Int. J. Inf. Secur.
DOI 10.1007/s10207-008-0072-z

REGULAR CONTRIBUTION

Efficient identity-based GQ multisignatures

Lein Harn · Jian Ren · Changlu Lin

Abstract ISO/IEC 14888 specifies a variety of digital signature mechanisms to sign messages of arbitrary length. These schemes can be applied to provide entity authentication, data origin authentication, non-repudiation, and data integrity verification. ISO/IEC 14888 consists of three parts under the general title Information technology—Security techniques—Digital signatures. Part II, or ISO/IEC 14888-2 specifies the general structure and the fundamental procedures for the generation and verification of an identity-based signature (IBS) mechanism for messages of arbitrary length. Particularly, the IBS scheme of Guillou and Quisquater (GQ) is described in Clauses 6–8. In this paper, an efficient identity-based multisignature (IBMS) scheme is proposed for the GQ IBS scheme, which allows multiple users using the ISO/IEC 14888-2 standard GQ scheme to generate multisignatures. The scheme is efficient in the sense that both the length and the verification time of the multisignatures are fixed. The proposed ID-based multisignature scheme is also secure against forgeability under adaptive chosen-message attack and adaptive chosen-identity attack in random oracle model.

Keywords ISO/IEC standard · Identity-based signature · Multisignature · Public-key cryptography

L. Harn
Department of Computer Science and Electrical Engineering, University of Missouri, Kansas City, MO 64110, USA
e-mail: harnl@umkc.edu

J. Ren (✉)
Department of Electrical and Computer Engineering, Michigan State University, East Lansing, MI 48824, USA
e-mail: renjian@egr.msu.edu

C. Lin
State Key Laboratory of Information Security, Graduate University of Chinese Academy of Sciences, 100049 Beijing, China
e-mail: lincl@is.ac.cn

C. Lin
The Key Laboratory of Network Security and Cryptology, Fujian Normal University, 350007 Fujian, People's Republic of China

1 Introduction

Public-key cryptography is playing an increasingly popular and important role in the era of modern communications. Public-key cryptosystem allows users to communicate securely without pre-shared secret key and to authenticate data exchange. In public-key cryptography, a digital signature, which is generated using the message signer's private key can provide non-repudiation evidence of the message signer. Non-repudiation is a way to guarantee that the message sender cannot later on deny having sent the message. Public-key cryptography can also be used in many other areas such as e-voting, e-cash and e-commerce.

A digital multisignature is similar to a regular digital signature of a message; however, a digital multisignature is generated by multiple signers with knowledge of multiple private keys. Generally speaking, the major difference between a hand-written multisignature and a digital multisignature is the length of the multisignature. In a hand-written multisignature, the length of a hand-written multisignature is linear in the number of signers, while in a digital multisignature, the length of the digital multisignature is determined by a cryptographic assumption, such as the difficulty of factoring a large composite integer into two prime numbers or the difficulty of solving the discrete logarithm, and the length of a digital multisignature can be identical to the length of a single digital signature. Digital multisignature is just a string of binary bits that can only be generated with the

Published online: 08 January 2009

knowledge of multiple private keys. Any verifier can validate the multisignature of a given message based on all signers' public keys. The verification time of the multisignature can be fixed (if we only consider the time needed for modulo exponentiations), instead of linear in the number of message signers.

The concept of digital multisignature is very similar to the concept of group-oriented threshold signature. The group-oriented cryptography was first introduced by Desmedt [1]. By applying the concept of group-oriented cryptography, threshold signature schemes can be developed. Several threshold signature schemes and their modifications have been developed [2–6]. In a threshold signature scheme, a signature is generated by a number of participating members, which is larger than or equal to a predefined threshold value. For instance, in a (t, n) threshold signature scheme, any t or more than t members can represent the group to generate a group signature. Later, the verifier can use the group's public key to validate the group signature. The special case of the threshold signature called the $(1, n)$ group signature was proposed by Chaum and Heyst [3]. In a $(1, n)$ group signature, a group signature could be generated by an employee (i.e. a group member) of a large company, and be verified by any outside verifier as a normal digital signature, but not be able to identify the particular employee who signed it. However, even all other group members (and the manager) collude, they cannot forge a signature for a non-participating group member. Boyd [7] proposed the first (n, n) threshold signature based on RSA scheme in which all signers share the same modulus. The length of the group signature is fixed and the verification time of the group signature is also constant. However, it is only an (n, n) threshold signature scheme. It is not a multisignature scheme since the signer's group is predefined and cannot be changed through the application. Although multiple signers are involved in generating a digital multisignature and a threshold signature, there is a main difference between these two signatures. In a threshold signature application, the signing group is predefined and cannot be changed. However, in a multisignature application, the signing group can be dynamically formed by any set of signers.

