ERPS
Journal of e-Business
第十四卷　第一期　2012 年 3 月（pp.143～160）

適用於電子商務系統之多文件簽密法設計

蘇品長*　高嘉言
國防大學資訊管理學系

摘要

網際網路盛行，全球資訊網出現，造就電子商務時代來臨，商業交易興盛與否，必須仰賴電子文件與數位簽章來確立其權利與義務，資訊的完整性、鑑別性及不可否認性則是電子簽章所提供的功能，因此每份傳遞出來的文件必經過嚴格簽章或加密的作業，然而數以萬計的文件往來勢必經過數以萬次的簽章或加密運算，所耗費的時間甚鉅，本研究以研究橢圓曲線快速運算為基礎，其計算效率比現行其他公開金鑰演算法快速，且運用背包型態提出多份文件一次簽密之應用方法，能夠大量減少企業間商用文件的簽章次數，達到提升效率縮短作業時間的效益。

關鍵詞：電子商務、橢圓曲線密碼學、多文件簽密

A Design of Multi-signcryption for E-commerce System

Pin-Chang Su　Chia-Yen Kao
Department of Information Management, National Defense University

Abstract

With the prevalence of the Internet, World Wide Web emerged, creating e-commerce era, the rise of commercial transactions, and otherwise must rely on electronic documents and digital signatures to establish their rights and obligations, authenticity of information, authenticity and non-repudiation is provided by the electronic signature function, digital signature technology affect the development of e-commerce. This paper is to study fast computing based on elliptic curve cryptosystem, computational efficiency than other existing public key algorithm quickly, and make multiple copies of documents used a signcryption method of application, this method can significantly reduce the number of commercial paper encryption and signature, to enhance the efficiency of the benefits of reduced operating time.

* 通訊作者
電子郵件：spc.cg@msa.hinet.net

Key Words: E-Commerce, Elliptic Curve Cryptosystem, Multi-Signcryption

1. 緒論

網路環境中，資料傳輸必需以電子形式交換，然而電子文件容易被竄改及需借助工具呈現等特性，使得法令規定某項行為必須以書面為之時，以電子文件做成該項行為，以滿足法定要式要求時，便會產生問題；此外，傳統實體簽名或蓋章之功能，目的為辨識簽章人之身分及防止事後否認，網路環境，電子文件可用電子方式簽章，然以何種技術方能與實體簽章之功能相當，亦生疑義。因此，隨著電子商務與電子服務的普及發展，規範電子簽章之法律效力，建立安全及可信賴的線上身分認證機制，確保資訊在網路傳輸及儲存過程中之安全性與完整性，為電子化應用能否普及的關鍵。對電子商務應用及各種基礎建設的推展而言，公共政策的配合和技術標準的研擬是兩大重要支柱，為了確保整體網路的相容性，在發展各項基礎建設及電子商務應用時，各種工具、使用者界面及傳輸協定的標準化是絕對必要的。因應全面發展電子商務之需要，近年來許多國家紛紛制訂電子簽章法或電子交易法，來規範電子文件、電子簽章效力，以解決因法令規定中對書面、簽章之要式要求，所造成電子化環境應用上的障礙。

網際網路為一高度透明、公開的環境，竊取、偽造、竄改資料、冒名欺騙，皆為常見的攻擊行為。如何建立一個安全及可信賴的網路環境，為電子商務能否全面普及的關鍵。所謂「安全及可信賴的網路環境」是確保資訊在網路傳輸過程中不易遭受偽造、非法存取、竄改、竊取或截聽等行為，並且亦能鑑別交易雙方身分，防止事後否認交易事實（許建隆，2000），達到上述觀點，必須仰賴數位簽章技術。傳統的印章或親筆簽名與數位簽章最大的差異，除了所欲簽署的文件之形式不同外，印章或親筆簽名與該文件內容是各自獨立。換言之，針對不同的文件，簽署者使用印章或親筆簽名以產生的簽章不會隨著文件內容不同而有所不同。從數位簽章產生過程來看，數位簽章是透過電子文件與簽署者所擁有的秘密資訊，經簽章產生機制計算所得的結果。因此，數位簽章與電子文件的內容息息相關，同一位簽署者所產生的數位簽章，會隨著電子文件內容不同而有所不同。數位簽章是以密碼學上的公開金鑰密碼系統（Public Key Cryptosystem），亦稱「非對稱密碼系統（Asymmetric Cryptosystem）」為基礎來實作，即系統中，每一位使用者必須擁有自己的金鑰對：一把密鑰與一把公鑰，使用者必須秘密地保存自己的密鑰，並且將其公鑰公佈於網路。之後，使用者可以利用自己的密鑰對文件進行簽署；而數位簽章的接收者可以利用該簽署者的公鑰來驗證數位簽章的有效性。目前較普遍的數位簽章機制有：RSA、ElGamal 以及 DSA。RSA 數位

簽章機制為 1978 年，由 Rives、Shamir 及 Adleman 三位學者利用分解大質數的困難度，提出 RSA 數位簽章機制。VISA、MasterCard、IBM、Microsoft、HP 等公司所協力制定的安全電子交易標準（Secure Electronic Transactions, SET）便是採用 RSA 數位簽章機制。ElGamal 數位簽章機制為 T. ElGamal 於 1985 年提出，此機制的安全性是建立在解決離散對數問題的困難度上。DSA 數位簽章機制，由美國國家標準局（National Institute of Standard and Technology, NIST）於 1991 年 8 月提出，其安全性與 ElGamal 數位簽章機制相同，皆建立在解決離散對數問題的困難度上。自從 80 年代中期被發表以來，橢圓曲線密碼系統（Elliptic Curve Cryptography, ECC）已成為一個十分令人感到興趣的密碼學分支，近年來橢圓曲線加密法越來越受重視，其主要原因包括⑴較快的運算速度。⑵金鑰長度較短：RSA 與 ElGamal 系統中需要使用長度為 1024 位元的模數，才能達到足夠的安全等級；而 ECC 只需使用長度為 160 位元的模數即可。⑶傳送密文或簽章所需頻寬（bandwidth）較少。對於橢圓曲線系統與目前系統作比較，橢圓曲線密碼系統較相對於植基於因式分解及離散對數難題的系統要快。ECC、RSA 及 DSA 的執行效率比較表詳如表 1 所示（蘇品長，2007）。

