IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received February 12, 2019, accepted March 1, 2019, date of publication March 11, 2019, date of current version April 2, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2904146

New Construction of Blind Signatures From Braid Groups

LICHENG WANG[1], YANMEI TIAN[1], YUN PAN[2], AND YIXIAN YANG[1]
[1]State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China
[2]School of Computer Sciences and Technology, Communication University of China, Beijing 100024, China

Corresponding author: Yun Pan (pany@cuc.edu.cn)

This work was supported in part by the National Key Research and Development Program of China under Grant 2016YFB0800602, and in part by the Shandong Provincial Key Research and Development Program of China under Grant 2018CXGC0701.

ABSTRACT A new construction of a blind signature scheme based on braid groups is proposed. In the random oracle model, the proposed scheme is provably unforgeable against chosen message attacks, assuming that the one-more matching conjugate problem in braid groups is intractable. Furthermore, in the infinite group model, the scheme is proved to be perfectly blind. Our construction represents a technique to lift a braid group to its conjugate subgroups for particular applications. The proposed scheme is very fast in signing but relatively slow in verifying and is thus suitable for scenarios that require signing as soon as possible but permit a slight delay in verifying. In addition, our proposal is invulnerable to known quantum attacks and therefore would be a good alternative to RSA-based and DLP-based blind signatures in the post-quantum era.

INDEX TERMS Blind signature, non-commutative cryptography, lightweight, quantum attack resistant.

I. INTRODUCTION

A. BACKGROUND, MOTIVATION AND CONTRIBUTIONS

The concept of blind signatures was invented by Chaum [1] as a key ingredient for anonymous electronic cash applications. Blind signatures allow a signer to issue signatures without knowing the content of the signed documents while simultaneously preventing users from forging signatures [2]–[9]. In general, a blind signature σ on a given message m is produced in three steps:

1) Blinding: The user transforms m into $\widehat{m}$ by employing a random and secret factor b, usually called a blind factor, and then sends $\widehat{m}$ to the signer.
2) Signing: Upon receiving a signing request on $\widehat{m}$, the signer produces signature $\widehat{\sigma}$ on $\widehat{m}$ and then sends $\widehat{\sigma}$ to the user.
3) Unblinding: Upon receiving $\widehat{\sigma}$, the user removes the blind factor b involved in $\widehat{\sigma}$ and then obtains the signature σ on the message m.

According to the scenario, a blind signature scheme allows users to output signatures that are *not* signed by the signer. Thus, we should assign new secure semantics to blind signatures. On one hand, a blind signature scheme is *unforgeable* if there is no adversary that can, with non-negligible probability, fulfill the so-called one-more forgery attack [2], which states that an adversary, usually modeled as a probabilistic polynomial time algorithm, breaks the unforgeability of a blind signature scheme if he/she can output at least $l + 1$ valid signatures after requesting l signing queries to the signer. On the other hand, as a restriction towards the signer, the *blindness* property of a secure blind signature scheme requires that a malicious signer has no more advantage than guessing to determine the order in which the messages are signed by interaction with an honest user [1], [3], [9]. The blindness property is an abstraction of two basic requirements of paper-made cash systems: *unlinkability* and *untraceability*.

Numerous blind signature schemes have been constructed based on the integer factoring problem (IFP), discrete logarithm problem (DLP), and other variant assumptions related to the IFP or DLP [4]–[7], [10], [11]. However, these schemes are vulnerable to the quantum algorithmic attacks invented by Shor [12], Kitaev [13], and Proos and Zalka [14]. Thus, a fundamental idea for securing electronic cash applications in the post-quantum era is to design new blind signature scheme based on new hard problems. In addition to the well-known lattice problems, such as the shortest vector problem (SVP), the closest vector problem (CVP) [15] and the learn with error (LWE) problem [16], [17], the conjugacy

The associate editor coordinating the review of this manuscript and approving it for publication was Zhitao Guan.

problems from some non-commutative groups also have potential advantages in resisting quantum attacks [18].

Our motivation is to design a quantum-secure blind signature scheme based on braid groups. Two key techniques were employed for ensuring the security of our design: a blinding technique embedded in the homomorphic property of the conjugate operation and a hash function that maps every message to the conjugate subgroup of the underlying group. These two techniques are not new: the former can be found in [19] and [20], and the latter first appeared in [21]. Our contribution is, for the first time, to couple the techniques to ensure the blindness property.

In brief, our work includes three aspects. First, a new blind signature scheme based on braid groups is presented. Next, a technique to lift the braid group to its conjugate subgroups for particular purposes is proposed. Finally, comparisons and evaluations of parameter selection, performance, and security levels are provided.

B. RELATED WORK

In 2008, Verma [22] proposed two blind signatures using braid groups. Unfortunately, Verma's schemes are insecure: they are linkable since the signer can decide the order in which the blind signatures are produced by judging the conjugate relationships between the blinded and unblinded signatures. Similar attacks on Verma's schemes were reported by Kumar [23]. In our opinion, Verma's schemes are insecure because they fail to combine the aforementioned two techniques in [19], [20], and [21]. Beyond [21], we provide a further investigation of the security of the involved hash function.

Other constructions of braid-based blind signatures have been reported. Yun *et al.* [24] proposed a strong blind signature scheme over braid groups. Their main contributions are based in two aspects: simultaneously working on four separate subgroups of B_n and using additional randomness to avoid the weakness of simultaneous conjugating. Li *et al.* [25] proposed a proxy blind signature by laying the security on the hardness of the conjugacy problem, simultaneous conjugacy problem and root problem. Ren and Chang [26] further built a threshold proxy blind multisignature over braid groups. However, all these designs need to be completed by, for example, providing more rigorous security proofs and more elaborate performance evaluations.

C. ORGANIZATION

The remaining content is organized as follows. The necessary preliminaries on braid groups and related hard problems are presented in Section II. Our main contributions, including the design of the building blocks, the intended braid-based blind signature scheme and the involved hash function, are given in Sections III, IV and V, respectively. A performance evaluation is presented in Section VI. Finally, all proofs of the theorems are provided in Appendix.

