Efficient Completely Context-Hiding Quotable and Linearly Homomorphic Signatures

Nuttapong Attrapadung[1] ⋆, Benoît Libert[2] ⋆⋆, and Thomas Peters[2] ⋆⋆⋆

[1] Research Center for Information Security, AIST (Japan)
[2]Technicolor (France)
[3]Université catholique de Louvain, ICTEAM Institute (Belgium)

Abstract. Homomorphic signatures are primitives that allow for public computations for a class of specified predicates over authenticated data. An enhanced privacy notion, called complete context-hiding security, was recently motivated by Attrapadung *et al.* (Asiacrypt'12). This notion ensures that a signature derived from any valid signatures is perfectly indistinguishable from a newly generated signatures (on the same message), and seems desirable in many applications requiring to compute on authenticated data. In this paper, we focus on two useful predicates – namely, substring quotation predicates and linear dependency predicates – and present the first completely context-hiding schemes for these in the standard model. Moreover, our new quotable signature scheme is the first such construction with signatures of linear size. In comparison with the initial scheme of Ahn *et al.* (TCC 2012), we thus reduce the signature size from $O(n \log n)$ to $O(n)$, where n is the message size. Our scheme also allows signing messages of arbitrary length using constant-size public keys.

Keywords. Homomorphic signatures, provable security, privacy, unlinkability, standard model.

1 Introduction

The recent years, much attention has been paid to homomorphic cryptographic primitives, which make it possible to publicly compute over encrypted [24, 36, 25] or signed [32, 10, 12] datasets.

In the latter case, anyone holding signatures $\{\sigma_i = \mathsf{Sign}(\mathsf{sk}, m_i)\}_{i=1}^k$ on messages $\{m_i\}_{i=1}^k$ can publicly derive pairs $(m, \sigma) = \mathsf{Evaluate}(\mathsf{pk}, \{(m_i, \sigma_i)\}_{i=1}^k, f)$ such that $\mathsf{Verify}(\mathsf{pk}, m, \sigma) = 1$, where $m = f(m_1, \ldots, m_k)$ for certain functions f. This has been possible for arithmetic functions [10, 22, 11, 12], logical predicates [35, 27, 14, 15, 13] and other kinds of algebraic signatures [34, 8, 28, 30]. In the case of arithmetic manipulations, homomorphic signatures notably allow untrusted remote parties (e.g. storage servers in cloud computing services) to authenticate their calculations on the clients' data. They also proved useful to prevent pollution attacks in network coding [10, 3, 22].

At TCC 2012, Ahn *et al.* [4] defined the general notion of P-homomorphic signature – for a predicate P – that captures all the aforementioned forms of homomorphic signatures. Specifically, it allows anybody who sees a signature on a message m to publicly obtain signatures on messages m' such that $P(m, m') = 1$. Informally, a P-homomorphic signature is said unforgeable when a signature on m only makes it possible to publicly derive signatures on messages m' such that $P(m, m') = 1$. Ahn *et al.* also formalized a strong privacy property, called *strong context hiding*, which mandates that original and derived signatures be unconditionally unlinkable.

Quite recently, Attrapadung, Libert and Peters [6] suggested even stronger privacy notions,

⋆ This author is supported by KAKENHI (Grant-in-Aid for Young Scientists B) No. 22700020. This work was done while the author visited ENS Paris.

⋆⋆ Part of this work was done while this author was a F.R.S.-F.N.R.S. scientific collaborator at the Université catholique de Louvain (Belgium).

⋆⋆⋆ Supported by the IUAP B-Crypt Project and the Walloon Region Camus Project.

of which the strongest one is termed *complete context-hiding* security. The difference between the definition of Ahn *et al.* [4] and the one of [6] lies in that the former requires the unlinkability of derived signatures to only *honestly generated* signatures. In contrast, the stronger *complete* context hiding property [6] requires unlinkability with respect to *any valid* signatures, including those signatures that might have been somehow maliciously re-randomized by the adversary. Not achieving this kind of security may raise some concerns in certain applications such as collusion attacks in network coding, as motivated in [6].

So far, in the standard model, complete context-hiding security has been achieved for only one specific kind of predicates, namely subset predicates [6]. For other predicates, completely context-hiding constructions are currently lacking. In particular, this is true for substring quotations – which were addressed by the main construction of [4] – and linear homomorphisms, that have been extensively studied in recent years [10, 22, 11, 5, 16, 17, 19]. This paper aims at filling these gaps by proposing the first completely context-hiding schemes for these predicates. Along the way, we also improve upon the best previously achieved efficiency for quoting predicates.

1.1 Related Work

Homomorphic signatures were first suggested by Desmedt [20] and further studied by Johnson, Molnar, Song and Wagner [32]. Later on, they were considered by Boneh, Freeman, Katz and Waters [10] who used them to sign linear sub-spaces so as to thwart pollution attacks in network coding. In the random oracle model, Boneh *et al.* [10] described a pairing-based scheme with short per-vector signatures. In a follow-up work, Gennaro, Katz, Krawczyk and Rabin [22] gave an RSA-based linearly homomorphic system [22] over the integers in the random oracle model. Boneh and Freeman [11] suggested to work over binary fields using lattices. They also motivated a notion, termed *weak privacy*, which requires derived signatures not to leak the original dataset they were derived from.

Constructions in the standard model came out in two independent papers by Attrapadung and Libert [5] and Catalano, Fiore and Warinschi [16, 17]. The construction of [5] was extended by Freeman [19] who defined a framework for the design of linearly homomorphic signatures satisfying a stronger definition of unforgeability. The latter framework of [19] was notably instantiated under standard assumptions like RSA, Diffie-Hellman and, more efficiently, the Strong Diffie-Hellman assumption. In the random oracle model, Boneh and Freeman [12] designed lattice-based homomorphic signatures for multivariate polynomial functions. Except [10, 5], all the aforementioned constructions are only weakly context-hiding in the sense of [11].

Strongly context-hiding P-homomorphic signatures were recently given by Ahn *et al.* [4] for both quoting and subset predicates. In [4], linearly homomorphic signatures [10, 11, 16, 19] were also shown to imply P-homomorphic signatures allowing for the computation of weighted averages and Fourier transforms. It was pinpointed in [4] that the Boneh *et al.* [10] system is strongly context-hiding thanks to the uniqueness of its signatures (in the random oracle model).

In the standard model, the construction of Attrapadung and Libert [5] can be proved strongly context hiding as well (unlike the schemes of [16, 17, 19]) but, as discussed in [6], it is demonstrably not completely context-hiding. Attrapadung *et al.* [6] came close to filling this gap by describing a more efficient strongly context-hiding realization simultaneously satisfying another privacy notion which had been elusive so far. Still, their use of the dual system technique [38, 23] prevented them from reaching the desired complete context-hiding level. In the standard model, no completely context-hiding linearly homomorphic signature has ever been reported to date.

1.2 Our Contributions

Linear-Size Homomorphic Signatures for Quoting Substring. Given a signature on a message m, quotable signatures allow for the public derivation of signatures on any substring of m. Ahn *et al.* [4] gave a system where signatures have quasi-linear size: for a message consisting of n symbols, each signature contains $O(n \log n)$ group elements[1]. Their construction is known to be only strongly context-hiding (in the sense of [4]) and selectively unforgeable in the random oracle model. It was argued that their scheme can be modified so as to be proved fully unforgeable in the standard model using the dual system encryption technique of Waters [38] (or, more precisely, its signature analogue [23]). The latter inherently involves two distinct distributions of signatures satisfying the verification algorithm. The very existence of an alternative distribution of valid signatures implies that the resulting system can hardly be completely context-hiding.

The first contribution of this paper is a quotable signature scheme whose design principle is very different from [4]. The new scheme is proved fully unforgeable in the standard model and also turns out to be the first completely context-hiding quotable signature. Moreover, it improves upon the worst-case efficiency of [4] in that a n-symbol message can be signed using $O(n)$ group elements.

Our construction builds on the structure-preserving signature of Abe, Haralambiev and Ohkubo [1], which is used to sign individual message symbols. An important property of the structure-preserving signature in [1] is that certain signature components can serve as a commitment to the message. Our quotable signature exploits this property to link signatures on individual symbols: each symbol is signed with a commitment to the next symbol. Quotable signatures are then obtained as a sequence of perfectly hiding commitments to these underlying signatures and non-interactive randomizable arguments of their validity.

Beyond its asymptotically shorter signatures, our scheme also allows signing messages of arbitrary length using a constant-size public key. In contrast, [4] requires the key generation algorithm to define a logarithmic bound on the maximal number of symbols in messages to be signed.

Completely Context-Hiding Linearly Homomorphic Signatures. We provide the first completely context-hiding linearly homomorphic signature in the standard model. So far, the random-oracle-based construction of Boneh *et al.* [10] was the only linearly homomorphic signatures satisfying that level of privacy. The scheme of [5] is strongly context-hiding in the standard model but, as pointed out in [6], it falls short of the enhanced privacy level advocated by [6].

To bypass the latter limitation – which seems inherent to all signature schemes [5, 23] based on the dual system technique – we take further advantage of the malleability properties [7, 21] of Groth-Sahai proofs [26] and build on a linearly homomorphic signature proposed by Attrapadung *et al.* [6]. The latter scheme is only weakly context-hiding (*i.e.*, the original message remains hidden as long as the original signature is not given) as its signatures contain components that cannot be randomized at each derivation and thus carry information about the original signatures. Our idea is to replace these signature components by perfectly hiding commitments to these values. The commitments are accompanied with non-interactive (randomizable) witness indistinguishable arguments that committed values satisfy appropriate algebraic relations.

One difficulty to solve is that, in the underlying weakly context-hiding construction [6], the "problematic" signature components are actually exponents that the reduction has to compute in the security proof. When Groth-Sahai proofs are used in their extractable mode, committed expo-

[1] In the signature derivation algorithm of [4], two kinds of signatures can be produced. Apart from Type I signatures, which are distributed as original signatures, Type II signatures have $O(\log n)$-size signatures but cannot be quoted any further.

nents cannot be fully extracted from their commitments. To solve this problem, we need to modify the weakly context-hiding scheme of [6] in such a way that its signatures only consist of group elements. We were able to do this at the expense of relying on a slightly stronger assumption in the security proof: instead of the standard Diffie-Hellman assumption, the unforgeability now relies on the Flexible Diffie-Hellman assumption [31], which is still a simple assumption.

2 Background

2.1 Definitions for Homomorphic Signatures

Definition 1 ([4]). *Let $\mathcal{M}$ be a message space and $2^{\mathcal{M}}$ be its powerset. Let $P : 2^{\mathcal{M}} \times \mathcal{M} \to \{0,1\}$ be a predicate. A message m' is said* **derivable** *from $M \subset \mathcal{M}$ if $P(M, m') = 1$. As in [4], $P^i(M)$ is the set of messages derivable from $P^{i-1}(M)$, where $P^0(M) := \{m' \in \mathcal{M} \mid P(M, m') = 1\}$. Finally, $P^*(M) := \cup_{i=0}^{\infty} P^i(M)$ denotes the set of messages derivable from M by iterated derivation.*

Definition 2 ([4]). *A P-homomorphic signature for a predicate $P : 2^{\mathcal{M}} \times \mathcal{M} \to \{0,1\}$ is a triple of algorithms* $(\mathsf{Keygen}, \mathsf{SignDerive}, \mathsf{Verify})$ *with the following properties.*

$\mathsf{Keygen}(\lambda)$: *takes as input a security parameter $\lambda \in \mathbb{N}$ and outputs a key pair $(\mathsf{sk}, \mathsf{pk})$. As in [4], the private key sk is seen as a signature on the empty tuple $\varepsilon \in \mathcal{M}$.*

$\mathsf{SignDerive}\big(\mathsf{pk}, (\{\sigma_m\}_{m\in M}, M), m'\big)$: *is a possibly randomized algorithm that takes as input a public key pk, a set of messages $M \subset \mathcal{M}$, a corresponding set of signatures $\{\sigma_m\}_{m\in M}$ and a derived message $m' \in \mathcal{M}$. If $P(M, m') = 0$, it returns $\perp$. Otherwise, it outputs a derived signature σ'*

$\mathsf{Verify}(\mathsf{pk}, \sigma, m)$: *is a deterministic algorithm that takes as input a public key pk, a signature σ and a message m. It outputs 0 or 1.*

Note that the empty tuple $\varepsilon \in \mathcal{M}$ satisfies $P(\varepsilon, m) = 1$ for each message $m \in \mathcal{M}$. Similarly to Ahn *et al.* [4], we define the algorithm $\mathsf{Sign}(\mathsf{pk}, \mathsf{sk}, m)$ that runs[2] $\mathsf{SignDerive}(\mathsf{pk}, (\mathsf{sk}, \varepsilon), m)$ and returns the resulting output. Also, for any message set $M = \{m_1, \ldots, m_k\} \subset \mathcal{M}$, we define $\mathsf{Sign}(\mathsf{sk}, M) := \{\mathsf{Sign}(\mathsf{sk}, m_1), \ldots, \mathsf{Sign}(\mathsf{sk}, m_k)\}$. Also, we write $\mathsf{Verify}(\mathsf{pk}, M, \{\sigma_m\}_{m\in M}) = 1$ to express that $\mathsf{Verify}(\mathsf{pk}, m, \sigma_m) = 1$ for each $m \in M$.

Correctness. For all key pairs $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Keygen}(\lambda)$, for any message set $M \subset \mathcal{M}$, any message $m' \in \mathcal{M}$ such that $P(M, m') = 1$, we must have: (i) $\mathsf{SignDerive}(\mathsf{pk}, (\mathsf{Sign}(\mathsf{sk}, M), M), m') \neq \perp$; (ii) $\mathsf{Verify}\big(\mathsf{pk}, m', \mathsf{SignDerive}(\mathsf{pk}, (\mathsf{Sign}(\mathsf{sk}, M), M), m')\big) = 1$.

Definition 3 ([4]). *A P-homomorphic signature* $(\mathsf{Keygen}, \mathsf{SignDerive}, \mathsf{Verify})$ *is* **unforgeable** *if no probabilistic polynomial-time (PPT) adversary has non-negligible advantage in this game:*

1. *The challenger generates $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Keygen}(\lambda)$ and gives pk to the adversary $\mathcal{A}$. It initializes two initially empty tables T and Q.*
2. *$\mathcal{A}$ adaptively interleaves the following queries.*
   - *Signing queries: $\mathcal{A}$ chooses a message $m \in \mathcal{M}$. The challenger replies by choosing a handle h, runs $\sigma \leftarrow \mathsf{Sign}(sk, m)$ and stores (h, m, σ) in a table T. The handle h is returned to $\mathcal{A}$.*

[2] The intuition is that any message can be derived when the original message contains the signing key.

