Hindawi Publishing Corporation
International Journal of Distributed Sensor Networks
Volume 2015, Article ID 101287, 11 pages
http://dx.doi.org/10.1155/2015/101287

*Research Article*

# Achieving Secure and Efficient Data Access Control for Cloud-Integrated Body Sensor Networks

**Zhitao Guan,[1] Tingting Yang,[1] and Xiaojiang Du[2]**

[1]*School of Control and Computer Engineering, North China Electric Power University, Beijing 102206, China*
[2]*Department of Computer and Information Science, Temple University, Philadelphia, PA 19122, USA*

Correspondence should be addressed to Zhitao Guan; guanzhitao@126.com

Received 29 May 2015; Revised 22 July 2015; Accepted 27 July 2015

Academic Editor: Antonio Puliafito



Body sensor network has emerged as one of the most promising technologies for e-healthcare, which makes remote health monitoring and treatment to patients possible. With the support of mobile cloud computing, large number of health-related data collected from various body sensor networks can be managed efficiently. However, how to keep data security and data privacy in cloud-integrated body sensor network (C-BSN) is an important and challenging issue since the patients' health-related data are quite sensitive. In this paper, we present a novel secure access control mechanism MC-ABE (Mask-Certificate Attribute-Based Encryption) for cloud-integrated body sensor networks. A specific signature is designed to mask the plaintext, and then the masked data can be securely outsourced to cloud severs. An authorization certificate composed of the signature and related privilege items is constructed which is used to grant privileges to data receivers. To ensure security, *a unique value* is chosen to mask the certificate for each data receiver. Thus, the certificate is unique for each user and user revocation can be easily completed by removing the mask value. The analysis shows that proposed scheme can meet the security requirement of C-BSN, and it also has less computation cost and storage cost compared with other popular models.

## 1. Introduction

Body sensor network (BSN) emerges recently with rapid development of wearable sensors, implantable sensors, and short range wireless communication, which make pervasive healthcare monitoring and management become increasingly popular [1, 2]. By the body sensor network, health-related data of the patient can be collected and transferred to the healthcare staff in real time, so the patient's state of health can be under monitoring and precautions can be taken if something bad happened.

In order to enhance the scalability of the body sensor network, some work focuses on combining cloud computing and body sensor network together. As shown in Figure 1, with the support of mobile cloud computing, cloud-integrated body sensor network (C-BSN) can be constructed [3]. In C-BSN, massive local body sensor networks are integrated together and mass data are collected and stored in cloud servers; healthcare staffs will continually monitor their patients' status and exchange views when it is difficult to make diagnosis; researchers can make data analysis to get some useful results such as regularity of disease development; government agencies also can take measures on disease prevention and control based on data analysis.

However, there are still several problems and challenges in C-BSN [3, 4]. For example, data security and data privacy must be concerned since patient-related data is private and sensitive. In this paper, we propose a secure data access control scheme named MC-ABE, which can efficiently ensure data security and data privacy. For data security, data can be securely transferred from data owners to the cloud servers and securely stored; for data privacy, data can be only accessed by authorized users with fine-grained policies.

For example, Bob (data owner) is a patient, and Alice (data requester) is his healthcare doctor. By C-BSN, Bob's health-related data can be collected and sent to cloud server in real time; and Alice gets Bob's information from cloud server to monitor his health status. Besides the authorized

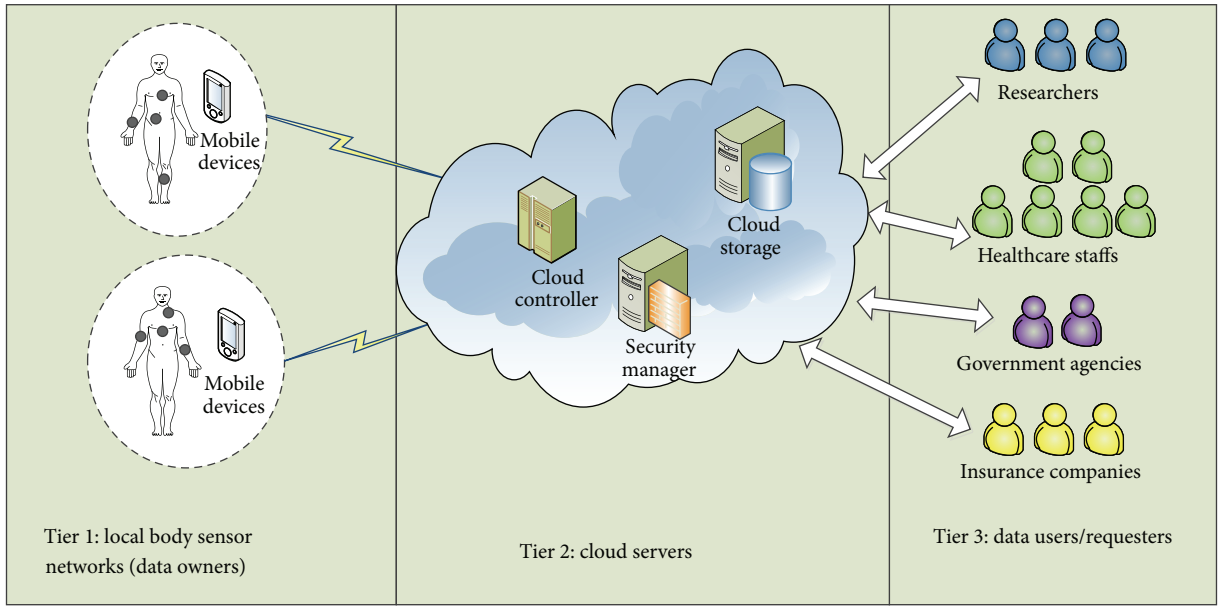


Figure 1: Conceptual architecture of cloud-integrated body sensor network.

person, Bob does not want anyone else to know about his health data. However, his information may be leaked in many ways: the cloud operator/administrator may access his data; malicious user may intrude into the cloud server to steal user data; unauthorized DR may exceed to access others' data. In summary, there are three key problems which need to be solved to ensure the users' data security and data privacy in C-BSN. Firstly, the cloud is semitrusted; that is, although we outsource the data to the cloud, we still need to prevent cloud operators from accessing the data content; secondly, we must take measures to keep malicious users out of C-BSN system; lastly, it is also important to study how to avoid the unauthorized access of other users.

In this paper, we propose a novel secure access control mechanism MC-ABE to tackle the aforementioned problems. And main contributions of this paper can be summarized as follows:

(i) We construct one specific signature to CP-ABE to mask the plaintext and then realize securely encryption/decryption outsourcing.

(ii) We construct the unique authentication certificate for each visitor, which makes the system achieve more effective control on malicious visitors; in particular, it also leads to a low cost for user revocation.

(iii) We introduce the third-party trust authority to manage above-mentioned signatures and certificates, which can guarantee data security even if the cloud server is semitrusted.

(iv) In C-BSN, processing data in time is quite necessary. Our proposed scheme can meet such requirement. From the section of performance evaluation, our scheme takes less time than other compared methods to do data collecting, data transmission, and data acquisition.

