改良之網路及伺服器服務斷線偵測回報機制

包蒼龍　　陳建伯　　黃立行
大同大學資訊工程學系
tlpao@ttu.edu.tw, jbchen@mcu.edu.tw, lshuang@mcu.edu.tw

摘要

本文提出五種機制來改良一般斷線偵測系統的問題，包括採用開啟 socket 的方式來偵測，以避免一般傳統使用 SNMP 的問題，採用緩衝回報來避免網路狀況不穩定而產生太多的回報或假警報，採用異地偵測的方式來避免 single point of failure 的問題，運用精簡回報方式來減少通知訊息，利用平行偵測方式來縮短偵測的時間。

關鍵詞：網路管理、偵測回報

Abstract

In this paper, we propose five mechanisms to improve the traditional network service interruption detection system. First, we open socket to detect the service instead of using SNMP method. Second, we use the buffered report mechanism to avoid the false report in an unstable network. Third, we use remote detection to avoid the problem of single point of failure. Forth, we use combined report to reduce the number of reports. Fifth, we use parallel detection to reduce the detection time.

Keywords: network management, network service interruption detection and reporting

1. 前言

由於網路服務的多樣化，網管人員所需要管理的網路設備以及網路伺服器也愈來愈多。如何能在第一時間內就知道網路或伺服器發生問題，一直是網管人員最迫切需要的功能。

許多學校都會建置斷線自動回報機制，在 TANET2004 中，戴江淮教授提出一篇「網路斷線自動警報系統」[5]，利用 Switch 上的 SNMP 功能，建置一套警報系統，當網路斷線時，系統會以電子郵件或是簡訊的方式通知網管人員。SNMP 是網管人員的一大幫手，但是要使用 SNMP 進行網路管理時，會有幾個問題。首先，網路設備必須要能支援 SNMP 功能，另外，若要監管的是伺服器，則伺服器必須要安裝並啟動 SNMP 服務。當網路設備或是伺服器啟動 SNMP 時，則會有被駭客入侵的潛在危險。因此，雖然 SNMP 功能強大，但是使用時必須非常的小心。

為了避免 SNMP 可能造成的問題，本研究不使用 SNMP 來建置系統，而是利用開啟 TCP socket 方式來偵測網路或伺服器的上線狀態。當我們利用開啟 TCP socket 方式來對網路設備或是伺服器進行連線狀態測試時，除了可以用基本的 ICMP 來測試設備的連線外，也可以針對伺服器特定的服務進行測試，以避免機器本身可以連線但是卻無法提供服務的狀況。同時，利用開啟 TCP socket 連線來測試的方式，可以對所有的 TCP 網路服務進行測試，而不需要對每一種服務撰寫一支專屬的程式。除了可以減少程式開發的時間之外，更可以增加系統的通用性。

除此之外，斷線偵測回報機制還存在幾個問題。第一個問題就是可能會出現假警報。為了避免發生假警報的情況，我們採用一種緩衝回報機制，除了可以減少假警報發生的次數之外，也還是可以偵測設備連線時好時壞的問題。第二個問題是斷線偵測回報主機所在的位置。如果這台主機是放置在校內，當學校對外連線不通時，斷線偵測回報機制對於校內所有設備都會認為是上線狀態。再者，即使發現有設備服務中斷，由於對外網路不通，因此可能無法發送電子郵件以及簡訊通知管理人員。再者，斷線偵測回報主機本身也有可發生問題。因此，必須提供偵測這台主機上線的機制以及異地交互偵測系統。

通常，斷線偵測回報機制會根據所需偵測的伺服器一一設定測試的項目，但是，通常一位管理者可能同時管理多台伺服器或網路設備。當其中一台設備出問題時，系統會通知該管理者。可是，當全校對外網路不通時，則系統會發現同時有多台設備斷線，然後一一發送電子郵件以及簡訊給管理者。假設某位管理者負責二十部伺服器，則在這種情況下，管理者可能會收到二十通簡訊，除了造成管理者的困擾之外，同時也浪費簡訊資源。因此，針對回報機制，本系統是根據管理者來發送通知訊息而不是根據每台設備發出通知訊息。在上述的情況中，即使該管理者所管的二十台伺服器同時被偵測到已斷線時，管理者也只會收到一封郵件以及一通簡訊，內容則會包含這二十台伺服器的詳細資訊。

