網路協定觀察與分析

Ⅰ. 實驗目的

本實驗的目的是希望同學能瞭解網路協定在網路作業環境下所扮演的角色，同時，藉由實際觀察封包的組成以及特定協定的運作，瞭解 TCP/IP 通訊協定階層的意義，並熟悉網路運作的機制。

實驗報告應該包括下列項目：實驗名稱、組員與系級、實驗目的、設備與操作環境、所觀察協定之背景知識、方法與步驟、觀察與記錄、討論（針對問題與討論的項目回答，或自行提出問題並討論之）及參考書目。報告之文字篇幅限定爲 8~10 頁(A4)，一律繳交雷射或噴墨列印之完稿。

Ⅱ. 實驗設備

一、硬體

項目	數量	備註
個人電腦 PC	1	
網路卡	1	視網路環境選用介面卡。

二、軟體

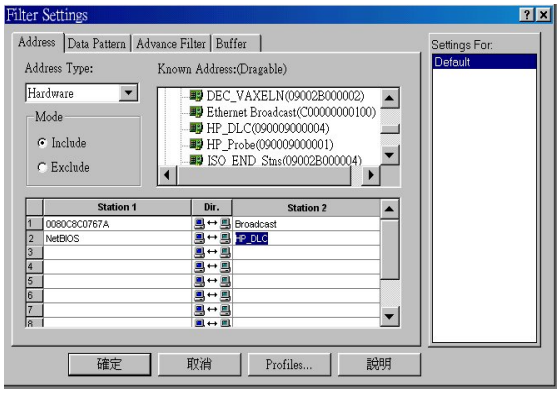
項目	數量	備註
Ethereal [17] 或 NetXRay [18]	1	安裝在各實驗電腦上。
各式網路應用軟體	不定	視所要觀察的網路協定來選用軟體。

Ⅲ. 背景資料

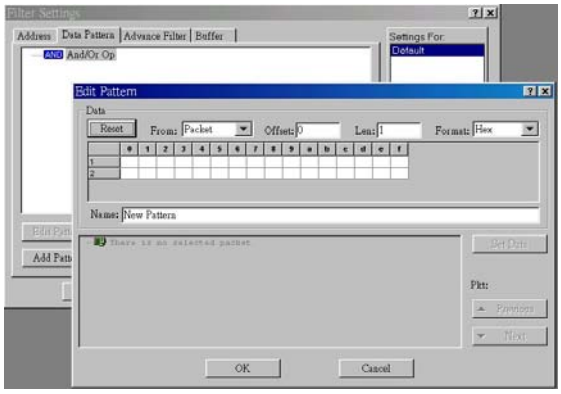
一、NetXRay 功能簡介

NetXRay 提供的功能與協定分析儀相當，只是 NetXRay 處理速度較慢，但是對於分析特定協定而言，並不需要處理大量的封包，所以這套軟體所

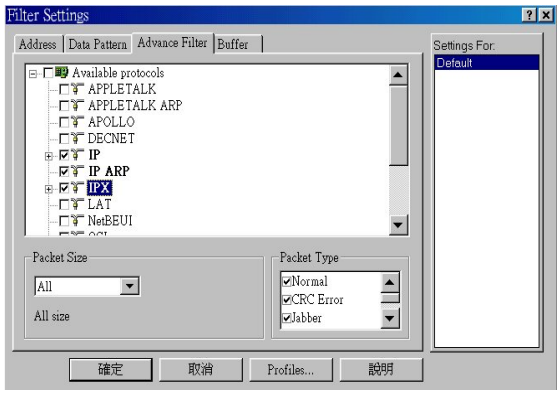
提供的功能足以分析一般的通訊協定。就封包擷取設定來講，可用三種方法來設定，分別是根據封包位址（見圖 4-1）、封包資料樣本（見圖 4-2）及封包採用的協定（見圖 4-3）。封包擷取的結果可有下列五種顯示方法：單一封包資料圖（見圖 4-4）、封包流向圖（見圖 4-5）、協定分佈圖（見圖 4-6）、封包大小分佈圖（見圖 4-7）及主機流量統計表（見圖 4-8）。各位同學可由這些圖表中分析出通訊協定在網路中的運作情形及分佈狀況。



