# Concise Multi-challenge CCA-Secure Encryption and Signatures with Almost Tight Security

Benoît Libert, Marc Joye, Moti Yung, Thomas Peters

# Concise Multi-Challenge CCA-Secure Encryption and Signatures with Almost Tight Security*

Benoît Libert[1] **, Marc Joye[2], Moti Yung[3], and Thomas Peters[4] ***

[1] Ecole Normale Supérieure de Lyon, Laboratoire de l'Informatique du Parallélisme (France)
[2] Technicolor (USA)
[3] Google Inc. and Columbia University (USA)
[4] Université catholique de Louvain, Crypto Group (Belgium)

**Abstract.** To gain strong confidence in the security of a public-key scheme, it is most desirable for the security proof to feature a *tight* reduction between the adversary and the algorithm solving the underlying hard problem. Recently, Chen and Wee (Crypto'13) described the first Identity-Based Encryption scheme with almost tight security under a standard assumption. Here, "almost tight" means that the security reduction only loses a factor $O(\lambda)$ – where $\lambda$ is the security parameter – instead of a factor proportional to the number of adversarial queries. Chen and Wee also gave the shortest signatures whose security almost tightly relates to a simple assumption in the standard model. Also recently, Hofheinz and Jager (Crypto'12) constructed the first CCA-secure public-key encryption scheme in the multi-user setting with tight security. These constructions give schemes that are significantly less efficient in length (and thus, processing) when compared with the earlier schemes with loose reductions in their proof of security. Hofheinz and Jager's scheme has a ciphertext of a few hundreds of group elements, and they left open the problem of finding truly efficient constructions. Likewise, Chen and Wee's signatures and IBE schemes are somewhat less efficient than previous constructions with loose reductions from the same assumptions. In this paper, we consider space-efficient schemes with security almost tightly related to standard assumptions. As a step in solving the open question by Hofheinz and Jager, we construct an efficient CCA-secure public-key encryption scheme whose chosen-ciphertext security in the multi-challenge, multi-user setting almost tightly relates to the DLIN assumption (in the standard model). Quite remarkably, the ciphertext size decreases to 69 group elements under the DLIN assumption whereas the best previous solution required about 400 group elements. Our scheme is obtained by taking advantage of a new almost tightly secure signature scheme (in the standard model) we develop here and which is based on the recent concise proofs of linear subspace membership in the quasi-adaptive non-interactive zero-knowledge setting (QA-NIZK) defined by Jutla and Roy (Asiacrypt'13). Our signature scheme reduces the length of the previous such signatures (by Chen and Wee) by 37% under the Decision Linear assumption, by almost 50% under the $K$-LIN assumption, and it becomes only 3 group elements long under the Symmetric eXternal Diffie-Hellman assumption. Our signatures are obtained by carefully combining the proof technique of Chen and Wee and the above mentioned QA-NIZK proofs.

**Keywords.** CCA-secure encryption, multi-user, multi-challenge, signature, IND-CCA2 security, QA-NIZK proofs, tight security, efficiency.

## 1 Introduction

Security of public-key cryptographic primitives is established by demonstrating that any successful probabilistic polynomial time (PPT) adversary $\mathcal{A}$ implies a PPT algorithm $\mathcal{B}$ solving an (assumed) hard problem. In order to be convincing, such "reductionist" arguments should be as *tight* as possible. Ideally, algorithm $\mathcal{B}$'s probability of success should be about as large as the adversary's advantage. The results of Bellare and Rogaway [9] initiated an important body of work devoted to the design of primitives validated by tight security reductions in the random oracle model [22,23,39,20,21,10,24,49,1,38] and in the standard model [21,7,49].

Tight security proofs may be hard to achieve and are even known not to exist at all in some

* This is the full version of a paper published at Asiacrypt 2014.
** Part of this work was done while this author was at Technicolor (France).
*** This author was supported by the CAMUS Walloon Region Project.

situations [23,38,34]. On the positive side, long-standing open problems have been resolved in the recent years. Hofheinz and Jager [32] showed the first public-key encryption scheme whose chosen-ciphertext security [46,47] in the multi-user setting tightly relates to a standard hardness assumption, which solved a problem left open by Bellare, Boldyreva and Micali [6] although their ciphertext is a few hundreds group elements long. Chen and Wee [25] answered an important open question raised by Waters [53] by avoiding the concrete security loss, proportional to the number of adversarial queries, that affected the security reductions of all prior identity-based encryption (IBE) [15,50] schemes based on simple assumptions, including those based on the dual system paradigm [54,41]. The results of [25] also implied the shortest signatures almost tightly related to simple assumptions[5] in the standard model. In the terminology of [25], "almost tight security" refers to reductions where the degradation factor only depends on the security parameter $\lambda$, and not on the number $q$ of adversarial queries, which is potentially much larger as it is common to assume $\lambda = 128$ and $q \approx 2^{30}$.

The tighter security results of Chen and Wee [25] overcame an important barrier since, as pointed out in [25], all earlier short signatures based on standard assumptions in the standard model [53,35,33,55,56,12] suffered a $\Theta(q)$ loss in terms of exact security. On the other hand, the Chen-Wee schemes are less efficient than previous solutions based on similar assumptions [53,41,19,12]. Likewise, encryption schemes with tight multi-challenge chosen-ciphertext security [32,5] come at the expense of much longer ciphertexts than constructions (e.g., [26]) in the single-challenge setting.[6] In order to exploit concrete security improvements in the choice of parameters, it is desirable to keep schemes as efficient —from both computational and space viewpoints— as their counterparts backed by loose reductions. This paper aims at rendering the constructions and techniques of [32,25] truly competitive with existing signatures and encryption schemes based on simple assumptions in the standard model.

Our Contributions. In this paper, we construct a new public-key encryption scheme with almost tight chosen-ciphertext (IND-CCA2) security in the multi-user, multi-challenge setting [6] under the DLIN assumption. As in the setting of Chen and Wee, the underlying reduction is not as tight as those of [32,5] since we lose a factor of $O(\lambda)$. On the other hand, our construction provides much shorter ciphertexts than previous tightly IND-CCA2-secure systems [32,5] based on the same assumption. Moreover, our security bound does not depend on the number of users or the number of challenges, so that our scheme can be safely instantiated in environments involving arbitrarily many users encrypting as many ciphertexts as they like.

As an tool for achieving our encryption scheme (and a result of independent interest), we devise a variant of the Chen-Wee signature scheme [25], which has been proved almost tightly secure under the DLIN assumption, with shorter signatures in prime-order groups. Under the DLIN assumption, each signature consists of 6 groups elements, instead of 8 in [25]. Under the $K$-linear assumption (which is believed weaker than DLIN when $K > 2$), we reduce the signature length of [25] from $4K$ to $2K + 2$ and thus save $\Theta(K)$ group elements.

By combining our technique and the recent non-interactive proof systems of Jutla and Roy [37], we can further shorten our signatures and obtain of 5 group elements per signature under the DLIN assumption and $2K + 1$ elements under the $K$-linear assumption. Our DLIN-based (resp. $K$-linear-based) system thus improves upon the Chen-Wee constructions [25] by 37% (resp. nearly 50%) in terms of signature length. Under the Symmetric eXternal Diffie-Hellman assumption (namely, the hardness of DDH in $\mathbb{G}$ and $\hat{\mathbb{G}}$ for asymmetric pairings $e : \mathbb{G} \times \hat{\mathbb{G}} \to \mathbb{G}_T$), the same optimizations yield signatures comprised of only 3 group elements, which only exceeds the length of Waters signatures [53]

[5] By "simple assumptions", we mean non-interactive (and thus falsifiable [44]) assumptions that can be described using a *constant* number of group elements. In particular, the number of input elements in the description of the assumption does not depend on the number of adversarial queries.

[6] Using a hybrid argument, Bellare, Boldyreva and Micali [6] showed that any CCA2-secure encryption scheme in the single-challenge setting remains secure if the adversary is given arbitrarily many challenge ciphertexts. However, the reduction is linearly affected by the number $q$ of challenge ciphertexts.

by one group element. Since the SXDH-based signatures of [25] live in $\mathbb{G}^4$, we also shorten them by one element (or 25%) under the same assumption. Our SXDH-based scheme turns out to be the shortest known signature with nearly tight security under a simple assumption.

While randomizable in their basic variant, our schemes can be made strongly unforgeable in a direct manner, without any increase of the signature length. In particular, we do not need generic transformations based on chameleon hash functions, such as the one of Boneh *el al.* [16], which tend to lengthen signatures by incorporating the random coins of the chameleon hashing algorithm. Using the SXDH assumption and asymmetric pairings, we thus obtain the same signature length as the CDH-based strongly unforgeable signatures of Boneh, Shen and Waters [16] with the benefit of a much better concrete security (albeit under a stronger assumption).

Our signature schemes readily apply to construct a new efficient public-key encryption scheme with almost tight chosen-ciphertext (IND-CCA) security in the multi-user, multi-challenge setting [6]. Indeed, they easily lend themselves to the construction of new unbounded simulation-sound proof systems (where the adversary remains unable to prove false statements after having seen polynomially many simulated proofs for possibly false statements) with almost tight security. In turn, this yields the most efficient constructions, to date, of IND-CCA-secure public-key encryption schemes in the multi-challenge setting. By following the approach of [30,32], we obtain an almost tightly simulation-sound proof system by showing that either: *(i)* a set of pairing product equations is satisfiable; and *(ii)* committed group elements form a valid signature on the verification key of a one-time signature. In this case, our randomizable signatures are very interesting candidates since they reduce the number of signature components that must appear in committed form. In addition, the specific algebraic properties of our signature scheme make it possible to construct an optimized simulation-extractable proof system that allows proving knowledge of the plaintext using only 62 group elements, which reduces our ciphertexts to only 69 group elements under the DLIN assumption. This dramatically improves upon previous tightly secure constructions based on the same assumption [32,5] which require several hundreds of group elements per ciphertext. Moreover, unlike [5], our system can also be instantiated in asymmetric pairing configurations. We stress that, unlike [43] (which has a loose security reduction), our simulation-sound proof system does not provide constant-size proofs of linear subspace membership. Still, for the specific application of nearly tight CCA-security, our proof system suffices to obtain relatively concise ciphertexts.

Concurrent to our work, Blazy, Kiltz and Pan [11] independently gave different constructions of signature schemes with tight security under the SXDH, DLIN and other simple assumptions. Their technique extends to provide (hierarchical) identity-based encryption schemes. Under the DLIN and SXDH assumption, our optimized signatures are as short as theirs. Our approach bears similiarities with theirs in that each signature can be seen as a NIZK proof that a message authentication code is valid w.r.t. a committed key.

Our Techniques. Underlying our results is a methodology of getting security proofs with a short chain of transitions from actual games to ideal ones. Our constructions build upon a signature scheme of Jutla and Roy [36, Section 5], which is itself inspired by [17, Appendix A.3]. In [36], each signature is a CCA2-secure encryption of the private key, where the message is included in the label [52] of the ciphertext. The signer also computes a non-interactive zero-knowledge proof that the encrypted value is the private key. The security proof uses the dual system encryption method [54,40,29] and proceeds with a sequence of hybrid games heading for a game where all signatures encrypt a random value while the NIZK proofs are simulated.

While Camenisch *el al.* [17] used Groth-Sahai proofs, Jutla and Roy obtained a better efficiency using *quasi-adaptive* NIZK (QA-NIZK) proofs, i.e., where the common reference string (CRS) may depend on the specific language for which proofs are being generated but a single CRS simulator works for the entire class of languages. For the common task of proving that a vector of $n$ group

elements belongs to a linear subspace of rank $t$, Jutla and Roy [36] gave computationally sound QA-NIZK proofs of length $\Theta(n-t)$ where the Groth-Sahai (GS) techniques entail $\Theta(n+t)$ group elements per proof. They subsequently refined their techniques, reducing the proof's length to a constant [37], regardless of the number of equations or the number of variables. Libert *el al.* [43] independently obtained similar improvements using different techniques.

Our signature schemes rely on the observation that the constant-size QA-NIZK proofs of [43,37] make it possible to encode the label (which contains the message) in a bit-by-bit manner without affecting the signature length. In turn, this allows applying the technique of Chen and Wee [25] so as to avoid the need for $q$ transitions, where $q$ is the number of signing queries. As in the security proof of [25], the signing oracle uses a semi-functional private key which is obtained by shifting a normal private key by a factor consisting of a random function that depends on increasingly many bits of the message in each transition. In the last game, the random function depends on all the message bits, so that the shifting factor is thus totally unpredictable by the adversary.

Our construction of almost tightly CCA2-secure encryption scheme is based on a modification of the Naor-Yung [46] paradigm due to [27,5]. The latter consists in combining an IND-CPA encryption and a simulation-extractable proof of knowledge of the plaintext. In order to build an optimized simulation-extractable proof, we take advantage of the simple algebraic structure of our signature scheme and its randomizability properties. Our proof system is a simplification of the one in [5] and shows that either: (i) A commitment is an extractable commitment to a function of the encryption exponents; or (ii) Another commitment contained in the proof contains a valid signature on the verification key of a one-time signature. Our signature scheme allows implementing this very efficiently. Specifically, a real proof used by the encryption algorithm involves a commitment to a pseudo-signature – which can be generated without the signing key – whereas a simulated proof uses a real signature instead of a pseudo-signature. The perfect witness indistinguishability of Groth-Sahai proofs on a NIWI CRS guarantees that the adversary will not be able to distinguish committed pseudo-signatures from real signatures.

## 2 Background and Definitions

### 2.1 Hardness Assumptions

We consider groups $(\mathbb{G}, \hat{\mathbb{G}}, \mathbb{G}_T)$ of prime-order $p$ endowed with a bilinear map $e : \mathbb{G} \times \hat{\mathbb{G}} \to \mathbb{G}_T$. In this setting, we rely on the standard Decision Linear assumption, which is a special case of the $K$-linear assumption (see Definition 4 in Appendix E)) for $K = 2$.

**Definition 1 ([14]).** *The* Decision Linear Problem *(DLIN) in $\mathbb{G}$, is to distinguish the distributions $(g^a, g^b, g^{ac}, g^{bd}, g^{c+d})$ and $(g^a, g^b, g^{ac}, g^{bd}, g^z)$, with $a, b, c, d \xleftarrow{R} \mathbb{Z}_p$, $z \xleftarrow{R} \mathbb{Z}_p$. The Decision Linear assumption asserts the intractability of DLIN for any PPT distinguisher.*

It will sometimes be convenient to use the following assumption, which is weaker than DLIN. As noted in [18], any algorithm solving SDP immediately yields a DLIN distinguisher.

**Definition 2.** *The* Simultaneous Double Pairing problem *(SDP) in $(\mathbb{G}, \hat{\mathbb{G}}, \mathbb{G}_T)$ is, given a tuple of group elements $(\hat{g_z}, \hat{g_r}, \hat{h_z}, \hat{h_u}) \in \hat{\mathbb{G}}^4$, to find a non-trivial triple $(z, r, u) \in \mathbb{G}^3 \backslash \{(1_\mathbb{G}, 1_\mathbb{G}, 1_\mathbb{G})\}$ such that $e(z, \hat{g_z}) \cdot e(r, \hat{g_r}) = 1_{\mathbb{G}_T}$ and $e(z, \hat{h_z}) \cdot e(u, \hat{h_u}) = 1_{\mathbb{G}_T}$.*

### 2.2 One-Time Linearly Homomorphic Structure-Preserving Signatures

Structure-preserving signatures [3,2] are signature schemes where messages and public keys all consist of elements of a group over which a bilinear map $e : \mathbb{G} \times \hat{\mathbb{G}} \to \mathbb{G}_T$ is efficiently computable.

Constructions based on simple assumptions were put forth in [4,5].

Libert *el al.* [42] considered structure-preserving with linear homomorphic properties (see Appendix B for formal definitions). This section recalls the one-time linearly homomorphic structure-preserving signature (LHSPS) of [42].

**Keygen($\lambda, n$):** Given a security parameter $\lambda$ and the dimension $n \in \mathbb{N}$ of the subspace to be signed, choose bilinear group $(\mathbb{G}, \hat{\mathbb{G}}, \mathbb{G}_T)$ of prime order $p$. Then, choose $\hat{g_z}, \hat{g_r}, \hat{h_z}, \hat{h_u} \xleftarrow{R} \hat{\mathbb{G}}$. For $i = 1$ to $n$, pick $\chi_i, \gamma_i, \delta_i \xleftarrow{R} \mathbb{Z}_p$ and compute $\hat{g}_i = \hat{g_z}^{\chi_i} \hat{g_r}^{\gamma_i}$, $\hat{h}_i = \hat{h_z}^{\chi_i} \hat{h_u}^{\delta_i}$. The private key is defined to be $\mathsf{sk} = \{(\chi_i, \gamma_i, \delta_i)\}_{i=1}^n$ while the public key is $\mathsf{pk} = \big(\hat{g_z},\ \hat{g_r},\ \hat{h_z},\ \hat{h_u},\ \{(\hat{g}_i, \hat{h}_i)\}_{i=1}^n\big) \in \hat{\mathbb{G}}^{2n+4}$.

**Sign($\mathsf{sk}, (M_1, \ldots, M_n)$):** To sign a vector $(M_1, \ldots, M_n) \in \mathbb{G}^n$ using $\mathsf{sk} = \{(\chi_i, \gamma_i, \delta_i)\}_{i=1}^n$, output $\sigma = (z, r, u) \in \mathbb{G}^3$, where $z = \prod_{i=1}^n M_i^{-\chi_i}$, $r = \prod_{i=1}^n, M_i^{-\gamma_i}$ and $u = \prod_{i=1}^n M_i^{-\delta_i}$.

**SignDerive($\mathsf{pk}, \{(\omega_i, \sigma^{(i)})\}_{i=1}^{\ell}$):** given $\mathsf{pk}$ as well as $\ell$ tuples $(\omega_i, \sigma^{(i)})$, parse $\sigma^{(i)}$ as $\sigma^{(i)} = \big(z_i, r_i, u_i\big)$ for $i = 1$ to $\ell$. Compute and return $\sigma = (z, r, u)$, where $z = \prod_{i=1}^{\ell} z_i^{\omega_i}$, $r = \prod_{i=1}^{\ell} r_i^{\omega_i}$, $u = \prod_{i=1}^{\ell} u_i^{\omega_i}$.

**Verify($\mathsf{pk}, \sigma, (M_1, \ldots, M_n)$):** Given a signature $\sigma = (z, r, u) \in \mathbb{G}^3$ and a vector $(M_1, \ldots, M_n)$, return 1 if and only if $(M_1, \ldots, M_n) \neq (1_{\mathbb{G}}, \ldots, 1_{\mathbb{G}})$ and $(z, r, u)$ satisfy

$$1_{\mathbb{G}_T} = e(z, \hat{g_z}) \cdot e(r, \hat{g_r}) \cdot \prod_{i=1}^{n} e(M_i, \hat{g}_i)\,, \qquad 1_{\mathbb{G}_T} = e(z, \hat{h_z}) \cdot e(u, \hat{h_u}) \cdot \prod_{i=1}^{n} e(M_i, \hat{h}_i)\ .$$

The one-time security of the scheme (in the sense of Definition 3 in Appendix B) was proved [42] under the SDP assumption under a tight reduction. In short, the security notion implies the infeasibility of deriving a signature on a vector outside the subspace spanned by the vectors authenticated by the signer. Here, "one-time" security means that a given public key allows signing only one subspace.

## 3 Shorter Signatures Almost Tightly Related to the DLIN Assumption

This section shows that LHSPS schemes and constant-size QA-NIZK proofs for linear subspaces can be used to construct shorter signatures with nearly optimal reductions under the DLIN assumption.

The scheme builds on ideas used in a signature scheme suggested by Jutla and Roy [36, Section 5], where each signature is a CCA2-secure encryption —using the message to be signed as a label— of the private key augmented with a QA-NIZK proof (as defined in [36] and recalled in Appendix A) that the encrypted value is a persistent hidden secret. As in [54,40,29], the security proof uses a sequence of hybrid games which gradually moves to a game where all signatures contain an encryption of a random value while the QA-NIZK proofs are simulated. At each step of the transition, increasingly many signatures are generated without using the private key and the CCA2-security of the encryption scheme ensures that this should not affect the adversary's probability to output a signature that does encrypt the private key. In the security proof of [36], the latter approach implies that: *(i)* the number of transitions depends on the number of signing queries; and *(ii)* a CCA2-secure encryption scheme is needed since, at each transition, the reduction has to decrypt the ciphertext contained in the forgery.

Here, our key observation is that, by using a QA-NIZK proof system where the proof length is independent of the dimension of the considered linear subspace, the approach of [36] can be combined with the proof technique of Chen and Wee [25] so as to reduce the number of game transitions while retaining short signatures. In addition, the techniques of [25] allow us to dispense with the need for a CCA2-secure encryption scheme. The security analysis actually departs from that of [36] and rather follows the one of Chen and Wee [25]. The techniques of [36,17,29] argue that, even if the adversary is given signatures where the private key is blinded by a semi-functional component, its forgery will retain the distribution of a normal signature unless some indistinguishability assumption is broken. Here, we follow [25] and blind the outputs of the signing oracle by a random function of increasingly many bits of the message. Instead of using the same argument as in [36], however, we argue that

the adversary's forgery will always have the same distribution as the signatures produced by the signing oracle. In the last game of the hybrid sequence, we prove that the adversary cannot retain the same behavior as the signing oracle since the latter's outputs are blinded by a random function of *all* the message bits. In order to come up with the same kind of signature as the signing oracle, the adversary would have to predict the value of the random function on the forgery message $M^\star$, which is information-theoretically infeasible.

As in [25], by guessing exactly one bit of the target message, the reduction can efficiently test whether the forgery has the same distribution as outputs of the signing oracle while remaining able to embed a DLIN instance in outputs of signing queries. For $L$-bit messages, by applying arguments similar to those of [45,25], we need $L$ game transitions to reach a game where each signature encrypts a random —and thus unpredictable— function of the message. As a result, we obtain DLIN-based signatures comprised of only 6 group elements.

