分散式數位版權管理網路系統

洪丞甫* 謝哲人* 李忠憲* 劉奕賢**
*國立成功大學電腦與通信工程研究所 **遠東技術學院資管系
q3694439@mail.ncku.edu.tw sai888@cc.fec.edu.tw jsli@mail.ncku.edu.tw
stmdanny@ms4.hinet.net

摘要

隨著現今網路技術的發達，資料傳遞的便利卻衍生出許多關於數位版權的問題，我們提出一個新的數位版權管理網路系統，可以有效提昇資料於網路中的生存時間，降低內容供應者的網路負擔，而且利用 P2P 網路的特性來避免單點失敗(single point of failure)。我們也採用公開金匙密碼系統產生出專屬於每位用戶且唯一的數位指紋，並且利用資料隱藏的技術將數位指紋內嵌於多媒體檔案中。此機制能夠追溯出非法散佈多媒體檔案的用戶，並且可以嚇阻企圖進行竄改或是偽造指紋的惡意攻擊者，所以可以有效地進行數位內容版權管理並保障下載檔案的用戶權益。

關鍵字：數位版權、公開金匙密碼系統、數位指紋、資料隱藏

1. 前言

隨著網路通訊快速發展，數位內容產業提供許多商業機會；相反的，由於網路資料可容易隨意拷貝，造成非法複製、發行與剽竊問題[3]。

在網際網路上傳遞多媒體內容相當方便及普遍，而一般的加密機制無法防制多媒體內容於下載後經由用戶非法授權的拷貝及散佈，因此才出現了數位浮水印的技術。數位浮水印是將一些識別資訊嵌入在多媒體內容中，而一般用戶無法將浮水印移除，並且浮水印可應用在數位版權管理(DRM)或辨識接收者等方面。數位浮水印可藉由嵌入某些資訊來識別多媒體內容提供者或版權擁有者；若是應用於識別接收者部份，則我們稱之為數位指紋[10]，藉由數位指紋機制，我們可由未經授權而拷貝的多媒體內容中之數位指紋去追溯出原先的檔案接收者。

本文中我們提出一個可以確保資料傳遞的完整性並且提供數位版權管理於多媒體內容的理論。在我們所建置的網路環境下，內容提供者先將多媒體內容進行來源資料編碼，利用此編碼方式可以大幅提昇檔案於網路中的生存時間，並且可以避免因為單點失敗(single point of failure)而造成檔案無法完整下載，同時也降低了內容提供者的網路負擔。編碼過的多媒體內容將於網路供應者端的媒體分享伺服器以 P2P 網路型態來進行資料分享，當收集到足夠編碼過的多媒體內容之後，即可解回原檔案內容。用戶端將透過所獲得的編碼內容並結合 RSA 公開金匙加密機制來產生出專屬於自身的數位指紋，並且送交該數位指紋來驗證指紋的正確性。所有的用戶端在完整獲得多媒體內容之前，內容供應者將會利用資料隱藏的技術，而把各個用戶所專屬的數位指紋嵌入於多媒體內容中，並且傳遞含有各自數位指紋的多媒體內容給各個用戶。所以只要有非法用戶散播多媒體內容在網路上時，內容供應者就能夠透過隱藏於多媒體檔案中的數位指紋來找出非法散佈檔案的用戶。

2. 研究背景

2.1 網路群播與相關安全機制

網路群播[13]是一種傳遞資料的網路技術，它可從單一來源者透過網路有效傳遞資料給許多接收者。透過網路安全加密機制，從內容提供者到接收者都可確保合法用戶才可下載資料[7]，然而接收者下載合法資料後，非法複製資料的問題卻無法解決，因此有研究提出在傳遞的內容加入浮水印或數位指紋方式來判斷非法複製的用戶，以維護數位版權。H. hua Chu 等人[4]提出傳送者在網路多媒體內容廣播環境，分別獨立傳送每一個接受者不同的資料流，這種方式優點是不需要額外網路環境建置但卻需要較多的網路頻寬去傳送多媒體內容。另一種方式是由 I. Brown 等人[5]建議在網路傳遞過程修改每一接受者的多媒體內容浮水印，這種方式缺點是需要額外網路環境建置支援，因此不能適用在許多網路環境。

