A policy to secure data and Self-destructing scheme in cloud.

Mayur Virkar[1], Madhukar Tarange[2], Amit Mane[3], Nilesh Pisal[4]

[1234] *Department of Information Technology,*
Dhole Patil College of Engineering, Pune, Maharashtra, India.

---***---

Abstract - *With the rapid development of versatile cloud service, it is becoming increasingly vulnerable to use cloud services to share data in a circle of friends in the cloud computing environment. Since it is not possible to implement the security of the entire privacy life cycle, access control becomes a daunting task, especially when we share sensitive data on servers in the cloud. To solve this problem, we suggest encryption based on key attributes of policy attributes specified time (KP-TSABE), a new system of self-disaster in cloud computing data. KP-TSABE in the schema, each ciphertext is marked with a time interval, while the private key is assigned with a time. Ciphertext can only be decrypted when the time is in the range allowed time, and attributes that are associated with the secret text meet the access structure of the key. The KP-TSABE is able to solve some important security problems to assist the authorization period by the user and provides access control precision in time. Confidential data safely self-destruct after a certain expiration by the user. Diffie-Hellman-Investment (l-Expanded BDHI) Course the KP-TSABE scheme was certainly under the decision l-bilinear. Complete comparisons of security features show that the KP-TSABE scheme proposed by us meets the security requirements and is better than other existing schemes*

Key Words: Sensitive data, secure self-destructing, fine-grained access control, privacy-preserving, cloud computing.

1. INTRODUCTION

Cloud computing is the next step in the evolution of on-demand information technology, which combines a number of existing and new technologies in search domains such as SOAs and virtualization. With the rapid development of multifaceted cloud technologies and services, it is a routine for users to use cloud storage services to share data with others in a group of friends like Dropbox, Google Drive and AL iCloud.

However, data shared in cloud servers usually contains sensitive information (such as personal profile, financial data, health records, etc.) and must be well protected. Because the property is separated from management data, the cloud server can migrate or users' information to other cloud servers from outsourcing in the cloud search. Therefore, it will be a great challenge to protect the privacy of shared data in the cloud, especially in the cloud environment and cloud data. To meet this challenge, It is necessary to create a complete solution to support the custom authorization period and provide fine grain access control during this period. Shared data must be destroyed even after the usual expiration date.

One way to mitigate problems is to store data as a common encrypted form. The disadvantage of encrypting data that the user cannot share at a fine-grained level, his encrypted data. When a data owner wants to convey any information that the owner needs to know exactly who he / she wants to share. In many applications, the data owner wants to share information with multiple users based on the security policy based on the user's credentials. Attribute-based encryption (ABE) has based on a traditional public key encryption rather than a significant one-to-one encryption benefits as it is a flexible encryption achieved [7]. ABE system provides a powerful way to get both data security.

The fine-grained access control policy in the ABE system key (KP-ABE) to be developed in this document, the cryptogram is marked by a number of descriptive attributes. Only when the amount of descriptive attributes to respond to the access structure to the key, the user can get the plain text.

2. LITERATURE SURVEY

A. A survey on Oruta: Privacy-preserving public auditing for shared data in the cloud:

In this paper, we propose Oruta, the first privacy preserving public auditing mechanism for shared data in the cloud. We utilize ring signatures to construct homomorphism authenticators, so the TPA is able to audit the integrity of shared data, yet cannot distinguish who is the signer on each block, which can achieve identity privacy. To improve the efficiency of verification for multiple auditing tasks, we further extend our mechanism to support batch auditing. An interesting problem in our future work is how to efficiently audit the integrity of shared data with dynamic groups while still preserving the identity of the signer on each block from the third party auditor.

B. A Survey on toward efficient and privacy-preserving computing in big data era:

Big data, because it can mine new knowledge for economic growth and technical innovation, has recently received considerable attention, and many research efforts have been directed to big data processing due to its high volume, velocity, and variety (referred to as "3V") challenges. However, in addition to the 3V challenges, the flourishing of big data also hinges on fully understanding and managing newly arising security and privacy challenges. If data are not authentic, new mined knowledge will be unconvincing; while if privacy is not well addressed, people may be reluctant to share their data. Because security has been investigated as a new dimension, "veracity," in big data, in this article, we aim to exploit new challenges of big data in terms of privacy, and devote our attention toward efficient and privacy-preserving computing in the big data era. Specifically, we first formalize the general architecture of big data analytics, identify the corresponding privacy requirements, and introduce an efficient and privacy-preserving cosine similarity computing protocol as an example in response to data mining's efficiency and privacy requirements in the big data era.