**Keygen($\boldsymbol{\lambda}$):** Choose bilinear groups $(\mathbb{G}, \hat{\mathbb{G}}, \mathbb{G}_T)$ of prime order $p$ together with $f, g, h, u_1, u_2 \stackrel{R}{\leftarrow} \mathbb{G}$.

1. For $\ell = 1$ to $L$, choose $V_{\ell,0}, V_{\ell,1}, W_{\ell,0}, W_{\ell,1} \stackrel{R}{\leftarrow} \mathbb{G}$ to assemble row vectors
$$\boldsymbol{V} = (V_{1,0}, V_{1,1}, \ldots, V_{L,0}, V_{L,1}) \in \mathbb{G}^{2L}, \qquad \boldsymbol{W} = (W_{1,0}, W_{1,1}, \ldots, W_{L,0}, W_{L,1}) \in \mathbb{G}^{2L} .$$
2. Define the matrix $\mathbf{M} = \big(M_{i,j}\big)_{i,j}$ given by
$$\mathbf{M} = \left(\begin{array}{c|c|c|c|c} \boldsymbol{V}^\top & \mathbf{Id}_{f,2L} & \mathbf{1}^{2L\times 2L} & \mathbf{1}^{2L\times 1} & \mathbf{1}^{2L\times 1} \\ \hline \boldsymbol{W}^\top & \mathbf{1}^{2L\times 2L} & \mathbf{Id}_{h,2L} & \mathbf{1}^{2L\times 1} & \mathbf{1}^{2L\times 1} \\ \hline g & \mathbf{1}^{1\times 2L} & \mathbf{1}^{1\times 2L} & u_1 & 1 \\ \hline g & \mathbf{1}^{1\times 2L} & \mathbf{1}^{1\times 2L} & 1 & u_2 \end{array}\right) \in \mathbb{G}^{(4L+2)\times(4L+3)} \qquad (1)$$
with $\mathbf{Id}_{f,2L} = f^{\mathbf{I}_{2L}} \in \mathbb{G}^{2L\times 2L}$, $\mathbf{Id}_{h,2L} = h^{\mathbf{I}_{2L}} \in \mathbb{G}^{2L\times 2L}$, where $\mathbf{I}_{2L} \in \mathbb{Z}_p^{2L\times 2L}$ is the identity matrix.
3. Generate a key pair $(\mathsf{sk}_{hsps}, \mathsf{pk}_{hsps})$ for the one-time linearly homomorphic signature of Section 2.2 in order to sign vectors of dimension $n = 4L+3$. Let $\mathsf{sk}_{hsps} = \{(\chi_i, \gamma_i, \delta_i)\}_{i=1}^{4L+3}$ be the private key, of which the corresponding public key is $\mathsf{pk}_{hsps} = \big(\hat{g_z}, \hat{g_r}, \hat{h_z}, \hat{h_u}, \{(\hat{g}_i, \hat{h}_i)\}_{i=1}^{4L+3}\big)$.
4. Using $\mathsf{sk}_{hsps} = \{\chi_i, \gamma_i, \delta_i\}_{i=1}^{4L+3}$, generate one-time homomorphic signatures $\{(Z_j, R_j, U_j)\}_{j=1}^{4L+2}$ on the rows $\boldsymbol{M}_j = (M_{j,1}, \ldots, M_{j,4L+3}) \in \mathbb{G}^{4L+3}$ of $\mathbf{M}$. These are obtained as
$$(Z_j, R_j, U_j) = \left(\prod_{i=1}^{4L+3} M_{j,i}^{-\chi_i}, \prod_{i=1}^{4L+3} M_{j,i}^{-\gamma_i}, \prod_{i=1}^{4L+3} M_{j,i}^{-\delta_i}\right), \qquad \forall j \in \{1, \ldots, 4L+2\}$$
and, as part of the common reference string for the QA-NIZK proof system of [43], they will be included in the public key.
5. Choose $\omega_1, \omega_2 \stackrel{R}{\leftarrow} \mathbb{Z}_p$ and compute $\Omega_1 = u_1^{\omega_1} \in \mathbb{G}$, $\Omega_2 = u_2^{\omega_2} \in \mathbb{G}$.

The private key consists of $SK = (\omega_1, \omega_2)$ and the public key is
$$PK = \Big(f,\ g,\ h,\ u_1,\ u_2,\ \Omega_1,\ \Omega_2,\ \boldsymbol{V},\ \boldsymbol{W},\ \mathsf{pk}_{hsps} = \big(\hat{g_z},\ \hat{g_r},\ \hat{h_z},\ \hat{h_u},\ \{(\hat{g}_i, \hat{h}_i)\}_{i=1}^{4L+3}\big), \{(Z_j, R_j, U_j)\}_{j=1}^{4L+2}\Big) .$$

**Sign($\boldsymbol{SK}, \boldsymbol{M}$):** Given an $L$-bit message $M = M[1] \ldots M[L] \in \{0,1\}^L$ and $SK = (\omega_1, \omega_2)$:

1. Choose $r, s \stackrel{R}{\leftarrow} \mathbb{Z}_p$ and compute
$$\sigma_1 = g^{\omega_1+\omega_2} \cdot H(\boldsymbol{V}, M)^r \cdot H(\boldsymbol{W}, M)^s , \qquad \sigma_2 = f^r , \qquad \sigma_3 = h^s , \qquad (2)$$
where $H(\boldsymbol{V}, M) = \prod_{\ell=1}^{L} V_{\ell,M[\ell]}$ and $H(\boldsymbol{W}, M) = \prod_{\ell=1}^{L} W_{\ell,M[\ell]}$.

2. Using $\{(Z_j, R_j, U_j)\}_{j=1}^{4L+2}$, derive a one-time homomorphic signature $(Z, R, U)$ which will serve as a non-interactive argument showing that the vector

$$(\sigma_1, \sigma_2^{1-M[1]}, \sigma_2^{M[1]}, \ldots, \sigma_2^{1-M[L]}, \sigma_2^{M[L]}, \sigma_3^{1-M[1]}, \sigma_3^{M[1]}, \ldots, \sigma_3^{1-M[L]}, \sigma_3^{M[L]}, \Omega_1, \Omega_2) \quad (3)$$

is in the row space of $\mathbf{M}$, which ensures that $(\sigma_1, \sigma_2, \sigma_3)$ is of the form (2). Namely, compute

$$\begin{cases} Z = Z_{4L+1}^{\omega_1} \cdot Z_{4L+2}^{\omega_2} \cdot \prod_{i=1}^{L} \left(Z_{2i-\overline{M[i]}}^{r} \cdot Z_{2L+2i-\overline{M[i]}}^{s}\right) \\ R = R_{4L+1}^{\omega_1} \cdot R_{4L+2}^{\omega_2} \cdot \prod_{i=1}^{L} \left(R_{2i-\overline{M[i]}}^{r} \cdot R_{2L+2i-\overline{M[i]}}^{s}\right) \\ U = U_{4L+1}^{\omega_1} \cdot U_{4L+2}^{\omega_2} \cdot \prod_{i=1}^{L} \left(U_{2i-\overline{M[i]}}^{r} \cdot U_{2L+2i-\overline{M[i]}}^{s}\right) \end{cases} . \quad (4)$$

Return the signature $\sigma = \left(\sigma_1, \sigma_2, \sigma_3, Z, R, U\right) \in \mathbb{G}^6$.

**Verify($\boldsymbol{PK}, \boldsymbol{M}, \boldsymbol{\sigma}$):** Parse $\sigma$ as $\left(\sigma_1, \sigma_2, \sigma_3, Z, R, U\right) \in \mathbb{G}^6$ and return 1 if and only if

$$e(Z, \hat{g_z}) \cdot e(R, \hat{g_r}) = e(\sigma_1, \hat{g}_1)^{-1} \cdot e(\sigma_2, \prod_{i=1}^{L} \hat{g}_{2i+M[i]})^{-1} \cdot e(\sigma_3, \prod_{i=1}^{L} \hat{g}_{2L+2i+M[i]})^{-1} \cdot e(\Omega_1, \hat{g}_{4L+2})^{-1} \cdot e(\Omega_2, \hat{g}_{4L+3})^{-1}$$

$$e(Z, \hat{h_z}) \cdot e(U, \hat{h_u}) = e(\sigma_1, \hat{h_1})^{-1} \cdot e(\sigma_2, \prod_{i=1}^{L} \hat{h}_{2i+M[i]})^{-1} \cdot e(\sigma_3, \prod_{i=1}^{L} \hat{h}_{2L+2i+M[i]})^{-1} \cdot e(\Omega_1, \hat{h}_{4L+2})^{-1} \cdot e(\Omega_2, \hat{h}_{4L+3})^{-1} .$$

Each signature consists of 6 elements of $\mathbb{G}$, which is as short as Lewko's DLIN-based signatures [41, Section 4.3] where the security proof incurs a security loss proportional to the number of signing queries. Under the same assumption, the Chen-Wee signatures [25] require 8 group elements.

We emphasize that our security proof allows using any QA-NIZK proof system for linear subspaces and not only the one of [43] (which we used in order to keep the description as simple and self-contained as possible). Our constructions can thus be optimized if we replace the QA-NIZK proof system of [43] —which entails $K+1$ group elements under the $K$-LIN assumption— by those recently suggested by Jutla and Roy, where only $K$ group elements per proof are needed. Under the DLIN (resp. $K$-linear) assumption, each signature is only comprised of 5 (resp. $2K+1$) group elements. We thus shorten signatures by 37% under the DLIN assumption. Under the $K$-Linear assumption, our improvement is more dramatic since, when $K$ increases, our signatures become almost 50% shorter as we reduce the signature length of [25] from $4K$ to $2K+1$.

Under the SXDH assumption (namely, the 1-linear assumption), a direct adaptation of the above scheme entails 4 elements of $\mathbb{G}$ per signature, which is as long as [25]. However, as explained in Appendix E, the QA-NIZK proof system of Jutla and Roy [37] can supersede the one of [43] since, under the SXDH assumption, it only requires one group element per proof, instead of two in [43]. The signature thus becomes a triple $(\sigma_1, \sigma_2, Z) = (u^{\omega} \cdot H(\boldsymbol{V}, M)^r, f^r, Z)$, where $Z$ is a QA-NIZK proof of well-formedness for $(\sigma_1, \sigma_2)$.

**Theorem 1.** *The scheme provides existential unforgeability under chosen-message attacks if the DLIN assumption holds in $\mathbb{G}$ and $\hat{\mathbb{G}}$. For $L$-bit messages, for any adversary $\mathcal{A}$, there exist DLIN distinguishers $\mathcal{B}$ and $\mathcal{B}'$ in $\hat{\mathbb{G}}$ and $\mathbb{G}$ such that*

$$\mathbf{Adv}_{\mathcal{A}}(\lambda) \leq \mathbf{Adv}_{\mathcal{B}}^{\mathrm{DLIN}}(\lambda) + 2 \cdot L \cdot \mathbf{Adv}_{\mathcal{B}'}^{\mathrm{DLIN}}(\lambda) + \frac{2}{p}$$

*and with running times $t_{\mathcal{B}}, t_{\mathcal{B}'} \leq t_{\mathcal{A}} + q \cdot \mathsf{poly}(\lambda, L)$.*

*Proof.* We proceed using a sequence of games where several kinds of signatures will be used.

**Type A signatures** are those produced by the real signing algorithm. If $\boldsymbol{V} = f^{\boldsymbol{v}}$ and $\boldsymbol{W} = h^{\boldsymbol{w}}$ for vectors $\boldsymbol{v} = (v_{1,0}, v_{1,1}, \ldots, v_{L,0}, v_{L,1}) \in \mathbb{Z}_p^{2L}$, $\boldsymbol{w} = (w_{1,0}, w_{1,1}, \ldots, w_{L,0}, w_{L,1}) \in \mathbb{Z}_p^{2L}$ and if we define functions $F(\boldsymbol{v}, M) = \sum_{\ell=1}^{L} v_{\ell,M[\ell]}$ and $F(\boldsymbol{w}, M) = \sum_{\ell=1}^{L} w_{\ell,M[\ell]}$, these signatures are such that

$$g^{\omega_1+\omega_2} = \sigma_1 \cdot \sigma_2^{-F(\boldsymbol{v},M)} \cdot \sigma_3^{-F(\boldsymbol{w},M)}$$

and $(Z, R, U)$ is a valid linearly homomorphic signature on the vector (3).

**Type B signatures** are valid signatures that are not Type A signatures. These are of the form

$$\sigma_1 = g^{\omega_1+\omega_2+\tau} \cdot H(\boldsymbol{V}, M)^r \cdot H(\boldsymbol{W}, M)^s\,, \qquad \sigma_2 = f^r\,, \qquad \sigma_3 = h^s\,,$$

for some $\tau \in_R \mathbb{Z}_p$, $r, s \in_R \mathbb{Z}_p$, and

$$\begin{cases} Z = g^{-\tau\cdot\chi_1} \cdot Z_{4L+1}^{\omega_1} \cdot Z_{4L+2}^{\omega_2} \cdot \prod_{i=1}^{L} \left(Z_{2i-\overline{M[i]}}^{r} \cdot Z_{2L+2i-\overline{M[i]}}^{s}\right) \\ R = g^{-\tau\cdot\gamma_1} \cdot R_{4L+1}^{\omega_1} \cdot R_{4L+2}^{\omega_2} \cdot \prod_{i=1}^{L} \left(R_{2i-\overline{M[i]}}^{r} \cdot R_{2L+2i-\overline{M[i]}}^{s}\right) \\ U = g^{-\tau\cdot\delta_1} \cdot U_{4L+1}^{\omega_1} \cdot U_{4L+2}^{\omega_2} \cdot \prod_{i=1}^{L} \left(U_{2i-\overline{M[i]}}^{r} \cdot U_{2L+2i-\overline{M[i]}}^{s}\right) \end{cases}.$$

Note that Type B signatures also satisfy the verification algorithm since $(Z, R, U)$ is a valid homomorphic signature on the vector (3). The term $g^\tau$ will be henceforth called the *semi-functional component* of the signature. Type B signatures include the following sub-classes.

**Type B-$k$ signatures ($1 \leq k \leq L$)** are generated by choosing $r, s \xleftarrow{R} \mathbb{Z}_p$ and setting

$$\sigma_1 = g^{\omega_1+\omega_2} \cdot R_k(M_{|_k}) \cdot H(\boldsymbol{V}, M)^r \cdot H(\boldsymbol{W}, M)^s\,, \qquad \sigma_2 = f^r\,, \qquad \sigma_3 = h^s\,,$$

where $H(\boldsymbol{V}, M) = \prod_{\ell=1}^{L} V_{\ell,M[\ell]}$ and $H(\boldsymbol{W}, M) = \prod_{\ell=1}^{L} W_{\ell,M[\ell]}$ and $R_k \colon \{0,1\}^k \to \mathbb{G}, M_{|_k} \mapsto R_k(M_{|_k})$ is a random function that depends on the first $k$ bits of $M$. The $(Z, R, U)$ components are simulated QA-NIZK proofs of subspace membership. They are obtained using $\{(\chi_i, \gamma_i, \delta_i)\}_{i=1}^{4L+3}$ to generate a homomorphic structure-preserving signature on the vector (3) by computing

$$\begin{cases} Z = \sigma_1^{-\chi_1} \cdot \sigma_2^{-\sum_{i=1}^{L} \chi_{2i+M[i]}} \cdot \sigma_3^{-\sum_{i=1}^{L} \chi_{2L+2i+M[i]}} \cdot \Omega_1^{-\chi_{4L+2}} \cdot \Omega_2^{-\chi_{4L+3}} \\ R = \sigma_1^{-\gamma_1} \cdot \sigma_2^{-\sum_{i=1}^{L} \sigma_{2i+M[i]}} \cdot \sigma_3^{-\sum_{i=1}^{L} \gamma_{2L+2i+M[i]}} \cdot \Omega_1^{-\gamma_{4L+2}} \cdot \Omega_2^{-\gamma_{4L+3}} \\ U = \sigma_1^{-\delta_1} \cdot \sigma_2^{-\sum_{i=1}^{L} \delta_{2i+M[i]}} \cdot \sigma_3^{-\sum_{i=1}^{L} \delta_{2L+2i+M[i]}} \cdot \Omega_1^{-\delta_{4L+2}} \cdot \Omega_2^{-\delta_{4L+3}} \end{cases}.$$

To prove the result, we consider the following sequence of games. For each $i$, we call $S_i$ the event that the adversary wins in Game $i$. We also define $E_i$ to be the event that, in Game $i$, $\mathcal{A}$'s forgery has the same type as the signatures it observes. Namely, if $\mathcal{A}$ obtains a Type A (resp. Type B-$k$) signature at each query, it should output a Type A (resp. Type $B$-k) forgery.

**Game 0:** This game is the real game. Namely, the adversary obtains Type A signatures at each signing query. At the end of the game, however, the challenger $\mathcal{B}$ checks if $\mathcal{A}$'s forgery is a Type A signature and we define $E_0$ to be the event that the forgery $\sigma^\star$ is a Type A forgery. We obviously have $\Pr[S_0] = \Pr[S_0 \wedge E_0] + \Pr[S_0 \wedge \neg E_0]$. Lemma 1 shows that, if the DLIN assumption holds in $\hat{\mathbb{G}}$, the adversary can only output a Type B signature with negligible probability. We have $\Pr[S_0 \wedge \neg E_0] \leq \mathbf{Adv}_{\hat{\mathbb{G}}}^{\mathrm{DLIN}}(\lambda) + 1/p$. We are thus left with the task of bounding $\Pr[S_0 \wedge E_0]$. To this end, we proceed using a sequence of $L$ games.

**Game 1:** This game is identical to Game 0 with the difference that, at each signing query, the signature components $(Z, R, U)$ are obtained as simulated QA-NIZK proofs of linear subspace membership. Namely, instead of computing $(Z, R, U)$ as per (4), the challenger uses $\{\chi_i, \gamma_i, \delta_i\}_{i=1}^{4L+3}$ to compute $(Z, R, U)$ as a one-time linearly homomorphic signature on the vector (3). Clearly $(Z, R, U)$ retains the same distribution as in Game 0, so that $\mathcal{A}$'s view remains unchanged. We have $\Pr[S_1 \wedge E_1] = \Pr[S_0 \wedge E_0]$, where $E_1$ is the counterpart of event $E_0$ in Game 1.

**Game 2.$k$ ($1 \leq k \leq L$):** In Game 2.$k$, all signing queries are answered by returning Type B-$k$ signatures. For each $k$, we call $E_{2.k}$ the event that $\mathcal{A}$ outputs a Type B-$k$ forgery in Game 2.$k$. Lemma 2 provides evidence that Game 2.1 is computationally indistinguishable from Game 1 under the DLIN assumption in $\mathbb{G}$: we have $|\Pr[S_{2.1} \wedge E_{2.1}] - \Pr[S_1 \wedge E_1]| \leq 2 \cdot \mathbf{Adv}_{\mathbb{G}}^{\mathrm{DLIN}}(\lambda)$. In Appendix D, Lemma 3 shows that, under the DLIN assumption in $\mathbb{G}$, the probability of $\mathcal{A}$'s forgery to be of the same type as the outputs of signing queries is about the same in Game 2.$k$ and in Game 2.$(k-1)$. We thus have $|\Pr[S_{2.k} \wedge E_{2.k}] - \Pr[S_{2.(k-1)} \wedge E_{2.(k-1)}]| \leq 2 \cdot \mathbf{Adv}_{\mathbb{G}}^{\mathrm{DLIN}}(\lambda)$.

When we reach Game 2.$L$, we know that $|\Pr[S_{2.L} \wedge E_{2.L}] - \Pr[S_{2.0} \wedge E_{2.0}]| \leq 2 \cdot L \cdot \mathbf{Adv}_{\mathbb{G}}^{\mathrm{DLIN}}(\lambda)$ by the triangle inequality. However, in Game 2.$L$, it is easy to prove that, even though $\mathcal{A}$ only obtains Type B-$k$ signatures throughout the game, its probability to output a Type B-$k$ forgery is negligible even with an unbounded computational power. Indeed, a legitimate adversary that outputs a forgery on a new message $M^\star$ has no information on $R_L(M^\star)$. Hence, it can only produce a Type B-$k$ forgery by pure chance and we thus have $\Pr[S_{2.L} \wedge E_{2.L}] \leq 1/p$. □

**Lemma 1.** *In Game 0, any PPT adversary outputting a Type B forgery with non-negligible probability implies an algorithm breaking the DLIN assumption in $\hat{\mathbb{G}}$ with nearly the same advantage.*

*Proof.* Let $\mathcal{A}$ be a PPT adversary that outputs a Type B forgery with probability $\epsilon$ in Game 0. We construct an algorithm $\mathcal{B}$ that takes as input an SDP instance $(\hat{g_z}, \hat{g_r}, \hat{h_z}, \hat{h_u}) \in \hat{\mathbb{G}}^4$ and finds a non-trivial $(Z, R, U) \in \mathbb{G}^3$ such that $e(Z, \hat{g_z}) \cdot e(R, \hat{g_r}) = 1_{\mathbb{G}_T}$ and $e(Z, \hat{h_z}) \cdot e(U, \hat{h_u}) = 1_{\mathbb{G}_T}$ with probability $\epsilon \cdot (1 - 1/p)$, which implies a DLIN distinguisher with the same advantage in $\hat{\mathbb{G}}$.

We actually use $\mathcal{A}$ to build a forger $\mathcal{B}$ against the one-time linearly homomorphic signature of Section 2.2 or, equivalently, an adversary defeating the soundness of the QA-NIZK proof system in [43]. Indeed, Type A signatures are exactly those for which the vector (3) is in the row space of $\mathbf{M}$ and $\mathcal{A}$ only obtains honestly generated Type A signatures at each query in Game 0. Hence, any adversary $\mathcal{A}$ creating a valid Type B signature in this game can be turned into a soundness adversary faking a QA-NIZK proof $(Z^\star, R^\star, U^\star)$ for a vector (3) that is linearly independent of the rows of $\mathbf{M}$.

