安全的 Paillier 代理簽章機制

丁培毅 徐嘉煇 黃曉偉

國立台灣海洋大學資訊工程學系

{pyting,m93570035,m94570010}@mail.ntou.edu.tw

摘要

1996 年日本學者 Mambo 等人[1]提出代理簽章(proxy signature)的概念之後，代理簽章技術被廣泛應用在電子商務應用上。大部份的代理簽章機制都是基於解離散對數難題，近幾年也有其它基於不同假設的代理簽章機制[2][3]相繼被提出。由於 Paillier 公開金鑰密碼系統[4]則是基於解合成數模數 n 次方根難題，而且 Paillier 密碼系統應用在電子投票與電子投標上有其不可取代的優點，所以本研究特別針對 Paillier 密碼系統，提出兩種代理簽章機制：第一種是無保護代理簽章者(proxy unprotected)的代理簽章機制，第二種則是有保護代理簽章者(proxy protected)的代理簽章機制，兩者的安全性都是基於解合成數模數 n 次方根難題(composite residuosity class)。

關鍵詞：*代理簽章、離散對數、解合成數模數 n 次方根、Paillier 公開金鑰密碼系統*

Abstract

Because of the property of additive homomorphism, the Paillier public-key cryptosystem is advantageous when employed in designing the tallying or comparison mechanisms of electronic voting or electronic auction systems. In the physical deployment of these systems, it could happen frequently that people, who cannot attend an event personally, would like to participate through a proxy. In this paper, two proxy-signature schemes based on the Paillier cryptosystem are proposed. The first one is a proxy-unprotected scheme and the second one is a proxy-protected scheme. The original signer of the first scheme retains the ability to issue a valid proxy signature while the original signer of the second one cannot. The security of both systems are based upon the intractability of the composite residuosity class problem.

Keyword: *proxy signature, composite residuosity assumption, Paillier public-key cryptosystem*

1. 緒論

在現實生活中，如果要證明某個文件是由自己發出的，或者要向其他人證明自己對文件負責，一般來說會在文件上親筆簽名或蓋上自己的印章來證明。然而在數位化資訊

中，必須採用數位簽章的方式，才能證明電子文件簽章者的身份以及電子文件內容的真偽。

公開金鑰密碼系統的概念使得數位簽章的技術能夠實現。在公開金鑰密碼系統中，每一個使用者都有一把密鑰(private key)與一把公鑰(public key)。每一把密鑰都是獨一無二的，且會被使用者視為秘密，只有使用者自己才知道;而公鑰則會被使用者公佈出去。根據上述特性，使用者可以利用密鑰對電子文件產生數位簽章，而驗證者則可以利用使用者公佈的公鑰來驗證數位簽章，許多數位簽章都具有不可偽造與不可否認的性質。

在實際應用中，當某人因出差或放假而無法簽署文件時，常常需要一個代理人來幫他簽署文件，因此在 1996 年，Mambo 等人率先提出代理簽章的作法。代理簽章系統允許原始簽章者(original signer)授權一個代理簽章者(proxy signer)，代表原始簽章者來簽署文件。基本有三種代理簽章的授權方式，分別是完全授權(full delegation)、部份授權(partial delegation)[5]和委託書授權(delegation by warrant)[6]。三種授權方式中，部份授權方式具有較多優點，完全授權方式無法提供原始簽章者足夠的安全性，而委託書授權方式在驗證時比較耗費時間。

