適用於供應鏈之連線 RFID 所有權轉移協定

許建隆[1、4]、詹宜勳[2]、蔡國裕[3*]、黃美涓[4*]
[1]長庚大學資訊管理學系、[2]長庚大學資訊管理學系、
[3]華夏科技大學資訊管理系、[4]長庚紀念醫院
[1]clhsu@mail.cgu.edu.tw、[2]tkshade@gmail.com、[3]kytsai@cc.hwh.edu.tw、
[4] walice@adm.cgmh.org.tw

摘要

本研究設計連線之無線射頻識別技術(Radio Frequency Identification, RFID)所有權轉移機制，允許標籤擁有者與標籤透過連線後端系統，進行雙向鑑別、金鑰協議及所有權轉移。再者，本研究進一步考量物流供應鏈管理的大量物品管理及群體鑑別、物品間的群體關係等議題，基於上述 RFID 所有權轉移協定設計連線之群體導向 RFID 所有權轉移機制，允許擁有者及標籤群體與後端系統連線後，進行雙向鑑別、金鑰協議及所有權轉移，而標籤上無需儲存任何有關群體的資訊。此外，倘若擁有者試圖欺騙供應鏈夥伴，我們所提出之機制具備稽核功能，以防範篡改標籤資訊等情況。

關鍵詞：RFID、所有權轉移、雙向鑑別、金鑰協議

Online RFID Ownership Transfer Protocol in Supply Chains

Chien-Lung Hsu [1], Yi-Shun Chan [2], Kuo-Yu Tsai[3*], May-Kuen Wong[4]
[1]Department of Information Management, Chang-Gung University, Tao-Yuan 33302, Taiwan,
[2]Department of Information Management, Chang-Gung University, Tao-Yuan 33302, Taiwan,
[3]Department of Management Information Systems, Hwa Hsia Institute of Technology, New Taipei City 235, Taiwan
[4]Chang Gung Memorial Hospital
[1]clhsu@mail.cgu.edu.tw, [2]tkshade@gmail.com, [3]kytsai@cc.hwh.edu.tw,
[4]walice@adm.cgmh.org.tw

Abstract

This paper focuses on the online RFID (Radio Frequency Identification) ownership transfer protocol, in which the tag owners (including the old tag owner and the new tag owner) and the tag cooperate to perform mutual authentication, key agreement, and ownership transfer via the online backend system. Furthermore, we consider some issues about commodities management, group authentication, and group relationship between commodities in supply chains. Based on the proposed online group-oriented RFID ownership transfer

protocol, the tag owners and the tags cooperate to perform mutual authentication, key agreement, and ownership transfer via the online backend system, and the tags have no need to store any group information. Both of the proposed protocols provide an auditing mechanism for creating tamper-proof logs to detect cheaters in supply chains.

Keywords: RFID, ownership transfer, mutual authentication, key agreement

壹、前言

無線射頻識別技術(Radio Frequency Identification, RFID)乃運用無線射頻(Radio Frequency, RF)發展的一種非接觸式識別技術，具備穿透性、重複使用性、自動化及即時存取性等特性。RFID 技術分為三部分組成，包括伺服器(Server)、讀取器(Reader)及標籤(Tag)。標籤是一種微型晶片貼附於物件上，用於識別此物件。讀取器用來連結標籤與伺服器間的通訊，可視為標籤與伺服器間的溝通橋樑。伺服器內存放標籤與讀取器的各類相關資料，使用者可透過伺服器接受服務。受惠於近來 RFID 晶片設計技術的進步與大量生產的助益，RFID 設備成本逐年下降。於 2001 年已設計出大小約 0.4mm^2 的晶片[3]，如此使得 RFID 技術的適用性大增。現今多種應用服務都相當適合結合 RFID 技術，諸如：物流、人員控管、藥品管理、食品管理及行動付款等，零售業龍頭 Wal-Mart、Procter & Gamble 及美國國防部均採用 RFID 技術來自動化管理其供應鏈[1]。自 2003 年，Wal-Mart 決定採用 RFID 技術為倉儲物流的標準方式後，並要求與其合作的上百家企業如 Gillette、Kraft Foods 等必須採用 RFID 技術，此舉引發 Bentton、Tesco、Target 等企業也投入 RFID 技術發展中。

對於企業營運而言，物流與資訊流的正確性與及時性是產業整體價值鏈的關鍵因素之一，相較於傳統條碼(Barcode)，RFID 技術具備較佳存取速度與寫入容量，並且容許業者進行多次存取資訊，因此業者便能對相對應的物件紀錄其位置、日期、產品類型等相關資訊。再者，RFID 技術不但能滿足傳統條碼的物件識別功能，藉由其無線射頻的特性，可在遠距離且無讀取死角的情況下讀取目標物件的相關訊息，來填補條碼技術所不足的部分。因此隨著 RFID 技術的成熟，RFID 技術已逐漸被採用於供應鏈活動中，用以協助企業提升物流的處理速度，進而提升整體供應鏈體系價值。此外，RFID 技術具備較高的安全性，RFID 技術亦可以運用在風險層級較高的應用，如商品防竊與防偽。

供應鏈中採用 RFID 技術後為了效益與方便，讓供應鏈夥伴的讀取器皆能夠存取標籤資料，將導致供應鏈運作產生資安疑慮，如下圖 1 所示。

* 通訊作者 (Corresponding author.)

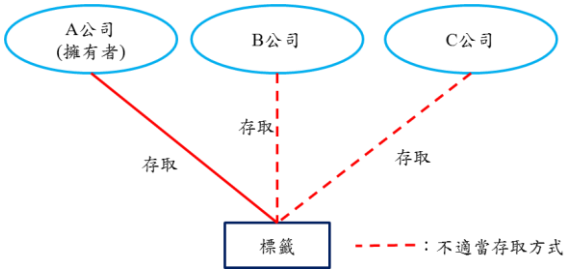

圖1：不適當存取方式

此應用方式導致供應鏈成員無法有效控標籤資訊存取，將會產生下列威脅：

- 未經授權篡改標籤資訊：假設供應鏈內存在AB兩家企業，當A企業持有標籤時，卻遭B企業篡改標籤相關資訊。如此將導致A企業之營運績效下降，甚至B企業能夠藉此掩蓋自己所發生之營運疏失。
- 未經授權讀取標籤資訊：假設供應鏈內存在AB兩家企業，若A企業可以使用讀取器讀取B企業之標籤資訊，反之亦然，此舉將導致企業之敏感資訊洩漏。

一般 RFID 系統均無法被預防此類威脅，原因在於 RFID 所設計之密碼機制均是用於防範外部攻擊者。若供應鏈內存在內部攻擊者，已持有供應鏈內所使用之金鑰，因此可輕易達到上述之攻擊。過往企業導入 RFID 技術需克服諸多問題，例如技術、標準、成本等 [4]，現今企業需再更進一步克服資安問題，否則無法讓使用者安心使用 RFID 系統。目前業界解決上述資安問題，採用 EPCglobal 所採行之方法，即對標籤輸入 Kill 指令。當標籤接受 Kill 指令後立即停止運作也無法再度喚醒，藉此排除上述的資安問題發生。然而，此一做法延生另外一個問題，當數以萬計的標籤停止運作後，其利害關係人將無法再享受到任何後續相關服務[2]，因此若無法提出適當的解決方法，而繼續使用 Kill 指令，其影響的層面將相當廣大。

本論文主要以 RFID 所有權轉移來解決上述問題，合法持有標籤之擁有者才具備標籤所有權，擁有標籤所有權方能對標籤進行存取，否則需透過擁有者授權或標籤所有權轉移後才能進行存取。本論文所提出之供應鏈架構中，供應鏈成員間存在可信任機構 (Trusted Authority)，用以擔任供應鏈成員間的溝通橋梁。此架構中僅有擁有者方能存取標籤，如圖 2 所示。