The rest of this paper is organized as follows. Section 2 introduces the related work. Then, in Section 3, some preliminaries are given. Our scheme is stated in Section 4. In Section 5, security analysis is given. In Section 6, the performance of our scheme is evaluated. The paper is concluded in Section 7.

## 2. Related Work

Recently, various techniques have been proposed to address the problems of data security and data privacy in C-BSN. In [5], Sahai and Waters proposed the Attribute-Based Encryption (ABE) to realize access control on encrypted data. In ABE, the ciphertext's encryption policy is associated with a set of attributes, and the data owner can be offline after data is encrypted. One year later, Goyal et al. proposed a new type of ABE, Key-Policy Attribute-Based Encryption (KP-ABE) [6]. In KP-ABE, the ciphertext's encryption policy is also associated with a set of attributes, but the attributes are organized into a tree structure (named access tree). The benefit of this approach is that more flexible access control strategy can be got and fine-grained access control can be realized. However, data owner was short of entire control over the encryption policy; that is, he cannot decide who can access the data and who cannot. To solve this problem, Bethencourt et al. proposed CP-ABE (Ciphertext-Policy Attribute-Based Encryption) [7], in which data owner constructed the access tree together with visitors' identity information. The user can decrypt the ciphertext if and only if attributes in his private key match the access tree. So, in CP-ABE, data owner can configure more flexible access policy. In [8], Yu et al. tried to achieve secure, scalable, and fine-grained access control in cloud environment. Their proposed scheme is based on KP-ABE and combines with the other two techniques, proxy reencryption and lazy reencryption. It is proved that the

proposed scheme can meet the security requirement in cloud quite well. Similarly, Wang et al. proposed an access control scheme based on CP-ABE, which is also secure and efficient in cloud environment [9].

In [10], Ahmad et al. proposed a multitoken authorization strategy to remedy the weaknesses of the authorization architecture in mobile cloud. It reduces the probability of unauthorized access to the cloud data and service when malicious activity happened; for example, IdM (Identity Management Systems) are compromised, network links are eavesdropped, or even communication tokens are stolen. In [11], Yadav and Dave presented an access model based on CP-ABE which could provide the remote integrity check by the way of augmenting secure data storage operations. To reduce computation overhead and achieve secure encryption/decryption outsourcing, the access tree is divided into two parts: one part is encrypted by the data owner and the other part is encrypted by the cloud sever. So a portion of computation overhead was transferred from data owner to cloud sever. The similar method is also adopted in the work of Zhou and Huang [12]. In addition to the access tree division, Zhou and Huang also propose an efficient data management model to balance communication and storage overhead to reduce the cost of data management operations. In [13], Li et al. presented a low complexity multiauthority attribute-based encryption scheme for mobile cloud computing which uses masked shared-decryption-keys to ensure the security of decryption outsourcing and adopts multiauthorities for authorization to enhance security assurance. The above schemes are based on CP-ABE, in which complex bilinear map calculation is performed. In [14], Yao el al. proposed a novel access control mechanism, in which data operation privileges are granted based on authorization certificates. The advantage of such mechanism is that the computation cost can be decreased remarkably, since there is no bilinear map calculation. And the disadvantage is that lots of operations need to be handled by data owner, such as privilege designation, and then it demands that the data owner must know all information about the visitors. In [15], the authors considered the problem of patient self-controlled access privilege to highly sensitive Personal Health Information. They proposed a Secure Patient-Centric Access Control scheme which allows data requesters to have different access privileges based on their roles and then assigns different attribute sets to them. However, they took the cloud server as trusted, and their scheme does not work well for user revocation. In [16], the authors proposed a novel CP-ABE scheme with constant-size decryption keys independent of the number of attributes. Their scheme is suitable for applications based on lightweight mobile devices but is not suitable for large scale C-BSN.

## 3. Preliminaries

*3.1. Notations.* The notations used in MC-ABE are listed as follows.

*Notations in MC-ABE.* Consider the following:

DO: data owner,

DR: data requester/receiver,

ESP: encryption service provider,

DSP: decryption service provider,

SSP: storage service provider,

TA: trust authority,

SetS: setup server,

PK: public key,

MK: master key,

SK: secret key,

$M$: plaintext,

CT: ciphertext,

$T$: access tree,

MM: masked plaintext,

Cert: authorization certificate,

MValue: mask value,

MCert: masked certificate.

DO and DR are cloud users. ESP is cloud server that can help DO do data encryption. SSP is cloud storage server. DSP is the server that is responsible for data decryption. TA is the third-party trust authority. SetS is the setup server whose responsibility is to generate PK and MK.

PK and MK are parameters that are used for data encryption/decryption. SK is held by DR which is used to decrypt ciphertext, which is generated using PK and MK. The data is plaintext before encryption, denoted as $M$, and CT is the ciphertext of $M$. $T$ is the access policy (access tree). MM is the masked plaintext; in MC-ABE, the plaintext will be masked to MM by a signature before being encrypted to achieve "double protection." Cert is the authorization certificate (see Section 4.2.1 for details). Mask value is used to mask Cert to generate MCert (see Section 4.2.2 for details).

*3.2. Basics*

*3.2.1. Bilinear Pairing.* Let $G_1$ and $G_2$ be two multiplicative cyclic groups of prime order $p$. Let $g$ be a generator of $G_1$ and let $e$ be a bilinear map, $e$: $G_1 \times G_1 \rightarrow G_2$. For $a, b \in Z_p$, the bilinear map $e$ has the following properties [3, 10]:

(1) Bilinearity: for all $u, v \in G_1$, we have $e(u^a, v^b) = e(u, v)^{ab}$.

(2) Nondegeneracy: $e(g, g) \neq 1$.

(3) Being symmetric: $e(g^a, g^b) = e(g, g)^{ab} = e(g^b, g^a)$.

*3.2.2. Discrete Logarithm (DL) Problem*

*Definition 1* (discrete logarithm (DL) problem). Let $G$ be a multiplicative cyclic group of prime order $p$ and let $g$ be its generator, for all $\alpha \in Z_p$, given $g, g^\alpha$ as input, output $\alpha$.

The DL assumption holds in $G$ if it is computationally infeasible to solve the DL problem in $G$ [17].

### 3.3. Ciphertext-Policy Attribute-Based Encryption (CP-ABE)

*3.3.1. Access Structure.* Let $P = \{P_1, P_2, \ldots P_n\}$ be the universal set [18]. Each element in $P$ is an attribute, that is, the descriptive identification information of a visitor [19]. An access structure is a collection (resp., monotone collection) $T$ of nonempty subsets of $P$. For example, $P$ = {Beijing, Shanghai, No. 1 Middle School, No. 2 Middle School, student, teacher, administrator}. Visitor 1 has the attributes set $A$ = {Beijing, No. 1 Middle School, student} [20].

The access structure in CP-ABE is the tree structure, which is named access tree [2]. For the access tree $T$, the leaf nodes are associated with descriptive attributes; each interior node is a relation function, such as AND ($n$ of $n$), OR (1 of $n$), and $n$ of $m$ ($m > n$).

Each DR has a set of attributes, which are associated with DR's SK. If DR's attributes set satisfies the access tree, the encrypted data can be decrypted by DR's SK.