- *Derivation queries:* $\mathcal{A}$ *chooses a vector of handles* $\vec{\mathsf{h}} = (\mathsf{h}_1, \ldots, \mathsf{h}_k)$ *and a message* $m' \in \mathcal{M}$. *The challenger retrieves the tuples* $\{(\mathsf{h}_i, m_i, \sigma_i)\}_{i=1}^k$ *from* T *and returns* $\perp$ *if one of these does not exist. Otherwise, it defines* $M := (m_1, \ldots, m_k)$ *and* $\{\sigma_m\}_{m \in M} = \{\sigma_1, \ldots, \sigma_k\}$. *If* $P(M, m') = 1$, *the challenger runs* $\sigma' \leftarrow \mathsf{SignDerive}\big(\mathsf{pk}, (\{\sigma_m\}_{m \in M}, M), m'\big)$, *chooses a handle* h', *stores* $(\mathsf{h}', m', \sigma')$ *in* T *and returns* h' *to* $\mathcal{A}$.
- *Reveal queries:* $\mathcal{A}$ *chooses a handle* h. *If no tuple of the form* $(\mathsf{h}, m', \sigma')$ *exists in* T, *the challenger returns* $\perp$. *Otherwise, it returns* σ' *to* $\mathcal{A}$ *and adds* (m', σ') *to the set* Q.

3. $\mathcal{A}$ *outputs a pair* (σ', m') *and wins if: (i)* $\mathsf{Verify}(\mathsf{pk}, m', \sigma') = 1$; *(ii) If* $M \subset \mathcal{M}$ *is the set of messages in* Q, *then* $m' \notin P^*(M)$.

Ahn *et al.* [4] formalized a strong notion of privacy which is recalled in Appendix A.1. It captures the inability of distinguishing derived signatures from original ones, *even* when these are given along with the private key. In [4], it was showed that, if a scheme is strongly context hiding, then Definition 3 can be simplified by only providing the adversary with an ordinary signing oracle.

As noted in [6], specific applications may require an even stronger definition. In particular, the following definition makes sense when homomorphic signature schemes are randomizable and/or the verification algorithm accepts several distributions of valid-looking signatures.

Definition 4 ([6]). *A homomorphic signature* $(\mathsf{Keygen}, \mathsf{Sign}, \mathsf{SignDerive}, \mathsf{Verify})$ *is* **completely context hiding** *for the predicate* P *if, for all key pairs* $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Keygen}(\lambda)$, *for all message sets* $M \subset \mathcal{M}^*$ *and* $m' \in \mathcal{M}$ *such that* $P(M, m') = 1$, *for all* $\{\sigma_m\}_{m \in M}$ *such that* $\mathsf{Verify}(\mathsf{pk}, M, \{\sigma_m\}_{m \in M}) = 1$, *the following distributions are statistically close*

$$\big\{(\mathsf{sk},\ \{\sigma_m\}_{m \in M},\ \mathsf{Sign}(sk, m'))\big\}_{sk, M, m'},$$
$$\big\{\big(\mathsf{sk},\ \{\sigma_m\}_{m \in M},\ \mathsf{SignDerive}\big(\mathsf{pk}, (\{\sigma_m\}_{m \in M}, M), m'\big)\big)\big\}_{sk, M, m'}.$$

2.2 Hardness Assumptions

We consider bilinear maps $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$ over groups of prime order p. In these groups, we rely on the following hardness assumptions.

Definition 5 ([9]). *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** *is the intractability of DLIN for any PPT distinguisher* $\mathcal{D}$.

We also use a weaker variant of an assumption used in [31, 33]. The latter is a variant of the Diffie-Hellman assumption, which posits the infeasibility of finding a pair $(g^\mu, g^{ab \cdot \mu})$ given $(g, g^a, g^b) \in \mathbb{G}^3$.

Definition 6. *The* **Flexible Diffie-Hellman Problem** *(FlexDH) in* $\mathbb{G}$, *is given* (g, g^a, g^b), *with* $a, b \xleftarrow{R} \mathbb{Z}_p$, *to find a triple* $(g^\mu, g^{a \cdot \mu}, g^{ab \cdot \mu}) \in \mathbb{G}^3$ *such that* $\mu \neq 0$.

The FlexDH assumption is known to imply the intractability of distinguishing g^{abc} from random given (g, g^a, g^b, g^c). For this reason, it can be seen as a *simple* assumption.

Finally, we also use a signature scheme based on the following q-type assumption.

Definition 7 ([1]). *In a group* $\mathbb{G}$, *the* q**-Simultaneous Flexible Pairing Problem** *(q-SFP) is, given* $\big(g_z, h_z, g_r, h_r, a, \tilde{a}, b, \tilde{b}\big) \in \mathbb{G}^8$ *and* q *tuples* $(z_j, r_j, s_j, t_j, u_j, v_j, w_j) \in \mathbb{G}^7$ *such that*

$$e(a, \tilde{a}) = e(g_z, z_j) \cdot e(g_r, r_j) \cdot e(s_j, t_j) \quad \text{and} \quad e(b, \tilde{b}) = e(h_z, z_j) \cdot e(h_r, u_j) \cdot e(v_j, w_j), \tag{1}$$

to find a new tuple $(z^\star, r^\star, s^\star, t^\star, u^\star, v^\star, w^\star) \in \mathbb{G}^7$ *satisfying (1) and such that* $z^\star \notin \{1_\mathbb{G}, z_1, \ldots, z_q\}$.

2.3 Structure-Preserving Signatures

Many protocols require to sign elements of bilinear groups while preserving their structure and, in particular, without hashing them. Abe, Haralambiev and Ohkubo [1, 2] (AHO) described such a signature. The description below assumes common public parameters $\mathsf{pp} = \big((\mathbb{G}, \mathbb{G}_T),\ g\big)$ consisting of symmetric bilinear groups $(\mathbb{G}, \mathbb{G}_T)$ of prime order $p > 2^\lambda$, where $\lambda \in \mathbb{N}$ and a generator $g \in \mathbb{G}$.

Keygen(pp, n)**:** given an upper bound $n \in \mathbb{N}$ on the number of group elements per message to be signed, choose generators $G_r, H_r \xleftarrow{R} \mathbb{G}$. Pick $\gamma_z, \delta_z \xleftarrow{R} \mathbb{Z}_p$ and $\gamma_i, \delta_i \xleftarrow{R} \mathbb{Z}_p$, for $i = 1$ to n. Then, compute $G_z = G_r^{\gamma_z}$, $H_z = H_r^{\delta_z}$ and $G_i = G_r^{\gamma_i}$, $H_i = H_r^{\delta_i}$ for each $i \in \{1, \ldots, n\}$. Finally, choose $\alpha_a, \alpha_b \xleftarrow{R} \mathbb{Z}_p$ and define $A = e(G_r, g^{\alpha_a})$ and $B = e(H_r, g^{\alpha_b})$. The public key is defined to be

$$pk = \big(G_r,\ H_r,\ G_z,\ H_z,\ \{G_i, H_i\}_{i=1}^n,\ A,\ B\big) \in \mathbb{G}^{2n+4} \times \mathbb{G}_T^2$$

while the private key is $sk = \big(\alpha_a, \alpha_b, \gamma_z, \delta_z, \{\gamma_i, \delta_i\}_{i=1}^n\big)$.

Sign$(sk, (M_1, \ldots, M_n))$**:** to sign a vector $(M_1, \ldots, M_n) \in \mathbb{G}^n$ using $sk = (\alpha_a, \alpha_b, \gamma_z, \delta_z, \{\gamma_i, \delta_i\}_{i=1}^n)$, choose $\zeta, \rho_a, \rho_b, \omega_a, \omega_b \xleftarrow{R} \mathbb{Z}_p$ and compute $\theta_1 = g^\zeta$ as well as

$$\theta_2 = g^{\rho_a - \gamma_z \zeta} \cdot \prod_{i=1}^n M_i^{-\gamma_i}, \qquad \theta_3 = G_r^{\omega_a}, \qquad \theta_4 = g^{(\alpha_a - \rho_a)/\omega_a},$$
$$\theta_5 = g^{\rho_b - \delta_z \zeta} \cdot \prod_{i=1}^n M_i^{-\delta_i}, \qquad \theta_6 = H_r^{\omega_b}, \qquad \theta_7 = g^{(\alpha_b - \rho_b)/\omega_b},$$

The signature consists of $\sigma = (\theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6, \theta_7) \in \mathbb{G}^7$.

Verify$(pk, \sigma, (M_1, \ldots, M_n))$**:** given a signature $\sigma = (\theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6, \theta_7) \in \mathbb{G}^7$, return 1 if and only if these values satisfy the equalities $A = e(G_z, \theta_1) \cdot e(G_r, \theta_2) \cdot e(\theta_3, \theta_4) \cdot \prod_{i=1}^n e(G_i, M_i)$ and $B = e(H_z, \theta_1) \cdot e(H_r, \theta_5) \cdot e(\theta_6, \theta_7) \cdot \prod_{i=1}^n e(H_i, M_i)$.

The scheme is known [1, 2] to be existentially unforgeable under chosen-message attacks under the q-SFP assumption, where q is the number of signing queries.

As pointed out in [1, 2], signature components $\{\theta_i\}_{i=2}^7$ can be publicly re-randomized so as to obtain a different signature $\{\theta'_i\}_{i=1}^7 \leftarrow \mathsf{ReRand}(pk, \sigma)$ on $(M_1, \ldots, M_n)$. After each randomization, we have $\theta'_1 = \theta_1$ whereas $\{\theta'_i\}_{i=2}^7$ are uniformly distributed among the group elements $(\theta_2, \ldots, \theta_7)$ such that the equalities $e(G_r, \theta'_2) \cdot e(\theta'_3, \theta'_4) = e(G_r, \theta_2) \cdot e(\theta_3, \theta_4)$ and $e(H_r, \theta'_5) \cdot e(\theta'_6, \theta'_7) = e(H_r, \theta_5) \cdot e(\theta_6, \theta_7)$ hold. As a result, $\{\theta'_i\}_{i \in \{3,6\}}$ are statistically independent of the message and other signature components.

It was also observed [1, 2] that signature components (θ_3, θ_6) can be used as a commitment to the message. Under the q-SFP assumption, it is infeasible to find signatures $\sigma = (\theta_1, \ldots, \theta_7)$, $\sigma' = (\theta'_1, \ldots, \theta'_7)$ on two distinct messages M, M' such that $(\theta_3, \theta_6) = (\theta'_3, \theta'_6)$. This is true even if the adversary has access to a signing oracle and obtains signatures on both M and M'.

2.4 Groth-Sahai Proof Systems

In [26], Groth and Sahai described efficient non-interactive witness indistinguishable (NIWI) proof systems that can be based on the DLIN assumption. In this case, they use prime order groups and a

common reference string containing three vectors $\vec{f_1}, \vec{f_2}, \vec{f_3} \in \mathbb{G}^3$, where $\vec{f_1} = (f_1, 1, g)$, $\vec{f_2} = (1, f_2, g)$ for some $f_1, f_2 \in \mathbb{G}$. To commit to a group element $X \in \mathbb{G}$, the prover chooses $r, s, t \xleftarrow{R} \mathbb{Z}_p^*$ and computes $\vec{C} = (1, 1, X) \cdot \vec{f_1}^{\,r} \cdot \vec{f_2}^{\,s} \cdot \vec{f_3}^{\,t}$. On a perfectly sound CRS, we have $\vec{f_3} = \vec{f_1}^{\,\xi_1} \cdot \vec{f_2}^{\,\xi_2}$ where $\xi_1, \xi_2 \in \mathbb{Z}_p^*$. Commitments $\vec{C} = (f_1^{r+\xi_1 t}, f_2^{s+\xi_2 t}, X \cdot g^{r+s+t(\xi_1+\xi_2)})$ are extractable commitments whose distribution is that of Boneh-Boyen-Shacham (BBS) ciphertexts [9]: committed values can be extracted using $\beta_1 = \log_g(f_1)$, $\beta_2 = \log_g(f_2)$. In the witness indistinguishability (WI) setting, vectors $\vec{f_3}$ is chosen outside the span of $(\vec{f_1}, \vec{f_2})$, so that $\vec{C}$ is a perfectly hiding commitment. Under the DLIN assumption, the two kinds of CRS are computationally indistinguishable.

To provide evidence that committed variables satisfy a set of relations, the prover computes one commitment per variable and one proof element per relation. Such efficient NIWI proofs are available for pairing-product equations, which are relations of the type

$$\prod_{i=1}^{n} e(\mathcal{A}_i, \mathcal{X}_i) \cdot \prod_{i=1}^{n} \cdot \prod_{j=1}^{n} e(\mathcal{X}_i, \mathcal{X}_j)^{a_{ij}} = t_T, \tag{2}$$

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 $i, j \in \{1, \ldots, n\}$.

In pairing-product equations, proofs for quadratic equations require 9 group elements whereas linear equations (*i.e.*, where $a_{ij} = 0$ for all i, j in equation (2)) only cost 3 group elements each.

In [7], Belenkiy *et al.* showed that Groth-Sahai proofs are perfectly randomizable. Given commitments $\{\vec{C}_{\mathcal{X}_i}\}_{i=1}^n$ and a NIWI proof $\vec{\pi}_{\text{PPE}}$ that committed variables $\{\mathcal{X}\}_{i=1}^n$ satisfy (2), anyone can publicly (*i.e.*, without knowing the witnesses) compute re-randomized commitments $\{\vec{C}_{\mathcal{X}'_i}\}_{i=1}^n$ and a re-randomized proof $\vec{\pi}'_{\text{PPE}}$ of the same statement. Moreover, $\{\vec{C}_{\mathcal{X}'_i}\}_{i=1}^n$ and $\vec{\pi}'_{\text{PPE}}$ are distributed as freshly generated commitments and proof. This property was notably used in [21, 18].

