一次性代理簽章方法

One Time Proxy Signature

吳宗杉
台灣海洋大學
ibox456@gmail.com

鍾國文
佛光大學
kuowen@seed.net.tw

丁培毅
台灣海洋大學
pyting@mail.ntou.edu.tw

陳益森
台灣海洋大學
ys3889@gmail.com

摘要

數位簽章是可用以滿足不可否認性及確認資料來源性的安全技術，但當原始簽署者如因某些因素無法親自簽署文件，則可將其簽署權利授與代理人，此即所謂的代理簽章。在本論文中，我們考慮一次性的特性在代理簽章環境下的應用需求，即，某些特定環境下，此代理權利只能使用一次的解決方案。如果代理簽章者重覆簽署，則可利用兩份代理簽章揭露其私鑰。

關鍵詞：數位簽章、代理簽章、一次性簽章、離散對數問題

Abstract

Digital signatures meet the requirements of non-repudiation and data origin. In case that the original signer cannot perform some signing tasks, he can delegate his signing ability to the proxy signer, which is so called proxy signatures. In this paper, we elaborate on the environments of one time property to propose a solution for one time proxy signature. That is, the proxy signer can only produce one proxy signature. If there are two proxy signatures are made, the private key of the proxy signer will be exposed.

Keywords: Digital signature, Proxy signature, One time signature, Discrete logarithm problem

一、緒論

1996 年，Mambo 等人提出一個新的代理簽章 (Proxy signature) 概念 [5]，代理簽章是原簽署者因某些因素無法親自簽署文件，而將簽署權力授與代理人，但是其效力如同原始簽章者簽署一樣。原始簽章者授予代理簽章者簽章的能力，代理簽章者簽署後，送給驗證者檢驗，驗證者可確認此簽章的合法性及正確性。他們所提出的方法，代理簽章者可以否認其所簽署過的代理簽章，並不具有不可否認的特性。因此，為改進這個缺點，Mambo 等人於同年又提出具有不可否認性的方法 [6]。一般而言，代理簽章滿足數位簽章的兩個特性，一為不可偽造性，另一為可驗證性。所謂的不可偽造性是指除原始簽章者或代理簽章者外，沒有任何人可以產生有效代理簽章。可驗證性是指當驗證者驗證代理簽章無誤時，就可以相信此簽章是合法的代理簽章。其代理授權種類如下：

完全授權 (Full delegation)：原始簽章者將其所擁有之秘密金鑰交付予代理簽章者，代理簽章者可依此秘密金鑰簽章簽署文件。因為代理簽章者得知原始簽章者之秘密金鑰後，便可任意簽署文件，且所產生之簽章則無法分辨是由原始簽章者亦或是代理簽章者所簽署。

部份授權 (Partial delegation)：基於上述授權方式之缺點，原始簽章者授予代理者簽署權力

時，並不是直接傳送其秘密金鑰給代理者，而是利用此秘密金鑰計算出代理金鑰，將此代理金鑰傳送給代理者。 可分為非保護代理(Proxy unprotected) [9]，保護代理(Proxy protected)，說明如下，代理者則可用代理金鑰簽署文件產生代理簽章。其所產生的代理簽章與原始簽章者利用秘密金鑰所產生的簽章是不同的，因此驗證者可分辨出簽章是否為代理簽章。此外此種授權方式可依「代理的保護」細分為兩類。一為非保護代理之代理簽章，即除了代理者可產生代理簽章外，原始簽章者亦握有代理金鑰來產生代理簽章。因此所產生出之代理簽章無法分辨是否真正由代理者所產生之簽章。這對代理者而言是不公平的，倘若原始簽章者產生代理簽章，此行為若是惡意的話，代理簽章者就被認定為是此簽章的簽署者，因而必須對此簽章負責。為解決這個問題，讓原始簽章者與代理簽章者保有獨立的簽署權力，另一種授權方式則為保護代理之代理簽章。當原始簽章者欲授予代理簽章者簽署權力時，首先，依據其秘密金鑰產生數值，將此數值秘密傳送給代理簽章者，代理簽章者再利用自己所擁有之秘密金鑰與所接收的數值計算之後，產生出的代理金鑰。此代理金鑰才可產生代理簽章。由於代理金鑰是由代理簽章者的秘密金鑰所計算出的，唯此擁有此金鑰的人方可得知代理金鑰之後產生的代理簽章。因此，除了代理簽章者，沒有任何人 (包括原始簽章者) 可以產生代理簽章。這種方式所產生的代理簽章使得簽章具有可追蹤性及不可否認性，讓簽署者必須為其所簽署的文件負責。

