ISSN 2348-1196 (print)
International Journal of Computer Science and Information Technology Research ISSN 2348-120X (online)
Vol. 3, Issue 1, pp: (241-248), Month: January - March 2015, Available at: www.researchpublish.com

Enlightening Security and Proficiency in Distributed Data Accessing For Cloud Computing

[1]Mrs. P. SIVAGAMI, [2]Ms. A. SANDHIYA

[1,2]Department of Computer Science, D.K.M College for Women, Vellore, Tamil Nadu, India

Abstract: **In recent days, data sharing using distributed systems is mainly carried out in 2 ways such as social networks and cloud computing. So, necessity of security is highly essential. Challenging issues are accessing and updating the policies of data sharing. In that one such policies is cipher text Policy Attribute Based Encryption (CP_ABE).for example primary health care center scenario for a patient attribute. The major drawback is key escrow problem. Advantage is to data owner can access easily with the patient details. In key generation center decryption carried out by private keys. In data sharing scenarios, attribute based methods are not highly suitable, since we can share only to the designated users. Cipher text policy introduces another challenge is a revocation to the user. The proposed scheme features the following achievements: 1) the Keyes row 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 the proposed scheme is efficient to securely manage the data distribute in the data sharing system.**

Keywords: **Attributed based Encryption, Cipher Text Policy, Data Sharing, and revocation.**

1. INTRODUCTION

Recent development of the network and computing technology enables many people to easily share their data with others uses online external storages. People can share their lives with friends by uploading their private photos or messages into the online social networks such as Face book and MySpace; or upload highly sensitive personal health records (PHRs) into online data servers such as Microsoft Health Vault, Google Health for ease of sharing with their primary doctors or for cost saving. As people enjoy the advantages of these new technologies and services, their concerns about data security and access control also arise. Improper use of the data by the storage server or unauthorized access by outside users could be potential threats to their data. People would like to make their sensitive or private data only accessible to the authorized people with credentials they specified.

Security is a most important thing in the data sharing. In the data sharing the main problem is leakage of data. The data can be protected by encrypting it with proper security key. In this system we have develop the data sharing using Attribute Based Encryption (ABE) Algorithm. By this our data becomes more secure than the existing system. Nevertheless, applying CP-ABE in the data sharing system has several challenges. In CP-ABE, the key generation center (KGC) generates private keys of users by applying the KGC's master secret keys to users' associated set of attributes. Thus, the major benefit of this approach is to largely reduce the need for processing and storing public key certificates under traditional public key infrastructure (PKI). However, the advantage of the CP-ABE comes with a major drawback which is known as a key escrow problem. The KGC can decrypt every cipher text addressed to specific users by generating their attribute keys.

1.1 Motivation:

Identity (ID)-based encryption, or IBE for short, is an exciting alternative to public-key encryption, which eliminates the need for a Public Key Infrastructure (PKI) that makes publicly available the mapping between identities, public keys, and validity of the latter. The senders using an IBE do not need to look up the public keys and the corresponding certificates of the receivers, because the identities (e.g. emails or IP addresses) together with common public parameters are sufficient for encryption. The private keys of the and Shamir [2],

1.2 Related Work:

ABE comes in two flavors called key-policy ABE (KP-ABE) and cipher text-policy ABE. In KP-ABE, attributes are used to describe the encrypted data and policies are built into users' keys; while in CP-ABE, the attributes are used to describe users' credentials, and an encryptor determines a policy on who can decrypt the data. Between the two approaches, CP-ABE is more appropriate to the data sharing system because it puts the access policy decisions in the hands of the data owners [2]Propose a solution to optimally distribute the traffic along multiple multicast trees. However, the solution covers the case when there is only one active source in the network. In addition, it is assumed that the gradient of an analytical cost function is available, which is continuously differentiable and strictly convex. These assumptions may not be reasonable due to the dynamic nature of networks.

Revocation has been studied in the ID-based setting with mediators [5, 8]. In this setting there is a special semi-trusted third party called a mediator who holds shares of all users' private keys and helps users to decrypt each ciphertext. If an identity is revoked then the mediator is instructed. to stop helping the user. But we want to focus on a much more practical standard IBE setting where users are able to decrypt on their own.

