蘇品長、賴怡聖、陳岳霖 (2024),「電腦資料交換之實體隔離機制探討─植基於身分驗證之 USB 存取管控研究」,*資訊管理學報*,第三十一卷,第一期,頁 93-121。

電腦資料交換之實體隔離機制探討─植基於身分驗證之 USB 存取管控研究

蘇品長
國防大學管理學院資訊管理學系

賴怡聖
國防大學管理學院資訊管理學系

陳岳霖*
國防大學管理學院資訊管理學系

摘要

為能杜絕來自網際網路的威脅並保護企業內部資料安全,多數企業採取實體隔離措施,並以集中式儲存媒體管控系統,結合白名單及權限管控進行管理,其中白名單係以硬體裝置序號為基礎,然考量裝置序號亦可透由人工燒錄而成,一旦有心人士偽冒為白名單 USB 裝置,則可在於企業內部主機獲合法資料存取。為能明確賦予企業內部公用 USB 裝置及公用電腦具備不可否認之數位身分,並以單次作業授權結合使用者資料交換作業行為,建立安全且可線上稽核之作業環境。本研究整合自我認證機制、橢圓曲線密碼系統及隨機背包難題之應用,建構具身分驗證之 USB 存取管控系統,管理者掌握管控內作業電腦及 USB 裝置授權資訊及稽核紀錄,使用者可依實務需求進行 USB 存取授權申請,並藉由橢圓曲線加密應用強化資料傳輸安全,落實裝置存取管理及後續稽核查驗,期能達到存取服務更具彈性及高安全性。

關鍵詞:身分認證、存取管控、自我認證、通用序列匯流排

* 本文通訊作者。電子郵件信箱:tomleo795@gmail.com
2023/02/10 投稿;2023/08/04 修訂;2023/11/07 接受

Su, P.C., Lai, Y.S., & Chen, Y.L. (2024). The Discussions in Physical Isolation of Computer Data: A Study on USB Devices Access Management Based on Identity Verification Mechanism. *Journal of Information Management*, 31(1), 93-121.

The Discussions in Physical Isolation of Computer Data: A Study on USB Devices Access Management Based on Identity Verification Mechanism

Pin-Chang Su
Department of Information Management, National Defense University

Yi-Sheng Lai
Department of Information Management, National Defense University

Yue-Lin Chen*
Department of Information Management, National Defense University

Abstract

In order to protect the internal data against the cyberthreats from the Internet, corporations mostly enforce the network isolation policy. Based on USB storage media control with a centralized management system, USB devices can be managed through a whitelist and permission access mechanism. In sight of the whitelist is based on the serial number of the hardware device, and that, those device serial numbers can be manually burned. What's more, a fake whitelisted USB device can be used to access internal computers legally. The research integrates the application of Self-Certified scheme, Elliptic Curve Cryptography and Random Knapsack mechanism, to ensure that the internal USB devices and computers for public use have an undeniable digital identity, constructing a USB access control system with Identity Verification. Besides, the administrator manages the authorization information and audit records of all internal computers and USB devices. Moreover, users can apply for USB access authorization according to practical needs. With the architecture of Elliptic Curve encryption, we can strengthen data transmission security to implement device access management and online audit. Through the control mechanism combining the one-time authorization and data exchange, in this way, we can establish a flexible and high security access services with a safer and auditable operating environment.

Keywords: Authentication, Access control, Self-Certification, USB device

* Corresponding author. Email: tomleo795@gmail.com
2023/02/10 received; 2023/08/04 revised; 2023/11/07 accepted

壹、導論

隨著時代的轉變，原本環境封閉的操作科技(Operation Technology, OT)環境，已經與資訊科技(Information Technology, IT)連接在一起，IT/OT 融合（IT/OT Convergence）指的是將 IT 和 OT 在組織內進行整合，前者涵蓋了用於數據處理、存儲和管理之計算能力與通信網路技術，後者涉及用於監控、控制與自動化處理過程(Paes et al. 2019)，例如在工廠自動化（Factory Automation, FA）等製程控制領域，IT/OT 融合提高作業效率，憑藉即時數據分析，優化整體運營，以創造出更大的生產價值(Felser, Rentschler, & Kleineberg 2019)，儘管許多組織已採取了積極的行動來加強 OT 環境的安全性，例如引入工業控制系統安全措施與融合資訊安全策略 IEC(International Electrotechnical Commission)62443 等國際標準，但仍有部分組織在這方面存在挑戰，這樣的情形不只在工業環境出現，倚賴各式電子設備的商務環境，也同樣面臨威脅。這一連串的問題，主要可區分成以下幾種面向所帶來的弱點：在 IT/OT 環境廣泛運用之資訊系統，例如 ERP(Enterprise Resource Planning)。雖然在 ISO/IEC27001 等國際標準中，針對資訊系統的安全性有相關規範，然而在融入 OT 等其他領域後，並無明確的規則可循。這種系統整合所涉及之安全性、網路架構以及操作差異等因素，可能帶來潛在風險(Paes et al. 2019)；人員的部分，除了內部員工沒有落實標準作業程序，也可能會在軟硬體的供應鏈中，出現不安全的操作。想要改善這樣的情況，首先要了解操作科技環境內的資產狀態，並且強化風險管理的能力；再者，制度面則要導入完整的權限與隔離制度，以及相關的標準、操作人員與外部廠商的資安意識等面向。因此，考量前列危安因素，政府機關或企業組織為確保資料安全，通常會要求存放國家、軍事及商務資訊等機密性資料之電腦主機限制連結對外網路，執行實體隔離的措施(資安人科技網 2021)。

一般來說，實體隔離作法不外乎限制連網能力、限制相關外接設備的使用(如磁碟機及光碟機)、使用 thin-client 等。限制連網的作法通常是不允許機密主機有存取網際網路的能力，或使其自成一個封閉式的網路環境。通用序列匯流排(Universal Serial Bus, USB)是目前用於電腦運算及物聯網設備支援最流行的通訊標準之一。近年來，USB 裝置已成為電腦周邊設備標準配備，傳統的 USB 攻擊是利用 USB 存儲攜帶惡意軟體並闖入氣隙網路(Air-Gap Network)，例如 Stuxnet。現代 USB 攻擊針對的是其規格或硬體中的缺陷，包括將惡意的 USB 功能注入到 USB 設備韌體中(Kim et al. 2022)，或透過 USB 端口途徑，從連接的設備中提取個人及機密資訊(Ayub et al. 2020)，故為落實 USB 裝置安全管控，企業多改採用集中式儲存媒體管控系統進行管理，透由裝置白名單及權限管控限制使用，考量裝置序號可由人工燒錄而成，無法作為唯一值進行設備檢查及驗證，但凡偵測程式確認介接之裝置序號與白名單註冊資訊相同，致生偽冒裝置合法存取內部資料風險，一旦肇生資安事件並進行相關事證蒐集時，無法有效發現事件與偽冒 USB 裝置之關聯，失去事件查處第一時機，故縱使 USB 裝置具備存儲容量大、傳輸

速度快等優點，但考量其安全方面的漏洞，在許多資訊機密和商業環境中仍被嚴格禁止使用(Haq, Wang, & Zhu 2019)。

USB 裝置由通用操作系統自動檢測和分類，無需經過任何身分驗證，這種默認信任設計原則很容易地導致許多攻擊，或惡意用戶(內部攻擊者)可能試圖繞過操作系統/內核強制保護機制(例如，通過操作系統替換)(Bhakte, Zavarsky, & Butakov 2016; Kang & Saiedian 2017)。攻擊 USB 成功的原因似乎是缺乏強大的身分驗證 USB 設備與僅具備不充分的防篡改解決機制(Baumann 2017; Tian et al. 2018; Meijer & Gastel 2019)，縱然針對 USB 裝置已存在數種驗證標準，如 USB Type-C Authentication 和 FIDO U2F (Universal 2nd Factor)等技術，然其目的在於提高 USB 設備身份驗證之安全性，唯實際成功應用，仍需考慮特定設備和平台支援程度及其相容性。而本研究聚焦於探討 USB 裝置存取管控線上稽核之可行性，將整合使用者、USB 裝置、作業電腦、管控軟體及認證中心等多個角色，並主要採用系統驗證方法，同時，專注於研究如何透過密碼技術來增強整體系統之安全性，以確保 USB 設備的存取管控與稽核過程，安全可靠。綜上，本研究將針對既有隨身碟管控系統，設計具自我認證及橢圓曲線密碼學之驗證機制，避免憑證中心私鑰遭偽造或竊取風險，確保資料之機密性及完整性。另運用於隨身碟管控系統之 USB 存取管控技術，同時驗證資料交換之電腦、USB 裝置及使用者身分，以集中式管控系統進行線上稽核，授權使用者單次作業權限，並導入伺服端自我認證機制，確認資料之不可否認性，以求符合 NIST(National Institute of Standards and Technology)SP800-53 存取控制標準。

貳、文獻探討

一、通用序列匯流排介紹及近年案例分析

通用序列匯流排，屬電腦與外部裝置介接之序列埠匯流排標準，亦為電腦輸出輸入介面的技術規範，為多數產品通用規格。至今多數企業採用集中式管控系統進行 USB 裝置管理，以裝置白名單進行管控，開放於指定電腦，搭配商用防毒產品進行掃毒，避免設備遭惡意程式感染致使資料外洩。每個 USB 裝置均有產品識別碼(Product ID, PID)及供應商識別碼(Vendor ID, VID)，理應可作為唯一識別使用，惟考量其安全性及識別度，學者 Verma & Singh (2012)提出 USB 裝置驗證模式，即阻擋所有 PID 及 VID 並開放白名單裝置 ID，精進 USB 存取管控成效。後續亦有學者 Mohammadmoradi & Gnawali (2018)提出植基於準確 USB 裝置特徵的指紋辨識方法，建立可信任之 USB 裝置清單作為白名單。儘管目前針對 USB 管控已有諸多防護措施，仍有疏漏防護不足之處，一旦駭客獲悉企業內部白名單清單並予以偽冒，系統經偵測確認裝置序號與白名單相符，即得以合法存取企業內部網路，以既有管控機制而言，系統無法辨識偽冒 USB 裝置與白名單裝置不同之處，成為資安管控罅隙。

早期隨身碟主要是透由 Autorun 方式感染電腦主機，演變至今，USB 攻擊手法變化萬千，依據 Tischer et al. (2016)研究測試發現，多數受測人員均認為電腦防護機制及防毒軟體可有效保護他們免於遭受惡意軟體攻擊，由此可知，好奇心容易成為駭客運用駭侵行為的突破口。侯皓薰 (2018)指出，因 USB 通訊協定特性，已衍生出多種新攻擊模式，包括 Android 手機攻擊及 USB 充電軟體病毒等攻擊手法，彙整近年常見案例如表 1；鑒於 USB 裝置使用便利且外型輕巧，易為有心人士運用之媒介，若無完善稽核管控機制，易成資安事件突破口，影響企業內部作業安全。學者 Bhakte et al. (2016)提出植基於 NIST SP800-53 標準之風險管控框架，包括權限控管、識別與稽核、檢查與保管、教育訓練及安全管控等項。截至目前，多數研究人員設法透由概念驗證的實機演練以識別威脅和漏洞，又以學者 Mamchenko & Sabanov (2019)蒐整各類 USB 攻擊手法並予以分析，可能運用的攻擊手法及影響層面，並預測未來可能運用於 USB 攻擊的媒介，認為可藉訂定良好管理措施，掌握標的物件的安全方法及對既有系統偵測的限制，可用於未來評估用以精進重要標的安全管控措施，以降低企業攻擊的影響範圍。

表 1：USB 資安事件近年案例彙整表

日期	類型	事件摘要(資料來源)	關係說明
2018 年	USB 病毒	總統府資安週，隨身碟贈品竟藏病毒(iThome-2019iT 邦幫忙鐵人賽，2018)	經查為 USB 裝置供應廠商所屬電腦已遭惡意程式「XtbSeDuA.exe」感染，於測試裝置容量與良率作業期間，將其與已感染主機對接，進而肇生大批隨身碟中毒情事。
2018 年	USB 病毒	深度剖析台積電產線中毒大當機始末(iThome-News，2018)	因新產線作業機臺安裝過程，未按照 SOP 程序先行實施掃毒(感染病毒為 WannaCry 變種)而釀災，再加上新機臺連接公司內部電腦網路而導致病毒擴散。
2022 年	USB 病毒	微軟公司警告，Raspberry Robin 蠕蟲病毒通過受感染的 USB 裝置傳播(TechNews 科技新報，2022)	Raspberry Robin 蠕蟲利用含有惡意 lnk 檔之感染 USB 隨身碟來散播至新 Windows 系統。當使用者將受感染 USB 裝置插入電腦並點擊連結時，蠕蟲會觸發 msiexec 程序，然後使用 cmd.exe 來啟動隨身碟中的惡意檔案，進而執行惡意封包負載。
2023 年	USB 病毒	新版 PlugX 惡意軟體，會藏於 USB 裝置內感染 Windows 系統(台灣電腦網路危機處理暨協調中心，2023)	新版 PlugX 利用 Unicode 字元，在 USB 裝置中新增一個資料匣(內含 desktop.ini 檔，以 USB 裝置圖示顯示，欺騙使用者)，當使用者點擊圖示時，啟動 cmd.exe 執行 x32.exe，即會讓 Windows 主機感染 PlugX 惡意軟體，若後續有新裝置接入，亦會遭受 PlugX 惡意軟體潛入安裝。

