International Journal of Security and Its Applications
Vol.8, No.4 (2014), pp.287-296
http://dx.doi.org/10.14257/ijsia.2014.8.4.26

Secure DRM Scheme Supporting Dynamic Authorization Using Attribute-Based Encryption

Fu Jingyi[1,2,3], Ma Zhaofeng[1,2,3], Huang Qinlong[1,2,3] and Yang Yixian[1,2]

[1]*Information Security Center, Beijing University of Posts and Telecommunications, Beijing, China*
[2]*National Engineering Laboratory for Disaster Backup and Recovery, Beijing University of Posts and Telecommunications, Beijing, China*
[3]*Beijing National Security Science and Technology Co., Ltd, Beijing, China*
fujingyi@bupt.edu.cn, mzf@bupt.edu.cn, longsec@bupt.edu.cn, yxyang@bupt.edu.cn

Abstract

Content abusing is increasingly common with the rapid development of the Internet, which damages the benefits of copyright owners, how to effectively prevent the abuse of digital contents is a big challenge. In this paper, we propose a secure DRM scheme supporting dynamic authorization, which first encrypts content with content encryption key (CEK), and then protects CEK based on distributed attribute-based encryption. At last encrypted CEK will be packaged and distributed with encrypted content, which eliminates independent key management and reduces the burden on the DRM server. In our scheme, user is labeled with a set of attributes and CEK is associated with access policy, only the user whose attributes satisfy access policy of content can recover CEK, which also achieves fine-grained access control. Moreover, to improve Muller et al.'s DRM scheme, our scheme achieves dynamic authorization by adding action control in the license, while the action control is related to user's payment. Hence, when the user accesses the content, attributes comparison and license checking must be enforced, which lowers the trust required in DRM client. The analysis and comparison show our scheme is efficient and secure.

Keywords: *Digital Rights Management, Attribute-Based Encryption, Dynamic Authorization, Access Control*

1. Introduction

Nowadays, unlimited copy, illegal edition and arbitrary distribution of digital contents are more and more flooding on the Internet, so digital rights management (DRM) system [1] emerges as the times require. It protects content integrity, availability, confidentiality, reliability and guarantees the rights of copyright owners through a series of ways, such as content encryption, digital watermark, license distribution *etc.*, However, the combination of content encryption and license distribution is the main way most current DRM schemes adopt.

However, current DRM schemes also involve a broad range of problems and limitations. One of them is fine-grained access control. The traditional DRM schemes achieve fine-grained access control through licenses, in which content providers can upload contents to DRM service provider who is responsible for setting access permission and distributing licenses, and user can access contents with the licenses. However, content providers who are the owner of contents cannot restrict user's rights to the contents. A new cryptography called

ISSN: 1738-9976 IJSIA

ciphertext-policy attribute-based encryption (CP-ABE) was proposed to solve the problem that the data owner's fine-grained control over access to his own data which is stored at third party. Based on the CP-ABE, the data owner can define the access policy of his data, and the users whose attributes satisfy the access policy can access the data. On this basis, there are some researches on DRM combined with CP-ABE, for example Muller et al. proposed the first DRM scheme to adopt CP-ABE to achieve fine-grained access control [2].

Another important problem which is associated with security is over-dependence on trusted DRM client. Nowadays, most of the DRM schemes must depend on the fully trusted DRM client. In these schemes, contents are protected with a master key which is encrypted and distributed along with license. Only with user's secret key can DRM client decrypt the encrypted master key, thus DRM client must hold the user's secret key, which means DRM client must be operated in fully trusted environment. But the fact is that there may exist malicious and unauthorized users who are hard to discover and almost impossible to prevent. Thus solving the problem of over-dependence on trusted DRM client is the key to deploy DRM in the actual environment.