An *efficient multisignature* scheme should possess the following two properties:

- *Fixed length*. Fixed length means that the length of the multisignature is the same as the length of a single signature.
- *Constant verification time*. Constant verification time means that to verify the multisignature, the number of modulo exponentiations required is the same as the verification of a single signature.

An efficient digital multisignature scheme [8] based on discrete logarithm problem has been proposed in 1994. In this scheme, the length and the verification time of the multisignature are both fixed. For RSA [9] based schemes, there exists no efficient multisignature scheme in the literature. The difficulty lies in the fact that for security purpose, each message signer has to select different modulus. On the other hand, to generate RSA based multisignatures, all signing processes are required to operate in the same domain [10,11]. Therefore, the moduli clashing problem has to be solved first. Though some schemes have been proposed in literature to overcome the moduli clashing problem [12,13]; however, for all these schemes, the verification time of each multisignature is still linear in the number of message signers involved.

In 1984, Shamir [14] introduced the concept of an identity-based (ID-based) cryptosystem to simplify the public-key authentication problem. In this system, each signer needs to register at a private key generator (PKG) and identify himself before joining the network. Once a signer is accepted, the PKG will generate a private key for that signer based on the signer's identity, which may include the signer's name, email address, etc. The signer's identity will be the signer's public key. In this way, a verifier only needs to know the "identity" of his communication partner and the public key of the PKG, to verify a digital signature or to send an encrypted message. There is no public-key digital certificate of each user needed in this system.

In the same paper, Shamir also proposed an identity-based signature (IBS) scheme [14] based on integer factorization problem (IFP). Bellare et al. [15] proved that the scheme is secure against forgeability under chosen-message attack. In 1988, Guillou and Quisquater introduced a "paradoxical" IBS scheme [16] also based on the IFP. Later on it was adopted as the International Standard ISO/IEC 14888-2 [17] and specified as the Guillou–Quisquater (GQ) signature.

In this paper, we propose an efficient ID-based multisignature (IBMS) scheme, which allows multiple message signers using the ISO/IEC 14888-2 standard GQ scheme to generate a multisignature. In our scheme, the length of the multisignature and the verification time of the multisignatures are both fixed. We also prove that the proposed IBMS is secure against forgeability under adaptive chosen-message attack and adaptive chosen-identity attack.

The paper is organized as follows. In Sect. 2, we present a brief review of the GQ IBS scheme. Our proposed multisignature scheme is described in Sect. 3. The security analysis of our proposed IBMS scheme is presented in Sect. 4. We conclude in Sect. 5.

2 Review of GQ IBS scheme in ISO/IEC 14888-2

In a GQ IBS scheme [17], we need a PKG to determine the private key for each signer.

PKG key generation: $\mathsf{K}_{\text{pkg}}(1^k)$

Similar to an RSA signature scheme with security parameter 1^k, the PKG runs the random oracle $\mathsf{K}_{\text{pkg}}(1^k) = \mathsf{K}_{\text{rsa}}(1^k)$ to generate two large prime numbers p and q. Then PKG performs the following three steps:

1. Calculates $n = pq$.
2. Selects a number e that is relatively prime to $\phi(n)$, where $\phi(n)$ is Euler's totient function.
3. Calculates d such that $d \cdot e = 1 \bmod \phi(n)$.
4. Chooses two cryptographic hash functions $H_1, H_2 : \{0,1\}^* \to \mathbb{Z}_n$.
   The system parameters are as follows:
   Public key: $\mathsf{Pk} = (n, e, H_1, H_2)$, and Master key: $\mathsf{Mk} = (p, q, d)$.

User key generation: $\mathsf{K}_u(1^k, \mathsf{Pk}, \mathsf{Mk}, ID_A)$

The user A with identity ID_A sends his identity to the PKG.

