行政院國家科學委員會專題研究計畫　成果報告

應用資料探勘技術於校園網路入侵封包偵測之研究

計畫類別：個別型計畫
計畫編號：NSC93-2213-E-216-015-
執行期間：93 年 08 月 01 日至 94 年 07 月 31 日
執行單位：中華大學工業管理學系

計畫主持人：張丁才

計畫參與人員：林廷洲、孫煒超

報告類型：精簡報告

處理方式：本計畫可公開查詢

中　華　民　國 94 年 10 月 30 日

應用資料探勘技術於校園網路入侵封包偵測之研究

計畫編號：NSC 93-2213-E-216-015
執行期間：2004.08.01 至 2005.07.31
主 持 人：張丁才 中華大學工業管理學系 副教授

摘要

隨著時代的變遷，資訊與通訊技術近年來顯然已漸漸的成為各種社會活動的關鍵基本角色。為了達成資訊的快速分享與日常作業處理的流暢，愈來愈多的組織機構建制日趨複雜的資訊與網路系統，以連繫客戶供應商與企業夥伴。然而，當網路基礎建設日趨完善，社會大眾對網路的依賴日趨重要，相對地網路安全問題也成為必須面對的重要課題。

資料挖礦就是對原始資料進行萃取與精練，藉以取得對企業有用的知識，可以有效的提升企業決策的品質與降低對資訊的需求。資料挖礦技術可透過一種或多種演算法對資料庫進行搜尋，並獲得可用於分類、分群、估計或預測的模式。

對於網路安全系統，不論是提供被動保護的防火牆或是主動偵測的入侵偵測系統，皆為條列式比對偵測，只能對已經知道的攻擊模式或封包進行攔阻。本文將應用類神經自組織映射圖網路(Self Organizing Map, SOM)和基因演算法(Genetic Algorithm)之資料探勘方法，以網路搜集正常封包及帶有攻擊性之封包作為分群輸入因子，透過上列分群的方法分析各封包群組，以提供校園及業者設計出更具精確性與效率的入侵偵測系統，創造出網路更安全的世界。

關鍵字：資料挖礦、自組織映射圖網路、基因演算法、網路安全

Abstract

Since ICT (Information and Communication Technology) has played a key role to support infrastructure of social activities. Therefore, many organizations have implemented sophisticated information and network system for information quickly sharing and general activities handling. Cooperates can contact with their customers, suppliers and partners through the Internet. Hence, the security of network has become an important issue.

For network system security, both firewall and intrusion detection system (IDS) provide rule-based intrusion detection function; they can just cut-off known intrusion models and attacked packets. However, using rule-based intrusion detection is time consuming. In this research, we propose to utilize self-organizing map (SOM) and genetic algorithm (GA) to detect attacking packets so as to increase the performance of IDS and keep the network actively.

Keywords: Data Mining, Self-Organizing Map, Genetic Algorithm, Network Security

1. 緒論

隨著時代的變遷、社會的進步，從農業社會進步到工業社會，再到現在的資訊爆發的年代，網際網路的發展日新月異，隨著網際網路使用的便利及快速，越來越多的商業活動都紛紛透過網際網路來進行，進一步的彰顯網際網路的重要性，隨之而來的網路安全問題，也變成了目前大家所關心的議題。網路上若無適當的安全機制，則無疑網路上的網路使用者、主機、提供網際網路服務的伺服器，將成為網路上駭客攻擊的目標。

駭客的攻擊目標，從獲得大型主機的使用權帳號到現在的竊取資料或發動大規模的攻擊癱瘓目標伺服器為主。而駭客攻擊會如此普及，造成這樣的情形最主要的原因是一些駭客網站提供一些自動化入侵工具和相關的入侵教學供下載，駭客不需要有任何的網路相關知識，就可以輕易的入侵想要攻擊的任何目標，甚至可以聚集大量的電腦主機同時對同一個攻擊目標發動分散式攻擊。而這些動作往往造成商業活動相當大的傷害，例如:yahoo、e-bay 等網站曾於遭受攻擊後，造成相當大的金錢上的損失。

