以 IC 卡加密瀏覽器提升網路郵局郵件安全

李維倫，國立高雄師範大學

楊中皇，國立高雄師範大學

摘要

電子郵件為網路應用中重要項目之一，使用 Web Mail 收發 E-Mail 的使用者逐漸增加，大多數提供 E-Mail 服務者，同時也會提供 Web Mail。但是，到目前為止，絕大多數的 Web Mail 的系統，在安裝與設定時常忽略安全考量，但大多數的人卻認同 E-Mail 的安全應該受到重視。在這個觀念的及實際情況的落差下，借由本文運用近年來逐漸風行的 IC 卡，配合加密瀏覽器，同時利用 AES、RSA 及數位簽章的觀念實作出 IC 卡加密瀏覽器，可運用在 Open Source 的 Web Mail 系統上。我們對 Web Mail 程式進行修改，並利用自行開發的 IC 卡加密瀏覽器，對電子郵件進行數位簽章、加密及解密的工作，並達到跨平台的目標。目前本 Web Mail 網路安全系統已於 Open WebMail 及 IMP 兩套 Web Mail 系統完成測試。

關鍵字：網路安全、網路郵局、IC 卡、電子郵件、Web Mail、E-Mail

1. 前言

近幾年來網際網路蓬勃發展，從早期的撥接系統，到今日許多人使用的寬頻，顯示了網路的魅力與方便性。過去，找資料一定要到圖書館，寄信一定要投郵筒，繳電話費、買東西一定要出門，現在已經改觀，網際網路影響了我們的生活習慣。許多原本要親自到場辦理的事，現在只要透過網路即可完成。但是在享受便利性的同時，進一步想想，這些事情在網路上面做，安全嗎？舉例來說，若有人利用一些監聽軟體(如：Sniffit)，監聽網路封包，即可取得使用者傳遞的內容[4]。Sniffit 是網路安全的大敵，不管是瀏覽網頁或者發收電子郵件都有可能被監聽，並且取得監聽者所需的資訊。Web Mail 是藉由 HTTP 傳送及接收網頁，有心人士若要竊取使用者的 E-Mail 資料，並非難事。因此，衍生出 HTTPS，用

來確保連線安全。但若有網管人員有意窺視，或主機安全性設定不當，HTTPS 就會失效。E-Mail 的安全在此可看出漏洞百出。目前台灣大型的網路公司—中華電信，對於 Web Mail 並無任何保護措失。其它國內知名的入口網站：Yahoo、Pchome…等，也是如此。因而引發本文對於 Web Mail 加密的討論。且近年來，IC 卡逐漸受到重視，有鑑於目前的潮流，衍生出設計出『IC 卡加密瀏覽器』的想法，藉以提高 Web Mail 的安全。

2. 文獻探討

本研究是以 Web Mail 的安全及加密瀏覽器跨平台(Web Mail 系統)能力為主要目標，其中有用到的加密法為 AES 加密法、RSA 加密法，為確認信件的寄件者，使用了數位簽章。由於數位簽章及 RSA 加密法，需有一憑證機構，產生 Private Key 及憑證(Certificate)，提供 RSA 加密法及數位簽章使用。還需要一個 Web Mail 系統進行實際操作。以下將上述的相關背景知識，分別做以下說明。

2.1. AES 加密演算法

AES 是一種傳統加密法，原本最常用的傳統加密法為 DES。但是近年來，電腦硬體技術發展迅速，電腦運算速度快速提升，造成 DES 的安全性面臨挑戰。AES 漸漸取代 DES 成為傳統加密法的主流。AES 出現的主要推手可說是特殊的加密演算法美國國家標準技術研究所(NIST)。在 1997 年四月，NIST 發起 AES 加密演算法的徵求活動，在三年後，由 Rijndael 資料加密演算法脫穎而出，在 2000 年 10 月由美國宣布，由 Rijndael 資料加密演算法獲選最佳的 AES 演算法[8]，成為目前廣為認同的一個 AES 演算法。

AES 演算法是資料區塊固定 128 位元，金鑰長度為 128、192 或 256 位元的一種加密演算法，其運算的過程也經過反覆多次的運算。雖然經過反覆運算，但是加解密的速度仍然十分迅速。所需資源低，安全性高、加解密速度快為 AES 加密法的主要特色[1]。目前使用 AES 加密，多以 Rijndael 資料加密演算法進行實作。