1. The PKG verifies the identity and calculates a "shadow" for user A as $J_A = H_1(ID_A)$. J_A serves as the public key of the user A with identity ID_A.
2. The PKG signs J_A as $s_A = J_A^{-d} \bmod n$ and sends s_A to A securely.
3. The user A verifies s_A as $s_A^e = J_A^{-1} \bmod n$ and uses it as his private key.

Message signing: $\mathsf{Sign}(\mathsf{Pk}, M, ID_A)$

For a user A with identity ID_A, to sign a message M, A randomly selects an $r \in \mathbb{Z}_n$, then calculates:

1. $u = r^e \bmod n$.
2. $b = H_2(u \| M)$, where $\|$ stands for concatenation.
3. $v = r \cdot s_A^b \bmod n$.

The signature for message M is then defined as: $\sigma = (b, v) \in \mathbb{Z}_n \times \mathbb{Z}_n$.

Signature verification: $\mathsf{Vf}(\mathsf{Pk}, \sigma, M, ID_A)$

To verify a signature $\sigma = (b, v)$ of a user A on a message M, the verifier computes:

$$u = J_A^b \cdot v^e \bmod n.$$

The verifier accepts the signature *if and only if* the following equation holds.

$$b = H_2(u \| M) \bmod n. \tag{1}$$

This is because if $\sigma = (b, v)$ is a legitimate signature, then we should have

$$J_A^b \cdot v^e = s_A^{-eb} r^e s_A^{be} = r^e = u \bmod n.$$

Therefore, the signature verification equation should hold true.

3 Proposed identity-based multisignature (IBMS)

In this section, we propose an ID-based multisignature scheme for the GQ signature. This scheme allows multiple users using the GQ IBS specified in ISO/IEC 14888-2 standard to generate multisignatures. In our proposed IBMS scheme, the PKG key generation and the user private key generation are both identical to the single IBS scheme specified in the standard.

PKG key generation: $\mathsf{K}_{\text{pkg}}(1^k)$

Similar to an RSA signature scheme with security parameter 1^k, the PKG runs the random oracle $\mathsf{K}_{\text{pkg}}(1^k) = \mathsf{K}_{\text{rsa}}(1^k)$ to generate two large prime numbers p and q. Then PKG performs the following three steps:

1. Calculates $n = pq$.
2. Selects a number e that is relatively prime to $\phi(n)$, where $\phi(n)$ is Euler's totient function.
3. Calculates d such that $d \cdot e = 1 \bmod \phi(n)$.
4. Chooses two cryptographic hash functions $H_1, H_2 : \{0,1\}^* \to \mathbb{Z}_n$.
   The system parameters are as follows.
   Public key: $\mathsf{Pk} = (n, e, H_1, H_2)$, and Master key: $\mathsf{Mk} = (p, q, d)$.

User private key generation: $\mathsf{K}_u(1^k, \mathsf{Pk}, \mathsf{Mk}, ID_i)$

In this algorithm, the signer with identity ID_i gets a copy of his private key from the PKG through a 2-step process described below:

1. A signer submits his identity ID_i to the PKG.
2. The PKG, with its private key d and the corresponding public key e, signs ID_i, which generates a private key s_i, such that $s_i = J_i^{-d} \bmod n$, where $J_i = H_1(ID_i)$ is the shadow of ID_i. s_i is the signer i's private key.
3. The user i verifies s_i as $s_i^e = J_i^{-1} \bmod n$ and uses it as his private key.

Message signing: $\mathsf{Sign}(\mathsf{Pk}, M, \{ID_i\}_{i=1}^l)$

We assume that there are l signers, u_i, $i = 1, 2, \ldots, l$, agree to sign a message M together. To generate the ID-based multisignature, each signer carries out the followings steps as shown in Fig. 1:

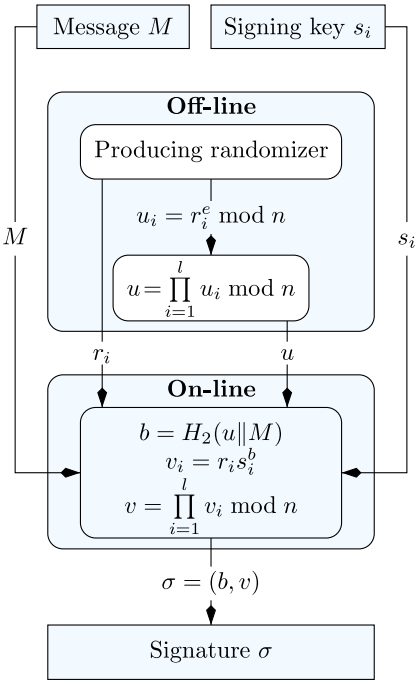

Fig. 1 Message signing process

Off-line steps (these are steps that do not depend on the message M and can be performed off-line)