II. PRELIMINARIES

For $n \geq 2$, one can define the braid group $B_n = \langle A|R_1, R_2\rangle$ by $n-1$ Artin generators $A = \{a_1, \cdots, a_{n-1}\}$ with two types of generating relations: $R_1 = \{a_i a_j = a_j a_i : |i-j| > 1\}$ and $R_2 = \{a_i a_j a_i = a_j a_i a_j : |i-j| = 1\}$. For $n \geq 3$, B_n is not commutative, and its center is an infinite cyclic subgroup. Two braids $x, y \in B_n$ are said to be *conjugate*, written as $x \sim y$, if $y = zxz^{-1}$ holds for some braid $z \in B_n$ (here, z or z^{-1} is called a *conjugator* of x and y). Over the braid group B_n, we can define the following cryptographic problems related to conjugacy [27]:

- Conjugacy Decision Problem (CDP): Decide whether $x \sim y$ holds for a given instance $(x, y) \in B_n^2$.
- Conjugator Search Problem (CSP): Find $z \in B_n$ such that $y = zxz^{-1}$ holds for a given instance $(x, y) \in B_n^2$ with $x \sim y$.
- Generalized Conjugator Search Problem (GCSP): Find $z \in B_m$ $(m < n)$ such that $y = zxz^{-1}$ holds for a given instance $(x, y) \in B_n^2$ with $x \sim y$.
- Matching Conjugate Problem (MCP): Find $y' \in B_n$ such that $y \sim y'$ and $xy \sim x'y'$ for a given instance $(x, x', y) \in B_n^3$ with $x \sim x'$.

At present, we know that all the above problems are solvable; however, in general cases, we do not know whether they are tractable. In the worst case, all the problems appear to be intractable [27], [28]. The assumption of intractability of the CSP is the common basis of most, if not all, existing braid-based cryptosystems. In their pioneering paper [27], Ko *et al.* proposed a one-way function based on the assumption of the intractability of the GCSP. Moreover, they defined an algorithm, $\mathcal{K}_{gcsp}$, for sampling hard GCSP pairs. The MCP was first formulated and proved to be equivalent to the CSP in [29]. A probabilistic algorithm for solving the CDP with high accuracy and acceptable complexity was proposed [29] (See Section VI). This algorithm is crucial for braid-based signature schemes that need to determine whether two given braids are conjugate.

III. THE BUILDING BLOCK

Our building block, denoted by V-SCSS, is a variant of the simple conjugate signature scheme (SCSS) [29]. Suppose that integer n is the braid index and the security parameter. Let B_n be the underlying group and $\mathcal{M} = \{0, 1\}^*$ the message space. For a given braid $p \in B_n$, $H_p : \mathcal{M} \to p^{B_n}$ is a cryptographic hash function that maps an arbitrary message to a braid conjugate to p. Here, p is the parameter braid that is fixed and made public in advance. Now, suppose $(n, B_n, \mathcal{M}, H_{(\cdot)})$ is the system public parameters, where $H_{(\cdot)}$ indicates H_p when p is unspecified or indifferent. Then, V-SCSS consists of the following algorithms:

- $\mathcal{G}(1^n)$, a probabilistic key generating algorithm that takes as input the security parameter n and then invokes $\mathcal{K}_{gcsp}(n)$. Suppose $(p, q, s) \xleftarrow{\$} \mathcal{K}_{gcsp}(n)$. Then, the signer takes s and the pair (p, q) as her (private)

signing key and (public) verifying key, respectively. According to the definition of $\mathcal{K}_{gcsp}(n)$, we know that $s \in B_{\lfloor n/2 \rfloor}$, $(p, q) \in B_n^2$ and $(p, q = sps^{-1})$ is a GCSP-hard pair. Furthermore, $H_{(\cdot)}$ is parameterized to H_p when p is published.

- $\mathcal{S}(s, m)$, a deterministic signing algorithm that takes as inputs the signing key s and the message $m \in \mathcal{M}$ and outputs

$$\sigma = s \cdot H_p(m) \cdot s^{-1} \tag{1}$$

as the signature on m.

- $\mathcal{V}(p, q; m, \sigma)$, a deterministic verifying algorithm that takes as inputs the public key pair (p, q) and the message-signature pair (m, σ) and outputs 1 if

$$\sigma \sim H_p(m) \text{ and } q\sigma \sim pH_p(m) \tag{2}$$

hold simultaneously, or 0 otherwise.

A. CONSISTENCY AND SECURITY

Theorem 1 (Consistency of V-SCSS): The proposed signature scheme V-SCSS is consistent, i.e., if the signer follows the signature generation protocol, the resulting signature satisfies the verification with probability 1.

Proof: See Appendix A. ■

Theorem 2 (Unforgeability of V-SCSS): Suppose n is the security parameter and $H_{(\cdot)}$ is a random oracle. The above signature scheme is existentially unforgeable against chosen message attack (EUF-CMA) under the intractability assumption of the matching conjugate problem over the braid group B_n.

Proof: See Appendix B. ■

Remark 1: Note that in the verification formula (2), the first conjugate relationship is necessary for resisting the following forgery: $\sigma^* = q^{-1}b^{-1} \cdot pH_p(m) \cdot b$ for some $b \in B_n$. The first verification condition is seemingly unnecessary since all signatures and all hash values on arbitrary messages are conjugate to each other; in fact, this is not the case. The forger can set $\sigma = q^{-1} \cdot pH_p(m) = s^{-1}p^{-1}s \cdot pH_p(m)$, and we can see that σ satisfies the second conjugate relationship. However, this type of forgery cannot pass the checking on the first conjugate relationship because $s^{-1}p^{-1}spH_p(m) \sim H_p(m)$ is unlikely to hold or to holds only with negligible probability.

IV. THE MAIN CONSTRUCTION

A. SCHEME DESCRIPTION

Suppose $(n, B_n, \mathcal{M}, H_{(\cdot)})$ is the system public parameters as defined in the above section. Our braid-based blind signature scheme, denoted by B3S, consists of the following components:

- $\mathcal{G}(1^n)$, a probabilistic key generating algorithm that is the same as $\mathcal{G}(1^n)$ of V-SCSS.
- Blind signature issuing protocol:
  - *blinding*(p, m), a probabilistic algorithm executed by the user that takes as inputs the public key p and the original message m and performs the following steps:
    (i) Selects a braid $b \in RB(\lfloor n/2 \rfloor + 1, n - 1)$ at random;
    (ii) Computes the blind message braid

$$\widehat{m} = b^{-1} \cdot H_p(m) \cdot b, \tag{3}$$

where $RB(j, k)$ $(j < k)$ is the subgroup generated by Artin generators $a_j, a_{j+1}, \cdots, a_k$;
    (iii) Sends $\widehat{m}$ to the signer.
  - *signing*$(s, \widehat{m})$, a deterministic algorithm executed by the signer that takes as inputs the signing key s and the blinded message $\widehat{m}$ and replies to the user with the blinded signature

$$\widehat{\sigma} = s \cdot \widehat{m} \cdot s^{-1}. \tag{4}$$

  - *unblinding*$(\widehat{\sigma}, b)$, a deterministic algorithm executed by the user that takes as inputs a blinded signature $\widehat{\sigma}$ and the corresponding blind factor b and outputs

$$\sigma = b\widehat{\sigma}b^{-1} \tag{5}$$

as the (unblinded) signature on the original message m.

