# 以 P2P 網路架構實作 Single Sign-On 機制之研究

廖鴻圖
世新大學資訊管理學系
htliaw@cc.shu.edu.tw

林金龍
中央研究院資訊所
世新大學資訊管理學系
eddy@iis.sinica.edu.tw

## 摘要

本篇論文主要的目的是如何在現成的 Web 環境中，以不改變原有的系統架構下，並透過 Server 與 Server 之間的信任關係，建立一個 Web 單一登入( Web Single Sign-On )之功能。

本研究將針對 Single Sign-On 服務所面臨的問題，提出一個可解決這些問題的服務架構。在這個服務架構中，將透過 Server 與 Server 之間的信任關係，交替驗證使用者的登入服務，並透過 Server 本身的權限管理機制，管理由信任伺服器所發出憑證的使用者，以提供良好的權限控管機制。

最後，本研究將根據此一架構，並於數位典藏系統網站中實作一套範例網站來驗證這個服務架構的可行性。

## 關鍵詞

Peer to Peer、P2P、Single Sign-On、SSO、Web Service

## 1. 前言

Single Sign-On(SSO) 一向是電子商務與網路服務的重要課題。使用者認證、授權管理的複雜性，也隨著網站服務市場不斷成長，而成為網站服務系統管理者的重大負擔。由於缺乏一個普遍的資訊交換技術，各個網站服務間的使用者認證資訊無法相互交換，致使這些服務的使用者必須一再地輸入認證資訊。這不僅增加使用者的困擾，也讓網站服務系統安全蒙上一層陰影。單一登入服務 Single Sign-On 技術的發展目標，是為了解決使用者一再輸入認證資料的問題。

而現有的 Single Sign-On 服務大多數皆為單一認證模式，即由一台認證伺服器來管理使用者資料庫，並對使用者做認證及發放憑證的動作，這樣一但同時上線的使用者人數過多，必定會造成認證伺服器的效能受到影響。且大多數的 Single Sign-On 服務皆不具有權限控管機制，採用這些 Single Sign-On 服務的系統，就必須另外找尋權限控管機制來配合。而單一認證模式也比較會有安全性的問題，一但認證伺服器遭駭客入侵或使用者帳號密碼被破解時，其所有相關的 Web 服務皆無法受到權限控管的保護了。

且現有的 Web 系統，皆已建制完成，並有獨立的認證機制及權限控管機制，如何在不影響現有的架構下做到 Single Sign-On。在本研究中提出一個在信任模式下的多伺服器交互認證的方法，以解決這些已有現成認證及權限控制服務的 Web 伺服器，達到以最低成本完成 Web Based Single Sign-On 的目的。

## 2. 文獻探討

本節將對 Web Based Single Sign-On 簡介，並對目前熱門的 SSO 產品 Microsoft .NET Passport 介紹並說明以上各機制中所包含的機制流程。並對本研相關的技術 Peer to Peer、Web Server 做簡單介紹。

### 2.1 Web Based Single Sign-On 簡介

當使用者在一個提供多個服務或系統的環境之中，為了存取這些服務或系統，使用者就必須在切換的同時又要經過一道身份的認證(通常為 username/password)，這樣將造成使用上的不便。所以 SSO 的概念便興起，使用者通過一次身份驗證後就可以自由存取網路上其它不同主機上之系統服務而不必再重複身份驗證步驟，如此便可以帶來以下的效益：