*3.3.2. Working Process.* In CP-ABE, the plaintext is encrypted with a symmetric key, and then the key is shared in the access tree. In the process of decryption, if DR's SK satisfies the access tree, then DR gets the shared secret and the data can be recovered.

*3.4. Assumptions.* In this work, we make the following assumptions.

*Assumption 1* (service providers (ESP, DSP, and SSP) are semitrusted). That is, they will follow our proposed protocol in general but try to find out as much secret information as possible. And the information may be accessed illegally by internal malicious employees or external attackers. In particular, although ESP and DSP undertake most of the computing cost, they do not have enough information to deduce the plaintext.

*Assumption 2* (SetS and TA are trusted). On no conditions will they leak information about data and related keys.

In order to deduce more information about encrypted data, service providers might combine their information to perform collusion attack. In our scheme, collusions between service providers are taken into consideration.

## 4. MC-ABE

*4.1. Overview.* Our proposed scheme MC-ABE is shown in Figure 2. Seven algorithms are included in MC-ABE: Setup, $\text{Encrypt}_{\text{DO}}$, $\text{Encrypt}_{\text{ESP}}$, KeyGen, CerGen, $\text{Decrypt}_{\text{DSP}}$, and $\text{Decrypt}_{\text{DR}}$.

For data outsourcing, DO encrypts $M$ with algorithm $\text{Encrypt}_{\text{DO}}$, in which signature is used to mask $M$. Then ESP encrypts $T$ with algorithm $\text{Encrypt}_{\text{ESP}}$ to finish the encryption. The encrypted data is stored in SSP.

For data access, when DR requests data from SSP, the request is sent to TA after verification. TA chooses a unique value to the mask certificate for DR. Then, using the attributes set of DR, TA computes SK with algorithm KeyGen. After that, SK is sent to DSP and the certificate is sent to DR. At the same time, SSP sends the CT to DSP. With SK and CT, DSP can do decryption and get $M$ that is masked by signature. Once DR receives the certificate, he decrypts the masked certificate with his *unique value* (TA sends the unique value to this DR when the first authorized request occurred. It will be used in the following requests until this DR is revoked) to get the certificate. Using the certificate, DR can decrypt the masked $M$ with signatures in the certificate.

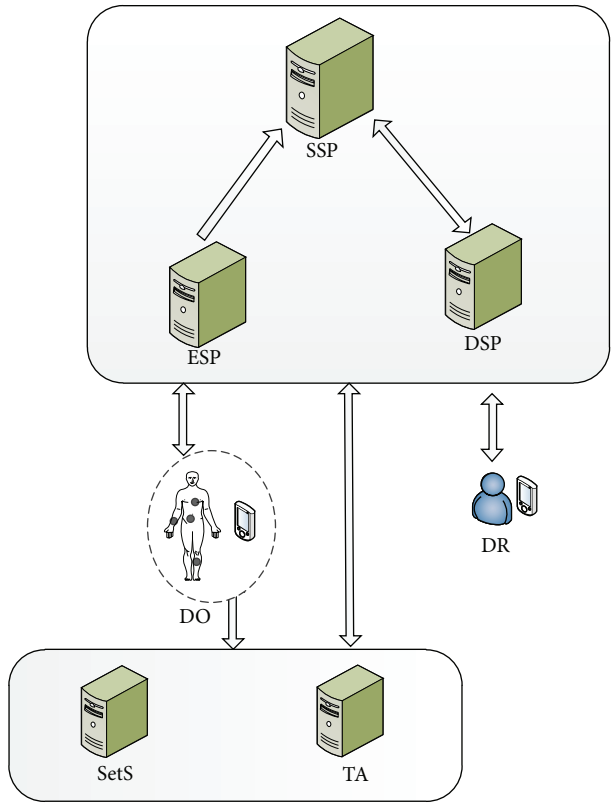


Figure 2: System model.

In addition, if a DR is revoked, TA will mark the DR as "revoked" and this DR's unique mask value will be invalid. No certificate will be granted to this DR any more.

### 4.2. Two Important Notions

*4.2.1. Authorization Certificate (Cert).* The authorization certificate is introduced in MC-ABE to grant data privileges for DR. As shown in Structure of Authorization Certificate, it includes five items that are privilege related information. DO provides the certificate related information to TA, and then TA constructs the unique authorization certificate for each authorized DR.

*Structure of Authorization Certificate*

File ID list ($f_1$, $f_2 \ldots$)

Valid Period (From the start time to the end time)

Signature ($\{\text{sign}_{f1}\}$, $\{\text{sign}_{f2}\} \ldots$)

Privilege ($\{p_{f1}\}$, $\{p_{f2}\} \ldots$)

PK, MK

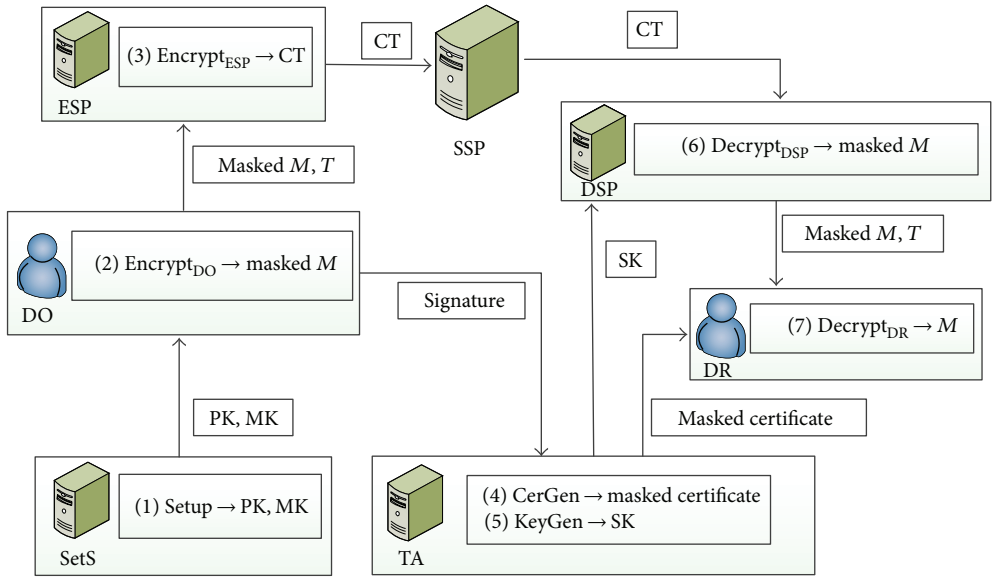


Figure 3: Algorithms' implementation in MC-ABE.

File ID is ID list of the authorized files. Valid Period denotes the valid period of the signature from the start time to the end time. Signature is used by DO to mask the plaintext in data encryption; it is used by DR to get the plaintext in data decryption. Privilege is the privilege denoted by the signature such as read, modify, or delete. PK, MK are two keys noted in Notations in MC-ABE.

*4.2.2. Mask Value (MValue).* To achieve fine-grained access control over DR, the mask value is introduced in MC-ABE. The mask value is maintained by TA. For each DR, TA sets a unique mask value for him. The mask value is used to blind the authorization certificate before the certificate is sent to DR. Thus, each DR receives its own unique blinded certificate since the mask value is unique. In the following, the process is described in detail.

