This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPDS.2015.2392752, IEEE Transactions on Parallel and Distributed Systems

Circuit Ciphertext-policy Attribute-based Hybrid Encryption with Verifiable Delegation in Cloud Computing

Jie Xu, Qiaoyan Wen, Wenmin Li and Zhengping Jin

Abstract—In the cloud, for achieving access control and keeping data confidential, the data owners could adopt attribute-based encryption to encrypt the stored data. Users with limited computing power are however more likely to delegate the mask of the decryption task to the cloud servers to reduce the computing cost. As a result, attribute-based encryption with delegation emerges. Still, there are caveats and questions remaining in the previous relevant works. For instance, during the delegation, the cloud servers could tamper or replace the delegated ciphertext and respond a forged computing result with malicious intent. They may also cheat the eligible users by responding them that they are ineligible for the purpose of cost saving. Furthermore, during the encryption, the access policies may not be flexible enough as well.

Since policy for general circuits enables to achieve the strongest form of access control, a construction for realizing circuit ciphertext-policy attribute-based hybrid encryption with verifiable delegation has been considered in our work. In such a system, combined with verifiable computation and encrypt-then-mac mechanism, the data confidentiality, the fine-grained access control and the correctness of the delegated computing results are well guaranteed at the same time. Besides, our scheme achieves security against chosen-plaintext attacks under the k-multilinear Decisional Diffie-Hellman assumption. Moreover, an extensive simulation campaign confirms the feasibility and efficiency of the proposed solution.

Index Terms—Ciphertext-policy attribute-based encryption, Circuits, Verifiable delegation, Multilinear map, Hybrid encryption.

1 Introduction

The emergence of cloud computing brings a revolutionary innovation to the management of the data resources. Within this computing environments, the cloud servers can offer various data services, such as remote data storage [1] and outsourced delegation computation [2], [3], etc. For data storage, the servers store a large amount of shared data, which could be accessed by authorized users. For delegation computation, the servers could be used to handle and calculate numerous data according to the user's demands. As applications move to cloud computing platforms, ciphertext-policy attribute-based encryption (CP-ABE) [4], [5] and verifiable delegation (VD) [6], [7] are used to ensure the data confidentiality and the verifiability of delegation on dishonest cloud servers.

Taking medical data sharing as an example (see Fig. 1), with the increasing volumes of medical images and medical records, the healthcare organizations put a large amount of data in the cloud for reducing data storage costs and supporting medical cooperation. Since the cloud server may not be credible, the file cryptographic storage is an effective method to prevent private data from being stolen or tampered. In the meantime, they may need to share data with the person who satisfies some requirements. The requirements, i.e, access policy, could be {Medical Association Membership ∧ (Attending Doctor ∨ Chief Doctor) ∧ Orthopedics}. To make such data sharing be achievable, attribute-based encryption is applicable.

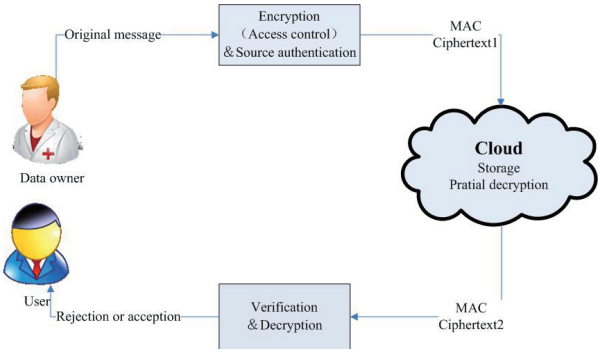

Fig. 1. Medical data sharing system

There are two complementary forms of attribute-based encryption. One is key-policy attribute-based encryption (KP-ABE) [8], [9], [10], and the other is ciphertext-policy attribute-based encryption (CP-ABE). In a KP-ABE system, the decision of access policy is made by the key distributor instead of the encipherer, which limits the practicability and usability for the system in practical applications. On

• *J. Xu, Q. Wen, W. Li and Z. Jin are with the State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing, 100876, China. E-mail: liwenmin02@gmail.com*

1045-9219 (c) 2015 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPDS.2015.2392752, IEEE Transactions on Parallel and Distributed Systems

the contrary, in a CP-ABE system, each ciphertext is associated with an access structure, and each private key is labeled with a set of descriptive attributes. A user is able to decrypt a ciphertext if the key's attribute set satisfies the access structure associated with a ciphertext. Apparently, this system is conceptually closer to traditional access control methods. On the other hand, in a ABE system, the access policy for general circuits could be regarded as the strongest form of the policy expression that circuits can express any program of fixed running time.

Delegation computing is another main service provided by the cloud servers. In the above scenario, the healthcare organizations store data files in the cloud by using CP-ABE under certain access policies. The users, who want to access the data files, choose not to handle the complex process of decryption locally due to limited resources. Instead, they are most likely to outsource part of the decryption process to the cloud server. While the untrusted cloud servers who can translate the original ciphertext into a simple one could learn nothing about the plaintext from the delegation.

The work of delegation is promising but inevitably suffers from two problems. a) The cloud server might tamper or replace the data owner's original ciphertext for malicious attacks, and then respond a false transformed ciphertext. b) The cloud server might cheat the authorized user for cost saving. Though the servers could not respond a correct transformed ciphertext to an unauthorized user, he could cheat an authorized one that he/she is not eligible.

Further, during the deployments of the storage and delegation services, the main requirements of this research are presented as follows.

1) *Confidentiality* (indistinguishability under selective chosen plaintext attacks (IND-CPA)). With the storage service provided by the cloud server, the outsourced data should not be leaked even if malware or hackers infiltrate the server. Besides, the unauthorized users without enough attributes to satisfy the access policy could not access the plaintext of the data. Furthermore, the unauthorized access from the untrusted server who obtains an extra transformation key should be prevented.

2) *Verifiability*. During the delegation computing, a user could validate whether the cloud server responds a correct transformed ciphertext to help him/her decrypt the ciphertext immediately and correctly. Namely, the cloud server could not respond a false transformed ciphertext or cheat the authorized user that he/she is unauthorized.

Thus, in this paper, we will attempt to refine the definition of CP-ABE with verifiable delegation in the cloud to consider the data confidentiality, the fine-grained data access control and the verifiability of the delegation. The related security definition and IND-CPA security game used in the proof are presented in section 3.2 to depict the above attacks of the adversaries.

1.1 Related Work

Attribute-based encryption. Sahai and Waters [11] proposed the notion of attribute-based encryption (ABE). In subsequent works [8], [12], they focused on policies across multiple authorities and the issue of what expressions they could achieve. Up until recently, Sahai and Waters [9] raised a construction for realizing KP-ABE for general circuits. Prior to this method, the strongest form of expression is boolean formulas in ABE systems, which is still a far cry from being able to express access control in the form of any program or circuit. Actually, there still remain two problems. The first one is their have no construction for realizing CP-ABE for general circuits, which is conceptually closer to traditional access control. The other is related to the efficiency, since the exiting circuit ABE scheme is just a bit encryption one. Thus, it is apparently still remains a pivotal open problem to design an efficient circuit CP-ABE scheme.

Hybrid encryption. Cramer and Shoup [13], [14] proposed the generic KEM/DEM construction for hybrid encryption which can encrypt messages of arbitrary length. Based on their ingenious work, a one-time MAC were combined with symmetric encryption to develop the KEM/DEM model for hybrid encryption [15], [16], [17]. Such improved model has the advantage of achieving higher security requirements.

ABE with Verifiable Delegation. Since the introduction of ABE, there have been advances in multiple directions. The application of outsourcing computation [18], [19] is one of an important direction. Green *et al*. [2] designed the first ABE with outsourced decryption scheme to reduce the computation cost during decryption. After that, Lai *et al*. [3] proposed the definition of ABE with verifiable outsourced decryption. They seek to guarantee the correctness of the original ciphertext by using a commitment. However, since the data owner generates a commitment without any secret value about his identity, the untrusted server can then forge a commitment for a message he chooses. Thus the ciphertext relating to the message is at risk of being tampered. Further more, just modify the commitments for the ciphertext relating to the message is not enough. The cloud server can deceive the user with proper permissions by responding the terminator $\perp$ to cheat that he/she is not allowed to access to the data.