隨著網路安全的觀念越來越受到重視後，發展出來的防禦機制，例如，防火牆(Firewall)，其作法為從簡單的阻擋 Port 的封包出入、限制某些 IP 位址的存取到更複雜的存取規則的訂定等。然而防火牆所能阻擋的攻擊仍有限，依然會被有經驗的駭客輕易的入侵。而入侵偵測系統(Intrusion Detection System，IDS)，主要係偵測出異常行為或攻擊行為的特徵，並給予入侵的警示，但仍需要花費相當多的時間對所有進出網路的封包進行攻擊特徵的條列式比對，故往往消耗龐大系統資源，既耗力又費時。

2. 文獻探討

2.1 攻擊模式

一般而言，依照攻擊的行為來區分，可大略的分為:單一封包攻擊、分段式攻擊、超載攻擊種。單一封包攻擊是指攻擊者所發出的封包含有入侵或破壞的行為稱之。大部分都是利用系統本身的漏洞來進行癱瘓目的主機或是藉此來取得系統使用者的權限，例如:CGI 漏洞、連線劫持。分段式攻擊是運用分割(Fragment)的封包來進行攻擊，此種類型是在 IP 層上處理封包重組的演算法有漏洞而使得駭客有可趁之機，例如:Jolt2 等。超載攻擊所用的方法就是消耗系統的資源而無法提供其他合法的使用者服務，也就是一般常見的 Dos 或 DDos 攻擊。而超載攻擊又可以分成兩種類型，一種是消耗系統的資源，另一種是消耗網路的資源；前者是消耗如 CPU 效率、記憶體或作業程序等系統資源，如 Land 攻擊。後者是以消耗網路寬頻的資源，使網路因為過多的封包阻塞而使封包無法移動，當然使用者也無法使用此寬頻資源，如 Smurf 攻擊。

如果依據攻擊性封包的共同特性來區分攻擊模式，可以簡單的分為 DOS 攻擊和非 DOS 攻擊兩種，Dos 攻擊是以封包直接令受害主機喪失正常運行的能力或提供正常的服務；非 DOS 攻擊是傳送內涵系統破壞程式或指令的封包，對於目標主機發動攻擊，使目標主機運作機制發生錯誤而陷入癱瘓。

DOS 攻擊又可以簡單的分為兩種，即為狹義與廣義。狹義的 DOS 攻擊是以一部攻擊主機來進行攻擊；而 DDOS 攻擊則是以多部攻擊主機來進行攻擊。一般廣義的 DOS 攻擊包含了狹義的 DOS 攻擊與 DDOS 攻擊。

一般的 DOS 攻擊為廣義的 DOS 攻擊，其約略包含了 Land、完全佔用整個寬頻、ICMP Smurf flood 等等。

2.1.1 Land

攻擊者利用作業系統處理封包時的系統漏洞所進行的攻擊行為，例如:攻擊者發送一個來源與目的 IP 位址相同的封包到目的主機的通訊埠，若目的主機對這封包不知道如何去處理時，在解譯時將耗費大量的資源，而導致該類服務或整個系統的癱瘓。

2.1.2 完全佔用整個寬頻

攻擊者由一台主機或多台主機發送大量的封包到同一個網域，而塞滿該網域的某一個網段的寬頻，使得該網段的伺服器無法發出服務的封包或接收要求服務的封包。例如:從 T1(1.544Mbps)發出大量的封包到串聯寬頻較小的網路線上，在其連接處造成大量棄置的封包，故因大量的封包無法順利到達目標主機，也就無法獲得應有的服務。

2.1.3 ICMP Smurf flood

攻擊者主機假造受害者主機 IP，對一個廣播網域(Broadcast)IP 位址發送 ping 指令封包；網域上所有主機都會回應 ICMP ECHO REPLY 封包給受害者主機，造成受害者主機網域的壅塞。

2.1.4 Teardrop 攻擊

主要是利用 IP 層封包重組程式的瑕疵。網路在傳送資料封包時，要考慮傳輸介面的最大傳輸單位(MTU, Maximum Transfer Unit)限制，將過大的封包切割成數個小封包，再將封包傳送到目標主機，在目標主機的 IP 層重組回原來的資料封包，Teardrop 攻擊就是利用封包分割碎片有些片段位置重疊，刻意造成不正常的序列，使得某些系統不知道要如何處理這種情形。

