# More Short Signatures without Random Oracles

Victor K. Wei and Tsz Hon Yuen

Dept. of Information Engineering, The Chinese Univ. of Hong Kong, Hong Kong
{kwwei,thyuen4}@ie.cuhk.edu.hk

December 20, 2005

**Abstract.** We construct three new signatures and prove their securities without random oracles. They are motivated, respectively, by Boneh and Boyen [6]'s, Zhang, et al. [35]'s, and Camenisch and Lysyanskaya [10]'s signatures without random oracles. The first two of our signatures are shorter than [6, 35]'s state-of-the-art short signatures by 22%, with essentially the same complexity and the exact security. Our third signature is reducible to a modified LRSW Assumption [25] which is without the sequential-one-more external signing oracle used in [10]. New and interesting variants of the $q$-SDH Assumption, the $q$-SR (Square Root) Assumption, and the LRSW Assumption are presented. New and independently interesting proof techniques extending the two-mode technique of [6] are used, including a combined three-mode simulation and rewinding in the standard model.

## 1 Introduction

The random oracle has been a popular technique in provable security before and after its formal introduction by Bellare and Rogaway [3]. The results of [15, 16, 30] used rewindings of hashings with observable hashing input-output pairs. The Schnorr signature and many other signatures and Poofs-of-Knowledge (PoK's) results [4, 5] used the Fiat-Shamir paradigm in their reductionist security proofs [27, 18, 26]. The random oracle rewinding technique [32, 31] is a particularly powerful proof technique.

Recently, the results of Barak, et al. [1, 2] and Goldwasser and Kalai [22] proved the insecurity of the random oracle model as it is commonly used in the Fiat-Shamir paradigm. The core contradiction is in the *predictability* of the random oracle, how much can the hash outputs be predicted based on prior computation transcripts. On one hand, proofs in the random oracle model for the Fiat-Shamir paradigm depends on this predictability to simulate the signing oracle. On the other hand, too much predictability enables the attackers to forge. [1, 22] were able to formalize the notion of *predictability* and prove that zero-knowledge cannot exist in the Fiat-Shamir paradigm for a very wide range of real-world hashing families. [2] proceeded to define an essentially necessary and sufficient condition for the existence of real-world hashing families that will enable zero-knowledge proofs in the Fiat-Shamir paradigm. However, [2] expressed pessimism of the construction of such qualified hashing families.

The research on signatures whose reductionist security proofs do no use random oracles has had a long history, and it received renewed vigor since the insecurity proof of the random oracles [1, 22, 2]. The signatures without random oracles in [20, 21, 29, 14, 11, 12, 17, 24, 9] contained various inefficiencies. See [9]'s Table 1 for a good summary.

Cramer and Shoup [13] presented three signatures which achieved good efficiency in $O(\lambda_s)$-bit signature length, $O(\lambda_s)$-bit public key length, servicing any number of Signing Oracle queries, and supporting the generation of any number of signatures in the Real World. Its existential unforgeability against adaptive-chosen-plaintext attackers (ACP-UF) is reducible

to the Strong RSA Assumption. The signature consists of three elements from $Z_N$, where the RSA modulus $N$ is 1024 (resp. 2048) bits for security level $\lambda_s = 128$ (resp. 256) bits, resulting in 3072-bit (resp. 6144-bit) signatures.

Boneh and Boyen [6] presented short signatures whose ACP-UF is reducible to the $q$-SDH (Strong Diffie-Hellman) Assumption. Let $\hat{\mathbf{e}} : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_3$, order$(\mathbb{G}_1) = q_1$. The signature of [6] consists of an element from $Z_{q_1}$ and an element from $\mathbb{G}_1$. According to high-security pairing parameters suggested by Koblitz and Menezes [23], $\lambda_s = 128$ (resp. 256), a $\mathbb{G}_1$ element is $(5/2)\lambda_s$ bits, $q_1$ is $2\lambda_s$ bits and [6]'s signature is 576 (resp. 1152) bits long.

Zhang, Chen, Susilo, and Mu [35] presented short signatures whose ACP-UF is reducible to the $q$-SR (Square Root) Assumption. The signature length is the same as that of [6], ranging from 576 to 1152 bits.

Camenisch and Lysyanskaya [10] presented three short signatures without random oracles and reduced their ACP-UF to the LRSW Assumption [25]. Their signature lengths are higher, around $(29/2)\lambda_s$ bits.

The proofs of [6, 35] are in the standard model, except the attacker must pre-announce the maximum number of Signing Oracle queries it will make. The proof of [10] is in the standard model, except that it assumes the availability of an external (hypothesized) sequential-one-more Signing Oracle to the Simulator. This requirement may be considered undesirable in some considerations. However, the security plausible even if the attacker has a Known-Target Discrete Logarithm Collision Oracle.

The signatures of [6, 35] also remain plausible against an attacker in possession of a Known-Target Discrete Logarithm Collision Oracle. The signature of Boneh, Lynn, and Shacham [8] can be proven ACP-UF given the hypothesis that the Simulator has an external sequential-one-more Signing Oracle similar to [10]. But it is broken if the attacker has a Known-Target Discrete Logarithm Collision Oracle.

Our **Contributions** are

1. We construct three new signatures without random oracles, i.e. the correctness and the existential unforgeability against adaptive-chosen-plaintext attackers (ACP-UF) of each is reducible to intractability assumptions without random oracles. The proof for each signature is in the standard model except the attacker pre-announces the maximum number of Signing Oracle queries it will make, just like Boneh and Boyen [6] and Zhang, Chen, Susilo, abd Mu [35].
2. Our three signatures are respectively motivated by the short signatures without random oracles in Boneh and Boyen [6], in Zhang, Chen, Susilo, and Mu [35], and in Camenisch and Lysyanskaya [10]. The security of our three signatures are respectively reducible to the $q$-SDH' Assumption which is a slight alteration of the $q$-SDH Assumption [28, 34, 6], the $(q, \ell)$-SR (Square Root) Assumption which is modified from [35]'s $q$-SR Assumption, and the $q$-wholesale LRSW Assumption which is modified from the LRSW Assumption [25]. These new assumptions are interesting in their own rights.
3. The first two of our new signatures without random oracles are 22% shorter than the state-of-the-art short signatures without random oracles from [6] and from [35]. Furthermore, the complexity and exact security of our first signature is essentially the same as that of [6].
4. Our third new signature is a modification of [10]'s Signature B. We improve the signature such that its security is proved without the external Signing Oracle used in the LRSW Assumption. Our signature is provably secure in the standard model except that the

attacker must pre-announce the maximum number of Signing Oracle queries just like [6, 35] but not like [10, 25].
5. During our proofs, we introduce new proof techniques which extend Boneh and Boyen [6]'s two-mode proof technique. For example, we introduce a proof technique which combines three-mode, and rewind simulation in the standard model. These new proof techniques are powerful and are interesting in their own right.

## 2 Security Model

We review security models [6, 18] for signatures.

**Syntax**: A signature is a tuple $(\mathsf{KGen}, \mathsf{Sign}, \mathsf{Vf})$ where

- Protocol $\mathsf{KGen}$ accepts input the security parameter $1^{\lambda_s}$, outputs system parameters $\mathsf{param}$, and sk-pk pair $(\mathsf{sk}, \mathsf{pk})$.
- Protocol $\mathsf{Sign}$ accepts inputs message $m$ and secrete key $\mathsf{sk}$, outputs a signature $\sigma$.
- Protocol $\mathsf{Vf}$ accepts inputs a message $m$, a signature $\sigma$, and a public key $\mathsf{pk}$, outputs 1 or 0 for $\mathsf{valid}$ or $\mathsf{invalid}$.

**Definition 1.** *(*Correctness*) A signature is* correct *if, for arbitrary message $m$, we have*

$$\Pr[\mathit{Vf}(m, \mathit{Sign}(m, \mathit{sk}), \mathit{pk}) = 1] = 1$$

**Oracles: maximum attacker cabilities.** The *Signing Oracle* $\mathcal{SO}$ accepts input public key $\mathsf{pk}$ and a message $m$, outputs a valid signature.

**Security notions**: The existential unforgeability against adaptive-chosen-plaintext attackers is defined in terms of the following security game:

The **ACP-UF Game**
1. (*Setup Phase*) Simulator $\mathcal{S}$ sets up system parameters and public keys.
2. (*Probe Phase*) Attacker $\mathcal{A}$ queries the Signing Oracle $\mathcal{SO}$ in arbitrary interleaf.
3. (*End Game*) $\mathcal{A}$ delivers a valid message-signature pair $(m^*, \sigma^*)$ which is not an $\mathcal{SO}$ query output.

The Attacker $\mathcal{A}$ is said to $(q_S, T, \epsilon)$-*forge* if it makes $q_S$ queries to $\mathcal{SO}$, has running time $T$, and has success probability $\epsilon$ where the probability is taken over random choices of system parameters, public keys, and the random bits it consumes.

**Definition 2.** *A signature scheme is* $(q_S, T, \epsilon)$-ACP-UF (existentially-unforgeable against adaptive-chosen-plaintext attackers)*, if no algorithm $\mathcal{A}$ can $(q_S, T, \epsilon)$-forge.*

## 3 New short RO-free signatures from the $q$-SDH Assumption

We present the first of our three short signatures without random oracles. It is motivated by Boneh and Boyen [6]'s state-of-the-art short signature without random oracles. Below, we discuss intractability assumptions, then review [6]'s signature, before presenting our new signature.

