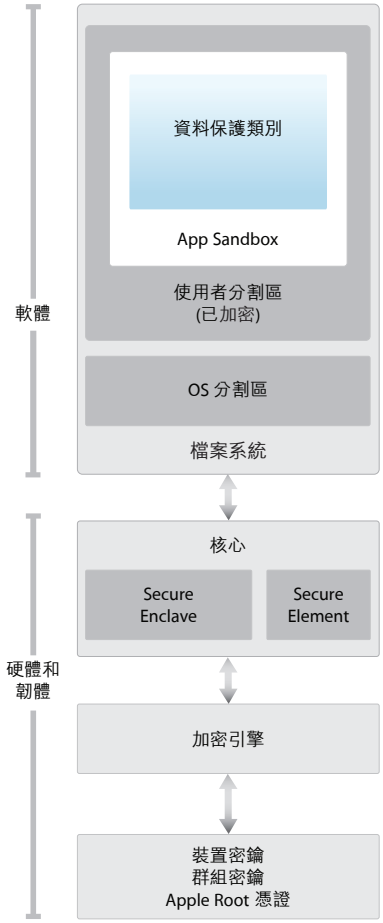
iOS 安全性

iOS 9.0 或以上版本

2015 年 9 月

目錄

簡介



iOS 的安全性架構圖提供視覺化的圖表概覽，以說明本文件中討論的各項技術。

Apple 所設計的 iOS 平台將安全性視為其核心訴求。當我們開始打造可能的最佳行動平台時，我們汲取數十年的經驗來建造一個全新的架構。我們考量了有關桌面系統環境的安全性風險，並在設計 iOS 時建立一套因應安全性的新方式。我們開發並整合創新功能，可緊密結合行動安全性並從一開始便保護整個系統。因此，iOS 對行動裝置而言，在安全性上往前邁進了一大步。

軟體、硬體和服務在每台 iOS 裝置上緊密合作，一同為使用者提供最高的安全性和直接的使用者體驗。iOS 不僅保護裝置和其中的靜態資料，同時也保護了整個生態系統，包含使用者在本機、網路上以及使用重要 Internet 服務所執行的所有操作。

iOS 和 iOS 裝置不僅提供進階的安全性功能，而且還容易使用。許多安全性功能預設便已啟用，因此 IT 部門無須執行大量的設定動作。而如裝置加密之類的重要安全性功能則無法設定，因此可避免使用者不小心地停用這些功能。其他功能（如 Touch ID）則讓裝置安全性的操作更簡單且直覺，進而提升了使用者體驗。

本文件提供有關 iOS 平台中安全性技術與功能如何導入的詳細資訊。本文件也能協助各個組織將 iOS 平台安全性技術與功能與其本身的政策和程序相結合，以滿足其特定的安全性需求。

本文件主要分為以下幾個主題：

- **系統安全性**：iPhone、iPad 和 iPod touch 上經過整合且安全的軟硬體平台。
- **加密與資料保護**：若裝置遺失或遭竊，或有未經授權的人員嘗試使用或修改裝置時，對使用者資料進行保護的架構和設計。
- **App 安全性**：可讓 App 安全執行且不犧牲平台完整性的系統。
- **網路安全性**：對傳輸中的資料提供安全認證和加密的產業標準網路通訊協定。
- **Apple Pay**：Apple 安全付款的方式。
- **Internet 服務**：Apple 以網路為基礎的架構，提供傳訊、同步和備份等服務。
- **裝置控制**：防止在未經授權的情況下使用裝置，以及在裝置遺失或遭竊時可進行遠端清除的方式。
- **隱私控制**：iOS 中可用來控制「定位服務」與使用者資料取用權限的功能。

系統安全性

進入裝置韌體升級（DFU）模式
在裝置進入 DFU 模式後加以回復，可讓裝置回到已知的正常狀態，在該狀態下只會有未經修改的 Apple 簽署的程式碼。DFU 模式可透過手動方式進入：首先，請使用 USB 接線將裝置連接到電腦，然後同時按住「主畫面」和「睡眠/喚醒」按鈕。8 秒後，繼續按住「主畫面」按鈕，同時放開「睡眠/喚醒」按鈕。注意：當裝置處於 DFU 模式時，螢幕上並不會顯示任何內容。若顯示 Apple 標誌，表示按住「睡眠/喚醒」按鈕的時間過長。

系統安全性旨在確保每部 iOS 裝置的所有核心元件都能為軟體和硬體提供安全保護。這包含啟動程序、軟體更新和 Secure Enclave。此架構是 iOS 安全性的核心，並不會影響裝置的正常使用。

iOS 裝置的硬體和軟體經過緊密的整合，可確保系統的每個元件獲得信任，並對系統整體進行驗證。從初次啟動到 iOS 軟體更新、再到第三方的 App，每個步驟都經過分析和審查，以確保硬體和軟體以最佳方式協同執行，並適當地使用資源。

安全啟動鏈

啟動程序中每個步驟包含的元件都經過 Apple 加密簽署以確保其完整性，且只有在驗證信任鏈結後，每個步驟才能繼續。這包含 bootloader、核心、核心延伸功能和基頻韌體。

開啟 iOS 裝置後，其應用程式處理器會立即執行唯讀記憶體（稱為 Boot ROM）中的程式碼。此類無法更改的程式碼（稱為硬體的信任 root）是在製造晶片時完成設定，且已間接獲得信任。Boot ROM 程式碼包含 Apple Root CA 公用密鑰，該公用密鑰用來驗證 Low-Level Bootloader（LLB）是否經過 Apple 簽署，以決定是否允許其載入。這是信任鏈結中的第一步，信任鏈結中的每個步驟都會確保下一個步驟經由 Apple 簽署。當 LLB 完成其任務後，便會驗證和執行下一階段的 bootloader（即 iBoot），接著會驗證並執行 iOS 核心。

此安全啟動鏈有助於確保底層的軟體未經竄改，並只允許 iOS 在經過驗證的 Apple 裝置上執行。

對於具有行動網路連線功能的裝置，基頻子系統也會利用其類似的安全啟動程序，包含已簽署的軟體以及由基頻處理器驗證的密鑰。

對於搭載 A7 或更新版本 A 系列處理器的裝置，Secure Enclave 副處理器也會利用安全啟動程序，以確保其獨立的軟體經過 Apple 驗證和簽署。

若此啟動程序的某個步驟無法載入或驗證下一個程序，啟動就會停止，裝置螢幕上會顯示「連接 iTunes」。這即是所謂的恢復模式。若 Boot ROM 無法載入或驗證 LLB，其會進入 DFU（裝置韌體升級）模式。在這兩種情況下，裝置都必須透過 USB 連接 iTunes，並回復到出廠設定。如需手動進入恢復模式的更多資訊，請參閱 support.apple.com/kb/HT1808?viewlocale=zh_TW。

系統軟體授權

Apple 會定期釋出軟體更新以解決新產生的安全性問題，同時提供新功能；這些更新會同時提供給所有受支援的裝置。使用者會在裝置上和透過 iTunes 收到 iOS 更新通知，而更新項目可透過無線方式傳送，以鼓勵使用者可儘快採用最新的安全性修正。

上述的啟動程序有助於確保裝置上只能安裝 Apple 簽署的程式碼。為了避免裝置降級到缺少最新安全性更新的較舊版本，iOS 使用了名為「**系統軟體授權**」的程序。若可將裝置降級，攻擊者一旦有了裝置的擁有權，便會安裝較舊版本的 iOS 並利用舊版本中尚未修復的漏洞進行破壞。

在搭載 A7 或更新版本 A 系列處理器的裝置上，Secure Enclave 副處理器也會利用「系統軟體授權」來確保軟體的完整性，並阻止降級的安裝作業。請參閱下面的「Secure Enclave」。

iOS 軟體更新可使用 iTunes 安裝，或在裝置上採用無線方式（OTA）進行安裝。若使用 iTunes，系統會下載並安裝完整的 iOS 拷貝。若採用 OTA 方式安裝軟體更新，系統將只會下載完成更新所需的元件（而非下載整套 OS），以改善網路效率。此外，可以在執行 OS X Server 快取服務的區域網路伺服器上快取軟體更新，這樣 iOS 裝置無須連接 Apple 伺服器便可取得必要的更新資料。

在 iOS 升級期間，iTunes（若採用 OTA 軟體更新方式，則為裝置本身）會連接 Apple 安裝授權伺服器，並向其傳送以下資料：要安裝之安裝套件中的各部分加密測量值列表（如 LLB、iBoot、核心及 OS 映像檔）、隨機反重播的值（隨機數）以及裝置的唯一識別碼（ECID）。

授權伺服器會將提供的測量值列表與允許安裝的版本進行比較，若找到相符項目，便會將 ECID 加入到測量值中並對結果進行簽署。伺服器會將完整的一組已簽署資料傳遞至裝置，這是升級程序的一部分。加入 ECID 可為要求的裝置「個人化」授權作業。藉由只對已知的測量值授權和簽署，伺服器可確保更新的內容與 Apple 所提供的完全相同。

啟動時的信任鏈結評估會驗證簽名是否來自 Apple，並確認從磁碟載入的項目測量值在結合裝置 ECID 後，是否與該簽名所涵蓋的內容相符。

這些步驟可確保授權是針對特定裝置進行，並且舊版 iOS 無法從一部裝置拷貝到另一部裝置。隨機數可阻止攻擊者儲存伺服器的回應，並阻止使用該回應來破壞裝置或以其他方式竄改系統軟體。

Secure Enclave

Secure Enclave 是 Apple A7 或更新版本 A 系列處理器精心打造的副處理器。它獨立於應用程式處理器之外，並利用本身的安全啟動和個人化軟體更新。它為「資料保護」密鑰管理提供所有加密操作，即使在核心遭到入侵的情況下，仍會維護「資料保護」的完整性。

Secure Enclave 使用加密記憶體，並包含一個硬體亂數產生器。其微核心是以 L4 系列為基礎，並由 Apple 加以修改。Secure Enclave 與應用程式處理器之間的通訊會被隔離到一個以中斷驅動的信箱和共享的記憶體資料緩衝區。

每個 Secure Enclave 在製作期間皆已提供其本身的 UID（唯一識別碼），此 UID 無法由系統的其他部分取用，且 Apple 亦無其相關資訊。當裝置啟動時，會製作一個臨時密鑰，此密鑰與 UID 配合使用，用來對 Secure Enclave 的裝置記憶體空間部分進行加密。

此外，由 Secure Enclave 儲存到檔案系統的資料會藉由 UID 搭配使用的密鑰和反重播計數器來進行加密。

Secure Enclave 負責處理來自 Touch ID 感應器的指紋資料，確定是否有與登記之指紋相符合的指紋資料，然後代表使用者啟用存取或購買。處理器與 Touch ID 感應器之間的通訊是透過序列週邊介面匯流排來執行。處理器會將資料轉送至 Secure Enclave，但處理器本身無法讀取這些資料。資料會藉由區段密鑰進行加密與認證，該密鑰透過為 Touch ID 感應器和 Secure Enclave 佈建的裝置共享密鑰來進行交涉。區段密鑰的交換會針對雙方使用 AES 密鑰封裝，並提供一個用來建立工作階段密鑰和使用 AES-CCM 傳輸加密的隨機密鑰。

Touch ID

Touch ID 是指紋感應系統，有助於更快、更輕鬆地對裝置進行安全性的存取。此技術可從任何角度來讀取指紋，隨著感應器每次使用時識別出其他重疊的節點而持續擴大指紋圖，逐漸提高對使用者指紋辨識的能力。

Touch ID 讓使用更長、更複雜的密碼變得更為實際，因為使用者無須經常輸入密碼。Touch ID 也克服了以密碼方式鎖定的不便性，它並不會取代密碼鎖定的機制，而是允許在精心設計的範圍和時間限制內，安全地取用裝置。

Touch ID 和密碼

若要使用 Touch ID，使用者必須設定其裝置以要求密碼來將其解鎖。當 Touch ID 掃描並可識別已登記的指紋時，裝置便會自動解鎖，使用者無須輸入裝置密碼。使用者可以隨時使用密碼來取代 Touch ID，並且在以下情況下必須使用密碼：

- 裝置剛剛開機或重新啟動。
- 裝置未解鎖的時間超過 48 小時。
- 裝置收到了遠端鎖定指令。
- 嘗試五次後未能成功符合指紋。
- 設定 Touch ID 或為其登記新指紋時。

當 Touch ID 啟用後，裝置會在按下「睡眠/喚醒」按鈕時立即鎖定。在只使用密碼的安全機制下，許多使用者會設定解鎖的寬限期，以避免每次使用裝置時必須輸入密碼。有了 Touch ID，裝置每次進入睡眠時便會鎖定，而每次喚醒時都需要掃描指紋（也可以選擇輸入密碼）。

使用者可以訓練 Touch ID 識別多達五個不同的指紋。對於已登記的指紋，該指紋與其他人的指紋隨機出現相符的機率為五萬分之一。不過，Touch ID 只允許五次不成功的指紋比對嘗試，之後使用者必須輸入密碼才能取得取用權限。

Touch ID 的其他用途

使用者也可對 Touch ID 進行設定，以核准從 iTunes Store、App Store 和 iBooks Store 購買項目，因此使用者無須輸入 Apple ID 密碼。當他們選擇授權某項購買項目時，裝置和商店之間會交換認證代號。代號和加密隨機數都會保留在 Secure Enclave 中。隨機數透過由所有裝置和 iTunes Store 共享的 Secure Enclave 密鑰進行簽署。

Touch ID 亦可搭配 Apple Pay 使用，此為 Apple 的安全付款方式。如需更多資訊，請參閱本文件的 Apple Pay 章節。

此外，第三方 App 可使用系統提供的 API 來要求使用者使用 Touch ID 或密碼進行認證。App 只會收到認證是否成功的通知；其無法取用 Touch ID 或與已登記之指紋相關的資料。

鑰匙圈也可使用 Touch ID 進行保護，只有透過指紋比對或裝置密碼，Secured Enclave 才會將其釋出。App 開發者也可以透過 API 來確認密碼已由使用者設定，因此可使用 Touch ID 來認證或解鎖鑰匙圈項目。

有了 iOS 9，開發者可要求 Touch ID 的 API 操作不依賴應用程式密碼或裝置驗證碼。藉由可取得登記指紋的代表權，這樣便可讓 Touch ID 在對安全性有高度要求的 App 中作為第二個驗證因子。

Touch ID 安全性

只有當「主畫面」按鈕周圍的電容金屬環偵測到手指觸摸時，指紋感應器才會作用，進而觸發進階成像陣列來掃描手指，並將掃描結果傳送到 Secure Enclave。

光柵掃描結果會暫時存放在 Secure Enclave 加密的記憶體中，同時系統會對其進行向量化處理以便分析，然後便會刪除相關資料。此分析利用皮下紋路流向角度的對應，這是一種有損性的程序，會在分析完成後刪除用來重建使用者實際指紋的精細資料。最後產生的節點圖會以一種只能由 Secure Enclave 讀取的加密格式進行儲存，其中不含任何身份資訊，且絕不會傳送給 Apple 或備份至 iCloud 或 iTunes。

Touch ID 如何解鎖 iOS 裝置

若 Touch ID 已關閉，在裝置鎖定時，保留在 Secure Enclave 中資料保護「完整」類別的密鑰將會被捨棄。除非使用者輸入密碼來解鎖裝置，否則便無法取用該類別中的檔案和鑰匙圈項目。

若 Touch ID 已開啟，在裝置鎖定時，這些密鑰不會被捨棄，而是透過提供給 Secure Enclave 中的 Touch ID 子系統的密鑰進行封裝。當使用者嘗試解鎖裝置時，若 Touch ID 可以識別使用者的指紋，它便會提供密鑰來解除封裝「資料保護」密鑰，裝置便會解鎖。此流程藉由要求「資料保護」和 Touch ID 子系統協同合作來解鎖裝置，因此提供了額外的保護。

若裝置重新開機，使用 Touch ID 解鎖裝置時所需的密鑰便會喪失，而在 48 小時後或 Touch ID 識別嘗試失敗五次後，該密鑰也會被 Secure Enclave 捨棄。

加密與資料保護

安全啟動鏈、程式碼簽署及執行階段程序安全性都有助於確保，只有受信任的程式碼與 App 可在裝置上執行。iOS 還有其他加密與資料保護功能可保護使用者資料的安全，即使是安全性基礎架構的其他部分遭到破壞（例如，在裝置上發生未經授權的修改）。這對於使用者與 IT 管理者都大有助益，可隨時保護個人與企業的資訊，並提供裝置遭竊或遺失時，於遠端立即完全清除的方式。

硬體安全性功能

在行動裝置上，速度與能源效率十分重要。加密操作非常複雜，若在設計與導入時未考慮到這兩個重要因素，可能會帶來一些效能或電池續航力的問題。

每部 iOS 裝置都配備專屬的 AES 256 加密引擎，其內建於快閃儲存空間與主系統記憶體間的 DMA 路徑中，可讓檔案加密具備高度效率。

裝置的唯一識別碼（UID）與裝置群組識別碼（GID）是 AES 256 位元的密鑰，該密鑰已在製作過程中融入（UID）或編譯（GID）到應用程式處理器和 Secure Enclave 中。任何軟體或韌體都無法直接讀取；只能將 UID 或 GID 用作密鑰，以便查看由矽晶片中建置的專屬 AES 引擎所執行加密或解密操作的結果。此外，Secure Enclave 的 UID 和 GID 只能由 Secure Enclave 專屬的 AES 引擎使用。每部裝置的 UID 都是唯一的，Apple 或其他供應商都不會有所記錄。GID 對同一類別的裝置（例如，使用 Apple A8 處理器的所有裝置）的所有處理器是通用的，它可用於非安全性關鍵的工作，例如在安裝與回復期間遞送系統軟體時。將這些密鑰整合到矽晶片中可防止它們遭到竄改或略過，或在 AES 引擎外部進行取用。UID 和 GID 也無法透過 JTAG 或其他除錯介面來使用。

UID 允許資料以加密方式與特定裝置相連結。例如，用來保護檔案系統的密鑰階層便包含 UID，因此若將記憶體晶片實際從一部裝置移至另一部裝置，檔案則無法取用。UID 與裝置上的任何其他識別碼並不相關。

除了 UID 和 GID 外，所有其他加密編譯密鑰都是由系統的亂數產生器（RNG）使用以 CTR_DRBG 為基礎的演算法製作。系統熵是在啟動期間從時間變化以及裝置啟動後從中斷時間來產生的。在 Secure Enclave 內產生的密鑰會使用其真正的硬體亂數產生器，透過以 CTR_DRBG 處理的多個環型振盪器製作而成。

安全清除儲存的密鑰與產生它們具有同等重要性。在快閃儲存空間上執行此項操作尤其具有挑戰性，因為耗損平衡（wear-leveling）可能意味著需要清除多份資料拷貝。為了解決此問題，iOS 裝置內建一種專門用於安全清除資料的功能，稱為 Effaceable Storage。此功能利用基礎儲存技術（如 NAND）直接在極低的層次上進行定址和清除小量區塊。

清除所有內容和設定

「設定」中的「清除所有內容和設定」選項會清除 Effaceable Storage（可抹除儲存空間）中的所有密鑰，透過加密方式讓裝置上的所有使用者資料無法使用。因此，若要將裝置送給其他人或送修，此選項便是移除所有個人資訊的理想方式。重要事項：除非裝置已備份，否則請勿使用「清除所有內容和設定」選項，因為清除的資料無法恢復。

檔案資料保護

除了 iOS 裝置內建的硬體加密功能，Apple 也使用名為「資料保護」的技術，來進一步保護裝置上快閃記憶體中儲存的資料。「資料保護」可讓裝置回應如來電之類的常見事件，也可以對使用者資料啟用較高層次的加密。「訊息」、「郵件」、「行事曆」、「聯絡資訊」、「照片」和「健康」資料值等主要系統 App 預設都會使用「資料保護」，而安裝於 iOS 7 或更新版本上的第三方 App 可自動獲得此項保護措施。

