A Note On One Secure Data Self-Destructing Scheme in Cloud Computing

Lihua Liu[1], Yang Li[1], Zhengjun Cao[2], and Zhen Chen[2]
(Corresponding author: Zhengjun Cao)
Department of Mathematics, Shanghai Maritime University, China[1]
Department of Mathematics, Shanghai University[2]
No.99, Shangda Road, 200444, Shanghai, China
(Email: caozhj@shu.edu.cn)
(Received May 19, 2018; Revised and Accepted Nov. 9, 2018; First Online June 14, 2019)

Abstract

Recently, Xiong *et al.* have proposed a secure data self-destructing scheme [IEEE TCC, vol. 2, no. 4, pp. 448-458, 2014] in cloud computing. The scheme aims to solve some important security problems by supporting user-defined authorization period and by providing fine-grained access control during the period. The sensitive data will be securely self-destructed after a user-specified expiration time. In this note, it shows that the scheme is flawed because its decryption mechanism is incorrect. The consistency between encryption mechanism and decryption mechanism is not kept. We also show that it seems difficult to revise its decryption mechanism.

Keywords: Attribute-based Encryption; Cloud Computing; Data Self-destructing Scheme; Fine-grained Access Control; Time-Specific Encryption

1 Introduction

Cloud computing greatly benefits data mining, computational financing, and many other data-intensive activities by supporting a paradigm shift from local to network-centric computing and network-centric content. It enables customers with limited computational resources to outsource large-scale computational tasks to the cloud [20–22, 27, 28, 31].

Attribute-based encryption (ABE), introduced by Sahai and Waters, is a type of fuzzy identity-based encryption. In the scenario, a user's identity is composed of a set of strings which serve as descriptive attributes of the user, and the sender only needs to know the receivers' description in order to determine their public key. ABE has attracted much attention [14]. For example, Lewko, Waters, Pirretti, Goyal, Yamada, *et al.* [1, 25, 37] studied the construction of ABE systems and its shortcomings. Ostrovsky, Sahai, and Waters [29] investigated some non-monotonic access structures of ABE. Bethencourt, Sahai, Waters, and Goyal, *et al.* proposed some ciphertext-policy ABE schemes [2, 16, 33]. Chase and Chow [8, 9] introduced the setting of multi-authority in ABE. Hohenberger and Waters [17] discussed oline/offline ABE. In 2018, Cao *et al.* [4] discussed an inherent shortcoming of the cryptographic primitive of ABE. Notice that these ABE schemes do not support user-defined authorization period and secure self-destruction after expiration for privacy-preserving of the data lifecycle in cloud computing, because of the lacking of time constraints.

The cryptographic primitive of data self-destructing, introduced by Geambasu *et al.* [15], enables users to control over the lifecycle of the sensitive data. Recently, Xiong *et al.* [36] employed identity-based timed release encryption algorithm [6] and the distributed hash table network and proposed a full lifecycle privacy protection scheme for sensitive data. The time-specific encryption [30] is an extension of timed release encryption (TRE) [6]. In TRE, a piece of protected data can be encrypted in such a way that it cannot be decrypted (even by a legitimate receiver who owns the decryption key for the ciphertext) until the time (called the release-time) that was specified by the encryptor. Most of the previous TRE schemes do not consider the sensitive data privacy after expiration [23, 24].

In 2013, Chen *et al.* [13, 38] investigated on achieving secure role-based access control on encrypted data in cloud storage. In 2014, Chen *et al.* proposed two computation outsourcing schemes for linear equations and for linear programming [10,11]. But the schemes are insecure because the technique of masking a vector with a diagonal matrix is vulnerable to statistical analysis attacks [5]. The Wang *et al.*'s scheme for outsourcing linear equations is flawed [3], too. Hsien *et al.* [7, 12, 18, 26, 32] have presented some good surveys on public auditing for secure data storage in cloud computing.

