國立交通大學

資訊管理研究所

碩士論文

在 MANET 下以 SIP 為基礎的安全群播協定

An SIP-Based Secure Multicasting Procotol on MANETs

研究生：楊仁豪

指導教授：羅濟群博士

中華民國 九十五 年 六 月

在 MANET 下以 SIP 為基礎的安全群播協定

An SIP-Based Secure Multicasting Protocol on MANETs

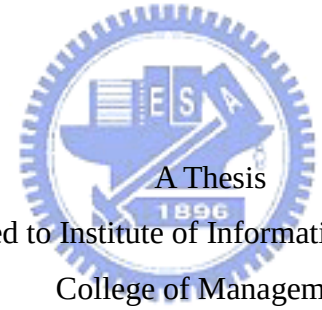
研 究 生：楊 仁 豪　　Student: Zin-How Yang

指導教授：羅 濟 群　　Advisor: Chi-Chun Lo

國立交通大學

資訊管理研究所

碩士論文



A Thesis

Submitted to Institute of Information Management

College of Management

National Chiao Tung University

in Partial Fulfillment of the Requirements

for the Degree of

Master of Business Administration

in

Information Management

June 2006

Hsinchu, Taiwan, the Republic of China

中華民國　九十五 年 六 月

在 MANET 下以 SIP 為基礎的安全群播協定

研究生：楊仁豪　　指導教授：羅濟群 老師

國立交通大學資訊管理研究所

摘要

Mobile ad-hoc network (MANET) 是一種可以在無固定中央管理建設下隨時隨地提供網路服務的一種網路架構， MANET 的特性包括: 多階性，動態拓墣，無線網路，SIP(Session Initiation Protocol, RFC 2345)是一種應用層的控制協定，用來建立、修改和終止多媒體的 Session，在西元 2000 年 SIP 被 3GPP 選擇當做 IP 行動網路的連線控制協定，SIP 使用明文方式來傳輸訊息，所以，SIP 在做訊息傳輸的同時會相當輕易的被竊聽者獲得相關的資訊，也相當容易遭到竄改或者是破壞，SIP 在 MANET 上的應用已被廣泛的討論，但是卻忽略安全性議題，在本論文中，我們將探討 SIP 在 Mobile Ad-hoc Network 上訊息傳遞間的數位簽章與金鑰交換，提出一個解決方案，並且實驗證明其安全性與效能。

關鍵字：SIP、數位簽章、金鑰交換、MANET、Ad Hoc Network

An SIP-based Secure Multicasting Protocol on MANETs

Student: Zin-How Yang Advisor: Dr. Chi-Chun Lo

Institute of Information Management

Nation Chiao Tung University

Abstract

Moblile ad-hoc network (MANET) is an active topic of research for it potential of providing pervasive services anytwhrere,anytime,including providing a platform in absence of a fixed, centeral-managed infrastructure.The characteristic of MANET is dynamic topolofy,wireless network and multi-hope.SIP is an application layer control protocol that can establish ,modify,and terminate multimedia session.SIP is selected as the call control signaling for 3G IP multimedia Core Network by 3G.Because SIP transfers message in plaintext,SIP can't defense eavesdropping and modifying perfectly.SIP application in MANET is discussed popularly.Ths security of SIP application in MANET offen be ignored.In this paper,we will discuss the security of SIP message transferring in MANET.By using digital signature and key exchange,we proposed a new message transferring protocol to enhance the security of SIP application in MANET.

Keyword: MANET、SIP、Digital Signature、Key exchange、ECC

誌謝

碩士班的兩年讓我成長與學習了很多，學習研究的過程，學習待人接物，學習包容與內斂，也學習到了感情的相處與包容成長，交大讓我學習到很多大學時代所沒辦法觸碰的領域，也讓我知道業界現實的走向，讓我在求職的時候獲得很大的幫助。

這兩年來，要感謝很多人，風格自由又不失嚴謹的指導教授羅濟群老師，明確的指引我論文研究的方向，雖然開會的次數不多，但是每次都能夠有很明確的方向，讓我不斷的修正錯誤與缺失，真心的感謝老師這兩年來的指導，接著要感謝這兩年陪伴我成長的實驗室夥伴，髒鬼義與帥氣超超，一起聊天喇賽，超董還會不定期的補充硬體資源，讓網路實驗室的同仁不會因為硬碟空間的限制而影響對點對點網路效能與服務品質的研究產生障礙，還要謝謝打電動的好朋友颯車欽，雖然親眼目睹過車神在我面前摔車，但是車神對於過彎技術與直線加速極限追求的氣魄卻從來沒有減少過，還有一起打冰風之谷的周末下午，跟打星海的深夜，真的很開心到研究所還有可以一起打電動的朋友，謝啦，小欽欽，還有最愛喇賽的中環，雖然你有時候很愛講廢話，不過在你身上還是學到很多東西，刺眼的香蕉，本來都以為你是超級乖學生，但是，後來發現，你其實白爛程度可以跟小欽欽還有髒鬼義比擬，跟你聊天很開心，其實對我論文成敗影響最多的，是帥氣強壯又結實的鼎元學長，謝謝學長在我口試的前幾天，仔細的指導我論文報告的方式，與呈現的技巧，讓我在口試的時候順利到不敢相信，還有學長平常對我也是超級照顧的，除了打球的時候會讓我，也會告訴我很多事情，真的讓我受益非淺，還有在英國的帥氣俊龍學長，在我做論文的初期給了我很多建議與幫助，還有在口試當天，超有義氣的時間控制，讓我論文多了幾分加持，跟常常麻煩到的俊傑學長都給我很多方向與照顧，最後要謝謝讓我順利的楊建民教授，不只和藹可親還幫我解答問題，政大的老師還是很棒低，陳彥良教授跟劉敦仁教授也謝謝你們的指導。

最後我要感謝土地公公讓我順利的考上交大與平安的唸完碩士班，最最重要的是支持我的家人，爸爸，媽媽，還有兩個姊姊，讓我沒有後顧之憂的生活與成長，真的很感謝你們的支持。最後最後，我要謝謝這兩年來不斷鼓勵我的栩嘉，很開心認識妳，喜歡妳，愛上妳，妳陪伴我的每一天都很開心，一起走過困難與開心的日子，希望以後也可以一直這麼下去，我的這兩年的生活因妳而閃耀與發光，希望在未來的日子也可以繼續快樂下去，一起加油，謝謝所有陪我走過這段日子的人。

目次

圖目次

表目次

第一章、緒論

1.1 研究背景與動機

Mobile ad-hoc network (MANET) 是一種可以在無固定中央管理建設下隨時隨地提供網路服務的一種網路架構， MANET 的特性包括: 多階性，動態拓撲，無線網路，不可信賴的通訊通道，由於動態拓撲所以每個點可以自由的進出這個網路，使得集中管理與分析的方法不適用於 MANET，也因為無線網路以及通訊通道間的不可信任，所以無線網路間節點的通訊，更是容易竊聽以及竄改以及破壞，所以通訊間的資料傳輸如何確保其完整性以及私密性，是 MANET 上受到矚目的一點。

在西元 2000 年 SIP[7]被 3GPP 選擇當做 IP 行動網路的連線控制協定，SIP 和 HyperText Transfer Protocol (HTTP)及 Simple Mail Transport Protocol (SMTP)使用明文方式來傳輸訊息，所以，SIP 在做訊息傳輸的同時會相當輕易的被竊聽者獲得相關的資訊，也相當容易遭到竄改或者是破壞，SIP 在 MANET 上的應用已被廣泛的討論[1][9]，但是卻忽略安全性議題，在本論文中，我們將探討 SIP 在 Mobile ad-hoc network 上訊息傳遞間的數位簽章與金鑰交換，並且提出一個有效的解決方案，實驗並證明安全性與效能。

1.2 研究目的

本論文中根據上述的研究動機，於達成以下目的：

1. 討論 SIP 在 MANET 上目前使用的運作架構[1][9]，包括使用廣播，以及群播的架構，分析其優缺點。

2. 討論使用群播的 SIP 架構在 MANET 上的安全議題，根據其在完整性以及私密性上的安全疑慮，提出藉由使用數位簽章，以及橢圓曲線 Diffie-Hellman 的金鑰交換法的新架構，並且討論與證明其在安全性上的加強。

3. 實作出系統並且分析記錄使用新架構後的效能以及拓樸建立的時間花費，並且討論架構的效能以及其安全性架構。

1.3 論文架構

本篇論文共分為五章，第一章為「緒論」，對研究動機、背景、目的以及論文整體架構作一簡單的說明。第二章為「文獻探討」，介紹與本研究相關的主題。第三章為介紹使用數位簽章以及金鑰交換法的新架構。第四章為介紹系統實作的流程以及實驗的數據與結果。第五章為「結論與未來工作」，為總結本研究的研究成果，並針對未來可能的研究方向加以說明。

第二章、文獻探討

本章將要對於目前 SIP 在 MANET 上的應用有關的文獻進行探討，並且根據其所提出的架構，討論其在安全性上的疑慮，再提出可能的解決方式，本研究欲在一個 MANET 網路上使用 SIP 協定架構起一個具有訊息完整性，以及私密性的通訊環境。

2.1 Session Initiation Protocol (SIP) 的介紹

SIP(Session Initiation Protocol, RFC 3261)[7]是一種應用層的控制協定，用來建立、修改和終止多媒體的Session，如網際網路電話。SIP 僅是 Session 發起時的一種初始化的協定，它可以與其它的協定結合以實做多媒體的通訊。例如與RTP(Real-time transport Protocol)協定結合可以傳送即時的資料及提供品質服務的回應；與RTSP(Real-time Streaming Protocol)協定結合可用來控制 streamingmedia 的派送； 與SAP(Session Announcement Protocol)協定結合可以用來advertising multimedia session via multicast 及透過SDP(Session DescriptionProtocol)協定可以用來描述多媒體的session。SIP 協定雖然可以與其它的協定結合以提供使用者完整的服務，然而SIP 協定的功能及運作並不需要依靠其它的協定來完成。

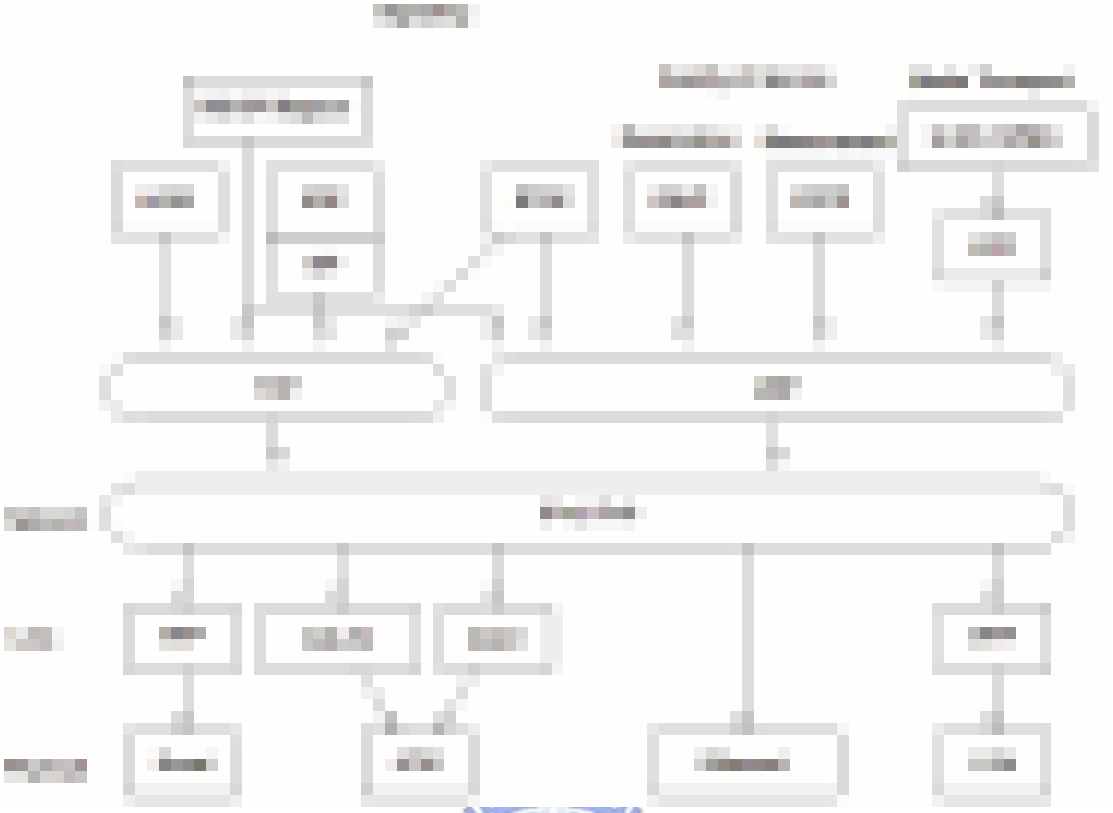


圖 1、SIP 架構圖

整個SIP 分為兩種部分，為一種Client/Server架構，所以，SIP元件可以分為，SIP Client與SIP Server的部份：

I. SIP Clinet部分:

UA(User Agent)：是一個位於被使用者端的程式，一般使用SIP Agent都使用5060的阜號(Port)，他可以用來撥打電話以及設定本身的資訊包括SIP URI與使用者姓名等，它也會詢問使用者要如何處理播打進來的通話，也就是要接聽、拒接或轉接，並送出一個回應(response)給呼叫者，打電話出去的UA可以稱為User Agent Client(UAC)，接收電話的UA稱為User Agent Server(UAS)。

II. SIP Server部分:

Proxy server：當一個proxy server 收到一個請求(request)的訊息，在查詢位於register server 中的相關位置資訊後，就會將這個請求的訊息轉交給被呼叫者，然而若被呼叫者離它很遠時，它就會將訊息轉送給下一個proxy server，再行轉送。

Register server：它能幫助終端找出他們希望通話的夥伴，透過使用者登錄時的註冊，它能夠紀錄使用者目前真正的位址，等待有使用者發出邀請的訊息去呼叫另一個使用者時，便可以查詢位於資料庫中的位置資訊，而將訊息送達被呼叫者端，此外他也能提供由UA設定的使用者偏好資訊。

Redirect Server: Redirect Server是當一個UAC只知道UAS的某些資訊的時候，UAC會將知道的訊息傳送給Redirect Server，當接收到Request的時候，他會將去Register Server查詢，Request中相關的資訊，然後會將Request送回去給原來發出Request的UA，裡面會將在Register查道的詳細資料放入Request中，一但原來的UA收到之後，就直接在將Request再送出去，送到UAS去。

SIP URI：在SIP 協定中，SIP URI 是用來識別UA 的為唯一方式，UA 完成註冊後，網路上的其他使用者若欲與該使用者進行聯繫時，只要知道SIP URI 就能連線，其形式類似於目前的E-mail address：user@domain。

當有一個呼叫者欲邀請另一個被呼叫者加入會談時，會先由其中端上的UA 先送出一個request 訊息給離他最近的proxy server，待proxy server 查詢register server 後，便可將此一訊息轉送給給被呼叫者，或者是下一個proxy server 令其去轉送，如果被呼叫者同意連線，也回送一個response 訊息後便可建立連線，SIP 通訊的架構圖如圖所示。

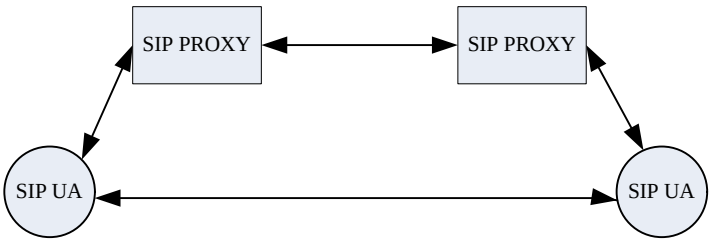


圖 2、SIP 對話架構

協定特點

◆ 簡單

只包括六個主要請求，六類回應；

基於內容編碼，易實現、易調變，便於追蹤和手動操作；

◆ 擴展性和彈性

具有靈活的擴展機制和強大的能力協商機制；

新的方法、標頭和功能增加，無須改變協定，網路簡單；

分散式體系結構提高了系統的靈活性和可靠性；

◆ 互通性

簡單、輕型協定，基於內容編碼方式，容易描述和分析；

應用層協定與底層傳輸無關。

2.2 Session Initiation Protocol (SIP)的安全威脅

SIP 不是一個容易提供安全防護的通訊協定，L.Qin,H.Xinhui,Z.Wei 在[6]文中，提出幾個典型的攻擊型態，讓我們容易的了解 SIP 對於安全性的需求：

I. Register Hijacking: 這是 SIP-Based 系統一種最簡單與最常用的攻擊方式，SIP 的 Register 機制允許一個 UA 去定義自己的所在位址，註冊的伺服器會將 REGISTER 訊息中的 FROM HEADER 儲存下來，當有 Request 進到註冊伺服器查詢某一個 UA 的所在位置時，TO HEADER 代表的就是要查詢的 UA 所在的位置，註冊伺服器會將 TO HEADER 的資訊到註冊的資料去查詢，當有不一樣的資料時，會將新的 TO HEADER 修改掉舊有 Request 中的 TO HEADER，但是在一個 REGISTER 訊息中的 FROM HEADER 可以隨意的被 UA 所修改，因此藉由這樣的性質等若是給有惡意的使用者輕易的破壞註冊伺服器的資料，一個攻擊者可以成功的仿造一個註冊訊息去改變這註冊的資料，舉例來說 :全部重新註冊目前某一個 UA 的所有的註冊資訊，因此所以有關於這個 UA 的需求都由原本的 UA 被重新導向到攻擊者所修改的位址去，這種攻擊告訴點出在 SIP 系統中需要去證明 Requst 需要經過認證。

II. Denial-of-service: 這個攻擊的重點在於讓網路中的某些節點無法使用，通常

是造成異常的網路流量使得節點無法處理提供正常的服務，而一個分散式的阻斷攻擊能讓一個網路使用者造成多個網路節點癱瘓，DoS(阻斷攻擊)與DDoS(分散式阻斷攻擊)會使得網路的節點的可用性以及資料保護失去效用。

III. Tearing Down Session: 這種攻擊模式會造成很大的安全風險，一個 Session 由一個 INVITE 訊息建立之後，SIP 會藉由一些後續的 Request 的傳送來維護通話的狀態，如何去防止攻擊者去仿造維護通話的訊息對於這樣的機制是相當關鍵的部份，最有效率的方式是去認證傳送過來的 BYE 訊息。

IV. Tempering with the Message Bodies: 攻擊者會藉由主動的去修改訊息的內容而獲得某些利益，這樣的攻擊不只會去竄改 Session Key，甚至可以去竄改貨者破壞訊息的內容在兩個的 SIP UA 間的傳送，舉例來說:就是攻擊者會去破壞 SIP 訊息中 SDP 的資料，或者是電話的訊號，也就是攻擊者可以將 RTP 的串流先經由攻擊者設定的位址在傳送到原本的目的地去，可以達成竊聽的效果，在 SIP-Based 的系統中不只是訊息內部甚至訊息的 HEADER 也會因為經過竄改而遭受某一程度的安全威脅，譬如：VIA 或者是 Request Routed 等等。訊息的完整性是對於這樣攻擊的最主要保護方式，如何確保訊息再傳送間的完整性對 SIP-Based 的系統扮演重要的腳色。