非 DOS 攻擊可約略分為三種：最容易的單一封包直接攻擊，即時性跳板攻擊和非即時性跳板攻擊三種。

所謂即時性跳板攻擊，即對 S 主機植入木馬，再即時直接對 S 主機下達對 F 主機攻擊指令，後由 S 主機攻擊 T 主機。如果是木馬程式預設每隔一段時間，S 主機就主動對 F 主機發動攻擊，即非即時性跳板攻擊。

非 DOS 攻擊模式簡單的說有：ICMP(ECHO、ECHO REPLY)與 UDP 封包、TCP 序號預測、電子郵件 SMTP(Simple Mail Transfer Protocol)與 POP(Post Office Protocol)等相關的通訊協定、SQL Injection、緩衝區溢位。

2.1.5 ICMP(ECHO、ECHO REPLY)與 UDP 封包

攻擊者傳送含有攻擊指令在 ICMP 或 UDP 封包的資料欄內的封包，至已經植入木馬的跳板主機中，木馬執行完程式後，會將執行完的結果再封裝在 ICMP(ECHO REPLY)或 UDP 資料欄中，並回傳給攻擊者。

2.1.6 TCP 序號預測

攻擊者 X 首先對目標主機 S 傳送短暫的 TCP SYN Flood 攻擊，使目標主機暫時無法回應其信任主機 Y，這時，攻擊者 X 假冒 Y 對目標主機 S 發送要求連線的封包，並將網路卡設為雜亂模式，且利用 Sniffer 機制觀察其回應的 TCP 封包序號，再假冒 Y 的 IP 位址向 S 發送 TCP 預測序號封包，使 S 誤以為真的是來自 Y，而與其取得連線關係，X 取代原來 Y 的地位，對 S 發動入侵。後來就算 Y 和 S 取得連線，X 仍可以和 S 保持連線狀態。

2.1.7 電子郵件 SMTP(Simple Mail Transfer Protocol)與 POP(Post Office Protocol)等相關的通訊協定

藉由使用者打開來路不明含有後門程式的病毒郵件，入侵系統後建立後門，而形成一個後門通道。

2.1.8 SQL Injection

網頁的程式撰寫，對於使用者輸入字元未作較為完善的檢查與過濾，往往特殊字元輸入將成為 SQL 的指令，傳送至後端執行資料庫系統，而對資料庫系統造成意想不到的破壞。

2.1.9 緩衝區溢位

這類攻擊是由於程式設計師當初設計完成後沒有經過完善的測試，導致一些尚未授權的使用者可以利用執行中的程序，來存取記憶體中的資料，而造成一些無法預期的損害。一般來說緩衝區溢位攻擊可以分成三的階段：一、字串長度溢位。二、覆蓋返回位址。三、執行攻擊程式。

一般的電腦程式在都會自動從記憶體中配置一個區塊來儲存或暫時儲存資料，而這個區塊就是所謂的緩衝區。一般緩衝區如果放超過區塊容量的資料時，過大的資料會發生無法預料的結果，通常都會覆蓋到鄰近的區塊。造成的結果會有三種情形：一、程式會以不正常的方式運行。二、程式會產生錯誤而當掉。三、程式仍能繼續進行並沒有顯示出異常狀況。

以上是發動攻擊的第一階段，第二階段一般是插入一小段的 shell 指令在攻擊程式中，來改變執行程式的順序，使其覆蓋返回位址，再寫入攻擊程式碼的記憶體位址，就可以進行第三階段的攻擊程式了。

一般來說單純的緩衝區溢位，並不會對系統的安全造成損害，但如果是系統管理者的權限程式運作發生緩衝區溢位的狀況，攻擊者便可以竊取系統管理者的權限，進一步的控制電腦所造成的損壞是無法估計的。

2.2 資料挖礦

資料挖礦就是將整個資料庫中的資料，利用一種或多種的電腦軟體技術來自動分析、歸納以擷取出知識來的過程。Michael[3]將之定義為：「為了發現有意義的模式或規則，以自動或半自動的方式，來勘查、分析大量資料所進行的流程」。

廣義上來說，資料挖礦簡單的說是一個具有四個步驟的處理過程[5]。其步驟如下:

一、組合所蒐集的資料來分析。

二、將這些資料丟到資料探勘的軟體程式。

三、解釋結果。

四、對新問題或狀況，運用這些結果。

圖2.1整合了上述四個步驟的資料挖礦處理模式的圖示。我們也可以經由這些圖示來描述資料挖礦的每一個步驟。

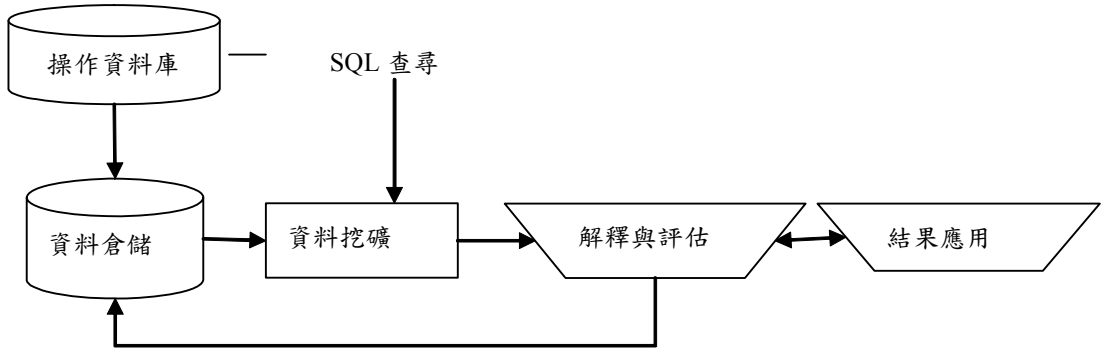

圖2.1　資料挖礦流程[5]

3. 研究方法與架構

3.1 研究架構

本文的研究架構如圖 3.1 所示，我們使用網路上所集得來的測試資料，使用使用基因演算法及自組織映射網路圖(SOM)來從事資料分群，並以部份隨機抽樣資料進行分群正確性測試。

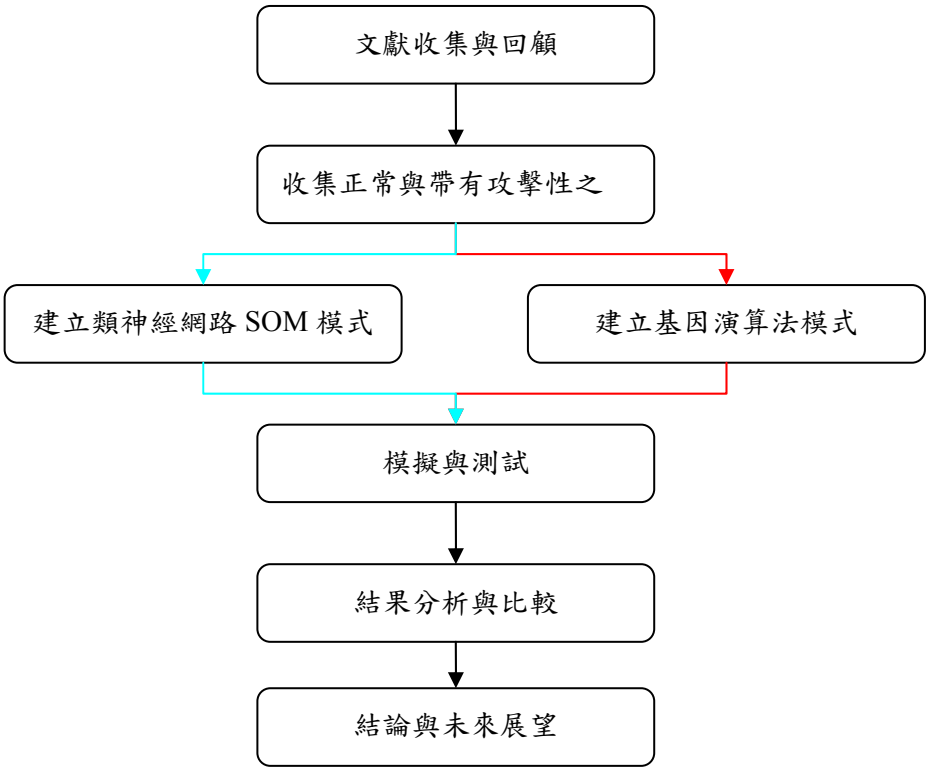

圖 3.1　研究流程步驟

3.2 無監督式基因學習法

基因演算法是應用達爾文的物競天擇原則的進化論的方法到它的學習模組之中。此模組是由 John Holland[5](1986)所發展出來的，另外基因演算法可以用來取代監督式或非監督式的資料探勘，以下呈現一個非監督式基因學習演算法步驟：

