Self Proxy Signature Scheme*

Young-Seol Kim**†*****, Jik Hyun Chang*****†***

†Department of Computer Science and Engineering, Sogang University, Seoul, Korea

Summary

The proxy signature scheme is a kind of digital signature scheme. In the proxy signature scheme, one user, called the original signer, can delegate his/her signing capability to another user called the proxy signer. This is similar to a person delegating his/her seal to another person in the real world. In this paper, we propose a new type of proxy signature scheme, the self proxy signature scheme. In this scheme, a signer, Alice, delegates her signing capability to herself and uses the proxy private/public key pair as temporary keys. Using this scheme, the signer protects her original private/public key pair and uses multiple private/public key pairs simultaneously. Therefore, the signer Alice uses a temporary key pair for only a particular work. Furthermore, it is easy to revoke the temporary private/public key pair. Thus, it can be said that the proposed scheme is practical.

Key words:

Digital signature, Proxy signature, Public key cryptography

1. Introduction

In 1996, Mambo, Usuda, and Okamoto proposed a new concept, the proxy signature scheme [1], [2]. In a proxy signature scheme, a user Alice, called the original signer, delegates her signing capability to another user, Bob, called the proxy signer, so that Bob can sign messages on behalf of Alice. A verifier can validate its correctness and can distinguish between a normal signature and a proxy signature. As a result, the verifier can be convinced of the original signer's agreement on the signed message. Proxy signature schemes have been suggested for use in a number of applications, including electronic commerce, mobile agents, distributed shared object systems, and so on. For example, the president of a company delegates a signing right to his/her secretary before a vacation. The secretary can make a signature on behalf of the president, and a verifier can be confident that the signature has been made by the authorized secretary. The verifier can also be convinced of the president's agreement on the signed message. Typically, a proxy signature scheme is as follows. The original signer Alice sends the proxy signer Bob a signature that is associated with a specific message. Bob makes a proxy private key using this information. Bob can then sign on a message with the proxy private key using a normal signature scheme. After the message and signature have been sent to the verifier, he/she recovers a proxy public key using public information and verifies the proxy signature using a normal signature scheme.

After Mambo, Usuda, and Okamoto [1], [2] proposed the proxy signature scheme, many kinds of proxy signature schemes were likewise proposed [3], [4], [5], [6]. For example, there are the proxy blind signature scheme and designated verifier proxy signature scheme.

In the real world, a person uses a legal seal and many other seals simultaneously. After registering, the legal seal is used in an important work, and the other seals are used for normal works. To use seals like this, the person protects the legal seal and uses another one for only a particular work.

In this paper, we propose a method to generate temporary keys like normal seals. It is a kind of proxy signature scheme, a self proxy signature scheme. The self proxy signature scheme is a new type digital signature that is proposed for the first time. In this scheme, a signer Alice delegates her signing capability to herself recursively. Using this scheme, the signer Alice generates many proxy private/public key pairs, uses them simultaneously, and revokes the temporary keys easily. Furthermore, it is easy to revoke the temporary private/public key pair. Thus is can be said that the self proxy signature scheme is practical. The proposed scheme is secure because the scheme satisfies all security properties.

The rest of this paper is organized as follows. Section 2 introduces the computational assumptions, definitions, and notations for a self proxy signature scheme. In Section 3, we briefly recall Mambo et al.'s proxy signature. Section 4 presents the self proxy signature scheme and Section 5 discusses its security. Finally, the conclusion is presented in Section 6.

2. Preliminaries

2.1 Assumption

Our scheme uses the following known difficult problem for its security.

Assumption 1: Discrete Logarithm (DL) assumption. Let $G_q = <g>$ be a cyclic multiplicative group generated by g of order q. Then on inputs (g, g^x) where $x \in_R Z_q$ is a random number, there is no

*This is the full version of a paper presented at Korea Information Science Society - Fall Conference(KISS'06 Fall).

probabilistic polynomial-time algorithm that outputs the value of x with a non-negligible probability.

This computational assumption is widely believed to be true for many cyclic groups, such as the multiplicative subgroup $G_q = < g >$ of the finite field Z_p, where p is a large prime and q is a prime factor of $p-1$. In practice, $|p| = 1024$ and $|q| = 160$ are considered to be suitable for most current security applications.

2.2 Definitions

Definition 1. A **self proxy signature scheme** is usually comprised of the following phases:
-Setup: The signer Alice generates her private, public key pair. This key pair is used in a normal signature scheme.
-Proxy delegation: The signer Alice performs the operations to generate a self proxy private, public key pair (x_p, y_p). x_p is known only to the signer Alice, and y_p is public or revocable publicly.
-Self proxy signature generation: The signer Alice signs on a message using a self proxy private key x_p and sends the message and its signature to a verifier.
-Self proxy signature verification: The verifier Bob verifies the message and its signature using a verification equation.

