國 立 交 通 大 學

管理學院（資訊管理學程）碩士班

碩　士　論　文

分散式阻斷攻擊檢測和防禦機制之探討與實作

Research on Detection / Prevention of DDoS and Its Implementation

研 究 生：周佳樟
指導教授：羅濟群　博士

中　華　民　國　九　十　八　年　七　月

分散式阻斷攻擊檢測和防禦機制之探討與實作

Research on Detection / Prevention of DDoS and Its Implementation

研 究 生：周佳樟　　Student：Chia-Chang Chou
指導教授：羅濟群 博士　　Advisor：Dr.Chi-Chun Lo

國 立 交 通 大 學

管理學院（資訊管理學程）碩士班

碩 士 論 文

A Thesis

Submitted to Institute of Information Management

College of Management

National Chiao Tung University

In Partial Fulfillment of the Requirements

For the Degree of

Master of Science

in

Information Management

July 2009

Hsinchu， Taiwan， the Republic of China

中　華　民　國　九　十　八　年　七　月

研究生：周佳樟　　指導教授：羅濟群 博士

國立交通大學管理學院（資訊管理學程）碩士班

摘要

DDoS 攻擊已經給網際網路帶來了嚴重的破壞，並且日益成為當今網際網路安全的嚴峻威脅之一。現有對 DDoS 的防禦系統的研究仍然存在著各種不足，它們或者未能針對 DDoS 攻擊的特點進行檢測，或者在檢測後未能對攻擊流即時有效的控制，或者在實際應用中難以展開部署。為此，針對 DDoS 攻擊所具有的異于正常流量的多位元組統計特徵，本文設計實現了一種基於網路處理器 IXP2400 的即時檢測及控制系統模組。該系統通過對 IP 網路流量進行多位元組統計異常性分析來發現網路中存在異常的位元元組流量，利用權杖桶技術對異常位元組段流量進行控制。經過一系列的性能評估實驗驗證，本系統可以在充分保護正常流量的基礎上，對 DDoS 作出有效的檢測及控制，使得網路在遭受 DDoS 攻擊的時候仍然可以提供最大限度的正常服務。本系統具有高性能處理能力，適合於部署在網際網路分佈層的關鍵出入口對網路安全進行有力的維護。

關鍵字：DDoS 攻擊，網路處理器 IXP2400，異常檢測，流量控制

Research on Detection / Prevention of DDoS and Its Implementation

Student：Chia-chang Chou　　Advisor：Dr. Chi-Chun Lo

Institue of Information Management
National Chiao Tung University
Hsinchu， Taiwan， Republic of China

Abstract

The DDoS attack has ripped the Internet seriously and remains a severe threat to the Internet， but the defense systems so far developed still have difficulties to cope with it. In this paper， we present a novel system module based on IXP2400 to fight against DDoS attack. The system carries out multi-dimensional real-time anomaly detection to analyze the statistics of traffics for each field， detects the abnormal traffic and uses Token Bucket Filter to control the abnormal traffic. Results of a series of experiments demonstrate that the legitimate traffic can go through the system remaining intact while the harmful DDoS attack get detected by the system's anomaly-detection mechanism and the attack traffics go under effective control by the system's flow control mechanism. The system developed is fit to be deployed at the edge of network's aggregate layer to maintain network security.

Key Words: DDoS Attack， Network Processor IXP2400， Anomaly Detection， Flow Control

誌 謝

感謝我的指導教授羅濟群博士，在羅教授的悉心指導下，讓我能對各個相關領域有了更深的認知。

特別要感謝的是，資訊業界的朋友提供的軟體和硬體設備，其次是交大資管所裡的老師和同學們，他們的指導與鼓勵，讓我能在各方面有所成長。

最後要感謝我的爸爸,媽媽,還我的太太鄭羽君因為他們的全力支持，我才得以安心的完成課業。

目錄

圖目錄

表目錄

符 號 說 明

A_n：統計平均值

AR_n ：分組到達率為

A_{n-1}：前 n-1 個統計間隔內的到達率的平均值為

DIF_n'：偏離變化

σ ：到達率的標準差

σ_{n-1}：到達率的滑動標準差為

A^K ：達率平均值為第 n 個統計間隔的異常程度

TI ：時脈速率

$TR'_m = {}^{TR_m} / 16$ ：計算硬體計時器的定時長度

第一章 緒論

1.1 研究動機

DDoS[1](Distributed Denial-of-Service)攻擊是網際網路目前面臨的最嚴峻的威脅之一，也是近幾年網路安全研究的熱門問題。不少國內外著名的網路服務商以及政治金融機構都曾經受到其侵襲：諸如波及之網站首當其衝為知名入口網站 Yahoo!（Yahoo.com），導致網站登錄管道一度癱瘓，令世界各地的網路使用者幾乎不得其門而入。緊接著遭受攻擊的商業網站，包括 Buy.com、拍賣網站電子灣（eBay）、購物網站亞馬遜(Amazon.com)、新聞網站美國有線電視新聞網(CNN.com）以及 E-trade 等，無法倖免而宣告癱瘓。而致使其中一些網站中斷服務長達數小時甚至幾天之久。對 DDoS 攻擊的檢測和防禦是迫在眉睫的問題。

由於其種種特點，使得 DDoS 攻擊較一般網路入侵更難以對付。由於這種攻擊以網路中常見的協定和服務作為載體，與正常網路資料流程相比，其引起的網路流量在表面上只是具有數量特別巨大這一特點，而並沒有類似其他網路入侵所具有的一些內容特徵，是以當攻擊流與正常流混雜在一起時，往往難以從中將攻擊流分離出來。另外，DDoS 攻擊一般採用三層控制體系，攻擊時通常使用 IP 欺騙技術，使攻擊的根源發動者極易隱藏自己，躲避追蹤，同時也導致攻擊難以被根除。再有，現在網路上仍然存在很多安全性薄弱的系統，有利於攻擊者控制大量攻擊幫兇來實施自己的攻擊計畫，致使現實中往往難以預防攻擊的發生。

面對 DDoS 攻擊，雖然可以選擇提高網路頻寬和增強網路設備性能來抗衡，但頻寬容量不可能無限增長，而攻擊的規模則可以容易地通過控制更多的僵屍主機（zombie)（實際執行攻擊的機器）來大大加

強，而且隨著僵屍主機的性能也在不斷增長，攻擊的強度也與日俱增。故此這種消極的應對方法並不值得提倡。

為此不少工作人員在這一安全領域進行大量的研究，研究並總結出一些防禦措施。這些方法根據其所處的防禦階段可以分為攻擊發生前後的預防和追蹤以及攻擊時的控制。由於網際網路管理的分散性，攻擊前的預防通常不易實施。而對攻擊機器的追蹤雖然在某種程度上可以制止攻擊的持續進行，但由於這種方法必須要求對網際網路中眾多網路設備進行變動設定，以使它們可以連動工作才可能取得預期的效果，所以現實中並不十分可行；於是人們紛紛將重點放在遭受攻擊過程中對攻擊的檢測與控制上。

而一個簡易可行的檢測與防禦辦法，就是在網路中一些關鍵的出口或入口處設定一個防禦監控設備，這個設備有能力在攻擊到來時及時檢測出攻擊發生，並採取有效的控制措施以達到資訊網路系統的正常服務。

1.2 研究方法

針對現有 DDoS 攻擊檢測或控制系統的不足，在充分研究 DDoS 攻擊流特性的基礎上，本文設計了一種基於網路處理器 IXP2400，具有即時工作功能的系統模組。本系統採用多位元組異常檢測及監控方法，對異常網路流量尤其是 DDoS 攻擊流進行限制，以保證網路的正常通訊。

系統模組設計能夠對流經本系統的網路資料流程的多元位統計資料作出異常性檢測分析，對發現異常的位元組流按照其異常程度指數的高低進行多個等級別的權杖桶(Token Bucket Algorithm)限速控制，以最大限度保證網路安全。實驗結果表明，該系統在基本不影響正常網路通信的基礎上對 DDoS 攻擊有良好的抵禦作用，可作為有力的防禦產品部署在 IP 網路中的邊緣出入口或骨幹網的關鍵出入口，並可作為保

護網站伺服器的一個防禦系統。

1.3 章節介紹

本文第二章將就現行的 DDoS 攻擊進行探討，對於文獻及資訊安全的防禦議題進行研究探討；第三章是系統設計的理論和演算法；第四章提出實作系統構並依其架構發展及系統評估與測試，第五章則是總結全文，並對實作系統提出兩項可供改進的方法。

第二章文獻探討

本章從入侵檢測的相關概念討論，分析 DDoS 攻擊的原理，詳細介紹它的分類和特點；接著介紹目前業界對 DDoS 攻擊以及相應的檢測和防禦技術的研究現狀，為本文的設計和實作提供理論和實踐的支撐。

2.1 入侵檢測概述

1980 年， James P.Anderson 系統闡述了入侵檢測（Intrusion Detection）的概念。Anderson 將入侵定義為潛在的、有預謀的、未經授權的訪問資訊、操作資訊、致使系統不可靠或無法使用的企圖。入侵檢測是對入侵行為的檢測。進行入侵檢測的軟硬體系統稱為入侵檢測系統（簡稱為 IDS）。入侵檢測系統有別於傳統的防火牆，它作為後者的補充為系統提供第二道安全防護，在不影響正常網路性能的基礎上對網路進行監控。

根據目標系統的不同入侵檢測系統分為基於伺服器(Host Base)和基於網路(Network Base)的入侵檢測系統兩大類。前者以單一伺服器的記錄作為依據，後者則以網路封包作為資料來源。根據入侵檢測方法的不同，入侵檢測系統分為基於特徵匹配（signature-based）和基於異常（anomaly-based）檢測兩種。特徵匹配方法根據已知的入侵攻擊資訊來檢測系統中的入侵和攻擊，異常檢測則利用系統正常行為的資訊作為檢測系統判斷的依據。

DDoS 攻擊屬於眾多入侵攻擊中的其中一種，由於其大規模、分散式和複雜的攻擊手段的特點，DDoS 攻擊破壞力強、防禦難度大，逐漸成為入侵攻擊的主流。

2.2 DDoS 攻擊原理

分散式阻斷服務（DDoS）攻擊的前身是所謂的『阻斷服務（Denial-of-Service，簡稱 DoS）攻擊』。DoS 攻擊並不以篡改或竊取主機資料為目的，而是癱瘓系統主機使之無法正常運作。換言之，由於一般網路系統的系統資源（例如記憶體、磁碟空間以及網路頻寬等）皆有限，因此駭客可以根據部分網路系統或者相關通信協定等之設計或實作上的漏洞，在一段期間內透過傳送大量且密集的封包至特定網站，使該網站無法立即處理這些封包而導致癱瘓，進而造成網路用戶無法連上該網站而被阻絕在外。這種攻擊對網站本身而言，並不具破壞性，只是造成系統無法即時處理駭客所送來的大量訊息而停滯或當機。DDoS 攻擊之是以能無孔不入、危害四方，主要是由於目前網際網路普遍採用的 TCP/IP[1]協定存在著缺陷所造成的。可以說，基於 TCP/IP 協定的分散式阻斷服務攻擊構成了目前 DDoS 攻擊的主體。

2.2.1 DDoS 攻擊拓撲[4][5]

圖 2-1 是一個典型的 DDoS 攻擊拓撲模型圖。

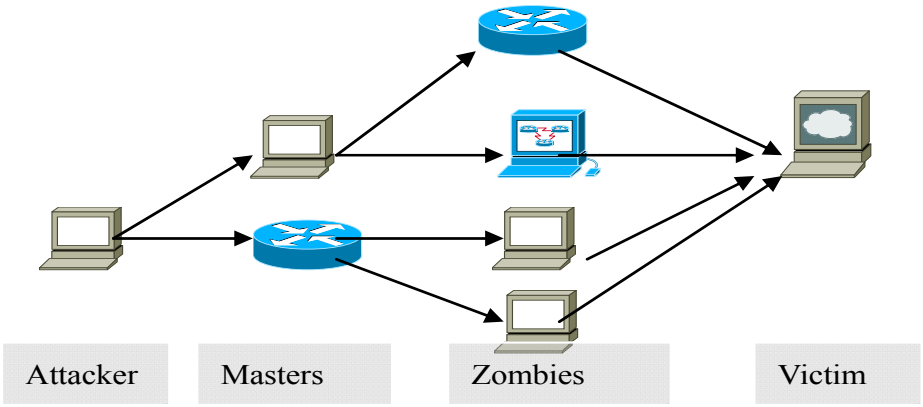

圖 2-1 DDoS 攻擊拓撲模型

入侵和攻擊主要過程如下：攻擊者（Attacker）通過掃描從網路中找出存在漏洞的伺服器或網路系統作為入侵控制的物件主控設備（Masters），通過控制主控設備重複掃描、入侵的步驟，攻擊者掌握了為數眾多的攻擊執行設備（Zombies）。需要發動攻擊時，攻擊者只需操縱主控設備置入攻擊指令和代碼於攻擊執行設備，後者即可對受害者發起 DDoS 攻擊。

從以上分析可以看出，主要是兩個方面的原因成就了 DDoS 的巨大威力。首先，真正發起 DDoS 攻擊的幕後黑手往往不正面接觸攻擊的受害者，甚至一旦完成攻擊指令的下達後，它就可以逃之夭夭了；即使受害者通過各種手段和途徑對攻擊者進行追蹤，能找到的也只是那些同為受害者的僵屍主機，而無法達到根治的目的。其次， DDoS 攻擊是分散式阻斷服務攻擊。所謂分散式，指的是其攻擊發動方無論在物理空間上還是在邏輯空間上來說，都是分散的，攻擊實施設備的數量、種類、性能等參數也是隨機的，這些分佈特性，進一步加大了對其進行防禦和檢測的難度。

