# 以 OSGi 服務平台結合 RBAC 建置企業即時通訊架構

# OSGi-based Enterprise Instant Messenger Framework Integrated with RBAC

林宸堂

嘉義大學資訊管理學系

Lin Chern Tang
National Chiayi University

Email: ctlin@mail.ncyu.edu.tw

張芳瑜

嘉義大學資訊管理學系

Chang Fang Yu
National Chiayi University

Email: blackkin0710@gmail.com

**摘要**

近年，OSGi（Open Service Gateway Initiative）服務平台在其國際聯盟的推廣下，逐漸與企業服務結合，並於 2010 年發佈 OSGi 4.2 企業版，許多資訊大廠也以此標準推出相關產品。而在現今的企業環境中，因跨區域產業的發展趨勢下，傳統通訊工具已無法滿足企業需求，而公共即時通訊軟體雖能達到企業追求高效率低成本的需求，但它卻有安全性的問題，這使得企業開始建置自己的企業即時通訊（Enterprise Instant Messenger, EIM）服務。

本研究探討文獻後認為一個安全的 EIM 服務需具備機密性(Confidentiality)、確認性(Authentication)、完整性(Integrity)、不可否認性(Nonrepudiation)、名單存取的控制和檔案的傳輸控管六項安全需求。根據上述要求，本研究欲以目前企業最常使用的「基於角色存取控制機制」（Role-Based Access Control, RBAC）結合 OSGi 服務平台來建置一個 EIM 系統架構。本研究提出以仲裁式數位簽章加上完整性驗證來達成上述的機密性、確認性、完整性、不可否認性，而以 RBAC 來做名單的存取控制和傳輸控管的要求。

*Abstract*-With the promotion of International Organizations, OSGi is combined with business services these days. OSGi 4.2 was issued at 2010, and many leading companies took it as the standard and launched products. But now, cross-regional industries' development, the traditional communication tools cannot meet the business needs. Although the low cost and high efficiency of Public Instant Messenger Software, it possesses security problems. This phenomenon makes companies start to develop their own Enterprise Instant Messenger services (EIM).

By studying literatures, the research found out that a safe EIM should have the characteristics of Confidentiality, Authentication, Integrity, Nonrepudiation, access control of the list, and the control of file transmission. This research aims to fulfill the first four requirements with Arbitrated Digital Signature and integrity, and meet the needs of access control of the list and the control of file transmission with RBAC.

## 一、 前言

在 2007 年 EclipseCon 研討會（由 IBM、Cisco、Oracle 等廠商舉辦）中，Spring-OSGi、Web Service 與 OSGi 等技術引起資訊產業高度關注，在"Of enterprise-class open source Java framework for OSGi"一文中也指出 OSGi 將在未來與企業應用緊密結合[1][2][3]。

現今企業的資訊應用服務中，使用即時通訊（Instant Messaging, IM）服務在同一組織或跨組織間進行資訊溝通已有增多的趨勢。近年，隨著通訊、軟體和網路三大現代訊息技術的發展，企業即時通訊（Enterprise Instant Messenger, EIM）已成為繼電話、E-mail 後，又一個企業界溝通的革命新浪潮。

然而即時通訊服務在導入企業的初期，不同企業對即時通訊抱持著不同的態度，第一，擔心影響員工的效率及企業機密的外洩，所以明文禁用。第二，肯定即時通訊的效率，讓員工使用即時通訊

軟體(MSN Messenger、Yahoo! Messenger、ICQ、AOL、Skype 等)互相聯絡事務，甚至使用 Skype 的網路電話解釋較複雜的事務，以節省電話費。現今，企業越來越重視 EIM 軟體，部份原因為公共即時通訊軟體(意指 IM Server 在公共網路中提供給所有人使用即時通訊服務的軟體，如前述的 MSN Messenger 等)的盛行，讓企業員工習慣使用即時通訊軟體作為溝通工具。再加上即時通訊滿足人們追求即時、快捷、近距離溝通的要求，更降低通訊的成本。所以，對企業而言，即時通訊軟體的使用已經不再是討論是否開放的問題，而是應關注於如何建置一套安全、便利且能提高員工在溝通上的效益及降低企業通訊成本的企業即時通訊軟體。