表 1：USB 資安事件近年案例彙整表(續)

日期	類型	事件摘要(資料來源)	關係說明
2023 年	USB 病毒	美國聯邦調查局（FBI）於 2022 年，針對 BadUSB 攻擊，發出警告。(Smartermsp 2023)	自 2021 年 8 月以來，FBI 接受到多份關於不明 USB 設備被寄送給美國境內運輸、保險與國防產業等相關企業之報告，若收件者將 USB 裝置插入所屬資訊設備，將自動啟用內嵌木馬程式，進而感染內部網路。企業需加強安全意識，以確保員工不會不慎插入未知的 USB 設備，以減少潛在的風險。
2023 年	USB 病毒	俄羅斯駭客使用 USB 裝置，傳播惡意軟體藉此攻擊烏克蘭政府和軍方，取得相關軍事機密 (SecurityWeek 2023)	俄羅斯駭客組織 Gamaredon，經知名資安公司-賽門鐵克(Symantec)指出使用新的 PowerShell 腳本，通過 USB 驅動程式傳播該組織自製的後門程式 Pterodo。該腳本將自身複製到受感染的資訊設備上，並轉移至可移動設備上，在受害者網絡中進行橫向移動，甚至可能用於幫助攻擊者接觸目標組織中實體隔離之機器。

二、NIST SP800-53

NIST SP800-53(Special Publication 800-53: Security and Privacy Controls for Federal Information Systems and Organizations)，是由美國國家標準與技術研究院（National Institute of Standards and Technology, NIST）所發布之標準文件，已於 2020 年 9 月更新至第五版(Rev 5)，最初用意為提供美國聯邦政府有關如何保護內部資訊系統與組織，其安全性及隱私之參考指南，後續通用於各類企業，成為最具代表性的資訊安全標準之一(Kurii & Opirskyy 2022)。

NIST SP800-53 中的存取控制要求項目，旨在確保資訊系統的存取安全受到適當保護(Force 2020)，相關條例摘述如次：

AC-1：政策與流程(Policy and Procedures)：組織應制定、維護和執行明確的存取控制政策與流程。

AC-2：帳戶管理(Account Management)：確保只有授權的用戶才能擁有系統帳戶，並管理這些帳戶的存取權限。

AC-3：監控執行(Access Enforcement)：實施帳戶活動監控，以及時檢測與回應不正常活動。

AC-5：權責劃分(Separation of Duties)：確保存取權責經適當分配，以減少潛在的濫用風險。

AC-7：未經授權的系統使用(Unsuccessful Logon Attempts)：警惕和預防未經授權的系統使用並實施適當措施來防止，如限制時間及次數等。

三、自我認證機制

在密碼學應用層面，公開金鑰基礎建設(Public Key Infrastructure, PKI)係透由認證中心將使用者的個人身分跟公開金鑰鏈結在一起，對每個憑證中心而言，使用者的身分必須是唯一，而憑證主要在於確認數位簽章的正確性及使用者身分的真實性，早期認證中心掌握使用者公開金鑰及私密金鑰資訊，當認證中心資料庫遭竊，使用者身分即有被偽冒的風險。

法國學者 Girault(1991)提出建構在公開金鑰密碼系統的自我認證機制(Self-Certified Scheme)，透由使用者與授權主機共同參與公鑰演算機制，無須倚賴第三方憑證中心參與使用者每次身分認證作業。該種互動模式允許使用者得以進行自我驗證，同時確保私鑰僅限於使用者本身知道，有效地避免憑證中心偽冒使用者身分的潛在風險，並且即使憑證中心資料外洩，他人也無法盜用使用者私鑰進行非法存取(蘇品長、夏君和、蘇泰昌 2022)。另外，此認證機制可有效減輕認證伺服器之金鑰存管負荷並可加速使用者身分認證效率。鑒於認證中心無法直接計算使用者密鑰資訊，僅有授權的證認中心方可產製憑證，當單一用戶同時存有 2 張以上憑證即為非法行為憑據，故偽冒使用者身分之行為是可被偵測得知。

四、橢圓曲線密碼系統

橢圓曲線密碼學(Elliptic Curve Cryptography, ECC)屬公開金鑰加密演算法，美國國家安全局(National Security Agency, NSA)於 2005 年宣布採用橢圓曲線密碼演算法作為美國政府標準，是目前公認在給定金鑰長度下最安全的加密演算法，其安全性條件相同下，橢圓曲線密碼系統僅須較小的金鑰長度，反之亦然，在同樣金鑰長度下，橢圓曲線密碼系統擁有更高的安全性，結果如表 2(蘇品長、葉昱宗 2017)所示。

表 2：RSA 和 ECC 在相同安全條件下金鑰長度之比較表

長度 / 項目	金鑰長度(bits)				
RSA	512	1024	2048	3072	7680
ECC	112	163	224	256	384
金鑰長度比 (RSA/ECC)	1：5	1：6	1：9	1：12	1：20

若q是大於 3 的質數，則在 Galois Field $E(F_q)$中，橢圓曲線之通式為$y^2 = x^3 + ax + b \bmod q$，其中 $0< x \leq q$，$a,\ b$為小於q的正整數，且$4a^3 + 27b^2 \bmod q \neq 0$。在橢圓曲線求點運算中，假設一個$\boldsymbol{E(F_q)}$的橢圓曲線，$\boldsymbol{P}$是其中的一個點，給定另一個點$\boldsymbol{Q}$若要找出一個整數$\boldsymbol{k}$使得$\boldsymbol{Q = kP}$，因為其特殊的點加法運算，破密者必須逐一的窮舉所有可能的點，直至目前為止，這仍是屬於無法於多項式時間內求解的問題(蘇品長、陳明心 2018)。

五、隨機背包理論

背包公鑰密碼系統，由學者 R. Merkle & M. Hellman 於 1978 年提出基於背包問題的公開密碼系統，利用難解背包問題作為公鑰，以超增序列背包問題作為私鑰，進行資料加解密，其優點為加解密速度快，惟超增序列數值易遭他人猜測破解，稱為低密度攻擊(Merkle & Hellman 1978)。

學者王保倉、韋永壯、朝予濮 (2010)提出「基於隨機背包的公鑰密碼」，該系統之加解密計算可透由加法與模減法運算，使其加解密速度較快；該算法是基於隨機背包問題而非為易解背包問題，藉以證明攻擊者若未能掌握私鑰，該密碼演算法能有效抵抗直接求解背包問題的攻擊。另學者費向東、丁燕艷、潘郁 (2012)提出針對王氏背包密碼系統之分析與改進，運用替代與混淆的技術，降低系統的冗餘度，防止攻擊者從公鑰出發利用初始序列及冗餘度完成破譯。後續學者王青龍、趙祥模 (2015)提出針對王保倉等人(2010)之隨機背包公鑰密碼系統的分析與改進之新型改進方案，改善隨機背包密碼系統之設計缺陷，並足以抵抗低密度攻擊及格基規約攻擊等常見攻擊，相較於王氏背包密碼系統更具代表性。

參、交易協定設計

本研究係運用橢圓曲線密碼系統之離散對數難題為基礎，結合單向雜湊演算法及隨機背包密碼系統，建立基於身分驗證之 USB 存取管控機制，採雙因子驗證(Two-factor Authentication)結合自我認證機制，對白名單 USB 裝置、作業電腦及指定人員帳號進行一次性授權認證，避免有心人士企圖以偽冒之 USB 設備、電腦或同仁帳號，介接企業網路進行非法資料存取，進而提升系統安全性。

本研究作業環境採用單一網路環境(實體隔離網路)，所有設備、主機、使用者與資源都處於同一個網域內(如國軍內部網路)，彼此間能相互交換資訊，僅限於該網域內人員間之通信及資源共享；另本研究系統架構預設搭配 VMware vSphere 虛擬化平台使用，將認證中心與隨身碟管控系統建置於虛擬主機群，藉其容錯及備援機制，以實現可用性目標(傅振華、張凱棠 2022)。本研究區分為系統初始階段、PKI 註冊階段、權限登記階段及執行階段等 4 個階段，另為提升本系統之安全性，單向雜湊演算法採用目前窮舉法無法破解之 SHA-256(黃建勛、蕭舜文 2022)；隨機背包密碼系統，採用王青龍、趙祥模於 2015 年所提出之改良式演算法。系統符號說明表如表 3、協定運作流程如圖 1 及整體運作示意圖如圖 2。