「資料保護」是透過建構和管理密鑰階層來完成導入，並建立在每部 iOS 裝置的硬體加密技術上。「資料保護」藉由將每個檔案指定給某個類別，進而對檔案逐一進行控制；可取用性則取決於該類密鑰是否已解鎖。

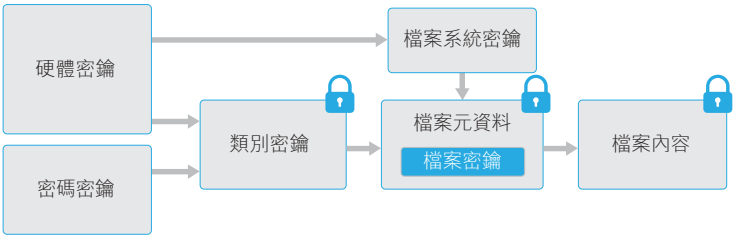
架構概覽

每次在資料分割區上製作檔案時，「資料保護」都會製作一個新的 256 位元密鑰（「檔案專屬」密鑰），並將其提供給硬體 AES 引擎，此引擎會使用該密鑰採用 AES CBC 模式對寫入快閃記憶體的檔案進行加密。（在配備 A8 處理器的裝置上，會使用 AES-XTS。）初始化向量（IV）使用檔案的區塊偏移量進行計算，它使用檔案專屬密鑰的 SHA-1 雜湊進行加密。

依據每個檔案的可取用性情況，檔案專屬密鑰會使用其中一個類別密鑰進行封裝。就像所有其他封裝一樣，這是使用 NIST AES 密鑰封裝、依據 RFC 3394 來執行。封裝的檔案專屬密鑰會儲存在檔案的元資料中。

當打開檔案時，系統會使用檔案系統密鑰來解密其元資料，以呈現封裝的檔案專屬密鑰以及表示其保護類別的記號。檔案專屬密鑰會使用類別密鑰來解除封裝，然後提供給硬體 AES 引擎，該引擎會在從快閃記憶體中讀取檔案時，對檔案進行解密。所有封裝檔案密鑰的處理作業會在 Secure Enclave 中進行；檔案密鑰永遠不會直接提供給應用程式處理器。在開機時，Secure Enclave 會與 AES 引擎協調臨時密鑰。當 Secure Enclave 解除封裝檔案密鑰時，它們會臨時密鑰來重新封裝，並傳送回應用程式處理器。

檔案系統中所有檔案的元資料都使用隨機密鑰進行加密，該密鑰是在首次安裝 iOS 或使用者清除裝置時製作而成。檔案系統密鑰則儲存在 Effaceable Storage 中。因為該密鑰儲存在裝置上，因此它不是用來維護資料的機密性，而是可以視需求快速清除（由使用者使用「清除所有內容和設定」選項來清除，或者由使用者或管理者從行動裝置管理（MDM）伺服器、Exchange ActiveSync 或 iCloud 發出遠端清除指令來清除）。以此方式清除密鑰將會透過加密的方式讓裝置上的所有檔案無法取用。



檔案的內容使用檔案專屬密鑰進行加密，該密鑰使用類別密鑰封裝並儲存在檔案的元資料中，檔案元資料接著又使用檔案系統密鑰進行加密。類別密鑰使用硬體 UID 取得保護，而某些類別則透過使用者密碼取得保護。此階層架構同時提供了彈性與效能。例如，更改檔案的類別只需要重新封裝其檔案專屬密鑰，更改密碼只需要重新封裝類別密鑰。

密碼考量事項

若輸入較長的純數字密碼，鎖定畫面上會顯示數字鍵盤，而非完整鍵盤。與較短的英數字密碼相比，較長的數字密碼更便於輸入，而且可提供類似的安全性。

密碼

藉由設定裝置密碼，使用者可以自動啟用「資料保護」。iOS 支援六位數、四位數和任意長度的英數字元密碼。除了用於解鎖裝置外，密碼還為特定加密密鑰提供熵。這表示攻擊者即使拿到裝置，在沒有密碼的情況下也無法取用特定保護類別中的資料。

密碼與裝置的 UID 搭配使用，因此暴力密碼破解只能在受攻擊的裝置上進行。因此，iOS 系統使用較大的反覆運算來延緩每次的嘗試。反覆運算計數已經過測定，每次嘗試會耗時約 80 毫秒。這意味著嘗試 6 個字元的英數字元密碼（小寫字母和數字）的所有組合將會耗時 5 年半的時間。

使用者密碼的強度越大，加密密鑰就越堅固。Touch ID 可用來提升這樣的因果關係，因為它可以讓使用者製作一個比實用密碼安全性更高的密碼。這增加了對用於「資料保護」的加密密鑰進行保護的密碼強度，而且不會對一天中多次解鎖 iOS 裝置的使用者體驗產生負面影響。

密碼嘗試次數間的延遲

嘗試次數	強制延遲
1-4	無
5	1 分鐘
6	5 分鐘
7-8	15 分鐘
9	1 小時

為了進一步阻止暴力密碼的破解攻擊，系統會延長在鎖定畫面上輸入無效密碼後的延遲時間。若「設定」>「Touch ID 與密碼」>「清除資料」已啟用，裝置將會在嘗試輸入密碼錯誤 10 次後自動清除。此設定還可透過行動裝置管理（MDM）和 Exchange ActiveSync 作為管理規則，並可設定為較低的臨界值。

在配備 A7 或後續 A 系列處理器的裝置上，Secure Enclave 會強制執行延遲。若裝置在定時延遲期間重新啟動，延遲仍會強制執行，但計時器會從目前期間重新開始。

資料保護類別

在 iOS 裝置上製作新檔案時，用來製作的 App 會替檔案指定一個類別。每個類別使用不同的規則來決定資料何時可供取用。基本類別和規則會在下面的章節中說明。

完整保護

（`NSFileProtectionComplete`）：類別密鑰會使用從使用者密碼和裝置 UID 所衍生的密鑰加以保護。使用者鎖定裝置後不久（若「需要密碼」設定為「立即」時，則為 10 秒），已解密的類別密鑰便會被捨棄，讓此類別中的所有資料只有在使用者再次輸入密碼或使用 Touch ID 解鎖裝置時，才可以取用。

未打開檔案的保護

（`NSFileProtectionCompleteUnlessOpen`）：有些檔案可能需要在裝置鎖定時寫入。其中一個不錯的例子是電子郵件的附件在背景下載。此行為是藉由使用非對稱橢圓曲線加密技術（Curve25519 的 ECDH）來達成。常見的檔案專屬密鑰則是使用 One-Pass Diffie-Hellman Key Agreement（如 NIST SP 800-56A 中所述）所衍生的密鑰加以保護。

該協議的臨時公用密鑰與封裝的檔案專屬密鑰一起儲存。KDF 是鏈結密鑰衍生函數（Approved Alternative 1），如 5.8.1 of NIST SP 800-56A 中所述。AlgorithmID 已忽略。PartyUInfo 和 PartyVInfo 分別是臨時靜態公用密鑰。SHA-256 則用於雜湊函數。檔案一旦關閉，檔案專屬密鑰便會從記憶體中清除。若要再次打開檔案，系統會使用「未打開檔案的保護」類別的專用密鑰和檔案的臨時公用密鑰來重新製作共享密鑰；其雜湊會用來解除封裝檔案專屬密鑰，該密鑰接著會被用來解密檔案。

首次使用者認證前的保護

（`NSFileProtectionCompleteUntilFirstUserAuthentication`）：此類別與「完整保護」類別的行為方式相同，只是在鎖定裝置時，已解密的類別密鑰不會從記憶體中移除。此類別中的保護和桌上型電腦完整卷宗的加密有類似的屬性，可防止資料因重新啟動而遭到攻擊。對於未指定至「資料保護」類別的所有第三方 App 資料，這是預設類別。

無保護

（`NSFileProtectionNone`）：此類別密鑰僅受到 UID 的保護，並且儲存在 Effaceable Storage 中。因為解密該類別中之檔案所需的所有密鑰都儲存在裝置上，因此加密的唯一好處就是可以進行快速的遠端清除。即使未對檔案指定「資料保護」類別，此檔案仍會以加密形式儲存（就像 iOS 裝置上的所有資料一樣）。

鑰匙圈資料保護

許多 App 需要處理密碼和其他簡短但較為敏感的資料，如密鑰和登入 Token。iOS 鑰匙圈提供了儲存這些項目的安全方式。

鑰匙圈是以儲存在檔案系統中的 SQLite 資料庫的方式導入。只有一個資料庫；securityd 服務程式會決定哪些鑰匙圈項目可被每個處理程序或 App 取用。鑰匙圈取用 API 會產生對服務程式的呼叫，進而查詢 App 的「keychain-access-groups」、「application-identifier」和「application-group」權限。取用群組允許在 App 間共享鑰匙圈項目，而非將取用權限限制為單一處理程序。

鑰匙圈項目只能在來自同一開發者的 App 間共享。管理方式是要求第三方 App 使用取用群組，並使用透過 iOS Developer Program（iOS 開發者計畫）或透過應用程式群組來為其分配前置碼。對前置碼的要求和應用程式群組唯一性，是透過程式碼簽署、佈建描述檔和 iOS Developer Program（iOS 開發者計畫）強制執行。

系統用來保護鑰匙圈項目的類別結構，與檔案「資料保護」中使用的類別結構相似。這些類別具有與檔案「資料保護」類別相同的行為，但使用的密鑰不同，所屬 API 的名稱也不同。

可用性	檔案資料保護	鑰匙圈資料保護
未鎖定時	NSFileProtectionComplete	kSecAttrAccessibleWhenUnlocked
鎖定時	NSFileProtectionCompleteUnlessOpen	N/A
首次解鎖後	NSFileProtectionCompleteUntilFirstUserAuthentication	kSecAttrAccessibleAfterFirstUnlock
總是	NSFileProtectionNone	kSecAttrAccessibleAlways
密碼已啟用	N/A	kSecAttrAccessible-WhenPasscodeSetThisDeviceOnly

利用背景重新整理服務的 App 可將 `kSecAttrAccessibleAfterFirstUnlock` 用於背景更新期間需要取用的鑰匙圈項目。

類別 `kSecAttrAccessibleWhenPasscodeSetThisDeviceOnly` 的行為與 kSecAttrAccessibleWhenUnlocked 相同，然而只有在使用驗證碼設定裝置時才能使用。此類別只存在於系統 Keybag 中；他們不會同步到 iCloud 鑰匙圈；不會進行備份；也不會納入託管 Keybag 中。若密碼遭移除或重置，類別密鑰便會被捨棄，這些項目也變得無法使用。

其他鑰匙圈類別都有對應的「僅限本裝置」項目，其在備份期間從裝置拷貝時一律受到 UID 保護，因此若回復到其他裝置，將會無法使用。

鑰匙圈項目的元件

除了取用群組，每個鑰匙圈項目還包含管理元資料（如「製作時間」和「上次更新時間」的時間戳記）。

其也包含屬性的 SHA-1 雜湊，用來查詢項目（如帳號和伺服器名稱），以在無須解密每個項目的情況下即可進行查找。最後，它還包含加密資料，其中包括下列項目：

- 版本編號
- 連線權限控制列表（ACL）資料
- 指出項目所屬保護類別的值
- 以保護類別密鑰加以封裝的檔案專屬密鑰
- 描述項目的屬性字典（傳遞到 SecItemAdd），編碼為二進位 plist 並使用檔案專屬密鑰加密

加密為 GCM（Galois/ Counter Mode）模式下的 AES 128；取用群組會納入屬性中，並受加密期間計算的 GMAC 標記保護。

Apple 依據所保護資訊的類型和 iOS 需要這些資訊的時間來選擇鑰匙圈類別，妥善地在安全性與可用性之間取得平衡。例如，VPN 憑證必須隨時可供使用，這樣裝置才能保持連續的連線，但其被歸類為「不可遷移」，因此無法將其移至另一部裝置。

對於 iOS 所製作的鑰匙圈項目，將會強制執行下列類別保護：

項目	可取用
Wi-Fi 密碼	首次解鎖後
郵件帳號	首次解鎖後
Exchange 帳號	首次解鎖後
VPN 密碼	首次解鎖後
LDAP、CalDAV、CardDAV	首次解鎖後
社群網路帳號代號	首次解鎖後
Handoff 廣播加密密鑰	首次解鎖後
iCloud 代號	首次解鎖後
家人共享密碼	未鎖定時
「尋找我的 iPhone」代號	總是
語音信箱	總是
iTunes 備份	解鎖時，不可遷移
Safari 密碼	未鎖定時
Safari 書籤	未鎖定時
VPN 憑證	總是，不可遷移
Bluetooth® 密鑰	總是，不可遷移
Apple 推播通知服務代號	總是，不可遷移
iCloud 憑證和專用密鑰	總是，不可遷移
iMessage 密鑰	總是，不可遷移
由設定描述檔所安裝的憑證和專用密鑰	總是，不可遷移
SIM PIN	總是，不可遷移

鑰匙圈存取控制

鑰匙圈可使用連線權限控制列表（ACL）來設定可取用性和認證需求的規則。項目可以建立條件來要求使用者操作，方法是指定使用 Touch ID 或輸入裝置密碼進行認證，否則無法存取。ACL 會在 Secure Enclave 中進行評估，只有符合其指定的限制條件時，才匯出到核心中。

取用 Safari 儲存的密碼

iOS App 可以與 Safari 儲存的鑰匙圈項目互動，使用下列兩個 API 進行密碼自動填寫：

- `SecRequestSharedWebCredential`
- `SecAddSharedWebCredential`

只有 App 開發者和網站管理者同時核准且使用者同意後，才會授予存取權限。App 開發者藉由在其 App 中包含授權，讓系統得知他們需要取用 Safari 已儲存的密碼。授權中列出了相關網站的完全合格的網域名稱。網站必須將檔案放在其伺服器上，並在其中列出已核准的 App 的唯一 App 識別碼。在安裝帶有 com.apple.developer.associated-domains 權限的 App 後，iOS 向每個列出的網站發出 TLS 要求來要求檔案 /apple-app-site-association。若檔案中列出了要安裝之 App 的識別碼，iOS 才會將網站和 App 標示為具有信任關係。只有在具有信任關係的情況下，才會呼叫這兩個 API 並向使用者發出提示，使用者同意後，密碼才會核發給 App，或者被更新或刪除。

Keybag

檔案和鑰匙圈「資料保護」類別的密鑰會收集在 Keybag 中加以管理。iOS 使用下列四種 Keybag：系統、備份、託管和「iCloud 備份」。

系統 Keybag 是裝置一般操作中使用的封裝類別密鑰的儲存位置。例如，輸入密碼後，會從系統 Keybag 中載入 `NSFileProtectionComplete` 密鑰並解除封裝。它是儲存在「無保護」類別中的二進位 plist，但其內容是使用 Effaceable Storage 中儲存的密鑰來加密。為了對 Keybag 提供更高的安全性，使用者每次更改密碼時，系統都會清除並重新產生此密鑰。AppleKeyStore 核心延伸功能會管理系統 Keybag，並可用於查詢裝置的鎖定狀態。只有在系統 Keybag 中的所有類別密鑰都可取用且已成功解除封裝，它才會報告裝置已解鎖。

備份 Keybag 是在 iTunes 進行加密時製作，其儲存在裝置進行備份的電腦中。新 Keybag 是使用一組新的密鑰製作而成，備份的資料會以這些新密鑰來重新加密。如前面所述，不可遷移的鑰匙圈項目仍會使用 UID 衍生的密鑰加以封裝，以使其可以回復到最初備份它們的裝置，但在其他裝置上則無法取用。

Keybag 使用 iTunes 中設定的密碼來加以保護，其執行了 10,000 次 PBKDF2 的反覆運算。雖然反覆運算的次數很多，但 Keybag 並未與特定裝置相連結，因此理論上可嘗試在多部電腦上對備份 Keybag 進行暴力密碼破解攻擊。而安全性夠高的密碼可以降低此威脅。

若使用者選擇不加密 iTunes 備份，那麼無論備份檔案屬於哪一種「資料保護」類別，備份檔案都不會加密，但鑰匙圈仍會使用 UID 衍生的密鑰獲得保護。這就是只有在設定備份密碼時，才能將鑰匙圈項目遷移到新裝置的原因。

託管 Keybag 用於 iTunes 同步和 MDM。此 Keybag 允許 iTunes 執行備份和同步，使用者無須輸入密碼，它還允許 MDM 伺服器遠端清除使用者密碼。它儲存在用來與 iTunes 進行同步的電腦，或者管理裝置的 MDM 伺服器上。

託管 Keybag 改善了裝置同步期間的使用者體驗，在該期間可能需要取用所有類別的資料。當使用密碼鎖定的裝置首次連接到 iTunes 時，會提示使用者輸入密碼。然後，裝置會製作託管 Keybag，其中包含的類別密鑰與裝置上使用的完全相同，該 Keybag 由新產生的密鑰加以保護。託管 Keybag 與用於保護它的密鑰被分割到裝置和主機或伺服器上，其資料以「首次使用者認證前的保護」類別儲存在裝置上。這就是重新啟動後首次使用 iTunes 進行備份之前，必須輸入裝置密碼的原因。

在 OTA 軟體更新的情況下，初始化更新時，系統會提示使用者輸入密碼。這會用來安全地建立「一次性解鎖代號」，其會在更新後解鎖系統 keybag。若未輸入使用者的驗證碼，便無法產生此代號，且若使用者驗證碼有所更改，任何先前產生的代號則會取消驗證。

「一次性解鎖代號」適用於軟體更新的自行或非自行安裝。它們會使用來自 Secure Enclave 中單調計數器目前值所衍生的密鑰、keybag 的 UUID 和 Secure Enclave 的 UID 來進行加密，

在 SEP 中導入「一次性解鎖代號」計數器會取消驗證任何現有的代號。在使用代號時、重新啟動裝置的第一次解鎖後、（由使用者或系統）取消軟體更新時或代號的規則計時器到期時，計數器都會遞增。

自行軟體更新的「一次性解鎖代號」會在 20 分鐘後到期。此代號可從 Secure Enclave 輸出，並寫入 Effaceable Storage 中。若裝置在 20 分鐘內未重新開機，規則計時器會遞增計數器。

針對非自行軟體更新（即當使用者收到更新通知時，選擇「稍後安裝」的設定），應用程式處理器會在 Secure Enclave 中保留有效的「一次性解鎖代號」，最長達 8 小時。在該時間後，規則計時器便會遞增計數器。

「iCloud 備份」Keybag 與備份 Keybag 類似。此 Keybag 中的所有類別密鑰都是非對稱式的（與「未打開檔案的保護」資料保護類別一樣，使用 Curve25519），因此可以在背景執行 iCloud 備份。對於「無保護」以外的所有「資料保護」類別，加密的資料會從裝置中讀取並傳送至 iCloud。對應的類別密鑰會以 iCloud 密鑰加以保護。鑰匙圈類別的密鑰會使用 UID 衍生的密鑰進行封裝，方式與未加密的 iTunes 備份相同。非對稱式 Keybag 也可用於「iCloud 鑰匙圈」其相關鑰匙圈恢復作業的備份中。

安全性認證和方案

加密驗證（FIPS 140-2）

iOS 中的加密模組經過驗證，符合美國聯邦資訊處理標準（FIPS）140-2 第 1 級規範（自 iOS 6 起的每次發行）。iOS 9 中的加密模組與 iOS 8 中的相同，但隨著每次發行，Apple 會提交這些模組進行重新驗證。此計畫會針對適當使用 iOS 加密服務的 Apple App 和第三方 App，驗證加密操作的完整性。

