Towards efficient sharing of encrypted data in cloud-based mobile social network

Xin Sun[1], Yiyang Yao[2], Yingjie Xia[3], Xuejiao Liu[4], Jian Chen[2], and Zhiqiang Wang[2]
[1] Electric Power Research Institute, State Grid Zhejiang Electric Power Company, China.
[2]Information and Telecommunication Branch, State Grid Zhejiang Electric Power Company, China.
[3] the College of Computer Science, Zhejiang University, Hangzhou, China.
[4]the Insitute of Service Engineering, Hangzhou Normal University, Hangzhou, China.
*Corresponding author: Yingjie Xia [e-mail: xiayingjie@zju.edu.cn]

Received November 4, 2015; revised February 17, 2016; accepted February 13, 2016; published April 30, 2016

Abstract

Mobile social network is becoming more and more popular with respect to the development and popularity of mobile devices and interpersonal sociality. As the amount of social data increases in a great deal and cloud computing techniques become developed, the architecture of mobile social network is evolved into cloud-based that mobile clients send data to the cloud and make data accessible from clients. The data in the cloud should be stored in a secure fashion to protect user privacy and restrict data sharing defined by users. Ciphertext-policy attribute-based encryption (CP-ABE) is currently considered to be a promising security solution for cloud-based mobile social network to encrypt the sensitive data. However, its ciphertext size and decryption time grow linearly with the attribute numbers in the access structure. In order to reduce the computing overhead held by the mobile devices, in this paper we propose a new Outsourcing decryption and Match-then-decrypt CP-ABE algorithm (OM-CP-ABE) which firstly outsources the computation-intensive bilinear pairing operations to a proxy, and secondly performs the decryption test on the attributes set matching access policy in ciphertexts. The experimental performance assessments show the security strength and efficiency of the proposed solution in terms of computation, communication, and storage. Also, our construction is proven to be replayable choosen-ciphertext attacks (RCCA) secure based on the decisional bilinear Diffie-Hellman (DBDH) assumption in the standard model.

Keywords: attributed-based encryption, decryption outsourcing, mobile device, social network, cloud

1. Introduction

As techniques development of mobile devices and increasing popularity of interpersonal sociality, the mobile social network becomes more and more popular in everyday life. This widely spreading trend also brings the challenges on the large amount of data and their efficient sharing. Nowadays, the cloud computing is so developed that can provide flexible, on-demand and low-cost usage of computing resources and IT services. Therefore, the mobile social network steps into a new architecture version called cloud-based mobile social network. The cloud-based mobile social network always works as building intercommunication paths between mobile clients and the cloud center, and transferring data on the paths with access control policy enforcement and efficient sharing. Yet, under such architecture, issues as risks of privacy exposure, and how to realize efficient data access and fine-grained data sharing has remained the most important challenges. They require researchers and practitioners to construct a trustworthy architecture for the cloud-based mobile social network, which includes a large numbers of lightweight, resource-constrained mobile devices.

Recently a new kind of encryption mechanism called Attribute-Based Encryption (ABE) was put forth by Sahai and Waters [1]. Since its introduction in 2005, researchers have proposed many variants [2], [3], [4], [5], [6], [7], [8] to extend its functionality. ABE achieves flexible one-to-many encryption instead of one-to-one, which has significant advantage over the traditional public key primitives. Ciphertext-policy attribute-based encryption (CP-ABE) [3] is regarded as a promising technique for access control of encrypted data in cloud computing,which allows the encryption of data by specifying an access control policy over attributes, so that only users with a set of attributes satisfying this policy can decrypt the corresponding data. As the complexity of the access control policy grows, the computation overhead of each decryption becomes really high because it usually requires many pairing operations in most of the existing CP-ABE schemes. As a result, this direct decryption method in sharing data for a user who has the right to access will suffer a severe efficiency drawback, especially for a large number of users with frequent data interaction and sharing, such as in the mobile social network.

