計算理論與演算法實驗室

古典與(後)量子理論密碼學

鐘楷閔
副研究員

計算理論與演算法實驗室的理論密碼學團隊主要致力於理論密碼學的研究，研究課題包含古典(非量子)密碼學到(後)量子密碼學。近幾十年來，密碼學的發展已經超越其原本安全傳輸的目的，可以安全的實現許多更複雜的任務。今日，密碼學無所不在的穿梭在我們現今的生活中，並且不知不覺地在使用它，例如：電子郵件、行動電話、SSL、電子商務與電子投票等。

在本文中，我們將試著以淺顯易懂的方式介紹密碼學最近的重大進展。我們將以加密法為例，先簡介其概念，再介紹近十年才被提出與達成的兩種進階加密法：全同態加密 (Fully Homomorphic Encryptions) 以及功能加密系統 (Functional Encryptions)。這兩種加密法都可以允許我們在不知道資料內容的情況下對加密的資料上做運算！我們將藉由幾個實際應用和類比的方式來介紹這些概念。最後，我們將簡述我們團隊近期的研究發展以及我們的研究方式。

在歷史上，密碼學早期主要提供安全的傳輸通道來傳輸私密資料(如軍事或是政府單位)，如同電影 -- 模仿遊戲所講述的故事 [1]。考慮 Alice 想要送一個私密訊息 m 去給 Bob，而訊息的傳輸可能會被一個竊聽者 Eve 所竊聽。因此，Alice 不能直接寄送訊息，否則會被 Eve 所聽到。安全傳輸的目標就是確保 Bob 可以正確地收到 Alice 傳來的訊息，同時確保 Eve 無法學習到訊息的相關內容。舉例來說，在線上消費時，信用卡以及安全碼的資訊必須傳送到銀行，而安全傳輸可以提供訊息的保護以防範中間的竊聽者。

公開金鑰加密法 (public key encryption, PKE) 是一個能達到此目的密碼機制。一個 PKE 主要由三個演算法組成 (KeyGen, Enc, Dec)。一個接收者可執行金鑰生成演算法 KeyGen 來生成一組金鑰對，一把為任何人都可以看到的公鑰 pk、另一把為只有接收者可以擁有的私鑰 sk。任何人可以透過公鑰 pk 來執行加密演算法 Enc 將訊息 m 製成密文 $ct = Enc_{pk}(m)$；擁有 sk 的人可以執行解密演算法 Dec 將 ct 回復成訊息 $m = Dec_{sk}(ct)$。

利用 PKE，Bob 可先使用 KeyGen 產生 (pk, sk)，將公鑰 pk 送給 Alice。Alice 可以利用 Bob 的公鑰 pk 來加密想傳送的訊息 m 來產生密文 $Enc_{pk}(m)$。只有握有私鑰 sk 的 Bob 可以解密而得到 m。沒有 sk 的人無法學習到訊息 m 的相關資訊。因此，雖然竊聽者 Eve 能學到 ct，因為 Eve 不知道 sk，能保證 Eve 不會學到任何有關 m 的資訊。

如何嚴謹的定義所謂的 “Eve 從 ct 學不到任何有關 m 的資訊” 的安全性其實是非常不容易的事情。 事實上，2012 年獲得圖靈獎 (A. M. Turing Award，資訊科學界的諾貝爾獎) 得主 Shafi Goldwasser 和 Silvio Micali 的獲獎主要貢獻之一就是提出正確的加密法的安全性定義。(有興趣的讀者可參考網路上的資料 [2, 3, 4] 來深入了解密碼學。) 在本文我們將不深入介紹如何嚴謹的定義安全性，但試著以類比的方式(如圖一)來幫助讀者理解：我們可以想像 sk 是一個實體的鑰匙，而 pk 是一個可用來製造箱子的模子，而製造出來的箱子只能以 sk 來開鎖。想像 Alice 要傳送的 “訊息” 是珍貴的珠寶。Alice 想安全的寄送珍貴的珠寶給 Bob，可以利用 Bob 所提供的模子製作出箱子，把珠寶鎖在這箱子裡，將上鎖的箱子寄給 Bob。這樣可以保證只有 Bob 可以開鎖取得珠寶，而無法被其他人竊取。

全同態加密法

在現今雲端運算發達的時代，一個常見的應用是外包我們的資料與計算給雲端伺服器 (如：Amazon EC2)。然而，當資料是較敏感的內容時(如：個人隱私資料、病歷紀錄)，外包給無法完全信任的伺服器可能會有損隱私。抽象來說，考慮用戶端想要外包其資料 m 與想計算的函數 f 給伺服器(如:對病歷進行統計分析)，我們是否可以在不洩漏資料 m 給伺服器的情況下讓伺服器幫我們算出答案 $f(m)$?

這個看似矛盾的任務，能由所謂的全同態加密法 (fully homomorphic encryption, FHE) 所達成的。FHE 是一個能允許在加密後的密文上做計算的進階加密法。具體來說，FHE 比一般的 PKE 多了一個同態運算演算法 Eval。Eval 演算法可對給定