(1) 減少多次身份認證所花的時間，相對的也減少身份認證出錯的可能性。

(2) 避免使用者同時保有多個認證資訊而造成混亂。

(3) 管理者可以集中管理使用者擁有的存取權並減少維護的時間。

常見的網站單一登入系統實作的架構有以下兩種模式：代理人模式(Proxy Model) (圖 1)及重導模式(Inform Model) (圖 2)。

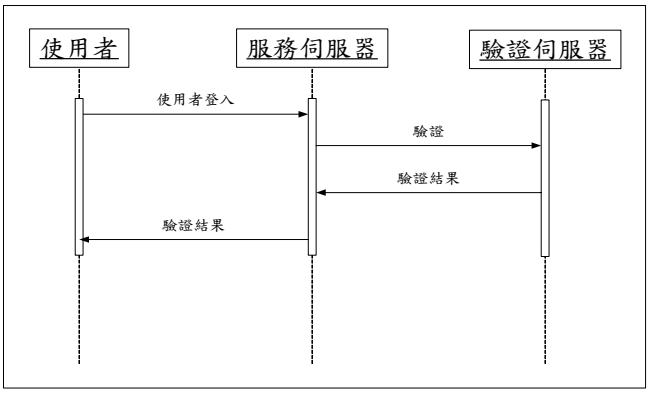


圖 1.單一登入系統之代理人模式[1]

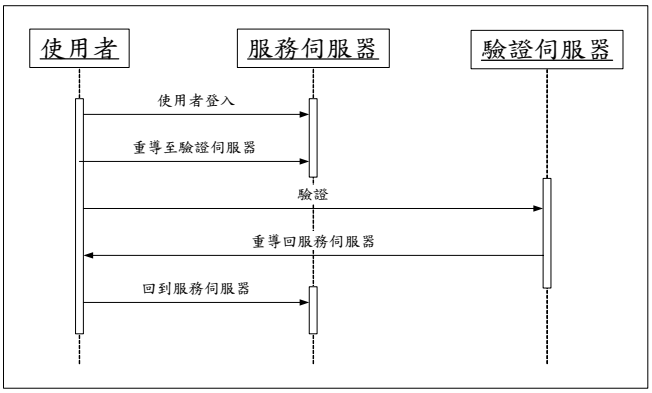


圖 2 單一登入系統之重導模式[1]

## 2.2 Microsoft .NET Passport

Microsoft .NET Passport 為軟體大廠 Microsoft 於 1999 年所提出之網站單一登入系統機制，為目前全球最為廣泛使用之網站單一登入系統，採用麻省理工學院研發之網路安全標準 Kerberos 為其安全機制，其設計的目的是要減少使用者必須記得其在不同網站的使用者識別碼及相關密碼，根據 Microsoft 官方統計，至 2002 年 1 月為止，已有 2 億名使用者登記使用，目前提供 27 種語言版本，每個月登入次數達 35 億人次。

目前.NET Passport 提供之功能包括：

單一登入服務 (Single Sign-On)：

單一登入為 .NET Passport 最主要之功能，讓使用者只需記得一組帳號及密碼即可通行所有 .NET Passport 之合作網站，並且能存取以 .NET Passport 為基礎的 Web Service。

電子錢包 (Express Purchase)：

使用者可以選擇在 .NET Passport 的電子錢包中儲存信用卡資料與收貨帳單地址等相關個人資訊，以方便線上交易時可自動由電子錢包讀取所需之資料，簡化線上購物流程及時間。

兒童護照 (Kids Passport)：

提供『兒童線上隱私保護法 (Children's Online Privacy Protection Act; COPPA)』之功能，該項法令要求網站必須取得父母的同意，才能允許收集，使用或顯示兒童的個人資訊。

Microsoft .NET Passport 是採用公證第三者驗證方式 (Trust Third Party Authentication) 對使用者進行驗證，待驗證成功後隨即簽發一組會議金鑰，服務網站於接收到該會議金鑰後自行以和 .NET Passport 共同持有之私密金鑰作對稱式加解密及驗證，於安全性上較為薄弱。