▼ 表 1　ECC、RSA 及 DSA 的執行效率之分析比較

Function	Encrypt	Decrypt	Sign	Verify	Sig.size
RSA	17	384	384	17	1024
DL_based	480	240	240	480	320
ECDL_based	120	60	60	120	320

- the unit of time is one 1024-bit modular multiplication
- RSA private-key operation uses CRT
- RSA－1024≈DL－1024≈ECDL－160/170≈AES－80

國際間有關數位簽章之研究主題，除一段性之應用外，亦包含：多重簽章、群體導向數位簽章、代理簽章、門檻式數位簽章、具訊息回復的數位簽章、簽章加密法、鑑別加密法、盲目簽章、不可否認簽章、簽章機制之攻擊等議題（張真誠、林祝興，2006）；ElGamal 在其博士論文另提出簽暑兩明文的方法（ElGamal, 1985），Horster、Michels、Petersen 等人亦提出一次簽暑三明文之 ElGamal 簽章演算法，主要就是希望藉由結合離散對數與資訊混淆之特點來增簽章系統之便捷性，打破以往檔案單一文件單一簽章的方式（Horster et al., 1994）。Naccache 等人以 DSA 為基礎提出了以整批簽章（batch signature）的架構（Naccache et al., 1994），同年 Lim 和 Lee 提出了能夠偽造簽章的攻擊方法，此方法可以偽造出通過整批式驗證的簽章，但是這些偽造簽章在個別的驗證是不正確的（Lim and Lee, 1994），之後又有許多的學者提出新的批次加密方法（Lin et al., 2005; Chou et al., 2010）；另綜整國內外實作軟體，對於多份文

件的加密運作方式，大致分為似「7-Zip 壓縮軟體」及似「資料匣加解密軟體」二類（蘇品長等人，2009）；上述方法均無法具有雪崩效應，安全性及效率性極待改進。本研究提出一種植基於橢圓曲線離散對數之多文件一次簽章及加密的演算法，導入背包型態問題與橢圓曲線離散對數的難題來加強其安全性，達成擬亂的效果，打破以往簽章演算法 - 單一文件單一簽章或多文件批次簽章的方式，能大幅縮短電子文件簽章的次數，不同於現行作業方式，藉由混淆機制，將全部文件擬亂成一份密文，使其密文具有雪崩效應，令竊密者既使截獲資訊亦無法得知明文，成為一個能增加文件解密難度，卻不會降低其演算效率之簽密法，進而提高傳送資料的安全性並可降低重要文件傳遞的遞所需耗費的時間，同時兼具加密功能，達到資訊的完整性、鑑別性、不可否認性與機密性等特點。

本研究共分為五節：首先為前言，說明研究動機與目的；其次為文獻探討，分類整理、歸納分析與相關的文獻，內容主要為介紹橢圓曲線公開金鑰密碼系統演算法、數位簽章演算法、ElGamal 及橢圓曲線等數位簽章機制之原理、演算架構；第三節為本研究所提的多份文件一次簽章及加密的方法；第四節為安全及效益分析，針對本研究所提的簽章及加密機制，進行安全性及效益分析；最後一節為結論，彙整本研究之預期成果。

2.文獻探討

本節將分類整理、歸納分析與本研究相關的文獻，主要介紹加解密技術及數位簽章機制之原理、演算架構，分述如後。

2.1 橢圓曲線密碼系統

公開金鑰密碼學早在百年前就已經很完備了，而橢圓曲線在代數學與幾何學上廣泛的研究也已超出百年之久，且有豐富且深奧的理論，橢圓曲線系統第一次應用於密碼學上是於 1985 年由 Koblitz 與 Miller 分別提出（蘇品長，2007），從此橢圓曲線在密碼學中就扮演重要的角色。假設一個橢圓曲線是屬於 F_q，而 P 是橢圓曲線 E 上的一個點，給定一個屬於橢圓曲線 E 上的一個點 Q，若要找出一整數 k 使得 $Kp = Q$，因為其特殊的點加法運算，破密者除了逐一的窮舉所有可能的點之外，別無他法，直至目前為止，這個問題仍無法於多項式時間內求解。ECC 已經被致力於標準化，包括 IEEE P1363 所採用的公開鑰匙密碼標準。橢圓曲線公開密碼技術比目前已知安全無虞的公開密碼系統，具有更高的加解密效能、較少的頻寬和儲存能量、較短的簽章長度及鑑別資料、快速的加密和簽章及容易實現於極小的硬體設備等優點，適用於計算資源被限制、積體電路空間有限、頻帶受限及需要高速的使用環境，諸如：智慧卡、無

線通信、電腦網路等環境。橢圓曲線加密法的基本操作原理如后：

金鑰產生

令系統公開參數為一個橢圓曲線 E 及模數 p。使用者執行：

任選一個整數 k，$0<k<p$。

任選一個點 $A \in E$，並計算 $B=kA$。

公鑰為 (A, B)，私鑰為 k。

註：從 (A, B) 中去推導相當於計算離散對數問題。

加密程序

令明文 M 為 E 上的一點。首先任選一個整數 $r \in Z_p$，然後計算密文 $(C_1, C_2)=(rA, M+rB)$。

註：密文為兩個點。

解密程序

計算明文 $M=C_2-kC_1$。

2.2　數位簽章演算法

1991 年，美國 NIST 公佈 DSA 為國家數位簽章標準。DSA 公佈後，雖引發以下爭議，但業界及學界仍是接受此一標準（許建隆，2000）：

⑴DSA 不能用來做加密或金鑰分配之用，只能用來做數位簽章。

⑵DSA 是由美國國家安全局所發展出來的一種 ElGamal 數位簽章法的變形，普遍上使用者仍是存有疑慮而擔心藏有暗門（trapdoor）設計，並不像 RSA 或 ElGamal 數位簽章法是由學術界人士所設計出來的而較能信任。