部分授權的方式中，依據原始簽章者是否可以產生跟代理簽章者一樣的代理簽章，可以再分成兩種代理簽章，分別是無保護代理簽章者(proxy unprotected)的代理簽章和有保護代理簽章者(proxy protected)的代理簽章。前者只使用原始簽章者給的代理簽章鑰匙來產生代理簽章，所以原始簽章者也可以產生合法的代理簽章，如此無法分辨出一個代理簽章是由誰簽的，一旦產生爭議，無法釐清責任的歸屬，只要能使用原始簽章者的公鑰和授權書驗證，代理簽章就是有效的，原始簽章者必須承擔所有責任。因此，在使用此種代理簽章時，原始簽章者必須謹慎挑選信賴的人作為他的代理人，這種方法比較適用於一些不重要的文件簽署，例如一般電子郵件。在有保護代理簽章者的代理簽章機制中，簽署文件時，代理簽章者不只需要原始簽章者給的代理簽章鑰匙，也需要自己的密鑰，因此任何人(包括原始簽章者)無法偽造代理簽章。在驗證的時候，需要同時使用原始簽章者的公鑰和代理簽章者的公鑰來驗證代理簽章，所以代理簽章者不能否認自己簽過的代理簽章，必須與原始簽章者一同承擔所有責任。此種代理簽章機制適用於重要文件的簽署，例如銀行支票的批准，帳單的簽收等等。

公開金鑰密碼技術的概念自從 Diffie 與 Hellman[7]揭露之後，陸續出現多種密碼系統[4][8][9][10]，其中 Paillier[4]提出的機率式公開金鑰密碼系統，因為具有加法同型性質(additive homomorphism)，經常應用來設計各式電子投票與投標系統，在這些應用系統中時常會有無法出席而需要代理人的需求。

第二節簡介 Paillier 機率式公開金鑰密碼系統，第三節與第四節分別介紹無保護以及有保護代理簽章者的 Paillier 代理簽章機制，第五節中分析所提出方法的安全性，第六節為結論。

2. Paillier 機率式公開金鑰密碼系統

1999 年 Paillier 提出了一種機率式公開金鑰密碼系統 [4]，其安全性是基於解合成

數模數 n 次方根難題。因為其具有加法同型性質，即 $D(E(m_1)E(m_2) \bmod n^2) = m_1 + m_2$，所以常被用來設計電子投票或電子投標系統。相較於 ElGamal 密碼系統，Paillier 密碼系統應用在電子投票上，並不會遭遇到解離散對數的問題 [11]；相較於同樣使用合成數模數的 RSA 密碼系統，Paillier 密碼系統則提供了機率式加密的優點。

以下簡單描述 Paillier 加解密機制與簽章機制。令 p 和 q 為兩個大質數且 $n = p \cdot q$，$\lambda(n) = \text{lcm}(p-1, q-1)$為 p-1 與 q-1 的最小公倍數，其中 p、q 和 $\lambda(n)$必須滿足 $\gcd(p \cdot q, \lambda(n)) = 1$，再任選 g 滿足 $\text{ord}_{n^2}(g) = n \cdot \alpha$，$\alpha \in Z_{\lambda(n)}$ 且 $\alpha \mid \lambda(n)$，為節省運算量，可令 $g = 1 + n$，此時 $\alpha = 1$。函式 $L(u) = (u-1)/n$，$\forall u \in S_n = \{u \mid 0<u<n^2 \text{ and } u \equiv 1 \pmod n\}$。此系統的公鑰為$(n, g)$，密鑰為$(p, q, \lambda(n))$。

加解密機制

欲加密明文 $m \in Z_n$，任選亂數 $r \in_R Z_n^*$

1.加密過程：

$$c = E(m) \equiv g^m \cdot r^n \pmod{n^2},$$

其中 $c \in Z_{n^2}^*$ 為密文，E(·)為加密函式。

2.解密過程：

$$m = D(c) \equiv \frac{L(c^{\lambda(n)} \pmod{n^2})}{L(g^{\lambda(n)} \pmod{n^2})} \pmod n$$

其中 D(·)為解密函式。

簽章機制

欲簽署明文 m，令 h(·)為單向無碰撞雜湊函式

1.簽署過程：

$$\begin{cases} s_1 \equiv \dfrac{L(h(m)^{\lambda(n)} \pmod{n^2})}{L(g^{\lambda(n)} \pmod{n^2})} \pmod n \\ s_2 \equiv (h(m) \cdot g^{-s_1})^{n^{-1} \bmod \lambda(n)} \pmod n \end{cases}$$