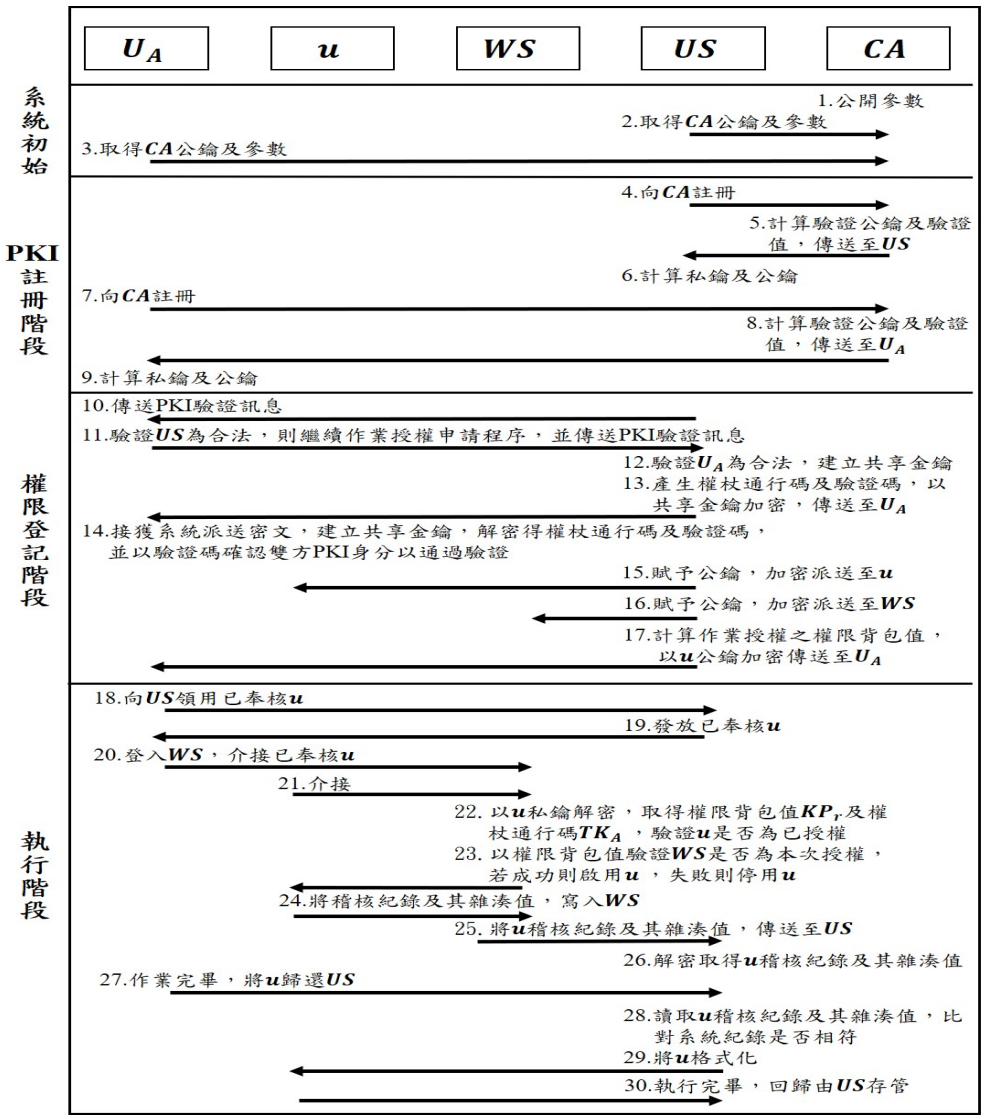

圖 1：協定運作流程

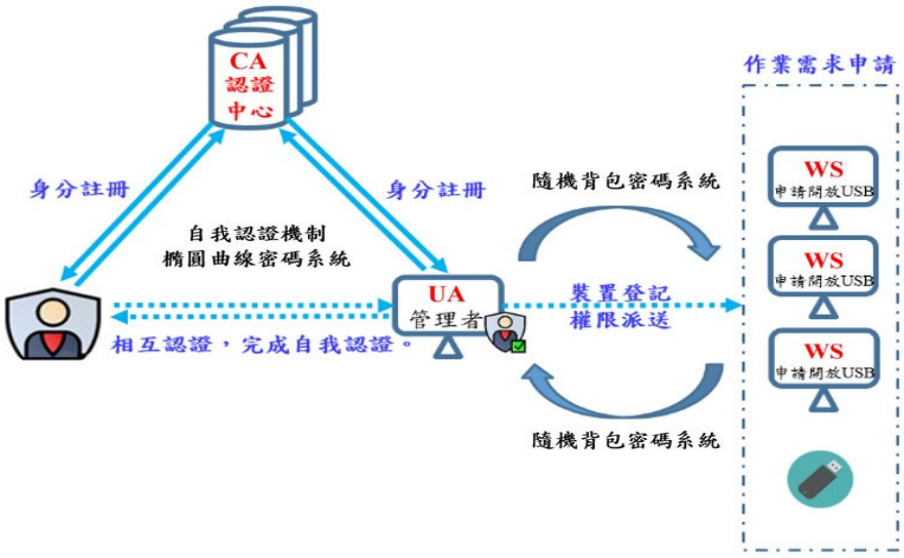

圖 2：整體運作示意圖

表 3：系統符號說明表

項次	符號	說明	項次	符號	說明
1	U_A	使用者 (User)	18	C_{KP}	權限背包密文 (Knapsack)
2	u	USB裝置	19	C_{log}	USB稽核紀錄密文
3	WS	作業電腦 (Workstation)	20	uid	USB裝置識別碼
4	US	隨身碟管控系統 (USB access control system)	21	$NCid$	作業電腦網卡識別碼
5	CA	憑證中心 (Certificate authority)	22	TK_A	權杖通行碼 (Token)
6	$E(F_q)$	位於有限域F_q之橢圓曲線	23	$K_{(US,A)}$ $K_{(A,US)}$	共享金鑰
7	n	階數 (Order)	24	sk_A, sk_{uid} sk_{NCid}, sk_{US}	各用戶端之私鑰資訊
8	G	橢圓曲線基點	25	PK_{U-} PK_{US-}	使用者及隨身碟管控系統之驗證公鑰
9	$h(.)$	產生簽名檔之單向雜湊函數	26	t_s, t_e	作業授權起訖時間
10	$h_1()$	轉換作業授權身分之單向雜湊函數	27	KP_r	權限背包值
11	id_A	使用者身分資料	28	SID_A	作業授權身分
12	id_{US}	隨身碟管控系統帳號	29	KP_{SID_A}	作業授權背包值
13	k_A	產生簽名之隨機參數	30	sk_{CA}	憑證中心私鑰
14	R	隨機參數	31	PK_{CA}	憑證中心公鑰
15	SIG_A	使用者的簽章	32	t	授權申請時戳亂數
16	V_A	使用者的簽名檔	33	r_A	存取授權序列
17	C_A	作業授權申請密文	34	PK_A, PK_{uid} PK_{NCid}, PK_{US}	各用戶端之公鑰資訊

一、初始階段

憑證中心CA選取一個長度為 224bits 以上大質數q，並在$E(F_q)$上選取一條安全的橢圓曲線，擇定一單向雜湊函數$h()$及私鑰sk_{CA}，計算公鑰PK_{CA}並公開系統參數$E(F_q)$、G、n、$h()$。

$$PK_{CA} = sk_{CA} \cdot G \tag{1}$$

二、註冊階段

由使用者U_A及隨身碟管控系統管理者US向CA註冊取得各自身分憑證，本階段使用者以王小美U_A為例說明，計算流程如次：

Step 1：王小美使用者帳號id_{A1}及個人密碼id_{A2}先行整合為註冊身分資料id_A，並擇一隨機參數k_A透過單向雜湊函數$h(\,)$產生簽名檔V_A，將(id_A, V_A)傳送至CA。

$$V_A = h(k_A \| id_A) \cdot G \tag{2}$$

Step 2：CA擇一隨機參數R計算U_A之驗證公鑰PK_{A-}及簽章SIG_A並回傳U_A。

$$PK_{A-} = V_A + (R - h(id_A)) \cdot G = (q_{ax}, q_{ay}) \tag{3}$$

$$SIG_A = R + sk_{CA}(q_{ax} + h(R)) \tag{4}$$

Step 3：U_A接收到CA回傳的驗證公鑰及簽章資料$(PK_{A-},\ SIG_A)$，計算本身私鑰sk_A，並驗證公鑰PK_A之正確性。

$$sk_A = SIG_A + h(k_A \| id_A) \tag{5}$$

$$PK_A = sk_A \cdot G \tag{6}$$

$$PK_A = PK_{A-} + h(id_A) \cdot G + (q_{ax} + h(id_A)) \cdot PK_{CA} \tag{7}$$

三、權限登記階段

U_A向US提出資料交換作業授權申請時，經雙方驗證成功則建立共享金鑰$K_{(US,A)}$及權杖通行碼TK_A；完成 PKI 身分驗證後解出作業授權背包值KP_{SID_A}，取得U_A作業授權申請之作業授權身分，賦予裝置公鑰並加密派送至指定裝置，另將本次作業授權申請之權限背包值KP_r及權杖通行碼TK'_A派送至U_A，步驟如次：

Step 1：用戶 PKI 驗證及使用者登記，U_A驗證US之 PKI 身分，經比對$PK'_{US} = PK_{US}$，確認US為合法 PKI 用戶，傳送U_APKI 資訊至US。

$$PK'_{US} = PK_{US-} + h(id_{US}) \cdot G + \left(q_{UAx} + h(id_{US})\right) \cdot PK_{CA} \tag{8}$$

US驗證U_APKI 身分，經比對$PK'_A = PK_A$，確認U_A為合法用戶，建立驗證碼TK_0及共享金鑰$K_{(US,A)}$。

$$PK'_A = PK_A \tag{9}$$

$$PK'_A = PK_{A-} + h(id_A) \cdot G + (q_{ax} + h(id_A)) \cdot PK_{CA} \tag{10}$$

$$K_{(US,A)} = sk_{US} \cdot PK_A \tag{11}$$

US產製本次作業授權之權杖通行碼TK_A，將TK_A與TK_0與$K_{(US,A)}$運算，以密文C_{A0}傳送至U_A。

$$TK_{A-} = (TK_A, TK_0) + K_{(US,A)} = (m_1, m_2) \tag{12}$$

$$C_{A1} = R' \cdot G \tag{13}$$

$$Y_A = \mathrm{E}(y_{A1}, y_{A2}) = R' \cdot PK_A \tag{14}$$

$$C_{A2} = (y_{A1} \cdot m_1 \bmod q,\ y_{A2} \cdot m_2 \bmod q) = (C_1, C_2) \tag{15}$$

$$C_{A0} = (C_{A1}, C_{A2}) \tag{16}$$

U_A接收密文C_{A0}，即完成通訊雙方身分確認，建立$K_{(A,US)}$並解密取得TK_A及TK_0。

$$K_{(A,US)} = sk_A \cdot PK_{US} \tag{17}$$

$$Z = sk_A \cdot C_{A1} = (Z_1, Z_2) \tag{18}$$

$$TK_{A-} = (C_1 \cdot Z_1^{-1} \bmod q,\ C_2 \cdot Z_2^{-1} \bmod q) = (m_1, m_2) \tag{19}$$

$$(TK_A, TK_0) = \mathrm{D}(m_1, m_2) - K_{(A,US)} \tag{20}$$

Step 2：建立存取授權序列，選取任一超遞增背包向量$\widehat{\boldsymbol{b}}=(\boldsymbol{b_1},\ \boldsymbol{b_2}\ \ldots,\boldsymbol{b_n})$，另任意選取兩個背包向量$\widehat{\boldsymbol{u}}=(\boldsymbol{u_1},\ \boldsymbol{u_2}\ \ldots,\ \boldsymbol{u_n})$，$\widehat{\boldsymbol{v}}=(\boldsymbol{v_1},\ \boldsymbol{v_2}\ \ldots,\ \boldsymbol{v_n})$，其中$\boldsymbol{u_i}$及$\boldsymbol{v_i}$均為正整數並滿足$\boldsymbol{b_i}=\boldsymbol{u_i}+\boldsymbol{v_i}$，$\boldsymbol{i}=1,\ldots,\boldsymbol{n}$。

任選兩個整數$\boldsymbol{m_{A1}}$及$\boldsymbol{m_{A2}}$，滿足$\boldsymbol{m_{A1}}\geq\sum_{i=1}^{n}\boldsymbol{u}$i，$\boldsymbol{m_{A2}}\geq\sum_{i=1}^{n}\boldsymbol{v}$i，滿足GCD($\boldsymbol{m_{A1}},\ \boldsymbol{m_{A2}}$)=1，以中國餘式定理計算存取授權序列向量$\widehat{\boldsymbol{A}}$，公式如下：$\widehat{\boldsymbol{A}}=(\boldsymbol{a_1},\ \boldsymbol{a_2},\ldots,\ \boldsymbol{a_n})$，$\boldsymbol{a_i}\equiv\boldsymbol{u_i}\ (\mathrm{mod}\ \boldsymbol{m_{A1}})$，$\boldsymbol{a_i}\equiv\boldsymbol{v_i}\ (\mathrm{mod}\ \boldsymbol{m_{A2}})$，$\boldsymbol{i}=1,\ldots,\boldsymbol{n}$。輸出存取授權序列公鑰$\widehat{\boldsymbol{A}}$，私鑰為$\boldsymbol{m_{A1}}$及$\boldsymbol{m_{A2}}$。存取授權序列$\boldsymbol{r_A}$以{0, 1}代表是否授權使用者註冊。

$$\boldsymbol{r_A}=(\boldsymbol{r_1},\ \boldsymbol{r_2},\ldots,\boldsymbol{r_n}),\ \boldsymbol{r_i}\in\{\boldsymbol{0},\boldsymbol{1}\}，且\boldsymbol{i}=\boldsymbol{1},\ldots,\ \boldsymbol{n} \tag{21}$$

Step 3：資料交換作業授權登記程序，以使用者身分$\boldsymbol{id_A}$、USB 裝置識別碼$\boldsymbol{uid_A}$、作業電腦網卡識別碼$\boldsymbol{NCid_A}$、作業授權時間之亂數時戳$\boldsymbol{t}$及起訖時間$(\boldsymbol{t_s},\boldsymbol{t_e})$計算使用者作業授權身分$\boldsymbol{SID_A}$。

$$\boldsymbol{SID_A}=(\boldsymbol{id_A}\parallel\boldsymbol{uid_A}\parallel\boldsymbol{NCid_A}\parallel\boldsymbol{t}\parallel\boldsymbol{t_s}\parallel\boldsymbol{t_e}) \tag{22}$$

將$\boldsymbol{SID_A}$轉換為$\boldsymbol{n}$個位元的二進位數值，使其與$\widehat{\boldsymbol{A}}$數值長度相同，得$(\boldsymbol{h_1}(\boldsymbol{SID_A}))2=\boldsymbol{m_1},\boldsymbol{m_2},\ldots\boldsymbol{m_n}$，$\boldsymbol{m_i}\in\{0,1\}$，且$\boldsymbol{i}=1,\ldots,\boldsymbol{n}$。計算授權背包值$\boldsymbol{KP_{SID_A}}$，$\boldsymbol{U_A}$擇一隨機參數$\boldsymbol{R'_{A1}}$，將$\boldsymbol{KP_{SID_A}}$與$\boldsymbol{TK_0}$進行運算，以密文$\boldsymbol{C_{A1}}$傳送至$\boldsymbol{UA}$。

$$\boldsymbol{KP_{SID_A}}=\boldsymbol{a_1m_1}+\boldsymbol{a_2m_2}+\cdots+\boldsymbol{a_nm_n} \tag{23}$$

$$\boldsymbol{KP_{SID_A}}=(\boldsymbol{KP_{SID_A}},\boldsymbol{TK_0})=(\boldsymbol{m_{11}},\boldsymbol{m_{12}}) \tag{24}$$

$$\boldsymbol{C_{A11}}=\boldsymbol{R'_{A1}}\cdot\boldsymbol{G} \tag{25}$$

$$\boldsymbol{Y_{A1}}=\mathrm{E}(\boldsymbol{y_{A11}},\boldsymbol{y_{A12}})=\boldsymbol{R'_{A1}}\cdot\boldsymbol{PK_{WS}} \tag{26}$$

$$\boldsymbol{C_{A12}}=(\boldsymbol{y_{A11}}\cdot\boldsymbol{m_{11}}\ \boldsymbol{mod}\ \boldsymbol{q},\boldsymbol{y_{A12}}\cdot\boldsymbol{m_{12}}\ \boldsymbol{mod}\ \boldsymbol{q})=(\boldsymbol{C_{11}},\boldsymbol{C_{12}}) \tag{27}$$

$$\boldsymbol{C_{A1}}=(\boldsymbol{C_{A11}},\boldsymbol{C_{A12}}) \tag{28}$$

$\boldsymbol{US}$接收到用戶端傳送之$\boldsymbol{KP_{SID_A}}$，解密得出$\boldsymbol{SID_A}$，建立關聯參數，登記作業申請之授權白名單裝置及使用者。

$$\boldsymbol{Z_1}=\boldsymbol{sk_{US}}\cdot\boldsymbol{C_{A11}}=(\boldsymbol{Z_{11}},\boldsymbol{Z_{12}}) \tag{29}$$

$$\boldsymbol{KP_{SID_A}}=(\boldsymbol{C_{11}}\cdot\boldsymbol{Z_{11}^{-1}}\mathrm{mod}\ \boldsymbol{q},\boldsymbol{C_{12}}\cdot\boldsymbol{Z_{12}^{-1}}\ \mathrm{mod}\ \boldsymbol{q})=(\boldsymbol{m_{11}},\boldsymbol{m_{12}}) \tag{30}$$

