Ann. Telecommun. (2014) 69:585–591
DOI 10.1007/s12243-013-0416-1

On the security of verifiably encrypted signature schemes in a multi-user setting

Kyung-Ah Shim

Received: 4 March 2012 / Accepted: 20 December 2013 / Published online: 12 January 2014

Abstract A verifiably encrypted signature provides a way to encrypt a signature under a third party's public key and proves that the resulting ciphertext contains such a signature. In this paper, we analyze the security of three verifiably encrypted signature schemes in a multi-user setting in which an adversary is allowed to access adjudication oracles for different users, but the same adjudicator.

Keywords Verifiably encrypted signature · Multi-user setting · Rogue-key attack · Bilinear pairing

1 Introduction

A verifiably encrypted signature provides a way to encrypt a signature under a third party's public key and proves that the resulting ciphertext contains such a signature. The signatures are used in applications such as online contract signing [1, 2]. Suppose Alice wants to show Bob that she has signed a message, but does not want Bob to possess her signature of that message. Alice can achieve this by encrypting her signature using the public key of a trusted third party, and sending this to Bob along with a proof that she has given him a valid encryption of her signature. Bob can verify that Alice has signed the message, but cannot deduce any information about her signature. Later in the protocol, if Alice is unable or unwilling to reveal her signature, Bob can ask the third party to reveal Alice's signature. Boneh et al. [6] proposed a verifiably encrypted signature (VES) scheme, based on BLS signatures [5], which is secure in the random oracle model. Gorantla and Saxena [10] proposed a VES scheme based on Boneh and Boyen's short signature scheme [4]. Lu et al. [12] presented the first VES scheme provably secure without random oracles under the Computational Diifie-Hellman (CDH) assumption. Zhang and Mao [18] proposed a more efficient VES scheme without random oracles and showed that the security of the scheme is based on the difficulty of solving the Chosen-Target-Inverse-CDH with Square problem. Recently, Shao [16] proposed a short VES scheme secure in the random oracle model.

Usually, security of a signature scheme is presented in a single-user setting, where there is only one target public key [9]. In the single-user setting, the strongest security notion is for adaptive attackers who are allowed to a'sk for a polynomial (in terms of the security parameter) number of signatures on messages of their choices under the target public key. The goal of the attacker is to produce a forgery on any message under the public key for which it has not already requested to the signing oracle. Galbraith et al. [8] addressed the security of signature schemes in a multi-user setting where a number of public keys, say n, are being used. Here, the goal of the adversary is, after a number of signature queries to a number of users, to output a forgery of a message from any single user. The attack model is that the adversary is given n signing oracles, one q_S queries to these oracles. At the end, the adversary should output one of the n public keys, a message m and a signature σ such that σ is a valid signature on the message m under the given public key. Clearly, m should not have been a query to the signing oracle corresponding to the public key, while m may have been a query to any of the other signing oracles. Menezes and Smart [13] also argued that the well-accepted security definition for signature schemes

K.-A. Shim (✉)
Division of Mathematical Modeling, National Institute for Mathematical Sciences, 305-390, KT Daedoek 2nd Research Center, 463-1 Jeonmin-dong, Yuseong-gu, Daejeon, Korea
e-mail: kashim@nims.re.kr

(existential unforgeability against adaptive chosen-message attacks) by Goldwasser et al. [9] is not adequate for a multi-user setting by demonstrating key substitution attacks on signature schemes secure in the single-user setting. The signature scheme in the multi-user setting require special care against rogue-key attacks, which can be mounted whenever adversaries are allowed to choose their public keys arbitrarily. Typical attacks have the adversary use a public key that is a function of an honest user's key, allowing him to produce forgeries easily. Although they might at first hearing sound far-fetched, rogue-key attacks are in fact possible to mount in practice and are a real threat. When, eventually, precise definitions [14] and proven secure schemes [3, 12, 14] emerged, they obviously paid a lot of attention to key setup. These schemes were, happily, proven secure against rogue-key attacks, but, unhappily, at the cost of complexity and expense in the scheme, or using unrealistic and burdensome assumptions on the public-key infrastructure (PKI) [7]. The security notions for VES schemes consist of unforgeability and opacity (or security against extraction). The opacity property is that no one except the adjudicator (or a signer) can extract an ordinary signature from a given VES. The security of all existing VES schemes is considered in the single-user setting. We analyze security of VES schemes in the multi-user setting, where an adversary is allowed to access adjudication oracles for different users, but the same adjudicator. In this setting, an adversary, who has its own public key related to Alice's public key, can extract Alice's signature from a signature which is obtained via adjudication of its own VES. We cannot say that the VES scheme with the weakness above is secure. In this paper, we show that three VES schemes [10, 16, 18] do not achieve the opacity property in the multi-user setting, i.e., an adversary can extract an ordinary signature of a target public key from a VES under another public key, where another public key is a rogue key produced from the target public key.

