Journal of **N**onlinear **A**nalysis and **O**ptimization
Vol. 13, Issue. 2 : 2022
ISSN : **1906-9685**

KEY-AGGREGATE CRYPTOSYSTEM FOR SCALABLE DATA SHARE AND CLOUD STORAGE

[#1]Mr.BOTLA RAMMURTHY, *Assistant Professor*
[#2]Mrs.BHEERAM SANKEERTHANA, *Assistant Professor*
Department of Computer Science and Engineering,
SREE CHAITANYA INSTITUTE OF TECHNOLOGICAL SCIENCES, KARIMNAGAR, TS.

ABSTRACT: Data sharing is an essential element of cloud storage. This post will illustrate the effective utilization of cloud storage for sharing data with others in a versatile, safe, and streamlined manner. We explore novel public-key cryptosystems that produce ciphertexts of consistent size, simplifying the process of assigning decryption keys for any set of ciphertexts. An intriguing aspect is that a set of confidential keys can be merged into a singular key without compromising their individual capabilities. Essentially, the possessor of the secret key has the ability to share a compact aggregate key for a versatile collection of encrypted data in cloud storage, all the while preserving the secrecy of the remaining encrypted files. This small aggregate key can be communicated to others or saved on a smart card with low safe storage needs. Our systems undergo a rigorous security analysis utilizing the standard approach. In addition, we explore potential uses for our systems. Our methods offer a novel form of encryption, where patients have control over their own encryption keys and can establish a hierarchical structure according to their needs.

Keywords: Data sharing, public key Cryptosystem, Encryption, Decryption, Aggregate key

1. INTRODUCTION

In recent times, there has been a surge in the use of cloud storage. The practice of outsourcing data is becoming common in business settings, facilitating strategic data management. Furthermore, it serves as the fundamental technology for several individual internet services. Registering for complimentary email accounts that offer ample storage space, photo albums, file sharing, and/or remote access has become uncomplicated.

With the help of a mobile phone and advanced wireless technology, users have the ability to remotely access nearly all of their files and emails from any location on Earth. The server is often used to enforce access control after authentication in relation to data privacy. This means that any unexpected increase in privileges could lead to the exposure of all data.

The condition of a cloud computing system with several tenants declines rapidly. Data belonging to many customers can be kept on separate virtual machines (VMs) within a single physical system. By deploying an additional virtual machine (VM) that shares the same physical server as the target VM, it is possible to illicitly extract data from the target VM.

Various cryptographic mechanisms exist to facilitate file accessibility verification by a third-party auditor on behalf of the data owner, while ensuring data confidentiality and preserving the data owner's identity. Similarly, cloud customers are unlikely to have a strong belief that their data is secure on the cloud server. If the user has doubts about the security of the virtual machine or the trustworthiness of the technical team, it is advisable to employ a cryptographic solution that relies on known security measures based on number theory assumptions.

These users are advised to employ encryption using their personal keys before to transmitting their data to the server. Data sharing is an essential element of cloud storage. Bloggers, for example, may provide their peers with

permission to view a portion of their personal photographs, while corporations may provide their employees with permission to read a piece of their classified information.

Sharing encrypted data is a significant challenge in terms of effectiveness. Users have the ability to get encrypted material from storage, decrypt it, and subsequently distribute it to others. However, engaging in such actions undermines the fundamental objective of cloud storage. Users should have the ability to authorize expedited retrieval of shared data from the server by providing others permission to access the shared data. Finding a trustworthy and safe way to share a specific portion of data stored in the cloud is challenging.

2. LITERATURE SURVEY

This paper introduces SPICE, the inaugural digital identity management system, along with additional desirable attributes. The uniqueness of our technique lies in its ability to merge and exploit two group signatures, enabling us to provide randomness to the signature and create the perception of novelty with each use, all the while concealing portions of the messages that are unrelated to the Cryptographic Service Provider (CSP). Due to its effectiveness and straightforwardness, our technology is highly suitable for cloud-based applications.