1.2 Our Contribution

Prompted by the requirements in the cloud, we modify the model of CP-ABE with verifiable delegation and present a concrete construction to realize circuits ciphertext-policy based hybrid encryption with verifiable delegation (VD-CPABE).

1045-9219 (c) 2015 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPDS.2015.2392752, IEEE Transactions on Parallel and Distributed Systems

To keep data private and achieve fine grain access control, our starting point is a circuit key-policy attribute-based encryption proposed by Sahai and Waters [9]. We give the anti-collusion circuit CP-ABE construction in this paper for the reason that CP-ABE is conceptually closer to the traditional access control methods. For the main efficiency drawbacks of ABE, previous constructions provided an agile method to outsource the most overhead of decryption to the cloud. However, there is no guarantee that the calculated result returned by the cloud is always correct. The cloud server may forge ciphertext or cheat the eligible user that he even does not have permissions to decryption. To validate the correctness, we extend the CP-ABE ciphertext into the attribute-based ciphertext for two complementary policies and add a MAC for each ciphertext, so that whether the user has permissions he/she could obtain a privately verified key to verify the correctness of the delegation and prevent from counterfeiting of the ciphertext. Aiming at further improving the efficiency and providing intuitive description of the security proof, the conception of hybrid encryption is also introduced in this work. Besides, security of the VD-CPABE system ensures that the untrusted cloud will not be able to learn anything about the encrypted message and forge the original ciphertext. After that, the proposed scheme is simulated in the GMP library [20]. Finally, the scheme is concluded to be practical in the cloud.

1.3 Our Techniques

Verifiable delegation (VD) is used to protect authorized users from being deceived during the delegation. The data owner encrypts his message M under access policy f, then computes the complement circuit $\bar{f}$, which outputs the opposite bit of the output of f, and encrypts a random element R of the same length to M under the policy $\bar{f}$. The users can then outsource their complex access control policy decision and part process of decryption to the cloud. Such extended encryption ensures that the users can obtain either the message M or the random element R, which avoids the scenario when the cloud server deceives the users that they are not satisfied to the access policy, however, they meet the access policy actually.

In CP-ABE we use a hybrid variant for two reasons: one is that the circuit ABE is a bit encryption, and the other is that the authentication of the delegated ciphertext should be guaranteed. The ciphertext of the hybrid VD-CPABE system is divided into two components: the CP-ABE for circuits f and $\bar{f}$ makes up the key encapsulation mechanism (KEM) [21] part, and a symmetric encryption plus the encrypt-then-mac mechanism [22] make up the authenticated encryption mechanism (AE) part. Each KEM encrypts a random group element and then maps it via key derivation functions into a symmetric encryption key dk and a one-time verified key vk. Then the random encryption key dk is used to encrypt the message of any length. vk and the data owner's ID are used to verify the MAC of the ciphertext. Only when the server dose not forge the original ciphertext and respond a correct partial decrypted ciphertext, the user could be able to properly validate the MAC.

For implementation, the recent work on multilinear maps over the integers [23] is applied to simulate the scheme in the GMP library in VC 6.0. Though the operation time for the pairing in the multilinear map is much more than the one in the bilinear map, we could achieve the strongest general circuits access policy up to now. Besides, by using verifiable delegation, the operation time for the user is short and independent of the complexity of the circuit. For the security, we show that the IND-CPA secure KEM combines with the IND-CCA secure authenticated (symmetric) encryption scheme yields our IND-CPA secure hybrid VD-CPABE scheme.

1.4 Organization

In the following section, we describe some related mathematical problems. A formal definition of hybrid VD-CPABE and its corresponding security model is given in section 3. In Section 4, we propose a concrete construction for VD-CPABE. In Section 5, we analyze the security of the proposed scheme. Subsequently, we present a brief performance analysis. Finally, the conclusions are given in Section 7.

2 Preliminary

In this section, we summarize the concepts about the system, the circuits and the multi-linear decisional Diffie-Hellman assumption.

2.1 Notation

In the rest of the paper, we let Z_p be a finite field with prime order p. $\perp$ is a formal symbol denotes termination. If X is a finite set then $x \leftarrow X$ denotes that x is randomly selected from X. If A is an algorithm then $A(x) \rightarrow y$ denotes that y is the output by running the algorithm A on input x. We denote $\mathcal{G}(\lambda, k)$ as a group generation algorithm where λ is the security parameter and k is the number of allowed pairing operation. As usual, a function ε: $Z_p \rightarrow R$ is $negligible$ if for every $c > 0$ there is a K such that $\varepsilon(k) < k^{-c}$ for all $k > K$.

2.2 system description and assumption

As shown in TABLE 1, the parties in the VD-CPABE construction are firstly summarized.

In the system, the data owner and the users are both registered entities and got private keys from the authority. The authority is supposed to be the only party that is fully trusted by all participants.

1045-9219 (c) 2015 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPDS.2015.2392752, IEEE Transactions on Parallel and Distributed Systems

TABLE 1
Role description

Role	Description
Authority	Attribute key generator center (trusted third party)
Data owner	Encrypting party who uploads his encrypted data to the cloud
User	Decrypting party who outsources the most overhead computation to the cloud
Cloud server	The party who provides storage and outsourced computation services

Similar to the previous schemes [3], [18], the server is supposed to be untrusted. Sound trust management standards as well as auditing standards could be used to establish good business relations between the cloud server and the user. According to this frame, the cloud server could be regarded as a trustworthy cloud service provider. Actually, the role-based access control is proposed based on this assumption. However, using this single mechanism, we will be at the risks of unknown attacks and the existing of the malicious system administrator, which may result in data leakage, invalidation of access control and failure of outsourcing. Besides, trust management mechanism may cause an extra workload for the auditor. Thus, it is high time to construct a practical cryptography scheme to protect data and control access with an untrusted server.

2.3 Circuits

In the context, we still restrict our attention to the monotone boolean circuit with a single output gate [9]. The definition of a circuit and its evaluation are as follows.

Definition 1. A single-output circuit is a 5-tuple $f = (n, q, A, B, G)$. Here n is the number of inputs, q is the number of gates, and $n + q$ is the number of wires. Let Inputs$= \{1, ..., n\}$, Wires$= \{1, ..., n + q\}$, Gates$= \{n + 1, ..., n + q\}$ and OutputWire$= \{n + q\}$. Then A: Gates$\rightarrow$ Wires/OutputWires is a function to identify each gate's first incoming wire, B: Gates$\rightarrow$ Wires/OutputWires is a function to identify each gate's second incoming wire and G: Gates$\rightarrow \{AND, OR\}$ is a function to identify a gate as either an AND or OR gate. Gates have two inputs, arbitrary functionality and a single fan-out. Every non-input wire is the outgoing wire of some gates. We require $A(w) < B(w) < w$ for all $w \in$ Gates. Let $depth(w)$ equals to the length of the shortest path to an input wire plus 1 and if $w \in$ Inputs then $depth(w) = 1$.

We define the evaluation of the circuit f as $f(x)$ on input the string $x \in \{0, 1\}^n$, and let $f_w(x)$ be the value of wire w on input x. Given the monotone boolean circuit f we can compute its complement circuit $\bar{f}$, which outputs the opposite bit of the output of f. For the circuit $\bar{f}$, negation gates will remain only at the input level by applying De Morgan's rule. We will ignore the depth of the negation gates. See Fig.2 for an illustration of a circuit f and the corresponding complement circuit $\bar{f}$.