The rest of the paper is organized as follows. In Section 2, we describe some definitions and security notions for VES schemes. In Section 3, we describe three VES schemes. In Section 4, we show that the schemes is insecure against rogue-key attacks in the multi-user setting in which an adversary is allowed to access adjudication oracles for different users, but the same adjudicator. We then discuss security models for opacity and the registered key model in the multi-user setting. Concluding remarks are given in Section 5.

2 Preliminaries

2.1 Basic tools

We briefly review the necessary facts about bilinear maps and groups in the notation of [4, 5].

- $(G_1, *)$, $(G_2, *)$, and $(G_T, *)$ are three cyclic groups of a prime order p,
- g_1 is a generator of G_1 and g_2 is a generator of G_2,
- $e : G_1 \times G_2 \rightarrow G_T$ is a bilinear map, i.e., a map satisfying the following properties;
  - Bilinearity: $e(u^a, v^b) = e(u, v)^{ab}$ for all $u \in G_1$, $v \in G_2$ and $a, b \in Z$.
  - Non-degeneracy: $e(g_1, g_2) \neq 1$ and is thus a generator of G_T.

Formally, one defines a bilinear group generation algorithm $\mathcal{G}$ that takes as input a security parameter $k \in Z^+$ and outputs the description of groups G_1, G_2, G_T, and a bilinear map $e : G_1 \times G_2 \rightarrow G_T$. There exist probabilistic polynomial time algorithms (in k) for computing the group operations in G_1, G_2, G_T, and the bilinear map e.

2.2 Security notion of VES schemes

We first describe the definition of VES schemes [6].

Components of verifiably encrypted signature schemes. A VES scheme $\mathcal{VES}$ = (**KeyGen, Sign, Vfy, AdjKeyGen, VE-Sign, VE-Vfy, Adjudication**) is specified by seven polynomial time algorithms with the following functionality:

- **KeyGen.** The randomized key generation algorithm **KeyGen** takes as input a security parameter $k \in Z^+$ and returns a public/private key pair $(PK, SK) \leftarrow$ **KeyGen**(1^k) for a signer.
- **Sign**. The randomized signing algorithm **Sign** inputs a message m and a secret key SK. The output is a signature $\sigma \leftarrow$ **Sign**(SK, m).
- **Vfy**. The verification algorithm **Vfy** takes input a public key PK, a messages m and a signature σ and outputs `valid` if **Vfy**$(m, PK, \sigma) = 1$, or $\perp$ otherwise.
- **AdjKeyGen.** Generate a public/private key pair (APK, ASK) for an adjudicator.
- **VE-Sign.** Given a secret key SK, a message m, and an adjudicator's public key APK, compute a VES $\eta \leftarrow$ **VE** $-$ **Sign**(SK, APK, m).
- **VE-Vfy.** Given a public key PK, a message m, an adjudicator's public key APK, and a VES η, output `Valid` if **VE** $-$ **Vfy**$(PK, APK, m, \eta) = 1$, $\perp$ otherwise.
- **Adjudication.** Given an adjudicator's private key ASK, a public key PK, and a VES η on a message m,

extract and output an ordinary signature σ on m under PK.

Besides the security of ordinary signature in the signature component, Boneh et al. [6] specified two security properties of VES schemes: unforgeability and opacity. We describe only the opacity property in a single-user setting.

