

國立交通大學

資訊工程系

碩士論文

SIPv6 即時協定分析測試系統的設計與實作
Design and Implementation of a SIPv6 Real-time
Protocol Analysis/Testing System

The logo of National Tsing Hua University is a circular seal. It features a central emblem with the letters 'ES' and 'A' and the year '1896' below it. The seal is surrounded by a blue border with Chinese characters.

研究生：賴健利

指導教授：林一平 教授

陳懷恩 教授

中華民國九十四年六月

SIPv6 即時協定分析測試系統的設計與實作

Design and Implementation of a SIPv6 Real-time Protocol Analysis/Testing System

研 究 生：賴健利

Student : Jain-Li Lai

指導教授：林一平 博士

Advisor : Dr. Yi-Bing Lin

陳懷恩 博士

Advisor : Dr. Whai-En Chen

國 立 交 通 大 學

資 訊 工 程 系

碩 士 論 文

A Thesis

Submitted to Department of Computer Science and Information Engineering

College of Electrical Engineering and Computer Science

National Chiao Tung University

in partial Fulfillment of the Requirements

for the Degree of

Master

in

Computer Science and Information Engineering

June 2005

Hsinchu, Taiwan, Republic of China

中華民國九十四年六月

SIPv6 即時協定分析測試系統的設計與實作

學生：賴健利

指導教授：林一平 博士

陳懷恩 博士

國立交通大學資訊工程學系碩士班

中文摘要

*Voice over IP (VoIP)*為第三代行動通訊 *IP Multimedia Subsystem (IMS)* 中最重要的服務之一。在 IMS 中，VoIP 分別以 *Session Initiation Protocol (SIP)*與 *Real-time Transport Protocol (RTP)* 協定來傳輸信令和多媒體資訊。此外，IMS 也採用 *IP version 6 (IPv6)*來提供大量的位址空間、*Quality of Service (QoS)*，以及隨插即用的功能。因此在建置 IMS 應用服務時，非常需要使用分析器來研究相關通訊協定。常見的 *Ethereal* 是一個開放原始碼的協定分析器，且能夠解析六百多種通訊協定標頭，但 *Ethereal* 並沒提供針對 IMS 分析的專屬功能。因此本論文針對 IMS 的應用服務，設計並實作出 *SIPv6 Analyzer*。*SIPv6 Analyzer* 不但能夠解析通訊協定標頭，也能用來分析 SIP 對話、產生 SIP 信令流程圖、播放 RTP 語音串流，並呈現封包統計功能。綜上所述，本論文所研發的 *SIPv6 Analyzer* 能有效地協助 IMS 應用服務的研發與建置。

Design and Implementation of a SIPv6 Real-time Protocol Analysis/Testing System

Student: Jain-Li Lai

Advisors: Dr. Yi-Bing Lin
Dr. Whai-En Chen

Department of Computer Science and Information Engineering
National Chiao Tung University

Abstract

Among the *IP Multimedia Subsystem* (IMS) services, *Voice over IP* (VoIP) is one of the most important services. IMS utilizes *Session Initiation Protocol* (SIP) and *Real-time Transport Protocol* (RTP) to transfer the VoIP signaling and multimedia information. *IP version 6* (IPv6) is also employed in the IMS to provide large address space and new features including security, *Quality of Service* (QoS), and Plug-and-play. During the IMS service deployment, it is essential to utilize an analyzer to debug and investigate the IMS-related protocols. Ethereal is an open-source analyzer that can dissect more than 600 protocol headers. However, Ethereal does not provide specific functions for IMS services. To provide the desirable functions for the IMS services, this paper designs and implements a *SIPv6 Analyzer*. The SIPv6 Analyzer not only dissects the protocol headers but also provides the SIP dialog, SIP message flow, RTP replay and statistic functions for IMS services. Therefore, the SIPv6 Analyzer can help to develop and deploy the IMS services.

誌謝

首先我要感謝林一平博士與陳懷恩博士。沒有他們的細心指導與專業建議，我無法獨立完成此篇碩士論文。在林一平博士嚴格的指導中，讓我學習到論文寫作的嚴謹與做學問研究的方法。在陳懷恩博士的指導中，讓我獲得論文方向、許多專業知識與寫作技巧。另外感謝賴薇如博士與逢愛君博士兩位口試委員的辛勞。

我也要感謝翁瑞鴻與蘇家永兩位學弟在程式系統開發上的支援。沒有他們的協助開發，程式系統的功能特色不會如此豐富。另外感謝宋岳鑫學弟在後續的程式系統之維護與功能擴充。



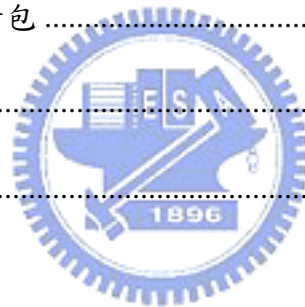
最後感謝我的家人與朋友在論文寫作的期間給予我鼓勵與支持。

目錄

中文摘要.....	i
Abstract.....	ii
誌謝.....	iii
目錄.....	iv
圖片目錄.....	vii
表目錄.....	x
一、論文動機.....	1
1.1 IPv4/IPv6 協定與標頭格式.....	1
1.2 SIP/SDP 協定與標頭格式.....	4
1.3 RTP 協定與標頭格式.....	7
二、SIPv6 即時協定分析系統架構.....	9
2.1 系統運作範例.....	10
三、實作原理.....	14
3.1 解封裝模組.....	14
3.2 Protocol Viewer 模組.....	16
四、SIPv6 即時協定分析測試系統安裝操作說明.....	18
4.1 系統安裝移除與啟動.....	18
4.1.1 軟硬體建議需求.....	18
4.1.2 系統安裝.....	18

4.1.3 系統啟動.....	20
4.1.4 系統移除.....	20
4.2 系統操作說明.....	20
4.2.1 系統 GUI 架構	20
4.2.2 系統選單與系統工具列功能.....	21
4.2.3 分析專案子視窗.....	22
4.2.4 分析專案子視窗之工具列控制項.....	22
4.2.5 分析專案模式.....	25
4.2.5.1 本地端分析.....	25
4.2.5.2.遠端分析.....	27
4.2.5.3 檔案分析.....	28
4.2.6 分析專案功能.....	29
4.2.6.1 封包解析 (Packet Viewer)	29
4.2.6.2.SIP 封包分析 (SIP Viewer)	30
4.2.6.3 RTP 監控與播放 (RTP Viewer)	32
4.2.6.4 IUA/M2UA/M3UA 信令流程圖.....	33
4.2.6.5 流量與通訊協定統計 (Statistic)	33
五、展示案例.....	35
5.1 SIP/IPv6 封包分析	35
六、結論與未來工作.....	39
參考文獻.....	40
附錄 A 系統需求與規格.....	42
A.1 系統需求.....	42

2 系統規格.....	42
附錄 B 6to4 通道與 Teredo 協定技術	43
B.1 6to4 通道.....	43
B.2 Teredo.....	45
附錄 C SIGTRAN 標準.....	47
C.1 SCTP	47
C.2 IUA/M2UA/M3UA.....	48
附錄 D 封包產生精靈.....	49
D.1 送出已擷取的封包.....	49
D.2 Step-by-step 產生封包	50
D.3 使用樣板產生封包.....	52
附錄參考文獻.....	53



圖片目錄

圖 1-1 IPv4 標頭格式圖.....	2
圖 1-2 IPv6 標頭格式圖.....	3
圖 1-3 SIP 協定運作範例圖	4
圖 1-4 SIP 信令範例圖	5
圖 1-5 RTP 標頭格式圖.....	7
圖 2-1 SIPv6 即時協定分析系統架構圖與運作範例圖	9
圖 3-1 解封裝模組架構圖	14
圖 3-2 通訊協定描述子的型態定義圖.....	15
圖 3-3 IPv6 通道封包的解封裝運作流程範例圖.....	15
圖 3-4 Protocol Viewer 模組架構圖.....	16
圖 4-1 系統安裝步驟流程圖	19
圖 4-2 系統啟動與系統移除選單圖	20
圖 4-3 系統 GUI 架構畫面圖	21
圖 4-4 系統選單與系統工具列畫面圖	21
圖 4-5 分析專案子視窗的子頁面畫面圖	22
圖 4-6 分析專案子視窗的顯示/切換控制項畫面圖	22
圖 4-7 分析專案子視窗之工具列控制項畫面圖	23
圖 4-8 儲存封包檔案的展示畫面圖	23

圖 4-9 功能選項對話盒的展示畫面圖	24
圖 4-10 過濾器對話盒畫面圖	25
圖 4-11 啟動本地端分析的選項按鈕圖	25
圖 4-12 本地端分析對話盒畫面圖	26
圖 4-13 遠端封包抓取服務狀態展示圖	27
圖 4-14 啟動遠端分析的選項按鈕圖	27
圖 4-15 遠端分析對話盒畫面圖	28
圖 4-16 啟動檔案分析的選項按鈕圖	28
圖 4-17 封包解析子頁面畫面圖	29
圖 4-18 SIP 封包解析子頁面畫面圖	30
圖 4-19 展開 SIP 信令內容範例圖	30
圖 4-20 SIP 信令流程圖選單圖	31
圖 4-21 SIP 信令流程圖範例	31
圖 4-22 RTP 監控與播放子頁面畫面圖	32
圖 4-23 IUA/M2UA/M3UA 信令流程圖	33
圖 4-24 流量與通訊協定統計子頁面畫面圖	33
圖 4-25 主機封包分佈長條圖	34
圖 5-1 SIP/IPv6 封包分析展示案例的環境架構圖	35
圖 5-2 遠端封包抓取服務圖	35
圖 5-3 Bob 端進行遠端分析圖	36
圖 5-4 SIP Proxy Server 與 SIP Proxy Server 設定圖	36
圖 5-5 SIP Viewer 子頁面與 SIP 信令流程圖輸出圖	37

圖 5-6 RTP Viewer 子頁面圖	38
圖 B-1 6to4 架構圖	43
圖 B-2 6to4 運作範例圖	44
圖 B-3 Teredo 架構圖	45
圖 B-4 Teredo 封裝格式圖	45
圖 B-5 Origin Indication 與 Authentication 標頭格式圖	46
圖 C-1 PSTN-based Signaling Over IP 的通訊協定堆疊圖	47
圖 C-2 SCTP 封包格式圖	48
圖 D-1 封包產生器精靈畫面圖	49
圖 D-2 送出已擷取封包的步驟流程圖	50
圖 D-3 TCP/IPv6 測試封包的建構步驟流程圖	51
圖 D-4 ARP 測試封包的建構步驟流程圖	52



表目錄

表 1-1 IPv4 標頭欄位說明表.....	2
表 1-2 IPv6 標頭欄位說明表.....	3
表 1-3 SIP 標頭欄位說明表	6
表 1-4 SDP 型態說明表	6
表 1-5 RTP 標頭欄位說明表.....	8
表 3-1 Windows Multimedia API 列表.....	17
表 4-1 軟硬體建議需求表	18
表 C-1 SCTP Chunk 功能說明表	48



一、論文動機

第三代 (3G) 行動通訊全 IP (All-IP) 核心網路之 IP Multi-media Subsystem (IMS) [1] 系統所採用的協定為 Session Initiation Protocol (SIP) [16] 與 Internet Protocol version 6 (IPv6) [27]。為了進一步研究 IMS 系統，在開發與佈建系統時需要分析軟體的協助。然而目前並無針對 SIP 與 IPv6 分析的免費軟體，且開放原始碼 (Open Source) 的封包分析軟體 (例如 Ethereal [5]、WinDump [33] 及 TCPDump [32] 等) 或商用的封包分析軟體 (例如 Sniffer Pro [31] 等) 都較著重於協定標頭分析，較少著墨於信令協定流程與多媒體串流兩者的整合分析；針對 SIP-based VoIP 通訊的專業分析軟體價格卻又極為昂貴 (例如 Hammer Call Analyzer [10] 要價美金 49,900，約新台幣壹佰五十萬元)。因此本論文針對第三代行動通訊之核心通訊協定 SIP 與 IPv6 實作出專業的協定分析軟體系統 (稱為 SIPv6 即時協定分析測試系統)。本軟體系統亦支援分析行動通訊全 IP 核心網路上由 Signal Transport (SIGTRAN) [11] 標準所定義的信令封包。此軟體系統不但可以協助國內廠商進行第三代行動通訊的研發，並可做為 IPv6 與 SIP-based VoIP 通訊教育改進計畫的實驗教學教材。

現有通訊協定眾多，為了提供更完整的協定分析，SIPv6 即時協定分析測試系統採用開放原始碼的函式庫套件來加速系統開發，並設計統一的程式介面與架構 (Framework)，使未來新增協定分析時或軟體系統維護上都變得很容易。關於 SIPv6 即時協定分析測試系統的主要設計需求，請參考附錄 A。

本章說明本論文所設計實作之 SIPv6 即時協定分析測試系統的主要相關協定，包括 Internet Protocol version 4 (IPv4)、IPv6、SIP、Session Description Protocol (SDP) [20] 及 Real-time Transport Protocol (RTP) [9] 協定。SIPv6 即時協定分析測試系統支援解析 6to4 通道 (Tunnel) [2] 與 Teredo [3] 兩種以建立通道方法為主的協定技術，亦能圖形化輸出 SIGTRAN 標準所定義之信令封包 (目前包括 Stream Transmission Control Protocol (SCTP) [26]、ISDN Q.921-User Adaptation (IUA) Layer [19]、Message Transfer Part2 – User Adaptation (M2UA) Layer [18] 與 Message Transfer Part3 – User Adaptation (M3UA) Layer [7] 協定) 的交換流程。關於 6to4 通道、Teredo 與 SIGTRAN 標準的說明，請參考附錄 B 及 C。

1.1 IPv4/IPv6 協定與標頭格式

SIPv6 即時協定分析測試系統支援解析 IPv4 [12] 與 IPv6 封包。以下簡單介紹 IPv4 與 IPv6，並說明其標頭格式。

IPv4 為非可靠 (Unreliable) 及非連接導向 (Connectionless) 的協定，主要負責封包路由 (Routing) 的工作。IPv4 標頭格式如圖 1-1 所示。IPv4 標頭欄位說明如表 1-1 所列。

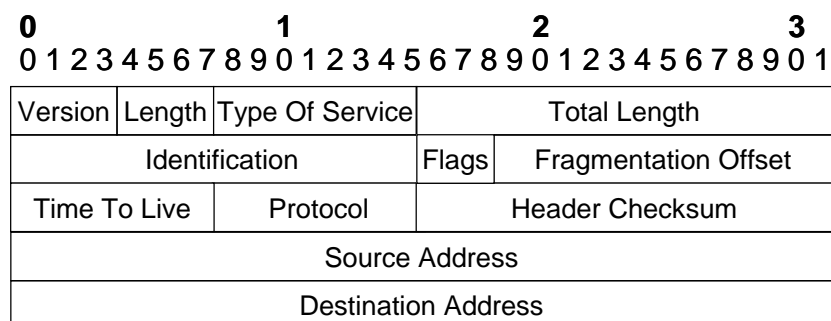


圖 1-1 IPv4 標頭格式圖

表 1-1 IPv4 標頭欄位說明表

IPv4 標頭欄位	說明
Version	IP 協定版本，此處必為 4
Length	IPv4 標頭長度(包括 Option 標頭長度)
Type of Service (TOS)	服務型態，詳細定義請參考 RFC760
Total Length	整個 IPv4 封包的長度
Identification	IPv4 封包的識別碼
Flags	協助 IPv4 封包進行切割重組的資訊旗標
Fragmentation offset	IPv4 封包的切割偏移值
Time to live (TTL)	IPv4 封包的路由限制節點數
Protocol	IPv4 封包上層的通訊協定
Header checksum	IPv4 封包標頭的檢查加總碼
Source address	IPv4 封包的來源 32 位元位址
Destination address	IPv4 封包的目的地 32 位元位址

IPv4 標準制定於 1980 年，而現今的網路環境與當時的網路環境相差甚遠，使得 IPv4 的功能設計不符合現今的網路環境需求，因此一些技術標準被提出來解決目前 IPv4 的不適問題，例如網路轉址 (Network Address Translation) [17] 解決 IPv4 位址不足問題及 Encapsulating Security Payload (ESP) 與 Authentication Header (AH)[28][29] 標準解決 IPv4 安全性問題等。Internet Engineering Task Force (IETF) 根據現今的網路架構，並考量解決如上述問題與未來網路發展需求而制定出 IPv6 標準。

IPv6 又稱 IPng (Next Generation Internet Protocol) 與 IPv4 的主要不同之處在於增

強的定址能力與自動設定 (Auto-configuration) 機制、簡化的標頭格式、新增的認證與隱私之延伸標頭以及資料流標籤 (Flow Label) 標頭欄位 [30]。IPv6 標頭格式如圖 1-2 所示。IPv6 標頭欄位說明如表 1-2 所列。

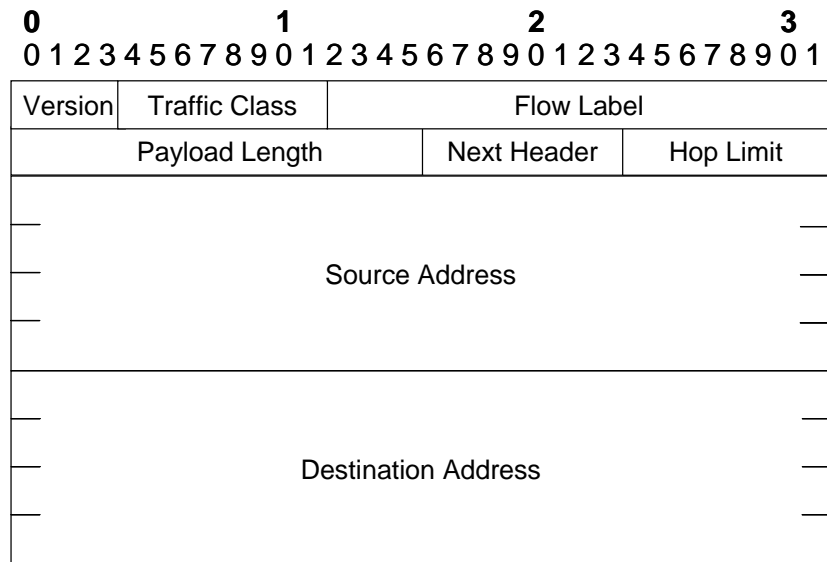


圖 1-2 IPv6 標頭格式圖

表 1-2 IPv6 標頭欄位說明表

IPv6 標頭欄位	說明
Version	IP 協定版本，此處必為 6
Traffic Class	服務等級，詳細定義請參考 RFC2474
Flow Label	訊流標示，路由器須以相同方式處理同訊流標示的封包
Payload Length	IPv6 標頭後的承載長度
Next Header	IPv6 封包上層的通訊協定
Hop Limit	同 IPv4 標頭中的 TTL 欄位
Source address	IPv6 封包的來源 128 位元位址
Destination address	IPv6 封包的目的地 128 位元位址