另外，以斷線偵測回報主機偵測數十部設備時會遇到延遲的問題。這是因為採用循序的方式來測試時，如果其中有某幾台設備發生斷線的狀況，則必須等待 timeout 時間到後，才可以繼續偵測下一台設備。針對這種現象，本系統採用平行偵測的執行方式，讓系統可以在同一個時間內，同時偵測

多台設備。

2. 系統架構

本節將說明本文所提出的系統架構與實作方法。我們分成五個部分來說明，包括偵測服務的方式、緩衝回報機制、異地交互偵測、斷線通知以及平行偵測的方式。

2.1 開啟 Socket 方式

為了避免使用 SNMP 所帶來的問題，本系統不使用 SNMP 來偵測，而是使用 PHP 程式的開啟 socket 方式，來偵測設備的上線狀態。之所以會選用 PHP 程式，是因為偵測上線狀態必須是一個可在背景定時執行的 script，發送回報通知的簡訊以及電子郵件也是 script，而偵測到的結果希望能以網頁的方式呈現出來。PHP 程式不僅能夠撰寫網頁程式，同時也可以 script 的方式來執行[2]，對於程式開發或是後續維護人員而言，僅需熟悉一種程式語言即可。

使用 PHP 來偵測設備或伺服器上線狀況的程式碼[1,3]如圖 1 所示。

```
function ping($host)
    $packet="\x08\x00\x8e\xff\x00\x00\x00\x00\x69";
    $socket=socket_create(AF_INET,SOCK_RAW,1);
    socket_connect($socket,$host,null);
    socket_send($socket,$packet,strlen($packet),0);
    $result=socket_read($socket,255);

function tcping($host,$port)
    $socket=socket_create(AF_INET,SOCK_STREAM,
            SOL_TCP);
    $result=socket_connect($socket,$host,$port);
```

圖 1 PHP 偵測程式碼

其中，ping 副程式是利用開啟 socket 的方式來送出 ICMP request 的封包；而 tcping 副程式則是以開啟 TCP socket 連線來檢查伺服器特定服務的狀態。並將這支 PHP 程式以 script 的方式，運用 crontab 定時執行，即可定時檢查設備或伺服器當時的上線狀態。

2.2 緩衝回報機制

一般的斷線回報機制，通常一偵測到斷線時，就會立刻發出警訊。採用這樣的方法，很可能會因為一時網路或是設備的忙碌而發出假警報。因此，有些回報機制會連續測試三次離線狀態才發出警報。其狀態圖如圖 2 所示，其中狀態說明如下：