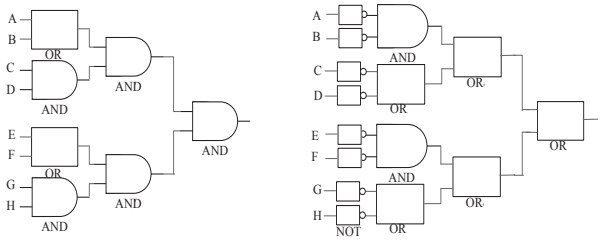

Fig. 2. **Left**: A conventional circuit $f = (A \vee B) \wedge (C \wedge D) \wedge (E \vee F) \wedge (G \vee H)$. **Right**: A complement circuit corresponding to the left circuit $\bar{f} = (\overline{A} \wedge \overline{B}) \vee (\overline{C} \vee \overline{D}) \vee (\overline{E} \wedge \overline{F}) \vee (\overline{G} \wedge \overline{H})$

2.4 Multilinear Map

Definition 2. (Multilinear map [9], [24]). It runs $\mathcal{G}(\lambda, k)$ and outputs k cyclic groups $\vec{G} = (G_1, ..., G_k)$ of the same prime order p. Let the elements $\{g_i \in G_i\}_{i=1,...,k}$ be the generators of the above groups and set $g = g_1$. Then their exist a set of bilinear maps $\{e_{ij} : G_i \times G_j \rightarrow G_{i+j} | i, j \geq 1, j + j \leq k\}$ (write as e for simple) that has the following properties.

For $a, b \leftarrow Z_p$, we have $e(g_i^a, g_j^b) = g_{i+j}^{ab}$.

Definition 3. (k- Multilinear Decision-Diffie-Hellman problem). A challenger runs $\mathcal{G}(\lambda, k)$ to get a sequence of groups $\vec{G} = (G_1, ..., G_k)$ of prime order p where each comes with a canonical generator $g = g_1, g_2..., g_k$. Then it picks $s, c, c_1, ..., c_k \leftarrow Z_p$. The advantage in distinguishing the tuple $(g, g^s, g^{c_1}, ..., g^{c_k}, g_k^{s \prod_{j \in [1,k]} c_j})$ from $(g, g^s, g^{c_1}, ..., g^{c_k}, g_k^c)$ is negligible in λ.

3 Our Model of Hybrid VD-CPABE

In this section, we present the definition and security model of our hybrid VD-CPABE. In such a system, a circuit ciphertext-policy attribute-based encryption scheme, a symmetric encryption scheme and an encrypt-then-mac mechanism are applied to ensure the confidentiality, the fine-grained access control and the verifiable delegation.

3.1 Hybrid VD-CPABE

Definition 4. A hybrid VD-CPABE scheme is defined by a tuple of algorithms (Setup, Hybrid-Encrypt, KeyGen, Transform, Verify-Decrypt). The description of each algorithm is as follows.

- **Setup**(λ, n, l). Executed by the authority, this algorithm takes as input a security parameter λ, the number of attributes n and the maximum depth l of a circuit. It outputs the public parameters PK and a master key MK which is kept secret.

1045-9219 (c) 2015 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPDS.2015.2392752, IEEE Transactions on Parallel and Distributed Systems

- **Hybrid-Encrypt**(PK, M, f). This algorithm is executed by the data owner. It could be conveniently divided into two parts: key encapsulation mechanism (KEM) and authenticated symmetric encryption (AE).
  - The KEM algorithm takes as input the public parameters PK and an access structure f for circuit. It computes the complement circuit $\bar{f}$ and chooses a random string R. Then it generates $K_M = \{dk_m, vk_m\}$, $K_R = \{dk_r, vk_r\}$ and the CP-ABE ciphertext (CK_M, CK_R).
  - The AE algorithm takes as input a message M, the random string R, the symmetric key K_M and K_R. Then it outputs the ciphertext $(C_M, C_R, \sigma_{ID,vk_m}(C_M||C_R), \sigma_{ID,vk_r}(C_M||C_R))$.

  The total ciphertext for our VD-CPABE scheme is the tuple $CT = (CK_M, CK_R, C_M, C_R, \sigma_{ID,vk_m}(C_M||C_R), \sigma_{ID,vk_r}(C_M||C_R))$.
- **KeyGen**$(MK, x \in \{0,1\}^n)$. The authority generates private keys for the users. This algorithm takes as input the master key MK and a bit string x. It outputs a private key SK and a transformation key TK.
- **Transform**(TK, CT). Executed by the cloud servers, this algorithm takes as input the transformation key TK and a ciphertext CT that was encrypted under f and $\bar{f}$. It outputs the partially decrypted ciphertext $CT' = (CK'_M, C_M, C_R, \sigma_{ID,vk_m}(C_M||C_R))$ or $CT' = (CK'_R, C_M, C_R, \sigma_{ID,vk_r}(C_M||C_R))$.
- **Verify-Decrypt**(SK, CT'). Executed by the users, this algorithm takes as inputs the secret key SK and the partially decrypted ciphertext CT'. Firstly, it verifies the validity of σ. Then it outputs the message M_b, which satisfies that if $f(x) = 1$ then $M_b = M$ and if $f(x) = 0$ then $M_b = R$.

3.2 Security Model

Since we use key encapsulation mechanism (KEM) and authenticated encryption (AE) to build our hybrid VD-CPABE scheme, we describe the security definition separately at first.

The confidentiality property (indistinguishability of encryptions under selective chosen plaintext attacks (IND-CPA)) required for KEM is captured by the following games against adversary $\mathcal{A}$.

Game.KEM

- **Init**. The adversary gives a challenge access structure f^*, where it wishes to be challenged.
- **Setup**. The simulator runs the Setup algorithm and gives the public parameters PK to the adversary.
- **KeyGen Queries I**. The adversary makes repeated private key queries corresponding to the sets of attributes $x_1, ..., x_{q_1}$. We require that $\forall i \in q_1$ we have $f^*(x_i) = 0$.
- **Encrypt**. The simulator encrypts K_0 under the structure f^*, random chooses K_1 from key space and flips a random coin b. Then the simulator sends K_b and the ciphertext CK^* to the adversary.
- **KeyGen Queries II**. The adversary makes repeated private key queries corresponding to the sets of attributes $x_{q_1}, ..., x_q$ where $f^*(x) = 0$.
- **Guess**. The adversary outputs a guess b' of b.

We define the advantage of an adversary $\mathcal{A}$ in this game is $Pr[b' = b] - \frac{1}{2}$. Then a KEM scheme is secure against selective chosen plaintext attacks if the advantage is negligible.

The confidentiality property (indistinguishability of encryptions under selective chosen ciphertext attacks (IND-CCA)) required for AE is captured by the following games against adversary $\mathcal{A}$.

Game.AE

- **Init**. The adversary submits two equal length messages M_0 and M_1.
- **Setup**. The simulator runs the Setup algorithm and generators the symmetric key K_{AE}.
- **Encrypt**. The simulator flips a random coin b, encrypts M_b under the symmetric key K_{AE}, generates the ciphertext C^* and gives it to the adversary.
- **Decrypt Queries**. The adversary makes repeated decryption queries. When the given ciphertext $C \neq C^*$, the simulator will return $D_{K_{AE}}(C)$ and $\sigma_{K_{AE}}(C)$ to the adversary.
- **Guess**. The adversary outputs a guess b' of b.

Let $Pr[b' = b] - \frac{1}{2}$ be the advantage of an adversary $\mathcal{A}$ in this game. Using the encrypt-then-mac method, We say that an AE scheme is IND-CCA secure if the advantage is negligible[21].

From the above, we present the security model for our scheme as follows.

Game.VD-CPABE

- **Init**. The VD-CPABE algorithm adversary submits the challenge access structure f^* and two equal length messages M_0 and M_1.
- **Setup**. The simulator runs the Setup algorithm and gives the public parameters PK to the adversary.
- **KeyGen Queries I**. The adversary makes repeated private key queries corresponding to the sets of attributes $x_1, ..., x_{q_1}$. We require that $\forall i \in q_1$ we have $f^*(x_i) = 0$.
- **Encrypt**. The simulator encrypts K_0 under the structure f^* by using the KEM algorithm. Then the simulator flips a random coin v and encrypts M_v under the symmetric key K_0 by using the AE algorithm. Then the total ciphertext is given to the VD-CPABE algorithm adversary.
- **KeyGen Queries II**. The adversary makes repeated private key queries corresponding to the sets of attributes $x_{q_1}, ..., x_q$ where $f^*(x) = 0$.
- **Guess**. The adversary outputs a guess v' of v.

We define the advantage of an adversary $\mathcal{A}$ in this

1045-9219 (c) 2015 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPDS.2015.2392752, IEEE Transactions on Parallel and Distributed Systems

game is $Pr[v' = v] - \frac{1}{2}$.