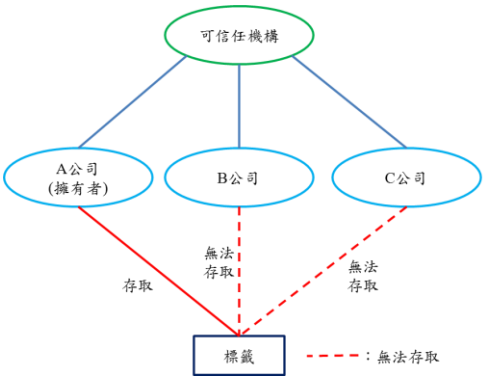

圖2：適當的存取方式

除此之外企業營運時需具備所有權轉移概念，當標籤由 A 企業交付至 B 企業時，則進行所有權轉移用以轉移標籤所有權。透過此概念與密碼機制的保護下，便可避免未經授權的篡改與存取，如下圖 3。

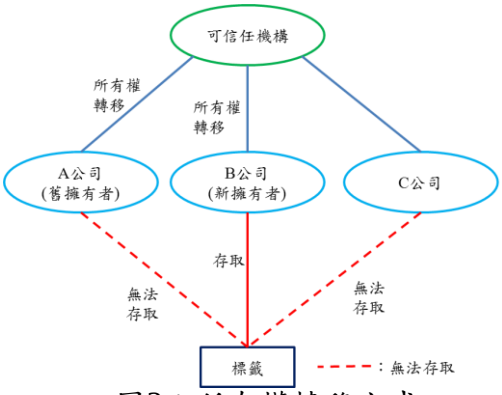

圖3：所有權轉移方式

企業供應鏈中存在相當高的作業複雜性，本論文提出連線環境下運作之 RFID 所有權轉移協定，並且協定的運作單位又可區分為標籤個體與標籤群體，藉此提升企業的彈性與效率。綜合以上運作環境與運作單位，本研究提出兩種所有權轉移協定，皆具備稽核機制，即存在公正第三者(Trusted Authority, TA)，TA 主責稽核擁有者讀取器與標籤所儲存之資訊是否正確無誤，以避免擁有者試圖欺瞞其他供應鏈成員。所有權轉移協定分別敘述如下。第一個所有權轉移協定為供應鏈內連線 RFID 所有權轉移協定，經由我們所提出之協定兩個企業間轉移標籤之所有權，滿足供應鏈基本需求；第二個所有權轉移為供應鏈內群體導向連線 RFID 所有權轉移協定，可進行多個標籤進行之所有權轉移，可增進供應鏈成員運用 RFID 技術的效益，而 TA 可稽核擁有者讀取器、標籤群體完整性及標籤群體所儲存之資訊是否正確無誤。

貳、供應鏈內連線 RFID 所有權轉移協定

當擁有者將標籤交付至新擁有者時，必須進行所有權轉移，使得標籤所有權可以移轉至新擁有者，避免擁有者試圖再次存取標籤。我們所提出之第一個協定在與伺服器連線的環境中，轉移標籤所有權至新擁有者。

2.1 連線 RFID 所有權轉移協定

第一個協定包含鑑別服務與所有權轉移服務，在進行鑑別服務時，需要包括初始、註冊、金鑰產生、鑑別等階段，而在進行所有權轉移服務時，所需之階段包括初始、註冊、金鑰產生、所有權轉移等階段。在所提出之協定中，擁有者在日常營運時透過鑑別服務確認擁有者、讀取器及標籤之身分合法性；新擁有者使用所有權轉移服務來變更標

籤所有權。所使用之符號如表一所示。

表一：協定中所使用之符號表

S	供應鏈中之 TA	$K_{sub,k+1}$	新擁有者與 TA 間之分享祕密
S_k	供應鏈成員之一，目前標籤擁有者，k 為索引值	$K_{sr,j}$	擁有者與讀取器間之分享秘密
MID_k	擁有者本身之識別符，k 為索引值	$K_{sr,j+1}$	新擁有者與讀取器間之分享秘密
S_{k+1}	標籤新擁有者，k 為其索引值	α_i	TA 與標籤間之分享秘密
R_j	擁有者持有之讀取器，j 為索引值	$K_{st,i}$	擁有者與標籤間之分享秘密
RID_j	讀取器本身之識別符，j 為索引值	$Knew_{st,i}$	擁有者與標籤間之新秘密
RID_{j+1}	新擁有者持有的讀取器，j 為索引值	r_x	伺服器所發出之亂數，x 為索引值
T_i	成員所擁有之標籤，i 為其索引值	M_x	通訊中傳遞之訊息，x 為索引值
TID_i	標籤本身之識別符，i 為其索引值	E	對稱式加密演算法
ms	TA 持有之秘密	H	雜湊演算法
$K_{sub,k}$	擁有者與 TA 間之分享祕密		

初始階段：TA 使用 **Setup** 演算法決定秘密參數與密碼演算法，定義如下。

- **Setup** 演算法
  - 輸入：(安全參數 1^l)
  - 輸出：(H, E, ms)
  - 演算法：
    - 選擇單向雜湊函數 H:$\{0,1\}^* \rightarrow Z_q$。
    - 選擇對稱式加密演算法 E。
    - 選擇亂數秘密 ms。

  伺服器將參數(H, E)公開，且秘密保存(ms)。

註冊階段：註冊演算法分為 **Reg_m** 與 **Reg_u**。其中，**Reg_m** 是供應鏈成員與標籤向 TA 之註冊方式，讓供應鏈成員與標籤需經由 **Reg_m** 演算向 TA 註冊取得合法金鑰，需要三者相互合作方可完成。**Reg_u** 是讀取器向供應鏈成員之註冊方式，讓讀取器向供應鏈成員註冊以取得合法金鑰。**Reg_m** 演算法與 **Reg_m** 演算法之說明分別如下所示：

- **Reg_m** 演算法：
  - 輸入：(ms, MID_k, TID_i)
  - 輸出：($K_{sub,k}$, $K_{st,i}$, α_i)
  - 演算法
    - 成員→TA：(MID_k)

      供應鏈成員傳送本身 MID_k 給與 TA，TA 計算 $K_{sub,k}=H(ms||MID_k)$，做為與成員間之共享金鑰。
    - TA→成員：($K_{sub,k}$)

      TA 透過安全方式將 $K_{sub,k}$ 交付給成員，而成員收到 $K_{sub,k}$ 後將其儲存。
    - 標籤→TA：(TID_i)

      標籤傳送本身 TID_i 給與 TA，TA 計算 $\alpha_i=H(ms||TID_i)$與選擇亂數 $K_{st,i}$ 做為與標籤間之共享金鑰。

- TA→標籤：(α_i, $K_{st,i}$)
  TA 透過安全方式將 α_i 與 $K_{st,i}$ 交付給標籤，而標籤收到後，將其儲存。
- TA→成員：($K_{st,i}$)
  TA 透過安全方式將 $K_{st,i}$ 交付給成員，而成員收到 $K_{st,i}$ 後將其儲存。

- **Reg_u 演算法**
  - 輸入：($K_{sub,k}$, RID_j)
  - 輸出：($K_{sr,j}$)
  - 演算法
    - 讀取器→成員：(RID_j)
      讀取器傳送本身 RID_j 給與成員，而成員計算 $K_{sr,j}=H(K_{sub,k}||RID_j)$，做為與讀取器間之共享金鑰。
    - 成員→讀取器：($K_{sr,j}$)
      成員透過安全方式將 $K_{sr,j}$ 交付給成員，而讀取器收到 $K_{sr,j}$ 後將其儲存。

