Journal of Universal Computer Science, vol. 25, no. 3 (2019), 294-308
submitted: 28/3/18, accepted: 30/12/18, appeared: 28/3/19 © J.UCS

A New Identification Scheme based on Syndrome Decoding Problem with Provable Security against Quantum Adversaries

Bagus Santoso
(University of Electro-Communications, Toyko, Japan
santoso.bagus@uec.ac.jp)

Chunhua Su
(Division of Computer Science, University of Aizu, Aizuwakamatsu, Japan
chsu@u-aizu.ac.jp)

Abstract: Recently, in order to guarantee security against quantum adversaries, several identification (ID) schemes based on computational problems which are supposed to be hard even for quantum computers have been proposed. However, their security are only proven against non-quantum adversaries. In this paper, we proposed a novel four-pass code-based identification scheme. By using quantum random oracle model, we provide a security proof for our scheme against quantum adversaries which aim to impersonate the prover under concurrent active attacks, based on the hardness assumption of syndrome decoding (SD) problem. Our security proof is interesting in its own right, since it only requires a non-programmable quantum random oracle, in contrast to existing security proofs of digital signatures generated from ID scheme via Fiat-Shamir transform which require programmable quantum random oracles.

Key Words: identification scheme, post-quantum, quantum random oracle, impersonation, concurrent active attacks
Category: E.3, C.2.0, D.4.6

1 Introduction

Identification scheme (ID) scheme is one of the most important cryptographic protocols. Not only that it is useful to protect a system against impersonation attacks, but it can be easily converted to construct essential building blocks for other cryptographic schemes. For example, we can convert an ID scheme into a trapdoor commitment scheme [Fischlin, 2001] or digital signature [Fiat and Shamir, 1986] to further achieve cloud security [Yu et al., 2017, Li et al., 2019], iot security [Yu et al., 2018b] and content security [Yu et al., 2018a]. Meanwhile, due to the rapid progress of the research on building quantum computers, the research on post-quantum cryptography, i.e., cryptographic schemes which remain secure even in the presence of adversaries with quantum computers, has been very active lately. It also includes the construction of ID schemes which remain secure even in the presence of quantum computers. Since it is well-known that quantum computers can easily break the discrete logarithm problem and solve integer factorization efficiently, we need to construct ID schemes based on computational problems which are still hard to solve even by using quantum computers. During the last decade, problems based on lattices, multivariate quadratic polynomials, and codes are considered as the most suitable computational problems for post-quantum cryptography.

Contribution and Main Idea

Stern proposed the first code-based ID schemes in his seminal paper in 1996 [Stern, 1996]. Santoso modified Stern's ID scheme such that the cheating probability of adversary per round decreases from 2/3 to 1/2. However, both ID schemes have not been proven secure against quantum adversaries. In this paper, we propose a modification of Santoso's ID Scheme and prove its security against quantum adversaries which try to impersonate honest provers under concurrently active attacks.

The most crucial part in our security proof against impersonation under concurrently active attacks is that we need to simulate honest provers in concurrent way without knowing the secret key. We prove the security of our scheme in *quantum random oracle (QRO) model*. We use the idea introduced by Unruh [Unruh, 2017] for constructing quantum random oracle, i.e., we select a hidden random univariate polynomial f with degree $2q_H$ to substitute the hash function in quantum random oracle model, where the total queries to hash function is upper-bounded by q_H. Using construction proposed by Unruh, [Unruh, 2017] we can compute the preimages of f using Berlekamp's algorithm [Berlekamp, 1971] efficiently. In the first pass of the interactive protocol, we require the verifier to send the hash value of its challenge which will be send later after the commitment. Since the hash function is simulated by QRO, which is f, inside the security proof, we can compute the preimage of f before sending the commitment and know the future value of challenge which verifier will send after receiving commitment. Since the ID scheme is such that one can response properly to the verifier without knowing secret key corresponding to the public verification key if one knows the challenge before hand, we can easily simulate the provers in concurrent way.

It should be noted that in the existing work of using quantum random oracle for proving the security of digital signatures against quantum adversaries, the programmability of quantum random oracle is necessary [Unruh, 2017]. However, we show in this paper that non-programmable quantum random oracle is sufficient to prove the security of ID scheme against quantum adversaries under concurrent active attacks.

2 Preliminaries

In this section, we provide the notations and definitions used throughout this paper and we recall the description of Stern's protocol.
NOTATIONS. The empty string is denoted by $\perp$. If x is a string, then $x \in \{0,1\}^n$ denotes that x is the n-bit binary string and if A is a matrix, then $A \in \{0,1\}^{m \times n}$ denotes that A is the binary matrix of m rows and n columns. Hamming weight of a string x, denoted by $\mathsf{hw}(x)$ is the number of 1s it includes and $\mathbb{S}_p^n$ is the set of n-bit binary strings of hamming weight p. Π_n denotes the set of permutations order n. The symbol $||$ denotes concatenation. A finite field with q elements is denoted by $\mathbb{F}_q$. In this paper, unless noted otherwise, any field is assumed to be the binary field $\mathbb{F}_2$. We use the notation $\{0,1\}$ and $\mathbb{F}_q$ interchangeably.

We say that any problem $\mathbb{P}$ is *hard* if there is no algorithm solves it within polynomial time with non-negligible probability. Unless noted otherwise, any algorithm is a probabilistic polynomial time algorithm. Also, unless noted otherwise, throughout this paper, an algorithm is considered as both classical and quantum algorithm.

Definition 1. (Syndrome Decoding (SD) Problem).A syndrome decoding problem is defined as follows.

Given: $v \in \mathbb{F}_2^m$, $H \in \mathbb{F}_2^{m \times n}$, $w \in \mathbb{N}$

Output: $s \in \mathbb{F}_2^n$ such that $Hs^{\mathrm{T}} = v$ and $\mathsf{hw}(s) = w$ hold for all $i \in [0, n]$.

