International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2013): 6.14 | Impact Factor (2013): 4.438

Privacy Preserving Data Sharing With CP-ABE

Neeta S. Nipane[1], Nutan M. Dhande[2]

[1, 2]Agnihotri College of Engineering, RTMNU University, Wardha, Maharashtra, India

Abstract: ***With the recent adoption and diffusion of the data sharing paradigm in distributed systems such as online social networks or cloud computing, there have been increasing demands and concerns for distributed data security. One of the most challenging issues in data sharing systems is the enforcement of access policies and the support of policies updates. With the development of cryptography, the attribute-based encryption (ABE) draws widespread attention of the researchers in recent years. The ABE scheme, which belongs to the public key encryption mechanism, takes attributes as public key and associates them with the ciphertext or the user's secret key. It is an efficient way to solve open problems in access control scenarios, for example, how to provide data confidentiality and expressive access control at the same time. Ciphertext policy attribute-based encryption (CP-ABE) is becoming a promising cryptographic solution to this issue. It enables data owners to define their own access policies over user attributes and enforce the policies on the data to be distributed. Therefore, in this study, we propose a novel CP-ABE scheme for a data sharing system by exploiting the characteristic of the system architecture. The proposed scheme features the following achievements: 1) the key escrow problem could be solved by escrow-free key issuing protocol, which is constructed using the secure two-party computation between the key generation center and the data-storing center, and 2) fine-grained user revocation per each attribute could be done by proxy encryption which takes advantage of the selective attribute group key distribution on top of the ABE. The performance and security analyses indicate that the proposed scheme is efficient to securely manage the data distributed in the data sharing system. Module.***

Keywords: Data sharing, attribute-based encryption, revocation, access control, removing key escrow, ABE, CP-ABE

1. Introduction

Recent development of the network and computing technology enables many people to easily share their data with others using online external storages. With the development of the Internet and the distributed computing technology, there is a growing demand for data sharing and processing in an open distributed computing environment. Cloud computing is an alternative to information technology because of its resource-sharing and low-maintenance characteristics. With the recent adoption and diffusion of the data sharing paradigm in distributed systems such as online social networks or cloud computing, there have been increasing demands and concerns for distributed data security. As people enjoy the advantages of these new technologies and services, their concerns about data security and access control also arise. People would like to make their sensitive or private data only accessible to the authorized people with credentials they specified. There are various other issues such as risks of privacy exposure, scalability in key management, flexible access and efficient user revocation. To achieve fine grained and scalable data access control for any records stored in semi trusted servers, leverage attribute based encryption (ABE) techniques is a promising cryptographic approach to encrypt record file. ABE is envisioned as an important tool for addressing the problem of secure and fine-grained data sharing and access control. In an ABE system, a user is identified by a set of attributes.

2. Literature Review

In 2009, M. Chase and S.S.M. Chow presented a distributed KP-ABE scheme that solves the key escrow problem in a multi-authority system. In this approach, all (disjoint) attribute authorities are participating in the key generation protocol in a distributed way such that they cannot pool their data and link multiple attribute sets belonging to the same user. One disadvantage of this kind of fully distributed approach is the performance degradation. Since there is no centralized authority with master secret information, all attribute authorities should communicate with the other authorities in the system to generate a user's secret key. This results is communication overhead on the system setup phase and on any rekeying phase, and requires each user to store additional auxiliary key components besides the attributes keys, where N is the number of authorities in the system [2].

In 2009, Recently, S.S.M. Chow proposed an anonymous private key generation protocol in identity-based literature such that the KGC can issue a private key to an authenticated user without knowing the list of users' identities. It seems that this anonymous private key generation protocol works properly in ABE systems when we treat an attribute as an identity in this construction. However, found that this cannot be adapted to ABE systems due to mainly two reasons. First, in Chow's protocol, identities of users are not public anymore, at least to the KGC, because the KGC can generate users' secret keys otherwise. Second, since the collusion attack between users is the main security threat in ABE [3].

In 2008, Bethencourt, V. Kumar and Boldyreva proposed first key revocation mechanisms in CP-ABE and KP-ABE settings, respectively. These schemes enable an attribute key revocation by encrypting the message to the attribute set with its validation time. These attribute-revocable ABE schemes have the security degradation problem in terms of the backward and forward secrecy. They revoke attribute itself using timed rekeying mechanism, which is realized by setting expiration time on each attribute. In ABE systems, it is a considerable scenario that membership may change frequently in the attribute group. Then, a new user might be able to access the previous data encrypted before his joining until the data are reencrypted with the newly updated attribute keys by periodic rekeying (backward secrecy). On the other hand, a revoked user would still be able to access

the encrypted data even if he does not hold the attribute any more until the next expiration time (forward secrecy). Such an uncontrolled period is called the window of vulnerability [4] [13].

