FORWARD-SECURE IDENTITY-BASED ENCRYPTION WITH
DIRECT CHOSEN-CIPHERTEXT SECURITY IN THE
STANDARD MODEL

YANG LU AND JIGUO LI
College of Computer and Information, Hohai University
No.8, Focheng Xi Road, Jiangning District
Nanjing, Jiangsu 211100, China

(Communicated by Jens Zumbraegel)

Abstract. The paradigm of forward security provides a promising approach to
deal with the key exposure problem as it can effectively minimize the damage
caused by the key exposure. In this paper, we develop a new forward-secure
identity-based encryption scheme without random oracles. We formally prove
that the proposed scheme is secure against adaptive chosen-ciphertext attacks
in the standard model. In the proposed scheme, the running time of the private
key extraction and decryption algorithms and the sizes of the user’s initial pri-
vate key and the ciphertext are independent on the total number of time peri-
ods, and any other performance parameter has at most log-squared complexity
in terms of the total number of time periods. Compared with the previous
forward-secure identity-based encryption schemes, the proposed scheme enjoys
obvious advantage in the overall performance. To the best of our knowledge,
it is the first forward-secure identity-based encryption scheme that achieves
direct chosen-ciphertext security in the standard model.

1. Introduction

In public key cryptography, each user has a pair of keys, namely a public key
and a private key. The public key is usually published to the public while the
corresponding private key is only known to its owner. However, in traditional public
key cryptography, the public key is generated randomly and does not contain any
information associated with its owner. Therefore, it is infeasible to prove that a user
is indeed the owner of a given public key. This problem can be solved by employing
a trusted certification authority (CA) to generate public key certificates. A public
key certificate is a digital signature issued by CA that binds a public key to the
identity of its owner. By verifying the public key certificate, anyone can confirm
whether a public key belongs to a user. This kind of certificate systems is referred
to as the public key infrastructure (PKI). However, the need for PKI-certificates is

2010 Mathematics Subject Classification: Primary: 94A60; Secondary: 11T71.

Key words and phrases: Identity-based encryption, key exposure, forward security, chosen-
ciphertext security, standard model.

This work is supported by the Nature Science Foundation of China under Grant Nos. 61272542
and 61672207, the Natural Science Foundation of Jiangsu Province Grant No. BK20161511,
the Fundamental Research Funds for the Central Universities Grant No. 2016B10114, a Project
Funded by the Priority Academic Program Development of Jiangsu Higher Education Institu-
tions and Jiangsu Collaborative Innovation Center on Atmospheric Environment and Equipment
Technology.
usually considered as the main difficulty in the deployment of traditional public key cryptography.

In 1984, Shamir \[37\] introduced the concept of identity-based cryptography to simplify public key certificate management. In identity-based cryptography, a user’s public key is his unique identity information such as an e-mail address or a telephone number, and his private key is generated by a trusted third party called Private Key Generator (PKG). Because the identity is a natural link to a user, the ability to use identities as public keys eliminates the need for public key certificates and certificate authorities. However, how to construct a secure and practical identity-based encryption scheme remained open for many years since the question was first posed by Shamir in \[37\]. Until 2001, Boneh and Franklin \[10\] presented a practical and provably secure identity-based encryption scheme using the bilinear pairings. Since then, identity-based encryption has undergone quite rapid development and many schemes have been proposed, e.g. \[8, 9, 15, 16, 20, 21, 24, 27, 41\].

The standard security of identity-based cryptography depends on the assumption that the private keys are kept perfectly secure. However, as cryptographic operations are performed frequently on the unprotected and insecure devices, private key exposure seems to be inevitable. Actually, it is much easier for an adversary to steal a user’s private key from his insecure device than to break the computational assumption(s) on which an identity-based cryptosystem is based. Undoubtedly, private key exposure has become one of the most devastating attacks on the identity-based cryptosystems, as it means all security guarantees are lost. To mitigate the damage caused by private key exposure, one effective way is to build forward-secure identity-based cryptosystems. Forward-secure identity-based cryptosystems enable a user to frequently update his private key while maintaining a fixed public key (i.e., his identity). More concretely, in a forward-secure identity-based cryptosystem the whole lifetime of the system (e.g., one year) is divided into \(T\) time periods (e.g., one day). A user’s device begins by storing an initial private key. At each time period, a new private key is computed from the former private key by a key update algorithm and then the old key is deleted. Meanwhile, this user’s public key (i.e., his identity) remains fixed throughout the lifetime of the system. The forward security property means that even if an adversary obtains the current private key, he still cannot compromise private keys and communications for the past time. Therefore, the paradigm of forward security provides a promising approach to deal with the private key exposure problem in the identity-based cryptosystems.

The notion of forward security was first proposed in the context of key-exchange protocols by Günther \[22\] and later by Diffie \textit{et al.} \[17\]. Subsequently, Anderson \[3\] suggested forward security for the non-interactive setting and proposed a general forward-secure signature scheme. The non-interactive forward security was first formalized in the context of signature by Bellare and Miner \[4\], in which the first practical forward-secure signature scheme was proposed. Inspired by the initial works in \[3, 4\], a number of forward-secure signature schemes have been proposed, e.g. \[1, 2, 11, 25, 29, 30, 31, 38, 49\]. Moreover, Bellare and Yee \[7\] proposed a forward-secure scheme in the symmetric-key encryption setting. The first non-interactive and forward-secure public key encryption scheme was proposed by Canetti \textit{et al.} \[13\] in 2003. Canetti \textit{et al.}'s scheme was constructed from the hierarchical identity-based encryption scheme proposed by Gentry and Silverberg \[21\] and proven to be chosen-plaintext secure in the standard model. Based on the hierarchical identity-based encryption scheme proposed by Boneh \textit{et al.} \[9\], Lu and Li \[32\] proposed an
efficient forward-secure public key encryption scheme with chosen-plaintext security in the standard model. Recently, Lu and Li [34] proposed an efficient forward-secure public key encryption scheme that achieves chosen-ciphertext security in the standard model.

The first forward-secure encryption scheme in the identity-based setting was proposed by Yao et al. [45]. In [45], Yao et al. constructed a forward-secure hierarchical identity-based encryption scheme by combining Canetti et al.'s forward-secure public key encryption scheme [13] with Gentry and Silverberg’s hierarchical identity-based encryption scheme [21]. The biggest drawback of Yao et al.’s scheme lies in that any performance parameter has the poly-logarithmic complexity in terms of the total number of time periods. Therefore, Yao et al.’s scheme is inefficient for large values of the total number of the time periods. In addition, the security of Yao et al.’s scheme only holds in the random oracle model [6]. In [44], Yang et al. proposed a forward-secure public key encryption scheme with short public parameters. But, their scheme only satisfies the weaker chosen-plaintext security in the random oracle model and any other performance parameter in their scheme has the poly-logarithmic complexity in terms of the total number of time periods. In [47], Yu et al. proposed the first forward-secure identity-based encryption scheme without random oracles. Compared with the schemes in [44, 45], Yu et al.’s scheme has shorter ciphertext and lower decryption cost as its ciphertext size and decryption time are both independent on the total number of time periods. However, Yu et al.’s scheme was only proven to be chosen-plaintext secure in the standard model. Recently, Lu and Li [33] proposed a generic construction of forward-secure identity-based encryption from binary tree encryption. In addition, Singh and Trichy [38] proposed a forward-secure identity-based encryption scheme from lattices which achieves chosen-plaintext security in the standard model. Singh and Trichy’s scheme is quite inefficient because it encrypts the messages bit-by-bit and has huge ciphertext extension.