After TA receives a data access request, it checks DRID firstly. If the requester is a new user, TA generates a random number $t_{\mathrm{DRID}} \in Z_p$ and inserts it into the mask value table.

Otherwise, if this DRID already exists in mask value table and the item of revocation is "N" (initial value of this item is "N." Only at the time when the DRID is revoked will this item be set as "Y"), TA invokes algorithm CerGen to compute the masked certificate (see Table 1).

*Algorithm* (CerGen($t_{\mathrm{DRID}}$, PK) → MCert). Construct a certificate Cert as Structure of Authorization Certificate shows. MCert is the masked Cert.

Then, compute as follows:

$$\begin{aligned} \mathrm{MValue} &= g^{t_{\mathrm{DRID}}} \\ \mathrm{MCert} &= \mathrm{Cert} \cdot e\left(g^{\theta}, g^{t_{\mathrm{DRID}}}\right) = \mathrm{Cert} \cdot e\left(g, g\right)^{\theta t_{\mathrm{DRID}}}. \end{aligned} \tag{1}$$

If DR is a new user, MValue and MCert will be sent to him. Otherwise, send MCert to the DR.

Table 1: Mask value table (maintained by TA).

| DRID | Mask value | Revocation |
|---|---|---|
| DR1 | MValue$_{DR1}$ | N |
| DR2 | MValue$_{DR2}$ | Y |
| DR3 | MValue$_{DR3}$ | N |

DRID: ID of DR.
Mask value: unique mask value for each DR.
Revocation: revocation mark. "Y" means this DR is revoked. "N" means this DR is authorized.

*4.3. Scheme Description.* The whole process of MC-ABE is shown in Figure 3. In this section, we describe each step in detail.

*4.3.1. Data Outsourcing.* In C-BSN, DO usually uses mobile devices that lack computing power and storage space. To reduce the encryption overhead of DO, the encryption process is divided into two parts: Encrypt$_{DO}$ and Encrypt$_{ESP}$. Encrypt$_{DO}$ is the encryption algorithm implemented by DO and Encrypt$_{ESP}$ is carried out by ESP. Since ESP is semitrusted, we introduce the signature in Encrypt$_{DO}$ to mask $M$. In general, there are three steps for data outsourcing.

*Firstly*, SetS generates PK and MK.

*Algorithm 2* (Setup → PK, MK). SetS performs the algorithm. Let $G_0$ be a multiplicative cyclic group of prime order $p$ and let $g$ be its generator, and four random numbers $\alpha, \beta, \varepsilon, \theta \in Z_p$ (further details in [7]). Consider

$$\begin{aligned} \mathrm{PK} &= \left(G_0, g, h = g^{\beta}, e\left(g, g\right)^{\alpha}, g^{\varepsilon}, g^{\theta}\right) \\ \mathrm{MK} &= \left(\beta, g^{\alpha}\right). \end{aligned} \tag{2}$$

*Secondly*, DO performs the first step of data encryption.

*Algorithm 3* ($\text{Encrypt}_{\text{DO}}(\text{PK}, M, K) \to \text{MM}$). DO implements the algorithm. PK is got from SetS; $M$ is DO's plaintext; MM is masked $M$; $K$ is the set of operation privileges, and $k$ is one of the elements in $K$.

For $k \subset K$, we choose a random number $v_k \subset Z_p$ and then compute the signature:

$$\text{signature}_k = e\left(g^{\varepsilon}, g^{v_k}\right) = e\left(g, g\right)^{\varepsilon v_k}. \tag{3}$$

For simplicity, let $v$ denote the set of $v_k$ : $v = \{v_k \mid k \in K\}$; signature denotes the set of $\text{signature}_k$ : signature = $\{\text{signature}_k \mid k \in K\}$.

Choose a random number $s \in Z_p$; then

$$\begin{aligned} \text{MM} = \widetilde{C} &= M \cdot e\left(g, g\right)^{as} \text{ signature} \\ &= M \cdot e\left(g, g\right)^{as} \cdot e\left(g, g\right)^{\varepsilon v}. \end{aligned} \tag{4}$$

*Lastly*, ESP performs the last step of data encryption.

*Algorithm 4* ($\text{Encrypt}_{\text{ESP}}(\text{PK}, s, T, \text{MM})$ [7, 11] $\to$ CT). ESP implements the algorithm. The access tree $T$ is encrypted from the root node $R$ to leaf nodes. For each node $x$ in $T$, choose a polynomial $q_x$.

For node $x$, consider the following:

$k_x$: it denotes the threshold value of $x$.

$d_x$: it denotes the degree of $q_x$; $d_x = k_x - 1$.

parent($x$): a function returns the parent node of $x$.

$\text{num}_x$: it is the number of child nodes of $x$. For a child node $y$, $y$ is uniquely identified by an index number index($y$), and $1 \leqslant \text{index}(y) \leqslant \text{num}_x$. Consider

$$q_x(0) = q_{\text{parent}(x)}\left(\text{index}\left(x\right)\right). \tag{5}$$

For root node $R$, $q_R(0) = s$. Choose $d_R$ other points randomly to completely define $q_R$. For any other node $x$ in $T$, let $q_x(0) = q_{\text{paerent}(x)}(\text{index}(x))$, and choose $d_x$ other points randomly to completely define $q_x$.

$Y$ is the set of leaf nodes in $T$. Compute as follows:

$$C = h^s \quad \forall y \in Y : C_y = g^{q_y(0)},\ C'_y = H\left(\text{att}\left(y\right)\right)^{q_y(0)}. \tag{6}$$

Then,

$$\begin{aligned} \text{CT} = \Big\{ T, \widetilde{C} = M \cdot e\left(g, g\right)^{as} \cdot e\left(g, g\right)^{\varepsilon v}, C = h^s,\ \forall y \\ \in Y : C_y = g^{q_y(0)}, C'_y = H\left(\text{att}\left(y\right)\right)^{q_y(0)} \Big\}. \end{aligned} \tag{7}$$

CT is stored in SSP. Detailed communication information is shown in Figure 4.

*4.3.2. Data Request.* When a DR requests data from SSP, TA generates SK and a certificate for DR. Most of decryption cost is taken by DSP but DSP cannot get $M$. Based on the effort of DSP, DR finishes the last step of decryption and gets $M$. Similar to data outsourcing, there are also three steps for data outsourcing.

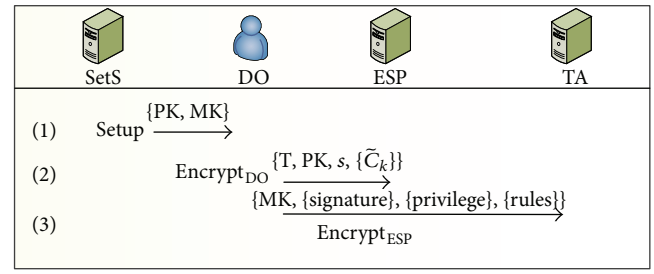


Figure 4: Communication information in data outsourcing.

*Firstly*, TA generates SK for DR.

