LARGE UNIVERSE CP-ABE WITH WHITEBOX TRACEABILITY

Anusha Sivanandhan[1] and Angel M Eldhose[2]

[1] Student,Department of Computer Science and Engineering, MBITS Nellimattom
[2]Assistant Professor, Department of Computer Science and Engineering, MBITS Nellimattom

ABSTRACT

In a ciphertext-policy attribute-based encryption (CP-ABE) system, decryption keys are defined over attributes shared by multiple users. Traceability is the ability of ABE to trace the malicious users or traitors who intentionally leak the partial or modified decryption keys for profits. Nevertheless, due to the nature of CP-ABE, it is difficult to identify the original key owner from an exposed key since the decryption privilege is shared by multiple users who have the same attributes. In this paper, we propose a new CP-ABE system that supportstraceability of malicious users who leaked their decryption privileges.Thistraceable CP-ABE does not weaken the expressivenessor efficiency when compared with the most efficient conventional(non-traceable) CP-ABE systems. In our newsystems attributes need not be fixed at system setup,the attributes' size is not polynomially bounded and thepublic parameters' size does not grow linearly with thenumber of attributes.

KEYWORDS

Attribute-based encryption,ciphertext-policy,large universe ,white-box traceability,malicious user.

1. INTRODUCTION

The notion of Attribute-Based Encryption (ABE) was introduced as a generalization of fuzzy Identity-Based Encryption (IBE)[1],[2]. In a CP-ABE system, each user is issued a decryption key by an authority according to the attributes he possesses, and the encryptor decides what attributes the eligible receivers should have by encrypting the messages with an access policy defined over some attributes. If and only if a user's attributes satisfy the access policy of a ciphertext, he can decrypt the ciphertext. Not only does ABE (especially CP-ABE) provide a new promising tool for implementing fine-grained access control over encrypted data, but also has it attracted much attention in the research community.

In general, an ABE system can be classified to "small universe" and "large universe" constructions. In the "small universe" construction, the attributes are fixed at system setup and the size of the attributes is polynomially bounded, and furthermore the size of public parameters grows linearly with the number of attributes. While in the "large universe" construction, the attributes need not be specified at system setup and the size of the attribute universe is unbounded. The "large universe" construction for ABE system brings an obvious advantage that the designer of the ABE system need not bother to choose a particular bound of the attributes at system setup.

In CP-ABE, each user possesses a set of attributes and can decrypt the ciphertext if his/her attributes satisfy the ciphertext's access policy. This results in an obvious consequence that the

DOI: 10.5121/ijci.2016.5222

encryptor or system does not know who leaks the decryption key to others intentionally. Due to the fact that the attributes are shared by multiple users and different users may have the same subset of attributes, the encryptor or system has no feasible method to trace the suspicious receiver if the decryption key is leaked. We take Alice (with attributes {Alice, Assistant Professor, Computer Science}) and Bob (with attributes {Bob, Assistant Professor, Computer Science}) as an example. They both have the same decryption keys corresponding to attributes {Assistant Professor, Computer Science} and can decrypt such a ciphertext encrypted by the attributes {Assistant Professor, Computer Science}. Suppose no other receiver in the system has both attributes ({Assistant Professor} and {Computer Science}) at the same time. If it happens to exist a user who can decrypt the ciphertext except Alice and Bob, it is significant to find out who leaks such decryption key to him, Alice or Bob?

This drawback should be fixed in practice in case of leaking decryption key. It is necessary to add the property of traceability to the original ABE scheme, to identify who exactly leaks the decryption key. The above traceability is called white-box traceability, which means that any user who leaks his/her decryption key to the third user or device intentionally or unintentionally will be identified. However, up to now, there exists no practical traceable CP-ABE system supporting the property of large universe as the (non-traceable) CP-ABE system. Large universe CP-ABE system with white-box traceability is not yet achieved in practice: (1) The CP-ABE systems supporting traceability so far proposed do not support the property of large universe, the attributes need to be fixed at system setup and the size of the attributes is polynomially bounded. Besides, public parameters' size grows linearly with the number of attributes. (2) The large universe CP-ABE system proposed is secure in the standard model; however, it does not support the property of traceability.

