網格計算之通訊安全設計

劉正義 [1]
[1]環球科技大學 資訊管理系
chengyi6129@gmail.com

黃慧鳳 [2,*] 詹昭文 [3] 程瑞芳 [4]
[2,3,4]國立台中科技大學 資訊工程系
{phoenix, ccwen, s17013003}@nutc.edu.tw

摘要

由於日益成長的計算服務，資源分享已愈來愈普及化，網格運算（Grid Computing）有可能成為未來的資訊基礎建設，因此，資通安全於網格運算之環境的應用是愈來愈受重視。又目前大部份之網格運算的研究，在執行群組成員的金鑰協定時都必須透過驗證中心，如此易增加驗證中心在計算方面與傳送時之負擔。基於二次剩餘，本研究提出一個具低運算量與傳輸量的群組通訊安全於網格計算(Grid Computing)應用服務之協定， 除了保護資料在傳輸的過程中免於遭受它人的存取或修改以外，所提之技術中，其傳送端與接收端的計算量，隨著群組通訊成員的數量增加並沒有明顯的差異，且執行群組的金鑰協定時不須要依賴驗證中心的參與而產生，將可降低驗證中心的負擔，更能提升資訊服務的品質，並提供安全與高效能的運算平台，可加速重要研究之進行。

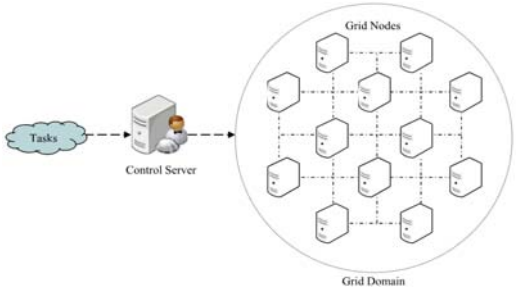
關鍵詞: 網格運算，二次剩餘 (Quadratic Residue) ，通訊安全

1. 前言

近數十年來網路科技的快速成長，人們逐漸習慣利用這些網際網路應用來處理一些日常工作，進一步享受便利的生活。舉凡郵寄信件、檔案分享、訊息交換、購物、醫療照護訊息等，使用者只需坐在電腦前即可完成。而網格計算 (Grid Computing)主要概念是將包括軟、硬體的分散伺服器資源、資料庫、儲存設備等，如(圖一)[14]，透過網路串聯以組成虛擬的超級電腦進而分享運算資源。以現今電腦科技進步，個人電腦的運算能力，早已遠遠超越當年計算主力之工作站電腦，然而一般大部份之桌上型電腦，其處理工作主要為「低運算需求」之文書處理、上網瀏覽、郵寄信件、與檔案分享等，因此可以想見，座落於校園內之「行政用電腦」，以及「電腦教室」之電腦，絕大多數時間，其 CPU 皆處於「閒置」的狀況。倘若能透過網路結集而充分運用其閒置資源，肯定能提供高效能之運算能力，且未來更能省下一筆可觀的軟硬體建置成本費用與預算。

而對企業而言，只需運用現有科技成本的一小部份，即能有效運用且聯結網路現有設備的資訊資源，隨而獲得更為強大的運算能力。網格運算還可讓分散於各地的虛擬組織，協調彼此的資源分享，同時滿足大量運算的需求。而集合分散的運算資源之外，Grid Computing 能夠經由網路管理組織內任何一個可使用的運算資源，進而降低伺服器的閒置時間。然而，為了防止非法入侵的存取或篡改以及保護系統資源的安全，所以散佈於網格計算(Grid Computing)中的節點通訊安全更顯得很重要。因此，近年來，陸續都有學者提出於網格運算之各節點間或群組通訊安全機制[1]，[2]，[4]-[7]，[17]，以確保資料在傳送的過程中免於遭受它人的存取或修改。但在這些方法中，有些學者所提之機制並不安全[3]，[8]-[11]，而在有些學者的方法中，其網格運算中很難動態刪除或加入新節點或群組成員[12]。