V. Impersonating a Server: 這也是常見的一種攻擊方式，攻擊者會造成一個遠端的伺服器，然後可以接收其他 UA 所傳送的 Request，而去竊取某些資訊，防止偽裝伺服器的主要方式，就是去認證伺服器本身再傳送訊息。

VI. Replaying Attack: 攻擊者去將竊聽來的訊息再重新傳送給其他的 UA，防止這種攻擊方式，主是世界由時間戳記以及 NONCE 去保證訊息的效益。

再上述的攻擊當中，我們可以統整出在一個使用 SIP 做為通訊協的系統當中，基本的安全需求包含:

I. 保證通訊時的完整性以及機密性，去防止重送攻擊以及訊息偽造。

II. 在通訊中提供認證與參予者的私密性，去防止阻斷攻擊。

III. 而 SIP 的訊息的內容需要提供私密性，完整性以及認證。

2.3 SIP 的安全機制與其限制與其在 AD HOC NETWORK 上的挑戰

這一節，我們根據[6][15] 來討論目前使用在 SIP 的安全機制並且分析這些機制在一個真實的 SIP 系統下的限制。

在目前 SIP 的安全架構使用從 HTTP 與 SMTP 延伸過來的安全模式：

I. Transport or network layer security: 傳輸層與網路層的安全是在訊號加密以用來保護訊息的機密性與完整性，常常用來來建立較低層級的安全，而也藉由這樣的機制在許多架構提供認證的功能，目前有兩個在在網路層以及傳輸層可以選擇的安全模式分別為 TLS 與 IPSec，而 SIP 也有許多安全的防護機制。

II. SIPS URI scheme: 就是用來傳送的資源安全的傳送，使用者傳送 SIPS 的訊息表示他已經選取某種的安全方式來安全的傳送資料。

III. HTTP Authentication: SIP:也沒有經過特別的修改，直接使用 HTTP Authentication 來提供 SIP 訊息防止 REPLY 攻擊與進行單向的認證。

IV. S/MIME: S/MIME 允許使用者可以在 SIP 中加密 MIME 的訊息內容，在不影響訊息 HEADER 的前提之下，提供訊息內容點對點的安全性，就如同成熟的認證機制，使用 S/MIME 對於 SIP HEADER 可以提供某種程度的完整性以及機密性，經由建立的 SIP 訊息通道。

雖然這些安全及制提供了 SIP 降低某種程度的風險，但是在使用這些架構也需要考慮依些限制[2][11] [12]：

I. 對於使用 S/MIME 的架構來說，最大的缺點在於缺乏一個對終端使用者普及的公開金鑰架構，特別在 MANET 這種活動性極強的環境，沒有公開的金鑰架構會面臨很大的挑戰。

II. 對於使用 TLS 的架構來說，最主要的非議來自於 TLS 只能架構在一個連線導向的通訊協定之上，無法使用 UDP 來做傳輸，這樣的缺點在 MANET 網

路上更會被注意，因為 MANET 網路的頻寬有限，所以，使用快速與不需額外頻寬使用的 UDP 是很重要的。

III. 而使用 SIPS 沒辦法保證 TLS 真的是由兩個端點來做傳輸。

IV. IPsec/IKE 使用 Pre-shared Key 或者是公開金鑰架構(PKI)可能會有一些問題，使用 Pre-shared Key 沒有辦法在一個公開的環境中獲得所有人的金鑰，因此有擴充性上的問題，PKI 很適合在一個公開的環境下與非特定的節點作金鑰交換，但是，在與非特定節點作金鑰交換的時候，需要經過一個認證中心(CA)，但是在 AD HOC 環境時，一個使用者可能會透過很多不同的裝置使用 SIP 系統，因此要一個使用者在每一個裝置中都存放自己的 Private Key，是不切實際的，在一個缺乏對方資訊的環境裡面，SIP 通常會在通話的雙方使用 Session Key，因為如果在訊息使用 Public Key 憑證(Public Key certificate)個人的資訊可能會有洩漏的潛在危險。

而且最重要的是 SIP 的訊息包含了許多具有意義的訊息在 HEADER 檔之上，並不僅止於 SIP 所要傳達的訊息，可能是訊息的來源以及訊息經過的節點，訊息通訊已經持續的時間，還有通話的參予者，這些部分都是極具有意義的，如何去保證這些訊息的完整性是一個重要的課題。

第二部份，我們討論在 AD HOC 這樣的網路架構下，SIP 架構可能需要面對的改變與挑戰[2][11]。

SIP 是一個架構在應用層級的路由機制上，但是因為在 MANET 網路上可能沒有辦法擁有固定的 SIP proxy，所以，SIP 可能沒有辦法像預設的架構般順利的運作，因此如何提出一個機制去對應接收訊息者的 SIP URI 與一般的 IP address 是很重要的，如何去組織一個應用層級的路由資訊來傳送訊息是在 MANET 上最重要的課題。

在一個 MANET 的架構下，一旦有一個節點想要去產生通話，可能需要去選擇某一個節點當作是 SIP proxy 或者是註冊伺服器(Registrar)[9]，或者是使用一種新廣播版本的 SIP 架構[9]，對於第一種方式來說，所有的節點需要去中央的 proxy

或者是 Registrar 註冊自己的資訊，可以讓其他的節點來查詢或者是獲得目前其他可以通訊的節點，但是，如何去處理這樣大量的組織架構，而且，在 MANET 上可能被選取為 proxy 或者是 Registrar 的節點會有消失或斷線的可能，如何持續去維持這樣的運作方式，是這種架構的關鍵[1]，在一種使用 SIP 廣播的架構下，可能只能夠在很小型的 MANET 上有良好的運作，一但較大型 MANET 網路使用那可能會造層網路頻寬的擁塞，所以大型的網路需要中央管理架構，而小型的可以使用廣播架構。

在這兩種架構下都有相同的安全與信任上的問題，使用者如何去確定回答這個訊息的人，是他所想要傳送的，還有如何去保證訊息傳遞在這種每一個節點都有可能竊聽與竄改的網路環境下是安全的，這是在 MANET 上很關鍵的問題。

2.4 MANET 上的 SIP 系統架構

在上面提出因為 SIP 在 MANET 上沒有辦法擁有固定的 SIP proxy，所以，如果要讓 SIP 能夠順利的在這樣的網路架構上運行，就需要不同於原本設計的架構，而目前也已經提出兩種分別基於廣播與群播方式的架構讓 SIP 可以在 MANET 上傳輸訊息，皆來要要依造這兩種被提出的架構分析其運作以及相關的安全議題。

2.4.1 A FRAMEWORK TO USE SIP IN AD-HOC NETWORK

這是 H.Khlifi,A.Agarywal,J.Gregoire 在 2003 年[9]提出的一種使用廣播方式來建立 SIP 通訊的架構，而這篇文章的主要目的就是提出這樣廣播的架構並且能夠順利的傳輸立即訊息(Instant Message)，這個架構允許使用者去發現 MANET 上的參予者，並且去建立，公開跟結束雙方或者是多方的通話，來提供 SIP 在 MENET 上的應用，為了去處理頻寬的限制，這篇文章提出獨特的網路路由通訊協定跟應用層的 SIP REGISTER 方法，

這個架構使用會議訊號(Conference signaling)的觀念[13]，可以視為是 SIP 應用的延伸，能讓使用者去發現其他使用 SIP 的節點，並且建立會議在 MANET 上

交換訊息，這樣的架構下不需要額外的設備，只需要去執行 SIP USER AGENT，而且傳統的 SIP proxy，Registrar 與 Redirect server 在這樣的架構下是不需要的。

I. Discovery

節點的發現，當一個 UA 一加入 MANET 之後，這個 UA 簡單的去廣播一個 REGISTER 的訊息通知其他節點自己的訊息，所有使用 SIP UA 的使用者都會開始一個 5060 的阜號，並且等待訊息經由這個阜號進入，當收到一個 REGISTER 訊息之後，會將這個訊息的相關資料儲存起來一段時間，每一個 UA 都會週期的傳送 REGISTER 來告訴其他節點，自己還存在這個 MANET 之上。

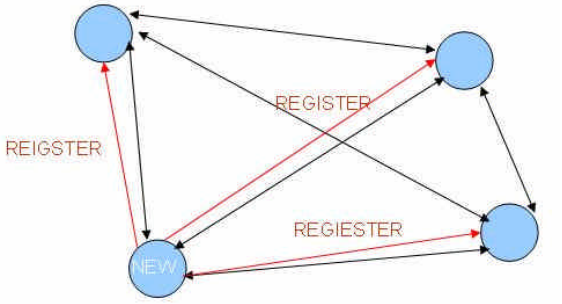


圖 3、註冊程序

II. Initiating a conference

這個架構可以建立兩種不同的會議，分別是私人會議與公開會議，私人會議是當一個 UA 發起時已經決定由哪幾個節點加入，其他的節點不能夠加入這個會議，而公開的會議則是任何節點都可以自由的加入，每一個發起這個會議的 UA 都會被當作是這個會議的領導者，而如果發起這個會議的 UA 斷線，則由加入順序第二的節點遞補，而一個發起會議的人會去廣播一個 REGISTER 訊息裡面會包完這篇 PAPER 自己定義的 HEADER 叫做 CONF-ID，這個 HEADER 是包含了一個 UA 的位址也就是目前這個會議的 CONF-ID，然後所有想要加入這個會議的其他 UA 可以經由傳送 INVATE 訊息給 CONF-ID 中的 UA 來表示願意加入會議。

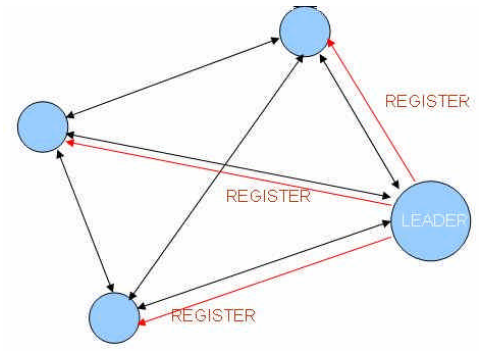


圖 4、會議建立程序

III. Joining a conference

每一個參與會議的 UA 都會建立起與其他 UA 的聯繫，然後會議的領導人會負責去宣傳這個會議以及接收新的聯信訊息，會議領導然人會定期的廣播參與名單給其他參與這個會議的 UA。

一個 UA 如我要去加入一個目前進行中的會議，這個 UA 會去傳送一個 INVITE 訊息給訊息的領導人，然後在訊息領導人接受新加入者之後，會傳送一個新的 CONF 訊息給其他所有加入這個節點的 UA，通知他們有新的參予者加入，然後這樣其他的會議參予者就會知道新加入 UA 的位址然後進行通訊。

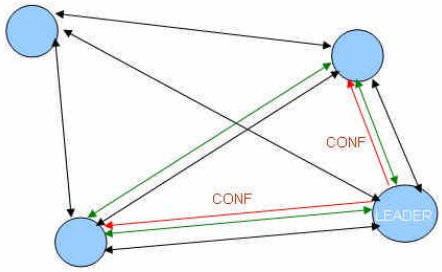


圖 5、加入會議程序

IV. Leaving a conference

當一個 UA 離開這個 CONFERENCE 的時候，就像所有的 NODE 傳送 BYE，然後所有人就會把送出 BYE 這 UA 從 CONF 檔中移除，如果是某一個 UA 斷線，沒有依據正常程序經由 BYE 就斷線的話，因為每一個 UA 都會定期會送出 REGISTER 去告訴其他 UA 他還在 AD HOC 線上，一旦 LEADER 有一段時間沒有收到 CONFERENCE 某一個 UA 的 REGISTER 他就會重送一份新的 CONF，裡面移除失聯的 UA，其他人就自動也移除該 UA。

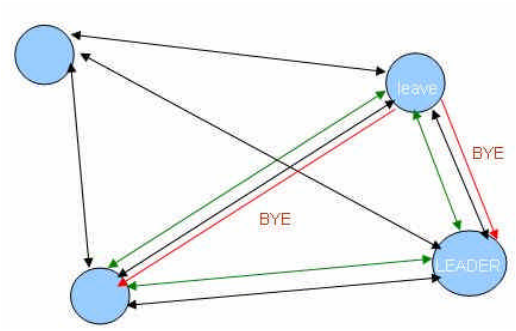


圖 6、結束程序

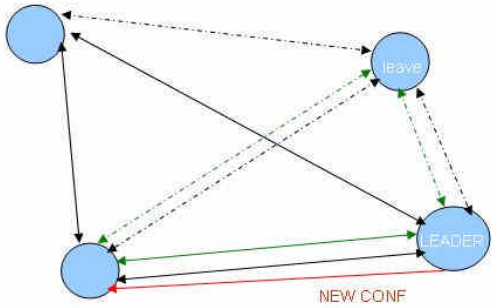


圖 7、異常結束程序

V. Failure if the leader of a conference

如果是 LEADER 發生斷線問題，如果 CONF 中 ORDER 第二 UA 有一段時間沒有收到 LEADER 的 REGISTER，他就會送出包含 REGISTER 包含 Conf-ID 的 PACKET 告訴大家，他會接手 CONFERENCE 變成 LEADER 然後再送出 CONF 給其他原本就在 CONFERENCE 的 UA，如果有同時兩個以上送出 REGISTER 就比較 CONF 中的 ORDER 比較早進入 CONFERENCE 為主，較晚的就停止發送 CONF.

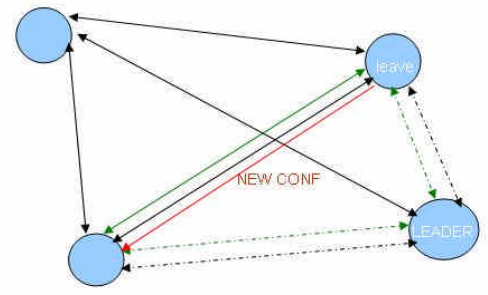


圖 8、LEADER 重建程序

這樣的架構下沒有考慮的任何的安全性，所以很可很輕易的就去取得一個 CONFERENCE 的資訊，甚至去做 DoS 或者偽裝不管是 LEADER 或者是一般 UA

做任何動作，而且每一個 UA 都會不斷的廣播自己的訊息，來通知其他 UA 自己還存在這個網路之上，這樣的架構對於頻寬有限的 MANET 是具有相當的挑戰性的，就如同在上一節有提到，這樣的架構只能在小型的 MANET 網路上使用，一旦節點的數量過多，會大量的提高訊息的延誤以及遺失，使用上具有許多限制。

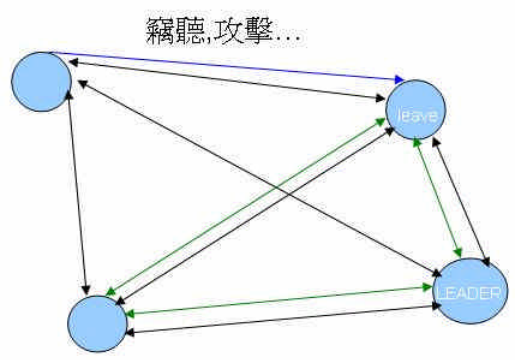


圖 9、安全疑慮圖

2.4.2 A SIP-Based Multicast FRAMEWORK IN MANET

A.Agarywal,Y.Yu 在 2005 年[1]提出一種 overlay multicast 的架構去處理 SIP 群播在 MANET 環境上的應用，這個架構中首先使用 SIP 訊息去發現其他的 SIP UA 然後先建立起一個 meshed overlay 網路，然後再依據需求建立起一個 overlay multicast 群聚結構，為了去處理網路頻寬的限制，這個架構中採用網狀的群聚架構(meshed clustering)，這個虛擬的應用層網狀網路拓樸是架構在網路層之上，用來更新與管理這種分散管理與廣播樹狀架構的變化資訊。

應用層級網狀網路(也可以稱作 Overlay Meshed)的廣播是建構在實體網路層之上，隱藏複雜的實體網路拓樸架構提供簡單明瞭的通訊架構，在這個應用層的網路會形成一個包含群播節點的邏輯性網路架構，下層資訊交換就透過網路層的通訊協定做 unicast 的資料交換。

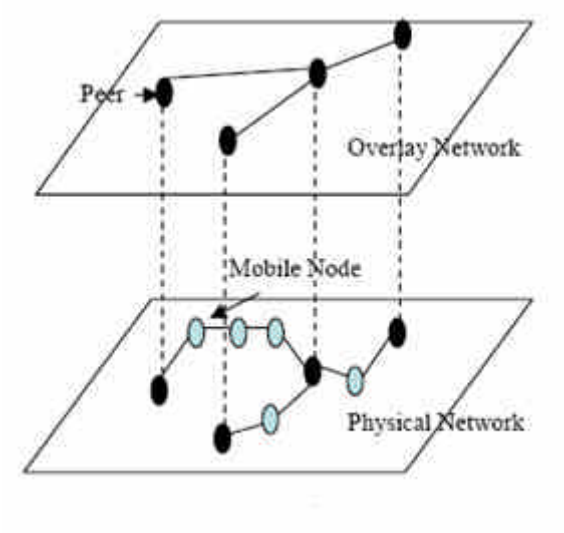


圖 10、Overlay Meshed 對照圖

在這篇研究中提出一種中介的架構，介於應用層與 MANET 的路由網路實體層級之間，如圖示，這個架構使用 SIP 的訊息去組織應用層的群播群體，用來在 MANET 上提供應用服務，這個架構中使用 SIP 先去組織一個網狀應用層的 MANET 拓樸，然後去產生應用層級的廣播群體，用來對應相對的 SIP 應用。

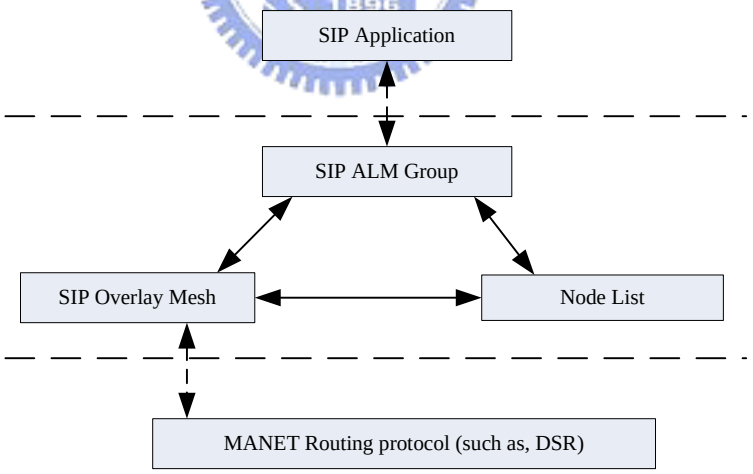


圖 11、在 MANET 上 SIP 群播架構

建立起一個有效率的中介軟體是這篇論文的目標之一，因此這篇論文採用網狀的群聚架構去達成這個目標。

許多提出來的網路群播協定，都是使用樹狀的架構來傳輸訊息，使用樹狀架構來傳輸資訊的確比起其他的傳輸方式更有效率，但是因為在 MANET 上無法去預測拓樸的改變與其改變的頻率，所以如何去維護一個群播的樹狀架構可以

說是艱難的任務。

群聚(Clustering)是一種將分散節點分為幾個不同的小群聚然後提供相互連結能力的架構，每一個群聚都有群聚的首領，負責來管理並且維護群聚內的名單，並且提供連結其他群聚的能力，群聚在通訊網路中常常用來處理分散計算，與樹狀架構相同的，群聚可以簡單的快速擴展，將節點分成小群聚並且每個群聚都會自我管理，而群聚也提供有效的群聚間的資訊交換不管是拓璞資訊或者是訊息的傳輸，因為每一個群聚的首領都會互相聯繫，它們可以交換彼此之間變動的訊息。

