強化機敏文件管理 — 具身分認證暨一次文件加密之設計

作者簡介

梁榮哲中尉，國防大學管理學院51期；曾任陸軍通信電子資訊學校學生五中隊區隊長，現為國防大學管理學院資訊管理系15期研究生。

蘇品長中校，國防管理學院33期資管系、長庚大學電機工程博士；曾任國防部主計局財務官、資訊官、統計官、資訊督導官，現為國防大學管理學院資訊管理系助理教授。

提要

一、現行國軍所使用的電子公文交換雖說已完成電子化的初步目標，但距離真正的「無紙化」尚有努力的空間，其原因不外乎是安全性的考量，透過本研究之加密方法，可使得國軍電子公文交換及各項重要軍事文件的傳輸更加安全及迅速，達成保密及速度雙贏的效果。

二、橢圓曲線公開金鑰系統的研究已經發展20多年，目前若要達到合理的安全等級，本研究利用橢圓曲線系統只要採用160位元模數即可，而RSA及ElGamal則須採用1024位元模數，才能達到相同等級的安全強度。

三、本研究可一次加密多份文件且具身分認證之效，利用背包問題混合橢圓曲線密碼系統使密文具有雪崩效應，直接增加密文破解的難度，進而提高網路傳送雙方更高的安全性，將多文件的資訊藉由混淆機制，將其變成一份密文來傳送。

關鍵詞：橢圓曲線、用戶識別、背包問題

前　言

本研究打破以往檔案單一文件單一加密的方式，研究出可將多份文件一次加密且具身分認證之效，利用背包問題混合橢圓曲線密碼系統使密文具有雪崩效應，直接增加密文破解的難度，進而提高網路傳送雙方更高的安全性，將多文件的資訊藉由混淆機制，將其變成一份密文來傳送。網際網路盛行與資訊科技蓬勃發展更使得資訊交換的機率日趨頻繁，為人們帶來生活上的便利，在此環境傳輸資訊時很容易遭到他人的竊取、偽冒、竄改或偷窺等風險，更由於電子檔案具有易於複製、傳輸的特性，一旦被獲取，竊取者具有完全的權利，這些因素都將會增加使用者對於傳送及軍中機敏性文件存放安全疑慮，有鑑於此，提出一種具身分認證且可一次文件加密之研究，利用橢圓曲線密碼系統金鑰長度，可遠較諸如RSA等其他公開金鑰密碼系統為小，以及背包式系統(Knapsack Cryptosystems)其加解密(或簽署驗證)速度非常快等特性，設計一套植基於ECDLP及Knapsack難題的公開金鑰密碼機制。

密碼技術重要性

網際網路盛行與資訊科技蓬勃發展，為國軍帶來任務上的便利，如何使得檔案在傳輸過程中維持其機密性與完整性，已成為備受重視的課題，在現今世界能將日常生活所需的資訊，藉著電腦處理、儲存，並透過網絡環境交換，可說都源自於電腦與網際網路蓬勃發展之故，但也由於網際網路是四通八達的，且資訊交換轉送的過程必經由許多中繼電腦才能到達目的地。因此，必須防止心懷不軌的人在中途攔截使用者資訊及資料；據行政院主計處電子處理資料中心調查統計結果顯示，在2010年間，有2,204家機構遭遇使用者資料遭竊或被破壞，顯見享受科技帶來的方便之餘，個人電腦、行動裝置等端點防護措施不足的安全問題亦逐一浮現。

另外，還有許多使用者私密的電子檔案，往往在使用者不知情的情況下就被洩漏、盜取。因此，網路安全應用上便著重於資料不被攔截、竄改、洩漏、盜取甚至要做到即使遭到洩漏、盜取，竊密者亦無法讀取其資料的目標，而這些都必須仰賴於「密碼技術」。密碼技術中，「加密」(Encrypt)就是我們將原來的文件檔案裡的每一個字，透過許多數學公式的運算，產生另一份讓人無法輕易瞭解其內容，再加上內容轉換成不具任何意義或與本身內容無關的代碼，這個處理的過程就稱之加密。而加密的目的在於提高資訊內容的隱私性，讓要傳遞的資訊在公開網路傳輸過程中，就像是加了信封的信件，具有保護資訊的外衣，只有特定的人能讀取內文資訊，而非像明信片一樣任何人均可讀到該信件之內容，而這些就是密碼系統之安全性功能。

但究竟什麼樣的密碼系統才是一個安全的密碼系統呢？Shannon[1]在1949年就提出密碼系統安全的定義應包含理論安全(Theoretical Security)與實際安全(Practical Security)兩種。所謂的理論安全是指不

1　於下頁。

管破密者截獲多少密文加以分析，其破密的難度和直接猜測明文是一樣的。而Shannon所謂的實際安全是假設每一密碼系統在給定n位密文時，均有一破解此系統的最少工作次數，稱為此系統的工作特性W(n)(Work Characteristic)。若一系統的W(n)大到使得具有有限計算能力及記憶體的破密者無法在合理的時間內破解此系統，則此系統即可稱為實際安全或計算上安全(Computational Security)。

就現今的公開金鑰密碼系統而言，目前都無法滿足理論安全，因此，如何設計出符合實際安全與維持良好系統效率的密碼系統則成為密碼學家努力的目標。所以，為強化系統之安全，Harn[2]於1994年提出一個植基於因式分解和離散對數的簽章系統，Lee和Hwang[3]於1996年改進Harn的方法以避免偽造攻擊，之後陸續有專家學者針對不同的因式分解與離散對數問題設計不同的演算機制，以強化系統安全性。可是從已發表植基於因式分解和離散對數的雙重難題的文章中，我們發現大多數都只在強化演算法安全性上，而對於演算效率卻不是很重視，因此，如何兼具安全性與效率性，是為本研究著眼的重點。