In details, algorithm $\mathcal{B}$ receives as input a public key $\mathsf{pk}_{hsps}$ for an instance of the LHSPS scheme allowing to sign vectors of dimension $n = 4L + 3$. It runs Steps 1, 2 and 5 of the real key generation algorithm on its own to obtain $f, g, h, u_1, u_2$ and $(\Omega_1, \Omega_2) = (u_1^{\omega_1}, u_2^{\omega_2})$, for randomly chosen $\omega_1, \omega_2 \xleftarrow{R} \mathbb{Z}_p$. It then queries its own LHSPS challenger to obtain signatures $\{(Z_i, R_i, U_i)\}_{i=1}^{4L+2}$ on the rows of the matrix (1). The adversary $\mathcal{A}$ is run on input of

$$PK = \Big( f,\ g,\ h,\ u_1,\ u_2,\ \Omega_1 = u_1^{\omega_1},\ \Omega_2 = u_2^{\omega_2},\ \boldsymbol{V},\ \boldsymbol{W},$$
$$\mathsf{pk}_{hsps} = \big(\hat{g_z},\ \hat{g_r},\ \hat{h_z},\ \hat{h_u},\ \{(\hat{g}_i, \hat{h}_i)\}_{i=1}^{4L+3}\big),\ \{(Z_j, R_j, U_j)\}_{j=1}^{4L+2}\Big) \ .$$

Since $\mathcal{B}$ knows $\omega_1, \omega_2 \in_R \mathbb{Z}_p$, it can answer all signing queries by faithfully running the real signing algorithm, which does not require $\{(\chi_i, \gamma_i, \delta_i)\}_{i=1}^{4L+3}$. The game ends with the adversary outputting Type B signature $\sigma^\star = (\sigma_1^\star, \sigma_2^\star, \sigma_3^\star, Z^\star, R^\star, U^\star)$ on a message $M^\star$. This implies that $(Z^\star, R^\star, U^\star)$ is a valid homomorphic signature on the vector

$$(\sigma_1^\star, \sigma_2^{\star 1-M^\star[1]}, \sigma_2^{\star M^\star[1]}, \ldots, \sigma_2^{\star 1-M^\star[L]}, \sigma_2^{\star M^\star[L]}, \sigma_3^{\star 1-M^\star[1]}, \sigma_3^{\star M^\star[1]}, \ldots, \sigma_3^{\star 1-M^\star[L]}, \sigma_3^{\star M^\star[L]}, \Omega_1, \Omega_2) \ .$$

Moreover, the latter is outside the row space of $\mathbf{M}$ since $(\sigma^\star, M^\star)$ is a Type B forgery. Consequently, $\mathcal{B}$ can output $(Z^\star, R^\star, U^\star)$ and the above vector as a valid forgery against the LHSPS scheme. The result of [42] implies that $\mathcal{B}$ can in turn be used to solve the SDP problem —and thus the DLIN assumption in $\hat{\mathbb{G}}$— with probability $\epsilon \cdot (1 - 1/p) \geq \epsilon - 1/p$. □

**Lemma 2.** *If the DLIN assumption holds in $\mathbb{G}$, $\mathcal{A}$'s probability to output a Type B-1 signature in Game 2.1 is about the same as its probability to output a Type A signature in Game 1.*

*Proof.* Let us assume that events $S_{2.1} \wedge E_{2.1}$ and $S_1 \wedge E_1$ occur with noticeably different probabilities in Game 2.1 and Game 1, respectively. We construct a DLIN distinguisher $\mathcal{B}$ in $\mathbb{G}$. Our algorithm $\mathcal{B}$ takes as input $(f, g, h, f^a, h^b, T)$ with the task of deciding if $T = g^{a+b}$ or $T \in_R \mathbb{G}$. Similarly to [25, Lemma 6], $\mathcal{B}$ uses the random self-reducibility of DLIN to build $q$ tuples $(F_j = f^{a_j}, H_j = h^{b_j}, T_j)$ such that, for each $j \in \{1, \ldots, q\}$, we have

$$T_j = \begin{cases} g^{a_j+b_j} & \text{if } T = g^{a+b} \\ g^{a_j+b_j+\tau_0} & \text{if } T \in_R \mathbb{G} \end{cases}$$

for some $\tau_0 \in_R \mathbb{Z}_p$. This is done by picking $\rho_0 \xleftarrow{R} \mathbb{Z}_p$ and $\rho_{a_j}, \rho_{b_j} \xleftarrow{R} \mathbb{Z}_p$, for $j \in \{1, \ldots, q\}$, and setting

$$(F_j, H_j, T_j) = \left((f^a)^{\rho_0} \cdot f^{\rho_{a_j}}, (h^b)^{\rho_0} \cdot h^{\rho_{b_j}}, T^{\rho_0} \cdot g^{\rho_{a_j}+\rho_{b_j}}\right), \qquad \forall j \in \{1, \ldots, q\} \ .$$

In addition, $\mathcal{B}$ generates an extra tuple $(u_1, u_2, \Omega_1, \Omega_2) \in \mathbb{G}^4$ by choosing $\alpha_{u,1}, \alpha_{u,2} \xleftarrow{R} \mathbb{Z}_p$ and setting

$$u_1 = f^{\alpha_{u,1}}, \qquad u_2 = h^{\alpha_{u,2}}, \qquad \Omega_1 = (f^a)^{\alpha_{u,1}}, \qquad \Omega_2 = (h^b)^{\alpha_{u,2}} \ .$$

Before generating the public key of the scheme, $\mathcal{B}$ flips a coin $b^\dagger \xleftarrow{R} \{0,1\}$ hoping that the first bit of the target message $M^\star = M[1]^\star \ldots M[L]^\star \in \{0,1\}^L$ will coincide with $b^\dagger$. To construct the public key $PK$, $\mathcal{B}$ chooses $\boldsymbol{\alpha} = (\alpha_{1,0}, \alpha_{1,1}, \ldots, \alpha_{L,0}, \alpha_{L,1}) \xleftarrow{R} \mathbb{Z}_p^{2L}$, $\boldsymbol{\beta} = (\beta_{1,0}, \beta_{1,1}, \ldots, \beta_{L,0}, \beta_{L,1}) \xleftarrow{R} \mathbb{Z}_p^{2L}$ and $\zeta \xleftarrow{R} \mathbb{Z}_p$. It defines the vectors $\boldsymbol{V} = (V_{1,0}, V_{1,1}, \ldots, V_{L,0}, V_{L,1})$, $\boldsymbol{W} = (W_{1,0}, W_{1,1}, \ldots, W_{L,0}, W_{L,1})$ as

$$\begin{aligned} (V_{\ell,0}, V_{\ell,1}) &= (f^{\alpha_{\ell,0}}, f^{\alpha_{\ell,1}}), & (W_{\ell,0}, W_{\ell,1}) &= (h^{\beta_{\ell,0}}, h^{\beta_{\ell,1}}), & &\text{if } \ell \neq 1 \\ (V_{1,1-b^\dagger}, V_{1,b^\dagger}) &= (f^{\alpha_{1,1-b^\dagger}} \cdot g^{\zeta}, f^{\alpha_{1,b^\dagger}}), & (W_{1,1-b^\dagger}, W_{1,b^\dagger}) &= (h^{\beta_{1,1-b^\dagger}} \cdot g^{\zeta}, h^{\beta_{1,b^\dagger}}) \ . \end{aligned}$$

The rest of $PK$, including $(\mathsf{sk}_{hsps}, \mathsf{pk}_{hsps})$ and $\{(Z_i, R_i, U_i)\}_{i=1}^{4L+2}$, is generated as in the real setup. The adversary $\mathcal{A}$ is run on input of

$$PK = \Big( f,\ g,\ h,\ u_1,\ u_2,\ \Omega_1 = {u_1}^a,\ \Omega_2 = {u_2}^b,\ \boldsymbol{V},\ \boldsymbol{W},$$
$$\mathsf{pk}_{hsps} = \big(\hat{g_z},\ \hat{g_r},\ \hat{h_z},\ \hat{h_u},\ \{(\hat{g}_i, \hat{h}_i)\}_{i=1}^{4L+3}\big),\ \{(\hat{Z}_j, \hat{R}_j, \hat{U}_j)\}_{j=1}^{4L+2}\Big)$$

and the challenger $\mathcal{B}$ keeps $(\{\chi_i, \gamma_i, \delta_i\}_{i=1}^{4L+3})$ to itself. Note that $a, b \in \mathbb{Z}_p$ are part of the original DLIN instance and are not available to $\mathcal{B}$. However, $\mathcal{B}$ will use the challenge value $T$ – which is either $g^{a+b}$ or a random element of $\mathbb{G}$ – to answer signing queries.

Throughout the game, signing queries are answered as follows. In order to handle the $j$-th signing query $M^j = M[1]^j \ldots M[L]^j \in \{0,1\}^L$, the answer of $\mathcal{B}$ depends on the first bit $M[1]^j$ of $M^j$. Specifically, $\mathcal{B}$ considers the following cases.

– If $M[1]^j = b^\dagger$, $\mathcal{B}$ chooses $r, s \xleftarrow{R} \mathbb{Z}_p$ and sets

$$\sigma_1 = T \cdot H(\boldsymbol{V}, M)^r \cdot H(\boldsymbol{W}, M)^s, \qquad \sigma_2 = f^r, \qquad \sigma_3 = h^s,$$

where $H(\boldsymbol{V}, M) = \prod_{\ell=1}^{L} V_{\ell,M[\ell]}$ and $H(\boldsymbol{W}, M) = \prod_{\ell=1}^{L} W_{\ell,M[\ell]}$. The $(Z, R, U)$ components of the private key are computed by generating a homomorphic structure-preserving signature on the vector

$$(\sigma_1, \sigma_2^{1-M[1]}, \sigma_2^{M[1]}, \ldots, \sigma_2^{1-M[L]}, \sigma_2^{M[L]}, \sigma_3^{1-M[1]}, \sigma_3^{M[1]}, \ldots, \sigma_3^{1-M[L]}, \sigma_3^{M[L]}, \Omega_1, \Omega_2)\,,$$

by computing

$$\begin{cases} Z = \sigma_1^{-\chi_1} \cdot \sigma_2^{-\sum_{i=1}^{L} \chi_{2i+M[i]}} \cdot \sigma_3^{-\sum_{i=1}^{L} \chi_{2L+2i+M[i]}} \cdot \Omega_1^{-\chi_{4L+2}} \cdot \Omega_2^{-\chi_{4L+3}} \\ R = \sigma_1^{-\gamma_1} \cdot \sigma_2^{-\sum_{i=1}^{L} \gamma_{2i+M[i]}} \cdot \sigma_3^{-\sum_{i=1}^{L} \gamma_{2L+2i+M[i]}} \cdot \Omega_1^{-\gamma_{4L+2}} \cdot \Omega_2^{-\gamma_{4L+3}} \\ U = \sigma_1^{-\delta_1} \cdot \sigma_2^{-\sum_{i=1}^{L} \delta_{2i+M[i]}} \cdot \sigma_3^{-\sum_{i=1}^{L} \delta_{2L+2i+M[i]}} \cdot \Omega_1^{-\delta_{4L+2}} \cdot \Omega_2^{-\delta_{4L+3}} \end{cases} . \tag{5}$$

Note that, if $T = g^{a+b+\tau}$ for some $\tau \in_R \mathbb{Z}_p$, the obtained $(Z, R, U)$ can be written

$$\begin{cases} Z = g^{-\chi_1 \cdot \tau} \cdot Z_{4L+1}^{a} \cdot Z_{4L+2}^{b} \cdot \prod_{i=1}^{L} \left(Z_{2i-\overline{M[i]}}^{r} \cdot Z_{2L+2i-\overline{M[i]}}^{s}\right) \\ R = g^{-\gamma_1 \cdot \tau} \cdot R_{4L+1}^{a} \cdot R_{4L+2}^{b} \cdot \prod_{i=1}^{L} \left(R_{2i-\overline{M[i]}}^{r} \cdot R_{2L+2i-\overline{M[i]}}^{s}\right) \\ U = g^{-\delta_1 \cdot \tau} \cdot U_{4L+1}^{a} \cdot U_{4L+2}^{b} \cdot \prod_{i=1}^{L} \left(U_{2i-\overline{M[i]}}^{r} \cdot U_{2L+2i-\overline{M[i]}}^{s}\right) \end{cases} .$$

We observe that $(\sigma_1, \sigma_2, \sigma_3, Z, R, U)$ matches the distribution of signatures in both Game 2.1 if $\tau \neq 0$ and Game 1 if $\tau = 0$. Indeed, in the former case, we implicitly define the constant function $R_0(\varepsilon) = g^{\tau}$ and define the function $R_1$ so that $R_1(b^{\dagger}) = R_0(\varepsilon)$.

– If $M[1]^j = 1 - b^{\dagger}$, $\mathcal{B}$ implicitly defines

$$R_1(M_{|_1}^{j}) = R_1(1 - b^{\dagger}) = \begin{cases} R_0(\varepsilon) \cdot g^{\zeta \cdot \tau_0} & \text{if } T \in_R \mathbb{G} \\ 1 & \text{if } T = g^{a+b} \end{cases} .$$

Namely, $\mathcal{B}$ uses the $j$-th tuple $(F_j, H_j, T_j)$ to set

$$\sigma_1 = T \cdot F_j^{\sum_{\ell=1}^{L} \alpha_{\ell,M[\ell]}} \cdot H_j^{\sum_{\ell=1}^{L} \beta_{\ell,M[\ell]}} \cdot T_j^{\zeta}\,, \qquad \sigma_2 = F_j = f^{a_j}\,, \qquad \sigma_3 = H_j = h^{b_j}\ .$$

If $T = g^{a+b}$ (and thus $T_j = g^{a_j+b_j}$), this implicitly defines $\sigma_1 = g^{a+b} \cdot H(\boldsymbol{V}, M^j)^{a_j} \cdot H(\boldsymbol{W}, M^j)^{b_j}$, so that $(\sigma_1, \sigma_2, \sigma_3)$ has the same distribution as in Game 1. If $T = g^{a+b+\tau}$ (so that $T_j = g^{a_j+b_j+\tau_0}$), we can write

$$\sigma_1 = g^{a+b} \cdot R_1(M_{|_1}^{j}) \cdot H(\boldsymbol{V}, M^j)^{a_j} \cdot H(\boldsymbol{W}, M^j)^{b_j}\,,$$

since $R_1(M_{|_1}^{j}) = R_0(\varepsilon) \cdot g^{\zeta \cdot \tau_0}$, which is distributed as in Game 2.1. In either case, $(Z, R, U)$ are computed using $\mathsf{sk}_{hsps} = \{(\chi_i, \gamma_i, \delta_i)\}_{i=1}^{4L+3}$ as in the previous case (i.e., as per (5)).

In the forgery stage, $\mathcal{A}$ outputs a new message $M^{\star}$ and a signature $\sigma^{\star} = (\sigma_1^{\star}, \sigma_2^{\star}, \sigma_3^{\star}, Z^{\star}, R^{\star}, U^{\star})$. Our distinguisher $\mathcal{B}$ must determine if this forgery has the same type as the outputs of the simulated signing oracle. At this point, $\mathcal{B}$ halts and outputs a random bit if it turns out that $M[1]^{\star} \neq b^{\dagger}$. Otherwise, $\mathcal{B}$ can compute $F(\boldsymbol{v}, M^{\star}) = \sum_{\ell=1}^{L} \alpha_{\ell,M[\ell]^{\star}}$ and $F(\boldsymbol{w}, M^{\star}) = \sum_{\ell=1}^{L} \beta_{\ell,M[\ell]^{\star}}$, which yields

$$\eta^{\star} = \sigma_1^{\star} \cdot \sigma_2^{\star -F(\boldsymbol{v},M^{\star})} \cdot \sigma_3^{\star -F(\boldsymbol{w},M^{\star})}\ .$$

If $\eta^{\star} = T$, $\mathcal{B}$ considers $(\sigma^{\star}, M^{\star})$ as a forgery of the same type as outputs of the signing oracle and returns 1. Recall that $R_0(\varepsilon) = T/g^{a+b}$, so that $\sigma^{\star}$ matches the output distribution of the signing oracle in both Game 1 and Game 2.1. Otherwise, $\mathcal{B}$ decides that $\sigma^{\star}$ has a different distribution than signatures produced by the signing oracle and outputs 0. If the difference between $\mathcal{A}$'s probability to output the same kind of signatures as the signing oracle in games 2.1 and 2.1 is $\epsilon$, then $\mathcal{B}$'s advantage as a DLIN distinguisher is at least $\epsilon/2$ since $\mathcal{B}$'s choice for $b^{\dagger} \in \{0, 1\}$ is independent of $\mathcal{A}$'s view. □

We remark that, while its signatures are randomizable, the system can be made strongly unforgeable in a simple manner and without increasing the signature length. In particular, we do not need a chameleon-hash-function-based transformation such as [16]. Using the QA-NIZK proofs of [37], we thus obtain strongly unforgeable signatures based on the SXDH assumption which are short as those of Boneh, Shen and Waters [16] with a nearly tight reduction. The details are given in Appendix F.

We believe that, analogously to Waters signatures [53], the above scheme can serve as a basis for signature schemes with enhanced properties. For example, it can conceivably lead to threshold signatures [13] that are simultaneously short and non-interactive while providing tighter reductions in the standard model.[7] In particular, unlike the prime-order-group constructions of [25], our key generation algorithm does not involve any non-linear operation "in the exponent" and seems amenable to an efficient distributed key generation phase.

## 4 Almost Tightly CCA-Secure Encryption with Shorter Ciphertexts

Equipped with our signature scheme, we now present a public-key encryption scheme whose IND-CCA2 security in the multi-challenge-multi-user setting is almost tightly related to the DLIN assumption. Like [32], our scheme instantiates a variant of the Naor-Yung paradigm using Groth-Sahai proofs (which are recalled in Appendix C) and the cryptosystem of Boneh, Boyen and Shacham (BBS) [14].

The construction can be seen as an instantiation of a technique suggested by Dodis *et al.* [27] as a modification of the Naor-Yung paradigm, where only one IND-CPA secure encryption suffices (instead of two in [46,48]) if it is accompanied with a NIZK proof of knowledge of the plaintext that is simulation-extractable (and not only simulation-sound). In [5], Abe *et al.* used a simulation-extractable proof system showing that either: (i) The IND-CPA encryption scheme encrypts the message containted in an extractable commitment; (ii) Another commitment included in the proof is a valid signature on the verification key $\mathsf{VK}$ of a one-time signature. Here, we show that, if this simulation-extractable proof system is combined with the BBS cryptosystem, it can be simplified by removing the commitment to the message and the proof that this commitment contains the encrypted plaintext. The reason is that, in each simulation-extractable proof, the commitments to the encryption exponents suffice to guarantee the extractability of the plaintext.

While our reduction is not quite as tight as in the results of [32,5] since we lose a factor of $\Theta(\lambda)$, our scheme is much more space-efficient as the ciphertext overhead reduces to 68 group elements. In comparison, the most efficient solution of [5] incurs 398 group elements per ciphertext.

For simplicity, the description below uses symmetric pairings $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$ (i.e., $\mathbb{G} = \hat{\mathbb{G}}$) but extensions to asymmetric pairings are possible.

**Par-Gen($\lambda$):** Choose bilinear groups $(\mathbb{G}, \mathbb{G}_T)$ with generators $g, f, h \xleftarrow{R} \mathbb{G}$. Define common public parameters $\mathsf{par} = ((\mathbb{G}, \mathbb{G}_T), g, f, h)$.

**Keygen(par):** Parse $\mathsf{par}$ as $\big((\mathbb{G}, \mathbb{G}_T), g, f, h\big)$ and conduct the following steps.

1. Choose random exponents $x_1, y_1 \xleftarrow{R} \mathbb{Z}_p$ and set $f_1 = g^{x_1}$, $h_1 = g^{y_1}$.
2. Choose a strongly unforgeable one-time signature $\Sigma = (\mathcal{G}, \mathcal{S}, \mathcal{V})$ with verification keys of length $L \in \mathsf{poly}(\lambda)$.
3. For $\ell = 1$ to $L$, choose $V_{\ell,0}, V_{\ell,1}, W_{\ell,0}, W_{\ell,1} \xleftarrow{R} \mathbb{G}$ to assemble row vectors

$$\boldsymbol{V} = (V_{1,0}, V_{1,1}, \ldots, V_{L,0}, V_{L,1}) \in \mathbb{G}^{2L}, \qquad \boldsymbol{W} = (W_{1,0}, W_{1,1}, \ldots, W_{L,0}, W_{L,1}) \in \mathbb{G}^{2L} \ .$$

4. Choose $\omega_1, \omega_2 \xleftarrow{R} \mathbb{Z}_p$, $u_1, u_2 \xleftarrow{R} \mathbb{G}$, and compute $\Omega_1 = u_1^{\omega_1} \in \mathbb{G}$, $\Omega_2 = u_2^{\omega_2} \in \mathbb{G}$.

[7] Note that, even in the random oracle model, non-interactive threshold signatures like [51,13] all have a loose reduction since the techniques of [23,39] cannot be applied without interaction.

5. Define the matrix $\mathbf{M} = \big(M_{i,j}\big)_{i,j} \in \mathbb{G}^{(4L+2)\times(4L+3)}$ as

$$\big(M_{i,j}\big)_{i,j} = \left(\begin{array}{c|c|c|c|c} \boldsymbol{V}^\top & \mathbf{Id}_{f,2L} & \mathbf{1}^{2L\times 2L} & \mathbf{1}^{2L\times 1} & \mathbf{1}^{2L\times 1} \\ \hline \boldsymbol{W}^\top & \mathbf{1}^{2L\times 2L} & \mathbf{Id}_{h,2L} & \mathbf{1}^{2L\times 1} & \mathbf{1}^{2L\times 1} \\ \hline g & \mathbf{1}^{1\times 2L} & \mathbf{1}^{1\times 2L} & u_1 & 1 \\ \hline g & \mathbf{1}^{1\times 2L} & \mathbf{1}^{1\times 2L} & 1 & u_2 \end{array}\right)$$

with $\mathbf{Id}_{f,2L} = f^{\mathbf{I}_{2L}} \in \mathbb{G}^{2L\times 2L}$, $\mathbf{Id}_{h,2L} = h^{\mathbf{I}_{2L}} \in \mathbb{G}^{2L\times 2L}$, where $\mathbf{I}_{2L} \in \mathbb{Z}_p^{2L\times 2L}$ is the identity matrix. Then, generate a key pair for the linearly homomorphic one-time signature of Section 2.2 with $n = 4L+3$. Let $\mathsf{pk}_{hsps} = \big(g_z,\ g_r,\ h_z,\ h_u,\ \{g_i, h_i\}_{i=1}^{4L+3}\big)$ be the public key and let $\mathsf{sk}_{hsps} = \{\chi_i, \gamma_i, \delta_i\}_{i=1}^{4L+3}$ be the underlying private key.

