Efficient multi-receiver identity-based signcryption from lattice assumption

Xiaojun Zhang*
School of Computer Science and Engineering,
University of Electronic Science and Technology of China,
2006 Xi Yuan Avenue, West High-tech Zone,
Chengdu 611731, China
and
School of Computer Science,
Southwest Petroleum University,
Xindu Avenue, No.8, Xindu Zone,
Chengdu 610500, China
Email: zhangxjdzkd2012@163.com
*Corresponding author

Chunxiang Xu and Jingting Xue
School of Computer Science and Engineering,
University of Electronic Science and Technology of China,
2006 Xi Yuan Avenue, West High-tech Zone,
Chengdu 611731, China
Email: chxxu@uestc.edu.cn
Email: 474028088@qq.com

Abstract: Signcryption is a public-key cryptographic primitive which combines the functions of public-key encryption and digital signature into a single logical step at low computational and communication costs. While multi-receiver signcryption is suited for a situation where a sender wants to send a signcrypted message to multiple receivers in a confidential and authenticated way. Due to this attractive property, recently, multi-receiver signcryption plays an important role in some practical applications such as virtual conference as well as authenticated mail transferring. In this paper, we present an efficient multi-receiver identity-based signcryption (MIBSC) scheme from lattice assumption which is believed to resist quantum computer attacks. The proposed scheme is provably secure in the random oracle model, which has the indistinguishability against chosen ciphertext attacks under the hardness of learning with errors (LWE), and existentially unforgeability against chosen message attacks under the small integer solution assumption (SIS). Moreover, we also compare our MIBSC scheme with existing schemes from performance efficiency and security, the result shows that our proposed scheme is more efficient and more secure. In particular, our scheme can be properly applied in the post-quantum communication environments.

Keywords: multi-receiver signcryption; lattice assumption; post-quantum cryptography; learning with errors; LWE; small integer solution assumption; SIS.
1 Introduction

As is known to all, encryption and signature are basic cryptographic primitives to achieve confidentiality and authenticity. If we want to achieve the two properties simultaneously, the traditional approach, for example, a secure emailing service, where the message should be encrypted and signed to provide confidentiality and authenticity. In 1997, Zheng proposed a new cryptographic primitive called signcryption that offers confidentiality and unforgeability simultaneously, but with less computational complexity and lower communication cost than the traditional signature-then-encryption technique. The performance advantage of signcryption has made it a suitable primitive for applications that require secure and authenticated message, such as smart cards, electronic commerce, mobile communications. Since Zheng’s work, most signcryption schemes have were proposed such as Steinfeld and Zheng (2000), Chen and Lee (2005), Malone-Lee and Mao (2003), Boyen (2003) and Barreto et al. (2005).

Shamir (1984) first introduced the concept of identity-based cryptography in 1984. The main idea is that a public key can be derived from arbitrary string such as telephone number, email address and so on, while the corresponding private key can be generated by the trusted key generation centre (KGC). Thus identity-based cryptography provides a more convenient alternative to the traditional public key infrastructure (PKI). Malone-Lee (2002) proposed the first identity-based signcryption scheme. However, after that it was found Malone-Lee’s scheme was not semantically secure. Since then, quite a few identity-based signcryption schemes have been proposed, such as Libert and Quisquator (2003, 2004) and Chow et al. (2003).
With the rapid growth of wireless network technology, group communications have become more and more important, such as video conference, collaborative work and so on. Multi-receiver signcryption is very useful to secure group communications, since it can make sure that a sender broadcasts a message to multiple receivers in a secure and authenticated manner, thus a group of persons are jointly working in the same team to communicate with each other while achieving both privacy and unforgeability.

The concept of multi-receiver setting was first proposed by Bellare et al. (2000). In 2002, Baek et al. (2007) formalised identity-based encryption to the multi-receiver setting. Duan and Cao (2006) were the first to come up with an identity-based scheme for multi-receiver signcryption. Their scheme needs only one pairing operation to signcrypt a message for multiple receivers. While Yu et al. (2007) gave another scheme which is more efficient in the unsigncryption phase than the scheme in Duan and Cao (2006). Li et al. (2009) pointed out Yu et al.’s scheme does not satisfy the unforgeability, and they presented a more efficient scheme than all previous schemes. However, according to Shor’s work (1997), once the quantum computer turns up, these traditional schemes are faced with security threats. Therefore, it is urgent to engage in researching cryptographic algorithms which can resist quantum computer attacks; we call them as post-quantum cryptography. In post-quantum cryptography, lattice-based cryptography is intriguing due to its security on the worst-case hardness of lattice problems under a quantum reduction. Moreover, the computational costs of lattice-based cryptography often require modular addition, which is very suitable for limited computing devices. In the recent decade, the GPV signature scheme in Gentry et al. (2008) and LWE-based encryption in Regev (2005) have been proposed, after that many novel lattice-based signature and encryption schemes (Zhang et al., 2014a, 2014b; Gorbunov et al., 2015; Boyen and Li, 2016; Brakerski and Vaikuntanathan, 2016; Mukherjee and Wichs, 2016) have also been proposed. Recently, combining the properties of signature and encryption, Li et al. (2013) made an attempt to construct lattice-based signcryption scheme in the random oracle model. Subsequently, Lu et al. (2014) proposed a lattice-based signcryption scheme without random oracles. However, compared with traditional signcryption schemes, research on signcryption based on lattice seems to be much less, which needs researcher’ more attention.

In this paper, we propose an identity-based signcryption for multiple receivers with lattice basis delegation and LWE-based encryption techniques. The scheme is provably secure against chosen ciphertext attacks in the random oracle model under the learning with errors (LWE) assumption. Moreover, the scheme is strongly unforgeable against adaptive chosen message attacks in the random oracle under the small integer solution (SIS) assumption. Our proposed scheme is constructed on identity-based systems, thus can avoid the complex management of PKI. We also compare the performance efficiency with existing schemes and show that our proposed scheme is more efficient and more practical in the distributed system of post-quantum cryptographic communication environments.

The rest of this paper is organised as follows. Some definitions and facts about lattice are given in Section 2. In Section 3, we define multi-receiver identity-based signcryption and its security model. In Section 4, we propose our MIBSC scheme from lattice assumption, and we also prove that it can guarantee confidentiality and unforgeability. In Section 5, we give the performance comparison of our MIBSC scheme with existing schemes. Finally, in Section 6, we make a conclusion.
2 Preliminaries

Let $B = \{b_1, \ldots, b_m\} \in \mathbb{R}^{m \times m}$ be $m$ linearly independent vectors. The $m$-dimensional lattice generated by $B$ is the set $\mathcal{L}(B) = \{Bs = \sum_{i=1}^{m} s_i b_i : s_i \in \mathbb{Z}^m\}$.

**Definition 1** [q-ary lattices in Ajtai (1999)]: For $q$ a prime, $A \in \mathbb{Z}_q^{n \times m}$ and $u \in \mathbb{Z}_q^n$, we define as follows.

$$\Lambda(A) = \{y \in \mathbb{Z}^n : \exists s \in \mathbb{Z}_q^n, y = A^T s \text{ mod } q\}$$

$$\Lambda_q^v(A) = \{e \in \mathbb{Z}^n : Ae = 0 \mod q\}$$

$$\Lambda_q^u(A) = \{e \in \mathbb{Z}^n : Ae = u \mod q\}$$

We observe that if $t \in \Lambda_q^v(A)$, then $\Lambda_q^v(A) = \Lambda_q^v(A) + t$ and hence $\Lambda_q^v(A)$ is a shift of $\Lambda_q^v(A)$.

