Toward Efficient and Privacy-Preserving Computing in Big Data Era

Rongxing Lu, Hui Zhu, Ximeng Liu, Joseph K. Liu, and Jun Shao

Abstract

Big data, because it can mine new knowledge for economic growth and technical innovation, has recently received considerable attention, and many research efforts have been directed to big data processing due to its high volume, velocity, and variety (referred to as "3V") challenges. However, in addition to the 3V challenges, the flourishing of big data also hinges on fully understanding and managing newly arising security and privacy challenges. If data are not authentic, new mined knowledge will be unconvincing; while if privacy is not well addressed, people may be reluctant to share their data. Because security has been investigated as a new dimension, "veracity," in big data, in this article, we aim to exploit new challenges of big data in terms of privacy, and devote our attention toward efficient and privacy-preserving computing in the big data era. Specifically, we first formalize the general architecture of big data analytics, identify the corresponding privacy requirements, and introduce an efficient and privacy-preserving cosine similarity computing protocol as an example in response to data mining's efficiency and privacy requirements in the big data era.

As stated by IBM, with pervasive handheld devices, machine-to-machine communications, and online/mobile social networks, we create 2.5 quintillion bytes of data each day — so much that 90 percent of the data in the world today has been created in the last two years alone [1]. Facing such a huge volume of data recently and quickly generated, it is hard for us to capture, store, manage, share, analyze, and visualize with existing data and processing tools. For this reason, the concept of big data, a buzzword today, has been proposed. Essentially, big data is a collection of data sets so large (in size) and complex (in real-time, streaming, or non-structured form) that it is difficult to use traditional data management and analysis tools to efficiently gain useful information. Therefore, big data is usually characterized by "3V" — volume, velocity, and variety. Volume captures the huge amount of data generated by organizations or individuals. Velocity shows the high speed at which data is generated, captured, and shared. Variety means a proliferation of new data types from social networks, machine devices, and mobile sources that are integrated into traditional transactional data types. Although complexity arises from the 3V challenges, big data, due to its potential value in healthcare, consumer and business analytics, governmental surveillance, and so on, has still received considerable attention not only from the government, but also from big companies and academia [2]. For example, in 2012 the U.S. government announced a national "Big Data Initiative" committing more than US$200 million to big data research projects [3]. Oracle released "big data" products and solutions for enterprises to help them derive real business values from big data [4]. The Massachusetts Institute of Technology (MIT) also hosted the Intel Science and Technology Center for Big Data Research [5].

Many efforts on big data are focused on the 3V challenges today. However, the flourishing of big data relies not only on the promised solutions for 3V challenges, but also on the security and privacy challenges in big data analytics. It is likely that if the security and privacy challenges are not well addressed, the concept of big data cannot be widely accepted. For example, when big data is exploited in the healthcare context, it could save the health care industry up to US$450 billion. Nevertheless, as patients' data are very sensitive, privacy issues become a major concern when exploiting big data in healthcare. Similarly, when big data is exploited in smart grid, a utility company can collect customers' data every 15 minutes in a residential area to structure conservation programs that analyze existing usage to forecast future use. Although this kind of big data analytics can lead to a strong competitive position for the utility company, the near-real-time data generated every 15 minutes may be abused to disclose the privacy of customers. More important, once the reported data is fabricated, big data analytics become useless [6]. Despite the fact that many security and privacy techniques have already been

Rongxing Lu is with Nanyang Technological University.

Hui Zhu is with Xidian University.

Ximeng Liu is with Nanyang Technological University and Xidian University.

Joseph K. Liu is with Institute for Infocomm Research (I2R).

Jun Shao is with Zhejiang Gongshang University.

0890-8044/14/$25.00 © 2014 IEEE

designed, they are inadequate for the newly emerging big data scenarios because they are mainly tailored to secure traditional small-size data. Therefore, in-depth research efforts dedicated to security and privacy challenges in big data are expected.