3 Linear-Size Quotable Signatures

In quotable signatures, given a signature on some message, one should only be able to derive signatures on arbitrary substrings of the original message. The message space $\mathcal{M}$ is also defined as the set of strings $\mathcal{M} := \Sigma^*$, where Σ is a set of symbols. The predicate P is univariate (*i.e.*, $|M| = 1$) and defined to have $P\big(\{\mathsf{Msg}_1\}, \mathsf{Msg}_2\big) = 1$ whenever Msg_2 is a substring of Msg_1.

The scheme bears resemblance with the homomorphic signature for subset predicates of [6] which also builds on structure-preserving signatures. In fact, the construction is itself a structure-preserving quotable signature as it allows signing sequences of group elements.

We actually use a variant of the unbounded AHO signature scheme which allows signing messages of arbitrary length with a public key of fixed size. In [1], this is achieved by taking advantage of the property called "signature binding" (and proved in [1, Lemma 3]), which informally says that signature components (θ_3, θ_6) can be used as a commitment to the message: namely, given only the public key and access to a signing oracle, unless the scheme is existentially forgeable under chosen-message attacks, it is infeasible to come up with two *distinct* messages $(M_1, \ldots, M_n), (M'_1, \ldots, M'_n)$ with corresponding valid signatures $\sigma = (\theta_1, \ldots, \theta_7)$ and $\sigma' = (\theta'_1, \ldots, \theta'_7)$ such that $\theta_3 = \theta'_3$ and $\theta_6 = \theta'_6$. This remains true *even* if $(M_1, \ldots, M_n)$ and $(M'_1, \ldots, M'_n)$ are both submitted to the signing oracle during the game. Using this observation, a basic signature scheme where the message space is $\mathbb{G}^3$ can be turned into an "unbounded" structure-preserving signature, where the signer can sign messages of arbitrary length. The idea is to use signature components $\{(\theta_{i,3}, \theta_{i,6})\}_{i=1}^n$ to link adjacent message blocks together: each block $m_i \in \mathbb{G}$ is signed along with the $(\theta_{i-1,3}, \theta_{i-1,6})$

components of the signature on the previous block $m_{i-1} \in \mathbb{G}$. In our scheme, we proceed in the same way but, unlike [1], we do not encode the total number of blocks within the message. This modification allows anyone to quote signatures by removing portions of the chain in its extremities. In order to prevent illegal combinations of two different chains, the signer processes the last block m_n of each message $(m_1, \ldots, m_n)$ by signing it with a pair of random group elements $(\tilde{\theta}_3, \tilde{\theta}_6)$ which are part of the private key. This allows us to prove security using the same arguments as in [1].

For the sake of privacy, the components of $\{\sigma_i\}_{i=1}^n$ are not explicitly given out but only appear within perfectly hiding Groth-Sahai commitments accompanied with appropriate NIWI arguments. At each signature derivation, commitments and NIWI arguments are suitably re-randomized.

An important difference with the construction for subset predicates in [6], is that underlying AHO signatures entirely appear in committed form. The reason is that using $(\theta_{i,3}, \theta_{i,6})$ in the chaining process prevents their re-randomization. For this reason, they also have to be committed so that we need to work with quadratic pairing-product equations.

In the following, when $X \in \mathbb{G}$ (resp. $X \in \mathbb{G}_T$), the notation $\iota(X)$ (resp. $\iota_{\mathbb{G}_T}(X)$) will be used to denote the vector $(1, 1, X) \in \mathbb{G}^3$ (resp. the 3×3 matrix containing X in position $(3, 3)$ and $1_{\mathbb{G}_T}$ everywhere else). Finally, we also use a symmetric bilinear map $F : \mathbb{G}^3 \times \mathbb{G}^3 \to \mathbb{G}_T^9$ such that, for any two vectors $\vec{X} = (X_1, X_2, X_3) \in \mathbb{G}^3$ and $\vec{Y} = (Y_1, Y_2, Y_3) \in \mathbb{G}^3$, $F(\vec{X}, \vec{Y}) = \tilde{F}(\vec{X}, \vec{Y})^{1/2} \cdot \tilde{F}(\vec{Y}, \vec{X})^{1/2}$, where the non-commutative mapping $\tilde{F} : \mathbb{G}^3 \times \mathbb{G}^3 \to \mathbb{G}_T^9$ sends $(\vec{X}, \vec{Y})$ onto the matrix $\tilde{F}(\vec{X}, \vec{Y})$ of entry-wise pairings (*i.e.*, containing $e(X_i, Y_j)$ in its entry (i, j)).

Keygen(λ)**:** given a security parameter $\lambda \in \mathbb{N}$, choose bilinear groups $(\mathbb{G}, \mathbb{G}_T)$ of prime order $p > 2^\lambda$.

1. Choose a Groth-Sahai CRS $\mathbf{f} = (\vec{f}_1, \vec{f}_2, \vec{f}_3)$ for the perfect WI setting. More precisely, choose $\vec{f}_1 = (f_1, 1, g)$, $\vec{f}_2 = (1, f_2, g)$, and $\vec{f}_3 = \vec{f}_1^{\,\xi_1} \cdot \vec{f}_2^{\,\xi_2} \cdot (1, 1, g)^{-1}$, with $f_1, f_2, g \xleftarrow{R} \mathbb{G}$, $\xi_1, \xi_2 \xleftarrow{R} \mathbb{Z}_p$.
2. Generate a key pair $(sk_{\mathsf{aho}}, pk_{\mathsf{aho}})$ for the AHO signature in order to sign messages consisting of three group elements. This key pair consists of $sk_{\mathsf{aho}} = \big(\alpha_a, \alpha_b, \gamma_z, \delta_z, \{\gamma_i, \delta_i\}_{i=1}^3\big)$ and

$$pk_{\mathsf{aho}} = \Big(G_r,\ H_r,\ G_z = G_r^{\gamma_z},\ H_z = H_r^{\delta_z},\ \{G_i = G_r^{\gamma_i},\ H_i = H_r^{\delta_i}\}_{i=1}^3,\ A,\ B\Big).$$

3. Choose two uniformly random group elements $\tilde{\theta}_3, \tilde{\theta}_6 \xleftarrow{R} \mathbb{G}$.

The public key consists of $\mathsf{pk} := \Big((\mathbb{G}, \mathbb{G}_T),\ \mathbf{f},\ pk_{\mathsf{aho}}\Big)$ whereas the private key is defined to be $\mathsf{sk} = \big(sk_{\mathsf{aho}},\ (\tilde{\theta}_3, \tilde{\theta}_6)\big)$. The public key defines the set of symbols $\Sigma = \mathbb{G}$.

Sign$(\mathsf{sk}, \mathsf{Msg})$**:** given $\mathsf{sk} = \big(sk_{\mathsf{aho}},\ \ (\tilde{\theta}_3, \tilde{\theta}_6)\big)$ and a length-n message $\mathsf{Msg} = (m_1, \ldots, m_n) \in \mathbb{G}^n$, for some $n \in \mathsf{poly}(\lambda)$ and where $m_i \in \mathbb{G}$ for each $i \in \{1, \ldots, n\}$, do the following.

1. Define $(\theta_{n+1,3}, \theta_{n+1,6}) = (\tilde{\theta}_3, \tilde{\theta}_6)$. Then, for $k \in \{3, 6\}$, compute Groth-Sahai commitments $\vec{C}_{\theta_{n+1,k}} = \iota(\theta_{n+1,k}) \cdot \vec{f}_1^{\,r_{\theta_{n+1,k}}} \cdot \vec{f}_2^{\,s_{\theta_{n+1,k}}} \cdot \vec{f}_3^{\,t_{\theta_{n+1,k}}}$.
2. For each $j = n$ down to 1, generate an AHO signature $(\theta_{j,1}, \ldots, \theta_{j,7}) \in \mathbb{G}^7$ on the message $(m_j, \theta_{j+1,3}, \theta_{j+1,6}) \in \mathbb{G}^3$. For each $k \in \{1, \ldots, 7\}$ and $j \in \{1, \ldots, n\}$, generate commitments $\vec{C}_{\theta_{j,k}} = \iota(\theta_{j,k}) \cdot \vec{f}_1^{\,r_{\theta_{j,k}}} \cdot \vec{f}_2^{\,s_{\theta_{j,k}}} \cdot \vec{f}_3^{\,t_{\theta_{j,k}}}$. Next, generate NIWI proofs $\vec{\pi}_{\mathsf{aho},j,1}, \vec{\pi}_{\mathsf{aho},j,2} \in \mathbb{G}^9$ that variables $(\theta_{j,1}, \theta_{j,2}, \theta_{j,3}, \theta_{j,4}, \theta_{j,5}, \theta_{j,6}, \theta_{j,7})$ satisfy

$$\begin{aligned} A \cdot e(G_1, m_j)^{-1} &= e(G_z, \theta_{j,1}) \cdot e(G_r, \theta_{j,2}) \cdot e(\theta_{j,3}, \theta_{j,4}) \cdot e(G_2, \theta_{j+1,3}) \cdot e(G_3, \theta_{j+1,6}) \\ B \cdot e(H_1, m_j)^{-1} &= e(H_z, \theta_{j,1}) \cdot e(H_r, \theta_{j,5}) \cdot e(\theta_{j,6}, \theta_{j,7}) \cdot e(H_2, \theta_{j+1,3}) \cdot e(H_3, \theta_{j+1,6}) \end{aligned} \tag{3}$$

These equations are quadratic, so that $\{\vec{\pi}_{\mathsf{aho},j,1}, \vec{\pi}_{\mathsf{aho},j,2}\}_{j=1}^n$ consist of 9 group elements each.

3. Return the signature

$$\sigma = \Big(\{\vec{C}_{\theta_{n+1,k}}\}_{k\in\{3,6\}}, \big\{\{\vec{C}_{\theta_{j,k}}\}_{k=1}^{7}, \vec{\pi}_{\mathsf{aho},j,1}, \vec{\pi}_{\mathsf{aho},j,2}\big\}_{j=1}^{n}\Big). \tag{4}$$

SignDerive$(\mathsf{pk}, \mathsf{Msg}, \mathsf{Msg}', \sigma)$**:** given the public key pk and two messages $\mathsf{Msg} = (m_1, \ldots, m_n) \in \mathbb{G}^n$ and $\mathsf{Msg}' = (m'_1, \ldots, m'_{n'}) \in \mathbb{G}^{n'}$, return $\perp$ if Msg' is not a substring of Msg. Otherwise, there exists an index $i \in \{1, \ldots, n - n' + 1\}$ such that $\mathsf{Msg}' = (m'_1, \ldots, m'_{n'}) = (m_i, \ldots, m_{i+n'-1})$. Then, parse σ as in (4) and, for each $i \in \{1, \ldots, n'\}$, conduct the following steps.

1. Define the sub-signature $\tilde{\sigma} = \big(\{\vec{C}_{\theta_{i+n',k}}\}_{k\in\{3,6\}}, \big\{\{\vec{C}_{\theta_{i+j,k}}\}_{k=1}^{7}, \vec{\pi}_{\mathsf{aho},i+j,1}, \vec{\pi}_{\mathsf{aho},i+j,2}\big\}_{j=0}^{n'-1}\big)$.
2. Re-randomize the commitments $\vec{C}'_{\theta_{i+j,k}} = \vec{C}_{\theta_{i+j,k}} \cdot \vec{f}_1^{\,r'_{\theta_{i+j,k}}} \cdot \vec{f}_2^{\,s'_{\theta_{i+j,k}}} \cdot \vec{f}_3^{\,t'_{\theta_{i+j,k}}}$ for $j = 0$ to $n' - 1$ and $k = 1$ to 7. Likewise, compute re-randomized versions $\{\vec{C}'_{\theta_{i+n',k}}\}_{k\in\{3,6\}}$ of $\{\vec{C}_{\theta_{i+n',k}}\}_{k\in\{3,6\}}$. Finally, re-randomize the proofs $\{\vec{\pi}_{\mathsf{aho},i+j,1} = (\vec{\pi}_{i+j,1}, \vec{\pi}_{i+j,2}, \vec{\pi}_{i+j,3})\}_{j=0}^{n'-1}$ and $\{\vec{\pi}_{\mathsf{aho},i+j,2} = (\vec{\pi}_{i+j,4}, \vec{\pi}_{i+j,5}, \vec{\pi}_{i+j,6})\}_{j=0}^{n'-1}$ as suggested in [7].
3. Return the signature

$$\sigma' = \Big(\{\vec{C}'_{\theta_{i+n',k}}\}_{k\in\{3,6\}}, \big\{\{\vec{C}'_{\theta_{i+j,k}}\}_{k=1}^{7}, \vec{\pi}'_{\mathsf{aho},i+j,1}, \vec{\pi}'_{\mathsf{aho},i+j,2}\big\}_{j=0}^{n'-1}\Big). \tag{5}$$

Verify$(\mathsf{pk}, \mathsf{Msg}, \sigma)$**:** given pk, a signature σ and a message $\mathsf{Msg} = (m_1, \ldots, m_n) \in \mathbb{G}^n$, parse σ as per (4) and do the following. For $j = 1$ to n, return 0 if $\vec{\pi}_{\mathsf{aho},j,1} = (\vec{\pi}_{j,1}, \vec{\pi}_{j,2}, \vec{\pi}_{j,3})$ and $\vec{\pi}_{\mathsf{aho},j,2} = (\vec{\pi}_{j,4}, \vec{\pi}_{j,5}, \vec{\pi}_{j,6})$ do not satisfy the equations below. Otherwise, return 1.

$$\begin{aligned}
\iota_{\mathbb{G}_T}(A) \cdot F\big(\iota(G_1), \iota(m_j)\big)^{-1} &= F\big(\iota(G_z), \vec{C}_{\theta_{j,1}}\big) \cdot F\big(\iota(G_r), \vec{C}_{\theta_{j,2}}\big) \cdot F\big(\vec{C}_{\theta_{j,3}}, \vec{C}_{\theta_{j,4}}\big) \\
&\quad \cdot F\big(\iota(G_2), \vec{C}_{\theta_{j+1,3}}\big) \cdot F\big(\iota(G_3), \vec{C}_{\theta_{j+1,6}}\big) \cdot \prod_{k=1}^{3} F\big(\vec{\pi}_{j,k}, \vec{f}_k\big) \qquad (6) \\
\iota_{\mathbb{G}_T}(B) \cdot F\big(\iota(H_1), \iota(m_j)\big)^{-1} &= F\big(\iota(H_z), \vec{C}_{\theta_{j,1}}\big) \cdot F\big(\iota(H_r), \vec{C}_{\theta_{j,5}}\big) \cdot F\big(\vec{C}_{\theta_{j,6}}, \vec{C}_{\theta_{j,7}}\big) \\
&\quad \cdot F\big(\iota(H_2), \vec{C}_{\theta_{j+1,3}}\big) \cdot F\big(\iota(H_3), \vec{C}_{\theta_{j+1,6}}\big) \cdot \prod_{k=1}^{3} F\big(\vec{\pi}_{j,k+1}, \vec{f}_k\big).
\end{aligned}$$

Unlike the scheme of [4], the above system allows signing arbitrarily long messages with a public key of constant size whereas [4] requires to set a logarithmic bound on the length of signed messages at key generation. The signature length is asymptotically optimal: a n-symbol message can be signed using $39n + 6$ group elements.