C. A Survey on Scalable, server-passive, user anonymous timed release cryptography:

In this paper, we provided a solution to the problem of server-passive and user-1anonymous timed release encryption. We proposed a construction that can achieve timed release encryption with a precisely specified absolute release time without needing any interaction between the server and the sender or the receiver. The scheme is scalable since only a single, identical time-bound key update for all users is needed for a certain time instant. It also has a number of useful additional properties like key insulation and simple public key renewal.

D. A Survey on Ciphertext-policy attribute-based encryption: An expressive, efficient, and provably secure realization:

In this section we describe how to realize a large universe construction in the standard model. In Appendix A the reduction applied the random oracle to \program" in the challenge ciphertext.the hash function. In the standard model we can achieve a similar type of programming by simply using a hash function that has enough degrees of randomness to plug in the same information. The trade is that the system must dene at setup, Attrmax, the maximum number of attributes any one key may have and the public paramters grow linearly with Attrmax. We emphasize that Attrmax does not limit the number of attributes that may be used in the system.

3. IMPLIMENTATION

Modules:

The system is proposed to have the following modules with functional requirements.

- Sensitive data,
- Secure self-development.
- Fine grain access control,
- Protection of privacy
- Cloud computing

Register

In this module, a new user records the information in the order of the list for the port of the client

Log in

In this module, the user can log in using his name and his key.

Sensitive data.

As the state of the art of Secure Self-Destruction Schema, SSDD and FullPP have limitations. Firstly, SSDD does not consider the problem of the desired release time of the sensitive data, the expiration time of the SSDD and FullPP schemes is limited by the DHT network and can not be determined by the user. Secondly, SSDD and many other systems depend on the ideal assumption of "no attack against VDO (data object disappearing) before it expires." Thirdly, it is demonstrated that the Vanish program is vulnerable to Sybil attacks from the DHT network, the SSDD scheme and other systems are similar. Accordingly, by indicating that the encrypted data item can not be decrypted between the data owner encrypts his or her data to share with the users of the system, wherein each user key is associated with an access tree and each Node is associated with an instant Instant, the access tree of each user can be defined.

Secure Self-Destructive Motorway:

A data self-destruction scheme, first proposed by Geambasu et al. , a promising approach that designs a Vanish system allows users to control the life cycle of sensitive data. Wang et al. Improved the Vanish system and proposed a secure self-destroying scheme for electronic data (SSDD). In the SSDD schema, data is encrypted in an auto-encryption and self-destruct scheme for data sharing in cloud computing. We first introduce the notion of KP-TSABE, formalize the KP-TSABE model and give it the security model. Next, we give a specific construction method about the schema. Finally, we prove that the KP-TSABE scheme is secure

Fine grain access control

In order to implement a fine-grained access control, we associate each attribute in the set of attributes with a time interval (authorization period). The attribute is valid if and only if the current time is within this time interval. Only if the valid attribute in the encrypted text satisfies the access tree in the key, the algorithm can decrypt the message correctly. The algorithm level of the KP-TSABE scheme includes four algorithms: Configuration, Encryption, KeyGen and Decryption.

Privacy Policy:

Due to the lack of time constraints, the ABE systems mentioned above do not support the user-defined authorization period and secure self-destruction after the privacy protection cycle expires. Life of data in cloud computing. As a result, it becomes a major challenge to protect the privacy of this shared data in the cloud, especially in cross-cloud and big data environments. In order to meet this challenge, it is necessary to design a complete solution to support the user-defined authorization period and provide fine grain access control during this period. Shared data should be self-degraded after the user-defined timeout

Cloud computing

Tysowski et al. Modified the ABE and the leveraged re-encryption algorithm to propose a new scheme to protect the data of mobile users in the cloud computing environment. Due to the lack of time constraints, the ABE systems mentioned above do not support the user-defined authorization period and secure self-destruction after the privacy protection cycle expires. Life of data in cloud computing. This is a time interval between the creation of shared data, the authorization period, and the expiration time. This article provides comprehensive privacy protection for the lifecycle of shared data in cloud computing.

Download:

The user wants all files to download here This module converts to your text file Ciper again Your process is finished.

4. ALGORITHMS USED

The KP-TSABE scheme can be described as a collection of the following four algorithms: Setup, Encrypt, KeyGen, and Decrypt.

Setup(1_, *U*): This algorithm is run by the Authority and takes as input the security parameter 1_ and attribute universe *U*, generates system public parameters *params* and the master key MSK. The Authority publishes *params* and keeps MSK secret to itself.

Encrypt(*M*, *params*, *S*, *TS*): Given the public parameters *params*, the shared message *M* which the owner wants to encrypt, the attribute set *S* and the set of time intervals *TS* in which every element in *TS* is associated with a corresponding attribute in *S*. This algorithm generates the ciphertext CT which is associated with the fuzzy attribute set *S*.