2. RELATED WORKS

Sahai and Waters introduced the notion of Fuzzy Identity- Based Encryption[36]. Goyal *et al.* later formalized two notions of ABE[14]: CP-ABE (where user keys are labeled with sets of attributes and ciphertexts are associated with policies) and KP-ABE (where ciphertexts are labeled with sets of attributes and private keys are associated with access structures). Subsequently, many constructions of selectively secure KP ABE and CP-ABE systems were proposed[6],[7],[8],[5],[4],[3]. Many advances have been made for ABE as the following directions: new proof techniques to obtain fully secure , decentralizing trust by setting multiple authorities and outsourcing computation.

The first large universe KP-ABE construction was proposed in unbounded hierarchical based encryption[9]n. It was built on composite order groups and proved selectively secure in the standard model. Then the first large universe KP-ABE construction on prime order groups proposed[10] was inspired by the dual pairing vector space framework. Recently, the first large universe CP-ABE construction built on prime order bilinear groups was proposed by Rouselakis and Waters. It was proved selectively secure in the standard model under "*q*-type" assumption. Another branch of ABE research considers the problem of traceability. The notion of accountable CP-ABE was first proposed to prevent illegal key sharing among colluding users. Then a multi-authority ciphertext-policy (AND gates with wildcard) ABE scheme with accountability was proposed in, which allowed tracing the identity of a misbehaving user who leaked the decryption key to others. Liu, Cao and Wong lately proposed a white-box and black-box traceability CP-ABE system which supported policies expressed in any monotone access structures.

Black-Box Traceable ABE Systems. In our construction, we target to make decryption key leakage to be traceable in the white-box model, i.e., the decryption keys leaked/sold will be used by the buyers to perform decryption using the ABE decryption algorithm. In practice, a stronger traceability notion is called black-box traceability, which is analogous to the notion of black-box traitor tracing in broadcast encryption [11], [12]. In particular, given a decryption equipment (where the embedded decryption key or algorithm could be unknown or hidden), the buyers can use it to retrieve plaintexts from ciphertexts. A black-box traceable ABE should allow an authority to find out the identity of the malicious user (i.e., whose decryption keys are used to create this decryption equipment).

3. PROBLEM DEFINITION

In ABE, the decryption privilege of a decryption key is shared by multiple users who possess the corresponding attributes, so that any malicious owner of a decryption key would have the intention or be very willing to leak partial or even his entire decryption privilege for financial interest or any other incentive, especially when there is no risk of getting caught. We refer to this issue as MaliciousKeyDelegation.Nevertheless, due to the nature of CP-ABE, it isdifficult to identify the original key owner from an exposed keysince the decryption privilege is shared by multiple users whohave the same attributes. In general, an ABE system is small universe, In the “small universe” construction, the attributes are fixed at system setup and the size of the attributes is polynomially bounded, and furthermore the size of public parameters grows linearly with the number of attributes.

Consider a commercial application such as a pay-TV system with huge number of users for example. Each user is labeled with lots of related attributes, which are defined as TV channels that the user have ordered. As a versatile one-to-many encryption mechanism, CP-ABE system is quite suitable in this scenario. The pay-TV system provides several TV channels for users, and those who have paid for the TV channels could satisfy the access policy to decrypt the ciphertext and enjoy the ordered TV channels. CPABE enables fine-grained access control to the encrypted data according to attributes in users’ ordered lists. However, there are two problems with this approach. First, if someone (who does not have the privilege to access to those TV chat a lower cost, she/he could also get access to the TV channels. It is necessary to find out who is selling the decryption key. Second, as the TV channels of the pay-TV system expand, an increasing number of new attributes need to be added to the system to describe the new channels. If the number of the attributes exceeds the bound set during the initial deployment of the pay-TV system, then the entire system has to be re-deployed and possibly all its data will have to be re-encrypted, which would be a disaster to the pay-TV in the commercial applications. The problems, as described above, are the main obstacles when CP-ABE is implemented in commercial applications such as pay-TV systems and social networks. Due to the nature of CP-ABE, if a malicious user leaks its decryption key to others for profits (such as selling the decryption key on the Internet), it is difficult to find out the original key owner from an exposed key since the decryption key is shared by multiple users who have the same attributes.

