Toward Efficient Certificateless Signcryption from (and without) Bilinear Pairings

Paulo S. L. M. Barreto[1]*, Alexandre Machado Deusajute[1], Eduardo de Souza Cruz[1], Geovandro C. F. Pereira[1], Rodrigo Rodrigues da Silva[1]

[1] Departamento de Engenharia de Computação e Sistemas Digitais,
Escola Politécnica, Universidade de São Paulo, Brazil.

{pbarreto,adeusajute}@larc.usp.br,
{eduardo.cruz,geovandro.pereira,rodrigo.silva1}@poli.usp.br

Abstract. *In this paper we describe how to construct an efficient certificateless signcryption scheme. Contrary to the usual paradigm of converting identity-based encryption and signature schemes into a combined certificateless protocol, we adopt the approach of extending a conventional signcryption method with a certificateless key validation mechanism, resorting to the underlying identity-based techniques, and as a consequence to pairings, exclusively to validate the associated public keys. The result is as efficient as the underlying signcryption method as long as the amortized cost of this validation is low, as is the case of our concrete proposal.*

1. Introduction

Conventional cryptosystems follow the paradigm that users choose their own private keys, compute the corresponding public keys and submit them to a certification authority, which verifies their identities and issues certificates linking those identities and public keys. This creates the need for digital certificate management in a certification infrastructure (also known as public-key infrastructure, or PKI) that may prove cumbersome to maintain.

Shamir introduced the notion of identity-based (IB) cryptography [Shamir 1984] in an attempt to mitigate the burden of a PKI. In an IB cryptosystem, private keys are not chosen by the users but rather issued by a trusted authority called the Key Generation Bureau (KGB) or Trust Authority (TA), and public keys are replaced by arbitrary strings representing users' identities, avoiding the need for certificates altogether. On the other hand, it has the drawback of implicitly establishing a key escrow mechanism, since the KGB has the ability to recover confidential information from any user.

The concept of certificateless (CL) cryptosystems [Al-Riyami and Paterson 2003] was introduced to address the key escrow issue while avoiding the use of certificates and the need for a public-key infrastructure. The usual principle behind a CL scheme is to partition private keys into two components: an identity-based partial key (known to the KGB and thus otherwise subject to escrow) and one conventional albeit non-certified partial key (unknown to the KGB). This technique potentially combines the best features of identity-based and certificate-based cryptography, and indeed a number of certificateless

*Supported by the Brazilian National Council for Scientific and Technological Development (CNPq) under research productivity grant 312005/2006-7 and universal grant 485317/2007-9.

encryption schemes derived from identity-based encryption algorithms have been successfully constructed and proven secure under certain assumptions. On the other hand, certificateless signatures remain much trickier to define and analyze, and as a consequence constructing certificateless signcryption schemes has been an elusive task only recently solved, though the only known such protocol can be hardly considered efficient. A signcryption scheme [Zheng 1997] is an integrated method to encrypt and sign a message in a more efficient way than to apply separately an encryption scheme and a signature scheme. The efficiency improvement may reside in the processing time, bandwidth occupation, key management, or any combination thereof; it may be simply a robust way to combine the two primitives so as to avoid deleterious interactions.

In this paper we follow a rather distinct approach to devise a certificateless signcryption method. Instead of combining an identity-based encryption method with an equally identity-based signature scheme and converting the result to a certificateless protocol, we extend a certificateless encryption method with a *conventional* signature scheme, but avoid the use of certificates for the latter component by using identity-based techniques to validate the public verification key.

The remainder of this paper is organized as follows. The underlying notions underlying the proposed technique are presented in section 2. We describe our proposed certificateless signcryption scheme in section 3. Formal assessment of the security implications of our proposal is carried out in section 4. Efficiency analysis and comparisons are performed in section 5. We conclude in section 6.

2. Preliminaries

Identity-based cryptography became feasible when instantiated with the help of bilinear maps, or *pairings* for short [Sakai et al. 2000, Boneh and Franklin 2001], and has since attracted widespread attention due to its unusual properties. Pairings are formally defined as follows. Let k be a security parameter and n be a k-bit prime number. Consider groups $\mathbb{G}_1$, $\mathbb{G}_2$ and $\mathbb{G}_T$ of order n. We say that $(\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T)$ are pairing groups if there exists a bilinear map (i.e. a pairing) $e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$ satisfying the following properties:

1. Bilinearity: $\forall (S, T) \in \mathbb{G}_1 \times \mathbb{G}_2$, $\forall a, b \in \mathbb{Z}_n$, $e(aS, bT) = e(S, T)^{ab}$.
2. Non-degeneracy: $\forall S \in \mathbb{G}_1$, $e(S, T) = 1$ for all $T \in \mathbb{G}_2$ iff $S = O_{\mathbb{G}_1}$.
3. Computability: $\forall (S, T) \in \mathbb{G}_1 \times \mathbb{G}_2$, $e(S, T)$ is efficiently computable.

