Available online at www.sciencedirect.com

ScienceDirect

December 2017, 24(6): 49–54
www.sciencedirect.com/science/journal/10058885

The Journal of China Universities of Posts and Telecommunications

http://jcupt.bupt.edu.cn

Access control scheme with attribute revocation for SWIM

Wu Zhijun (✉), Cui Zihan, Wang Caiyun, Lei Jin

School of Electronic Information and Automation, Civil Aviation University of China, Tianjin 300300, China

Abstract

Access control scheme is proposed for System Wide Information Management (SWIM) to address the problem of attribute revocation in practical applications. Based on the attribute based encryption (ABE), this scheme introduces the proxy re-encryption mechanism and key encrypting key (KEK) tree to realize fine-grained access control with attribute revocation. This paper defines the attributes according to the status quo of civil aviation. Compared with some other schemes proposed before, this scheme not only shortens the length of ciphertext (CT) and private key but also improves the efficiency of encryption and decryption. The scheme can resist collusion attacks and ensure the security of data in SWIM.

Keywords SWIM, access control, proxy re-encryption, attribute revocation

1 Introduction

The concept of SWIM originated in the late 1990's in parallel in Europe and the United States. International Civil Aviation Organization (ICAO) defined it as the international aviation information release system in 2005. SWIM is a large-scale distributed system. Its ultimate aim is to build a flexible, unified and efficient information interaction platform where the subsystems of business can interact safely [1]. As the cloud platform and other large-scale network, SWIM faces a lot of security threats. Since the data of SWIM involves sensitive information in the national aviation domain, preventing data leakage and privacy protection are the most important issues. The main reason for these problems is the occurrence of illegal access and unauthorized access. Access control is an important means to solve these problems. In order to ensure that authorized users can access the key information legally in SWIM, Federal Aviation Administration stipulates that the implementation of the SWIM concept seeks to provide quality information to the right people at the right time [2].

With continually deepening of security research on SWIM, the ABE has become a hot research topic in the field of access control. ABE can achieve fine-grained access control because of 'one-to-many' nature [3]. Depending on the decryption policy binding location, ABE is classified into two categories: key-policy ABE (KP-ABE) and Ciphertext-Policy ABE (CP-ABE). In CP-ABE, private keys are integrated with attribute sets and CT are integrated with access policies. Only when the attribute set conform to an access policy, users can decrypt the CT, thus realizing access control. Due to the large number of subjects, the complex types of services and the changes of attributes in practical applications, SWIM puts a higher requirement for access control. Users whose attributes are revoked cannot access the previously authorized resource, thereby implementing the permission revocation [4].

Presently, most of the schemes are focused on supporting the decryption strategies which have richer descriptions rather than attribute revocation. Attribute revocation is divided into indirect revocation and direct revocation. Indirect revocation refers to updating private keys by an authorization agency or introducing a trusted third party to implement attribute revocation. This method is flexible, but the cost of revocation is relatively large. Direct revocation is that data owners embed the attribute

Received date: 03-03-2017
Corresponding author: Wu Zhijun, E-mail: zjwu@cauc.edu.cn
DOI: 10.1016/S1005-8885(17)60241-3

revocation list in the CT when implementing encryption, so the cost of revocation will be small. Boldyreva et al. [5] proposed to set a due date for each attribute. But the disadvantage was that attributes cannot be revoked immediately. Hur et al. [6] designed a scheme which revokes attributes immediately, but its efficiency of CT and key updating were low. Xie et al. [7] constructed an anonymous attribute-based encryption scheme, but attributes revocation was not taken into account. Yu et al. [8] used the version number to label the key and the CT, and introduced the proxy server. This method reduced the workload of the authorization agency greatly, but the encryption and the decryption time were associated with the number of the attributes. Consequently, the efficiency is low.