*Algorithm 5* ($\text{KeyGen}(\text{MK}, S) \to \text{SK}$). $S$ is the attributes set of DR.

We generate a random number $r \in Z_p$ and then generate the random number $r_j \in Z_p$ for each $j \in S$. Compute as follows:

$$\begin{aligned} \text{SK} = \Big( D = g^{(\alpha+r)/\beta},\ \forall j \in S : D_j = g^r \cdot H\left(j\right)^{r_j}, D'_j \\ = g^{r_j} \Big). \end{aligned} \tag{8}$$

Then, TA sends SK to DSP.

*Secondly*, DSP performs the first step of data decryption: decrypt the access tree in CT to get MM.

*Algorithm 6* ($\text{Decrypt}_{\text{DSP}}(\text{SK}, \text{CT}) \to \text{MM}$). When $x$ is a leaf node, let $i = \text{att}(x)$. Function att($x$) denotes the attribute associated with the leaf node $x$ in the tree.

If $i \in S$,

$$\begin{aligned} \text{DecryptNodeL}\left(\text{CT}, \text{SK}, x\right) &= \frac{e\left(D_i, C_x\right)}{e\left(D'_i, C'_x\right)} \\ &= \frac{e\left(g^r \cdot H\left(i\right)^{q_x(0)}, g^{q_x(0)}\right)}{e\left(g^{r_i}, H\left(i\right)^{q_x(0)}\right)} = e\left(g, g\right)^{r \cdot q_x(0)}. \end{aligned} \tag{9}$$

Otherwise,

$$i \notin S, \quad \text{DecryptNodeL}\left(\text{CT}, \text{SK}, x\right) = \perp. \tag{10}$$

When $x$ is an interior node, call the algorithm DecryptNodeNL(CT, SK, $x$).

For all of the children $z$ of node $x$, call DecryptNodeL(CT, SK, $z$), and the output is $F_z$. Let $S_x$ be $k_x$ (the threshold value of interior node) random set and let $F_z \neq \perp$. If no such set exists, the function cannot be satisfied, so return $\perp$.

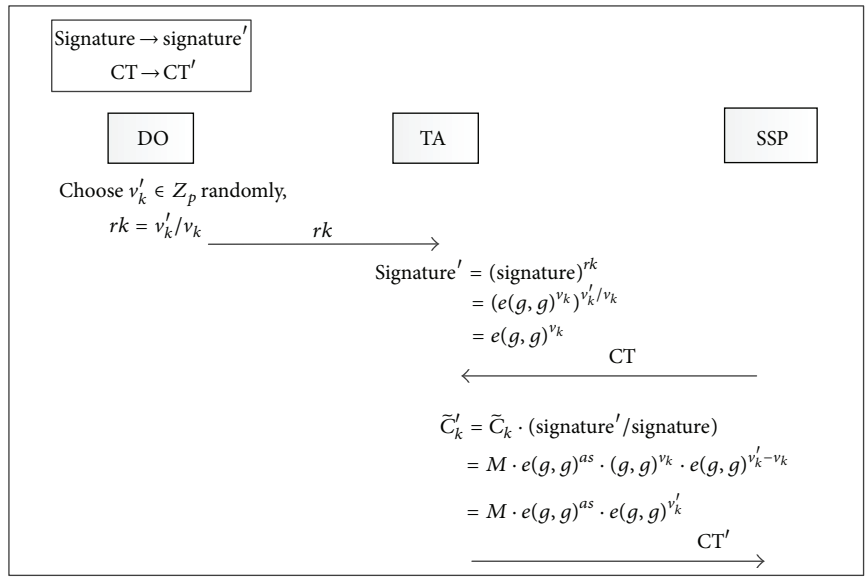


Figure 5: Signature updating.

Otherwise, compute as follows and return the result:

$$
\begin{aligned}
F_x &= \prod_{z \in S_x} F_z^{\Delta_{i,S_x'(0)}}, \\
&\quad \text{where } i = \text{index}(z),\ S_x' = \{\text{index}(z) : z \in S_x\} \\
&= \prod_{z \in S_x} \left(e(g,g)^{r \cdot q_z(0)}\right)^{\Delta_{i,S_x'(0)}} \\
&= \prod_{z \in S_x} \left(e(g,g)^{r \cdot q_{\text{parent}(z)}(\text{index}(z))}\right)^{\Delta_{i,S_x'(0)}} \\
&= \prod_{z \in S_x} \left(e(g,g)^{r \cdot q_x(i)}\right)^{\Delta_{i,S_x'(0)}} \\
&= e(g,g)^{r \cdot q_x(0)}.
\end{aligned}
\tag{11}
$$

In particular, for root node $R$,

$$
A = e(g,g)^{r q_T(0)} = e(g,g)^{r \cdot s}. \tag{12}
$$

Finally,

$$
\begin{aligned}
\frac{\widetilde{C}_k}{(e(C,D)/A)} &= \frac{\widetilde{C}}{\left(e\left(h^s, g^{(\alpha+r)/\beta}\right)/e(g,g)^{r \cdot s}\right)} \\
&= M \cdot \text{signature}.
\end{aligned}
\tag{13}
$$

Then, $M \cdot$ signature is sent to DR.

Receiving $M \cdot$ signature and MCert, DR implements algorithm $\text{Decrypt}_{\text{DR}}$ to finish data decryption.

*Lastly*, DR performs the last step of data decryption: remove the masked value in MM to get $M$.

*Algorithm 7* ($\text{Decrypt}_{\text{DR}}(M \cdot \text{signature}, \text{MCert}) \rightarrow M$). DR retrieves Cert to get related signatures:

$$
\frac{\text{MCert}}{e\left(g^{\theta}, g^{t_{\text{DRID}}}\right)} = \text{Cert} \cdot \frac{e(g,g)^{\theta t_{\text{DRID}}}}{e(g,g)^{\theta t_{\text{DRID}}}} = \text{Cert}. \tag{14}
$$

Then, DR gets $M$ with the signature:

$$
M \cdot \frac{\text{signature}}{\text{signature}} = M. \tag{15}
$$

*4.3.3. User Revocation.* An invalid DR is a DR who is thought to be malicious or whose certificate is expired. The invalid DR should be revoked from the authorized access list. In MC-ABE, we can remove the MValue record in Table 1 to revoke DR. Firstly, TA modifies the revoked DR's "Revocation" item from "N" to "Y" in mask value table. Secondly, current signature must be updated to a new one (signature updating is shown in Figure 5). After these two steps, the invalid DR is revoked. When he requests new data, he will be taken as new comer (the signature is updated, and he does not have the new one), and TA will refuse his request since he is marked as revoked. For valid DR, they will get the new signature and access the system as usual.

## 5. Security Analysis

*5.1. Encryption and Decryption Outsource.* In CP-ABE, both data encryption and data decryption are only done by the cloud users. Meanwhile, in MC-ABE, data encryption is done by DO and the cloud server collaboratively, and data decryption is undertaken by DR and the cloud server together. $M$ is masked by DO before it is sent to ESP. DO and authorized DR can get $M$. ESP and DSP can get MM (Masked $M$), but they cannot deduce $M$ from MM.