For security analysis purposes, which in general follow the random oracle model [Bellare and Rogaway 1993], the following problems are considered computationally infeasible:

Definition 1. *The **Discrete Logarithm Problem** (DLP): given $P, \alpha P \in \mathbb{G}_1$ (or $Q, \alpha Q \in \mathbb{G}_2$, or $g, g^\alpha \in \mathbb{G}_T$), compute $\alpha \in \mathbb{Z}_n$.*

Definition 2. *The **Gap Diffie-Hellman Problem** (GDHP): given $P, \alpha P \in \mathbb{G}_1$, $Q, \beta Q \in \mathbb{G}_2$, compute $\alpha\beta P$ and/or $\alpha\beta Q$ with the help of the pairing on these groups.*

Definition 3. *The **Fixed Argument Pairing Inversion Problem** (FAPIP): given $P \in \mathbb{G}_1$ and $g \in \mathbb{G}_T$ such that $g = e(P, Q)$ for some $Q \in \mathbb{G}_2$, compute Q. Equivalently, given $P \in \mathbb{G}_1$, $Q \in \mathbb{G}_2$, and $y = e(P, Q)^\alpha \in \mathbb{G}_T$ for some $\alpha \in \mathbb{Z}_n$, compute αQ.*

Definition 4. *The **q-Strong Diffie-Hellman Problem** (q-SDHP): given the $(q + 2)$-tuple $\langle P, Q, \alpha Q, \alpha^2 Q, \ldots, \alpha^q Q \rangle \in \mathbb{G}_1 \times \mathbb{G}_2^{q+1}$, find a pair $\langle c, (c + \alpha)^{-1} P \rangle \in \mathbb{Z}_n \times \mathbb{G}_1$.*

Finally, the notation $x \xleftarrow{\$} V$ means that variable x is uniformly sampled at random from set V.

2.1. BLMQ identity-based signatures

We now review the BLMQ identity-based signature (IBS) scheme [Barreto et al. 2005]. It consists of the following algorithms:

- **Setup:** given a security parameter k, this algorithm chooses a k-bit prime number n, bilinear map groups $(\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T)$ of order n supporting an efficiently computable, non-degenerate pairing $e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$, generators $P \in \mathbb{G}_1$, $Q \in \mathbb{G}_2$ and hash functions $h_0 : \mathbb{G}_T \times \{0,1\}^* \to \mathbb{Z}_n^*$, $h_1 : \{0,1\}^* \to \mathbb{Z}_n^*$. A master key $s \xleftarrow{\$} \mathbb{Z}_n^*$ is also chosen, to which the public key $P_{pub} = sP \in \mathbb{G}_1$ is associated. The generator $g = e(P,Q) \in \mathbb{G}_T$ is also included among the public parameters which are $\texttt{params} = (k, n, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, P, Q, g, P_{pub}, e, h_0, h_1)$.
- **Private-Key-Extract:** takes as input entity A's identifier $\mathsf{ID}_A \in \{0,1\}^*$ and extracts A's identity-based private key $Q_A \leftarrow (h_1(\mathsf{ID}_A) + s)^{-1} Q \in \mathbb{G}_2$. Entity A can verify the consistency of this key by checking that $e(h_1(\mathsf{ID}_A)P + P_{pub}, Q_A) = g$. This setting is called the Sakai-Kasahara key style [Sakai and Kasahara 2003].
- **Sign:** to sign $m \in \{0,1\}^*$ under the private key P_A, the signer picks $u \xleftarrow{\$} \mathbb{Z}_n^*$ and computes
  1. $r \leftarrow g^u$
  2. $h \leftarrow h_0(r, m)$
  3. $S \leftarrow (u - h)Q_A$

  The signed message is the triple $(m, h, S) \in \{0,1\}^* \times \mathbb{Z}_n^* \times \mathbb{G}_2$.
- **Verify:** given an identity ID_A, upon reception of (m, h, S) the verifier computes
  1. $r \leftarrow e(h_1(\mathsf{ID}_A)P + P_{pub})g^h$
  2. $v \leftarrow h_0(r, m)$

  The verifier accepts the signed message iff $v = h$.

This scheme can be shown to be existentially unforgeable under adaptively chosen message attacks (EUF-IBS-CMA for short) in the random oracle model under the q-SDHP assumption [Barreto et al. 2005, section 3.1]. Notice that in this description we choose to define $P_{pub} \in \mathbb{G}_1$, $Q_A \in \mathbb{G}_2$ to avoid $\mathbb{G}_2$ arithmetic during verification, but an analogous description with $Q_{pub} \in \mathbb{G}_2$, $P_A \in \mathbb{G}_1$ and signed messages in $\{0,1\}^* \times \mathbb{Z}_n^* \times \mathbb{G}_1$ would be equally secure, while keeping the signature as short as possible in practice.

