

SIP 即時傳訊之繞送效能研究

張文萍

國立暨南國際大學 通訊工程研究所
s95325528@ncnu.edu.tw

吳坤熹

國立暨南國際大學 通訊工程研究所
solomon@ipv6.club.tw

摘要

即時傳訊(IM)是人們最常使用的溝通方式，而 SIP (Session Initiation Protocol)被發展出來是為了在網路上提供多媒體傳輸的需求。由於其架構簡單、彈性佳、擴展性高，所以被廣泛應用，而即時傳訊的服務就是其應用之一。但是使用 SIP 傳送即時訊息在效率上卻不佳，因而提出了新的通訊協定—MSRP (The Message Session Relay Protocol)用於點對點傳送即時訊息，改善了 SIP 在提供即時傳訊服務上的缺點。除此之外，穿越 NAT (Network Address Translation)也是 SIP 和 MSRP 所需要解決的問題。本文中將會針對相關的機制與效能分析上一一詳加介紹。

關鍵詞：IM、SIP、MSRP、NAT

Abstract

Instant message is a convenient communication way for people to keep in touch with friends and colleagues. Session Initiation Protocol (SIP) is a protocol proposed to support the multimedia transmission. Because of its low overhead, simple syntax, rich flexibility, and high extensibility, it is widely adopted in building real-time communication services. Instant message (IM) is one example among rich SIP applications. However, using SIP to transmit instant messages is somehow inefficient, based on its request/response model. It is possible to improve SIP to provide the instant message service in a more efficient way by utilizing a recently proposed protocol MSRP (Message Session Relay Protocol). Moreover, Network Address Translation (NAT) traversal is also a problem which SIP and MSRP must solve.

1. 前言

由於數位時代的來臨，造就網際網路的蓬勃發展，上網瀏覽、搜尋資訊、收發電子郵件、以及目前受歡迎的部落格(Blog)，幾乎已經成

為每個人日常生活中的一部份。而 Web2.0 的服務更是增加了人們與網際網路的互動關係。此外，網際網路的發展也改變了人與人之間溝通的方式，從傳統的電話溝通方式，到利用電子郵件網路進行溝通，一直到近幾年深受歡迎和受矚目，並且同時具備即時性和低建置成本等優點的即時通訊服務。市面上即時通訊服務主要有 ICQ、Yahoo 即時通、MSN 以及席捲全球的 Skype。目前越來越多人透過即時訊息(Instant Messaging：IM)做為主要且必備的溝通方式。

一般而言，不論是在有線或是無線的網路環境中，訊息服務(Messaging Service)依架構和傳送技術的差異，可分成兩種型態：(1)先儲存再轉送(Store and Forward)訊息服務，主要以簡訊(SMS)、電子郵件(E-Mail)、多媒體訊息(MMS)為代表。此服務會有一個集中式管理的伺服器，用來接收並儲存傳送端的訊息，對每個已儲存的訊息做處理後，轉送至接收端或是等待接收端來存取，這種服務的特性是在時間上不具即時性；(2)透過伺服器或是點對點(peer-to-peer)的方式來傳送訊息的即時訊息(Real-time)服務，主要以即時訊息(IM)為代表。傳送端的訊息不會被中介設備儲存，而是直接傳送到接收端，若是接收端無法接收訊息或是接收失敗，則立刻告知傳送端此訊息已傳送失敗，因此在時間上具有即時性。

在本文中，我們主要以即時訊息(IM)為重點。首先說明在即時訊息服務中，常見的兩種訊息模式：(1)呼叫模式(page mode)—以 SIP (Session Initiation Protocol [1])通訊協定為代表；(2)會議模式(session mode)—以 MSRP 通訊協定為代表，並且介紹此兩種模式的架構與功能，以及其各自所代表的運作和服務流程。在接下來的文章中，第二節將介紹即時訊息的基本運作方式，第三節與第四節分別介紹 Session Initiation Protocol (SIP) 與 Message Session Relay Protocol (MSRP) 兩套通訊協定下，即時訊息的傳遞方式。第五節比較利用呼叫模式與會議模式的優缺點，第六節敘述網路位址分享器 (Network Address Translation)

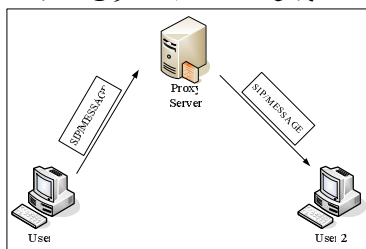
對於 SIP 與 MSRP 兩種即時訊息所造成的困難，以及我們建議的解決之道。結論與未來工作則敘述於第七節與第八節。

2. 即時訊息技術

在即時訊息服務中，目前通用的有兩種訊息模式，亦即「呼叫模式即時訊息」及「會議模式即時訊息」。以下說明此兩種模式的架構及其代表性的通訊協定。

2.1 呼叫模式即時訊息(Page Mode Instant Message)

在這個模式中，每次傳送一筆單一的即時訊息，所傳遞的資料是透過伺服器在使用者之間做即時交換。在交談的過程中，雙方互相傳送的資料通常都經過多次的往來傳遞，而且每次資料量都比較小。基於此模式，SIP 通訊協定在 RFC 3428 中制定了 MESSAGE 延伸命令，用來傳送交談者之間的訊息資料。MESSAGE 命令在表頭(header)欄位中指定接收端的位址，而內文(body)部分則包含想傳送的訊息內容。在使用者之間的完整交談是由許多各自獨立的訊息多次往返而達成。