### 3.1 Intractability assumptions

We present both existing and new intractability assumptions needed in this paper. There are two categories of intractability assumptions: those in the SDH (Strong Diffie-Hellman) family of assumptions, and those assumptions involving hash functions.

**3.1.1 SDH-family of intractability assumptions.** Let $\hat{\mathbf{e}} : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_3$ be a pairing, order$(\mathbb{G}_1) = q_1$ is a prime, $g_1$ (resp. $g_2$) be a generator of $\mathbb{G}_1$ (resp. $\mathbb{G}_2$). The original SDH (Strong Diffie-Hellman) Assumption [7] is as follows:

**Definition 3.** *The q-SDH (Strong Diffie-Hellman) Problem [7] is that, given* $g_1 \in \mathbb{G}_1$, $g_2^{x^i} \in \mathbb{G}_2$, $0 \leq i \leq q$, *output* $(c, g_1^{1/(x+c)})$. *An algorithm* $\mathcal{A}$ *is said to* $(T, \epsilon)$*-solves the q-SDH Problem if*

$$\Pr[\mathcal{A}(g_1, g_2, g_2^x, \cdots, g_2^{(x^q)}) = (c, g_1^{1/(x+c)})] \geq \epsilon \tag{1}$$

*with running time* $T$, *where the probability is over the random choice of* $x$ *and the random bits consumed by* $\mathcal{A}$. *The* $(q, T, \epsilon)$*-SDH Assumption is that no algorithm can* $(q, T, \epsilon)$*-solve the q-SDH Problem.*

Wei [33] presented the following variant, the SDH' Assumption, which better suits our purposes in this paper.

**Definition 4.** *The q-SDH' (Strong Diffie-Hellman') Problem [33] is that, given* $g_2$, $g_2^x \in \mathbb{G}_2$, $g_1^{x^i} \in \mathbb{G}_1$, $0 \leq i \leq q$, *output* $(c, g_1^{1/(x+c)})$. *An algorithm* $\mathcal{A}$ *is said to* $(T, \epsilon)$*-solves the q-SDH' Problem if*

$$\Pr[\mathcal{A}(g_2, g_2^x, g_1, g_1^x, \cdots, g_1^{(x^q)}) = (c, g_1^{1/(x+c)})] \geq \epsilon \tag{2}$$

*with running time* $T$, *where the probability is over the random choice of* $x$ *and the random bits consumed by* $\mathcal{A}$. *The* $(q, T, \epsilon)$*-SDH' Assumption is that no algorithm can* $(q, T, \epsilon)$*-solve the q-SDH' Problem.*

The following relationship between the $q$-SDH Assumption and the $q$-SDH' Assumption is straightforward, and its proof is omitted.

**Lemma 1** *Assume a homomorphic map* $\psi : \mathbb{G}_2 \rightarrow \mathbb{G}_1$ *with* $\psi(g_2) = g_1$ *is given. Then the* $(q, T, \epsilon)$-SDH Assumption *implies the* $(T, q, \epsilon)$-SDH' Assumption.

Zhang, et al. [34] defined the $q$-CAA (Coalition Attacker Algorithm) Assumption which is related to the SDH-family of assumptions specified above. Wei [33] presented a variant, named the $q$-CAA' Assumption in this paper, which better suits our purposes. Both CAA-family assumptions are specified below.

**Definition 5.** *The q-CAA' (Coalition Attacker Algorithm') Problem [33] is that, given* $g_2, g_2^x \in \mathbb{G}_2$, $g_1 \in \mathbb{G}_1$, *distinct nonzero* $\{a_1, \cdots, a_1\}$, $\{g_1^{1/(x+h_1)}, \cdots, g_1^{1/(x+h_q)}\}$, *output* $(h, g_1^{1/(x+h)})$, $h \notin \{h_1, \cdots, h_q\}$. *An algorithm* $\mathcal{A}$ *is said to* $(T, \epsilon)$*-solves the q-CAA' Problem if*

$$\Pr[\mathcal{A}(g_2, g_2^x, g_1, (h_1, g_1^{1/(x+h_1)}), \cdots, (h_q, g_1^{1/(x+h_q)})) = (h, g_1^{1/(x+h)}) \wedge \ h \notin \{h_1, \cdots, h_q\}] \geq \epsilon \tag{3}$$

*with running time $T$, where the probability is over the random choice of $x$, distinct nonzero $\{a_1, \cdots, a_q\}$, and the random bits consumed by $\mathcal{A}$. The $(q, T, \epsilon)$-CAA' Assumption is that no algorithm can $(T, \epsilon)$-solve the q-CAA' Problem.*

Let $\mathbb{G}_1 = \mathbb{G}_2$ in the pairing $\hat{\mathbf{e}}$, and let the generators $g_1 = g_2 = g$.

**Definition 6.** *The q-CAA (Coalition Attacker Algorithm) Problem [34] is that, given $g, g^x \in \mathbb{G}_1$, $(h_i, g^{1/(x+h_i)})$, $1 \leq i \leq q$, output $(h, g^{1/(x+h)})$, $h \notin \{h_1, \cdots, h_q\}$. An algorithm $\mathcal{A}$ is said to $(T, \epsilon)$-solves the q-CAA Problem if*

$$\Pr[\mathcal{A}(g, g^x, (h_1, g^{1/(x+h_1)}), \cdots, (h_q, g^{1/(x+h_q)})) = (h, g^{1/(x+h)}) \wedge\ h \notin \{h_1, \cdots, h_q\}] \geq \epsilon \quad (4)$$

*with running time $T$, where the probability is over the random choice of $x$, $h_1$, $\cdots$, $h_q$, and the random bits consumed by $\mathcal{A}$. The $(q, T, \epsilon)$-CAA Assumption is that no algorithm can $(q, T, \epsilon)$-solve the q-CAA Problem.*

Quoting Wei [33]'s Theorem 1, and easily detailing its exact security from the proof [33], we have the following equivalence:

**Theorem 2.** *The* $(q, T, \epsilon)$-SDH' Assumption *implies the* $(q, T - T_{mi}, \epsilon)$-CAA' Assumption*, where $T_{mi} \leq O(q^3)$ is the time to invert a non-singular $q \times q$ matrix. The* $(q, T', \epsilon')$-CAA' Assumption *implies the* $(q, T' - T_{pic}, \epsilon')$-SDH' Assumption*, where $T_{pic} \leq O(q^2)$ is the time to convert a q-SDH' Problem instance to a q-CAA' Problem instance.*

### 3.1.2 Intractability assumptions about hash functions.

**Definition 7.** *Let $\mathcal{H}$ be a mapping whose range is $\{0,1\}^\ell \setminus \{0^\ell\}$. The* $(\mathcal{H}, q, \ell)$-Know-Target-Sum Collision ($(\mathcal{H}, q, \ell)$-KTSC) Problem *is, given distinct nonzero $a_1, \cdots, a_q \in \{0,1\}^\ell$, output $b$ and $(i, j)$, $1 \leq i < j \leq q$, satisfying $\mathcal{H}(b) = a_i \oplus a_j$. An algorithm $\mathcal{A}$ is said to* $(T, \epsilon)$-solve *the $(\mathcal{H}, q)$-KTSC Problem if*

$$\Pr[\mathcal{A}(\mathcal{H}, q, a_1, \cdots, a_q) = (b, i, j) \wedge 1 \leq i < j \leq q \wedge \mathcal{H}(b) = a_i \oplus a_j] = \epsilon$$

*with running time $T$, and the probability is over random choices of distinct nonzero $a_1$, $\cdots$, $a_q$ and random bits $\mathcal{A}$ consumes. The* $(\mathcal{H}, q, T, \epsilon)$-KTSC Assumption *is that no algorithm can $(T, \epsilon)$-solve the $(\mathcal{H}, q)$-KTSC Problem. A mapping $\mathcal{H}$ is called a* $(q, T, \epsilon)$-KTSCR (Known-Target-Sum Collision-Resistant) hash function *if the $(\mathcal{H}, q, T, \epsilon)$-KTSC Assumption holds.*

## 3.2 Review: The SDH Signature from Boneh and Boyen [6]

We review Boneh and Boyen [6]'s short signature without random oracles. Let $\hat{\mathbf{e}} : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_3$ be a pairing, $\text{order}(\mathbb{G}_1) = \text{order}(\mathbb{G}_2) = q_1$, $g_2$ is a generator of $\mathbb{G}_2$. Let $\psi : \mathbb{G}_2 \to \mathbb{G}_1$ be a homomorphic mapping with $\psi(g_2) = g_1$.

**Signature** $\text{Sig}_{SDH}$ [6]:

1. $\mathsf{sk} = (x, y)$, $\mathsf{pk} = (g_1, g_2, g_2^x, g_2^y, \hat{\mathbf{e}})$.
2. **Signing Protocol** Given $\mathsf{sk}$, $\mathsf{pk}$, and message $m$, randomly generate $R \in Z_{q_1}^*$. Output the signature $(m, \sigma = g_1^{1/(x+m+Ry)})$.

