一個基於混合式網路流量分析之新的異常偵測模型

A New Model of Anomaly Detection Relying on the Analysis of Network Traffic

王智弘
國立嘉義大學資訊工程研究所
wangch@mail.ncyu.edu.tw

郭力瑋
國立嘉義大學資訊工程研究所
s0930313@mail.ncyu.edu.tw

摘要

入侵偵測系統扮演著為系統偵測可能的攻擊行為、再通知防火牆機制進行各種惡意行為應變機制之重要角色；入侵偵測技術主分為「異常偵測」與「誤用偵測」，前者定義網路上的正常行為，應用各種相關的技術來判斷、預測可能的異常行為，後者已定義已知攻擊行為為主，為目前入侵偵測軟體所使用之方式。在許多異常偵測的相關研究中，其偵測的工作，往往仰賴著他人所蒐集好之人工網路資料來進行異常分析，但是使用此類資料來分析異常行為，即使在分析該離線資料時有良好的偵測率，但是只要混合了真實網路的資料，往往會使誤報率大幅增加、偵測率下降。因此，本論文在此提出一個混合人工、真實網路資料之入侵偵測模組，使得過濾後的真實網路資料對於訓練過程的妨礙降到最低，為未來實現線上即時異常偵測之重要歷程。

關鍵詞：異常偵測、網路攻擊、錯誤通報、網路資料合併、離線式封包分析

Abstract

Intrusion detection systems play an important role in detecting possible attacks and notifying the firewall to carry out the process of providing counter-measures for malicious behaviors. There are two primary techniques of intrusion detection systems: “Anomaly detection” and “Misuse detection.” The former defines normal network behaviors with specific techniques in order to identify and predict anomalous behaviors. The latter models known attacks and is popular in most recent utilization of IDS products. In several similar researches, the analysis and evaluation of the anomaly detection usually rely on manually pre-captured network traffic. Nevertheless, even though the detector can perform well in off-line simulated data, it might not work as well as expected in real traffic. That is because merging real traffic into simulated traffic for a detector usually leads to a lower detection rate and higher false alarm rate. Consequently, in this paper we provide an intrusion detection module which relies on the analysis of the combination of manual and real traffic. To minimize the obstruction of training process, the real traffic is required to be filtered by a misuse detector to remove all known attacks. This work has an important result that can be used to get an “on-line” anomaly detector into practice in the future.

Keywords：Anomaly detection, Network intrusion, False alarms, Network traffic combination, Off-line packet analysis

壹、介紹

在電腦資訊技術日益發達的今日，各項軟硬體技術日益進步，且網路技術也隨之突飛猛進，頻寬亦越來越寬廣，同時也為使用者創造了更舒適的網路環境；然而，在如此方便的環境下，表面上乍看是相當舒適的資訊環境，其實檯面下卻是暗潮洶湧，病毒（Virus）、木馬（Trojan）以及各種入侵行為等，正以倍速成長當中 ，資訊系統的資料保全、網路交易的安全性等因此日亦受到重視，因此理想上必須要能夠有效偵測各種「惡意碼」（Malicious codes）的傳播與發生，為了達到這樣的目標，除了一般的防毒軟體與防火牆，常常還要搭配專門負責偵測各種入侵行為的「入侵偵測系統」（Intrusion Detection System，簡稱 IDS）機制，成為了資訊安全的領域中不可或缺的一環。

入侵偵測系統，就「偵測範圍」上，分成 network-based 以及 host-based 兩種，前者為特定網域的偵測而後者較偏向單一主機上系統行為（如 system call）、狀態（如有限狀態機制）的偵測。

而就「分析方式」而言，分成「誤用偵測」(Misuse detection)「異常偵測」(Anomaly detection) 兩種。在 [1][3] 以及許多相關研究皆指出，目前所有的入侵偵測產品皆使用前者的方式來定義已知的攻擊，建立成很多個「模板」（Patterns）來偵測可能的攻擊行為，優點在於偵測上的誤判會比較低，因為已知的網路攻擊行為都已經定義成特定的規格，然而必須要常常更新攻擊的模板以應付更新的攻擊行為，當然在未更新前如果遭遇到模板中沒有定義的攻擊，這樣的入侵偵測系統就會發生「漏報」（False negative error）而偵測不到；而「異常偵測」的入侵偵測系統，透過學習、統計甚至是封包分析或監測系統行為狀態的方式，來定義所謂的正常行為模組，然而所謂的正常行為定義是非常困難的，而且會隨著不同的網路環境而改變 [2]，因此當正常行為定義得不夠完善時，在收錄正常行為資訊庫（Normal profile）中未蒐集到的部分，當發生時，就會被異常偵測系統視為異常的行為而發生「誤報」（False positive error），這種入侵偵測方式不需要內建攻擊資料庫或相關模板，能夠從正常的行為中來分析甚至預測可能出現的異常行為，然而這樣的異常分析方式，目前尚欠缺更有系統的整合，而且比較高的「誤判」（False alarm）問題仍未獲得良好的改善，因此目前還很難應用在今日的資訊安全產品上，然而這種能夠偵測系統未定義的異常行為之技術，是未來入侵偵測領域的趨勢之一。

因此，「誤用偵測」以及「異常偵測」彼此的優缺點是彼此互補的，如果能針對兩大偵測技術進行有效的整合，彼此截長補短，相信能夠成為一個更加理想化的入侵偵測系統 [1]。

「誤用偵測」以及「異常偵測」等兩大入侵偵測技術之整合上，是不容易有效地實現的；例如，在論文 [14] 中提到了利用「模板」（Pattern）對應的方式來偵測異常行為，並且以改善貝式 TCP 網路（Bayes TCP Network）的方式來作為誤用偵測技術的根基，並提出了結合兩大入侵偵測技術「混合式系統」之觀念，然而該論文中所偵測的異常行為主要為「攻擊的意圖」，即 DARPA 資料集中之 probe 類掃描行為，尚未擴展至其他種類攻擊行為的異常偵測上，成為本論文中針對各種攻擊行為進行異常偵測研究的一個動機。