On the other hand, we lose a useful feature of the construction in [4]. The latter allows the derivation algorithm to produce two kinds of derived signatures: when the message m' consists of ℓ symbols, Type I signatures contain $O(\ell \log \ell)$ group elements and support subsequent quoting. Alternatively, the quoting algorithm can derive a much shorter Type II signature, which comprises $O(\log \ell)$ elements, but cannot be quoted any further. In our scheme, the quoter can only produce Type I signatures and does not have the same flexibility as in [4]. It would be interesting to retain the latter while keeping linear-size Type I signatures.

We now turn to the security of the scheme and first observe that it is clearly completely context-hiding due to the use of a witness indistinguishable Groth-Sahai CRS.

Theorem 1. *The above quotable signature scheme is completely context hiding.*

Proof. Each signature only consists of perfectly hiding commitments and perfectly NIWI arguments, which can be perfectly re-randomized at each derivation. □

The unforgeability relies on the DLIN assumption and the security properties of AHO signatures, as established by Theorem 2.

Theorem 2. *The scheme is existentially unforgeable against chosen-message attacks under the $(q \cdot L + 1)$-SFP and DLIN assumptions, where q denotes the maximal number of signing queries and L is the maximal number of symbols per signing query.*

Proof. Since the scheme is completely context hiding, we only need to prove unforgeability using the simpler definition where the adversary $\mathcal{A}$ only has a signing oracle. The proof uses a sequence of games where, for each i, S_i stands for the event that $\mathcal{A}$ produces a valid forgery in Game_i.

Game_0: This game is the real game. We denote by S_0 the event that the adversary $\mathcal{A}$ manages to output a successful forgery. Obviously, $\mathcal{A}$'s advantage is $\Pr[S_0]$.

Game_1: We change the generation of the public key and set up $\mathbf{f} = (\vec{f}_1, \vec{f}_2, \vec{f}_3)$ as a perfectly sound Groth-Sahai CRS. Concretely, the only change is that the challenger $\mathcal{B}$ chooses $\vec{f}_3$ in the span of $\vec{f}_1 = (f_1, 1, g)$ and $\vec{f}_2 = (1, f_2, g)$, where $f_1 = g^{\phi_1}$ and $f_2 = g^{\phi_2}$, for random chosen $\phi_1, \phi_2 \xleftarrow{R} \mathbb{Z}_p$. Signing queries are answered as in Game_0, using the private key $(sk_{\mathsf{aho}}, (\tilde{\theta}_3, \tilde{\theta}_6))$ and generating NIWI arguments faithfully. Under the DLIN assumption, this change should not significantly affect $\mathcal{A}$'s behavior and we have $|\Pr[S_1] - \Pr[S_0]| \leq \mathbf{Adv}^{\text{DLIN}}(\mathcal{B})$. Note that the reduction is immediate as $\mathcal{B}$ does not need the trapdoor (ϕ_1, ϕ_2) at any time. In Game_1, perfectly hiding Groth-Sahai commitments (and NIWI arguments) are traded for perfectly binding commitments (and perfectly sound proofs).

Game_2: This game is identical to Game 1 except that we bring a conceptual change in the generation of sk. Instead of merely choosing $(\tilde{\theta}_3, \tilde{\theta}_6)$ at random, the challenger $\mathcal{B}$ picks a uniformly random group element $\tilde{m} \xleftarrow{R} \mathbb{G}$ and computes an AHO signature $\{\tilde{\theta}_k\}_{k=1}^{7}$ on the "dummy" message $(\tilde{m}, 1, 1)$. The resulting $(\tilde{\theta}_3, \tilde{\theta}_6)$ are included in the private key sk whereas $\tilde{m}$ and $\{\tilde{\theta}_k\}_{k\in\{1,2,4,5,7\}}$ are retained by $\mathcal{B}$. We argue that this change does not alter $\mathcal{A}$'s view whatsoever since $(\tilde{\theta}_3, \tilde{\theta}_6)$ have the same distribution either way. Indeed, in Game_2, they remain uniformly distributed in $\mathbb{G}^2$ and statistically independent of the message $\tilde{m}$ and other signature components. We have $\Pr[S_2] = \Pr[S_1]$.

In Game_2, $\mathcal{B}$ uses the values $(\phi_1, \phi_2) = (\log_g(f_1), \log_g(f_2))$ that were defined in Game_1. When $\mathcal{A}$ outputs a forgery $\sigma^\star$ on a message $(m_1^\star, \ldots, m_{n^\star}^\star)$, $\mathcal{B}$ uses (ϕ_1, ϕ_2) to extract $(\theta_{n^\star+1,3}^\star, \theta_{n^\star+1,6}^\star)$ as well as a sequence of AHO signatures $\{\sigma_j^\star = (\theta_{j,1}^\star, \ldots, \theta_{j,7}^\star)\}_{j=1}^{n^\star}$ from the Groth-Sahai commitments contained in $\sigma^\star$. The perfect soundness of $\{\vec{\pi}_{\mathsf{aho},j,1}^\star, \vec{\pi}_{\mathsf{aho},j,2}^\star\}_{j=1}^{n^\star}$ guarantees that extracted values $(m_1^\star, \ldots, m_{n^\star}^\star)$, $\{\sigma_j^\star\}_{j=1}^{n^\star}$ and $(\theta_{n^\star+1,3}^\star, \theta_{n^\star+1,6}^\star)$ satisfy equations (3).

In Game_2, we can prove that event S_2 occurs with negligible probability if the $(q \cdot L + 1)$-SFP assumption holds. Indeed, if $\mathcal{A}$ is successful in Game_3, $\{\sigma_j^\star = (\theta_{j,1}^\star, \ldots, \theta_{j,7}^\star)\}_{j=1}^{n^\star}$ is a sequence of valid AHO signatures on the messages $\{(m_j^\star, \theta_{j+1,3}^\star, \theta_{j+1,6}^\star)\}_{j=1}^{n^\star}$ but $(m_1^\star, \ldots, m_{n^\star}^\star)$ is not a subsequence involved in any of the signing queries. We can thus distinguish two situations.

Case A. There exists $j^\dagger \in \{1, \ldots, n\}$ such that $\mathcal{B}$ never had to sign $(m_{j^\dagger}^\star, \theta_{j^\dagger+1,3}^\star, \theta_{j^\dagger+1,6}^\star)$ in any signing query.

Case B. The messages $\{(m^\star_j, \theta^\star_{j+1,3}, \theta^\star_{j+1,6})\}_{j=1}^{n^\star}$ were all signed by $\mathcal{B}$ at some point of the game but not all of them were involved in the same query. This covers the case of an adversary mixing substrings of two different messages for which it received signatures.

In Case A, it is immediate that $\mathcal{A}$ necessarily broke the chosen-message security of the AHO signature: the reduction $\mathcal{B}$ simply outputs $(m^\star_{j^\dagger}, \theta^\star_{j^\dagger+1,3}, \theta^\star_{j^\dagger+1,6})$ and the signature $\sigma^\star_{j^\dagger}$.

We are thus left with Case B for which we know that $(m^\star_1, \theta^\star_{2,3}, \theta^\star_{2,6})$ was involved in the κ-th signing query $\mathsf{Msg}_\kappa = (m_{\kappa,1}, \ldots, m_{\kappa,n_\kappa})$, for some integers $\kappa \in \{1, \ldots, q\}$ and $n_\kappa \in \{1, \ldots, L\}$. Let $\{(\theta_{\kappa,j,1}, \ldots, \theta_{\kappa,j,7})\}_{j=1}^{n_\kappa}$ be the AHO signatures that were used to answer the κ-th signing query. Let also $t \in \{1, \ldots, n_\kappa\}$ be such that $(m_{\kappa,t}, \theta_{\kappa,t+1,3}, \theta_{\kappa,t+1,6}) = (m^\star_1, \theta^\star_{2,3}, \theta^\star_{2,6})$.

We now define $j^\star$ to be the largest index in $\{1, \ldots, n^\star - 1\}$ such that

$$(m_{\kappa,t+j^\star-1}, \theta_{\kappa,t+j^\star,3}, \theta_{\kappa,t+j^\star,6}) = (m^\star_{j^\star}, \theta^\star_{j^\star+1,3}, \theta^\star_{j^\star+1,6}).$$

At this step, we further consider two sub-cases of Case B:

Case $t + j^\star < n_\kappa + 1$: Since $m_{\kappa,t+j^\star} \neq m_{j^\star+1}$ or $(\theta_{\kappa,t+j^\star+1,3}, \theta_{\kappa,t+j^\star+1,6}) \neq (\theta^\star_{j^\star+2,3}, \theta^\star_{j^\star+2,6})$, the signature binding property of the AHO signature is broken since we have two distinct messages whose signatures share the same $\theta^\star_{j^\star+1,3}, \theta^\star_{j^\star+1,6}$ components. As implied by the results of [1], this contradicts the $(q \cdot L + 1)$-SFP assumption since $\mathcal{B}$ computes at most $q \cdot L + 1$ AHO signatures.

Case $t + j^\star = n_\kappa + 1$: We have $(\theta^\star_{j^\star+1,3}, \theta^\star_{j^\star+1,6}) = (\theta_{\kappa,n_{\kappa+1},3}, \theta_{\kappa,n_{\kappa+1},6}) = (\tilde{\theta}_3, \tilde{\theta}_6)$, which means that $(m^\star_{j^\star}, \theta^\star_{j^\star+1,3}, \theta^\star_{j^\star+1,6})$ was the message of an "end-of-chain" signature produced by $\mathcal{B}$. Said otherwise, this is a forgery where $(m^\star_1, \ldots, m^\star_{n^\star})$ is a super-string of $(m_{\kappa,t}, \ldots, m_{\kappa,n_k})$. In this case, thanks to the modification introduced in Game$_2$, $\mathcal{B}$ knows $\{\tilde{\theta}_k\}_{k \in \{1,2,4,5,7\}}$ as well as a dummy message $\tilde{m}$ such that $(\tilde{\theta}_1, \ldots, \tilde{\theta}_7)$ is a valid AHO signature on $(\tilde{m}, 1, 1)$. With overwhelming probability, we obtain distinct messages $(\tilde{m}, 1, 1)$ and $(m^\star_{j^\star+1}, \theta^\star_{j^\star+2,3}, \theta^\star_{j^\star+2,6})$ that share the same signature components $(\theta^\star_{j^\star+1,3}, \theta^\star_{j^\star+1,6}) = (\tilde{\theta}_3, \tilde{\theta}_6)$. Indeed, the pair $(\tilde{\theta}_3, \tilde{\theta}_6)$ is statistically independent of the dummy message $\tilde{m}$ and the latter was uniformly chosen in $\mathbb{G}$. It comes that we can only have $m^\star_{j^\star+1} = \tilde{m}$ by pure chance.

In Case B, the signature binding property of AHO signatures is thus broken either way and we can eventually write $\Pr[S_2] \leq 2 \cdot \mathbf{Adv}^{(q \cdot L+1)\text{-SFP}}(\mathcal{B})$, where the factor 2 accounts for the fact that the reduction has to guess beforehand which of Case A or Case B will come about. Depending on this guess, $\mathcal{B}$ undertakes to either attack the standard unforgeability of AHO signatures or, alternatively, break their signature-binding property. In either case, $\mathcal{B}$ answers $\mathcal{A}$'s queries by invoking the signing oracle in its interaction with the appropriate challenger.

Putting the above altogether, we find the upper bound

$$\Pr[S_0] \leq \mathbf{Adv}^{\text{DLIN}}(\mathcal{B}) + 2 \cdot \mathbf{Adv}^{(q \cdot L+1)\text{-SFP}}(\mathcal{B})$$

on the forger's advantage. □

4 Completely Context-Hiding Linearly Homomorphic Signatures

We now turn to linearly homomorphic signatures for which the syntax and the security definitions of Section 2 can be simplified as explained in Appendix B.

Our starting point is the weakly context-hiding linearly homomorphic signature of [6], which

is described in Appendix C. Its public key includes group elements g^α, v and $\{g_i\}_{i=1}^n$, where n is the dimension of vectors to be signed. Signatures of vectors $\vec{v} = (v_1, \ldots, v_n)$ are of the form $(\sigma_1, \sigma_2, s) = \big((\prod_{i=1}^n g_i^{v_i} \cdot v^s)^\alpha \cdot H_{\mathbb{G}}(\tau)^r, g^r, s\big)$, where $r, s \in_R \mathbb{Z}_p$ and τ identifies the linear subspace.

The reason why the scheme is only weakly context-hiding is that the signature component s cannot be re-randomized. Hence, it always allows linking a derived signature to those it was obtained from. To render the scheme completely context-hiding, we need to modify the signing algorithm so as to hide $s \in \mathbb{Z}_p$. In signatures, the exponent s is replaced by Groth-Sahai commitments to group elements $(g^s, g^{\alpha \cdot s})$, where g^α is the public key, together with NIWI arguments that these are correctly formed. Then, the randomizability properties of Groth-Sahai proofs come in handy to guarantee that derived signatures will be statistically independent of original signatures.

In the notations hereunder, for any element $h \in \mathbb{G}$ and any vector $\vec{g} = (g_1, g_2, g_3) \in \mathbb{G}^3$, $E(h, \vec{g})$ stands for the vector $\big(e(h, g_1), e(h, g_2), e(h, g_3)\big) \in \mathbb{G}_T^3$.