In this paper we propose a secure DRM scheme supporting dynamic authorization, in which content is encrypted with content encryption key (CEK), and CEK is further protected and distributed with encrypted content, thus independent key management is not needed and the burden on the DRM server is reduced. Moreover, in our scheme user is labeled with a set of attributes and CEK is associated with access policy, thus only the user whose attributes satisfy access policy can recover CEK and then decrypt the encrypted content, which achieves fine-grained access control. In addition, we improve Muller, *et al.*, scheme [2] by adding action control which is related to user's payment in the license. Thus license server can grant users different rights to the same content, which achieves dynamic authorization. In our scheme, attributes comparison and license checking must be enforced before content decryption, which provides protection against malicious users to a certain extent. Hence our scheme has a better robustness and can reduce the trust required in DRM client.

The rest of this paper is organized as follows. In Section 2 we introduce related work. In Section 3 we present the encryption algorithms involved in our scheme, and we describe the framework of our scheme in Section 4. Section 5 gives the detailed processes of our model. Section 6 gives security and performance analysis, and shows the comparison with other DRM schemes. Finally, we conclude in Section 7.

2. Related Work

Currently public-key cryptography, proxy re-encryption (PRE), and homomorphic encryption (HE) are widely used techniques in DRM systems to ensure data security. Ma, *et al.*, presented a DRM sharing model based on PRE [3], which is the first model for the content sharing between different users' devices who are not family members. Petrlic, *et al.*, proposed a DRM scheme for cloud computing [4], which combines secret sharing and HE to achieve a privacy-friendly manner. Although these schemes are able to ensure data security, they cannot support fine-grained access control and limit a set of users to access encrypted data.

To solve these problems, Sahai, *et al.*, made some initial steps [5]. They introduced attribute-based encryption (ABE) as a new way for encrypted access control. In their scheme both the secret key of user and ciphertext are viewed as a set of descriptive attributes, and the "set overlap" distance metric is the only standard to match them, which means people can decrypt ciphertext if they are up to standard. But this method is not applicable for fine-grained control required system since the standard cannot be expressed only by several attributes accurately.

The ABE is divided into two directions that are key-policy ABE (KP-ABE) and CP-ABE [6-9]. In order to satisfy fine-grained access control requirement, Goyal, *et al.*, developed a fine-grained cryptosystem called KP-ABE [8], in which ciphertexts are labeled with a set of attributes and private keys are associated with a tree-access structure that control which ciphertexts a user is able to decrypt. Bethencourt, *et al.*, proposed an encryption scheme called CP-ABE [9], in which access structures are associated with ciphertexts, and attributes are used to describe a user's credential. In their scheme encrypted data can be kept confidential even if the storage server is untrusted. Currently, the ABE technique is widely adopted, for example Wang, *et al.*, were the first to improve PHR sharing mechanism with the idea of CP-ABE [10], which can provide privacy protection and fine-grained access control.

Furthermore, some researchers have focused on DRM combined with ABE. Dutta et al. presented a multi-level multi-distributor based DRM architecture which facilitates client mobility [11], and proposed key management mechanism using identity-based encryption (IBE) and ABE. Muller, *et al.*, proposed a new DRM architecture [2], which utilizes two-steps enforcement process to enable strong security even in the case of a compromised DRM viewer by using ABE. In their model, access policy is extracted from license which is prepared to corresponding media and packaged with encrypted media, and finally distributed to user by media distributor, which causes burden on server once the number of users is increasing and can't support dynamic authorization.

Compared with Muller, *et al.*, scheme, our proposed DRM scheme supports dynamic authorization. In our scheme access policy is decided by content provider, which is just the basic requirement to access the encrypted content. To achieve dynamic authorization, we separate license distribution from content distribution and add action control in the license. The analysis indicates our scheme with good practice in the protection of content is efficient and secure.

3. Preliminaries

The adopted encryption algorithm in our scheme involves basic CP-ABE [8, 12] and distributed attribute-based encryption (DABE) which is proposed by Muller, *et al.*, [13]. The used notations are shown in Table 1.