Motivated by the works on forward security, the notion of key insulation [5, 19, 23] was introduced into the identity-based setting as a mean of mitigating the harmful effects caused by private key exposure, e.g. [20, 35, 39, 40, 42, 43]. Similar to the forward-secure paradigm, the key-insulated paradigm also applies some key-evolving approach to update the users’ private keys so as to limit the effect of the private key exposure. Interestingly, this paradigm achieves a stronger security level compared with forward security, as it guarantees that the exposure of the private key at some periods does not compromise the security of all non-exposed time periods. However, the drawback of the key-insulated paradigm is that each user in a key-insulated cryptosystem must use a physically-secure device named helper to update his private key in each time period. Obviously, it makes key-insulated cryptosystems unable to be used in many circumstances.

Recently, the notion of intrusion-resilience [18, 26] was also introduced into the identity-based setting to fight against private key exposure. The intrusion-resilient paradigm combines both the notions of forward security and key insulation. As in the key-insulated cryptosystems, the intrusion-resilient paradigm assumes that a user performs all cryptographic operations and interacts with a helper to refresh his private keys at discrete time intervals. The difference is that all the users and the helper should update their private keys periodically as in the forward-secure cryptosystems. Thus, an intrusion-resilient cryptosystem remains secure in the face of multiple compromises of both the user and the helper, as long as they are not
both compromised simultaneously. Furthermore, in case the user and the helper are compromised simultaneously, prior time periods remain secure as in the forward-secure cryptosystems.

1.1. Our motivation and contribution. Indistinguishable security against adaptive chosen-ciphertext attacks (i.e., chosen-ciphertext security) is the de facto level of security required for the public key cryptographic schemes used in practice. However, the previous forward-secure identity-based encryption schemes without random oracles [38, 47] merely satisfy the weaker chosen-plaintext security. Although Yao et al.’s scheme [45] achieves chosen-ciphertext security, it’s security only holds in the random oracle model. A proof in the random oracle model may not necessarily imply the security in the reality [12]. As shown by Canetti et al. in [12], when the random oracles are instantiated with the concrete cryptographic hash functions, the resulting cryptographic schemes may not be practically secure. Therefore, the search for forward-secure identity-based encryption schemes that can achieve chosen-ciphertext security without resorting to the random oracles is of great interesting and importance.

In this paper, we design a new forward-secure identity-based encryption scheme without random oracles. The construction of the proposed scheme is based on Gentry’s identity-based encryption scheme [20] and its variant [27]. We formally prove in the standard model that the proposed scheme is chosen-ciphertext secure under the truncated decision \( q \)-augmented bilinear Diffie-Hellman exponent assumption. In the proposed scheme, the running time of key extraction and decryption algorithms and the sizes of the user’s initial private key and the ciphertext are independent on the total number of time periods, and any other performance parameter has at most log-squared complexity in terms of the total number of time periods. Compared with the previous forward-secure identity-based encryption schemes, our scheme enjoys obvious advantage in the overall performance while achieving direct chosen-ciphertext security in the standard model.

1.2. Paper organization. The rest of this paper is organized as follows. Some related definitions are briefly reviewed in Section 2. The proposed forward-secure identity-based encryption scheme is presented in Section 3. The security and performance of the proposed scheme are analyzed in Section 4 and 5 respectively. Finally, conclusions are given in Section 6.

2. Preliminaries

In this section, we briefly review some preliminaries that are related to our paper.

2.1. Bilinear map and computational assumption. Let \( k \) be a security parameter and \( p \) be a \( k \)-bit prime number, \( G \) and \( G_T \) denote two multiplicative cyclic groups of the same order \( p \) respectively. A mapping \( e: G \times G \to G_T \) is called an admissible bilinear map if it satisfies the following three properties:

- Bilinearity: \( e(u^a, v^b) = e(u, v)^{ab} \) for all \( u, v \in G \) and \( a, b \in \mathbb{Z}_p^* \).
- Non-degeneracy: \( e(g, g) \neq 1_{G_T} \) for a random generator \( g \in G \).
- Computability: \( e(u, v) \) can be efficiently computed for all \( u, v \in G \).

The security of our forward-secure identity-based encryption scheme is based on the following truncated decision \( q \)-augmented bilinear Diffie-Hellman exponent (\( q \)-ABDHE) assumption proposed by Gentry in [11].
The truncated decision $q$-ABDHE problem in $(\mathbf{G}, \mathbf{G}_T)$ is as follows: Given a vector of $q + 3$ elements $(g', g^{a^{q+2}}, g, g^a, \ldots, g^{a^3}) \in \mathbf{G}^{q+3}$ and an element $Z \in \mathbf{G}_T$, output 1 if $Z = e(g, g')^{a^{q+3}}$ and 0 otherwise.

Let $\mathcal{B}$ be a probabilistic polynomial-time (PPT) algorithm that takes as input a random truncated decision $q$-ABDHE problem instance and outputs a bit $b \in \{0, 1\}$. The algorithm $\mathcal{B}$ has advantage $\epsilon$ in solving the truncated decision $q$-ABDHE problem if

$$\Pr\{\mathcal{B}(\mathbf{G}, \mathbf{G}_T, p, g', g^{a^{q+2}}, g, g^a, \ldots, g^{a^3}, e(g, g')^{a^{q+1}}) = 1\} - \Pr\{\mathcal{B}(\mathbf{G}, \mathbf{G}_T, p, g', g^{a^{q+2}}, g, g^a, \ldots, g^{a^3}, Z) = 1\} \geq \epsilon,$$

where the probability is over the random choice of generators $g, g' \in \mathbf{G}$, the random choice of the element $Z \in \mathbf{G}_T$, the random choice of the value $a \in \mathbb{Z}_{p}^*$, and the random bits consumed by the algorithm $\mathcal{B}$.

**Definition 2.1.** We say that the truncated decision $(t, \epsilon, q)$-ABDHE assumption holds in $(\mathbf{G}, \mathbf{G}_T)$ if no $t$-time algorithm has advantage at least $\epsilon$ in solving the truncated decision $q$-ABDHE problem in $(\mathbf{G}, \mathbf{G}_T)$.

### 2.2. Forward-secure identity-based encryption

Formally, a forward-secure identity-based encryption scheme is specified by a 5-tuple of algorithms $(\text{Setup}, \text{KeyExtract}, \text{KeyUpdate}, \text{Encrypt}, \text{Decrypt})$ such that:

1. **Setup** is the system setup algorithm that takes a security parameter $k$ and the total number of time periods $T$ as input, and outputs a master secret key $msk$ and a list of public parameters $\text{params}$. Usually, this algorithm is performed by a PKG. After the algorithm is performed, the PKG keeps the master secret key $msk$ secret and publishes the public parameters $\text{params}$ to the public.

2. **KeyExtract** is the key extraction algorithm that takes $\text{params}$, $msk$ and an identity $ID$ as input, and outputs an initial private key $SK_{ID}^1$ for the user with identity $ID$. This algorithm is also performed by a PKG. After the algorithm is performed, the PKG sends $SK_{ID}^1$ to the user with identity $ID$ via a secure channel.