令$\boldsymbol{b}=\boldsymbol{C_p}+\boldsymbol{C_q}$，由$\boldsymbol{b}$和超遞增背包$\boldsymbol{b_1},\ \boldsymbol{b_2}\ldots,\ \boldsymbol{b_n}$，以超遞增背包解密方式即可恢復出身分$\boldsymbol{SID_A}$。$\boldsymbol{US}$賦予 USB 裝置公鑰為$\boldsymbol{PK_{uid}}$，擇一隨機參數$\boldsymbol{R'_{A2}}$，計算授權公鑰密文$\boldsymbol{C_{A2}}$並派送至 USB 裝置。

$$\boldsymbol{C_{uid}}=\boldsymbol{PK_{uid}}+\boldsymbol{K_{(US,A)}}=(\boldsymbol{m_{21}},\boldsymbol{m_{22}}) \tag{31}$$

$$\boldsymbol{C_{A21}}=\boldsymbol{R'_{A2}}\cdot\boldsymbol{G} \tag{32}$$

$$\boldsymbol{Y_{A2}}=\mathrm{E}(\boldsymbol{y_{A21}},\boldsymbol{y_{A22}})=\boldsymbol{R'_{A2}}\cdot\boldsymbol{PK_{uid}} \tag{33}$$

$$\boldsymbol{C_{A22}}=(\boldsymbol{y_{A21}}\cdot\boldsymbol{m_{21}}\boldsymbol{mod}\ \boldsymbol{q},\boldsymbol{y_{A22}}\cdot\boldsymbol{m_{22}}\boldsymbol{mod}\ \boldsymbol{q})=(\boldsymbol{C_{21}},\boldsymbol{C_{22}}) \tag{34}$$

$$\boldsymbol{C_{A2}}=(\boldsymbol{C_{A21}},\boldsymbol{C_{A22}}) \tag{35}$$

USB 裝置接收到$\boldsymbol{US}$派送之$\boldsymbol{C_{A2}}$，解密得出$\boldsymbol{C_{uid}}$。

$$\boldsymbol{Z_2}=\boldsymbol{sk_{uid}}\cdot\boldsymbol{C_{A21}}=(\boldsymbol{Z_{21}},\boldsymbol{Z_{22}}) \tag{36}$$

$$\boldsymbol{C_{uid}}=\mathrm{D}(\boldsymbol{C_{A21}}\cdot\boldsymbol{Z_{21}^{-1}}\boldsymbol{mod}\ \boldsymbol{q},\boldsymbol{C_{A22}}\cdot\boldsymbol{Z_{22}^{-1}}\boldsymbol{mod}\ \boldsymbol{q}) \tag{37}$$

$\boldsymbol{US}$計算每次授權之權限值$\boldsymbol{KP_r}$，擇一隨機參數$\boldsymbol{R'_{A3}}$及$\boldsymbol{sk_{uid}}$，以密文$\boldsymbol{C_r}$傳送予$\boldsymbol{U_A}$。

$$\boldsymbol{KP_r}=\sum_{i=1}^{n}\widehat{\boldsymbol{A}}\times\boldsymbol{r_A} \tag{38}$$

$$\boldsymbol{A_r}=(\boldsymbol{KP_r},\boldsymbol{TK'_A}) \tag{39}$$

$$\boldsymbol{PK_{A_{r1}}}=\boldsymbol{PK_{US}}+\boldsymbol{PK_{uid}} \tag{40}$$

$$\boldsymbol{PK_{A_{r2}}}=\boldsymbol{sk_{uid}}\cdot\boldsymbol{PK_{A_{r1}}} \tag{41}$$

$$\boldsymbol{C_r} = (\boldsymbol{C_{r1}}, \boldsymbol{C_{r2}}) = \mathrm{E}(\boldsymbol{R'_{A3}} \cdot \boldsymbol{PK_{A_{r1}}}, \boldsymbol{A_r} + \boldsymbol{R'_{A3}} \cdot \boldsymbol{PK_{A_{r2}}}) \quad (42)$$

四、執行階段

$\boldsymbol{U_A}$將已註冊之 USB 裝置介接於指定作業電腦，進行使用者、USB 裝置及作業電腦驗證，通過驗證則啟用 USB 裝置，步驟如次：

Step 1：USB 裝置驗證程序，將 USB 裝置介接並與$\boldsymbol{U_A}$之$\boldsymbol{K_{(A,US)}}$進行運算$\boldsymbol{PK_{uid}} = \boldsymbol{C_{uid}} - \boldsymbol{K_{(A,US)}}$，即可取得$\boldsymbol{PK_{uid}}$，$\boldsymbol{U_A}$將$\boldsymbol{C_r}$解密取得$\boldsymbol{KP_r}$及$\boldsymbol{TK'_A}$，解密成功即表示 USB 裝置為本次授權裝置。

Step 2：使用者身分驗證，解密取得$\boldsymbol{TK'_A}$與$\boldsymbol{TK_A}$進行比對，經比對$\boldsymbol{Tk'_A} = \boldsymbol{TK_A}$，確認$\boldsymbol{U_A}$為合法 PKI 用戶端，已授權合法使用。

$$\boldsymbol{A_r} = \boldsymbol{C_{r2}} - \boldsymbol{sk_{uid}} \cdot \boldsymbol{C_{r1}} = \mathrm{D}(\boldsymbol{KP_r}, \boldsymbol{TK'_A}) \quad (43)$$

Step 3：作業電腦身分驗證，解密取得$\boldsymbol{KP_r}$及$\boldsymbol{r_A}$，驗證相符確認作業電腦為本次授權裝置。

Step 4：資料同步階段，使用者作業完畢後，其電腦擇一隨機亂數$\boldsymbol{R'_{A4}}$，將 USB 裝置稽核紀錄$\boldsymbol{log}$及其雜湊值$\boldsymbol{h(log)}$加密傳送至$\boldsymbol{US}$及 USB 裝置。

$$\boldsymbol{S_{log}} = \mathrm{E}(\boldsymbol{log}, \boldsymbol{h(log)}) \quad (44)$$

$$\boldsymbol{S_{log1}} = \boldsymbol{S_{pid}} + \boldsymbol{S_{uid}} \quad (45)$$

$$\boldsymbol{S_{log2}} = \boldsymbol{sk_{pid}} \cdot \boldsymbol{S_{log1}} \quad (46)$$

$$\boldsymbol{C_{log}} = \left(R'_{A4} \cdot \boldsymbol{S_{log1}}, \boldsymbol{S_{log}} + R'_{A4} \cdot \boldsymbol{S_{log2}}\right) = \mathrm{E}(\boldsymbol{C_{log1}}, \boldsymbol{C_{log2}}) \quad (47)$$

$\boldsymbol{US}$以作業電腦私鑰$\boldsymbol{sk_{pid}}$解密得本次 USB 裝置稽核$\boldsymbol{log}$及其雜湊值$\boldsymbol{h(log)}$。

$$\boldsymbol{S_{log}} = \boldsymbol{C_{log2}} - \boldsymbol{sk_{NCid}} \cdot \boldsymbol{C_{log1}} = \mathrm{D}(\boldsymbol{log}, \boldsymbol{h(log)}) \quad (48)$$

Step 5：作業結束後歸還申領之 USB 裝置，由管理者確認 USB 裝置稽核紀錄是否與系統紀錄相符，其目的在於管理者可以同步驗證所使用之 USB 裝置(含使用者身分及作業電腦)是否都經授權，並及時發現與處理任何不相符情況，最大程度地保護系統免受未授權之存取行為。若 USB 裝置稽核紀錄與系統紀錄相符，則將 USB 裝置格式化後存管；若不相符，應有以下後續處理：警示與通知、停用裝置、隔離與調查及強化與更新系統安全等措施，以避免威脅擴大，保護系統安全。

肆、系統實作

本章節藉以實際數據，模擬資料交換作業情境，將本研究運用之橢圓曲線密碼系統及隨機背包密碼系統機制，進行各階段 USB 存取管控數據驗算，證明本研究作業流程演算法之正確性。

一、初始階段

憑證中心在有限域F_q選取橢圓曲線$E(F_q)$：$y^2 \equiv x^3 + 2x + 6 mod\ q$，選擇一個大質數$q$ =9013，在$E(F_q)$橢圓曲線上選一階數為n =8908 的基點$G = (1, 3)$，使得$8908 \cdot G = O$。選擇一個單向雜湊函數$h(\ )$及私鑰sk_{CA} =7，計算憑證中心ECC環境金鑰PK_{CA}，得$PK_{CA} = sk_{CA} \cdot G = 7(1, 3) = (8036, 5437)$。憑證中心公開$E(F_q)$：$y^2 \equiv x^3 + 2x + 6 mod\ q$、$G = (1, 3)$、$n$ =8908、$PK_{CA} = (8036, 5437)$及$h(\ )$等參數值。

二、註冊階段

本研究之身分認證機制，係由用戶端與第三方憑證中心共同參與金鑰建置，進行身分註冊作業，通訊雙方完成雙向身分驗證後即完成自我認證。

(一) 使用者註冊

業務部王小美以使用者帳號id_{A1}及個人密碼id_{A2}整合產生PKI用戶註冊所需之身分認證資料id_A =317，擇一隨機參數k_A =419，透過單向雜湊函數$h(id_A)$ =13，$h(k_A \| id_A)$ =19產生簽名檔V_A，將id_A與V_A傳送至憑證中心。

1.計算王小美簽名檔

$$\boldsymbol{V_A = h(419317)\cdot G = 19\,(1,3) = (8842, 3056)}$$

2.憑證中心註冊，憑證中心CA選取隨機參數R_A =21，計算王小美公鑰PK_{A-}及簽章SIG_A後回傳：

Step 1：計算王小美驗證公鑰

$$\boldsymbol{PK_{A-} = V_A + \left(R_A - h(id_A)\right)\cdot G = (8842,\ 3056) + 8\,(1,3) = (736, 8398) = (q_{ax}, q_{ay})}$$

Step 2：計算王小美簽章

$$\boldsymbol{SIG_A = R_A + sk_{CA}(q_{ax} + h(id_A)) = 21 + 7(736 + 13) = 5264}$$

3.使用者金鑰建置，憑證中心CA回傳驗證公鑰及簽章資料$(PK_{A-},\ SIG_A)$，計算私鑰sk_A及公鑰PK_A：

Step 1：計算私鑰$\boldsymbol{sk_A = SIG_A + h(k_A \| id_A)}$ = **(5264+ 19) =5283**

Step 2：計算公鑰$\boldsymbol{PK_A = sk_A \cdot G}$ =**5283(1, 3)** = **(2846,7958)**

(二) 隨身碟管控系統註冊

隨身碟管控系統US以管理者帳號id_{US}產生用戶註冊所需之id_{US} =307，選取隨機參數k_{US} =409，透過單向雜湊函數$h(id_{US})$ =16，$h(k_{US} \| id_{US})$ =22 產生簽名檔V_{UA}，並將id_{US}與V_{US}傳送至憑證中心。

1.計算隨身碟管控系統管理者簽名檔

$$\boldsymbol{V_{US} = h(k_{US} \| id_{US})\cdot G = h(409307)\cdot G = 22(1, 3) = (4661, 7595)}$$