根據 IDC 調查，2007 年全球整合通訊(Unified Communications, UC)在微軟積極佈局後，提升 UC 市場熱度，而其它資訊大廠也認為企業即時通訊軟體的市場潛力無窮，相繼推出即時通訊整合產品，例如，微軟推出企業即時通訊軟體與協同合作平台 Office Live Communication Server，將電話、即時通訊、E-mail 多種通訊工具整合在同一平台；IBM 也提出 Workplace 平台，將企業即時通訊軟體、郵件等溝通工具整合在同一平台上[1]。

## 二、 文獻

### 2.1 即時通訊

IETF(Internet Engineering Task Force)於 2000 年提出 RFC 2778 標準[4]，其定義即時通訊系統由兩種服務構成，分別為 Presence Service 與 Instant Messaging Service，如圖 1 所示，圖 1(a)的 Presence Service 負責使用者狀態訊息的交換與通知，Watcher 可以透過 Presence Service 得知 Presentity 的狀態資訊。圖 1(b)的 Instant Messaging Service 負責使用者之間的即時訊息交換，當 Sender 欲傳送訊息給接收者，由 Instant Messaging Service 將即時訊息轉送至接收者的 Instant Box，接收者再透過即時通訊 Client 軟體接收儲存於 Instant Box 上的即時訊息。從圖 1 可以發現，即時通訊的本質為三方架構，而 Client 之間的訊息傳遞需透過 Server 扮演傳遞的角色。

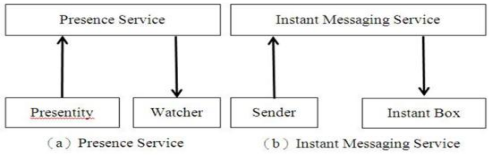


圖 1 即時通訊系統兩種基本服務模型

RFC 2779[5]定義了 RFC 2778 的服務模型需求，其中 RFC 2779 提出 Instant Messaging 訊息傳遞過程不可遭到竊聽、篡改等安全需求。其中 RFC 2779 定義其理想狀況為透過 Server 交換訊息和金鑰，也可確保其確認性(Authentication)、機密性(Confidentiality)、完整性(Integrity)與不可否認性(Nonrepudiation)，且 Server 無法得知通訊雙方的私密金鑰。

### 2.2 即時通訊軟體的安全問題

IETF 分別在 2002 和 2004 年提出以 RFC 2778 和 RFC 2779[4][5]為基礎的即時通訊協定標準，分別為 RFC 3428 的 SIMPLE(Session Initiation Protocol (SIP) Extension for Instant Messaging)和 RFC 3920 的 XMPP(Extensible Messaging and Presence Protocol)。在安全機制上 SIMPLE 使用 S/MIME 安全訊息格式或以 TLS 建立加密通道保護通訊安全。而 XMPP 使用 XML 作為訊息傳輸格式，其以 SASL 支援身份驗證機制或以 TLS/SSL 建立加密通道以保護通訊安全。

雖然 SIMPLE 和 XMPP 通訊協定早已提出，然而在實務上，目前幾種公共的即時通訊系統卻未完全遵循 SIMPLE 和 XMPP 通訊協定，表 1 為目前公共即時通訊系統的安全機制的比較。由此表我們可以發現除 Skype 以外其餘的 MSN Messenger、Yahoo! Messenger、ICQ、AOL 皆只有身份驗證機制，而身份驗證成功後，皆是以明文方式傳輸。表中 C/R 表示 Challenge/Response 的身份驗證方法，如圖 2 所示，Server 會先傳送一 Challenge string 給 Client，Client 再將其與 Password 以 Hash function 計算雜湊值後，將其 ID 與雜湊值回傳給 Server 作認證，在身份驗證成功後則以明文傳遞訊息，有心人士只要使用 Wireshark 等軟體即可輕易竊聽通訊[6]。