- $\mathcal{V}(p, q; m, \sigma)$, a deterministic verifying algorithm that is the same as $\mathcal{V}(p, q; m, \sigma)$ of V-SCSS.

B. CONSISTENCY AND SECURITY

Theorem 3 (Consistency of B3S): If the signer and the user follow the blind signature issuing protocol, the resulting signature satisfies the verification with probability 1.

Proof: See Appendix C. ■

Theorem 4 (Unforgeability of B3S): Suppose n is the security parameter and $H_{(\cdot)}$ is a random oracle. The proposed blind signature scheme B3S is existentially unforgeable against chosen message attack (EUF-CMA) under the intractability assumption of the matching conjugate problem over braid group B_n. More specifically, if there is a one-more forger $\mathcal{F}$ that can break B3S, then there also exists a forger $\mathcal{F}'$ that can break V-SCSS.

Proof: See Appendix D. ■

Theorem 5 (Blindness of B3S): The proposed blind signature scheme B3S is blind assuming that the GCSP over braid groups is intractable. Furthermore, B3S achieves perfect blindness in the infinite group model.

Proof: See Appendix E. ■

Remark 2: Here, the infinite group model means that we take the whole braid group B_n as the background for constructing our proof, which is different from reality in that B_n is replaced by a finite chopped subspace. However, if we choose a truncated braid group that contains all our inputs and outputs under the chosen security parameters, all the theories developed in the whole braid group should hold [30].

TABLE 1. Parameter selection.

Parameter	Domain or Range	Size bound	Size in bits (when $n = 50, l = 100$)
Work group	B_n	—	—
Private key	$s \in B_{\lfloor n/2 \rfloor}$	$l\lfloor n/2 \rfloor \log\lfloor n/2 \rfloor$	$< 2^{14} = 16K$
Public key	$p \in B_n,\ q = sps^{-1} \in B_n$	$4(ln \log n)$	$< 2^{17} = 128K$
Blinding factors	$b \in RB(\lceil n/2 \rceil, n-1)$	$l\lfloor n/2 \rfloor \log\lfloor n/2 \rfloor$	$< 2^{14} = 16K$
Hash h	$h(m) \in B_n$	$ln \log n$	$< 2^{15} = 32K$

TABLE 2. Complexities and security levels.

	RSA-based Schemes			Braid-based Schemes	
	Technique	$k = 1024$	$k = 2048$	Technique	$n = 50, l = 100$
(Blind) Signing Complexity	Modular exponentiation	2^{30}	2^{33}	Braid operation	2^{22}
Verifying Complexity				Solving CDP	2^{38}
Security Level	Factoring	exp(69.69)	exp(92.80)	Solving CSP	exp(92.80)

V. DESIGN OF H_p

Both V-SCSS and B3S use a special hash function H_p that maps the message space $\mathcal{M}$ to p^{B_n}. In the above sections, H_p is modeled as a random oracle for security reductions. For practical purposes, we need to discuss how to implement H_p.

In 2000, Ko *et al.* [27] described an implementation of cryptographic hashes that map bit strings to braids. However, the images of these hashes are not necessarily conjugate to a common braid p that is fixed and made public in advance; therefore, we cannot use Ko *et al.*'s constructions directly. Instead, we must proceed further. Let B_n be the underlying group and $\mathcal{M} = \{0,1\}^*$ the message space and suppose that we already have a hash function $h : \mathcal{M} \to B_n$ that maps an arbitrary message to a random braid [27], [31]. Then, a practical instantiation of the parameterized hash function H_p can be defined as follows:

$$H_p : \mathcal{M} \to p^{B_n}, \quad m \mapsto h(m)ph(m)^{-1}. \tag{6}$$

That is, $H_p(m)$ is a braid that is conjugate to the braid p by taking $h(m)$ as the conjugator. Note that from the perspective of implementation, both B_n and p^{B_n} are replaced by some finite subsets of B_n.

Sibert *et al.* [21] used a similar method for hashing from braids to braids, but our main motivation for using this hash is to ensure blindness. Furthermore, we prove the following theorems on the security of this design.

Theorem 6: Suppose p is fixed and public. Then, we have

(i) *If h is one-way, so is H_p;*

(ii) *If h is second pre-image resistant, so is H_p, assuming that the GCSP in the braid groups is intractable;*

(iii) *If h is collision resistant, so is H_p.*

Proof: See Appendix F. ■

VI. PERFORMANCE EVALUATION

The complexities of braid operations, including multiplication and canonical form transformation, are bounded by $\mathcal{O}(l^2\ n \log n)$ in the sense of bit operations [32] and [33], where n and l are the braid index and canonical length of the involved braids, respectively. In addition, the verifying process of our schemes needs to solve the CDP. Ko *et al.* [29] invented a probabilistic algorithm that can solve the CDP with complexity $\mathcal{O}(rln^3)$, and the probability for making an erroneous decision, denoted by p_{err}, is bounded by $(\frac{ln^2}{2p})^r$, where p and r should be sufficiently large such that p_{err} is acceptable. Here, the complexity for deciding conjugacy is evaluated in the basic operation in finite field F_p that takes $\mathcal{O}(\log^2\ p)$ bit operations in turn. Thus, the total complexity would be $\mathcal{O}(rln^3 \log^2 p)$ in the sense of bit operations. Typically, if we set p to be a prime that closes to $2ln^2$ and $r = n$, then p_{err} is bounded by $(\frac{1}{2})^{2n}$, and the total bit complexity for deciding conjugacy is $\mathcal{O}(ln^4 \log^2(ln^2))$, which is proportional to 2^{38} when $n = 50$ and $l = 100$.

A braid in B_n with l canonical factors can be represented by a bit string of size $ln \log n$ [27], and for generic choices of $b, x \in B_n$, the *canonical length* of bxb^{-1} can be assumed to be between $2l$ and $3l$ or, equivalently, $2(ln \log n) \sim 3(ln \log n)$ bit size. Thus, when $n = 50$ and $l = 100$, the sizes of the private key and public key are approximately 16K and 128K bits, respectively (See Table 1 for details). Clearly, the keys are considerably large. At present, all braid-based cryptosystems suffer from this disadvantage.

The security levels of the proposal can be evaluated from two different aspects. First, according to [33], the bit complexity of existing heuristic attacks (say towards the weak keys) can be evaluated by $\binom{50}{150} \approx$ exp(92.80). Furthermore, if the private keys are selected carefully, all known heuristic attacks on braid-based cryptographic schemes would be frustrated [34]. Then, according to [27], the hardness for conducting a brute force attack is proportional to $\exp(\frac{1}{2}ln \log n)$. Thus, when $n = 50$ and $l = 100$, the security level against brute force attack is proportional to exp(978). This result suggests that it is impossible to launch exhaustion attacks on our scheme in the foreseeable future.