1.2 SIP/SDP 協定與標頭格式

本節簡單介紹 SIP 與 SDP 兩種協定，並說明其標頭格式。

SIP 為應用層協定，可架構於 Transmission Control Protocol (TCP) [14]或 User Datagram Protocol (UDP) [15]上層（預設埠號為 5060）。SIP 主要被用來控制多媒體會期（Session）的建立、修改或中斷。SIP 信令分為 SIP 要求（Request）與 SIP 回應（Response）。SIP 要求以 SIP 方法（Method）指明要求的目的。RFC 3261 定義的六種 SIP 方法分別為 INVITE、ACK、CANCEL、OPTIONS、REGISTER 與 BYE。INVITE 方法用來建立會期。ACK 方法主要用來作為 INVITE 方法的最後回應（Final Response；包括 2xx、3xx、4xx、5xx、6xx 的回應）。CANCEL 方法用來取消未建立完成的 SIP 要求。OPTIONS 方法用來查詢 SIP 伺服器的能力。REGISTER 方法用來向 SIP 伺服器註冊位置資訊。BYE 方法用來結束已建立完成的會期。SIP 回應以回應碼來表示 SIP 要求的結果。回應碼 1xx（100~199）表示暫時性回應。回應碼 2xx（200~299）表示要求方法成功。回應碼 3xx（300~399）表示重導（Redirect）要求方法。其餘的回應碼包括 4xx、5xx 與 6xx 表示要求方法失敗。

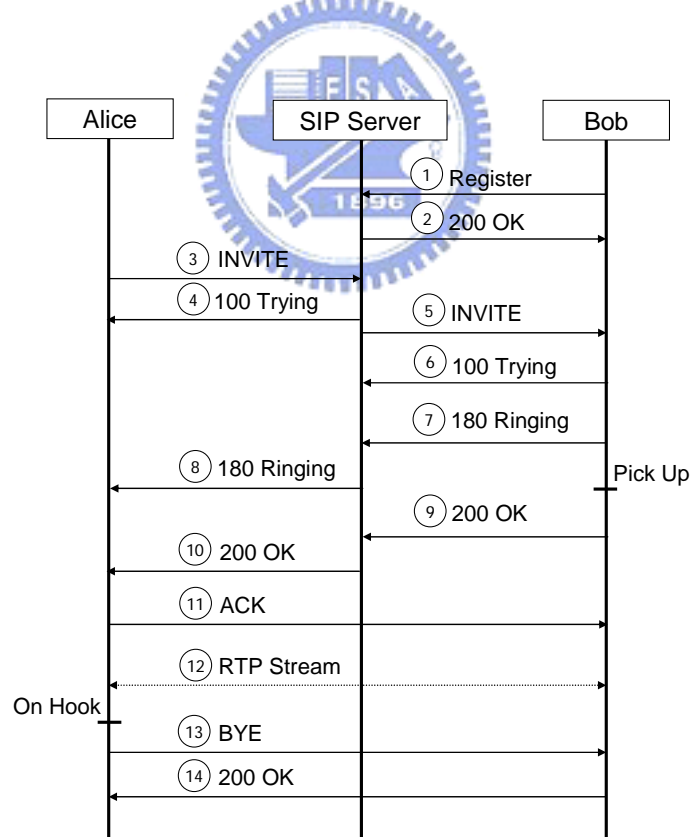


圖 1-3 SIP 協定運作範例圖

SIP 的網路原件包括 User Agent (UA)、Proxy Server、Redirect Server 與 Registrar。UA 利用 SIP 要求與 SIP 回應來進行多媒體會期的建立。Proxy Server 負責路由並

控管 SIP 要求與 SIP 回應。Redirect Server 會產生 3xx 回應使 UA 重導 SIP 要求到其他 SIP 伺服器。Registrar 讓 UA 以 Register 方法註冊位置資訊。

圖 1-3 的 SIP 信令流程範例展示簡單的 SIP 運作。圖中的 Alice 與 Bob 為 SIP UA，而 SIP Server 提供 Proxy Server 與 Registrar 功能。當 Bob 的 IP 位址資訊改變時會發出 REGISTER 要求向 SIP Server 註冊目前的 IP 位址（如圖 1-3①）。SIP Server 回應 200 OK 表示 Bob 成功的完成註冊（如圖 1-3②）。Alice 欲與 Bob 進行 SIP VoIP 通訊。首先 Alice 向 Bob 所在的 SIP Server 發出 INVITE 要求（如圖 1-3③）。SIP Server 收到 INVITE 要求後，首先回應 100 Trying 表示嘗試處理 INVITE 要求（如圖 1-3④）。SIP Server 將 INVITE 要求傳送到 Bob 之前所註冊的位置（如圖 1-3⑤）。Bob 收到 INVITE 要求後，回應 100 Trying 表示嘗試處理 INVITE 要求（如圖 1-3⑥）。Bob 回應 180 Ringing 表示目前正處於響鈴狀態（如圖 1-3⑦）。SIP Server 將 180 Ringing 傳回 Alice（如圖 1-3⑧）。當 Bob 接受語音通訊時，Bob 回應 200 OK 表示接受 INVITE 要求（如圖 1-3⑨）。SIP Server 將 200 OK 傳回 Alice（如圖 1-3⑩）。Alice 由 Bob 回應的 200 OK 中得知目前 Bob 的位置，並發出 ACK 要求回應 Bob 的 200 OK（如圖 1-3⑪）。Alice 與 Bob 以 RTP 進行雙向的語音通訊（如圖 1-3⑫）。當 Alice 結束語音通訊時，Alice 發出 BYE 要求給 Bob（如圖 1-3⑬）。最後 Bob 回應 200 OK 表示結束語音通訊（如圖 1-3⑭）。

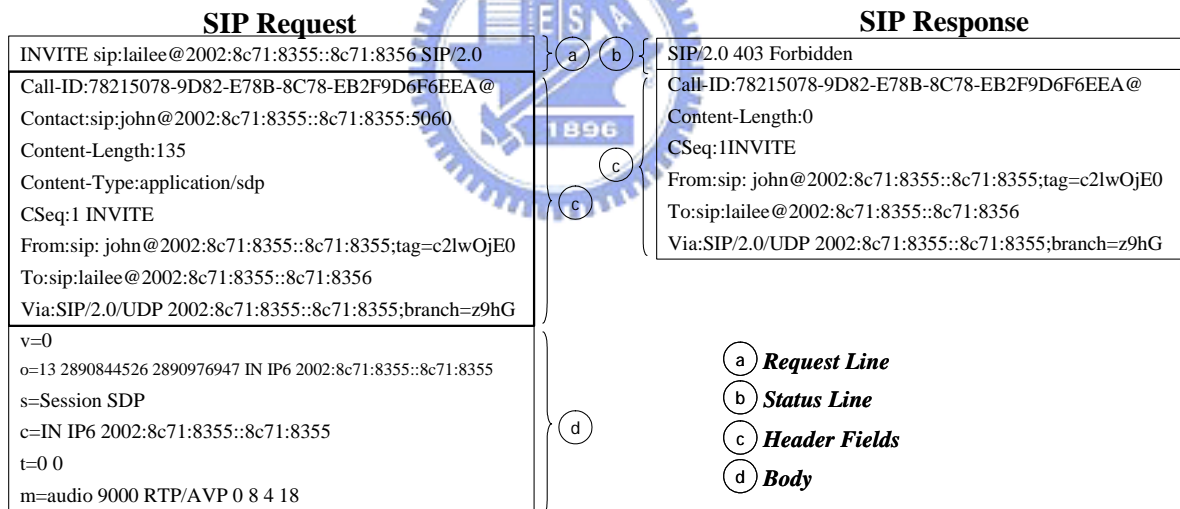


圖 1-4 SIP 信令範例圖

SIP 為採用 Universal Character Set Transformation Format (UTF) -8 [6]字集的信令協定。圖 1-4 中的左部分為 SIP 要求，右部分為 SIP 回應。SIP 信令的結構分為四部份。這四部分為要求行 (Request Line; 圖 1-4(a))、狀態行 (Status Line; 圖 1-4(b))、標頭欄位 (Header Fields; 圖 1-4(c)) 與主體 (Body; 圖 1-4(d))。要求行指明要求方法、目的 SIP Uniform Resource Identifier (URI) 與版本。狀態行指明版本、回應代碼與回應代碼的文字描述。標頭欄位擺放標頭資訊，必要的標頭欄位與欄位說明如表 1-3 所列。主體擺放的資訊是根據 Content-Type 標頭欄位的定義。

表 1-3 SIP 標頭欄位說明表

SIP 標頭欄位	說明
Call-ID	Call-ID 值用來辨識 SIP 對話 (Dialog)，同 SIP 對話會有相同 Call-ID 值
From	進行初始 SIP 會期的來源端 SIP URI
To	SIP 會期目的端的 SIP URI
Via	指明 SIP 回應的回程路由路徑
Cseq	用來匹配 SIP 要求與 SIP 回應
Max-Forwards	SIP 要求信令經過 SIP 伺服器，伺服器會將此欄位值減一，當此欄位值為零，SIP 要求將被伺服器丟棄

SIPv6 即時協定分析測試系統利用 Call-ID 標頭欄位資訊來做 SIP 對話的分類，並根據 Via 與 Route 標頭欄位資訊來產生 SIP 信令流程圖。故在此說明 SIP 中用來辨識 SIP 對話的 Call-ID，以及用來決定路由的 Via、Route 與 Record-Route 標頭欄位，其餘的標頭欄位請參考 RFC 3261。Call-ID 標頭欄位用來辨識 SIP 對話，同 SIP 對話會有相同 Call-ID 值。比如兩個 UA 進行 SIP VoIP 通訊中，從開始用來建立會期的 INVITE 要求到最後用來結束會期的 BYE 要求中的所有 SIP 信令，都有相同的 Call-ID 值，並都在同一個 SIP 對話中。Via 頭欄位指明 SIP 回應的回程路由路徑。Route 標頭欄位指明 SIP 要求的傳送路由路徑。Record-Route 標頭欄位則是用來做為 Route 標頭欄位的學習。透過 Via 與 Route 標頭欄位可得知 SIP 信令的行進路徑。

表 1-4 SDP 型態說明表

型態	說明
v	SDP 版本
o	會期 ID 與產生會期的主機資訊
s	會期名稱
c	連接資訊
b	頻寬資訊
z	時區調整資訊
k	加密鑰匙 (Key)
a	會期屬性
t	會期的起始時間
r	會期的重複次數
m	多媒體名稱與傳輸位址
i	多媒體標題

u	URI 資訊
e	E-Mail 位址
p	電話號碼

SDP 為採用 UTF-8 字集的描述協定。SDP 提供多媒體串流傳送或接收的相關資訊。SDP 包含數行<Type>=<Value>格式的文字，如圖 1-4 中 SIP 要求的主體部分所示。表 1-4 列出所有 SDP 協定中的型態 (Type) 說明。

以圖 1-4④的 SIP 主體為例，v=0 表示 SDP 版本為 0。o=13 2890844526 2890976947 IN IP6 2002:8c71:8355::8c71:8355 表示產生會期的使用者名稱為 13、會期識別碼為 2890844526、會期版本為 2890976947、網路型態為 IN 即 Internet、主機位址型態為 IP6 即主機位址型態為 IPv6 位址與主機位址為 2002:8c71:8355::8c71:8355。s=session SDP 表示會期名稱為 session SDP。c=IN IP6 2002:8c71:8355::8c71:8355 表示會期連結的網路型態為 IN 即 Internet、主機位址型態為 IPv6 與主機位址為 2002:8c71:8355::8c71:8355。t=0 0 表示會期的起始時間為與結束時間為 0 (起始於結束時間都為 0，表示會期會一直持續)。m=audio 9000 RTP/AVP 0 8 4 18 表示會期的多媒體型態為語音、接收多媒體資訊的埠號為 9000、傳輸層協定採 RTP/AVP 即 RTP over UDP 與語音編碼格式依序為 0 (表示 PCMU)、8 (表示 PCMA)、4 (表示 G.723) 及 18 (表示 G.729)，SDP 詳細對照請參考 RFC3550。

1.3 RTP 協定與標頭格式

本節簡單介紹 RTP 協定並說明其標頭格式。

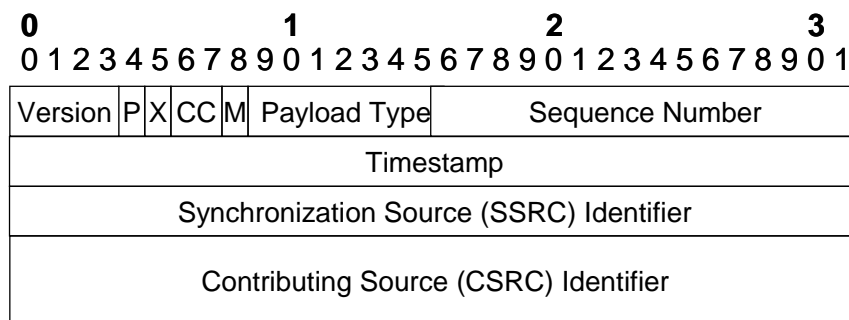


圖 1-5 RTP 標頭格式圖

RTP 針對需要傳送即時資料的應用系統提供點對點的即時傳輸功能。RTP 沒有保留網路資源的能力，亦不保證服務品質 (Quality of Service)。然而 RTP 所提供的即時傳輸功能是指 RTP 標頭能提供應用系統進行即時傳輸所需的資訊，包括封包序號 (由 Sequence Number 標頭欄位提供)、編碼 (由 Payload Type 標頭欄位提

供)、取樣時戳(由 TimeStamp 標頭欄位提供)等資訊。當應用系統採 RTP 進行多媒體串流傳輸時,傳送端會將多媒體串流進行切割,並將切割的多媒體片段加上 RTP 標頭傳送出去,同時接收端會將收到的 RTP 封包根據 RTP 標頭提供的資訊將多媒體串流片段進行即時撥放處理。RTP 標頭格式如圖 1-5 所示。RTP 標頭欄位說明如表 1-5 所列。

表 1-5 RTP 標頭欄位說明表

RTP 標頭欄位	說明
Version	RTP 協定版本,目前值為 1
P (Padding bit)	如果這個欄位被設定,RTP 封包尾端會包含一個以上的 Padding 位元,但它們並不是 RTP 承載的一部份
X(Extension bit)	如果這個欄位被設定,在固定的 RTP 標頭後還要加上延伸標頭
CC(CSRC count)	紀錄 CSRC 的個數
M(Mark bit)	用做重要事件,像是影像中影格(Frame)的界限識別。應用系統決定此位元值的意義
Payload type	RTP 承載的格式
Sequence number	每送出一個 RTP 封包,序號就會加一。序號可以被接收端用來偵測 RTP 封包遺失,以及 RTP 封包次序錯誤
Timestamp	RTP 承載取樣時間
SSRC	用來識別串流,同一串流會著同相同的 SSRC 值
CSRC	被 RTP Mixer 混入的串流 SSRC 列表,詳細定義請參考 RFC 3550

本論文後續的章節組織描述如下。第二章說明 SIPv6 即時協定分析測試系統的系統架構與運作範例。第三章說明 SIPv6 即時協定分析測試系統的實作原理。第四章說明 SIPv6 即時協定分析測試系統的的安裝步驟與操作方法。第五章以展示案例來說明 SIPv6 即時協定分析測試系統的功能特點。第六章為結論。

二、SIPv6 即時協定分析系統架構

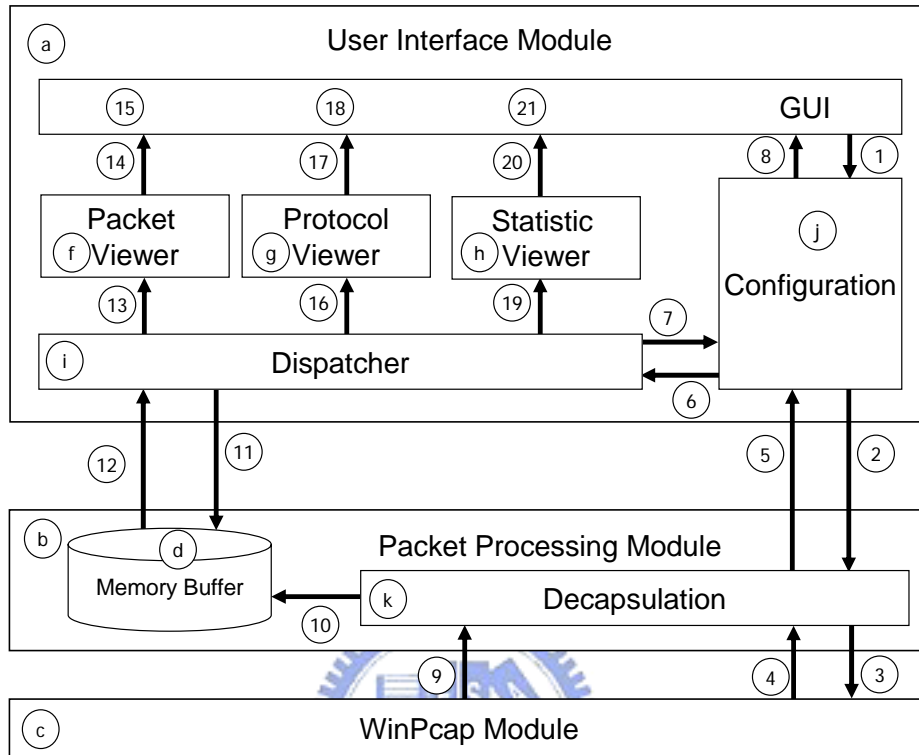


圖 2-1 SIPv6 即時協定分析系統架構圖與運作範例圖

SIPv6 即時協定分析系統架構圖如圖 2-1 所示。本系統包括三大模組，分別為使用者介面模組（User Interface Module；圖 2-1(a)）、封包處理模組（Packet Processing Module；圖 2-1(b)）以及 WinPcap 模組（WinPcap Module；圖 2-1(c)）。使用者介面模組負責進行系統初始化工作，並根據通訊協定將擷取的封包加以分類，最後以圖形化介面與音訊多媒體的方式呈現封包分析與統計的結果。封包處理模組透過 WinPcap 模組取得封包，並將封包解封裝（Decapsulation）為使用者介面模組可以分析的格式，最後將已解封裝的封包儲存於記憶體緩衝區（Memory Buffer；圖 2-1(d)）。WinPcap 模組提供三種方式來擷取封包，可以由本機的網路卡驅動程式取得封包、讀取本機檔案系統中已存檔的封包，或是藉由 TCP/IP 介面到遠端主機擷取封包。為加速系統開發，SIPv6 即時協定分析系統採用開放原始碼的 WinPcap 函式庫套件 [34] 為底層進行封包擷取的模組；而使用者介面模組與封包處理模組則為本論文所研發，以下分別介紹本論文所研發的模組。

使用者介面模組包括圖形使用者介面（Graphic User Interface，簡稱 GUI）模組（圖 2-1(e)）、Packet Viewer 模組（圖 2-1(f)）、Protocol Viewer 模組（圖 2-1(g)）、Statistic Viewer 模組（圖 2-1(h)）、Dispatcher 模組（圖 2-1(i)）以及組態（Configuration）