On mobile terminal devices (like cell phone or pad), whose processors are often one to two orders of magnitude slower than their desktop counterparts, there is a significant challenge for these terminal users to handle the computation intensive paring operations in ABE decryption. Experiments in [9] show that, for an ABE ciphertext containing 100 attributes, the decrypting time required on a mobile terminal device of a high performance machine 1 would be about 30 seconds, following a significant consumption of battery power. To address the aforementioned issue, Green et al. proposed to outsource the decryption of ABE ciphertext, which allows the user to perform heavy decryption through “borrowing” the computing resources from a third party service provider without revealing data [9]. However, in the tradional outsourcing decryption of ABE work, a user knows whether the attributes and the policy match or not only after repeating decryption attempts, which usually requires many pairings in most of the existing ABE schemes. Therefore, this direct decryption method in ABE still suffers a severe efficiency drawback and causes serious time lagged for the users.

In this paper, we propose an Outsourcing decryption and Match-then-decrypt CP-ABE algorithm (OM-CP-ABE) scheme over cloudbased mobile social network, in which a matching phase is additionally introduced before the decryption phase and most of the decryption task is delegated by the proxy for the user. The matching algorithm works by computing special components in ciphertexts, which are used to perform the test that if the attribute private key matches the hidden attributes policy in ciphertexts without decryption. Meanwhile, in the case that the mobile clients decrypt the ciphertexts, the proxy performs most of the decryption computation task instead of the mobile devices without revealing data. This technique works by dividing the secret key into two parts, which leaves the larger part on the proxy, and keeps the smaller one private to the user. We authorize the proxy to do the complex calculation in decryption operation.

The proposed scheme makes the following two contributions:

(1) We propose an optimized construction with outsourcing decryption, in which the master key is divided into two parts, and then partial private key of a user is short and constant, which greatly saves the storage overhead at the user side. After outsourcing, the computational cost at user side during decryption is reduced to approximate one pairing, which is constant.

(2) To reduce computation overhead, we formulize a novel paradigm of outsourcing decryption with a matching technique. By introducing the matching algorithm, once a mobile device wants to retrieve data from cloud, the proxy can efficiently return a transformed encrypted data or cannot access.

The rest of this paper is organized as follows. Section 2 presents some related work. In section 3, we present the system models and the main idea of our scheme. In section 4, we give the security proof of our scheme in detail. In section 5, we illustrate the performance of our scheme and discuss it with comparison to several existing works. Finally, we conclude our work in Section 6.

2. Related Work

2.1 Security in mobile social network

The emerging mobile social networks provide enormous novel approaches for efficiently adopting advanced networking communications and data analysis schemas by using the existing devices, datasets, networks, and infrastructures. Considering the uniqueness of the mobile social networks, using the mechanism is encountering a variety of security challenges from multiple dimensions, such as mobile apps, wireless communications, cloud systems, big data, and security operations. Compared with traditional security issues, the applications of mobile social networks are operated in a dynamic circumstance involving different internal and external inputs or elements, which requires various security mechanisms in distinct operational scenarios [10]. For the purpose of preventing from the threats of mobile social networks, some cyber security approaches based on the mobile device have been created.

Some research work is conducted on the privacy problem in location-based services [28] and mobile social network. They can provide access control over sharing objects, such as photos, in online and mobile cloud environments [11], [12], [13], [27]. Manweiler et al. present a security mechanism to resolve the tension between the value provided by location-based services and the location-privacy concerns on mobile services [14]. They

utilize k-anonymity and cryptographic techniques to protect the systems from information snooping and man-in-the-middle attacks. SmokeScreen designed a protocol for mobile devices to broadcast their identities, which can be used to achieve presence-sharing with flexible privacy controls [15]. Beach et al. propose a mechanism to allow location-based services to query local mobile devices for user social information, without disclosing user identity or compromising user privacy and security [16]. All these work focus on solving the privacy problem on mobile devices, but neglecting access control on the sharing data.