ON 服務/設備的狀態是正常

ON-OFF 服務/設備無回應，但連續無回應的次數，尚未累積到被認定成已離線的標準

OFF-SMS 服務/設備被認定為已離線，系統會寄發警告訊息

OFF 服務/設備被認定為已離線，系統不寄發警告訊息

OFF-ON 服務/設備正常回應，但連續正常回應的次數，尚未累積到被認定成狀態是正常的標準

F 用來判斷狀態是否應改變的變數

圖中的 transition a 和 b 為偵測的結果，其說明如下：

a 偵測到服務/設備正常

b 偵測到服務/設備斷線

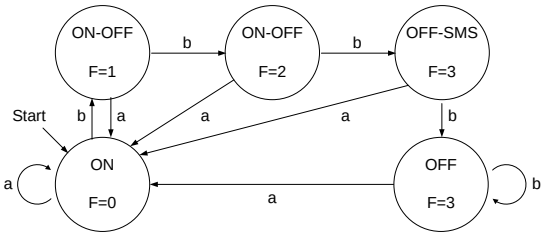


圖 2 一般回報狀態圖

這種方式雖然可以解決大部分的假警報，但是在網路或是設備狀態不穩定時，偵測的結果可能是時好時壞的情況，將會發出許多不必要的警報。本文所採用的緩衝回報機制，就是為了要解決這樣的問題。緩衝回報機制的運作原理說明如下：當偵測到設備斷線時，必須連續偵測三次斷線才會發出警報；當發出警報之後，如果有偵測到上線情況，則不會立即認為該設備已經上線，而是採用緩衝的方式，連續偵測三次上線，才會認為設備真正上線。利用這樣的方法，最主要的目的是為了避免設備一下子上線，一下子斷線，造成系統不斷的發出警報。緩衝回報的狀態圖如圖 3 所示。其中，OFF-ON 狀態為設備從斷線狀態轉換到上線狀態的過程。