模組 (圖 2-1j) 。GUI 模組以圖形化介面與音訊多媒體的方式呈現封包的統計結果。Packet Viewer 模組主要採用開放原始碼的 Ethereal 封包分析器 [5] 對 Hypertext Transfer Protocol (HTTP) [24]、File Transfer Protocol (FTP) [13] 與 Domain Name System (DNS) [22] 等六百多種通訊協定封包進行通訊協定解析，並將解析結果輸出至 GUI 模組。Protocol Viewer 模組針對 VoIP 以及電信相關的 SIP、RTP 與 Stream Control Transmission Protocol (SCTP) 三種通訊協定的封包進行分析。對於 SIP 通訊協定，Protocol Viewer 模組將 SIP 封包以 SIP 對話分類，並繪出 SIP 對話的 SIP 信令流程圖，最後將 SIP 對話資訊與流程圖輸出至 GUI 模組。對於 RTP 通訊協定，Protocol Viewer 模組將 RTP 封包以 RTP 會期分類，並輸出 RTP 會期 (包括 SSRC 標頭欄位、編碼格式與語音長度) 資訊至 GUI 模組；使用者也可以經由使用者介面模組控制播放 RTP 會期所帶的語音資訊 (目前可支援 PCMU/PCMA 編碼)。對於 SCTP 通訊協定，Protocol Viewer 模組將位於 SCTP 封包中的 IUA/M2UA/M3UA 信令流程圖輸出至 GUI 模組。Statistic Viewer 模組負責進行主機流量排行統計、封包分佈統計與網路流量變化統計。Dispatcher 模組以詢問 (Pooling) 方式檢查封包處理模組的記憶體緩衝區是否有新的封包，如有則依序通知 Packet Viewer 模組、Protocol Viewer 模組與 Statistic Viewer 模組進行封包的統計工作。組態模組會根據使用者在 GUI 模組的組態設定進行系統初始化以及參數設定的工作。

封包處理模組中的解封裝模組 (Decapsulation Module) (圖 2-1k) 負責建立儲存封包的記憶體緩衝區，並透過 WinPcap 模組以中斷 (Interrupt) 方式取得封包。為減少封包漏失 (Lost) 的可能性，解封裝模組取得封包後只對標頭長度較為固定與標頭格式簡單的第二層 (Layer 2，例如：Ethernet)、第三層 (Layer 3，例如：IPv4 與 IPv6)、第四層 (Layer 4，例如：TCP、UDP 與 SCTP) 通訊協定進行解封裝，並將解封裝後的封包存放至記憶體緩衝區。使用者介面模組將再由記憶體緩衝區取得解封裝封包，並進行應用層 (Application Layer，如：SIP) 的通訊協定解析工作。

2.1 系統運作範例

本節以圖 2-1 說明 SIPv6 即時協定分析系統的運作範例。

步驟①:

使用者首先經由 GUI 模組選取欲擷取封包的網路介面卡、設定網路介面卡是否以全收 (Promiscuous) 模式進行封包擷取、設定網路介面卡的封包過濾規則 (Packet Filter Rule)，也可以設定 Dispatcher 模組進行詢問的間隔時間等四個參數。當網路卡處於全收模式時，網路卡可以收集封包目的 Media Access Control (MAC) 位

元址不是該網路卡 MAC 位址的封包。封包過濾規則可以用來指定系統進行特定封包的收集，例如封包過濾規則 udp port 5060 是用來指定系統收集 SIP 協定的封包。詢問間隔時間的設定將影響到使用者介面模組進行封包分析的即時性與封包處理模組進行封包收集的效率。詢問的間隔時間越短，使用者介面模組進行封包分析的即時性越高，但封包處理模組進行封包收集的封包漏失可能性會越高。當使用者完成設定後，由於底層的 WinPcap 模組是以設備名稱 (Device Name，例如 \Device\NPF_{8A4895E6-4131-414D-9E1B-3358CDBDA221}) 來取得網路卡驅動程式的參考 (Reference)，而使用者看到的是可讀的網路卡名稱 (例如 SiS 900-Based PCI Fast Ethernet Adapter)，因此 GUI 模組需要作一個轉換。GUI 模組向作業系統取得網路介面卡於作業系統內部對應的設備名稱，並將設備名稱與組態設定傳給組態模組以進行系統初始化工作。

步驟②:

組態模組將設備名稱、封包收集模式與封包過濾規則等三項資訊傳給解封裝模組以進行封包收集與解封裝的工作。

步驟③、④及⑤:

解封裝模組首先建立鏈結資料結構的記憶體緩衝區，然後經由 WinPcap Application Programming Interface (API) 要求 WinPcap 模組透過設備名稱對應的網路卡驅動程式進行封包擷取，並設定封包收集模式與封包過濾規則。WinPcap 模組完成初始化工作並回傳成功。封包處理模組完成初始化工作並回傳成功。

步驟⑥及⑦:

組態模組將詢問的間隔時間資訊傳給 Dispatcher 模組，讓 Dispatcher 模組進行初始化工作。Dispatcher 模組建立進行詢問工作的執行緒 (Thread)，並根據組態模組傳遞的資訊 (步驟⑥) 設定該執行緒進行詢問的間隔時間。執行緒開始檢查封包處理模組的記憶體緩衝區是否存放新收集的封包。Dispatcher 模組完成初始化工作並回傳成功。

步驟⑧:

組態模組完成系統初始化工作並回傳成功。

步驟⑨:

WinPcap 模組透過網路介面卡驅動程式擷取封包，並將封包以中斷方式傳遞給解封裝模組進行封包解封裝的工作。

步驟⑩:

封包擷取模組根據第二層、第三層、第四層通訊協定標頭，將封包解封裝，並將解封裝的封包存放至記憶體緩衝區。

步驟⑪與⑫:

Dispatcher 模組以詢問方式檢查到記憶體緩衝區內有新收集的封包後，從記憶體緩衝區取出新收集的封包。

步驟⑬:

Dispatcher 模組將封包傳給 Packet Viewer 模組進行封包分析工作。

步驟⑭:

Packet Viewer 模組首先將封包儲存成暫存檔，然後以 Ethereal 封包分析器對暫存檔解析。最後將 Ethereal 封包分析器的協定解析輸出與封包的第二層、第三層、第四層通訊協定標頭傳到 GUI 模組。

步驟⑮:

GUI 模組根據通訊協定標頭欄位輸出封包資訊至封包列表，並將轉換 Ethereal 封包分析器的協定解析輸出為圖形化的樹狀通訊協定解析輸出。

步驟⑯:

Dispatcher 模組將封包傳給 Protocol Viewer 模組進行封包分析工作。

步驟⑰:

Protocol Viewer 模組首先解析應用層的通訊協定來判斷封包是否為 SIP 或 RTP 協定。若封包為 SIP 協定，Protocol Viewer 模組根據封包中的 Call-ID 標頭欄位將封包以 SIP 對話分類，並更新 SIP 對話的 SIP 封包計數器。最後將 SIP 對話資訊與 SIP 封包計數器傳到 GUI 模組。若封包為 RTP 協定，Protocol Viewer 模組將根據封包的第三層、第四層通訊協定標頭欄位與 RTP SSRC 標頭欄位將 RTP 封包以 RTP 會期分類，並更新 RTP 會期的 RTP 封包計數器，最後將封包的 RTP 會期資訊與 RTP 封包計數器傳到 GUI 模組。若封包不為 SIP 或 RTP 協定，Protocol Viewer 模組根據第三層通訊協定標頭欄位判斷封包是否為 SCTP 協定。若封包為 SCTP 協定，Protocol Viewer 模組再根據 SCTP Common 標頭判斷 SCTP 封包上層是否為 IUA/M2UA/M3UA 信令。若封包為 IUA/M2UA/M3UA 信令，Protocol Viewer

模組將 SCTP Common 標頭與第三層通訊協定標頭傳到 GUI 模組。其餘的協定封包不在 Protocol Viewer 中處理。

步驟⑱:

若封包為 SIP 協定，GUI 模組輸出 SIP 對話資訊至 SIP 對話列表，並根據 SIP 封包計數器更新 SIP 封包個數統計資訊。使用者可以從 SIP 對話列表選取 SIP 對話做圖形化 SIP 信令流程輸出。若封包為 RTP 協定，GUI 模組輸出 RTP 會期資訊至 RTP 會期列表，並根據 RTP 封包計數器更新 RTP 封包個數統計資訊。當使用者點選 GUI 模組上的播放鈕時，GUI 模組將以 Windows Multimedia API [21] 進行 RTP 會期的解碼與播放工作。若封包為 IUA/M2UA/M3UA 信令，GUI 模組根據 SCTP Common 標頭更新 IUA/M2UA/M3UA 信令流程圖。

步驟⑲:

Dispatcher 模組將封包傳給 Statistic Viewer 模組進行封包統計工作。

步驟⑳:

Statistic Viewer 模組首先根據第二層、第三層、第四層通訊協定標頭欄位更新紀錄主機流量統計資訊中的總封包個數與總封包大小計數器，然後更新封包分佈統計資訊中的 IPv4/IPv6 總封包個數計數器，並計算流量變化計數器，最後將計數器傳給 GUI 模組。

步驟㉑:

GUI 模組根據計數器更新主機流量統計表、封包分佈圓餅圖以及流量變化圖。

三、實作原理

本系統的實作重點在封包的解封裝處理與提供 IMS 相關協定的通訊協定解析及分析，因此本章節說明解封裝模組（圖 2-1(k)）與 Protocol Viewer 模組（圖 2-1(g)）的實作原理。

3.1 解封裝模組

解封裝模組包含九種解封裝處理（Handler）單元（圖 3-1(a)-(i)）。解封裝處理單元負責解析封包中的通訊協定標頭，並由通訊協定標頭欄位判斷封包上層的封裝型態，最後將解封裝的封包存放至記憶體緩衝區中。

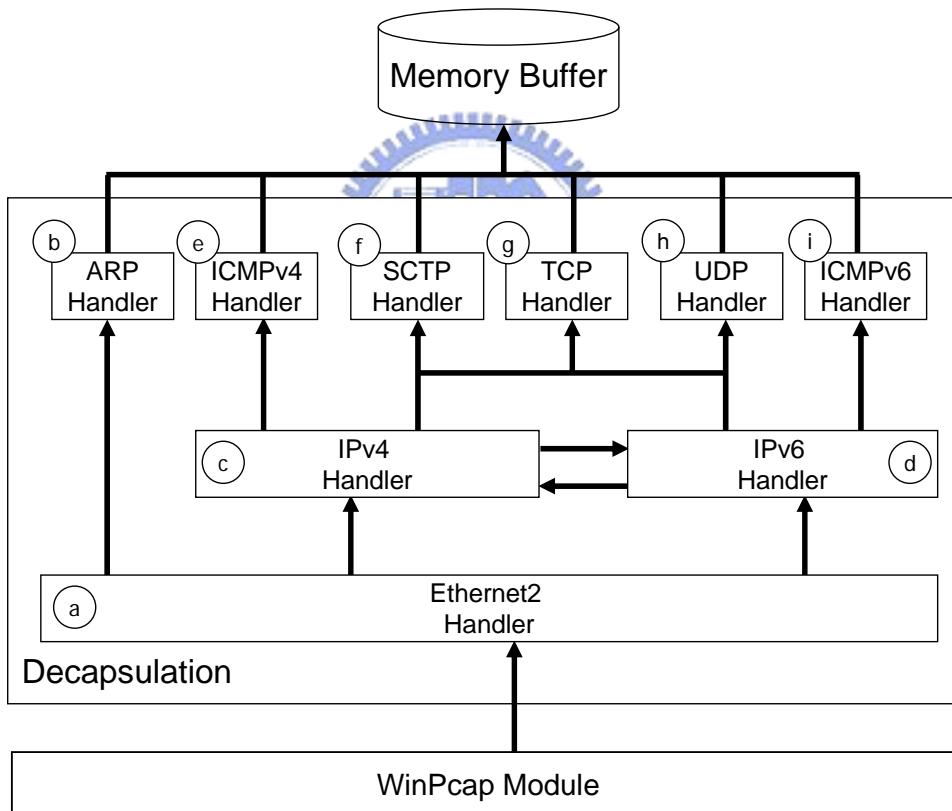


圖 3-1 解封裝模組架構圖

解封裝模組所支援解析的第二層協定包括 Ethernet2（圖 3-1(a)）與 Address Resoulution Protocol (ARP)（圖 3-1(b)）協定的解封裝處理單元；第三層協定包括 IPv4（圖 3-1(c)）與 IPv6（圖 3-1(d)）協定的解封裝處理單元；第四層協定包括 ICMPv4（圖 3-1(e)）、Sctp（圖 3-1(f)）、TCP（圖 3-1(g)）、UDP（圖 3-1(h)）與 ICMPv6（圖 3-1(i)）協定的解封裝處理單元。

```

/* Definition of Protocol Descriptor */
typedef struct L234DMProtocolDescriptor
{
    unsigned int type; /* Header type, i.e. ETH2, ARP, IPV6 */
    unsigned int len; /* Header length */
    unsigned int bytesoffset; /* Start offset */
    L234DMProtocolDescriptor *next; /* Next header */
} *PL234DMProtocolDescriptor, **PPL234DMProtocolDescriptor;

```

圖 3-2 通訊協定描述子的型態定義圖

本系統內部以通訊協定描述子（Protocol Descriptor）資料結構（圖 3-2）表示封包的封裝資訊，而封裝資訊包括通訊協定類別、起始位址與長度。解封裝模組以解封裝處理單元建置表示封包封裝結構的通訊協定描述子串列資料結構。

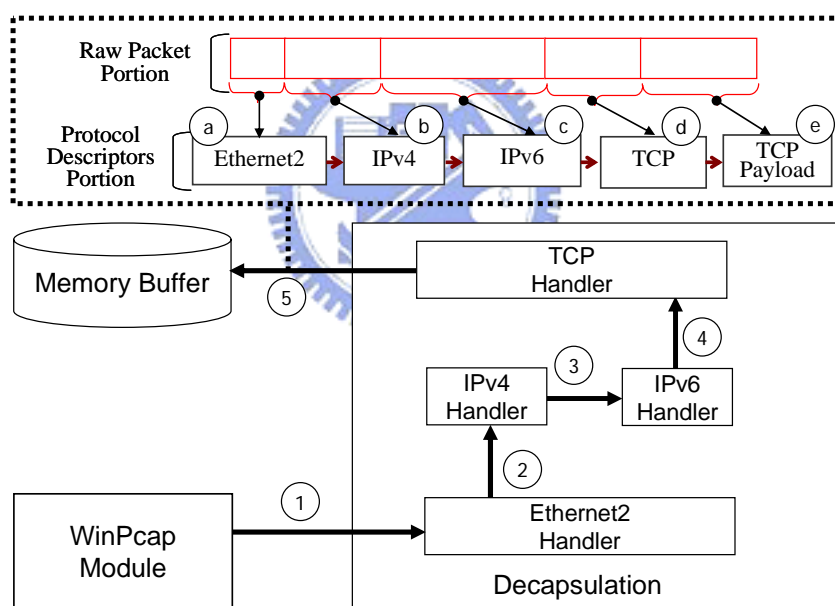


圖 3-3 IPv6 通道封包的解封裝運作流程範例圖

圖 3-3 說明解封裝模組對 IPv6 通道封包的解封裝運作流程。首先解封裝模組由 WinPcap 模組取得 IPv6 通道封包，並由 Ethernet2 處理單元開始進行解封裝處理（圖 3-3①）。Ethernet2 處理單元建構 Ethernet2 通訊描述子結點（圖 3-3(a)），並經由 Ethernet2 Type 標頭欄位得知其上層為 IPv4 格式，然後將封包交由 IPv4 處理單元（圖 3-3②）。IPv4 處理單元建構 IPv4 通訊描述子結點（圖 3-3(b)），並經由 IPv4 Protocol 標頭欄位得知其上層為 IPv6 格式，然後將封包交由 IPv6 處理單元（圖 3-3③）。IPv6 處理單元建構 IPv6 通訊描述子結點（圖 3-3(c)），並經由 IPv6 Next Header 標頭欄位得知其上層為 TCP 格式，然後將封包交由 TCP 處理單元（圖

3-3④)。TCP Handler 處理單元建構 TCP 通訊描述子結點 (圖 3-3④)，然後將封包扣掉前端所有通訊協定標頭取得 TCP 承載，並加上 TCP 承載通訊描述子結點 (圖 3-3③)，如此完成通訊描述子串列資料結構的建置。最後通訊描述子串列資料結構將存放於記憶體緩衝區中 (圖 3-3⑤) 提供使用者介面模組 (圖 2-1①) 進行後續的分析處理。

3.2 Protocol Viewer 模組

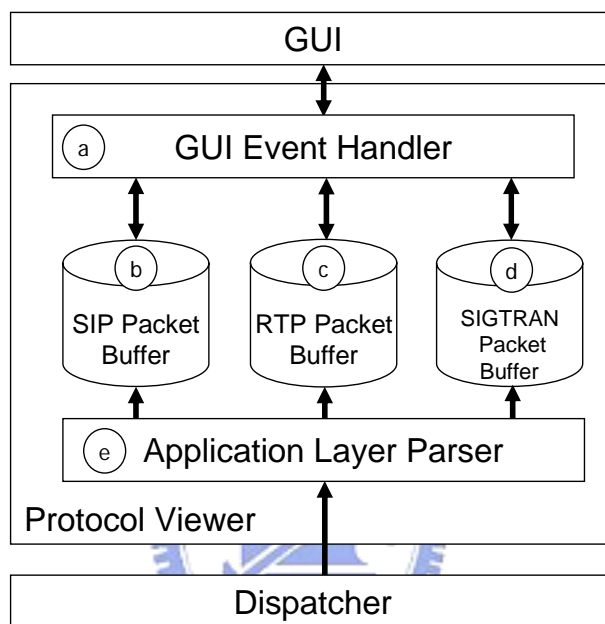


圖 3-4 Protocol Viewer 模組架構圖

Protocol Viewer 模組針對 SIP、RTP 與 SCTP 三種通訊協定的封包進行分析。Protocol Viewer 模組內部包括 GUI 事件處理 (Event Handler) 單元 (圖 3-4(a))、SIP 封包指標緩衝區 (圖 3-4(b))、RTP 封包指標緩衝區 (圖 3-4(c))、SCTP 封包指標緩衝區 (圖 3-4(d)) 與應用層解析 (Application Layer Parser) 單元 (圖 3-4(e))。

GUI 事件處理單元負責處理使用者操作 GUI 所產生的視窗事件 (Windows Event [21]；例如點選 SIP 流程圖選單、點選 RTP 串流播放按鈕與點選 RTP 播放停止按鈕) 以協助使用者產生圖形化 SIP 信令流程圖、產生圖形化 IUA/M2UA/M3UA 信令流程圖與進行 RTP 串流的播放混音功能。

SIP 封包指標緩衝區為以 SIP Call-ID 標頭欄位值為鍵值的 Map 資料結構 (C++ 標準樣版函式庫所提供的資料結構) [4]，主要用來存放 SIP 封包的記憶體指標 (Pointer)。當使用者要求產生圖形化 SIP 信令流程圖時，GUI 事件處理單元以 Call-ID 標頭欄位值為鍵值向 SIP 封包指標緩衝區取得 SIP 對話中的所有 SIP 信令

封包，並根據 SIP 封包第三層協定標頭欄位元（包括目的 IP 位元址標頭欄位與來源 IP 位元址標頭欄位）、Via 標頭欄位與 Route 標頭欄位值產生 SIP 信令流程圖。RTP 封包指標緩衝區為以 SSRC 標頭欄位與第三層、第四層協定標頭欄位元值（包括目的 IP 位元址標頭欄位元與目的埠號標頭欄位）為鍵值的 Map 資料結構，主要用來存放 RTP 封包的記憶體指標。當使用者進行 RTP 播放混音功能時，GUI 事件處理單元以 SSRC 標頭欄位與第三層、第四層協定標頭欄位值為鍵值向 RTP 封包指標緩衝區取得 RTP 會期中的所有 RTP 封包，並產生執行緒以 Window Multimedia API（表 3-1 所示）進行 RTP 封包的解碼與播放混音功能。

