Hindawi Publishing Corporation
International Journal of Distributed Sensor Networks
Volume 2013, Article ID 818917, 10 pages
http://dx.doi.org/10.1155/2013/818917

Research Article

Noncommutative Lightweight Signcryption for Wireless Sensor Networks

Lize Gu,[1] Yun Pan,[2] Mianxiong Dong,[3] and Kaoru Ota[4]

[1] *Information Security Center, State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China*
[2] *School of Computer Science, Communication University of China, Beijing 100024, China*
[3] *School of Computer Science and Engineering, The University of Aizu, Aizu Wakamatsu 965-8580, Japan*
[4] *Department of Information and Electronic Engineering, Muroran Institute of Technology, Muroran 050-8585, Japan*

Correspondence should be addressed to Yun Pan; pany@cuc.edu.cn

Received 3 January 2013; Accepted 6 February 2013

Academic Editor: Anfeng Liu

Key management techniques for secure wireless-sensor-networks-based applications must minimally incorporate confidentiality, authenticity, integrity, scalability, and flexibility. Signcryption is the proper primitive to do this. However, existing signcryption schemes are heavyweight and not suitable for resource-limited sensors. In this paper, we at first propose a braid-based signcryption scheme and then develop a key establishment protocol for wireless sensor networks. From the complexity view, our proposal is 2^{15} times faster than RSA-based ones. As far as we know, our proposal is the first signcryption scheme based on noncommutative algebraic structures.

1. Introduction

Wireless sensor networks (WSNs) consist of a large number of micro, low-cost, low-power, and spatially distributed autonomous devices using sensors to cooperatively monitor physical or environmental conditions [1, 2]. WSNs are often deployed in potentially adverse or even hostile environment so that there are concerns on security issues therein. To protect the confidentiality and privacy of WSN-oriented applications, the traditional symmetric (i.e., private-key), even lightweight, cryptography is often used. A well-known drawback to do this is that the symmetric cryptography is not as flexible as the asymmetric (i.e., public-key) cryptography. The main obstacle of using public-key cryptography in WSNs is that with limited memory, computing and communication capacity, and power supply, sensor nodes cannot employ sophisticated cryptographic operations such as modular exponentiation and pairing computation. Therefore, it is interesting to probe new efficient and lightweight implementations on some wellknown public-key cryptographic primitives, such as what has been done in TinyECC [3] and in MicroECC [4]. No matter which type cryptography is adopted, key establishment is one of the utmost concerns. At least, key establishment techniques for a secure WSN-based application must minimally incorporate confidentiality, authenticity, integrity, scalability, and flexibility [5].

Signcryption, now an international standard for data protection (ISO/IEC 29150, Dec 2011), was invented in 1996 and first disclosed to the public at CRYPTO 1997 [6, 7]. It is a data security technology by which confidentiality is protected and authenticity is achieved seamlessly at the same time. This will also allow smaller devices, such as smartphones and PDAs, 3G and 4G mobile communications, as well as emerging technologies, such as radio frequency identifiers (RFIDs) and wireless sensor networks, to perform high-level security functions. And, by performing these two functions simultaneously, we can save resources, be it an individual's time or be it energy, as it will take less time to perform the task. Therefore, signcryption is very suitable for key management in wireless sensor networks and other resource-constrained environments.

Since the invention of the primitive of signcryption, various constructions were proposed and most of them are based on three kinds of cryptographic assumptions. The first

category assumes that the integer factoring problem (IFP) is intractable, such as the constructions in [8, 9]. The second category assumes that the discrete logarithm problem (DLP) over finite fields or elliptic curves (i.e., ECDLP) is intractable, such as the constructions in [10, 11]. In this category, some constructions further utilize the bilinear pairing to enhance the functionalities and performance, such as the constructions in [12, 13]. The third category is based on some lattice hard problems [14, 15]. Up to now, the last category attracts a lot of attention since the so-called quantum attack-resistant property. However, these existing lattice-based signcryptions have disadvantage in key sizes. Thus, it is interesting to probe new construction of signcryption based on other cryptographic primitives than IFP- and DLP-related ones and meanwhile keeping the potential of quantum attack resistance.

Under this background, some noncommutative groups have attracted the attention. One of the most popular groups in this category is the braid group. At CRYPTO 2000, Ko et al. [16] proposed the first fully fledged braid-based cryptosystem. In braid-based cryptographic schemes [16–24], the conjugacy search problem (CSP) (i.e., given two braids a and xax^{-1}, output the braid x) and its variants play a core role. Although many heuristic attacks, such as length-based attacks linear representation attacks, have obtained remarkable success in attacking braid-based cryptosystems and lowered the initial enthusiasm on this subject, there is no deterministic polynomial algorithms that can solve the CSP problem over braid groups [25] till now. On one hand, Birman et al. launched a project, referred to as BGGM project, to find polynomial algorithms for solving the CSP problem over Garside groups, including braid groups [26–28]. The BGGM project might be the strongest efforts known for solving the CSP problem over braid groups in polynomial-time (with respect to the input size). Up to now, the BGGM project has already made a great progress; except for rigid pseudo-Anosolov braids, the CSP instances over other braids can be solved in polynomial time [28]. On the other hand, some researchers still keep on finding hard instances of the CSP problem in braid groups. For examples, in 2007, Ko et al. [29] proposed some ideas on generating hard instances for braid cryptography, and in 2010, Prasolov [30] constructed some small braids with large ultra summit set (USS). Prasolov's result represents a frustration toward the BGGM project, but an encouragement toward the intractability assumption of the CSP problem over braid groups. According to [31], if p and s are random braids, then the length of sps^{-1} is, with a high probability, about the length of p plus the double of the length of s. This is the reason why the length-based attacks work. This also suggests that one can defeat the length-based attacks by requiring that the length of sps^{-1} is closer to the length of p. This in turn requires that p should lie in its super summit set (SSS) [31]. We know that USS $\subset$ SSS. Therefore, if we can work with the braids suggested by Prasolov, then we reach the point to instantiate our proposal with braid groups in a secure manner.

Another promising observation coming from [23] is that braid operations can be implemented with a complexity level of about 2^{15} bit operations, while the complexity level of the exponentiation over 1024 bit RSA modular is about 2^{30} bit operations. This suggests that braid-based cryptosystems admit ultra efficient, even lightweight, implementations.

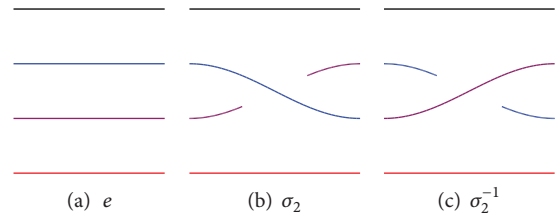

Figure 1: Geometrical illustration on identity and Artins generators [23].