3. **KeyUpdate** is the key update algorithm that takes $\text{params}$, the index $i \in [0, N]$ of the current time period, a user’s identity $ID$ and private key $SK_{ID}^{i-1}$ in the time period $i - 1$ as input, and outputs a private key $SK_{ID}^{i}$ in the time period $i$ for that user. Specially, when $i = 0$, $SK_{ID}^{i-1} = SK_{ID}^{i}$. This algorithm is performed by each user. After the algorithm is performed, the former private key $SK_{ID}^{i-1}$ is deleted.

4. **Encrypt** is the encryption algorithm that takes $\text{params}$, the index $i \in [0, N]$ of the current time period, the receiver’s identity $ID$ and a message $M$ as input, and outputs a ciphertext $C$ in the time period $i$. For simplicity, we represent a ciphertext $C$ in the time period $i$ as a pair $(i, C)$ and write $(i, C) \leftarrow \text{Encrypt}(\text{params}, i, ID, M)$.

5. **Decrypt** is the decryption algorithm that takes $\text{params}$, the receiver’s current private key $SK_{ID}^{i}$ and a ciphertext $(i, C)$ as input, and outputs a message $M$ or a special symbol $\bot$ if the ciphertext is invalid.

For correctness, it is required that, for any message $M$, if $(i, C) \leftarrow \text{Encrypt}(\text{params}, i, ID, M)$, then $M = \text{Decrypt}(\text{params}, SK_{ID}^{i}, (i, C))$.

As introduced in [45], the chosen-ciphertext security for forward-secure identity-based encryption schemes $(fs-ID-CCA2)$ is defined via an adversarial game, in which a game challenger/simulator $\mathcal{S}$ interacts with an adversary $\mathcal{A}$ in the following way:
A forward-secure identity-based encryption scheme is said to be
secure identity-based encryption schemes if the adversary is disallowed to make any

decryption queries in the above game. The concrete definition can be found in [47].

An identity
ID∗, a time period
i∗ and two length-equal plaintexts
M0 and
M1 on which it wishes to be challenged. The constraint is that no key extraction query has
been issued on (ID∗, j), where
j ∈ \{0, i∗\}. The challenger
S chooses a random bit
b ∈ \{0, 1\}, computes (i∗, C∗) ← Encrypt(params, i∗, ID∗, M0), and then outputs
⟨i∗, C∗⟩ as the challenge ciphertext to the adversary
A.

Phase 2. In this phase, the adversary
A issues more key extraction queries and decryption queries. The constraint is that: (1) no key extraction query can be issued on (ID∗, j), where
j ∈ \{0, i∗\}; (2) no decryption query can be issued on (ID∗, ⟨i∗, C∗⟩). The challenger
S responds the adversary
A’s queries as in Phase 1.

Guess. Finally, the adversary
A outputs a guess
b′ ∈ \{0, 1\} for the bit
b and wins the game if
b = b′. The adversary
A’s advantage in the above game is defined to be
AdvA(k) = |Pr[b = b′] − 1/2|.

We call an adversary in the above game an fs-ID-CCA2 adversary.

Definition 2.2. A forward-secure identity-based encryption scheme is said to be
(t, qE, qD, ε) - fs-ID-CCA2 secure if for any t-time fs-ID-CCA2 adversary that makes
at most
qE key extraction queries and
qD decryption queries has advantage at most
ε in the above game.

Similarly, the chosen-plaintext security (fs-ID-CPA) can be defined for forward-
secure identity-based encryption schemes if the adversary is disallowed to make any
decryption queries in the above game. The concrete definition can be found in [47].

3. The proposed scheme

In this section, we propose a new forward-secure identity-based encryption scheme
without random oracles. We first give an overview of the key-evolving mechanism
in our scheme. Then, we propose the detailed construction.

3.1. Overview. Like the previous forward-secure identity-based encryption
schemes [13, 45, 47], we use the full binary tree structure to update the users’
private keys. In [13, 45, 47], all the time periods are associated with the leaf nodes
of a full binary tree. Thus, to produce a forward-secure identity-based encryption
scheme with
T time periods, a full binary tree with level log2 T should be used.

Inspired by the key-evolving mechanism proposed by Canetti et al. in [13], we as-
sociate the time periods with all non-root nodes of the binary tree rather than with
the leaf nodes only. Note that, in Canetti et al.’s original proposal \cite{canetti2003}, the time periods are associated with all nodes of the binary tree. Such minor modification in our construction enables us to generate a short initial private key with constant length for each user, which consists of only four elements. We note that the user’s initial private key consists of $O(\log T)$ elements in \cite{canetti2003} and $O(\log^2 T)$ elements in \cite{canetti2005}. Therefore, our scheme needs a lower communication bandwidth to send the initial private key from the PKG to each user. Moreover, to produce a cryptosystem with same $T$ time periods, our scheme only needs a full binary tree with level $l = \log_2 (T + 2) - 1$ which is less than the level $\log_2 T$ in \cite{canetti2003, canetti2005, canetti2005}.

We label each node of the binary tree with a binary string. The root node is labelled with an empty string $\varepsilon$ and if an internal node is labelled with a binary string $\omega$, then its left child and right child are labelled with $\omega 0$ and $\omega 1$ respectively. Let $\omega^i$ denote the node associated with the time period $i$. We associate the time periods with all non-root nodes of a full binary tree in a pre-order style as follows:

- $\omega^0 = 0$ is the left child of the root node;
- If $\omega^i$ is an internal node, then $\omega^{i+1} = \omega^i 0$;
- If $\omega^i$ is a leaf node, then $\omega^{i+1} = \omega^i 1$, where $\omega^i$ is a binary string such that $\omega^0$ is equal to $\omega^i$ or the longest prefix of $\omega^i$.

FIGURE 1 gives a concrete example of how to associate the time periods \{0, 1, \ldots, 13\} with the nodes in a full binary tree with level 3.

![Figure 1](image.png)

**Figure 1.** An example of how to associate the time periods \{0, 1, \ldots, 13\} with the nodes in a full binary tree with level 3

In the binary tree, each node has a secret key. We denote the secret key of a node labelled with $\omega$ by $sk_\omega$. In our construction, the secret key of a node at level $d$ is a vector $(a_0, a_1, a_2, a_3, a_4, b_{l+1}, \ldots, b_l)$ of $5 + l - d$ elements and the private key of a user in a time period $i$ is a set of node secret keys. If let $\omega^i$ denote the node associated with the time period $i$, then a user’s private key in the time period $i$ is composed of the secret key of the node $\omega^i$ and all secret keys of the right siblings of the nodes on the path from the root to the node $\omega^i$. FIGURE 2 gives a concrete example to show which node secret keys are included in the private key of a user with identity $ID$ in each time period $i(0 \leq i \leq 13)$. 

Advances in Mathematics of Communications  Volume 11, No. 1 (2017), 161–177
The private key of a user with identity ID in each time period $i$

\[
\begin{align*}
SK_{ID}^0 &= \{sk_0, sk_1\} \\
SK_{ID}^1 &= \{sk_{00}, sk_{01}, sk_1\} \\
SK_{ID}^2 &= \{sk_{000}, sk_{001}, sk_{01}, sk_1\} \\
SK_{ID}^3 &= \{sk_{00,0}, sk_{01}, sk_1\} \\
SK_{ID}^4 &= \{sk_{01}, sk_1\} \\
SK_{ID}^5 &= \{sk_{010}, sk_{011}, sk_1\} \\
SK_{ID}^6 &= \{sk_{011}, sk_1\}
\end{align*}
\]

Figure 2. An example to show which node secret keys are included in the private key of a user with identity ID in each time period $i (0 \leq i \leq 13)$

For simplicity of description, we represent a private key of a user as a stack of node keys. The secret key of the node $\omega^i$ is on top of the stack and the following are the secret keys of the right siblings of the nodes on the path from the node $\omega^i$ to the root in the binary tree.