在本研究中，我們提出一種植基於橢圓曲線離散對數及背包理論之密碼機制[4]運用於文件加密上之加密演算法，主要就是希望藉由結合離散對數與資訊混淆之特點來增加密碼系統之安全性，打破以往檔案單一文件單一次加密的方式，將多文件的資訊藉由混淆機制，將其變成一份密文，並使其密文具有雪崩效應，令竊密者即使截獲部分資訊亦無法得知明文資訊，成為一套能增加文件解密之難度，卻不會降低其演算效率之加密法。在本文中分為三大部分論述，第一部分為相關理論簡介；第二部分為本文所提之新的一次文件加密方法介紹；第三部分則分析本研究系統之優點及其安全性。

橢圓曲線及背包系統簡介

一、橢圓曲線公開金鑰密碼系統

於1985年由Miller[5]及1987年 Koblitz[6]分別提出將橢圓曲線(Elliptic Curve)用來實作公開金鑰密碼系統。至今已經發

1 Shannon, C.E., "Communication Theory of Secret Systems", Bell System Technical Journal, Vol.28, No.4, 949,pp.656～7153.

2 Harn, L., "Public-key cryptosystem design based on factoring and discrete logarithms", Computers and Digital Techniques, IEEE Proceedings,Vol. 141, Issue 3,1994, pp. 193～195.

3 Hwang, T. and Chen J. L.(1994), "Identity-based conference key broadcast system", Computers and Digital Techniques, IEEE Proceedings, Vol. 141, Issue 1, pp. 57～60.

4 蘇品長，〈植基於LSK和ECC技術之公開金鑰密碼系統〉，長庚大學電機工程研究所博士論文，2007年。

5 Miller, V. S., "Use of Elliptic Curve in Cryptography", Advance in Cryptography- Crypto '85, New York: Spring-Verlag, 1985, pp.417～426.

6 於下頁。

展20多年。在這期間，已有許多關於ECC(Elliptic Curve Cryptosystem)的研究成果先後被發表。ECC演算法和RSA演算法相比，其優點為ECC的加密強度更強(如表一)。

表一　RSA與ECC相同安全度金鑰比較表

相同安全度之金鑰長度(bits)					
RSA	512	1024	2048	3072	7680
ECC	112	163	224	256	384
Key	1:5	1:6	1:9	1:12	1:20

資料來源：本研究

由表一可知ECC密鑰長度為163位元時，RSA密鑰長度必須提高到1024位元才可達到一樣的安全性。也正因為如此，ECC的應用越來越廣泛，將漸漸取代RSA演算法，成為主要的加密標準，以下為橢圓曲線演算法。

橢圓曲線的一般通式為$y^2+axy+by=x^3+cx^2+dx+e$，其中a、b、c、d、e是實數。在橢圓曲線中，點加法運算是經過特別定義的，除此之外，也另外定義一個無窮遠點O，假使一條直線與此橢圓曲線相交於三點，則此三點的和為無窮遠點O。

如果q是大於3的質數，則在Galois Field $E(F_q)$中，橢圓曲線的通式如下：$y2=x3+ax+b \bmod q$其中 $0 \le x \le q$，a、b為小於q的正整數且$4a^3+27b^2 \bmod q \neq 0$ 。我們假設下面兩點 $P(x_1,y_1)$及$Q(x_2,y_2)$ 為橢圓曲線群$E(F_q)$中的兩個點，則此橢圓曲線群 $E(F_q)$中的點加法運算為如下定義：

(1)$P+\infty=\infty+P=P$

(2)如果，$x_1=x_2$，$y_1=-y_2$，$p=(x_1,y_1)$，$Q=(x_2,y_2)=(x_1,-y_1)=-P$且$P+Q=O$

(3)如果$P \neq Q$則$P+Q=(x_3,y_3)$,$x_3 \equiv \lambda^2-x_1-x_2 \bmod q$, $y_3 \equiv \lambda(x_1-x_3)-y_1 \bmod q$

如果$x_1 \neq x_2$則 $\lambda = \dfrac{y_1-y_2}{(x_2-x_1)}$,若$x_1=x_2$且$y_1 \neq 0$則 $\lambda = \dfrac{(3x_1^2+a)}{2y_1}$

(4)在橢圓曲線的求點運算中，若要計算$2P$則等同計算$P+P$，相同的，若要計算$3P$則等同計算$3P=2P+P$，假設一個橢圓曲線是屬於E_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出解答。橢圓曲線密碼系統的另一個優點是其加密的密鑰長度短，在同樣的安全度之下，它僅需要較小的密鑰長度，相同地，在同樣的密鑰長度下卻擁有更高的安全性。

二、背包式公開金鑰密碼系統

自從1976年Diffie及Hellman[7]向世人介紹了公開金鑰密碼的概念之後，不久之後就有了公鑰密碼的實作系統被提出，其

6　Koblitz, Neal., "Elliptic Curve Cryptosystems", Mathematics of Computation American Mathematical Society, Vol. 48, 1987, pp.203～209.

7　Lagarias J. C., "Knapsack-type Public Key Cryptosystems and Diophantine Approximation, "Proc. CRYPTO'83, Springer-Verlag, 1984, pp.3～24.