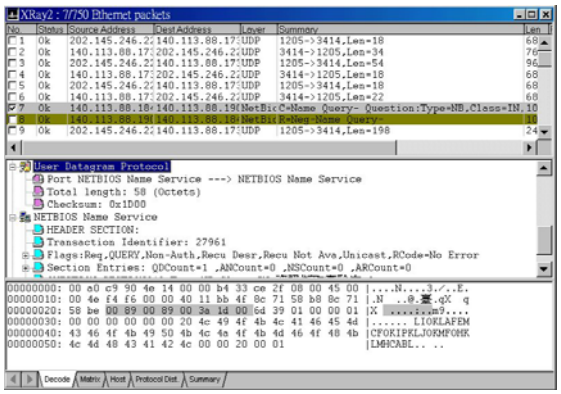
【圖 4-1】指定封包位址



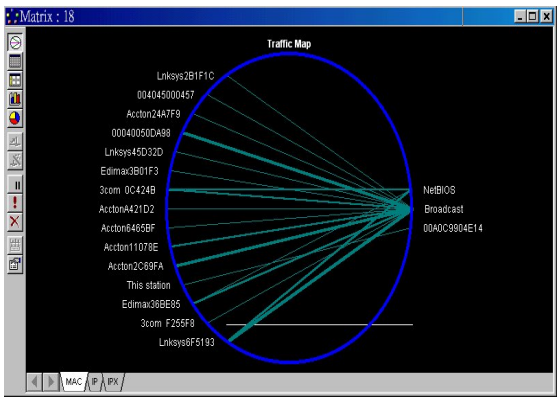
【圖 4-2】指定封包資料樣本



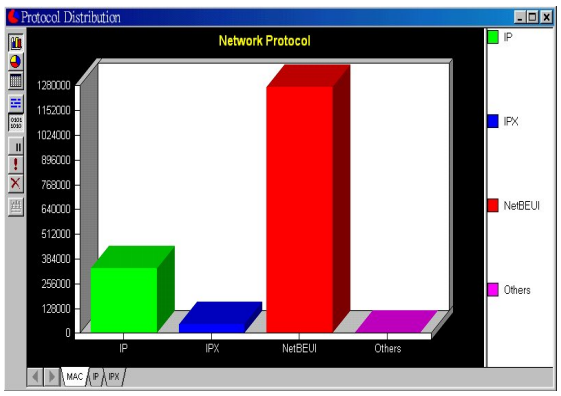
【圖 4-3】指定封包協定



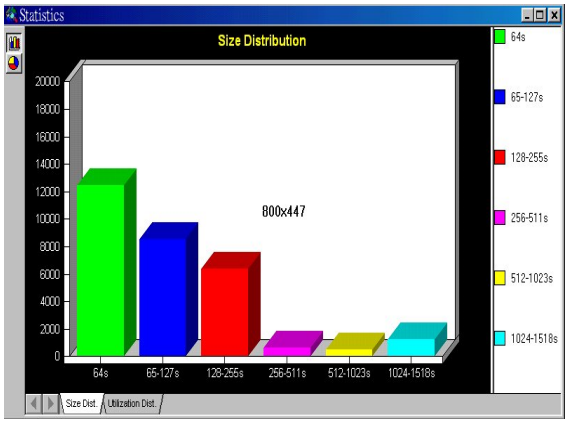
【圖 4-4】單一封包資料圖



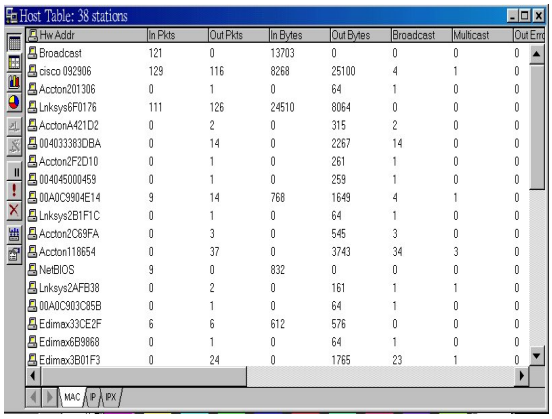
【圖 4-5】封包流向圖



【圖 4-6】協定分佈圖



【圖 4-7】封包大小分佈圖



【圖 4-8】主機流量統計表

二、Ethereal 功能簡介

儘管 NetXRay 功能強大，但價錢也不低，大約需要台幣五萬元左右。另外，NetXRay 只能在 Windows 95/98/NT[1]系統下執行，對近年來推出的 Windows XP/2000 和其他作業系統並沒有支援。

Ethereal 是一套可以在多種作業系統下執行的網路協定分析軟體，它可在 Linux, FreeBSD, Windows 等系統下執行[2]。由於它是免費的，相較於 NetXRay 可以省下一筆可觀的費用。雖然功能不如 NetXRay 來的強大，但是仍可用來觀察大部分協定的封包。

Sniffer 和 NetXRay 是近年來網管人員最熟悉的封包監聽軟體，但 Ethereal 這套免費的軟體，由於採取開放原始碼的方式，更新通訊協定 Protocol 迅速，支援不同軟體匯出的封包擷取檔案格式，目前廣爲世界各地專業網管使用。可以很容易的選取擷取封包時間，並透過圖形介面來表示，清晰易懂。此外，使用過濾的功能，可以輕易地判別出封包種類和分析網路中各式各樣流竄的封包內容。支援 620 種不同的 Protocol，且仍在持續增加之中。相容的封包擷取檔案格式包含：tcpdump、, NAI 的 Sniffer，NetXray，Sun snoop，AIX 的 iptrace，Microsoft 的 Network Monitor，Novell 的 LANalyzer，Cisco 的 IDS iplog 等，幾乎所有知名的封包擷取軟體，通通都可以在這套軟體中讀取檢視。

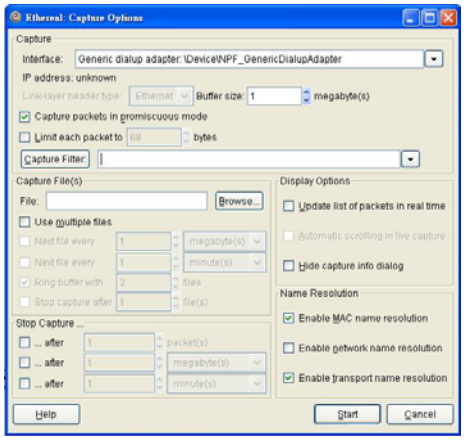
讀者可由底下的圖表中了解 Ethereal 大致的使用情形。

---

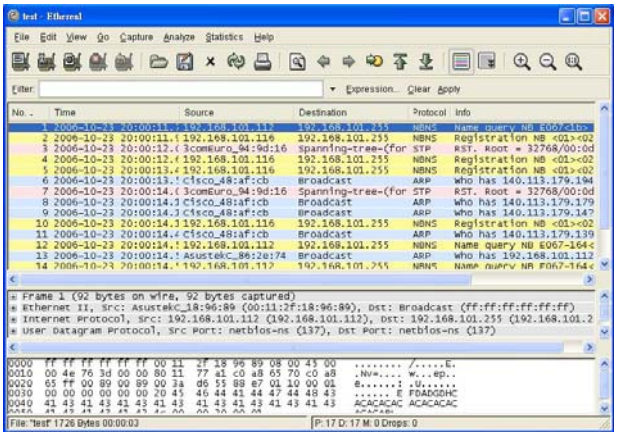
[1] NetXRay 截至 3.0 版爲止仍無法在 Windows 2000/XP 下執行。

[2] Ethereal 在不同系統的安裝方式並不相同，詳細安裝步驟請參閱下載網站[17]的說明。

3 Ethereal 改名爲 Wireshark，亦爲一套自由免費開放原始碼的軟體，有興趣可參閱其官網[19]。



【圖 4-9】封包過濾的設定



【圖 4-10】單一封包資料圖

三、網路協定列表

在本實驗中，HTTP 必須列入實驗觀察對象。另外，實驗者必須從下列協定中選擇另一個協定作爲觀察與分析的對象，所有 RFC 可由[1]或 NCTUCCCA 取得。由於 ARP 協定的分析流程已詳述在實驗報告範例，所以這個協定〝不可〞列入實驗報告觀察對象。可選擇的協定包括：SNMP[2,3]、ARP[4]、RARP[5]、DNS[6,7]、SMTP[8]、RPC[9] 、RIP[10] 、HTTP[11]、DVMRP[12] 、POP3[13] 、NFS[14]以及 NetBIOS[15,16]等。這些通訊協定簡介如表 4-1。

【表 4-1】通訊協定簡表

協定	OSI layer	功能
SNMP	Application	網路設備與資料流量的監督與管理。
ARP	Network	由 IP 位址查詢 MAC 位址。
RARP	Network	由 MAC 位址查詢 IP 位址。
DNS	Application	由 domain name 查詢 IP 位址。
SMTP	Application	寄送電子郵件至指定的電子郵件帳號。
POP3	Application	接收並保存電子郵件。
RPC	Session	呼叫並執行遠端主機上的程序。
RIP	Network	Unicast routing protocol。
DVMRP	Network	Multicast routing protocol。
NFS	Application	分散式檔案管理與存取系統。
NetBIOS	Presentation	在一群指定的主機間提供溝通機制，共享資源。
HTTP	Application	超媒體文件傳送、接收與管理。
RTP/RTSP	Application	支援在單和多目標廣播網路服務中傳輸即時資料。
SIP	Application	提供整合語音與其它多媒體的通訊服務。

Ⅳ. 實驗方法

分析網路協定方法如下：首先要充實對所選擇要觀察之協定的背景知識，瞭解制訂該協定的目的，要解決的問題。例如由於超媒體文件(HyperText)的資訊渲染力強，於是制訂 HTTP 協定來傳輸 Web 超媒體文件。另外，爲了增加頻寬的使用效率，也定義了相關快取伺服器的運作機制。第一步要瞭解協定溝通時所使用的語言，也就是標頭(Header)中各欄位所代表的意義，以及協定運作的流程（或者說封包傳遞的順序）。一旦這些背景知識都具備之後，就可以設定擷取過濾器及顯示模式來觀察該協定並記錄結果，對於擷取到的協定封包要能給予合理的解釋，並詳細描述整個協定運作的流程。底下以 NetXRay 爲例，觀察 HTTP 協定的流程，同學可作爲參考。HTTP 的運作原理請參考[11]。

