Verifiably Encrypted Signatures: Security Revisited and a New Construction

Christian Hanser[1†], Max Rabkin[2,3], and Dominique Schröder[2]

[1] IAIK, Graz University of Technology
[2] CISPA, Saarland University
[3] International Max Planck Research School for Computer Science

Abstract In structure-preserving signatures on equivalence classes (SPS-EQ-$\mathcal{R}$), introduced at Asiacrypt 2014, each message M in $(\mathbb{G}^*)^\ell$ is associated to its projective equivalence class, and a signature commits to the equivalence class: anybody can transfer the signature to a new, scaled, representative.

In this work, we give the first black-box construction of a public-key encryption scheme from any SPS-EQ-$\mathcal{R}$ satisfying a simple new property which we call perfect composition. The construction does not involve any non-black-box technique and the implication is that such SPS-EQ-$\mathcal{R}$ cannot be constructed from one-way functions in a black-box way. The main idea of our scheme is to build a verifiable encrypted signature (VES) first and then apply the general transformation suggested by Calderon et al. (CT-RSA 2014).

The original definition of VES requires that the underlying signature scheme be correct and secure in addition to other security properties. The latter have been extended in subsequent literature, but the former requirements have sometimes been neglected, leaving a hole in the security notion. We show that Calderon et al.'s notion of resolution independence fills this gap.

Keywords: Structure preserving signatures, verifiably encrypted signatures, resolution independence, public-key encryption

1 Introduction

Structure-preserving signatures on equivalence classes (SPS-EQ-$\mathcal{R}$s) have been introduced at Asiacrypt 2014, and a corrected instantiation was given in a joint work with Fuchsbauer [9]. In an SPS-EQ-$\mathcal{R}$, each message M is a vector of group elements from a group of prime order p, and a signature commits the signer only to its projective equivalence class $[M]_{\mathcal{R}} = \{\lambda M : \lambda \in \mathbb{Z}_p^*\}$: anybody can transfer the signature to a new representative, scaling the message by an arbitrary factor

† Part of this work was done while visiting CISPA (Saarbrücken, Germany); supported by COST Action IC1306. Further supported by EU FP7 through project MATTHEW (GA No. 610436) and EU Horizon 2020 through project PRISMACLOUD (GA No. 644962).

and obtaining a new signature for the scaled message. SPS-EQ-$\mathcal{R}$s have many applications such as anonymous credentials [6] and have appealing properties, such as being compatible with Groth-Sahai zero-knowledge proofs [15]. In this work, we show how to construct verifiably encrypted signatures and public-key encryption from an SPS-EQ-$\mathcal{R}$.

Verifiably Encrypted Signatures. Bob wants to buy a theater ticket with an electronic check. That is, he wants to exchange one document, signed by himself, for another document, signed by the theater. If he sends the check before receiving the ticket, he worries that the theater will cash his check without issuing the ticket. On the other hand, the theater is not willing to issue the ticket without receiving a check.

A verifiably encrypted signature scheme (VES), introduced by Boneh, Gentry, Lynn and Shacham [3], can be used to resolve this impasse. A VES has two forms of signatures: plain and encrypted. Both forms of signature can be verified, and if the signer refuses to reveal the plain signature at the end of negotiations, the other party can appeal to a trusted third party (called the arbiter), who can recreate a plain signature given the corresponding encrypted signature.

Thus, in our example, the theater can provisionally send Bob a ticket with an encrypted signature, and once they receive Bob's signed check they can reveal the corresponding plain signature, and thus validate the provisional ticket. If they fail to do so, Bob can take the encrypted signature to the arbiter. The arbiter's investigation will reveal that Bob has indeed upheld his side of the deal, and so recreate the corresponding plain signature, giving Bob the ticket he has paid for. This protocol has the advantage that the arbiter need not participate unless there is a dispute.

VES from SPS-EQ-$\mathcal{R}$. We introduce a simple new property for SPS-EQ-$\mathcal{R}$ schemes, called *perfect composition*, which is satisfied by an existing construction in the generic group model, and show how to construct VESes from such schemes. In particular, this is the first VES construction from any kind of structure-preserving signature scheme, underlining the versatility of SPS-EQ-$\mathcal{R}$s. In our construction, each message is associated to a projective equivalence class. To create a plain signature, the signer signs one representative; to create an encrypted signature, she signs another. The scaling factor between these two representatives depends on the arbiter's key, allowing the arbiter to recover the plain signature from the encrypted one using the SPS-EQ-$\mathcal{R}$'s *change representative* algorithm.

Public-Key Encryption from SPS-EQ-$\mathcal{R}$. If the SPS-EQ-$\mathcal{R}$ allows perfect composition, then our VES construction satisfies resolution duplication, a property introduced very recently by Calderon, Meiklejohn, Shacham and Waters [5], which requires that a signature extracted by the arbiter is identical to that which would have been created by the signer. Not only does this prevent discrimination between arbiter-issued and signer-issued signatures, but VESes satisfying this property imply public-key encryption. This is particularly interesting because it is not possible to construct PKE from ordinary signatures (or equivalently, from

one-way functions) in a black-box way. Looking at this from the other side, it means that such an SPS-EQ-$\mathcal{R}$ cannot be constructed (black-box) from one-way functions.

1.1 Our Contribution

Our main contribution is twofold:

Verifiably Encrypted Signatures. We propose the first black-box construction of verifiably encrypted signature scheme from any structure-preserving signature scheme on equivalence classes satisfying a simple property. This construction does *not* combine an encryption scheme with an SPS-EQ-$\mathcal{R}$. Furthermore, all our security proofs hold in the standard model, under the Diffie-Hellman Inversion assumption.

We also revisit the security definitions of VES. The original definition of VES [3] requires that the underlying (ordinary) signature scheme be correct and secure in addition to other security properties. The latter properties have been extended in subsequent literature [18,27] but the requirements on the underlying scheme are sometimes neglected. We show that with this omission, resolution independence is absolutely essential not only to the unforgeability, but even to the correctness, of the underlying signature scheme. From the alternative viewpoint, we show that security including resolution independence is sufficient for the correctness and security of the underlying signature scheme.

Public-Key Encryption. We propose the first black-box construction of a CPA-secure public-key encryption scheme from any structure-preserving signature scheme on equivalence classes allowing perfect composition. The construction follows the idea of Calderon et. al [5]; it is black-box and does not involve known non-black-box techniques such as zero-knowledge. Given the well-known impossibility results, this shows that SPS-EQ-$\mathcal{R}$s allowing perfect composition cannot be constructed from one-way functions in a black-box way.

1.2 Related Work

Verifiably encrypted signatures and a first instantiation in the random oracle model were proposed by Boneh, Gentry, Lynn and Shacham [3]. After their invention, several instantiations were suggested in the RO model [29,26] and in the standard model [22,27,8]. The security model is treated in [3,18,27,5].

Impagliazzo and Rudich [20] show in their seminal work that cryptographic primitives can be classified as lying in one of two "worlds". The "Minicrypt" world contains those primitives that are equivalent to the weakest known assumption, the existence of one-way functions (OWFs), such as digital signatures [21,19,14,24,17]. The second world, "Cryptomania", includes primitives that require stronger assumptions such as public-key encryption (PKE), key-agreement (KA), oblivious transfer (OT) [11,12,25,4,28] and now SPS-EQ-$\mathcal{R}$.

1.3 Outline

In Section 2 we state the preliminaries. In Section 3 we discuss the relationship between resolution independence and the correctness and unforgeability of the underlying signature scheme of a VES. Then, in Section 4, we show how to generically build a VES from an SPS-EQ-$\mathcal{R}$ scheme. In Section 5, we then discuss the implication of PKE by certain SPS-EQ-$\mathcal{R}$. Finally, we conclude this paper in Section 6.

2 Preliminaries

A function $\epsilon : \mathbb{N} \to \mathbb{R}^+$ is called *negligible* if for all $c > 0$ there is a k_0 such that $\epsilon(k) < 1/k^c$ for all $k > k_0$. In the remainder of this paper, we use ϵ to denote such a negligible function. By $a \xleftarrow{\$} A$, we denote that a is chosen uniformly at random from the set A. We use the notation $A(a_1, \ldots, a_n; r)$ if we make the randomness r used by a probabilistic algorithm $A(a_1, \ldots, a_n)$ explicit.

Definition 1 (Bilinear Map). *Let $\mathbb{G}_1$, $\mathbb{G}_2$ and $\mathbb{G}_T$ be cyclic groups of prime order p, where we denote $\mathbb{G}_1$ and $\mathbb{G}_2$ additively and $\mathbb{G}_T$ multiplicatively. We write $\mathbb{G}_i^*$ for $\mathbb{G}_i \setminus \{0_{\mathbb{G}_i}\}$ where $i \in \{1, 2\}$. Let P and $\hat{P}$ be generators of $\mathbb{G}_1$ and $\mathbb{G}_2$, respectively. We call $e\colon \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$ a bilinear map or pairing if it is efficiently computable and the following holds:*

Bilinearity: $e(aP, b\hat{P}) = e(P, \hat{P})^{ab} \quad \forall\, a, b \in \mathbb{Z}_p$.
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 [7].