2.2 拉格朗日多項式插入法

拉格朗日(Lagrange)多項式插入法可用於數值資料於幾何上的分析，以幾何觀點來看一個最高次為 t-1 次的多項式 h(x)，該多項式即為 X-Y 平面上的某條曲線，只要獲得$(x_0, h(x_0))$、$(x_1, h(x_1))$、…、$(x_{t-2}, h(x_{t-2}))$、$(x_{t-1}, h(x_{t-1}))$等 t 個以上的相異點，即可利用拉格朗日多項式插入法解回原多項式，如(1)式所示。

$$h(x)=\sum_{i=0}^{t-1}h(x_i)\prod_{j=1,j\neq i}^{t}\frac{x-x_j}{x_i-x_j} \quad \ldots\ldots\ldots (1)$$

where i = 0, 1, 2, …, t-1

拉格朗日多項式插入法也被應用於秘密分享[1]，1979 年 Shamir 根據拉格朗日多項式插入法提出(t, n)門檻方法，該方法可以動態調整門檻值 t 而有效達到系統的安全性與靈活度，主要是將一份主密鑰分成 n 份相異子密鑰，只有當子密鑰的數量大於等於門檻值時，才能夠推導出主密鑰，利用此方法來達到動態性的秘密分享。

2.3 RSA 公開金匙密碼系統

RSA 公開金匙加密演算法[12]由麻省理工學院的學者 Rivest，Shamir，Adleman 於 1977 年所提出。RSA 為非對稱式加密演算法，加密與解密並非使用同一把密鑰，而是利用公鑰進行加密動作，且利用私鑰進行解密動作。RSA 演算法流程則是先產生兩個大質數 p、q，並且使得 N = p×q 以及 Φ(N) = (p-1)(q-1)，之後再選出一 e 值滿足兩點要求：(1) 1<e<Φ(N)，(2) e 須與 Φ(N)互質。之後在選擇一 d 值滿足 $d = e^{-1} \bmod \Phi(N)$，所以公鑰即為(e, N)，而私鑰則是(d, N)，加密與解密的方法如(2)、(3)兩式所示。在此 M 為明文，而 C 則為加密過的內容。

$$\text{加密}：C=M^e \bmod N \quad \ldots\ldots\ldots\ldots (2)$$
$$\text{解密}：M=C^d \bmod N \quad \ldots\ldots\ldots\ldots (3)$$

RSA 加密演算法也應用於數位簽章方面，傳送者送出資料時附上自身的數位簽章可便於接收者驗證傳送者的身份。RSA 於數位簽章的應用恰好與加密解密相反，因為數位簽章是利用私鑰進行簽章，當接收者收到資料時則可利用公鑰來驗證該簽章是否屬於傳送者，簽章與驗證的動作如(4)、(5)兩式所示。在此 ID 為傳送者的身份，S 則為傳送者的數位簽章。

$$\text{簽章}：S=ID^d \bmod N \quad \ldots\ldots\ldots\ldots (4)$$
$$\text{驗證}：ID=S^e \bmod N \quad \ldots\ldots\ldots\ldots (5)$$

2.4 資料隱藏

現今有許多資料隱藏的技術，包括數位浮水印[8]，DCT[15]，Wavelet transform[6]…等，顧名思義則是要將重要的資訊隱藏在其他一般的資訊上，在此我們所要提到的重點則是以多媒體檔案為主的資料隱藏技術，所以我們將重要資訊隱藏於多媒體檔案時，主要是透過多媒體之影像或聲音為媒介，並且加入該重要資訊於檔案後則須不被人體知覺所感受到多媒體內容有所改變。