6. Generate one-time linearly homomorphic signatures $\{(z_j, r_j, u_j)\}_{j=1}^{4L+2}$ on the rows of $\mathbf{M}$.
7. Choose a perfectly witness indistinguishable Groth-Sahai CRS $\mathbf{g} = (\boldsymbol{G}_1, \boldsymbol{G}_2, \boldsymbol{G}_3)$ defined by vectors $\boldsymbol{G}_1 = (G_1, 1, G)$, $\boldsymbol{G}_2 = (1, G_2, G)$ and $\boldsymbol{G}_3 \in \mathbb{G}^3$, with $G, G_1, G_2 \xleftarrow{R} \mathbb{G}$ and $\boldsymbol{G}_3 \xleftarrow{R} \mathbb{G}^3$.
8. Define the private key as $SK = (x_1, y_1) \in \mathbb{Z}_p^2$. The public key is defined to be

$$PK = \big(g, f_1,\ h_1,\ \boldsymbol{V},\ \boldsymbol{W},\ u_1,\ u_2,\ \Omega_1,\ \Omega_2,\ \mathsf{pk}_{hsps},\ \{(z_j, r_j, u_j)\}_{j=1}^{4L+2}, \mathbf{g} = (\boldsymbol{G}_1, \boldsymbol{G}_2, \boldsymbol{G}_3),\ \Sigma\big),$$

whereas $\omega_1, \omega_2 \in \mathbb{Z}_p$ and $\mathsf{sk}_{hsps}$ are erased.

**Encrypt($\boldsymbol{M}, \boldsymbol{PK}$):** To encrypt $M \in \mathbb{G}$, generate a one-time signature key pair $(\mathsf{SK}, \mathsf{VK}) \leftarrow \mathcal{G}(\lambda)$ and conduct the following steps:

1. Choose $\theta_1, \theta_2 \xleftarrow{R} \mathbb{Z}_p$ and compute $(C_0, C_1, C_2) = (M \cdot g^{\theta_1+\theta_2}, f_1^{\theta_1}, h_1^{\theta_2})$.
2. Choose $r, s \xleftarrow{R} \mathbb{Z}_p$ and compute a pseudo-signature

$$\sigma_1 = H(\boldsymbol{V}, \mathsf{VK})^r \cdot H(\boldsymbol{W}, \mathsf{VK})^s\,, \qquad \sigma_2 = f^r\,, \qquad \sigma_3 = h^s\,,$$

where $H(\boldsymbol{V}, \mathsf{VK}) = \prod_{\ell=1}^{L} V_{\ell,\mathsf{VK}[\ell]}$ and $H(\boldsymbol{W}, \mathsf{VK}) = \prod_{\ell=1}^{L} W_{\ell,\mathsf{VK}[\ell]}$.

3. Define the variables $(W_1, W_2) = (g^{\theta_1}, g^{\theta_2})$ and compute Groth-Sahai commitments $\{\boldsymbol{C}_{W_i}\}_{i=1}^{2}$ to these.
4. Define the bit $b = 1$ and generate a commitment $\boldsymbol{C}_b = (1, 1, G^b) \cdot \boldsymbol{G}_1^{r_b} \cdot \boldsymbol{G}_2^{s_b} \cdot \boldsymbol{G}_3^{t_b}$, where $r_b, s_b, t_b \xleftarrow{R} \mathbb{Z}_p$ to it. Then, compute a Groth-Sahai commitment $\boldsymbol{C}_{\sigma_1}$ to $\sigma_1$ and commitments $\boldsymbol{C}_{\Theta_1}, \boldsymbol{C}_{\Theta_2} \in \mathbb{G}^3$ and $\boldsymbol{C}_{\Gamma_g}$ to the variables

$$\Theta_1 = \Omega_1^{1-b}, \qquad \Theta_2 = \Omega_2^{1-b}, \qquad \Gamma_g = g^b. \tag{6}$$

The vector

$$\begin{aligned}(\sigma_1, \sigma_2^{1-\mathsf{VK}[1]}, \sigma_2^{\mathsf{VK}[1]}, \ldots, \sigma_2^{1-\mathsf{VK}[L]}, \sigma_2^{\mathsf{VK}[L]},& \\ \sigma_3^{1-\mathsf{VK}[1]}, \sigma_3^{\mathsf{VK}[1]}, \ldots, \sigma_3^{1-\mathsf{VK}[L]}, \sigma_3^{\mathsf{VK}[L]}, \Theta_1, \Theta_2) \in \mathbb{G}^{4L+3}&\end{aligned} \tag{7}$$

belongs to the subspace spanned by the first $4L$ rows of the matrix $\mathbf{M} \in \mathbb{G}^{(4L+2)\times(4L+3)}$. Hence, the algorithm can use $r, s \in \mathbb{Z}_p$ to derive a one-time linearly homomorphic signature $(Z, R, U) \in \mathbb{G}^3$ on the vector (7). Note that $(\sigma_1, \sigma_2, \sigma_3, Z, R, U)$ can be seen as a signature on $\mathsf{VK}$, for the degenerated private key $(\omega_1, \omega_2) = (0, 0)$.

5. Generate commitments $\boldsymbol{C}_Z, \boldsymbol{C}_R, \boldsymbol{C}_U \in \mathbb{G}^3$. Then, compute a NIWI proof $\boldsymbol{\pi}_b \in \mathbb{G}^9$ that $b$ satisfies $b^2 = b$ (which ensures that $b \in \{0, 1\}$) and NIWI proofs $\boldsymbol{\pi}_{\mathrm{PPE1}}, \boldsymbol{\pi}_{\mathrm{PPE2}} \in \mathbb{G}^3$ that

committed variables $(\sigma_1, Z, R, U, \Theta_1, \Theta_2)$ satisfy the pairing product equations

$$e(Z, g_z) \cdot e(R, g_r) = e(\sigma_1, g_1)^{-1} \cdot e(\sigma_2, \prod_{i=1}^{L} g_{2i+\mathsf{VK}[i]})^{-1} \cdot e(\sigma_3, \prod_{i=1}^{L} g_{2L+2i+\mathsf{VK}[i]})^{-1} \cdot e(\Theta_1, g_{4L+2})^{-1} \cdot e(\Theta_2, g_{4L+3})^{-1},$$

$$e(Z, h_z) \cdot e(U, h_u) = e(\sigma_1, h_1)^{-1} \cdot e(\sigma_2, \prod_{i=1}^{L} h_{2i+\mathsf{VK}[i]})^{-1} \cdot e(\sigma_3, \prod_{i=1}^{L} h_{2L+2i+\mathsf{VK}[i]})^{-1} \cdot e(\Theta_1, h_{4L+2})^{-1} \cdot e(\Theta_2, h_{4L+3})^{-1} .$$

6. Generate NIWI proofs $\boldsymbol{\pi}_g$, $\{\boldsymbol{\pi}_{\Theta_i}\}_{i=1}^2$ that elements $(b, \Gamma_g, \Theta_1, \Theta_2)$, which are committed in $\boldsymbol{C}_b, \boldsymbol{C}_{\Gamma_g}, \boldsymbol{C}_{\Theta_1}, \boldsymbol{C}_{\Theta_2}$, satisfy (6). Each such proof requires 3 elements of $\mathbb{G}$.
7. Generate a simulation-extractable proof that $(W_1, W_2)$ satisfy

$$e(C_1, g) = e(f_1, W_1), \qquad e(C_2, g) = e(h_1, W_2) . \tag{8}$$

   To this end, prove that $(W_1, W_2, \Gamma_g)$ satisfy

$$e(C_1, \Gamma_g) = e(f_1, W_1), \qquad e(C_2, \Gamma_g) = e(h_1, W_2) . \tag{9}$$

   This requires proofs $\boldsymbol{\pi}_1, \boldsymbol{\pi}_2$ for linear pairing product equations, which cost 3 group elements each.
8. Finally, compute a one-time signature $sig = \mathcal{S}(\mathsf{SK}, C_0, C_1, C_2, \pi)$ and output the ciphertext $C = (\mathsf{VK}, C_0, C_1, C_2, \pi, sig)$, where

$$\pi = (\boldsymbol{C}_b, \boldsymbol{\pi}_b, \boldsymbol{C}_{\sigma_1}, \sigma_2, \sigma_3, \{\boldsymbol{C}_{W_i}\}_{i=1}^2, \boldsymbol{C}_Z, \boldsymbol{C}_R, \boldsymbol{C}_U, \{\boldsymbol{C}_{\Theta_i}\}_{i=1}^2, \boldsymbol{C}_{\Gamma_g}, \boldsymbol{\pi}_g, \{\boldsymbol{\pi}_{\Theta_i}\}_{i=1}^2, \boldsymbol{\pi}_{\mathrm{PPE1}}, \boldsymbol{\pi}_{\mathrm{PPE2}}, \boldsymbol{\pi}_1, \boldsymbol{\pi}_2) \tag{10}$$

   is a simulation-extractable proof of plaintext knowledge consisting of 62 elements of $\mathbb{G}$.

**Decrypt($\boldsymbol{SK}, \boldsymbol{C}$):** Parse $C$ as $C = (\mathsf{VK}, C_0, C_1, C_2, \pi, sig)$ and do the following.

1. Return $\perp$ if $\mathcal{V}(\mathsf{VK}, (C_0, C_1, C_2, \pi), sig) = 0$ or if $\pi$ does not properly verify.
2. Using $SK = (x_1, y_1) \in \mathbb{Z}_p^2$, compute and return $M = C_0 \cdot C_1^{-1/x_1} \cdot C_2^{-1/y_1}$.

Note that $\pi$ forms a proof that either $(\sigma_1, \sigma_2, \sigma_3)$ is a valid signature or $\{\boldsymbol{C}_{W_i}\}_{i=1}^2$ are commitments to $(W_1, W_2) = (g^{\theta_1}, g^{\theta_2})$, where $\theta_1, \theta_2 \in \mathbb{Z}_p$ are the encryption exponents. A simulator holding the private key $(\omega_1, \omega_2) \in \mathbb{Z}_p^2$ of the signature scheme can thus simulate a proof $\pi$ of plaintext knowledge by computing $(\sigma_1, \sigma_2, \sigma_3)$ as a real signature, by setting $b = 0$ at step 4 of the encryption algorithm and using the witnesses $(W_1, W_2) = (1_{\mathbb{G}}, 1_{\mathbb{G}})$ to prove relations (9).

From an efficiency standpoint, we remark that each ciphertext must contain a proof comprised of 62 group elements. In an instantiation using the one-time signature of Hofheinz and Jager [32], the entire ciphertexts thus costs 69 group elements. The scheme can also be adapted to asymmetric pairings in a simple manner.

For the sake of simplicity, we follow [5] and only prove security in the single-user, multi-challenge case. However, as pointed out in [5], the single-user security results can always be simply extended to the scenario of multiple public keys by leveraging the random self-reducibility of the DLIN assumption in a standard manner.

**Theorem 2.** *The scheme is $(1, q_e)$-IND-CCA secure provided: (i) $\Sigma$ is a strongly unforgeable one-time signature; and (ii) the DLIN assumption holds in $\mathbb{G}$. For any adversary $\mathcal{A}$, there exist a one-time signature forger $\mathcal{B}'$ and a DLIN distinguisher $\mathcal{B}$ with running times $t_{\mathcal{B}}, t_{\mathcal{B}'} \leq t_{\mathcal{A}} + q_e \cdot \mathsf{poly}(\lambda, L)$ such that*

$$\mathbf{Adv}_{\mathcal{A}}^{(1,q_e)\text{-cca}}(\lambda) \leq 2 \cdot \mathbf{Adv}_{\mathcal{B}'}^{n\text{-suf-ots}}(\lambda) + (4 \cdot L + 5) \cdot \mathbf{Adv}_{\mathcal{B}}^{\text{DLIN}}(\lambda) + 5/p\,,$$

*where $L$ is the length of one-time verification keys and $q_e$ denotes the number of encryption queries.*

*Proof.* The proof is given in Appendix H.2. □

In order to extend the result to the multi-user setting, the main changes are that we need to rely on: *(i)* The random self-reducibility of DLIN, which is used as in [32]; *(ii)* The almost tight security of the signature scheme of Section 3 in the multi-user setting [28], which can also be proved using the random self-reducibility of DLIN. The latter proof notably relies on the tight security of the homomorphic signature of Section 2.2 in the multi-key setting, which is proved in Appendix G.

## 5 Conclusion

In this paper, we described a new efficient signature scheme with an almost tight security reduction under a standard assumption. This signature scheme allows constructing an efficient public-key encryption scheme with (almost) tight chosen-ciphertext security in the multi-challenge setting via an efficient simulation-extractable proof of knowledge of the plaintext. While our ciphertexts are relatively short, the underlying NIZK proofs still have linear size in the number of group elements contained in the inner IND-CPA encryption layer. Towards realizing truly practical tightly secure systems, it remains an interesting open problem to construct efficient simulation-sound quasi-adaptive NIZK arguments of linear subspace membership – such as those of [43] – which simultaneously provide tight security under a standard assumption and constant-size proofs.

## Acknowledgements

We thank the anonymous reviewers for very useful comments. In particular, we are grateful to one reviewer for suggesting a more efficient approach to the scheme of Section 4. The first author's work was supported in part by the ERC Starting Grant ERC-2013-StG-335086-LATTAC. This work has also been supported by the "Programme Avenir Lyon Saint-Etienne de l'Université de Lyon" in the framework of the programme "Inverstissements d'Avenir" (ANR-11-IDEX-0007).

## References

1. M. Abdalla, P.-A. Fouque, V. Lyubashevsky, M. Tibouchi. Tightly-secure signatures from lossy identification schemes. In *Eurocrypt '12*, LNCS 7237, pp. 572–590, Springer, 2012.
2. M. Abe, G. Fuchsbauer, J. Groth, K. Haralambiev, M. Ohkubo. Structure-preserving signatures and commitments to group elements. In *Crypto '10*, LNCS 6223, pp. 209–236, Springer, 2010.
3. M. Abe, K. Haralambiev, M. Ohkubo. Signing on elements in bilinear groups for modular protocol design. In Cryptology ePrint Archive: Report 2010/133, 2010.
4. M. Abe, M. Chase, B. David, M. Kohlweiss, R. Nishimaki, M. Ohkubo. Constant-size structure-preserving signatures: Generic constructions and simple assumptions. In *Asiacrypt '12*, LNCS 7658, pp. 4–24, Springer, 2012.
5. M. Abe, B. David, M. Kohlweiss, R. Nishimaki, M. Ohkubo. Tagged one-time signatures: Tight security and optimal tag size. In *PKC '13*, LNCS 7778, pp. 312–331, Springer, 2013.
6. M. Bellare, A. Boldyreva, S. Micali. Public-key encryption in a multi-user setting: Security proofs and improvements. In *Eurocrypt '00*, LNCS 1807, pp. 259–274, Springer, 2000.
7. M. Bellare, T. Ristenpart. Simulation without the Artificial Abort: Simplified Proof and Improved Concrete Security for Waters' IBE Scheme. In *Eurocrypt '09*, LNCS 5479, pp. 407–424, 2009.

8. M. Bellare, P. Rogaway. Random oracles are practical: A paradigm for designing efficient protocols. In *ACM CCS '93*, pp. 62–73, ACM Press, 1993.
9. M. Bellare, P. Rogaway. The exact security of digital signatures - How to sign with RSA and Rabin. In *Eurocrypt '96*, LNCS 1070, pp. 399–416, Springer, 1996.
10. D. Bernstein. Proving tight security for Rabin-Williams signatures. In *Eurocrypt '08*, LNCS 4965, pp. 70–87, Springer, 2008.
11. O. Blazy, E. Kiltz, J. Pan. (Hierarchical) Identity-Based Encryption from Affine Message Authentication. In *Crypto '14*, LNCS 8616, pp. 70–87, pp. 408–425, Springer, 2014.
12. F. Böhl, D. Hofheinz, J. Koch, T. Jager, J.-H. Seo, C. Striecks. Practical signatures from standard assumptions. In *Eurocrypt '13*, LNCS 7881, pp. 461–485, Springer, 2013.
13. A. Boldyreva. Threshold signatures, multisignatures and blind signatures based on the gap-Diffie-Hellman-group signature scheme. In *PKC '03*, LNCS 2567, pp. 31–46, Springer, 2003.
14. D. Boneh, X. Boyen, H. Shacham. Short group signatures. In *Crypto '04*, LNCS 3152, pp. 41–55, Springer, 2004.
15. D. Boneh, M. Franklin. Identity-based encryption from the Weil pairing. In *SIAM J. of Computing* 32(3), pp. 586–615, 2003. Earlier version in *Crypto '01*.
16. D. Boneh, E. Shen, B. Waters. Strongly unforgeable signatures based on computational Diffie-Hellman. In *PKC '06*, LNCS 3958, pp. 229–240, Springer, 2006.
17. J. Camenisch, N. Chandran, and V. Shoup. A public key encryption scheme secure against key dependent chosen plaintext and adaptive chosen ciphertext attacks. In *Eurocrypt '09*, LNCS 5479, pp. 351–368, Springer, 2009.
18. J. Cathalo, B. Libert, M. Yung. Group encryption: Non-interactive realization in the standard model. In *Asiacrypt '09*, LNCS 5912, pp. 179–196, Springer, 2009.
19. J. Chen, H.-W. Lim, S. Ling, H. Wang, H. Wee. Shorter IBE and signatures via asymmetric pairings. In *Pairing '12*, LNCS 7708, pp. 122–140, Springer, 2012.
20. B. Chevallier-Mames. An efficient CDH-based signature scheme with a tight security reduction. In *Crypto '05*, LNCS 3621, pp. 511–526, Springer, 2005.
21. B. Chevallier-Mames, M. Joye. A practical and tightly secure signature scheme without hash function. *CT-RSA '07*, LNCS 4377, pp. 339–356, Springer, 2007.
22. J.-S. Coron. On the exact security of full domain hash. In *Crypto '00*, LNCS 1880, pp. 229–235, Springer, 2000.
23. J.-S. Coron. Optimal security proofs for PSS and other signature schemes. In *Eurocrypt '02*, LNCS 2332, pp. 229–235, Springer, 2002.
24. J.-S. Coron. A variant of Boneh-Franklin IBE with a tight reduction in the random oracle model. *Designs, Codes & Cryptography* 50(1), pp. 115–133, 2009.
25. J. Chen, H. Wee. Fully, (almost) tightly secure IBE and dual system groups. In *Crypto '13*, LNCS 8043, pp. 435-460, Springer, 2013.
26. R. Cramer, V. Shoup. A practical public key cryptosystem provably secure against adaptive chosen ciphertext attack. In *Crypto '98*, LNCS 1462, pp. 13–25, Springer, 1998.
27. Y. Dodis, K. Haralambiev, A. López-Alt, D. Wichs. Efficient Public-Key Cryptography in the Presence of Key Leakage. In *Asiacrypt '10*, LNCS 6477, pp. 613–631, Springer, 2010.
28. S. Galbraith, J. Malone-Lee, N. Smart. Public-key signatures in the multi-user setting. In *Information Processing Letters*, vol. 83(5), pp. 263–266, 2002.
29. M. Gerbush, A. Lewko, A. O 'Neill, B. Waters. Dual form signatures: An approach for proving security from static assumptions. In *Asiacrypt '12*, LNCS 7658, pp. 25–42, Springer, 2012.
30. J. Groth. Simulation-sound NIZK proofs for a practical language and constant size group signatures. In *Asiacrypt '06*, LNCS 4284, pp. 444–459, Springer, 2006.
31. J. Groth, A. Sahai. Efficient non-interactive proof systems for bilinear groups. In *Eurocrypt́08*, LNCS 4965, pp. 415-432, Springer, 2008.
32. D. Hofheinz, T. Jager. Tightly secure signatures and public-key encryption. In *Crypto '12*, LNCS 7417, pp. 590–607, Springer, 2012.
33. D. Hofheinz, T. Jager, E. Kiltz. Shorter signatures from weaker assumptions. In *Asiacrypt '11*, LNCS 7073, pp. 647–666, Springer, 2011.
34. D. Hofheinz, T. Jager, E. Knapp. Waters signatures with optimal security reduction. In *PKC '12*, LNCS 7293, pp. 66–83, Springer, 2012.
35. S. Hohenberger, B. Waters. Short and stateless signatures from the RSA assumption. In *Crypto '09*, LNCS 5677, pp. 654–670, Springer, 2009.
36. C. Jutla, A. Roy. Shorter quasi-adaptive NIZK proofs for linear subspaces. In *Asiacrypt '13*, LNCS 8269, pp. 1–20, Springer, 2013.
37. C. Jutla, A. Roy. Switching lemma for bilinear tests and constant-size NIZK proofs for linear subspaces. In *Crypto '14*, LNCS 8617, pp. 295–312, Springer, 2014.
38. S. Kakvi, E. Kiltz. Optimal security proofs for full domain hash, revisited. In *Eurocrypt '12*, LNCS 7237, pp. 537–553, Springer, 2012.