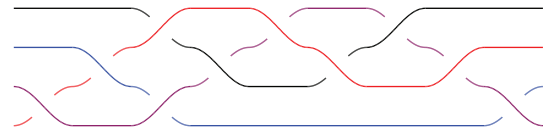

Figure 2: An example of geometric braids [23].

The main motivation of this paper covers two aspects: the first is to design a lightweight signcryption scheme based on noncommutative groups assuming that the CSP problem over the underlying groups are intractable, and the second is to construct efficient key management protocols for wireless sensor networks.

The rest contents are organized as follows. In Section 2, we at first give a simple introduction to the braid group, and then introduce the left self-distributive system and its properties. A building block—braid-based signcryption scheme is proposed in Section 3, and the full description of the key management protocol for wireless sensor networks is developed in Section 4. Performance evaluation and comparisons, including security level analysis, are given in Section 5, respectively. Concluding remarks are given in Section 6.

2. Preliminaries

2.1. Braid Group and Related Cryptographic Problems. The n-braid group B_n is presented by the Artin generators $\sigma_1, \ldots, \sigma_{n-1}$ and relations $\sigma_i\sigma_j = \sigma_j\sigma_i$ for $|i-j| > 1$ and $\sigma_i\sigma_j\sigma_i = \sigma_j\sigma_i\sigma_j$ for $|i-j| = 1$ $(1 \le i, j \le n-1)$. Braid groups also admit a very intuitively geometrical illustration: the identity of braid groups, that is, the empty braid e, and the Artin generators (e.g., $\sigma_2^{\pm 1}$ in B_4) as shown in Figure 1 [23].

Geometrically, the product of two braids is the braid obtained by merging the tail of the first braid with the head of the second braid. For example, Figure 2 shows the braid $\sigma_1\sigma_2\sigma_1^{-1}\sigma_3^{-1}\sigma_2\sigma_3\sigma_2\sigma_3^{-1}\sigma_2^{-1}\sigma_1$ [23].

There is a natural automorphism from B_2 to the integer additive group $\mathbb{Z}$ and this means that B_2 is infinite and commutative. But for $n \ge 3$, the braid group B_n is infinite and noncommutative. In addition, for each m $(\le n)$, the identity mapping on $\{\sigma_1, \ldots, \sigma_{m-1}\}$ naturally induces an embedding of B_m into B_n [23].

For arbitrary two braids $x, y \in B_n$, we say they are conjugate, written as $x \sim y$, if $y = a^{-1}xa$ for some $a \in B_n$. Here a or a^{-1} is called a conjugator. The conjugacy deciding problem (CDP) is to determine whether $x \sim y$ for a given instance $(x, y) \in B_n^2$, while the conjugator searching problem (CSP) is to find a braid $z \in B_n$ such that $y = z^{-1}xz$ for a given instance $(x, y) \in B_n^2$ with $x \sim y$. At present, we know that both CDP and CSP over braid groups are solvable; that is, there is a deterministic algorithm that stops after finite steps, not necessarily polynomially bounded, and outputs an accurate solution. However, it seems that both of them are, at least in worst cases, intractable; that is, there is no probabilistic polynomial time algorithms that output an accurate solution with nonnegligible probability (with respect to the length of description of the input instances) [20, 21, 23].

In sequel, we use x^a to denote the conjugate braid $a^{-1}xa$ when $a \in B_n$. Meanwhile, we also use x^t to denote the multiplication braid $\underbrace{x \cdots x}_{t \text{ times}}$ when $t \in \mathbb{N}$.

2.2. Conjugacy-Based Left Self-Distributive Systems. Under the intractability assumption of the conjugator search problems over certain noncommutative semigroups, Wang et al. [24] proposed several public-key cryptosystems based on conjugacy-based left self-distributive systems. The notations and related constructions are helpful for developing our main proposal in this paper. Therefore, let us recall the definition of the left self-distributive system that was firstly postulated by Dehornoy [32].

Definition 1 (left self-distributive system LD [32]). Suppose that S is a nonempty set, $F : S \times S \to S$ is a well-defined function and let us denote $F(a, b)$ by $F_a(b)$. If the following rewritten formula holds

$$F_r\left(F_s\left(p\right)\right) = F_{F_r(s)}\left(F_r\left(p\right)\right), \quad \left(\forall p, r, s \in S\right), \tag{1}$$

then, we call $F_\cdot(\cdot)$ a left self-distributive system, abbreviated as LD system.

The terminology "left self-distributive" arises from the following analogical observation: if we consider $F_r(s)$ as a binary operation $r * s$, then the formula (1) becomes

$$r * \left(s * p\right) = \left(r * s\right) * \left(r * p\right); \tag{2}$$

that is, the operation "$*$" is left self-distributive with respect to itself [32].

One can define the following LD system, named as Conj-LD system, which means an abbreviation of left self-distributive system defined by conjugate operations.

Definition 2 (Conj-LD system [24]). Let G be a noncommutative semigroup and $G^{-1} \subset G$ the set of all invertible elements. The binary function F given by the following conjugate operation:

$$F : G^{-1} \times G \longrightarrow G, \qquad (a, b) \longmapsto a^{-1}ba \triangleq b^a \tag{3}$$

is an LD system, abbreviated as Conj-LD.

It is easy to see that F caters to the rewritten formula (1). Thus, $F_a(b)$ is an LD system [24].

Table 1: Experiments for define CSP-DDH problem.

Experiment $\mathbf{Exp}_{F,\mathscr{A}}^{\text{csp-ddh-real}}$	Experiment $\mathbf{Exp}_{F,\mathscr{A}}^{\text{csp-ddh-rand}}$
$i \xleftarrow{\$} \mathbb{N}; X \leftarrow F_{a^i}(b);$	$i \xleftarrow{\$} \mathbb{N}; X \leftarrow F_{a^i}(b);$
$j \xleftarrow{\$} \mathbb{N}; Y \leftarrow F_{a^j}(b);$	$j \xleftarrow{\$} \mathbb{N}; Y \leftarrow F_{a^j}(b);$
$Z \leftarrow F_{a^{i+j}}(b);$	$\ell \xleftarrow{\$} \mathbb{N}; Z \leftarrow F_{a^\ell}(b);$
$b \leftarrow \mathscr{A}(X, Y, Z);$	$b \leftarrow \mathscr{A}(X, Y, Z);$
Return b.	Return b.

Proposition 3 (power law [24]). *Let F be a Conj-LD system defined over a noncommutative semigroup G. Suppose that $a \in G^{-1} \subset G$ and $b \in G$ are given and fixed. Then, for arbitrary three positive integers m, s, and t such that $m = s + t$, one has*

$$\begin{aligned} F_a\left(b^m\right) &= F_a\left(b^s\right) F_a\left(b^t\right) = F_a^m\left(b\right), \\ F_{a^m}\left(b\right) &= F_{a^s}\left(F_{a^t}\left(b\right)\right). \end{aligned} \tag{4}$$

Remark 4. By using the notation of $F_\cdot(\cdot)$, the intractability assumption of the CSP problem in G can be reformulated as follows: it is hard to retrieve a' from the given pair $(a, F_a(b))$ such that $F_a(b) = F_{a'}(b)$ (see more details in [24]).