2.2 Attribute-based encryption

Attributed-based encryption has become a promising cryptographic tool for fine-grained access control of encrypted data since its first introduction. Following their work, ABE is developed into two complimentary forms: Key-Policy ABE (KP-ABE) [2] and Ciphertext-Policy ABE (CP-ABE) [3]. Subsequently, a number of variants of ABE schemes have been proposed [4], [17], [18], [5], [19], [20], [6]. However, almost all of these existing ABE schemes require a large number of exponentiations at the users'side during decryption. The computational cost in decryption commonly grows with the complexity of access control policy, which becomes a bottleneck limiting its application in mobile cloud computing.

Recently, some works have been done to improve the decryption efficiency for ABE which will largely eliminate the computation overhead for users [21], [9], [22], [23]. Green et al.[9] introduce the notion of ABE with outsourced decryption. They produce two keys for a user: a transformation key (TK) and a secret key (SK) by a key blinding technique. In their work, a proxy is delegated to translate ABE ciphertext by using TK sent from the user. Zhou et al. [21] present a privacy preserving Ciphertext Policy Attribute-based Encryption (PP-CP-ABE) to protect users' data. They propose an attribute based data storage (ABDS) to outsource computation-intensive encryption and decryption operations to cloud service providers. However, the system imposes a heavy burden on the data owner during data operations. Baden et al proposed to protect online social network with user-defined privacy [26], however at that time they do not consider the efficiency with limited mobile devices.

3. System Model and Algorithm Definition

In this section, we present an overview of system model and a formal definition of algorithms in our proposed solution.

3.1 System Model

We consider a mobile social network with cloud storage system where users have weak terminal device to decrypt the ciphertext. As shown in **Fig. 1**, the system model is composed of the following five parts: the data owners (DO), cloud server (CS), a certificate authority(CA), the data users (Users) and a proxy (Proxy). DO define the access policies, encrypt data under the access policies and send the ciphertext to the cloud server. When a user sends a reading request to the data users, CS will send the corresponding ciphertext to Proxy. In our scheme, Proxy is semi-trusted which means that the proxy is trusted to carry out computations delegated by the users. Proxy undertakes most of complicated bilinear pairing operations in the decryption process which relieves the decryption workload at the user side. CA is a fully trusted organization. It is responsible for

managing system attributes and distributing the private key according to the user attributes. The cloud server provides data storage service for data owners and data access service to users.

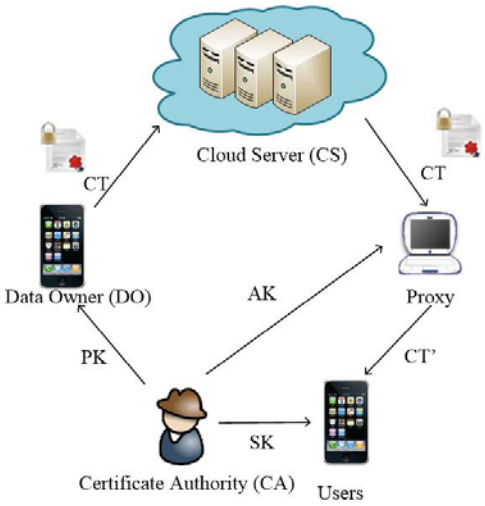

Fig. 1. System Model

In our scheme, the security key for the users is divided into two parts. One part is concealed in what we call an "attribute key" AK which is shared with the proxy and the other one is hidden in SK which must be kept private by the users. Once the proxy receives the ciphertext CT, there are two operations needed to be done. Firstly, the proxy performs the "matching" decryption test for the user, to check whether the user has the decryption capability. If the decryption test returns true, the proxy will then generate CT' using AK of the responding user and send CT' to the user. In this process, the proxy can not recover the plaintext only by AK. Given the secret key SK and transformed ciphertext CT', the user can acquire the plaintext *m* by running the decryption algorithm. This kind of decryption only requires one pairing computation. Our scheme not only guarantees the security of the data, but also achieves fast decryption at the user side.

3.2 Algorithm Definition

The construction of our OM-CP-ABE scheme consists of four phases: system initialization, key generation, encryption and decryption.