⑶DSA 的計算速度比 RSA 要來得慢。DSA 所需簽署時間與 RSA 大約相同，但所需驗證簽章的時間要比 RSA 慢約 10 至 40 倍。

⑷RSA 雖然不是政府頒布的一項標準（牽涉到專利的問題），但是全世界的使用者早已將之視為一項重要的數位簽章標準來使用。

⑸DSA 數位簽章系統運作方式，簡述如下：

系統公開參數

p：512 至 1024 位元的大質數

q：160 位元的 $p-1$ 之質因數

e_1：$e_1=w^{(p-1)/q} \bmod p$，其中 $w<p-1$ 且 $w^{(p-1)/q} \bmod p>1$

h：一個單向雜湊函數（one-way hash function），輸出值為 160 位元。

註：搭配 DSA 的單向雜湊函數標準 SHA-1（Secure Hash Algorithm）

金鑰產生

每一個使用者任選一個整數 $d \in Z_q$ 為私鑰，並計算公鑰 $e_2 = e_1{}^d \bmod p$。

簽署程序

欲簽署訊息為 M，任選一數 $r < q$。

計算 $S_1 = (e_1{}^r \bmod p) \bmod q$。

計算 $S_2 = r^{-1}(h(M) + dS_1) \bmod q$。

(S_1, S_2) 為 M 的數位簽章。

驗證程序

計算下列各數

$a = S_2{}^{-1} \bmod q$

$b = ah(M) \bmod q$

$c = S_1 a \bmod q$

$V = (e_1{}^b \times e_2{}^c \bmod p) \bmod q$

若 $V = S_1$，則 (S_1, S_2) 通過驗證。

2.3 ElGamal 簽署三明文演算法

ElGamal 方法主要為值基於離散對數問題，且在其博士論文提到簽署三份明文之方法（ElGamal, 1985）。

系統公開參數

p：一個大質數

q：為 $p-1$ 或 $p-1$ 的一個大質因數

g：$1 < g < q$，滿足 $g^q = 1 \bmod p$

h：一個單向雜湊函數

金鑰產生

使用者任選整數 $x \in Z_q$ 為私鑰，計算公鑰 $y = g^x \bmod p$。

簽署

簽署三個訊息 m_1, m_2, m_3 時（通常需要有特殊格式或用單向函數保護防止選擇密文攻擊），簽署者首先任選一個亂數 $k \in Z_q$，滿足 $\gcd(k, q) = 1$；接下來，計算 $r = g^k \bmod p$ 與 $m_1 \equiv x_A f(m_2, r) + kg(m_3, s)$，求得$(r, s)$，將訊息 m_1, m_2, m_3 的簽章(m_1, m_2, m_3, s, r)傳給收訊方。

驗證簽章

收訊方取得資訊後，需計算驗證下列算式：$\alpha^{m_1} \equiv y_A^{f(m_2, r)} r^{g(m_3, s)}$，是則驗收，否則拒絕。

2.4　橢圓曲線簽章演算法

橢圓曲線簽章演算法（Elliptic Curve Digital Signature Algorithm, ECDSA），是一種標準的橢圓曲線數位簽章演算法（蘇品長，2007），方法總共分為三部分，系統初始階段、簽章階段及驗證簽章階段，各階段分述如下：

系統初始階段

在有限域 F_q 上選取一條安全的橢圓曲線 $E(F_q)$，（q 為一個 160bit 以上之大質數）並在 $E(F_q)$上選一階數（order）為 n 的基點 G，使得 $nG = O$，其中 O 為此橢圓曲線之無窮遠點。簽章者隨機選擇一整數 n_A 當成私鑰，其中 n_A 介於$[1, n-1]$計算簽章者公鑰 $PK_A = n_A G$，將(E, G, PK_A)公開。

簽章階段

假設欲簽章之訊息為 m，隨機選擇一整數 m 介於$[1, n-1]$，計算 $R = rG = (x, y)$、計算 $s = r^{-1}(h(m) - n_A x)$，將訊息 m 的簽章(m, s, R)傳給收訊方。

驗證簽章階段

取得簽章者之公鑰及系統公開訊息(E, G, PK_A)，檢驗 r 及 s 是否介於$[1, n-1]$，若不在範圍內則否定其簽章，計算 $V_1 = x \cdot PK_A + s \cdot R$ 及 $V_2 = h(m) \cdot G$，若 $V_1 = V_2$ 則驗收，否則拒絕。

3. 多文件簽密法

隨著網路盛行，企業與企業間的電子文件往來已成為趨勢，數以萬計的電子文件

需進行數以萬次的簽密與驗證，因此，安全性不佳、運算能力緩慢及龐大的金鑰管理機制會拖慢電子商務的發展腳步。本研究中，針對上述問題提出一種植基於橢圓曲線離散對數，可運用於多文件簽密之演算法，以解橢圓曲線離散對數的難題，增加系統安全性並設計擬亂功能，增加系統的強度及改進以往簽章演算法，能大幅縮短電子文件簽章的次數。另一個優點則是納入加解密方法，在同樣的安全度之下，橢圓曲線密碼系統僅需要較小的密鑰長度。本研究的系統設計示意如圖 1 及圖 2 所示。