(s_1, s_2)為 m 的 Paillier 簽章。

2.驗證過程：

$$h(m) \equiv g^{s_1} \cdot s_2^n \pmod{n^2}$$

通過驗證式則(m, s_1, s_2)為合法的 Paillier 簽章。

3. 無保護代理簽章者的 Paillier 代理簽章機制

本節將描述如何設計基於 Paillier 密碼系統的代理簽章機制。

3.1 系統參數

原始簽章者 P_0 任選兩個大質數 p 和 q，其中 $p = 2p'+1$、$q = 2q'+1$ 且 p'、q'亦為質數，

並且計算 $n = p \cdot q$。而 g、$\lambda(n)$和其餘函式皆滿足 Paillier 公開金鑰密碼系統的假設。函式 $h(\cdot)$為一個單向無碰撞的雜湊函式。

3.2 授權階段

原始簽章者 P_0 的公鑰為(n, g)，密鑰為$(p, q, \lambda(n))$，每個代理簽章者 P_i 都有一個唯一的身份識別代碼 ID_i。原始簽章者 P_0 計算代理簽章者 P_i 的代理簽章密鑰(v_i, y_i)如下：

$$v_i \equiv \frac{L(h(m_{w_i}, ID_i)^{\lambda(n)} \pmod{n^2})}{L(g^{\lambda(n)} \pmod{n^2})} \pmod{n}$$

$$y_i \equiv (h(m_{w_i}, ID_i) \cdot g^{-v_i})^{n^{-1} \pmod{\lambda(n)}} \pmod{n}$$

其中 m_{w_i} 為 P_0 對 P_i 的代理授權書，紀錄一些授權資訊，例如權力的限制、授權的有效期限、原始簽章者的公鑰和身份等等。接著，原始簽章者 P_0 透過一個安全的管道將代理簽章密鑰(v_i, y_i)及代理授權書 m_{w_i} 傳送給代理簽章者 P_i。

代理簽章者 P_i 收到後採用下列驗證式進行驗證：

$$h(m_{w_i}, ID_i) \equiv g^{v_i} \cdot y_i^n \pmod{n^2}$$

請注意 (v_i, y_i) 亦即原始簽章者對於 (m_{w_i}, ID_i) 所作的 Paillier 數位簽章。

3.3 代理簽章產生階段

代理簽章者 P_i 代替原始簽章者 P_0 簽署一份文件 m 時，需要以其代理簽章密鑰(v_i, y_i)執行下列動作：

1. 隨機挑選亂數 $t \in Z_n$，並計算

   $$r \equiv g^t \cdot y_i^n \pmod{n^2}$$，

   此 r 值除了知道 t 和 y_i 者可以算出來之外，只能透過代理簽章驗證式 (1) 求得。
2. 計算 $k = h(m, r)$，且 r 滿足 k 為偶數，

   此處將簽署文 m 與 r 值透過單向無碰撞雜湊函式 h 得到雜湊值 k，k 將在驗證階段用到。k 為偶數可以使得在稍後步驟 4 中計算 $r_2 \equiv y_i^{k+1} \pmod{n^2} \equiv y_i^{k+1} \pmod{n}$ 有相當大的機率等同於某一 RSA 密文，所以在知道 r_2 與 k 的情況下，反求 y_i 是非常困難的。
3. 計算 $r_1 = t + v_i \cdot k$，

   此處計算出來的 r_1 包含了代理簽章密鑰 v_i 和亂數 t，因此能保證在知道 r_1 與 k 的情況下，求取 v_i 的數值非常的困難。也正因為此 r_1 包含 v_i 和 t 的資訊，所以能透過驗證階段的運算式求得 r 且滿足 $k = h(m, r)$。
4. 計算 $r_2 \equiv y_i^{k+1} \pmod{n^2}$，

   此處計算出來的 r_2 包含了代理簽章密鑰 y_i，因為 r_2 有很大的機率等同於某一 RSA 密文，因此由 r_2 與 k 反求 y_i 是非常困難的事。因為此 r_2 包含 y_i 的資訊，所以透過驗證階段的運算式可以求得 r 且滿足 $k = h(m, r)$。