The SD problem is said to be ε_{SD}-hard if there is no algorithm can solve the problem with success probability at least ε_{SD} within polynomial time. It is has been proven by Berlekamp et al. [Berlekamp et al., 1978] that SD problem is NP complete. Thus, the hardest case of SD problem is guaranteed to be hard even for quantum computers.

Definition 2. (Identification (ID) Scheme). An identification scheme ID is a tuple of algorithms, i.e., a setup parameter generator algorithm, a key-generation algorithm $\mathcal{K}_{\mathrm{gen}}$, a prover P and a verifier V which are defined as follows. A setup parameter generator takes input the security parameter and outputs setup parameter *param*. The key-generation algorithm $\mathcal{K}_{\mathrm{gen}}$ takes input setup parameter *param*, and outputs a public key and a secret key (pk, sk). A pair of algorithms (P, V) denotes an interactive protocol consisting of a prover P and a verifier V, where a common input is $(param, pk)$ and an auxiliary input of P is sk. After interactions, V outputs a bit as a verification result. Security against *impersonation under concurrent active attacks* considers an adversary whose goal is to impersonate an honest prover without the knowledge of the secret key, after the adversary is allowed to launch identification sessions with number of honest provers concurrently without any synchronization.

Definition 3. (Statistically Hiding and Computationally Binding String Commitment Scheme [Sakumoto et al., 2011]). The string commitment scheme com is a two-stage interactive protocol between a sender and a receiver using a string commitment function com. In the first stage, the sender computes a commitment value $c \leftarrow \mathsf{com}(s;\rho)$ and sends c to the receiver, where s is a string and ρ is a random string. In the second stage, the sender gives (s,ρ) to the receiver and the receiver verifies $c = \mathsf{com}(s;\rho)$. Informally, the string commitment scheme com is called statistically hiding if and only if no receiver can distinguish two commitment values generated from two different strings even if the receiver is computationally unbounded. And the string commitment scheme com is called computationally binding if and only if no polynomial time sender can change the committed string after the first phase.

Remark On Commitment. In this paper, we follow the style of Stern [Stern, 1996] and Sakumoto et al. [Sakumoto et al., 2011], where we omit the explicit handling of ρ, although the commitment is computed using an auxiliary random string ρ. In practice, ρ is chosen randomly with a sufficient length such that the statistical hiding is guaranteed. We abuse the exact notation of the commitment scheme with randomness by calling it *commitment function.* For a detailed concrete construction of a practical commitment scheme, please refer to [Stern, 1996, Sakumoto et al., 2011].

3 Proposed ID Scheme

The detailed description of one elementary round of our proposed ID scheme is shown in Fig. 1 (on the next page). We set the public key pk and secret key sk as follows.

$\mathsf{pk}:\ H \in \mathbb{F}_q^{m\times n}, v \in \mathbb{F}_q^m, w \in \mathbb{N}$, commitment function $\mathsf{com}(\cdot)$,
hash function $\mathsf{hash}(\cdot) : \{0,1\}^* \to \{0,1\}^{4n}$

$\mathsf{sk}:\ s \in \mathbb{F}_q^n$ such that $Hs^\top = v, \mathsf{hw}(s) = w$ holds.

Our proposed ID scheme is constructed based on the ID scheme proposed by Santoso [Santoso, 2018].

The full round ID scheme consists of the repetitions of the elementary round shown in Fig. 1 for ℓ times. The size of ℓ is closely related to the success probability of impersonation attack. The detailed relation between ℓ and the impersonation attack will be revealed shortly when we show the soundness property of the proposed ID scheme.

It is easy to see that an honest prover with the knowledge of s who is interacting honestly according to the protocol shown in Fig. 1 is always accepted by verifier.

0. Verifier Commitment: Verifier V picks randomly $\gamma, \overline{\gamma} \xleftarrow{\$} \mathbb{F}_2^{2n-1}$ and $b \xleftarrow{\$} \{0,1,2,3\}$. V sends $c_V = \mathsf{hash}(b_{[2]} \| \gamma \| \overline{\gamma})$ and $\overline{\gamma}$ to the prover, where $b_{[2]}$ denotes the representation of b in binary.

1. Prover Commitment: Prover P computes the followings:

$r_1, r_2 \xleftarrow{\$} \mathbb{F}_q^n, \sigma_1, \sigma_2 \xleftarrow{\$} \Pi_n$

$c_1 = \mathsf{com}(Hr_1^\top \oplus v \| \sigma_1)$, $c_2 = \mathsf{com}(r_1 \cdot \sigma_1)$, $c_3 = \mathsf{com}((r_1 \oplus s) \cdot \sigma_1)$,

$c_4 = \mathsf{com}(Hr_2^\top \oplus v \| \sigma_2)$, $c_5 = \mathsf{com}(r_2 \cdot \sigma_2)$, $c_6 = \mathsf{com}((r_2 \oplus s) \cdot \sigma_2)$,

$h_0 = \mathsf{hash}(r_1 \| \sigma_1 \| r_2 \oplus s \| \sigma_2)$, $h_1 = \mathsf{hash}(r_1 \oplus s \| \sigma_1 \| r_2 \| \sigma_2)$

$h_2 = \mathsf{hash}(r_1 \cdot \sigma_1 \| (r_1 \oplus s) \cdot \sigma_1 \| r_2 \| \sigma_2)$, $h_3 = \mathsf{hash}(r_1 \| \sigma_1 \| r_2 \cdot \sigma_2 \| (r_2 \oplus s) \cdot \sigma_2)$

Prover sends $\{c_i\}_{i=1}^6$ and $\{h_i\}_{i=0}^3$ to verifier V.