2.1 Discrete Gaussian on lattice

Now we introduce Gaussian distribution on lattice (Gentry et al., 2008) briefly. Let $\Lambda$ be a subset of $\mathbb{Z}_q^n$, for any vector $c \in \mathbb{R}^n$ and any positive parameter $\sigma \in \mathbb{R}$, let $\rho_{\sigma,c}(x) = \exp(-\pi \|x-c\|^2)/\sigma^\pi$ and $\rho_{\sigma,c}(\Lambda) = \sum_{x \in \Lambda} \rho_{\sigma,c}(x)$. The discrete Gaussian distribution on $\Lambda$ with parameter $\sigma$ and centre $c$ is defined as follows: $\forall y \in \Lambda$, $\mathcal{D}_{\Lambda,\sigma,c} = \rho_{\sigma,c}(y)/\rho_{\sigma,c}(\Lambda)$.

Here the distribution $\chi = \mathcal{D}_{\Lambda,\sigma,c}$ is often defined on the lattice $\Lambda^\perp(A)$ for $A \in \mathbb{Z}_q^{m \times n}$ or on a coset $\Lambda^\perp(A) = \Lambda^\perp(A) + z$ where $z \in \mathbb{Z}_q^n$.

2.2 The LWE and SIS hardness assumptions

We recall the LWE and the SIS problems, which will be considered as average-case problems related to the family of modular lattices.

We first denote an integer $q = q(n)$, a Gaussian error distribution $\chi$ and a vector $s \in \mathbb{Z}_q^u$, the distribution of the variable $(a, a^T s + x)$ on $\mathbb{Z}_q^n \times \mathbb{Z}_q$ is denoted by $A_{s,\chi}$, where the vector $a \in \mathbb{Z}_q^n$ is uniformly random and $x \in \mathbb{Z}_q$ is sampled from $\chi$. The LWE problem is defined as follows (Regev, 2005):

**Definition 2:** For an integer $q = q(n)$ and a Gaussian error distribution $\chi$ on $\mathbb{Z}_q$, the goal of the learning with errors problem $LWE_{q,\chi}$ is to distinguish between the distribution $A_{s,\chi}$ for some random secret $s \in \mathbb{Z}_q^n$ and the uniform distribution on $\mathbb{Z}_q^n \times \mathbb{Z}_q$.

**Definition 3** [The SIS problem (Gentry et al., 2008)]: Given an integer $q$, a matrix $A \in \mathbb{Z}_q^{m \times n}$, and a real $\beta$, find a non-zero integer vector $e \in \mathbb{Z}^n$ such that $Ae = 0 \mod q$ and $0 < \|e\| \leq \beta$. 


For functions \( q(n), m(n) \) and \( \beta(n) \), \( \text{SIS}_{q,m,\beta} \) is the ensemble over instances \( (q(n), m(n), \beta(n)) \) where \( A \in \mathbb{Z}_q^{n \times m} \) is uniformly random.

**Lemma 1** (see Regev, 2005): Let \( q = q(n) \) be a prime and \( \alpha = \alpha(n) \in (0, 1) \) such that \( \alpha q > 2\sqrt{n} \). If there is an efficient (possibly quantum) algorithm that solves \( \text{LWE}_{q,n} \), then there is an efficient quantum algorithm for approximating shortest independent vectors problem and Gap shortest vectors problem in the \( l_2 \) norm, in the worst case, to within \( \tilde{O}(n/\alpha) \) factors.

**Lemma 2** (see Gentry et al., 2008): For any poly-bounded \( m, \beta = \text{poly}(n) \) and for any prime \( q \), the average-case problems \( \text{SIS}_{q,m,\beta} \) is as hard as approximating the problems \( \text{SIVP} \) in the worst case to within certain factors \( (n \log q)^{\gamma} \).

### 2.2.1 Trapdoor functions and preimage sampleable functions

Now we show an algorithm that generates a uniform matrix \( A \in \mathbb{Z}_q^{n \times m} \) and a trapdoor matrix \( T \in \mathbb{Z}_q^{n \times m} \) satisfying the following conditions in Lemma 3.

**Lemma 3** (see Alwen and Peikert, 2009): For a prime \( q \geq 2 \) and \( m \geq 5n \log q \), the probabilistic polynomial-time algorithm \( \text{TrapGen}(q, n) \) takes as input \( 1^n, 1^m \) and \( q \), outputs matrices \( A \in \mathbb{Z}_q^{n \times m} \) and \( T \in \mathbb{Z}_q^{n \times m} \) such that:

1. The distribution on \( A \) is statistically close to uniform over \( \mathbb{Z}_q^{n \times m} \).
2. The columns of \( T \) form a basis of the lattice \( \Lambda_q^{\perp} \). Particularly, \( AT = 0 \mod q \).
3. \( \| T \| = O(n \log q) \) and \( \| \tilde{T} \| \leq C\sqrt{n \log q} \), for some absolute constant \( C < 40 \).

Now, we utilise the technique in Gordon et al. (2010) to show how to recover an LWE instance \( (A, \mu = A^\perp s + e) \), where \( e \) is a short vector sampled from a discrete Gaussian distribution \( \chi \). Once we get the knowledge of the trapdoor \( T \). Especially, if \( \| T \| < L = O(n \log q) \) and \( e \) is drawn from \( \Psi_{\alpha}^{m,n} \) for \( \alpha \leq 1/(L \log(n)) \), then \( s \) can be easily recovered. Since \( T^\perp \mu(\text{mod } q) = T^\perp (A \cdot s + e)(\text{mod } q) = (AT)^\perp s + T^\perp e(\text{mod } q)T^\perp e(\text{mod } q) \), where both \( T \) and \( e \) contain only small entries, each entry of the vector \( T^\perp e \) is smaller than \( q \) and thus \( T^\perp e(\text{mod } q) \) is equal to \( T^\perp e \). Multiplying by \( (T^\perp)^{-1} \), thus gets \( e \), after which it is easy to recover \( s \).

In 2008, Gentry et al. employed the trapdoor sampling procedure to construct the one-way preimage sampleable function, which consists of the following three algorithms:

1. Generating a function with trapdoor: Let \( q \) be odd prime and \( m = \lceil 6n \log q \rceil \). \( \text{TrapGen}(q, n) \) outputs a pair \( (A \in \mathbb{Z}_q^{n \times m}, T \in \mathbb{Z}_q^{n \times m}) \) such that \( A \) is statistically close to a uniform matrix in \( \mathbb{Z}_q^{n \times m} \) and \( T \) is a basis of \( \Lambda_q^{\perp} \) satisfying \( \| T \| \leq m(\sqrt{\log m}) \) and \( \| T \| \leq O(n \log q) \) with all but negligible probability.
2. Domain sampling with a trapdoor: \( \text{SampleDom}(1^n) \) samples \( x \) from distribution \( \mathcal{D}_{2^m,n} \).
3 Preimage sampling with a trapdoor: $\text{SamplePre}(A, T, u, r)$ on input of $A \in \mathbb{Z}_q^{n \times m}$, a good basis $T \in \mathbb{Z}_q^{n \times m}$ of $\Lambda_\perp(A)$ as the trapdoor, a vector $u \in \mathbb{R}_m$ and a parameter $r$, satisfying $r > \|T\|_\omega \log m$.