2.2. RSA 加密演算法

RSA 加密演算法是一種公開金鑰 (Public Key)加密演算法，與傳統加密演算法最大的差別是，公開金鑰加密演算法的加密與解密鑰匙是不同的，分別為

Public Key 與 Private Key。在演算法中，用 Public Key 加密的郵件只能用 Private Key 解開。Rivest、Shamir 和 Adleman 發表 RSA 時，得到許多人的認同，成為一個大多數人所接受的公開鑰題加密演算法 [5] 。

利用 RSA 加密前，需要為參與的使用者，產生公開金鑰與私密金鑰，使用者要傳送資料前，先以對方的公開金鑰加密成為密文，對方收到後，再用自己的私密金鑰解密成為明文。

2.3. 數位簽章

數位簽章的主要目的，是要保護雙方在傳遞資料的過程中，不被有心人士破壞或篡改。數位簽章可分為兩大類直接式與仲裁式[5]：

2.3.1. 直接式數位簽章

直接式數位簽章只要傳送者和接收者參與簽章過程，即可進行數位簽章。由於數位簽章加密時需要傳送者的私密金鑰，解密時需要傳送者的公開金鑰。本研究主要運用此法，進行數位簽章，並假設接收者已經取得傳送者的公開金鑰憑證。如此才能完成直接式數位簽章的完整過程。

2.3.2. 仲裁式數位簽章

仲裁式數位簽章，主要是為解決數位簽章的紛爭。舉例來說，若進行數位簽章加密的傳送者，不承認自己曾經傳送資料，此時，需要一個仲裁者來中介。在傳送過程中會經過仲裁者，由仲裁者確認來源，再將資料加上日期傳給接收者。同時，也會通知接收者，此資料已經由仲裁者檢查確認。在未來如有紛爭時，仲裁者將扮演解決糾紛的主要人員。

2.4. 憑證管理中心(CA)

為了有效解決公開金鑰的傳送問題，Kohnfelder 提出使用者憑證的觀念。為了能確認公開金鑰的正確性與完整性，將公開金鑰以特定的格式，包裝成為憑證，讓使用者可以透過各種方式來傳遞憑證。這個憑證管理中心至少需符合以下條件[2]：

(1) 任何人都可以讀取憑證，得到公開金鑰及擁有人的資料。

(2) 任何人都可以確認憑證的來源。

(3) 憑證的產生及更新，完全交由憑證管理中心完成。

(4) 任何人可以檢查憑證是否到期。

運用這樣的機制，可以讓使用者能自由傳遞憑證，確認憑證是否安全可靠。

2.5. Web Mail 系統

Web Mail 系統主要目的是利用瀏覽器收發 E-Mail，可以在不固定的電腦上進行 E-Mail 的收發。目前，Web Mail 的系統種類繁多，為實驗本研究中加密瀏覽器的跨平台之能力，選擇兩種瀏覽器做為研究對象，一套是 Open WebMail[10]；另一套是 IMP[3]。Open WebMail 是以 Perl 語言寫成的 CGI 程式，其功能十分完整，除一般的收發信外，還包含通訊錄、行事曆、網路硬碟...等功能[7]。而 IMP 是利用 PHP 語言所寫成，架構於 Horde 上，單純只提供收發信件，若需其它功能(如：通訊錄、行事曆...等)，可藉由擴充模組來加強其功能。

