An extended abstract of this paper appears in the proceedings of CRYPTO'15. This is the full version.

Practical Round-Optimal Blind Signatures in the Standard Model

Georg Fuchsbauer[1,†] Christian Hanser[2,‡,§] Daniel Slamanig[2,§]

[1] Institute of Science and Technology Austria
georg.fuchsbauer@ist.ac.at
[2] IAIK, Graz University of Technology, Austria
{christian.hanser|daniel.slamanig}@iaik.tugraz.at

Abstract

Round-optimal blind signatures are notoriously hard to construct in the standard model, especially in the malicious-signer model, where blindness must hold under adversarially chosen keys. This is substantiated by several impossibility results. The only construction that can be termed theoretically efficient, by Garg and Gupta (EUROCRYPT'14), requires complexity leveraging, inducing an exponential security loss.

We present a construction of practically efficient round-optimal blind signatures in the standard model. It is conceptually simple and builds on the recent structure-preserving signatures on equivalence classes (SPS-EQ) from ASIACRYPT'14. While the traditional notion of blindness follows from standard assumptions, we prove blindness under adversarially chosen keys under an interactive variant of DDH. However, we neither require non-uniform assumptions nor complexity leveraging.

We then show how to extend our construction to partially blind signatures and to blind signatures on message vectors, which yield a construction of one-show anonymous credentials à la "anonymous credentials light" (CCS'13) in the standard model.

Furthermore, we give the first SPS-EQ construction under non-interactive assumptions and show how SPS-EQ schemes imply conventional structure-preserving signatures, which allows us to apply optimality results for the latter to SPS-EQ.

Keywords: (Partially) Blind Signatures, Standard Model, SPS-EQ, One-Show Anonymous Credentials

1 Introduction

The concept of blind signatures [Cha82] dates back to the beginning of the 1980s. A blind signature scheme is an interactive protocol where a user (or obtainer) requests a signature on a message which the signer (or issuer) must not learn. In particular, the signer must not be able to link a signature to the execution of the issuing protocol in which it was produced (*blindness*). Furthermore, it should even for adaptive adversaries be infeasible to produce a valid blind signature without the signing key (*unforgeability*). Blind signatures have proven to be an important building block for cryptographic protocols, most prominently for e-cash, e-voting and one-show anonymous credentials. In more than 30 years of research, many

[†] Supported by the European Research Council, ERC Starting Grant (259668-PSPC).
[‡] Supported by EU FP7 through project MATTHEW (GA No. 610436).
[§] Supported by EU FP7 through project FutureID (GA No. 318424).

different (> 50) blind signature schemes have been proposed. The spectrum ranges from RSA-based (e.g., [Cha82, CKW04]) over DL-based (e.g., [Oka92, Abe01]) and pairing-based (e.g., [Bol03, BFPV11]) to lattice-based (e.g., [Rüc10]) constructions, as well as constructions from general assumptions (e.g., [JLO97, HKKL07, Fis06]).

Blind signatures and their round complexity. Two distinguishing features of blind signatures are whether they assume a common reference string (CRS) set up by a trusted party to which everyone has access; and the number of rounds in the signing protocol. Schemes which require only one round of interaction (two moves) are called *round-optimal* [Fis06]. Besides improving efficiency, round optimality also directly yields concurrent security (which otherwise has to be dealt with explicitly; e.g., [KZ06, HKKL07]). There are very efficient round-optimal schemes [Cha83, Bol03, BNPS03] under interactive assumptions (chosen target one more RSA inversion and chosen target CDH, respectively) in the random oracle model (ROM), as well as under the interactive LRSW [LRSW00] assumption in the CRS model [GS12]. All these schemes are in the honest-key model, where blindness only holds against signers whose keys are generated by the experiment.

Fischlin [Fis06] proposed a generic framework for constructing round-optimal blind signatures in the CRS model with blindness under malicious keys: the signer signs a commitment to the message and the blind signature is a non-interactive zero-knowledge (NIZK) proof of a signed commitment which opens to the message. Using structure-preserving signatures (SPS) [AFG$^+$10] and the Groth-Sahai (GS) proof system [GS08] instead of general NIZKs, this framework was efficiently instantiated in [AFG$^+$10]. In [BFPV11, BPV12], Blazy et al. gave alternative approaches to compact round-optimal blind signatures in the CRS model which avoid including a GS proof in the final blind signature. Another round-optimal solution with comparable computational costs was proposed by Seo and Cheon [SC12] building on work by Meiklejohn et al. [MSF10].

Removing the CRS. Known impossibility results indicate that the design of round-optimal blind signatures in the standard model has some limitations. Lindell [Lin03] showed that concurrently secure (and consequently also round-optimal) blind signatures are impossible in the standard model when using simulation-based security notions. This can however be bypassed via game-based security notions, as shown by Hazay et al. [HKKL07] for non-round-optimal constructions.

Fischlin and Schröder [FS10] showed that black-box reductions of blind-signature unforgeability to non-interactive assumptions in the standard model are impossible if the scheme has three moves or less, blindness holds statistically (or computationally if unforgeability and blindness are unrelated) and protocol transcripts allow to verify whether the user is able to derive a signature. Existing constructions [GRS$^+$11, GG14] bypass these results by making non-black-box use of the underlying primitives (and preventing signature-derivation checks in [GRS$^+$11]).

Garg et al. [GRS$^+$11] proposed the first round-optimal generic construction in the standard model, which can only be considered as a theoretical feasibility result. Using fully homomorphic encryption, the user encrypts the message sent to the signer, who evaluates the signing circuit on the ciphertext. To remove the CRS, they use two-round witness-indistinguishable proofs (ZAPs) to let the parties prove honest behavior; to preserve round-optimality, they include the first fixed round of the ZAP in the signer's public key.

Garg and Gupta [GG14] proposed the first efficient round-optimal blind signature constructions in the standard model. They build on Fischlin's framework using SPS. To remove a trusted setup, they use a two-CRS NIZK proof system based on GS proofs, include the CRSs in the public key while forcing the signer to honestly generate the CRS.

Their construction, however, requires complexity leveraging (the reduction for unforgeability needs to solve a subexponential DL instance for every signing query) and is proven secure with respect to non-uniform adversaries. Consequently, communication complexity is in the order of hundreds of KB (even at a 80-bit security level) and the computational costs (not considered by the authors) seem to limit their practical application even more significantly.

Partially blind signatures. Partially blind signatures are an extension of blind signatures, which additionally allow to include common information in a signature. Many non-round-optimal partially blind signature schemes in the ROM are based on a technique by Abe and Okamoto [AO00]. The latter [Oka06] proposed an efficient construction for non-round-optimal blind as well as partially blind signatures in the standard model. Round-optimal partially blind signatures in the CRS model can again be obtained from Fischlin's framework [Fis06]. Round-optimal partially blind signatures in the CRS model are constructed in [BPV12, MSF10, SC12]. To date, there is—to the best of our knowledge—no round-optimal partially blind signature scheme that is secure in the standard model.

One-show anonymous credentials systems. Such systems allow a user to obtain a credential on several attributes from an issuer. The user can later selectively show attributes (or prove relations about attributes) to a verifier without revealing any information about undisclosed attributes. No party (including the issuer) can link the issuing of a credential to any of its showings, yet different showings of the same credential are linkable. An efficient implementation of one-show anonymous credentials is Microsoft's U-Prove [BP10].

Baldimtsi and Lysyanskaya [BL13b] showed that the underlying signature scheme by Brands [Bra00] cannot be proven secure using known techniques. To mitigate this problem, in [BL13a] they presented a generic construction of one-show anonymous credentials in the vein of Brands' approach from so-called blind signatures with attributes [Bra00]. They also present a scheme based on a non-round-optimal blind signature scheme by Abe [Abe01] and prove their construction secure in the ROM.

Our Contribution

Blind signatures and anonymous credentials. Besides Fischlin's generic *commit-prove* paradigm [Fis06], there are other classes of schemes. For instance, RSA and BLS blind signatures [Cha83, Bol03, BNPS03] follow a *randomize-derandomize* approach, which exploits the homomorphic property of the respective signature scheme. Other approaches follow the *commit-rerandomize-transform* paradigm, where a signature on a commitment to a message can be transformed into a rerandomized (unlinkable) signature on the original message [BFPV11, GS12]. Our construction is based on a new concept, which one may call *commit-randomize-derandomize-open* approach. It does not use non-interactive proofs at all and is solely based on the recent concept of structure-preserving signature schemes on equivalence classes (SPS-EQ) [HS14] and commitments. As we also avoid a trusted setup of the commitment parameters, we do not require a CRS. We do however prove our scheme secure under interactive hardness assumptions.

In SPS-EQ the message space is partitioned into equivalence classes and given a signature on a message anyone can *adapt* the signature to a different representative of the same class. SPS-EQ requires that after signing a representative a signer cannot distinguish between an adapted signature for a new representative of the same class and a fresh signature on a completely random message.

In our blind-signature scheme the obtainer combines a commitment to the message with a normalization element yielding a representative of an equivalence class (*commit*). She chooses a random representative of the same class (*randomize*), on which the signer produces a signature. She then adapts the signature to the original representative containing the commitment (*derandomize*), which can be done without requiring the signing key. The blind signature is the rerandomized (unlinkable) signature for the original representative plus an opening for the commitment (*open*). Our contributions to blind signatures are the following:

- We propose a new approach to constructing blind signatures in the standard model based on SPS-EQ. It yields conceptually simple and compact constructions and does not rely on techniques such as complexity leveraging. Our blind signatures are practical in terms of key size, signature size, communication and computational effort (when implemented with known instantiations of SPS-EQ [FHS14], a blind signature consists of 5 bilinear-group elements).
- We provide the first construction of round-optimal partially blind signatures in the standard model, which follow straightforwardly from our blind signatures and are almost as efficient.
- We generalize our blind signature scheme to message vectors, which yields one-show anonymous credentials à la "anonymous credentials light" [BL13a]. We thus obtain one-show anonymous credentials secure in the standard model (whereas all previous ones have either no security proof or ones in the ROM).

SPS-EQ. We give the first structure-preserving signatures on equivalence classes satisfying all security notions from [HS14] under non-interactive assumptions. (Unfortunately, the scheme does not have all the properties required for building blind signatures from it, for which we strengthen the notions from [HS14].)

Moreover, we show how any SPS-EQ scheme can be turned into a standard structure-preserving signature scheme. This transformation allows us to apply the optimality criteria by Abe et al. [AGHO11, AGO11] to SPS-EQ. We conclude that the scheme from [FHS14] is optimal in terms of signature size and verification complexity and that it cannot be proven unforgeable under non-interactive assumptions.

Organization. Section 2 discusses preliminaries including signature schemes on equivalence classes (SPS-EQ). Section 3 presents our new results for SPS-EQ. Section 4 introduces our construction of round-optimal blind signatures, the extension to partially blind signatures and discusses the implications for SPS-EQ. Finally, Section 5 shows how we can construct anonymous one-show credentials by generalizing the blind signature scheme to message vectors.

2 Preliminaries

A function $\epsilon \colon \mathbb{N} \to \mathbb{R}^+$ is called *negligible* if for all $c > 0$ there is a k_0 such that $\epsilon(k) < 1/k^c$ for all $k > k_0$. By $a \xleftarrow{R} S$ we denote that a is chosen uniformly at random from a set S. Furthermore, we write $\mathsf{A}(a_1, \ldots, a_n; r)$ if we want to make the randomness r used by a probabilistic algorithm $\mathsf{A}(a_1, \ldots, a_n)$ explicit. If $\mathbb{G}$ is an (additive) group, then we use $\mathbb{G}^*$ to denote $\mathbb{G} \setminus \{0_{\mathbb{G}}\}$.

Definition 1 (Bilinear map)**.** Let $(\mathbb{G}_1, +)$, $(\mathbb{G}_2, +)$, generated by P and $\hat{P}$, resp., and $(\mathbb{G}_T, \cdot)$ be cyclic groups of prime order p. We call $e\colon \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$ a *bilinear map (pairing)* if it is efficiently computable and the following holds:

Bilinearity: $e(aP, b\hat{P}) = e(P, \hat{P})^{ab} = e(bP, a\hat{P}) \quad \forall\, a, b \in \mathbb{Z}_p$.
Non-degeneracy: $e(P, \hat{P}) \neq 1_{\mathbb{G}_T}$, i.e., $e(P, \hat{P})$ generates $\mathbb{G}_T$. ◇

If $\mathbb{G}_1 = \mathbb{G}_2$, then e is *symmetric* (Type-1) and *asymmetric* (Type-2 or 3) otherwise. For Type-2 pairings there is an efficiently computable isomorphism $\Psi\colon \mathbb{G}_2 \to \mathbb{G}_1$; for Type-3 pairings no such isomorphism is known. Type-3 pairings are currently the optimal choice in terms of efficiency and security trade-off [CM11].

Definition 2 (Bilinear-group generator)**.** A bilinear-group generator is a polynomial-time algorithm BGGen that takes a security parameter 1^κ and outputs a bilinear group $\mathsf{BG} = (p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e, P, \hat{P})$ consisting of groups $\mathbb{G}_1 = \langle P \rangle$, $\mathbb{G}_2 = \langle \hat{P} \rangle$ and $\mathbb{G}_T$ of prime order p with $\log_2 p = \kappa$ and a pairing $e\colon \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$. In this work we assume that BGGen is a *deterministic* algorithm.[1] ◇

Definition 3 (Decisional Diffie-Hellman assumption)**.** Let BGGen be a bilinear-group generator that outputs $\mathsf{BG} = (p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e, P_1 = P, P_2 = \hat{P})$. The DDH assumption holds in $\mathbb{G}_i$ for BGGen if for all probabilistic polynomial-time (PPT) adversaries $\mathcal{A}$ there is a negligible function $\epsilon(\cdot)$ such that

$$\Pr\left[\begin{array}{l} b \xleftarrow{R} \{0,1\},\ \mathsf{BG} = \mathsf{BGGen}(1^\kappa),\ r, s, t \xleftarrow{R} \mathbb{Z}_p \\ b^* \leftarrow \mathcal{A}\big(\mathsf{BG}, rP_i, sP_i, ((1-b)\cdot t + b \cdot rs)P_i\big) \end{array} : \ b^* = b \right] - \frac{1}{2} \leq \epsilon(\kappa) \ . \qquad \diamondsuit$$

Definition 4 ((Symmetric) external Diffie-Hellman assumption)**.** The XDH and SXDH assumptions hold for BGGen if the DDH assumption holds in $\mathbb{G}_1$ and holds in both $\mathbb{G}_1$ and $\mathbb{G}_2$, respectively. ◇

The next assumption is a static computational assumption derived from the SXDH version of the q-Diffie-Hellman inversion assumption [CM11].

Definition 5 (Co-Diffie-Hellman inversion assumption)**.** Let BGGen be a bilinear-group generator that outputs $\mathsf{BG} = (p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e, P_1 = P, P_2 = \hat{P})$. The co-DHI$_i^*$ assumption holds for BGGen if for every PPT adversary $\mathcal{A}$ there is a negligible function $\epsilon(\cdot)$ such that

$$\Pr\left[\mathsf{BG} = \mathsf{BGGen}(1^\kappa),\ a \xleftarrow{R} \mathbb{Z}_p^* : \tfrac{1}{a}P_i \leftarrow \mathcal{A}(\mathsf{BG}, aP_1, aP_2)\right] \leq \epsilon(\kappa) \ . \qquad \diamondsuit$$

co-DHI$_1^*$ is implied by a variant of the decision linear assumption in asymmetric groups stating that given $(\mathsf{BG}, (aP_j, bP_j)_{j\in[2]}, raP_2, sbP_2)$ for $a, b, r, s \xleftarrow{R} \mathbb{Z}_p^*$ it is hard to distinguish $T = (r+s)P_2$ from a random $\mathbb{G}_2$ element. (A co-DHI$_i^*$ solver could be used to compute $\frac{1}{a}P_1$ and $\frac{1}{b}P_1$, which enables to check whether $e(\frac{1}{a}P_1, raP_2)\, e(\frac{1}{b}P_1, sbP_2) = e(P_1, T)$.) This holds analogously for co-DHI$_2^*$.

Generalized Pedersen commitments. These are commitments to a vector of messages $\boldsymbol{m} = (m_i)_{i\in[n]} \in \mathbb{Z}_p^n$ that consist of one group element. They are perfectly hiding and computationally binding under the discrete-log assumption.

$\mathsf{Setup}_{\mathsf{P}}(1^\kappa, n)$**:** Choose a group $\mathbb{G}$ of prime order p with $\log_2 p = \kappa$ and $n+1$ distinct generators $(P_i)_{i\in[n]}, Q$ and output parameters $\mathsf{cpp} \leftarrow (\mathbb{G}, p, (P_i)_{i\in[n]}, Q)$ (which is an implicit input to the following algorithms).

[1] This is e.g. the case for BN-curves [BN05]; the most common choice for Type-3 pairings.

$\mathsf{Commit}_{\mathsf{P}}(\boldsymbol{m}; r)$**:** On input a vector $\boldsymbol{m} \in \mathbb{Z}_p^n$ and randomness $r \in \mathbb{Z}_p$, output a commitment $C \leftarrow \sum_{i\in[n]} m_i P_i + rQ$ and an opening $O \leftarrow (\boldsymbol{m}, r)$.

$\mathsf{Open}_{\mathsf{P}}(C, O)$**:** On input $C \in \mathbb{G}$ and $O = (\boldsymbol{m}, r)$, if $C = \sum_{i\in[n]} m_i P_i + rQ$ then output $\boldsymbol{m} = (m_i)_{i\in[n]}$; else output $\perp$.

Remark 1. $\mathsf{Setup}_{\mathsf{P}}$ is typically run by a trusted party and thus cpp can be seen as a CRS. Note however that the receiver can know the logarithms of the elements in cpp, as the commitment is perfectly hiding and the binding property protects against malicious senders/committers.

2.1 Structure-Preserving Signatures on Equivalence Classes

Structure-preserving signatures (SPS) [AFG+10, AGHO11, CDH12, AGOT14] can sign elements of a bilinear group without requiring any prior encoding. In such a scheme public keys, messages and signatures consist of group elements only and the verification algorithm evaluates a signature by deciding group membership and evaluating pairing-product equations (PPEs).

The notion of SPS on equivalence classes (SPS-EQ) was introduced by Hanser and Slamanig [HS14]. Their initial instantiation turned out to only be secure against random-message attacks (cf. [Fuc14] and the updated full version of [HS14]), but together with Fuchsbauer [FHS14] they subsequently presented a scheme that is unforgeable under chosen-message attack (EUF-CMA) in the generic group model.

The concept of SPS-EQ is as follows. Let p be a prime and $\ell > 1$; then $\mathbb{Z}_p^\ell$ is a vector space and we can define a projective equivalence relation on it, which propagates to $\mathbb{G}_i^\ell$ and partitions $\mathbb{G}_i^\ell$ into equivalence classes. Let $\sim_{\mathcal{R}}$ be this relation, i.e., for $M, N \in \mathbb{G}_i^\ell$: $M \sim_{\mathcal{R}} N \Leftrightarrow \exists\, s \in \mathbb{Z}_p^* : M = sN$. An SPS-EQ scheme signs an equivalence class $[M]_{\mathcal{R}}$ for $M \in (\mathbb{G}_i^*)^\ell$ by signing a representative M of $[M]_{\mathcal{R}}$. It then allows for switching to other representatives of $[M]_{\mathcal{R}}$ and updating the signature without access to the secret key. An important property of SPS-EQ is *class-hiding*, which roughly means that two message-signature pairs corresponding to the same class should be unlinkable.