In 2014, Xiong *et al.* [35] proposed a data self-destructing scheme in cloud computing by using key-policy attribute-based encryption with time-specified attributes. In the scheme, every ciphertext can only be

decrypted if both the time instant is in the allowed time interval and the associated attributes satisfy the key's access structure. The scheme aims to provide an encryption mechanism with multipurpose, such as confidentiality, data self-destructing function, and flexible control on legitimate receivers. In this note, we would like to stress that Xiong *et al.*'s scheme is flawed because the user cannot finish the calculations in the decryption phase. Furthermore, we want to point out that it is difficult to simply revise the decryption mechanism, because it requires that the authority should share the secret exponents with the user, which enables the user to decrypt any ciphertext.

The remainder of this paper is organized as follows. It reviews Xiong *et al.*'s scheme in Section 2, and then points out that the scheme has three drawbacks in Section 3. The first is that its consistency between encryption mechanism and decryption mechanism is not kept, which means a legitimate receiver cannot successfully recover the plaintext. We then point out that the scheme cannot be simply revised because the authority has to share the session exponents with any legitimate user. We also explain the reason for setting lots of parameters in Xiong *et al.*'s scheme.

2 Review of Xiong *et al.*'s Scheme

The entities in the scheme [35] comprises data owner, the authority, time server, cloud servers, users, potential adversary. It consists of four phases: Setup, Encryption, KeyGeneration and Decryption.

Setup. Let G be a bilinear group of prime order p, $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}'$ be a bilinear map, where $g \in \mathbb{G}$ is a generator. Let T be the maximum time in the system, where $|T| = n'$. Let $\mathcal{U} = \{1, \cdots, n\}$ be the universe of all attributes. The authority picks $y \in \mathbb{Z}_p$ and sets $g_1 = g^y$. Choose

$$g_2, u'_{1,1}, \cdots, u'_{n,1}, u'_{1,2}, \cdots, u'_{n,2}, u_1, \cdots, u_T \in \mathbb{G}.$$

Set public parameters as

$$g, g_1, g_2, u'_{1,1}, \cdots, u'_{n,1}, u'_{1,2}, \cdots, u'_{n,2}, u_1, \cdots, u_T.$$

The master key is set as $MSK = g_2^y$.

Encryption. To encrypt a message M under a set of attributes S_{att} with every attribute $i \in S_{att}$, where i is constrained by a time interval $T'_i \in [t_{m_{L,i}}, t_{m_{R,i}}]$ (the double-subscript notation $t_{m_{L,i}}$ indicates that the time is associated with the attribute hierarchy m and the concrete attribute i), the data owner picks $s \in \mathbb{Z}_p$, defines $c_{L,i} = n' - m_{L,i}$ and sets the ciphertext as

$$\begin{aligned} &S_{att}, C = g^s, C_M = M \cdot e(g, g_2)^{sy}, \\ &\{E = \left(u'_{i,1}\Pi_{j=1}^{m_{R,i}+1} u_j^{t_j}\right)^s, \\ &E' = \left(u'_{i,2}\Pi_{j=1}^{c_{L,i}} u_j^{T-t_j}\right)^s, T'_i\}_{i \in S_{att}} \end{aligned}$$

$\diamond$ *Notice that the encryption mechanism is well-defined because of*

$$C_M = M \cdot e(g_1, g_2^s) = M \cdot e(g^y, g_2^s) = M \cdot e(g, g_2)^{sy}.$$

That is, the encryptor can complete the phase by invoking the system's parameters and the picked exponent s.

KeyGeneration. For non-leaf node x in access tree Υ, the authority sets the degree d_x of the polynomial q_x and its threshold value k_x such that $d_x = k_x - 1$. For the root node r, set $q_r(0) = y$ and choose other d_r points to completely define the polynomial q_r. For any other node x, set

$$q_x(0) = q_{parent(x)}(index(x))$$

and pick d_x other points to define the polynomial q_x completely. Define a leaf node $x \in S_Y$ in the tree as an attribute which is constrained by a time instant t'_{n_x}, where S_Y denotes the leaf node set of Υ. Set the index $n_x = n' - c_x$.