3. IC 卡加密瀏覽器設計

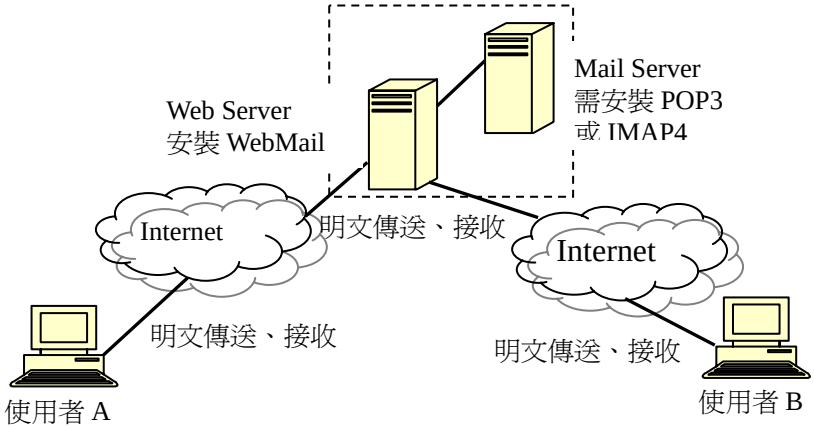
3.1. 系統架構

要設計 IC 卡加密瀏覽器前，需先瞭解目前大多數 Web Mail 系統架構，再設計出 IC 卡加密瀏覽器架構，有效對電子郵件加密。

3.1.1. Web Mail 架構

Web Mail 在 Client-Server 間主要可分成三層[3,10]，如圖一。使用者利用瀏覽器(Client 端)連線到網頁伺服器(Server 端)，這個網頁伺服器會架設 Web Mail 系統，使用者藉由 Web Mail 程式，透過 SMTP、POP3 或 IMAP4 這些協定，傳遞對 Mail Server 的要求，進行登入及收發郵件等功能。

在現在 Web Mail 的 Client-Server 的架構中，所有的資料都是未加密的明文，極易被竊取或篡改，若有較敏感的資料，應避免以這種方式傳送或接收。為了加強安全性，必需改善這種傳遞郵件的方式，加入資訊安全的觀念，導入密碼學來提昇 Web Mail 的安全性。另外補充說明，圖一中的 Web Server 和 Mail Server 可以是同一台主機，就算如此，在實際運作上仍是分層運作。本研究的加密方式主

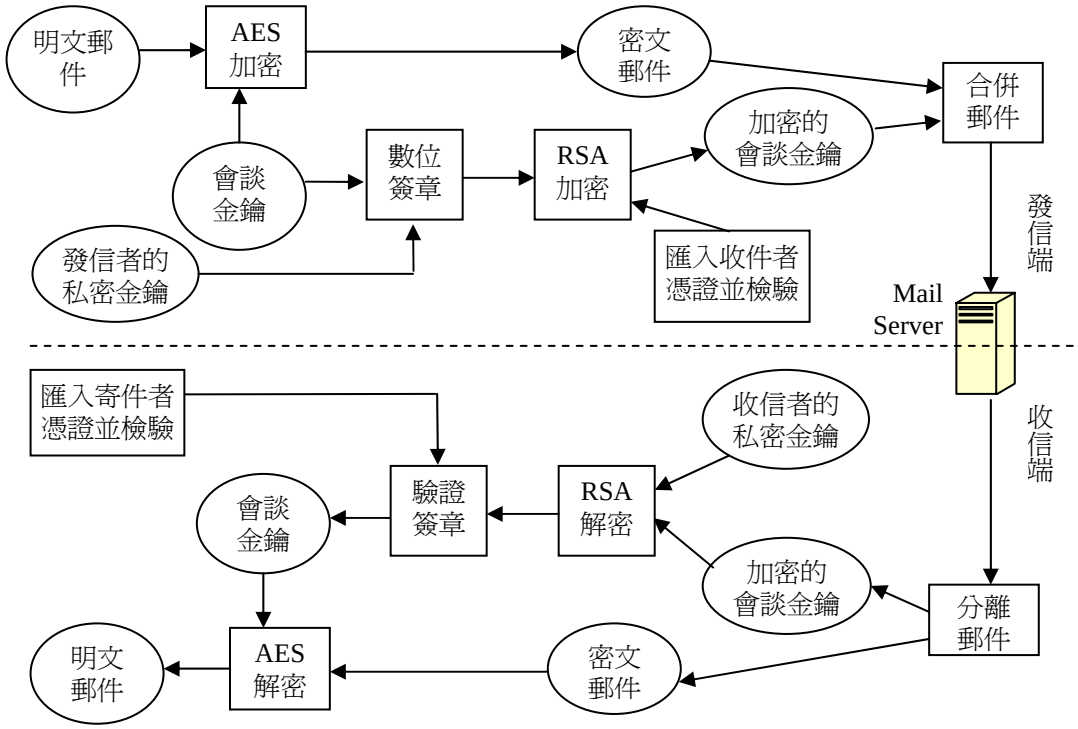


圖一 未加密的 Web Mail 架構

要是在 Cilent 端的瀏覽器對電子郵件內容進行加密。因此，在信件傳送的過程中，完全以密文方式傳遞，不會有如 HTTPS 一般，信件到達 Server 即解密的問題，也不怕遭受 Sniffit 軟體的攻擊。