此時 $(m, m_{w_i}, ID_i, r_1, r_2, k)$ 就是代理簽章者 P_i 對文件 m 所簽署的代理簽章。

3.4 代理簽章驗證階段

驗證者收到代理簽章$(m, m_{w_i}, ID_i, r_1, r_2, k)$之後，可以下列兩步驟驗證其有效性：

1. 計算

$$r' \equiv g^{r_1} \cdot r_2^n \cdot h(m_{w_i}, ID_i)^{-k} \pmod{n^2} \quad \ldots\ldots (1)$$

只有有效的代理簽章才能經由此運算式得到滿足下列(2)式的 r'值。

2. 驗證 $h(m, r') = k$ …… (2)

若通過上述驗證式，則代理簽章$(m, m_{w_i}, ID_i, r_1, r_2, k)$為一個有效的代理簽章。如果代理簽章以 3.3 節的步驟產生，則由 (1) 式可以求出祕密的 r 值，並且可以滿足 (2) 式的驗證。

4. 有保護代理簽章者的 Paillier 代理簽章機制

4.1 系統參數

原始簽章者 P_0 任選兩個大質數 p_0 和 q_0，其中 $p_0 = 2p_0' + 1$、$q_0 = 2q_0' + 1$ 且 p_0'、q_0'亦為質數，並且計算 $n_0 = p_0 \cdot q_0$。而 g_0、$\lambda(n_0)$和其餘函式皆滿足 Paillier 公開金鑰密碼系統的假設。同樣地，每個代理簽章者 P_i 亦挑選其本身之 Paillier 密碼系統參數 p_i, q_i, n_i, g_i, 並滿足前述性質。

<u>授權階段</u>

原始簽章者 P_0 的公鑰為(n_0, g_0)，密鑰為$(p_0, q_0, \lambda(n_0))$。每個代理簽章者 P_i 的公鑰(n_i, g_i)，密鑰為$(p_i, q_i, \lambda(n_i))$ ，且具有唯一的身份識別碼 ID_i。原始簽章者 P_0 計算代理簽章者 P_i 的代理簽章密鑰(v_i, y_i)如下：

$$v_i \equiv \frac{L(h(m_{w_i}, ID_i)^{\lambda(n_0)} \pmod{n_0^2})}{L(g_0^{\lambda(n_0)} \pmod{n_0^2})} \pmod{n_0}$$

$$y_i \equiv (h(m_{w_i}, ID_i) \cdot g_0^{-v_i})^{n_0^{-1} \pmod{\lambda(n_0)}} \pmod{n_0}$$

在此代理簽章機制中，每個代理簽章者 P_i 都有自己的公鑰(n_i, g_i)，所以原始簽章者可以採用 Paillier 密碼系統加密方式傳送代理簽章密鑰(v_i, y_i)。但因為 n_i 可能小於 n_0，為了避免 reblocking 問題，我們可以將代理密鑰(v_i, y_i)依照 n_i 的位元長度來做適當的分段，接著再進行加密傳送。

代理簽章者 P_i 收到後，以下列驗證式進行驗證：

$$h(m_{w_i}, ID_i) \equiv g_0^{v_i} \cdot y_i^{n_0} \pmod{n_0^2}$$

4.2 代理簽章產生階段

代理簽章者 P_i 代替原始簽章者 P_0 簽署一份文件 m 時，需要以其代理密鑰(v_i, y_i)及個人密鑰 $\lambda(n_i)$執行下列動作：

1. 隨機挑選亂數 $t \in Z_n^*$ ，並計算

$$r \equiv g_0^t \cdot y_i^{n_0} \pmod{n_0^2} \text{ ，}$$

2. 計算 $k \equiv h(m, r) \pmod{n_i^2}$，且且重複上一步驟直到 r 滿足 k 為偶數，h() 為單向無碰撞之雜湊函式，
3. P_i 對 k 簽章得到