Keygen(λ, n)**:** given a security parameter $\lambda \in \mathbb{N}$ and an integer $n \in \mathsf{poly}(\lambda)$, choose bilinear groups $(\mathbb{G}, \mathbb{G}_T)$ of prime order $p > 2^\lambda$.

1. Choose $\alpha \xleftarrow{R} \mathbb{Z}_p$, $g, v \xleftarrow{R} \mathbb{G}$ and $u_0, u_1, \ldots, u_L \xleftarrow{R} \mathbb{G}$, for some $L \in \mathsf{poly}(\lambda)$. Elements $(u_0, \ldots, u_L) \in \mathbb{G}^{L+1}$ will define hash function $H_{\mathbb{G}} : \{0,1\}^L \to \mathbb{G}$ mapping any L-bit string $m = m[1] \ldots m[L] \in \{0,1\}^L$ onto a hash value $H_{\mathbb{G}}(m) = u_0 \cdot \prod_{i=1}^L u_i^{m[i]}$.
2. Pick $g_i \xleftarrow{R} \mathbb{G}$ for $i = 1$ to n. Also, define the identifier space $\mathcal{T} := \{0,1\}^L$.
3. Generate Groth-Sahai CRS $\mathbf{f} = (\vec{f}_1, \vec{f}_2, \vec{f}_3)$ for the perfect WI setting. Namely, choose vectors $\vec{f}_1 = (f_1, 1, g)$, $\vec{f}_2 = (1, f_2, g)$, and $\vec{f}_3 = \vec{f}_1^{\,\xi_1} \cdot \vec{f}_2^{\,\xi_2} \cdot (1, 1, g)^{-1}$, with $f_1, f_2 \xleftarrow{R} \mathbb{G}$, $\xi_1, \xi_2 \xleftarrow{R} \mathbb{Z}_p$.

The private key is $\mathsf{sk} := \alpha$ and the public key consists of

$$\mathsf{pk} := \Big((\mathbb{G}, \mathbb{G}_T),\ g,\ g^\alpha,\ v,\ \{g_i\}_{i=1}^n,\ \{u_i\}_{i=0}^L,\ \mathbf{f}\Big).$$

Sign$(\mathsf{sk}, \tau, \vec{v})$**:** given a vector $\vec{v} = (v_1, \ldots, v_n) \in \mathbb{Z}_p^n$, a file identifier $\tau \in \{0,1\}^L$ and the private key $\mathsf{sk} = \alpha \in \mathbb{Z}_p$, conduct the following steps.

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

$$\sigma_1 = (g_1^{v_1} \cdots g_n^{v_n} \cdot v^s)^\alpha \cdot H_{\mathbb{G}}(\tau)^r, \qquad \sigma_2 = g^r, \qquad \sigma_3 = g^s, \qquad \sigma_4 = g^{\alpha \cdot s}.$$

2. Compute Groth-Sahai commitments to $(\sigma_1, \sigma_3, \sigma_4)$. Namely, for each $j \in \{1, 3, 4\}$, choose $r_{\sigma_j}, s_{\sigma_j}, t_{\sigma_j} \xleftarrow{R} \mathbb{Z}_p$ and compute $\vec{C}_{\sigma_j} = (1, 1, \sigma_j) \cdot \vec{f}_1^{\,r_{\sigma_j}} \cdot \vec{f}_2^{\,s_{\sigma_j}} \cdot \vec{f}_3^{\,t_{\sigma_j}}$.
3. Generate a NIWI proof that variables $(\sigma_1, \sigma_3, \sigma_4) \in \mathbb{G}^3$ satisfy the linear equations

$$e(\sigma_1, g) = e(\prod_{i=1}^n g_i^{v_i}, g^\alpha) \cdot e(v, \sigma_4) \cdot e(H_{\mathbb{G}}(\tau), \sigma_2), \tag{7}$$

$$e(\sigma_3, g^\alpha) = e(g, \sigma_4). \tag{8}$$

These proofs are obtained as $\vec{\pi}_1 = (\pi_{1,1}, \pi_{1,2}, \pi_{1,3}) = \big(g^{r_{\sigma_1}} \cdot v^{-r_{\sigma_4}},\ g^{s_{\sigma_1}} \cdot v^{-s_{\sigma_4}},\ g^{t_{\sigma_1}} \cdot v^{-t_{\sigma_4}}\big)$ and $\vec{\pi}_2 = (\pi_{2,1}, \pi_{2,2}, \pi_{2,3}) = \big((g^\alpha)^{r_{\sigma_3}} \cdot g^{-r_{\sigma_4}},\ (g^\alpha)^{s_{\sigma_3}} \cdot g^{-s_{\sigma_4}},\ (g^\alpha)^{t_{\sigma_3}} \cdot g^{-t_{\sigma_4}}\big)$, which satisfy

$$E(g, \vec{C}_{\sigma_1}) = E\big(\prod_{i=1}^n g_i^{v_i}, (1, 1, g^\alpha)\big) \cdot E(v, \vec{C}_{\sigma_4}) \cdot E\big(H_{\mathbb{G}}(\tau), (1, 1, \sigma_2)\big) \cdot \prod_{j=1}^3 E(\pi_{1,j}, \vec{f}_j) \tag{9}$$

$$E(g^\alpha, \vec{C}_{\sigma_3}) = E(g, \vec{C}_{\sigma_4}) \cdot \prod_{j=1}^3 E(\pi_{2,j}, \vec{f}_j). \tag{10}$$

The signature eventually consists of $\sigma = \big(\vec{C}_{\sigma_1}, \sigma_2, \vec{C}_{\sigma_3}, \vec{C}_{\sigma_4}, \vec{\pi}_1, \vec{\pi}_2\big) \in \mathbb{G}^{16}$.

SignDerive$(\mathsf{pk}, \tau, \{(\beta_i, \sigma^{(i)})\}_{i=1}^{\ell})$**:** given pk, a file identifier τ and ℓ tuples $(\beta_i, \sigma^{(i)})$, parse each signature $\sigma^{(i)}$ as $\sigma^{(i)} = \big(\vec{C}_{\sigma_{i,1}}, \sigma_{i,2}, \vec{C}_{\sigma_{i,3}}, \vec{C}_{\sigma_{i,4}}, \vec{\pi}_{i,1}, \vec{\pi}_{i,2}\big) \in \mathbb{G}^{16}$ for $i = 1$ to ℓ.

1. Choose $\tilde{r} \xleftarrow{R} \mathbb{Z}_p$. Then, compute $\sigma_2 = \prod_{i=1}^{\ell} \sigma_{i,2}^{\beta_i} \cdot g^{\tilde{r}}$ and

$$\vec{C}_{\sigma_1} = \prod_{i=1}^{\ell} \vec{C}_{\sigma_{i,1}}^{\beta_i} \cdot (1, 1, H_{\mathbb{G}}(\tau)^{\tilde{r}}) \qquad \vec{C}_{\sigma_3} = \prod_{i=1}^{\ell} \vec{C}_{\sigma_{i,3}}^{\beta_i} \qquad \vec{C}_{\sigma_4} = \prod_{i=1}^{\ell} \vec{C}_{\sigma_{i,4}}^{\beta_i}$$

   as well as $\vec{\pi}_1 = \prod_{i=1}^{\ell} \vec{\pi}_{i,1}^{\beta_i}$ and $\vec{\pi}_2 = \prod_{i=1}^{\ell} \vec{\pi}_{i,2}^{\beta_i}$. We note that, thanks to the linearity properties of equations (9)-(10), the new values $\big(\vec{C}_{\sigma_1}, \sigma_2, \vec{C}_{\sigma_3}, \vec{C}_{\sigma_4}, \vec{\pi}_1, \vec{\pi}_2\big)$ still satisfy these equations.
2. Re-randomize commitments $\vec{C}_{\sigma_1}, \vec{C}_{\sigma_3}, \vec{C}_{\sigma_4}$ and the proofs $\vec{\pi}_1, \vec{\pi}_2$. Finally, return the re-randomized signature $\sigma' = \big(\vec{C}'_{\sigma_1}, \sigma'_2, \vec{C}'_{\sigma_3}, \vec{C}'_{\sigma_4}, \vec{\pi}'_1, \vec{\pi}'_2\big)$.

Verify$(\mathsf{pk}, \tau, \vec{y}, \sigma)$**:** given pk, a signature $\sigma = \big(\vec{C}_{\sigma_1}, \sigma_2, \vec{C}_{\sigma_3}, \vec{C}_{\sigma_4}, \vec{\pi}_1, \vec{\pi}_2\big) \in \mathbb{G}^{16}$ and a message $(\tau, \vec{y})$, where $\tau \in \{0,1\}^L$ and $\vec{y} = (y_1, \ldots, y_n) \in (\mathbb{Z}_p)^n$, return $\perp$ if $\vec{y} = \vec{0}$. Otherwise, return 1 if and only if equations (9)-(10) are satisfied.

It is immediate that the scheme is completely hiding as established by Theorem 3.

Theorem 3. *The scheme is completely context hiding.*

Proof. The statement follows from the fact that, on a perfectly hiding CRS $(\vec{f}_1, \vec{f}_2, \vec{f}_3)$, all commitments are perfectly hiding and arguments are perfectly WI. Moreover, signature components σ_2, commitments $\vec{C}_{\sigma_1}, \vec{C}_{\sigma_3}, \vec{C}_{\sigma_4}$ and $\vec{\pi}_1, \vec{\pi}_2$ are perfectly re-randomized by the derivation algorithm. For this reason, the output of SignDerive has the same distribution as a fresh signature. □

In the proof of unforgeability, we will need a slightly stronger (but still simple) assumption than the standard CDH assumption.

The proof assumes that the adversary only obtains signatures on linearly independent vectors. This is not a limitation since, in practice (see, e.g., [10]), one usually augments the signed vectors (e.g., by unit vectors) so that they are always linearly independent. As in [19] and [6, Appendix F], we also assume that a given pair $(\tau, \vec{v})$ is always signed using the same s. This can be enforced by deriving s from a pseudo-random function of τ and $\vec{v}$.

Theorem 4. *The scheme is unforgeable assuming that the DLIN and FlexDH assumption both hold in the group* $\mathbb{G}$. (The proof is available in Appendix D).

Acknowledgements

The authors thank the anonymous reviewers for useful comments.

References

1. M. Abe, K. Haralambiev, M. Ohkubo. Signing on Elements in Bilinear Groups for Modular Protocol Design. Cryptology ePrint Archive: Report 2010/133, 2010.
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, 2010.

3. S. Agrawal, D. Boneh, X. Boyen, D. Freeman. Preventing Pollution Attacks in Multi-source Network Coding. In *PKC'10*, *LNCS* 6056, pp. 161–176, 2010.
4. J.-H. Ahn, D. Boneh, J. Camenisch, S. Hohenberger, a. shelat, B. Waters. Computing on Authenticated Data. In *TCC 2012*, *LNCS* 7194, pp. 1–20, 2012.
5. N. Attrapadung, B. Libert. Homomorphic Network Coding Signatures in the Standard Model. In *PKC'11*, *LNCS* 6571, pp. 17–34, 2011.
6. N. Attrapadung, B. Libert, T. Peters. Computing on Authenticated Data: New Privacy Definitions and Constructions. In *Asiacrypt'12*, *LNCS* 7658, pp. 367–385, 2012.
7. M. Belenkiy, J. Camenisch, M. Chase, M. Kohlweiss, A. Lysyanskaya, H. Shacham. Randomizable Proofs and Delegatable Anonymous Credentials. In *Crypto'09*, *LNCS* 5677, pp. 108–125, 2009.
8. M. Bellare, G. Neven. Transitive Signatures Based on Factoring and RSA. In *Asiacrypt'02*, *LNCS* 2501, 397–414, 2002.
9. D. Boneh, X. Boyen, H. Shacham. Short Group Signatures. In *Crypto'04*, *LNCS* 3152, pp. 41–55. Springer, 2004.
10. D. Boneh, D. Freeman, J. Katz, B. Waters. Signing a Linear Subspace: Signature Schemes for Network Coding. In *PKC'09*, *LNCS* 5443, pp. 68–87, 2009.
11. D. Boneh, D. Freeman. Linearly Homomorphic Signatures over Binary Fields and New Tools for Lattice-Based Signatures. In *PKC'11*, *LNCS* 6571, pp. 1–16, 2011.
12. D. Boneh, D. Freeman. Homomorphic Signatures for Polynomial Functions. In *Eurocrypt'11*, *LNCS* 6632, pp. 149–168, 2011.
13. C. Brzuska, H. Busch, O. Dagdelen, M. Fischlin, M. Franz, S. Katzenbeisser, M. Manulis, C. Onete, A. Peter, B. Poettering, D. Schröder Redactable Signatures for Tree-Structured Data: Definitions and Constructions. In *ACNS'10*, *LNCS* 6123, pp. 87–104, 2010.
14. C. Brzuska, M. Fischlin, T. Freudenreich, A. Lehmann, M. Page, J. Schelbert, D. Schröder, F. Volk. Security of Sanitizable Signatures Revisited. In *PKC'09*, *LNCS* 3376, pp. 317–336, 2009.
15. C. Brzuska, M. Fischlin, A. Lehmann, D. Schröder. Unlinkability of Sanitizable Signatures. In *PKC'10*, *LNCS* 6056, pp. 444-461, 2010.
16. D. Catalano, D. Fiore, B. Warinschi. Adaptive Pseudo-free Groups and Applications. In *Eurocrypt'11*, *LNCS* 6632, pp. 207–223, 2011.
17. D. Catalano, D. Fiore, B. Warinschi. Efficient Network Coding Signatures in the Standard Model. In *PKC'12*, *LNCS* 7293, pp. 680–696, 2012.
18. M. Chase, M. Kohlweiss, A. Lysyanskaya, S. Meiklejohn. Malleable Proof Systems and Applications. In *Eurocrypt'12*, *LNCS* 7237, pp. 281-300, 2012.
19. D. Freeman. Improved security for linearly homomorphic signatures: A generic framework. In *PKC'12*, *LNCS* 7293, pp. 697–714, 2012.
20. Y. Desmedt. Computer security by redefining what a computer is. In *New Security Paradigms Workshop (NSPW) 1993*, pp. 160–166, 1993.
21. G. Fuchsbauer. Commuting Signatures and Verifiable Encryption. In *Eurocrypt'11*, *LNCS* 6632, pp. 224-245, 2011.
22. R. Gennaro, J. Katz, H. Krawczyk, T. Rabin. Secure Network Coding over the Integers. In *PKC'10*, *LNCS* 6056, pp. 142–160, 2010.
23. M. Gerbush, A. Lewko, A. O'Neill, B. Waters. Dual Form Signatures: An Approach for Proving Security from Static Assumptions. Cryptology ePrint Archive: Report 2012/261, May 2012.
24. C. Gentry. Fully homomorphic encryption using ideal lattices. In *STOC'09*, pp. 169–178, 2009.
25. C. Gentry, S. Halevi, N. Smart. Fully Homomorphic Encryption with Polylog Overhead. In *Eurocrypt'12*, *LNCS* 7237, pp. 465–482, 2012.
26. J. Groth, A. Sahai. Efficient non-interactive proof systems for bilinear groups. In *Eurocrypt'08*, LNCS 4965, pp. 415–432, 2008.
27. S. Haber, Y. Hatano, Y. Honda, W. Horne, K. Miyazaki, T. Sander, S. Tezoku, D. Yao. Efficient signature schemes supporting redaction, pseudonymization, and data deidentification. In *AsiaCCS'08*, pp. 353–362, 2008.
28. A. Hevia, D. Micciancio. The Provable Security of Graph-Based One-Time Signatures and Extensions to Algebraic Signature Schemes. In *Asiacrypt'02*, LNCS 2501, pp. 379-396, 2002.
29. D. Hofheinz, E. Kiltz. Programmable Hash Functions and Their Applications. In *Crypto'08*, *LNCS* 5157, pp. 21–38, 2008.
30. E. Kiltz, A. Mityagin, S. Panjwani, B. Raghavan. Append-Only Signatures. In *ICALP'05*, LNCS 3580, pp. 434–445, 2005.
31. S. Kunz-Jacques, D. Pointcheval. About the Security of MTI/C0 and MQV. In *SCN'06*, LNCS 4116, pp. 156–172, 2006.