Recently, a new dimension, "veracity," has been investigated in big data security to ensure the trustworthiness and accuracy of big data, while the research of big data privacy is still at an early stage. Hence, in this article, we exploit new privacy challenges of big data, especially devoting attention toward efficient and privacy-preserving computing in the big data era. First, we give a general architecture of big data analytics and its privacy requirements. Then we review some existing privacy-preserving techniques to check whether or not they are suitable for privacy-preserving big data analytics. To identify big data privacy research more clearly, we also present an efficient and privacy-preserving cosine similarity computing protocol as an example. In the end, we draw our conclusions.

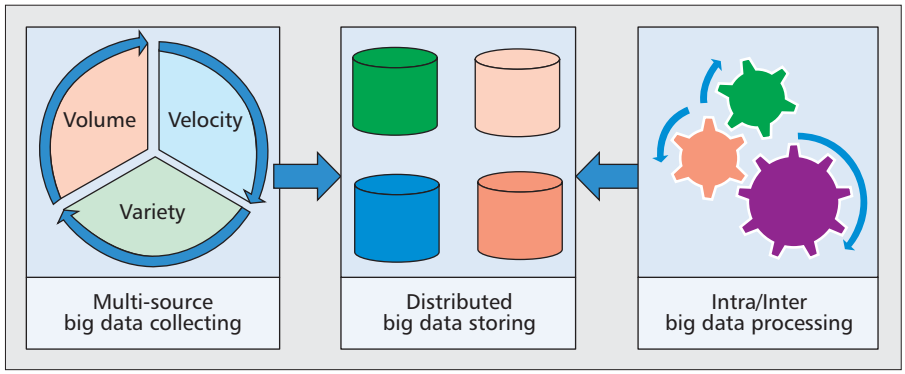

Figure 1. General architecture of big data analytics.

Architecture of Big Data Analytics and Privacy Requirements

In this section, we formalize the general architecture of big data analytics and identify the privacy requirements.

General Architecture of Big Data Analytics

Figure 1 shows a general architecture of big data analytics, which is mainly composed of three parts: multi-source big data collecting, distributed big data storing, and intra/inter big data processing.

Multi-Source Big Data Collecting — Unlike traditional data collecting, big data collecting is multi-source. Particularly, due to pervasive and ubiquitous sensing technologies and popular online/mobile social networks, big data collection is characterized by high volume, high velocity, and high variety. It was reported by IBM [1] that 2.5 quintillion bytes of data are created every day, and 90 percent of the data (including structured, unstructured, streaming, and near-real-time data) in the world today were produced with the past two years. This tangibly presents new challenges, including how to efficiently store and organize high-volume data, how to quickly process streaming and near-real-time data, and how to accurately analyze structured and unstructured data in order to maximize the value of big data.

Distributed Big Data Storing — With rapid advances in data storage technologies (e.g., the boom of cloud computing technology), storing high-volume data is no longer a big challenge. However, since big data collecting is pervasive, it is not efficient to move high volumes of collected data around for centralized storing. To this end, distributed big data storage is suggested; that is, big data will be stored and organized in their original location.

Intra/Inter Big Data Processing — Big data processing is crucial to the success of big data. Since big data are stored in a distributed way, they should be processed in parallel such that new knowledge and innovation can be mined in a reasonable amount of time. According to whether the processed big data belong to the same organization or not, big data processing can be divided into two types: intra big data processing if all data belong to the same organization, and inter big data processing if big data belong to different organizations. Compared to intra big data processing, inter big data processing is more challenging, as big data sharing should first be executed before processing, and during the duration of big data sharing, many new security and privacy issues will arise.

Privacy Requirements

In the general architecture of big data analytics, both distributed big data storing and parallel big data processing are driven by the big data 3V challenges. In addition to the 3V challenges, big data also faces new security and privacy challenges. If big data is not authentic, newly mined knowledge becomes useless. Recently, a new dimension, *veracity*, has been advocated to address the security challenges in big data. However, the study of privacy in big data is still in its early stage. Therefore, we focus ourselves on big data privacy in this article and identify the privacy requirements of big data analytics as follows.

While big data creates enormous values for economic growth and technical innovation, we are already aware that the deluge of data also raises new privacy concerns. Thus, privacy requirements in big data architecture should be identified as deeply as possible to balance the benefits of big data and individual privacy preservation.