Common Criteria Certification (ISO 15408)

Apple 已依據 Common Criteria Certification (CCC) 計畫執行 iOS 認證。前兩項認證根據 Mobile Device Fundamental Protection Profile v2.0 (MDFPP2) 與 VPN IPSecPP1.4 Client Protection Profile (VPNIPSecPP1.4) 目前皆有效。Apple 在 International Technical Community (ITC) 中扮演積極角色，開發目前仍無法取得的「保護描述檔」（PPs），致力於評估關鍵行動安全性技術。Apple 會依據目前可取得之新版本與更新版本的 PPs 來評估和執行認證。

Commercial Solutions for Classified (CSfC)

在合適的情況下，Apple 也已提交 iOS 平台與各種服務，以納入 Commercial Solutions for Classified (CSfC) 計畫元件列表中。特別是行動平台的 iOS 和 IPSec VPN 用戶端（僅 IKEv2 Always-On VPN）的 IKEv2。因為 Apple 平台與服務具備 Common Criteria Certification，他們也會依據 CSfC 計畫元件列表來提交以便納入。

安全性設定指南

Apple 已與世界各地的政府協同合作，提供指導與建議，以維護一個更安全的環境（即所謂的「裝置強化」（device hardening））。這些指南會針對在 iOS 中設定與使用功能，提供已定義且經過審核的資訊，以增強保護。

如需 iOS 安全性認證、驗證及指引的相關資訊，請參閱 support.apple.com/kb/HT202739?viewlocale=zh_TW。

App 安全性

App 是現代行動安全架構最關鍵的要素之一。雖然 App 可顯著提高使用者的生產力，但若處理不當，也可能對系統安全性、穩定性和使用者資料產生負面影響。

有鑑於此，iOS 提供了多重保護來確保 App 經過簽署和驗證，且以 Sandbox 技術限制，進而保護使用者資料。這些要素為 App 提供了穩定且安全的平台，讓成千上萬的開發者能夠在 iOS 上提供數十萬款的 App，而不會影響系統的完整性。使用者可以在其 iOS 裝置上取用這些 App，無須過度擔心病毒、惡意軟體或未經授權的攻擊。

App 程式碼簽署

一旦 iOS 核心啟動後，它將控制可執行哪些使用者程序和 App。為了確保所有 App 均來自核准的已知來源且未被竄改，iOS 會要求所有可執行的程式碼均使用 Apple 核發的憑證進行簽署。裝置所隨附的 App（如「郵件」和 Safari）則由 Apple 簽署。第三方 App 也必須使用 Apple 核發的憑證進行驗證和簽署。強制性程式碼簽署將信任鏈的概念從作業系統延伸至 App，可防止第三方 App 載入未簽署的程式碼資源，或使用自行修改的程式碼。

若要在 iOS 裝置上開發並安裝 App，開發者必須向 Apple 註冊並加入 iOS Developer Program（iOS 開發者計畫）。Apple 會先驗證每位開發者（無論是個人或企業）的真實身份，然後再核發憑證。開發者可使用該憑證對 App 進行簽署，並將其提交至 App Store 進行發佈。因此，App Store 中的所有 App 都是由身份可識別的個人或組織提交的，藉此阻止製作惡意 App。這些 App 都經過 Apple 嚴格審核，以確保它們可以如所述方式執行，且沒有明顯的程式錯誤或其他問題。除了已討論過的技術外，此挑選過程還會讓客戶對所購買的 App 的品質更加放心。

iOS 允許開發者將架構嵌入 App 中，讓其可被 App 本身使用，也可被 App 內嵌入的延伸功能使用。為了保護系統並防止其他 App 在其位址空間中載入第三方的程式碼，系統將為啟動時程序所連結的所有動態資源庫執行程式碼簽名驗證。此驗證過程受過團隊識別碼（Team ID）來達成，該識別碼擷取自 Apple 核發的憑證。團隊識別碼是 10 個字元的英數字元字串，例如 1A2B3C4D5F。程式可透過連結到隨系統發佈的任何資源庫平台，或其程式碼簽名中具有相同團隊的資源庫平台來成為主要執行檔。因為作為系統一部分發佈的可執行檔不具有團隊識別碼，所以它們只能連結到隨系統本身發佈的資源庫。

企業也可以編寫供組織內部使用的企業內部 App，並分發給員工。企業和組織可以使用 D-U-N-S 編號申請加入 Apple Developer Enterprise Program（ADEP，Apple 開發者企業計畫）。Apple 會在驗證申請者的身份和資格後核准其請求。一旦組織成為 ADEP 的成員，便可以註冊以獲得一個「佈建描述檔」，該描述檔允許企業內部 App 在其授權的裝置上執行。使用者必須安裝「佈建描述檔」才能執行企業內部 App。這可以確保只有組織要求的使用者能夠將 App 載入到其 iOS 裝置上。透過 MDM 安裝的 App 會間接獲得信任，因為組織與裝置間的關係已建立。在其他情況下，使用者必須在「設定」中核准 App 的「佈建描述檔」。組織可以限制使用者，不允許其核准來自未知開發者的 App。第一次啟動任一企業 App 時，裝置必須從 Apple 收到允許執行 App 的肯定確認。

與其他行動平台不同，iOS 不允許使用者安裝來自網站可能有惡意性質且未經簽署的 App，或者執行不受信任的程式碼。執行時，會在載入所有可執行記憶體頁面後對其進行程式碼簽名檢查，以確保 App 自安裝或上次更新後未遭修改過。

執行階段程序安全性

一旦確認 App 來自核准的來源後，iOS 會強制執行相關的安全措施，以防止其危害其他 App 或系統的其他部分。

所有第三方的 App 均會以 Sandbox 技術限制，因此在存取其他 App 儲存的檔案或對裝置進行更動時會受到限制。這樣可以防止 App 收集或修改其他 App 儲存的資訊。每個 App 都有唯一的主目錄來存放其檔案，主目錄是在安裝 App 時隨機指定的。如果第三方的 App 需要存取除了本身資訊以外的其他資訊，只能透過 iOS 明確提供的服務來執行。

系統檔案和資源也會與使用者的 App 保持區隔。iOS 的大部分操作與所有第三方的 App 一樣，以非特殊權限使用者「mobile」的身份執行。整個作業系統分割區都裝載為唯讀。不必要的工具（如遠端登入服務）並未包含在系統軟體中，並且 API 不允許 App 提升自己的特殊權限來修改其他 App 或 iOS 本身。

系統使用宣告的授權來控制第三方 App 對使用者資訊與功能（如 iCloud）和延伸功能的存取權。授權是簽署到 App 中的成對密鑰值，允許對執行階段因素以外的內容（如 unix 使用者 ID）進行認證。授權已經過數位簽署，因此無法更改。系統 App 和服務程式廣泛使用授權來執行特定權限的操作，如果不使用授權，則需要以 root 使用者身份執行程序。這大幅降低了遭入侵的系統應用程式或服務程式提升權限的可能性。

此外，App 只能透過系統提供的 API 來執行背景處理。這讓 App 能夠繼續執行，而不會減緩效能或大幅影響電池續航力。

位址空間配置隨機載入（ASLR）可防止利用記憶體損壞錯誤的攻擊。內建 App 會使用 ASLR 來確保啟動時隨機安排所有記憶體區域。藉由隨機安排可執行檔程式碼、系統資源庫和相關程式設計結構的記憶體位址，便降低遭到許多複雜攻擊的可能性。例如，「return-to-libc」攻擊試圖藉由操縱堆疊和系統資源庫的記憶體位址來誘使裝置執行惡意的程式碼。隨機安排這些項目的位置便大幅增加執行攻擊的難度，尤其是對多部裝置的攻擊。Xcode 作為 iOS 開發環境，可自動編譯啟用了 ASLR 支援的第三方 App。

iOS 使用 ARM 的 Execute Never（XN）功能來提供進一步的保護，該功能會將記憶體頁面標示為不可執行。App 使用標示為可寫入和可執行的記憶體頁面，須符合以下嚴格的控制條件：核心會檢查 Apple 專屬的動態程式碼簽署授權是否存在。即使如此，也只有單個 mmap 呼叫能用於要求一個可執行且可寫入的記憶體頁面（系統為其指定了隨機位址）。Safari 對其 JavaScript JIT 編譯器使用了此功能。

延伸功能

iOS 透過延伸功能來對其他 App 增加功能。延伸功能是具有特殊用途的已簽署可執行二進位程式碼，封裝在 App 內。系統會在安裝時自動偵測延伸功能，並讓使用相符系統的其他 App 使用這些延伸功能。

支援延伸功能的系統區域稱為擴充點。每個擴充點都提供 API，並為該區域強制執行規則。系統依據擴充點特定的比對規則來決定哪些延伸功能可供使用。系統會自動視需要啟動延伸功能程序，並管理這些程序的生命週期。授權可用來限制特定系統應用程式的延伸功能可用性。例如，「今天」顯示方式 Widget 只顯示在「通知中心」內，而共享的延伸功能則只能從「共享」面板中使用。擴充點有「今天」Widget、「分享」、「自定」動作、「照片編輯」、「文件提供程式」和「自定鍵盤」。

延伸功能會在其自己的位址空間中執行。App 與其啟動的延伸功能之間的通訊使用由系統架構所協調的程序間通訊。它們無法存取彼此的檔案或記憶體空間。延伸功能的設計旨在將它們彼此區隔、與其包含的 App 區隔，並且與使用它們的 App 加以區隔。與其他第三方 App 類似，延伸功能也以 Sandbox 技術限制，且擁有的容器會與包含 App 的容器隔開。不過，延伸功能與其容器 App 對隱私控制具有相同的存取權限。因此，若使用者對 App 授予「聯絡資訊」的存取權限，該 App 中嵌入的延伸功能也會獲得此許可權，但由 App 啟動的延伸功能則不具有該許可權。

自定鍵盤是一種特殊類型的延伸功能，因為是由使用者啟用並適用於整個系統。一旦啟用後，該延伸功能將會用於所有的文字欄位，除了密碼輸入和任何安全文字的顯示方式。有鑑於隱私保護的考量，在預設情況下自定鍵盤是在一個十分受限的 Sandbox 中執行，該 Sandbox 會阻止連接網路、阻止代表程序執行網路操作的服務，並阻止可允許延伸功能暗中輸入資料的 API。自定鍵盤的開發者可以要求其延伸功能擁有「開放存取」的權限，讓系統在得到使用者的同意後在預設的 Sandbox 中執行延伸功能。

對於在行動裝置管理中登記的裝置，文件和鍵盤延伸功能將遵循「受管理的打開方式」規則。例如，MDM 伺服器可阻止使用者將受管理 App 中的文件輸出到未受管理的「文件提供程式」，或阻止他們在受管理的 App 中使用未受管理的鍵盤。此外，App 開發者可避免在其 App 中使用第三方的鍵盤延伸功能。

App 群組

指定之開發者帳號所擁有的 App 和延伸功能在設定為「App 群組」的一部分後，便可共享內容。開發者可決定是否在 Apple Developer Portal（Apple 開發者入口網站）上製作適合的群組，並納入想要的 App 和延伸功能。在 App 被設定為「App 群組」的一部分後，便可存取以下內容：

- 磁碟上共享的儲存容器，只要 App 群組內有一個 App 被安裝，它就會一直保留在裝置上
- 共享的偏好設定
- 共享的鑰匙圈項目

Apple Developer Portal（Apple 開發者入口網站）則保證「App 群組 ID」在整個 App 生態系統的是唯一的。

App 中的資料保護

iOS 軟體開發套件（SDK）提供全套 API，讓第三方和企業內部開發者能夠輕鬆地採用「資料保護」功能，協助確保在 App 中享有最高層級的保護。「資料保護」適用於檔案和資料庫 API，包括 NSFileManager、CoreData、NSData 和 SQLite。

「郵件」App（包括附件）、受管理的書籍、Safari 書籤、App 啟動影像和位置資料也將加密儲存，而加密密鑰會以使用者裝置上的密碼進行保護。「行事曆」（不包括附件）、「聯絡資訊」、「提醒事項」、「備忘錄」、「訊息」和「照片」會導入「首次使用者認證前的保護」。

沒有選擇加入某個特定「資料保護」類別且由使用者安裝的 App 預設會接受「首次使用者認證前的保護」。

配件

Made for iPhone、iPod touch, and iPad（MFi）授權計畫允許已審查的配件製造商對 iPod Accessories Protocol（iAP）和必要的支援硬體元件進行存取。

當 MFi 配件使用 Lightning 接頭或透過藍牙與 iOS 裝置進行通訊時，裝置會要求配件使用 Apple 提供的憑證（裝置會對此憑證進行驗證）進行回應，以證明配件經過 Apple 授權。然後，裝置會傳送一個質詢，配件必須使用已簽署的回應來回應。這個過程完全由 Apple 向經過核准的配件製造商提供的自定積體電路處理，而且對於配件本身是透明的。

配件可以要求取用不同的傳輸方式和功能；例如，透過 Lightning 接線取用數位音訊串流，或透過藍牙取用位置資訊。認證積體電路會確保只有經過核准的裝置才能取得對裝置的完全存取權限。如果配件並未提供認證，其存取權限僅限於類比音訊和一小部分的序列（UART）音訊播放控制。

AirPlay 也會利用認證積體電路來驗證接收器已經過 Apple 核准。AirPlay 音訊和 CarPlay 視訊串流使用 MFi-SAP（安全關聯通訊協定），此通訊協定使用 AES-128 在 CTR 模式下對配件和裝置之間的通訊進行加密。臨時密鑰則使用 ECDH 密鑰交換（Curve25519）進行交換，並使用認證電路的 1024 位元 RSA 密鑰進行簽名以作為端到端（STS）通訊協定的一部分。

HomeKit

HomeKit 提供家庭自動化的基礎架構，利用 iCloud 與 iOS 安全性來保護與同步私密資料，無須將其透露給 Apple。

HomeKit 身份

HomeKit 身份與安全性是以 Ed25519 公用-專用密鑰組為基礎。iOS 裝置會為 HomeKit 的每位使用者產生 Ed25519 密鑰組，其會變成他/她的 HomeKit 身份。它會用來認證 iOS 裝置之間以及 iOS 裝置與配件之間的通訊。

密鑰會儲存在「鑰匙圈」中，並僅納入加密的「鑰匙圈」備份。密鑰會使用「iCloud 鑰匙圈」在裝置間進行同步。

與 HomeKit 配件的通訊

HomeKit 配件會產生其自己的 Ed25519 密鑰組，以用於與 iOS 裝置的通訊。若配件被回復成原廠設定，便會產生新的密鑰組。

為了在 iOS 裝置與 HomeKit 配件之間建立關係，密鑰會使用「安全遠端密碼」（3072 位元）通訊協定來交換，使用配件製造商所提供並由使用者於 iOS 裝置上輸入的 8 位數代碼，然後使用 ChaCha20-Poly1305 AEAD 與 HKDF-SHA-512 產生的密鑰來加密。配件的 MFi 認證也會在設定期間進行驗證。

當 iOS 裝置與 HomeKit 配件在使用期間進行通訊時，每個項目會使用上述過程中交換的密鑰來認證另一個項目。每個區段都會使用端到端的通訊協定來建立，並使用以各個區段 Curve25519 密鑰為基礎的 HKDF-SHA-512 衍生密鑰來進行加密。這會同時適用於 IP 型與低功耗藍牙的配件。

本機資料儲存

HomeKit 會在使用者的 iOS 裝置上儲存家庭、配件、場景和使用者的相關資料。儲存的資料會使用自使用者 HomeKit 身份密鑰所衍生的密鑰加上隨機數來進行加密。此外，HomeKit 資料會使用「資料保護」類別的「首次使用者認證前的保護」來儲存。HomeKit 資料只會在加密的備份資料中進行備份，因此舉例來說，未加密的 iTunes 備份便不包含 HomeKit 資料。

裝置和使用者之間的資料同步

您可以使用 iCloud 和「iCloud 鑰匙圈」在使用者的 iOS 裝置間同步 HomeKit 資料。同步期間，HomeKit 資料會使用自使用者 HomeKit 身份與隨機數衍生的密鑰來進行加密。此資料在同步期間會以艱深難懂的資料圖樣處理。最近的資料圖樣會儲存在 iCloud 中以啟用同步，但其不會用於任何其他用途。因為它是使用僅可於使用者 iOS 裝置上取得的密鑰進行加密，因此它的內容在傳輸與 iCloud 儲存期間是無法存取的。

HomeKit 資料也會在同一家庭的多位使用者間進行同步。此處理會使用認證與加密，就像 iOS 裝置與 HomeKit 配件間使用的一樣。認證是以 Ed25519 公用密鑰為基礎，當使用者加入家庭時，便會在裝置間交換這些密鑰。在新使用者加入家庭後，每次進一步的通訊都會使用端到端的通訊協定與各區段的密鑰經過認證和加密。

只有一開始在 HomeKit 中建立家庭的使用者可以加入新的使用者。其裝置會使用新使用者的公用密鑰來設定配件，以便配件可認證和接受來自新使用者的指令。設定 Apple TV 與 HomeKit 的程序會使用與加入其他使用者相同的認證與加密，但會在先前建立家庭的使用者於 Apple TV 上登入 iCloud 且 Apple TV 位於家中時自動執行。

若使用者沒有多部裝置，且並未將其家庭的存取權限授予其他使用者，便不會將 HomeKit 資料同步至 iCloud。

家庭資料與 App

App 對家庭資料的存取權是受使用者的「隱私」設定所控制。使用者會被要求在 App 請求家庭資料時（類似於「聯絡資訊」、「照片」和其他 iOS 資料來源），授予存取權。若使用者核准，App 便可存取房間的名稱、配件名稱、每個配件所在的房間，以及 HomeKit 開發者說明文件中所載明的其他資訊。

Siri

Siri 可用來查詢和控制配件，並可啟動場景。匿名提供給 Siri 的家庭配置資訊會盡量最小化（如本文件的 Siri 一節所述），以提供房間名稱、配件和指令辨識所需的場景。

HomeKit 配件的 iCloud 遠端存取

HomeKit 配件可直接與 iCloud 連接，讓 iOS 裝置在無法使用藍牙或 Wi-Fi 通訊時控制配件。

「iCloud 遠端存取」經過精密設計，因此無須對 Apple 顯示配件為何或傳送的指令與通知為何，即可控制配件和傳送通知。HomeKit 不會透過「iCloud 遠端存取」來傳送有關住家的資訊。

當使用者使用 iCloud 遠端存取來傳送指令時，配件和 iOS 裝置會彼此認證，而資料會使用專為區域連線所設定的相同程序來進行加密。通訊的內容會經過加密，且不會對 Apple 顯示。透過 iCloud 設定位址的作業是基於在設定處理期間所註冊的 iCloud 識別碼。

支援 iCloud 遠端存取的配件會在配件的設定處理期間加以佈建。佈建處理會在使用者登入 iCloud 時開始。接著，iOS 裝置會要求配件使用「Apple 認證副處理器」（內建於所有 Built for HomeKit 配件）來簽署詢問。配件也會產生 prime256v1 橢圓曲線密鑰，而公用密鑰則會連同簽署的詢問和認證副處理器的 X.509 憑證，一起傳送至 iOS 裝置。這些會用來從 iCloud 佈建伺服器要求配件的憑證。憑證會由配件儲存，但不會包含配件的相關識別資訊，除非已將存取權限授予 HomeKit iCloud 遠端存取。執行佈建的 iOS 裝置也會傳送 bag 至配件，其中包含要連接 iCloud 遠端存取伺服器所需的 URL 和其他資訊。此資訊不特定於任何使用者或配件。

