Cost-Effective Authentic and Anonymous Data Sharing with Forward Security

Xinyi Huang, Joseph K. Liu, Shaohua Tang, *Member, IEEE*, Yang Xiang, *Senior Member, IEEE*, Kaitai Liang, Li Xu, *Member, IEEE*, and Jianying Zhou

Abstract—Data sharing has never been easier with the advances of cloud computing, and an accurate analysis on the shared data provides an array of benefits to both the society and individuals. Data sharing with a large number of participants must take into account several issues, including efficiency, data integrity and privacy of data owner. Ring signature is a promising candidate to construct an anonymous and authentic data sharing system. It allows a data owner to anonymously authenticate his data which can be put into the cloud for storage or analysis purpose. Yet the costly certificate verification in the traditional public key infrastructure (PKI) setting becomes a bottleneck for this solution to be scalable. Identity-based (ID-based) ring signature, which eliminates the process of certificate verification, can be used instead. In this paper, we further enhance the security of ID-based ring signature by providing forward security: If a secret key of any user has been compromised, all previous generated signatures that include this user still remain valid. This property is especially important to any large scale data sharing system, as it is impossible to ask all data owners to re-authenticate their data even if a secret key of one single user has been compromised. We provide a concrete and efficient instantiation of our scheme, prove its security and provide an implementation to show its practicality.

Index Terms—Authentication, data sharing, cloud computing, forward security, smart grid

1 Introduction

The popularity and widespread use of "CLOUD" have brought great convenience for data sharing and collection [8], [29], [37], [46], [51]. Not only can individuals acquire useful data more easily, sharing data with others can provide a number of benefits to our society as well [14], [49], [53]. As a representative example, consumers in Smart Grid [40] can obtain their energy usage data in a fine-grained manner and are encouraged to share their personal energy usage data with others, e.g., by uploading the data to a third party platform such as Microsoft Hohm [39] (Fig. 1). From the collected data a statistical report is created, and one can compare their energy consumption with others (e.g., from the same block). This ability to access, analyze, and respond to much more precise and detailed data from all levels of the electric grid is critical to efficient energy usage.

Due to its openness, data sharing is always deployed in a hostile environment and vulnerable to a number of security threats. Taking energy usage data sharing in Smart Grid as an example, there are several security goals a practical system must meet, including:

- *Data Authenticity*. In the situation of smart grid, the statistic energy usage data would be misleading if it is forged by adversaries. While this issue alone can be solved using well established cryptographic tools (e.g., message authentication code or digital signatures), one may encounter additional difficulties when other issues are taken into account, such as anonymity and efficiency;
- *Anonymity*. Energy usage data contains vast information of consumers, from which one can extract the number of persons in the home, the types of electric utilities used in a specific time period, etc. Thus, it is critical to protect the anonymity of consumers in such applications, and any failures to do so may lead to the reluctance from the consumers to share data with others; and
- *Efficiency*. The number of users in a data sharing system could be HUGE (imagine a smart grid with a country size), and a practical system must reduce the computation and communication cost as much as possible. Otherwise it would lead to a waste of energy, which contradicts the goal of smart grid.

This paper is devoted to investigating fundamental security tools for realizing the three properties we described. Note that there are other security issues in a data sharing system which are equally important, such as *availability* (service is provided at an acceptable level even under network attacks) and *access control* (only eligible users can have the access to the data). But the study of those issues is out of the scope of this paper.

- *X. Huang and L. Xu are with the Fujian Provincial Key Laboratory of Network Security and Cryptology, School of Mathematics and Computer Science, Fujian Normal University, Fuzhou, China, 350108. E-mail: {xyhuang, xuli}@fjnu.edu.cn.*
- *J.K. Liu and J. Zhou are with the Infocomm Security Department, Institute for Infocomm Research, 1 Fusionopolis Way, #21-01 Connexis, South Tower, Singapore 138632. E-mail: {ksliu, jyzhou}@i2r.a-star.edu.sg.*
- *S. Tang is with the School of Computer Science, South China University of Technology, Guangzhou, China. E-mail: shtang@ieee.org.*
- *Y. Xiang is with the School of Information Technology, Deakin University, Australia. E-mail: yang@deakin.edu.au.*
- *K. Liang is with the Department of Computer Science, City University of Hong Kong, Hong Kong. E-mail: kliang4-c@my.cityu.edu.hk.*

Manuscript received 16 Sept. 2013; revised 15 Feb. 2014; accepted 16 Mar. 2014. Date of publication 2 Apr. 2014; date of current version 13 Mar. 2015. Recommended for acceptance by K. Li.
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier Below.
Digital Object Identifier no. 10.1109/TC.2014.2315619

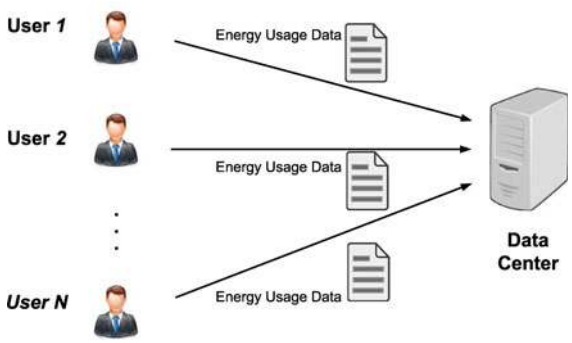


Fig. 1. Energy usage data sharing in smart grid.

1.1 Identity-Based Ring Signature

The aforementioned three issues remind us a cryptographic primitive "identity-based ring signature", an efficient solution on applications requiring data authenticity and anonymity.

1.1.1 ID-Based Cryptosystem

Identity-based (ID-based) cryptosystem, introduced by Shamir [45], eliminated the need for verifying the validity of public key certificates, the management of which is both time and cost consuming. In an ID-based cryptosystem, the public key of each user is easily computable from a string corresponding to this user's publicly known identity (e.g., an email address, a residential address, etc.). A private key generator (PKG) then computes private keys from its master secret for users. This property avoids the need of certificates (which are necessary in traditional public-key infrastructure) and associates an implicit public key (user identity) to each user within the system. In order to verify an ID-based signature, different from the traditional public key based signature, one does not need to verify the certificate first. The elimination of the certificate validation makes the whole verification process more efficient, which will lead to a significant save in communication and computation when a large number of users are involved (say, energy usage data sharing in smart-grid).

Ring signature is a group-oriented signature with privacy protection on signature producer. A user can sign anonymously on behalf of a group on his own choice, while group members can be totally unaware of being conscripted in the group. Any verifier can be convinced that a message has been signed by one of the members in this group (also called the Rings), but the actual identity of the signer is hidden. Ring signatures could be used for whistle blowing [42], anonymous membership authentication for ad hoc groups [11] and many other applications which do not want complicated group formation stage but require signer anonymity. There have been many different schemes proposed (e.g., [1], [13], [19], [23], [30], [31], [32], [33], [34], [38], [44], [50]) since the first appearance of ring signature in 1994 [21] and the formal introduction in 2001 [42].

1.1.2 An Affirmative Advantage in Big Data

Due to its natural framework, ring signature in ID-based setting has a significant advantage over its counterpart in traditional public key setting, especially in the big data analytic

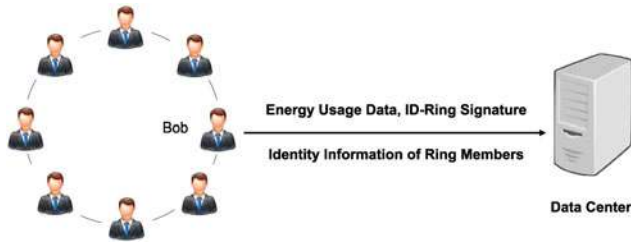


Fig. 2. A solution based on ID-based ring signature.

environment. Suppose there are 10,000 users in the ring, the verifier of a traditional public key based ring signature must first validate 10,000 certificates of the corresponding users, after which one can carry out the actual verification on the message and signature pair. In contrast, to verify an ID-based ring signature, only the identities of ring users, together with the pair of message and signature are needed. As one can see, the elimination of certificate validation, which is a costly process, saves a great amount of time and computation. This saving will be more critical if a higher level of anonymity is needed by increasing the number of users in the ring. Thus, as depicted in Fig. 2, ID-based ring signature is more preferable in the setting with a large number of users such as energy data sharing in smart grid:

- Step 1: The energy data owner (say, Bob) first setups a ring by choosing a group of users. This phase only needs the public identity information of ring members, such as residential addresses, and Bob does not need the collaboration (or the consent) from any ring members.
- Step 2: Bob uploads his personal data of electronic usage, together with a ring signature and the identity information of all ring members.
- Step 3: By verifying the ring signature, one can be assured that the data is indeed given out by a valid resident (from the ring members) while cannot figure out who the resident is. Hence the anonymity of the data provider is ensured together with data authenticity. Meanwhile, the verification is efficient which does not involve any certificate verification.

The first ID-based ring signature scheme was proposed in 2002 [57] which can be proven secure in the random oracle model. Two constructions in the standard model were proposed in [4]. Their first construction however was discovered to be flawed [24], while the second construction is only proven secure in a weaker model, namely, selective-ID model. The first ID-based ring signature scheme claimed to be secure in the standard model is due to Han et al. [25] under the trusted setup assumption. However, their proof is wrong and is pointed out by [48]. Other existing ID-based ring signature schemes include [5], [17], [18], [20], [26], [41], [48], [58].

