Vandana Publications
IJEMR

Volume-7, Issue-2, March-April 2017
International Journal of Engineering and Management Research
Page Number: 58-60

Key Aggregate Cryptosystem

Aiswarya R[1],Aiswarya Ramnadh[2], Aravind Krishnan M.P[3], Arunima M[4], Athira M.G[5], Kavitha S[6]
[1,2,3,4,5]B Tech Students,Department of Computer Science and Engineering, INDIA
[6]Assistant Professor,Department of Computer Science and EngineeringUniversity of Calicut, INDIA

ABSTRACT

Data sharing is an important functionality in cloud storage. In this article, we show how to securely, efficiently, and flexibly share data with others in cloud storage. We describe new public-key cryptosystems which produce constant-size ciphertexts such that efficient delegation of decryption rights for any set of ciphertexts 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 ciphertext 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*—**Aggregate key, Cipher texts, Public key, Cloud storage.

I. INTRODUCTION

In this project, we implement a technique to securely, efficiently, and flexibly share data with cloud storage. We introduce a new public-key cryptosystems which produce constant-size cipher texts such that efficient delegation 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. Data sharing is an important functionality in cloud storage. If the sender wants to send the portion of data to another person, naturally there are two extreme ways for the sender under thetraditional encryption paradigm.

- Sender encrypts all files with a single encryption key and gives the receiver the corresponding secret key directly.
- Sender encrypts files with distinct keys and sends the receiver the corresponding secret keys.

Obviously, the first method is inadequate since all un-chosen data may be also leaked to receiver. For the second method, there are practical concerns on efficiency. The purpose of this is to solve this problem 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. That means the cipher texts are further categorized into different classes. 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.

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 its employees access to a portion of data. The challenge is how to effectively do it. Users can download the encrypted data from storage and decrypt them but it the value of cloud storage is lost. Finding an efficient and secured way of sharing required partial data in cloud storage is a nontrivial and serious issue. This challenging issue is our motivation in this project.Goal of the project is to provide user to have flexible choices of cipher text set in cloud to be shared with another user keeping other encrypted files in cloud confidential. The new public key cryptosystems which produce constant size cipher texts such that the efficient delegation of decryption rights is possible. Secret aggregate key generated for this (decryption) will have the power of all keys aggregated. Storage space required to store keys is very limited. The data owner uses master key to generate this aggregate key which is kept secret. Also one of the major ideas is to get a constant size aggregate key no matter which one among the power set of classes of data is chosen.

II. EXISTING SYSTEM

There exist several expressive ABE schemes where the decryption algorithm only requires a constant number of pairing computations. Recently, Green *et al.* proposed a remedy to this problem by introducing the notion of ABE with outsourced decryption, which largely eliminates the decryption overhead for users. Based on the existing ABE schemes, Green *et al.* also presented concrete ABE schemes with outsourced decryption.

In these existing schemes, a user provides an untrusted server, say a proxy operated by a cloud service provider, with a transformation key TK that allows the latter to translate any ABE cipher text CT satisfied by that user’s attributes or access policy into a simple cipher text CT’, and it only incurs a small overhead for the user to recover the plaintext from the transformed cipher text CT’. The security property of the ABE scheme with outsourced decryption guarantees that an adversary (including the malicious cloud server) be not able to learn anything about the encrypted message; however, the scheme provides no guarantee on the correctness of the transformation done by the cloud server. In the cloud computing setting, cloud service providers may have strong financial incentives to return incorrect answers, if such answers require less work and are unlikely to be detected by users.

One of the main efficiency drawbacks of the most existing ABE schemes is that decryption is expensive for resource-limited devices due to pairing operations, and the number of pairing operations required to decrypt a cipher text grows with the complexity of the access policy.The above observation motivates us to study ABE with verifiable outsourced decryption in this thesis work. Here emphasized that an ABE scheme with secure outsourced decryption does not necessarily guarantee verifiability (i.e., correctness of the transformation done by the cloud server).