KeyGen(MSK,ϒ, *T′*): This algorithm takes as input the master key MSK, the access tree ϒ and the time set *T′*. Every attribute *x* in ϒ is associated with a time instant $tx \in T'$. It outputs a private key SK which contains ϒ.

Decrypt(CT, SK): This algorithm takes as input the ciphertext CT and the private key SK. When a set of time-specific attributes satisfies ϒ, it is able to decrypt the ciphertext and return the plaintext *M*.

5. ARCHITECTURAL DIAGRAM:

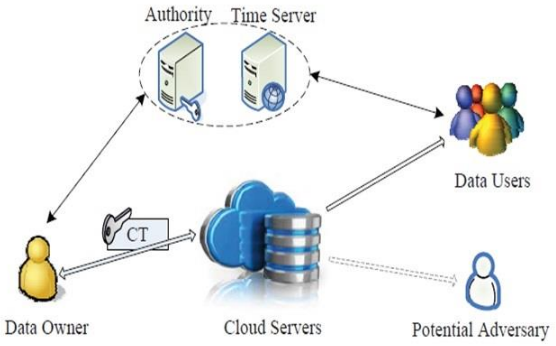

Fig-1. Architecture

6. ADVANTAGES

- ✓ Attribute based encryption (ABE) has significant advantages based on the tradition public key encryption instead of one-to-one encryption because it achieves flexible advantages
- ✓ with regard to security and fine-grained access control compared to other secure self-destructing schemes.

- ✓ supporting user-defined time-specific authorization, fine-grained access control and data secure self-destruction.

7. CONCLUSION

With the rapid development of multi-service cloud, many new challenges have emerged. One of the biggest problems is how to remove securely stored data stored in the cloud selector. In this paper, we have proposed a new KP-TSABE scheme that allows for the encrypted text time to solve these problems by allowing flexible access control and fine quality throughout the duration of the authorization to introduce the controllable self-destruction in time the process to split data and be outsourced in the cloud. We have also given a system model and a security model for KPTSABE scheme. In addition, we have proven that KPTSABE is safe in the standard model with the decision Expanded BDHI assumption. The comprehensive analysis showed that the KP-TSABE proposed scheme is superior to other existing systems.

REFERENCES

[1] B. Wang, B. Li, and H. Li, “Oruta: Privacy-preserving public auditingfor shared data in the cloud,” Cloud Computing, IEEE Transactions on, vol. 2, no. 1, pp. 43–56, 2014.

[2] J. Xiong, Z. Yao, J. Ma, X. Liu, Q. Li, and J. Ma, “Priam: Privacy preserving identity and access management scheme in cloud,” KSII Transactions on Internet and Information Systems (TIIS), vol. 8, no. 1, pp. 282–304, 2014.

[3] J. Xiong, F. Li, J. Ma, X. Liu, Z. Yao, and P. S. Chen, “A full lifecycle privacy rotection

scheme for sensitive data in cloud computing,” Peerto- Peer Networking and Applications. [Online]. Available: http://dx.doi.org/10.1007/s12083-014-0295-x

[4] P. Jamshidi, A. Ahmad, and C. Pahl, “Cloud migration research: A systematic review,” Cloud Computing, IEEE Transactions on, vol. 1, no. 2, pp. 142–157, 2013.

[5] R. Lu, H. Zhu, X. Liu, J. K. Liu, and J. Shao, “Toward efficient and privacy-preserving computing in big data era,” Network, IEEE, vol. 28, no. 4, pp. 46–50, 2014.

[6] X. Liu, J. Ma, J. Xiong, and G. Liu, “Ciphertext-policy hierarchical attribute-based encryption for fine-grained access control of encryption data,” International Journal of Network Security, vol. 16, no. 4, pp. 351–357, 2014.

[7] A. Sahai and B. Waters, “Fuzzy identity-based encryption,” in Advances in Cryptology–EUROCRYPT 2005, ser. LNCS, vol. 7371. Springer, 2005, pp. 457–473.

[8] V. Goyal, O. Pandey, A. Sahai, and B. Waters, “Attribute-based encryption for fine-grained access control of encrypted data,” in Proceedings of the 13th ACM conference on Computer and Communications Security. ACM, 2006, pp. 89–98.

[9] A. F. Chan and I. F. Blake, “Scalable, server-passive, useranonymous timed release cryptography,” in Proceedings of the International Conference on Distributed Computing Systems. IEEE, 2005, pp. 504–513.

[10] K. G. Paterson and E. A. Quaglia, “Time-specific encryption,” in Security and Cryptography for Networks. Springer, 2010, pp. 1–16.

[11] Q. Li, J. Ma, R. Li, J. Xiong, and X. Liu, “Large universe decentralized key-policy attribute-based encryption,” Security and Communication Networks, 2014. [Online]. Available: http://dx.doi.org/10.1002/sec.997