1. Selects $r_i \in \mathbb{Z}_n$ randomly.
2. Computes $u_i = r_i^e \bmod n$.
3. Broadcasts u_i to all other signers.
4. Upon receiving of $u_i, i = 1, 2, \ldots, l$, each signer computes

$$u = \prod_{i=1}^{l} u_i \bmod n.$$

On-line steps (These are steps that depend on the message M)

1. Each signer ID_i computes $b = H_2(u\|M)$ and $v_i = r_i \cdot s_i^b \bmod n$, and broadcasts v_i to all other signers.
2. After receiving of $v_i, i = 1, 2, \ldots, l$, the multisignature component v can be computed as

$$v = \prod_{i=1}^{l} v_i \bmod n = \prod_{i=1}^{l} r_i \cdot s_i^b \bmod n.$$

The multisignature for message M is $\sigma = (b, v) \in \mathbb{Z}_n \times \mathbb{Z}_n$.

From the above algorithm, it is clear that the signing phase of each individual signature is identical to the original IBS scheme. It is also clear that the length of each multisignature is the same as the individual IBS signature.

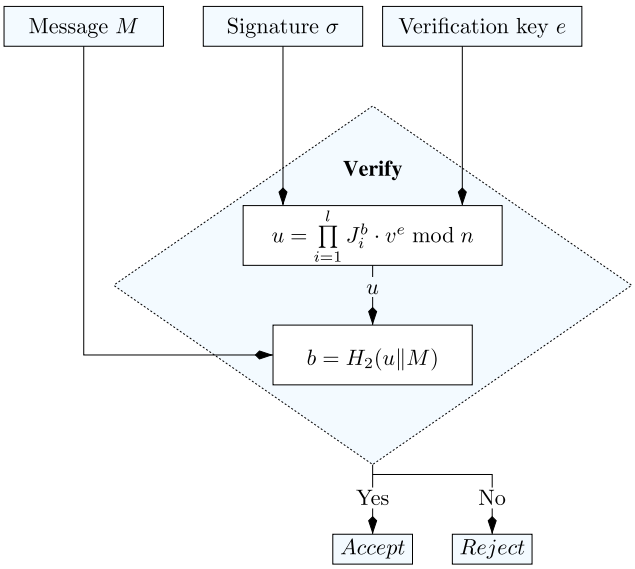

Fig. 2 Multisignature verification

Multisignature verification: $\mathsf{Vf}(\mathsf{Pk}, \sigma, M, \{ID_i\}_{i=1}^{l})$

To verify a multisignature $\sigma = (b, v)$ on a message M, as shown in Fig. 2, the verifier computes

$$\prod_{i=1}^{l} J_i^b \cdot v^e \bmod n = \prod_{i=1}^{l} J_i^b \cdot r_i^e \cdot J_i^{-b} \bmod n$$
$$= \prod_{i=1}^{l} r_i^e \bmod n = u.$$

The verifier accepts the signature *if and only if* the following equation holds

$$b = H_2(u\|M) \bmod n. \tag{2}$$

From the above verification algorithm, we can see that for the proposed IBMS scheme, the number of modulo exponentiations is identical to the verification of an individual IBS. However, it does require $l - 1$ extra modulo multiplications. Since modulo multiplications are much more efficient than modulo exponentiations, they can simply be ignored. Therefore, the verification time of each multisignature is fixed.

4 Security analysis

In this section, we will analyze the security of our proposed IBMS scheme from existential forgery under adaptive chosen-message attack and adaptive chosen-identity attack.

Informally, an *existential forgery* in IBMS scheme refers to that the adversary attempts to forge an IBMS of messages and identities of signes at its choice. We now formalize the notion of security for IBMS scheme as follows. To perform existential forgery of an IBMS, the adversary may corrupt arbitrary signers and send messages during multisignature generation. We allow the adversary to choose the identities, forge multisignatures and extract the private key for identities at its choice. We also allow the adversary to access the signing oracle on any desired identity and message. The adversary's advantage $Adv(\mathcal{A})$ is defined as the probability of $\mathcal{A}$ against a challenger $\mathcal{C}$ to win the following game.