授權書授權 (Delegation by warrant)：原始簽章者產生授權書給代理簽章者，此授權書是利用原始簽章者的秘密金鑰所簽署產生，除了宣告某代理簽章者可代理其行使簽署的權力外，尚可包含一些特殊的宣告，如代理的權限、可簽署文件的類型等皆可包括在此授權書中。代理簽章者得到此授權書後，即可利用自己的秘密金鑰簽署所代簽的文件，並將此授權書包含於簽章之中 [7]。

結合授權書之部份授權 (Partial delegation with warrant)：在 1997 年，Kim 等人提出新的代理簽章方法 [3]，即結合部分授權及授權書授權的授權類型。原始簽章者先設定好該代理簽章者之簽署權力，如簽署期限、簽署文件類型等，之後利用其秘密金鑰連同所規範的簽署權力計算之後所得的結果，將之交付給代理簽章者，代理簽章者再依據所接收到的訊息，利用自己的秘密金鑰，計算而得到代理金鑰。之後，代理簽章者即可利用此代理金鑰簽署文件產生代理簽章。而所產生的代理簽章亦包含原始簽章者所規定的簽署權力，唯有完全符合此簽署權力的代理簽章方可認定為合法的代理簽章 [10]。

門檻式授權 (Threshold delegation)：原始簽章者授權給一個包含 n 人的代理群體簽章，只有當此代理簽章群體中任 t 人以上共同合作，才能產生有效的代理簽章，但是門檻式授權另外有它一個特性 [11]，那就是驗證者並不知道有那些成員參與簽署。由於門檻值數位簽章並不需要群體內的所有成員皆參與，且驗證者並不知道有那些成員參與簽署，因此造成發生爭議時，無法得知是群體中那些成員實際參與簽署文件。為提供不同之應用，故有所謂可追蹤簽署者 (Traceable signer) 之門檻式數位簽章法 [3]。也就是說，驗證者在驗證門檻值群體簽章時，亦可同時知道有那些成員實際參與簽署文件。

本文主要探討的是 Kim 與 Chang 所提出一次代理簽章方法 [2]，其研究內容提出代理者所欲代理原始簽章者簽署的能力，僅是將其量化，透過簽署過程傳達欲簽署一次或多次的量來當代理簽章次數，與其傳統代理簽章方法並無明顯不同之處。本論文提出如何讓代理者僅能代理簽章一次的方法，來改善 Kim 與 Chang 所提出一次代理簽章方法，不能滿足不可否認性的缺點。

二、Kim 與 Chang 一次代理簽章方法

Kim 與 Chang 使用授權書的一次代理簽章方法，是當原始簽章者 Alice 與代理簽章者 Bob 共同地產生代理簽章的金鑰對 (x_p, y_p) ，而驗證簽章者可以還原代理簽章金鑰對應之公鑰 y_p 來驗證其真實性及確認合法性。其方法可分以下五個階段，詳述如下：

系統初始階段：系統選取大質數 p 與 q，滿足 $q \mid p-1$，以及序 (Order) 為 q 的生成子 (Generator)。(x_u, y_u) 為參與者 U 的私鑰與公鑰對，ID_u 則為其身分識別，另外定義單向雜湊函數 (One-way hash function) $H(.)$及原始簽章者簽章函數 $Sig\ (.)$。