本論文接下來將探討的各章節如下：第二部分將探討本次研究所應用到的一些相關知識，以及異常偵測系統運作過程中所遭遇到的問題，第三部分將探討與分析利用誤用偵測的一個方式—SNORT，與異常偵測系統之間的搭配，並提出一個整合模型以提高系統在真實網路上之可用性，而整合後的實驗過程與結果分析將於第四部份討論，而第五部份將闡述本研究將來能夠再深入研究、改善之處。

貳、相關研究工作

異常偵測系統往往需要定義正常的網路行為以利於系統對於正常行為以外的異常行為進行判斷與分析，然而，對於正常行為的定義相對於已知攻擊的定義是較為困難的工作；因此，在本部份將詳細介紹所謂的異常偵測系統並闡述各種所謂的異常行為，以及介紹各種相關的異常偵測模組，來有效定義系統的正常行為。

(一) 異常偵測系統概觀

異常偵測系統在入侵偵測系統的領域中，目前仍然是不夠成熟的一項技術，除了上部分所提到的正常行為定義困難與誤報之問題外，顧名思義，使用「異常偵測」技術之入侵偵測系統，負責偵測系統上的異常行為，然而，「異常行為」卻不見得必定代表著「攻擊行為」[4]，包含網路狀況的異常以及使用者不熟悉網路環境所造成的異常等等，均非出自於有侵入他人系統的意圖，然而在某些狀況之下，「異常」與「攻擊」，兩者之間的界定卻是很不容易的。

所謂的「異常行為」(Anomalous behaviors)，經研究與觀察後主要可以歸納成五大類 [6][10]：

(1) 使用者行為異常：未經授權的使用者，可能以網路掃描的方式來偵測目的主機有提供的服務，為入侵系統前常有的暖身動作，如 *ipsweep* 以及 *portsweep* 均屬此類。

(2) 系統漏洞：系統有漏洞，如程式有 bug 一般，成為各種入侵行為能夠得逞的捷徑，當然在電腦上有軟體具有漏洞時，便容易遭遇特定種類

的攻擊，而會被偵測出的異常情形便因漏洞種類而異 。

(3) 回應異常：遭受攻擊的主機，其所回應的封包，亦可能使得入侵偵測系統發出警告。

(4) 攻擊行為本身之異常：攻擊行為多數由所謂的「駭客」（Hacker）所造成，而攻擊的過程，可能因駭客本身認知的不足或者是掌握的狀況不足之下，將出現攻擊本身的問題，成了入侵偵測系統的線索；例如，文字性的協定如 FTP、SMTP 以及 HTTP 等，能夠允許大、小寫的指令，入侵者可能為了方便全部用小寫，但是一般使用者利用相關應用軟體所產生的指令通常是大寫的，因此將造成顯而易見的異常狀況。

(5) 試圖閃避網路安全機制之異常：如同病毒、木馬等惡意碼，攻擊行為常常會試圖閃避入侵偵測系統，可能從應用層的異常向下隱藏至網路、傳輸層的異常，必須由低層的封包欄位來分析才能偵測出該攻擊行為。

一般而言，異常偵測系統的系統運作上，必須要有兩大階段：「訓練」（Training）以及「測試」（Testing）。在「訓練」異常偵測模組時，根據異常偵測系統的本質定義，此類入侵偵測系統並非以已知攻擊行為作為知識基底，而是利用特定的方式模組化正常的行為，為了能夠順利地進行異常偵測，必須要讓系統先經過訓練的階段，利用「無攻擊」（Attack-free）的網路資料作為訓練系統的根基，訓練完成後會得到記錄正常行為的正常行為資訊庫，當然偵測效果之優劣將仰賴訓練結果之好壞而定；得到正常行為資訊庫之後，即可開始進行「測試」，也就是正式針對新的網路資料進行異常偵測的工作。

當然在訓練異常偵測系統的過程中，要達到網路資料完全無攻擊事實上是很難達成的，如果在訓練系統時遭遇攻擊行為，系統將把這些攻擊行為定義為正常行為，到了正式偵測的階段時，只要碰到同樣的攻擊，系統就會遺漏掉。針對這樣的問題，如 M. V. Mahoney 曾試過將訓練切割成很多部分來進行 [12]，以期各個訓練分段能夠互相補正彼此訓練時發生錯誤的地方（意即訓練時遭遇攻擊行為），然而，經實驗證實，在各分段所能夠偵測到的攻擊，在混合之後偵測率卻大幅下降；因此，可以結合誤用偵測的入侵偵測技術來輔助原異常偵測系統使得用來訓練的網路資料更趨近於無攻擊狀態，讓偵測工作能夠得到更加良好的結果。

(二) 異常偵測系統模組

訓練網路導向（Network-based）異常偵測模組的方式，有些採用統計的方式、有些採用機器學習甚至是累神經網路的方式來進行，而在本論文中，所應用到的異常偵測放式，主要採用網路封包分析的方式，再利用條件機率的原理建立正常行為的規則，所應用到的模組分列如下：

(1) PHAD Model

在異常偵測系統中，所謂的「異常」泛指在正常行為資訊庫中所不存在的值，有許多種攻擊行為所造成的異常，在應用層層面是無法觀測出，而必須由網路層以及傳輸層來分析才得以順利偵測出。

在 [8] 的工作中，提到了利用封包標頭分析異常的方式，而在 [12] 中，更應用 [8] 的方式提出分析封包標頭異常的模組—PHAD（Packet Header Anomaly Detector），此模組利用 PPMC 法（Bell, Witten, and Cleary, 1989），針對低層網路封包的 33 個欄位進行觀察，並以異常值（Anomaly score）計算的方式來調查封包標頭欄位的異常度。