We'll show that if a KEM scheme is IND-CPA secure and an AE scheme is IND-CCA secure then our hybrid encryption scheme is IND-CPA secure in section 5.

4 Our Hybrid VD-CPABE Scheme

In this section, we propose a concrete circuit ciphertext-policy attribute-based hybrid encryption with verifiable delegation scheme based on the multilinear maps and the verifiable computing technology under cloud environment.

We give a brief description of the protocol in Fig.3. Authority generates private keys for the data owner and user. The data owner encrypts his data using hybrid encryption system, generates a privately verifiable MAC for each symmetric ciphertext and then uploads the whole ciphertext to the cloud server. Then the data owner could be offline. The user, who wants to access to the data, interacts with the cloud server. In the figure, the dashed arrows indicate that the value is transferred secretly, while the solid arrows indicate that the value is transferred without a secure channel.

Using general circuits to express the access control policy, we construct a monotone circuit with depth l and input size to be n. The proposed hybrid VD-CPABE scheme consists of the following probabilistic polynomial time (PPT) algorithms.

- **Setup**(λ, n, l). This algorithm is executed by the authority. It takes as input a security parameter λ, the number n of input size and the maximum depth l of a circuit. Then it runs $\mathcal{G}(\lambda, k = l + 1)$, outputs a sequence of groups $\vec{G} = (G_1, ..., G_k)$ of prime order p and their corresponding generators $g_1, ..., g_k$ and sets $g = g_1$. After that it chooses three one-way hash functions $H_1 : G_k \to \{0,1\}^m$, $H_2 : G_k \to Z_p$, $H_3 : \{0,1\}^* \to G_1$, random $\alpha \in Z_p$, $a \in Z_p$, $h_{11}, ..., h_{1n}, h_{21}, ..., h_{2n} \in G_1$ and sets $y = g^a$. The public key PK as well as the system master key MK are as follows:
  $PK = (g_k^\alpha, H_1, H_2, H_3, y, h_1, ..., h_n, h_{n+1}, ..., h_{2n})$, $MK = g^\alpha$.
- **Hybrid-Encrypt**$(PK, f = (n, q, A, B, GateType), M \in \{0,1\}^m))$. This algorithm is executed by the data owner. Taking the public parameters PK, a description f of a circuit and a message $M \in \{0,1\}^m$ as input, the hybrid encryption algorithm works as follows.
  1) It chooses random $R \in \{0,1\}^m$, $s_1, s_2, s_3 \in Z_p$ and computes
  $C'_M = g_{k-1}^{s_1}$, $r_1 = H_2(g_k^{\alpha s_1})$, $C_M = M \oplus H_1(g_k^{\alpha s_1})$, $C'_R = g_{k-1}^{s_2}$, $r_2 = H_2(g_k^{\alpha s_2})$, $C_R = R \oplus H_1(g_k^{\alpha s_2})$, $\sigma_1 = MAC.Sign_{ID_o, r_1}(C_M||C_R)$, $\sigma_2 = MAC.Sign_{ID_o, r_2}(C_M||C_R)$.
  Where
  $\sigma_1 = g^{\alpha s_3} y^{t s_3} H_3^{t s_3}(ID_0) H_3^{r_1 s_3}(ID_0||C_M||C_R)$ and
  $\sigma_2 = g^{\alpha s_3} y^{t s_3} H_3^{t s_3}(ID_0) H_3^{r_2 s_3}(ID_0||C_M||C_R)$.
  Set
  $\sigma_M = \{\sigma_1, g_k^{\alpha s_3}, g_{k-1}^{t s_3}, H_{3,k-1}^{s_3}(ID_o||C_M||C_R)\}$ and
  $\sigma_R = \{\sigma_2, g_k^{\alpha s_3}, g_{k-1}^{t s_3}, H_{3,k-1}^{s_3}(ID_o||C_M||C_R)\}$.
  The partial ciphertext is $(C_M, C'_M, \sigma_M, C_R, C'_R, \sigma_R)$.
  Note that the value $g^\alpha y^t$, g^t and $H_3^t(ID_o)$ are the private keys for the encrypter shown in the KeyGen algorithm.
  2) Given the circuit access structure f, it generates a complement circuit $\bar{f}$ using De Morgan's rule such that negation gates appear only at the input wires.
  Takes f for example, the encryption algorithm chooses random $r_1, ..., r_{n+q-1} \in Z_p$ and lets $r_{n+q} = s_1$. The randomness r_w is associated with wire w. We then describe how the circuit f shares the encryption exponent s_1. We use the monotone boolean circuits given by Garg *et al.* [9]. The structure of the shares depends on if w is an Input wire, an OR gate, or an AND gate. The circuit descriptions are as follows.
  - **Input wire**. For $w \in [1, n]$, this algorithm chooses random $z_w \in Z_p$. The shares are: $C_{w,1} = y^{r_w}(y h_w)^{-z_w}$, $C_{w,2} = g^{z_w}$.
  - **Gate OR**. Let $j = depth(w)$. This algorithm choose random $a_w \in Z_p$. The shares are: $C_{w,1} = g^{a_w}$, $C_{w,2} = g_j^{a(r_w - a_w r_{A(w)})}$, $C_{w,3} = g_j^{a(r_w - a_w r_{B(w)})}$.
  - **Gate AND**. Let $j = depth(w)$. This algorithm choose random $a_w, b_w \in Z_p$. The shares are: $C_{w,1} = g^{a_w}$, $C_{w,2} = g_j^{a(r_w - a_w r_{A(w)} - b_w r_{B(w)})}$.

  For the OR and AND gates in circuit $\bar{f}$, the sharing methods are as the same as in f. When negation gates appear in the input level, setting $f_w(x) = \bar{x_w}$, the shares of the corresponding input wire w will be:
  $C_{w,1} = y^{r_w} h_{n+w}^{-z_w}$, $C_{w,2} = g^{z_w}$.
  Then we could utilize the circuit $\bar{f}$ to share the encryption exponent s_2.

  The full ciphertext CT contains $C_M, C'_M, C_R, C'_R, \sigma$, the ciphertext of f and $\bar{f}$ (C'_M, C'_R, the ciphertext of f and $\bar{f}$ are considered as the KEM part denoted by (CK_M, CK_R). (C_M, C_R, σ) is considered as the AE part). In summary, the total ciphertext for our VD-CPABE scheme is the tuple
  $CT = (CK_M, CK_R, C_M, C_R, \sigma_M, \sigma_R)$.
- **KeyGen**$(MK, x \in \{0,1\}^n)$ The authority generates the private key for the user. Then the user sends his transformation key to the cloud server. This algorithm takes as input the master secret key and a description of the attribute $x \in \{0,1\}^n$. It firstly chooses a random $t \in Z_p$. Then it creates

1045-9219 (c) 2015 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPDS.2015.2392752, IEEE Transactions on Parallel and Distributed Systems

