網路安全的理論與實務

楊中皇 著

第七章 金鑰管理

http://crypto.nknu.edu.tw/textbook/

伴 您 學 習 成 長 的 每 一 天

金禾資訊

第七章 金鑰管理 (Key Management)

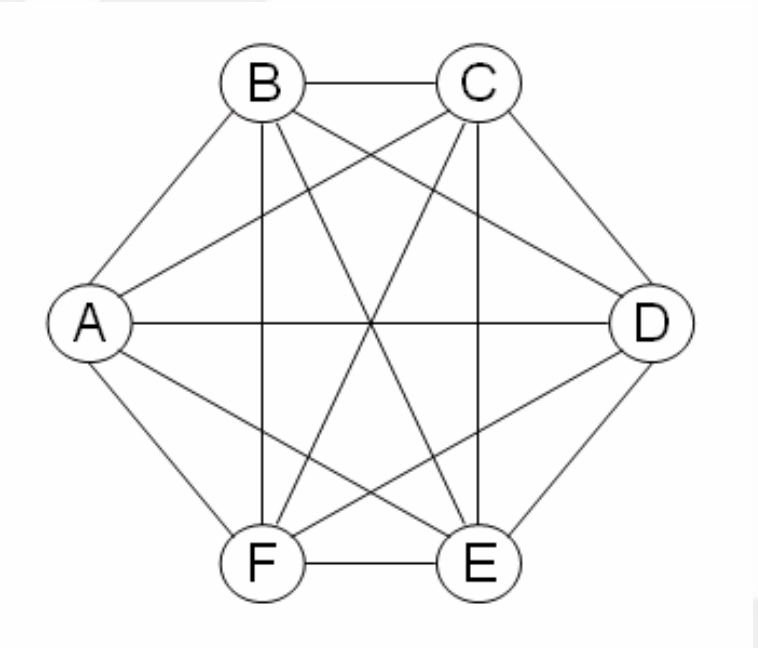
- 金鑰分配中心(KDC)
- 憑證機構(CA)
- 一次通行碼 (One-Time Password)

金鑰管理

- 網路安全的設計細節都可公開，唯一的秘密是金鑰，金鑰的管理與應用自然成爲重要議題
- 金鑰管理包含金鑰的生成 (Generation)、儲存 (Storage)、配送 (Distribution)、認定 (Verification)、啓用 (Activation)、更改 (Replacement)、撤銷 (Revocation)、終止 (Termination)等
- 我們介紹金鑰管理的機制包括KDC (金鑰分配中心)與CA(憑證機構)，同時介紹挑戰與回應(Challenge and Response)認證與一次通行碼(One-Time Password)的設計

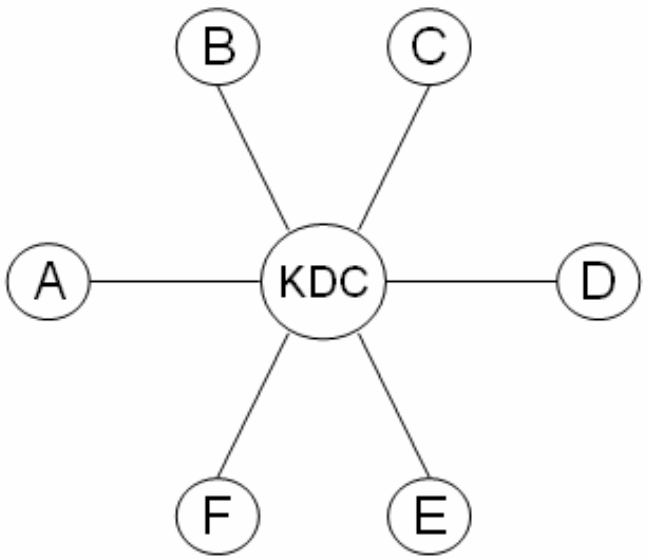
直接金鑰配送

- 私密金鑰密碼系統(如AES或DES)用於保密或防偽時通信任意兩方都需有共同的金鑰。最簡單的方式是讓通信雙方自行約定共通的金鑰，每位使用者需記憶與其他所有使用者的共通金鑰。如果系統有*n*個使用者，且任意兩使用者皆需互相安全通信，那麼每位使用者需記住與其他*n-1*個使用者之間的共通金鑰，系統則共有*n*(*n-1*)/2金鑰