表 1 公共即時通訊軟體的安全機制比較

| | MSN Messenger | Yahoo Messenger | ICQ | AOL | Skype |
|---|---|---|---|---|---|
| 身份驗證機制 | C/R | C/R | C/R | C/R | 獨家演算法 |
| Client - Server 通訊過程加密 | × | × | × | × | AES |
| 訊息繞經 Server 時加密 | × | × | × | × | |
| 金鑰交換架構基礎 | × | × | × | × | Certificate Authority |

「×」：沒有 C/R：Challenge/Response

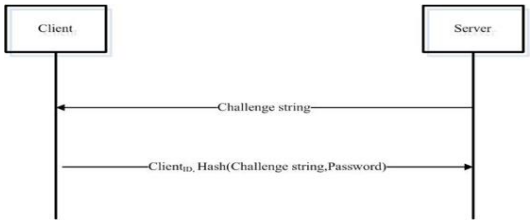


圖 2 Challenge/Response 身份驗證模型

**2.3 OSGi**

Open Service Gateway Initiative（OSGi）[3]創立於 1993 年 3 月，為一非營利組織。在面對產品生命週期縮短、功能需求提昇及多樣的作業系統與硬體，導致軟體複雜度越來越高，造成軟體研發與維

護成本日漸增加，OSGi 認為需要一種標準化的工具去整合軟體的設計觀點，達到能夠再使用(reuse)現存元件的目標。OSGi 技術便是一種基於 JAVA 的動態模組系統，藉由 JAVA 跨平台的特性能夠於不同的作業環境下執行，並且將應用軟體模組化，能夠依需求而快速部署的系統環境，其服務平台的分層架構如圖 3。

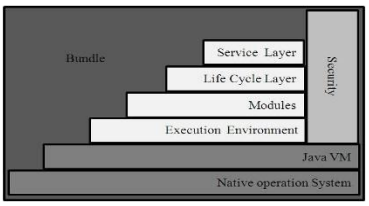


圖 3 OSGi Framework 的分層架構

OSGi 目前有 24 家正式成員（如：IBM、NEC、NTT、Oracle、Red Hat、VMWare 等），14 家協會採用 OSGi，還有超過一百家廠商的產品及服務支援 OSGi，其應用領域包含企業應用、居家環境、行動通訊、消費性電子，目標為建立一個市場通用的中介軟體。目前最新的標準是 2010 年 3 月推出的 OSGi Enterprise Version 4.2。

**2.4 以角色為基礎的存取控制**

以角色為基礎的存取控制(Role-Based Access Control) [7][8][9]，簡稱 RBAC 是以角色為基礎的存取控制，是在 1992 年由 Ferraiolo 和 Richard.Kuhn 等學者所提出的概念，之後經過 Sandhu 等學者加以研究改進，由 National Institute of Standards and Technology(NIST)組織加以收編、整理之後訂定標準，稱為 NIST RBAC，其主要是利用使用者 - 角色 - 權限這三層之間的對應來控管使用者的權限，也就是我們根據使用者在組織中的工作內容或是其職務來分配角色給該使用者，藉由角色 - 權限之間的對應，每個角色可以擁有多個權限，如此使用者便可以經由所分配到的角色間接地獲得權限，而當使用者的職務有所更動時，我們只需要調整使用者所擁有的角色即可，可以降低管理的成本及增加管理上的便利性。