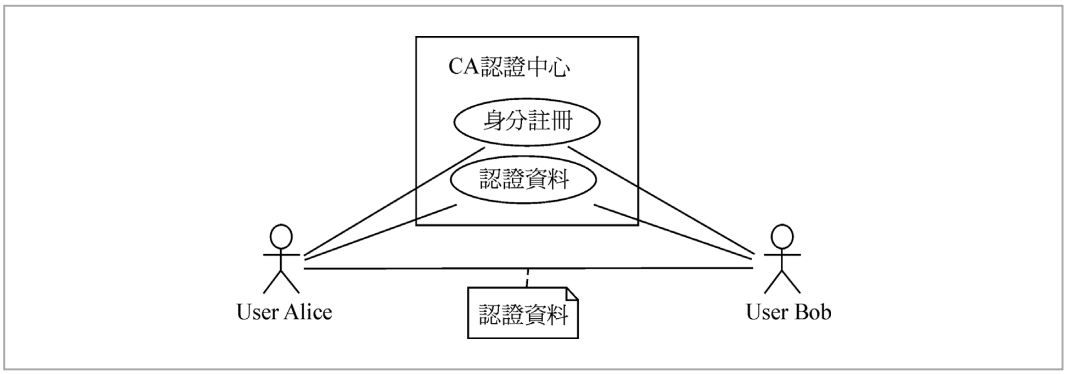


圖 1　系統設計示意圖（一）

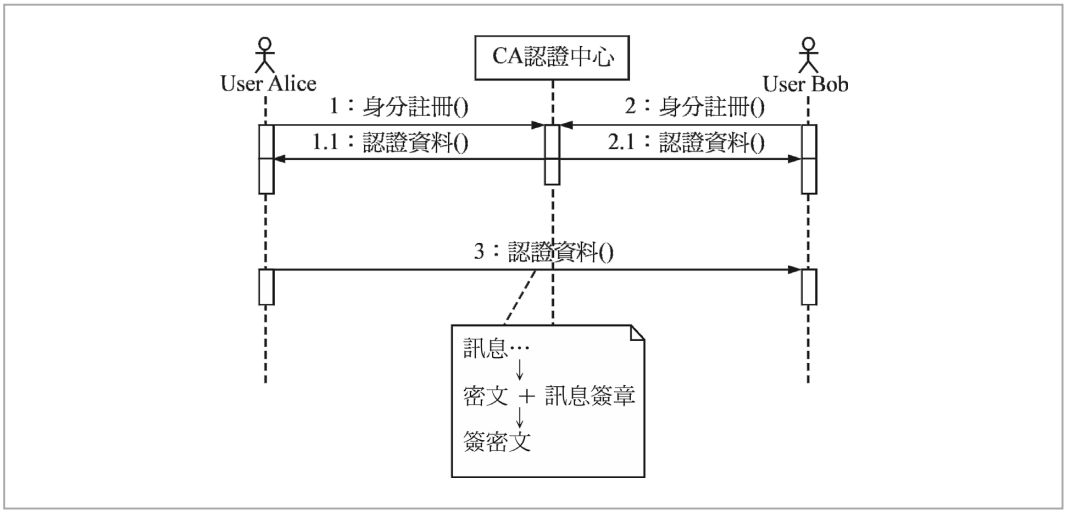


圖 2　系統設計示意圖（二）

3.1　系統參數設定

本研究主要是結合背包理論及橢圓曲線的簽密系統來作密碼系統設計，系統初始時，針對密碼系統作一個參數選擇及設定。簽密法在完整性上面，除了單向雜湊函數

的檢查之外，同時密文之間執行橢圓曲線點加運算，使得密文產生雪崩效果，亦可當成完整性檢查的一環；本法於橢圓曲線簽密過程以背包問題概念導入區塊位移的觀念來增加破密困難。表 2 為本系統中所使用的符號說明，本方法各階段的執行步驟如后：

▼ 表 2　系統使用符號說明表

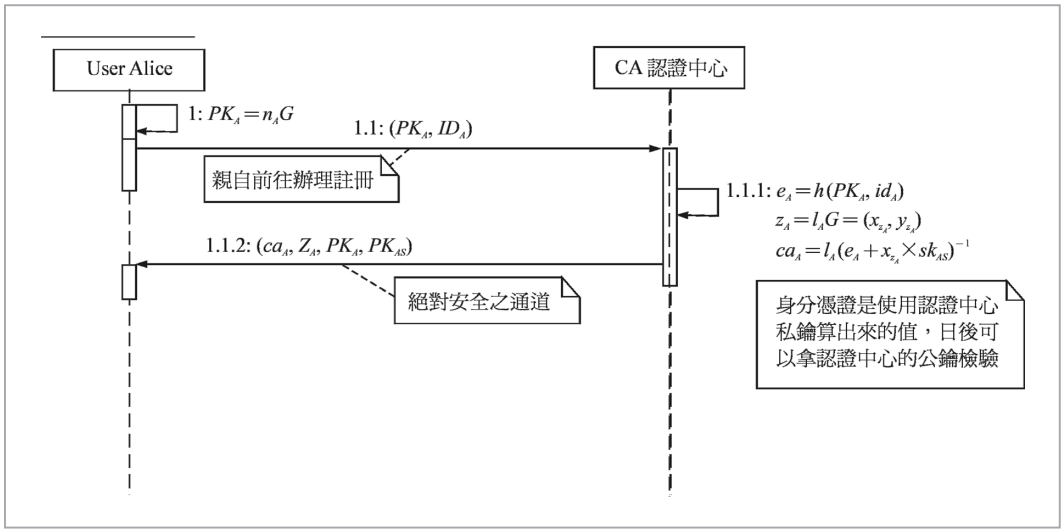
項次	符號	說明
1	CA	認證中心
2	$E(F_q)$	有限域 F_q 中的一條橢圓曲線
3	G	橢圓曲線中的基點
4	n	橢圓曲線上基點的秩（order）
5	q	$q > 2^{160}$ 之質數
6	id_A, id_B	A、B 的 id 資訊
7	PK_{cA}, sk_{cA}	CA 的公私鑰
8	PK_A, PK_B	使用者 A、B 之公鑰
9	n_A, n_B	使用者 A、B 所選擇之私鑰，$PK_A = n_AG$, $PK_B = n_BG$。
10	ca_A, ca_B	使用者 A、B 之憑證
11	$h()$	認證中心公開之雜湊函數
12	$f_{m2p}()$	將訊息轉為橢圓曲線點之函數
13	$f_{p2m}()$	將橢圓曲線點轉為訊息之函數

