ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 12, December 2014

Cipher Text Policy Based Attribute Encryption

A.Rohidha[1], D.Sivaharani[2], P.Aurchana[3]

Dept of MCA, Sri ManakulaVinayagar Engineering College, Madagadipet, Pondicherry India[1,2]

Assistant Professor, Sri ManakulaVinayagar Engineering College, Madagadipet, Pondicherry India[3]

ABSTRACT:In a cipher text policy attribute based encryption system, a user's private key is associated with a set of attributes (describing the user) and an encrypted cipher text will specify an access policy over attributes. A user will be able to decrypt if and only if his attributes satisfy the cipher text's policy. 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. Cipher text 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. However, the advantage comes with a major drawback which is known as a key escrow problem .The key generation center could decrypt any messages addressed to specific users by generating their private keys. This is not suitable for data sharing scenarios where the data owner would like to make their private data only accessible to designated users. In addition, 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.

KEYWORDS*:* Data sharing, attribute-based encryption, stored procedure, removing escrow, Security, Algorithms, Design.

I. INTRODUCTION

The network and computing technology enables many people to easily share their data with others are 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 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. We often identify people by their attributes. In 2005, Sahai and Waters proposed a system (described in more recent terminology as a key-policy attribute-based encryption (ABE) system for threshold policies) in which a sender can encrypt a message specifying an attribute set and a number d, such that only a recipient with at least d of the given attributes can decrypt the message. However, the deployment implications of their scheme may not be entirely realistic, in that it assumes the existence of a single trusted party who monitors all attributes and issues all decryption keys. In CP-ABE, the key generation centre (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. This could be a potential threat to the data confidentiality or privacy in the data sharing systems. Another challenge is the key revocation. Since some users may change their associate attributes at some time, or some private keys might be compromised, key revocation or update for each attribute is necessary in order to make systems secure. This issue is even more difficult especially in ABE, since each attribute is conceivably

10.15680/ijircce.2014.0212067

shared by multiple users (henceforth, we refer to such a set of users as an attribute group). This implies that revocation of any attribute or any single user in an attribute group would affect all users in the group. It may result in bottleneck during rekeying procedure or security degradation due to the windows of vulnerability.

II. 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.
- CP-ABE is more appropriate to the data sharing system because it puts the access policy decisions in the hands of the data owners.

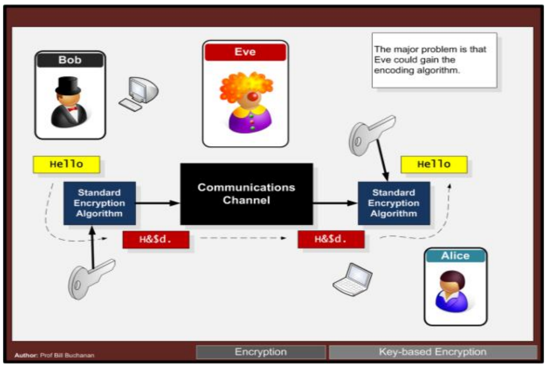
Bob
Eve
The major problem is that Eve could gain the encoding algorithm.
Hello
Standard Encryption Algorithm
Communications Channel
Standard Encryption Algorithm
Hello
H&$d.
H&$d.
Alice
Author: Prof Bill Buchanan
Encryption
Key-based Encryption

Fig 1 Key Generation Centre

III. 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 cipher text addressed to users in the system by generating their secret keys at any time.

IV. REVOCATION

In 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 re encrypted 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. Recently, the importance of immediate user revocation (rather than attribute revocation) has been taken notice of in many practical ABE-based systems. The user revocation can be done by using ABE that supports negative clauses, proposed by Ostrovsky et al. To do so, one just adds conjunctively the AND of negation of revoked user identities (where each is considered as an attribute here). One drawback in this scheme is that the private key size increases by a multiplicative factor.

V. DATA SHARING ARCHITECTURE

1. Key generation centre. 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. It is assumed to be honest-but-curious. 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 centre. 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 centre 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. Similar to the Previous schemes we assume theData-storing centre is also semi trusted (that is, honest-but-curious) like the KGC.