3. 研究方法

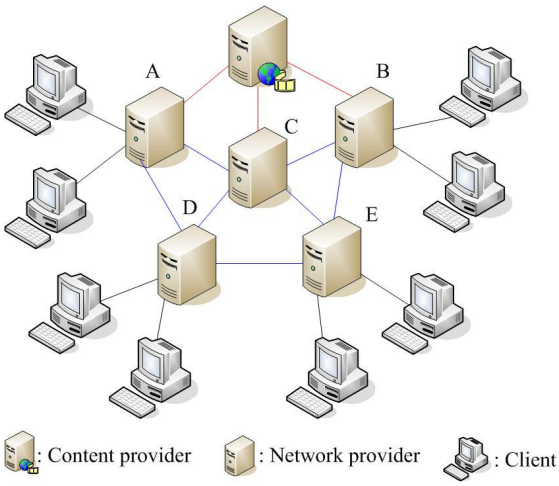

圖 1 系統架構圖

整體系統環境主要由三大角色所組成，分別為檔案發行者(content provider)、網路供應者(network provider)、用戶端(client)，如圖 1 所示。檔案發行者可視作某些電影或是多媒體的發行公司，而網路供應者則為提供網路設備及環境的供應商，如 ISP 業者。所以於此環境下，檔案發行者將與網路供應者進行合作並佈置媒體分享伺服器於網路供應者端，並且所有的媒體分享伺服器則自成一個 P2P 網路。所以檔案發行者可透過網路供應者端釋出檔案，而用戶端則透過網路供應者來下載多媒體檔案。另外，用戶端皆各自擁有自己的 User ID(以下皆簡寫為 UID)，而 UID 主要則是用來做為識別該用戶身份的依據，例如用戶的來源 IP 即可當作其 UID。在此環境下，在此我們提出一個新的數位指紋理論，可以針對每個用戶來產生專屬且唯一的數位指紋，並且利用資料隱藏的技術將數位指紋嵌入於多媒體檔案內。檔案發行者則可從未經授權散佈之檔案中取出該份檔案的數位指紋，由此查出是哪位用戶在未經授權之情況下將檔案散佈於網路之中並且採取相對應的法律行動。

3.1 來源資料編碼

檔案發行者利用拉格朗日多項式 h(x)來對多媒體檔案進行「來源資料編碼」。所謂「來源資料編碼」意指先將該檔案切割成 b_0、b_1、b_2、b_3、…、b_{n-1} 等 n 份資料區塊，而任一個資料區塊 b_i 可再切割為 $a_{i,0}$、$a_{i,1}$、$a_{i,2}$、…、$a_{i,t-2}$、$a_{i,t-1}$ 等 t 部份。之後選定拉格朗日多項式 $h_i(x)$之最高項次為 t-1，而 $h_i(x)$的各項係數即為 b_i 所切割出的 $a_{i,0}$、$a_{i,1}$、$a_{i,2}$、…、$a_{i,t-2}$、$a_{i,t-1}$ 等 t 部份，如(6)式所示，在此 i = 0, 1, 2, …, n-1。

$$h_i(x) = a_{i,t-1}x^{t-1} + a_{i,t-2}x^{t-2} + \ldots + a_{i,1}x + a_{i,0} \quad \ldots (6)$$

所以每一個資料區塊 b_i 皆有一個所對應的

$h_i(x)$。檔案發行者進行來源資料編碼之後，將 $h_i(x)$ 直接傳送到置放於網路供應者端的媒體分享伺服器 x_0、x_1、x_2、…、x_{k-1} 等 k 部主機，所以此 k 部主機將會散佈出 $h_i(x_0)$、$h_i(x_1)$、$h_i(x_2)$、…、$h_i(x_{k-1})$等 k 份數值(k > t)。所以當用戶對上層的媒體分享伺服器發出下載請求時，該媒體分享伺服器只要於所在的 P2P 網路內收到任意 t 組相異的$(x_i, h(x_i))$資料之後，則並不採用拉格朗日多項式插入法來解回原方程式的係數，而是利用高斯消去法解回原方程式並取得各項係數，如(7)式所示。