Definition 5 (CSP-based decisional Diffie-Hellman: CSP-DDH [24]). Let F be a Conj-LD system defined over a noncommutative semigroup G and let $\mathscr{A}$ be an adversary. For arbitrary $a \in G^{-1}$ and $b \in G$, consider the following two experiments in a paralleled manner (see Table 1). Now define the advantage of $\mathscr{A}$ in violating the CSP-based decisional Diffie-Hellman assumption as

$$\begin{aligned} \mathbf{Adv}_{F,\mathscr{A}}^{\text{csp-ddh}} = \Big|&\Pr\left[\mathbf{Exp}_{F,\mathscr{A}}^{\text{csp-ddh-real}} = 1\right] \\ &- \Pr\left[\mathbf{Exp}_{F,\mathscr{A}}^{\text{csp-ddh-rand}} = 1\right]\Big|. \end{aligned} \tag{5}$$

Intuitively, the CSP-DDH assumption states that the distributions:

$$\begin{aligned} \mathscr{D}_1 &\triangleq \left(F_{a^i}\left(b\right), F_{a^j}\left(b\right), F_{a^{i+j}}\left(b\right)\right), \\ \mathscr{D}_2 &\triangleq \left(F_{a^i}\left(b\right), F_{a^j}\left(b\right), F_{a^\ell}\left(b\right)\right) \end{aligned} \tag{6}$$

are computationally indistinguishable when $i, j, \ell \in \mathbb{N}$ are drawn at random.

Remark 6. Intuitively, it is hard to solve the CSP-DDH problem without solving the CSP problem if G is modeled as a generic semigroup model. According to [33], we know that the discrete logarithm problem (DLP) over finite fields and the corresponding DDH problem are polynomially equivalent in a generic cyclic group. By an analogical manner, we speculate that the CSP problem and the CSP-DDH problem in a generic noncommutative semigroup are polynomially equivalent (see more details in [24]).

2.3. The Fujisaki-Okamoto Transformation [34, 35]. Without loss of generality, a public-key encryption scheme can be defined as a triple $\pi = (\mathscr{K}, \mathscr{E}, \mathscr{D})$, where

(i) $\mathscr{K}$ is the key generation algorithm that takes as input a system security parameter 1^k and outputs a public-/private-key pair (pk, sk). In general, this algorithm can be formulated as $(pk, sk) \leftarrow \mathscr{K}(1^k)$.

(ii) $\mathscr{E}$ is the encryption algorithm that takes as inputs the public-key pk and a message $m \in \mathscr{M}$ and outputs a ciphertext $c \in \mathscr{C}$, where $\mathscr{M}$ and $\mathscr{C}$ are message space and ciphertext space, respectively. In general, this algorithm can be formulated as $c \leftarrow \mathscr{E}_{pk}(m)$ or $c \leftarrow \mathscr{E}_{pk}(m; r)$ when it is necessary to specify the random salt r used in the encryption process.

(iii) $\mathscr{D}$ is the decryption algorithm that takes as inputs the secret key sk and a ciphertext $c \in \mathscr{C}$ and outputs a message $m \in \mathscr{M}$ or a symbol $\perp$, which indicates that c is invalid. In general, this algorithm can be formulated as $m/\perp \leftarrow \mathscr{D}_{sk}(c)$.

In general, as for public-key encryption, one-wayness against chosen plaintext attacks (OW-CPA) is the lowest security requirement, while indistinguishability against adaptively chosen ciphertext attacks (IND-CCA2) is the most desirable and the standard security requirement. Cryptographic practise shows that it is always easier to design an OW-CPA secure encryption scheme than to directly design an IND-CCA2 secure one. Thus, it is desirable to have a general method for transforming an OW-CPA secure encryption scheme to an IND-CCA2 secure one [35]. Fortunately, one of this methods was invented by Fujisaki and Okamoto [34] at PKC 1999.

Theorem 7 (FO transformation [34]). *Suppose H_1 and H_2 are two random oracles with required domains and ranges, respectively. Given a public-key encryption scheme*

$$\pi = (\mathscr{K}, \mathscr{E}, \mathscr{D}) \tag{7}$$

that achieves the security of one-wayness against chosen plaintext attacks (OW-CPA), one can get another public-key encryption scheme

$$\pi' = \left(\mathscr{K}', \mathscr{E}', \mathscr{D}'\right) \tag{8}$$

that achieves the security of indistinguishability against adaptively chosen ciphertext attacks (IND-CCA2), where

(1) *key generation algorithm $\mathscr{K}'$ is identical to $\mathscr{K}$;*

(2) *encryption algorithm is defined as*

$$\mathscr{E}'_{pk}(m) = \left(\mathscr{E}_{pk}(r), m \oplus H_1(r), H_2(m, r)\right), \tag{9}$$

where r is picked at random;

(3) *decryption algorithm $\mathscr{D}'_{sk}(c_1, c_2, c_3)$ performs the following steps:*

(a) $r' \leftarrow \mathscr{D}_{sk}(c_1)$;
(b) $m' \leftarrow c_2 \oplus H_1(r')$;
(c) *output m' if $c_3 = H_2(m', r')$ and $\perp$ otherwise.*

3. Building Block: Noncommutative Signcryption

Before describing our proposal for WSN key management, let us at first propose a signcryption scheme from noncommutative semigroups where the CSP-related assumptions hold. We will see later, when this scheme is instantiated by using braids, we obtain a very efficient signcryption scheme that is 2^{15} times faster than RSA-based signcryption (suppose that 1024 bit RSA modulus were used).

Suppose that G is a noncommutative semigroup so that the CSP problem and the CSP-DDH problem over G are intractable. Then, the public parameters of the proposed signcryption are given by a quintuple $\langle \mathfrak{D}, a, b, H_1, H_2 \rangle$, where

(i) $\mathfrak{D}$ is a description of G and $G^{-1} \subset G$. Without loss of generality, we assume the length of $\mathfrak{D}$ is bounded by $\mathcal{O}(\log |G|)$ for finite G. When G is infinite but admits a finite presentation, say $fp(G) = \langle X \mid R \rangle$, the length of $\mathfrak{D}$ is the sum of the length of X and the length of R. However, for braid group B_n, $\mathfrak{D}$ admits even efficient description since whenever the braid index n is given, the generator set $X = \{\sigma_1, \ldots, \sigma_{n-1}\}$ and the relation set $R = \{\sigma_i \sigma_j = \sigma_j \sigma_i : |i - j| > 1\} \cup \{\sigma_i \sigma_j \sigma_i = \sigma_j \sigma_i \sigma_j : |i - j| = 1\}$ $(1 \le i, j \le n - 1)$ is totally specified. That is, for braid group B_n, $\mathfrak{D} = n$;

(ii) $a \in G^{-1} \subset G$ and $b \in G$ are two fixed elements that are picked at random;

(iii) $H_1 : G \to G^2$ and $H_2 : G^2 \to G$ are two cryptographic hash functions that are modeled as random oracles.