其核心架構如圖 4，包含三個主要的元件，使用者( Users )、角色( Roles )與權限( Permission, PRMS ), 在權限之中又分為操作( Operations, OPS ) 與物件（Objects, OBS）。使用者需要等到被指派角色之後，才可以角色去行使物件的權限。其中有使用者-角色的指派（Users Assignment, UA）與角色-權限的指派（Permission Assignment, PA），兩者都是多對多的關係，意即一個使用者可以具有多個角色，同個角色也可以指派給多個使用者；同時一個角色可以包含多個權限在內，而權限也可指派給多個角色。RBAC 是利用角色指派權限，與傳統以使用者指派權限的方式不一樣，RBAC 的特點在於，當使用者發生變動時，不會連帶產生權限的變動；同樣的，當權限有變動時，也不會去牽動到使用者。變更的管理可以全交由角色來處理，如此可讓管理的複雜度降低，而且比較有效率。另外，會期（Sessions）運用(即 User_sessions 關係 - 使用者對 Session 為一對多的關係，也就是一個使用者可以對應到多個 Session，但是一個 Session 只會對應到一個使用者)，而這個 Session 會在使用者

所擁有的眾多角色之中啟動一個或數個這個任務所會利用到的角色(即 Session_roles 關係 - Session 對角色為多對多的關係，也就是一個 Session 可以擁有多個角色，而每個角色也可以同時被多個 Session 所利用)，以此限制使用者不會有過多且過大的權限,以達到最少權限的安全性原則。

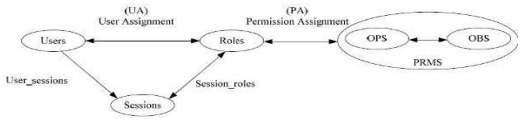


圖 4 核心 RBAC

階層 RBAC 比核心 RBAC 則多了 Role Hierarchy 的關係，如圖 5，角色和角色之間有繼承的關係。當角色 A 繼承了角色 B 的話，那麼角 A 則擁有角色 A 以及角色 B 自己的權限。因此不需要位每一個角色設定全部所需要的權限,一些基本的權限，可以利用繼承的方法獲得。

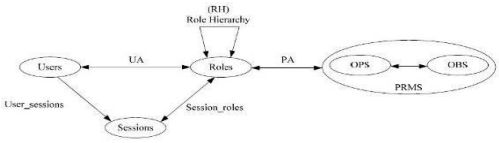


圖 5 階層 RBAC

## 三、 研究環境與需求

由於即時訊息具有便利、快捷、直接等優點，非常適合企業內部或企業和客戶間的交流。但如果把公共即時通訊系統帶到公司內部使用,會產生下列幾項問題：

- 降低工作效率：企業無法區分公事使用和非公事使用，以至於若員工上班聊天，將影響工作效率。
- 機密資料外洩的風險：公共即時通訊軟體的文字和檔案傳送功能比電子郵件更快速，使公司資訊安全管理面臨更大的挑戰。
- 訊息傳遞的安全性：經由即時通訊系統傳至公司內部的文件檔案，可能包含病毒及有害的程式，將有可能造成公司嚴重的資訊安全問題。
- 無企業需要的功能：目前的公共即時通訊軟體是由遊戲、娛樂、資訊等的整合，無企業所需功能。

正因為上述這些問題，適用於企業的即時通訊服務因應而生。目前資訊大廠也紛紛提出企業即時通訊服務，如微軟推出 Office Live Communication Server、IBM 提出 Workplace 平台。

而本研究所提即時通訊服務有以下幾點的需求：

1. 為使企業可安全的使用即時通訊服務,本研究使用仲裁式的數位簽章和 RBAC 的概念設計即時通訊服務的協定，以達到 Client-Server 和 Client-Client 通訊的確認性、機密性、完整性、不可否認性及對檔案的存取控制。
2. 為達到企業機密的不外洩的需求,本研究使在 EIM Server 上用一過濾機制,並將每一員工所傳輸之訊息做紀錄，以監控員工使用即時通訊,並防止員工將企業機密訊息或檔案外洩。

而本研究的最大的優勢為 OSGi 為開放式的架構，這有利於將企業其他系統做整合，而微軟的 Office Live Communication Server 則只能整合微軟自行推出的產品(如 Microsoft Office System、以 Microsoft Office SharePoint Portal Server 建置的企業入口網站、電子郵件系統等);IBM 的 Workplace 雖也可整合企業其他的系統，但其為付費軟體。