Opacity. An extractor $\mathcal{E}$'s advantage $Adv_{\mathcal{VES},\mathcal{E}}$ is defined as its probability of success in the following game between a challenger $\mathcal{C}$ and $\mathcal{E}$:

- **Setup.** The challenger runs **KeyGen** and **AdjKeyGen** algorithms and (PK, APK) is given to $\mathcal{E}$.
- **VE-Sign Queries.** Proceeding adaptively, $\mathcal{E}$ requests a VES on messages m_i for PK and APK, $\mathcal{C}$ returns a VES $\eta_i \leftarrow \mathbf{VE-Sign}(SK, APK, m_i)$.
- **Adjudication Queries.** When $\mathcal{E}$ requests adjudication on a VES η, $\mathcal{C}$ returns an ordinary signature $\sigma \leftarrow \mathbf{Adjudication}(ASK, \eta)$.
- **Output.** Eventually, $\mathcal{E}$ outputs an ordinary signature σ^* on a message m^* for PK and wins the game if (a) m^* is not requested to the **Adjudication** oracle under APK, and (b) $\mathbf{Vfy}(PK, m^*, \sigma^*) = \texttt{Valid}$.

Note that verifiably encrypted signature extraction is, thus, no more difficult than forgery in the underlying signature scheme.

Definition 1 An algorithm $\mathcal{E}(t, q_{VS}, q_A, \epsilon)$ extracts a VES if $\mathcal{E}$ runs in time at most t, makes at most q_{VS} queries to the verifiably encrypted signing oracle, at most q_A queries to the adjudication oracle, and $Adv_{\mathcal{VES},\mathcal{E}}$ is at least ϵ. A VES scheme is $(t, q_{VS}, q_A, \epsilon)$ secure against extraction if no algorithm $(t, q_{VS}, q_A, \epsilon)$ extracts it.

3 Review of three verifiably encrypted signature schemes

In this section, we describe three VES schemes proposed in [10, 16, 18].

3.1 Shao's VES scheme

Shao [16] proposed a short VES scheme based on the Boneh et al.'s short signature scheme [5]. They claimed that the scheme is strongly unforgeable in the random oracle model under a stronger security model, where two inside adversaries, malicious adjudicator and malicious verifier, have more power than ever. Shao's scheme runs as follows.

- **KeyGen.** Take as input 1^n, run a randomized algorithm to generate the system parameters $\langle q, G_1, G_2, e, P\rangle$ and two cryptographic hash functions $R : \{0,1\}^* \times G_1 \times G_1 \rightarrow \{0,1\}^n$, $H : \{0,1\}^* \times \{0,1\}^n \rightarrow G_1$. A signer picks up at random $x \in Z_p^*$ as his private key and computes his public key $Y = xP$.
- **Adjudicator key generation.** A adjudicator picks up at random $x_A \in Z_p^*$ as his private key and computes his public key $Y_A = x_A P$.
- **Sign.** Given a message $M \in \{0,1\}^*$ and the key pair $\{x, Y\}$ of the signer, first pick up at random r in $\{0,1\}^n$, and compute an ordinary signature (σ, r), where $\sigma = xH(M, r)$.
- **Verfy.** Each ordinary signature (σ, r) can be verified with respect to the public key Y of the signer by checking $e(\sigma, P) = e(H(M, r), Y)$.
- **VES-Sign.** Given the private key x of the signer, the public key Y_A of the adjudicator and a message $M \in \{0,1\}^*$, pick k at random in Z_p^* and compute $U = kP$, $r = R(M, Y_A, U)$, $\sigma = xH(M, r)$, and $W = \sigma + kY_A$. The verifiably encrypted signature is (U, W).
- **VES-Verify.** Given a verifiably encrypted signature (U, W) on the public key Y of the signer, the public key Y_A of the adjudicator and a message M, accept if

$$e(W, P) = e(H(M, R(M, Y_A, U)), Y) \cdot e(U, Y_A).$$

- **Adjudication.** Given the key pair $\{x_A, Y_A\}$ of the adjudicator, a certified public key Y, and a verifiably encrypted signature (U, W) on message M, first ensure that the verifiably encrypted signature is valid and then output the ordinary signature (σ, r), where $\sigma = W - xU$ and $r = R(M, Y_A, U)$.

3.2 Gorantla-Saxena's VES scheme

Gorantla and Saxena [10] proposed a VES scheme based on Boneh-Boyen's short signature scheme [4]. In this case, $G_1 = G_2 = G$. We review Gorantla-Saxena's VES scheme in the notation of [4].