The goal of broadcast encryption is to prevent revoked users from accessing secret information being broadcast. The broadcast encryption solutions, however, and in particular ID-based broadcast encryption ones, do not directly translate into solutions for our problem. In broadcast encryption, a non-revoked user can help a revoked user gain access to the sensitive information being broadcast (since this information is the same for all parties). On the other hand, in the IBE setting a revoked user, or the adversary holding its private key, should not be able to decrypt messages even if it colludes with any number of non-revoked users.

1.3 A Comparison between current attribute-based Access control schemes and ours:

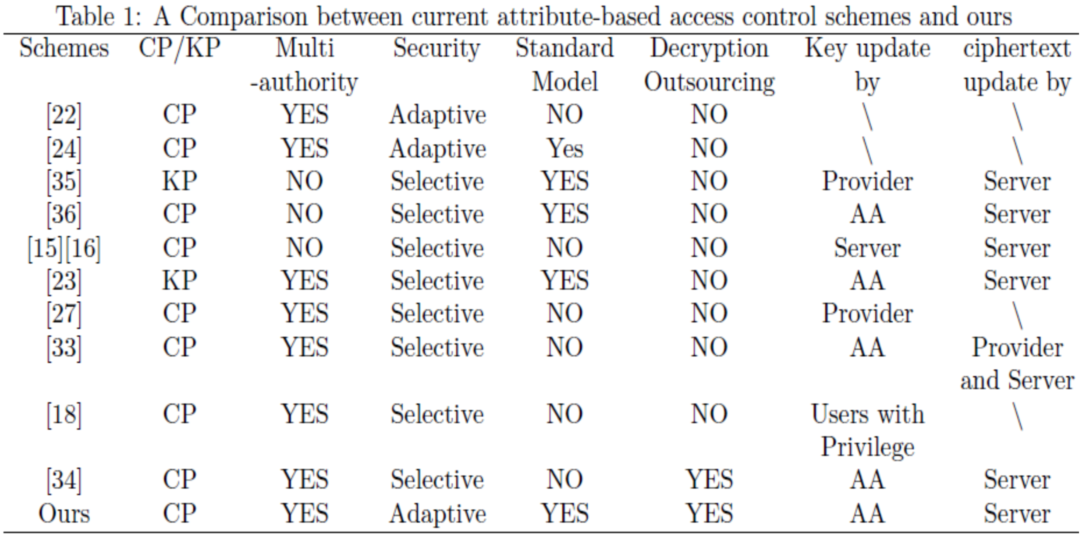

Table 1: A Comparison between current attribute-based access control schemes and ours

Schemes	CP/KP	Multi -authority	Security	Standard Model	Decryption Outsourcing	Key update by	ciphertext update by
[22]	CP	YES	Adaptive	NO	NO	\	\
[24]	CP	YES	Adaptive	Yes	NO	\	\
[35]	KP	NO	Selective	YES	NO	Provider	Server
[36]	CP	NO	Selective	YES	NO	AA	Server
[15][16]	CP	NO	Selective	NO	NO	Server	Server
[23]	KP	YES	Selective	YES	NO	AA	Server
[27]	CP	YES	Selective	NO	NO	Provider	\
[33]	CP	YES	Selective	NO	NO	AA	Provider and Server
[18]	CP	YES	Selective	NO	NO	Users with Privilege	\
[34]	CP	YES	Selective	NO	YES	AA	Server
Ours	CP	YES	Adaptive	YES	YES	AA	Server

Fig1: Attribute based Access Control

1.4 DETAILED PROBLEM DEFINITION:

Security is a most important thing in the data sharing. In the data sharing the main problem is leakage of data. The data can be protected by encrypting it with proper security key. In this system we have develop the data sharing using Attribute Based Encryption (ABE) Algorithm. By this our data becomes more secure than the existing system.

The scope of this project is to protect the data from other persons in the network by encrypting it and send it in the social networks. The authorized person who was received the message will send the key request to the data owner. After receiving the key from proposed a DAC-MACS system by employing the decryption outsourcing technique [13]. The heavy bilinear pairing operations are outsourced to the clouds. However, their scheme caused heavy computation of the AAs in revocation and was proved selectively secure in the random oracle model. Table 1 describes some characteristics of current attribute-based access control schemes in the clouds and ours.