3. **Verification Protocol** Upon receiving a signature $(R, \sigma)$ for message $m$, verify $\hat{\mathbf{e}}$ $(\sigma, g_2^{x+m+Ry}) = \hat{\mathbf{e}}$ $(g_1, g_2)$.

Boneh and Boyen [6] proved that the SDH Assumption implies the unforgeability of $\text{Sig}_{SDH}$:

**Theorem 3.** *[6] Assume a homomorphic map $\psi : \mathbb{G}_2 \to \mathbb{G}_1$ with $\psi(g_2) = g_1$ is known. Then signature scheme* $\text{Sig}_{SDH}$ *is* correct *and* $(q_S, T, \epsilon)$-ACP-UF *provided the* $(q_S, T+O(q_s^2), (\epsilon/2)-(q_S/q_1))$-SDH Assumption *holds.*

Note $O(q_S^2)$ is the time cost to convert an SDH Problem instance to the public parameters of the signature. Using a similar proof, we can also easily reduce the unforgeability of $\text{Sig}_{SDH}$ to the CAA' Assumption:

**Theorem 4.** *Assume a homomorphic map $\psi : \mathbb{G}_2 \to \mathbb{G}_1$ with $\psi(g_2) = g_1$ is known. The signature scheme* $\text{Sig}_{SDH}$ *is* correct *and* $(q_S, T, \epsilon)$-ACP-UF *provided the* $(q_S, T+O(q_S), (\epsilon/2)-(q_S/q_1))$-CAA' Assumption *holds.*

Note $O(q_S)$ is the time cost to service $q_S$ signing oracle queries. Combining Theorem 4 and Theorem 2, we also easily prove that the SDH' Assumption implies the unforgeability of $\text{Sig}_{SDH}$:

**Corollary 5** *Assume a homomorphic map $\psi : \mathbb{G}_2 \to \mathbb{G}_1$ with $\psi(g_2) = g_1$ is known. Signature* $\text{Sig}_{SDH}$ *is* correct *and* $(q_S, T, \epsilon)$-ACP-UF *provided the* $(q_S, T + O(q_S^3), (\epsilon/2) - (q_S/q_1))$-SDH' Assumption *holds.*

### 3.3 New short signature: the Product SDH Signature

We present the first of our three new short signatures without random oracles. It is motivated by Boneh and Boyen [6]'s state-of-the-art short signature. Let $\hat{\mathbf{e}} : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_3$ be a pairing, $\text{order}(\mathbb{G}_1) = q_1$ is a prime, $g$ is a generator of $\mathbb{G}_1$. Let $\mathcal{H}$ be a hashing function from the message space to the output range $\{0,1\}^{\lambda_s} \setminus \{0^{\lambda_s}\}$, where $\lambda_s$ is the security parameter.

**Signature** $\text{Sig}_{PSDH}$:

1. $\mathsf{sk} = (x, y)$, $\mathsf{pk} = (g, g^x, g^y, g^{xy}, \hat{\mathbf{e}}, \mathcal{H})$.
2. **Signing Protocol** Given $\mathsf{sk}$, $\mathsf{pk}$, and message $m$, randomly generate nonzero $m_1, m_2 \in \{0,1\}^{\lambda_s}$ with $m_1 \oplus m_2 = \mathcal{H}(m)$. Output the signature $(m_1, \sigma = g^{1/((x+m_1)(y+m_2))})$.
3. **Verification Protocol** Upon receiving a signature $(m_1, \sigma)$ for message $m$, compute $m_2 = \mathcal{H}(m) \oplus m_1$, verify $m_1 \neq 0$, $m_2 \neq 0$, and $\hat{\mathbf{e}}(\sigma, g^{(x+m_1)(y+m_2)}) = \hat{\mathbf{e}}(g, g)$.

The unforgeability of $\text{Sig}_{PSDH}$ is reducible to the CAA' Assumption.

**Theorem 6.** *The signature scheme* $\text{Sig}_{PSDH}$ *is* correct *and* $(q_S, T, \epsilon)$-ACP-UF *provided the* $(q_S, T + O(q_S^3), \epsilon/4 - q_S/q_1)$-CAA' Assumption *holds and $\mathcal{H}$ is a* $(q_S, T + O(q_S^3), \epsilon/4 - q_S/q_1)$-KTSCR (Known-Target-Sum Collision-Resistant) hash function.

Combining Theorems 2 and 6, we reduce the unforgeability of $\text{Sig}_{PSDH}$ to the SDH' Assumption:

**Corollary 7** *The signature scheme* $\text{Sig}_{PSDH}$ *is* correct *and* $(q_S, T, \epsilon)$-ACP-UF *provided the* $(q_S, T + O(q_S^3), \epsilon/4 - q_S/q_1)$-SDH' Assumption *holds and $\mathcal{H}$ is a* $(q_S, T + O(q_S^3), \epsilon/4 - q_S/q_1)$-KTSCR hash function.

*Proof of Theorem 6*: The correctness is trivial. Next we use an ACP-UF attacker to build a Simulator $\mathcal{S}$ to solve the intractability problems.

**Setup**: (*Transforming the problem instance*) Upon receiving a $q_S$-CAA' Problem instance including $\bar{g}_1$, $g_2$, $g_2^z$, $(\bar{m}_\tau, \bar{\sigma}_\tau)$, satisfying $\bar{\sigma}_\tau^{z+\bar{m}_\tau} = \bar{g}_1$, $1 \leq \tau \leq q_S$, Simulator $\mathcal{S}$ proceeds as follows:

Using the techniques contained in the proof of Theorem 1 of Wei [33], $\mathcal{S}$ computes $\tilde{g}^{z^i}$, $0 \leq i \leq q_S$, where $\tilde{g} = \bar{g}_1^{1/f(z)}$, $f(z) = \prod_{i=1}^{q_S}(z + \bar{m}_i)$. Note the complexity of the above transformation is $O(q_S^3)$ which is dominated by the inversion of a non-singular $q_S \times q_S$ matrix. There are further optimized matrix inversion techniques in the literature we elect not to pursue in favor of simplicity.

Then $\mathcal{S}$ randomly picks distinct nonzero $\hat{m}_1, \cdots, \hat{m}_{q_S} \in \{0,1\}^{\lambda_s}$, and computes $g_1 = \tilde{g}^{f_2(z)}$ where $f_2(z) = \prod_{i=1}^{q_S}(z + \hat{m}_i)$, and computes $\hat{\sigma}_i = g_1^{1/(z+\hat{m}_i)}$, $1 \leq i \leq q_S$. Note the complexity of the above transformation of the problem instance is $O(q_S^3)$ which is dominated

(*Setting up either of two modes*) At this point, $\mathcal{S}$ flips a fair coin $c_{mode}$ and sets up as follows:

1. If $c_{mode} = 1$, $\mathcal{S}$ randomly picks $y$, sets $g = g_2$, $\mathsf{pk} = (g^z, g^y, g^{zy})$.
2. If $c_{mode} = 2$, $\mathcal{S}$ picks $x$, sets $g = g_2$, $\mathsf{pk} = (g^x, g^z, g^{xz})$.

**Simulating** $\mathcal{SO}$: If $c_{mode} = 1$, do the following: Upon the $\tau$-th $\mathcal{SO}$ query input $m_\tau$, $1 \leq \tau \leq q_S$, abort if $\mathcal{H}(m_\tau) = \hat{m}_\tau$. Else set $m_{1,\tau} = \hat{m}_\tau$, $m_{2,\tau} = \mathcal{H}(m_\tau) \oplus m_{1,\tau}$ and output the signature $(m_{1,\tau}, \sigma_\tau = (\hat{\sigma}_\tau)^{1/(y+m_{2,\tau})})$.

If $c_{mode} = 2$, the $\mathcal{SO}$ is serviced similarly except with the roles of $g^x$ and $g^y$ swapped.

The **simulation deviation** [19]: It can be shown that any pairwise simulation deviation among (1) Real World, (2) Ideal World-1 where $c_{mode} = 1$, and (3) Ideal-World-2 where $c_{mode} = 2$, is negligible. The proof is tedious and mechanical. We omit it here.

The **extractions**: With probability $\epsilon$, Attacker $\mathcal{A}$ eventually delivers a valid message-signature pair ( $m^*$, $(m_1^*, \sigma^*)$ ), $m^* \neq m_\tau$, $\forall\tau$. There are these cases:

Case 1: $c_{mode} = 1$ and $x = z$. Let $\epsilon_{1,1}$ denote the conditional probability (conditioned on this Case) that $m_1^* \neq \hat{m}_\tau$ $\forall\tau$, which implies that $(m_1^*, (\sigma^*)^{y+m_2^*})$ solves the $q$-CAA' Problem.

Case 2: $c_{mode} = 2$ and $y = z$. Let $\epsilon_{2,2}$ denote the conditional probability (conditioned on this Case) that $m_2^* \neq m_{2,\tau} = \hat{m}_\tau$ for any $\tau$. Then $(m_2^*, (\sigma^*)^{x+m_1^*})$ solves the $q$-CAA' Problem.

Case 3: $c_{mode} = 1$ or 2, and $m_1^*, m_2^*$ are both in $\{\hat{m}_1, \cdots, \hat{m}_{q_S}\}$. Then $(m^*, m_1^*, m_2^*)$ solves the $(\mathcal{H}, q)$-KTSC Problem.