2.2 擁有者鑑別服務

在日常營運中，擁有者需要對伺服器、讀取器及標籤進行三方鑑別，以確定三者身分正確無誤，以下將詳細說明各階段。

金鑰產生階段：進行鑑別階段前，擁有者必須進行此階段方能獲取讀取器合法金鑰。下述為產生讀取器合法金鑰之 **Key_gen** 演算法。

- **Key_gen** 演算法
  - 輸入：($K_{sub,k}$, RID_j)
  - 輸出：($K_{sr,j}$)
    - 供應鏈成員計算 $K_{sr,j}=H(K_{sub,k}||RID_j)$，以獲取讀取器金鑰。

鑑別階段：擁有者、讀取器及標籤進行三方鑑別之演算法，稱為 **Auth** 演算法。

- **Auth** 演算法
  - 輸入：(TID_i, $K_{sr,j}$, $K_{st,i}$)
  - 輸出：*True* 或 *False*。*True* 代表通過鑑別；*False* 則代表無通過鑑別。
  - 演算法
    - 讀取器→標籤：(r_1)
      讀取器產生亂數 r_1，傳送至標籤。
    - 標籤→讀取器：(TID_i, M_1, r_2)
      標籤收到 r_1 後，計算 $M_1=H(K_{st,i}||r_1)$與產生亂數 r_2，並傳送 TID_i、M_1 及 r_2 至讀取器。
    - 讀取器→成員：(TID_i, M_1, r_1, r_2)
      讀取器收到訊息後，再傳送(TID_i, M_1, r_1, r_2)至供應鏈成員。

- 成員→讀取器：(M_3)
  檢驗 M_1 是否有等於 $H(K_{st,i}||r_1)$。若驗證失敗，則回傳 *False* 訊息。若訊息驗證正確，則產生亂數 r_3，並使用 $K_{st,i}$ 計算鑑別訊息 $M_2=H(K_{st,i}||r_2)$與密文 $M_3= E_{K_{sr,j}}(r_1, r_3, M_2)$，接著傳送 M_3 至讀取器。

- 讀取器→成員：(r_3)
  讀取器接收 M_2 與 M_3 後，解密 M_3 以獲取 r_1、r_3 及 M_2，並檢驗 r_1 是否與所選取之 r_1 相等。若驗證失敗，則回傳 *False* 訊息。若驗證正確，則回傳 *True*，代表讀取器可信任成員與標籤身分，並回傳 r_3 至供應鏈成員。
- 成員收到 r_3 後，檢驗 r_3 是否與所加密之 r_3 相等。若驗證失敗，回傳 *False* 訊息。若驗證正確，則回傳 *True*，代表成員可信任讀取器與標籤。
- 讀取器→標籤：(M_2)
  接著傳送 M_2 至標籤。
- 標籤收到 M_2 後，檢驗 M_2 是否與 $H(K_{st,i}||r_2)$相等。若驗證失敗，則回傳 *False* 訊息。若驗證正確，則回傳 *True*，代表標籤可信任讀取器與成員身分。

2.3 新擁有者所有權轉移服務

以下描述新擁有者如何進行金鑰產生階段與所有權轉移階段。

金鑰產生階段：於所有權轉移階段前，新擁有者與 TA 需先執行需先執行 **OT_key_gen**，以產生讀取器金鑰與供應鏈成員金鑰。

- **OT_key_gen 演算法**
  - 輸入：(ms, MID_{k+1}, $K_{sub,k+1}$, RID_{j+1})
  - 輸出：($K_{sr,j+1}$, $K_{sub,k+1}$)
    - 供應鏈成員計算 $K_{sr,j+1} =H(K_{sub,k+1}|| RID_{j+1})$，以獲取讀取器金鑰。
    - TA 計算 $K_{sub,k+1} =H(ms|| MID_{k+1})$，以獲取供應鏈成員金鑰。

所有權轉移階段：此階段包含鑑別標籤與所有權轉移兩部分，步驟說明如下。

- **OT** 演算法
  - 輸入：(TID_i, $K_{sub,k+1}$, $K_{sr,j+1}$, $K_{st,i}$)
  - 輸出：*True* 或 *False*。*True* 代表所有權轉移成功，*False* 代表所有權轉移失敗。
  - 演算法：
    - 讀取器→標籤：(r_1)
      讀取器產生亂數 r_1，傳送至標籤。
    - 標籤→讀取器：(TID_i, M_1, r_2)
      標籤收到 r_1 後，產生亂數 r_2 與計算 $M_1=H(K_{st,i}||r_1)$，並傳送(TID_i, M_1, r_2)至讀取器。

- 讀取器→成員：(TID_i, M_1, r_1, r_2)
  讀取器收到訊息後，再傳送(TID_i, M_1, r_1, r_2)至成員。
- 成員→TA：(TID_i, M_1, r_1, r_3)
  成員接受訊息後，產生亂數 r_3，並傳送 TID_i、M_1、r_1 及 r_3 給 TA。
- TA→成員：(M_2)
  接收訊息後，TA 檢驗 M_1 是否等於 $H(K_{st,i}\| r_1)$。若驗證失敗，回傳 *False*。若驗證成功，則計算標籤新金鑰 $Knew_{st,i}= H(K_{st,i}\| \alpha_i)$與 $M_2= E_{K_{sub,k+1}}(Knew_{st,i}, r_3,)$，並傳送 M_2 給成員。
- 成員→讀取器：(M_4)
  成員解密 M_2 獲得 $Knew_{st,i}$ 與 r_3，並檢驗密文中之 r_3 是否有等於先前傳送之 r_3。若驗證失敗，回傳 *False*。若驗證成功，則成員便可相信標籤與 TA，並能獲取標籤新金鑰。接著計算鑑別訊息 $M_3= H(Knew_{st,i}\| r_2)$與密文 $M_4= E_{K_{sr,j+1}}(r_2, M_3)$，並傳送 M_4 至讀取器。
- 讀取器→標籤：(M_3)
  讀取器解密 M_4 獲得 r_2 與 M_3，並檢驗 r_2 是否等於所傳送之 r_2。若驗證失敗，回傳 *False*。若驗證成功，讀取器可相信成員與標籤，並傳送 M_3 至標籤。
- 標籤→讀取器：(M_5)
  標籤接受訊息後，計算新金鑰 $Knew_{st,i}=H(K_{st,i}\| \alpha_i)$，並透過新金鑰檢驗 M_3 是否等於 $H(Knew_{st,i}\| r_2)$。若驗證成功，標籤可相信金鑰正確無誤且 TA、讀取器及成員身分是正確，並計算 $M_5= H(Knew_{st,i}\| r_1)$與更新金鑰為 $Knew_{st,i}$。若驗證失敗，計算 $M_5=H(K_{st,i}\| r_1)$，並傳送 M_5 給讀取器。
- 讀取器→成員：(M_5)
  讀取器接著傳送 M_5 給成員。
- 成員檢驗 M_5 是否等於 $H(Knew_{st,i}\|r_1)$。若驗證成功，則通過所有權轉移。若驗證失敗，則重新執行所有權轉移演算法。

2.4 TA 稽核服務

供應鏈成員間採用所有權轉移機制後，能確保唯有擁有者方能存取標籤。然而，於供應鏈中，唯有擁有者能夠存取標籤，其他供應鏈成員只能單方面相信擁有者所提供之訊息而無法查證，可能造成擁有者欺瞞其它供應鏈成員。例如，在供應鏈中倘若擁有者之疏忽照成溫度失控，擁有者將試圖篡改標籤中所記錄之溫度，以掩飾自身疏忽，接著再交付篡改後之標籤給其他供應鏈成員，導致其它成員之權益受損。因此，本協定提供稽核服務，若供應鏈成員有不法情事，可透過 TA 進行稽核，以嚇阻擁有者試圖進行不法行為。稽核服務包括初始階段、註冊階段、金鑰產生及稽核階段，初始與註冊階段與上述內容相同。以就僅說明 TA 應如何進行金鑰產生與稽核。