2. Challenge: V sends b and γ to P.

3. Response: Upon receiving b, Prover checks whether $c_V = \mathsf{hash}(b_{[2]} \| \gamma \| \overline{\gamma})$ holds. If it does not hold, P aborts. Otherwise, P computes z_0, z_1, z_2, z_3 as follows:

if $b = 0$: $z_0 = r_1$, $z_1 = \sigma_1$, $z_2 = r_2 \oplus s$, $z_3 = \sigma_2$
if $b = 1$: $z_0 = r_1 \oplus s$, $z_1 = \sigma_1$, $z_2 = r_2$, $z_3 = \sigma_2$
if $b = 2$: $z_0 = r_1 \cdot \sigma_1$, $z_1 = (r_1 \oplus s) \cdot \sigma_1$, $z_2 = r_2$, $z_3 = \sigma_2$
if $b = 3$: $z_0 = r_1$, $z_1 = \sigma_1$, $z_2 = r_2 \cdot \sigma_2$, $z_3 = (r_2 \oplus s) \cdot \sigma_2$

P sends (z_0, z_1, z_2, z_3) to verifier V.

4. Verify: Verifier V performs the verification procedure on (z_0, z_1, z_2, z_3) as follows $\mathsf{hash}(z_0 \| z_1 \| z_2 \| z_3) \stackrel{?}{=} h_b$.

if $b = 0$: $\mathsf{com}(Hz_0^\top \oplus v \| z_1) \stackrel{?}{=} c_1$, $\mathsf{com}(z_0 \cdot z_1) \stackrel{?}{=} c_2$,
$\mathsf{com}(Hz_2^\top \| z_3) \stackrel{?}{=} c_4$, $\mathsf{com}(z_2 \cdot z_3) \stackrel{?}{=} c_6$

if $b = 1$: $\mathsf{com}(Hz_0^\top \| z_1) \stackrel{?}{=} c_1$, $\mathsf{com}(z_0 \cdot z_1) \stackrel{?}{=} c_3$,
$\mathsf{com}(Hz_2^\top \oplus v \| z_3) \stackrel{?}{=} c_4$, $\mathsf{com}(z_2 \cdot z_3) \stackrel{?}{=} c_5$

if $b = 2$: $\mathsf{com}(z_0) \stackrel{?}{=} c_2$, $\mathsf{com}(z_1) \stackrel{?}{=} c_3$, $\mathsf{hw}(z_0 \oplus z_1) \stackrel{?}{=} w$,
$\mathsf{com}(Hz_2^\top \oplus v \| z_3) \stackrel{?}{=} c_4$, $\mathsf{com}(z_2 \cdot z_3) \stackrel{?}{=} c_5$

if $b = 3$: $\mathsf{com}(Hz_0^\top \oplus v \| z_1) \stackrel{?}{=} c_1$, $\mathsf{com}(z_0 \cdot z_1) \stackrel{?}{=} c_2$, $\mathsf{com}(z_2) \stackrel{?}{=} c_5$,
$\mathsf{com}(z_3) \stackrel{?}{=} c_6$, $\mathsf{hw}(z_2 \oplus z_3) \stackrel{?}{=} w$

V outputs a bit $acc \in \{0,1\}$ such that $acc = 1$ if responses from P satisfies all checking equations above, and $acc = 0$, otherwise.

Figure 1: One elementary round of (P, V) in proposed ID Scheme.

Theorem 4 (Completeness). *Any honest (true) prover P with the knowledge of secret key who is interacting honestly with an honest verifier V according to the protocol shown in Fig. 1 will always be accepted by the honest verifier V.*

Proof. We show that for any true prover P who performs the "Commitment" and "Response" step using the knowledge of secret key s such that $Hs^\top = v$ and $\mathsf{hw}(s) = w$, will always pass the checks performed by the honest verifier V who sends challenge $b \in \{0, 1, 2, 3\}$ to P in "Challenge" step. First, if P is an honest prover, based on the construction of commitments shown in step 1 of Fig. 1, we obtain the following equations.

$$\begin{aligned} &c_1 = \mathsf{com}(Hr_1^\top \oplus v\|\sigma_1), c_2 = \mathsf{com}(r_1 \cdot \sigma_1),\\ &c_3 = \mathsf{com}((r_1 \oplus s) \cdot \sigma_1), c_4 = \mathsf{com}(Hr_2^\top \oplus v\|\sigma_2),\\ &c_5 = \mathsf{com}(r_2 \cdot \sigma_2), c_6 = \mathsf{com}((r_2 \oplus s) \cdot \sigma_2) \end{aligned} \tag{1}$$

Now, let us check whether the response (z_0, z_2, z_2, z_3), computed by P upon receiving $b \in \{0, 1, 2, 3\}$ according the "Response" step in Fig. 1, will pass the verification procedure in "Verification" step described in Fig, 1.

Case $b = 0$: P computes the followings.

$$z_0 = r_1, z_1 = \sigma_1, z_2 = r_2 \oplus s, z_3 = \sigma_2. \tag{2}$$

And V performs the followings checks.

$$\begin{aligned} \mathsf{com}(Hz_0^\top \oplus v\|z_1) &\stackrel{?}{=} c_1, \mathsf{com}(z_1(z_0)) \stackrel{?}{=} c_2,\\ \mathsf{com}(Hz_2^\top\|z_3) &\stackrel{?}{=} c_4, \mathsf{com}(z_3(z_2)) \stackrel{?}{=} c_6 \end{aligned} \tag{3}$$

Using (2), we can transform the checking equations (3) into the followings.

$$\begin{aligned} &\mathsf{com}(Hr_1^\top \oplus v\|\sigma_1) \stackrel{?}{=} c_1, \mathsf{com}(\sigma_1(r_1)) \stackrel{?}{=} c_2,\\ &\mathsf{com}(H(r_2 \oplus s)^\top\|\sigma_2) = \mathsf{com}(Hr_2^\top \oplus v\|\sigma_2) \stackrel{?}{=} c_4,\\ &\mathsf{com}((r_2 \oplus s) \cdot \sigma_2) \stackrel{?}{=} c_6. \end{aligned}$$

Based on (1), it is easy to see that each equation in question above is equal.