- **KeyGen.** Select a random generator $g \in G$, and random integers $x, y \in Z_p$ and compute $u = g^x$, $v = g^y$ and $z = e(g, g)$. The public key is (g, u, v, z) and the secret key is (x, y).
- **Sign.** Given a secret key (x, y) and a message $m \in Z_p$, pick a random $r \in Z_p - \left\{-\frac{x+m}{y}\right\}$ and compute $\sigma = g^{\frac{1}{x+m+yr}} \in G$. The signature is (σ, r).
- **Vfy.** Given a public key (g, u, v, z), a message m, and a signature (σ, r), accept if $e(\sigma,\ u \cdot g^m \cdot v^r) = z$ holds; otherwise, reject.
- **VE-KeyGen.** Select a random generator $g' \in G$ and random integers $x', y' \in Z_p$ and compute $u = g^{x'}$, $v = g^{y'}$ and $z' = e(g', g')$. The public key is (g', u', v', z'), and the secret key is (x', y').

- **VE-Sign.** Given a secret key (x, y), a message $m \in Z_p$ and an adjudicator's public key (g', u', v', z'), select $r \in Z_p$ and compute $\omega = [u' \cdot (v')^r]^{\frac{1}{x+m+yr}}$. The verifiably encrypted signature is $\eta = (\omega, r)$.
- **VE-Vfy.** Given a public key (g, u, v, z), a message m, an adjudicator's public key (g', u', v', z') and a verifiably encrypted signature $\eta = (\omega, r)$, accept if $e(\omega,\ u \cdot g^m \cdot v^r) = e(u' \cdot (v')^r, g)$ holds; otherwise, reject.
- **Adjudication.** Given an adjudicator's public key (g', u', v') and corresponding secret key (x', y'), a certified public key (g, u, v, z) and a verifiably encrypted signature $\eta = (\omega,\ r)$ on some message m, ensure that the verifiably encrypted signature is valid; then compute $\sigma = \omega^{\frac{1}{x'+ry'}}$ and output (σ, r).

3.3 Zhang-Mao's VES scheme

Zhang and Mao [18] proposed a new VES scheme based on Waters' scheme [17], which is provably secure in the standard model under the difficulty of solving the Chosen-Target-Inverse-CDH with Square problem. Let $H_n : \{0, 1\}^* \to \{0, 1\}^n$ be a collision-resistant hash function for some $n \in Z^+$. For convenience, hashed messages m are represented as $(m_1, m_2, \cdots, m_n)$ with $m_i \in \{0, 1\}$ for all $i \in \{1, \cdots, n\}$.

- **Setup.** Output $< q, P, G_1, G_2, e >$ of a BDH parameter generator as well as integer n, random elements $P, U \in G_1$, and a random n-tuple $(U_1, \cdots, U_n) \in G_1^n$. Set a map $F : \{0, 1\}^n \to G_1$ with mapping string m on $F(m) = U' + \sum_{i=1}^{n} m_i U_i$. Finally, the public parameters is $\texttt{Params} = (n, q, G_1, G_2, P, P', U', U_1, \cdots, U_n)$.
- **KeyGen.** A signer, Alice, randomly chooses $a \in Z_p$ as her private key and computes the corresponding public key $P_A = aP$.
- **Sign.** Let $\mathrm{m} = H_n(M)$, for a message $M \in \{0, 1\}^*$, $\mathrm{m} = (m_1, m_2, \cdots, m_n)$ with $m_i \in \{0, 1\}$. Pick a random $r \in_R Z_p^*$ and compute a VES $\sigma = (\sigma_1, \sigma_2)$, where

$$\sigma_1 = aP' + r\left(U' + \sum_{i=1}^{n} m_i U_i\right), \quad \sigma_2 = rP.$$