This work introduces a mechanism for private public auditing in a secure cloud storage system. To enhance efficiency, we propose expanding the scope of the TPA's audits to encompass many clients concurrently. Based on a thorough examination of security and performance, the suggested solutions are shown to be secure and highly efficient. The initial testing conducted on an Amazon EC2 instance verifies the effective performance of the design.

This work presents a straightforward and efficient approach for publicly verifying the integrity of cloud data, while both protecting the privacy of data owners and minimizing the need for extensive verification information. We have created a security-mediator (SEM) that is capable of producing signatures and verification information for data owners on externally stored data.

This study investigates the challenges associated with developing a robust cloud storage system that can effectively accommodate changing users and maintain accurate records of data origin. In order to put our idea into action, we establish rigorous security assurances against adaptive chosen-ciphertext decryption and update attacks, and enhance broadcast encryption with the ability to dynamically update ciphertexts.

The authors of this study established the notion of an aggregate signature, along with security models and a wide range of applications for aggregate signatures. Boneh, Lynn, and Shacham devised an effective aggregate signature by employing a newly discovered concise signature technique that relies on bilinear maps. Secure routing systems, like SBGP, can utilize aggregate signatures to decrease the size of messages and certificate chains by combining all signatures within the chain.

This study showcases the authors' approach to handling "limited depth" and reverse inheritance modifications to Crampton's [2003] conventional hierarchies.

The study examined the utilization of access control and encryption as means to enforce system security. To safeguard patients' privacy in case of a data center breach, they also explored techniques that allow patients to create and keep encryption keys.

This work introduces a new scheme called MISKD and verifies its security by utilizing the q-BSDH in the random oracle model of the Bilinear Strong Diffie-Hellman issue. This work introduces a new technique called MISKD, which is based on the Decisional Bilinear Diffie-Hellman (DBDH) assumption. The DBDH assumption has been proven to be secure in the selective-ID situation. Our approach effectively decrypts data.

Key-Policy Attribute-Based Encryption (KP-ABE) is a novel cryptographic method that allows for the secure sharing of encrypted data with fine-grained access control. Our cryptosystem utilizes attribute sets to identify ciphertexts, while private keys are associated with access structures that dictate the user's ability to decrypt specific ciphertexts. We showcase the adaptability of our architecture by exchanging audit-log data and implementing broadcast encryption. By integrating HIBE

(hierarchical identity-based encryption) into our solution, we are able to facilitate the transfer of private keys.

Individuals and data entities are categorized into security classes and arranged in a hierarchical structure, with the highest level of privilege at the top. Every user possesses an exclusive, unchanging key of a specific size that corresponds to their level of security. The proposed approach utilizes conventional cryptosystems rather than public key cryptosystems.

The study's authors introduced a multi-group key management technique that enables hierarchical access control by utilizing an integrated key graph and administering group keys to users with different access capabilities. With an increasing number of access levels, the suggested method offers enhanced scalability and significantly decreases the burden on communication, computation, and storage related to key management.

3. PROPOSED METHODOLOGY

This endeavor is partitioned into five segments. MK-Key enables the processes of data encryption and uploading, data sharing, key generation, and data decryption.

- Identity User Registration
- Data Encryption and Data Uploading
- Data Sharing
- Key Generation
- Data Decryption using MK-Key

Identity User Registration:

The data owner must create an account on a server that is not considered trustworthy. Every user individually registers their unique identity and acquires a public key to transmit data to a cloud server. The configuration technique necessitates solely the implicit security parameter as input. It generates both the master key (MK) and the public parameters (PK), which is also known as the public key.

Data Encryption and Data Uploading:

The data owner utilizes symmetric data encryption to encrypt the data elements using the symmetric encryption key PK, in compliance with the access control policy. In this situation, the data is encrypted using a public key (PK). Data proprietors utilize their public key to upload encrypted data objects to the cloud.

