Journal of Universal Computer Science, vol. 21, no. 3 (2015), 483-501
submitted: 30/6/14, accepted: 28/2/15, appeared: 1/3/15 © J.UCS

Multi-Authority Attribute-Based Encryption Scheme from Lattices

Guoyan Zhang
(School of Computer Science and Technology, Shandong University
Jinan, China
guoyanzhang@sdu.edu.cn)

Jing Qin
Corresponding author
(School of Mathematics, Shandong University, Jinan, China
qinjing@sdu.edu.cn)

Shams Qazi
(School of Computer Science and Software Engineering (SCSSE)
University of Wollongong, Wollongong, Australia
shams@uow.edu.au)

Abstract: Access control can selectively restrict access to sensitive information stored by third-party sites on the Internet. Attribute-based encryption (ABE) schemes can strengthen the effective combination of flexibility and operability of access control. They allow one sender to encrypt a message for more than one recipient, and to specify who should be able to decrypt, using attributes alone. Since 2005, many powerful ABE schemes have been presented, but there are two types of problem that haven't be efficiently resolved so far. On the one hand, as practical extension of identity-based encryption (IBE) schemes, ABE schemes are also confronted with key escrow problem. On the other hand, attribute set belonging to one user is usually monitored by different authorities in this era of collaboration. Multi-authority ABE (MA-ABE) schemes can simultaneously resolve these problems, but now they have not been thoroughly investigated yet. More precisely, MA-ABE schemes against quantum attack are the main barrier of the development of ABE schemes in a 'post-quantum' world.

In this paper, we firstly present a MA-ABE scheme from lattices, in which identities of users are authenticated by a central authority, which improves the efficiency of authentication. Furthermore, different attribute private keys are still distributed by different authorities, and the central authority cannot obtain any secret information of other attribute authorities, which resolves key escrow problem to some extent. In MA-ABE, attribute private keys belonging to one user are generated by different authorities, and how to ensure correct decryption is one of the crux of schemes. Our scheme gives a simple solution, and each user's attribute private keys are combined using sharing of common public information to automatically realize correct decryption. To our best knowledge, this is the first MA-ABE scheme from lattices, and it is more efficient than the MA-ABE presented by Melissa Chase. Finally, we present a multi-authority large universe ABE scheme, in which the sizes of the public key and the ciphertext are only relative to the number of the attribute authorities, and a user will be able to decrypt a ciphertext if and only if he has at least t_K attributes from each authority K.

Key Words: LWE, MA-ABE, Lattices, Preimage Sampling Functions.
Category: E.3

1 Introduction

Cloud computing paradigm is viewed as a big step to make computation as a public utility, which provides an opportunity for users, companies, and public organizations to reduce costs and increase efficiencies. Information in local computers is now transferred to cloud platforms, then confidentiality and integrity of information are becoming more and more important in cloud computing. Access control is one of the key technologies used for security enforcement of information. An access control model formally specifies how to decide whether an access request should be allowed or repudiated. There were many schemes to consider information security in cloud computing [Vleju, M. 14, Rastogi and Solms 12], but they couldn't enable efficient one-to-many broadcast encryption and fine-grained access control. In order to make access control more flexible, Sahai, A. and Waters, B. introduced the concept of attribute-based encryption (ABE) schemes in 2005 [Sahai and Waters 05], in which a user's keys and ciphertexts were labeled with sets of descriptive attributes and a particular key set can decrypt a particular ciphertext only if there is a match between the attributes of the ciphertext and the user's keys. After that, ABE attracted much attention as a relatively new encryption technology.

There were two variants of ABE schemes proposed: key-policy variant (KP-ABE) [Goyal et al. 06] and ciphertext-policy variant (CP-ABE) [Bethencourt et al. 07]. In the KP-ABE, every ciphertext is associated with a set of attributes, and each user's secret keys are associated with a threshold access structure on attributes. Reversely, in the CP-ABE, attributes are associated with user private keys and access structures with ciphertexts. Many schemes have been presented: Schemes [Goyal et al. 06, Bethencourt et al. 07, Ostrovsky et al. 07, Lewko et al. 10, Goyal et al. 08, Attrapadung et al. 11, Cheung and Newport 07] contributed to make the access structure more expressive. Schemes [Daza et al. 10, Emura et al. 09, Herranz et al. 10] were devoted to get constant-size ciphertexts. Scheme [Xie et al. 13] proposed construction with efficient attribute and user revocation.

Similar to identity-based encryption schemes, the attribute authority is able to compute private key corresponding to any attribute of any user such that he is free to engage in malicious activities without any risk of being confronted in a court of law, which is called key escrow problem. There were many cryptosystems to resolve this problem such as certificateless cryptosystem, certificate-based cryptosystem and multi-authority cryptosystem. Multi-authority cryptosystem has more extensive application, because the cooperation of different departments often makes it possible for multi-authority to authenticate one common user. MA-ABE schemes allow the sender to specify for each authority K a set of attributes monitored by that authority and a number t_K so that the message can be decrypted only by a user who has at least t_K of the given attributes from each authority K. MA-ABE schemes also allow any number of attribute authorities to be corrupted, and guarantee the security of encryption as long as the required attributes cannot be obtained exclusively from those authorities and the trusted authority remains honest. This is an attractive solution and also successfully avoids placing trust in a single entity by making the system operate in a distributed way.

1.1 Related Works

Chase proposed the first MA-ABE scheme with a global identifier which combined users' private keys together to ensure correct decryption [Chase 07], but the scheme relied on a central authority who knew all the secret keys of any attribute authority in order to combine all the attribute private keys belonging to the same user. Compared with the scheme [Chase 07], Muller, S. et al. gave a scheme with a centralized authority that realized any LSSS access structure [Muller et al. 08], but their proof was limited to non-adaptive queries only. The scheme achieved roughly the same functionality as the engineering approach above, except one could still acquire attributes from additional authorities without revisiting the central authority. Chase, M. and Chow, S. gave a scheme without central authority using a distributed pseudo random function [Chase and Chow 09]. However, it only supported an AND policy. Lin, H. et al. [Lin et al. 08] gave a threshold-based scheme that was also somewhat decentralized, in which they must interact during the system setup. Furthermore, the scheme was only secure up to collusion of m users, where m was a system parameter chosen at setup. Lewko, A. and Waters, B. proposed a new MA-ABE scheme [Lewko and Waters 11]. In their scheme, any party could simply act as an authority by creating a public key and issuing private keys to different users, and different authorities needed not even be aware of each other.