Definition 2 (Bilinear Group Generator). *A polynomial-time algorithm* BGGen *is a bilinear-group generator if it takes as input a security parameter 1^κ and outputs $\mathsf{BG} = (p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e, P, \hat{P})$ where the common group order p of the groups $\mathbb{G}_1, \mathbb{G}_2$ and $\mathbb{G}_T$ is a prime of bit-length κ, e is a pairing, and P and $\hat{P}$ are generators of $\mathbb{G}_1$ and $\mathbb{G}_2$, respectively.*

In this work we assume BGGen to be deterministic.[4]

Definition 3 (Diffie-Hellman Inversion Assumption (DHI) [23]). *Let $\mathbb{G}$ be a group of prime order p with $\log_2 p = \kappa$ and let $a \xleftarrow{\$} \mathbb{Z}_p^*$. Then, for every PPT adversary $\mathcal{A}$ there is a negligible function $\epsilon(\cdot)$ such that* $\Pr\left[\frac{1}{a}P \leftarrow \mathcal{A}(P, aP)\right] \leq \epsilon(\kappa)$.

[4] This is e.g. the case for BN-curves [2], the most common choice for Type-3 pairings.

2.1 Digital Signatures

Definition 4 (Digital Signature Scheme). *A* digital signature scheme *consists of the following polynomial time algorithms:*

$\mathsf{KeyGen}(1^\kappa)$**:** *A probabilistic algorithm that takes input a security parameter* $\kappa \in \mathbb{N}$ *and outputs a key pair* $(\mathsf{sk}, \mathsf{pk})$ *for message space* $\mathcal{M}$*.*

$\mathsf{Sign}(m, \mathsf{sk})$**:** *A probabilistic algorithm that takes input a message* $m \in \mathcal{M}$*, a secret key* sk *and outputs a signature* σ*.*

$\mathsf{Verify}(m, \sigma, \mathsf{pk})$**:** *A deterministic algorithm that takes input a message* $m \in \mathcal{M}$*, a signature* σ*, a public key* pk *and outputs* 1 *if* σ *is a valid signature for* M *under* pk *and* 0 *otherwise.*

A digital signature scheme is *secure* if it is *correct* and existentially unforgeable under adaptively chosen-message attacks. We define the properties below:

Definition 5 (Correctness). *A digital signature scheme* $(\mathsf{KeyGen}, \mathsf{Sign}, \mathsf{Verify})$ *is called* correct *if*

$$\forall \kappa > 0 \ \forall (\mathsf{sk}, \mathsf{pk}) \xleftarrow{\$} \mathsf{KeyGen}(1^\kappa) \ \forall m \in \mathcal{M} : \quad \mathsf{Verify}(m, \mathsf{Sign}(m, \mathsf{sk}), \mathsf{pk}) = 1$$

Definition 6 (EUF-CMA). *A digital signature scheme* $(\mathsf{KeyGen}, \mathsf{Sign}, \mathsf{Verify})$ *is called* existentially unforgeable under adaptively chosen-message attacks *if for all PPT algorithms* $\mathcal{A}$ *having access to a signing oracle* $\mathcal{O}(\cdot, \mathsf{sk})$*, there is a negligible function* $\epsilon(\cdot)$ *such that:*

$$\Pr\left[\begin{matrix}(\mathsf{sk}, \mathsf{pk}) \xleftarrow{\$} \mathsf{KeyGen}(1^\kappa), \\ (m^*, \sigma^*) \xleftarrow{\$} \mathcal{A}^{\mathcal{O}(\cdot, \mathsf{sk})}(\mathsf{pk})\end{matrix} : \begin{matrix} m^* \notin Q \ \ \wedge \\ \mathsf{Verify}(m^*, \sigma^*, \mathsf{pk}) = 1\end{matrix}\right] \leq \epsilon(\kappa),$$

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

2.2 Structure-Preserving Signatures on Equivalence Classes

In a structure-preserving signature scheme [1], public keys, messages and signatures consist only of group elements of a bilinear group. The verification algorithm verifies a signature solely through group membership tests and by evaluating pairing-product equations.

An SPS-EQ-$\mathcal{R}$ scheme is a structure-preserving signature scheme that is defined either on the message space $(\mathbb{G}_1^*)^\ell$ or $(\mathbb{G}_2^*)^\ell$, where $\ell > 1$ and $|\mathbb{G}_1| = |\mathbb{G}_2| = p$ is prime. Since $\mathbb{Z}_p^\ell$ is a vector space, it is possible to define—in analogy to the projective space—a projective equivalence relation $\sim_\mathcal{R}$ that partitions $\mathbb{Z}_p^\ell$ into projective equivalence classes. This equivalence relation then further propagates onto $(\mathbb{G}_i^*)^\ell$ for $i \in \{1, 2\}$.

Now, an SPS-EQ-$\mathcal{R}$ scheme signs such equivalence classes by signing arbitrary representatives of such classes. When given a message-signature pair, anyone can derive a valid message-signature pair for every other representative of this class. This is done by multiplying each message vector component by the

same scalar and by consistently updating the corresponding signature. Clearly, this requires unforgeability to be defined with respect to equivalence classes. This means that after querying signatures for messages M_i, no adversary should be able to output a forgery for a message M^* belonging to a class different from the classes $[M_i]_{\mathcal{R}}$.

We restate the syntax and the security properties of structure-preserving signatures on equivalence classes from [16,9,10]:

Definition 7 (Structure-Preserving Signature Scheme on Equivalence Classes (SPS-EQ-$\mathcal{R}$)). *An SPS-EQ-$\mathcal{R}$ scheme* SPSEQ *on* $(\mathbb{G}_i^*)^\ell$ *consists of the following polynomial-time algorithms:*

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

$\mathsf{KeyGen}_{\mathcal{R}}(\mathsf{BG}, \ell)$**:** *A probabilistic algorithm, which on input a bilinear group* BG *and a vector length* $\ell > 1$ *outputs a key pair* $(\mathsf{sk}, \mathsf{pk})$.

$\mathsf{Sign}_{\mathcal{R}}(M, \mathsf{sk})$**:** *A probabilistic algorithm, which on input a representative* $M \in (\mathbb{G}_i^*)^\ell$ *of an equivalence class* $[M]_{\mathcal{R}}$ *and a secret key* sk *outputs a signature* σ *for the representative* M *of equivalence class* $[M]_{\mathcal{R}}$.

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

$\mathsf{Verify}_{\mathcal{R}}(M, \sigma, \mathsf{pk})$**:** *A deterministic algorithm, which given 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.*

$\mathsf{VKey}_{\mathcal{R}}(\mathsf{sk}, \mathsf{pk})$**:** *A deterministic algorithm, which given a secret key* sk *and a public key* pk *checks both keys for consistency and returns* 1 *on success and* 0 *otherwise.*

Definition 8 (Correctness). *An SPS-EQ-$\mathcal{R}$ scheme* SPSEQ *on* $(\mathbb{G}_i^*)^\ell$ *is called* correct *if for all security parameters* $\kappa \in \mathbb{N}$*, for all* $\ell > 1$*, all bilinear groups* $\mathsf{BG} \leftarrow \mathsf{BGGen}_{\mathcal{R}}(1^\kappa)$*, all key pairs* $(\mathsf{sk}, \mathsf{pk}) \xleftarrow{\$} \mathsf{KeyGen}_{\mathcal{R}}(\mathsf{BG}, \ell)$*, all messages* $M \in (\mathbb{G}_i^*)^\ell$ *and all* $\lambda \in \mathbb{Z}_p^*$ *we have:*

$$\begin{aligned}
&\mathsf{VKey}_{\mathcal{R}}(\mathsf{sk}, \mathsf{pk}) = 1 \quad \textit{and} \\
&\Pr\left[\mathsf{Verify}_{\mathcal{R}}(M, \mathsf{Sign}_{\mathcal{R}}(M, \mathsf{sk}), \mathsf{pk}) = 1\right] = 1 \quad \textit{and} \\
&\Pr\left[\mathsf{Verify}_{\mathcal{R}}(\mathsf{ChgRep}_{\mathcal{R}}(M, \mathsf{Sign}_{\mathcal{R}}(M, \mathsf{sk}), \lambda, \mathsf{pk}), \mathsf{pk}) = 1\right] = 1.
\end{aligned}$$

Definition 9 (EUF-CMA). *An SPS-EQ-$\mathcal{R}$ scheme* SPSEQ *on* $(\mathbb{G}_i^*)^\ell$ *is called* existentially unforgeable under adaptively chosen-message attacks *if, for all PPT algorithms* $\mathcal{A}$ *having access to a signing oracle* $\mathcal{O}(\mathsf{sk}, M)$*, there is a negligible function* $\epsilon(\cdot)$ *such that:*

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

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

We now introduce the following new property:

Definition 10. *An SPS-EQ-$\mathcal{R}$ scheme* SPSEQ allows perfect composition *if for all random tapes r and tuples* $(\mathsf{sk}, \mathsf{pk}, M, \sigma, \lambda)$:

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

it holds that $(\lambda M, \mathsf{Sign}_{\mathcal{R}}(\lambda M, \mathsf{sk}; r)) = \mathsf{ChgRep}_{\mathcal{R}}(M, \sigma, \lambda, \mathsf{pk}; 1)$.