2.2. Schnorr signatures

We now briefly review Schnorr signatures [Schnorr 1991a], which, in spite of being entirely conventional, can be successfully combined with a CL encryption scheme to yield an efficient CL signcryption protocol. The Schnorr signature method consists of the following algorithms:

- **Setup:** given a security parameter k, this algorithm chooses a k-bit prime number n, a group $\mathbb{G}_T$ of order n, a generator $g \in \mathbb{G}_T$ and a hash function $h_0 : \mathbb{G}_T \times \{0,1\}^* \to \mathbb{Z}_n^*$. The public parameters are $\texttt{params} = (k, n, \mathbb{G}_T, g, h_0)$.
- **Set-Private-Key:** given params, this algorithm picks $x_A \xleftarrow{\$} \mathbb{Z}_n^*$ to be entity A's secret key.

- **Set-Public-Key:** given entity A's private key $x_A \in \mathbb{Z}_n^*$, compute $y_A \leftarrow g^{x_A} \in \mathbb{G}_T$ as A's public key.
- **Sign:** to sign $m \in \{0,1\}^*$ under the private key $x_A \in \mathbb{Z}_n^*$, the signer picks $u \xleftarrow{\$} \mathbb{Z}_n^*$ and computes
  1. $r \leftarrow g^u$
  2. $h \leftarrow h_0(r, m)$
  3. $z \leftarrow u - x_A h$

  The signed message is the triple (m, h, z).
- **Verify:** given a certified public key y_A, upon reception of (m, h, z) the verifier computes
  1. $r \leftarrow g^z y_A^h$
  2. $v \leftarrow h_0(r, m)$

  The verifier accepts the signed message iff $v = h$.

This scheme can be shown to be existentially secure against adaptively chosen message attacks in the random oracle model under the assumption that computing discrete logarithms in $\mathbb{G}_T$ is computationally infeasible [Pointcheval and Stern 1996, section 6]. Schnorr signatures were formerly covered by the US patent #4995082, which expired in February 2008 [Schnorr 1991b].

2.3. Zheng signcryption

The Zheng signcryption protocol [Zheng 1997] consists of the following algorithms:

- **Setup:** given a security parameter k, this algorithm chooses a k-bit prime number n, a group $\mathbb{G}_T$ of order n, a generator $g \in \mathbb{G}_T$, and hash functions $h_2 : \mathbb{G}_T \to \{0,1\}^*$, $h_3 : \mathbb{G}_T \times \{0,1\}^* \times \mathbb{G}_T^2 \to \mathbb{Z}_n^*$. The public parameters which are `params` $= (k, n, \mathbb{G}_T, g, h_2, h_3)$.
- **Set-Private-Key:** given `params`, this algorithm picks $x_A \xleftarrow{\$} \mathbb{Z}_n^*$ to be entity A's secret key.
- **Set-Public-Key:** given entity A's private key $x_A \in \mathbb{Z}_n^*$, compute $y_A \leftarrow g^{x_A} \in \mathbb{G}_T$ as A's public key.
- **Signcrypt:** to encrypt $m \in \{0,1\}^*$ under the receiver's public key $y_B \in \mathbb{G}_T$ and the sender's private key $x_A \in \mathbb{Z}_n^*$ and public key $y_A \in \mathbb{G}_T$, the sender picks $u \xleftarrow{\$} \mathbb{Z}_n^*$ and computes
  1. $r \leftarrow y_B^u$
  2. $c \leftarrow h_2(r) \oplus m$
  3. $h \leftarrow h_3(r, m, y_A, y_B)$
  4. $z \leftarrow u/(h + x_A)$

  The signcryptogram is the triple $(c, h, z) \in \{0,1\}^* \times \mathbb{Z}_n^2$.
- **Unsigncrypt:** given the sender's public key $y_A \in \mathbb{G}_T$ and the receiver's private key $x_B \in \mathbb{Z}_n^*$ and public key $y_B \in \mathbb{G}_T$, upon reception of the triple (c, h, z) the receiver checks that $h, z \in \mathbb{Z}_n^*$ and computes
  1. $r \leftarrow y_A^{x_B z} y_B^{hz}$
  2. $m \leftarrow h_2(r) \oplus c$
  3. $v \leftarrow h_3(r, m, y_A, y_B)$

  The receiver accepts the message iff $v = r$.

This scheme can be shown to sport the property of ciphertext indistiguishability under chosen-ciphertext attacks by flexible unsigncryption oracles (i.e. it is FUO-IND-CCA2-secure for short) in the random oracle model under the GDHP assumption. It is also existentially unforgeable against adaptive chosen message attacks (EUF-ACM-secure for short) under the DLP assumption [Baek et al. 2002].