39. J. Katz, N. Wang. Efficiency improvements for signature schemes with tight security reductions. In *ACM-CCS '03*, pp. 155–164 , ACM Press, 2003.
40. A. Lewko, B. Waters. New techniques for dual system encryption and fully secure HIBE with short ciphertexts. In *TCC '10*, LNCS 5978, pp. 455–479, Springer, 2010.
41. A. Lewko. Tools for simulating features of composite order bilinear groups in the prime order setting. In *Eurocrypt '12*, LNCS 7237, pp. 318–335, Springer, 2012.
42. B. Libert, T. Peters, M. Joye, M. Yung. Linearly homomorphic structure-preserving signatures and their applications. In *Crypto '13*, LNCS 8043, pp. 289–307, Springer, 2013.
43. B. Libert, T. Peters, M. Joye, M. Yung. Non-malleability from malleability: Simulation-sound quasi-adaptive NIZK proofs and CCA2-secure encryption from homomorphic signatures. In *Eurocrypt '14*, LNCS 8441, pp. 514–532, Springer, 2014.
44. M. Naor. On cryptographic assumptions and challenges. In *Crypto '03*, LNCS 2729, pp. 96–109, Springer, 2003.
45. M. Naor, O. Reingold. Number-theoretic constructions of efficient pseudo-random functions. In *FOCS '97*, pp. 458–467, IEEE Press, 1997.
46. M. Naor, M. Yung. Public-key cryptosystems provably secure against chosen ciphertext attacks. In *STOC '90*, ACM Press, 1990.
47. C. Rackoff, D. Simon. Non-interactive zero-knowledge proof of knowledge and chosen ciphertext attack. In *Crypto,́91*, LNCS 576, pp. 433–444, Springer, 1991.
48. A. Sahai. Non-malleable non-interactive zero knowledge and adaptive chosen-ciphertext security. In *FOCS '99*, pp. 543–553, IEEE Press, 1999.
49. S. Schäge. Tight proofs for signature schemes without random oracles. *Eurocrypt '11*, LNCS 6632, pp. 189–206, Springer, 2011.
50. A. Shamir. Identity-based cryptosystems and signature schemes. In *Crypto '84*, LNCS 196, pp. 47–53, Springer, 1984.
51. V. Shoup. Practical threshold signatures. In *Eurocrypt '00*, LNCS 1807, pp. 207–220, Springer, 2000.
52. V. Shoup. A proposal for an ISO standard for public key encryption. Manuscript, December 20, 2001.
53. B. Waters. Efficient identity-based encryption without random oracles. In *Eurocrypt '05*, LNCS 3494, Springer, 2005.
54. B. Waters. Dual system encryption: Realizing fully secure IBE and HIBE under simple assumptions. In *Crypto '09*, LNCS 5677, pp. 619–636, Springer, 2009.
55. S. Yamada, G. Hanaoka, N. Kunihiro. Two-dimensional representation of cover free families and its applications: Short signatures and more. In *CT-RSA '12*, LNCS 7178, pp. 260–277, Springer, 2012.
56. S. Yamada, G. Hanaoka, N. Kunihiro. Space efficient signature schemes from the RSA assumption. In *PKC '12*, LNCS 7293, pp. 102–119, Springer, 2012.

## A Quasi-Adaptive NIZK Proofs

Quasi-Adaptive NIZK (QA-NIZK) proofs [36] are NIZK proofs where the CRS is allowed to depend on the specific language for which proofs have to be generated. The CRS is divided into a fixed part $\Gamma$, produced by an algorithm $\mathsf{K}_0$, and a language-dependent part $\psi$. However, there should be a single simulator for the entire class of languages.

Let $\lambda$ be a security parameter. For public parameters $\Gamma$ produced by $\mathsf{K}_0$, let $\mathcal{D}_\Gamma$ be a probability distribution over a collection of relations $\mathcal{R} = \{R_\rho\}$ parametrized by a string $\rho$ with an associated language $\mathcal{L}_\rho = \{x \mid \exists w : R_\rho(x, w) = 1\}$.

A tuple of algorithms $(\mathsf{K}_0, \mathsf{K}_1, \mathsf{P}, \mathsf{V})$ is a QA-NIZK proof system for $\mathcal{R}$ if there exists a PPT simulator $(\mathsf{S}_1, \mathsf{S}_2)$ such that, for any PPT adversaries $\mathcal{A}_1, \mathcal{A}_2$ and $\mathcal{A}_3$, we have the following properties:

**Quasi-Adaptive Completeness:**

$$\Pr[\Gamma \leftarrow \mathsf{K}_0(\lambda);\ \rho \leftarrow \mathcal{D}_\Gamma;\ \psi \leftarrow \mathsf{K}_1(\Gamma, \rho);\ (x, w) \leftarrow \mathcal{A}_1(\Gamma, \psi, \rho);\ \pi \leftarrow \mathsf{P}(\psi, x, w) : \mathsf{V}(\psi, x, \pi) = 1 \ \text{ if } R_\rho(x, w) = 1] = 1 \ .$$

**Quasi-Adaptive Soundness:**

$$\Pr[\Gamma \leftarrow \mathsf{K}_0(\lambda);\ \rho \leftarrow \mathcal{D}_\Gamma;\ \psi \leftarrow \mathsf{K}_1(\Gamma, \rho);\ (x, \pi) \leftarrow \mathcal{A}_2(\Gamma, \psi, \rho) : \mathsf{V}(\psi, x, \pi) = 1 \ \wedge \ \neg(\exists w : R_\rho(x, w) = 1)] \in \mathsf{negl}(\lambda) \ .$$

**Quasi-Adaptive Zero-Knowledge:**

$$\Pr[\Gamma \leftarrow \mathsf{K}_0(\lambda);\ \rho \leftarrow \mathcal{D}_\Gamma;\ \psi \leftarrow \mathsf{K}_1(\Gamma,\rho)\ :\ \mathcal{A}_3^{\mathsf{P}(\psi,.,.)}(\Gamma,\psi,\rho)=1]$$
$$\approx \Pr[\Gamma \leftarrow \mathsf{K}_0(\lambda);\ \rho \leftarrow \mathcal{D}_\Gamma;\ (\psi,\tau_{sim}) \leftarrow \mathsf{S}_1(\Gamma,\rho)\ :\ \mathcal{A}_3^{\mathsf{S}(\psi,\tau_{sim},.,.)}(\Gamma,\psi,\rho)=1]\,,$$

where

- $\mathsf{P}(\psi,.,.)$ emulates the actual prover. It takes as input a pair $(x,w)$ and outputs a proof $\pi$ if $(x,w) \in R_\rho$. Otherwise, it outputs $\perp$.
- $\mathsf{S}(\psi,\tau_{sim},.,.)$ is an oracle that takes as input $(x,w)$. It outputs a simulated proof $\mathsf{S}_2(\psi,\tau_{sim},x)$ if $(x,w) \in R_\rho$ and $\perp$ if $(x,w) \notin R_\rho$.

We assume that the CRS $\psi$ contains an encoding of $\rho$, which is thus available to $\mathsf{V}$. The definition of Quasi-Adaptive Zero-Knowledge requires a single simulator for the entire family of relations $\mathcal{R}$.

## B Definitions for Linearly Homomorphic Structure-Preserving Signatures

Let $(\mathbb{G},\mathbb{G}_T)$ be groups of prime order $p$ such that a bilinear map $e : \mathbb{G}\times\mathbb{G} \to \mathbb{G}_T$ can be efficiently computed.

A signature scheme is *structure-preserving* [3,2] if messages, signatures and public keys all live in the group $\mathbb{G}$. In linearly homomorphic structure-preserving signatures, the message space $\mathcal{M}$ consists of pairs $\mathcal{M} := \mathcal{T}\times\mathbb{G}^n$, for some $n \in \mathbb{N}$, where $\mathcal{T}$ is a tag space. Depending on the application, one may want the tags to be group elements or not. In this paper, they can be arbitrary strings.

**Definition 3.** *A* linearly homomorphic structure-preserving signature scheme over $(\mathbb{G},\mathbb{G}_T)$ *is a tuple of efficient algorithms* $\Sigma = (\mathsf{Keygen},\mathsf{Sign},\mathsf{SignDerive},\mathsf{Verify})$ *for which the message space consists of* $\mathcal{M} := \mathcal{T}\times\mathbb{G}^n$*, for some integer* $n \in \mathsf{poly}(\lambda)$ *and some set* $\mathcal{T}$*, and with the following specifications.*

**$\mathsf{Keygen}(\lambda, n)$** *is a randomized algorithm that takes in a security parameter* $\lambda \in \mathbb{N}$ *and an integer* $n \in \mathsf{poly}(\lambda)$ *denoting the dimension of vectors to be signed. It outputs a key pair* $(\mathsf{pk},\mathsf{sk})$*, where* $\mathsf{pk}$ *includes the description of a tag space* $\mathcal{T}$*, where each tag serves as a file identifier.*

**$\mathsf{Sign}(\mathsf{sk},\tau,\boldsymbol{M})$** *is a possibly randomized algorithm that takes as input a private key* $\mathsf{sk}$*, a file identifier* $\tau \in \mathcal{T}$ *and a vector* $\boldsymbol{M} = (M_1,\ldots,M_n) \in \mathbb{G}^n$*. It outputs a signature* $\sigma \in \mathbb{G}^{n_s}$*, for some* $n_s \in \mathsf{poly}(\lambda)$*.*

**$\mathsf{SignDerive}(\mathsf{pk},\tau,\{(\omega_i,\sigma^{(i)})\}_{i=1}^{\ell})$** *is a (possibly randomized) derivation algorithm. It inputs a public key* $\mathsf{pk}$*, a file identifier* $\tau$ *as well as* $\ell$ *pairs* $(\omega_i,\sigma^{(i)})$*, each of which consists of a coefficient* $\omega_i \in \mathbb{Z}_p$ *and a signature* $\sigma^{(i)} \in \mathbb{G}^{n_s}$*. It outputs a signature* $\sigma \in \mathbb{G}^{n_s}$ *on the vector* $\boldsymbol{M} = \prod_{i=1}^{\ell} \boldsymbol{M}_i{}^{\omega_i}$*, where* $\sigma^{(i)}$ *is a signature on* $\boldsymbol{M}_i$*.*

**$\mathsf{Verify}(\mathsf{pk},\tau,\boldsymbol{M},\boldsymbol{\sigma})$** *is a deterministic verification algorithm that takes as input a public key* $\mathsf{pk}$*, a file identifier* $\tau \in \mathcal{T}$*, a signature* $\sigma$ *and a vector* $\boldsymbol{M} = (M_1,\ldots,M_n)$*. It outputs* 0 *or* 1 *depending on whether* $\sigma$ *is deemed valid or not.*

In a *one-time* linearly homomorphic SPS, the tag $\tau$ can be omitted in the specification as a given key pair $(\mathsf{pk},\mathsf{sk})$ only allows signing one linear subspace.

As in all linearly homomorphic signatures, the desired security notion mandates the adversary's inability to come up with a valid triple $(\tau^\star,\boldsymbol{M}^\star,\sigma^\star)$ for a new file identifier $\tau^\star$ or, if $\tau^\star$ appeared in signatures generated by the signing oracle, for a vector $\boldsymbol{M}^\star$ outside the linear span of the vectors that have been legitimately signed for the tag $\tau^\star$.

## C Groth-Sahai Proof Systems

In their instantiation based on the DLIN assumption in symmetric pairing configurations, the Groth-Sahai (GS) proof systems [31] use a common reference string (CRS) consisting of three vectors $\boldsymbol{g_1}, \boldsymbol{g_2}, \boldsymbol{g_3} \in \mathbb{G}^3$, where $\boldsymbol{g_1} = (g_1, 1, g)$, $\boldsymbol{g_2} = (1, g_2, g)$ for some $g_1, g_2 \in \mathbb{G}$. To commit to a group element $X \in \mathbb{G}$, the prover computes $\boldsymbol{C} = (1, 1, X) \cdot \boldsymbol{g_1}^r \cdot \boldsymbol{g_2}^s \cdot \boldsymbol{g_3}^t$ with $r, s, t \xleftarrow{R} \mathbb{Z}_p$. When the proof system is configured to provide perfectly sound proofs, $\boldsymbol{g_3}$ is set as $\boldsymbol{g_3} = \boldsymbol{g_1}^{\xi_1} \odot \boldsymbol{g_2}^{\xi_2}$ with $\xi_1, \xi_2 \xleftarrow{R} \mathbb{Z}_p$. In this case, commitments $\boldsymbol{C} = (g_1^{r+\xi_1 t}, g_2^{s+\xi_2 t}, X \cdot g^{r+s+t(\xi_1+\xi_2)})$ can be interpreted as Boneh-Boyen-Shacham (BBS) ciphertexts as $X$ can be recovered by running the BBS decryption algorithm using the private key $(\alpha_1, \alpha_2) = (\log_g(g_1), \log_g(g_2))$. When the CRS is set up to give perfectly witness indistinguishable (WI) proofs, $\boldsymbol{g_1}, \boldsymbol{g_2}$ and $\boldsymbol{g_3}$ are linearly independent vectors, so that $\boldsymbol{C}$ is a perfectly hiding commitment to $X \in \mathbb{G}$: a typical choice is $\boldsymbol{g_3} = \boldsymbol{g_1}^{\xi_1} \odot \boldsymbol{g_2}^{\xi_2} \odot (1, 1, g)^{-1}$. Under the DLIN assumption, the two distributions of CRS are computationally indistinguishable.

In order to prove that a set of algebraic equations is satisfiable, the prover generates one commitment per variable and one proof element (made of a constant number of elements) per relation. Efficient proofs are available for pairing-product relations, which are of the type

$$\prod_{i=1}^{n} e(\mathcal{A}_i, \mathcal{X}_i) \cdot \prod_{i=1}^{n} \cdot \prod_{j=1}^{n} e(\mathcal{X}_i, \mathcal{X}_j)^{a_{ij}} = t_T, \tag{11}$$

for variables $\mathcal{X}_1, \ldots, \mathcal{X}_n \in \mathbb{G}$ and constants $t_T \in \mathbb{G}_T$, $\mathcal{A}_1, \ldots, \mathcal{A}_n \in \mathbb{G}$, $a_{ij} \in \mathbb{Z}_p$, for $i, j \in \{1, \ldots, n\}$.

Non-interactive proofs for quadratic equations require 9 group elements. Linear pairing-product equations (with $a_{ij} = 0$ for all $i, j$) only cost 3 group elements each.

## D Deferred Proofs for Theorem 1

The following lemma shows that, unless there exists a DLIN distinguisher, the adversary $\mathcal{A}$ necessarily outputs the same kind of signatures as the signing oracle during the entire subsequence of games between Game 2.1 and Game 2.$L$.

**Lemma 3.** *If the DLIN assumption holds in $\mathbb{G}$, $\mathcal{A}$'s probability to output a Type B signature is about the same in Game 2.$k$ and Game 2.$(k-1)$ for any $k \in \{2, \ldots, L\}$.*

*Proof.* Towards a contradiction, we assume that there exist an adversary $\mathcal{A}$ and an index $k \in \{2, \ldots, L\}$ such that events $S_{2.k} \wedge E_{2.k}$ and $S_{2.(k-1)} \wedge E_{2.(k-1)}$ occur with significantly different probabilities in Game 2.$k$ and Game 2.$(k-1)$, respectively. We turn $\mathcal{A}$ into a DLIN distinguisher $\mathcal{B}$ in $\mathbb{G}$. Algorithm $\mathcal{B}$ takes as input $(f, g, h, f^a, h^b, T)$ and decides if $T = g^{a+b}$ or $T \in_R \mathbb{G}$. As in [25, Lemma 6], $\mathcal{B}$ uses the random self-reducibility of DLIN to build $q$ tuples

$$(F_j = f^{a_j}, H_j = h^{b_j}, T_j)$$

such that, for each $j \in \{1, \ldots, q\}$, we have

$$T_j = \begin{cases} g^{a_j + b_j} & \text{if } T = g^{a+b} \\ g^{a_j + b_j + \tau_j} & \text{if } T \in_R \mathbb{G} \end{cases}$$

for $\tau_1, \ldots, \tau_q \in_R \mathbb{Z}_p$. This is done by picking $\rho_j, \rho_{a_j}, \rho_{b_j} \xleftarrow{R} \mathbb{Z}_p$ and setting

$$(F_j, H_j, T_j) = \left((f^a)^{\rho_j} \cdot f^{\rho_{a_j}}, (h^b)^{\rho_j} \cdot h^{\rho_{b_j}}, T^{\rho_j} \cdot g^{\rho_{a_j} + \rho_{b_j}}\right), \qquad \forall j \in \{1, \ldots, q\} \ .$$

Before generating the public key, $\mathcal{B}$ flips a fair binary coin $b^\dagger \xleftarrow{R} \{0,1\}$ as a guess that the $k$-th bit of the forgery message $M^\star = M[1]^\star \dots M[L]^\star \in \{0,1\}^L$ will happen to be $b^\dagger$. To construct $PK$, $\mathcal{B}$ picks $u_1, u_2 \xleftarrow{R} \mathbb{G}$, $\omega_1, \omega_2 \xleftarrow{R} \mathbb{Z}_p$, $\boldsymbol{\alpha} = (\alpha_{1,0}, \alpha_{1,1}, \dots, \alpha_{L,0}, \alpha_{L,1}) \xleftarrow{R} \mathbb{Z}_p^{2L}$, $\boldsymbol{\beta} = (\beta_{1,0}, \beta_{1,1}, \dots, \beta_{L,0}, \beta_{L,1}) \xleftarrow{R} \mathbb{Z}_p^{2L}$ and $\zeta \xleftarrow{R} \mathbb{Z}_p$. It sets $\Omega_1 = u_1^{\omega_1}$, $\Omega_2 = u_2^{\omega_2}$ and defines the vectors $\boldsymbol{V} = (V_{1,0}, V_{1,1}, \dots, V_{L,0}, V_{L,1})$, $\boldsymbol{W} = (W_{1,0}, W_{1,1}, \dots, W_{L,0}, W_{L,1})$ as

$$\begin{aligned}
&(V_{\ell,0}, V_{\ell,1}) = (f^{\alpha_{\ell,0}}, f^{\alpha_{\ell,1}}), && (W_{\ell,0}, W_{\ell,1}) = (h^{\beta_{\ell,0}}, h^{\beta_{\ell,1}}), && \text{if } \ell \neq k,\\
&(V_{k,1-b^\dagger}, V_{k,b^\dagger}) = (f^{\alpha_{k,1-b^\dagger}} \cdot g^{\zeta}, f^{\alpha_{k,b^\dagger}}), && (W_{k,1-b^\dagger}, W_{k,b^\dagger}) = (h^{\beta_{k,1-b^\dagger}} \cdot g^{\zeta}, h^{\beta_{k,b^\dagger}}) .
\end{aligned}$$

The rest of $PK$, including $(\mathsf{sk}_{hsps}, \mathsf{pk}_{hsps})$ and $\{(Z_i, R_i, U_i)\}_{i=1}^{4L+2}$, is generated as in the real setup algorithm.

The adversary $\mathcal{A}$ is run on input of

$$PK = \Big( f,\ g,\ h,\ u_1,\ u_2,\ \Omega_1 = {u_1}^{\omega_1},\ \Omega_2 = {u_2}^{\omega_2},\ \boldsymbol{V},\ \boldsymbol{W},\ \mathsf{pk}_{hsps} = \big(\hat{g_z},\ \hat{g_r},\ \hat{h_z},\ \hat{h_u},\ \{(\hat{g}_i, \hat{h}_i)\}_{i=1}^{4L+3}\big),\ \{(\hat{Z}_j, \hat{R}_j, \hat{U}_j)\}_{j=1}^{4L+2}\Big)$$

and the challenger $\mathcal{B}$ keeps $(\omega_1, \omega_2, \{\chi_i, \gamma_i, \delta_i\}_{i=1}^{4L+3})$ to itself.

At the beginning of the game, $\mathcal{B}$ also chooses a random function $R_{k-1} : \{0,1\}^{k-1} \to \mathbb{G}$ which will be implicitly used to construct another random function $R_k : \{0,1\}^k \to \mathbb{G}$ with larger domain such that, for any string $M \in \{0,1\}^{k-1}$, we have $R_k(M||b^\dagger) = R_{k-1}(M)$ while $R_k(M||1-b^\dagger)$ takes an independent random value.

Then, $\mathcal{B}$ starts answering signing queries as follows. In order to handle the $j$-th signing query $M^j = M[1]^j \dots M[L]^j \in \{0,1\}^L$, the response of $\mathcal{B}$ depends on the $k$-th bit $M[k]^j$ of $M^j$. Specifically, $\mathcal{B}$ considers three cases.

– If $M[k]^j = b^\dagger$, $\mathcal{B}$ uses the property that $R_k(M^j_{|k}) = R_{k-1}(M^j_{|k-1})$. It chooses $r, s \xleftarrow{R} \mathbb{Z}_p$ and sets

$$\sigma_1 = g^{\omega_1+\omega_2} \cdot R_{k-1}(M^j_{|k-1}) \cdot H(\boldsymbol{V}, M)^r \cdot H(\boldsymbol{W}, M)^s, \qquad \sigma_2 = f^r, \qquad \sigma_3 = h^s,$$

where $H(\boldsymbol{V}, M) = \prod_{\ell=1}^{L} V_{\ell,M[\ell]}$ and $H(\boldsymbol{W}, M) = \prod_{\ell=1}^{L} W_{\ell,M[\ell]}$. The $(Z, R, U)$ components of the private key are computed by generating a homomorphic structure-preserving signature on the vector

$$(\sigma_1, \sigma_2^{1-M[1]}, \sigma_2^{M[1]}, \dots, \sigma_2^{1-M[L]}, \sigma_2^{M[L]}, \sigma_3^{1-M[1]}, \sigma_3^{M[1]}, \dots, \sigma_3^{1-M[L]}, \sigma_3^{M[L]}, \Omega_1, \Omega_2),$$

by computing

$$\begin{cases}
Z = \sigma_1^{-\chi_1} \cdot \sigma_2^{-\sum_{i=1}^{L} \chi_{2i+M[i]}} \cdot \sigma_3^{-\sum_{i=1}^{L} \chi_{2L+2i+M[i]}} \cdot \Omega_1^{-\chi_{4L+2}} \cdot \Omega_2^{-\chi_{4L+3}} \\
R = \sigma_1^{-\gamma_1} \cdot \sigma_2^{-\sum_{i=1}^{L} \gamma_{2i+M[i]}} \cdot \sigma_3^{-\sum_{i=1}^{L} \gamma_{2L+2i+M[i]}} \cdot \Omega_1^{-\gamma_{4L+2}} \cdot \Omega_2^{-\gamma_{4L+3}} \\
U = \sigma_1^{-\delta_1} \cdot \sigma_2^{-\sum_{i=1}^{L} \delta_{2i+M[i]}} \cdot \sigma_3^{-\sum_{i=1}^{L} \delta_{2L+2i+M[i]}} \cdot \Omega_1^{-\delta_{4L+2}} \cdot \Omega_2^{-\delta_{4L+3}}
\end{cases} . \tag{12}$$

Note that the obtained $(Z, R, U)$ can be written

$$\begin{cases}
Z = R_{k-1}(M^j_{|k-1})^{-\chi_1} \cdot Z_{4L+1}^{\omega_1} \cdot Z_{4L+2}^{\omega_2} \cdot \prod_{i=1}^{L} \left(Z^r_{2i-\overline{M[i]}} \cdot Z^s_{2L+2i-\overline{M[i]}}\right) \\
R = R_{k-1}(M^j_{|k-1})^{-\gamma_1} \cdot R_{4L+1}^{\omega_1} \cdot R_{4L+2}^{\omega_2} \cdot \prod_{i=1}^{L} \left(R^r_{2i-\overline{M[i]}} \cdot R^s_{2L+2i-\overline{M[i]}}\right) \\
U = R_{k-1}(M^j_{|k-1})^{-\delta_1} \cdot U_{4L+1}^{\omega_1} \cdot U_{4L+2}^{\omega_2} \cdot \prod_{i=1}^{L} \left(U^r_{2i-\overline{M[i]}} \cdot U^s_{2L+2i-\overline{M[i]}}\right)
\end{cases}$$

We remark that $(\sigma_1, \sigma_2, \sigma_3, Z, R, U)$ matches the distribution of signatures in both Game 2.$(k-1)$ and Game 2.$k$.