Phase 1: System Initialization

CA is responsible for generating public parameters and secret keys. For system initialization, the $Setup(\lambda,U)$ algorithm takes security parameter and the number of attributes as input in the system. It then chooses a group G of prime order p, a generator g and random group elements $h_1,...,h_U \in G$ that are associated with the attributes U in the system. In addition, it chooses random exponents $\alpha_1 \in Z_p, \alpha_2 \in Z_p$, and let $\alpha = (\alpha_1 + \alpha_2) \bmod p$. The public key PK is published as:

$$g,e(g,g)^{\alpha},g^{a},h_1,...,h_U$$

CA sets $MK = \alpha_1, \alpha_2$, α as the master secret key.

Phase 2: Key Generation

For every mobile user, CA runs the $KeyGenerate(MK,S)$ algorithm, which takes the master secret key and a set of attributes as input, and outputs security keys for the user. The algorithm firstly chooses a random t, and creates the private key as two parts, one part is AK, that is "attribute key" (AK) kept in the proxy, and the other one is SK, which is private "security key" (SK) for the users.

$$AK:(K_1 = g^{\alpha_1} g^{\alpha t}, L = g^t, \forall x \in S : K_x = h_x^t)$$

$$SK:(K_2 = g^{\alpha_2} g^{\alpha t})$$

Phase 3: Encryption

Data owners run $Encrypt(PK,(M,\rho),m)$ algorithm to encrypt their data and share with the specified users. The algorithm takes the public parameters PK and a message m to encrypt as input. In addition, it takes an LSSS access structure (M,ρ) as input. The function ρ associates rows of M to attributes. Let M be an $l \times n$ matrix. The algorithm first chooses a random vector $v = (s, y_2, \ldots, y_n)^T \in Z_p^n$. These values will be used to share the encryption exponent s. For $i = 1$ to l, it calculates $\lambda_i = M_i v$, where M_i is the vector corresponding to the i th row of M. In addition, the algorithm chooses random $r_1, \ldots r_l \in Z_p$. The ciphertext CT is published as

$$C = me(g,g)^{\alpha s}, C' = g^s, (C_1 = g^{\alpha\lambda_1} h_{\rho(1)}^{-r_1}), D_1 = g^{r_1}, \ldots, C_l = g^{\alpha\lambda_l} h_{\rho(l)}^{-r_l}, Dl = g^{r_l})$$

along with a description of (M,ρ).

Phase 4: Decryption

Once a data user wants to access some encrypted data in the cloud, the proxy firstly transforms the ciphertext using *Transform Ciphertext* (CT, AK) algorithm, and if and only if the users' attributes satisfy the access policy encrypted in the encrypted data, the user can do the final encryption. This *Transform Ciphertext* (CT, AK) algorithm takes a ciphertext CT for access structure (M,ρ) and an attribute key AK for a set of attributes S as input. Suppose that S satisfies the access structure and let $I \subset \{1,2,...,l\}$ be defined as $I \subset \{1,2,...,l\}$. Then, let $\{w_i \in Z_p\}_{i \in I}$ be a set of constants such that if $\{\lambda_i\}$ are valid shares of any secrets according to M, then $\sum_{i \in I} w_i \lambda_i = s$. (Note there could potentially be different ways of choosing the w_i values to satisfy this.) The construction mainly consists of the following two steps.

1) Matching Phase: the proxy checks whether the users can decrypt the ciphertext in terms of the following equation.

$$\sum_{i \in I} M_i w_i = (1,0,..0)$$

2) Generating CT' Phase: the proxy transforms the ciphertext CT into transformed ciphertext CT'.

$$CT' = e(C', K_1) / (\prod_{i \in I} (e(C_i, L) e(D_i, K_{\rho(i)}))^{wi})^2$$
$$= e(g^s, g^{\alpha_1} g^{at}) / (e(g,g)^{ats})^2$$
$$= e(g,g)^{\alpha_1 s} / e(g,g)^{ats}$$

The proxy will then send CT' to users.