Then, the proposed signcryption scheme consists of the following three algorithms:

(i) $\mathscr{KG}(1^k)$, key generation algorithm that takes as input the system security parameter 1^k, picks an integer $s \in \{0, 1\}^k$ at random calculates $x = b^{a^s} \in G$, and finally outputs (s, x) as the private-/public-key pair.

(ii) $\mathscr{SC}(s, y; m)$, signcryption algorithm that takes as inputs the sender's private-key $s \in \{0, 1\}^k$, the receiver's public-key $y \in G$, and the message $m \in G$, and performs the following steps:

(1) pick $t \in \{0, 1\}^k$ at random;
(2) compute

$$\begin{aligned} c_1 &= b^{a^t}, \\ h &= H_2(m, c_1), \\ \sigma &= \left(a^t\right)^{-1} a^s h c_1, \\ c_2 &= (m \,\|\, \sigma) \oplus H_1\left(y^{a^t}\right), \end{aligned} \tag{10}$$

where operator "$\oplus$" should be viewed as XOR operation over bit-strings that are encoding results of a pair in G^2;
(3) output (c_1, c_2).

Theorem 8. *The proposed signcryption is consistent.*

Proof. Suppose that the sender and the receiver performs honestly, and their inputs are well formed. That is, $x = b^{a^s}$ and $y = b^{a^r}$. Then, since

$$\begin{aligned} c_1^{a^r} &= \left(b^{a^t}\right)^{a^r} \\ &= b^{a^{t+r}} \\ &= \left(b^{a^r}\right)^{a^t} \\ &= y^{a^t}, \\ m' \,\|\, \sigma' &= c_2 \oplus H_1\left(c_1^{a^r}\right) \\ &= (m \,\|\, \sigma) \oplus H_1\left(y^{a^t}\right) \oplus H_1\left(y^{a^t}\right) \\ &= m \,\|\, \sigma, \\ h' &= H_2\left(m', c_1\right) \\ &= H_2\left(m, c_1\right) \\ &= h, \end{aligned} \tag{11}$$

we have

$$\begin{aligned} c_1^{\sigma'} &= \left(b^{a^t}\right)^{\sigma} \\ &= \left(b^{a^t}\right)^{(a^t)^{-1}a^s h c_1} \\ &= \left(b^{a^s}\right)^{h c_1} \\ &= x^{h c_1} \\ &= x^{h' c_1}. \end{aligned} \tag{12}$$

Then, $m' = m$ will be output correctly. □

Theorem 9. *Suppose that H_1 and H_2 are random oracles. The proposed signcryption is indistinguishable against adaptively chosen ciphertext attack (IND-CCA2) assuming that the CSP-DDH problem over the underlying noncommutative semigroup G is intractable.*

Proof. To apply the well-known Fujisaki-Okamoto transformation theorem [34], we at first need to define an IND-CPA secure encryption scheme $\pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ and then establish the security relationship between the proposed signcryption scheme and the enhanced encryption scheme π', that is, an FO transformation from π. This can be done by setting $\pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ as follows:

(i) $\mathcal{K}(1^k) := \mathcal{KG}(1^k)$. That is, the key generation algorithm remains unchanged.

(ii) The encryption algorithm $\mathcal{E}(y; m)$ that takes as inputs the receiver's public-key $y \in G$ and the intended message $m \in G$ and then performs the following steps:

(1) pick $t \in \{0,1\}^k$ at random;
(2) compute $c_1 = b^{a^t}$ and $c_2 = y^{a^t} m$;
(3) output (c_1, c_2).

(iii) The decryption algorithm $\mathcal{D}(r; c_1, c_2)$ that takes as inputs the receiver's private-key $r \in \{0,1\}^k$ and the ciphertext pair $(c_1, c_2) \in G^2$ and then outputs the intended message $m = c_2(c_1^{a^r})^{-1}$.

Apparently, this is just the ElGamal-like variant based on CSP-DDH assumption. According to Theorem 1 of [24], this is IND-CPA secure. Then, according to Theorem 7, the FO variant $\pi' = (\mathcal{K}', \mathcal{E}', \mathcal{D}')$ is IND-CCA2 secure when H_1 and H_2 are modeled as random oracles, where

(i) $\mathcal{K}'(1^k) := \mathcal{K}(1^k)$.

(ii) $\mathcal{E}'(y; m)$ performs the following steps:

(1) pick $u \in G$ at random;
(2) let $(c_1, c_2) \leftarrow \mathcal{E}(y; u)$;
(3) let $c_3 = m \oplus H_1(u)$ and $c_4 = H_2(m, u)$;
(4) output (c_1, c_2, c_3, c_4).

(iii) The decryption algorithm $\mathcal{D}(r; c_1, c_2, c_3, c_4)$ that takes as inputs the receiver's private-key $r \in \{0,1\}^k$ and the ciphertext qudruple (c_1, c_2, c_3, c_4), and then performs the following steps:

(1) let $u' \leftarrow \mathcal{D}(r; c_1, c_2)$;
(2) let $m' \leftarrow c_3 \oplus H_1(u')$;
(3) output m' if $c_4 = H_2(m', u')$ and $\perp$ otherwise.

Now, let us show that in the same random oracle models, if there is a polynomlai-time adversary $\mathcal{A}$ that can, with non-negligible probability, break the IND-CCA2 security of the proposed signcryption scheme, there is another polynomial-time adversary $\mathcal{B}$ that can, by controlling the response of the random oracles H_1 and H_2, break the IND-CCA2 security of π'. However, this is contrary to the fact that π' is IND-CCA2 secure. Therefore, $\mathcal{A}$'s advantage of breaking the proposed signcryption scheme must be negligible.

In fact, if $\mathcal{B}$ controls the response of the random oracles H_1 and H_2, then it can break the IND-CCA2 security of π' with nonnegligible probability. This is apparently, since B controls the response of H_2, whenever seeing a ciphertext (c_1, c_2, c_3, c_4), it can retrieve the message m and random salt u by looking up the response list of H_2 under the reasonable assumption that the probability for different pair (m', u') with same hash value with the pair (m, u) is negligible.