Microsoft .NET Passport 使用者認證流程如圖 3 所示：

(1) 使用者連結到 .NET Passport 相關之合作網站請求服務。

(2) 合作網站導引使用者至 Microsoft .NET Passport 驗證網站進行資料登錄。

(3) 合作網站將使用者重導至 Microsoft .NET Passport 驗證網站進行身份驗證。

(4) 使用者取得 .NET Passport 的認證及授權。

(5) 使用者持授權證明向合作網站請求服務。

(6) 合作網站經由 .NET Passport Manager 判讀授權無誤後允許使用者進入。

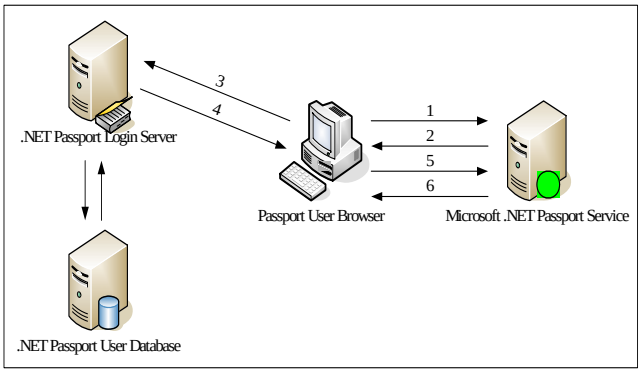


圖 3 Microsoft .NET Passport 使用者認證流程示意圖[1]

## 2.3 Peer to Peer

“Peer to Peer(P2P)”是繼”主從架構(Client-Server)”後新興的網路應用模式。在傳統的主從架構應用系統中，用戶端(client)與伺服端(server)有明確的分界，常常發生用戶端能力過剩、伺服端能力不足或網路壅塞的現象。而 P2P 系統中的使用者則能同時扮演用戶端及伺服端等多重角色，任兩個使用者之間能不透過伺服器而直接進行資訊分享或內容交換，以建構具有自主、開放、異質、延展等特性的分散式網際網路應用系統。

## 2.4 Web Service

Web Services 是一種軟體元件，它透過 Web 通訊協定及資料格式的開放式標準(例如 HTTP、XML 及 SOAP 等)來為其他的應用程式提供服務。在這裡面有兩個重點，一是它是一個提供服務的元件。二是它以 Web 的開放標準為基礎。

作為提供服務的元件，它可用來建構分散式架構系統，實現分散式架構動態整合、平衡負擔、單元升級等優點。

以 Web 的開放標準為基礎，在已經廣被使用的 Web 網路架構上來運作，採用開放式標準讓 Web Services 具有良好互通性，在不同平台上用不同程式語言建置的系統也可以輕易整合，克服目前分散式系統各自使用不同機制造成整合困難的情形。

前面說過 Web Services 是以 Web 的開放標準為基礎，其中最基本的是 HTTP 和 XML。但建構完整的 Web Services 運作還需要更多基礎，以下這些都是以 XML 為基本語法建立的重要標準。

**UDDI：**

(Universal Description Discovery and Integration) 提供註冊與搜尋 Web Service 資訊的一個標準。

**WSDL：**

(Web Service Description Language) 描述一個 Web Services 的運作方式，以及指示用戶端與它可能的互動方式。

**SOAP：**

(Simple Object Access Protocol) 在網路上交換結構化和型別資訊的一種簡易通訊協定。

## 3. 研究架構

本研究的主要目的在於提出一個多伺服器在信任模式下相互認證的 Web Based Single Sign-On 的服務架構，這個服務架構必須能夠解決前言所描述的問題：如各個 Single Sign-On 服務間的相容性、使用者認證資訊的交換、使用者資料庫存取效能、以及權限控管機制之類的問題。

因為現有的 Single Sign-On 大多數皆為單一集中認證，因前述的相關問題，接下來我們將說明本研究所提出系統架構的設計概念、系統架構。

## 3.1 設計概念

目前，大多數的 Single Sign-On 服務架構皆僅有單一認證伺服器，使用者登入這些 Single Sign-On 服務，皆由此認證伺服器來發放憑證，且僅限於同一個網域，若欲前往其他網域的 Single Sign-On 服務時仍必須再進行登入動作；所以，目前的作法就只是 Sign On 到一個網域、或者是一組網域而已。且這些 Single Sign-On 服務多數都沒有內建的權限控制機制，系統建置者必須另行製作使用者權限控制的服務，讓 Single Sign-On 服務的使用者、系統建置者或者是一般的使用者，皆無法充分利用到 Single Sign-On 的便利性。