關於異常值的計算，讓我們給定變數n為某封包欄位在訓練模組時所觀測的次數，而r為在這n次的觀測中所遭遇到的不同值個數，因此我們可以說「在訓練時遭遇到r個異常狀況」，如果再繼續觀察下去，再出現新值（即異常的欄位值）的機率便為r / n；然而我們再給定t代表所經過的時間，異常值A_i與機率呈倒數關係n / r，因此，某異常封包欄位i的異常值 1 便成為下列方程式 (1)：

$$A_i = t*n / r \qquad \textbf{(1)}$$

對於每個觀察的封包欄位，每當出現異常欄位值，便會計算一次異常值A_i，最後再加總得到該封包p的總異常值A_p，如下列方程式 (2) 所示：

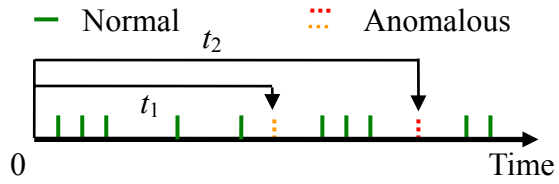
$$A_p = \Sigma_i A_i = \Sigma\, t_i * n_i / r \qquad \textbf{(2)}$$

PHAD異常偵測模組觀察 33 個封包欄位，然而每個欄位的可能值從 1 種到 2^{32}種不等，倘若全部儲存在記憶裝置上是不必要且相當浪費空間的；因此，PHAD-C32 模組針對每一個欄位實際出現的值進行叢集化，最接近的值群分成一塊，最大分成 32 個叢集。

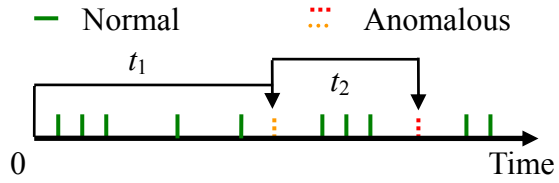
PHAD 由於負責分析網路層到傳輸層的低階層封包標頭資料，擅長於偵測從應用層藏匿的攻擊行為，由於特定種類的攻擊如 *land*（來源 IP＝目的 IP）、*smurf*（ICMP 的 *Echo* DDoS）等，在應用層面是無法偵測出來的。

(2) ALAD Model

由於 PHAD 是屬於靜態模組，異常值的計算上，其 t 值是從系統開始學習時起算，其異常值是採用觀察數量的平均，當如果訓練過程結束後馬上出現異常值時，將使得異常值的計算出現問題。



圖一　靜態模組（PHAD）



圖二　非靜態模組（ALAD 後的模組）

當異常值的計算上超過門檻值時，系統當然會發出警告，但是由於是平均異常值，該值可能一時小不下來，因此會造成重複發出警報的狀況，而出現更多的誤報情形，如上圖一。

在 ALAD（Application Layer Anomaly Detector）的異常偵測模組 [10] 中，以「非靜態模組」（Non-stationary model）改進了 PHAD 中靜態訓練模組的缺點，時間的計算上採用每次出現異常值的時間間隔而非自開始訓練系統時起算（如上圖二），且時間非指實際的時間，而為封包數，因此解決了重複警報的問題。

ALAD 是一個負責針對應用層的 TCP session 進行異常偵測的模組，它利用「貝氏定理」（Bayes law）並透過實驗歸納出與攻擊最相關的封包欄位，得到五個固定式的「規則」：

① P (src IP | dest IP)：針對與某主機連線的所有使用者 IP 清單。

② P (src IP | dest IP, dest port)：針對利用某服務的使用者 IP 清單。

③ P (dest IP, dest port)：針對某主機所提供的服務，本規則常偵測到 probe 類的掃描行為。

④ P (TCP flags | dest port)：某主機上所提供的某服務，其接受之 TCP 連線旗標（flags）順序組合。

⑤ P (keyword | dest port)：針對某主機服務上的連線，特定通訊協定所產生的「關鍵字」（Keyword），ALAD 偵測前 1,000 位元組的關鍵字；如 R2L（Remote to Local）類的攻擊較容易被此規則偵測出。

ALAD 的偵測範圍上，與 PHAD 並無任何的重疊（Overlap），且 ALAD 模組只儲存有興趣的欄位資料，故儲存上比起 PHAD 更節省空間，因此結合二異常偵測模組能夠得到更佳的偵測率。

(3) LERAD Model

前述之 ALAD 模組，其規則建立方式為固定的，因此在異常與攻擊的偵測上較無彈性，因此需要一個更有彈性的動態規則建立方式。

在 LERAD（LEarning Rules Anomaly Detector）[9][11] 中，它延伸了 PHAD、ALAD 二模組的異常值計算方式與條件機率式規則建立原理，動態地建立正常行為規則。

我們給定 $A, B, C, X, Y, Z\ldots$ 等表示封包的欄位變數，而 $a, b, c, x, y, z\ldots$ 表示封包欄位的值，在 X = x, Y = y, Z = z 的「前提」（Antecedent）之下，會有 p 的機率出現 A = a, B = b, C = c 的「結果」（Consequent），因此可以表示成方程式 (3)：

$$p = \Pr(X=x, Y=y, Z=z \mid A=a, B=b, C=c) \quad \textbf{(3)}$$

其異常值的計算，與 ALAD 相同，且異常值在訓練過程結束後就此固定，作為正式偵測時發出警報的依據。至於建立的規則，可以把它改寫成：

$$\text{If } A=a, B=b,\ldots \text{ then } X=x, Y=y,\ldots \quad \textbf{(4)}$$

LERAD 之動態規則建立的演算法如下列三步驟：

① 自訓練好的網路封包資料中，隨機挑選符合 $n / r = 2 / 1$ 的配對，並且隨機挑選 L 對以及隨機排序 n / r 值相同的規則。此步驟會產生很多「候選規則」（Candidate rules），仍需要進一步篩選來減少規則的重複性。

