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

Using Two Components to Improve Factor Revocability and Data Security Protection Mechanism for Cloud Storage

Jyosthna Kumari Ponnuru
M.Tech (CSE),
Dept. of Computer Science & Engineering,
Lingayas Institute of Management and Technology,
A.P., India.

Khaleelullah Shaik
Assistant Professor,
Dept. of Computer Science & Engineering,
Lingayas Institute of Management and Technology,
A.P., India.

Abstract

Cloud computing is one of the hottest buzzwords in technologies. It provides access to its users to various services that it provides. Cloud computing is meant to satisfy requirement of data storage and data outsourcing for data owners. Though cloud service provides such services but security and privacy of owner's data is major concern in cloud storage.

Therefore secure data access is critical issue in cloud storage. In this paper Proposed system an improve data security protection mechanism for cloud using two components. In this system sender sends an encrypted message to a receiver with the help of cloud system. The sender requires to know identity of receiver but no need of other information such as certificate or public key. To decrypt the cipher text, receiver needs two parts. The first thing is a unique personal security device or some hardware device connected to the computer system. Second one is private key or secrete key stored in the computer.

Without having these two things cipher text never decrypted. The important thing is the security device lost or stolen, then cipher text cannot be decrypted and hardware device is revoked or cancelled to decrypt cipher text. The efficiency and security analysis show that the system is secure as well as practically implemented. The system uses a new hardware device like pen drive etc. to decrypt the cipher text together with the private key.

Keywords — Security, Two-Components, factor revocability, public Cloud Storage.

INTRODUCTION

In cloud computing, users store their data files in cloud servers. Thus, it is crucial to prevent unauthorized access to these resources and realize secure resource sharing. In traditional access control methods, we generally assume data owners and the storage server are in the same secure domain and the server is fully trusted. However, in the cloud computing environment, cloud service providers can be attacked by malicious attackers.

These attacks may leak the confidential information of users for commercial interests as the data owners commonly store decrypted data in cloud servers. How to realize access control to the encrypted data and ensure the confidentiality of data files of users in an untrusted cloud environment are the major problems.

Moreover, since the number of users is large in a cloud computing environment, how to realize scalable, flexible and fine-grained access control is strongly desired in the service-oriented cloud computing model.

Cloud computing provides shared processing environment for data storage and accessing also known as internetbased computing. It is a model which provides configurable computing resources such as networks, servers, storage, applications and services. Cloud computing has a high computation power, lowest cost of services, higher performance, scalability, accessibility and availability for that reason it is highly demanded. Data outsourcing brings with it many advantages. metadata and data are stored in cloud database and can access by client through encrypted database engine. Encrypted engine fetch required metadata to execute SQL queries from cloud database and decrypt it through master key which is with client side application. Adaptive encryption scheme consider many SQL aware encryption algorithm such as Random, Deterministic which supports equality operators, order preserving encryption, homomorphic sums, plain and search. Adaptive encryption scheme consider many SQL aware encryption algorithm such as Random,Deterministic which supports equality operators, order preserving encryption, homomorphic sums,plain and search.

If each column is encrypted through only one algorithm then administrator has to decide database operations at design time only for each column. Here encryption algorithms are organized into structure called onions, where each onion is made up of ordered set of encryption algorithm called layer. Onions layers are used for equality, comparison, summation, string equality operators.

Each plaintext column is encrypted into one or more encrypted column each one corresponding to an onion. Each plain text is encrypted through all the layers of its onion i.e.encrypted through more than on encryption algorithm. Thought this approach provides more adaptive mechanism for accessing cloud database, access policies are assigned by data owner or single authority only which can result in system bottleneck. Multi-User Encrypted SQL Operation on Cloud approach provides scalable and confidential access to cloud database. This architecture called MultiUser relational Encrypted Data Base (Mute DB) that guarantees data confidentiality by executing SQL operation on data by applying access control policies. The Mute DB does not rely on any intermediate proxy to avoid single point bottleneck.