圖一 呼叫模式即時訊息

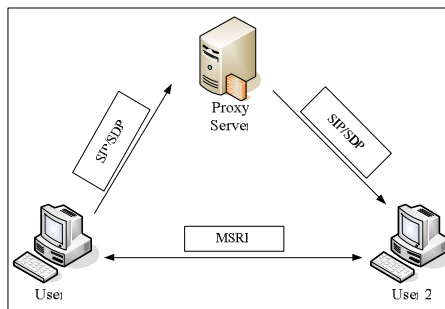
圖一說明當 User1 要傳送即時訊息給 User2 時，和所有的 SIP 命令一樣，User1 必須先將包含即時訊息的 MESSAGE 命令傳送給 Proxy Server，再由 Proxy Server 轉送給 User2；反之，當 User2 要對即時訊息做回覆時，也同樣發出一個包含有交談訊息的 MESSAGE 命令，經由 Proxy Server 轉送給 User1。

2.2 會議模式即時訊息(Session Mode Instant Message)

相對於呼叫模式，會議模式在交談者之間實際存在一個「會議」(Session)的連結。這個連結明確地定義了包含會議起始和結束的步驟及時間點、所有參與交談的成員、欲傳遞的媒體資料。在會議的存活期間，使用者之間所交談的訊息都要依附在這個連結上面做傳送。

在 SIP 運作的環境中，如欲建立一個即時

訊息的會議，必須要送出邀請的請求；如果要終止會議，則要送出結束的請求。此外，在會議中所有參與者之間傳送的即時訊息是根據 SDP (Session Description Protocol [2])所描述的。而在建立起來的會議中，可由 IETF 所提出的 MSRP (The Message Session Relay Protocol [4]) 通訊協定，在使用者之間傳遞即時訊息資料。



圖二 會議模式即時訊息

圖二在說明當 User1 和 User2 彼此要透過 MSRP 傳遞即時訊息時，User1 要先發出 SIP INVITE 的請求，並在 SDP 中包含 User1 的 MSRP 相關資訊，之後將請求經由 Proxy Server 轉送給 User2，接著 User2 回覆 SIP 200OK 的訊息，並在 SDP 中包含 User2 的 MSRP 相關資訊，同樣經由 Proxy Server 回傳給 User1。當雙方溝通並建立連線後，接下來就不必透過 Proxy Server，而可以直接透過 MSRP 傳送即時訊息。在路徑的選擇上，將可有較佳的效率。

3. SIP 通訊協定介紹

會議初始協定(Session Initiation Protocol，簡稱 SIP [1])是由 IETF 工作小組所訂定的，是一種簡單的信號傳輸通訊協定，用於建立、修改和終止多媒體的會議(Session)，如：網際網路會議、網路電話等。在 OSI (Open System Interconnection)七層架構中屬於會議層的控制協定。

SIP 是一個主從式(Client-Server)架構，當用戶端(Client)發出請求(Request)，而伺服器端(Server)收到請求後，會依據情況適當的送出回應(Response)給用戶端。

3.1 SIP 架構

在 SIP 運作的環境中，定義了幾個元件，包含 User Agent(UA)、Location Server、Registrar Server、Proxy Server 和 Redirect Server。以下說明各個元件的基本功能。

User Agent：是一個終端設備，負責產生

請求(Request)，用來建立或終止多媒體會議，且傳送及接收多媒體資料。User Agent 依行為又分成 User Agent Client(UAC)和 User Agent Server(USA)兩種角色。UAC 負責產生請求(Request)以及處理回應(Response)，而 UAS 則是負責接收請求(Request)和產生回應(Response)。通常在一個多媒體會議建立過程中，UA 需要同時扮演這兩種角色。

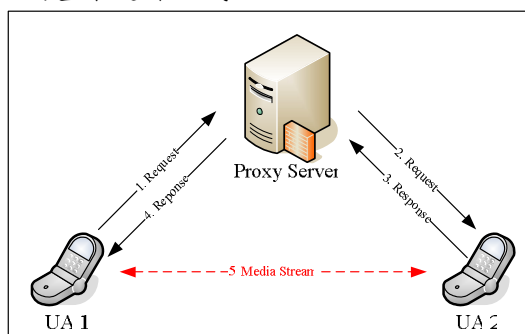
Location Server：即一個資料庫，用來存放使用者的位址資訊，如：URL、IP Address。當 UA 要存取資料時，必須要透過 Proxy Server、Registrar Server 或 Redirect Server 來做存取 Location Server 中的資訊。

Registrar Server：負責接收 REGISTER 請求(Request)，並且更新 UA 在 Location Server 或是其它資料庫的位址資訊。

Proxy Server：負責接收 UA 或是其它 Proxy 發送的請求(Request)，並轉送到其它地方。

Redirect Server：負責接收 UA 或是其它 Proxy 發送的請求(Request)，並且回傳 Redirection 回應(Response)，指出該請求(Request)應重新發送至何處。

在上述伺服器中，本文主要將說明與 Proxy Server 相關之運作。圖三所示為 Proxy Server 的基本運作方式。



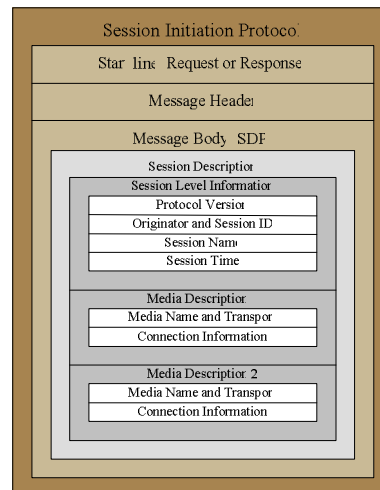
圖三 Proxy Server 運作方式

當 UA1 要邀請 UA2 建立多媒體會議時，UA1 會發出 Request 並傳至 Proxy Server，再由 Proxy Server 將該請求轉送至 UA2，而 UA2 會視情況回覆 Response，同樣經由 Proxy Server 將 Response 轉送給 UA1。此時雙方建立連線，UA1 和 UA2 就可以直接進行多媒體資料的傳輸。