All the above schemes were based on traditional number theory hard problems which were proved to have polynomial-time solutions in the environment of quantum computers. In contrast, lattice hardness problems can resist quantum cryptanalysis and have strong worst-case/average-case security guarantees. Furthermore, the mathematical properties of lattices make them both relatively efficient and flexible to enable the construction of powerful cryptosystems. So lattices have recently emerged as a powerful mathematical platform on which to build a rich variety of cryptographic primitives. Since the work [Ajtai and Dwork 97], there were many schemes proposed: one-way functions and collision-resistant hash functions [Ajtai and Dwork 97, Micciancio 02], public-key encryption schemes [Ajtai and Dwork 97, Regev 05, Sahai and Waters 05], identity-based encryption schemes [Gentry et al. 08, Cash et al. 10, Agrawal et al. 10], trapdoor functions [Gentry et al. 08], fully homomorphic encryption schemes [Gentry 10] and attribute-based encryption schemes [Boyen 13, Agrawal et al. 12], but there were no MA-ABE schemes from lattices yet.

1.2 Our Contributions

In this paper, we firstly adopt the method in [Chase 07] to give the first MA-ABE scheme from lattices, and also introduce a multi-authority large universe ABE scheme whose public key and the ciphertext are independently with the size of attribute universe.

- Our first MA-ABE scheme from lattices has many excellent characteristics:
  - Our scheme can resolve key escrow problem: It has also a central authority, but the central authority doesn't know any authorities' secret keys different with scheme [Chase 07], and he cannot be free to engage in malicious activities. He can only authenticate identities of users and combine attributes with users to tie users' private keys together.
  - Setup phase has no private interactivity among attribute authorities: Attribute authorities generate their master public key and secret key pairs respectively, and they also independently present attribute private keys to users without any interactivity. Sharing of public information combines all the attribute private keys of a user to ensure correct decryption.
  - Our scheme keeps the characteristic of ABE: During encryption, the global identifier of user is not be used such that decryptors can correctly recover plaintext only using attributes alone.
  - Our scheme can resist collusion attack of users: By using of preimage sampling functions, the images of different users' attribute private keys cannot be recovered into useful information, and C_A combines the attribute sets with the identities of users by using sharing of public information.
  - Our scheme ensures backward security and forward security: When attributes or attribute authorities need to be updated, attribute authorities can reshare the public information to generate and distribute attribute private keys of users again.
- We also give a MA-IBE scheme with large attribute universe, in which our generation of attribute private keys is a bit different from the above scheme. Each attribute authority can only have one pair of master public key and master secret key, but it can generate different private keys according to different attributes such that the sizes of public key and the ciphertext are proportional to the total number of attribute authorities.

2 Preliminaries and Definitions

2.1 Notation

In the following, we assume that the attribute universe U can be partitioned into d disjoint sets $U_1, U_2, \ldots, U_d$, where d is the number of authorities. Each set will be monitored by a different authority. There is also one trusted central authority C_A. We let U_{GID} denote the attribute set of user with global identifier GID and let U_C denote the attribute set combined with a ciphertext. U^K_{GID} and U^K_C are the attribute sets handled by authority K in the attribute sets of the user GID and the ciphertext C respectively. U^C_{GID} is the intersection of user's attribute set and ciphertext attribute set.

2.2 MA-ABE Scheme

Our definition and security model of MA-ABE schemes are similar to those in [Chase 07], but with significant differences from [Chase 07] that we believe more reasonable. Each attribute authority K is also assigned a tuple (t_K, d_K). A MA-ABE scheme is composed of four algorithms as follows:

Setup. A randomized algorithm run by both central authority C_A and attribute authorities. Taking as input security parameters, it outputs system public key and each attribute authority's master public key and secret key.

Attribute Private Key Generation. A randomized algorithm run by a central authority C_A and attribute authority K, taking as input the attribute authority's secret key, the attribute authority's tuple (t_K, d_K), a user's global identifier GID, and a set of attributes in the authority's domain U^K_{GID}. C_A combines GID with the attribute set, and gives the result to attribute authority K (We will assume that the user's claim of these attributes has been verified by C_A before this algorithm is run). Attribute authority K outputs attribute private keys for the user.

Encryption. A randomized algorithm run by a sender. Taking as input a set of attributes for each authority, a message, and the system public key, it outputs the ciphertext.

Decryption. A deterministic algorithm run by a user with global identifier GID. Taking as input a ciphertext, which was encrypted under attribute set U_C and decryption keys for an attribute set U_K. Output a message m, if $|U^K_{GID} \bigcap U^C_{GID}| \geq t_K$ satisfies for $K = 1, 2, \ldots, d$.

2.3 Security Model of Multi-Authority ABE System

Let λ be the security parameter. Consider the following game:

Setup

- The adversary sends a list of attribute sets $U_C = U_C^1, \ldots, U_C^d$, one for each authority. He must also provide a list of corrupted authorities which cannot include the central authority.
- The challenger generates parameters for the system and sends them to the adversary. These mean the system public key, master public keys for all honest authorities, and secret keys for all corrupt authorities.

Secret Key Queries

The adversary can make as many secret key queries as he wants to attribute authorities. The only requirement is that for each GID, there must be at least one honest authority from which the adversary requests fewer than t_K of the attributes given in U_C^K, i.e. the adversary never requests enough attributes to decrypt the challenge ciphertext.

Challenge

The challenger chooses a message $b \in \{0, 1\}$, computes the ciphertext of b for attribute set U_C, and sends this ciphertext to the adversary.

More Secret Key Queries

The adversary may make more secret key queries subject to the requirements described above.

Guess

The adversary outputs a guess b'. If $b = b'$, the adversary is said to succeed.

A MA-ABE scheme is selective attributes secure if there is a negligible function ε such that, in the above game, any PPT adversary will succeed with probability at most $1/2 + \varepsilon(\lambda)$.

2.4 Concrete Trapdoor Functions with Preimage Sampling [Gentry et al. 08]

Let $q = poly(n)$, $m \geq 5n\,lg\,q$ and $L = m^{1+\varepsilon}$ for any $\varepsilon > 0$. The collection is parameterized by some Gaussian parameter $s \geq L.\omega(\sqrt{log\,m})$, and trapdoor functions with preimage sampling are described as **(TrapGen, SampleDom, SamplePre)**.