32. R. Johnson, D. Molnar, D. Song, D. Wagner. Homomorphic Signature Schemes. In *CT-RSA'02*, LNCS 2271, pp. 244–262, 2002.
33. B. Libert, D. Vergnaud. Multi-use unidirectional proxy re-signatures. In *ACM-CCS'08*, pp. 511–520, 2008.
34. S. Micali, R. Rivest. Transitive Signature Schemes. In *CT-RSA'02*, *LNCS* 2271, pp. 236-243, 2002.
35. K. Miyazaki, G. Hanaoka, H. Imai. Digitally signed document sanitizing scheme based on bilinear maps. In *AsiaCCS'06*, pp. 343-354, 2006.
36. M. van Dijk, C. Gentry, S. Halevi, V. Vaikuntanathan. Fully homomorphic encryption over the integers. In *Eurocrypt'10*, *LNCS* 6110, pp. 22–43, 2010.
37. B. Waters. Efficient Identity-Based Encryption Without Random Oracles. In *Eurocrypt'05*, *LNCS* 3494, pp. 114–127, 2005.
38. B. Waters. Dual System Encryption: Realizing Fully Secure IBE and HIBE under Simple Assumptions. In *Crypto'09*, *LNCS* series, 2009.

A Previous Definitions of Context-hiding Homomorphic Signatures

A.1 Strong Context-hiding Security

The following definition was given by Ahn *et al.* [4]

Definition 8. *A homomorphic signature* $(\mathsf{Keygen}, \mathsf{Sign}, \mathsf{SignDerive}, \mathsf{Verify})$ *is* **strongly context hiding** *for the predicate P if, for all key pairs $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Keygen}(\lambda)$, for all message sets $M \subset \mathcal{M}^*$ and $m' \in \mathcal{M}$ such that $P(M, m') = 1$, the following two distributions are statistically close:*

$$\Big\{(\mathsf{sk}, \{\sigma_m\}_{m\in M} \leftarrow \mathsf{Sign}(\mathsf{sk}, M),\ \mathsf{Sign}(sk, m'))\Big\}_{sk,M,m'},$$
$$\Big\{\big(\mathsf{sk}, \{\sigma_m\}_{m\in M} \leftarrow \mathsf{Sign}(\mathsf{sk}, M),\ \mathsf{SignDerive}\big(\mathsf{pk}, (\{\sigma_m\}_{m\in M}, M), m'\big)\big)\Big\}_{sk,M,m'}.$$

A.2 Weakly Context-Hiding Security

For completeness, this section recalls the definition of weakly context hiding homomorphic signatures of Boneh and Freeman [11].

Definition 9. *A P-homomorphic signature* $(\mathsf{Keygen}, \mathsf{Sign}, \mathsf{SignDerive}, \mathsf{Verify})$ *is* **weakly context hiding** *if no PPT adversary has non-negligible advantage in the following game:*

1. *The challenger runs $(\mathsf{sk}, \mathsf{pk}) \leftarrow \mathsf{Keygen}(\lambda)$ and gives $(\mathsf{sk}, \mathsf{pk})$ to the adversary.*
2. *The adversary $\mathcal{A}$ chooses two message sets $M_0, M_1 \subset \mathcal{M}$ and a message $m' \in \mathcal{M}$ such that $P(M_0, m') = P(M_1, m') = 1$. The challenger flips a fair coin $\beta \xleftarrow{R} \{0,1\}$ and computes $\{\sigma_m\}_{m\in M_\beta} \leftarrow \mathsf{Sign}(sk, M_\beta)$ as well as $\sigma^\star \leftarrow \mathsf{SignDerive}\big(\mathsf{pk}, (\{\sigma_m\}_{m\in M_\beta}, M_\beta), m'\big)$. In either case, $\sigma^\star$ is sent as a challenge to $\mathcal{A}$.*
3. *The adversary $\mathcal{A}$ outputs a bit $\beta' \in \{0,1\}$ and wins if $\beta' = \beta$. As always, $\mathcal{A}$'s advantage is measured as the distance $\mathbf{Adv}(\mathcal{A}) = |\Pr[\beta' = \beta] - 1/2|$.*

B Definitions for Linearly Homomorphic Signatures

In linearly homomorphic signatures, the message space $\mathcal{M}$ consists of pairs $\mathcal{M} := \mathcal{T} \times \mathbb{Z}_p^n$ for some positive integers p, n and where $\mathcal{T}$ is a tag space. The predicate P is defined in such a way that $P(\varepsilon, m) = 1$ for each $m \in \mathcal{M}$ and

$$P\Big(\{(\tau_1, \vec{v}_1), \ldots, (\tau_k, \vec{v}_k)\}, (\tau, \vec{v})\Big) = 1 \quad \Longleftrightarrow \quad \big(\tau = \tau_1 = \cdots = \tau_k\big) \wedge \big(\vec{v} \in \mathrm{span}(\vec{v}_1, \ldots, \vec{v}_k)\big)$$

In the context of linearly homomorphic signatures, the definitions can be specialized as in [10].

Definition 10. *A linearly homomorphic signature scheme* $\Sigma = (\mathsf{Keygen}, \mathsf{Sign}, \mathsf{SignDerive}, \mathsf{Verify})$ *is a tuple of efficient algorithms with the following specifications.*

Keygen(λ, 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* $(\mathsf{pk}, \mathsf{sk})$ *and the description of a tag (i.e., a file identifier) space* $\mathcal{T}$*.*

Sign$(\mathsf{sk}, \tau, \vec{v})$**:** *is a possibly probabilistic algorithm that takes as input a private key* sk*, a file identifier* $\tau \in \mathcal{T}$ *and a vector* $\vec{v}$*. It outputs a signature* σ*.*

SignDerive$(\mathsf{pk}, \tau, \{(\beta_i, \sigma^{(i)})\}_{i=1}^{\ell})$**:** *is a (possibly probabilistic) signature derivation algorithm. It takes as input a public key* pk*, a file identifier* τ *as well as* ℓ *pairs* $(\beta_i, \sigma^{(i)})$*, each of which consists of a weight* β_i *and a signature* $\sigma^{(i)}$*. The output is a signature* σ *on the vector* $\vec{y} = \sum_{i=1}^{\ell} \beta_i \vec{v}_i$*, where* $\sigma^{(i)}$ *is a signature on* $\vec{v}_i$*.*

Verify$(\mathsf{pk}, \tau, \vec{y}, \sigma)$**:** *is a deterministic algorithm that takes as input a public key* pk*, a file identifier* $\tau \in \mathcal{T}$*, a signature* σ *and a vector* $\vec{y}$*. It outputs* 1 *if the signature is deemed valid and* 0 *otherwise.*

Correctness is expressed by imposing that, for all security parameters $\lambda \in \mathbb{N}$, all integers $n \in \mathsf{poly}(\lambda)$ and all triples $(\mathsf{pk}, \mathsf{sk}, \mathcal{T}) \leftarrow \mathsf{Keygen}(\lambda, n)$, the following holds.

1. For all $\tau \in \mathcal{T}$ and all n-vectors $\vec{y}$, if $\sigma = \mathsf{Sign}(\mathsf{sk}, \tau, \vec{y})$, then we have $\mathsf{Verify}(\mathsf{pk}, \tau, \vec{y}, \sigma) = 1$.
2. For all $\tau \in \mathcal{T}$, any $\ell > 0$ and any set of triples $\{(\beta_i, \sigma^{(i)}, \vec{v}_i)\}_{i=1}^{\ell}$, if $\mathsf{Verify}(\mathsf{pk}, \tau, \vec{v}_i, \sigma^{(i)}) = 1$ for each $i \in \{1, \ldots, \ell\}$, then $\mathsf{Verify}\big(\mathsf{pk}, \tau, \sum_{i=1}^{\ell} \beta_i \vec{v}_i, \mathsf{SignDerive}(\mathsf{pk}, \tau, \{(\beta_i, \sigma^{(i)})\}_{i=1}^{\ell})\big) = 1$.

Security. In linearly homomorphic signatures, we can re-write Definition 3 as follows. We note that Definition 11 implies security in the stronger model used by Freeman [19] since the adversary can interleave signing queries for individual vectors belonging to distinct subspaces. Moreover, file identifiers can be chosen by the adversary (which strengthens the definition of [10]) and are not assumed to be uniformly distributed. As a result, a file identifier can be a low-entropy, easy-to-remember string such as the name of the dataset's owner.

Definition 11. *A linearly homomorphic signature scheme* $\Sigma = (\mathsf{Keygen}, \mathsf{Sign}, \mathsf{Verify})$ *is secure if no PPT adversary has non-negligible advantage (as a function of the security parameter* $\lambda \in \mathbb{N}$*) in the game below:*

1. *The adversary* $\mathcal{A}$ *chooses an integer* $n \in \mathbb{N}$ *and sends it to the challenger who runs* $\mathsf{Keygen}(\lambda, n)$ *and obtains* $(\mathsf{pk}, \mathsf{sk})$ *before sending* pk *to* $\mathcal{A}$*.*
2. *On polynomially-many occasions,* $\mathcal{A}$ *can interleave the following kinds of queries.*
   - *Signing queries:* $\mathcal{A}$ *chooses a tag* $\tau \in \mathcal{T}$ *and a vector* $\vec{v}$*. The challenger chooses a handle* h *and computes* $\sigma \leftarrow \mathsf{Sign}(\mathsf{sk}, \tau, \vec{v})$*. It stores* $(\mathsf{h}, (\tau, \vec{v}, \sigma))$ *in a table* T *and returns* h *to* $\mathcal{A}$*.*
   - *Derivation queries:* $\mathcal{A}$ *chooses a vector of handles* $\vec{\mathsf{h}} = (\mathsf{h}_1, \ldots, \mathsf{h}_k)$ *and a message* $(\tau, \vec{y}) \in \mathcal{M}$*. The challenger retrieves the tuples* $\{(\mathsf{h}_i, (\tau, \vec{v}_i), \sigma^{(i)})\}_{i=1}^{k}$ *from* T *and returns* $\perp$ *if one of these does not exist or if there exists* $i \in \{1, \ldots, k\}$ *such that* $\tau_i \neq \tau$*. Otherwise, it defines the set* $M := \big((\tau, \vec{v}_1), \ldots, (\tau, \vec{v}_k)\big)$*. If* $\vec{y} \notin \mathrm{span}(\vec{v}_1, \ldots, \vec{v}_k)$*, the challenger returns* $\perp$*. Otherwise, it determines* $\beta_1, \ldots, \beta_k$ *such that* $\vec{y} = \sum_{i=1}^{k} \beta_i \vec{v}_i$ *and runs* $\sigma' \leftarrow \mathsf{SignDerive}\big(\mathsf{pk}, \{(\beta_i, \sigma^{(i)})\}_{i=1}^{k}, \vec{y}\big)$*, chooses a handle* h'*, stores* $(\mathsf{h}', (\tau, \vec{y}), \sigma')$ *in* T *and returns* h' *to* $\mathcal{A}$*.*
   - *Reveal queries:* $\mathcal{A}$ *chooses a handle* h*. If no tuple of the form* $(\mathsf{h}, (\tau, \vec{v}), \sigma')$ *exists in* T*, the challenger returns* $\perp$*. Otherwise, it returns* σ' *to* $\mathcal{A}$ *and adds* $((\tau, \vec{v}), \sigma')$ *to the set* Q*.*

3. *$\mathcal{A}$ outputs an identifier $\tau^\star$, a signature $\sigma^\star$ and a vector $\vec{y} \in \mathbb{Z}_N^n$. The adversary $\mathcal{A}$ wins if* $\mathsf{Verify}(\mathsf{pk}, \tau^\star, \vec{y}^\star, \sigma^\star) = 1$ *and one of the conditions below is satisfied:*
   - *(Class I): $\tau^\star \neq \tau_i$ for any entry $(\vec{\tau}_i, .)$ in Q and $\vec{y}^\star \neq \vec{0}$.*
   - *(Class II): $\tau^\star = \tau_i$ for $k_i > 0$ entries $(\tau_i, .)$ in Q and $\vec{y}^\star \notin V_i$, where V_i denotes the subspace spanned by all vectors $\vec{v}_1, \ldots, \vec{v}_{k_i}$ (which we assume to be linearly independent) for which an entry of the form $(\tau^\star, \vec{v}_j)$, with $j \in \{1, \ldots, k_i\}$, appears in Q.*

$\mathcal{A}$'s advantage is its probability of success taken over all coin tosses.