Intuitively, this requires that $\mathsf{ChgRep}_{\mathcal{R}}$ executed with random coins fixed to 1 updates only the parts of a signature that are affected by updating the representative from M to λM, not changing the randomness of $\mathsf{Sign}_{\mathcal{R}}$.

In [10], a standard model SPS-EQ-$\mathcal{R}$ construction is presented. Unfortunately, it does not satisfy the above definition, but the scheme in [9], which is secure in the generic group model, does.

2.3 Verifiably Encrypted Signatures

Below, we give the abstract model of verifiably encrypted signatures, adapted from [3].

Definition 11 (Verifiably Encrypted Signature Scheme (VES)). *A verifiably encrypted signature scheme* VES *consists of the following polynomial time algorithms:*

$\mathsf{AKeyGen}(1^\kappa)$**:** *Given a security parameter κ, this probabilistic algorithm outputs a key pair* $(\mathsf{ask}, \mathsf{apk})$*, where* ask *is the private key and* apk *the corresponding public key of the arbiter.*

$\mathsf{KeyGen}(1^\kappa)$**:** *Given a security parameter κ, this probabilistic algorithm outputs a private signing key* sk *and a public verification key* pk *for message space $\mathcal{M}$.*

$\mathsf{Sign}(m, \mathsf{sk})$**:** *Given a message $m \in \mathcal{M}$ and a signing key* sk*, this probabilistic algorithm outputs a signature σ under* sk *on m.*

$\mathsf{Verify}(m, \sigma, \mathsf{pk})$**:** *Given a message $m \in \mathcal{M}$ and a public key* pk*, this deterministic algorithm outputs 1 iff σ is a valid signature on m under* pk *and 0 otherwise.*

$\mathsf{VESign}(m, \mathsf{sk}, \mathsf{apk})$**:** *Given a message $m \in \mathcal{M}$, a signing key* sk *and an arbiter public key* apk*, this probabilistic algorithm outputs an encrypted signature ω under* sk *on message m.*

$\mathsf{VEVerify}(m, \omega, \mathsf{pk}, \mathsf{apk})$**:** *Given a message $m \in \mathcal{M}$, an encrypted signature ω, a public key* pk *and an arbiter public key* apk*, this deterministic algorithm outputs 1 if ω is a valid encrypted signature on m under* pk *and 0 otherwise.*

$\mathsf{Resolve}(m, \omega, \mathsf{ask}, \mathsf{pk})$**:** *Given a message $m \in \mathcal{M}$, an encrypted signature ω, an arbiter secret key* ask *and a public key* pk*, this (probabilistic) algorithm outputs a valid signature σ on m under* pk*.*

We call a VES *secure* if it is *complete*, *unforgeable*, *opaque*, *extractable*, *abuse free* and *resolution independent*. We define these properties below.

Completeness says that any honestly computed VES always verifies and that moreover the arbiter can always extract a valid signature.

Definition 12 (Completeness). *A VES* VES *is* complete *if for all* $\kappa > 0$*, all* $(\mathsf{ask}, \mathsf{apk}) \xleftarrow{\$} \mathsf{AKeyGen}(1^\kappa)$*, all* $(\mathsf{sk}, \mathsf{pk}) \xleftarrow{\$} \mathsf{KeyGen}(1^\kappa)$*, and all messages* $m \in \mathcal{M}$*, for* $\omega \xleftarrow{\$} \mathsf{VESign}(m, \mathsf{sk}, \mathsf{apk})$ *it holds that*

$$\Pr\left[\mathsf{VEVerify}(m, \omega, \mathsf{pk}, \mathsf{apk}) = 1\right] = 1 \quad \textit{and}$$
$$\Pr\left[\mathsf{Verify}(m, \mathsf{Resolve}(m, \omega, \mathsf{ask}, \mathsf{pk}), \mathsf{pk}) = 1\right] = 1.$$

Unforgeability says that it should be infeasible to produce a valid encrypted signature for an unknown secret key.

Definition 13 (Unforgeability). *A VES* VES *is* unforgeable *if for all PPT algorithms* $\mathcal{A}$ *having access to oracles* $\mathcal{O} \leftarrow \{\mathsf{VESign}(\cdot, \mathsf{sk}, \mathsf{apk}), \mathsf{Resolve}(\cdot, \cdot, \mathsf{ask}, \mathsf{pk}), \mathsf{Sign}(\cdot, \mathsf{sk})\}$*, there is a negligible function* $\epsilon(\cdot)$ *such that:*

$$\Pr\left[\begin{array}{l|ll}(\mathsf{ask}, \mathsf{apk}) \xleftarrow{\$} \mathsf{AKeyGen}(1^\kappa), & & m^* \notin Q \ \ \wedge \\ (\mathsf{sk}, \mathsf{pk}) \xleftarrow{\$} \mathsf{KeyGen}(1^\kappa), & : & \mathsf{VEVerify}(m^*, \omega^*, \mathsf{pk}, \mathsf{apk}) = 1 \\ (m^*, \omega^*) \xleftarrow{\$} \mathcal{A}^{\mathcal{O}}(\mathsf{pk}, \mathsf{apk}) & & \end{array}\right] \leq \epsilon(\kappa),$$

where Q *is the set of messages which were queried to the oracles.*

Opacity basically requires that only the arbiter should be able to pull out the underlying signature.

Definition 14 (Opacity). *A VES* VES *is* opaque *if for all PPT algorithms* $\mathcal{A}$ *having access to oracles* $\mathcal{O} \leftarrow \{\mathsf{VESign}(\cdot, \mathsf{sk}, \mathsf{apk}), \mathsf{Resolve}(\cdot, \cdot, \mathsf{ask}, \mathsf{pk})\}$*, there is a negligible function* $\epsilon(\cdot)$ *such that:*

$$\Pr\left[\begin{array}{l|ll}(\mathsf{ask}, \mathsf{apk}) \xleftarrow{\$} \mathsf{AKeyGen}(1^\kappa), & & m^* \notin Q \ \ \wedge \\ (\mathsf{sk}, \mathsf{pk}) \xleftarrow{\$} \mathsf{KeyGen}(1^\kappa), & : & \mathsf{Verify}(m^*, \sigma^*, \mathsf{pk}) = 1 \\ (m^*, \sigma^*) \xleftarrow{\$} \mathcal{A}^{\mathcal{O}}(\mathsf{pk}, \mathsf{apk}) & & \end{array}\right] \leq \epsilon(\kappa),$$

where Q *is the set of messages queried to the* Resolve *oracle.*

In addition to the above property, we have to guarantee that it is indeed possible for the arbiter to extract the underlying signature, which is covered by the following property.

Definition 15 (Extractability). *A VES* VES *is* extractable *if for all PPT algorithms* $\mathcal{A}$ *having access to oracles* $\mathcal{O} \leftarrow \{\mathsf{Resolve}(\cdot, \cdot, \mathsf{ask}, \cdot)\}$*, there is a negligible function* $\epsilon(\cdot)$ *such that:*

$$\Pr\left[\begin{array}{l|ll}(\mathsf{ask}, \mathsf{apk}) \xleftarrow{\$} \mathsf{AKeyGen}(1^\kappa), & & \sigma \xleftarrow{\$} \mathsf{Resolve}(m^*, \omega^*, \mathsf{ask}, \mathsf{pk}^*) \ \ \wedge \\ (\mathsf{pk}^*, m^*, \omega^*) \xleftarrow{\$} \mathcal{A}^{\mathcal{O}}(\mathsf{apk}) & : & \mathsf{VEVerify}(m^*, \omega^*, \mathsf{pk}^*, \mathsf{apk}) = 1 \ \ \wedge \\ & & \mathsf{Verify}(m^*, \sigma, \mathsf{pk}^*) = 0 \end{array}\right] \leq \epsilon(\kappa).$$

Abuse freeness guarantees that even if an adversary is colluding with the arbiter, it is unable to forge a valid encrypted signature.

Definition 16 (Abuse Freeness). *A VES* VES *is* abuse free *if for all PPT algorithms* $\mathcal{A}$ *having access to oracles* $\mathcal{O} \leftarrow \{\mathsf{VESign}(\cdot, \mathsf{sk}, \mathsf{apk})\}$*, there is a negligible function* $\epsilon(\cdot)$ *such that:*

$$\Pr\left[\begin{array}{l|ll}(\mathsf{ask},\mathsf{apk}) \xleftarrow{\$} \mathsf{AKeyGen}(1^\kappa), & & m^* \notin Q \ \ \wedge \\ (\mathsf{sk},\mathsf{pk}) \xleftarrow{\$} \mathsf{KeyGen}(1^\kappa), & : & \mathsf{VEVerify}(m^*, \omega^*, \mathsf{pk}, \mathsf{apk}) = 1 \\ (m^*, \omega^*) \xleftarrow{\$} \mathcal{A}^{\mathcal{O}}(\mathsf{pk}, \mathsf{ask}, \mathsf{apk}) & & \end{array}\right] \leq \epsilon(\kappa),$$

where Q *is the set of messages queried to the* VESign *oracle.*

Extractability and abuse freeness were introduced by Rückert and Schröder in [27].