$$\begin{bmatrix} x_1^{t-1} & x_1^{t-2} & \cdots & x_1 & 1 \\ x_2^{t-1} & x_2^{t-2} & \cdots & x_2 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{t-1}^{t-1} & x_{t-1}^{t-2} & \cdots & x_{t-1} & 1 \\ x_t^{t-1} & x_t^{t-2} & \cdots & x_t & 1 \end{bmatrix} \cdot \begin{bmatrix} a_{i,t-1} \\ a_{i,t-2} \\ \vdots \\ a_{i,1} \\ a_{i,0} \end{bmatrix} = \begin{bmatrix} h(x_1) \\ h(x_2) \\ \vdots \\ h(x_{t-1}) \\ h(x_t) \end{bmatrix} \quad \dots (7)$$

媒體分享伺服器解出方程式的各項係數之後便可重組回原資料區塊 b_i，所以其他資料區塊也是可以利用相同的方法重組回來。另外，媒體分享伺服器在收到下載請求時將會獲得用戶的 UID(用戶的來源 IP)，所以將該用戶的 UID 代入所解出的方程式後即可得到一份新的 h_i(UID)值。媒體分享伺服器則可將此份新資料(UID, h_i(UID))分享於 P2P 網路中，用以增加 P2P 網路中 h_i(UID)值的數量。因為只要有 t 份以上的資料存在於此 P2P 網路中，其他的媒體分享伺服器就能夠持續下載檔案並有足夠的資料數量可以解回該方程式，所以此機制間接地提昇了檔案於 P2P 網路中的生存時間，並且因為來源資料編碼的緣故，可以避免某些分享檔案的主機下線或是其他不明原因而造成部份檔案區塊的遺失，使得整體檔案無法被完整下載，也就是所謂單點失敗(single point of failure)。當然透過來源資料編碼的機制，檔案發行者傳送出 h(x)之後也不須時時保持上線，可以有效地降低檔案發行者端的網路負載。

3.2 數位指紋的產生

我們在此架構下提出一套新的數位指紋理論，藉由我們所提出的理論可以追查出未經授權而散佈多媒體檔案的用戶並且對該用戶採取相對應的法律行動，也能夠有效地嚇阻進行非法散佈檔案的用戶。首先媒體分享伺服器收集資料時，將會收到不同資料區塊 b_i 所對應之 $h_i(UID_j)$的數值，直到媒體分享伺服器收集到 m 組不同的$(UID_j, h_i(UID_j))$資料時，對此 m 份不同的 $h_i(UID_j)$數值同時作 XOR 的動作後即可得到 seed。此時媒體分享伺服器中的對稱式金鑰加密系統將會產生一把秘密金鑰(secret-key) (E, N_e)，並且將此秘密金鑰同時傳送給檔案發行者端以及用戶端，並且也將 seed 傳送給用戶端，所以用戶接收到 seed 之後，將其引入 RSA 公開金匙密碼系統的流程來產生出我們所定義的公鑰(checking-key) (C, N)與私鑰(producing-key) (P, N)。

數位指紋的產生則在用戶端進行，因為需要用到用戶的私鑰，並且私鑰一旦洩漏出去，則數位指紋將可能會受到偽造等破壞行為，所以選擇在用戶本機端進行可以避免私鑰洩漏的安全顧慮。根據上述所提之 RSA 公開金匙密碼系統於數位簽章方面的應用，我們利用 UID 與私鑰(P, N)來產生數位指紋，如(8)式所示，在此 UID 是指用戶的 User ID，(P, N)則是指用戶的私鑰，而 FP 就是用戶的數位指紋。用戶產生出數位指紋之後，為了避免傳送過程中數位指紋遭到竊聽，所以用戶再利用先前收到的秘密金鑰對數位指紋加密，如(9)式所示。

$$FP = UID^P \bmod N \quad \dots\dots\dots (8)$$
$$FP^* = FP^E \bmod N_e \quad \dots\dots\dots (9)$$