Ⅴ. 實驗步驟

1. **Ethereal 啓動與設定**

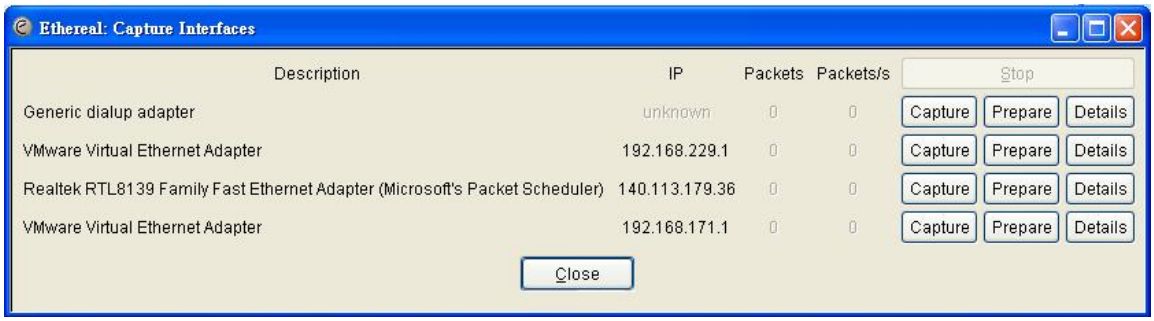
   1.1 由「程式集」→「Ethereal」啓動 Ethereal。

   1.2 由「Capture」→「Options」設定擷取模式。

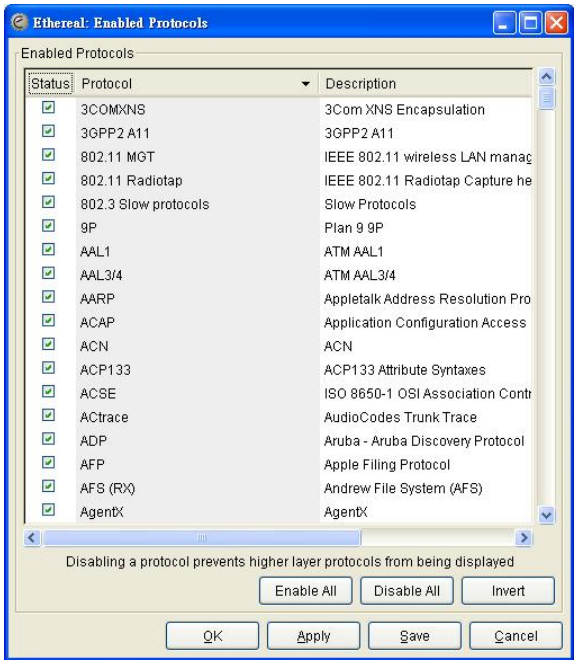
      1.2.1 首先選擇 Interfaces 設定頁，設定欲擷取封包之來源及目的位址(如圖 4-11)。

      1.2.2 再來選擇 Capture Filters 設定頁，根據你對所觀察的通訊協定的了解設定欲觀察的協定(如圖 4-12)。

   1.3 由「Capture」→「Start」開始擷取封包。



【圖 4-11】設定封包位址



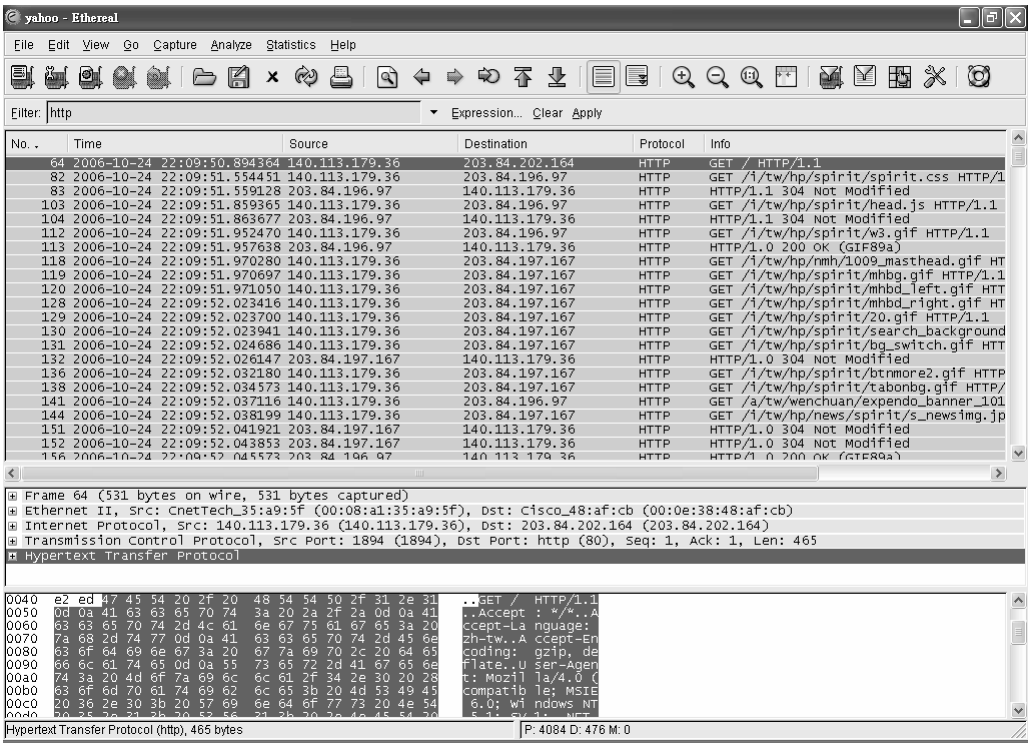
【圖 4-12】設定封包協定

2. 範例：HTTP 協定觀察

假設主機已經與交通大學首頁建立好 TCP 連線（如圖 4-13），如果要觀察 HTTP 協定如何傳輸超媒體文件，可任選一超連結(Hyper-Link)來發起 HTTP 要求，在本範例中是選取 Yahoo 首頁（見圖 4-14）。圖 4-15 爲存取 Yahoo 首頁時，HTTP 協定運作所產生的各個封包，封包的意義如表 4-2 所示。