1.2 The Motivation

1.2.1 Key Exposure

ID-based ring signature seems to be an optimal trade-off among efficiency, data authenticity and anonymity, and provides a sound solution on data sharing with a large number of participants. To obtain a higher level protection,

one can add more users in the ring. But doing this increases the chance of key exposure as well.

Key exposure is the fundamental limitation of ordinary digital signatures. If the private key of a signer is compromised, all signatures of that signer become worthless: future signatures are invalidated and no previously issued signatures can be trusted. Once a key leakage is identified, key revocation mechanisms must be invoked immediately in order to prevent the generation of any signature using the compromised secret key. However, this does not solve the problem of forgeability for past signatures.

The notion of forward secure signature was proposed to preserve the validity of past signatures even if the current secret key is compromised. The concept was first suggested by Anderson [2], and the solutions were designed by Bellare and Miner [7]. The idea is dividing the total time of the validity of a public key into T time periods, and a key compromise of the current time slot does not enable an adversary to produce valid signatures pertaining to past time slots.

1.2.2 Key Exposure in Big Data Sharing System

The issue of key exposure is more severe in a ring signature scheme: if a ring member's secret key is exposed, the adversary can produce valid ring signatures of any documents on behalf of that group. Even worse, the "group" can be defined by the adversary at will due to the spontaneity property of ring signature: The adversary only needs to include the compromised user in the "group" of his choice. As a result, the exposure of one user's secret key renders all previously obtained ring signatures invalid (if that user is one of the ring members), since one cannot distinguish whether a ring signature is generated prior to the key exposure or by which user. Therefore, forward security is a necessary requirement that a big data sharing system must meet. Otherwise, it will lead to a huge waste of time and resource.

While there are various designs of forward-secure digital signatures [54], [55], [56], adding forward security on ring signatures turns out to be difficult. As far as the authors know, there are only two forward secure ring signature schemes [35], [36]. However, they are both in the traditional public key setting where signature verification involves expensive certificate check for every ring member. This is far below satisfactory if the size of the ring is huge, such as the users of a smart grid. To summarize, the design of ID-based ring signature with forward security, which is the fundamental tool for realizing cost-effective authentic and anonymous data sharing, is still an open problem.

1.3 Contribution

In this paper, we propose a new notion called forward secure ID-based ring signature, which is an essential tool for building cost-effective authentic and anonymous data sharing system:

- For the first time, we provide formal definitions on forward secure ID-based ring signatures;
- We present a concrete design of forward secure ID-based ring signature. No previous ID-based ring signature schemes in the literature have the property of forward security, and we are the first to provide this feature;
- We prove the security of the proposed scheme in the random oracle model, under the standard RSA assumption; and
- Our implementation is practical, in the following ways:
  1) It is in ID-based setting. The elimination of the costly certificate verification process makes it scalable and especially suitable for big data analytic environment.
  2) The size of a secret key is just one integer.
  3) Key update process only requires an exponentiation.
  4) We do not require any pairing in any stage.

2 Definition

2.1 Mathematical Assumption

Definition 1 (RSA Problem). *Let $N = pq$, where p and q are two k-bit prime numbers such that $p = 2p' + 1$ and $q = 2q' + 1$ for some primes p', q'. Let e be a prime[1] greater than 2^ℓ for some fixed parameter ℓ, such that $gcd(e, \phi(N)) = 1$. Let y be a random element in $\mathbb{Z}_N^*$. We say that an algorithm $\mathcal{S}$ solves the RSA problem if it receives an input the tuple (N, e, y) and outputs an element z such that $z^e = y \bmod N$.*

2.2 Security Model

A $(1, n)$ ID-based forward secure ring signature (IDFSRS) scheme is a tuple of probabilistic polynomial-time (PPT) algorithms:

- Setup. On input an unary string 1^λ where λ is a security parameter, the algorithm outputs a master secret key msk for the third party private key generator and a list of system parameters param that includes λ and the descriptions of a user secret key space $\mathcal{D}$, a message space $\mathcal{M}$ as well as a signature space Ψ.
- Extract. On input a list param of system parameters, an identity $ID_i \in \{0,1\}^*$ for a user and the master secret key msk, the algorithm outputs the user's secret key $sk_{i,0} \in \mathcal{D}$ such that the secret key is valid for time $t = 0$. In this paper, we denote time as nonnegative integers. When we say identity ID_i corresponds to user secret key $sk_{i,0}$ or vice versa, we mean the pair $(ID_i, sk_{i,0})$ is an input-output pair of Extract with respect to param and msk.
- Update. On input a user secret key $sk_{i,t}$ for a time period t, the algorithm outputs a new user secret key $sk_{i,t+1}$ for the time period $t + 1$.
- Sign. On input a list param of system parameters, a time period t, a group size n of length polynomial in λ, a set $\mathcal{L} = \{ID_i \in \{0,1\}^* | i \in [1, n]\}$ of n user identities, a message $m \in \mathcal{M}$, and a secret key $sk_{\pi,t} \in \mathcal{D}, \pi \in [1, n]$ for time period t, the algorithm outputs a signature $\sigma \in \Psi$.

1. Note that we use a slightly modified version [22], [26] of the original RSA problem definition in which we require the exponent to be a prime number.

- Verify. On input a list param of system parameters, a time period t, a group size n of length polynomial in λ, a set $\mathcal{L} = \{ID_i \in \{0,1\}^* | i \in [1,n]\}$ of n user identities, a message $m \in \mathcal{M}$, a signature $\sigma \in \Psi$, it outputs either valid or invalid.

Correctness. A $(1,n)$ IDFSRS scheme should satisfy the *verification correctness*—signatures signed by honest signer are verified to be invalid with negligible probability.

2.3 Notions of Security

The security of IDFSRS consists of two aspects: forward security and anonymity. Before giving their definition, we consider the following oracles which together model the ability of the adversaries in breaking the security of IDFSRS.

- *Extration Oracle* ($\mathcal{EO}$). On input an identity ID_i and a time period t, the corresponding secret key $sk_{i,t} \in \mathcal{D}$ for that time period is returned.
- *Signing Oracle* ($\mathcal{SO}$). On input a time period t, a group size n, a set $\mathcal{L}$ of n user identities, a message $m \in \mathcal{M}$, a valid signature σ is returned.

Now we are ready to define the security of IDFSRS:

1) *Forward Security*. Forward security of IDFSRS scheme is defined in the following game between the simulator $\mathcal{S}$ and the adversary $\mathcal{A}$ in which $\mathcal{A}$ is given access to oracles $\mathcal{EO}$ and $\mathcal{SO}$:
   a) $\mathcal{S}$ generates and gives $\mathcal{A}$ the system parameters param.
   b) $\mathcal{A}$ may query the oracles according to any adaptive strategy.
   c) $\mathcal{A}$ chooses a time t^*, a group size $n^* \in \mathbb{N}$, a set $\mathcal{L}^*$ of n^* identities and a message $m^* \in \mathcal{M}$.
   d) $\mathcal{A}$ may continue to query the oracles according to any adaptive strategy.
   e) $\mathcal{A}$ outputs a signature $\sigma^*_{t^*}$.

   $\mathcal{A}$ wins the game if:
   - $\mathsf{Verify}(t^*, \mathcal{L}^*, m^*, \sigma^*_{t^*}) = \mathsf{valid}$.
   - None of the identities in $\mathcal{L}^*$ has been queried to $\mathcal{EO}$ with time $t \le t^*$ as the time input parameter. (Unlimited query to $\mathcal{EO}$ with time $t > t^*$ to be the time input parameter.)
   - $(t^*, \mathcal{L}^*, m^*)$ are not queried to $\mathcal{SO}$.

   We denote $\mathbf{Adv}^{fs}_{\mathcal{A}}(\lambda)$ the probability of $\mathcal{A}$ winning the game.

   Definition 2 (Forward Secure). *A $(1,n)$ IDFSRS scheme is forward secure if for any PPT adversary $\mathcal{A}$, $\mathbf{Adv}^{fs}_{\mathcal{A}}(\lambda)$ is a negligible function of λ.*
2) *Anonymity*. It should not be possible for an adversary to tell the identity of the signer with a probability larger than $1/n$, where n is the cardinality of the ring, even assuming that the adversary has unlimited computing resources. We denote

$$\mathbf{Adv}^{anon}_{\mathcal{A}}(\lambda) = \Pr[\mathcal{A} \text{ guesses the identity of the signer correctly}] - 1/n.$$

Definition 3 (Anonymity). *A $(1,n)$IDFSRS scheme is unconditional anonymous if for any group of n users, any message $m \in \mathcal{M}$, any time t and any valid signature, any adversary $\mathcal{A}$, even with unbounded computational power, cannot identify the actual signer with probability better than random guessing. In other words, $\mathcal{A}$ can only output the identity of the actual signer with probability no better than $1/n$. That is, $\mathbf{Adv}^{anon}_{\mathcal{A}}(\lambda) = 0$.*

3 Our Proposed ID-Based Forward Secure Ring Signature Scheme

This section is devoted to the description and analysis of our proposed ID-based forward secure ring signature scheme.

3.1 The Design

We assume that the identities and user secret keys are valid into T periods and makes the time intervals public. We also set the message space $\mathcal{M} = \{0,1\}^*$.