步驟一

在金鑰管理方面，使用者皆必須經過認證中心 CA 的檢驗認證之後，方可擁有自己的公鑰和私鑰來進行加解密動作，因為使用者 A、B 在資料傳遞前，必定先進行使用者身分的認證，利用以達成不可否認性。系統在有限域 F_q 上選取一條安全的橢圓曲線 $E(F_q)$（q 為一個大質數）並在 $E(F_q)$ 上選一階數（order）為 n 的基點 G，使得 $nG = O$，其中 O 為此橢圓曲線之無窮遠點。

步驟二

使用者 A、B 分別選擇 $n_A, n_B \in Z_q^*$ 當成私鑰，計算出相應之公鑰 $PK_A = n_A \cdot G$、$PK_B = n_B \cdot G$（PK_A, PK_B 不能為無窮遠點），並透過一個絕對安全的通道將本身的公鑰及身分 id_A, id_B 送至認證中心計算驗證，圖 3 為憑證申請流程示意圖。



▲ 圖 3　憑證申請流程示意圖

步驟三

認證中心利用雙方的公鑰及身分資料計算出關聯值 $e_A = h(id_A || PK_A)$、$e_B = h(id_B || PK_B)$，並為使用者 A、B 分別選擇 l_A, l_B，使 $Z_A = l_A G = (x_{Z_A}, y_{Z_A})$、$Z_B = l_B G = (x_{Z_B}, y_{Z_B})$，產生憑證 $ca_A = l_A^{-1}(e_A + x_A n_{AS})$、$ca_B = l_B^{-1}(e_B + x_B n_{AS})$，認證中心 $(ca_A, Z_A, PK_A, PK_{AS})$ 將傳回給使用者 A、$(ca_B, Z_B, PK_B, PK_{AS})$ 傳回給使用者 B，系統選擇的一個單向無碰撞雜湊函數 $h()$，最後公開 $E(F_q), G, n, q, PK_A, PK_B, h()$。

3.2　簽密階段

步驟一

A 將多份訊息明文分成數個區塊且定義 m_i，其中每份文件各切割為兩塊，並對 m_i 實施雜湊運算，利用明文轉點方式將明文轉成點坐標 P_i，$\overline{P_i} = \{P_1, P_2, \cdots, P_n\}$ 的相關計算如式(1)-式(3)：

$$\overline{m_i} = \{m_{11}, m_{12}, m_{21}, m_{22}, \cdots, m_{n1}, m_{n2}\} \tag{1}$$

$$h(\overline{m_i}) = m \tag{2}$$

$$f_{m2p}\{(m_{11}, m_{12}), (m_{21}, m_{22}), \cdots, (m_{n1}, m_{n2})\} = \{P_1, P_2, \cdots, P_n\} = \overline{P_i} \tag{3}$$

步驟二

使用者 A 隨機選取 $k_A \in (1, n-1)$ 值接著計算出 R_A 如 (4) ，且求出 r_A 如(5)，定義 $\overline{K} = \{K_1, K_2, \cdots, K_i\} \in (0, 1)$ 算出 w 如(6)，以二進位表達 w 值，對應 1 代表右旋一個區塊，對應 0 代表左旋一個區塊，準備作簽章值 s_A 之計算。相關計算公式如后說明：

$$R_A = k_A G = (x_R, y_R) \tag{4}$$

$$r_A = x_R \bmod n \text{（假如 } r_A = 0 \text{ 則需重新選取參數及運算）} \tag{5}$$

$$w = \{K_1 2^{i-1} + K_2 2^{i-2} + \cdots + K_i 2^0\} \tag{6}$$

步驟三

計算簽章值，使用者 A 利用式(4)至(6)計算出簽章值 s_A，計算公式如(7)。

$$s_A = k_A^{-1}(h(\overline{m_i}) + n_A r_A) \bmod \alpha \tag{7}$$

步驟四

執行簽密計算，利用明文轉點的方式將 w 值以十進位表示及簽章 s_A 轉成點坐標，並以使用者 A 本身的隨機值 k_A 及收方 B 的公鑰 $n_B G$ 來實施加密成 C_0 如式(8)，定義 $\overline{C} = \{C_0, C_1, C_3, \cdots, C_l\}$，並將 w 值以二進位方式對應到 $C_0, C_1, C_3, \cdots, C_l$ 序列，如式(9)

$$C_0 = [f_{m2p}(w, s_A) + k_A \cdot n_B G] \tag{8}$$

$$\begin{aligned} C_1 &= [P_1 + C_0 + K_1 k_A \cdot n_B G] \\ C_2 &= [P_2 + C_1 + K_2 k_A \cdot n_B G] \\ &\vdots \\ C_l &= [P_l + P_{l-1} + K_i k_A \cdot n_B G],\ 2 \le l \le n \end{aligned} \tag{9}$$

步驟五

A 將 $\{id_A, cd_A, PK_A, Z_A, R_A, \overline{C}\}$ 送出給 B。

3.3　解簽密

步驟一

當 B 收到 A 所傳送過來的 $\{id_A, cd_A, PK_A, Z_A, R_A, \overline{C}\}$ 之後，先行驗證身分。

首先計算：

$$u_1 = ca_A^{-1} \tag{10}$$

$$u_2 = e_A \times u_1 \tag{11}$$

$$u_3 = x_{Z_A} \times u_1 \tag{12}$$

接著以認證中心的公開金鑰來驗證身分的正確性：

計算：

$$u_2 G + u_3 PK_{AS} = (v_x, v_y) \tag{13}$$

驗證：

$$x_{Z_A} = v_x \tag{14}$$

若(14)等式成立則確認傳送方的身分的確合法無誤。

步驟二

接著，依以下方式解簽章：

$$\begin{aligned} f_{m2p}(w, s_A) &= C_0 - R_A n_B \\ (w, s_A) &= f_{p2m}[f_{m2p}(w, s_A)] \end{aligned} \tag{15}$$