Privacy requirements in big data collection: As big data collection takes place pervasively, eavesdropping is possible, and the data could be incidentally leaked. Therefore, if the collected data is personal and sensitive, we must resort to physical protection methods as well as information security techniques to ensure data privacy before it is securely stored.

Privacy requirements in big data storage: Compared to eavesdropping an individual's data during the big data collection phase, compromising a big data storage system is more harmful. It can disclose more individual personal information once it is successful. Therefore, we need to ensure the confidentiality of stored data in both physical and cyber ways.

Privacy requirements in big data processing: The key component of big data analytics is big data processing, as it indeed mines new knowledge for economic growth and technical innovation. Because big data processing efficiency is an important measure for the success of big data, the privacy requirements of big data processing become more challenging. We never sacrifice big efficiency for big privacy, and should not only protect individual privacy but also ensure efficiency at the same time. In addition, since inter big data processing runs over multiple organizations' data, big data sharing is essential, and ensuring privacy in big data sharing becomes one of the most challenging issues in big data processing. Therefore, it is desirable to design efficient and privacy-preserving algorithms for big data sharing and processing.

In recent years, we have witnessed plenty of privacy-pre-

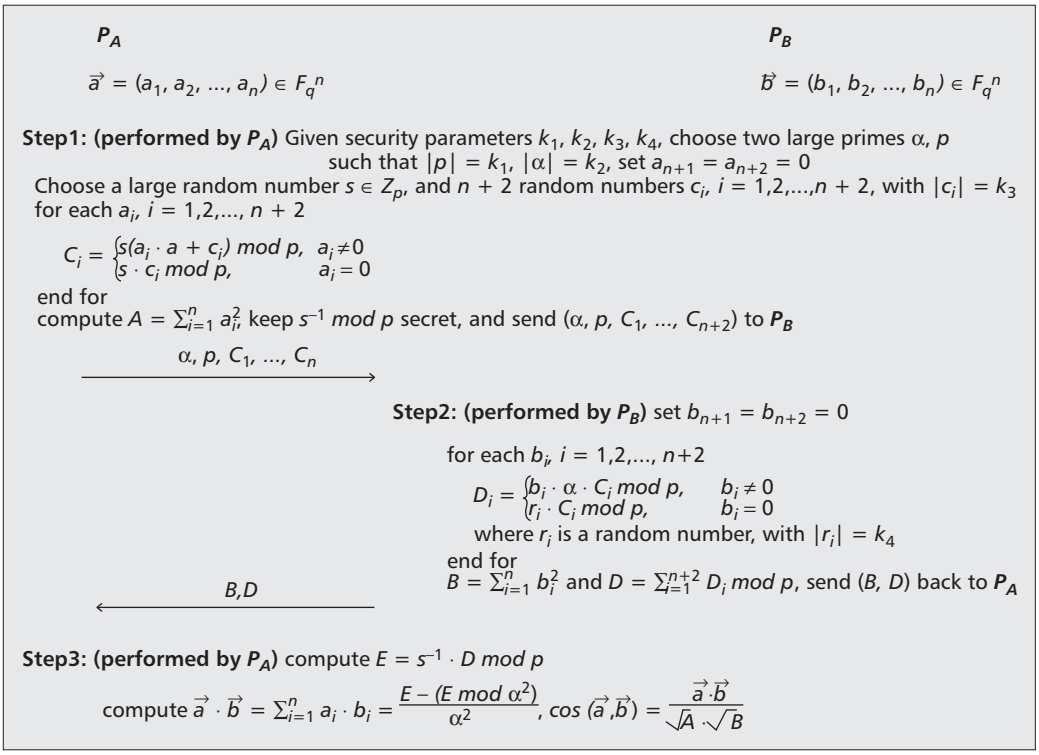

Figure 2. Proposed efficient and privacy-preserving cosine similarity computing protocol.

serving techniques being proposed. However, as they are tailored for privacy requirements in traditional analytics, they are not sufficient to satisfy the privacy requirements in big data analytics. In the following, we examine some existing privacy-preserving techniques and explain why they may not be suitable for big data analytics scenarios.