3. Data owner. It is a client who owns data, and wishes to upload it into the external data-storing centre 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 distributingit.

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 cipher text and obtain the data.

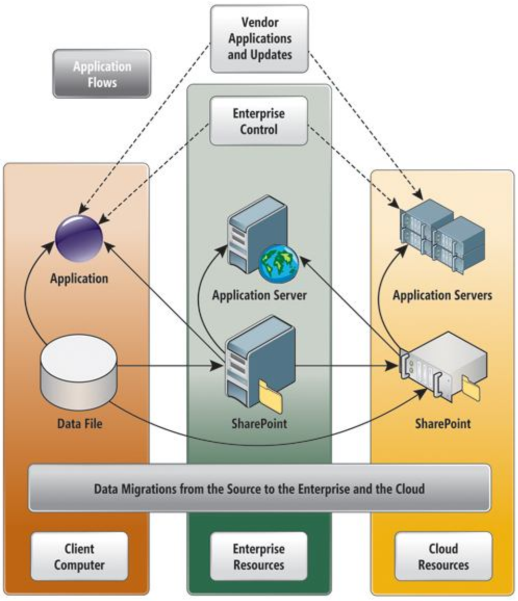

Fig 2 .Data Sharing Architecture

VI. PROTECTING THE USER PRIVACY

Since each authority is responsible for different attributes, we want to allow them to issue decryption keys independently, without having to communicate with one another. As argued in order to prevent collusion in such a setting, we need some consistent notion of identity. (Otherwise, a user could easily obtain keys from one authority and then give them all to a friend.) The solution in that work is to require that each user have a unique global identifier (GID), which they must present to each authority (and to require that the user prove in some way that he is the owner of the GID he presents).1 Unfortunately, the mere existence *of GID* makes it very hard for the users to guarantee any kind

10.15680/ijircce.2014.0212067

of privacy. Because a user must present the same GID to each authority, it is very easy for colluding authorities to pool their data and build a “complete profile” of all of the attributes corresponding to each GID. However, this might be undesirable, particularly if the user uses the ABE system in many different settings, and wishes to keep information about some of those settings private. This situation seems to be unavoidable if all one’s attributes are determined by some kind of public identity like a name or SSN – in that case users will need to identify themselves in any case in order to get the decryption keys for a certain set of attributes, so privacy is unavoidably com-1see for further discussion promised.

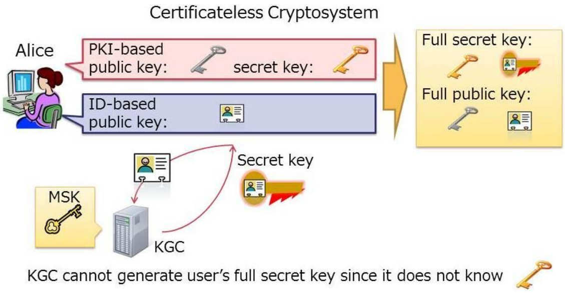

Fig 3 .Decryption and Encryption

6.1 ABE FOR DIFFERENT POLICIES

ABE is actually a generalization of IBE (identity-based encryption): in an IBE system, cipher texts are associated with only one attribute (the identity).The ABE scheme of Sahai-Waters [13] was proposed as a fuzzy IBE scheme, which allowed for some error tolerance around the chosen identity. In more recent terminology, it would be described as a key-policy (KP) ABE scheme that allows for threshold policies. Key-policy means that the encrypt or only gets to label a cipher text with a set of attributes. The authority chooses a policy for each user that determines which cipher texts he can decrypt. A threshold policy system would be one in which the authority specifies an attribute set for the user, and the user is allowed to decrypt whenever the overlap between this set and the set associated with a particular cipher text is above a threshold.