Additionally, Calderon et al. [5] have identified another property that is called resolution independence. This property is crucial for a VES to be secure, as we will discuss in Section 3.

Definition 17 (Resolution Independence). *A VES* VES *is* resolution independent *if for all* $\kappa > 0$*, all* $(\mathsf{ask},\mathsf{apk}) \xleftarrow{\$} \mathsf{AKeyGen}(1^\kappa)$*,* $(\mathsf{sk},\mathsf{pk}) \xleftarrow{\$} \mathsf{KeyGen}(1^\kappa)$*, and all messages* m*, the outputs of* $\mathsf{Sign}(m, \mathsf{sk})$ *and* $\mathsf{Resolve}(m, \mathsf{VESign}(m, \mathsf{sk}, \mathsf{apk}), \mathsf{ask}, \mathsf{pk})$ *are distributed identically.*

In [5], the authors showed that VES constructions imply public key encryption if they additionally satisfy a property called *resolution duplication*. Loosely speaking, a VES is resolution duplicate if the signatures returned by the signer and the arbiter are identical.

Definition 18 (Resolution Duplication). *A VES* VES *is* resolution duplicate *if it is resolution independent and fulfills the following properties:*

Deterministic Resolution: *The algorithm* $\mathsf{Resolve}$ *is deterministic.*

Extraction: *There exists an additional PPT algorithm* $\mathsf{Extract}(\cdot,\cdot,\cdot)$*, such that for all* $\kappa > 0$*, all* $(\mathsf{ask},\mathsf{apk}) \xleftarrow{\$} \mathsf{AKeyGen}(1^\kappa)$*,* $(\mathsf{sk},\mathsf{pk}) \xleftarrow{\$} \mathsf{KeyGen}(1^\kappa), m \in \mathcal{M}$*, and random tapes* $r \in \{0,1\}^*$*, it is the case that*

$$\mathsf{Extract}(m, \mathsf{sk}, r) = \mathsf{Resolve}(m, \mathsf{VESign}(m, \mathsf{sk}, \mathsf{apk}; r), \mathsf{ask}, \mathsf{pk}).$$

Up to now numerous standard-model VES constructions have been proposed, but not all constructions so far are resolution-duplicate; in particular not the ones with a randomized $\mathsf{Resolve}$ algorithm [5].

3 The Importance of Resolution Independence

In Boneh et al.'s original definition of a VES [3], the underlying signature scheme is required to be secure, in addition to the security properties of the encrypted signatures: completeness, unforgeability and opacity. Rückert and Schröder [27]

added the properties of extractability and abuse freeness, and Calderon et al. [5] added the properties of resolution independence, but both omit (or are at least unclear about) the requirement that the underlying signature scheme be secure. Indeed, the latter paper says that they "additionally provide the adversary with access to the Sign oracle, as otherwise the underlying signature scheme could be completely broken and the VES would still be considered unforgeable." In fact, it can be completely broken anyway.

We will show that, with this omission, resolution independence is absolutely essential to not only the unforgeability, but even the correctness, of the underlying scheme. Resolution independence supplies the necessary glue to connect the security properties of the encrypted scheme to the underlying scheme. Contrapositively, we show that security including resolution independence is sufficient for the correctness and security of the underlying signature scheme, so that does not need to be proven separately. To be clear, we formally define what is meant by the underlying signature scheme.

Definition 19. *Let* VES *be a VES. Then we call* $\mathsf{Sig} = (\mathsf{SKeyGen}, \mathsf{Sign}, \mathsf{Verify})$ *the underlying signature scheme of* VES*, where* $\mathsf{SKeyGen}(1^\kappa)$ *outputs* $(\mathsf{sk}, \mathsf{pk}) \xleftarrow{\$} \mathsf{KeyGen}(1^\kappa)$.

3.1 Counterexample

We now show that completeness, unforgeability, opacity, extractability and abuse freeness together do not imply the correctness or security of the underlying scheme.

Let $\mathsf{VES} = (\mathsf{AKeyGen}, \mathsf{KeyGen}, \mathsf{Sign}, \mathsf{Verify}, \mathsf{VESign}, \mathsf{VEVerify}, \mathsf{Resolve})$ be a VES with messages of length n, and let $\mathsf{VES}' = (\mathsf{AKeyGen}, \mathsf{KeyGen}, \mathsf{Sign}', \mathsf{Verify}, \mathsf{VESign}, \mathsf{VEVerify}, \mathsf{Resolve})$ where $\mathsf{Sign}'(m, \mathsf{sk})$ computes and outputs $\mathsf{Sign}(0^n, \mathsf{sk})$.

Theorem 1. *If* VES *is complete, unforgeable, opaque, extractable and abuse free, then so is* VES'.

Proof. The adversary in the unforgeability game must output a valid encrypted signature, but the set of valid encrypted signatures in VES and VES' are the same, and we have only weakened the oracles (by making Sign provide signatures only on 0^n), so unforgeability is preserved. The other properties do not mention the Sign algorithm at all, so they are unaffected. □

This scheme is intuitively both incorrect (as the signatures produced by Sign' cannot be verified) and insecure (as it gives away a forgery as soon as it is called on a message other than 0^n. Nevertheless, VES' is secure as defined in [27], since their definition does not include the security of the underlying signature scheme. It is also much more catastrophically insecure than the separating example in [5, Section 3], which motivated the definition of resolution independence.

Theorem 2. *The underlying signature scheme* Sig *of* VES' *is neither correct nor secure.*

3.2 Filling the Gap

Lemma 1. *If* VES *is a complete and resolution independent VES, then its underlying signature scheme* Sig *is correct.*

Proof. By completeness, for all $\kappa > 0$, $(\mathsf{ask}, \mathsf{apk}) \xleftarrow{\$} \mathsf{AKeyGen}(1^\kappa), (\mathsf{sk}, \mathsf{pk}) \xleftarrow{\$} \mathsf{KeyGen}(1^\kappa)$ and all messages $m \in \mathcal{M}$, for $\omega \xleftarrow{\$} \mathsf{VESign}(m, \mathsf{sk}, \mathsf{apk})$, with probability 1,

$$\mathsf{Verify}(m, \mathsf{Resolve}(m, \omega, \mathsf{ask}, \mathsf{pk}), \mathsf{pk}) = 1.$$

By resolution independence, $\mathsf{Resolve}(m, \omega, \mathsf{ask}, \mathsf{pk})$ is identically distributed to $\mathsf{Sign}(m, \mathsf{sk})$, so with probability 1,

$$\mathsf{Verify}(m, \mathsf{Sign}(m, \mathsf{sk}), \mathsf{pk}) = 1. \qquad \square$$

Lemma 2. *If* VES *is an opaque and resolution independent VES, then its underlying signature scheme* Sig *is EUF-CMA-secure.*

Proof. Let VES be a resolution independent VES, and let Sig be the underlying signature scheme. We assume that there is an efficient adversary $\mathcal{A}$ breaking the EUF-CMA security of Sig with non-negligible probability, and construct an adversary $\mathcal{B}$ that uses $\mathcal{A}$ to break the opacity of VES.

$\mathcal{B}$ takes as input an arbiter's public key apk and a signer's public key pk (with unknown corresponding private keys ask and sk), and passes pk as input to $\mathcal{A}$. Whenever $\mathcal{A}$ tries to query the Sign oracle on message m, $\mathcal{B}$ forwards m to its VESign oracle, obtaining $\omega = \mathsf{VESign}(m, \mathsf{sk}, \mathsf{apk})$; $\mathcal{B}$ then queries (m, ω) to its Resolve oracle, obtaining $\sigma = \mathsf{Resolve}(m, \mathsf{VESign}(m, \mathsf{sk}, \mathsf{apk}), \mathsf{ask}, \mathsf{pk})$, which it returns to $\mathcal{A}$. When $\mathcal{A}$ outputs (m^*, σ^*), $\mathcal{B}$ outputs the same.

By resolution independence, $\mathsf{Sign}(m, \mathsf{sk})$ and $\mathsf{Resolve}(m, \mathsf{VESign}(m, \mathsf{sk}, \mathsf{apk}), \mathsf{ask}, \mathsf{pk})$ are identically distributed, so we perfectly simulate $\mathcal{A}$'s Sign oracle.

If $\mathcal{A}$ never queried m^* to Sign, then $\mathcal{B}$ never queried m^* to Resolve, and so $\mathcal{B}$ has the same non-negligible success probability as $\mathcal{A}$. $\square$

Theorem 3. *If a VES is complete, opaque and resolution independent, then its underlying signature scheme* Sig *is correct and secure.*

Proof. By Lemmas 1 and 2. $\square$

4 Verifiably Encrypted Signatures from SPS-EQ-$\mathcal{R}$

In Scheme 1, we show how a VES can be built using any SPS-EQ-$\mathcal{R}$ construction that allows perfect composition as a black box. In particular, the VES construction only requires the SPS-EQ-$\mathcal{R}$ construction to be correct, EUF-CMA secure and to fulfill perfect composition (Definition 10).