The authority picks $r_x, r'_x \in \mathbb{Z}_p$, computes and sends the following secret key d to the user:

$$\begin{aligned} d = \{&D_{x,1}, D_{x,2}, g^{r_x}, g^{r'_x}, u_{n_x+2}^{r_x}, \cdots, u_T^{r_x}, \\ &u_{c_x+1}^{r'_x}, \cdots, u_T^{r'_x}, t_{n_x}\}_{x \in S_Y}, \end{aligned}$$

where

$$\begin{aligned} D_{x,1} &= g_2^{q_x(0)+\tau_x}\left(u'_{i,1}\Pi_{j=1}^{n_x+1} u_j^{t_j}\right)^{r_x} \\ D_{x,2} &= g_2^{-\tau_x}\left(u'_{i,2}\Pi_{j=1}^{c_x} u_j^{T-t_j}\right)^{r'_x} \end{aligned}$$

Decryption. This is a recursive algorithm from bottom to up, performed by the user. For a leaf node x: If $t_{n_x} \notin [t_{m_{L,x}}, t_{m_{R,x}}]$, the algorithm simply outputs $\perp$. Otherwise, it picks $r''_x, r'''_x \in \mathbb{Z}_p$ and computes

$$\begin{aligned} &\{a_0, g^{r_{R,x}} \cdot g^{r''_x}, u_{m_{R,x}+2}^{r_{R,x}} \cdot u_{m_{R,x}+2}^{r''_x}, \cdots, u_T^{r_{R,x}} \cdot u_T^{r''_x}\} \\ &\{b_0, g^{r_{L,x}} \cdot g^{r'''_x}, u_{c_{L,x}+1}^{r_{L,x}} \cdot u_{c_{L,x}+1}^{r'''_x}, \cdots, u_T^{r_{L,x}} \cdot u_T^{r'''_x}\} \end{aligned}$$

where

$$\begin{aligned} a_0 =& D_{x,1}(u'_{i,1}\Pi_{j=n_x+1}^{m_{R,x}+1} u_j^{t_j})^{r_{R,x}}(u'_{i,1}\Pi_{j=1}^{m_{R,x}+1} u_j^{t_j})^{r''_x} \\ =& g_2^{q_x(0)+\tau_x}(u'_{i,1}\Pi_{j=1}^{m_{R,x}+1} u_j^{t_j})^{r_{R,x}+r''_x} \\ b_0 =& D_{x,2}(u'_{i,2}\Pi_{j=c_x}^{c_{L,x}} u_j^{T-t_j})^{r_{L,x}}(u'_{i,2}\Pi_{j=1}^{c_{L,x}} u_j^{T-t_j})^{r'''_x} \\ =& g_2^{-\tau_x}(u'_{i,2}\Pi_{j=1}^{c_{L,x}} u_j^{T-t_j})^{r_{L,x}+r'''_x} \end{aligned}$$

It then calculates

$$DN = \frac{e(g^s, a_0) \cdot e(b_0, g^s)}{e(E, g^{r_{R,x}+r''_x}) \cdot e(g^{r_{L,x}+r'''_x}, E')} = e(g, g_2)^{sq_x(0)}.$$

For a non-leaf node x with all nodes z that are the children of x, use Lagrange's interpolation method to compute

$$\begin{aligned}F_x &= \prod_{c\in S_x}(e(g,g_2)^{sq_c(0)})^{\triangle_{i,S'_x}(0)}\\ &= \prod_{c\in S_x}(e(g,g_2)^{sq_{parent(c)}(0)})^{\triangle_{i,S'_x}(0)}\\ &= \prod_{c\in S_x}e(g,g_2)^{sq_x(i)\cdot\triangle_{i,S'_x}(0)} = e(g,g_2)^{sq_x(0)}\end{aligned}$$

Finally, for the root node r,

$$e(g,g_2)^{sq_r(0)} = e(g,g_2)^{sy}$$

can be recovered. It then computes

$$M = C_M/e(g,g_2)^{sy}.$$

3 Cryptanalysis

The Xiong *et al.*'s scheme involves lots of parameters and secret exponents. It tries to link time intervals to attributes and provides flexible access control strategy. But we find the scheme is flawed.