- Setup. On input of a security parameter λ, the PKG generates two random k-bit prime numbers p and q such that $p = 2p' + 1$ and $q = 2q' + 1$ where p', q' are some primes. It computes $N = pq$. For some fixed parameter ℓ, it chooses a random prime number e such that $2^\ell < e < 2^{\ell+1}$ and $gcd(e, \phi(N)) = 1$. It chooses two hash functions $H_1 : \{0,1\}^* \to \mathbb{Z}^*_N$ and $H_2 : \{0,1\}^* \to \{0,1\}^\ell$. The public parameters param are $(k, \ell, e, N, H_1, H_2)$ and the master secret key msk is (p,q).
- Extract. For user i, where $i \in \mathbb{Z}$, with identity $ID_i \in \{0,1\}^*$ requests for a secret key at time period t (denoted by an integer), where $0 \le t < T$, the PKG computes the user secret key

$$sk_{i,t} = [H_1(ID_i)]^{\frac{1}{e^{(T+1-t)}}} \bmod N$$

  using its knowledge of the factorization of N.
- Update. On input a secret key $sk_{i,t}$ for a time period t, if $t < T$ the user updates the secret key as

$$sk_{i,t+1} = (sk_{i,t})^e \bmod N.$$

  Otherwise the algorithm outputs "$\perp$" meaning that the secret key has expired.
- Sign. To sign a message $m \in \{0,1\}^*$ in time period t, where $0 \le t < T$, on behalf of a ring of identities $\mathcal{L} = \{ID_1, \ldots, ID_n\}$, a user with identity $ID_\pi \in \mathcal{L}$ and secret key $sk_{\pi,t}$:
  1) For all $i \in \{1, \ldots, n\}, i \ne \pi$, choose random $A_i \in \mathbb{Z}^*_N$ and compute

$$R_i = {A_i}^{e^{(T+1-t)}} \bmod N$$

  and

$$h_i = H_2(\mathcal{L}, m, t, ID_i, R_i).$$

  2) Choose random $A_\pi \in \mathbb{Z}^*_N$ and compute

$$R_\pi = {A_\pi}^{e^{(T+1-t)}} \cdot \prod_{i=1, i\ne\pi}^{n} H_1(ID_i)^{-h_i} \bmod N$$

and

$$h_\pi = H_2(\mathcal{L}, m, t, ID_\pi, R_\pi).$$

3) Compute $s = (sk_{\pi,t})^{h_\pi} \cdot \prod_{i=1}^{n} A_i \bmod N$.
4) Output the signature for the list of identities $\mathcal{L}$, the message m, and the time period t as $\sigma = (R_1, \ldots, R_n, h_1, \ldots, h_n, s)$.

- Verify. To verify a signature σ for a message m, a list of identities $\mathcal{L}$ and the time period t, check whether $h_i = H_2(\mathcal{L}, m, t, ID_i, R_i)$ for $i = 1, \ldots, n$ and

$$s^{e^{(T+1-t)}} = \prod_{i=1}^{n} (R_i \cdot H_1(ID_i)^{h_i}) \bmod N.$$

Output valid if all equalities hold. Otherwise output invalid.

3.2 Correctness

We shall show that signatures signed by honest signers are always verified to be valid. For the verification equation

$$s^{e^{(T+1-t)}} = \prod_{i=1}^{n} \left(R_i \cdot H_1(ID_i)^{h_i}\right) \bmod N:$$

Left Hand Side:

$$\begin{aligned}
&= s^{e^{(T+1-t)}} \\
&= \left((sk_{\pi,t})^{h_\pi} \cdot \prod_{i=1}^{n} A_i \bmod N \right)^{e^{(T+1-t)}} \\
&= \left(\left(H_1(ID_\pi)^{\frac{1}{e^{(T+1-t)}}} \right)^{h_\pi} \cdot \prod_{i=1}^{n} A_i \bmod N \right)^{e^{(T+1-t)}} \\
&= H_1(ID_\pi)^{h_\pi} \cdot \prod_{i=1}^{n} (A_i)^{e^{(T+1-t)}} \bmod N.
\end{aligned}$$

Right Hand Side:

$$\begin{aligned}
&= \prod_{i=1}^{n} \left(R_i \cdot H_1(ID_i)^{h_i}\right) \bmod N \\
&= \left(\prod_{i=1, i\neq\pi}^{n} \left(R_i \cdot H_1(ID_i)^{h_i}\right) \right) \cdot \left(R_\pi \cdot H_1(ID_\pi)^{h_\pi} \right) \bmod N \\
&= \left(\prod_{i=1, i\neq\pi}^{n} \left({A_i}^{e^{(T+1-t)}} \cdot H_1(ID_i)^{h_i} \right) \right) \\
&\quad \cdot \left({A_\pi}^{e^{(T+1-t)}} \cdot \prod_{i=1, i\neq\pi}^{n} H_1(ID_i)^{-h_i} \cdot H_1(\mathsf{ID}_\pi)^{h_\pi} \right) \bmod N \\
&= \left(\prod_{i=1}^{n} {A_i}^{e^{(T+1-t)}} \right) \cdot H_1(ID_\pi)^{h_\pi} \bmod N \\
&= \text{Left hand side.}
\end{aligned}$$

3.3 Security Analysis

Theorem 1 (Forward Security). *Let $\mathcal{A}$ be a PPT forger. For some message m and a set $\mathcal{L}$ containing n identities, suppose $\mathcal{A}$ on inputs the security parameter λ, queries an extraction oracle q_E times, a signing oracle q_s times, the H_1 random oracle q_{H_1} times and the H_2 random oracle q_{H_2} times, outputs a forged signature with non-negligible probability ϵ. Then we can solve the strong RSA problem with probability at least ϵ' in polynomial time where $\epsilon' \geq \frac{\epsilon^2}{1560 q_e V_{q_{H_2}, n} T}$, and $V_{q_{H_2}, n} = q_{H_2} \cdot (q_{H_2} - 1) \ldots (q_{H_2} - n + 1)$ and T is the total number of time period.*

Proof. Our proof consists of the following parts.

Setup. Let (N, e, y) be an instance of the RSA problem as stated in Definition 1. We are going to construct a PPT algorithm $\mathcal{S}$ that will use the adversary $\mathcal{A}$ to solve the given instance of the strong RSA problem. $\mathcal{S}$ must simulate all necessary oracles and responds to $\mathcal{A}$'s queries.

First, $\mathcal{S}$ sends the public parameter (N, e) to $\mathcal{A}$. Without loss of generality, we can assume that $\mathcal{A}$ asks the random oracle H_1 for the value $H_1(ID)$ before asking for the secret key of ID. We define t^* be the breakin period such that $\mathcal{A}$ is not allowed to query $\mathcal{EO}$ for any identities included in $\mathcal{L}^*$ (the set of identities of the forged signature output by $\mathcal{A}$), while there is no limitation for time input parameter $t > t^*$. $\mathcal{A}$ is allowed to choose any $t^* \leq T$. $\mathcal{S}$ needs to guess the breakin period t^* chosen by $\mathcal{A}$. $\mathcal{S}$ randomly chooses $\hat{t}$, $1 < \hat{t} \leq T$, hoping that the breakin period falls at $\hat{t}$ or later, so that the forgery will be for a time period earlier than $\hat{t}$.

Oracle Simulation.

- H_1 random oracle. Let us define $\mu = (5/6)^{1/q_e}$. $\mathcal{S}$ constructs a table TAB_{H_1} to simulate the random oracle H_1. Every time an identity ID_i is asked by $\mathcal{A}$, $\mathcal{S}$ acts as follows: first $\mathcal{S}$ checks if this input is already in the table. If it is the case, $\mathcal{S}$ sends to $\mathcal{A}$ the corresponding value $H_1(ID_i) = h_{ID_i}$. Otherwise, $\mathcal{S}$ chooses a bit $\beta_i \in \{0, 1\}$, with $\Pr[\beta_i = 0] = \mu$ and $\Pr[\beta_i = 1] = 1 - \mu$. Then $\mathcal{S}$ chooses at random a different element $x_i \in \mathbb{Z}_N^*$ and defines $h_{ID_i} = y^{\beta_i e^{(T+1-\hat{t})}} {x_i}^{e^{(T+1)}} \bmod N$. The entry $(ID_i, h_{ID_i}, x_i, \beta_i)$ is stored in the table TAB_{H_1}. The value $H_1(ID_i) = h_{ID_i}$ is sent to $\mathcal{A}$. The condition $h_{ID_i} \neq h_{ID_j}$ must be satisfied for all different entries $i \neq j$ of the table. If this is not the case, $\mathcal{S}$ repeats this process.
- H_2 random oracle. $\mathcal{S}$ constructs another table TAB_{H_2} to simulate the random oracle H_2. If the input element is stored in the table, $\mathcal{S}$ returns the corresponding value as the output. Otherwise, $\mathcal{S}$ randomly chooses an element $r \in \{0, 1\}^\ell$ as the output, and stores the value in the table for consistency.
- Extraction Oracle $(\mathcal{EO})$. When $\mathcal{A}$ asks for the secret key of a user with identity ID_i at time period t, $\mathcal{S}$ looks for ID_i in the table TAB_{H_1}. If $\beta_i = 0$, then $\mathcal{S}$ extracts the x_i value and computes $sk_{i,t} = {x_i}^{e^t} \bmod N$ as the secret key of the user with identity ID_i at time t. It returns $sk_{i,t}$ to $\mathcal{A}$. If $\beta_i = 1$, $\mathcal{S}$ halts if $t < \hat{t}$. Otherwise (that is, $t \geq \hat{t}$), $\mathcal{S}$ extracts the x_i value and computes $sk_{i,t} = y^{e^{(t-\hat{t})}} {x_i}^{e^t} \bmod N$ as the secret key of the user with identity ID_i at time t. Note that the

probability that $\mathcal{S}$ halts in this step is less than $1-\mu^{q_e}=1/6$.