**Theorem 8.** *The security in encryption and decryption in MC-ABE is not weaker than that of CP-ABE.*

*Proof.* In algorithm $\text{Encrypt}_{\text{ESP}}$, ESP encrypts the access tree $T$ with the parameters $s$, $T$, and MM. Consider

$$\begin{aligned}\widetilde{C} &= M \cdot e(g,g)^{as} \cdot \text{signature} \\ &= M \cdot e(g,g)^{as} \cdot e(g,g)^{\varepsilon v}.\end{aligned} \tag{16}$$

Using PK and $s$, ESP can get $e(g,g)^{as}$; what ESP got is $M \cdot e(g,g)^{\varepsilon v}$.

The encrypted data in CP-ABE is $\widetilde{C} = M \cdot e(g,g)^{as}$; both of $\alpha$ and $s$ are random; let $z = \alpha \cdot s$; $z$ is also random; then $\widetilde{C} = Me(g,g)^{z}$ is equal to $Me(g,g)^{\varepsilon v_k}$. According to security proof in [7], the structure of $\widetilde{C} = M \cdot e(g,g)^{as}$ is secure to prevent the adversary from deducing $M$. Thus, $Me(g,g)^{\varepsilon v_k}$ in our scheme is secure. That is to say, ESP cannot deduce $M$ with $Me(g,g)^{\varepsilon v_k}$, and encryption outsourcing is secure in MC-ABE.

For DSP, it can decrypt CT using SK and get the masked $M = M \cdot \text{signature}$. The information DSP gets is the same as ESP. So, in MC-ABE, data decryption outsourcing is also secure since it is similar to data encryption outsourcing. □

*5.2. Certificate.* From the above statement, the signature is vitally important to the security of our scheme. Since the signature is an item of the certificate, the security of the signature relies on the certificate. Each DR has his unique masked certificate; DR can retrieve his certificate only by his own MValue. In the following, we prove that malicious DR cannot get MCert without the right MValue.

**Theorem 9.** *MCert cannot be decrypted without the right MValue.*

*DR1 has MCert1 = Cert1 · MValue1 = Cert1$e(g,g)^{\theta t_{DR1}}$; DR2 wanted to retrieve Cert1 without $e(g,g)^{\theta t_{DR1}}$.*

*Proof.* DR1 forged $\text{MValue1}' = e(g,g)^{\theta t'_{\text{DR1}}}$, to get Cert1:

$$\begin{aligned}\text{Cert1} &= \frac{\text{MCert1}}{\text{MValue1}'} = \text{Cert1} \cdot \frac{\text{MValue1}}{\text{MValue1}'} \\ &= \text{Cert1} \cdot e(g,g)^{\theta(t_{\text{DR1}} - t'_{\text{DR1}})}.\end{aligned} \tag{17}$$

In other words, if the forged $\text{MValue2}'$ is right, we must have $t_{\text{DR1}} = t'_{\text{DR1}}$ to solve the DL problem. The DL problem is computationally infeasible; thus, MValue is difficult to be forged and MCert cannot be decrypted without the right MValue. □

*5.3. Collusion.* Service providers might collude with each other to combine their information to deduce $M$. In the above statement, ESP and DSP hold similar information to retrieve $M$. If ESP colluded with DSP, the most information they could get is $M \cdot$ signature. We have given the security proof of $M \cdot$ signature in Theorem 8. Thus, MC-ABE is quite qualified for anticollusion.

SSP is a semitrusted server, which stores CT. If SSP colluded with ESP and DSP, it provides no useful information to deduce $M$. So, MC-ABE can defend against collusion among SSP, ESP, and DSP.

*5.4. Revocation.* If a DR is revealed to be malicious, he will be revoked from the authorized user list. We update the signature encrypted in CT; after that, as shown in the following, the revoked DR cannot get authorized data any more:

Revoked signature held by DR: signature = $e(g,g)^{\varepsilon v_k}$.

Updated signature: $\text{signature}' = e(g,g)^{\varepsilon v'_k}$.

Masked $M' = M \cdot \text{signature}' = Me(g,g)^{v'_k}$.

Masked $M'/\text{signature} = Me(g,g)^{\varepsilon v'_k}/e(g,g)^{\varepsilon v_k} = Me(g,g)^{\varepsilon v'_k}/e(g,g)^{\varepsilon v_k} = Me(g,g)^{\varepsilon(v'_k - v_k)}$.

It is the same with the proof of Theorem 9. MC-ABE is secure in revocation.

## 6. Performance Evaluation

In this section, we numerically analyze the communication and computation cost of MC-ABE. We also give the simulation results in detail.

### 6.1. Numerical Analysis

#### 6.1.1. Computation Cost

*Setup*. The setup procedure includes defining multiplicative cyclic group and generating PK and MK that will be used in encryption and key generation. There are four exponentiation operations and one pairing operation in setup procedure. Time complexity of the procedure is $O(1)$. The computation cost has nothing to do with the number of attributes.

*Encrypt*$_{DO}$. In this procedure, DO is responsible for generating signature and masking $M$. Two operations are included in signature computation, which are random number generation and bilinear map computation. And operations performed in mask computation include random number generation and three multiplication operations. Thus, it needs to do two exponentiation operations, two multiplication operations, and one pairing operation for each file. But if more privileges are permitted at the same time, more signatures will be computed. For each privilege, computation cost is fixed, so the total cost is proportional to the number of privileges.

*Encrypt*$_{ESP}$. ESP encrypts the access tree in this procedure. The computation cost is proportional to the number of attributes in the tree. If the universal attributes set in $T$ is $I$ ($|I|$ denotes the total number of attributes in set $I$), for each element in $I$, it needs two exponentiation operations; totally, the computation complexity is $O(|I|)$.

*KeyGen*. This procedure is carried out to generate SK for DR. Computation cost is proportional to the number of

attributes in SK. For each attribute, two pairing operations and one multiplication operation are needed. If the universal attributes set is $S$ ($|S|$ is the total number of attributes in set, $|S| \leq |I|$), the time complexity of SK computation is $O(|S|)$.

*CerGen*. In this procedure, we construct the certificate and mask it. Items in certificate are denoted by DO. TA needs to do one exponentiation operation, one multiplication operation, and one pairing operation. Computation cost is fixed; the computation complexity is $O(1)$.

$Decrypt_{DSP}$. In this procedure, DSP decrypts the ciphertext. The main overhead is incurred at the decryption of every attribute. The cost is proportional to the number of attributes in the access tree. Thus, the complexity is $O(|I|)$.

$Decrypt_{DR}$. In this procedure, DR gets $M$ from the masked $M$ by a divide operation. Thus, the complexity is $O(1)$.

*6.1.2. Storage Cost.* Compared to CP-ABE, more storage cost is incurred in MC-ABE because the certificate and the unique value are introduced. As shown in Table 2, the items in certificates are related to data access privileges, so the storage space of the certificate is proportional to the number of the documents (data). For each DR, one record is kept in mask value table (Table 1). Thus, the storage space for mask value table is proportional to the number of DR. Since the items in mask value table are quite simple, the total storage cost is not heavy.