因為集中式的群聚需要比較少的訊號聯結，而且網狀的群聚有較高的錯誤容忍能力，考量到 MANET 上的機動性，所以這在這篇研究中作者就採用網狀的群聚架構，這種階層式的網路架構是用來減少路由資訊的傳輸量，這樣的架構中群聚的首領稱為通訊閘(GATEWAY)，其他的稱為非通訊閘節點(non-gateway node)，而其所提出的 SIP 系統中，通訊閘節點稱為 SIP 通訊閘(SIP GATEWAY)，而非通訊閘節點則稱為 SIP 非通訊閘節點(SIP non-gateway node)。

I. Thr Architecture of the Framework

SIP-ALM 為一應用層級的群體通訊中介軟體，並不需要倚靠網路架構的支援，因此能夠快速的佈署網路與簡化在設定期間的成本，它用來建立一個上層的邏輯性網路來讓節點能夠互相傳遞訊息，並且提供目前存在群播的團體資訊，可以提應用層網狀網路上的曾員加入使用提供的服務。

這個中介系統使用 SIP 訊息組織群播團體來提供在 MANET 上的應用，這個中介系統會隱藏分散架構而造層的網路不一致性，並且提供一制性的應用服務介面，這個架構不需要其他的額外需求，只需要網路上的節點(MN)使用這個中介軟體即可互相通訊。

這個 SIP-ALM 的中介軟體讓參予的節點(MN)可以自我組織與分散管理並且建立一個應用層級的網路拓樸架構，而參予的節點會藉由這個架構去得知網路拓樸的資訊，建立應用層的群播群體然後藉由這樣的群播的群體來互相傳輸應用程

式使用的資料，這樣的架構下將網路的節點分為，非 SIP 節點(Non-SIP node)，SIP 節點(SIP node)，SIP-ALM 節點(SIP-ALM node)。

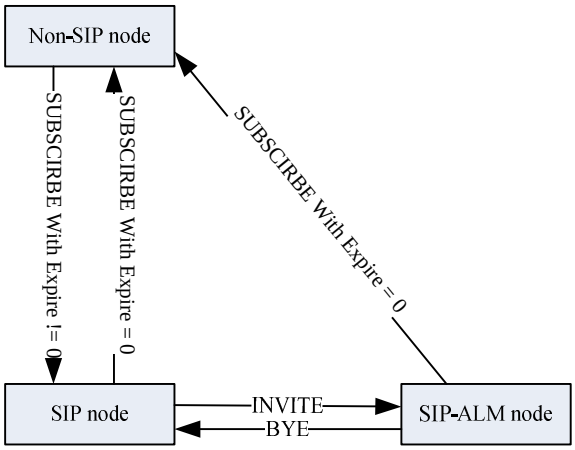


圖 12、節點分類圖

A. 非 SIP 節點(Non-SIP node)為一般的 MANET 上節點，就是沒有 SIP 功能的節點或者是尚未啟動 SIP 功能的節點。

B. SIP 節點(SIP node)就是開啟這個中介軟體的 MANET 節點，這個 SIP 節點會去創造或者是加入 SIP 應用層的網狀網路，但是尚未加入群播團體的節點。

C. SIP-ALM 節點(SIP-ALM node)就是已經加入 ALM 群播團體的 SIP 節點，這樣的可以透過 IP 的群播通話接收與傳送資料，此外，資料可以是只傳給鄰近的某一個節點，或者是藉由這樣的方式與所有的群體成員對話。

II. The Gateway Decision Process

SIP 的系統是架構在要求與回應的傳送模組之下(Request/Response)，在此研究中提出的 SIP 網狀網路(SIP Overlay Mesh)是先建立起 SIP Gateway 來接受與傳送要求訊息(Request)，每一個 SIP Gateway 會建立一個只有一個節點距離的群聚，每當有新的節點產生，可能會將原本的非 SIP Gateway 的節點變成 SIP Gateway。

當開始要建立 SIP 網狀拓樸時，每一個節點(MN)一開始將自己當作是 SIP Gateway，然後這個節點會去廣播一個訊息說要加入其他的現存的應用層網狀網

路，如果沒有收到任何的回應(response)，那麼這個節點就真的把自己當成 SIP Gateway 來運作去接受其他的要求訊息(Request)，如果節點收到其他節點的回應訊息，它會根據收到的那個回覆訊息(Response)將回應的那個節點當作是自己的 SIP Gateway，再將自己變成 Non SIP Gateway 節點，如果有兩個以上的回應時，就將第一個收到的訊息當作是 SIP Gateway。

這樣的 Gateway 決定程序，一但鄰近的節點改變時，在 SIP 的網狀拓樸架構便會啟動，每一個節點都會去決定自己當一個 SIP Gateway 或者是 Non SIP Gateway 節點。

III. The Peer Discovery Mechanism/Joining an Overlay Meshed Network

當有一個新的 SIP Mobile UA 要加入 SIP 的上層網狀網路(Overlay Meshed)時，它會簡單的廣播一個 SUBSCRIBE 的要求訊息出去，每一個目前已經存在在 SIP 上層網路的 UA 都會聆聽一個 5060 的阜號，用這個阜號來接收 SIP 訊息，如下圖示：

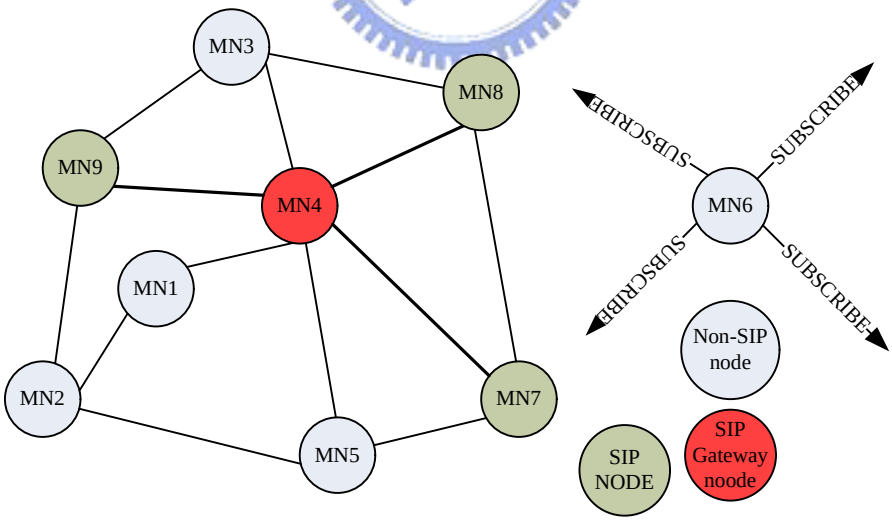


圖 13、Overlay Meshed 節點加入示意圖

在這個 MANET 網路上，MN8 與 MN7 是已經參加 SIP 上層網路的節點，並且聆聽在 5060 的阜號，MN6 想要加入 SIP 的網路，於是他廣播了一個 SUBSCRIBE 的要求訊息出去，所有的節點會去聆聽 5060 阜號，一旦有節點收到這個訊息的時候，會去詢問自己的 SIP Gateway 確認他是否有收到同樣的訊

息，如果沒有，這個節點會將這個新節點的訊息儲存在節點名單(Node List)中，之後這個接收到 SUBSCRIBE 訊息的節點，回傳送一個 NOTIFY 回去，這個 NOTIFY 的訊息包括了: 目前這個 SIP 網狀網路中所有的節點與目前所有已經啟動的 ALM 群體，而其他的節點也會收到更新的節點名單，於是就可以知道目前有新的節點加入這個網路。

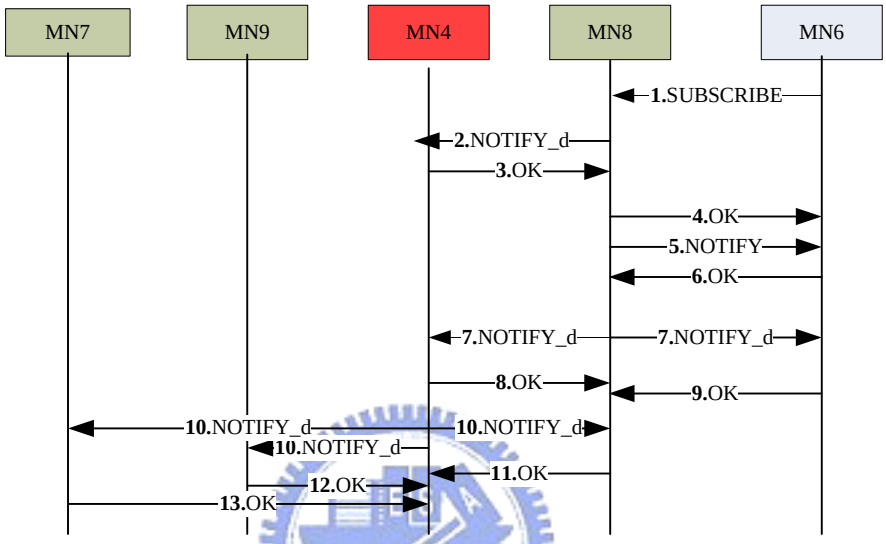


圖 14、Overlay Meshed 節點加入流程圖

上圖，是一個發現端點與決定 SIP Gateway 的流程圖，MN6 是一個新的節點，它先廣播一個 SUBSCRIBE 出去，然後 MN8 收到了 MN6 的廣播，MN8 會去詢問 MN4，MN4 是距離 MN8 一個節點距離的 SIP Gateway，詢問 MN4 是否有收到同樣的訊息，因為 MN4 的距離沒辦法接收到 MN6 的訊息，所以他告訴 MN8 他沒收到，而因此，MN8 就會變成 MN6 的 SIP Gateway，然後 MN8 會傳送一個 Notify 的訊息給 MN6，讓他知道者應用層的網狀拓樸網路目前的資訊，之後 MN8 會傳送 Notify_d 給 MN4，告訴 MN4，MN8 已經變成 SIP Gateway，而且，有一個新的節點 MN6 加入。

在 MN6 加入之後，MN8 變成 SIP Gateway，於是就形成了兩個群聚，分別是 SIP Gateway 為 MN4 與 MN8 的兩個群聚，根據此研究提出的方法，以後，MN6 如果要跟 MN7 做訊息的溝通，會經過 MN8 與 MN4 再到 MN7 去做通訊。

在這此研究提出來的方法中，每一個節點都會包含了這個網狀網路的節點資

訊也會包含目前已經建立的 ALM 群聚的資訊，這些目前存在的節點，路由資訊，以及以建立的 ALM 群聚，都是包含在 SIP Gateway 傳送過來的 Notify 裡面，根據收的 Notify 裡面的訊息，新加入的節點就會得到目前網路的訊息以及成為這個網路的一份子。

在這篇論文中提出一個訊息類型，週期性的 NOTIFY，這個訊息簡稱為 NOTIFY_d 裡面包含的訊息，就是如果網路的狀況有改變就會將改變的訊息放在 NOTIFY_d 裡面，只有在第一個 NOTIFY 裡面會有完整的網路資訊，在 NOTIFY_d 裡面只有放入改變的資料，也就是說，如果網路的狀況沒有改變，NOTIFY_d 裡面可能是沒有含有任何資訊的，那既然沒有改變為何還要傳送 NOTIFY_d，因為 SIP Gateway 會定期傳送這個訊息，然後依據收到的 Response 來判斷網路上的節點是否依舊存在。

IV. Initiating an Application Layer Multicast(ALM) group

SIP-ALM 使用集中控制的方式去維護群播團體的一致性以及效率，這樣的設計可以讓新節點加入 ALM 群體的時候，用來做服務執行允入控制(admission control)與減少系統的負荷，每一個 ALM 群體都有一個領導者，來負責維護這個 ALM 群體以及接受新的成員。

A. ALM group creation :

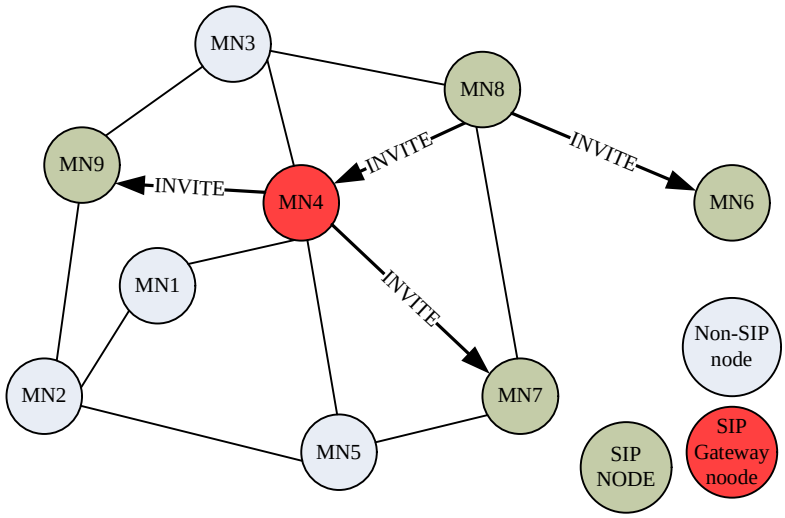


圖 15、群體建立圖

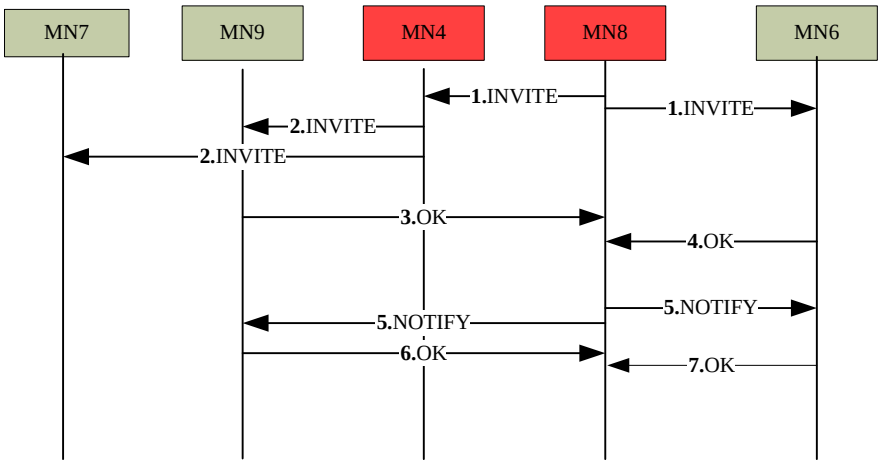


圖 16、ALM 群體建立流程圖

上圖所示，當一個應用層網狀的網路的成員要去建立一個 ALM 群體的時候，這個節點會向所有在這個虛擬網狀網路的成員傳送 INVITE 訊息，這個訊息裡面會包含一個這篇論文定義的 HEADER，Group-ID 這個 HEADER 的內容就是放置這個 ALM 群體領導人的位置，這樣一來所有人在這個網狀網路裡面的人就知道如果要加入這個 ALM 群體，要向哪一個節點傳送加入的訊息。

因為在 MANET 上面沒有中央的管理機制，所以要如何去產生一個唯一個的 Event-ID 難度是很高的，在 SIP 的通訊中，Call-ID 是獨立的，所以在這篇論文中，將 Call-ID 加入 Event-ID 裡面，使這個 Event-ID 不會重複，舉例來說 : Event-ID: 9334554@NCTU.EDU.TW IM，在這個 Event-ID 裡面，”9334554@NCTU.EDU.TW”是由 carmelo@NCTU.EDU.TW 傳送的產生 ALM 群體 INVITE 訊息的 Call-ID，而 IM 則是這個 ALM 群體所要提供的服務，而其他的 MN 可以依據這個 Event-ID 來回傳加入的請求。

SIP 的主要任務在於建立，維護以及結束通訊，在一個多方通話的環境裡面，在 MANET 上因為可能面對頻寬與傳輸的方式的限制，如何去有效的將每一個節點的訊息傳送給其他點是一個很大的問題，在此研究的 SIP 傳輸架構中，當有一個 Reqest 要傳輸給多個接收點時，只需要將每個要傳輸節點的 SIP URI 放在訊息 HEADER 裡面，這個 Request 會在 MANET 的應用

層的網狀網路上使用單點廣播的方式傳輸給所有的點，而不需要依靠網路層與實體層的訊息。

“Forward-To”這個 HEADER 就是用來完成上述群播功能的 HEADER，這個 HEADER 用來作群播多點接收 Request 的傳輸，在這個 HEADER 裡面會放置所有要傳送的 SIP URI，如果要傳送的節點需要經過 SIP Gateway，那麼這個節點就會使用”Forward-To”這個 HEADER 將要傳輸的資料放進去，因此藉由這樣的傳輸兩個節點間只需要傳輸一次，這麼一來可以減輕傳輸的負荷。

當一個節點要產生 ALM Group，這個 ALM 的領導人在傳送出 INVITE 之後，它會等待一段特定的時間 INVITE_TIME，來等待 OK 的回覆訊息，然後當等待時間結束之後，這個 ALM 的領導者會傳送一個 NOTIFY 的訊息給其他所有參予這個 ALM 的節點，當所有人都收到回傳 OK 回覆訊息之後，ALM 群體就建立完成。

B. Multicast tree creation :

在 MANET 上為了使群播群體有效率與具有可靠度，群播樹只在群播成員中使用單點廣播的方式建立，每一個節點都知道自己群體鄰近的節點以及需要透過 SIP Gateway 的節點，在這篇論文的架構中，每一個節點都會知道自己的網路拓樸，因此每一個來源的節點都會依據資訊去決定在這個虛擬應用層網狀群播樹。

如果發起的來源節點不是一個 SIP Gateway，它會將要傳送的群播資料送到一個節點距離的 SIP Gateway 去，然後 SIP Gateway 會去查詢自己的節點的名單，如果在同一個 SIP Gateway 下，SIP Gateway 會主動的去通知同一個群聚(Cluster)中的節點，如果在不同的群聚，SIP Gateway 會使用 Forward-To 的 HEADER 去交訊息傳送到下一個 SIP Gateway，依此類推完成群播動作。

舉例來說 : 在圖 14 中，當 MN4 要去群播一個 INVITE 邀請去建立一

個 ALM 群體時，因為 MN4 也為一 SIP Gateway 且 MN7，MN8，MN9 為只有一個點距來請屬於同一個群聚，所以，MN4 送 INVITE 到這三個點，然後會再傳送到 MN8 的訊息，再加上使用 Forward-To 的 HEADER 傳送到 MN8 之後，MN8 會再將訊息傳送給 MN6，因此，當 MN6 收到群播的 INVITE 訊息時，這個 Forward-To 的 HEADER 為 Forward-To: MN8@SIP.ADDRESS MN6@SIP.ADDRESS。

當一個節點接收到這個 INVITE 訊息時，如果他決定要加入這個 ALM 群體，它會回傳一個 200 OK 的回應訊息回去，當一個發出 INVITE 的節點收到 200 訊息時，它會知道這個節點加入其發起的 ALM 群體，然後再等待回收 Response 時間結束之後，這個發起 ALM 群體的領導者會再送出一份 NOTIFY 訊息給所有回復參與這個 ALM 群體的節點，完成 ALM 群體建立。