In brief, we can summarize the performance comparison for two cases: Case I takes the currently acceptable parameter settings and Case II increases the security level of RSA-based systems to exp(92.80). The results are shown in Table 2. In both cases, our scheme is very fast in signing and acceptably slow in verifying.

Remark 3: About 10 years ago, we [35] made the performance evaluation on braid-based signature schemes in

similar angles. But in today's view, the results presented in [35] is too optimistic to be secure. The main difference is the setting on the parameter l, i.e. canonical length of working braids. In [35], we set $l = 10$ *by taking into consideration of Maffre's suggestion [33] and the feasibility of Ko's CDP algorithm [29], while neglected the probability of keeping random working braids non-commutative. Now, with the purpose of ensuring the collision resistance property of braid-lifting hash function* H_p*, we suggested to set* $l = 100$ *so that the probability of two random braids being commutative vanishes.*

VII. CONCLUSIONS

Blind signatures have been used in numerous applications, most prominently in anonymous voting systems and anonymous e-cash systems. However, the development of quantum computation creates distrust in most number-theory-based blind signature schemes. In this paper, we proposed a new blind signature scheme based on hard braid problems. The scheme has several merits: round optimal (in communications), perfectly blind (in the infinite group model), invulnerable to known quantum attacks, etc. Our scheme is very fast in signing and acceptably slow in verifying and is thus suitable for scenarios that require promptness of signature generation but tolerate a delay in verifications. An observable inferiority of our scheme is that the private/public keys are large.

APPENDIX
PROOFS OF THE THEOREMS

A. PROOF OF THEOREM 1

Proof: The first conjugate relationship $\sigma \sim H_p(m)$ is apparently implied by the formula (1). The second can also be immediately derived from

$$q\sigma = sps^{-1} \cdot sH_p(m)s^{-1} = sp \cdot H_p(m)s^{-1} = s \cdot pH_p(m) \cdot s^{-1}.$$

■

B. PROOF OF THEOREM 2

Proof: First, let us introduce an one-more version of the MCP, i.e., the one-more matching conjugate problem (OM-MCP), which was introduced in [19] based on the corresponding intractability assumption. Ko *et al.*'s simple conjugate signature scheme (SCSS) was first proved to be unforgeable against chosen message attacks (UF-CMA) in the random oracle model (ROM). The OM-MCP is defined via an experiment involving a GCSP-hard pair generator $\mathcal{K}_{gcsp}$ and an OM-MCP attacker $\mathcal{A}$ are involved:

- $\mathcal{K}_{gcsp}$ is a probabilistic polynomial-time algorithm that takes as input the security parameter n and outputs a triple $(p, q, w) \in B_n \times B_n \times B_{\lfloor n/2 \rfloor}$ such that $q = wpw^{-1}$ and (p, q) is a GCSP hard pair, i.e., finding a conjugator $w' \in B_{\lfloor n/2 \rfloor}$ for the pair (p, q) is intractable.
- $\mathcal{A}$ is a probabilistic polynomial-time algorithm that receives input p, q and has access to two oracles — the matching conjugate oracle $\mathcal{O}_{mc}(\cdot)$ and the challenge oracle $\mathcal{O}_{ch}()$ — and wants to *win* the experiment.

Experiment $\mathcal{E}^{om-mcp}_{\mathcal{K}_{gcsp},\mathcal{A}}(n)$	Oracle $\mathcal{O}_{mc}(b)$
$(p, q, w) \xleftarrow{\$} \mathcal{K}_{gcsp}(n)$; $k \leftarrow 0; l \leftarrow 0$;	$l \leftarrow l + 1$; Return wbw^{-1}
$(r_1, \cdots, r_{k'}) \xleftarrow{\$} \mathcal{A}^{\mathcal{O}_{mc},\mathcal{O}_{ch}}(p, q, n)$; If $k' = k$ and $l < k$ and $\forall\, i = 1, \cdots, k$: $(r_i \sim c_i) \wedge (qr_i \sim pc_i)$ then return 1 else return 0	Oracle $\mathcal{O}_{ch}()$ $k \leftarrow k + 1$; $c_k \xleftarrow{\$} B_n$; Return c_k

FIGURE 1. One-more matching conjugate experiment.

The verb *win* means that $\mathcal{A}$ succeeds in matching conjugates with all $\eta(n)$ braids output by the challenge oracle $\mathcal{O}_{ch}()$ but submits strictly less than $\eta(n)$ queries to the matching conjugate oracle $\mathcal{O}_{mc}(\cdot)$, where $\eta : \mathbb{N} \to \mathbb{N}$ is arbitrary polynomials over $\mathbb{N}$. The formal definitions of the experiment, the oracle $\mathcal{O}_{mc}(\cdot)$ and the oracle $\mathcal{O}_{ch}()$ are depicted in Fig.1. (The symbol $x \xleftarrow{\$} \mathcal{X}$ indicates the process of random braid sampling by a probabilistic procedure $\mathcal{X}$ or from a subset $\mathcal{X} \subseteq B_n$.

The OM-MCP advantage of $\mathcal{A}$ is defined by

$$\mathbf{Adv}^{om-mcp}_{\mathcal{K}_{gcsp},\mathcal{A}}(n) = \Pr[\mathcal{E}^{om-mcp}_{\mathcal{K}_{gcsp},\mathcal{A}}(n) = 1], \tag{7}$$

i.e., the probability that the above experiment returns 1, taken over the coins of $\mathcal{K}_{gcsp}$, the coins of $\mathcal{A}$, and the coins used by the challenge oracle across its invocations. The *one-more matching conjugate assumption* states that the one-more matching conjugate problem associated with $\mathcal{K}_{gcsp}$ is intractable, i.e., $\mathbf{Adv}^{om-mcp}_{\mathcal{K}_{gcsp},\mathcal{A}}(n)$ is negligible with respect to the security parameter n for all probabilistic polynomial-time adversaries $\mathcal{A}$.

Wang *et al.* [19] presented a sufficient discussion of the OM-MCP assumption and the relationship between the OM-MCP and CSP (also GCSP). Note that in one-more type experiments, the adversaries are not permitted to choose challenges by themselves, but they can submit queries on their own choices. The adversaries win the experiments only if the number of submitted queries is strictly less than the number of correctly answered challenges [19], [36].

Next, let us sketch the proof of the theorem. Suppose $H_{(\cdot)}$ is parameterized by p. That the forger $\mathcal{F}$ can (t, q_h, q_s, ϵ)-break V-SCSS means that $\mathcal{F}$ can output a forged signature (m^*, σ^*) successfully with probability at least ϵ after he has made q_h hash queries and q_s signing queries and then obtained the corresponding signatures σ_i for messages m_i, $i = 1, \cdots, q_s$. A successful forgery σ^* on message m^* means that $\sigma^* \sim H_p(m^*)$ and $q\sigma^* \sim pH_p(m^*)$ hold while $\mathcal{F}$ has never made signature query on message m^*.