Note 1. Observe that, independently of the instantiation of Scheme 1 with a concrete SPS-EQ-$\mathcal{R}$, the efficiency of the Verify resp. VEVerify can be improved by precomputing parts of the pairing product equations that solely depend on

$\mathsf{AKeyGen}(1^\kappa)$**:** Given a security parameter κ, compute $\mathsf{BG} \leftarrow \mathsf{BGGen}_{\mathcal{R}}(1^\kappa)$, pick $a \xleftarrow{\$} \mathbb{Z}_p^*$, compute $A \leftarrow aP$ and output $(\mathsf{ask}, \mathsf{apk}) \leftarrow (a, (\mathsf{BG}, A))$.

$\mathsf{KeyGen}(1^\kappa)$**:** Given a security parameter κ, compute $\mathsf{BG} \leftarrow \mathsf{BGGen}_{\mathcal{R}}(1^\kappa)$ and output $(\mathsf{sk}, \mathsf{pk}) \xleftarrow{\$} \mathsf{KeyGen}_{\mathcal{R}}(\mathsf{BG}, \ell = 3)$.

$\mathsf{Sign}(m, \mathsf{sk}; (r_1, r_2))$**:** Given a message $m \in \mathbb{Z}_p^*$, secret key sk and a random tape $(r_1, r_2) \in \{0,1\}^*$, pick $s \xleftarrow{\$} \mathbb{Z}_p^*$ using r_1 and compute $\sigma' \leftarrow \mathsf{Sign}_{\mathcal{R}}((msP, sP, P), \mathsf{sk}; r_2)$ using the remaining coins r_2. Finally, output $\sigma \leftarrow (\sigma', sP)$.

$\mathsf{Verify}(m, \sigma, \mathsf{pk})$**:** Given a message $m \in \mathbb{Z}_p^*$, a signature $\sigma = (\sigma', S)$ and a public key pk, output whatever $\mathsf{Verify}_{\mathcal{R}}((mS, S, P), \sigma', \mathsf{pk})$ outputs.

$\mathsf{VESign}(m, \mathsf{sk}, \mathsf{apk}; (r_1, r_2))$**:** Given a message $m \in \mathbb{Z}_p^*$, secret key sk, the arbiter public key $\mathsf{apk} = A$ and a random tape $(r_1, r_2) \in \{0,1\}^*$, pick $s \xleftarrow{\$} \mathbb{Z}_p^*$ using r_1 and compute $\omega' \leftarrow \mathsf{Sign}_{\mathcal{R}}((msA, sA, A), \mathsf{sk}; r_2)$ using the remaining coins r_2. Finally, output $\omega \leftarrow (\omega', sA)$.

$\mathsf{VEVerify}(m, \omega, \mathsf{pk}, \mathsf{apk})$**:** Given a message $m \in \mathbb{Z}_p^*$, an encrypted signature $\omega = (\omega', W)$, a public key pk and an arbiter public key $\mathsf{apk} = A$, output whatever $\mathsf{Verify}_{\mathcal{R}}((mW, W, A), \omega', \mathsf{pk})$ outputs.

$\mathsf{Resolve}(m, \omega, \mathsf{ask}, \mathsf{pk})$**:** Given a message $m \in \mathbb{Z}_p^*$, an encrypted signature $\omega = (\omega', sA)$, a public key pk and an arbiter secret key $\mathsf{ask} \leftarrow a$, check whether $\mathsf{VEVerify}(m, \omega, \mathsf{pk}, \mathsf{apk}) \stackrel{?}{=} 1$ and return $\perp$ if this is not the case. Otherwise, compute $((msP, sP, P), \sigma') \leftarrow \mathsf{ChgRep}_{\mathcal{R}}((msA, sA, A), \omega, \frac{1}{a}, \mathsf{pk}; 1)$ and output $\sigma \leftarrow (\sigma', sP)$.

Scheme 1: A VES Construction from SPS-EQ-$\mathcal{R}$.

P and pk resp. A and pk, and including the resulting $\mathbb{G}_T$ elements into (the updated) user public key pk.

In the following, we are going to analyze the security of Scheme 1 and prove completeness, unforgeability, opacity and abuse freeness as well as resolution duplication.

Theorem 4. *The VES in Scheme 1 is complete.*

Proof. The completeness proof of Scheme 1 is straight-forward and therefore omitted here. □

Theorem 5. *The VES in Scheme 1 is unforgeable given that the underlying SPS-EQ-$\mathcal{R}$ scheme is unforgeable.*

Proof. We assume that there is an efficient adversary $\mathcal{A}$ winning the unforgeability game with non-negligible probability; then we are able to construct an adversary $\mathcal{B}$ that uses $\mathcal{A}$ to break the EUF-CMA security of the underlying SPS-EQ-$\mathcal{R}$ scheme with non-negligible probability.

$\mathcal{B}$ obtains $\mathsf{pk}_{\mathcal{R}}$ of the SPS-EQ-$\mathcal{R}$ scheme with $\ell = 3$ (and thereby implicitly the bilinear group BG) from the challenger $\mathcal{C}$ of the EUF-CMA security game, and sets $\mathsf{pk} \leftarrow \mathsf{pk}_{\mathcal{R}}$. Then $\mathcal{B}$ picks $a \xleftarrow{\$} \mathbb{Z}_p^*$, computes $A \leftarrow aP$ and sets $(\mathsf{ask}, \mathsf{apk}) \leftarrow (a, (\mathsf{BG}, A))$. Next, $\mathcal{B}$ sets up a list $L \leftarrow \emptyset$ to keep track of representatives queried to $\mathcal{C}$, runs $\mathcal{A}$ on $(\mathsf{pk}, \mathsf{apk})$ and answers $\mathcal{A}$'s oracle queries to the

Resolve oracle as in the real game and simulates queries to all other oracles as follows:

Sign$(\cdot, \mathsf{sk})$: If $\mathcal{A}$ submits a query for $m \in \mathbb{Z}_p^*$, $\mathcal{B}$ queries $\mathcal{C}$'s signing oracle for the message (msP, sP, P) for $s \xleftarrow{\$} \mathbb{Z}_p^*$, gets in return a corresponding signature σ', sets $L[m] \leftarrow L[m] \cup \{(msA, sA, A)\}$ and gives $\sigma \leftarrow (\sigma', sP)$ to $\mathcal{A}$.

VESign$(\cdot, \mathsf{sk}, \mathsf{apk})$: If $\mathcal{A}$ submits a query for $m \in \mathbb{Z}_p^*$, $\mathcal{B}$ queries $\mathcal{C}$'s signing oracle for the message (msA, sA, A) for $s \xleftarrow{\$} \mathbb{Z}_p^*$, gets in return a corresponding signature ω', sets $L[m] \leftarrow L[m] \cup \{(msA, sA, A)\}$ and gives $\omega \leftarrow (\omega', sA)$ as encrypted signature to $\mathcal{A}$.

If at some point $\mathcal{A}$ outputs a valid encrypted message-signature pair $(m^*, \omega^* = (\omega'^*, W^*))$, such that it has not previously queried m^* to any of the oracles, then $\mathcal{B}$ will output $((m^*W^*, W^*, A), \omega'^*)$ to $\mathcal{C}$.

Note that the distribution of all values returned to $\mathcal{A}$ during the simulation is identical to the distribution of these values during a real game.

By construction, $((m^*W^*, W^*, A), \omega'^*)$ constitutes a valid SPS-EQ-$\mathcal{R}$ message-signature pair. It remains to show that for $M^* = (m^*W^*, W^*, A)$, the class $[M^*]_{\mathcal{R}}$ is different from all classes represented by elements in L, if m^* is different from all messages queried to the oracles. VEVerify demands that the third vector component of M^* be A, which uniquely determines the representative for each class and allows for comparison. Now, if there is some $M_i = (m_iW_i, W_i, A) \in L$ queried to the VESign or the Sign oracle coinciding with M^* in the second component, then both vectors still differ in the first component for $m^* \neq m_i$. Likewise, if they coincide in the first component for $m^* \neq m_i$, then they cannot have equal second components. Hence, $M^* \neq M_i$ for any M_i in L. □

Theorem 6. *The VES in Scheme 1 is opaque given that the DHI assumption holds in $\mathbb{G}_1$ and that the underlying SPS-EQ-$\mathcal{R}$ is unforgeable.*

Proof. We assume that there is an efficient adversary $\mathcal{A}$ winning the opacity game with non-negligible probability. Then we are able to construct an adversary $\mathcal{B}$ that uses $\mathcal{A}$ either to break with non-negligible probability the EUF-CMA security of the underlying SPS-EQ-$\mathcal{R}$ scheme (Type-1 adversary) if $\mathcal{A}$ has neither queried the VESign nor the Resolve oracle for m^*; or the DHI assumption (Type-2 adversary) if $\mathcal{A}$ has only queried the VESign but not the Resolve oracle for m^*.

In the following, $\mathcal{B}$ guesses $\mathcal{A}$'s strategy, i.e., the type of forgery $\mathcal{A}$ will conduct. We are now going to describe the setup, the initialization of the environment, the reduction and the abort conditions for each type.