表 3-1 Windows Multimedia API 列表

API 名稱	呼叫用途說明
waveOutOpen	開啟語音播放設備並設定解碼格式
waveOutPrepareHeader	準備語音資料區塊(詳細定義請參考 MSDN)
waveOutWrite	寫入資料區塊，此時 windows 將對語音資料區塊指定的緩衝區進行解碼播放動作
waveOutUnprepareHeader	釋放語音資料區塊
waveOutReset	重置語音播放設備
waveOutClose	釋放語音播放設備的開啟
waveOutPause	暫停解碼播放動作

SIGTRAN 封包指標緩衝區為鏈結資料結構，主要用來存放 IUA/M2UA/M3UA 封包的記憶體指標。當使用者要求產生圖形化 IUA/M2UA/M3UA 信令流程圖時，GUI 事件處理單元由 SIGTRAN 封包指標緩衝區取出所有 IUA/M2UA/M3UA 信令封包，並根據封包的第三層協定標頭欄位元（包括目的 IP 位元址標頭欄位與來源 IP 位元址標頭欄位）與第四層協定標頭（SCTP 標頭欄位）產生 IUA/M2UA/M3UA 信令流程圖。

應用層解析單元負責解析由 Dispatcher 模組（圖 2-1①）傳遞上來的封包。首先以開放原始碼的 GNU oSIP2 [8] 套件解析封包是否為 SIP 封包。若封包為 SIP 封包，應用層解析單元將封包的記憶體指標存放於 SIP 封包指標緩衝區以供後續分析處理。若封包不為 SIP 封包，應用層解析單元將檢查封包的應用層協定標頭以判斷封包是否為 RTP 封包。若封包為 RTP 封包，應用層解析單元將封包的記憶體指標存放於 RTP 封包指標緩衝區以供後續分析處理。若封包不是 RTP 封包，應用層解析單元根據第三層協定標頭判斷封包是否為 SCTP 協定。若封包為 SCTP 封包。應用層解析單元根據 SCTP Common 標頭判斷 SCTP 封包上層是否為 IUA/M2UA/M3UA 信令。若封包為 IUA/M2UA/M3UA 信令，應用層解析單元將封包的記憶體指標存放於 SIGTRAN 封包指標緩衝區以供後續分析處理。其他協定封包，應用層解析模組不做任何處理。

四、SIPv6 即時協定分析測試系統安裝操作說明

本章節展示本系統的成果內容，主要包括系統安裝移除與啟動、系統操作說明。

4.1 系統安裝移除與啟動

本節首先說明本系統的軟硬體建議需求，接著說明本系統的安裝程式，再說明如何啟動本系統。最後說明如何在安裝本系統的電腦中移除本系統。

4.1.1 軟硬體建議需求

本系統的軟硬體建議需求如下表 4-1 所列。

表 4-1 軟硬體建議需求表

軟硬體建議需求	
作業系統	Microsoft Windows 2000/XP/2003
CPU	Intel Celeron 1GHz 或以上
記憶體	128MB
磁碟空間	18.6MB
函式庫套件	WinPcap 3.0 (已包含於安裝檔中)
開放原始碼封包分析器	Ethereal 0.10.0 (已包含於安裝檔中)
網路卡	乙太網路卡或 802.11a/b/g 無線網路卡

4.1.2 系統安裝

本系統的安裝步驟流程，如圖 4-1 所示。使用者點選本系統的安裝檔後會出現安裝步驟①的畫面，此時點選下一步按鈕 (Next) 進入安裝步驟②。安裝步驟②進行本系統採取的 General Public License (GPL) 協定宣告，選擇接受授權。完成此動作後點選下一步按鈕進入安裝步驟③。

安裝步驟③中，使用者可以更改預設安裝目錄，如不更改可以點選下一步按鈕進入安裝步驟④。安裝步驟④中，程式提供三種安裝模式。完全安裝 (Full Installation) 模式將安裝複製所有軟體套件。精簡安裝 (Compact Installation) 模式將安裝複製必要的軟體套件。客製化安裝 (Customized Installation) 模式將安裝複製使用者所選擇的軟體套件。選擇安裝模式後點選下一步按鈕進入安裝步驟⑤。安裝步驟⑤中，使用者可以修改本系統在程式集選單中的名稱，修改後點選下一步按鈕進入

安裝步驟⑥。安裝步驟⑥中會列出完整的系統安裝資訊，再點選安裝按鈕，此時就會進行檔案複製動作，完成本系統的安裝。

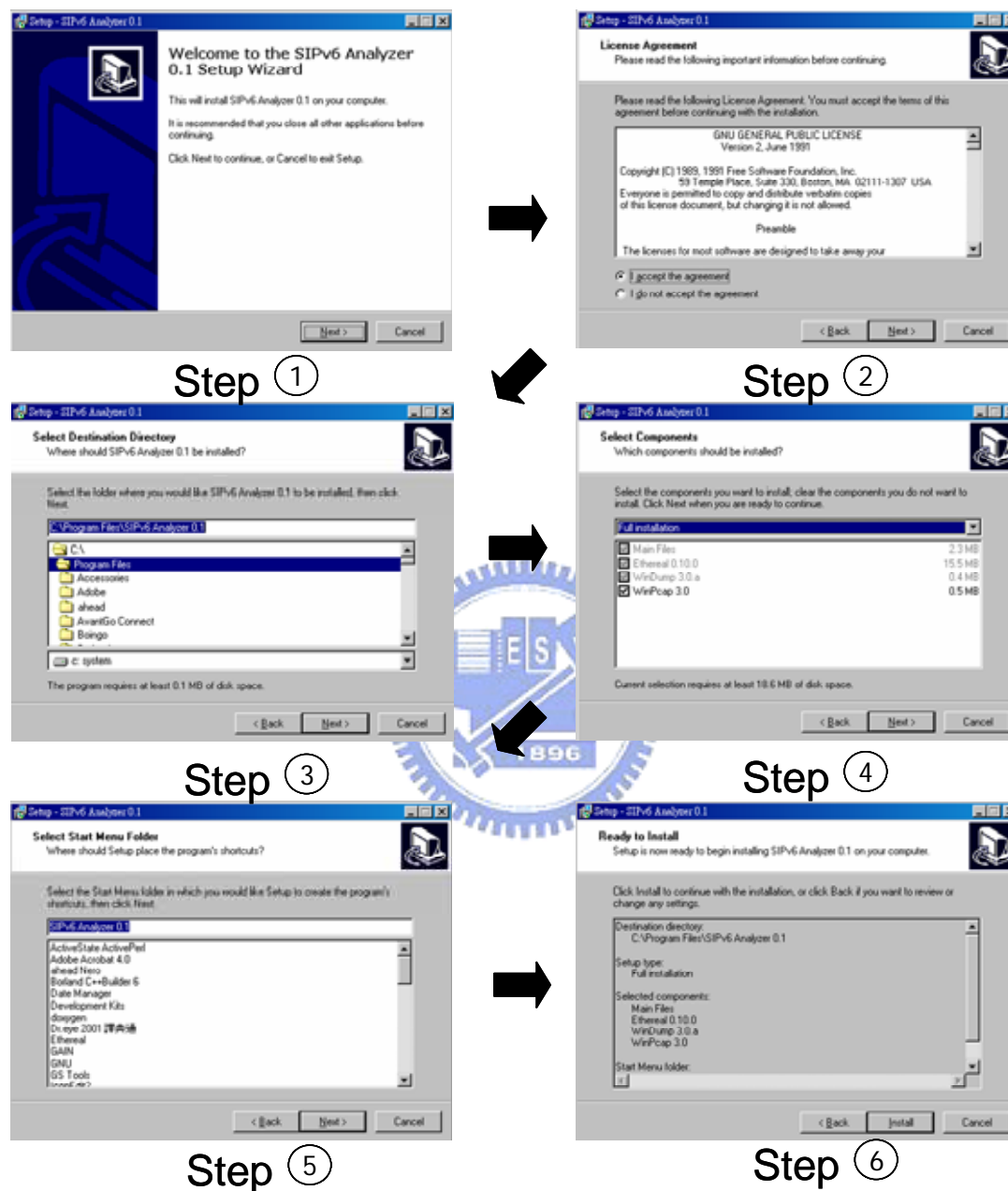


圖 4-1 系統安裝步驟流程圖

4.1.3 系統啟動

使用者點選「開始」按鈕，然後點選「程式集」，再點選「SIPv6 Analyzer 0.1」選單，最後點選「SIPv6 Analyzer 0.1」圖示就可以啟動本系統，如圖 4-2 所示。

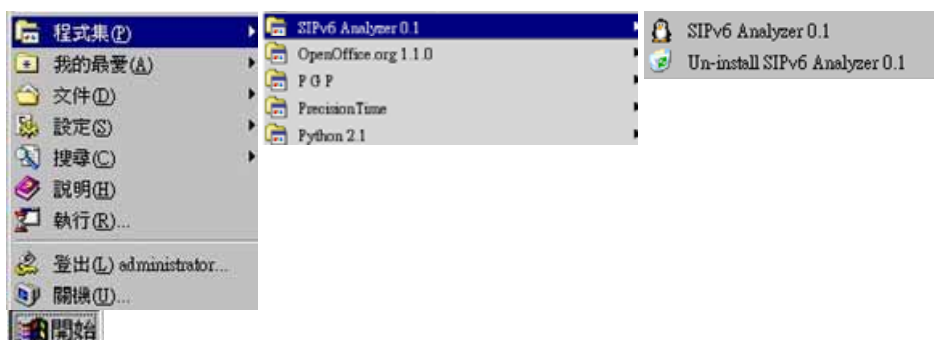


圖 4-2 系統啟動與系統移除選單圖

4.1.4 系統移除

使用者依照 4.1.3 節的操作步驟，最後改點選「Un-install SIPv6 Analyzer 0.1」圖示，就可以移除本系統。



4.2 系統操作說明

本節以功能為分類來說明 SIPv6 即時協定分析測試系統的操作方法。

4.2.1 系統 GUI 架構

本系統以 Multiple Document Interface (MDI) [21]做為 GUI 架構，如圖 4-3 所示。圖 4-3(a) MDI 母視窗的工作區域內會有三種類別的 MDI 子視窗，分別為圖 4-3(b) IUA/M2UA/M3UA 信令流程圖子視窗、圖 4-3(c)分析專案子視窗與圖 4-3(d) SIP 信令流程圖子視窗。

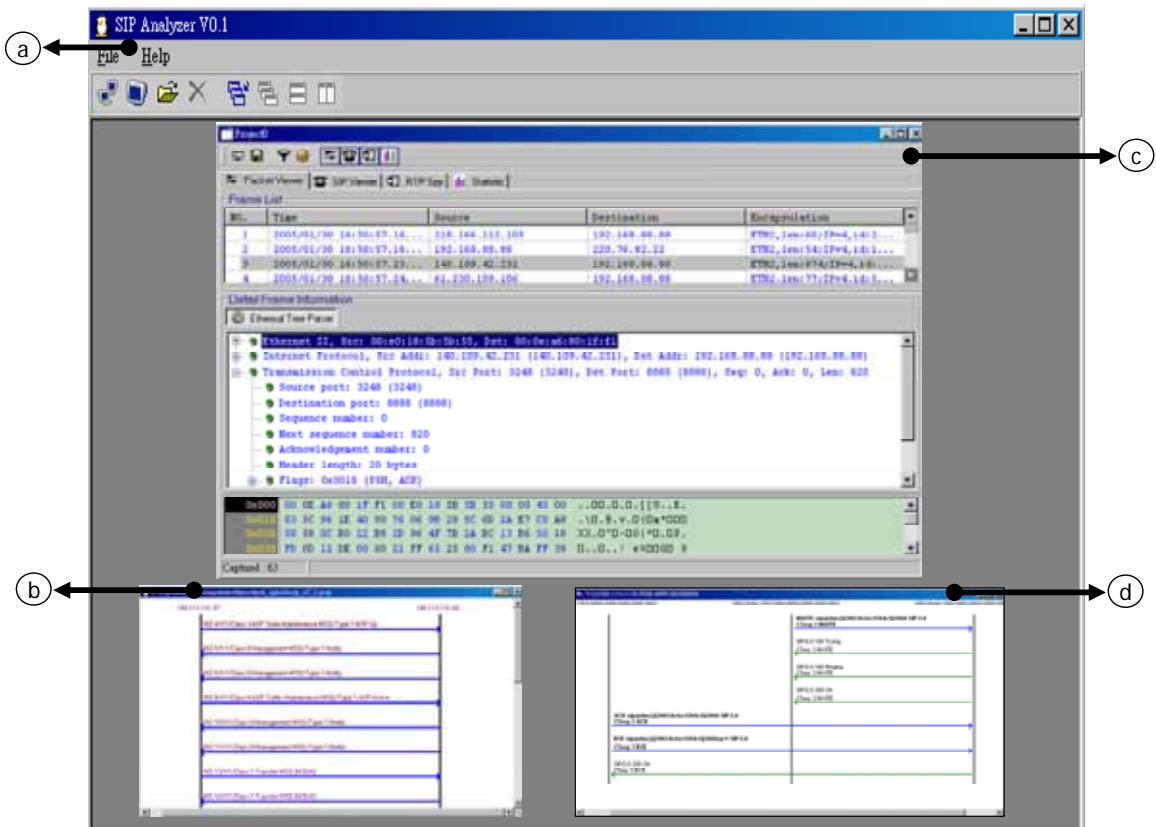


圖 4-3 系統 GUI 架構畫面圖

4.2.2 系統選單與系統工具列功能

系統選單包括檔案 (File) 與求助 (Help) 兩主選單，如圖 4-4(a) 與圖 4-4(b) 所示。檔案選單提供的選項功能為啟動遠端分析 (Remote Capture)、啟動本地端分析 (Local Capture)、啟動封包檔案分析 (Open Offline Packet)、關閉 MDI 子視窗 (Close Form) 及結束程式 (Quit)。求助選單提供的選項功能為啟動關於說明 (About)。

系統工具列如圖 4-4(c) 所示。系統工具列所提供的按鈕功能由左到右依序為啟動遠端分析、啟動本地端分析、啟動封包檔案分析、關閉 MDI 子視窗、切換 MDI 子視窗、並列 MDI 子視窗、平行並列 MDI 子視窗與垂直並列 MDI 子視窗。

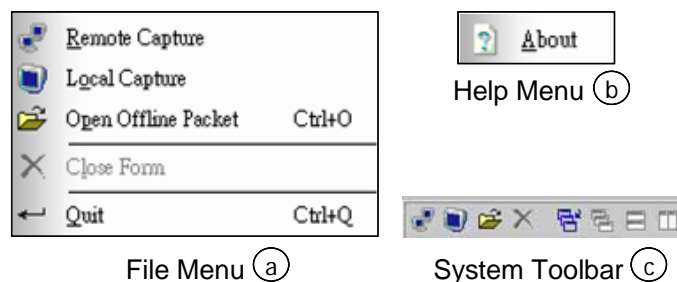


圖 4-4 系統選單與系統工具列畫面圖

4.2.3 分析專案子視窗

分析專案子視窗提供四種分析功能並對應到四個子頁面，分別是封包解析子頁面（Packet Viewer；圖 4-5(a)）、SIP 封包分析子頁面（SIP Viewer；圖 4-5(b)）、RTP 監控與播放子頁面（RTP Viewer；圖 4-5(c)）與流量/通訊協定統計子頁面（Statistic；圖 4-5(d)）。

使用者可以點選如圖 4-6(a)所示的子頁面顯示列來控制子頁面的顯示。使用者亦可以點選如圖 4-6(b)所示的子頁面切換列來進行子頁面的切換。

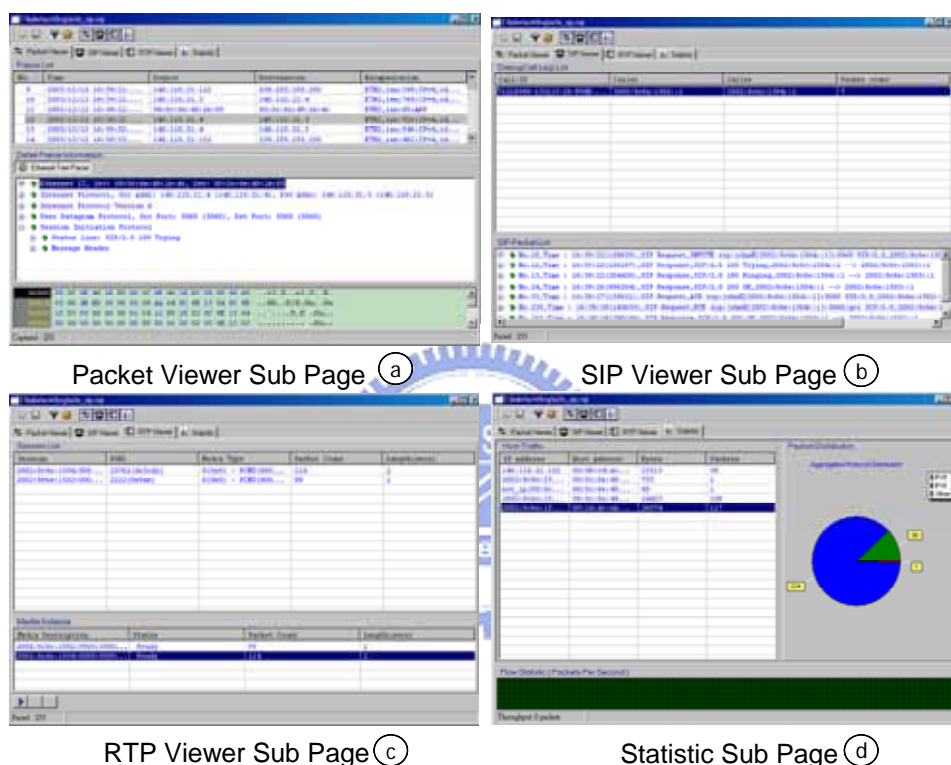


圖 4-5 分析專案子視窗的子頁面畫面圖

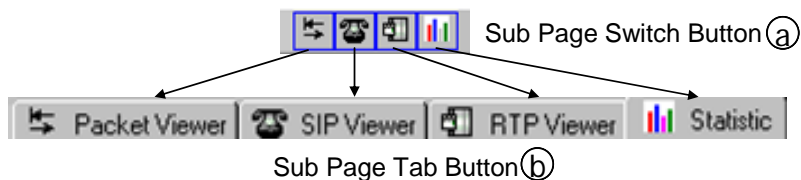


圖 4-6 分析專案子視窗的顯示/切換控制項畫面圖

4.2.4 分析專案子視窗之工具列控制項

分析專案子視窗之工具列控制項如圖 4-7 所示。工具列控制項中提供的按鈕功能由左到右依次為啟動/停止封包抓取、儲存封包檔案、啟動/停止顯示過濾器、

功能設定與 4.2.3 節說明的子頁面顯示列。



圖 4-7 分析專案子視窗之工具列控制項畫面圖

4.2.4.1 啟動/停止封包抓取



使用者可以點選此按鈕控制啟動與停止封包抓取的動作。

4.2.4.2 儲存封包檔案



圖 4-8 儲存封包檔案的展示畫面圖

當收集到封包框架，使用者可以點選此按鈕將目前收集的封包框架儲存成檔案。圖 4-8 展示點選儲存封包檔案鈕後彈出的儲存檔案對話盒。

4.2.4.3 啟動/停止顯示封包框架過濾



透過點選啟動/停止顯示過濾器按鈕，使用者可以對封包框架進行過濾以觀察所感興趣的封包框架。啟用顯示過濾器之前，必須先於功能選項中設定過濾規則 (Filter Rule)，請參考 4.2.4.4 節的說明。

