隱私保護 NFC 留言系統
A Privacy Preserving NFC Guestbook System

簡正欽(Zheng-Qin Jian)　黃郁誠(Yu-Chung Huang)　江振瑞(Jehn-Ruey Jiang)

國立中央大學
資訊工程研究所

摘要 — 近場通訊(Near Field Communication, NFC)由無線射頻辨識(Radio Frequency Identification, RFID)技術演變而來，在 13.56MHz 頻率運行於 20 公分傳輸距離內。它是短距無線通訊技術最受注目的典型範例之一，經常被整合到智慧型手機中，具有各式各樣的應用。NFC 設備具有卡片模擬模式(card emulation mode)，在這個模式下 NFC 設備就相當於一張 RFID 標籤。NFC 設備另外具有讀取器模式(reader/writer mode)可以讀取 NFC 標籤中資料或將資料寫入 NFC 標籤中。在本論文中，我們提出一個使用 NFC 技術的電子化留言系統，並且達到使用者的隱私性(privacy)、不可否認(non-repudiation)及完整性(integrity)。在此系統中，使用者的 NFC 設備必須先向系統中的伺服器進行註冊，取得一對公鑰與私鑰以及憑證。我們採用非對稱式加密機制進行留言加密與數位簽章的嵌入，最後 NFC 設備使用讀取器模式將留言寫入 NFC 標籤。我們並於 Android 平台實作 NFC 留言系統以驗證系統的實用性。

關鍵詞：近場通訊、NFC 標籤、非對稱式加密、數位簽章、無線射頻辨識、Android 平台

1. 簡介

近年來，由於網路及通訊技術的飛速發展，無線通訊在我們的生活當中扮演著很重要的角色，而短距無線通訊(short-range wireless communication)技術成為關注的焦點。短距無線通訊技術包括藍芽、IEEE 802.11(Wi-Fi)、IEEE 802.15.4(ZigBee)、超寬頻(Ultra WideBand, UWB)、近場通訊(Near Field Communication, NFC)等標準，或基於傳輸速度、距離、耗電量的特殊要求，它們都有其立足的特點，其中 NFC 是最近最熱門最受到注目的技術。

NFC[1]自無線射頻辨識(Radio Frequency Identification, RFID)技術演變而來，由 Philips、NOKIA 和 Sony 主推，在 13.56MHz 頻率運行於 20 公分傳輸距離內，其傳輸速度有 106 kbps、212 kbps 或者 424 kbps 三種。NFC 設備具有卡片模擬模式(card emulation mode)，在這個模式下 NFC 設備就相當於一張 RFID 標籤(tag)。NFC 設備另外具有讀取器模式(reader/writer mode)可以讀取 NFC 標籤中資料或將資料寫入 NFC 標籤中，也具有點對點模式(P2P mode)，可以與另一個 NFC 設備直接交換資料。

雖然以往只有高階手機才具備 NFC 功能，但是近來 NFC 已是一般智慧型手機的必備功能。根據國際市場研究機構 IHS 的資料表示[2]，預計從 2013 年到 2018 年底，全球 NFC 手機出貨量將增加 325%，預估將達到 12 億隻，這代表著在未來 NFC 手機勢必成為智慧型手機的主流。

有鑑於 NFC 應用的普及，本論文預計進行 NFC 電子化留言系統(guestbook system)的研究。傳統上，大多數人喜愛採用便利貼(post-it note)[3]當作留言的工具，然而在極重視個人隱私的現代社會中，如果想利用紙張留言寫上幾句較為隱私的話，無疑是令人擔心的。透過加密技術，我們讓 NFC 留言系統可以保有隱私性(privacy)、不可否認(non-repudiation)及完整性(integrity)等特性。

隱私性為防止未被授權者發現明文;完整性為確定明文沒有被有意或無意的更改;不可否認性為發送方在事後不可否認其傳送過的資訊。我們運用非對稱式加密機制來達到上述三項特性，此機制需要一對金鑰，一是個私人金鑰，另一個則是公開金鑰。這兩個金鑰是相關的，用某用戶的一個金鑰加密後所得的資訊，只能用該用戶的另一個金鑰才能解密。如果知道了其中一個金鑰，並不能計算出另外一個金鑰。因此如果公開了一對金鑰中的一個，並不會危害到另外一個金鑰的秘密性質。

本究提出一個有別於以往的 NFC 應用 — 一個具有隱私保護的留言系統，利用價格便宜的 NFC 標籤當作儲存訊息的載具，並以 NFC 手機進行留言的讀取與寫入，再運用非對稱式加密機制來確保留言系統的隱私性、完整性與不可否認性。由於並非所有的環境都提供上網服務，我們利用 NFC 點對點的通訊模式,研究與伺服器無連線狀況下，達成不可否認性的驗證。

我們於 Android 平台實作所提的 NFC 留言系統，可依照使用者需求提供訊息加密或數位簽章的功能，也可以選擇是否驗證簽章。並且保證只有標籤持有人可以正確讀取留言，留言內容若遭惡意者修改也可以檢測出來。

本系統可應用於多種不同環境中，可以當作家庭、教室留言板，也可以當作賣場的意見留言箱以保證投訴者的隱私性，更可以在辦公室中與同事之間進行隱私留言。如果使用者是開門做生意的，在店門口裝設推銷員留言箱可以避免推銷員於營業時間的拜訪。我們期望本系統讓 NFC 的用服務應用更加多元化。

論文其他部份的內容如下所述。第二節介紹近場通訊(Near Field Communication, NFC)、非對稱式加密等相關知識。第三節詳細描述系統架構與流程，並進行系統安全性分析。第四節為系統實作，說明我們如何在 Android 平台上