The algorithm works as follows. First, it chooses an arbitrary $t \in \mathbb{Z}_q^n$ via linear algebra equation $At = u \mod q$ (except for a negligible fraction of $A$, such that $t$ always exists). Then it outputs $e \leftarrow D_{\Lambda_\perp(A)+\ast}(A)$.

### 2.3 Lattice basis delegation

We first denote a matrix $R \in \mathbb{Z}_q^{m \times n}$ is $\mathbb{Z}_q$-invertible if $R \mod q$ is invertible and all the columns of $R$ are low norm.

Now we introduce the algorithm $\text{SampleR}(1^n)$ described in Agrawal et al. (2010) as follows:

1. Let $T$ be the canonical basis of the lattice $\mathbb{Z}^n$.
2. For $i = 1, \ldots, m$ do $r_i \leftarrow \text{SamplePreGaussian}(\mathbb{Z}_q^n, T, \sigma_R, 0)$.
3. If $R$ is $\mathbb{Z}_q$-invertible, output $R$, otherwise repeat Step 2.

Here we describe the distribution $\mathcal{D}_{m \times n}$ as follows:

$\mathcal{D}_{m \times n}$ denotes the distribution on matrix in $\mathbb{Z}_q^{m \times n}$ which is defined as $(\mathcal{D}_{Z^m, \sigma_R})^m$ conditioned on the resulting matrix being $\mathbb{Z}_q$-invertible. Here we define the parameter $\sigma_R = \sqrt{n \log q O(\log m)}$, and $\mathbb{Z}_q$-invertible means that the matrix $R \mod q$ is invertible as a matrix in $\mathbb{Z}_q^{m \times n}$.

Now, we introduce lattice basis delegation technique from Agrawal et al. (2010). Set $A$ to be a matrix in $\mathbb{Z}_q^{n \times m}$ and $T_A$ to be a short basis of $\Lambda_\perp(A)$. We define $B = AR^{-1}$ in $\mathbb{Z}_q^{n \times m}$, where $R$ is low norm matrix in $\mathbb{Z}_q^{m \times m}$.

We note that the dimension of $B$ is the same as dimension of $A$. It is also required that it is hard to recover short basis of $\Lambda_\perp(A)$ from the short basis of $\Lambda_\perp(B)$. Now the basis delegation algorithm $\text{BasisDel}(A, R, T_A, \sigma)$ is described in the following lemma.

**Lemma 4** (see Agrawal et al., 2010): Let $q > 2$, $A \in \mathbb{Z}_q^{n \times m}$ and $R \in \mathbb{Z}_q^{m \times n}$ be a matrix sampled from $\mathcal{D}_{m \times n}$. Let $T_A$ be a basis of $\Lambda_\perp(A)$, there exists a PPT algorithm $\text{BasisDel}(A, R, T_A, \sigma)$ that outputs a random basis $B$ for $\Lambda_\perp(AR^{-1})$ such that $\|B\| \leq \sigma \sqrt{m}$, where $\sigma \geq \|T_A\| \mod (\log^2 m)$.

All of our security proofs employ an algorithm $\text{SampleRwithBasis}(A)$, the algorithm takes a matrix $A \in \mathbb{Z}_q^{n \times m}$ as input, and outputs a low-norm matrix $R$ sampled from $\mathcal{D}_{m \times n}$ and a short basis $T_A$ for $\Lambda_\perp(AR^{-1})$. The algorithm is described as follows:

1. Let $a_1, \ldots, a_m \in \mathbb{Z}_q^n$ be the $m$ columns of the matrix $A \in \mathbb{Z}_q^{n \times m}$.
2 Run TrapGen\((q, n)\) to generate a random rank \(n\) matrix \(B \in \mathbb{Z}_q^{n \times m}\) and a basis \(T_B\) of \(\Lambda_q(B)\) such that \(||\vec{T}_B|| \leq \mathcal{L} ||G|| = \sigma_R/\omega(\sqrt{\log m})\).

3 For \(i = 1, \ldots, m\) do:
   3a Sample \(r_i \in \mathbb{Z}^n\) as the output of SamplePre\((B, T_B, a_i, \sigma_R)\), then \(Br_i = a_i \mod q\) and \(r_i\) is sampled from a distribution statistically close to \(\mathcal{D}_\Lambda(B, a\sigma_R)\).
   3b Repeat Step 3a until \(r_i\) is \(\mathbb{Z}_q\) linearly independent of \(r_1, \ldots, r_{i-1}\).

4 Let \(R \in \mathbb{Z}_q^{n \times m}\) be the matrix whose columns are \(r_1, \ldots, r_m\). Then \(R\) has rank \(m\) over \(\mathbb{Z}_q\). Output \(R\) and \(T_B\).

According to the construction \(BR = A \mod q\) and thus \(B = AR^{-1} \mod q\). Therefore, the basis \(T_B\) is a short basis of \(\Lambda_q(AR^{-1})\).

The following lemma shows that \(R\) is sampled from a distribution close to \(\mathcal{D}_{\Lambda_q}\).

Lemma 5 (see Agrawal et al., 2010): For \(m > 2n \log q\), \(q\) an odd prime and for all but at most a \(q^{-x}\) fraction of rank \(n\) matrix \(A \in \mathbb{Z}_q^{n \times m}\), the algorithm SampleRwithBasis\((A)\) outputs a random matrix \(R\) in \(\mathbb{Z}_q^{n \times m}\) sampled from a distribution statistically close to \(\mathcal{D}_{\Lambda_q}\). Moreover, the generated basis \(T_B\) of \(\Lambda_q(AR^{-1})\) satisfies \(||\vec{T}_B|| \leq \mathcal{L} ||G|| = \sigma_R/\omega(\sqrt{\log m})\) with overwhelming probability.

3 Definitions of multi-receiver identity-based signcryption

In this section, we first give a general definition of multi-receiver identity-based signcryption for sending a single signcrypted ciphertext to multiple receivers which was first formalised in Duan and Cao (2006).

Definition 4: A multi-receiver identity-based signcryption (MIBSC) scheme consists of the following four probability polynomial time algorithms:

- **Setup**: Given a security parameter \(\kappa\), the KGC generates a master public key \(mpk\) and a master secret key \(msk\). The \(mpk\) is given to all interested parties while the \(msk\) is kept secret.
- **Extract**: Providing an identity \(ID\) received from a user and its master secret key \(msk\) as input, the KGC computes the corresponding secret key associated with \(ID\), denoted by \(SK_{ID}\), and transmits it to \(ID\) in a secure way.
- **Signcrypt**: To send a message \(M\) to a set of receivers with identities are \(ID_1, \ldots, ID_l\) respectively, the sender whose identity is \(ID_s\) runs the signcrypt algorithm to obtain a signcryption \(C = \text{Signcrypt}(M, SK_{ID_s}, ID_1, \ldots, ID_l)\).
- **Unsigncrypt**: Upon receiving a signcryption \(C\) from the sender \(ID_s\), the receiver \(ID_t\) runs the algorithm \(\text{Unsigncrypt}(C, SK_{ID_t}, ID_t)\) to obtain either the plaintext message \(M\) or invalid according to the fact whether the \(C\) was a valid signcryption.
For consistency, we require that if the signcryption \( C = \text{Signcrypt}(M, SK_{ID_s}, ID_1, \ldots, ID_l) \), then \( M = \text{Unsigncrypt}(C, SK_{ID_s}, ID_i) \) for \( 1 \leq i \leq l \).