V. Maintaining an ALM group

在 ALM 群體當中每一個成員都會動態的加入，離開，或者是斷訊甚至會面對節點失敗，每一個群體中的成員會去負責追蹤群體的改變，便並且告知他人他發現的改變，一旦發現有改變，這個節點會去發送一個 NOTIFY_d 的訊息，告訴其成員目前有發生什麼改變，因此所有的群體成員能夠及時的了解目前群體中的即時狀況。

VI. Joining an ALM group

每一個節點會從收到的週期 NOTIFY 的訊息，ALM 群體的領導者會藉由這個訊息來維護這個 ALM 群體，當群體中有改變 ALM 群體領導者便會傳送這個訊息來告知其他節點改變的狀況。

如果想要加入一個目前已經建立的 ALM 群體，想要加入的節點傳送一個 INVITE 訊息給群體的領導者，在個 INVITE 的訊息裡面要包含一個 Event-ID 的訊息放在 Join 這個 HEADER 中，這個 HEADER 就是表示節點所要加入的群體是哪一個，舉例來說 : Join: netlab@example.nctu.edu.tw <IM>，其中 netlab@example.nctu.edu.tw <IM>，這個訊息就是這個 ALM 群體的 Event-ID。

一但群體的領導者收到這個訊息並且接受這個訊息，這個群體的領導者會傳送一個額外的 NOTIFY_d 的訊息給所有已經加入這個群體的成員，告訴其他的成員有關目前 ALM 群體的改變，這個訊息裡面只會包括目前新加入的這個節點，而其他成員一收到這個訊息，就會將手邊的成員列表做改變，所以藉由這樣的方式每一個節點都可以及時的維護節點名單。

以下圖為例，當 MN4 想要去加入目前已經成立的 ALM 群體，當 MN4 送出一個 INVITE 的訊息裡面包含一個 Join 傳送給 ALM 的領導者，MN8 在上圖中代表目前已經建立的 ALM 群體的領導者，根據 Join 與裡面的 Event-ID 可以判斷出哪個 ALM 群體是要加入的，並且傳送給 MN4 這個領導者。

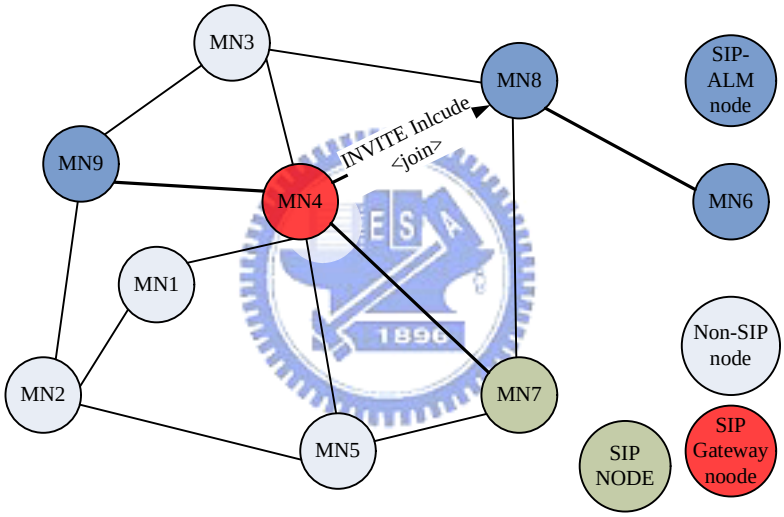


圖 17、ALM 群體節點加入圖

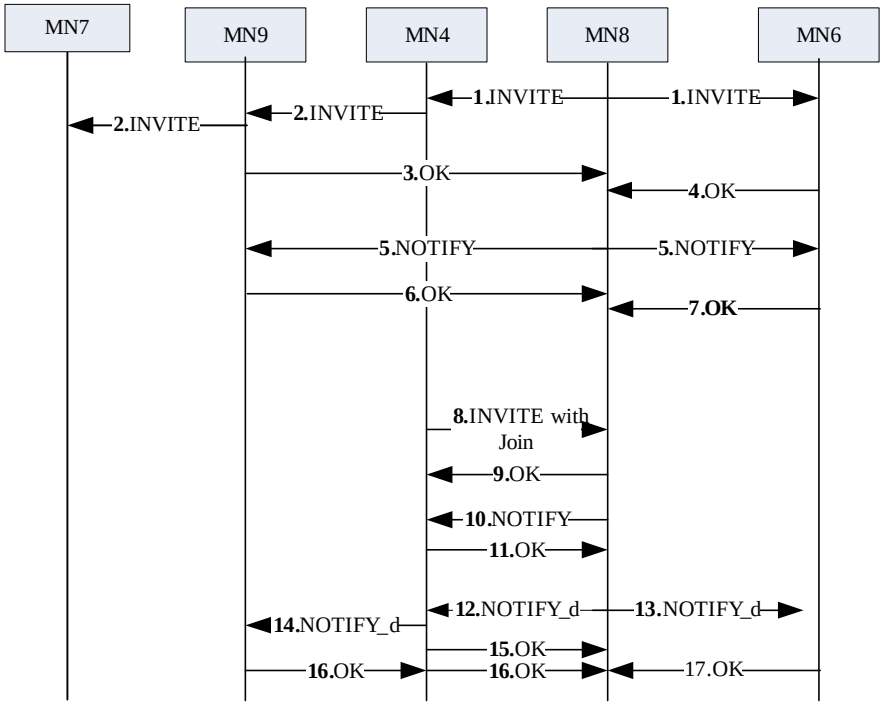


圖 18、ALM 節點建立與加入流程圖

Join 這個 HADER 就是被提來讓一個新的 SIP 通話，加入一個現存邏輯性的 SIP 通話，這樣的方式在點對點的通話控制環境中相當有效，因為這篇論文提出用 Event-ID 來代表不同的 ALM 群體，因此在 Join 的 HEADER 裡面放進 Evebt-ID 作為表示加入的群體。

因為 ALM 群體領導者要負責管控通訊建立，而 INVITE 的訊息就是傳送給訊息領導者，一旦 ALM 群體領導者收到這個 INVITE 的訊息可以依據之前建立的建立的通訊的控制政策，來判斷是否接受這個 INVITE 訊息，一旦這個 INVITE 被建立，那 ALM 的領導者會傳送一個 NOTIFY_d 訊息去告訴已經加入這個群體所有的成員，也包括新加入的成員，如果這個訊息不被接受，就傳送 481 Call/Transaction Does Not Existed 的訊息回去。

VII. Leaving an overlay meshed network

A. Graceful leaving

當一個網狀網路上的成員經過正常的程序離開這個網路，這篇論文稱為光榮的離開，當這個網狀網路上的節點想要去離開時，它會去告知這個網路上的節點，它將要離開這個網路，一旦接收到這個訊息，所有人就會將這個

節點從自己的節點名單中移除，在這篇論文的架構中，當一個網路要去光榮的離開時，會傳一個 SUBSCRIBE 的訊息給他的 SIP Gateway，裡面會包含一個 Expires 的 HEADER，而 HEADER 的內容為 0，當 SIP Gateway 收到這個訊息，便知道這個節點要離開這個應用層網狀網路，於是 SIP Gateway 再傳送 NOTIFY_d 給所有人，裡面為這個節點要離開的訊息，如此所有人便會將這個節點移除。

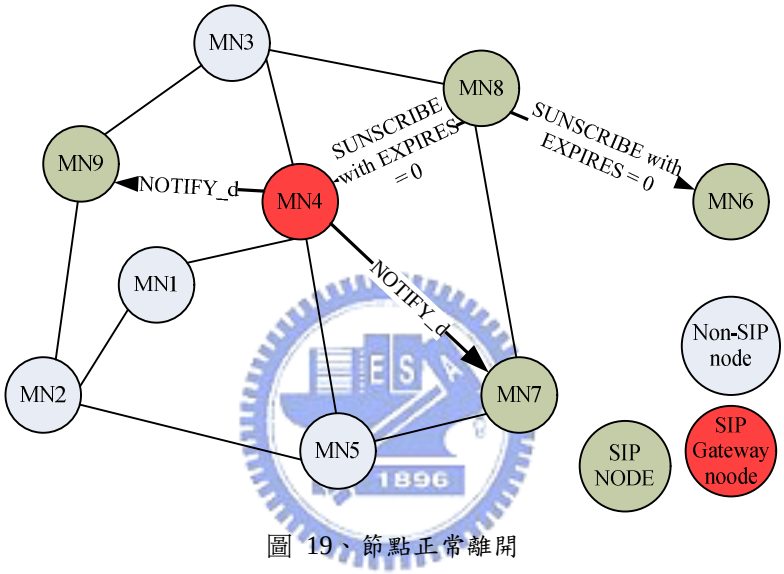


圖 19、節點正常離開

如上圖所示，當節點 8 決定要去離開這個網狀網路，因為它本身為 MN6 的 SIP Gateway，它會告訴他所屬的群聚裡面的成員他要離開，如果是一個非 SIP Gateway 節點，它會先傳給 SIP Gateway 再傳送，再藉由 SIP Gateway 告訴其他人，而身為 SIP Gateway 的 MN8 也會去轉告另一個 SIP Gateway，告訴有一個節點要離開，而另一個 SIP Gateway MN4 再去轉告其他節點，當 MN8 離開之後，節點 MN6 便須再重新尋找自己的 SIP Gateway 節點。

B. Detection of a brupt disconnection

當節點在 MANET 上移動的時候，很有可能會遇到網路斷訊的狀況，這篇論文提出一種監視網路狀況的機制，就是使用之前提到的 NOTIFY_d 訊息，NOTIFY_d 每一段時間都會傳送並且回傳 200 OK 的回應，透過這樣個機制，如果一個節點沒有去即時的回傳回應訊息，那 SIP Gateway

就會去判斷這個節點已經離開網路，如果 SIP Gateway 太長時間沒有傳送 NOTIFY_d 訊息，每一個節點就會去判斷可能 SIP Gateway 可能離開網路，會去重新組織目前新的 SIP Gateway。

VIII. Leaving an ALM group

當一個 SIP 的節點在網狀的網路上，可能會加入 ALM 群體，而每一個 ALM 的成員也可能會光榮的離開，也可能遇到網路狀況失敗而離開網路，而導致在 ALM 群體中離開，這兩種狀況分別為 ：

A. Leaving an ALM group gracefully

在這此研究中的架構中，當一個節點想要去離開 ALM 群體時，它會透過 Gateway 傳送 BYE 訊息給所有其他在這個 ALM 群體的節點，因此所有節點都會依據收到的這個訊息來知道這個節點要離開，並且將他從 ALM 群體的名單中移除。

B. Leaving an ALM group abruptly

對一個 MANET 節點來說，這個很能離開這個 ALM 而沒有傳送 BYE 的訊息，這個節點可能離開網路的範圍，或者無線網路的連結發生錯誤，都會造成離開 ALM 群體而沒有經過正常程序，當一個節點收到 NOTIFY_d 訊息，會去比對是否有離開的節點屬於自己的 ALM 群體，如果有，它會馬上傳送訊息告訴其他在這個群體上的參予者去移除這個節點，因此其他的群體成員可以及時的就將這個節點移除。

如果 MANET 上的節點發現自己遇到網路錯誤，它會重新發送 SUBSCRIBE 訊息去重新加入應用層的網狀網路，並且在加入之前參與的 ALM 群體。

IX. Conclusion

本研究就是依據這篇論文提出來的架構，並且依據其可能面對的安全問題以及風險，使用簽章以及金鑰交換的方式來去加強訊息的完整性與私密性，透過這樣的風險加強，評估效能以及安全性的差距，在第三章會根據這篇論文的架構討

論其安全上的風險，並且提出改進的方式。

2.5 應用橢圓曲線於數位簽章與 Diffie-Hellman 金鑰交換機制

橢圓曲線密碼系統(ECC)[18]近年來已被廣泛地制訂於國際標準。在相同的安全強度下，ECC 的密碼學參數可遠較諸如RSA 的其他公開金鑰密碼系統為小，這使得ECC 非常適合在行動裝置的有限資源環境如MANET下使用。

2.5.1 橢圓曲線密碼學

橢圓曲線密碼系統是由 Neil Koblitz(Koblitz, 1985)和 Victor Miller(Miller, 1985)兩位學者分別於 1985 年首先提出[4][10]，大多數的橢圓曲線密碼系統是在模 p 或 F_{2^n} 下運算。此密碼系統仍是存有 RSA 或 ElGamal 常見的弱點(e.g. 同模數攻擊、低指數攻擊)。RSA 與 ElGamal 系統中需要使用長度為 1024 位元的模數，才能達到足夠的安全等級，而 ECC 只需使用長度為 160 位元的模數即可，且傳送密文或簽章所需頻寬(bandwidth)較少，並已正式列入 IEEE 1363 標準。

ECC 類似於RSA，但並非某家公司的專利品，可以自由發展，也不受任何國家的規定管制。此外ECC 也沒有“adaptive chosen-message attack”。橢圓曲線密碼系統的安全性是建立於解橢圓曲線離散對數問題之困難度，來保障系統的安全性。橢圓曲線密碼系統其最大優點為可用較其他系統短的位元數，例如橢圓曲線密碼系統金鑰長度為160 位元，其他著名的系統如RSA 用的金鑰長度為1024 位元，二者的安全度是相等的，因此在相同的安全強度下，ECC 系統速度比RSA 系統快上數倍，同時可節約金鑰儲存空間，它們之間金鑰長度的比較見下表。

表 1、破解比較表

破解時間 MIPS(年)	RSA/DSA 金鑰長度	ECC 金鑰長度	RSA/ECC 金鑰長度比
10^8	768	132	6 : 1
10^{11}	1024	160	7 : 1
10^{20}	2048	210	10 : 1
10^{78}	21000	600	35 : 1

在相同的安全強度下，ECC的金鑰長度與RSA 的金鑰長度比較，如下表所示。從中可見ECC 的金鑰長度或數位簽章的長度遠比RSA小。隨著加解密演算法由DES/Triple-DES 改進為AES，且128 位元金鑰到256 位元金鑰AES 的安全性為2^{128}至2^{256}，相同安全性的RSA金鑰長度需3072位元到15360位元，但ECC僅需256位元到512 位元。若RSA 或ECC 是用做金鑰交換來保護256 位元金鑰的AES 時，RSA 應該用15360 位元的公開金鑰，而對應的ECC僅需使用512位元金鑰。所以無論從增快執行速度或節省空間的角度，我們可見ECC是優於RSA。

表 2、安全性比較表

安全性 / 演算法	2^{80}	2^{112}	2^{128}	2^{192}	2^{256}
RSA 長度 (Bytes)	1024	2048	3072	7680	15360
ECC 長度 (Bytes)	161	224	256	384	512
金鑰長度比	6 : 1	9 : 1	12 : 1	20 : 1	30 :1

橢圓曲線的通用格式為 $y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6$。用於密碼學技術的橢圓曲線是由滿足上述方程式的所有點(x, y)及一個無限遠點(point at infinity) O所形成的集合，其中座標x與y屬於某個有限體(finite field)。

橢圓曲線所分佈之有限體 (finite field)，為$GF(q)$ ，其中$q = p^m$ 為質數的次冪，m 為正整數。橢圓曲線方程式之型式為$y^2 = x^3 + ax + b \bmod p$ 與$y^2 + xy = x^3 + ax + b \bmod 2^m$ 兩種，但這都是已經經過許多化簡上述通用方程式之後所到的結果。

密碼學的橢圓曲線是由滿足該方程式的所有點(x, y)及一個無限遠點(Point at infinity)O 所形成的集合，座標x 與y 屬於某個有限體(finite field)。目前軟硬體具體實現的有限體為質數體(Prime field，$GF(p)$)、二元體(Binary field，$GF(2_n)$)、最佳擴展體(Optimal extension field，$GF(p_n)$)等三種(Bailey and Paar, 2001)。

橢圓曲線上的點可進行兩點間之加法(Menezes, 1993; Silverman and Tate, 1992)。幾何上，如果要計算相異兩點P 與Q 的和，則先找出通過這兩點的直線，然後找出這條直線與橢圓曲線相交的第三點$(-R)$，再將此點對x 軸做鏡射得到和(R)，如圖一所示。如果橢圓曲線上的某兩點共線的話，兩點相加之和就是O。

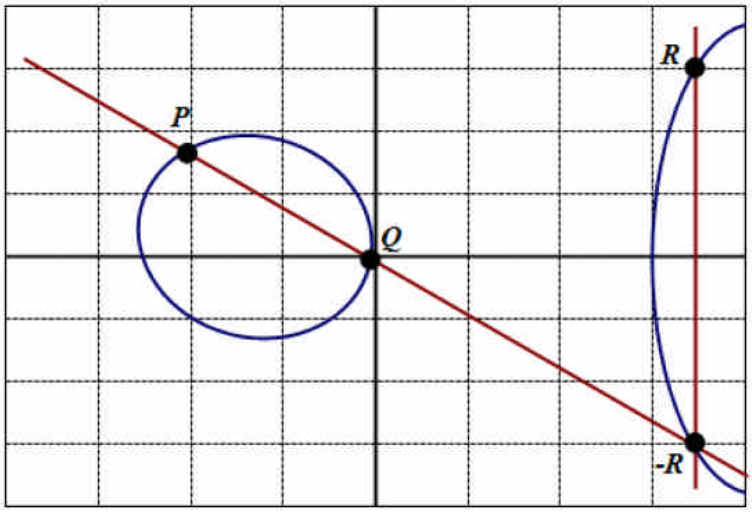


圖 20、橢圓曲線圖

若$P = (x_1, y_1)$與$Q = (x_2, y_2)$為橢圓曲線上的任意兩點，而$P \neq O \neq Q$，且選取質數體，此時橢圓曲線方程式為 ：

$$y^2 = x^3 + ax + b \qquad (1)$$

兩點加法的運算規則如下所示：

1. $P + O = O + P = P$
2. $P + (-P) = (X_1 + Y_2) + (X_1 + (-Y_2)) = O$

3. $P + Q = R = (X_3, Y_3)$

$$X_3 = \lambda^2 - X_1 - X_2，Y_3 = \lambda(X_1 - X_3) - Y_1，\lambda = \begin{cases} Y_2 - Y_1 / X_2 - X_1 & \text{if } P \neq Q \\ 3X_1^2 + a / 2Y_1 & \text{if } P = Q \end{cases}$$

如果選擇二元體，則橢圓曲線方程式為：

$$Y^2 + XY = X^3 + aX^2 + b \qquad (2)$$

而上述公式(3)的加法規則3 必須改為：

$$P + Q = (X_3, Y_3)$$

$$X3 = \begin{cases} \lambda^2 + \lambda + X_1 + X_2 + a & \text{if } P \neq Q \\ \lambda^2 + \lambda + a & \text{if } P = Q \end{cases}$$

$$Y_3 = \lambda (X_1 + X_3) + X_3 + Y_1$$

$$\lambda = \begin{cases} Y_{2+} Y_1 / X_{2+} X_1 & \text{if } P \neq Q \\ X_1 + X_1 / Y_1 & \text{if } P = Q \end{cases}$$

上下兩個公式的計算(加法、減法、乘法、除法/反元素)必須在相關的有限體進行，若選取質數體時僅需進行模算術(Modular arithmetic)，若選取二元體則需進行多項式算術(Polynomial arithmetic)。點乘法計算$k \cdot P$ 為橢圓曲線密碼系統的基礎，其中k 為正整數，而P 為橢圓曲線上的一個點：