實作，展示實作成果與測試系統隱私性與完整性。第五節則為結論與未來展望。

2. 相關研究

2.1 NFC 概述

近場通訊（Near Field Communication，NFC），又稱近距離無線通訊，是一種短距的高頻無線通訊技術，允許電子裝置之間進行非接觸式點對點資料傳輸。這個技術由RFID 演變而來，目前已通過成為 ISO/IEC IS 18092 國際標準、EMCA-340 標準與 ETSI TS 102 190 標準。NFC 採用主動和被動兩種讀取模式。被動模式不需要電池，但是缺乏獨立發射訊號的能力；而主動則反之。

NFC 近場通信的最大範圍大約是 20cm，典型的使用距離是 4cm 至 5cm，這對安全性很有好處。近場通信範圍距離天線約一個波長之內(波長/(2π))，其中電磁場不斷交換能量並沿著信號路徑相互恢複。根據麥克斯韋方程(Maxwell's equation, [4])，近場通訊中磁場占主導地位，磁場強度隨傳遞距離(d)增加而衰減，其衰減因子為 $1/d^3$。

NFC設備可以區分為主動式和被動式兩種類型。主動式設備具有電力來源，可以擔任通訊發起設備(initiator)，以其他設備為標的(target)，透過產生無線射頻場域，向其他發出建立連線請求。被動式設備則可以接受連線請求，擔任標的設備。如表1所示，主動式設備可以扮演發起設備或目標設備；被動式的設備因為本身無提供電力來源，無法扮演發起設備角色，僅能扮演標的設備。

表 1. 主動/被動式設備與發起設備(Initiator)/目標設備(Target)的組合

Device Role	Active Device	Passive Device
Initiator	Possible	Not Possible
Target	Possible	Possible

NFC設備的工作模式分為以下三種:

1. 卡模擬模式（Card emulation mode）：此模式其實就是相當於一張RFID標籤，屬於被動模式，可以替代傳統的IC卡，包括門禁管制卡、車票卡與門票卡等等。此種模式有一個優點，那就是標籤通過非接觸讀取器的無線射頻場域來供電，即便是NFC設備沒有電力來源也可以在此模式下工作。
2. 點對點模式（P2P mode）：這個模式和紅外線傳輸模式差不多，可用於資料交換，只是傳輸距離較短，傳輸建立速度較快，傳輸速度也快些，功耗低。將兩個具備NFC點對點模式傳輸功能的設備互相靠近，即能以NFC資料交換格式(NFC Data Exchange Format, NDEF)進行資料點對點傳輸，如下載音樂、交換圖片等。
3. 讀取器讀寫模式（Reader/Writer mode）：在此模式下設備可作為非接觸讀取器使用，比如從海報或者展覽資訊電子標籤上讀取相關資訊，屬於主動模式。

一個NFC設備可以具有上述工作模式中的全部或部分。常見的NFC設備包括NFC手機、NFC讀取器與NFC標籤。如表2，NFC手機與NFC讀取器屬於主動設備，可以擔任發起設備(Initiator)或目標設備(Target)，而NFC標籤是一種被動式設備，本身不含電池，不提供電力來源，需要透過主動式設備提供的無線射頻場域取得驅動的電力，因此只能擔任目標設備。

表 2. 主動/被動式設備與發起設備(Initiator)/目標設備(Target) 的組合

Initiator	Target
NFC 手機	NFC 標籤
NFC 手機	NFC 手機
NFC 讀取器	NFC 手機

2.2 非對稱式加密(Asymmetric Cryptography)

在本小節我們說明非對稱式加密機制(asymmetric cryptography)，我們首先說明較簡單的對稱式加密機制(symmetric cryptography)。對稱式加密機制是基於接收端與發送端共享相同的秘密金鑰(secret key)，發送端利用秘密金鑰將明文(plaintext)加密得到密文(ciphertext)，接收端同樣利用相同的秘密金鑰將密文解密得到明文。然而，對稱式加密機制的必要條件在於發送端與接受端必須同時具共同的秘密金鑰，如何安全的將秘密金鑰傳送給發送端與接受端是一個具挑戰性的問題，這稱為金鑰發送(key distribution)問題[5]。

Diffie 與 Hellman 於 1976 年提出公鑰加密機制(public-key cryptography)[6]，也稱非對稱式加密機制(asymmetric cryptography)，此機制可用來解決對稱式加密機制的金鑰發送問題。非對稱式加密機制的使用者具有兩把金鑰，分別為公開金鑰(public key)與私密金鑰(private key)。公開金鑰(公鑰)是公開給所有其他的使用者知道，而私密金鑰(私鑰)則只有使用者自己知道。發送端與接收端不需要共享一個金鑰，而且使用者無法由公鑰推算出私鑰。一個訊息如果使用某一個使用者的公鑰加密，就必需使用此使用者的私鑰解密;相反的，若用私鑰加密就必須用公鑰解密。