- **TrapGen.** The function generator uses the algorithm from Ajt99 algorithm to choose (A, T), where $A \in Z_q^{n\times m}$ is statistically close to uniform and $T \subset \Lambda^{\perp}(A)$ is a good basis with $\|T\| \leq L$. The matrix A (and q) defines the function $f_A(.)$, and the good basis T is its trapdoor.
- **SampleDom.** The function $f_A(.)$ is defined as $f_A(e) = Ae\,mod\,q$, with domain $D_n = \{e \in Z^m : \|e\| \leq s\sqrt{m}\}$ and range $R_n = Z_q^n$. The input distribution is $D_{Z^m,\,s}$, which can be sampled using discrete Gaussian probability distribution scheme with the standard basis for Z^m.

– **SamplePre.** The trapdoor inversion algorithm SampleISIS(A, T, s, u) samples from $f_A^{-1}(u)$ as follows: first, choose via linear algebra an arbitrary $t \in Z^m$ such that $At = u \, mod \, q$(such a t exists for all but an at most q^{-n} fraction of A). Then sample v using Sample$D(T, s, -t)$, and output $e = t+v$.

2.5 Simple and Efficient "Hash-and-Sign" Digital Signature Schemes [Gentry et al. 08]

The scheme is built upon a collection of trapdoor functions with Preimage Sampling given by **(TrapGen, SampleDom, SamplePre)**, and operates relative to a function $H = \{H_n : \{0, 1\} \rightarrow R_n\}$ that is modeled as a random oracle (recall that D_n and R_n are the domain and range, respectively, of the collection for security parameter n).

SigKeyGen (1^n). Let $(a, t) \leftarrow TrapGen(1^n)$, where a describes a function f_a and t is its trapdoor. The verification key is a and the signing key is t.

Sign (t, m)**.** If (m, σ_m) is in local storage, output σ_m. Else, let $\sigma_m \leftarrow SamplePre(t, H(m))$, store (m, σ_m), and output σ_m.

Verify (a, m, σ**).** If $\sigma \in D_n$ and $f_a(\sigma) = H(m)$, accept. Else, reject.

Proposition. The scheme described above is SUF-CMA-secure in the random oracle model.

2.6 Learning With Errors

The LWE (learning with errors) problem was first defined by Regev, O. [Regev 05], and has been extensively studied and used. We use the decisional version of the LWE problem.

Definition 2.1. A prime q, a positive integer n, and a distribution χ over Z_q are all public. A (Z_q, n, χ)-problem instance consists of access to an unspecified challenge oracle O, being either a noisy pseudo-random sampler O_s carrying some constant random secret key s or a truly random sampler O'_s, whose behaviors are respectively as follows:

O_s: outputs noisy pseudo-random samples of the form $(\omega_i, \upsilon_i) = (\omega_i, \omega_i^T s + x_i) \in Z_q^n \times Z_q$, where $s \in Z_q^n$, is a uniformly distributed persistent secret key that is invariant across invocations, $x_i \in Z_q^n$ is a freshly generated ephemeral additive noise component with distribution χ and $\omega_i \in Z_q^n$ is a fresh uniformly distributed vector revealed as part of the output.

O'_s: outputs truly random samples $(\omega_i, \upsilon_i) \in Z_q^n \times Z_q$, drawn independently uniformly at random in the entire domain $Z_q^n \times Z_q$.

The (Z_q, n, χ)-problem statement allows an unspecified number of queries to be made to the challenge oracle O, with no stated prior bound. An algorithm A decides the (Z_q, n, χ)-LWE problem, if $|Pr[A^{O_s} = 1] - Pr[A^{O'_s} = 1]|$ is non-negligible for a random $s \in Z_q^n$.

Definition 2.2. The interactive LWE_q problem is described as follows [Gentry et al. 08]: On input a matrix $A \in Z_q^{n\times m}$, a vector $p \in Z_q^m$, a hash function $H : (0, 1)^* \to Z_q^n$, a value z, and access to an oracle, returning a sample from $f_A^{-1}(H(z))$ (the same value is returned for repeated queries on the same z). The goal is to distinguish whether p is either a LWE instance or uniform, i.e., between the case that $p = A^T s + x$ for some $s \leftarrow Z_q^n$, $x \leftarrow \chi^m$, and the case that $p \leftarrow Z_q^m$ is uniform. When H is modeled as a random oracle, the interactive LWE problem is hard as long as the standard LWE problem is hard.

3 MA-ABE Scheme from Lattices

In the previous MA-IBE schemes, each authority must authenticate identities of users which greatly reduces the efficiency. In our scheme, there is a central authority C_A which is responsible to authenticate users' identities and to combine attribute sets with identities of users, which efficiently prevents the collusion of different users. The central authority C_A must always honestly combine attribute sets with identities of users. Because central authority C_A does not know all secret keys of other authorities, he cannot generate all attribute private keys on behalf of other attribute authorities. Simultaneously, we adopt the strategy in [Chase 07] to require that each user has a unique global identifier (GID), and a user must present his GID to central authority C_A in order to receive a coherent set of keys. However, encryption need not the unique global identifier, and the ability to decrypt is the same with traditional ABE scheme independent of the GID.

3.1 Concrete Protocol

Assuming there are d attribute authorities, and each authority K can authenticate d_K attributes. C_A is a central authority that any authority could be. Let λ be a security parameter, $q = q(\lambda)$, $p = p(\lambda)$ be two primes, $n = n(\lambda)$, $m = m(\lambda)$ be two positive integers, $\sigma = \sigma(\lambda)$, $\alpha = \alpha(\lambda)$ be two positive Gaussian parameters. t_K is the number that a user can only decrypt if he has at least t_K of the given attributes from each authority K. $H : (0, 1)^* \to Z_q$ is a hash function. Let $[d_K] \subseteq \{1, 2, \ldots, d_K\}$ denote users' attribute set that authority K monitors, and $|[d_K]| \geq t_K$. U denotes attribute universe. For simplicity, we can take the first $|U|$ elements of Z_p to be the universe. Namely, integers $1, 2, \ldots, |U|$.