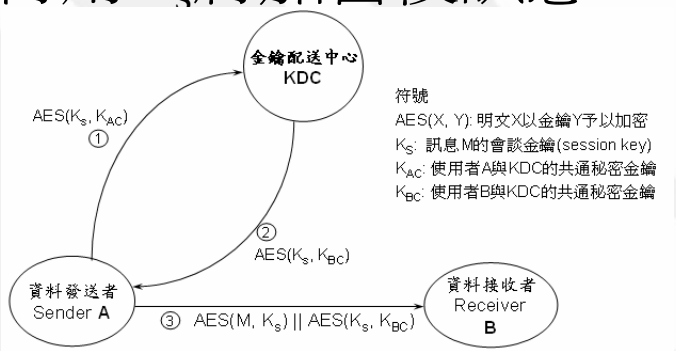
金鑰分配中心(KDC)

- 大量使用者或通信雙方事先不認識時可採金鑰配送中心(KDC，Key Distribution Center)的方式
- 系統所有使用者都需信賴KDC，而KDC伺服器中存有所有使用者與KDC共享的秘密金鑰，也就是說每位使用者與KDC有秘密共通的金鑰
- 透過KDC的參予，任意兩使用者便可安全的通信
- 但是萬一有駭客入侵並破解KDC時，整個系統就完全無安全性可言



利用KDC取得雙方共通金鑰

- 假設使用者A與B之間沒有共通秘密金鑰，但A與KDC有共通金鑰K_{AC}，B與KDC有共通金鑰K_{BC}，而A想將訊息M採用AES加密後傳給B
- A先將隨機產生的會談金鑰K_s以K_{AC}加密後，意即AES(Ks, K_{AC})，再與預定的資料接收者B的身分傳至KDC
- KDC收到AES(K_s, K_{AC})後先用共通金鑰K_{AC}解密得到會談金鑰K_s，再將K_s以A訊息預定接收者B與KDC的共通金鑰K_{BC}加密後得到AES(K_s, K_{BC})傳回給A
- A將訊息M採用AES以K_s當金鑰加密得到AES(M, K_s)後與AES(K_s, K_{BC})傳給B。B收到AES(K_s, K_{BC})後先用共通金鑰K_{BC}解密得到會談金鑰K_s，再用K_s將加密後訊息AES(M, K_s)解密得到訊息M



憑證機構(CA)

- 公開金鑰與私密金鑰總是成對出現，而且是經過計算而配對，不是隨意挑選的。雖然公開金鑰本身是可以公開的，可是我們還是有管理上的問題，也就是如何確保公開金鑰與實體(entity)的關係
- 憑證機構 (CA，Certification Authority)與KDC相同的是它也是一個可信任的第三者
- 每個人可以在CA上面註冊自己的公開金鑰並取得CA以自己私密金鑰簽發的憑證，而使用者自己保管對應的私密金鑰
- 憑證的內容包含申請者的個人資料與其私密金鑰相對應的公開金鑰。相對於KDC，CA較不擔心被破壞。若CA放在一個不安全的場所而遭受到侵入時，就算CA遭破壞，但是破壞者仍無法得知CA的私密金鑰，因此無法產生僞造憑證。CA因爲要負責管理所有人的憑證，所以需使用長度較長較安全的金鑰(例如2048位元RSA金鑰)來製作憑證
- CA也可提供類似LDAP(本書第十七章)的輕量級目錄存取協定，也就是將使用者的公開金鑰公開以提供查詢。當使用者B要送需要加密的文件給A時，就到CA/LDAP上面去查詢A的公開金鑰，然後將文件以A的公開金鑰加密後直接傳送給A。A收到後就以自己的私密金鑰解密以得到明文。在整個過程中CA只負責將使用者的公開金鑰提供查詢，並不介入文件的傳遞過程
- 我們在第十五章還有進一步的CA建置與使用說明。

