Short Structure-Preserving Signatures

Essam Ghadafi(✉)

University College London, London, UK
e.ghadafi@ucl.ac.uk

Abstract. We construct a new structure-preserving signature scheme in the efficient Type-III asymmetric bilinear group setting with signatures shorter than all existing schemes. Our signatures consist of 3 group elements from the first source group and therefore they are shorter than those of existing schemes as existing ones have at least one component in the second source group whose elements bit size is at least double that of their first group counterparts.

Besides enjoying short signatures, our scheme is fully re-randomizable which is a useful property for many applications. Our result also constitutes a proof that the impossibility of unilateral structure-preserving signatures in the Type-III setting result of Abe et al. (Crypto 2011) does not apply to constructions in which the message space is dual in both source groups. Besides checking the well-formedness of the message, verifying a signature in our scheme requires checking 2 Pairing Product Equations (PPE) and require the evaluation of only 5 pairings in total which matches the best existing scheme and outperforms many other existing ones. We give some examples of how using our scheme instead of existing ones improves the efficiency of some existing cryptographic protocols such as direct anonymous attestation and group signature related constructions.

Keywords: Structure-preserving · Digital signatures · Bilinear groups

1 Introduction

Structure-Preserving Signatures (SPS) [3] are digital signature schemes defined over bilinear groups ($e : \mathbb{G} \times \tilde{\mathbb{G}} \rightarrow \mathbb{T}$). Their messages, verification key and signatures are all group elements and signature verification involves evaluating Pairing Product Equations (PPE). They are a useful tool for the design of modular cryptographic protocols since they compose nicely with existing popular tools such as Groth-Sahai proofs [31] and ElGamal encryption scheme [20]. They are prominently used in combination with Groth-Sahai proofs and other tools to design cryptographic protocols that do not rely on heuristic assumptions such

The research leading to these results has received funding from the European Research Council under the European Union's Seventh Framework Programme (FP/2007–2013) / ERC Grant Agreement no. 307937 and EPSRC grant EP/J009520/1.

K. Sako (Ed.): CT-RSA 2016, LNCS 9610, pp. 305–321, 2016.
DOI: 10.1007/978-3-319-29485-8_18

as random oracles [21]. They have numerous applications which include group signatures, e.g. [3,34,35], blind signatures, e.g. [3,23], tightly secure encryption schemes, e.g. [2,32], malleable signatures, e.g. [9], anonymous credentials, e.g. [23], network coding, e.g. [9], oblivious transfer, e.g. [28].

Related Work. The notion was formally defined by Abe et al. [3] but earlier schemes conforming to the definition were given by Groth [29] and Green and Hohenberger [28]. Because of its importance, the notion has received a significant amount of attention from the cryptographic community and many results relating to proving lower bounds for the design of such schemes as well as new schemes meeting those lower bounds have been published in the literature. Abe et al. [3] gave two constructions of structure-preserving signatures both relying on non-interactive intractability assumptions. Abe et al. [4] proved that any structure-preserving signature scheme in the most efficient Type-III bilinear group setting (cf. Sect. 2.1) must have at least 3 group elements and 2 pairing product verification equations. They also ruled out the existence of unilateral signatures and argued that the signature must contain elements from both source groups. They also gave constructions meeting the lower bound and proved them secure in the generic group model [40]. Abe et al. [5] proved the impossibility of the existence of a 3 group element structure-preserving signature in the Type-III setting that is based on non-interactive intractability assumptions. In essence, their result implies that in the Type-III setting, the only way to meet the 3 group element lower bound is to either employ interactive intractability assumptions or resort to direct proofs in the generic group model. Ghadafi [25] gave a structure-preserving variant of the Camenisch-Lysyanskaya signature scheme [15] that is secure under an interactive assumption in the Type-III setting. Abe et al. [7] constructed a scheme in the Type-II setting (where there is an efficiently computable isomorphism from the second source group to the first) which contains only 2 group elements. Chatterjee and Menezes [17] revisited the work of [7] and showed that Type-III constructions outperform their Type-II counterparts [17] also gave constructions in Type-III setting meeting the 3 group element lower bound. Barthe et al. [10] also gave optimal constructions of structure-preserving signatures in Type-II setting. Constructions relying on standard assumptions (such as DLIN and DDH) were given by [1,2,14,16,33,35]. Constructions based on standard assumptions are less efficient than those based on non-standard assumptions or proven directly in the generic group model. Recently, Abe et al. [8] and Groth [30] gave fully structure-preserving constructions where even the secret key consists of only group elements.

While by now there exist a number of schemes, e.g. [4,6,10,17,30], with signatures meeting the 3 group element lower bound in the Type-III setting proved by Abe et al. [4], all those schemes have at least one component of the signature in group $\tilde{\mathbb{G}}$ whose elements bit size is at least double that of those in $\mathbb{G}$. To the best of our knowledge, the only existing structure-preserving signature scheme in the Type-III setting whose all signature components are in $\mathbb{G}$ is that

of Ghadafi [25]. However, signatures of latter consist of 4 group elements and require 3 pairing-product verification equations.

Our Contribution. We construct a (unilateral) structure-preserving signature scheme with signatures shorter than all existing structure-preserving signatures. Our scheme yields fully re-randomizable signatures consisting of 3 group elements from the first short source group.

Our results also serve as a proof that the impossibility of unilateral structure-preserving signature schemes in the Type-III setting result of Abe et al. [4] does not apply when the message space is dual in both source groups. We stress that Abe et al. never claimed that their Type-III lower bounds apply to this setting since their proofs only considered schemes with unilateral messages. As is the tradition with most existing structure-preserving schemes, we prove the security of our scheme directly in the generic group model. Our scheme can be viewed as an extension of the recent non-structure-preserving signature scheme of Pointcheval and Sanders [38].

We show that replacing some existing schemes used as building blocks in some protocols with ours improves the efficiency of those protocols which include direct anonymous attestation and group signature related constructions.

Paper Organization. In Sect. 2, we give some preliminary definitions. In Sect. 3, we present our signature scheme and prove its security. We give some applications of our signature scheme in Sect. 4.

Notation. We write $y = A(x; r)$ when the algorithm A on input x and randomness r outputs y. We write $y \leftarrow A(x)$ for the process of setting $y = A(x; r)$ where r is sampled at random. We also write $y \leftarrow S$ for sampling y uniformly at random from a set S. A function $\nu(.) : \mathbb{N} \rightarrow \mathbb{R}^+$ is negligible (in n) if for every polynomial $p(.)$ and all sufficiently large values of n, it holds that $\nu(n) < \frac{1}{p(n)}$. By PPT we mean running in probabilistic polynomial time in the relevant security parameter. By $[k]$, we denote the set $\{1, \ldots, k\}$.

2 Preliminaries

In this section we provide some preliminary definitions.

2.1 Bilinear Groups

A bilinear group is a tuple $\mathcal{P} := (\mathbb{G}, \tilde{\mathbb{G}}, \mathbb{T}, p, G, \tilde{G}, e)$ where $\mathbb{G}$, $\tilde{\mathbb{G}}$ and $\mathbb{T}$ are groups of a prime order p, and G and $\tilde{G}$ generate $\mathbb{G}$ and $\tilde{\mathbb{G}}$, respectively. The function e is a non-degenerate bilinear map $e : \mathbb{G} \times \tilde{\mathbb{G}} \longrightarrow \mathbb{T}$.

For clarity, elements of $\tilde{\mathbb{G}}$ will be accented with $\tilde{}$. We use multiplicative notation for all the groups. We let $\mathbb{G}^\times := \mathbb{G} \setminus \{1_\mathbb{G}\}$ and $\tilde{\mathbb{G}}^\times := \tilde{\mathbb{G}} \setminus \{1_{\tilde{\mathbb{G}}}\}$. In this paper, we work in the efficient Type-III setting [24], where $\mathbb{G} \neq \tilde{\mathbb{G}}$ and there is no efficiently computable isomorphism between the source groups in either direction. We assume there is an algorithm $\mathsf{BGSetup}$ that on input a security parameter λ, outputs a description of bilinear groups.