- Signing Oracle ($\mathcal{SO}$). When $\mathcal{A}$ asks for a valid signature for message m for a set of identities $\mathcal{L}$ containing n' identities in the time period t, where $n' \leq n$, $\mathcal{S}$ simulates the signing oracle. First we assume $\mathcal{A}$ has not asked for the secret key of any identities in $\mathcal{L}$, otherwise $\mathcal{A}$ could obtain a valid ring signature by itself. We also assume that $\mathcal{A}$ has asked for the H_1 random oracle query for $H_1(ID_i)$ for all identities $ID_i \in \mathcal{L}$. Therefore there exist entries $(ID_i, h_{ID_i}, x_i, \beta_i)$ in the table TAB_{H_1}, for all $i \in \{1,\ldots,n'\}$. $\mathcal{S}$ answers the query as follow:
  - If $\beta_i = 0$ for some $i \in \{1,\ldots,n'\}$, or $t \geq \hat{t}$, $\mathcal{S}$ knows the corresponding secret key by simulating the $\mathcal{EO}$ and uses the secret key to compute a valid signature according to the algorithm.
  - If $\beta_i = 1$ for all $i \in \{1,\ldots,n'\}$ and $t < \hat{t}$:
    1) $\mathcal{S}$ randomly chooses $\pi \in \{1,\ldots,n'\}$.
    2) For all $i \in \{1,\ldots,n'\}, i \neq \pi$, choose random $A_i \in \mathbb{Z}_N^*$, pairwise different, compute
    $$R_i = A_i^{e^{(T+1-t)}} \bmod N,$$
    and by querying the random oracle H_2, compute $h_i = H_2(\mathcal{L}, m, t, ID_i, R_i)$.
    3) Choose random $h_\pi \in \{0,1\}^\ell$ and $s \in \mathbb{Z}_N^*$.
    4) Compute
    $$R_\pi = s^{e^{(T+1-t)}} \cdot H_1(ID_\pi)^{-h_\pi} \cdot \prod_{i=1, i\neq\pi}^{n'} \left(R_i^{-1} \cdot H_1(ID_i)^{-h_i}\right) \bmod N.$$
    If $R_\pi = 1 \bmod N$ or $R_\pi = R_i$ for some $i \neq \pi$, go back to the previous step.
    5) Backpack the random oracle H_2 by setting $H_2(\mathcal{L}, m, t, ID_\pi, R_\pi) = h_\pi$.
    6) Return the signature $\sigma = (R_1,\ldots,R_{n'}, h_1,\ldots,h_{n'}, s)$.

Forgery. Assume $\mathcal{A}$ chooses a breakin period $t^* < \hat{t}$. That is, the forged signature σ^* is valid for time period $t^* < \hat{t}$. $\mathcal{S}$ randomly chooses an H_2 query and hopefully it is the H_2 query for closing the gap of the ring. Using standard rewind technique, $\mathcal{S}$ rewinds to the point just before supplying the answer of the H_2 query, and supplies a different answer to this particular query. Denote $\sigma^* = (R_1^*,\ldots,R_{n'}^*, h_1^*,\ldots,h_{n'}^*, s^*)$ as the first signature forgery given by $\mathcal{A}$. If $\mathcal{S}$ guesses correctly, $\mathcal{A}$ outputs a different signature $\sigma'^* = (R_1'^*,\ldots,R_{n'}'^*, h_1'^*,\ldots,h_j'^*,\ldots, h_{n'}'^*, s'^*)$ such that

- For all $i \in \{1,\ldots,n'\}$, $R_i^* = R_i'^*$.
- For all $i \in \{1,\ldots,n'\}, i \neq j$, $h_i^* = h_i'^*$. But $h_j^* \neq h_j'^*$ for one $j \in \{1,\ldots,n'\}$.
- $s^* \neq s'^*$.

Then we can obtain

$$s^{*e^{(T+1-t^*)}} = \prod_{i=1}^{n'} \left(R_i^* \cdot H_1(ID_i)^{h_i^*}\right) \bmod N$$

and

$$s'^{*e^{(T+1-t^*)}} = \prod_{i=1}^{n'} \left(R_i^* \cdot H_1(ID_i)^{h_i'^*}\right) \bmod N.$$

Dividing two equations, we obtain

$$\left(\frac{s^*}{s'^*}\right)^{e^{(T+1-t^*)}} = H_1(ID_j)^{h_j^* - h_j'^*} \bmod N.$$

Now we look at the table TAB_{H_1} for the entry $(ID_j, h_{ID_j}, x_j, \beta_j)$ corresponding to the identity ID_j. Since the forged signatures are valid and the secret key of user ID_j for any time $t \leq t^* < \hat{t}$ has not been queried, with probability $1-\mu$, we have $\beta_j = 1$ and $h_{ID_j} = H_1(ID_j) = y^{e^{(T+1-\hat{t})}} x_j^{e^{(T+1)}} \bmod N$. Then the relation becomes

$$\left(\frac{s^*}{s'^*}\right)^{e^{(T+1-t^*)}} \cdot x_j^{(h_j'^* - h_j^*)\cdot e^{(T+1)}} = y^{e^{(T+1-\hat{t})}\cdot(h_j^* - h_j'^*)} \bmod N. \quad (1)$$

After simplification, equation (1) becomes

$$\left(\frac{s^*}{s'^*}\right)^{e^{(\hat{t}-t^*)}} \cdot x_j^{(h_j'^* - h_j^*)\cdot e^{\hat{t}}} = y^{h_j^* - h_j'^*} \bmod N. \quad (2)$$

Since h_j^* and $h_j'^*$ are outputs of the hash function $H_2 : \{0,1\}^* \to \{0,1\}^\ell$, we have $|h_j^* - h_j'^*| < 2^\ell < e$. Also as e is a prime number, it holds that $gcd(e, |h_j^* - h_j'^*|) = 1$. That means there exists two integers a and b such that $ae + b(h_j^* - h_j'^*) = 1$. Finally we have

$$z = \left(\left(\frac{s^*}{s'^*}\right)^{e^{(\hat{t}-t^*-1)}} \cdot x_j^{(h_j'^* - h_j^*)\cdot e^{\hat{t}-1}}\right)^b \cdot y^a \bmod N$$

as the solution of the given instance of the RSA problem. It is the correct solution because

$$\begin{aligned} z^e &= \left(\left(\frac{s^*}{s'^*}\right)^{e^{(\hat{t}-t^*-1)}} \cdot x_j^{(h_j'^* - h_j^*)\cdot e^{\hat{t}-1}}\right)^{eb} \cdot y^{ea} \bmod N \\ &= y^{(h_j^* - h_j'^*)b} \cdot y^{ea} \bmod N \text{ (from Equation (2))} \\ &= y^{ae + b(h_j^* - h_j'^*)} \bmod N \\ &= y \bmod N. \end{aligned}$$

Probability Analysis. For a successful simulation, there should be no collision in any stage. There are two kinds of collisions that may occur:

1) A tuple $(\mathcal{L}, m, t, ID_i, R_i)$ that $\mathcal{S}$ outputs from the signing oracle, inside a simulated ring signature, has been asked before to the random oracle H_2 by $\mathcal{A}$. The probability of such a collision is less than $q_2 \cdot q_s \cdot \frac{1}{2^k} \leq \frac{1}{6}$.
2) The same tuple $(\mathcal{L}, m, t, ID_i, R_i)$ is output by $\mathcal{S}$ in two different simulated ring signatures. Using the

TABLE 1
Computation Complexity of Algorithms

	Exponentiation	Multiplicative inverse	Multiplication	Hash
Setup	-	-	1	-
Extract (for 1 user)	2	1	1	1
Update	1	-	-	-
Sign	$3 \times n \times \ell/\ell'$	-	$(2 \times n - 1) \times \ell/\ell'$	$n \times \ell/\ell'$
Verify	$(n + 2) \times \ell/\ell'$	-	$(2 \times n - 1) \times \ell/\ell'$	$n \times \ell/\ell'$

Notation:
n: number of users in the ring;
ℓ': length of the output of hash function in the scheme;
ℓ: length of the output of a secure hash function (usually it is 160).

birthday paradox from probability theory, we have the probability of this collision is less than $\frac{q_s^2}{2} \cdot \frac{1}{2^k} \leq \frac{1}{6}$.