一共K個P

$$K \cdot P = P \cdot P \cdot P \cdot P \cdot P \ldots \cdot P$$

如果$n \cdot P = O$ 則n 為點P 的級數(order)。在合適的橢圓曲線上，可以找到一個級數$n > 2^{160}$ 的基點(Base point) G，而此橢圓曲線系統參數基點G 可公開；另隨機選取小於n 的正整數d 當作私密金鑰計算$Q = d \cdot G$ 為對應的公開金鑰。點乘法的計算如果直接做k 個點相加，則需要執行k-1 次加法運算，效率不佳，目前已有許多可以加速點乘法計算的演算法。

橢圓曲線密碼系統的實現必須考慮下列因素：

1. 有限體的選擇
2. 橢圓曲線的挑選
3. 有限體元素的運算(加法、減法、乘法、除法/反元素)
4. 橢圓曲線點的運算(加法、減法、乘法)

2.5.2 橢圓曲線的 Diffie-Hellman 金鑰交換法

Diffie-Hellman 公開金鑰分配協定使曾謀面的兩個人，透過公開通道獲得他們兩人共同金鑰。例Alice 與John 欲共同使用一對話金鑰，首先他們先選擇同一條的橢圓曲線，設基點為P， K_A及K_J分別為他們的私鑰，計算以下對應的公開點:

$$R_A = K_A P$$

$$R_J = K_J P$$

1. Alice 傳送R_A 給John; John 送R_J 給Alice。
2. 此對應的對話金鑰為 $R_{AJ} = X_A R_J = X_R J_A = X_A X_J P$。

一旦此對話金鑰建立，之後兩人之通訊可使用其他對稱式密碼系統進行更進一步之通訊。

接下來使用清楚一點的算式來說明橢圓曲線如何應用在Diffie-Hellman的金鑰交換法(IEEE 1363)當中，首先選取一個很大的整數q(這個整數如果不是質數P的話，就是一個2^M的整數)，然後再選取a、b兩個橢圓曲線的參數，不管是選用質數體(1)或者是二元體公式(2)皆可，這樣就可以定義出橢圓曲線$E_q(a,b)$，接下來從$E_q(a,b)$選取一個積點$G = (X_1 , Y_1)$，這個G的級數是一個非常大的n，所謂橢圓曲線上的點G的級數n，就是最小的正整數n，使得 $nG = O$。$E_q(a,b)$與G是所有密碼系統的參予者都知道的參數。

使用者A和B用這樣的方式做金鑰交換

1.A選擇一個小於n的整數n_A，只為A的私密金鑰。然後A產生一公開金鑰

$P_A = n_A \cdot G$；此公開金鑰為$E_q(a,b)$上的一點。

2.B同樣選擇一私密金鑰n_B，並且計算公開金鑰P_B。

3.A產生密鑰 $K_{AB} = n_A \cdot P_B$。B產生密鑰$K_{BA} = n_B \cdot P_A$。

4.最後$K_{AB}=K_{BA}$因為:

$K_{AB} = n_A \cdot P_B = n_A \cdot (n_B \cdot G) = n_B \cdot (n_A \cdot G) = n_B \cdot P_A = K_{BA}$

如果想要破解這個機制，攻擊者必須在給定G以及kG的情況下求出k，但是這是非常困難的，舉例來說: 取 $p = 211$、$E_P(0,-4)$(橢圓曲線為$Y^2 = X^3 - 4$)，$G = (2,2)$，如此可以算出$240G = O$。A的私密金鑰$n_A = 121$，所以A的公開金鑰$P_A = 121(2 , 2) = (115 , 48)$。B的私密金鑰$n_B = 203$，所以B的公開金鑰$PB = 203(2 , 2) = (130 , 203)$，其共享金鑰為$121(130 , 203) = 203(115 , 48) = (161,69)$。

2.5.3 橢圓曲線的加解密

目前已有多種橢圓曲線的加解密法[4] [14] [16][17]，這邊以最簡單的一種為例，首先系統將送出去的明文 m 編碼成 x-y 形式的點 P_m，點 P_m 會被加密成為密文，並且在解密，我們不能單純的將訊息編碼成某個點的 x 或 y 座標，因為並不是所有的這類座標都會在 $E_q(a,b)$，因為編碼的方式有很多這邊不多加討論，目前已經有許多技術可供選用。

在一個橢圓曲線的金鑰交換系統當中，其需加解密系統需要兩個參數: G 與 $E_q(a,b)$，使用者 A 選擇一私密金鑰 n_A，然後再產生一公開金鑰 $P_A = n_A \cdot G$，同理 B 也選擇一私密金鑰 n_B，然後產生一公開金鑰 $P_B = n_B \cdot G$。為了將加密後的訊息 Pm 傳送給 B，A 選擇一個隨機的正整數 k，並且產生一個由兩個點所組成的密文 Cm

$$C_m = \{kG，P_m+kP_B\}$$

其中 A 所使用的為 B 的公開金鑰 P_B。為了解開祕文，B 用自己的私密金鑰乘上第一個點，再用第二個點減去其結果可得:

$$P_m + kP_B - n_B(kG) = P_m - k(n_B G) - n_B(kG) = P_m$$

A藉由加上kP_B來遮蔽訊息P_m，除了A之外沒有人知道k的值，所以即使P_B是公開金鑰，也沒有人能夠移除遮蔽物kP_B。而A在訊息中加入了移除遮蔽物的線索，如果有人知道的私密金鑰n_B的話，就可以移除kP_B，給定G與kG，攻擊者必須計算出k才可以破解，但這是很困難的。

舉一個例子，取p=751，$E_P(-1, 188)$，以及G = (0 , 376)。假設A要傳送一個橢圓曲線P_m =(562 , 201)所編碼的訊息給B，A隨機選取一個k = 386。B的公開金鑰 P_B= (201 , 5)。因此可以得到386(0,376) = (676 , 558)，而且(562 , 201) + 386(201 , 5) = (385 , 328)，因此A傳送的密文為{(676 , 558) , (385 , 328)}。

橢圓曲線的安全性取決於下列計算的困難度，在給定Kp與P求出K。這被稱為橢圓曲線的對數問題，目前已知解決對數問題最快的方法為Pollard rho法，下列表格可以清楚的看出RSA與ECC破解法效能上的差異。

表 3、使用 Pollard rho 法來破解橢圓曲線表

鑰匙長度	MIPS-年
150	$3.8*10^{10}$
205	$7.1*10^{18}$
234	$1.6*10^{28}$

表 4、破解 RSA 質數分解時間表

鑰匙長度	MIPS-年
512	$3*10^{4}$
768	$2*10^{8}$
1024	$3*10^{11}$
1280	$1*10^{14}$
1536	$3*10^{16}$
2048	$3*10^{26}$

由上表可清楚的發現，相對於RSA，橢圓曲線只需要相當短的鑰匙，就可以達到相同或者是更佳的安全性，並且可以節省傳輸之訊息的長度，可以節省網路頻寬，也因為較小的位元數可以獲得較快的效果，也可以節省計算時所需電力的消耗。

2.5.4 橢圓曲線數位簽章演算法(ECDSA)

訊息摘要主要是用以擷取一數位資訊之特徵，目前是採用密碼學中單向雜湊演算法 (One-way hash algorithm)，如MD5 與SHA-1 等演算法，來產生安全訊息摘要，用以產生資料之“數位指紋(Digital fingerprints)”，常常配合數位簽章使用，以達成資料之完整性。

數位簽章(digital signature)是指使用數學演算法(或稱雜湊函數)將電子文件轉化為固定長度之數位資料(訊息摘要)，並用簽署者之私鑰對其加密形成一簽

體，使任何可藉未轉化前之原始資料訊息、簽體及私鑰相關連之公鑰，驗證該簽體是否使用與簽章公鑰相對應之製作，以及簽製作後，原始資料訊息是遭受竄改。數位簽章是指以「非對稱型」密碼技術製作的電子簽章。生常生活中，契約、支票等書面資料必須以簽名蓋章，重要之數位資訊也有數位簽章的需求，而數位簽章是伴隨著數位文件連帶產生的。當數位文件嵌入某人的簽名時，這份文件的真實性和其簽名的真實性是要能夠被驗證出來的。數位簽章的法定效力將等於一般的手簽名，使用數位簽署系統更得靠單向赫序函數才能抵抗各類演算法的攻擊。

數位簽章是將訊息以單向雜湊演算法取得訊息摘要後再將此訊息摘要透過使用者個人專屬之密鑰 (private key) 來加以作簽章處理，再發送給相關之接收方，此處理後之資訊具有以下三項特性：

一、 第三者能以該使用者公鑰(Public key)驗證使用者之身份。

二、 接收方無法偽造或修改該簽章之相關資訊內容。

三、 簽章者不能於事後否認對該資訊之簽章。

下面描述橢圓曲線數位簽章演算法 (Elliptic Curve Digital Signature Algorithm，ECDSA)，在ECDSA[8][19][23][24] 相關標準ANSI X9.62(ANSI, 1998)與FIPS 186-2(NIST, 2001)中提及對於訊息m 的數位簽章(r, s)產生步驟如下:

假設G是橢圓曲線系統基點且其級數為n，正整數d為簽署者的私密金鑰，而$Q = d \bullet G$則為簽署者的對應公開金鑰，$h(m)$為訊息m的雜湊函數值。在ECDSA標準中，對應於訊息m的簽章(r, s)產生步驟如下:

1. 挑選一亂數 k， $n - 1 \geqq k \geqq 1$

2. 計算 $k \bullet G = (x_1, y_1)$ 且 $r = x_1 \bmod n$.

   如果 $r = 0$，則回到步驟1

3. 計算$s = k^{-1} \{h(m) + dr\} \bmod n$

4. 如果$s = 0$，則回到步驟1。

ECDSA 簽章檢驗的步驟如下:

1. 計算$w = s^{-1} \bmod n$

2. 計算$u_1 = h(m)\, w \bmod n$ 與 $u_2 = r\, w \bmod n$.

3. 計算$u_1 \bullet G + u_2 \bullet Q = (x_0, y_0)$ 與 $v = x_0 \mod n$.

若且唯若$v = r$，則簽章正確。

ECDSA 簽章產生時至少須進行一次點乘法以及一些模算術，簽章檢驗時則進行兩次點乘法以及一些模算術。表二為 RSA 與ECDSA 用於數位簽章的比較，從表二可見相同安全性時，兩種演算法所產生出來的簽章長度差異甚大，這將影響簽章傳遞的時間與儲存的空間。

表 5、RSA 與 ECDSA 比較表

演算法	RSA	ECDSA
簽章長度	安全性2^{128}:384 Bytes 安全性2^{192}:960 Bytes 安全性2^{256}:1920 Bytes	安全性2^{128}:64 Bytes(質數體) 安全性2^{192}:96 Bytes(質數體) 安全性2^{256}:132 Bytes(質數體)
安全基礎	大數分解	橢圓曲線離散對數
優點	歷史悠久，容易說明亦可同時用以加解密	計算速度快，簽章長度小，傳輸快速
區點	數度慢，簽章長度較大	理論不易理解，實現技術較複雜

第三章、在 MANET 上以 SIP 為基礎的安全群播協定

本章將根據[1]中的架構，依據第二章當中分析SIP所可能面臨的安全威脅，與SIP在MANET上所需要的安全需求，提出一個使用數位簽章以及金鑰交換法來解決其安全疑慮之流程，並且使用橢圓曲線來實作數位簽章與金鑰交換機制，最後提出一個新的訊息交換協定。

3.1 數位簽章

當訊息在MANET上進行訊息傳遞時，其他的節點能夠很容易的就擷取到訊息並且竄改之，如何在訊息傳遞的過程中保持訊息的完整性，是在MANET上通訊時很重要的工作，本論文的協定中提出使用橢圓曲線數位簽章來保證訊息傳遞時，訊息完整性相關的安全議題。

3.1.1 Overlay Meshed Network上的安全威脅

第二章中提到在新節點進入網路時，會先傳送一個SUBSCRIBE的訊息去尋找網路上目前所建立起的SIP Gateway，如果目前MANET網路上已經存在SIP Gateway，存在的SIP Gateway就會發送NOTIFY訊息回去，這樣新節點就會完成新節點加入的程序，新的節點也會由NOTIFY訊息的內容得知目前網路的狀況以及目前啟動的ALM群體服務，所以當傳送NOTIFY訊息如果遭到竄改，那新節點所獲得的訊息，將會是錯誤的網路狀態與服務，所以如何維持NOTIFY訊息的完整性是極其重要的。

以第二章中描述新節點加入為例，分析其可能會面臨的安全漏洞，節點的分布圖如圖21。

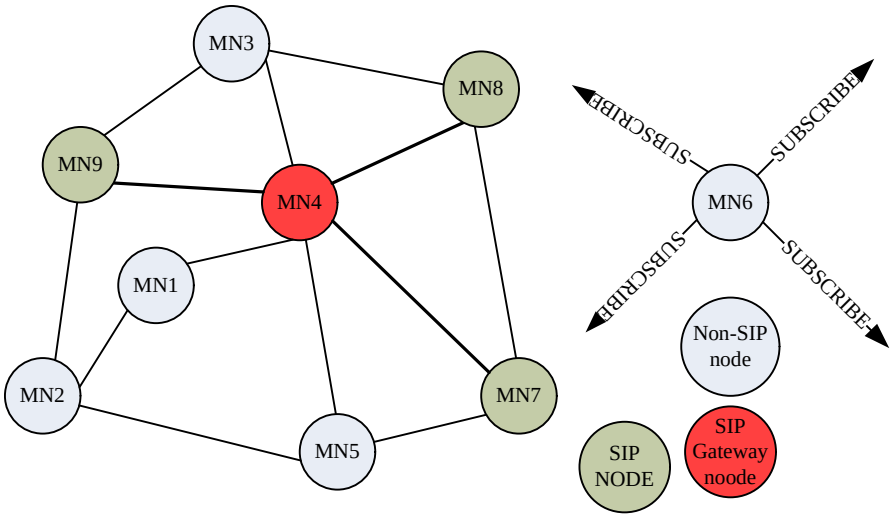


圖 21、節點分布圖

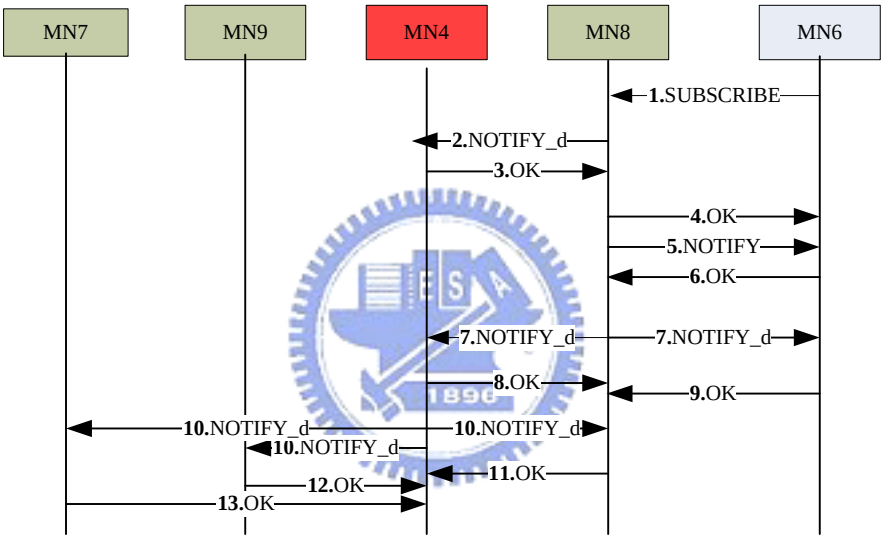


圖 22、加入 Overlay Meshed 網路流程圖

MN6是一個新加入的節點，MN8是目前已經存在於應用層級的網狀網路的一個節點，MN4為目前建立的SIP Gateway，其加入的流程如下:

Step 1. MN6向外廣播一個SUBSCRIBE，詢問是否有應用層的網狀網路存在。

Step 2. MN8向MN4詢問是否有收到MN6所廣播的SUBSCRIBE訊息。

Step 3. MN4會回到MN8是否自己有收到這份資訊，在本例中MN4並沒有收到MN6的資訊，所以回傳的OK訊息中告訴MN8自己並無收到。

Step 4. MN8回傳OK訊息給MN6告知其目前已存在應用層網狀網路，並且接受他的加入，MN6便將MN8設為自己的SIP Gateway。

Step 5. MN8傳送NOTIFY訊息給MN6，NOTIFY訊息包含目前網狀網路上存在的

節點與節點的分布情況，還有目前已經存在的ALM群體提供哪些服務。

Step 6. MN6回覆給MN8，已接收到NOTIFY訊息。

Step 7. MN8告訴MN4與MN6，MN8成為SIP Gateway以及新的節點MN6加入到應用層的網狀網路上。

Step 8,9. MN4與MN6回覆MN8以收到訊息。

Step 10. MN4傳送NOTIFY_d訊息給子節點，通知她們網路的狀況有改變，需要更新節點名單。

Step 11,12,13. MN7,MN8,MN9回傳訊息，完成新節點加入的程序。

在上面的流程圖中，在Step 5中SIP Gateway(MN8)送出一個NOTIFY訊息告訴MN6目前應用層網狀網路的資訊，這個時候如果經過竄改，那MN6就會使用錯誤的節點名單，一直向錯誤的節點送資料，在 Step 7中MN8會發出NOTIFY_d告知他原本SIP Gateway，網路的變化，所以，這個訊息需要確保沒有經過竄改，否則收到的節點，會對網路的狀況產生誤判，同理Step 10在MN4送出NOTIFY_d如果經過竄改也會產生問題，所以要保證NOTIFY的完整是非常重要的。

3.1.2 在MANET上SIP的數位簽章協定

一個節點在MANET上首先會廣播SUBSCRIBE去詢問是否有存在的SIP Gateway，如果已經存在一個SIP Gateway他便會加入這個節點，如果沒有，這個節點便會成為SIP Gateway，因此在加入已經存在的SIP Gateway時，便會產生訊息在傳送時是否被竄改的安全疑慮。

為了防止訊息經過竄改或偽造，在本研究中，提出使用數位簽章的方式來增加訊息傳遞時的安全性，當一個節點成為SIP Gateway之後，首先要決定橢圓曲線的參數$E_q(a,b)$與G，產生出一條橢圓曲線，然後再產生自己數位簽章的Public key，當有一個新的節點送出SUBSCRIBE時，使用一個<Public key>與<ECC>的HEADER，將橢圓曲線的參數包括p、a、b、G放入<ECC>(在本研究中為一個ECCParameterSpec的Object)裡面並請將SIP Gateway產生的Public key放入<Public

key>的HEADER裡面加上簽章傳送給新加入的節點，之後SIP Gateway會傳送NOTIFY訊息告訴新節點目前網路的狀態，新的節點就可以使用SIP Gateway的Public key與已知的橢圓曲線，來做簽章的認證，如此可以保證接收到的網路訊息沒有經過竄改，然後，新的節點在傳送回去的OK回應訊息裡面，將自己的Public key放進<Public key>的HEADER裡面，之後SIP Gateway會再傳送NOTIFY_d給所有目前存在這個應用層網狀網路的成員，裡面會有新節點的訊息，以及這個節點橢圓曲線簽章的Public key，這個時候新節點可以比對收到的NOTIFY_d裡面自己的Public key是否正確，如果有錯，節點重新發出SUBSCRIBE訊息重新進行加入網路的動作，如果無誤之後新節點的Public key，可以拿來驗證這個節點的訊息是否經過竄改，或者是否經過偽造。