The message space of our signature scheme are elements of the subgroup $\hat{G}$ of $\mathbb{G} \times \tilde{\mathbb{G}}$ defined as the image of the map

$$\psi : \begin{cases} \mathbb{Z}_p \longrightarrow \mathbb{G} \times \tilde{\mathbb{G}} \\ x \longmapsto (G^x, \tilde{G}^x) \end{cases}$$

Given an element $(M, \tilde{N}) \in \mathbb{G} \times \tilde{\mathbb{G}}$, one can efficiently test whether $(M, \tilde{N}) \in \hat{G}$ by checking $e(M, \tilde{G}) = e(G, \tilde{N})$.[1]

2.2 Complexity Assumptions

Definition 1 (Decisional Diffie-Hellman (DDH) Assumption). *The DDH assumption holds relative to a group setup $\mathcal{G}$ if for all PPT adversaries $\mathcal{A}$*

$$\Pr\left[\begin{array}{l}(\mathbb{G}, G, p) \leftarrow \mathcal{G}(1^\lambda);\ r, s, t \leftarrow \mathbb{Z}_p;\ b \leftarrow \{0,1\}; \\ R := G^r;\ S := G^s;\ T := G^{brs+(1-b)t} \ : \mathcal{A}(G, R, S, T) = b\end{array}\right] \leq \frac{1}{2} + \nu(\lambda)\ .$$

Definition 2 (Symmetric External Diffie-Hellman (SXDH) Assumption). *Given a bilinear group $\mathcal{P} := (\mathbb{G}, \tilde{\mathbb{G}}, \mathbb{T}, p, G, \tilde{G}, e)$, the SXDH assumption requires that the DDH assumption holds in both groups $\mathbb{G}$ and $\tilde{\mathbb{G}}$.*

2.3 Digital Signatures

A digital signature scheme (over a bilinear group $\mathcal{P}$ generated by $\mathsf{BGSetup}$) for a message space $\mathcal{M}$ is a tuple $\mathcal{DS} := (\mathsf{KeyGen}, \mathsf{Sign}, \mathsf{Verify})$ whose definitions are:

- $\mathsf{KeyGen}(\mathcal{P})$ this probabilistic algorithm takes as input a bilinear group $\mathcal{P}$ and outputs a pair of secret/verification keys $(\mathsf{sk}, \mathsf{vk})$.
- $\mathsf{Sign}(\mathsf{sk}, m)$ this probabilistic algorithm takes as input a secret key sk and a message $m \in \mathcal{M}$, and outputs a signature σ.
- $\mathsf{Verify}(\mathsf{vk}, m, \sigma)$ this deterministic algorithm outputs 1 if σ is a vlaid signature on m w.r.t. the verification key vk.

Definition 3 (Correctness). *A signature scheme $\mathcal{DS}$ over a bilinear group generator* $\mathsf{BGSetup}$ *is* (perfectly) correct *if for all $\lambda \in \mathbb{N}$*

$$\Pr\left[\begin{array}{c}\mathcal{P} \leftarrow \mathsf{BGSetup}(1^\lambda); (\mathsf{sk}, \mathsf{vk}) \leftarrow \mathsf{KeyGen}(\mathcal{P}); \\ m \leftarrow \mathcal{M}; \sigma \leftarrow \mathsf{Sign}(\mathsf{sk}, m) : \mathsf{Verify}(\mathsf{vk}, m, \sigma) = 1\end{array}\right] = 1.$$

Definition 4 (Existential Unforgeability). *A signature scheme $\mathcal{DS}$ over a bilinear group generator* $\mathsf{BGSetup}$ *is* existentially unforgeable *against adaptive chosen-message attack if for all $\lambda \in \mathbb{N}$ for all PPT adversaries $\mathcal{A}$*

$$\Pr\left[\begin{array}{c}\mathcal{P} \leftarrow \mathsf{BGSetup}(1^\lambda); (\mathsf{sk}, \mathsf{vk}) \leftarrow \mathsf{KeyGen}(\mathcal{P}); (\sigma^*, m^*) \leftarrow \mathcal{A}^{\mathsf{Sign}(\mathsf{sk}, \cdot)}(\mathcal{P}, \mathsf{vk}) \\ : \mathsf{Verify}(\mathsf{vk}, m^*, \sigma^*) = 1 \quad \textit{and} \quad m^* \notin Q_{\mathsf{Sign}}\end{array}\right] \leq \nu(\lambda),$$

where Q_{Sign} is the set of messages queried to Sign.

[1] The elements of this group are called Diffie-Hellman pairs in [3,22].

We consider schemes which are re-randomizable (i.e. weakly unforgeable) in the sense that given a signature on a message m, anyone without knowledge of the signing key, can compute a fresh signature on the same message. A desirable property for such class of schemes is that randomized signatures are indistinguishable from fresh signatures on the same message. Thus, we define an algorithm $\mathsf{Randomize}$ which on input (vk, m, σ), where σ being a valid signature on m, outputs a new signature σ' on m.

Definition 5 (Randomizability). *A signature scheme $\mathcal{DS}$ over a bilinear group generator* $\mathsf{BGSetup}$ *is* randomizable *if for all $\lambda \in \mathbb{N}$ for all stateful adversaries $\mathcal{A}$*

$$\Pr\left[\begin{array}{l}\mathcal{P} \leftarrow \mathsf{BGSetup}(1^\lambda); (\mathsf{sk}, \mathsf{vk}) \leftarrow \mathsf{KeyGen}(\mathcal{P});\\ (\sigma^*, m^*) \leftarrow \mathcal{A}(\mathcal{P}, \mathsf{sk}, \mathsf{vk}); b \leftarrow \{0,1\};\\ \sigma_0 \leftarrow \mathsf{Sign}(\mathsf{sk}, m^*); \sigma_1 \leftarrow \mathsf{Randomize}(\mathsf{vk}, m^*, \sigma^*);\\ \quad : \mathsf{Verify}(\mathsf{vk}, m^*, \sigma^*) = 1 \ \textit{and}\ \mathcal{A}(\sigma_b) = b\end{array}\right] \leq \frac{1}{2} + \nu(\lambda).$$

We say the scheme has *Perfect Randomizability* when $\nu(\lambda) = 0$. Note that the above definition of randomizability is stronger than the variant where the signature σ^* is generated by the challenger rather than the adversary herself.

Structure-Preserving Signatures. Structure-preserving signatures [3] are signature schemes defined over bilinear groups where the messages, the verification key and signatures are all group elements and verifying signatures only involves deciding group membership of the signature components and evaluating Pairing Product Equations (PPE) of the form of Eq. 1.

$$\prod_i \prod_j e(A_i, \tilde{B}_j)^{c_{i,j}} = 1_{\mathbb{T}}, \tag{1}$$

where $A_i \in \mathbb{G}$ and $\tilde{B}_j \in \tilde{\mathbb{G}}$ are group elements appearing in $\mathcal{P}, m, \mathsf{vk}, \sigma$, whereas $c_{i,j} \in \mathbb{Z}_p$ are constants.

2.4 Randomizable Weakly Blind Signatures

A randomizable weakly blind signature scheme, as defined by Bernhard et al. [12], is similar to a standard blind signature scheme [18] but unlike the latter, in the former, the signer never gets to see the signed message. More precisely, in the blindness game of the former (referred to as *weak blindness*), the challenge messages are chosen by the challenger rather than the adversary and are never revealed to the adversary. Formally, a randomizable weakly blind signature scheme BS (with a two-move signature request phase) for a message space $\mathcal{M}_{\mathsf{BS}}$ consists of the following polynomial-time algorithms $\mathsf{BS} := (\mathsf{Setup}_{\mathsf{BS}}, \mathsf{KeyGen}_{\mathsf{BS}}, \mathsf{Request}_{\mathsf{BS}}, \mathsf{Issue}_{\mathsf{BS}}, \mathsf{Verify}_{\mathsf{BS}}, \mathsf{Randomize}_{\mathsf{BS}})$. All algorithms (bar $\mathsf{Setup}_{\mathsf{BS}}$) are assumed to take as (implicit) input a parameter set $\mathsf{param}_{\mathsf{BS}}$ output by $\mathsf{Setup}_{\mathsf{BS}}$.