代理授權階段：首先，Alice 與 Bob 共同執行如下的步驟：

1. Alice 挑選兩個數 k_A, k_1，計算下列各式：

$$r_A = g^{k_A} \bmod p \tag{1}$$

$$r_1 = g^{k_1} \bmod p \tag{2}$$

$$c = H(r_A) \tag{3}$$

然後將 c 傳送給 Bob。

2. Bob 也挑選 k_B，並且計算

$$r_B = g^{k_B} \bmod p \tag{4}$$

回傳 (c, r_B) 給 Alice。

3. Alice 收到 (c, r_B) 後，檢查 $r_B^q = 1 \bmod p$ 等式是否為真，倘若為真，則計算下列各式：

$$r_P = r_A\, r_B \bmod p \tag{5}$$

$$s_A = k_A + x_A\, H(m_w \parallel r_P) \bmod q \tag{6}$$

其中 ‖ 為連結符號，而這裡所稱的授權書是：

m_o = [授權期限，ID_A 與 ID_B，訊息， r_1]，

$m_w = (m_o, Sig\,[m_o])$。

最後，將 (r_A, s_A, k_1, m_w) 安全的傳送給 Bob。

4. Bob 接收到 (r_A, s_A, k_1, m_w)，首先計算下列兩式：

$$r_P = r_A\, r_B \bmod p \tag{7}$$

$$c = H(r_A) \tag{8}$$

然後驗證下列等式

$$r_A^q \stackrel{?}{=} 1 \bmod p \tag{9}$$

$$g^{s_A} \stackrel{?}{=} y_A^{H(m_w \| r_P)} r_A \bmod p \tag{10}$$

倘若 (10) 式為真，則 Bob 將計算下列式子

$$s_B = k_B + x_B\, H(m_w \parallel r_P) \bmod q \tag{11}$$

最後則產生出代理簽章金鑰對 (x_p, y_p)，如下兩式：

$$x_p = s_A + s_B \bmod q \tag{12}$$

$$y_P = g^{x_P} \bmod p \tag{13}$$

簽章階段：因為 Bob 僅能使用 r_1 去簽署文件一次，所以 Bob 計算下列兩式：

$$r_1 = g^{k_1} \bmod p \tag{14}$$

$$s = k_1 + x_P H(m \| m_w \| r_1) \bmod q \tag{15}$$

最後將產生的代理簽章 $\sigma = (m, r_P, m_w, r_1, s)$ 傳給驗證者。

驗證階段：驗證者接到訊息後，首先檢查 Alice 與 Bob 的身分別與其授權書，接著還原 Bob 的代理簽章金鑰對應之公鑰 y_p，且驗證 Bob 所簽署的簽章是否為一次，步驟如下：

1. 檢查 ID_A 與 ID_B。
2. 檢查 m 與 m_w。
3. 還原 Bob 的代理簽章金鑰對應之公鑰如下：

$$y_P = (y_A y_B)^{H(m_w \| r_P)} r_P \bmod p \tag{16}$$

4. 最後驗證者驗證其代理簽章的合法性?

$$g^s \stackrel{?}{=} y_P^{H(m \| m_w \| r_P)} r_1 \bmod p \tag{17}$$

當驗證者從授權書檢核與簽章驗證式中，得到相同且僅有的 r_1 參數值時，即可證明 Bob 僅能代理簽章一次。

擴張簽章次數： Kim 與 Chang 方法中，倘若 Alice 允許 Bob 可以簽署多次文件，擴大

其簽章的次數，首先授權書定義如下：

m_o = [授權期限，ID_A 與 ID_B，訊息描述，$r_1, r_2, r_3,\ldots$]，

$m_w = (m_o, Sig_{Alice}[m_o])$.

也就是 Alice 得將 $r_1, r_2, r_3,\ldots, r_n$ 等參數放於授權書中後，經過指數運算後獲得其相對應之值 $k_1, k_2, k_3, \ldots, k_n$ ，並透過安全的方式傳給 Bob，讓 Bob 得以簽署過程時加入此參數值，最後，驗證者從授權書檢核中與簽章驗證式確認後，可以得知 Alice 授與 Bob 所可以代理簽章的次數。

在 Kim 與 Chang 的方法中，我們發現 Alice 可以偽造 Bob 的代理簽章 $\sigma = (m, r_p, m_w, r_1, s)$，因為在 (5) 與 (7) 中，Alice 與 Bob 在代理授權階段利用金鑰交換方法，產生代理簽章金鑰 r_p ，因此 Alice 與 Bob 兩者都可使用此一簽章金鑰來做代理簽章時使用，這樣一來 Alice 若要偽裝 Bob 行使代理簽章任務，驗證者並無法確定是哪位所簽署之簽章，無法滿足不可否認性。

三、一次性代理簽章

我們的方法中，授權方式過程為保護代理之代理簽章，也就是當原始簽章者 Alice 欲授予代理者 Bob 簽署權力時，Bob 可依據此方法，來向 Alice 證明，依據其私鑰產生數值，再將此數值透過此方法秘密傳送給 Alice，在整個求證過程中，Alice 不知道 Bob 的私鑰，Bob 利用自己所擁有之私鑰與所接收到 Alice 私鑰所產生的數值，共同產生代理金鑰，唯此擁有此金鑰的人方可得知代理金鑰產生的代理簽章。Bob 產生有效代理簽章後，送指定或無指定驗證者來驗證其簽章的合法性。當發生糾紛或被簽署文件超過一次時，驗證者可以從代理簽章中揭露 Alice 所授予 Bob 的代理簽章金鑰，以證明 Bob 越權。本文可分為系統初始階段、授權代理階段、代理簽章階段 (指定或無指定第三者驗證) 、簽章驗證階段等四個階段。敘述如下：