Altogether the probability of collision is less than $1/3$. Now we can compute the probability that $\mathcal{S}$ obtains a valid signature:

$$\begin{aligned}\epsilon_{\mathcal{S}} &= \Pr[\mathcal{S} \text{ succeeds}] \\ &= \Pr[\mathcal{S} \text{ does not halt AND no collisions in the simulations AND } \mathcal{A} \text{ succeeds}] \\ &\geq \Pr[\mathcal{A} \text{ succeeds } | \mathcal{S} \text{ does not halt AND no collisions in the simulations}] \\ &\quad \cdot (1 - \Pr[\mathcal{S} \text{ halts OR collisions in the simulations}]) \\ &\geq \epsilon \cdot \left(1 - \frac{1}{6} - \frac{1}{3}\right) \\ &= \frac{\epsilon}{2}.\end{aligned}$$

By the ring forking lemma [27], $\mathcal{S}$ obtains two valid ring signatures for the same message and same ring such that $h_j^* \neq h_j'^*$ with probability

$$\tilde{\epsilon} \geq \frac{\epsilon_{\mathcal{S}}^2}{65 V_{q_{H_2},n}},$$

where

$$V_{q_{H_2},n} = q_{H_2} \cdot (q_{H_2} - 1) \ldots (q_{H_2} - n + 1).$$

In additional, $\mathcal{S}$ also needs to guess the breaking period correctly. The probability is at least $\frac{1}{T}$. Summing up, the probability of $\mathcal{S}$ in solving the given instance of the RSA problem is

$$\begin{aligned}\epsilon' &\geq \frac{1}{T} \cdot (1 - \mu)\tilde{\epsilon} \\ &\geq (1 - \mu) \cdot \frac{\epsilon_{\mathcal{S}}^2}{65 V_{q_{H_2},n} T} \\ &\geq (1 - \mu) \cdot \frac{\left(\frac{\epsilon}{2}\right)^2}{65 V_{q_{H_2},n} T} \\ &\geq \frac{\epsilon^2}{1560 q_e V_{q_{H_2},n} T}.\end{aligned}$$

□

Theorem 2. *Our proposed scheme is unconditional anonymous.*

Proof. Given a valid signature $\sigma = (R_1, \ldots, R_n, h_1, \ldots, h_n, s)$, all R_i are generated from A_i, where A_i is a random number. h_i are outputs from the hash function H_2 which are also random numbers in the random oracle model. s is determined by all A_is. Thus, given a message and a ring, valid signatures produced by two different signers have the same distribution, and the signature gives no information about the identity of the actual signer. □

3.4 Efficiency Analysis

Our scheme requires the exponent e greater than 2^ℓ, where ℓ is the bit length of the hash function H_2. Usually a secure hash function requires at least 160 bits output. However, if we set $\ell = 160$ it will be quite inefficient. In order to offset this, we can use γ different hash functions such that each hash function outputs ℓ' bits where $\gamma\ell' = \ell$. For example, we can use 8 different hash functions such that each outputs 20 bits. In this way, we only need to set $2^{20} < e < 2^{21}$ which is still an acceptable value. But we need to repeat the signing and verification procedures (those require H_2 operation) 8 times for each time using a different hash function H_2 in order to achieve 160-bit hash function security.

On the other side, the size of public parameters is a constant, which only consists of some security parameters, two integers and some hash functions. The secret key is very short. It is only an integer. Assume we use 1,024-bit RSA security level, the secret key is just 1,024 bits. For every key update process, we just require an exponentiation with exponent e over modulus N operation. The signing and verification algorithms do not require any pairing operation.

The computation complexity and space requirement of our scheme are shown in Tables 1 and 2 respectively.

3.5 Comparison

We compare our scheme with related work in terms of features, computation, and space requirement, as shown in Table 3. Note that the verification for non ID-based 1-out-of-n ring signature schemes require additional certificate verification for n users. We exclude the cost and space for those n certificates verification in this comparison, as it may vary in different scenarios.

3.6 Implementation and Experimental Results

We implement the smart grid example introduced in Section 1, and evaluate the performance of our IDFSRS scheme with respect to three entities: the private key generator, the energy data owner (user), and the service provider

TABLE 2
Space Requirement

	Space required
Public parameters	$\mathcal{O}(1)$ (4 integers + descriptions of 2 hash functions)
Secret key	$\|N\|$ bits
Signature	$(n \times (\|N\| + \ell') + \|N\|) \times \ell/\ell'$ bits

, *Notation:*
N: RSA modulus;
$|N|$: the length of N in binary bits;
n: number of users in the ring.

TABLE 3
Comparison with Other Forward Secure Ring Signature Schemes

		[35]	[36]	Our Scheme
Features	Unconditional Anonymity	✓	X	✓
	ID-based	X	X	✓
	Assumption	Factorization	CDH, Subgroup Decisional	RSA
	Without ROM	X	✓	X
Computation complexity of Sign	Pairing	0	0	0
	Multiplication	$n^2 + 2$	$3 \times n + 5 + \|H\| + \log_2(T)$	$(2 \times n - 1) \times \ell/\ell'$
	Exponentiation	n	$2 \times n + 8$	$3 \times n \times \ell/\ell'$
Computation complexity of Verify	Pairing	0	$n + 4$	0
	Multiplication	$n^2 + n$	$3 \times n + 2 + \|H\| + \log_2(T)$	$(2 \times n - 1) \times \ell/\ell'$
	Exponentiation	$n^2 + n$	0	$(n + 2) \times \ell/\ell'$
Space Requirement	Secret Key (bits)	$2 \times \|N\|$	$2 \times (\log_2(T) + 1) \times \|\mathbb{G}\|$	$\|N\|$
	Signature (bits)	$n \times (\|N\| + \ell')$	$(2 \times n + 3) \times \|\mathbb{G}\|$	$(n \times (\|N\| + \ell) + \|N\|) \times \ell/\ell'$

Notation:
ROM: Random Oracle Model;
T: number of time slots;
$|H|$: the length of the hash of the message in binary bits;
$|N|$: the length of N in binary bits;
$|\mathbb{G}|$: the length of a group element in $\mathbb{G}$ for elliptic curve group; (If 160 bit elliptic curve is used, $|\mathbb{G}| = 160$ bits.)
n: number of users in the ring;
ℓ: a fixed integer.

(data center). In the experiments, the programs for three entities are implemented using the public cryptographic library MIRACL [43], programmed in C++. All experiments were repeated 100 times to obtain average results shown in this paper, and all experiments were conducted for the cases of $|N| = 1{,}024$ bits and $|N| = 2{,}048$ bits respectively.

The average time for the PKG to setup the system is shown in Table 4, where the testbed for the PKG is a DELL T5500 workstation equipped with 2.13 GHz Intel Xeon dual-core dual-processor with 12 GB RAM and running Windows 7 Professional 64-bit operating system. It took 151 and 2,198 ms for the PKG to setup the whole system for $|N| = 1{,}024$ bits and $|N| = 2{,}048$ bits respectively.

TABLE 4
Average Time for the PKG to Setup the System

$\|N\|$ (bits)	Time (ms)
1024	151
2048	2198

The average time for the data owner (user) to sign energy usage data with different choices of n and T are shown in Figs. 3 and 4, for $|N| = 1{,}024$ bits and $|N| = 2{,}048$ bits respectively. The testbed for the user is a laptop personal computer equipped with 2.10 GHz Intel CPU with 4 GB RAM and running Windows 7 operating system.

The average time for the service provider (data center) to verify the ring signature with different choices of n and T are shown in Figs. 5 and 6, for $|N| = 1{,}024$ bits and $|N| = 2{,}048$ bits respectively. The testbed for the date center is a DELL T5500 workstation equipped with 2.13 GHz Intel Xeon dual-core dual-processor with 12 GB RAM and running Windows 7 Professional 64-bit operating system.

4 Applications of Forward Secure ID-Based Ring Signatures

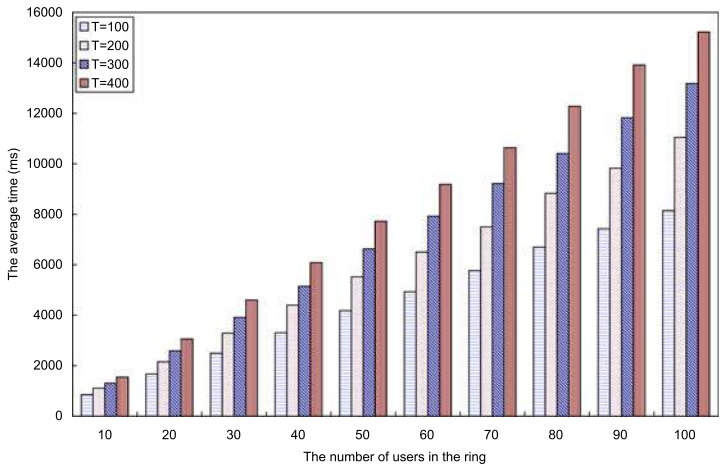
In addition to energy data sharing in smart-grid, we sketch three other situations which may also need forward secure ID-based ring signatures.

(Unit: ms)

	T=100	T=200	T=300	T=400
n=10	852	1108	1310	1545
n=20	1669	2152	2590	3057
n=30	2496	3291	3916	4602
n=40	3307	4399	5148	6084
n=50	4180	5522	6630	7722
n=60	4929	6505	7924	9188
n=70	5768	7503	9220	10639
n=80	6693	8830	10406	12277
n=90	7426	9828	11825	13916
n=100	8144	11045	13182	15226

Parameters: $|N| = 1024$, $|k| = 512$, $|\ell| = 160$.

(a)



(b)

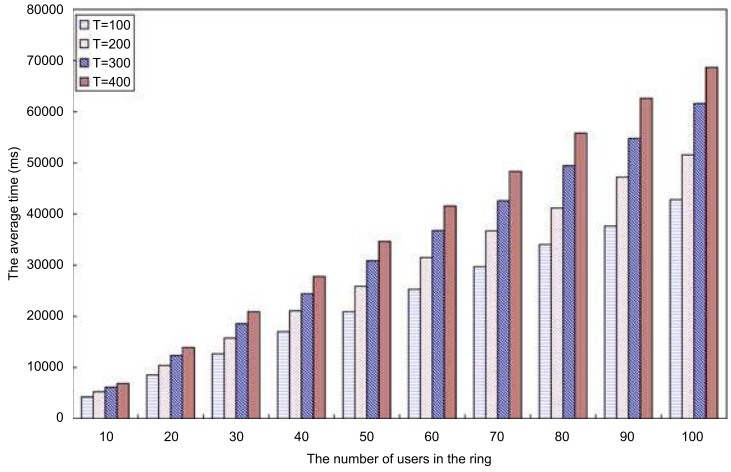
Fig. 3. The average time for the data owner to sign energy usage data, $|N| = 1{,}024$.