中一個較著名的為Merkle-Hellman背包式公鑰密碼系統。[8]而背包問題是一種組合優化問題，假設我們有一個背包，它所能裝載的物品重量是固定的，但現在我們有非常多的物品，每個物品的重量都不一定相同，我們將如何選擇最合適的物品放置於給定背包中。它的數學描述是給定一個自然數數列$B=\{b1,b2,....,bn\}$及一數S，是否存在一子數列$B' \subseteq B$，其中$B'=(b'_1,....,b'_m)$，使得$\sum_{i=1}^{m} b'_i = S$？。

背包問題已經被證實是一個NP-Complete問題。並無法在多項式時間內解決，而且背包式密碼系統的最大優點為加解密(或簽署驗證)速度相當快，故將其套用在密碼系統的設計上有其優點。但是這種密碼系統被發現是不安全的，因為這種密碼系統中的「超增數列」被許多學者發現其弱點並對此展開攻擊，第一位成功攻擊Merkle-Hellman背包式公開金鑰密碼系統的學者是Shamir，[9]在Shamir 攻擊之後，陸陸續續又有許多學者提出攻擊的方式，其中較為特別的是一位學者Brickell[10]所提出的攻擊方式，它的攻擊方式稱為低密度攻擊，只要是背包式密碼系統中的超增數列中的數值，在整個數列中所占的密度太低，就有可能被猜出來，而背包式密碼系統也就可能因此瓦解。

以下介紹Merkle-Hellman的背包式公鑰密碼系統：給定一正整數集合$\{b_1,b_2...,b_k\}$及一個和S，計算$m_i \in \{0,1\}$(for i = 1, 2, ..., k)，使得$S=\sum_{i=1}^{k} b_i \times m_i$超增背包(Super-increasing Knapsack)：

一數列$< a_i >$，若$a_{i+1}=\sum_{j=1}^{i} a_j$，則$< a_i >$稱為超增數列。

(一)金鑰產生 (每k-bit為一次加解密的單位)

步驟一：任選一個超增數列$< a_i >$(for i = 1 to k)及一整數n，其中$n > \sum_{i=1}^{k} a_i$。

步驟二：任選一數e滿足gcd(e,n)=1，並求出$d=e^{-1} \bmod n$。

步驟三：將$< a_i >$轉換成$< b_i >$，使得$b_i=e \times a_i \bmod n$。

步驟四：$< b_i >$為公鑰，d為私鑰。

(二)加密程序

步驟一：令明文M=$m_1 m_2 ... m_k$ (每一個m_i為一個位元)。

步驟二：計算密文$C=\sum_{i=1}^{k} b_i \times m_i$。

(三)解密程序

步驟一：計算$C'=C \times d \bmod n = C \times d = \sum_{i=1}^{k} b_i \times m_i \times d = \sum_{i=1}^{k} a_i \times m_i (\bmod n)$

步驟二：利用超增數列求解演算法，代入C'求出m_i。

8 Merkle, R.C. and Hellman, M., "Hiding Information and Signatures in Trap-door Knapsacks", IEEE Trans. Inform. Theory, Vol. IT-24, 1978, pp. 525～530.

9 Shamir, A., "A Polynomial Time Algorithm for Breaking the Basic Merkle-Hellman Cryptosystem", Proceedings of the Twenty-Third Annual Symposium on Foundations of Computer Science, IEEE, 1982,pp. 145～152.

10 Brickell, E.F., "Breaking Iterated Knapsacks", Advances in Cryptology: Proceedings of Crypto84, G.R. Blakely and D. Chaum. eds., Springer-Verlag, 1985,pp. 342～358.

設計具身分認證且可一次文件加密之研究

本節將基於橢圓曲線離散對數(ECDLP)及背包問題(Knapsack)設計一個具身分認證且可一次文件加密之研究。方法分為四個階段：初始化階段、身分認證階段、加密階段、解密階段。詳述如下：

一、初始化階段

(一)參數選擇與設計

令E代表一個在有限域$GF(q)$中的一條橢圓曲線，我們假定這條橢圓曲線的各項參數選擇都符合安全橢圓曲線的參數設定[11]，接著我們在這條曲線上隨意找一個點G當作基點，表二為本研究使用之符號說明。

表二　本研究使用之符號說明

項次	符號	說明
1	ID_A、ID_B	A、B的ID資訊
2	SK_A,n_A、SK_B,n_B	A、B的私鑰，n_A、n_B為F_q中任意選擇之整數
3	PK_A,n_AG、PK_B,n_BG	A、B的公鑰為EC上的點
4	SK_{AS},n_{AS}	伺服器的私鑰，n_{AS}為F_q中任意選擇之整數
5	$PK_{AS},n_{AS}G$	認證伺服器的公鑰，為EC上的一個點
6	$SK_{A2}=\{a_1,a_2,\cdots,a_n\}$ $SK_{B2}=\{b_1,b_2,\cdots,b_n\}$	於通信時，A、B所產生用於通信之私鑰，為一組整數向量
7	$PK_{A2}=\{a_1G,a_2G,\cdots,a_nG\}$ $PK_{B2}=\{b_1G,b_2G,\cdots,b_nG\}$	於通信時，A、B所產生用於通信之公鑰，為一組EC上之點集合
8	CA_A、CA_B	使用者A、B之憑證
9	M_1, M_2	明文訊息
10	m_i，m_j	明文之分解區塊
11	$\overline{X}$	為一0、1數列，用於記錄文件存放位置
12	W	明文之0、1背包值
13	E	本研究中之橢圓曲線
14	E()	橢圓曲線加密函數
15	H()	雜湊函數，SHA512
16	M_1, M_2	解密後所得之明文
17	$\overline{R}_n=\{R_1,R_2,\cdots,R_n\}$	明文加混淆訊息所產生之點集合
18	$C=\{C_0,C_1,C_2,\cdots,C_n\}$	密文之點集合