Inorder to solve this issue,large universe enhanced traceable system was introduced. The main features of this system are:

1) *White-box traceability*- Our new systems can trace the malicious users or traitors who may leak the partial or modified decryption keys to others for profits.
2) *Large universe*. In our new systems attributes need not be fixed at system setup, the attributes’ size is not polynomially bounded and the public parameters’ size does not grow linearly with the number of attributes.
3) *Constant storage overhead*- we adopt the Shamir’s *(.t, . n)* threshold scheme in tracing the malicious users or traitors, the storage cost for traceability does not grow linearly with the number of the users, it is constant which only depends on the Threshold *.t* .

4) *Dynamical scalability*- It yields another result that the stored data for traceability need not be updated when new users are added into the system or malicious users are removed from the system, which makes the system more practical for applications.

4. Enhanced Traceable Large Universe CP-ABE System

To realize large universe construction, we adopt the “individual randomness” and “layer” technique from [9] and [13]. We use the “layer” technique to encrypt data securely and to be able to decrypt. We employ two “layers”: the “attribute” layer and the “secret sharing” layer, and use a “binder term” to connect these two layers securely. In the “attribute” layer, we utilize *u, h* terms to provide a Boneh-Boyen-style [14] hash function *(u Ah).* As for the “secret sharing” layer, during **KeyGen**and **Encrypt** phases we use *w* term to hold the secret randomness *r* and the secret randomness*s* shares respectively. Finally, we use the *v* term to “bind” this two layers together.

To realize traceability, we use the Boneh-Boyen-style signature [14] in both the T-LU-CPABE system and the eT-LU-CPABE system. Furthermore, we find that the identity table *T* with the tuple identity and its randomness used in [26] and the T-LU-CPABE system grows linearly with the number of the users.2 With the number of the users in a system scaling large, the corresponding identity table *T* for traceability will expand as a result, which leads to heavy burden of the storage space for *T*. Besides, the corresponding cost of searching *K_* in *T* during the **Trace** phase is relatively huge. In our eT-LU-CPABE system, we utilize the Shamir’s *(.t, .n)* threshold scheme to optimize the property of traceability. We only need store *.t* − 1 points and the value *f* (0*)* on a polynomial *f (x)* at system setup. Consequentially, our storage for traceability does not grow linearly with the number of the users and is a constant.

4.1.Model

An enhanced Traceable Large Universe CP-ABE system (eT-LU-CPABE system) is a CP-ABE system where attributes need not be fixed at system setup and can trace the user by his/her decryption key. Moreover, we enhance the T-LU-CPABE system by eliminating the identity table *T*. In this eT-LU-CPABE system, we utilize the Shamir’s*(.t, n)* threshold scheme to optimize the property of traceability. In our eT-LU-CPABE system, we utilize the Shamir’s *(.t, n)* threshold scheme to optimize the property of traceability.

In a Traceable CP-ABE system (T-CPABE system) it is not required that the attributes for the encryption process be fixed at the setup phase.A Traceable Large Universe CP-ABE system (T-LU-CPABE system) is a CP-ABE system where attributes need not be fixed at system setup and can trace the user by his/her decryption key. We enhance the original large universe CP-ABE system by adding users’ identities and a **Trace** algorithm. The identities of the user are added whenever they register into the system.Compared with the T-LU-CPABE System, the significant and remarkable advantage of our new eT-LU-CPABE system is that the system does not need to maintain the identity table *T* and the storage overhead for traitor tracing is constant.

The main idea of our traceability in the eT-LU-CPABE system is as follows.