If multiple users collude, they may be able to decrypt a ciphertext by combining their attributes even if each of the users cannot decrypt the ciphertext alone. We do not want these colluders to be able to decrypt the private data in the server by combining their attributes.Since we assume the KGC and data-storing center are honest, we do not consider any active attacks from them by colluding with revoked users.

2. SYSTEM MODEL, THREAT MODEL AND SECURITY REQUIREMENT

In this section, we introduce the system model, threats model and security requirements of our multi-authority access control scheme for cloud storage.

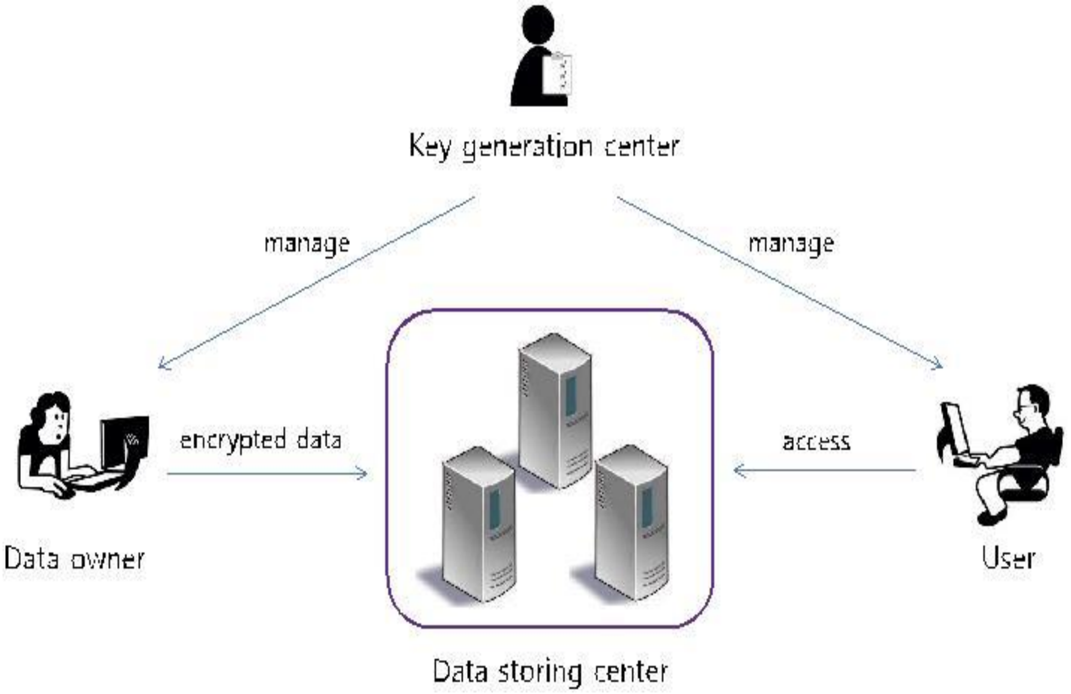

Fig2: System Architecture

2.1 System model:

1. Central Authority (CA): The CA sets up its public parameters. It is in charge of issuing an *gid*-related key to the user. It will not participate in any attribute-related operations.

2. Attribute Authorities (AAs): Each AA is responsible to administer a distinct attribute domain, which is a subset of the system attribute universe. In our scheme, every attribute is managed by a single AA, but each AA can govern an arbitrary scale of attribute domain. While receiving the private key request from a user, it responds the attribute-related keys. Additionally, once one or more attributes are revoked from one or more users, it also executes the key updating process for unrevoked users.

3. Server: It is an entity which provides data storage service and decryption outsourcing service. Moreover, it also gives service to cipher text re-encryption.

4. Data Providers: Before transmitting the data file to the cloud server, a data provider has to encrypt it under a DEK (Data Encryption Key).