Type-1: $\mathcal{B}$ obtains $\mathsf{pk}_{\mathcal{R}}$ of the SPS-EQ-$\mathcal{R}$ scheme with $\ell = 3$ (and thereby implicitly the bilinear group BG) from the challenger $\mathcal{C}$ of the EUF-CMA security game and sets $\mathsf{pk} \leftarrow \mathsf{pk}_{\mathcal{R}}$. Furthermore, $\mathcal{B}$ picks $a \xleftarrow{\$} \mathbb{Z}_p^*$, computes $A \leftarrow aP$ and sets $(\mathsf{ask}, \mathsf{apk}) \leftarrow (a, (\mathsf{BG}, A))$. Next, $\mathcal{B}$ runs $\mathcal{A}$ on $(\mathsf{pk}, \mathsf{apk})$ and answers $\mathcal{A}$'s oracle queries to the Resolve oracle as in the real game and simulates queries to the other oracle as follows:

$\mathsf{VESign}(\cdot, \mathsf{sk}, \mathsf{apk})$**:** If $\mathcal{A}$ submits a query for $m \in \mathbb{Z}_p^*$, $\mathcal{B}$ queries $\mathcal{C}$'s signing oracle for the message (msA, sA, A) for $s \xleftarrow{\$} \mathbb{Z}_p^*$, then $\mathcal{B}$ gets in return a signature ω' and outputs (ω', sA).

If at some point $\mathcal{A}$ outputs a valid message-signature pair (m^*, σ^*) with $\sigma^* = (\sigma'^*, S^*)$, and neither has queried to the VESign nor to the $\mathsf{Resolve}$ oracle for m^*, then $\mathcal{B}$ will output $((m^*S^*, S^*, P), \sigma'^*)$ to $\mathcal{C}$. In case that $\mathcal{A}$ has queried the VESign oracle for m^*, $\mathcal{B}$ will abort.

Note that the distribution of all values returned to $\mathcal{A}$ during the simulation is identical to the distribution of these values during a real game, which makes the simulation perfect.

By construction, $((m^*S^*, S^*, P), \sigma'^*)$ constitutes a valid SPS-EQ-$\mathcal{R}$ message-signature pair. It remains to show that for $M^* = (m^*S^*, S^*, P)$, the class $[M^*]_\mathcal{R}$ is different from all classes queried to $\mathcal{C}$, if m^* is different from all messages queried to the VESign oracle. Verify demands that the third vector component of M^* be P, which uniquely determines the representative for each class and allows for comparison. Now, if there is some $M_i = (m_iS_i, S_i, P)$ coinciding with M^* in the second component, then both vectors still differ in the first component for $m^* \neq m_i$. Likewise, if they coincide in the first component for $m^* \neq m_i$, then they cannot have equal second components. Hence, $M^* \neq M_i$ for any M_i queried to $\mathcal{C}$.

Type-2: In the following, let $\mathbf{p}$ be some fixed probability, which we will set later. $\mathcal{B}$ obtains an instance (P, aP) of the DHI problem in group $\mathbb{G}_1 \in \mathsf{BG}$ (and thereby implicitly the bilinear group BG) from the challenger $\mathcal{C}$. $\mathcal{B}$ executes $(\mathsf{sk}, \mathsf{pk}) \xleftarrow{\$} \mathsf{KeyGen}_\mathcal{R}(\mathsf{BG})$, runs $\mathcal{A}$ on $(\mathsf{pk}, \mathsf{apk} \leftarrow (\mathsf{BG}, A))$ for $A \leftarrow aP$, sets up a list $L \leftarrow \emptyset$ and simulates queries to the oracles as follows:

$\mathsf{VESign}(\cdot, \mathsf{sk}, \mathsf{apk})$**:** If $\mathcal{A}$ submits a query for $m \in \mathbb{Z}_p^*$, $\mathcal{B}$ picks $s \xleftarrow{\$} \mathbb{Z}_p^*$ and random coins r_2, sets

- $W \leftarrow sA$ with probability $\mathbf{p}$, or
- $W \leftarrow s(P + A)$ with probability $1 - \mathbf{p}$, and

runs $\omega' \leftarrow \mathsf{Sign}_\mathcal{R}((mW, W, A), \mathsf{sk}; r_2)$. Then, it sets $\omega \leftarrow (\omega', W)$, stores $L[m] \leftarrow (s, r_2, \omega)$ and returns ω.

$\mathsf{Resolve}(\cdot, \cdot, \mathsf{ask}, \mathsf{pk})$**:** If $\mathcal{A}$ submits a query for $m \in \mathbb{Z}_p^*$ and ω, then $\mathcal{B}$ checks whether $\mathsf{VEVerify}(m, \omega, \mathsf{pk}, \mathsf{apk}) \stackrel{?}{=} 1$ and returns $\perp$ if this is not the case. Otherwise, it retrieves the entry $(s, r_2, \omega = (\omega', W)) \leftarrow L[m]$. If $W \stackrel{?}{=} s(P + A)$, then $\mathcal{B}$ aborts. Otherwise, $\mathcal{B}$ computes $\sigma' \leftarrow \mathsf{Sign}_\mathcal{R}((msP, sP, P), \mathsf{sk}; r_2)$ and returns $\sigma \leftarrow (\sigma', sP)$.

If at some point $\mathcal{A}$ outputs a valid message-signature pair $(m^*, \sigma^* = (\sigma'^*, S^*))$ and has queried the VESign oracle for m^*, but not the $\mathsf{Resolve}$ oracle, then $\mathcal{B}$ retrieves $(s^*, r_2^*, \omega^*) \leftarrow L[m^*]$. If $S^* = s^*P$, then $\mathcal{B}$ aborts. Otherwise, we have $S^* = s^*(\frac{1}{a}P + P)$ and $\mathcal{B}$ outputs $\frac{1}{a}P \leftarrow \frac{1}{s^*}S^* - P$ as a solution to the DHI problem.

Note that the distribution of all values returned to $\mathcal{A}$ during the simulation is identical to the distribution of these values during a real game, which makes the simulation perfect.

Let q_R be the number of resolve queries. Then, with probability $\mathbf{p}^{q_R}$, $\mathcal{B}$ does not abort during the simulation. Given that the simulation works out, $\mathcal{A}$ outputs a "useful" forgery with probability $1-\mathbf{p}$. In total, $\mathcal{B}$ is able to return a solution to the DHI problem with probability $\mathbf{p}^{q_R}(1-\mathbf{p})$. The function $f(\mathbf{p}) = \mathbf{p}^{q_R}(1-\mathbf{p})$ reaches its maximum for $\frac{q_R}{q_R+1}$ and after few calculations we obtain $f(\mathbf{p}) = O(\frac{1}{q_R})$. Therefore, if $\mathcal{A}$ is able to break the opacity of the scheme with non-negligible probability $\epsilon(\kappa)$, then $\mathcal{B}$ is able to break the DHI assumption with non-negligible probability $O(\frac{\epsilon(\kappa)}{q_R})$. □

Theorem 7. *The VES in Scheme 1 is unconditionally extractable.*

Theorem 8. *The VES in Scheme 1 is abuse free given that the underlying SPS-EQ-$\mathcal{R}$ scheme is unforgeable.*

The following theorem states that Scheme 1 is resolution duplication. In particular, it is resolution independent, the importance of which was established in Section 3. It will allow also us to derive a PKE scheme (cf. Section 5).

Theorem 9. *The VES in Scheme 1 is resolution duplicate given that the underlying SPS-EQ-$\mathcal{R}$ scheme allows perfect composition.*

The proofs of Theorems 7-9 are given in Appendix A.

5 Public-Key Encryption From SPS-EQ-$\mathcal{R}$

In this section, we show how to convert any SPS-EQ-$\mathcal{R}$ satisfying perfect composition (Definition 10) into a public-key encryption scheme. This connection is somewhat surprising, as it is well known that regular signature schemes do not imply public-key encryption (in a black-box way). However, there is no contradiction as SPS-EQ-$\mathcal{R}$ have more structure than a regular signature scheme.

The basic idea is to instantiate the transformation of Calderon et al. [5]. This transformation turns any secure, resolution duplicate VES scheme into a public-key encryption scheme, in a black-box way. We have already shown how to construct a secure VES scheme, and that it is resolution duplicate, in Section 4. The basic idea of the transformation is an application of the Goldreich-Levin trick [14] to the setting of VES. That is, we view $\langle \sigma, r \rangle$ as the hard-core predicate for VESign, i.e., given ω and r it should be hard to predict the value of $\langle \sigma, r \rangle$. This intuition is formally shown in the following lemma.

Lemma 3. *Let* VES *be a VES and let* $b(x,r) := \langle x, r \rangle \mod 2$ *for any* x *and* r *such that* $|x| = |r|$*. Then, if the VES is opaque for all messages* $m \in \mathcal{M}$*, all* $(\mathsf{ask}, \mathsf{apk}) \xleftarrow{\$} \mathsf{AKeyGen}(1^\kappa)$ *and* $(\mathsf{sk}, \mathsf{pk}) \xleftarrow{\$} \mathsf{KeyGen}(1^\kappa)$*, it is hard to compute* $b(\sigma, r)$ *given* $m, \mathsf{apk}, \mathsf{pk}, \omega \xleftarrow{\$} \mathsf{VESign}(m, \mathsf{sk}, \mathsf{apk})$*, and* $r \xleftarrow{\$} \{0,1\}^{|\sigma|}$*, where* $\sigma := \mathsf{Resolve}(\omega, \mathsf{ask}, \mathsf{pk})$.