- $\mathsf{Setup}_{\mathsf{BS}}(1^\lambda)$ outputs public parameters $\mathsf{param}_{\mathsf{BS}}$.
- $\mathsf{KeyGen}_{\mathsf{BS}}(\mathsf{param}_{\mathsf{BS}})$ outputs a public/secret key pair $(\mathsf{vk}_{\mathsf{BS}}, \mathsf{sk}_{\mathsf{BS}})$ for the signer.
- $(\mathsf{Request}^0_{\mathsf{BS}}, \mathsf{Issue}^1_{\mathsf{BS}}, \mathsf{Request}^1_{\mathsf{BS}})$ is an interactive protocol run between a user and a signer. The protocol is initiated by the user by calling $\mathsf{Request}^0_{\mathsf{BS}}(\mathsf{vk}_{\mathsf{BS}}, m)$ to obtain a value ρ_0 and some state information st^0_R (which is assumed to contain the message m). Then the signer and user execute, respectively,

$$(\beta_1, \mathsf{st}^1_I) \leftarrow \mathsf{Issue}^1_{\mathsf{BS}}(\mathsf{sk}_{\mathsf{BS}}, \rho_0) \quad \text{and} \quad \sigma \leftarrow \mathsf{Request}^1_{\mathsf{BS}}(\beta_1, \mathsf{st}^0_R),$$

  where σ is a signature on the message m (or the reject symbol $\perp$). We write $\sigma \leftarrow \langle \mathsf{Request}_{\mathsf{BS}}(\mathsf{vk}_{\mathsf{BS}}, m), \mathsf{Issue}_{\mathsf{BS}}(\mathsf{sk}_{\mathsf{BS}}) \rangle$ for the output of correct running of this protocol on the given inputs.
- $\mathsf{Verify}_{\mathsf{BS}}(\mathsf{vk}_{\mathsf{BS}}, m, \sigma)$ outputs 1 if σ is a valid signature on m and 0 otherwise.
- $\mathsf{Randomize}_{\mathsf{BS}}(\mathsf{vk}_{\mathsf{BS}}, \sigma)$ given a signature σ on an unknown message m, produces another valid signature σ' on the same message.

Definition 6 (Correctness). *A randomizable weakly blind signature scheme is* (perfectly) correct *if for all* $\lambda \in \mathbb{N}$

$$\Pr\left[\begin{array}{l} \mathsf{param}_{\mathsf{BS}} \leftarrow \mathsf{Setup}_{\mathsf{BS}}(1^\lambda); (\mathsf{vk}_{\mathsf{BS}}, \mathsf{sk}_{\mathsf{BS}}) \leftarrow \mathsf{KeyGen}_{\mathsf{BS}}(\mathsf{param}_{\mathsf{BS}}); \\ m \leftarrow \mathcal{M}_{\mathsf{BS}}; \sigma \leftarrow \langle \mathsf{Request}_{\mathsf{BS}}(\mathsf{vk}_{\mathsf{BS}}, m), \mathsf{Issue}_{\mathsf{BS}}(\mathsf{sk}_{\mathsf{BS}}) \rangle; \\ \sigma' \leftarrow \mathsf{Randomize}_{\mathsf{BS}}(\mathsf{vk}_{\mathsf{BS}}, \sigma) \\ \quad : \mathsf{Verify}_{\mathsf{BS}}(\mathsf{vk}_{\mathsf{BS}}, m, \sigma) = 1 \quad \textit{and} \quad \mathsf{Verify}_{\mathsf{BS}}(\mathsf{vk}_{\mathsf{BS}}, m, \sigma') = 1 \end{array}\right] = 1.$$

Definition 7 (Unforgeability). *A randomizable weakly blind signature scheme is* unforgeable *if for all* $\lambda \in \mathbb{N}$*, all PPT adversaries* $\mathcal{A}$ *have a negligible advantage in the game in Fig. 1.*

Experiment: $\mathsf{Exp}^{\mathrm{Unforge}}_{\mathsf{BS},\mathcal{A}}(\lambda)$:

- $\mathsf{param}_{\mathsf{BS}} \leftarrow \mathsf{Setup}_{\mathsf{BS}}(1^\lambda)$.
- $(\mathsf{vk}_{\mathsf{BS}}, \mathsf{sk}_{\mathsf{BS}}) \leftarrow \mathsf{KeyGen}_{\mathsf{BS}}(\mathsf{param}_{\mathsf{BS}})$.
- $\Big((m_1, \sigma_1), \ldots, (m_{n+1}, \sigma_{n+1})\Big) \leftarrow \mathcal{A}^{\mathsf{Issue}_{\mathsf{BS}}(\cdot,\cdot)}(\mathsf{vk}_{\mathsf{BS}}, \mathsf{param}_{\mathsf{BS}})$.
- Return 0 if any of the following holds. Otherwise, Return 1:
  - $\mathcal{A}$ called its oracle more than n times.
  - $\exists i, j \in \{1, \ldots, n+1\}$ s.t. $i \neq j$, but $m_i = m_j$.
  - $\exists i \in \{1, \ldots, n+1\}$ s.t. $\mathsf{Verify}_{\mathsf{BS}}(\mathsf{vk}_{\mathsf{BS}}, m_i, \sigma_i) = 0$.

Fig. 1. The unforgeability game for randomizable weakly blind signatures

Definition 8 (Weak Blindness). *A randomizable weakly blind signature scheme is* weakly blind *if for all* $\lambda \in \mathbb{N}$*, all PPT adversaries* $\mathcal{A}$ *have a negligible advantage in the game in Fig. 2.*

Experiment: $\mathsf{Exp}^{\mathrm{wBlind}}_{\mathsf{BS},\mathcal{A}}(\lambda)$:
- $\mathsf{param}_{\mathsf{BS}} \leftarrow \mathsf{Setup}_{\mathsf{BS}}(1^\lambda)$.
- $(\mathsf{vk}_{\mathsf{BS}}, \mathsf{sk}_{\mathsf{BS}}) \leftarrow \mathsf{KeyGen}_{\mathsf{BS}}(\mathsf{param}_{\mathsf{BS}})$.
- $m_0, m_1 \leftarrow \mathcal{M}_{\mathsf{BS}}$.
- $(\rho_0, \mathsf{st}^0_R) \leftarrow \mathsf{Request}^0_{\mathsf{BS}}(\mathsf{vk}_{\mathsf{BS}}, m_0)$.
- $(\beta_1, \mathsf{st}_{\mathcal{A}}) \leftarrow \mathcal{A}(\mathsf{param}_{\mathsf{BS}}, \mathsf{vk}_{\mathsf{BS}}, \mathsf{sk}_{\mathsf{BS}}, \rho_0)$.
- $\sigma_0 \leftarrow \mathsf{Request}^1_{\mathsf{BS}}(\beta_1, \mathsf{st}^0_R)$.
- If $\sigma_0 = \perp$ or $\mathsf{Verify}_{\mathsf{BS}}(\mathsf{vk}_{\mathsf{BS}}, m_0, \sigma_0) = 0$ Then Return 0.
- $b \leftarrow \{0, 1\}$.
- If $b = 0$ Then $\sigma_1 \leftarrow \mathsf{Randomize}_{\mathsf{BS}}(\mathsf{vk}_{\mathsf{BS}}, \sigma_0)$.
- Else $\sigma_1 \leftarrow \langle \mathsf{Request}_{\mathsf{BS}}(\mathsf{vk}_{\mathsf{BS}}, m_1), \mathsf{Issue}_{\mathsf{BS}}(\mathsf{sk}_{\mathsf{BS}}) \rangle$.
- $b^* \leftarrow \mathcal{A}(\mathsf{st}_{\mathcal{A}}, \sigma_0, \sigma_1)$.
- Return 1 If $b = b^*$ Else Return 0.