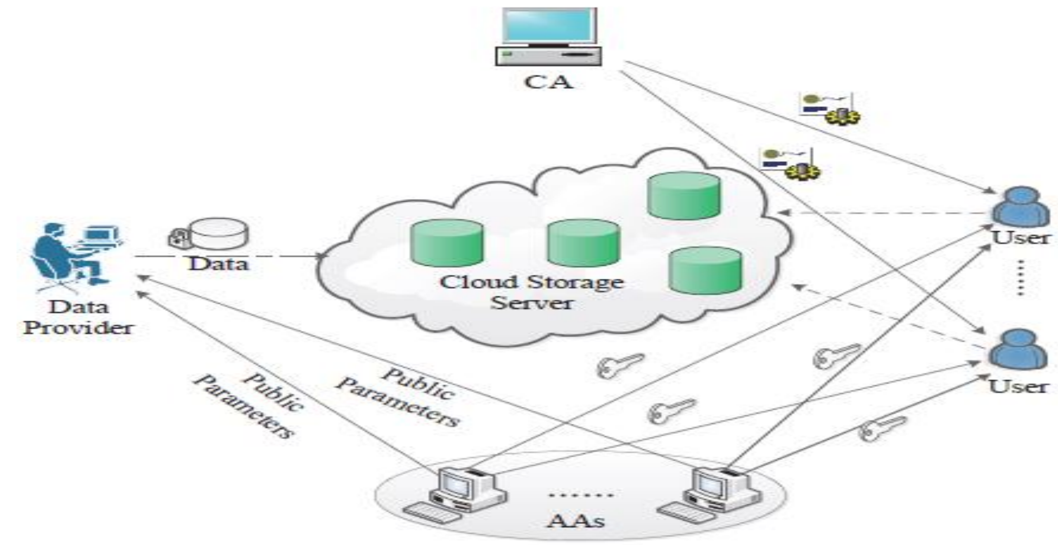

Fig 3: Storage Server

5. Users: Each user with a *gid* is labeled by a set of attributes. He has to request the attribute related keys from corresponding AAs. A user can download the encrypted data and call for decryption outsourcing service from the cloud server. But only the user who owns proper attributes can successfully decrypt the encrypted data.

2.2 Threats model:

In this work, the CA is the only one which can be fully trusted. The AAs honestly distribute the keys and send the key updating message, but some of them may be corrupted by the adversary which attempts to find out information of the data file as much as possible. We assume that the AAs will never collude with any user.

As similar as the assumption in [10], the cloud server is assumed to be honest but curious. That is, the cloud server will follow the presented protocol in general, but may collude with malicious users or data providers to get illegal access privileges. However, it will not collude with the revoked users. We assume that the cloud server mostly focuses on information of data contents.

We assume that the users are malicious all the time. They may collude with the others and even the cloud server, and try to access the data that they are not authorized. The authorized person who was received the message will send the key request to the data owner. After receiving the key from the sender only the message gets decrypted.

3. PRELIMINARIES AND DEFINITION

3.1 Access Structure:

Notations:

In this paper, x 2R S denotes the operation of picking an element x at random and uniformly from a finite set S. For a probabilistic algorithm A; x $ A assigns the output of A to the variable x. 1_ denotes a string of _ ones, if _ 2 IN. A function _ : IN ! IR is negligible (negl(k)) if for every constant c _ 0 there exists kc such that _ðkÞ < k_c for all k > kc.

Definition 1. Access Structure [2]: Let P = {P1, P2, . . . , PT } denote a set of parties. A collection A ⊆ 2{P1,P2,...,PT } is monotonic if ∀A1,A2: if A1 ∈ A and A1 ⊆ A2 then we have A2 ∈ A. An access structure (respectively, monotone access structure) is a collection (respectively, monotone collection) A of non-empty subsets of P. That is, A ⊆ 2{P1,P2,...,PT }\{∅ }. We say that the sets in A are the authorized sets, and the sets outside A are the unauthorized sets.

Among ABE systems, the role of the parties is replaced by the descriptive attributes. In this way, the authorized set of attributes will be contained in the access structure A. We focus on the monotonic access structure in this paper. To realize common access structures, one can simply consider the negation of an attribute as a separate attribute,

3.2 One-Way Anonymous Key Agreement:

In a Boneh-Franklin identity-based encryption setup [11], a trusted key authority called private key generator (PKG) generates private keys di for users with identities IDi using a master secret s. A user with identity IDi receives the private key di ¼ HðIDiÞs 2 GG0, where H : f0; 1g_ ! GG0 is a cryptographic hash function. On the basis of this setup, Kate et al. [16] proposed an oneway anonymous key agreement scheme by replacing the identity hashes with pseudonyms generated by users. One-way anonymous key agreement is to guarantee anonymity for just one of the participants; the other participant works as a nonanonymous service provider and the anonymous participant needs to confirm the service provider's identity. In this setting, two participants can agree on a session key in a no interactive manner.