之後用戶將加密過的數位指紋 FP^*與公鑰(C, N)傳送給網路供應者端的媒體分享伺服器進行驗證動作，首先媒體分享伺服器需將加密過的數位指紋解密，如(10)式所示。接著再根據 RSA 公開金匙密碼系統於數位簽章方面的應用，媒體分享伺服器利用數位指紋 FP 與公鑰(C, N)可得到運算結果 UID，如(11)式所示。因為媒體分享伺服器早在接收用戶的下載請求之時就已經獲得過用戶的 UID，所以可以比對運算後所得到的 UID 是否與先前所獲得的 UID 是否符合。

$$FP^E = (FP^*)^E \bmod N_e \quad \dots\dots\dots (10)$$
$$UID = FP^C \bmod N \quad \dots\dots\dots (11)$$

如果驗證結果符合，媒體分享伺服器則會將加密過的數位指紋 FP^*與公鑰(C, N)傳送給檔案發行者端，而檔案發行者則會先將 FP^*解密而得出數位指紋之後，再把用戶的數位指紋 FP 與(C, N)存入資料庫；但若驗證結果不符，也就是該用戶的數位指紋無法反推出其 UID，則此結果指出用戶端在數位指紋產生的過程當中可能有所錯誤，或是該用戶可能帶有惡意並且試圖利用假造的數位指紋來矇騙媒體分享伺服器。所以一旦驗證結果不符，媒體分享伺服器將發出錯誤訊息給用戶端，而用戶端則回到 RSA 公開金匙密碼系統步驟來重新產生公、私鑰以及數位指紋。

3.3 數位指紋嵌入技術

檔案發行者會事先隨機篩選出多媒體檔案中某些影像畫面內的「不連續處」(discontinuous area)。在此我們定義「不連續處」是指同一畫面中相鄰像素點之間有顯著差異或是兩個連續畫面之間同一座標的像素點有大幅變化的地方，這邊所指的顯著差異與大幅變化的程度可以定義一個色階臨界值，只要超過此臨界值的變化處即為我們所定

義的「不連續處」。

檔案發行者收到數位指紋之後，須從已篩選出的眾多不連續處中再隨機挑選出與數位指紋的位元個數相同數目的不連續處。例如：數位指紋的長度有 1024 個位元，所以檔案發行者則須從已篩選出的眾多不連續處中挑出 1024 個不連續處。事實上，依據數位指紋的位元個數而隨機挑選出的不連續處即為數位指紋對於該多媒體檔案的嵌入點(embedded point，以下皆簡稱為 EP)，因此這些嵌入點的位置資料將與該用戶的數位指紋及公鑰一併存入資料庫，例如以{FP, (C, N), EP data}的資料格式儲存。影像畫面中的不連續處是被用來嵌入數位指紋的絕佳位置，以人體視覺系統(Human Visual System)的觀點來看，若是在不連續處進行些微的色階變化，不但不會大幅影響整體影像的品質並且人體視覺也不易察覺出來。

數位指紋嵌入多媒體檔案時，根據數位指紋的位元資料而逐一將所對應的嵌入點進行色階變化。若數位指紋的位元為 1 時，則該位元所對應的嵌入點之像素值將被隨機選擇增加一個色階或是減少一個色階，如圖 2 所示；但若數位指紋的位元為 0 時，該位元所對應的嵌入點之像素值則不作更動，如圖 3 所示。