– If $M[k]^j = 1 - b^\dagger$ and $R_k(M^j_{|_k})$ has not been defined yet, $\mathcal{B}$ implicitly defines

$$R_k(M^j_{|_k}) = R_k(M^j_{|_{k-1}}||1-b^\dagger) = \begin{cases} R_{k-1}(M^j_{|_{k-1}}) \cdot g^{\zeta \cdot \tau_j} & \text{if } T \in_R \mathbb{G} \\ R_{k-1}(M^j_{|_{k-1}}) & \text{if } T = g^{a+b} \end{cases}$$

Namely, $\mathcal{B}$ uses the $j$-th tuple $(F_j, H_j, T_j)$ to set

$$\sigma_1 = g^{\omega_1+\omega_2} \cdot R_{k-1}(M^j_{|_{k-1}}) \cdot F_j^{\sum_{\ell=1}^L \alpha_{\ell,M[\ell]}} \cdot H_j^{\sum_{\ell=1}^L \beta_{\ell,M[\ell]}} \cdot T_j^{\zeta},$$
$$\sigma_2 = F_j = f^{a_j}, \qquad \sigma_3 = H_j = h^{b_j}.$$

If $T_j = g^{a_j+b_j}$, the above implicitly defines

$$\sigma_1 = g^{\omega_1+\omega_2} \cdot R_{k-1}(M^j_{|_{k-1}}) \cdot H(\boldsymbol{V}, M^j)^{a_j} \cdot H(\boldsymbol{W}, M^j)^{b_j},$$

so that $(\sigma_1, \sigma_2, \sigma_3)$ has the same distribution as in Game 2.$(k-1)$. If $T_j = g^{a_j+b_j+\tau_j}$, we can write

$$\sigma_1 = g^{\omega_1+\omega_2} \cdot R_k(M^j_{|_k}) \cdot H(\boldsymbol{V}, M^j)^{a_j} \cdot H(\boldsymbol{W}, M^j)^{b_j},$$

since $R_k(M^j_{|_k}) = R_{k-1}(M^j_{|_{k-1}}) \cdot g^{\zeta \cdot \tau_j}$, which is distributed as in Game 2.$k$. In either case, $(Z, R, U)$ are computed using $\mathsf{sk}_{hsps} = \{(\chi_i, \gamma_i, \delta_i)\}_{i=1}^{4L+3}$ as in the previous case (i.e., as per (12)).

– If $M[k]^j = 1 - b^\dagger$ and $R_k(M^j_{|_k})$ was previously defined, $\mathcal{B}$ recalls the index $j' < j$ of the query where this value was defined. It picks $r, s \xleftarrow{R} \mathbb{Z}_p$ and re-uses the $j'$-th tuple $(F_{j'}, H_{j'}, T_{j'})$ to set

$$\sigma_1 = g^{\omega_1+\omega_2} \cdot R_{k-1}(M^{j'}_{|_{k-1}}) \cdot F_{j'}^{\sum_{\ell=1}^L \alpha_{\ell,M[\ell]}} \cdot H_{j'}^{\sum_{\ell=1}^L \beta_{\ell,M[\ell]}} \cdot T_{j'}^{\zeta} \cdot H(\boldsymbol{V}, M^{j'})^r \cdot H(\boldsymbol{W}, M^{j'})^s,$$
$$\sigma_2 = F_{j'} = f^{a_{j'}} \cdot f^r, \qquad \sigma_3 = H_{j'} = h^{b_{j'}} \cdot h^s,$$

and generates $(Z, R, U)$ as in the previous cases.

In the forgery stage, $\mathcal{A}$ outputs a new message $M^\star$ with a forgery $\sigma^\star = (\sigma_1^\star, \sigma_2^\star, \sigma_3^\star, Z^\star, R^\star, U^\star)$ and our DLIN distinguisher $\mathcal{B}$ must figure out if this forgery is of the same type as the outputs of the signing oracle. At this point, $\mathcal{B}$ halts and outputs a random bit in the event that $M[k]^\star \neq b^\dagger$. Otherwise, i.e., if $M[k]^\star = b^\dagger$, $\mathcal{B}$ is able to compute $F(\boldsymbol{v}, M^\star) = \sum_{\ell=1}^L \alpha_{\ell,M[\ell]^\star}$ and $F(\boldsymbol{w}, M^\star) = \sum_{\ell=1}^L \beta_{\ell,M[\ell]^\star}$, which yields

$$\eta^\star = \sigma_1^\star \cdot \sigma_2^{\star -F(\boldsymbol{v},M^\star)} \cdot \sigma_3^{\star -F(\boldsymbol{w},M^\star)}.$$

If $\eta^\star = g^{\omega_1+\omega_2} \cdot R_{k-1}(M^\star_{|_{k-1}})$, $\mathcal{B}$ concludes that $(\sigma^\star, M^\star)$ is a forgery of the same type as signatures generated by the signing oracle and outputs 1. Recall that $R_{k-1}(M^\star_{|_{k-1}}) = R_k(M^\star_{|_k})$, so that $\sigma^\star$ has the same distribution as outputs of the signing oracle in both Game 2.$k$ and Game 2.$(k-1)$. Otherwise, it concludes that the distribution of $\sigma^\star$ departs from the output distribution of the signing oracle and outputs 0. If the difference between the forgery's probability to mimic the behavior of the signing oracle in games 2.$k$ and 2.$(k-1)$ is $\epsilon$, we find that $\mathcal{B}$'s advantage as a DLIN distinguisher is at least $\epsilon/2$ since $\mathcal{B}$'s choice for $b^\dagger \in \{0,1\}$ is independent of $\mathcal{A}$'s view. □

## E Shorter Signatures Under the $K$-Linear Assumption

Let us first recall the following generalization of the DDH and Decision Linear assumptions.

**Definition 4.** *The* $K$-Linear assumption *states that, given* $(g, g_1, \ldots, g_K, g_1^{a_1}, \ldots, g_K^{a_K}, T) \in \mathbb{G}^{2K+2}$ *for* $K > 0$*, no PPT algorithm can decide if* $T = g^{a_1+\cdots+a_K}$ *or* $T \in_R \mathbb{G}$*.*

For $K = 1$ (resp. $K = 2$), the $K$-linear assumption coincides with the DDH (resp. DLIN) assumption.

To instantiate our signature schemes under the $K$-linear assumption with $K > 2$, we first need to extend the one-time linearly homomorphic structure-preserving signature of [42]. To this end, we need to define the following assumption which is implied by the $K$-linear assumption in the same way as SDP is implied by DLIN.

**Definition 5.** *The* Simultaneous $K$-wise Pairing *($K$-SDP) problem is, given a random tuple* $(\hat{g}_{1,z}, \ldots, \hat{g}_{K,z}, \hat{g}_{1,r}, \ldots, \hat{g}_{K,r}) \in_R \hat{\mathbb{G}}^{2K}$, *to find a non-trivial vector* $(z, r_1, \ldots, r_K) \in \mathbb{G}^{K+1}$ *such that* $e(z, \hat{g}_{j,z}) \cdot e(r_j, \hat{g}_{j,r}) = 1_{\mathbb{G}_T}$ *for each* $j \in \{1, \ldots, K\}$ *and* $z \neq 1_{\mathbb{G}}$.

For a $K$-linear instance $(\hat{g}_{1,r}, \ldots, \hat{g}_{K,r}, \hat{g}_{1,r}^{a_1}, \ldots, \hat{g}_{K,r}^{a_K}, T) \in \hat{\mathbb{G}}^{2K+1}$, given any algorithm that is able to find a non-trivial tuple $(z, r_1, \ldots, r_K) \in \mathbb{G}^{K+1}$ satisfying $e(z, \hat{g}_{j,r}^{a_j}) \cdot e(r_j, \hat{g}_{j,r}) = 1_{\mathbb{G}_T}$ for each $j \in \{1, \ldots, K\}$, we can use the equality

$$T = \hat{g}^{\sum_{j=1}^{K} a_j} \iff e(\prod_{j=1}^{K} r_j, \hat{g}) \cdot e(z, T) = 1_{\mathbb{G}_T} \ .$$

to solve the given $K$-linear instance. For this reason, any algorithm solving $K$-SDP with non-negligible probability implies a $K$-linear distinguisher.

Under the $K$-SDP assumption, the one-time linearly homomorphic structure-preserving signature of [42] can be extended as follows.

**Keygen($\lambda, n$):** Given a security parameter $\lambda$ and the dimension $n \in \mathbb{N}$ of vectors to be signed, choose bilinear group $(\mathbb{G}, \hat{\mathbb{G}}, \mathbb{G}_T)$ of prime order $p$. For $j = 1$ to $K$, choose generators $\hat{g}_{j,z}, \hat{g}_{j,r} \xleftarrow{R} \hat{\mathbb{G}}$. Then, for each $i = 1$ to $n$, $j = 1$ to $K$, pick $\chi_i \xleftarrow{R} \mathbb{Z}_p$ , $\gamma_{j,i} \xleftarrow{R} \mathbb{Z}_p$ and compute $\hat{g}_{j,i} = \hat{g}_{j,z}^{\chi_i} \hat{g}_{j,r}^{\gamma_{j,i}}$. The private key is $\mathsf{sk} = \big(\{\chi_i, \{\gamma_{j,i}\}_{j=1}^{K}\}_{i=1}^{n}\big)$ while the public key is

$$\mathsf{pk} = \Big(\{\hat{g}_{j,z}, \hat{g}_{j,r}, \{\hat{g}_{j,i}\}_{i=1}^{n}\}_{j=1}^{K}\Big) \ .$$

**Sign($\mathsf{sk}, (M_1, \ldots, M_n)$):** To sign $(M_1, \ldots, M_n) \in \mathbb{G}^n$ using $\mathsf{sk} = \big(\{\chi_i, \{\gamma_{j,i}\}_{j=1}^{K}\}_{i=1}^{n}\big)$, compute and output $\sigma = (z, r_1, \ldots, r_K) \in \mathbb{G}^{K+1}$, where

$$\begin{cases} z = \prod_{i=1}^{n} M_i^{-\chi_i}, \\ r_j = \prod_{i=1}^{n} M_i^{-\gamma_{j,i}} \quad j \in \{1, \ldots, K\} \ . \end{cases}$$

**SignDerive($\mathsf{pk}, \{(\omega_i, \sigma^{(i)})\}_{i=1}^{\ell}$):** Given a public key $\mathsf{pk}$ and $\ell$ tuples $(\omega_i, \sigma^{(i)})$, where $\omega_i \in \mathbb{Z}_p$ for each $i$, parse $\sigma^{(i)}$ as $\sigma^{(i)} = \big(z_i, r_{i,1}, \ldots, r_{i,K}\big) \in \mathbb{G}^{k+1}$ for $i = 1$ to $\ell$. Then, compute and return $\sigma = (z, r_1, \ldots, r_k)$, where $z = \prod_{i=1}^{\ell} z_i^{\omega_i}$, $r_j = \prod_{i=1}^{\ell} r_{i,j}^{\omega_i}$ for $j = 1$ to $K$.

**Verify($\mathsf{pk}, \sigma, (M_1, \ldots, M_n)$):** Given $\sigma = (z, r_1, \ldots, r_K) \in \mathbb{G}^{K+1}$ and $(M_1, \ldots, M_n)$, return 1 if and only if $(M_1, \ldots, M_n) \neq (1_{\mathbb{G}}, \ldots, 1_{\mathbb{G}})$ and, for each $j \in \{1, \ldots, K\}$, the following equality holds:

$$1_{\mathbb{G}_T} = e(z, \hat{g}_{j,z}) \cdot e(r_j, \hat{g}_{j,r}) \cdot \prod_{i=1}^{n} e(M_i, \hat{g}_{j,i}) \ .$$

Using the above LHSPS scheme, our signature scheme of Section 3 can be modified so as to rely on the $K$-linear assumption with $K > 2$. The construction goes as follows.

**Keygen($\lambda$):** Choose bilinear groups $(\mathbb{G}, \hat{\mathbb{G}}, \mathbb{G}_T)$ of prime order $p$ together with generators $f_1, \ldots, f_K, u_1, \ldots, u_K \xleftarrow{R} \mathbb{G}$.

1. For $j = 1$ to $K$ and $\ell = 1$ to $L$, choose $V_{j,\ell,0}, V_{j,\ell,1} \xleftarrow{R} \mathbb{G}$ to assemble row vectors

$$\boldsymbol{V}_j = (V_{j,1,0}, V_{j,1,1}, \ldots, V_{j,L,0}, V_{j,L,1}) \in \mathbb{G}^{2L} \qquad \forall j \in \{1, \ldots, K\} \ .$$

2. Define $\mathbf{M} \in \mathbb{G}^{K(2L+1)\times(K(2L+1)+1)}$ as the matrix

$$\big(M_{i,j}\big)_{i,j} = \left(\begin{array}{c|c|c|c|c|c|c|c} \boldsymbol{V}_1^\top & \mathbf{Id}_{f_1,2L} & \ldots & \mathbf{1}^{2L\times 2L} & \mathbf{1}^{2L\times 1} & \ldots & \ldots & \mathbf{1}^{2L\times 1} \\ \hline \vdots & \ddots & \ddots & & \vdots & \ddots & & \vdots \\ \hline \boldsymbol{V}_K^\top & \mathbf{1}^{2L\times 2L} & \ldots & \mathbf{Id}_{f_K,2L} & \mathbf{1}^{2L\times 1} & & \vdots & \mathbf{1}^{2L\times 1} \\ \hline g & \mathbf{1}^{1\times 2L} & & \mathbf{1}^{1\times 2L} & u_1 & 1 & \ldots & 1 \\ \hline g & \mathbf{1}^{1\times 2L} & & \mathbf{1}^{1\times 2L} & 1 & u_2 & & \vdots \\ \hline g & \mathbf{1}^{1\times 2L} & & \mathbf{1}^{1\times 2L} & \vdots & \ldots & \ddots & \vdots \\ \hline g & \mathbf{1}^{1\times 2L} & \ldots & \mathbf{1}^{1\times 2L} & 1 & \ldots & & u_K \end{array}\right)$$

with $\mathbf{Id}_{f_j,2L} = f_j^{\mathbf{I}_{2L}} \in \mathbb{G}^{2L\times 2L}$ for each $j \in \{1, \ldots, K\}$, where $\mathbf{I}_{2L} \in \mathbb{Z}_p^{2L\times 2L}$ is the identity matrix.

3. Generate a key pair $(\mathsf{sk}_{hsps}, \mathsf{pk}_{hsps})$ for the one-time homomorphic signature of Section 2.2 in order to sign vectors of dimension $n = K(2L+1)+1$. Let $\mathsf{sk}_{hsps} = \big(\{\chi_i, \{\gamma_{j,i}\}_{j=1}^K\}_{i=1}^{K(2L+1)+1}\big)$ be the private key, of which the corresponding public key is

$$\mathsf{pk}_{hsps} = \Big(\{\hat{g}_{j,z}, \hat{g}_{j,r}, \{\hat{g}_{j,i}\}_{i=1}^n\}_{j=1}^{K(2L+1)+1}\Big) \ .$$

4. Using $\mathsf{sk}_{hsps}$, generate one-time homomorphic signatures $\{(Z_i, R_{i,1}, \ldots, R_{i,K})\}_{i=1}^{K(2L+1)}$ on the rows $\boldsymbol{M}_i = (M_{i,1}, \ldots, M_{i,K(2L+1)+1}) \in \mathbb{G}^{K(2L+1)+1}$ of $\mathbf{M}$ and erase $\mathsf{sk}_{hsps}$.
5. Choose $\omega_1, \ldots, \omega_K \xleftarrow{R} \mathbb{Z}_p$ and compute $\Omega_i = u_i^{\omega_i} \in \mathbb{G}$ for $i = 1$ to $K$.

The private key consists of $SK = (\omega_1, \ldots, \omega_K)$ and the public key is

$$PK = \Big(\{(f_i, u_i, \Omega_i),\ \boldsymbol{V}_i\}_{i=1}^K,\ \mathsf{pk}_{hsps},\ \{(Z_i, R_{i,1}, \ldots, R_{i,K})\}_{i=1}^{K(2L+1)}\Big) \ .$$

**Sign($SK, M$):** Given a message $M = M[1] \ldots M[L] \in \{0,1\}^L$ and $SK = \{(\omega_1, \ldots, \omega_K), \mathsf{sk}_{hsps}\}$:

1. Choose $r_1, \ldots, r_K \xleftarrow{R} \mathbb{Z}_p$ and compute

$$\begin{aligned} \sigma_0 &= g^{\sum_{j=1}^K \omega_j} \cdot \prod_{j=1}^K H(\boldsymbol{V}_j, M)^{r_j} \\ \sigma_j &= f_j^{r_j} \qquad\qquad \forall j \in \{1, \ldots, K\} \end{aligned}$$

where $H(\boldsymbol{V}_j, M) = \prod_{\ell=1}^L V_{j,\ell,M[\ell]}$ for each $j \in \{1, \ldots, K\}$.

2. Using $\{(Z_i, R_{i,1}, \ldots, R_{i,K})\}_{i=1}^{K(2L+1)}$, derive a one-time homomorphic signature $(Z, R_1, \ldots, R_K)$ which will argue that the vector

$$(\sigma_0, \sigma_1^{1-M[1]}, \sigma_1^{M[1]}, \ldots, \sigma_1^{1-M[L]}, \sigma_1^{M[L]}, \ldots, \sigma_K^{1-M[1]}, \sigma_K^{M[1]}, \ldots, \sigma_K^{1-M[L]}, \sigma_K^{M[L]}, \Omega_1, \ldots, \Omega_K)$$

is in the row space of $\mathbf{M}$ and guarantee that $(\sigma_0, \sigma_1, \ldots, \sigma_K)$ was generated as per Step 1.

Return the signature $\sigma = \big(\sigma_0, \sigma_1, \ldots, \sigma_K, Z, R_1, \ldots, R_K\big) \in \mathbb{G}^{2K+2}$.

**Verify($PK, M, \sigma$):** Parse $\sigma$ as $(\sigma_0, \sigma_1, \ldots, \sigma_K, Z, R_1, \ldots, R_K) \in \mathbb{G}^{2K+2}$ and return 1 if and only if the following equations hold for each $j \in \{1, \ldots, K\}$.

$$e(Z, \hat{g}_{j,z}) \cdot \prod_{j=1}^{K} e(R_j, \hat{g}_{j,r}) = e(\sigma_0, \hat{g}_{j,1})^{-1} \cdot e(\sigma_1, \prod_{i=1}^{L} \hat{g}_{j,2i+M[i]})^{-1} \cdot \cdots e(\sigma_K, \prod_{i=1}^{L} \hat{g}_{j,2(K-1)L+2i+M[i]})^{-1} \cdot \prod_{i=1}^{K} e(\Omega_i, \hat{g}_{j,2KL+1+i})^{-1} \ .$$

The security proof is completely similar to the proof of Theorem 1.

Under the SXDH assumption (with $K = 1$), we have the same signature size as [25]. However, the above scheme saves $2K - 2$ elements when $K > 1$ and even $2K - 1$ if the QA-NIZK proof of [36] is used. We further note that, under the SXDH assumption, it is possible to obtain signatures consisting of only 3 group elements if we replace the LHSPS-based QA-NIZK proof $(Z, R_1)$ of [43] by the one of Jutla and Roy [37].

Under the $K$-linear assumption, the proof sizes of [43] and [37] are $K + 1$ and $K$, respectively. In the SXDH-based variant with shorter signatures, each signature consists of a pair

$$(\sigma_1, \sigma_2) = (u^{\omega} \cdot H(\boldsymbol{V}, M)^r, f^r) \in \mathbb{G}^2$$

and a QA-NIZK proof $Z \in \mathbb{G}$, obtained from [37], that $(\sigma_1, \sigma_2)$ has the correct form.

## F Strongly Unforgeable Signatures

This section shows a simple modification of our DLIN-based signature scheme which provides strong unforgeability.

A useful property of the one-time linearly homomorphic signature of Section 2.2 is that, while the signing algorithm is deterministic, signatures are not unique. However, even if the private key is available, it is computationally infeasible to find two distinct signatures on a given vector (unless the SDP assumption is false). This property comes in handy to build strongly unforgeable signatures.