It is assumed that, for each subspace V_i, the adversary only queries linearly independent vectors $\vec{v}_1, \ldots, \vec{v}_{k_i}$. This is not a limitation since, in practical applications like network coding, one usually appends a unit vector to signed vectors so as to guarantee their independence.

Ahn *et al.* [4] provided evidence that, when the scheme is strongly context-hiding in the sense of Definition 8, the definition of unforgeability can be simplified by only giving the adversary access to a signing oracle. In this case, the definition goes as follows.

Definition 12. *A linearly homomorphic signature scheme $\Sigma = (\mathsf{Keygen}, \mathsf{Sign}, \mathsf{Verify})$ is unforgeable if no PPT adversary has non-negligible advantage (as a function of the security parameter $\lambda \in \mathbb{N}$) in the following game:*

1. *The adversary $\mathcal{A}$ chooses an integer $n \in \mathbb{N}$ and sends it to the challenger. The latter runs* $\mathsf{Keygen}(\lambda, n)$ *and obtains a pair* $(\mathsf{pk}, \mathsf{sk})$ *before sending* pk *to $\mathcal{A}$.*
2. *On polynomially many occasions, the adversary $\mathcal{A}$ chooses a tag $\tau \in \mathcal{T}$ and a vector $\vec{v}$. The challenger returns $\sigma = \mathsf{Sign}(\mathsf{sk}, \tau, \vec{v})$ to $\mathcal{A}$.*
3. *The adversary $\mathcal{A}$ outputs an identifier $\tau^\star$, a signature $\sigma^\star$ and a vector $\vec{y} \in \mathbb{Z}_N^n$. Then, $\mathcal{A}$ is declared successful if* $\mathsf{Verify}(\mathsf{pk}, \tau^\star, \vec{y}^\star, \sigma^\star) = 1$ *and one of the following conditions holds:*
   - *(Class I): $\tau^\star \neq \tau_i$ for any i and $\vec{y}^\star \neq \vec{0}$.*
   - *(Class II): $\tau^\star = \tau_i$ for some $i \in \{1, \ldots, q\}$ and $\vec{y}^\star \notin V_i$, where V_i denotes the subspace spanned by all vectors $\vec{v}_1, \ldots, \vec{v}_{k_i}$ that have been queried for τ_i.*

C Review of the ALP CDH-Based Linearly Homomorphic Signature

The following scheme was described by Attrapadung *et al.* in [6]. It can be seen as a combination of Waters signatures [37] with a randomized vector hash system [3].

$\mathsf{Keygen}(\lambda, n)$: given a security parameter $\lambda \in \mathbb{N}$ and an integer $n \in \mathsf{poly}(\lambda)$, choose bilinear groups $(\mathbb{G}, \mathbb{G}_T)$ of prime order $p > 2^\lambda$. Choose $\alpha \xleftarrow{R} \mathbb{Z}_p$, $g, v \xleftarrow{R} \mathbb{G}$ and $u_0, u_1, \ldots, u_L \xleftarrow{R} \mathbb{G}$, for some $L \in \mathsf{poly}(\lambda)$. These elements $(u_0, \ldots, u_L) \in \mathbb{G}^{L+1}$ will be used to implement a programmable hash function $H_{\mathbb{G}} : \{0,1\}^L \to \mathbb{G}$ such that any L-bit string $m = m[1] \ldots m[L] \in \{0,1\}^L$ has a hash value $H_{\mathbb{G}}(m) = u_0 \cdot \prod_{i=1}^{L} u_i^{m[i]}$. Pick $g_i \xleftarrow{R} \mathbb{G}$ for $i = 1$ to n. Finally, define the identifier space $\mathcal{T} := \{0,1\}^L$. The private key is $\mathsf{sk} := \alpha$ and the public key consists of

$$\mathsf{pk} := \Big((\mathbb{G}, \mathbb{G}_T),\ g,\ g^\alpha,\ v,\ \{g_i\}_{i=1}^n,\ \{u_i\}_{i=0}^L \Big).$$

$\mathsf{Sign}(\mathsf{sk}, \tau, \vec{v})$: given a vector $\vec{v} = (v_1, \ldots, v_n) \in \mathbb{Z}_p^n$, a file identifier $\tau \in \{0,1\}^L$ and the private key $\mathsf{sk} = \alpha \in \mathbb{Z}_p$, choose $r, s \xleftarrow{R} \mathbb{Z}_p$. Then, compute a signature $\sigma = (\sigma_1, \sigma_2, s) \in \mathbb{G}^2 \times \mathbb{Z}_p$ as

$$\sigma_1 = (g_1^{v_1} \cdots g_n^{v_n} \cdot v^s)^\alpha \cdot H_{\mathbb{G}}(\tau)^r, \qquad \sigma_2 = g^r.$$

SignDerive$(\mathsf{pk}, \tau, \{(\beta_i, \sigma^{(i)})\}_{i=1}^{\ell})$**:** given pk, a file identifier τ and ℓ tuples (β_i, σ_i), parse each $\sigma^{(i)}$ as $\sigma^{(i)} = (\sigma_{i,1}, \sigma_{i,2}, s_i)$ for $i = 1$ to ℓ. Choose $\tilde{r} \xleftarrow{R} \mathbb{Z}_p$. Then, compute and output (σ_1, σ_2, s), where

$$\sigma_1 = \prod_{i=1}^{\ell} \sigma_{i,1}^{\beta_i} \cdot H_{\mathbb{G}}(\tau)^{\tilde{r}} \qquad \sigma_2 = \prod_{i=1}^{\ell} \sigma_{i,2}^{\beta_i} \cdot g^{\tilde{r}} \qquad s = \sum_{i=1}^{\ell} \beta_i \cdot s_i$$

Verify$(\mathsf{pk}, \tau, \vec{y}, \sigma)$**:** given pk, a signature $\sigma = (\sigma_1, \sigma_2, s)$ and a message $(\tau, \vec{y})$, where $\tau \in \{0,1\}^L$ and $\vec{y} = (y_1, \ldots, y_n) \in (\mathbb{Z}_p)^n$, return $\perp$ if[3] $\vec{y} = \vec{0}$. Otherwise, return 1 if and only if

$$e(\sigma_1, g) = e(g_1^{y_1} \cdots g_n^{y_n} \cdot v^s, g^{\alpha}) \cdot e(H_{\mathbb{G}}(\tau), \sigma_2). \tag{11}$$

In [6], the above scheme was proved unforgeable under the computational Diffie-Hellman assumption.

D Proof of Theorem 4

Since the scheme is completely context hiding, we only need to consider the simplified definition of unforgeability where the adversary is only given access to a signing oracle. The proof proceeds with a sequence of two games where, for each $i \in \{0,1\}$, we denote by S_i the event that the adversary $\mathcal{A}$ manages to come up with a valid forgery at the end of Game_i.

Game_0: This is the actual game. We call S_0 the event that the forger $\mathcal{A}$ produces a successful forgery.

Game_1: In this game, we change the distribution of the public key and set up $\mathbf{f} = (\vec{f}_1, \vec{f}_2, \vec{f}_3)$ as a perfectly sound Groth-Sahai common reference string. Concretely, the challenger $\mathcal{B}$ sets up $\mathbf{f}$ as $\vec{f}_1 = (f_1, 1, g)$, $\vec{f}_2 = (1, f_2, g)$ and $\vec{f}_3 = \vec{f}_1^{\,\xi_1} \cdot \vec{f}_2^{\,\xi_2}$, with $f_1 = g^{\phi_1}$ and $f_2 = g^{\phi_2}$, with $\phi_1, \phi_2, \xi_1, \xi_2 \xleftarrow{R} \mathbb{Z}_p$. Clearly, under the DLIN assumption, this should not significantly affect $\mathcal{A}$'s view. Specifically, assuming that $\mathcal{A}$ outputs a forgery with significantly higher probability in Game_1 than in Game_0, we can turn $\mathcal{B}$ into a DLIN distinguisher such that $|\Pr[S_1] - \Pr[S_0]| \leq \mathbf{Adv}^{\text{DLIN}}(\mathcal{B})$. We remark that, in Game_1 and in later games, all Groth-Sahai commitments are perfectly binding commitments.

In Game 1, we claim that $\mathcal{B}$ can break the FlexDH assumption if event S_1 occurs with non-negligible probability. To this end, $\mathcal{B}$ makes use of the BBS decryption keys $(\phi_1, \phi_2) = (\log_g(f_1), \log_g(f_2))$ that were defined in Game_1. When the adversary $\mathcal{A}$ outputs a forgery $(\tau^\star, \vec{y}^\star, \sigma^\star)$, $\mathcal{B}$ parses $\sigma^\star$ as $\big(\vec{C}^\star_{\sigma_1}, \sigma_2^\star, \vec{C}^\star_{\sigma_3}, \vec{C}^\star_{\sigma_4}, \vec{\pi}_1^\star, \vec{\pi}_2^\star\big)$ and uses (ϕ_1, ϕ_2) to extract $(\sigma_1^\star, \sigma_3^\star, \sigma_4^\star)$ from $\big(\vec{C}^\star_{\sigma_1}, \vec{C}^\star_{\sigma_3}, \vec{C}^\star_{\sigma_4}\big)$ (recall that, from Game_1 onwards, all Groth-Sahai commitments are extractable). Note that the underlying $(\sigma_1^\star, \sigma_2^\star, \sigma_3^\star, \sigma_4^\star)$ necessarily satisfy the relations (7)-(8) since $\pi_1^\star, \pi_2^\star$ are perfectly sound proofs. We now define $S_{1.1}$ (resp. $S_{1.2}$) to be the event that $(\tau^\star, \vec{y}^\star, \sigma^\star)$ is a Class I (resp. Class II) forgery, according to the distinction of Definition 12. Below, we prove that event $S_{1.1}$ (resp. $S_{1.2}$) occurs with negligible probability if the Diffie-Hellman assumption (resp. the Flexible Diffie-Hellman assumption) holds. Specifically, Lemmas 1 and 2 establish the inequality

$$\Pr[S_1] \leq \Pr[S_{1.1}] + \Pr[S_{1.2}] \leq 16 \cdot q \cdot (L+1) \cdot \Big(\mathbf{Adv}^{\text{FlexDH}}(\mathcal{B}) + \frac{1}{p}\Big).$$

[3] In the construction, we disallow signatures on the all-zeroes vector $\vec{0}$. This is not a restriction since, in all applications of linearly homomorphic signatures, a unit vector $(0, \ldots, 1, \ldots, 0)$ of appropriate length is appended to signed vectors.

When putting the above altogether, we eventually obtain

$$\Pr[S_0] \leq \mathbf{Adv}^{\mathrm{DLIN}}(\mathcal{B}) + 16 \cdot q \cdot (L+1) \cdot \Big(\mathbf{Adv}^{\mathrm{FlexDH}}(\mathcal{B}) + \frac{1}{p}\Big).$$

□

Lemma 1. *In Game$_1$, any Class I forger $\mathcal{A}$ implies a forger $\mathcal{B}$ against Waters signatures. As a consequence, any Class I forger $\mathcal{A}$ can be turned into a computational Diffie-Hellman solver $\mathcal{B}$ for which $\mathbf{Adv}(\mathcal{A}) \leq 8 \cdot q \cdot (L+1) \cdot \big(\mathbf{Adv}^{\mathrm{CDH}}(\mathcal{B}) + \frac{1}{p}\big)$, where q is the number of distinct tags appearing in signing queries.*

Proof. It is easy to see that a Class I forger $\mathcal{A}$ implies an equally powerful forger $\mathcal{B}$ against Waters signatures. Recall that Waters signatures involve a public key containing group elements $(u_0, \ldots, u_L) \in \mathbb{G}^{L+1}$ and $(g, g^\alpha, h) \in \mathbb{G}^3$. The private key $\alpha \in \mathbb{Z}_p$ allows signing L-bit messages $m \in \{0,1\}^L$ using signatures of the form $(\sigma_1, \sigma_2) = (h^\alpha \cdot H_{\mathbb{G}}(m)^r, g^r)$, with $r \xleftarrow{R} \mathbb{Z}_p$ and $H_{\mathbb{G}}(m) = u_0 \cdot \prod_{i=1}^{L} u_i^{m[i]}$.

Algorithm $\mathcal{B}$ is given a Waters public key $(g, g^\alpha, h, \{u_i\}_{i=0}^L)$. To create a public key for the linearly homomorphic scheme, it first sets $\vec{f}_1 = (f_1, 1, g)$, $\vec{f}_2 = (1, f_2, g)$ and $\vec{f}_3 = \vec{f}_1^{\,\xi_1} \cdot \vec{f}_2^{\,\xi_2}$, where $(f_1, f_2) = (g^{\phi_1}, g^{\phi_2})$ and $\phi_1, \phi_2 \xleftarrow{R} \mathbb{Z}_p$. Note that $\mathbf{f} = (\vec{f}_1, \vec{f}_2, \vec{f}_3)$ forms a perfectly sound Groth-Sahai CRS as should be the case in Game$_2$. Then, $\mathcal{B}$ defines $(g_1, \ldots, g_n, v)$ by setting $v = g^{\gamma_v}$ and $g_i = h^{\gamma_i} \cdot g^{\delta_i}$, for $i \in \{1, \ldots, n\}$, using randomly chosen $\gamma_v, \gamma_1, \ldots, \gamma_n \xleftarrow{R} \mathbb{Z}_p$ and $\delta_1, \ldots, \delta_n \xleftarrow{R} \mathbb{Z}_p$. The linearly homomorphic adversary $\mathcal{A}$ receives $\mathsf{pk} = \big((\mathbb{G}, \mathbb{G}_T), g, g^\alpha, v, \{g_i\}_{i=1}^n, \{u_i\}_{i=0}^L, \mathbf{f}\big)$.