Fig. 2. The weak blindness game for randomizable weakly blind signatures

2.5 Groth-Sahai Proofs

Groth-Sahai (GS) proofs [31] are non-interactive proofs in the CRS model. We will use GS proofs that are secure under the SXDH assumption, which is the most efficient instantiation of the proof system [27], and that prove knowledge of witnesses to pairing-product equations of the form

$$\prod_{j=1}^{n} e(A_j, \underline{\tilde{Y}_j}) \prod_{i=1}^{m} e(\underline{X_i}, \tilde{B}_i) \prod_{i=1}^{m} \prod_{j=1}^{n} e(\underline{X_i}, \underline{\tilde{Y}_j})^{\gamma_{i,j}} = \prod_{\ell=1}^{k} e(G_\ell, \tilde{H}_\ell) \tag{2}$$

All underlined variables are part of the witness whereas the rest of the values are public constants. The language for these proofs is of the form $\mathcal{L} := \{\mathsf{statement} \mid \exists\, \mathsf{witness} : E(\mathsf{statement}, \mathsf{witness}) \text{ holds } \}$, where $E(\mathsf{statement}, \cdot)$ is a set of pairing-product equations. The system is defined by a tuple of algorithms $(\mathsf{GSSetup}, \mathsf{GSProve}, \mathsf{GSVerify}, \mathsf{GSExtract}, \mathsf{GSSimSetup}, \mathsf{GSSimProve})$. $\mathsf{GSSetup}$ takes as input the description of a bilinear group $\mathcal{P}$ and outputs a *binding* reference string crs and an extraction key xk. $\mathsf{GSProve}$ takes as input the string crs, a set of equations $\mathsf{statement}$ and a witness, and outputs a proof Ω for the satisfiability of the equations. $\mathsf{GSVerify}$ takes as input a set of equations, a string crs and a proof Ω and outputs 1 if the proof is valid, and 0 otherwise. $\mathsf{GSExtract}$ takes as input a binding crs, the extraction key xk and a valid proof Ω, and outputs the witness used for the proof. $\mathsf{GSSimSetup}$, on input a bilinear group $\mathcal{P}$, outputs a *hiding* string $\mathsf{crs}_{\mathrm{Sim}}$ and a trapdoor key tr that allows to simulate proofs. $\mathsf{GSSimProve}$ takes as input $\mathsf{crs}_{\mathrm{Sim}}$, a statement and the trapdoor tr and produces a simulated proof Ω_{Sim} without a witness. The distributions of strings crs and $\mathsf{crs}_{\mathrm{Sim}}$ are computationally indistinguishable and simulated proofs are indistinguishable from proofs generated by an honest prover. The proof system has perfect completeness, (perfect) soundness, composable witness-indistinguishability/composable zero-knowledge. We refer to [31] for the formal definitions and the details of the instantiations.

3 Our Structure-Preserving Signature Scheme

Given the description of Type-III bilinear groups $\mathcal{P}$ output by $\mathsf{BGSetup}(1^\lambda)$, our scheme is given by the following four algorithms.

- $\mathsf{KeyGen}(\mathcal{P})$: Select $x, y \leftarrow \mathbb{Z}_p^\times$. Set $\mathsf{sk} := (x, y)$ and $\mathsf{vk} := (\tilde{X}, \tilde{Y}) := (\tilde{G}^x, \tilde{G}^y)$.
- $\mathsf{Sign}(\mathsf{sk}, (M, \tilde{N}))$: To sign a message $(M, \tilde{N}) \in \hat{G}$, (i.e. $(M, \tilde{N}) \in \mathbb{G} \times \tilde{\mathbb{G}}$ and $e(M, \tilde{G}) = e(G, \tilde{N})$), select $a \leftarrow \mathbb{Z}_p^\times$, and set $A := G^a$, $B := M^a$, $C := A^x \cdot B^y$. Return $\sigma := (A, B, C) \in \mathbb{G}^3$.
- $\mathsf{Verify}(\mathsf{vk}, (M, \tilde{N}), \sigma = (A, B, C))$: Return 1 iff $A \in \mathbb{G}^\times$ (i.e. $A \neq 1_\mathbb{G}$), $B, C \in \mathbb{G}$, $(M, \tilde{N}) \in \hat{G}$, and all of the following hold:

$$e(A, \tilde{N}) = e(B, \tilde{G})$$
$$e(C, \tilde{G}) = e(A, \tilde{X})e(B, \tilde{Y})$$

- $\mathsf{Randomize}(\mathsf{vk}, (M, \tilde{N}), \sigma = (A, B, C))$: Select $r \leftarrow \mathbb{Z}_p^\times$, and set $A' := A^r$, $B' := B^r$, $C' := C^r$. Return $\sigma' := (A', B', C')$.

Remark 1. Note that verifying the well-formedness of the message pair, i.e. that $(M, \tilde{N}) \in \hat{G}$, need only be done once when verifying multiple signatures on the same message. A similar argument applies to signature schemes with the same message space, e.g. [3,22,25].

Also, note that requiring checking that $A \neq 1_\mathbb{G}$ in the verification can in some sense be considered a slight deviation from the rigorous variant of the definition of structure-preserving signatures. However, since A is information-theoretically independent of the message, even when proving knowledge of a signature, one can reveal A after re-randomizing it which allows for verifying such a condition for free. We end by noting that Ghadafi [25] gave efficient Groth-Sahai proofs that a committed Groth-Sahai value is not the identity element.

Correctness of the scheme follows by inspection and is straightforward to verify. Also, that the signature is perfectly randomizable is straightforward. The distributions of valid signatures returned by the $\mathsf{Randomize}$ algorithm are identical to those returned by the Sign algorithm on the same message. Also, note that assuming the signature to be re-randomized is valid, one only needs the old signature to be able to produce a new one.

The following theorem proves that the scheme is unforgeable in the generic group model [37,40]. We note here that the unforgeability of the scheme could also be based on an interactive assumption.

Theorem 1. *The structure-preserving signature scheme is (weakly) existentially unforgeable against adaptive chosen-message attack in the generic group model.*

Proof. The proof follows from the proof of the following theorem:

Theorem 2. *Let $\mathcal{A}$ be an adversary in the generic group model against our scheme. Assume $\mathcal{A}$ makes q_G group operation queries, q_P pairing queries, and q_S sign queries. The probability ϵ of adversary $\mathcal{A}$ winning the game is bounded by $\epsilon \leq \frac{(q_G+q_P+3q_S+4)^2 \cdot 3}{p}$, where p is the (prime) order of the generic groups.*

Proof. We start by re-stating the following Schwartz Zippel lemma [39]:

Lemma 1. *Let p be a prime and $P(x_1, \ldots, x_n) \in \mathbb{F}_p[x_1, \ldots, x_n]$ be a non-zero polynomial with a total degree $\leq d$. Then the probability that $P(x_1, \ldots, x_n) = 0$ is $\leq \frac{d}{p}$.*

Adversary $\mathcal{A}$ interacts with those oracles via group handles. We define three random encoding functions $\xi_1 : \mathbb{G} \longrightarrow \{0,1\}^*$, $\xi_2 : \tilde{\mathbb{G}} \longrightarrow \{0,1\}^*$ and $\xi_3 : \mathbb{T} \longrightarrow \{0,1\}^*$ where ξ_i maps elements from the corresponding group into random strings. The challenger keeps three lists $\mathcal{L}_1, \mathcal{L}_2, \mathcal{L}_T$ which contain pairs of the form (τ, P) where τ is a "random" encoding of the group element (i.e. τ is an output of the map ξ_i) and P is some polynomial in $\mathbb{F}_p[X, Y, A_1, \ldots, A_{q_S}]$.

To each list we associate an Update algorithm, that takes as input the specific list $\mathcal{L}_i$ and a polynomial P. The algorithm $\mathsf{Update}(\mathcal{L}_i, P)$ searches the list in question for a pair whose second component is equal to P, if such a pair is found, the algorithm returns its first component as a result. Otherwise, a new random encoding τ, different from all other elements used so far, is chosen and the pair (τ, P) is added to the list $\mathcal{L}_i$. The value τ is then returned. Note that at no point $\mathcal{A}$ gets access to the second element in the pairs.

The challenger starts by calling: $\mathsf{Update}(\mathcal{L}_1, 1)$, $\mathsf{Update}(\mathcal{L}_2, 1)$, $\mathsf{Update}(\mathcal{L}_2, X)$ and $\mathsf{Update}(\mathcal{L}_2, Y)$. Those correspond to the group elements $G \in \mathbb{G}$ and $\tilde{G}, \tilde{X}, \tilde{Y} \in \tilde{\mathbb{G}}$ of the verification key and public elements the adversary gets in the scheme.