Firstly, the **Setup** algorithm initializes an instance of Shamir’s *(.t ,. n)* threshold scheme *INS(.t, . n)* and keeps a polynomial *f (x)* and *.t* − 1 points {*(x*1*, y*1*),(x*2*, y*2*), . . . , (x.t*−1*, y.t*−1*)*} on *f (x)* secret. Then we insert *c* into the decryption key *sk*during the **KeyGen**phase where *c* = *Enck.*2 *(x*||*y), x* = *Enck.*1 *(id), y* = *f (x).* Note that the tuple *(x, y)* is a point on *f (x)*. During the **Trace** phase, the algorithm extracts *(x*∗ = *x_, y*∗ = *y_)* from *x_*||*y_* = *Dec .k*2 *(K_)* in the decryption key *sk*,and then it checks whether *sk*is issued by system.If*(x*∗ = *x_ , y*∗ = *y_)* ∈ {*(x*1*, y*1*), (x*2*, y*2*), . . .*

, *(x.t*−1*, y.t*−1*)*}, the algorithm computes *Deck*.1 *(x∗)* to get *id* to identify themalicious user directly. Otherwise, the algorithm computesthe secret of *INS (.t, .n)* by interpolating with *.t* − 1 points{*(x*1*, y*1*), (x*2*, y*2*), . . . , (x.t*−1*, y.t*−1*)*} and *(x∗, y∗)*. If therecovered secret is equal to *f (*0*)*, the algorithm computes *Deck*.1 *(x∗)* to get *id* to identify the malicious user. If the equation fails, *sk*is not issued by the system. In this way, we could trace the owner of the decryption key. Meanwhile, it brings the benefit that the system only stores *f (*0*)* and *.t* − 1 points on *f (x)*, and thus the storage for traceability is a constant.This system is more secure when compared with others.

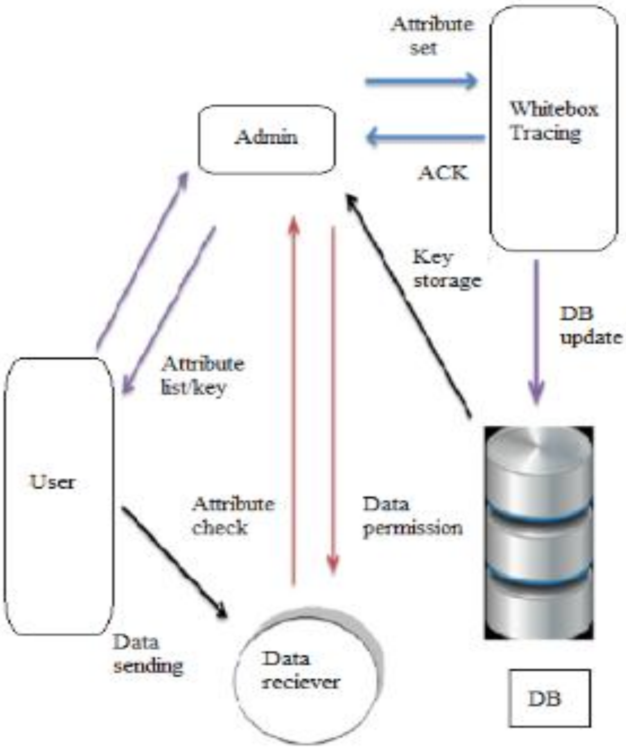

Figure 1. System Architecture

An eT-LU-CP-ABE system consists of six algorithms as follows:

• **Setup**: The algorithm takes as inputs a security parameter *λ* ∈ N encoded in unary. It outputs the public parameters *pp* and the master secret key *msk*. We assume that the description of the attribute universe *U* is contained in the public parameters. In addition, it initializes an instance of Shamir's *(.t ,.n)* threshold scheme denoted by *INS(.t, . n)*.

• **KeyGen**: The key generation algorithm takes as inputs the public parameters *pp*, the master secret key *msk*and a set of attributes *S* ⊆*U* for a user with identity *id*. The security parameter in the inputs ensures that it is polynomial time in *λ*. The algorithm outputs a secret key *skid,S*corresponding to *S*.

• **Encrypt**: The encryption algorithm takes as inputs the public parameters *pp*, a plaintext message *m*, and an access structure A over *U*. It outputs the ciphertext*ct*.

• **Decrypt**: The decryption algorithm takes as inputs the public parameters *pp*, a secret key *skid,S*, and a ciphertext*ct*. It outputs the plaintext *m* or ⊥.

• **KeySanityCheck**: The decryption algorithm takes as inputs the public parameters *pp* and a secret key *sk*. If *sk*passes the key sanity check, it outputs 1. Otherwise, it outputs 0. The key sanity check is a deterministic algorithm, which is used to guarantee the secret key to be well-formed in the decryption process.