Here every data and metadata is stored on cloud in encrypted format. Here data managed and create by DBA, who is also responsible storing encrypted data and metadata on the cloud. DBA is the trusted entity who owns root credentials, manages user accounts and enforces access control policies. This ACP defines which user can have access on which data. Each user will be provided set of credentials including all the information that allows him/her to access legitimate data. In this case access policies are also encrypted and stored in cloud. The DBA is the only authority who can have control on all system entity; this can leads toward DBA overloading and can resulton performance degradation.

IMPLEMENTATION

Two-Factor Data Security Protection Mechanism For Cloud Storage System:

This algorithm allows a sender to send an encrypted message to a receiver through a cloud storage server. The sender only needs to know the identity of the receiver but no other information (such as its public key or its certificate). The receiver needs to possess two things in order to decrypt the cipher text. The first thing is his/her secret key stored in the computer. The second thing is a unique personal security device which connects to the computer. It is impossible to decrypt the cipher text without either piece.

Approach:

The encryption process is executed twice. First encrypt the plaintext corresponding to the public key or identity of the user. Then encrypt it again corresponding to the public key or serial number of the security device. For the decryption stage, the security device first decrypts once. The partially decrypted ciphertext is then passed to the computer which uses the user secret key to further decrypt it. Without either part (user secret key or security device) one cannot decrypt the ciphertext. If the user has lost his security device, then his/her corresponding ciphertext in the cloud cannot be decrypted forever! That is, the approach cannot support security device update/ revocability.

Real world implemented example: At AT&T labs, in a druva system, a message is first encrypted under a user key k1, and next uploaded to a cloud server. The user key k1 is further encrypted by another user key k2, and stored in the server as well. The key k2 is held by the user. When retrieving the message, the user needs to use k2 to recover k1 which is further used to recover m. It is undeniable that this message-key encrypt mechanism is much better than the mode only using a single key to encrypt an outsourced data, and storing the ciphertext along with the key in the server. Nevertheless, this mechanism suffers from a potential risk in practice. Once the user loses the key k2, all data of the user stored in the cloud cannot be retrieved. The lack of revocability for encryption factor limits the flexibility of the system

RELATED WORK

Our system is an IBE (Identity-based encryption)-based mechanism. That is, the sender only needs to know the identity of the receiver in order to send an encrypted data (ciphertext) to him/her. No other information of the receiver (e.g. public key, certificate etc.) is required. Then the sender sends the ciphertext to the cloud where the receiver can download it at anytime.

Our system provides two-factor data encryption protection. In order to decrypt the data stored in the cloud, the user needs to possess two things. First, the user needs to have his/her secret key which is stored in the computer. Second, the user needs to have a unique personal security device which will be used to connect to the computer (e.g. USB, Bluetooth and NFC). It is impossible to decrypt the ciphertext without either piece.

More importantly, our system, for the first time, provides security device (one of the factors) revocability. Once the security device is stolen or reported as lost, this device is revoked. That is, using this device can no longer decrypt any ciphertext (corresponding to the user) in any circumstance. The cloud will immediately execute some algorithms to change the existing ciphertext to be un-decryptable by this device. While the user needs to use his new / replacement device (together with his secret key) to decrypt his/her ciphertext. This process is completely transparent to the sender.

The cloud server cannot decrypt any ciphertext at any time. We provide an estimation of the running time of our prototype to show its practicality, using some benchmark results. We also note that although there exist some naive approaches that seem to achieve our goal, that there are many limitations by each of them and thus we believe our mechanism is the first to achieve all the above mentioned features in the literature.

User Revocation Based ABE ALGORITHM:

The concept of **attribute based encryption** is a type of public-key encryption in which the secret key of a user and the ciphertext are dependent about attributes. In a system, the decryption of a cipher text is possible only if the set of attributes of the user key matches the attributes of the cipher text. A crucial security feature of Attribute-Based Encryption is collusion-resistance: An adversary that holds multiple keys should only be able to access data if at least one individual key grants access.