2.3 DDoS 攻擊技術[2]

根據標準的不同，DDoS 攻擊可以有不同的分類方法，下面介紹的是其中最常用的一種。

2.3.1 攻擊分類

DDoS 攻擊的目的是消耗目標系統資源或者網路頻寬。按照攻擊採用手段的不同可以分為邏輯攻擊和泛洪攻擊兩大類[12]。

1. 邏輯攻擊

邏輯攻擊是指攻擊者 利用系統所運行的協定或者軟體的邏輯漏

洞對受害者進行的 DDoS 攻擊。如 Ping of death 攻擊，就是利用向系統發送超過 64KB 這一最大的 ICMP 請求封包、導致系統為之分配記憶體時出錯而耗盡資源。

例如:

（1）TearDrop 攻擊：利用 IP 封包重組的漏洞來進行攻擊。當資料要經由網路傳送時，其 IP 封包常被切割成許多小片段；每個小片段和原來封包的結構除了某些記載位移的資訊不同外，其餘大致都相同，其中這些位移資訊是要使網路主機在收到這些小片段時能夠正確地重組 IP 封包。TearDrop 攻擊則憑空創造出一些 IP 片段，但這些片段封包含了重疊的位移值。當這些片段被傳送到達目的地時，會重組成原來的 IP 封包，此時可能會造成系統當機。

(2) LAND 攻擊：運用 IP Spoofing 技術送出一連串 SYN 封包給目標主機，讓目標主機系統誤以為這些封包是由自己發送的。由於目標主機在處理這些封包的時候，它自己並無法回應給自己 SYN-ACK 封包，因而造成系統當機。

2. 泛洪攻擊

泛洪攻擊是當攻擊者對受害者系統發送大量偽造的封包、讓其在反應中消耗盡系統 CPU、存儲等資源或者所在系統的網路頻寬的 DDoS 攻擊。DDoS 攻擊，其他相似的還包括 TCP ACK、 TCP RST、 UDP 泛洪和 ICMP 泛洪（Ping flooding）等。事實上，所有基於 IP 協定的上層協定均可被利用來進行泛洪攻擊。泛洪攻擊種類繁多，構成了當前網際網路上 DDoS 攻擊的主流，下面是其中的幾種。

(1)TCP SYN 攻擊:TCP 建立連接所需三次握手階段的漏洞為例，說明攻擊者利用 TCP/IP 協定的缺陷採取的 DDoS 攻擊方法原理。

TCP 握手協定[3]

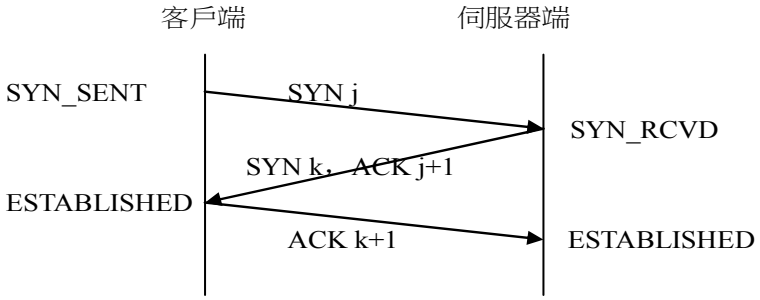

圖 2-2 TCP 三次握手過程

在圖 2-1 中，在 TCP/IP 協定中，TCP 協定提供可靠的連接服務，採用三次握手建立一個連接。 第一次握手發生在建立連接時，客戶埠發送 SYN 封包（SYN=j）到伺服器，並進入 SYN_SEND 狀態，等待伺服器確認。

第二次握手發生在伺服器收到 SYN 封包後，對客戶的 SYN 連接請求進行確認（ACK=j+1），同時自己發送一個 SYN 標記（SYN=k）即 SYN+ACK 封包，此時伺服器進入 SYN_RCVD 狀態。

第三次握手是客戶埠收到伺服器的 SYN＋ACK 封包後，向伺服器發送確認封包 ACK(ACK=k+1)，這樣，客戶埠和伺服器進入 ESTABLISHED 狀態，完成三次握手，客戶埠和伺服器埠的通信連接完成，可以開始傳輸資料了。

在圖 2.1 中可看到，伺服器接收到連接請求（SYN=j），將此資訊加入未連接佇列，並發送請求封包給客戶（SYN =k， ACK=j+1），此時進入 SYN_RCVD 狀態。當伺服器未收到客戶埠的確認封包時，預設會重發請求封包，一直到超時，才將此條目從未連接佇列刪除。可見，如果客戶埠在短時間內偽造大量不存在的 IP 位址，向伺服器不斷地發送 SYN 封包，伺服器照常回復確認封包，並等待客戶的確認，而由於來源位址是不存在的，不會有客戶埠回復伺服器的這些確認封包，於是伺服器就不得不保存越來越多的這種處於半打開狀態的連接，最後伺服器的記憶體和 CPU 時間就會被這種等待的佇列占滿，而合法使用者則無法與伺服器建立連接，從而達到阻斷服務攻擊目的。

(3) ARP 攻擊：ARP 位址解析協定 (Address Resolution Protocol) 是用來將 IP 位址解析成局網域網路硬體所使用的媒體存取控制位址(MAC)，這是一個 48 位元的乙太網位址。這兩種位址間的某種靜態或演算法的映射通常被 ARP 表或路由器所存儲。當電腦接收到 ARP 應答封包時，會對本地的 ARP 暫存器進行更新，將封包中的 IP 和 MAC 位址對存儲在 ARP 暫存器中，這一特點直接造成了 ARP 欺騙攻擊實施的可能性。直接向 Windows 系統伺服器發送大量無關的 ARP 封包，也會導致系統耗盡資源而停止回應，如果向廣播位址發送 ARP 請求，可能導致整個局網域網路停止回應。

(4) UDP 攻擊：UDP 使用者通訊協定 (User Datagram Protocol)是基於無連接的傳輸層協定,它不提供錯誤更正和重發也不檢查封包的丟失或重複,適用於僅需要詢問/回應的應用中，所需的開銷和耗費較少。但由於它沒有流量控制和差錯修正的機制，是以容易造成資料遺失或錯誤。與 TCP 攻擊手段一樣， UDP Flood 也很常見。

(5) ICMP 攻擊：ICMP 網際網路控制協定 (Internet Control Message Protocol)，它是用來監控 TCP/ UDP 層連接狀態的底層機制，通常用來報告路徑中出現的問題，監測網路中的故障，同時也提供了一些底層網路診斷工具，如常見的 Ping 命令。基於 ICMP 最典型的攻擊就是由 Ping 命令演化成的 Smurf 攻擊，另外有 Ping to Death 等、ICMP 重定向和 ICMP 目標不可到達攻擊等。

2.3.2 攻擊工具[6][7][8][9]

DDoS 攻擊工具主要有 Trin00、TFN、TFN2K、Stacheldracht 等。

Trin00 是一個較早的 DDOS 攻擊工具。Trin00 只能進行 UDP FLOOD 攻擊。Trin00 的主控埠位址必須手動加入到攻擊埠的來源程式中再進行編譯。

TFN 是一個可以進行多種攻擊的 DDOS 工具。它可以進行 UDP FLOOD，TCP SYN FLOOD，ICMP FLOOD 以及 SMURF 攻擊。

TFN2K 是 TFN 的後續版本。與 TFN 相比，TFN2K 具有更多的功能和更大的靈活性，包括可對目標的進行包括 TCP SYN FLOOD、UDP FLOOD 、ICMP/PING FLOOD 或 BROADCAST /PING (SMURF)FLOOD 等的攻擊。

Stacheldracht 是一種與 TFN 極為相似的工具。stacheldraht 也使用了與 TFN 攻擊工具一樣的阻斷服務攻擊方法，如：ICMP flood、SYN flood、UDP flood 和"Smurf"等。

2.4 DDoS 檢測和防禦研究狀況[10]

現今之 DDoS 攻擊對策可概分如下

對 DDoS 攻擊的防禦和檢測可分為兩個層次：

1.攻擊檢測和控制：在 DDoS 攻擊發生的時候迅速準確的檢測和控制，可減少攻擊危害。

2.攻擊源追蹤。在攻擊發生的同時或攻擊結束之後追蹤和標識攻擊源，讓攻擊者得到懲治，則是從源頭上消滅攻擊。

2.4.1 攻擊檢測和控制

對 DDoS 的檢測和控制可在攻擊源和攻擊目標之間的任何位置進行。越靠近攻擊源所在的網路，攻擊分組就越容易被檢測出來；反之，越靠近攻擊目標所在的網路，攻擊封包被檢測出來的難度就越大。但是，雖然在靠近攻擊目標的地方實施檢測和控制的效果最好，這也會造成攻擊分組氾濫在中間網路、佔用網路頻寬，從而造成中間網路的

擁塞；在靠近攻擊源的地方進行檢測和控制雖然效果不好，卻是消除攻擊影響的最根本辦法。

1.來源埠防禦

J. Mirkovic 等中提出了一種稱為 D-WARD 的來源埠檢測和控制 DDoS 的技術。D-WARD 將系統安裝在來源所在網路的邊界路由器上，對輸出流量進行監控和統計並根據目的地封包進行分類，通過跟內置的正常流量模型對比，定義正常流量和潛在攻擊流量，在讓正常流順利通過的同時，限制攻擊流量的速率，並隨著攻擊速率的增加幅度降低攻擊封包的速率。

D-WARD 面臨的問題在於只統計了封包的部分資訊，在大規模分散式攻擊中，來自每個網路的攻擊封包所占的分量往往是很小的，有時會小到無法從簡單的資料統計看出異常來；與所有來源埠防禦技術一樣，部署這樣一個系統的網路往往無法從其投入的資源得到直接可見的收益，這就潛在地限制了基於此類方法系統的推廣和使用。

2.過濾技術

過濾技術對進入網路的所有流量進行過濾。當攻擊者通過偽造的 IP 位址進行 DDoS 攻擊時，受害者不得不往虛假的 IP 位址發送大量的 SYN-ACK 封包卻永遠也得不到應有的 ACK 回應。P. Ferguson 和 D. Senie 在 RFC 2827 中描述了一種通過對輸入流量來源 IP 位址進行過濾、防止利用偽造來源 IP 位址進行 DDoS 攻擊的方法。K. Park and H. Lee 提出了一種基於路由的過濾技術 ，通過引入封包篩檢程式，對骨幹網路封包的來源和目的 IP 位元址進行檢查，根據 BGP 路由資訊分析判斷是否屬於來自一條正常的鏈路。 Chen Jin 等人則將 IP 封包的 TTL 位元組跟其來源 IP 位址進行綁定，異於綁定資料庫資訊的流量被認為是潛在的 DDoS 攻擊一種稱為基於跳數的過濾技術。

過濾技術也面臨著困難，其實施節點的選擇對過濾效果的影響有著很大的影響，例如：在靠近來源的網路和靠近目的的網路所需要的尺度就有著明顯區別；而對於骨幹節點設備來說，全力轉發高速資料

流程是最主要的功能，在這裡實施封包過濾將產生網路擁塞等所有 ISP 不願意看到的副作用。

2.4.2 攻擊源追蹤

攻擊源追蹤也稱為「IP Traceback」， 其功能是在 DDoS 攻擊發生時或者結束後確定攻擊者的攻擊路徑和攻擊發起源。

常見的 IP Traceback 方法包括：基於雜湊的 IP Traceback、ICMP Traceback、概率分組標記法、確定分組標記法等。

1. 基於雜湊的 IP Traceback：首先將網路在邏輯上分為不同的地方，用於重構攻擊路徑；其次為經過路由器的每個分組都保存一份摘要，摘要的雜湊輸入通常為分組的 IP 頭部和負載的部分資訊，經過雜湊運算得到的這個摘要就代表了分組的資訊。當系統從攻擊受害者的入侵檢測系統得到受攻擊通知後，從資料庫中查找出哪些路由器參與了轉發攻擊包，結合分組摘要資料庫資訊重構出本網路中詳細的攻擊路徑。

2. ICMP Traceback:，按照統計方法從經過路由器的分組中選擇出一些並向其目的 IP 位元址發送特殊的 ICMP 封包，此 ICMP 封包，包含相鄰路由器的資訊以及時間戳，封包中的 TTL 位元組設定為 255 以便封包能到達足夠遠的攻擊路徑。假設此封包的目的 IP 設備正遭受 DDoS 攻擊，根據流向它的巨大的攻擊流量，受害者最終能獲得攻擊路徑上所有的路由資訊，經過對 TTL 值的分析，將很容易重構出攻擊路徑。該方法的缺點在於額外的 ICMP 分組增加了網路的流量，加重了網路負擔，並且攻擊者可能發送假的 ICMP Traceback 封包欺騙目標伺服器。

3.分組標記的思想是路由器對封包某個位元組元根據需要進行標識再行路由轉發。位元組的選擇是分組標記技術首要解決的問題。雖然 IPv4 的 ID 位元組和 Option 位元組均可用來標記，但是使用前者將