金鑰產生階段：進行稽核時，必須具讀取器金鑰與供應鏈成員金鑰，若無合法金鑰，則搜集之資訊也無法轉換成有意義的資訊。因此，TA 應先進行 **Key_gen** 產生相關金鑰。

- **Key_gen** 演算法
  - 輸入：(ms, MID_k, RID_j)
  - 輸出：($K_{sr,j}$)
    - TA 先計算欲被稽核之成員金鑰 $K_{sub,k} = H(ms||MID_k)$，以獲取成員金鑰，並計算欲被稽核之讀取器金鑰算 $K_{sr,j}=H(K_{sub,k}||RID_j)$，以獲取讀取器金鑰。

稽核階段：透過 **Audit** 演算法，TA 可確認擁有者之讀取器與標籤的身分是否合法，並存取其中所儲存之資訊。

- **Audit** 演算法
  - 輸入：(TID_i, $K_{sr,j}$, $K_{st,i}$)
  - 輸出：*True* 或 *False*。*True* 代表通過稽核;*False* 則代表無通過稽核。
  - 演算法
    - 讀取器→標籤：(r_1)
      讀取器產生亂數 r_1，傳送至標籤。
    - 標籤→讀取器：(TID_i, M_1, r_2)
      標籤收到 r_1 後，計算 $M_1= H(K_{st,i} ||r_1)$與產生亂數 r_2，並傳送 TID_i、M_1 及 r_2 至讀取器。
    - 讀取器→TA：(TID_i, M_1, r_1, r_2)
      讀取器收到訊息後，再傳送 TID_i、M_1、r_1 及 r_2 至 TA。
    - TA→讀取器：(M_3)
      檢驗 M_1 是否有等於 $H(K_{st,i}|| r_1)$。若驗證失敗，則回傳 *False* 訊息。若驗證成功,則產生亂數 r_3,並計算鑑別訊息 $M_2= H(K_{st,i}|| r_2)$與密文 $M_3= E_{K_{sr,j}}(r_1, r_3, M_2)$，接著傳送 M_3 至讀取器。
    - 讀取器→TA：(r_3)
      讀取器接收 M_2 與 M_3 後，解密 M_3 以獲取 r_1、r_3 及 M_2，並檢驗 r_1 是否與所傳送之 r_1 相等。若驗證失敗,則回傳 *False* 訊息。若驗證成功,則回傳 *True*,代表讀取器可信任成員與標籤身分，並回傳 r_3 至 TA。
    - TA 收到 r_3 後，檢驗 r_3 是否與適才所加密之 r_3 相等。若驗證失敗，則回傳 *False*。若驗證成功，則回傳 *True*，代表 TA 可信任讀取器與標籤身分。
    - 讀取器→標籤：(M_2)
      接著傳送 M_2 至標籤。
    - 標籤收到 M_2 後,檢驗 M_2 是否與 $H(K_{st,i}|| r_2)$相等。若驗證失敗,則回傳 *False* 訊息。若驗證成功，則回傳 *True*，代表標籤可信任讀取器與 TA 身分。

參、 供應鏈內群體導向連線 RFID 所有權轉移協定

標籤群體運作乃將多個標籤個體視為群體，並對群體進行鑑別與所有權轉移，縮短協定運作之時間與將發展更多元的應用，諸如：貨物批次管理、貨櫃管理、棧板管理等。

3.1 群體導向連線 RFID 所有權轉移協定

群體導向所有權轉移協定具備擁有者群體鑑別服務與新擁有者群體所有權轉移服務，其中擁有者群體鑑別服務包括初始、註冊、金鑰產生、裝箱及群體鑑別等階段，新擁有者群體所有權轉移服務初始、註冊、金鑰產生、token 交付及群體所有權轉移等階段，其中初始與註冊與前章相同。本論文使用動態群體之組成，亦即群體中的標籤成員組成可自由增加或移除，並且標籤無需紀錄其他額外訊息。擁有者可透過群體導向鑑別服務確認標籤群體之完整性，檢視標籤群體是否有所缺漏，並透過所有權轉移來交付標籤群體給予新擁有者。本章節所使用之變數與表一相同，新增變數如下表二所示。

表二：群體導向連線所有權轉移協定符號表

$msg_{x,i}$	傳送至標籤群體或標籤群體傳送出之訊息，x 與 i 為其索引值	R	標籤群體識別碼

3.2 擁有者群體導向鑑別服務

群體導向鑑別服務中不僅鑑別單一的標籤個體，更鑑別標籤群體是否正確，若是標籤群體中標籤個體發生錯置或遺漏，都無法通過鑑別。

金鑰產生階段：進行群體鑑別階段前，擁有者必須進行此階段方能獲取讀取器合法金鑰，下述為產生讀取器合法金鑰之 **Key_gen** 演算法。

- **Key_gen** 演算法
  - 輸入：$(K_{sub,k}, RID_j)$
  - 輸出：$(K_{sr,j})$
  - 演算法
    - 供應鏈成員計算 $K_{sr,j} = H(K_{sub,k} \| RID_j)$，以獲取讀取器金鑰。

裝箱階段：此階段將決定那些標籤要被歸屬於同一群體。使用者先決定那些標籤要屬同一群體，再將此群體放置於箱中。接著持用讀取器執行 **Packing** 演算法便可以計算出標籤群體識別碼，用以代表此群體。

- **Packing** 演算法：
  - 輸入：$(TID_i, K_{st,i}, K_{sr,j},)$
  - 輸出：R。
  - 演算法

- 讀取器→標籤：(r_1)
  讀取器產生亂數 r_1 傳送至標籤，直到群體中每個標籤都有收到 r_1。
- 標籤→讀取器：(TID_i, $msg_{1,i}$)
  收到訊息後，標籤計算 $msg_{1,i}=H(K_{st,i}||r_1)$，回覆給讀取器。
- 讀取器→成員：(TID_i, $msg_{1,i}$, r_1)
  讀取器將標籤群體之 TID_i、$msg_{1,i}$ 與 r_1 傳送給成員。
- 成員→讀取器：(M_1)
  成員收到訊息後，先檢驗 $msg_{1,i}$ 是否等於 $H(K_{st,i}||r_1)$。若等於，則進行 **R_value** 演算法以計算 R，接著計算密文 $M_1= E_{K_{sr,j}}(r_1, R)$，並傳送致讀取器。
- 讀取器接收訊息後，解密 M_1 獲取 r_1 與 R，接著檢驗 r_1 是否等於適才所發送的 r_1。若等於，則儲存 R。

- **R_value** 演算法
  - 輸入：(n, TID[])，其中 n 為標籤數量，TID[]為一陣列，儲存標籤群體之 TID_i。
  - 輸出：R
  - 演算法

```
R_value (int n, int TID[])
{
    int R, tmep;
    for (int i=n; i≧1; i--)
       {
          temp=temp||TID[i];
       }
    R=Hash(temp);
    return R;
}
```

圖 4：**R_value** 演算法

群體導向鑑別階段：若標籤群體中有錯置或遺落皆無法通過鑑別，讀取器中只需儲存 R 即可達到目的，而不需要儲存全部標籤群體之訊息，標籤中也不需要額外紀錄其他訊息。

- **G_auth** 演算法
  - 輸入：(TID_i, R, $K_{sr,j}$, $K_{st,i}$)
  - 輸出：*True* 或 *False*。*True* 代表通過鑑別；*False* 則無通過鑑別。
  - 演算法
    - 讀取器→標籤：(r_1)
      讀取器產生亂數 r_1 傳送至標籤，直到群體中每個標籤都有收到 r_1。
    - 標籤→讀取器：(TID_i, $msg_{1,i}$, r_2)
      收到訊息後，標籤先產生亂數 r_2，並計算 $msg_{1,i}=H(K_{st,i}||r_1)$，回覆給讀取器。
    - 讀取器→成員：(TID_i, $msg_{1,i}$, r_1, r_2)