Here, we discuss the abstract model and the security model of such a signature scheme, as introduced in [HS14].

Definition 6 (Structure-preserving signatures on equivalence classes)**.** An SPS-EQ scheme $\mathsf{SPS\text{-}EQ}$ on $(\mathbb{G}_i^*)^\ell$ (for $i \in \{1, 2\}$) consists of the following PPT algorithms:

$\mathsf{BGGen}_{\mathcal{R}}(1^\kappa)$, a bilinear-group generation algorithm, which on input a security parameter κ outputs an asymmetric bilinear group BG.

$\mathsf{KeyGen}_{\mathcal{R}}(\mathsf{BG}, \ell)$, on input BG and vector length $\ell > 1$, outputs a key pair $(\mathsf{sk}, \mathsf{pk})$.

$\mathsf{Sign}_{\mathcal{R}}(M, \mathsf{sk})$, given a representative $M \in (\mathbb{G}_i^*)^\ell$ and a secret key sk, outputs a signature σ for the equivalence class $[M]_{\mathcal{R}}$.

$\mathsf{ChgRep}_{\mathcal{R}}(M, \sigma, \mu, \mathsf{pk})$, on input a representative $M \in (\mathbb{G}_i^*)^\ell$ of class $[M]_{\mathcal{R}}$, a signature σ on M, a scalar μ and a public key pk, returns an updated message-signature pair (M', σ'), where $M' = \mu \cdot M$ is the new representative and σ' its updated signature.

$\mathsf{Verify}_{\mathcal{R}}(M, \sigma, \mathsf{pk})$ is deterministic and, on input a representative $M \in (\mathbb{G}_i^*)^\ell$, a signature σ and a public key pk, outputs 1 if σ is valid for M under pk and 0 otherwise.

$\mathsf{VKey}_{\mathcal{R}}(\mathsf{sk}, \mathsf{pk})$ is a deterministic algorithm, which given a secret key sk and a public key pk outputs 1 if the keys are consistent and 0 otherwise. ♢

$\mathsf{BGGen}_{\mathcal{R}}(1^\kappa)$: Generate a Type-3 bilinear group BG with order p of bitlength κ.

$\mathsf{KeyGen}_{\mathcal{R}}(\mathsf{BG}, \ell)$: Given BG and vector length $\ell > 1$, choose $(x_i)_{i\in[\ell]} \xleftarrow{R} (\mathbb{Z}_p^*)^\ell$, set $\mathsf{sk} \leftarrow (x_i)_{i\in[\ell]}$, $\mathsf{pk} \leftarrow (\hat{X}_i)_{i\in[\ell]} = (x_i\hat{P})_{i\in[\ell]}$ and output $(\mathsf{sk}, \mathsf{pk})$.

$\mathsf{Sign}_{\mathcal{R}}(M, \mathsf{sk})$: Given a representative $M = (M_i)_{i\in[\ell]} \in (\mathbb{G}_1^*)^\ell$ of class $[M]_{\mathcal{R}}$ and secret key $\mathsf{sk} = (x_i)_{i\in[\ell]}$, choose $y \xleftarrow{R} \mathbb{Z}_p^*$ and output $\sigma = (Z, Y, \hat{Y})$ with

$$Z \leftarrow y \textstyle\sum_{i\in[\ell]} x_i M_i \qquad\qquad Y \leftarrow \frac{1}{y}P \qquad\qquad \hat{Y} \leftarrow \frac{1}{y}\hat{P}$$

$\mathsf{Verify}_{\mathcal{R}}(M, \sigma, \mathsf{pk})$: Given $M = (M_i)_{i\in[\ell]} \in (\mathbb{G}_1^*)^\ell$, $\sigma = (Z, Y, \hat{Y}) \in \mathbb{G}_1 \times \mathbb{G}_1^* \times \mathbb{G}_2^*$ and public key $\mathsf{pk} = (\hat{X}_i)_{i\in[\ell]}$, output 1 if the following hold and 0 otherwise:

$$\textstyle\prod_{i\in[\ell]} e(M_i, \hat{X}_i) = e(Z, \hat{Y}) \qquad\qquad e(Y, \hat{P}) = e(P, \hat{Y})$$

$\mathsf{ChgRep}_{\mathcal{R}}(M, \sigma, \mu, \mathsf{pk})$: Given representative $M = (M_i)_{i\in[\ell]} \in (\mathbb{G}_1^*)^\ell$, $\sigma = (Z, Y, \hat{Y})$, scalar $\mu \in \mathbb{Z}_p^*$ and pk, return $\perp$ if $\mathsf{Verify}_{\mathcal{R}}(M, \sigma, \mathsf{pk}) = 0$. Otherwise pick $\psi \xleftarrow{R} \mathbb{Z}_p^*$ and return $(\mu M, \sigma')$ with $\sigma' \leftarrow \big(\psi\mu Z, \frac{1}{\psi}Y, \frac{1}{\psi}\hat{Y}\big)$.

$\mathsf{VKey}_{\mathcal{R}}(\mathsf{sk}, \mathsf{pk})$: Given $\mathsf{sk} = (x_i)_{i\in[\ell]} \in (\mathbb{Z}_p^*)^\ell$ and $\mathsf{pk} = (\hat{X}_i)_{i\in[\ell]} \in (\mathbb{G}_2^*)^\ell$, output 1 if for all $i \in [\ell] : x_i\hat{P} = \hat{X}_i$ and 0 otherwise.

Scheme 1: EUF-CMA-secure construction of an SPS-EQ scheme

An SPS-EQ scheme must satisfy *correctness*, *EUF-CMA security* and *class-hiding*.

Definition 7 (Correctness)**.** An SPS-EQ scheme $\mathsf{SPS\text{-}EQ}$ on $(\mathbb{G}_i^*)^\ell$ is *correct* if for all $\kappa \in \mathbb{N}$, all $\ell > 1$, all key pairs $(\mathsf{sk}, \mathsf{pk}) \leftarrow \mathsf{KeyGen}_{\mathcal{R}}(\mathsf{BGGen}_{\mathcal{R}}(1^\kappa), \ell)$, all messages $M \in (\mathbb{G}_i^*)^\ell$ and all $\mu \in \mathbb{Z}_p^*$: $\mathsf{VKey}_{\mathcal{R}}(\mathsf{sk}, \mathsf{pk}) = 1$,

$$\begin{aligned} &\Pr\big[\mathsf{Verify}_{\mathcal{R}}(M, \mathsf{Sign}_{\mathcal{R}}(M, \mathsf{sk}), \mathsf{pk}) = 1\big] = 1 \quad \text{and} \\ &\Pr\big[\mathsf{Verify}_{\mathcal{R}}(\mathsf{ChgRep}_{\mathcal{R}}(M, \mathsf{Sign}_{\mathcal{R}}(M, \mathsf{sk}), \mu, \mathsf{pk}), \mathsf{pk}) = 1\big] = 1 \ . \end{aligned}$$ ◇

In contrast to standard signatures, EUF-CMA security is defined with respect to equivalence classes, i.e., a forgery is a signature on a message from an equivalence class from which no message has been signed.

Definition 8 (EUF-CMA)**.** An SPS-EQ scheme $\mathsf{SPS\text{-}EQ}$ on $(\mathbb{G}_i^*)^\ell$ is *existentially unforgeable under adaptively chosen-message attacks*, if for all PPT algorithms $\mathcal{A}$ with access to a signing oracle $\mathcal{O}$, there is a negligible function $\epsilon(\cdot)$ such that:

$$\Pr\left[\begin{array}{l} \mathsf{BG} \leftarrow \mathsf{BGGen}_{\mathcal{R}}(1^\kappa), \\ (\mathsf{sk}, \mathsf{pk}) \leftarrow \mathsf{KeyGen}_{\mathcal{R}}(\mathsf{BG}, \ell), \\ (M^*, \sigma^*) \leftarrow \mathcal{A}^{\mathcal{O}(\cdot, \mathsf{sk})}(\mathsf{pk}) \end{array} : \begin{array}{l} [M^*]_{\mathcal{R}} \neq [M]_{\mathcal{R}} \quad \forall M \in \mathcal{Q} \ \wedge \\ \mathsf{Verify}_{\mathcal{R}}(M^*, \sigma^*, \mathsf{pk}) = 1 \end{array}\right] \leq \epsilon(\kappa) \ ,$$

where $\mathcal{Q}$ is the set of queries that $\mathcal{A}$ has issued to the signing oracle $\mathcal{O}$. ◇

Class-hiding is defined in [HS14] and uses the following oracles and a list $\mathcal{Q}$ to keep track of queried messages M.

$\mathcal{O}^{RM}$: Pick a message $M \xleftarrow{R} (\mathbb{G}_i^*)^\ell$, append it to $\mathcal{Q}$ and return M.

$\mathsf{BGGen}'_{\mathcal{R}}(1^\kappa)$: Output $\mathsf{BG} \leftarrow \mathsf{BGGen}_{\mathcal{R}}(1^\kappa)$.

$\mathsf{KeyGen}'_{\mathcal{R}}(\mathsf{BG}, \ell)$: On input BG and $\ell > 1$, output $(\mathsf{sk}, \mathsf{pk}) \leftarrow \mathsf{KeyGen}_{\mathcal{R}}(\mathsf{BG}, \ell + 2)$.

$\mathsf{Sign}'_{\mathcal{R}}(M, \mathsf{sk})$: On input $M = (M_i)_{i \in [\ell]} \in (\mathbb{G}_1^*)^\ell$ and sk, choose $(R_1, R_2) \overset{R}{\leftarrow} (\mathbb{G}_1^*)^2$, compute $\tau \leftarrow \mathsf{Sign}_{\mathcal{R}}((M, R_1, R_2), \mathsf{sk})$ and output $\sigma \leftarrow (\tau, R_1, R_2)$.

$\mathsf{Verify}'_{\mathcal{R}}(M, \sigma, \mathsf{pk})$: On input $M = (M_i)_{i \in [\ell]} \in (\mathbb{G}_1^*)^\ell$, signature $\sigma \leftarrow (\tau, R_1, R_2)$ and pk, return $\mathsf{Verify}_{\mathcal{R}}((M, R_1, R_2), \tau, \mathsf{pk})$.

$\mathsf{ChgRep}'_{\mathcal{R}}(M, \sigma, \mu, \mathsf{pk})$: Given $M = (M_i)_{i \in [\ell]} \in (\mathbb{G}_1^*)^\ell$, $\sigma \leftarrow (\tau, R_1, R_2)$, $\mu \in \mathbb{Z}_p^*$ and pk, run $((\tilde{M}, \tilde{R}_1, \tilde{R}_2), \tilde{\tau}) \leftarrow \mathsf{ChgRep}_{\mathcal{R}}((M, R_1, R_2), \tau, \mu, \mathsf{pk})$ and output $(\tilde{M}, \tilde{\sigma})$ with $\tilde{\sigma} \leftarrow (\tilde{\tau}, \tilde{R}_1, \tilde{R}_2)$ (or $\perp$ if $\mathsf{ChgRep}_{\mathcal{R}}$ output $\perp$).

$\mathsf{VKey}'_{\mathcal{R}}(\mathsf{sk}, \mathsf{pk})$: Return $\mathsf{VKey}_{\mathcal{R}}(\mathsf{sk}, \mathsf{pk})$.

Scheme 2: Standard-model SPS-EQ construction from Scheme 1

$\mathcal{O}^{RoR}(M, \mathsf{sk}, \mathsf{pk}, b)$: Given message M, key pair $(\mathsf{sk}, \mathsf{pk})$ and bit b, return $\perp$ if $M \notin \mathcal{Q}$. On the first valid call, record M and $\sigma \leftarrow \mathsf{Sign}_{\mathcal{R}}(M, \mathsf{sk})$ and return (M, σ). If later called on $M' \neq M$, return $\perp$; else pick $R \overset{R}{\leftarrow} (\mathbb{G}_i^*)^\ell$ and $\mu \overset{R}{\leftarrow} \mathbb{Z}_p^*$, set $(M_0, \sigma_0) \leftarrow \mathsf{ChgRep}_{\mathcal{R}}(M, \sigma, \mu, \mathsf{pk})$ and $(M_1, \sigma_1) \leftarrow (R, \mathsf{Sign}_{\mathcal{R}}(R, \mathsf{sk}))$ and return (M_b, σ_b).

Definition 9 (Class-hiding)**.** An SPS-EQ scheme $\mathsf{SPS\text{-}EQ}$ on $(\mathbb{G}_i^*)^\ell$ is called *class-hiding* if for all $\ell > 1$ and PPT adversaries $\mathcal{A}$ with oracle access to $\mathcal{O} \leftarrow \{\mathcal{O}^{RM}, \mathcal{O}^{RoR}(\cdot, \mathsf{sk}, \mathsf{pk}, b)\}$ there is a negligible function $\epsilon(\cdot)$ such that

$$\Pr\left[\begin{array}{l} \mathsf{BG} \leftarrow \mathsf{BGGen}_{\mathcal{R}}(1^\kappa),\ b \overset{R}{\leftarrow} \{0,1\}, \\ (\mathsf{st}, \mathsf{sk}, \mathsf{pk}) \leftarrow \mathcal{A}(\mathsf{BG}, \ell),\ b^* \leftarrow \mathcal{A}^{\mathcal{O}}(\mathsf{st}, \mathsf{sk}, \mathsf{pk}) \end{array} : \begin{array}{c} b^* = b\ \wedge \\ \mathsf{VKey}_{\mathcal{R}}(\mathsf{sk}, \mathsf{pk}) = 1 \end{array}\right] - \frac{1}{2} \leq \epsilon(\kappa) \ .$$

◇

Fuchsbauer, Hanser and Slamanig [FHS14] present an EUF-CMA-secure scheme, which we give as Scheme 1, and prove the following.

Theorem 1. *Scheme 1 is EUF-CMA secure against generic forgers and class-hiding under the DDH assumption.*

3 New Results on SPS-EQ

In the following, we present the first standard-model construction of SPS-EQ as modeled in [HS14]. We then introduce new properties to characterize SPS-EQ constructions, strengthening the notion of class-hiding. Finally, we show how to turn any SPS-EQ construction into an SPS construction. This does not only provide a new, efficient standard-model SPS scheme derived from our SPS-EQ scheme; it also allows us to infer optimality of the SPS-EQ scheme from [FHS14], (Scheme 1) and the impossibility of basing its EUF-CMA security on non-interactive assumptions.

3.1 A Standard-Model SPS-EQ Construction

Following the approach by Abe et al. [AGHO11], we construct from scheme $\mathsf{SPS\text{-}EQ}$, given as Scheme 1, an SPS-EQ scheme $\mathsf{SPS\text{-}EQ}'$, given as Scheme 2, and prove that it satisfies EUF-CMA and class-hiding, both under non-interactive assumptions.

The scheme for ℓ-length messages is simply Scheme 1 with message space $(\mathbb{G}_1^*)^{\ell+2}$, where before signing a message one appends two random group elements to it. Scheme 2 features constant-size signatures ($4\ \mathbb{G}_1 + 1\ \mathbb{G}_2$ elements), has public keys of size $\ell + 2$ and still uses 2 PPEs for verification.

Unforgeability follows from a q-type assumption that states that Scheme 1 for $\ell = 2$ is secure against *random-message attacks*. (That is, no PPT adversary, given the public key and signatures on q random messages, can, with non-negligible probability, output a message-signature pair for an equivalence class that was not signed.) Class-hiding follows from class-hiding of Scheme 1. Both proofs can be found in Appendix A.

3.2 Perfect Adaption of Signatures

We now introduce new definitions characterizing the output distribution of $\mathsf{ChgRep}_{\mathcal{R}}$, which lead to stronger notions than class-hiding. The latter only guarantees that given an *honestly* generated signature σ on M, the output $(\mu M, \sigma')$ of $\mathsf{ChgRep}_{\mathcal{R}}$ for a random μ looks like a random message-signature pair. This however does not protect a user against a signer when the user randomizes a pair obtained from the signer. We thus explicitly require that an adaption of any valid (not necessarily honestly generated) signature is distributed like a fresh signature.

Definition 10 (Perfect adaption of signatures)**.** $\mathsf{SPS\text{-}EQ}$ on $(\mathbb{G}_i^*)^\ell$ *perfectly adapts signatures* if for all tuples $(\mathsf{sk}, \mathsf{pk}, M, \sigma, \mu)$ with

$$\mathsf{VKey}_{\mathcal{R}}(\mathsf{sk}, \mathsf{pk}) = 1 \qquad \mathsf{Verify}_{\mathcal{R}}(M, \sigma, \mathsf{pk}) = 1 \qquad M \in (\mathbb{G}_i^*)^\ell \qquad \mu \in \mathbb{Z}_p^*$$

$\mathsf{ChgRep}_{\mathcal{R}}(M, \sigma, \mu, \mathsf{pk})$ and $(\mu M, \mathsf{Sign}_{\mathcal{R}}(\mu M, \mathsf{sk}))$ are identically distributed. ◊

We now show the relation between Def. 10 and 9. The following is proven analogously to the proof of class-hiding of Scheme 1 in [FHS14].

Proposition 1. *Let* $\mathsf{SPS\text{-}EQ}$ *be an SPS-EQ scheme on* $(\mathbb{G}_i^*)^\ell$, $\ell > 1$, *with perfect adaption of signatures. If* $M \xleftarrow{R} [M]_{\mathcal{R}}$ *is computationally indistinguishable from* $M \xleftarrow{R} (\mathbb{G}_i^*)^\ell$ *then* $\mathsf{SPS\text{-}EQ}$ *is class-hiding.*

Corollary 1. *If the DDH assumption holds in* $\mathbb{G}_i$ *then any SPS-EQ scheme on* $(\mathbb{G}_i^*)^\ell$ *satisfying Def. 10 is class-hiding (Def. 9).*

We note that the converse is not true, as witnessed by Scheme 2: it satisfies class-hiding, but the discrete logs of (R_1, R_2) contained in a signature σ have the same ratio as those of $(\tilde{R}_1, \tilde{R}_2)$ from the output of $\mathsf{ChgRep}_{\mathcal{R}}$.

Maliciously chosen keys. Whereas Def. 10 strengthens Def. 9 in that it considers maliciously generated signatures, the next definition strengthens this further by considering maliciously generated public keys. As there might not even be a corresponding signing key, we cannot compare the outputs of $\mathsf{ChgRep}_{\mathcal{R}}$ to those of $\mathsf{Sign}_{\mathcal{R}}$. We therefore require that $\mathsf{ChgRep}_{\mathcal{R}}$ outputs a random element that satisfies verification.