會影響網際網路中碎片封包的重組（儘管需分片的流量極少，僅占總流量的 0.5%），後者則會改變原有封包的長度，造成碎片的發生。概率分組標記的思想是隨機地將到達路由器的分組按照某個概率選擇出來進行標記，受害者收集到足夠多的標記封包就能將攻擊路徑重建出來。但是這種方法一旦被攻擊者獲悉，後者故意發送封包含錯誤資訊的分組就能夠逃避標記，從而失去其效用。確定分組標記部署在所有的邊緣路由器上，主要方法是將路由器靠近來源的埠的 IP 位址分成各 16 位元的兩半，其中的任何一半存儲在 IP 封包的 ID 網域，RF 標誌位元則指明存儲的具體是哪一半。為了防止攻擊者採用偽造大量隨機來源 IP 位址造成的計算消耗，對來源 IP 位址進行雜湊摘要運算後再行標記。然而無論採取何種措施，這種分組標記都會大大加重路由器和被攻擊者的負擔。

2.5 現有防禦和檢測技術的不足[11][12][13]

從上面對目前 DDoS 防禦和檢測技術的介紹可見，無論是在對攻擊的檢測和控制還是對攻擊源的追蹤技術面對當前的 DDoS 攻擊均存在各自的缺陷和不足，這也是在這個領網域的研究一直方興未艾的一個主要動因所在。概括地說，這些缺陷和不足主要包括兩個方面：

1.是攻擊手段的多樣性和網路拓撲的複雜性所致。受研究條件限制，這些技術和方法只是對攻擊的一個或多個局部方面進行研究，其系統的適應性必然有限。

2.是每個研究者都面臨資源問題，當前大規模分散式攻擊已經成為 DDoS 攻擊的主流，傳統的路由交換設備既要完成封包的轉發處理這個基本任務，還要分配資源用於攻擊檢測和防禦，即使研究者能設想出一個完美的理論，往往也受限於設備的可用資源而不得實施。

2.6 網路處理器和 DDoS[14]

網路處理器從誕生起就倍受各方、特別是網路安全領網域研究人員的青睞。這都得歸功於網路處理器的突出優點：高度可編程和高速處理能力。

麻州大學(University of Massachusetts –Amherst)的 R. Ramaswamy 等人將 IXP2400 網路處理器用於被動網路測量系統的研究，致力於建立一個用於下一代千兆網路的分散式、高效的網路測量系統。IXP2400 的引入很好地解決了系統設計中的封包分解、IP 位址前置匿名化和線上資料統計等三個棘手的問題。

NetBouncer 是 R. Thomas 等人，先後在 IXP1200 和 IXP2400 網路處理器上實現的一個 DDoS 攻擊防禦系統。NetBouncer 維護一個保存經過確認為合法使用者的資料庫並為其分配一個合法期限，當接收到不屬合法使用者之列的封包時,系統將對其進行一系列的合法性測試。如果通過測試，此使用者將被加入合法使用者資料庫，直到其合法期限到期；否則不提供服務。在合法期限內，所有來自合法使用者的資料將由一個流量管理子系統進行統一的頻寬分配和速率限制後進行轉發。

清華大學的 RONG-TAI LIU 等人提出了一個稱為「快速字元串匹配」的特徵演算法，結合利用流行的開源特徵檢測系統 Snort 中的特徵模組，在 Vitesse IQ2000 網路處理器上實現了一個基於特徵的 NIDS，獲得了性能上的顯著提高。

其他方面的研究還包括：T. Verdickt 等人，把網路處理器應用到防火牆開發中 ， 交通大學 Yi-Neng Lin 等在網路處理器上則設計了一個 VPN 閘道。結合研究思想和研究方法來看，以 BoonPing Lim、Md. Safi 和 Uddin 在網路處理器上實現的一個 SYN flooding 檢測系統跟本文的設計最為相近。該系統實現了一個稱為 SYNmon 的嵌入式原型結構，

通過對 TCP 互動協商過程中 SYN 和 ACK 封包的監控實現了一個基於流的攻擊檢測系統。

2.7 異常統計的相關探討

2.7.1 異常值的統計準則

常用的異常值的統計判別方法有 3σ 準則 (RaiiTa criterion)、格拉布斯準則(Grubbs criterion)、狄克遜準則(Dixon criterion)等

1. 3σ 準則

3σ 準則是一種最常用、最簡單的異常資料判定準則。其演算法簡捷，不需查表，可以應用於試驗檢測次數較多的場合。3σ 準則認為當試驗次數較多（>10）時，可簡單地用 3 倍標準偏差作為確定可疑異常資料的標準。由於該方法是以 3 倍標準偏差作為判別標準，所以亦稱 3 倍標準偏差法，簡稱 3σ 法。

一般情況下，對於一組樣本資料，如果樣本資料中存在隨機誤差，則根據隨機誤差的正態分佈規律，其偏差落在±3σ 以內的概率為 99.7%，出現在該範圍之外的概率僅為 0.3%。即：

$$P(|x-\mu|>3\sigma)\le 0.003 \qquad (1)$$

上式中，μ 與 σ 分別表示正態總體的數學期望和標準差。即：對於資料 x_1，x_2，……x_n，其均值 $\bar{x}=\frac{1}{n}\sum_{i=1}^{n}x_i$，其殘差為 $v_i = x_i - \bar{x}$，(i=1，2，3，…n)，則標準差為：$\sigma=\sqrt{\frac{1}{(n-1)}\sum_i v_i^{\,2}}=\sqrt{\frac{1}{(n-1)}\left(\sum_i (x_i-\bar{x})^2\right)}$

（式 1）表明，在實驗資料中出現大於 μ＋3σ 或小於 μ—3σ 資料的概率是很小的。是以，根據上式對於大於 μ＋3σ 或小於 μ—3σ，即偏差大於 3σ 的資料，則可以認為它是異常資料，而且偏離越大的資料其異常程度也越大。另外，當測量值與平均值之差大於 2 倍標準偏差時，則該測量值應保留，但也屬於可疑資料。

2. 格拉布斯準則(Grubbs criterion)

格拉布斯準則是在未知總體標準差情況下，對正態樣本或接近正態樣本異常值的一種判別方法。對於置信度為 p%的置信區間，可以查表得到其格拉布斯係數 k，則對於資料 x_i 如果滿足：

$$\left|x_i - \bar{x}\right| > kv \quad （其中 \quad \bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i \quad , \quad v = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(x_i - \bar{x}\right)^2} \quad ） \qquad （2）$$

則成為在置信度為 p%下的異常值。

7. 狄克遜準則 (Dixon criterion)

狄克遜準則認為對於服從正態分佈的資料 $X_1<=X_2<=\ldots<=X_n$，其異常值出現在排列的兩埠。該準則通過計算極差比（對於不同的資料量其極差比計算式也不同），並將極差比與對應的異常置信度表值比較，當極差比大於相應的異常置信度時則判定為異常，否則視為正常。

2.8 網路流量模型分析與探討

2.8.1 常用的網路流量分析模型及其作用

現在對網路流量所進行的模型分析封包括有分形分析、網路流量相關性、神經網路、協定位元元組概率統計分析、隱半馬爾可夫模型（Hidden semi-Markov models，HSMM）等。通過建模分析可以找出網

路流量各種特點以量度網路運行狀況，亦可以根據這些特點來改進原有的或提出嶄新的網路流量處理方法，如通過對網路自相似性的分析提出了一種改進的 Random Early Detection (RED) Algorithm 演算法；路流量模型分析可以得出正常網路的一些正常模型參數值及其範圍，當有攻擊發生的時候，就會破壞這些原有的正常特性使其偏離正常範圍，通過監察這些偏離便能夠有效的監測攻擊的發生，如使用隱半馬可波夫模型來檢測網路流量的異常並發現攻擊。

2.9 結語

在這些模型分析中，協定位元元組統計分析方法是一種比較簡單而行之有效的方法。它通過對各種 IP 網路協定、協定位元元組的流量參數以及它們之間的比例參數進行概率分析以實現網路狀況的異常檢測。在網路正常時，這些參數會根據其所服從的分佈（例如對於網路邊緣分佈層出入口或骨幹網路節點的網路流量，由於其具有足夠多的封包分組數量，根據中心極限定理，其網路流量均值會服從或接近正態分佈）顯現自有的統計特徵，而攻擊流（尤其是 DDoS）則極有可能會引起這些參數的的統計特徵發生異常變化，通過檢測這些異常變化就能檢測到攻擊的發生。通過觀察並比較各協定以及協定位元組的比例分佈來觀測 DDoS 攻擊是否存在及其一些攻擊特性。

而在檢測系統實現方面，過去已有不少文獻嘗試利用協定位元組統計分析對 DDoS 攻擊進行檢測並取得一定效果。一些做法如[9]利用流的四元組標籤（封包來源、目的地址以及來源、目的埠號）作為衡量標準來進行檢測；另外一些文獻則利用各協定比例作為檢測參數，如[10]中以五元組參數（UDP 協定比例、ICMP 協定比例，以及流的長度、持續時間、產生速率等)作為特徵參數進行檢測；還有如[11]、[12]、[14]則以 TCP 封包頭協定的某些位元組作為參數進行檢測，前兩者分別通過簡單地比較 TCP 封包中的 SYN 位元組與 ACK、FIN 位元組的數量大小來判定是否出現 SYN 洪水攻擊，而後者則分析 TCP 協定的 FLAG 位元元組中各個標誌位元之間的協方差，得出正常範圍以及受到

DDoS 攻擊時候的水準，並據此設定簡單的閥值便可以較好的對攻擊的發生進行判定。

以上這些文獻均取得了一定程度的成果，但由於它們的監察參數或者只是針對流標籤，而缺乏對流中的各種成分進行更細緻的分類檢查；或者只是針對個別 TCP 位元組（FLAG 位元組），僅適用於 TCP 類攻擊，而不能檢測前面介紹的其他多種非 TCP 協定的 DDoS 攻擊，故此實際中應該同時對其他一些應用廣泛的協定如 IP、UDP、ICMP、IGMP 及其協定位元組均作出分析，才能有效防範多種類型以及混合型的攻擊。

對此，文獻作出改進，將統計範圍擴大到十個位元組或多位元組組合，取得了更細緻和精確的檢測效果，能夠檢測的攻擊種類也大為增加，但不足的是，它的設計仍然未能對更多的位元組做出全面的分析以更好的反映網路狀況，另外其設計需要對網路中現有的大量路由設備進行改動才能取得效果，這使得其研究僅限於實驗研究而不能做成一個實際應用的系統。

另外這些文獻中當發現有可疑的異常流時只是進行粗細微性處理，即僅僅簡單的將可疑資料分組或可疑流量丟棄的，而未能以細緻的分級限制措施來處理，從而導致在系統誤判率較高時會發生大量錯誤丟棄封包。

第三章 系統設計

本系統設計是一個可部署在IP網路分佈層關鍵出入口或骨幹網路出入口的網路安全設備，用於抑制異常的網路流量以保護網路正常通信，使網路即使在遭受諸如SYN泛洪、ICMP泛洪等一種或多種已知或未知的分散式阻斷服務攻擊情況下仍然能夠提供最大限度的正常網路服務。

3.1 系統架構設計說明

本系統使用多位元組統計檢測方法對網路異常進行監測。多位元組統計，就是對IP網路中的資料分組表頭的各位元組進行解析，並根據需要分別分項統計，本模組設計是在系統所完成的多位元組累積數量統計的基礎上，進一步完成多位元組檢測及資料流程分級控制。

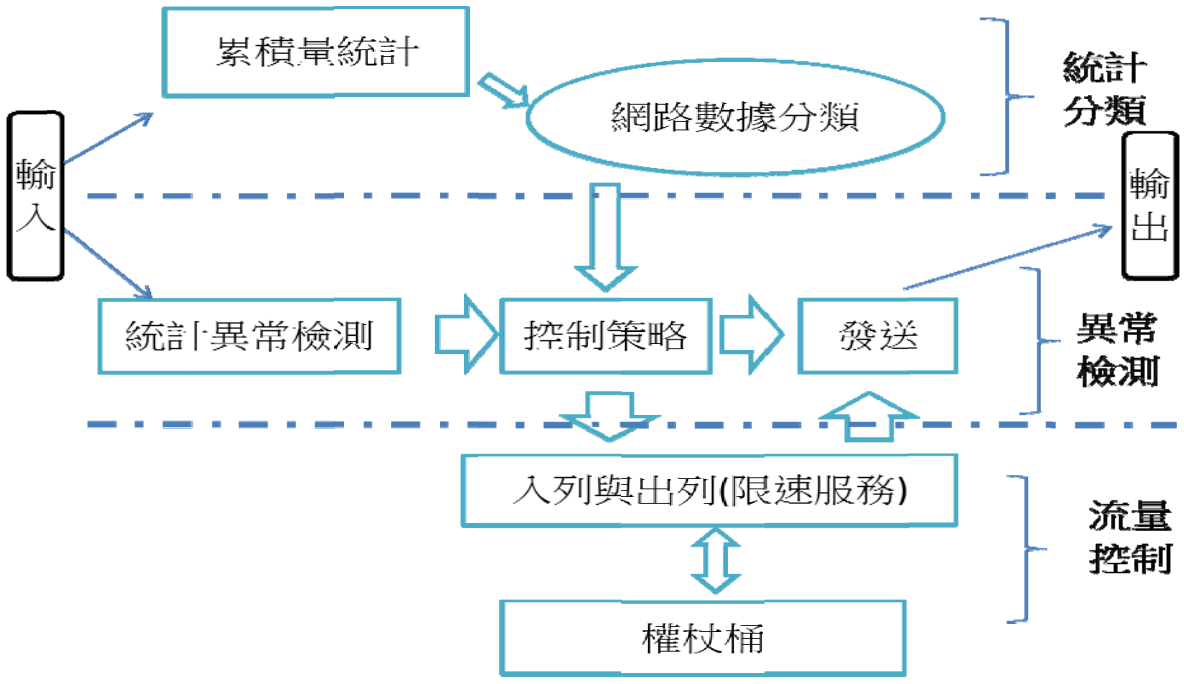

圖 3-1 整體設計圖

3.1.1 系統架構模組

整個系統設計主要包含下列三大模組

1.統計分類: 由於網路位元組攜帶有網路通信的各種重要資訊，對它們進行統計分析有助於深入瞭解網路的運行狀況。在網路正常運行的情況下，各位元組項會因應網路所提供的服務以及網路狀況而各自出現特定統計特徵同時，各個位元組之間也會因應網路行為而出現在數量及比例上的種種特殊聯繫。