3. The proposed certificateless signcryption scheme

We now show how to integrate BLMQ identity-based signatures, Schnorr signatures, and Zheng signcryption into a certificateless signcryption scheme. Following Al-Riyami and Paterson's original CL-PKC model [Al-Riyami and Paterson 2003], we do this by allowing users to choose their conventional but uncertified key pairs independently of their identity-based keys and also independently of their very identities. These keys are validated afterwards via the identity-based mechanism.

Our proposed certificateless signcryption protocol consists of the following algorithms:

- **Setup:** given a security parameter k, this algorithm chooses a k-bit prime number n, bilinear map groups $(\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T)$ of order n supporting an efficiently computable, non-degenerate pairing $e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$, generators $P \in \mathbb{G}_1$, $Q \in \mathbb{G}_2$ and hash functions $h_0 : \mathbb{G}_T^2 \times \{0,1\}^* \to \mathbb{Z}_n^*$, $h_1 : \mathbb{G}_T \times \{0,1\}^* \to \mathbb{Z}_n^*$, $h_2 : \mathbb{G}_T \to \{0,1\}^*$, $h_3 : (\mathbb{G}_T \times \{0,1\}^*)^3 \to \mathbb{Z}_n^*$. A master key $s \xleftarrow{\$} \mathbb{Z}_n^*$ is also chosen, to which the public key $P_{pub} = sP \in \mathbb{G}_1$ is associated. The generator $g \leftarrow e(P,Q) \in \mathbb{G}_T$ is also included among the public parameters which are $\texttt{params} = (k, n, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e, P, Q, g, P_{pub}, h_0, h_1, h_2, h_3)$.
- **Set-Secret-Value:** given params, this algorithm picks $x_A \xleftarrow{\$} \mathbb{Z}_n^*$ to be entity A's secret value.
- **Set-Public-Value:** given entity A's secret value $x_A \in \mathbb{Z}_n^*$, compute $y_A \leftarrow g^{x_A} \in \mathbb{G}_T$ as A's public value.
- **Private-Key-Extract:** given entity A's identifier $\mathsf{ID}_A \in \{0,1\}^*$ and public value $y_A \in \mathbb{G}_T$, compute A's identity-based private key $Q_A \leftarrow (h_1(y_A, \mathsf{ID}_A) + s)^{-1} Q \in \mathbb{G}_2$. Entity A can verify the consistency of this key by checking that $e(h_1(y_A, \mathsf{ID}_A)P + P_{pub}, Q_A) = g$.
- **Set-Private-Key:** given entity A's partial private key $Q_A \in \mathbb{G}_2$ and secret value $x_A \in \mathbb{Z}_n^*$, this algorithm sets the pair $(x_A, Q_A) \in \mathbb{Z}_n^* \times \mathbb{G}_2$ as entity A's complete private key pair.
- **Set-Public-Key:** given entity A's partial private key $Q_A \in \mathbb{G}_2$, secret value $x_A \in \mathbb{Z}_n^*$, and the corresponding public value $y_A \in \mathbb{G}_T$, the signer picks $u_A \xleftarrow{\$} \mathbb{Z}_n^*$ and computes
  1. $r_A \leftarrow g^{u_A}$
  2. $h_A \leftarrow h_0(r_A, y_A, \mathsf{ID}_A)$
  3. $T_A \leftarrow (u_A - x_A h_A) Q_A$

  Entity A's complete public key is the triple $(y_A, h_A, T_A) \in \mathbb{G}_T \times \mathbb{Z}_n^* \times \mathbb{G}_2$.
  This setting is a combination of a Schnorr signature (under key x_A) with a BLMQ signature (under key Q_A) on the public value y_A and the identity ID_A. The hard problem underlying the forgery of such a signature is the FAPIP rather than the DLP as it is for Schnorr; in other words, it consists of computing $Q'_A := x_A Q_A$

given $y_A := g^{x_A}$ and $P'_A := h_1(y_A, \mathsf{ID}_A)P + P_{pub}$. Notice that it is easy to compute y_A given Q'_A and P'_A, since it amounts to computing $e(P'_A, Q'_A)$ (and checking that $P'_A = h_1(e(P'_A, Q'_A), \mathsf{ID}_A)P + P_{pub}$).