## 四、 系統設計

了解企業即時通訊服務的現況以及使用環境後，本研究將規劃出適合企業使用的即時通訊服務，如圖 6。

本研究所採取的即時通訊架構是依據 RFC 2778,也就是使用者登入、改變狀態和訊息傳遞皆需透過 EIM Server 來進行(如圖 7),原因為企業即時通訊服務所傳遞的訊息和使用情形皆需接受企業的控管,為此本研究使用仲裁式數位簽章的方法(在傳送者與接收者之間加入一仲裁者,藉以檢查來源和內容，而在本研究中，仲裁者為 EIM Server)且使用者和 EIM Server 皆建立在 OSGi 的服務平台上，其各自平台上的 Bundle 如圖 8，在使用者端上有認證服務和通訊服務兩個 Bundle，而 EIM Server 上有過濾訊息、紀錄訊息、認證服務和通訊服務四個 Bundle，而在同一企業內本研

究視為內部網（Intranet）環境，而跨企業的為外部網（Extranet）環境。為解說方便表 2 為使用的符號對應表。

以下為企業即時通訊服務功能做詳細說明，其環境假設如下列：

1. 在 EIM Server 中皆存有企業中每個使用者的帳號密碼。
2. 在 EIM Server 中皆存有和各合作企業的預設共享金鑰（Pre Share Key）。
3. 本研究假設其傳送之訊息皆可成功的傳送到接受方。
4. 本研究假設使用者自訂的密碼與企業間預先共享金鑰其設定符合系統安全的基礎。

表 2 符號說明表

| 符號 | 說明 |
|---|---|
| $ID_a$ | 使用者 a 的 EIM 帳號 |
| $Firm_ID_X$ | 企業 X 的 ID |
| $PW_a$ | 使用者 a 登入即時通訊服務的密碼 |
| $PSK_{XY}$ | 企業 X 與企業 Y 的預先共享金鑰 |
| $EIM_X$ | X 企業的 EIM Server |
| $TYPE$ | 使用者的狀態 |
| $T$ | 時戳 |
| $LIST_a$ | 使用者 a 的聯絡人清單 |
| $K_{ac}$ | a 和 c 的連線金鑰（Session Key） |
| $Req$ | 需求訊息 |
| $MSG$ | 即時通訊服務的聊天訊息 |