This paper proposes a new access control scheme based on the KEK tree, which integrates the version number and introduces the proxy re-encryption mechanism. According to the safety and the efficiency analysis, this scheme protects the confidentiality of data, and it has advantages in the time efficiency of encryption and decryption.

2 Definitions

2.1 Access structure

$P=\{P_1,P_2,...,P_n\}$ is a collection of participants and the access structure [9], τ is a nonempty subset of the set P. If τ is monotonous, there is $\forall A,B, A\in\tau$ and $A\subseteq B$, then $B\in\tau$. The attribute set belonging to τ is called the authorized set. Otherwise, the set is the non-authorized set.

Access structure tree is a common method used to represent access structures. Any monotonic access policy can be represented by an access structure tree [10]. In access structure tree, each non-leaf node is a threshold gate, which is described by a certain threshold and all the child nodes of the non-leaf node. If x is an arbitrary node, $n_{\text{child},x}$ represents the child of x and k_x is the threshold value of x, then $0<k_x<n_{\text{child},x}$. When $k_x=1$, x represents an 'OR' gate and when $k_x=n_{\text{child},x}$, x is an 'AND' gate. If x is a leaf node, $k_x=1$, and it is described by a specific attribute.

2.2 Bilinear mapping

G_1 and G_T are cyclic groups whose order is a prime p. If the mapping $e: G_1\times G_1\to G_T$ satisfies the following three properties, then e is a bilinear mapping from G_1 to G_T.

1) Bilinear: $\forall a,b\in Z_p, e(g^a,g^b)=e(g,g)^{ab}$.

2) Non-degenerative: if g is the generator of G_1, $e(g,g)$ is the generator of G_T.

3) Computability: $\forall u,v\in G, e(u,v)$ can be effectively calculated.

2.3 KEK tree

KEK tree which is based on the binary tree of users is shown in Fig. 1.

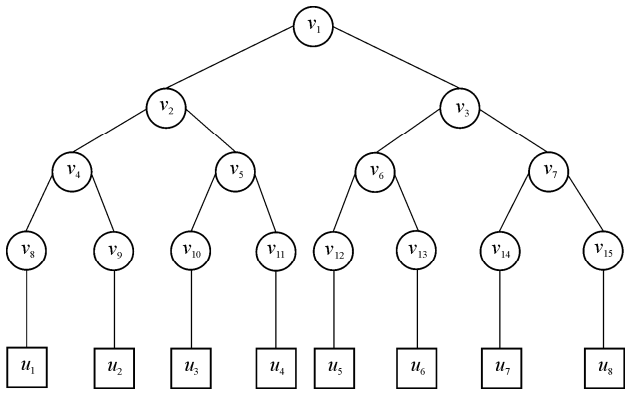

Fig. 1 Example of KEK tree

Users are distributed on leaf nodes which represent a collection of all users in the system. Each node v_j is given a random number K_j. It is obvious that there is a path between a leaf node and the root node, and the set of nodes on this path is defined as the path key $K_{\text{PK}}(t)$. For example, user 6 corresponds to the path key $K_{\text{PK}}(6)=\{K_{13},K_6,K_3,K_1\}$. Minimum coverage element [11] is the minimum node set which can cover all users of the attribute group G_t, $1\leqslant t\leqslant T$, in the KEK tree. For example, the minimum coverage element of $G_t=\{u_1,u_2,u_3,u_6,u_8\}$ is $K(G_t)=\{v_4,v_{10},v_{13},v_{15}\}$.

3 The access control scheme with attribute revocation

3.1 Attribute definition

In order to achieve attribute based access control (ABAC), the attributes of SWIM are defined.

ABAC classifies attributes into three categories: subject,

resource and environment. Since the users and the service of SWIM are mainly from government departments, airlines and related institutions, this paper uses Air Traffic Management Bureau (ATMB), airlines and airports as the main research objects. Considering that the specific business processes of civil aviation, attributes of SWIM are divided by region, function and other factors.