4.2.4.4 功能選項



使用者可以點選功能選項按鈕以顯示功能選項對話盒。如圖 4-9 所示，虛線圓框標明啟動功能選項對話盒的按鈕，而功能選項對話盒上的虛線方框標明可以供使

用者設定的項目。功能選項對話盒中提供的設定項目說明如表 4-2 所列。

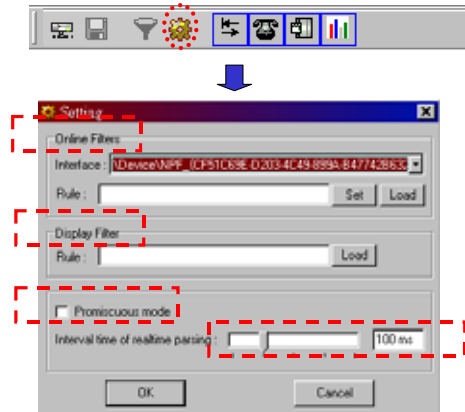


圖 4-9 功能選項對話盒的展示畫面圖

表 4-2 設定項目說明表

設定項目	用途
即時過濾器 (Online Filter)	本系統即時抓取封包框架的過濾規則
顯示過濾器 (Display Filter)	本系統顯示分析的封包框架過濾規則
全收模式 (Promiscuous Mode)	進入全收模式
即時解析間隔 (Interval Time of realtime parsing)	本系統解析封包框架的間隔

若要設定即時過濾器，使用者可以於即時過濾器 (Online Filters) 項目中先選取要設定過濾規則的網路介面 (Interface)，再於規則 (Rule) 文字輸入欄中輸入封包框架過濾器規則 (本系統所採取的封包框架過濾規則語法為 Pcap 語法格式 [34])，或透過點選載入 (Load) 按鈕載入已定義的過濾規則，最後點選設定 (Set) 按鈕使過濾規則生效。

若要設定顯示過濾器 (Display Filter)，使用者可以於顯示過濾器 (Display Filter) 項目中設定過濾規則，設定方法與設定即時過濾器相同，但無須選取網路介面。設定顯示過濾器後，使用者可以依照 4.2.4.3 節的說明來啟動顯示過濾器。

在設定即時過濾器或顯示過濾器中，使用者可以點選載入 (Load) 按鈕載入已定義的過濾規則。點選載入 (Load) 按鈕會出現如圖 4-10 所示的過濾器對話盒。過濾器對話盒中的過濾規則名稱列表 (Name List) 列出已定義的過濾規則，而過濾規則名稱 (Name) 與規則 (Rule) 可以供使用者新增或修改過濾規則名稱列表。

使用者可以點選儲存 (Save) 按鈕將新增或修改的過濾規則儲存在過濾規則名稱列表中。使用者可以點選刪除 (Delete) 按鈕刪除所選取的過濾規則。使用者可以

在過濾規則名稱列表中選取過濾規則名稱，然後點選載入（Load）按鈕來載入已定義的濾規則。使用者可以點選取消（Cancel）鈕來關閉過濾器對話盒。

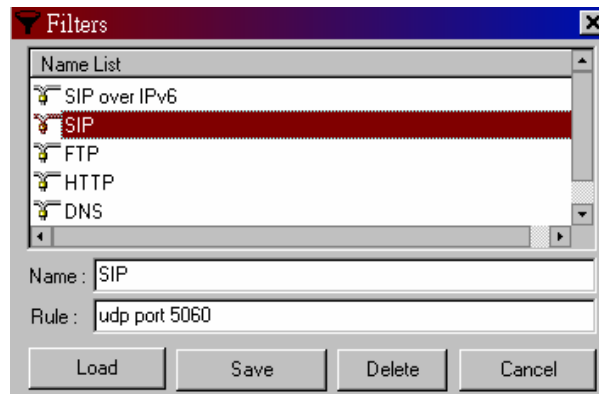


圖 4-10 過濾器對話盒畫面圖

4.2.5 分析專案模式

本系統的分析專案分為三種模式。第一種為本地端分析（Local Capture），即透過目前所在電腦的網路介面收集封包框架來進行分析；第二種是遠端分析（Remote Capture），即透過遠端電腦的網路介面收集封包框架來進行分析；第三種是封包檔案分析，即讀取 Pcap 格式 [34] 的封包檔案來進行分析。

接下來說明如何啟動這三種模式的分析專案。

4.2.5.1 本地端分析

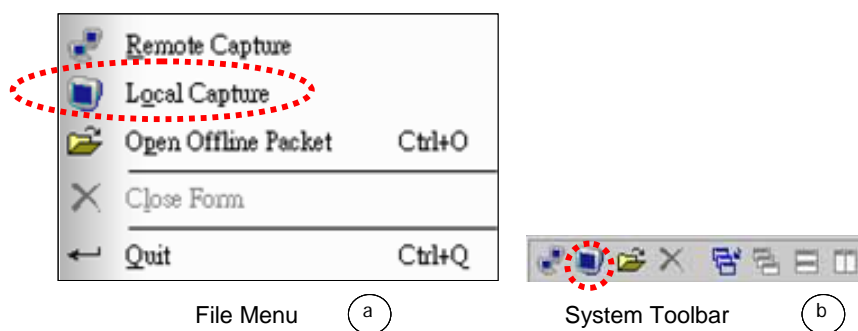


圖 4-11 啟動本地端分析的選項按鈕圖

使用者可以透過兩種方式啟動本地端分析，分別為點選 MDI 母視窗檔案選單中的啟動本地端分析（Local Capture）選項，或點選系統工具列中的啟動本地端分析按鈕，如圖 4-11(a)(b) 中的虛線圓框所標示。

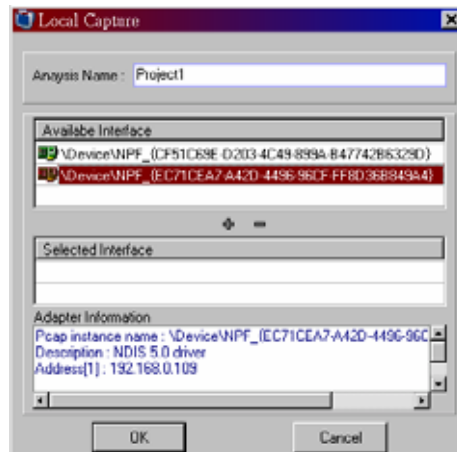


圖 4-12 本地端分析對話盒畫面圖

使用者啟動本地端分析時，會出現本地端分析（Local Capture）對話盒，如圖 4-12 所示。使用者必須在本地端對話盒中選取欲進行分析的網路介面後，才能開始本地端分析。首先在分析專案名稱（Analysis Name）文字輸入欄中輸入此分析專案的名稱，並透過雙擊分析網路介面（Available Interface）列表中的網路介面或點選加號（+）圖示選取欲進行分析的網路介面，而選取的網路介面將列出於選取網路介面（Selected Interface）列表中。雙擊選取網路介面（Selected Interface）列表中的網路介面或點選減號（-）圖示可以移除已選取的網路介面。介面卡資訊（Adapter Information）中，會列出目前被滑鼠選取的網路介面資訊。網路介面資訊包括裝置名稱（Pcap instance name）、描述（Description）、IP 位址（Address）與子網路遮罩（Net mask）。最後點選確認（OK）按鈕就開始本地端分析專案，接下來使用者可以進行 4.2.4 節所說明的操作。

4.2.5.2.遠端分析

使用者在啟動遠端分析前，要先確認被分析的遠端電腦已經啟動遠端封包抓取（Remote Packet Capture）服務，如圖 4-13 所示。遠端封包抓取服務包含在 WinPcap3.0 或 WinPcap3.0 以上的版本中。

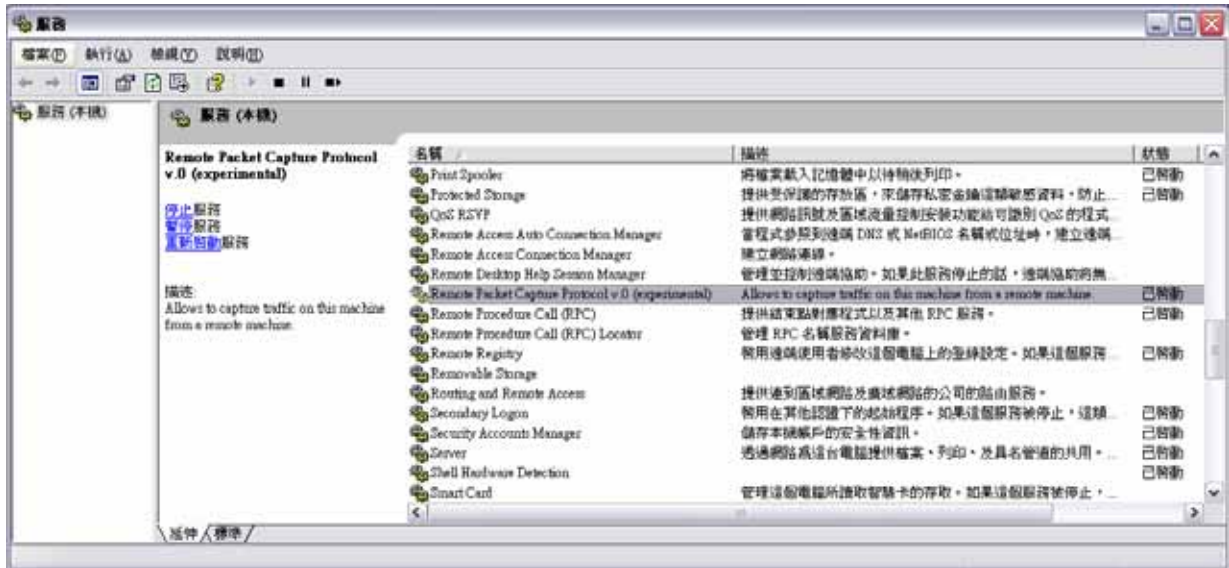


圖 4-13 遠端封包抓取服務狀態展示圖

使用者可以透過兩種方式啟動遠端分析，分別為點選 MDI 母視窗檔案選單中的啟動遠端分析（Remote Capture）選項，或點選系統工具列中的啟動遠端分析按鈕，如圖 4-14(a)(b)中的虛線圓框所標示。

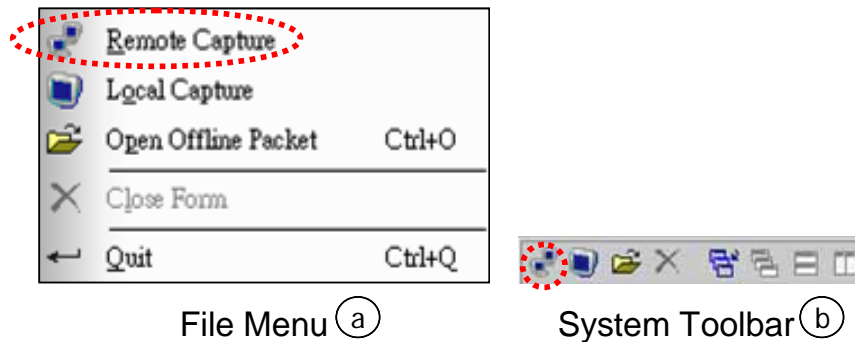


圖 4-14 啟動遠端分析的選項按鈕圖

使用者啟動遠端分析時，會出現遠端分析（Remote Capture）對話盒，如圖 4-15 所示。使用者必須在遠端對話盒中選取欲進行分析的遠端網路介面後，才能開始遠端分析。首先在分析專案名稱（Analysis Name）文字欄中填入此分析專案的名稱。再來使用者必須填入遠端主機的遠端主機位址（Remote Host）、埠號（Port）、

使用者名稱 (Username) 與密碼 (Password) 相關資訊，並點選掃描 (Scan) 按鈕進行遠端網路介面的掃描工作。成功完成掃描後，在分析網路介面 (Available Interface) 列表中將會出現可供分析的遠端網路介面。再來的操作步驟需將分析網路介面列表中的網路介面加入到選取網路介面列表 (Selected Interface) 中與 4.2.5.1 節的說明相同，本文不再贅述。

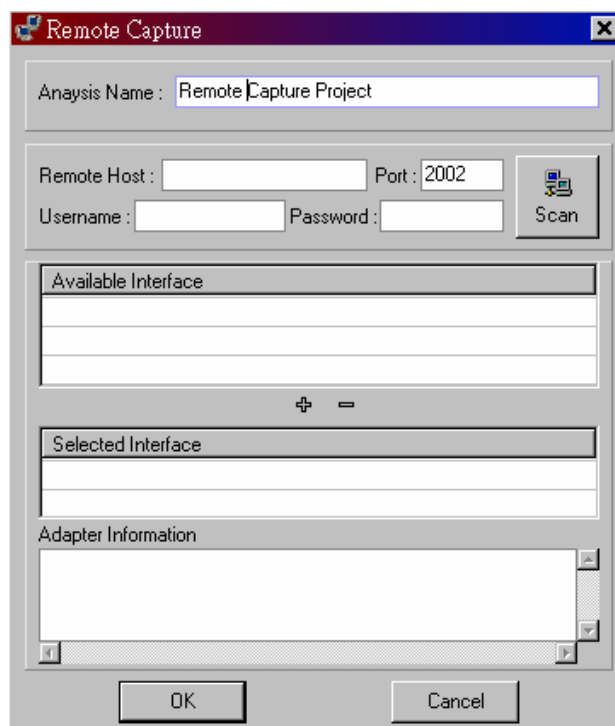


圖 4-15 遠端分析對話盒畫面圖

4.2.5.3 檔案分析

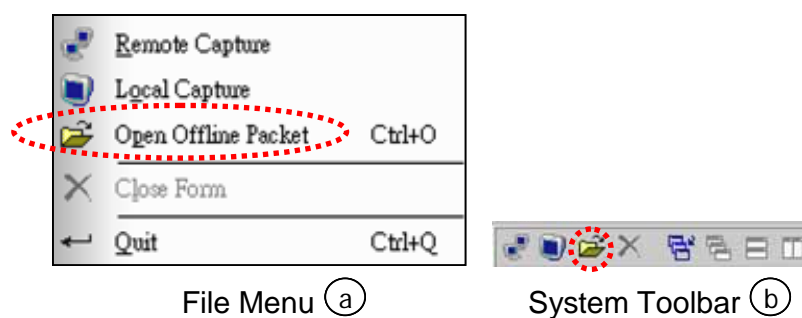


圖 4-16 啟動檔案分析的選項按鈕圖

使用者可以透過三種方式啟動檔案分析。點選 MDI 母視窗檔案選單中的檔案分析 (Open Offline Packet) 選項，或點選系統工具列中的檔案分析按鈕，如圖 4-16(a) (b) 中的虛線圓框所標示。最後一種方式是將檔案直接拖放至 MDI 母視窗的工作區。

4.2.6 分析專案功能

接下來將說明本系統所提供的五種分析功能。

4.2.6.1 封包解析 (Packet Viewer)



封包解析功能提供封包框架的解析資訊。封包解析子頁面提供封包解析功能，如圖 4-17 所示。

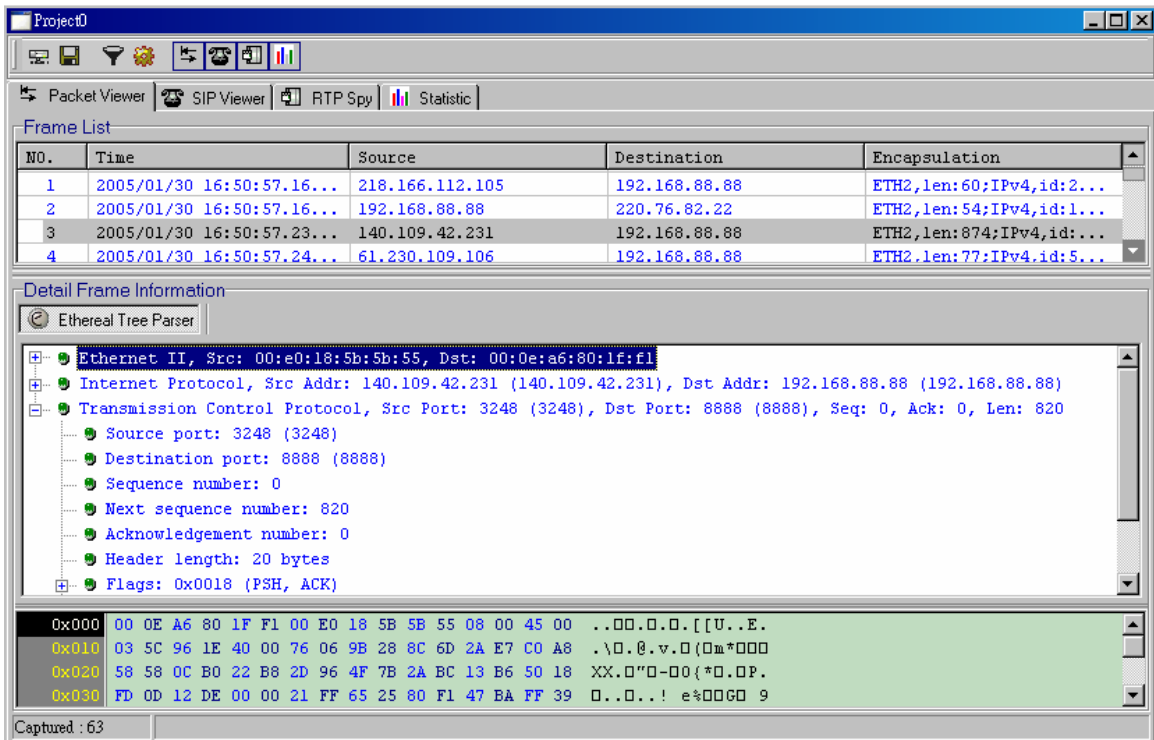


圖 4-17 封包解析子頁面畫面圖

使用者可以於封包框架列表 (Frame List) 中觀察到本系統目前所抓取的封包框架資訊。封包框架資訊包括編號 (NO.)、封包框架被抓取的時間 (Time)、來源位元址 (Source)、目的位元址 (Destination) 與封包框架封裝 (Encapsulation) 資訊。當封包框架為 IPv4 或 IPv6 封包，封包框架列表中的來源位元址與目的位元址會顯示封包的 IP 位址，否則會顯示封包的 MAC 位址。

使用者可以點選封包框架列表中的封包框架，此時詳細封包框架資訊 (Detail Frame Information) 將輸出 Ethernet 封包分析器對封包框架進行通訊協定 (Protocol) 解析的結果，同時最下層的文字輸出將更新封包框架的十六進位與 ASCII 內容輸出。

輸出是根據實體網路通訊 IP 來繪圖，而標頭流程圖繪圖是根據 SIP 信令內的標頭資訊來繪圖。SIP 流程圖與 SIP 標頭流程圖範例分別如圖 4-21(a)與(b)所示。