The left thing is to show that $\mathcal{B}$, without knowing the receiver's private-key $r \in \{0,1\}^k$, how to simulate the response on decryption queries for $\mathcal{A}$ in a perfect manner. Whenever

A invokes a decryption query by submitting a signcryption pair (c_1, c_2), $\mathscr{B}$ responds as follows:

(1) look up (h_2, m_i, c_1) in H_2-list. If there is no matched triple, $\mathscr{B}$ sends $\perp$ to $\mathscr{A}$ as the response;

(2) for each matched triple (h_2, m_i, c_1), $\mathscr{B}$ performs the following steps:

(a) for each (h_1, Y_i) in H_1-list, do the following steps:

(i) extract a possible σ_i according to the following formula:

$$c_2 = (m_i \| \sigma_i) \oplus h_1. \tag{13}$$

This can be done since $\mathscr{B}$ knows c_2, m_i and h_1 at this stage;

(ii) test whether the equality $c_1^{\sigma_i} = x^{h_2 c_1}$ holds? (recall that x is the verification key of the singer). If so, replies $\mathscr{A}$ with m_i and end of the response; otherwise, continue;

(3) if up to now, $\mathscr{B}$ has not output response to $\mathscr{A}$ yet, then $\mathscr{B}$ sends $\perp$ to $\mathscr{A}$ as the response.

Now, let us show that $\mathscr{B}$'s simulation is perfect. It is reasonable to assume that without accessing hash queries on H_1 and H_2, $\mathscr{A}$'s probability for submitting a valid signcryption pair (c_1, c_2) is negligible. Thus, whenever $\mathscr{A}$ invokes hash queries on H_1 and H_2 for forming a valid signcryption pair, related materials are recorded and $\mathscr{B}$ can retrieve them and finally send $\mathscr{A}$ a perfect response. □

Remark 10. Note that although the signature scheme embedded in the proposed signcryption scheme merely achieves unforgeable against no-message attacks, the resulted signcryption is existentially unforgeable against external adaptively chosen message attack. Here, external forgeries means that it is neither the singer, nor the intended receiver. We know that it is reasonable to exclude the signer from forgeries. Let us explain why we further exclude the intended receiver from the forgeries. In fact, the primitive of signcryption provides confidentiality of the message against all entities except the intended receiver and meanwhile it provides the authenticity of the sender (i.e., the signer) for the intended receiver. That is, the authenticity embedded in the signcryption primitive is unidirectional, instead of bidirectional. Therefore, it seems that there is no reason for an intended receiver to forge a signature on behalf of some signer and then encrypt the signature for himself/herself, except for planting false evidence against some senders. In other words, in our proposal, we assume that the receiver who possesses the corresponding private-key for performing designcryption is honest. Otherwise, an existentially unforgeable signature scheme, such as the noncommutative signature scheme in [36] should be embedded therein. For further consideration of the insider security and the outsider security of signcryptions, one can refer to [37, 38].

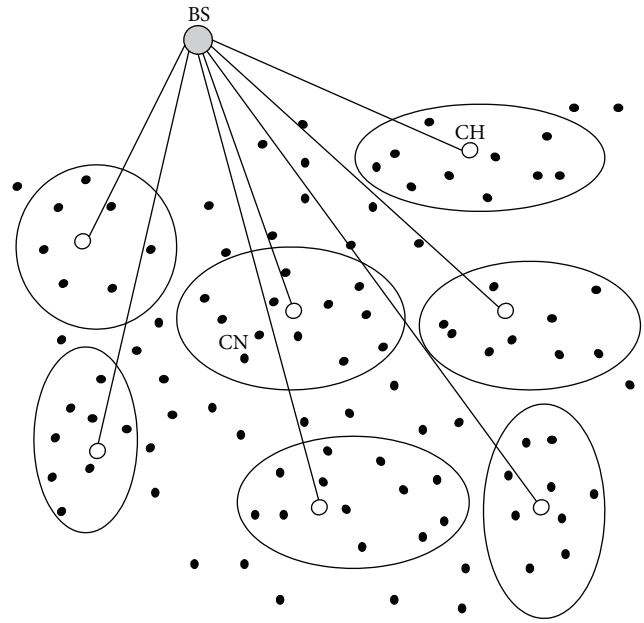

Figure 3: WSN Architecture.

4. Lightweight Implementation of Key Management Protocols for WSNs

In [5], Hagras et al. described an efficient key management scheme for WSNs based on elliptic curve signcryption. Our proposal follows their diagram. However, the main differences of our work lie in the following aspects:

(i) firstly, the signcryption algorithm used by Hagras et al. is abstract and essentially hybrid where a symmetric encryption algorithm is involved. However, we will give a detailed specification of each algorithm;

(ii) secondly, Hagras et al.'s proposal is based on commutative platforms, while as far as we known, our proposal is firstly based on noncommutative platforms.

Similar to [5], suppose that the network architecture is the standard clustered WSN architecture depicted in Figure 3. The proposed key management scheme supports three protocols: the first is used to generate private-/public-keys for each individual nodes, including base nodes, cluster headers, and cluster nodes; the second is essentially a signcryption scheme that is used by base node to send session keys to cluster heads; and the third is essential also a signcryption scheme that is used by cluster heads to send session keys to cluster nodes.

Let B_n be the braid group and $a, b \in B_n$. Suppose that $F(\cdot, \cdot)$ is the Conj-LD system defined over braid group B_n, while $H_1 : G \to G^2$ and $H_2 : G^2 \to G$ are two cryptographic hash functions. Our proposal consists of three protocols that are described in the following subsections.

4.1. Key Generation Protocol. This protocol is responsible for creating public-/private-key pairs for base nodes (BNs), cluster heads (CHs), and cluster nodes (CNs).

Step 1. Generate public-/private-key for based nodes.

$V_{\mathrm{BN}} \in \{0,1\}^k$: the private-key for the base node is a positive integer chosen uniformly at random.

$P_{\mathrm{BN}} \in B_n$: the corresponding public-key for the base node is calculated as $P_{\mathrm{BN}} = F(a^{V_{\mathrm{BN}}}, b)$.

Step 2. Generate public-/private-key for cluster heads.

$V_{\mathrm{CH}_j} \in \{0,1\}^k$: the private-key for the jth cluster head is a positive integer chosen uniformly at random.

$P_{\mathrm{CH}_j} \in B_n$: the corresponding public-key for the jth cluster head is calculated as $P_{\mathrm{CH}_j} = F(a^{V_{\mathrm{CH}_j}}, b)$.

Step 3. Generate public-/private-key for cluster nodes.

$V_{\mathrm{CN}_i} \in \{0,1\}^k$: the private-key for the ith cluster head is a positive integer chosen uniformly at random.

$P_{\mathrm{CN}_i} \in B_n$: the corresponding public-key for the ith cluster head is calculated as $P_{\mathrm{CN}_i} = F(a^{V_{\mathrm{CN}_i}}, b)$.

Step 4. Session key generation for base node and cluster heads.

(1) The base node creates the session key $K_{\mathrm{BN-CH}_j}$ which will be used for secure communication between the jth cluster head and the base node.

(2) The jth cluster head creates the session key $K_{\mathrm{CH}_j-\mathrm{CN}_i}$ which will be used for secure communication between the jth cluster head and the ith cluster node.

