Available online at www.sciencedirect.com
ScienceDirect
Computer Standards & Interfaces 29 (2007) 191–195
COMPUTER STANDARDS & INTERFACES
www.elsevier.com/locate/csi

A novel proxy key generation protocol and its application

Xiaoming Hu *, Shangteng Huang

Department of Computer Application Technology, Shanghai Jiao Tong University, Shanghai 200030, PR China

Received 20 November 2005; accepted 16 March 2006
Available online 4 May 2006

Abstract

Proxy signature is an important technology in secure e-commerce. Short signature and fast verification are of paramount importance for its practical application. In this paper, we propose a proxy signature key generation protocol in which the warrant is the proxy public key for the first time. This protocol can be combined with ID-based signature scheme from bilinear pairing to construct an ID-based proxy signature scheme. Furthermore, compared with all previous proxy signature schemes, the scheme constructed has two virtues: (1) the proxy signature is shorter because it does not include any parameters for rebuilding the proxy public key; (2) the verification of the proxy signature is faster because the public proxy key does not have to be computed. Also we give such a proxy signature scheme.

Keywords: Proxy signature; Bilinear pairings; Proxy key; ID-based signature

1. Introduction

In 1996 [1], Mambo, Usuda, and Okamoto introduced the concept of proxy signature. A proxy signature scheme consists of three entities: original signer, proxy signer and verifier. In this scheme, an original signer can delegate his signing capability to a proxy signer in such a way that the proxy signer can sign any message on behalf of the original signer and the verifier can verify and distinguish between original signature and proxy signature by a verification equation. There are three types of delegation, full delegation, partial delegation and delegation by warrant. In 1997, S. Kim et al. gave a new type of delegation called partial delegation with warrant [2], which can be considered as the combination of partial delegation and delegation by warrant. After Mambo et al.'s first scheme was proposed, many proxy signature schemes have been proposed [2–5]. In 1984, Shamir [6] asked for an ID-based public key encryption scheme in which the public key can be an arbitrary string. In 2001, Dan Boneh et al. [7] proposed a fully functional ID-based encryption scheme from bilinear pairing. From then on, some ID-based proxy signature schemes were proposed as such [8–11]. In their schemes, the proxy signer generates the proxy key according to the delegation of the original signer. And the proxy public key is a function of the proxy warrant, the original signer's public key, the proxy's public key and some parameters introduced in the proxy key generation protocol, so the proxy signatures would include the parameters for rebuilding the proxy public keys. Furthermore, the proxy public keys would be computed to verify the proxy signatures. So these schemes are complicated and computational cost is high.

In this paper, we propose a novel proxy signature key generation protocol in which we consider the warrant as the proxy public key for the first time. The protocol consists of three entities: original signer, proxy signer and KGC that establishes the identity-based cryptosystem and generates private keys for users. The proxy key pair is a tuple (warrant, secret key). It is an ID-based key pair as (ID, secret key), so this protocol can be combined with any ID-based signature scheme from bilinear pairing to construct an ID-based proxy signature scheme. Compared with all previous proxy signature schemes, the scheme has two virtues: (1) the proxy signature is shorter because it does not include any parameters for rebuilding the proxy public key; (2) the verification of the proxy signature is faster because the public proxy key does not have to be computed. If the ID-based

* Corresponding author.
E-mail addresses: huxm@sjtu.edu.cn, yd_hxm@yahoo.com.cn (X. Hu).

0920-5489/$ - see front matter © 2006 Elsevier B.V. All rights reserved.
doi:10.1016/j.csi.2006.03.005

signature scheme is secure, the proxy signature scheme has the properties listed as follows [12]:

1. Strong unforgeability: A designated signer, called proxy signer, can create a valid proxy signature for the original signer. But the original signer and third parties who are not authorized cannot create a valid proxy signature.
2. Verifiability: From proxy signature a verifier can be convinced of the original signer's agreement on the signed message either by a self-authenticating form or by an interactive form.
3. Strong identity: any one can determine the identity of the corresponding proxy signer from a proxy signature.
4. Strong undeniability: Once a proxy signer creates a valid proxy signature for an original signer, the proxy signer cannot repudiate his signature creation.
5. Prevention of misuse: it should be confident that proxy key pair cannot be used for other purposes. Because the responsibility of proxy signer should be determined with warrant explicitly.