3.1.2. IC 卡加密瀏覽器架構

為能利用 IC 卡加密瀏覽器對郵件內容簽章加密，本研究中運用以下的加密架構(如圖二)。



圖二 郵件加密架構

利用 AES 先對明文郵件加密，此時，AES 加密過程會產生一把會談金鑰及密文郵件，由於 AES 屬於對稱式加密，所以在郵件送出的同時也要把這把會談金鑰傳送到收件者，收件者再利用這把會談金鑰把郵件解開。完成郵件加密後利用 RSA 來處理這把金鑰，首先取得收件者的憑證，檢驗憑證，驗證收件者的憑證無誤後，利用寄件者的私密金鑰對金鑰簽章，再從收件者的憑證中取出公開金鑰加密，得到加密過的會談金鑰，附於郵件中，傳送給收件者。收件者收到郵件後，會得到密文郵件及加密過的會談金鑰，利用自行保管的私密金鑰及寄件者的憑證中取出的公開金鑰對加密過的會談金鑰進行解密並驗證簽章，得到 AES 當時加密時的會談金鑰，利用會談金鑰，將密文郵件解密，得到明文郵件。其中使用者的 Private Key 以 IC 卡儲存，提高私密金鑰的安全性。

3.2. 系統實作

3.2.1. 利用 C++ Builder 實作瀏覽器

C++Builder[9]中有一個 TCppWebBrowser 的元件，這個元件中提供了一些瀏覽器的基本功能。其製作的流程如下：製作按鈕示→建立專案→插入元件→撰寫程式→處理已知 TCppWebBrowser 的問題。在這裡值得注意的是，TCppWebBrowser 元件在<textarea>標籤中 Enter 鍵會無法跳行，必須進行修正。

3.2.2. 加入加解密功能

在研究中分別運用 AES 及 RSA 做加密功能，在 AES 的部份以 Rijndael 的加密法進行加密，而簽章及加密的部份則利用 OpenSSL[6]進行實作，同時，也利用 OpenSSL 架設一 mini CA 提供實際運作時所需的私密金鑰及憑證。在加密的過程中將網頁中<textarea>標籤中的信件內容取出，進行簽章加密的流程後再將密文寫回<textarea>標籤中，提供寄件者寄出。

3.2.3. 加入匯入及驗證憑證功能

要加密、簽章及驗證憑證需要有三個檔案：收件者的憑證、寄件者私密金鑰及 CA 的憑證。要解密、證證簽章及驗證憑證需要有三個檔案：寄件者的憑證、收件者私密金鑰及 CA 的憑證。在程式中提供匯入以上三個檔案的功能，匯入完成後，再以 CA 的憑證，驗證憑證的可靠性。此時，匯入憑證的功能除了能以一般檔案形式匯入私密金鑰外，亦可選擇由 IC 卡匯入私密金鑰。

3.2.4. 導入 IC 卡到加密瀏覽器

由於私密金鑰存於 IC 卡中，故需將 IC 卡的讀取功能加入瀏覽器中，才能將 IC 卡內的私密金鑰讀出，並寫回檔案，成為憑證的標準格式。準備在簽章或解密的過程中使用。但是，金鑰要寫入 IC 需利用另一支程式(如圖三)，選擇要寫入的私密金鑰後，進行寫入 IC 卡的動作，再將 IC 卡交給使用者使用。

圖三 金鑰寫入程式

3.2.5. Web Mail 系統的修改

目前的 Web Mail 系統非常多樣，但是最後一定是以 HTML 的形式送給瀏覽器，加密瀏覽器利用這樣的特性，將標籤中設有『id=body』的內容取出，進行加密。因此，不管任何的 Web Mail 系統，只要在寫信或讀信內容所使用的 HTML 標籤中，加入『id=body』，即可由加密瀏覽器進行加解密，在本研究中以 Open WebMail 及 IMP 兩套系統實作，證明其跨平台的能力。

4. IC 卡加密瀏覽器應用

IC 卡加密瀏覽器有關加解密的功能如圖四所示，其中包含(1)IC 卡讀取偵測；(2)IC 取出；(3)匯入 PEM；(4)檢驗憑證；(5)加密；(6)解密。其中功能有部份功能相依性，減低操作錯誤的機率。如：未插入 IC 卡，其它功能則無法使用；未匯入憑證，檢檢憑證功能將被取消...等。