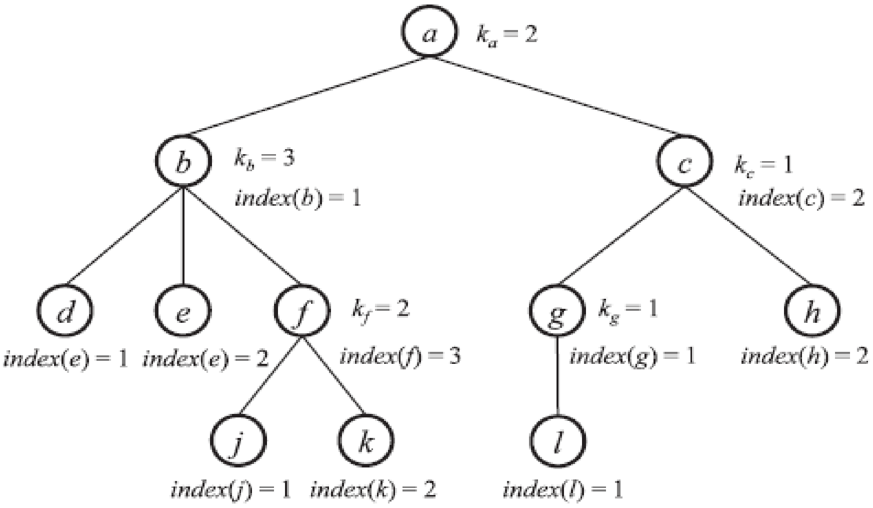

Fig 4: KGC

On receipt of the request, the KGC notifies the data storing center of the event and sends the updated membership list of the attribute group to it. When the data-storing center receives the notification, it rekeys the corresponding attribute group key.

4. ALGORITHM

Key Generation and Exchange

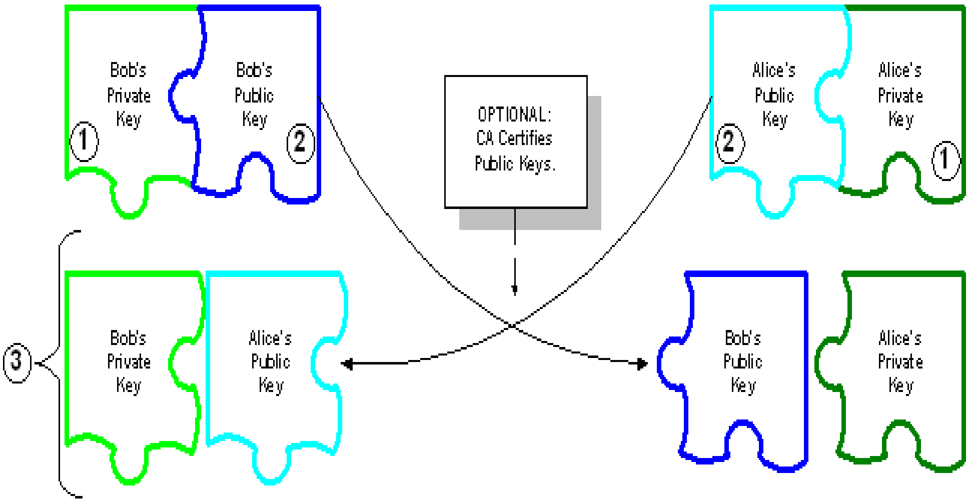

Fig 5: D-H Key Generation and Exchange

- Alice and Bob agree to use a prime number p=23 and base g=5.
- Alice chooses a secret integer a=6, then sends Bob (ga mod p) 156 mod 23 = 8.

- Bob chooses a secret integer b=15, then sends Alice (gb mod p) 515 mod 23 = 19.
- Alice computes (gb mod p)a mod p• 196 mod 23 = 2.
- Bob computes (ga mod p)b mod p 815 mod 23 = 2.
- Both Alice and Bob have arrived at same value, because gab and gba are equal. Note that only a, b and gab = gba are kept secret.

All the other values are sent in the clear. Once Alice and Bob compute the shared secret they can use it as an encryption key, known only to them, for sending messages across the same open communications channel. Of course, much larger values of a, b, and p would be needed to make this example secure, since it is easy to try all the possible values of gab mod 23 (there will be, at most, 22 such values, even if a and b are large). If p were a prime of at least 300 digits, and a and b were at least 100 digits long, then even the best algorithms known today could not find a given only g, p, and ga mod p, even using all of mankind's computing power.