讀取器先進行 **checkR** 演算法檢查標籤群體是否正確。若正確，則將標籤群體之 TID_i、$msg_{1,i}$、r_1 及 r_2 傳送給成員。

- 成員→讀取器：(M_1)
  成員收到訊息後，先檢驗 $msg_{1,i}$ 是否等於 $H(K_{st,i}||r_1)$，以檢查標籤個體是否正確。若驗證失敗，則回傳 *False*。若驗證成功，則產生亂數 r_3，接著計算 $msg_{2,i}= H(K_{st,i}|| r_2)$與密文 $M_1= E_{K_{sr,j}}(r_1, r_3, msg_{2,i})$，並 M_1 傳送至讀取器。
- 讀取器→成員：(r_3)
  讀取器解密 M_1 獲得 r_1、$msg_{2,i}$ 及 r_3 後，先檢驗 r_1 是否等於適才所傳送之 r_1。若驗證失敗，則回傳 *False*。若驗證成功，則回傳 *True*，且讀取器可相信成員與標籤身分，接著傳送 r_3 至成員。
- 讀取器→標籤：($msg_{2,i}$)
  讀取器緊接著傳送 $msg_{2,i}$ 至標籤。
- 成員接收訊息後，檢驗 r_3 是否等於適才所傳送之 r_3。若不等於，則回傳 *False*。若等於，則回傳 *True*，且成員可相信讀取器與標籤身分。
- 標籤檢驗 $msg_{2,i}$ 是否會等於 $H(K_{st,i}||r_2)$。若驗證失敗，則回傳 *False*。若驗證成功，則回傳 *True*，且標籤可相信成員與讀取器身分合法。

- **checkR** 演算法
  - 輸入：(n, *TID*[], R_1)，其中 n 代表標籤個數，*TID*[]為陣列， R_1 為讀取器所持有之 R 值。
  - 輸出：*True* 或 *False*。*True* 代表檢查通過，*False* 代表檢查未通過。
  - 演算法：

```
checkR (int n, int TID[], int R1)
{
    int R2, tmep;
    for (int i=n; i≧1; i--)
       {
          temp=temp||TID[i];
       }
    R2=Hash(temp);
    if (R1==R2)
    return True;
       else
    return False;
}
```

圖 5：**checkR** 演算法

3.3 新擁有者群體導向所有權轉移服務

初始與註冊階段與前述相同，以下僅說明金鑰產生、token 交付及群體所有權轉移等

階段。

金鑰產生階段：進行群體所有權轉移前，新擁有者與 TA 必須進行此階段方能獲取合法金鑰，以下為產生合法金鑰之 **OT_Key_gen** 演算法。

- **OT_Key_gen** 演算法
  - 輸入：(ms , MID_{k+1}, MID_k, RID_j, $K_{sub,k+1}$)
  - 輸出：($K_{sr,j+1}$, $K_{sub,k+1}$, $K_{sub,k}$)
  - 演算法
    - 新擁有者計算 $K_{sr,j+1}=H(K_{sub,k+1}||RID_j)$，藉以獲取讀取器合法金鑰。
    - TA 計算 $K_{sub,k+1}=H(ms||MID_{k+1})$與 $K_{sub,k}=H(ms||MID_k)$，藉以獲取新擁有者與舊擁有者之合法金鑰。

token 交付階段：進行所有權轉移前，新擁有者需要透過 **Token_D** 演算法向舊擁有者索取 R(稱為 *token*)，之後方能確認標籤群體之組成是否正確。

- **Token_D 演算法**
  - 輸入：(R, $K_{sub,k}$, $K_{sub,k+1}$, $K_{sr,j+1}$, $K_{st,i}$)
  - 輸出：R
  - 演算法：
    - 新擁有者→讀取器：(r_1)
      新擁有者產生亂數 r_1 與並發送需求至 TA。
    - TA→舊擁有者：(r_2)
      TA 產生亂數 r_2 與並發送需求至 TA。
    - 舊擁有者→TA：(M_1, R)
      舊擁有者計算 $M_1= H(K_{sub,k}|| r_2|| R)$傳送給 TA。
    - TA→新擁有者：(M_2, R)
      TA 接收訊息後，檢驗 M_1 是否等於 $H(K_{sub,k}|| r_2|| R)$。若驗證成功，則計算 $M_2= H(K_{sub,k+1}|| r_2|| R)$。並傳送給新擁有者。
    - 新擁有者→讀取器：(M_3, R)
      新擁有者接收訊息後，檢驗 M_2 是否等於 $H(K_{sub,k+1}|| r_2|| R)$。若驗證成功，則計算 $M_3=H(K_{sr,j+1}|| R)$。並傳送給讀取器。
    - 讀取器接收後，檢驗 M_3 是否等於 $H(K_{sr,j+1}|| R)$。若驗證成功，則儲存 R。

群體導向所有權轉移階段：新擁有者可以使用 R 鑑別標籤群體是否正確，再經由 TA 鑑別標籤個體。而，標籤新金鑰將由 TA 計算並交付給新擁有者，如此可避免舊擁有者使用持有之金鑰存取標籤資訊。

- **G_OT** 演算法
  - 輸入：(TID_i, $K_{sub,k+1}$, $K_{sr,j+1}$, $K_{st,i}$, α_i)
  - 輸出：*True* 或 *False*。*True* 代表所有權轉移成功，*False* 代表所有權轉移失敗。
  - 演算法

- 讀取器→標籤：(r_1)
  讀取器產生亂數 r_1 傳送至標籤，直到群體中每個標籤都有收到 r_1。
- 標籤→讀取器：(TID_i, $msg_{1,i}$, r_2)
  標籤收到訊息後，產生亂數 r_2，並計算 $msg_{1,i}= H(K_{st,i}|| r_1)$，回傳給讀取器。
- 讀取器→新擁有者：(TID_i, $msg_{1,i}$, r_1, r_3)
  讀取器先進行 **checkR** 演算法檢查標籤群體是否正確。若不正確，則回傳 *False*。若正確，則產生亂數 r_3，並將 TID_i、$msg_{1,i}$、r_1 及 r_3 傳送給成員。
- 新擁有者→TA：(M_1)
  TA 接收訊息，先檢驗 $msg_{1,i}$ 是否等於 $H(K_{st,i}||r_1)$。若驗證不正確，則回傳 *False*。若驗證正確，則計算新標籤金鑰 $Knew_{st,i}= (K_{st,i}|| \alpha_i)$與密文 $M_1= E_{K_{sub,k+1}}$ ($Knew_{st,i}$, r_3)，並 M_1 傳送給新擁有者。
- 新擁有者→讀取器：(M_2)
  新擁有者接收訊息後，先解密 M_1 獲得 $Knew_{st,i}$ 與 r_3，並比較 r_3 是否與所傳送之 r_3 相等。若不相等，則回傳 *False*。若相等，代表 TA、標籤個體及標籤群體均正確無誤。接著計算 $msg_{2,i}= H(Knew_{st,i}|| r_2)$與密文 $M_2= E_{K_{sr,j+1}}$ ($msg_{2,i}$, r_1)，並傳送給讀取器。
- 讀取器→標籤：(TID_i, $msg_{2,i}$)
  讀取器接收訊息後，先解密 M_2 獲得 $msg_{2,i}$ 與 r_1，並比較 r_1 是否與所傳送之 r_1 相等。若不等於，則回傳 *False*。若等於，代表 TA、新擁有者與標籤群體及標籤個體均正確無誤，並接著傳送 TID_i、$msg_{2,i}$ 給予標籤。
- 標籤→讀取器：($msg_{3,i}$)
  標籤接受訊息後，將 TID_i 與本身 TID_i 進行比對，接著標籤計算新金鑰 $Knew_{st,i}=H(K_{st,i}||\alpha_i)$，並檢驗 $msg_{2,i}$ 是否等於 $H(Knew_{st,i}||r_2)$。若等於，則更新金鑰，並回傳 $msg_{3,i}= H(Knew_{st,i}|| r_1)$，代表金鑰更新成功。若不等於，則回傳 $msg_{3,i}= H(K_{st,i}|| r_1)$，代表金鑰更新失敗。
- 讀取器→新擁有者：($msg_{3,i}$)
  讀取器接收訊息後，轉送 $msg_{3,i}$ 至新擁有者。
- 新擁有者接收訊息後，檢驗 $msg_{3,i}$ 是否等於 $H(Knew_{st,i}||r_1)$，若不等於，則代表所有權轉移失敗。若相等，則回傳 *True*，代表所有權轉移成功。