The security requirements for the self proxy signature are specified in the next section.

Definition 2. A **secure** self proxy signature scheme should satisfy the following requirements:
-Unforgeability: Only the valid signer can create the self proxy signature.
-Undeniability: The signer cannot deny his/her signatures to anyone.
-Distinguishability: The self proxy signature must be distinguishable from the normal signature.
-Verifiability: The self proxy signature can be verified by everyone.

2.3 Notations

For the convenience of describing our work, we define the parameters as follows:

- p, q : two large prime numbers, $q \mid p-1$.
- g : an element of Z_p^*, its order is q.
- x_U, y_U : a participant U 's private key and public key, respectively, $y_U = g^{x_U} \bmod p$.
- $H()$: a public cryptographically strong hash function
- $||$: denotes the concatenation of strings
- m_w : the warrant which specifies the delegation period for the kind of message m is delegated, the identities of the signer, etc.

3. Related Work

The first proxy signature scheme was Mambo et al.'s scheme [1], [2]. This was constructed by partial delegation and was based on a discrete logarithm problem (DLP). The original signer has the private key x and the public key $y = g^x \bmod p$. Mambo et al.'s scheme is as follows:

(1) Generating the secret information: The original signer chooses k randomly and computes $r = g^k \bmod p$, $s = x + kr \bmod q$.

(2) Sending the secret information to the proxy signer: The original signer sends (r, s) to the proxy signer.

(3) Checking the secret information: The proxy signer checks the validity of (r, s) using the next equality:

$$g^s = yr^r \bmod p$$

If the equality holds, the proxy signer accepts (r, s) as the valid proxy secret key. If not, the protocol stops.

(4) Signing
The proxy signer signs a message m, then its signature S_p is generated. After that, the proxy signer sends the message and its signature, which are (m, S_p, r), to the verifier.

(5) Verification
Upon receiving (m, S_p, r), the verifier recovers y' by $y' = yr^r \bmod p$ and substitutes y' for y. After that, the verifier performs the verification phase of the normal signature scheme.

4. Description of a Self Proxy Signature Scheme

We now present a self proxy signature scheme. Our scheme is based on the normal proxy signature scheme in which a signer Alice delegates her signing capability to herself recursively. We assume that signer Alice has her original signing private/public key pair (x_a, y_a). We skip the setup phase.

4.1 Generating Self Proxy Singing Key Pair

The signer Alice generates the temporary self proxy private/public key pair by using her original signing key pair (x_a, y_a) as follows:

(1) The signer Alice chooses $k, x_t \in_R Z_q^*$ randomly, and computes $r = g^k \bmod p$, $y_t = g^{x_t} \bmod p$.

(2) The signer Alice computes $x_p = k + (x_a + x_t)H(m_w) \bmod q$, and sets her temporary self proxy private/public keys by x_p, $y_p = g^{x_p} \bmod p$.

(3) The signer Alice publishes y_t.

4.2 Signing

To generate a self proxy signature for the verifier, the signer Alice performs the following operations.

(1) The signer Alice chooses $k' \in_R Z_q^*$ randomly, and computes for the following equations:

$$r' = g^{k'} \bmod p \qquad (1)$$

$$s' = k' + x_p H(m) \bmod q \qquad (2)$$

(2) The signer Alice sends $(m, (r', s'), r, m_w)$ to the verifier Bob.

4.3 Verification

First of all, the verifier Bob checks the signer's identity and the delegation lifetime of the warrant m_w. If all validations hold, the verifier Bob follows the next operations.

(1) The verifier Bob recovers the self proxy public key y_p as follows:

$$y_p = r(y_a y_t)^{H(m_w)} \bmod p$$

(2) The verifier Bob checks the validity of the next equality. If the equality holds, the verifier Bob accepts (r', s') as the valid self proxy signature.

$$g^{s'} = r' y_p^{H(m)} \bmod p \qquad (3)$$

This is because of the following:

$$\begin{aligned} g^{s'} &= g^{(k' + x_p H(m))} \bmod p \\ &= g^{k'} (g^{x_p})^{H(m)} \bmod p \\ &= r' y_p^{H(m)} \bmod p \end{aligned}$$

5. Analysis of the Proposed Scheme

Theorem 1. The self proxy signature scheme satisfies the unforgeability property.

Proof. The proposed scheme is based on assumption 1, the discrete logarithm problem (DLP). The security analysis about unforgeability is as follows:

We assume that an attacker, Cindy, tries to forge the self proxy signature. Cindy can attack our scheme in two ways. In the first way, Cindy computes for the self proxy private key x_p, and in the second way, Cindy forges the valid self proxy signature without the self proxy private key.