$Decrypt(CT', SK)$: This algorithm takes transformed ciphertext CT' and user's private key SK as input, and then takes the plaintext m as output.

$$e(SK, C')CT' = e(g^{\alpha_2} g^{at}, g^s) e(g,g)^{\alpha_1 s} / e(g,g)^{ats}$$
$$= e(g,g)^{\alpha_2 s} e(g,g)^{ats} e(g,g)^{\alpha_1 s} / e(g,g)^{ats}$$
$$= e(g,g)^{\alpha s}$$

The plaintext will be got by $m = C / e(g,g)^{\alpha s}$.

4. Performance Analysis

In order to evaluate the performance of our OM-CP-ABE scheme, we use libfenc library [24], which uses key encapsulation mechanism, and adopt a 224-bit MNT elliptic curve from the Stanford Pairing-Based Crypto library [25] to implement our scheme in software. Our experiments are done on two dedicated hardware platforms: a 3.20GHz Intel Core CPU with 4GB of RAM running 32-bit Linux Kernel version 3.2.0, and a 1536MHz ARM-based HTC G18 with 768 MB of RAM running Android OS.

In our experiments, we choose 100 of the most complex policies as the form (A_1 *and* $A_2 \ldots A_n$) of which A_i is an attribute, and the values of N increase from 1 to 100, and we construct a corresponding standard decryption key that contains exact N attributes. This approach ensures that all the ciphertext components are involved in the decryption computation. We show the efficiency of our OM-CP-ABE scheme in **Fig. 2**.

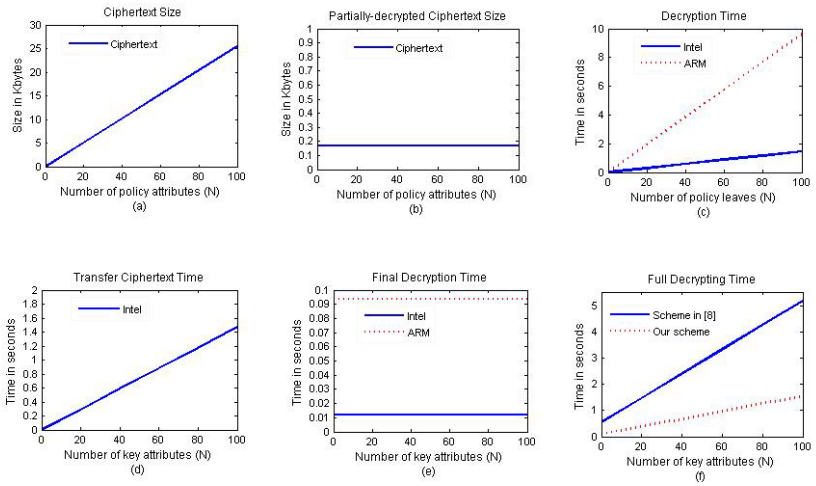

Fig. 2. Performance of our scheme

To smooth any experimental variability, we repeat each of our experiment 100 times on the Intel device and 30 times on the ARM device (due to the time consuming nature of the experiments) for each ciphertext policy and we take the average values as the experimental

results.

In **Fig. 2(a)**, an encryption under a ciphertext policy with 100 attributes results in an ABE ciphertext of nearly 25.4KB and it takes about 1.47 seconds for the Intel platform to decrypt this ciphertext. On the other hand, decryption time degrades considerably on the ARM platform: it requires almost 9.6 seconds under a policy with 100 attributes. **Fig. 2(c)** shows the size of the resulting ABE ciphertexts as a function of policy attribute N, along with the measured decryption times on the Intel and ARM test platforms without outsourcing.

Then we outsource the partial decryption computation by generating an attribute key (AK) and applying the *Transform Ciphertext* algorithm to the ABE ciphertext using this key. **Fig. 2(d)** shows the time of transform ciphertext using our OM-CP-ABE. **Fig. 2(e)** shows the time required for final decryption of transformed ciphertext in Intel and ARM test platforms. The final decryption requires only 11 milliseconds on the Intel platform and approximately 93 milliseconds on the ARM platform.