3.1 The Consistency Between Encryption Mechanism and Decryption Mechanism is not Kept

It is easy to find that

- The true *master key* is y, not g_2^y. In KeyGeneration phase, the authority has to directly invoke y and set $q_r(0) = y$. However, g_2^y is not invoked at all.
- It fails to check the consistency between encryption mechanism and decryption mechanism. Concretely, the user cannot finish the calculations of a_0, b_0 and DN. In fact,

$$\begin{aligned}a_0 =&D_{x,1}(u'_{i,1}\Pi_{j=n_x+1}^{m_{R,x}+1}u_j^{t_j})^{r_{R,x}}(u'_{i,1}\Pi_{j=1}^{m_{R,x}+1}u_j^{t_j})^{r''_x}\\ =&g_2^{q_x(0)+\tau_x}\left(u'_{i,1}\Pi_{j=1}^{n_x+1}u_j^{t_j}\right)^{r_x}\\ &\cdot(u'_{i,1}\Pi_{j=n_x+1}^{m_{R,x}+1}u_j^{t_j})^{r_{R,x}}(u'_{i,1}\Pi_{j=1}^{m_{R,x}+1}u_j^{t_j})^{r''_x}\\ \neq&g_2^{q_x(0)+\tau_x}(u'_{i,1}\Pi_{j=1}^{m_{R,x}+1}u_j^{t_j})^{r_{R,x}+r''_x},\end{aligned}$$

$$\begin{aligned}b_0 =&D_{x,2}(u'_{i,2}\Pi_{j=c_x}^{c_{L,x}}u_j^{T-t_j})^{r_{L,x}}(u'_{i,2}\Pi_{j=1}^{c_{L,x}}u_j^{T-t_j})^{r'''_x}\\ =&g_2^{-\tau_x}\left(u'_{i,2}\Pi_{j=1}^{c_x}u_j^{T-t_j}\right)^{r'_x}\\ &\cdot(u'_{i,2}\Pi_{j=c_x}^{c_{L,x}}u_j^{T-t_j})^{r_{L,x}}(u'_{i,2}\Pi_{j=1}^{c_{L,x}}u_j^{T-t_j})^{r'''_x}\\ \neq&g_2^{-\tau_x}(u'_{i,2}\Pi_{j=1}^{c_{L,x}}u_j^{T-t_j})^{r_{L,x}+r'''_x},\end{aligned}$$

$$\begin{aligned}DN =&\frac{e(g^s,a_0)\cdot e(b_0,g^s)}{e(E,g^{r_{R,x}+r''_x})\cdot e(g^{r_{L,x}+r'''_x},E')}\\ \neq&e(g,g_2)^{sq_x(0)},\end{aligned}$$

3.2 The Scheme cannot be Simply Revised

To revise the above equations, in KeyGeneration phase $D_{x,1}, D_{x,2}$ should be replaced by

$$\begin{aligned}D_{x,1} &= g_2^{q_x(0)+\tau_x}\left(u'_{i,1}\Pi_{j=1}^{n_x}u_j^{t_j}\right)^{r_x},\\ D_{x,2} &= g_2^{-\tau_x}\left(u'_{i,2}\Pi_{j=1}^{c_x-1}u_j^{T-t_j}\right)^{r'_x}.\end{aligned}$$

Besides, it should specify that

$$r_{R,x} = r_x, \quad r_{L,x} = r'_x.$$

◇ *Notice that in the simple revision the authority has to share the session exponents r_x, r'_x with the user.*