B 可用 n_B, R_A 解開 w, s_A 值。

步驟三

接著將 w 還原成 $\overline{K}$ 數列，還原方式如(16)，將其二進位表示的 w 值，對應 1 代表左旋，對應 0 代表右旋依序解開 $\overline{C}$ 可以取得 $\overline{P'}_i = \{P'_1, P'_2, \cdots, P'_n\}$ 執行點轉成明文的動作詳如式(16)及(18)。

$$w = \{K_1, K_2, \cdots, K_n\}_2 \tag{16}$$

$$f_{m2p}(w, s_A) = C_0 - R_A n_B \tag{17}$$

$$\overline{m'}_i = f_{p2m}\{P'_1, P'_2, \cdots, P'_n\} \tag{18}$$

相關運算說明如后：

$$(w, s_A) = f_{p2m}[f_{m2p}(w, s_A)]$$
$$P_1 = [C_1 - C_0 - K_1 R_A n_B]$$
$$P_2 = [C_2 - C_1 - K_2 R_A n_B]$$
$$\vdots$$
$$P_l = [C_l - C_{l-1} - K_i R_A n_B]$$

步驟四

還原明文，$\overline{m'}_i$ 為解密後所得到的多重文件集合如式(18)，再將多重文件的集合實施一次雜湊如式(19)。

$$\overline{m'}_i = m'_{11}, m'_{12}, m'_{21}, m'_{22}, \cdots, m'_{n1}, m'_{n2} \tag{19}$$

$$h(\overline{m'}_i) = m' \tag{20}$$

步驟五

驗證明文正確性，收方 B 計算：

$$m' \stackrel{?}{=} m \tag{21}$$

若(21)等式成立則收方所收之訊息正確無誤。

本研究之多文件簽密示意圖，請參閱圖 4。

4. 安全性及效益分析

本節將針對所提方法之安全性及效益分析進行探討。

4.1　安全性分析

本研究所提之簽密機制，其安全性植基於橢圓曲線離散對數難題，以下針對機密性（Confidentiality）、完整性（Integrity）、鑑別性（Authenticity）性不可否認性（Non-repudiation）等四項特性滿足狀況進行分析：

⑴機密性：對電子商務而言，機密性是指交易不能經由公共網路來追蹤及未經授權的中間人無法取得交易複本。電子商務的環境中，所有重要的訊息交換均需要保密，訊息在成功地送達目的地之後，除了稽核資訊外，所有存在於公共環