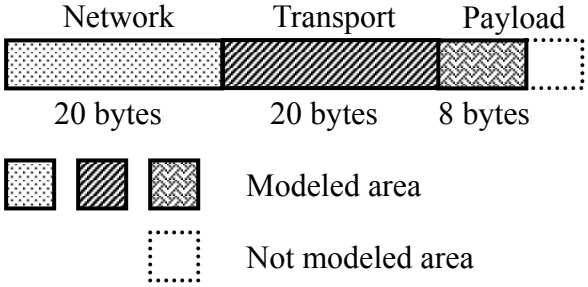
② 利用「涵蓋性測試」（Coverage test）從候選規則中篩選出適合的規則，依照排序好的規則來依序比對符合的訓練樣本（Training example），如果有沒有涵蓋到新樣本的規則，表示該規則為一個重複的規則，將會遭到刪除；最終篩選後所剩下的規則，方為 LERAD 模組真正欲利用的正常規則。

③ 逐漸擴大包含的訓練樣本範圍至全部的訓練資料為止，重複步驟①與②。

由於 LERAD 模組的規則建立過程具有高度之隨機性，因此針對同樣的訓練資料，最後會產生的規則及其數量，將有些微的差異。

(4) NETAD Model

NETAD（NEtwork Traffic Anomaly Detector）模組專門針對流入的封包進行分析，並且以逐位元組檢測的方式來分析自 IP 標頭起始之前 48 位元組的資料（如下頁圖三），檢視網路交通可能含有的異常與攻擊行為 [6]。



圖三　NETAD 的封包分析範圍

在前述之 PHAD、ALAD 以及 LERAD 三模組中，主要的差別在於偵測的範圍以及規則建立的方式，而異常值的計算方式大同小異，在 NETAD 之異常偵測模組中，基於前三個異常偵測模組尚有部份考量點上的不足，針對了 PPMC 法的異常分數計算方式做了以下的改善，分成了以兩個主要的子模組：

① 因為訓練的過程中是沒有攻擊事件的，因此n值應為 0，並以n_a來表示前次發生新值至訓練過程結束為止分析到的封包數，而t為封包數而不為實際的時間。

② 當r值接近 255 時（一個位元組能出現的可能值從 0~255 共 2^8種），正象徵著該欄位的狀態幾近均勻分配，因此，需要視r值減少產生規則之權重（weight），如下列 (5) 表示NETAD的第一個子模組：

$$t * n_a (1 - r / 256) / r \qquad \textbf{(5)}$$

③ 令某值i的位元組發生的頻率為f_i，t_i為該值i第一次出現回溯至上次出現新值所歷經的封包數，因此第二個子模組為 (6)：

$$t_i / (f_i + r / 256) \qquad \textbf{(6)}$$

綜合兩個子模組(5)以及(6)，能夠得到NETAD模組中單一封包的異常值A_{NETAD}，如方程式 (7) 所示：

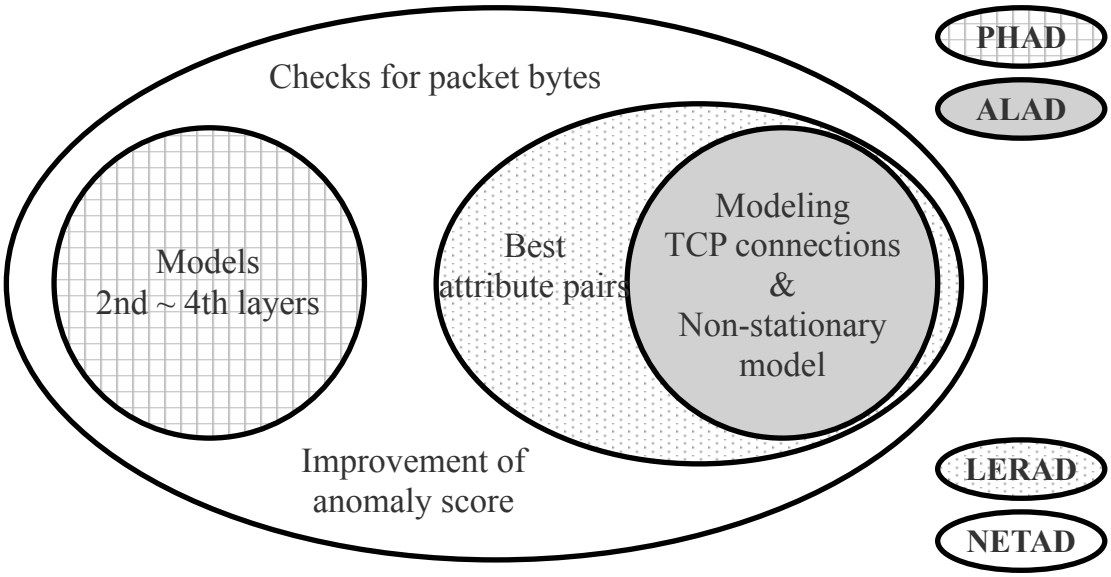
$$A_{NETAD} = \Sigma [t * n_a (1 - r / 256) / r + t_i / (f_i + r / 256)] \qquad \textbf{(7)}$$

以上 LERAD 與 NETAD 二異常模組演算法為本論文中測試的主要對象。

(5) SAD Model

SAD（Simulated Artifact Detector） [7]為一將真實網路資料與模擬的網路資料混合，並進行訓練以及測試，發現其偵測效率較原來模擬網路下的測試還要低，原因主要在於真實網路的資料，很難確保完全無攻擊存在，並且突顯出種種模擬網路資料的問題；因此，以真實的網路資料來訓練系統並且偵測真實網路中的異常與攻擊，是本論文目前正積極努力的目標。

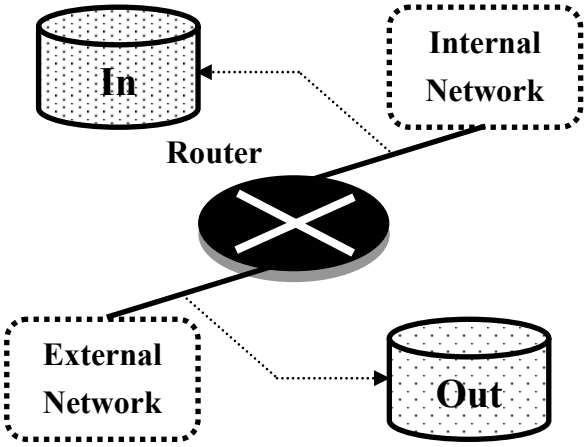
針對 PHAD、ALAD、LERAD 以及 NETAD 四大異常偵測模組，其之間的功能涵蓋關係如下圖四所示；下方圖四中，PHAD 只偵測資料連結層到傳輸層的範圍，ALAD、LERAD 以及 NETAD 等均只偵測流入的封包且具有非固定式學習模組的特性，LERAD 以及 NETAD 均有演算法篩選出最好的規則組合，而 NETAD 模組則改善了異常值的計算方式。