(Unit: ms)

	T=100	T=200	T=300	T=400
n=10	4238	5241	6162	6880
n=20	8548	10405	12355	13915
n=30	12683	15756	18579	20920
n=40	17004	21076	24430	27799
n=50	20904	25881	30872	34664
n=60	25287	31496	36785	41574
n=70	29718	36722	42604	48345
n=80	34039	41200	49468	55817
n=90	37674	47221	54788	62634
n=100	42822	51574	61635	68675

Parameters: $|N| = 2048$, $|k| = 1024$, $|\ell| = 320$.

(a)



(b)

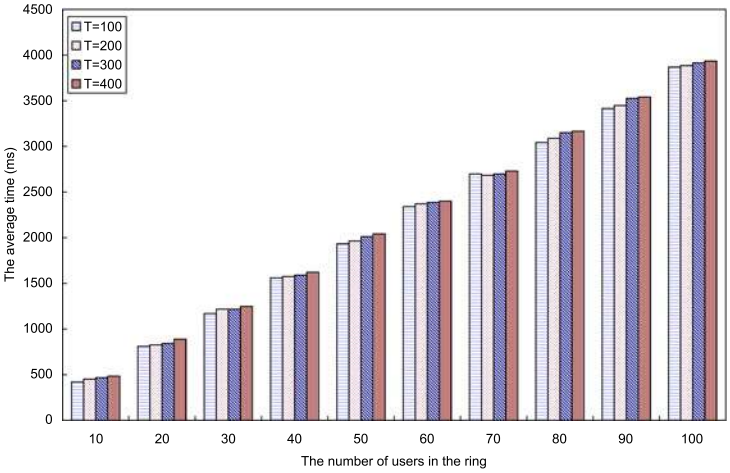
Fig. 4. The average time for the data owner to sign energy usage data, $|N| = 2{,}048$.

(Unit: ms)

	T=100	T=200	T=300	T=400
n=10	421	452	468	484
n=20	811	827	843	889
n=30	1170	1217	1217	1248
n=40	1560	1576	1591	1622
n=50	1934	1965	2013	2043
n=60	2340	2372	2387	2401
n=70	2698	2683	2699	2730
n=80	3042	3089	3151	3167
n=90	3416	3448	3526	3541
n=100	3869	3885	3915	3935

Parameters: $|N| = 1024$, $|k| = 512$, $|\ell| = 160$.

(a)



(b)

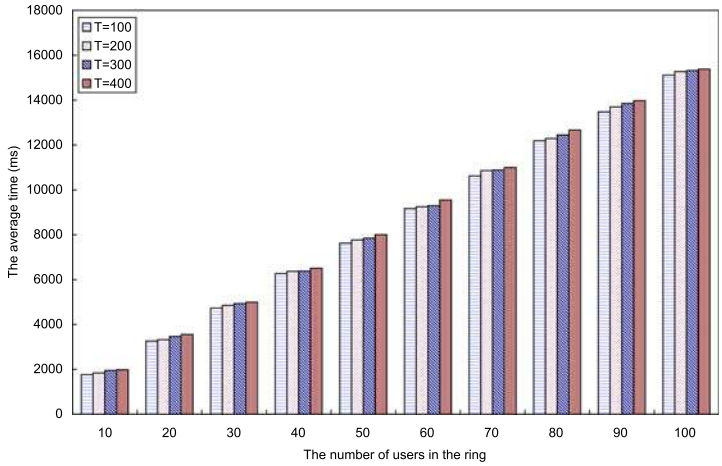
Fig. 5. The average time for the service provider to verify the ring signature, $|N| = 1{,}024$.

(Unit: ms)

	T=100	T=200	T=300	T=400
n=10	1763	1841	1950	1981
n=20	3261	3323	3464	3557
n=30	4727	4851	4930	4992
n=40	6271	6365	6380	6505
n=50	7628	7769	7846	8003
n=60	9173	9250	9297	9547
n=70	10623	10858	10889	10998
n=80	12184	12293	12449	12667
n=90	13479	13697	13853	13978
n=100	15116	15272	15319	15382

Parameters: $|N| = 2048$, $|k| = 1024$, $|\ell| = 320$.

(a)



(b)

Fig. 6. The average time for the service provider to verify the ring signature, $|N| = 2{,}048$.

WHISTLE BLOWING. Suppose Bob is a member of the city council. One day he wishes to leak a secret news from the council meeting to a journalist. The news is supposed to be kept secret. Thus Bob wants to remain anonymous, yet such that the journalist is convinced that the leak was indeed from a council member. Bob cannot send to the journalist a standard digitally signed message, since such a message, although it convinces the journalist that it came from a council member, does so by directly revealing Bob's identity. Neither does it work for Bob to send the journalist a message through a standard anonymizer, since the anonymizer strips off all source identification and authentication in a way that the journalist would have no reason to believe that the message really came from a council member at all. Using another primitive called group signature [3], [6], [9], [10], [12], [15] does not solve the problem neither. A group signature allows a signer to sign a message on behalf of a group. The verifier only knows that one of the users of the group signs the message yet does not know who is the actual signer. It does not work in this case, because it requires prior cooperation of the other group members to set up, and leaves Bob vulnerable to later identification by the group manager, who may be controlled by the government. The correct approach for Bob is to send the secret information to the journalist through an anonymizer, signed with a ring signature that names each council member including himself as a ring member. The journalist can verify the ring signature on the message, and learn that it definitely came from a council member. However, neither he nor anyone (including those council members inside the ring) can determine the actual source of the leak. Forward security enhances the protection of all entities. Without forward security, if a secret key of a council member Alice is exposed, every ring signature containing Alice in the ring will become invalid. That means any previous ring signature given by Bob will be invalid (assuming Alice is included in the signature). This will greatly affect the accuracy of the report by the journalist who may rely on Bob for leaking important secret information.

E-contract Signing. A 1-out-of-2 ring signature (containing two users in the ring) can be used to construct concurrent signature [16], [47]. A concurrent signature allows two entities to produce two signatures in such a way that, from the point of view of any third party, both signatures are ambiguous with respect to the identity of the signing party until an extra piece of information (the keystone) is released by one of the parties. Upon release of the keystone, both signatures become binding to their true signers concurrently.

Concurrent signature is one of the essential tools for building e-contract signing and fair exchange protocol in the paradigm. It can protect both parties against a cheating party. Consider the following example of fair tendering of contracts. Suppose that A has a building construction contract that she wishes to put out to tender, and suppose companies B and C wish to put in proposals to win the contract. This process is sometimes open to abuse by A since she can privately show B's signed proposal to C to enable C to better the proposal. Using concurrent signatures, B would sign his proposal to construct the building for an amount X, but keep the keystone private. If A wishes to accept the proposal, she returns a payment instruction to pay B amount X. She knows that if B attempts to collect the payment, then A will obtain the keystone through the banking system to allow the public to verify that the signature is really generated by B. But A may also wish to examine C's proposal before deciding which to accept. However there is no advantage for A to show B's signature to C since at this point B's signature is ambiguous and so C will not be convinced of anything at all by seeing it. We see that the tendering process is immune to abuse by A.

Adding forward security to it can further improve the security protection level. With forward security, the key exposure of either party does not affect the e-contracts previously signed. This provides a more fair, justice, safety and efficient environment for commercial users doing business in an e-commerce platform.

E-auction. Similar to e-contract signing, ring signature schemes can be used to construct e-auction protocols [28], [52]. By using ring signature, a winner-identifiable

anonymous auction protocol can be build efficiently. That is to say, the auctioneer can authenticate the real identity of the winner at the end of the protocol without additional interactions with the winning bidder even though all the bidders bid anonymously. Adding forward security further provides additional security to all entities involved in the auction activity. The loss of secret key by anybody does not affect the overall result. It is one of the best way to safeguard the robustness of the e-auction.

5 Conclusion

Motivated by the practical needs in data sharing, we proposed a new notion called forward secure ID-based ring signature. It allows an ID-based ring signature scheme to have forward security. It is the first in the literature to have this feature for ring signature in ID-based setting. Our scheme provides unconditional anonymity and can be proven forward-secure unforgeable in the random oracle model, assuming RSA problem is hard. Our scheme is very efficient and does not require any pairing operations. The size of user secret key is just one integer, while the key update process only requires an exponentiation. We believe our scheme will be very useful in many other practical applications, especially to those require user privacy and authentication, such as ad-hoc network, e-commerce activities and smart grid.

Our current scheme relies on the random oracle assumption to prove its security. We consider a provably secure scheme with the same features in the standard model as an open problem and our future research work.

Acknowledgments

The authors would like to thank Ke Jiang, Zhiliang Peng and Ming Lu, who are postgraduate students with South China University of Technology, for doing the implementations of the proposed scheme. This work was supported by National Natural Science Foundation of China (Grant NOs. 61202450, U1135004 and 61170080), Distinguished Young Scholars Fund of Department of Education, Fujian Province, China (JA13062), Ph.D. Programs Foundation of Ministry of Education of China (Grant NO. 20123503120001), Fujian Normal University Innovative Research Team (NO. IRTL1207), Guangdong Province Universities and Colleges Pearl River Scholar Funded Scheme (2011), High-level Talents Project of Guangdong Institutions of Higher Education (2012), Natural Science Foundation of Fujian Province (No. 2013J01222), and Fok Ying Tung Education Foundation (Grant No. 141065). Joseph K. Liu is the corresponding author.