3.4 TA 群體導向稽核服務

本論文亦提供群體稽核服務，TA 可以稽核標籤群體是否與當初宣稱的一致，並稽核標籤個體是否合法。初始與註冊階段與前述相同，以下僅說明 TA 應如何進行金鑰產生

階段與群體稽核階段之演算法。

金鑰產生階段：進行稽核前，TA 必須進行 **Key_gen**，產生讀取器與供應鏈成員金鑰。

- **Key_gen** 演算法
  - 輸入：(ms, MID_k, RID_j)
  - 輸出：($K_{sub,k}$, $K_{sr,j}$)
  - 演算法
    - TA 計算欲被稽核之成員金鑰 $K_{sub,k} = H(ms || MID_k)$。
    - TA 計算欲被稽核之讀取器金鑰算 $K_{sr,j}= H(K_{sub,k}||RID_j)$。

稽核階段：透過 **G_audit** 演算法，TA 可進行稽核以確認擁有者之讀取器與標籤之身分是否合法，以及標籤群體是否正確，並存取其中所儲存之資訊。演算法中所使用的 R，在所有權轉移階段已交付給 TA。

- **G_audit** 演算法
  - 輸入：(TID_i, R, $K_{sr,j}$, $K_{st,i}$)
  - 輸出：*True* 或 *False*。*True* 代表通過鑑別；*False* 則無通過鑑別。
  - 演算法
    - 讀取器→標籤：(r_1)
      讀取器產生亂數 r_1 傳送至標籤，直到群體中每個標籤都有收到 r_1。
    - 標籤→讀取器：(TID_i, $msg_{1,i}$, r_2)
      收到訊息後，標籤先產生亂數 r_2，並計算 $msg_{1,i}=H(K_{st,i}||r_1)$，回覆給讀取器。
    - 讀取器→TA：(TID_i, $msg_{1,i}$, r_1, r_2)
      讀取器將標籤群體之 TID_i、$msg_{1,i}$、r_1 及 r_2 傳送給 TA。
    - TA→讀取器：(M_1)
      TA 收到訊息後，先檢驗 $msg_{1,i}$ 是否等於 $H(K_{st,i}|| r_1)$，以檢查標籤個體是否正確。若不等於，則回傳 *False*。若等於，則產生亂數 r_3，接著計算 $msg_{2,i}= H(K_{st,i}|| r_2)$與密文 $M_1= E_{K_{sr,j}}(r_1 , r_3, msg_{2,i})$，並 $msg_{2,i}$ 傳送至讀取器。
    - 讀取器→TA：(r_3)
      讀取器解密 M_1 獲得 r_1、r_3 及 $msg_{2,i}$ 後，先檢驗 r_1 是否等於所傳送之 r_1。若不等於，則回傳 *False*。若等於，則回傳 *True*，且讀取器可相信成員與標籤身分，接著傳送 r_3 至 TA。
    - TA 接收訊息後，檢驗 r_3 是否等於所傳送之 r_3。若不等於，則回傳 *False*。若等於，則回傳 *True*，代表 TA 能相信讀取器與標籤身分。
    - 讀取器→標籤：($msg_{2,i}$)
      讀取器傳送 $msg_{2,i}$ 至標籤。
    - 標籤檢驗 $msg_{2,i}$ 是否會等於 $H(K_{st,i}||r_2)$。若不等於，則回傳 *False*。若等於，

則回傳 *True*，代表標籤可相信 TA 與讀取器身分合法。

肆、 安全分析

本節將對於我們所提出之協定進行安全分析，分成兩小節說明。

4.1 第一個協定之安全性分析

本節對於擁有者鑑別服務、新擁有者所有權轉移服務及 TA 稽核服務進行安全性分析。考量標籤之運算能力不足，因此我們所提出之協定僅採用對稱式加密演算法、雜湊演算法及虛擬亂數產生器，因此安全性也仰賴在上述密碼機制的破解難度上。

1. 雙向鑑別：雙向鑑別用於確認通訊雙方身分是否合法。
   - **Auth** 演算法：擁有者、讀取器及標籤三者之間透過詰問問答(Challenge-response)來鑑別對方身分。擁有者透過檢驗 $M_1=H(K_{st,i}\ \|r_1)$與 r_3，鑑別讀取器與標籤身分。讀取器經由解密 M_2 並檢驗其中之 r_1 是否與所傳送的 r_1 相等，鑑別擁有者與標籤身分。標籤透過 M_3 檢驗是否等於 $H(K_{st,i}\|r_2)$，鑑別擁有者與讀取器身分。
   - **OT** 演算法：新擁有者、讀取器及標籤三者之間亦經由詰問問答與 TA 之協助鑑別對方身分。由於新擁有者一開始並無與標籤分享金鑰，因此新擁有者需仰賴 TA 通知 $M_1=H(K_{st,i}\ \|r_1)$之的檢驗結果，並且 TA 加密標籤新金鑰後傳送給新擁有者。新擁有者解密 M_2 並檢驗其中之 r_3 是否等於所傳送 r_3，鑑別 TA 與標籤之身分。讀取器透過解密 M_4 並檢驗其中之 r_2 是否等於所傳送 r_2，鑑別 TA 與標籤身分。標籤則是檢驗 M_3 是否等於 $H(Knew_{st,i}\|r_2)$，鑑別新擁有者與讀取器身分。
   - **Audit** 演算法：TA、讀取器及標籤三者之間透過詰問問答鑑別對方身分。TA 透過檢驗 $M_1=H(K_{st,i}\ \|r_1)$與 r_3，鑑別讀取器與標籤身分。讀取器藉由解密 M_2 並檢驗其中之 r_1 是否與所傳送的 r_1 相等，鑑別 TA 與標籤身分。標籤透過 M_3 檢驗是否等於 $H(K_{st,i}\|r_2)$，鑑別 TA 與讀取器身分。
2. 機密性與完整性：若傳送之密文遭受篡改，其解密後之明文將成為無意義的訊息。經由檢驗明文，便可確保資訊之完整性。因此採用加密機制可以達到機密性與完整性。
   - **OT** 演算法：TA 產生標籤新金鑰後，加密標籤新金鑰與新擁有者所傳送之 r_3，並傳送至新擁有者。新擁有者接收訊息後，解密訊息並檢驗 r_3 是否等於所傳送的 r_3。若等於，則能相信標籤新金鑰未遭受篡改。
3. 重送攻擊：避免惡意人士重送訊息。每次進行通訊時，虛擬亂數均重新產生，

因此可透過虛擬亂數檢驗訊息是否被重送。

- **Auth** 演算法：鑑別協定中傳送之訊息有 M_1、M_2 與 M_3，訊息內容包含亂數 r_1、r_2 及 r_3，可經由驗證亂數，以確認是否遭受重送攻擊。
- **OT** 演算法：所有權轉移協定中傳送之訊息有 M_1、M_2、M_3、M_4 及 M_5，訊息內容包含亂數 r_1、r_2 及 r_3，分析與上述說明相同。
- **Audit** 演算法：稽核協定中傳送的訊息有 M_1、M_2 與 M_3，訊息內容包含亂數 r_1、r_2 及 r_3，分析與上述說明相同。