圖四　PHAD、ALAD、LERAD 以及 NETAD 等模組之功能涵蓋關係圖

(三) 分析方式

在異常偵測系統的偵測效能評估上，倘若由研究者自行從真實網路上抓取資料並測試，由於網路的資料表現方式以及分布情形，隨著網路環境的不同而異，因此將是一件更加麻煩的工作；因此，多數的研究都會採用美國 MIT 的 Lincoln 實驗室所製作的 DARPA 模擬網路資料集 [5]，如下圖五所示，這個網路資料是由人工的方式製作出來的，是一個專為異常偵測系統所設計的資料集，根據異常偵測系統的設計規格，主要分成內部網路（In）與外部網路（Out）兩種資料集。



圖五　DARPA 網路資料之蒐集環境

由左圖五，本論文將進行研究的部分主要為內部網路資料（In-spec），此部份資料已包含 DARPA 中攻擊種類的大多數。

DARPA 的網路資料集中，主要包含了五個星期的 TCPDump 格式網路資料（下表一）以及四大類別 201 種攻擊（下表二）等。

表一　DARPA 網路資料集組

網路資料集	敘　　述
Week 1 & 3 **(Training)**	全然不含攻擊的網路資料，主要作為異常偵測系統訓練系統之用。
Week 2 **(Training)**	含有經標記過的攻擊行為，目的在於幫助異常偵測系統的建立，得不使用。
Week 4 ~ 5 **(Testing)**	一般的測試資料，訓練好的異常型入侵偵測系統，針對這兩週的資料發出警報，再由評估程式 IDEVAL 來得之偵測率資訊與誤報情形。

表二　DARPA 網路資料集之攻擊類別

攻擊分類	敘　述　與　實　例
Probe	通常為攻擊的意圖，正式發動攻擊前往網會先蒐集目的主機的相關資訊，並尋找脆弱點（Vulnerabilities）作為下手目標。如： *queso*、*portsweep* 等…
DoS	為知名的「阻斷服務式攻擊」（Denial of Service），主要目的在於透過通訊協定弱點、癱瘓等方式讓受害主機（Victim）無法繼續正常提供服務；近年來更出現許多 DDoS 攻擊方式，使得追溯與偵測更加困難，成為網路安全領域中的頭號敵人。如：*mailbomb*、*CrashIIS*、*land*、*smurf* 等…
R2L	R2L（Remote to Local），顧名思義，利用網路軟體的弱點，未經授權非法登入他人的系統而稱之。如：*guess passwd*、*CrashIIS*、*land*、*smurf* 等…
U2R	U2R（User to Root）為一般的正常使用者，利用伺服軟體的弱點，得到使用特權指令的權限，成為管理者的角色，在異常偵測領域中屬於最不易偵測出之攻擊種類。如：*fdformat*、*casesen*、*sqlattack*、*loadmodule* 等…

參、混合式入侵偵測模組

多數異常偵測模組常常使用 DARPA 的網路資料集作為系統訓練與測試的根基，然而這些資料集卻有著潛在性的問題；茲在此將提出使用混和網路的方式來讓異常偵測模組更能夠適應真實網路的資料。

(一) 人工模擬網路資料的問題

儘管多數異常偵測系統採用 MIT Lincoln 實驗室的 DARPA 1998 / 1999 來進行異常偵測系統的偵測評估工作，但是這樣的人工網路資料卻存在著一些潛在的問題，雖然這些問題多數不至於嚴重影響偵測率，但是突顯出與真實網路資料的顯著差異性：