SetUp. Given a security parameter λ and n, m, σ, α, q as inputs, each authority K runs Ajt99's lattice trapdoor generation algorithm d_K times to get $mpk_K^i = A_K^i \in Z_q^{n\times m}$, $msk_K^i = T_K^i \in \Lambda_q^{\perp}$, $(i = 1, 2, \ldots, d_K)$, as master public key and master secret key of authority K. Each authority randomly chooses a vector $u^K \in Z_q^n$, and C_A computes random vector $u \in Z_q^n = u^1 + u^2 + \ldots + u^K (mod\, q) = (u_1, u_2, \ldots, u_n)$. C_A chooses a family of $(d-1)$-degree polynomial sets $F = \{f_i = (f_i^1, f_i^2, \ldots, f_i^n) : \{0, 1\}^l \to Z_q^n\}$ such that for $i = 1, 2, \ldots, d$, $f_i^1(0) = u_1$, $f_i^2(0) = u_2, \ldots, f_i^n(0) = u_n$. Define system public key as $MPK = mpk_1^1, mpk_1^2, \ldots, mpk_1^{d_1}, \ldots, mpk_d^1, mpk_d^2, \ldots, mpk_d^{d_d}, u, F$.

Attribute Private Key Generation. Given user's identity GID, the central authority C_A computes $j = H(GID)$ and uses it as index to choose $(d-1)$-degree polynomial set $f_j = (f_j^1, f_j^2, \ldots, f_j^n)$ from the family of $(d-1)$-degree polynomial set F. C_A computers d sharing of u to d authorities. More precisely, he sets $(u_{r,1}, u_{r,2}, \ldots, u_{r,d}) = (f_j^r(1), f_j^r(2), \ldots, f_j^r(d))$ as sharing of u_r for $r = 1, 2, \ldots, n$. Each authority K gets his share $u'_K = (u_{1,K}, u_{2,K}, \ldots, u_{n,K})$ and divides it into d_K sharing $u_{K_1}, u_{K_2}, \ldots, u_{K_{d_K}}$ by using (t_K, d_K) Shamir secret sharing scheme on every coordinate of u'_K, $K = 1, \ldots, d$, and he runs the algorithm SamplePre to find e_{K_i} such that $A_K^i e_{K_i} = u_{K_i}$, $i \in [d_K]$ and sends e_{K_i}, $i \in [d_K]$ to user with identity GID. The user's private key is $SK_{GID} = (e_{K_i})_{i\in[d_K]}$, $K = 1, \ldots, d$.

Encrypt. Given system public key MPK, the attribute set $U_C^1, \ldots, U_C^d$ and a message $b \in \{0, 1\}$:

- Let $D = (d!l!)^2$, where $l = \max_{K=1}^{d} d_K$, choose a uniformly random $s \leftarrow Z_q^n$, a noise term $x \leftarrow \chi_{\alpha,q}$ and $x_{K,j} \leftarrow \chi_{\alpha,q}^m$, $j \in U_C^K$, $K = 1, 2, \ldots, d$.
- Compute
$$c \leftarrow u^T s + Dx + b\lfloor q/2 \rfloor \in Z_q,$$
$$c_{K,j} \leftarrow (A_K^j)^T s + Dx_{K,j} \in Z_q^m,\ j \in U_C^K,\ K = 1, 2, \ldots, d.$$
- Output ciphertext $CT = (c, \{c_{K,j}\}, j \in U_C^K, K = 1, 2, \ldots, d)$.

Decrypt. Given system public key MPK, the user's private key SK_{GID}, and a ciphertext CT. If $\forall K$, $|U_{GID}^K \bigcap U_{GID}^C| \geq t_K$, then the entity does:

- Computes Lagrangian coefficients $L_{K,j}$, L_K so that
$$\sum_{j\in[d_K]} L_{K,j} A_K^j e_{K_j} = u'_K\,(mod\,q),\ \sum_{K=1}^{d} L_K u'_K = u\,(mod\,q).$$
- Computes
$$b' \leftarrow c - \sum_{K=1}^{d} L_K \sum_{j\in[d_K]} L_{K,j} e_{K_j}^T c_{K,j}\,(mod\,q).$$
- Outputs 0 if b' is closer to 0 than to $\lfloor q/2 \rfloor\, mod\, q$, otherwise outputs 1.

3.2 Correctness and Parameter Declaration

Firstly, we note that

$$c - \sum_{K=1}^{d} L_K \sum_{j\in[d_K]} L_{K,j}(e_{K_j})^T c_{K,j}\,(mod\,q)$$

$$= u^T s + b\lfloor q/2\rfloor + Dx - \sum_{K=1}^{d} L_K \sum_{j\in[d_K]} L_{K,j}((e_{K_j})^T (A_K^j)^T s + Dx_{K,j})\,mod\,q$$

$$= u^T s + b\lfloor q/2\rfloor + Dx - \sum_{K=1}^{d} L_K (\sum_{j\in[d_K]} (L_{K,j} A_K^j e_{K_j})^T s + L_{K,j}(e_{K_j})^T Dx_{K,j})mod\,q$$

$$= u^T s + b\lfloor q/2\rfloor + Dx - \sum_{K=1}^{d} L_K (\sum_{j\in[d_K]} (L_{K,j} u_{K_j})^T s + \sum_{j\in[d_K]} L_{K,j}(e_{K_j})^T Dx_{K,j})mod\,q$$

$$= u^T s + b\lfloor q/2\rfloor + Dx - \sum_{K=1}^{d} (L_K u'_K)^T s - \sum_{K=1}^{d} L_K \sum_{j\in[d_K]} L_{K,j}(e_{K_j})^T Dx_{K,j}\,mod\,q$$

$$= u^T s + b\lfloor q/2\rfloor + Dx - u^T s - \sum_{K=1}^{d} L_K \sum_{j\in[d_K]} L_{K,j}(e_{K_j})^T Dx_{K,j}\,mod\,q$$

$$= b\lfloor q/2\rfloor + (Dx - \sum_{K=1}^{d} L_K \sum_{j\in[d_K]} L_{K,j}(e_{K_j})^T Dx_{K,j}\,mod\,q.$$

In order to ensure the correctness, we should require that

$$|Dx - \sum_{K=1}^{d} L_K \sum_{j\in[d_K]} L_{K,j}(e_{K_j})^T Dx_{K,j}| \le q/4.$$

Because $D = (d!l!)^2$, where $l = \max_{K=1}^{d}\{d_K\}$,

$$\sum_{K=1}^{d} L_K \sum_{j\in[d_K]} L_{K,j}(e_{K_j})^T Dx_{K,j} = \sum_{K=1}^{d} (d!)^2 L_K \sum_{j\in[d_K]} ((l!)^2 L_{K,j}(e_{K_j})^T x_{K,j}).$$

Especially, $L_K(d!)^2$ and $L_{K,j}(l!)^2$ are all integers.
Furthermore,

$$|Dx - \sum_{K=1}^{d} L_K \sum_{j\in[d_K]} L_{K,j}(e_{K_j})^T Dx_{K,j}|$$

$$\leq D|x| + |\sum_{K=1}^{d} L_K \sum_{j\in[d_K]} L_{K,j}(e_{K_j})^T Dx_{K,j}|.$$

$$\leq D|x| + |(d!)^2(l!)^2 \sum_{K=1}^{d} \sum_{j\in[d_K]} x_{K,j}|.$$

So we can pick the noise vectors appropriately so that

$$|Dx - \sum_{K=1}^{d} L_K \sum_{j\in[d_K]} L_{K,j}(e_{K_j})^T Dx_{K,j}| \leq q/4.$$

3.3 System Update

Here, we discuss system update caused by the update of attributes and attribute authorities. In our scheme, each authority needn't to regenerate his master public key and secret key during system update, and they only distribute part of attribute private keys again.