【圖 4-13】交通大學首頁

【圖 4-14】Yahoo 首頁



【圖 4-15】存取 Yahoo 首頁所產生的 HTTP 協定封包

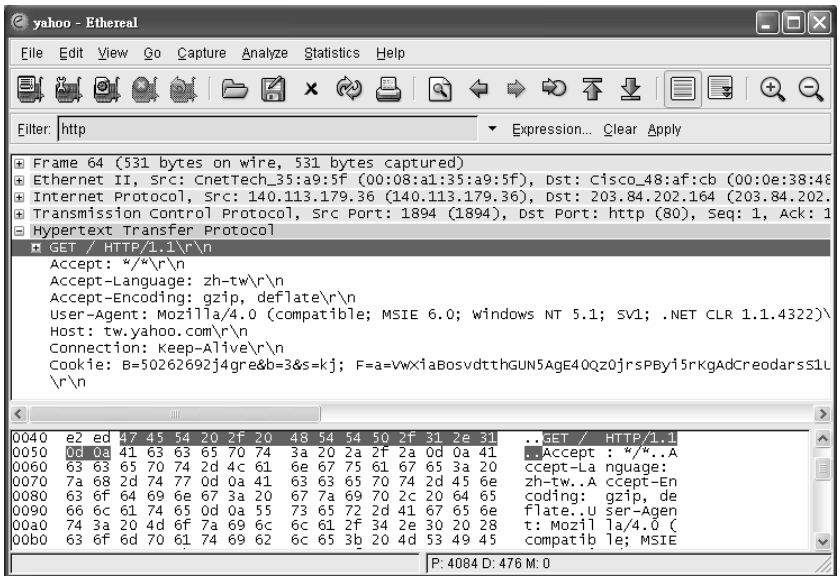
【表 4-2】HTTP 協定封包的意義

順序	封包來源	協定封包意義
64	本機	GET /news/ HTTP/1.1 要求傳送 /news 目錄下的首頁。
		由 Accept、Accept-Language 以及 Accept-Encoding 指明主機端可接受的文件樣式。 由 User-Agent 指明本機端採用 MSIE 作爲 HTTP 協定處理器。 由 Connection 指明 TCP 層繼續保持連線。
82	網頁端	HTTP/1.1 200 OK 傳回 /news 目錄下的首頁。
		由 Server 指明網頁端採用 Apache/1.2.4 作爲 HTTP 協定處理器。 由 Last-Modified 指明最後修改時間爲 02 Oct 2006 13:43:55。 由 Content-type 指明內容爲 text/html 的文件樣式。 Data 部分包含/news 目錄下的首頁。
83	本機	GET /news/head.html HTTP/1.1 要求傳送 /news 目錄首頁中的 head.html。
		由 Accept、Accept-Language 以及 Accept-Encoding 指明主機端可接受的文件樣式。 由 User-Agent 指明本機端採用 MSIE 作爲 HTTP 協定處理器。 由 Connection 指明 TCP 層繼續保持連線。

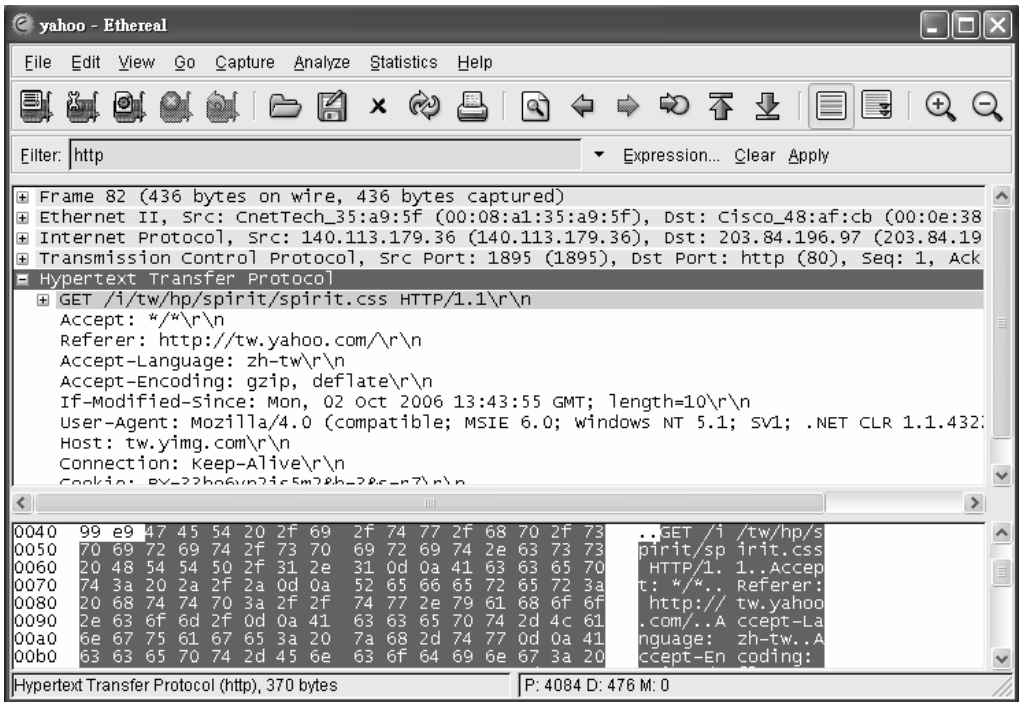
103	網頁端	HTTP/1.1 200 OK 傳回 head.html。
		由 Server 指明網頁端採用 Apache/1.2.4 作爲 HTTP 協定處理器。 由 Last-Modified 指明最後修改時間 19 Oct 2006 06:37:22。 由 Content-type 指明內容爲 text/html 的文件樣式。 Data 部分包含 head.html。
104	本機	GET /news/read-news/listall.pl.cgi HTTP/1.1 要求執行 listall.pl.cgi。
		由 Accept、Accept-Language 以及 Accept-Encoding 指明主機端可接受的文件樣式。 由 User-Agent 指明本機端採用 MSIE 作爲 HTTP 協定處理器。 由 Connection 指明 TCP 層繼續保持連線。
112	本機	GET /cgi-bin/Count.cgi?dd=E&df=news.dat HTTP/1.1 要求執行計數器。
		由 Accept、Accept-Language 以及 Accept-Encoding 指明主機端可接受的文件樣式。 由 User-Agent 指明本機端採用 MSIE 作爲 HTTP 協定處理器。 由 Connection 指明 TCP 層繼續保持連線。
113	網頁端	HTTP/1.1 200 OK 回覆計數器結果及圖樣。
		由 Server 指明網頁端採用 Apache/1.2.4 作爲 HTTP 協定處理器。 由 Transfer-Encoding 指明圖樣及計數結果分爲 chunks 傳送。 由 Content-type 指明內容爲 image/gif 的文件樣式。 Data 部分包含圖樣及計數結果。
118	網頁端	HTTP/1.1 200 OK 回覆 listall.pl.cgi 執行結果 listall_pl.html。
		由 Server 指明網頁端採用 Apache/1.2.4 作爲 HTTP 協定處理器。 由 Transfer-Encoding 指明執行結果分爲 chunks 傳送。 由 Content-type 指明內容爲 text/html 的文件樣式。 Data 部分包含 listall_pl.thml 的部分結果。
119	本機	GET /Image/bga.gif 要求傳送 listall_pl.html 中的 bga.gif。
		由 Accept、Accept-Language 以及 Accept-Encoding 指明主機端可接受的文件樣式。 由 User-Agent 指明本機端採用 MSIE 作爲 HTTP 協定處理器。 由 Connection 指明 TCP 層繼續保持連線。
120	網頁端	HTTP/1.1 200 OK 傳回 bga.gif。

		由 Server 指明網頁端採用 Apache/1.2.4 作爲 HTTP 協定處理器。 由 Last-Modified 指明最後修改時間 22 May 2006 13:17:54。 由 Content-type 指明內容爲 image/gif 的文件樣式。 Data 部分包含 bga.gif。
128	網頁端	HTTP/1.1 200 OK 回覆 listall.pl.cgi 執行結果 listall_pl.html (重送封包 15)。
		由 Server 指明網頁端採用 Apache/1.2.4 作爲 HTTP 協定處理器。 由 Transfer-Encoding 指明執行結果分爲 chunks 傳送。 由 Content-type 指明內容爲 text/html 的文件樣式。 Data 部分包含 listall_pl.thml 的部分結果。
129	網頁端	More data 回覆 listall.pl.cgi 執行結果 listall_pl.html。
		Data 部分包含 listall_pl.thml 的部分結果。
130	網頁端	More data 回覆 listall.pl.cgi 執行結果 listall_pl.html。
		Data 部分包含 listall_pl.html 的部分結果。