- **Public-Key-Validate:** given entity A's complete public key (y_A, h_A, T_A), this algorithm checks that y_A has order n (i.e. $y_A \neq 1$ but $y_A^n = 1$) and computes
  1. $r_A \leftarrow e(h_1(y_A, \mathsf{ID}_A)P + P_{pub}, T_A)y_A^{h_A}$
  2. $v_A \leftarrow h_0(r_A, y_A, \mathsf{ID}_A)$

  The verifier accepts the public key iff $v_A = h_A$. The validation process combines the verification of a Schnorr signature with that of a BLMQ signature.
- **Signcrypt:** to encrypt $m \in \{0,1\}^*$ under the receiver's public key $y_B \in \mathbb{G}_T$ previously validated for identity ID_B and P_{pub}, and the sender's private key $x_A \in \mathbb{Z}_n^*$, public key $y_A \in \mathbb{G}_T$ and identity ID_A, the sender picks $u \xleftarrow{\$} \mathbb{Z}_n^*$ and computes
  1. $r \leftarrow y_B^u$
  2. $c \leftarrow h_2(r) \oplus m$
  3. $h \leftarrow h_3(r, m, y_A, \mathsf{ID}_A, y_B, \mathsf{ID}_B)$
  4. $z \leftarrow u/(h + x_A)$

  The signcryptogram is the triple $(c, h, z) \in \{0,1\}^* \times \mathbb{Z}_n^2$. Compared to Zheng's original scheme, the identities of both the sender and the receiver are included in the authentication equation 3.
- **Unsigncrypt:** given the sender's public key $y_A \in \mathbb{G}_T$ previously validated for identity ID_A and P_{pub}, and the receiver's private key $x_B \in \mathbb{Z}_n^*$, public key $y_B \in \mathbb{G}_T$ and identity ID_B, upon reception of the triple (c, h, z) the receiver checks that $h, z \in \mathbb{Z}_n^*$ and computes
  1. $r \leftarrow y_A^{x_B z} y_B^{hz}$
  2. $m \leftarrow h_2(r) \oplus c$
  3. $v \leftarrow h_3(r, m, y_A, \mathsf{ID}_A, y_B, \mathsf{ID}_B)$

  The receiver accepts the message iff $v = h$.

Zheng's scheme, and as a consequence also the above certificateless scheme, have the drawback that signatures are untransferable in the sense that the recipient cannot convince third parties that the sender really signed the message, since the verification equation depends on the recipient's private key. However, the key validation mechanism (algorithms **Set-Public-Key** and **Public-Key-Validate**) can be used with other conventional signcryption protocols (like Libert's SEG-signcryption scheme [Libert 2006] or the Libert-Quisquater q-DH-signcryption scheme [Libert and Quisquater 2006, section 5]) which address and remove this restriction.

4. Security analysis

Our proposed scheme would reduce to Zheng's signcryption as long as the public values y_A and y_B were conventional certified public keys, in which case the security analysis in [Baek et al. 2002] would hold almost unchanged. Specifically, the only difference would be the presence of y_A and y_B and the corresponding identities in the computation of h, which thwarts key replacement denial-of-decryption attacks (whereby an adversary would be able to replace a receiver's key queried by a sender in such a way that the receiver would later be able to verify a signcryptogram yet unable to decrypt it).

That same security proof continues to hold as long as those public values can be shown to be legitimate. Intuitively, the key validation process binds ID_A and y_A in

three ways. First, it shows that the signer knows the identity-based private key Q_A (a feature inherited from BLMQ); second, it shows that the signer knows the secret value x_A (a feature inherited from Schnorr signatures) in addition to Q_A; and third, it prevents dishonest users from generating more than one conventional key pair, since otherwise they could claim that the KGB is masquerading a legitimate user A by faking another key pair (x'_A, y'_A) that would be successfully validated. We now formalize this intuition, by showing that an adversary capable of forging key validation signatures is also able to forge plain BLMQ signatures and solve the FAPIP.

Consider the signature scheme underlying the key validation mechanism, i.e.

1. $r_A \leftarrow g^{u_A}$
2. $h_A \leftarrow h_0(r_A, y_A, \mathsf{ID}_A)$
3. $T_A \leftarrow (u_A - x_A h_A) Q_A$

where $g = e(P, Q)$ and $y_A = g^{x_A}$, with verification via

1. $r_A \leftarrow e(P_A, T_A) y_A^{h_A}$
2. $v_A \leftarrow h_0(r_A, y_A, \mathsf{ID}_A)$

where $P_A = h_1(y_A, \mathsf{ID}_A)P + P_{pub}$ and hence $e(P_A, Q_A) = g$. The following theorem holds regarding this scheme:

Theorem 1. *An adversary who can forge a signature for a fixed private value can also existentially forge a BLMQ signature.*

Proof. If the adversary $\mathcal{A}$ can forge (h_A, T_A) for a fixed private value $x_A \in \mathbb{Z}_n^*$, then $\mathcal{A}$ can trivially produce the BLMQ signature $(h_A, \hat{T}_A := x_A^{-1} T_A)$ for generators $\hat{P} := x_A P \in \mathbb{G}_1$ and $\hat{g} := e(\hat{P}, Q) = g^{x_A} = y_A \in \mathbb{G}_T$:

1. $\hat{u}_A \leftarrow u_A / x_A$
2. $r_A \leftarrow g^{u_A} = y_A^{u_A / x_A} = \hat{g}^{\hat{u}_A}$
3. $h_A \leftarrow h_0(r_A, y_A, \mathsf{ID}_A)$
4. $\hat{T}_A \leftarrow x_A^{-1} T_A = x_A^{-1}(u_A - x_A h_A) Q_A = (u_A / x_A - h_A) Q_A = (\hat{u}_A - h_A) Q_A$

□

Conversely, the following theorem holds:

Theorem 2. *A polynomial-time adversary that can forge a signature for a fixed identity with non-negligible probability can also solve the FAPIP with non-negligible probability.*

Proof. Suppose that a polynomial time adversary $\mathcal{A}$ can forge a signature (r_A, h_A, T_A) for the "message" $m_A := (y_A, \mathsf{ID}_A)$ with non-negligible probability, where h_A depends only on m_A and r_A. The Forking Lemma [Pointcheval and Stern 1996, lemma 1] then ensures that $\mathcal{A}$ can obtain a *pair* of signatures (r_A, h_A, T_A) and (r_A, h'_A, T'_A) with $h_A \neq h'_A$ with non-negligible probability, still in polynomial time. From the verification equation $r_A = e(P_A, T_A) g^{x_A h_A} = e(P_A, T_A) e(P_A, x_A h_A Q_A) = e(P_A, T_A + x_A h_A Q_A)$ and $r_A = e(P_A, T'_A) g^{x_A h'_A} = e(P_A, T'_A) e(P_A, x_A h'_A Q_A) = e(P_A, T'_A + x_A h'_A Q_A)$, whence $e(P_A, T_A + x_A h_A Q_A) = e(P_A, T'_A + x_A h'_A Q_A)$ and thus $e(P_A, T_A + x_A h_A Q_A) / e(P_A, T'_A + x_A h'_A Q_A) = e(P_A, T_A - T'_A + (h_A - h'_A) x_A Q_A) = 1$. Since the pairing is non-degenerate and hence

$e(P_A, R) \neq 1$ for any $R \in \mathbb{G}_2^*$, necessarily $T_A - T'_A + (h_A - h'_A)x_A Q_A = O$ and hence $x_A Q_A = (h'_A - h_A)^{-1}(T_A - T'_A)$. In other words, given $P_A \in \mathbb{G}_1$ and $y_A \in \mathbb{G}_T$, adversary $\mathcal{A}$ can solve the FAPIP and compute $Z_A \leftarrow (h'_A - h_A)^{-1}(T_A - T'_A) \in \mathbb{G}_2$ such that $e(P_A, Z_A) = y_A$ with non-negligible probability. Notice that knowledge of Q_A is not required once $\mathcal{A}$ has obtained the pair of forking signatures. □

We point out that that the KGB could still masquerade as entity A by generating a fake key pair $(\tilde{x}_A, \tilde{y}_A)$ unbeknownst to A and signing it under A's private key Q_A. However, this feature is shared by *all* conventional signature schemes, in the sense that a certification authority is always able to generate a fake key pair, sign a certificate for it, and impersonate any user at will. However, such unfair behavior is detectable and traceable, and can thus be deterred by legal means.

5. Efficiency

Pairing computation is in general the most expensive operation in a pairing-based cryptosystem, hence it generally pays to minimize their number. On the other hand, discarding too many pairings may mean giving up some functionality, which is as undesirable as high cost. Not surprisingly, a balance between these two constraints is likely to produce a scheme that does not need to sacrifice its functionality nor its efficiency.

We compare the costs of our method with the Barbosa-Farshim certificateless signcryption scheme based on pairings [Barbosa and Farshim 2008], the BLMQ identity-based signcryption scheme, the LXH self-certified signcryption scheme [Li et al. 2007] and the CLPKE scheme [Baek et al. 2005] certificateless encryption-only scheme without pairings on the upper lines of tables 1, 2 and 3 in terms of operations counts. CLPKE does not support a signature feature and is included in the comparison (defined on the same group $\mathbb{G}_T$ as our method) because of its explicit avoidance of pairings; BLMQ is not certificateless but constitutes one of the most efficient identity-based signcryption protocols known. Operations that only depend of keys or identities can be precomputed so as not to impair on the cost per message. The observed efficiency if of course mostly inherited from Zheng signcryption (in fact, slightly faster because of its Schnorr signature style), which is however a certificate-based rather than certificateless scheme and hence has the burden of a PKI.