3.2 SIP Message Format

SIP 是利用文字(Text-based)的方式來進行編碼，看起來類似於 HTTP 協定，使得訊息內容的編碼顯得簡單又明瞭，但相對之下自然較耗費頻寬。

SIP message 分成三個部份：第一行用來表示該訊息(message)為 Request 或 Response 的 Start-line、之後是連續數行的 Message Header 用來描述 session 的資訊、最後才是使用 SDP 來描述多媒體資訊的 Message Body。如圖四所示。



圖四 SIP message 架構

3.2.1 SIP Request

在 RFC3261 定義了六種不同的方法，包含 REGISTER(註冊)、INVITE(邀請)、ACK(確認)、BYE(終止)、CANCEL(取消)、OPTIONS(選項)，可以提供使用者發出請求。在表一中列出了這六個方法以及使用該方法的目的。

表一 SIP Request

方法	說明
REGISTER	註冊使用者的位址資訊
INVITE	邀請及建立會議
ACK	對 INVITE 做最終的確認
BYE	結束一個已存在的會議
CANCEL	取消尚未建立的會議
OPTIONS	詢問另一方的支援能力

3.2.2 SIP Response

對於不同的事件或是請求的結果也定義了不同的回應，在表二中列出六種回應訊息的分類以及所代表的意義。例如：200 OK 是表示訊息傳遞成功或是對方接受邀請的回應訊息。

表二 SIP status codes

種類	範例
1XX : Provisional	100 Trying 180 Ringing
2XX : Success	200 OK
3XX : Redirection	301 Moved

	permanently 302 Moved temporarily
4XX : Request Failure	400 Bad request 404 Not found 486 Busy here
5XX : Server Failure	502 Bad gateway 504 Server timeout
6XX : Global Failure	603 Decline 606 Not acceptable

3.2.3 Header Fields

在 SIP 訊息中包含許多不同的 header field，這些 header field 存在 request 或 response 中，用來提供更多有關 SIP 訊息的資訊或是能夠適當的處理 SIP 訊息的能力。在表三中列出了四種不同的 header field 以及其範例。

表 三 SIP Header

種類	範例
General Headers	To：表示被呼叫端的 SIP URI。 From：表示呼叫端的 SIP URI。
Request Headers	Subject：會議的主旨，提供對呼叫的概述或表明呼叫的性質。 Priority：定義本次呼叫的優先權，有“urgent”、“normal”等型式。
Response Headers	Unsupported：列出 Server 端未支援的功能。 Retry-After：當本次呼叫因為沒有足夠的資源而失敗時，在多長的時間內允許 Server 端繼續重試。
Entity Headers	Content-Length：指出 message body 的資料長度。 Content-Type：表示 message body 中所使用的媒體類型。

3.2.4 SDP 通訊協定介紹

會議描述協定(Session Description Protocol，簡稱 SDP [2])也是一個純文字格式的通訊協定，用來描述會議的訊息以及會議參與者的相關資訊。一個會議基本上是由數個多媒體串流所組成，在 SDP 訊息中包括了會議

整體的共通參數，以及許多與每個多媒體串流相關的參數。會議整體的共通參數(Session Level Information)包含 SDP 協定的版本(Protocol Version)、會議的發起人(Originator and Session ID)、會議的名稱(Session Name)、會議的時間(Session Time)等；而媒體的相關參數(Media description)包含媒體的類型(Media Type)、通訊埠號碼(Transport Port)、傳輸協定(Transport Protocol)、媒體的格式(Media Format)等。在圖四中說明了 SDP 的基本架構。

由於 SDP 和 SIP 都是使用文字(Text-based)方式來描述會議的訊息，因此它很容易作為 SIP 的訊息內容(Message Body)來傳送。在表四中列出了數個常用的必要性欄位以及對每個欄位的說明。

表 四 SDP 必要欄位

欄位名稱	說明
v(版本)	描述 SDP 的版本
o(來源)	描述會議的來源或發起人
s(名稱)	描述會議的名稱
t(時間)	描述會議開始和結束的時間
m(媒體)	描述媒體的類型、通訊埠、傳輸協定、媒體的格式

在 SDP 的會議訊息中除了必要的欄位外，還包括了選擇性的欄位。在 SDP 的選擇性欄位中，有一些只能用在 Session Level，而有一些只能用在 Media Level，此外還有一些欄位可同時用在這兩種 Level。在表五中列出了多個選擇性欄位以及針對每個欄位的說明。

表 五 SDP 選擇性欄位

欄位名稱	說明
i(資訊)	描述有關會議的相關資訊，即有註解的意味
u(URI)	定義 Conference URI 或 WWW，可透過此欄位得到更多會議的訊息
e(e-mail)	以 e-mail 的型式描述欲連線的對象
p(電話號碼)	以電話號碼的型式描述欲連線的對象
c(連線)	描述本地端的連線資料，包括連線的類型、網路的類型、連線的位址
b(頻寬)	描述頻寬的狀況，單位是 kbps
k(加密)	描述一個加密的鑰匙或是取得鑰匙的機制
a(屬性)	描述媒體編碼型式和相關支援能力的參數

3.3 SIP 的運作方式

當使用者要建立 SIP 連線時，呼叫端要先使用 INVITE 方法，然後透過適當的 Proxy Server 轉送出去，而不是直接與被呼叫端連繫；反之，當使用者想要終止連線時，則必須使用 BYE 方法，同樣透過 Proxy Server 將請求轉送給對方。

