TCP/IP 網路效能量測

影響 TCP 效能的因素很多，除了協定本身的標準、演算法，也包括實作的方式、記憶體管理的機制以及系統的架構和底層使用的通訊技術等。藉由網路的量測，可以讓我們測試網路元件 (component) 的可靠性，幫助我們瞭解參數的設定是否適宜？網路效能的瓶頸為何？在本文中，我們將介紹幾種在 Unix/Linux (或 Windows) 系統上常見的量測工具並說明其使用方法。

為何要對網路進行量測

網路量測是一項很複雜的工作，但是經由資料的蒐集分析，可以幫助網路管理人員找出效能的瓶頸並調整網路的架構。除此之外，分析的結果也可以讓我們瞭解網路的特性並提供修改網路系統或設計網路模擬實驗所需要的參考依據。不管是數學模型的分析或是網路模擬實驗也都經常會需要實際量測的到的數據。在發展網路協定時，有時可能也會需要藉由觀察封包的內容來驗證實作的正確性。

量測的工具的種類

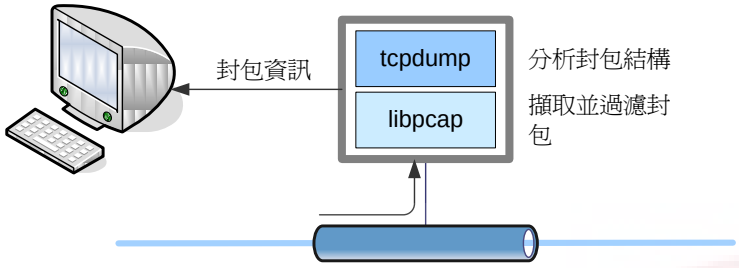
由於網際網路的普及，目前在網路上已經有許多的工具可用來量測並評估 TCP/IP 的效能了。這些工具以量測目的來區分的話大致上可以分成兩種：一是網路的監測工具，如 Tcpdump、Tcpstat。另一種則效能評比的工具，如 Ttcp、NetPerf 以及 NePIPE 等。一般而言，監測的工具大多可以檢視封包內容，這一點在發展和驗證網路協定時是很有幫助的，這類的工具除了在實作網路協定時經常被採用外，也常被作為網路管理用途使用。效能評比工具可以產生網路流量、評估網路的執行效能，如傳輸量、延遲時間、延遲時間的差異量以及可用的頻寬等。類似這樣的工具除了使用在網路管理外，也經常在調整實作細節時被使用。

幾種常見的量測工具及其使用方法

接下來，我們要介紹幾種常見的工具，這些工具都可以在網路上免費取得。我們在介紹這些工具的同時也會使用例子說明如何使用這些工具來量測網路的效能。

Tcpdump

Tcpdump 是一種使用命令列輸入參數選項的網路流量監測工具程式，Tcpdump 提供許多不同的命令列選項，可用來檢視 TCP/IP 封包的內容。如下圖所示，tcpdump 使用 Packet capture library (libpcap) 來捕捉網路上的封包，經由 tcpdump 的分析將 header 和 payload 等相關資訊以較容易觀看的格式輸出。由於簡單容易使用，目前幾乎所有的 Unix/Linux 平台都有內建 tcpdump 程式。



Tcpdump 的命令列格式

tcpdump [-adeflnNOpqRStuvxX] [-c count] [-C file_size] [-F file] [-i interface] [-r file] [-s snaplen] [-T type] [-U user] [-w file] [-E algo:secret] [expression]

以下簡略地列出幾個 tcpdump 的選項

-c : 在收到指定的包的數目，tcpdump 就會停止

-F: 從指定的文件中讀取條件表示式,忽略其它的條件表示式

-i: 指定監聽的網路介面

-w 直接將包寫入文件中,不分析也不顯示在螢幕上 (稍後可用-r 選項讀取)