As an illustration, for 256-bit BN curves [Barreto and Naehrig 2006] implementing optimal pairings [Vercauteren 2008] and the GLV technique for scalar multiplication and exponentiation [Galbraith et al. 2008], our method not surprisingly takes about twice as long as Barbosa-Farshim at key validation (which, on the other hand, has to be done only once for each key), while signcryption is 2.5 times faster than Barbosa-Farshim and unsigncryption is 7 times faster. In this setting our method is about twice as fast as BLMQ for signcryption and over 4 times faster for unsigncryption. For 256-bit MNT curves [Miyaji et al. 2001] with embedding degree $k = 4$ and the Ate pairing [Hess et al. 2006], the balance in favor of our method is more pronounced. While key validation still takes about 50% longer than Barbosa-Farshim for key validation (although only 70% of the time needed by LXH), signcryption is almost 15 times faster than Barbosa-Farshim, 8 times faster than BLMQ and 11 times faster than LXH, while unsigncryption is 38 times faster than either Barbosa-Farshim or LXH, and 19 times faster than BLMQ. Regardless of the curves chosen, our method is 3 times faster than CLPKE

Table 1. Key validation/preprocessing efficiency

	Barbosa-Farshim	BLMQ	LXH	CLPKE	ours
pairings	1	0	2	NA	1
exponentiations	0	0	0	1	2
scalar multiplications	0	1	0	0	1
time (ms, BN-256)	97.0	11.5	197.8	41.7	195.5
time (ms, MNT4-256)	65.5	15.7	133.4	5.4	93.5

Table 2. Signcryption efficiency

	Barbosa-Farshim	BLMQ	LXH	CLPKE	ours
exponentiations	1	1	2	3	1
scalar multiplications	$3 + \varepsilon^{\dagger}$	2	3	0	0
time (ms, BN-256)	104.8	76.1	122.3	124.3	41.2
time (ms, MNT4-256)	77.6	44.3	57.8	16.0	5.3

† 4 scalar multiplications, but two of them are simultaneous.

for signcryption and twice as fast for unsigncryption. The lower lines of tables 1, 2 and 3 contain experimental results obtained from Java implementations running on an AMD Turion™64 X2 platform at 2.3 GHz.

The bandwidth overhead of Barbosa-Farshim is 1 $\mathbb{G}_1$ element and 1 $\mathbb{G}_2$ element, while the overhead of our method is 2 $\mathbb{Z}_n$ elements. In practice this means that the Barbosa-Farshim bandwidth overhead only matches our method for pairings on supersingular elliptic curves; for all other pairing-friendly curves it occupies more space, even on the rare ordinary curves that admit twists of order equal to the embedding degree (which is thus limited to $k = 2, 4, 6$) since in such cases each element from $\mathbb{G}_1$ (or $\mathbb{G}_2$) already takes about twice the size of a $\mathbb{Z}_n$ element [Freeman et al. 2006]. For instance, Barbosa-Farshim takes 50% more bandwidth for curves with twists over quadratic extensions of the base field like BN curves, and three times as much as our method when instantiated over Freeman curves. The bandwidth occupation of our method and CLPKE are approximately the same; CLPKE attaches only one group element to the ciphertext, but it also needs a Fujisaki-Okamoto nonce [Fujisaki and Okamoto 1999] of comparable size.

Clearly the heaviest stage of our proposal is key validation. While its amortized cost per message is low, it would be desirable to have a cheaper alternative method for this process. A different key validation method would also be necessary to enable the core operation (i.e. signcryption/unsigncryption) of our proposal to be implemented on $\mathbb{G}_1$ instead of $\mathbb{G}_T$. We leave this problem for further research on the subject.

6. Conclusion

We have presented an efficient certificateless signcryption scheme based on the BLMQ identity-based signature, the Schnorr conventional signature, and the Zheng signcryption protocol. Our proposal limits pairing-based operations to key validation, thus inheriting the efficiency of its underlying components while introducing only a mild overhead that only applies once to each key. The result arguably takes less bandwidth and is much faster than existing alternatives.

Table 3. Unsigncryption efficiency

	Barbosa-Farshim	BLMQ	LXH	CLPKE	ours
pairings	4	2	4	NA	0
exponentiations	0	1	2	3	$1 + \varepsilon^{\ddagger}$
scalar multiplications	1	0	0	0	0
time (ms, BN-256)	399.0	236.0	475.5	124.3	54.8
time (ms, MNT4-256)	282.6	138.7	282.2	16.0	7.3

‡ 2 simultaneous exponentiations.

We are grateful to Benoît Libert and Mike Scott for enlightening discussions during the preparation of this work.

References

Al-Riyami, S. S. and Paterson, K. G. (2003). Certificateless public key cryptography. In *Advanced in Cryptology – Asiacrypt'2003*, volume 2894 of *Lecture Notes in Computer Science*, pages 452–473. Springer.

Baek, J., Safavi-Naini, R., and Susilo, W. (2005). Certificateless public key encryption without pairing. In *Information Security Conference – ISC'2005*, volume 3650 of *Lecture Notes in Computer Science*, pages 134–148. Springer.