Moreover, we compare the full decrypting time in our scheme with the scheme [9] in **Fig. 2(f)**. It shows that our scheme outperforms theirs for the decryption time on a mobile device at the user terminal. Due to the differences in the system model, there are three parts needed for a user to decrypt a ciphertext for a user in their scheme–AK transformation time, transformed ciphertext transformation time and final decryption time, while our scheme only contains two parts. The solution in [9] introduces an ABE decryption outsourcing scheme, which adds the random number z to make the user's key blind, but blow up the key size. Meanwhile, there is no matching operation before decryption, which will cause a waste of time when the user can not decrypt the ciphertext, and thus is not quite efficient. Our OM-CP-ABE scheme is not only efficient due to its matching operation, but also termination resource saving by outsourcing decryption computation.

Discussion: As expected, outsourcing substantially reduces the computation time required for devices with limited computing resource to recover the plaintext. The bulk of the decryption operation is now handled by the proxy. The transformed ciphertext is not only much more efficient to decrypt but also much smaller in size. As is shown in **Fig. 2(b)**, each partially-decrypted ciphertext in our implementation, with a constant size of 176 bytes, regardless the complexity of its corresponding ciphertext policy.

Recall that in our OM-CP-ABE scheme, we delegate the most complex decryption calculation to the proxy, and leave only one pairing to the users. Also, we add a matching operation before the proxy's transforming of the ciphertext, which used to check whether the user can decrypt the ciphertext or not. It saves a lot of calculation time because if it is not satisfied, then the proxy will not transform the corresponding ciphertext. As a result, this process shortens the whole response time for the user when retrieving encrypted data from cloud service. Meanwhile, as is shown in our system model of **Fig. 1**, the CA sends users' AK to the proxy, which saves the storage space in terminal and network bandwidth by outsourcing AK.

Theorem 1 Suppose the decisional q-BDHE assumption holds. Then no polynomial time adversary can selectively break our system with a challenge matrix $M*$ of size $l^* \times n^*$, where $l^*, n^* \le q$.

Proof：Assume that we have an adversary A with non negligible advantage $\varepsilon = \mathrm{Adv}_A$ in the selective security game against our construction. Moreover, suppose it chooses a challenge matrix $M*$ where both dimensions are at most q. In the security game for CP-ABE similar to [6], the adversary can query any secret keys that cannot be used for

decryption for both proxy and user. With these conditions, the security game can be treated equally to the fast decryption construction of CP-ABE in [22]. Therefore, we can build a simulator B that plays the decisional q-BDHE problem with non-negligible advantage.

5. Conclusion and Future Work

In this paper, we consider the CP-ABE applications in cloud-based mobile social network which mobile devices are used as data clients and cloud is used as the data center. In order to realize efficient and secure data sharing on the resource-limited mobile clients, we enhance the model of outsourcing decryption in LSSS based CP-ABE construction by introducing "matching" algorithm before decryption and modifying the key generation. Experimental performance analysis demonstrates that our scheme not only reduces private key overhead but also provides fast decryption for the users.

In the future, we would like to consider the verifiability requirement of ABE with outsourced decryption, this is to verify the correctness of transformation done by the proxy in a formal definition. We also plan to refine this scheme, such as to protect the access pattern as well as the privacy of the client in ABE applications.

Acknowledgement

This research is supported in part by the following funds: National Natural Science Foundation of China under grant number 61472113, 61304188 and 61502134, Zhejiang Provincial Natural Science Foundation of China under grant number LZ13F020004 and LR14F020003, and Zhejiang Provincial Science and Technology Innovation Program under grant number 2013TD03.