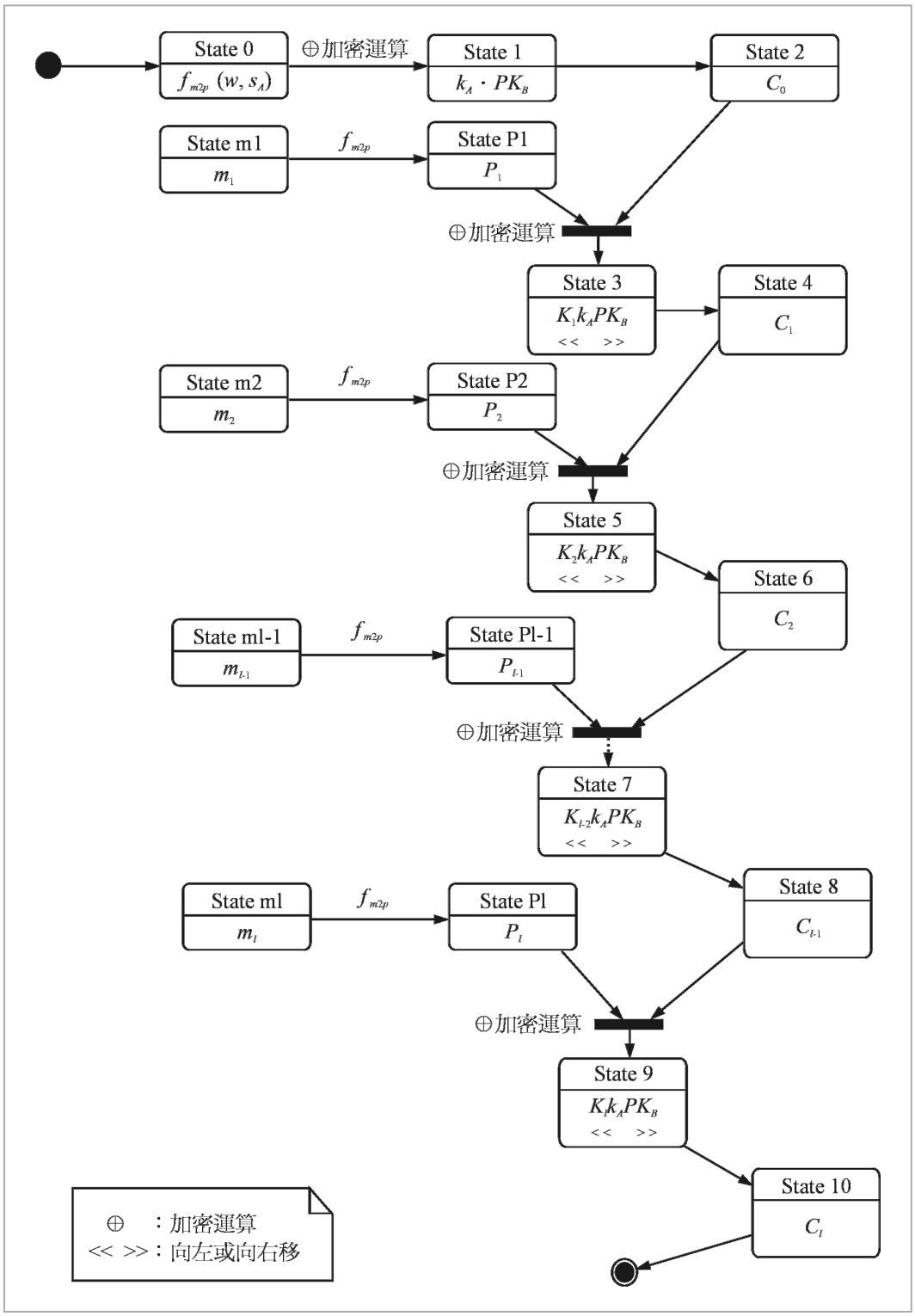

State 0
$f_{m2p}(w, s_A)$
⊕加密運算
State 1
$k_A \cdot PK_B$
State 2
C_0
State m1
m_1
f_{m2p}
State P1
P_1
⊕加密運算
State 3
$K_1 k_A PK_B$
<< >>
State 4
C_1
State m2
m_2
f_{m2p}
State P2
P_2
⊕加密運算
State 5
$K_2 k_A PK_B$
<< >>
State 6
C_2
State ml-1
m_{l-1}
f_{m2p}
State Pl-1
P_{l-1}
⊕加密運算
State 7
$K_{l-2} k_A PK_B$
<< >>
State 8
C_{l-1}
State ml
m_l
f_{m2p}
State Pl
P_l
⊕加密運算
State 9
$K_l k_A PK_B$
<< >>
State 10
C_l
⊕ ：加密運算
<< >>：向左或向右移

圖 4 系統簽密示意圖

境中的相關資訊會被刪除，且訊息的儲存必須在具備完善保護的系統下進行。本方法於網路中傳送之資料為 $\overline{C}$，如第三方於通訊過程中竊聽，則他必須面對公式(9)：$C_l = [P_l + P_{l-1} + K_i k_A \cdot n_B G]$，即破解橢圓曲線離散對數的難題。且因為是隨機產生的值，以二進位方式作擬亂排列，對應 1 區塊向右旋，對應 0 向左旋，達成加密區塊擬亂的效果，想要解開 w 同樣面臨解橢圓曲線離散對數的問題。

⑵完整性：電子文件的完整性必須依賴電子簽章的技術來實現，訊息在傳遞過程中不能被破壞或干擾，當電子文件被非法變更時，則數位簽章將無法通過驗證機制。舉例說明：傳送方欲透過網路傳遞電子公文，以自己的私鑰簽章，並將有簽章訊息的電子公文”A”傳送給接收方。然而，如果傳送方沒有針對該訊息進行完整性的保護，則非法攻擊者可能會將電子公文改為“AA”，如此將影響傳送者的權益。若是接收方發現該數位簽章驗證無效時，則表示此公文可能已被修改，因此可以拒絕受理。在本方法中，公式 (4)-(7)：(s_A, R_A)即為完成簽章文件的完整性，若非法攻擊者想要竄改明文來偽造簽章(s_A, R_A)而不被發現，或想要破解得知傳送方的私密金鑰，均必須面對橢圓曲線離散對數的難題。

⑶鑑別性：鑑別性是指訊息的接收方可以利用一些公開參數來驗證該訊息來源的合法性，以保證該訊息確實是由宣稱的送方所送來的。公開金鑰密碼系統，使用者的公鑰與密鑰有唯一的對應關係，因此藉由金鑰對可達到鑑別使用者身分的功能。數位簽章的機制設計，簽章產生必須使用簽署者的密鑰，驗證則要簽署者的公鑰，才能驗證該簽章的有效性，如果驗證者利用認證中心及傳送者的公鑰驗證所收到數位簽章為有效時，則表示此簽章與電子文件的確是由具有該公鑰的使用者所簽署。本方法在公式 (5-15)即說明此鑑別性的設計，對惡意的第三方而言，他必須面對破解單向雜湊函數及橢圓曲線離散對數問題。

⑷不可否認性：為了能防止傳送端否認曾經發出訊息給接收端，則接收端必須握有傳送端傳送過該訊息的證據或收方也不能否認已收到此訊息，有了上述這些特性，一個安全的電子交易行為才有辦法進行。本方法具有送方不可否認性，系統中若非送方本人無法完成簽章(s_A, R_A)，則後續的驗證亦無法完成，收方在檢驗送方簽章時，必須利用認證中心及送方之公鑰進行驗證，故可證明確實由送方 A 所簽署送出；另送方依認證中心公告的公開金鑰與收方彼此相互認證，再共同產生會議金鑰，方能執行後續的簽密程序，以公式(17)：$f_{m2p}(w, s_A) = C_0 - R_A \cdot n_B$ 為例，發送方將簽章及密文產生之後，傳給接收方，若雙方有人否認時，將會議金鑰提交認證中心即可仲裁，防止雙方的事後否認。對惡意的第三方而言，嘗試破解會議金鑰，以取得雙方的私密金鑰，同樣必須面對解橢圓曲線離散對數的難題。

4.2 效益分析

表 3 以簽章長度及運算基礎，比較三種常用的簽章演算法的安全性及優缺點，其他的應用類型其複雜度與原型相同而不加以補述。本研究所提出的橢圓曲線簽章方式，在相同的安全性度，只要提供較短的金鑰即可達到，這將影響簽章傳遞的時間與儲存的空間，RSA 數位簽章演算法為目前大部分軟體所採用，Elgamal 三明文簽章雖具有三份文件一次簽署之功能，然其簽章驗算速度與 RSA 雷同，皆不如本研究所採用的橢圓曲線數位簽密。表 4 為 Elgamal 三文件簽章與本篇簽密的比較表，從金鑰產生、運算與驗證等三方面可見本篇運算速度遠高於 Elgamal 三文件簽章方式。