如圖1，使用者A想要將明文訊息M傳送給目的使用者B，B產生一對金鑰，公開金鑰KU_b與私密金鑰KR_b。KR_b只有B自己知道，KU_b則是公開的，因此A可以容易地取得KU_b。A利用KU_b對訊息M加密做加密計算密文C，並且將密文C傳送給B。

$$C = E_{KR_a}(M) \tag{1}$$

使用者B收到密文C後則利用KR_b解密，取得明文訊息M。由於KR_b屬於B個人保管，只有B才可解開密文C得到明文M。

$$M = D_{KR_b}(C) \quad (2)$$

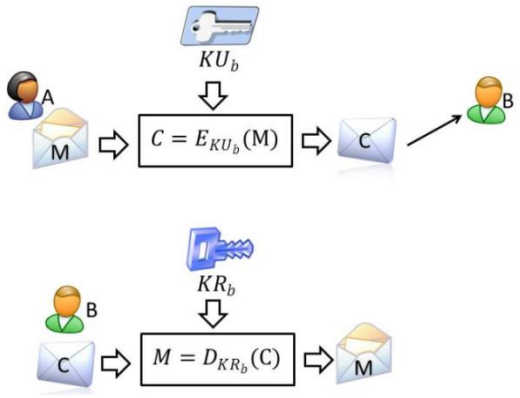

圖 1. 非對稱式加密與解密

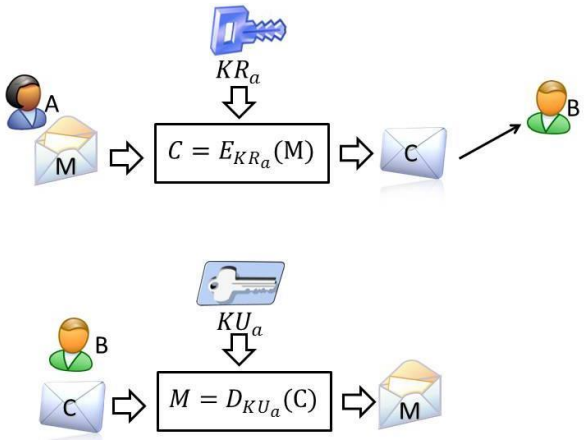

圖 2. 非對稱式數位簽章與驗證簽章

非對稱式加密機制還可以用以執行數位簽章(digital signature)動作。如圖2，使用者A想將對明文訊息M傳送給目的使用者B，並想藉由數位簽章證明這個訊息無法否認地確實由A所發出。作法為A產生一對金鑰，公開金鑰是KU_a，私密金鑰是KR_a。A利用KR_a對訊息M做加密計算密文C，並且將C傳送給B。

$$C = E_{KR_a}(M) \quad (3)$$

使用者B收到密文C後則利用KU_a解密，取得明文訊息M。因為訊息M是使用A的私密金鑰來加密且此金鑰只有A個人擁有，因此可以視為A的數位簽章。由於只能使用A的公開金鑰解密，藉此可驗證訊息的來源與完整性。

$$M = D_{KR_b}(C) \quad (4)$$

RSA (Rivest-Shamir-Adleman)演算法[7, 8]是當前最著名，應用最廣泛的非對稱式加密機制。RSA 是在 1978 年由美國麻省理工學院三位學者 Rivest、Shamir 及 Adleman 研究發展出來的，它是一個基於數論的非對稱式加密碼機制，它的安全性是基於大整數因數分解(integer factorization)的困難性。大整數的因數分解問題是數學上的著名難題，即使採用目前速度最快的計算機，都需要相當長的時間(數年或數百年)才可以將一個大整數進行分數分解。至今仍然沒有有效的方法可以迅速地解決大整數因數分解問題，RSA 演算法的安全性因此得以確保。

3. NFC 留言系統

本節介紹我們所設計的 NFC 留言系統。我們先說明 NFC 留言系統的系統架構，然後說明系統流程，描述如何利用非對稱式加密機制讓來源端進行加密與數位簽章(digital signature)運算，並且將訊息透過 NFC 標籤傳遞給目的端，以及目的端如何進行解密與數位簽章運算。最後我們進行安全性分析，說明系統可以達成隱私性、不可否認性與完整性。

3.1 系統架構

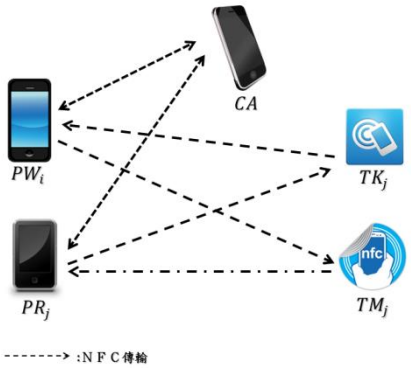

圖 3. NFC 留言系統之系統架構圖

圖 3 為 NFC 留言系統的系統架構圖，以下介紹架構圖中用到的記號:

1. PW_i: 編號為 i 的留言寫入者。
2. PR_j: 編號為 j 的留言讀取者。
3. $Cert_i$: 編號為 i 的留言讀取者憑證。
4. CA: 憑證管理機構(Certificate Authority)，負責公鑰與私鑰的產生、註冊、管理與與憑證(certificate)發放。(圖 3 中採用手機圖案來表示 CA 是因為本系統於實作時採用 Android APP 於手機上開發憑證管理機構並且 CA 在發行金鑰時是採用 NFC 方式傳輸。
5. TK_j: 編號為 j 的金鑰標籤，用於存放編號為 j 的留言讀取者的公鑰及其憑證。
6. TM_j: 編號為 j 的留言標籤，用於存放編號為 j 的留言讀取者的留言密文。