資料來源：本研究

11　肖攸安，《橢圓曲線密碼體系研究》，華中科技大學出版，2006年。

(二)本研究循序示意圖

以使用者A與B為範例，本研究循序示意圖如圖一所示。

二、身分認證階段

在加密階段之前，雙方必須先確認彼此身分，假設使用者B現在要透過網路傳送訊息M_1, M_2給使用者A，就必須先身分認證。以下為使用者A、B雙方認證通信步驟。

Step1：雙方向認證中心註冊身分

A將自己的ID_A、PK_A，傳送至AS註冊。B將自己的ID_B、PK_B，傳送至AS註冊。

Step2：認證伺服器AS簽發雙方憑證

認證伺服器AS接收到使用者A的ID_A、PK_A之後，以$K_{AS\text{-}A}$為共同金鑰建立共同私密通道，共同金鑰產生方式如下：

$K_{AS\text{-}A} = n_{AS} n_A G$) ..... (1)

接著AS將A的憑證CA_A傳送給A，並同時將PK_{AS}也傳給A，憑證內容如下式：

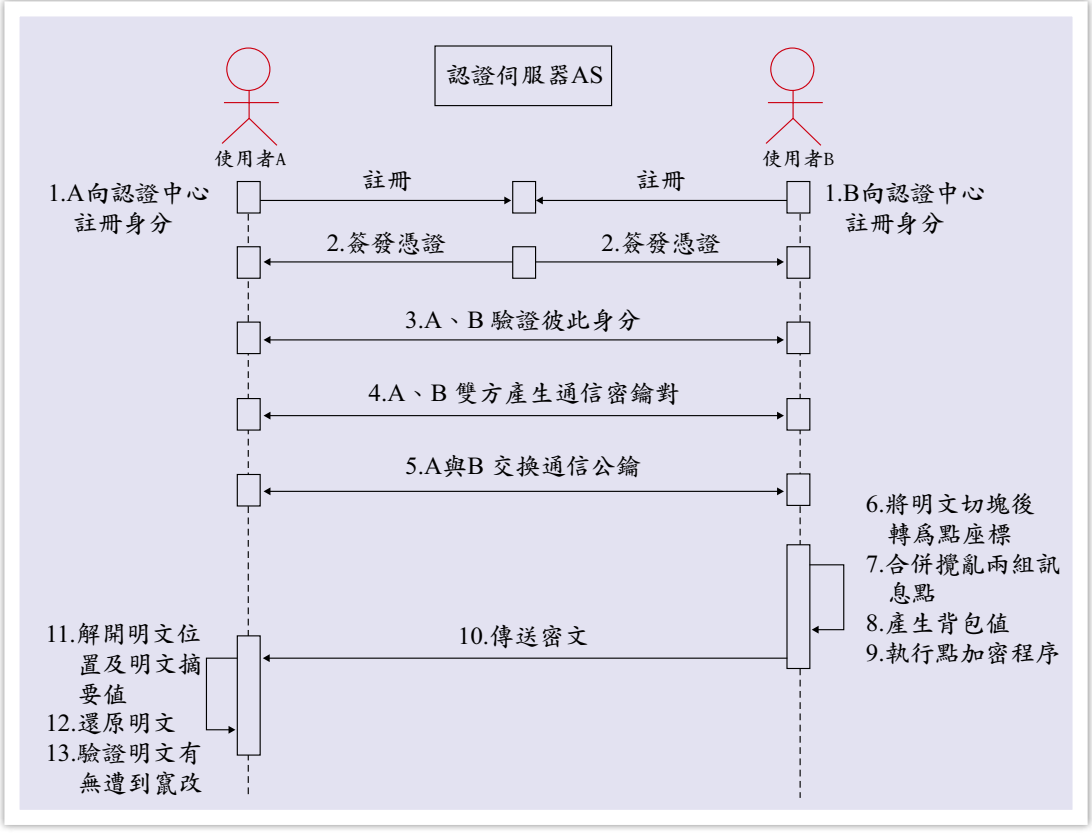
$CA_A = E_{SK\ AS}(ID_A, PK_A)$ ..... (2)

使用者B的作法同使用者A。

Step3：A、B驗證彼此身分

A將$CA_A = E_{SK\ AS}(ID_A, PK_B)$傳送給B，而B將$CA_B = E_{SK\ AS}(ID_B, PK_B)$傳送給A。此時雙方利用AS之公鑰$PK_{AS} = n_{AS}G$將對方的身分認證資訊解開，此時便可獲得對方

圖一　本研究循序示意圖



資料來源：本研究

之ID並可利用對方公鑰進行私密通訊。

Step4：A、B雙方產生通信密鑰對

A的通信私鑰如式(3)，其中之數列為A任選整數。

$SK_{A2}=\{a_1,a_2\cdots,a_n\}$ .....(3)

A的通信公鑰如式(4)(為一組點序列)。

$PK_{A2}=\{a_1G,a_2G,\cdots a_nG\}$ .....(4)

B的通信私鑰如式(5) ，其中之數列為B任選整數。

$SK_{B2}=\{b_1,b_2,\cdots b_n\}$.....(5)

B的通信公鑰如式(6)(為一組點序列)。

$PK_{B2}=\{b_1G,b_2G,\cdots b_nG\}$ .....(6)