每個配件都會向 iCloud 遠端存取伺服器註冊所允許使用者的列表。這些使用者已從將配件加入住家的人員，獲得控制配件的能力。使用者會取得由 iCloud 伺服器授予的識別碼並可對應到 iCloud 帳號，以從配件傳送通知訊息和回應。同樣地，配件具有 iCloud 發出的識別碼，但這些識別碼難以識別且不會顯示有關配件本身的任何資訊。

當配件連接至 HomeKit iCloud 遠端存取伺服器時，便會提供其憑證與通行證。通行證是從不同的 iCloud 伺服器中取得，且並非對每個配件都是唯一的。當配件要求通行證時，其會在要求中包含製造商、型號和韌體版本。此要求中並不會傳送任何使用者識別或住家識別的資訊。與通行證伺服器的連線不會經過認證，以協助保護隱私。

配件會使用 HTTP/2 來連接至 iCloud 遠端存取伺服器，並使用 TLS 1.2 及 AES-128-GCM 和 SHA-256 加以保護。配件會讓其與 iCloud 遠端存取伺服器的連線保持開啟，以便其接收傳入的訊息和將回應與外寄通知傳送給 iOS 裝置。

HealthKit

HealthKit 架構提供通用資料庫，App 在取得使用者許可後，便可用來儲存和存取健身與健康資料。HealthKit 也可直接用於健康與健身裝置，如相容的藍牙 LE 心率監視器以及許多 iOS 裝置內建的動作副處理器。

健康資料

HealthKit 會使用資料庫來儲存使用者的健康資料，如身高、體重、步行距離、血壓等等。此資料庫會儲存於「資料保護」類別的「完整保護」，這表示只有在使用者輸入其密碼或使用 Touch ID 來解鎖裝置後才可存取該資料庫。

其他資料庫會儲存程式運作資料，如 App 的存取表、連接 HealthKit 的裝置名稱及排程資訊（用來在新資料可用時啟動 App）。此資料庫會儲存在「資料保護」類別的「首次使用者認證前的保護」中。

臨時日誌檔會儲存健康記錄，當裝置鎖定時便會產生這些記錄（例如，當使用者從事運動時）。這些會儲存在「資料保護」類別的「未打開檔案的保護」。當裝置解鎖時，它們會被輸入到主要的健康資料庫，然後在合併作業完成時刪除。

健康資料不會透過 iCloud 共享或在裝置間同步。健康資料庫會被包含在 iCloud 或 iTunes 的加密裝置備份中。健康資料並不會包含在未加密的 iTunes 備份中。

資料完整性

儲存在資料庫中的資料包含追蹤每筆資料記錄出處的元資料。此元資料包含應用程式識別碼，會識別哪個 App 儲存了該記錄。此外，選擇性的元資料項目可包含記錄的數位簽署副本。此用意是提供記錄（由受信任之裝置所產生）的資料完整性。用於數位簽名的格式為 IETF RFC 5652 中所指定的加密訊息語法（Cryptographic Message Syntax，CMS）。

第三方 App 的存取

對 HealthKit API 的存取是使用授權來控制，而 App 必須符合資料使用方式的限制。例如，App 不允許將健康資料用於廣告用途。App 也必須提供隱私權政策給使用者，並詳述其對健康資料的使用方式。

App 對健康資料的存取權是受使用者的「隱私」設定所控制。使用者會被要求在 App 請求健康資料的存取權時（類似於「聯絡資訊」、「照片」和其他 iOS 資料來源），授予存取權。然而，使用健康資料時，App 會被授予讀取和寫入資料的獨立存取權，以及各種類型之健康資料的獨立存取權。使用者可以在「健康」App 的「來源」標籤頁中檢視和撤銷他們授予存取健康資料的權限。

若 App 取得寫入資料的權限，便也可讀取其寫入的資料。若 App 取得讀取資料的權限，便可讀取所有來源所寫入的資料。然而，App 無法決定授予給其他 App 的存取權。此外，App 無法確切地得知它們是否已被授予健康資料的讀取存取權。當 App 沒有讀取存取權時，所有查詢並不會傳回資料—就如同空白資料庫會傳回的相同回應一樣。這可避免 App 藉由得知使用者正在追蹤的資料類型，來推測使用者的健康狀態。

醫療卡

「健康」App 可讓使用者選擇填寫「醫療卡」表單和發生緊急醫療事故時所需的重要資料。此資訊是手動輸入或更新，並不會與健康資料庫中的資料進行同步。

您可點一下鎖定畫面上的「緊急服務」按鈕來檢視「醫療卡」資訊。此資訊會使用「資料保護」類型的「無保護」來儲存於裝置上，以便無須輸入裝置密碼即可取用。「醫療卡」是選擇性的功能，可讓使用者決定如何同時在安全性和隱私考量上取得平衡點。

Apple Watch

Apple Watch 使用為 iOS 打造的安全性功能與技術來協助保護裝置上的資料，同時亦保護與已配對 iPhone 和 Internet 的通訊。這包含如「資料保護」與鑰匙圈存取控制等技術。使用者的密碼也會與裝置 ID 結合以建立加密密鑰。

將 Apple Watch 與 iPhone 配對是使用頻外（OOB）處理加以保護以交換公用密鑰，其後面並接著 BTLE 連結共享密鑰。Apple Watch 會顯示動畫圖形，會使用 iPhone 上的相機加以捕捉。圖形包含已編碼的密鑰，用於 BTLE 4.1 頻外配對。若有需要，「標準 BTLE 密鑰項目」（Standard BTLE Passkey Entry）會用作備用配對方式。

一旦建立 BTLE 工作階段，Apple Watch 和 iPhone 便會使用自 IDS 改寫的處理程序來交換密鑰，如本白皮書的 iMessage 一節所述。一旦已交換密鑰，藍牙工作階段密鑰便會被捨棄，且 Apple Watch 與 iPhone 間的所有通訊都會使用 IDS 加密，而加密後的 BTLE 與 Wi-Fi 連結則會提供次要加密層級。每隔 15 分鐘會使用密鑰輪詢來限制曝光期間，以防流量遭到侵害。

為了支援需要連續播送資料的 App，會使用本白皮書 FaceTime 一節中提到的方式來提供加密，使用由已配對 iPhone 所提供的 IDS 服務。

Apple Watch 會為檔案與鑰匙圈項目導入以硬體加密的儲存空間與類別式保護，如本白皮書的「資料保護」一節所述。鑰匙圈項目的存取控制 Keybag 也會一併使用。手錶與 iPhone 間通訊所使用的密鑰也會利用類別式保護來確保其安全性。

當 Apple Watch 不在的藍牙範圍內時，可改為使用 Wi-Fi。Apple Watch 將不會加入 Wi-Fi 網路，除非在已配對的 iPhone 上已有此動作的認證，其會自動提供已知網路的列表給手錶。

Apple Watch 可藉由按住側邊按鈕來手動加以鎖定。此外，還會使用動作啟發（motion heuristics）以嘗試在手錶從手腕取下不久後，即自動鎖定裝置。鎖定後，便無法使用 Apple Pay。若手腕偵測提供的自動鎖定在設定中已關閉，則會停用 Apple Pay。需使用 iPhone 上的 Apple Watch App 關閉手腕偵測。此設定也可使用行動裝置管理來強制執行。

若正穿戴著手錶，也可使用已配對的 iPhone 來解鎖手錶。這是藉由使用在配對期間建立的密鑰進行認證連線來達成。iPhone 會傳送密鑰，而手錶會使用該密鑰來解鎖其資料保護密鑰。手錶密碼並不會讓 iPhone 取得，也不會被傳輸。此功能可在 iPhone 上使用 Apple Watch App 來加以關閉。

Apple Watch 一次只能與一台 iPhone 配對。與新 iPhone 配對時，會自動從 Apple Watch 中清除所有內容與資料。

在已配對 iPhone 上啟用「尋找我的 iPhone」也會在 Apple Watch 上啟用「啟用鎖定」。「啟用鎖定」可讓 Apple Watch 在遺失或遭竊時，其他人無法輕易使用或銷售。「啟用鎖定」需要使用者的 Apple ID 和密碼才能取消配對、清除或重新啟用 Apple Watch。

網路安全性

除了 Apple 用於保護 iOS 裝置上所儲存資料的內建安全保護，也有許多網路安全措施可供企業組織採用並確保資訊從 iOS 裝置來回傳輸時安全無虞。

行動使用者必須能在全球各處存取公司網路，因此很重要的一點是確保他們獲得授權並且其資料在傳輸期間受到保護。iOS 使用標準網路通訊協定並使開發者能夠存取這些通訊協定，以進行受認證、已授權且已加密的通訊。為了達成這些安全性的目標，iOS 整合了經過實證的技術和最新標準來進行 Wi-Fi 和行動數據網路的連線。

在其他平台上，需要用防火牆軟體來保護開放式通訊埠，以防止入侵。因為 iOS 透過限制監聽埠以及移除不必要的網路工具程式（如 telnet、shell 或網頁伺服器），使受攻擊的範圍減小，因此在 iOS 裝置上不需要額外的防火牆軟體。

TLS

iOS 支援傳輸層安全性（TLS v1.0、TLS v1.1、TLS v1.2）和 DTLS。Safari、「行事曆」、「郵件」和其他 Internet App 會自動使用這些機制在裝置和網路服務之間建立一條加密的通訊通道。高階 API（如 CFNetwork）讓開發者可以輕鬆在其 App 中採用 TLS，而低階 API（SecureTransport）則提供精細的控制。依照預設，CFNetwork 不允許 SSLv3，而使用 WebKit 的 App（如 Safari）也被禁止進行 SSLv3 連線。

App 傳輸安全性

「App 傳輸安全性」提供預設連線的需求，以便 App 在使用 NSURLConnection、CFURL 或 NSURLSession API 時，遵循安全連線的最佳做法。

伺服器必須支援 TLS 1.2 的最低限度（Forward Secrecy），且憑證必須有效並使用 SHA-256 加以簽署（最好使用 2048 位元 RSA 密鑰或 256 位元橢圓曲線密鑰的最低限度）。

不符合這些要求的網路連線作業將會失敗，除非 App 修訂「App 傳輸安全性」。無效憑證隨時會造成嚴重的作業失敗和連線中斷。「App 傳輸安全性」會自動套用到針對 iOS 9 編譯的 App。

VPN

與虛擬專用網路類似的安全網路服務通常只需要簡單的設定和配置，便可配合 iOS 裝置使用。與 iOS 裝置搭配使用的 VPN 伺服器支援以下通訊協定和認證方式：

- IKEv2/IPSec，含以共用密鑰、RSA 憑證、ECDSA 憑證、EAP-MSCHAPv2 或 EAP-TLS 進行的認證。
- Pulse Secure、Cisco、Aruba Networks、SonicWALL、Check Point、Palo Alto Networks、Open VPN、AirWatch、MobileIron、NetMotion Wireless 以及 F5 Networks SSL-VPN，它們在 App Store 中提供適用的用戶端 App。
- Cisco IPSec，此通訊協定透過密碼、RSA SecurID 或 CRYPTOCard 進行使用者認證，並藉由共享密鑰和憑證進行機器認證。
- L2TP/IPSec，此通訊協定透過 MS-CHAPV2 密碼、RSA SecurID 或 CRYPTOCard 進行使用者認證，並藉由共享密鑰進行機器認證。
- PPTP，此通訊協定透過 MS-CHAPV2 密碼和 RSA SecurID 或 CRYPTOCard 進行使用者認證（支援此協定，但不建議）。

對於使用以憑證為基礎的認證網路，iOS 支援「隨選即用 VPN」。IT 規則會藉由使用設定描述檔來指定哪些網域需要 VPN 連線。

iOS 也支援為應用程式單獨設定 VPN 支援，在更精確的基礎上完成建立 VPN 連線。行動裝置管理（MDM）可為每個受管理的 App 和/或 Safari 中特定的網域指定連線。這有助於確保進出公司網路的資料始終是安全的，而使用者的個人資料並不會進出公司網路。

iOS 支援「總是開啟 VPN」，透過 MDM 管理的裝置、使用 Apple Configurator 或 Device Enrollment Program（裝置登記方案）監督的裝置可進行該設定。這可讓使用者在連接到行動數據與 Wi-Fi 網路時，不需要手動開啟 VPN 以啟用保護。「總是開啟 VPN」透過將所有 IP 流量回傳至組織，讓組織對裝置流量擁有完整的控制權。預設通道的通訊協定 IKEv2 使用資料加密對流量傳輸進行安全保護。現在，組織可以監控並過濾傳入其裝置或自其裝置傳出的流量、保護組織網路內的資料安全並限制裝置連接 Internet。

Wi-Fi

iOS 支援業界標準的 Wi-Fi 通訊協定，包括「WPA2 企業級」，可針對無線企業網路提供連線認證服務。「WPA2 企業級」使用 128 位元 AES 加密，可為使用者提供最高等級的安全保障：在透過 Wi-Fi 網路連線傳送和接收通訊時，確保使用者的資料始終受到保護。有了 802.1X 的支援，iOS 裝置可被整合到各種 RADIUS 認證環境中。iPhone 和 iPad 上支援的 802.1X 無線認證方式包括 EAP-TLS、EAP-TTLS、EAP-FAST、EAP-SIM、PEAPv0、PEAPv1 和 LEAP。

當 iOS 沒有與 Wi-Fi 網路產生關聯且裝置處理器處於睡眠狀態時，iOS 8 會在執行 PNO（Preferred Network Offload）掃描時，使用隨機「媒體存取控制」（MAC）位址。裝置的處理器會在螢幕關閉後不久進入睡眠。PNO 掃描目的為判別使用者是否已連接偏好的 Wi-Fi 網路以執行作業，例如以無線方式與 iTunes 同步。

當 iOS 沒有與 Wi-Fi 網路產生關聯或裝置處理器處於睡眠狀態時，iOS 8 會在執行 ePNO（enhanced Preferred Network Offload）掃描時，使用隨機 MAC 位址。當裝置上針對會利用地理柵欄的 App 使用定位服務時（例如會判定裝置是否接近特定位置的位置相關提醒事項），便會執行 ePNO 掃描。

現在因為裝置未連接 Wi-Fi 網路時其 MAC 位址會更改，即使裝置已連接行動網路，Wi-Fi 流量的被動觀察程式亦無法使用該位址持續追蹤裝置。

我們與 Wi-Fi 製造商合作，讓他們知曉背景掃描會使用隨機 MAC 位址，且 Apple 及製造商皆無法預測這些隨機 MAC 位址。

iPhone 4s 上不支援 Wi-Fi MAC 位址隨機載入。

藍牙

iOS 的藍牙支援旨在提供實用的功能，而不會增加對私密資料不必要的訪問。iOS 裝置支援 Encryption Mode 3、Security Mode 4 和 Service Level 1 連線。iOS 支援以下藍牙描述檔：

- 免持描述檔（HFP 1.5）
- 電話簿取用描述檔（PBAP）
- 進階音訊分配描述檔（A2DP）
- 音訊/視訊遠端控制描述檔（AVRCP）
- 個人區域網路描述檔（PAN）
- 人機介面裝置描述檔（HID）

對這些描述檔的支援因設備而異。如需更多資訊，請參訪 support.apple.com/kb/ht3647?viewlocale=zh_TW。

單一登入

iOS 支援透過單一登入（SSO）對企業網路進行認證。SSO 搭配以 Kerberos 為基礎的網路使用，針對使用者獲授權取用的服務對使用者進行認證。SSO 可用於各種網路活動，從安全的 Safari 區段到第三方的 App。

iOS SSO 利用 SPNEGO 代號和 HTTP Negotiate 通訊協定，與以 Kerberos 為基礎的認證閘道和支援 Kerberos ticket 的 Windows Integrated Authentication 系統配合使用。同時還支援以憑證為基礎的認證作業。SSO 的支援以開放原始碼 Heimdal 專案為基礎。

支援下列加密類型：

- AES128-CTS-HMAC-SHA1-96
- AES256-CTS-HMAC-SHA1-96
- DES3-CBC-SHA1
- ARCFOUR-HMAC-MD5

Safari 支援 SSO，且使用標準 iOS 網路連線 API 的第三方 App 也可進行設定來使用。為了設定 SSO，iOS 支援設定描述檔的承載資料，允許 MDM 伺服器向下推播必要的設定。其中包括：設定使用者主要名稱（即 Active Directory 使用者帳號）和 Kerberos 領域設定，以及設定應允許哪些 App 和/或 Safari Web URL 使用 SSO。

AirDrop 安全性

支援 AirDrop 的 iOS 裝置使用低功耗藍牙（BLE）和 Apple 建立的點對點 Wi-Fi 技術來向附近的裝置傳送檔案和資訊，包括具有 AirDrop 功能並執行 OS X Yosemite 或更新版本的 Mac 電腦。Wi-Fi 訊號用來在裝置之間進行直接通訊，無需使用任何 Internet 連線或 Wi-Fi 存取點。

當使用者啟用 AirDrop 後，裝置上就會儲存一個 2048 位元的 RSA 識別身份。此外，裝置還會依據與使用者的 Apple ID 相關聯的電子郵件位址和電話號碼，建立一個 AirDrop 識別身份的雜湊值。

當使用者選擇使用 AirDrop 共享項目時，設備會透過低功耗藍牙發出 AirDrop 訊號。附近處於喚醒狀態且啟用了 AirDrop 的其他裝置偵測到此訊號後，便會使用其持有人的識別身份雜湊值的簡短版本進行回應。

在預設情況下，AirDrop 的共享設定為「僅限聯絡人」。使用者也可以選擇是否希望使用 AirDrop 與所有人進行共享，或者完全關閉此功能。在「僅限聯絡人」模式下，接收到的識別身份雜湊值會與發起者其「聯絡資訊」App 中的人員的雜湊值進行比對。若找到相符項目，發送裝置會建立一個點對點 Wi-Fi 網路並使用 Bonjour 告知已建立 AirDrop 連線。接收裝置會使用此連線將其完整的識別身份雜湊值傳送給發起者。如果完整雜湊值仍與「聯絡資訊」相符，接收者的名字和照片（如果「聯絡資訊」中有的話）便會顯示在 AirDrop 共享表單中。

使用 AirDrop 時，由傳送方使用者選擇要與其共享內容的對象。發送裝置會與接收裝置建立一個加密的（TLS）連線，此連線會交換它們的 iCloud 識別身份憑證。憑證中的識別身份會針對每位使用者的「聯絡資訊」App 進行驗證。然後會要求接收方使用者接受來自經驗證之人員或裝置所傳送的內容。如果選擇了多位接收者，則將針對每個目標重複此過程。

在「所有人」模式中，會使用相同的過程，但若未能在「聯絡資訊」中找到相符項目，接收裝置會顯示在 AirDrop 傳送表單中，並帶有一個小圖像和裝置名稱，該名稱可在「設定」>「一般」>「關於本機」>「名稱」中找到。

組織可限制裝置對 AirDrop 的使用以及透過行動裝置管理解決方案鎖管理的 App。

Apple Pay

使用 Apple Pay，使用者可以使用受支援的 iOS 裝置和 Apple Watch 來以簡單、安全又保密的方式來進行付款。對於使用者來說很容易，且在硬體和軟體方面皆具備整合性的安全措施。

Apple Pay 的目標也在於保護使用者的個人資訊。Apple Pay 不會收集任何可追蹤使用者的交易資訊。付款交易僅於使用者、商家和發卡機構之間流通。

Apple Pay 元件

Secure Element：Secure Element 是符合工業標準、受認證的晶片，於 Java Card 平台上運作，符合金融業的電子付款要求。

NFC 控制器：NFC 控制器處理「近場通訊」通訊協定並傳送應用程式處理器與 Secure Element 之間的通訊，以及 Secure Element 與銷售點終端機之間的通訊。