Combined with the characteristics of both ABAC and SWIM, attributes are defined as follows.

Table 1 Attribute definition

Attribute	Category	Value
Area	Subject/Resource	Specific cities (Beijing/Shanghai/ Shenyang/…)
Organ	Subject/Resource	Specific institutions
Position	Subject	Specific position of subject (aviation control/dispatch/…)
Type	Resource	Information category (flight plan/weather information/…)
Level	Subject/Resource	Level of subject/Resources (Lv.1/Lv.2/Lv.3/Lv.4)
Time	Environment	Current time
Status	Environment	Current network status (normal/ abnormal)

We can combine the different attributes in Table 1 to provide a detailed description of each individual subject, resource, and environment.

This paper chose the attributes involved in Table 1 as representatives to implement access control. Based on CP-ABE, this paper drew up access policy for resource. For example, resource R is the Lv.3 aeronautical meteorological information of Shanghai which is distributed by Civil Aviation Administration of China (CAAC) East China Regional Administration. The policy of resource R is that an air controller of CAAC East China Regional Administration whose level is greater than 2 can access R when the status of network is normal. S_1 is the Lv.3 air controller of CAAC East China Regional Administration who works in Shanghai and S_2 is the Lv.2 air controller of CAAC North China Regional Administration who works in Tianjin. Obviously, S_1 can access the resource R in the case of normal network, but S_2 cannot.

3.2 System model

The implementation of this program covers five parts: data server, authorization agency, users, data owner and re-encrypt server. The system model is given in Fig. 2.

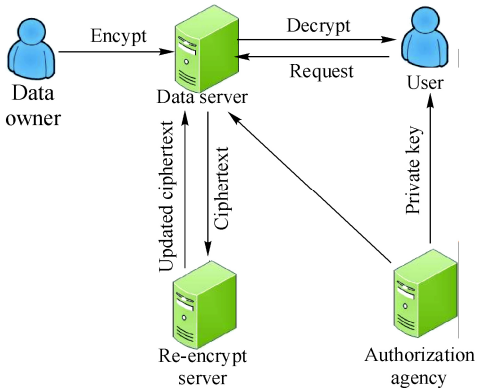

Fig. 2 System model

1) Authorization agency: this agency is responsible for establishing system, generating public keys (PKs) and master secret keys (MSKs), initializing the version number. The agency generates a private key based on the set of user's attributes and the MSK, then it sends the key to users. In order to ensure the security of transmission, the transmission between the authorization agency and users can use the PK system.

2) Data owner: draw up a corresponding access policy based on the access structure. Use KeyGen algorithm to generate a PK according to the access policy. Encrypt plaintext to obtain CT by the PK, and then upload the CT to the data server.

3) User: access the data server and obtain CT, then decrypt the CT by the private key that authorization agency distributes.

Considering the attribute revocation, this scheme introduces two steps: CT updating and private key updating.

1) Authorization agency: according to the updated attribute list, users get an attribute group key. Use ReKeyGen algorithm to generate a new private key based on the attribute group key. The version number of private key has to plus one.

2) Re-encrypt server: use ReEncrypt algorithm to re-encrypt the CT to get new CT. The version number of CT has to plus one.

3) Users: if the version number of the CT is consistent with the version number of the private key, users can decrypt the information. Otherwise, the information cannot be decrypted.

3.3 Special algorithm

This method can achieve fine-grained attribute

revocation and effective access control to guarantee the security of SWIM. This method is composed of six algorithms which are defined as follows:

$L=\{\theta_1,\theta_2,...,\theta_t\}$ is a collection of user's attributes. $\Omega=\{\theta_i, i\in[1,n]\ n\leqslant t\}$ is a collection of all possible attributes. $A_i=\{v_{i,1},v_{i,2},...,v_{i,n}\}$ is a collection of possible values for θ_i. v_{i,n_i} represents the n_ith value of θ_i; $\Gamma=\{G_k\}_{\theta_k\in L}$ is a collection of attribute groups. G_k is the collection of users with attribute θ_k.