3.2. Description of the Proposed Scheme. The construction of our scheme is based on Gentry’s identity-based encryption scheme [20] and its variant [27]. We assume that the identities in our scheme are elements of $\mathbb{Z}_p^*$. Of course, we can extend our scheme to identities over $\{0, 1\}^*$ by first hashing them into the elements of $\mathbb{Z}_p^*$ using a collision-resistant hash function. The details of our scheme is described as follows:

(1) Setup($k, T$): This algorithm performs as follows: Generate two cyclic groups $G$ and $G_T$ of some prime order $p$ and an admissible bilinear map $e : G \times G \rightarrow G_T$; Randomly choose three generators $g, h_1, h_2 \in G$ and a random value $\alpha \in \mathbb{Z}_p^*$, compute $g_1 = g^\alpha$; Randomly choose two $l$-length vectors $\overrightarrow{U} = (u_1, u_2, \ldots, u_l) \in G^l$, $\overrightarrow{V} = (v_1, v_2, \ldots, v_l) \in G^l$; Choose a collision-resistant hash function $H : G \times G_T \times G \rightarrow \mathbb{Z}_p^*$; Set the public system parameters $\text{params} = \{T, p, G, G_T, e, g, g_1, h_1, h_2, \overrightarrow{U}, \overrightarrow{V}, H\}$ and the master secret key $\text{msk} = \alpha$.

(2) KeyExtract($\text{params}, \text{msk}, ID$): This algorithm randomly chooses two random values $\beta_1, \beta_2 \in \mathbb{Z}_p^*$ and generates an initial private key for a user with identity $ID$ as

\[
SK_{ID}^\phi = (\beta_1, \beta_2, (g^{-\beta_1}h_1)^\frac{-1}{\overrightarrow{U}}, (g^{-\beta_2}h_2)^\frac{-1}{\overrightarrow{V}}).
\]

(3) KeyUpd($\text{params}, i, ID, SK_{ID}^{i-1}$): This algorithm performs as follows:

If $i = 0$, choose a random value $r \in \mathbb{Z}_p^*$, generate the node keys $sk_0$ and $sk_1$ from the user’s initial private key $SK_{ID}^\phi$ as

\[
\begin{align*}
sk_0 &= (a_0, a_1, a_2, a_3, a_4, b_2, \ldots, b_l) \\
&= (\beta_1, \beta_2, (g^{-\beta_1}h_1)^\frac{-1}{\overrightarrow{U}}, (g^{-\beta_2}h_2)^\frac{-1}{\overrightarrow{V}}, (g^{-ID}g_1)^{-r}, z_2, \ldots, z_l),
\end{align*}
\]

\[
\begin{align*}
sk_1 &= (a_0', a_1', a_2', a_3', a_4', b_2', \ldots, b_l') \\
&= (\beta_1, \beta_2, (g^{-\beta_1}h_1)^\frac{-1}{\overrightarrow{U}} \cdot z_1^r, (g^{-\beta_2}h_2)^\frac{-1}{\overrightarrow{V}} \cdot z_1^r, (g^{-ID}g_1)^{-r}, z_2, \ldots, z_l'),
\end{align*}
\]
where \( z_j = u_j^{-ID} v_j \) for \( j = 1, 2, \ldots, l \). Push \( sk_1 \) and then \( sk_0 \) onto the stack. Then, the node keys in the stack compose the user’s private key \( SK_{ID}^0 \) in the time period 0, namely that \( SK_{ID}^0 = (sk_0, sk_1) \).

Otherwise, let \( \omega^{i-1} = \omega_1 \omega_2 \cdots \omega_i \in \{0, 1\}^{d \leq t} \) be the node associated with the time period \( i - 1 \) and \( sk_{\omega^{i-1}} \) be the node key associated with the node \( \omega^{i-1} \). Pop \( sk_{\omega^{i-1}} \) off the stack, in which the user’s private key \( SK_{ID}^{i-1} \) is stored.

- If \( \omega^{i-1} \) is a leaf node, set the remaining node keys in the stack as the user’s private key \( ID \)’s private key \( SK_{ID}^i \) in the time period \( i \). It is easy to see that the node key on the top of the stack is \( sk_i \) and then sets

\[
\langle \omega_{1}, c \rangle = \left( \omega_{1}, c \right) \rho \cdot \left( \prod_{j=1}^{d} z_{j}^{\omega_{j}} \right)^{r} \cdot \left( \prod_{j=1}^{d} h_{2}^{\omega_{j}} \right)^{\frac{1}{p - \alpha}}.
\]

and respectively compute two node keys

\[
sk_{\omega_{1}, \omega_{d+1}} = \left( a_{0}, a_{1}, a_{2}, a_{3}, a_{4}, b_{d+1}, \ldots, b_{l} \right)
\]

and

\[
\omega^{d+1} \in \{0, 1\} \text{ and } r' = r + s. \text{ Push } sk_{\omega_{1}, \omega_{d+1}} \text{ and then } sk_{\omega_{1}, \omega_{d+1}} \text{ onto the stack. Now, the node keys in the stack compose the user’s private key } SK_{ID}^i \text{ in the time period } i.
\]

(4) Encrypt((params, i, ID, M)): Let \( \omega^{i} = \omega_{1} \omega_{2} \cdots \omega_{d} \in \{0, 1\}^{d \leq t} \) be the node associated with the time period \( i \). This algorithm chooses a random value \( t \in Z_{p}^{*} \), computes

\[
C = (c_1, c_2, c_3, c_4) = \left( \left( g^{-ID} g_i \right)^{t}, e(g, g)^{t}, \left( \prod_{j=1}^{d} z_{j}^{\omega_{j}} \right)^{t}, M \cdot \left( e(g, h_1)^{t} \cdot e(g, h_2)^{\gamma} \right)^{-t} \right),
\]

and then sets \( (i, C) \) as the ciphertext, where \( z_j = u_j^{-ID} v_j \) and \( \gamma = H(c_1, c_2, c_3) \).

(5) Decrypt((params, SK_{ID}^i, (i, C))): Let \( \omega^{i} = \omega_{1} \omega_{2} \cdots \omega_{d} \in \{0, 1\}^{d \leq t} \) be the node associated with the time period \( i \) and \( sk_{\omega^{i}} \) be the node key associated with the
node $\omega^i$. This algorithm parses $sk_{\omega^i} = (a_0, a_1, a_2, a_3, a_4, b_{d+1}, \ldots, b_1)$ and decrypt the ciphertext to get the message

$$M = \frac{e(c_1, a_2^3 a_3) \cdot c_2^q a_0 + a_1 \cdot c_4}{e(a_4, c_3)^{\gamma + 1}},$$

where $\gamma = H(c_1, c_2, c_3)$.