Definition 11 (Perfect adaption under malicious keys)**.** $\mathsf{SPS\text{-}EQ}$ on $(\mathbb{G}_i^*)^\ell$ *perfectly adapts signatures under malicious keys* if for all tuples $(\mathsf{pk}, M, \sigma, \mu)$ with

$$\mathsf{Verify}_{\mathcal{R}}(M, \sigma, \mathsf{pk}) = 1 \qquad M \in (\mathbb{G}_i^*)^\ell \qquad \mu \in \mathbb{Z}_p^* \tag{1}$$

we have that $\mathsf{ChgRep}_{\mathcal{R}}(M, \sigma, \mu, \mathsf{pk})$ outputs $(\mu M, \sigma')$ such that σ' is a random element in the space of signatures, conditioned on $\mathsf{Verify}_{\mathcal{R}}(\mu M, \sigma', \mathsf{pk}) = 1$. ◊

Proposition 2. *Scheme 1, from [FHS14], satisfies both Definitions 10 and 11.*

Proof sketch. For any $M \in (\mathbb{G}_1^*)^\ell$ and $\mathsf{pk} \in (\mathbb{G}_2^*)^\ell$, let $(x_i)_{i\in[\ell]}$ be s.t. $\mathsf{pk} = (x_i\hat{P})_{i\in[\ell]}$. A signature $(Z, Y, \hat{Y}) \in \mathbb{G}_1 \times \mathbb{G}_1^* \times \mathbb{G}_2^*$ satisfying $\mathsf{Verify}_\mathcal{R}(M, (Z, Y, \hat{Y}), \mathsf{pk}) = 1$ must be of the form $(Z = y\sum x_iM_i, Y = \frac{1}{y}P, \hat{Y} = \frac{1}{y}\hat{P})$ for some $y \in \mathbb{Z}_p^*$. $\mathsf{ChgRep}_\mathcal{R}$ outputs $\sigma' = (y\psi\sum x_i\mu M_i, \frac{1}{y\psi}P, \frac{1}{y\psi}\hat{P})$, which is a random element in $\mathbb{G}_1 \times \mathbb{G}_1^* \times \mathbb{G}_2^*$ satisfying $\mathsf{Verify}_\mathcal{R}(M, \sigma', \mathsf{pk}) = 1$. □

3.3 From SPS-EQ to (Rerandomizable) SPS Schemes

We now show how *any* EUF-CMA-secure SPS-EQ scheme that signs equivalence classes of $(\mathbb{G}_i^*)^{\ell+1}$ with $\ell > 0$ can be turned into an EUF-CMA-secure SPS scheme signing vectors of $(\mathbb{G}_i^*)^\ell$. (We note that SPS schemes typically allow messages from $\mathbb{G}_1$ and/or $\mathbb{G}_2$, which is preferable when used in combination with Groth-Sahai proofs.) The transformation works by embedding messages $(M_i)_{i\in[\ell]} \in (\mathbb{G}_i^*)^\ell$ into $(\mathbb{G}_i^*)^{\ell+1}$ as $M' = ((M_i)_{i\in[\ell]}, P)$ and signing M'. To verify a signature σ on a message $(M_i)_{i\in[\ell]} \in (\mathbb{G}_i^*)^\ell$ under key pk, one checks whether $\mathsf{Verify}_\mathcal{R}(((M_i)_{i\in[\ell]}, P), \sigma, \mathsf{pk}) = 1$.

What we have done is to allow only one single representative of each class, namely the one with P as its last element, a procedure we call *normalization*. EUF-CMA of the SPS-EQ states that no adversary can produce a signature on a message from an unqueried class, which therefore implies EUF-CMA of the resulting SPS scheme.

Moreover, from any SPS-EQ with perfect adaption of signatures the above transformation yields a rerandomizable SPS scheme, since signatures can be rerandomized by running $\mathsf{ChgRep}_\mathcal{R}$ for $\mu = 1$ (Def. 10 guarantees that this outputs a random signature). This also means that the lower bounds for SPS over Type-3 groups given by Abe et al. in [AGHO11, AGO11] carry over to SPS-EQ: any SPS must use at least 2 PPEs for verification and must have at least 3 signature elements, which cannot be from the same group. Moreover, EUF-CMA security of optimal (that is, 3-element-signature) SPS-EQ schemes cannot be reduced to non-interactive assumptions.

Finally, let us investigate the possibility of SPS-EQ in the Type-1 and Type-2 pairing setting and implied lower bounds. Class-hiding requires the DDH assumption to hold on the message space. This excludes the Type-1 setting, while in Type-2 settings the message space must be $(\mathbb{G}_1^*)^\ell$. In [AGOT14] Abe et al. identified the following lower bounds for Type-2 SPS schemes with messages in $\mathbb{G}_1$: 2 PPEs for verification and 3 group elements for signatures. The above transformation converts a Type-2 SPS-EQ into a Type-2 SPS, hence these optimality criteria apply to Type-2 SPS-EQ schemes as well.

Implications. Applying the above transformation to the SPS-EQ scheme from [FHS14] (Scheme 1) yields a perfectly rerandomizable SPS scheme in Type-3 groups with constant-size signatures of unilateral length-ℓ message vectors and public keys of size $\ell+1$. Scheme 1 is optimal as it only uses 2 PPEs and its signatures consist of 3 bilateral group elements. Hence, by [AGO11] there is no reduction of its EUF-CMA security to a non-interactive assumption and the generic group model proof in [FHS14] is the best one can achieve.

Applying our transformation to Scheme 2 yields a new standard-model SPS construction for unilateral length-ℓ message vectors in Type-3 groups. It has constant-size signatures (4 $\mathbb{G}_1$ + 1 $\mathbb{G}_2$ elements), a public key of size $\ell + 3$ and uses 2 PPEs for verification; it is therefore almost as efficient as the best known direct SPS construction from non-interactive assumptions in [AGHO11], whose signatures consist of 3 $\mathbb{G}_1$ + 1 $\mathbb{G}_2$ elements. Scheme 2 is partially rerandomizable [AFG+10], whereas the scheme in [AGHO11] is not.

4 Blind Signatures from SPS-EQ

We first present the abstract model for blind signature schemes. Security is defined by unforgeability and blindness and was initially studied in [PS00, JLO97] and then strengthened in [FS09, SU12].

Definition 12 (Blind signature scheme)**.** A blind signature scheme BS consists of the following PPT algorithms:

$\mathsf{KeyGen}_{\mathsf{BS}}(1^\kappa)$, on input κ, returns a key pair $(\mathsf{sk}, \mathsf{pk})$. The security parameter κ is also an (implicit) input to the following algorithms.

$(\mathcal{U}_{\mathsf{BS}}(m, \mathsf{pk}), \mathcal{S}_{\mathsf{BS}}(\mathsf{sk}))$ are run by a user and a signer, who interact during execution. $\mathcal{U}_{\mathsf{BS}}$ gets input a message m and a public key pk and $\mathcal{S}_{\mathsf{BS}}$ has input a secret key sk. At the end $\mathcal{U}_{\mathsf{BS}}$ outputs σ, a signature on m, or $\perp$ if the interaction was not successful.

$\mathsf{Verify}_{\mathsf{BS}}(m, \sigma, \mathsf{pk})$ is deterministic and given a message-signature pair (m, σ) and a public key pk outputs 1 if σ is valid on m under pk and 0 otherwise. $\diamondsuit$

A blind signature scheme BS must satisfy *correctness*, *unforgeability* and *blindness*.

Definition 13 (Correctness)**.** A blind signature scheme BS is *correct* if for all $\kappa \in \mathbb{N}$, all $(\mathsf{sk}, \mathsf{pk}) \leftarrow \mathsf{KeyGen}_{\mathsf{BS}}(1^\kappa)$, all messages m and $\sigma \leftarrow (\mathcal{U}_{\mathsf{BS}}(m, \mathsf{pk}), \mathcal{S}_{\mathsf{BS}}(\mathsf{sk}))$ it holds that $\mathsf{Verify}_{\mathsf{BS}}(m, \sigma, \mathsf{pk}) = 1$. $\diamondsuit$

Definition 14 (Unforgeability)**.** BS is *unforgeable* if for all PPT algorithms $\mathcal{A}$ having access to a signer oracle, there is a negligible function $\epsilon(\cdot)$ such that:

$$\Pr\left[\begin{array}{l} (\mathsf{sk}, \mathsf{pk}) \leftarrow \mathsf{KeyGen}_{\mathsf{BS}}(1^\kappa), \\ (m_i^*, \sigma_i^*)_{i=1}^{k+1} \leftarrow \mathcal{A}^{(\cdot, \mathcal{S}_{\mathsf{BS}}(\mathsf{sk}))}(\mathsf{pk}) \end{array} : \begin{array}{c} m_i^* \neq m_j^* \ \forall i, j \in [k+1], i \neq j \ \wedge \\ \mathsf{Verify}_{\mathsf{BS}}(m_i^*, \sigma_i^*, \mathsf{pk}) = 1 \ \forall i \in [k+1] \end{array}\right] \leq \epsilon(\kappa) \ ,$$

where k is the number of completed interactions with the oracle. $\diamondsuit$

There are several flavors of blindness. The strongest definition is blindness in the *malicious signer* model [ANN06, Oka06], which allows the adversary to create pk, whereas in the *honest-signer* model the key pair is set up by the experiment. We prove our construction secure under the stronger notion, which was also considered by the recent round-optimal standard-model constructions [GRS$^+$11, GG14].

Definition 15 (Blindness)**.** BS is called *blind* if for all PPT algorithms $\mathcal{A}$ with one-time access to two user oracles, there is a negligible function $\epsilon(\cdot)$ such that:

$$\Pr\left[\begin{array}{l} b \xleftarrow{R} \{0,1\}, \ (\mathsf{pk}, m_0, m_1, \mathsf{st}) \leftarrow \mathcal{A}(1^\kappa), \\ \mathsf{st} \leftarrow \mathcal{A}^{(\mathcal{U}_{\mathsf{BS}}(m_b, \mathsf{pk}), \cdot)^{(1)}, (\mathcal{U}_{\mathsf{BS}}(m_{1-b}, \mathsf{pk}), \cdot)^{(1)}}(\mathsf{st}), \\ \text{Let } \sigma_b \text{ and } \sigma_{1-b} \text{ be the resp. outputs of } \mathcal{U}_{\mathsf{BS}}, \\ \text{If } \sigma_0 = \perp \text{ or } \sigma_1 = \perp \text{ then } (\sigma_0, \sigma_1) \leftarrow (\perp, \perp), \\ b^* \leftarrow \mathcal{A}(\mathsf{st}, \sigma_0, \sigma_1) \end{array} : \quad b^* = b \right] - \frac{1}{2} \leq \epsilon(\kappa) \ .$$

$\diamondsuit$

4.1 Construction

Our construction uses commitments to the messages and SPS-EQ to sign these commitments and to perform blinding and unblinding. Signing an equivalence class with an SPS-EQ scheme lets one derive a signature for arbitrary representatives of this class without knowing the private signing key. This concept provides an elegant way to realize a blind signing process as follows.

$\mathsf{KeyGen}_{\mathsf{BS}}(1^\kappa)$: Compute $\mathsf{BG} \leftarrow \mathsf{BGGen}_{\mathcal{R}}(1^\kappa)$, $(\mathsf{sk}, \mathsf{pk}_{\mathcal{R}}) \xleftarrow{R} \mathsf{KeyGen}_{\mathcal{R}}(\mathsf{BG}, \ell = 2)$, pick $q \xleftarrow{R} \mathbb{Z}_p^*$ and set $Q \leftarrow qP$, $\hat{Q} \leftarrow q\hat{P}$. Output $(\mathsf{sk}, \mathsf{pk} = (\mathsf{pk}_{\mathcal{R}}, Q, \hat{Q}))$.

$\mathcal{U}_{\mathsf{BS}}^{(1)}(m, \mathsf{pk})$: Given $\mathsf{pk} = (\mathsf{pk}_{\mathcal{R}}, Q, \hat{Q})$ and $m \in \mathbb{Z}_p$, compute $\mathsf{BG} \leftarrow \mathsf{BGGen}_{\mathcal{R}}(1^\kappa)$. If $Q = 0_{\mathbb{G}_1}$ or $e(Q, \hat{P}) \neq e(P, \hat{Q})$, return $\perp$; else choose $s \xleftarrow{R} \mathbb{Z}_p^*$ and $r \xleftarrow{R} \mathbb{Z}_p$ s.t. $mP + rQ \neq 0_{\mathbb{G}_1}$ and output

$$M \leftarrow (s(mP + rQ), sP) \qquad \mathsf{st} \leftarrow (\mathsf{BG}, \mathsf{pk}_{\mathcal{R}}, Q, M, r, s)$$

$\mathcal{S}_{\mathsf{BS}}(M, \mathsf{sk})$: Given $M \in (\mathbb{G}_1^*)^2$ and secret key sk, output $\pi \leftarrow \mathsf{Sign}_{\mathcal{R}}(M, \mathsf{sk})$.

$\mathcal{U}_{\mathsf{BS}}^{(2)}(\mathsf{st}, \pi)$: Parse st as $(\mathsf{BG}, \mathsf{pk}_{\mathcal{R}}, Q, M, r, s)$. If $\mathsf{Verify}_{\mathcal{R}}(M, \pi, \mathsf{pk}_{\mathcal{R}}) = 0$ then return $\perp$. Run $((mP + rQ, P), \sigma) \leftarrow \mathsf{ChgRep}_{\mathcal{R}}(M, \pi, \frac{1}{s}, \mathsf{pk}_{\mathcal{R}})$ and output $\tau \leftarrow (\sigma, rP, rQ)$.

$\mathsf{Verify}_{\mathsf{BS}}(m, \tau, \mathsf{pk})$: Given $m \in \mathbb{Z}_p^*$, blind signature $\tau = (\sigma, R, T)$ and $\mathsf{pk} = (\mathsf{pk}_{\mathcal{R}}, Q, \hat{Q})$, with $Q \neq 0_{\mathbb{G}_1}$ and $e(Q, \hat{P}) = e(P, \hat{Q})$, output 1 if the following holds and 0 otherwise.

$$\mathsf{Verify}_{\mathcal{R}}((mP + T, P), \sigma, \mathsf{pk}_{\mathcal{R}}) = 1 \qquad e(T, \hat{P}) = e(R, \hat{Q})$$

Scheme 3: Blind signature scheme from SPS-EQ

The signer's key contains an element Q under which the obtainer makes a Pedersen commitment $C = mP + rQ$ to the message m. (Since the commitment is perfectly hiding, the signer can be aware of q with $Q = qP$.) The obtainer then forms a vector (C, P), which can be seen as the canonical representative of equivalence class $[(C, P)]_{\mathcal{R}}$. Next, she picks $s \xleftarrow{R} \mathbb{Z}_p^*$ and moves (C, P) to a random representative (sC, sP), which hides C. She sends (sC, sP) to the signer and receives an SPS-EQ signature on it, from which she can derive a signature on the original message (C, P), which she can publish together with an opening of C. As verification will check validity of the SPS-EQ signature on a message ending with P, the unblinding is unambiguous.

Let us now discuss how the user opens the Pedersen commitment $C = mP + rQ$. Publishing (m, r) directly would break blindness of the scheme (a signer could link a pair $M = (D, S)$, received during signing, to a signature by checking whether $D = mS + rqS$). We therefore define a tweaked opening, for which we include $\hat{Q} = q\hat{P}$ in addition to $Q = qP$ in the signer's public key. We define the opening as (m, rP), which can be checked via the pairing equation $e(C - mP, \hat{P}) = e(rP, \hat{Q})$. This opening is still computationally binding under the co-DHI$_1^*$ assumption (in contrast to standard Pedersen commitments, which are binding under the discrete-log assumption). Hiding of the commitment still holds unconditionally, and we will prove the constructed blind-signature scheme secure in the malicious-signer model without requiring a trusted setup.

The scheme is presented as Scheme 3. (Note that for simplicity the blind signature contains $T = rQ$ instead of C.) Correctness follows by inspection.

4.2 Security

Theorem 2. *If the underlying SPS-EQ scheme is EUF-CMA secure and the co-DHI$_1^*$ assumption holds then Scheme 3 is unforgeable.*

The proof, which is given in in Appendix B, follows the intuition that a forger must either forge an SPS-EQ signature on a new commitment or open a commitment in two different ways. The reduction has a natural security loss proportional to the number of signing queries.

Blindness. For the honest-signer model, blindness follows from the DDH assumption and perfect adaption of signatures (Def. 10) of the underlying SPS-EQ scheme. Let $Q \leftarrow qP$ and let q be part of the signing key, and let (P, rP, sP, tP) be a DDH instance. In the blindness game we compute M as $(m \cdot sP + q \cdot tP, sP)$. When the adversary returns a signature on M, we must adapt it to the unblinded message—which we cannot do as we do not know the blinding factor s. By perfect adaption however, an adapted signature is distributed as a fresh signature on the unblinded message, so, knowing the secret key, we can compute a signature σ on $(m{\cdot}P{+}q{\cdot}rP, P)$ and return the blind signature $(\sigma, rP, q{\cdot}rP)$. If the DDH instance was *real*, i.e., $t = s \cdot r$, then we perfectly simulated the game; if t was random then the adversary's view during issuing was independent of m.

For blindness in the malicious-signer model, we have to deal with two obstacles. (1) We do not have access to the adversarially generated signing key, meaning we cannot recompute the signature on the unblinded message. (2) The adversarially generated public-key values $Q, \hat{Q}$ do not allow us to embed a DDH instance for blinding and unblinding.

We overcome (1) by using the adversary $\mathcal{A}$ itself as a signing oracle by rewinding it. We first run $\mathcal{A}$ to obtain a signature on $(s'(mP + rQ), s'P)$, which, knowing s', we can transform into a signature on $(mP + rQ, P)$. We then rewind $\mathcal{A}$ to the point after outputting its public key and run it again, this time embedding our challenge. In the second run we cannot transform the received signature, instead we use the signature from the first run, which is distributed identically, due to perfect adaption under malicious keys (Def. 11) of the SPS-EQ scheme.

To deal with the second obstacle, we use an interactive variant of the DDH assumption: Instead of being given P, rP, sP and having to distinguish rsP from random, the adversary, for some Q of its choice, is given rP, rQ, sP and must distinguish rsQ from random.

Definition 16 (Assumption 1)**.** Let BGGen be a bilinear-group generator that outputs $\mathsf{BG} = (p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e, P_1 = P, P_2 = \hat{P})$. We assume that for all PPT algorithms $\mathcal{A}$ there is a negligible function $\epsilon(\cdot)$ such that:

$$\Pr\left[\begin{array}{l} b \xleftarrow{R} \{0,1\},\ \mathsf{BG} = \mathsf{BGGen}_{\mathcal{R}}(1^\kappa) \\ (\mathsf{st}, Q, \hat{Q}) \leftarrow \mathcal{A}(1^\kappa),\ \ r, s, t \xleftarrow{R} \mathbb{Z}_p \\ b^* \leftarrow \mathcal{A}(\mathsf{st}, rP, rQ, sP, ((1-b)\cdot t + b\cdot rs)Q) \end{array} : \begin{array}{rl} e(Q, \hat{P}) & = e(P, \hat{Q}) \\ b^* & = b \end{array}\right] - \frac{1}{2} \leq \epsilon(\kappa) \ .$$

$\diamondsuit$

Proposition 3. *The assumption in Def. 16 holds in generic groups and reaches the optimal, quadratic simulation-error bound.*

Theorem 3. *If the underlying SPS-EQ scheme has perfect adaption of signatures under malicious keys and Assumption 1 holds then Scheme 3 is blind.*

The proofs can be found in Appendices F and C, respectively.