Case $b = 1$: P computes the followings.

$$z_0 = r_1 \oplus s, z_1 = \sigma_1, z_2 = r_2, z_3 = \sigma_2 \tag{4}$$

And V performs the followings checks.

$$\begin{aligned} \mathsf{com}(Hz_0^\top\|z_1) &\stackrel{?}{=} c_1, \mathsf{com}(z_0 \cdot z_1) \stackrel{?}{=} c_3,\\ \mathsf{com}(Hz_2^\top \oplus v\|z_3) &\stackrel{?}{=} c_4, \mathsf{com}(z_2 \cdot z_3) \stackrel{?}{=} c_5 \end{aligned} \tag{5}$$

Using (4), we can transform the checking equations (5) into the followings.

$$\begin{aligned}
&\mathsf{com}(H(r_1 \oplus s)^\top \| \sigma_1) = \mathsf{com}(Hr_1^\top \oplus v \| \sigma_1) \stackrel{?}{=} c_1,\\
&\mathsf{com}((r_1 \oplus s) \cdot \sigma_1) \stackrel{?}{=} c_3,\\
&\mathsf{com}(Hz_2^\top \oplus v \| z_3) \stackrel{?}{=} c_4, \mathsf{com}(z_2 \cdot z_3) \stackrel{?}{=} c_5
\end{aligned}$$

Based on (1), it is easy to see that each equation in question above is equal.

Case $b = 2$: P computes the followings.

$$z_0 = r_1 \cdot \sigma_1, z_1 = (r_1 \oplus s) \cdot \sigma_1, z_2 = r_2, z_3 = \sigma_2 \tag{6}$$

And V performs the followings checks.

$$\begin{aligned}
&\mathsf{com}(z_0) \stackrel{?}{=} c_2, \mathsf{com}(z_1) \stackrel{?}{=} c_3, \mathsf{hw}(z_0 \oplus z_1) \stackrel{?}{=} w,\\
&\mathsf{com}(Hz_2^\top \oplus v \| z_3) \stackrel{?}{=} c_4, \mathsf{com}(z_2 \cdot z_3) \stackrel{?}{=} c_5
\end{aligned}$$

Using (6), we can transform the above checking equations into the followings.

$$\begin{aligned}
&\mathsf{com}(r_1 \cdot \sigma_1) \stackrel{?}{=} c_2, \mathsf{com}((r_1 \oplus s) \cdot \sigma_1) \stackrel{?}{=} c_3,\\
&\mathsf{hw}(r_1 \cdot \sigma_1 \oplus (r_1 \oplus s) \cdot \sigma_1) = \mathsf{hw}(s \cdot \sigma_1) = \mathsf{hw}(s) \stackrel{?}{=} w,\\
&\mathsf{com}(Hr_2^\top \oplus v \| \sigma_2) \stackrel{?}{=} c_4, \mathsf{com}(r_2 \cdot \sigma_2) \stackrel{?}{=} c_5
\end{aligned}$$

Based on (1), it is easy to see that each equation in question above is equal.

Case $b = 3$: P computes the followings.

$$z_0 = r_1, z_1 = \sigma_1, z_2 = r_2 \cdot \sigma_2, z_3 = (r_2 \oplus s) \cdot \sigma_2 \tag{7}$$

And V performs the followings checks.

$$\begin{aligned}
&\mathsf{com}(Hz_0^\top \oplus v \| z_1) \stackrel{?}{=} c_1, \mathsf{com}(z_0 \cdot z_1) \stackrel{?}{=} c_2,\\
&\mathsf{com}(z_2) \stackrel{?}{=} c_5, \mathsf{com}(z_3) \stackrel{?}{=} c_6, \mathsf{hw}(z_2 \oplus z_3) \stackrel{?}{=} w
\end{aligned} \tag{8}$$

Using (7), we can transform the checking equations (8) into the followings.

$$\begin{aligned}
&\mathsf{com}(Hr_1^\top \oplus v \| \sigma_1) \stackrel{?}{=} c_1, \mathsf{com}(\sigma_1(r_1)) \stackrel{?}{=} c_2,\\
&\mathsf{com}(r_2 \cdot \sigma_2) \stackrel{?}{=} c_5, \mathsf{com}((r_2 \oplus s) \cdot \sigma_2) \stackrel{?}{=} c_6,\\
&\mathsf{hw}(r_2 \cdot \sigma_2 \oplus (r_2 \oplus s) \cdot \sigma_2 = \mathsf{hw}(s \cdot \sigma_2) = \mathsf{hw}(s) \stackrel{?}{=} w
\end{aligned}$$

Based on (1), it is easy to see that each equation in question above is equal.

Finally, one can easily see that for each b, the true prover P always constructs h_b such that $h_b = \mathsf{hash}(z_0 \| z_1 \| z_2 \| z_3)$ holds, where z_0, z_1, z_2, z_3 are the responses corresponding to b as indicated above. Hence, we have shown here that the true prover P is always accepted by the honest verifier V. This completes the proof of Theorem 4. □

Definition 5 (Accepting Transcript). A tuple (c, b, z) is said to be an accepting transcript if and only if c is a valid commitment produced by an honest prover, b is a challenge sent by an honest verifier, z is the response from prover to the challenge, and (c, b, z) passes all checks performed by an honest verifier.

Remark. One can easily see in Fig. 1 that actually, a full communication transcript of one elementary round includes not only commitment c, challenge b, and response z, but also includes the hash values c_V, h_0, h_1, h_2, h_3, γ and $\overline{\gamma}$. However, since in this paper we care more about the issue whether the party in the prover side is acting or able to act as a true prover, in this paper we often omit c_V, γ, $\overline{\gamma}$ when we discuss about the transcript. Also, we often omit h_0, h_1, h_2, h_3 since only h_j where $j = b$ is checked by the verifier and h_j can be easily computed from a valid response of the prover. Other hash values can be generated arbitrarily since they are not checked by the verifier.

4 Security Proof

In this section, we will build step by step the security proof for our proposed scheme against quantum adversaries which launch impersonation under concurrent active attacks.