The proof is given in [5] and follows that of Goldreich [13] closely. It leads to the following construction of a CPA-secure public-key encryption scheme $(\mathsf{EKeyGen}, \mathsf{Enc}, \mathsf{Dec})$ as follows:

- $\mathsf{EKeyGen}(1^\kappa)$: Output $(\mathsf{apk}, \mathsf{ask}) \xleftarrow{\$} \mathsf{AKeyGen}(1^\kappa)$.
- $\mathsf{Enc}(m, \mathsf{apk})$: Generate signing keys $(\mathsf{sk}, \mathsf{pk}) \xleftarrow{\$} \mathsf{KeyGen}(1^\kappa)$ and pick a random tape r and $r_\sigma \xleftarrow{\$} \{0,1\}^{|\sigma|}$. Now, compute $\omega := \mathsf{VESign}(0, \mathsf{sk}, \mathsf{apk}; r)$, $\sigma \xleftarrow{\$} \mathsf{Extract}(m, \mathsf{sk}, r)$, and set $c_0 := m \oplus \langle \sigma, r_\sigma \rangle$. Output $c = (\mathsf{pk}, \omega, r_\sigma, c_0)$.
- $\mathsf{Dec}(c, \mathsf{ask})$: Parse $c = (\mathsf{pk}, \omega, r_\sigma, c_0)$ and return $\perp$ if $\mathsf{VEVerify}(0, \omega, \mathsf{pk}, \mathsf{apk}) = 0$. Otherwise, compute $\sigma := \mathsf{Resolve}(0, \mathsf{pk}, \omega, \mathsf{ask}, \mathsf{pk})$ and output $m := c_0 \oplus \langle \sigma, r_\sigma \rangle$.

Regarding security, it was shown that the above construction is CPA secure [5]:

Theorem 10. *If the verifiably encrypted signature is resolution duplicate (according to Definition 18) and opaque, then the above scheme is IND-CPA secure.*

6 Conclusion

We have shown that the property of resolution independence is crucial, not only for constructing public-key encryption from verifiably encrypted signatures, but even for the correctness and security of the VES.

We gave for the first time a construction of resolution duplicate (and in particular resolution independent) VES from SPS-EQ-$\mathcal{R}$. Our VES has short keys and signatures. This result demonstrates further applications of SPS, and SPS-EQ-$\mathcal{R}$ in particular. Using our VES, we constructed public-key encryption. Since the construction is generic, it proves that SPS-EQ-$\mathcal{R}$s allowing perfect composition cannot be constructed from one-way functions.

References

1. M. Abe, G. Fuchsbauer, J. Groth, K. Haralambiev, and M. Ohkubo. Structure-preserving signatures and commitments to group elements. In T. Rabin, editor, *CRYPTO 2010*, volume 6223 of *LNCS*, pages 209–236, Santa Barbara, CA, USA, Aug. 15–19, 2010. Springer, Berlin, Germany.
2. P. S. L. M. Barreto and M. Naehrig. Pairing-friendly elliptic curves of prime order. In B. Preneel and S. Tavares, editors, *SAC 2005*, volume 3897 of *LNCS*, pages 319–331, Kingston, Ontario, Canada, Aug. 11–12, 2005. Springer, Berlin, Germany.
3. D. Boneh, C. Gentry, B. Lynn, and H. Shacham. Aggregate and verifiably encrypted signatures from bilinear maps. In E. Biham, editor, *EUROCRYPT 2003*, volume 2656 of *LNCS*, pages 416–432, Warsaw, Poland, May 4–8, 2003. Springer, Berlin, Germany.
4. D. Boneh, P. A. Papakonstantinou, C. Rackoff, Y. Vahlis, and B. Waters. On the impossibility of basing identity based encryption on trapdoor permutations. In *49th FOCS*, pages 283–292, Philadelphia, Pennsylvania, USA, Oct. 25–28, 2008. IEEE Computer Society Press.

5. T. Calderon, S. Meiklejohn, H. Shacham, and B. Waters. Rethinking verifiably encrypted signatures: A gap in functionality and potential solutions. In J. Benaloh, editor, *CT-RSA 2014*, volume 8366 of *LNCS*, pages 349–366, San Francisco, CA, USA, Feb. 25–28, 2014. Springer, Berlin, Germany.
6. J. Camenisch and A. Lysyanskaya. Dynamic accumulators and application to efficient revocation of anonymous credentials. In M. Yung, editor, *CRYPTO 2002*, volume 2442 of *LNCS*, pages 61–76, Santa Barbara, CA, USA, Aug. 18–22, 2002. Springer, Berlin, Germany.
7. S. Chatterjee and A. Menezes. On cryptographic protocols employing asymmetric pairings – the role of ψ revisited. *Discrete Applied Mathematics*, 159(13):1311 – 1322, 2011. `http://www.sciencedirect.com/science/article/pii/S0166218X11001648`.
8. G. Fuchsbauer. Commuting signatures and verifiable encryption. In K. G. Paterson, editor, *EUROCRYPT 2011*, volume 6632 of *LNCS*, pages 224–245, Tallinn, Estonia, May 15–19, 2011. Springer, Berlin, Germany.
9. G. Fuchsbauer, C. Hanser, and D. Slamanig. EUF-CMA-secure structure-preserving signatures on equivalence classes. Cryptology ePrint Archive, Report 2014/944, 2014. `http://eprint.iacr.org/2014/944`.
10. G. Fuchsbauer, C. Hanser, and D. Slamanig. Practical round-optimal blind signatures in the standard model. In *Advances in Cryptology - CRYPTO 2015, 35th Annual International Cryptology Conference, Santa Barbara, California, USA, August 16-20, 2015, Proceedings*, 2015. to appear.
11. Y. Gertner, S. Kannan, T. Malkin, O. Reingold, and M. Viswanathan. The relationship between public key encryption and oblivious transfer. In *41st FOCS*, pages 325–335, Redondo Beach, California, USA, Nov. 12–14, 2000. IEEE Computer Society Press.
12. Y. Gertner, T. Malkin, and S. Myers. Towards a separation of semantic and CCA security for public key encryption. In S. P. Vadhan, editor, *TCC 2007*, volume 4392 of *LNCS*, pages 434–455, Amsterdam, The Netherlands, Feb. 21–24, 2007. Springer, Berlin, Germany.
13. O. Goldreich. *Foundations of Cryptography: Basic Tools*, volume 1. Cambridge University Press, Cambridge, UK, 2001.
14. O. Goldreich and L. A. Levin. A hard-core predicate for all one-way functions. In *21st ACM STOC*, pages 25–32, Seattle, Washington, USA, May 15–17, 1989. ACM Press.
15. J. Groth and A. Sahai. Efficient non-interactive proof systems for bilinear groups. In N. P. Smart, editor, *EUROCRYPT 2008*, volume 4965 of *LNCS*, pages 415–432, Istanbul, Turkey, Apr. 13–17, 2008. Springer, Berlin, Germany.
16. C. Hanser and D. Slamanig. Structure-preserving signatures on equivalence classes and their application to anonymous credentials. In P. Sarkar and T. Iwata, editors, *ASIACRYPT 2014, Part I*, volume 8873 of *LNCS*, pages 491–511, Kaoshiung, Taiwan, R.O.C., Dec. 7–11, 2014. Springer, Berlin, Germany.
17. J. Håstad, R. Impagliazzo, L. A. Levin, and M. Luby. A pseudorandom generator from any one-way function. *SIAM Journal on Computing*, 28(4):1364–1396, 1999.
18. F. Hess. On the security of the verifiably-encrypted signature scheme of boneh, gentry, lynn and shacham. *Inf. Process. Lett.*, 89(3):111–114, 2004.
19. R. Impagliazzo and M. Luby. One-way functions are essential for complexity based cryptography (extended abstract). In *30th FOCS*, pages 230–235, Research Triangle Park, North Carolina, Oct. 30 – Nov. 1, 1989. IEEE Computer Society Press.