The correctness of the proposed scheme can be verified as follows:

$$\frac{e(c_1, a_2^3 a_3) \cdot c_2^q a_0 + a_1 \cdot c_4}{e(a_4, c_3)^{\gamma + 1}} = e((g^{-ID} g_1)^t, (g^{-\beta_1} h_1)^{r_1} \cdot (\prod_{c=1}^q z_j^{r_j})^{\gamma} \cdot (g^{-\beta_2} h_2)^{r_2} \cdot (\prod_{c=1}^q z_j^{r_j})^{\gamma}) \cdot c_2^q a_0 + a_1 \cdot c_4$$

$$= e((g^{-ID} g_1)^t, (g^{-\beta_1} h_1)^{r_1} \cdot (g^{-\beta_2} h_2)^{r_2} \cdot (\prod_{c=1}^q z_j^{r_j})^{\gamma}) \cdot c_2^q a_0 + a_1 \cdot c_4$$

$$= e(g, g_1)^{-t(\gamma \beta_1 + \beta_2)} \cdot e(g, h_1^q)^{r_1} \cdot e(g, g)^{(\gamma \beta_1 + \beta_2)} \cdot M \cdot e(g, h_1^q)^{-t}$$

$$= e(g, g')^{-t(\gamma \beta_1 + \beta_2)} \cdot e(g, h_1^q)^{r_1} \cdot e(g, g)^{(\gamma \beta_1 + \beta_2)} \cdot M \cdot e(g, h_1^q)^{-t}$$

$$= M.$$

\section{Security Proof}

The security of the proposed scheme can be proved by the following theorem.

\textbf{Theorem 4.1.} Let $q = (q_E + q_D) \cdot l + 1$. Assume that the truncated decision $(t, \epsilon, q) - ABDHE$ assumption holds in the group $(G, G_T)$. Then, the above forward-secure identity-based encryption scheme is $fs-ID-CCA2$ secure for $t' \geq t - O(t_{exp} \cdot q^2 \cdot l)$, where $t_{exp}$ is the time required to compute an exponentiation in the group $G$.

\textbf{Proof.} Let $A$ be a $(t', \epsilon, q_E, q_D) - fs-ID-CCA2$ adversary against our forward-secure identity-based encryption scheme. We show how to construct a PPT algorithm $B$ to solve the truncated decision $q$-ABDHE problem in $(G, G_T)$ with advantage at least $\epsilon$ and in time at most $t' + O(t_{exp} \cdot q^2 \cdot l)$.

Assume that the algorithm $B$ is given a random truncated decision $q$-ABDHE problem instance $(G, G_T, p, g', g_1, g_0^\alpha \ldots g_0^{\alpha^{l-1}}, Z)$, where $Z$ is either $e(g, g')^{\alpha^{l-1}}$ or a random element of $G_T$. The goal of the algorithm $B$ is to decide whether $Z = e(g, g')^{\alpha^{l-1}}$. To do so, the algorithm $B$ interacts with the adversary $A$ as follows:

\textbf{Setup.} The algorithm $B$ first generates two random polynomial functions $f_1(x)$, $f_2(x) \in \mathbb{Z}_p[x]$ of degree $q$. It then sets $g_1 = g_0^\alpha$, computes $h_1 = g_1 f_1(x)$ and $h_2 = g_2 f_2(x)$ respectively. Clearly, $h_1$ and $h_2$ can be completely computed from the tuple $(g, g_0, \ldots, g_0^{\alpha^{l-1}})$ which is known to the algorithm $B$. It further chooses $2l$ random values $r_1, \ldots, r_l, s_1, \ldots, s_l \in \mathbb{Z}_p$, computes $u_i = g^{r_i}$ and $v_i = g_1^{s_i}$ for $i = 1, \ldots, l$, and sets $U = (u_1, u_2, \ldots, u_l), \overrightarrow{V} = (v_1, v_2, \ldots, v_l)$. Let $T$ be the total number of the time periods and $H$ be a collision-resistant hash function $H : G \times G_T \times G \rightarrow Z_p^*$. The algorithm $B$ outputs $\text{params} = \{T, p, G, G_T, e, g, g_1, h_1, h_2, U, \overrightarrow{V}, H\}$ to the adversary $A$ as the public system parameters.

To answer the adversary $A$’s queries, the algorithm $B$ uses the following subalgorithm $\text{NodeKeyExtract}$ to extract the node keys in the binary tree that is used to update the users’ private keys.
**NodeKeyExtract**(params, ID, ω): Let ω ∈ {0, 1}^{d+1}. This algorithm generates a secret node key skω for the node ω as follows:

1. Parse ω = ω1ω2...ωd;
2. Choose a random value r ∈ Z^*_p and compute

\[ a_0 = f_1(ID), a_1 = f_2(ID), a_2 = g^F_{ID,1}(α) \cdot \left( \prod_{j=1}^{d} z^\omega_j \right)^r, \]
\[ a_3 = g^F_{ID,2}(α) \cdot \left( \prod_{j=1}^{d} z^\omega_j \right)^r, \]
\[ a_3 = (g^{-ID} g_1)^{-r}, b_{d+1} = z^r_{d+1}, \ldots, b_l = z^r_l, \]

where \( z_j = u_j^{-ID} v_j, F_{ID,1}(α) = \frac{f_1(α) - f_1(ID)}{α-ID} \) and \( F_{ID,2}(α) = \frac{f_2(α) - f_2(ID)}{α-ID}; \)
3. Output \( sk_ω = (a_0, a_1, a_2, a_3, b_{d+1}, \ldots, b_l). \)

It is easy to see that \( g^{F_{ID,1}(α)} \) and \( g^{F_{ID,2}(α)} \) can be computed from the tuple \((g, g^α, \ldots, g^{α−1})\) which is known to the algorithm B and \( sk_ω \) is a valid secret node key for the node ω as

\[ a_2 = g^F_{ID,1}(α) \cdot \left( \prod_{j=1}^{d} z^\omega_j \right)^r = g^{\frac{f_1(α) - f_1(ID)}{α-ID}} \cdot \left( \prod_{j=1}^{d} z^\omega_j \right)^r, \]
\[ a_3 = g^F_{ID,2}(α) \cdot \left( \prod_{j=1}^{d} z^\omega_j \right)^r = g^{\frac{f_2(α) - f_2(ID)}{α-ID}} \cdot \left( \prod_{j=1}^{d} z^\omega_j \right)^r. \]

**Phase 1.** In this phase, the adversary A adaptively makes a series of key extraction queries and decryption queries and the algorithm B responds as follows:

- **KeyExtract**(ID, i): When receiving a key extraction query on \((ID, i)\), let ωᵢ denote the node associated with the time period \( i \), the algorithm B generates the user’s private key \( SK^i_D \) in the time period \( i \) by performing the subalgorithm **NodeKeyExtract** recursively to extract the secret node keys of the node \( ωᵢ \) and all right siblings of the nodes on the path from the root to the node \( ωᵢ \). It then outputs the resulting private key \( SK^i_D \) to the adversary A.
- **Decrypt**(ID, \( \langle i, C \rangle \)): When receiving a decryption query on \((ID, \langle i, C \rangle)\), the algorithm B first generates the user’s private key \( SK^i_D \) in the time period \( i \) as above. It then performs the algorithm **Decrypt** to decrypt the ciphertext \( \langle i, C \rangle \) and outputs the result to the adversary A.