系統初始階段：我們的方法分四個階段，系統初始階段與 Kim 與 Chang 的方法相同

授權代理階段：首先，Alice 與 Bob 利用零知識證明法共同執行如下程序 [1]：

1. Bob 擁有公開數值組合 (g_1, g_2, x, y)，其中 $g_1, g_2 \in Z_p$，p 與 q 是大質數且滿足 $q|(p-1)$，$k_B \in_R Z_P^*$ 則下列關係式為：

$$x = g_1^{k_B} \pmod p \tag{18}$$

$$y = g_2^{k_B} \pmod p \tag{19}$$

2. Bob 計算下列三式：

$$r_B = g^{k_B} \bmod p \tag{20}$$

$$a = g_1^{r_B} \bmod p \tag{21}$$

$$b = g_2^{r_B} \bmod p \tag{22}$$

然後將 (a, b) 傳給 Alice。

3. Alice 收到 (a, b) 後，任選 $k_A \in_R Z_P^*$ 並計算

$$r_A = g^{k_A} \bmod p \tag{23}$$

然後將 r_A 傳給 Bob。

4. Bob收到r_A後，計算下列式子：

$$w = r_B + k_B \cdot r_A \bmod q \tag{24}$$

再回傳 w 給 Alice。

5. Alice 收到 w 後，透過下列兩驗證式：

$$g_1^w \stackrel{?}{=} a \cdot x^{r_A} \tag{25}$$

$$g_2^w \stackrel{?}{=} b \cdot y^{r_A} \tag{26}$$

得知 Bob 真的擁有 k_B 。

6. Alice 確認無誤後，則計算下列算式：

$$s_A = k_A + x_A H(m_w \| r_P) \bmod q \tag{27}$$

再將 (s_A, m_w) 傳送給 Bob，其中授權書於指定或無指定驗證者時，分述如下：

I. 無指定驗證者

m_o = [授權期限，ID_A 與 ID_B，訊息]，

$m_w = (m_o, Sig[m_o])$。

II. 指定驗證者

m_o = [授權期限，ID_A、ID_B 與 ID_c，訊息]，

$m_w = (m_o, Sig[m_o])$。

7. Bob 接收到 (r_A, s_A, m_w)，首先驗證下列等式：

$$g^{s_A} \stackrel{?}{=} y_A^{H(m_w \| r_A)} r_A \bmod p \quad (28)$$

驗證完 s_A 的合法性，倘若為真，則計算下列兩式：

$$r_P = r_A\, r_B \bmod p \quad (29)$$

$$s_B = k_B + x_B\, H(m_w \| r_P) \bmod q \quad (30)$$

最後，由兩人共同產生出 Bob 的代理簽章金鑰對 (x_p, y_p) 如下 ：

$$x_p = s_A + s_B \bmod q \quad (31)$$

$$y_P = g^{x_P} \bmod p \quad (32)$$

代理簽章階段：Bob產生代理簽章傳給驗證者時，我們的方法可分指定或無指定驗證者，分述如下：

I. 無指定驗證者

當無指定驗證者時，直接執行計算式產生

$$s = k_B + x_P H(m \| m_w \| r_B) \bmod q \quad (33)$$

最後得到代理簽章 $\sigma = (m, r_P, m_w, r_B, s)$，並將這代理簽章 σ 傳給任一驗證者。

II. 指定驗證者

這個方法是指 Bob 產生完有效的簽章後，簽章僅能傳給指定的驗證者做驗證，Bob 執行以下列兩個計算式：

$$c = y_c^{k_B} \bmod p \quad (34)$$

$$s = k_B + x_p H(m \| m_w \| c) \bmod q \quad (35)$$

最後，得到代理簽章 $\sigma = (m, r_P, m_w, c, s)$，並將這代理簽章 σ 傳給指定之驗證者。

驗證階段：當驗證者 (指定或無指定) 收到 Bob 產生的代理簽章 σ 時，首先檢查 Alice 與 Bob 的身分及其授權書，接著還原 Bob 的代理簽章金鑰對應之公鑰 y_p，並且驗證 Bob 所簽署文件的合法性，分述如下：