Table 1. Notations in Proposed Scheme

Notation	Description
K_P, K_M	System public key/master key
B	Attribute authority
P, S	Public key/Secret key
R	Attribute secret keys
A	User's Attributes
A	An attribute
T	Access policy
G	Absent attributes
I	Identity
U	User
K_E	Content encryption key
L	License
Q_L	License request

3.1. CP-ABE Algorithms

The basic CP-ABE scheme consists of four algorithms:

Setup(k): The algorithm takes a security parameter k as input and outputs K_P and K_M.

KeyGen(K_M, A): The algorithm takes K_M and A , then outputs the attributes secret key R_U to user.

Encrypt(K_P, M, T): The algorithm inputs K_P, plaintext M, and T over the set of attributes. The algorithm encrypts M and produces a ciphertext C.

Decrypt(K_P, C, R_U): The algorithm takes K_P, C and R_U as input. If the user's attributes A satisfy T , then the algorithm returns a plaintext M.

3.2. DABE Algorithms

The DABE scheme is based on CP-ABE, which allows an arbitrary number of authorities to independently maintain attributes. To improve the reliability of our scheme, we adopt DABE scheme in which Setup algorithm is the same as basic CP-ABE, but the other algorithms are described as follow:

UserReg(K_P, K_M, I_U): The algorithm takes K_P, K_M, and identity of user I_U as input, then outputs P_U and S_U.

AuthorityReg(K_P, I_B): It is executed by the attribute authority, and the algorithm takes K_P, the identity of attribute authority I_B as input, and outputs secret key of attribute authority S_B.

PublicKeyGen(K_P, A, S_B): The algorithm takes K_P, A and S_B as input, and outputs attribute public key P_A if attribute authority has responsibility, otherwise *NULL*.

SecretKeyGen(K_P, A, S_B, I_U, P_U): It takes K_P, A, S_B, I_U and P_U as input, then outputs the attribute secret key R_U which is associated to I_U if attribute authority has responsibility of A, otherwise *NULL*.

Encrypt(K_P, T , P_A, M): The algorithm takes K_P, plaintext M, T and P_A, then outputs ciphertext C.

Decrypt(K_P, T , S_U, R_U, C): The algorithm decrypts C with K_P, T , S_U, and R_U if A are sufficient to T , then outputs plaintext M, otherwise it outputs *NULL*.

4. Framework of Proposed Scheme

We propose the following DRM framework. It involves eight parties, and its relationships are described in Figure 1.

Central authority: It manages registered users and attributes authorities, and generates the key pair for users and secret key for attribute authority whose attributes are also assigned by central authority.

Attribute authority: The duty of attribute authority is creating attribute public keys and attribute secret keys of user. Each attribute authority is responsible for numbers of attributes, and can generate attribute secret keys of user within the scope of rights.

Content server: We define it as a trusted center, and it encrypts content with random CEK that is further encrypted based DABE, and then packages encrypted content according to the standard format, and provides it to content platform.

License server: It generates license to user through business platform after successful payment. The license contains attribute secret keys of user which are all encrypted by public key of user and action control (such as expiration date, usage count *etc.*,). The license can be expressed with rights expression language.

Content platform: It is a visual webpage platform through which users are able to query and download interesting contents, but these contents are all encrypted.

Business platform: It is responsible for processing the payment and applying for the license to the user.

Content provider: They provide content to content server to encrypt content, and also provide the usage control to content server to generate the access policy for the content. The communication channel between the content provider and content server is ensured by certain existing security protocols such as TLS.

User: They can access the encrypted content with DRM client that is responsible for parsing license and decrypting the content. Before content decryption, DRM client must compare attributes of user with access policy which is embedded in the head of packaged content, and check action control in the license. Only these two steps are both satisfied, can users get the content.