2.異常檢測:一旦發生攻擊，就極有可能破壞這些固有的特徵或關係而顯現異常。本模組針對這種特點，對各項統計資料進行異常性分析，並得出其異常程度指數，當有異常情況顯現並超出一定範圍時可以將異常程度指數最高的位元組編號記錄下來（本設計中對異常指數最高的四個位元組項進行記錄)，然後根據此數據把流經本系統的網路資料流程中含有這些異常位元組項的資料分組根據其異常程度指數的高低編入相應的分級速率限制佇列進行流量控制，從而抑制異常流量的影響，以保證在遭受攻擊時仍然能夠維持網路的正常運行。

3.流量控制：

流量控制應用於正常流量時，就是對正常流量按照需求進行資料流程分類並引導到不同的控制佇列，以達到按照應用需求進行頻寬管理的目的。流量控制亦可以應用於異常流量的控制，這時則根據預定的策略對檢測到的異常流量進行限制以避免或減低這些流量對網路所造成的影響，而對合理佔用的則給予較高 QoS 的佇列調度來達到限制攻擊的目的。

當今最著名而被廣泛應用的流量控制技術莫過於 TCP 協定中採用的滑動視窗(Sliding Window)；另外，權杖桶篩檢(Token Bucket Filter，TBF)流量控制也是一種較常用的流量控制技術。

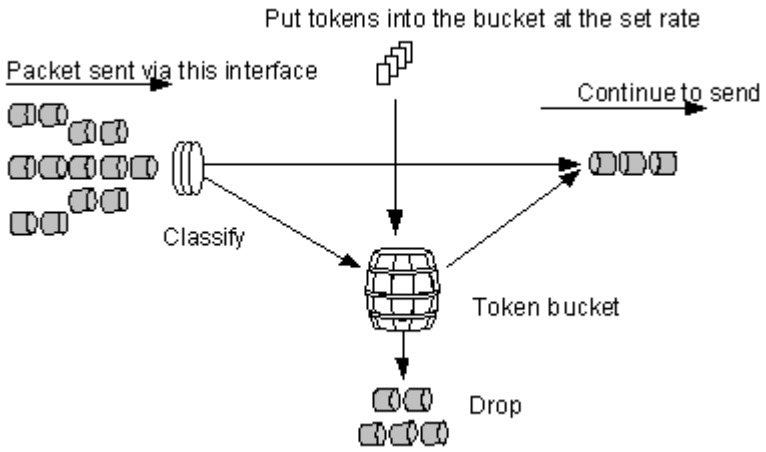

圖 3-2 權杖桶篩檢

權杖桶篩檢（Token Bucket Filter，TBF)，簡稱為權杖桶。其基本思想是通過控制權杖流入流出權杖桶來調控網路中流經的封包（如圖3-2)，從而達到調控資料流速率，使網路流量平滑，避免過大的突發流量，以達防禦的目的。

3.2 檢測演算法

本系統的異常檢測演算法需要對多位元統計中的異常資料進行準確挖掘並且能夠對異常程度進行恰當評估，同時本系統是一個基於IXP2400 的系統，而 IXP2400 沒有除法指令，不支援浮點運算等特點決定了在此平臺上不適合進行一些高等的或過於繁複的數學運算，故此尋找一種能線上實現而且能有效進行檢測的演算法是本系統設計的關鍵。

3σ 異常判定準則在計算上簡捷有效，同時具有不用查表的優點。它適用於對於正態分佈的統計資料進行異常檢測，如果將網路中各個用戶的各個流量資料看作獨立而服從同一分佈的序列，則根據獨立同分佈中心極限定理，在數量足夠大的時候，所有這些流量的累加所構

成的流量均值可以看成正態分佈或近似正態分佈（對於骨幹網或分佈層上的網路資料流程一般均能達到較大的資料量，從而滿足中心極限定理條件）。為此，本設計中可採取 3σ 準則對多元統計資料進行異常性檢測。但鑒於在 IXP2400 中運行該演算法仍然存在一定困難，本系統根據需要並結合硬體條件，在保留了原準則演算法特點的基礎上對一些變數的求法進行了適當改造，分別利用滑動標準差以及滑動平均來分別簡化原演算法求標準差和平均數的計算，這樣一方面可以避免或減少諸如開平方等不適合在 IXP2400 上實現的計算，另一方面也可以避免儲存大量的歷史統計資料，以節省儲存空間和減少記憶體通路時間，提高效率。由此得到的新的演算法如下：

設自統計開始第 n 個統計間隔內的資料分組到達率為 AR_n，而自統計開始的前 n-1 個統計間隔內的到達率的平均值為 A_{n-1}，則可以計算 n 個統計間隔後的統計平均值 A_n：

$$A_n = (1-\beta)* A_{n-1} + \beta *AR_n \quad , \quad 0 \le \beta \le 1 \qquad (1)$$

其中，β 越接近 0 則舊有統計平均值（歷史狀態值）的權重越大，A_n 越能反映歷史狀態資料的重要性，但是這時 A_n 對新資料的變化的反應會相對比較慢；反之，β 越接近 1，越能體現當前新資料的影響。鑒於 β 一般設為較接近 0 的值以便更能體現歷史狀態值的重要性，同時考慮到在網路處理器上的計算效率，本設計中 β 取為 1/4。

在本模組中，每個統計間隔設為 1 秒，則 AR_n 的獲取方法為：利用微處理器中的計時器，每定時 1 秒讀取多位元組解析模組統計的各位元組數量累積值，減去前一次的讀取值，即可得到本統計間隔內的資料分組速率。

根據 3σ 準則原演算法，計算資料分組到達速率的標準差需要進行開平方運算，而本設計中使用滑動標準差來求得。首先計算 AR_n 相對於上一個統計平均值 A_{n-1} 的偏離變化 DIF_n' 的絕對值：

$$DIF_n = |DIF_n'| = |AR_n - A_{n-1}| \qquad (2)$$

接著使用偏離變化 DIF_n 來計算到達率的標準差 σ。

設自統計開始第 n 個統計間隔內的資料分組到達率偏離變化為 DIF_n(單位：資料分組個數/秒)，前 n-1 個統計間隔內的到達率的滑動標準差為 σ_{n-1}，則可以計算 n 個統計間隔後的滑動標準差值 σ_n：

$$\sigma_n = (1-\beta) * \sigma_{n-1} + \beta * DIF_n \qquad (3)$$

其中 β 的取值討論同前；此外，本設計中初始值均設為零，即 $A_0 = 0$ 以及 $\sigma_0 = 0$。

當進入檢測階段後，將求取各位元組的異常程度指數。設封包含某一位元組項 k 的第 n 個統計間隔內的資料分組到達速率為 AR_n^k，該位元組項在學習階段求得的正常到達率平均值為 A^K，則令

$$DIFF_n^k = | AR_n^k - A^K | \qquad (4)$$

則當 $DIFF_n^k > 0$ 時，該位元組在第 n 個統計間隔的異常程度為：

$$ABL_n^k = \frac{DIFF_n^k}{\sigma^K} \qquad (5)$$

其中 σ^K 為在學習階段求得的位元組 k 的標準差的正常參考值（這裏如果 $\sigma^K = 0$，則令 $ABL_n^k = 0$）。

如果 $ABL_n^k \geq 3$，根據 3σ 準則理論，僅有小於千分之三的概率會出現這種情況，即表明該位元元組第 n 個統計間隔內的到達率的偏離

程度超出正常範圍，該位元組將被納入為可疑位元組進行規管。如果這種偏離一直持續下去，可進一步確定該位元組流中含有異常（攻擊流）成分。

第四章 系統實作

4.1 系統流程

本節系統流程以資料流程圖(Data Flow Chart)說明，以下分別是實作一:異常檢測的流程和實作二:異常防禦的流程。

4.1.1 實作一

異常檢測的流程

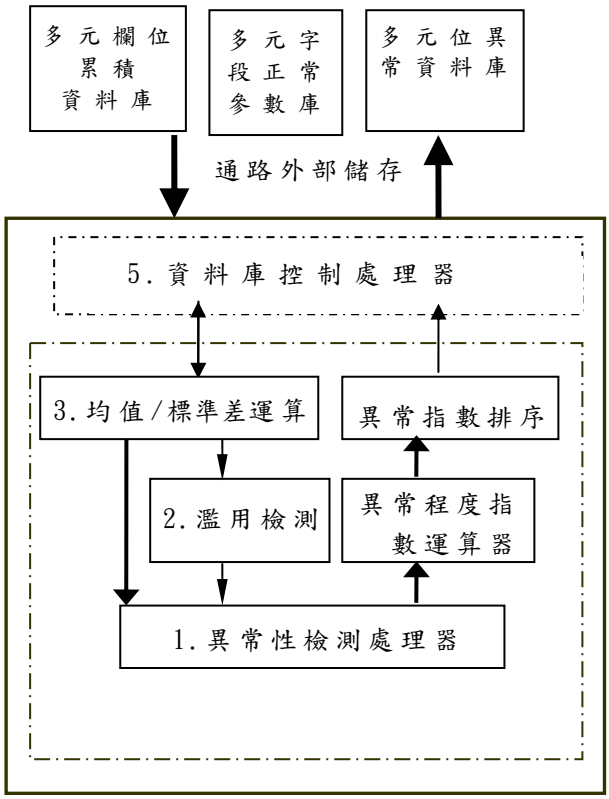

圖 4-1 統計異常檢測模組流程圖

1.異常性檢測處理器:對封包的統計資料中的每一個位元組項求異常程度指數並對超出一定範圍的異常程度指數進行排序，得出前 4 位及其對應的位元組編號，保存在異常指數排序表中

2.結合異常檢測法（Anomaly Detection）和濫用檢測法(Misuse Detection)兩種檢測方法，並且採用以前者為主，後者為輔的檢測方式來對多元統計資料進行監控和分析， 據各種網路協定自身的特點及協定之間關係的一些固有特點而顯現出特定的統計特徵。

3. 對正常資料流程各位元組元統計資料進行均值及標準差分析，求出在沒有遭受攻擊的情況下，網路流量統計中每一個多位元組項的正常特徵參數。開始根據以上學習獲得的參數作為正常狀況的標準參數來進行異常性檢測。

4.資料庫控制處理器:負責對三組資料庫進行全部瀏覽讀取以及將運算結果寫回，同時負責根據預設的定時程式去更新與其他模組的介面資料結構。

4.1.2 實作二

異常防禦的流程，主要用權杖桶參數進行控制和管理，以達到對進行流量控制，也就是防禦功能。

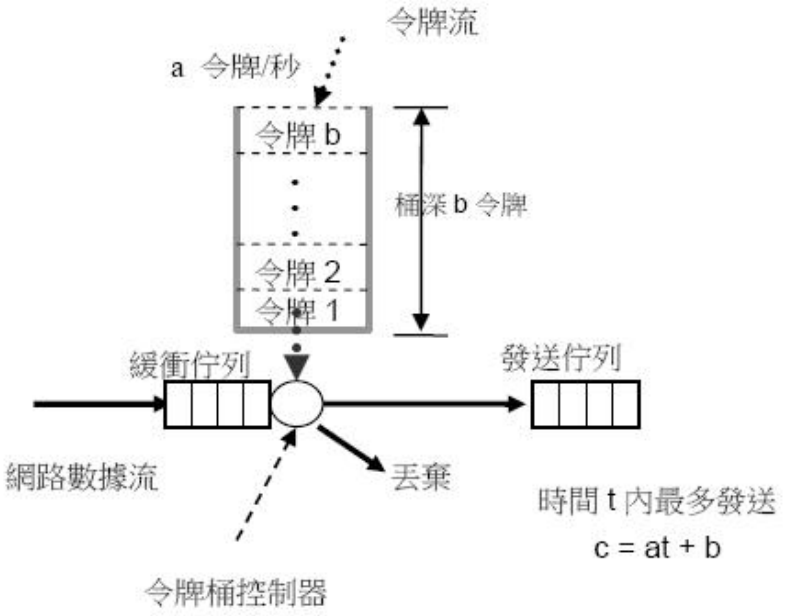

圖 4-2 權杖桶流量控制示意圖

1. 資料流程以等於權杖流的速率到達權杖桶。這種情況下，每個到來的封包都能對應一個權杖，然後無延遲地通過佇列。

2. 資料流程以小於權杖流的速度到達權杖桶。通過佇列的封包只消耗了一部分權杖，剩下的權杖會在桶裏積累下來，直到桶被裝滿。剩下的權杖可以在需要以高於權杖流速率來發送資料流程的時候消耗掉，這種情況下就會產生小量而且可控的隨機傳輸流。