Setup: The challenger $\mathcal{C}$ takes a security parameter 1^k and runs the PKG key generation algorithm, it gives $\mathcal{A}$ the resulting system parameters.

Oracle Queries: $\mathcal{A}$ requests the following queries adaptively.

- Extract Query: $\mathcal{A}$ may choose identity ID_i and request the user private key generation. $\mathcal{C}$ outputs the private key corresponding to ID_i.
- Sign Query: $\mathcal{A}$ may request an IBMS on $(\mathsf{Pk}, M, \{ID_i\}_{i=1}^{l})$. $\mathcal{C}$ outputs the corresponding multisignature σ to $\mathcal{A}$.

Forgery: For some $(\mathsf{Pk}, M^*, \{ID_i^*\}_{i=1}^{l})$, $\mathcal{A}$ outputs an IBMS σ^*. The restriction is that for some i, $1 \le i \le l$, $\mathcal{A}$ does not extract the private key for ID_i^* and does not request a multisignature including the pair (M^*, ID_i^*). $\mathcal{A}$ wins the game if $\mathsf{Vf}(\mathsf{Pk}, M^*, \sigma^*, \{ID_i^*\}_{i=1}^{l})$ is $Accept$.

Definition 1 We say that an IBMS scheme IBMS is secure against existential forgery under adaptive chosen-message attack and adaptive chosen-identity attack, if for all polynomial-time adversary $\mathcal{A}$, the advantage $Adv(\mathcal{A})$ that $\mathcal{A}$ wins the above defined game is negligible.

Note 1 The above definition is similar to the definitions presented in [18,19].

In 2004, Bellare et al. [15] proved that the basic identity-based signature scheme is secure against existential forgery under adaptive chosen-message attack and adaptive chosen-identity attack in the random oracle model. The two hash functions H_1 and H_2 which are used in our proposed scheme will be treated as random oracles in the following security analysis.

Theorem 1 *The proposed IBMS scheme is secure against existential forgery under adaptive chosen-message attack and adaptive chosen-identity attack in the random oracle model.*

Proof Let $\mathcal{A}_{\mathrm{IBMS}}$ and $\mathcal{A}_{\mathrm{IBS}}$ be polynomial-time adversaries of our proposed IBMS scheme and the basic IBS scheme, respectively. We can prove this theorem according to [15]. The main idea is that if the adversary $\mathcal{A}_{\mathrm{IBMS}}$ can forge a valid multisignature on an arbitrary message M without interacting with the honest signer, then the adversary $\mathcal{A}_{\mathrm{IBS}}$ can also forge a valid basic signature on message M of an honest message signer.

We now describe the detailed proof as follows. The adversary $\mathcal{A}_{\mathrm{IBS}}$ chooses an identity ID and requests the hash oracle and signing oracle of any message. $\mathcal{A}_{\mathrm{IBS}}$ will run $\mathcal{A}_{\mathrm{IBMS}}$ to simulate a single honest singer. When $\mathcal{A}_{\mathrm{IBMS}}$ wants to get a valid multisignature, it runs the signing oracle for $\mathcal{A}_{\mathrm{IBS}}$, $\mathcal{A}_{\mathrm{IBS}}$ simulates $\mathcal{A}_{\mathrm{IBMS}}$'s random hash oracle using its own oracle. $\mathcal{A}_{\mathrm{IBS}}$ will then request the signing oracle with the identity of honest signer ID and the corresponding message. The signing oracle will generate the output to $\mathcal{A}_{\mathrm{IBMS}}$. It is easy to know that the output of $\mathcal{A}_{\mathrm{IBMS}}$ is a valid multisignature (a successful forgery for message) as long as the answer from $\mathcal{A}_{\mathrm{IBS}}$ is a valid signature. However, according to [15], no valid basic identity-based signature can be generated from $\mathcal{A}_{\mathrm{IBS}}$. Therefore, our proposed IBMS scheme is secure under the random oracle model. □

5 Conclusion

In this paper, we proposed an efficient ID-based multisignature scheme for the GQ IBS scheme as specified in International Standard ISO/IEC 14888-2. Our scheme has fixed signature length and the signature verification time is independent of the number of signers involved. The proposed scheme is secure against forgeability under adaptive chosen-message attack and adaptive chosen-identity attack in random oracle model.