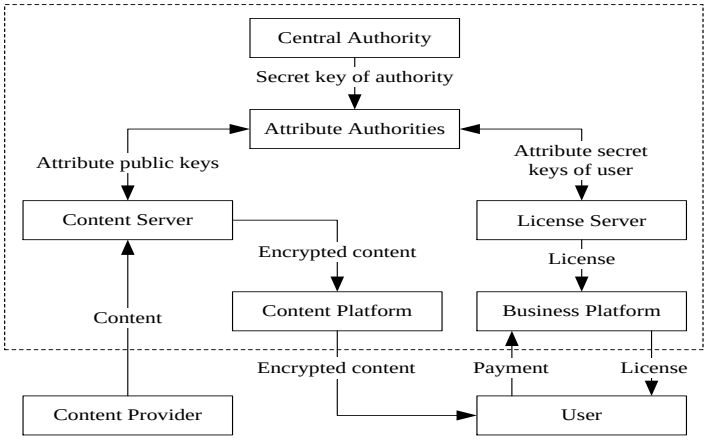

Figure 1. Components of DRM System

5. Details of the Proposed Framework

The proposed scheme consists of initialization phase, content encryption phase, license acquisition phase and content decryption phase.

5.1. Initialization Phase

The initialization phase includes attribute authority registration and user registration. The central authority takes a security parameter as the input and generates K_P and K_M, which will be used in attribute authority registration and user registration.

For attribute authority registration, each attribute authority is associated with an identity I_B. The central authority runs Authority Reg. algorithm to output S_B.

$$S_B = AuthorityReg(K_P, I_B)$$

And the central authority then assigns a number of attributes to attribute authority which will generate public keys for these attributes.

$$P_{A,i} = PublicKeyGen(K_P, A_i, S_B)$$

where $P_{A,i}$ is the public key of attribute A_i.

For user registration, central authority runs UserReg algorithm to output P_U and S_U. P_U will be used by attribute authorities to generate R_U, while S_U will be used to decrypt content and kept secretly by the user.

$$(P_U, S_U) = UserReg(K_P, K_M, I_U)$$

5.2. Content Encryption Phase

Content encryption phase is showed in Figure 2. The steps are as follow:

Step1: Content provider sends plaintext M and usage control to content server in a secure channel.

Step2: Content server generates K_E randomly and $\mathtt{T}$ over the set of attributes based on usage control.

Step3: Plaintext M is then encrypted by K_E using symmetric encryption algorithm.

$$m_1 = E(K_E, M)$$

Step4: In order to encrypt K_E, content server sends $\mathtt{T}$ and signatures to request for P_A from attribute authorities.

Step5: Each attribute authority checks whether it has authority over the attributes of $\mathtt{T}$ when obtaining this request, then returns $P_{A,i}$.

Step6: The attribute authorities return P_A to content server.

$$P_A = (P_{A,1}, \ldots, P_{A,n})$$

where n is the number of attributes in the $\mathtt{T}$.

Step7: Content server encrypts K_E with K_P, $\mathtt{T}$, P_A, using the Encrypt algorithm.

$$m_2 = Encrypt(K_P, \mathtt{T}, P_A, K_E)$$

Step8: Content server packages the encrypted content according to the standard format.

$$C = (\mathtt{T}, m_1, m_2)$$

Step9: Content server uploads the packaged content to content platform.

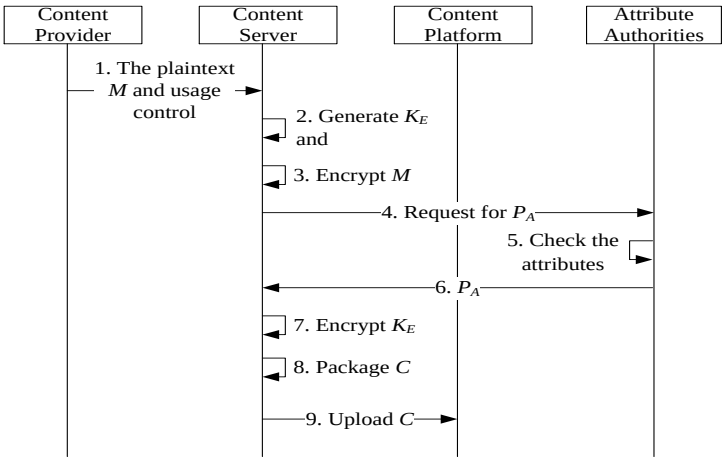

Figure 2. Content Encryption Process

5.3. License Acquisition Phase

License acquisition phase is showed in Figure 3. The steps are as follow:

Step1: When user wants to access the content which is downloaded from content platform, DRM client compares the $\mathtt{A}$ with $\mathtt{T}$.

Step2: If there is a mismatch attribute, user will be notified the content cannot be used, and if there are absent attribute or there is no appropriate license of this content, DRM client will send identities of absent attributes $I_G = (I_{G,1},..., I_{G,t})$ to business platform, and guide the user to business platform to pay for the absent attributes and rights of this content.

Step3: After successful purchase, business platform send license request Q_L including user rights V_U etc. and the signature to license server to apply for a license for the user.

$$Q_L = E(P_L, I_G, I_U, I_C, V_U)$$

Step4: License server first verifies the signature, if it is the case, then decrypts Q_L with S_L to get I_G, I_U, I_C, V_U.

$$(I_G, I_U, I_C, V_U) = D(S_L, Q_L)$$

Step5: License server sends a message including I_G, I_U to attribute authorities in a secure channel to request for R_U and absent attributes G.

Step6: Attribute authorities first verify the signature with P_L, if it is the case, they will apply P_U according to I_U from central authority and G according to corresponding I_G, then check which attributes are their responsibility, then G and Z_U in their scope of rights will be output, otherwise *NULL*.

$$R_{U,i} = SecretKeyGen(K_P, G_i, S_B, I_U, P_U)$$
$$Z_{U,i} = E(P_U, R_{U,i})$$
$$Z_U = (Z_{U,1},..., Z_{U,t})$$

where G_i is the ith attribute in G, $R_{U,i}$ is the attribute secret key of G_i for the user U. and t is the number of attributes in G.

Step7: Attribute authorities return G and Z_U to license server.

Step8: License server generates action control W based on V_U, and then creates the license L with the signature H_L.

$$H_L = Signature(S_L, W, I_C, Z_U, G)$$
$$L = (W, I_C, Z_U, G, H_L)$$

Step9: License server sends L to business platform.

Step10: Business platform returns L to user at last.

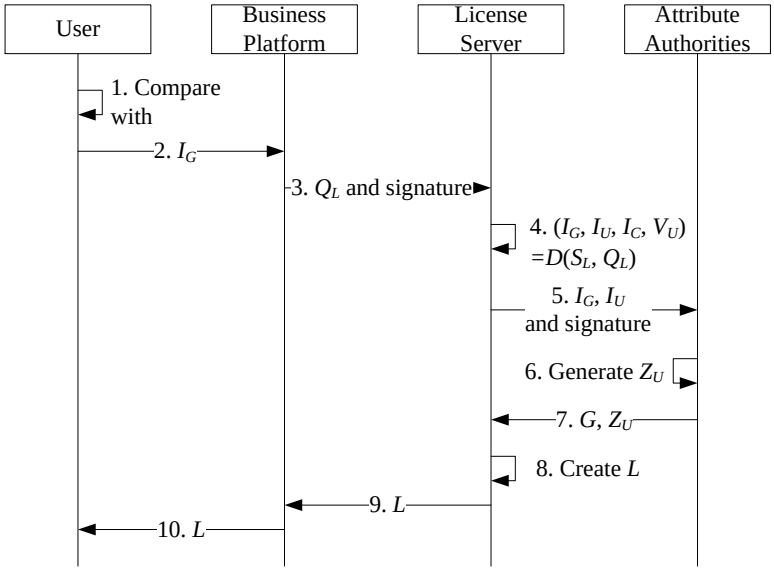

Figure 3. License Distribution Process

5.4. Content Decryption Phase

In content decryption phase, A need to be compared with the T before the content decryption. If attributes are sufficient, the DRM client checks the *w* in license, and executes content decryption according to *W*. Thus, the DRM client first gets R_U from license with S_U.