The key idea of the proof is that we can construct another algorithm $\mathcal{A}$ that can win the one-more matching conjugate experiment with probability at least $\epsilon' = \epsilon$ by taking the forgery $\mathcal{F}$ as a subroutine.

Let $\mathcal{A}$ engage in an interactive process with adversary $\mathcal{F}$ and record all necessary information. Whenever $\mathcal{F}$ invokes a hash query, $\mathcal{A}$ calls the oracle $\mathcal{O}_{ch}$ and forwards its response to $\mathcal{F}$. Whenever $\mathcal{F}$ invokes a signature query on some message m, then $\mathcal{A}$ performs the following steps:

- Finds[1] the response c for the corresponding hash query on m;
- Sends c to the oracle $\mathcal{O}_{mc}$;
- Obtains the response braid b and forwards b to $\mathcal{F}$ as the request signature.

Note that the probability that $\mathcal{F}$ forges a valid signature without invoking a hash query is negligible since the hash function is modeled as a random oracle. Then, whenever $\mathcal{F}$ finally outputs a forgery that can pass the validation, this forgery enables $\mathcal{A}$ to win the experiment $\mathcal{E}^{om-mcp}_{\mathcal{K}_{gcsp},\mathcal{A}}(n)$. ■

C. PROOF ON THEOREM 3

Proof: According to the unblinding formula (5), the signing formula (4) and the blinding formula (3), the first conjugate relationship $\sigma \sim H_p(m)$ holds apparently.

Note that $RB(\lceil n/2\rceil, n-1)$ is generated by Artin generators $a_{\lceil n/2\rceil}, a_{\lceil n/2\rceil+1}, \cdots, a_{n-1}$ and $B_{\lfloor n/2\rfloor}$ is generated by Artin generators $a_1, a_2, \cdots, a_{\lfloor n/2\rfloor-1}$. From the relations in the definition of braid groups, we know that for $\forall s \in B_{\lfloor n/2\rfloor}$ and $\forall b \in RB(\lceil n/2\rceil, n-1)$, $sb = bs$ holds. Thus, the second conjugate relationship can also be immediately derived from

$$\begin{aligned} q\sigma &= sps^{-1} \cdot b\widehat{\sigma}b^{-1} \\ &= sps^{-1} \cdot b \cdot s \cdot \widehat{m} \cdot s^{-1} \cdot b^{-1} \\ &= sps^{-1} \cdot b \cdot s \cdot b^{-1} \cdot H_p(m) \cdot b \cdot s^{-1} \cdot b^{-1} \\ &= sps^{-1} \cdot s \cdot b \cdot b^{-1} \cdot H_p(m) \cdot b \cdot b^{-1} \cdot s^{-1} \\ &= s \cdot pH_p(m) \cdot s^{-1} \\ &\sim pH_p(m). \end{aligned}$$

■

D. PROOF OF THEOREM 4

Proof: According to the semantics of the non-forgeability of blind signatures, we know that the one-more forger $\mathcal{F}$ can (t, q_h, q_s, ϵ)-break B3S means that within time t, $\mathcal{F}$ can output q_s+1 valid signatures with probability at least ϵ after he has made q_h hash queries and q_s signing queries. Now, we can construct a forger $\mathcal{F}'$ as follows:

(1) Let $\mathcal{F}'$ engage in an interactive process with $\mathcal{F}'$ and the signer of V-SCSS, denoted by $\mathcal{S}_{V-SCSS}$, and record all necessary information. In fact, $\mathcal{F}'$ will intercept all communications between $\mathcal{F}$'s and the signer of B3S, denoted by S_{B3S}. $\mathcal{F}'$ will attempt to simulate the behavior of S_{B3S} for $\mathcal{F}$, i.e., to provide responses to $\mathcal{F}$'s signing query on behalf of S_{B3S}. Moreover, since $H_{(\cdot)}$ is modeled as a random oracle, $\mathcal{F}'$ can control $H_{(\cdot)}$ and provide responses on hash queries $H_{(\cdot)}$ for all participants.

(2) $\mathcal{F}'$ will maintain a hash list, denoted by H-List, of the whole interactive process. H-List is initialized as empty, and the items in H-List consist of two fields: $m-field$ and $h-field$.

(3) When $\mathcal{S}_{V-SCSS}$ publishes his verifying key (p, q) and the system parameters (n, H_p), $\mathcal{F}'$ forwards all these parameters to $\mathcal{F}$ and claims that they are $\mathcal{S}_{B3S}$'s verifying key and the corresponding system parameters.

(4) Whenever $\mathcal{F}$ or $\mathcal{S}_{V-SCSS}$ makes a hash query on $m \in \mathcal{M}$, $\mathcal{F}'$ executes the following steps:
   (i) Tries to locate m in the m-field of H-List;
   (ii) If found, obtains h from the corresponding h-field; otherwise, randomly selects a braid $x \in B_n$, computes $h = xpx^{-1}$, and adds a new item (m, h) into H-List;
   (iii) Replies h to the corresponding requester, i.e., $\mathcal{F}$ or $\mathcal{S}_{V-SCSS}$.

(5) Similarly, whenever $\mathcal{S}_{V-SCSS}$ makes a hash query on m', $\mathcal{F}'$ executes the following steps:
   (i) Tries to locate m' in the m'-field of H-List;
   (ii) If found, obtain h' from the corresponding h'-field; otherwise, randomly selects a braid $y \in B_n$, computes $h = ypy^{-1}$, and adds a new item (m', h', m', h') into H-List;
   (iii) Replies to $\mathcal{S}_{V-SCSS}$ with h'.

(6) Whenever $\mathcal{F}$ makes a signing query on a blinded message braid $\widehat{m}$, $\mathcal{F}'$ executes the following steps (See Fig.2):
   (i) Randomly selects a message $m' \in \mathcal{M}$ such that m' does not appear in the $m-field$ of H-List,[2] and then adds a new item $(m', \widehat{m})$ to H-List;
   (ii) Requests $\mathcal{S}_{V-SCSS}$ to sign m'. According to the signing protocol, $\mathcal{S}_{V-SCSS}$ will ask for the hash value on the message m'. Therefore, upon receiving the hash query on m', $\mathcal{F}'$ replies $\widehat{m}$;
   (iii) Upon receiving the signature σ' of the message m' from $\mathcal{S}_{V-SCSS}$, $\mathcal{F}'$ forwards it to $\mathcal{F}$ as the response.