CA基本功能

- CA提供下列的基本功能：

憑證申請
  - 提供實體申請其所屬的數位憑證。憑證目前多採國際標準X.509第三版的格式

憑證註銷
  - 註銷尚在有期間內實體憑證

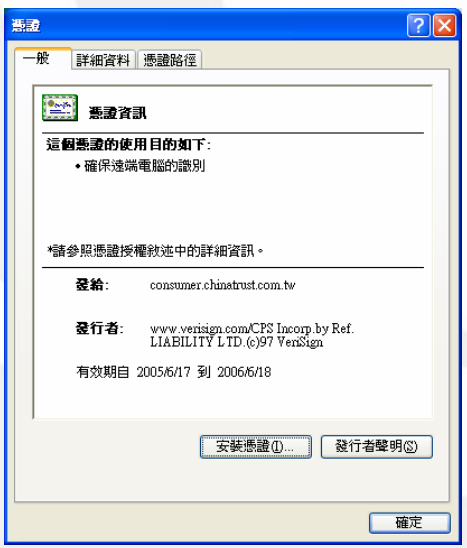
憑證查詢
  - 可以查詢實體的憑證資料，以便得到他人的公開金鑰

憑證展期
  - 若因爲其他因素必須延長有效期限，CA也提供憑證延期的服務

憑證廢止清冊(CRL，Certificate Revocation List)列表
  - 當憑證註銷後，系統便會將憑證資料轉入此表中，讓使用者可以查詢

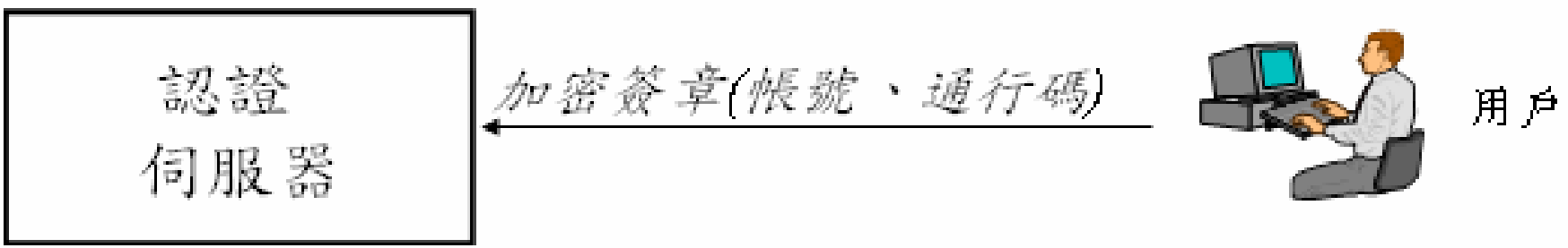
憑證的內容

- 微軟公司從Windows 2000開始內建對公開金鑰與憑證的支援。目前瀏覽器安全上網(https://...)時可顯示網頁伺服器憑證的內容
- 下圖以某家網路銀行爲例，上網時可檢視憑證(例如選擇檔案->內容->憑證，或是雙擊IE瀏覽器右下角黃色鎖的符號)，包括是誰發行這個憑證的與這個憑證的有效日期等。如果檢視詳細資料還會看到這個實體(如伺服器)公開金鑰值，還有憑證版本序號與憑證所採用的演算法



一次通行碼 (One-Time Password)

