IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

KEY-AGGREGATE CRYPTOSYSTEM FOR SCALABLE DATA SHARING IN CLOUD STORAGE

SELVALAKSHMI.K[1], Dr. MARY SHYLA.E[2]

1.Student, Nirmala College for Women, Red Fields, Coimbatore, Tamil Nadu, India

2.Assistant Professor, Nirmala College for Women, Red Fields, Coimbatore, Tamil Nadu, India

ABSTRACT:

Data sharing is an important functionality in cloud storage. In this paper, we show how to securely, efficiently, and flexibly share data with others in cloud storage. We describe new public-key cryptosystems that produce constant-size cipher texts such that efficient delegations of decryption rights for any set of cipher texts are possible. The novelty is that one can aggregate any set of secret keys and make them as compact as a single key, but encompassing the power of all the keys being aggregated. In other words, the secret key holder can release a constant-size aggregate key for flexible choices of cipher text set in cloud storage, but the other encrypted files outside the set remain confidential. This compact aggregate key can be conveniently sent to others or be stored in a smart card with very limited secure storage. We provide formal security analysis of our schemes in the standard model. We also describe other application of our schemes. In particular, our schemes give the first public-key patient-controlled encryption for flexible hierarchy, which was yet to be known.

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

INTRODUCTION

Cloud storage is gaining popularity recently. In enterprise settings, we see the rise in demand for data outsourcing, which assists in the strategic management of corporate data. It is also used as a core technology behind many online services for personal applications. Nowadays, it is easy to apply for free accounts for email, photo album, file sharing and/or remote access, with storage size. Together with the current wireless technology, users can access almost all of their files and emails by a mobile phone in any corner of the world. 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. Data from different clients can be hosted on separate virtual machines (VMs) but reside on a single physical machine. Data in a target VM could be stolen by instantiating another VM coresident with the target one. 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 owners 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, for example, 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. These users are motivated to encrypt their data with their own keys before uploading them to the server. Data sharing is an important functionality in cloud storage. For example, bloggers can let their friends view a subset of their private pictures; an enterprise may grant her employees access to a portion of sensitive data. The challenging problem is how to effectively share encrypted data. Of course users can download the encrypted data from the storage, decrypt them, then send them to others for sharing, but it loses the value of cloud storage. Users should be able to delegate the access rights of the sharing data to others so that they can access these data from the server directly. However, finding an efficient and secure way to share partial data in cloud storage is not trivial.

LITERATURE SURVEY

This paper [1] presents SPICE – the first digital identity management system that can satisfy these properties in addition to other desirable properties. The novelty of our scheme stems from combining and exploiting two group signatures so that we can randomize the signature to make the same signature look different for multiple uses of it and hide some parts of the messages which are not the concerns of the CSP. Our scheme is quite applicable to cloud systems due to its simplicity and efficiency.

In this paper [2], we propose a secure cloud storage system supporting privacy-preserving public auditing. We further extend our result to enable the TPA to perform audits for multiple users simultaneously and efficiently. Extensive security and performance analysis show the proposed schemes are provably secure and highly efficient. Our preliminary experiment conducted on Amazon EC2 instance further demonstrates the fast performance of the design.

In this paper [3], we propose a simple and efficient publicly verifiable approach to ensure cloud data integrity without sacrificing the anonymity of data owners nor requiring significant verification metadata. Specifically, we introduce a security-mediator (SEM), which is able to generate verification metadata (i.e., signatures) on outsourced data for data owners.

This paper [4] addresses the problem of building a secure cloud storage system which supports dynamic users and data provenance.. To realize our concept, we equip the broadcast encryption with the dynamic ciphertext update feature, and give formal security guarantee against adaptive chosen-ciphertext decryption and update attacks.

The authors of this paper [5] introduced the concept of an aggregate signature, present security models for such signatures, and give several applications for aggregate signatures. They constructed an efficient aggregate signature from a recent short signature scheme based on bilinear maps due to Boneh, Lynn, and Shacham. Aggregate signatures are useful for reducing the size of certificate chains (by aggregating all signatures in the chain) and for reducing message size in secure routing protocols such as SBGP.

The authors of this paper [6] show how to handle extensions proposed by Crampton [2003] of the standard hierarchies to "limited depth" and reverse inheritance.

The paper [7] explored that security in such systems should be enforced via encryption as well as access control. Further they explained about the approaches that enable patients to generate and store encryption keys, so that the patients' privacy is protected should the host data center be compromised.