I. 無指定驗證者

驗證者收到簽章後檢查 m_w ，還原 y_p (36) 式，並驗證 (37) 式：

$$y_p = (y_A y_B)^{H(m_w \| r_P)} r_A r_B \quad (36)$$

$$g^s \stackrel{?}{=} y_P^{H(m \| m_w \| r_B)} \cdot r_B \quad (37)$$

倘若 (37) 式為真，則簽章 $\sigma = (m, r_P, m_w, r_B, s)$ 為有效的。

我們提出一次性代理簽章方法，也就是 Bob 僅能代理簽章一次的情況下，如何驗證出 Bob 有無越權產生第二次簽章，說明如下：

當 Bob 代理第一次簽章時，則代理簽章為 $\sigma_1 = (m_1, r_P, m_w, r_B, s_1)$，其中

$$s_1 = k_B + x_P H(m_1 \| m_w \| r_B) \quad (38)$$

倘若 Bob 欲再代理簽章時，則代理簽章為 $\sigma_2 = (m_2, r_P, m_w, r_B, s_2)$，其中

$$s_2 = k_B + x_P H(m_2 \| m_w \| r_B) \quad (39)$$

Bob 於每次代理簽章後，傳送給驗證者，驗證者可從 (38) 及 (39) 兩式算出 x_p，即揭露 Bob 的代理簽章金鑰 x_P，也就是說，驗證者確認此代理簽章的合法性外，也可以驗證 Bob 在代理期間有無超過簽章一次，讓 Bob 無法否認自己越權。

II. 指定驗證者

被指定的驗證者收到簽章後檢查 m_w，並計算下列驗證式：

同 (36) 還原 y_p，

$$(g^{s} y_P^{-H(m\|m_w\|c)})^{x_c} \stackrel{?}{=} c(= y_C^{k_B}) \quad (40)$$

倘若為真，則可證明指定驗證者，且為 Cindy。在這，我們提出一次性代理簽章方法，也就是 Bob 僅能代理簽章一次的情況下，如何驗證出 Bob 有無越權產生第二次簽章，說明如下：

當 Bob 代理第一次簽章時，則代理簽章為 $\sigma_1 = (m_1, r_P, m_w, c, s_1)$，其中

$$s_1 = k_B + x_P H(m_1 \| m_w \| c) \quad (41)$$

倘若 Bob 欲再代理簽章時，則代理簽章為 $\sigma_2 = (m_2, r_P, m_w, c, s_2)$，其中

$$s_2 = k_B + x_P H(m_2 \| m_w \| c) \quad (42)$$

Bob 於每次代理簽章後，傳送給驗證者，可從(41)及(42)兩式算出 x_p，即揭露 Bob 的代理簽章金鑰 x_P，也就是說驗證者確認此代理簽章的合法性外，也可以驗證 Bob 在代理期間有無超過簽章一次，讓 Bob 無法否認自己越權。

四、安全性分析

我們所提出的一次性代理簽章方法，是植基於 DLP (Discrete logarithm problem) 與 CDH (Computational Diffie-Hellman assumption) 的密碼假設。為證明一次性代理簽章方法可以滿足鑑別可區別性、不可偽造性、可驗證性、可識別性與不可否認性等五項安全需求。我們將從被簽章的訊息、原始簽章者之私鑰、代理者私鑰、代理原始簽章者之私鑰以及驗證者之私鑰來一一探討。

可區別性 (Distinguishability)：由代理者所代理的簽章 $\sigma = (m, r_p, m_w, r_B, s)$，與原始簽章者所產生的簽章中 (r_A) 是可區別的，不僅是原始簽章者，連代理簽章者的身分別與公鑰都在驗證式子裡被引用來做驗證，因此滿足可區別性。

不可偽造性 (Unforgeability)：唯有原始簽章者及代理簽章者可產生有效之簽章，除此之外，沒有任何人能夠偽造出合法的代理簽章。

我們的方法，具有代理保護特性，在產生代理簽章金鑰時，利用零知識證明法產生代理簽章金鑰，在不需要透露任何密秘訊息下，亦可達到訊息交換的效果，來保護代理簽章者，因此原始簽章者無法偽裝任何代理的簽章，因為原始簽章者偽造的代理簽章無法通過以下的驗證式 $g^{s} = g^{k_B + x_P H(m\|m_w\|r_B)} = y_P^{H(m\|m_w\|r_B)} r_B$，因此滿足不可偽造性。