4. 中間人攻擊：為了避免惡意人士居中篡改訊息達到惡意攻擊的目的。藉由在訊息中加入虛擬亂數與密碼機制，以抵抗中間人攻擊。
   - **Auth** 演算法：鑑別協定中傳送之訊息中，$M_1= H(K_{st,i} \| r_1)$與 $M_2= H(K_{st,i} \| r_2)$ 為訊息鑑別碼，$M_3= E_{K_{sr,j}}(r_1, r_3, M_2)$為密文，其中皆需要使用共享金鑰計算。若攻擊者欲執行中間者攻擊，將面臨單向雜湊函數之困難度假設或加密演算法的計算困難度。即，我們所提出之協定確保訊息的來源性。
   - **OT** 演算法：所有權轉移協定中傳送之訊息中，$M_1=H(K_{st,i} \| r_1)$、$M_5= H(Knew_{st,i} \| r_1)$及 $M_3= H(Knew_{st,i} \| r_2)$ 為訊息鑑別碼，$M_2= E_{K_{sub,k+1}}(Knew_{st,i}, r_3,)$與 $M_4= E_{K_{sr,j+1}}(r_2, M_3)$為密文，分析與 **Auth** 演算法之說明相同。
   - **Audit** 演算法：稽核協定中傳送之訊息 $M_1= H(K_{st,i} \| r_1)$與 $M_2= H(K_{st,i} \| r_2)$ 為訊息鑑別碼，$M_3= E_{K_{sr,j}}(r_1, r_3, M_2)$為密文，分析與 **Auth** 演算法之說明相同。

5. 金鑰非同步攻擊：所有權轉移協定為一種金鑰更新方式，因此協定應能抵抗金鑰非同步攻擊，以避免惡意人士試圖以金鑰非同步攻擊阻斷服務。
   - **OT** 演算法：協定中新擁有者傳送 $M_3= H(Knew_{st,i} \| r_2)$進行金鑰更新，若攻擊者試圖篡改或攔截 M_3 以進行金鑰非同步攻擊。在不知道 $K_{st,i}$、α_i 及 $Knew_{st,i}$ 之情況下，攻擊者若要計算出有效之 $M_3= H(Knew_{st,i} \| r_2)$，將面臨單向雜湊函數之困難度假設。
6. 金鑰鑑別：用於確定更新後之金鑰是否正確。
   - **OT** 演算法：協定中新擁有者傳送訊息鑑別碼 $M_3= H(Knew_{st,i} \| r_2)$進行金鑰更新，標籤經由驗證 M_3 確認更新金鑰之正確性；新擁有者則檢查標籤回復之 M_5 確認更新金鑰之正確性。
7. 假冒標籤：若攻擊者試圖假冒標籤，需要具備欺騙讀取器與擁有者的能力。**Auth** 演算法、**OT** 演算法與 **Audit** 演算法中，假冒標籤須有能力能夠計算 $M_1=H(K_{st,i} \| r_1)$。在不知道 $K_{st,i}$ 之情況下，攻擊者欲計算出有效之 M_1，將面臨單向雜湊函數之困難度假設。

8. 假冒讀取器：若攻擊者試圖假冒讀取器，需要具備欺騙標籤與擁有者的能力。
   - **Auth** 演算法：欲假冒讀取器需有能力解密 M_3 取得 M_2。然而，攻擊者無持有合法金鑰，欲解密取得 M_2，將面臨加密演算法之計算困難度。
   - **OT** 演算法：欲假冒讀取器需有能力解密 M_4，分析與上述說明相同。
   - **Audit** 演算法：欲假冒讀取器需有能力解密 M_3，分析與上述說明相同。
9. 假冒擁有者：若攻擊者試圖假冒擁有者，需要具備欺騙標籤與讀取器的能力。
   - **Auth** 演算法：若欲假冒擁有者需有能力加密 M_2。然而，攻擊者無持有合法金鑰，欲計算有效之密文，將面臨加密演算法之計算困難度。
   - **OT** 演算法：若欲假冒擁有者需有能力加密 M_4，分析與上述說明相同。
   - **Audit** 演算法：若欲假冒擁有者需有能力加密 M_2，分析與上述說明相同。
10. 向前安全性：當惡意人士獲得現在標籤金鑰之後，若試圖破解過往的金鑰或資訊，將面臨單向雜湊函數之困難度假設。
11. 標籤舊擁有者竊取標籤金鑰：若舊擁有者試圖從持有之標籤舊金鑰計算標籤新金鑰 $Knew_{st,i}$。若舊擁有者欲從 $M_3= H(Knew_{st,i}\| r_2)$與 $M_5= H(Knew_{st,i}\| r_1)$反推取得 $Knew_{st,i}$，將面臨單向雜湊函數之困難度假設。

4.2 第二協定之安全性分析

本節對於擁有者群體導向鑑別服務、新擁有者群體導向所有權轉移服務及 TA 群體導向稽核服務進行安全性分析。此外，將分析標籤群體鑑別，標籤群體鑑別是探討標籤群體組成的合法性。其中，向前安全性與標籤舊擁有者竊取標籤金鑰之分析說明與 4.1 節相同。

1. 雙向鑑別：
   - **G_auth** 演算法：擁有者、讀取器及標籤三者之間透過詰問問答鑑別身分。擁有者經由驗證 $msg_{1,i}=H(K_{st,i}\|r_1)$與 r_3，鑑別讀取器與標籤身分。讀取器藉由解密 M_1 並檢驗其中的 r_1 是否與所傳送的 r_1 相等，鑑別擁有者與標籤身分。標籤經由驗證 $msg_{3,i}$ 是否等於 $H(K_{st,i}\|r_2)$，鑑別擁有者與讀取器身分。
   - **G_OT** 演算法：新擁有者、讀取器及標籤三者之間經由詰問問答與 TA 之協助以鑑別身分。由於新擁有者一開始並無與標籤分享金鑰，因此新擁有者需 TA 通知 $msg_{1,i}=H(K_{st,i}\|r_1)$之檢驗結果，並且 TA 加密標籤新金鑰後傳送給新擁有者。新擁有者解密 M_2 並檢驗其中之 r_3 是否等於所傳送的 r_3，鑑別 TA 與標籤身分。讀取器經由解密 M_3 並檢驗其中之 r_1 是否等於所傳送的 r_1，鑑別 TA 與標籤身分。標籤則驗證 $msg_{2,i}$ 是否等於 $H(Knew_{st,i}\|r_2)$，鑑別新擁有者與讀取器身分。
   - **G_audit** 演算法：TA、讀取器與標籤三者之間透過詰問問答鑑別身分。TA 透過檢驗 $msg_{1,i}=H(K_{st,i}\|r_1)$與 r_3，鑑別讀取器與標籤身分。讀取器藉由解密

M_1 並檢驗其中之 r_1 是否與適才所傳送的 r_1 相等，鑑別 TA 與標籤身分。標籤驗證 $msg_{3,i}$ 是否等於 $H(K_{st,i}||r_2)$，鑑別 TA 與讀取器身分。