Existing Privacy-Preserving Techniques

In this section, we review some existing privacy-preserving techniques, including privacy-preserving aggregation, operations over encrypted data, and de-identification techniques.

Privacy-preserving aggregation: Privacy-preserving aggregation, which is built on some homomorphic encryption [7], is a popular data collecting technique for event statistics. Given a homomorphic public key encryption algorithm $E(\cdot)$, different sources can use the same public key to encrypt their individual data $m_1, m_2, \cdots, m_n$ into ciphertexts $c_1 = E(m_1)$, $c_2 = E(m_2)$, $\cdots$, $c_n = E(m_n)$. By taking the sum aggregation as an example, these ciphertexts can be aggregated as $C = \prod_{i=1}^{n} c_i = E(\sum_{i=1}^{n} m_i)$. With the corresponding private key, the aggregated result $\sum_{i=1}^{n} m_i$ can be recovered from C. Obviously, privacy-preserving aggregation can protect individual privacy in the phases of big data collecting and storing. However, since aggregation is purpose-specific, one-purpose aggregated data usually cannot be used for other purposes. Since its inflexibility prevents running complex data mining to exploit new knowledge, privacy-preserving aggregation is insufficient for big data analytics.

Operations over encrypted data: Currently, searching over encrypted data has been widely studied in cloud computing [8]. To keep sensitive documents private, documents and their associated keywords are encrypted and stored in a cloud server. When a user submits a "capability" encoding some query conditions, the server can return a set of encrypted documents that meet the underlying query conditions without knowing other details of the query. In such a way, the user can retrieve the desired data in a privacy-preserving way. Motivated by searching over encrypted data, our first feeling is that we can also run operations over encrypted data to protect individual privacy in big data analytics. However, as operations over encrypted data are usually complex and time-consuming, while big data is high-volume and needs us to mine new knowledge in a reasonable timeframe, running operations over encrypted data is inefficient in big data analystics.

De-identification: De-identification is a traditional technique for privacy-preserving data mining, where in order to protect individual privacy, data should be first sanitized with generalization (replacing quasi-identifers with less specific but semantically consistent values) and suppression (not releasing some values at all) before the release for data mining. Compared to privacy-preserving aggregation and operations over encrypted data, de-identification can make data analytics and mining more effective and flexible [9]. However, many real examples indicate that data which may look anonymous is actually not after de-identification; for example, only (5-digit zip code, birth date, gender) can uniquely identify 80 percent of the population in the United States. Therefore, to mitigate the threats from re-identification, the concepts of k-anonymity, l-diversity, and t-closeness have been introduced to enhance traditional privacy-preserving data mining. Obviously, de-identification is a crucial tool in privacy protection, and can be migrated to privacy-preserving big data analytics. However, as an attacker can possibly get more external information assistance for de-identification in the big data era, we have to be aware that big data can also increase the risk of re-identification. As a result, de-identification is not sufficient for protecting big data privacy.

From the above discussion, we can see that:

- Privacy-preserving big data analytics is still challenging due to either the issues of flexility and efficiency or deidentification risks.
- However, compared with privacy-preserving aggregation and operations over encrypted data, de-identification is more feasible for privacy-preserving big data analysitcs if we can develop efficient and privacy-preserving algorithms to help mitigate the risk of re-identification.

With these two points in mind, future research work on big data privacy should be directed toward efficient and privacy-preserving computing algorithms in the big data era, and these algorithms should be efficiently implemented and output correct results while hiding raw individual data. In such a way, they can reduce the re-identification risk in big data analytics and mining. To identify the research line more clearly, in the following, we introduce an efficient and privacy-preserving cosine similarity computing protocol that can efficiently calculate the cosine similarity of two vectors without disclosing the vectors to each other, and thus be very useful for privacy-preserving in big data analytics.