Theorem 6 (Special Soundness). *If the commitment function* com *is computationally binding, then there is an efficient procedure to extract* s *such that* $Hs^\top = v$ *and* $\mathsf{hw}(s) = w$ *given: (1) a set of valid commitments* c *(such as the one created by an honest prover in the "Prover Commitment" step), (2) a tuple of three distinct challenges* $(b^{(1)}, b^{(2)}, b^{(3)})$ *such that* $b^{(i)} \neq b^{(j)}$ *for any* $i, j, i \neq j$*, and (2) a set of three correct prover's responses* $z^{(b^{(1)})}, z^{(b^{(2)})}, z^{(b^{(3)})}$ *such that for each* $i \in \{1, 2, 3\}$*,* $(c, b^{(i)}, z^{(b^{(i)})})$ *is an accepting transcript.*

Proof. Since the challenge sent by the verifier is selected from the set $\{0, 1, 2, 3\}$, we have four possible combinations of three (distinct) challenges $(b^{(1)}, b^{(2)}, b^{(3)})$, where $b^{(i)} \in \{0, 1, 2, 3\}$, $b^{(i)} \neq b^{(j)}$, for any distinct (i, j). It is sufficient to show that from the responses of each challenge combination, we can compute s described above efficiently. Since we assume that com is computationally binding, we can assume that if $\mathsf{com}(x) = \mathsf{com}(y)$ for some x, y, then $x = y$ must hold.

1. **Case** $(b^{(1)}, b^{(2)}, b^{(3)}) = (0, 1, 2)$
From the head part of c_1 we have $Hz_0^{(0)\top} \oplus v = Hz_0^{(1)\top}$. Then, the following must hold:

$$v = H(z_0^{(0)} \oplus z_0^{(1)})^\top. \tag{9}$$

From the tail part of c_1 we have $z_1^{(0)} = z_1^{(1)}$. Then, we can set $\sigma = z_1^{(0)} = z_1^{(1)}$. Combine this with c_2 and c_3, we obtain $z_0^{(2)} = z_0^{(0)} \cdot \sigma$ and $z_1^{(2)} = z_0^{(1)} \cdot \sigma$. Then, we obtain as follows.

$$\begin{aligned} \mathsf{hw}(z_0^{(0)} \oplus z_0^{(1)}) &= \mathsf{hw}((z_0^{(0)} \oplus z_0^{(1)}) \cdot \sigma) \\ &= \mathsf{hw}(z_0^{(0)} \cdot \sigma \oplus z_0^{(1)} \cdot \sigma) \\ &= \mathsf{hw}(z_0^{(2)} \oplus z_1^{(2)}) = w. \end{aligned} \tag{10}$$

The last equation comes from the property which holds if all verifications for the case $b^{(3)} = 2$ are correct. Hence, from Eq. (9) and Eq. (10), we can conclude that for this case, we can set $s = z_0^{(0)} \oplus z_0^{(1)}$.

2. **Case** $(b^{(1)}, b^{(2)}, b^{(3)}) = (0, 1, 3)$
From the head part of c_4 we have $Hz_2^{(1)\top} \oplus v = Hz_2^{(0)\top}$. Then, the following must hold:

$$v = H(z_2^{(0)} \oplus z_2^{(1)})^\top. \tag{11}$$

From the tail part of c_4 we have $z_3^{(0)} = z_3^{(1)}$. Then, we can set $\sigma = z_3^{(0)} = z_3^{(1)}$. Combine this with c_5 and c_6, we obtain $z_3^{(3)} = z_2^{(0)} \cdot \sigma$ and $z_2^{(3)} = z_2^{(1)} \cdot \sigma$. Then, we obtain as follows.

$$\begin{aligned} \mathsf{hw}(z_2^{(1)} \oplus z_2^{(0)}) &= \mathsf{hw}((z_2^{(1)} \oplus z_2^{(0)}) \cdot \sigma) \\ &= \mathsf{hw}(z_2^{(1)} \cdot \sigma \oplus z_2^{(0)} \cdot \sigma) \\ &= \mathsf{hw}(z_2^{(3)} \oplus z_3^{(3)}) = w. \end{aligned} \tag{12}$$

The last equation comes from the property which holds if all verifications for the case $b^{(3)} = 3$ are correct. Hence, from Eq. (11) and Eq. (12), we can conclude that for this case, we can set $s = z_2^{(0)} \oplus z_2^{(1)}$.

3. **Case** $(b^{(1)}, b^{(2)}, b^{(3)}) = (0, 2, 3)$
From the head part of c_4 we have $Hz_2^{(2)\top} \oplus v = Hz_2^{(0)\top}$. Then, the following must hold:

$$v = H(z_2^{(0)} \oplus z_2^{(2)})^\top. \tag{13}$$

From the tail part of c_4 we have $z_3^{(0)} = z_3^{(2)}$. Then, we can set $\sigma = z_3^{(0)} = z_3^{(2)}$. Combine this with c_5 and c_6, we obtain $z_3^{(3)} = z_2^{(2)} \cdot \sigma$ and $z_2^{(3)} = z_2^{(0)} \cdot \sigma$.

Then, we obtain as follows.

$$\begin{aligned}\mathsf{hw}(z_2^{(2)} \oplus z_2^{(0)}) &= \mathsf{hw}((z_2^{(2)} \oplus z_2^{(0)}) \cdot \sigma) \\ &= \mathsf{hw}(z_2^{(2)} \cdot \sigma \oplus z_2^{(0)} \cdot \sigma) \\ &= \mathsf{hw}(z_2^{(3)} \oplus z_3^{(3)}) = w.\end{aligned} \tag{14}$$

The last equation comes from the property which holds if all verifications for the case $b^{(3)} = 3$ are correct. Hence, from Eq. (13) and Eq. (14), we can conclude that for this case, we can set $s = z_2^{(0)} \oplus z_2^{(2)}$.