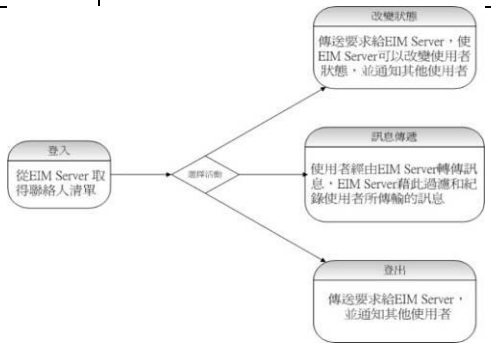


圖 6 企業即時通訊服務功能圖

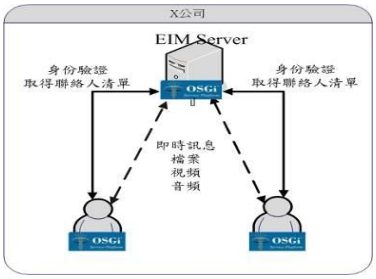


圖 7 系統架構圖

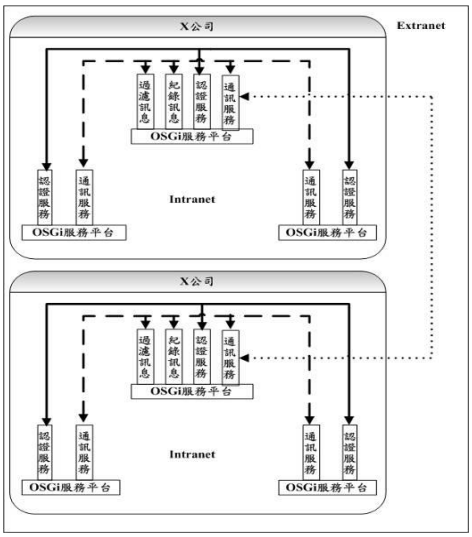


圖 8 Bundle 運作圖

● 登入

在使用企業即時通訊服務前，使用者需向 EIM Server 做認證的動作（如圖 9），此功能必須達到機密性、防止重送攻擊和完整性驗證的安全性考量。在使用者身份確認後，EIM Server 會依據此使用者在企業中所擔任的職位和角色，回傳此使用者可看到的聯絡人清單。登入的 Login 協定設計如圖 9，圖中使用者 a 為 X 企業的一個員工，而 $EIM_X$ 為 X 企業的 EIM Server，首先，使用者 a 會先傳送一加密訊息給 $EIM_X$，訊息內容包含 $ID_a$、$TYPE$（使用者的初始狀態，如線上或忙碌等）、和 $T$（目的為防止重送攻擊），此訊息以 $PW_a$ 作為加密金鑰傳送給 $EIM_X$。而 $EIM_X$ 收到後，會經由資料庫進行身份認證，確認後 $EIM_X$ 回傳 $LIST_a$，和使用者 a 和 $EIM_X$ 的連線金鑰（Session Key）$K_{aX}$（此金鑰用於雙方傳輸資料時可以用此 Session Key 作加解密的處理），此金鑰為每次使用者登入時，EIM Server 以亂數產生，其時效為一次登入直至登出的時間，最後加上目前系統時間 $T$ 作為

$EIM_X$ 傳輸時的時戳，此訊息以 $PW_a$ 為加密金鑰加密 $K_{aX}$ 和時戳 $T'$，而 $LIST_a$ 則使用 $K_{aX}$ 加密回傳給使用者 C，這樣可降低 $PW_a$ 的加密量，以增強其安全性。

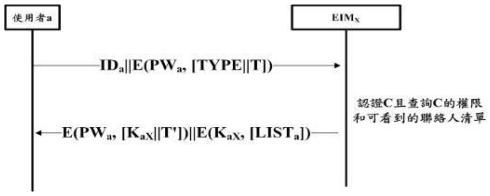


圖 9 Login 協定

此服務當中的聯絡人清單（此清單包含聯絡人的帳號、狀態），其清單的產生則是依據階層 RBAC 所產生，如圖 10 所示，X 公司的總工程師會繼承硬體工程師和軟體工程師的名單,而硬體工程師會繼承硬體助理和工程師的名單,以此類推，此方法可縮短設定時間，及維護的複雜度。

而此清單不但包含使用者 a 自己所屬的部門同事清單且包含使用者 a 有參與的跨企業的專案的清單，以圖 10 舉例來說，由於 X 公司軟體部門與 Y 公司的會計部門有合作，所以 X 公司的軟體工程師不只可看見其軟體助理和工程師的名單,還可看見 Y 公司會計師和稽核師的名單。所以 EIM Server 需定時去向合作企業的 EIM Server 拿取其名單，此協定的設計如圖 11，圖中企業 X 向企業 Y 拿取名單，企業 X 會以和 Y 企業的預先共享金鑰（$PSK_{XY}$）加密要取名單的要求（$Req$）和時戳（$T$），再加上企業 X 的 ID（$Firm_ID_X$），傳送給企業 Y 做認證，$EIM_Y$ 以資料庫做認證後，以亂數產生企業 X 和企業 Y 間 EIM Server 的連線金鑰（$K_{XY}$，其時效為一天的時間）和時戳以 $PSK_{XY}$ 加密，而名單（$LIST_X$）以 $K_{XY}$ 加密後傳送給 $EIM_X$。