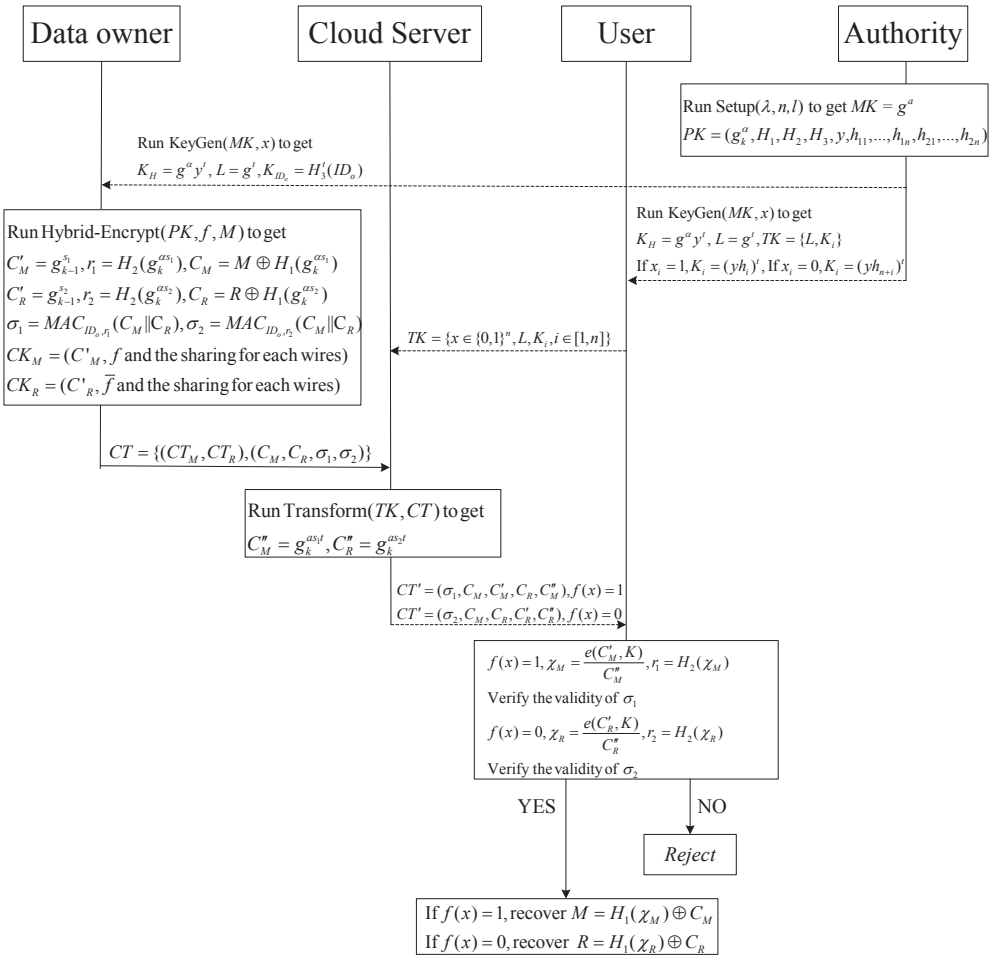

Fig. 3. Our hybrid VD-CPABE scheme in the cloud

the private key as
$K_H = g^{\alpha} y^t$, $L = g^t$, if $x_i = 1$ $K_i = (yh_i)^t$, if $x_i = 0$ $K_i = (yh_{n+i})^t$, $i \in [1, n]$.
The transformation key is $TK = \{L, K_i, i \in [1, n]\}$. Note that, for the data owner ID_o, the authority generates his private key with the identity attribute ID_0 as
$K_H = g^{\alpha} y^t$, $L = g^t$, $K_{ID_o} = H_3^t(ID_o)$.

- **Transform**(TK, CT). The transformation algorithm is executed by the cloud server. It takes as input the transformation key TK and the original ciphertext CT. The algorithm partially decrypts the ciphertext as follows.
  Taking TK with x as input, we evaluate the circuit from the bottom up. If $f(x) = 1$ we will be able to partially decrypt the ciphertext for M and if $f(x) = 0$ we will be able to partially decrypt the ciphertext for R. Consider the wire w at depth j, if $f_w(x) = 1$ then the algorithm computes $E_w = (g_{j+1})^{ar_w t}$ and if $f_w(x) = 0$ the algorithm dose nothing. The evaluation depends on if w is an Input wire, an OR gate, or an AND gate. The partial decryption algorithm is as follows.
  - **Input wire**. For $w \in [1, n]$, if $x_w = f_w(x) = 1$, the algorithm computes:
    $E_w = e(K_w, C_{w,1}) \cdot e(L, C_{w,1}) = e(y^t h_w^t, g^{Z_w}) \cdot e(g^t, g^{ar_w} y^{-z_w} h_w^{-z_w}) = g_2^{ar_w t}$.
    When negation gates appears at the input level, $f_w(x) = \bar{x}_w$. If $f_w(x) = 1$, the algorithm computes:
    $E_w = e(K_w, g^{Z_w}) \cdot e(L, C_{w,1}) = e(y^t h_{n+w}^t, g^{Z_w}) \cdot e(g^t, g^{ar_w} y^{-z_w} h_{n+w}^{-z_w}) = g_2^{ar_w t}$.
  - **Gate OR**. Let $j = depth(w)$. If $f_{A(w)(x)} = 1$, the algorithm computes:
    $E_w = e(E_{A(w)}, C_{w,1}) \cdot e(C_{w,2}, L) = e(g_j^{ar_{A(w)} t}, g^{a_w}) \cdot e(g_j^{a(r_w - a_w r_{A(w)})}, g^t) = g_{j+1}^{ar_w t}$.
  - **Gate AND**. Let $j = depth(w)$. If $f_{A(w)(x)} = f_{A(w)(x)} = 1$, the algorithm computes:
    $E_w = e(E_{A(w)}, C_{w,1}) \cdot e(E_{B(w)}, C_{w,2}) \cdot e(C_{w,3}, L) = e(g_j^{ar_{A(w)} t}, g^{a_w}) \cdot e(g_j^{ar_{B(w)} t}, g^{b_w}) \cdot e(g_j^{a(r_w - a_w r_{A(w)} - b_w r_{B(w)})}, g^t) = g_{j+1}^{ar_w t}$.
    If $f(x) = f_{n+q} = 1$, the algorithm computes $C''_M = (g_k)^{as_1 t}$, otherwise, if $f(x) = 0$ then $\bar{f} = 1$, the algorithm computes $C''_R = (g_k)^{as_2 t}$. It finally outputs the partially decrypted ciphertext

1045-9219 (c) 2015 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPDS.2015.2392752, IEEE Transactions on Parallel and Distributed Systems

$CT' = (\sigma_M, C_M, C_R, C'_M, C''_M)$ if $f(x) = 1$
and $CT' = (\sigma_R, C_M, C_R, C'_R, C''_R)$ if $f(x) = 0$.

- **Verify-Decrypt**(SK, CT'). The verifying and decryption algorithm is executed by the user. Given the partially decrypted ciphertext CT' which contains a signature σ and the data owner's identity ID_o, the user does as follows.
  1) If $f(x) = 1$, the user will compute $\chi_M = \frac{e(C'_M, K)}{C''_M}$, $r_1 = H_2(\chi_M)$ and use the signature using ID_o and verified key g^{r_1} to check whether
  $e(\sigma_1, g_{k-1}) = g_k^{\alpha s_3} \cdot e(yH_3(ID_o), g_{k-1}^{ts_3}) \cdot e(H_{3,k-1}^{s_3}(ID_o||C_M||C_R), g^{r_1})$.
  Then the user will compute $M = H_1(\chi_M) \oplus C_M$.
  2) If $f(x) = 0$, the user will compute $\chi_R = \frac{e(C'_R, K)}{C''_R}$, $r_2 = H_2(\chi_R)$ and verifier the signature using ID_o and g^{r_2}, then the user will compute $R = H_1(\chi_R) \oplus C_R$.

5 Security Proof

In our proposed hybrid VD-CPABE scheme, the AE part is implemented by a one-time symmetric-key encryption and the encrypt-then-mac paradigm. (C, σ) is considered as the IND-CCA secure AE part [8]. The following theorem shows that the KEM part is IND-CPA secure.

Suppose there exists a PPT attacker $\mathcal{A}$ in our KEM system for a circuit of depth l and inputs of length n in the selective chosen plaintext security game, we can construct a PPT algorithm that solves the $l+1$- multilinear assumption with non-negligible advantage.

- **Theorem 5.1.** The proposed CP-ABE scheme that constitutes the KEM part is secure in the sense of IND-CPA for arbitrary circuits of depth $k-1$ under the k-MDDH assumption.

Proof. For VD-CPABE cryptosystems, we should consider two types of adversaries. The adversary $\mathcal{A}_1$ represents a normal third party attacker against the VD-CPABE scheme. The adversary $\mathcal{A}_2$ represents a malicious cloud server who obtains partial private key of the users.