4.3 Discussion

Basing our scheme on non-interactive assumptions. Fischlin and Schröder [FS10] show that the unforgeability of a blind-signature scheme cannot be based on non-interactive hardness assumptions if (1) the scheme has 3 moves or less, (2) its blindness holds statistically and (3) from a transcript one can efficiently decide whether the interaction yielded a valid blind signature. Our scheme satisfies (1) and (3), whereas blindness only holds computationally.

They extend their result in [FS10] to computationally blind schemes that meet the following conditions: (4) One can efficiently check whether a public key has a matching secret key; this is the case in our setting because of group-membership tests and pairings. (5) Blindness needs to hold relative to a forgery oracle. As written in [FS10], this does e.g. not hold for Abe's scheme [Abe01], where unforgeability is based on the discrete-log problem and blindness on the DDH problem.

This is the case in our construction too (as one can forge signatures by solving discrete logarithms), hence the impossibility result does not apply to our scheme. Our blind signature construction is black-box from any SPS-EQ with perfect adaption under malicious keys (Def. 11). However, the only known such scheme is the one from [FHS14], which is EUF-CMA secure in the generic-group model, that is, it is based on an interactive assumption. Plugging this scheme into Scheme 3 yields a round-optimal blind signature scheme with unforgeability under this interactive assumption and co-DHI$_1^*$, and blindness (under adversarially chosen keys) under Assumption 1 (Def. 16), which is also interactive.

To construct a scheme under non-interactive assumptions, we would thus have to base blindness on a non-interactive assumption; and find an SPS-EQ scheme satisfying Def. 11 whose unforgeability is proven under a non-interactive assumption.

Efficiency of the construction. When instantiating our blind-signature construction with the SPS-EQ scheme from [FHS14] (given as Scheme 1), which we showed optimal, this yields a public key size of $1\ \mathbb{G}_1 + 3\ \mathbb{G}_2$, a communication complexity of $4\ \mathbb{G}_1 + 1\ \mathbb{G}_2$ and a signature size of $4\ \mathbb{G}_1 + 1\ \mathbb{G}_2$ elements. For a 80-bit security setting, a blind signature has thus 120 Bytes.

The most efficient scheme from standard assumptions is based on DLIN [GG14]. Ignoring the increase of the security parameter due to complexity leveraging, their scheme has a public key size of 43 $\mathbb{G}_1$ elements, communication complexity $18 \log_2 q + 41\ \mathbb{G}_1$ elements (where, e.g., we have $\log_2 q = 155$ when assuming that the adversary runs in $\leq 2^{80}$ steps) and a signature size of 183 $\mathbb{G}_1$ elements.

4.4 Round-Optimal Partially Blind Signatures

Partially blind signatures are an extension of blind signatures, where messages contain *common information* γ, which is agreed between the user and the signer. This requires slight modifications to the unforgeability and blindness notions: An adversary breaks unforgeability if after k signing queries it outputs $k+1$ distinct valid message-signature pairs for the same common information γ^*. In the partial-blindness game m_0 and m_1 must have the same common information γ to prevent the adversary from trivially winning the game. (Formal definitions for partially blind signatures are given in Appendix D.)

Construction. We construct a round-optimal partially blind signature scheme $\mathsf{PBS} = (\mathsf{KeyGen}_{\mathsf{PBS}}, (\mathcal{U}_{\mathsf{PBS}}, \mathcal{S}_{\mathsf{PBS}}), \mathsf{Verify}_{\mathsf{PBS}})$ secure in the standard model from an SPS-EQ scheme $\mathsf{SPS\text{-}EQ}$ by modifying Scheme 3 as follows. To include common information $\gamma \in \mathbb{Z}_p^*$, $\mathsf{SPS\text{-}EQ}$ is set up for $\ell = 3$. On input $M \leftarrow (s(mP + rQ), sP)$, $\mathcal{S}_{\mathsf{PBS}}$ returns a signature for $M \leftarrow (s(mP + rQ), \gamma \cdot sP, sP)$ and $\mathcal{U}_{\mathsf{PBS}}^{(2)}$ additionally checks correctness of the included γ and returns $\perp$ if this is not the case. Otherwise, it runs $((mP + rQ, \gamma P, P), \sigma) \leftarrow \mathsf{ChgRep}_{\mathcal{R}}(M, \pi, \frac{1}{s}, \mathsf{pk})$ and outputs signature $\tau \leftarrow (\sigma, rP, rQ)$ for message m and common information γ. For this construction we obtain the following, whose proofs are analogous to those for Scheme 3and thus omitted.

Theorem 4. *If* $\mathsf{SPS\text{-}EQ}$ *is EUF-CMA secure and the co-DHI$_1^*$ assumption holds, then the resulting partially blind signature scheme is unforgeable.*

Theorem 5. *If* SPS-EQ *has perfect adaption under malicious keys and Assumption 1 holds, then the resulting partially blind signature scheme is partially blind.*

5 One-Show Anonymous Credentials from SPS-EQ

Baldimtsi and Lysyanskaya [BL13a] introduced blind signatures with attributes and show that they directly yield a one-show anonymous credential system in the vein of Brands [Bra00]. In contrast to Brands' original construction, their construction relies on a provably secure three-move blind signature scheme (in the ROM). In this section we show how to construct two-move blind signatures on message vectors, which straightforwardly yield anonymous one-show credentials that are secure in the standard model.

5.1 Blind Signatures on Message Vectors

Our construction BSV of round-optimal blind signatures on message vectors $\boldsymbol{m} \in \mathbb{Z}_p^n$ simply replaces the Pedersen commitment $mP + rQ$ in Scheme 3 with a generalized Pedersen commitment $\sum_{i\in[n]} m_i P_i + rQ$. Thus, $\mathsf{KeyGen}_{\mathsf{BSV}}$, on input $1^\kappa, n$, additionally outputs generators $(P_i)_{i\in[n]}$ and $\mathsf{Verify}_{\mathsf{BSV}}(\boldsymbol{m}, (\sigma, R, T), \mathsf{pk})$ checks $\mathsf{Verify}_{\mathcal{R}}((\sum_{i\in[n]} m_i P_i + T, P), \sigma, \mathsf{pk}_{\mathcal{R}}) = 1$ and $e(T, \hat{P}) = e(R, \hat{Q})$. The construction is presented as Scheme 4 in Appendix E (p. 29). We can prove the following, where the correctness of the scheme, again, follows by inspection.

Theorem 6. *If the underlying SPS-EQ scheme is EUF-CMA secure and the co-DHI$_1^*$ assumption holds then Scheme 4 is unforgeable.*

Proof. The proof is analogous to the unforgeability proof of Scheme 3, except that for Type-2 adversaries, the reduction obtains $\frac{1}{q}P$ from the relation

$$(r_j^* - r_i^*)P = \frac{(\sum_{\ell\in[n]} m_{i,\ell}^* p_\ell - \sum_{\ell\in[n]} m_{j,\ell}^* p_\ell)}{q} P \enspace ,$$

implied by the following: $M_i^* - M_j^* = \left(\sum_{\ell\in[n]} m_{i,\ell}^* P_\ell - \sum_{\ell\in[n]} m_{j,\ell}^* P_\ell\right) + (r_i^* - r_j^*)Q = \left(\sum_{\ell\in[n]} m_{i,\ell}^* p_\ell - \sum_{\ell\in[n]} m_{j,\ell}^* p_\ell\right)P + (r_i^* - r_j^*)Q$. □

Theorem 7. *If the underlying SPS-EQ scheme has perfect adaption under malicious keys and Assumption 1 holds then Scheme 4 is blind.*

The proof is identical to the proof for Scheme 3 and thus omitted.

5.2 Anonymous Credentials Light

The intuition behind our construction is comparable to [BL13a], which roughly works as follows. In the *registration phase*, a user registers (once) a generalized Pedersen commitment C to her attributes and gives a zero-knowledge (ZK) proof of the opening (some attributes may be opened and some may remain concealed). In the *preparation* and *validation phase*, the user engages in a blind-signature-with-attributes protocol for some message m (which is considered the credential serial number) and another commitment C'. C' is a so-called combined commitment obtained from C and a second credential-specific commitment provided by the user. Finally, the credential is the user output of a blind-signature-with-attributes protocol resulting in a signature on message m and a so-called blinded Pedersen commitment C''. The latter contains the same attributes as C, but is unlinkable to C and C'. Showing a credential amounts to presenting C'' along with the blind signature and proving in ZK a desired relation about attributes within C''.

Our construction combines Scheme 4 with efficient ZK proofs and is conceptually simpler than the one in [BL13a]. For issuing, the user sends the issuer a blinded version $M \leftarrow (sC, sP)$ of a commitment C to the user's attributes (M corresponds to the blinded generalized Pedersen commitment in [BL13a]). In addition, the user engages in a ZK proof (denoted PoK) proving knowledge of an opening of C (potentially revealing some of the committed attributes). The user obtains a BSV-signature π on M and turns it into a blind signature σ for commitment C by running $((C,P),\sigma) \leftarrow \mathsf{ChgRep}_{\mathcal{R}}(M, \pi, \frac{1}{s}, \mathsf{pk})$. The credential consists of C, σ and the randomness r used to produce the commitment. It is showed by sending C and σ and proving in ZK a desired relation about attributes within C.

For ease of presentation, we only consider selective attribute disclosure below. We note that proofs for a rich class of relations [CDS94, CM99, BS02] w.r.t. generalized Pederson commitments, as used by our scheme, could be used instead. Henceforth, we denote by S the index set of attributes to be shown and by U those to be withheld. During a showing, a ZK proof of knowledge for a commitment $C = \sum_{i \in [n]} m_i P_i + rQ$ to attributes $(m_i)_{i \in [n]}$ amounts to proving

$$\mathsf{PoK}_{\mathsf{P}}\big\{\big((\alpha_j)_{j \in U}, \beta\big) \;:\; C = \textstyle\sum_{i \in S} m_i P_i + \sum_{j \in U} \alpha_j P_j + \beta Q\big\} \;. \tag{2}$$

The proof for a *blinded* commitment $(A, B) = (sC, sP)$ during the obtain phase is done as follows.

$$\mathsf{PoK}_{\mathsf{BP}}\left\{ \big((\alpha_j)_{j \in U}, \beta, \gamma\big) \;:\; \begin{array}{l} A = \sum_{i \in S} m_i H_i + \sum_{j \in U} \alpha_j H_j + \beta H_Q \;\wedge \\ \bigwedge_{i \in [n]} (H_i = \gamma P_i) \wedge H_Q = \gamma Q \wedge B = \gamma P \end{array} \right\} \;. \tag{3}$$

Here the representation is with respect to bases $H_i = sP_i$, $H_Q = sQ$, which are published and guaranteed to be correctly formed by $\mathsf{PoK}_{\mathsf{BP}}$.[2]

Construction. As we combine Scheme 4 with ZK proofs, we need the following conceptual modifications. The signature $\tau \leftarrow (\sigma, R, T)$ reduces to $\tau \leftarrow \sigma$, since the user provides a ZK-PoK proving knowledge of the randomness r in C. Moreover, verification takes C instead of $\boldsymbol{m}$ as verifiers have only access to the commitment. Consequently, $\mathsf{Verify}_{\mathsf{BSV}}$ of Scheme 4 only runs $\mathsf{Verify}_{\mathcal{R}}$.

Setup. The issuer runs $(\mathsf{sk}, \mathsf{pk}) \leftarrow \mathsf{KeyGen}_{\mathsf{BSV}}(1^\kappa, n)$, where n is the number of attributes in the system, and publishes pk as her public key.

Issuing. A user with attribute values $\boldsymbol{m}$ runs $(M, \mathsf{st}) \leftarrow \mathcal{U}^{(1)}_{\mathsf{BSV}}(\boldsymbol{m}, \mathsf{pk}; (s, r))$ (where (s, r) is the chosen randomness), sends the blinded commitment $M = (sC, sP)$ to the issuer and gives a proof $\mathsf{PoK}_{\mathsf{BP}}$ from (3) that M commits to $\boldsymbol{m}$ (where the sets U and S depend on the application). The issuer returns $\pi \leftarrow \mathcal{S}_{\mathsf{BSV}}(M, \mathsf{sk})$ and after running $\sigma \leftarrow \mathcal{U}^{(2)}_{\mathsf{BSV}}(\mathsf{st}, \pi)$ (the outputs rP and rQ are not needed), the user holds a credential (C, σ, r).

Showing. Assume a user with credential (C, σ, r) to the attributes $\boldsymbol{m} = (m_i)_{i \in [n]}$ wants to conduct a selective showing of attributes with a verifier who holds the issuer's public key pk. They engage in a proof $\mathsf{PoK}_{\mathsf{P}}$ from (2) and the verifier additionally checks the

[2] In the blindness game, given $B = sP$ from a DDH instance, these bases are simulated as $H_j \leftarrow p_j B$ and $H_Q \leftarrow qB$. We can even prove security in the malicious-signer model by extending the assumption from Def. 16: in addition to Q the adversary outputs $(P_i)_{i \in [n]}$ and receives $(sP_i)_{i \in [n]}$ and sQ.

signature for the credential by running $\mathsf{Verify}_{\mathsf{BSV}}(C, \sigma, \mathsf{pk})$. If both verifications succeed, the verifier accepts the showing.

Let us finally note that there is no formal security model for one-show credentials. Theorem 2 in [BL13a] informally states that a secure commitment scheme together with a blind signature scheme with attributes implies a one-show credential system. Using the same argumentation as [BL13a], our construction yields a one-show credential system in the standard model.

Acknowledgements

We would like to thank the anonymous reviewers of CRYPTO'15 for their valuable comments.

References

[Abe01] Masayuki Abe. A Secure Three-Move Blind Signature Scheme for Polynomially Many Signatures. In *EUROCRYPT*, volume 2045 of *LNCS*, pages 136–151. Springer, 2001.

[AFG$^+$10] Masayuki Abe, Georg Fuchsbauer, Jens Groth, Kristiyan Haralambiev, and Miyako Ohkubo. Structure-Preserving Signatures and Commitments to Group Elements. In *CRYPTO*, volume 6223 of *LNCS*, pages 209–236. Springer, 2010.

[AGHO11] Masayuki Abe, Jens Groth, Kristiyan Haralambiev, and Miyako Ohkubo. Optimal Structure-Preserving Signatures in Asymmetric Bilinear Groups. In *CRYPTO*, volume 6841 of *LNCS*, pages 649–666. Springer, 2011.

[AGO11] Masayuki Abe, Jens Groth, and Miyako Ohkubo. Separating Short Structure-Preserving Signatures from Non-interactive Assumptions. In *ASIACRYPT*, volume 7073 of *LNCS*, pages 628–646. Springer, 2011.

[AGOT14] Masayuki Abe, Jens Groth, Miyako Ohkubo, and Mehdi Tibouchi. Structure-Preserving Signatures from Type II Pairings. In *CRYPTO*, volume 8616 of *LNCS*, pages 390–407. Springer, 2014.

[ANN06] Michel Abdalla, Chanathip Namprempre, and Gregory Neven. On the (Im)possibility of Blind Message Authentication Codes. In *CT-RSA*, volume 3860 of *LNCS*, pages 262–279. Springer, 2006.

[AO00] Masayuki Abe and Tatsuaki Okamoto. Provably Secure Partially Blind Signatures. In *CRYPTO*, volume 1880 of *LNCS*, pages 271–286. Springer, 2000.

[BFPV11] Olivier Blazy, Georg Fuchsbauer, David Pointcheval, and Damien Vergnaud. Signatures on Randomizable Ciphertexts. In *PKC*, volume 6571 of *LNCS*, pages 403–422. Springer, 2011.

[BL13a] Foteini Baldimtsi and Anna Lysyanskaya. Anonymous Credentials Light. In *ACM CCS*, 2013.

[BL13b] Foteini Baldimtsi and Anna Lysyanskaya. On the Security of One-Witness Blind Signature Schemes. In *ASIACRYPT (2)*, volume 8270 of *LNCS*, pages 82–99. Springer, 2013.

[BN05] Paulo S. L. M. Barreto and Michael Naehrig. Pairing-friendly elliptic curves of prime order. In *SAC*, pages 319–331, 2005.

[BNPS03] Mihir Bellare, Chanathip Namprempre, David Pointcheval, and Michael Semanko. The One-More-RSA-Inversion Problems and the Security of Chaum's Blind Signature Scheme. *J. Cryptology*, 16(3):185–215, 2003.

[Bol03] Alexandra Boldyreva. Threshold Signatures, Multisignatures and Blind Signatures Based on the Gap-Diffie-Hellman-Group Signature Scheme. In *PKC*, volume 2567 of *LNCS*, pages 31–46. Springer, 2003.

[BP10] Stefan Brands and Christian Paquin. U-Prove Cryptographic Specification v1. 2010.

[BPV12] Olivier Blazy, David Pointcheval, and Damien Vergnaud. Compact Round-Optimal Partially-Blind Signatures. In *SCN*, volume 7485 of *LNCS*, pages 95–112. Springer, 2012.

[Bra00] Stefan Brands. *Rethinking Public-Key Infrastructures and Digital Certificates: Building in Privacy.* MIT Press, 2000.

[BS02] Emmanuel Bresson and Jacques Stern. Proofs of Knowledge for Non-monotone Discrete-Log Formulae and Applications. In *ISC*, volume 2433 of *LNCS*, pages 272–288. Springer, 2002.

[CDH12] Jan Camenisch, Maria Dubovitskaya, and Kristiyan Haralambiev. Efficient Structure-Preserving Signature Scheme from Standard Assumptions. In *SCN*, volume 7485 of *LNCS*, pages 76–94. Springer, 2012.

[CDS94] Ronald Cramer, Ivan Damgård, and Berry Schoenmakers. Proofs of Partial Knowledge and Simplified Design of Witness Hiding Protocols. In *CRYPTO'94*, volume 839 of *LNCS*, pages 174–187. Springer, 1994.

[Cha82] David Chaum. Blind Signatures for Untraceable Payments. In *CRYPTO'82*, pages 199–203. Plenum Press, 1982.

[Cha83] David Chaum. Blind signature system. In *CRYPTO*, page 153, 1983.

[CKW04] Jan Camenisch, Maciej Koprowski, and Bogdan Warinschi. Efficient Blind Signatures Without Random Oracles. In *SCN*, pages 134–148, 2004.

[CM99] Jan Camenisch and Markus Michels. Proving in Zero-Knowledge that a Number Is the Product of Two Safe Primes. In *EUROCRYPT*, volume 1592 of *LNCS*, pages 107–122. Springer, 1999.

[CM11] Sanjit Chatterjee and Alfred Menezes. On cryptographic protocols employing asymmetric pairings - the role of ψ revisited. *Discrete Applied Mathematics*, 159(13):1311–1322, 2011.

[FHS14] Georg Fuchsbauer, Christian Hanser, and Daniel Slamanig. EUF-CMA-Secure Structure-Preserving Signatures on Equivalence Classes. Cryptology ePrint Archive, Report 2014/944, 2014.

[Fis06] Marc Fischlin. Round-Optimal Composable Blind Signatures in the Common Reference String Model. In *CRYPTO*, volume 4117 of *LNCS*, pages 60–77. Springer, 2006.

[FS09] Marc Fischlin and Dominique Schröder. Security of Blind Signatures under Aborts. In *PKC*, volume 5443 of *LNCS*, pages 297–316. Springer, 2009.