(7) After obtaining q_s signatures on messages $m_1, \cdots, m_{q_s}$, $\mathcal{F}$ outputs his forgery σ^* on the message m^*. If σ^* is not a valid signature on m^*, or if $\mathcal{F}$ has made signing query on m^* in the previous phases (i.e., there exists some $1 \leq j \leq q_s$ such that $m_j = m^*$), then $\mathcal{F}'$ aborts the simulation; otherwise, $\mathcal{F}'$ outputs (m^*, σ^*) as his own forgery.

From the perspective of $\mathcal{F}$, $\mathcal{F}'$'s responses to the hash and signing queries are perfect, and from the perspective of $\mathcal{S}_{V-SCSS}$, $\mathcal{F}'$'s responses to the hash queries are also perfect. Even if $\mathcal{F}$ and $\mathcal{S}_{V-SCSS}$ are allowed to compare their hash values, they cannot distinguish the responses from the output of a real hash H_p. Clearly, $\mathcal{F}'$ must intercept all signing queries from $\mathcal{F}$ and prevent $\mathcal{S}_{V-SCSS}$ from directly replying to $\mathcal{F}$'s signing queries. This agreement is rational because $\mathcal{S}_{V-SCSS}$ is not the signer of the scheme that $\mathcal{F}$ implements. In brief, $\mathcal{F}'$ directly forwards $\mathcal{F}$'s forgery as his own forgery since B3S and V-SCSS share the same verification algorithm and essentially similar signing protocols. $\mathcal{F}'$'s advantage is $\mathcal{F}$'s advantage. ■

[1] $\mathcal{A}$ can assume that $\mathcal{F}$ has already made the hash query on m. Otherwise, $\mathcal{A}$ can make this query on behalf of $\mathcal{F}$.

[2] Note that m' coincidentally appears in the $m-field$ of H-List with only a negligible probability.

$\mathcal{F}$'s action	comm.	$\mathcal{F}'$'s action	comm.	$\mathcal{S}_{V-SCSS}$'s action
$\widehat{m} = b^{-1}H_p(m)b$; Sending[a] $(SQ, \widehat{m})$;	$\xrightarrow{(SQ,\widehat{m})}$	Receiving $(SQ, \widehat{m})$;		
		$m' \xleftarrow{\$} \mathcal{M}$; H-List←H-List∪$\{(m', \widehat{m})\}$; Sending (SQ, m');	$\xrightarrow{(SQ,m')}$	Receiving (SQ, m');
		Receiving (HQ, m');	$\xleftarrow{(HQ,m')}$	Sending (HQ, m');
		Sending $(HR, \widehat{m})$;	$\dashrightarrow$	Receiving $(HR, \widehat{m})$; $\sigma' = s\widehat{m}s^{-1}$;
		Receiving (SR, σ');	$\xleftarrow{(SR,\sigma')}$	Sending (SR, σ');
Receiving $(SR, \widehat{\sigma})$; $\sigma = b\widehat{\sigma}b^{-1}$.	$\xleftarrow{(SR,\widehat{\sigma})}$	$\widehat{\sigma} \leftarrow \sigma'$; Sending $(SR, \widehat{\sigma})$;		

[a]Here, "SQ", "SR", "HQ" and "HR" represent "signing-query", "signing-reply", "hash-query" and "hash-reply", respectively.

FIGURE 2. $\mathcal{F}'$**s simulation of the production of blind signatures for** $\mathcal{F}$**.**

E. PROOF OF THEOREM 5

Proof: According to the definition of blindness, the attack objective of the adversarial signer $\mathcal{A}$ is to distinguish the order of the production of the signatures. For every message m, $H_p(m) \in p^{B_n}$ holds. Then, every signature σ conjugates to p and thus also conjugates to each other. Thus, $\mathcal{A}$ cannot break blindness by determining whether two braids are conjugate. Further, $\mathcal{A}$ cannot break the blindness by determining whether two braids are identical since $\mathcal{A}$ does not know the original messages he signed. $\mathcal{A}$'s signing was performed on the blinded messages; thus, $\mathcal{A}$'s has no means to extract the corresponding original messages.

Further, let us prove that in the infinite group model, even if $\mathcal{A}$ has the capability to solve the GCSP, he has no advantages to break the blindness of B3S. W.l.o.g., given two message-signature pairs (m_0, σ_0) and (m_1, σ_1), we will prove that $\mathcal{A}$'s success probability for determining which of them is produced earlier is no more than 1/2. Note that conjugators are always not unique for a given conjugate pair, and all valid signatures, all blind message braids, and all hash values of the original messages in our schemes are conjugate to each other. Therefore, we obtain a commutative diagram (8) in which double links indicate conjugate relationships. (Conjugate relationships are equivalent; thus, some double links are omitted for clarity.)

$$\begin{array}{ccccccc} m_0 & \longrightarrow & H_p(m_0) & = & \widehat{m}_0 & = & \sigma_0 \\ & & \| & & \| & & \| \\ m_1 & \longrightarrow & H_p(m_1) & = & \widehat{m}_1 & = & \sigma_1 \end{array} \tag{8}$$

Suppose that $\mathcal{A}$ has sufficient memory to record all blinded message braids $\widehat{m}_i$ $(i = 1, 2, \cdots)$ according to the time sequence of receiving them, i.e., $\widehat{m}_0$ appeared before $\widehat{m}_1$ from $\mathcal{A}$'s perspective. For a successful, he needs to find a witness that supports the *correlativity* between $H_p(m_0)$ and $\widehat{m}_0$ and simultaneously denies the *correlativity* between $H_p(m_0)$ and $\widehat{m}_1$; or vice versa. However, no well-defined concept exists for this type of *correlativity*. Even if $\mathcal{A}$ has worked out a blind factor $b_0 \in RB(\lfloor n/2 \rfloor + 1, n-1)$ that conjugates $H_p(m_0)$ to $\widehat{m}_0$, he cannot determine that σ_0 is produced earlier since $RB(\lfloor n/2 \rfloor + 1, n-1)$ is *infinite* and $\mathcal{A}$ has no reason to exclude the possibility of the existence of another blind factor $b_1 \in RB(\lfloor n/2 \rfloor + 1, n-1)$ that conjugates $H_p(m_0)$ to $\widehat{m}_1$. All real used blind factors are selected by the users and kept unknown to $\mathcal{A}$. Thus, $\mathcal{A}$ has no more advantage than guessing to decide whether m_0 or m_1 is the original message corresponding to blinded message braid $\widehat{m}_0$ or $\widehat{m}_1$. ■

F. PROOF OF THEOREM 6

Proof: The proof consists of three steps.