Step5：A、B交換通信公鑰

當A、B完成相互身分驗證後，利用共同金鑰如式(7)建立私密通道，雙方利用私密通道將通信用公鑰傳送給對方。

$K_{A\text{-}B}=n_An_BG$.....(7)

三、加密階段

圖二為加密階段示意圖，而加密步驟從Step6至Step10。詳述如下：

Step6：圖二中步驟1與2將明文切塊後轉為點座標

B將訊息M_1 ,M_2拆解成2n個m_i及n_j區塊，並將每兩個m_i及n_j區塊編碼成為點，$P_i=(m_i,m_{i+1}),1\le i\le 2n\text{-}1$，及$Q_j=(n_i,n_{i+1}),1\le j\le 2n\text{-}1$，將文件轉換為兩組點序列，如式(8)(9)。

$M_1\rightarrow\overline{P}_i=\{P_1,P_2\cdots P_j\}$ .....(8)

$M_2\rightarrow\overline{Q}_i=\{Q_1,Q_2\cdots Q_j\}$ .....(9)

Step7：圖二中步驟3合併攪亂兩組訊息點

將P_i點依序按任意間隔插入Q_j點序列中成為R_n點序列，混合完成後之點序列以(10)式表示。

$\overline{R}_n=\{R_1,R_2\cdots R_n\}$，其中$i+j=n$....(10)

Step8：圖二中步驟4產生背包值

註記M_1位置，製造數列如式(11)

$\overline{X}_n=\{x_1,x_2\cdots x_n\},xi\in\{0,1\}$ .....(11)

對應$\overline{R}_n$點序列，若$\overline{R}_n$中的點為P_i所生成，則$x_1=1$，否則$x_1=0$。

計算M_1所在之背包值如式(12)

$W=\{x_1.2^{n-1}+x_2.2^{n-2}+\cdots+x_n.2^0\}$ ....(12)

Step9：圖二中步驟5執行點加密程序

B將W、$H(M)$值及點序列$\overline{R}_n=\{R_1,R_2\cdots R_n\}$以下列方式加密如式(13)。

$C=\{C_0,C_1,C_2,\ldots C_n\}$.....(13)

密文點$C_0,C_1,\ldots C_n$如式(14)(15)(16)

$\overline{C_0}=[(W,H(M_1,M_2))+(n_Bn_AG)]$ ....(14)

$C_1=[R_1+C_0+(n_Ba_1G)]$.....(15)

$C_n=[(R_n+R_{n-1})+(n_Ba_nG)]$,$2\le n$ ..(16)