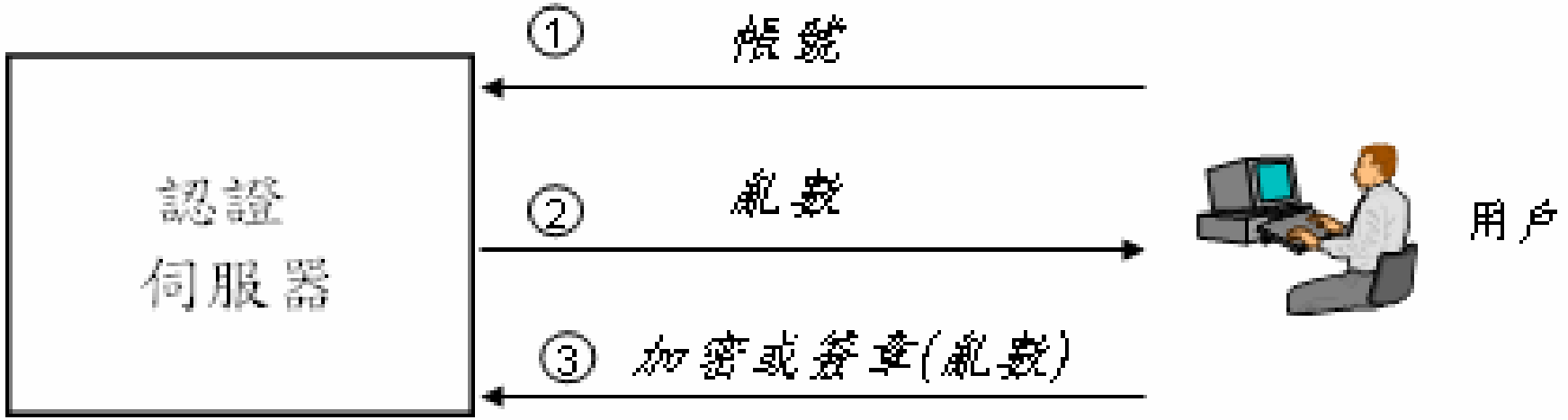
- 使用加密或簽章雖可防止資料在公開網路上外洩或被竄改，但仍必須慎重使用，否則仍會受到攻擊。
- 以登入爲例，如果用戶將帳號與通行碼利用認證伺服器(authentication server)的公開金鑰加密，甚至利用自己的私密金鑰做簽章，這可達到網路保密與防僞的功能。雖然無法窺視通信的內容，然而聰明的駭客可將整個通信的封包擷取，事後重送(replay)封包到伺服器進行登入。



防止重送攻擊

- 要防止上述的重送攻擊，可在傳送的訊息加上時戳(time stamp)或序號(sequence number)讓伺服器檢查是否封包被重送或是採用挑戰與回應(challenge and response)的方式
- 假設用戶與伺服器有共通的金鑰或是用戶的公開金鑰事先登錄於伺服器。要登入時，用戶先將帳號傳給伺服器，伺服器在隨機產生亂數(挑戰)給用戶，用戶再將亂數以共通金鑰加密或是以用戶的私密金鑰簽章(回應)傳回給認證伺服器，認證伺服器再檢驗回應與挑戰是否符合
- 這種方式每次的通行碼都不同，故稱爲一次通行碼(one-time password)，這是以動態的通行碼取代傳統靜態的通行碼

使用動態的一次通行碼登入



參考資料

- Department of Defense, *Public-Key Infrastructure and Key Management Infrastructure Token Protection Profile V3.0*, http://www.niap.nist.gov/cc-scheme/PP_PKIKMITKNPP-MR_V3.0.pdf, March 2002.
- A.J. Menezes, P.C. van Oorschot, and S.A. Vanstone, *Handbook of Applied Cryptography*, CRC Press Series on Discrete Mathematics and Its Applications, 1996.
- William Stallings, *Cryptography and Network Security: Principles and Practice*, 3rd edition, Prentice-Hall, Inc. 2002.