(*Transforming back the solution to the problem instance*) One final detail. When $\mathcal{S}$ obtains a solution consisting of message signature $(m_1^*, \tilde{\sigma})$ with $\tilde{\sigma}^{z+m_1^*} = g_1 = \tilde{g}^{f_2(z)}$, the solution is transformed back to a solution to the CAA' Problem instance as follows: Compute $\tilde{g}^{1/(z+m_1^*)}$ from $\tilde{g}^{z^i}$, $0 \leq i \leq q_S$, and $g_1$. Then compute $\bar{g}_1^{1/(z+m_1^*)} = \tilde{g}^{f(z)/(z+m_1^*)}$ which solves the CAA' Problem instance.

The *Exact Security*: The negligible statistical distance between the real world and the two ideal worlds imply $\epsilon_{2,2} = \epsilon/2 - \epsilon_{1,1}$. Combining, $\mathcal{S}$ has a probability at least $\epsilon/4 - (q_S^2 + q_S)/q_1$ of solving the $q$-CAA' Problem, or at least the same probability of solving the $(\mathcal{H}, q)$-KTSC Problem. Note the total probability of aborting during $\mathcal{SO}$ simulation with $\mathcal{H}(m_\tau) = \hat{m}_\tau$ is bounded by $q_S/q_1$. □

**Efficiency discussions** Using high-security pairings suggested by Koblitz and Menezes [23], with security level $\lambda_s = 128, 192, 256$ bits, we have $q_1$ is $2\lambda_s$ bits and a $\mathbb{G}_1$ element is represented by $2.5\lambda_s$ bits. The BB04 [6] signature $(R, \sigma)$ is $||q_1|| + ||g_1|| = 4.5\lambda_s$ bits long. Our

signature $\text{Sig}_{PSDH}$ is $||g_1|| + \lambda_s = 3.5\lambda_s$ bits long. Our signature $\text{Sig}_{PSDH}$, is (2/9)≈22% shorter than the state-of-the-art short signature of Boneh and Boyen [6]. We use hashing outputs of $\lambda_s$ bits which is typically considered sufficiently secure.

The complexity of verifying $\text{Sig}_{PSDH}$ consists of one multi-base exponentiation in $\mathbb{G}_1$ and one pairing. Essentially the same as that of $\text{Sig}_{SDH}$ [6]. The exact security of $\text{Sig}_{PSDH}$ in Theorem 6 (resp. Corollary 7) is essentially the same as that of $\text{Sig}_{SDH}$ [6] in Theorem 4 (resp. Corollary 5).

## 4 Another short RO-free signature: the Product Square Root (PSR) signature

We present the second of our three new short signatures without random oracles. It is motivated by Zhang, et al. [35]'s state-of-the-art short signature from the $q$ SR (Square Root) Assumption. Below, we discuss intractability assumptions both new and old, then review Zhang, et a;. [35]'s signature, before presenting our new signature.

Let $\hat{\mathbf{e}} : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_3$ be a pairing, order$(\mathbb{G}_1) = q_1$ is a prime, $g$ is a generator of $\mathbb{G}_1$.

### 4.1 Intractability Assumptions

We present several needed assumptions. The $q$-SR Assumption is from [35]. The other assumptions are new.

**Definition 8.** *The* $q$-Square Root ($q$-SR) Problem *is, given random* $g$, $g^x$, $Z_\tau$, $a_\tau$ *satisfying* $\hat{\mathbf{e}}(Z_\tau, Z_\tau) = \hat{\mathbf{e}}(g^{x+a_\tau}, g)$, $1 \leq \tau \leq q$, *output* $(a, Z)$, *satisfying* $\hat{\mathbf{e}}(Z, Z) = \hat{\mathbf{e}}(g^{x+a}, g)$, $a \notin \{a_1, \cdots, a_q\}$. *An algorithm* $\mathcal{A}$ *is said to* $(T, \epsilon)$*-solve the q-SR Problem if*

$$\Pr[\mathcal{A}(g, g^x, g^{(x+a_1)^{1/2}}, \cdots, g^{(x+a_q)^{1/2}}) = (a, g^{(x+a)^{1/2}}) \ \wedge \ a \text{ is distinct from all } a_i\text{'s}] \geq \epsilon$$

*with running time* $T$, *where the probability is taken over qualified random choices of* $x$, $a_1$, $\cdots$, $a_q$, *and random bits consumed by* $\mathcal{A}$. *The* $(q, T, \epsilon)$*-SR Assumption is that no algorithm can solve the* $(T, \epsilon)$*-solve the q-SR Problem.*

**Definition 9.** *The* $(q, \ell)$-Bounded-Length Square Root Problem*, or simply the* $(q, \ell)$-Square Root ($(q, \ell)$-SR) Problem *is, given random* $g$, $g^x$, $Z_\tau$, $a_\tau \in \{0,1\}^\ell$ *satisfying* $\hat{\mathbf{e}}(Z_\tau, Z_\tau) = \hat{\mathbf{e}}(g^{x+a_\tau}, g)$, $1 \leq \tau \leq q$, *output* $(a, Z)$, *satisfying* $\hat{\mathbf{e}}(Z, Z) = \hat{\mathbf{e}}(g^{x+a}, g)$, $a \in \{0,1\}^\ell \setminus \{a_1, \cdots, a_q\}$. *An algorithm* $\mathcal{A}$ *is said to* $(T, \epsilon)$*-solve the* $(q, \ell)$*-SR Problem if*

$$\Pr[\mathcal{A}(g, g^x, g^{(x+a_1)^{1/2}}, \cdots, g^{(x+a_q)^{1/2}}) = (a, g^{(x+a)^{1/2}}) \ \wedge \ a \in \{0,1\}^\ell \setminus \{a_1, \cdots, a_q\}] \geq \epsilon$$

*with running time* $T$, *where the probability is taken over qualified random choices of* $x$, $\{a_1$, $\cdots$, $a_q\} \subset \{0,1\}^\ell$, *and random bits consumed by* $\mathcal{A}$. *The* $(q, \ell, T, \epsilon)$*-SR Assumption is that no algorithm can solve the* $(T, \epsilon)$*-solve the* $(q, \ell)$*-SR Problem.*

**Definition 10.** *The* $(q, \ell)$-Square Root Quadratic Non-Residue ($(q, \ell)$-SRQNR) Problem *is, given random* $g$, $g^x$, $Z_\tau$, $a_\tau \in \{0,1\}^\ell$ *satisfying* $\hat{\mathbf{e}}(Z_\tau, Z_\tau) = \hat{\mathbf{e}}(g^{x+a_\tau}, g)$, $1 \leq \tau \leq q$, *output*

$(a, Z, \gamma)$, *satisfying* $\gamma \in QNR(q_1)$, $\hat{\mathbf{e}}(Z, Z) = \hat{\mathbf{e}}(g^{x+a}, g^{\gamma})$, $a \in \{0,1\}^{\ell} \setminus \{a_1, \cdots, a_{q_S}\}$. *An algorithm* $\mathcal{A}$ *is said to* $(T, \epsilon)$*-solve the* $(q, \ell)$*-SRQNR Problem if*

$$\Pr[\mathcal{A}(g, g^x, g^{(x+a_1)^{1/2}}, \cdots, g^{(x+a_q)^{1/2}}) = (a, Z, \gamma) \ \wedge \ \gamma \in QNR(q_1) \\ \wedge \ \hat{\mathbf{e}}(Z, Z) = \hat{\mathbf{e}}(g^{x+a}, g^{\gamma}) \wedge a \in \{0,1\}^{\ell} \setminus \{a_1, \cdots, a_{q_S}\}] \geq \epsilon$$

*with running time* $T$*, where the probability is taken over qualified random choices of* $x$*,* $\{a_1, \cdots, a_q\} \subset \{0,1\}^{\ell}$*, and random bits consumed by* $\mathcal{A}$*. The* $(q, \ell, T, \epsilon)$*-SRQNR Assumption is that no algorithm can solve the* $(T, \epsilon)$*-solve the* $(q, \ell)$*-SRQNR Problem.*

**Intractabillity assumptions about hash functions.**

**Definition 11.** *Let* $\mathcal{H}_1 : \{0,1\}^{\ell_m} \to \{0,1\}^{\ell}$ *and* $\mathcal{H}_2 : \{0,1\}^{\ell} \to \{0,1\}^{\ell}$ *be two mappings. The* $(\mathcal{H}_1, \mathcal{H}_2, q)$-Know-Target-Sum Tandem-Collision ($(\mathcal{H}, q, \ell)$-KTSTC) Problem *is, given random* $a_1, \cdots, a_q \in \{0,1\}^{\ell}$*, output* $(b, i, j, k)$*,* $1 \leq i, j \leq q$*,* $k$ *is a non-negative integer, satisfying* $\mathcal{H}_2^k(\mathcal{H}_1(b)) = a_i \oplus a_j$*. An algorithm* $\mathcal{A}$ *is said to* $(T, \epsilon)$-solve *the* $(\mathcal{H}_1, \mathcal{H}_2, q)$*-KTSTC Problem if*

$$\Pr[\mathcal{A}(\mathcal{H}_1, \mathcal{H}_2, q, a_1, \cdots, a_q) \\ = (b, i, j, k) \wedge 1 \leq i \leq j \leq q \wedge \mathcal{H}_2^k(\mathcal{H}_1(b)) = a_i \oplus a_j] = \epsilon$$

*with running time* $T$*, and the probability is over random choices of* $a_1, \cdots, a_q$ *and random bits* $\mathcal{A}$ *consumes. The* $(\mathcal{H}_1, \mathcal{H}_2, q, T, \epsilon)$-KTSTC Assumption *is that no algorithm can* $(T, \epsilon)$*-solve the* $(\mathcal{H}_1, \mathcal{H}_2, q)$*-KTSTC Problem. A pair of mappings* $(\mathcal{H}_1, \mathcal{H}_2)$ *is called a* $(q, T, \epsilon)$-KTSTCR (Known-Target-Sum Tandem-Collision-Resistant) hashing tandem *if the* $(\mathcal{H}, q, T, \epsilon)$*-KTSTC Assumption holds.*

### 4.2 Review: Zhang, et al. [35]'s *q*-Square Root signature

. We review Zhang, et al. [35]'s short signature withour random oracles from the $q$-SR Assumption.