Goyal*et al.* proposed a KP-ABE scheme which supports any monotonic access formula consisting of AND, OR, or threshold gates. A construction for KP-ABE with no monotonic access structures (which also include NOT gates,i.e. negative constraints in a key’s access formula) waspro-2See footnote 6.posed by Ostrovsky, Sahai and Waters .All of these schemes are characterized as key-policy ABE since the access structure is specified in the private key, while the attributes are used to describe the cipher texts. The roles of the cipher texts and keys are reversed in the cipher text-policy ABE (CP-ABE) introduced by Bethencourt, Sahai and Waters, in that the cipher text is encrypted with an access policy chosen by an encrypt or but a key is simply created with respect to an attributes set. The security of their scheme is argued in the generic group model. Recently, proposed CP-ABE constructions based on a few different pairing assumptions which work for any access policy that can be expressed in terms of an LSSS matrix. In this paper, we will look only at the KP-ABE setting. We will look at both the simple threshold, and the more complicated monotonic access structure case, and will build a construction based on the same assumptions as Sahai and Waters and Goyal*et al.* Both non-monotonic access structures and the cipher text policy schemes require much stronger assumptions, and very different techniques, so we will not consider these cases in our work.

VII. PRELIMINARIES

7.1 NOTATIONS AND COMPLEXITY ASSUMPTIONS

Let G1 and G2 be two cyclic multiplicative groups of prime order q generated by g1 and g2 respectively, $\hat{e} : G1 \times G2 \rightarrow GT$ be a bilinear map such that $\forall x \in G1$, $y \in G2$ and $a,b \in Zq$, $\hat{e}(xa, yb) = \hat{e}(x, y)ab$ and $\hat{e}(g1, g2)$ 6= 1. Let $\psi : G2 \rightarrow G1$ be a computable isomorphism from G2 to G1,with $\psi(g2) = g1$. (G1,G2) are said to be admissible bilinear groups if the group action in G1, G2, the isomorphism ψ andthe bilinear mapping $\hat{e}$ are all efficiently computable.

10.15680/ijircce.2014.0212067

Definition 1. The Decisional Diffie-Hellman (DDH) problem in prime order group G = hgiis defined as follows: on input g, ga, gb, gc∈G, decide if c = ab or c is a randomelement of Zq.
Definition 2. Let algorithm BDH Gen(1_) output the parameters (ˆe(• , •), q, g1, g2,G1,G2,GT) where there is an efficientlycomputable isomorphism ψ from G2 to G1. The Decisional Bilinear Diffie-Hellman (DBDH) problem is defined as follows: given g1 ∈G1, g2, ga2 , gb2 , gc2 ∈G2 and Z ∈GT as input, decide if Z = ˆe(g1, g2)abc or ˆe(g1, g2)R for R ∈R Zq.The security of the ABE schemes by Sahai-Waters Goyal et al., and Chase, and of our construction rely on the intractability of the DBDH problem.
Definition 3.The k-Decisional Diffie-Hellman Inversion (k-DDHI) problem in prime order group G = hgiis defined as follows: On input a (k + 2)-tuple g, gs, gs2, . . .gsk, gu∈Gk+2, decide if u = 1/s or u is a random element of Zq. For our key issuing protocol, we will use a modified version of the of the Dodis-Yampolskiy PRF, suggested in which relies on the intractability of the k-DDHI problem in group G1 of a pairing. Note that k-DDHI is solvable when given a DDH oracle, thus we must also make the following assumption:
Definition 4.*Let* BDH Gen(1_) output the parameters for a bilinear mapping ˆe : G1 × G2 → GT . The external Differ Hellman (XDH) assumption states that, for all probabilistic polynomial time adversaries A, the DDH problem is hard in G1. This implies that there does not exist an efficiently computable isomorphism ψ′ : G1 → G2.

7.2 DEFINITIONS OF MULTI-AUTHORITY ABE

We begin by defining a multi-authority ABE scheme with a trusted setup (but without an online trusted CA), and without any privacy guarantees. For now, we consider a key policy threshold scheme, where the user's decryption key corresponds to a set of attributes and a threshold value. (See Section 6 for an extension to more general policies.) In Section 4 we will discuss an extension which allows a user to obtain decryption keys without leaking his GID, and in Section 5 we will discuss an extension which replaces Setup with an interactive protocol between the authorities.