3. 資料流程以大於權杖流的速率到達權杖桶。這時桶裏的權杖很快就會被耗盡，而且緩衝佇列將會被占滿。如果封包持續超出限制地到來，則超出的那部分封包將會被丟棄。這時雖然流進權杖桶佇列的

流量速率大於權杖產生速率，但流出的流量將可以比較穩定的維持在權杖產生速率上。

實際應用中，除了權杖產生速率和桶深度外，還可以對緩衝等待佇列的長度進行調節來控制權杖桶的工作性能。一個權杖桶可以由 2 個基本參數描述：權杖流入權杖桶的速度 a（權杖產生速率），權杖桶的深度（最大容量）b。

通過將資料流程關聯到權杖流上，即每個到來的權杖從資料佇列中收集一個封包，然後從桶中被刪除，則某段時間 t 內流過權杖桶的封包的數目 c 最多只能為：c = at + b，這樣便可以達到調控資料流達到防禦的目的。

4.2 系統平台

系統便用到的設備如下:

1. 伺服器 a、b、c:Intel base，安裝 Windows 2003
2. PC，並安裝網路流量抓取器 Ethereal Version 0.10.9。
3. PC，攻擊流量則使用 tfn2k 攻擊器以及一些發封包程式產生
4. 網路交換器使用:Cisco 2950G Switch（Layer 2 Switch）
5. 檢測與防禦系統: Red Hat 7.3.並安裝，ENP-2611，Intel IXA SDK
6. PC，安裝 IXA SDK 3.51 Tools，

4.3 系統模組實作

4.3.1 統計異常檢測模組

統計異常檢測模組的主要功能是對多元統計資料中的每一個位元組項求異常程度指數，並對超出一定範圍的異常程度指數進行排序，得出前 4 位及其對應的位元組編號，保存在異常指數排序表中，表項格式為：

佇列號1-4	異常位元組編號	異常程度指數

圖 4-3 異常指數排序表結構格式

本異常指數排序表將會傳遞給控制策略模組，使其可以根據位元組的異常指數在表中所處位置來控制相應異常資料分組的走向。

本模組分為

1.異常性檢測處理器

本系統中結合異常檢測法（Anomaly Detection）和濫用檢測法(Misuse Detection)兩種檢測方法，來對多元統計資料進行監控和分析。

檢測原理**:**資料流程中各位元組元統計能反映出網路運行狀況，故此會依據各種網路協定自身的特點及協定之間關係的一些固有特點而顯現出特定的統計特徵。例如在一般 IP 網路應用中，TCP 比 UDP、ICMP

協定使用更廣泛和頻繁，而且 TCP 以流作為單位來傳輸資訊，同樣信息量需要產生比 UDP、ICMP 流產生更多的資料分組，由此會導致含有 TCP 位元組的分組在 IP 封包中所占比例比較高，根據對網路核心線路的流量統計，正常時該比例會比較穩定的達到八成以上；而 ICMP 分組只是用於傳遞少數網路控制資訊，一般只占不到資料分組總數的百分之二。又如在 TCP 封包中的標誌位元元組中，SYN 位元組與 Fin 位元組存在互斥關係，在正常情況下不會出現同時具有這兩個位元組的分組；另一方面，由於這兩個位元組在同一個流當中出現的次數是一樣的，故此在理想情況下其總體統計數量應該大致相當（考慮到現實中會存在一定數量的半連接未能完成三次握手，或者一些 TCP 流未能以正常的 FIN 結束，SYN 會比 FIN 的大，根據對流量進行分析可知 SYN 數量一般為 FIN 數量的 3-5 倍）。又再如，在正常 TCP 流中 ACK 遍佈 TCP 連接整個（包括開始和結束時的三次握手）過程，而 SYN 或 FIN 位元組只會出現在流開頭結束時的三次握手過程，故此 ACK 位元組的比例應當會遠高於 SYN 與 Fin 等位元元組（根據對的流量進行分析，ACK 位元組的數量一般可達到 SYN 位元組數量 8 倍以上）。

根據網路所提供的服務以及本系統所處的網路結構也會進一步使某些位元組出現其特定統計特徵。例如如果網路所提供的主要是 http 服務，則正常情況下 TCP 封包會佔所有 IP 封包中比較大的部分，而且埠 80 的數量會占所有 TCP 埠總數的較大比例。相反，如果是網功能變數名稱解析服務（DOMAIN NAME SERVER， DNS），則其中 UDP 封包的數量會比較多，而且在埠位元組元比重中，目的埠號為 53 的分組比例也會明顯比較高。

而一旦出現攻擊則很可能破壞以上各種數量及比例關係結構，而攻擊者一般難以知道其將要攻擊的網路獨有的特徵參數，亦難以在攻擊的同時對正常結構進行類比，例如發動 SYN 洪水攻擊時攻擊者為了取到良好的攻擊效果，極有可能在發出巨大的 SYN 流量後不回應 SYN+ACK 封包（否則就會消耗攻擊機器的計算及網路資源，在很大程度上減緩了攻擊速度），這就必定導致在受到攻擊時，受害系統接收的 SYN 封包比 ACK 封包的比例異常增大。或者當發生 ICMP 洪水攻擊攻

擊時，則出現 ICMP 數量異常增大，而且很可能破壞 ICMP 與總分組數量以及與 TCP 的數量或比例關係。

綜上所述，各種協定及位元組數量及比例關係的自然結構是網路正常的顯著特徵之一，而且在攻擊時難以類比這種結構，故此這種特徵可以用於區分正常與異常流量。由於異常檢測法正是以正常行為作為準則進行判定，為此本設計中以這種方法作為主要檢測手段。這種檢測方法具有易於自我適應不同的網路環境（根據不同情況設定不同的正常範圍）、漏報率低以及能夠探測未知攻擊的優點，運用這種檢測方法對所有位元組進行檢測，可以最大限度地探測網路運行狀況。而本系統的已經對多位元組進行累積量統計，本模組則進一步以到達率平均值及標準差為量度參數對各位元組統計資料進行訓練，以求得網路流量正常狀況下各位元組參數大小及其變化範圍等特徵，並在此基礎上進行異常性檢測。

本模組分成兩個階段來進行：

學習階段：對正常資料流程各位元組元統計資料進行均值及標準差分析，求出在沒有遭受攻擊的情況下，網路流量統計中每一個多位元組項的正常特徵參數(到達率的正常特徵平均值 A' 和正常特徵標準差 σ')，並存放在正常流量統計特徵資料庫中。為了保證所學習的參數能夠充分反映網路流量各個位元元組的統計特徵，這裏需要為學習階段的結束設定一定條件，包括：

(1)已經學習一段充分長的時間

(2)對足夠多的資料分組進行學習

(3)沒有發現明顯異常

檢測階段：學習階段結束後，將可以進入檢測階段，開始根據以上學習獲得的參數作為正常狀況的標準參數來進行異常性檢測。本階段在對資料流程進行滑動平均和滑動標準差統計的同時，將計算異常程度指數以進行異常分析。當發現異常程度指數超出一定範圍後，將其輸出到異常指數排序器以便通知控制策略模組。

本檢測方法在學習階段需要滿足三個條件才能判定學習參數有效繼而轉入檢測階段。做出前兩條限制的目的是為了使學習的結果能滿足大數定理的要求。按照大數定理，只有在統計資料數量足夠的情況下才能將統計量中的隨機性基本去除而體現內裏的概率規律，而為了獲得足夠的統計資料也需要經歷一定時間。當未能符合條件，則學習階段可能要反復進行。為了在學習階段仍能作出一定程度的檢測，本模組還需使用濫用檢測方法作為此階段的替代檢測手段。

濫用檢測方法無需為了獲得所在網路的正常參數而經過學習階段，只需根據一些已知的常見 DDoS 攻擊所引起的異常流量特點，預設一定的安全規則，當某些位元元組或位元組之間的統計特徵符合這些規則時，就可以肯定地判斷為攻擊的發生，為此，在本設計中引入濫用檢測法，可以幫助在異常檢測方法的學習階段監測流量中是否存在符合預設入侵規則的明顯的攻擊特徵，並以此作為異常檢測的學習結果是否有效的判斷依據之一。而根據前面對一些常用攻擊發生時對流量特徵產生的影響的分析，可設定的一些安全規則包括：

表 1 濫用檢測規則

	異常特徵	
SYN 攻擊	SYN 平均到達率 > 1/2 Ack 平均到達率	SYN 平均到達率 > 1/4 TCP 平均到達率
ICMP 攻擊	ICMP 平均到達率 > 1/6 TCP 平均到達率	ICMP 平均到達率 > 1/8 IP 平均到達率
對埠的泛洪攻擊	某一埠口平均到達率 > 1/2 IP 平均到達率	

另外，濫用檢測也可以在異常檢測進入檢測階段後使用，但由於其判斷依據只是在系統運行前對一些異常行為所作的預設，而不能根據系統運行後的網路狀況自適應調整參數範圍，是以只能作為一種輔助手段，即僅作為對異常檢測法的補充。

最後本檢測器會通過異常程度指數排序器對指數超過預設範圍的位元組進行排序。由於本系統設計中的限速模組設定了四級流量控制佇列以適應多種攻擊同時發生時所出現的數個位元組同時出現異常的情況，故此本模組最多將能記錄異常程度指數最高的四組位元組項，並依照異常指數的大小去分別對應這四條佇列。

2.資料庫控制處理器

資料庫控制處理器主要執行對各組資料庫的讀寫更新操作。

4.3.2 控制策略與模組

本模組主要任務是區分正常和異常資料分組，並且根據異常程度來對其進行不同的出口控制。

首先由位元組匹配檢查器讀取由統計異常檢測模組提供的異常程度排序表，得到異常程度最高前 4 位的位元組；而佇列指派器據此對網路資料分組進行分類：將含有排序表中訓示的異常位元組的分組按照異常程度高低（如果同一分組出現兩個或兩個以上異常位元組則只需取異常指數最高的位元組）打上對應的限速佇列的標籤，並將分組送入本模組，以便服務模組根據該標籤將分組指派到相應的限速佇列；對於不含有異常位元組的分組，佇列指派器則將其送到與發送模組的介面佇列中，使其可以被直接發送出去。

本模組與服務模組的介面消息為一個長字（32bit）的入列請求（如圖 4-4）。通過設定其中的限速佇列號，可以使服務模組按照本模組的要求將資料分組歸入相應的限速隊伍中接受進一步處理。

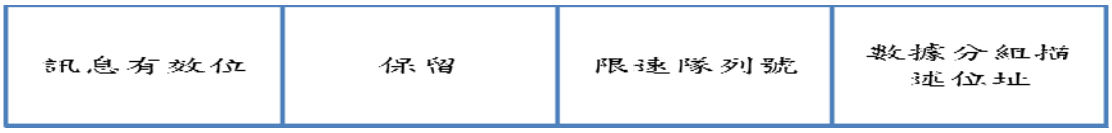

圖 4-4 控制策略模組與服務模組介面消息結構

上述請求格式同樣適用於本模組與發送模組的介面消息，此時只需要將佇列號置零以表示是直接發送佇列。

4.3.3 權杖桶控制模組

本模組主要用於對權杖桶參數進行控制和管理，以達到對異常網路資料分組進行流量控制的功能。權杖桶佇列將資料流程關聯到權杖流上，通過調整權杖桶運行參數便可以達到對網路流量進行調節和管控的目的。為此，能夠根據流量控制需要準確地確定權杖桶的權杖產生速率、桶的深度還有限速緩衝佇列的長度等工作參數，是本模組對網路流量取得良好控制效果的關鍵。

權杖產生速率的確定 在網路處理器上有兩種途徑來計算權杖的產生速率，一種是根據相對頻寬要求，另一種是根據封包發送絕對速率來確定。

按照頻寬要求來計算權杖產生速率必須要結合 IXP2400 封包處理速率來考慮。可參照的計算公式為：如果處理速率為 T 個時鐘週期，則以 N 分之一頻寬的封包發送速率來發送資料分組時的權杖產生速率為：T * N（單位：時鐘週期/權杖）。而如果以位元/秒（bps）為單位分配頻寬，則還要結合網路流中的平均封包長來計算，這時的計算公式為：如果處理速率為 T 個時鐘週期，則以 N 分之一頻寬的位元元發送速率來發送平均封包長為 L（bit）的資料報時，權杖產生速率應該為：T * N * L（時鐘週期/權杖）。