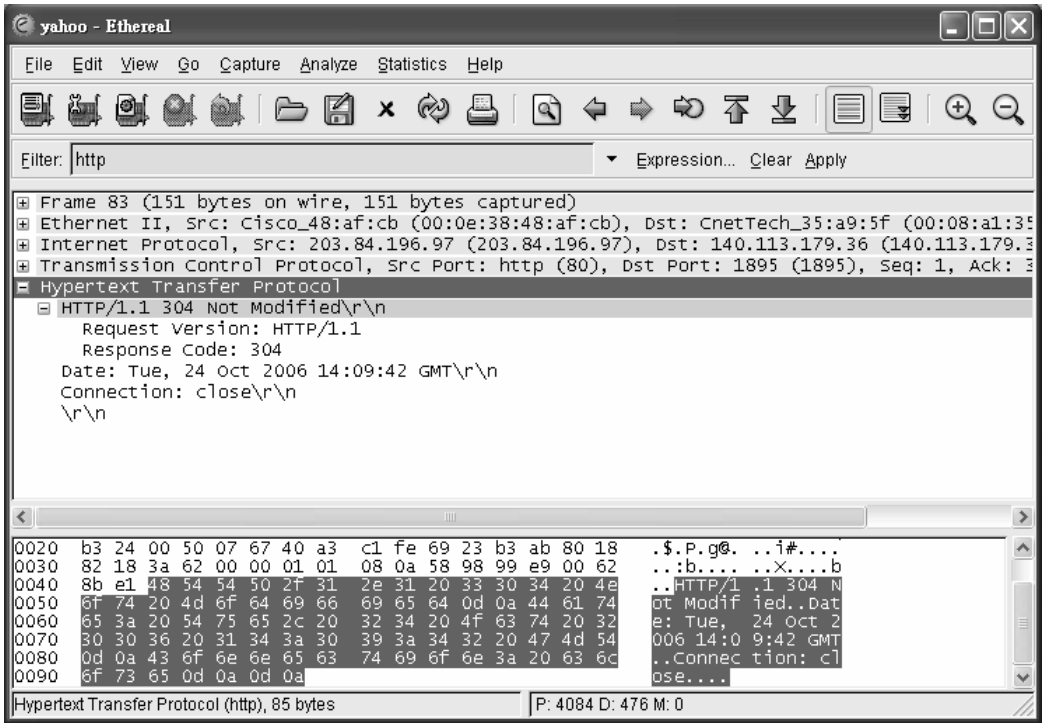
上述封包的內容詳見圖 4-16 ~ 圖 4-28。除了上述列舉出來的封包之外，其他封包主要是用在 TCP 層溝通，包括建立連線與識別，在此並不列舉。