Wallet：Wallet 用來加入和管理信用卡、金融卡、酬賓卡和商店卡，並使用 Apple Pay 付款。使用者可以檢視其卡片和發卡機構的其他資訊、發卡機構的隱私政策、近期交易，以及 Wallet 中的其他內容。使用者也可以在「設定輔助程式」和「設定」中加入卡片。

Secure Enclave：在 iPhone 和 iPad 上，Secure Enclave 會管理認證程序並使付款交易生效。Secure Enclave 會儲存 Touch ID 的指紋資料。

在 Apple Watch 上，裝置必須解鎖，而使用者必須按兩下側邊按鈕。按兩下的動作會被偵測並直接傳遞到 Secure Element，不會經過應用程式處理器。

Apple Pay 伺服器：Apple Pay 伺服器會管理信用卡和金融卡在 Wallet 中的狀態，以及儲存在 Secure Element 中的「裝置帳號號碼」。它們會與裝置和付款網路伺服器進行通訊。Apple Pay 伺服器也負責為 App 內的付款重新加密付款憑證。

Apple Pay 使用 Secure Element 的方式

Secure Element 主控著一種特殊設計的 Applet 來管理 Apple Pay。其中也包含經由付款網路認證的付款 Applet。加密這些付款 Applet 的付款網路或發卡機構會使用密鑰傳送信用卡或金融卡資料，只有付款網路和 Applet 的安全網域擁有密鑰的相關資訊。此資料會儲存在這些付款 Applet 中，並受到 Secure Element 的安全性功能保護。在交易期間，終端機會經由專用硬體匯流排，透過近場通訊（NFC）控制器直接與 Secure Element 進行通訊。

Apple Pay 使用 NFC 控制器的方式

作為 Secure Element 的閘道，NFC 控制器會確保所有非接觸式付款交易均使用接近該裝置的銷售點終端機進行。只有來自內場終端機的付款要求才會被 NFC 控制器標示為非接觸式付款交易。

卡片持有人一旦使用 Touch ID 或密碼或在解鎖的 Apple Watch 上按兩下側邊按鈕來授權付款，Secure Element 內由付款 Applet 準備的非接觸式付款回應便會特地由控制器傳送至 NFC 內場。因此，非接觸式付款交易的付款授權詳細資料便會包含至本機 NFC 內場，且絕對不會暴露於應用程式處理器。相反地，App 內付款的付款授權詳細資料會傳送至應用程式處理器，但在此之前會先由 Secure Element 將資訊加密至 Apple Pay 伺服器。

信用卡和金融卡佈建

當使用者加入信用卡或金融卡（包含商店卡）到 Apple Pay 時，Apple 會以安全的方式將卡片資訊以及使用者帳號和裝置的其他資訊傳送到發卡機構。透過此資訊，發卡機構會判定是否要核准將該卡片加入 Apple Pay。

Apple Pay 會使用三種伺服器端調用來傳送和接收與發卡機構或網路之間的通訊，作為卡片佈建程序的一部分：**必要欄位**、**檢查卡片**與**連結和佈建**。發卡機構或網路會使用這些調用來驗證、核准卡片以及將卡片加入 Apple Pay。這些主從式架構工作階段皆使用 SSL 加密。

完整卡號不會儲存在裝置或 Apple 伺服器上。反之，系統會建立一個唯一的「裝置帳號號碼」，並對其加密，然後儲存在 Secure Element 中。這個唯一的「裝置帳號號碼」會以此方式進行加密，讓 Apple 無法取用。「裝置帳號號碼」是唯一的號碼，且與一般信用卡或金融卡號不同，發卡機構可以防止將此號碼用於磁條卡、電話或網站。Secure Element 中的「裝置帳號號碼」與 iOS 和 WatchOS 相隔離，且絕對不會儲存在 Apple Pay 伺服器上，亦不會備份至 iCloud。

搭配 Apple Watch 使用的卡片是使用 iPhone 上的 Apple Watch App 來提供給 Apple Pay。提供卡片給 Apple Watch 會要求手錶必須位於藍牙通訊範圍內。卡片會特別註冊為搭配 Apple Watch 使用且具有其「裝置帳號」，此資訊會儲存在 Apple Watch 上的 Secure Element 內。

有兩種方式可以將信用卡或金融卡佈建到 Apple Pay 中：

- 手動將信用卡或金融卡加入 Apple Pay
- 將來自 iTunes Store 帳號歸檔的信用卡或金融卡加入 Apple Pay

手動將信用卡或金融卡加入 Apple Pay

若要手動加入卡片（包含商店卡），則需要姓名、信用卡卡號、到期日和 CVV 來執行佈建程序。從「設定」內、Wallet App 或 Apple Watch App，使用者可以藉由打字方式或使用 iSight 相機來輸入該資訊。當相機擷取到卡片資訊時，Apple 會嘗試填入姓名、卡號和到期日。照片並不會儲存在裝置上或照片圖庫中。所有欄位填妥後，「卡片檢查」程序會驗證 CVV 以外的欄位。所有資訊會經過加密並傳送到 Apple Pay 伺服器。

若「卡片檢查」程序傳回使用條款 ID，則 Apple 會下載發卡機構的使用條款與條件，並顯示給使用者閱覽。如果使用者接受使用條款，Apple 便會將所接受條款的 ID 連同 CVV 傳送至「連結」和「佈建」程序。此外，作為「連結」和「佈建」程序的一部分，Apple 會與發卡機構或網路分享裝置的資訊，像是您 iTunes 和 App Store 帳戶活動的相關資訊（例如，您是否在 iTunes 內有長期的交易記錄）、您裝置的資訊（例如，電話號碼、裝置名稱及裝置機型以及任何設定 Apple Pay 必要的輔助 iOS 裝置），以及加入卡片時的概略位置（若您啟用「定位服務」）。透過此資訊，發卡機構會判定是否要核准將該卡片加入 Apple Pay。

「連結」和「佈建」程序會產生兩個結果：

- 裝置會開始下載代表信用卡或金融卡的 Wallet 票券檔案。
- 裝置會開始將卡片綁定至 Secure Element。

票券檔案包含 URL，可供下載卡片封面、與卡片相關的元資料（例如聯絡資訊）、相關發卡機構 App 和支援的功能。它也包含票券狀態，其中的資訊包含 Secure Element 的個人化是否已完成、卡片目前是否已遭發卡銀行停用，或是在卡片可以搭配 Apple Pay 進行付款前，還需要進行哪些額外驗證。

將來自 iTunes Store 帳號的信用卡或金融卡加入 Apple Pay

對於使用 iTunes 歸檔的信用卡或金融卡，使用者可能需要重新輸入其 Apple ID 密碼。系統會從 iTunes 擷取卡號，並隨之啟動「卡片檢查」程序。如果卡片符合使用 Apple Pay 的資格，裝置會下載並顯示使用條款，並連帶條款 ID 和卡片安全碼傳送至「連結」和「佈建」程序。iTunes 帳號存檔的卡片資料可能需要額外驗證。

從發卡機構的 App 加入信用卡或金融卡

當 App 註冊使用 Apple Pay 時，便會為 App 和商家伺服器建立密鑰。這些密鑰用來加密傳送給商家的卡片資訊，以避免資訊遭 iOS 裝置讀取。此佈建流程類似於手動加入卡片所使用的流程（如上所述），但替代 CVV 所用的一次性密碼除外。

其他驗證

發卡機構有權決定信用卡或金融卡是否需要其他驗證。視發卡機構提供的選項而定，使用者可能有多種額外驗證選項，例如文字訊息、電子郵件、客服電話，或是經核准第三方 App 所提供的方法來完成驗證。若使用文字訊息或電子郵件，使用者需從發卡機構存檔的聯絡資訊中選擇。接著會傳送一組代碼，供使用者輸入 Wallet、「設定」或 Apple Watch App。若使用客服或 App 驗證，發卡機構會實行其自有的通訊流程。

付款授權

Secure Element 只會在從 Secure Enclave 接收到授權，確認使用者已透過 Touch ID 或裝置密碼授權後，才會允許付款。若裝置適用 Touch ID，其便為預設方法；但使用者隨時都可以使用密碼來取代 Touch ID。在三次嘗試比對指紋失敗後，會自動建議改用密碼；而嘗試失敗五次後，則必須輸入密碼。當使用者未設定 Touch ID 或是沒有針對 Apple Pay 啟用 Touch ID 時，也需要輸入密碼。

Secure Enclave 和 Secure Element 之間的通訊會在序列介面上進行，Secure Element 會連接到 NFC 控制器，接著再連接到應用程式處理器。即使未直接連接，Secure Enclave 和 Secure Element 可以使用共享的密鑰組來安全地通訊，此密鑰組是在製造過程中佈建的。通訊的加密和認證是基於 AES，兩端的通訊方皆會使用加密隨機數來防止重播攻擊。密鑰組是從 Secure Enclave 內的 UID 密鑰和 Secure Element 的唯一識別碼產生。密鑰組接著會安全地從 Secure Enclave 傳送至工廠內的硬體安全性模組（HSM），其中包含所需的密鑰材料，再將密鑰組植入 Secure Element。

當使用者授權交易時，Secure Enclave 會將認證類型的簽署資料和交易類型的詳細資料（非接觸式付款或 App 內）傳送至 Secure Element，繫結至「授權隨機」（AR）值。當使用者首次佈建信用卡並於 Apple Pay 啟用時保存，便會在 Secure Enclave 內產生 AR，AR 受 Secure Enclave 的加密和反復原機制的保護。它會透過密鑰組安全地傳送到 Secure Element。在接收到新的 AR 值時，Secure Element 會將任何先前加入過的卡片標示為已刪除。

只有在 Secure Element 使用與加入卡片時相同的密鑰組和 AR 值出示授權時，才能使用加入到 Secure Element 的信用卡和金融卡。這使 iOS 在下列情況發出指令，讓 Secure Enclave 將 AR 拷貝標示為無效，將卡片轉譯為無法使用的狀態：

當密碼停用時。

- 使用者登出 iCloud 時。
- 使用者選擇清除所有內容和設定時。
- 裝置從恢復模式回復時。

使用 Apple Watch 時，卡片會在下列情況標示為無效：

- 手錶的密碼已停用。
- 手錶未與 iPhone 配對。
- 手腕偵測已關閉。

使用密鑰組及其目前 AR 值的拷貝，Secure Element 會先驗證從 Secure Enclave 接收到的授權，才會啟用付款 Applet 來進行非接觸式付款。在 App 內進行交易時，會擷取來自付款 Applet 的加密付款資料，此時亦會套用此程序。

交易特定動態安全碼

來自付款 Applet 的所有付款交易均包含交易特定動態安全碼，以及「裝置帳號號碼」。此一次性安全碼的計算是使用每次新交易遞增的計數器，以及個人化期間付款 Applet 所佈建且付款網路和/或發卡機構已知的密鑰來計算。此一次性安全碼的計算是使用每次新交易遞增的計數器，以及個人化期間付款 Applet 所佈建且付款網路和/或發卡機構已知的密鑰來計算。視付款方案而定，其他資料也會用於計算這些密碼，包含下列資料：

- 付款 Applet 產生的隨機數
- 另一個由終端機產生的隨機數，用於 NFC 交易，

  或是

- 另一個由伺服器產生的隨機數，用於 App 內交易

這些安全碼會提供給付款網路和發卡機構，供他們驗證每筆交易。這些安全碼的長度會視所完成交易的類型而有所不同。

使用 Apple Pay 進行非接觸式付款

如果 iPhone 已開啟並偵測到 NFC 磁場，便會向使用者顯示「設定」中管理的相關信用卡或金融卡，或預設的卡片。使用者也可前往 Wallet App 並選擇信用卡或金融卡，或當裝置鎖定時，按兩下主畫面按鈕。

接下來，在發送付款資訊前，使用者必須使用 Touch ID 或其密碼來授權。當 Apple Watch 解鎖時，按兩下側邊按鈕來啟用付款的預設卡片。所有付款資訊皆需經過使用者授權始得發送。

使用者授權後，在處理交易時便會使用「裝置帳號號碼」和交易特定動態安全碼。無論是 Apple 或使用者的裝置皆不會將完整的實際信用卡或金融卡號碼傳送至商家。Apple 可能會接收到匿名的交易資訊，如交易的約略時間和地點，這可協助改進 Apple Pay 及其他的 Apple 產品和服務。

在 App 中使用 Apple Pay 付款

Apple Pay 也可以用來在 iOS App 內進行付款。當使用者以 Apple Pay 在 App 內付款時，Apple 會收到加密的交易資訊，並以商家特定密鑰再次加密，才會傳送給商家。Apple Pay 會保留匿名的交易資訊，例如約略購買金額。此資訊無法用來追蹤使用者，且從不包含使用者購買的商品。

當 App 起始 Apple Pay 付款交易時，Apple Pay 伺服器會比商家先收到來自裝置的加密資訊。Apple Pay 伺服器接著會以商家特定密鑰再次加密，才會將交易傳遞給商家。

當 App 要求付款時，會呼叫 API 以判別裝置是否支援 Apple Pay，以及使用者所使用的信用卡或金融卡是否可在商家認可的付款網路上進行付款。App 會要求所有資料需經過處理以完成交易，例如帳單和送貨地址，以及聯絡資訊。App 接著會要求 iOS 出示 Apple Pay 表單，其會要求 App 的資訊以及其他必要資訊，如要使用的卡片。

此時，App 會出示城市、州和郵遞區號資訊來計算最終運費。全部的資訊並不會提供給 App，直到使用者以 Touch ID 或裝置密碼授權付款。付款一經授權，Apple Pay 表單內出示的資訊便會傳送給商家。

當使用者授權付款時，系統會呼叫 Apple Pay 伺服器以取得加密隨機數，這與進行店內交易時 NFC 終端機傳回的數值類似。隨機數和其他交易資料會傳遞到 Secure Element 以產生付款憑證，此付款憑證會以 Apple 密鑰進行加密。當加密的付款憑證從 Secure Element 發出後，會傳遞到 Apple Pay 伺服器，伺服器會解密憑證、比對憑證中的隨機數與 Secure Element 發送的隨機數，然後使用與「商家 ID」關聯的商家密鑰重新加密付款憑證。接著付款憑證會傳回裝置，透過 API 傳送回 App。App 接下來會將付款憑證傳遞到商家系統進行處理。商家便可以使用其專用密鑰解鎖付款憑證以進行處理。這會結合來自 Apple 伺服器的簽名，允許商家驗證交易是針對此特定商家所進行的。

API 會要求一個授權，此授權用來指定支援的商家 ID。App 也可能包含要傳送至 Secure Element 加以簽署的其他資料，例如訂單號碼或消費者身份，以確保交易不會被移轉到其他消費者。此項作業由 App 開發者執行。App 開發者能夠指定 PKPaymentRequest 上的 applicationData。此資料的雜湊值會包含在加密的付款資料中。商家接著會負責驗證 applicationData 雜湊值是否與付款資料內包含的內容相符。

酬賓卡

自 iOS 9 起，Apple Pay 支援「加值服務」（VAS）通訊協定，以將商家酬賓卡傳輸至相容的的 NFC 終端機。VAS 通訊協定可在商家終端機上導入，並使用 NFC 來與支援的 Apple 裝置進行通訊。VAS 通訊協定可在短距離間運作，並用來提供輔助式服務作為 Apple Pay 交易的一部分，如傳輸酬賓卡資訊。

NFC 終端機會藉由傳送卡片的要求來啟動卡片資訊的接收作業。若使用者具有含有商店識別碼的卡片，便會被要求授權卡片的使用。若商家支援加密，卡片資訊、時間戳記及單一使用隨機 ECDH P-256 密鑰便會與商家的公用密鑰一起使用，以衍生卡片資料的加密密鑰供傳送至終端機。若商家並未提供加密，使用者便會被要求向終端機再次出示裝置，才會傳送酬賓卡資訊。

停用、移除和清除卡片

使用者可以使用「尋找我的 iPhone」來將裝置設為「遺失模式」，藉此停用 iPhone 和 iPad 上的 Apple Pay。使用者也可以使用「尋找我的 iPhone」、「iCloud 設定」或直接在裝置上使用 Wallet 來從 Apple Pay 清除卡片。在 Apple Watch 上，卡片可使用 iCloud 設定、iPhone 上的 Apple Watch，或直接在手錶上進行移除。在裝置上使用卡片來進行付款的功能將會由發卡機構或個別付款網路從 Apple Pay 停用或移除，即使裝置為離線狀態且未連線至行動網路或 Wi-Fi 網路。使用者也可撥打電話給其發卡機構來從 Apple Pay 停用或移除卡片。

此外，當使用者透過「清除所有內容和設定」來清除整部裝置、使用「尋找我的 iPhone」，或從恢復模式將裝置回復時，iOS 將會指示 Secure Element 將所有卡片標記為已刪除。這會立即將卡片更改為無法使用的狀態，直到可聯絡到 Apple Pay 伺服器來從 Secure Element 完全清除卡片為止。除此之外，Secure Enclave 會將 AR 標示為無效，使先前登記的卡片無法進行進一步的付款授權。當裝置為線上狀態時，會嘗試聯絡 Apple Pay 伺服器，以確保 Secure Element 中的所有卡片皆已清除。

Internet 服務

製作安全的 Apple ID 密碼

Apple ID 會用來連接許多服務，包含 iCloud、FaceTime 和 iMessage。為了協助使用者製作安全密碼，所有新帳號都必須包含下列密碼屬性：

- 至少有八個字元
- 至少有一個字母
- 至少有一個大寫字母
- 至少有一個數字
- 連續的相同字元不得超過三個
- 不能與帳號名稱相同

Apple 已內建一套強大的服務來協助使用者更充分地使用裝置並提高生產力，其中包含 iMessage、FaceTime、Siri、「Spotlight 建議」、iCloud、「iCloud 備份」和「iCloud 鑰匙圈」。

這些 Internet 服務都具備了 iOS 在整個平台上推動的安全性目標。這些目標包含資料的安全處理，無論是裝置上儲存的靜態資料或是透過無線網路傳輸的資料；保護使用者的個人資訊；以及對資料和服務的惡意或未經授權的存取威脅加以防護。每項服務在使用其本身的強大安全性架構時，絲毫不影響 iOS 整體的易用性。

Apple ID

Apple ID 是由使用者名稱和密碼組成，用來登入 Apple 服務，如 iCloud、iMessage、FaceTime、iTunes Store、iBooks Store、App Store 等等。對使用者而言，安全地保護其 Apple ID 以防止帳號遭未經授權的存取，是十分重要的。為了達成此目標，Apple 要求使用長度至少為八個字元的安全密碼，同時包含字母和數字，連續的相同字元不得超過三個，且不能為常用的密碼。使用者更可以超越此規則，藉由加入更多的字元和標點符號，來讓密碼變得更為安全。在對帳號進行重要更動時，例如密碼或帳單資訊經變更，或者 Apple ID 被用來在新裝置上登入，Apple 也會向使用者發送電子郵件和推播通知。若有任何事件看似異常，Apple 會提示使用者立即更改其 Apple ID 的密碼。

Apple 還為 Apple ID 提供了兩步驟式的驗證，為使用者的帳號提供了第二重的安全性保護。啟用兩步驟式驗證後，使用者必須藉由發送至其中一部受信任裝置的臨時代碼來驗證其身份，才有權更改其 Apple ID 帳號資訊、登入 iCloud、iMessages、FaceTime 和 Game Center，以及在新裝置的 iTunes Store、iBooks Store 或 App Store 中進行購物。即使其他人得知密碼，也可以避免其取用使用者的帳號。若使用者忘記密碼或無法取用其受信任的裝置，也可以使用存放在安全位置、由 14 個字元組成的「恢復密鑰」。