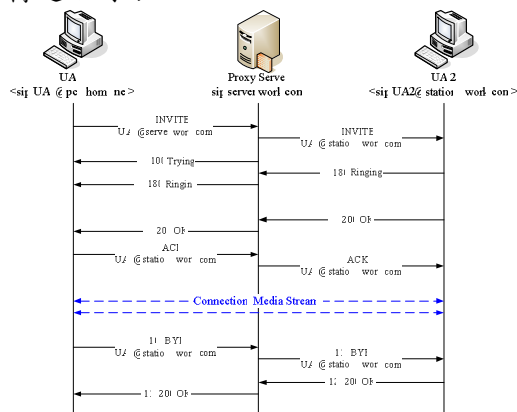


圖 五 SIP 建立與終止會議的流程

在圖五中說明了經由代理伺服器(Proxy Server)建立與終止 SIP 會議的運作流程，其步驟如下：

- (1) 首先 UA1 發出 INVITE 請求對 UA2 送出邀請。
- (2) 代理伺服器(Proxy Server)收到 INVITE 請求後，會將該請求轉送給 UA2。
- (3) 當代理伺服器(Proxy Server)將 INVITE 請求轉送出去後，會回覆 100 Trying 訊息給 UA1，告知 UA1 該請求正試著在處理中。
- (4) 當 UA2 發出響鈴後，會回覆 180 Ringing 訊息。
- (5) 代理伺服器(Proxy Server)收到 180 Ringing 訊息後，會將該訊息轉送給 UA1。
- (6) 當 UA2 接受 UA1 的邀請後，會回覆 200 OK 訊息。
- (7) 代理伺服器(Proxy Server)收到 200 OK 訊息後，會將該訊息轉送給 UA1。
- (8) UA1 收到 200 OK 訊息後，可以知道對方接受邀請，並發出 ACK 請求做最後的確認。
- (9) 代理伺服器(Proxy Server)收到 ACK 請求後，會將該請求轉送給 UA2。當 UA2 收到最後的確認後，雙方完成呼叫連線，就可以開始互相傳遞媒體資料流。
- (10) 當 UA1 要結束連線時，會發出 BYE 請求給 UA2。
- (11) 代理伺服器(Proxy Server)收到 BYE 請求後，會將該請求轉送給 UA2。

(12) UA2 收到結束訊息後，會回覆 200 OK 訊息給 UA1，表示同意結束連線。

(13) 代理伺服器(Proxy Server)收到 200 OK 訊息後，會將該訊息轉送給 UA1。UA1 收到該訊息後，雙方完成結束連線。

3.4 SIP and Instant Message

目前 IETF 組織為了在即時訊息上採用 SIP 作為其標準化的制定與討論，成立了 SIMPLE(SIP for Instant Message and Presence Leveraging Extensions)工作小組，除了制定即時訊息標準外，還包括線上狀態服務的通訊協定。在即時訊息制定上，已經制定完成並且成為 RFC 文件的，像是 RFC 3428 制定了 SIP 的延伸訊息 MESSAGE，在不改變 SIP 的任何架構下，提供呼叫模式即時訊息(Page Mode IM)的即時訊息服務。

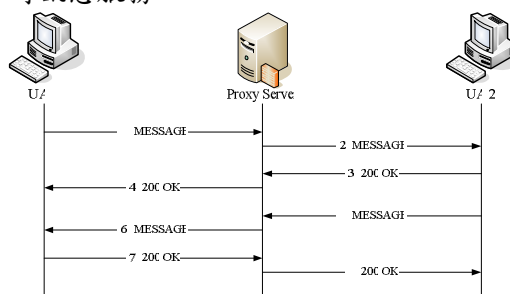


圖 六 SIP-MESSAGE 即時傳訊的流程

如圖六所示，只能單一使用者傳送訊息給單一使用者，無法建立會議(Session)一次傳至多人。而且所有的訊息都是由使用者先傳給代理伺服器(Proxy Server)然後再轉送給對方。

4. MSRP 通訊協定介紹

訊息會議中繼協定(Message Session Relay Protocol, 簡稱 MSRP [4])提供會議模式即時訊息(Session Mode IM)的即時訊息服務。在利用 SIP 建立語音連線時，是透過 INVITE 請求，利用 SDP 交換媒體的相關資訊，最後建立 RTP 通訊協定的連線，語音封包就是在 RTP 連線上面做傳輸；同樣地，要建立 MSRP 連線時，也可透過 INVITE 請求，藉由 SDP 來協議雙方的資訊。而 RTP 與 MSRP 的差別在於所建立的連線，一是即時影音，一是即時訊息。為了能夠支援 MSRP，SDP 必須新增一些欄位。

```

Session Initiation Protocol
Request-Line: INVITE sip:2270906125.231.3.80:18826;rinstance=ff1e977402344b49 SIP/2.0
Message Header
Message body
Session Description Protocol
Session Description Protocol version (v): 0
Owner/Creator: Session id (o): 22709 123456 654321 IN IP4 125.231.6.234
Session Name (s): A conversation
Connection Information (c): IN IP4 125.231.6.234
Time Description, active time (t): 0 0
Media Description, name and address (m): audio 7088 RTP/AUP 111 110 0 3 8 101
Media Attribute (a): rtpmap:111 speex/16000/1
Media Attribute (a): rtpmap:110 speex/8000/1
Media Attribute (a): rtpmap:0 PCMU/8000/1
Media Attribute (a): rtpmap:3 G729/8000/1
Media Attribute (a): rtpmap:8 PCMA/8000/1
Media Attribute (a): rtpmap:101 telephone-event/8000
Media Attribute (a): rtpmap:101 0-11
Media Description, name and address (m): message 10000 TCP/MSRP
Media Attribute (a): accept-type:text/plain
Media Attribute (a): path=msrp:/125.231.2.125:10000/xxxxxxxxxxxxx:tcp

```

圖 七 支援 MSRP 的 SDP 欄位

如圖七為 SIP INVITE 要求中所包含的 SDP 訊息，其中虛線框的部份是 SDP 為了支援 MSRP 所新增的欄位，包括：以‘m=’為起始的 media-line 新增‘message TCP/MSRP*’，表示 SDP 是在 TCP 上建立 MSRP 的參數協商；‘a=accept-type’欄位用來表示支援的 MIME 類型；‘a=path’欄位用來做 MSRP URL 的協商，交換彼此的 MSRP URL，表示 MSRP 欲連線的位置。