一、設定欲分群的數目、並預先推論出 K 個可能的大解(每個大解含分群數目的小解)。

二、再輸入欲分群的訓練範例資料。

三、計算輸入訓練範例與各個小解間的歐基里得距離。

四、每個大解的適合函數等於各個大解中之小解與各範例間取最短歐基里得距離之和。共有 K 個適合函數，並求出最小適合函為最適函數。並隨機執行基因交配與基因突變。

五、依據基因演算法的交配，將非最適函數的解，K-1 個解之間彼此交換內部的點之屬性值。

六、依據基因演算法的突變，將非最適函數的解，K-1 個解中自己解內部彼此交換點之屬性值。

七、反覆執行步驟二到步驟五直到學習狀態穩定為止。

八、得最後求出之最小的最適函數，並依各個測試範例與最適函數解內的點之最短歐

基里得距離來劃分群集。

如圖 3.2 所示為本文使用的非監督式基因演算法流程圖。

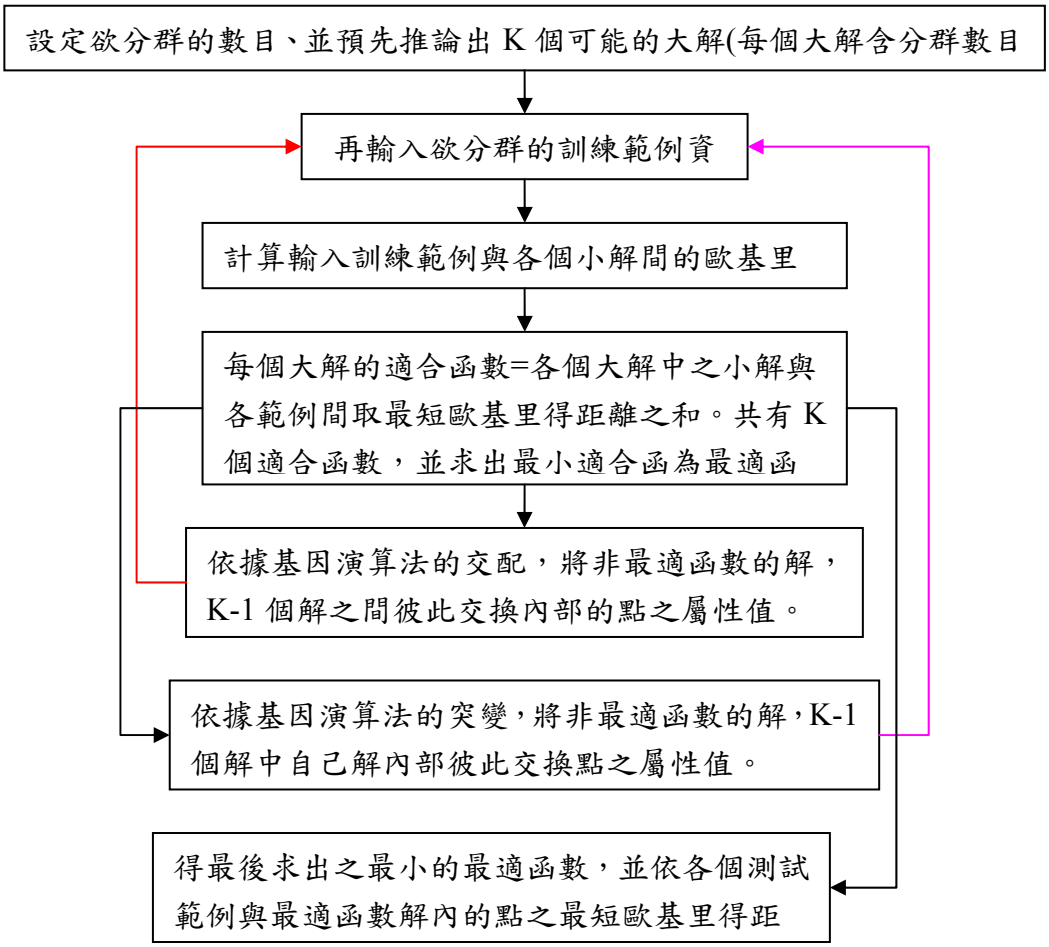

圖 3.2 非監督式基因演算法流程圖

3.3 SOM 模式

自組織映射網路圖(Self-Organizing Map，SOM)是無監督式網路學習的一種，此模式在 1980 年由 T. Kohonen[6]所提出來。依輸入值的特性來區分，無監督式網路學習可以分成二類:(1)輸入值為二元值者(2)輸入值為連續值者。