如需 Apple ID 兩步驟式驗證的更多資訊，請參訪 support.apple.com/kb/ht5570?viewlocale=zh_TW。

iMessage

Apple 的 iMessage 是一項適用於 iOS 裝置和 Mac 電腦的傳訊服務。iMessage 支援文字和附件，如照片、聯絡資訊和位置資訊。資訊會顯示在使用者所有註冊的裝置上，這樣使用者就可以在其他裝置上繼續對話。iMessage 充分使用了 Apple 推播通知服務（APNs）。Apple 不會記錄資訊或附件，且其受端對端的加密服務保護，因此只有傳送者和接收者可以取用。Apple 無法解密這些資料。

當使用者在裝置上打開 iMessage 後，裝置會產生以下兩對密鑰供服務使用：用於加密的 1280 位元 RSA 密鑰和 NIST P-256 曲線上用於簽署的 256 位元 ECDSA 密鑰。每一組密鑰的專用密鑰會儲存在裝置的鑰匙圈中，公用密鑰則與裝置的 APNs 位址一起傳送至 Apple 的目錄服務（IDS），在目錄服務中，公用密鑰會與使用者的電話號碼或電子郵件位址相關聯。

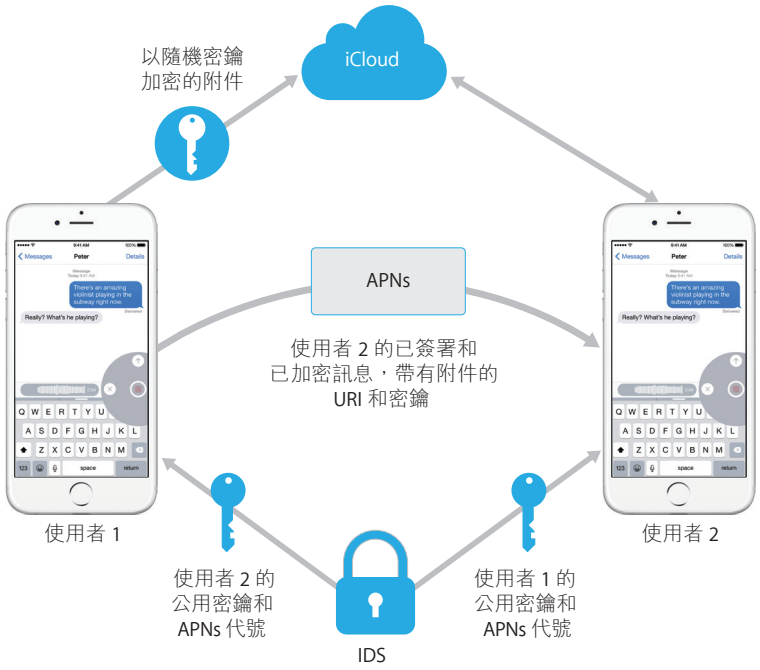
當使用者啟用其他裝置來使用 iMessage 時，他們的加密和簽署公用密鑰、APNs 位址及所關聯的電話號碼都會加入到目錄服務中。使用者還可以加入更多電子郵件位址，這些電子郵件位址會藉由傳送確認連結來進行驗證。電話號碼則透過電信業者網路和 SIM 卡進行驗證。此外，當有新裝置、電話號碼或電子郵件位址被加入時，使用者所有已註冊的裝置都會顯示一則提示訊息。

iMessage 傳送和接收訊息的方式

使用者藉由輸入位址或姓名來開始 iMessage 對話。如果他們輸入電話號碼或電子郵件位址，裝置就會聯絡 IDS 以擷取與該位址相關聯的所有裝置的公用密鑰和 APNs 位址。如果使用者輸入的是名字，裝置會先利用使用者的「聯絡資訊」App 來收集與該名字相關聯的電話號碼和電子郵件位址，然後再從 IDS 中取得公用密鑰和 APNs 位址。

使用者的外送訊息會針對接收者的每個裝置進行個別加密。接收裝置的公用 RSA 加密密鑰會從 IDS 擷取。針對每部接收裝置，傳送裝置會產生隨機 128 位元的密鑰，並以 CTR 模式使用 AES 來加密訊息。每則訊息的 AES 密鑰會使用 RSA-OAEP 對接收裝置的公用密鑰進行加密。加密訊息文字與加密訊息密鑰的組合接著會以 SHA-1 進行雜湊運算，而雜湊值會使用傳送裝置的專用簽署密鑰以 ECDSA 進行簽署。產生的訊息（每部接收裝置一則）是由加密訊息文字、加密訊息密鑰及傳送者的數位簽名所組成。這些訊息隨即會被分送至 APNs 進行遞送。時間戳記和 APNs 路由資訊等元資料則不會加密。與 APNs 的通訊會使用秘密轉送的 TLS 通道進行加密。

APNs 最多只可以中繼大小為 4 KB 或 16 KB 的訊息，視 iOS 版本而定。若訊息文字過長或隨附附件（如照片），附件會使用 AES 在 CTR 模式下以隨機產生的 256 位元密鑰進行加密並上傳至 iCloud。附件的 AES 密鑰、其 URI（統一資源識別碼）和其加密表單的 SHA-1 雜湊值隨後會以 iMessage 的內容被傳送給收件者，這些內容會透過正規的 iMessage 加密保有資料的機密性和完整性，如下所示。



對於群組對話，每位接收者與其裝置之間都會重複此過程。

在接收方，每部裝置接收到的是來自 APNs 的訊息拷貝，且如有需要，裝置會從 iCloud 來擷取附件。若傳送者的電話號碼或電子郵件位址與接收者的聯絡資訊相符，則會顯示一個名字。

與所有推播通知一樣，訊息在遞送後便會從 APNs 中刪除。然而，與其他 APNs 通知不同的是，若裝置離線，iMessage 訊息會排入佇列等待發送。目前訊息最多可儲存 30 天。

FaceTime

FaceTime 是 Apple 的視訊和語音通話服務。與 iMessage 類似，FaceTime 通話使用 Apple「推播通知」服務來與使用者已註冊的裝置建立初始連線。FaceTime 通話的語音/視訊內容受到端對端的加密保護，因此只有傳送者和接收者可以取用。Apple 無法解密這些資料。

FaceTime 使用 Internet Connectivity Establishment（ICE）在裝置間建立點對點的連線。藉由使用「區段初始通訊協定」（SIP）訊息，裝置會驗證其識別憑證並為每個區段建立共用密鑰。每部裝置提供的加密隨機數會合併到每個媒體通道的鹽密鑰（salt key）中，藉由使用 AES-256 加密的「安全即時通訊協定」（SRTP）進行串流。

iCloud

iCloud 會儲存使用者的聯絡資訊、行事曆、照片、文件和更多項目，並在其所有裝置間自動保持最新的資料。iCloud 也可供第三方 App 用來儲存和同步文件以及 App 資料的重要數值，視開發人員所定義而定。使用者透過 Apple ID 登入並選擇他們想要使用的服務來設定 iCloud。IT 管理者可透過設定描述檔來停用 iCloud 功能（包含「我的照片串流」、iCloud Drive 和「備份」）。該服務無法得知正在儲存的內容，並會以位元組集合的方式對所有檔案進行處理。

每個檔案被區分為區塊，並由 iCloud 使用 AES-128 以及從利用 SHA-256 的每個區塊內容衍生的密鑰進行加密。這些密鑰與檔案的元資料會由 Apple 儲存在使用者的 iCloud 帳號中。檔案的加密區塊會使用第三方的儲存服務（例如 Amazon S3 和 Windows Azure）進行儲存，不會含有任何使用者的識別資訊。

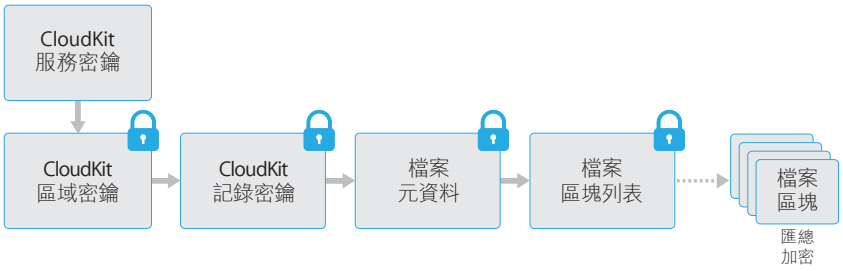
iCloud Drive

iCloud Drive 會加入以帳號為基礎的密鑰來保護儲存在 iCloud 中的文件。和現有的 iCloud 服務一樣，iCloud Drive 會將檔案內容分塊並進行加密，然後使用第三方的服務來儲存這些加密區塊。不過，檔案內容密鑰是由記錄密鑰所封裝，與 iCloud Drive 元資料儲存在一起。而這些記錄密鑰則由使用者的 iCloud Drive 服務密鑰所保護，儲存在使用者的 iCloud 帳號中。使用者可以藉由與 iCloud 進行認證來取用其 iCloud 文件元資料，但也必須擁有 iCloud Drive 服務密鑰才能顯示 iCloud Drive 儲存空間中受保護的部分。

CloudKit

CloudKit 允許 App 開發者在 iCloud 中儲存鍵值資料、結構資料和資產。對 CloudKit 的存取是使用 App 授權來加以控制。CloudKit 同時支援公用和專用資料庫。公用資料庫是由 App 的所有拷貝使用，通常用於一般資產，且並未加密。專用資料庫則會儲存使用者的資料。

如同使用 iCloud Drive 一樣，CloudKit 會使用以帳號為基礎的密鑰來保護存放在使用者專用資料庫中的資訊，就像其他 iCloud 服務，檔案會使用第三方的服務加以切割、加密並儲存。CloudKit 使用的是階層式密鑰（與「資料保護」類似）。檔案專屬密鑰是以 CloudKit 記錄密鑰加以封裝。「記錄」密鑰則會依序由整個區域的密鑰加以保護，其受到使用者的 CloudKit 服務密鑰所保護。CloudKit 服務密鑰會存放在使用者的 iCloud 帳號，且僅可在使用者已向 iCloud 認證後才可使用。



iCloud 備份

iCloud 還會每天通過 Wi-Fi 備份資訊，包括裝置設定、App 資料、「相機膠卷」中的照片和影片，以及「訊息」App 中的對話。透過 Internet 傳送內容時，iCloud 會對其進行加密，以加密的格式儲存並使用安全代號進行認證，進而保護內容。只有當裝置處於鎖定狀態、連接到電源且可透過 Wi-Fi 連接 Internet 時，「iCloud 備份」才會進行。多虧 iOS 中所使用的加密技術，系統經過精心設計，既可保護資料安全，又能兼顧增量、自發式的備份和還原動作。

以下是 iCloud 備份的項目：

- 已購買的音樂、影片、電視節目、App 和書籍的相關資訊，但不包括已購買的內容本身
- 「相機膠卷」中的照片和影片
- 聯絡資訊、行事曆事件、提醒事項和備忘錄
- 裝置設定
- App 資料
- 加入到 iBooks 但未購買的 PDF 和書籍
- 通話記錄
- 主畫面和 App 佈局
- iMessage、文字簡訊（SMS）和 MMS 訊息
- 鈴聲
- HomeKit 資料
- HealthKit 資料
- Visual Voicemail

當檔案從鎖定裝置時無法取用的「資料保護」類別中製作時，其檔案專屬密鑰會使用「iCloud 備份」Keybag 中的類別密鑰進行加密。檔案會以其原始的加密狀態備份至 iCloud。在資料保護類別為「無保護」中的檔案會在傳輸期間進行加密。

「iCloud 備份」Keybag 內含每個「資料保護」類別的非對稱（Curve25519）密鑰，這些密鑰用於加密檔案專屬密鑰。有關備份 Keybag 和「iCloud 備份」Keybag 內容的更多資訊，請參閱「加密與資料保護」一節中的「鑰匙圈資料保護」。

備份集是儲存於使用者的 iCloud 帳號中，由使用者的檔案拷貝和「iCloud 備份」Keybag 組成。「iCloud 備份」Keybag 受到隨機密鑰的保護，其也會與備份集一起儲存。（使用者的 iCloud 密碼不會用於加密，因此更改 iCloud 密碼不會使現有的備份資料失效。）

當使用者的鑰匙圈資料庫備份至 iCloud 時，仍會受到與 UID 連結的密鑰保護。這樣可讓鑰匙圈只能回復至原先產生它的同一台裝置，這意味著任何人（包括 Apple）都無法讀取使用者的鑰匙圈項目。

回復後，備份的檔案、「iCloud 備份」Keybag 和 Keybag 的密鑰將會從使用者的 iCloud 帳號取回。「iCloud 備份」Keybag 使用其密鑰進行解密，然後 Keybag 中的檔案專屬密鑰則用於解密備份集中的檔案，這些檔案會被作為新檔案寫入到檔案系統中，進而根據其「資料保護」類別對其重新加密。

Safari 與 iCloud 鑰匙圈的整合
Safari 可以自動為網站密碼產生加密的高安全性隨機字串，然後將其儲存在鑰匙圈中並與您的其他裝置同步。鑰匙圈項目透過 Apple 伺服器在不同的裝置之間傳輸，但會嚴格進行加密，Apple 和其他裝置均無法讀取其內容。

iCloud 鑰匙圈

iCloud 鑰匙圈可讓使用者在 iOS 裝置和 Mac 電腦之間安全地同步其密碼，不會將此資訊提供給 Apple。除了強大的隱私保護和安全性，易用性和回復鑰匙圈的功能對「iCloud 鑰匙圈」的設計和架構也具有重要影響。「iCloud 鑰匙圈」由兩項服務組成：鑰匙圈同步和鑰匙圈恢復。

Apple 設計的「iCloud 鑰匙圈」和鑰匙圈恢復可確保使用者的密碼在下列情況下仍然受到保護：

- 使用者的 iCloud 帳號被盜。
- iCloud 遭到外部攻擊者或員工入侵。
- 第三方取用使用者帳號。

鑰匙圈同步

當使用者第一次啟用「iCloud 鑰匙圈」時，裝置將建立信任圈並為自己製作同步身份。同步身份包括專用密鑰和公用密鑰。同步身份的公用密鑰會置於信任圈中，該信任圈已經過兩次簽署：第一次由同步身份的專用密鑰簽署，第二次由來自使用者 iCloud 帳號密碼的非對稱橢圓金鑰（使用 P256）簽署。連同信任圈一起儲存的還有參數（隨機密鑰和反覆運算次數），用於製作以使用者 iCloud 密碼為基礎的密鑰。

已簽署的同步信任圈會置於使用者的 iCloud 密鑰值儲存區域。如果不知道使用者的 iCloud 密碼，就無法對其進行讀取，如果沒有信任圈成員同步身份的專用密鑰，就無法對其進行有效修改。

當使用者在其他裝置上啟用「iCloud 鑰匙圈」時，新裝置將在 iCloud 中通知使用者該裝置不是之前已建立的同步信任圈的成員之一。該裝置會製作其同步身份的成對密鑰組，然後製作應用程式申請單以請求加入該信任圈。該申請單包括裝置的同步身份公用密鑰，系統將要求使用者以其 iCloud 密碼進行認證。橢圓密鑰產生參數會從 iCloud 取回並產生用於簽署應用程式申請單的密鑰。最終，應用程式申請單會置於 iCloud 中。

當第一部裝置接收到應用程式申請單時，會顯示一則通知，讓使用者確認新裝置正在請求加入同步信任圈。該使用者輸入其 iCloud 密碼，應用程式申請單則藉由比對專用密鑰的簽名進行驗證。這樣便確認產生請求要加入信任圈的人員，在發出請求時輸入了使用者的 iCloud 密碼。

使用者核准將新裝置加入到信任圈後，第一部裝置會將新成員的公用密鑰加入到同步信任圈，使用其同步身份和來自使用者 iCloud 密碼的密鑰再次簽署。新的同步信任圈會置於 iCloud 中，該信任圈的新成員會以類似方式進行簽名。

假設簽名信任圈有兩個成員，並且每個成員擁有與其配對的公用密鑰。他們現在開始透過 iCloud 鍵值儲存空間來交換個別的鑰匙圈項目。如果兩個信任圈成員擁有相同的項目，將同步修改日期最近的項目。如果另一個成員擁有該項目並且修改日期相同，則這些項目將被略過。每個同步的項目會特別針對其傳送的目標裝置進行加密。其他裝置和 Apple 均無法對其進行解密。此外，加密的項目只會在 iCloud 中短暫存放；該項目會被同步的每個新項目所覆寫。

當新裝置加入同步信任圈時，將會重複該過程。例如，當第三部裝置加入時，另外兩名使用者的裝置上均會出現確認訊息。使用者可以從其中任一部裝置來核准新成員。隨著新的同級裝置加入，每部同級裝置均與新裝置進行同步，以確保所有成員擁有相同的鑰匙圈項目。

但是，整個鑰匙圈不會進行同步。某些項目僅限於特定裝置（例如，VPN 身份），這些項目不會離開裝置。僅具有 `kSecAttrSynchronizable` 屬性的項目會被同步。Apple 已經為 Safari 使用者資料（包括使用者名稱、密碼和信用卡卡號）、Wi-Fi 網路密碼以及 HomeKit 加密密鑰設定了此屬性。

此外，依照預設，第三方 App 所加入的鑰匙圈項目不會進行同步。將項目加入到鑰匙圈時，開發者必須設定 `kSecAttrSynchronizable`。

鑰匙圈恢復

鑰匙圈恢復功能讓使用者可以將其鑰匙圈交由 Apple 託管，但不允許 Apple 讀取密碼和鑰匙圈包含的其他資料。即使使用者只有一部裝置，鑰匙圈恢復也可以提供安全網來防止資料遺失。當 Safari 被用來為 Web 帳號產生隨機且安全的密碼時，這尤其重要，因為這些密碼的唯一記錄是在鑰匙圈中。

鑰匙圈恢復包含兩大基本要素：輔助認證和安全託管服務，後者是 Apple 專為支援此功能而建立的服務。使用者的鑰匙圈會使用安全密碼進行加密，只有在滿足一系列嚴格的條件時，託管服務才會提供鑰匙圈拷貝。

當「iCloud 鑰匙圈」開啟時，系統會要求使用者製作「iCloud 安全碼」。恢復託管的鑰匙圈需有此安全碼。依照預設，系統會要求使用者提供簡單的四位數安全碼數值。然而，使用者也可以自行指定較長的代碼或允許其裝置製作加密的隨機密碼，他們可以自行記錄和保存。

然後，iOS 裝置會匯出使用者的鑰匙圈拷貝，加密封裝至非對稱式 Keybag 的密鑰中，並將其放置在使用者的 iCloud 鍵值儲存區域中。Keybag 會以使用者的 iCloud 安全碼和儲存託管記錄的 HSM（硬體安全性模組）叢集公用密鑰進行封裝。這會變成使用者的「iCloud 託管記錄」。

如果使用者決定接受隨機加密的安全碼而不自行指定或使用四位數值，則不再需要託管記錄。相反地，iCloud 安全碼會用來直接封裝隨機密鑰。

除了建立安全碼，使用者必須註冊電話號碼。這是用來在鑰匙圈恢復期間提供第二層的身份認證。使用者將會收到一則簡訊，必須回覆此簡訊才能繼續恢復程序。

託管安全性

iCloud 為鑰匙圈託管提供了安全的基礎架構，可確保只有經過授權的使用者和裝置才能執行恢復作業。iCloud 背後部署的是硬體安全性模組（HSM）叢集。這些叢集會保護託管記錄。叢集的每位成員都有一個密鑰，用來對其監管的託管記錄進行加密，如前文所述。

若要恢復鑰匙圈，使用者必須使用其 iCloud 帳號和密碼進行身份認證，並對傳送至其註冊的電話號碼的訊息進行回覆。回復完成後，使用者必須輸入其 iCloud 安全碼。HSM 叢集會使用「安全遠端密碼」通訊協定（SRP）來驗證使用者是否知道其 iCloud 安全碼；安全碼本身不會傳送給 Apple。叢集的每個成員單獨驗證使用者是否未超過擷取記錄所允許的最大嘗試次數，如以下所述。如果多數成員同意，叢集會將託管記錄解除封裝並將其傳送至使用者的裝置。