$$R_{U,i} = D(S_U, Z_{U,i})$$

$$R_U = (R_{U,1}, \ldots, R_{U,t})$$

Then gets K_E on condition of both T and action control *W* are satisfied.

$$K_E = Decrypt(K_P, \mathrm{T}, S_U, R_U, m_2)$$

Finally, the DRM client uses K_E to decrypt encrypted content.

$$M = D(K_E, m_1)$$

6. Security and Performance Analysis

6.1. Security Analysis

We analyze the security of our scheme with the following properties:

Data confidentiality: We adopt random CEK to encrypt content and then encrypt CEK based on DABE. Further, attribute secret keys of the user are protected by user's secret key in the license. Only the user who passes attributes comparison and license checking can decrypt the encrypted content. Moreover, our scheme can resist collusion attack, since the attribute secret keys are associated with identity of the user and only can decrypt encrypted content together with corresponding secret key of the user. Thus, even if two or more users conclude attributes and attribute secret keys to access content, the collusion attack cannot be effective.

Reducing reliance on the credibility of DRM client: We analyze the degree of credibility by comparing our scheme with the current DRM schemes, in which DRM client decrypts the content only need the license and must be operated in fully trusted environment. But in our scheme two steps, attributes comparison and action control in the license, are performed before decryption, which provides protection against malicious users to a certain extent: If the malicious user attacks the DRM client to get the license, he also cannot satisfy the access policy of the encrypted content, so he cannot decrypt the encrypted content. Thus our scheme has a better robustness compared with current DRM schemes and can reduce reliance on the credibility of DRM client.

Reliability: Our scheme adopts numbers of attribute authorities; each attribute authority maintains a certain amount of attributes and has its own scope of rights. It will not destroy whole system once several attribute authorities have been wrecked. Thus, our scheme can avoid single point of failure, which improves reliability of the system.

Table 2. Comparison with other DRM Systems

	Ref.[3]	Ref.[11]	Ref.[10]	Ref.[2]	Proposed scheme
Encryption technology	PRE	IBE, ABE	CP-ABE	DABE	DABE
Dynamic authorization	No	N/A	No	No	Yes
Fine-grained access control	No	Yes	Yes	Yes	Yes
Independent key management	Yes	Yes	No	N/A	No
Robustness	Weak	Middle	Weak	Strong	Strong

7. Conclusion and Next Works

In this paper, we improve Muller, *et al.*, scheme and propose a secure DRM scheme based on DABE and show the processes from system initialization to content decryption in details. Our DRM scheme achieves fine-grained access control and enables dynamic authorization. User can download content at will, but only the user who passes attributes comparison and license checking can access the content, which lowers the trust required in DRM client. Moreover, there is no independent key management in our scheme, which reduces the burden on the server. The analysis and comparison indicate the advantage and security of our scheme. The next central work includes perfecting our scheme with revocation function, such as user revocation and attributes revocation [14].

Acknowledgements

This work has been supported by the National Natural Science Foundation of China under Grant No. 60803157, 90812001, 61272519.