並且，由於單一認證伺器的 Single Sign-On 服務所記錄的使用者資料量十分龐大(因為僅有單一使用者資料庫)，讓傳統的關聯式資料庫的運作效率，隨著使用者資料量的增加而逐漸降低。雖然，後來，有學者提出平衡負載單一登入(Load Balance SSO Domain)的方法，以減輕單一網站主機之負荷；但使用平衡負載機制將會面臨由不同認證伺服器所發的數位憑證，是否仍然可以繼續辨識使用者身份的問題。有鑑於此，本研究所提出的服務架構，所要改進的地方就在於：

(1) 使用各個 Web 系統既有的權限控管機制

使用各個 Web 系統既有的權限控管機制，不但可以省去修改原有系統架構的問題，也可以保留使用者之前所註冊過的使用者帳號，免除使用者須重新註冊帳號的問題。另外，因為在每個 Web 系統中，均有一個獨立的權限控管機制，所以 Single Sign-On 服務管理者不必另外再行製作或找尋其他的權限控管機制(產品)。

(2) 利用 Peer to Peer(P2P)的網路架構

點對點(Peer to Peer)資訊系統是一個分散式系統，每各個體都是獨立的，擁有自己的決策，而 Peer to Peer 的網路架構最大的特色就是分散式的處理，沒有中央管理（centralized control），方便了資源的共用。所以本研究即利用此一特性以達到各個伺服器之間認證的機制。

(3) 基於一個信任模式下

在這整個研究架構中，信任的管理是很重要的一環。因為使用者在 A Server 中可能並沒有申請過帳號，但 A Server 仍可透過 B Server 通過驗證，發給憑證，所以，伺服器間信任的管理將影響整個系統的安全性。此外，除了信任管理之外，還須考慮了其他系統階層的管理，如伺服器的階層理等。

(4) 採用 Web Services

Web Services 技術由於它們不受技術平台的限制，所以十分適合用於本研究所提出的服務架構。在 Web Services 的開放架構之下，並透過 P2P 的網路架構，將使任何使用本服務架構的服務網站建置者都能夠繼續延用自己原本的技術平台。

## 3.2 系統架構

在本系統服務架構中有三個主要的角色(如圖 4 所示)

(1) Web Sites

即在本系統架構下提供使用者各項服務的服務網站。這些網站本身擁有獨立的使用者認證及權限控管機制，認證資訊必須記錄在系統服務單位中，其認證及權限控管機制主要有以下功能：

1. 驗證一般使用者的登入並管理使用權限(即系統原有之權限認證模組)。
2. 定義由信任服務網站中所發給認可憑證的使用者之權限角色(即指定信任網站之使用權限)。
3. 驗證由信任服務網站所發出的認證請求(提供登入服務供其它服務網站進行認證請求)。

(2) Trust Server

在本研究架構下，主要是架構在一個信任的環境下，而 Trust Server 是此架構下負責與其它 Web Server 進行溝通與傳遞訊息的角色。Trust Server 也扮演此架構下 Peer to Peer 的個體，對架構下的其它 Trust Server 進行訊息的交換。

並且 Trust Server 也必須記錄該服務網站所信任的其它服務網站，如此方能建立可信任的關係，以利服務網站之間的認證以及訊息的交換。協助使用者透過原有的伺服器認證機制通過欲請求服務的其它伺服器，以達到 Mutli-Server Single Sign-On 的目的。