3.2 系統流程

系統流程分為五大部份: 註冊、訊息加密、訊息解密、數位簽章、驗證簽章，以下我們分別說明流程的各個部份。

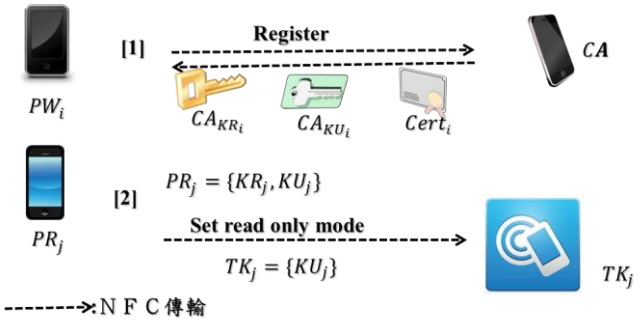

圖 4. 註冊流程

圖 4 為註冊流程，不論是訊息讀取者或訊息寫入者都可以向 CA 註冊以取得公鑰、私鑰與憑證，以做為數位簽章與身份驗證(authentication)之用。但若不執行簽章或身份驗證動作，則不論是讀取者或寫入者都不須向 CA 註冊。以下我們以訊息寫入者選擇向 CA 註冊，而訊息讀取者選擇不向 CA 註冊來說明註冊步驟:

(1) 訊息寫入者 PW_i 以使用者登入的名字為參數，透過 NFC 傳輸向 CA 進行註冊，CA 以非對稱式加密機制計算一對公鑰 CA_{KU_i} 與私鑰 CA_{KR_i} 及憑證 $Cert_i$，並以 CA 本身的私鑰加密後發給 PW_i(我們假設所有使用者都已經擁有或很容易取得 CA 的公鑰)，以進行訊息加解密與數位簽章使用。

(2) 訊息讀取者 PR_j 以使用者登入的名字為參數，利用非對稱式加密機制計算一對公鑰與私鑰 (KU_j, KR_j)，以做為訊息加解密的處理。PR_j 透過 NFC 傳輸，將公鑰 KU_j 寫入到 TK_j 並且將 TK_j 設定為唯讀模式，以防止公鑰遭到惡意者修改。

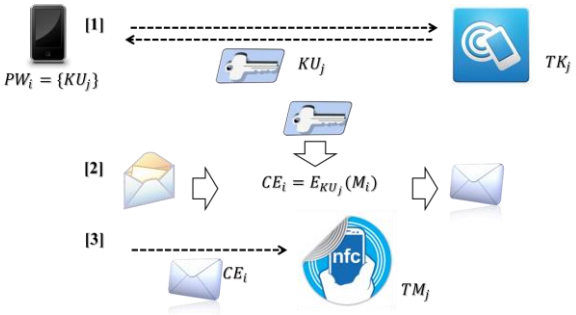

圖 5. 訊息加密流程

圖 5 為訊息(留言)加密流程，PW_i 欲將留言寫入到 NFC 標籤前，必須先進行訊息加密的計算。以下新增幾個記號，以方便說明流程之用，其中也有可以用於訊息簽章相關的記號:

1. $E_X(m)$：使用 X 的公鑰，對訊息 m 做加密。
2. $D_X(m)$：使用 X 的公鑰，對訊息 m 做解密。
3. $S_X(m)$：使用 X 的私鑰對訊息 m 做簽章。
4. $V_X(m)$：使用 X 的公鑰去驗證訊息 m 。
5. $Cert_i$：訊息寫入者 PW_i 的憑證。
6. M_i: PW_i 欲寫入的明文，亦即留言的原始訊息。
7. CE_i: M_i 的密文。
8. CS_i: M_i 的簽章。

圖 5 中的訊息加密計算流程說明如下:

(1) PW_i 從 TK_j 取得 KU_j。

(2) PW_i 利用 KU_j 對 M_i 作加密計算，得到 CE_i。

(3) PW_i 透過 NFC 傳輸，將 CE_i 寫入到 TM_j。

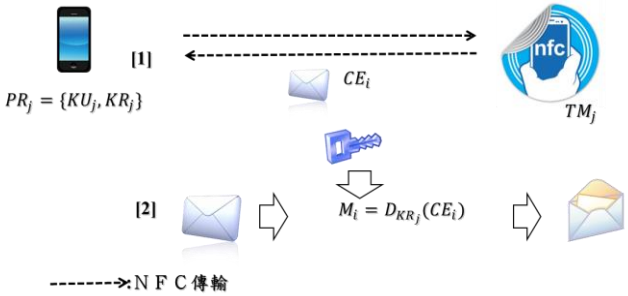

圖 6. 訊息解密流程

圖 6 為訊息解密流程，首先留言讀取者 PR_j 從 TM_j 取得訊息密文 CE_i，再利用 KR_j 對 CE_i 進行解密計算，還原明文 M_i，完成讀取留言的步驟。由於 M_i 是利用 PR_j 的 KU_j 作加密計算所得，因此只有利用 KR_j 可進行解密計算。因為只有 PR_j 知道 KU_j，因此可以確保只有 PR_j 可以解密取得明文。

以下說明圖 6 的訊息解密流程:

(1) PR_j 從 TM_j 取得 CE_i。

(2) PR_j 利用 KR_j 對 CE_i 作解密計算，得到明文 (M_i)。

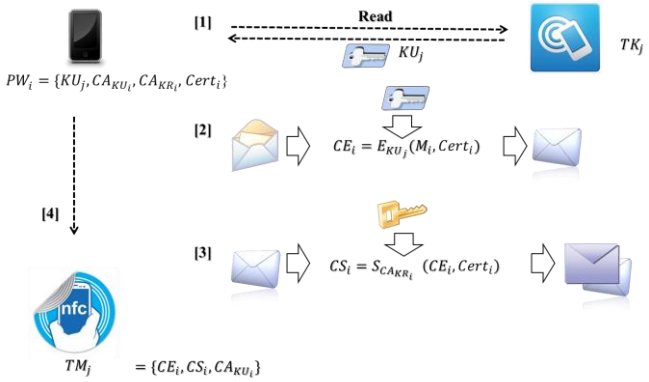

圖 7. 訊息簽章流程

圖 7 為留言寫入者加入訊息簽章流程，說明如下:

(1) PW_i 從 TK_j 取得 KU_j。

(2) PW_i 利用 KU_j 對明文 M_i 及 $Cert_i$ 作加密計算，得到 CE_i

(3) PW_i 利用 CA_{KR_i} 對 CE_i 及 $Cert_i$ 作簽章運算，得到簽章 CS_i。

(4) 透過 *NFC* 傳輸，將 CE_i 、CS_i 、CA_{KU_i} 寫入到 TM_j，完成訊息簽章流程。

圖 8 為訊息驗證簽章流程，包含留言讀取者 PR_j 從標籤 TM_j 讀取訊息後，將密文進行解密與驗證簽章的計算:

(1) PR_j 從 TM_j 取得 $(CE_i$ 、CS_i 、$CA_{KU_i})$。

(2) PR_j 利用 KR_j 對 CE_i 作解密計算，得到 M_i 及 $Cert_i$。

(3) PR_j 利用 CA_{KU_i} 對 CS_i 及 $Cert_i$ 作驗證簽章運算。

留言讀取者PR_j在上述的步驟(2)中取得明文訊息M_i，如果留言讀取者認為M_i無被竄改的可能性，可以無需進行步驟(3)。如此可以增加系統的便利性。

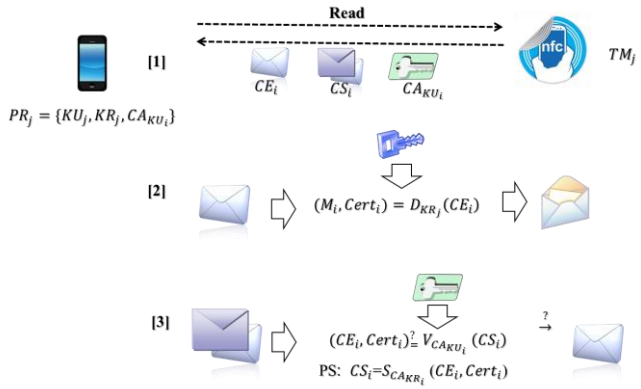

圖 8. 訊息驗證簽章流程

3.3 安全分析

以下我們分析系統的不可否認性、隱私性與完整性。我們首先分析不可否認性。不可否認性的定義為發送方不得否認其傳送過的資訊，因此簽章採用金鑰必須具備以下兩個特性:

1. 公開與私密金鑰必須為第三方機構所發行。
2. 簽章使用金鑰必須為發送方所私人保管。

本系統簽章運算式如下所列:

$$CS_i = S_{CA_{KR_i}}(CE_i, Cert_i) \quad (5)$$

本系統驗證簽章運算式如下所列:

$$CE_i \stackrel{?}{=} V_{CA_{KU_i}}(CS_i, Cert_i) \quad (6)$$

CA_{KU_i}、CA_{KR_i}及$Cert_i$分別為 CA 所發行給PW_i的公開金鑰、私密金鑰及物件憑證，PW_i使用此私密金鑰及憑證進行簽章，驗證方則採用此公開金鑰進行驗證簽章，若驗證成功，則此簽章必為PW_i所以，因此能達到不可否認性。

以下我們分析隱私性與完整性進行分析，此兩特性定義如下:

1. 隱私性:防止未被授權者發現明文。
2. 完整性:驗證明文沒有遭到竄改。

在 2-1 節曾提到 NFC 典型的使用距離是 4cm 至 5cm，而且近場的訊號強度衰減非常快，這代表在 NFC 標籤與 NFC 讀取器中間進行攔截或阻擋訊號很難不被發現。但是惡意者仍然可以直接對 NFC 標籤的內容進行修改，因此確保完整性的確有其必要。

本節使用記號沿用 3-2 節，表 3 為本節新增的記號表。在圖 9 中，未經授權留言讀取者PR_{j+1}欲讀取標籤內的訊息。

(1) PR_{j+1}讀取TM_j內容，得到CE_i、CS_i、CA_{KU_i}，其中CE_i為密文。

(2) 因為CE_i是採用KU_j進行加密，只有KR_j可以進行解密，PR_{j+1}無法進行解密，因此保證隱私性。

表 3. 安全分析之記號表

PR_{j+1}	未經授權留言讀取者
KU_{j+1} 、KR_{j+1}	PR_{j+1}利用非對稱加密自行計算產生的公開金鑰與私密金鑰。
PW_{i+1}	修改明文的惡意者。
KU_{i+1} 、KR_{i+1}	PW_{i+1}利用非對稱加密自行計算產生的公開金鑰與私密金鑰。
M_{i+1}	遭修改的明文。
CE_{i+1}	遭修改的密文。