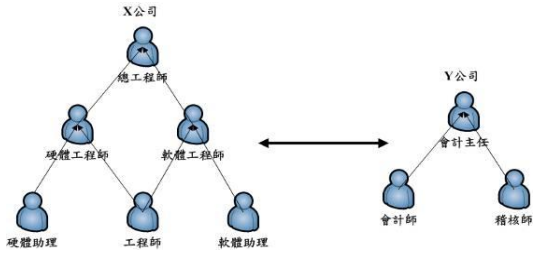


圖 10 公司階層圖

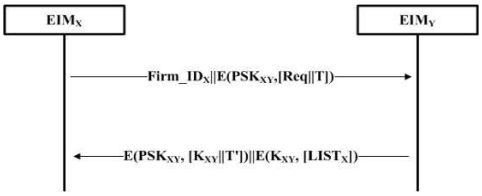


圖 11 EIM Server 和合作公司拿名單協定

● 改變狀態和登出

使用者在使用即時通訊服務時，必須依據實際情形改變狀態，如離開、開會中等，所以本研究也依據此需求來設計其對應的協定，在此協定中，當使用者改變其狀態時，會傳送需求給 EIM Server，EIM Server 在接受到要求後，會將此使用者改變後的狀態傳遞給被允許看到該使用者狀態的所有人，在此項服務中，本研究希望能達到機密性、防止重送攻擊、完整性驗證安全考量。改變狀態和登出協定的設計如圖 12，圖中使用者 a 會將新的狀態和時戳，用使用者 a 和 $EIM_X$ 的連線金鑰 $K_{aX}$ 加密後傳送給 EIM Server，再由 EIM Server 去通知其他名單有 a 的使用者。而 $EIM_X$ 收到使用者 a 的要求後，會去查詢被允許看到使用者 a 狀態的人，然後通知變更，在此協定中會有兩種情形，一是 $EIM_X$ 通知企業 X 中被允許看到使用者 a 狀態的所有使用者,如圖 13 的使用者 b,在圖中訊息用 $EIM_X$ 和使用者 b 的連線金鑰 $K_{bX}$，將使用者 a 的 $ID_a$、狀態和時戳加密後傳送給使用者 b 通知使用者 a 的狀態已改變。二是 $EIM_X$ 會通知合作企業中可看到使用者 a 狀態的名單，如圖 14，圖中 $EIM_X$ 要通知企業 Y 使用者 a 的狀態已改變，$EIM_X$ 會將使用者 a 的 $ID_a$ 和其狀態跟時戳用與 Y 企業的連線金鑰 $K_{XY}$ 加密之後傳送給 $EIM_Y$，$EIM_Y$ 在收到後會將其再傳送給可看到使用者 a 狀態的使用者 c。

圖 12 改變狀態&登出協定

圖 13 狀態改變通知協定

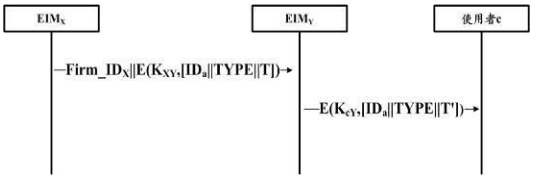


圖 14 狀態改變跨企業通知協定

● 訊息傳遞

在訊息傳遞的服務中，本研究將其分為企業內部網（Intranet）與企業外部網（Extranet）的溝通服務。在企業內部網（Intranet）的溝通服務中，所有使用者的訊息需透過 EIM Server 轉傳給接受方，EIM Server 轉傳訊息時會將此使用者所有的訊息做記錄，以方便日後發生問題時的查證。而在企業外部網（Extranet）的溝通服務中，使用者欲將訊息傳遞給合作公司的使用者時，也需透過自己企業的 EIM Server 轉傳訊息給合作公司的 EIM Server，而合作公司的 EIM Server 再將此訊息傳遞給接收方，然而在企業外部網的訊息傳遞服務中，發送方的 EIM Server 不只會記錄所有的訊息，還會透過一篩選機制（包含關鍵字過濾法和 RBAC 對檔案的存取機制）來過濾訊息，以防止使用者將企業的機密外洩。而在此項服務中，本研究希望能達到機密性、防止重送攻擊、完整性驗證、不可否認性和檔案的存取控制。