1) Setup (k): according to a security parameter k, the setup algorithm chooses a bilinear group G_1 of prime order with generator g and a bilinear map function $e: G_1\times G_1\to G_T$. Then this algorithm selects a random exponent $\alpha\in Z_p$, $t_{i,j}\in Z_p, (i\in[1,n], j\in[1,n_i])$, n_i is the total number of all values of θ_i, and computes $E=e(g,g)^{\alpha}$. The PK is published as:

$$S_{\mathrm{PK}}=(e,g,E,T_{i,j}=g^{t_{i,j}};\quad i\in[1,n], j\in[1,n_i]) \tag{1}$$

The master secret key is published as:

$$S_{\mathrm{MSK}}=(\alpha,t_{i,j}(i\in[1,n], j\in[1,n_i])) \tag{2}$$

It initializes the version number ($n_{\mathrm{ver}}=1$).

2) KeyGen (MSK, L): the key generation algorithm takes a set of attributes L and a MSK as input. This algorithm computes $T_{i,j}{}^{\gamma}$, $\gamma\in Z_p$ for each attribute $v_{i,j}\in L$ to get a secret key (SK). The key is published as:

$$S_{\mathrm{SK}}=(n_{\mathrm{ver}},h=g^{\alpha+\gamma},\forall v_{i,j}\in L: T_{i,j}{}^{\gamma},L) \tag{3}$$

3) Encrypt (PK, m, τ): the encryption algorithm encrypts message m under the access policy τ. Enter the PK and select a random exponent $s\in Z_p$. Then it can generate CT and upload CT to the data server.

$$S_{\mathrm{CT}}=(n_{\mathrm{ver}},C'=mE^{s},C=g^{s},\tau,\prod_{v_{i,j}\in\tau}T_{i,j}^{s}) \tag{4}$$

4) Decrypt (CT, SK): if the set of attributes satisfies the access policy, this algorithm selects the smallest set of access structures $\tau'(\tau'\subseteq\tau)$ and uses it to decrypt CT.

$$\frac{C'e\left(g^{\gamma},C\prod_{v_{i,j}\in\tau}T_{i,j}^{s}\right)}{e\left(g^{s},h\prod_{v_{i,j}\in\tau'}T_{i,j}^{\gamma}\right)}=\frac{me(g,g)^{\alpha s}e(g^{\gamma},g^{s}g^{sp})}{e(g^{s},g^{\alpha+\gamma}g^{\gamma q})}=$$

$$m\frac{e(g,g)^{\alpha s}e(g,g)^{s\gamma}e(g,g)^{\gamma sp}}{e(g,g)^{\alpha s}e(g,g)^{s\gamma}e(g,g)^{\gamma sq}}=$$

$$m;\quad p=\sum_{v_{i,j}\in\tau}t_{i,j},q=\sum_{v_{i,j}\in\tau'}t_{i,j} \tag{5}$$

5) ReEncrypt: if a user loses attribute θ_k, the re-encrypt algorithm updates the CT by a random exponent $s'\in Z_p$ and the attribute group key $L_{\theta_k}\in Z_p$ to get new CT.

$$S'_{\mathrm{CT}'}=(n_{\mathrm{ver}},C'=mE^{s+s'},C=g^{s},\tau,C''=g^{s+s'},$$
$$C_0=E_k(L_{\theta_k})_{k\in K(G_k)},\prod_{v_{i,j}\in\tau}T_{i,j}^{s}) \tag{6}$$

The version number of CT will plus one.

6) ReKeyGen: according to the property of the KEK tree, users with the attribute θ_k can obtain the attribute group key L_{θ_k} by decrypting C_0. Then this algorithm selects a random exponent $\beta\in Z_p$ and generate a new key SK′.

$$S'_{\mathrm{SK}'}=(n_{\mathrm{ver}},h=g^{\frac{\alpha}{\beta+\gamma}},K=g^{\alpha+\beta},L_{\theta_k},\forall v_{i,j}\in L: T_{i,j}{}^{\gamma}) \tag{7}$$

The version number of private key will plus one.