• **Trace**: This algorithm is used to identify the malicious user. From the above algorithm we can conclude whether the key was well-formed or not. The tracingalgorithm takes as inputs the public parameters *pp*,an instance of Shamir's *(.t, . n)* threshold scheme *INS (.t, . n)*, the master secret key *msk*, and a secretkey*sk*. The algorithm first verifies whether *sk*is well formedto determine whether *sk*needs to be traced.If*sk*is well-formed and could recover the secret of *INS (.t, .n)*, the algorithm outputs an identity *id* implyingthat*sk*is linked to *id*. Otherwise, it outputs a specialsymbol _ implying that *sk*does not need to be traced.We define a secret key *sk*is *well-formed* which meansthatKeySanityCheck*(pp, sk)*→ 1.

4.2. SHAMIRS THRESHOLD SCHEME

It is well known for Shamirs (t, n) threshold scheme (or Shamirs secret sharing scheme) in cryptography. The essential idea of that scheme is that t points on a t - 1 degree curve are sufficient to confirm such a curve, that is, t points are enough to determine a t - 1 degree polynomial. For a (t; n) threshold scheme, a secret can be divided into n parts (or even more), which are sent to each participant a unique part. All of them can be used to reconstruct the secret. Suppose that the secret is assumed to be an element in a finite field Fp . Choose t - 1 number of random coefficients a1, a2, at-2 element of Fp and at-1 element of Fp and set the secret in the constant term a0. Every participant is given a point (x, y) on the above curve, that is, the input to the polynomial x and its output y = f(x). Given a subset with any t points, recover the constant term a0 using the Lagrange interpolation. White box traceability is implemented in this paper as a web application.

4.3. PROBABILISTIC EQUATION

Probabilistic encryption is an encryption algorithm with some randomness during the encryption, which leads that encrypting same messages yields different ciphertexts in the various times. The first provably-secure probabilistic public-key encryption scheme was proposed by GoldwasseranMicali, based on the hardness of the quadratic residuosity assumption. Later, some efficient probabilistic encryption schemes appeared including ElGamal, Paillier and various constructions under the random oracle model. In our scheme, we only use the property of probabilism to output a ciphertext that cannot be distinguished from a random number from the view of the adversary. Without loss of generality, we define such a probabilistic encryption *(Enc.k ,Dec.k)* in our scheme where *.k* is the secret key for encryption and decryption. From the point of efficiency, symmetric encryption scheme is quite suitable since encryption and decryption are easy to perform.

The main idea in ABE is that the role of the users is taken by the attributes. Thus, the access structure A will contain the authorized sets of attributes. For CP-ABE, if a user of the system posses an authorized set of attributes then he can decrypt the ciphertext, otherwise, he can't get any information from ciphertext if the set he possed is unauthorized. In our construction, we restrict our attention to monotone access structure.

5. CONCLUSIONS

In this work, we constructed An Enhanced Traceable CP-ABE system which achieved the efficiency and security level as one of the best existing (non-traceable) CP-ABE systems.CPABE systems which include white box traceability of the authorized malicious users have been developed. We can trace the malicious users leaking the partial or modified decryption keys to others for profit. The attribute size is unbounded and the public parameters size does not grow linearly with the number of attributes. The cost of achieving traceability in our system is also very low. In addition, we optimize the system in tracing the malicious users to cut down the storage cost for traceability and to make the system efficient in the revocation of the users. Based on the above advantages, our new systems could be applied to many scenarios such as pay-TV systems and social networks. This system is selectively secure in the standard model, when compared with others.

REFERENCES

[1] A. Shamir, “Identity-based cryptosystems and signature schemes,” in Proc. CRYPTO, 1984, pp. 47–53

[2] V. Goyal, “Reducing trust in the PKG in identity based cryptosystems,” in Advances in Cryptology. Berlin, Germany: Springer-Verlag, 2007, pp. 430–447.

[3] T. Okamoto and K. Takashima, “Fully secure functional encryption with general relation from,,. the decisional linear assumption,” in Advances inCryptology. Berlin, Germany: Springer-Verlag, 2010, pp. 191–208

[4] A. Lewko, T. Okamoto, A. Sahai, K. Takashima, and B. Waters, “Fully secure functional encryption: Attribute-based encryption and (hierarchical) inner product encryption”.