在應用中也可以循另一途徑，即按照封包發送絕對速率（以 pps 為單位）來確定權杖速率，這時只需根據網路處理器的時鐘速率來計算。計算公式為：

權杖產生速率 = 處理器的時鐘速率 / 封包發送速率。

例如對於 IXP2400 的時鐘速率是 600M/sec，則每秒發送 100 個封包的權杖產生速率為 600M/100 = 6000000 時鐘週期。

本設計中為了在受到攻擊時使經過本流量控制後的各位元組流量能夠維持正常情況下的自然結構，將會按照以下公式來動態調整各級權杖桶佇列權杖的產生速率：

假設權杖桶佇列 m 中對應的是含有位元元組 k 的資料分組流，而學習後獲得的位元組 k 平均到達率的正常參考值為 AR_k'，總數據分組平均到達率的正常參考值為 AR_{tot}'，而在檢測階段發現異常前的總數據分組平均到達率為 AR_k，IXP2400 的時鐘速率為 TI，那麼該佇列的權杖產生速率 TR_m 為：

$$TR_m = TI / (AR_k * (AR_k' / AR_{tot}')) \quad (6)$$

本設計中利用硬體計時器來定時產生權杖，故此可以計算硬體計時器的定時長度

$$TR'_m = TR_m / 16 \quad (7)$$

權杖桶深度以及緩衝佇列長度的確定 這組參數的確定原則是既要避免權杖桶出口出現過大的突發流量，又要能夠起到一定的緩衝作用，避免因為網路中的某些波動而導致立刻丟封包。這裏要結合權杖產生速度來考慮，實際設計中將權杖桶深度設為 64 比較適宜。這樣 4 個桶深度合共 256。當四個權杖佇列都充滿時的突發封包發送速率（設每個佇列的權杖產生速率平均為 100 權杖/秒）最大不會超過（100 權杖/秒/佇列 * 1 秒 + 64 權杖/佇列 ）* 4 佇列 = 656（pps）。對於最大長度為 1500 位元元組的資料封包，其最大突發速率為 656 * 1500 * 8 =

7.8 Mbps。這對於千兆級的頻寬容量的影響並不算大。而緩衝佇列的最大長度可以設為與其對應的權杖桶深度兩倍，這時緩衝長度已經超過每秒發送量，可以起到較好的緩衝作用。在流量限制許可範圍以內到達本模組的資料封包將會被暫存在佇列中，等待權杖到來後才被發送出去；而當限速佇列充滿時，就開始執行將超過限制的資料分組予以丟棄的管制措施。

4.3.4 入列與出列服務模組

入列與出列服務模組主要是為異常資料分組進出四條權杖桶限速佇列提供佇列管理控制功能，分為入列服務和出列服務兩部分。

入列服務部分主要負責回應前一模組的排隊要求，接收傳過來的介面消息，根據其中的佇列號將其加入到相應的佇列中，直到佇列排滿，此時就會對新到資料封包採取丟棄策略，並維持至佇列不滿為止。

出列服務模組主要負責回應權杖桶控制模組的出列要求，從相應的佇列按照先入先出的順序，將其中排在佇列最前的分組描述符取出，傳到下一發送模組中。

結合 IXP2400 的處理特點及儲存器通路優化技術所設計的入列/出列服務及權杖更新流程。保證每條佇列在較短的均勻的間隔內能接受服務。由於 4 條限速佇列共用一條發送佇列出口，為此在這個出口占滿時，所有佇列均停止發送，同時也應當停止入列操作以等待公共出口可用為止。而第二階段則對入列請求進行解析，並將合法請求所對應的資料封包入列，並修改佇列參數，而對於不符合要求的請求就會釋放其所佔用的系統資源；同時本階段也會對每個佇列的權杖數目及權杖桶參數進行更新。

4.4 系統評估與測試

本節設計了一系列實驗和模擬來檢驗系統的功能，並做出實驗的結果。

1.對各模組的基本功能測試，以及用於實際網路運行時的性能評估。本評估實驗中所使用的正常流量輸入樣本均取自上對網際網路骨幹線路所採集的流量資料

2.對攻擊流的檢測與控制能力測試，以評估本系統的防禦性能

。

4.4.1 系統功能評估測試

本測試部分將對本系統設計中的統計檢測通道及資料分組控制通道分別進行基本功能驗證測試。同時，能否正確識別正常流量並讓其透明的通過本系統也是本系統基本性能的重要指標，本部分也會對此進行評估。

本測試的網路拓撲示意圖如下：

圖 4-5 模組基本功能測試示意圖

主機 a 與主機 b 分別與基於網路處理器 IXP2400 的多元檢測控制系統的埠口 0 和埠口 2 相連，主機 c 與 IXP2400 的調試埠連接。主機 a 向 IXP2400 的埠口 0 發送資料封包，IXP2400 對這些資料封包進行處理後從埠 2 發出去，使主機 b 可以將其接收。而主機 c 則通過 IXP2400 調試平臺在運行期間即時監控網路處理器內部的各種資訊，包括即時查看微引擎內部寄存器值、信號以及線程狀態，各級儲存器以及輸入輸出緩衝器中的資料，還有各種控制狀態寄存器的資訊。

首先測試統計檢測通道能否按照預定的演算法及流程工作，主要檢驗統計異常檢測模組能否正確地對多位元組累積量統計資料庫作出定時遍曆讀取並進行運算分析，然後將結果用於更新多位元元組異常分析資料庫及異常指數排序表等資料結構。由於只需要通過檢驗能否在 SRAM 的資料庫中最終得到正確的運算結果就可以驗證整個流程能否正常工作，為此只需對 SRAM 中的多位元元組異常分析資料庫進行觀察檢驗。IXP2400 中的即時運行結果可以通過調試平臺上的 SRAM 觀察視窗觀察相應的資料庫儲存位址範圍內的資料

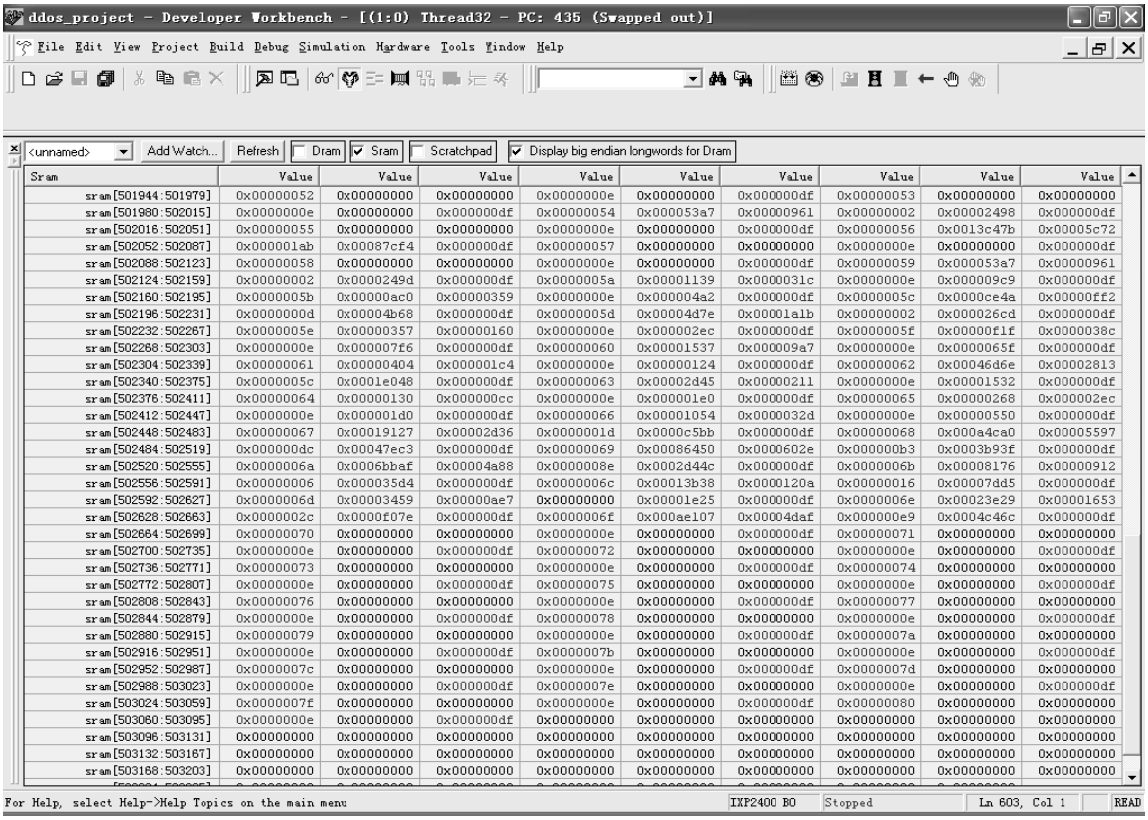

圖 4-6　IXP2400 儲存器資料觀察視窗

為了作出對比驗證，在主機 a 發出封包的同時，利用網路流量抓取器獲取被發送到本系統中的網路流量，這樣就可以複製得到與本系統所需處理的同樣資料流程，然後按照本系統中的同樣演算法對抓取的資料流程的各欄位元流量進行統計分析，並將其結果與網路處理器中的結果列成表格進行比較分析，如表所示（表中僅列出 5 個關鍵位元組結果，其他位元組結果類似）。

表 2 功能驗證結果比較表

結果 / 類型	網路處理器中資料結果			網路流量抓取器統計結果		
	滑動平均值	滑動標準差值	異常指數	滑動平均值	滑動標準差值	異常指數
IP	5144	91	<3	5065.7	83.0817	<3
ICMP	115	10	<3	113.9767	17.4209	<3
TCP	3872	68	<3	3928.3	78.7358	<3
SYN	315	18	<3	316.8290	19.40	<3
HTTP	1133	40	<3	1093.9	46.2722	<3

從表中資料對比可見，在 IXP2400 中的處理結果與網路流量抓取器的統計後的計算結果基本一致：表中各項滑動平均值的正規化差別均在百分之四以內，而滑動標準差的絕對大小差別也在 10 以內（但二者並不需要完全相等，這是因為雖然二者對資料流程的取樣週期間隔大小相同，但取樣不可能完全同步，故此只要差別在一個合理的水準範圍內即可），而異常指數都在 3 以內（即均沒有發現異常流量），從這些比較結果可以驗證本模組可以成功完成預定的任務。

第二步將測試本系統的資料分組控制通道，主要測試其能否讓正常佇列資料透明地通過本系統，以及能否按照預設的速度指標對速率控制佇列資料進行流量控制。

首先檢驗本系統對正常流的影響。為此在 IXP2400 的出埠和入埠進行流量對比，即在圖 4-5 中的主機 a 發封包（重放正常流量）並抓取流量，而在主機 b 接收並抓取流經本系統後的流量（結果分別見圖 4-7a 及 4-7b），並對兩個資料流程作相似性分析得到下表。

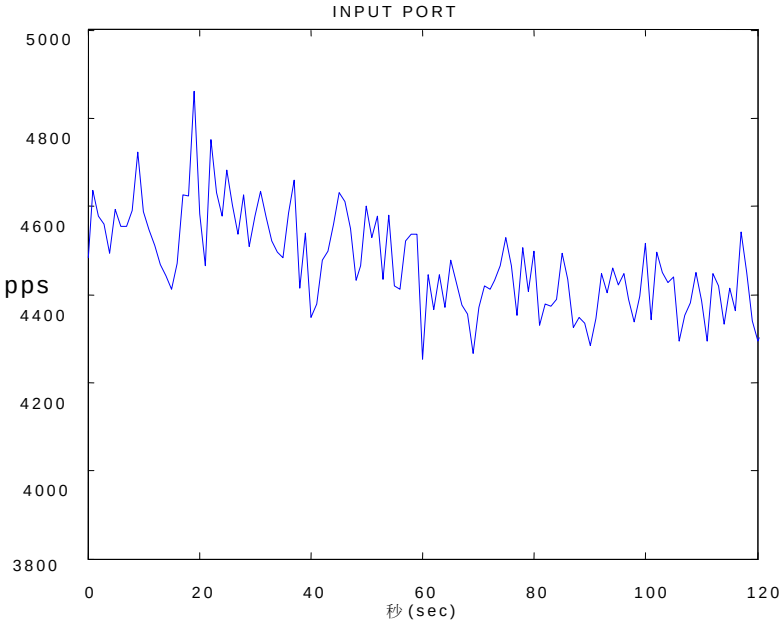

圖 4-7a 輸入埠資料流

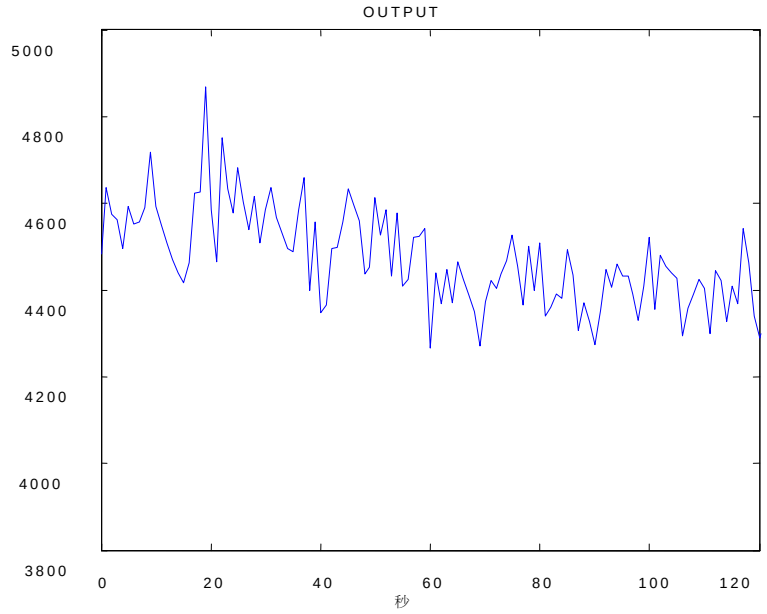

圖 4-7b 輸出埠資料流

表 3 本系統對正常流處理前後變化對照表

處理延時	正規化流量持續時間差異	總封包數差異	每秒平均流量差異	正規化每秒流量差異（每秒流量大小的標準差/ 每秒流量平均值）	封包順序差異數
<0.08ms	0.000065	0	0	0.0019	0