### 3.1 Security definitions

Now we propose security models for confidentiality and unforgeability of MIBSC scheme given by Duan and Cao (2006). During the selective identity attack, an adversary commits ahead of time to multiple target identities on which they will be challenged.

**Confidentiality:** In our multi-receiver identity-based signcryption scheme, with respect to confidentiality, we refer to it as indistinguishability of ciphertexts under selective multiple identities, chosen ciphertext attacks (IND-sMIBSC-CCA), which is defined as follows.

**Definition 5:** A multi-receiver identity-based signcryption scheme is semantically secure against chosen ciphertext attacks (IND-sMIBSC-CCA) if no probabilistic polynomial time adversary has a non-negligible advantage in the following game:

- **Setup:** The challenger \( B \) sends the system public key \( mpk \) to \( A \). After receiving the system parameter, the adversary \( A \) chooses a list of multiple target identities, denoted by \( ID_1^*, \ldots, ID_l^* \) of users \( \{R_1, \ldots, R_l\} \), for which \( A \) is not allowed to query their secret keys.

- **Phase 1:** In this phase, \( A \) makes polynomial number of queries to the following oracles:
  1. **Secret key extraction queries:** \( A \) produces an identity \( ID \) and queries for the secret key of user \( ID \), the challenger runs the secret key extraction algorithm to get \( SK_{ID} = \text{Extract}(msk, ID) \). A restriction here is that \( ID \neq ID_i^* \) for \( i = 1, \ldots, l \).
  2. **Signcrypt queries:** \( A \) produces a message \( M \), a sender identity \( ID_s \), and a list of receiver identities \( ID_1, \ldots, ID_l \), the challenger \( B \) computes the secret key \( S_{ID}^* \) and runs the signcrypt algorithm to generate \( C = \text{Signcrypt}(M, ID_s^*, ID_1^*, \ldots, ID_l^*) \).
  3. **Unsigncrypt queries:** \( A \) produces a sender identity \( ID_s \), a receiver identity \( ID_i^* \) and a signcryption \( C \). Then \( A \) requires the result of \( \text{Unsigncrypt}(C, SK_{ID_s}, ID_i^*) \). \( B \) runs the secret key extraction algorithm to get \( SK_{ID_s} \) and returns the corresponding message \( M \) to \( A \) if \( C \) is a valid signcryption from \( ID_s \) to \( ID_i^* \); otherwise the challenger \( B \) returns invalid.

- **Challenge:** The adversary \( A \) produces two equal length plaintexts \( M_0, M_1 \) and an arbitrary sender identity \( ID_s \). The challenger \( B \) flips a coin \( b \leftarrow \{0, 1\} \) to compute a signcryption \( C = \text{Signcrypt}(M_b, SK_{ID_s}, ID_s^* \ldots, ID_l^*) \) with the sender’s secret key \( SK_{ID_s} \). Finally, \( C \) is sent to \( A \) as a challenged signcryption.
• Phase 2: \( \mathcal{A} \) issues new queries as in Phase 1, with the following restrictions. It should not query the unsigncryption of the challenge \( C \) with the secret key \( S_{i \ast} \) for any \( i \in [1, n] \).

• Guess: At the end of the game, \( \mathcal{A} \) outputs a bit \( b' \) and wins if \( b' = b \).

\( \mathcal{A} \)'s advantage is defined to be \( \text{Adv}_{\text{IND-\text{sMIBSC-CCA}}}^{b \to b'}(\mathcal{A}) = 2 \Pr[b' = b] - 1 \).

**Strongly existential unforgeability:** A multi-receiver identity-based signcryption scheme against strong existential unforgeability under selective multi-identity, chosen message attacks (SUF-sMIBSC-CMA) is defined as follows.

**Definition 6:** A multi-receiver identity-based signcryption scheme is said to be strongly existential unforgeable against chosen-message attacks (SUF-sMIBSC-CMA) if no probabilistic polynomial time forger has a non-negligible advantage against a challenger \( \mathcal{B} \) in the following game:

• Setup: The challenger \( \mathcal{B} \) sends the system public key \( mpk \) to \( \mathcal{F} \). After receiving the system parameter, the adversary \( \mathcal{F} \) chooses a list of multiple target identities, denoted by \( \{ID_1, \ldots, ID_l\} \), for which \( \mathcal{F} \) is not allowed to query their secret keys.

• Attack: \( \mathcal{F} \) makes polynomial number of queries to the same oracle as described in Definition 5.

• Forger: After a polynomial number queries as above during the attack phase, the adversary \( \mathcal{F} \) generates \( C \) as the signcryption of a message \( M \) with a sender \( ID_s \) and \( l \) arbitrary receivers \( ID_{R_1}, \ldots, ID_{R_l} \). \( \mathcal{F} \) wins if \( C \) decrypts under any secret key of these receivers, to be a signed message \( (M, e, ID_s') \) for some \( s \in [l] \) that satisfies the pair \((M, e)\) is valid for the sender \( ID_s' \) and \( e \) was not the output of a signcrypt query \( \text{Signcrypt}(M, ID_s, ID_{R_1}, \ldots, ID_{R_l}) \).

4 Multi-receiver identity-based signcryption from lattice assumption

In this section, we present our construction from lattice assumption that is based on the GPV signature scheme (Gentry et al., 2008), and our scheme is motivated by Li et al. (2013).

Let \( k_0 \) be the identity length and \( k_1 \) be the message length. We assume a hash function \( H_1 \) that outputs matrix in \( \mathbb{Z}_q^{m \times n} \): \( H_1 : \{0,1\}^{k_0} \to \mathbb{Z}_q^{m \times n} \), namely \( ID \to H_1(ID) \sim \mathcal{D}_{\text{noiv}} \), where \( H_1(ID) \) is distribution as \( \mathcal{D}_{\text{noiv}} \). We also define two secure hash functions as \( H_2 : \{0,1\}^* \to \mathbb{Z}_q^* \), \( H_3 : \{0,1\}^* \to \{0,1\}^{k_0+k_1} \). Our multi-receiver identity-based signcryption scheme consists of the following four algorithms:
Efficient multi-receiver identity-based signcryption from lattice assumption

• **Setup:** Given the security parameter $n$. The KGC runs the algorithm $\text{TrapGen}(q, n)$ to generate a matrix $A \in \mathbb{Z}_q^{nm \times m}$ and a corresponding short basis $T_A \in \mathbb{Z}_q^{m \times m}$. Then, it sets $T_A$ as a master secret key, $A$ as the master public key.

• **Extract:** Given a user identity string $ID \in \{0, 1\}^{k_A}$, the KGC calculates the user’s secret key as follows:
  
  1. Let $R_{ID} = H_1(ID) \in \mathbb{Z}_q^{nm \times m}$, set $F_{ID} = A(R_{ID})^{-1}$ as the public key of the user
  2. the KGC evaluates $\text{BasisDel}(A, R_{ID}, T_A, \sigma)$ to generate a corresponding secret key $SK_{ID}$ for $ID$, where $SK_{ID}$ is a random basis for $\Lambda_q(m_A)$.