自組織映射網路圖屬於第二類，其基本原理仿大腦結構特性，大腦中功能相似的腦細胞聚集在一起，如人類的大腦中有專司管理視覺、聽覺、味覺、觸覺等的組織區塊，換句話說人類的大腦細胞具有「物以類聚」的特性。自組織映射網路圖就是依據這種特性，其輸出處理單元會相互影響，當網路學習完成後，輸出處理單元相鄰近者會具有相似的功能，也就是說有極高相似度的連結加權值。

SOM 模式在封包分群的符號說明如下：

*J**：輸出層中競爭後的優勝單元，為鄰近中心，此鄰近中心表示其在分群中最具有代表性的資料欄位因子。

*(Cx,Cy)j**：鄰近中心的拓撲座標，即該具代表性資料在二維矩陣中的座標位置。

(Px,Py)j：範例資料在輸出層第 j 個處理單元的拓撲座標。

Rt：範例資料分群時在第 t 次學習循環時的鄰近半徑。

R_rate：在範例資料分群網路設定時，所設定的鄰近半徑縮小因子。

rj：鄰近距離，也就是由鄰近中心的範例資料和輸出層處理單元中第 j 個範例欄位因子的拓撲座標所決定。

αt：範例資料分群網路所設定之學習速率。

α_rate：在範例資料分群網路設定時，所設定的學習速率縮小因子。

R_factor：鄰近係數，為範例資料分群時鄰近半徑和鄰近距離的函數。

SOM 模式的演算步驟如下：

一、 首先設定樣本範例資料的起始權重，此起始權重*Wij*通常由隨機亂數決定，同時決定資料分群的鄰近半徑R^t和學習速率α^t值輸入欲訓練樣本範例資料輸入欄位的因子向量。