References

1. Desmedt, Y.: Society and group oriented cryptography: a new concept. In: Pomerance, C. (ed.) Advances in Cryptology—Crypto'87. Lecture Notes in Computer Science, vol. 293, pp. 120–127. Springer, Berlin (1987)
2. Chang, C., Lee, H.: A new generalized group oriented cryposcheme without trusted centers. IEEE J. Selected Areas Commun. **11**(5), 725–729 (1993)
3. Chaum, D., Heyst, E.v: Group signatures. In: Davies, D.W. (ed.) Advances in Cryptology—EuroCrypt'91. Lecture Notes in Computer Science, vol. 547, pp. 257–265. Springer, Berlin (1991)
4. Desmedt, Y., Frankel, Y.: Threshold cryptosystems. In: Brassard, G. (ed.) Advances in Cryptology—Crypto'89. Lecture Notes in Computer Science, vol. 435, pp. 307–315. Springer, Berlin (1989)
5. Desmedt, Y., Frankel, Y.: Shared generation of authenticators and signatures. In: Feigenbaum, J. (ed.) Advances in Cryptology—Crypto'91. Lecture Notes in Computer Science, vol. 576, pp. 457–469. Springer, Berlin (1991)
6. Laih, C., Harn, L.: Generalized threshold cryposystem. In: Advances in Cryptology—ASIACRYPT, pp. 159–169 (1991)

7. Boyd, C.: Digital multisignatures. Cryptography and Coding, pp. 241–246 (1989)
8. Harn, L.: Group-oriented (t, n) threshold digital signature scheme and digital multisignature. IEEE Proc. Comput. Digit. Tech. **141**(5), 307–313 (1994)
9. Rivest, R., Shamir, A., Adleman, L.: A method for obtaining digital signatures and public-key cryptosystems. Commun. Assoc. Comp. Mach. **21**(2), 120–126 (1978)
10. Kohnfelder, L.M.: On the signature reblocking problem in public-key cryptography. Commun. ACM **21**(2), 179 (1978)
11. Kiesler, T., Harn, L.: RSA blocking and multisignature schemes with no bit expansion. Electron. Lett. **26**(18), 1490–1491 (1990)
12. Harn, L., Kiesler, T.: New scheme for digital multisignature. Electron. Lett. **25**(15), 1002–1003 (1989)
13. Pon, S.-F., Lu, E.-H., Lee, J.-Y.: Dynamic reblocking rsa-based multisignatures scheme for computer and communication. IEEE Commun. Lett. **6**(1), 43–44 (2002)
14. Shamir, A.: Identity-based cryptosystems and signature schemes. In: Blakley, G.R., Chaum, D. (eds.) Advances in Cryptology: Proceedings of Crypto'84. Lecture Notes in Computer Science, vol. 196, pp. 47–53. Springer, Berlin (1985)
15. Bellare, M., Namprempre, C., Neven, G.: Security proofs for identity-based identification and signature schemes. In: Koblitz, N. (ed.) Advances in Cryptology—EurcoCrypt '04. Lecture Notes in Computer Science, vol. 3027, pp. 268–286. Springer, Berlin (2004)
16. Guillou, L.C., Quisquater, J.J.: A "paradoxical" identity-based signature scheme resulting from zero-knowledge. In: Goldwasser, S. (ed.) Advances in Cryptology—Crypto'88. Lecture Notes in Computer Science, vol. 403, pp. 216–231. Springer, Berlin (1989)
17. I.S.I. 14888-2, Information technology—security techniques—digital signatures wi appendix—part 2: Identity-based mechanisms. December (1999)
18. Micali, S., Ohta, K., Reyzin, L.: Accountable subgroup multi-signatures. ACM Conference on Computer and communications Security, pp. 245–254. ACM, New York (2004)
19. Boldyreva, A.: Threshold signatures, multi-signatures and blind signatures based on the GDH group signature scheme. In: Goos, G., Hartmanis, J., van Leeuwen, J. (eds.) Proc. Public Key Cryptography. Lecture Notes in Computer Science, vol. 2567, pp. 31–46. Springer, Berlin (2003)