$$u \equiv \frac{L(k^{\lambda(n_i)} \pmod{n_i^2})}{L(g_i^{\lambda(n_i)} \pmod{n_i^2})} \pmod{n_i}$$

$$y_p \equiv (k \cdot (g_i^u)^{-1})^{n_i^{-1} (\bmod \lambda(n_i))} \pmod{n_i}$$

如此可以保證只有代理簽章者 P_i 可以做出 u 和 y_p(數位簽章的基本性質)，連原始簽章者 P_0 都無法偽造，以達到保護代理簽章者的目的。

4. 計算 $r_1 = t + v_i \cdot k$ ，
5. 計算 $r_2 \equiv y_i^{k+1} \pmod{n_0^2}$ ，

$(m, m_{w_i}, ID_i, r_1, r_2, u, y_p)$就是代理簽章者 P_i 對文件 m 所簽署的代理簽章。

4.3 代理簽章驗證階段

驗證者收到代理簽章$(m, m_{w_i}, ID_i, r_1, r_2, u, y_p)$之後，以下列三步驟驗證其有效性：

1. 驗證 $k \equiv g_i^u \cdot y_p^{n_i} \pmod{n_i^2}$ …… (3)
2. 計算

$$r' \equiv g_0^{r_1} \cdot r_2^{n_0} \cdot h(m_{w_i}, ID_i)^{-k} \pmod{n_0^2} \quad \text{…… (4)}$$

3. 驗證 $h(m, r') \equiv k \pmod{n_i^2}$ …… (5)

若通過上述驗證式，則$(m, m_{w_i}, ID_i, r_1, r_2, u, y_p)$為一個有效的代理簽章。同樣地如果代理簽章根據 4.2 節的步驟產生，則 (4) 式可以求出祕密的 r 值，並且滿足 (5) 式之驗證。

5. 安全性分析

5.1 代理簽章的正確性

5.1.1 無保護代理簽章者的 Paillier 代理簽章機制：

在驗證階段，驗證者可以用 3.3 節的步驟算出來的 r_1, r_2, k 代入 (1) 式計算

$$
\begin{aligned}
r' &\equiv g^{r_1} \cdot r_2^n \cdot h(m_{w_i}, ID_i)^{-k} \pmod{n^2} \\
&\equiv g^{(t+v_i k)} \cdot (y_i^{k+1})^n \cdot g^{-v_i k} \cdot y_i^{-nk} \pmod{n^2} \\
&\equiv g^t \cdot y_i^n \pmod{n^2} \\
&\equiv r \pmod{n^2}
\end{aligned}
$$

所以 $h(m, r') = h(m, r) = k$, 如此可以驗證代理簽章的合法性。

5.1.2 有保護代理簽章者的 Paillier 代理簽章機制：

在驗證階段，驗證者必須以下列式子驗證代理簽章者的身份

$$k \equiv g_i^u \cdot y_p^{n_i} \pmod{n_i^2}$$

接著以 4.3 節的步驟算出來的 r_1, r_2, k 代入 (3) 計算

$$
\begin{aligned}
r' &\equiv g_0^{r_1} \cdot r_2^{n_0} \cdot h(m_{w_i}, ID_i)^{-k} \pmod{n_0^2} \\
&\equiv g_0^{(t+v_i \cdot k)} \cdot (y_i^{k+1})^{n_0} \cdot g_0^{-v_i k} y_i^{-n_0 k} \pmod{n_0^2} \\
&\equiv g_0^t \cdot y_i^{n_0} \pmod{n_0^2} \\
&\equiv r \pmod{n_0^2}
\end{aligned}
$$