- **Vfy.** Given a VES $\sigma = (\sigma_1, \sigma_2)$ on a message M, verify whether

$$e(\sigma_1, P) = e(P_A, P') \cdot e\left(\sigma_2, U' + \sum_{i=1}^{n} m_i U_i\right)$$

holds or not. If it holds accept it; otherwise, reject.
- **VE-KeyGen.** An adjudicator chooses $s_A \in Z_p^*$ as its private key and computes the corresponding public key $Q_A = s_A P$.
- **VE-Sign.** Given a signer, Alice's private key a and the adjudicator's public key Q_A, to sign the message M, Alice computes $\mathrm{m} = H_n(M)$ and randomly picks $r_A \in Z_p$ to compute the signature $\gamma = (\gamma_1, \gamma_2)$, where

$$\gamma_1 = aQ_A + r_A F(m),\ \gamma_2 = r_A P.$$

- **VE-Vfy.** Given a verifiably encrypted signature $\gamma = (\gamma_1, \gamma_2)$, accept if the following equation holds:

$$e(\gamma_1,\ P) = e(P_A,\ Q_A)e\left(\gamma_2,\ U' + \sum_{i=1}^{n} m_i U_i\right).$$

- **Adjudication.** Given a verifiably encrypted signature $\gamma = (\gamma_1, \gamma_2)$ on the message M, the adjudicator first verifies whether the verifiably encrypted signature γ is valid. If it is valid, compute $\gamma_1' = s_A^{-1}\gamma_1$ and $\gamma_2 = s_A^{-1}\gamma_2$ and output an ordinary signature $\gamma' = (\gamma_1', \gamma_2')$.

4 Rogue-key attacks on the three VES schemes

Multi-user signature schemes must be secure against rogue-key attacks in which an adversary can choose its public key(s) arbitrarily, previously considered in the contexts of aggregate signature and multisignature schemes [3, 6, 14, 15]. Recently, Hess [11] showed that Boneh et al.'s VES scheme is insecure against the rogue-key attacks in the multi-user setting in which an adversary is allowed to access adjudication oracles for different users, but the same adjudicator. In this section, we show that the three VES schemes described in the previous section are insecure against the rogue-key attacks in the multi-user setting.

- **Shao's scheme.** Let $Y = xP$ and $Y_A = x_A P$ be Alice's public key and an adjudicator's public key, respectively. First, Carol chooses $\alpha \in Z_p^*$ and issues $Y' = Y + \alpha \cdot P$ as its public key. Then, its secret key is $x + \alpha$. Note that Carol cannot know this secret key. Let $\tau = (U, W) = (kP, \sigma + kY_A)$ be Alice's VES on a message m under $\{Y, Y_A\}$. Suppose that Carol wants to extract an ordinary signature of Alice from τ.
  - Carol first computes $W' = W + \alpha H(M, r)$, where $r = R(M, Y_A, U)$ and then $\tau' = (U, W')$ is a valid VES of Carol on the same message m under $\{Y, Y_A\}$, as

$$\begin{aligned} W' &= W + \alpha H(M, r) \\ &= xH(M, r) + kY_A + \alpha H(M, r) \\ &= (x + \alpha)H(M, r) + kY_A. \end{aligned}$$

  - Next, Carol requests adjudication on τ' to the adjudicator and receives an ordinary signature $\sigma' = (x + \alpha)H(M, r)$ from τ' of η'.

– Finally, Carol computes $\sigma = \sigma' - \alpha H(M, r) = xH(M, r)$, and then σ is an ordinary signature of Alice.

Shao [16] claimed that the verification equation of Boneh et al.'s scheme [5] is $e(W, P) = e(H(M), Y)e(U, Y_A)$, so the addition of $R(M, Y_A, U)$ in his enhanced scheme makes the scheme probabilistic such that the malicious adjudicator cannot choose freely the rogue keys of users and the partial signature U. However, his method cannot prevent the rogue-key attacks in which a malicious user (not an adjudicator) can choose its public key(s) arbitrarily. To prevent the attacks, the user's public key Y must be included to the input of the hash function, i.e., $R(M, Y_A, Y, U)$.