References

[1] A. Sahai and B. Waters, "Fuzzy identity-based encryption," in *Proc. of Eurocrypto on Advances in Cryptology*, pp. 457–473, 2005. Article (CrossRef Link)

[2] V. Goyal, O. Pandey, A. Sahai and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data," in *Proc. of the 13th ACM Conference on Computer and Communications Security*, pp. 89–98, 2006. Article (CrossRef Link)

[3] J. Bethencourt, A. Sahai and B. Waters, "Ciphertext-policy attribute-based encryption," in *Proc. of IEEE Symposium on Security and Privacy*, pp. 321–334, 2007. Article (CrossRef Link)

[4] L. Cheung and C. Newport, "Provably secure ciphertext policy abe," in *Proc. of the 14th ACM Conference on Computer and Communications Security*, pp. 456–465, 2007. Article (CrossRef Link)

[5] R. Ostrovsky, A. Sahai and B. Waters, "Attribute-based encryption with non-monotonic access structures," in *Proc. of the 14th ACM Conference on Computer and Communications Security*, pp. 195–203, 2007. Article (CrossRef Link)

[6] B. Waters, "Ciphertext-policy attribute-based encryption: An expressive, efficient, and provably secure realization," *Public Key Cryptography*, pp. 53–70, 2011. Article (CrossRef Link)

[7] N. D. Han, L. Han, D. M. Tuan, H. P. In, M. Jo, "A scheme for data confidentiality in cloud-assisted wireless body area networks," *Information Sciences, Vol. 284*, pp. 157–166, Nov 2014. Article (CrossRef Link)

[8] Q. Kang, X. Liu, Y. Yao, Z. Wang, Y. Li, "Efficient authentication and access control of message dissemination over vehicular ad hoc network," *Neurocomputing*.2015.Article (CrossRef Link)

[9] M. Green, S. Hohenberger, B. Waters, "Outsourcing the decryption of abe ciphertexts," in *Proc. of the 20th Usenix Conference on Security*, pp. 34–34, 2011.
[10] M. Raento, A. Oulasvirta, "Designing for privacy and self-presentation in social awareness," *Personal Ubiquitous Computing*, pp. 527–542, 2008. Article (CrossRef Link)
[11] S. Ahern, D. Eckles, N. S. Good, S. King, M. Naaman and R. Nair, "Over-exposed: privacy patterns and considerations in online and mobile photo sharing," in *Proc. of SIGCHI Conference on Human Factors in Computing Systems*, pp. 357–366, 2007. Article (CrossRef Link)
[12] X. Liu, Z. Shan, L. Zhang, W. Ye and R. Yan, "An efficient message access quality model in vehicular communication networks," *Signal Processing*, pp.682-690, 2016. Article (CrossRef Link)
[13] Y. Xia, F. Xia, X. Liu, X. Sun, Y. Liu and Y. Ge, "An improved privacy preserving construction for data integrity verification in cloud storage," *KSII Transactions on Internet and Information Systems, Vol.8, No.10*, pp. 3607-3623, Oct 2014. Article (CrossRef Link)
[14] J. Manweiler, R. Scudellari, Z. Cancio and L. P. Cox, "We saw each other on the subway: secure, anonymous proximity-based missed connections," in *Proc. of 10th Workshop on Mobile Computing Systems and Applications*, pp. 1–6, 2009. Article (CrossRef Link)
[15] L. P. Cox, A. Dalton, V. Marupadi, "Smokescreen: flexible privacy controls for presence-sharing," in *Proc. of 5th International Conference on Mobile Systems*. Applications and Services, pp. 233–245, 2007.
[16] A. Beach, M. Gartrell, R. Han, "Solutions to security and privacy issues in mobile social networking," in *Proc. of International Conference on Computational Science and Engineering*, pp. 1036–1042, 2009. Article (CrossRef Link)
[17] A. Lewko, T. Okamoto, A. Sahai, K. Takashima and B. Waters, "Fully secure functional encryption: Attribute-based encryption and (hierarchical) inner product encryption," in *Proc. of Eurocrypt on Advances in Cryptology*, pp. 62–91, 2010. Article (CrossRef Link)
[18] J. Li, K. Ren, B. Zhu and Z. Wan, "Privacy-aware attribute-based encryption with user accountability," *Information Security*, pp. 347–362, 2009. Article (CrossRef Link)
[19] A. Sahai, H. Seyalioglu and B. Waters, "Dynamic credentials and ciphertext delegation for attribute-based encryption," in *Proc. of Crypto on Advances in Cryptology*, pp. 199–217, 2012. Article (CrossRef Link)
[20] B. Waters, "Dual system encryption: realizing fully secure ibe and hibe under simple assumptions," in *Proc. of Crypto on Advances in Cryptology*, pp. 619–636, 2009. Article (CrossRef Link)
[21] Z. Zhou and D. Huang, "Efficient and secure data storage operations for mobile cloud computing," in *Proc. of the 8th International Conference on Network and Service Management*, pp. 37–45, 2012.
[22] S. Hohenberger, B. Waters, "Attribute-based encryption with fast decryption," *Public Key Cryptography*, pp. 162–179, 2013. Article (CrossRef Link)
[23] Y. Zhang, X. Chen, J. Li, D. S. Wong, H. Li, "Anonymous attribute-based encryption supporting efficient decryption test," in *Proc. of the 8th ACM Sigsac Symposium on Information, Computer and Communications Security*, pp. 511–516, 2013. Article (CrossRef Link)
[24] A. Akinyele, M. Green, M. Rushanan, "Libfenc: the functional encryption library," http://code.google.com/p/libfenc/.
[25] B. Lynn, "Stanford pairings-based crypto library," *http://crypto.stanford.edu/pbc/*.
[26] R. Baden, A. Bender, N. Spring et al, "Persona: an online social network with user-defined privacy," *ACM SIGCOMM Computer Communication Review. ACM*, 39(4): 135-146, 2009. Article (CrossRef Link)
[27] Y. Xia, L. Kuang, M. Zhu, "A hierarchical access control scheme in cloud using HHECC," *Information Technology Journal*, 9(8): 1598-1606, 2010. Article (CrossRef Link)
[28] Y. Xia, M. Zhu, Y. Li, "Towards topology-and-trust-aware P2P grid," *Journal of computers*, 5(9): 1315-1321, 2010. Article (CrossRef Link)