-r 從指定的文件中讀取資料 (一般通過-w 選項產生)

tcpdump 的命令列表式法主要分成選項 (option) 與運算式 (expression) 兩個部份。使用的時候，可以指定 expression 作為過濾資料的條件 (預設會擷取網路上所有攔截的封包)。在 expression 中可指定(1)監測的範圍及 Port，如 host、net 或 port (預設值為 host)。(2)傳輸的方向，如 src 、dst、dst or src 以及 dst and src。(3)協定,如 ip 、arp、rarp、tcp、udp 等。除此之外,也可使用像是 gateway,broadcast、less 或是 greater 邏輯運算，以及 not、and 或是 or 運算等，這些關鍵字可以依照需要做不同的變化。

Tcpdump 的輸出格式

src > dst: flags data-seqno ack window urgent option

其中 src 和 dst 分別表示來源和目的 IP 位址。Flags 表示 header 的 flag 狀態，如 S (SYN)、F (FIN)、P (PUSH)或 R (RST)，若無 flag 時則以單一的”.”符號表示。data-seqno 表示此封包載送的資料序號 (sequence number) 範圍,ack 則是接收端預期下次應該收到的封包序號。window 表示接收端的緩衝區大小，urgent 表示 urgent data，最後的 option 則是指 TCP header 的選項 (例如 <mss 1024>)。

要善用 tcpdump 所提供的功能就必須知道如何使用命令列選項,接下來我們以實際的例子來說明。

範例：擷取特定主機的封包

#tcpdump host *xlinux*

tcpdump: listening on eth0

(1) 20:55:09.085786 192.168.0.2.1203 > 192.168.0.252.telnet: S 3625556889:3625556889(0) win 16384 <mss 1460,nop,nop,sackOK> (DF)
(2) 20:55:09.085813 192.168.0.252.telnet > 192.168.0.2.1203: S 3457556464:3457556464(0) ack 3625556890 win 5840 <mss 1460,nop,nop,sackOK> (DF)
(3) 20:55:09.088025 192.168.0.2.1203 > 192.168.0.252.telnet: . ack 1 win 17520 (DF)
(4) 20:55:09.163477 192.168.0.252.telnet > 192.168.0.2.1203: P 1:13(12) ack 1 win 5840 (DF) [tos 0x10]
(5) ……

在上面的例子中，192.168.0.2.1203 是 Client 端行程，而 192.168.0.252.telnet 則是 Server 端行程。在(1)中，Client 端行程送出一個帶有 SYN flag 的封包，表示想與向 Server 端行程建立一個連線 TCP 連線。此為一建立初始化連線的封包，封包序號由 3625556889 到 3625556889，共 0 個位元組。MSS 是 1460 個位元組。接著 (2) Server 端行程送出一個 SYN-ACK 的封包，表示願意和 Client 行程建立連線並告藉由 ACK 的封包序號讓 Client 端行程知道此封包已到達 Server 端。(3) Client 接著送出一個帶有 ACK flag 的封包；如此一來，Client/Server 行程之間的溝通動作已確認完成，接下來可以正式進行資料傳送的工作。

範例：觀察特定的應用程式

以 FTP 為例， FTP 使用 Port 20 傳送資料，Port 21 則作為傳送控制訊息使用，因此我們可以在 tcpdump 的參數中指定觀察特定的 Port 編號，或者以關鍵字 ftp (20)、ftp-data (21) 代替。

格式：tcpdump ‘host host and (port ftp or ftp-dtat)’

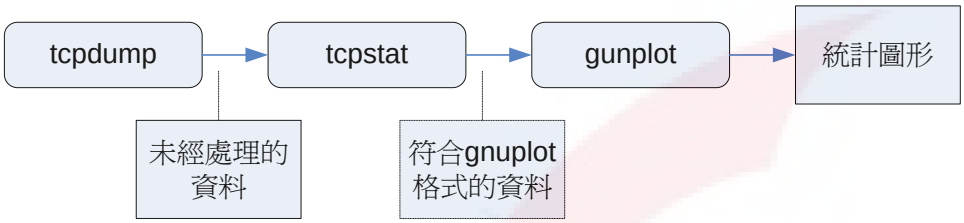
tcpdump 'host 192.168.0.252 and (port ftp)'

tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 96 bytes
(1) 02:30:26.897677 IP 192.168.0.5.1560 > 192.168.0.252.ftp: S 302452555:302452555(0) win 16384 <mss 1460,nop,nop,sackOK>
(2) 02:30:26.897724 IP 192.168.0.252.ftp > 192.168.0.5.1560: S 3081197035:3081197035(0) ack 302452556 win 5840 <mss 1460,nop,nop,sackOK>
(3) 02:30:26.899513 IP 192.168.0.5.1560 > 192.168.0.252.ftp: . ack 1 win 17520
(4) 02:30:26.975499 IP 192.168.0.252.ftp > 192.168.0.5.1560: P 1:21(20) ack 1 win 5840
(5) 02:30:26.998957 IP 192.168.0.5.1560 > 192.168.0.252.ftp: P 1:15(14) ack 21 win 17500

Tcpstat

- Report TCP related statistics periodically
  - Bandwidth used
  - No. of packets exchanged

- Avg packet size
- libpcap: Capture and filter packets
- tcpstat: Count packets
- Invoked with superuser privilege



Netperf

- Active testing tool (benchmarking)
- Measure available bandwidth between two nodes
- Major features:
  - Generate different traffic patterns
    - Bulk data transfer (e.g. FTP)
    - Interactive data exchange (e.g. rlogin)
  - Detailed and precise measurement
  - Besides TCP/UDP, also support datalink and other network protocols
- Develop by HP

安裝 netperf

1. 下載 netperf: nhttp://www.netperf.org/netperf/NetperfPage.html
   解開壓縮檔 tar -zxvf netperf-2.3.tar.gz
2. 編輯 makefile, 修改有關 NETPERF_HOME 及 CFLAGS 的設定
   NETPERF_HOME = /usr/local/netperf
   #CFLAGS = -O -D$(LOG_FILE) -DNEED_MAKEFILE_EDIT
3. 執行 make
4. 執行 make install
5. 接著將 netperf 12865/tcp 加入/etc/services 中

Netperf－使用方法及選項

量測前要先確定 netserver 是否在另一端執行

netserver

Starting netserver at port 12865

TCP 量測

netperf -H 140.116.142.168

TCP STREAM TEST to 140.116.142.168

Recv Socket Size bytes	Send Socket Size bytes	Send Message Size bytes	Elapsed Time secs.	Throughput 10^6bits/sec
87380	16384	16384	10.71	0.71

UDP 量測

netperf -H 140.116.142.168 -t UDP_STREAM -- -m 1024
UDP UNIDIRECTIONAL SEND TEST to 140.116.142.168

Socket Size bytes	Message Size bytes	Elapsed Time secs	Messages Okay #	Errors #	Throughput 10^6bits/sec
65535	1024	10.00	114656	0	93.95
65535		10.00	0		0.00

參數說明:

-H remote_host

-t testname (TCP_STREAM, TCP_RR, UDP_STREAD…)

-m 設定訊息的大小

Netpref 預設的 UDP 資料流 (stream) 大小是 9216 位元組。在傳送時很容易引起緩衝區溢位使得封包在接收端遺失；假如有太多的封包遺失，Netpref 量測到的的頻寬會較實際的小很多。所以在這裡我們使用-m 參數重設訊息的大小，--符號是要告訴 Netpref 將-m 這個參數加入 UDP_STREAM 這個模組中。

Interactive data exchange

netperf -H 140.116.142.168 -- -r 64, 1024
TCP STREAM TEST to 140.116.142.168

Recv Socket Size bytes	Send Socket Size bytes	Send Message Size bytes	Elapsed Time secs.	Throughput 10^6bits/sec
87380	16384	16384	11.34	0.68

-r 參數定義傳送 request 以及 response 的速度，單位 (位元組/秒)。在這個例子中，請求的訊息是 64 位元組，回應訊息是 1024 位元組。