2. 標籤群體鑑別：若要破壞標籤群體之組成合法性，有三種作法，包括置換、移除及新增。若攻擊者試圖置換除標籤群體中之成員，需具有假冒標籤之能力。由上述分析可知，將面臨單向雜湊函數之困難度假設。若試圖移除或新增標籤群體中之成員，則計算出之標籤群體識別碼會與其他成員所計算的識別碼不同，將被其他成員發現。
3. 機密性與完整性：說明與 4.1 節相同。
   - **G_OT** 演算法：TA 產生標籤新金鑰後，加密標籤新金鑰與新擁有者所送來的 r_3，並傳送至新擁有者。新擁有者接收訊息後，解密訊息並檢驗 r_3 是否等於所傳送之 r_3。若等於，則能相信標籤新金鑰並未被篡改。
4. 重送攻擊：說明與 4.1 節相同。
   - **G_auth** 演算法：鑑別協定中傳送的訊息有 $msg_{1,i}$、$msg_{2,i}$ 與 M_1，訊息內容包含亂數 r_1、r_2 與 r_3，可經由驗證亂數，以確認是否遭受重送攻擊。
   - 新擁有者所有權轉移服務：所有權轉移協定中傳送的訊息有 $msg_{1,i}$、$msg_{2,i}$、$msg_{3,i}$、M_1 與 M_2，訊息內容包含亂數 r_1、r_2 與 r_3，分析與上述說明相同。
   - **G_audit** 演算法：稽核協定中傳送的訊息有 $msg_{1,i}$、$msg_{2,i}$ 與 M_1，訊息內容包含亂數 r_1、r_2 與 r_3，分析與上述說明相同。
5. 中間人攻擊：說明與 4.1 節相同。
   - **G_auth** 演算法：鑑別協定中傳送之訊息中，$msg_{1,i}= H(K_{st,i}|| r_1)$與 $msg_{2,i}= H(K_{st,i}|| r_2)$為訊息鑑別碼，$M_1= E_{K_{sr,j}}(r_1, r_3, msg_{2,i})$為密文，其中皆需要使用共享金鑰計算。若攻擊者欲執行中間者攻擊，將面臨單向雜湊函數之困難度假設與加密演算法的計算困難度。即，確保訊息之來源性。
   - **G_OT** 演算法：所有權轉移協定中傳送之訊息，$msg_{1,i}= H(K_{st,i}|| r_1)$、$msg_{2,i}= H(Knew_{st,i}|| r_2)$及 $msg_{3,i}= H(Knew_{st,i}|| r_1)$為訊息鑑別碼，$M_1= E_{K_{sub,k+1}}(Knew_{st,i}, r_3)$ 與 $M_2= E_{K_{sr,j+1}}(msg_{2,i}, r_1)$ 為密文，分析與上述說明相同。
   - **G_audit** 演算法：稽核協定中傳送之訊息中，$msg_{1,i}= H(K_{st,i}|| r_1)$、$msg_{2,i}= H(K_{st,i}|| r_2)$為訊息鑑別碼，$M_1= E_{K_{sr,j}}(r_1 , r_3, msg_{2,i})$為密文，分析與上述說明相同。
6. 金鑰非同步攻擊：說明與 4.1 節相同。
   - **G_OT** 演算法：協定中新擁有者傳送 $msg_{2,i}= H(Knew_{st,i}|| r_2)$進行金鑰更新，若攻擊者試圖篡改或攔截 $msg_{2,i}$ 進行金鑰非同步攻擊。在不知道 $K_{st,i}$、α_i 及 $Knew_{st,i}$ 之情況下，攻擊者若要計算出有效之 $msg_{2,i}= H(Knew_{st,i}|| r_2)$，將

面臨單向雜湊函數之困難度假設。

7. 金鑰鑑別：說明與 4.1 節相同。
   - **G_OT** 演算法：協定中新擁有者傳送 $msg_{2,i}$ 進行金鑰更新，標籤經由驗證 $msg_{2,i}$ 確認更新金鑰之正確性；新擁有者則檢查標籤回復的 $msg_{3,i}$ 驗證更新金鑰之正確性。
8. 假冒標籤：說明與 4.1 節相同。**G_auth** 演算法、**G_OT** 演算法與 **G_audit** 演算法服務中，假冒標籤須有能力能夠計算 $msg_{1,i}=H(K_{st,i}||r_1)$。在不知道 $K_{st,i}$ 之情況下，攻擊者欲計算出有效之 $msg_{1,i}$，將面臨單向雜湊函數之困難度假設。
9. 假冒讀取器：說明與 4.1 節相同。
   - **G_auth** 演算法：欲假冒讀取器需有能力解密 M_1 取得 $msg_{2,i}$。然而，攻擊者無持有合法金鑰，欲解密取得 $msg_{2,i}$，將面臨加密演算法之計算困難度。
   - **G_OT** 演算法：若欲假冒讀取器需有能力解密 M_2 取得 $msg_{2,i}$，分析與上述相同。
   - **G_audit**：若欲假冒讀取器需有能力解密 M_1 取得 $msg_{2,i}$，分析與上述相同。
10. 假冒擁有者：說明與 4.1 節相同。
    - **G_auth** 演算法：若欲假冒擁有者需有能力加密 $msg_{2,i}$。然而，攻擊者無持有合法金鑰，欲計算有效之密文，將面臨加密演算法之計算困難度。
    - **G_OT** 演算法：若欲假冒擁有者需有能力加密 $msg_{2,i}$，分析與上述相同。
    - **G_audit**：若欲假冒擁有者需有能力加密 $msg_{2,i}$，分析與上述相同。

伍、 安全分析

本論文基於企業供應鏈環境的弱點，針對可能遭受的威脅與攻擊，設計 RFID 所有權轉移協定，並且兼顧企業之可用性與低成本標籤需求。其中，供應鏈內連線 RFID 所有權轉移協定為企業營運之基本需求，此協定能在連線環境下，進行鑑別、所有權轉移並且進行稽核；供應鏈內群體導向連線 RFID 所有權轉移協定為增加企業營運效能，此協定能夠在連線環境下，進行群體導向鑑別、群體導向所有權轉移並且進行群體導向稽核。透過由標籤群體為運作單位的方式，能夠讓企業進行大規模的供應鏈運作。

[誌謝]

本研究部份接受科技部研究計畫經費補助，計畫編號：103-2221-E-146-005-MY2 、103-2221-E-011-090-MY2、104-2119-M-011-003 及 105-2923-E-182 -001 -MY3；長庚醫院計畫經費補助：CMRPG5D0182。

參考文獻

[1] A. Juels, "RFID Security and Privacy: A Research Survey", *IEEE Journal on Selected Areas in Communications*, Vol. 24, No. 2, 2006, pp. 381-394.

[2] M. Ohkubo, K. Suzuhi, S. Kinoshita, "RFID privacy issues and technical challenges", *Communications of the ACM*, Vol. 48, No. 9, 2005, pp.66-71.

[3] K. Takaragi, M. Usami, R. Imura, R. Itsuki, and T. Satoh, "An Ultra Small Individual Recognition Security Chip", *IEEE Micro*, Vol. 21, No. 6, 2001, pp.42-49.

[4] N. C. Wu, M. A. Nystrom, T. R. Lin, H. C. Yu, "Challenges to Global RFID Adoption", *Technovation*, Vol. 26, No. 12, 2006, pp.1317-1323.

[作者簡介]

許建隆博士分別於 1997 年與 2002 年取得臺灣科技大學資管系碩士與博士學位，自 2011 年 8 月起擔任長庚大學資管系教授，並自 2013 年 8 月起兼任系主任一職，亦兼任長庚大學 RFID 物流與供應鏈應用學程與資訊與醫療安全學程之召集人、中華民國資訊安全學會之理事與會員委員會主任委員。專長領域包括智慧家庭、行動商務、電腦與通訊安全、資訊安全、應用密碼學、健康照護、數位版權管理、自動辨識技術、數位鑑識。

詹宜勳先生分別於 2009 年與 2011 取得長榮大學資訊管理學系學士與長庚大學資訊管理學系碩士學位，其研究興趣為 RFID、訊安全、密碼學。

蔡國裕博士分別於 2001 年與 2009 年取得臺灣科技大學資訊管理系碩士與博士學位。2009 年 9 月，蔡博士於臺灣科大資通安全研究與教學中心服研發替代役。2012 年 8 月，進入華夏科技大學資管系擔任助理教授。研究領域包括密碼學、資訊安全、物聯網應用安全及雲端運算應用安全等。