- If there is an attribute withdrawn from attribute universe U, and we assume that attribute authority κ who monitors this attribute. Authority κ needn't to reset his setup phase, and he can delete the public key corresponding to this attribute and share his u'_κ to $(u_{\kappa,1}, u_{\kappa,2}, \ldots, u_{\kappa,d_\kappa-1})$ again. Then he generates the other attribute private keys according to the new sharing and distributes them. If there are many attributes withdrawn from attribute universe, the similar proceeding can also do. When there are new attributes added to attribute universe U, attribute authorities who monitor these attributes also need to renew their sharing and to distribute the attribute private keys as the above step after C_A verifies the validation of these attributes. Renewing the sharing makes the scheme ensure both backward security and forward security.
- The number of authorities in the system can be changed because of the joining or withdrawing of authorities: it is possible to allow the central authority to add additional attribute authorities to the system at any point or to cancel the right of attribute authorities. C_A can run system update only by computing new sharing of u, and the rest of attribute authorities also needn't renew their public keys and private keys. They only renew the sharing and generate new attribute private keys to realize update.

3.4 Security Analysis

In our system, each authority chooses his own public key and secret key pairs respectively, so even if the adversary corrupts almost all of the authorities, he cannot obtain any information about other authorities without being corrupted. In the following security analysis, we adopt a weaker model in which the adversary can get any attribute private keys, even if the challenge attributes, but we don't present him the master secret keys of the corrupted authorities. We prove its security under an "interactive" version of the of LWE hardness assumption in the presence of a signing oracle for the (stateful) "Hash-and-Sign" Digital Signature Schemes in random oracle model. A similar "interactive" assumption about the hardness of "interactive quadratic residuosity assumption" was used for the IBE [Gentry et al. 08, Boneh et al. 07].

Theorem 1. Let A be a PPT adversary with advantage $\varepsilon > 0$ against the following selective attributes secure game for the MA-ABE scheme. If A can query any polynomial signature oracles, assuming that s_i, $(i = 1, 2, \ldots, d)$ is the size of the challenge attribute set to the ith authority, then there is a PPT algorithm B that decides the LWE problem with advantage $\varepsilon/(\sum_{i=1}^{d} s_i + 1)$.

Proof. Suppose A is a polynomial-time adversary, and he can succeed against the proposed scheme in the following selective attributes secure game with advantage ε, then we can construct an algorithm B to resolve the decisional version of LWE problem with advantage $\varepsilon/(\sum_{i=1}^{d} s_i + 1)$ by using A as a sub-routine algorithm.

Setup

- The adversary A sends a list of challenge attribute sets $U_C = U_C^1, \ldots, U_C^d$, one for each authority, assuming $|U_C^i| = s_i$. Without loss of generality, we assume that the attributes in U_C^i, $(i = 1, \ldots, d)$ are the first s_i attributes monitored by authority i.
- B requests a sampling oracle O provided by LWE problem instance, and receives $m \sum_{i=1}^{d} s_i + 1$ LWE samples that we denote as

$$\{(W_1, V_1)\}, \{(W_1^1, V_1^1), (W_1^2, V_1^2), \ldots, (W_1^m, V_1^m)\}, \{(W_2^1, V_2^1), (W_2^2, V_2^2), \ldots, (W_2^m, V_2^m)\}, \ldots, \{(W^1_{\sum_{i=1}^{d} s_i}, V^1_{\sum_{i=1}^{d} s_i}), (W^2_{\sum_{i=1}^{d} s_i}, V^2_{\sum_{i=1}^{d} s_i}), \ldots, (W^m_{\sum_{i=1}^{d} s_i}, V^m_{\sum_{i=1}^{d} s_i})\}.$$

B chooses hash function $H_1 : (0, 1)^* \to Z_q$ and $H_2 : (0, 1)^* \to Z_q^n$, and he constructs system public key PP as follows:

– The $\sum_{i=1}^{d} s_i$ matrices (A_j^i, $j = 1, 2, \ldots, s_i$, $i = 1, 2, \ldots, d$) are chosen from

$$\{W^1_{\sum_{l=1}^{i} s_l+j}, W^2_{\sum_{l=1}^{i} s_l+j}, \ldots, W^m_{\sum_{l=1}^{i} s_l+j}\}$$

of the LWE challenge

$$\{(W^1_{\sum_{l=1}^{i} s_l+j}, V^1_{\sum_{l=1}^{i} s_l+j}), (W^2_{\sum_{l=1}^{i} s_l+j}, V^2_{\sum_{l=1}^{i} s_l+j}), \ldots, (W^m_{\sum_{l=1}^{i} s_l+j}, V^m_{\sum_{l=1}^{i} s_l+j})\}.$$

– The other matrices A_j^i, $j = s_i + 1, \ldots, d_i$ are chosen using Ajt99's lattice trapdoor generation scheme with a trapdoor T_j^i.

– The vector u is constructed from the LWE challenge, $u = W_1$. The system public key is returned to the adversary A.