- One-wayness of H_p. If there is an adversary $\mathcal{A}$ that can derive m from $H_p(m)$ (i.e., $h(m)ph(m)^{-1}$) with non-negligible probability, then we can construct another adversary $\mathcal{B}$ that can also extract m from $h(m)$ with non-negligible probability, as follows:
  (1) Upon receiving challenge h, $\mathcal{B}$ computes $\mathcal{A}$'s challenge $H = hph^{-1}$;
  (2) $\mathcal{B}$ calls $\mathcal{A}$ to derive m with the input H;
  (3) Upon receiving response m from $\mathcal{A}$, $\mathcal{B}$ outputs m as the reply.

  Clearly, if $\mathcal{A}$ wins its challenge, so does $\mathcal{B}$; thus, the one-wayness of h implies the one-wayness of H_p.
- Second pre-image resistance of H_p. For a given m_1, only if one could find $m_2 (\neq m_1)$ such that
  (1) $h(m_1) = x \neq y = h(m_2)$ and $xpx^{-1} = ypy^{-1}$ or
  (2) $h(m_1) = x = h(m_2)$

hold, can he successfully break the second pre-image resistance of H_p.

In general, given x and p, it is difficult to find $y(\neq x)$ such that $xpx^{-1} = ypy^{-1}$ since extracting such y from the pair (x, xpx^{-1}) is equivalent to solving a GCSP instance. Therefore, no polynomial adversary can break the second pre-image resistance of H_p by taking Case (1) as the starting point.

As for Case (2), a successful attack means the breaking of the second pre-image resistance of h, which is a contradiction.

- Collision resistance of H_p. First, for randomly chosen braids $x, y \in B_n$, let us calculate $P_{x\leftrightarrow y}$, the probability that $(xy)p = p(xy)$. Let us use $P_{i\leftrightarrow j}$ to denote the commutative probability of two random Artin generators a_i and a_j $(i, j \in \{1, \cdots, n-1\})$. Apparently, we have

$$\begin{aligned} P_{i\leftrightarrow j} &\triangleq \mathbf{Pr}[a_i a_j = a_j a_i : i, j \in \{1, \cdots, n-1\}] \\ &= \mathbf{Pr}[|i-j| > 1 : i, j \in \{1, \cdots, n-1\}] \\ &= 1 - \mathbf{Pr}[|i-j| = 0 : i, j \in \{1, \cdots, n-1\}] \\ &\quad - \mathbf{Pr}[|i-j| = 1 : i, j \in \{1, \cdots, n-1\}] \\ &= 1 - \frac{n-1}{(n-1)^2} - \frac{2(n-2)}{(n-1)^2} \\ &= \frac{(n-2)(n-3)}{(n-1)^2} \\ &< \frac{(n-2)^2}{(n-1)^2} < 1. \end{aligned}$$

Suppose that $x = x_1 \cdots x_{|x|}$ and $y = y_1 \cdots y_{|y|}$, where each x_i and y_j are Artin generators. W.l.o.g., we assume that each x_i $(i = 1, \cdots, |x|)$ and each y_j $(j = 1, \cdots, |y|)$ are independent. Then,

$$\begin{aligned} P_{x\leftrightarrow y} &\triangleq \mathbf{Pr}[xy = yx : x, y \in B_n] \\ &= \prod_{i=1}^{|x|} \prod_{j=1}^{|y|} \mathbf{Pr}[x_i y_j = y_j x_i] \\ &= (P_{i\leftrightarrow j})^{|x||y|} \\ &= \left(\frac{n-2}{n-1}\right)^{2|x||y|}. \end{aligned}$$

For typical parameter settings, such as $n = 50$ and $|x| \approx |y| \approx 100$, the above probability vanishes.

Next, for two given random messages $m_1 \neq m_2$, we have $H_p(m_1) = h(m_1)ph(m_1)^{-1}$ and $H_p(m_2) = h(m_2)ph(m_2)^{-1}$. If $H_p(m_1) = H_p(m_2)$, we have $h(m_1)ph(m_1)^{-1} = h(m_2)ph(m_2)^{-1}$; then, we have $h(m_2)^{-1}h(m_1)p = ph(m_2)^{-1}h(m_1)$. If h is collision resistant, then we have $h(m_2)^{-1}h(m_1) \neq 1_{B_n}$ with overwhelming probability. Let $h(m_2) = x$ and $h(m_1) = y$. Then, we have $x^{-1}yp = px^{-1}y$. However, according to the above calculation, this occurs only with negligible probability, which suggests that H_p is collision resistant. ■