[FS10] Marc Fischlin and Dominique Schröder. On the Impossibility of Three-Move Blind Signature Schemes. In *EUROCRYPT*, volume 6110 of *LNCS*, pages 197–215. Springer, 2010.

[Fuc14] Georg Fuchsbauer. Breaking Existential Unforgeability of a Signature Scheme from Asiacrypt 2014. Cryptology ePrint Archive, Report 2014/892, 2014.

[GG14] Sanjam Garg and Divya Gupta. Efficient Round Optimal Blind Signatures. In *EUROCRYPT*, volume 8441 of *LNCS*, pages 477–495. Springer, 2014.

[GRS$^+$11] Sanjam Garg, Vanishree Rao, Amit Sahai, Dominique Schröder, and Dominique Unruh. Round Optimal Blind Signatures. In *CRYPTO*, pages 630–648, 2011.

[GS08] Jens Groth and Amit Sahai. Efficient Non-interactive Proof Systems for Bilinear Groups. In *EUROCRYPT'08*, volume 4965 of *LNCS*, pages 415–432. Springer, 2008.

[GS12] Essam Ghadafi and Nigel P. Smart. Efficient Two-Move Blind Signatures in the Common Reference String Model. In *ISC*, volume 7483 of *LNCS*, pages 274–289. Springer, 2012.

[HKKL07] Carmit Hazay, Jonathan Katz, Chiu-Yuen Koo, and Yehuda Lindell. Concurrently-Secure Blind Signatures Without Random Oracles or Setup Assumptions. In *TCC*, volume 4392 of *LNCS*, pages 323–341. Springer, 2007.

[HS14] Christian Hanser and Daniel Slamanig. Structure-Preserving Signatures on Equivalence Classes and their Application to Anonymous Credentials. In *ASIACRYPT*, 2014. Full version: Cryptology ePrint Archive, Report 2014/705.

[JLO97] Ari Juels, Michael Luby, and Rafail Ostrovsky. Security of Blind Digital Signatures (Extended Abstract). In *CRYPTO '97*, volume 1294 of *LNCS*, pages 150–164. Springer, 1997.

[KZ06] Aggelos Kiayias and Hong-Sheng Zhou. Concurrent Blind Signatures Without Random Oracles. In *SCN*, volume 4116 of *LNCS*, pages 49–62. Springer, 2006.

[Lin03] Yehuda Lindell. Bounded-concurrent secure two-party computation without setup assumptions. In *STOC*, pages 683–692. ACM, 2003.

[LRSW00] Anna Lysyanskaya, Ronald L. Rivest, Amit Sahai, and Stefan Wolf. Pseudonym Systems. In *SAC '00*, volume 1758 of *LNCS*, pages 184–199. Springer, 2000.

[MSF10] Sarah Meiklejohn, Hovav Shacham, and David Mandell Freeman. Limitations on Transformations from Composite-Order to Prime-Order Groups: The Case of Round-Optimal Blind Signatures. In *ASIACRYPT*, volume 6477 of *LNCS*, pages 519–538. Springer, 2010.

[Oka92] Tatsuaki Okamoto. Provably Secure and Practical Identification Schemes and Corresponding Signature Schemes. In *CRYPTO*, volume 740 of *LNCS*, pages 31–53. Springer, 1992.

[Oka06] Tatsuaki Okamoto. Efficient Blind and Partially Blind Signatures Without Random Oracles. In *TCC*, volume 3876 of *LNCS*, pages 80–99. Springer, 2006.

[PS00] David Pointcheval and Jacques Stern. Security Arguments for Digital Signatures and Blind Signatures. *J. Cryptology*, 13(3):361–396, 2000.

[Rüc10] Markus Rückert. Lattice-based blind signatures. In *ASIACRYPT*, pages 413–430, 2010.

[SC12] Jae Hong Seo and Jung Hee Cheon. Beyond the Limitation of Prime-Order Bilinear Groups, and Round Optimal Blind Signatures. In *TCC*, volume 7194 of *LNCS*, pages 133–150. Springer, 2012.

[SU12] Dominique Schröder and Dominique Unruh. Security of Blind Signatures Revisited. In *PKC*, volume 7293 of *LNCS*, pages 662–679. Springer, 2012.

A Security of Scheme 2

In order to prove the EUF-CMA security of Scheme 2, we introduce the following non-interactive q-type assumption. It is derived from Scheme 1 for $\ell = 2$, essentially stating that Scheme 1 is secure against random-message attacks.

Definition 17 (Assumption 2)**.** Given a bilinear group $\mathsf{BG} = (p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e, P, \hat{P})$, $(\hat{V}_1, \hat{V}_2) \xleftarrow{R} (\mathbb{G}_2^*)^2$ and q instances $(N_{j1}, N_{j2}, Z_j, Y_j, \hat{Y}_j) \in (\mathbb{G}_1^*)^2 \times \mathbb{G}_1 \times \mathbb{G}_1^* \times \mathbb{G}_2^*$ which are independently uniformly random conditioned on

$$e(N_{j1}, \hat{V}_1) \cdot e(N_{j2}, \hat{V}_2) = e(Z_j, \hat{Y}_j) \quad \wedge \quad e(Y_j, \hat{P}) = e(P, \hat{Y}_j) \ ,$$

it is hard to output $(N_1^*, N_2^*, Z^*, Y^*, \hat{Y}^*) \in (\mathbb{G}_1^*)^2 \times \mathbb{G}_1 \times \mathbb{G}_1^* \times \mathbb{G}_2^*$ such that $(N_1^*, N_2^*) \neq k \cdot (N_{j1}, N_{j2})$ for all $k \in \mathbb{Z}_p^*$, $j \in [q]$, and

$$e(N_1^*, \hat{V}_1) \cdot e(N_2^*, \hat{V}_2) = e(Z^*, \hat{Y}^*) \quad \wedge \quad e(Y^*, \hat{P}) = e(P, \hat{Y}^*) \ . \qquad \diamondsuit$$

Theorem 1, proven in [FHS14], implies that Assumption 2 holds in the generic group model. When reconsidering the simulation-error analysis in the proof, we see that the degree of all involved polynomials is constant. Therefore, a generic adversary making $O(q)$ queries to the group oracles has probability $O(\frac{q^2}{p})$ of breaking the assumption and thus the assumption reaches the optimal simulation error bound.

We are now going to prove the unforgeability and the class-hiding property of Scheme 2. Correctness follows from correctness of Scheme 1.

Theorem 8. *If Assumption 2 holds then Scheme 2 is an EUF-CMA-secure SPS-EQ scheme.*

Proof. We assume that there is an efficient adversary $\mathcal{A}$ against the unforgeability of Scheme 2 that makes q' signing queries and use $\mathcal{A}$ to build an efficient adversary $\mathcal{B}$ against Assumption 2 for $q = q'$.

$\mathcal{B}$ is given $\mathsf{BG} = (p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e, P, \hat{P})$, $(\hat{V}_1, \hat{V}_2) \in (\mathbb{G}_2^*)^2$ and instances $(N_{j1}, N_{j2}, Z_j, Y_j, \hat{Y}_j)$ for $j \in [q]$. For all $i \in [\ell]$, $\mathcal{B}$ chooses $a_i, b_i \xleftarrow{R} \mathbb{Z}_p$ and computes $\hat{X}_i \leftarrow a_i\hat{V}_1 + b_i\hat{V}_2$. It sets $\hat{X}_{\ell+1} \leftarrow \hat{V}_1$, $\hat{X}_{\ell+2} \leftarrow \hat{V}_2$, $\mathsf{pk} \leftarrow (\hat{X}_i)_{i\in[\ell+2]}$ and runs $\mathcal{A}^{\mathcal{O}(\cdot,\mathsf{sk})}(\mathsf{pk})$. With overwhelming probability all elements $X_1, \ldots, X_{\ell+2}$ are non-trivial, in which case pk is distributed as a key in Scheme 2.

Next, $\mathcal{B}$ simulates $\mathcal{A}$'s queries to its signing oracle $\mathcal{O}(\cdot, \mathsf{sk})$ as follows. On the jth signing query for message $M_j = (M_{ji})_{i\in[\ell]} \in (\mathbb{G}_1^*)^\ell$, $\mathcal{B}$ computes

$$R_{j1} \leftarrow N_{j1} - \textstyle\sum_{i\in[\ell]} a_i M_{ji} \quad \text{and} \quad R_{j2} \leftarrow N_{j2} - \textstyle\sum_{i\in[\ell]} b_i M_{ji} \tag{4}$$

and returns the signature $\sigma_j \leftarrow ((Z_j, Y_j, \hat{Y}_j), R_{j1}, R_{j2})$ to $\mathcal{A}$. Note that the elements R_{j1} and R_{j2} are random, since N_{j1} and N_{j2} from the instance are random. (There is a small simulation error, as N_{j1} is uniform in $\mathbb{G}_1^*$, whereas R_{j1} is uniformly random in $\mathbb{G}_1^* \setminus \{-\sum_{i\in[\ell]} a_i M_{ji}\}$, but this error is negligible.) Moreover, they perfectly mask the scalars a_i and b_i.

Observe that the simulated signature satisfies the first verification equation:

$$\begin{aligned}
&\textstyle\prod_{i\in[\ell]} e(M_{ji}, \hat{X}_i)\, e(R_{j1}, \hat{X}_{\ell+1})\, e(R_{j2}, \hat{X}_{\ell+2}) = \\
&\quad \textstyle\prod_{i\in[\ell]} e(M_{ji}, a_i\hat{V}_1 + b_i\hat{V}_2)\, e(N_{j1} - \sum_{i\in[\ell]} a_i M_{ji}, \hat{V}_1)\, e(N_{j2} - \sum_{i\in[\ell]} b_i M_{ji}, \hat{V}_2) = \\
&\textstyle\prod e(M_{ji}, a_i\hat{V}_1) \prod e(M_{ji}, b_i\hat{V}_2)\, e(N_{j1}, \hat{V}_1)\, e(N_{j2}, \hat{V}_2) \prod e(a_i M_{ji}, \hat{V}_1)^{-1} \prod e(b_i M_{ji}, \hat{V}_2)^{-1} \\
&\qquad = e(N_{j1}, \hat{V}_1)\, e(N_{j2}, \hat{V}_2) = e(Z_j, \hat{Y}_j) \ .
\end{aligned}$$

Since $(Y_j, \hat{Y}_j)$ from the instance are uniformly random in $\mathbb{G}_1^* \times \mathbb{G}_2^*$ conditioned on $e(Y_j, \hat{P}) = e(P, \hat{Y}_j)$ and together with $(M_{ji})_{i\in[\ell]}$, (R_{j1}, R_{j2}) and $(\hat{X}_i)_{i\in[\ell+2]}$, they uniquely determine Z_j as per the above equation, this shows that $(Z_j, Y_j, \hat{Y}_j)$ is a correctly distributed Scheme-1 signature. Furthermore, with overwhelming probability we have that $R_{j1} \neq 0_{\mathbb{G}_1}$ and $R_{j2} \neq 0_{\mathbb{G}_1}$, in which case the signatures $\sigma_j = ((Z_j, Y_j, \hat{Y}_j), R_{j1}, R_{j2})$ are perfectly simulated.

If $\mathcal{A}$ outputs a forgery $(M^*, \sigma^*) = ((M_i^*)_{i\in[\ell]}, (Z^*, Y^*, \hat{Y}^*, R_1^*, R_2^*))$ then $\mathcal{B}$ computes

$$N_1^* \leftarrow R_1^* + \textstyle\sum_{i\in[\ell]} a_i M_i^* \quad \text{and} \quad N_2^* \leftarrow R_2^* + \textstyle\sum_{i\in[\ell]} b_i M_i^* \tag{5}$$

and returns $(N_1^*, N_2^*, Z^*, Y^*, \hat{Y}^*)$.

In order to show that $\mathcal{B}$'s output breaks Assumption 2, we need to show the following: (1) $(N_1^*, N_2^*, Z^*, Y^*, \hat{Y}^*)$ satisfies the last pair of equations in Def. 17; (2) $(N_1^*, N_2^*) \in (\mathbb{G}_1^*)^2$ and (3) $(N_1^*, N_2^*) \neq \mu \cdot (N_{j1}, N_{j2})$ for all $\mu \in \mathbb{Z}_p^*$, $j \in [q]$.

1) We have:

$$\begin{aligned} e(N_1^*, \hat{V}_1)\, e(N_2^*, \hat{V}_2) &\overset{(5)}{=} e(\textstyle\sum_{i\in[\ell]} a_i M_i^*, \hat{V}_1)\, e(\textstyle\sum_{i\in[\ell]} b_i M_i^*, \hat{V}_2)\, e(R_1^*, \hat{V}_1)\, e(R_2^*, \hat{V}_2) \\ &= \textstyle\prod_{i\in[\ell]} e(M_i^*, a_i\hat{V}_i) e(M_i^*, b_i\hat{V}_2)\, e(R_1^*, \hat{V}_1)\, e(R_2^*, \hat{V}_2) \\ &= \textstyle\prod_{i\in[\ell]} e(M_i^*, \hat{X}_i)\, e(R_1^*, \hat{V}_1)\, e(R_2^*, \hat{V}_2) = e(Z^*, \hat{Y}^*) \ , \end{aligned}$$

where the last equation follows from $\mathcal{A}$ outputting a valid signature. Since for the same reason, $e(Y^*, \hat{P}) = e(P, \hat{Y}^*)$, we have that $\mathcal{B}$'s output satisfies the required equations. (Note also that $Y^* \neq 0$ and $\hat{Y}^* \neq 0$ when $\mathcal{A}$'s output is valid.)

2) The only information about $(a_i)_{i\in[\ell]}$ and $(b_i)_{i\in[\ell]}$ revealed to $\mathcal{A}$ is

$$x_i = a_i \cdot v_1 + b_i \cdot v_2 \ , \tag{6}$$

where x_i, v_1 and v_2 are s.t. $\hat{X}_i = x_i\hat{P}$, $\hat{V}_1 = v_1\hat{P}$ and $\hat{V}_2 = v_2\hat{P}$, for all $i \in [\ell]$.

Since $M_i^* \neq 0$ for all $i \in [\ell]$, the probability that either $N_1^* = R_1^* + \sum_{i\in[\ell]} a_i M_i^* = 0$ or $N_2^* = R_2^* + \sum_{i\in[\ell]} b_i M_i^* = 0$ is therefore negligible.

3) Since M^* is a valid forgery, we have that for all $\mu \in \mathbb{Z}_p^*$ and $j \in [q]$: $M^* \neq \mu \cdot M_j$. The reduction could however fail if for some $\mu \in \mathbb{Z}_p$ and $j \in [q]$, we had $(N_1^*, N_2^*) = \mu \cdot (N_{j1}, N_{j2})$, that is

$$n_1^* \cdot n_{j2} = n_2^* \cdot n_{j1} \ , \tag{7}$$

where we let lower-case letters denote the logarithms of the corresponding upper-case letters to the basis P. We now show that even for an unbounded adversary, the probability that this happens is negligible.

$\mathcal{A}$ has no information about $(a_i)_{i\in[\ell]}$, however, by (6), each a_i determines b_i as

$$b_i = x_i v_2^{-1} - v_1 v_2^{-1} a_i \ .$$

Together with (4) and (5) this means that Equation (7) can be written as

$$\begin{aligned} &\big(r_1^* + \textstyle\sum_{i\in[\ell]} a_i m_i^*\big) \cdot \big(r_{j2} + \textstyle\sum_{i\in[\ell]} x_i v_2^{-1} m_{ji} - v_1 v_2^{-1} \textstyle\sum_{i\in[\ell]} a_i m_{ji}\big) \\ &\qquad = \big(r_2^* + \textstyle\sum_{i\in[\ell]} x_i v_2^{-1} m_i^* - v_1 v_2^{-1} \textstyle\sum_{i\in[\ell]} a_i m_i^*\big) \cdot \big(r_{j1} + \textstyle\sum_{i\in[\ell]} a_i m_{ji}\big) \ . \end{aligned}$$

This can be rewritten as (note that the terms containing products of a_i's cancel):

$$\sum_{i\in[\ell]} \Big(- r_1^* v_1 v_2^{-1} m_{ji} + r_{j2} m_i^* + \textstyle\sum_{k\in[\ell]} x_k v_2^{-1} m_{jk} m_i^* \\ + r_{j1} v_1 v_2^{-1} m_i^* - r_2^* m_{ji} - \textstyle\sum_{k\in[\ell]} x_k v_2^{-1} m_{ji} m_k^* \Big) a_i \\ = -r_1^* \big(r_{j2} + \textstyle\sum_{i\in[\ell]} x_i v_2^{-1} m_{ji} \big) + \big(r_2^* + \textstyle\sum_{i\in[\ell]} x_i v_2^{-1} m_i^* \big) r_{j1} \enspace .$$

Since $\mathcal{A}$ has no knowledge of the a_i's, $\mathcal{A}$ can only make the equation be satisfied with non-negligible probability by setting all coefficients of the a_i's to 0. That is, for all $i \in [\ell]$:

$$\big(r_{j2} + r_{j1} v_1 v_2^{-1} + \textstyle\sum_k x_k v_2^{-1} m_{jk} - x_i v_2^{-1} m_{ji} \big) m_i^* - \textstyle\sum_{k\neq i} x_k v_2^{-1} m_{ji} m_k^* \\ = \big(r_1^* v_1 v_2^{-1} + r_2^* \big) m_{ji} \enspace . \qquad (8)$$

We now argue that the above system of ℓ linear equations in the variables $(m_1^*, \ldots, m_\ell^*)$ is regular with overwhelming probability. Indeed, $\mathcal{A}$ can choose the m_{jk}'s contained in the coefficients to its liking, however, it only learns r_{j2} afterward, which is uniformly random (determined via the random N_{j2} from the instance). Thus to the matrix determined by $\mathcal{A}$'s choices a (the same) random element is added to each entry in the diagonal; that is, a random multiple of the unity matrix I is added. It follows from the following claim that this makes the matrix regular with overwhelming probability.

Claim 1. *Let $A \in \mathbb{Z}_p^{\ell\times\ell}$. Then $A + \eta I$ for $\eta \xleftarrow{R} \mathbb{Z}_p$ is regular with overwhelming probability.*

Proof. Consider the Schur decomposition of A, that is, a regular matrix Q and an upper triangular matrix U, such that $A = QUQ^{-1}$. A is regular if all diagonal elements of U are non-zero. $A + \eta I = Q(U + \eta I)Q^{-1}$ is regular if $(U + \eta I)$ has no zeros in the diagonal, which holds with overwhelming probability since the probability that $-\eta$ occurs in the diagonal of U is negligible. □

Let $r_1^*, r_2^*, m_{j1}, \ldots, m_{j\ell}$ be arbitrary. We then argue that the only assignment to $m^* = (m_1^*, \ldots, m_\ell^*)$ that satisfies the equation system in (8) is a multiple of m_j. This however means that the adversary did not win the unforgeability game.