4. **Case** $(b^{(1)}, b^{(2)}, b^{(3)}) = (1, 2, 3)$
   From the head part of c_1 we have $Hz_0^{(3)\top} \oplus v = Hz_0^{(1)\top}$. Then, the following must hold:

   $$v = H(z_0^{(3)} \oplus z_0^{(1)})^\top. \tag{15}$$

   From the tail part of c_1 we have $z_1^{(3)} = z_1^{(1)}$. Then, we can set $\sigma = z_1^{(3)} = z_1^{(1)}$. Combine this with c_2 and c_3, we obtain $z_0^{(2)} = z_0^{(3)} \cdot \sigma$ and $z_1^{(2)} = z_0^{(1)} \cdot \sigma$. Then, we obtain as follows.

   $$\begin{aligned}\mathsf{hw}(z_0^{(3)} \oplus z_0^{(1)}) &= \mathsf{hw}((z_0^{(3)} \oplus z_0^{(1)}) \cdot \sigma) \\ &= \mathsf{hw}(z_0^{(2)} \oplus z_1^{(2)}) = w.\end{aligned} \tag{16}$$

   The last equation comes from the property which holds if all verifications for the case $b^{(2)} = 2$ are correct. Hence, from Eq. (15) and Eq. (16), we can conclude that for this case, we can set $s = z_0^{(3)} \oplus z_0^{(1)}$.

Hence, we have shown that from a valid commitment with any combination of three distinct challenges and the corresponding valid responses, we can extract s as described above. This completes the proof of Theorem 6. □

We need the following lemma to guarantee that we can simulate the provers without secret key in our security proof against impersonation with concurrent active attacks.

Lemma 7 (Simulatability). *If* com *is statistically hiding, there exists an algorithm* $\mathcal{M}$ *such that given the public key and the challenge, produces an accepting transcript of one round identification.*

Proof. For each challenge $b' \in \{0, 1, 2, 3\}$, $\mathcal{M}$ proceeds as follows.

1. **Case** $b' = 0$: Choose $r_1, r_2' \xleftarrow{\$} \mathbb{F}_q^n$, $\sigma_1, \sigma_2 \xleftarrow{\$} \Pi_q^n$, then set responses $z_0 = r_1, z_1 = \sigma_1, z_2 = r_2', z_3 = \sigma_2$, and commitments $c_1 = \mathsf{com}(Hz_0^\top \oplus v \| z_1)$, $c_2 = \mathsf{com}(z_0 \cdot z_1)$, $c_4 = \mathsf{com}(Hz_2^\top \| z_3)$, $c_6 = \mathsf{com}(z_2 \cdot z_3)$.

2. **Case** $b' = 1$: Choose $r_1', r_2 \xleftarrow{\$} \mathbb{F}_q^n$, $\sigma_1, \sigma_2 \xleftarrow{\$} \Pi_q^n$, then set responses $z_0 = r_1', z_1 = \sigma_1, z_2 = r_2, z_3 = \sigma_2$, and commitments $c_1 = \mathsf{com}(Hz_0^\top \oplus v \| z_1)$, $c_3 = \mathsf{com}(z_0 \cdot z_1)$, $c_4 = \mathsf{com}(Hz_2^\top \oplus v \| z_3)$, $c_5 = \mathsf{com}(z_2 \cdot z_3)$.

3. **Case** $b' = 2$: Choose $r_1, s', r_2 \xleftarrow{\$} \mathbb{F}_q^n$ s.t. $\mathsf{hw}(s') = w$, $\sigma_1, \sigma_2 \xleftarrow{\$} \Pi_q^n$, then set responses $z_0 = r_1 \cdot \sigma_1, z_1 = (r_1 \oplus s') \cdot \sigma_1, z_2 = r_2, z_3 = \sigma_2$, and commitments $c_2 = \mathsf{com}(z_0)$, $c_3 = \mathsf{com}(z_1)$, $c_4 = \mathsf{com}(Hz_2^\top \oplus v \| z_3)$, $c_6 = \mathsf{com}(z_2 \cdot z_3)$.

4. **Case** $b' = 3$: Choose $r_1, s' r_2 \xleftarrow{\$} \mathbb{F}_q^n$ s.t. $\mathsf{hw}(s') = w$, $\sigma_1, \sigma_2 \xleftarrow{\$} \Pi_q^n$, then set responses $z_0 = r_1, z_1 = \sigma_1, z_2 = r_2 \cdot \sigma_2, z_3 = (r_2 \oplus s')\sigma_2$, and commitments $c_1 = \mathsf{com}(Hz_0^\top \oplus v \| z_3)$, $c_2 = \mathsf{com}(z_0 \cdot z_1)$, $c_5 = \mathsf{com}(z_2)$, $c_6 = \mathsf{com}(z_3)$.

Using the fact that in an accepting transcript $h_{b'} = \mathsf{hash}(z_0 \| z_1 \| z_2 \| z_3)$ holds, $\mathcal{M}$ can easily produce the correct $h_{b'}$ for each b'. For other hash values $h_{i_1}, h_{i_2}, h_{i_3}$ where $i_j \neq b'$ for $j = 1, 2, 3$, $\mathcal{M}$ can pick randomly $\mu_1, \mu_2, \mu_3 \in \{0, 1\}^{4n}$ and set $h_{i_j} = \mu_j$ for $j = 1, 2, 3$. For each case of b', commitments which are not explicitly mentioned above can be constructed easily by putting random values as the inputs since they are not checked during verification. Since com is statistically hiding, the inputs can be considered completely hidden from any algorithm. □

Theorem 8 (Security against Concurrent Active Attacks). *Let an algorithm $\mathcal{A}$ be an adversary which impersonates a prover in the ID scheme shown in Fig. 1 under concurrent active attacks with probability $\varepsilon_{\mathcal{A}}$ by launching separate q_{h} "quantum" hash queries and interacting with k honest provers concurrently. If SD problem is ε_{SD}-hard, then the following holds.*

$$\varepsilon_{\mathcal{A}} - 1/2^\ell - 2k\ell q_H / 2^{2n-1} \leqq \varepsilon_{SD} \tag{17}$$

Proof. We will construct an algorithm $\mathcal{B}$ which breaks SD problem using $\mathcal{A}$ described above. Let $H \in \mathbb{F}_2^{m \times n}$, $v \in \mathbb{F}_2^m$, $w \in \mathbb{N}$ be the inputs to $\mathcal{B}$. $\mathcal{B}$ sets H, v and w as the public keys of the ID scheme for $\mathcal{A}$.