In the first way, the attacker Cindy has to compute x_p from y_p or generate x_p using Equation (1), (2) and the information $(m, (r', s'), r, m_w)$ that is transferred between A and B. However, it is computationally hard to compute x_p from y_p or $(m, (r', s'), r, m_w)$ because it is a DLP. Therefore, it is computationally difficult for the attacker Cindy to compute x_p.

In the second way, the attacker Cindy has to forge the valid signature (r', s') on the message m without the private key x_p. In Equation (3), because y_p, m, and p are public information, (r', s') is the unknown. But, s' is an exponent. Therefore, this is also the DLP.

Consequently, because the two attacks are not possible, it is computationally difficult for the attacker Cindy to forge the self proxy signature. Therefore, the proposed scheme satisfies the unforgeability property.

Theorem 2. The self proxy signature scheme satisfies the undeniability property.

Proof. As to the property of undeniability, it implies that the signer cannot deny the valid message and its signature. In the proposed scheme, when the self proxy signature $(m, (r', s'), r, m_w)$ is verified, the warrant m_w is checked, and the signer's public key, y_a, and the public information y_t are used in the verification phase. The signer cannot deny the valid message and its signature. Therefore, the proposed scheme satisfies the undeniability property.

Theorem 3. The self proxy signature scheme satisfies the distinguishability property.

Proof. In the proposed scheme, when the self proxy signature $(m,(r',s'),r,m_w)$ is verified, the signer's public key and identity are used in the verification phase; therefore, we can consider it as a self proxy signature and not a normal signature. Thus, anyone can distinguish the self proxy signature from normal signatures. The proposed scheme thus satisfies the distinguishability property

Theorem 4. The self proxy signature scheme satisfies the verifiability property.

Proof. The security property implies that from the self proxy signature, a verifier can be convinced of the signer's agreement on the signed message. In the proposed scheme, on one hand, from the warrant m_w, the verifier can identify who the signer is. On the other hand, when the self proxy signature is verified, the signer's public key and identity are used in the verification phase. Thus, any verifier can be convinced of the signer's agreement on the signed message, and he/she can verify the message and its signature. Furthermore, the verifier who wants to verify the message-signature can recover proxy public key y_p by public information. Therefore, the proposed scheme satisfies the verifiability property.

6. Conclusion

In this paper, we propose a new type of proxy signature, the self proxy signature scheme. Using this scheme, a signer can have multiple temporary private/public keys and use them simultaneously. Our signature scheme is secure because it satisfies all the security properties: Unforgeability, Undeniability, Distinguishability, and Verifiability.

References

[1] M. Mambo, K. Usuda and E. Okamoto. "Proxy Signatures: Delegation of the Power to Sign Message", IEICE Trans. Fundamentals, Vol. E79 A, No. 9, 1996

[2] M. Mambo, K. Usuda, E. Okamoto. Proxy signatures for delegating signing operation. In: 3rd ACM Conference on Computer and Communications Security(CCS'96), pp. 48-57. New York: ACM Press, 1996

[3] G. Wang. Designated-Verifier Proxy Signature Schemes. In: Security and Privacy in the Age of Ubiquitous Computing (IFIP/SEC 2005), pp. 409-423. Springer, 2005.

[4] Z. Tan, Z. Liu, C. Tang. Digital Proxy Blind Signature Schemes Based on DLP and ECDLP. In: MM Research Preprints, No. 21, MMRC, AMMS, Academia Sinica, Beijing, 2002, pp. 212-217

[5] S. Lal, A. K. Awasthi. Proxy Blind Signature Scheme. In: Journal of Information Science and Engineering. Cryptology ePrint Archive, Report 2003/072. Available at http://eprint.iacr.org/.

[6] Guilin Wang, Feng Bao, Jianying Zhou, and Robert H. Deng. Security Analysis of Some Proxy Signatures. In: Information Security and Cryptology - ICISC 2003, LNCS 2971, pp. 305-319. Springer-Verlag, 2004.

Young-Seol Kim received the B.S. and M.S. degrees in Computer Science and Engineering from Sogang University. He is currently a Ph.D. candidate at Sogang University, Korea. His research interests include cryptography and information security.

Jik Hyun Chang received the B.S. and M.S. degrees in Mathematics from Seoul National University, Korea. He received his Ph.D degree in the department of Computer Science & engineering from University of Minnesota, USA. Since 1986, he serves as a professor at Sogang University, Korea. His research interests include algorithms design and analysis, and cryptographic algorithms.