可驗證性 (Verifiability)：從代理簽章中，驗證者可以相信原始簽章者同意此份簽章，從授權書 m_w，驗證者可以得知誰是原始簽章者，誰是代理簽章者；甚至當代理簽章 $\sigma = (m, r_p, m_w, r_B, s)$ 產生後，不僅是原始簽章者，連代理簽章者的身分別與公鑰都在驗證式子裡被引用來做驗證。因此原始簽章者不能否認有授權簽章的能力給指定的代理者，滿足可驗證性。

可識別性 (Identifiability)：從代理簽章中，原始簽章者可知道代理簽章者的身分，從當初所簽的授權書 m_w 中即可容易的確定代理者的身分別，因此滿足可識別性。

不可否認性 (Non-repudiation)：代理簽章者不可否認自己的代理簽章，當代理者產生代理簽章 $\sigma = (m, r_p, m_w, r_B, s)$ 可被驗證時，這裡的授權書 m_w 是會被檢核的且原始簽章者與代理簽章者的公鑰在驗證式還原回來，代表原始簽章者與代理簽章者都不能否認此事，因此滿足不可否認性。

五、結論

數位簽章經常被用於簽署各式各樣的電子文件。簽署者必須利用自己所擁有的私鑰簽署文件，驗證則利用簽署者之公鑰以決定簽章的合法

性，達到簽署文件之數位簽章具有效性及不可否認性。

我們所提一次性代理簽章，代理簽章金鑰的產生需要原始簽章者的簽章，與本身的私鑰計算，以用來產生屬於代理簽章者本身的代理簽章，於驗證階段，驗證者可以對代理簽章進行驗證其正確性，並可以分辨出原始簽章簽署者與代理簽章簽署者的身分。

Kim 與 Chang 所提方法，必須事先決定代理者欲簽章的次數及參數，如要擴大簽章次數，原始簽章者需將欲簽署的次數放在證書中，與一般代理簽章種類無不同之處，僅是量化而已，卻無法有效的避免被代理者濫用。因此我們提出改進的方法，除運用零知識證明方法來保護代理者的私鑰，滿足安全性需求；並運用簡易數學方程式的方法運算求證，僅限於代理一次簽章來限制代理者，更有效率，也更明確來限定代理簽章者僅能利用代理簽章來簽署一次文件。

我們運用簡易數學方程式的方法，運算求證來建構一次性代理簽章。於未來研究上，希望能以更精簡的計算方法來建構門檻式一次性不可否認代理簽章方法，並與其他門檻式代理簽章方法做進一步的探討。

參考文獻

[1] S. Glodwasser, S. Micali and C. Rackoff, “The knowledge complexity of interactive proof systems,” Proceedings of the 17th Annual ACM Symposium on Theory of Computing, pp. 291-304, 1985.

[2] Y.S Kim and H.H Chang, “New one-time proxy signature scheme based on DLP using the warrant,” International Journal of Computer Science and Network Security, Vol. 7, No. 2, pp. 215-230, February 2007.

[3] S. Kim, S. Park and D. Won, “Proxy signature, Revisited,” International Conference and Information Communication Security, Beijing, China, pp. 223-232, 1997.

[4] L. Lamport, “Password authentication with insecure communication,” Communications of the ACM, Vol. 24 No. 11, 1981, pp. 770-772.

[5] M. Mambo, K. Usuda and E. Okamoto, “Proxy signatures for delegating signing operation,” Proceedings of 3rd ACM Conference on Computer and Communication Security, New Delhi, pp. 48-57, 1996.

[6] M. Mambo, K. Usuda and E. Okamoto, “Proxy signatures: Delegating of the power to sign messages,” IEICE Transactions on Fundamentals, Vol. E-79-A, No. 9, pp. 1338-1354, 1996.

[7] B.C. Neuman, “Proxy-based authorization and accounting for distributed system”, Proceedings of the 13th International Conference on Distributed System, pp. 283-291, 1993.

[8] M.O. Rabin, “Digitalized signatures,” Foundations of Secure Communication, Academic Press, 1978, 155-168.

[9] Z. Shao, “Proxy signature schemes based on factoring,” Information Processing Letters, Vol. 85, No. 3, pp. 137-143, 2003.

[10] T.S. Wu and C.L. Hsu, “Cryptanalysis of group-oriented (t, n) threshold digital signature schemes with traceable signers,” Computer Standards and Interfaces, Vol. 26, pp. 477-481, 2004.

[11] T.C. Wu, T.S. Wu and C.L. Hsu, “New nonrepudiable threshold proxy signature scheme with known signer”, Journal of Systems and Software, Vol. 58, No. 6, pp. 119-124, 2001.