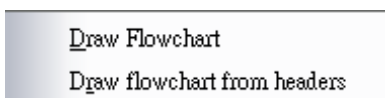
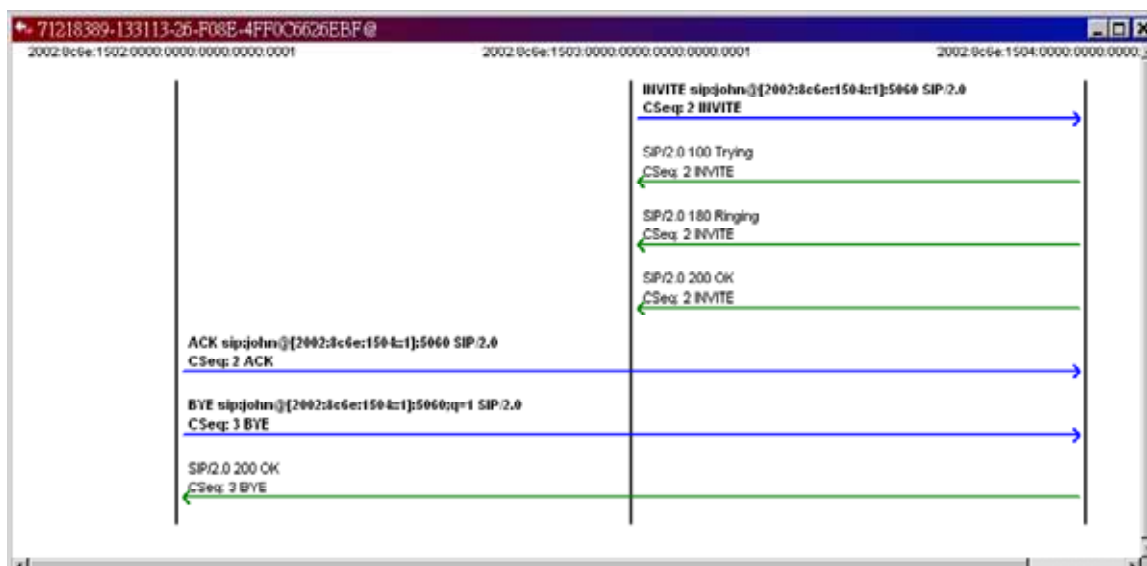
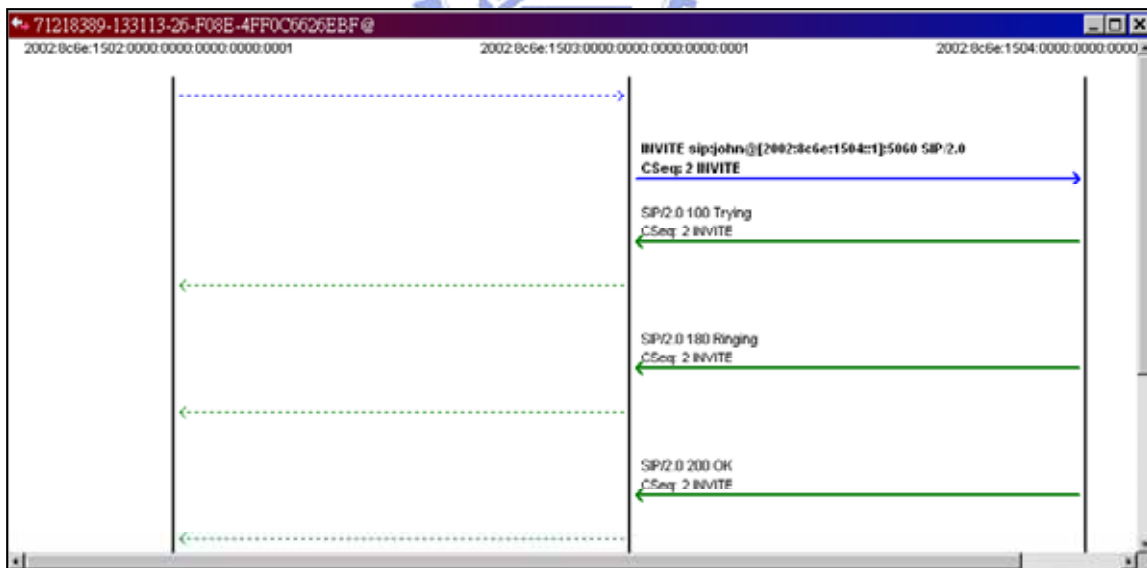


圖 4-20 SIP 信令流程圖選單圖



SIP flowchart (a)



SIP flowchart from headers (b)

圖 4-21 SIP 信令流程圖範例

4.2.6.3 RTP 監控與播放 (RTP Viewer)



RTP 監控與播放功能提供 RTP 串流監控與播放。RTP 監控與播放子頁面提供 RTP 監控與播放功能，如圖 4-22 所示。

使用者可以於會期列表 (Session List) 中觀察到本系統目前所分析出的 RTP 串流會期資訊。RTP 串流會期資訊包括會期名稱 (Session)、SSRC (SSRC) 串流編碼格式 (Media Type)、串流中的封包個數 (Packet Count) 與串流長度 (Length) 資訊。

使用者可以雙擊會期列表中的 RTP 串流將本系統所支援的編碼格式串流加入多媒體實例 (Media Instance) 列表中。本系統目前支援 PCMA 與 PCMU 兩種編碼格式的串流。多媒體實例列表下方的播放控制項可以用來控制播放、暫停與停止播放所選取的多媒體實例。使用者可以按下鍵盤的刪除 (Delete) 鍵將選取的多媒體實例移出多媒體實例列表。

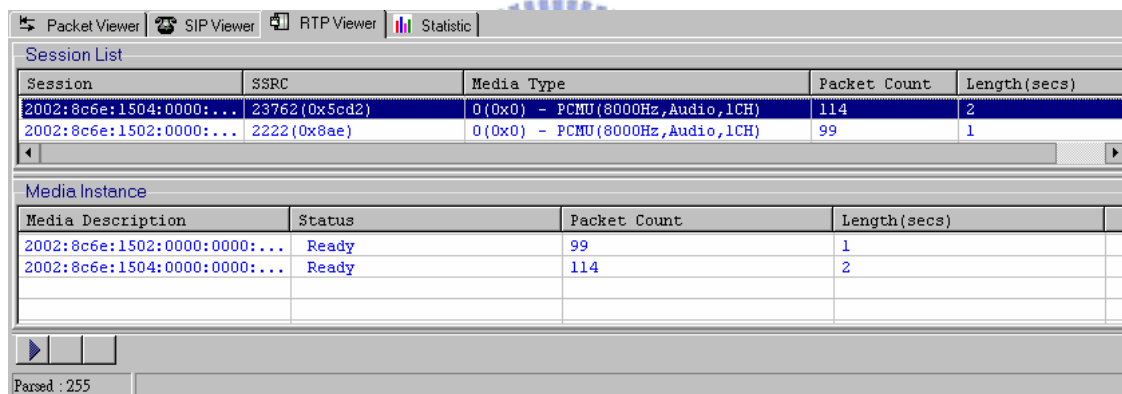


圖 4-22 RTP 監控與播放子頁面畫面圖

4.2.6.4 IUA/M2UA/M3UA 信令流程圖

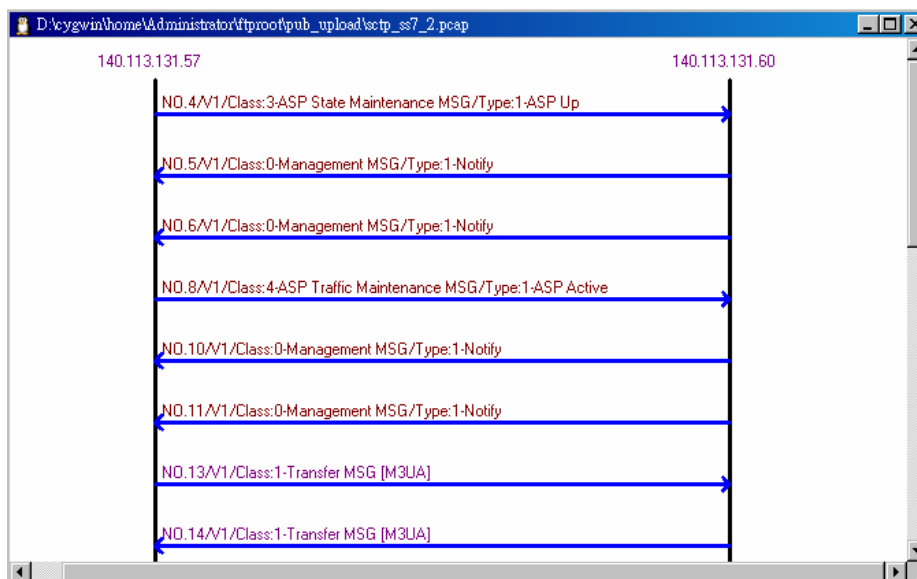


圖 4-23 IUA/M2UA/M3UA 信令流程圖

IUA、M2UA 或 M3UA 信令封包信令流程圖如圖 4-23 所示。IUA/M2UA/M3UA 信令流程圖以圖形介面協助使用者觀察 IUA/M2UA/M3UA 信令流程圖。

4.2.6.5 流量與通訊協定統計 (Statistic)

流量與通訊協定統計功能提供流量監控與封包分佈狀態。流量與通訊協定統計子頁面提供流量與通訊協定統計功能，如圖 4-24 所示。

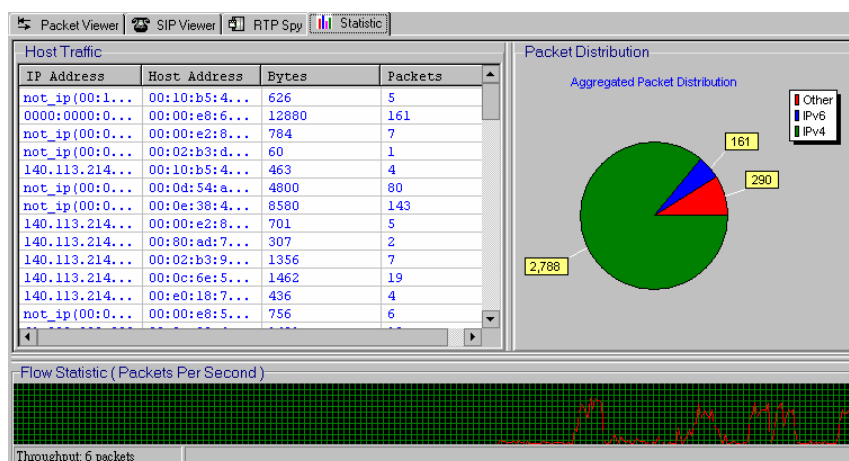


圖 4-24 流量與通訊協定統計子頁面畫面圖

使用者可以在主機流量 (Host Traffic) 列表中觀察到主機的流量資訊。流量資訊包括傳送的位元組 (Bytes) 數與封包框架個數 (Packets)。封包分佈 (Packet Distribution) 提供封包分佈圓餅圖讓使用者瞭解封包分佈的狀態。流量統計 (Flow Statistic) 方格圖會隨時間變動繪出流量變動圖。狀態列的效能輸出 (Throughput) 則顯示目前的瞬間封包框架擷取量。

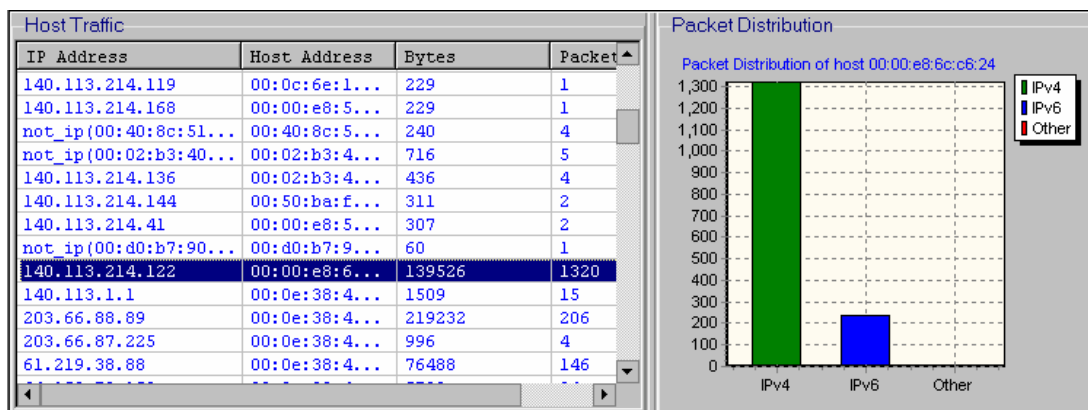


圖 4-25 主機封包分佈長條圖

使用者可以點選主機流量 (Host Traffic) 列表中的特定主機，此時封包分佈會以長條圖繪出該主機的封包分佈，如圖 4-25 所示。使用者可以於封包分佈中敲擊滑鼠右鍵以切換回原封包分佈原餅圖。



五、展示案例

本節以 SIP/IPv6 封包分析展示案例來說明 SIPv6 即時協定分析測試系統的功能特點。SIP/IPv6 封包分析案例主要展示的功能特點包括遠端分析、6to4 通道封包分析、SIP 信令分析、SIP 信令流程圖形化、進階 SIP 信令流程圖形化、RTP 語音串流分析與混音播放功能。

5.1 SIP/IPv6 封包分析

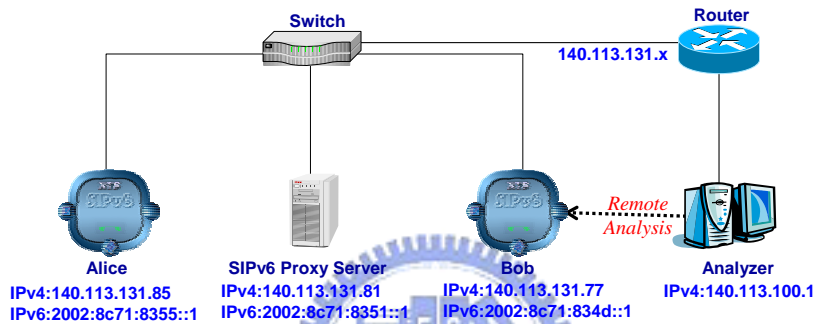


圖 5-1 SIP/IPv6 封包分析展示案例的環境架構圖

SIP/IPv6 封包分析展示案例的環境架構圖與 IP 位址設定如圖 5-1 所示。本展示的四台電腦主機包含 SIPv6 UA(文後簡稱 Alice)、SIPv6 Proxy Server、SIPv6 UA(文後簡稱 Bob)與 Analyzer。Alice 與 Bob 端分別執行 IPv6-Enabled SIP UA (此 UA 軟體由工研院開發，本實驗室修改為 IPv6-Enabled)。SIPv6 Proxy Server 端執行 IPv6-Enabled SIP Proxy Server(此伺服器軟體為 PartySIP [23])。Analyzer 端執行 SIPv6 即時協定分析測試系統。展示案例為由 Analyzer 端對 Bob 端進行遠端分析。在進行遠端分析的同時，Alice 會透過 SIPv6 Proxy Server 與 Bob 進行 SIP VoIP 通訊。Alice、SIPv6 Proxy Server 與 Bob 三台電腦主機會以 6to4 通道進行 IPv6 通訊。



圖 5-2 遠端封包抓取服務圖

首先在 Bob 端啟動遠端封包抓取服務，如圖 5-2 所示，然後 Analyzer 端開始對 Bob 端進行遠端分析。首先於 Analyzer 端的遠端抓取 (Remote Capture) 對話盒中填入 Bob 的電腦主機資訊 (包括主機位址、遠端封包抓取服務埠號、使用者帳號與密

碼)，再點選掃描按鈕列出遠端網路介面，然後選取欲進行觀察的遠端網路介面。最後點選確認按鈕開始遠端分析。開始遠端分析的過程如圖 5-3 所示。

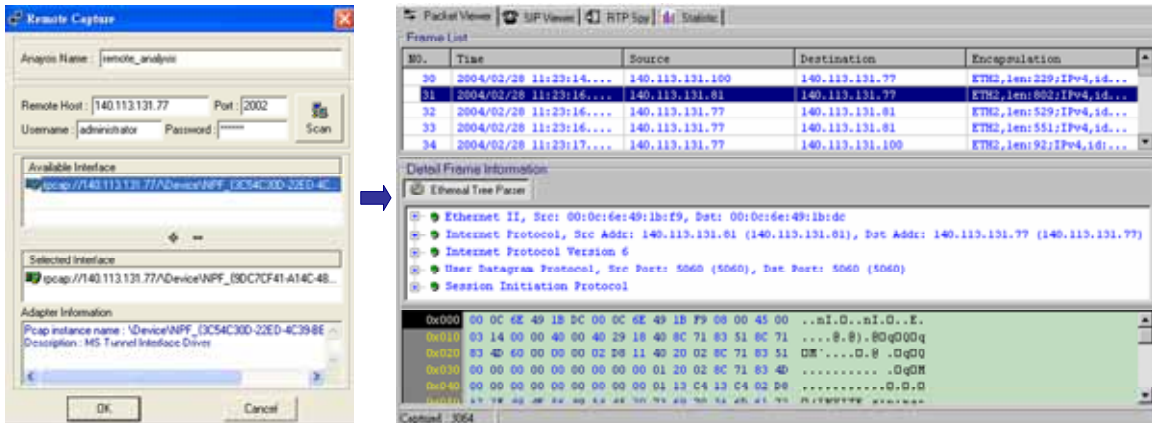
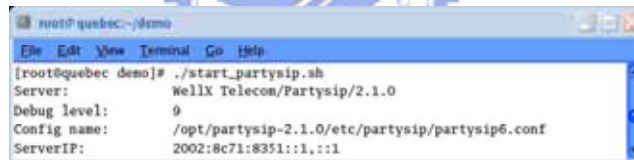


圖 5-3 Bob 端進行遠端分析圖

在 Alice 與 Bob 進行 SIP VoIP 通訊之前，SIPv6 Proxy Server 端先啟動 SIP Proxy Server，如圖 5-4(a)所示。然後 Alice 端設定其 SIP UA 使用的 SIP Proxy Server 為 SIPv6 Proxy Server，如圖 5-4(b)所示。最後 Alice 與 Bob 開始進行 SIP VoIP 通訊。

圖 5-4 SIP Proxy Server 與 SIP Proxy Server 設定圖

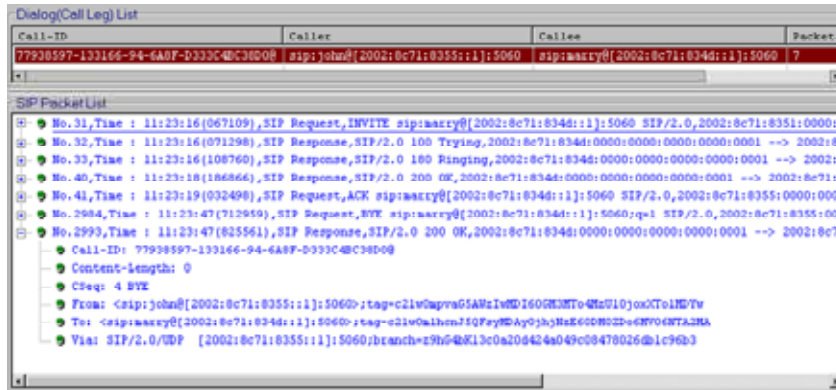


Run SIP Proxy Server(a)