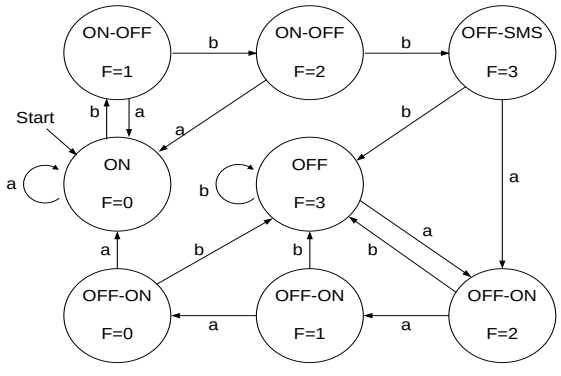


圖 3 緩衝回報狀態圖

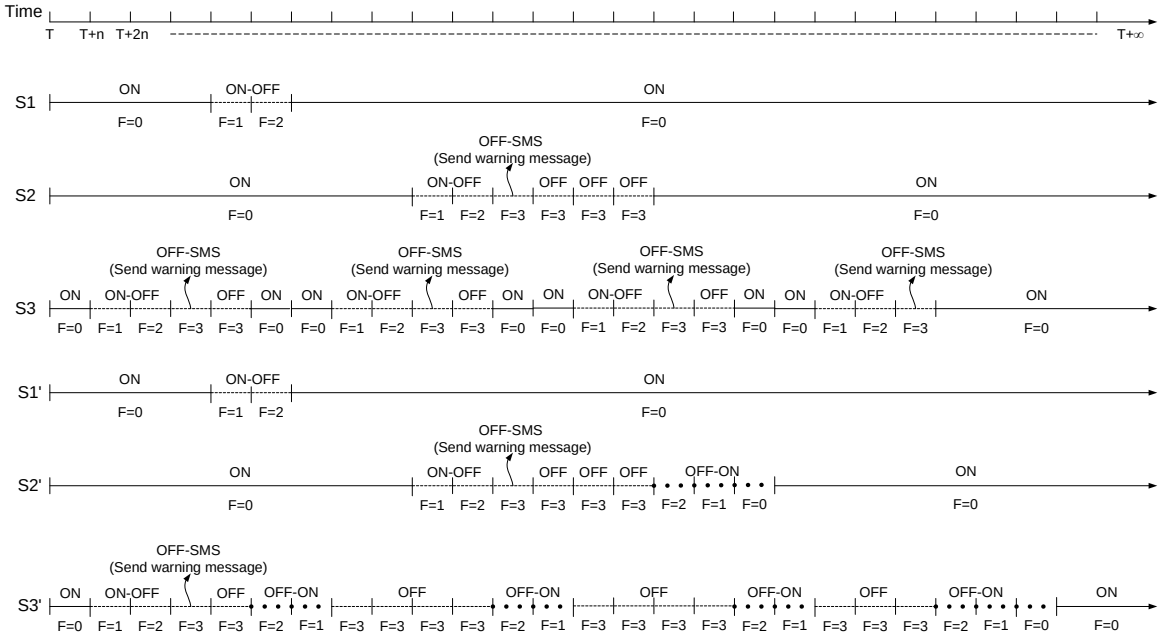


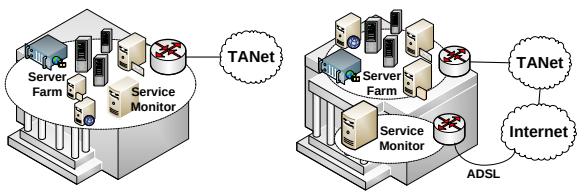
圖 4 一般回報與緩衝回報時間軸圖示比較

一般回報與緩衝回報機制，最主要的差異可以用圖 4 的時間軸圖示說明。在圖 4 中，S1、S2、S3 為一般回報；S1’、S2’、S3’則是緩衝回報。在 S1 和 S1’中，只有兩次偵測斷線並不會發出回報通知。在 S2 和 S2’中，偵測到連續三次以上的斷線，也只會發出一次的回報通知。在比較 S3 和 S3’可知，S3 在網路不穩定的情況之下，會發出多次的回報通知訊息，而採用緩衝回報機制的 S3’，在網路不穩定的情況之下，只會發出一次的回報通知，不會發出額外不需要的回報通知。

2.3 異地偵測

斷線偵測回報主機主要目的是為了要偵測校內的網路設備或是伺服器發生不正常斷線的狀況。通常這台主機會放置在校在的 Server Farm 或是其他的網段，如圖 5(a)所示，其中 Service Monitor 為斷線偵測回報主機。這時會有幾種狀況發生。首先，假設全校對外的網路連線發生問題時，由於斷線偵測主機是放置在校內，因此可能偵測到校內設備以及伺服器仍然是上線狀態，即使可以偵測對外連線不通，管理者也無法收到發出的電子郵件，而且系統也無法透過網路來傳送簡訊。為了避免這種問題發生，斷線偵測伺服器不應該放置在校內的骨幹網路上。一般學校除了有連接到 TANET 的網路外，通常都還會申請其他固網業者的 ADSL 線路，當成是備援或是負載平衡來使用，因此，本文建議將該主機放置在 ADSL 線路上，才能偵測到 TANET 斷線。其網路架構如圖 5(b)所示。

即使將斷線偵測回報主機放置在 ADSL 線路上，仍然還有幾個問題存在。該主機在線路上雖然已經和 TANET 網路分開，但是實體放置的位置卻是在同一個機房內，這會面臨到另一個問題，那就是假設全校或是機房因故停電，則斷線偵測主機自然也會因為沒有電源而無法發揮作用。另一個問題是，斷線偵測主機本身的 Single Point of Failure，也就是說，假如這台主機本身無法正常運作，則所有的異常斷線都無法偵測到。因此，本文建議採用異地偵測的機制，同時也必須存在交互偵測的系統，才能確保系統能正常運作。異地偵測架構如圖 6 所示。



(a) 位於校內網路上　　(b) 位於 ADSL 線路上

圖 5 偵測伺服器位置架構圖

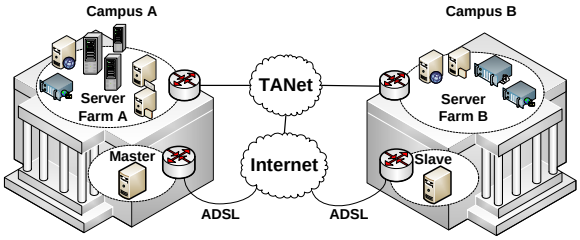


圖 6 異地偵測架構圖

此架構的說明如下。有些學校可能會有多個校區，或是可以與其他學校合作，在校本部以及第二地分別架設一台斷線偵測回報主機。放置在校內 ADSL 線路上的主機稱為 master，提供主要的偵測回報服務。放置在第二地的伺服器稱為 slave，其主要的目的是定時偵測 master 的上線狀態。一旦 slave 發現 master 已經斷線，則 slave 除了會回報給斷線偵測回報系統的管理者之外，同時 slave 也會接管 master 的工作，繼續偵測校內設備及伺服器的狀態，如此才能避免 Single Point of Failure 的問題。當然，為了確保 slave 能夠正運作，master 當然也必須要隨時偵測 salve 的上線狀態。