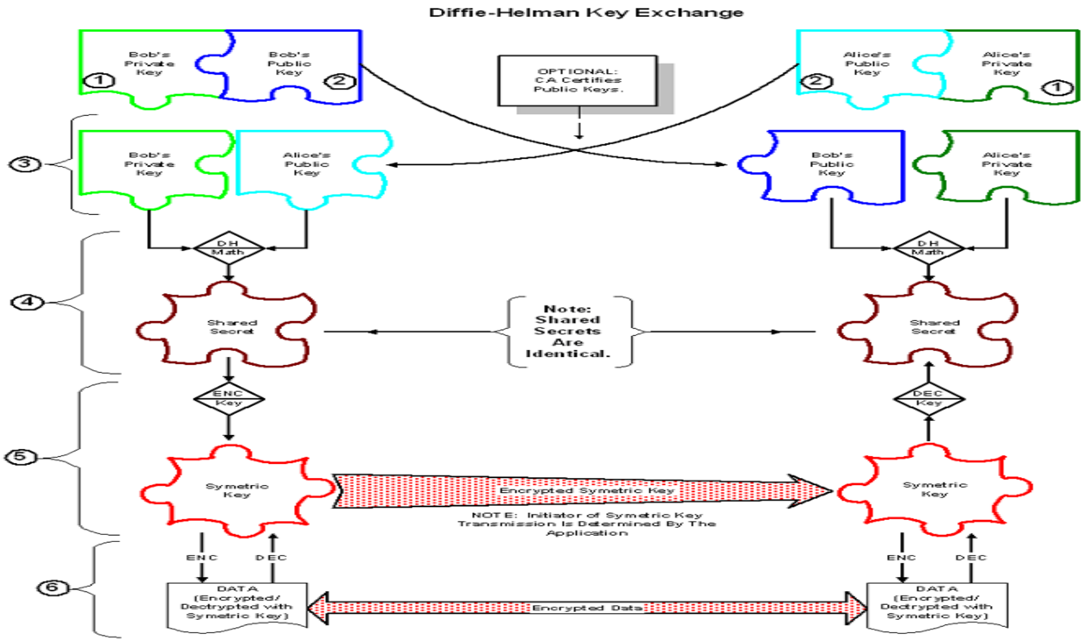

Fig 6: Diffie Hellman Key

5. SECURITY

Data confidentiality: Unauthorized users who do not have enough attribute satisfying the access policy should be prevented from accessing the plaintext of the data. Additionally, the KGC is no longer fully trusted in the data sharing system. Thus, unauthorized access from the KGC as well as the data-storing center to the plaintext of the encrypted data should be prevented.

Key Update: When a user comes to hold or drop an attribute, the corresponding key should be updated to prevent the user from accessing the previous or subsequent encrypted data for backward or forward secrecy, respectively. The key update procedure is launched by the KGC when it receives a join or leave request for some attribute groups from a user. On receipt of the request, the KGC notifies the data storing center of the event and sends the updated membership list of the attribute group to it. When the data-storing center receives the notification, it rekeys the corresponding attribute group key.

Collusion Resistance: Collusion resistance is one of the most important security property required in ABE systems. If multiple users collude, they may be able to decrypt a ciphertext by combining their attributes even if each of the users cannot decrypt the ciphertext alone. We do not want these colluders to be able to decrypt the private data in the server by combining their attributes. Since we assume the KGC and data-storing center are honest, we do not consider any active attacks from them by colluding with revoked users

Backward and forward secrecy: In the context of attribute-based encryption, backward secrecy means that any user who comes to hold an attribute (that satisfies the access policy) should be prevented from accessing the plaintext of the previous data distributed before he holds the attribute. On the other hand, forward secrecy means that any user who drops an attribute should be prevented from accessing the plaintext of the subsequent data distributed after he drops the attribute, unless the other valid attributes that he is holding satisfy the access policy.