**Signature** $\text{Sig}_{SR}$:

1. $\mathsf{sk} = x$, $\mathsf{pk} = (g, g^x, \hat{\mathbf{e}})$.
2. *Signing Protocol* Given $\mathsf{sk}$, $\mathsf{pk}$, and message $m$, randomly generate $R$ satisfying $x+my+R \in QR(q_1)$. Output the signature $(R, \sigma = g^{(x+my+R)^{1/2}})$. Randomly choose either square root of $x + my + R$.
3. *Verification Protocol* Upon receiving a signature $(R, \sigma)$ for message $m$, verify $\hat{\mathbf{e}}(\sigma, \sigma) = \hat{\mathbf{e}}(g^{x+my+R}, g)$.

Zhang, et al. [35] reduced the unforgeability of $\text{Sig}_{SR}$ to the $q$-SR Assumption:

**Theorem 8.** [35] *The signature scheme* $Sig_{SR}$ *is* correct *and* $(q_S, T, \epsilon)$-ACP-UF *provided the* $(q, T + O(q_S), (\epsilon/2) - (q_S/q_1))$-SR Assumption *holds.*

### 4.3 New signature from the Product Square-Root (PSR) Assumption: $\mathbf{Sig}_{PSR}$

We present the second of our three new short signatures without random oracles. This signature is modified from Zhang, et al. [35]'s short signature without random oracles, but improves upon it.

**Signature** $\mathrm{Sig}_{PSR}$:

1. $\mathsf{sk} = (x, y)$, $\mathsf{pk} = (g, g^x, g^y, \hat{\mathbf{e}}, \mathcal{H}_1, \mathcal{H}_2)$, where $\mathcal{H}_1 : \{0,1\}^{ell_m} \rightarrow \{0,1\}^{\lambda_s}$, $\mathcal{H}_2 : \{0,1\}^{\lambda_s} \rightarrow \{0,1\}^{\lambda_s}$.
2. *Signing Protocol*: Given $\mathsf{sk}$, $\mathsf{pk}$, and message $m$, do the following:
   (a) Initialize $k = 0$.
   (b) Randomly pick nonzero $m_1$, $m_2$ from $\{0,1\}^{\lambda_s}$.
   (c) If $x+m_1, y+m_2 \in QR(q_1)$, then output the signature $(m_1, {}'0^k1', \sigma = g^{(x+m_1)^{1/2}(y+m_2)^{1/2}})$ and terminate. Randomly choose either square root. Else increment $k$ by one and go back to the previous step.

   Note $'0^k1'$ is the string with $k$ zeros followed by a $'1'$, and it is the Null string when $k = 0$.
3. *Verification Protocol*: Upon receiving a signature $(m_1, {}'0^k1', \sigma)$ for message $m$, parse the signature, recover $k$ from the second entry, compute $m_2 = \mathcal{H}_2^k(\mathcal{H}_1(m)) \oplus m_1$, verify $m_1, m_2 \in \{0,1\}^{\lambda_s} \setminus \{0\}$ and $\hat{\mathbf{e}}(\sigma, \sigma) = \hat{\mathbf{e}}(g^{x+m_1}, g^{y+m_2})$.

**Theorem 9.** *The signature scheme* $Sig_{PSR}$ *is* correct *and* $(q_S, T, \epsilon)$-ACP-UF *provided the* $(q_S, \lambda_s, T + O(q_S), \epsilon/6 - (q_S^2 + 13q_S)/q_1)$-SR Assumption*, the* $(q_S, \lambda_s, T + O(q_S), \epsilon/6 - (q_S^2 + 13q_S)/q_1)$-SRQNR Assumption *all hold, and* $(\mathcal{H}_1, \mathcal{H}_2)$ *is a* $(q_S, \lambda_s, T + O(q_S), \epsilon/6 - (q_S^2 + 13q_S)/q_1)$-KTSTCR hashing tandem.

*Proof Sketch*: The correctness is trivial. Next we use a successful ACP-UF attacker to build a solver of the intractability problem.

**Setup**: Denote $\ell = \lambda_s$. Simulator $\mathcal{S}$ received a $(q, \ell)$-SR Problem instance including $\hat{m}_\tau \in \{0,1\}^\ell, \hat{\sigma}_\tau = g^{(x+\hat{m})^{1/2}}$, $1 \le \tau \le q_S$. $\mathcal{S}$ aborts if $\{\hat{m}_\tau : 1 \le \tau \le q_S\}$ contains zeros or duplicates. The probability of this abort is $(q_S^2 + q_S)/q_1$. If not abort, $\mathcal{S}$ flips a fair coin $c_{mode}$ and sets up as follows:

1. If $c_{mode} = 1$, $\mathcal{S}$ randomly picks $y$, sets $\mathsf{pk} = (g^z, g^y)$.
2. If $c_{mode} = 2$, $\mathcal{S}$ randomly picks $x$, sets $\mathsf{pk} = (g^x, g^z)$.

**Simulating** $\mathcal{SO}$: If $c_{mode} = 1$, do the following: Upon the $\tau$-th $\mathcal{SO}$ query input $m_\tau$, $1 \le \tau \le q_S$, do the following:

1. Initialize $k_\tau = 0$.
2. Compute $m_{1,\tau} = \hat{m}_\tau \neq 0$, $m_{2,\tau} = \mathcal{H}_2^k(\mathcal{H}_1(m_\tau)) \oplus m_{1,\tau}$. Abort if $m_{2,\tau} = 0$. Else flip a fair coin $coin_\tau$.
3. If $coin_\tau = 1$ and $y + m_{2,\tau} \in QR(q_1)$, then output the signature $(m_{1,\tau}, {}'0^{k_\tau}1', \sigma = g^{(x+m_{1,\tau})^{1/2}(y+m_{2,\tau})^{1/2}})$ and terminate this $\mathcal{SO}$ query. Randomly choose either square root. Else increment $k_\tau$ by one and return to Step 2 above.

If $c_{mode} = 2$, do the following: Simulate $\mathcal{SO}$ similarly to the case $c_{mode} = 1$, except with the roles of $x$ and $y$ swapped.

**Simulation deviation**: There are three *worlds* to consider: (1) Real World, (2) Ideal World-1 where $c_{mode} = 1$, and (3) Ideal-World-2 where $c_{mode} = 2$. The simulation deviation

between the two Ideal Worlds is negligible due to symmetry. That the simulation deviation between the Real World and either Ideal World is negligible is proved below.

Without loss of generality, let $c_{mode} = 1$. Given any $\mathcal{SO}$ output for query $m_\tau$, denoted $(m_{1,\tau}, '0^{k_\tau}1', \sigma = g^{(x+m_{1,\tau})^{1/2}(y+m_{2,\tau})^{1/2}})$, there exists a sequence of random bits consumed by Signer in the Real World that produces the same output with the same probability, as follows: Real World Signer, for each $k$, $0 \le k < k_\tau$, randomly generates nonzero $\tilde{m}_{1,k}$, $\tilde{m}_{2,k} \in \{0,1\}^\ell$ satisfying $\mathcal{H}_2^k(\mathcal{H}_1(m_\tau)) = \tilde{m}_{1,k} \oplus \tilde{m}_{2,k}$. But it occurs that $(x+\tilde{m}_{1,k}, y+\tilde{m}_{2,k}) \notin QR(q_1)^2$ for each $k < k_\tau$. Then Real World Signer generates, in the $k_\tau$-th try, $(\tilde{m}_{i,k_\tau}, \tilde{m}_{2,k_\tau}) = (\hat{m}_\tau, m_{2,\tau}) \in QR(q_1)^2$. The probability of the above event equals the probability of $\mathcal{SO}$ outputting the same signature. Therefore the simulation deviation between Real World and Ideal World-1 is negligible.

**Extractions**: With probability $\epsilon$, Attacker $\mathcal{A}$ eventually delivers a valid message-signature pair ( $m^*$, $(m_1^*, 0^k1', \sigma^*)$ ), $m^* \neq m_\tau$, $\forall\tau$. There are these events:

- Event A: We have $m_1^* \neq \hat{m}_\tau$ for any $\tau$, or $m_2^* \neq \hat{m}_\tau$ for any $\tau$. We also have $x+m_1^*, y+m_2^* \in QR(q_1)$. In this case, we have solved the $(q,\ell)$-SR Problem.
- Event B: We have $m_1^* \neq \hat{m}_\tau$ for any $\tau$, or $m_2^* \neq \hat{m}_\tau$ for any $\tau$. We also have $x+m_1^*, y+m_2^* \in QNR(q_1)$. In this case, we have solved the $(q,\ell)$-SRQNR Problem.
- Event C: We have $m_1^* = \hat{m}_\tau$ for some $\tau$, and $m_2^* = \hat{m}_{\tau'}$ for some $\tau'$. In this case, we have solved the $(\mathcal{H}_1, \mathcal{H}_2, q)$-KTSTC Problem.