The oracles used in the game are defined as follows:

- Group Oracles: Oracles $\mathcal{O}_1$, $\mathcal{O}_2$ and $\mathcal{O}_T$ allow $\mathcal{A}$ access to the group operations in groups $\mathbb{G}, \tilde{\mathbb{G}}$ and $\mathbb{T}$, respectively, via subtraction/addition operations. On a call to $\mathcal{O}_i(\tau_1, \tau_2)$ $\mathcal{B}$ searches list $\mathcal{L}_i$ for pairs of the form (τ_1, P_1) and (τ_2, P_2). If both pairs exist, $\mathcal{B}$ returns the output of $\mathsf{Update}(\mathcal{L}_i, P_1 \pm P_2)$. Otherwise, it returns $\perp$. Note that exponentiation operations can be performed by calls to the group operation oracles.
- Pairing Oracle: Oracle $\mathcal{O}_P$ allows $\mathcal{A}$ to perform pairing operations. On a call to $\mathcal{O}_P(\tau_1, \tau_2)$, $\mathcal{B}$ searches the list $\mathcal{L}_1$ for the pair (τ_1, P_1), and the list $\mathcal{L}_2$ for the pair (τ_2, P_2). If both pairs exist, $\mathcal{B}$ returns the output of $\mathsf{Update}(\mathcal{L}_T, P_1 \cdot P_2)$. Otherwise, it returns $\perp$.
- Sign Oracle: The adversary may make up to q_S queries $\mathcal{O}_S(\tau_1, \tau_2)$.
  The challenger searches list $\mathcal{L}_1$ for a pair (τ_1, P_1) and list $\mathcal{L}_2$ for a pair (τ_2, P_2). If they do not exist or $P_1 \neq P_2$, $\mathcal{B}$ returns $\perp$. Otherwise, it executes the following operations, where A_i, X and Y are indeterminants:

$$\tau_{A_i} \leftarrow \mathsf{Update}(\mathcal{L}_1, A_i),$$
$$\tau_{B_i} \leftarrow \mathsf{Update}(\mathcal{L}_1, A_i \cdot P_1),$$
$$\tau_{C_i} \leftarrow \mathsf{Update}(\mathcal{L}_1, A_i \cdot (X + P_1 \cdot Y)).$$

Returning the tuple $(\tau_{A_i}, \tau_{B_i}, \tau_{C_i})$ to $\mathcal{A}$.

By using the above oracles, we can simulate the entire run of the adversary. At the end of the game, the total number of non-constant polynomials contained in the three lists $\mathcal{L}_1, \mathcal{L}_2$ and $\mathcal{L}_T$ is bounded from above by $t = q_G + q_P + 3q_S + 4$.

The Adversary Output. Eventually, $\mathcal{A}$ outputs a tuple $(\tau_{A^*}, \tau_{B^*}, \tau_{C^*}, \tau_{M^*}, \tau_{\tilde{N}^*})$, where τ_{A^*}, τ_{B^*}, τ_{C^*}, and τ_{M^*} are on list $\mathcal{L}_1$ while $\tau_{\tilde{N}^*}$ is on list $\mathcal{L}_2$. Let $P_{A^*}, P_{B^*}, P_{C^*}, P_{M^*}, P_{\tilde{N}^*}$ denote their associated polynomials. For $\mathcal{A}$'s output to be valid, those polynomials can be assumed to satisfy, for some assignment $(x, y, a_1, \ldots, a_{q_S}) \in \mathbb{F}_p^{2+q_S}$ to the variables $(X, Y, A_1, \ldots, A_{q_S})$, the equations:

$$P_{B^*} = P_{A^*} \cdot P_{\tilde{N}^*} \tag{3}$$
$$P_{C^*} = P_{A^*} \cdot X + P_{B^*} \cdot Y \tag{4}$$
$$P_{M^*} = P_{\tilde{N}^*} \tag{5}$$

From this we derive a contradiction, i.e. conclude that the adversary cannot win the game. To achieve this, we need to first ensure that these polynomial identities cannot hold identically, i.e. regardless of any particular assignment $(x, y, a_1, \ldots, a_{q_S}) \in \mathbb{F}_p^{2+q_S}$ to the variables $(X, Y, A_1, \ldots, A_{q_S})$.

Let $(M_i, \tilde{N}_i)$ denote the i-th signing query where we discount queries where $(M_i, \tilde{N}_i) \notin \hat{G}$. Note that $P_{\tilde{N}_i}$ can only be a linear combination of the terms $1, X$ and Y. Thus, we have $P_{\tilde{N}_i} = r_i + s_i \cdot X + t_i \cdot Y$. Since we must have $P_{M_i} = P_{\tilde{N}_i}$, this implies that the above polynomials must also appear on the list $\mathcal{L}_1$. However, there is no operation in $\mathbb{G}$ which creates a polynomial with a monomial term of X, nor one of Y. Thus, we conclude that all queries to the sign oracle correspond to elements whose polynomials are a constant term of the form $P_{M_i} = P_{\tilde{N}_i} = r_i$. By a similar argument, we can also deduce that the output of the adversary corresponds to polynomials with $P_{M^*} = P_{\tilde{N}^*} = r^*$. This is precisely where we use the property that the oracle will return $\perp$ unless the input query lies in $\hat{G}$.

Note that P_{A^*}, P_{B^*}, and P_{C^*} can only by a linear combination of the polynomials appearing on the list $\mathcal{L}_1$. Therefore, we have:

$$P_{A^*} = w_1 + \sum_{i=1}^{q} u_{1,i} \cdot A_i + \sum_{i=1}^{q} v_{1,i} \cdot A_i \cdot (X + r_i \cdot Y) \tag{6}$$

$$P_{B^*} = w_2 + \sum_{i=1}^{q} u_{2,i} \cdot A_i + \sum_{i=1}^{q} v_{2,i} \cdot A_i \cdot (X + r_i \cdot Y) \tag{7}$$

$$P_{C^*} = w_3 + \sum_{i=1}^{q} u_{3,i} \cdot A_i + \sum_{i=1}^{q} v_{3,i} \cdot A_i \cdot (X + r_i \cdot Y), \tag{8}$$

where $w_j, u_{j,i}, v_{j,i} \in \mathbb{F}_p$.

Note that P_{C^*}, i.e. Eq. (8), there is no monomial with a power > 1 of Y. Also, there is no monomial in $X \cdot Y$. Thus, by Eq. (4), we must have $v_{1,i} = v_{2,i} = 0$ for all i. Thus, we have

$$P_{A^*} = w_1 + \sum_{i=1}^{q} u_{1,i} \cdot A_i \qquad P_{B^*} = w_2 + \sum_{i=1}^{q} u_{2,i} \cdot A_i$$

Now by Eq. (3) we must have that

$$w_2 + \sum_{i=1}^{q} u_{2,i} \cdot A_i = r^* \cdot w_1 + \sum_{i=1}^{q} r^* \cdot u_{1,i} \cdot A_i$$

For the above to hold, we must have $w_2 = r^* \cdot w_1$ and $r^* \cdot u_{1,i} = u_{2,i}$ for all i.

By Eq. (4), we must have

$$w_3 + \sum_{i=1}^{q} u_{3,i} \cdot A_i + \sum_{i=1}^{q} v_{3,i} \cdot A_i \cdot (X + r_i \cdot Y)$$
$$= w_1 \cdot X + \sum_{i=1}^{q} u_{1,i} \cdot A_i \cdot X + r^* \cdot w_1 \cdot Y + \sum_{i=1}^{q} r^* \cdot u_{1,i} \cdot A_i \cdot Y$$

There is no term in X on the left-hand side so we must have $w_1 = 0$. Also, no constant terms or terms in A_i on the right-hand side so we must have $w_3 = 0$ and $u_{3,i} = 0$ for all i. Thus, we must have

$$\sum_{i=1}^{q} v_{3,i} \cdot A_i \cdot X + \sum_{i=1}^{q} v_{3,i} \cdot r_i \cdot A_i \cdot Y = \sum_{i=1}^{q} u_{1,i} \cdot A_i \cdot X + \sum_{i=1}^{q} r^* \cdot u_{1,i} \cdot A_i \cdot Y$$

By the monomial $A_i \cdot X$, we must have $u_{1,i} = v_{3,i}$ for all i. Since we must have $A^* \neq 1_{\mathbb{G}}$, we must have at least one pair $u_{1,i} = v_{3,i} \neq 0$ for some i. By the monomial $A_i \cdot Y$, we must have $v_{3,i} \cdot r_i = r^* \cdot u_{1,i}$. Since as we have seen we must have $u_{1,i} = v_{3,i}$, we have $r_i = r^*$ which contradicts the unforgeability requirement as the forgery is on a message pair that was queried to the sign oracle.