(1) 由於 DARPA 資料集是以人工方式，利用有限的主機與模擬的網路環境來進行網路資料的蒐集，其封包部分欄位內容的複雜度並不如真實網路，如右方表三所示。

(2) 承前述問題，使得本身就具有學習、預測功能的異常偵測模組，能夠猜測出此模擬資料的規律性，造成偵測效率測定上的失真。

(3) 即使在 DARPA 中表現得很出色的異常偵測模組，到了真實網路環境下，其表現便完全走樣。

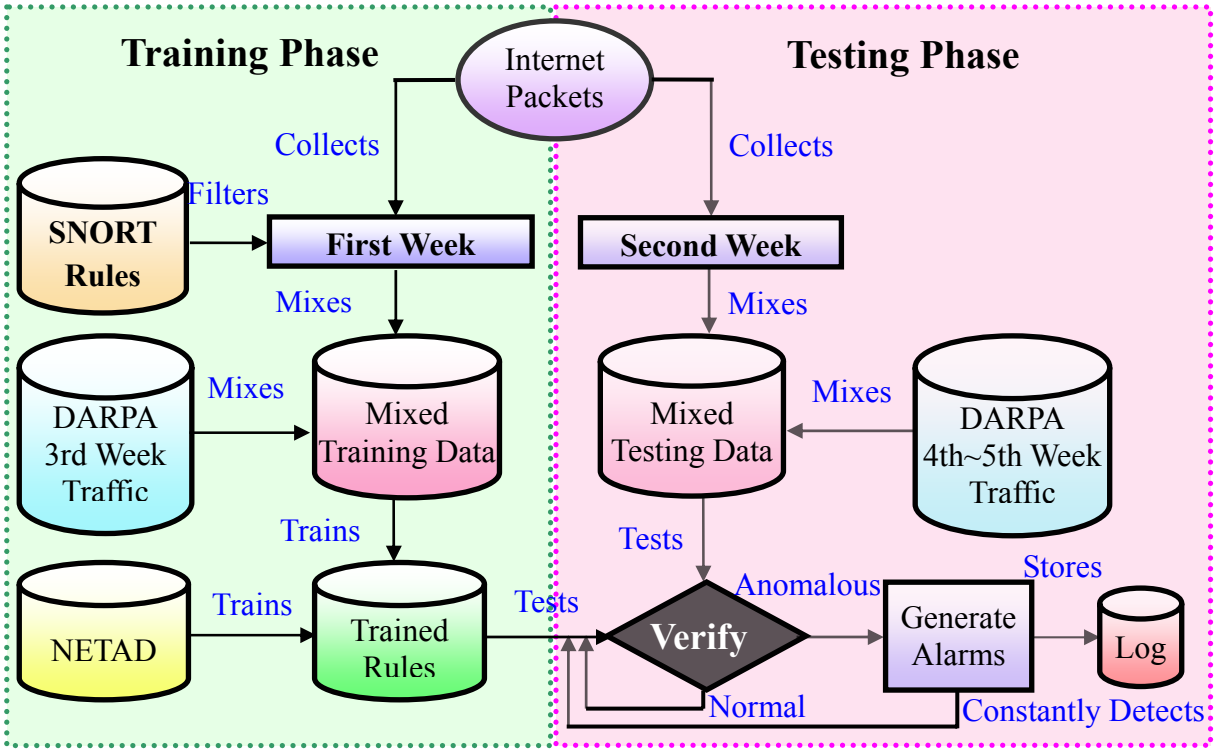
(4) 真實網路環境下，無攻擊網路環境不易實現。

(5) 資料蒐集的替代方案少，除了 DARPA 外，出於其他組織機構的網路資料，多有版權以及隱私權問題，不利於用在深入的研究。

表三　DARPA 中觀察到的問題欄位

欄位	DARPA	真實網路
TTL	9 種	不一定
Checksum Error	無	通常有
TOS	4 種	不一定
IP Fragment	無	通常有
HTTP Request	變異性少	多變化
SSH	只有 1 種請求訊息	多樣化請求訊息

DARPA 的 IDEVAL，對於異常偵測具有一定的參考價值，因為真實網路的模擬是相當困難的；因此，倘若能與真實網路資料相互結合，更能夠增加網路資料的變異性。然而，目前混合真實網路資料的方式，如 [7] 中所述，因為真實網路環境很難有無攻擊的狀況，因此混合真實網路資料往往會妨礙異常偵測系統的訓練過程，造成特定種類的攻擊在訓練系統時被建立至正常行為的模組中，造成往後若遭遇相同網路攻擊行為，便無法順利偵測，出現更多的「漏報」。



圖六　離線式之結合異常偵測與誤用偵測技術偵測混合網路中異常行為之模型

有鑒於以上問題，茲將採用結合誤用偵測的方式來過濾真實網路環境的資料，作為真實網路環境下異常偵測系統的訓練資料，以求模擬與真實網路之間更進一步的連接。

(二) 模擬、真實網路資料的結合與偵測

SNORT，是一個著名、原始碼公開的規則式入侵偵測工具，截至目前為止已經超過 4,000 個攻擊規則，並且動態地隨時更新中；SNORT 能夠協助蒐集網路的資訊，並且可以依照自己的需求改寫封包抓取、記錄、忽略甚至發出警報的規則，來過濾出符合需求的網路交通資訊，屬於誤用型之入侵偵測工具。本論文將採用 SNORT 來作為異常偵測工作的輔助工具，以規則過濾後的封包資料來作為異常偵測系統訓練的所需資訊。

要達成這樣的目的，就必須要混合模擬與真實的網路資訊，之間必須要經過很多封包的篩選與過濾之過程。

在上頁圖六中，我們提出的異常偵測模型描述如下：承襲了一般異常型入侵偵測系統的方式，將整個過程分成「訓練」與「測試」兩大步驟。在訓練資料時，我們採用第一週的真實資料利用 SNORT 規則過濾後，與 DARPA 的訓練資料做混合，再利用 NETAD 演算法根據混合的資料動態產生正常行為的規則（其正常行為規則建立方式請參閱第二部分的 LERAD、NETAD）；進入測試步驟後，以第二週的真實資料與 DARPA 之測試資料混合，並根據訓練時所建立出之正常行為規則進行異常行為分析，然後根據規則發出警報並記錄到異常行為的資料庫中。

其關於圖六之詳細步驟如以下三大步驟所示：

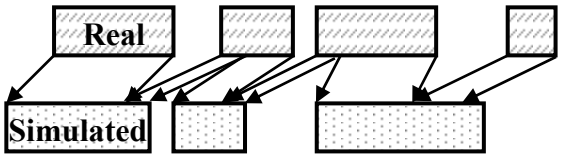
(1) 自行從網路蒐集封包

研究工作至目前為止，先以單一主機的資料為研究對象，蒐集期間為兩星期，第一個星期資料利用 SNORT 規則過濾後再與 DARPA 第三週資料混合，作為訓練用資料，第二週資料不經過濾直接與 DARPA 第四～五週資料混合，作為測試用資料。

(2) 混合網路資料與過濾封包

由於是離線式的網路資料分析，欲與模擬資料進行混合，首先必須修改封包內容使得真實資料與模擬資料的時間格式得以一致，避免在發出警報時可能出現的錯誤情形。

由於封包的時間不一定是連續的，造成在資料中的時間將會有很多「縫隙」(Gap) 因而產生；因此，必須從真實資料中逐一「對應」至模擬的網路資料上，如右上圖七為雙方網路資料混合時之對應的方式，此方式參考了 M. V. Mahoney 在 [7] 中所提到的封包資料合併方式，而在本實驗中盡量以 1 : 1 的比例混合雙方網路資料。