For each signing query $(\tau, \vec{v} = (v_1, \ldots, v_n))$, our adversary $\mathcal{B}$ invokes its own signing oracle and asks for a Waters signature on the message $\tau = \tau[1] \ldots \tau[1] \in \{0,1\}^L$. The oracle replies by returning a pair $(\tilde{\sigma_1}, \tilde{\sigma_2}) = (h^\alpha \cdot H_{\mathbb{G}}(\tau)^r, g^r)$, which $\mathcal{B}$ transforms into a linearly homomorphic signature. To do this, $\mathcal{B}$ picks $s, r' \xleftarrow{R} \mathbb{Z}_p$, computes

$$(\sigma_1, \sigma_2, \sigma_3, \sigma_4) = \Big(\tilde{\sigma_1}^{\sum_{i=1}^n \gamma_i \cdot v_i} \cdot (g^\alpha)^{s \cdot \gamma_v + \sum_{i=1}^n \delta_i \cdot v_i} \cdot H_{\mathbb{G}}(\tau)^{r'},\ \tilde{\sigma_2}^{\sum_{i=1}^n \gamma_i \cdot v_i} \cdot g^{r'},\ g^s,\ (g^\alpha)^s\Big).$$

Next, $\mathcal{B}$ generates Groth-Sahai commitments $\vec{C}_{\sigma_1}$, $\vec{C}_{\sigma_3}$ and $\vec{C}_{\sigma_4}$ to $(\sigma_1, \sigma_3, \sigma_4)$ and NIWI proofs $\vec{\pi}_1, \vec{\pi}_2$ that they satisfy relations (7)-(8). It is easy to see that $\sigma = (\vec{C}_{\sigma_1}, \sigma_2, \vec{C}_{\sigma_3}, \vec{C}_{\sigma_4}, \vec{\pi}_1, \vec{\pi}_2)$ forms a valid signature on $(\tau, \vec{v})$ in Game$_1$.

Eventually, the Class I adversary $\mathcal{A}$ outputs a valid signature $(\vec{C}^\star_{\sigma_1}, \sigma_2^\star, \vec{C}^\star_{\sigma_3}, \vec{C}^\star_{\sigma_4}, \vec{\pi}_1^\star, \vec{\pi}_2^\star)$ on a message $(\tau^\star, \vec{v}^\star = (v_1^\star, \ldots, v_n^\star))$. From $(\vec{C}^\star_{\sigma_1}, \vec{C}^\star_{\sigma_3}, \vec{C}^\star_{\sigma_4})$, $\mathcal{B}$ extracts $(\sigma_1^\star, \sigma_3^\star, \sigma_4^\star)$ which, thanks to the perfect soundness of $(\pi_1^\star, \pi_2^\star)$, are guaranteed to satisfy (7)-(8). Since $\mathcal{A}$ is a Class I forger, we know that $\tau^\star$ did not appear in any signing query. For this reason, $\mathcal{B}$ can forge a Waters signature by computing

$$(\sigma_1, \sigma_2) = \Big(\big(\frac{\sigma_1^\star}{\sigma_4^{\star\gamma_v} \cdot (g^\alpha)^{\sum_{i=1}^n \delta_i \cdot v_i^\star}}\big)^{1/\sum_{i=1}^n \gamma_i v_i^\star}, \sigma_2^{\star\, 1/\sum_{i=1}^n \gamma_i v_i^\star}\Big)$$

provided $\sum_{i=1}^n \gamma_i v_i^\star \neq 0$. However, since $\{\gamma_i\}_{i=1}^n$ are independent of $\mathcal{A}$'s view and $\vec{v}^\star \neq \vec{0}$, we can only have $\sum_{i=1}^n \gamma_i v_i^\star = 0$ with negligible probability $1/p$. □

Lemma 2. *In Game$_1$, for any Class II forger $\mathcal{A}$, there is an algorithm $\mathcal{B}$ solving the FlexDH problem such that $\mathbf{Adv}(\mathcal{A}) \leq 8 \cdot q \cdot (L+1) \cdot \big(\mathbf{Adv}^{\mathrm{FlexDH}}(\mathcal{B}) + \frac{1}{p}\big)$, where q is the number of distinct tags appearing in signing queries.*

Proof. From a Class II forger $\mathcal{A}$ with advantage ε, we construct an algorithm $\mathcal{B}$ that solves a FlexDH instance (g, g^a, g^b) with probability about $\varepsilon/(8q(L+1))$.

Algorithm $\mathcal{B}$ prepares the public key by choosing $(u_0, u_1, \ldots, u_L) \in \mathbb{G}^{L+1}$ so as to define a "programmable" hash function [29] as in the security proof of Waters' signatures [37]. For any identifier $\tau \in \{0,1\}^L$, $H_{\mathbb{G}}(\tau)$ is linked to integer-valued functions $J, K : \{0,1\}^L \to \mathbb{Z}_p$ such that $H_{\mathbb{G}}(\tau) = (g^a)^{J(\tau)} \cdot g^{K(\tau)}$.

Remaining public key components are obtained by setting $\mathbf{f} = (\vec{f}_1, \vec{f}_2, \vec{f}_3)$ as an extractable Groth-Sahai CRS, as in the proof of Lemma 1, and defining $g^\alpha = g^a$, $v = g^b$ and $g_i = (g^b)^{\gamma_i} \cdot g^{\rho_i}$ with $\gamma_i, \rho_i \xleftarrow{R} \mathbb{Z}_p$ for each $i \in \{1, \ldots, n\}$.

The Class II forger $\mathcal{A}$ comes up with a forgery $(\tau^\star, \vec{y}^\star, \sigma^\star)$ for a file identifier $\tau^\star$ that appeared in some signing query but for which $\vec{y}^\star \notin \mathrm{span}(\vec{v}_1, \ldots, \vec{v}_{n-1})$, where $\vec{v}_1, \ldots, \vec{v}_{n-1}$ are the vectors queried for $\tau^\star$ (we assume w.l.o.g. that $\mathcal{A}$ makes exactly $n-1$ signing queries). We denote by $\tau_1, \ldots, \tau_q$ the distinct file identifiers successively appearing in $\mathcal{A}$'s queries throughout the game. During the game, $\mathcal{A}$'s view is simulated as follows.

Signing queries: at each signing query $\big(\tau_{l_1}, \vec{v} = (v_1, \ldots, v_n)\big)$ involving the l_1-th distinct tag τ_{l_1}, $\mathcal{B}$ evaluates the function $J(\tau_{l_1})$ and considers two distinct cases.

- If $J(\tau_{l_1}) \neq 0 \bmod p$, $\mathcal{B}$ picks $r, s \xleftarrow{R} \mathbb{Z}_p$ and computes

$$\sigma_1 = H_{\mathbb{G}}(\tau_{l_1})^r \cdot (g^b)^{-\frac{K(\tau_{l_1})}{J(\tau_{l_1})} \cdot (\langle \vec{\gamma}, \vec{v} \rangle + s)} \cdot (g^a)^{\langle \vec{\rho}, \vec{v} \rangle}, \qquad \sigma_2 = g^r \cdot (g^b)^{-\frac{\langle \vec{\gamma}, \vec{v} \rangle + s}{J(\tau_{l_1})}}. \tag{12}$$

  Note that σ_1 and σ_2 have the correct distribution since, if we define $\tilde{r} = r - \frac{b \cdot (\langle \vec{\gamma}, \vec{v} \rangle + s)}{J(\tau_{l_1})}$, we have

$$(\sigma_1, \sigma_2) = \Big((\prod_{i=1}^{n} g_i^{v_i} \cdot v^s)^a \cdot H_{\mathbb{G}}(\tau_{l_1})^{\tilde{r}},\ g^{\tilde{r}} \Big).$$

  The signature generation is completed by computing $(\sigma_3, \sigma_4) = (g^s, g^{\alpha \cdot s})$ as well as commitments $\vec{C}_{\sigma_1}, \vec{C}_{\sigma_3}, \vec{C}_{\sigma_4}$ to $(\sigma_1, \sigma_3, \sigma_4)$ and returning $(\vec{C}_{\sigma_1}, \sigma_2, \vec{C}_{\sigma_3}, \vec{C}_{\sigma_4}, \vec{\pi}_1, \vec{\pi}_2)$, where $\vec{\pi}_1, \vec{\pi}_2$ are the NIWI proofs for equations (7)-(8).
- If $J(\tau_{l_1}) = 0 \bmod p$, $\mathcal{B}$ aborts in the event that the function J already canceled for a tag $\tau_j \neq \tau_{l_1}$ involved in a previous signing query. Otherwise (*i.e.*, if τ_{l_1} is the first queried tag for which $J(\tau_{l_1}) = 0$), $\mathcal{B}$ sets $s = -\langle \vec{\gamma}, \vec{v} \rangle$. Then, $\mathcal{B}$ picks $r \xleftarrow{R} \mathbb{Z}_p$ and computes

$$(\sigma_1, \sigma_2, \sigma_3, \sigma_4) = \Big((g^a)^{\langle \vec{\rho}, \vec{v} \rangle} \cdot H_{\mathbb{G}}(\tau_{l_1})^r,\ g^r,\ g^s,\ g^{a \cdot s} \Big).$$

  Note that $\mathcal{A}$ can make signing queries of the form $(\tau_{j^\star}, \vec{v}_{l_2})$ for at most $n-1$ linearly independent vectors $\vec{v}_{l_2}$. This means that it will get to see at most $n-1$ independent values $s_1, \ldots, s_{n-1} \in \mathbb{Z}_p$ such that $s_{l_2} = -\langle \vec{\gamma}, \vec{v}_{l_2} \rangle$ for each $l_2 \in \{1, \ldots, n-1\}$. Since $\mathcal{A}$ has initially no information about $\vec{\gamma}$, from $\mathcal{A}$'s view, the values $\{s_{l_2}\}_{l_2=1}^{n-1}$ are statistically indistinguishable from uniformly random values. The query is answered by returning $(\vec{C}_{\sigma_1}, \sigma_2, \vec{C}_{\sigma_3}, \vec{C}_{\sigma_4}, \vec{\pi}_1, \vec{\pi}_2)$, where $\vec{C}_{\sigma_1}, \vec{C}_{\sigma_3}, \vec{C}_{\sigma_4}$ are Groth-Sahai commitments to $(\sigma_1, \sigma_3, \sigma_4)$ and $\vec{\pi}_1, \vec{\pi}_2$ are the corresponding NIWI proofs.

Forgery: when $\mathcal{A}$ terminates, it outputs a Class II forgery $(\tau^\star, \vec{y}^\star, \sigma^\star)$, where $\vec{y}^\star = (y_1^\star, \ldots, y_n^\star)$ and $\sigma^\star = (\vec{C}_{\sigma_1}^\star, \sigma_2^\star, \vec{C}_{\sigma_3}^\star, \vec{C}_{\sigma_4}^\star, \vec{\pi}_1^\star, \vec{\pi}_2^\star)$ satisfies the verification equations (9)-(10).

At this point, $\mathcal{B}$ evaluates $J(\tau^\star)$ and reports failure if $J(\tau^\star) \neq 0$ or if the set $\{\tau_1, \ldots, \tau_q\}$

contains at least two tags τ_{j_1}, τ_{j_2} such that $J(\tau_{j_1}) = J(\tau_{j_2}) = 0$. The same analysis as in [37] shows that, with probability $1/(8(q-1)(L+1))$, we have $J(\tau^\star) = 0$ and $J(\tau_j) \neq 0$ for each $\tau_j \in \{\tau_1, \ldots, \tau_q\} \backslash \{\tau^\star\}$. We thus find that $\mathcal{B}$'s probability not to abort during the entire game is at least $1/(8(q-1)(L+1)) > 1/(8q(L+1))$.

We henceforth assume that $\mathcal{B}$ does not abort. It thus extracts $(\sigma_1^\star, \sigma_3^\star, \sigma_4^\star)$ from commitments $(\vec{C}_{\sigma_1}^\star, \vec{C}_{\sigma_3}^\star, \vec{C}_{\sigma_4}^\star)$. Since equations (9-(10) are satisfied, the perfect soundness of $(\pi_1^\star, \pi_2^\star)$ guarantees that $(\sigma_1^\star, \sigma_2^\star, \sigma_3^\star, \sigma_4^\star)$ are of the form

$$(\sigma_1^\star, \sigma_2^\star, \sigma_3^\star, \sigma_4^\star) = \Big((\prod_{i=1}^{n} g_i^{y_i^\star} \cdot v^{s^\star})^a \cdot H_{\mathbb{G}}(\tau^\star)^r,\ g^r,\ g^{s^\star},\ g^{a \cdot s^\star} \Big),$$

for some $r, s^\star \in \mathbb{Z}_p$.

As long as $\mathcal{B}$ does not fail, we necessarily have $H_{\mathbb{G}}(\tau^\star) = g^{K(\tau^\star)}$, so that $\mathcal{B}$ can compute

$$\begin{aligned}
\eta_1^\star &:= \sigma_1^\star \cdot \sigma_2^{\star -K(\tau^\star)} \cdot (g^a)^{-\langle \vec{\rho}, \vec{y}^\star \rangle} = (\prod_{i=1}^{n} g_i^{y_i^\star} \cdot v^{s^\star})^a \cdot g^{-a \cdot \langle \vec{\rho}, \vec{y}^\star \rangle} = (g^{ab})^{\langle \vec{\gamma}, \vec{y}^\star \rangle + s^\star} \\
\eta_2^\star &:= \sigma_4^\star \cdot (g^a)^{\langle \vec{\gamma}, \vec{y}^\star \rangle} = (g^a)^{\langle \vec{\gamma}, \vec{y}^\star \rangle + s^\star} \\
\eta_3^\star &:= \sigma_3^\star \cdot g^{\langle \vec{\gamma}, \vec{y}^\star \rangle} = g^{\langle \vec{\gamma}, \vec{y}^\star \rangle + s^\star}.
\end{aligned}$$

Provided $s^\star \neq -\langle \vec{\gamma}, \vec{y}^\star \rangle$, the triple $(\eta_1^\star, \eta_2^\star, \eta_3^\star)$ forms a valid solution to the given FlexDH instance. We are thus left with arguing that $s^\star \neq -\langle \vec{\gamma}, \vec{y}^\star \rangle$ with high probability. By hypothesis, we know that $\vec{y}^\star \notin \mathrm{span}(\vec{v}_1, \ldots, \vec{v}_{n-1})$. During the game, $\mathcal{A}$ can observe at most $n-1$ encrypted inner products of the form $\{g^{-\langle \vec{\gamma}, \vec{v}_{l_2} \rangle}\}_{l_2=1}^{n-1}$. The value $\langle \vec{\gamma}, \vec{y}^\star \rangle$ is thus totally independent of $\mathcal{A}$'s view. Since $\vec{\gamma}$ was chosen uniformly in $\mathbb{Z}_p^n$, we can only have $s^\star = -\langle \vec{\gamma}, \vec{y}^\star \rangle$ with probability $1/p$. $\square$