以下將針對 IC 卡加密瀏覽器進行加解密流程測試，測試順序如下：準備以 Open WebMail 發出郵件(如圖四)→插入 IC 卡→進行憑證及私密金鑰匯入(如圖五)→憑證檢驗→簽章加密成為密文(如圖六)→寄出。

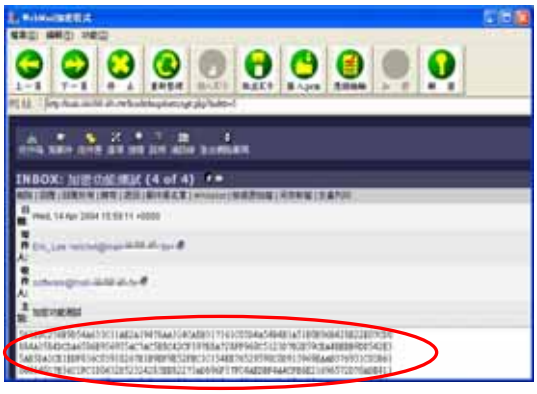
圖四 瀏覽器加密功能及 Open WebMail 信件內容加密前

圖五 匯入憑證及金鑰

圖六 加密後內容成為密文

其中在匯入憑證時需分別匯入 CA 憑證(驗證用)、對方(收件人)憑證(加密用)及自己(寄件人)的私密金鑰(數位簽章用)，在私密金鑰部份為增加安全性及便利性，加入以 IC 卡匯入私密鑰的功能(如圖五)。

為展現本程式之跨平台能力，收信端解密利用 IMP 實作，解密流程如下：我們以 IMP 讀取密文郵件(如圖七)→插入 IC 卡→匯入憑證及金鑰(如圖五)→ 驗證憑證→將信件內容解密(如圖八)。



圖七 IMP 讀信畫面信件為密文

圖八 IMP 信件解密後成為明文

在研究中不僅可於 Open WebMail 發信，可以以 IMP 收信解密。經過測試，不管由 Open WebMail 或 IMP 任何一方收信或發信，均能完成郵件的加解密。

5. 結論

IC 卡加密瀏覽器目前的運用主要在於 Web Mail 的加密，希望能夠提供一個安全的 E-Mail 環境，它的主要特色如下：

(1) 能在使用者端將信件加密完成，讓信件到達收件者的電腦之前完全以密文形式傳送。

(2) 運用 IC 卡儲存私密金鑰增加安全性。

(3) 跨平台能力強，只要 Web Mail 系統的程式程做些許改變，即可使用。

(4) 配合 AES、數位簽章及 RSA，提高信件傳遞的安全性與可靠性。

(5) 在使用者解密後，Mail Server 上的信件仍然是密文，不會因閱讀後即解

密。

目前此軟體已經能實際運作，可以運用在需要安全傳輸 E-Mail 的環境。由於與 IC 卡整合，在未來政府的自然人憑證若是普及，只要進行 IC 卡讀取程式的部份修改，即可結合自然人憑證，成為多數人皆可使用的電子郵件加密方式。

6. 參考文獻

1. Daemen J., Vincent Rijmen"AES Proposal: Rijndael", Document Version 2, Mar, 9, 1999.
2. Kohnfelder L., Towards a Practical Public-Key Cryptosystem. Bachelor's Thesis, M.I.T., May 1978.
3. Rostetter E., http://www.horde.org/faq/, April 4, 2004.
4. Schwarz M., Anderson J., Curtis P., Murphy S., Multitool Linux: Practical Uses for Open Source Software, May 2002, Addison Wesley.
5. Stallings W., Cryptography and Network Security Principles and Practices, 3/e, Nov. 2002, Prentice Hall.
6. Viega J., Messier M., Chandra P., Network Security with OpenSSL, June 2002, O'Reilly.
7. 王俊斌，FreeBSD架設管理與應用，2003年6月初版，台北縣，博碩文化。
8. 林祝興、葉義雄、楊國鴻，Rijndael加密演算法的介紹，資訊安全通訊，2000年9月，第六卷第四期。
9. 黃嘉輝，C++ Builder揭開互助社群軟體－Napster的密秘 Internet與TCP/IP進階程式設計，2001年3月初版一刷，台北市，文魁資訊。
10. 董仲愷，http://turtle.ee.ncku.edu.tw/~tung/openwebmail/contest/final.doc，2003年，未出版，台南市，成功大學。