7) Decrypt: if the CT is updated, this algorithm will compare the version number of CT to the version number of private key. If version numbers are different, users cannot decrypt the information. On the contrary, users can use the private key to decrypt CT to get m.

$$\frac{C'e\left(g^{\beta},C''h\prod_{v_{i,j}\in\tau}T_{i,j}^{s}\right)}{e\left(K,C''C\prod_{v_{i,j}\in\tau'}T_{i,j}^{\gamma}\right)}=me(g,g)^{\alpha(s+s')}e(g^{\beta},g^{s+s'}g^{\alpha/(\beta+\gamma)}\cdot$$

$$g^{sp})\left[e(g^{\alpha+\beta},g^{s+s'}g^{s}g^{\gamma q})\right]^{-1}=$$

$$m\frac{e(g,g)^{\alpha(s+s')}}{e(g,g)^{\alpha(s+s')}}=m \tag{8}$$

4 Scheme analysis

4.1 Safety analysis

When a user's attribute is revoked, it represents that the user does not belong to this attribute group. According to the construction of the KEK tree, when the CT is updated, the user no longer obtains the attribute group key by decrypting C_0, so the private key cannot be updated. This method controls the access rights of users effectively to ensure the security of systems. Conversely, when a user adds an attribute, the system compares the version number of the private key with the version number of CT. If the numbers are same and the user's attributes match the access policy, the CT can be decrypted. In other cases, users cannot obtain the message. In addition, this scheme introduces the proxy re-encryption mechanism. As the work of updating CT is carried out by the re-encryption server, this mechanism reduces the workload of data owners.

Anti-conspiracy attack: in this scheme, the private key is randomized by a random exponent γ which the setup algorithm distributes to each user. If private keys need to be updated, the ReKeyGen algorithm randomizes the key by a random exponent β too. For the above reasons, unauthorized users whose joint attributes conform to the access policy cannot conspire to decrypt the CT.

4.2 Applicability analysis

SWIM is a large-scale distributed network essentially, so it is distributed, heterogeneous and dynamic. Aiming at these characteristics of SWIM, this paper presents a fine-grained access control scheme based on ABE. In order to apply to SWIM, this scheme made the following design.

This scheme adopts attributes as the basis of authorization, and updates the private key according to the attribute group key correspondingly. Due to the large number of users in SWIM, the method of obtaining attribute group keys by attributes can efficiently implement multiple grouping for users. This scheme dynamically generates authorization policies according to the attributes of users and resource. It is platform-independent and conforms to heterogeneity and dynamics. Users access external domains through inter-domain protocols, so intra-domain access policy can better ensure the integrity of local data and the assignment of server permissions is more decentralized.

4.3 Performance analysis

This section provides computational complexity analysis and experimental measurements of this scheme. The test machine is an Intel Pentium 3 GHz desktop with 4 GB memory running Linux 14.0.3.

$|G_1|$, $|G_T|$ and $|Z_P|$ represents the byte length of elements in the set G_1, G_T and Z_P respectively. n is the total number of attributes in the system. G_1, G_T, C_e respectively represents the computational complexity of group G_1, the computational complexity of G_T and the complexity of bilinear pairings. r_1 is the number of attributes associated with the length of keys, which varies with different users. $N' = \sum_{i=1}^{n} n_i$, n_i is the total number of all values of θ_i. r_2 is the number of attributes related to the length of private keys.

The scheme not only realizes the attribute revocation, but also ensures that the length of CT is short. The size of each relevant parameter is shown in Table 2.