2.憑證中心註冊，憑證中心CA選取一隨機參數R_{US} =32，計算本次註冊隨身碟管控系統管理者US之驗證公鑰PK_{US-}及簽章SIG_{US}後回傳，算式說明如次：

Step 1：計算管理者驗證公鑰

$$\boldsymbol{PK_{US-} = V_{US} + \left(R_{US} - h(id_{US})\right)\cdot G = (6865, 6608) = (q_{WSx}, q_{USy})}$$

Step 2：計算管理者簽章

$$\boldsymbol{SIG_{US} = R_{US} + sk_{CA}\left(q_{USx} + h(id_{US})\right) = 32 + 7(6865 + 16) = 48199}$$

3.隨身碟管控系統金鑰建置，隨身碟管控系統US藉由憑證中心CA回傳之驗證公鑰及簽章資料$(PK_{US-},\ SIG_{US})$，計算私鑰sk_{US}及公鑰PK_{US}。

Step 1：計算隨身碟管控系統私鑰

$$\boldsymbol{sk_{US} = SIG_{US} + h(k_{US} \| id_{US}) = (48199+ 22) = 48221}$$

Step 2：計算隨身碟管控系統公鑰

$$\boldsymbol{PK_{US} = sk_{US} \cdot G} = \mathbf{48221(1, 3)} = \mathbf{3681(1, 3)} = \mathbf{(137, 4686)}$$

Step 3：完成身分註冊並取得公鑰PK_n及簽章SIG_n，後續不需與憑證中心實施線上驗證，即可由憑證中心核發之id_n、PK_n與S_n等資訊進行身分認證

三、權限登記階段

完成身分註冊程序，當王小美向隨身碟管控系統提出資料交換作業申請需求時，通聯雙方進行相互驗證程序，確認雙方為合法使用者，進行後續申請作業。

(一) 用戶 PKI 驗證及使用者登記

1.驗證隨身碟管控系統PKI身分

以憑證中心回傳之驗證公鑰PK_{US-}，進行隨身碟管控系統PKI身分之合法性驗證，如驗證$PK'_{US} = PK_{US}$，代表隨身碟管控系統為合法PKI用戶，繼續登記作業，並將使用者PKI資訊傳送至隨身碟管控系統進行校驗；如驗證失敗，則終止登記作業。

Step 1：計算隨身碟管控系統公鑰PK'_{US}

$$\boldsymbol{PK'_{US} = PK_{US-} + h(id_{US}) \cdot G + (q_{USx} + h(id_{US})) \cdot PK_{CA}}$$
$$= \mathbf{(6865,6608)} + \mathbf{16(1, 3)} + \mathbf{(6865+16)} \cdot \mathbf{(8036, 5437)}$$
$$= \mathbf{(137,4686)} = \boldsymbol{PK_{US}}$$

Step 2：驗證$PK'_{US} = PK_{US}$，確認隨身碟管控系統為合法PKI用戶，繼續系統登記作業，並將使用者PKI資訊傳送至隨身碟管控系統進行身分校驗

2.驗證用戶端PKI身分

由隨身碟管控系統進行用戶端PKI身分合法性驗證，如驗證$PK'_A = PK_A$，則王小美為合法PKI用戶，即建立共享金鑰$K_{(US,A)}$並產生權杖通行碼TK_A；如驗證失敗則終止。

Step 1：計算王小美公鑰資訊PK'_A

$$\boldsymbol{PK'_A = PK_{A-} + h(id_A) \cdot G + (q_{ax} + h(id_A)) \cdot PK_{CA}}$$
$$= \mathbf{(736, 8398)} + \mathbf{13(1, 3)} + \mathbf{(736+13)(8036, 5437)} = \boldsymbol{S_A}$$

Step 2：驗證比對$PK'_A = PK_A$，確認王小美為合法PKI用戶，即完成通聯雙方身分驗證，隨身碟管控系統產生本次授權申請之通行碼TK_A並建立共享金鑰$K_{(US,A)}$。

3.建立共享金鑰

由隨身碟管控系統建立共享金鑰$K_{(US,A)}$及驗證碼TK_0，驗證碼TK_0以{0, 1}代表是否通過PKI身分驗證作業。計算共享金鑰$\boldsymbol{K_{(US,A)} = sk_{US} \times PK_A}$ $\mathbf{=48221 \times (2846,7958) = 3681(2846, 7958) = (5670, 3457)}$

4.產製權杖通行碼

經通訊雙方完成PKI身分驗證，由隨身碟管控系統產製本次作業授權申請之權杖通行碼TK_A，透由加密運算，以密文C_{A0}傳送至使用者。

Step 1：隨身碟管控系統加密程序

共享金鑰$K_{(US,A)}$將權杖通行碼TK_A =23及驗證碼TK_0 =1，選取一隨機參數R'_A =11與使用者公鑰(2846,7958)資訊加密運算為C_{A0}。

- 以共享金鑰計算點訊息P_A

  $\boldsymbol{P_A = (TK_A, TK_0) + K_{(US,A)} = (23, 1) + (5670, 3457)}$

  $\boldsymbol{= (4759, 1054) = (m_1, m_2)}$

- 計算權杖授權密文C_{A0}

  $\boldsymbol{C_{A1} = R'_A \cdot G}$ **=11 (1,3) =(1173, 865)**

  $\boldsymbol{Y_A = (y_{A1}, y_{A2}) = R'_A \cdot S_A}$ **=11(2846, 7958) =(4763, 552)**

  $\boldsymbol{C_{A2} = (y_{A1} \cdot m_1 \bmod q, y_{A2} \cdot m_2 \bmod q) = (8435, 4976) = (C_1, C_2)}$

  $\boldsymbol{C_{A0} = (C_{A1}, C_{A2})}$ **=[(1173, 865), (8435, 4976)]**

Step 2：使用者解密程序

使用者接收密文C_{A0}，即完成身分確認，建立共享金鑰$K_{(A,US)}$，解密取得權杖通行碼TK_A及驗證碼TK_0，經比對驗證碼TK_0確認雙方PKI身分已通過驗證。

- 建立共享金鑰

  使用者接收隨身碟管控系統密文C_{A0}，即已完成通訊雙方身分確認，建立共享金鑰$K_{(A,US)}$，解密取得權杖通行碼TK及驗證碼TK_0。

  $\boldsymbol{K_{(A,US)} = sk_A \times PK_{WS} = 5283 \times (137, 4686) = (5670, 3457)}$

- 使用者以使用者本身私鑰sk_A計算Z值

  $\boldsymbol{Z = sk_A \cdot C_{A1} = (Z_1, Z_2)}$ **=5283·(1173, 865)= (4763, 552)**

- 使用者以密文C_{A0}及Z值計算權杖通行碼點訊息P_A

  $\boldsymbol{P_A = (C_1 \cdot Z_1^{-1} \bmod q, C_2 \cdot Z_2^{-1} \bmod q) = (m_1, m_2)}$

  $\boldsymbol{= (8435 \cdot 4763^{-1} \bmod 9013, 4976 \cdot 552^{-1} \bmod 9013) = (4759, 1054)}$

- 將點訊息P_A與共享金鑰計算取得權杖通行碼TK_A及驗證碼TK_0

  $\boldsymbol{(p_A, p_0) = P_A - K_{(A,US)} = (4759, 1054) - (5670, 3457) = (23, 1)}$

(二) 建立存取授權模組

本階段建立隨身碟管控系統之存取授權模組，授權隨身碟可於指定作業電腦群開放使用，避免因單一主機故障無法使用，提供使用者彈性且安全的選擇，用戶端作業授權申請環境初始：

Step 1：隨機選取任一超遞增背包向量$\hat{b}$ =(5, 8, 14)。

Step 2：任意選兩個向量$\hat{u}$ =(2, 3, 5)，$\hat{v}$ =(3, 5, 8)，其中任選取兩個整數m_{A1} =11及m_{A2} =17，滿足$m_{A1} \geq \sum_{i=1}^{n} u_i$，$m_{A2} \geq \sum_{i=1}^{n} v_i$，並滿足GCD($m_{A1}$, m_{A2}) =1，運用中國餘式定理計算存取權限序列向量$\hat{A} = (a_1, \ldots, a_n)$ =(156, 69, 93)。

Step 3：依本研究情境一為例，使用者欲申請USB-1開放於指定作業電腦(PC-1)，得存取授權序列r_A =(1, 0, 0)。

(三) 資料交換作業授權登記程序

使用者以存取授權序列向量公鑰$\hat{A}$與作業授權身分SID_A計算作業授權背包值KP_{SID_A}，傳送至隨身碟管控系統進行登記，公式計算說明如次：

1.使用者作業授權申請

Step 1：計算使用者作業授權身分

計算作業授權身分$SID_A = (id_A \parallel uid_A \parallel NCid_A \parallel t \parallel t_s \parallel t_e)$，並

將作業授權身分SID_A轉換為n個位元的二進位數值，假定$(h_1(SID_A))_2 = m_1, \ldots m_n$ =101。

Step 2：計算作業授權背包值

以作業授權身分轉換之單向雜湊函數$(h_1(SID_A))_2$=101與存取權限序列向量公鑰$\hat{A}$ =(156, 69, 93)運算得作業授權背包值KP_{SID_A}。計算作業授權背包值KP_{SID_A} =156·1+69·0+93·1=249

Step 3：計算權限背包密文

使用者擇一隨機參數R'_{A1} =18，將作業授權背包值KPc_{SID_A}與驗證碼TK_0轉換為點訊息C_{SID_A}，以密文C_{A1}傳送至隨身碟管控系統。

$\boldsymbol{C_{SID_A} = (KP_{SID_A}, p_0)}$**=(249, 1)=**$\boldsymbol{(m_{11}, m_{12})}$

$\boldsymbol{C_{A11} = R'_{A1} \cdot G}$ **=18(1, 3) =(2069, 3455)**

$\boldsymbol{Y_{A1} = (y_{A11}, y_{A12}) = R'_{A1} \cdot PK_{US}}$ **=18(137, 4686) =(3727, 5327)**

$\boldsymbol{C_{A12} = (y_{A11} \cdot m_{11} \bmod q, y_{A12} \cdot m_{12} \bmod q)}$ **=(8697, 5327)=**$\boldsymbol{(C_{11}, C_{12})}$

$\boldsymbol{C_{A1}}$ **= [(2069, 3455),(8697, 5327)]**

Step 4：隨身碟管控系統解出作業授權身分

隨身碟管控系統接收到用戶端傳送之授權背包密文C_{A1}，解密得出KP_{SID_A}。

$\boldsymbol{Z_1 = sk_{US} \cdot C_{A11} = (Z_{11}, Z_{12})}$ **=48221(2069, 3455) =(3727, 5327)**

$\boldsymbol{C_{SID_A} = (C_{11} \cdot Z_{11}^{-1} \bmod q, C_{12} \cdot Z_{12}^{-1} \bmod q)}$ **=**$\boldsymbol{(m_{11}, m_{12})}$

=(249, 1)= $\boldsymbol{(c_{SID_A}, p_0)}$

Step 5：經超遞增背包運算計算後反推SID_A值，得id_A =317、uid_A =123、$NCid_A$ =24、t =2234、t_s =202012010800、t_e =202012011000，建立關聯參數，登記作業申請之裝置、使用者及起訖時間。

2.計算授權裝置公鑰資訊

管理者針對已登記之USB裝置，賦予對應公鑰PK_{uid}，並將授權公鑰資訊密文C_{A2}，派送至指定授權裝置，即完成白名單裝置授權程序，計算如次：(作業電腦授權派送程序與USB裝置相同，不在本節重複論述)

Step 1：計算授權裝置公鑰資訊並派送至指定裝置

管理者賦予USB裝置公鑰為PK_{uid}，擇一隨機參數R'_{A2} =47，假定USB裝置私鑰=26，計算密文C_{A2}，派送至指定授權USB裝置。

$\boldsymbol{PK_{uid} = sk_{uid} \cdot G}$ **=26(1, 3) =(1560, 5826)**

$\boldsymbol{C_{uid} = PK_{uid} + K_{(US,A)}}$ **=(1560, 5826)+(5670, 3457)=(1968, 4159)=**$\boldsymbol{(m_{21}, m_{22})}$

$\boldsymbol{C_{A21} = R'_{A2} \cdot G}$ **=47(1, 3) = (1174, 1687)**

$\boldsymbol{Y_{A2} = (y_{A21}, y_{A22}) = R'_{A2} \cdot PK_{uid}}$ **=47(1560, 5826) = (1303, 8930)**

$\boldsymbol{C_{A22} = (y_{A21} \cdot m_{21} \bmod q, y_{A22} \cdot m_{22} \bmod q)}$ **=(4612, 6310)=** $\boldsymbol{(C_{21}, C_{22})}$