III. PROPOSED SYSTEM

We considered the verifiability of the cloud’s transformation and provided a method to check the correctness of the transformation. However, the we did not formally define verifiability. But it is not feasible to construct ABE schemes with verifiable outsourced decryption following the model defined in the existing. Moreover, the method proposed in existing relies on random oracles (RO). Unfortunately, the RO model is heuristic, and a proof of security in the RO model does not directly imply anything about the security of an ABE scheme in the real world. It is well known that there exist cryptographic schemes which are secure in the RO model but are inherently insecure when the RO is instantiated with any real hash function.

In this thesis work, firstly modify the original model of ABE with outsourced decryption in the existing to allow for verifiability of the transformations. After describing the formal definition of verifiability, we propose a new ABE model and based on this new model construct a concrete ABE scheme with verifiable outsourced decryption. Our scheme does not rely on random oracles.

In this paper we only focus on CP-ABE with verifiable outsourced decryption. The same approach applies to KP-ABE with verifiable outsourced decryption. To assess the performance of our ABE scheme with verifiable outsourced decryption, we implement the CP-ABE scheme with verifiable outsourced decryption and conduct experiments on both an ARM-based mobile device and an Intel-core personal computer to model a mobile user and a proxy, respectively.

The data privacy and security is maintained by designing a public key cryptosystem called as Key Aggregate Cryptosystem (KAC). This KAC helps user to share their data partially over cloud with constant size key pair of public-master keys and also receiver can decrypt this data with single constant size aggregate key.There are some limitation to the existing system like predefined bound of the number of maximum ciphertext classes and system is prompt to leakage of key.

IV. FRAME WORK

1. SETUP PHASE

The setup algorithm takes no input other than the implicit security parameter. It outputs the public parameters PK and a master key MK.

2.ENCRYPT PHASE

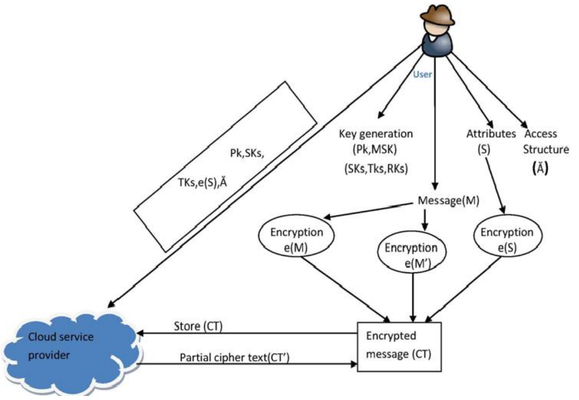
Encrypt(PK,M, A). The encryption algorithm takes as input the public parameters PK, a message M, and an access structure A over the universe of attributes. The algorithm will encrypt M and produce a ciphertext CT such that only a user that possesses a set of attributes that satisfies the access structure will be able to decrypt the message. We will assume that the ciphertext implicitly contains A.

3.KEY GEN PHASE

Key Generation(MK,S). The key generation algorithm takes as input the master key MK and a set of attributes S that describe the key. It outputs a private key SK

4.DECRYPT PHASE

Decrypt (PK, CT, SK). The decryption algorithm takes as input the public parameters PK, a cipher text CT, which contains an access policy A, and a private key SK, which is a private key for a set S of attributes. If the set S of attributes satisfies the access structure A then the algorithm will decrypt the cipher text and return a message M.



V. CONCLUSION

Our project makes it easier to share data flexibly over cloud. The cryptographic tool public key crypto system is implemented in this scenario. Outsourcing of data to a user may lead to leaking of private data which is preferred to be kept confidential if other techniques are used. Also the sharing of decryption keys in secure way is important. Public key cryptosystems provide delegation of secret keys for different cipher text classes in cloud. The delegate gets an aggregate key of constant size securely.

REFERENCES

[1]A Journal by Sanoop Jacob Thomas, Dr P. Balakumar , Department of Computer Science and Engineering.

[2] A paper by Qingling Wang, Carlos A. Varela, Department of Computer Science Rensselaer polytechnic Institute.

[3] IEEE paper on parallel and distributed systems by Cheng-Kong Chu, Sherman S.M. Chow, Wen Guey Tzeng, Robert H. Deng, Jianaying Chow

[4]ieeexplore.ieee.org.

[5] International Journal of Innovative Research in Computer and communication Engineering.