We now want to stress that the session exponents r_x, r'_x cannot be exposed to the user [19]. Otherwise, g_2^y will be exposed to the user (inner adversary) and the user can freely recover any ciphertext. In fact, the adversary can recover the *session key* $g_2^{q_x(0)}$ by calculating

$$\begin{aligned}g_2^{q_x(0)} =&D_{x,1}D_{x,2}\left(u'_{i,1}\Pi_{j=1}^{n_x+1}u_j^{t_j}\right)^{-r_x}\\ &\cdot\left(u'_{i,2}\Pi_{j=1}^{c_x}u_j^{T-t_j}\right)^{-r'_x}\end{aligned}$$

Consequently, the *secret key* $g_2^{q_r(0)} = g_2^y$ will be recovered. Once the adversary obtains g_2^y, he can recover the plaintext by computing

$$C_M/e(C,g_2^y) = M\cdot e(g,g_2)^{sy}/e(g^s,g_2^y) = M.$$

3.3 The Reason for Setting Lots of Parameters in the Scheme

In the past years, the general instruction for designing a new cryptographic scheme is to build the new on some preliminary schemes. Consequently, the method to introduce more parameters in a new scheme is broadly adopted. To achieve different purposes, it is usual to set different parameters *separately*. As a result, the whole scheme becomes gross and the consistency between different phases becomes difficult to check.

The Xiong *et al.*'s scheme combined many techniques developed in [15, 24, 34, 36]. It has to set lots of parameters, including that for representing the universe of all attributes, time intervals, access tree and its nodes, session key, secret key, and master key. Thus, it becomes more difficult to check the consistency as the quantity of parameters increases. Moreover, the security argument becomes gloomy, intricate and unintelligible. We would like to remark that designing a cryptographic scheme with all-sided characters is inadvisable in practice.

4 Conclusion

In this note, we show that Xiong *et al.*'s scheme is flawed. We want to stress that the concepts of session key, secret

key, and master key should be accurately specified. Moreover, the consistency in a cryptographic scheme must be checked carefully.

Acknowledgements

We thank the National Natural Science Foundation of China (Project 61303200, 61411146001). The authors gratefully acknowledge the reviewers for their valuable suggestions.

References

[1] N. Attrapadung and *et al.*, "Attribute-based encryption schemes with constant-size ciphertexts," *Theoretical Computer Sciences*, no. 422, pp. 15–38, 2012.

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

[3] Z. J. Cao and L. H. Liu, "Comment on 'harnessing the cloud for securely outsourcing large-scale systems of linear equations'," *IEEE Transactions on Parallel and Distributed Systems*, vol. 27, no. 5, pp. 1551–1552, 2016.

[4] Z. J. Cao, L. H. Liu, and Z. Z. Guo, "Ruminations on attribute-based encryption," *International Journal of Electronics and Information Engineering*, vol. 8, no. 1, pp. 9–19, 2018.

[5] Z. J. Cao, L. H. Liu, and O. Markowitch, "Comment on 'highly efficient linear regression outsourcing to a cloud'," *IEEE Transactions on Cloud Computing*, pp. 1-1, 2017. DOI: 10.1109/TCC.2017.2709299

[6] A. F. Chan and I. F. Blake, "Scalable, server-passive, user-anonymous timed release cryptography," in *Proceedings of 25th International Conference on Distributed Computing Systems (ICDCS'05)*, pp. 504–513, June 2005.

[7] W. Y. Chao, C. Y. Tsai, and M. S. Hwang, "An improved key-management scheme for hierarchical access control," *International Journal of Network Security*, vol. 19, no. 4, pp. 639–643, 2017.

[8] M. Chase, "Multi-authority attribute based encryption," in *Proceedings of 4th Theory of Cryptography Conference (TCC'07)*, pp. 515–534, Feb. 2007.

[9] M. Chase and S. Chow, "Improving privacy and security in multi-authority attribute-based encryption," in *Proceedings of ACM Conference on Computer and Communications Security (CCS'09)*, pp. 121–130, Nov. 2009.