- **Gorantla-Saxena's Scheme.** Let $PK_A = (g, u, v, z) = (g, g^x, g^y, e(g, g))$ and $APK = (g', u', v', z')$ be Alice's public key and an adjudicator's public key, respectively. First, Carol chooses $s, t \in Z_p^*$ and issues $PK_C = (g^t, u^{ts}, v^{ts}, z^{t^2})$ as its public key. Then, its secret key is (sx, sy). Let $\eta = (\omega, r) = ([u' \cdot (v')^r]^{\frac{1}{x+m+yr}}, r)$ be Alice's VES on $m \in Z_p$ under $\{PK_A, APK\}$. Suppose that Carol wants to extract an ordinary signature of Alice from η.
  - Carol first computes $\omega' = \omega^{\frac{1}{s}}$ and then $\eta' = (\omega', r)$ is a valid VES of Carol on the message $s \cdot m \in Z_p$ under $\{PK_C, APK\}$, as
    $$\omega' = ([u' \cdot (v')^r]^{\frac{1}{x+m+yr}})^{\frac{1}{s}} = [u' \cdot (v')^r]^{\frac{1}{sx+sm+syr}}.$$
  - Next, Carol requests adjudication on η' to the adjudicator and receives an ordinary signature (σ', r) of η', where $\sigma' = g^{\frac{1}{sx+sm+syr}}$.
  - Finally, Carol computes $\sigma = (\sigma')^s = g^{\frac{1}{x+m+yr}}$, and then (σ, r) is an ordinary signature of Alice's VES η.
- **Zhang-Mao's VES scheme.** Let $PK_A = P_A = aP$ and $APK = Q_A = s_A P$ be Alice's public key and an adjudicator's public key, respectively. First, Carol chooses $t \in Z_p$ and issues $PK_C = P_C = t \cdot P_A = taP$ as its public key. Then, its secret key is ta. Let $\gamma = (\gamma_1, \gamma_2) = \left(aP + r\left(U' + \sum_{i=1}^n m_i U_i\right), rP\right)$ be Alice's VES on $m \in Z_p$ under $\{PK_A, APK\}$. Suppose that Carol wants to extract an ordinary signature of Alice from γ.
  - Carol first computes
    $$\begin{aligned}\overline{\gamma} &= (\overline{\gamma_1}, \overline{\gamma_2}) \\ &= (t \cdot \gamma_1,\ t \cdot \gamma_2) \\ &= \left(taP + tr\left[U' + \sum_{i=1}^n m_i U_i\right],\ trP\right)\end{aligned}$$
    and then $\overline{\gamma}$ is a valid VES of Carol on the same message m under $\{PK_C, APK\}$.
  - Next, Carol requests adjudication on $\overline{\gamma}$ to the adjudicator and receives an ordinary signature
    $$\begin{aligned}\overline{\sigma} &= (\overline{\sigma_1}, \overline{\sigma_2}) = (s_A^{-1} \cdot \overline{\gamma_1},\ s_A^{-1} \cdot \overline{\gamma_2}) \\ &= \left(s_A^{-1}[taP + tr(U' + \sum_{i=1}^n m_i U_i)],\ s_A^{-1} \cdot trP\right)\end{aligned}$$
    of $\overline{\gamma}$.
  - Finally, Carol computes $\sigma = (t^{-1} \cdot \overline{\sigma_1},\ t^{-1} \cdot \overline{\sigma_2})$, and then σ is an ordinary signature of Alice's VES γ.

The results above show that the three VES scheme are insecure against rogue-key attacks in the multi-user setting, where which an adversary is allowed to access adjudication oracles for different users, but the same adjudicator. Also, these mean that the schemes do not achieve the opacity property in the multi-user setting.

There have been proposed countermeasures against the rogue-key attacks. Hess [11] proposed a modified version to prevent the attacks by adding a user public key together with a message being signed to the input of the hash function. But, such a countermeasure cannot work in the schemes without using hash functions. Another countermeasure is to require the adversary to prove knowledge on the discrete logarithms of his published public keys. In Boldyreva's multisignature scheme [3], it requires the proof of knowledge on secret keys during the public key registration. Micali et al. [14] also discussed a series of more sophisticated approaches based on zero-knowledge proofs, again with the effect that the adversary is constrained in his key selection. For these schemes, provable security has only been established under the knowledge on secret key (KOSK) assumption where the adversary is required to reveal the secret keys it utilizes. In practice, certifying authorities rarely require the strong proofs of knowledge on secret keys required to substantiate the KOSK assumption. Instead, proofs of possession (POP) on secret keys are required and can be as simple as just a signature over the certificate request message. These POP-based techniques can prevent the rogue-key attacks. Ristenpart and Yilek [15] suggested analyzing the security of multiparty signature schemes in a registered key model, which allows modeling a variety of key registration assumptions including those based on POP. Using the model, they proved the security of Boldyreva's and Lu et al.'s multisignature schemes [3, 12] against the rogue key attacks under POP protocols. Their registered key model based on POP can be applied to the VES setting.