An Efficient and Privacy-Preserving Cosine Similarity Computing Protocol}

Cosine similarity,

$$\cos(\vec{a},\vec{b}) = \frac{\sum_{i=1}^{n} a_i \cdot b_i}{\sqrt{\sum_{i=1}^{n} a_i^2}\sqrt{\sum_{i=1}^{n} b_i^2}}$$

is an important measure of similarity of two objectives captured by vectors $\vec{a} = (a_1, \ldots, a_n)$ and $\vec{b} = (b_1, \ldots, b_n)$, respectively. In big data analytics, $\cos(\vec{a}, \vec{b})$ has become a critical building block for many data mining techniques. In the following, we introduce an efficient and privacy-preserving cosine similarity computing protocol for big data analytics.

Description of Protocol

Given $\vec{a} = (a_1, \ldots, a_n)$ and $\vec{b} = (b_1, \ldots, b_n)$, we can directly calculate the cosine similarity $\cos(\vec{a}, \vec{b})$ in an efficient way. However, when we consider inter big data processing (i.e., $\vec{a}$ and $\vec{b}$ do not belong to the same organization), the direct cosine similarity computation would disclose each other's privacy. To achieve the privacy-preserving cosine similarity computation, we can apply homomorphic encryption (HE), such as Pallier encryption (PE) [7], to compute $\vec{a} \cdot \vec{b}$ first (see [10] for details), and subsequently $\cos(\vec{a}, \vec{b})$ with the values of

$$\sqrt{\sum_{i=1}^{n} a_i^2} \text{ and } \sqrt{\sum_{i=1}^{n} b_i^2}.$$

However, since HE requires time-consuming exponentiation operations, it is inefficient in big data processing. Therefore, based on lightweight multi-party random masking and polynomial aggregation techniques [10], we introduce an efficient and privacy-preserving cosine similarity computing (PCSC) protocol for big data processing. A detailed description of PCSC is shown in Fig. 2. The key point of PCSC is how to calculate $\vec{a} \cdot \vec{b}$ in a privacy-preserving way. As shown in the figure, we can see that for each $i = 1, 2, \ldots, n + 2$, only both a_i and b_i are not equal to 0. D_i will contain $a_i b_i \alpha^2$. Thus, $D = \sum_{i=1}^{n+2} D_i$ will include $\sum_{i=1, a_i \neq 0, b_i \neq 0}^{n+2} a_i b_i \alpha^2$, which obviously equals $\sum_{i=1}^{n} a_i b_i \alpha^2$, as $a_{n+1} = a_{n+2} = b_{n+1} = b_{n+2} = 0$. To correctly extract the coefficient of α^2 (i.e., $\vec{a} \cdot \vec{b} = \sum_{i=1}^{n} a_i b_i$) from D, we should properly set the parameters as follows. From computing $E = s^{-1} \cdot D \bmod p$, we actually obtain $E = \sum_{a_i \neq 0, b_i \neq 0} a_i b_i \alpha^2 + \sum_{b_i \neq 0} b_i c_i \alpha + \sum_{a_i \neq 0, b_i = 0} r_i (a_i \alpha + c_i) + \sum_{a_i = 0, b_i = 0} r_i c_i \bmod p$. Therefore, to correctly obtain

$$\sum_{i=1}^{n} a_i b_i = \frac{E - E \bmod \alpha^2}{\alpha^2},$$

we need the constraints $\sum_{a_i \neq 0, b_i \neq 0} a_i b_i \alpha^2 + \sum_{b_i \neq 0} b_i c_i \alpha + \sum_{a_i \neq 0, b_i = 0} r_i (a_i \alpha + c_i) + \sum_{a_i = 0, b_i = 0} r_i c_i < p$ and $\sum_{b_i \neq 0} b_i c_i \alpha + \sum_{a_i \neq 0, b = 0} r_i (a_i \alpha + c_i) + \sum_{a_i = 0, b_i = 0} r_i c_i < \alpha^2$. When $q \leq 2^{32}$, $n \leq 2^{32}$, we can just set $|p| = k_1 = 512$, $|\alpha| = k_2 = 200$, $|c_i| = k_3 = 128$, and $|r| = k_4 = 128$, which can ensure that we get the correct result.