REFERENCES

[1] D. Chaum, "Blind signatures for untraceable payments," in *Advances in Cryptology*. New York, NY, USA: Plenum, 1982, pp. 199–203.
[2] D. Pointcheval and J. Stern, "Provably secure blind signature schemes," in *Advances in Cryptology* (Lecture Notes in Computer Science). Berlin, Germany: Springer-Verlag, 1996, pp. 252–265.
[3] A. Juels, M. Luby, and R. Ostrovsky, "Security of blind digital signatures," in *Advances in Cryptology* (Lecture Notes in Computer Science), vol. 1294. Berlin, Germany: Springer-Verlag, 1997, pp. 150–164.
[4] M. Abe, "A secure three-move blind signature scheme for polynomially many signatures," in *Proc. EUROCRYPT*, in Lecture Notes in Computer Science, vol. 2045. Springer-Verlag, 2001, pp. 136–151.
[5] A. Boldyreva, "Threshold signatures, multisignatures and blind signatures based on the gap-diffie-hellman-group signature scheme," in *Public Key Cryptography* (Lecture Notes in Computer Science), vol. 2567, Y. Desmedt, Ed. Springer-Verlag, Jan. 2003, pp. 31–46. [Online]. Available: http://link.springer.de/link/service/series/0558/bibs/2567/25670031.htm
[6] J. Camenisch, M. Koprowski, and B. Warinschi, "Efficient blind signatures without random oracles," in *Security in Communication Networks* (Lecture Notes in Computer Science), vol. 3352. Berlin, Germany: Springer-Verlag, 2004.
[7] T. Okamoto, "Efficient blind and partially blind signatures without random oracles," in *Proc. TCC*, in Lecture Notes in Computer Science, vol. 3876. Berlin, Germany: Springer, 2006, pp. 80–99.
[8] A. Kiayias and H.-S. Zhou, "Concurrent blind signatures without random oracles," in *Proc. SCN*, in Lecture Notes in Computer Science, vol. 4116, R. D. Prisco and M. Yung, Eds. Berlin, Germany: Springer, 2006, pp. 49–62.
[9] A. B. Buan, K. Gjøsteen, and L. Kråkmo, "Universally composable blind signatures in the plain model," *IACR Cryptol. ePrint Arch.*, vol. 2006, pp. 1–18, Nov. 2006.
[10] J. L. Camenisch, J.-M. Piveteau, and M. A. Stadler, "Blind signatures based on the discrete logarithm problem," in *Advances in Cryptology*. Berlin, Germany: Springer-Verlag, 1994, pp. 428–432.
[11] D. Pointcheval and J. Stern, "New blind signatures equivalent to factorization (extended abstract)," in *Proc. ACM Conf. Comput. Commun. Secur.*, 1997, pp. 92–99.
[12] P. W. Shor, "Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer," *SIAM Rev.*, vol. 41, no. 2, pp. 303–332, 1999.
[13] A. Y. Kitaev. (1995). "Quantum measurements and the abelian stabilizer problem." [Online]. Available: https://arxiv.org/abs/quant-ph/9511026
[14] J. Proos and C. Zalka, "Shor's discrete logarithm quantum algorithm for elliptic curves," *Quantum Inf. Comput.*, vol. 3, no. 4, pp. 317–344, 2003.
[15] D. Aharonov and O. Regev, "Lattice problems in NP ⋂ coNP," in *Proc. 45th Annu. IEEE Symp. Found. Comput. Sci.*, Oct. 2004, pp. 362–371.
[16] O. Regev, "On lattices, learning with errors, random linear codes, and cryptography," *J. ACM*, vol. 56, no. 6, 2005, Art. no. 34.
[17] V. Lyubashevsky, C. Peikert, and O. Regev, "On ideal lattices and learning with errors over rings," *J. ACM*, vol. 60, no. 6, p. 43, 2013.
[18] A. Abdallah. (2016). "Public key cryptography based on some extensions of group." [Online]. Available: https://arxiv.org/abs/1604.04474
[19] L. Wang, Z. Cao, P. Zeng, and X. Li, "One-more matching conjugate problem and security of braid-based signatures," in *Proc. ASIACCS*, Mar. 2007, pp. 295–301.
[20] L. Wang, Z. Cao, S. Zheng, X. Huang, and Y. Yang, "Transitive signatures from braid groups," in *Progress in Cryptology* (Lecture Notes in Computer Science). Berlin, Germany: Springer, 2007.
[21] H. Sibert, P. Dehornoy, and M. Girault, "Entity authentication schemes using braid word reduction," *Discrete Appl. Math.*, vol. 154, no. 2, pp. 420–436, 2006.
[22] G. K. Verma. (2008). *Blind Signature Scheme Over Braid Groups*. [Online]. Available: https://eprint.iacr.org/2008/027
[23] M. Kumar, "Linkability of blind signature schemes over braid groups," *IACR Cryptol. ePrint Arch.*, vol. 2009, p. 192, 2009.
[24] W. Yun, G. H. Xiong, X. K. Zhang, and W. S. Bao, "A strong blind signature scheme over braid groups," *IACR Cryptol. ePrint Arch.*, vol. 2009, p. 622, 2009.
[25] F. Li, A. Guo, and X. Zhao, "Proxy blind signature scheme based on braid group," *Appl. Res. Comput.*, vol. 27, no. 7, pp. 2641–2642, 2010.
[26] Y. Ren and M. Chang, "Threshold proxy blind multisignature based on braid groups," *J. Shanxi Normal Univ. Sci.*, vol. 25, no. 3, pp. 59–62, 2011.

[27] K. H. Ko, S. J. Lee, J. H. Cheon, J. W. Han, J.-S. Kang, and C. Park, "New public-key cryptosystem using braid groups," in *Proc. CRYPTO*, in Lecture Notes in Computer Science, vol. 1880. Berlin, Germany: Springer, 2000, pp. 166–183.
[28] E. Lee, S. J. Lee, and S. G. Hahn, "Pseudorandomness from braid groups," in *Proc. CRYPTO*, in Lecture Notes in Computer Science, vol. 2139. Berlin, Germany: Springer, 2001, pp. 486–502.
[29] K. Ko, D. H. Choi, M. S. Cho, and J. W. Lee. (2002). *New Signature Scheme Using Conjugacy Problem*. [Online]. Available: http://eprint.iacr.org/2002/168
[30] K. H. Ko, "Discussion on braid cryptography," Personal Communications, 2009.
[31] P. Dehornoy, "Braid-based cryptography," *Contemp. Math., Amer. Math. Soc.*, vol. 360, pp. 5–33, 2004.
[32] J. C. Cha, K. H. Ko, S. J. Lee, J. W. Han, and J. H. Cheon, "An efficient implementation of braid groups," in *Proc. ASIACRYPT* in (Lecture Notes in Computer Science), vol. 2248. Springer-Verlag, 2001, pp. 144–156.
[33] S. Maffre, "A weak key test for braid based cryptography," *Des., Codes Cryptogr.*, vol. 39, no. 3, pp. 347–373, 2006.
[34] K. H. Ko, J. W. Lee, and T. Thomas, "Towards generating secure keys for braid cryptography," *Des., Codes Cryptogr.*, vol. 45, no. 3, pp. 317–333, 2008.
[35] L. Wang, L. Wang, Z. Cao, Y. Yang, and X. Niu, "Conjugate adjoining problem in braid groups and new design of braid-based signatures," *Sci. China Inf. Sci.*, vol. 53, no. 3, pp. 524–536, 2010.
[36] M. Bellare, C. Namprempre, D. Pointcheval, and M. Semanko, "The one-more-RSA-inversion problems and the security of Chaum's blind signature scheme," *J. Cryptol.*, vol. 16, no. 3, pp. 185–215, 2003.

LICHENG WANG received the B.S. degree in engineering from Northwest Normal University, in 1995, the M.S. degree in mathematics from Nanjing University, in 2001, and the Ph.D. degree in engineering from Shanghai Jiaotong University, in 2007. He is currently an Associate Professor with the Beijing University of Posts and Telecommunications. His current research interests are cryptography, blockchain, and the future Internet architecture.

YANMEI TIAN received the B.S. degree from the Department of Mathematics, Qufu Normal University, in 2014. She is currently pursuing the Ph.D. degree with the Beijing University of Posts and Telecommunications. Her current research interests include network security and isogeny-based cryptography.

YUN PAN received the B.S. degree in engineering from Northwest Normal University, in 1995, the M.S. degree in engineering from Liaoning Shihua University, in 2001, and the Ph.D. degree in engineering from the China University of Mining Undefined Technology, Beijing, in 2003. She is currently a Full Professor with the Communication University of China. Her current research interests include network security, blockchain, and future Internet architecture.

YIXIAN YANG received the M.S. degree in applied mathematics and the Ph.D. degree in electronics and communication systems from the Beijing University of Posts and Telecommunications (BUPT), in 1986 and 1988, respectively, where he is currently a Yangtze River Scholar Program Professor. He majors in coding and cryptography, information and network security, signal and information processing. He was a recipient of the National Outstanding Youth Funding.