Volume 2, Issue 3, March 2012 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

Amending Protection and Efficiency in Attribute Information Sharing

Vivekananda G.N.[1]
SVEC, I.T. Department, *JNTUA University*,
Andhra Pradesh, India
E-mail: gnvivekananda@hotmail.com

Obulesu O [2]
SVEC, I.T. Department, *JNTUA University*,
Andhra Pradesh, India

Abstract: **With the high data sharing options there have been increasing demands and concerns for distributed data security; enforcement of access policies is one of the challenging issues. Cipher text policy attribute-based encryption is becoming a promising cryptographic solution to this issue. It enables data owners to define their own policies over user attributes distributed. The advantage comes with a major drawback which is known as a key escrow problem. This is not suitable for data sharing scenarios where the data owner would like to make their private data only accessible to designated users. Applying CP-ABE in the data sharing system introduces another challenge with regard to the user revocation since the access policies are defined only over the attribute universe. In this approach we propose a novel CP-ABE scheme for a data sharing system by exploiting the Characteristic of the system architecture. (1) Escrow-free key issuing protocol solves the key escrow problem. (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.**

Keywords: **Software Security, Efficiency, Info sharing, Attribute Encryption, Cryptography**

1. INTRODUCTION

Recent development of the network and com- puting technology enables many people to easily share their data with others using online external storages. People can share their lives with friends by uploading their private photos or messages into the online social networks such as Facebook and MySpace; or upload highly sensitive personal health records (PHRs) into online data servers such as Microsoft HealthVault, 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.

1.1 Related Work

ABE comes in two flavors called key-policy ABE (KP- ABE) and ciphertext-policy ABE (CP-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 encrypt or determines a policy on who can decrypt the data. Between the two approaches, CP-ABE is more appropriate to the data sharing sys- tem because it puts the access policy decisions in the hands of the data owners.

1.1.1 Removing Escrow

Most of the existing ABE schemes are constructed on the architecture where a single trusted authority, or KGC has the power to generate the whole private keys of users with its master secret information. Thus, the key escrow problem is inherent such that the KGC can decrypt every ciphertext addressed to users in the system by generating their secret keys at any time.

1.2 Contribution

In this paper, we propose a novel CP-ABE scheme for a secure data sharing system, which features the following achievements. First, the key escrow problem is resolved by a key issuing protocol that exploits the characteristic of the data sharing system architecture. The key is- suing protocol generates and issues user secret keys by performing a secure two-party computation (2PC) protocol
between the KGC and the data storing center with their own master secrets.
fine-grained user access control can be possible. Even if a user is revoked from some attribute groups, he would still be able to decrypt the shared data as long as the other attributes that he holds satisfy the access policy of the ciphertext. Data owners need not be concerned about defining any access policy for users, but just need to define only the access policy for attributes as in the previous ABE schemes. The proposed

scheme delegates most laborious tasks of membership management and user revocation to the data storing center while the KGC is responsible for the attribute key management as in the previous CP-ABE schemes without leaking any confidential information to the other parties. Therefore, the proposed scheme is the most suitable for the data sharing scenarios where users encrypt the data only once and upload it to the data storing centers.

2. Data Sharing Architecture

In this section, we describe the data sharing architecture and define the security model.

2.1 System Description and Key Management

Fig. 1 shows the architecture of the data sharing sys- tem, which consists of the following system entities:

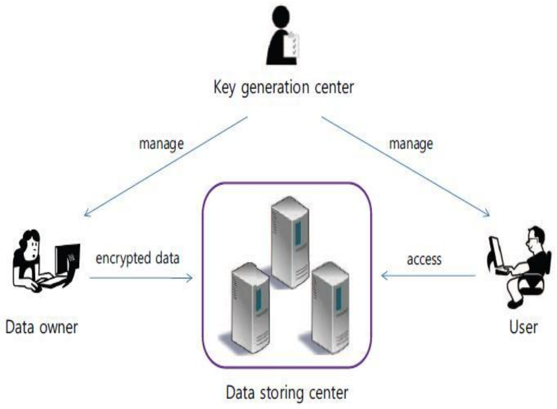

Fig 1. Architecture of data sharing system

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.
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 ser- vices. The data storing center is another key authority that generates personalized user key with the KGC.