(圖一)



在2008年，Zhu-Wu[16]雖然提出網格運算之節點或群組成員可動態加入，但是各個成員須存有大量相關之資料量； 後來Yoon-Yoo[14]學者亦證明Zhu-Wu方法易遭受偽造攻擊並提出新的改進方式。因此，Yoon-Yoo [14]提出新的網格節點或群組成員之安全通訊機制，以改進Zhu-Wu等人的安全性與效能。 然而，我們發現在Yoon-Yoo與Zhu-Wu.等人之方法中，其傳送者與信賴驗證中心的運算量會隨著網格計算中之節點或群組成員之增加而明顯的增加。 除此之外，每次執行群組的金鑰協定都必須依賴驗證中心的參與而產生，因此，很容意增加驗證中心在計算方面與傳送時之負擔，如此一來，對日益成長的資訊服務，會使得效能易受影響而導致瓶頸。因此，為了克服因計算量及傳輸量所導致的瓶頸，本研究設計一個高效能且具安全的網格計算之通訊環境，除了保護資料在傳輸的過程中免於遭受它人的存取或修改外，所提之技術中，其傳送端與接收端的計算量，隨著群組成員

的數量增加並沒有明顯的差異，且執行群組的金鑰協定不須要依賴驗證中心的參與產生，將可減低驗證中心的負擔，因此，更能提升資訊服務的品質，並提供更好效能且安全的資源共享與合作分工的服務環境。

2. 相關研究的探討

由於目前網格計算(Grid Computing)的群組通訊安全協定中以 Yoon-Yoo [14]之方法較為安全，所以在提出本設計之前，以下先簡單介紹 Yoon-Yoo 之方法，其方法分為起始階段、訊息的廣播階段、與解密階段。

Yoon-Yoo 之方法主要是植基 RSA 密碼系統，在起始階段，信賴驗證中心(central authority system 簡稱 CAS)負責產生各個結點或使用者之參數與私鑰，CAS 先選擇兩個很大的質數 p 與 q 且令 $N = p \times q$，接著選擇一對數(e,d), 使得 $e \times d = 1 \mod \phi(N)$ [16],並選擇一加密函數 $f(x) = x^d \bmod \phi(N)$。如此一來, e及 N 即為CAS的公開金鑰,而 d 為CAS秘密金鑰; 同時地, p 與 q 亦是秘密參數，然後 CAS 選擇一個公開之單向雜湊函數 $h(\,)$。

假設目前參與群組有 n 個人為$U_1, U_2, ..., U_n$，首先信任驗證中心 CAS 選擇私密參數K_0，r_c 與t_i，而 $i = 1, 2, ..., n$。接著由信任驗證中心 CAS 產生 n 把私密金鑰，$K_i = K_0^{t_i}$，並透過安全通道配送K_i給使用者 U_i，而 $i = 1, 2, ..., n$。另一方面，並公開 $P_i = t_i^{-1} r_c \bmod \phi(N)$，$i = 1, 2, ..., n$。接著介紹 Yoon-Yoo 方法之訊息廣播階段與解密階段:

訊息廣播階段:

假設目前傳送者U_1欲將重要訊息傳送至網格計算(Grid Computing)中之某一群組 $G = \{U_i\}_{i=1}^{a}, a \le n$, 以下為所執行之步驟:

1. U_1 先將目前之群組 $G = \{U_i\}_{i=1}^{a}$ 傳於 CAS。
2. CAS 選擇一數 Z，並計算 $Z_i = E_{K_i}(Z)$ 而 $i = 1, 2, ..., a$，$B = t_1 t_2 ... t_a \bmod N$，$f(B) = B^d \bmod \phi(N)$，以及 $Y = h(z, f(B))$。然後，CAS 將 $Z_i = E_{K_i}(Z)$ 而 $i = 1, 2, ..., a$，與 $f(B), Y$ 廣播於群組 $G = \{U_i\}_{i=1}^{a}$ 中。
3. 當收到訊息後，U_1 利用其私鑰 K_1 解開 $Z = E_{K_1}(Z_1)$，並驗證 $Y = h(z, f(B))$ 是否成立，若成立則接受其為合法之 CAS; 否則停止這次協定。
4. 接著U_1 計算這次群組通訊之共同私鑰 $sk = K_1^{Z(f(B)P_1)e} \bmod N = K_1^{Z(Bt_1^{-1}r_c)ed} \bmod N = K_0^{ZBr_c} \bmod N$，並利用 sk 將重要資料 M 加密為密文 $C = E_{sk}(M)$，並計算 $V = h(sk, M)$，然後，將 C 與 V 廣播於群組 $G = \{U_i\}_{i=2}^{a}$ 中。

解密階段:

當這次合法群組 $G = \{U_i\}_{i=2}^{a}$ 收到 U_1 之訊息 C 與 V 後，則解密步驟說明於下:

1. 首先群組成員 $G = \{U_i\}_{i=2}^{a}$ 由其私鑰 K_i 解開 $Z = D_{K_i}(Z_i)$，而 $i = 2, 3, ...a$，並驗證 $Y = h(z, f(B))$ 是否成立，若成立則接受其為合法之 CAS; 否則停止這次解密階段。
2. 接著 U_i，$i = 2, 3, ...a$，計算這次群組通訊之共同私鑰 $sk = K_i^{Z(f(B)P_i)e} \bmod N = K_i^{Z(Bt_i^{-1}r_c)ed} \bmod N = K_0^{ZBr_c} \bmod N$，並利用 sk 將重要資料解密為 $M = E_{sk}(C)$，並驗證 $V = h(sk, M)$ 是否成立，若成立，則接受 M 為其合法者U_1所廣播之資訊; 否則停止這次解密階段。

而網格計算 (Grid Computing) 中之群組 $G = \{U_i\}_{i=1}^{a}, a \le n$, 可利用此共同私鑰 sk 來傳送重要資源與協同分工運算後之結果。

由上述 Yoon-Yoo [14] 兩人所提方法可知, 當群組 $G = \{U_i\}_{i=1}^{a}, a \le n$ 共有 a 位時, 對傳送者U_1 而言,須要執行一個指數運算、二次對稱式加密與二個雜湊函數運算，而對每位接收者而言,也須要執行一個指數運算、二次對稱式加密與二個雜湊函數運算，另一方面，信任驗證中心 CAS 仍須執行一個指數運算、a 次對稱式加密與一個雜湊函數運算。雖然 Yoon-Yoo 兩人所提方法比之前其它學者所提方法更具安全性，但是，其 CAS 的運算量會隨著網格計算中之群組成員之數量增加而明顯的增加。 除此之外，每次執行群組的金鑰協定都必須透過驗證中心 CAS 的參與而產生,很容意增加 CAS 在計算方面與傳送時之負擔,對日益成長的資訊服務，會使效能易受影響而導致瓶頸。

3. 研究方法

由於日益成長的計算服務，資源分享已愈來愈普及化，網格運算（Grid Computing）有可能成為下一代的資訊基礎建設，又目前大部份之研究，在執行群組的金鑰協定都必須透過驗證中心 CAS，因此，本研究提出一個具低運算量與傳輸量的群組通訊安全於網格計算(Grid Computing)應用服務之協定。