References

[1] M. Abe, M. Ohkubo, and K. Suzuki, "1-out-of-n signatures from a variety of keys," in *Proc. 8th Int. Conf. Theory Appl. Cryptol. Inform. Security: Adv. Cryptol.*, 2002, vol. 2501, pp. 415–432.

[2] R. Anderson, "Two remarks on public-key cryptology," Manuscript, Sep. 2000. (Relevant material presented by the author in an invited lecture at the Fourth ACM Conference on Computer and Communications Security, 1997.)

[3] G. Ateniese, J. Camenisch, M. Joye, and G. Tsudik, "A practical and provably secure coalition-resistant group signature scheme," in *Proc. 20th Annu. Int. Cryptol. Conf. Adv. Cryptol.*, 2000, vol. 1880, pp. 255–270.

[4] M. H. Au, J. K. Liu, T. H. Yuen, and D. S. Wong, "ID-based ring signature scheme secure in the standard model," in *Proc. 1st Int. Workshop Security Adv. Inform. Comput. Security*, 2006, vol. 4266, pp. 1–16.

[5] A. K. Awasthi and S. Lal, "Id-based ring signature and proxy ring signature schemes from bilinear pairings," *CoRR*, vol. abs/cs/0504097, 2005.

[6] M. Bellare, D. Micciancio, and B. Warinschi, "Foundations of group signatures: Formal definitions, simplified requirements and a construction based on general assumptions," in *Proc. 22nd Int. Conf. Theory Appl. Cryptographic Techn.*, 2003, vol. 2656, pp. 614–629.

[7] M. Bellare and S. Miner, "A forward-secure digital signature scheme," in *Proc. 19th Annu. Int. Cryptol. Conf.*, 1999, vol. 1666, pp. 431–448.

[8] J.-M. Bohli, N. Gruschka, M. Jensen, L. L. Iacono, and N. Marnau, "Security and privacy-enhancing multicloud architectures," *IEEE Trans. Dependable Sec. Comput.*, vol. 10, no. 4, pp. 212–224, Jul. \Aug. 2013.

[9] A. Boldyreva, "Efficient threshold signature, multisignature and blind signature schemes based on the gap Diffie-Hellman group signature scheme," in *Proc. 6th Int. Workshop Theory Practice PublicKey Cryptography: Public Key Cryptography*, 2003, vol. 567, pp. 31–46.

[10] D. Boneh, X. Boyen, and H. Shacham, "Short group signatures," in *Proc. Annu. Int. Cryptol. Conf. Adv. Cryptol.*, 2004, vol. 3152, pp. 41–55.

[11] E. Bresson, J. Stern, and M. Szydlo, "Threshold ring signatures and applications to ad-hoc groups," in *Proc. 22nd Annu. Int. Cryptol. Conf. Adv. Cryptol.*, 2002, vol. 2442, pp. 465–480.

[12] J. Camenisch, "Efficient and generalized group signatures," in *Proc. Int. Conf. Theory Appl. Cryptographic Techn.*, 1997, vol. 1233, pp. 465–479.

[13] N. Chandran, J. Groth, and A. Sahai, "Ring signatures of sublinear size without random oracles," in *Proc. 34th Int. Colloq. Automata, Lang. Programming*, 2007, vol. 4596, pp. 423–434.

[14] K. Chard, K. Bubendorfer, S. Caton, and O. F. Rana, "Social cloud computing: A vision for socially motivated resource sharing," *IEEE Trans. Serv. Comput.*, vol. 5, no. 4, pp. 551–563, Fourth Quarter 2012.

[15] D. Chaum and E. van Heyst, "Group signatures," in *Proc. Workshop Theory Appl. Cryptographic Techn.*, 1991, vol. 547, pp. 257–265.

[16] L. Chen, C. Kudla, and K. G. Paterson, "Concurrent signatures," in *Proc. Int. Conf. Theory Appl. Cryptographic Techn.*, 2004, vol. 3027, pp. 287–305.

[17] H.-Y. Chien, "Highly efficient ID-based ring signature from pairings," in *Proc. IEEE Asia-Pacific Serv. Comput. Conf.*, 2008, pp. 829–834.

[18] S. S. Chow, R. W. Lui, L. C. Hui, and S. Yiu, "Identity based ring signature: Why, how and what next," in *Proc. 2nd Eur. Public Key Infrastructure Workshop*, 2005, vol. 3545, pp. 144–161.

[19] S. S. M. Chow, V.K.-W. Wei, J. K. Liu, and T. H. Yuen, "Ring signatures without random oracles," in *Proc. ACM Symp. Inform., Comput., Commun. Security*, 2006, pp. 297–302.

[20] S. S. M. Chow, S.-M. Yiu, and L. C. K. Hui, "Efficient identity based ring signature," in *Proc. 3rd Int. Conf. Appl. Cryptography Netw. Security*, 2005, vol. 3531, pp. 499–512.

[21] R. Cramer, I. Damgård, and B. Schoenmakers, "Proofs of partial knowledge and simplified design of witness hiding protocols," in *Proc. 14th Annu. Int. Cryptol. Conf. Adv. Cryptol.*, 1994, vol. 839, pp. 174–187.

[22] R. Cramer and V. Shoup, "Signature schemes based on the strong RSA assumption," in *Proc. ACM Conf. Comput. Commun. Security*, 1999, pp. 46–51.

[23] Y. Dodis, A. Kiayias, A. Nicolosi, and V. Shoup, "Anonymous identification in Ad Hoc groups," in *Proc. Int. Conf. Theory Appl. Cryptographic Techn.*, 2004, vol. 3027, pp. 609–626.

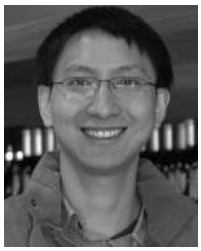
[24] A. L. Ferrara, M. Green, S. Hohenberger, and M. Ø. Pedersen. (2009). Practical short signature batch verification," in *Proc. Cryptographers' Track RSA Conf. Topics Cryptol.*, vol. 5473, pp. 309–324 [Online]. Available: http://eprint.iacr.org/2008/015

[25] J. Han, Q. Xu, and G. Chen, "Efficient ID-based threshold ring signature scheme," in *Proc. IEEE/IFIP Int. Conf. Embedded Ubiquitous Comput.*, 2008, pp. 437–442.

[26] J. Herranz, "Identity-based ring signatures from RSA," *Theor. Comput. Sci.*, vol. 389, no. 1-2, pp. 100–117, 2007.

[27] J. Herranz and G. Sáez, "Forking lemmas for ring signature schemes," in *Proc. 4th Int. Conf. Cryptol. India*, 2003, vol. 2904, pp. 266–279.