4.1 MSRP Message

MSRP 是以文字(Text-based)的方式做編碼且為連接導向(connection-oriented)的通訊協定。傳送的内容可以是任意的 MIME 格式，尤其是 Instant Message(IM)。在 MSRP 中制定了兩種請求(request)，分別是 SEND、REPORT。

4.1.1 MSRP SEND Request

用來傳送完整的訊息或是訊息中的一部份。

```

MSRP: 786hjs2 SENI
To-Path: msrp://biloxi.example.com:12763/kjhd37s2s20w2a.tcp
From-Path: msrp://atlanta.example.com:7654/jshA7weztas.tcp
Message-ID: 87652491
Byte-Range: 25/25
Success-Report: yes
Failure-Report: no
Content-Type: text/plain

Hey Bob, are you there?
----- a786hjs23

```

圖 八 MSRP SEND 訊息內容

圖八說明了 SEND Request 的訊息內容，其中 To-Path 和 From-Path 分別是 SDP 訊息中雙方所使用的 MSRP URL，可用來識別一個 MSRP 訊息所屬的會議；Message-ID 用來識別同一個會議中各個 MSRP 訊息；Success-Report 和 Failure-Report 用來標明是否要接受 Success report 和 Failure report；Connect-Type 用來表示該 MSRP 訊息所屬的 MIME 型態。

4.1.2 MSRP REPORT Request

用來傳達先前被傳送的訊息狀態，當 SEND request 的訊息中含有 Success-Report 或

Failure-Report 時，REPORT request 就會被送出。

```
MSRP dkei3fsd REPORT
```

```

To-Path: msrp://atlanta.example.com:7654/jshA7weztas.tcp
From-Path: msrp://biloxi.example.com:12763/kjhd37s2s20w2a.tcp
Message-ID: 87652491
Byte-Range: 1-25/25
Status: 000 200 OK
----- dkei3fsd

```

圖 九 MSRP REPORT 訊息內容

圖九說明了 Report Request 的訊息內容，其中 Status 用來標明 SEND request 的回覆代碼。

4.2 SIP and MSRP 運作方式

當使用者要建立 MSRP 連線時，呼叫端要先使用 INVITE 方法，且在 SDP 內容中含有 MSRP 的相關參數，然後透過 Proxy Server 將請求轉送出去；反之，當使用者想要終止連線時，則與一般通話的終止相同，使用 BYE 方法，同樣透過 Proxy Server 將請求轉送給對方。

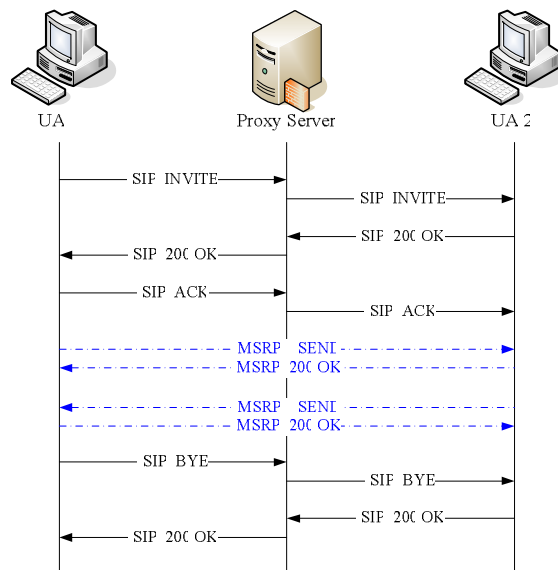


圖 十 SIP 搭配 MSRP 的運作流程

在圖十中說明了使用 SIP 來建立與終止 MSRP 連線，其步驟如下：

- (1) 首先 UA1 發出 INVITE 對 UA2 送出 MSRP 連線的邀請。
- (2) 代理伺服器(Proxy Server)收到 INVITE 後，會將該訊息轉送給 UA2。
- (3) 當 UA2 接受 UA1 的邀請後，會回覆 200 OK，UA1 收到回覆後，可以知道對方接受邀請。
- (4) 此時，UA1 會再發出 ACK 給 UA2 做最後的確認。
- (5) UA2 收到最後的確認後，雙方就可建立

MSRP 連線，開始利用 MSRP 互相傳遞即時訊息。

- (6) 當 UA1 要結束 MSRP 連線時，會發出 BYE 給 UA2。
- (7) UA2 收到結束訊息後，會回覆 200 OK 給 UA1，表示同意結束連線。
- (8) UA1 收到回覆後，雙方完成結束 MSRP 連線。

5. SIP/MESSAGE 和 MSRP 比較

在上述中，介紹由呼叫模式傳送即時訊息的 SIP/MESSAGE，此模式要透過中介設備做轉送且每次傳送的訊息都是各自獨立，前後並無關聯性；而以會議模式傳送即時訊息的 MSRP，則是在使用者之間建立一個會議(session)的連結，以在其間傳遞訊息。兩種模式的優缺點比較如下。

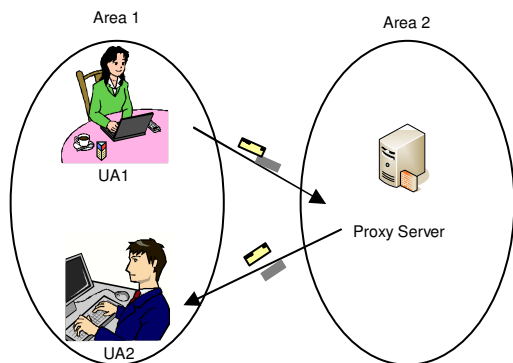


圖 十一 即時訊息的路由

如圖十一所示，UA1 和 UA2 都已經各自向 Proxy Server 完成註冊請求。假使 UA1 和 UA2 同時在同一個地區(Area 1)中，而 Proxy Server 卻是在另一個地區(Area2)，當 UA1 和 UA2 要使用呼叫模式傳送即時訊息時，都必須先將訊息送給 Proxy Server，再由 Proxy Server 轉送給對方。如此一來，由於繞送路徑較長，易造成封包遺失而且封包路由的路徑也不是最佳化；反之，用會議模式傳送即時訊息，可以讓使用者的訊息循點對點的最佳路徑傳送，同時還可以減輕 Proxy Server 的負擔。