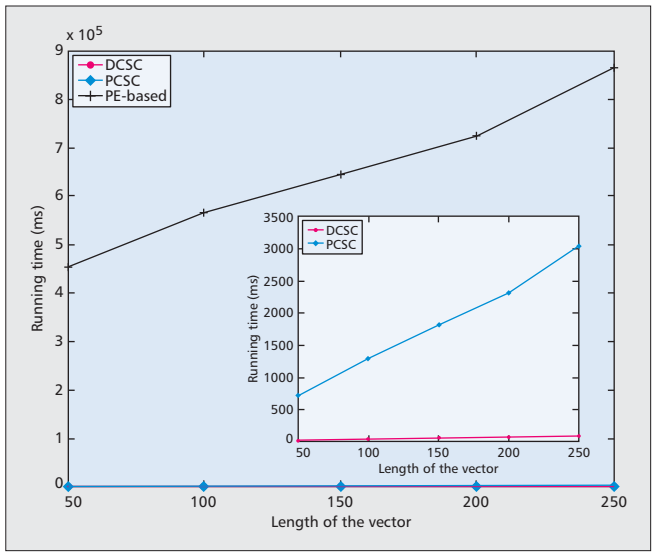

Figure 3. Average running time vs. length of vector.

For the privacy preservation of PCSC, since each a_i, $i = 1,2, \ldots, n$ is one-time masked with random $C_i = s(a_i \alpha + c_i) \bmod p$, P_A can ensure that each a_i is privacy-preserving. Note that the function of adding $a_{n+1} = a_{n+2} = b_{n+1} = b_{n+2} = 0$ is to ensure that at least two random numbers, r_{n+1}, r_{n+2}, are included in D, which can prevent P_A from guessing P_B's vector $\vec{b} = (b_1, \ldots, b_n)$. Suppose we do not add these additional random elements; when all $b_i \neq 0$, $i = 1, 2, \ldots, n$, there is no randomness from the view of P_A in D. Once both q and n are small, it is possible for P_A to guess $\vec{b}$ from the deterministic D. Therefore, adding the random values r_{n+1}, r_{n+2} is necessary, which can eliminate the above guessing attack. As a result, the proposed PCSC protocol should be privacy-preserving.

For efficiency, compared with the HE-based protocol, the proposed PCSC protocol does not require time-consuming exponential operations. As a result, it is more efficient. In the following, we use experiments to evaluate the efficiency of the proposed PCSC protocol.

Performance Evaluation

To evaluate the proposed PCSC protocol, we compare it with direct cosine similarity computation and the HE-based protocol. We first implement the PCSC protocol, HE-based protocol (using Pallier encryption and referred to as PE-based protocol here), and direct cosine similarity computation (DCSC) with Java. We run them with the same input on a PC with an Inter Pentium CPU B980 running at 2.40 GHz, and with 6 Gbytes of RAM for evaluation. Concretely, with the parameter settings $q = 2^8$, $k_1 = 512$, $k_2 = 200$, $k_3 = k_4 = 128$, $n = \{50, 100, 150, 200, 250\}$, each time we first randomly generate a vector $\vec{a} = (a_1, \ldots, a_n) \in F_q^n$ and read 10,000 vectors $\vec{b_i} = (b_{i1}, \ldots, b_{in}) \in F_q^n$ with $i = 1, \ldots, 10{,}000$, from an existing big data set, and use three ways to respectively calculate $\cos(\vec{a}, \vec{b})$, with $i = 1, \ldots, 10{,}000$ for performance evaluation in terms of total computation overhead. For each parameter setting, we run the experiments 50 times, and the average performance results over 50 runs are reported. Note that communication overhead is not currently considered in these experiments.

In Fig. 3, we plot the computation overheads of DCSC, PCSC, and PE-based varying with different parameter n. From the figure, we can see that by increasing n, the computation overhead of the PE-based protocol increase hugely

which is much higher than that of the direct cosine similarity computation. Therefore, it is not efficient in big data processing. Although the computation overhead of our proposed PCSC protocol also increases when n is large, it is still close to the computation overhead of the direct cosine similarity computation. Therefore, the experiment results show that our proposed PCSC protocol is not only privacy-preserving but also efficient. It is particularly suitable for big data analytics.