(3) User

即一般的使用者，泛指任何使用本系統服務架構、以及系統服務架構下之服務網站的使用者。

## 4. 系統流程

本系統依使用者登入狀況分為兩個模式，分別為由本地端登入(即原有之登入機制)及由其它信任服務網站提供認證。

## 4.1 系統運作流程

(1) 使用者向服務網站 A(Web Sites A)提出服務請求。

(2) 服務網站 A 查檢使用者的登入記錄，若有登入記錄則回覆使用者，並提供網站服務。

(3) 若無登入記錄，則回覆使用者，並要求使用者選擇由本地端登入或由其它信任服務網站提供認證。

(4) 若使用者選擇由其它信任服務網站(Web Sites B) 提供認證，則系統將 URL 重導至 Web Sites B ，並檢查使用者的登入記錄。

(5) 若使用者無登入記錄，則由 Web Sites B 向使用者提出認證須求，並要求使用者登入。

(6) 使用者由 Web Sites B 登入完成後，系統將登入結果回傳給 Web Sites A，並記錄使用者在 Web Sites B 的登入記錄。

(7) Web Sites A 收到 Web Sites B 傳回之使用者登入結果後，將使用者的權限使用權對應到指定的權限角色中(預設的信任網站使用者權限)。

(8) 將結果回覆使用者，並提供網站服務，且記錄使用者之登入記錄。

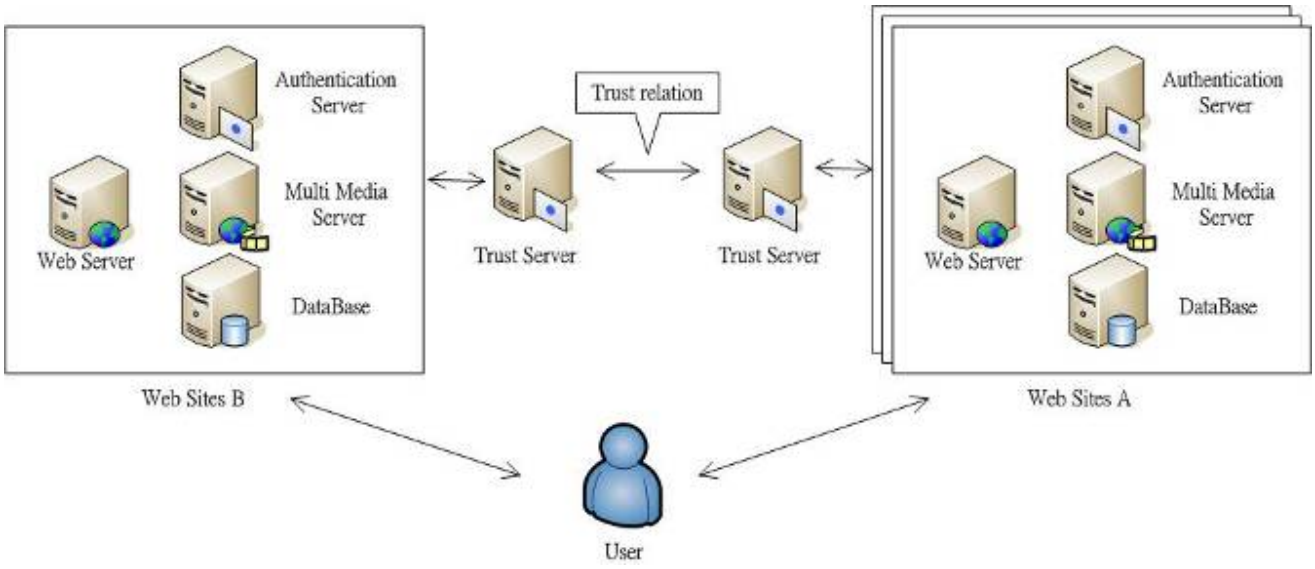


圖 4 系統架構圖

## 4.2 實例

在本章節將列舉一個案例來說明系統的運作流程及互動關係。在這案例中將提到一個網路使用者在 Web Site B 中已有使用權限且也已登入，但在 Web Site A 中並無權限，現在使用者欲透過 Web Site B 的憑證來要求 Web Site A 提供服務，其實際的運作流程及互動關係如圖 5 所示，詳細步驟如下：