▼ 表 3 RSA、Elgamal 及本研究之優缺點比較

演算法	RSA 多文件簽章	Elgamal 多文件簽章	本研究
安全理論	因數分解	離散對數	橢圓曲線離散對數
金鑰長度	較長	較長	較短
簽章方式	為資料匣（壓縮後）簽章	只能對 3 份文件簽章	能多份文件同時簽密
效率分析	次佳	欠佳	佳
安全分析	欠佳	次佳	佳
優點	容易說明，亦可同時用以加、解密。	可達到三文件一次簽章。	速度快、簽章小，可達成多文件一次簽密。
缺點	速度慢，簽章長度較大；單一簽署；加密演算法需附加。	速度慢，簽章長度較大；文件簽章不具彈性，無加解密功能。	理論不易理解，實現技術較複雜。

5. 結論

電子商務是建立在網際網路上的一種商業應用，WWW 讓電子商務能以比較低廉的成本，來從事比較大經濟規模的商業活動。而電子商務是否可以蓬勃發展，進而掌控未來的經濟命脈，則完全依賴各種資料安全技術的研究發展，以及安全交易架構之建立。近年來，政府正大力推展「產業電子化」政策，結合電子商務發展的趨勢，利用資訊科技與資訊管理兩項重要的觀念與技術，提升國內產業的競爭力。電子化交易要安全地進行就必須達到身分可認證及資料傳輸安全的要求，要達到這樣的要求就必須將簽章及加密的技術結合在一起，這樣的通訊技術我們稱為簽密法。在過去的數十年中，有許多根基於離散對數的簽章方法被提出，雖然這些方法都能完成簽章及加密動作，但在系統安全及執行效率上有改進的空間。由於橢圓曲線密碼演算法具有簡潔

▼ 表 4　Elgamal 與本研究運算速度比較

演算法		Elgamal 三文件簽章		本研究（以三份文件為例）	
比較項目		時間複雜度	概估	時間複雜度	概估
金鑰產生		T_{EXP}	$\approx 240T_{MUL}$	$5T_{ECCMUL}+2T_{ECCADD}+1hash$	$\approx 125T_{MUL}$
運算	簽章	$T_{EXP}+T_{ADD}+2T_{MUL}+1hash$	$\approx 243T_{MUL}$	$(T_{ECCADD}+T_{ECCMUL})l+1hash$ $2<l<n$	$\approx 88.36T_{MUL}$
	加密	無	無	$2T_{ECCMUL}+T_{INVS}+1hash$	$\approx 88T_{MUL}$
驗證		$3T_{EXP}+T_{MUL}$	$\approx 723T_{MUL}$	$T_{ECCADD}+T_{ECCMUL}+2hash$	$\approx 31.12T_{MUL}$
備註		T_{ECCMUL}：進行一次 ECC 乘法運算所需時間 $\approx 29T_{MUL}$		T_{INVS}：進行一次模式乘法反元素運算所需時間 $\approx (0.843\ln(p)+1.47)T_{DIV}$	
		T_{EXP}：進行一次模式指數運算所需時間 $\approx 240T_{MUL}$		T_{ECCADD}：進行一次 ECC 加法運算所需時間 $\approx 0.12T_{MUL}$	
		T_{ADD}：進行一次模式加法運算所需時間		T_{MUL}：進行一次模式乘法運算所需時間 $\approx T_{DIV}$	
		$hash$：進行一次雜湊運算所需時間 T_{MUL}			

及效率之優點，近來是密碼學領域的專家學者的一個研究熱點，本研究將應用橢圓曲線密碼機制，提出一個新的多文件簽密法，除了具有機密性、完整性、鑑別性及不可否認性等安全需求，同時具備雪崩效應。本研究可達成的貢獻如后：⑴簽章認證與加密功能可於一次通信步驟中完成；⑵在認證中心離線情況下系統成員依然能夠進行對方身分認證；⑶滿足機密性、完整性、鑑別性、不可否認性等基本安全需求；⑷於密文中混入擬亂資料增加破密困難；⑸即使遭受部分截獲第三方將無法藉由片段密文進行破密工作；⑹視傳輸頻寬，可彈性調整一次簽密的文件數量。

參考文獻

蘇品長（2007），《植基於 LSK 和 ECC 技術之公開金鑰密碼系統》，博士論文，長庚大學電機工程系。

蘇品長、楊威儀、蔡建華、林瑞興（2009），“應用於國軍網路環境之多重文件加密機制研究與設計”，刊於《2009 年第十七屆國防管理學術暨實務研討會》，117-137。

許建隆（2000），“數位簽章”，《中央研究院週報第 9 期》。

張真誠、林祝興（2006），《資訊安全-技術與應用》，台北，全華出版社。

Chor, B. and Rivest, R. L. (1988), "A knapsack type public key cryptosystem based on arithmetic in finite fields," *IEEE Transaction on Information Theory*, 34, 901-909.

Chou, C. F., Cheng, W. C., and Golubchik, L.(2010), "Performance study of online batch-based digital signature schemes," *Journal of Network and Computer Applications*, 33 (2), 98-114.

ElGamal, T. (1985), "A public key cryptosystem and a signature scheme based on discrete logarithms," *IEEE Transactions on Information Theory*, 31(4), 469-472.

Horster, P., Petersen, H., and Michels, M. (1994), "Meta-ElGamal signature schemes," in *Proceedings of the 2nd ACM Conference on Computer and communications security*, 96-107.

Lim, C. H. and Lee, P. J. (1994), "Security of interactive DSA batch verification," *Electronics Letters*, 1592-1593.

Lin,C. H., Hsu, R. H., and Harn, L. (2005), "Improved DSA variant for batch verification," *Applied Mathematics and Computation*, 169(11), 75-81.

Naccache, D., M'Raïhi, D., Raphaeli, D., and Vaudenay, S. (1994), "Can DSA be improved: Complexity trade-offs with the digital signature standard," *Lecture Notes in Computer Science*, 950, 85-94.