為了降低各方面的傳輸量與計算量，以及提供更有效率之資訊服務與共享的平台，因此，本研究採二次剩餘(Quadratic Residue，簡稱 QR)的密碼系統方式，因為 QR 的加解密方式很明瞭簡單,其安全是建立在因數分解之難題上[15], 與 RSA 金鑰系統的安全性是一樣的[13]。在提出本研究之前，以下我們先簡單描述二次剩餘(QR)密碼系統之性質。$n \in N$，若一整數 a 且 $\gcd(a, n) = 1$，滿足 $x^2 = a (\bmod n)$ 有解，則稱 a 為模 n 之二次剩餘((Quadratic Residue a modulo n)QR)，否則稱 a 為模 n 之非二次剩餘(Quadratic Nonresidue of n)。而滿足 $x^2 = a (\bmod n)$ 之解 x，則稱為模 n 之二次剩餘的根(modular square roots of quadratic residue a modulo n)。二次剩餘(QR)密碼系統之性質有下列之性質[15], [16]。

(1) 當 $p = 3 (\bmod 4)$ 且 a 為模 p 之二次剩餘 ($x^2 = a (\bmod p)$,則有兩個模 p 之二次剩餘的根為 $r_{1,2} = \pm a^{\frac{p+1}{4}} (\bmod p)$，使得 $(r_{1,2})^2 = a (\bmod p)$。

(2) $n = p \times q$ 且 $\gcd(a,n) = 1$，a 為模 n 之二次剩餘（$x^2 = a(\bmod n)$），其中 p 和 q 為兩個質數，當 $p = q = 3(\bmod 4)$，則有四個模 n 之二次剩餘的根為 $r_{1,2,3,4}$，使得 $(r_{1,2,3,4})^2 = a(\bmod n)$，而 $r_{1,2} = \alpha \cdot q \cdot q^* \pm \beta \cdot p \cdot p^* (\bmod n)$，$r_{3,4} = -\alpha \cdot q \cdot q^* \pm \beta \cdot p \cdot p^* (\bmod n)$，其中 $\alpha = a^{\frac{p+1}{4}} (\bmod p)$，$\beta = a^{\frac{q+1}{4}} (\bmod q)$，$p^* = p^{-1} (\bmod q)$，且 $q^* = q^{-1} (\bmod p)$。

上述之性質(2)，若 p 和 q 兩個質數已知時，則 $x^2 = a(\bmod n)$ 之四個根均可在多項式時間內求出[15]，但 p 和 q 兩個質數未知時，要求出 $x^2 = a(\bmod n)$ 之四個根，則須先將 n 因數分解，在此情形下與 RSA 金鑰系統的安全性是一樣，都是建立在因數分解之難題上[15]。底下將針對所要設計的網格計算之安全通訊協定，分別說明其過程步驟。

在本協定剛開始階段，假設目前參與群組有 n 個人為 $U_1, U_2, ..., U_n$，首先信任驗證中心(central authority system 簡稱 CAS)挑選一個對稱式加解密系統如 AES 或 DES [15]，而 $E_K(.)$ 和 $D_K(.)$ 分別表示當秘密金匙 K 時之加密和解密之意義，然後，選擇一個公開之單向雜湊函數 $h(\,)$，CAS 接著計算並公開 $h(ID_i, x)$，而 $i = 1,2,...n$，其中 ID_i 為使用者 U_i 的辨識碼，x 為 CAS 之秘密私鑰。另一方面，每位 U_i 先選擇兩個很大的質數 p_i 與 q_i 且令 $N_i = p_i \times q_i$，其中 $p_i = q_i = 3(\bmod 4)$ 且 p_i, q_i 均為兩個很大的質數，接著選擇一對數 (e_i, d_i)，使得 $e_i \times d_i = 1 \bmod \phi(N_i)$，而 (N_i, e_i) 即為每位 U_i 之公鑰，d_i 為 U_i 之密鑰且 p_i 與 q_i 亦是 U_i 的秘密參數。接著介紹本研究中之訊息廣播階段與解密階段的步驟於下:

訊息廣播階段:

假設目前傳送端 U_1 欲將重要訊息 M 傳送至網格計算(Grid Computing)中之一某群組成員 $G = \{U_i\}_{i=1}^{a}, a \le n$，以下為所執行之步驟:

1. U_1 先選擇這次群組通訊之共同私鑰 sk。
2. U_1 計算 $C = E_{sk}(M)$，$S = (h(M))^{d_1} \bmod N_1$，與 $Z_i = (h(ID_i, x) \| sk)^2 \bmod N_i$，而 $i = 2,3,...,a$。
3. U_1 將 C，S，與 $Z_i = (h(ID_i, x) \| sk)^2 \bmod N_i$，$i = 2,3,...,a$，廣播於群組 $G = \{U_i\}_{i=2}^{a}$ 中。

解密階段:

當這次合法群組 $G = \{U_i\}_{i=2}^{a}$ 收到 U_1 之訊息 C，S，與 $Z_i = (h(ID_i, x) \| sk)^2 \bmod N_i$，$i = 2,3,...,a$，則解密步驟說明於下:

1. 首先 U_i，$i = 2,3,...a$，計算這次群組通訊之共同私鑰 sk，U_i 計算並取出滿足 $Z_i = (h(ID_i, x) \| sk)^2 \bmod N_i$ 之四個模 N_i 之二次剩餘的根為 $r_{1,2,3,4}$ 使得 $(r_{1,2,3,4})^2 = (h(ID_i, x) \| sk)^2 \bmod N_i$，並從四個根 $r_{1,2,3,4}$ 中挑出附有 $h(ID_i, x)$ 為開頭之私鑰出 $sk = r_{1,2,3,4} \oplus h(ID_i)$。
2. 接收者 U_i，$i = 2,3,...a$，利用 sk 將重要資料解密為 $M = D_{sk}(C)$，並驗證 $h(M) = S^{e_1} \bmod N_1$ 是否成立，若成立，則接受 M 為其合法傳送端 U_1 所廣播之資訊; 否則停止這次解密階段。

而網格計算(Grid Computing)服務中之群組 $G = \{U_i\}_{i=1}^{a}, a \le n$，可利用此共同私鑰 sk 來傳送重要資源與協同分工運算後之結果。由上述之訊息廣播階段與解密階段，可明顯看出協定機制中，目前授權合法之群組 $G = \{U_i\}_{i=1}^{a}, a \le n$ 通訊成員之數量 a，可以動態的加入或刪除成員，並不影響其它成員原本之參數與系統之通訊安全。

4 方法之討論與分析

由上一節所提之研究方法可知，系統協定的程序很簡單，可提高實際的應用價值，並提供安全與便利之網格運算環境的服務。基於安全考量，為了避免重送攻擊(replay attack)與每次網格運算環境的資訊安全，以上之群組通訊之私鑰 sk，都不可以重覆使用，以免導致資訊被不同之群組所竊取或修改。本研究採二次剩餘(簡稱 QR)的密碼系統，當使用者 U_1 先選擇這次群組通訊之共同私 sk，並計算 $C = E_{sk}(M)$，$S = (h(M))^{d_1} \bmod N_1$，與 $Z_i = (h(ID_i, x) \| sk)^2 \bmod N_i$ 並將其廣播於 $G = \{U_i\}_{i=2}^{a}$ 時，所以當群組 $G = \{U_i\}_{i=1}^{a}, a \le n$ 的私鑰 sk 被求出時，則網格運算環境的訊息隱私就被得知，這時就不具有使用者安全了，然而，由已知 $Z_i = (h(ID_i, x) \| sk)^2 \bmod N_i$ 要得到 sk 是很困難的，其安全是建立在因數分解 $N_i = p_i \times q_i$ 之難題上，與 RSA 金鑰系統[13]的安全特性是一樣，因此，本協定可提供網格運算中群組通訊之安全。除此之外還具備下列的安全性質:

1. 由於使用者可因目前之需求而決定網格運算之成員，而每次各個群組通訊之私鑰 sk 也不同，因此，機制中群組之成員可以動態的加入或刪除，不須更改原本成員之參數。
2. 信任中心 CAS 只須公開網格環境中，目前合法使用者之相關參數(包含新加入之新成員與被刪除之成員)，不須參與各個群組之私鑰產生。所以 CAS 無法得知各個群組之私鑰，因此，更可提高網格環境中群組通訊之安全。
3. 具 forward security 和 backward security 的安全性: 即使使用者的某回合私鑰 sk 被得知，因每次的 sk 都由傳送端任選且不重覆，又被保護於 $Z_i = (h(ID_i, x) \| sk)^2 \bmod N_i$，因此，惡意者很難得知使用者過去與未來之回合私鑰。所以本研究擁有 forward security 和 backward security 安全性。

另一方面，我們所提出之方法中,傳輸量方面只須一次(one round) 的傳輸次數, 比先前 Yoon-Yoo 學者的三次回合較少[14]。在計算量上，本研究將採二次剩餘(Quadratic Residue，簡稱 QR)的密碼系統方式，且系統在開始階段，信任中心 CAS 已計算並公開 $h(ID_i, x)$ 之值

且 $i=1,2,3,...n$，所以當群組 $G=\{U_i\}_{i=1}^{a}, a \le n$ 有 a 個使用者時，在傳送端的計算量只須一次高次的指數運算、一次對稱式加密運算、一個雜湊函數運算、與 a 次的乘法運算；而對每位接收者而言，只須一次二次剩餘根的計算、一次高次的指數運算、一次對稱式解密運算與一個雜湊函數運算；而信任中心 CAS 無須任何之計算，因此，本研究之方法其運算速度方面比 Yoon-Yoo 學者之方法更加快速，相對也減少很多傳輸量，所以可提高整體之計算速度，更能提供高效能的資訊服務品質且更適合於網格運算之環境。

有關計算效能方面我們與目前較安全的 Yoon-Yoo 之方法作比較，如下表(一)之敘述:其中 E 表示模之指數運算，h 為雜湊函數運算，Mul 為模之乘法運算，r 為二次剩餘根的計算，而 S 為對稱式加解密運算，又 XOR 計算在此省略不計。由表(一)可得知無論在傳送端 U_1、接收者(群組成員)、以及信任中心 CAS 方面，我們所提之方法都比 Yoon-Yoo 方法快速。雖然 Yoon-Yoo 兩人所提方法比之前其它學者所提方法更具安全性，但是，Yoon-Yoo 方法中 CAS 的運算量都須要 a 次對稱式加解密，其運算量會隨著網格計算中之群組成員之增加而明顯的增加。除此之外，Yoon-Yoo 方法每次執行群組的金鑰協定都必須透過驗證中心 CAS 的參與而產生，因此，很容意增加 CAS 在計算方面與傳送時之負擔，如此一來，對日益成長的資訊服務，會使得效能易受影響而導致瓶頸。

在我們所提之方法中，信任中心 CAS 無須任何之計算，雖然傳送者仍須 a 次的乘法運算，但是 a 次的乘法運算，隨著成員數量的增加比較沒有明顯的差異了。因此，由上述的安全分析與效能分析，本研究可以大幅改善傳送者和信任中心 CAS 的計算量，更適用於網格運算的環境，除此之外，CAS 亦無法得知群組 $G=\{U_i\}_{i=1}^{a}, a \le n$ 之回合私鑰 sk。透過本協定使用者可在任何時間、任何地點，藉由此安全通訊來達成存取與分享資源之理念，並提供安全與高效能的運算平台，進而加速各項重要研究之進行。

表(一)

成員有 a 位時	群組之成員	傳送者 U_1	信任中心 CAS
Yoon-Yoo 方法	$1E+2S+2h$	$1E+2S+2h$	$1E+aS+1h$
所提之本方法	$1E+1S+1h+1r$	$1E+1S+1h+aMul$	0