The rest of the paper is organized as follows: The next section briefly explains bilinear pairings. In Section 3, the proposed proxy key generation protocol is presented. Section 4 gives the analysis of the protocol. Section 5 gives its application in proxy signature scheme. Section 6 concludes this paper.

2. Bilinear pairings

Let G_1,+ be a cyclic additive group generated by P, whose order is prime q, and denotes by (G_1+). And let G_2 be cyclic multiplicative group of the same order q. H_1 and H_2 are two cryptographic hash functions. A bilinear pairing is map e: $G_1 \times G_1 \rightarrow G_2$ with the following properties:

1. Bilinear: For all P, $Q \in G_1$ and all a, $b \in z_q^*$, we have $e(aP, bQ) = e(P,Q)^{ab}$.
2. Non-degenerate: There exists P, $Q \in G_1$ such that $e(P, Q) \neq 1$.
3. Computable: There is an efficient algorithm to compute $e(P, Q)$ for all P, $Q \in G_1$.

Now we describe some mathematical problems in relation to bilinear pairing.

- **Discrete logarithm problem (DLP):** Given two group elements P and $Q \in G_1$, where P is a generator of G_1, find an integer n, such that $Q = nP$ whenever such an integer exists.
- **Decision Diffie–Hellman Problem (DDHP):** For a, b, $c \in z_q^*$, given P, aP, bP, $cP \in G_1$, where P is a generator of G_1, decide whether $c = ab \bmod q$.
- **Computational Diffie–Hellman Problem (CDHP):** For a, $b \in z_q^*$, given P, aP, $bP \in G_1$, where P is a generator of G_1, compute $abP \in G_1$.

We assume through this paper that CDHP and DLP are intractable. When the DDHP is easy but the CDHP is hard on the group G, we call *G Gap Diffie–Hellman* (*GDH*) *group*. Such groups can be found on supersingular elliptic curves or hyperelliptic curves over finite field, and the bilinear parings can be derived from the Weil or Tate pairing. We can refer to Galbraith et al. [13] for more details.

3. Proposed protocol

In this section, we propose a novel proxy signature key generation protocol from bilinear pairing which consists of three entities: original signer, proxy signer and key generation center (KGC). There are three procedures in the scheme: Setup, Extract, Generation of the proxy key.

3.1. Setup

Initially, KGC selects q, G_1, G_2, and e, as defined in the previous section. Then KGC chooses P as the generator of G_1 and defines two one-way hash functions $H_1: \{0,1\}^* \rightarrow G_1$ and $H_2: \{0,1\}^* \rightarrow z_q$. KGC selects $t \in z_q^*$ and computes $P_{\text{pub}} = t_{\text{P}}$, then he keeps t secretly as *master key* and publishes Params = $\{G_1, G_2, e, q, P, P_{\text{pub}}, H_1, H_2\}$.

3.2. Extract

- The original signer and the proxy submit their identity information ID_o, ID_p, to KGC.
- KGC computes a public/private key pair for them according to the following equations:

$$Q_\text{o} = H_1(\text{ID}_\text{o}), \quad Q_\text{p} = H_1(\text{ID}_\text{p}), \quad S_\text{o} = tQ_\text{o}, \quad S_p = tQ_\text{p}.$$

- KGC return them to the original signer and the proxy respectively. In this way, the public and private key of the original signer and the proxy can be denoted by Q_o, S_o and Q_p, S_p.

3.3. Generation of the proxy key

To delegate the signing capacity to proxy signer, the original signer uses Hess's ID-based signature scheme [15] to make the signed warrant W. The proxy key generation protocol is shown in Fig. 1.

- The original signer creates a warrant W where there is an explicit description of the delegation relation including the identity of the original signer and the proxy signer, the message to be signed, and so on. The original signer publish W and compute $S_1 = H_2(W, S_\text{o})$. S/He sends (W, S_1) to KGC.
- Since KGC know the secret keys S_o and S_p, KGC can verify

$$S_1 = H_2(W, S_\text{o}).$$

KGC accepts (W, S_1) if the above equation holds, then computes

$$Q_\text{w} = H_1(W)$$

$$S_\text{w} = tQ_\text{w}$$