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 data.

2.2 Threat Model and Security Requirements

1) 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..
2) 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.

3) 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.

3. Preliminaries and Definition

3.1 Cryptographic Background

We first provide a formal definition for access struc- ture by recapitulating the definitions in [4], [5]. Then we will briefly review the cryptographic background about the bilinear map and its security assumption.

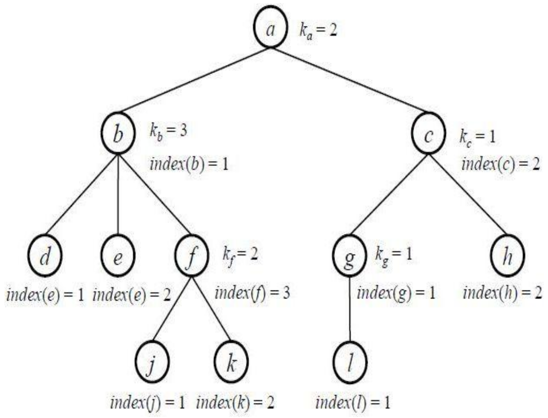

Fig. 2. An example of access tree

4. Proposed CP-ABE Scheme

The first step of the key issuing protocol is to generate the user secret keys using secure 2PC protocol between the KGC and the data storing center.

4.1.1 Description

Let T be a tree representing an access structure. Each non-leaf node of the tree represents a threshold gate.

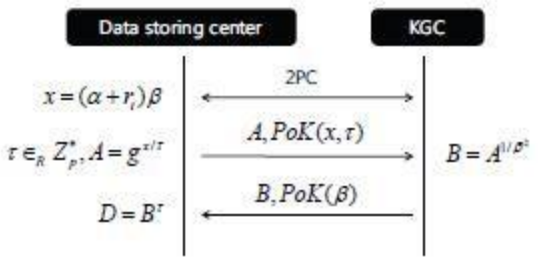

Fig. 3. Key generation protocol

4.1.2 Satisfying an Access Tree

Let Tx be the subtree of *T rooted* at the node x. If a set of attributes γ satisfies the access tree T_x, we denote it as T_x (γ) = 1. We compute T_x (γ) recursively as follows. If x is a non-leaf node, evaluate T_x (γ) for all children x of node x. T_x (γ) returns 1 iff at least k_x children return 1. If x is a leaf node, then Tx (γ) returns 1 iff λ_x ϵ γ.

4.2 Scheme Construction

Let G0 be a bilinear group of prime order p, and let g be a generator of G_0. Let e: $G_0 \times G_0 \rightarrow G_1$ denote the bilinear map. A security parameter, κ, will determine the size of the groups. We will also make use of Lagrange coefficients $\Delta_{i,\Lambda}$ for any i ϵ Z^* and a set.

4.3 Key Update

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 re- quest, 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. Without loss of generality, sup- pose there is any membership change in Gi (e.g., a user comes to hold or drop an attribute λ_i at some time instance).

5. Scheme Analysis

In this section, we analyze and compare the efficiency of the proposed scheme with the previous CP-ABE schemes (that is, Bethencourt et al.'s scheme (BSW) Attrapadung's scheme , and Yu et al.'s scheme) in theoretical and practical aspects. Then, the efficiency of the proposed scheme is demonstrated in the network simulation in terms of the communication cost. We also discuss its efficiency when implemented with specific parameters and com- pare these results with those obtained by the other schemes.

5.1 Key Escrow and Revocation

Table 1 shows the revocation granularity and key escrow problem of each scheme. The rekeying in the proposed scheme can be done in an immediate way as opposed to BSW. Therefore, a user can be revoked at any time even before the expiration time which might be set to the attribute. This enhances security of the shared data in terms of the backward/forward secrecy by reducing the windows of vulnerability. In addition, the proposed scheme realizes more fine-grained user revocation for each attribute rather than for the whole system. Thus, even if a user drops some attributes during the service in the proposed scheme, he can still access the data with other attributes that he is holding as long as they satisfy the access policy. The proposed scheme also resolves the key escrow problem due to the escrow-free key issuing protocol exploiting secure 2PC protocol as opposed to the other schemes.