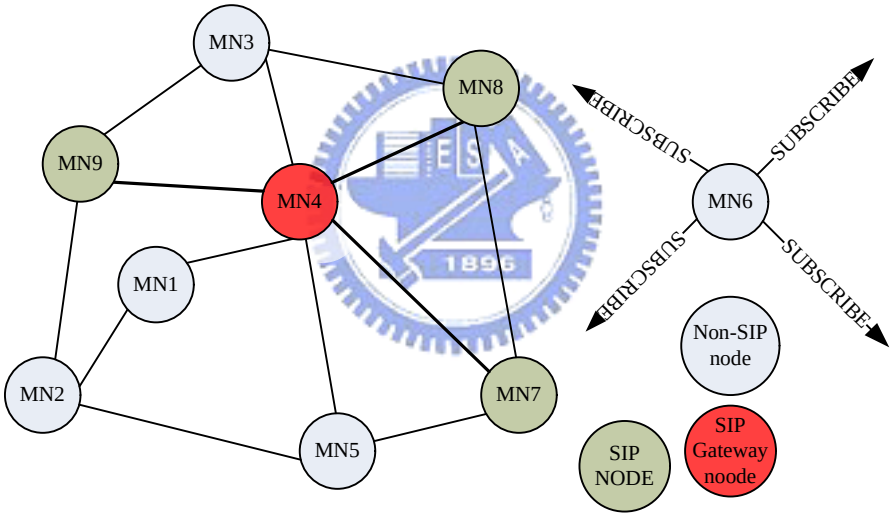


圖 23、實驗節點分布圖

使用與3.1.1相同的網路節點分布結構(圖23)來做比較，本論文所提出的訊息交換流程如下:

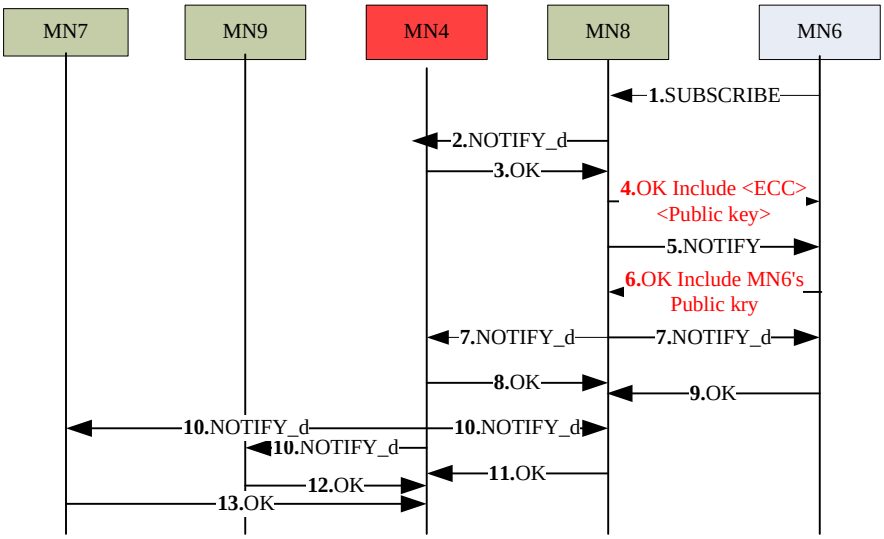


圖 24、加入數位簽章節點流程圖

Step 1. MN6向外廣播一個SUBSCRIBE，詢問是否有應用層的網狀網路存在。

Step 2. MN8向MN4詢問是否有收到MN6所廣播的SUBSCRIBE訊息。

Step 3. MN4會回到MN8是否自己有收到這份資訊，在本例中MN4並沒有收到MN6的資訊，所以回傳的OK訊息中告訴MN8自己並無收到。

Step 4. **MN8回傳OK訊息給MN6告知其目前已存在應用層網狀網路，並且接受他的加入，傳送的OK訊息會加上數位簽章並且裡面會包含，<ECC>與<Public key>兩個HEADER讓MN6了解MN8的Public key與目前使用的橢圓曲線，MN6可以依據收到的<ECC>與<Pubkiv key>來做驗證訊息是否經過竄改，之後就可以使用這兩個訊息來對MN8所傳送的訊息做驗證，此後訊息在傳遞時皆加上簽章。**

Step 5. MN8傳送NOTIFY訊息給MN6，NOTIFY訊息包含目前網狀網路上存在的節點與節點的分布情況，還有目前已經存在的ALM群體提供哪些服務，MN6收到NOTIFY訊息之後，便先使用MN8的Public key做簽章驗證，如果沒有錯誤，便使用此訊息建立節點名單，如果有便丟棄此訊息重新加入應用層網狀網路。

Step 6. **MN6回覆MN8已接收到NOTIFY訊息，並且加上MN6使用<ECC>參數所計算出的Public key回傳給MN8。**

Step 7. MN8告訴MN4與MN6，MN8成為SIP Gateway以及新的節點MN6加入到應用層的網路網路上，這個訊息裡面會包含MN6的Public key。

Step 8,9. MN4與MN6回覆MN8已收到訊息，MN6會比對是否自己的Public key沒有錯誤，如果有錯，MN6會重新傳送給MN8。

Step 10. MN4傳送NOTIFY_d訊息給子節點，通知她們網路的狀況有改變，需要更新節點名單。

Step 11,12,13. MN7,MN8,MN9回傳訊息，完成新節點加入的程序。

透過本研究所提出的通訊流程，可以去確保訊息在傳遞時，訊息內容的完整性以及不可否認性，不會因為節點在傳送中經竄改或者是接收偽造的訊息，而產生安全性的疑慮。

3.2 金鑰交換

MANET上的每一個節點，都需要經過其他節點的轉傳，所以，當每一個節點在轉傳其他節點的資料時，可以很輕易的就擷取到訊息的內容，特別是使用SIP做為通訊協定傳輸時，每個訊息都是明文，所以，轉傳的節點可以很容易的就獲得訊息的內容，因此要如何保護訊息內容的私密性，是SIP在MANET上傳輸訊息的重要工作。

3.2.1 ALM GROUP上的安全威脅

當一個存在於應用層網狀網路的節點想要發起一個ALM群體，它會先將訊息傳播到他的SIP Gateway，再由SIP Gateway傳送給其他所有的點，Group-ID即代表這個ALM群體的發起人，也就ALM的領導者，一旦其他節點收到這個訊息，如果要加入這個群體，便會回傳一個OK的回覆給群體的領導者，群體的領導者會等待一段時間來接收OK的回覆訊息，然後，在等待時間結束之後，群體領導者會發送，NOTIFY給所有這個ALM群體的參予者。

當有一個新的節點要加入這ALM群體時，因為它加入這個應用層的網狀網路時，SIP Gateway傳送的NOTIFY的訊息裡面，就會包括目前已經啟動的ALM

群體，因此這個節點就可以依據裡面的訊息，送INVITE訊息給ALM的群體領導者，這個INVITE的訊息裡面會包含一個Join的HEADER，依據這個HEADER，收到的ALM群體領導者比對自己的Event-ID，與可能會有某些節點的加入限制(ADMISSION CONTROL)，如果ALM群體領導者允許這個節點加入，就傳送一個NOTIFY回去給新加入的節點，裡面會包含目前所有加入這個ALM的所有資訊，當新節點回傳OK即完成加入ALM群體程序，之後SIP Gateway再送NOTIFY_d告訴其他目前已經存在於這個ALM群體的成員，有新節點的加入，並且訊息包含這個節點的資訊。

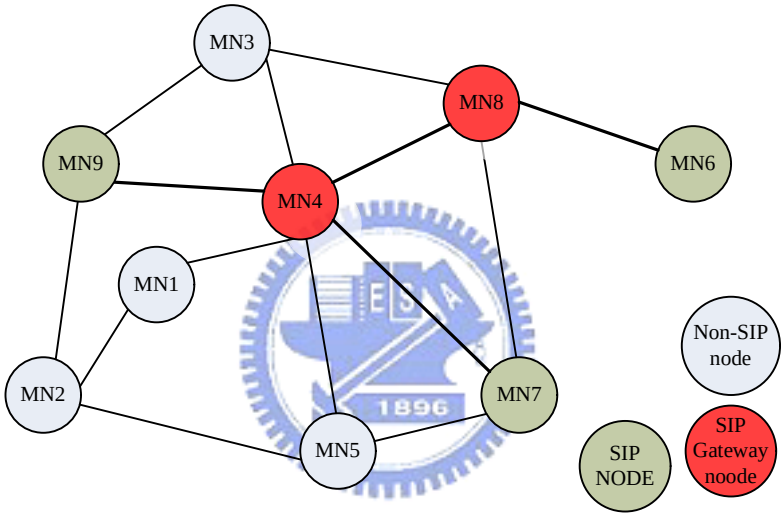


圖 25　ALM 建立節點分布圖

以在第二章中所討論ALM群體建立的流程為例，圖22為節點的分布圖，其ALM群體的建立流程如下:

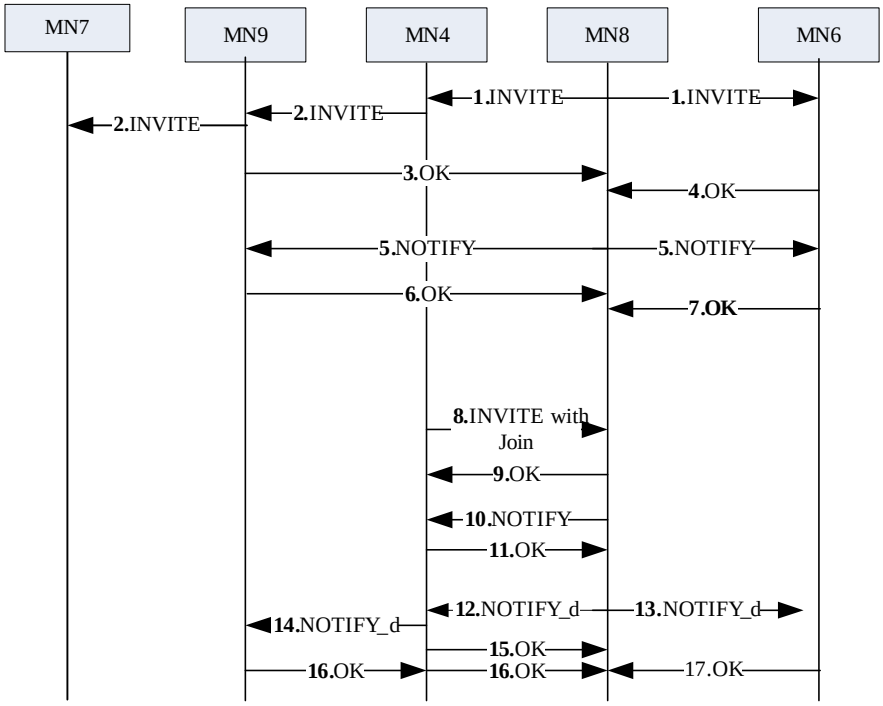


圖 26、加入 ALM 群體流程圖

Step 1. MN8決定發起一個ALM群體，傳送INVITE訊息給其他的節點。

Step 2. 因為MN9、MN7為MN4的子節點，所以透過MN4來轉傳MN8的INVITE訊息。

Step 3,4. MN6與MN9決定加入MN8所成立的ALM群體。

Step 5. MN8在等待回應期間結束之後，就傳送NOTIFY訊息給決定加入的節點MN6與MN9告訴她們，參與這個ALM群體的成員為MN8、MN6與MN9。

Step 6,7. MN9與MN6回傳OK訊息給MN8，回覆MN8收到NOTIFY訊息。

Step 8. MN4決定加入MN8的ALM群體，因此傳送INVITE訊息給MN8其中包含一個Join的Header。

Step 9. MN8接受MN4的加入請求，回傳OK訊息回去給MN4。

Step 10. MN8傳送NOTIFY訊息給MN4，裡面包含參與這個ALM群體的成員。

Step 11. MN4回覆MN8已收到NOTIFY訊息。

Step 12,13,14 MN8傳送NOTIFY_d訊息給ALM群體上的成員，有新成員加入。

Step 15,16,17 收到NOTIFY_d的節點回傳OK訊息，確認以收到，如此便完成新節點加入的流程。

當MN8在Step 1傳送INVITE訊息時，如果在MN4為MN9的SIP Gateway，MN4惡意的去竄改INVITE訊息，將INVITE訊息變成由MN4發起的ALM群體，那就可能會發生一個節點加入的是一個錯誤的ALM群體，而產生MAN-IN-MIDDLE ATTACK，所有的訊息都會先透過錯誤的節點MN4在送到MN8去，因此如何保證在傳送MN8所產生的訊息完整性，是在發起ALM群體時的一個問題。

其次，當ALM群體在進行訊息交換時，以傳送Instant Message為例，因為SIP為一個明文傳送的通訊協定，而且MANET上，每一個節點都有可能幫其他節點傳送訊息，如果在訊息的內容沒有做私密性的處理，那訊息的內容會在傳送的期間簡單的被別有居心的節點窺探甚至被竄改，如何去保證在這個架構下，維持訊息內容的私密性與完整性也是另一個重要問題。

3.2.2 在MANET上SIP的金鑰交換協定

一個節點要建立一個新的ALM群體時，首先會送出INVITE訊息，在3.2.1中討論到，如果這個訊息經過竄改，那之後所要加入這個ALM群體的成員，可能會遭受竊聽或者是中介者攻擊(MAN IN MIDDLE ATTACK)，因此首先要保證這個INVITE訊息的完整性，而當ALM群體建立完成之後，如果兩個群體成員進行這個ALM群體提供的服務做通訊，例如在一個<IM>的群體，就是進行Instant Message服務，會使用MESSAGERequest(RFC 3428) [5]訊息來做通訊，但是因為在SIP中Request MESSAGE的是明文，所以，在MANET上所有轉送的節點，都可以簡單就竊聽到兩方通訊的訊息，因此，在MANET上，兩點通訊的私密性更是顯得額外重要，因此本研究提出使用橢圓曲線的Diffie-Hellamn金鑰交換法來做訊息加密，在MANET環境裡，使用金鑰長度短與運算量小的加解密法是很重要的，因此使用橢圓曲線來做加解密。

因此，在本研究的應用層的網狀網路建立之後，每一個節點皆有其他節點的Public key，可以用來用驗證訊息的完整性，同時也可以驗證訊息是否為發起的節點傳送，因為數位簽章具有不可否認性，當一個節點發起一個ALM群體之後，

會傳輸一個INVITE訊息給其他在這個應用層網狀網路的成員，之後這個節點會等待一段時間收集回傳的OK訊息，每一個成員在接收到這個訊息之後，如果想要去加入這個新建立的ALM群體，就使用發起節點的Public key去驗證這個訊息，如果沒有遭受竄改以及確實由這個ALM群體領導者所發起，便回傳OK訊息回去，並且加上簽章，如果驗證有錯就遺棄這個訊息，發起的ALM群體的節點在收到OK訊息之後，就依據這個請求加入的節點，使用這個節點的Public key去做驗證是否經過驗證，以及是否確實為要求加入節點所發起的訊息，如果無誤就接受這個節點，在等待時間結束之後，這個發起ALM群體的領導者便傳送NOTIFY__k給所有參予者個ALM群體的節點，這個訊息裡面包含了這個ALM群體所產生的橢圓曲線的參數，將橢圓曲線的參數包括p、a、b、G放入<ECC>(在本研究中為一個ECCParameterSpec的Object)，然後加上簽章傳送給加入這個ALM群體的節點，之後收到的ALM成員在確定橢圓曲線參數沒有被竄改之後，就傳送OK訊息回去裡面包含節點本身計算出來的Public key在<Public key>HEADER裡面，加上簽章回傳給ALM的領導者，之後ALM領導者收到所有人的回傳之後，再傳送一個NOTIFY回去給其他的參予者，裡面會有這個ALM群體所有的參予者，以及這個些參予者的Public key，之後，如果要傳送訊息，便可以使用節點本身的Private key與對方節點的Public key一起做加密，對方節點收到之後，就使用節點本身的Private key加上對方節點的Public key做解密，再加上簽章，這樣在傳送的期間，可以保護訊息的私密性以及訊息的完整性，在MANET上傳輸經過節點轉傳時，訊息的內容可以保密，也不怕訊息被竄改。

以在3.2.1中所討論ALM群體建立的流程為例，使用同樣的節點的分布圖(圖24)，本研究所提出之使用金鑰交換的ALM群體建立流程如下:

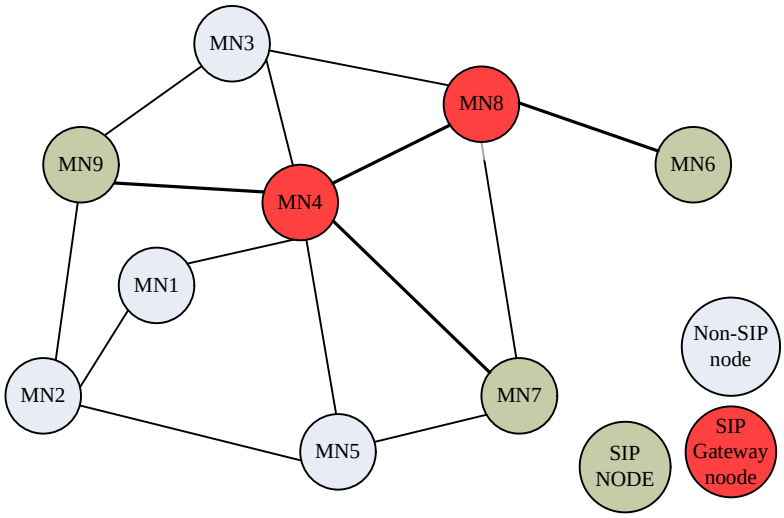


圖 27、ALM 建立節點分布圖

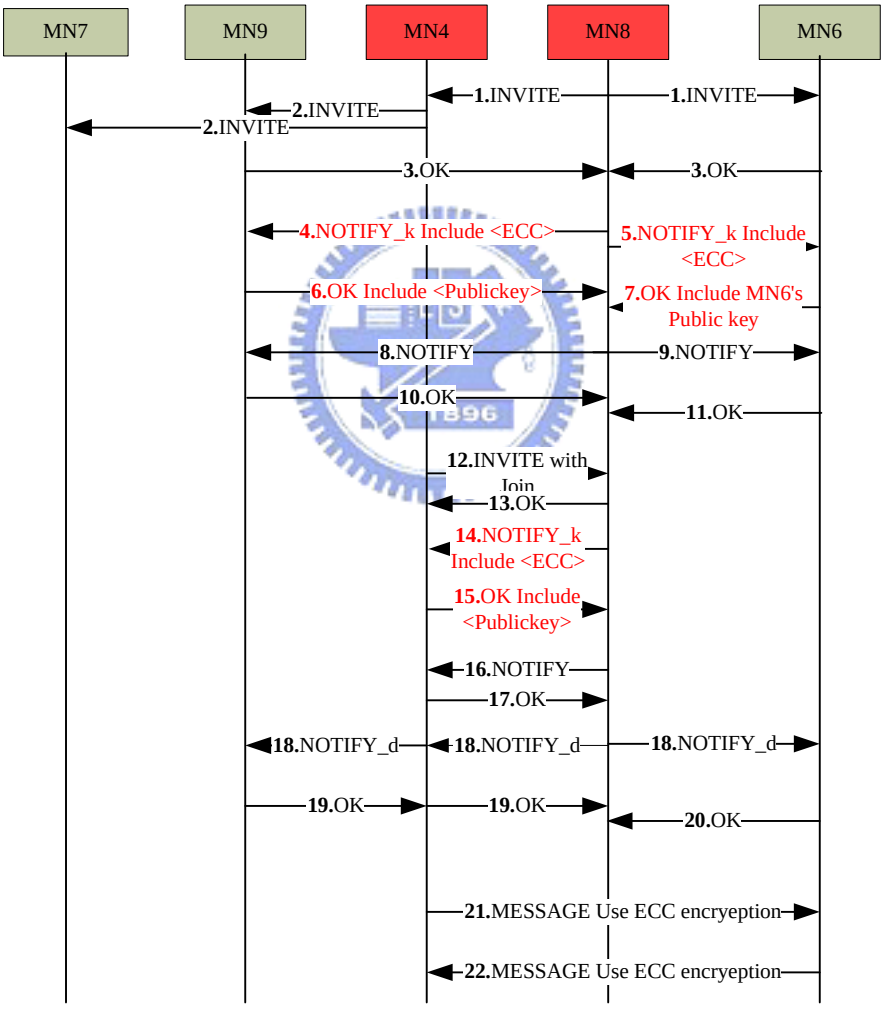


圖 28、架入金鑰節點流程圖

Step 1. MN8決定發起一個ALM群體，傳送INVITE訊息給其他的節點，這個INVITE的訊息會使用MN8在應用層網狀網路所交換的數位簽章的Private key加上簽章。