Without loss of generality, here we assume that $K_{\mathrm{BN-CH}_j}$ and $K_{\mathrm{CH}_j-\mathrm{CN}_i}$ are elements of G picked at random. (In fact, we can always employ an encoding algorithm to map elements of G into valid session keys.)

Remark 11. Note that in the last step, all session keys are newly generated by the base node and the cluster nodes, respectively. In fact, after the execution of Steps 1, 2 and 3, we know that the base node and the jth cluster head can calculate the shared session key $K_{\mathrm{BN-CH}_j} = F(a^{V_{\mathrm{BN}}+V_{\mathrm{CH}_j}}, b)$, and the jth cluster head and the ith cluster node can calculate the shared session key $K_{\mathrm{CH}_j-\mathrm{CN}_i} = F(a^{V_{\mathrm{CH}_j}+V_{\mathrm{CN}_i}}, b)$. However, it is not a good choice to use this kind of session keys since they are totally determined by long-term private-keys. Instead, we suggest to renew a session key instantly to guarantee its freshness.

4.2. BN-CHs *Signcryption.* The base node signcrypts the session key $K_{\mathrm{BN-CH}_j}$ using its private-key and sends the ciphertext (c_1, c_2) to the jth cluster head as follows:

(1) pick $t \in \{0,1\}^k$ at random;

(2) $c_1 = F(a^t, b)$;

(3) $h = H_2(K_{\mathrm{BN-CH}_j}, c_1)$;

(4) $\sigma = (a^t)^{-1} a^{V_{\mathrm{BN}}} h c_1$;

(5) $c_2 = (K_{\mathrm{BN-CH}_j} \| \sigma) \oplus H_1(F(a^t, P_{\mathrm{CH}_j}))$;

(6) send (c_1, c_2) to the jth cluster head.

Upon receiving the ciphertext (c_1, c_2) from the base node, the jth cluster head designcrypts the session key as follows:

(1) compute $K \| \sigma = c_2 \oplus H_1(F(a^{V_{\mathrm{CH}_j}}, c_1))$, $h = H_2(K, c_1)$;

(2) accept K if $F(\sigma, c_1) = F(hc_1, P_{\mathrm{BN}})$ and report "FAILURE" otherwise.

4.3. CH-CNs *Signcryption.* The jth cluster head signcrypts the session key $K_{\mathrm{CH}_j-\mathrm{CN}_i}$ using its private-key and sends the ciphertext (d_1, d_2) to the ith cluster node as follows:

(1) pick $s \in \{0,1\}^k$ at random.

(2) $d_1 = F(a^s, b)$.

(3) $g = H_2(K_{\mathrm{CH}_j-\mathrm{CN}_i}, d_1)$.

(4) $\sigma = (a^s)^{-1} a^{V_{\mathrm{CH}_j}} g d_1$.

(5) $d_2 = (K_{\mathrm{CH}_j-\mathrm{CN}_i} \| \sigma) \oplus H_1(F(a^s, P_{\mathrm{CN}_i}))$.

(6) Send (d_1, d_2) to the ith cluster node.

Upon receiving the ciphertext (d_1, d_2) from the jth cluster head, the ith cluster node designcrypts the session key as follows:

(1) compute $K \| \sigma = d_2 \oplus H_1(F(a^{V_{\mathrm{CN}_i}}, d_1))$, $h = H_2(K, d_1)$;

(2) accept K if $F(\sigma, d_1) = F(gd_1, P_{\mathrm{CH}_j})$ and report "FAILURE" otherwise.

5. Performance Evaluation

5.1. Complexity of Basic Operations. Now, let us compare the braid-based signcryption schemes with the RSA-based ones. According to Cha et al.'s implementation [39] and Maffre's test [40], the complexities of the braid operations, such as multiplication, inversion, and canonical form computation, are bounded by $\mathcal{O}(l^2 n \log n)$ in the sense of bit operations, where n and l are the braid index and the canonical length of involved braids, respectively. If we follow Maffre's suggestions by setting $n = 50$ and $l = 10$, then the number of bit operations for implementing these braid operations is proportional to 2^{15}. We know that the number of bit operations for implementing modular exponentials involved in RSA-based schemes is proportional to 2^{30} when the bit length of RSA modulus is set to 1024. This suggests that the proposed braid-based signcryption is about 2^{15} times faster than RSA-based ones.

Further, if we lift the security level of the RSA-based schemes to exp(92.80), which is comparable to the security level of our scheme (see Section 5.3), then the RSA modulus should be at least 2008 bits (see [23] for details). Then, the number of bit operations for implementing modular exponentials involved in RSA-based schemes is proportional to 2^{33}. This suggests that at the same security level, our braid-based signcryption is even efficient than that of RSA-based ones.

Table 2: Parameter length.

Parameter	Components and domains	Size	Size in bits ($n = 50$, $l = 10$)
System parameters	$n \in \mathbb{N}, a, b \in B_n$	$\lceil \log n + 2 \ln \log n \rceil$	5650
Private key[1]	$s \in \{0,1\}^k$	k	80
Public key	$b^{as} \in B_n$	$\lceil \ln \log n \rceil$	2822
Signcryption[2]	$(c_1, c_2) \in B_n \times B_n^2$	$\lceil 3 \ln \log n \rceil$	8466
Total	—	$\approx \lceil 6 \ln \log n \rceil$	≈17 K

[1]It is enough to use 80-bit private keys in WSN-oriented applications.
[2]The length of c_2 is about equivalent to the length of two braids.

Table 3: Complexities and security levels.

	RSA-based schemes [23]			Braid-based schemes	
	Technique	$k = 1024$	$k = 2008$	Technique	$n = 50, l = 10$
Signcryption	Modular Exp.	2^{30}	2^{33}	Braid operation	2^{15}
Security level[1]	Factoring	exp (69.69)	exp (92.80)	Solving CSP	exp (92.80)

[1]The security level of RSA-based schemes are evaluated according to the best known factoring method, that is, the number field sieve (NFS) method [41].

5.2. Parameter Size. A braid in B_n with l canonical factors can be represented by a bit string of size $\lceil \ln \log n \rceil$ [16]. Thus, when $n = 50$ and $l = 10$, the sizes of the system parameters, the private-key, the public-key, and the ciphertexts are 5650 bits, 80 bits, 2822 bits, and 8466 bits, respectively. In total, it is about 17 Kbits (see Table 2). According to [5], a typical WSN node, MICA2 mote, developed by the University of California at Berkeley has an 8-bit 7.3 MHz processor with 4 KB (i.e., 32 Kbits) RAM and 128 KB programmable ROM. This suggests that although our scheme will take more memory than RSA-based ones, it is still compact enough to be deployed in typical WSN environments.

5.3. Security Levels. In [23], Wang et al. presented an analysis of the security levels of braid-based cryptosystems against two typical attacks: heuristic attacks and brute force attacks. In a similar manner, we can discuss the security levels of the proposed signcryption scheme. According to [23], the security level of a cryptosystem is modeled as the number of bit operations for breaking the cryptosystem. Since this number is in general huge, we always use its logarithm in evaluation and refer to as the logarithmic security level.