所以 $h(m, r') \equiv h(m, r) \equiv k \pmod{n_i^2}$, 如此可以驗證代理簽章的合法性。

5.2 代理簽章密鑰(v_i, y_i)的安全性分析

5.2.1 無保護代理簽章者的 Paillier 代理簽章機制：

假設惡意攻擊者 A 想要假冒原始簽章者 P_0 的身份授權給代理簽章者 P_i，A 必須偽造對應於授權書 m_{w_i} 的代理簽章密鑰(v_i, y_i)且滿足驗證式 $h(m_{w_i}, ID_i) \equiv g^{v_i} \cdot y_i^n \pmod{n^2}$。($v_i$, y_i)相當於原始簽章者對於 m_{w_i} 及 ID_i 的 Paillier 簽章，由於我們假設 $h(\cdot)$為單向無碰撞的雜湊函式，因此在 Random Oracle 模型且在合成數模數 n 次方根無法有效率地計算的假設下，Paillier 簽章系統可證明為無法偽造的，因此上述攻擊是非常困難的事情。

若 A 想從代理簽章(m, m_{w_i}, ID_i, r_1, r_2, k)中的 r_1 及 k 計算出密鑰 v_i，因為 $r_1 = t + v_i \cdot k$，在完全不知道 t 的情況下要計算 v_i 的機率只有 1/n，實際上 t 的 Paillier 密文 r 可由 (1) 式計算出來，因此 v_i 的安全性等同於 t 的安全性，架構在 Paillier 的語意安全性之上。若 A 想從代理簽章(m, m_{w_i}, ID_i, r_1, r_2, k)中的 r_2 及 k 計算出 y_i，因為 $r_2 \equiv y_i^{k+1} \pmod{n^2} \equiv y_i^{k+1} \pmod{n}$，這相當於解 RSA 的密文。

5.2.2 有保護代理簽章者的 Paillier 代理簽章機制：

同樣的，攻擊者 A 要假冒授權需要偽造 Paillier 簽章，相信這是非常困難的一件事情。而傳送過程改採 Paillier 加密方式進行加密傳送，除非攻擊者 A 能破解 Paillier 密碼系統，才能獲得代理簽章密鑰(v_i, y_i)，在我們的假設下這也是非常困難的事情。若 A 想從代理簽章(m, m_{wi}, ID_i, r_1, r_2, u, y_p)中的 r_1 及 k 計算出 v_i 或從 r_2 及 k 計算出 y_i，與無保護的 Paillier 代理簽章機制一樣，都是非常困難的事情。

5.3 代理簽章的安全性分析

5.3.1 無保護代理簽章者的 Paillier 代理簽章機制：

除了原始簽章者之外，沒有人可以偽造合法的代理簽章。因為偽造的代理簽章(m, m_{wi}, ID_i, r_1, r_2, k)必須滿足 (1) 及 (2) 式。

a. 假設攻擊者 A 先挑選 r_1、r_2 和 k 的值，再根據(1)式計算出 r。由於 h(·)為單向無碰撞雜湊函式，如此得到的 r 與 k 能夠滿足(2)這個方程式的機率是計算上可以忽略的。

b. 假設攻擊者 A 先挑選 r 的值，根據(2)式算出 k，再根據(1)式計算出 r_1 與 r_2。這樣子相當於是偽造 $r \cdot h(m_{w_i}, ID_i)^k$ 的 Paillier 簽章如下：

$$r \cdot h(m_{w_i}, ID_i)^k \equiv g^{r_1} \cdot r_2^n \pmod{n^2}$$

必須要有密鑰 λ(n)才能計算 r_1 與 r_2，如果先挑選 r_1 的數值，計算 r_2 的困難度可以等效至“模數 n^2 下求取 n 次方根”問題的困難度。如果先挑選 r_2 的數值，計算 r_1。的困難度等效於破解 Paillier 單向函式(破解 Class[n, g] 問題, [9]中的定理 5)。