Let $\lambda = (r_1^* v_1 v_2^{-1} + r_2^*)(r_{j2} + r_{j1} v_1 v_2^{-1})^{-1}$. Then $m_i^* \leftarrow \lambda m_{ji}$, for all $i \in [\ell]$, is a solution to equation system in (8):

$$\big(r_{j2} + r_{j1} v_1 v_2^{-1} + \textstyle\sum_k x_k v_2^{-1} m_{jk} - x_i v_2^{-1} m_{ji} \big) \lambda m_{ji} - \textstyle\sum_{k\neq i} x_k v_2^{-1} m_{ji} \lambda m_{jk} \\ = \big(r_{j2} + r_{j1} v_1 v_2^{-1} \big) \lambda m_{ji} = \big(r_1^* v_1 v_2^{-1} + r_2^* \big) m_{ji} \enspace .$$

Since the system is regular with overwhelming probability, this is the only solution, meaning in this case the adversary did not win. With overwhelming probability $\mathcal{B}$ thus returns a pair (N_1^*, N_2^*), which is not the multiple of any pair (N_{j1}, N_{j2}) from the given instance. As we have constructed from an adversary $\mathcal{A}$ breaking unforgeability of Scheme 2 an algorithm $\mathcal{B}$ which breaks Assumption 1 with almost the same probability, this completes the proof. □

Theorem 9. *If Scheme 1 is class-hiding then Scheme 2 is class-hiding.*

Proof. We assume that there is an efficient adversary $\mathcal{A}$ against class-hiding of Scheme 2 with message length ℓ and use $\mathcal{A}$ to build an efficient adversary $\mathcal{B}$ against class-hiding of Scheme 1 with length $\ell + 2$.

$\mathcal{B}$ interacts with a class-hiding challenger $\mathcal{C}$, which creates the bilinear group BG and runs $\mathcal{B}$ on $(\mathsf{BG}, \ell+2)$. $\mathcal{B}$ runs $(\mathsf{st}_{\mathcal{A}}, \mathsf{sk} = (x_i)_{i\in[\ell+2]}, \mathsf{pk} = (\hat{X}_i)_{i\in[\ell+2]}) \leftarrow \mathcal{A}(\mathsf{BG}, \ell)$ and forwards this to $\mathcal{C}$. When $\mathcal{C}$ then runs $\mathcal{B}$ on $(\mathsf{st}_{\mathcal{A}}, \mathsf{sk}, \mathsf{pk})$, $\mathcal{B}$ runs $b^* \leftarrow \mathcal{A}^{\mathcal{O}}(\mathsf{st}_{\mathcal{A}}, \mathsf{sk}, \mathsf{pk})$ and simulates $\mathcal{A}$'s oracles as follows.

On $\mathcal{A}$'s jth call to $\mathcal{O}^{RM}$, $\mathcal{B}$ calls its $\mathcal{O}^{RM}$ oracle to receive $M_j = (M_{ji})_{i\in[\ell+2]}$. $\mathcal{B}$ returns $(M_{ji})_{i\in[\ell]}$ to $\mathcal{A}$ and records $(M_{ji})_{i\in[\ell+2]}$.

When $\mathcal{A}$ calls the $\mathcal{O}^{RoR}$ oracle for message M_j, $\mathcal{B}$ looks for the first occurrence of M_j in its record, retrieves $(M_{ji})_{i\in[\ell+2]}$ and submits it to its $\mathcal{O}^{RoR}$ oracle. (If no entry in $\mathcal{B}$'s record starts with $(M_{j1}, \ldots, M_{j\ell})$ then $\mathcal{B}$ returns $\perp$.) Upon receiving $(M' = (M'_i)_{i\in[\ell+2]}, \sigma')$, $\mathcal{B}$ returns $((M'_i)_{i\in[\ell]}, (\sigma', M'_{\ell+1}, M'_{\ell+2}))$ to $\mathcal{A}$. Finally, $\mathcal{B}$ forwards $\mathcal{A}$'s output b^* to $\mathcal{C}$.

The simulation is perfect: On $\mathcal{A}$'s first valid call $M_j = (M_{ji})_{i\in[\ell]}$ to $\mathcal{O}^{RoR}$, it receives $M = M_j$ and $\sigma = (\sigma', M_{\ell+1}, M_{\ell+2})$, which is distributed as $(M, \mathsf{Sign}'_{\mathcal{R}}(M, \mathsf{sk}))$, since $M_{\ell+1}$, $M_{\ell+2}$ are uniformly random elements (picked by $\mathcal{O}^{RM}$) and σ is a signature on $(M_i)_{i\in[\ell+2]}$, computed by $\mathcal{O}^{RoR}$.

Moreover, if $\mathcal{C}$'s bit $b = 0$ then at all further queries of M to $\mathcal{O}^{RoR}$, $\mathcal{B}$ receives $((M'_i)_{i\in[\ell+2]}, \sigma') \leftarrow \mathsf{ChgRep}_{\mathcal{R}}((M_i)_{i\in[\ell+2]}, \sigma, \mu, \mathsf{pk})$ for $\mu \xleftarrow{R} \mathbb{Z}_p^*$, and sends $((M'_i)_{i\in[\ell]}, (\sigma', M'_{\ell+1}, M'_{\ell+2}))$ to $\mathcal{A}$, which is distributed as the output of $\mathsf{ChgRep}'_{\mathcal{R}}((M_i)_{i\in[\ell]}, (\sigma, M_{i+1}, M_{i+2}), \mu, \mathsf{pk})$, and thus what $\mathcal{A}$ expects to receive.

Finally, if $\mathcal{C}$'s bit $b = 1$ then at all further queries of M to $\mathcal{O}^{RoR}$, $\mathcal{B}$ receives $((R_i)_{i\in[\ell+2]}, \sigma')$ where $R_i \xleftarrow{R} (\mathbb{G}_i^*)^{\ell+2}$ and $\sigma' \leftarrow \mathsf{Sign}_{\mathcal{R}}(R, \mathsf{sk})$, and returns $((R_i)_{i\in[\ell]}, (\sigma', R_{\ell+1}, R_{\ell+2}))$ to $\mathcal{A}$, which is distributed as $R \xleftarrow{R} (\mathbb{G}_i^*)^{\ell}$ and $\mathsf{Sign}'_{\mathcal{R}}((R_i)_{i\in[\ell]}, \mathsf{sk})$, and thus what $\mathcal{A}$ expects to receive. $\mathcal{B}$ thus wins the class-hiding game with the same probability as $\mathcal{A}$ does. □

B Proof of Theorem 2

To prove unforgeability of Scheme 3, assume there is an efficient adversary $\mathcal{A}$ winning the unforgeability game with non-negligible probability $\epsilon(\kappa)$. We then construct an adversary $\mathcal{B}$ that uses $\mathcal{A}$ to either break the EUF-CMA security of the underlying SPS-EQ scheme or to break the binding property of the underlying commitment scheme, that is, break co-DHI$_1^*$.

$\mathcal{B}$ first guesses $\mathcal{A}$'s strategy, i.e., the type of forgery $\mathcal{A}$ will conduct. We call a forgery *Type 1* if for $\mathcal{A}$'s output $(m_i, \tau_i = (\sigma_i, R_i, T_i))_{i\in[k]}$, we have $m_iP + T_i \neq m_jP + T_j$ for all $i \neq j$; otherwise we call it *Type 2*.

Type 1: $\mathcal{B}$ uses $\mathcal{A}$ to break EUF-CMA of the SPS-EQ scheme with $\ell = 2$. $\mathcal{B}$ obtains $\mathsf{pk}_{\mathcal{R}}$ from its challenger $\mathcal{C}$, chooses $q \xleftarrow{R} \mathbb{Z}_q^*$, computes $(Q, \hat{Q}) \leftarrow q(P, \hat{P})$, sets $\mathsf{pk} \leftarrow (\mathsf{pk}_{\mathcal{R}}, Q, \hat{Q})$ and runs $\mathcal{A}(\mathsf{pk})$. Whenever $\mathcal{A}$ queries the $(\cdot, \mathcal{S}_{\mathsf{BS}}(\cdot, \mathsf{sk}))$ oracle with message M, $\mathcal{B}$ queries its SPS-EQ signing oracle $\mathcal{O}(\cdot, \mathsf{sk})$ on M and forwards the reply to $\mathcal{A}$.

If $\mathcal{A}$ outputs $((m_1, \tau_1), \ldots, (m_{k+1}, \tau_{k+1}))$ with $\tau_i = (\sigma_i, R_i, T_i)$ after k successful queries to $(\cdot, \mathcal{S}_{\mathsf{BS}}(\cdot, \mathsf{sk}))$ such that $m_i \neq m_j \;\; \forall i, j \in [k+1], i \neq j$ and $\mathsf{Verify}_{\mathsf{BS}}(m_i, \tau_i, \mathsf{pk}) = 1 \;\forall\; i \in [k+1]$ then $\mathcal{B}$ aborts if for some $i \neq j \in [k+1]$: $m_iP + T_i = m_jP + T_j$ (we have a Type 2 forgery).

Otherwise, we have $(m_iP+T_i, P) \neq (m_jP+T_j, P)$ for all $i, j \in [k+1], i \neq j$. $\mathcal{A}$ has made k signing queries, but $((m_iP + T_i, P), \sigma_i)_{i\in[k+1]}$ are $k+1$ valid SPS-EQ message-signature pairs for *distinct* classes. Consequently, there must exist $i^* \in [k+1]$ such that the message-signature pair $((m_{i^*}P + T_{i^*}, P), \sigma_{i^*})$ represents a class that was not queried to $\mathcal{C}$'s signing oracle. Hence, one of these $k+1$ message-signature pairs enables $\mathcal{B}$ to break the EUF-CMA security of the SPS-EQ scheme. Due to the blindness, however, $\mathcal{B}$ cannot link the pairs to the messages $M_i = (s_i(m_iP + r_iQ), s_iP)$ which $\mathcal{A}$ has queried to the $(\cdot, \mathcal{S}_{\mathsf{BS}}(\cdot, \mathsf{sk}))$

oracle. Therefore $\mathcal{B}$ guesses an index $i^* \in [k+1]$ and outputs $((m_{i^*}P + T_{i^*}, P), \sigma_{i^*})$ as a forgery to $\mathcal{C}$. If $\mathcal{A}$ wins the unforgeability game then $\mathcal{B}$ breaks the EUF-CMA security of the underlying SPS-EQ scheme incurring a polynomial loss of $1/(k+1)$.

Type 2: $\mathcal{B}$ obtains an instance $(\mathsf{BG}, P, \hat{P}, Q = qP, \hat{Q} = q\hat{P})$ of the co-DHI$_1^*$ problem, and its goal is to compute $q^{-1}P$. It computes $(\mathsf{sk}, \mathsf{pk}_{\mathcal{R}}) \leftarrow \mathsf{KeyGen}_{\mathcal{R}}(\mathsf{BG}, \ell = 2)$, and runs $\mathcal{A}$ on $\mathsf{pk} \leftarrow (\mathsf{pk}_{\mathcal{R}}, Q, \hat{Q})$ simulating its $(\cdot, \mathcal{S}_{\mathsf{BS}}(\cdot, \mathsf{sk}))$ oracle as in the real game using sk.

If $\mathcal{A}$ outputs $(m_i, \tau_i = (\sigma_i, R_i, T_i))_{i \in [k+1]}$ after k successful oracle queries such that $m_i \neq m_j$ for all $1 \leq i < j \leq k+1$ and $\mathsf{Verify}_{\mathcal{BS}}(m_i, \tau_i, \mathsf{pk}) = 1$ for all $i \in [k+1]$, then $\mathcal{B}$ aborts if $m_iP + T_i \neq m_jP + T_j$ for all $i, j \in [k+1]$ (we have a Type 1 forgery).

Otherwise, let $i, j \in [k+1]$ be such that $m_iP + T_i = m_jP + T_j$ $(*)$. From the second equation in $\mathsf{Verify}_{\mathsf{BS}}$, since $\hat{Q} = q\hat{P}$, we get $T_i = qR_i$ and $T_j = qR_j$. Together with $(*)$ we have $m_iP + qR_i = m_jP + qR_j$, that is $(m_i - m_j)P = q(R_j - R_i)$, and since $m_i \neq m_j$: $q^{-1}P = (m_i - m_j)^{-1}(R_j - R_i)$. The the latter, which $\mathcal{B}$ can efficiently compute, is thus a solution to the co-DHI$_1^*$ problem. □

C Proof of Theorem 3

In the proof of blindness of our blind signature scheme, we will use the following implication of Def. 11:

Corollary 2. *Let* $\mathsf{SPS\text{-}EQ}$ *be an SPS-EQ scheme on* $(\mathbb{G}_i^*)^\ell$ *satisfying Def. 11. If for a tuple* $(\mathsf{pk}, M, s_0, s_1, \sigma_0, \sigma_1)$ *we have* $\mathsf{Verify}_{\mathcal{R}}(s_0M, \sigma_0, \mathsf{pk}) = 1$ *and* $\mathsf{Verify}_{\mathcal{R}}(s_1M, \sigma_1, \mathsf{pk}) = 1$ *then* $\mathsf{ChgRep}_{\mathcal{R}}(s_0M, \sigma_0, 1/s_0, \mathsf{pk})$ *and* $\mathsf{ChgRep}_{\mathcal{R}}(s_1M, \sigma_1, 1/s_1, \mathsf{pk})$ *are identically distributed.*

Proof. The statement follows, since for $b = 0, 1$ the tuple $(\mathsf{pk}, s_bM, \sigma_b, 1/s_b)$ satisfies (1) (in Def. 11), and for $(M, \sigma_b) \leftarrow \mathsf{ChgRep}_{\mathcal{R}}(s_bM, \sigma_b, 1/s_b, \mathsf{pk})$, by Def. 11 σ_b is random conditioned on $\mathsf{Verify}_{\mathcal{R}}(M, \sigma_b, \mathsf{pk}) = 1$. Thus σ_0 and σ_1 are identically distributed. □

Proof of Theorem 3. Let $\mathbf{Exp}^{\text{blind}}$ be the blindness game (with adversarially/maliciously generated public keys) defined in Def. 15. Consider $\mathbf{Exp}^{\text{blind}}_{\mathcal{A},\mathsf{BS}}$ with BS being Scheme 3 and any PPT adversary $\mathcal{A}$, which we assume w.l.o.g. makes both calls to its $(\mathcal{U}_{\mathsf{BS}}(m_b, \mathsf{pk}), \cdot)$ oracle. Written out, we have:

```
$\mathbf{Exp}^{\text{blind}}_{\mathcal{A},\mathsf{BS}}$:
  $b \overset{R}{\leftarrow} \{0,1\}$
  $((\mathsf{pk}_{\mathcal{R}}, Q, \hat{Q}), m_0, m_1, \mathsf{st}_{\mathcal{A}}) \leftarrow \mathcal{A}(1^\kappa)$
  $\mathsf{BG} \leftarrow \mathsf{BGGen}_{\mathcal{R}}(1^\kappa)$
  If $Q = 0_{\mathbb{G}_1}$ or $e(Q, \hat{P}) \neq e(P, \hat{Q})$ then $M_0, M_1 \leftarrow \perp$
  Else
    $r_0, s_0 \overset{R}{\leftarrow} \mathbb{Z}_p$ ; $r_1, s_1 \overset{R}{\leftarrow} \mathbb{Z}_p$
    $M_0 \leftarrow (s_0(m_0P + r_0Q), s_0P)$ ; $M_1 \leftarrow (s_1(m_1P + r_1Q), s_1P)$
  $(\pi_b, \mathsf{st}_{\mathcal{A}}) \leftarrow \mathcal{A}(\mathsf{st}_{\mathcal{A}}, M_b)$ ; $(\pi_{1-b}, \mathsf{st}_{\mathcal{A}}) \leftarrow \mathcal{A}(\mathsf{st}_{\mathcal{A}}, M_{1-b})$
  If $(M_0, M_1) = (\perp, \perp)$ or $\mathsf{Verify}_{\mathcal{R}}(M_0, \pi_0, \mathsf{pk}) = 0$ or $\mathsf{Verify}_{\mathcal{R}}(M_1, \pi_1, \mathsf{pk}) = 0$
    then $b^* \leftarrow \mathcal{A}(\mathsf{st}_{\mathcal{A}}, \perp, \perp)$
  Else
    $(N_0, \sigma_0) \leftarrow \mathsf{ChgRep}_{\mathcal{R}}(M_0, \pi_0, 1/s_0, \mathsf{pk})$ ; $(N_1, \sigma_1) \leftarrow \mathsf{ChgRep}_{\mathcal{R}}(M_1, \pi_1, 1/s_1, \mathsf{pk})$
    $b^* \leftarrow \mathcal{A}(\mathsf{st}_{\mathcal{A}}, (\sigma_0, r_0P, r_0Q), (\sigma_1, r_1P, r_1Q))$
  Return $(b^* = b)$
```

We have slightly modified the game, in that (for $i = 0, 1$) we allowed s_i to also take the value 0 and r_i to be such that $m_i P + r_i Q = 0_{\mathbb{G}_1}$. However, these events only happen with negligible probability.

We first argue that if $\mathcal{A}$ outputs an inconsistent public key or if π_0 or π_1 do not pass $\mathsf{Verify}_{\mathcal{R}}$ then the bit b is information-theoretically hidden from $\mathcal{A}$. This is because if one of the above is the case then in the second phase $\mathcal{A}$ receives $(\perp, \perp)$, and r_0, r_1 information-theoretically hide m_0, m_1, and thus the bit b is also information-theoretically hidden, meaning $\Pr[\mathbf{Exp}^{\text{blind}}_{\mathcal{A},\mathsf{BS}} = 1] = 1/2$.