References

[1] Q. Huang, Z. Ma, J. Fu, Y. Yang and X. Niu, "Towards an Efficient and Secure Online Digital Rights Management Scheme in Cloud Computing", International Journal of Security and its Applications, vol. 8, no. 1, **(2014)**, pp. 159-168.
[2] S. Muller and S. Katzenbeisser, "A New DRM Architecture with Strong Enforcement", Proceedings of the International Conference on Availability, Reliability, and Security, **(2010)** February 15-18, Krakow.
[3] G. Ma, Q. Pei, X. Jiang and Y. Wang, "A Proxy Re-encryption based Sharing Model for DRM", International Journal of Digital Content Technology and its Applications, vol. 5, no. 11, **(2011)**, pp. 385-393.
[4] R. Petrlic and C. Sorge, "Privacy-Preserving DRM for Cloud Computing", Proceedings of the International Conference on Advanced Information Networking and Applications Workshops, **(2012)** March 26-29, Fukuoka.
[5] A. Sahai and B. Waters, "Fuzzy Identity Based Encryption", Computer Science, **(2005)**, pp. 457-473.
[6] C. Wang and J. Luo, "A Key-policy Attribute-based Encryption Scheme with Constant Size Ciphertext", Proceedings of the 2012 8th International Conference on Computational Intelligence and Security, **(2012)** November 17-18, Guangzhou.
[7] D. Xu, F. Luo, L. Gao and Z. Tang, "Fine-grained Document Sharing Using Attribute-based Encryption in Cloud Servers", Proc. of 2013 3rd International Conference on Innovative Computing Technology, **(2013)** August 29-31, pp. 65-70, London.
[8] V. Goyal, O. Pandey, A. Sahai and B. Waters, "Attribute-Based Encryption for Fine-Grained Access Control of Encrypted Data", Proceedings of the ACM Conference on Computer and Communications Security, **(2006)**.
[9] J. Bethencourt, A. Sahai and B. Waters, "Ciphertext-Policy Attribute-Based Encryption", IEEE Symposium on Security and Privacy, **(2007)**, pp. 321-334.
[10] C. Wang, X. Liu and W. Li, "Implementing a Personal Health Record Cloud Platform Using Ciphertext-Policy Attribute-Based Encryption", Proceedings of the 2012 International Conference on Intelligent Networking and Collaborative Systems, **(2012)** September 19-21, Bucharest.
[11] R. Dutta, D. Mishra and S. Mukhopadhyay, "Access Policy Based Key Management in Multi-level Multi-distributor DRM Architecture". Lecture Notes in Computer Science, vol. 7011, **(2011)**, pp. 57-71.
[12] P. Wang, D. Feng and L. Zhang, "CP-ABE Scheme Supporting Fully Fine-Grained Attribute Revocation", Journal of Software, vol. 23, no.10, **(2012)**, pp. 2805-2816.
[13] S. Muller, S. Katzenbeisser and C. Eckert, "Distributed Attribute-Based Encryption", Computer Science, **(2009)**, pp. 20-36.
[14] M. Li, S. Yu, Y. Zheng, K. Ren and W. Lou, "Scalable and Secure Sharing of Personal Health Records in Cloud Computing Using Attribute-Based Encryption", IEEE Transactions on Parallel and Distributed Systems, vol. 24, no. 1, **(2013)**, pp. 131-143.

Authors

Fu Jingyi, she received BS degree in information security from Chongqi University of Posts and Telecommunications in 2012. She is currently a MS candidate at the School of Computer Science, Beijing University of Posts and Telecommunications. Her research interest includes Information Security and Digital Rights Management.

Ma Zhaofeng, he is an associate professor in the School of Computer Sicence, Beijing University of Posts and Telecommunications. He got the PhD degree from Xi'an Jiaotong University in 2004. His research interest includes Information Security and Network Security and Digital Rights Management.

Huang Qinlong, he received BS degree in information security from Yunnan University in 2009. He is currently a PhD candidate at the School of Computer Science, Beijing University of Posts and Telecommunications. His research interest includes Information Security and Cloud Computing Security and Digital Rights Management.

Yang Yixian, he received the BS degree in Applied Mathematics from Chengdu Institute of Telecommunication Engineering, China, in 1983, the MS degree and PhD degree from Beijing University of Posts and Telecommunications (BUPT), China, in 1986 and 1988, respectively. He is a professor of BUPT from 1992. He is also doctoral supervisor in school of computer science. His research interests are Information and Network Security, Cryptography, Chaos, and Fuzzy Systems.