Table 2 Comparison of related parameters

Scheme	Public key size	Master secret key size	Secret key size	Ciphertext size
In Ref. [12]	$(2N'+1)\|G_1\|+\|G_T\|$	$(2N'+1)\|Z_P\|$	$(3n+1)\|G_1\|$	$(2N'+1)\|G_1\|+\|G_T\|$
In Ref. [13]	$2\|G_1\|+\|G_T\|$	$\|G_1\|$	$(1+n+r_2)\|G_1\|$	$(1+nr_1)\|G_1\|+\|G_T\|$
Our scheme	$(N'+1)\|G_1\|+\|G_T\|$	$(N'+1)\|Z_P\|$	$(1+n)\|G_1\|$	$2\|G_1\|+\|G_T\|$

The algorithm execution time depends on the exponential operation and the operation of the linear pair. Efficiency comparison between this method and other methods is shown in Table 3.

Table 3 Comparison of computation complexity

Scheme	Encrypt	Decrypt
In Ref. [12]	$(2N'+1)G_1+2G_T$	$(1+3n)C_e+(3n+1)G_T$
In Ref. [13]	$(1+3r_1n)G_1+2G_T$	$(1+r_1+n)C_e+(3r_1-1)G_1+3G_T$
Our scheme	$2G_1+G_T$	$2C_e+2G_1$

According to the comparison of Table 3, this scheme can reduce the number of calculation and computational complexity.

Time cost is an important factor in the algorithm evaluation. Under the premise of the same length of each attribute, the experiment was carried out with subjects and access policies, which had different number of attributes. Then the encryption time, the decryption time and the key generation time were measured in the experiment. As the number of attributes changes, Fig. 3 indicates the variety of the key generation time. Fig. 4 shows the variety of the encryption time and the decryption time.

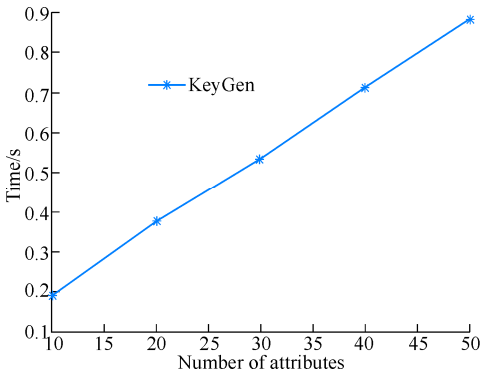

Fig. 3 Relation between key generation time and number of attributes

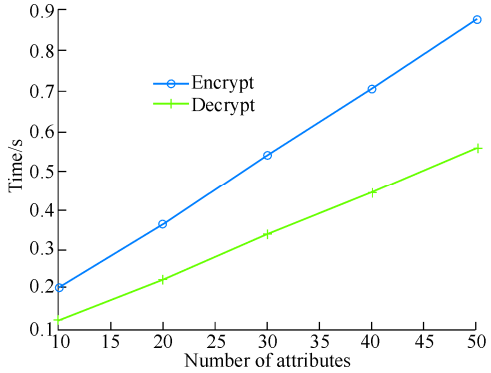

Fig. 4 Relation between time cost and number of attributes

As shown in Fig. 3, the key generation time increases linearly with the number of attributes. When the number of attributes is 50, the key generation time is still less than 1 s. Actually, a precise lock on a user or a kind of resource usually requires less than 10 attributes in SWIM.

As shown in Fig. 4, the encryption and decryption time increases linearly with the number of attributes respectively. When the number of attributes is 50, both of the encryption time and the decryption time are less than 1 s. In fact, the number of attributes involved in an access policy of a single resource is often less than 50 in SWIM.

A complete authorization process includes setup algorithm, key generation algorithm, encryption and decryption algorithm. After several tests, the average time of the setup algorithm is 0.053 2 s. According to the data in Figs. 3 and 4, the time cost of one authorization process is not more than 2 s for the users with 10 attributes and the access policies involving 50 attributes. Therefore, the time cost of this scheme is acceptable in SWIM.