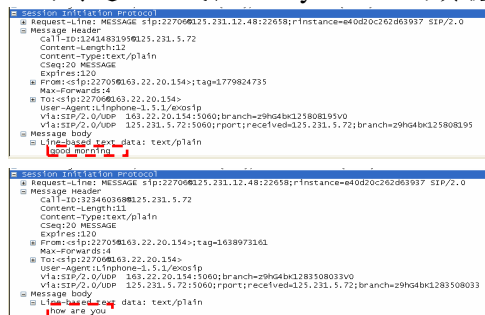


圖 十二 SIP/MESSAGE 訊息內容

另外，如圖十二所示，這兩個訊息內容都

是由同一個使用者用 SIP/MESSAGE 以呼叫模式來傳送即時訊息給同一個對象。其中這兩個訊息內容的差別只在虛線框的部份，即文字訊息內容的不同，而在 SIP header 的部份都是相同的。以這兩個訊息內容而言，傳送一個 12Bytes 的訊息，其 SIP header 的長度是 353Bytes，將近是所欲傳送的即時訊息的 30 倍。這是相當高的額外負擔(overhead)！因此，若用會議模式傳送即時訊息，就可以大大減少頻寬浪費來傳送這些非訊息的額外負擔。

6. NAT

在 Internet 網路中，為了解決 IPv4 位址不足的問題，經常會建立網路位址轉換(Network Address Translation, 簡稱 NAT [5])機制。在內部網路中使用私有網路位址(Private IP address)，通過 NAT 時對外通訊時，則把內部位址轉換成公開的網路位址(Public IP address)在網際網路上使用

NAT 主要分成三種類型：靜態 NAT、動態 NAT、網路位址通訊埠轉換(Network Address Port Translation, 簡稱 NAPT)。

靜態 NAT：其架構最簡單，其內部網路中每個主機都會對應到外部網路中的公開網路位址。

動態 NAT：在外部網路中定義一連串的公開網路位址，採用動態分配的方法映射到內部網路中的主機。

網路位址通訊埠轉換(NAPT)：讓內部網路的主機共用同一個對外的公開網路位址並以不同通訊埠來做區別。

其中網路位址通訊埠轉換(NAPT)是最常被使用的一種轉換方式，在本論文中也是使用此類型，而依照選擇通訊埠的不同又有四種分別：

- (1) Full Cone NAT：即一個內部的 IP 位址對應到同一個外部的 IP 位址和通訊埠，因此可藉由此外部的 IP 位址和通訊埠對內部的 IP 位址進行溝通。

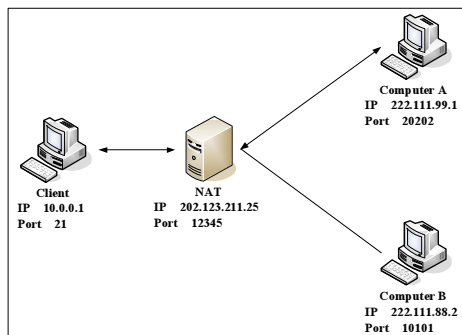


圖 十三 Full Cone 機制

如圖十三所示，當 client 送封包給 computer A 時，NAT 會分配 port 12345 給 port 21。此時，所有來自外界的網路封包，不論是來自 computer A 或 computer B，只要目的通訊埠是 NAT 的 port 12345，都會經由 NAT 送給 port 21。

(2) Restricted Cone NAT：即一個內部的 IP 位址對應到同一個外部的 IP 位址和通訊埠，但是內部的 IP 位址必須先對目的端傳送資料，目的端才可以藉由外部的 IP 位址和通訊埠與內部的 IP 位址做溝通。

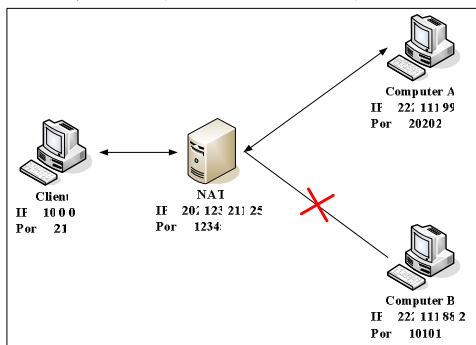


圖 十四 Restricted Cone 機制

如圖十四所示，當 client 送封包給 computer A 時，NAT 會分配 port 12345 給 port 21。此時，來自 computer A 進入 NAT 的封包，都會經由 port 12345 送給 port 21，但是 computer B 所傳送的封包就會被 NAT 給丟棄。

(3) Port Restricted Cone NAT：和 Restricted Cone NAT 的差別在於除了對外部的 IP 位址有限制外還加上通訊埠的限制。

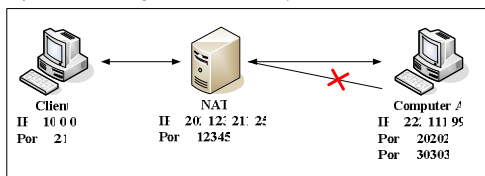


圖 十五 Port Restricted Cone 機制

如圖十五所示，當 client 送封包給 computer A 的 port 20202 時，NAT 會分配 port 12345 給 port 21。此時，來自 computer A 的 port 20202 進入 NAT 的封包，都會經由 port 12345 送給 port 21；但是 computer A 的 port 30303 所傳送的封包則會被 NAT 給丟棄。