Then the KGC transmits the secret key $SK_{ID}$ to its owner in a secure way.

• **Signcrypt:** Suppose Alice whose identity is $ID_A$ wants to signcrypt a message $M \in \{0, 1\}^{k_M}$ to $l$ different receivers $ID_1, \ldots, ID_l$. Alice performs as follows:
  
  1. Alice selects a random string $r \in \{0, 1\}^{k_R}$, and computes $\mu = H_2(ID_A \parallel M \parallel r)$
  2. Alice evaluates $e \leftarrow \text{SamplePre}(F_{ID_A}, SK_{ID_A}, \mu, \delta)$
  3. Alice chooses distinct $s_1, s_2, \ldots, s_l \in \mathbb{Z}_q^m$ randomly
  4. for $i = 1$ to $l$, Alice computes $U_i = (c_i, d_i)$, where $c_i = H_3(s_i, e) \oplus (M \parallel ID_A)$ and $d_i = F_{ID_A}^{s_i} + e$. $F_{ID_A} = A(R_{ID_A})^{-1}$.

The signcryption is $C = (r, U_1, \ldots, U_l, \mathcal{L})$, where $\mathcal{L}$ is a label that contains information about how $U_i$ is associated with each receiver.

• **Unsigncrypt:** Once receiving a signcryption $C = (r, U_1, \ldots, U_l, \mathcal{L})$ from Alice whose identity is $ID_A$, each receiver only needs to extract the $i^{th}$ component $U_i = (c_i, d_i)$ from $(U_1, U_2, \ldots, U_l)$, then the receiver $ID_i$ performs in the following steps:
  
  1. With the secret key $SK_{ID_i}$, the receiver $ID_i$ can recover $(s_i, e)$ from $d_i$.
  2. With the pairing $(s_i, e)$, the receiver $ID_i$ can compute $M' = H_3(s_i, e) \oplus c_i$ and check whether the length equals to $k_0 + k_1$ bits. If the last $k_0$ bits of $M'$ equals to $ID_A$, then it can receive $M$ from the first $k_1$ bits of $M'$. Otherwise, it aborts.

Once receiving the message $M$, with the public key $F_{ID_A} = A(R_{ID_A})^{-1}$ of $ID_A$, the receiver accepts the signcryption if and only if both the following conditions satisfied:

  1. $r \in \{0, 1\}^{k_R}$, $0 < |e| \leq \delta \sqrt{m}$
  2. $F_{ID_A} e = H_2(ID_A \parallel M \parallel r)$.

• **Correctness:** It is easy to see that the above Unsigncrypt algorithm is consistent. Since each receiver $ID_i$ can use his owner secret key $SK_{ID_i}$ to decrypt the signcryption $C$ and obtained the valid signed message $M$ such that $F_{ID_i} e = H_2(ID_A \parallel M \parallel r)$ and $0 < |e| \leq \delta \sqrt{m}$, thus $C$ is valid signcryption.
4.1 Security analysis

Theorem 1: In the random oracle model, if there exists a polynomial time adversary \( A \) that can break the IND-sMIBSC-CCA security of our MIBSC scheme with non-negligible probability \( \varepsilon \) when performing \( q_{hi} \) queries to oracles \( (1, 2, 3) \), \( q_{sc} \) signcrypt queries and \( q_u \) unsigncrypt queries, then there is an algorithm \( B \) that can also solve the LWE problem with a non-negligible probability \( \varepsilon' \geq \frac{1}{l}(1 - \frac{q_{hi}q_u}{2^n}) \).

Proof: For contradiction, we assume that there is an adversary \( A \) who can win the IND-sMIBSC-CCA security with non-negligible probability \( \varepsilon \). In the following, we show how to build a challenger \( B \) that solves the LWE problem by running the adversary \( A \) as a subroutine. Suppose \( B \) is given an instance of the LWE problem \( (A_*, \vec{a}_u = A_*^T s_u + e_u) \). \( B \)'s goal is to find the solution of the LWE problem. We assume that \( A \) will ask for \( H_i(ID) \) before \( ID \) is used in any key extraction, signcrypt, and unsigncrypt queries. \( B \) also sets the random oracles \( H_1, H_2, H_3 \). The answers to the oracles \( H_1, H_2, \) and \( H_3 \) are randomly selected. Therefore, to maintain consistency, \( B \) will maintain three lists \( L_1, L_2 \) and \( L_3 \) which are initialised to be empty.

Actually, \( B \) plays the role of \( A \)'s challenger and works by interacting with \( A \) in a game defined as follows:

- **Setup:** \( B \) first obtains \( lm \) samples \( (u_i, v_i) \in Z_q^n \times Z_q \) from the LWE oracle. Next, \( B \) parses these LWE samples as \( (A'_i, d'_i) \in Z_q^n \times Z_q \). \( B \) samples \( l \) random matrices \( R'_1, R'_2, \ldots, R'_l \) by running \( R'_i \leftarrow \text{SampleR}(1^n) \) for \( i = 1, \ldots, l \). Finally, \( B \) chooses a random \( w \in [l] \), sets \( A = A'_w \), and sends it to the adversary \( A \). Then \( A \) outputs multiple target identities, denoted by \( L = (ID'_1, \ldots, ID'_l) \).

- **Phase 1:** The adversary \( A \) now adaptively queries on the various oracles \( O_{hi}, O_{hi}, O_{hi} \), key extraction, signcrypt, and unsigncrypt queries. The descriptions of these queries as follows.

  Oracle \( O_{hi}(ID) \): \( B \) checks if there exists a tuple \( (ID, R, B, T) \) in \( L_1 \). If such a tuple exists, \( B \) answers with \( R \). Otherwise, \( B \) does the following.

  If \( ID = ID'_j \) for some \( j \in [l] \), then \( B \) returns \( R'_j \) to \( A \). Otherwise \( B \) runs \( \text{SampleRwithBasis}(A) \) to obtain a random matrix \( R_t \sim D_{\text{ Rand}} \) and a short basis \( T_t \) for \( B_t = AR_{t}^{-1} \), stores the tuple \( (ID, R, B, T) \) in list \( L_1 \) and returns \( O_{hi}(ID) = R \) to \( A \).

  Oracle \( O_{hi}(ID, ||M||, r) \): For an \( O_{hi}(ID, ||M||, r) \) query, \( B \) first checks if the value of \( O_{hi} \) was previously defined for the input \( (ID, M, r) \). If it was, the previously defined value is returned. Otherwise, the challenger \( B \) randomly chooses \( h_{2i} \in R_n \), inserts \( (ID, M, r, h_{2i}) \) into \( L_2 \), and returns \( h_{2i} \) to \( A \).
Oracle $O_{H_1}(s, e)$: For an $O_{H_1}(s, e)$ query, $B$ first checks if the value of $O_{H_1}$ was previously defined for the input $(s, e)$. If it was, the previously defined value is returned. Otherwise, $B$ randomly chooses $h_0 \leftarrow \{0, 1\}^{k^*}$, inserts $(s, e, h_0)$ into $L_3$, and returns $h_3$ to $A$.