Let $\epsilon_{c_{mode},A}$ (resp. $\epsilon_{c_{mode},B}$, $\epsilon_{c_{mode},C}$) denote the probability that $c_{mode}$ is as denoted and Event A (resp. Event B, Event C) occurs. The negligible statistical distance between any worlds implies that $\epsilon_{1,A} = \epsilon_{2,A}$, $\epsilon_{1,B} = \epsilon_{2,B}$, and $\epsilon_{1,C} = \epsilon_{2,C}$.

*Exact Security*: The expected value of $k$ equals $\sum_{k=0}^{\infty} k(3/4)^k = 12$. The probability of $\mathcal{SO}$ aborting due to $\mathcal{H}_2^k(\mathcal{H}_1(m_\tau)) = \hat{m}_\tau$ is upper bounded by $12q_s/q_1$. Noting $\epsilon/2 = \epsilon_{1,A} + \epsilon_{1,B} + \epsilon_{1,C}$, we have the Theorem. □

**Efficiency discussions** *Bandwidth*: Our signature $\mathrm{Sig}_{PSR}$ consists of one $\mathbb{G}_1$ element and $m_1$ whose length equals that of the hash output and a string $'0^k1'$. Using high-security pairings parameters from [23], the $\mathbb{G}_1$ element is $(5/2)\lambda_s$ bits long. The hash output $m_1$ is chosen to be $\lambda_s$ bits long, which is typically sufficiently secure for cryptographically secure collision-resistant hashing fuctions. The expected length of the string $'0^k1'$ is $1 + \sum_{k=0}^{\infty} k(3/4)^{-k} = 1 + 12 = 13$ bits. The total signature length is $(7/2)\lambda_s + 13$ bits. It is 909 bits for $\lambda_s = 256$. In comparison, either Boneh, et al. [6]'s or Zhang, et al. [35]'s signature is $(9/2)\lambda_s = 1152$ bits. Our $\mathrm{Sig}_{PSR}$ is 21% shorter.

*Verifiction's online complexity*: Verifying $\mathrm{Sig}_{PSR}$ costs one pairing and one multi-base exponentiations in $\mathbb{G}_3$. The latter is

$$\hat{\mathbf{e}}(g^{x+m_1}, g^{y+m_2}) = \hat{\mathbf{e}}(g^x, g^y)\hat{\mathbf{e}}(g^x, g)^{m_2}\hat{\mathbf{e}}(g, g^y)^{m_1}\hat{\mathbf{e}}(g, g)^{m_1 m_2}$$

This cost is essentially the same as verifying $\mathrm{Sig}_{SR}$. In comparison, verifying $\mathrm{Sig}_{SDH}$ (resp. $\mathrm{Sig}_{PSDH}$) costs one pairing and one multi-base exponentiation in $\mathbb{G}_1$. For $\lambda_s = 128$ (resp. 256) bits, the number of bits of $q_1$ is $2\lambda_s = 256$ (resp. 512) bits, and the number of bits of elements of $\mathbb{G}_3$ is 3072 (resp. 15360) bits [23]. Verifying $\mathrm{Sig}_{SDH}$ or $\mathrm{Sig}_{PSDH}$ is that much more efficient than verifying $\mathrm{Sig}_{SR}$ or $\mathrm{Sig}_{PSR}$.

*Exact Security*: The exact security of $\mathrm{Sig}_{PSR}$ in Theorem 9 is similar to that of $\mathrm{Sig}_{SR}$ in Theorem 8.

## 5 Yet another RO-free signature: the CL04B' Signature

Camenisch and Lysyanskaya [10] presented three signatures without random oracles, Schemes A, B, and C. We modify their Scheme B, hereby named $\text{Sig}_{CL04B}$, into a variant we name $\text{Sig}_{CL04B'}$. We prove the security of $\text{Sig}_{CL04B'}$ without random oracles and without the sequential one-more external signing oracle $O_{X,Y}(\cdot)$ used in all previous results containing the LRSW Assumption.

Let $\hat{\mathbf{e}} : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_3$ be a pairing, $\text{order}(\mathbb{G}_1) = q_1$ is a prime, $g$ is a generator of $\mathbb{G}_1$.

### 5.1 Intractability assumptions

First we review some existing, and define some new, intractability assumptions:

**Definition 12.** *[25] The* LRSW Problem *is: Given random $g, X = g^x, Y = g^y$, and an oracle $O_{X,Y}(\cdot)$ which, upon input $m$, returns a random tuple $(m, a, b, c)$ satisfying $\hat{\mathbf{e}}(b, g) = \hat{\mathbf{e}}(a, Y)$, $\hat{\mathbf{e}}(c, g) = \hat{\mathbf{e}}(ab^m, X)$; output $(m^*, a^*, b^*, c^*)$ satisfying $\hat{\mathbf{e}}(b^*, g) = \hat{\mathbf{e}}(a^*, g^y)$, $\hat{\mathbf{e}}(c^*, g) = \hat{\mathbf{e}}(a^*(b^*)^{m^*}, g^x)$, $m^*$ has never been queried to $O_{X,Y}$. The* LRSW Assumption *is that no PPT algorithm can solve the LRSW Problem with non-negligible probability.*

The following new variant of the LRSW Assumption will be useful. Note its formulation is without the external sequential one-more signing oracle $O_{X,Y}(\cdot)$.

**Definition 13.** *The* $q$-wholesale LRSW ($q$-whLRSW) Problem *is: Given random $g, g^x, g^y$, random $(m_\tau, a_\tau, b_\tau, c_\tau)$ satisfying $\hat{\mathbf{e}}(b_\tau, g) = \hat{\mathbf{e}}(a_\tau, g^y)$, $\hat{\mathbf{e}}(c_\tau, g) = \hat{\mathbf{e}}(a_\tau b_\tau^{m_\tau}, g^x)$, $1 \le \tau \le q$; output $(m^*, a^*, b^*, c^*)$ satisfying $\hat{\mathbf{e}}(b^*, g) = \hat{\mathbf{e}}(a^*, g^y)$, $\hat{\mathbf{e}}(c^*, g) = \hat{\mathbf{e}}(a^*(b^*)^{m^*}, g^x)$, $m^* \neq m_\tau \ \forall \tau$. An algorithm $\mathcal{A}$ is said to $(T, \epsilon)$-solve the q-whLRSW Problem if*

$$\begin{aligned}\Pr[\mathcal{A}(g, g^x, g^y, (m_1, a_1, b_1, c_1), \cdots, (m_q, a_q, b_q, c_q)) = (m, a, b, c) \\ \wedge \ \hat{\mathbf{e}}(b, g) = \hat{\mathbf{e}}(a, g^y) \wedge \hat{\mathbf{e}}(c, g) = \hat{\mathbf{e}}(ab^m, g^x) \wedge m \neq m_\tau \forall \tau] = \epsilon\end{aligned}$$

*with running time $T$, where the probability is taken over qualified random choices of $x$, $y$, $(m_1, a_1, b_1, c_1)$, $\cdots$, $(m_q, a_q, b_q, c_q)$ and random bits consumed by $\mathcal{A}$. The $(q, T, \epsilon)$-whLRSW Assumption is that no algorithm can $(T, \epsilon)$-solve the q-whLRSW Problem.*

### 5.2 Review: The CL04B signature [10]

**Signature** $\text{Sig}_{CL04B}$:

1. $\mathsf{sk} = (x, y, z)$, $\mathsf{pk} = (g, g^x, g^y, g^z, \hat{\mathbf{e}})$.
2. **Signing Protocol** Given $\mathsf{sk}$, $\mathsf{pk}$, and message $m = (m_1, m_2)$, randomly generate $a$, compute $A = a^z$, $b = a^y$, $B = a^{yz}$, $c = a^{x+m_1xy+m_2xyz}$. Output the signature $(a, A, b, B, c)$.
3. **Verification Protocol** Upon receiving a signature $(a, A, b, B, c)$ for message $m$, verify

$$\begin{aligned}\hat{\mathbf{e}}(A, g) = \hat{\mathbf{e}}(a, g^z), \quad \hat{\mathbf{e}}(b, g) = \hat{\mathbf{e}}(a, g^y), \quad \hat{\mathbf{e}}(B, g) = \hat{\mathbf{e}}(A, g^y), \\ \hat{\mathbf{e}}(B, g) = \hat{\mathbf{e}}(b, g^z), \quad \hat{\mathbf{e}}(c, g) = \hat{\mathbf{e}}(ab^{m_1}B^{m_2}, g^x)\end{aligned} \tag{5}$$

Camenisch and Lysyanskaya [10] supplied the following security result:

**Theorem 10.** *[10] Signature $\text{Sig}_{CL04B}$ is* correct *and* ACP-UF *provided the* LRSW Assumption *holds.*

### 5.3 New RO-free signature: The CL04B' Signature

Camenisch and Lysyanskaya [10]'s second signature, $\text{Sig}_{CL04B}$, is provable in the plain model provided the LRSW Assumption holds. But the LRSW Assumption is formulated with an (external) oracle $O_{X,Y}(\cdot)$. Below, we prove a slightly modified version of $\text{Sig}_{CL04B}$, which we name $\text{Sig}_{CL04B'}$, to be secure in the plain model provided the $q$-whLRSW Assumption holds. Note the $q$-whLRSW Assumption is specified without any oracle similar to $O_{X,Y}(\cdot)$.