5 結論

基於二次剩餘(簡稱 QR)的密碼系統，本研究提出一個具低運算量與傳輸量的通訊安全於網格計算(Grid Computing)應用服務之協定， 除了保護資料在傳輸的過程中免於遭受它人的存取或修改以外，所提之技術其傳送端與接收端的計算量，隨著群組成員的數量增加並沒有明顯的差異。而機制中群組之成員可以動態的加入或刪除，不須更改原本成員之參數，且執行群組的金鑰協定時，不須要依賴驗證中 CAS 的參與而產生，將可減低 CAS 的負擔。

參考文獻

[1] W. Chung, R. Chang, "A new mechanism for resource monitoring in grid computing", *Future Generation Computer Systems*, Vol. 25, No. 1, 2009, pp. 1–7.

[2] M. Smith, M. Schmidt, N. Fallenbeck, T. Dornemann, C. Schridde, B. Freisleben, "Secure on-demand grid computing", *Future Generation Computer Systems* Vol.25, No. 3, 2009, pp. 315–325.

[3] J. Masque, A. Peinado, Cryptanalysis of improved Liaw's broadcasting cryptosystem, *Journal of Information Science and Engineering*, Vol. 22, 2006, pp. 391–399.

[4] X. Zou, Y. Dai, X. Ran, "Dual-level key management for secure grid communication in dynamic and hierarchical groups", *Future Generation Computer Systems*, Vol. 23, No. 6, 2007, 7pp. 76–786.

[5] D. Zou, W. Zheng, J. Long, H. Jin, X. Chen, "Constructing trusted virtual execution environment in P2P grids", *Future Generation Computer Systems*, Vol. 26, No.5, 2010, pp. 769–775.

[6] F. Martinelli, P. Mori, "On usage control for GRID systems", *Future Generation Computer Systems*, Vol. 26, No. 7, 2010, pp. 1032–1042.

[7] J. Perez, J. Bernabe, J. Calero, F. Clemente, G. Perez, A. Skarmeta, "Semantic-based authorization architecture for grid", *Future Generation Computer Systems*, Vol. 27, No.1, 2011, pp. 40–55.

[8] C. Chang, T. Wu, "Broadcasting cryptosystem in computer networks using interpolating polynomials", *Computer Systems Science and Engineering*, Vol. 6, No. 3, 1991, pp.85–188.

[9] G. Chiou, W. Chen, "Secure broadcasting using the secure lock", *IEEE Transactions on Software Engineering*, Vol. 15, No.8, 1989, pp. 929–934.

[10] H. Liaw, "Broadcasting cryptosystem in computer networks", *Computers & Mathematics with Applications*, Vol. 37, 1999, pp. 85–87.

[11] Y. Tseng, J. Jan, "Cryptanalysis of Liaw's broadcasting cryptosystem", *Computers & Mathematics with Applications*, Vol. 41, 2001, pp.1575–1578.

[12] M. Ramkumar, "Broadcast authentication with preferred verifiers", *International Journal of Network Security*, Vol. 4, No.2, 2007, pp. 166–178.

[13] R.L. Rivest, A. Shamir and L. M. Adleman, "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems", *Communications of the ACM*, Vol. 21, 1978, pp. 120-126.

[14] E. J. Yoon, K. Y. Yoo, "A secure broadcasting cryptosystem and its application to grid computing", *Future Generation Computer System*, Vol. 27, 2011, pp. 620-626

[15] S. S. Yan, "Number Theory for Computing", Berlin, Germany, 2000.

[16] W. Zhu, C. Wu, "Security of the redefined Liaw's broadcasting cryptosystem", *Computers & Mathematics with Applications*, Vol.56, 2008, pp. 1665–1667.

[17] K. Huang, D. Zhang, "DHT-based lightweight broadcast algorithms in large scale computing infrastructures", *Future Generation Computer Systems*, Vol.26, No. 3, 2010, pp. 291–303.。