6. CONCLUSION

The enforcement of access policies and the support of policy updates are important challenging issues in the data sharing systems. In this study, we proposed an attribute based data sharing scheme to enforce a fine-grained data access control by exploiting the characteristic of the data sharing system. The proposed scheme features a key issuing mechanism that removes key escrow during the key generation. The user secret keys are generated through a secure two-party computation such that any curious key generation center or data-storing center cannot derive the private keys individually. Thus, the proposed scheme enhances data privacy and confidentiality in the data sharing system against any system managers as well as adversarial outsiders without corresponding (enough) credentials. The proposed scheme can do an immediate user revocation on each attribute set while taking full advantage of the calable access control provided by the cipher text policy attribute-based encryption. Therefore, the proposed scheme achieves more secure and fine-grained data access control in the data sharing system. We demonstrated that the proposed scheme is efficient and scalable to securely manage user data in the data sharing

REFERENCES

[1] J. Anderson, “Computer Security Planning Study,” Technical Report 73-51, Air Force Electronic System Division, 1972.

[2] L. Ibraimi, M. Petkovic, S. Nikova, P. Hartel, and W. Jonker, “Mediated Ciphertext-Policy Attribute-Based Encryption and Its Application,” Proc. Int’l Workshop Information Security Applications (WISA ’09), pp. 309-323, 2009.

[3] A. Sahai and B. Waters, “Fuzzy Identity-Based Encryption,” Proc. Int’l Conf. Theory and Applications of Cryptographic Techniques (Eurocrypt ’05), pp. 457-473, 2005.

[4] V. Goyal, O. Pandey, A. Sahai, and B. Waters, “Attribute-Based Encryption for Fine-Grained Access Control of Encrypted Data,” Proc. ACM Conf. Computer and Comm. Security, pp. 89-98, 2006.

[5] J. Bethencourt, A. Sahai, and B. Waters, “Ciphertext-Policy Attribute-Based Encryption,” Proc. IEEE Symp. Security and Privacy, pp. 321-334, 2007.

[6] 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.

[7] Lewko, A. Sahai, and B. Waters, “Revocation Systems with Very Small Private Keys,” Proc. IEEE Symp. Security and Privacy, pp. 273-285, 2010.

[8] . A. Boldyreva, V. Goyal, and V. Kumar, “Identity-Based Encryption with Efficient Revocation,” Proc. ACM Conf. Computer and Comm. Security, pp. 417-426, 2008.

[9] 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.

[10] M. Pirretti, P. Traynor, P. McDaniel, and B. Waters, “Secure Attribute-Based Systems,” Proc. ACM Conf. Computer and Comm. Security, 2006.

[11] S. Rafaeli and D. Hutchison, “A Survey of Key Management for Secure Group Communication,” ACM Computing Surveys, vol. 35, no. 3, pp. 309-329, 2003.

[12] P. Golle, J. Staddon, M. Gagne, and P. Rasmussen, “A Content- Driven Access Control System,” Proc. Symp. Identity and Trust on the Internet, pp. 26-35, 2008.

[13] 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.

[14] S.D.C. Vimercati, S. Foresti, S. Jajodia, S. Paraboschi, and P. Samarati, “Over-Encryption: Management of Access Control Evolution on Outsourced Data,” Proc. Int’l Conf. Very Large Data Bases (VLDB ’07), 2007.

[15] D. Boneh and M.K. Franklin, “Identity-Based Encryption from the Weil Pairing,” Proc. Ann. Int’l Cryptology Conf. Advances in Cryptology (CRYPTO), pp. 213-229, 2001.

[16] A. Kate, G. Zaverucha, and I. Goldberg, “Pairing-Based Onion Routing,” Proc. Privacy Enhancing Technologies Symp., pp. 95-112, 2007.

[17] L. Cheung and C. Newport, “Provably Secure Ciphertext Policy ABE,” Proc. ACM Conf. Computer and Comm. Security, pp. 456-465, 2007.

[18] V. Goyal, A. Jain, O. Pandey, and A. Sahai, “Bounded Ciphertext Policy Attribute-Based Encryption,” Proc. Int’l Colloquium Automata, Languages and Programming (ICALP), pp. 579-591, 2008.

[19] X. Liang, Z. Cao, H. Lin, and D. Xing, “Provably Secure and Efficient Bounded Ciphertext Policy Attribute Based Encryption,” Proc. Int’l Symp. Information, Computer, and Comm. Security (ASIACCS), pp. 343-352, 2009.

[20] The Pairing-Based Cryptography Library, http://crypto.stanford. edu/pbc/, 2012