- **KeyExtract queries**: When $A$ asks for a user's secret key for $ID_i$, $B$ does the following. Assume that $A$ has made a $H_1$ query on $ID_i$. If $(ID_i, R_i, B_i, T_i)$ is contained in list $L_1$, then $B$ retrieves it. By construction $B_i = AR_i^{-1}$, and $T_i$ is a short basis for $\Lambda_i(R_i)$. Return $T_i$ as the secret key to the adversary. The restriction is that $ID_i \neq ID_j$ for $j = 1, \ldots, l$.

- **Signcrypt queries**: On receiving this query $(M, ID_i, L_i)$, where $ID_i$ is a sender, $L_i = \{ID_{R_1}, ID_{R_2}, \ldots, ID_{R_l}\}$ is the list of intended receivers. If $O_{H_1}(ID_i)$ is not contained in the set $\{R_1', R_2', \ldots, R_l'\}$, then $B$ checks if it is contained in list $L_1$, and gets its secret key, generates the signcryption in a normal way and returns it to $B$. If it is not contained in $L_1$, then $B$ runs SampleRwithBasis$(A)$ again to generate its secret key, and regenerates signcryption as before, and adds $(ID_i, R_i, B_i, T_i)$ into $L_1$. If $O_{H_1}(ID_i) = R_i'$ is just contained in the set $\{R_1', R_2', \ldots, R_l'\}$, $B$ generates the signcryption as follows.

1. $B$ randomly chooses a new $h_2, \in Z_q^n$, and solves the linear equation $A(R_i')^{-1}x = h_2$, gets it's a particular solution $e_i$ such that $0 < \|e_i\| \leq \delta \sqrt{m}$, then $B$ chooses a random string $r \in \{0, 1\}^l$, and adds the tuple $(ID_i, M, r, h_2)$ into $L_2$ list.

2. $B$ chooses distinct $s_1, s_2, \ldots, s_l \in Z_q^n$ randomly.

3. For all receivers $L_i = \{ID_{R_1}, ID_{R_2}, \ldots, ID_{R_l}\}$ $(i = 1, \ldots, l)$, $B$ computes $U_i = (c_i, d_i)$, where $c_i = O_{H_1}(s_i, e_i) \Theta (M \mid ID_i)$, $d_i = F_{ID_i}^T s_i + e_i$ and $F_{ID_i} = A(O_{H_1}(ID_i))^{-1}$.

The signcryption is $C = (r, U_1, \ldots, U_l, L_i)$.

Since $B$ does not know the secret key of $ID_i$, $C$ is generated by using a different method by $B$, it indeed generates a valid signcryption on message $M$ from $ID_i$ to the set of receivers $L_i$.

- **Unsigncrypt queries**: On receiving this query $(C, ID_i, ID_R)$, where the signcryption $C = (r, U_1, \ldots, U_l, L_i), U_i = (c_i, d_i)$. If $ID_R$ is not contained in $L$, then $B$ computes the secret key according to KeyExtract query on $ID_R$, unsigncrypts $C$ in a normal way and returns the corresponding message to $A$. Otherwise, $B$ proceeds as follows. It looks into the list $L_3$ for tuple $(s, e, h_3)$ such that $d_i = F_{ID_R}^T s_i + e_i$ and for the corresponding element $h_3, M_i \mid ID_i = c_i \oplus h_3$, such that there exists an entry $(ID_i, M_i, r, h_2)$ into $L_2$ list. If no such tuple is found, the challenger $B$ aborts. Otherwise,
\(B\) further checks if \(F_{ih_3} e_i = h_{2j}\), where \(F_{ih_3} = A(O_{h_3}(ID_r))^{-1}\). If this equation holds, \(M_i\) is returned to \(A\). Otherwise, \(B\) aborts.

- **Challenge:** After various queries in phase 1, \(A\) eventually outputs two plaintext messages \(M_0, M_1\) of equal length together with a sender’s identity \(ID_s\) on which he wishes to be challenged. \(A\) now waits for a challenged signcryption built under the receivers’ identities \(L = (ID_{r1}, \ldots, ID_{rT})\). Now, the challenger \(B\) ignores \(M_0\) and \(M_1\), selects \(r^* \leftarrow \{0, 1\}^t\), \(e^*_w \leftarrow \{0, 1\}^{k_0 + k_1}\) randomly and sets \(d^*_w = \overline{d^*_w}\). While others \((c^*_i, d^*_i) (i \neq w)\) are computed as before, then the challenger \(B\) sends the challenged signcryption \(C^* = (r^*, (c^*_1, d^*_1), \ldots, (c^*_w, d^*_w), \ldots, (c^*_l, d^*_l), L)\) to \(A\).

- **Phase 2:** \(A\) can perform queries as in phase 1. However, \(A\) cannot make the unsigncrypt query on the challenge signcryption \(C^*\). Once \(A\) obtains \(C^*\), it randomly chooses the \(i^{th}\) component of \(C^*\). For our purpose, here we assume \(w = l\). Thus, \(A\) gets \(C_w^* = (r^*, (c^*_w, d^*_w))\). Since \(A\) cannot make sure whether \(C_w^*\) is a valid signcryption to \(ID_r^*_w\) unless it has queried for oracle \(O_{h_3}(s^*_w, e^*_w)\). Thus, the solution of the LWE problem would be inserted in \(L\) just at that moment and it does not matter whether the simulation of \(A\)’s view is no longer perfect.

- **Guess:** At the end of the simulation, \(A\) eventually outputs a bits \(b^*\) for which, \(A\) believes that the challenged signcryption \(C_w^*\) is the signcryption of \(M_b^*\) from the sender to receiver \(ID_r^*_w\). While \(B\) ignores \(A\)’s output result, \(B\) just scans the list \(L\) for tuples of the form \((s^*_w, e^*_w, h^*_3)\). For each of them, \(B\) needs to check whether \(d^*_w = A^*_w s^*_w + e^*_w\) holds. If it holds, \(B\) outputs \(s^*_w\) as a solution of the LWE problem. Otherwise, \(B\) aborts.

Now we calculate \(B\)’s probability of winning the game. We first set \(E\) to be the event that \(A\) queries the oracle \(O_{h_3}(s^*_w, e^*_w)\) during the simulation. Following the above discussion, we get that as long as the simulation of the attack’s environment is perfect, the probability for \(E\) to happen is the same as in a real attack. In a real attack, we get that:

\[
P_b = \frac{P[b = b^*]}{P[b = b^*]} 
\]

Thus we obtain that

\[
e = 2P[b = b^*] - 1 \leq P(E)
\]

We consider the probability that the simulation is not perfect. The only case where it happens is when a valid signcryption is discarded in an unsigncrypt query. For each tuple \((s^*_w, e^*_w, h^*_3)\) in \(L\), since there is only one \(h^*_3\) in the range of oracles \(O_{h_3}\) providing a valid signcryption, thus the probability to discard a valid signcryption is not greater than \(q_{h_3}/2^*\). Actually, since the adversary \(A\) randomly chooses \(C_w^* = (r^*, (c^*_w, d^*_w))\) from \(C^*\) as the signcryption of the receiver \(ID_r^*_w\), and it can get \(C_w^*\) from \(C^*\) with the probability \(1/l\), thus we conclude that the probability of solving
the instance LWE problem is $v' \geq \frac{1}{l}(v - q_Hq_\nu)$. This is a non-negligible probability, which contradicts to the Lemma 1. Therefore, our MIBSC scheme can guarantee confidentiality which is indistinguishability of ciphertexts under selective multiple identities, chosen ciphertext attacks.