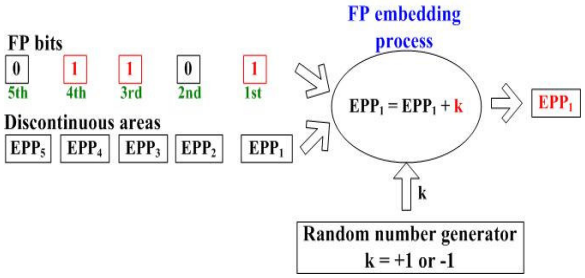

圖 2 當數位指紋位元為1時，則將所對應的嵌入點之像素值隨機增加或是減少一個色階。EPP_i：所選出的嵌入點 i 之像素值。

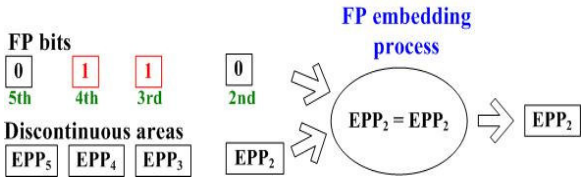

圖 3 當數位指紋位元為0時，所對應的嵌入點之像素值則不作更動。EPP_i：所選出的嵌入點 i 之像素值。

表 1 為數位指紋嵌入多媒體檔案的範例，由表中可以清楚看到，當數位指紋的位元為 1 時則將所對應的嵌入點之像素值隨機增加或減少一個色階；若當數位指紋的位元為 0 時則不更改原本的像素值。

檔案發行者端將嵌入點之位置資料與該用戶的數位指紋與公鑰存入資料庫之後，便將嵌入點的位置資料傳送到置於網路供應者端的媒體分享伺服器，而媒體分享伺服器接收到位置資料時必須確認多媒體檔案已經成功下載並完整解回之後，才能開始進行數位指紋的嵌入動作。媒體分享伺服器完成數位指紋的嵌入動作之後，則產生了一份帶有該用戶之數位指紋的多媒體檔案，並且再由用戶從媒體分享伺服器端下載回去。因為該用戶所下載到的檔案只供用戶自己觀賞用，在未經授權的情況下則不可散佈於網路上，所以用戶若是私自散佈已嵌入數位指紋的多媒體檔案於網路中，一旦檔案發行者查獲此未經授權散佈的多媒體內容，則可由檔案中的數位指紋與資料庫內的資料比對來反推出是哪位用戶散佈此份多媒體檔案，並且採取相對應的法律行動。

表 1 數位指紋嵌入多媒體檔案之範例

FP=10110…		
FP	**EP data**	**EP with embedded FP**
1	(EPL_1 , EPP_1)	(EPL_1 , EPP_1+1)
0	(EPL_2 , EPP_2)	(EPL_2 , EPP_2)
1	(EPL_3 , EPP_3)	(EPL_3 , EPP_3-1)
1	(EPL_4 , EPP_4)	(EPL_4 , EPP_4-1)
0	(EPL_5 , EPP_5)	(EPL_5 , EPP_5)
…	…	…
P.S. EPL_i：Location of EP_i EPP_i：Pixel data of EP_i		

4. 系統分析與討論

我們利用媒體分享伺服器根據各用戶端的下載請求而收集到 t 組相異的(x, h(x))資料之後，則可解回原資料區塊，並以此資料為 seed，配合 RSA 公開金匙密碼系統於數位簽章機制的應用來產生出各用戶端的數位指紋，以下我們則對此系統的特性進行探討與分析。

一、資料生存時間

我們利用多項式來進行來源資料編碼後，收集到固定組數的編碼內容，則可利用高斯消去法解回原資料區塊。事實上，來源資料編碼的用意在於提昇相異 h(x)值的數量，h(x)值並不是單純的資料內容，但是只要收到固定組數即可解回原資料區塊並且產生出新的 h(x)值。而此機制與傳統網路資料傳遞型態的不同之處在於來源資料編碼可以避免單點失敗(single point of failure)，並不會因為某些資料區塊的遺失而造成無法下載回完整資料，只要該多項式的 h(x)值持續存在固定數量於網路上，則此資料區塊必然可以成功下載完成。