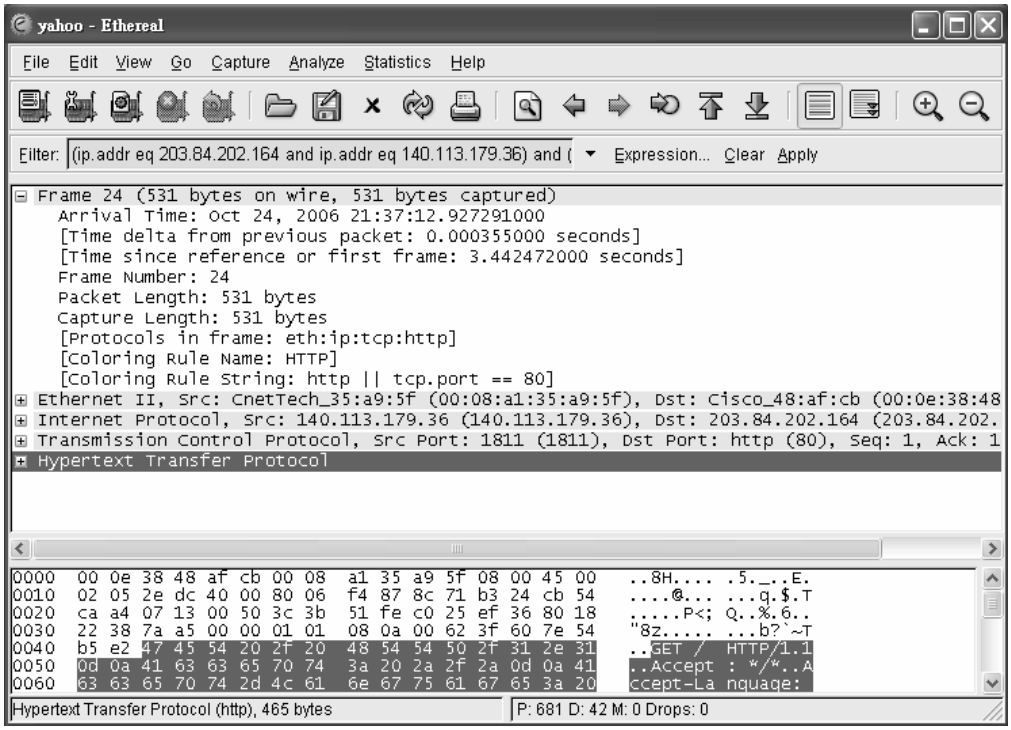
【圖 4-16】第一個 HTTP 封包內容



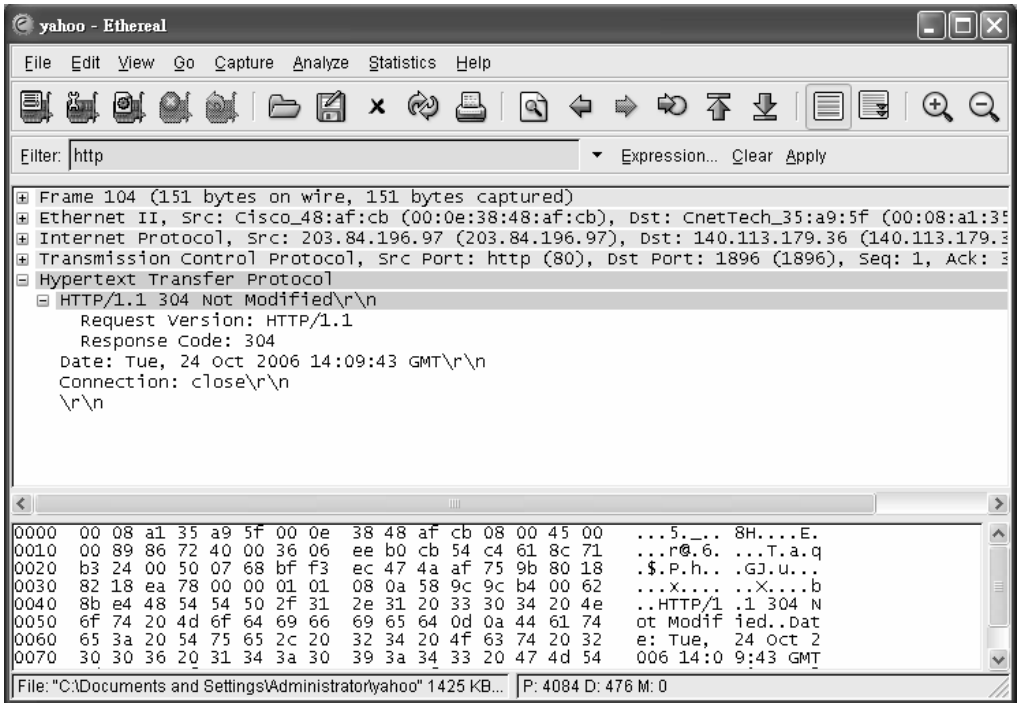
【圖 4-17】第二個 HTTP 封包內容



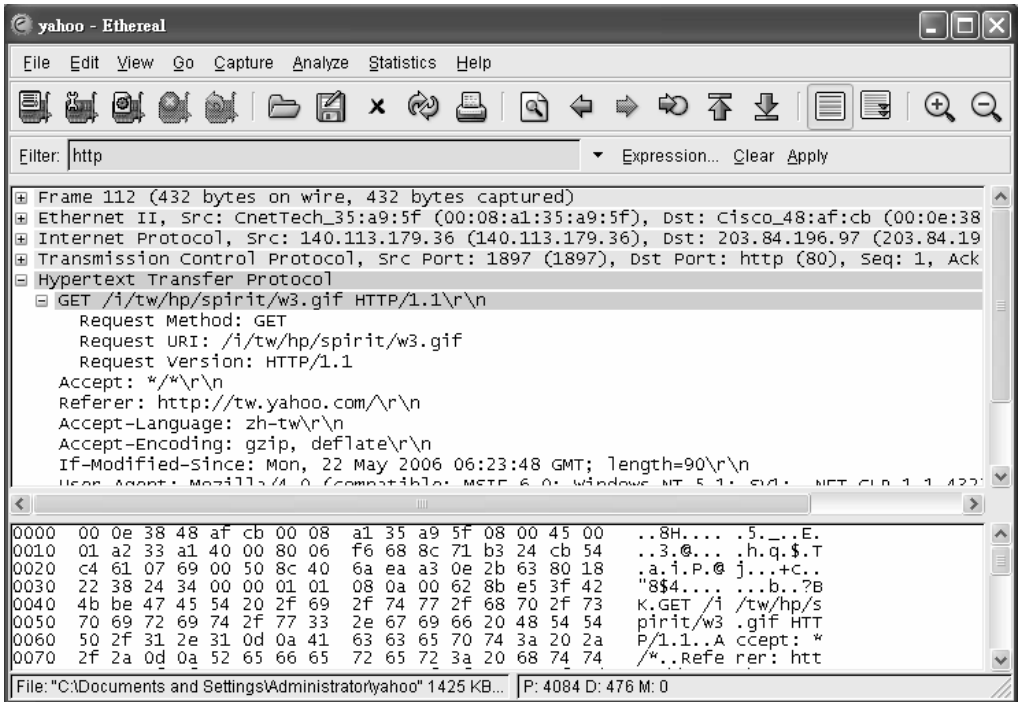
【圖 4-18】第三個 HTTP 封包內容



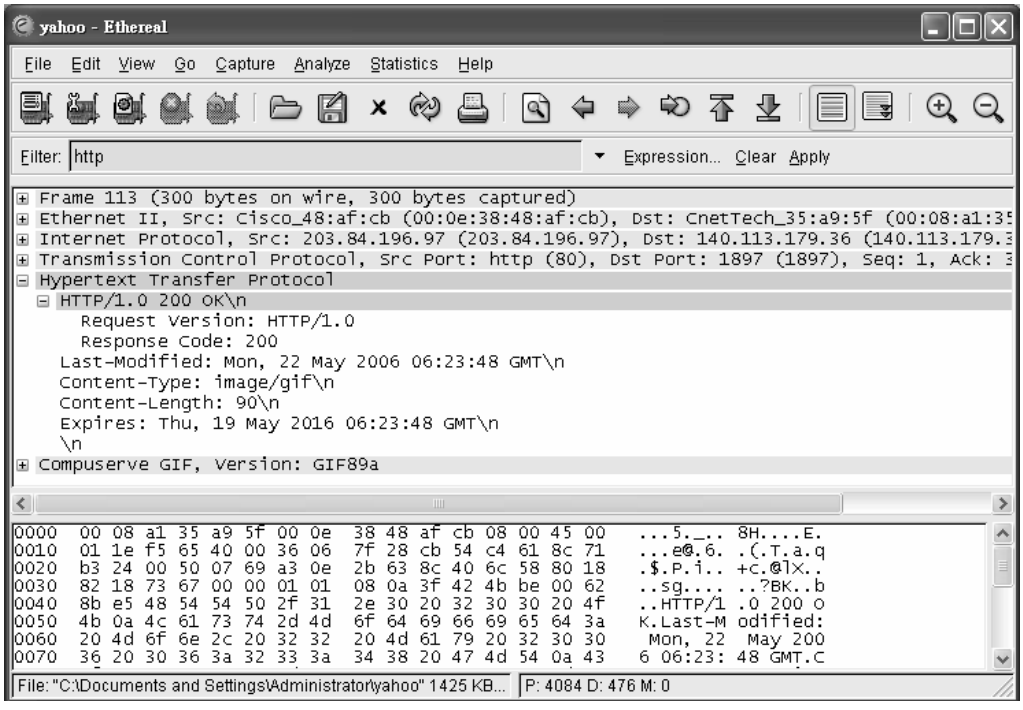
【圖 4-19】第四個 HTTP 封包內容



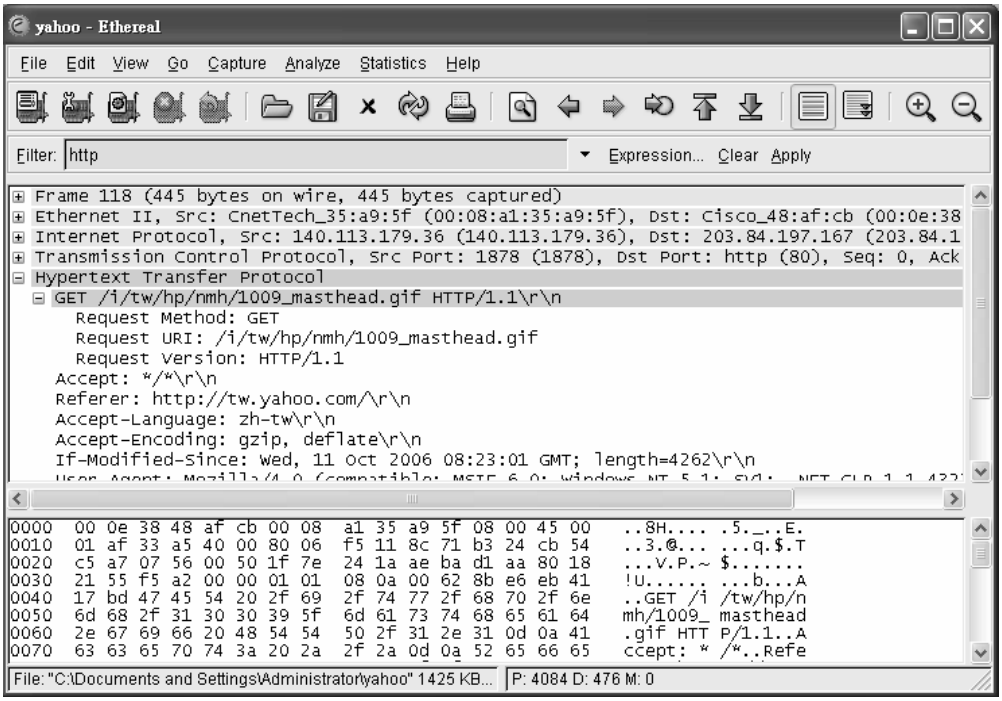
【圖 4-20】第五個 HTTP 封包內容



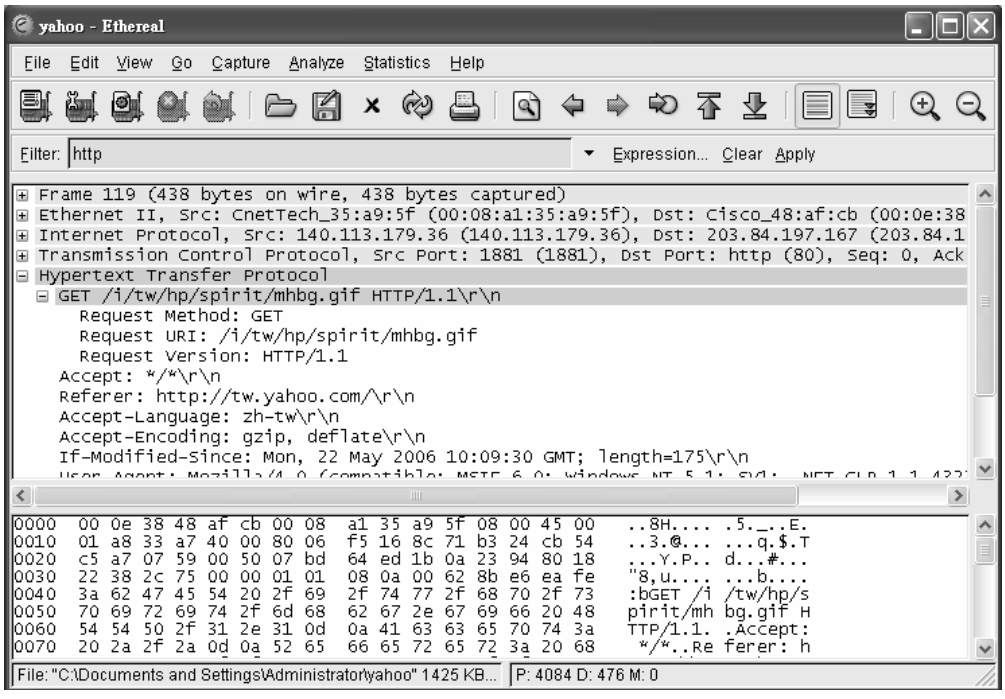
【圖 4-21】第六個 HTTP 封包內容



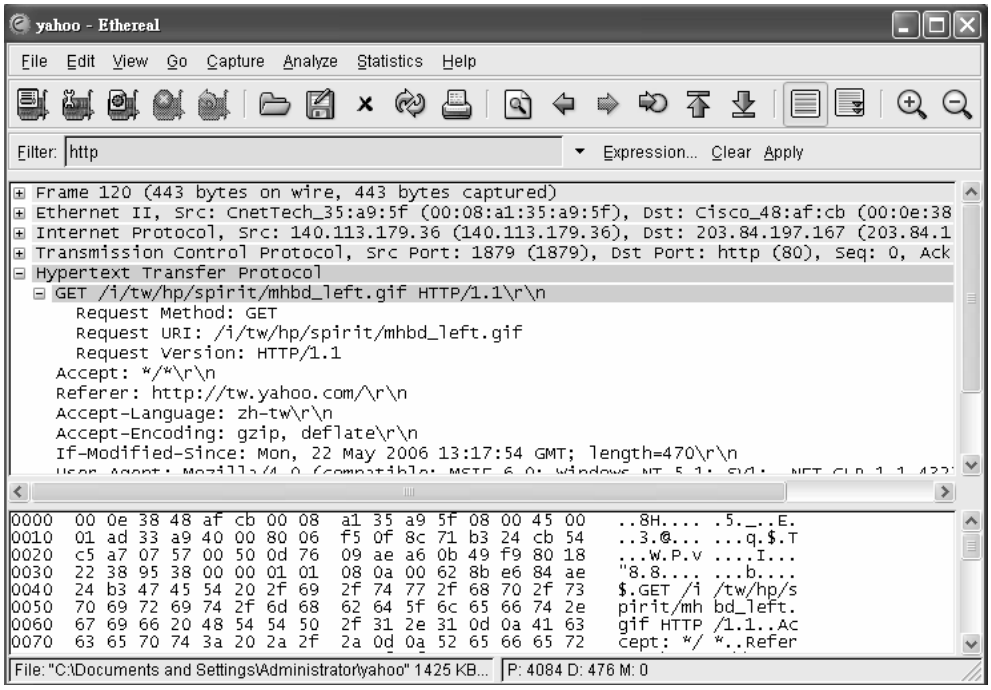
【圖 4-22】第七個 HTTP 封包內容



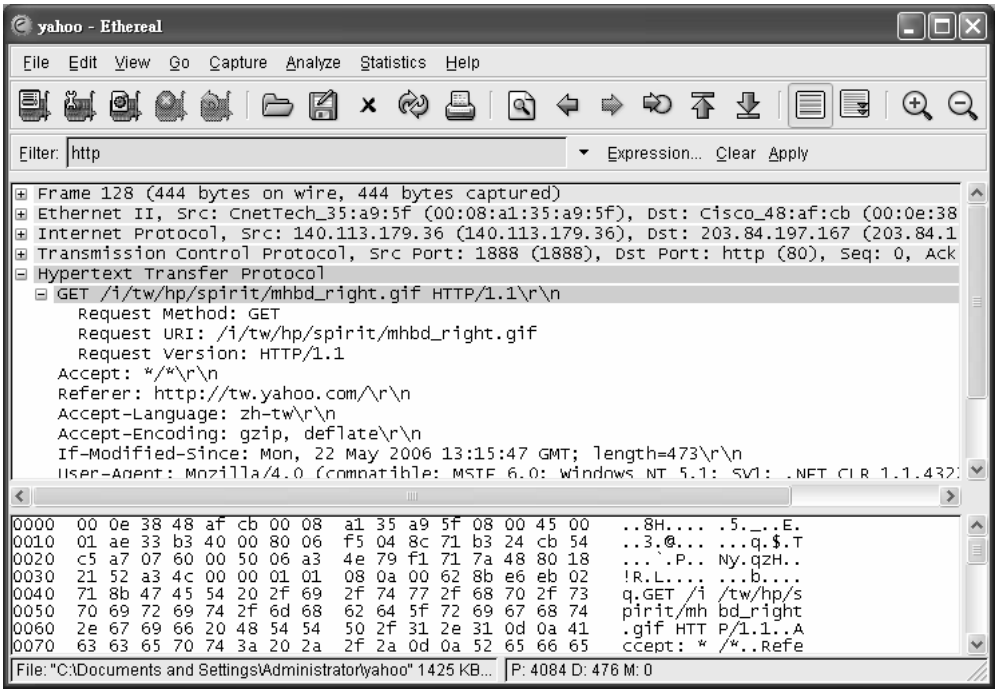
【圖 4-23】 第八個 HTTP 封包內容



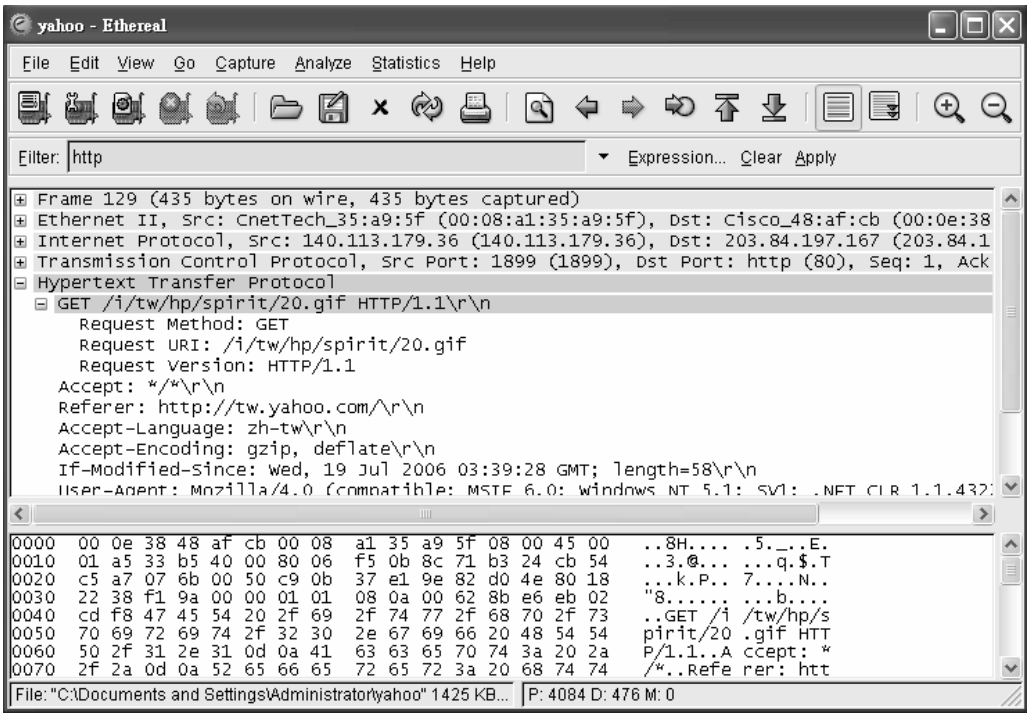
【圖 4-24】第九個 HTTP 封包內容



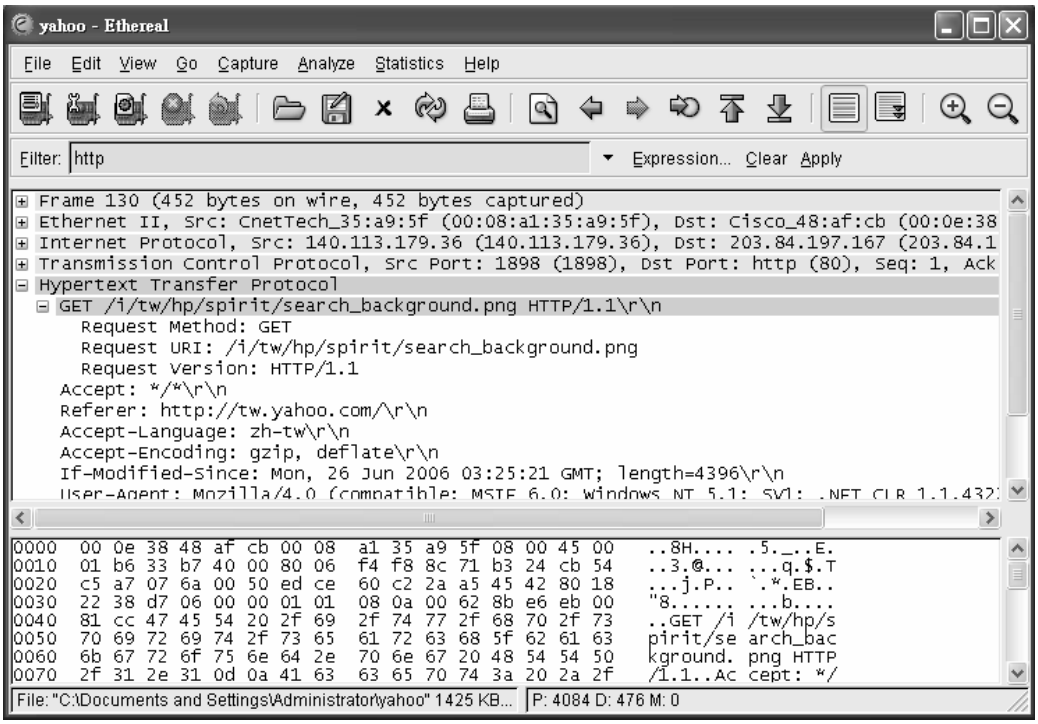
【圖 4-25】第十個 HTTP 封包內容



【圖 4-26】第十一個 HTTP 封包內容



【圖 4-27】第十二個 HTTP 封包內容



【圖 4-28】第十三個 HTTP 封包內容

VI. 問題與討論

1. 請各位同學以點對點(peer-to-peer)或主從架構(client-server)的觀點來討論所觀察的協定，包括用途、協定封包格式及運作流程。

2. 請討論所觀察的協定是使用哪種下層協定？及被哪些上層協定所使用？並討論之間的關係。

3. 由 PDU(Protocol Data Unit)（即封包）間 inter-arrival time 的記錄，討論所觀察協定之延遲狀況，及延遲主要發生在何時何處。

4. 估計所觀察協定產生之封包資料量大小，討論對網路負擔之大小。（以數值討論）

5. 請同學自行設計協定運作的錯誤狀況，由所觀察的 PDU 中描述協定本身對錯誤狀況的處理。