In 2007, R. Ostrovsky, A. Sahai, and B. Waters proposed the user revocation can be done by using ABE that supports negative clauses. To do so, one just adds conjunctively the AND of negation of revoked user identities. One drawback in this scheme is that the private key size increases by a multiplicative factor of log n, where n is the maximum number of attributes [5].

In 2010, Lewko, A. Sahai, and B. Waters proposed more efficient instantiations of framework for non-monotonic ABE, where public parameters is only group elements O(1), and private keys for access structures involving t leaf attributes is of size O(t). However, these user-revocable schemes also have a limitation with regard to the availability. When a user is revoked even from a single attribute group, he loses all the access rights to the system, which is not desirable in many pragmatic scenarios since the other attributes may be still valid [6].

In 2009, N. Attrapadung and H.Imai suggested another user-revocable ABE schemes addressing this problem by combining broadcast encryption schemes with ABE schemes. However, in this scheme, the data owner should take full charge of maintaining all the membership lists for each attribute group to enable the direct user revocation. This scheme is not applicable to the data sharing system, because the data owners will no longer be directly in control of data after storing their data to the external storage server [7].

In 2010, Recently S. Yu, C. Wang, K. Ren, and W. Lou addressed the user revocation in the ABE-based data sharing system. In this scheme, the user revocation is realized using proxy reencryption by the data server. However, in order to revoke users, the KGC should generate all secret keys including the proxy key on behalf of the data server. Then, the server would reencrypt the ciphertext under the proxy key received from the KGC to prevent revoked users from decrypting the ciphertext. Thus, the key escrow problem is also inherent in this scheme, since the KGC manages all secret keys of users as well as the proxy keys of the data server [8].

In 2010, X. Liang, R. Lu, X. Lin, and X. Shen Liang proposed a CP-ABE scheme with efficient revocation. Their construction uses linear secret sharing and binary tree techniques, and can be proved secure in the standard model. In addition to the attribute set, each user is also assigned a unique identifier. Therefore, a user can be easily revoked by using his/her unique identifier. Above scheme support user revocation, but they have no effect on attribute revocation [9].

In 2009, L. Ibraimi, M. Petkovic, S. Nikova, P. Hartel, and W. Jonker , proposed to reduce the burden of authority and achieve immediate attribute revocation, two CP ABE schemes with immediate attribute revocation with the help of semi honest service. However, they also have failed to achieve fine-grained user access control in the data outsourcing environment [10].

In 2011, J. Hur and D. K. Noh proposed a CP-ABE scheme with fine-grained attribute revocation with the help of the honest-but-curious proxy deployed in the data service provider. It is an efficient revocation method by employing the binary tree representing and re-encrypting the cipher-text. However, their scheme cannot resist the collusion attack. Aiming at reducing the computation overhead of data service manager [11].

In 2011, Lekwo and Waters proposed a new multiauthority scheme. Although their scheme may become inefficient for large attribute universe, it is the first adaptively secure multiauthority CP-ABE scheme proved in the random oracle model. This scheme improves the previous multiauthority ABE schemes, because it does not require collaboration among multiple authorities in the setup and key generation phases, and there is no central authority. Note that the authority in this scheme can join or leave the system freely without reinitializing the system. Besides the low efficiency, this scheme has another drawback that the attributes of the user can be collected by tracing his GID [12].

3. Problem Statement

Storing data on un-trusted storage makes secure data sharing a challenge issue. Fine-grained data access control mechanisms often need to be in place to assure appropriate disclosure of sensitive data among multiple 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 ciphertext grows with the complexity of the access policy and ,decryption keys only support user attributes that are organized logically as a single set, so the users can only use all possible combinations of attributes in a single set issued in their keys to satisfy policies.

4. Security Analysis Of The Proposed System

According to the existing schemes, the functionalities in an ideal ABE scheme is listed as follows:

- Data confidentiality: Unauthorized participants cannot know the information about the encrypted data.
- Fine-grained data access control: In order to achieve flexible access control, even for users in the same group, their access rights are not the same.
- User/attribute revocation: If a user quits the system, the scheme can revoke his access right. Similarly, attribute revocation is inevitable.
- Collusion resistance: The dishonest users cannot combine their attributes to decrypt the encrypted data.
- Scalability: The number of authorized users cannot affect the performance of the scheme. That is to say, the scheme can deal with the case that the number of the authorized users increases dynamically.

5. System Architecture

Cloud computing, is an emerging computing paradigm, enabling users to remotely store their data in a server and provide services on-demand. In cloud computing cloud users and cloud service providers are almost certain to be from different trust domains. Data security and privacy are the critical issues for remote data storage. A secure user enforced data access control mechanism must be provided before cloud users have the liberty to outsource sensitive data to the cloud for storage.