**Challenge.** Once the adversary A decides that Phase 1 is over, it outputs \((ID^*, i^*, M_0, M_1)\) on which it wants to be challenged. Let \( ω^* = ω^*_1ω^*_2...ω^*_n ∈ \{0, 1\}^{n≤l} \) be the node associated with the time period \( i^* \) and \( F^*(x) = x^{i^*+2-(ID^*)^{l+2}} \)
\[ = \sum_{j=0}^{q+1} F_j^* \cdot x^j \] be a polynomial function of degree \( q+1 \), where \( F_j^* \) is the coefficient of \( x^j \) in the function \( F^*(x) \). The algorithm B chooses a random bit \( b ∈ \{0, 1\} \) and
computes
\[ c_1^* = g^{\alpha h^{q+2}} \cdot g^{-(1D^*)^{q+2}}, \]
\[ c_2^* = ZF^q_{q+1} \cdot e\left(\prod_{i=1}^{n} (g^{j_i})^{F_{q+1}}, g\right), \]
\[ c_3^* = \prod_{j=1}^{n} (c_1^*)^{j_1 h_2}, \]
\[ c_4^* = M_b \cdot \left(e\left(c_1^*, \left(g^{F_{q+1}}\right)^\gamma \cdot g^{F_{q+1} \cdot 2(q)}\right) \cdot (c_2^*)^{\gamma f_1(1D^*) + f_2(1D^*)}\right)^{-1}, \]

where \( F_{1D^*+1}(\alpha) = \frac{f_1(\alpha) - f_1(1D^*)}{\alpha - 1D^*}, \)
\( F_{1D^*+2}(\alpha) = \frac{f_2(\alpha) - f_2(1D^*)}{\alpha - 1D^*} \) and \( \gamma = H(c_1^*, c_2^*, c_3^*). \)

It then sets \( C^* = (c_1^*, c_2^*, c_3^*, c_4^*) \) and outputs \( \langle i^*, C^* \rangle \) to the adversary \( A \) as the challenge ciphertext.

**Phase 2.** In this phase, the adversary \( A \) issues more key extraction and decryption queries and the algorithm \( B \) responds as in **Phase 1**.

**Guess.** Finally, the adversary \( A \) outputs a guess \( b' \in \{0, 1\} \) for the bit \( b \). If \( b = b' \), then the algorithm \( B \) outputs 1 meaning \( Z = e(g, g')^{a^{q+1}} \). Otherwise, it outputs 0 meaning \( Z \) is a random element of \( G_T \).

Below, we analyze the advantage of the algorithm \( B \) in solving the given truncated decision \( q \)-ABDHE problem.

Let \( s = \log_g g' \cdot F^q(\alpha) \). If \( Z = e(g, g')^{a^{q+1}} \), then we have
\[ c_1^* = g^{a^{q+2}} \cdot g^{-(1D^*)^{q+2}} = g^{a^{q+2} - (1D^*)^{q+2}} = g^{-(1D^*)} = (g^{-1D^*} g_1)^s, \]
\[ c_2^* = ZF^q_{q+1} \cdot e\left(\prod_{i=1}^{n} (g^{j_i})^{F_{q+1}}, g\right) = e\left(\prod_{i=1}^{n} (g^{j_i})^{F_{q+1}}, g\right) = e(g, g)^{\log_g g' \cdot \sum_{j=0}^{q+1} (F_{q+1} a^{j} j_i)} \]
\[ c_3^* = \prod_{j=1}^{n} (c_1^*)^{j_i h_2} = \prod_{j=1}^{n} \left(\sum_{j=1}^{q+1} (g^{-1D^*} g_j)^{j_i h_2}\right)^{j_i h_2} = \left(\prod_{j=1}^{n} (z_i^*)^{j_i h_2}\right)^{j_i h_2}, \]
\[ c_4^* = M_b \cdot \left(e\left(c_1^*, \left(g^{F_{q+1}}\right)^\gamma \cdot g^{F_{q+1} \cdot 2(q)}\right) \cdot (c_2^*)^{\gamma f_1(1D^*) + f_2(1D^*)}\right)^{-1} \]
\[ = M_b \cdot \left(e\left(g^{-1D^*} g_1^s, g^{\gamma \cdot (f_1(\alpha) - f_1(1D^*)) + (f_2(\alpha) - f_2(1D^*))}\right) \cdot \left(e\left(g, g^s\right)^{\gamma f_1(1D^*) + f_2(1D^*)}\right)^{-1} \]
\[ = M_b \cdot \left(e\left(g, h_1^s \cdot e\left(g, h_2\right)^{-s}\right) \right)^{-1}, \]

where \( z_i^* = u_j^{-1D^*} v_j \).

Clearly, when \( Z = e(g, g')^{a^{q+1}} \), then the view of the adversary \( A \) is identical to her view in a real attack game and therefore her guess satisfies \( \Pr\{b = b'\} = 1/2 \geq \varepsilon \). On the other hand, when \( Z \) is only a random element of \( G_T \), then the challenge ciphertext provides no information to the adversary \( A \) and therefore her guess satisfies \( \Pr\{b = b'\} = 1/2 \). Therefore, we get that the advantage of the algorithm \( B \) in solving the given truncated decision \( q \)-ABDHE problem satisfies
\[ \left| \Pr\{B(g', g'^{a^{q+2}}, g, g'^{a^{q+1}}, e(g, g')^{a^{q+1}}) = 1\} - \Pr\{B(g', g'^{a^{q+2}}, g, g'^{a^{q+1}}, Z) = 1\} \right| \]
\[ \geq \left| (1/2 - \varepsilon) - (1/2 - \varepsilon) \right| = \varepsilon. \]

As for the running time, the algorithm \( B \)'s overhead is dominated by performing sub-algorithm **NodeKeyExtract** in response to the adversary \( A \)'s various queries. This sub-algorithm requires computing \( O(q \cdot l) \) exponentiations in the group \( G \) to
generate a node secret key. From the above simulation, the algorithm \( \mathcal{B} \) performs this sub-algorithm at most \( q - 1 = (q_E + q_D) \cdot l \) times. Therefore, we have that the time complexity of the algorithm \( \mathcal{B} \) is bounded by \( t' + O(t_{exp} \cdot q^2 \cdot l) \).

This completes the proof.

5. Comparison analysis

To evaluate the performance of the proposed scheme, we compare it with the forward-secure identity-based encryption schemes \([45, 47]\) in terms of security, storage cost, computation cost (including key extraction time, key update time, encryption time and decryption time) and communication cost (including public parameters size, initial private key size and ciphertext size). For ease of comparison, we assume that \( l = \log_2(T + 2) - 1 \) and \( l' = \log_2 T \), where \( T \) is the total number of the time periods.