Step 1: Select File attribute1 – say File name
Step 2: Convert the file name to Binary Codes
Step 3: Select File attribute 2 – say file size
Step 4 : Convert the file size to Binary Codes
Step 5: Perform AND Operation of File Attribute 1 and 2
Step 6: Perform OR Operation of File Attribute 1 and 2
Step 7: Result of AND Operation Stored as Secret Key
Step 8: Result of OR Operation Stored as Public Key

LITERATURE SURVEY

1) Improving privacy and security in multi-authority attribute-based encryption

AUTHORS: M. Chase and S. S. M. Chow

Attribute based encryption (ABE) [13] determines decryption ability based on a user's attributes. In a multi-authority ABE scheme, multiple attribute-authorities monitor differ-ent sets of attributes and issue corresponding decryption keys to users, and encryptors can require that a user ob-tain keys for appropriate attributes from each authority be-fore decrypting a message. Chase [5] gave a multi-authority ABE scheme using the concepts of a trusted central author-ity (CA) and global identifiers (GID). However, the CA in that construction has the power to decrypt every ciphertext, which seems somehow contradictory to the original goal of distributing control over many potentially untrusted author-ities. Moreover, in that construction, the use of a consistent GID allowed the authorities to combine their information to build a full profile with all of a user's attributes, which unnecessarily compromises the privacy of the user. In this paper, we propose a solution which removes the trusted central authority, and protects the users ' privacy by preventing the authorities from pooling their information on particular users, thus making ABE more usable in practice.

2) Secure threshold multi authority attribute based encryption without a central authority

AUTHORS: H. Lin, Z. Cao, X. Liang, and J. Shao

An attribute based encryption scheme (ABE) is a cryptographic primitive in which every user is identified by a set of attributes, and some function of these attributes is used to determine the ability to decrypt each ciphertext. Chase proposed the first multi authority ABE scheme in TCC 2007 as an answer to an open problem presented by Sahai and Waters in EUROCRYPT 2005.

However, her scheme needs a fully trusted central authority which can decrypt every ciphertext in the system. This central authority would endanger the whole system if it's corrupted. This paper presents a threshold multi authority fuzzy identity based encryption (MA-FIBE) scheme without a central authority for the first time. An encrypter can encrypt a message such that a user could only decrypt if he has at least d k of the given attributes about the message for at least $t+1$, $t \le n/2$ honest authorities of all the n attribute authorities in the proposed scheme.

The security proof is based on the secrecy of the underlying joint random secret sharing protocol and joint zero secret sharing protocol and the standard decisional bilinear Diffie-Hellman assumption. The proposed MA-FIBE could be extended to the threshold multi authority attribute based encryption (MA-ABE) scheme and be further extended to a proactive MA-ABE scheme.

3) Multi-authority attribute-based encryption with honest-but-curious central authority

AUTHORS: V. Božovi´c, D. Socek, R. Steinwandt, and V. I. Villányi

An attribute-based encryption scheme capable of handling multiple authorities was recently proposed by Chase. The scheme is built upon a single-authority attribute-based encryption scheme presented earlier by Sahai and Waters. Chase's construction uses a trusted central authority that is inherently capable of decrypting arbitrary ciphertexts created within the system. We present a multi-authority attribute-based encryption scheme in which only the set of recipients defined by the encrypting party can decrypt a corresponding ciphertext. The central authority is viewed as 'honest-but-curious': on the one hand, it honestly follows the protocol, and on the other hand, it is curious to decrypt arbitrary ciphertexts thus violating the intent of the encrypting party.

The proposed scheme, which like its predecessors relies on the Bilinear Diffie–Hellman assumption, has a complexity comparable to that of Chase's scheme. We prove that our scheme is secure in the selective ID model and can tolerate an honest-but-curious central authority.

4) Attribute-based secure data sharing with hidden policies in smart grid

AUTHORS: J. Hur

Smart grid uses intelligent transmission and distribution networks to deliver electricity. It aims to improve the electric system's reliability, security, and efficiency through two-way communication of consumption data and dynamic optimization of electric-system operations, maintenance, and planning. The smart grid systems use fine-grained power grid measurements to provide increased grid stability and reliability. Key to achieving this is securely sharing the measurements among grid entities over wide area networks. Typically, such sharing follows policies that depend on data generator and consumer preferences and on time-sensitive contexts. In smart grid, as well as the data, policies for sharing the data may be sensitive because they directly contain sensitive information, and reveal information about underlying data protected by the policy, or about the data owner or recipients. In this study, we propose an attribute-based data sharing scheme in smart grid.