圖七　將真實時間對應至模擬的時間

在混合資料時，為了追求更好的執行效率，必須將與主要分析工作較無關之封包濾除，如下：

① 流出之封包。

② 對應到超過 1023 埠的 UDP 封包（一般為伺服端回應給用戶端的封包）。

③ 本端主機上提供的服務之所有伺服器回應之封包。

④ 非 IP 之封包。

⑤ 60 秒內倘若 ICMP、TCP 以及 UDP 指向同一 IP 之封包流入超過 16 個的時候，濾除此 60 秒內多餘的封包，以減少訓練時的計算量。

(3) 異常偵測

異常偵測的工作，主要應用到 LERAD 模組的動態規則建立以及 NETAD 的異常值計算方式，而 SNORT 之規則則成為本實驗中誤用偵測技術的根基；關於異常值計算方式，在前面第貳部分已有詳盡敘述，而其偵測結果與分析將在下一部分深入探討。

肆、實驗與分析

本實驗的環境為區域網路內之終端主機，使用 Windows XP SP2 之作業系統環境並且以 SNORT 進行網路資料的截取與過濾的工作，利用其入侵規則來過濾真實網路可能的攻擊行為，在誤用偵測領域中佔有很重要的地位；此外，並以 Lincoln 實驗室的 DARPA IDEVAL 異常偵測評估工具來分析偵測的結果。

表四　實驗所使用的各種網路資料集

資 料 集	說　　明
S	原 DARPA 資料集第 3～5 週。
S+U	DARPA 資料集與未經 SNORT 過濾之混合資料集。
S+F	DARPA 資料集與已經 SNORT 過濾之混合資料集。

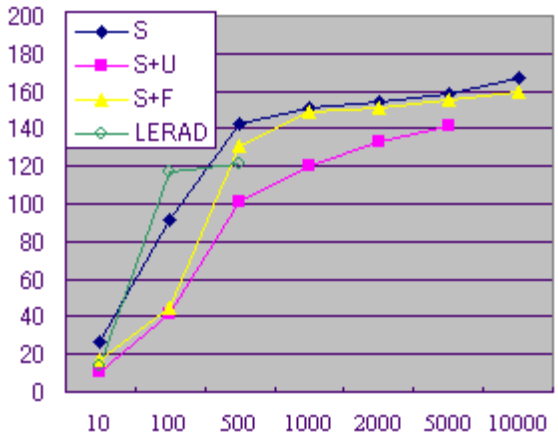
在上頁表四中，茲擷取兩週的 TCPDump 格式之網路資料，以用於與 DARPA 模擬網路資料集之結合。讓我們令原來 DARPA 的資料為 S，而未經 SNORT 規則過濾的真實網路資料為 U，而經過過濾的真實網路資料則為 F，第一週的真實資料用於跟 DARPA 之第三週資料混合作為訓練資料，而第二週的真實資料用於與 DARPA 之第四至五週資料混合作為測試資料，分別用以做 S、S+U、S+F 三種實驗的偵測性比較。其中第四～五週資料不論使否利用 SNORT 規則過濾，對於偵測結果影響不大，因為本實驗著重於系統訓練的過程。

在利用 SNORT 過濾真實網路中的攻擊行為，在本實驗中，兩週內自行蒐集的真實網路資料中，共蒐集到了超過 600 萬個以上的封包，其中，在第一週用於訓練系統用的網路資料中，共透過 SNORT 之入侵規則過濾出了 2,997 個攻擊行為，意即平均約每 3~4 分鐘會遭遇一次攻擊的行為，如此也突顯出了在異常偵測系統的訓練上，無攻擊的網路環境是相當難以實現的，因此需要搭配誤用偵測的技術營造幾近無攻擊的環境來幫助異常偵測系統的訓練工作。

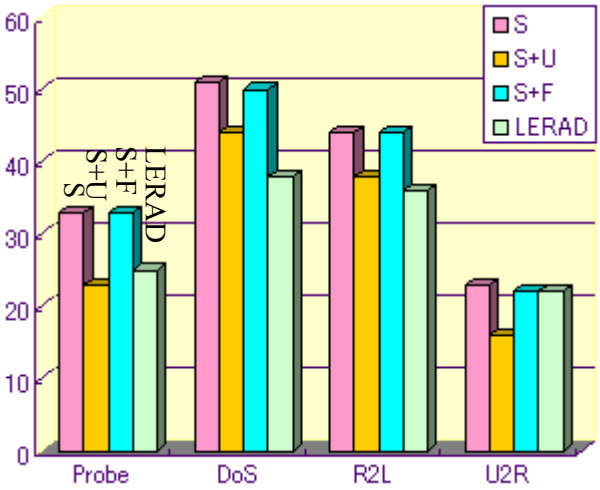
使用合併之後的網路資料來進行異常偵測，其結果如右上圖八所示；其中僅 LERAD（綠線）非使用 NETAD 演算法，而在 DARPA 的標準(1,000 個誤報之內）

之內有 121 個成功偵測數，而內部網路（In-spec，內部主機至路由器）共有 177 種攻擊可以評估，剩下的 24 種為外部網路（Out-spec，路由器至模擬網際網路）之攻擊，並不在本實驗的評估範圍內，因此 LERAD 在 DARPA 所規範之標準內其偵測率為 68.4%，共產生 143 個誤報。其他三實驗使用 NETAD 演算法，其中 S（藍線）為原始 DARPA 資料，在 1,000 誤報數的規範下有 151 個偵測數，偵測率為 85.3%；S+U（粉紅線）為混合未經 SNORT 規則過濾之網路資料，規範內只剩下 120 個偵測數，偵測率為 67.4%；而 S+F（黃線）為混合經 SNORT 規則過濾之網路資料，規範內有 148 個偵測數，偵測率為 83.6%。

右頁圖九為各實驗中對於 DARPA 網路資料集中各種攻擊類型的偵測效率，原資料集中有略勝一籌的趨勢，同時也驗證了 NETAD 針對異常值計算方式的改善，在實驗中偵測效率有實質性的提升。在 S+U 的實驗中，不論哪一種攻擊行為皆不如於原資料集 S 的偵測效率，突顯出未經過濾的真實網路資料，內部含有的攻擊行為確實會對異常偵測系統的訓練結果上有一定程度的衝擊；而 S+F 的實驗中，各項偵測效率均呈現出與原資料集 S 相同或相近的情形。