二、計算複雜度

高階多項式為非線性運算，因此我們利用高斯消去法來解該多項式，則可避免線性相依的問題，

並且也省去檢測矩陣內的列向量是否皆為線性獨立所需的計算量。而數位指紋則是利用 RSA 加密演算法所產生，所以將此數位指紋的產生過程分散至各用戶端來產生，也就是由各用戶端將各自承擔 RSA 加密演算法的計算複雜度，如此可以減輕媒體分享伺服器的負擔，並且數位指紋於用戶本機端產生也能夠避免不必要的安全顧慮。

三、安全性

目前 RSA 加密演算法的金鑰長度至少需達到 1024 位元，方能稱得上是安全。所以數位指紋的產生過程將遵循目前 RSA 演算法的基本安全需求，則數位指紋的安全強度會與 RSA 應用於數位簽章機制相同。並且在我們的網路管理系統中，數位指紋的產生過程於用戶端進行，所以各用戶的私鑰不會洩漏於網路中，因此其他惡意攻擊者若是企圖偽造他人的數位指紋，則難以得到他人的私鑰；但若該攻擊者捨棄偽造一路，改為直接竊聽他人與媒體分享伺服器之間的通訊而從中聽取數位指紋的資料，則我們也利用對稱式加密機制對數位指紋加密後再進行傳送，所以此機制亦能抵擋竊聽攻擊。

另外，我們將媒體分享伺服器所收到的編碼內容為 seed，將其引入 RSA 公開金匙密碼系統來產生出公鑰與私鑰，在此媒體分享伺服器收到下載請求並開始收集編碼資料時，事實上不同用戶端透過媒體分享伺服器所收到的編碼資料必然不可能出現一模一樣的資料內容，所以媒體分享伺服器傳給各用戶端的 seed 也不可能出現相同的 seed 內容，所以各用戶端將 seed 引入 RSA 公開金匙密碼系統所產生出來的金鑰也不會是相同的組合，所以在產生數位指紋的過程將不會出現不同用戶卻產生出相同數位指紋的潛在問題。

至於資料隱藏的部份，只有取得檔案發行者端的資料庫中所紀錄之數位指紋資料，才能夠將嵌有數位指紋的多媒體檔案還原回未嵌入數位指紋的多媒體檔案。一旦帶有數位指紋的多媒體檔案被還原並且散佈於網路上，則檔案發行者將無法追溯出散佈檔案的非法用戶，所以必須盡力保護資料庫內容不對外洩漏，如此一來用戶的數位指紋資料將不會受到有心人士所擷取，並且網路上也不會出現不含數位指紋的原始多媒體檔案，則可保護檔案發行者與用戶的數位版權不受侵犯。

5. 結論與未來展望

隨著現今網路技術的發達，資料傳播變得越來越容易的情況之下，衍生出許多關於數位內容的版權問題，目前已經有許多數位浮水印等相關技術應用於多媒體內容版權的管理。我們提出一個新的數位版權管理方法，不但能夠針對不同用戶而給予不同且唯一的數位指紋，並且可以增加多媒體檔案於網路上的生存時間，提昇檔案的傳輸效率，也有效地降低內容供應者端的網路負載。我們提出來源資料編碼的方法將多媒體檔案進行編碼之後，再配合 RSA 公開金匙加密機制於數位簽章的應用，因而產生出專屬於用戶且唯一的數位指紋。我們也利用資料隱藏的方法，由人體視覺系統的觀點出發，將數位指紋隱藏在多媒體影像的不連續處，因此檔案發行者可以由非法散佈的多媒體影像中萃取出隱藏於內的數位指紋，之後則利用 RSA 驗證數位簽章的機制來追溯出非法散佈檔案的用戶，並且採取相對應的法律行動。此機制不但能夠於事後追溯出非法侵權的用戶，並且還可以於事前威嚇企圖非法散佈檔案的用戶，同時也提供了避免用戶的數位指紋遭到竊聽或是偽造的機制，所以我們所提出的理論的確可以同時提昇資料傳輸效率並且提供數位版權管理於多媒體內容。