Algorithm $\mathcal{B}$-1 (For the adversary $\mathcal{A}_1$ who does not comply with the challenge access policy)

- **Init.** Firstly, the challenger set the group $\vec{G} = (G_1, ..., G_k)$ with an efficient multilinear map e, a generator g and an instance $g, g^a, g^{c_1}, ..., g^{c_k} \in G_1, T \in G_k$. Then it flips a fair binary coin u outside of $\mathcal{B}$'s view. If $u = 0$. The challenger sets $T = g_k^{a\prod_{j\in[1,k]} c_j}$; otherwise it sets T as a random group element in G_k.
  Next, the attacker declares the challenge access policy f^*.
  Remark. When the attacker declares the challenge access policy f^*, for simplicity, we will focus our attention on the original policy f^*. Similarly, we can prove the security of the encryption for policy $\bar{f}^*$.
- **Setup.** Given a security parameter λ, the depth l for the circuit and the number of attributes n, $\mathcal{B}$ chooses random $y_1, ..., y_{2n} \in Z_p$. For $i \in [1, 2n]$, $\mathcal{B}$ sets $h_i = g^{-a+v_i}$, $y = g^a$, $g_k^\alpha = g_k^{ac_k}$ and sends $PK = (g_k^\alpha, H_1, H_2, y, h_1, ..., h_n, h_{n+1}, ..., h_{2n})$ to $\mathcal{A}_1$.
- **KeyGen Queries I**. The adversary makes repeated private keys corresponding to sets of attributes $x \in \{0,1\}^n$. We require that $\forall i \in q_1$ we have $f^*(x_i) = 0$. $\mathcal{B}$ chooses random $t = -c_k + \xi$ and computes $K_H = g^\alpha y^t = g^{a\xi}$, $L = g^t = g^{-c_k+\xi}$, $K_i = (yh_i)^t = g^{v_i(-c_k+\xi)}$ if $x_i = 1$.
- **Encrypt.** $\mathcal{B}$ sets $g^\alpha = g^{ac_k}$ as the master key. Then $\mathcal{B}$ computes the challenge ciphertext as follows.
  1) $\mathcal{B}$ sets $C'_1 = g_{k-1}^s = g_{k-1}^{\prod_{j\in[1,k-1]} c_j + y_{n+q}}$, where y_{n+q} is chosen at random.
  2) For the circuit $f^* = (n, q, A, B, GateType)$, $\mathcal{B}$ computes the ciphertext components for each wire w as follows.
     - **Input wire**. For $w \in [1, n]$, $\mathcal{B}$ chooses x_w at random, sets $z_w = c_1$ and computes
     $C_{w,1} = g^{a(c_1+y_w)}(h_w)^{-c_1} = g^{ay_w+v_w c_1}$
     $C_{w,2} = g^{z_w} = g^{c_1}$
     When $x_w = 0$, we can see r_w as $a(c_1 + y_w)$ and the adversary is try to compute $g_2^{a(c_1+y_w)t}$ without knowing h_w^t.
     Remark, in our practical scheme, when $x_w = 0$ the user needs to compute nothing for the wire. When $x_w = 1$, we can see r_w as ay_w and knowing $y^t h_w^t$ the adversary can compute $g_2^{ay_w t}$ correctly.
     - **Gate OR**. For $w \in [n+1, n+q-1]$, GateType(w)= OR and $j = depth(w)$. $\mathcal{B}$ chooses random y_w, sets $a_w = c_j$ and computes
     $C_{w,1} = g^{a_w} = g^{c_j}$
     $C_{w,2} = g_j^{a(r_w - a_w r_{A(w)})} = g_j^{ay_w - ac_j y_{A(w)}}$
     $C_{w,3} = g_j^{a(r_w - a_w r_{B(w)})} = g_j^{ay_w - ac_j y_{B(w)}}$
     When $x_w = 0$, we can see r_w as $ac_1c_2\cdots c_j + y_w$. Otherwise, we can see r_w as ay_w.
     - **Gate AND**. For $w \in [n+1, n+q-1]$, GateType(w)= AND and $j = depth(w)$. $\mathcal{B}$ chooses random y_w and computes g^{c_j}. It sets $(C_{w,1}, C_{w,2}) = (g^{c_j}, g)$. Then it computes the ciphertext for the gate as the following tuples.
     $C_{w,3} = g_j^{a(r_w - a_w r_{A(w)} - b_w r_{B(w)})} = (g_j^{a(y_w - c_j y_{A(w)} - y_{B(w)})}, g_j^{a(y_w - c_j y_{B(w)} - y_{A(w)})}, g_j^{a(y_w - c_j y_{A(w)} - y_{B(w)} - a_1\cdots a_{j-1})})$.
     the adversary could select the appropriate tuple to compute the value $g_{j+1}^{ar_w t}$. When $x_w = 0$ and $x_{A_w} = 0$, we can see r_w and a_w as $ac_1c_2\cdots c_j + y_w$ and g^{c_j}. When $x_w = 0$ and $x_{B_w} = 0$, we can see r_w and b_w as $ac_1c_2\cdots c_j + y_w$ and g^{c_j}. Otherwise, we can

1045-9219 (c) 2015 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPDS.2015.2392752, IEEE Transactions on Parallel and Distributed Systems

see r_w as ay_w.
$\mathcal{B}$ generates the challenge ciphertext as $T \cdot g_k^{ac_k x_{n+q}}$ and the description of the circuit f^*. Then $\mathcal{B}$ sends them to $\mathcal{A}_1$.

- **KeyGen Queries II**. The adversary makes repeated private key queries corresponding to the sets of attributes $x_{q_1}, ..., x_q$ where $f^*(x) = 0$. The challenger responds the key queries as in phase **I**.
- **Guess**. The adversary outputs a guess b' of b. If $b' = b$ it guesses that T is a tuple; otherwise, it guesses that it is random.

This immediately shows that the adversary $\mathcal{A}_1$ with non-trivial advantage in the KEM security game will have an identical advantage in breaking the k-MDDH assumption.

Algorithm $\mathcal{B}$-2 (For the adversary $\mathcal{A}_2$ who is the malicious cloud server)

Though the malicious cloud server could partial decrypt the ciphertext g_k^{ast}. We'll show that the adversary $\mathcal{A}_2$ having non-negligible advantage to distinguish $g_k^{\alpha s}$ from a random element in G_k.

- **Init.** Firstly, the challenger generates k-MDDH instance as in **Algorithm** $\mathcal{B}$-1.
  Next, the attacker declares the challenge partial decrypted ciphertext $g_k^{ast^*}$.
- **Setup.** Given a security parameter λ, the depth l for the circuit and the number of attributes n, $\mathcal{B}$ chooses random $y_1, ..., y_{2n} \in Z_p$. For $i \in [1, 2n]$, $\mathcal{B}$ sets $h_i = g^{v_i}$, $Y = g^a$, $g_k^{\alpha} = g_k^{ac_k}$ and sends $PK = (g_k^{\alpha}, H_1, H_2, Y, h_1, ..., h_n, h_{n+1}, ..., h_{2n})$ to $\mathcal{A}_2$.
- **KeyGen Queries I**. The adversary makes repeated private keys for any set S.
  $\mathcal{B}$ chooses random $t = -c_k + \xi$ and computes $K_H = g^{\alpha} y^t = g^{a\xi}$. We require that the private key query for t^* have not be answered.
- **Encrypt.** $\mathcal{B}$ computes $T \cdot g^{ac_k x_{n+q}}$ and the challenge ciphertext $C^* = g_k^{at^*(x_{r_{n+q}} + \prod_{j \in [1,k-1]} c_j)}$. Then $\mathcal{B}$ sends them to $\mathcal{A}_2$.
- **KeyGen Queries II**. The adversary makes repeated private keys for any set S.
  $\mathcal{B}$ responds the key queries as in phase **I** where $t \neq t^*$.
- **Guess**. The adversary outputs a guess b' of b. If $b' = b$ it guesses $u' = 0$ to indicate that T is a tuple; otherwise, it guesses $u' = 1$ to indicate that T is random in G_k.

We suppose the polynomial-time adversaries $\mathcal{A}_1$, $\mathcal{A}_2$ can attack this scheme with advantage ε_k. We will compute the probability that the simulator $\mathcal{B}$ can solve the k-MDDH problem.

When $u = 0$ the adversary has an advantage ε to attack this scheme that is $Pr[b = b' | u = 0] = \frac{1}{2} + \varepsilon_k$. The simulator will guess $u' = 0$ if $b = b'$, so we have $Pr[u = u' | u = 0] = \frac{1}{2} + \varepsilon_k$.