企業內部網（Intranet）協定的設計如圖 15，圖中使用者 a 會將欲傳送對象的 ID 和傳送的訊息（*MSG*）加上時戳用 $K_{aX}$ 加密之後傳遞給 EIM Server，EIM Server 在收到後，會將其解密，檢查其完整性，並紀錄使用者 a 所傳遞的訊息，然後再傳送給 a 欲傳送的對象。企業外部網（Extranet）協定的設計如圖 16，其協定設計與 Intranet 的設計相同，只是在傳送過程中，需先將訊息傳到合作企業的 EIM Server，再經由其轉傳。

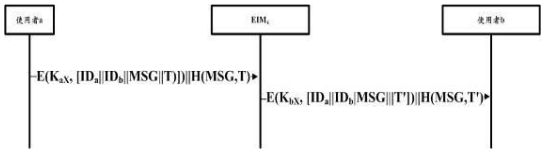


圖 15 Intranet 訊息傳遞協定

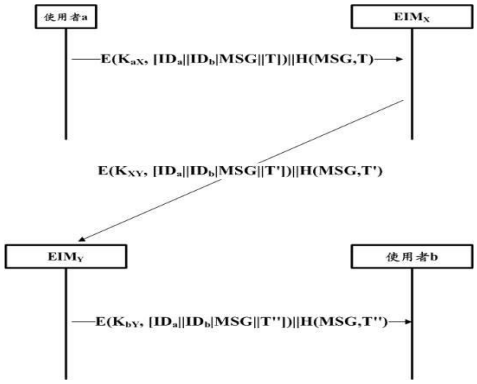


圖 16 Extranet 訊息傳遞協定

## 五、 結論

本研究依據 OSGi 服務架構，使用仲裁式數位簽章加上完整性驗證達成機密性、確認性、完整性、不可否認性，並透過 RBAC 的概念過濾訊息及設定存取名單，使企業可以達到安全也便於監控的即時通訊服務，且由於 OSGi 開放式平台的優勢，便於企業整合其他的資訊系統，以提供更多元的服務。

但本研究尚有下列問題尚未解決，期待未來研究者可針對其問題加以解決：

1. 本研究假設其傳送之訊息皆可成功的傳送到接受方，但實際上，由於網路的因素，有可能會導致訊息傳輸失敗。
2. 本研究假設使用者自訂的密碼與企業間預先共享金鑰其設定符合系統安全的基礎，但實際上，使用者自訂的密碼往往過短，且由於這兩把金鑰使用頻繁，所以其容易被破解。

## 參考文獻

[1] IBM. [Online]. Available: http://www.ibm.com/tw/zh/.

[2] Eclipse. [Online]. Available: http://www.eclipse.org/.

[3] OSGi. [Online]. Available: http://www.osgi.org/Main/HomePage.

[4] Day, M., Rosenberg, J., and Suugano, H., “A Model for Presence and Instant Messaging,” IETF RFC 2778, 2000.

[5] Day, D., Aggarwal, S., Mohr, G., and Vincent, J., “Instant Messaging / Presence Protocol

Requirements," IETF RFC 2779, 2000.

[6] 羅于庭，民 98，安全即時通訊之視訊加密系統設計與實現,國立高雄師範大學資訊教育研究所碩士論文。

[7] Sandhu R. S., Samarati P., "Access Control: Principles and Practice", IEEE Communications Magazine, pp. 40-48, 1994.

[8] R. S. Sandhu, E. J. Coyne and C. E. Youman, "Role-Based Access Control Models, " IEEE Computer , pp. 38-47, 1996.

[9] R. S. Sandhu, D. Ferraiolo and R. Kuhn, "The NIST Model for Role-Based Access Control： Towards a Unified Standard, " Proceedings of the 5th ACM Workshop on Role-Based Access Control, pp. 26-27, 2000.