5 Conclusions

In order to solve the problem of the attribute revocation in SWIM, this paper proposes a scheme which combines the KEK tree with the version number and introduces the proxy re-encryption mechanism. In the scheme, the KEK tree is used to provide the updating information of the key for users whose attributes are not revoked. Then users decrypt the attribute group key to update their private keys, thus they can decrypt the updated CT. The scheme analysis shows that this scheme not only realizes attribute revocation, but also has advantages in encryption and decryption time. At the same time, it can shorten the length of keys and CT. In addition, as the number of users and attributes increase, the advantages of CT storage space and efficiency will be more obvious. The applicability analysis shows that this scheme is suitable for distributed systems and meets the basic requirements of access control in SWIM. In the follow-up work, we try to apply this method to the actual platform. Meanwhile, reducing the leakage of attributes and access structures in delivery can be a key issue in further research.

Acknowledgements

This work was supported by the National Natural Science Foundation of China and Civil Aviation Administration of China Joint Fund Project (U1533107), the Major Program of Natural Science Foundation of Tianjin (17JCZDJC30900).

References

1. Meserole J S, Moore J W. What is system wide information management (SWIM)? IEEE Aerospace and Electronic Systems Magazine, 2007, 22(5): 13–19
2. Stephens B. System-wide information management (SWIM) demonstration security architecture. Proceedings of the IEEE/AIAA 25th Digital Avionics Systems Conference, Oct 15–19, 2006, Portland, OR, USA. Piscataway, NJ, USA: IEEE, 2006: 12p
3. Wu Y D, Wei Z, Deng R H. Attribute-based access to scalable media in cloud-assisted content sharing networks. IEEE Transactions on Multimedia, 2013, 15(4): 778–788
4. Su J S, Cao D, Wang X F, et al. Attribute based encryption schemes. Journal of Software, 2011, 22(6): 1299–1315 (in Chinese)
5. Boldyreva A, Goyal V, Kumar V. Identity-based encryption with efficient revocation. Proceedings of the 15th ACM Conference on Computer and Communications Security (CCS '08), Oct 27–31, 2008, Alexandria, VA, USA. New York, NY, USA: ACM, 2008: 417–426
6. Hur J, Noh D K. Attribute-based access control with efficient revocation in data outsourcing systems. IEEE Transactions on Parallel and Distributed Systems, 2011, 22(7): 1214–1221
7. Xie X X, Ma H, Li J, et al. An efficient ciphertext-policy attribute-based access control towards revocation in cloud computing. Journal of Universal Computer Science, 2013, 19(16): 2349–2367
8. Yu S C, Wang C, Ren K, et al. Attribute based data sharing with attribute revocation. Proceedings of the 5th ACM Symposium on Information, Computer and Communications Security (ASIACCS'10), Apr 13–16, 2010, Beijing, China. New York, NY, USA: ACM, 2010: 261–270
9. Hur J. Attribute-based secure data sharing with hidden policies in smart grid. IEEE Transactions on Parallel and Distributed Systems, 2013, 24(11): 2171–2180
10. Wang P P, Feng D G, Zhang L W. CP-ABE scheme supporting fully fine-grained attribute revocation. Journal of Software, 2012, 23(10): 2805–2816 (in Chinese)
11. Ma H, Bai C C, Li B, et al. Attribute-based encryption scheme supporting attribute revocation and decryption outsourcing. Journal of Xidian University, 2015, 42(6): 6–10 (in Chinese)
12. Guo L F, Lu B. Eiffcient proxy re-encryption with keyword search scheme. Journal of Computer Research and Development, 2014, 51(6): 1221–1228 (in Chinese)
13. Wang J X, Zhang M, Chen Q. An efficient attribute based encryption with attribute revocation. Journal of Computer Applications, 2012, 32(S1): 39–43 (in Chinese)

(Editor: Wang Xuying)