Baek, J., Steinfeld, R., and Zheng, Y. (2002). Formal proofs for the security of signcryption. In *Public Key Cryptography – PKC'2002*, volume 2274 of *Lecture Notes in Computer Science*, pages 80–98. Springer.

Barbosa, M. and Farshim, P. (2008). Certificateless signcryption. In *ACM Symposium on Information, Computer and Communications Security – ASIACCS'2008*, Lecture Notes in Computer Science, Tokyo, Japan. Springer. To appear.

Barreto, P. S. L. M., Libert, B., McCullagh, N., and Quisquater, J.-J. (2005). Efficient and provably-secure identity-based signatures and signcryption from bilinear maps. In *Advanced in Cryptology – Asiacrypt'2005*, volume 3788 of *Lecture Notes in Computer Science*, pages 515–532. Springer.

Barreto, P. S. L. M. and Naehrig, M. (2006). Pairing-friendly curves of prime order. In *Selected Areas in Cryptography – SAC'2005*, volume 3897 of *Lecture Notes in Computer Science*, pages 319–331. Springer.

Bellare, M. and Rogaway, P. (1993). Random oracles are practical: A paradigm for designing efficient protocols. In *1st ACM Conference on Computer and Communications Security*, pages 62–73, Fairfax, USA. ACM Press.

Boneh, D. and Franklin, M. (2001). Identity-based encryption from the Weil pairing. In *Advanced in Cryptology – Crypto'2001*, volume 2139 of *Lecture Notes in Computer Science*, pages 213–229. Springer.

Freeman, D., Scott, M., and Teske, E. (2006). A taxonomy of pairing-friendly elliptic curves. IACR ePrint Archive, report 2006/372. `http://eprint.iacr.org/2006/372`.

Fujisaki, E. and Okamoto, T. (1999). Secure integration of asymmetric and symmetric encryption schemes. In *Advanced in Cryptology – Crypto'1999*, volume 1666 of *Lecture Notes in Computer Science*, pages 537–554. Springer.

Galbraith, S. D., Lin, X., and Scott, M. (2008). Endomorphisms for faster elliptic curve cryptography on general curves. IACR ePrint Archive, report 2008/194. `http://eprint.iacr.org/2008/194`.

Hess, F., Smart, N. P., and Vercauteren, F. (2006). The eta pairing revisited. *IEEE Transactions on Information Theory*, 52(10):4595–4602.

Li, F., Xin, X., and Hu, Y. (2007). A pairing-based signcryption scheme using self-certified public keys. *International Journal of Computers and Applications*, 29(3):278–282.

Libert, B. (2006). *New Secure Applications of Bilinear Maps in Cryptography*. PhD thesis, Université Catholique de Louvain, Louvain-La-Neuve, Belgium.

Libert, B. and Quisquater, J. J. (2006). On constructing certificateless cryptosystems from identity based encryption. In *Public Key Cryptography Workshop – PKC'2006*, volume 3958 of *Lecture Notes in Computer Science*, pages 474–490. Springer.

Miyaji, A., Nakabayashi, M., and Takano, S. (2001). New explicit conditions of elliptic curve traces for FR-reduction. *IEICE Transactions on Fundamentals*, E84-A(5):1234–1243.

Pointcheval, D. and Stern, J. (1996). Security proofs for signature schemes. In *Advanced in Cryptology – Eurocrypt'1996*, volume 1070 of *Lecture Notes in Computer Science*, pages 387–398. Springer.

Sakai, R. and Kasahara, M. (2003). ID based cryptosystems with pairing on elliptic curve. In *SCIS'2003*, Hamamatsu, Japan.

Sakai, R., Ohgishi, K., and Kasahara, M. (2000). Cryptosystems based on pairing. In *Symposium on Cryptography and Information Security – SCIS'2000*, Okinawa, Japan.

Schnorr, C. P. (1991a). Efficient signature generation by smart cards. *Journal of Cryptology*, 4(3):161–174.

Schnorr, C. P. (1991b). Method for identifying subscribers and for generating and verifying electronic signatures in a data exchange system. US Patent #4995082. `http://patft.uspto.gov/netacgi/nph-Parser?patentnumber=4995082`. Expired in February 2008.

Shamir, A. (1984). Identity based cryptosystems and signature schemes. In *Advances in Cryptology – Crypto'84*, volume 0196 of *Lecture Notes in Computer Science*, pages 47–53. Springer.

Vercauteren, F. (2008). Optimal pairings. IACR ePrint Archive, report 2008/096. `http://eprint.iacr.org/2008/096`.

Zheng, Y. (1997). Digital signcryption or how to achieve cost(signature & encryption) ¡¡ cost(signature) + cost(encryption). In *Advanced in Cryptology – Crypto'97*, volume 1294 of *Lecture Notes in Computer Science*, pages 165–179. Springer.