6. 自行發掘問題並尋找解答。

VII. 參考文獻

[1] IETF Homepage, "http://www.ietf.org".

[2] J.D. Case, M. Fedor, M.L. Schoffstall, and C. Davin, "*Simple Network Management Protocol (SNMP),"* RFC1157, May-01-1990.

[3] M. Rose, "*SNMP over OSI,"* RFC1418, March 1993.

[4] D.C. Plummer, "*Ethernet Address Resolution Protocol: Or converting network protocol addresses to 48.bit Ethernet address for transmission on Ethernet hardware,"* RFC826, Nov-01-1982.

[5] R. Finlayson, T. Mann, J.C. Mogul, M. Theimer, "*Reverse Address Resolution Protocol," RFC903,* Jun-01-1984.

[6] P.V. Mockapetris, "*Domain names - concepts and facilities,"* RFC1034, Nov-01-1987.

[7] P.V. Mockapetris, "*Domain names – implementation and specification,"* RFC1035, Nov-01-1987.

[8] J. Postel, "*Simple Mail Transfer Protocol*", RFC821, Aug-01-1982.

[9] Sun Micro-systems, "*RPC: Remote Procedure Call Protocol specification:*

Version 2," RFC1057, Jun-01-1988.

[10] G. Malkin, "*RIP Version 2,"* RFC2453, Nov 1998.