**Signature** $\text{Sig}_{CL04B'}$:

1. $\mathsf{sk} = (x, y, z)$, $\mathsf{pk} = (g, g^x, g^y, g^z, \hat{\mathbf{e}})$.
2. **Signing Protocol** Given $\mathsf{sk}$, $\mathsf{pk}$, and message $m$, randomly generate $a$, compute $A = a^z$, $b = a^y$, $B = a^{yz}$, $c = a^{x+(m+zR)xy}$. Output the signature $(R, a, A, b, B, c)$.
3. **Verification Protocol** Upon receiving a signature $(R, a, A, b, B, c)$ for message $m$, verify

$$\begin{gathered}\hat{\mathbf{e}}(A, g) = \hat{\mathbf{e}}(a, g^z), \quad \hat{\mathbf{e}}(b, g) = \hat{\mathbf{e}}(a, g^y), \quad \hat{\mathbf{e}}(B, g) = \hat{\mathbf{e}}(A, g^y), \\ \hat{\mathbf{e}}(B, g) = \hat{\mathbf{e}}(b, g^z), \quad \hat{\mathbf{e}}(c, g) = \hat{\mathbf{e}}(ab^m B^R, g^x)\end{gathered} \tag{6}$$

**Theorem 11.** *The signature scheme* $\text{Sig}_{CL04B'}$ *is* correct *and* $(q_S, T, \epsilon)$-ACP-UF *provided the* $(q_S, 2T + O(q_S), (2\epsilon^2/9) - (q_S^2 + q_S)/q_1)$-whLRSW Assumption *holds.*

*Proof*: The correctness is trivial. Next we use a successful ACP-UF attacker to build a solver of the intractability problem. In a nutshell. assume a PPT attacker $\mathcal{A}$ who can win the ACP-UF Game in average time $T$ and probability $\epsilon$. We use $\mathcal{A}$ to build a Simulator $\mathcal{S}$ who can solve the $q_S$-whLRSW Problem.

**Setup**: $\mathcal{S}$ received a $q_S$-whLRSW Problem instance: $g, g^u, g^v$, $(\hat{m}_\tau, \hat{a}_\tau, \hat{b}_\tau, \hat{c}_\tau)$, $1 \le \tau \le q_S$. $\mathcal{S}$ aborts if there are duplicates among $\hat{m}_\tau$'s. Note the probability of this abort is $q_s^2/q_1$. If it does not abort, $\mathcal{S}$ flips a three-way fair coin $c_{mode}$ and sets up as follows:

1. If $c_{mode} = 1$, $\mathcal{S}$ randomly picks $z$, sets $\mathsf{pk} = (g^u, g^v, g^z)$.
2. If $c_{mode} = 2$, $\mathcal{S}$ picks $x, y$, sets $\mathsf{pk} = (g^x, g^y, g^u)$.
3. If $c_{mode} = 3$, $\mathcal{S}$ picks $x, y$, sets $\mathsf{pk} = (g^x, g^y, g^v)$.

**Simulating** $\mathcal{SO}$: If $c_{mode} = 1$, do the following: Upon the $\tau$-th $\mathcal{SO}$ query input $m_\tau$, $1 \le \tau \le q_S$, solve for $R_\tau$ in $\hat{m}_\tau = m_\tau + R_\tau z$. Output the signature $(R_\tau, a_\tau = \hat{a}_\tau, b_\tau = \hat{b}_\tau, c_\tau = \hat{c}_\tau)$.

If $c_{mode} = 2$ or 3, do the following: Upon the $\tau$-th $\mathcal{SO}$ query input $m_\tau$, $1 \le \tau \le q_S$, randomly pick $\alpha_\tau$, $R_\tau$. Output the signature $(R_\tau, a_\tau = g^{\alpha_\tau}, b_\tau = g^{\alpha_\tau y}, c_\tau = g^{\alpha_\tau(1+(m_\tau+R_\tau z)xy)} = (g^z)^{R_\tau xy} g^{\alpha_\tau(1+m_\tau xy)})$.

The **simulation deviation**: It can be shown that the pairwise simulation deviation between any two of the following *worlds* are negligible: (1) Real World, (2) Ideal World-1 where $c_{mode} = 1$, (3) Ideal-World-2 where $c_{mode} = 2$, and (4) Ideal-World-3 where $c_{mode} = 3$. The proof is tedious but mechanical. We omit it.

**Extraction**: With probability $\epsilon$, Attacker $\mathcal{A}$ eventually delivers a valid message-signature pair ( $m^*$, $(a^*, A^*, b^*, B^*, c^*)$ ), $m^* \neq m_\tau$, $\forall\tau$. There are two events:

- Event A: $m^* + R^* z \neq \hat{m}_\tau$, $\forall\tau$.
- Event B: $m^* + R^* z = \hat{m}_\tau$, form some $\tau$.

For $i = 1, 2, 3$, let $\epsilon_{i,A}$ (resp. $\epsilon_{c_{mode},B}$) denote the probability that $c_{mode} = i$ and Event A (resp. Event B). The negligibility of simulation deviations implies that $\epsilon_{1,A} = \epsilon_{2,A} = \epsilon_{3,A} = \epsilon_A$ and $\epsilon_{1,B} = \epsilon_{2,B} = \epsilon_{3,B} = \epsilon_B$. Note $\epsilon = 3\epsilon_A + 3\epsilon_B$.

In Event A, the tuple $(m^* + R^*z, a^*, b^*, c^*)$ solves the $q_S$-whLRSW Problem instance at hand. In Event B, we have $m^* + R^*z = \hat{m}_\tau = m_\tau + R_\tau z$, $m^* \neq m_\tau$, and the discrete logarithm $z = -(R^* - R_\tau)^{-1}(m^* - m_\tau)$ is solved where $z = u$ if $c_{mode} = 2$, and $z = v$ if $c_{mode} = 3$.

Finally, we rewind $\mathcal{A}$ to the beginning and resimulate it with a new randomness tape but with the same inputs of system parameters and $q_S$-whLRSW Problem instance, and flipping a new three-way fair coin $c'_{mode}$. Combining the result of both simulation *forks*, we obtain

1. The probability of Event A and $c_{mode}$=1 in the first fork or the second fork is $1 - (1 - \epsilon_A/3)^2 = (2/3)\epsilon_A - (1/9)\epsilon_A^2$. With this probability, we solve the $q_S$-whLRSW Problem instance at hand.
2. The probability of Event B in the first fork and the second fork, and $(c_{mode}, c'_{mode}) = (2, 3)$ or (3,2) is $(2/9)\epsilon_B^2$. With this probability, we obtains both $u$ and $v$, and consequently solve the $q_S$-whLRSW Problem instance.

*Exact Security* In summary, we have a probability at least $(2/9)\epsilon^2$ of solving the $q_S$-whLRSW Problem instance, with time complexity twice that of the attacker algorithm $\mathcal{A}$ plus $O(q_S)$. The constant coefficient 2/9 can be further optimized, but we forgo that pursuit in order to simplify our core presentation. □

**Efficiency discussions** The length of signature $\text{Sig}_{CL04B'}$ is 5 $\mathbb{G}_1$ elements and one $Z_{q_1}$ element, for a total of $5(5/2)\lambda_s + 2\lambda_2 = (29/2)\lambda_s$ bits according to [23]. The online verification complexity is 10 pairings, plus one exponentiation.

## 6 Discussions

Using our variable-length coding technique for $k$ in $\text{Sig}_{PSR}$, we can improve the efficiency of Zhang, et al. [35]'s second signature with the modification below, named $\text{Sig}_{SR'}$. Let $\hat{\mathbf{e}} : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_3$ be a pairing, order$(\mathbb{G}_1) = q_1$ be a prime, $g$ be a generator of $\mathbb{G}_1$.

Signature $\mathbf{Sig}_{SR'}$:

1. $\mathsf{sk} = x$, $\mathsf{pk} = (g, g^x, \hat{\mathbf{e}}, \mathcal{H}, \mathsf{crs}, \ell)$. Note the *Common Reference String* is denoted $\mathsf{crs} = r_1 r_2 r_3 \cdots$.
2. *Signing Protocol*: Upon inputs $\mathsf{sk}$ and message $m \in \{0,1\}^\ell$, compute the smallest nonnegative integer $k$ such that $x + (r_k \cdots r_1 || m) \in QR(q_1)$. Output the signature $('0^k1', \sigma = g^{(x+(r_k \cdots r_1 || m))^{1/2}})$. Randomly choose either square root. Note the binary string $'0^k1'$ consists of $k$ zeros followed by a one.
3. *Verification Protocol*: Upon recieving sigmature $('0^k1', \sigma)$ for message $m$, recover $k$ from the first entry, and verify $\hat{\mathbf{e}}(\sigma, \sigma) = \hat{\mathbf{e}}(g^{x+(r_k \cdots r_1 || m)}, g) = \hat{\mathbf{e}}(g^x, g)\hat{\mathbf{e}}(g, g)^{(r_k \cdots r_1 || m)}$.