Secret Key Queries

B answers each secret key query for attribute set $U_q = U_q^1, \ldots, U_q^d$ and global identifier GID as follows:

For GID, B computes index $l = H_1(GID)$ to choose $(d-1)$-degree polynomial $f_l = (f_{l,1}, f_{l,2}, \ldots, f_{l,n})$ such that $f_l(0) = (f_{l,1}(0), f_{l,2}(0), \ldots, f_{l,n}(0)) = u = (u_1, u_2, \ldots, u_d)$, and obtains the d sharing $u_1', u_2', \ldots, u_d'$ of u, and $u_i' = (f_{l,1}(i), f_{l,2}(i), \ldots, f_{l,n}(i))$. There are two cases about secret key queries:

Case 1. Set $[d]$ denotes the set of attribute authorities whose attributes satisfy $|U_q^i \bigcap U_C^i| = |I_i| \geq t_i$

– Let $U_q^i \bigcap U_C^i = I_i$, $|I_i| \geq t_i$. Then, note that B has trapdoors for the matrices corresponding to the set $U_q^i - I_i$.

– Choose $\{U_{i,j}, j = 1, 2, \ldots, t_i - 1\} \subset I_i$. Pick $e_{i,j}$ randomly using algorithm SampleGaussian. Set

$$A_j^i e_{i,j} = u_{i,j},\ j = 1, 2, \ldots, t_i - 1,\ i \in [d] \subseteq \{1, 2, \ldots, d\}.$$

– Represent the sharing of u_i' symbolically as $u_{i,j} = u_i' + a_{i,1} j + a_{i,2} j^2 + a_{i,t_i-1} j^{t_i-1}$, $i = 1, \ldots, d$. Where, $a_{i,1}, a_{i,2}, \ldots, a_{i,t_i-1}$, $i = 1, \ldots, d$ are vector variables (each is of length n). $u_{i,1}, u_{i,2}, \ldots, u_{i,t_i-1}, u_i'$ commonly determine the values for $a_{i,1}, a_{i,2}, \ldots, a_{i,t_i-1}$, $i \in [d]$, which determine all sharing $u_{i,1}, u_{i,2}, \ldots, u_{i,|U_q^i|}$, $i = 1, \ldots, d$.

– For $j = t_i, t_i + 1, \ldots, |I_i|$, the trapdoors cannot be known. B defines $H_2(g_{i,j}) = u_{i,j}$, $i \in [d]$. Query the signature oracle for $g_{i,j}$ ($i \in [d]$) according to the different signature oracle machine and returned by $e_{i,j} = f^{-1}_{A_j^i}(H_2(g_{i,j}))$.

- For $j = |I_i| + 1, |I_i| + 2, \ldots, |U_q^i|$, B can invoke SamplePre(A_j^i, T_j^i, $u_{i,j}$, $i \in [d]$) to get $e_{i,j}$ satisfying $A_j^i e_{i,j} = u_{i,j}$, $i \in [d]$.
- Return $e_{i,j}$, $j = 1, 2, \ldots, |U_q^i|$, $i \in [d]$.

Case 2. Let $[d]$ denote the set of attribute authorities whose attributes satisfy $|U_q^i \bigcap U_C^i| = |I_i| < t_i$.

- Let $U_q^i \bigcap U_C^i = I_i$, $|I_i| < t_i$. Then, note that B has trapdoors for the matrices corresponding to the set $U_q^i - I_i$.
- For $U_{i,j} \in I_i$, pick $e_{i,j}$ randomly using algorithm SampleGaussian. Set $A_j^i e_{i,j} = u_{i,j}$, $j = 1, \ldots, |I_i|$, $i \in [d]$.
- Represent the sharing of u_i' symbolically as $u_{i,j} = u_i' + a_{i,1}j + a_{i,2}j^2 + a_{i,t_i-1}j^{t_i-1}$, $i \in [d]$. Where, $a_{i,1}, a_{i,2}, \ldots, a_{i,t_i-1}$, $i \in [d]$ are vector variables and each is of length n.
- Since $|I_i| < t_i$, and there are $t_i - 1$ variables $a_{i,1}, a_{i,2}, \ldots, a_{i,t_i-1}$, $i \in [d]$ by choosing $t_i - 1 - |I_i|$ sharing $u_{i,s_i+2}, u_{i,s_i+3}, \ldots, u_{i,t_i}$ randomly, the values for $a_{i,1}, a_{i,2}, \ldots, a_{i,t_i-1}$, $i \in [d]$ are determined. This determines all sharing $u_{i,1}, u_{i,2}, \ldots, u_{i,|U_i^q|}$, $i \in [d]$. For $j = |I_i| + 1, |I_i| + 2, \ldots, |U_i^q|$, and B can invoke SamplePre(A_j^i, T_j^i, $u_{i,j}$, $i \in [d]$) to get $e_{i,j}$ satisfying $A_j^i e_{i,j} = u_{i,j}$, $i \in [d]$.
- Return $e_{i,j}$, $j = 1, 2, \ldots, |U_i^q|$, $i \in [d]$.

Remark. The adversary can make as many secret key queries as he wants to the honest attribute authorities. The only requirements are that for each GID, there must be at least one honest authority K from which the adversary requests fewer than t_K of the attributes given in U_C^K, i.e. the adversary never requests enough attributes to decrypt the challenge ciphertext.

Challenge

B chooses a bit b, computes the encryption of b for attribute set U_C as follows:

- Computes $D = (d!l!)^2$, where $l = \max_{K=1}^{d}\{d_K\}$.
- Let

$$c_0 \leftarrow DV_1 + b\lfloor q/2 \rfloor$$

  and

$$c_i = (DV_i^1, DV_i^2, \ldots . DV_i^m),\ i = 1, 2, \ldots, \sum_{i=1}^{d} s_i.$$

– Outputs ciphertext $CT = (c_0, \{c_i\}, i = 1, 2, \ldots, \sum_{i=1}^{d} s_i)$, and sends this ciphertext to the adversary.

More Secret Key Queries

The adversary may make more secret key queries subjecting to the requirements described above.

Guess

When the adversary A outputs a guess b^*, the simulator B uses that guess to determine an answer on the LWE oracle: Output "genuine", if $b = b^*$, else output "random".

4 MA-ABE Scheme from Lattices with Constant Size

In the above scheme, the size of system public keys is proportional to the total number of attributes in the system, and the size of the ciphertext is proportional to the total number of attributes in the ciphertext. All these cause low efficiency when there is a large attribute universe. In this section, we give a multi-authority large universe ABE scheme. In the following scheme, the sizes of the public key and the ciphertext are only relative to the number of the attribute authorities. A user will be able to decrypt a ciphertext if and only if he has at least t_K of the attributes from each authority K.