Step 2. 因為MN9、MN7為MN4的子節點，所以透過MN4來轉傳MN8的INVITE訊息。

Step 3,4. 所有收到INVITE訊息的節點，都會使用MN8的Public key做簽章的驗證，如果有錯誤，就丟棄這個訊息，如果簽章驗證正常，再決定是否加入這個ALM群體，在本例中MN6與MN9決定加入MN8所成立的ALM群體，並且回傳OK訊息給MN8，MN6與MN9也在傳遞的OK訊息中加入自己的簽章，這樣一來可以防止有節點偽造節點要加入的訊息也竊取資料。

Step 5. MN8收到回傳的OK訊息之後，先去驗證是否簽章正確，如果產生問題則丟棄這個OK訊息，如果簽章正確，即將傳送OK訊息的節點設為ALM群體的成員，在等待期間結束之後，MN8傳送NOTIFY_k訊息給MN6與MN9，這個NOTIFY_k訊息為MN8所選定的橢圓曲線參數，用來計算Diffie_Hellman金鑰交換的每個ALM成員的Public key，此訊息加上簽章，以防止訊息遭受竄改。

Step 6,7. MN6與MN9驗證完傳送的訊息之後，依據<ECC>的參數，計算出自己的Public key，加上簽章回傳OK訊息給MN8。

Step 8,9. MN8將加入ALM群體的成員資料，包括各個節點計算的Public key，加上簽章傳送給ALM群體的成員MN6與MN9。

Step 10,11. MN6與MN9收到NOTIFY訊息之後，先比對簽章，然後再比對自己的Public key是否正確，沒有錯誤的話，就回傳OK訊息給MN8。

Step 12. MN4決定加入MN8的ALM群體，因此傳送INVITE訊息給MN8其中包含一個Join的Header，加上簽章傳送給MN8，如此一來可以確保這個加入訊息沒有被竄改，更重要的是透過簽章，可以確定這個訊息是由MN4所傳送。

Step 13. MN8驗證無誤之後，接受MN4的加入請求，回傳OK訊息回去給MN4。

Step 14. MN8傳送一個NOTIFY_k訊息給MN4，裡面包含這個ALM群體所用來做為金鑰交換的橢圓曲線參數<ECC>。

Step 15. MN4收到之後，驗證簽章，之後便使用<ECC>中的參數，計算自己的Public key，加上簽章回傳給MN8。

Step 16. MN8驗證簽章之後，便加上簽章傳送NOTIFY訊息給MN4，裡面包含目前ALM群體上的成員以及成員的Public key。

Step 17. MN4收到NOTIFY訊息之後，驗證簽章接著回傳OK訊息給MN8。

Step 18. MN8傳送NOTIFY_d訊息給所有的ALM群體成員，有ALM成員加入，將新成員的資訊以及新籌員的Public key一起傳送給ALM群體的所有成員。

Step 19,20. 所有的ALM成員驗證簽章之後，就回傳OK訊息給MN8表示已經收到更新的訊息，MN4也比對自己的公開金鑰是否有錯，如果無誤就回傳OK訊息回去，如此便完成新節點加入的流程。

Step 21,22. MN4使用MN6的Public key與MN4自己的Private key做加密傳送Instant Message給MN6，MN6收到之後使用自己的Private key與MN4的Public key做解密，雙方進行通訊。

而當節點要離開ALM群體以及應用層網狀網路時，網路與ALM群體成員只需要將這個節點從節點名單與ALM群體名單移除即可，不需要額外的動作就可以移除離開的成員。

透過本研究所提出的協定，可以在ALM群體加入以及訊息傳遞的過程中，保障其訊息傳遞的不可否認性，完整性與私密性，提升了SIP系統在MANET上運作的安全性。

第四章、系統設計與效能分析

本章將依據本研究所提出的協定，做測試與實作出SIP基礎的群播系統在MANET上的協定，並且透過效能的計算與比較，來分析本研究所提出利用橢圓曲線數位簽章與橢圓曲線Diffie-Hellman金鑰交換和原本架構的效能分析，利用偽造攻擊與訊息竄改等技術，證明原本的架構在MANET上會面臨許多安全威脅，再比較加入本研究所提出的安全協定，所能克服的問題。

4.1 環境說明

本實驗架構在交通大學的學術網路環境之上，使用四台電腦，建立起一個MANET環境，並且在這個環境上，進行本研究的建置與分析，這些電腦的硬體設備及軟體環境如下。

開發平台 ：

CPU: AMD K8 3000+

RAM: 1G DDR RAM

HARDDISK: WD 400G

NIC: Marvell Yukon Gigabit Ethernet card

DLINK DWL-AG132 802.11a/b/g

作業系統: Windows XP

設計平台: Eclipse JAVA整合開發工具

程式語言: JAVA

測試節點 ：

IBM X31*3

CPU: Intel Pentium M 1.4 GHz

RAM: 512 MB DDR RAM

HARDDISK: 40GB

NIC: 802.11b+g

作業系統: Windows XP

4.2 系統設計與實作

本研究的系統在實作建置一個在MANET上SIP群播的通訊系統，本研究中包含三個部份，1.SIP的群播系統的實作 2.橢圓曲線的數位簽章 3.橢圓曲線Diffie-Hellman金鑰交換機制，本研究使用JAVA程式語言去實做這個系統，使用的JAVA版本為1.5版，並且額外加入JCE，NIST SIP RI[20]，IAIK ECC[21]，以及FlexiProvider[22]的FlexiECProvider等套件，這些套件工具來實做出在MANET上SIP基礎的群播架構，並且加入橢圓曲線的簽章以及橢圓曲線的Diffie-Hellam金鑰加換並且加密訊息，進而做效能分析與探討。

4.2.1 SIP的群播系統

在這個部份的實作當中，SIP的群播架構[1]分成三個部份與一個架構在這個群播系統之上的應用程式，本研究以SIP進行Instant Message作為SIP應用層級的程式，SIP群播系統分為: SIP應用層網狀網路(SIP Overlay Mesh)、網路節點(Node List)與SIP ALM群體三個部份(SIP ALM Group)，這四個部份的關係如下圖:

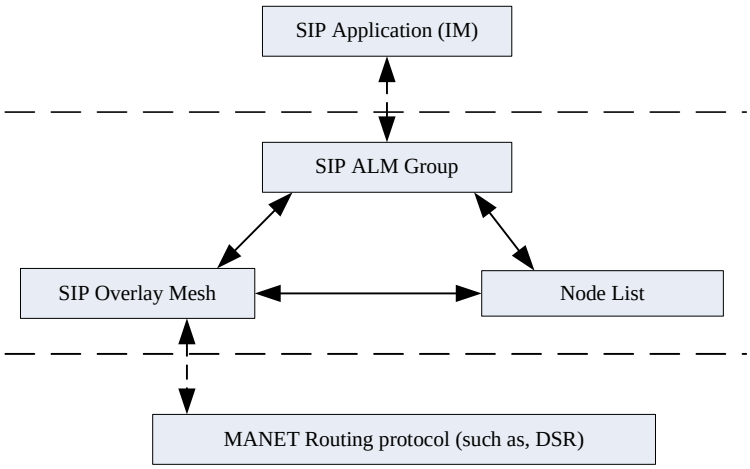


圖 29、SIP 在 MANET 上的架構圖

在本實驗中架構在ALM群體之上的SIP應用程式，以利用SIP做Instant Message的訊息傳送為主，這個部份的設計是基於RFC 3428[5]的標準，使用Message的HEADER進行訊息的傳送，為了單純實驗的狀態，在本實驗中即簡化ALM群體的情況，以產生訊息傳遞的ALM群體作為主，並且在產生ALM群體之後，進行雙方面的訊息對話。

SIP 應用層的網狀網路(SIP Overlay Mesh)元件，在本實驗中以一個overlayMeshImp物件作對應，overlayMeshImp繼承在NIST SIP RI中的SIP Listerner，並且引用NIST SIP RI中定義的SIP元件，實現使用SIP做訊息交換的架構，這個overlayMeshImp的物件使用init作為啟動，當程式啟動之後，便會發送SUBSCRIBE詢問目前網路上是否已經存在SIP Gateway，若有收到回覆訊息，即以收到回覆訊息的FROM HEADER中的SIP URI當作是自己的SIP Gateway並且將自己的isGateway設為false，如果沒有人回應isGateway即為true，成為一個SIP Gateway之後要去聆聽5020的port，並且接收新節點傳送的SUBSCRIBE，與定期傳送NOTIFY_d告訴其他網路成員目前網路狀況的改變。

節點名單(Node List)在本實驗中，使用nodeList物件來實現，當overlayMeshImp收到SIP Gateway送來的NOTIFY訊息之後，便將NOTIFY訊息的內容使用nodeList的rebuild方法，將NOTIFY訊息內容放進去，之後就會回傳一個繼承HashMap的nodeList的Object，之後當收到NOTIFY_d時，便可以使用setNode

、getNode、setSipGateWay、removeNode、removeSipgateway對節點名單做修改。

SIP ALM群體(SIP ALM Group)在本實驗中，使用ALMGroupImp物件做實作，這個物件包含幾個主要的方法(method)，initALM、JoinALM、exitALM，getALMList，以及一個sendMessage的方法，因為本實驗主要是以即時訊息(Instant Message)傳送作為本實驗SIP應用程式，所以，在ALM群體建立之後，群體的成員可以互相傳送訊息，因此本實驗將此功能整合在ALM群體之內，initALM會送出INVITE訊息去建立ALM群體並請成為ALM群體領導者，成為ALM領導者也會如SIP Gateway一樣會傳送NOTIFY_d訊息給ALM成員，告訴群體成員目前的群體變動，JoinALM會去送出一個包含Join HEADER的INVITE訊息去加入一個目前已經建立的ALM群體，exitALM會送Bye訊息給ALM領導者，在領導者收到之後會發送NOTIFY_d告訴其他的ALM成員，有ALM成員要離去，所有的ALM群體成員在收到之後，就將這個節點從ALM名單中移除。一個ALM群體的成員可以依據ALM名單(ALM List)來選擇進行通訊的成員，使用sendMessage來做通訊，在這個方法中將目的成員與傳輸的字串當作參數放入執行，雙方便可以進行通訊。

4.2.2 數位簽章實作

本實驗使用IAIK所公開的橢圓曲線簽章套件(iaik_ecc.jar)，為本實驗的SIP訊息傳遞加上數位簽章，在這個套件中，需要加入JAVA的官方JCE套件(Java Cryptographic Extensions)來做支援，在ECCSignature物件當中首先決定使用的金鑰長度，利用IAIK的物件ECCParameterFactory用來產生一條橢圓曲線的參數，之後根據金鑰的長度產生一個ECCParameterSpec，這個物件便是橢圓曲線的參數物件，之後產生一對KeyPair，之後再依據這個參數產生橢圓曲線數位簽章的金鑰對(KeyPair)，之後使用這個KeyPair的Private key就可以進行橢圓曲線的數位簽章，會產生一個Byte[]的簽章，將這個簽章放入SIP的<SIGNATURE>的HEADER裡面，當對方收到就可以使用發起者的簽章做驗證。

在本研究中當加入應用層的網狀網路(Overlay Mesh)就會依據橢圓曲線的參數ECCParameterSpec產生自己的KeyPair物件，所有在這個網狀網路的成員都會知道對方的公開金鑰，當收到一個訊息時，便將<SIGNATURE>取出與訊息做驗證，當簽章沒有錯誤時，再繼續進行訊息的處理。

4.2.3 金鑰交換實作

本實驗使用FlexiProvider的FlexiECProvider來做橢圓曲線的金鑰交換與訊息加解密，在本研究中，ALM群體會使用橢圓曲線來做金鑰交換以及使用Diffie-Hellman的交換機制產生一把只有雙方才能產生的Session Key，並且利用這把Session Key來做加解密動作，在程式當中，先取得橢圓曲線的ECCParameterSpec，並且產生KeyPair物件，將自己的Public Key公開，在ALM群體的所有成員皆有對方的Public Key與橢圓曲線的參數ECCParameterSpec，當要傳送訊息時，一方將自己的Private Key與對方的Public Key使用ECDH物件中的generateSessionKey產生一把Session Key，之後Session Key當作參數，使用messageEncrypt方法加密要傳送的訊息內容，再將加密後的訊息放入MESSAGE訊息的內容(content)裡面，當對方收到訊息之後，跟去FROM HEADER裡面的SIP URI去ALM List裡面取出對方的Public Key，與自己的Private Key產生Session Key之後，解出訊息，這樣的過程。

4.3 安全性分析

這一節依據在第二章中討論的安全威脅，以幾種簡單的攻擊實驗，去證明這個架構在簡單的竊聽以及攻擊下都有可能產生問題，並且使用本研究所提出的架構去防止類似攻擊，並且分析，舊有的架構與本研究所提出的架構在安全性上所呈現差異。

4.3.1 Tearing Down攻擊

首先，使用Tearing Down攻擊來比較原有架構與本研究所提出的架構對於Tearing Down攻擊的防禦能力，在一個ALM群體當中如果有一個群體的成員要離

開ALM群體，就傳送BYE訊息給ALM群體領導者，ALM群體領導者在收到這個訊息之後，便會傳送NOTIFY_d給其他的ALM群體的成員，告訴她們這個成員已經離開這個ALM群體，於是所有的群體成員會將這個節點的資料從ALM名單移除，這個實驗使用下列的拓撲狀態:

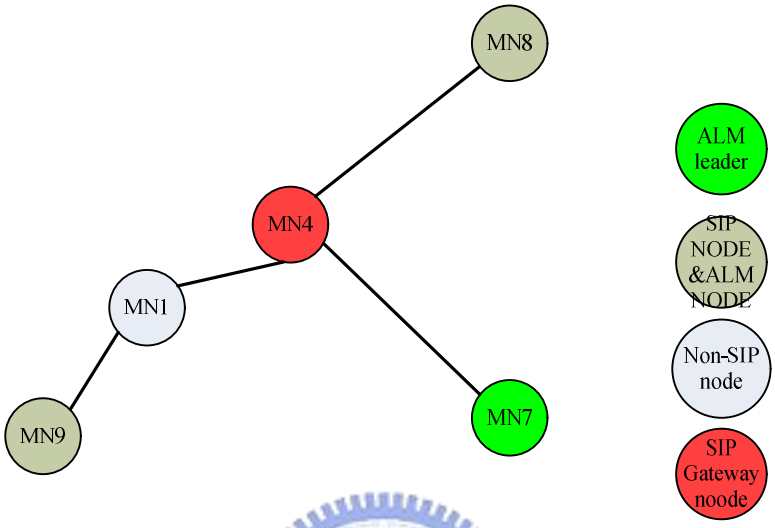


圖 30、實體層拓樸

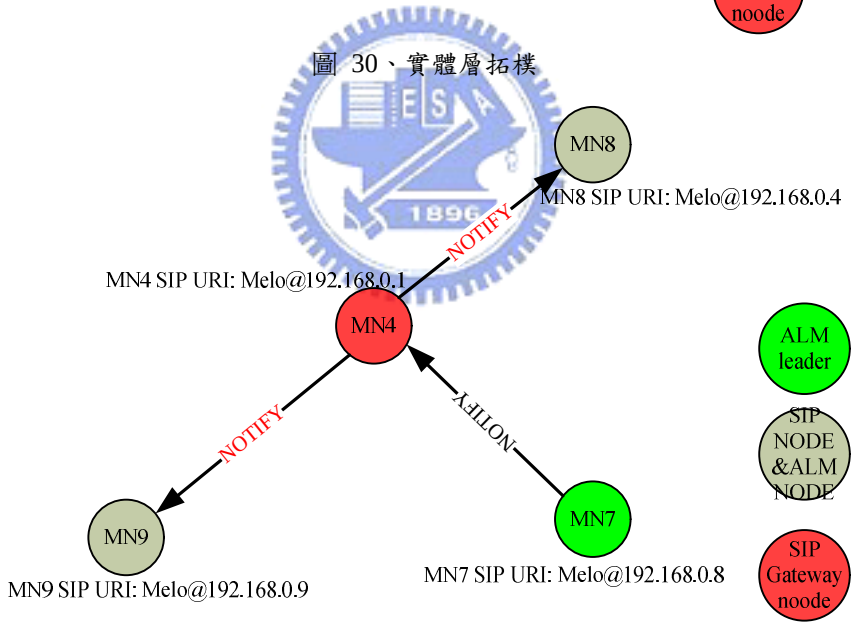


圖 31、應用層網狀網路拓撲(Overlay Meshed Network)

在原有的架構中，首先使用MN8與MN1兩個節點傳送FROM HEADER為MN9的SIP URI: Melo@192.168.0.9給MN7，首先使用非SIP節點的MN1，使用偽造MN9傳送BYE訊息給MN7，MN7收到之後回傳OK訊息給MN1，之後，MN8就收到NOTIFY_d訊息，如此，MN9即從ALM群體中被阻斷，第二次使用MN8傳送BYE訊息給MN7，FROM訊息依舊為MN9的SIP URI: Melo@192.168.0.9，

MN9依舊輕易的被攻擊者從ALM群體中阻斷，而且更清楚的發現，甚至是在同一個ALM群體的成員都可以輕易的將其他的ALM群組成員從ALM群體中阻斷。

在本研究的架構中，在傳送出訊息之前，會先對訊息產生簽章，並且將簽章做驗證，本實驗中使用金鑰長度為192來做簽章產生， 並且將簽章放入<Signature> HEADER裡面，在MN7收到之後就先去驗證<Signature>中的簽章，如果使用FROM HEADER裡面傳送者的Public Key做驗證，如果驗證成功，表示確實為FROM HEDER傳送並且沒有遭受破壞，如果驗證失敗，即將此訊息丟棄，在實驗中，首先使用MN1來做攻擊，BYE訊息中，不加入<Signature>HEADER，在MN7收到之後，因為沒有<Signature>HEADER就直接丟棄，而MN8也沒有收到NOTIFY_d訊息，其次使用MN8傳送MN9的BYE訊息，並且加上MN8的簽章，在MN7收到訊息之後，去使用MN9的Public Key去驗證這個簽章，結果，發現這個簽章根本錯誤，所以，也直接遺棄，最後使用MN9傳送BYE訊息給MN7，並且加上自己的簽章，MN7在收到之後，使用MN9的Public Key去驗證這個簽章正確，便傳送NOTIFY_d訊息給MN8，MN9離開這個ALM群體。