As for braid-based cryptosystems, heuristic attacks mean currently known smart attacks, such as length-based attacks [42, 43] and linear representation attacks. According to Maffre's test [40] and Wang et al.'s summarization [23], the logarithmic complexity of existing heuristic attacks against braid-based cryptosystems can be expressed as $\log(C_{150}^{50}) \approx 92.80$.

Let us proceed to analyze the security level against brute force attacks. According to Ko et al. [29], when the private-keys of braid-based schemes are selected carefully, that is, avoiding the weak keys mentioned by Maffre [40], all known heuristic attacks will be unsuccessful. Further, according to the previous analysis given by Ko et al. [16], the complexity of carrying brute force attacks towards braid-based schemes is proportional to $\exp((1/2) \ln \log n)$. Therefore, when we adopt Maffre's suggestion by setting the braid index and the canonical length of the involved braids to $n = 50$ and $l = 10$, respectively, the security level of our scheme against brute force attacks is proportional to exp(978). This suggests that in the foreseeable future it is infeasible to launch exhaustive attacks towards our proposal.

In brief, we can summarize the performance comparisons in two cases: in Case I, we consider the currently acceptable parameter settings, and in Case II, we lift the security level of the RSA-based schemes to exp(92.80) by increasing the length of the corresponding RSA modulus. The results are listed in Table 3. We can conclude that our scheme is very fast in signcrypting and designcrypting, but acceptably larger in storage requirement.

Remark 12. Although Table 3 seems very similar to that in [23], there are remarkable differences as follows: on one hand, in [23], the efficiencies of the signing process and the verifying process of the braid-based signature scheme in [23] are much different; signing can be implemented in the complexity proportional to 2^{15}, while the complexity of verifying is proportional to 2^{34}. However, the efficiencies of the signcrypting process and the designcrypting process in this paper are same: both of them are proportional to 2^{15} since in our new proposal it is unnecessary to solve the CDP problem over braid groups; on the other hand, the braid-based scheme in [23] is merely a signature scheme, while the proposal in this paper is a signcryption scheme. This suggests that our signcryption scheme is much efficient than Wang et al.'s signature scheme [23]. In brief, our proposal does more and faster than that in [23].

6. Conclusion

Lightweight cryptographic schemes are useful for securing WSN-oriented applications. To minimally incorporate confidentiality, authenticity, integrity, scalability, and flexibility, signcryption is the proper primitive to realize key management protocols for WSNs. However, most existing

signcryption schemes are heavyweight and not suitable for resource-limited sensors. In this paper, we propose a braid-based signcryption scheme and then develop a key establishment protocol for wireless sensor networks. From the complexity view, the proposed scheme is 2^{15} times faster than RSA-based ones. As far as we know, this proposal is the first signcryption scheme based on noncommutative algebraic structures. In addition, the analysis of the basic operations and parameter sizes suggests that our proposal can be efficiently deployed in typical WSN environments.

Acknowledgments

This work is partially supported by the National Natural Science Foundation of China (NSFC) (nos. 61003285, 61070251, 61103198), the NSFC A3 Foresight Program (no. 61161140320) and the JSPS A3 Foresight Program, JSPS Research Fellowships for Young Scientists Program, and NEC C&C Foundation.

References

[1] M. Dong, K. Ota, X. Li, X. Shen, S. Guo, and M. Guo, "HARVEST: a task-objective efficient data collection scheme in wireless sensor and actor networks," in *Proceedings of the 3rd International Conference on Communications and Mobile Computing (CMC '11)*, pp. 485–488, April 2011.

[2] K. Ota, M. Dong, and X. Li, "TinyBee: mobile-agent-based data gathering system in wireless sensor networks," in *Proceedings of the IEEE International Conference on Networking, Architecture, and Storage (NAS '09)*, pp. 24–31, IEEE Press, July 2009.

[3] A. Liu and P. Ning, "TinyECC: a configurable library for elliptic curve cryptography in wireless sensor networks," in *Proceedings of the International Conference on Information Processing in Sensor Networks (IPSN '08)*, pp. 245–256, April 2008.

[4] M. Varchola, T. Guneysu, and O. Mischke, "MicroECC: a lightweight reconfigurable elliptic curve crypto-processor," in *Proceedings of the International Conference on Reconfigurable Computing and FPGAs (RECONFIG '11)*, pp. 204–210, IEEE Computer Society, Washington, DC, USA, 2011.

[5] E. A. A. A. Hagras, D. El-Saied, and H. H. Aly, "Energy efficient key management scheme based on elliptic curve signcryption for Wireless Sensor Networks," in *Proceedings of the 28th National Radio Science Conference (NRSC '11)*, April 2011.

[6] A. Dent and Y. Zheng, *Practical Signcryption*, Springer, Berlin, Germany, 2010.

[7] Y. Zheng, "Digital signcryption or how to achieve cost(signature & encryption) ≪ cost(signature) + cost(encryption), advances," in *Proceedings of the Advances in Cryptology (CRYPTO '97)*, vol. 1294 of *Lecture Notes in Computer Science*, pp. 165–179, Springer, 1997.

[8] R. Steinfeld and Y. Zheng, "A signcryption scheme based on integer factorization," in *Proceedings of the Information Security Workshop (ISW '00)*, vol. 1975 of *Lecture Notes in Computer Science*, pp. 308–322, Springer, 2000.

[9] J. Malone-Lee and W. Mao, "Two birds one stone: signcryption using RSA," in *Proceedings of the Cryptographers' Track at the RSA Conference (CTRSA '03)*, vol. 2612 of *Lecture Notes in Computer Science*, pp. 211–225, Springer, 2003.

[10] Y. Zheng and H. Imai, "How to construct efficient signcryption schemes on elliptic curves," *Information Processing Letters*, vol. 68, no. 5, pp. 227–233, 1998.

[11] M. Toorani and A. A. Beheshti, "A directly public verifiable signcryption scheme based on elliptic curves," in *Proceedings of the IEEE Symposium on Computers and Communications (ISCC '09)*, pp. 713–716, IEEE Computer Society, July 2009.

[12] L. Zhang and T. Mo, "A signcryption scheme for WEP in WLAN based on bilinear pairings," in *Proceedings of the International Conference on Computer Application and System Modeling (ICCASM '10)*, vol. 8, pp. 126–130, IEEE Computer Society, October 2010.

[13] J. Zhang, Y. Yang, and X. Niu, "A novel identity-based multi-signcryption scheme," *International Journal of Distributed Sensor Networks*, vol. 1, no. 5, p. 28, 2009.

[14] F. Li, F. Muhaya, M. Khan, and T. Takagi, "Lattice-based signcryption," *Concurrency and Computation*, vol. 2, pp. 1–10, 2012.