5.3.1 有保護代理簽章者的 Paillier 代理簽章機制：

任何人(包含原始簽章者)均無法偽造合法的代理簽章。因為偽造的代理簽章(m, m_{w_i}, ID_i, r_1, r_2, u, y_p)必須滿足 4.3 節中等式 (3), (4) 及 (5)：

a. 假設攻擊者 A 想要偽造 u 和 y_p，但這相當於是偽造 m 和 r 的 Paillier 簽章一樣，單向無碰撞雜湊函式 h(·)以及 Paillier 單向函式的安全性質使得這個工作非常困難。

b. 假設攻擊者 A 先挑選 r 的數值，根據 (5) 式計算出 k，再挑選 r_1 的數值，然後根據 (4) 式計算出 r_2，但這樣相當於是偽造 $r \cdot h(m_{w_i}, ID_i)^k$ 的 Paillier 簽章如下：

$$r \cdot h(m_{w_i}, ID_i)^k \equiv g_0^{r_1} \cdot r_2^{n_0} \pmod{n_0^2}$$

必須要有密鑰 $\lambda(n_0)$才能算 r_2，否則計算 r_2 的困難度和“模數 n_0^2 下求取 n_0 次方根”的困難度相同。假設攻擊者 A 先挑選 r_2 的值，根據 (4) 式計算 r_1。這樣還是相當於偽造 $r \cdot h(m_{w_i}, ID_i)^k$ 的 Paillier 簽章如下：

$$r \cdot h(m_{w_i}, ID_i)^k \equiv g_0^{r_1} \cdot r_2^{n_0} \pmod{n_0^2}$$

必須要有密鑰 $\lambda(n_0)$才能算 r_1，否則計算 r_1 相當於破解 Paillier 單向函式。

5. 結論

近年來，由於電子投標與電子投票系統等等安全的多方運算開始逐漸被重視，加上 Paillier 密碼系統具有加法同型與機率式加密的特性，使得 Paillier 密碼系統特別適合應用在電子投標與電子投票相關技術上。然而，現存的代理簽章機制大都是基於解離散對數的問題上，考慮到這些代理簽章機制無法直接應用在基於 Paillier 機制的電子投票與投標系統系統，為了因應往後基於 Paillier 機制下的代理簽章需求，我們在本文中提出兩種可以應用在 Paillier 公開金鑰密碼系統之代理簽章機制，並且在文中分析其安全性。

參考文獻

[1] M. Mambo, K. Usuda, E. Okamoto, “Proxy signatures: Delegation of the power to sign message”, IEICE Transaction Functional E79-A(9), pp.1338-1354, 1996.

[2] Z. Shao, “Proxy signature schemes based on factoring”, Information Processing Letters, 85(3), pp.137-143, 2003.

[3] R. Lu, Z. Cao, “A proxy-protected signature scheme based on conic”, ACM International Conference Proceeding Series, Vol.85, Proceedings of the 3rd international conference on Information security, pp.22-26, 2004.

[4] P. Paillier, “Public-key cryptosystems based on composite degree residuosity classes residues”, Advances in Cryptology - Eurocrypt’99, LNCS 1592, Springer-Verlag, pp.223-238, 1999.

[5] Masahiro Mambo , Keisuke Usuda , Eiji Okamoto, “Proxy signatures for delegating signing operation”, Proceedings of the 3rd ACM conference on Computer and communications security, pp.48-57, March 14-15, 1996.

[6] B.C. Neuman, “Proxy-based authorization and accounting for distributed systems”, Proc. 13th International Conference on Distributed Systems, pp. 283-291, 1993.

[7] W. Diffie, M.E. Hellman, “New directions in cryptography”, IEEE Trans. IT-22, pp.644-654, 1976.

[8] R. Rivest, A. Shamir, and L. Adleman, “A method for obtaining digital signatures and public key cryptosystems”, Comm. ACM 21, pp.120–126, 1978.

[9] T. El Gamal, “A public key cryptosystem and a signature scheme based on discrete logarithms”, IEEE Trans. IT-31, pp.469–472, 1985.

[10] S. Goldwasser and S. Micali, “Probabilistic encryption”, Journal of Computer and System Sciences, 28, pp.270-299, 1984.

[11] R. Cramer, R. Gennaro and B. Schoenmakers “A Secure and Optimally Efficient Multi-Authority Election Scheme”, Advances in Cryptology - Eurocrypt’97, LNCS 1233, pp. 103-118, 1997.