The registered key model. Let $\mathcal{P}$ and $\mathcal{S}$ be sets and $\mathcal{K} \subset \mathcal{P} \times \mathcal{S}$ be a relation on the sets (representing public keys,

secret keys, and valid key pairs, respectively). A key registration protocol is a pair of interactive algorithms (RegP, RegV). A party registering a key runs RegP with inputs $pk \in \mathcal{P}$ and $sk \in \mathcal{S}$. A certifying authority (CA) runs RegV. We require that running RegP(pk, sk) with RegV results in RegV's final message being pk whenever $(pk, sk) \in \mathcal{K}$. To utilize the POP-based registered key model, we define the registration protocol $\mathsf{S} - \mathsf{Pop} = (\mathsf{PopP}, \mathsf{PopV})$ as follows. Let $\mathsf{S} = (\mathsf{Kg}, \mathsf{Sign}, \mathsf{Ver})$ be a signature scheme. Running PopP on inputs pk, sk results in sending the message $pk||\mathsf{Sign}(sk, < pk >)$ to the CA. Upon receiving message $pk||\sigma$, a CA running PopV replies with pk if $\mathsf{Ver}(pk, < pk >, \sigma) = 1$ and otherwise replies with $\perp$. We consider security definition that is captured by a game between an adversary and an environment. Adversaries are given an additional key registration oracle OKReg that, once invoked, runs a new instance of RegV for some key registration protocol (RegP, RegV). If the last message from RegV is a public key pk, then pk is added to a table $\mathcal{R}$. This table can now be used to modify winning conditions or restrict which public keys are utilized by the adversary in interactions with the environment. Security of schemes under the definition is therefore always with respect to some registration protocol. The key registration protocols mentioned so far are two round protocols: the registrant sends a first message to the CA, which replies with a second message being either pk or $\perp$. For any two round protocol $\mathsf{Reg} = (\mathsf{RegP}, \mathsf{RegV})$, the OKReg oracle can be simplified as follows. An adversary queries with a first message, at which point RegP is immediately run and supplied with the message. The oracle halts RegP before it sends its reply message. The message is added to $\mathcal{R}$ if it is not $\perp$. The oracle finally returns pk or $\perp$ appropriately.

We describe the opacity property in the multi-user setting by allowing VE-Sign oracles under several public keys $PK_1, \cdots, PK_n$. It means that the i-th signer cannot extract a signature of the j-th signer from its own signature obtained from adjudicate oracle under APK. This is formalized as follows:

Opacity. An extractor $\mathcal{E}$'s advantage $Adv_{\mathcal{VES},\mathcal{E}}$ is defined as its probability of success in the following game between a challenger $\mathcal{C}$ and $\mathcal{E}$:

- **Setup.** The challenger runs **KeyGen** and **AdjKeyGen** algorithms and (PK_i, APK) is given to $\mathcal{E}$, for $i = 1, \cdots, l$. The challenger generates other public keys, $PK_{l+1}, \cdots, PK_n$, which might be rogue keys.
- **VE-Sign Queries.** Proceeding adaptively, $\mathcal{E}$ requests a VES on messages m_i^j for PK_i and APK, $\mathcal{C}$ returns a VES $\eta_i^j \leftarrow \mathbf{VE-Sign}\left(SK_i, APK, m_i^j\right)$.
- **Adjudication Queries.** When $\mathcal{E}$ requests adjudication on a VES η_j^k on m_j^k for PK_j and APK for $j = 1, \cdots, n$, $\mathcal{C}$ returns an ordinary signature $\sigma_j \leftarrow$ **Adjudication** $\left(ASK, PK_j, \eta_j^k\right)$.
- **Output.** Eventually, $\mathcal{E}$ outputs an ordinary signature σ^* on a message m^* for PK_k and wins the game if (a) (m^*, PK_k) is not requested to the Adjudication oracle under APK and $k < l + 1$ and (b) $\mathbf{Vfy}(PK_k, m^*, \sigma^*) =$ `Valid`.

