Information Sciences 276 (2014) 21–30

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Certificateless proxy multi-signature

Hongzhen Du [a,*], Qiaoyan Wen [b]

[a] Mathematics Department, Baoji University of Arts and Sciences, Baoji 721007, China
[b] State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China

ARTICLE INFO

Article history:
Received 19 December 2008
Accepted 6 February 2014
Available online 19 February 2014

Keywords:
Certificateless cryptography
Proxy signature
Proxy multi-signature
Bilinear pairing
Computational Diffie–Hellman problem

ABSTRACT

Proxy multi-signature allows a group of original signers to delegate their signing capabilities to a proxy signer in such a way that the proxy signer can sign messages on behalf of the group of original signers. Existing constructions of proxy multi-signatures are based on traditional Public Key Infrastructure or Identity-based Public Key Setting, but the former needs certificates which bring about many certificate management problems, and the latter has a drawback of key escrow. In contrast to the existing constructions, in this paper, we study proxy multi-signature in Certificateless Public Key Cryptography (CL-PKC) which combines the advantages of both certificate-based and identity-based cryptosystems as it avoids the use of certificates and does not suffer from key escrow. We provide the definition and the security model for Certificateless Proxy Multi-Signature (CLPMS), and propose the first CLPMS scheme which is proved to be secure in the random oracle model under the computational Diffie–Hellman assumption. Our scheme is computationally efficient and has the property that the size of a proxy multi-signature is independent of the number of the original signers.

1. Introduction

In a traditional Public Key Cryptography (PKC), anyone who wants to send messages to others must obtain their certificates that contain the public keys. However, this requirement brings about lots of certificate management problems in practice. In order to avoid these problems, Shamir [15] first introduced the concept of Identity-based PKC (ID-PKC) in 1984, which allows a user to use his identity information as his own public key. It means that there is no need for a user to obtain other users' certificates before communication. However, in an ID-PKC, there inherently exists a drawback called private key escrow because this cryptosystem involves a Private Key Generator (PKG), which is responsible for generating a user's private key based on his identity. As a result, the PKG can literally decrypt any ciphertext and forge any user's signature on any message.

Certificateless Public Key Cryptography (CL-PKC), which is an intermediate model between traditional PKC and ID-PKC, was first introduced by Al-Riyami and Paterson [1] in 2003. Unlike in an ID-PKC, a user's private key in a certificateless cryptosystem consists of a partial private key generated by a Key Generation Center (KGC) and a secret value chosen by the user. As a result, the KGC cannot impersonate the user since it does not know the user's secret value. In such a way, the key escrow problem of ID-based PKC can be solved. Meanwhile, the CL-PKC schemes are not purely ID-based, and there exists an additional public key for each user. Fortunately, this additional public key does not need to be certified by any trusted authority

* Corresponding author at: Mathematics Department, Baoji University of Arts and Sciences, Baoji 721007, China.
E-mail address: duhongzhen@gmail.com (H. Du).

http://dx.doi.org/10.1016/j.ins.2014.02.043
0020-0255/

as in the conventional PKC. The structure of the certificateless scheme ensures that the public key can be verified without a certificate. Following the pioneering work of Al-Riyami and Paterson [1], many novel certificateless encryption and signature schemes [3–5,12,18] have been proposed.

The concept of proxy signature was introduced by Mambo et al. in 1996 [13]. In a proxy signature scheme, an original signer is allowed to authorize a designated person as his proxy signer. Then the proxy signer is able to sign messages on behalf of the original signer. Proxy signature schemes are useful in many applications. For example, a manager can delegate his secretaries to sign documents when he is on vacation. Proxy signature schemes can also be used in electronics transaction [8] and mobile agent environment [14,9]. A secure proxy signature scheme must satisfy these properties which were stated in [9]: verifiability, strong unforgeability, strong identifiability, strong undeniability, and prevention of misuse.