Thus, the adversary must win, or tell it is in a simulation, via a specific (random) assignment to the variables. We now turn to bounding the probability that the adversary wins (or detects the simulation) in this case.

The Simulation. Now the challenger chooses random values $x, y, a_i \in \mathbb{F}_p$ and evaluates the polynomials. We need to show that the challenger's simulation is sound. If $\mathcal{A}$ learned it was interacting in a simulated game, there would be two different polynomials $P_{i,j}(x, y, a_i) = P_{i,j'}(x, y, a_i)$ in list $\mathcal{L}_i$ where $P_{i,j} \neq P_{i,j'}$. The simulation will fail if any of the following is correct:

$$P_{1,j}(x, y, a_i) = P_{1,j'}(x, y, a_i) \tag{9}$$
$$P_{2,j}(x, y, a_i) = P_{2,j'}(x, y, a_i) \tag{10}$$
$$P_{T,j}(x, y, a_i) = P_{T,j'}(x, y, a_i) \tag{11}$$

Since the maximum degree of any polynomial in list $\mathcal{L}_1 \leq 2$, by applying [40][Lemma 1], we have that the probability of Eq. (9) holding is $\leq \frac{2}{p}$. Similarly, since the maximum degree of any polynomial in list $\mathcal{L}_2 \leq 1$, we have that the probability of Eq. (10) holding is $\leq \frac{1}{p}$. Finally, the probability of Eq. (11) holding is $\leq \frac{3}{p}$.

Summing over all possible values of j in each case, we have

$$\epsilon \leq \binom{|\mathcal{L}_1|}{2} \frac{2}{p} + \binom{|\mathcal{L}_2|}{2} \frac{1}{p} + \binom{|\mathcal{L}_T|}{2} \frac{3}{p},$$

where $|\mathcal{L}_i|$ denotes the size of list $\mathcal{L}_i$.

In conclusion, the probability that an adversary wins the unforgeability game is bounded by $\epsilon \leq \frac{(q_G+q_P+3q_S+4)^2 \cdot 3}{p}$. □

3.1 Efficiency Comparison

We compare in Table 1 the efficiency of our scheme with that of existing schemes for a single a message in the Type-III setting. For concrete comparison, for instance, at 128-bit security, elements of $\mathbb{G}$ and $\tilde{\mathbb{G}}$ in Type-III are 256 and 512 bits long, respectively. Therefore, our signatures at this security level are at least 256 bits shorter than the best existing scheme. The efficiency gain is even better as the security level increases. Also, as can be seen, our scheme compares favorably to existing ones in terms of the efficiency of the verification equation. For the schemes whose message space is $\hat{G}$, the cost does not include checking membership of the message in the relevant group. As discussed earlier, such a check only needs to be performed once when verifying multiple signatures on the same message. Note that many applications require the signer to prove possession of/provide multiple signatures/credentials (possibly from different signers/issuers).

Also, our scheme works well in association with the (less efficient) automorphic structure-preserving signature scheme of [3,22] since the message and key spaces of the latter lie in the message space of our scheme.

It is obvious that structure-preserving signatures (on unilateral messages) in the Type-III setting have shorter messages than schemes, including ours, whose message space is $\hat{G}$. However, we stress that this is a small price to pay to get shorter signatures and more efficient verification while remaining in the most efficient Type-III bilinear group setting.

4 Applications of Our Scheme

In this section we give some examples of how using our signature scheme improves the efficiency of some existing cryptographic protocols.

Table 1. Efficiency comparison between our scheme and other schemes

Scheme	Size				Randomize?	Assumptions	Verification	
	σ	vk	Param	m			#PPE	#Pairings
[28][a]	$\mathbb{G}^4 \times \tilde{\mathbb{G}}$	$\tilde{\mathbb{G}}^2$	-	$\mathbb{G}$	Yes	q-HLRSW	4	8
[22]	$\mathbb{G}^3 \times \tilde{\mathbb{G}}^2$	$\mathbb{G} \times \tilde{\mathbb{G}}$	$\mathbb{G}^3$	$\hat{G}$	No	q-ADHSDH + AWFCDH	3	7
[3] I	$\mathbb{G}^5 \times \tilde{\mathbb{G}}^2$	$\mathbb{G}^{10} \times \tilde{\mathbb{G}}^4$	-	$\mathbb{G}$	Partially	q-SFP	2	12
[3] II	$\mathbb{G}^2 \times \tilde{\mathbb{G}}^5$	$\mathbb{G}^{10} \times \tilde{\mathbb{G}}^4$	-	$\tilde{\mathbb{G}}$	Partially	q-SFP	2	12
[4] I	$\mathbb{G}^2 \times \tilde{\mathbb{G}}$	$\mathbb{G} \times \tilde{\mathbb{G}}^3$	-	$\mathbb{G} \times \tilde{\mathbb{G}}$	No	GGM	2	7
[4] II	$\mathbb{G}^2 \times \tilde{\mathbb{G}}$	$\mathbb{G} \times \tilde{\mathbb{G}}$	-	$\tilde{\mathbb{G}}$	Yes	GGM	2	5
[25]	$\mathbb{G}^4$	$\tilde{\mathbb{G}}^2$	-	$\hat{G}$	Yes	DH-LRSW	3	6
[17] I	$\mathbb{G} \times \tilde{\mathbb{G}}^2$	$\mathbb{G}^2$	-	$\tilde{\mathbb{G}}$	No	GGM	2	5
[17] II	$\mathbb{G} \times \tilde{\mathbb{G}}^2$	$\mathbb{G}^2$	-	$\tilde{\mathbb{G}}$	Yes	GGM	2	6
[17] III	$\mathbb{G}^2 \times \tilde{\mathbb{G}}$	$\tilde{\mathbb{G}}^2$	-	$\mathbb{G}$	Yes	GGM	2	6
[6] I	$\mathbb{G}^3 \times \tilde{\mathbb{G}}$	$\tilde{\mathbb{G}}$	$\mathbb{G}$	$\mathbb{G}$	Yes	GGM	2	6
[6] II	$\mathbb{G}^2 \times \tilde{\mathbb{G}}$	$\tilde{\mathbb{G}}$	$\mathbb{G}$	$\mathbb{G}$	No	GGM	2	6
[10]	$\mathbb{G} \times \tilde{\mathbb{G}}^2$	$\mathbb{G}^2$	-	$\tilde{\mathbb{G}}$	Yes	GGM	2	5
[30] I	$\mathbb{G} \times \tilde{\mathbb{G}}^2$	$\mathbb{G}$	$\tilde{\mathbb{G}}$	$\tilde{\mathbb{G}}$	Yes	GGM	2	6
[30] II	$\mathbb{G} \times \tilde{\mathbb{G}}^2$	$\mathbb{G}$	$\tilde{\mathbb{G}}$	$\tilde{\mathbb{G}}$	No	GGM	2	7
Ours	$\mathbb{G}^3$	$\tilde{\mathbb{G}}^2$	-	$\hat{G}$	Yes	GGM	2	5

[a]This scheme is only secure against a random message attack.

4.1 Direct Anonymous Attestation

Bernhard et al. [11] gave the first instantiations of Direct Anonymous Attestation (DAA) [13] which do not rely on random oracles. Their constructions are instantiations of Bernhard et al. [12] generic construction. Among other things, the generic construction of the latter requires a randomizable weakly blind signature. The weakly blind signature is used in the join protocol to issue a credential to the user without learning her secret key. Note that unlike in group signatures [19], in DAA users do not have public keys matching their secret keys.

To get an efficient instantiation of the notion and hence an efficient instantiation of DAA (without relying on random oracles), the efficient instantiation of Bernhard et al. [11] combined Ghadafi's structure-preserving signature scheme [25] with Groth-Sahai proofs [31] to construct an efficient weakly blind signature scheme. Their weakly blind signature instantiation yields signatures of size $\mathbb{G}^4$ and require 3 PPE equations (7 pairings or 6 pairings and 1 elliptic curve point addition in total) to verify. Exploiting the fact that our signature scheme has a similar structure to Ghadafi's scheme but yet has shorter signatures and the verification algorithm is more efficient, we get a more efficient instantiation of weakly blind signatures and hence DAA by using our scheme instead. The weakly blind signature (see Fig. 3) obtained by combining our signature scheme with Groth-Sahai proofs yields signatures of size $\mathbb{G}^3$ and require only 2 PPE equations (5 pairings in total) to verify. Also, the communication complexity of both the user and the signer in the signing protocol is the same as that in the instantiation in [11]. Thus, using our scheme one gets more efficient instantiations of DAA without relying on random oracles.