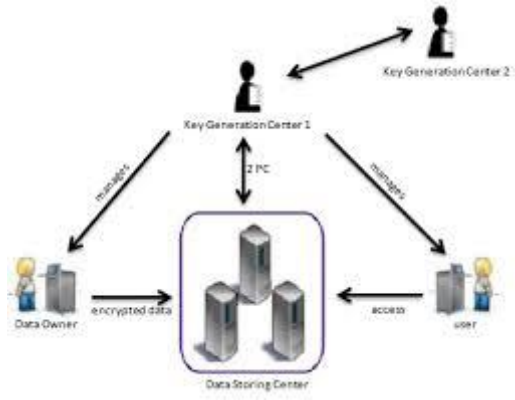

Figure 1: Architecture of Data Sharing

The architecture of data sharing system, which consist of the following entities [1].

1) Key generation center: It is a key authority that generates public and secret parameters for CP-ABE. It is in charge of issuing, revoking, and updating attribute keys for users. It grants differential access rights to individual users based on their attributes. That is, it will honestly execute the assigned tasks in the system; however, it would like to learn information of encrypted contents as much as possible. Thus, it should be prevented from accessing the plaintext of the encrypted data even if it is honest.

2) Data-storing center: It is an entity that provides a data sharing service. It is in charge of controlling the accesses from outside users to the storing data and providing corresponding contents services. The data-storing center is another key authority that generates personalized user key with the KGC, and issues and revokes attribute group keys to valid users per each attribute, which are used to enforce a fine-grained user access control.

3) **Data owner:** It is a client who owns data, and wishes to upload it into the external data-storing center for ease of sharing or for cost saving. A data owner is responsible for defining (attribute-based) access policy, and enforcing it on its own data by encrypting the data under the policy before distributing it.

4)User: It is an entity who wants to access the data. If a user possesses a set of attributes satisfying the access policy of the encrypted data, and is not revoked in any of the valid attribute groups, then he will be able to decrypt the ciphertext and obtain the data.

6. Algorithm

1)Key Generation: When the KGC authenticates a user u_t, it selects a random exponent $r_t \in R$ for the user. This value is a personalized and unique secret to the user. Then, the KGC and the data storing center engage in a secure 2PC protocol, where the KGC's private input is (r_t, β), and the data-storing center's private input is α. The secure 2PC protocol returns a private output $X = (\alpha + r_t)\ \beta$ to the data-storing center.

2)Encryption: When a data owner wants to upload its data M to the data-storing center for sharing using RSA algorithm
Generate randomly two "large" primes p and q.
Compute $n = pq$ and $\phi = (p - 1)(q - 1)$.
Randomly public key is generated is
$K_E = (n, e)$.

3)Data Reencryption : Before distributing the ciphertext, the data-storing center reencrypts it by running ReEncrypt(CT, G) using a set of the membership information for each attribute group G that appears in the access tree of CT. The reencryption algorithm enforces user access control per each attribute group.

Given nodes $n_1,\ldots n_N$ each holding an data item d_i
$r_{i,1}+\ldots\ldots+r_{i,N} = d_i$
Now each node n_i simply broadcasts si to all other nodes so that each node can compute:
$T = s_1+\ldots..+s_N$

4)Decryption: Data decryption phase consists of the attribute group key decryption using one-way anonymous key agreement protocol, followed by the message decryption from CT'.

7. Result Analysis

The performance of the project is evaluated with the existing system and suitable graphs are constructed. Parameters like Encryption Time, Decryption time are compared. The results are indicating modified algorithm is performed better than the existing CP-ABE in terms of security and time of encryption and decryption.

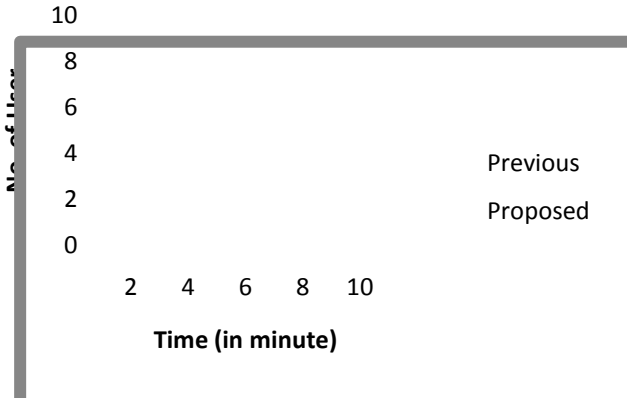

Figure 2: Comparison between previous and proposed system in terms of time and No. of user in data sharing