接著，裝置會使用 iCloud 安全碼來將用於加密使用者鑰匙圈的隨機密鑰解除封裝。有了該密鑰，您便可解密從 iCloud 鍵值儲存空間取回的鑰匙圈，並將其回復到裝置上。最多允許對託管記錄認證和擷取 10 次。多次嘗試失敗後，記錄將被鎖定，使用者必須聯絡「Apple 支援」才能進行更多嘗試。第 10 次嘗試失敗後，HSM 叢集將銷毀託管記錄，鑰匙圈將會永久失去。這種方式以犧牲鑰匙圈資料為代價，防止強行取用擷取記錄。

這些規則已寫入 HSM 韌體程式碼中。允許更改韌體的管理存取卡已經被銷毀。任何嘗試更改韌體或取用專用密鑰的操作，都會導致 HSM 叢集刪除專用密鑰。萬一發生這種情況，受叢集保護的所有鑰匙圈持有人將會收到訊息，告知其已失去託管記錄。他們之後可以選擇重新登記。

Siri

使用者只需自然地開口說話，即可透過 Siri 來傳送訊息、排程會議、撥打電話以及執行其他操作。Siri 使用語音辨識、文字到語音轉換和用戶端-伺服器模型，來回應各種請求。Siri 支援的任務經過專門設計，可確保盡量只使用最少的個人資訊，並對這些資訊提供完善的保護。

Siri 開啟後，裝置將會製作隨機識別碼，以用於語音辨識和 Siri 伺服器。這些識別碼僅用於 Siri 內部，並用來改善服務。如果 Siri 隨後被關閉，裝置將會產生新的隨機識別碼，以便在 Siri 重新開啟時使用。

為了改進 Siri 的功能，裝置中的某些使用者資訊會被傳送給伺服器。這些資訊包括：音樂資料庫（歌曲名稱、演出者和播放列表）、「提醒事項」列表名稱以及「聯絡資訊」中定義的姓名和關係。裝置與伺服器所進行的所有通訊均透過 HTTPS 來完成。

啟動 Siri 對話後，系統會將使用者的名字和姓氏（來自「聯絡資訊」）以及約略的地理位置傳送至伺服器。這樣，Siri 便可以使用姓名回應或回答那些只需要大概位置的問題，例如，天氣相關資訊。

如果需要更精確的位置，例如，確定附近電影院的位置，伺服器將會要求裝置提供更精確的位置。以上範例說明了在預設情況下，資訊如何傳送至伺服器（情況僅限於為了處理使用者請求而有必要加以傳送）。在任何情況下，只要 10 分鐘內沒有任何動作，對話資訊就會被捨棄。

從 Apple Watch 使用 Siri 時，手錶會製作其本身隨機的專屬識別碼，如上所述。然而，其要求也會傳送已配對 iPhone 的 Siri 識別碼來提供該資訊的參考，而非再次傳送使用者的資訊。

使用者的說話內容錄音會被傳送至 Apple 的語音辨識伺服器。如果任務僅涉及聽寫，辨識出的文字將被傳送回裝置中。否則，Siri 會對文字進行分析，必要時會將其與來自裝置相關之描述檔的資訊相結合。例如，如果請求是「發訊息給媽媽」，系統將會使用從「聯絡資訊」上傳的關係和姓名。然後已確認動作的指令將被傳送回要執行指令的裝置。

許多 Siri 功能是由裝置依照伺服器的指示來完成的。例如，當使用者要求 Siri 朗讀收到的訊息時，伺服器就會告訴裝置朗讀其未讀訊息的內容。訊息的內容和傳送者資訊不會被傳送到伺服器。

使用者的語音錄音將被保存 6 個月，讓辨識系統能夠加以利用，以便更有效地理解使用者的語音內容。6 個月後，將會儲存另一份不含識別碼的拷貝，以供 Apple 持續改善和開發 Siri，保存時間最長為兩年。此外，某些引用音樂、運動隊伍和隊員以及商家或興趣點的錄音同樣會被儲存，以用於改善 Siri。

無需動手，透過語音也可以啟動 Siri。語音觸發偵測會在本機裝置上進行。在此模式下，只有當傳入的音訊模式十分符合指定觸發詞語的原聲時，Siri 才會啟動。偵測到語音觸發後，對應的音訊（包括後續的 Siri 指令）會傳送到 Apple 的語音辨識伺服器作進一步處理，此過程會遵循與透過 Siri 執行其他使用者語音錄音相同的規則。

接續互通

「接續互通」會善用如 iCloud、藍牙和 Wi-Fi 的技術，可讓使用者從一部裝置到另一部裝置繼續作業、撥打和接聽電話通話、傳送和接收文字訊息，以及共享行動網路的 Internet 連線。

Handoff

當使用者的 Mac 和 iOS 裝置彼此接近時，使用者可以使用 Handoff 功能，自動將正在處理的內容從一部裝置傳送到另一部裝置。使用者可以使用 Handoff 功能來切換裝置並立即繼續作業。

當使用者在第二部支援 Handoff 功能的裝置上登入 iCloud 時，兩部裝置會透過「Apple 推播通知服務」（APNs）來建立頻段外的低功耗藍牙 4.0 配對。各個訊息會採用與 iMessage 相似的加密方式。裝置配對後，每部裝置都會產生對稱的 256 位元 AES 密鑰，並儲存在裝置的鑰匙圈中。此密鑰用於加密和認證低功耗藍牙廣告，其會在 GCM 模式下使用 AES-256 並採用重播保護措施，將裝置目前的活動傳遞給其他已配對的 iCloud 裝置。裝置首次接收到來自新密鑰的廣告時，會建立與起始裝置之間的低功耗藍牙連線，並執行廣告加密密鑰的交換。此連線使用標準的低功耗藍牙 4.0 加密進行保護，和個別訊息的加密相同（與 iMessage 的加密方式類似）。在某些情況下，這些訊息會使用「Apple 推播通知服務」，而非低功耗藍牙。活動的承載資料會使用與 iMessage 相同的方式進行保護和傳輸。

在原生 App 和網站之間使用 Handoff 功能

Handoff 功能可允許 iOS 的原生 App 繼續存取由 App 開發者合法控制之網域中的網頁。Handoff 也允許原生 App 的使用者活動在網頁瀏覽器中繼續進行。

為了阻止原生 App 要求繼續存取不是由開發者控制的網站，App 必須證明對其要繼續存取的網域具有合法控制權。對網站網域的控制是透過共用的網頁憑證所使用的機制來建立。如需詳細資訊，請參閱「加密與資料保護」一節中的「取用 Safari 儲存的密碼」。在允許 App 接受使用 Handoff 功能的使用者活動前，系統必須驗證 App 的網域名稱控制。

使用 Handoff 功能傳送的網頁來源可以是任何採用了 Handoff API 的瀏覽器。當使用者檢視網頁時，系統會使用加密的 Handoff 廣告位元組來廣告網頁的網域名稱。只有使用者的其他裝置能夠解密該廣告位元組（如先前「Handoff」一節中所述）。

在接收裝置上，系統會偵測到已安裝的原生 App 接受了來自已廣告網域名稱的 Handoff，並將該原生 App 圖像顯示為 Handoff 選項。啟動後，原生 App 會接收完整的 URL 和網頁標題。瀏覽器中的其他資訊則不會被傳送到原生 App。

相反地，若 Handoff 接收裝置未安裝相同的原生 App，原生 App 可能會指定後援 URL。在此情況下，系統會將使用者的預設瀏覽器顯示為 Handoff App 選項（若該瀏覽器已採用 Handoff API）。請求使用 Handoff 時，系統會啟動瀏覽器並使用來源 App 提供的後援 URL。後援 URL 並不一定要限制為由原生 App 開發者控制的網域名稱。

使用 Handoff 傳送較大筆的資料

除了 Handoff 的基本功能外，部分 App 可能會選擇使用支援傳送大量資料的 API（透過 Apple 建立的點對點 Wi-Fi 技術，與 AirDrop 類似）。例如，「郵件」App 使用這些 API 來提供 Handoff 功能的支援，以傳送可能包含較大附件的郵件草稿。

當 App 使用此功能時，兩部裝置間會開始交換，如同使用 Handoff 傳送一樣（請參閱前面章節）。不過，在使用低功耗藍牙收到初始承載資料後，接收裝置會透過 Wi-Fi 來啟用新的連線。此連線會使用 TLS 加密（交換其 iCloud 身份憑證）。憑證中的識別標誌會針對每位使用者的身份進行驗證。其他承載資料會透過此加密的連線進行傳送，直到傳輸完成為止。

iPhone 行動網路通話中繼

當您的 Mac、iPad 或 iPod 位於與 iPhone 相同的 Wi-Fi 網路上時，可以使用 iPhone 行動網路連線來撥打和接聽電話。設定會要求使用相同的 Apple ID 帳號來同時登入 iCloud 和 FaceTime。

收到來電時，會透過「Apple 推播通知服務」（APNs）來通知所有已設定的裝置，每個通知都會使用與 iMessage 所用相同的端對端加密技術。位於相同網路上的裝置會顯示來電通知使用者介面。接聽電話時，會使用安全的點對點連線技術在兩部裝置間無縫傳輸 iPhone 的音訊。

送出的通話也將會透過「Apple 推播通知服務」中繼到 iPhone，並透過安全的點對點連結在裝置間傳輸音訊。

使用者可以在 FaceTime 設定中關閉「iPhone 行動網路通話」來停用裝置的電話中繼功能。

iPhone 訊息轉寄

「訊息轉寄」會自動將 iPhone 上收到的 SMS 文字簡訊傳送到使用者已登記的 iPad、iPod touch 或 Mac。每部裝置必須使用相同的 Apple ID 帳號來登入 iMessage 服務。當「SMS 訊息轉寄」啟用時，會透過輸入由 iPhone 產生的隨機六位數驗證碼來驗證登記作業。

一旦裝置完成連結，iPhone 便會加密傳入的 SMS 文字訊息並轉寄至每部裝置，此作業使用的方式如本文件的 iMessage 一節所述。回覆會被傳回到使用相同方式的 iPhone，然後 iPhone 可使用電信業者的 SMS 傳輸機制以文字訊息來傳送回覆。「訊息轉寄」的功能可在「訊息」設定中啟用或停用。

Instant Hotspot

支援 Instant Hotspot 的 iOS 裝置使用低功耗藍牙來搜尋裝置並與其進行通訊，前提是裝置必須使用相同的 iCloud 帳號進行登入。與 Instant Hotspot 相容且運行 OS X Yosemite 或更新版本的 Mac 電腦，可使用相同的技術來搜尋支援 Instant Hotspot 的 iOS 裝置，並與其進行通訊。

當使用者在進入 iOS 裝置上的 Wi-Fi 設定時，裝置會發出低功耗藍牙訊號，該訊號包含可被所有登入相同 iCloud 帳號之裝置接受的識別碼。該識別碼由綁定到 iCloud 帳號的 DSID（Destination Signaling Identifier）產生，並會定期更新。當其他登入相同 iCloud 帳號的裝置彼此接近且支援個人熱點時，這些裝置會偵測到訊號並加以回應，以表示其處於可用狀態。

當使用者選擇可用於個人熱點的裝置時，會向該裝置傳送打開「個人熱點」的請求。而該請求會透過加密的連結（使用標準低功耗藍牙加密方法）進行傳送；請求的加密方式與 iMessage 的加密方式類似。裝置接著會使用包含個人熱點連線資訊的相同訊息專屬加密方式，透過相同的低功耗藍牙連結進行回應。

Spotlight 建議

Safari 搜尋和 Spotlight 搜尋包含來自 Internet、iTunes、App Store、電影放映時間、周遭位置和其他項目的搜尋建議。

若要讓建議的相關性更貼近使用者，使用者內容和搜尋回饋會連同搜尋查詢要求一併傳送給 Apple。以搜尋要求傳送的內容可提供 Apple 以下資訊：i) 裝置的概略位置；ii) 裝置類型（如 Mac、iPhone、iPad 或 iPod）；iii) 用戶端 App，其可以為 Spotlight 或 Safari；iv) 裝置的預設語言和地區設定；v) 裝置上最近使用的三個 App；以及 vi) 匿名區段 ID。與伺服器所進行的所有通訊均透過 HTTPS 來加密。

為了協助保護使用者隱私，「Spotlight 建議」絕不會傳送精確位置，而是在傳送前先在用戶端上產生模糊位置。模糊的程度取決於對裝置所在位置所估算的人口密度；例如，在鄉村位置的模糊程度較高，而在市中心的模糊程度較低，因為那裡的使用者彼此之間的距離通常較靠近。此外，使用者可以在「設定」中關閉「Spotlight 建議」的「定位服務」，停止將所有位置資訊傳送給 Apple。若「定位服務」已停用，Apple 便可能使用用戶端的 IP 位址來推測概略位置。

匿名區段 ID 可讓 Apple 分析 15 分鐘的期間內所執行之查詢間的模式。例如，若使用者常在搜尋「咖啡店」後隨即又搜尋「咖啡店電話號碼」，Apple 便可嘗試讓電話號碼更容易在搜尋結果中出現。然而，不像多數的搜尋引擎，Apple 搜尋服務並不會在使用者的搜尋記錄中，使用持續性的個人識別碼來將查詢與使用者或裝置連結在一起；相反地，在捨棄該 ID 前，Apple 裝置會使用臨時的匿名區段 ID 來處理最多 15 分鐘的作業。

裝置上最近使用之三個 App 中的資訊會被納入作為額外的搜尋內容。為了保護使用者的隱私，只有在 Apple 維護之熱門 App 白名單中的 App，以及最近三小時內曾取用的 App 會被納入。

傳送給 Apple 的搜尋回饋可提供 Apple 以下資訊：i) 使用者動作之間的時間，如按下按鍵和選取結果；ii) 若有的話，所選取的「Spotlight 建議」結果；以及 iii) 本機上所選取的結果類型（如「書籤」或「聯絡資訊」）。與搜尋內容一樣，搜尋回饋並不會與個人或裝置連結在一起。

Apple 會保留「Spotlight 建議」的查詢、內容和回饋，最長保留 18 個月。只包含查詢、國家、語言、日期（單位至小時）和裝置類型的精簡記錄則最長保留兩年。IP 位址並不會隨查詢記錄加以保留。

在某些情況下，「Spotlight 建議」可能會將常見單字或詞句的查詢轉送到合格的合作廠商，以接收和顯示合作廠商的搜尋結果。合格的合作廠商並不會儲存這些查詢，且不會接收搜尋回饋。合作廠商也不會接收使用者的 IP 位址。與合作廠商所進行的通訊是透過 HTTPS 來加密。Apple 將會依據從重複搜尋所取得的位置、裝置類型和語言，來提供城市位置、裝置類型和用戶端語言給合作廠商作為搜尋內容。

您可在「設定」中關閉 Spotlight、Safari 或此兩者的「Spotlight 建議」功能。若替 Spotlight 關閉此功能，則 Spotlight 會回復到本機（僅限於裝置本身）的搜尋用戶端，並不會傳送資訊給 Apple。若在 Safari 中關閉此功能，使用者的搜尋查詢、搜尋內容和搜尋回饋便不會傳送給 Apple。

Spotlight 也包含商家，以讓當地和裝置上的內容可供搜尋：

- CoreSpotlight API，可讓 Apple 和第三方 App 將可編列索引的內容傳遞至 Spotlight。
- NSUserActivity API，可讓 Apple 和第三方 App 將使用者所參訪的 App 頁面相關資訊傳遞至 Spotlight。

Spotlight 也會使用這兩種方式來維護所收到的裝置上資訊索引，因此可顯示此資料的結果來回應使用者的搜尋，或於 Spotlight 啟動時自動顯示。還有一個裝置上相關連的搜尋 API，僅可用於 Apple 提供的 App，可讓 Spotlight 將使用者搜尋查詢傳遞至 App 進行處理，並接收其結果。

裝置控制

iOS 支援具彈性的安全性原則和設定，讓使用者容易實施並管理。這可讓各機構保護公司資訊並確保員工遵守企業要求。例如，作為「員工自攜裝置」（BYOD）計畫的一部分，員工甚至可以使用自己攜帶的裝置。

公司可以使用密碼保護、設定描述檔、遠端清除和第三方 MDM 解決方案等資源來管理裝置流通並協助確保公司的資料安全，甚至在員工使用私人的 iOS 裝置取用資料時，亦能保障安全。

密碼保護

依照預設，使用者的密碼可定義為數值的 PIN。在具備 Touch ID 的裝置上，密碼最小長度為六位數。在其他裝置上，最小長度為四位數。使用者可在「設定」>「密碼」中的「密碼選向」中選取「自定英數密碼」，來指定較長的英數字元密碼。較長且複雜的密碼比較難以猜測或攻擊，建議企業使用此類密碼。

管理者可以使用 MDM 或 Exchange ActiveSync，或是要求使用者手動安裝設定描述檔，來強制實施複雜密碼要求和其他原則。可使用下列密碼原則：

- 允許簡易數值
- 需要英數數值
- 最短密碼長度
- 最短複雜字元數量
- 最長密碼使用期限
- 密碼總覽
- 自動鎖定逾時
- 無需密碼的裝置鎖定時間
- 嘗試失敗的最大數
- 允許 Touch ID

如需關於每個原則的詳細資訊，請參閱位於 developer.apple.com/library/ios/featuredarticles/iPhoneConfigurationProfileRef 的「Configuration Profile Key Reference」（設定描述檔密鑰參考）文件。

iOS 配對機型

iOS 使用配對機型來從主機電腦控制裝置取用權。配對會透過公用密鑰交換來在裝置與其連接的主機之間建立信任關係。iOS 會憑藉這種信任關係來在與所連接的主機之間啟用附加功能，例如資料同步。在 iOS 9 中，需要配對的服務會等到裝置已由使用者解鎖後才會啟動。

配對程序需要使用者解鎖裝置並接受來自主機的配對要求。使用者完成此步驟後，主機和裝置會交換並儲存 2048 位元的 RSA 公用密鑰。主機會得到 256 位元密鑰，可解鎖儲存在裝置上的託管 Keybag（請參閱 Keybag 章節中的「託管 Keybag」）。在裝置將受保護的資料傳送到主機或啟動服務（iTunes 同步、檔案傳送、Xcode 開發等）前，需要使用交換的密鑰來啟動加密 SSL 工作階段。裝置需要透過 Wi-Fi 與主機連線，以將此加密工作階段用於所有通訊，因此在這之前必須先透過 USB 配對。配對也會啟用一些診斷功能。在 iOS 9 中，若配對記錄已超過六個月未使用，便會過期。如需更多資訊，請參訪 support.apple.com/kb/HT6331?viewlocale=zh_TW。

部分服務（包含 com.apple.pcapd）會限制為僅能透過 USB 執行。此外，需要安裝 Apple 簽署的設定描述檔才能使用 com.apple.file_relay 服務。

使用者可以使用「重置網路設定」或「重置定位服務與隱私權」選項來清除信任的主機列表。如需更多資訊，請參訪 support.apple.com/kb/HT5868?viewlocale=zh_TW。

設定強制執行

設定描述檔為 XML 檔案，可讓管理者分配設定資訊到 iOS 裝置。使用者無法更改由已安裝的設定描述檔定義的設定。如果使用者刪除設定描述檔，亦會移除描述檔所定義的所有設定。如此一來，管理者便可以結合原則和取用權來強制執行設定。例如，提供電子郵件設定的設定描述檔也可以指定裝置密碼原則。使用者的密碼必須符合管理者的需求，否則無法取用郵件。

iOS 設定描述檔包含幾項可指定的設定，包含：

- 密碼原則
- 裝置功能的限制（例如，停用相機）
- Wi-Fi 設定
- VPN 設定
- 郵件伺服器設定
- Exchange 設定
- LDAP 目錄服務設定
- CalDAV 行事曆服務設定
- Web Clip
- 憑證和密鑰
- 進階行動網路設定