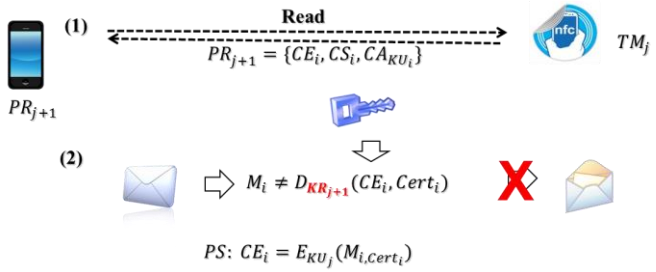

圖 9. 隱私性分析

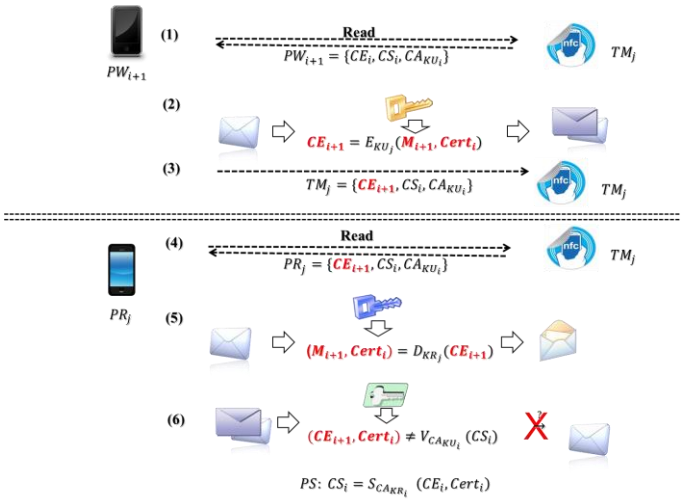

圖 10. 完整性分析

在圖 10 中，惡意者PW_{i+1}企圖將明文修改為M_{i+1}。說明如下:

(1) PW_{i+1}讀取TM_j內容，得到CE_i、CS_i、CA_{KU_i}，其中CE_i為密文。

(2) KU_j為已知的公開金鑰(參考圖 7)，利用KU_j對$(M_{i+1}, Cert_i)$進行加密得到CE_{i+1}。

(3) 將修改後的密文與原本的簽章CS_i、憑證CA_{KU_i}重新寫入到 NFC 標籤。完成修改明文的動作。

(4) PR_j讀取TM_j內容，得到CE_{i+1}、CS_i、CA_{KU_i}，其中CE_{i+1}為修改後的密文。

(5) KR_j為PR_j的私密金鑰，利用此金鑰對CE_{i+1}進行解密，得到修改後明文M_{i+1}及$Cert_i$。PR_j認為此明文內容有異，恐遭修改，選擇執行驗證簽章步驟，則進行步驟(6)。

(6) 利用憑證KU_{CA_i}對CS_i進行驗證簽章運算，結果為驗證失敗。因為此簽章是針對CE_i所作的簽章，並非CE_{i+1}。因此明文遭修改後可以在驗證簽章同時檢查完整性。

在第三節中我們推導留言系統在讀取者、寫入者與兩組公鑰與私鑰之間的運算與訊息交換過程，並且確保本系統的隱私性、完整性與的不可否認性。下一節我們將介紹如何將本系統實作成 Android APP。

4. 系統實作

我們在 Android 平台上實做了本系統，主要利用兩個 Android SDK(android.nfc、android.security)用來存取 NFC 標籤與 RSA 金鑰、加密、簽章等的處理。

我們實作兩個 Android APP，分別命名為 NFCTagMsg 與 NFCTagMsg-CA。NFCTagMsg 可以讓使用者將留言訊息做加密或簽章嵌入後，寫入 NFC 標籤內，同樣也提供讀取留言所需的解密與簽章驗證的處裡，讀取留言或寫入留言可以在使用者設定介面上進行切換。基於市售常見的大容量 NFC 標籤為 1k bytes 以及一般性的辦公室留言並不會是機密訊息，因此 RSA 加密長度採用 64 bytes。未來可視需求調整。

NFCTagMsg-CA APP 是模擬憑證管理機構的 APP，具有簡易的憑證管理機構(Certificate Authority，*CA*)功能，提供註冊與發行公鑰與私鑰的功能。由於我們的核心價值著重於 NFC 標籤的留言、加密與簽章。CA 一般屬於商業行為居多，因此我們沒有架設憑證管理機構伺服器。安裝 NFCTagMsg-CA APP 的手機必須是專屬獨立的，並非使用者的手機，建議放置在單位組織的門口警衛或總機，由他們負責管理，不論是內部同仁或者外來訪客，都可以在此手機輸入註冊資料，NFCTagMsg-CA APP 會透過 NFC 介面將公鑰與私鑰傳輸到使用者手機內。在這節中我們稱這手機為憑證手機。

我們在使用者介面的設計上，力求簡單易懂，並且希望用產品開發的概念來展示實作成果及實作測試則提供測試數據以及測試留言系統的隱私性與完整性。

圖 2. 實測之 NFC 手機與標籤

圖 11 為實測時採用的手機與 NFC 標籤，左邊手機安裝 NFCTagMsg APP，中間位置手機則安裝 NFCTagMsg-CA APP，右方為兩張 NFC 標籤。

本節將提供實際測試數據、驗證加密與數位簽章的真實性。測試設備部分挑選舊款低階與新款高階手機且不同品牌來進行測試。NFC 標籤的規格主要訴求則是容量要大，與手機的相容性較好。另外還一併介紹寫入到標籤內的留言封包格式，最後則是測試加密與數位簽章的效果，表 4 與表 5 為測試手機與 NFC 標籤的規格。

表 4. 測試手機規格

	測試手機	
手機型號	Sony Xperia SP C5320	LG G2 D802
處裡器	雙核	四核
螢幕	4.6 吋	5.2 吋
記憶體	2G	16G
Android 版本	4.3	4.4

表 5. NFC 標籤的規格

	測試 NFC 標籤	
標籤型號	NXP Mifare Classic 1K	NXP Mifare DESFire 8K
RF Technology	ISO/IEC 14443 Type A	ISO/IEC 14443 Type A
Tag Type	N/A	NFC Forum Type 4
容量大小	1K bytes	8K bytes
實際可使用容量	703 bytes	7665 bytes
零售價格(參考)	NT$20	NT$90