4.1 Concrete Protocol

Assuming there are d authorities, and each authority K can authenticate d_K attributes. There is a central authority C_A that any authority can act as. Let λ be a security parameter. Let $q = q(\lambda)$, $p = p(\lambda)$ be two primes, $n = n(\lambda)$, $m = m(\lambda)$ be two positive integers, and $\sigma = \sigma(\lambda)$, $\alpha = \alpha(\lambda)$ be two positive Gaussian parameters. t_K is the number that a user can only decrypt if he has at least t_K of the given attributes from each authority K. Let $[d_K] \subseteq \{1, 2, \ldots, d_K\}$, and $|[d_K]| \geq t_K$. Define the universe U of attributes. For simplicity, we can take the first $|U|$ elements of Z_p, to be the universe. Namely, the integers $1, 2, \ldots, |U|$.

SetUp. Given a security parameter λ and n, m, σ, α, q as input, each authority K runs Ajt99's lattice trapdoor generation algorithm to get $mpk_K = A_K \in Z_q^{n\times m}$, $msk_K = T_K \in \Lambda_q^{\perp}(A_K)$ as master public key and master secret key of authority K. Each authority randomly chooses a vector $u^K \in Z_q^n$, and C_A computes random vector $u \in Z_q^n = u^1 + u^2 + \ldots + u^K (mod\, q) = (u_1, u_2, \ldots, u_n)$. C_A chooses a family of $(d-1)$-degree polynomial sets $F = \{f_i = (f_i^1, f_i^2, \ldots, f_i^n) : \{0,1\}^l \to Z_q^n\}$ such that $\forall i$, $f_i^1(0) = u_1$, $f_i^2(0) = u_2, \ldots, f_i^n(0) = u_n$. Define the system master public key as $MPK = mpk_1, mpk_2, \ldots, mpk_d, u, F$.

Attribute Key Generation. Given a global identifier GID and attribute set $U_{GID} = U_{GID}^1, U_{GID}^2 \ldots U_{GID}^d$, the central authority C_A computes $j = H(GID)$ and uses it as index to choose $(d-1)$-degree polynomial f_j from the family of $(d-1)$-degree polynomials $F = \{f_n : \{0,1\}^l \rightarrow Z_q^n\}$. C_A computers d sharing of u to d authorities. Namely, he sets $(u_{r,1}, u_{r,2}, \ldots, u_{r,d}) = (f_j^r(1), f_j^r(2), \ldots, f_j^r(d))$ as sharing of u_r for $r = 1, 2, \ldots, n$. Each authority K gets his sharing $u'_K = (u_{1,K}, u_{2,K}, \ldots, u_{n,K})$ and divides it into d_K sharing $(u_{K_1}, u_{K_2}, \ldots, u_{K_{d_K}})$ by using (t_K, d_K) Shamir secret sharing scheme on every coordinate of u'_K, $K = 1, \ldots, d$. Each authority K runs the algorithm SamplePre, and finds e_{K_i} such that $A_K e_{K_i} = u_{K_i}$, $i \in U_{GID}^K$, $\forall K$, and sends e_{K_i}, $i \in U_{GID}^K$, $K = 1, \ldots, d$, to user with global identifier GID. The private key is $SK_{ID} = (e_{K_i})_{i \in U_{GID}^K}$, $K = 1, \ldots, d$.

Encrypt. Given system public key MPK and a message $b \in \{0, 1\}$:

- Let $D = (d!l!)^2$, where $l = \max_{K=1}^{d} d_K$, choose a uniformly random $s \leftarrow Z_q^n$, a noise term $x \leftarrow \chi_{\alpha,q}$ and $x_i \leftarrow \chi_{\alpha,q}^m$, $i = 1, \ldots, d$.
- Compute

  $$c_0 \leftarrow u^T s + Dx + b\lfloor q/2 \rfloor \in Z_q,\ c_i \leftarrow A_i^T s + Dx_i \in Z_q^m (mod\, q),\ i = 1, \ldots, d.$$

- Output ciphertext $CT = (c_0, \{c_i\}, i = 1, \ldots, d)$.

Decrypt. Given system public key MPK, the private key SK_{GID}, and a ciphertext CT. If $|U_{GID}^K| \geq t_K$, $\forall K$, then the entity does:

- Computes Lagrangian coefficients $L_{i,j}$, L_i so that

  $$\sum_{j \in [d_K]} L_{i,j} A_i e_{i_j} = u'_i\, (mod\, q),\ \sum_{i=1}^{d} L_i u'_i = u\, (mod\, q).$$

- Computes

  $$b' \leftarrow c_0 - \sum_{i=1}^{d} L_i \sum_{j \in [d_K]} L_{i,j} e_{i_j}^T c_i\, (mod\, q).$$

- Outputs 0 if b' is closer to 0 than to $\lfloor q/2 \rfloor\, (mod\, q)$, otherwise outputs 1.

4.2 Security Analysis

Theorem 2. Assuming A is a PPT adversary with non-negligible advantage $\varepsilon > 0$ that succeeds against the large universe MA-ABE scheme in the following selective attributes secure game. If A can query any polynomial signature oracles, assuming that s_i, $(i = 1, 2, \ldots, d)$ is the size of the challenge attribute set to the ith authority, then there exists a PPT algorithm B that decides the LWE problem with advantage $\varepsilon/(\sum_{i=1}^{d} s_i + 1)$.

Proof. It is similar to the proof of scheme in Section 3, and thus we omit it here.

5 Conclusions and Future Work

In this paper, we present the first MA-ABE scheme from lattices. Similar to Chase's scheme, there is a central authority in our scheme, but the central authority in our scheme cannot generate any attribute private key, which really avoids the key escrow problem without adding any burden. Furthermore, different users' attribute private keys cannot be combined to give correct decryption, which avoids the collusion threat of different users. Finally, we give a MA-ABE scheme under large universe of attributes, in which the sizes of the public key and the ciphertext are only relative to the number of the attribute authorities. MA-ABE schemes from lattices without central authority are more difficult to design, and this is our future work direction.

Acknowledgement

We wish to thank the reviewers for their accurate suggestions.

This work is supported by the National Natural Science Foundation of China (No. 61173139, 61272091, 61103237), Key Project of National Natural Science Foundation of Shandong Province under Grant (No. ZR2011FZ005), and Shandong Natural Science Foundation (No. ZR2012FQ028, ZR2012FM005). This work is also supported by China Scholarship Council.

References

[Agrawal et al. 10] Agrawal, S., Boneh, D. and Boyen, X.: "Efficient lattice (H)IBE in the standard model"; In Advances in Cryptology-EUROCRYPT 2010, LNCS 6110, 2010, pages 553-572.

