bytes () Transformation

The substitute transformation is an S-Box process that is independent of the key. Each of the bytes of the State is replaced by a different byte, according to a table. The table is fixed and derived from two transformations defined in the standard. The table is an 8 x 8 array, indexed with the State byte.

b. Shift Rows () Transformation

The Shift Rows () transformation is a permutation that is performed row by row on the State array, independently of the key. The first row is not shifted. The 2nd row is circularly shifted left 1 byte. The 3rd row is circularly shifted left 2 bytes. The 4th row is circularly shifted left 3 bytes.

c. Mix Columns() Transformation

The Mix Columns () transformation manipulates each column of the state array. The process can be described as a matrix multiplication of a polynomial and the state array. This process does not depend on the key.

d. Add Round Key() Transformation

The Add Round Key() transformation uses the key schedule word. The process is a bitwise XOR of the columns of the state array, with the key schedule word.

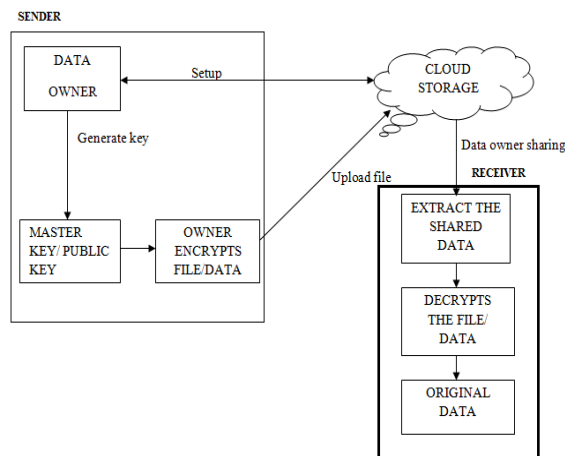
Data Encryption Standard (DES)

The encryption process is made of two permutations (P-boxes), which we call initial and final permutations, and sixteen Feistel rounds. The data encryption standard (DES) is a common standard for data encryption and a form of secret key cryptography (SKC), which uses only one key for encryption and decryption. Public key cryptography (PKC) uses two keys, i.e., one for encryption and one for decryption.

The four steps of DES Algorithm techniques.

- a) Initial Permutation
- b) DES function
- c) S- box
- d) Key generation

System Architecture



Modules Description

1. Access Control

In this module, the user registration process is done by the admin. Here every user's give their personal details for registration process. After registration every user will get an ID for accessing the cloud space. If any of the user wants to edit their

information they have submit the details to the admin after that the admin will do the edit and update information process. This process is controlled by the Admin.

2. Sharing Information

In this module, every user's share their information and data's in their own cloud space provided by the admin. That information may be sensitive or important data's. For providing security for their information every user's storing the information in their specific cloud. Registered users only can store the data in cloud.

3. Multi Encryption Process

In this module, the information and data's shared by the user in the cloud is encrypted by using MES (Multi Encryption Standard) algorithm. All the information shared by every user is encrypted based on the data sensitivity and stored in the cloud. Involves in client side configuration, performs two actions.

The two actions are access control and permission control.

- Access control – MES algorithm.
- Permission control – Iconic Encryption algorithm.

Access control process is based on the server control features.

Permission control process is based on the client control features.

4. Integrity Checking

Integrity checking is the process of comparing the encrypted information with altered cipher-text. If there is any change in detection a message will send to the user that the encryption process is not done properly. If there is no change in detection means then it will allow doing the next process. Integrity checking is mainly used for anti-malware controls. In this module, the encrypted data is decrypted by the user using the public key of owner of the data. Decryption is the process of converting cipher text into plain text. MES algorithm is used for encrypting and decrypting the information. The user can view the data and also can download the data with high security.

5. Data Forwarding

In this module, the encrypted data or information stored in the cloud is forwarded to another user account by using that user's public key. If any user wants to share their information with their friends or someone they can directly forward the encrypted data to them. Without downloading the data the user can forward the information to another user. Secure Data Forwarding is implemented by detecting flag generation where for sharing flags will be 0-1 and where for forwarding flags 1-1 are detected. Is flag 1-1 is detected then by applying Filtering technique data's are filtered out.

Conclusion

In this project achieves to protect user's data in cloud storage. With more mathematical tools, Key-aggregate cryptosystem techniques are getting more versatile and often involve multiple keys for a single application and the Multiple Encryption Standard is used to encrypt the data or message uploaded by delegatee. In this paper

consider how to “compress” secret keys in New-public-key cryptosystems which support delegation of secret keys for different cipher-text classes in cloud storage. The delegatee can always get an aggregate key of constant size. After downloading the file, they verify the key with the xml format in the file itself. If key is incorrect the attacker get invalid and corrupted file will be downloaded.

Future Enhancement

A limitation in the project is the predefined bound of the number of maximum cipher-text classes. In cloud storage, the number of cipher-texts usually grows rapidly. So have to reserve enough cipher text classes for the future extension.

References

- [1] Ashutosh Kumar Dubey , Animesh Kumar Dubey , Mayank Namdev, Shiv Shakti Shrivastava “ Cloud-User Security Based on RSA and MD5 Algorithm for Resource Attestation and Sharing in Java Environment”, IEEE ,2012
- [2] B R Ambedkar , Ashwani Gupta , Pratiksha Gautam , SS Bedi ,“An Efficient Method to Factorize the RSA Public Key Encryption”, IEEE 2011 International Conference on Communication Systems and Network Technologies.
- [3] B. Wang, S.S.M. Chow, M. Li, and H. Li, “Storing Shared Data on the Cloud via Security-Mediator,” Proc. IEEE 33rd Int’l Conf. Distributed Computing Systems (ICDCS), 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 systems, vol. 25, no. 2, February 2014.
- [5] 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, Feb. 2013.
- [6] F. Guo, Y. Mu, Z. Chen, and L. Xu, “Multi-Identity Single-Key Decryption without Random Oracles,” Proc. Information Security and Cryptology (Inscrypt ’07), vol. 4990, 2007.
- [7] J. Benaloh, M. Chase, E. Horvitz, and K. Lauter, “Patient Controlled Encryption: Ensuring Privacy of Electronic Medical Records,” Proc. ACM Workshop Cloud Computing Security (CCSW ’09), pp. 103-114, 2009.
- [8] M.J. Atallah, M. Blanton, N. Fazio, and K.B. Frikken, “Dynamic and Efficient Key Management for Access Hierarchies,” ACM Trans. Information and System Security, vol. 12, no. 3, 2009.
- [9] Mebae Ushida, Kouichi Itoh, Yoshinori Katayama, Fumihiko Kozakura and Hiroshi Tsuda, “A Proposal of Privacy-Preserving Data Aggregation

- on the Cloud Computing” ,2013 16th International Conference on Network-Based Information Systems.
- [10] Mercy Gnana Rani , Dr.A.Marimuthu, “Key Insertion and Splay Tree Encryption Algorithm for Secure Data Outsourcing in Cloud”, 2014 World Congress on Computing and Communication Technologies.
 - [11] Shu Qin Ren , Khin Mi Mi Aung , Jong Sou Park , “A Privacy Enhanced Data Aggregation Model”, 2010 10th IEEE International Conference on Computer and Information Technology (CIT 2010).
 - [12] Uma Somani, Kanika Lakhani, Manish Mundra “Implementing Digital Signature with RSA Encryption Algorithm to Enhance the Data Security of Cloud in Cloud Computing”, 2010 1st International Conference on Parallel, Distributed and Grid Computing (PDGC - 2010).