We can now assume w.l.o.g. that $\mathcal{A}$ outputs a valid pk and π_0 and π_1 verify: If $\mathcal{A}$ was not like this, we could construct a well-behaving adversary $\mathcal{A}'$ from $\mathcal{A}$: $\mathcal{A}'$ simulates $\mathcal{A}$ and whenever $\mathcal{A}$ misbehaves (which $\mathcal{A}'$ can efficiently detect), it aborts the simulation and outputs a random bit. By the above, $\mathcal{A}'$ wins with the same probability as $\mathcal{A}$. With this assumption on $\mathcal{A}$ the experiment simplifies thus to:

```
$\mathbf{Exp}^{\text{blind-non-}\perp}_{\mathcal{A},\mathsf{BS}}$:
    $((\mathsf{pk}_{\mathcal{R}}, Q, \hat{Q}), m_0, m_1, \mathsf{st}_{\mathcal{A}}) \leftarrow \mathcal{A}(1^\kappa)$
    $\mathsf{BG} \leftarrow \mathsf{BGGen}_{\mathcal{R}}(1^\kappa)$
    $r_0, r_1 \overset{R}{\leftarrow} \mathbb{Z}_p$        $(*)$
    $s_0, s_1 \overset{R}{\leftarrow} \mathbb{Z}_p$ ; $b \overset{R}{\leftarrow} \{0,1\}$
    $M_0 \leftarrow (s_0(m_0 P + r_0 Q), s_0 P)$ ; $M_1 \leftarrow (s_1(m_1 P + r_1 Q), s_1 P)$
    $(\pi_b, \mathsf{st}_{\mathcal{A}}) \leftarrow \mathcal{A}(\mathsf{st}_{\mathcal{A}}, M_b)$ ; $(\pi_{1-b}, \mathsf{st}_{\mathcal{A}}) \leftarrow \mathcal{A}(\mathsf{st}_{\mathcal{A}}, M_{1-b})$
    $(N_0, \sigma_0) \leftarrow \mathsf{ChgRep}_{\mathcal{R}}(M_0, \pi_0, 1/s_0, \mathsf{pk})$ ; $(N_1, \sigma_1) \leftarrow \mathsf{ChgRep}_{\mathcal{R}}(M_1, \pi_1, 1/s_1, \mathsf{pk})$
    $b^* \leftarrow \mathcal{A}(\mathsf{st}_{\mathcal{A}}, (\sigma_0, r_0 P, r_0 Q), (\sigma_1, r_1 P, r_1 Q))$
    Return $(b^* = b)$
```

Execution 1. Now we do the following: We run $\mathbf{Exp}^{\text{blind-non-}\perp}$ with $\mathcal{A}$, in particular choosing $r_0, r_1, s_0^{(1)}, s_1^{(1)}$ and $b^{(1)}$, constructing

$$M_0^{(1)} \leftarrow \big(s_0^{(1)}(m_0 P + r_0 Q),\, s_0^{(1)} P\big) \ , \qquad M_1^{(1)} \leftarrow \big(s_1^{(1)}(m_1 P + r_1 Q),\, s_1^{(1)} P\big)$$

and running $\mathcal{A}$ on $M^{(1)}_{b^{(1)}}$ and then on $M^{(1)}_{1-b^{(1)}}$ to obtain signatures $\pi_0^{(1)}, \pi_1^{(1)}$. Then we *rewind* the experiment to the point $(*)$ and run it again. We choose independent uniform random $s_0^{(2)}, s_1^{(2)} \overset{R}{\leftarrow} \mathbb{Z}_p$, $b^{(2)} \overset{R}{\leftarrow} \{0,1\}$ (but use the same r_0, r_1 as in the first run), set

$$M_0^{(2)} \leftarrow \big(s_0^{(2)}(m_0 P + r_0 Q),\, s_0^{(2)} P\big) \ , \qquad M_1^{(2)} \leftarrow \big(s_1^{(2)}(m_1 P + r_1 Q),\, s_1^{(2)} P\big) \ ,$$

run $\mathcal{A}$ on $M^{(2)}_{b^{(2)}}$ and then on $M^{(2)}_{1-b^{(2)}}$ to obtain signatures $\pi_0^{(2)}, \pi_1^{(2)}$, and finish the experiment: For $i = 0, 1$ we compute

$$(N_i^{(2)}, \sigma_i^{(2)}) \leftarrow \mathsf{ChgRep}_{\mathcal{R}}\big(M_i^{(2)}, \pi_i^{(2)}, 1/s_i^{(2)}, \mathsf{pk}\big) \ , \qquad \text{run } \ b^* \leftarrow \mathcal{A}(\mathsf{st}_{\mathcal{A}}, (\sigma_0^{(2)}, r_0 P, r_0 Q), (\sigma_1^{(2)}, r_1 P, r_1 Q)) \ , \tag{9}$$

and return $(b^* = b^{(2)})$. Since the second run simply constitutes an independent run of $\mathcal{A}$ we have that the probability of returning 1 is precisely $\Pr[\mathbf{Exp}^{\text{blind-non-}\perp}_{\mathcal{A},\mathsf{BS}} = 1] = \Pr[\mathbf{Exp}^{\text{blind}}_{\mathcal{A},\mathsf{BS}} = 1]$ (by our assumption on $\mathcal{A}$).

Execution 2. We now introduce a modification. We proceed as in Execution 1, but instead of (9), we compute for $i = 0, 1$:

$$
\begin{aligned}
(N_i^{(1)}, \sigma_i^{(1)}) \leftarrow \mathsf{ChgRep}_{\mathcal{R}}\big(M_i^{(1)}, \pi_i^{(1)}, 1/s_i^{(1)}, \mathsf{pk}\big) \ , & \\
\text{run} \ \ b^* \leftarrow \mathcal{A}(\mathsf{st}_{\mathcal{A}}, (\sigma_0^{(1)}, r_0P, r_0Q), (\sigma_1^{(1)}, r_1P, r_1Q)) \ , & \qquad (10)
\end{aligned}
$$

and return $(b^* = b^{(2)})$. That is, we use the signatures $\pi_0^{(1)}, \pi_1^{(1)}$ from the *first* run, adapt them to signatures on $N_i^{(1)} = (m_iP + r_iQ, P) = N_i^{(2)}$ and give them to $\mathcal{A}$ as part of our blind signatures. We now argue that the winning probability of the adversary does not change. For $i = 0, 1$ we have the following. Since by assumption we have

$$
\begin{aligned}
\mathsf{Verify}_{\mathcal{R}}\big(s_i^{(1)} \cdot (m_iP + r_iQ, P), \pi_i^{(1)}, \mathsf{pk}\big) = 1 \quad \text{and} \\
\mathsf{Verify}_{\mathcal{R}}\big(s_i^{(2)} \cdot (m_iP + r_iQ, P), \pi_i^{(2)}, \mathsf{pk}\big) = 1 \ \ ,
\end{aligned}
$$

the tuple $\big(\mathsf{pk}, (m_iP + r_iQ, P), s_i^{(1)}, s_i^{(2)}, \pi_i^{(1)}, \pi_i^{(2)}\big)$ satisfies the premise of Corollary 2 and therefore the outputs $\sigma_i^{(1)}$ and $\sigma_i^{(2)}$ of $\mathsf{ChgRep}_{\mathcal{R}}(M_i^{(1)}, \pi_i^{(1)}, 1/s_i^{(1)}, \mathsf{pk})$ and $\mathsf{ChgRep}_{\mathcal{R}}(M_i^{(2)}, \pi_i^{(2)}, 1/s_i^{(2)}, \mathsf{pk})$, resp., are identically distributed. Therefore Executions 1 and 2 are identically distributed and the probability that after Execution 2 we have $(b^* = b^{(2)})$ is $\Pr[\mathbf{Exp}^{\mathsf{blind}}_{\mathcal{A},\mathsf{BS}} = 1]$.

Let us write down Execution 2:

$$
\begin{array}{l}
((\mathsf{pk}_{\mathcal{R}}, Q, \hat{Q}), m_0, m_1, \mathsf{st}_{\mathcal{A}}) \leftarrow \mathcal{A}(1^\kappa) \\
\mathsf{BG} \leftarrow \mathsf{BGGen}_{\mathcal{R}}(1^\kappa) \\
r_0, r_1 \xleftarrow{R} \mathbb{Z}_p \qquad (*) \\
\begin{array}{l|ll}
s_0^{(1)}, s_1^{(1)} \xleftarrow{R} \mathbb{Z}_p \ ; \ b^{(1)} \xleftarrow{R} \{0,1\} & s_0^{(2)}, s_1^{(2)} \xleftarrow{R} \mathbb{Z}_p \ ; \ b^{(2)} \xleftarrow{R} \{0,1\} & \\
M_0^{(1)} \leftarrow \big(s_0^{(1)}(m_0P + r_0Q),\ s_0^{(1)}P\big) & M_0^{(2)} \leftarrow \big(s_0^{(2)}(m_0P + r_0Q),\ s_0^{(2)}P\big) & \| \\
M_1^{(1)} \leftarrow \big(s_1^{(1)}(m_1P + r_1Q),\ s_1^{(1)}P\big) & M_1^{(2)} \leftarrow \big(s_1^{(2)}(m_1P + r_1Q),\ s_1^{(2)}P\big) & \| \\
(\pi^{(1)}_{b^{(1)}}, \mathsf{st}'_{\mathcal{A}}) \leftarrow \mathcal{A}(\mathsf{st}_{\mathcal{A}}, M^{(1)}_{b^{(1)}}) & (\pi^{(2)}_{b^{(2)}}, \mathsf{st}_{\mathcal{A}}) \leftarrow \mathcal{A}(\mathsf{st}_{\mathcal{A}}, M^{(2)}_{b^{(2)}}) & \\
(\pi^{(1)}_{1-b^{(1)}}, \mathsf{st}'_{\mathcal{A}}) \leftarrow \mathcal{A}(\mathsf{st}_{\mathcal{A}}, M^{(1)}_{1-b^{(1)}}) & (\pi^{(2)}_{1-b^{(2)}}, \mathsf{st}_{\mathcal{A}}) \leftarrow \mathcal{A}(\mathsf{st}_{\mathcal{A}}, M^{(2)}_{1-b^{(2)}}) &
\end{array} \\
(N_0, \sigma_0) \leftarrow \mathsf{ChgRep}_{\mathcal{R}}(M_0^{(1)}, \pi_0^{(1)}, 1/s_0^{(1)}, \mathsf{pk}) \\
(N_1, \sigma_1) \leftarrow \mathsf{ChgRep}_{\mathcal{R}}(M_1^{(1)}, \pi_1^{(1)}, 1/s_1^{(1)}, \mathsf{pk}) \\
b^* \leftarrow \mathcal{A}(\mathsf{st}_{\mathcal{A}}, (\sigma_0, r_0P, r_0Q), (\sigma_1, r_1P, r_1Q)) \\
\text{Return } (b^* = b^{(2)})
\end{array}
$$

Execution 3. We define another variant, where in Execution 2 we replace the two lines marked with $\|$ by

$$
\begin{array}{rl}
t_0 \xleftarrow{R} \mathbb{Z}_p \,;\, M_0^{(2)} \leftarrow \big(s_0^{(2)} m_0P + t_0Q,\ s_0^{(2)}P\big) & \| \\
M_1^{(2)} \leftarrow \big(s_1^{(2)} m_1P + s_1^{(2)} r_1Q,\ s_1^{(2)}P\big) & \|
\end{array}
$$

that is, in the definition of $M_0^{(2)}$ we replaced the value $s_0^{(2)}r_0$ with a random element t_0.

Execution 4. Our final execution also replaces $s_1^{(2)}r_1$ in the definition of $M_1^{(2)}$ with a random element t_1. That is, it is defined as Execution 2 above, except with the lines marked with $\|$ replaced by the following:

$$t_0 \xleftarrow{R} \mathbb{Z}_p\,;\; M_0^{(2)} \leftarrow \big(s_0^{(2)} m_0 P + t_0 Q,\; s_0^{(2)} P\big) \quad \Big\|$$
$$t_1 \xleftarrow{R} \mathbb{Z}_p\,;\; M_1^{(2)} \leftarrow \big(s_1^{(2)} m_1 P + t_1 Q,\; s_1^{(2)} P\big) \quad \Big\|$$

Claim 2. *If Assumption 1 (Def. 16) holds then Executions 2 and 3 are indistinguishable; likewise, Executions 3 and 4 are indistinguishable.*

Proof. Assume that there exists an adversary $\mathcal{A}$, for whom the probability that $(b^* = b^{(2)})$ is noticeably different in Executions 2 and 3. Then we construct an adversary $\mathcal{B}$ against the Assumption 1 as follows:

On input 1^κ, $\mathcal{B}$ runs $((\mathsf{pk}_{\mathcal{R}}, Q, \hat{Q}), m_0, m_1, \mathsf{st}_{\mathcal{A}}) \leftarrow \mathcal{A}(1^\kappa)$ and outputs

$$(\mathsf{st}_{\mathcal{B}} \leftarrow (\mathsf{pk}_{\mathcal{R}}, m_0, m_1, \mathsf{st}_{\mathcal{A}}), Q, \hat{Q}) \enspace ;$$

$\mathcal{B}$ then receives a challenge (rP, rQ, sP, tQ) and needs to decide whether $t = rs$.
$\mathcal{B}$ simulates Execution 2 with $\mathcal{A}$, except that it implicitly sets $r_0 \leftarrow r$ and $s_0^{(2)} \leftarrow s$ as well as $s_0^{(2)} r_0 \leftarrow t$ from the assumption. $\mathcal{B}$'s output is $(b^* = b^{(2)})$. If $t = rs$ then $\mathcal{B}$ simulated Execution 2, whereas if t is uniformly random, it simulated Execution 3. In particular, after receiving the challenge, $\mathcal{B}$ runs as follows (which shows that the simulation can be done using the challenge, which we underline):

```
$\underline{\mathcal{B}\big(\mathsf{st}_{\mathcal{B}} = (\mathsf{pk}_{\mathcal{R}}, m_0, m_1, \mathsf{st}_{\mathcal{A}}), rP, rQ, sP, tQ\big):}$
    $r_1 \xleftarrow{R} \mathbb{Z}_p \qquad (*)$
    $s_0^{(1)}, s_1^{(1)} \xleftarrow{R} \mathbb{Z}_p\ ;\ b^{(1)} \xleftarrow{R} \{0,1\}$
    $M_0^{(1)} \leftarrow \big(s_0^{(1)}(m_0 P + \underline{(rQ)}),\ s_0^{(1)} P\big)\ ;\ M_1^{(1)} \leftarrow \big(s_1^{(1)}(m_1 P + r_1 Q),\ s_1^{(1)} P\big)$
    $(\pi_{b^{(1)}}^{(1)}, \mathsf{st}'_{\mathcal{A}}) \leftarrow \mathcal{A}(\mathsf{st}_{\mathcal{A}}, M_{b^{(1)}}^{(1)})\ ;\ (\pi_{1-b^{(1)}}^{(1)}, \mathsf{st}'_{\mathcal{A}}) \leftarrow \mathcal{A}(\mathsf{st}_{\mathcal{A}}, M_{1-b^{(1)}}^{(1)})$
    REWIND to $(*)$
        $s_1^{(2)} \xleftarrow{R} \mathbb{Z}_p\ ;\ b^{(2)} \xleftarrow{R} \{0,1\}$
        $M_0^{(2)} \leftarrow \big(m_0 \underline{(sP)} + \underline{(tQ)},\ \underline{(sP)}\big)\ ;\ M_1^{(2)} \leftarrow \big(s_1^{(2)} m_1 P + s_1^{(2)} r_1 Q,\ s_1^{(2)} P\big)$
        $(\pi_{b^{(2)}}^{(2)}, \mathsf{st}_{\mathcal{A}}) \leftarrow \mathcal{A}(\mathsf{st}_{\mathcal{A}}, M_{b^{(2)}}^{(2)})\ ;\ (\pi_{1-b^{(2)}}^{(2)}, \mathsf{st}_{\mathcal{A}}) \leftarrow \mathcal{A}(\mathsf{st}_{\mathcal{A}}, M_{1-b^{(2)}}^{(2)})$
    $(N_0, \sigma_0) \leftarrow \mathsf{ChgRep}_{\mathcal{R}}(M_0^{(1)}, \pi_0^{(1)}, 1/s_0^{(1)}, \mathsf{pk})\ ;\ (N_1, \sigma_1) \leftarrow \mathsf{ChgRep}_{\mathcal{R}}(M_1^{(1)}, \pi_1^{(1)}, 1/s_1^{(1)}, k)$
    $b^* \leftarrow \mathcal{A}(\mathsf{st}_{\mathcal{A}}, (\sigma_0, \underline{(rP)}, \underline{(rQ)}), (\sigma_1, r_1 P, r_1 Q))$
    Return $(b^* = b^{(2)})$
```

We have thus that the probability that $\mathcal{B}$ outputs 1 when given a DDH instance is the probability that Execution 2 outputs 1; and the probability that $\mathcal{B}$ outputs 1 when given a random instance is the probability that Execution 3 outputs 1. Thus, if $\mathcal{A}$ behaved differently in Executions 2 and 3 then $\mathcal{B}$ would break the assumption.

Analogously we can construct an adversary $\mathcal{B}$ which breaks the assumption given an adversary $\mathcal{A}$ that distinguishes Executions 3 and 4. □

Finally, let us consider Execution 4 in Fig. 1 below.
We now see that for $i = 0, 1$, since $s_i^{(2)}$ and t_i are uniformly random and used nowhere other than in the definition of $M_i^{(2)}$, the latter is a uniform random element from $\mathbb{G}_1 \times \mathbb{G}_1$. Since $b^{(2)}$ is only used to determine the order in which $M_0^{(2)}$ and $M_1^{(2)}$ (which are both random elements) are sent to $\mathcal{A}$, the bit $b^{(2)}$ is information-theoretically hidden. We thus have that the probability that $(b^* = b^{(2)})$ in Execution 4 is exactly 1/2.

Overall, we have that $\Pr[\mathbf{Exp}_{\mathcal{A},\mathsf{BS}}^{\mathrm{blind}} = 1]$ can only be negligibly different from 1/2, which proves blindness. □

```
$((\mathsf{pk}_{\mathcal{R}}, Q, \hat{Q}), m_0, m_1, \mathsf{st}_{\mathcal{A}}) \leftarrow \mathcal{A}(1^\kappa)$
$\mathsf{BG} \leftarrow \mathsf{BGGen}_{\mathcal{R}}(1^\kappa)$
$r_0, r_1 \xleftarrow{R} \mathbb{Z}_p$
$s_0^{(1)}, s_1^{(1)} \xleftarrow{R} \mathbb{Z}_p \; ; \; b^{(1)} \xleftarrow{R} \{0,1\}$
$M_0^{(1)} \leftarrow \big(s_0^{(1)}(m_0P + r_0Q),\, s_0^{(1)}P\big)$
$M_1^{(1)} \leftarrow \big(s_1^{(1)}(m_1P + r_1Q),\, s_1^{(1)}P\big)$
$(\pi_{b^{(1)}}^{(1)}, \mathsf{st}'_{\mathcal{A}}) \leftarrow \mathcal{A}(\mathsf{st}_{\mathcal{A}}, M_{b^{(1)}}^{(1)})$
$(\pi_{1-b^{(1)}}^{(1)}, \mathsf{st}'_{\mathcal{A}}) \leftarrow \mathcal{A}(\mathsf{st}_{\mathcal{A}}, M_{1-b^{(1)}}^{(1)})$
    | $s_0^{(2)}, s_1^{(2)}, t_0, t_1 \xleftarrow{R} \mathbb{Z}_p \; ; \; b^{(2)} \xleftarrow{R} \{0,1\}$
    | $M_0^{(2)} \leftarrow \big(s_0^{(2)} m_0P + t_0Q,\, s_0^{(2)}P\big)$
    | $M_1^{(2)} \leftarrow \big(s_1^{(2)} m_1P + t_1Q,\, s_1^{(2)}P\big)$
    | $(\pi_{b^{(2)}}^{(2)}, \mathsf{st}_{\mathcal{A}}) \leftarrow \mathcal{A}(\mathsf{st}_{\mathcal{A}}, M_{b^{(2)}}^{(2)})$
    | $(\pi_{1-b^{(2)}}^{(2)}, \mathsf{st}_{\mathcal{A}}) \leftarrow \mathcal{A}(\mathsf{st}_{\mathcal{A}}, M_{1-b^{(2)}}^{(2)})$
$(N_0, \sigma_0) \leftarrow \mathsf{ChgRep}_{\mathcal{R}}(M_0^{(1)}, \pi_0^{(1)}, 1/s_0^{(1)}, \mathsf{pk})$
$(N_1, \sigma_1) \leftarrow \mathsf{ChgRep}_{\mathcal{R}}(M_1^{(1)}, \pi_1^{(1)}, 1/s_1^{(1)}, \mathsf{pk})$
$b^* \leftarrow \mathcal{A}(\mathsf{st}_{\mathcal{A}}, (\sigma_0, r_0P, r_0Q), (\sigma_1, r_1P, r_1Q))$
Return $(b^* = b^{(2)})$
```

Figure 1: Final experiment in the proof

D Partially Blind Signatures

For the sake of completeness, we state the abstract model and the security properties of partially blind signature schemes.