(1) 使用者向 Web Server B 提出服務需求，並進行登入動作。

(2) Web Server B 將使用者的登入資訊傳給認證伺服器 B。

(3) 認證伺服器 B 確認使用者之權限，並將結果傳回 Web Server B。

(4) Web Server B 收到使用者權限認證結果後，將結果傳回使用者，並產生一組 Cookies 記錄，且提供系統服務。

(5) 使用者向 Web Server A 提出服務需求。

(6) Web Server A 將使用者的資訊傳送至認證伺服器 A，並判斷使用者有無登入記錄，若無登入記錄則回覆使用者並要求選擇本地端 Login 或選擇一個信任的服務網站進行驗證。

(7) 使用者選擇由 Web Site B 提供驗證，此時認證伺服器 A 將使用者之資訊傳送至 Trust Server A。

(8) Trust Server A 收到請求後，依使用者所選擇之信任服務網站(Web Site B)，送出認請求。

(9) Trust Server B 收到請求後，向認證伺服器 B 確認使用者之登入記錄。

(10) 認證伺服器 B 確認使用者已有登入記錄，將結果回傳給 Trust Server B。

(11) Trust Server B 將 Web Site B 驗證結果傳回給 Trust Server A。

(12) Trust Server A 確認使用者通 Web Site B 的驗證，即將結果傳送給認證伺服器 A。

(13) 認證伺服器 A 將使用者指向對應的使用權限(預設的信任網站使用者權限)，並將相關資訊傳至 Web Server A。

(14) Web Server A 收到使用者權限認證結果後，將結果傳回使用者，並產生一組 Cookies 記錄，且提供系統服務。

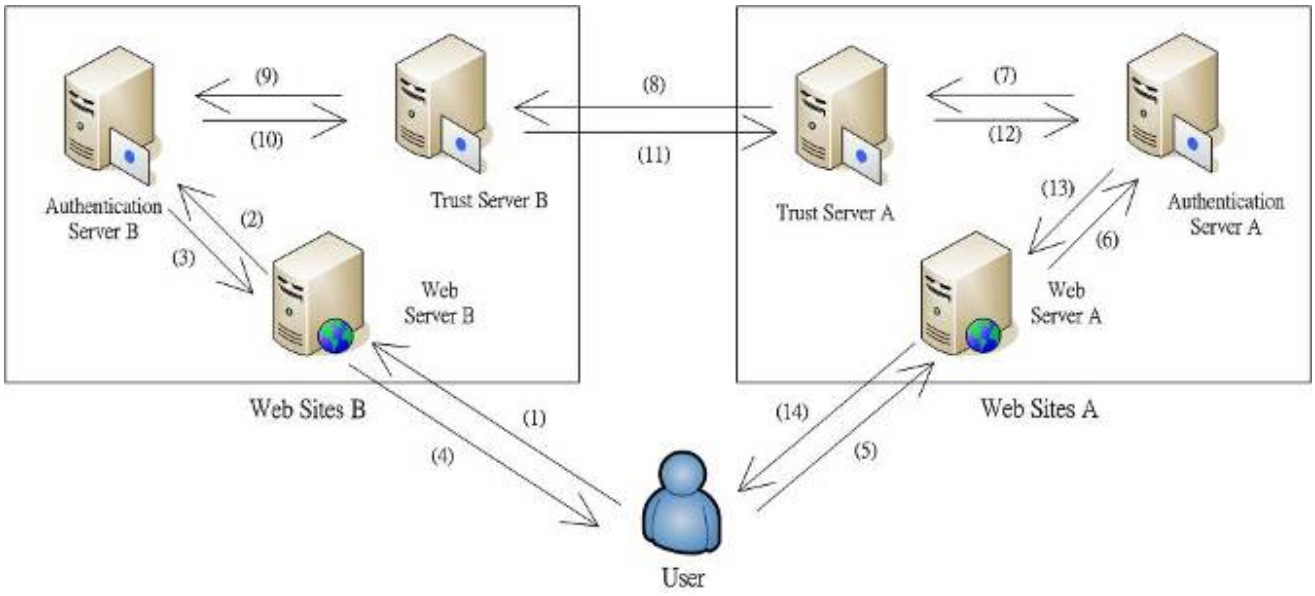


圖 5 系統流程圖

## 5. 系統實作

在這個章節中，我們將透過數位典藏系統來建立一個範例系統，來說明本研究所提出的研究架構。這個範例系統將套用本研究為主要的系統架構，並且能夠與其他信任的系統網站相互溝通、認證。