*6.2. Simulation Results.* To evaluate the performance of MC-ABE, we develop simulation codes based on CP-ABE toolkit [21]. We make a comparison between MC-ABE and other two popular models (CP-ABE and PP-CP-ABE [11]) in four aspects: computation cost for data encryption, computation cost for key generation, computation cost for data decryption, and computation cost for user revocation.

*(1) Computation Cost for Data Encryption*. Most of the computation cost in encryption is incurred for the encryption of the access tree, which is proportional to the number of the leaf nodes. In CP-ABE, data encryption is done by DO. In PP-CP-ABE, data encryption/decryption is outsourced to service providers; the access tree was divided into two parts: one part is encrypted by DO and the other part is encrypted by ESP. In MC-ABE, the access tree is encrypted by ESP. In Figure 6(a), the computation cost of three different schemes is compared. $x$-axis indicates the number of leaf nodes in $T$ (the access tree), and $y$-axis indicates time to encrypt $M$ (computation cost). For $x$, ten values are selected evenly $(10, 20, \ldots, 100)$. For each $x$ value, we run simulation codes 10 times and take the average value of the results as the final result. It is shown that MC-ABE has better performance than the other two ones. In PP-CP-ABE, the number of leaf nodes in DO's subtree will change with different tree division. So, for simplicity, we set the number of DO's subtrees to be half of the number of the whole leaves. As shown in Figure 6(b), we also show confidence interval to assess the results in Figure 6(a) (only results about DO's computation cost in MC-ABE are given, since the results in PP-CP-ABE and CP ABE are consistent with MC-ABE). In Figure 6(b), it is shown that all average results lie in the confidence interval.

Table 2: Impact factor of storage cost.

| | Number of docs | Number of DR |
|---|---|---|
| Certificate storage space | Related | No |
| Mask value storage space | No | Related |

Table 3: Computation cost of key generation (source data of Figure 6(c); the 95% confidence interval assuming random data with normal distribution is shown). Att_num indicates the number of DR's attributes, CI indicates confidence interval, and Ave indicates the average value.

| Att_num | CI | Ave |
|---|---|---|
| 5 | [5.291941, 5.30779403] | 5.2998678 |
| 10 | [11.90953, 11.9336295] | 11.9215787 |
| 15 | [18.54277, 18.5893065] | 18.5660398 |
| 20 | [25.12693, 25.1588600] | 25.1428953 |
| 25 | [31.65159, 31.7310845] | 31.6913405 |
| 30 | [38.26554, 38.3662469] | 38.3158938 |
| 35 | [44.86881, 44.9555189] | 44.9121638 |
| 40 | [51.45481, 51.6333506] | 51.5440794 |
| 45 | [58.04029, 58.1582152] | 58.0992549 |
| 50 | [64.54168, 64.6776467] | 64.6096648 |

*(2) Computation Cost for Key Generation*. Same with simulation about data encryption, we also take the average value of key generation cost as the final result. As shown in Figure 6(c), the average value is very close to lower bound and upper bound of the confidence interval, so we also list source data of the simulation results in Table 3. It shows that all average results lie in the confidence interval, so the simulation result is confident. From the results, we can get that the computation cost will grow with the number of attributes in private key. The algorithm KeyGen is implemented by TA, so there is no cost for DO.

*(3) Computation Cost for Data Decryption*. In MC-ABE, most of the computation cost has been shifted to DSP, so the computation cost of DR is constant. The comparison results are shown in Figure 6(d).

*(4) Computation Cost for User Revocation*. In MC-ABE, user revocation simplified for the signature is introduced. When user revocation happens, the revoked DR's "Revocation" item in mask value table is set as "Y"; his new data request will not be responded to; his former signature encrypted in ciphertext will be also changed. It needs one multiplication operation and one exponentiation operation for the above operations. The simulation results are as shown in Figure 6(e).

## 7. Conclusion

The C-BSN is one promising technology that can change people's healthcare experiences greatly. However, how to keep