首先我們挑選這兩款手機與兩款 NFC 標籤進行測試的理由。挑選的 NFC 標籤有兩款，Mifare Classic 1K 是比較舊款型的 NFC 標籤而且規格的 NFC 標籤 Type 並不在 NFC Forum 內，因此相容性不好，很多手機都不支援，例如 LG G2，優點則是價格便宜且台灣的網路拍賣很容易購買的到。Mifare DESFire 則具備容量大相容性好的優點但價格也較高，而且由於大容量的 NFC 標籤目前的應用少導致市場小，因此一般通路並無取得管道。

表 6 與表 7 為一則留言內所包含的封包格式，定義封包格式的目的有三:

1. 區分同一標籤內不同留言。
2. 辨別採用加密或簽章演算法。
3. 區分密文、簽章，以及數位金鑰。

針對 RSA 加密與簽章的長度的設計，考量到普遍 NFC 標籤的容量都不大，將 RSA 加密長度採用 64bytes。加密後的密文再拿來進行 RSA 數位簽章，因此簽章的長度必須大於 RSA 加密的 64bytes，所以採用 80bytes。

表 6. 加密留言封包格式

RSA 加密的 NFC 留言封包格式	定義	備註
yyy_____	該則留言的開頭	長度 3bytes
___A_____	該則留言使用哪種 RSA 演算法	1:RSA 加密 4:RSA 簽章 長度 1bytes
____xxx_____	分隔符號	長度 3bytes
_____0_____	無使用，填 0	長度 1bytes
_____xxx_____	分隔符號	長度 3bytes
_____E	RSA 加密後的密文	長度 64bytes
_____xxx	分隔符號	長度 3bytes
_____0	無使用，填 0	長度 1bytes
總長度為 79 bytes		

表 7. 留言封包格式

RSA 數位簽章的 NFC 留言封包格式	定義	備註
yyy_____	該則留言的開頭	長度 3bytes
___A_____	該則留言使用哪種 RSA 演算法	1:RSA 加密 4:RSA 簽章 長度 1bytes
____xxx_____	分隔符號	長度 3bytes
_____K_____	留言者的 CA 公鑰	長度 80bytes
_____xxx_____	分隔符號	長度 3bytes
_____E____	RSA 加密後的密文	長度 64bytes
_____xxx_	分隔符號	長度 3bytes
_____S	RSA 簽章後的密文	長度 80bytes
總長度為 237 bytes		

下面我們將進行一個測試，這個測試有三個角色，角色說明如表 8，角色於 NFCTagMsg APP 的設定如圖 12。

表 8. 實測之角色說明

Alice	標籤持有人，讀取留言角色。
Bob	留言者， Bob 欲留言給 Alice。
Candy	非標籤持有者、欲讀取 Alice 的標籤留言。

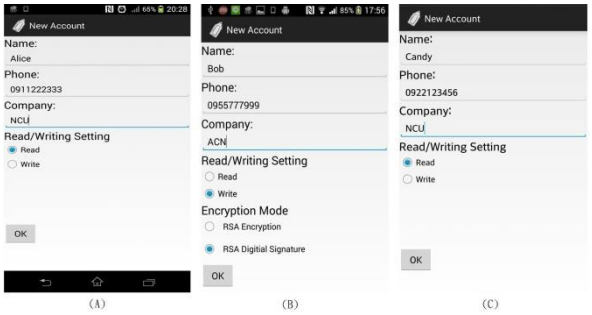

圖 3. 實測之設定畫面

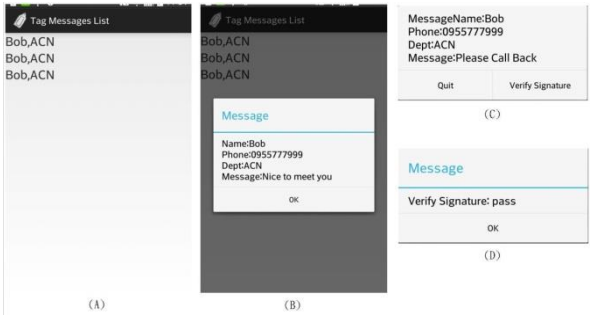

圖 4. 實測之讀取留言結果

如圖 13 中，Bob 先於 NFC 標籤內寫入了三則留言，而後標籤持有人 Alice 讀取標籤後顯示留言的畫面，圖中 A、B、C、D 四張圖的說明如表 9。

表 9. 實測之留言內容說明

圖 13-A	Alice 讀取標籤後，顯示有三則留言。
圖 13-B	第一則留言內容顯示:Nice to meet you。且該則留言並無數位簽章，所以驗證簽章功能。
圖 13-C	第二則留言內容顯示:Please Call Back。且該留言有提供驗證簽章功能。
圖 13-D	Alice 執行驗證簽章功能，顯示驗證成功。

