

資訊安全期末專案

林永松老師

專案目的：練習使用非對稱式加解密演算法 (Asymmetric Cryptography Algorithm) 及雜湊函數 (Hash Function)，達成機密性 (Confidentiality) 與完整性 (Integrity) 之需求，並開發具備防範 Replay Attack 之傳輸機制。

專案分組：至多二個人一組，分別扮演設備端及原廠端。程式開發語言及執行平台：不限。

非對稱式加解密演算法：須提供 RSA、ECC 二種不同的加密方式，供使用者擇一使用。雜湊函數：須提供 SHA-1、MD5、Whirlpool 三種不同的方式，供使用者擇一與上述任一種加解密演算法搭配使用。

計畫展示環境：自備，須現場使用產生設備端及原廠端各自的公開金鑰 (Public Key) 及私密金鑰 (Private Key)，並交換彼此的公開金鑰。

授權碼及更新檔檔案格式：皆為 html 或 XML 檔案；授權碼檔案內容須為一百個英數字混合的字串 (自訂)，更新檔檔案內容須出現 "Update Successfully" 字串 (其餘內容自訂，惟此檔案大小須大於 5K Bytes)。

加密檔案格式：分成三個部分，包括 Header、Package、Trail。

1. Header：大小為 1024 bytes
 - 描述區：512 bytes，助教於展示時隨機輸入的文字須放入此區域。
 - 保留區：512 bytes。
2. Package：此區域存放加密過的授權碼或更新檔。
3. Trail：大小係依照所選定之雜湊函數而定 (例如 MD5 為 16 bytes)；將 Header 及 Package 合併後，運用所選定之雜湊函數進行運算，產生雜湊值 (Hash Value) 放入此區域。

加密檔案傳送方式：設備端與原廠端須在不同主機上運作，程式本身須具備加密檔案傳送能力，不透過外部程式的輔助，以網路傳輸方式自動交換資料。

作業流程一：

1. 原廠端：廠商人員透過系統上傳更新檔，系統收到更新檔後自動通知所有設備端有 patch 檔可供更新。
2. 設備端：收到通之後，設備端將授權碼使用原廠端的公開金鑰加密後，依照前述之加密檔案格式需求，發送至原廠端進行驗證。
3. 原廠端：收到設備端發送之授權碼後，使用本身的私密金鑰解密，驗證無誤後，再將更新檔使用設備端的公開金鑰加密後，依照前述之加密檔案格式需求，發送至設備端進行更新。
4. 設備端：收到原廠端發送之更新檔後，使用本身的私密金鑰解密，以進行系統升級。

展示檢查項目：

1. 各組展示時，須先行向助教顯示原始程式中，有關非對稱式加解密演算法及雜湊函數之程式碼。
2. 助教打開任一加密檔案檢視時，除了 Header 的描述區外，不可有任何明文 (Plaintext) 存在。
3. 加密檔案發送至對方後，助教之前所隨機輸入的文字，須能獨立顯示在螢幕上。
4. 加密檔案解密前，須先進行雜湊值的比對。亦即先將加密檔案的 Trail 部分獨立取出，並針對該加密檔案的 Header 及 Package 之合併部分，運用所選定之雜湊函數進行運算，產生雜湊值。二個雜湊值須相同，且須以十六進位表示法顯示在螢幕上，並經助教確認後，方得進行後續解密動作（加密檔案之 Package 部分）；若二者不相同，則捨棄所得之檔案並顯示警告訊息。
5. 加密檔案解密後，須自動開啟檔案內容，以便助教進行驗收。

作業流程二：

透過作業流程一，可確保授權碼或更新檔之機密性與完整性，但無法達成發送方之不可否認性 (Non-Repudiation)，即無法保證授權碼是由設備端所發送，或無法保證更新檔是由原廠端所發送。作業流程二即是要達成此項保證。

加密方：

1. 將加密後之授權碼或更新檔，運用所選定之雜湊函數進行運算，產生雜湊值。
2. 將前述的雜湊值，使用本身的私密金鑰加密，以確保授權碼或更新檔的發送來源。
3. 將雜湊值加密後的結果，放入 Header 之保留區，隨著加密檔案一起發送至對方。

解密方：

1. 將加密檔案 Header 之保留區中的資料獨立取出，並使用對方的公開金鑰解密，得到一個雜湊值。
2. 將加密檔案之 Package 部分，運用所選定之雜湊函數進行運算，產生雜湊值。
3. 前二步驟中所得之雜湊值相同，方得進行後續解密動作（加密檔案之 Package 部分）；若二者不相同，則捨棄所得之檔案並顯示警告訊息。

展示檢查項目：

解密方須顯示所得之加密雜湊值可成功解密，並顯示自行運算所得之雜湊值，二者須相同並以十六進位表示法顯示在螢幕上。

作業流程三：

實作 Public-key certificates 流程，雙方將自身 public key 送至 Certificates authority，取得自己的 CA(內容如課堂所述)，並且傳送給對方，對方要能驗證此為合法的 CA。

1. 請事先在 Certificate Authority 建立一個合法使用者的 IP List，利用 IP 合法性驗證使用者身分。若無第三台電腦，Certificate Authority 可建在原廠端或設備端其一，但傳輸還是必須透過網路連線機制。
2. 送出自身的 public key 向 Certificate Authority 提出申請。
3. 在 Certificate Authority 判明申請者的身份後，將該公開金鑰與申請者的身份資訊綁在一起，並為之簽字後，便形成證書發給那個用戶（申請者）。
4. CA 的格式： $CA = E (PR_{auth}, [T_i || ID_A || PU_a])$ ，T 為任意訊息、ID 為雙方各自的 IP 位置。
5. 雙方互相驗證對方的 CA，利用 Certificate Authority 的公鑰解開 CA，取得對方的 public key。
6. 之後利用此 public key，再次實作作業流程一。

展示檢查項目：

1. Certificate Authority 需顯示製作 CA 過程，包含接收到 public key 與產生 CA，以及將雙方 CA 未加密詳細內容自動產生為 txt 檔案供助教檢查內容(格式自訂)。
2. 雙方傳送的文字 T 必須可在流程開始前提供助教隨意輸入文字，當傳送自對方解密完成後，文字必須與原先輸入的一致，否則即算是失敗。
3. 流程一展示部分與流程一需求一致。