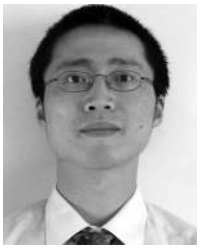
[28] M. Klonowski, L. Krzywiecki, M. Kutylowski, and A. Lauks, "Step-out ring signatures," in *Proc. 33rd Int. Symp. Math. Found. Comput. Sci.*, 2008, vol. 5162, pp. 431–442.
[29] M. Li, S. Yu, Y. Zheng, K. Ren, and W. Lou, "Scalable and secure sharing of personal health records in cloud computing using attribute-based encryption," *IEEE Trans. Parallel Distrib. Syst.*, vol. 24, no. 1, pp. 131–143, Jan. 2013.
[30] D. Y. W. Liu, J. K. Liu, Y. Mu, W. Susilo, and D. S. Wong, "Revocable ring signature," *J. Comput. Sci. Tech.*, vol. 22, no. 6, pp. 785–794, 2007.
[31] J. K. Liu, M. H. Au, W. Susilo, and J. Zhou, "Online/offline ring signature scheme," in *Proc. 11th Int. Conf. Inform. Commun. Security*, 2009, vol. 5927, pp. 80–90.
[32] J. K. Liu, W. Susilo, and D. S. Wong, "Ring signature with designated linkability," in *Proc. 1st Int. Conf. Security*, 2006, vol. 4266, pp. 104–119.
[33] J. K. Liu, V. K. Wei, and D. S. Wong, "A separable threshold ring signature scheme," in *Proc. 6th Int. Conf. Inform. Security Cryptol.*, 2003, vol. 2971, pp. 12–26.
[34] J. K. Liu and D. S. Wong, "On the security models of (Threshold) ring signature schemes," in *Proc. 6th Int. Conf. Inform. Security Cryptol.*, 2004, pp. 12–26.
[35] J. K. Liu and D. S. Wong, "Solutions to key exposure problem in ring signature," *I. J. Netw. Secur.*, vol. 6, no. 2, pp. 170–180, 2008.
[36] J. K. Liu, T. H. Yuen, and J. Zhou, "Forward secure ring signature without random oracles," in *Proc. 13th Int. Conf. Inform. Commun. Security*, 2011, vol. 7043, pp. 1–14.
[37] X. Liu, Y. Zhang, B. Wang, and J. Yan, "Mona: Secure multi-owner data sharing for dynamic groups in the cloud," *IEEE Trans. Parallel Distrib. Syst.*, vol. 24, no. 6, pp. 1182–1191, Jun. 2013.
[38] C. A. Melchor, P.-L. Cayrel, P. Gaborit, and F. Laguillaumie, "A new efficient threshold ring signature scheme based on coding theory," *IEEE Trans. Inform. Theory*, vol. 57, no. 7, pp. 4833–4842, Jul. 2011.
[39] Microsoft. (2009). Conserve energy, save money-Microsoft Hohm. [Online]. Available: http://www.microsoft-hohm.com/
[40] *NIST IR 7628: Guidelines for Smart Grid Cyber Security*, NIST IR 7628: Guidelines for Smart Grid Cyber Security, Aug. 2010.
[41] L. Nguyen, "Accumulators from bilinear pairings and applications," in *Proc. Int. Conf. Topics Cryptol.*, 2005, vol. 3376, pp. 275–292.
[42] R. L. Rivest, A. Shamir, and Y. Tauman, "How to leak a secret," in *Proc. 7th Int. Conf. Theory Appl. Cryptol. Inform. Security: Adv. Cryptol.*, 2001, vol. 2248, pp. 552–565.
[43] M. Scott. (2013). MIRACL-A multiprecision integer and rational arithmetic C/C++ library, Shamus Software Ltd., [Online]. Available: http://www.shamus.ie/index.php
[44] H. Shacham and B. Waters, "Efficient ring signatures without random oracles," in *Proc. 10th Int. Conf. Practice Theory Public Key Cryptography*, 2007, vol. 4450, pp. 166–180.
[45] A. Shamir, "Identity-based cryptosystems and signature schemes," in *Proc. CRYPTO 84 Adv. Cryptol.*, 1984, vol. 196, pp. 47–53.
[46] S. Sundareswaran, A. C. Squicciarini, and D. Lin, "Ensuring distributed accountability for data sharing in the cloud," *IEEE Trans. Dependable Secure Comput.*, vol. 9, no. 4, pp. 556–568, Jul./Aug. 2012.
[47] W. Susilo, Y. Mu, and F. Zhang, "Perfect concurrent signature schemes," in *Proc. 6th Int. Conf. Inform. Commun. Security*, Oct. 2004, vol. 3269, pp. 14–26.
[48] P. P. Tsang, M. H. Au, J. K. Liu, W. Susilo, and D. S. Wong, "A suite of non-pairing ID-based threshold ring signature schemes with different levels of anonymity (extended abstract)," in *Proc. 4th Int. Conf. Provable Security*, 2010, vol. 6402, pp. 166–183.
[49] C. Wang, S. S. M. Chow, Q. Wang, K. Ren, and W. Lou, "Privacy-preserving public auditing for secure cloud storage," *IEEE Trans. Comput.*, vol. 62, no. 2, pp. 362–375, Feb. 2013.
[50] D. S. Wong, K. Fung, J. K. Liu, and V. K. Wei, "On the RS-Code construction of ring signature schemes and a threshold setting of RST," in *Proc. 5th Int. Conf. Inform. Commun. Security*, 2003, vol. 2836, pp. 34–46.
[51] Y. Wu, Z. Wei, and R. H. Deng, "Attribute-based access to scalable media in cloud-assisted content sharing networks," *IEEE Trans. Multimedia*, vol. 15, no. 4, pp. 778–788, Jun. 2013.
[52] H. Xiong, Z. Qin, and F. Li, "An anonymous sealed-bid electronic auction based on ring signature," *I. J. Netw. Secur.*, vol. 8, no. 3, pp. 235–242, 2009.
[53] G. Yan, D. Wen, S. Olariu, and M. Weigle, "Security challenges in vehicular cloud computing," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 1, pp. 284–294, Mar. 2013.
[54] J. Yu, R. Hao, F. Kong, X. Cheng, J. Fan, and Y. Chen, "Forward-secure identity-based signature: Security notions and construction," *Inform. Sci.*, vol. 181, no. 3, pp. 648–660, 2011.
[55] J. Yu, F. Kong, H. Zhao, X. Cheng, R. Hao, and X.-F. Guo, "Non-interactive forward-secure threshold signature without random oracles," *J. Inform. Sci. Eng.*, vol. 28, no. 3, pp. 571–586, 2012.
[56] T. H. Yuen, J. K. Liu, X. Huang, M. H. Au, W. Susilo, and J. Zhou, "Forward secure attribute-based signatures," in *Proc. 14th Int. Conf. Inform. Commun. Security*, 2012, vol. 7618, pp. 167–177.
[57] F. Zhang and K. Kim, "ID-based blind signature and ring signature from pairings," in *Proc. 8th Int. Conf. Theory Appl. Cryptol. Inform. Security*, 2002, vol. 2501, pp. 533–547.
[58] J. Zhang, "An efficient identity-based ring signature scheme and its extension," in *Proc. Int. Conf. Comput. Sci. Appl.*, 2007, vol. 4706, pp. 63–74.



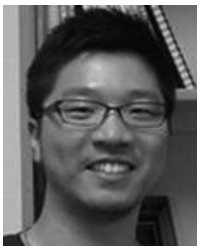
Xinyi Huang received the PhD degree from the School of Computer Science and Software Engineering, University of Wollongong, Australia, in 2009. He is currently a professor at the Fujian Provincial Key Laboratory of Network Security and Cryptology, School of Mathematics and Computer Science, Fujian Normal University, China. His research interests include cryptography and information security. He has published more than 70 research papers in refereed international conferences and journals. His work has been cited more than 1,000 times at Google Scholar. He is in the editorial board of *International Journal of Information Security* (IJIS, Springer) and has served as the program/general chair or program committee member in more than 40 international conferences.

Joseph K. Liu received the PhD degree in information engineering from the Chinese University of Hong Kong in July 2004, speciallizing in cryptographic protocols for securing wireless networks, privacy, authentication and provable security. He is currently a research scientist in the Infocomm Security Department at the Institute for Infocomm Research, Singapore. His current technical focus is particularly lightweight cryptography, wireless security, security in smart grid system and cloud computing environment.

Shaohua Tang received the BSc and MSc degrees in applied mathematics, and the PhD degree in communication and information system from the South China University of Technology, in 1991, 1994, and 1998, respectively. He was a visiting scholar with North Carolina State University, USA, and a visiting professor with University of Cincinnati, USA. He has been a full professor with the School of Computer Science and Engineering, South China University of Technology since 2004. His current research interests include information security, networking, and information processing. He is a member of the IEEE and the IEEE Computer Society.



Yang Xiang received the PhD degree in computer science from Deakin University, Australia. He is currently a full professor at the School of Information Technology, Deakin University. He is the director of the Network Security and Computing Lab (NSCLab). His research interests include network and system security, distributed systems, and networking. In particular, he is currently leading his team developing active defense systems against large-scale distributed network attacks. He is the chief investigator of several projects in network and system security, funded by the Australian Research Council (ARC). He has published more than 130 research papers in many international journals and conferences, such as *IEEE Transactions on Computers, IEEE Transactions on Parallel and Distributed Systems, IEEE Transactions on Information Security and Forensics*, and *IEEE Journal on Selected Areas in Communications*. Two of his papers were selected as the featured articles in the April 2009 and the July 2013 issues of *IEEE Transactions on Parallel and Distributed Systems*. He has published two books, *Software Similarity and Classification* (Springer) and *Dynamic and Advanced Data Mining for Progressing Technological Development* (IGI-Global). He has served as the program/general chair for many international conferences such as ICA3PP 12/11, IEEE/IFIP EUC 11, IEEE TrustCom 13/11, IEEE HPCC 10/09, IEEE ICPADS 08, NSS 11/10/09/08/07. He has been the PC member for more than 60 international conferences in distributed systems, networking, and security. He serves as the associate editor of *IEEE Transactions on Computers, IEEE Transactions on Parallel and Distributed Systems, Security and Communication Networks* (Wiley), and the editor of *Journal of Network and Computer Applications*. He is the coordinator, Asia for IEEE Computer Society Technical Committee on Distributed Processing (TCDP). He is a senior member of the IEEE.



Kaitai Liang received the BEng degree of software engineering, and the MSc degree of computer applied technology from South China Agricultural University, China in 2008 and 2011, respectively. He is currently working toward the PhD degree in the Department of Computer Science at the City University of Hong Kong. His primary research interest is applied cryptography; in particular, cryptographic protocols, encryption/signature, RFID, and cloud security. He is also interested in other topics in information security, such as network security, wireless security database security, and security in cloud computing.

Li Xu received the BS and MS degrees from Fujian Normal University, in 1992 and 2001, and the PhD degree from the Nanjing University of Posts and Telecommunications in 2004. He is a professor and a doctoral supervisor at the School of Mathematics and Computer Science at Fujian Normal University. He is currently the vice dean of the School of Mathematics and Computer Science and the director of the Key Lab of Network Security and Cryptography in Fujian Province. His interests include wireless networks and communication, network and information security, complex networks and systems, intelligent information in communication networks, etc. He has been invited to act as a PC chair or member at more than 30 international conferences. He has published over 100 papers in refereed journals and conferences. He is a member of the IEEE and ACM, and a senior member of CCF and CIE in China.

Jianying Zhou received the PhD degree in information security from the University of London in 1997. He is currently a senior scientist at the Institute for Infocomm Research and the head of Infocomm Security Department. His research interests include computer and network security, mobile and wireless security. He is a co-founder and steering committee member of International Conference on Applied Cryptography and Network Security (ACNS).