They proposed a concrete MISKD scheme in this paper [8] and proved its security based on the Bilinear Strong Diffie-Hellman problem (q-BSDH) in random oracle model. In this paper, we present a novel MISKD scheme that is provably secure in the selective-ID model based on the Decisional Bilinear Diffie-Hellman (DBDH) assumption. Our scheme is more efficient in decryption.

Developed a new cryptosystem for fine-grained sharing of encrypted data that is Key-Policy Attribute-Based Encryption (KP-ABE) in the paper [9]. In our cryptosystem, ciphertexts are labeled with sets of attributes and private keys are associated with access structures that control which ciphertexts a user is able to decrypt. We demonstrate the applicability of our construction to sharing of audit-log information and broadcast encryption. Our construction supports delegation of private keys which subsumes Hierarchical Identity-Based Encryption (HIBE).

A cryptographic implementation is proposed [10] for access control in a situation where users and information items are classified into security classes organized as a rooted tree, with the most privileged security class at the root. Each user stores a single key of fixed size corresponding to the user's security class. The scheme proposed here is based on conventional cryptosystems (as opposed to public key cryptosystems).

The authors of the paper [11] presented a multi-group key management scheme that achieves such a hierarchical access control by employing an integrated key graph and by managing group keys for all users with various access privileges. The proposed scheme significantly reduces the communication, computation and storage overhead associated with key management and achieves better scalability when the number of access levels increases

PROPOSED METHODOLOGY

This project consists of five modules. They are,

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

Identity User Registration:

The data owner has to setup an account on an untrusted server.Each user registers its own identity and gets public key to the upload data in the cloud server. The setup algorithm takes no input other than the implicit security parameter. It outputs the public parameters are PK (public key) and a master key MK.

Data Encryption and Data Uploading:

According to the access control policy, the data owner uses a symmetric data encryption and encrypts the data items using symmetric encryption key PK. Here public key (PK) is used to encrypt the data. Data owner uploads encrypted data items using public key to the cloud.

Data Sharing:

This module facilitates user to share the data. Data owner can share the encrypted data from the cloud storage to some other user. Here data owner selects the data to share and compress the selected data. KAC is used for data sharing.

Key Generation:

This module is executed by the data owner to randomly generate a public/master-secret key pair pk; msk. Public key is used for data encryption. After the selection of data to be share. Master key is generated for the selected encrypted data. And this key is sent to the user using secure channel.

Data Decryption using MK-Key:

This module takes an efficient public-key encryption scheme which 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. User receives master key for decrypting the cipher text. The decryption algorithm takes the input cipher text CT and decrypts it using the Master Key and return a message M to the user.

RESULT AND DISCUSSION

The solution to how to make decryption key more powerful in the sense that it allows decryption of multiple cipher texts, without increasing its size. By introducing a special type of public-key encryption which we call key-aggregate cryptosystem (KAC). In KAC, users encrypt a message not only under a public-key, but also under an identifier of cipher text called class. 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 cipher text classes. The sizes of cipher text, public-key, master-secret key, and aggregate key in our KAC schemes are all of constant size. The public system parameter has size linear in the number of cipher text 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. Our work is flexible in the sense that this constraint is eliminated, that is, no special relation is required between the classes.

- We can share encrypted files also using public key aggregation.
- Key size is constant.
- No special relation is required between the classes.

CONCLUSION

How to protect users' data privacy is a central question of cloud storage. With more mathematical tools, cryptographic schemes are getting more versatile and often involve multiple keys for a single application. In this project, we consider how to "compress" secret keys in public-key cryptosystems which support delegation of secret keys for different ciphertext 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. Our approach is more flexible than hierarchical key assignment which can only save spaces if all key-holders share a similar set of privileges.

FUTURE WORK

A limitation in our work is the predefined bound of the number of maximum ciphertext classes. In cloud storage, the number of ciphertexts usually grows rapidly. So we have to reserve enough ciphertext classes for the future extension. Although the parameter can be downloaded with ciphertexts, it would be better if its size is independent of the maximum number of ciphertext classes. On the other hand, when one carries the delegated keys around in a mobile device without using special trusted hardware, the key is prompt to leakage, designing a leakage-resilient cryptosystem yet allows efficient and flexible key delegation is also an interesting direction.

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.

[11] G.C. Chick and S.E. Tavares, “Flexible Access Control with Master Keys,” Proc. Advances in Cryptology (CRYPTO ’89), vol. 435, pp. 316-322, 1989.