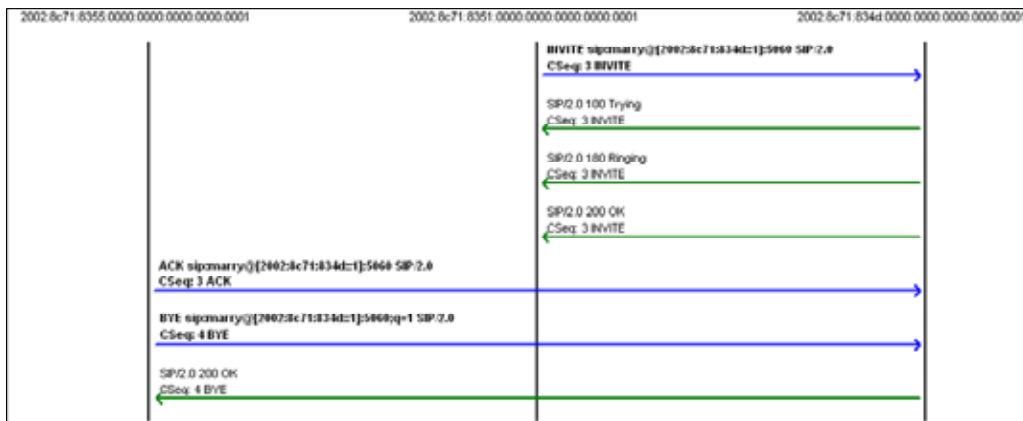


Assign SIP Proxy Server(b)

從 Alice 與 Bob 的 SIP VoIP 通訊之開始到結束的過程，Analyzer 會對 Bob 端進行遠端分析。SIPv6 即時協定分析測試系統在 Analyzer 端所的分析結果如下說明。



SIP Viewer Sub Page (a)



SIP Flowchart (I) (b)



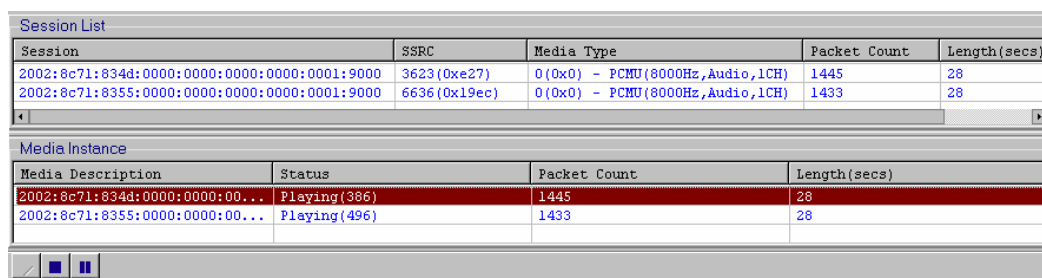
SIP Flowchart (II) (c)

圖 5-5 SIP Viewer 子頁面與 SIP 信令流程圖輸出圖

SIP Viewer 能將網路上的 SIP 封包以對話做分類並整理成 SIP 封包列表，方便使用者進行信令觀察，這是一般封包解析軟體沒有提供的（例如 Ethereal、WinDump 及 TCPDump 等）。如圖 5-5(a) 所展示，SIP Viewer 子頁面提供了 Call-ID、Caller（即 Alice）及 Callee（即 Bob）資訊，亦可以得知 SIP VoIP 通訊中 Bob 端共有七個進出的 SIP 信令封包。SIP Viewer 子頁面中的 SIP 封包列表（SIP Packet List）提供詳細的 SIP 封包資訊，透過展開 SIP 封包列表中的 SIP 封包可以直接觀察其 SIP 信令內容，如圖 5-5(a) 顯示 SIP 封包列表中所展開的 SIP 封包。

SIP 信令流程圖能讓使用者更容易的觀察到 SIP 信令的交換流程。圖 5-5(b)(c) 為 SIP 信令流程圖輸出。由流程圖中的 IPv6 位址資訊可以得知流程圖中由左到右的電腦主機節點分別是 Alice、SIPV6 Proxy Server 與 Bob。基本 SIP 信令流程圖顯示所有進出 Bob 端之 SIP 信令的流程，如圖 5-5(b) 所示。進階 SIP 信令流程圖顯示 Alice、SIPV6 Proxy Server 與 Bob 三者間的 SIP 信令流程，如圖 5-5(c) 所示。進階 SIP 信令流程圖中的虛線路徑是根據解析 SIP 信令中的 Via 與 Route 標頭欄位值來產生。透過進階 SIP 信令流程圖，我們可以完整的觀察到此次 SIP VoIP 通訊的所有 SIP 信令交換流程。

RTP Viewer 將解析網路封包是否為 RTP 封包並整理成 RTP 串流供使用者觀察，亦能進行播放混音，以評估即時通訊品質。如圖 5-6 所示，RTP Viewer 子頁面提供 Alice 與 Bob 的 RTP 串流資訊。由會期列表（Session List）中的會期（Session）欄位資訊可以得知會期列表中的第一項會期為 Bob 的 RTP 串流，第二項會期為 Alice 的 RTP 串流。多媒體實例（Media Instance）列表中的 RTP 串流實例可以進行播放混音功能。



Session List				
Session	SSRC	Media Type	Packet Count	Length(secs)
2002:8c71:834d:0000:0000:0000:0001:9000	3623(0xe27)	0(0x0) - PCMU(8000Hz,Audio,1CH)	1445	28
2002:8c71:8355:0000:0000:0000:0001:9000	6636(0x19ec)	0(0x0) - PCMU(8000Hz,Audio,1CH)	1433	28

Media Instance			
Media Description	Status	Packet Count	Length(secs)
2002:8c71:834d:0000:0000:00...	Playing(386)	1445	28
2002:8c71:8355:0000:0000:00...	Playing(496)	1433	28

圖 5-6 RTP Viewer 子頁面圖

六、結論與未來工作

本論文針對第三代行動通訊的核心協定 SIP 與 IPv6 所設計實作之 SIPv6 即時協定分析測試系統主要提供遠端分析、封包分析、SIP 信令分析、SIP 信令流程圖形化、RTP 串流分析、IUA/M2UA/M3UA 信令流程圖形化與流量統計功能。遠端分析功能讓使用者可以於近端進行遠端網路分析的工作。封包分析功能提供 Ethereal 封包分析器進行封包協定解析。SIP 信令分析功能提供使用者觀察以對話分類的 SIP 信令封包。SIP 信令流程圖形化功能以圖形化模式協助使用者觀察 SIP 信令流程。RTP 串流分析功能提供 RTP 串流蒐集、監看與監聽。流量統計功能讓使用者瞭解目前的封包分佈狀態。IUA/M2UA/M3UA 信令流程圖形化以圖形化模式協助使用者觀察 IUA/M2UA/M3UA 信令流程。

本系統的研發成果為贏得 2003 年第三屆國家高速電腦中心軟體程式設計競賽冠軍，並由宋岳鑫學弟進行解析功能擴充贏得 2004 年日本 IPv6 Appli- Contest 2004 實作組冠軍之殊榮。目前亦做為 IPv6 與 SIP-based VoIP 通訊教育改進計畫的實驗教學教材。

本系統與其他封包分析軟體的最大差異在於整合信令協定流程與多媒體串流的分析功能。未來本系統的功能擴充為 SIP 信令產生器、SIP 信令偵錯分析、RTP 串流型態支援擴充與 Secure RTP (SRTP) 分析。

使用者可以自訂 SIP 信令產生流程，而 SIP 信令產生器將根據自訂流程來產生 SIP 信令封包。透過 SIP 信令產生器功能使用者可以進行 SIP 原件模擬或進行 SIP 原件壓力測試等工作。

SIP 信令偵錯分析功能將協助使用者進行 SIP 信令封包的錯誤分析。透過 SIP 信令偵錯分析功能可以協助 SIP 應用系統開發人員進行 SIP 信令的錯誤分析以加速系統開發。

RTP 串流型態支援擴充為擴充本系統所支援播放的 RTP 串流格式。SRTP 為 RTP 加上安全機制。SRTP 分析功能為支援分析 SRTP 串流。藉由 SRTP 的分析來說明安全通訊的重要性。

參考文獻

- [1] 3GPP. Technical Specification Group Services and System Aspects; IP Multimedia Subsystem (IMS). 3GPP TS 23.228, v.6.0.1, (2003-01)
- [2] B. Carpenter, K. Moore. Connection of IPv6 Domains via IPv4 Clouds. IETF RFC3056, Feb. 2001.
- [3] C. Huitema. Teredo: Tunneling IPv6 over UDP through NATs. IETF Internet Draft, Feb. 2004.
- [4] C++ Standard Template Library, <http://www.stlport.org/>
- [5] Ethereal, <http://www.ethereal.com/>
- [6] F. Yergeau, Alis Technologies. UTF-8, a transformation format of ISO 10646. IETF RFC2279, Jan. 1998.
- [7] G. Sidebottom, K. Morneault, J. Pastor-Balbas, et al. Signaling System 7 (SS7) Message Transfer Part 3 (MTP3) - User Adaptation Layer (M3UA). IETF RFC3332, Sep. 2002.
- [8] GNU oSIP2, <http://www.gnu.org/software/osip/>
- [9] H. Schulzrinne, S. Casner, R. Frederick, et al. RTP: A Transport Protocol for Real-Time Applications. IETF RFC3550, Jul. 2003.
- [10] Hammer Call Analyzer, <http://www.empirix.com/default.asp?action=article&ID=69>
- [11] IETF SIGTRAN WG, <http://www.ietf.org/html.charters/sigtran-charter.html>
- [12] J. Postel. Internet Protocol. IETF RFC791, Sep. 1980.
- [13] J. Postel, J. Reynolds, ISI. File Transfer Protocol. IETF RFC959, Oct. 1985.
- [14] J. Postel. Transmission Control Protocol. IETF 793, Jan. 1980.
- [15] J. Postel. User Datagram Protocol. IETF 768, Aug. 1980.
- [16] J. Rosenberg, H. Schulzrinne, G. Camarillo, et al. SIP: Session Initiation Protocol. IETF RFC3261, Jun. 2002.
- [17] K. Egevang, P. Francis, NTT. The IP Network Address Translator (NAT). IETF RFC1631, May. 1994.
- [18] K. Morneault, R. Dantu, G. Sidebottom, et al. Signaling System 7 (SS7) Message Transfer Part 2 (MTP2) - User Adaptation Layer. IETF RFC3331, Sep. 2002.
- [19] K. Morneault, S. Rengasami, M. Kalla, et al. ISDN Q.921-User Adaptation Layer. IETF RFC3057, Feb. 2001.
- [20] M. Handley, V. Jacobson, ISI/LBNL. SDP: Session Description Protocol. IETF RFC2327, Apr. 1998.
- [21] MSDN, <http://msdn.microsoft.com/library/>

- [22] P. Mockapetris, ISI. Domain names - Implementation and Specification. IETF RFC1035, Nov. 1987.
- [23] PartySIP, <http://www.nongnu.org/partysip/partysip.html>
- [24] R. Fielding, UC Irvine, J. Gettys, et al. Hypertext Transfer Protocol -- HTTP/1.1. IETF RFC2068, Jan. 1997.
- [25] R. Gilligan, E. Nordmark, Sun Microsystems. Transition Mechanisms for IPv6 Hosts and Routers. IETF RFC2893, Aug.2000.
- [26] R. Stewart, Q. Xie, K. Morneault, et al. Stream Control Transmission Protocol. IETF RFC2960, Oct. 2000.
- [27] S.Deering, R. Hinden. Internet Protocol, Version 6 (IPv6) Specification. IETF RFC2460, Dec. 1998.
- [28] S.Kent, R. Atkinson.IP Authentication Header. IETF RFC2402, Nov.1998.
- [29] S.Kent, R. Atkinson. IP Encapsulating Security Payload (ESP) . IETF RFC2406,
- [30] Silvia Hagen. IPv6 Essentials. Oreilly. 2002.
- [31] SnifferPro, <http://www.asl-sniffer.co.uk/>
Nov.1998.
- [32] TcpDump, <http://www.tcpdump.org/>
- [33] WinDump, <http://windump.polito.it/>
- [34] WinPcap, <http://winpcap.polito.it/>



附錄 A 系統需求與規格

A.1 系統需求

- 提供人性化的圖形介面
- 讓其他程式設計師能輕易使用統一的架構開發新模組
- 提供與開放原始碼封包分析器相容的儲存格式
- 提供 On-line/Off-line 封包過濾功能
- 提供跨網路之封包擷取與分析
- 分析 RFC2460 所定義之 IPv6 封包標頭
- 分析 IP-in-IP 形式的通道封包
- 進階分析 Teredo 通道封包
- 產生圖形化 SIP 信令流程圖與 SIP 對話報表
- 提供監聽多媒體串流(RTP)封包功能
- 提供通訊協定與主機流量統計
- 提供程式人性化的安裝與反安裝功能

2 系統規格

- 執行平臺 Microsoft Windows 2000/XP/2003
- 支援乙太網路(10/100/1000Mbps Ethernet)與 802.11 無線網路卡
- 支援開放原始碼 Ethereal 協定解析器
- 封包過濾功能採 BPF 中所制定之規則語法
- 封包存檔格式遵循 PCAP 格式
- 支援撥放之 RTP 多媒體格式包括 PCMU 與 PCMA
- RTP 多媒體撥放方面採統一的程式介面與架構供其他程式設計師可以加入開發(新增可撥放的多媒體格式)
- 遠端封包抓取功能採 WinPcap3.0 之 rpcap
- 符合 Windows 系統安裝與反安裝標準

附錄 B 6to4 通道與 Teredo 協定技術

SIPv6 即時協定分析測試系統支援解析 6to4 通道與 Teredo 兩種以建立通道方法為主的協定技術。本節簡單介紹 6to4 通道與 Teredo 兩種協定技術與標頭格式。

目前 IPv4 仍為主要的網路層協定。由於 IPv4 與 IPv6 應用程式的相容性與網路設備的 IPv6 支援度等問題，因此無法直接以 IPv6 取代現存多數的 IPv4。而這也使得 IPv4 與 IPv6 會共存一段時間。

目前有三大類技術能讓 IPv4 與 IPv6 能夠共存並提供互通性。這三大類技術分別為雙堆疊（Dual Stack）、建立通道與轉換（Translation）。雙堆疊讓 IPv4 與 IPv6 共存於同一裝置與網路上。建立通道讓 IPv6 封包能在 IPv4 基礎架構上傳遞。轉換能讓 IPv6-only 主機與 IPv4-only 主機相互通訊。[3]

B.1 6to4 通道

6to4 通道技術為上文所說明的一種建立通道技術。6to4 通道技術讓網路主機能在 IPv4 基礎架構上進行 IPv6 通訊。首先說明 6to4 位址前置 (Prefix) 碼與 6to4 位址。6to4 位址前置碼為 2002:IPv4_Address::/48 格式的 IPv6 位址前置碼。比如擁有 16.32.48.64 IPv4 位址的路由器可產生 2002:1020:3040::/48 的 6to4 位址前置碼。6to4 位址為以 6to4 位址前置碼為前置的 IPv6 位址。6to4 位址的後 80 位元可由網管人員自行進行規劃使用。

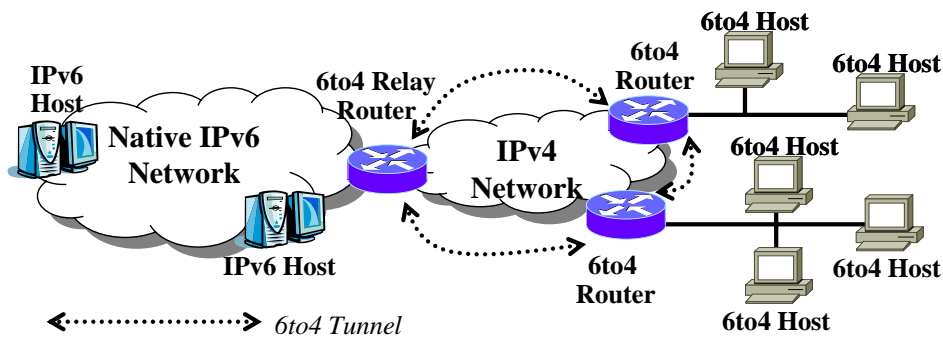


圖 B-1 6to4 架構圖

6to4 的架構如圖 B-1 所示。6to4 通道技術中的三個主要網路元件為 6to4 主機、6to4 路由器與 6to4 轉達 (Relay) 路由器。6to4 主機為擁有 6to4 位址的主機。6to4 路由器負責建構通道以協助位於其內端的 6to4 主機於 IPv4 基礎架構上進行 IPv6 通訊。6to4 轉達路由器負責建構通道以協助原生 IPv6 網路中的 IPv6 主機於 IPv4 基

基礎架構上進行 IPv6 通訊。6to4 通道封包的封裝格式定義於 RFC 2893，採用 IPv6-in-IPv4 的封裝格式。

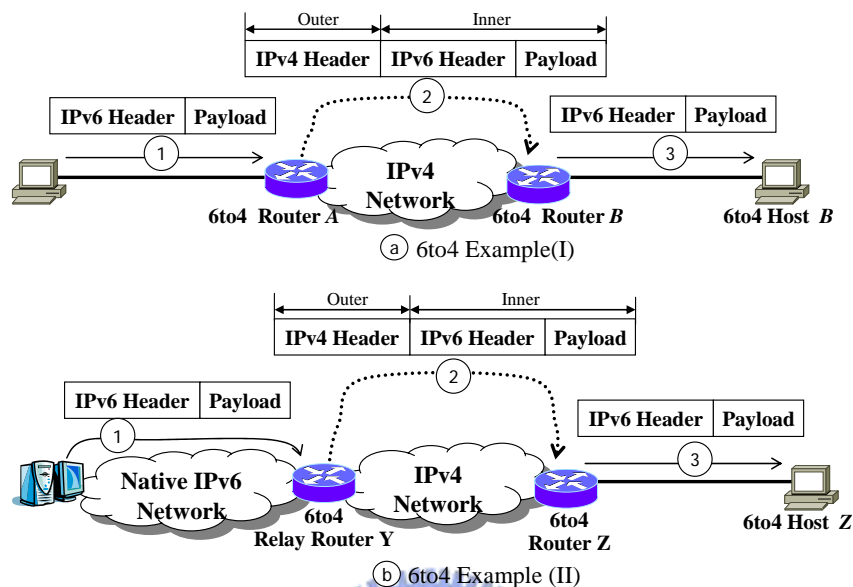


圖 B-2 6to4 運作範例圖

圖 B-2(a)說明 6to4 主機如何在 IPv4 基礎架構上傳送 IPv6 封包。6to4 主機 A 要傳送 IPv6 封包至 6to4 主機 B。首先 6to4 主機 A 發出 IPv6 封包至 6to4 路由器 A，如圖 B-2(a)①所示。6to4 路由器 A 由 IPv6 封包中的目的 6to4 位元址(即 6to4 主機 B 的 6to4 位址)得知通道終點(6to4 位址前置碼中的 IPv4_Address)，並建立通道至 6to4 路由器 B，如圖 B-2(a)②所示。6to4 路由器 B 將解封裝的通道封包傳送至 6to4 主機 B，如圖 B-2(a)③所示。圖 B-2(b)說明 IPv6 主機與 6to4 主機於 IPv4 基礎架構上傳送 IPv6 封包。IPv6 主機 X 要傳送 IPv6 封包至 6to4 主機 Z。首先 IPv6 主機 X 發出 IPv6 封包至 6to4 轉達路由器 Y，如圖 B-2(b)①所示。6to4 轉達路由器 Y 由 IPv6 封包中的目的 6to4 位元址(即 6to4 主機 Z 的 6to4 位址)得知通道終點，並建立通道至 6to4 路由器 Z，如圖 B-2(b)②所示。6to4 路由器 Z 將解封裝的通道封包傳送至 6to4 主機 Z，如圖 B-2(b)③所示。