Step10：傳送密文

B將加密訊息C送出。

四、解密階段

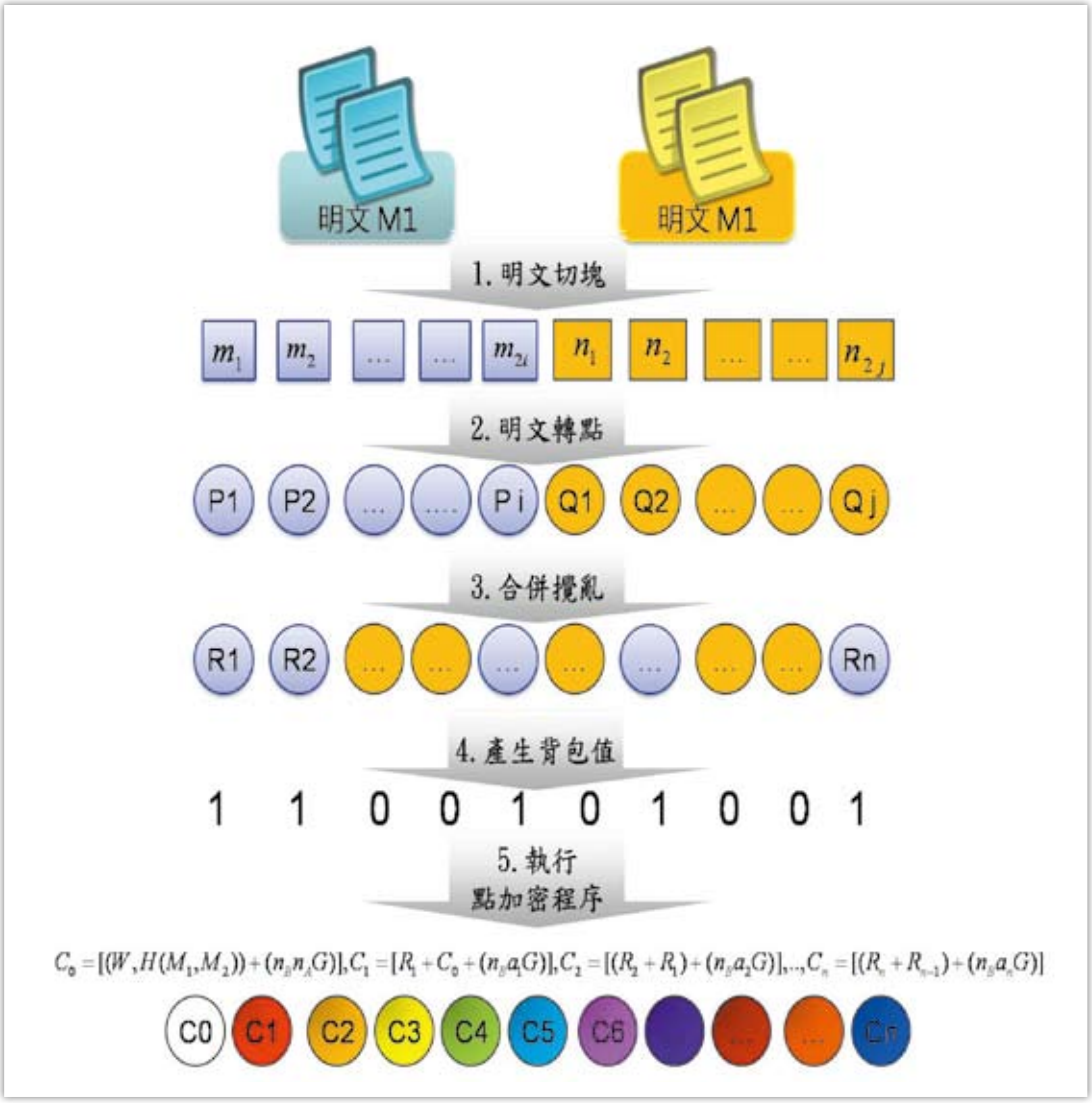
圖三為解密階段示意圖，而解密步驟從Step11至Step13。詳述如下：

Step11：圖三中步驟1解開密點

解開明文位置及明文摘要值(解開密點) A收到B送來的上述公式(14)$C_0=[(W,H(M_1,M_2))+(n_Bn_AG)]$之後，A以自身的私鑰$SK_A=n_A$乘上B的公鑰$PK_B=n_BG$得$n_An_BG$並將$C_0$減去點$n_An_BG$即可得到點$(W,H(M_1,M_2))$，接著將W還原成2進位數，即可得知$M_1$存放位置。

Step12：圖三中步驟2、3、4解開密點還

圖二　加密階段示意圖



資料來源：本研究

原明文(挑出P、Q點、轉換點資訊、組合明文)

解R_n點序列，運算方式如(17)(18)。

$R_1=C_1-(a_1.n_BG)-C_0$ ……(17)

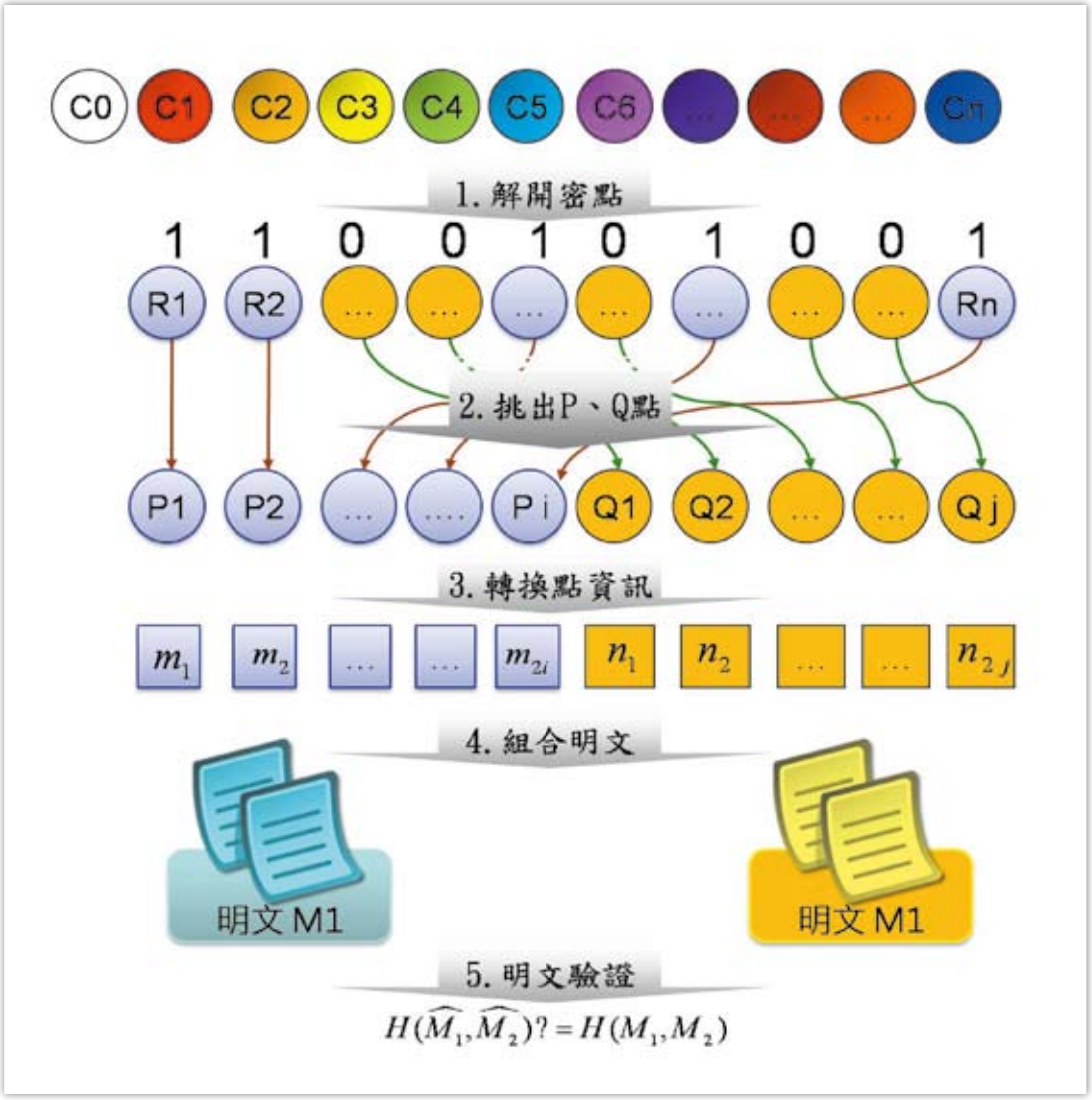
$R_n=C_n-(a_n.n_BG)-R_{n-1}$, $2\leq n$ ……(18)