在系統方面，我們選擇兩個現有的且也已經在對外服務的系統做為範例。所選的系統為：

(1) 數位典藏聯合目錄

(http://catalog.iis.sinica.edu.tw)
作業平台：RedHat Linux 9.0
Web Server：Tomcat 4.1 + Apache 2.0
資料庫：Oracle 8i

(2) 傅斯年圖書館藏善本古籍數位典藏系統

(http://ndweb.iis.sinica.edu.tw/rarebook)
作業平台：RedHat Linux 9.0
Web Server：Tomcat 4.1 + Apache 2.0
資料庫：Oracle 8i

在系統實作的方面，我們將這兩個系統加入 Trust Server 模組，並設定其所信任的其它伺服務。在系統權限控管上，新增一個角色，為透過信任伺服務器所認證的使用者，並設定權限。

Trust Server 模組，使用 JAVA 程式撰寫，依照 Web Service 架構設計，且在 Server 與 Server 之間的傳遞採用 peer to peer 的網路架構，直接對其 Server 進行溝通與交換資料。

實作結果如下所示，圖 6 為使用者登入聯合目錄的畫面，圖 7 為使用者欲登入傅斯年善本典藏系統，且選擇一信任網域(聯合目錄)登入，而圖 8 為傅斯年善本內的權限控管機制，且已新增一角色為"聯合目錄訪客"，並付予相關權限，最後的圖 9 則為透過聯合目錄認而登入傅斯年善本的結果。

圖 6.聯合目錄登入畫面


圖 7.傅斯年善本登入畫面

圖 8.傅斯年善本權限控管畫面

圖 9.傅斯年善本登入後畫面

## 6. 結論

本研究在現有的 Web 環境中，以不改變原有的系統架構下，且透過 Server 與 Server 之間的信任關係，以不透過額外的認證伺服器下，建立一個 Web 單一登入的架構，解決了單一認證伺服器安全性及效能上的問題，且達到以最低成本完成 Web Based Single Sign-On 的目的

未來之研究，可考慮擴大本研究以整合其它的單一登入系統，使單一登入系統之範圍更為廣泛，功能性更為完善。

## 7. 參考文獻

[1] 廖英彥，「網際網路單一登入系統應用」，世新大學資訊管理學系碩士論文，2005 年。

[2] 李長庚，「一個開放的 Web-Base Single Sign-On 服務架構」，國立交通大學資訊管理研究所碩士論文，2002 年。

[3] 簡毓麟，「網站系統單次簽入之企業應用整合」，國立交通大學電機資訊學院 資訊學程研究所碩士論文，2003 年。

[4] 吳威震，「多重伺服器驗證機制之研究」，世新大學資訊管理學系碩士論文，2004 年。

[5] 張益嘉，林金龍「數位典藏系統之權限控管」，第二屆 數位典藏技術研討會，2003 年。

[6] 黃子恆，「無線點對點資訊分享網路中對無貢獻資訊享用者之無滲透研究」，輔仁大學資訊管理學系碩士論文，2003 年

[7] Microsoft, "Implement a Single-Sign On solution by using basic authentication and Internet Explorer client", http://support.microsoft.com/default.aspx?scid=kb;EN-US;837104.

[8] Microsoft, "Microsoft .NET Passport 2.5 Software Development Kit", http://msdn.microsoft.com/library/default.asp?url=/downloads/list/websrvpass.asp.

[9] Novell ," Novell SecureLogin ", http://www.novell.com/products/securelogin/index.html.

[10] Novell ," Novell SecureLogin ", http://www.novell.com.tw/Solution/Flyer/SecureLogin/SecureLogin9110HKdoc.pdf.

[11] 李昇暾、詹智安，「JAVA Web Services 實務程式設計」，旗標出版股份有限公司，2004 年