2.4 精簡回報

由於校內的伺服器及網路設備很多，通常一位管理者可能同時管理數十台的網路設備或是伺服器。當網路或是電源發生問題時，會造成同一網段或是全校設備不通。此時，如果程式是根據每一台設備斷線就發出一個通知的話，則管理人員有可能在同一個瞬間接收到大量的電子郵件及簡訊，不僅造成管理者的困擾，也因為大量發送簡訊造成成本的提高。因此，本系統採用了一種稱為 message pool 的方式，來解決上述問題。當系統針對某一設備發出斷線回報時，斷線回報訊息會先被放置在 message pool 中。真正負責發出簡訊以及電子郵件的程式，會定時掃描 message pool，並且先把 pool 中同一個接收者的多個 messages 合併成一個之後才發出。實際運作方式如圖 7 所示。圖 7 的上半部是一般的回報方式，下半部為採用 message pool 的回報方式。在圖中可明顯看出，採用 message pool 方式可以減少大量的通知訊息給同一位使用者。

再者，當系統偵測到設備斷線時會發出斷線回報通知訊息，如果這個斷線通知訊息僅發出一次的話，則可能會因為管理者一時的疏忽而漏掉這個通知訊息。系統初期的設計，是每隔一段固定的時間(例如五分鐘)就發出一次訊息。但是根據實際執行的結果發現，以這種持續性的發出簡訊，對管理者而言也是一種困擾，因為如果設備是在半夜發生問題，則管理者可能整夜收到數十通簡訊。因此，本系統最後決定採用累進時間的方式來傳送訊息。也就是說，系統可以在第一次發現離線狀態時發出簡訊，如果持續離線的話，半小時後發出第二通簡訊，二小時後再發出第三通，八小時後發出第四通，而不同的伺服器或設備，發出回報通知的間隔時間，是可以依需求調整的。如此，才能夠持續地用較不擾人的方式提醒管理者。

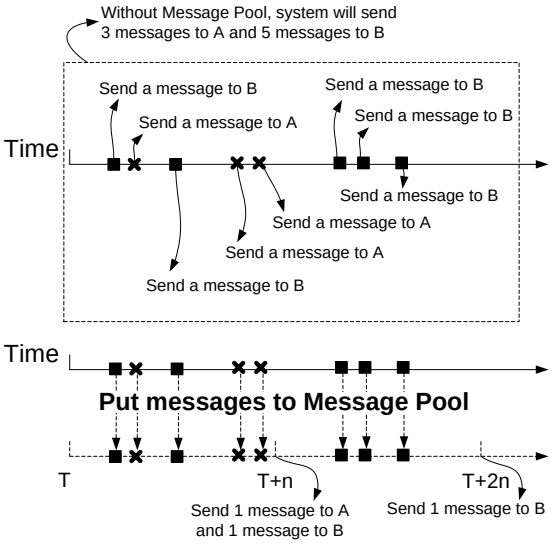


圖 7 Message pool 回報運作方式

2.5 平行偵測

在緩衝回報機制中，我們可以發現時間對系統的影響是非常重要的。通常斷線偵測回報機制所要偵測設備的 IP 以及 port number 都是儲存在資料庫中。當程式從資料庫中取出所要偵測的設備後，通常是以循序的方式一台一台偵測。如果設備是上線的情況，則很快就會有回應；如果設備是斷線狀態，則系統必須等待 timeout 時間到之後，才能確定設備為斷線狀態。由於偵測設備數量很多，且要等待 timeout，因此可能會造成所有的設備測試完成一次所花的時間較長，而影響到緩衝回報機制的運作。為了解決這個問題，由於 PHP 缺乏 multithread 的功能，所以在本系統中，我們將偵測的程式獨立在一個 PHP script 中。主程式偵測每一個設備時，都必須以 exec()的方式呼叫偵測程式[4]。如此，系統中就可以有許多偵測程式同時地偵測不同的設備，然後各自將偵測的結果儲存到資料庫中。而偵測程式之間不會相互影響。

3. 執行結果

系統執行後，除了發生斷線時會以電子郵件及簡訊通知管理者之外，也提供一個網頁介面讓管理者登入觀察設備上線狀態。在本系統執行的環境中，有兩個校區分別稱為 Campus A 以及 Campus B，兩個校區分別有該校區的 Server Farm 及該校