**Keygen($\lambda$):** Choose bilinear groups $(\mathbb{G}, \hat{\mathbb{G}}, \mathbb{G}_T)$ of prime order $p$ together with $f, g, h, u_1, u_2 \xleftarrow{R} \mathbb{G}$.

1. For $\ell = 1$ to $L$, choose $V_{\ell,0}, V_{\ell,1}, W_{\ell,0}, W_{\ell,1} \xleftarrow{R} \mathbb{G}$ to assemble row vectors

$$\boldsymbol{V} = (V_{1,0}, V_{1,1}, \ldots, V_{L,0}, V_{L,1}) \in \mathbb{G}^{2L}, \qquad \boldsymbol{W} = (W_{1,0}, W_{1,1}, \ldots, W_{L,0}, W_{L,1}) \in \mathbb{G}^{2L} \ .$$

2. Define the matrix

$$\mathbf{M} = \big(M_{i,j}\big)_{i,j} = \left(\begin{array}{c|c|c|c|c} \boldsymbol{V}^{\top} & \mathbf{Id}_{f,2L} & \mathbf{1}^{2L\times 2L} & \mathbf{1}^{2L\times 1} & \mathbf{1}^{2L\times 1} \\ \hline \boldsymbol{W}^{\top} & \mathbf{1}^{2L\times 2L} & \mathbf{Id}_{h,2L} & \mathbf{1}^{2L\times 1} & \mathbf{1}^{2L\times 1} \\ \hline g & \mathbf{1}^{1\times 2L} & \mathbf{1}^{1\times 2L} & u_1 & 1 \\ \hline g & \mathbf{1}^{1\times 2L} & \mathbf{1}^{1\times 2L} & 1 & u_2 \end{array}\right) \in \mathbb{G}^{(4L+2)\times(4L+3)}$$

with $\mathbf{Id}_{f,2L} = f^{\mathbf{I}_{2L}} \in \mathbb{G}^{2L\times 2L}$, $\mathbf{Id}_{h,2L} = h^{\mathbf{I}_{2L}} \in \mathbb{G}^{2L\times 2L}$, where $\mathbf{I}_{2L} \in \mathbb{Z}_p^{2L\times 2L}$ is the identity matrix.

3. Generate a key pair $(\mathsf{sk}_{hsps}, \mathsf{pk}_{hsps})$ for the one-time linearly homomorphic signature of Section 2.2 in order to sign vectors of $n = 4L + 3$ group elements. Let $\mathsf{sk}_{hsps} = \{(\chi_i, \gamma_i, \delta_i)\}_{i=1}^{4L+3}$ be the private key. The matching public key is $\mathsf{pk}_{hsps} = \big(\hat{g_z},\ \hat{g_r},\ \hat{h_z},\ \hat{h_u},\ \{(\hat{g}_i, \hat{h}_i)\}_{i=1}^{4L+3}\big)$.

4. Using $\mathsf{sk}_{hsps} = \{\chi_i, \gamma_i, \delta_i\}_{i=1}^{4L+3}$, generate one-time homomorphic signatures $\{(Z_j, R_j, U_j)\}_{j=1}^{4L+2}$ on the rows $\boldsymbol{M}_j = (M_{j,1}, \ldots, M_{j,4L+3}) \in \mathbb{G}^{4L+3}$ of the matrix $\mathbf{M}$. These are obtained as

$$(Z_j, R_j, U_j) = \Big( \prod_{i=1}^{4L+2} M_{j,i}^{-\chi_i},\ \prod_{i=1}^{4L+2} M_{j,i}^{-\gamma_i},\ \prod_{i=1}^{4L+2} M_{j,i}^{-\delta_i} \Big) \qquad \forall j \in \{1, \ldots, 4L+2\}\ .$$

5. Choose $\omega_1, \omega_2 \stackrel{R}{\leftarrow} \mathbb{Z}_p$ and compute $\Omega_1 = u_1^{\omega_1} \in \mathbb{G}$, $\Omega_2 = u_2^{\omega_2} \in \mathbb{G}$.
6. Choose a collision-resistant hash function $H : \{0,1\}^* \to \{0,1\}^L$.

The private key consists of $SK = (\omega_1, \omega_2)$ and the public key is

$$PK = \Big( f,\ g,\ h,\ u_1,\ u_2,\ \Omega_1,\ \Omega_2,\ \boldsymbol{V},\ \boldsymbol{W},\ \mathsf{pk}_{hsps} = \big(\hat{g_z},\ \hat{g_r},\ \hat{h_z},\ \hat{h_u},\ \{(\hat{g}_i, \hat{h}_i)\}_{i=1}^{4L+3}\big),\ \{(Z_j, R_j, U_j)\}_{j=1}^{4L+2},\ H \Big)\ .$$

**Sign($\boldsymbol{SK}$, Msg):** Given a message $\mathsf{Msg} \in \{0,1\}^*$ and the private key $SK = (\omega_1, \omega_2)$, do the following.

1. Choose $r, s \stackrel{R}{\leftarrow} \mathbb{Z}_p$ and compute

$$\sigma_2 = f^r \qquad\qquad \sigma_3 = h^s\,, \tag{13}$$

as well as $M = H(\mathsf{Msg}, \sigma_2, \sigma_3) = M[1] \ldots M[L] \in \{0,1\}^L$. Then, compute

$$\sigma_1 = g^{\omega_1+\omega_2} \cdot H(\boldsymbol{V}, M)^r \cdot H(\boldsymbol{W}, M)^s$$

where $H(\boldsymbol{V}, M) = \prod_{\ell=1}^{L} V_{\ell, M[\ell]}$ and $H(\boldsymbol{W}, M) = \prod_{\ell=1}^{L} W_{\ell, M[\ell]}$.

2. Using $\{(Z_j, R_j, U_j)\}_{j=1}^{4L+2}$, derive a homomorphic signature $(Z, R, U)$ which will serve as a non-interactive argument showing that

$$(\sigma_1, \sigma_2^{1-M[1]}, \sigma_2^{M[1]}, \ldots, \sigma_2^{1-M[L]}, \sigma_2^{M[L]}, \sigma_3^{1-M[1]}, \sigma_3^{M[1]}, \ldots, \sigma_3^{1-M[L]}, \sigma_3^{M[L]}, \Omega_1, \Omega_2) \tag{14}$$

is in the row space of $\mathbf{M}$, which ensures that $(\sigma_1, \sigma_2, \sigma_3)$ is of the form (13).

Return the signature $\sigma = \big(\sigma_1, \sigma_2, \sigma_3, Z, R, U\big) \in \mathbb{G}^6$.

**Verify($\boldsymbol{PK}$, Msg, $\boldsymbol{\sigma}$):** Parse the purported signature $\sigma$ as $\big(\sigma_1, \sigma_2, \sigma_3, Z, R, U\big) \in \mathbb{G}^6$ and compute the $L$-bit string $M = H(\mathsf{Msg}, \sigma_2, \sigma_3) = M[1] \ldots M[L] \in \{0,1\}^L$. Then, return 1 if and only if

$$\begin{aligned} e(Z, \hat{g_z}) \cdot e(R, \hat{g_r}) = e(\sigma_1, \hat{g}_1)^{-1} \cdot e(\sigma_2, \prod_{i=1}^{L} \hat{g}_{2i+M[i]})^{-1} \cdot e(\sigma_3, \prod_{i=1}^{L} \hat{g}_{2L+2i+M[i]})^{-1} \\ \cdot e(\Omega_1, \hat{g}_{4L+2})^{-1} \cdot e(\Omega_2, \hat{g}_{4L+3})^{-1} \\ e(Z, \hat{h_z}) \cdot e(U, \hat{h_u}) = e(\sigma_1, \hat{h}_1)^{-1} \cdot e(\sigma_2, \prod_{i=1}^{L} \hat{h}_{2i+M[i]})^{-1} \cdot e(\sigma_3, \prod_{i=1}^{L} \hat{h}_{2L+2i+M[i]})^{-1} \\ \cdot e(\Omega_1, \hat{h}_{4L+2})^{-1} \cdot e(\Omega_2, \hat{h}_{4L+3})^{-1}\ . \end{aligned}$$

We observe that the signature length is the same as in the scheme of Section 3. The security proof is a simple adaptation of the proof of Theorem 1.

**Theorem 3.** *The scheme provides strong existential unforgeability under chosen-message attacks assuming that (i) $H$ is a collision-resistant hash function; (ii) The DLIN assumption holds in $\mathbb{G}$ and $\hat{\mathbb{G}}$. For $L$-bit messages, for any adversary $\mathcal{A}$, there exist a collision-finding algorithm $\mathcal{B}_0$ for $H$ and DLIN distinguishers $\mathcal{B}$ and $\mathcal{B}'$ in $\hat{\mathbb{G}}$ and $\mathbb{G}$, respectively, such that*

$$\mathbf{Adv}_{\mathcal{A}}(\lambda) \leq \mathbf{Adv}^{\mathrm{CR}}_{\mathcal{B}_0}(\lambda) + \mathbf{Adv}^{\mathrm{DLIN}}_{\mathcal{B}}(\lambda) + 2 \cdot L \cdot \mathbf{Adv}^{\mathrm{DLIN}}_{\mathcal{B}'}(\lambda) + \frac{2}{p}$$

*and with running times $t_{\mathcal{B}_0}, t_{\mathcal{B}}, t_{\mathcal{B}'} \leq t_{\mathcal{A}} + q \cdot \mathsf{poly}(\lambda, L)$.*

*Proof.* The proof is widely similar to that of Theorem 1 and we only outline the changes in the sequence of games. For each $i$, we denote by $S_i$ the event that the challenger outputs 1 in Game $i$.

**Game 0:** This game is the real game. In particular, the adversary always obtains Type A signatures at each signing query. We denote by $S_0$ the event that the adversary wins, in which case the challenger $\mathcal{B}$ outputs 1.

**Game 1:** This game is identical to Game 0 but we raise a failure event $F_1$ which causes $\mathcal{B}$ to output 0 if it occurs. This event $F_1$ is the event that $\mathcal{A}$'s forgery $(\mathsf{Msg}^\star, \sigma^\star = (\sigma_1^\star, \sigma_2^\star, \sigma_3^\star, Z^\star, R^\star, U^\star))$ is such that the signing oracle produced an output $(\mathsf{Msg}_j, \sigma_j = (\sigma_{j,1}, \sigma_{j,2}, \sigma_{j,3}, Z_j, R_j, U_j))$ for which $(\mathsf{Msg}_j, \sigma_{j,2}, \sigma_{j,3}) \neq (\mathsf{Msg}^\star, \sigma_2^\star, \sigma_3^\star)$ but $H(\mathsf{Msg}_j, \sigma_{j,2}, \sigma_{j,3}) = H(\mathsf{Msg}^\star, \sigma_2^\star, \sigma_3^\star)$. Clearly, if event $F_1$ occurs with non-negligible probability, $\mathcal{B}$ can be turned into an algorithm $\mathcal{B}_0$ that breaks the collision-resistance of $H$. We have $|\Pr[S_1] - \Pr[S_0]| \leq \mathbf{Adv}^{\mathrm{CR}}_{\mathcal{B}_0}(\lambda)$.

**Game 2:** This game is like Game 1 but we introduce another failure event $F_2$ that also leads the challenger $\mathcal{B}$ to output 0. We define $F_2$ as the event that $\mathcal{A}$'s forgery $(\mathsf{Msg}^\star, \sigma^\star = (\sigma_1^\star, \sigma_2^\star, \sigma_3^\star, Z^\star, R^\star, U^\star))$ contains a triple $(\sigma_1^\star, \sigma_2^\star, \sigma_3^\star)$ for which there exists an output $(\mathsf{Msg}_j, \sigma_j = (\sigma_{j,1}, \sigma_{j,2}, \sigma_{j,3}, Z_j, R_j, U_j))$ of the signing oracle such that $(\sigma_{j,1}, \sigma_{j,2}, \sigma_{j,3}) = (\sigma_1^\star, \sigma_2^\star, \sigma_3^\star)$ but $(Z^\star, R^\star, U^\star) \neq (Z_j, R_j, U_j)$. Lemma 4 shows that, if $F_2$ happens with non-negligible probability, the DLIN assumption can be broken with nearly the same advantage. We thus have $|\Pr[S_2] - \Pr[S_1]| \leq \Pr[F_2] \leq \mathbf{Adv}^{\mathrm{DLIN}}_{\mathcal{B}}(\lambda)$.

**Game 3:** This game is like Game 2 with the following difference. At the end of the game, the challenger $\mathcal{B}$ checks if $\mathcal{A}$'s forgery is a Type A signature and we define $E_3$ to be the event that the forgery $\sigma^\star$ is a Type A signature. We obviously have $\Pr[S_3] = \Pr[S_3 \wedge E_3] + \Pr[S_3 \wedge \neg E_3]$. The proof of Lemma 1 readily extends to show that, if the DLIN assumption holds in $\hat{\mathbb{G}}$, the adversary cannot output a Type B signature with non-negligible chance. We find that $\Pr[S_3 \wedge \neg E_3] \leq \mathbf{Adv}^{\mathrm{DLIN}}_{\hat{\mathbb{G}}}(\lambda)$. In the following, we only need to determine an upper bound on $\Pr[S_3 \wedge E_3]$. To this end, we proceed using a sequence of $L$ games.

**Game 4:** This game is identical to Game 2 with a difference in the generation of $(Z, R, U)$ in each signing query. Instead of computing them as per (4), the challenger uses $\{\chi_i, \gamma_i, \delta_i\}_{i=1}^{4L+3}$ to compute $(Z, R, U)$ as a one-time linearly homomorphic signature on the vector (3). Clearly $(Z, R, U)$ retains the same distribution as in Game 3, so that the adversary's view remains unchanged. We have $\Pr[S_4 \wedge E_4] = \Pr[S_3 \wedge E_3]$.

**Game 5.$k$ ($1 \leq k \leq L$):** In Game 2.$k$, all signing queries are answered by returning Type B-$k$ signatures. The proof of Lemma 2 is immediately adapted to demonstrate that Game 5.1 is indistinguishable from Game 4 under the DLIN assumption in $\mathbb{G}$: concretely, we have $|\Pr[S_{5.1} \wedge E_{5.1}] - \Pr[S_4 \wedge E_4]| \leq 2 \cdot \mathbf{Adv}^{\mathrm{DLIN}}_{\mathbb{G}}(\lambda)$. The proof of Lemma 3 is also straightforward to adapt in order to prove the inequality $|\Pr[S_{5.k} \wedge E_{5.k}] - \Pr[S_{5.(k-1)} \wedge E_{5.(k-1)}]| \leq 2 \cdot \mathbf{Adv}^{\mathrm{DLIN}}_{\mathbb{G}}(\lambda)$. □

**Lemma 4.** *Game 2 is computationally indistinguishable from Game 1 under the DLIN assumption in $\hat{\mathbb{G}}$. In Game 2, the probability of event $F_2$ is at most $\Pr[F_2] \leq \mathbf{Adv}^{\mathrm{DLIN}}_{\mathcal{B}}(\lambda)$.*

*Proof.* The proof is straightforward. We show an algorithm $\mathcal{B}$ that inputs an instance $(\hat{g_z}, \hat{g_r}, \hat{h_z}, \hat{h_u})$ of the SDP problem (see Definition 2) which it solves with advantage $\Pr[F_2]$.

To generate the public key $PK$, $\mathcal{B}$ chooses $\{(\chi_i, \gamma_i, \delta_i)\}_{i=1}^{4L+3}$ and computes $pk_{hsps}$ as in step 3 of the key generation algorithm. It also runs steps 1, 2, 4, 5, 6 of the real key generation algorithm. Knowing $SK$, $\mathcal{B}$ can thus faithfully answer all signing queries made by the adversary $\mathcal{A}$.

By hypothesis, we know that $F_2$ occurs with non-negligible probability. In this case, $\mathcal{B}$ necessarily obtains two distinct one-time linearly homomorphic signatures $(Z_j, R_j, U_j)$ and $(Z^\star, R^\star, U^\star)$ on the vector (14). At this point, $\mathcal{B}$ immediately obtains a non-trivial solution $(Z_j/Z^\star, R_j/R^\star, U_j/U^\star)$ of its SDP instance, which also implies a DLIN distinguisher with advantage $\Pr[F_2]$. □

Under the SXDH assumption, we can also obtain a strongly unforgeable signature made of 3 group elements using the QA-NIZK proof system of Jutla and Roy [37]. The signature thus consists of a triple $(\sigma_1, \sigma_2, Z) = (u^\omega \cdot H(\boldsymbol{V}, M)^r, f^r, Z)$, where $Z$ is a QA-NIZK proof that $(\sigma_1, \sigma_2) = (u^\omega \cdot H(\boldsymbol{V}, M)^r, f^r)$, for some $r \in \mathbb{Z}_p$, where $M = H(\mathsf{Msg}, \sigma_2) \in \{0,1\}^L$.

In the security proof, the only modification is that the transition from Game 1 to Game 2 does no longer rely on a computational argument. Instead, it appeals to the uniqueness of proofs in [37] (i.e., for a given CRS, each statement has a unique proof).

We thus obtain a strongly unforgeable signature comprised of 3 elements of $\mathbb{G}$, which is as short as those of Boneh, Shen and Waters [16] with a much better concrete security. Indeed, it eliminates the $\Omega(q)$ degradation factor that [16] inherits from [53].

## G Proof of Tight Multi-User Security for the LHSPS Scheme of Section 2.2

In the multi-user setting, the security definition of linearly homomorphic structure-preserving signatures [42] can be generalized as follows.

**Definition 6.** *A LHSPS scheme is one-time secure in the multi-user setting if no PPT adversary has non-negligible advantage in the game below.*

1. *The adversary $\mathcal{A}$ chooses integers $\mu, n_1, \ldots, n_\mu \in \mathsf{poly}(\lambda)$. For each $i \in \{1, \ldots, \mu\}$, the challenger runs $(\mathsf{sk}_i, \mathsf{pk}_i) \leftarrow \mathsf{Keygen}(\lambda, n_i)$ and initializes a set $Q_i = \emptyset$. It gives $\{\mathsf{pk}_i\}_{i=1}^{\mu}$ to $\mathcal{A}$.*
2. *On polynomially occasions, the adversary $\mathcal{A}$ chooses an index $i \in \{1, \ldots, \mu\}$ and a $n_i$-vector $\boldsymbol{M} = (M_1, \ldots, M_{n_i}) \in \mathbb{G}^{n_i}$. The challenger returns $\sigma_i \leftarrow \mathsf{Sign}(\mathsf{sk}_i, \boldsymbol{M})$ and sets $Q_i = Q_i \cup \{\boldsymbol{M}\}$.*
3. *The adversary $\mathcal{A}$ outputs a triple $(i^\star, \sigma^\star, \boldsymbol{M}^\star)$, where $i^\star \in \{1, \ldots, \mu\}$, and wins if the following conditions are satisfied: (i) $\mathsf{Verify}(\mathsf{pk}_{i^\star}, \boldsymbol{M}^\star, \sigma^\star) = 1$; (ii) $Q_{i^\star}$ contains at most $n_{i^\star} - 1$ linearly independent vectors; (iii) $\boldsymbol{M}^\star$ is linearly independent of the vectors in $Q_{i^\star}$. The adversary's advantage is its probability of success taken over all random coins.*

In the above definition, we assume that the challenger can efficiently recognize when the adversary wins, by using the private key $\mathsf{sk}_{j^\star}$.

The next theorem shows that the one-time LHSPS scheme of Section 2.2 provides tight security under the DLIN assumption in the sense of Definition 6. The proof considers the case of symmetric bilinear groups (i.e., where $\mathbb{G} = \hat{\mathbb{G}}$) in order to be consistent with Section 4.

**Theorem 4.** *The scheme is unforgeable in the multi-user setting if the DLIN assumption holds in $\mathbb{G}$,. Concretely, the multi-user advantage of any adversary $\mathcal{A}$ is at most*

$$\mathbf{Adv}^{\text{m-ots}}(\mathcal{A}) \leq \mathbf{Adv}^{\text{DLIN}}_{\mathcal{B}}(\lambda) + \frac{1}{p},$$

*where $\mathcal{B}$ is a DLIN distinguisher running in time $t_{\mathcal{B}} \leq t_{\mathcal{A}} + \mathsf{poly}(\lambda, \mu, \max\{n_1, \ldots, n_\mu\})$.*

*Proof.* We describe an algorithm $\mathcal{B}$ that takes as input a DLIN instance $(g, f, h, f^a, h^b, T) \in \mathbb{G}^6$ and uses a multi-user forger $\mathcal{A}$ to decide if $T = g^{a+b}$ with nearly the same advantage as the forger's success probability. First, $\mathcal{B}$ constructs $\mu$ tuples $(g, f_j, h_j, f_j^{a_j}, h_j^{b_j}, T_j) \in \mathbb{G}^6$ such that $T_j = g^{a_j+b_j}$ (resp. $T_j \in_R \mathbb{G}$) for each $j \in \{1, \ldots, \mu\}$ if $T = g^{a+b}$ (resp. $T \in_R \mathbb{G}$). To this end, $\mathcal{B}$ picks $\alpha_j, \beta_j \xleftarrow{R} \mathbb{Z}_p$ and computes

$$(\tilde{f}_j, \tilde{h}_j, \tilde{f}_j^a, \tilde{h}_j^b) = \left(f^{\alpha_j}, h^{\beta_j}, (f^a)^{\alpha_j}, (h^b)^{\beta_j}\right), \qquad \forall j \in \{1, \ldots, \mu\} \ .$$

Next, $\mathcal{B}$ picks $\omega_j, \rho_j, \zeta_j \xleftarrow{R} \mathbb{Z}_p$ and computes

$$(g, f_j, h_j, f_j^{a_j}, h_j^{b_j}, T_j) = \left(g, \tilde{f}_j, \tilde{h}_j, (\tilde{f}_j^a)^{\omega_j} \cdot \tilde{f}_j^{\rho_j}, (\tilde{h}_j^b)^{\omega_j} \cdot \tilde{h}_j^{\zeta_j}, T^{\omega_j} \cdot g^{\rho_j+\zeta_j}\right), \qquad \forall j \in \{1, \ldots, \mu\} \ .$$

If $T = g^{a+b}$, we clearly have $T_j = g^{a_j+b_j}$ for each $j \in \{1, \ldots, \mu\}$. In contrast, if $T \in_R \mathbb{G}$, the values $\{T_j\}_{j=1}^{\mu}$ are uniformly random and independent.