TABLE 1: Key escrow and revocation comparison

Scheme	Revocation granularity	Key escro
BSW	timed attribute revocation	yes
BCP-ABE2	immediate user revocation	yes
YWRL	immediate user revocation	yes
Proposed	immediate user revocation	no

5.2 Efficiency

C_0	bit size of an element in G_0
C_1	bit size of an element in G_1
C_p	bit size of an element in Z^*
C_T	bit size of an access tree T in the ciphertext
C_k	bit size of a set of attributes associated with private key of a user
t	the number of attributes appeared in T
T	the maximum size allowed for t
m	the number of users in an attribute group
r	the number of revoked users
k	the number of attributes associated with private key of a user
K	the maximum size allowed for k
u	the size of the attribute universe

TABLE 2: Efficiency comparison

Scheme	Revocation granularity	Key escrow
BSW	timed attribute revocation	yes
BCP-ABE2	immediate user revocation	yes
YWRL	immediate user revocation	yes
Proposed	immediate user revocation	No

As shown in Table 2, the proposed scheme requires ciphertext size of $(2t + 1)C0 + C1 + CT$, which is the same as that of BSW.

The proposed scheme requires rekeying message (Hdr) size of $(m + 2)C0$ to realize the user revocation for each attribute in the system. In the proposed scheme, each user stores one more private KEK for decrypting the rekeying messages and obtaining attribute group keys than the basic BSW scheme. YWRL incurs high communication and storage overhead compared to the other schemes in all aspects, of which size are linear to the number of the whole attributes in the system. In YWRL, the KGC should send $2u$ proxy keys to the data server, and $2u$ key components to m users on every revocation in order to re-encrypt the ciphertext and prevent any revoked users from decrypting it.

Although BCPABE2 does not need to send additional rekeying message for user revocations as opposed to the other schemes, it requires ciphertext of which size increases in proportion to the number of revoked users in the system. The proposed scheme is as efficient as BSW in terms of the ciphertext and public key size, while guaranteeing immediate rekeying.

5.3 Simulation

Now we measure the communication cost of the schemes. In this simulation, we consider the online data sharing system connected into the Internet. Almeroth et al. demonstrated the group behavior in the Internet's multicast backbone network .

They showed that the number of users joining a multicast group follows a Poisson distribution with rate $\tilde{\lambda}$, and the membership duration time follows an exponential distribution with a mean duration $1/\mu$. Since each attribute group can be seen as an independent network multicast group where the members of the group share a common attribute, we show the simulation result following this probabilistic behavior distribution