Not only the data but also the access policies are obfuscated in grid operators' point of view during the data sharing process. Thus, the data privacy and policy privacy are preserved in the proposed scheme. The access policy can be expressed with any arbitrary access formula. Thus, the expressiveness of the policy is enhanced. The security is also improved such that the unauthorized key generation center or the grid manage systems that store the data cannot decrypt the data to be shared. The computation overhead of recipients are also reduced by delegating most of the laborious decryption operations to the more powerful grid manage systems.

CONCLUSION&FUTURE WORK

In this paper, In this paper, we introduced a novel two-factor data security protection mechanism for cloud storage system, in which a data sender is allowed to encrypt the data with knowledge of the identity of a receiver only, while the receiver is required to use both his/her secret key and a security device to gain access to the data. Our solution not only enhances the confidentiality of the data, but also offers the revocability of the device so that once the device is revoked, the corresponding ciphertext will be updated automatically by the cloud server without any notice of the data owner. Furthermore, we presented the security proof and efficiency analysis for our system.

To minimize the communication and computation overhead ECC-128 bit algorithm is proposed. And due to fault tolerance problem in cloudserver we are maintain the another copy of sender data in different cloud server .in case deleted file is regenerated from other cloud server. Our solutionnot only enhances the communication and computation overhead, but also offers the regeneration of the corrupted data.

References

Good Teachers are worth more than thousand books, we have them in Our Department.

[1] A. Akavia, S. Goldwasser, and V. Vaikuntanathan. Simultaneous hardcore bits and cryptography against memory attacks. In TCC, volume 5444 of Lecture Notes in Computer Science, pages 474–495. Springer, 2009.

[2] S. S. Al-Riyami and K. G. Paterson. Certificateless public key cryptography. In ASIACRYPT, volume 2894 of Lecture Notes in Computer Science, pages 452–473. Springer, 2003.

[3] M. H. Au, J. K. Liu, W. Susilo, and T. H. Yuen. Certificate based (linkable) ring signature. In ISPEC, volume 4464 of Lecture Notes in Computer Science, pages 79–92. Springer, 2007.

[4] M. H. Au, Y. Mu, J. Chen, D. S. Wong, J. K. Liu, and G. Yang. Malicious kgc attacks in certificateless cryptography. In ASIACCS, pages 302–311. ACM, 2007.

[5] M. Blaze, G. Bleumer, and M. Strauss. Divertible protocols and atomic proxy cryptography. In K. Nyberg,

editor, EUROCRYPT, volume 1403 of LNCS, pages 127–144. Springer, 1998.

[6] A. Boldyreva, V. Goyal, and V. Kumar. Identity-based encryption with efficient revocation. In P. Ning, P. F. Syverson, and S. Jha, editors, ACM Conference on Computer and Communications Security, pages 417–426. ACM, 2008.

[7] D. Boneh, X. Ding, and G. Tsudik. Fine-grained control of security capabilities. ACM Trans. Internet Techn., 4(1):60–82, 2004.

[8] D. Boneh and M. Franklin. Identity-based encryption from the weil pairing. In CRYPTO '01, volume 2139 of LNCS, pages 213– 229. Springer, 2001.

[9] R. Canetti and S. Hohenberger. Chosen-ciphertext secure proxy re-encryption. In P. Ning, S. D. C. di Vimercati, and P. F. Syverson, editors, ACM Conference on Computer and Communications Security, pages 185–194. ACM, 2007.

[10] C.-K. Chu, S. S. M. Chow, W.-G. Tzeng, J. Zhou, and R. H. Deng. Key-aggregate cryptosystem for scalable data sharing in cloud storage. IEEE Trans. Parallel Distrib. Syst., 25(2):468–477, 2014.

Author Details:

Ms. Jyosthna Kumari Ponnuru is a student of Lingayas Institute of Management and Technology, Madalavarigudem, Nunna. She is presently pursuing his M.Tech degree from JNTU, Kakinada.

Mr. Khaleelullah Shaik is presently working as Assistant professor in CSE department, Lingayas Institute of Management and Technology, Madalavarigudem, Nunna.