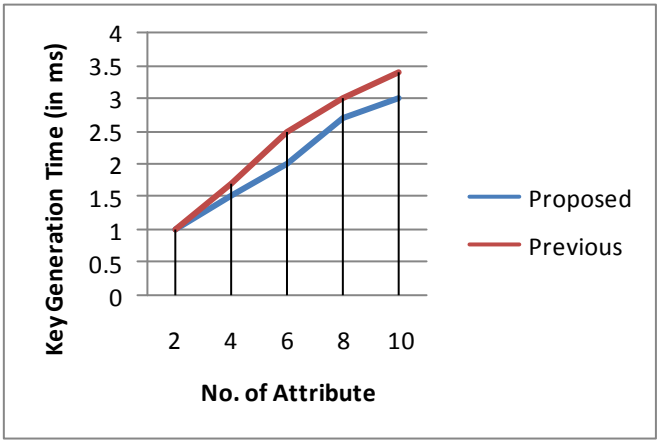

Figure 3: Comparison between previous and proposed system for key generation time with respective No. of attribute

8. Conclusion

Issues such as scalability in key management, flexible access and efficient user revocation, security have remained the most important challenges toward achieving fine-grained, cryptographically enforced data access control. For improving the security and fine grained access control of stored data on un-trusted server the anonymous id generated for attribute based encryption is introduced. It achieved higher security than the existing system. This scheme also enables dynamic modification of access policies or file attributes. Finally it generates a performance evaluation graph it take less time for encryption compared to the existing attribute based encryption.

9. Future Work

Future work is Alert and Notification System. The system needs an alert and notification system to give both providers and users the ability to track their data. Alerts can help users to prevent any miss-use of their data. Also the alert system can be used to identify suspicious activities.

References

[1] Junbeom Hur, "Improving Security and Efficiency in Attribute-Based Data Sharing" IEEE Transactions On Knowledge And Data Engineering Vol:25 No:10 Year 2013
[2] M. Chase and S.S.M. Chow, "Improving Privacy and Security in Multi-Authority Attribute-Based Encryption," Proc. ACM Conf. Computer and Comm. Security, pp. 121-130, 2009.
[3] S.S.M. Chow, "Removing Escrow from Identity-Based Encryption," Proc. Int'l Conf. Practice and Theory in Public Key Cryptography (PKC '09), pp. 256-276, 2009.
[4] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-Policy Attribute-Based Encryption," Proc. IEEE Symp. Security and Privacy, pp. 321-334, 2007.
[5] R. Ostrovsky, A. Sahai, and B. Waters, "Attribute-Based Encryption with Non-Monotonic Access Structures," Proc. ACM Conf. Computer and Comm. Security, pp. 195-203, 2007.
[6] A. Lewko, A. Sahai, and B. Waters, "Revocation Systems with Very Small Private Keys," Proc. IEEE Symp. Security and Privacy, pp. 273-285, 2010.
[7] N. Attrapadung and H. Imai, "Conjunctive Broadcast and Attribute-Based Encryption," Proc. Int'l Conf. Palo Alto on Pairing-Based Cryptography (Pairing), pp. 248-265, 2009.
[8] S. Yu, C. Wang, K. Ren, and W. Lou, "Attribute Based Data Sharing with Attribute Revocation," Proc. ACM Symp. Information, Computer and Comm. Security (ASIACCS '10), 2010.
[9] X. Liang, Z. Cao, H. Lin, and D. Xing, "Provably secure and efficient bounded ciphertext policy attribute based encryption," in Proceedings of the 4th International Symposium on ACM Symposium on Information, Computer and Communications Security (ASIACCS '09), pp. 343–352, March 2009.
[10] L. Ibraimi, Q. Tang, P. Hartel, and W. Jonker, "Efficient and provable secure ciphertext-policy attribute-based encryption schemes," in Information Security Practice and Experience (ISPE 2009), pp. 1–12, Springer, Berlin, Germany, 2009.
[11] J. Hur and D. K. Noh, "Attribute-based access control with efficient revocation in data outsourcing systems," IEEE Transactions on Parallel and Distributed Systems, vol. 22, no. 7, pp. 1214–1221, 2011.
[12] A. Lewko, T. Okamoto, A. Sahai, and B. Waters, "Fully secure functional encryption: attribute-based encryption and (hierarchical) inner product encryption," in Advances in Cryptology: EUROCRYPT 2010, vol. 6110 of Lecture Notes in Computer Science, pp. 62–91, Springer, Berlin, Germany, 2010.
[13] A. Boldyreva, V. Goyal, and V. Kumar, "Identity-Based Encryption with Efficient Revocation," Proc. ACM Conf. Computer and Comm. Security, pp. 417-426, 2008.

Author Profile

Neeta S. Nipane, pursuing M.Tech 2nd year, Computer Science & Engineering in Nagpur Universty. Two years of experience as an Assistant Lecturer in an engineering college.