20. R. Impagliazzo and S. Rudich. Limits on the provable consequences of one-way permutations. In *21st ACM STOC*, pages 44–61, Seattle, Washington, USA, May 15–17, 1989. ACM Press.
21. L. Lamport. Constructing digital signatures from a one-way function. Technical Report SRI-CSL-98, SRI International Computer Science Laboratory, Oct. 1979.
22. S. Lu, R. Ostrovsky, A. Sahai, H. Shacham, and B. Waters. Sequential aggregate signatures and multisignatures without random oracles. In S. Vaudenay, editor, *EUROCRYPT 2006*, volume 4004 of *LNCS*, pages 465–485, St. Petersburg, Russia, May 28 – June 1, 2006. Springer, Berlin, Germany.
23. B. Pfitzmann and A.-R. Sadeghi. Anonymous fingerprinting with direct nonrepudiation. In T. Okamoto, editor, *ASIACRYPT 2000*, volume 1976 of *LNCS*, pages 401–414, Kyoto, Japan, Dec. 3–7, 2000. Springer, Berlin, Germany.
24. J. Rompel. One-way functions are necessary and sufficient for secure signatures. In *22nd ACM STOC*, pages 387–394, Baltimore, Maryland, USA, May 14–16, 1990. ACM Press.
25. A. Rosen and G. Segev. Chosen-ciphertext security via correlated products. In O. Reingold, editor, *TCC 2009*, volume 5444 of *LNCS*, pages 419–436. Springer, Berlin, Germany, Mar. 15–17, 2009.
26. M. Rückert. Verifiably encrypted signatures from RSA without NIZKs. In B. K. Roy and N. Sendrier, editors, *INDOCRYPT 2009*, volume 5922 of *LNCS*, pages 363–377, New Delhi, India, Dec. 13–16, 2009. Springer, Berlin, Germany.
27. M. Rückert and D. Schröder. Security of verifiably encrypted signatures and a construction without random oracles. In H. Shacham and B. Waters, editors, *PAIRING 2009*, volume 5671 of *LNCS*, pages 17–34, Palo Alto, CA, USA, Aug. 12–14, 2009. Springer, Berlin, Germany.
28. Y. Vahlis. Two is a crowd? a black-box separation of one-wayness and security under correlated inputs. In D. Micciancio, editor, *TCC 2010*, volume 5978 of *LNCS*, pages 165–182, Zurich, Switzerland, Feb. 9–11, 2010. Springer, Berlin, Germany.
29. F. Zhang, R. Safavi-Naini, and W. Susilo. Efficient verifiably encrypted signature and partially blind signature from bilinear pairings. In T. Johansson and S. Maitra, editors, *INDOCRYPT 2003*, volume 2904 of *LNCS*, pages 191–204, New Delhi, India, Dec. 8–10, 2003. Springer, Berlin, Germany.

A Omitted proofs

Theorem 7. *The VES in Scheme 1 is unconditionally extractable.*

Proof. This follows directly from the correctness property of any SPS-EQ-$\mathcal{R}$ scheme. To see this, observe that for (m, ω) with $\omega = (\omega', sA)$ it holds that $\mathsf{VEVerify}_{\mathcal{R}}((msA, sA, A), \omega', \mathsf{pk}) = 1$ if and only if $\mathsf{Verify}_{\mathcal{R}}((msP, sP, P), \sigma', \mathsf{pk}) = 1$, where $((msP, sP, P), \sigma') \leftarrow \mathsf{ChgRep}_{\mathcal{R}}((msA, sA, A), \omega', \frac{1}{a}, \mathsf{pk}; 1)$, since

$$[(msA, sA, A)]_{\mathcal{R}} = [(msP, sP, P)]_{\mathcal{R}}.$$ □

Theorem 8. *The VES in Scheme 1 is abuse free given that the underlying SPS-EQ-$\mathcal{R}$ scheme is unforgeable.*

Proof. We assume that there is an efficient adversary $\mathcal{A}$ winning the abuse freeness game with non-negligible probability; then we are able to construct an

adversary $\mathcal{B}$ that uses $\mathcal{A}$ to break the EUF-CMA security of the underlying SPS-EQ-$\mathcal{R}$ scheme with non-negligible probability.

$\mathcal{B}$ obtains $\mathsf{pk}_{\mathcal{R}}$ of the SPS-EQ-$\mathcal{R}$ scheme with $\ell = 3$ (and thereby implicitly the bilinear group BG) from the challenger $\mathcal{C}$ of the EUF-CMA security game, sets $\mathsf{pk} \leftarrow \mathsf{pk}_{\mathcal{R}}$. Furthermore, $\mathcal{B}$ picks $a \xleftarrow{\$} \mathbb{Z}_p^*$, computes $A \leftarrow aP$ and sets $(\mathsf{ask}, \mathsf{apk}) = (a, (\mathsf{BG}, A))$. Next, $\mathcal{B}$ runs $\mathcal{A}$ on $(\mathsf{pk}, \mathsf{ask}, \mathsf{apk})$ and answers $\mathcal{A}$'s oracle queries as follows:

$\mathsf{VESign}(\cdot, \mathsf{sk}, \mathsf{apk})$**:** If $\mathcal{A}$ submits a query for $m \in \mathbb{Z}_p^*$, $\mathcal{B}$ queries $\mathcal{C}$'s signing oracle for the message $(m \cdot sA, sA, A)$ for $s \xleftarrow{\$} \mathbb{Z}_p^*$, gets in return a corresponding encrypted signature ω' and gives $\omega \leftarrow (\omega', sA)$ to $\mathcal{A}$.

If at some point $\mathcal{A}$ outputs a valid encrypted message-signature pair $(m^*, \omega^* = (\omega'^*, W^*))$, such that it has not previously queried m^* to any of the oracles, then $\mathcal{B}$ will output $((m^*W^*, W^*, A), \omega'^*)$ to $\mathcal{C}$. In case that $\mathcal{A}$ has queried the VESign oracle for m^*, $\mathcal{B}$ will abort.

Note that the distribution of all values returned to $\mathcal{A}$ during the simulation is identical to the distribution of these values during a real game.

By construction, $((m^*W^*, W^*, A), \omega'^*)$ constitutes a valid SPS-EQ-$\mathcal{R}$ message-signature pair. It remains to show that for $M^* = (m^*W^*, W^*, A)$, the class $[M^*]_{\mathcal{R}}$ is different from all classes queried to $\mathcal{C}$, if m^* is different from all messages queried to the VESign oracle. $\mathsf{VEVerify}$ demands that the third vector component of M^* be A, which uniquely determines the representative for each class and allows for comparison. Now, if there is some $M_i = (m_i \cdot W_i, W_i, A)$ coinciding with M^* in the second component, then both vectors still differ in the first component for $m^* \neq m_i$. Likewise, if they coincide in the first component for $m^* \neq m_i$, then they cannot have equal second components. Hence, assuming that $m^* \neq m_i$ and $M^* = M_i$ for some M_i queried to $\mathcal{C}$, immediately gives a contradiction. □

Theorem 9. *The VES in Scheme 1 is resolution duplicate given that the underlying SPS-EQ-$\mathcal{R}$ scheme allows perfect composition.*

Proof. Here, we have to show (1) that the outputs of $\mathsf{Sign}(m, \mathsf{sk})$ and $\mathsf{Resolve}(m, \mathsf{VESign}(m, \mathsf{sk}, \mathsf{apk}), \mathsf{ask}, \mathsf{pk})$ are distributed identically, (2) that $\mathsf{Resolve}$ is deterministic and (3) that there exists a PPT algorithm $\mathsf{Extract}(\cdot, \cdot, \cdot)$, such that for all $(\mathsf{ask}, \mathsf{apk}) \xleftarrow{\$} \mathsf{AKeyGen}(1^\kappa)$, $(\mathsf{sk}, \mathsf{pk}) \xleftarrow{\$} \mathsf{KeyGen}(1^\kappa), m \in \mathcal{M}$, and random tapes $r \in \{0,1\}^*$, it is the case that

$$\mathsf{Extract}(m, \mathsf{sk}, r) = \mathsf{Resolve}(m, \mathsf{VESign}(m, \mathsf{sk}, \mathsf{apk}; r), \mathsf{ask}, \mathsf{pk}).$$

Property (2) is easy to see, since $\mathsf{Resolve}$ controls the internal randomness of $\mathsf{ChgRep}_{\mathcal{R}}$, runs it with randomness 1 and, thereby, executes it deterministically. All other parts of $\mathsf{Resolve}$ are deterministic as well.

The extract algorithm for Property (3) can be specified as $\mathsf{Extract}(m, \mathsf{sk}, r) \coloneqq \mathsf{Sign}(m, \mathsf{sk}; r) = \mathsf{Sign}(m, \mathsf{sk}; (r_1, r_2)) = (\mathsf{Sign}_{\mathcal{R}}((msP, sP, P), \mathsf{sk}; r_2), sP)$ where s

is drawn uniformly from $\mathbb{Z}_p^*$ using random coins r_1. For the RHS, we have

$$\begin{aligned}
&\mathsf{Resolve}(m, \mathsf{VESign}(m, \mathsf{sk}, \mathsf{apk}; r_2), \mathsf{ask}, \mathsf{pk}) = \\
&\mathsf{Resolve}(m, (\mathsf{Sign}_{\mathcal{R}}((msA, sA, A), \mathsf{sk}; r_2), sA), \mathsf{ask}, \mathsf{pk}) = \\
&(\mathsf{ChgRep}_{\mathcal{R}}((msA, sA, A), \mathsf{Sign}_{\mathcal{R}}((msA, sA, A), \mathsf{sk}; r_2), \frac{1}{a}, \mathsf{pk}; 1)[2], sP),
\end{aligned}$$

where s and t are as above. If the underlying SPS-EQ-$\mathcal{R}$ scheme allows perfect composition, this gives the same output as the specified $\mathsf{Extract}$ algorithm.

With regard to (1) observe that Property (3) and the fact that the $\mathsf{Extract}$ algorithm can be expressed by Sign implies that the distributions of $\mathsf{Sign}(m, \mathsf{sk})$ and $\mathsf{Resolve}(m, \mathsf{VESign}(m, \mathsf{sk}, \mathsf{apk}), \mathsf{ask}, \mathsf{pk})$ are identical. □