When $u = 1$ the adversary has no advantage to guess b. Therefore $Pr[b \neq b' | u = 1] = \frac{1}{2}$. The simulator will guess $u' = 1$ if $b \neq b'$, so we have $Pr[u = u' | u = 1] = \frac{1}{2}$.

Thus, we compute the overall advantage that the simulator solves the k-MDDH problem is

$Pr[u = u'] = Pr[u = 0]Pr[u = u' | u = 0] + Pr[u = 1]Pr[u = u' | u = 1] = \frac{1}{2} + \frac{\varepsilon_k}{2}$.

- **Theorem 5.2.** If the KEM is CPA secure and the AE is CCA secure then the proposed hybrid CP-ABE scheme is CPA secure.

Proof. Suppose there exist a polynomial-time adversary attacks the AE scheme with advantage ε_a and an adversary attacks the KEM scheme with advantage ε_k, then the advantage for the adversary, who attacks the proposed hybrid encryption, is $\varepsilon < 2\varepsilon_k + \varepsilon_a$.

Now we define two games to prove security. The experiment Exp_1 is specified by our VD-CPABE game that interacts with the adversary in the manner described in the definition of the CPA experiment. The experiment Exp_2 modifies the VD-CPABE algorithm that the encryption key for the AE algorithm is chosen at random from key space rather than the legitimate one generated by the KEM algorithm.

Let A and B be the events that $v' = v$ appears in Exp_1 and Exp_2 respectively. Then we show that the adversary's views in Exp_1 and Exp_2 are indistinguishable. In particular, $|Pr[A] - Pr[B]| \leq 2\varepsilon_k$.

Consider a simulator $\mathcal{B}$ that interacts with an adversary $\mathcal{A}_1$ who attacks the KEM scheme by using $\mathcal{A}_A$. $\mathcal{B}$ runs setup algorithm and gives PK to $\mathcal{A}_1$. $\mathcal{A}_1$ passes PK to $\mathcal{A}_A$ and queries K_b to $\mathcal{B}$ by using the encryption oracle of Game.KEM. Then $\mathcal{A}_1$ flips a coin v and computes $C = AE.Enc_{K_b}(M_v)$. It sends C to $\mathcal{B}$ and get a KEM ciphertext CK for C. Then $\mathcal{A}_1$ sends (C, CK) to $\mathcal{A}_A$. When $\mathcal{A}_A$ outputs $v' = v$, $\mathcal{A}_1$ outputs $b' = 0$ to indicate that K_b is the real key. Otherwise, if $v' \neq v$, $\mathcal{A}_1$ outputs $b' = 1$ to indicate that K_b is a random element.It is clear by construction that when $b = 0$ the view of $\mathcal{A}_A$ is identical to that in Exp_1 and when $b = 1$ the view of $\mathcal{A}_A$ is identical to that in Exp_2. That is $Pr[v' = v | b = 0] = Pr[A]$ and $Pr[v' = v | b = 1] = Pr[B]$. Therefore,

$$\begin{aligned}
&\tfrac{1}{2}(\Pr[A] - \Pr[B]) \\
&= \tfrac{1}{2}(\Pr[v' = v | b = 0] - \Pr[v' = v | b = 1]) \\
&= \tfrac{1}{2}(\Pr[b' = 0 | b = 0] - \Pr[b' = 0 | b = 1]) \\
&= \tfrac{1}{2}(\Pr[b' = b | b = 0] - (\tfrac{1}{2} - \tfrac{1}{2}\Pr[b' = b | b = 1])) \\
&= \Pr[b' = b] - \tfrac{1}{2}
\end{aligned}$$

Since $|Pr(b' = b) - \frac{1}{2}| \leq \varepsilon_k$, we have $|Pr[A] - Pr[B]| \leq 2\varepsilon_k$.

Next, we will show that $|Pr[B] - \frac{1}{2}| \leq \varepsilon_a$.

Consider a simulator $\mathcal{B}$ that interacts with an adversary $\mathcal{A}_2$ who attacks the modified VD-CPABE scheme by using $\mathcal{A}_A$. When receives M_0 and M_1 from $\mathcal{A}_A$, $\mathcal{A}_2$ submits them to simulator to get a ciphertext C_M by using AE encryption algorithm. It then requests a KEM encryption query and get a ciphertext CK for C. Then $\mathcal{A}_2$ sends (C, CK) to $\mathcal{A}_A$. When $\mathcal{A}_A$ outputs v', $\mathcal{A}_2$ outputs v'. We can see that Exp_2 can be perfectly

1045-9219 (c) 2015 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPDS.2015.2392752, IEEE Transactions on Parallel and Distributed Systems

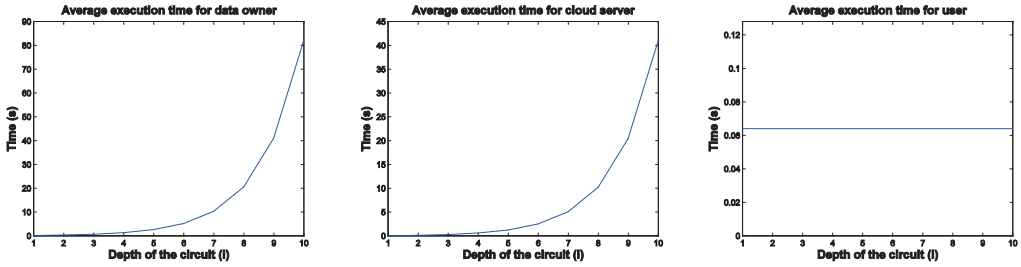

Fig. 4. Performance of our hybrid VD-CPABE scheme

TABLE 2
Pairing operation time

Parameter	$\lambda = 62$ $\beta = 80$	$\lambda = 80$ $\beta = 160$	$\lambda = 160$ $\beta = 200$
Time	15ms	16ms	31ms

simulated and whenever $\mathcal{A}_A$ wins, $\mathcal{A}_2$ wins. Hence $|Pr[B] - \frac{1}{2}| \leq \varepsilon_a$.

We define the advantage that the adversary wins in Exp_1 as ε, that is $|Pr[A] - \frac{1}{2}| \leq \varepsilon$. Since $|Pr[B] - \frac{1}{2}| \leq \varepsilon_a$, $|Pr[A] - Pr[B]| \leq 2\varepsilon_k$.

Then we have $\varepsilon < 2\varepsilon_k + \varepsilon_a$, where ε_k and ε_a are assumed negligible.

Thus, the proposed system could be applied to protect the data's confidentiality.

6 Implementation

In this section, we simulate the cryptographic operations by using of the Gnu MP library [20] in vc 6.0. The experiments are performed on a computer using the Intel Core i5-2400 at a frequency of 3.10 GHz with 4GB memory and Windows 7 operation system. Without considering the addition of two elements over the integer, the hash function and exclusive-OR operations, we denote the cost of a multilinear pairing by P. λ denotes the security parameter. β denotes the group elements size in bits. With different parameters, the average running time of P operation in 100 times is obtained and demonstrated in TABLE 2. For P operations, in order to implement in practice efficiently, we use the optimized definition in [23].

We instantiate our hybrid VD-CPABE scheme with $\lambda = 80$ and $\beta = 160$. When we operate the encryption and partial decryption algorithms, the input wire and the AND gate need to garble twice and the OR gate needs to garble triple. The algorithm for generating MAC needs one garbling operation and other addition operations over the integer, and the algorithm for verifying MAC needs to garble triple. Based on the above parameter settings, the most running time to finish our encryption and decryption algorithms are illustrated in Fig. 4.

In addition, suppose that the symmetric cipher is 128-bit. The bandwidth of the transmitted ciphertext for the data owner grows with the increase of the depths of circuit. For the user, The bandwidth of the transmitted ciphertext is $(128 \times 2 + 160 \times 3)/8 = 92$ bytes. Obviously, for the data owner and the cloud server, the computation time grows exponentially with the increase of the depth of circuit. When $depth(C) = 1$, these computation are $96ms$ and $0ms$, respectively. While the cost of computation consumption at the user side is just $64ms$ which is independent of the depth of the circuit. Thus our scheme enables to provide an efficient method to share and protect the confidential information between users with limited power and data owners with vast amount of data in the cloud.