5.1. Security. As shown in Table 1, Yu et al.’s scheme \([47]\) can only achieve the weaker fs-ID-CPA security in the standard model; Yao et al.’s scheme \([45]\) achieves the fs-ID-CCA2 security, but in the random oracle model. To our knowledge, our scheme is the first forward-secure identity-based encryption scheme that achieves the fs-ID-CCA2 security directly in the standard model. Therefore, it can provide stronger security guarantee for the practical applications.

| Compared items | Yao et al.’s \([45]\) | Yu et al.’s \([47]\) | Ours |
|----------------|----------------------|----------------------|------|
| Standard model? | No                   | Yes                  | Yes  |
| Security level   | fs-ID-CCA2           | fs-ID-CPA            | fs-ID-CCA2 |

5.2. Storage cost. The storage cost of a forward-secure identity-based encryption scheme is mainly dominated by the private key size because a user’s private key in any time period includes several node secret keys. The storage costs of the compared schemes are listed in Table 2. In Yao et al.’s scheme \([45]\), a user’s private key is composed of at most \( l' + 1 \) node secret keys and each node secret key includes two elements, thus the private key size is at most \( O(l') \) bits. In Yu et al.’s scheme \([47]\), a user’s private key is composed of at most \( l' + 1 \) node secret keys and each node secret key includes at most \( l' + 2 \) elements, thus the private key size is at most \( O(l'^2) \) bits. In our scheme, a user’s private key is composed of at most \( l + 1 \) node secret keys and each node secret key includes at most \( l + 4 \) elements, thus the private key size is at most \( O(l^2) \) bits. The storage cost of our scheme is higher than that of Yao et al.’s scheme, but is lower than that of Yu et al.’s scheme.

| Compared item | Yao et al.’s \([45]\) | Yu et al.’s \([47]\) | Ours |
|---------------|----------------------|----------------------|------|
| Private key size | \( O(l') \)      | \( O(l'^2) \)      | \( O(l^2) \) |
5.3. Computation cost. In our scheme, the key extraction algorithm requires computing four exponentiations in $G$ to generate an initial private key for a user, which only needs $O(1)$ operation time. The key update algorithm requires computing at most $O(l)$ exponentiations in $G$ to generate a new private key, which needs at most $O(l)$ operation time. In the encryption algorithm, computing $\prod_{j=1}^{l'} z_j^{\omega_j}$ requires computing at most $O(l)$ multiplications in $G$ and other operations only need $O(1)$ time. Therefore, the encryption algorithm needs at most $O(l)$ operation time in total. In addition, the decryption algorithm needs $O(1)$ time as the decryption operation is independent of the total number of time periods $T$. As shown in TABLE 3, the time complexities of these four algorithms are $O(l')$, $O(l')$, $O(l')$ and $O(l')$ respectively in Yao et al.’s scheme [45] and $O(l'^2)$, $O(l'^2)$, $O(l')$ and $O(1)$ respectively in Yu et al.’s scheme [47]. Obviously, our scheme is better than Yao et al.’s and Yu et al.’s schemes in the computation performance.

Table 3. Computation costs of the compared forward-secure identity-based encryption schemes

| Compared items | Yao et al.’s [45] | Yu et al.’s [47] | Ours |
|----------------|-------------------|-----------------|------|
| Key extraction time | $O(l')$ | $O(l'^2)$ | $O(1)$ |
| Key update time | $O(l')$ | $O(l'^2)$ | $O(l)$ |
| Encryption time | $O(l')$ | $O(l')$ | $O(l)$ |
| Decryption time | $O(l')$ | $O(1)$ | $O(1)$ |

5.4. Communication cost. In our scheme, the public parameters include $2l + 4$ elements in $G$, therefore, the public parameters size is $O(l)$ bits. The user’s initial private key consists of two elements in $G$ and two elements in $\mathbb{Z}_p^*$. So, the initial private key size is $O(1)$ bits. The ciphertext is composed of two elements in $G$ and two elements in $G_T$. So, the ciphertext size is $O(1)$ bits. As shown in TABLE 4, the sizes of the public parameters, the initial private key and the ciphertext are $O(l')$, $O(l')$ and $O(l')$ bits respectively in Yao et al.’s scheme [45] and $O(l')$, $O(l'^2)$ and $O(1)$ bits respectively in Yu et al.’s scheme [47]. Obviously, our scheme has lower communication cost because the sizes of initial private key and ciphertext are both independent on the total number of time periods $T$.

Table 4. Communication costs of the compared forward-secure identity-based encryption schemes

| Compared items | Yao et al.’s [45] | Yu et al.’s [47] | Ours |
|----------------|-------------------|-----------------|------|
| Public parameters size | $O(l')$ | $O(l')$ | $O(l)$ |
| Initial private key size | $O(l')$ | $O(l'^2)$ | $O(1)$ |
| Ciphertext size | $O(l')$ | $O(1)$ | $O(1)$ |

6. Conclusions

We have presented a new forward-secure identity-based encryption scheme without random oracles. To the best of our knowledge, our scheme is the first forward-secure identity-based encryption scheme that achieves direct chosen-ciphertext security in the standard model. We have proved that the proposed scheme is fs-ID-CCA2 secure under the truncated decision $q$-ABDHE assumption. In the proposed
scheme, the time of key extraction and decryption algorithms and the sizes of initial private key and ciphertext are independent on the total number of time periods $T$ and any other performance parameter has at most log-squared complexity in terms of $T$. Compared with previous forward-secure identity-based encryption schemes, our scheme has obvious advantage in the overall performance.

As the previous forward-secure identity-based encryption schemes, the total number of time periods in our scheme is bounded and known at the time of the system setup. However, in some environments where the number of time periods is very large, a scheme with bounded time periods may be inefficient as its performance depends on the number of time periods. Therefore, the construction of forward-secure identity-based encryption schemes that can support unbounded number of time periods becomes an interesting topic in our future work.

ACKNOWLEDGMENTS

We would like to thank the anonymous referees for their helpful comments. This work is supported by the Nature Science Foundation of China under Grant Nos. 61272542 and 61672207, the Natural Science Foundation of Jiangsu Province Grant No. BK20161511, the Fundamental Research Funds for the Central Universities Grant No. 2016B10114, a Project Funded by the Priority Academic Program Development of Jiangsu Higher Education Institutions and Jiangsu Collaborative Innovation Center on Atmospheric Environment and Equipment Technology.

REFERENCES

[1] M. Abdalla, S. K. Miner and C. Namprempre, Forward-secure threshold signature schemes in Proc. CT-RSA 2001, Springer-Verlag, 2001, 441–456.

[2] M. Abdalla and L. Reyzin, A new forward-secure digital signature scheme in Proc. Asiacrypt. 2000, Springer-Verlag, 2000, 116–129.

[3] R. Anderson, Two Remarks on public key cryptology, in 4th ACM Conf. Comp. Commun. Secur., 1997.

[4] M. Bellare and S. K. Miner, A forward-secure digital signature scheme in Proc. CRYPT. 1999, Springer-Verlag, 1999, 431–448.

[5] M. Bellare and A. Palacio, Protecting against key-exposure: strongly key-insulated encryption with optimal threshold Appl. Algebra Engin. Commun. Comp., 16 (2006), 379–396.

[6] M. Bellare and P. Rogaway, Random oracles are practical: a paradigm for designing efficient protocols in Proc. ACM CCS 1993, ACM, 1993, 62–73.

[7] M. Bellare and B. Yee, Forward security in private-key cryptography in Proc. CT-RSA 2003, Springer-Verlag, 2003, 1–18.

[8] D. Boneh and X. Boyen, Efficient selective-id identity based encryption without random oracles in Proc. Eurocrypt. 2004, Springer-Verlag, 2004, 223–238.

[9] D. Boneh, X. Boyen and E. J. Goh, Hierarchical identity based encryption with constant size ciphertext in Proc. Eurocrypt. 2005, Springer-Verlag, 2005, 440–456.