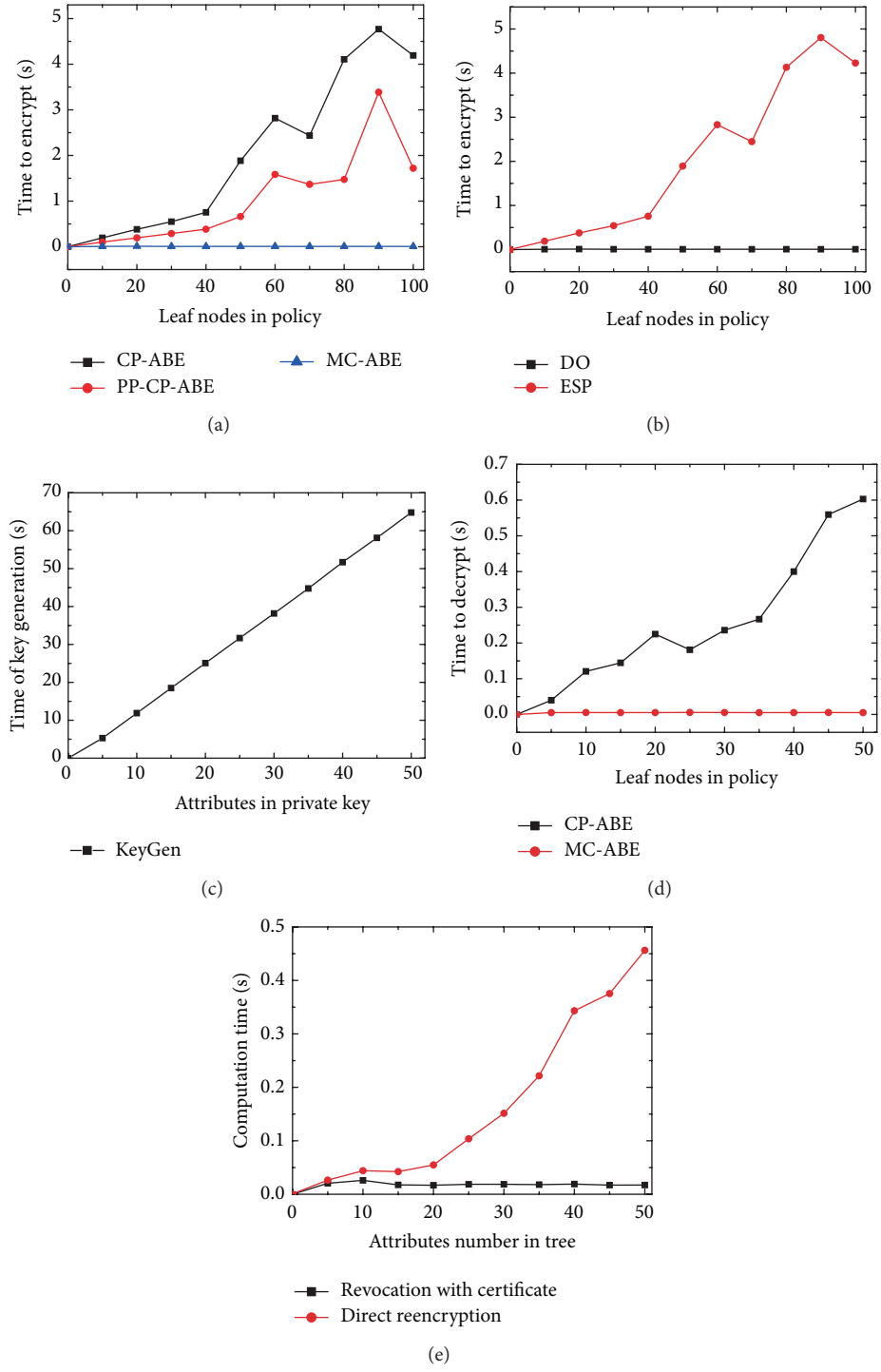


Figure 6: (a) DO's computation cost for data encryption in CP-ABE, PP-CP-ABE, and MC-ABE. In PP-CP-ABE, part of encryption computation is transferred to cloud sever to reduce DO's cost. In MC-ABE, more efforts are made to reduce computation cost undertaken by DO. (b) Computation cost of DO (the 95% confidence interval assuming random data with normal distribution is shown). (c) Computation cost of key generation (the 95% confidence interval assuming random data with normal distribution is shown). (d) Computation cost of DR in CP-ABE and MC-ABE. Similar to ESP in MC-ABE, DSP also undertakes most of the computation in decryption. The cost is proportional to attributes number in private key. (e) Computation cost for user revocation. With the authorization certificate in MC-ABE, revocation cost can be reduced obviously.

data security and data privacy in C-BSN is an important and challenging issue since the patients' health-related data are quite sensitive. In this paper, we propose a novel encryption outsourcing scheme MC-ABE that meets the requirements of data security and data privacy in C-BSN. In MC-ABE, one specific signature is constructed to mask the plaintext; the unique authentication certificate for each visitor is constructed; the third-party trust authority to manage above-mentioned signatures and certificates is also introduced. By security analysis, we prove that MC-ABE can meet the security requirement of C-BSN. And, by performance evaluation, it shows that MC-ABE has less computation cost and storage cost compared with other popular models. In future work, we plan to explore the possibility of improving the scalability of MC-ABE.

## Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

## Acknowledgments

This work is partially supported by Natural Science Foundation of China under Grant no. 61402171, Central Government University Foundation under Grant no. JB2014075, and US Army Research Office under Grant no. WF911NF-14-1-0518.

## References

[1] J. Wan, C. Zou, S. Ullah, C.-F. Lai, M. Zhou, and X. Wang, "Cloud-enabled wireless body area networks for pervasive healthcare," *IEEE Network*, vol. 27, no. 5, pp. 56–61, 2013.

[2] Y. Lu and S.-D. Bao, "Efficient fuzzy vault application in node recognition for securing body sensor networks," in *Proceedings of the IEEE International Conference on Communications (ICC '14)*, pp. 3648–3651, IEEE, Sydney, Australia, June 2014.

[3] M. M. Hassan, B. Song, and E.-N. Huh, "A framework of sensor-cloud integration opportunities and challenges," in *Proceedings of the 3rd International Conference on Ubiquitous Information Management and Communication (ICUIMC '09)*, pp. 618–626, ACM, Suwon, The Republic of Korea, January 2009.

[4] M. Li, W. Lou, and K. Ren, "Data security and privacy in wireless body area networks," *IEEE Wireless Communications*, vol. 17, no. 1, pp. 51–58, 2010.

[5] A. Sahai and B. Waters, "Fuzzy identity-based encryption," in *Advances in Cryptology—EUROCRYPT 2005*, vol. 3494 of *Lecture Notes in Computer Science*, pp. 457–473, Springer, Berlin, Germany, 2005.

[6] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "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)*, pp. 89–98, November 2006.

[7] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute-based encryption," in *Proceedings of the IEEE Symposium on Security and Privacy (S&P '07)*, pp. 321–334, 2007.

[8] S. Yu, C. Wang, K. Ren, and W. Lou, "Achieving secure, scalable, and fine-grained data access control in cloud computing," in *Proceedings of the IEEE INFOCOM*, pp. 1–9, March 2010.

[9] G. Wang, Q. Liu, and J. Wu, "Hierarchical attribute-based encryption for fine-grained access control in cloud storage services," in *Proceedings of the 17th ACM Conference on Computer and Communications Security (CCS '10)*, pp. 735–737, October 2010.

[10] A. Ahmad, M. M. Hassan, and A. Aziz, "A multi-token authorization strategy for secure mobile cloud computing," in *Proceedings of the 2nd IEEE International Conference on Mobile Cloud Computing, Services, and Engineering (MobileCloud '14)*, pp. 136–141, April 2014.

[11] H. Yadav and M. Dave, "Secure data storage operations with verifiable outsourced decryption for mobile cloud computing," in *Proceedings of the International Conference on Recent Advances and Innovations in Engineering (ICRAIE '14)*, pp. 1–5, IEEE, Jaipur, India, May 2014.

[12] Z. Zhou and D. Huang, "Efficient and secure data storage operations for mobile cloud computing," in *Proceedings of the 8th International Conference on Network and Service Management (CNSM '12)*, pp. 37–45, October 2012.

[13] F. Li, Y. Rahulamathavan, M. Rajarajan, and R. C.-W. Phan, "Low complexity multi-authority attribute based encryption scheme for mobile cloud computing," in *Proceedings of the IEEE 7th International Symposium on Service-Oriented System Engineering (SOSE '13)*, pp. 573–577, IEEE, March 2013.

[14] X. Yao, X. Han, and X. Du, "A lightweight access control mechanism for mobile cloud computing," in *Proceedings of the IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS '14)*, pp. 380–385, May 2014.

[15] M. Barua, X. Liang, R. Lu, and X. Shen, "ESPAC: enabling security and patient-centric access control for ehealth in cloud computing," *International Journal of Security & Networks*, vol. 6, no. 2-3, pp. 67–76, 2011.

[16] F. Guo, Y. Mu, W. Susilo, D. S. Wong, and V. Varadharajan, "CP-ABE with constant-size keys for lightweight devices," *IEEE Transactions on Information Forensics and Security*, vol. 9, no. 5, pp. 763–771, 2014.

[17] B. Wang, B. Li, and H. Li, "Public auditing for shared data with efficient user revocation in the cloud," in *Proceedings of the 32nd IEEE Conference on Computer Communications (INFOCOM '13)*, pp. 2904–2912, IEEE, Turin, Italy, April 2013.

[18] A. Beimel, *Secure schemes for secret sharing and key distribution [Ph.D. thesis]*, Technion—Israel Institute of Technology, Haifa, Israel, 1996.

[19] D. Boneh and K. M. Franklin, "Identity-based encryption from the Weil pairing," in *Advances in Cryptology—CRYPTO 2001: 21st Annual International Cryptology Conference, Santa Barbara, California, USA, August 19-23, 2001 Proceedings*, vol. 2139 of *Lecture Notes in Computer Science*, pp. 213–229, Springer, Berlin, Germany, 2001.

[20] H. Cheng, M. Zhang, and D. Feng, "AB-ACCS: a cryptographic access control scheme for cloud storage," *Journal of Computer Research and Development*, vol. 47, no. z1, pp. 259–265, 2010.

[21] J. Bethencourt, A. Sahai, and B. Waters, "CP-ABE toolkit," 2006, http://acsc.cs.utexas.edu/cpabe/.