Definition 2 An algorithm $\mathcal{E}(t, q_{VS}, q_A, N, \epsilon)$ extracts a VES if $\mathcal{E}$ runs in time at most t, makes at most $q_{VS,N}$ queries to the verifiably encrypted signing oracle, at most q_A queries to the adjudication oracle, and $Adv_{\mathcal{VES},\mathcal{E}}$ is at least ϵ, where N is the maximal number of signers. A VES scheme is $(t, q_{VS}, q_A, N, \epsilon)$ secure against extraction in the N-user setting if no algorithm $(t, q_{VS}, q_A, N, \epsilon)$ extracts it.

5 Conclusion

We have shown that the three VES schemes in [10, 12, 16] are insecure against rogue-key attacks in a multi-user setting where an adversary is allowed to access adjudication oracles for different users, but the same adjudicator. These results demonstrate that the security of VES schemes involved in the third party including VES schemes must be analyzed in the multi-user setting.

References

1. Asokan N, Shoup V, Waidner M (2000) Optimistic fair exchange of digital signatures. IEEE J Sel Areas Comm 18(4):593–610
2. Bao F, Deng R, Mao W (1998) Efficient and practical fair exchange protocols with offline TTP. In: IEEE symposium on security and privacy, pp 77–85
3. Boldyreva A (2003) Efficient threshold signature, multisignature and blind signature schemes based on the Gap-Diffie-Hellman-group signature scheme. In: PKC'03, LNCS 2567. Springer, pp 31–46
4. Boneh D, Boyen X (2004) Short signatures without random oracles. In: Advances in Cryptology-Eurocrypt'04, LNCS 3027. Springer, pp 56–73
5. Boneh D, Lynn B, Shacham H (2002) Short signatures from the Weil pairing. In: Advances in Cryptology-Asiacrypt'01, LNCS 2248. Springer, pp 514–532
6. Boneh D, Gentry C, Lynn B, Shacham H (2003) Aggregate and verifiably encrypted signatures from bilinear maps. In: Advances in Cryptology-Eurocrypt'03, LNCS 2656. Springer, pp 416–432
7. Bellare M, Neven G (2006) Multi-signatures in the plain public-key model and a general forking lemma. In: ACM CCS
8. Galbraith S, Malone-Lee J, Smart NP (2002) Public key signatures. Inf Process Lett 83:263–266
9. Goldwasser S, Micali S, Rivest RL (1988) Digital signature scheme secure against adaptive chosen-message attacks. SIAM J Comput 17(2):281–308
10. Gorantla MC, Saxena A (2005) Verifiably encrypted signature scheme without random oracles. In: ICDCIT 2005, LNCS 3816. Springer, pp 357–363

11. Hess F (2004) On the security of the verifiably-encrypted signature scheme of Boneh, Gentry, Lynn and Shacham. Inf Process Lett 89(3):111–114
12. Lu S, Ostrovsky R, Sahai A, Shacham H, Waters B (2006) Sequential aggregate signatures and multisignatures without random oracles. In: Advances in Cryptology-Eurocrypt'06, LNCS 4004. Springer, pp 465–485
13. Menezes A, Smart N (2004) Security of signature schemes in a multi-user setting. Des Codes Cryptogr 33(3):261–274
14. Micali S, Ohta K, Reyzin L (2001) Accountable-subgroup multisignatures. In: Proc. of A CMCCS'01. ACM, pp 245–254
15. Ristenpart T, Yilek S (2007) The power of proofs-of-possession: securing multiparty signatures against rogue-key attacks. In: Eurocrypt 2007, LNCS 4515. Springer, pp 228–245
16. Shao Z (2012) Verifiably encrypted short signatures from bilinear maps. Ann Telecommun 67(9–10):437–445
17. Waters B (2005) Efficient identity-based encryption without random oracles. In: Advances in Cryptology-Eurocrypt'05, LNCS 3494. Springer, pp 114–127
18. Zhang J, Mao J (2007) A novel verifiably encrypted signature scheme without random oracle. In: ISPEC 2007, LNCS 4464, pp 65–78