[5] V. Goyal, A. Jain, O. Pandey, and A. Sahai, “Bounded ciphertext policy attribute based ,.encryption,” in Automata, Languages and Programming .Berlin, Germany: Springer-Verlag, 2008, pp. 579–591.

[6] N. Attrapadung, B. Libert, and E. de Panafieu,“Expressive key-policy attribute-based encryption with constant-size ciphertexts,” in Public KeyCryptography. Berlin, Germany: Springer-Verlag, 2011, pp. 90–108.

[7] J. Bethencourt, A. Sahai, and B. Waters, “Ciphertext-policy attribute based ., encryption,” in Proc.and EEE Symp. Secure. Privacy (SP), May 2007, pp. 321–334.

[8] L. Cheung and C. Newport, “Provably secure cipher text policy ABE,”in,Proc. 14th ACM Conf. Comput. Commun.Secur., 2007, pp. 456–465.

[9] A. Lewko and B. Waters, “Unbounded HIBE and attribute-based encryption ,,(HABE)”in Advances in Cryptology. Berlin, Germany: Springer-Verlag, 2011, pp. 547–567.

[10] T. ElGamal, “A public key crypto system and a signature scheme based on discrete logarithms,” and in Advances in Cryptology. Berlin, Germany: Springer-Verlag, 1985, pp. 10–18..

[11] B. Chor, A. Fiat, and M. Naor, “Tracing traitors,” in Proc. CRYPTO, Y. Desmedt, Ed.,1994, vol.839 257–270, ser. Lecture Notes in Computer Science, Springer.

[12] D. Boneh, A. Sahai,and B. Waters,“Fully collusion resistant traitor tracing with short Ciphertexts and Private keys,” in Proc. EUROCRYPT,,. S. Vaudenay, Ed., 2006, vol. 4004, pp. 573– 592, ser. Lecture Notes in Computer Science, Springer.

[13] Y.Rouselakis and B Waters “Practical constructions and new proof methods for large universe attribute based encryption,” in Proc. ACMSIGSAC Conf. Comput. Commun.Secur., 2013, pp 463–474.

[14] D. Boneh and X. Boyen, “Short signatures without random oracles, in Advances in Cryptology (Lecture Notes in Computer Science),, vol.,, 3027, C. Cach in and J. L.Camenisch, Eds and also Berlin ,Germany: Springer-Verlag, 2004, pp. 56–73.

[15] Z. Liu, Z.. Cao, and D. S. Wong, “White-box traceable cipher text- policy attribute- based encryption Supporting any monotone access structures,” IEEE Trans. Inf. Forensics Security, vol,... 8, no. 1, pp. 76–88, Jan. 2013.

[16] A. Sahai, H. Seyalioglu,,,.and B. Waters, “Dynamic credentials and cipher text delegation for attribute based encryption,” in Advances inCryptology. Berlin, Germany: Springer-Verlag, 2012, pp. 199217.

[17] A. Sahai.., and B. Waters, “Fuzzy identity-based encryption,” in Advances in Cryptology. Berlin.., and Germany: Springer-Verlag, 2005, pp. 457–473.

[18] T. Oka moto and K. Takashima, “Homomorphic encryption.., and,. signatures from vector., and their decomposition,” in Pairing-Based Cryptography. Berlin, Germany: Springer-Verlag, 2008, pp. 57– 74.

AUTHORS

AnushaSivanandhan. is currently pursuing M.Tech in Cyber Security in Mar Baselios Institute of Technology and Science, Nellimattom, Kerala, India. She completed her B.Tech from Mar Baselios Institute of Technology and Science, Nellimattom. Her areas of research are Network Security and Information Forensics.

Angel M Eldhose is currenty working as the Assistant Professor in Department of Computer Science and Engineering at mar baselios institute of technology and science, Nellimattom, Kerala, India. He recieved his B-Tech Degree in Computer Science and Engineering from Illahia College of Engineering and Technology and M-Tech in Computer Science and Engineering from KarunyaUniversity.His research interest include Image Processing and Database Security.