區所申請的 ADSL 線路，網路架構如上述圖 6 所示。在網頁中可以分別檢視 Campus A 和 Campus B 的設備狀態。圖 8 和 9 所示為兩個校區所有的設備都處於正常狀態。當有設備發生斷線時，設備會以不同顏色來顯示、左下方即時資訊也會顯示斷線的設備，同時網頁會發出警示的聲音來提醒管理者設備異常，如圖 10 所示。因此，管理者在上班時間只需開啟此網頁，也可以在第一時間知道設備發生異常。

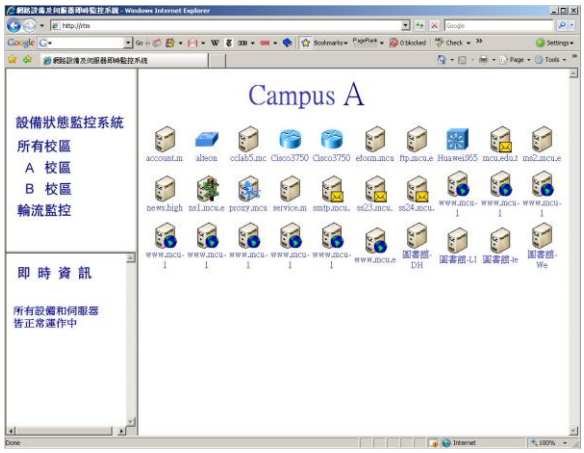


圖 8 A 校區設備正常狀態

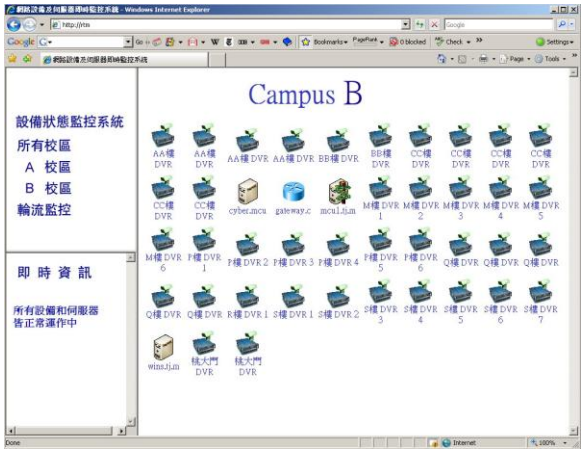


圖 9 B 校區設備正常狀態

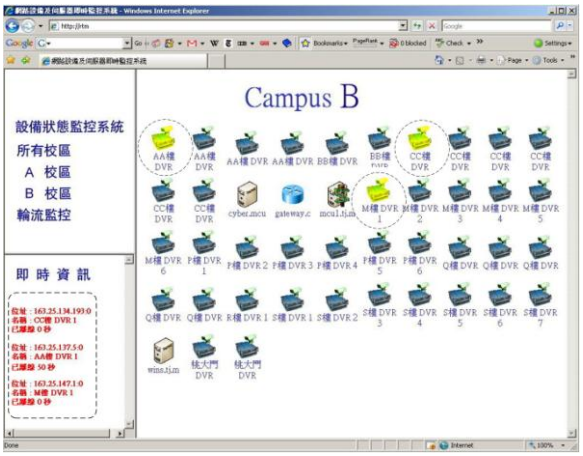


圖 10 設備異常狀態

4. 結論

本文主要的目的是改良傳統斷線偵測回報系統的問題。本系統已經上線多年，並根據實際發生的問題加以改良，因此可以滿足目前大多數學校使用上的需求。

參考文獻

[1] INTERNET CONTROL MESSAGE PROTOCOL, RFC 792

[2] David Sklar and Adam Trachtenberg, PHP Cookbook, 2002

[3] PHP Manual, http://www.php.net/

[4] 葉昌福，PHP 函式庫參考手冊，旗標出版，2004

[5] 戴江淮、賴正延、姜玲鳳，網路斷線自動警報系統，TANET2004