二、 計算訓練樣本範例資料輸入欄位的因子向量與輸出層各個神經元的距離，通常以d來表示。$d_i = \sum (X_i - W_{ij})^2$

三、 找出訓練樣本範例資料輸入欄位因子的優勝輸出單元，即其具有最小距離的單元，用來代表該群組的中心位置，以dj表示。$d_j = \min(d_i)$

四、 調整範例資料輸入欄位因子輸入層與輸出層之間的連結權重，以Wij 表示。

五、 以j*為鄰近中心，為範例資料的欄位因子輸出層中競爭後的優勝單元。

六、 $w_{ij} = w_{ij} + \alpha t \times (x_i - w_{ij}) \times R\_factor_j$

七、 其中αt，為樣本範例資料分群網路所設定之學習速率。

八、 $R\_factor_j = f(R, r_j) = \exp(-(r\_from\_win / R))$

九、 r_from_win 為各訓練樣本範例與優勝單元的最小歐基里得距離。

十、 R_factor_j 為第j個訓練樣本範例資料分群時輸出層處理單元的鄰近係數。

十一、 在學習過程中必須將學習速率與鄰近半徑降低，重複步驟2到步驟5的動作，直到網路學習穩定。

十二、 $\alpha t = \alpha t - 1 \bullet \alpha\_rate$

十三、 $Rt = Rt - 1 \bullet R\_rate$

十四、 輸入欲測試樣本範例資料輸入欄位的因子向量。

十五、 計算測試樣本範例資料輸入欄位的因子向量與輸出層各個神經元的距離，通常以di來表示。

十六、 $d_i = \sum (x_i - w_{ij})^2$

十七、 找出所有測試樣本範例資料輸入欄位因子的優勝輸出單元，即其具有最小距離的輸出單元，以dj表示。$d_j = \min(d_i)$

十八、 將所有的優勝輸出單元列於二維拓樸座標，依二維拓樸座標上優勝輸出單元的分佈群聚情形，來進一步分群。

本文所使用的類神經網路SOM模式訓練流程如圖3.3所示，而測試流程則如圖3.4所示。

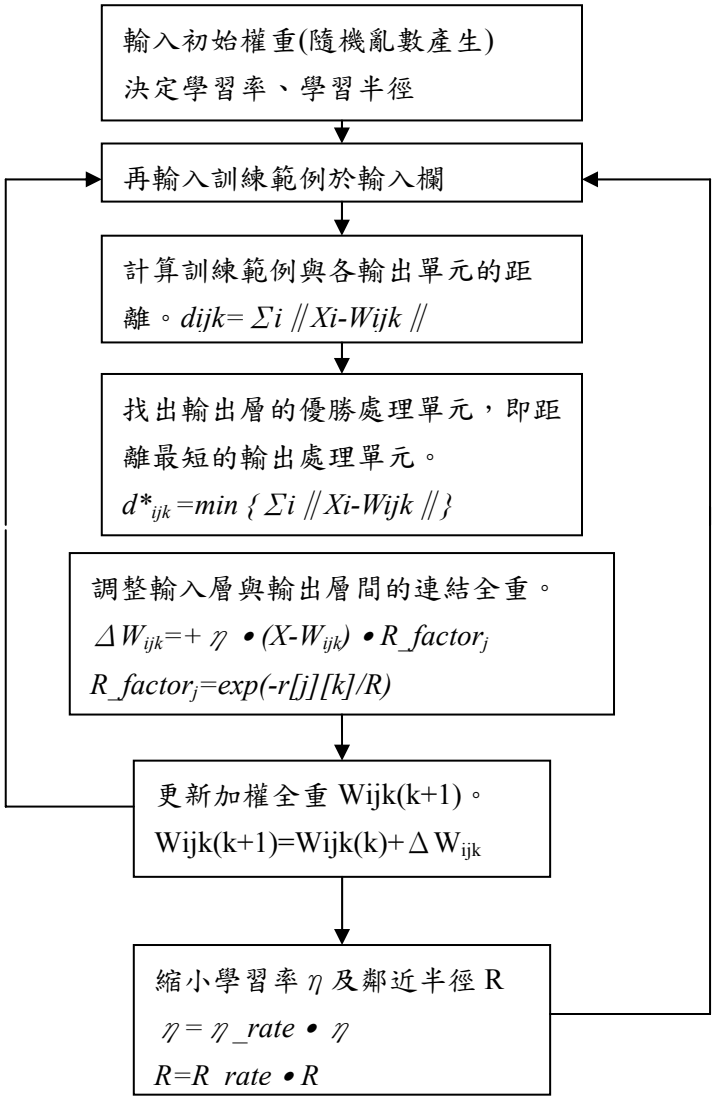

圖3.3 SOM訓練流程圖

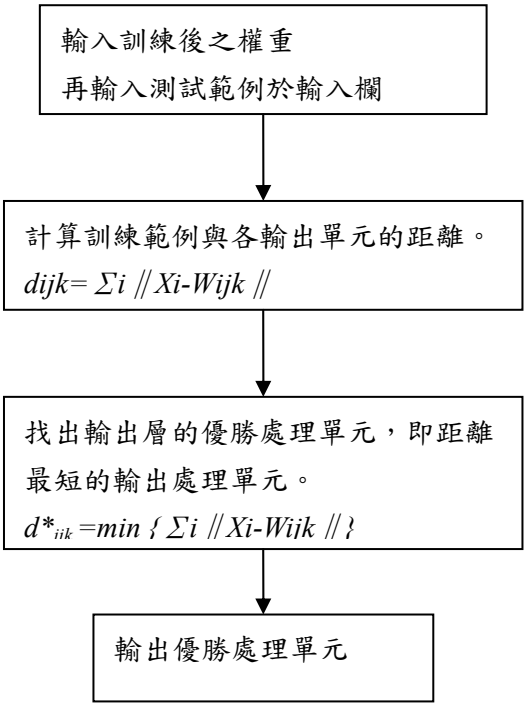

圖 3.4 SOM 測試流程圖

4. 實驗結果

為方便建立偵測模型，本文使用來自於 KDD99cup data 10_percent[9]的資料，從中取出 5000 筆封包資料，其中 2500 筆用於訓練、另外 2500 筆用於測試。而其中測試資料與訓練資料內均含攻擊封包，其分佈狀況如下表 4.1 所示。

表 4.1 測試與訓練資料之資料內容分布

資料類別	Normal 封包	DOS 封包	非 DOS 封包	總共筆數
訓練資料	2251	170	79	2500
測試資料	2249	170	81	2500

首先，先將這 5000 筆資料先標準化後，訓練資料輸入於類神經自組織映射圖網路程式中，程式執行 200 次循環藉此來修正相關權重參數，使其最小學習率達到 0.1 且最小修正半徑達 0.25，然後再將測試資料輸入。實驗所得的 GA 與 SOM 分群測試結果如表 4.2 所示。

表 4.2 GA 與 SOM 分群結果正確率。