$\boldsymbol{C_{A2}}$ **=**$\boldsymbol{(C_{A21}, C_{A22})}$**=[(1174, 1687), (4612, 6310)]**

Step 2：USB裝置取得授權公鑰資訊密文進行解密

USB裝置收到管控系統派送之資訊C_{A2}，解密得出裝置授權資訊密文C_{uid}。

$\boldsymbol{Z_2 = sk_{uid} \cdot C_{A21}}$ **=26(1174, 1686) = (1303, 8930) =** $\boldsymbol{(Z_{21}, Z_{22})}$

$\boldsymbol{C_{uid} = (C_{A21} \cdot Z_{21}^{-1} \bmod q, C_{A22} \cdot Z_{22}^{-1} \bmod q)}$ **= (1968, 4159)**

3.計算使用者作業授權背包密文

隨身碟管控系統建立各使用者對應開通USB裝置之權限表，計算每次作業授權之權限背包值KP_r，已知$r_A = (1, 0, 0)$。

Step 1：計算權限背包值$\boldsymbol{KP_r}$ **=(156×1)+(69×0)+ (93×0) =156**。

Step 2：計算使用者權限背包密文並傳送至使用者

隨身碟管控系統選取一隨機參數R'_{A3} =12及USB裝置私鑰sk_{uid} =26，將權限背包值KP_r及權杖通行碼TK'_A以密文C_r後傳送至使用者。

$\boldsymbol{A_r = (KP_r, TK'_A)}$ **=(156, 23)**

$\boldsymbol{PK_{A_{r1}} = PK_{US} + PK_{uid}}$ **= (137, 4686) + (1560, 5826) =(1034, 2823)**

$\boldsymbol{PK_{A_{r2}} = sk_{uid} \cdot PK_{A_{r1}}}$ **=26(1034, 2823) = (5730, 7045)**

$\boldsymbol{C_r = \left(R'_{A3} \cdot PK_{A_{r1}}, A_r + R'_{A3} \cdot PK_{A_{r2}}\right)}$

=(12(1034, 2823),(156, 23) +12(5730, 7045)) =(C_{r1}, C_{r2})

四、執行階段

本階段則由使用者向管理者領用已註冊之 USB 裝置，並至指定作業電腦介接進行設備檢查作業，以使用者共享金鑰$K_{(A,US)}$計算取得授權裝置公鑰(PK_{NCid}、PK_{uid})，並以 USB 裝置私鑰sk_{uid}將權限背包密文C_{KP}進行解密，解密成功則驗證 USB 裝置為已授權，取得本次作業授權申請之權限背包值KP_r及權杖通行碼TK'_A，以權杖通行碼TK'_A驗證使用者身分，以權限背包值KP_r驗證作業電腦為本次授權，並針對介接之裝置與系統登記公鑰資訊是否相符，上述驗證程序均成功則啟用 USB 裝置，驗證失敗則停用 USB 裝置。

(一) 使用者執行階段

使用者登入作業電腦並介接 USB 裝置，以共享金鑰$K_{(A,US)}$分別與 USB 裝置公鑰及作業電腦裝置公鑰點訊息(C_{uid}, C_{NCid})進行計算取得授權裝置公鑰(PK_{uid}, PK_{NCid})，並與系統登記裝置公鑰資訊比對是否相符(另作業電腦之公鑰解密程序與 USB 裝置相同)。

1.USB裝置驗證程序

Step 1：由式(31)已知USB裝置取得授權資訊密文C_{uid}，將USB裝置介接並與使用者共享金鑰$K_{(A,US)}$運算取得USB授權裝置公鑰PK_{uid}。

$\boldsymbol{PK_{uid} = C_{uid} - K_{(A,US)}}$ **=(1968, 4159)−(5670, 3457) =(1560, 5826)**

Step 2：經比對USB裝置公鑰與系統登註相符，使用者以USB裝置私鑰進行權限背包密文C_{KP}解密，解密成功即表示為本次授權裝置。

2.使用者身分驗證

完成權限背包密文C_{KP}解密並取得權限背包值KP_r及權杖通行碼TK'_A，經比對$TK'_A = TK_A$，表示使用者已通過隨身碟管控系統之PKI身分驗證程序，屬已授權合法使用者。

$\boldsymbol{A_r = C_{r2} - sk_{uid} \cdot C_{r1}}$ **=(6361, 3401)−26(284, 4095)**

=(6361, 3401)−(5969, 3619)=(6361,3401)+(5969, 3619)= (156, 23) = $\boldsymbol{(a_r, p'_A)}$

3.作業電腦身分驗證

解密取得本次作業授權之權限背包值KP_r及存取授權序列r_A，驗證相符確認作業電腦為本次授權裝置。

(二) 資料同步階段

作業電腦擇一隨機亂數R'_{A4} =13，將USB稽核紀錄log =2479及其雜湊值$h(log)$ =19，以密文C_{log}回傳系統及USB裝置。

1.建立USB裝置稽核紀錄密文

給定作業電腦私鑰sk_{NCid} =29，得$\boldsymbol{PK_{NCid} = sk_{NCid} \cdot G}$ **=29(1, 3)=(8004, 5724)**

$\boldsymbol{S_{log} = (log, h(log)) = (2479, 19)}$

$\boldsymbol{S_{log1} = PK_{NCid} + PK_{uid}}$ **= (8004, 5724) +(1560, 5826) = (646, 8342)**

$\boldsymbol{S_{log2} = sk_{NCid} \cdot S_{log1}}$ **=29(646, 8342) =(152, 2904)**

$\boldsymbol{C_{log} = (R'_{A4} \cdot S_{log1}, S_{log} + R'_{A4} \cdot S_{log2}) = (C_{log1}, C_{log2})}$

2.隨身碟管控系統解密取得USB裝置稽核紀錄

隨身碟管控系統以作業電腦之私鑰sk_{NCid} =29解密取得本次USB裝置稽核log及其雜湊值$h(log)$。

$\boldsymbol{S_{log} = C_{log2} - sk_{NCid} \cdot C_{log1}}$ **= (410, 6428) −29(4779, 8015) = (2479, 19)**

3.管理者回收裝置並比對USB裝置稽核紀錄

作業完畢，使用者歸還申領之USB裝置，由管理者取得USB裝置稽核紀錄，比對系統紀錄是否相符，若相符，則將USB裝置格式化後存管；若不相符，則將案件交由相關負責部門進行後續處理(包括警示與通知、停用裝置、隔離與調查及強化與更新系統安全等措施)。

伍、安全性分析與評估

本研究係將身分驗證機制建立於使用者、USB 裝置及作業電腦等基礎之上，並運用一次性作業授權建立複式驗證手段，有效防範未經授權之使用者、USB 裝置或作業電腦遭偽冒第三方合法存取內部資料態樣，此為線上稽核第一道防線，以提升整體資料交換作業安全強度。另有關本研究應用之安全機制，列出「安全性分析」與「效益分析」說明如後：

一、安全性分析

避免內部資料外洩並考量管理便利性，企業多數採用商用儲存媒體管控系統，以 USB 裝置識別碼建立許可使用之白名單清單，並指定作業電腦授權使用，以達存取管控效果，惟現行階段並無法有效將使用者身分及整體作業流程納入管控，但凡以系統登記之白名單裝置進行資料交換作業，相關 USB 稽核紀錄均視為合法行為。

謹依國際標準化組織(International Organization for Standardization, ISO)提出要求事項及資訊安全管理作業規範，本研究符合機密性(Confidentiality)、完整性(Integrity)及可用性(Availability)等要求外，亦具備不可否認性(Non-Repudiation)、身分認證(Authentication)及存取控制(Access Control)等特點，可有效防堵共謀攻

擊(Collusion-Attack)、重送攻擊(Replay-Attack)及偽冒攻擊(Spoofing-Attack)等網路攻擊；另參照先前章節所提及 NIST SP800-53 資訊系統安全性及隱私標準，驗證本研究符合存取控制要求項目。本研究系統運作核心係植基於隨機背包及橢圓曲線解離散對數難題之上，建構具身分驗證之 USB 存取管控機制，相關說明如次：

(一) 機密性：

本研究方法提及之授權資訊派送，係結合隨機背包密碼系統運算，將授權資訊以橢圓曲線點加法運算並結合橢圓曲線加密後派送至指定裝置，如式(31)；另將權限背包值$\boldsymbol{a_r}$及權限驗證碼$\boldsymbol{p'_A}$以 USB 裝置公鑰執行橢圓曲線加密運算派送至使用者，當使用者介接指定 USB 裝置並以其私鑰方可解密成功，如式(39)至(43)式。破譯者若想解開裝置授權資訊及權限背包值，將面對橢圓曲線離散對數難題及隨機背包問題，即便取得存取權限背包值$\boldsymbol{a_r}$，已知由背包值反推導出已登記授權作業電腦及存取授權序列向量$\widehat{\boldsymbol{A}}$，此類問題已被證明是一個 NP-Complete 問題，無法在多項式運算時間內解出，故符合機密性之要求。

(二) 可用性：

VMware vSphere 為現今廣泛使用之虛擬化平台，旨在幫助企業管理和優化其資源運用，提高 IT 基礎設施之效能性、可用性與安全性(VMware 2022)，架構圖如圖 3 所示。本研究系統架構預設搭配 VMware vSphere 虛擬化平台使用，將認證中心與隨身碟管控系統建置於虛擬主機群，除可藉 vCenter Server 統一管理主機群及調配運算資源外，其亦提供故障容忍(Fault Tolerance)、虛擬機轉移(vMotion)及備份與復原等備援機制，允許虛擬主機在硬體故障、系統維護甚至於遭受外部網路攻擊期間實現無感知移動與故障轉移，並迅速於災後恢復系統正常運作，故符合可用性之要求。

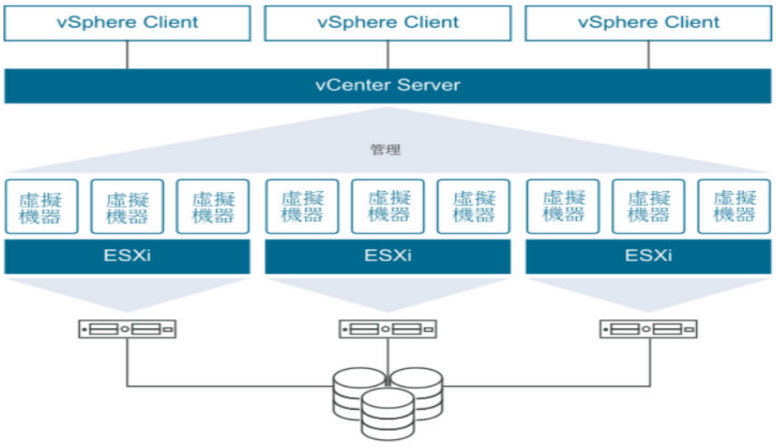

圖 3：虛擬化平台架構示意圖

(三) 完整性：

本研究方法在 PKI 用戶端註冊階段，先以使用者帳號$\boldsymbol{id_{A1}}$及個人密碼$\boldsymbol{id_{A2}}$整合為註冊身分資料$\boldsymbol{id_A}$，隨即將$\boldsymbol{id_A}$及註冊之簽名檔$\boldsymbol{V_A}$送至憑證中心，如式(2)；另憑證中心完成使用者身分雜湊運算後，回傳使用者驗證公鑰$\boldsymbol{PK_A}$及簽章$\boldsymbol{w_A}$，如

式(3)及式(4)。破譯者如欲偽冒或竄改使用者 PKI 身分，則須面對破解單向雜湊函數及橢圓曲線離散對數難題;即便破解使用者 PKI 身分與共享金鑰$\boldsymbol{K_{(US,A)}}$取得權杖通行碼$\boldsymbol{p_A}$，仍須面對隨機背包難題及橢圓曲線離散對數難題，否則傳遞至隨身碟管控系統進行白名單註冊之權限背包值$\boldsymbol{a_r}$是無法被偽冒或是變更，故符合完整性之要求。

(四) 不可否認性：

使用者與隨身碟管控系統完成雙向 PKI 身分驗證後，即建立共享金鑰$\boldsymbol{K_{(US,A)}}$及$\boldsymbol{K_{(A,US)}}$，如式(11)及式(17)，並產制權杖通行碼$\boldsymbol{p_A}$與驗證碼$\boldsymbol{p_0}$，將權杖通行碼點訊息$\boldsymbol{P_A}$以橢圓曲線點加法與進行運算，以密文$\boldsymbol{C_{A0}}$傳送至使用者，如式(12)。作業授權申請之身分$\boldsymbol{SID_A}$，經單向雜湊運算後與存取授權序列公鑰$\boldsymbol{\widehat{A}}$運算得作業授權背包值$\boldsymbol{c_{SID_A}}$，如式(22)及式(23)；當隨身碟管控系統解出$\boldsymbol{SID_A}$並完成註冊，透由共享金鑰$\boldsymbol{K_{(US,A)}}$與裝置授權資訊運算，以密文$\boldsymbol{C_{A2}}$派送至指定裝置，如式(31)。另在本研究之共享金鑰組成包括雙方私鑰及公鑰資訊，其中私鑰建置已涵蓋使用者個人身分資訊，如式(5)，使用者無法否認提出作業授權申請或已執行之行為。