Its correctness is straightforward. Its ACP-UF (existential unforgeability against adaptive-chosen-plaintext attackers) can be proved similar to [35]'s Theorem 2. The expected value of $k$ is $\langle k \rangle \sum_{i=1}^{\infty} k 2^{-k} = 2$. The signature length is one $\mathbb{G}_1$ element plus $1 + \langle k \rangle = 3$ bits, or $(5/2)\lambda_s + 3$ bits according to [23]. The Signing complexity is two square-root tests in $Z_{q_1}$ and one exponentiation in $\mathbb{G}_1$. The Verification complexity can be optimized by this technique

$$\hat{\mathbf{e}}(g^{x+(r_k \cdots r_1 || m)}, g) = \hat{\mathbf{e}}(g^x, g)\hat{\mathbf{e}}(g, g)^{(r_k \cdots r_1 || m)} = \hat{\mathbf{e}}(g^x, g)\hat{\mathbf{e}}(g, g)^m \prod_{i=1}^{k} \hat{\mathbf{e}}(g, g)^{r_i 2^{\ell+i}}$$

Note only $\hat{\mathbf{e}}(g,g)^m$ cannot be pre-computed, and the expected value of $k$ is 2. Then the online verification cost is just one pairing and one $\mathbb{G}_3$ exponentiation. For $\lambda_s = 128$ (resp. 256), signature $\text{Sig}_{SR'}$ is 323 (resp. 643) bits long, and its online verification costs is one pairing with 320-bit (resp. 640-bit) $\mathbb{G}_1$ elements and one exponentiation with 3072-bit (resp. 15360-bit) $\mathbb{G}_3$ elements according to [23]'s Table 1.

## 7 Conclusions

We presented three new signatures without random oracles, and reduced their securities to new or old intractability assumptions. Two of our signatures are shorter than the previous state-of-the-art short signatures without random oracles.

The following remain interesting open problems: more varieties of efficient ordinary signatures without random oracles, and efficient signatures for specific applications without random oracles, such as ring signatures, group signatures, blind signatures, group-oriented signatures, identity-based signatures, ..., etc.

**Acknowledgements** to Jin Li and Aldar Chan for helpful discussions, and to Hong Kong Earmarked Grants 4232-03E and 4328-02E for financial support.

## References

1. Boaz Barak. How to go beyond the black-box simuation barrier. In *42d FOCS*, pages 106–115. IEEE Computer Soceity, 2001. Also http://www.wisdom.weizmann.ac.il/ boaz.
2. Boaz Barak, Yehuda Lindell, and Salil P. Vadhan. Lower bounds for non-black-box zero knowledge. In *FOCS 2003*, pages 384–393. IEEE Computer Soceity, 2003.
3. M. Bellare and P. Rogaway. Random oracles are practical: A paradigm for designing efficient protocols. In *Proc. 1st ACM Conference on Computer and Communications Security*, pages 62–73. ACM Press, 1993.
4. Mihir Bellare and Oded Goldreich. On defining proofs of knowledge. In *CRYPTO 1992*, volume 740 of *LNCS*, pages 390–420, 1992.
5. Mihir Bellare and Oded Goldreich. Proving computational ability. manuscript, 2005. http://www-cse.ucsd.edu/users/mihir/papers/pok.html.
6. D. Boneh and X. Boyen. Short signatures without random oracles. In *Proc. EUROCRYPT 2004*, pages 56–73. Springer-Verlag, 2004. Lecture Notes in Computer Science No. 3027.
7. D. Boneh, X. Boyen, and H. Shacham. Short group signatures. In *Proc. CRYPTO 2004*, pages 41–55. Springer-Verlag, 2004. Lecture Notes in Computer Science No. 3152.
8. Dan Boneh, Ben Lynn, and Hovav Shacham. Short signatures from the Weil pairing. In *ASIACRYPT 2001*, volume 2248 of *LNCS*, pages 514–532. Springer-Verlag, 2001.
9. Dan Boneh, Ilya Mironov, and Victor Shoup. A secure signature scheme from bilinear maps. In *CT-RSA 2003*, pages 98–110, 2003.
10. J. Camenisch and A. Lysyanskaya. Signature schemes and anonymous credentials from bilinear maps. In *Proc. CRYPTO 2004*, volume 3152 of *LNCS*, pages 56–72. Springer-Verlag, 2004.
11. R. Cramer and I. Damgaard. Secure signature schemes based on interactive protocols. In *CRYPTO95*, pages 297–310, 1995.
12. R. Cramer and I. Damgaard. New generation of secure and practical rsabased signature. In *CRYPTO96*, pages 173–185, 1996.
13. R. Cramer and V. Shoup. Signature schemes based on the strong RSA assumption. In *ACM CCS99*, pages 161–185, 1999. Full version appeared in ACM TISSEC, v. 3(3), pp. 161C185, 2000.
14. C. Dwork and M. Naor. An efficient existentially unforgeable signature scheme and its applications. In *CRYPTO94*, pages 234–246, 1994. Full version appeared in J. of Cryptology, v. 11(2), pp. 187C208, 1998.
15. U. Feige, A. Fiat, and A. Shamir. Zero-knowledge proofs of identity. *Journal of Cryptology*, 1(2):77–94, 1988.
16. U. Feige and A. Shamir. Witness indistinguishable and witness hiding protocols. In *Proc. 22nd Annual ACM Sympsoium on Theory of Computing*, pages 416–426. ACM Press, 1990.

17. R. Gennaro, S. Halevi, and T. Rabin. Secure hash-and-sign signatures without the random oracle. In *Eurocrypt99*, pages 123–139, 1999.
18. O. Goldreich. *Foundations of Cryptography*, volume volumes 1 and 2. Cambridge Univesity Press, 2001 and 2005.
19. Oded Goldreich, Amit Sahai, and Salil P. Vadhan. Honest-verifier statistical zero-knowledge equals general statistical zero-knowledge. In *STOC'98*, pages 399–408, 1998.
20. S. Goldwasser, S. Micali, and R. Rivest. A paradoxical solution to the signature problem (extended abstract). In *FOCS'84*, page 441C448, 1984. Journal version in [21].
21. S. Goldwasser, S. Micali, and R. Rivest. A digital signature scheme secure against adaptive chosen-message attacks. *SIAM J. on Computing*, 17(2):281C308, 1988.
22. Shafi Goldwasser and Yael Tauman Kalai. On the (in)security of the Fiat-Shamir paradigm. In *FOCS 2003*, pages 102–. IEEE Computer Soceity, 2003.
23. N. Koblitz and A. Menezes. Pairing-based cryptography at high security levels. In *10th IMA International Conference*, volume 3796 of *LNCS*, pages 13–36, 2005.
24. A. Lysyanskaya. Unique signatures and verifiable random functions from DH-DDH separation. In *CRYPTO02*, pages 597–612, 2002.
25. Anna Lysyanskaya, Ronald L. Rivest, Amit Sahai, and Stefan Wolf. Pseudonym systems. In *Selected Areas in Cryptography (SAC) 1999*, volume 1758 of *LNCS*, pages 184–199. Springer-Verlag, 1999.
26. W. Mao. *Modern Cryptography: Theory and Practice*. Pearson Education, 2004.
27. A. J. Menezes, P. C. van Oorschot, and S. A. Vanstone. *Handbook of Applied Cryptography*. CRC Press LLC, 1996.
28. S. Mitsunari, R. Sakai, and M. Kasahara. A new traitor tracing. *IEICE Trans. Fundamentals*, E85-A(2):481–484, 2002.
29. M. Naor and M. Yung. Universal one-way hash functions and their cryptographic applications. In *STOC'89*, pages 33–43, 1989.
30. T. Okamoto. Provably secure and practical identification schemes and corresponding signature schemes. In *Proc. CRYPTO 92*, pages 31–53. Springer-Verlag, 1993. Lecture Notes in Computer Science No. 740.
31. D. Pointcheval and J. Stern. Provably secure blind signature shcemes. In *Proc. ASIACRYPT 96*, pages 252–265. Springer-Verlag, 1996. Lecture Notes in Computer Science No. 1163.
32. D. Pointcheval and J. Stern. Security proofs for signature schemes. In *Proc. EUROCRYPT 96*, pages 387–398. Springer-Verlag, 1996. Lecture Notes in Computer Science No. 1070.
33. Victor K. Wei. Tight reductions among strong Diffie-Hellman Assumptions. Cryptology ePrint Archive, Report 2005/057, 2005. http://eprint.iacr.org/.
34. F. Zhang, R. Safavi-Naini, and W. Susilo. An efficient signature scheme from bilinear pairings and its applications. In *Proc. PKC'2004*, pages 277–290. Springer-Verlag, 2004. Lecture Notes in Computer Science No. 2947.
35. Fangguo Zhang, Xiaofeng Chen, Willy Susilo, and Yi Mu. A new short signature scheme without random oracles from bilinear pairings. Cryptology ePrint Archive, Report 2005/386, 2005. http://eprint.iacr.org/.