[10] D. Boneh and M. Franklin, Identity-based encryption from the Weil pairing in Proc. CRYPT. 2001, Springer-Verlag, 2001, 213–229.

[11] X. Boyen, H. Shacham, E. Shen and B. Waters, Forward-secure signatures with untrusted update in Proc. ACM CCS 2006, ACM, 2006, 191–200.

[12] R. Canetti, O. Goldreich and S. Halevi, The random oracle methodology, revisited J. ACM, 51 (2004), 209–218.

[13] R. Canetti, S. Halevi and J. Katz, A forward-secure public-key encryption scheme in Proc. Eurocrypt. 2003, Springer-Verlag, 2003, 255–271.

[14] R. Canetti, S. Halevi and J. Katz, A forward-secure public-key encryption scheme J. Cryptology, 30 (2007), 265–294.

[15] L. Chen and Z. Cheng, Security proof of Sakai-Kasahar’s identity-based encryption scheme in Proc. CRYPT. Coding 2005, Springer-Verlag, 2005, 442–459.
[16] C. Cocks, An identity based encryption scheme based on quadratic residues, in Proc. Crypt. Coding 2001, Springer-Verlag, 2001, 360–363.

[17] W. Diffie, P. C. Van-Oorschot and M. J. Weiner, Authentication and authenticated key exchanges, Des. Codes Crypt., 2 (1992), 107–125.

[18] Y. Dodis, M. Franklin, J. Katz, A. Miyaji and M. Yung, Intrusion-resilient public-key encryption, in Proc. CT-RSA 2003, Springer-Verlag, 2003, 19–32.

[19] Y. Dodis, J. Katz, S. Xu and M. Yung, Key-insulated public-key cryptosystems, in Proc. Eurocrypt. 2002, Springer-Verlag, 2002, 65–82.

[20] C. Gentry, Practical identity-based encryption without random oracles, in Proc. Eurocrypt. 2006, Springer-Verlag, 2006, 445–464.

[21] C. Gentry and A. Silverberg, Hierarchical ID-based cryptography, in Proc. Asiacrypt. 2002, Springer-Verlag, 2002, 548–566.

[22] C. G. Günther, An identity-based key-exchange protocol, in Proc. Eurocrypt. 1989, Springer-Verlag, 1990, 29–37.

[23] G. Hanaoka, Y. Hanaoka and H. Imai, Parallel key-insulated public key encryption, in Proc. PKC 2006, Springer-Verlag, 2006, 105–122.

[24] J. Horwitz and B. Lynn, Toward hierarchical identity-based encryption, in Proc. Eurocrypt. 2002, Springer-Verlag, 2002, 466–481.

[25] G. Itkis and L. Reyzin, Forward-secure signatures with optimal signing and verifying, in Proc. Crypt. 2001, Springer-Verlag, 2002, 499–514.

[26] G. Itkis and L. Reyzin, SiBIR: Signer-base intrusion-resilient signatures, in Proc. Crypt. 2002, Springer-Verlag, 2002, 499–514.

[27] E. Kiltz and Y. Vahlis, CCA2 secure IBE: standard model efficiency through authenticated symmetric encryption, in Proc. CT-RSA 2008, Springer-Verlag, 2008, 221–238.

[28] A. Kozlov and L. Reyzin, Forward-secure signatures with fast key update, in Proc. SCN 2002, Springer-Verlag, 2002, 247–262.

[29] H. Krawczyk, Simple forward-secure signatures from any signature scheme, in Proc. ACM CCS 2000, ACM, 2000, 108–115.

[30] J. Li, F. Zhang and Y. Wang, A strong identity-based key-insulated cryptosystem, in Proc. EUC Workshops 2006, Springer-Verlag, 2006, 352–361.

[31] B. Libert, J. Quisquater and M. Yung, Forward-secure signatures in untrusted update environments, in Proc. ACM CCS 2007, ACM, 2007, 266–275.

[32] Y. Lu and J. G. Li, A practical forward-secure public-key encryption scheme, J. Networks, 6 (2011), 1254–1261.

[33] Y. Lu and J. G. Li, Generic construction of forward-secure identity-based encryption, J. Computers, 7 (2012), 3068–3074.

[34] Y. Lu and J. G. Li, New forward-secure public-key encryption without random oracles, Int. J. Comp. Math., 90 (2013), 2603–2613.

[35] Y. Lu and J. G. Li, An improved certificateless strong key-insulated signature scheme in the standard model, Adv. Math. Commun., 9 (2015), 353–373.

[36] T. Malkin, D. Micciancio and S. K. Miner, Efficient generic forward-secure signatures with an unbounded number of time periods, in Proc. Eurocrypt. 2002, Springer-Verlag, 2002, 400–417.

[37] A. Shamir, Identity-based cryptosystems and signature schemes in Proc. Crypt. 1984, Springer-Verlag, 1984, 17–23.

[38] K. Singh and N. Trichy, Lattice forward-secure identity based encryption scheme, J. Internet Serv. Inf. Sec., 2 (2012), 118–128.

[39] Z. Wan, X. Lai, J. Weng, S. Liu, Y. Long and X. Hong, Certificateless key-insulated signature without random oracles, J. Zhejiang Univ. Sci. A., 10 (2009), 1790–1800.

[40] Z. Wan, X. Meng and X. Hong, Certificateless strong key-insulated signature without random oracles, J. Shanghai Jiao tong Univ. (Sci), 16 (2011), 571–576.

[41] B. Waters, Efficient identity-based encryption without random oracles, in Proc. Eurocrypt. 2005, Springer-Verlag, 2005, 114–127.

[42] J. Weng, X. Li, K. F. Chen and S. L. Liu, Identity-based parallel key-insulated encryption without random oracles, in Proc. Indocrypt. 2006, Springer-Verlag, 2006, 409–423.

[43] J. Weng, S. L. Liu, K. F. Chen, D. Zheng and W. D. Qiu, Identity-based threshold key-insulated encryption without random oracles, in Proc. CT-RSA 2008, Springer-Verlag, 2008, 203–220.

[44] H. Yang, S. Sun and H. Li, Forward-secure identity-based encryption scheme (in Chinese), J. Univ. Electr. Sci. Techn. China, 36 (2007), 534–537.
[45] D. Yao, N. Fazio, Y. Dodis and A. Lysyanskaya, ID-based encryption for complex hierarchies with applications to forward security and broadcast encryption, in Proc. ACM CCS 2004, ACM, 2004, 354–363.

[46] J. Yu, R. Hao, H. Zhao, M. Shu and J. Fan, RIBE: Intrusion-resilient identity-based encryption, Inf. Sci., 329 (2016), 90–104.

[47] J. Yu, F. Y. Kong, X. G. Cheng, R. Hao and J. X. Fan, Forward-secure identity-based public-key encryption without random oracles, Fundam. Inf., 111 (2011), 241–256.

[48] J. Yu, F. Y. Kong, X. G. Cheng, R. Hao and J. X. Fan, Intrusion-resilient identity-based signature: security definition and construction, J. Syst. Softw., 85 (2012), 382–391.

[49] J. Yu, F. Y. Kong, X. G. Cheng, R. Hao and G. W. Li, Construction of yet another forward-secure signature scheme using bilinear maps, in Proc. ProvSec 2008, Springer-Verlag, 2008, 83–97.

Received August 2015; revised December 2015.

E-mail address: luyangnsd@163.com
E-mail address: ljg1688@163.com