Data Sharing:

This module enables the transfer of data among users. The data owner has the authority to authorize another user to access encrypted data stored in the cloud. In this case, the data owner selects the data to be distributed and subsequently compresses that selected data. KAC is used to share data.

Key Generation:

The data owner utilizes this module to create a set of public and master-secret keys (pk, msk). Utilizing public keys is essential for encrypting data. After making the decision on which data to distribute. Create a master key for the designated encrypted data. Subsequently, this key is conveyed through a secure medium to the user.

Data Decryption using MK-Key:

This module utilizes an effective public-key encryption technique that allows for versatile delegation by allowing any subset of the cipher texts (created by the encryption method) to be deciphered using a decryption key of fixed size, generated by the holder of the master-secret key. The user is given the master key to decipher the encrypted text. The user is notified of the message M when the decryption process employs the Master Key to decrypt the ciphertext CT.

4. RESULT AND DISCUSSION

To enhance the effectiveness of a decryption key without increasing its size, one can enable it to decipher several cipher messages. The key-aggregate cryptosystem (KAC) is a new type of public-key encryption that was created by developing the key-aggregate cryptosystem (KAC). KAC users employ a public key and a cipher text identity to encrypt messages.

The master-secret key is the exclusive key possessed by the key proprietor and can be utilized to acquire secret class keys. Furthermore, the resulting key can have a minimal size, consisting of only one secret key from a certain class, but still possessing the combined decryption capabilities of numerous keys. Alternatively, it can encompass any subset of cipher text classes. All the components of our KAC schemes, namely the

aggregate key, master-secret key, public key, and cipher text, possess fixed quantities. The magnitude of the public system parameter is directly related to the quantity of cipher text classes, however, it can be obtained upon request from a vast (but non-confidential) cloud storage as only a small piece of it is ever required.

Our technique obviates this prerequisite, therefore rendering a distinct linkage between classes unnecessary.

Encrypted data can be distributed using public key aggregation, as the key size remains constant and there is no need for a specific link between the classes.

5. CONCLUSION

The preservation of user data confidentiality is a major concern in relation to cloud storage. With the increasing availability of mathematical tools, cryptographic methods are becoming more adaptable, leading to the frequent usage of numerous keys for a single application. This study focuses on investigating the process of "compressing" confidential keys in public-key cryptosystems that allow for the delegation of secret keys across several categories of encrypted data in cloud storage. The delegate can consistently obtain a fixed-size aggregate key, independent of the specific class chosen from the power set of classes. Our technique offers greater flexibility compared to hierarchical key assignment, as it can only save space if all key holders have identical privileges.

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," Proc. 10th Int'l Conf. Applied Cryptography and Network Security (ACNS), vol. 7341, pp. 526-543, 2012.
2. L. Hardesty, Secure Computers Aren't so Secure. MIT press, http:// www.physorg.com/news176107396.html, 2009. [3] 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.
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. S.S.M. Chow, C.-K. Chu, X. Huang, J. Zhou, and R.H. Deng, "Dynamic Secure Cloud Storage with Provenance," Cryptography and Security, pp. 442-464, Springer, 2012.
5. D. Boneh, C. Gentry, B. Lynn, and H. Shacham, "Aggregate and Verifiably Encrypted Signatures from Bilinear Maps," Proc. 22nd Int'l Conf. Theory and Applications of Cryptographic Techniques (EUROCRYPT '03), pp. 416-432, 2003.
6. 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, pp. 18:1-18:43, 2009.
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. 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, pp. 384-398, 2007.
9. V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-Based Encryption for Fine-Grained Access Control of Encrypted Data," Proc. 13th ACM Conf. Computer and Comm. Security (CCS '06), pp. 89-98, 2006.
10. S.G. Akl and P.D. Taylor, "Cryptographic Solution to a Problem of Access Control in a Hierarchy," ACM Trans. Computer Systems, vol. 1, no. 3, pp. 239-248, 1983.