To generate $\mu$ independent LHSPS key pairs $\{(\mathsf{sk}_j, \mathsf{pk}_j)\}_{j=1}^{\mu}$, $\mathcal{B}$ defines

$$g_{j,r} = f_j \,, \qquad g_{j,z} = f_j^{a_j} \,, \qquad h_{j,u} = h_j \,, \qquad h_{j,z} = h_j^{b_j} \,, \qquad \forall j \in \{1, \ldots, \mu\} \ .$$

Next, $\mathcal{B}$ picks $\chi_{j,i}, \gamma_{j,i}, \delta_{j,i} \xleftarrow{R} \mathbb{Z}_p$ for $j \in \{1, \ldots, \mu\}$ and $i \in \{1, \ldots, n_j\}$ and sets

$$g_{j,i} = g_{j,z}^{\chi_{j,i}} \cdot g_{j,r}^{\gamma_{j,i}} \,, \qquad h_{j,i} = h_{j,z}^{\chi_{j,i}} \cdot h_{j,u}^{\delta_{j,i}} \,, \qquad \forall j \in \{1, \ldots, \mu\},\ i \in \{1, \ldots, n_j\}$$

which defines the $j$-th public key to be $\mathsf{pk}_j = \left(g_{j,z},\ g_{j,r},\ h_{j,z},\ hj, u,\ \{(g_{j,i}, h_{j,i})\}_{i=1}^{n_j}\right)$, for which the private key is $\mathsf{sk}_j = \{(\chi_{j,i}, \gamma_{j,i}, \delta_{j,i})\}_{i=1}^{n_j}$.

The adversary is given $\{\mathsf{pk}_j\}_{j=1}^{\mu}$. For each public key $\mathsf{pk}_j$, the adversary is allowed to obtain up to $n_j - 1$ signatures on linearly independent vectors. Since $\mathcal{B}$ knows all private keys $\mathsf{sk}_j$, it can answer by faithfully running the signing algorithm. The game ends with the adversary $\mathcal{A}$ outputting and index $j^\star \in \{1, \ldots, \mu\}$ and a vector $\boldsymbol{M}^\star = (M_1^\star, \ldots, M_{n_{j^\star}}^\star) \in \mathbb{G}^{n_{j^\star}}$ with a valid signature $(z^\star, r^\star, u^\star)$ such that $\boldsymbol{M}^\star$ is linearly independent of the vectors for which $\mathcal{A}$ obtains signatures on behalf of $\mathsf{pk}_{j^\star}$. At this point, $\mathcal{B}$ uses $\mathsf{sk}_{j^\star}$ to compute its own signature

$$(z^\dagger, r^\dagger, u^\dagger) = (\textstyle\prod_{i=1}^{n_{j^\star}} M_i^{\star -\chi_{j^\star,i}}, \prod_{i=1}^{n_{j^\star}} M_i^{\star -\gamma_{j^\star,i}}, \prod_{i=1}^{n_{j^\star}} M_i^{\star -\delta_{j^\star,i}}) \tag{15}$$

on $(M_1^\star, \ldots, M_{n_j^\star}^\star)$. We claim that $(z^\ddagger, r^\ddagger, u^\ddagger) = (z^\star/z^\dagger, r^\star/r^\dagger, u^\star/u^\dagger)$ is such that $z^\ddagger \neq 1_{\mathbb{G}}$ with all but negligible probability.

To see this, we first note that $\mathsf{pk}_{j^\star}$ perfectly hides the vector $(\chi_{j^\star,1}, \ldots, \chi_{j^\star,n_{j^\star}})$. Moreover, for a given $\mathsf{pk}_{j^\star}$, each vector $(M_1, \ldots, M_{n_{j^\star}}) \in \mathbb{G}^{n_{j^\star}}$ has exponentially many valid signatures but the one produced by the signing algorithm is completely determined by $(\chi_{j^\star,1}, \ldots, \chi_{j^\star,n_{j^\star}})$. It is easy to see that, in $\mathcal{A}$'s view, guessing the $z^\dagger$ of (15) amounts to inferring which vector $(\chi_{j^\star,1}, \ldots, \chi_{j^\star,n_{j^\star}})$.

Throughout the game, $\mathcal{A}$ obtains signatures $\{(z_i, r_i, u_i)\}_{i=1}^{n_{j^\star}-1}$ on at most $n_{j^\star} - 1$ linearly independent vectors of $\mathbb{G}^{n_{j^\star}}$ on behalf of $\mathsf{sk}_{j^\star}$. These signatures only provide $\mathcal{A}$ with $n_{j^\star} - 1$ linearly independent equations because, for each triple $(z_i, r_i, u_i)$, $z_i$ uniquely determines $(r_i, u_i)$. In the public key $\mathsf{pk}_{j^\star}$, the group elements $\{(g_{j^\star,i}, h_{j^\star,i})\}_{i=1}^{n_{j^\star}}$ yield $2n_{j^\star}$ linear equations. An unbounded adversary is thus faced with $3n_{j^\star} - 1$ linear equations in $3n_{j^\star}$ unknowns. Since $\boldsymbol{M}^\star = (M_1^\star, \ldots, M_{n_{j^\star}}^\star)$ must be independent of the vectors that have been signed using $\mathsf{sk}_{j^\star}$, predicting $z^\dagger$ is only possible with probability $1/p$. With probability $1 - 1/p$, we thus have $z^\dagger \neq z^\star$, so that $(z^\ddagger, r^\ddagger, u^\ddagger)$ is a non-trivial tuple satisfying

$$e(g_{j^\star,z}, z^\ddagger) \cdot e(g_{j^\star,r}, r^\ddagger) = e(h_{j^\star,z}, z^\ddagger) \cdot e(h_{j^\star,u}, u^\ddagger) = 1_{\mathbb{G}_T} \ .$$

At this point, since $g_{j^\star,z} = g_{j^\star,r}^{a_{j^\star}}$ and $h_{j^\star,z} = h_{j^\star,u}^{b_{j^\star}}$ , we know that

$$T_{j^\star} = g^{a_{j^\star}+b_{j^\star}} \iff e(r^\ddagger \cdot u^\ddagger, g) \cdot e(T_{j^\star}, z^\ddagger) = 1_{\mathbb{G}_T} \ .$$

Hence, $\mathcal{B}$ returns 1 (meaning that $T = g^{a+b}$) if the equality $e(r^\ddagger \cdot u^\ddagger, g) \cdot e(T_{j^\star}, z^\ddagger) = 1_{\mathbb{G}_T}$ is satisfied and 0 otherwise. □

## H Proof of Theorem 2

### H.1 Definitions for Public-Key Encryption in the Multi-User Setting

Before giving the proof, we first recall the definition of chosen-ciphertext security in the multi-user setting [6].

In the multi-user setting [6], a public-key encryption scheme consists of algorithms (Par-Gen, Keygen, Encrypt, Decrypt), where Par-Gen takes as input a security parameter $\lambda$ and generates common public parameters $\Gamma$ shared by all users, Keygen takes as input $\Gamma$ and outputs a key pair $(SK, PK)$, and algorithms Encrypt and Decrypt that proceed in the usual way.

**Definition 7 ([6,32]).** *A public-key encryption scheme is $(\mu, q_e)$-IND-CCA secure, for integers $\mu, q_e \in \mathsf{poly}(\lambda)$, if no PPT adversary has noticeable advantage in this game:*

1. *The challenger generates $\Gamma \leftarrow \mathsf{Par\text{-}Gen}(\lambda)$ and runs $(SK^{(i)}, PK^{(i)}) \leftarrow \mathsf{Keygen}(\Gamma)$ for $i = 1$ to $\mu$. It gives $\{PK^{(i)}\}_{i=1}^{\mu}$ to the adversary $\mathcal{A}$ and retains $\{SK^{(i)}\}_{i=1}^{\mu}$. In addition, the challenger initializes a set $\mathcal{D} \leftarrow \emptyset$ and a counter $j_q \leftarrow 0$. Finally, it chooses a random bit $d \xleftarrow{R} \{0,1\}$.*
2. *The adversary $\mathcal{A}$ adaptively makes queries to the following oracles on multiple occasions:*
   - *Encryption query: $\mathcal{A}$ chooses an index $i \in \{1, \ldots, \mu\}$ and a pair $(M_0, M_1)$ of equal-length messages. If $j_q = q_e$, the oracle returns $\perp$. Otherwise, it computes $C \leftarrow \mathsf{Encrypt}(PK^{(i)}, M_d)$ and returns $C$. In addition, it sets $\mathcal{D} \leftarrow \mathcal{D} \cup \{(i, C)\}$ and $j_q \leftarrow j_q + 1$.*
   - *Decryption query: $\mathcal{A}$ can also invoke the decryption oracle on arbitrary ciphertexts $C$ and indexes $i \in \{1, \ldots, \mu\}$. If $(i, C) \in \mathcal{D}$, the oracle returns $\perp$. Otherwise, the oracle returns $M \leftarrow \mathsf{Decrypt}(SK^{(i)}, C)$, which may be $\perp$ if $C$ is an invalid ciphertext.*
3. *The adversary $\mathcal{A}$ outputs a bit $d'$ and is deemed successful if $d' = d$. As usual, $\mathcal{A}$'s advantage is measured as the distance $\mathbf{Adv}(\mathcal{A}) = |2 \cdot \Pr[d' = d] - 1|$.*

### H.2 Security Proof

The proof that the scheme of Section 4 provides $(1, q_e)$-IND-CCA security uses standard techniques [46,48] and proceeds as follows.

*Proof.* The proof uses of a sequence of games starting with a game where the challenger's hidden bit is $d = 0$ and ending with a game where $d = 1$. For each $i$, $S_i$ is the event that the challenger outputs 1 in Game $i$.

**Game 1:** This is the real attack game where the challenger's bit is $d = 0$. In details, the adversary is given the public key $PK$ while the challenger keeps the private key $SK$ to itself. At each decryption query, the challenger $\mathcal{B}$ faithfully runs the real decryption algorithm using the private key $SK = (x_1, y_1)$. At the $j$-th encryption query, for $j \in \{1, \ldots, q_e\}$, the adversary $\mathcal{A}$ chooses two distinct messages $M_0^{(j)}, M_1^{(j)} \in \mathbb{G}$ and obtains a ciphertext $C_j^\star = (\mathsf{VK}_j^\star, C_{j,0}^\star, C_{j,1}^\star, C_{j,2}^\star, \pi_j^\star, sig_j^\star)$ which is an encryption of $M_0^{(j)}$ under $PK$. Of course, $\mathcal{A}$ is disallowed to invoke the decryption oracle on any ciphertext produced by the encryption oracle. Eventually, $\mathcal{A}$ halts and outputs a bit $d' \in \{0,1\}$. We denote by $S_1$ the event that $d' = 0$, which causes the challenger to output 1.

**Game 2:** In this game, we add a failure event $E_1$ which causes the challenger to halt and output 0 if it occurs. When $\mathcal{A}$ invokes the decryption oracle on a ciphertext $C = (\mathsf{VK}, C_0, C_1, C_2, \pi, sig)$, we define $E_1$ to be the event that $\mathsf{VK}$ is a recycled one-time verification key that appeared in an output of the encryption oracle. Clearly, $E_1$ can only occur with negligible probability if the one-time signature is strongly unforgeable in the multi-key setting[8] (as defined in [32, Definition 4]). We have $|\Pr[S_2] - \Pr[S_1]| \leq \Pr[E_1] \leq \mathbf{Adv}^{n\text{-suf-ots}}(\lambda)$.

**Game 3:** This game is like Game 2 except that, at each encryption query $(M_0^{(j)}, M_1^{(j)})$, the returned ciphertext $C_j^\star = (\mathsf{VK}_j^\star, C_{j,0}^\star, C_{j,1}^\star, C_{j,2}^\star, \pi_j^\star, sig_j^\star)$ is obtained by computing $\pi_j^\star$ as a simulated proof using the trapdoor $\omega_1, \omega_2$. This is achieved by computing $(\sigma_1, \sigma_2, \sigma_3, Z, R, U)$ as a real signature (i.e., where $\sigma_1 = g^{\omega_1+\omega_2} \cdot H(\boldsymbol{V}, \mathsf{VK})^r \cdot H(\boldsymbol{W}, \mathsf{VK})^s$, $\sigma_2 = f^r$ and $\sigma_3 = h^s$) via the signing algorithm of Section 3, setting $b = 0$ at Step 4 of the encryption algorithm and choosing the vector (7) (whose last 2 coordinates now contain $(\Omega_1, \Omega_2)$ instead of $(1_{\mathbb{G}}, 1_{\mathbb{G}})$) in the entire span of the rows of $\mathbf{M}$. Note that, in this case, the commitments $\{\boldsymbol{C}_{W_i}\}_{i=1}^2$ and the proofs $\boldsymbol{\pi}_2, \boldsymbol{\pi}_2$ can be generated without using the encryption exponents $(\theta_1, \theta_2)$ in steps 3 and 7 of the encryption algorithm. Indeed, since $b = 0$, we have $\Gamma_g = 1_{\mathbb{G}}$ and the witnesses $W_1 = W_2 = 1_{\mathbb{G}}$ can be used to prove relations (9). Thanks to the perfect witness indistinguishability of Groth-Sahai proofs for the CRS $(\boldsymbol{G}_1, \boldsymbol{G}_2, \boldsymbol{G}_3)$, the proofs $\{\pi_j^\star\}_{j=1}^{q_e}$ have exactly the same distribution as in Game 2 and $\mathcal{A}$'s view will not be affected by this change. We have $\Pr[S_3] = \Pr[S_2]$.

**Game 4:** In this game, we modify the distribution of the public key. In step 7 of the key generation algorithm, we choose $\boldsymbol{G}_3 = \boldsymbol{G}_1^{\xi_1} \cdot \boldsymbol{G}_2^{\xi_2}$, with $\xi_1, \xi_2 \stackrel{R}{\leftarrow} \mathbb{Z}_p$, instead of choosing $\boldsymbol{G}_3 \stackrel{R}{\leftarrow} \mathbb{G}^3$ uniformly. Under the DLIN assumption, this change should not significantly affect $\mathcal{A}$'s behavior and we have $|\Pr[S_4] - \Pr[S_3]| \leq \mathbf{Adv}^{\mathrm{DLIN}}(\lambda)$. Note that $(\boldsymbol{G}_1, \boldsymbol{G}_2, \boldsymbol{G_3})$ now forms a perfectly sound CRS.

**Game 5:** We modify the decryption oracle. When the adversary $\mathcal{A}$ queries the decryption of a ciphertext $C = (\mathsf{VK}, C_0, C_1, C_2, \pi, sig)$, instead of using the private key $SK = (x_1, y_1)$ to compute $M = C_0 \cdot C_1^{-1/x_1} \cdot C_2^{-1/x_2}$, $\mathcal{B}$ uses the extraction trapdoor $(\beta_1, \beta_2) = (\log_G(G_1), \log_G(G_2))$ of the Groth-Sahai CRS $(\boldsymbol{G}_1, \boldsymbol{G}_2, \boldsymbol{G_3})$ to extract the witnesses $(W_1, W_2)$ from the commitments $\{\boldsymbol{C}_{W_i}\}_{i=1}^2$ contained in $\pi$ and return $M = C_0 \cdot W_1^{-1} \cdot W_2^{-1}$. Since $(\boldsymbol{G}_1, \boldsymbol{G}_2, \boldsymbol{G_3})$ is a perfectly sound Groth-Sahai CRS, the proof $\pi$ (10) guarantees that either: (i) $(W_1, W_2) = (g^{\theta_1}, g^{\theta_2})$, where $(\theta_1, \theta_2) = (\log_{f_1}(C_1), \log_{h_1}(C_2))$, in which case the decryption oracle gives the same answer as in Game 4; (ii) $\pi$ contains a commitment $\boldsymbol{C}_b$ to $b = 0$, which means that $\boldsymbol{C}_{\sigma_1}$ and $\boldsymbol{C}_Z, \boldsymbol{C}_R, \boldsymbol{C}_U$ are extractable commitments to $(\sigma_1, Z, R, U)$ such that $(\sigma_1, \sigma_2, \sigma_3, Z, R, U)$ is a valid signature on the one-time verification key $\mathsf{VK}$. Since $\mathsf{VK} \notin \{\mathsf{VK}_j^\star\}_{j=1}^{q_e}$ unless the event $E_1$ introduced in Game 2 occurs, it follows that situation (ii) would contradict the security of the signature scheme in Section 3. We thus have $|\Pr[S_5] - \Pr[S_4]| \leq (2 \cdot L + 1) \cdot \mathbf{Adv}_{\mathcal{B}}^{\mathrm{DLIN}}(\lambda) + \frac{2}{p}$.

**Game 6:** We modify the treatment of encryption queries $\{(M_0^{(j)}, M_1^{(j)})\}_{j=1}^{q_e}$. In this game, when $\mathcal{B}$ computes $C_j^\star = (\mathsf{VK}_j^\star, C_{j,0}^\star, C_{j,1}^\star, C_{j,2}^\star, \pi_j^\star, sig_j^\star)$, it computes $(C_{j,0}^\star, C_{j,1}^\star, C_{j,2}^\star)$ as a BBS encryption of $M_1^{(j)}$ rather than $M_0^{(j)}$. It is easy to prove that any PPT adversary $\mathcal{A}$ having noticeably different behaviors in Game 6 and Game 5 would imply an adversary against the semantic security of the BBS cryptosystem in the multi-challenge setting, which would contradict the DLIN assumption. Indeed, Hofheinz and Jager proved [32, Theorem 6] that the multi-challenge semantic security of BBS is tightly related to the DLIN assumption. The result of [32, Theorem 6] implies the inequality $|\Pr[S_6] - \Pr[S_5]| \leq \mathbf{Adv}^{\mathrm{DLIN}}(\lambda) + 1/p$.

**Game 7:** In this game, we modify again the decryption oracle. This time, instead of using the extraction trapdoor $(\beta_1, \beta_2) = (\log_G(G_1), \log_G(G_2))$ of the Groth-Sahai CRS $(\boldsymbol{G}_1, \boldsymbol{G}_2, \boldsymbol{G}_3)$ to recover

[8] This notion (see Definition 4 in [32]) refers to a game where the adversary is given $\mu$ verification keys $\{\mathsf{VK}_i\}_{i=1}^{\mu}$ and an oracle that returns exactly one signature for each key. The adversary's tasks is to output a triple $(i^\star, M^\star, \sigma^\star)$, where $i^\star \in \{1, \ldots, \mu\}$ and $(M^\star, \sigma^\star)$ was not produced by the signing oracle for $\mathsf{VK}_{i^\star}$. Hofheinz and Jager [32, Section 4.2] described a discrete-log-based one-time signature with tight security in the multi-key setting.

the plaintext $M = C_0 \cdot W_1^{-1} \cdot W_2^{-1}$ at each valid decryption query $C = (\mathsf{VK}, C_0, C_1, C_2, \pi, sig)$, the challenger $\mathcal{B}$ uses the private key $SK = (x_1, y_1)$ to compute $M = C_0 \cdot C_1^{-1/x_1} \cdot C_2^{-1/y_1}$. It is easy to see that $\mathcal{A}$'s view will be the same as in Game 6 until $\mathcal{A}$ manages to query the decryption oracle on a valid-looking ciphertext $C = (\mathsf{VK}, C_0, C_1, C_2, \pi, sig)$ for which $\pi$ contains a commitment $\boldsymbol{C}_b$ to $b = 0$. The same arguments as in Game 5 show that the latter event would contradict the security of the signature scheme in Section 3. We have $|\Pr[S_7] - \Pr[S_6]| \leq (2 \cdot L + 1) \cdot \mathbf{Adv}_{\mathcal{B}}^{\mathrm{DLIN}}(\lambda) + \frac{2}{p}$.

**Game 8:** In this game, we modify again the generation of the public key. We restore the Groth-Sahai CRS $(\boldsymbol{G}_1, \boldsymbol{G}_2, \boldsymbol{G}_3)$ back to its original distribution and choose $\boldsymbol{G}_3 \xleftarrow{R} \mathbb{G}^3$ as a uniformly random vector, so that $(\boldsymbol{G}_1, \boldsymbol{G}_2, \boldsymbol{G}_3)$ is configured for the perfect NIWI setting. As in the transition from Game 3 to Game 4, a straightforward argument shows that $|\Pr[S_8] - \Pr[S_7]| \leq \mathbf{Adv}^{\mathrm{DLIN}}(\lambda)$.

**Game 9:** We bring one last change to the generation of the challenge ciphertexts $\{C_j^\star\}_{j=1}^{q_e}$. For each ciphertext $C_j^\star$ generated by the encryption oracle, instead of computing $\pi_j^\star$ using the simulation trapdoor $(\omega_1, \omega_2)$, we compute it using the real witnesses $(\theta_{j,1}, \theta_{j,2}) \in \mathbb{Z}_p^2$ and thus set $b = 1$ in step 4 of the encryption algorithm. This change is only conceptual since, due to the perfect witness indistinguishability of Groth-Sahai proofs on a perfectly hiding CRS, the obtained proofs $\pi_j^\star$ have the same distribution as in Game 8. We have $\Pr[S_9] = \Pr[S_8]$.

**Game 10:** This game is like Game 9 with the difference that the challenger does no longer output 0 in the event that $\mathcal{A}$ queries the decryption of a ciphertext $C = (\mathsf{VK}, C_0, C_1, C_2, \pi, sig)$ such that $\mathsf{VK}$ appeared in an output of the encryption oracle. The multi-key unforgeability of the one-time signature ensures that this change should not make a difference and we have the inequality $|\Pr[S_{10}] - \Pr[S_9]| \leq \mathbf{Adv}^{n\text{-suf-ots}}(\lambda)$.

We observe that Game 10 corresponds to the actual game where the challenger's bit is $d = 1$. If we combine the above, we thus find the announced upper bound for the distance $|\Pr[S_1] - \Pr[S_{10}]|$. □