當完成所有R_n點解密後，再利用$\overline{X}$註記之資訊，挑出P_1及Q_1後還原成明文區塊m_i及n_j，再將所有m_i及n_j合成訊息R_i, Q_j。

Step13：圖三中步驟5驗證明文有無遭到竄改(明文驗證)

假設還原後明文之摘要值為$H(M_1,M_2)$，若要確定所得之明文是否正確，只要與密文中$H(M_1,M_2)$值相比，如(19)式若不相等代表密文在傳送過程中遭到竄改；若正確無誤則代表

圖三　解密階段示意圖



資料來源：本研究

密文正確解密為明文並未遭竄改。

$$H(M_1,M_2)=H(M_1,M_2) \cdots\cdots\cdots\cdots (19)$$

安全性分析

本研究所提之加密機制，其安全性主要植基於橢圓曲線離散對數問題(Elliptic Curve Discrete Logarithm Problem, ECDLP)、背包問題(Knapsack Problem)及單向雜湊函數(One-Way Hash Function, OWHF)，根據ISO組織所提之資訊系統安全管理需求[12]包含資訊之機密性、完整性、不可否認性、鑑別性、可歸責性、可用性與可靠性。以下針對與密碼系統有關之安全需求

12　於下頁。

進行討論。

一、機密性

機密性指的是資料不得被未經授權之個人、實體或程序所取得或揭露的特性。本系統中傳輸之密文使用了橢圓曲線公開金鑰之加密方法，第三方若竊聽傳輸之密文，可獲得如(13)式$C=\{C_0,C_1,C_2,....C_n\}$中之密文點，第一個點$C_0$點，如(14)$C_0=[(W,H(M_1,M_2))+(n_Bn_AG)]$式，要破譯此式第三方將面對解橢圓曲線離散對數問題。而要解開後續的每一個密文點如(16)$C_n=[(R_n+R_{n-1})+(n_Ba_nG)]$,$2\leq n$中的C_n點，必須已經正確完成C_0點之解譯之後才能繼續解譯後續之密文點，而每一個密文點皆依存到之前的密文點，必須剛好成功破解前面的$n-1$個點，才能夠破譯第n個點，而破譯每一個密文點都是由不同鍵值所加密的，使得破譯困難度成線性增加，竊聽者面對線性增加的解橢圓曲線離散對數困難度，使得本系統密文之機密性得以確保。

二、完整性

完整性指的是對資產之精確與完整安全保證的特性。本系統中之密文中包含了明文之摘要值，當接收方完成解譯密文後，可將所得明文利用(19)$H(M_1,M_2)=H(M_1,M_2)$式計算其摘要值是否與密文中之摘要值相同，若相同代表所得明文為原始加密之明文。若第三方截聽密文後想要變造原始明文，則他必須同時變造明文以及明文摘要值，若他無法破解明文，隨意捏造密文傳送至接收方，則接收方經由計算其明文摘要值即可得知密文是否遭竄改，這使得本系統密文之完整性得以確保。

三、不可否認性

不可否認性指的是對一已發生之行動或事件的證明，使該行動或事件往後不能被否認的能力。系統中密文如(14)$C_0=[(W,H(M_1,M_2))+(n_Bn_AG)](15)C_1=[R_1+C_0+(n_Ba_1G)]$ (16)$C_n=[(R_n+R_{n-1})+(n_Ba_nG)]$,$2\leq n$式使用傳送方之私鑰n_B加密，接收方未來必須利用傳送方之公開金鑰才能將密文解開。對於公開金鑰密碼系統來說，公鑰及私鑰是成對的，而且是具有唯一性的，也就是說使用者B用其私鑰加密日後也僅能以其公鑰進行解密。故當使用者B將明文加密傳出後，他便無法否認密文是經由他加密的，若第三方想要藉由傳送方使用者B之公鑰n_BG推斷其私鑰n_B進而假造B的身分進行加密傳輸，則破譯者會面對橢圓曲線離散對數之問題，這使得傳送方發出密文具有不可否認性。

四、背包值部分無法進行低密度攻擊

在破解背包式密碼的學者中以Lagarias[13]之低密度方法最著名。一個背包式密碼系統其背包序列$B=\{b_1,b_2,...b_n\}$的密度定義為$n/\log2(\max(b_i))$，幾乎所有密度小於0.645的背包序列，低密度攻擊法均可破解。原始的背包式密碼系統，容易遭受到低密度攻擊，使得背包值遭到

12 ISO,2005, Information technology - Security techniques - Code of practice for information security management, ISO/IEC 17799:2005-06-15, 2.6,pp. 1.

13 Lagarias J. C. (1984), "Knapsack-type Public Key Cryptosystems and Diophantine Approximation, "Proc. CRYPTO'83, Springer-Verlag, pp.3～24.