由簡單的對照可以知道，在原本的架構當中，很容易就受到Tearing Down攻擊，而且不僅外部的惡意節點就算ALM群體的成員也可以輕易將某個節點做Tearing Down攻擊，使其ALM群體，加上本研究使用的簽章方式之後，就可以輕易的解決可能面對的攻擊。

4.3.2 訊息竊聽

在原本的架構中，在ALM群體之上使用SIP應用程式作訊息交換時，訊息的內容在傳送的過程當中，會相當容易的被惡意者做竊聽，特別是在MANET環境上，每個節點在傳送訊息時，會經過其他節點的轉傳，就算不特別使用竊聽技術，在訊息傳送的同時，也可以簡單的就獲得通訊的內容，同樣使用下列的拓撲狀態進行實驗:

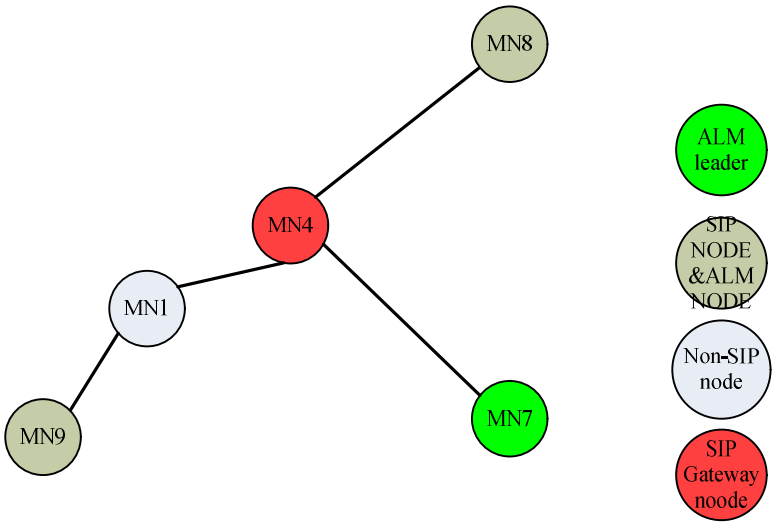


圖 32、實體層拓樸

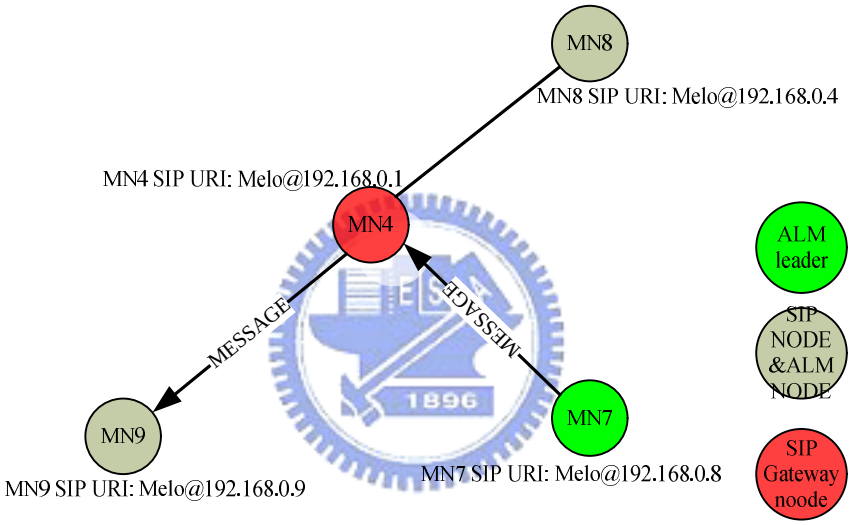


圖 33、應用層網狀網路拓撲(Overlay Meshed Network)

在原本的架構當中使用，在ALM群體的MN9要與MN8要進行SIP訊息對話時使用Message的訊息交換，在實體層的傳輸時，會傳輸的訊息會經過MN1與MN4再到MN8，因此，MN1可以輕易的在轉傳訊息時獲得MESSAGE訊息內容，而在應用層網狀網路中MN4是SIP Gateway更可以很簡單的，就取得經過他所傳送出去的訊息。

因此在本研究中提出，使用橢圓曲線來做加密，經過Diffie-Hellman已經產生一把Session Key，利用這把Session Key將訊息加密，在當MN9傳輸訊息經過MN1到MN4到MN8時，因為訊息的內容經過加密，並且MN1與MN4無MN8的Private Key，所以，無法將訊息解開，而MN8在取得資料之後，先對訊息做簽章

驗證，在訊息內容做解密，如果簽章錯誤比較訊息已經不完整，就將訊息遺棄，若訊息簽章無誤，就產Session Key，將訊息解密，透過這樣的實驗，可以確保在ALM群體相互傳訊息時，訊息的私密性以及完整性獲得維持。

4.4 效能分析

在本節當中對於本研究的架構提出效能分析，首先以加入應用層網狀網路(Overlay Mesh)的時間(setup time)，分析原本架構、使用橢圓曲線簽章與RSA數位簽章做分析比較，第二、計算ALM群體在傳送訊息所花費的時間，比較原有架構與使用橢圓曲線Diffie-Hellman機制做加解密所花費的時間，最後比較原有架構、使用橢圓曲線加密，與使用橢圓曲線簽章與加密在訊息傳輸時所花費的時間。

4.4.1 應用層網狀網路(Overlay Mesh)加入時間比較

這個實驗使用原本論文的架構，及本研究使用橢圓曲線的簽章，以及使用RSA簽章做比較，由節點發送SUBSCRIBE出去至收到NOTIFY回傳OK訊息的這一段期間，計算所花費的時間，基於安全性的考量，使用橢圓曲線金鑰長度192與RSA金鑰長度1024來做比較，計算傳送一百次，並且分別計算並平均其完成的時間，得到的結果如下圖:

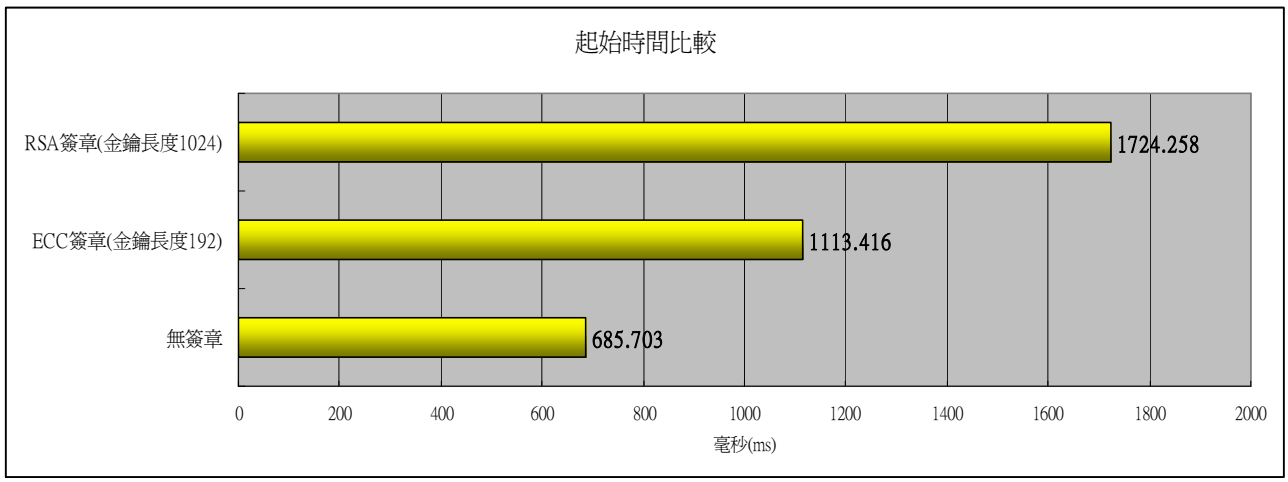


圖 34、RSA 與 ECC 簽章效能比較表

為了達到相同安全度所以RSA的金鑰長度選取1024的長度，但是可以很清楚

的看到需要的傳輸時間與簽章時間，幾乎達到原本無簽章架構的兩倍以上，而使用ECC簽章在一樣的安全度下，所需要的時間只並沒有明顯的增加。

4.4.2 MESSAGE傳遞的時間比較

這個實驗使用原本沒有使用加密系統的架構，與本研究所提出使用橢圓曲線Diffie-Hellam金鑰交換所產生的Session Key，做加密保護訊息私密性，比較其從Message訊息發出至收到OK回覆訊息之間的時間差異，在一百次的互相傳輸之後所統計的時間分別為以下，多出的加密與且解密時間花費的時間並不多。

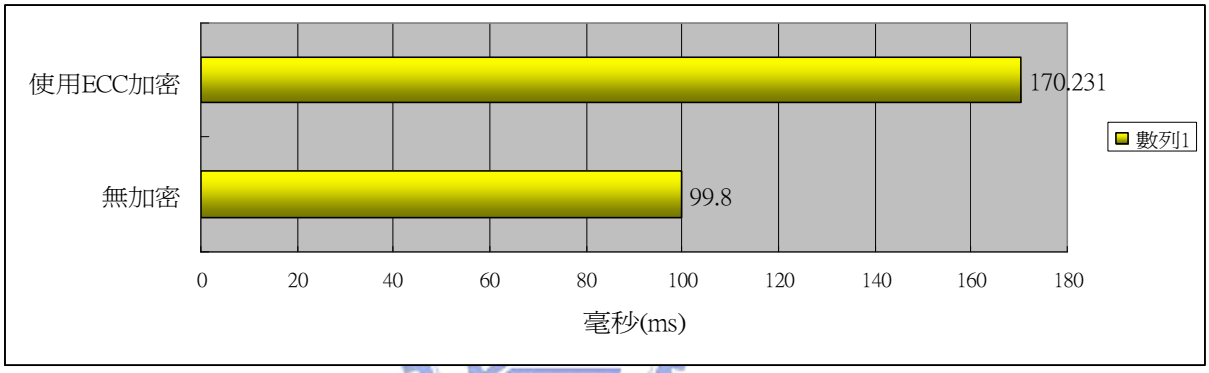


圖 35、加密效能比較表

4.4.2 MESSAGE使用金鑰交換與橢圓曲線簽章花費時間比較

這個實驗使用，使用原有的論文架構，與橢圓曲線Diffie-Hellam金鑰交換所產生的Session Key，做加解密保護訊息私密性，再加上橢圓曲線Diffie-Hellam金鑰交換所產生的Session Key，做加密保護訊息私密性與橢圓曲線的數位簽章，保護訊息的完整性與不可否認性，與使用RSA做數位簽章與訊息加密，四個做比較分析。

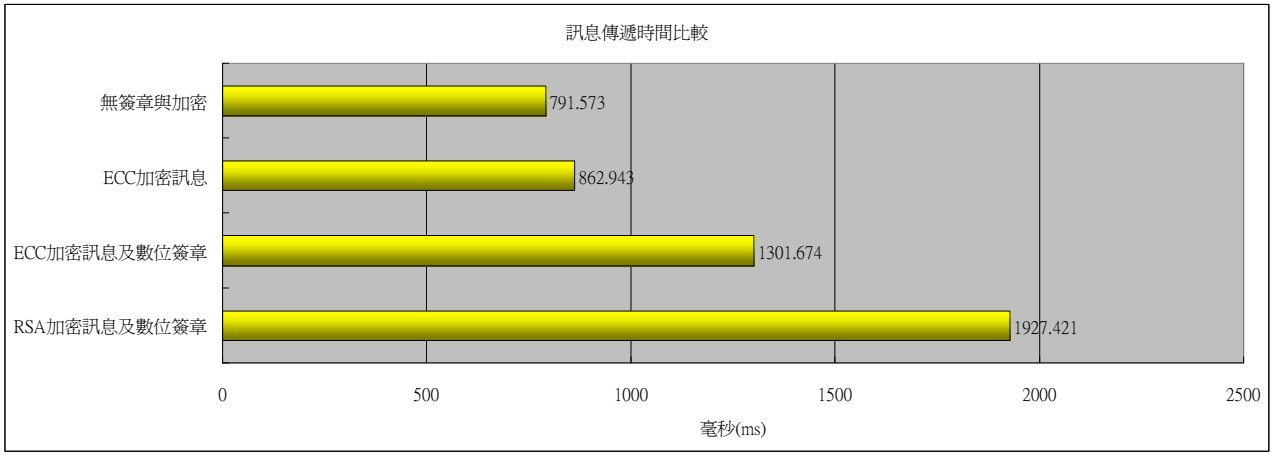


圖 36、訊息傳遞時間比較表

在使用橢圓曲線加密與數位簽章的安全架構下，所需要的時間甚至比RSA簽章所需的時間更點並且傳輸的時間與RSA做簽章與加密遠遠節省時間，而且與原始架構所需的傳遞時間也不需要太多，由此可見，使用橢圓曲線做簽章與加密，並不會造成系統過多的負擔，但是能夠顯著的提升系統的安全性。

第五章、結論與未來展望

5.1 結論

本研究的貢獻在於改進SIP在MANET上的群播架構的安全性，並且根據MANET的特性，討論目前SIP現有的安全機制可能會面對的問題，提出一個使用數位簽章與金鑰交換的協定，可以在MANET這種公開，動態的網路環境中提供訊息的私密性以及完整性，最後進行實驗比較橢圓曲線在數位簽章以及金鑰交換上的效率，橢圓曲線在相同的安全性下，需要的簽章長度相當短，並且也因為這樣的特性，橢圓曲線加解密的計算量較少，計算速度較快，因此很適合在動態，記體體受限以及計算能力較弱的裝置上使用，並且因為較少的計算量，對於在MANET上的裝置，也能夠達到節省電力的目標。

本研究的實作系統，使用SUN JSR 32[20]中定義的SIP API架構與NIST公佈的SIP RI來進行SIP通訊系統的建置，並且根據IEEE 1363定義的橢圓曲線加解密架構，使用IAIK與FlexiProvider提供的免費套件來進行橢圓曲線的數位簽章以及橢圓曲線的Diffie-Hellman金鑰交換，最後建置出一個在MANET環境上使用橢圓曲線作數位簽章與金鑰交換的SIP群播系統，透過這個系統可以進行即時訊息(Instant Message)交換，並且利用此實作證明使用本論文的協定可以獲得較高的效能與安全性。

5.2 未來展望

本研究中並沒有特別針對MANET的分布情況做討論，例如，節點分布特別密集與特別分散對於系統的影響，並且在SIP的應用程式的部份，沒有做較大範圍的討論與比較，特別是語音能否穩定的透過這樣的架構進行通訊，在橢圓曲線方面，對於認證使用的橢圓曲線以及金鑰交換的橢圓曲線，如果使用同樣一條橢圓曲線，是否對於安全性有疑慮，或者是能夠增加效率，這是未來可以著手討論的方向。

參考文獻

[1] A.Agarywal and Y.Yu, ”A SIP-based Multicast Framework in MANET” Wireless And Mobile Computing, Networking And Communications, 2005. IEEE International Conference on Volume 3, 22-24 Aug. 2005.

[2] E.T.Aire, B.T.Maharaj and L.P.Linde, ”Implementation Considerations in a SIP based secure Voice over IP Network”, AFRICON, 2004. 7th AFRICON Conference in Africa Volume 1, 2004.

[3] T.C.Chiang and Y.N.Huang, ”Group Keys and Multicast Security in Ad Hoc Networks”, Parallel Processing Workshops, 2003. Proceedings. 2003 International Conference on 6-9 Oct. 2003.

[4] H. Cohen, A. Miyaji and T. Ono, “Efficient Elliptic Curve Exponentiation Using Mixed Coordinates,” Advance in Cryptology-ASIACRYPT’98,LNCS 1514, Springer, pp. 51-65, 1997.

[5] B. Campbell, J. Rosenberg and H. Schulzrinne, ” Session Initiation Protocol (SIP) Extension for Instant Messaging”,RFC 3428,December 2002.

[6] S. Duanfeng, L. Qin, H. Xinhui, and Z. Wei, “Security mechanisms for SIP-based multimedia communication infrastructure” ,Communications,Circuits and Systems, 2004. ICCCAS 2004. 2004 International Conference on Volume 1, 27-29 June 2004.

[7] M.Handley, H.Schulzrinne, E.Schooler, and J.Rosenberg, ”Session Initiation Protocol, ” RFC 3261,June 2002.

[8] D. Johnson and A. Menezes, “The Elliptic Curve Digital Signature Algorithm (ECDSA),” Technical Report CORR 99-34, Centre for Applied Cryptographic Research (CACR), University of Waterloo, August 1999.

[9] H.Khlifi, A.Agarywal and J.Gregoire, ”A Framework To Use SIP in AD-HOC Network” Electrical and Computer Engineering, 2003. IEEE CCECE 2003. Canadian Conference on Volume 2, 4-7 May 2003.

[10] N. Koblitz, “Introduction to Elliptic Curves and Modular Forms, Springer-Verlag”, New York, 1984.

[11] E.T.Lakay and J.I. Agbinya, ”Security issues in SIP signaling in wireless networks and services”, Mobile Business, 2005. ICMB 2005. International Conference on 11-13 July.

[12] J.Manner and K.Raatikinen, ”Research Challenges in IP QoS ans Session Management in Moblie Ad-Hoc Networks”, Wireless Ad-Hoc Networks, 2004 International Workshop on 31 May-3 June 2004.

[13] Miladinovic and J.Stadler, “Sip extension for multiparty conferencing”,

draft-miladinovic-sipmultiparty-ext-01 .at, September 2002.
[14] A. Menezes, Kluwer,“Elliptic Curve Public Key Cryptosystems”, 1993.
[15] K.Ono,S.Tachimoto,”SIP signaling security for end-to-end communication”, Communications, 2003. APCC 2003. The 9th Asia-Pacific Conference on Volume 3, 21-24 Sept. 2003
[16] Christof Paar, “Implementation Options for Finite Field Arithmetic for Elliptic Curve Cryptosystems,” 3rd workshop on Elliptic Curve Cryptography 99, Nov 1999.
[17] M. Rosing, “Implementation Elliptic Curve Cryptography,” Manning Pub Co, 1998.
[18] Stallings,”Cryptiography and Network Security Principles and Practices”,2005,P.324-P.333
[19] ANSI X9.62-1998,”Public Key Cryptography for the Financial Services Industry: TheElliptic Curve Digital Signature Algorithm (ECDSA), 1998.
[20] Sun Microsystem,”The JAIN APIs: Integrated APIs for JAVA Platform”,white Paper,2002
[21] http://jce.iaik.tugraz.at/
[22] http://www.cdc.informatik.tu-darmstadt.de/flexiprovider/overview.html
[23] 楊中皇,“橢圓曲線密碼系統軟體實現技術之探討”, Communications of the CCISA Vol. 11 No. 1 January 2005
[24] 張惟宗,楊中皇,“結合智慧卡的ECDSA數位簽章軟體設計與實現”, 2006 電子商務與數位生活研討會