Xin Sun received his Bachelor of Science and Master of Science degree in Zhejiang University. Now he is an Engineer in Electric Power Research Institute of State Grid, Zhejiang Electric Power Company, Hangzhou, Zhejiang, China. His research interests are application security testing and penetration testing.

Yiyang Yao received his Bachelor of Engineering degree in Software Engineering in 2007, and subsequently in 2010 a Master of Science degree in Computer Science from Northwestern Polytechnical University, Xi'an, China. Since 2010 he is a member of State Grid Zhejiang electric power company Information & telecommunication Branch. His research interests include network security and cloud computing.

Yingjie Xia received his Phd degree in the College of Computer Science, Zhejiang University in 2009. He was affiliated as a research scientist in National Center for Supercomputing Applications (NCSA), University of Illinois at Urbana-Champaign (UIUC), USA. Now he is an associate professor in Zhejiang University, Hangzhou, China. His research interests focus on information security, especially in connected vehicle networks. He has published more than 70 research papers.

Xuejiao Liu received her Phd degree from Huazhong Normal University, China in 2011. She is currently a Lecture at Institute of Service Engineering, Hangzhou Normal University, China. Her research interests include network security, and cloud security. She has published several research papers in many international journals and conferences, including signal processing, IEEE TrustCom, Security and Communication Networks and etc.

Jian Chen received a Bachelor of Engineering degree in computer science in 1983 from Xidian University, Xi'an, China. Currently he is the Deputy General Manager of State Grid Zhejiang Electric Power Company Information & Telecommunication Branch. His research interest includes internet security.

Zhiqiang Wang received a Bachelor of Engineering degree in Computer Science in 1985 from Hohai Unversity, Nanjing China. Currently he is the chief engineering of State Grid Zhejiang electric power company Information & telecommunication Branch. His research interest includes information security administration.