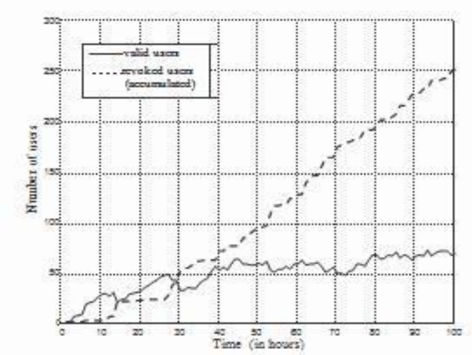

Fig. 4. The number of users in an attribute group

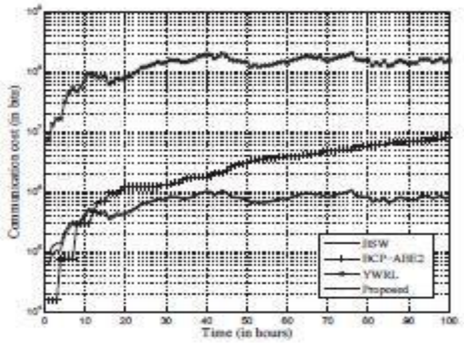

Fig. 5. Communication cost in the system

We suppose that user join and leave events are independently and identically distributed in each attribute group in G following Poisson distribution. The membership duration time for an attribute is assumed to follow an exponential distribution. We set the inter arrival time between users as 20 minutes ($\tilde{\lambda} = 3$) and the average membership duration time as 20 hours ($1/\mu = 20$). *Fig. 4* represents the number of users in a single attribute group during 100 hours. The solid line and dotted line represent the number of current valid users and accumulated revoked users in an attribute group, respectively. *Fig. 5* shows the total communication costs in log scale that the data owner or the data storing center needs to send on a membership change in the network system. It includes the ciphertext and rekeying messages for non-revoked users. It is measured in bits. For a fair comparison with regard to the security perspective, we set the rekeying periods in BSW as $1/\tilde{\lambda}$ minutes. In this simulation, the total number of users in the network is 9150 and the number of attributes to be updated in the system is $30(= t)$. We set $u = 100$. To achieve an 80-bit security level, we set $C0 = 512$, $Cp = 160$. $C1$ and CT are not added to the simulation result because they are common in all of the schemes. As it is shown in *Fig. 5*, YWRL requires the largest amount of communication cost because the rekeying message increases linear to the size of the attribute universe in the whole system. The communication cost in BCP-ABE2 is the lowest in the beginning of the simulation time.

Table 3: Comparison of computation cost

Operation	Time(ms)	BSW		BCP-ABE2		YWRL		Proposed scheme	
		owner	user	owner	user	owner	user	owner	user
Pairing	2.9		$2k + 1$		$2t + 2r + 1$		$u + 1$		$2k + 2$
Exp. in G0	1.0	$2t + 1$		$(K + T + 2)t + 3r + 1$		$u + 1$	k	$2t + 1$	mk
Exp. in G1	0.2	1	log$t$	1	$t + r$	1		1	log$t$
Computation (ms)		$2t + 1.2$	$5.8k + 2.9 + 0.2$log$t$	$(K + T + 2)t + 3r + 1.2$	$6t + 6r + 2.9$	$u + 1.2$	$2.9u + k + 2.9$	$2t + 1.2$	$(5.8 + m)k +$ 0.2log$t + 5.8$

5.4 Implementation

Next, we analyze and measure the computation cost for encrypting (by a data owner) and decrypting (by a user) a data. The decryption cost by a user includes the operations for decrypting the rekeying message as well as the data (in and the proposed scheme). We used a Type A curve (in the pairing-based cryptography (PBC) library) providing groups in which a bilinear map $e : G0 \times G0 \rightarrow G1$ is defined. Although such curves provide good computational efficiency (especially for pairing computation), the same does not hold from the point of view of the space required to represent group elements. Indeed each element of G0 needs 512 bits at an 80-bit security level and 1536 bits when 128-bit of security are chosen.

Table 3 shows the computational time results. For each operation, we include a benchmark timing. Each cryptographic operation was implemented using the PBC library ver. 0.4.18 on a 3.0 GHZ processor PC. The public key parameters were selected to provide 80-bit security level. The implementation uses a 160- bit elliptic curve group based on the super singular curve $y^2 = x^3 + x$ over a 512-bit finite field. The computational cost is analyzed in terms of the pairing, exponentiation operations in G0 and G1. The comparatively negligible hash operations are ignored in the time result. In this analysis, we assume that the access tree in the ciphertext is a binary tree.

6. Security

In this section, we prove the security of the proposed scheme with regard to the security requirements dis- cussed in Section 2.

6.1 Collusion Resistance

In the ciphertext-policy attribute based encryption; the secret sharing must be embedded into the ciphertext instead to the private keys of users. Like the previous ABE schemes, the private keys (SK) of users are randomized with personalized random values selected by the KGC such that they cannot be combined in the proposed scheme. In order to decrypt a cipher- text, the colluding attacker should recover $e(g, g)^{\alpha s}$. To recover this, the attacker must pair Cy from the ciphertext and Dy from the other colluding users' private keys for an attribute λy.

6.2 Data Confidentiality

In our trust model, the KGC is no longer fully trusted as well as the data storing center even if they are honest. Therefore, the plain data to be shared should be kept secret from them as well as from unauthorized users.

6.3 Backward and Forward Secrecy

When a user comes to hold a set of attributes that satisfy the access policy in the ciphertext at some time instance, the corresponding attribute group keys are updated and delivered to the valid attribute group members securely (including the user). In addition, all of the components encrypted with a secret key s in the ciphertext are re-encrypted by the data storing center with a new secret s^i, and the ciphertext components corresponding to the attributes are also re-encrypted with the updated attribute group keys. Even if the user has stored the previous ciphertext before he obtains the attribute keys and the holding attributes satisfy the access policy, he cannot decrypt the pervious ciphertext.