從表中可見，處理延時（即資料流程在本系統中的延遲時間，注意這一項的數值是一個估計測量結果）非常小，說明本系統處理速度非常高，處理過程幾乎不會對正常資料流程產生延遲；兩個資料流程的總封包數差異為零，說明處理後並沒有遺失的封包，而每秒平均流量差異為 0,每秒流量差異值不到千分之二,以及流量持續時間差異(發埠與收埠流量持續時間之差除以發埠總流量時間）微小均說明本系統的處理過程對資料流程所造成的流量抖動非常小，而封包順序差異數為零則說明本系統能夠完全按照原資料流程進來的順序將封包發出。從以上資料可以得出，本系統對正常流的處理過程幾乎沒有對資料流產生任何影響，換句話說，正常資料流程可以完好地通過本系統。

而對異常資料流程的速率控制模組的測試主要是為了檢驗其能否按照檢測模組對其制定的指標進行流量控制。這裏需要首先以正常流對本系統的檢測模組進行訓練，使本系統在流量控制時按照學習到的正常流量參數去制定相應的流量控制需完成的數量。接著在進入檢測階段後在主機 a 持續發出大量 ICMP 封包（>5000pps)，這時查看系統依照學習階段得到的正常參數所制定的流量控制需完成的數量 （見下表)，並且在主機 b 對網路流量進行抓取後提取 ICMP 封包的流量作出統計分析。

表 4 流量控制需完成的數量

正常總到達率參數(pps)	5167
正常 ICMP 比重	0.02
指定 ICMP 限制速率(pps)	5144*0.02 = 103

表 5 流量控制效果

ICMP 到達率樣本 (pps)	(103，104)
ICMP 到達率平均值(pps)	103.3592
ICMP 到達率標準差(pps)	0.4821

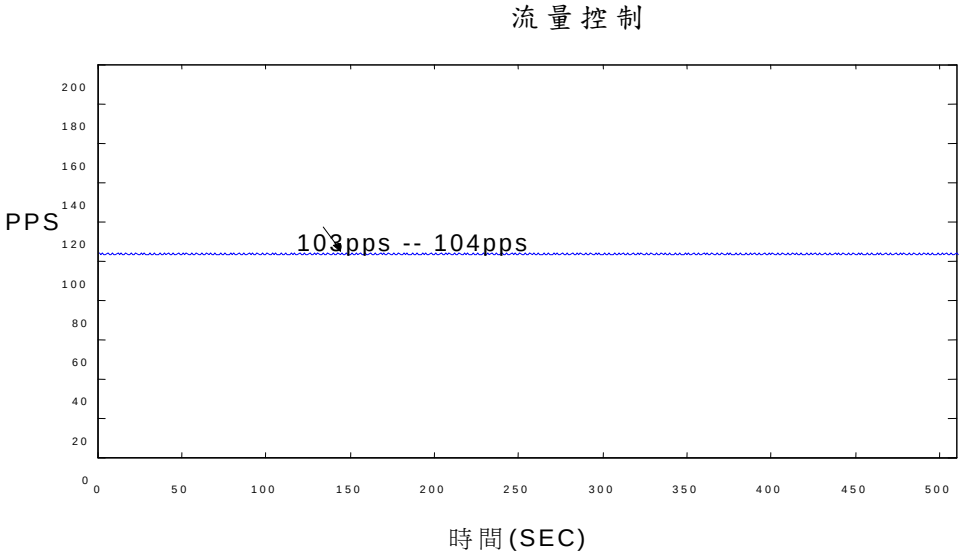

圖 4-8 量控制效果示意圖

從表中資料可見，經過本系統流量控制之後的 ICMP 分組到達率與指定的限速速率 103 pps 基本吻合，雖然稍稍有輕微波動，但也被嚴格控制 1pps 的誤差範圍內（103 與 104 pps），這已經能完全滿足本系統在應用中的設計需要。由此可以驗證，本系統的限速模組能夠根據學習階段結束後所動態制定的速率控制指標穩定地完成流量控制功能。

4.4.2 檢測與防禦性能測試

防禦性能是本系統的設計目標，故此本部分主要測試當攻擊出現時，本系統能否迅速地檢測其發生並且對其作出有效的控制，從而最大限度地維護正常網路服務。

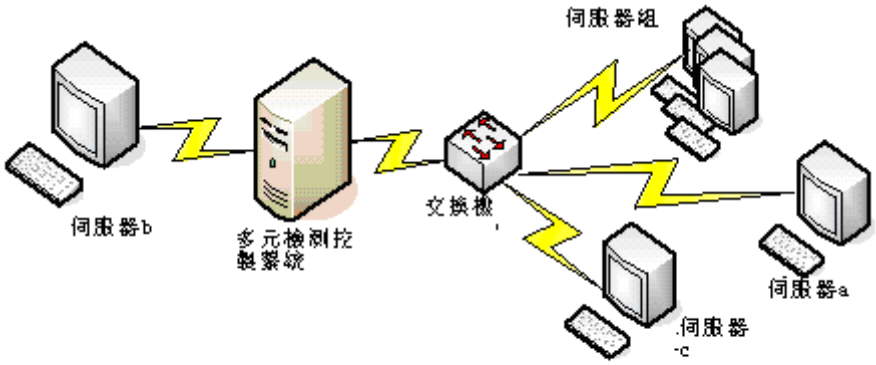

圖 4-9 防禦性能評估實驗網路拓撲圖

本測試的網路拓撲示意圖如圖 4-9。這裏使用一個主機組充當攻擊僵屍機器發動 DDoS 攻擊，而主機 a 則輸出正常流量，它們的流量可通過交換機匯集至本系統的輸入埠，同時利用主機 c 並行採集本系統的輸入埠流量（這裡可以設定交換機的工作模式使主機 c 可以獲得與主機組和主機 a 相連的交換機埠的資料流程量）以及通過 IXP2400 調試平臺即時觀察系統內部情況。而主機 b 則用以採集本系統出口資料流程量。

CASE1: 單種攻擊防禦性能測試

首先進行已知或典型的 DDoS 攻擊防禦測試。在主機組發動高強度的 ICMP 攻擊，並彙聚正常流量，使其一起通過本系統。這時在本系統輸入埠使用網路流量抓取器捕獲網路流量所得到的流量示意圖如圖 4-10 所示。

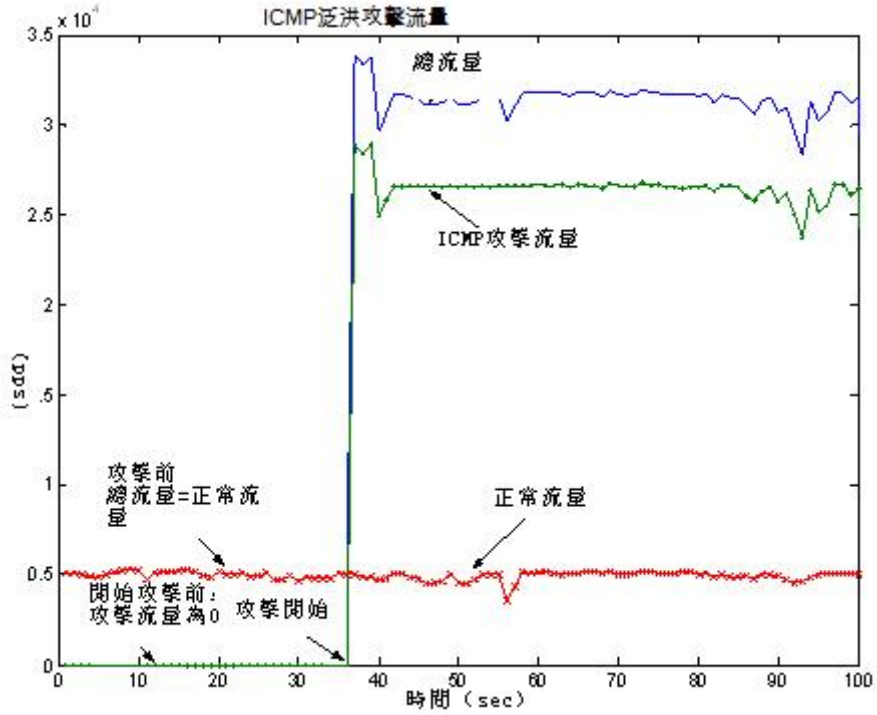

圖 4-10 ICMP 泛洪攻擊流量示意圖（輸入埠）

從圖中可見，當攻擊流加入於正常流量，使總流量迅速提高到原來的 6 倍以上，這時的攻擊流不但會消耗整個網路的巨大帶寬，造成網路速度緩慢，如果讓攻擊流量到達受害目標，就有可能使受害目標的網路及主機資源受到嚴重損耗直至其提供的網路服務陷於癱瘓。

這時線上觀察本系統內部的檢測模組及流量控制模組對流量發生突變後所作出的反應（即一些關鍵的檢測及控制參數的數值，結果見下表），並在系統輸出埠利用流量抓取器經過本系統處理後的流量。

表 6 ICMP 攻擊期間系統檢測結果及控制需完成的數量

攻擊警告訓示器	[異常位元組，異常指數] 集	位元組流量控制需完成的數量
1	[77 (ICMP) ，1808 (>>3)]	77(ICMP) -- 103 pps

從表中可見，系統已經檢測到 ICMP 位元元組存在異常流量（攻擊警告訓示器不為 0），而且異常指數（異常指數實際上在不斷變化，這裏只是列出攻擊期間某一時刻的值，以下各實驗亦同）遠遠大於 3（臨界範圍標準），而按照流量控制演算法得出來的對 ICMP 封包的控制需完成的數量是 103 pps。

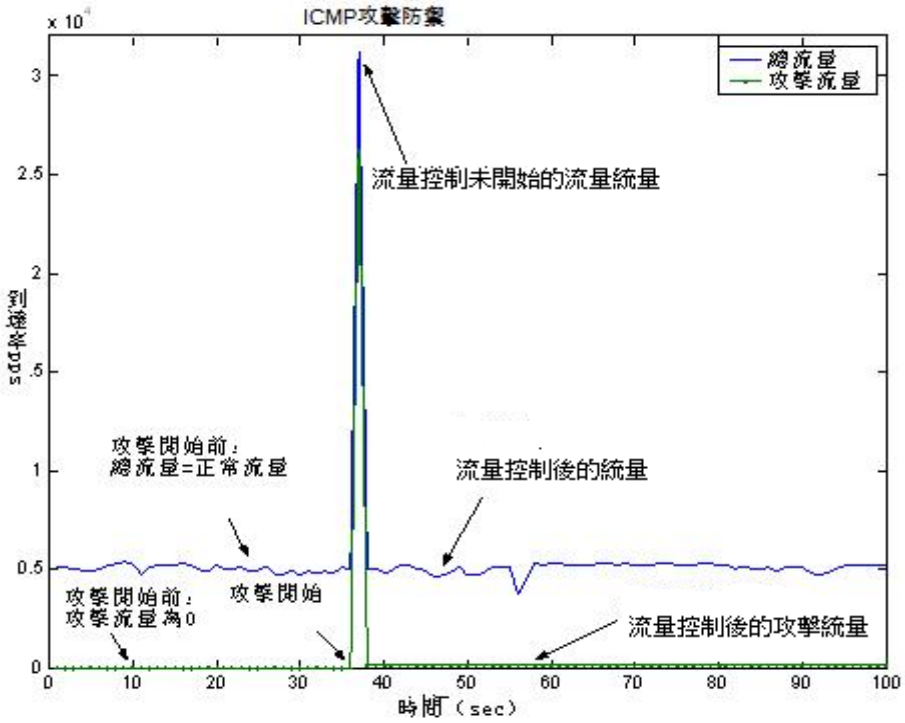

圖 4-11 ICMP 攻擊防禦示意圖

從本系統的出口流量可以分析防禦效果。從圖 4-11 中可見，自第 36 秒攻擊啟動後本系統出口的流量雖然經歷了一個短暫時間(第 36-37 秒)的流量急升，但是由於攻擊流量被迅速壓制在一個非常低的水準，總流量也立即回落到攻擊發生前的大致水準，這時雖然不能將攻擊流完全遮罩，但比較輸出埠總流量與輸入埠正常流量可見，已經幾乎感覺不到攻擊的存在(總流量僅比正常流量增加不到 2%)。同時，比較輸出的正常流量與輸入的正常流量統計資料可以知道，絕大部分(>98%)的正常流量仍然可以如常地通過本系統。由此可見，本防護系統的檢測控制功能可以對實驗中的 ICMP 攻擊作出非常好的防禦作用，最大限度地維護正常流量的同時有效地遏止了攻擊將會導致的嚴重的破壞後果。

CASE2: 高強度的 SYN 攻擊測試

從圖 4-12 中可見，對 SYN 攻擊的防禦效果與對 ICMP 攻擊的防禦效果類似，但由於在正常流中含有 SYN 位元元組的資料封包所占的比例較含有 ICMP 位元組的封包的大，故此對其的流量需完成的數量也會相應較大，但對於總流量的影響也只是增大了約 5%左右的流量規模。這說明本系統對於 SYN 攻擊也能做出較好的抵禦。