[Agrawal et al. 12] Agrawal, S., Boyen, X., Vaikuntanathan, V., Voulgaris, P. and Wee, H.: "Functional Encryption for Threshold Functions (or Fuzzy IBE) from Lattices"; In Public Key Cryptography 2012, LNCS 7293, 2012, pages 280-297.

[Ajtai and Dwork 97] Ajtai. M. and Dwork, C.: "A public-key cryptosystem with worst-case/average-case equivalence"; In STOC, 1997, pages 284-293.

[Attrapadung et al. 11] Attrapadung, N., Libert, B. and Panafieu, E.: "Expressive Key-Policy Attribute-Based Encryption with Constant-Size Ciphertexts"; In PKC 2011, LNCS 6571, 2011, pages 90-108.

[Bethencourt et al. 07] Bethencourt, J., Sahai, A. and Waters, B.: "Ciphertext-policy attribute-based encryption"; In Proceedings of the 28th IEEE Symposium on Security and Privacy (Oakland), 2007, pages 321-334.

[Boneh et al. 07] Boneh, D., Gentry, C. and Hamburg, M.: "Space-efficient identity based encryption without pairings"; In FOCS2007, 2007, pages 647-657. Full version at http://eprint.iacr.org/2007/177.

[Boyen 13] Boyen, X.: "Attribute-Based Functional Encryption on Lattices"; In Theory of Cryptography Conference (TCC2013), LNCS 7785, 2013. pages 122-142.

[Brakerski and Vaikuntanathan 11] Brakerski, Z. and Vaikuntanathan, V.: "Fully homomorphic encryption from ring-LWE and security for key dependent messages"; In CRYPTO 2011, LNCS 6841, 2011, pages 505-524.

[Cash et al. 10] Cash, D., Hofheinz, D., Kiltz, E. and Peikert, C.: "Bonsai trees or, how to delegate a lattice basis"; In EUROCRYPT 2010, LNCS 6110, 2010, pages 523-552.

[Chase 07] Chase, M.: "Multi-authority Attribute Based Encryption"; In Vadhan, S., editor, Theory of Cryptography-TCC 2007, LNCS 4392, 2007, pages 515-534.

[Chase and Chow 09] Chase, M. and Chow, S.: "Improving privacy and security in multi-authority attribute-based encryption"; In ACM Conference on Computer and Communications Security, 2009, pages 121-130.

[Cheung and Newport 07] Cheung, L. and Newport, C.: "Provably secure ciphertext policy abe"; In ACM Conference on Computer and Communications Security, 2007, pages 456-465.

[Daza et al. 10] Daza, V., Herranz, J., Morillo, P., Rafols, C.: "Extensions of access structures and their cryptographic applications"; AAECC (2010) 21, 2010, pages 257-284.

[Emura et al. 09] Emura, K., Miyaji, A., Nomura, A., Omote, K., Soshi, M.: "A ciphertextpolicy attribute-based encryption scheme with constant ciphertext length"; In: Bao, F., Li, H., Wang, G. (eds.) ISPEC 2009, LNCS 5451, 2009, pages 13-23.

[Gentry et al. 08] Gentry, C., Peikert, C. and Vaikuntanathan, V.: "Trapdoors for hard lattices and new cryptographic constructions"; In STOC2008, 2008, pages 197-206.

[Gentry 10] Gentry, C.: "Fully homomorphic encryption using ideal lattices"; PHD thesis, Stanford University, 2009.

[Goyal et al. 06] Goyal, V., Sahai, A. and Waters, B.: "Attribute-based encryption for fine-grained access control of encrypted data"; In Proceedings of the 13th ACM conference on Computer and Communications Security (CCS'06), 2006, pages 89-98.

[Goyal et al. 08] Goyal, V., Jain, A., Pandey, O. and Sahai, A.: "Bounded ciphertext policy attribute-based encryption"; In ICALP2008, LNCS 5126, 2008, pages 579-591.

[Herranz et al. 10] Herranz, J., Laguillaumie, F., Rafols, C.: "Constant-Size Ciphertexts in Threshold Attribute-Based Encryption"; In PKC2010, LNCS 6056, 2010, pages 19-34.

[Lewko et al. 10] Lewko, A., Sahai, A., Waters, B.: "Revocation Systems with Very Small Private Keys"; In Proceedings of the 28th IEEE Symposium on Security and Privacy (Oakland), 2010, pages 273-285.

[Lewko and Waters 11] Lewko, A., Waters, B.: "Decentralizing Attribute-Based Encryption"; EUROCRYPT 2011, LNCS 6632, 2011, pages 568-588.

[Lin et al. 08] Lin, H., Cao, Z., Liang, X. and Shao, J.: "Secure threshold multi-authority attribute based encryption without a central authority"; In INDOCRYPT, LNCS 5365, 2008, pages 426-436.

[Micciancio 02] Micciancio, M.: "Generalized compact knapsacks, cyclic lattices, and efficient one-way functions from worst-case complexity assumptions"; In FOCS2002, 2002, pages 356-365.

[Muller et al. 08] Muller, S., Katzenbeisser, S. and Eckert, C.: "Distributed attribute-based encryption"; In ICISC2008, LNCS 5461, 2008, pages 20-36.

[Ostrovsky et al. 07] Ostrovsky. R, Sahai, A. and Waters, B.: "Attribute-based encryption with non-monotonic access structures"; In ACM Conference on Computer and Communications Security, 2007, pages 195-203.

[Rastogi and Solms 12] Rastogi, R. and Solms, R.: "Information Security Service-Culture Information Security for End-users"; In Journal of Universal Computer Science, Vol. 18(12), 2012, pages 1628-1642.

[Regev 05] Regev, O.: "On lattices, learning with errors, random linear codes, and cryptography"; In STOC 2005, 2005, pages 84-93.

[Sahai and Waters 05] Sahai, A and Waters, B.: "Fuzzy identity-based encryption"; In EUROCRYPT2005, LNCS 3494, 2005, pages 457-473.

[Vleju, M. 14] Vleju, M.: "Automatic Authentication to Cloud-Based Services"; In Journal of Universal Computer Science, Vol. 20(3), 2014, pages 385-405.

[Xie et al. 13] Xie, X., Ma, H., Li, J. and Chen, X.: "An Efficient Ciphertext-Policy Attribute-Based Access Control towards Revocation in Cloud Computing"; In Journal of Universal Computer Science, Vol. 19(16), 2013, pages 2349-2367.