[10] F. Chen, T. Xiang, X. Lei, and J. Chen, "Highly efficient linear regression outsourcing to a cloud," *IEEE Transactions on Cloud Computing*, vol. 2, no. 4, pp. 499–508, 2014.

[11] F. Chen, T. Xiang, and Y. Y. Yang, "Privacy-preserving and verifiable protocols for scientific computation outsourcing to the cloud," *Journal of Parallel and Distributed Computing*, vol. 74, pp. 2141–2151, 2014.

[12] J. S. Chen, C. Y. Yang, and M. S. Hwang, "The capacity analysis in the secure cooperative communication system," *International Journal of Network Security*, vol. 19, no. 6, pp. 863–869, 2017.

[13] T. Y. Chen, C. C. Lee, M. S. Hwang, and J. K. Jan, "Towards secure and efficient user authentication scheme using smart card for multi-server environments," *The Journal of Supercomputing*, vol. 66, no. 2, pp. 1008–1032, 2013.

[14] P. S. Chung, C. W. Liu, and M. S. Hwang, "A study of attribute-based proxy re-encryption scheme in cloud environments", *International Journal of Network Security*, vol. 16, no. 1, pp. 1-13, 2014.

[15] R. Geambasu, T. Kohno, A. Levy, and H. M. Levy, "Vanish: Increasing data privacy with self-destructing data," in *Proceedings of 18th USENIX Security Symposium*, pp. 299–315, Aug. 2009.

[16] V. Goyal and *et al.*, "Bounded ciphertext policy attribute based encryption," in *Proceedings of 35th International Colloquium on Automata, Languages and Programming (ICALP'08)*, pp. 579–591, July 2008.

[17] S. Hohenberger and B. Waters, "Online/offline attribute-based encryption," in *Proceedings of 17th International Conference on Practice and Theory in Public-Key Cryptography (PKC'14)*, pp. 293–310, Mar. 2014.

[18] W. F. Hsien, C. C. Yang, and M. S. Hwang, "A survey of public auditing for secure data storage in cloud computing," *International Journal of Network Security*, vol. 18, no. 1, pp. 133–142, 2016.

[19] L. C. Huang, T. Y. Chang, and M. S. Hwang, "A conference key scheme based on the diffie-hellman key exchange," *International Journal of Network Security*, vol. 20, no. 6, pp. 1221–1226, 2018.

[20] M. S. Hwang, C. C. Lee, T. H. Sun, "Data error locations reported by public auditing in cloud storage service," *Automated Software Engineering*, vol. 21, no. 3, pp. 373–390, Sep. 2014.

[21] M. S. Hwang, T. H. Sun, "Using smart card to achieve a single sign-on for multiple cloud services," *IETE Technical Review*, vol. 30, no. 5, pp. 410–416, 2013.

[22] M. S. Hwang, T. H. Sun, C. C. Lee, "Achieving dynamic data guarantee and data confidentiality of public auditing in cloud storage service," *Journal of Circuits, Systems, and Computers*, vol. 26, no. 5, 2017.

[23] K. Kasamatsu and *et al.*, "Time-specific encryption from forwardsecure encryption," in *Proceedings of International Conference on Security and Cryptography for Networks*, pp. 184–204, Sep. 2012.

[24] R. Kikuchi, A. Fujioka, Y. Okamoto, and T. Saito, "Strong security notions for timed-release public-key encryption revisited," in *Proceedings of 14th International Conference on Information Security and Cryptology (ICISC'11)*, pp. 88–108, Nov. 2012.

[25] A. Lewko and B. Waters, "New proof methods for attribute-based encryption: Achieving full security through selective techniques," in *Proceedings of 32nd Annual Cryptology Conference, Advances in Cryptology (CRYPTO'12)*, pp. 180–198, Aug. 2012.