破解，致使密碼系統崩潰。但本研究所提之方法，捨棄超增數列來作為加密資料之媒介，背包值改以利用橢圓曲線加密，因此，以往的低密度攻擊對本方法並不能產生威脅，所以本方法可摒除原始背包式密碼系統之缺點。

五、密文具雪崩效果

本方法先將明文在加密前分割成若干點，求算密文時將每一點導入後一點作橢圓曲線點相加運算如(16)$C_n=[(R_n+R_{n-1})+(n_Ba_nG)]$,$2\leq n$式，使得連續的兩點間具有依存關係。而且每一點都使用接收方的通信公鑰來作加密，每一個點都擁有不同的加密密鑰，因此，在傳送過程中若遭部分截獲，第三方並無法由部分密文解讀任何資訊。

效益評估

一、安全性提升

要破解本方法之密文除了必須面對橢圓曲線離散對數之難題外，同時還必須猜測明文所在位置如(14)$C_0=[(W,H(M_1,M_2))+(n_Bn_AG)](16)C_n=[(R_n+R_{n-1})+(n_Ba_nG)]$,$2\leq n$式才有辦法成功破解密文還原明文。也就是對每一密文點來說，竊聽者將會面對解橢圓曲線離散對數的困難。竊聽者若要判斷原文之密文點藏身處，則他將會面對解背包問題。經由結合橢圓曲線與背包問題，使得本加密機制擁有更高的安全性。

二、可離線之身分認證

本加密機制另加入了認證伺服器AS，系統中需要傳輸資料的各方，將本身的身分資料送交AS註冊後獲得憑證如(2)$CA_A=E_{SK_{AS}}(ID_A,PK_A)$式，日後當需要傳輸資料時，傳收雙方傳輸前先驗證對方的憑證。因為註冊憑證階段已獲得認證伺服器公鑰，所以當對方將他的憑證如(2)式中的CA_A因為是由認證伺服器AS私鑰加密而成，所以此時只要拿出認證伺服器AS的公鑰，即可解開對方憑證中的身分資料。所以本系統之身分驗證不需要隨時保持與伺服器連線的狀態，只要有保存認證伺服器AS的公鑰，即使離線狀態接收方仍可確認傳送方的身分。

三、單一文件加密與一次文件加密法之分析

本研究可將多份文件一次加密且具身分認證之效，其優缺點整理如表三所示；另與Elgamal加密兩份文件與本研究方法運算時間複雜度比較如表四所示。

結　　論

本研究中提供快速運算且安全性高的演算法，增加機率式的認證程序達到快速建立並確保加解密過程中的資訊保護，確保使用者的私密資訊及位置不被他人獲得，讓整個加解密建立模式更安全且有彈性，並以橢圓曲線離散對數為基礎，結合背包理論來設計出一個具身分認證暨一次文件加密，利用橢圓曲線密碼系統所具有之金鑰長度較短與計算複雜度較低的特性設計，以達到機密性、完整性、不可否認性等安全需求，目的就是希望防止網路犯罪的發生，達到資訊保護之目標。

綜整本研究成果，貢獻如下：

一、利用雪崩效應，使加密資料獲得更高的完整性及機密性安全保護機制，即使片段資訊被截取，亦不會對明文資料產生威脅。

二、破解本方法須同時面對離散對數與資訊混淆兩種難題，使得本系統擁有更高的安全性。

表三　單一文件加密與一次文件加密法之分析比較表

比較項目	Elgamal橢圓曲線加密法	本研究一次文件加密法
密文完整性	未進行完整性檢查(無雜湊函數)	密文具有雪崩效應，若有一個位元遭到竄改可能會造成整體密文無法解密的狀況
密文機密性	植基於橢圓曲線離散對數	植基於橢圓曲線離散對數
不可否認性	具不可否認性	具不可否認性
密文通訊量	僅有橢圓曲線密碼系統密文擴充現象	除有橢圓曲線密碼系統密文擴充現象外，以多文件混合，有不同密文擴充現象
加密金鑰	收方公鑰	會議公鑰
明文拆解	拆解成一個點	拆解成為多個點
加密程序	以收方公鑰加密	除以雙方會議金鑰加密外，且密文點之間相互進行點加運算
解密程序	以橢圓曲線點減法解密	以橢圓曲線點減法解密
密文種類	機率式	機率式
加密次數	單一文件單一加密	多文件一次加密

資料來源：本研究

表四　Elgamal加密兩份文件與本研究方法運算時間複雜度比較

演算法	Elgamal加密兩份文件		本研究方法	
比較項目	時間複雜度	概估	時間複雜度	概估
金鑰產生	T_{EXP} x2	$\approx 480T_{MUL}$	$T_{ECC\,MUL}$	$\approx 29T_{MUL}$
加密運算	$(T_{EXP}+T_{INVS}+T_{ADD}+3T_{MUL})$ x2	$\approx 480T_{MUL}+(6.94+(1.686\ln(p))T_{DIV})$	$T_{ECC\,ADD}+2T_{ECC\,MUL}$	$\approx 58.12T_{MUL}$
解密驗證	$(3T_{EXP}+T_{MUL})$x2	$\approx 1440T_{MUL}+6T_{DIV}$	$T_{ECC\,ADD}+2T_{ECC\,MUL}$	$\approx 58.12T_{MUL}$
備註	$T_{ECC\,MUL}$：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 $\approx 240T_{MUL}$			
	$T_{ECC\,ADD}$：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(可忽略不計)			
	T_{MUL}：進行一次模式乘法運算所需時間 $\approx T_{DIV}$			

資料來源：本研究

三、資料傳輸雙方，只要經由AS認證後，未來即可利用離線之方式進行身分驗證作業，節省網路資源，增進作業效率。

收件：100年9月30日
第1次修正：100年11月14日
第2次修正：100年12月9日
接受：100年12月12日