設定描述檔可以簽署並加密來驗證其來源，以確保其正當性並保護內容。設定描述檔使用支援 3DES 和 AES-128 的 CMS（RFC 3852）進行加密。

設定描述檔也可以鎖定到裝置上，以徹底防範遭移除，或要求輸入密碼才能移除。由於許多企業的使用者皆持有私人的 iOS 裝置，可以移除將裝置綁定至 MDM 伺服器的設定描述檔，但這麼做也會移除所有受管理的設定資訊、資料和 App。

使用者可以使用 Apple Configurator 直接在裝置上安裝設定描述檔，也可以透過 Safari 下載、傳送電子郵件，或使用 MDM 伺服器以無線方式傳送來取得設定描述檔。

行動裝置管理（MDM）

iOS 支援 MDM，可讓企業在組織內安全地設定和管理大量的 iPhone 和 iPad 部署。MDM 功能內建於現有的 iOS 技術，如設定描述檔、無線登記和 Apple 推播通知服務（APNs）。例如，APNs 可用來喚醒裝置，以便其可透過安全連線與 MDM 伺服器直接進行通訊。機密或所有權資訊並不會透過 APNs 進行傳輸。

藉由使用 MDM，IT 部門便可以為企業環境中的 iOS 裝置登記、以無線方式設定配置和更新設定、監控公司政策的遵守狀況，甚至可以遠端清除或鎖定受管理的裝置。如需更多行動裝置管理的相關資訊，請參訪 www.apple.com/iphone/business/it/management.html。

裝置登記方案

「裝置登記方案」（DEP）提供快速又流暢的方式來讓公司部署直接向 Apple 或 Apple 授權的零售商和電信業者購買的 iOS 裝置。在使用者得到裝置前，公司可以在 MDM 中自動登記裝置，無需實際操作或準備。藉由移除「設定輔助程式」中的特定步驟，使用者的設定程序可以更加簡化，方便他們快速使用。管理者也可以控制使用者是否可從裝置上移除 MDM 描述檔，並確定裝置限制從一開始時即已就緒。例如，他們可以向 Apple 訂購裝置，設置好所有管理設定，然後將裝置直接寄送到使用者的住家地址。裝置經開箱和啟用後，會於組織的 MDM 中進行登記，使用者便可開始使用所有管理設定、App 和書籍。

程序很簡單：在方案中登記後，管理者會登入方案網站，將方案連結到 MDM 伺服器，然後對透過 Apple 購買的 iOS 裝置進行「宣告」。接著便可透過 MDM 將裝置指定給使用者。使用者一經指定後，任何 MDM 指定的設定、限制或控制項目便會自動安裝。如需更多資訊，請參訪 deploy.apple.com。

注意：「裝置登記方案」無法在部分國家或地區使用。

Apple Configurator

除了 MDM 以外，OS X 的 Apple Configurator 也可讓任何人輕鬆部署 iOS 裝置。Apple Configurator 可以使用 App、資料、限制和設定來快速設定大量的裝置。

監管

在裝置設定期間，組織可設定要監管的裝置。監管意味著裝置列入機構所有，這樣會對其設定和限制提供額外的控制權。透過「裝置登記方案」或 Apple Configurator，裝置可在設定期間加以監管。

如需關於使用 MDM 和 Apple Configurator 設定和管理裝置的資訊，請參訪 iOS Deployment Reference（iOS 部署參考）網站：help.apple.com/deployment/ios。

如需對受監管裝置之其他控制的相關資訊，請參閱 Configuration Profile Reference：developer.apple.com/library/ios/featuredarticles/iPhoneConfigurationProfileRef/iPhoneConfigurationProfileRef.pdf.

裝置限制

管理者可以藉由安裝設定描述檔來限制裝置的功能。部分可用限制包含：

- 允許安裝 App
- 允許信任企業 App
- 允許使用相機
- 允許 FaceTime
- 允許螢幕快照
- 允許於鎖定時語音撥號
- 允許漫遊時自動同步
- 允許 App 內購買
- 允許同步「郵件」的最近郵件
- 強制使用者輸入商店密碼以進行所有購買
- 允許裝置鎖定時使用 Siri
- 允許使用 iTunes Store
- 在未管理目標中允許來自已管理來源的文件
- 在已管理目標中允許來自未管理來源的文件
- 允許 iCloud 鑰匙圈同步
- 允許以無線方式更新憑證信任資料庫
- 允許在鎖定畫面顯示通知
- 強制使用 AirPlay 連線來使用配對密碼
- 允許 Spotlight 顯示 Internet 上使用者產生的內容
- 在 Spotlight 中啟用 Spotlight 建議
- 允許 Handoff
- 將 AirDrop 視為未託管的目標
- 允許備份企業級書籍
- 允許在使用者的裝置之間同步企業級書籍中的筆記和重點
- 允許使用 Safari
- 啟用 Safari 自動填寫
- 強制執行詐騙網站警告
- 啟用 JavaScript
- 限制 Safari 中的廣告追蹤
- 阻擋彈出式項目
- 接受 Cookie
- 允許 iCloud 備份
- 允許 iCloud 文件和鍵值同步
- 允許 iCloud 照片共享
- 允許將診斷資料傳送至 Apple
- 允許使用者接受不受信任的 TLS 憑證
- 強制加密的備份
- 允許 Touch ID
- 允許從鎖定畫面取用「控制中心」
- 允許從鎖定畫面顯示「今天」顯示方式
- 要求 Apple Watch 手腕偵測

僅受監管的限制

- 允許 iMessage
- 允許移除 App
- 允許手動安裝設定描述檔
- HTTP 的全域網路代理伺服器
- 允許配對電腦以同步內容
- 以白名單和選用的連線密碼來限制 AirPlay 連線
- 允許 AirDrop
- 允許「尋找我的朋友」修改
- 對部分管理的 App 允許自發性單一 App 模式
- 允許帳號修改
- 允許行動數據修改
- 允許主機配對（iTunes）
- 允許啟動鎖定
- 避免清除所有內容和設定
- 避免啟用限制
- 第三方內容過濾器
- 單一 App 模式
- 總是開啟 VPN
- 允許修改密碼
- 允許配對 Apple Watch
- 允許自動 App 下載
- 允許鍵盤預測、自動更正、拼字檢查及快速鍵

如需有關限制的更多資訊，請參閱 developer.apple.com/library/ios/featuredarticles/iPhoneConfigurationProfileRef/iPhoneConfigurationProfileRef.pdf

遠端清除

管理者或使用者可以遠端清除 iOS 裝置。藉由從 Effaceable Storage 安全地刪除區塊儲存裝置加密密鑰，讓所有資料無法讀取，即可執行立即遠端清除。遠端清除可由 MDM、Exchange 或 iCloud 起始。

當 MDM 或 iCloud 觸發遠端清除指令時，裝置會傳送確認通知並執行清除作業。若是透過 Exchange 執行遠端清除，裝置會在執行清除之前登入 Exchange 伺服器。

使用者也可以使用「設定」App 來清除他們的裝置。如前面提到的，可以將裝置設定為在連續多次輸入密碼失敗後，自動清除裝置。

尋找我的 iPhone 與啟用鎖定

如果裝置遺失或遭竊，請務必停用並清除裝置。在安裝 iOS 7 或以上版本的裝置上，當「尋找我的 iPhone」啟用時，若沒有輸入持有人的 Apple ID 憑證，便無法重新啟用裝置。建議組織監管其裝置，或實施守則要求使用者停用此功能，如此一來，「尋找我的 iPhone」便不會使組織無法將裝置指定給其他人。

在安裝 iOS 7.1 或以上版本的裝置上，當使用者啟用「尋找我的 iPhone」時，相容的 MDM 解決方案可在監管的裝置上啟用「啟用鎖定」。MDM 管理者可以使用 Apple Configurator 或「裝置登記方案」來監管裝置，進而管理「尋找我的 iPhone」啟用鎖定。MDM 解決方案接著便會在「啟用鎖定」啟用時，儲存略過代碼，並於稍後需要清除裝置並指定給新的使用者時，使用此代碼來自動清除「啟用鎖定」。請參閱您的 MDM 解決方案文件以取得詳細資訊。

重要事項：依照預設，使用者即使開啟了「尋找我的 iPhone」，監管的裝置也不會啟用「啟用鎖定」。但是，MDM 伺服器可能會擷取略過代碼並在裝置上允許「啟用鎖定」。若在 MDM 伺服器啟用「啟用鎖定」時，「尋找我的 iPhone」已開啟，「啟用鎖定」便會在此時啟用。如果 MDM 伺服器啟用「啟用鎖定」時，「尋找我的 iPhone」為關閉狀態，則會在下一次使用者啟用「尋找我的 iPhone」時啟用「啟用鎖定」。

隱私控制

Apple 十分重視客戶的隱私，且具備許多內建控制項目和選項，可讓 iOS 使用者決定 App 可以如何使用其資訊、於何時使用，以及可以使用哪些資訊。

定位服務

定位服務會使用 GPS、藍牙、眾人共享 Wi-Fi 熱點和基地台位置來判定使用者的約略位置。可以使用「設定」中的單一切換來關閉定位服務，使用者也可以對使用定位服務的各個 App 核准此功能。使用者可以讓 App 只有在使用時才要求接收位置資料，或是隨時都可以接收位置資料。使用者可以選擇不允許使用定位服務，也可以隨時在「設定」中更改選擇。視 App 要求的定位用途而定，在「設定」中可以將定位服務設為永不允許、在使用時允許或總是允許。此外，當有權隨時使用定位服務的 App 在背景中運作時，系統會提醒使用者已核准定位服務，且可以更改 App 的取用權。

另外，使用者可細微控制系統服務對定位資訊的使用權。這可讓使用者關閉以下資訊中所涵蓋的定位資訊：診斷資訊和用量服務所收集的資訊中涵蓋的定位資訊（供 Apple 用於改進 iOS）、與位置相關的 Siri 資訊、「Spotlight 建議」搜尋結果中與位置相關的內容、當地交通狀況，以及用於評估路程時間的常用位置資訊。

取用個人資料

iOS 可防止 App 未經授權取用使用者的個人資訊。此外，使用者可以在「設定」中查看哪些 App 有權取用特定資訊，亦可授予或撤銷往後的任何取用權。包含以下項目的取用權：

- 聯絡資訊
- 行事曆
- 提醒事項
- 照片
- iPhone 5s 或較新機型上的運動記錄
- 社群媒體帳號（例如 Twitter 和 Facebook）
- 麥克風
- 相機
- HomeKit
- HealthKit
- 藍牙共享

如果使用者登入 iCloud，依照預設便會授予 App 取用 iCloud Drive 的權限。使用者可以在「設定」中控制每個 App 的 iCloud 取用權。此外，iOS 所提供的取用限制可防止資料在 MDM 和使用者所安裝的 App 和帳號之間移動。

隱私權政策

Apple 的隱私權政策可於線上取得，網址為：www.apple.com/tw/legal/privacy。

結論

對安全性的承諾

Apple 致力於以最先進的隱私與安全性技術保護個人資訊，進而保障客戶的安全；並採用全方位的方式來保護企業環境中的公司資料。

iOS 本身便具有安全性防護。囊括平台、網路，再到 App，企業所需的一切都可以在 iOS 平台上完成。這些元素造就了 iOS 在業界安全性的領先地位，同時兼顧優良的使用者體驗。

Apple 在 iOS 和 iOS App 生態系統之間運用一致且統合的安全性基礎架構。硬體式儲存加密可在裝置遺失時提供遠端清除的功能；並讓使用者在將裝置贈予或轉讓給其他人時，徹底移除所有公司和個人資訊。診斷資訊也會以匿名方式收集。

Apple 在設計 iOS App 時便將提昇安全性納入考量。Safari 可支援線上憑證的狀態通訊協定（OCSP）、EV 憑證和憑證驗證警告以進行安全瀏覽。「郵件」可透過 S/MIME 來從受認證和加密的郵件取得憑證，其可對單一郵件執行 S/MIME，這樣 S/MIME 使用者便可選擇依照預設總是簽署和加密郵件，或是選擇性控制保護個別郵件的方式。iMessage 和 FaceTime 也提供了用戶端對用戶端的加密。

對於第三方的 App，結合必要的程式碼簽署、Sandbox 技術限制和授權可為使用者提供強力的保護，免受病毒、惡意軟體和其他入侵程式的危害，因為這些程式可能會影響到其他平台的安全性。App Store 的提交流程會先審核每個 iOS App，通過審核後才能販售，這個流程可以進一步保障使用者免受這些風險的威脅。

為了充分利用 iOS 與生俱來的強大安全性功能，我們建議企業審核其 IT 部門和安全性原則，以確保可充分使用到此平台所提供的多重安全性技術。

Apple 擁有一個專業的安全性團隊，專門為所有的 Apple 產品提供支援。這個團隊會為開發中和已發佈的產品提供安全性審核和測試。Apple 團隊亦提供安全性工具和訓練，並積極監控新的安全性問題和威脅報告。Apple 為「資安事件應變小組論壇」（Incident Response and Security Teams, FIRST）的成員。若要深入瞭解如何向 Apple 提報問題以及訂閱安全性通知的相關資訊，請參訪 apple.com/support/security。

詞彙表

位址空間配置隨機載入（ASLR）	iOS 使用的一項技術，可讓透過軟體漏洞作亂的惡意程式成功率大幅降低。藉由確保記憶體位址和位移無法預測，入侵程式代碼便無法對這些值進行硬式編碼。在 iOS 5 或以上版本中，所有系統 App 和資料庫的位置都是隨機安排的，而所有第三方 App 均編譯為不受位置限制即可執行。
Apple 推播通知服務（APNs）	由 Apple 提供的全球性服務，可傳送推播通知到 iOS 裝置。
開機 ROM	裝置在第一次啟動時，由處理器所執行的第一個程式碼。作為處理器的必要部分之一，無論是 Apple 或攻擊者皆無法對其進行修改。
資料保護	iOS 的檔案和鑰匙圈保護機制。也可以指 App 用來保護檔案和鑰匙圈項目的 API。
裝置韌體升級（DFU）	裝置的開機 ROM 程式碼在等待透過 USB 回復時所處的模式。螢幕在 DFU 模式時為黑色，但在連接到正在執行 iTunes 的電腦時，便會顯示以下提示：「iTunes 偵測到一台在恢復模式中的 iPad。若要與 iTunes 一同使用，您必須回復 iPad。」
ECID	每部 iOS 裝置處理器中的一個 64 位元唯一識別碼。用作個人化程序的一部分，此識別碼並非機密。
Effaceable Storage	NAND 儲存區中專門用於儲存加密編譯密鑰的區域，可以直接定址和安全清除。若攻擊者實際持有裝置，Effaceable Storage 便無法提供保護，但其中的密鑰可以用作密鑰階層的一部分，以執行快速清除並提高安全性。
檔案系統密鑰	用於加密每個檔案元資料的密鑰，包含其類別密鑰。檔案系統密鑰會保存在 Effaceable Storage 中以進行快速清除，而不被視為機密。
群組識別碼（GID）	類似 UID，同一類別中每個處理器的 GID 皆相同。
硬體安全模組（HSM）	專門用來防止竄改的電腦，可保護並管理數位密鑰。
iBoot	由 LLB 載入的代碼，接著會載入 XNU，作為安全啟動鏈的一部分。
識別服務（IDS）	Apple 的 iMessage 公用密鑰、APNs 位址、電話號碼和電子郵件位址的目錄，用於查詢密鑰和裝置位址。
積體電路（IC）	亦稱為微晶片。
聯合測試工作群組（JTAG）	程式設計師和電路開發者使用的標準硬體除錯工具。
Keybag	用於儲存一組類別密鑰的資料結構。每種類型（系統、備份、託管或 iCloud 備份）的格式皆相同： • 標題包含以下內容： – 版本（在 iOS 5 中設為 3） – 類型（系統、備份、託管或 iCloud 備份） – Keybag UUID – 如果 Keybag 已簽署則包含 HMAC – 用於封裝類別密鑰的方式：與 UID 或 PBKDF2，以及 salt 和反覆運算次數相配合 • 類別密鑰列表： – 密鑰 UUID – 類別（所屬的檔案或鑰匙圈資料保護類別） – 封裝類型（僅限 UID 衍生密鑰；UID 衍生密鑰和密碼衍生密鑰） – 封裝的類別密鑰 – 非對稱式類別的公用密鑰

鑰匙圈	iOS 和第三方 App 用來儲存和擷取密碼、密鑰和其他敏感性憑證的基礎架構和 API 組。
密鑰封裝	使用一個密鑰來加密另一個密鑰。iOS 依照 RFC 3394 使用 NIST AES 密鑰封裝。
Low-Level Bootloader（LLB）	由開機 ROM 呼叫的代碼，接著會載入 iBoot，作為安全啟動鏈的一部分。
檔案專屬密鑰	用於在檔案系統上加密檔案的 AES 256 位元密鑰。檔案專屬密鑰由類別密鑰封裝，且儲存在檔案的元資料內。
佈建描述檔	一個由 Apple 簽署的 plist，其包含一組實體和授權，允許在 iOS 裝置上安裝並測試 App。開發佈建描述檔會列出開發人員選擇要用來臨機操作分配的裝置；分配佈建描述檔中包含企業開發 App 的 App ID。
指紋紋路角度對應	從指紋的一部分擷取出，描述紋路走向和寬度的數學表示式。
智慧卡	一種積體、內嵌的電路，可提供安全的識別、認證和資料儲存。
晶片式系統（SoC）	一種積體電路（IC），可將多重元件整合到單片晶片上。Secure Enclave 是 Apple 的 A7 或以上中央處理器內的 SoC。
Tangling	使用者的密碼轉換為加密編譯密鑰，並使用裝置 UID 強化的過程。這可確保暴力密碼破解攻擊必須在指定裝置上才能執行，進而降低發生率，且可避免多部裝置同時受到攻擊。Tangling 演算法為 PBKDF2，其使用 AES 密鑰搭配裝置 UID，作為每次反覆運算的偽隨機函數（PRF）。
統一資源識別碼（URI）	一個可識別網頁式資源的字元字串。
唯一識別碼（UID）	在製造過程便直接燒入每個處理器的 A 256 位元 AES 密鑰。唯一識別碼無法由韌體或軟體讀取，只能由處理器的硬體 AES 引擎使用。若要取得實際密鑰，攻擊者必須裝載極為複雜且昂貴的實體攻擊來入侵處理器的矽晶片。UID 與裝置上的任何其他識別碼均無關聯，包含但不限於 UDID。
XNU	iOS 和 OS X 作業系統中央的核心。預設為受信任的狀態，且會強制執行安全措施如程式碼簽署、Sandbox 技術限制、授權檢查和 ASLR。

文件版次歷史記錄

日期	摘要
2015 年 9 月	**為 iOS9 更新** • Apple Watch 啟用鎖定 • 密碼原則 • Touch ID API 支援 • A8 上的資料保護使用 AES-XTS • 非自行軟體更新的 Keybag • 認證更新 • 企業級 App 的信任模型 • Safari 書籤的資料保護 • App 傳輸安全性 • VPN 規格 • HomeKit 的 iCloud 遠端存取 • Apple Pay 酬賓卡 • Apple Pay 發卡機構的 App • Spotlight 裝置上索引編列 • iOS 配對機型 • Apple Configurator • 限制 • 如需有關 iOS 9 安全性內容的更多資訊，請參閱：support.apple.com/HT205212?viewlocale=zh_TW