Quantum Random Oracle. Based on technique introduced by Unruh [Unruh, 2017], we will use *quantum random oracle* (QRO) to substitute hash in our proof. First, we randomly select a univariate polynomial $f(x)$ from $\mathbb{F}_{2^{4n}}[x]$ with degree $2q_{\mathsf{h}}$. Then, we use $f(x)$ as the core function inside QRO. Precisely, for any (superpositioned) quantum hash queries $|\psi\rangle = \sum_{x \in \mathbb{F}_{2^{4n}}} \alpha_x |x\rangle$ sent to QRO, QRO will response with $\sum_x \alpha_x |x, f(x)\rangle$. According to Unruh [Unruh, 2017], no algorithm (including quantum algorithm) will be able to distinguish QRO constructed such as above from a true QRO which picks randomly from uniform distribution.

Simulation of Provers. To simulate the provers *concurrently* for $\mathcal{A}$, for each interaction request from $\mathcal{A}$, $\mathcal{B}$ proceeds as follows:

(1) Upon receiving c_V from $\mathcal{A}$, $\mathcal{B}$ computes the set of preimages of c_V, i.e., $f^{-1}(c_V)$, using algorithm in [Ben-Or, 1981] or [Berlekamp, 1971].

(2) Find $\mu \in f^{-1}(c_V)$ such that the least $(2n-1)$ significant bits of μ equals to γ. If $f^{-1}(c_V)$ contains more than one candidate of μ with such property, select the one found first.

(3) Once we got such μ, $\mathcal{B}$ sets b' equals to the most two significant bits of μ.

(4) Using the "simulation strategies" shown in the proof of Lemma 7, $\mathcal{B}$ construct the commitment and responses corresponding to b' and sends the commitment to $\mathcal{A}$.

(5) After receiving b from $\mathcal{A}$, $\mathcal{B}$ checks whether $b = b'$ holds. If $b \neq b'$, $\mathcal{B}$ aborts. Otherwise, $\mathcal{B}$ sends the corresponding response to $\mathcal{A}$.

Note that the event that $b \neq b'$ happens *only if* the set of preimages of c_V, i.e., $f^{-1}(c_V)$, contains more than one member with the property that that the least $(2n-1)$ significant bits equals to $\overline{\gamma}$. Since Unruh [Unruh, 2017] has shown that f is picked such that it is distinguishable from a random function, we can assume that the least $(2n-1)$ significant bits of preimages are distributed uniformly. This means that the probability that the least $(2n-1)$ significant bits of a member of any preimage set equals to $\overline{\gamma}$ is $2^{-(2n-1)}$. Meanwhile, since f is a $2q_H$ degree polynomial, for each c_V, the size of the set of preimages $f^{-1}(c_V)$ is at most $2q_H$. Hence, for each single c_V, the probability that $f^{-1}(c_V)$ contains members with the same $(2n-1)$ least significant bits is at most $2q_H$. Note that there are at most $k \times \ell$ interactions between the prover simulated by $\mathcal{B}$ and the adversarial verifier $\mathcal{A}$. Therefore, the total probability that $b \neq b'$ occurs in some interactions is at most $k\ell \times 2q_H/2^{2n-1}$.
Extraction of Secret keys. After $\mathcal{A}$ stops all interactions with provers (which are simulated by $\mathcal{B}$), $\mathcal{B}$ starts a new session of identification with $\mathcal{A}$. In this session, $\mathcal{B}$ acts as the verifier and let $\mathcal{A}$ to act as a prover. First, $\mathcal{B}$ picks randomly $\gamma, \overline{\gamma} \xleftarrow{\$} \mathbb{F}_2^{2n-1}$ and $b \xleftarrow{\$} \{0,1,2,3\}$, then computes $c_V = \mathsf{hash}(b_{[2]} \| \gamma \| \overline{\gamma})$, where $b_{[2]}$ is the representation of b in $\mathbb{F}_2^2$. $\mathcal{B}$ sends c_V and $\overline{\gamma}$ to $\mathcal{A}$. After $\mathcal{A}$ sends the commitment, $\mathcal{B}$ sends back b, retrieves the responses z_0, z_1, z_2, z_3 from $\mathcal{A}$, and checks their validity according to the description of the ID scheme in Fig. 1. Then $\mathcal{B}$ selects two out of $\{h_j\}_{j=0,1,2,3}$ where $j \neq b$. Let those be denoted by h_{j_1} and h_{j_2}. $\mathcal{B}$ computes the set of preimages $f^{-1}(h_{j_1})$ and $f^{-1}(h_{j_2})$. After parsing each element of the preimages into $z_0^{(j_1)} \| z_1^{(j_1)} \| z_2^{(j_1)} \| z_3^{(j_1)}$ and $z_0^{(j_2)} \| z_1^{(j_2)} \| z_2^{(j_2)} \| z_3^{(j_2)}$ respectively according to the sets, by using the same procedure shown in the proof of Theorem 6 on special soundness, $\mathcal{B}$ can extract easily s' such that $Hs'^{\top} = v$ and $\mathsf{hw}(s') = w$ hold.

Note that $\mathcal{B}$ can successfully extract the secret key with the above procedure only if we get at least three out of h_0, h_1, h_2, h_3 "properly" created. Here, a hash value h_j ($j \in \{0,1,2,3\}$) is said to be "properly" created if the set of preimages $f^{-1}(h_j)$ contains a member which can be parsed into $z_0^{(j)} \| z_1^{(j)} \| z_2^{(j)} \| z_3^{(j)}$ such that $z^{(j)} = (z_0^{(j)}, z_1^{(j)}, z_2^{(j)}, z_3^{(j)})$ is a valid response for the challenge j and corresponding commitments. Note that here we do not need to care about the failure of extracting secret key when $\mathcal{A}$ is fail to act as a true prover, i.e., unable to give a valid response which passes the verification check. We only need to care about the probability of fail on extracting the secret key when $\mathcal{A}$ is successfully giving valid responses. Let us look into the following cases.