	正常	DOS	非 DOS	其他
GA	100%	56.38%	1.97%	0%
SOM	98.48%	35.9%	10.4%	4.62%

從以上的分群結果可知，封包入侵偵測系統如果可以預先對進入系統之封包給予分群，由 GA 和 SOM 分群結果可知，由 GA 模式分群後所得的第一群集均為正常封包，第二群集為 DOS 攻擊模式封包，這兩群準確率均大於 SOM 模式；後兩類分群準去率卻小於 SOM 模式。則我們可利用 GA 模式分群後，當中的第一群集均為正常封包封包偵測系統可以不需任何比對封包攻擊特徵；第二群集優先比對封包偵測資料庫的 DOS 攻擊模式特徵值，而 SOM 模式的第三群集則優先比對封包偵測資料庫的非 DOS 攻擊模式特徵值，如此將可以有效的降低封包偵測系統的比對時間；而 SOM 模式的第四群集則為可能含有未知攻擊模式的封包，封包偵測系統應拒絕給予進入。

5. 結論

對於網路安全系統，不論是提供被動保護的防火牆或是主動偵測的入侵偵測系統，皆為條列式比對偵測，只能對已經知道的攻擊模式或封包進行攔阻。本文應用類神經自組織映射圖網路和基因演算法之資料探勘方法，以網路搜集正常封包及帶有攻擊性之封包作為分群輸入因子，透過上列分群的方法分析各封包群組，以提供校園及業者設計出更具精確性與效率的入侵偵測系統，創造出網路更安全的世界。

參考文獻

1. 洪嘉鴻，「UDIDT 下建構於安全機制之入侵追蹤系統」，東海大學資訊工程與科學研究所碩士論文（2003）。
2. 鄭真真，「UDIDT 下之多階段入侵偵測系統」，東海大學資訊工程與科學研究所碩士論文（2003）。
3. 劉世琪，「應用資料挖掘探討顧客價值-以汽車維修業為例」，朝陽科技大學工業工程與管理系研究所碩士論文（2003）。

4. 葉怡成，「類神經網路模式應用與實作」，儒林圖書公司，台北(2003)。
5. 曾新穆、李建億譯, Richard J. Roiger and Michael W. Geatz 著, Data Mining, 東華書局，台北(2003)。
6. Teuvo Kohonen, “The Self-Organizing Map,” *Proceedings of the IEEE*, 78(9), 1464-1480 (1990).
7. Yongguo Liu, Kefei Chen, Xiaofeng Liao and Wei Zhang, “A Genetic Clustering Method for Intrusion,” *Pattern Recognition*, 37, 927-942 (2004).
8. Randall S. Sexton and Naheel A. Sikander, “Data Mining Using a Genetic Algorithm-Trained Neural Network,” *International Journal of Intelligent Systems in Accounting, Finance & Management* , 10, 201-210 (2001).
9. KDD99cup dataset, http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html