7. 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 a attribute-based data sharing scheme to en- force 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 scalable access control provided by the ciphertext 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 system.

References

[1] M. Petkovic, S. Nikova, P. Hartel, W. Jonker, ”Mediated Ciphertext-Policy Attribute-Based Encryption and Its Application,” Proc. WISA 2009, LNCS 5932, pp. 309–323, 2009.

[2] Bethencourt, A. Sahai, B. Waters, ”Ciphertext-Policy Attribute-Based Encryption,” Proc. IEEE Symposium on Security and Privacy 2007, pp. 321–334, 2007.

[3] R. Ostrovsky, A. Sahai, B. Waters, ”Attribute-Based Encryption with Non-Monotonic Access Structures,” Proc. ACM Conference on Computer and Communications Security 2007, pp. 195–203, 2007.

[4] A. Lewko, A. Sahai, B. Waters, ”Revocation Systems with Very Small Private Keys,” Proc. IEEE Symposium on Security and Privacy 2010, pp. 273–285, 2010.

[5] N. Attrapadung, H. Imai, ”Conjunctive Broadcast and Attribute-Based Encryption,” Proc. Pairing 2009, LNCS 5671, pp. 248–265, 2009.

[6] S. Yu, C. Wang, K. Ren, W. Lou, ”Attribute Based Data Sharing with Attribute Revocation,” Proc. ASIACCS ’10, 2010.

[7] S. D. C. Vimercati, S. Foresti, S. Jajodia, S. Paraboschi, P.Samarati, ”Over-encryption: Management of Access Control Evolution on Outsourced Data,” Proc. VLDB’07, 2007.

[8] A. Kate, G. Zaverucha, and I. Goldberg, ”Pairing-based onion routing,” Proc. Privacy Enhancing Technologies Symposium 2007, LNCS vol. 4776, pp. 95–112, 2007.

[9] L. Cheung, C. Newport, ”Provably Secure Ciphertext Policy ABE,” ACM Conference on Computer and Communications Security, pp. 456–465, 2007.

[10] V. Goyal, A. Jain, O. Pandey, A. Sahai, ”Bounded Ciphertext Policy Attribute-Based Encryption,” Proc. ICALP, pp. 579–591,2008.

[11] X. Liang, Z. Cao, H. Lin, D. Xing, ”Provably Secure and Efficient Bounded Ciphertext Policy Attribute Based Encryption,” Proc. ASIACCS, pp. 343–352, 2009.

[12] M. Chase, S.S.M. Chow, ”Improving Privacy and Security in Multi-Authority Attribute-Based Encryption,” Proc. ACM Conference on Computer and Communications Security, pp.121–130, 2009.

[13] S.S.M. Chow, ”Removing Escrow from Identity-Based Encryption,” Proc. PKC 2009, LNCS 5443, pp. 256–276, 2009.

[14] M. Belenkiy, J. Camenisch, M. Chase, M. Kohlweiss, A.Hysyanskaya, H. Shacham, ”Randomizable Proofs and Delegatable Anonymous Credentials,” Proc. Crypto 2009, LNCS 5677, pp. 108–125, 2009.

G.N.Vivekananda received the B.Tech. Degree in Computer Science and Engineering from JNTUA University in 2010 and pursuing M.Tech. Degree final in Software Engineering, Sree Vidyanikethan Engineering College. His research interests include information security, Information Retrieval systems, network and system security, and applied cryptography

O.Obulesu received the B.Tech.degree in Computer Science and Engineering from Sri Venkateswara University and M.Tech. Degree in Computer Science from JNTUA, Anantapur. He received Gold Medal award in M.Tech (CS) Course in the year 2008.He is currently pursuing Ph.D. (CSE) in J.N.T.U.A, Anantapur. He has 4 years of teaching experience. He is currently an Assistant Professor in the Information Technology Department at Sree Vidyanikethan Engineering College, Tirupati.His current research interests include Spatial Data Mining and Spatiotemporal Databases.