[11] R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, T. Berners-Lee, "*Hypertext Transfer Protocol -- HTTP/1.1,"* RFC2616, Jun 1999.

[12] D. Waitzman, C. Partridge, S.E. Deering, "*Distance Vector Multicast Routing Protocol,"* RFC1075, Nov-01-1988.

[13] J. Myers & M. Rose, "*Post Office Protocol - Version 3,"* RFC1939, May 1996.

[14] S. Shepler, B. Callaghan, D. Robinson, R. Thurlow, C. Beame, M. Eisler, D. Noveck, "*NFS: Version 4,"* RFC3010, Nov 2000.

[15] NetBIOS Working Group. Defense Advanced Research Projects Agency, Internet Activities Board, End-to-End Services Task Force, "*Protocol standard for a NetBIOS service on a TCP/UDP transport: Concepts and methods,"* RFC1001, Mar-01-1987.

[16] NetBIOS Working Group. Defense Advanced Research Projects Agency, Internet Activities Board, End-to-End Services Task Force, "*Protocol standard for a NetBIOS service on a TCP/UDP transport: Detailed specifications,"* RFC1002, Mar-01-1987.

[17] Ethereal Homepage, "http://www.ethereal.com".

[18] NetXRay Homepage, "http://www.netxray.co.uk".

[19] Wireshark Homepage, "http://www.wireshark.org".