**Theorem 2:** In the random oracle model, if there is a polynomial-time adversary $F$ that can break the SUF-sMIBSC-CMA security of our MIBSC scheme with non-negligible probability $\varepsilon$ when performing $q_H$ queries to oracles $O_{H_i}$ ($i = 1, 2, 3$), $q_{sc}$ signcrypt queries, then there is a challenger $B$ that can also solve the SIS problem with a non-negligible probability.

**Proof:** We prove it also by contradiction, if an adversary $F$ can forge a signcryption in our scheme, then one can construct a challenger $B$ to solve the SIS problem. $B$ first receives an instance of SIS problem $(A'_n \in Z_q^{n \times m}, q, n, \delta)$. His goal is to receive a vector $v$ such that $0 < |v'\| \leq 2\delta\sqrt{m}$ and $A'_nv' = 0 \mod q$. Suppose there exists a SUF-sMIBSC-CMA adversary $F$ for our MIBSC scheme. We show that $B$ can use the adversary $F$ to solve the SIS problem. $B$ also sets the random oracles $H_O$, $H_O$ ($i = 1, 2, 3$), $H_O$. To maintain consistency, $B$ also maintains three lists $L_1$, $L_2$ and $L_3$ respectively.

We assume that $F$ will query the oracle $H_O$ before $ID$ is used in any KeyExtraction, signcrypt and unsigncrypt queries.

**Setup:** This phase is the same to Theorem 1. $B$ also chooses a random $w \in [l]$, then sets $A = A'_nR_n^*$ and sends it to the adversary $F$. $F$ outputs multiple target identities, denoted by $L = \{ID_1^*, \ldots, ID_3^*\}$. $F$ then chooses randomly $w \in [l]$, here $F$ claims to be able to forge the signcryption of the sender whose identity is $ID_i^*$.

**Attack:** During this phase, $F$ performs a polynomial number of $O_{H_i}$, $O_{H_1}$, $O_{H_2}$, signcrypt queries and KeyExtract queries adaptively. Considering that the most general known attack to signature scheme is chosen message attacks, in this game we assume the forger is not allowed to make the unsigncrypt queries. Here, $O_{H_i}$, $O_{H_1}$ and KeyExtract queries are described as the same in Theorem 1. The oracle $O_{H_2}$ and signcrypt queries are described below:

- **Oracle $O_{H_2}(ID, [M_i || r_i])$:** When querying $O_{H_2}(ID, [M_i || r_i])$, firstly, the algorithm $B$ checks if the value of $O_{H_2}$ was previously defined for the input $(ID, M_i, r_i)$. If it was, the previously defined value is returned. Otherwise, $B$ runs $e_i \leftarrow SampleDom(V^*)$, inserts $(ID_1, M_i, r_i, e_i, F_{ID_1}e_i)$ into $L_2$ list, and returns $F_{ID_1}e_i$ to $B$ as the hash value.

- **Signcrypt queries:** On receiving this query $(M, ID_i, L_1)$, where $L_1 = \{ID_{i_1}, \ldots, ID_{i_1}\}$ is the set of intended receivers. If $ID_i \neq ID_{i_1}^*$, the challenger $B$ computes $SK_{ID_1}$ according to KeyExtract query on $ID_1$, generates the signcryption in a normal way and returns it. If $ID_i = ID_{i_1}^*$, $B$ generates the signcryption as follows:
1. For the message $M \in \{0, 1\}^h$, $B$ firstly chooses a random string $r \in \{0, 1\}^k$, then chooses a random $e \leftarrow D_{2^{\kappa}}$ by running the algorithm $\text{SampleDom}(1^n)$, $B$ inserts $(ID_r, M, r, e, F_{ID_r} e)$ into $L_2$ list, where $F_{ID_r} = A(R_r)^{-1}$.

2. $B$ chooses $s_1, \ldots, s_l \in Z_q^*$ randomly, and computes
   \[ c_i = \mathcal{O}_{ID_i}(s_i, e) \oplus (M \parallel ID_r), \quad d_i = F_{ID_r} s_i + e, \quad F_{ID_r} = A(\mathcal{O}_{ID_r}(ID_r))^{-1}. \]

Then the signcryption $(r, (c_1, d_1), \ldots, (c_l, d_l), L_2)$ is returned to $\mathcal{F}$.

**Forgery:** The adversary $\mathcal{F}$ eventually outputs $l$ arbitrary receivers’ key pairs $(ID_{R_1}, SK_{ID_{R_1}}), \ldots, (ID_{R_l}, SK_{ID_{R_l}})$, where each $ID_{R_j} \neq ID_{R_i}$ for all $j \in \{1, 2, \ldots, l\}$, and produces a signcryption $C^* = (r^*, (c_1^*, d_1^*), \ldots, (c_l^*, d_l^*), L_2)$ under the sender’s public key $A_r^*$ and the receivers’ public keys $F_{ID_{R_j}} = A(\mathcal{O}_{ID_{R_j}}(ID_{R_j}))^{-1}, i = 1, \ldots, l$. $\mathcal{F}$ wins the game if the result of the unsigncrypt query on the signcryption $C^*$ under the secret key of any of the receivers is a valid message-signature pair $(M^*, e^*)$ such that no signcrypt query involved $M^*$, $ID_{R_i}$ and some receiver $ID_{R_j}$, thus results in a signcryption $C_j^* = (r^*, (c_j^*, d_j^*))$ whose decryption under the secret key $SK_{ID_{R_j}}$ of the receiver $ID_{R_j}$ is a valid forgery signcryption. $B$ performs as follows:

1. Compute $s_i^*$ and $e^*$ from $d_i^*$ using the receiver’s secret key $SK_{ID_{R_j}}$.
2. Compute $M^* = \mathcal{O}_{ID_j}(s_i^*, e^*)$, and recover $M^*$ from the first $k_1$ bit of $M^*$.
3. Output $(r^*, e^*)$ as a forged signature on message $M^*$ for the sender $ID_{R_j}^*$.

For the $\mathcal{O}_{ID_j}$ oracle, before outputing forgery $(r^*, e^*)$ of $M^*$ for the sender $ID_{R_j}^*$, the adversary $\mathcal{F}$ queries $\mathcal{O}_{ID_j}$ on $(ID_{R_j}^*, M^*, r^*)$, $B$ evaluates $e' \leftarrow \text{SampleDom}(1^n)$, inserts $(ID_{R_j}^*, M^*, r^*, e', A_r^* e')$ into $L_2$ list, and returns $A_r^* e'$ to $\mathcal{F}$ as the hash value. As claimed above, $e'$ is a valid forgery signature on $M^*$ for $ID_{R_j}^*$, we have $0 < ||e'|| \leq \delta \sqrt{m}$ and $A_r^* e' = \mathcal{O}_{ID_j}(ID_{R_j}^* \parallel M^* \parallel r^*) = A_r^* e'$. It remains to check $e^* \neq e'$ that they form a collision in $A_r^*$.