(五) 身分認證：

當使用者向隨身碟管控系統提出資料交換申請需求時，通聯雙方向彼此傳送 PKI 驗證訊息，經比對隨身碟管控系統($\boldsymbol{id_{US}}$、$\boldsymbol{S_{US}}$、$\boldsymbol{PK_{US}}$)及使用者($\boldsymbol{id_A}$、$\boldsymbol{S_A}$、$\boldsymbol{PK_A}$)公開資訊，確認$\boldsymbol{S'_{US} = S_{US}}$且$\boldsymbol{S'_A = S_A}$，確認雙方均為合法 PKI 使用者，則進行後續權限登記作業，如式(8)及式(10)。破譯者須面對破解單向雜湊函數及橢圓曲線解離散對數難題，方可成功破解通訊雙方之身分驗證程序，若無法破解式(8)及式(10)，則符合身分認證之要求。

另本研究採用自我認證機制，係以用戶端身分資料建立驗證公鑰，如式(3)；當通訊雙方欲連線進行資料傳輸前，藉由彼此身分資料及驗證公鑰進行身分驗證，確認是否為合法 PKI 使用者，如式(2)至式(7)，而無須透由憑證中心進行身分驗證，可有效減輕伺服器連線管理負荷。

(六) 存取控制：

存取控制(Access Control)係指用來判斷使用者是否符合其存取權限，以避免系統遭非合乎設定權限之存取行為。本研究中權杖通行碼(即存取權限 Scope)，均由隨身碟管控系統依使用者個別權限製成，經基於橢圓曲線之共享金鑰加密後派送至 USB 裝置及作業電腦，單次作業授權則以作業權限背包值加密後派送至使用者，故攻擊者需面對離散對數及 NP-Complete 難題。

本研究隨身碟管控系統派送至 USB 裝置之授權資訊**密文**$\boldsymbol{C_{uid}}$，如式(31)，僅授權 USB 裝置私鑰可解密成功並取得權限背包值$\boldsymbol{a_r}$及權杖通行碼$\boldsymbol{p'_A}$，經比對$\boldsymbol{p'_A = p_A}$，則確認使用者及 USB 裝置為本次授權使用，亦符合存取管控要求。另本研究使用之存取權限授權序列$\boldsymbol{r_A}$，係以隨機背包密碼系統求得權限背包值$\boldsymbol{a_r}$，如式(21)、式(38)，此方式可提供使用者在指定之作業電腦群進行授權使用，增加使用者進行資料交換作業之彈性，避免單一設備故障影響該次資料交換作業。

(七) 抗共謀攻擊：

共謀攻擊係指由兩個或兩個以上的人、單位、組織等，為達到一個非法或詐欺的目的所進行的秘密協議或合作（林建銘 2019），即非授權之存取行為。例如在本研究預設單一網路環境(同一網域且實體隔離網路)，憑證中心恐與不法使用者共謀，以獲取與隨身碟管控系統間之共享金鑰，進而竄改權杖通行碼，越權存取。

在本研究方法中，使用者以個人身分簽名檔傳送至憑證中心註冊，如式(2)，由於使用者共同參與金鑰建置，故認證中心無法偽造使用者公鑰，且使用者會自行驗算認證中心所傳過來的公鑰之正確性，故認證中心無法主導使用者金鑰產生及驗證，可有效避免內部管理人員監守自盜情形。當通訊雙方確認彼此 PKI 身分後建立共享金鑰，如式(11)及如式(17)，並由隨身碟管控系統產生權杖通行碼$\boldsymbol{p_A}$及驗證碼$\boldsymbol{p_0}$，以共享金鑰運算並加密傳送至使用者，如式(12)，因憑證中心僅參與金鑰建置，無法取得隨身碟管控系統與使用者建立之共享金鑰以取得存取控制權，故可防範共謀攻擊。

(八) 抗重送攻擊：

重送攻擊是指攻擊者蒐集合法使用者所傳送的過期資訊，並將上述資訊在最短時間內重送，企圖通過認證，以取得使用者的私密資訊或是有用之回應(蘇品長、葉家維、黃啟清 2019)。例如不法使用者以過期資訊，重新向隨身碟管控系統發送驗證，以獲得權杖通行碼。

使用者之作業授權身分$\boldsymbol{SID_A}$涉及使用者身分資料$\boldsymbol{id_A}$、USB 裝置、作業電腦資訊、亂數時戳及授權起訖時間，如式(22)，並以隨機背包密碼系統運算產生作業授權背包值$\boldsymbol{KP_{SID_A}}$，加密傳送至隨身碟管控系統進行註冊，如式(23)；且使用者每次資料交換作業申請均由隨身碟管控系統產製權杖通行碼$\boldsymbol{TK_A}$，作為使用者身分及存取權限之驗證，如式(12)及式(43)，故攻擊者無法藉由蒐集過往傳送之資訊進行重送攻擊。

(九) 抗偽冒攻擊：

偽冒攻擊是指攻擊者偽冒成合法使用者的身分，為群組中其它成員提供非法的服務，以進行任何可能的破壞行為或提供簽署錯誤的憑證(蘇品長、葉家維、黃啟清 2019)。例如不法使用者竊取已通過憑證中心註冊之合法使用者身分資料(即帳號、密碼)，以進行違規存取。

使用者將個人資料經單向雜湊運算產生之簽名檔，傳送至憑證中心進行身分註冊作業，如式(2)；憑證中心完成用戶端註冊資料處理後回傳使用者簽章，如式(4)；使用者藉由憑證中心回傳之驗證公鑰及簽章資料($\boldsymbol{PK_A}$, $\boldsymbol{w_A}$)，計算本身金鑰資訊，如式(5)；接續由使用者驗證憑證中心回傳驗證公鑰之正確性，如式(7)。鑑此，憑證中心未能掌握用戶端金鑰產生及驗證資訊，且欲偽冒用戶端身分之行為會被偵測發現，況且在通訊雙方驗證彼此 PKI 身分時，係以公開資訊及使用者身分資料驗證其合法性，如式(7)，綜上所述，可有效防堵偽冒攻擊。

(十) NIST SP800-53 存取控制標準：

本小節說明本研究如何滿足 NIST SP800-53 資訊系統安全性及隱私標準中之存取控制要求項目，以確保資訊系統的存取安全受到適當保護，分述說明如次：

AC-1：政策與流程：本研究係運用橢圓曲線密碼系統之離散對數難題為基礎，結合單向雜湊演算法及隨機背包密碼系統，建立基於身分驗證之USB存取管控機制，結合自我認證機制，對白名單USB裝置、作業電腦及指定人員帳號進行一次性授權認證，避免偽冒行為，進而提升系統安全性。

AC-2：帳戶管理：本研究中權杖通行碼(即存取權限Scope)，均由隨身碟管控系統依使用者個別權限製成(若未經授權，則不派送權杖通行碼)，經基於橢圓曲線之共享金鑰加密後派送至USB裝置及作業電腦，單次作業授權則以作業權限背包值加密後派送至使用者，故攻擊者需面對離散對數難題及NP-Complete難題。

AC-3：監控執行：本研究由隨身碟管控系統負責使用者活動監控，除僅允許符合使用者權限之存取行為外，於使用完畢後，回收USB裝置並比對其稽核紀錄，若有不相符情事，則將案件交由相關負責部門進行後續處理(包括警示與通知、停用裝置、隔離與調查及強化與更新系統安全等措施)。

AC-5：權責劃分：本研究區分憑證中心、隨身碟管控系統、使用者、USB裝置及作業電腦，各個角色具備適當權責且相互制衡，避免產生共謀攻擊情事。

AC-7：未經授權的系統使用：本研究於「權制登記階段-資料交換作業授權登記程序」，亦會使用作業授權時間之亂數時戳$\boldsymbol{t}$及起訖時間$(\boldsymbol{t_s}, \boldsymbol{t_e})$，即限制本次存取行為之使用時間。

二、效益評估

表 4：本研究與現行作業比較表

比較項目	現行商用儲存媒體管控系統	本研究系統
機密性	部分符合	符合
完整性	無	符合
不可否認性	部分符合	符合
身分認證	部分符合	符合
存取控制	部分符合	符合
抗共謀攻擊	無	符合
抗重送攻擊	無	符合
抗偽冒攻擊	無	符合

本研究係基於實體隔離環境下，整合隨機背包密碼系統及橢圓曲線密碼系統解離散對數難題之特性及自我認證機制之應用，結合當次資料交換作業授權、使用者身分、作業電腦群及 USB 裝置之複式驗證程序，建立具備身分驗證能力之 USB 存取管控方法，深化使用者單次資料交換作業安全及線上稽核管控力度，並透由加密及雜湊運算建立 USB 裝置稽核紀錄，完善整體安全管控強度，可達到機密性、完整性、不可否認性等安全需求，亦具備離線身分驗證、彈性存取控制

機制，同時具備抵抗共謀攻擊、重送攻擊及偽冒攻擊能力，綜整本研究與現行作業流程比較及說明如表 4、表 5。

表 5：本研究與現行作業安全性比較說明表

比較項目	現行商用儲存媒體管控系統	本研究系統
機密性	無法針對未經授權之使用者限制其存取管控行為；亦無法限制偽冒白名單管控之USB裝置進行資料存取。	採用隨機背包密碼系統建立使用者單次作業授權身分及申請裝置參數，以破解隨機背包之難題確保資料安全。
完整性	無	採用以使用者帳號及個人密碼整合為註冊身分資料並進行單向雜湊運算及註冊，第三方如進行竄改或藉以偽冒使用者簽章而不被發現，則須面對破解單向雜湊函數及橢圓曲線離散對數難題。
不可否認性	僅針對作業電腦及 USB 裝置進行白名單裝置清單比對，無結合使用者相關申請資訊，如作業當下前一使用者帳號未登出，系統亦無法限制其使用。	共享金鑰組成包括雙方公、私鑰，而私鑰係以使用者資料參與金鑰建置，已涵蓋個人資訊，使用者無法否認提出之作業授權申請或已執行之行為。
身分認證	僅針對作業電腦及 USB 裝置進行主從式架構 (Client-Server Model)之網路認證。	第三方須面對破解單向雜湊函數及橢圓曲線解離散對數難題，始能成功破解通訊雙方之身分驗證程序。另以自我認證機制取代憑證中心線上認證機制，減少認證伺服主機管理 負荷及憑證儲存管理問題。
存取控制	僅授權白名單 USB 裝置於指定作業電腦進行資料存取。	運用隨機背包密碼系統，建置一維存取管控方案，可依實際需求調整 USB 裝置授權於指定作業電腦群，避免申請單一作業電腦臨時變故無法使用。
抗共謀攻擊	無	使用者共同參與金鑰建置，認證中心無法主導使用者金鑰產生及驗證，可避免管理者監守自盜。
抗重送攻擊	無	使用者作業授權申請，以當下隨機亂數時戳、授權起訖時間等資料，建構唯一作業授權身分SID_A，可有效避免重送攻擊。
抗偽冒攻擊	無	通訊雙方共同參與金鑰建置，以公開資訊及使用者身分資料驗證其合法性，憑證中心未能掌握用戶端金鑰及驗證資訊；執行階段另針對使用者、USB 裝置及作業電腦進行身分及存取管控檢查。

考量本研究屬客製化設計系統之演算架構，囿於既有執行流程無相關參照演算法，故無既有系統進行比較；學者 Ayub et al. (2020)針對 USB 儲存裝置所提出之雙因子認證方案，系統架構限制用戶端在存取 USB 設備中的數據之前，必須

登錄憑證，經過服務器端認證後，始能傳輸資料及使用已儲存之相關數據，參考其研究所提及，執行 1 次 hash(SHA-1)所需時間為 3.5 微秒，即 0.0000035 秒。本研究以上述研究數據做為時間複雜度運算參考，運算符號及運算時之相互關係見表 6(蘇品長、葉昱宗 2017)，並依此參考表完成本研究各階段時間複雜度運算，計算簡表說明見表 7。