Conclusions

In this article, we have investigated the privacy challenges in the big data era by first identifying big data privacy requirements and then discussing whether existing privacy-preserving techniques are sufficient for big data processing. We have also introduced an efficient and privacy-preserving cosine similarity computing protocol in response to the efficiency and privacy requirements of data mining in the big data era. Although we have analyzed the privacy and efficiency challenges in general big data analytics to shed light on the privacy research in big data, significant research efforts should be further put into addressing unique privacy issues in some specific big data analytics.

References

[1] IBM, "Big Data at the Speed of Business," http://www-01.ibm.com/software/data/bigdata/, 2012.
[2] X. Wu *et al.*, "Data Mining with Big Data," *IEEE Trans. Knowledge Data Eng.*, vol. 26, no. 1, 2014, pp. 97–107.
[3] S. Liu, "Exploring the Future of Computing," *IT Professional*, vol. 15, no. 1, 2013, pp. 2–3.
[4] Oracle, "Oracle Big Data for the Enterprise," http://www.oracle.com/ca-en/technologies/big-data, 2012.
[5] "Big Data at CSAIL," http://bigdata.csail.mit.edu/.
[6] R. Lu *et al.*, "EPPA: An Efficient and Privacy-Preserving Aggregation Sheme for Secure Smart Grid Communications," *IEEE Trans. Parallel Distrib. Sys.*, vol. 23, no. 9, 2012, pp. 1621–31.
[7] P. Paillier, "Public-Key Cryptosystems based on Composite Degree Residuosity Classes," *EUROCRYPT*, 1999, pp. 223–38.
[8] M. Li *et al.*, "Toward Privacy-Assured and Searchable Cloud Data Storage Services," *IEEE Network*, vol. 27, no. 4, 2013, pp. 1–10.
[9] A. Cavoukian and J. Jonas, "Privacy by Design in the Age of Big Data," Office of the Information and Privacy Commissioner, 2012.
[10] R. Lu, X. Lin, and X. Shen, "SPOC: A Secure and Privacy-Preserving Opportunistic Computing Framework for Mobile-Healthcare Emergency," *IEEE Trans. Parallel Distrib. Sys.*, vol. 24, no. 3, 2013, pp. 614–24.

Biographies

RONGXING LU (rxlu@ntu.edu.sg) received his Ph.D. degree in computer science from Shanghai Jiao Tong University, China, in 2006 and his Ph.D. degree (awarded Canada Governor General Gold Medal) in electrical and computer engineering from the University of Waterloo, Canada, in 2012. Since May 2013, he has been an assistant professor at the School of Electrical and Electronics Engineering, Nanyang Technological University. His research interests include computer network security, mobile and wireless communication security, and big data security and privacy.

HUI ZHU (zhuhui@xidian.edu.cn) received his B.Sc. and Ph.D. from Xidian University, China, in 2003 and 2009. Since 2010, he has been an associate professor in the School of Telecommunications Engineering, Xidian University. His research interests are in the areas of wireless network security and information system security.

XIMENG LIU (snbnix@gmail.com) is currently a Ph.D. student in the School of Telecommunications Engineering at Xidian University, China, and also a visiting Ph.D. student at the School of Electrical and Electronics Engineering, Nanyang Technological University. His research interests include applied cryptography and big data security.

JOSEPH K. LIU (ksliu@i2r.a-star.edu.sg) received his Ph.D. degree in information engineering from the Chinese University of Hong Kong in July 2004, specializing in cryptographic protocols for securing wireless networks, privacy, authentication, and provable security. He is now a research scientist in the Infocomm Security Department at the Institute for Infocomm Research, Singapore. His current technical focus is particularly lightweight security, wireless security, and security in the big data and cloud computing environment.

JUN SHAO (chn.junshao@gmail.com) obtained his Ph.D. degree from Shanghai Jiaotong University in 2008. Soon after, he was a postdoctoral researcher at Pennsylvania State University until 2010. Now, he is an associate professor in Zhejiang Gongshang University. His research interests include applied cryptography and network security.