B.2 Teredo

採 6to4 通道或 Automatic 通道 [25]的主機至少需有一個公用 (Public) 且可路由 (Routable) 的 IPv4 位址才能進行建立通道，因此位於 NAT 後端私有網路並擁有私有 (Private) IPv4 位址的主機就無法以這些建立通道技術來進行 IPv6 通訊。Teredo 技術讓這些主機能與 NAT 外端的 IPv6 網路進行通訊。

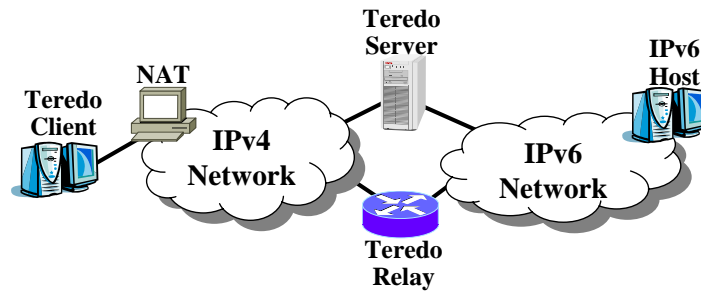


圖 B-3 Teredo 架構圖

Teredo 的架構如圖 B-3 所示。Teredo 中的主要三個網路元件分別為 Teredo Client、Teredo Server 與 Teredo Relay。Teredo Client 是位於 NAT 後端且欲與 NAT 外端進行 IPv6 通訊的主機。Teredo Server 協助 Teredo Client 連接 IPv6 網路與建立 Teredo IPv6 位址 [1]。Teredo Relay 負責轉達 Teredo Client 與 IPv6 網路之間的 IPv6 封包。在 Teredo 的運作過程首先 Teredo Client 向 Teredo Server 取得建立 Teredo IPv6 位址的資訊，並建立 Teredo IPv6 位址，然後 Teredo Client 與 Teredo Relay 建立通道。最後 Teredo Client 透過通道與 IPv6 網路進行通訊。

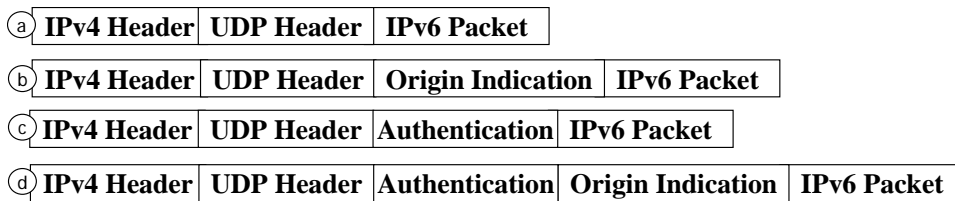


圖 B-4 Teredo 封裝格式圖

Teredo 通道的封裝格式分為四種，如圖 B-4 所示。封裝型態 a 用來傳遞一般 IPv6 封包。封裝型態 b 主要用於 Teredo Client 向 Teredo Server 取得建立 Teredo IPv6 位址的資訊。封裝型態 c 是封裝型態 a 加上安全認證功能。封裝型態 d 是封裝型態 b 加上安全認證功能。

Origin Indication 與 Authentication 標頭格式如圖 B-5 所示。Origin Indication 標頭提供 Teredo Client 建立 Teredo IPv6 位址的資訊。Origin port 標頭欄位是 Teredo Client 於 NAT 外端的對應埠號值。Origin IPv4 Address 標頭欄位是 Teredo Client

於 NAT 外端的對應 IPv4 位址值。Authentication 標頭用來進行安全認證，其標頭欄位說明請參考 Teredo 標準文件。

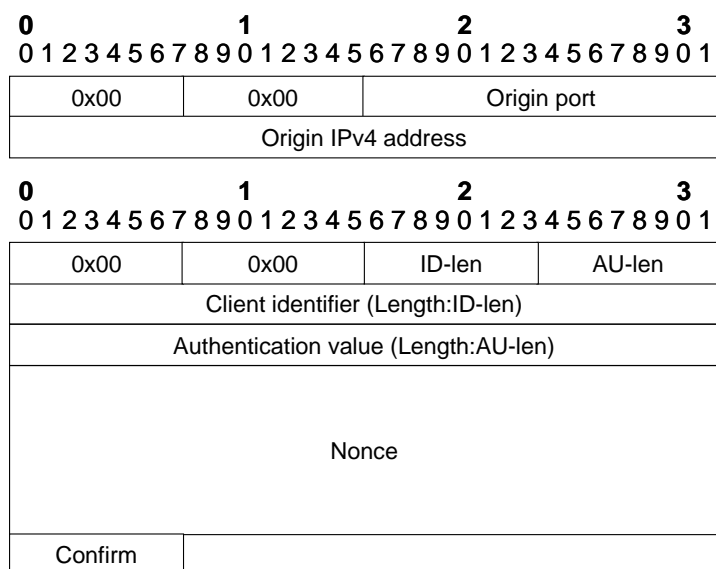


圖 B-5 Origin Indication 與 Authentication 標頭格式圖



附錄 C SIGTRAN 標準

SIGTRAN 標準為 IETF SIGTRAN 工作小組所制定的技術標準，該標準規範如何透過 IP 網路傳送以 PSTN 為基礎的信令（PSTN-based Signaling）。SIGTRAN 標準包括 SCTP、IUA Layer、M2UA Layer 與 M3UA Layer，如圖 C-1 的粗體框線所示。

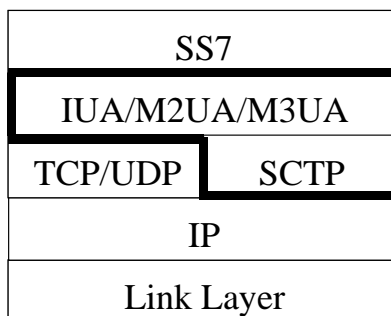


圖 C-1 PSTN-based Signaling Over IP 的通訊協定堆疊圖

C.1 SCTP

SIGTRAN 工作小組建議傳送以 PSTN 為基礎的信令時採用 SCTP 傳輸層協定，亦可採用 TCP 傳輸層協定。SCTP 為連接導向（Connection-Oriented）與訊息導向（Message-Oriented）的通訊協定。SCTP 的特點在於多串流（Multi-Stream）的支援、Multi-Homing 的支援、非次序的可靠訊息傳送（Unordered Reliable Message Delivery）功能與安全性的加強 [2]。

SCTP 的封包格式包括 SCTP Common 標頭與許多 Chunk 如圖 C-2 所示。SCTP Common 標頭中的 Source port number 欄位指名來源埠號、Destination port number 欄位元指名目的埠號、Verification tag 欄位用來作驗證 SCTP 封包發送端的合法性與 Checksum 欄位為封包檢查碼。SCTP 標準定義的十二種 Chunk 與其功能如表 C-1 所列。

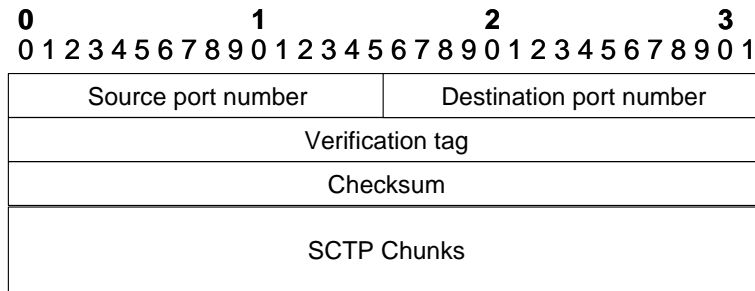


圖 C-2 SCTP 封包格式圖

表 C-1 SCTP Chunk 功能說明表

Chunk 型態	Chunk 功能
Payload Data (DATA)	使用者資料
Initiation (INIT)	初始化 SCTP 連線
Initiation Acknowledgement (INIT ACK)	回應收到 INIT Chunk
Selective Acknowledgement (SACK)	回應收到使用者資料
Heartbeat Request (HEARTBEAT)	用來確保 SCTP 連線持續
Heartbeat Acknowledgement (HEARTBEAT ACK)	回應收到 HEARTBEAT Chunk
Abort (ABORT)	跳離 SCTP 連線
Shutdown (SHUTDOWN)	停止 SCTP 連線
Shutdown Acknowledgement (SHUTDOWN ACK)	回應收到 SHUTDOWN Chunk
Operation Error (ERROR)	回應錯誤訊息
State Cookie (COOKIE ECHO)	初始化 SCTP 連線，詳細定義請參考 RFC2960
Cookie Acknowledgement (COOKIE ACK)	回應收到 COOKIE chunk

C.2 IUA/M2UA/M3UA

IUA Layer 提供承載 Q.921 上層的 Signaling System Number 7 (SS7) 協定，如 Q.931。M2UA Layer 上層提供承載 Message Transfer Part2 (MTP2) 上層的 SS7 協定，如 Message Transfer Part3 (MTP3)。M3UA Layer 上層提供承載 MTP3 上層的 SS7 協定，如 Signaling Connection Control Part (SCCP)、ISDN User Part (ISUP)。

附錄 D 封包產生精靈

本系統為符合 2003 年第三屆國家高速電腦中心軟體程式設計競賽的程式需求，而實作封包產生精靈，如圖 D-1 所示。封包產生精靈提供三種方式協助使用者產生測試封包，分別為送出已擷取的封包、Step-by-step 產生封包與使用樣板產生封包。送出已擷取的封包的目的是重現網路上的流量狀況。當網路設備因網路上的封包流量發生故障時，網管人員可以先擷取封包以便分析設備故障的原因。當網管人員完成網路設備的故障排除後可以利用本功能來重現故障時的網路流量狀況，並驗證故障排除是否有效。Step-by-step 產生封包將逐步帶領使用者建立並送出測試封包。使用樣板產生封包讓使用者可以快速的建立並送出測試封包。



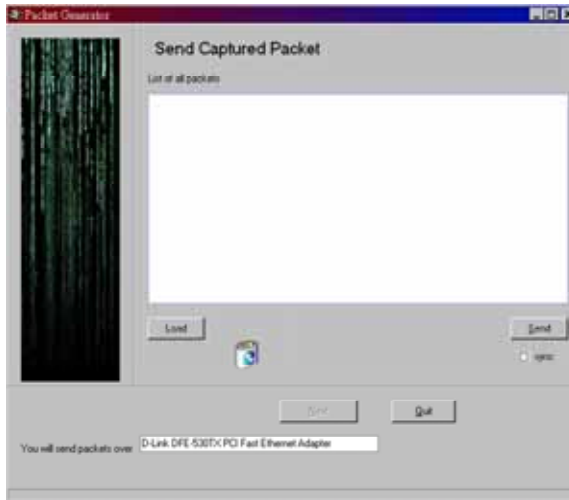
圖 D-1 封包產生器精靈畫面圖

D.1 送出已擷取的封包

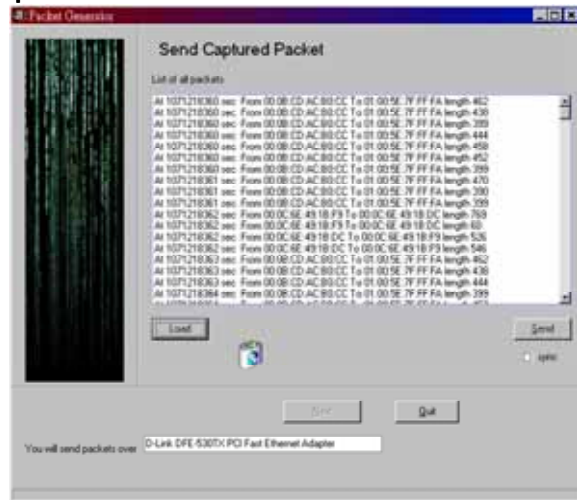
送出已擷取的封包流程圖 D-2 所示。使用者在啟動封包產生器精靈並選取要送出測試封包的網路介面後，在下方導覽列選擇以送出已擷取封包（Send captured packets）的方式產生測試封包，如圖 D-2 步驟①所示。步驟①中點選下一步按鈕後進入步驟②。使用者可以在步驟②中點選載入（Load）按鈕來載入欲送出的測試封包擷取檔，此時進入步驟③。使用者可以於步驟③中確認測試封包，並點選送出（Send）按鈕來送出測試封包。



Step ①



Step ②



Step ③

圖 D-2 送出已擷取封包的步驟流程圖

D.2 Step-by-step 產生封包

Step-by-step 產生封包中，使用者將透過封包產生器精靈的協助，由上層通訊協定往下層通訊協定的建置方式來建立測試封包，最後送出測試封包。比方產生 TCP 測試封包的步驟流程，首先使用者必須先建構 TCP 承載，然後建構 TCP 標頭，再建構 IP 標頭，最後建構乙太網路標頭並將此測試封包送出。

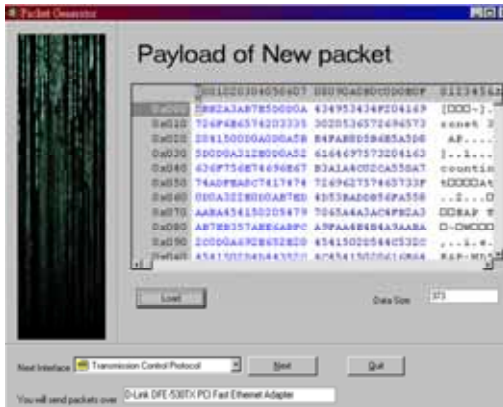
圖 D-3 展示如何建立並送出 TCP/IPv6 的測試封包。使用者啟動封包產生器精靈並選取送出測試封包的網路介面後，在下方導覽列選擇建立新封包（Create new packet）。步驟①中點選下一步按鈕進入步驟②。步驟②中選取新封包承載（New packet payload），並點選下一步按鈕進入步驟③。步驟③中，首先點選載入（Load）按鈕載入檔案資料做為 TCP 承載，然後選取下層通訊協定為 TCP，並點選下一步按鈕進入步驟④，此時完成 TCP 承載建立。步驟④中，先填入相關的 TCP 標頭資訊，然後選取下層通訊協定為 IPv6，並點選下一步按鈕進入步驟⑤，此時完成 TCP 標頭建立。步驟⑤中，先填入相關的 IPv6 標頭資訊，然後選取下層通訊協定為乙太網路，點選下一步按鈕進入步驟⑥，此時完成 IPv6 標頭建立。步驟⑥中，先填入相關的乙太網路標頭資訊，然後選取送出（Send Out）並點選下一步按鈕進入步驟⑦，此時完成乙太網路標頭建立。步驟⑦中，封包產生器精靈將顯示建立完成的測試封包資訊。使用者確定無誤後，在步驟⑦中填入稍後欲送出的測試封包個數與測試封包送出間隔，最後點選送出按鈕就可以將測試封包送出。



Step ①



Step ②



Step ③



Step ④



Step ⑤



Step ⑥



Step ⑦

圖 D-3 TCP/IPv6 測試封包的建構步驟流程圖

D.3 使用樣板產生封包

使用樣板產生封包協助使用者快速建立，並送出特定通訊協定的測試封包。首先使用者選擇特定封包樣板，然後填入封包樣版中的必要封包資訊來建立測試封包，最後送出測試封包。目前提供的封包樣板有 ARP 協定與 TCP 協定。

圖 D-4 展示如何透過樣板產生並送出 ARP 測試封包。使用者在啟動封包產生器精靈並選取要送出測試封包的網路介面後，在下方導覽列選擇建立新封包（Create new packet），如圖 D-4 步驟①所示。步驟①中點選下一步按鈕進入步驟②。步驟②中選取封包樣板（Packet template）並點選下一步按鈕進入步驟③。步驟③中，選取 ARP 封包樣板並填入必要的 ARP 封包資訊，最後點選送出按鈕將測試封包送出。

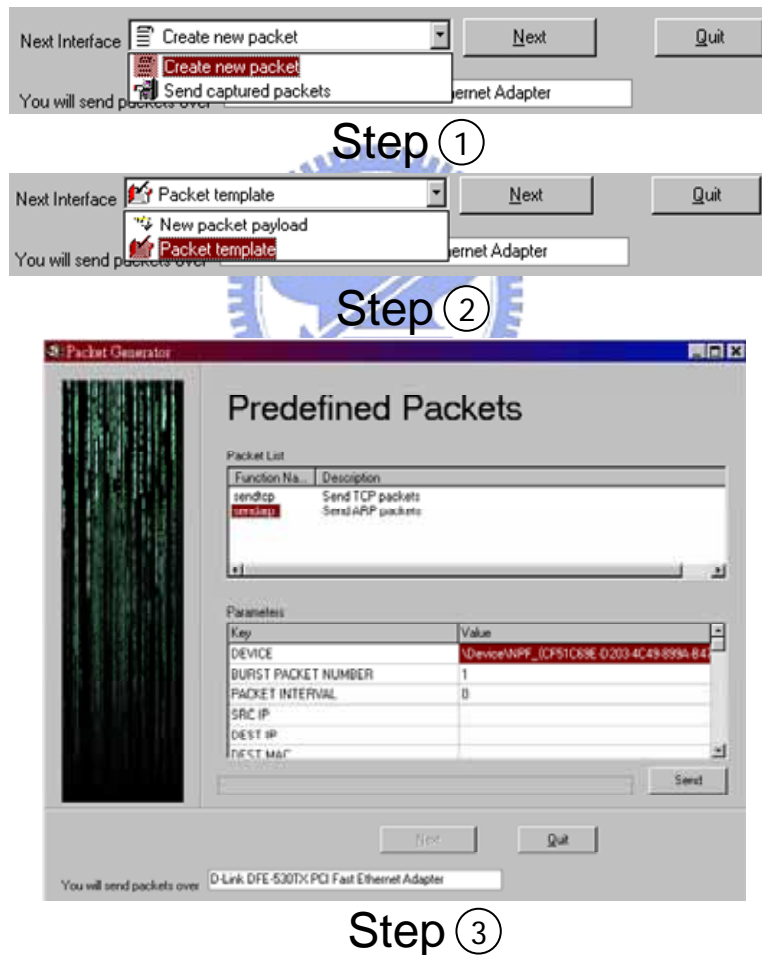


圖 D-4 ARP 測試封包的建構步驟流程圖

附錄參考文獻

- [1] C.Huitema. Teredo: Tunneling IPv6 over UDP through NATs. IETF Internet Draft, Feb.2004.
- [2] R. Stewart, Q. Xie, K. Morneault, et al. Stream Control Transmission Protocol. IETF RFC2960, Oct. 2000.
- [3] Silvia Hagen. IPv6 Essentials. Oreilly. 2002.