Now, we fix the value $\mathcal{O}_{ID_j}(ID_{R_j}^* \parallel M^* \parallel r^*)$. By the preimage min-entropy property of the hash family, the min-entropy of $e'$ given $A_r^* e'$ (and the rest of the view $\mathcal{F}$, which is independent of $e'$) is $\omega(\log n)$. Thus, $e^* = e'$ with negligible probability $2^{-\omega(\log n)}$.

Therefore, we get the conclusion that $B$ outputs $v' = e^* - e'$ such that $A_r^* v' = 0$ with probability negligibly close to $\varepsilon$, thus it solves the SIS problem with non-negligible probability, which leads to a contradiction.

Therefore, our MIBSC scheme can guarantee the unforgeability which can provide signcryption non-repudiation.
4.2 Comparison with existing schemes

Now we give the comparison of computational efficiency of our MIBSC scheme with existing classic multi-receiver signcryption schemes in Table 1, and we also give the comparison of communication overhead and hard problems in Table 2. Since the computation efficiency including signcryption cost and unsigncryption cost, and the communication overhead of signcrypted ciphertext are important factors affecting the performance of MIBSC, we present the comparison with respect to them. We consider the costly operations, which include modular exponentiation operation (Ex), modular multiplication operation (Mu), multiplication operation between a matrix and a vector (mu), exclusive or operation (XOR), pairing computation (Pa), hash function operation (H), inverse operation (In), and the preimage sampleable function operation ($T_{sample}$). Here, we don’t consider the computation of the hash value of direct identities which can be precomputed for frequently communicating parties. Moreover, the number of bits in the representation of an element in $G_1$ and $Z_q$ are denoted by $ℓ_{G_1}$, $ℓ_q$ respectively. The number of bits in the representation of a message, an identity are denoted by $ℓ_M$ and $ℓ_ID$ respectively, and the number of bits in the representation of $L$ is denoted by $ℓ_L$.

**Table 1** Comparison of computational efficiency

<table>
<thead>
<tr>
<th>Schemes</th>
<th>Signcryption cost</th>
<th>Unsigncryption cost</th>
<th>Total cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>Duan and Cao</td>
<td>$(l + 2)Mu + Pa$</td>
<td>$3Mu + 4Pa + Ex + 3H$</td>
<td>$(l + 5)Mu + 5Pa + 2XOR + 2In$</td>
</tr>
<tr>
<td>(2006)</td>
<td>+ $(l + 4)Ex + 2H + XOR$</td>
<td>+ $XOR + 2In$</td>
<td>+ $(l + 5)Ex + 5H + 2XOR + 2In$</td>
</tr>
<tr>
<td>Li et al.</td>
<td>$(2l + 1)Mu + Ex + 2H$</td>
<td>$2Mu + 2Pa + Ex + 2H$</td>
<td>$(2l + 3)Mu + 2Pa + 2Ex + 4H + 2XOR$</td>
</tr>
<tr>
<td>(2009)</td>
<td>+ $XOR$</td>
<td>+ $XOR$</td>
<td>+ $4H + 2XOR$</td>
</tr>
<tr>
<td>Pang et al.</td>
<td>$5Mu + Pa + (l + 2)H$</td>
<td>$Mu + 2Pa + 3H + XOR$</td>
<td>$6Mu + 3Pa + (l + 5)H + 2XOR$</td>
</tr>
<tr>
<td>(2015)</td>
<td>+ $XOR$</td>
<td>+ $XOR$</td>
<td></td>
</tr>
<tr>
<td>Ming et al.</td>
<td>$(l + 1)Mu + 1Pa$</td>
<td>$5Mu + 6Pa + 2H$</td>
<td>$(l + 6)Mu + 7Pa + 2XOR + In$</td>
</tr>
<tr>
<td>(2011)</td>
<td>+ $(l + 2)Ex + 2H + XOR$</td>
<td>+ $XOR + In$</td>
<td>+ $(l + 2)Ex + 4H + 2XOR + In$</td>
</tr>
<tr>
<td>Li et al.</td>
<td>$(l + 4)Mu + 2Pa + Ex + 2H$</td>
<td>$3Mu + 5Pa + Ex + 3H$</td>
<td>$(l + 7)Mu + 7Pa + 2Ex + 5H + 2XOR + In$</td>
</tr>
<tr>
<td>(2016)</td>
<td>+ $2H + XOR$</td>
<td>+ $XOR + In$</td>
<td>+ $5H + 2XOR + In$</td>
</tr>
<tr>
<td>Ours</td>
<td>$T_{sample} + limu + (l + 1)H$</td>
<td>$mu + H + XOR + In$</td>
<td>$T_{sample} + (l + 2)mu + (l + 2)H + (l + 1)XOR$</td>
</tr>
</tbody>
</table>

From Tables 1 and 2, we can see that although our MIBSC scheme has greater communication overhead, there is no modular exponentiation operation, modular multiplication operation, pairing computation in our MIBSC scheme. Since the pairing computation and modular exponentiation operation are much more time-consuming, especially in the unsigncryption process, our scheme is absolute to be faster than the existing schemes. Moreover, our MIBSC scheme is based on the hardness of lattice assumption, thus it will be more advantage in post-quantum cryptographic environments. Therefore, considering the security and the performance efficiency, our scheme is much better and practical than existing schemes.
Table 2: Comparison of communication overhead and hard problems

<table>
<thead>
<tr>
<th>Schemes</th>
<th>Communication overhead</th>
<th>Hard problems</th>
</tr>
</thead>
<tbody>
<tr>
<td>Duan and Cao (2006)</td>
<td>$(l + 3)\ell_a + \ell_m + \ell_ID + \ell_L$</td>
<td>BDH/CDH</td>
</tr>
<tr>
<td>Li et al. (2009)</td>
<td>$(l + 1)\ell_a + \ell_m + \ell_L$</td>
<td>$q$ – BDHI/$q$ – SDH</td>
</tr>
<tr>
<td>Pang et al. (2015)</td>
<td>$3\ell_a + \ell_m + \ell_ID + \ell_L + \ell_q$</td>
<td>Gap – BDH/CDH</td>
</tr>
<tr>
<td>Ming et al. (2011)</td>
<td>$(l + 3)\ell_a + \ell_m + \ell_L$</td>
<td>DBDH/CDH</td>
</tr>
<tr>
<td>Li et al. (2016)</td>
<td>$(l + 2)\ell_a + \ell_q + \ell_m + \ell_L$</td>
<td>BDH/CDH</td>
</tr>
<tr>
<td>Ours</td>
<td>$k + k(\ell_m + \ell_ID + m\ell_q)$</td>
<td>LWE/SIS</td>
</tr>
</tbody>
</table>

5 Conclusions

In this paper, we have proposed an efficient multi-receiver identity-based signcryption from hardness of lattice assumptions in the random oracle model. We have also proved our scheme is IND-sMIBSC-CCA secure under the LWE assumption and is SUF-sMIBSC-CMA secure under the SIS assumption. Meanwhile, compared with the existing schemes, our scheme can resist quantum attacks and performs faster. The extension of our MIBSC scheme to an MIBSC scheme in the standard model will be our future work.

Acknowledgements

This work is supported by the National Natural Science Foundation of China (No.61370203), Scientific Research Starting Project of SWPU (No. 2017QHZ023) and China Postdoctoral Science Foundation Funded Project (No. 2017M623008).

References


Efficient multi-receiver identity-based signcryption from lattice assumption