本系統可以應用於多媒體的合法下載，多媒體發行業者(如：唱片公司或是電影公司)透過本系統來供與已訂購或授權的用戶端進行合法下載，利用本系統中類似 P2P 網路架構來進行檔案傳輸，並且擁有與一般的 P2P 網路相同的特性，也就是能夠提昇資料的生存時間以及避免資料的單點失敗，並且加入數位指紋來防止多媒體內容的非法散佈以及有心人士的惡意攻擊，可以提供給用戶有效率且安全的多媒體下載環境。

6. 誌謝

本論文特別感謝台灣網路安全測試平台(TWANST)與成功大學資通安全研究與教學中心(TWISC@NCKU)以及中華民國行政院國家科學委員會計畫於設備及經費上的補助。(計畫編號：NSC94-2219-E-006-005、NSC94-2219-E-006-007、NSC94-3114-P-006-001-Y)

7. 參考文獻

[1] Adi Shamir, "How to share a secret", Comm. of the ACM, 22(1), pp612–613, 1979

[2] Dan Boneh, James Shaw, “Collusion-Secure Fingerprinting for Digital Data”, IEEE transactions on information theory, Vol. 44, NO. 5, Sep. 1998

[3] Eugene, T.L., Reginald L.L., “Advances in Digital Video Content Protection”, Proc. of the IEEE, Special Issue on Advances in Video Coding and Delivery, 2004.

[4] H. hua Chu, L. Qiao, K. Nahrstedt, “A secure multicast protocol with copyright protection”, Proc. SPIE Security Watermarking of Multimedia Contents, Jan. 1999.

[5] I. Brown, C. Perkins, J. Crowcroft, ”Watercasting: Distributed watermarking of multicast media”, Proc. Networked Group Communication 1999.

[6] Inoue. H., Miyazaki. A., Yamamoto. A., Katsura. T., ”A digital watermark based on the wavelet

transform and its robustness on image compression", IEEE International Conference on Image Processing, 1998.
[7] Judge. P., Ammar. M., "Security issues and solutions in multicast content distribution: a survey", IEEE Network, 2003.
[8] Kirovski. D., Malvar. H., Yacobi. Y., "A dual watermark-fingerprint system", Multimedia, IEEE Volume 11, Issue 3, July-Sept. 2004
[9] Min Wu, Trappe, W., Wang, Z.J., Liu, K.J.R., "Collusion-resistant fingerprinting for multimedia", Signal Processing Magazine, IEEE Vol. 21, Issue 2, Mar 2000
[10] N.R. Wagner, "Fingerprinting", Proc.of the 1983 Symposium on Security and privacy, (Oakland, California), IEEE, April 1983
[11] P.H.W. Wong, O.C. Au, Y.M. Yeung, "Novel blind multiple watermarking technique for images", IEEE Transactions on Circuits and Systems for Video Technology, Volume 13, Issue 8, Aug. 2003
[12] R.L. Rivest., A. Shamir., L.A. Adleman, "A method for obtaining digital signatures and public-key cryptosystems", Comm. of the ACM, Vol.21, Nr.2, 1978
[13] S.E. Deering. "Host Extensions for IP Multicasting", RFC 1112, August 1989
[14] Trappe, W. Jie Song Poovendran, R. Liu, K.J.R." Key management and distribution for secure multimedia multicast" , IEEE Transactions on Multimedia ,2003
[15]Wen-Nung Lie, Guo-Shiang Lin, Chih-Liang Wu, Ta-Chun Wang, "Robust image watermarking on the DCT domain", Circuits and Systems, 2000. Proceedings of ISCAS 2000 Geneva. The 2000 IEEE International Symposium on, Volume 1, 28-31 May 2000