7 Conclusion

To the best of our knowledge, we firstly present a circuit ciphertext-policy attribute-based hybrid encryption with verifiable delegation scheme. General circuits are used to express the strongest form of access control policy. Combined verifiable computation and encrypt-then-mac mechanism with our ciphertext-policy attribute-based hybrid encryption, we could delegate the verifiable partial decryption paradigm to the cloud server. In addition, the proposed scheme is proven to be secure based on k-multilinear Decisional Diffie-Hellman assumption. On the other hand, we implement our scheme over the integers. The costs of the computation and communication consumption show that the scheme is practical in the cloud computing. Thus, we could apply it to ensure the data confidentiality, the fine-grained access control and the verifiable delegation in cloud.

Acknowledgments

The authors would like to thank NSFC (Grant Nos. 61300181, 61272057, 61202434, 61170270, 61100203, 61121061), the Fundamental Research Funds for the Central Universities (Grant Nos. 2012RC0612, 2011YB01).

References

[1] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. H. Katz, A. Konwinski, G. Lee, D. A. Patterson, A. Rabkin, I. Stoica and M. Zaharia, "Above the Clouds: A Berkeley View of Cloud Computing," University of California, Berkeley, Technical Report, no. UCB/EECS-2009-28, 2009.

1045-9219 (c) 2015 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPDS.2015.2392752, IEEE Transactions on Parallel and Distributed Systems

[2] M. Green, S. Hohenberger and B. Waters, "Outsourcing the Decryption of ABE Ciphertexts," in Proc. USENIX Security Symp., San Francisco, CA, USA, 2011.

[3] J. Lai, R. H. Deng, C. Guan and J. Weng, "Attribute-Based Encryption with Verifiable Outsourced Decryption," in Proc. IEEE Transactions on information forensics and security, vol. 8, NO. 8, pp.1343-1354, 2013.

[4] A. Lewko and B. Waters, "Decentralizing Attribute-Based Encryption," in Proc. EUROCRYPT, pp.568-588, Springer-Verlag Berlin, Heidelberg, 2011.

[5] B. Waters, "Ciphertext-Policy Attribute-Based Encryption: an Expressive, Enficient, and Provably Secure Realization," in Proc. PKC, pp.53-70, Springer-Verlag Berlin, Heidelberg, 2011.

[6] B. Parno, M. Raykova and V. Vaikuntanathan, "How to Delegate and Verify in Public: verifiable computation from attribute-based encryption," in Proc. TCC, pp.422-439, Springer-Verlag Berlin, Heidelberg, 2012.

[7] S. Yamada, N. Attrapadung and B. Santoso, "Verifiable Predicate Encryption and Applications to CCA Security and Anonymous Predicate Authentication," in Proc. PKC, pp.243-261, Springer-Verlag Berlin, Heidelberg, 2012.

[8] J. Han, W. Susilo, Y. Mu and J. Yan, "Privacy-Preserving Decentralized Key-Policy Attribute-Based Encryption," in Proc. IEEE Transactions on Parallel and Distributed Systems, 2012.

[9] S. Garg, C. Gentry, S. Halevi, A. Sahai and B. Waters, "Attribute-Based Encryption for Circuits from Multilinear Maps," in Proc. CRYPTO, pp.479-499, Springer-Verlag Berlin, Heidelberg, 2013.

[10] S. Gorbunov, V. Vaikuntanathan and H. Wee, "Attribute-Based Encryption for Circuits," in Proc. STOC, pp.545-554, ACM New York, NY, USA, 2013.

[11] A. Sahai and B. Waters, "Fuzzy Identity Based Encryption," in Proc. EUROCRYPT, pp.457-473, Springer-Verlag Berlin, Heidelberg, 2005.

[12] V. Goyal, O. Pandey, A. Sahai and B. Waters, "Attribute-based Encryption for Fine-grained access control of encrypted data," in Proc. CCS, pp.89-98, ACM New York, NY, USA, 2006.

[13] R. Cramer and V. Shoup, "A Practical Public Key Cryptosystem Provably Secure against Adaptive Chosen Ciphertext Attack," in Proc. CRYPTO, pp.13-25, Springer-Verlag Berlin, Heidelberg, 1998.

[14] R. Cramer and V. Shoup, "Design and Analysis of Practical Public-Key Encryption Schemes Secure against Adaptive Chosen Ciphertext Attack," in Proc. SIAM Journal on Computing, vol. 33, NO. 1, pp.167-226, 2004.

[15] D. Hofheinz and E. Kiltz R, "Secure hybrid encryption from weakened key encapsulation," in Proc. CRYPTO, pp.553-571, Springer-Verlag Berlin, Heidelberg, 2007.

[16] M. Abe, R. Gennaro and K. Kurosawa, "Tag-KEM/DEM:A New Framework for Hybrid Encryption," in Proc. CRYPTO, pp.97-130, Springer-Verlag New York, NJ, USA, 2008.

[17] K. Kurosawa and Y. Desmedt, "A New Paradigm of Hybrid Encryption Scheme," in Proc. CRYPTO, pp.426-442, Springer-Verlag Berlin, Heidelberg, 2004.

[18] J. Li, X. Huang, J. Li, X. Chen and Y. Xiang, "Securely Outsourcing Attribute-based Encryption with Checkability," in Proc. IEEE Transactions on Parallel and Distributed Systems, 2013.

[19] J. Hur and D. K. Noh, "Attribute-Based Access Control with Efficient Revocation in Data Outsourcing Systems," in Proc. IEEE Transactions on Parallel and Distributed Systems, 2011.

[20] T. Granlund and the GMP development team, "GNU MP: The GNU Multiple Precision Arithmetic Library, 5.1.1," 2013, http://gmplib.org/.

[21] W. Nagao, Y. Manabe and Tatsuaki Okamoto, "A Universally Composable Secure Channel Based on the KEM-DEM Framework," in Proc. CRYPTO, pp.426-444, Springer-Verlag Berlin, Heidelberg, 2005.

[22] M. Bellare and C. Namprempre, "Authenticated Encryption: Relations among Notions and Analysis of the Generic Composition Paradigm," in Proc. ASIACRYPT, pp.531-545, Springer-Verlag Berlin, Heidelberg, 2000.

[23] J. Coron, T. Lepoint and M. Tibouchi, "Practical Multilinear Maps over the Integer," in Proc. CRYPTO, pp.476-493, Springer-Verlag Berlin, Heidelberg, 2013.

[24] S. Garg, C. Gentry and Shai Halevi, "Candidate Mulitilinear Maps from Ideal Lattices and Applications," in Proc. EUROCRYPT, pp.1-17, Springer-Verlag Berlin, Heidelberg, 2013.

Jie Xu received the B.S. degree in information and computation science from Qingdao University of Science and Technology, China, in 2009. She is currently working toward the PhD degree in computer science and technology in Beijing University of Posts and Telecommunications. Her research interests include functional encryption and cloud security.

Qiaoyan Wen received the B.S. and M.S. degrees in Mathematics from Shaanxi normal University, Xi'an, China, in 1981 and 1984, respectively, and the Ph.D degree in cryptography from Xidian University, Xi'an, China, in 1997. She is a professor of Beijing University of Posts and Telecommunications. Her present research interests include coding theory, cryptography, information security, internet security and applied mathematics.

Wenmin Li received the B.S. and M.S. degrees in Mathematics and Applied Mathematics from Shaanxi Normal University, Xi'an, Shaanxi, China, in 2004 and 2007, respectively, and the Ph.D. degree in Cryptology from Beijing University of Posts and Telecommunications, Beijing, China, in 2012. Now she is a lecturer of Beijing University of Posts and Telecommunications. Her research interests include cryptography and information security.

Zhengping Jin received the BS degree in Math and Applied Math, MS degree in Applied Math from Anhui Normal University in 2004 and in 2007 respectively, and the Ph.D degree in Cryptography from Beijing University of Posts and Telecommunications in 2010. Now he is a lecturer of Beijing University of Posts and Telecommunications. His research interests include cryptography, information security, internet security and applied mathematics.

1045-9219 (c) 2015 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.