圖八　可容忍之誤報數（橫軸）與偵測率（縱軸）之關係圖



圖九　各種資料集對於不同種類攻擊偵測結果之比較

對於真實網路封包資料的分析，在利用相關異常偵測模組偵測時，其訓練用封包資料的篩選與調整，其對於訓練結果以及真實網路中偵測上的影響，仍有許多考量點需要進一步的研究與探討。

本實驗對於實現線上即時訓練、偵測的異常型入侵偵測系統而言是一個很重要的過程，同時也證明出結合誤用偵測的技術，能夠大幅減少系統訓練時遭遇攻擊為的可能性；因此，有助於異常偵測系統正常行為資訊庫的訓練，使其正確性得以提升，並且在未來將不再是一個只能偵測模擬網路資料或離線真實資料的模組，進而達到成為混合式系統（Hybrid system）的理想。

伍、結論與未來展望

異常型入侵偵測系統，由於必須定義正常之行為來訓練系統，因此在網路封包分析的方法上，無攻擊的網路環境對於系統訓練而言是相當重要的，然而傳統的異常偵測研究中，多使用人工的模擬網路環境來評估偵測的效率，如此並無法與複雜的真實網路環境順利接軌。

在本論文中，提出了一個結合誤用偵測技術來輔助異常偵測模組進行離線式網路異常封包分析的方法，得到使用未經過濾的資料來訓練異常偵測系統，對於其訓練結果上具有一定程度的衝擊；因此，茲利用 SNORT 的規則當作誤用偵測技術之根基來過濾真實網路資料，得到了與原方是相當接近之偵測結果，象徵著未來可以朝向線上直接以真實網路資料即時訓練模組並即時偵測系統，為傳統離線式人工資料分析與線上即時真實資料分析之橋樑，在此實驗之後，將繼續朝向本方向繼續努力與研究，並嘗試在不同網路環境之下，不同的正常行為資訊庫對於其偵測效率之影響及差異。

今日的網路攻擊，除了 Lincoln 實驗室之 DARPA IDEVAL 網路資料中所定義的四大類別之攻擊外，近年來從阻斷服務攻擊延伸出的分散式阻斷服務攻擊（Distributed Denial of Service; DDoS），出現頻率更是日益增加，並且有更多步驟、複雜化的趨勢，而其攻擊之「意圖」也更難被今日之入侵偵測系統所察覺。

因此在未來的方向上，除了本身關於綜合式入侵偵測系統的研究之外，可以針對封包資料的「混亂度」(Entropy)進行統計分析，並且利用「卡方分配」(Chi-Square Distribution)的方式來分析、統計各種異常行為的意圖 [4]，進而偵測各種多步驟且複雜之分散式阻斷攻擊，對於混合式入侵偵測系統而言，仍是一個可再進一步延伸的研究方向。

陸、參考文獻

[1] E. Biermann, E. Cloete, L. M. Venter, A Comparison of Intrusion Detection Systems, Computer & Security 20 (2001) 676-683.

[2] LTC B. D. Caulkins USA, J. Lee, M. Wang, Packet- vs. Session-Based Modeling for Intrusion Detection Systems, Proceedings of the International Conference on Information Technology: Coding and Computing (ITCC'05) 0-7695-2315- 3/05 IEEE, 2005.

[3] J. M. Estevez-Tapiador, P. Garcia- Teodoro, J. E. Diaz-Verdejo, Anomaly Detection Methods in Wired Networks: A Survey and Taxonomy, Computer Communications 27 (2004) 1569-1584.

[4] L. Feinstein, D. Schnackenberg, R. Balupari, D. Kindred, Statistical Approaches to DDoS Attacks Detection and Response, Proceedings of the DARPA Information Survivability Conference on Exposition (DISCEX'03) 0-7695-1897-4/03, IEEE, 2003.

[5] R. Lippmann, J. W. Haines, D. J. Fried, J. Korba, K. Das, The 1999 DARPA Off-Line Intrusion Detection Evaluation, Draft of paper submitted to Computer Networks, In Press, 2000.

[6] M. V. Mahoney, Network Traffic Anomaly Detection Based on Packet Bytes, Proceedings of the 18th ACM Symposium on Applied Computing (SAC), Melbourne, FL, USA, 346-350, 2003.

[7] M. V. Mahoney, P. K. Chan, An Analysis of the 1999 DARPA/Lincoln Laboratory Evaluation Data for Network Anomaly Detection, Computer Science Department, Florida Institute of Technology Technical Report CD-2003-02, 2003.

[8] M. V. Mahoney, P. K. Chan, Detecting Novel Attacks by Identifying Anomalous Network Packet Headers, Florida Institute of Technology Technical Report CS-2001-2

[9] M. V. Mahoney, P. K. Chan, Learning Models of Network Traffic for Detecting Novel Attacks, Florida Institute of Technology Technical Report CS-2002-08.

[10] M. V. Mahoney, P. K. Chan, Learning Nonstationary Models of Normal Network Traffic for Detecting Novel Attacks, Proceedings of the 8th International Conference on Knowledge Discovery and Data Mining, 376-385, 2002.

[11] M. V. Mahoney, P. K. Chan, Learning Rules for Anomaly Detection of Hostile Network Traffic, Proceedings of the 3rd IEEE International Conference on Data Mining 2003.

[12] M. V. Mahoney, P. K. Chan, PHAD; Packet Header Anomaly Detection for Identifying Hostile Network Traffic, Florida Institute of Technology Technical Report CS-2001-4.

[13] J. McHugh, Testing Intrusion Detection Systems: A Critique of the 1998 and 1999 DARPA Intrusion Detection System Evaluations as Performed by Lincoln Laboratory, Proc. ACM TISSEC 3(4) 262-294, 2000.

[14] A. Valdes, Detecting Novel Scans Through Pattern Anomaly Detection, Proceedings of the DARPA Information Survivability Conference and Ex- position (DISCEX'03) 0-7695-1897-4/03, IEEE, 2003.