$\underline{\mathsf{Setup}_{\mathsf{BS}}(1^\lambda)}$
$\mathcal{P} \leftarrow \mathsf{BGSetup}(1^\lambda)$. $(\mathsf{crs}_1, \mathsf{xk}_1) \leftarrow \mathsf{GSSetup}(\mathcal{P})$.
$(\mathsf{crs}_2, \mathsf{xk}_2) \leftarrow \mathsf{GSSetup}(\mathcal{P})$.
Return $\mathsf{param}_{\mathsf{BS}} := (\mathcal{P}, \mathsf{crs}_1, \mathsf{crs}_2)$.

$\underline{\mathsf{KeyGen}_{\mathsf{BS}}(\mathsf{param}_{\mathsf{BS}})}$
$x, y \leftarrow \mathbb{Z}_p$. $\tilde{X} := \tilde{G}^x; \tilde{Y} := \tilde{G}^y$.
Return $\big(\mathsf{sk}_{\mathsf{BS}} := (x, y), \mathsf{vk}_{\mathsf{BS}} := (\tilde{X}, \tilde{Y})\big)$.

$\underline{\mathsf{Request}^0_{\mathsf{BS}}(\mathsf{vk}_{\mathsf{BS}}, (M, \tilde{N}))}$
$\pi \leftarrow \mathsf{GSProve}\left(\mathsf{crs}_1, \{\tilde{N}, \tilde{G}'\} : M \in \mathcal{L}_1\right)$.
Return $(\rho_0 := (M, \pi), \mathsf{st}^0_R := (M, \tilde{N}))$.

$\underline{\mathsf{Issue}^1_{\mathsf{BS}}(\mathsf{sk}_{\mathsf{BS}}, \rho_0)}$
Parse ρ_0 as (M, π).
If $\mathsf{GSVerify}(\mathsf{crs}_1, M \in \mathcal{L}_1, \pi) = 0$, Return $\perp$.
$a \leftarrow \mathbb{Z}_p^\times$; $A := G^a$; $B := M^a$; $C := A^x \cdot B^y$.
$\Omega \leftarrow \mathsf{GSProve}(\mathsf{crs}_2, \{\tilde{A}, \tilde{G}'\} : (A, B, M) \in \mathcal{L}_2)$.
Return $\beta_1 := \big((A, B, C), \Omega\big)$.

$\underline{\mathsf{Request}^1_{\mathsf{BS}}(\mathsf{vk}_{\mathsf{BS}}, \beta_1, \mathsf{st}^0_R)}$
Parse β_1 as $((A, B, C), \Omega)$.
Parse st^0_R as $(M, \tilde{N})$.
Return $\perp$ if any of the following hold:
- $A = 1_{\mathbb{G}}$.
- $e(C, \tilde{G}) \neq e(A, \tilde{X})e(B, \tilde{Y})$.
- $\mathsf{GSVerify}(\mathsf{crs}_2, (A, B, M) \in \mathcal{L}_2, \Omega) = 0$.

Return $\sigma \leftarrow \mathsf{Randomize}_{\mathsf{BS}}\big(\mathsf{vk}_{\mathsf{BS}}, (A, B, C)\big)$.

$\underline{\mathsf{Verify}_{\mathsf{BS}}(\mathsf{vk}_{\mathsf{BS}}, (M, \tilde{N}), (A, B, C))}$
If $A = 1_{\mathbb{G}}$ or $e(A, \tilde{N}) \neq e(B, \tilde{G})$
or $e(C, \tilde{G}) \neq e(A, \tilde{X})e(B, \tilde{Y})$
Then Return 0.
Else Return 1.

$\underline{\mathsf{Randomize}_{\mathsf{BS}}(\mathsf{vk}_{\mathsf{BS}}, \sigma)}$
Parse σ as (A, B, C).
$r \leftarrow \mathbb{Z}_p^\times$; $A' := A^r$; $B' := B^r$; $C' := C^r$.
Return (A', B', C').

Fig. 3. Our weakly blind signature scheme

In the construction detailed in Fig. 3, we use the following languages for the zero-knowledge proofs for the user and signer respectively[2]:

$$\mathcal{L}_1 : \left\{ \big(M, (\tilde{N}, \tilde{G}')\big) : e(G, \underline{\tilde{N}}) = e(M, \underline{\tilde{G}'}) \ \wedge \ \underline{\tilde{G}'} \cdot \tilde{G}^{-1} = 1_{\tilde{\mathbb{G}}} \right\}$$

$$\mathcal{L}_2 : \left\{ \big((A, B, M), (\tilde{A}, \tilde{G}')\big) : e(G, \underline{\tilde{A}}) = e(A, \underline{\tilde{G}'}) \ \wedge \ e(M, \underline{\tilde{A}}) = e(B, \underline{\tilde{G}'}) \ \wedge \ \underline{\tilde{G}'} \cdot \tilde{G}^{-1} = 1_{\tilde{\mathbb{G}}} \right\}$$

We prove following theorem in the full version of the paper [26].

Theorem 3. *If the SXDH assumption holds and the signature scheme is existentially unforgeable, the weakly blind signature scheme in Fig. 3 is secure.*

4.2 Group Signatures and Similar Primitives

In all constructions of group signatures [19], the issuer (the group manager) issues membership certificates by certifying users' verification keys. The message space of our scheme being the set of Diffie-Hellman pairs makes our scheme ideal to be combined with the automorphic structure-preserving signature scheme of Fuchsbauer [3,22]. For instance, combining our signature scheme with Fuchsbauer's blind signature scheme [3,22], we get more efficient instantiations of group blind signatures [25,36] (without relying on random oracles) than those in [25]. An instantiation using our signature scheme yields group blind signatures of size $36 \cdot |\mathbb{G}| + 34 \cdot |\tilde{\mathbb{G}}|$ compared to $38 \cdot |\mathbb{G}| + 36 \cdot |\tilde{\mathbb{G}}|$ and $42 \cdot |\mathbb{G}| + 38 \cdot |\tilde{\mathbb{G}}|$ for the original constructions given in [25]. Also, since the final signature involves less

[2] The purpose of the two multi-scalar multiplication equations is to make the equations simulatable so that the proofs are zero-knowledge [31].

Groth-Sahai proofs, the verification algorithm is much more efficient as each Groth-Sahai proof requires a few pairings to verify.

Acknowledgments. We thank anonymous CT-RSA reviewers for their comments.