Definition 18 (Partially blind signature scheme)**.** A *partially blind signature scheme* PBS consists of the following PPT algorithms:

$\mathsf{KeyGen}_{\mathsf{PBS}}(1^\kappa)$, on input κ, returns a key pair $(\mathsf{sk}, \mathsf{pk})$. The security parameter κ is also an (implicit) input to the following algorithms.

$(\mathcal{U}_{\mathsf{PBS}}(m, \gamma, \mathsf{sk}), \mathcal{S}_{\mathsf{PBS}}(\gamma, \mathsf{sk}))$ are run by a user and a signer, who interact during execution. $\mathcal{U}_{\mathsf{PBS}}$ gets input a message m, common information γ and a public key pk. $\mathcal{S}_{\mathsf{PBS}}$ gets input common information γ and a secret key sk. At the end $\mathcal{U}_{\mathsf{PBS}}$ outputs σ, a signature on (m, γ), or $\perp$ if the interaction was not successful.

$\mathsf{Verify}_{\mathsf{PBS}}(m, \gamma, \sigma, \mathsf{pk})$ is deterministic and given a message-signature tuple (m, γ, σ) and a public key pk outputs 1 if σ is valid on (m, γ) under pk and 0 otherwise. $\diamondsuit$

A partially blind signature scheme PBS must satisfy *correctness*, *unforgeability* and *partial blindness*. These properties can be seen as a generalization of blind signatures (where the common information is the empty string) and are defined as follows [Oka06].

Definition 19 (Correctness)**.** A partially blind signature scheme PBS is *correct* if for all $\kappa \in \mathbb{N}$, all key pairs $(\mathsf{sk}, \mathsf{pk}) \leftarrow \mathsf{KeyGen}_{\mathsf{PBS}}(1^\kappa)$, all messages m and common information γ, and all $\sigma \leftarrow (\mathcal{U}_{\mathsf{PBS}}(m, \gamma, \mathsf{pk}), \mathcal{S}_{\mathsf{PBS}}(\gamma, \mathsf{sk}))$ it holds that $\mathsf{Verify}_{\mathsf{PBS}}(m, \sigma, \mathsf{pk}) = 1$. $\diamondsuit$

Definition 20 (Unforgeability)**.** PBS is *unforgeable*, if for all PPT algorithms $\mathcal{A}$ having access to a signer oracle, there is a negligible function $\epsilon(\cdot)$ such that:

$$\Pr\left[\begin{array}{lcr} (\mathsf{sk}, \mathsf{pk}) \leftarrow \mathsf{KeyGen}_{\mathsf{PBS}}(1^\kappa), & & m_i^* \neq m_j^* \;\forall i, j \in [k_{\gamma^*} + 1], i \neq j \\ (\gamma^*, (m_i^*, \sigma_i^*)_{i \in [k_{\gamma^*}+1]}) & : & \wedge\;\; \mathsf{Verify}_{\mathsf{PBS}}(m_i^*, \gamma^*, \sigma_i^*, \mathsf{pk}) = 1 \\ \qquad\qquad \leftarrow \mathcal{A}^{(\cdot, \mathcal{S}_{\mathsf{PBS}}(\mathsf{sk}))}(\mathsf{pk}) & & \forall i \in [k_{\gamma^*} + 1] \end{array}\right] \leq \epsilon(\kappa) \;,$$

where k_{γ^*} is the number of completed interactions with the oracle involving γ^*. $\diamondsuit$

$\mathsf{KeyGen}_{\mathsf{BSV}}(1^\kappa, n)$: On input security parameter κ and vector length n, compute $\mathsf{BG} \leftarrow \mathsf{BGGen}_{\mathcal{R}}(1^\kappa)$, $(\mathsf{sk}, \mathsf{pk}_{\mathcal{R}}) \overset{R}{\leftarrow} \mathsf{KeyGen}_{\mathcal{R}}(\mathsf{BG}, \ell = 2)$, pick $q \overset{R}{\leftarrow} \mathbb{Z}_p^*$ and $(p_i)_{i\in[n]} \overset{R}{\leftarrow} (\mathbb{Z}_p^*)^n$, set $Q \leftarrow qP$, $\hat{Q} \leftarrow q\hat{P}$ and $(P_i)_{i\in[n]} \leftarrow (p_iP)_{i\in[n]}$ and output $(\mathsf{sk}, \mathsf{pk} = (\mathsf{pk}_{\mathcal{R}}, (P_i)_{i\in[n]}, Q, \hat{Q}))$.

$\mathcal{U}^{(1)}_{\mathsf{BSV}}(\mathsf{pk}, \boldsymbol{m})$: Given $\mathsf{pk} = (\mathsf{pk}_{\mathcal{R}}, (P_i)_{i\in[n]}, Q, \hat{Q})$ and $\boldsymbol{m} \in \mathbb{Z}_p^n$, compute $\mathsf{BG} \leftarrow \mathsf{BGGen}_{\mathcal{R}}(1^\kappa)$. If $\mathsf{Check}_{\mathsf{P}}((P_i)_{i\in[n]}, Q, \hat{Q}) = 0$ then return $\perp$; else choose $s \overset{R}{\leftarrow} \mathbb{Z}_p^*$ and $r \overset{R}{\leftarrow} \mathbb{Z}_p$ such that $\sum_{i\in[n]} m_iP_i + rQ \neq 0_{\mathbb{G}_1}$ and output

$$M \leftarrow \big(s(\textstyle\sum_{i\in[n]} m_iP_i + rQ), sP\big) \qquad \mathsf{st} \leftarrow (\mathsf{BG}, \mathsf{pk}_{\mathcal{R}}, Q, M, r, s)$$

$\mathcal{S}_{\mathsf{BSV}}(M, \mathsf{sk})$: Given $M \in (\mathbb{G}_1^*)^2$ and a secret key sk, output $\pi \leftarrow \mathsf{Sign}_{\mathcal{R}}(M, \mathsf{sk})$.

$\mathcal{U}^{(2)}_{\mathsf{BSV}}(\mathsf{st}, \pi)$: Parse st as $(\mathsf{BG}, \mathsf{pk}_{\mathcal{R}}, Q, M, r, s)$. If $\mathsf{Verify}_{\mathcal{R}}(M, \pi, \mathsf{pk}_{\mathcal{R}}) = 0$, return $\perp$. Else run $((\sum_{i\in[n]} m_iP_i + rQ, P), \sigma) \leftarrow \mathsf{ChgRep}_{\mathcal{R}}(M, \pi, \frac{1}{s}, \mathsf{pk}_{\mathcal{R}})$ and output $\tau \leftarrow (\sigma, rP, rQ)$.

$\mathsf{Verify}_{\mathsf{BSV}}(\boldsymbol{m}, \tau, \mathsf{pk})$: Given $\boldsymbol{m} \in \mathbb{Z}_p^n$, a blind signature $\tau = (\sigma, R, T)$ and $\mathsf{pk} = (\mathsf{pk}_{\mathcal{R}}, (P_i)_{i\in[n]}, Q, \hat{Q})$ with $\mathsf{Check}_{\mathsf{P}}((P_i)_{i\in[n]}, Q, \hat{Q}) = 1$, output 1 if the following holds and 0 otherwise.

$$\mathsf{Verify}_{\mathcal{R}}\big((\textstyle\sum_{i\in[n]} m_iP_i + T, P), \sigma, \mathsf{pk}_{\mathcal{R}}\big) = 1 \qquad e(T, \hat{P}) = e(R, \hat{Q})$$

Scheme 4: Blind signature scheme on message vectors

Definition 21 (Partial blindness)**.** PBS is *partially blind*, if for all PPT algorithms $\mathcal{A}$ with one-time access to two user oracles, there is a negligible function $\epsilon(\cdot)$ such that:

$$\Pr\left[\begin{array}{l} b \overset{R}{\leftarrow} \{0,1\},\ (\mathsf{pk}, m_0, m_1, \gamma, \mathsf{st}) \leftarrow \mathcal{A}(1^\kappa), \\ \mathsf{st} \leftarrow \mathcal{A}^{(\mathcal{U}_{\mathsf{PBS}}(m_b,\gamma,\mathsf{pk}),\cdot)^{(1)},(\mathcal{U}_{\mathsf{PBS}}(m_{1-b},\gamma,\mathsf{pk}),\cdot)^{(1)}}(\mathsf{st}), \\ \text{Let } \sigma_b \text{ and } \sigma_{1-b} \text{ be the resp. outputs of } \mathcal{U}_{\mathsf{PBS}}, \\ \text{If } \sigma_0 = \perp \text{ or } \sigma_1 = \perp \text{ then } (\sigma_0, \sigma_1) \leftarrow (\perp, \perp), \\ b^* \leftarrow \mathcal{A}(\mathsf{st}, \sigma_0, \sigma_1) \end{array} \ :\ b^* = b \right] - \frac{1}{2} \leq \epsilon(\kappa)\ .$$

◇

E Blind Signatures on Message Vectors

Scheme 4 presents our construction for blind signatures on message vectors. We use $\{1, 0\} \leftarrow \mathsf{Check}_{\mathsf{P}}(\mathsf{cpp})$ to denote the check for valid commitment parameters: For a generalized Pedersen commitment in $\mathbb{G}_1$ of a Type-3 bilinear group BG with tweaked opening we have $\mathsf{cpp} = ((P_i)_{i\in[n]}, Q, \hat{Q})$ and the check holds if

- $\mathsf{cpp} \in (\mathbb{G}_1^*)^{n+1} \times \mathbb{G}_2$ and the $\mathbb{G}_1^*$ elements are pairwise distinct;
- $e(Q, \hat{P}) = e(P, \hat{Q})$.

F Proof of Proposition 3

Let $\mathcal{A}$ be a generic PPT adversary and let $\sigma\colon \mathbb{G}_1 \to \{0,1\}^{m_1}$, $\hat{\sigma}\colon \mathbb{G}_2 \to \{0,1\}^{m_2}$ and $\tau\colon \mathbb{G}_T \to \{0,1\}^{m_T}$ be random, homomorphic encoding functions where w.l.o.g. $m_1 < m_2 < m_T$. $\mathcal{A}$ cannot work directly with group elements, but is forced to work with their image under $\sigma, \hat{\sigma}$ and τ. Furthermore, $\mathcal{A}$ is given oracle access to perform generic bilinear group operations (operations in $\mathbb{G}_1$, $\mathbb{G}_2$ and $\mathbb{G}_T$ and pairings). Since $\mathcal{A}$ is given access to the group-element encodings, it can perform equality checks on its own through string equality tests. At last, we require that $\mathcal{A}$ can only submit already queried encodings to the group oracles. (Note that we can enforce this by choosing m_1, m_2 and m_T large enough making the probability of guessing bitstrings in the image of $\sigma, \hat{\sigma}$ and τ, respectively, negligible.)

Now, let $\mathcal{B}$ be an algorithm interacting with $\mathcal{A}$ as follows. $\mathcal{B}$ picks a random bit $b \xleftarrow{R} \{0,1\}$, picks $\sigma_P \xleftarrow{R} \{0,1\}^{m_1}$ and $\hat{\sigma}_{\hat{P}} \xleftarrow{R} \{0,1\}^{m_2}$ as encoding of the generators of $\mathbb{G}_1$ and $\mathbb{G}_2$, respectively. $\mathcal{B}$ stores $(1, \sigma_P)$ in a list L_1 and $(1, \hat{\sigma}_{\hat{P}})$ in a list L_2 and gives the respective encodings to $\mathcal{A}$. Furthermore, it initializes a list L_T to manage elements of $\mathbb{G}_T$. At first, $\mathcal{B}$ simulates the group oracles as follows.

Group action in $\mathbb{G}_1$: Given two bitstrings σ_0, σ_1 representing elements in $\mathbb{G}_1$, $\mathcal{B}$ looks them up in L_1 and recovers the first components $f_0, f_1 \in \mathbb{Z}_p$ of the corresponding entries (f_i, σ_i). It computes $f_0 + f_1$ and if L_1 already contains an entry starting with $f_0 + f_1$, $\mathcal{B}$ returns its associated bitstring σ; otherwise, $\mathcal{B}$ chooses $\sigma \xleftarrow{R} \{0,1\}^{m_1}$, returns σ and stores $(f_0 + f_1, \sigma)$ in L_1.

Inversion in $\mathbb{G}_1$: Given a bitstring σ representing an element in $\mathbb{G}_1$, $\mathcal{B}$ recovers the corresponding values $f \in \mathbb{Z}_p$ and computes $-f$. In case L_1 already contains $-f$, $\mathcal{B}$ returns its associated bitstring σ'. Otherwise, $\mathcal{B}$ chooses $\sigma' \xleftarrow{R} \{0,1\}^{m_1}$, returns σ' and stores $(-f, \sigma')$ in L_1.

Group action in $\mathbb{G}_2$: Given two bitstrings $\hat{\sigma}_0, \hat{\sigma}_1$ representing elements in $\mathbb{G}_2$, $\mathcal{B}$ recovers the corresponding values $\hat{f}_0, \hat{f}_1 \in \mathbb{Z}_p$ and computes $\hat{f}_0 + \hat{f}_1$. In case L_2 already contains $\hat{f}_0 + \hat{f}_1$, $\mathcal{B}$ returns its associated bitstring $\hat{\sigma}$. Otherwise, $\mathcal{B}$ chooses $\hat{\sigma} \xleftarrow{R} \{0,1\}^{m_2}$, returns $\hat{\sigma}$ and stores $(\hat{f}_0 + \hat{f}_1, \hat{\sigma})$ in L_2.

Inversion in $\mathbb{G}_2$: Given a bitstring $\hat{\sigma}$ representing an element in $\mathbb{G}_2$, $\mathcal{B}$ recovers the corresponding values $\hat{f} \in \mathbb{Z}_p$ and computes $-\hat{f}$. In case L_2 already contains $-\hat{f}$, $\mathcal{B}$ returns its associated bitstring $\hat{\sigma}'$. Otherwise, $\mathcal{B}$ chooses $\hat{\sigma}' \xleftarrow{R} \{0,1\}^{m_2}$, returns $\hat{\sigma}'$ and stores $(-\hat{f}, \hat{\sigma}')$ in L_2.

Pairing: Given two bitstrings $\sigma, \hat{\sigma}$ representing elements in $\mathbb{G}_1$ and $\mathbb{G}_2$, $\mathcal{B}$ recovers the corresponding values f from L_1 and $\hat{f}$ from L_2. In case L_T already contains $f \cdot \hat{f}$, $\mathcal{B}$ returns its associated bitstring τ. Otherwise, $\mathcal{B}$ chooses $\tau \xleftarrow{R} \{0,1\}^{m_T}$, returns τ and stores $(f \cdot \hat{f}, \tau)$ in L_T.

The group action and inversion oracle for $\mathbb{G}_T$ are simulated analogously to those for $\mathbb{G}_1$ and $\mathbb{G}_2$.

When $\mathcal{A}$ publishes σ_Q and $\hat{\sigma}_{\hat{Q}}$ such that $(f_Q, \sigma_Q) \in L_1$ and $(\hat{f}_{\hat{Q}}, \hat{\sigma}_{\hat{Q}}) \in L_2$ and $f_Q = \hat{f}_{\hat{Q}}$, $\mathcal{B}$ chooses four bitstrings $\sigma_0, \sigma_1, \sigma_2, \sigma_3 \xleftarrow{R} \{0,1\}^{m_1}$ and assigns polynomials $R, f_Q \cdot R, S, f_Q \cdot ((1-b) \cdot T + b \cdot U) \in \mathbb{Z}_p[R, S, T, U]$ to these values (in that order) in order to keep track of them. $\mathcal{B}$ stores $(R, \sigma_0), (f_Q \cdot R, \sigma_1), (S, \sigma_2), (f_Q \cdot ((1-b) \cdot T + b \cdot U), \sigma_3)$ in L_1 and provides $\mathcal{A}$ with $\sigma_0, \sigma_1, \sigma_2, \sigma_3$.

After this, $\mathcal{B}$ simulates the $\mathbb{G}_1$ group oracles as follows.

Group action in $\mathbb{G}_1$: Given two bitstrings σ_0, σ_1 representing elements in $\mathbb{G}_1$, $\mathcal{B}$ recovers the corresponding polynomials $f_0, f_1 \in \mathbb{Z}_p[R, S, T, U]$ and computes $f_0 + f_1$. In case L_1 already contains $f_0 + f_1$, $\mathcal{B}$ returns its associated bitstring. Otherwise, $\mathcal{B}$ chooses $\sigma \xleftarrow{R} \{0,1\}^{m_1}$, returns σ and stores $(f_0 + f_1, \sigma)$ in L_1.

Inversion in $\mathbb{G}_1$: Given a bitstring σ representing an element in $\mathbb{G}_1$, $\mathcal{B}$ recovers the corresponding values $f \in \mathbb{Z}_p[R,S,T,U]$ and computes $-f$. In case L_1 already contains $-f$, $\mathcal{B}$ returns its associated bitstring. Otherwise, $\mathcal{B}$ chooses $\sigma' \xleftarrow{R} \{0,1\}^{m_1}$, returns σ' and stores $(-f, \sigma')$ in L_1.

The group oracles for $\mathbb{G}_T$ are modified analogously to handle polynomials in $\mathbb{Z}_p[R,S,T,U]$.

When $\mathcal{A}$ has finished querying the group oracles, $\mathcal{A}$ outputs a bit b^*. Then, $\mathcal{B}$ chooses $r, s, t \xleftarrow{R} \mathbb{Z}_p$ and sets $R \leftarrow r, S \leftarrow s, T \leftarrow t, U \leftarrow rs$.

Now, if the simulation was consistent, no information about b got revealed and hence $\mathcal{A}$ can only guess b with probability $1/2$. Nevertheless, the simulation can be inconsistent, if two distinct polynomials in L_1 or in L_T evaluate to the same value after choosing concrete values for R, S, T, U. Note that such collisions cannot occur in L_2, since L_2 contains only polynomials of degree 0.

We need to prove that such a collision in L_1 (and likewise in L_T) cannot be caused by $\mathcal{A}$ itself. All substitutions in the formal variables R, S, T are independent, whereas only U depends on R and S. Therefore, $\mathcal{A}$ can only produce collisions using RS. In the beginning, the list L_1 only contains polynomials of degree 0, whereas later polynomials of total degree 1 are being added to L_1. Moreover, the group oracles do not increase the degree of the polynomials in L_1 as they only cover addition and inversion. Thus, $\mathcal{A}$ is not able to generate such collisions on purpose.

The same argumentation holds for L_T. Observe that the polynomials contained in L_T have at most total degree 1, since they arise from the multiplication of degree-0 polynomials in L_2 and polynomials of total degree at most 1 in L_1.

What remains to be shown is that the probability of a collision is negligible, i.e., that two distinct polynomials in L_1 and L_T evaluate to the same value after the substitution (or alternatively that their difference polynomial evaluates to 0). Suppose that $\mathcal{A}$ has issued q queries to the group oracles. Let $|L_1| = O(q)$ and $|L_T| = O(q)$, then there are $O(\binom{q}{2})$ possibilities of colliding polynomials. By the Schwartz-Zippel lemma and the collision argument, the probability of such an error in the simulation of the generic group is $O(\frac{q^2}{p})$ and is therefore negligible in the security parameter. The same kind of argument also holds for $\mathbb{G}_T$. □