Case 1: If $\mathcal{A}$ "properly" creates three out of h_0, h_1, h_2, h_3, if $\mathcal{A}$ gives a valid response, $\mathcal{B}$ will always be able to extract the keys using the above procedure.

Case 2: If $\mathcal{A}$ only "properly" creates two out of h_0, h_1, h_2, h_3, then the probability that $\mathcal{B}$ sends the challenge b such that h_b is properly created is $2/4 = 1/2$. Note that here $\mathcal{B}$ can not extract the keys. Thus, the probability of failure on extracting the keys in a single elementary round of identification is upper-bounded by $1/2$. And the probability of failure on extracting the keys in a full identification round consisting of ℓ repetition of elementary rounds is upper-bounded by $1/2^\ell$.

Case 3: If $\mathcal{A}$ only "properly" creates one out of h_0, h_1, h_2, h_3, then the probability that $\mathcal{B}$ sends the challenge b such that h_b is properly created is $1/4$. Note that here $\mathcal{B}$ can not extract the keys. The probability of failure on extracting the keys in single elementary round is $1/4$. This probability is smaller than the one in **Case 2** above.

From the explanation above, it is easy to see that the probability of $\mathcal{B}$ failing to extract the keys in a full identification round is upper-bounded by $1/2^\ell$.

Hence, since the success probability of $\mathcal{A}$ impersonating the prover is $\varepsilon_{\mathcal{A}}$, the total success probability of $\mathcal{B}$ simulating the provers for $\mathcal{A}$ and extracting the keys from $\mathcal{A}$ is at least $\varepsilon_{\mathcal{A}} - 1/2^\ell - 2k\ell q_H/2^{2n-1}$. Meanwhile, since $\mathcal{B}$ itself is an SD solver algorithm, by assumption, the success probability of $\mathcal{B}$ is upper-bounded by ε_{SD}. This ends the proof of Theorem 8. $\square$

5 Conclusions and Future Directions

We have shown a new construction of code-based identification (ID) scheme with provable security against quantum adversaries which launch impersonation under concurrent active attacks. Our security proof is based on the (quantum) hardness of syndrome decoding (SD) problem and uses the quantum random oracle model. We instantiated the quantum random oracle using a randomly picked univariate polynomial with a certain degree. The paradigm we apply to transform the canonical three-pass ID scheme into four-pass ID scheme with security against impersonation by quantum adversaries is independent from the inner procedure of the original ID scheme and SD problem. Therefore, as a final note, we conjecture that our paradigm can be applied to any canonical ID scheme based on computational problems which are supposed to be secure against quantum adversaries (e.g., lattices, multivariate quadratic polynomials, isogeny) to construct a new ID scheme with provable security against quantum adversaries. We leave the proof of our conjecture as the open problem for future works.

Acknowledgements

Bagus Santoso is supported by JSPS Kiban (B) 18H01438 and JPSPS Kiban(C) 18K11292. Chunhua Su is supported by JSPS Kiban(B) 18H03240 and JSPS Kiban(C) 18K11298.

References

[Ben-Or, 1981] Ben-Or, M. (1981). Probabilistic algorithms in finite fields. In 22nd Annual Symposium on Foundations of Computer Science (sfcs 1981), pages 394–398.

[Berlekamp et al., 1978] Berlekamp, E., McEliece, R., and Van Tilborg, H. (1978). On the inherent intractability of certain coding problems (corresp.). IEEE Transactions on Information Theory, 24(3):384–386.

[Berlekamp, 1971] Berlekamp, E. R. (1971). Factoring polynomials over large finite fields*. In Proceedings of the Second ACM Symposium on Symbolic and Algebraic Manipulation, SYMSAC '71, pages 223–, New York, NY, USA. ACM.

[Fiat and Shamir, 1986] Fiat, A. and Shamir, A. (1986). How to prove yourself: Practical solutions to identification and signature problems. In CRYPTO '86, volume 263 of Lecture Notes in Computer Science, pages 186–194. Springer.

[Fischlin, 2001] Fischlin, M. (2001). Trapdoor Commitment Schemes and Their Applications. PhD thesis, Johann Wolfgang Goethe-Universität.

[Li et al., 2019] Li, Y., Yu, Y., Min, G., Susilo, W., Ni, J., and Choo, K. (2019). Fuzzy identity-based data integrity auditing for reliable cloud storage systems. IEEE Trans. Dependable Sec. Comput., 16(1):72–83.

[Sakumoto et al., 2011] Sakumoto, K., Shirai, T., and Hiwatari, H. (2011). Public-key identification schemes based on multivariate quadratic polynomials. In CRYPTO 2011, volume 6841 of LNCS, pages 706–723. Springer.

[Santoso, 2018] Santoso, B. (2018). A new three-pass code-based zero-knowledge identification scheme with cheating probability of exactly half. In International Symposium on Information Theory and Its Applications. IEICE.

[Stern, 1996] Stern, J. (1996). A new paradigm for public key identification. IEEE Trans. Information Theory, 42(6):1757–1768.

[Unruh, 2017] Unruh, D. (2017). Post-quantum security of fiat-shamir. In ASIACRYPT (1), volume 10624 of Lecture Notes in Computer Science, pages 65–95. Springer.

[Yu et al., 2017] Yu, Y., Au, M., Ateniese, G., Huang, X., Susilo, W., Dai, Y., and Min, G. (2017). Identity-based remote data integrity checking with perfect data privacy preserving for cloud storage. IEEE Trans. Information Forensics and Security, 12(4):767–778.

[Yu et al., 2018a] Yu, Y., Li, Y., Du, X., Chen, R., and Yang, B. (2018a). Content protection in named data networking: Challenges and potential solutions. IEEE Communications Magazine, 56(11):82–87.

[Yu et al., 2018b] Yu, Y., Li, Y., Tian, J., and Liu, J. (2018b). Blockchain-based solutions to security and privacy issues in the internet of things. IEEE Wireless Commun., 25(6):12–18.