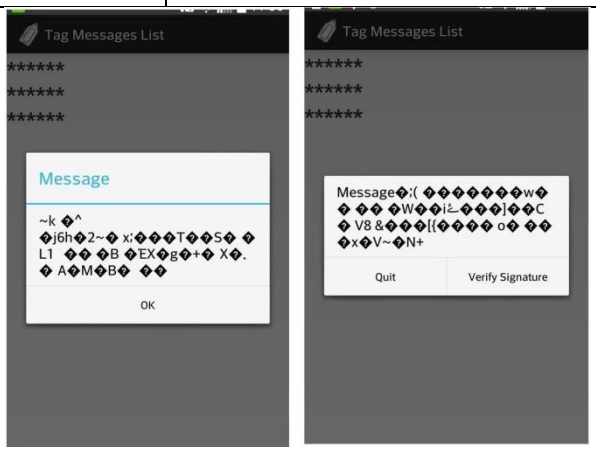

圖 5. 實測之非標籤持有者讀取結果

圖 14 為非標籤持有者 Candy 欲讀取留言的情況，其解密後顯示結果皆為亂碼。只有標籤持有者 Alice 才可以正確顯示留言訊息。此測試得以確保本留言系統的隱私性。

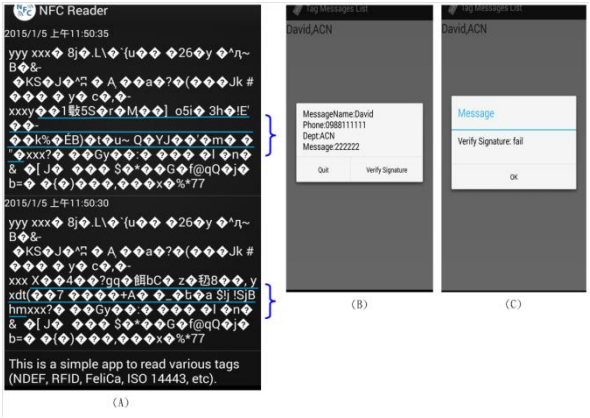

圖 15. 實作之完整性測試

圖 15，為模擬 3-3 節中的完整性分析(如圖 10)的實作測試。我們將明文修改後分別存入不同的 NFC 標籤，並且利用第三方軟體-NFC Reader[8]讀取測試數據，測試說明如表 10。本測試結果與 3-3 節相同，明文遭修改後，將於驗證簽章階段檢查出來，確保本留言系統的完整性。

表 1. 測試說明之竄改密文

Bob	留言者，留言內容為：111111。
David	惡意者，修改明文訊息為：222222。
圖 15-A	該圖中分別為修改前後的兩筆 NFC 標籤訊息，圖中}部分為遭修改明文加密而成，其餘部分不予更動。
圖 15-B	解密後留言為遭修改的內容：222222。
圖 15-C	驗證結果為失敗。

5. 結論與未來展望

隨著 NFC 手機的普及，加上 Android SDK 支援 NFC 後，越來越多團隊加入 NFC 應用研究中，但是在我們的生活中有幾個人會使用到手機的 NFC 功能呢？個人曾經因為便利貼的黏性不好，遺失了同事給我的紙張留言，這催生我將紙張式留言電子化的想法。在基於 NFC 應用越來越廣泛、現代人日益重視的個人隱私以及並非所有地區都有支援上網服務的三項前提下，我們提出一個無需連線到伺服器的狀況下確保隱私性、不可否認性與完整性的 NFC 留言系統，並且將系統實作於 Android 平台上，除了確認系統的可行性並且驗證系統的隱私性與完整性的測試，為 NFC 推廣盡一份心力。

在系統設計上,我們將讀取與寫入留言的功能合併，於使用者設定功能中可進行切換，寫入留言時可依照使用者需求選擇加密模式也可以採用數位簽章模式，加密模式的留言容量較小，簽章模式可提供讀取留言者驗證留言者身分的功能。讀取留言時如果讀取者認為此留言內容並無驗證留言者身分的需要，可以選擇不驗證簽章以節省時間。我們以產品開發為出發點設計一套使用者快速操作手冊，除了展示成果外更表達出應用層面的實用性。

典型的憑證機構採用的非對稱式金鑰長度多為 2048 bytes 以上[9]，而本系統實作採用的 RSA 加密長度只有 64 bytes，暴力破解並不需要太多時間。我們當然可以採用更長的加密長度以增加安全複雜度，但考慮到市售的 NFC 標籤容量大多在 1k bytes 以下，因此我們採用較短的加密長度，未來可視使用者需求進行調整。

市售的 NFC 標籤並無提供交互認證(mutual authentication)協定的功能，這導致我們無法防止 NFC 標籤被任何人讀取、寫入與抹除的行為。儘管攻擊者無法破解密文與偽造簽章，但其仍可以對 NFC 標籤進行抹除的行為。未來我們希望採用具有交互認證的 NFC 標籤來驗證 NFC 讀取器與 NFC 標籤雙方的合法性以增加系統安全性。

參考文獻

[1] Wikipedia, “Near field communication” http://en.wikipedia.org/wiki/Near_field_communication
[2] NFC-Enabled Cellphone Shipments to Soar Fourfold in Next Five Years,
[3] Wikipedia, “post-it note”, http://en.wikipedia.org/wiki/Post-it_note
[4] Wikipedia, “Maxwell’s equations”, http://en.wikipedia.org/wiki/Maxwell%27s_equations
[5] U. Maurer, M. Abadi, R. Anderson, M. Bellare,O. Goldreich, T. Okamoto, et al., *Information Security and Cryptography*, Second Edition, Springer Press, 2007.
[6] W. Diffie and M. E. Hellman, “Multiuser Cryptographic techniques", in *Proc. of AFIPS National Computer Conference*, Vol. 45, pp. 109, 1976.
[7] 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, No. 2, pp. 120-126, Feb 1978.
[8] Google Play, “NFC Reader”, https://play.google.com/store/apps/details?id=se.anyro.nfc_reader
[9] Wikipedia, “RSA (Cryptosystem)”, http://en.wikipedia.org/wiki/RSA_(cryptosystem)