表 6：運算成本(Computation cost)參考表

符號	定義
T_{MUL}	進行一次模式乘法運算所需時間
T_{ECMUL}	進行一次ECC乘法運算所需時間≈ 29 T_{MUL}
T_{ECADD}	進行一次ECC加法運算所需時間≈ 5 T_{MUL}
T_{INVS}	進行一次模式乘法反元素運算所需時間≈ 240 T_{MUL}
T_{EXP}	進行一次模式指數運算所需時間≈ 240 T_{MUL}
T_{ADD}	進行一次模式加法運算所需時間(可忽略不計)
t_h	進行一次$hash$(SHA-1)所需時間≈ 0.4 T_{MUL}
備註	1.$T_{\sum MUL}$：若該序列長度為n，則以nT_{MUL}表示。 2.$T_{\sum ADD}$：若該序列長度為n，則以$(n-1)T_{MUL}$表示。

表 7：本研究時間複雜度概估計算簡表

階段	成本預估	小計
系統初始	$1T_{ECMUL}$	29 T_{MUL}
PKI 註冊	$2t_h + 5T_{ECMUL} + 4T_{ADD} + 3T_{ECADD} + 1T_{MUL}$	161.8 T_{MUL}
權限登記	＊PKI 驗證與使用者登記 $4t_h + 1T_{ADD} + 8T_{ECMUL} + 6T_{ECADD} + 4T_{MUL} + 2T_{INVS}$	747.6 T_{MUL}
	＊資料交換作業授權登記 $1t_h$ +$10T_{ECMUL}$ +$3T_{ECADD}$ +$(2n+8)T_{MUL}$ +$4T_{INVS}$ + $(2n-2)T_{ADD}$	$2n$+1273.4 T_{MUL}
執行階段	$1t_h$ +$4T_{ECADD}$ +$5T_{ECMUL}$	165.4 T_{MUL}
小計	$8t_h + 24T_{ECMUL}$ +$16T_{ECADD}$ +$(2n+13)T_{MUL}$ +$6T_{INVS}$ + $(2n+3)T_{ADD}$	$(2377.2+2n)$ T_{MUL}
結論	1.本研究時間複雜度運算係參照學者 Ayub et al. (2020)，以執行 1 次 hash(SHA-1)所需時間為 3.5 微秒進行運算，可得： $(2377.2+2n)T_{MUL}$ ≈ 20818 微秒 ≈ 0.021 秒 ＜ 0.1 秒 2.本研究運算成本為 0.021 秒，學者 Ayub et al. (2020)其研究運算成本為 261.94 微秒，即 0.00026194 秒，然該研究運算成本僅包含初始、使用者註冊(無隨身碟管控系統註冊)及認證與資料加密等 3 階段，在系統架構不對等情況下，因此作為參考之用。	

綜上可知，本研究方法設計之系統演算架構，整合現行資料交換作業流程及儲存媒體管控，依作業申請、身分驗證、授權派送、實際作業及資料同步等流程進行運算達$(2377.2+2\boldsymbol{n})\boldsymbol{T_{MUL}}$，參照學者 Ayub et al. (2020)，以執行 1 次 hash(SHA-1)所需時間為 3.5 微秒進行運算，可在 0.1 秒內完成全系統運算，其中權限登記

階段計算量達$(2021+2n)T_{MUL}$，系統運算量比例最大，但實際上對於使用者而言係無需等待，屬系統後端執行程序；另於使用者執行階段之效能估算為$165.4T_{MUL} \approx 0.0014$秒，可在不影響使用者執行效率下完成使用者、USB 裝置及作業電腦之單次授權認證，有效強化資料安全管控，降低人為誤用風險。本研究運用橢圓曲線密碼系統、改良式隨機背包密碼系統及自我認證機制環境下，具備離線認證、資料加解密、單次作業授權及複試檢查機制，這些優點使本研究方法更適用於企業或公家機關實際資料交換作業情境，強化線上稽核管控力度。

陸、結論

本研究係針對單一網路環境隨身碟管控系統之授權存取應用，結合 USB 裝置、作業電腦、使用者身分及線上審查機制，建立作業單次授權許可，未來仍可研究如何精進整體 USB 裝置使用之資安管控措施，以精進整體防護安全強度，具體貢獻如次：

一、利用橢圓曲線長度短且安全性高之特性，搭配隨機背包密碼系統增加其複雜度，使系統達到更安全的運作。

二、建立使用者、USB 裝置及作業電腦唯一身分識別碼，結合隨身碟管控系統單次作業授權，強化端點設備使用之不可否認性，可有效避免白名單裝置遭偽冒運用，可降低非法存取風險，精進資料交換安全。

三、管理端採用自我認證機制，即作業端離線狀態亦可進行身分認證，認證效率高，可有效降低認證伺服主機之管理負荷及憑證存管問題，避免憑證中心或使用者遭偽冒而衍生安全疑慮。

另本研究主要聚焦於探討 USB 裝置存取管控線上稽核之可行性，故並未將現行白名單機制納入研究架構中，及不需配合額外白名單使用；然而，白名單亦屬於實體隔離環境中，有效阻絕未經授權裝置存取之措施，在不考慮系統負荷下，整合該機制，可提供更全面性的保護力。為落實儲存媒體管理及資料輸出入管控，現階段多數公務機關以實體隔離方式限制於單一環境資料交換區集中使用，未來期能結合單向光纖機制進行跨網域管理，並結合彈性存取管控模組，建構多維度 USB 裝置存取管控系統，期能在日趨便利之 USB 裝置使用前提下，建構更具安全之存取稽核管控環境。

參考文獻

iThome-2019iT 邦幫忙鐵人賽 (2018)，「總統府資安週，隨身碟贈品竟藏病毒」，https://ithelp.ithome.com.tw/articles/10204726.

iThome-News (2018)，「深度剖析台積產線中毒大當機始末」，https://www.ithome.com.tw/news/125098.

TechNews 科技新報 (2022)，「Raspberry Robin 高風險蠕蟲再起！數百家微軟科技業及製造業客戶網路遭入侵」，https://technews.tw/2022/07/07/microsoft-finds-raspberry-robin-worm-in-hundreds-of-windows-networks/.

王青龍、趙祥模 (2015)，「隨機背包公鑰密碼的分析與改進」，*計算機科學*，第 42 卷，第 6 期，頁 158-161。

王保倉、韋永壯、朝予濮 (2010)，「基于隨機背包的公鑰密碼」，*電子與信息學報*，第 32 卷，第 7 期，頁 1580-1584。

台灣電腦網路危機處理暨協調中心 (2023)，「新版 PlugX 惡意軟體，會藏於 USB 裝置內感染 Windows 系統」，https://www.twcert.org.tw/tw/cp-104-6897-c8d06-1.html.

林建銘 (2019)，「植基於任務導向存取控制之空軍情資整合系統設計」，未出版碩士論文，國防大學管理學院資訊管理學系研究所，台北市。

侯皓薰 (2018)，「BadUSB 攻擊分析與預防」，未出版碩士論文，中國文化大學商學院資訊管理學系研究所，台北市。

傅振華、張凱棠 (2022)，「虛擬化環境告警推播機制建置實作-以即時通訊軟體 LINE 為例」，*中正嶺學報*，第 51 卷，第 2 期，頁 1-14。

費向東、丁燕艷、潘郁 (2012)，「重新認識背包公鑰密碼的安全性」，*計算機應用*，第 32 卷，第 3 期，頁 694-698。

黃建勛、蕭舜文(2022)，「以區塊鏈技術、流程安全與選民隱私設計之去中心化投票架構」，*資訊管理學報*，第 29 卷，第 3 期，頁 133-159。

資安人科技網 (2008)，「國防部發現變種 USB 病毒，對岸網軍針對性攻擊」，https://www.informationsecurity.com.tw/article/article_detail.aspx?aid=4659.

資安人科技網 (2021)，「實體隔離是資安萬靈丹!?」，https://www.informationsecurity.com.tw/article/article_detail.aspx?aid=5313.

蘇品長、夏君和、蘇泰昌 (2022)，「建構具安全性的智慧合約共享方案-以房屋共享為例」，*資訊管理學報*，第 29 卷，第 3 期，頁 253-275。

蘇品長、陳明心 (2018)，「具交易匿名性之電子商務協定設計─以第三方支付為例」，*電子商務學報*，第 20 卷，第 2 期，頁 165-188。

蘇品長、葉昱宗 (2017)，「新型態之電子投票機制」，*電子商務學報*，第 19 卷，第 1 期，頁 29-50。

蘇品長、葉家維、黃啟清 (2019)，「設計輕量化即可離線認證之訊息保護機制」，*國防管理學報*，第 40 卷，第 2 期，頁 19-37。

Ayub, M. F., Shamshad, S., Mahmood, K. , Islam, S. H., Parizi, R. M., & Choo, K. R. (2020). A Provably Secure Two-Factor Authentication Scheme for USB Storage Devices. *Transactions on Consumer Electronics*, 66(4), 396-405.

Baumann, A. (2017). *Hardware is the new software. in Proc. HotOS'17*, Whistler, Canada, 132-137.

Bhakte, R., Zavarsky, P., & Butakov, S. (2016). Security Controls for Monitored Use of USB Devices Based on the NIST Risk Management Framework. *IEEE 40th Annual Computer Software and Applications Conference*, 461-466.

Felser, M., Rentschler, M., & Kleineberg, O. (2019). Coexistence standardization of operation technology and information technology. *Proceedings of the IEEE*, 107(6), 962-976.

Girault, M. (1991). Self-certified public keys. *Paper presented at Workshop on the Theory and Application of Cryptographic Techniques*, 490-497.

Haq, I. U., Wang, J., & Zhu, Y. (2019). An Efficient Authenticated Key Agreement Scheme for Consumer USB MSDs Resilient to Unauthorized File Decryption. *IEEE Transactions on Consumer Electronics*, 65(1), 80-89.

Kang, M. & Saiedian, H. (2017). USBWall: A novel security mechanism to protect against maliciously reprogrammed USB devices. *Information Security Journal: A Global Perspective*, 26(4), 166-185.

Kim, K., Kim, T., Warraich, E., Lee, B., Butler, K. R. B., Bianchi, A., Tian, D. (2022). FuzzUSB: Hybrid Stateful Fuzzing of USB Gadget Stacks. *2022 IEEE Symposium on Security and Privacy (SP)*, 2212-2229.

Kurii, Y. & Opirskyy, I. (2022). Analysis and Comparison of the NIST SP 800-53 and ISO/IEC 27001:2013. *CEUR Workshop Proceedings*, 3288(3), 21-32.

Mamchenko, M. & Sabanov, A. (2019). Exploring the Taxonomy of USB-Based Attacks. *2019 Twelfth International Conference Management of large-scale system development (MLSD)*, 1-4.

Meijer, C. & Gastel, B. V. (2019). Self-Encrypting Deception: Weaknesses in the Encryption of Solid State Drives. *2019 IEEE Symposium on Security and Privacy (SP)*, 72-87.

Merkle, R. & Hellman, M. (1978). Hiding information and signatures in trapdoor knapsacks. *1978 IEEE Transactions on Information Theory*, 24(5), 525-530.

Mohammadmoradi, H. & Gnawali, O. (2018). Making whitelisting-based defense work against badUSB. *In Proceedings of the 2nd International Conference on Smart Digital Environment*, 127-134.

Force, J. T. (2020). Security and privacy controls for information systems and organizations (No. NIST Special Publication (SP) 800-53 Rev. 5 (Final Public Draft)). National Institute of Standards and Technology.

Paes, R., Mazur, D. C., Venne, B. K., & Ostrzenski, J. (2019). A guide to securing industrial control networks: Integrating IT and OT systems. *IEEE Industry Applications Magazine*, 26(2), 47-53.

Tian, D. J., Scaife, N., Kumar, D., Bailey, M., Bates, A., & Butler, K. R. (2018). SoK: "Plug & Pray" Today - Understanding USB Insecurity in Versions 1 Through C. *2018 IEEE Symposium on Security and Privacy (SP)*, 1032-1046.

Tischer, M., Durumeric, Foster, Z. S., Duan, S., Mori, A., Bursztein, E., & Bailey, M. (2016). Users Really Do Plug in USB Drives They Find. *2016 IEEE Symposium on Security and Privacy*, 306-319.

Verma, S. & Singh, A. (2012). Data theft prevention & endpoint protection from unauthorized USB devices-Implementation. *2012 IEEE Fourth International Conference on Advanced Computing (ICoAC)*, 1-4.

VMware. (2022). VMware vSphere description document,https://docs.vmware.com/tw/VMware-vSphere/index.html

SecurityWeek. (2023). Russian Hackers Using USB-Spreading Malware in Attacks on Ukrainian Government, Military, https://www.securityweek.com/russian-hackers-using-usb-spreading-malware-in-attacks-on-ukrainian-government-military/

Smartermsp. (2023). Beware! USB drop attacks persist- FBI warns against attacks, https://smartermsp.com/beware-of-the-usb-usb-drop-attack-threats-persist/