Proxy Multi-Signature (PMS), an extension of the basic proxy signature, was introduced by Yi et al. in 2000 [17]. In a PMS scheme, a designated proxy signer can generate the signature on behalf of two or more original signers. PMS can play an important role in the following scenario: A company releases a document that may involve the financial department, engineering department, and program office, etc. The document must be signed jointly by these entities, or signed by a proxy signer authorized by these entities. One solution to the latter case of this problem is to use a PMS scheme. Several novel PMS schemes [7,6,10,11,16,2] have been proposed since 2000. However, none of these schemes was based on CL-PKC.

1.1. Our contributions

This paper studies PMS in CL-PKC. We present the notion and the security model for Certificateless Proxy Multi-Signature (CLPMS). And we propose a secure CLPMS scheme from bilinear pairings for the first time. And then we provide the security proofs for the scheme in the random oracle model under the assumption that the computational Diffie–Hellman problem is intractable.

1.2. Organization

In Section 2, we provide the framework of CLMPS. In Section 3, we propose an efficient CLMPS scheme and provide its security proofs in the random oracle model. Conclusion is drawn in the last section.

2. Framework of certificateless proxy multi-signature

In this section, we first present the formal definition of CLPMS. Then, we give the security model for CLPMS.

2.1. Definition of certificateless proxy multi-signature

In a certificateless proxy multi-signature scheme, there is one proxy signer and n (>1) original signers. Let $\mathcal{O}_i$ ($1 \leqslant i \leqslant$ n) be the original signer and Π be the proxy signer designated by every original signer $\mathcal{O}_i$. Here assume that each $\mathcal{O}_i$ has an identity ID_i and a corresponding public key pk_i, for $1 \leqslant i \leqslant$ n, and Π has an identity ID_p and a corresponding public key pk_p.

Definition 1. A CLPMS scheme consists of the following eight algorithms:

1. **Setup** is a probabilistic polynomial time (PPT) algorithm, run by a Key Generation Centre (KGC). It takes a security parameter k as input and returns the master-key s and the system parameters params.
2. **Partial-Private-Key-Extract** is a PPT algorithm, run by the KGC. Taking params, master-key s, and a user's identity ID as inputs, this algorithm returns a partial private key d_{ID} for the user.
3. **User-Key-Generation** is a PPT algorithm, run by the user with identity ID. Taking params and the user's ID as inputs, this algorithm outputs the user secret value x_{ID} and the user public key pk_{ID}.
4. **Sign** is a (possibly) PPT ordinary certificateless signing algorithm. Taking as inputs params, a message m, a signer's identity ID and his public key pk_{ID}, and his partial private key d_{ID} and his secret value x_{ID}, the algorithm outputs a signature σ on message m.
5. **Verify** *is* a deterministic verification algorithm. Taking as inputs params, the signer's identity ID and his public key pk_{ID}, and a message-signature pair (m, σ), this algorithm returns $\mathcal{T}$ if σ is valid, and returns $\perp$ otherwise.
6. **PMD, PMP** (interactive) are algorithms Designation and Proxy, where PMD and PMP are owned by every original signer $\mathcal{O}_i$ *for* $1 \leqslant i \leqslant n$ *and the proxy signer* $\mathcal{P}$, respectively. The inputs to each algorithm include all participants' identities ID_p, ID_1, ID_n and the corresponding public keys $pk_P, pk_1, \ldots, pk_n$. PMD also takes as inputs every $\mathcal{O}_i$'s partial private key d_i and the secret value x_i ($1 \leqslant i \leqslant n$), and a warrant $w = \{ID_1, \ldots, ID_{n+1}, pk_1, \ldots, pk_{n+1}, MessageScope, TimePeriod, etc.\}$ which includes the type of the information delegated, the period of delegation, the identities and the public keys of the original signers and the proxy signer, etc. PMP also takes as inputs the proxy signer's partial private key d_p and the secret value x_p. As a result of the interaction, the expected output of PMP is the warrant w and a proxy signing key sk_P which the proxy signer uses to produce proxy multi-signatures on behalf of all original signers. PMD has no output.