(4) Symmetric NAT：和前面兩種 NAT 最大差別就是，隨著目的端的 IP 位址和通訊埠的不同，所對應的外部 IP 位址和通訊埠就會不同。

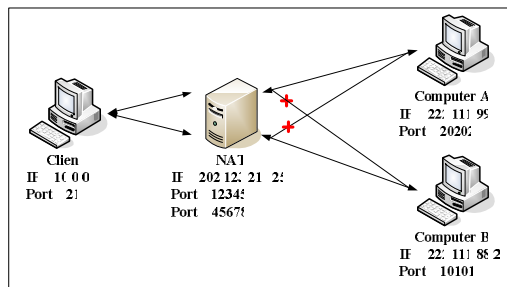


圖 十六 Symmetric NAT 機制

如圖十六所示，當 client 送封包給 computer A 的 port 20202 時，NAT 會分配 port 12345 給 port 21；送封包給 computer B 的 port 10101 時，NAT 會分配 port 45678 給 client 的 port 21。此時，來自 computer A (port 20202) 送到 port 12345 的封包，可經由 NAT 送給 port 21；但是由 computer B (port 10101) 所傳送到同樣 port 12345 的封包，則會被 NAT 丟棄。Computer B 必須送到另一個專屬的 port 45678，才可經由 NAT 正確地送給 client。

6.1 SIP 協定穿越 NAT 的問題

雖然 NAT 在舒緩網路位址不足的問題上頭，帶來許多的好處以及幫助，但是也造成了一些問題。其主要的問題是內部主機使用私有網路位址 (Private IP address)，使得某些應用程式無法成功與外部網路建立連線。例如 SIP 在經過網路位置轉換 (NAT) 時，會發生無法穿透 NAT 的問題，主要的麻煩在於：

- (1) SIP 和 SDP 的訊息內容中都包含自己的網路位址，而這個網路位址在經過網路位置轉換 (NAT) 後，可能已經無法存取或不存在對應，使得外部網路無法回應。
- (2) SIP 通常使用特定的通訊埠 (5060)，但 RTP 所使用的通訊埠是由雙方 SDP 動態溝通決定，無法事先設定。

目前有許多穿越 NAT 的方法被提出，其中 STUN、TURN 都必須有 SIP 設備的支援，才可以順利運作。

6.1.1 STUN

Simple Traversal of UDP Through NAT，簡稱 STUN [7]，為 Client/Server 架構的通訊協定。其中 client 是位於 NAT 內部的 SIP UA，server 則是放在外部網路上。當 client 要知道對應的外部 IP 位址與通訊埠時，會用相同的來源 IP 位址和通訊埠送封包給 STUN server，由 server 告知此封包所對應的外部 IP 位址與通訊埠，透過此協定可讓 SIP UA 知道 NAT 所分配的對外 IP 位址和通訊埠。

STUN 訊息可分為 Binding Request 和 Binding Response 兩種。其中 Binding Request 是要求知道內部 client 對外的 IP 位址和通訊埠；而 Binding Response 是由 server 回應，告知內部 client 對外的 IP 位址和通訊埠。

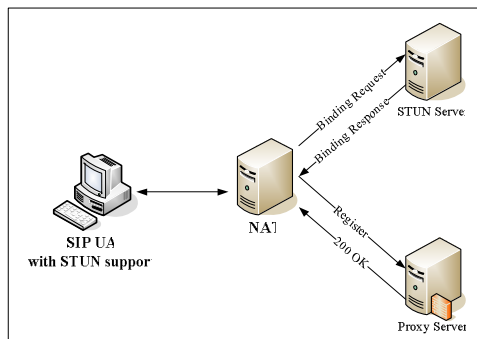


圖 十七 SIP/Register 運作流程

如圖十七所示，當內部 SIP UA 要發出 SIP REGISTER 訊息給 Proxy Server 做註冊請求時，要以 STUN 機制先發出 Binding Request 的訊息；STUN Server 收到後，會回應 Binding Response 的訊息，告知內部 SIP UA 對外的 IP 位址和通訊埠。內部 SIP UA 知道後，會送出 REGISTER 的請求訊息給 SIP Server，而且在訊息內的 Contact 欄位會被修改為其外部 IP 位址與通訊埠。

6.1.2 TURN

Traversal Using Relay NAT，簡稱TURN [8]，是使用Relay的方式穿越NAT，其方法和STUN相似，都是讓內部網路的SIP UA知道對外的IP位址與通訊埠。差別在於TURN所得到的是TURN Server的IP位址與通訊埠，因此內部SIP UA所發出的封包都要經由TURN Server轉送出去，這種方式可以解決STUN無法穿越Symmetric NAT的問題。

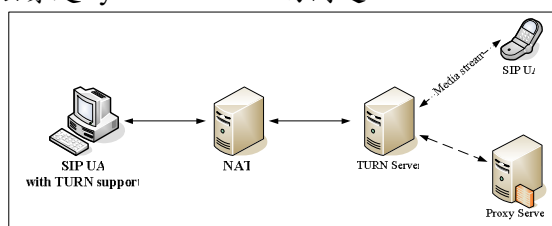


圖 十八 SIP/Register 運作流程

如圖十八所示，當內部 SIP UA 要送出封包時，要先和 TURN Server 建立連線。之後要送到外部的封包都會經由 TURN Server 轉送出去，而外部的封包也會先送給 TURN Server，再由 TURN Server 將封包轉送給內部 SIP UA。

6.2 MSRP 解決穿越 NAT 的問題

由於 MSRP 也是藉由 SIP 來建立連線，因此上述 SIP 無法穿透 NAT 的問題，在此也會發生。而在這裡我們提出搭配 MSRP 使用的，也是新的穿越 NAT 的技術—ICE。

6.2.1 ICE

互動式連線建立(Interactive Connectivity Establishment，簡稱ICE [9])是由 IETF 的 MMUSIC 工作小組所開發的。它並不是一個單一的通訊協定，而是一個框架，將各種穿越 NAT 的技術適切地整合在一起。