[26] C. W. Liu, W. F. Hsien, C. C. Yang, and M. S. Hwang, "A survey of attribute-based access control with user revocation in cloud data storage," *International Journal of Network Security*, vol. 18, no. 5, pp. 900–916, 2016.

[27] C. W. Liu, W. F. Hsien, C. C. Yang, and M. S. Hwang, "A survey of public auditing for shared data storage with user revocation in cloud computing", *International Journal of Network Security*, vol. 18, no. 4, pp. 650–666, 2016.

[28] L. Liu, Z. Cao, C. Mao, "A note on one outsourcing scheme for big data access control in cloud," *International Journal of Electronics and Information Engineering*, vol. 9, no. 1, pp. 29–35, 2018.

[29] R. Ostrovsky, A. Sahai, and B. Waters, "Attribute-based encryption with non-monotonic access structures," in *Proceedings of the ACM Conference on Computer and Communications Security (CCS'07)*, pp. 195–203, Oct. 2007.

[30] K. G. Paterson and E. A. Quaglia, "Time-specific encryption," in *Proceedings of 7th International Conference on Security and Cryptography for Networks (SCN'10)*, pp. 1–16, Sep. 2010.

[31] S. Rezaei, M. Ali Doostari, and M. Bayat, "A lightweight and efficient data sharing scheme for cloud computing," *International Journal of Electronics and Information Engineering*, vol. 9, no. 2, pp. 115–131, 2018.

[32] Y. L. Wang, J. J. Shen, and M. S. Hwang, "A survey of reversible data hiding for vq-compressed images," *International Journal of Network Security*, vol. 20, no. 1, pp. 1–8, 2018.

[33] B. Waters, "Ciphertext-policy attribute-based encryption: An expressive, efficient, and provably secure realization," in *Proceedings of International Conference on Practice and Theory in Public-Key Cryptography (PKC'11)*, pp. 53–70, Mar. 2011.

[34] S. Wolchok and *et al.*, "Defeating vanish with low-cost sybil attacks against large dhts," in *Proceedings of the Network and Distributed System Security Symposium (NDSS'10)*, pp. 1–15, Mar. 2010.

[35] J. Xiong and *et al.*, "A secure data self-destructing scheme in cloud computing," *IEEE Transactions on Cloud Computing*, vol. 2, no. 4, pp. 448–458, 2014.

[36] J. Xiong and *et al.*, "A full lifecycle privacy protection scheme for sensitive data in cloud computing," *Peer-to-Peer Networking and Applications*, vol. 8, no. 6, pp. 1025–1037, 2015.

[37] S. Yamada and *et al.*, "A framework and compact constructions for non-monotonic attribute-based encryption," in *Proceedings of International Conference on Practice and Theory in Public-Key Cryptography (PKC'14)*, pp. 275–292, Mar. 2014.

[38] L. Zhou, V. Varadharajan, and M. Hitchens, "Achieving secure role-based access control on encrypted data in cloud storage," *IEEE Transactions on Information Forensics and Security*, vol. 8, no. 12, pp. 1947–1960, 2013.

Biography

Lihua Liu is an associate professor with the Department of Mathematics at Shanghai Maritime University. She received her Ph.D. degree in applied mathematics from Shanghai Jiao Tong University. Her research interests include combinatorics, cryptography and information security.

Yang Li is currently pursuing his M.S. degree from Department of Mathematics, Shanghai Maritime university. His research interests include combinatorics and cryptography.

Zhengjun Cao is an associate professor with the Department of Mathematics at Shanghai University. He received his Ph.D. degree in applied mathematics from Academy of Mathematics and Systems Science, Chinese Academy of Sciences. He served as a post-doctor in Computer Sciences Department, Université Libre de Bruxelles, from 2008 to 2010. His research interests include cryptography, discrete logarithms and quantum computation.

Zhen Chen is currently pursuing his M.S. degree from Department of Mathematics, Shanghai university. His research interests include information security and cryptography.