[15] F. Wang, Y. Hu, and C. Wang, "Post-quantum secure hybrid signcryption from lattice assumption," *Applied Mathematics and Information Sciences*, no. 6, pp. 23–28, 2012.

[16] K. Ko, S. Lee, J. Cheon, and J. Han, "New public-key cryptosystem using braid groups," in *Proceedings of the Advances in Cryptology (CRYPTO '00)*, vol. 1880 of *Lecture Notes in Computer Science*, pp. 166–183, Springer, Berlin, Germany, 2000.

[17] I. Anshel, M. Anshel, B. Fisher, and D. Goldfeld, "New key agreement protocols in braid group cryptography," in *The Cryptographers' Track at RSA Conference (CT-RSA '01)*, vol. 2020 of *Lecture Notes in Computer Science*, pp. 13–27, Springer, Berlin, Germany, 2001.

[18] I. Anshel, M. Anshel, and D. Goldfeld, "An algebraic method for public-key cryptography," *Mathematical Research Letters*, vol. 6, no. 3-4, pp. 287–291, 1999.

[19] M. Anshel, "Braid group cryptography and quantum cryptoanalysis," in *Proceedings of the 8th International Wigner Symposium*, pp. 13–27, GSUCCUNY, May 2003.

[20] K. Ko, D. Choi, M. Cho, and J. Lee, "New signature scheme using conjugacy problem," Preprint, 2002, http://eprint.iacr.org/2002/168.

[21] L. Wang, Z. Cao, P. Zeng, and X. Li, "One-more matching conjugate problem and security of braid-based signatures," in *Proceedings of the 2nd ACM Symposium on Information, Computer and Communications security (ASIACCS '07)*, pp. 295–301, ACM Press, March 2007.

[22] L. Wang, Z. Cao, S. Zheng, X. Huang, and Y. Yang, "Transitive signatures from braid groups," in *Proceedings of the Progress in Cryptology (INDOCRYPT '07)*, vol. 4859 of *Lecture Notes in Computer Science*, Springer, December 2007.

[23] L. Wang, L. Wang, Z. Cao, Y. Yang, and X. Niu, "Conjugate adjoining problem in braid groups and new design of braid-based signatures," *Science in China, Series F*, vol. 53, no. 3, pp. 524–536, 2010.

[24] L. Wang, L. Wang, Z. Cao, E. Okamoto, and J. Shao, "New constructions of public-key encryption schemes from conjugacy search problems," in *Proceedings of the International Conference Information Security and Cryptology (Inscrypt '11)*, vol. 6584 of *Lecture Notes in Computer Science*, pp. 1–17, Springer, 2011.

[25] V. Shpilrain and A. Ushakov, "An authentication scheme based on the twisted conjugacy problem," in *Proceedings of the Applied Cryptography and Network Security (ACNS '08)*, vol. 5037 of *Lecture Notes in Computer Science*, pp. 366–372, Springer, Berlin, Germany, 2008.

[26] J. S. Birman, V. Gebhardt, and J. González-Meneses, "Conjugacy in garside groups—I: cyclings, powers, and rigidity," *Groups, Geometry and Dynamics*, vol. 1, no. 3, pp. 221–279, 2007.

[27] J. S. Birman, V. Gebhardt, and J. González-Meneses, "Conjugacy in Garside groups—III: periodic braids," *Journal of Algebra*, vol. 316, no. 2, pp. 746–776, 2007.

[28] J. S. Birman, V. Gebhardt, and J. González-Meneses, "Conjugacy in garside groups II: structure of the ultra summit set," *Groups, Geometry and Dynamics*, vol. 2, no. 1, pp. 16–31, 2008.

[29] K. H. Ko, J. W. Lee, and T. Thomas, "Towards generating secure keys for braid cryptography," *Designs, Codes, and Cryptography*, vol. 45, no. 3, pp. 317–333, 2007.

[30] M. Prasolov, "Small braids having a big ultra summit set," http://arxiv.org/abs/0906.0076.

[31] P. Dehornoy, "Braid-based cryptography," *Contemporary Mathematics—American Mathematical Society*, vol. 360, pp. 5–33, 2004.

[32] P. Dehornoy, "Using shited conjugacy in braid-based cryptography," *Algebraic Methods in Cryptography, Contemporary Mathematics—American Mathematical Society*, vol. 418, pp. 65–74, 2006.

[33] U. Maurer, "Abstract models of computation in cryptography," in *Proceedings of the Cryptography and Coding*, N. P. Smart, Ed., vol. 3796 of *Lecture Notes in Computer Science*, pp. 1–12, Springer, Heidelberg, Germany, 2005.

[34] E. Fujisaki and T. Okamoto, "How to enhance the security of public-key encryption at minimum cost," in *Proceedings of the International Workshop on Practice and Theory in Public Key Cryptography (PKC '99)*, H. Imai and Y. Zheng, Eds., vol. 1560 of *Lecture Notes in Computer Science*, pp. 53–68, Springer, Heidelberg, Germany, 1999.

[35] L. Gu, L. Wang, K. Ota, M. Dong, Z. Cao, and Y. Yang, "New public key cryptosystems based on non-abelian factorization problems," *Security and Communication Networks*. In press.

[36] D. Kahrobaei and C. Koupparis, "Non-commutative digital signatures," *Groups Complexity and Cryptology*, vol. 4, pp. 377–384, 2012.

[37] A. Dent, "Hybrid signcryption schemes with insider security," in *Proceedings of the 10th Australasian Conference on Information Security and Privacy (ACISP '05)*, vol. 3574 of *Lecture Notes in Computer Science*, pp. 253–266, Springer, 2005.

[38] A. Dent, "Hybrid signcryption schemes with outsider security," in *Proceedings of the 8th International Conference on Information Security (ISC '05)*, vol. 3650 of *Lecture Notes in Computer Science*, pp. 203–217, Springer, 2005.

[39] J. Cha, K. Ko, S. Lee et al., "An efficient implementation of braid groups," in *Proceedings of the International Conference on the Theory and Application of Cryptology and Information Security (ASIACRYPT '01)*, vol. 2248 of *Lecture Notes in Computer Science*, pp. 144–156, Springer, Berlin, Germany, 2001.

[40] S. Maffre, "A weak key test for braid based cryptography," *Designs, Codes, and Cryptography*, vol. 39, no. 3, pp. 347–373, 2006.

[41] D. Coppersmith, "Modifications to the number field sieve," *Journal of Cryptology*, vol. 6, no. 3, pp. 169–180, 1993.

[42] J. Hughes, "The left sss attack on ko-lee-cheon-han-kang-park key agreement scheme in b_{45}," in *Rump Session Crypto*, 2000.

[43] J. Hughes, "A linear algebraic attack on the aafg1 braid group cryptosystem," in *Proceedings of the 7th Australasian Conference on Information Security and Privacy (ACISP '02)*, vol. 2384 of *Lecture Notes in Computer Science*, pp. 176–189, Springer, Berlin, Germany, 2002.