In a multi-authority ABE system, we have many attribute authorities, and many users. There are also a set of system wide public parameters available to everyone (either created by a trusted party, or by a distributed protocol between the authorities). A user can choose to go to an attribute authority, prove that it is entitled to some of the attributeshandled by that authority, and request the corresponding decryption keys. The authority will run the attribute key generation algorithm, and return the result to the user. Any party can also choose to encrypt a message, in which case he uses the public parameters together with an attribute set of his choice to form the cipher text. Any user who has decryption keys corresponding to an appropriate attribute set can use them for decryption. In what follows, we use GID to denote the global identity of a user and A to denote a set of attributes. We use Au and AC to denote the attribute set of a user and that specified by a cipher text respectively. We assume all the attribute sets can be partitioned into N disjoint sets, handled by the N attribute authorities, and we use a subscript k to denote the attributes handled by the authority k.

Definition 5. An N-authority ABE scheme consists of four algorithms:
1. via (params, {(apkk, askk)}k∈{1,...N}) $←Setup(1_,N) the randomized key generation algorithm takes a security parameter λ ∈N and the number of authorities N ∈N, and outputs the system parameters paramsand N public/private key pairs (apkk, askk), one for each attribute authority k ∈ {1, . . .N}. The threshold values {dk}k∈{1,...N} for each authority are also included in params. For simplicity, we assume paramsand {apkk}k∈{1,...N} are the implicit inputs of the rest of the algorithms.
2. viauskk[GID,Ak] $←AKeyGen(askk,GID,Ak) the attributeauthorityk uses its secret key askk to output
a decryption key corresponding to the attribute set Ak for the user with identity GID.
3. viaC $←Enc({Ak}k∈{1,...N},m) a sender encrypts a message m for the set of attributes {Ak}, resulting in a ciphertextC, where Ak denotes a subset of the attribute domain of the authority k.
4. via m← Dec({uskk[GID,Ak]}k∈{1,...N},C) a user GID who possesses a sufficient set of decryption keys {uskk[GID,Ak]} from each authority k decrypts C to recover m.
Definition 6. An N-authority ABE scheme satisfies the consistency property if for all λ,N∈N, all identities GID A Bounded Cipher text Policy Attribute Based Encryption scheme consists of four algorithms. Setup (d, num) This is a randomized algorithm that takes as input the implicit security parameter asnd a pair of system parameters (d, num). These parameters will be used to restrict the access trees under which messages can be encrypted in our system. It outputs the public parameters PK and a master key MK. Key Generation (γ,MK) This is a randomized algorithm that

takes as input – the master key MK and a set of attributes γ. It outputs a decryption key D corresponding to the attributes in γ. Encryption (M,PK, T ′) This is a randomized algorithm that takes as input – the public parameters PK, a message M, and an access tree T ′ over the universe of attributes, with depth d′ ≤ d, and where each non-leaf node x has at most num child nodes. The algorithm will encrypt M and output the cipher text E. We will assume that the cipher text implicitly contains T ′. Decryption (E,D) This algorithm takes as input – the cipher text E that was encrypted under the access tree T ′, and the decryption key D for an attribute set γ. If the set γ of attributes satisfies the access tree T ′ (i.e. γ ∈ T ′), then the algorithm will decrypt the cipher text and return a message M.

VIII. CONCLUSION

The partnership of recipients in an ABE system plays a key role in analysing the security of the system which has been neglected in the past decade. We find an ABE system cannot resist decryption-key-sharing attack. The flaw renders the primitive impractical. This scheme is very expressive and provably secure under the decisional Bilinear Diffie-Hellman assumption.

IX. FUTURE ENHANCEMENT

Future works includes the revocation of key escrow problem and provide data sharing in multiple ways and provide only keys without any attribute to access the data provide for the user.

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

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

10.15680/ijircce.2014.0212067