References

1. Abe, M., Chase, M., David, B., Kohlweiss, M., Nishimaki, R., Ohkubo, M.: Constant-size structure-preserving signatures: generic constructions and simple assumptions. In: Wang, X., Sako, K. (eds.) ASIACRYPT 2012. LNCS, vol. 7658, pp. 4–24. Springer, Heidelberg (2012)
2. Abe, M., David, B., Kohlweiss, M., Nishimaki, R., Ohkubo, M.: Tagged one-time signatures: tight security and optimal tag size. In: Kurosawa, K., Hanaoka, G. (eds.) PKC 2013. LNCS, vol. 7778, pp. 312–331. Springer, Heidelberg (2013)
3. Abe, M., Fuchsbauer, G., Groth, J., Haralambiev, K., Ohkubo, M.: Structure-preserving signatures and commitments to group elements. In: Rabin, T. (ed.) CRYPTO 2010. LNCS, vol. 6223, pp. 209–236. Springer, Heidelberg (2010)
4. Abe, M., Groth, J., Haralambiev, K., Ohkubo, M.: Optimal structure-preserving signatures in asymmetric bilinear groups. In: Rogaway, P. (ed.) CRYPTO 2011. LNCS, vol. 6841, pp. 649–666. Springer, Heidelberg (2011)
5. Abe, M., Groth, J., Ohkubo, M.: Separating short structure-preserving signatures from non-interactive assumptions. In: Lee, D.H., Wang, X. (eds.) ASIACRYPT 2011. LNCS, vol. 7073, pp. 628–646. Springer, Heidelberg (2011)
6. Abe, M., Groth, J., Ohkubo, M., Tibouchi, M.: Unified, minimal and selectively randomizable structure-preserving signatures. In: Lindell, Y. (ed.) TCC 2014. LNCS, vol. 8349, pp. 688–712. Springer, Heidelberg (2014)
7. Abe, M., Groth, J., Ohkubo, M., Tibouchi, M.: Structure-preserving signatures from Type II pairings. In: Garay, J.A., Gennaro, R. (eds.) CRYPTO 2014, Part I. LNCS, vol. 8616, pp. 390–407. Springer, Heidelberg (2014)
8. Abe, M., Kohlweiss, M., Ohkubo, M., Tibouchi, M.: Fully structure-preserving signatures and shrinking commitments. In: Oswald, E., Fischlin, M. (eds.) EUROCRYPT 2015. LNCS, vol. 9057, pp. 35–65. Springer, Heidelberg (2015)
9. Attrapadung, N., Libert, B., Peters, T.: Computing on authenticated data: new privacy definitions and constructions. In: Wang, X., Sako, K. (eds.) ASIACRYPT 2012. LNCS, vol. 7658, pp. 367–385. Springer, Heidelberg (2012)
10. Barthe, G., Fagerholm, E., Fiore, D., Scedrov, A., Schmidt, B., Tibouchi, M.: Strongly-optimal structure preserving signatures from Type II pairings: synthesis and lower bounds. In: Katz, J. (ed.) PKC 2015. LNCS, vol. 9020, pp. 355–376. Springer, Heidelberg (2015)
11. Bernhard, D., Fuchsbauer, G., Ghadafi, E.: Efficient signatures of knowledge and DAA in the standard model. In: Jacobson, M., Locasto, M., Mohassel, P., Safavi-Naini, R. (eds.) ACNS 2013. LNCS, vol. 7954, pp. 518–533. Springer, Heidelberg (2013)
12. Bernhard, D., Fuchsbauer, G., Ghadafi, E., Smart, N.P., Warinschi, B.: Anonymous attestation with user-controlled linkability. Int. J. Inf. Secur. **12**(3), 219–249 (2013)
13. Brickell, E., Camenisch, J., Chen, L.: Direct anonymous attestation. In: CCS 2004, pp. 132–145. ACM (2004)

14. Camenisch, J., Dubovitskaya, M., Haralambiev, K.: Efficient structure-preserving signature scheme from standard assumptions. In: Visconti, I., De Prisco, R. (eds.) SCN 2012. LNCS, vol. 7485, pp. 76–94. Springer, Heidelberg (2012)
15. Camenisch, J.L., Lysyanskaya, A.: Signature schemes and anonymous credentials from bilinear maps. In: Franklin, M. (ed.) CRYPTO 2004. LNCS, vol. 3152, pp. 56–72. Springer, Heidelberg (2004)
16. Chase, M., Kohlweiss, M.: A new hash-and-sign approach and structure-preserving signatures from DLIN. In: Visconti, I., De Prisco, R. (eds.) SCN 2012. LNCS, vol. 7485, pp. 131–148. Springer, Heidelberg (2012)
17. Chatterjee, S., Menezes, A.: Typpe 2 Structure-Preserving Signature Schemes Revisited. Cryptology ePrint Archive, Report 2014/635 (2014)
18. Chaum, D.: Blind signatures for untraceable payments. In: Chaum, D., Rivest, R.L., Sherman, A.T. (eds.) Advances in Cryptology, pp. 199–203. Springer, Heidelberg (1983)
19. Chaum, D., van Heyst, E.: Group signatures. In: Davies, D.W. (ed.) EUROCRYPT 1991. LNCS, vol. 547, pp. 257–265. Springer, Heidelberg (1991)
20. ElGamal, T.: A public key cryptosystem and a signature scheme based on discrete logarithms. IEEE Trans. Inf. Theory **31**(4), 469–472 (1985)
21. Fiat, A., Shamir, A.: How to prove yourself: practical solutions to identification and signature problems. In: Odlyzko, A.M. (ed.) CRYPTO 1986. LNCS, vol. 263, pp. 186–194. Springer, Heidelberg (1987)
22. Fuchsbauer, G.: Automorphic Signatures in Bilinear Groups and an Application to Round-Optimal Blind Signatures. Cryptology ePrint Archive, Report 2009/320 (2009)
23. Fuchsbauer, G.: Commuting signatures and verifiable encryption. In: Paterson, K.G. (ed.) EUROCRYPT 2011. LNCS, vol. 6632, pp. 224–245. Springer, Heidelberg (2011)
24. Galbraith, S., Paterson, K., Smart, N.P.: Pairings for cryptographers. Discrete Appl. Math. **156**, 3113–3121 (2008)
25. Ghadafi, E.: Formalizing group blind signatures and practical constructions without random oracles. In: Boyd, C., Simpson, L. (eds.) ACISP. LNCS, vol. 7959, pp. 330–346. Springer, Heidelberg (2013)
26. Ghadafi, E.: Short Structure-Preserving Signatures. Cryptology ePrint Archive, Report 2015/961 (2015). http://eprint.iacr.org/2015/961.pdf
27. Ghadafi, E., Smart, N.P., Warinschi, B.: Groth–sahai proofs revisited. In: Nguyen, P.Q., Pointcheval, D. (eds.) PKC 2010. LNCS, vol. 6056, pp. 177–192. Springer, Heidelberg (2010)
28. Green, M., Hohenberger, S.: Universally composable adaptive oblivious transfer. In: Pieprzyk, J. (ed.) ASIACRYPT 2008. LNCS, vol. 5350, pp. 179–197. Springer, Heidelberg (2008)
29. Groth, J.: Simulation-sound nizk proofs for a practical language and constant size group signatures. In: Lai, X., Chen, K. (eds.) ASIACRYPT 2006. LNCS, vol. 4284, pp. 444–459. Springer, Heidelberg (2006)
30. Groth, J.: Efficient Fully Structure-Preserving Signatures for Large Messages. Cryptology ePrint Archive, Report 2015/824 (2015)
31. Groth, J., Sahai, A.: Efficient non-interactive proof systems for bilinear groups. SIAM J. Comput. **41**(5), 1193–1232 (2012)
32. Hofheinz, D., Jager, T.: Tightly secure signatures and public-key encryption. In: Safavi-Naini, R., Canetti, R. (eds.) CRYPTO 2012. LNCS, vol. 7417, pp. 590–607. Springer, Heidelberg (2012)

33. Kiltz, E., Pan, J., Wee, H.: Structure-preserving signatures from standard assumptions, revisited. In: Gennaro, R., Robshaw, M. (eds.) CRYPTO 2015. LNCS, vol. 9216, pp. 275–295. Springer, Heidelberg (2015)
34. Libert, B., Peters, T., Yung, M.: Scalable group signatures with revocation. In: Pointcheval, D., Johansson, T. (eds.) EUROCRYPT 2012. LNCS, vol. 7237, pp. 609–627. Springer, Heidelberg (2012)
35. Boneh, D., Boyen, X., Shacham, H.: Short group signatures. In: Franklin, M. (ed.) CRYPTO 2004. LNCS, vol. 3152, pp. 41–55. Springer, Heidelberg (2004)
36. Lysyanskaya, A., Ramzan, Z.: Group blind digital signatures: a scalable solution to electronic cash. In: Hirschfeld, R. (ed.) FC 1998. LNCS, vol. 1465, pp. 184–197. Springer, Heidelberg (1998)
37. Maurer, U.M.: Abstract models of computation in cryptography. In: Smart, N.P. (ed.) Cryptography and Coding 2005. LNCS, vol. 3796, pp. 1–12. Springer, Heidelberg (2005)
38. Pointcheval, D., Sanders, O.: Short Randomizable Signatures. Cryptology ePrint Archive, Report 2015/525 (2015)
39. Schwartz, J.T.: Fast probabilistic algorithms for verification of polynomial identities. J. ACM **27**, 701–717 (1980)
40. Shoup, V.: Lower bounds for discrete logarithms and related problems. In: Fumy, W. (ed.) EUROCRYPT 1997. LNCS, vol. 1233, pp. 256–266. Springer, Heidelberg (1997)