ICE 定義的是一種標準化的方法，使 SIP UA 能夠確定雙方之間存在的是哪一種類型的 NAT，並且確定一連串可用於連線的 IP 位址。

當一個具備 ICE 功能的來源端希望與目的端建立連線時，首先會先利用 STUN 與 TURN 等機制，從來源端收集多個可能連線的 IP 位址，並在本地對那些位址進行配置，選擇其中可以向來源端成功建立 IP 連線的位址。ICE 的優點就是能夠統一這些連線的 IP 位址，並盡可能以最佳路徑建立至目的端的連線。

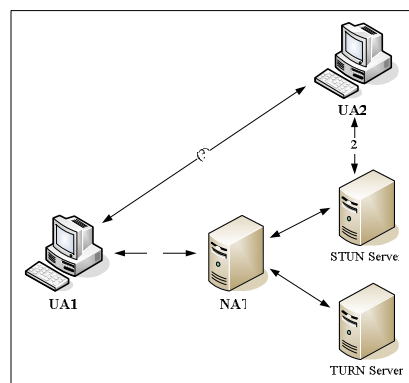


圖 十九 ICE 運作流程

在圖十九中說明了 ICE 要在雙方建立連線的運作流程，其步驟如下：

- (1) UA1 會透過 STUN 或 TURN 來收集所有可能連線的 IP 位址。
- (2) 接著 UA1 會送一份已排序的位址表給 STUN Server，然後 STUN Server 會依照位址表傳送訊息給 UA2。
- (3) UA2 會把訊息中每一個 IP 位址發出 STUN 請求給 STUN Server。
- (4) UA1 收到 STUN 請求後，會回覆訊息給 UA2。
- (5) UA2 收到回覆後，會從這些 IP 位址中找出 UA1 和 UA2 可以建立連線的位址。
- (6) 最後利用位址表中排序最高的 IP 位址來進行連線。

7. 結論

在本篇論文中，我們介紹了即時傳訊常見的兩種模式：呼叫模式(page mode)、會議模式(session mode)。由於以 SIP/MESSAGE 進行呼叫模式時，傳送即時訊息必須經由中介設備做轉送，路徑經常會較長，這方式容易使封包遺失且在效能上也較差。我們在本文中分析了新的通訊協定 MSRP，以會議模式在傳送端和接收端之間直接做即時傳訊，改善了呼叫模式的缺點外，還大大減少不必要的 header overhead。

以 SIP 為基礎的即時傳訊在穿越 NAT 的問題上，已具有一些解決方法，其中 STUN 的技術可用來解決前三種 NAT 類型：Full Cone、Restricted Cone、Port Restricted Cone。為了解決 STUN 無法穿越 Symmetric NAT 的問題，提出了新的技術—TURN。

而針對 MSRP 在穿越 NAT 的解決方法，我們在本文中提出應予採用 ICE，結合 STUN 和 TURN 選用最佳的連線方式以傳送即時訊息。

8. 未來研究方向

目前市面上的 SIP UA 都只支援呼叫模式即時傳訊—SIP/MESSAGE。主要的原因在於未能解決穿越 Symmetric NAT 的問題，因此以較簡單的呼叫模式，可以確保中介設備與使用者間的即時訊息傳送可順利穿越 NAT。在未來的研究主題上，希望透過 Open Source 的 SIP UA，在當中將加入 MSRP 通訊協定，擴大其應用性，讓使用者能夠選擇要使用會議模式或呼叫模式來傳送即時訊息。此外，還要加上 ICE 技術，用來解決 MSRP 穿越 NAT 的問題。

參考文獻

- [1] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley, E. Schooler, “SIP : Session Initiation Protocol”, Network Working Group, RFC 3261, IETF, June 2002.
- [2] M. Handley, V. Jacobson, C. Perkins, “SDP : Session Description Protocol”, Network Working Group, RFC 4566, IETF, July 2006.
- [3] J. Rosenberg, H. Schulzrinne, “An Offer/Answer Model with the Session Description Protocol (SDP)”, Network Working Group, RFC 3264, IETF, June 2002.
- [4] B. Campbell, R. Mahy, C. Jennings, “The Message Session Relay Protocol (MSRP)”, Network Working Group, RFC 4975, IETF, September 2007.
- [5] K. Egevang, P. Francis, “The IP Network Address Translator (NAT)”, Network Working Group, RFC 1631, IETF, May 1994.
- [6] P. Srisuresh, M. Holdrege, “IP Network Address Translator (NAT) Terminology and Considerations”, Network Working Group, RFC 2663, IETF, August 1999.
- [7] J. Rosenberg, J. Weinberger, C. Huitema, R. Mahy, “STUN - Simple Traversal of User Datagram Protocol (UDP) Through Network Address Translators (NATs)”, Network Working Group, RFC 3489, IETF, March 2003.
- [8] J. Rosenberg, C. Huitema, R. Mahy, “Traversal Using Relay NAT (TURN)”, MIDCOM, draft-rosenberg-midcom-turn-08, IETF, September 2005.
- [9] J. Rosenberg, “Interactive Connectivity Establishment (ICE): A Protocol for Network Address Translator (NAT) Traversal for Offer/Answer Protocols”, MMUSIC, draft-ietf-mmusic-ice-19, IETF, October 2007.
- [10] 何哲勳、沈婉婉、沈惠如, “下世代 IP 多媒體子系統之即時訊息服務技術”, 電腦與通訊, 117 期, 2006/09
- [11] 蘇暉凱、姚志臻、劉柏廷、陳景章, “SIP VoIP 電話總機系統於 NAT 閘道器之設計與實作”, 2004 台灣網際網路研討會論文發表論文集
- [12] 趙啟時、呂思霈、黃柏鈞, “建立及分析 SIP-bases VoIP 穿透 NAT 機制”, TANET 2006 台灣網際網路研討會論文集