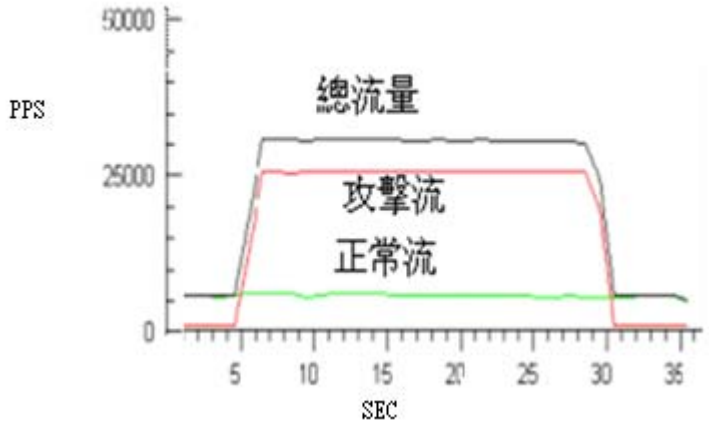

圖 4-12 SYN 攻擊流量（輸入埠）

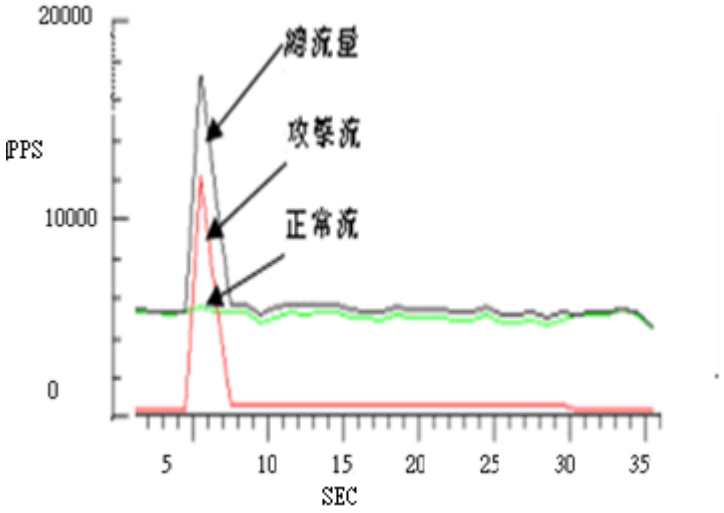

圖 4-13 SYN 攻擊防禦（輸出埠）

CASE3:針對未知或變種的攻擊測試

本實驗模擬在網路中發動針對郵件系統服務(25)埠口（非 SYN 分組）的 DDoS 攻擊，類似於前兩個實驗，通過比較這些流量在通過本系統前後的變化便可以測試系統對其所作出的檢測與控制的表現。

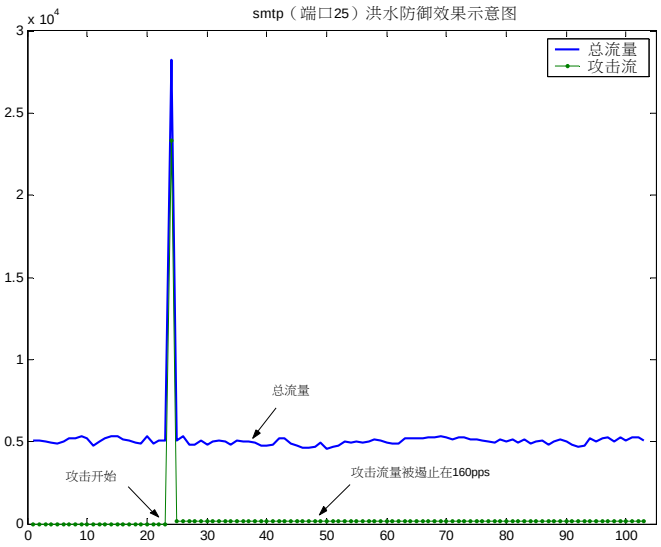

圖 4-14 埠泛洪攻擊流量（輸入埠）

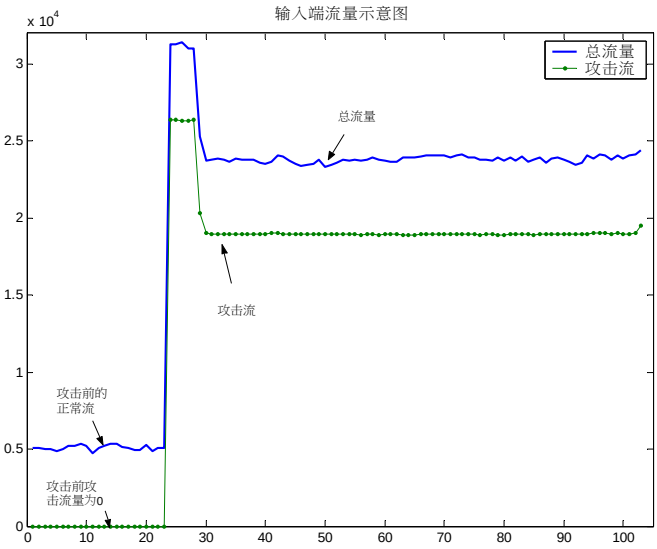

圖 4-15 埠泛洪攻擊防禦圖（輸出埠）

表 7 埠口攻擊期間系統檢測結果及控制需完成的數量

攻擊警告訓示器	[異常位元組，異常指數] 集	位元組流量控制需完成的數量
1	[92(port25)，1047(>>3)]	92 (port25) -- 160 pps

混有埠口泛洪攻擊的網路流量進入本系統時的流量如圖 5-11 所示，而從系統出來的流量如圖 4-14 所示。從圖 4-15 中可見攻擊流於以正常流後使流量大幅飆升。這時本系統能夠檢測到位元組 92 (port25)出現異常並對其進行了適當的流量控制。故此，與輸入埠的流量進行對比後可知，經過本系統的檢測與控制後，輸出埠的攻擊流量已經得到極大的抑制。在本實驗中雖然僅使用埠 25 作攻擊防禦測試，但由於系統在做多元統計及檢測分析時對其他埠的處理與對埠 25 的處理類似，故此可以表明即使在其他埠發生類似的泛洪流量（例如針對某一未能在檢測前預知的應用程式通信埠口所進行的網路蠕蟲攻擊或攻擊者為了逃避檢測而隨機選擇攻擊埠口的攻擊），本系統也能夠對其進行同樣的抑制。由此可以驗證本系統對一些未知或發生變種的分散式阻斷服務攻擊仍然具有良好的抵禦能力。

CASE4:強度較小的攻擊測試。

從主機組發動攻擊總強度約為 1Kpps 的 SYN 攻擊。這時的攻擊規模雖然達到了正常時的 SYN 分組流量的近 4 倍，但僅為正常時的總流量的約 20% (圖 4-16)。而從下表中可知，在攻擊啟動後系統能夠檢測到 SYN 位元元組流量出現異常，並且對其制定的流量限制需完成的數量為不超過 309 pps。

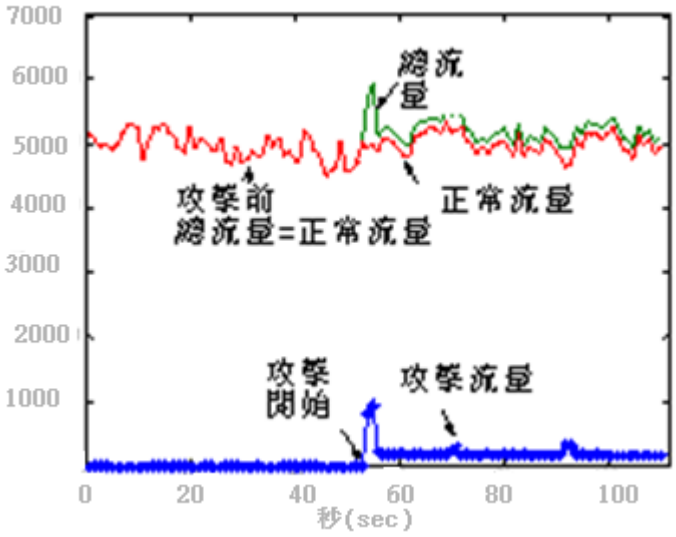

圖 4-16　SYN 攻擊流量示意圖（輸入埠）

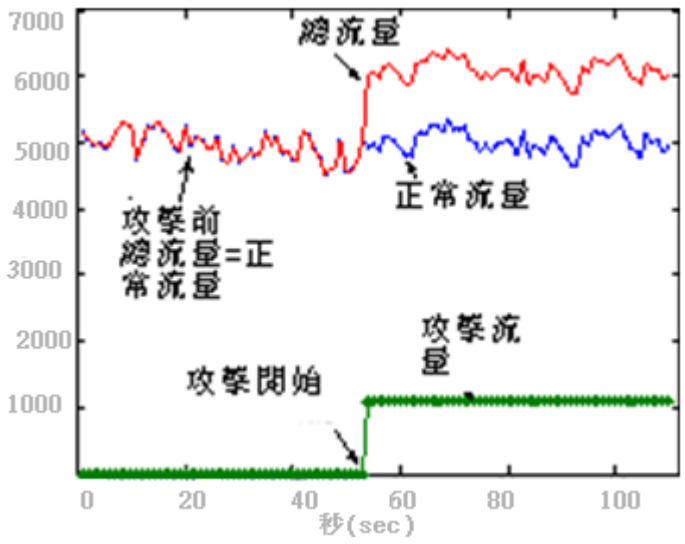

圖 4-17 SYN 攻擊防禦示意圖（輸出埠）

表 8　SYN 攻擊期間系統檢測結果及控制需完成的數量

攻擊警告訓示器	[異常位元組，異常指數] 集	位元組流量控制需完成的數量
1	[108 (SYN) ， 51 (>>3)]	108 (SYN) -- 309 pps

攻擊流從攻擊開始的第 3 秒起就被持續限制在小於原攻擊流的五分一的水準，這時對總流量的影響也僅為增加了不到 4%的流量規模。這說明本系統對較小規模的攻擊仍然具有良好的檢測及控制作用。

CASE5: 混合攻擊測試

接下來評估本系統對網路中的混合攻擊（ICMP、SYN 與埠泛洪攻擊同時進行）的抵禦能力。

圖 4-18 是系統輸入埠的流量示意圖。從圖中可見，ICMP、SYN 與埠泛洪攻擊同時向受害目標發出，三者分別以超過 10000pps 的強度共同迭加於規模約 5000pps 的正常流量，使系統的輸入總流量達到 35000pps（正常流量七倍）以上，如果不對其中的異常流量進行限制，將有可能使網路遭受嚴重的破壞。

表是系統異常檢測器以及流量控制器在發生攻擊期間的資料記錄。從表中可見：攻擊警告訓示器不為 0，即表明系統檢測到網路中存在異常流量；而異常位元組分別為位元組 77 (ICMP) ，位元組 92 (port 25) 和位元組 108 (SYN)，其異常指數均遠大于正常範圍需完成的數量；同時流量控制部分也分別給出各個位元組的流量控制需完成的數量。

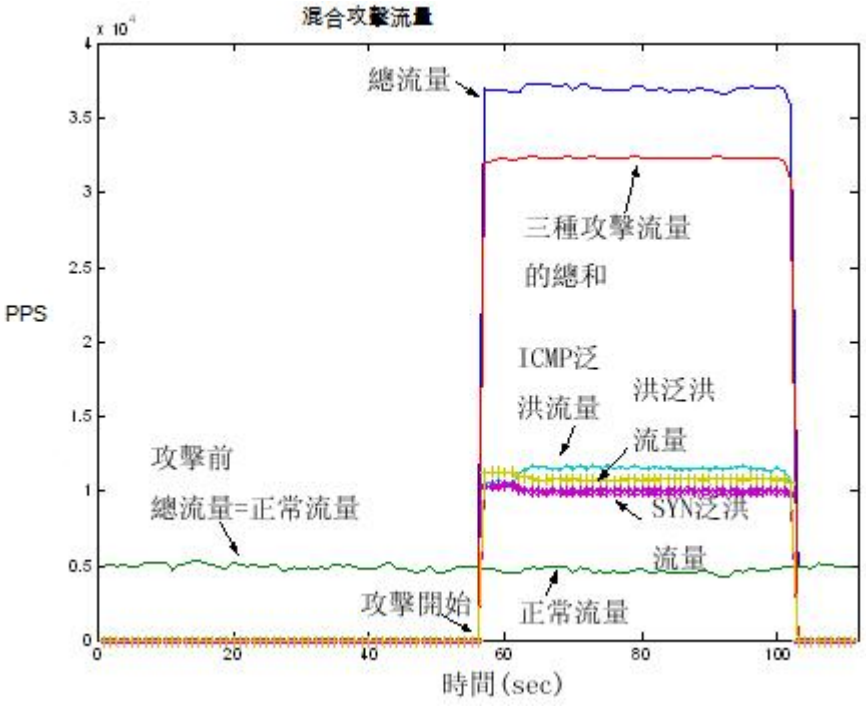

圖 4-18 混合攻擊流量示意圖（輸入埠）

圖 4-18 是本系統攻擊前後出口埠的流量示意圖。從圖中可見，混合攻擊開始後本系統出口的流量在經歷了一個急升（約為正常時候流量規模的五倍）後隨著攻擊流受到控制，總流量迅速回落。這時，大部分的正常流量均能通過本系統，而三種攻擊的 98%以上攻擊流量已經被擋在本防禦系統之前，剩下部分的攻擊流其攻擊力已被極大地削弱。由此可見，本防護系統的檢測控制功能對混合攻擊同樣能作出非常好的防禦作用，在最大程度上保障了系統出口埠的網路以及目標伺服器安全。

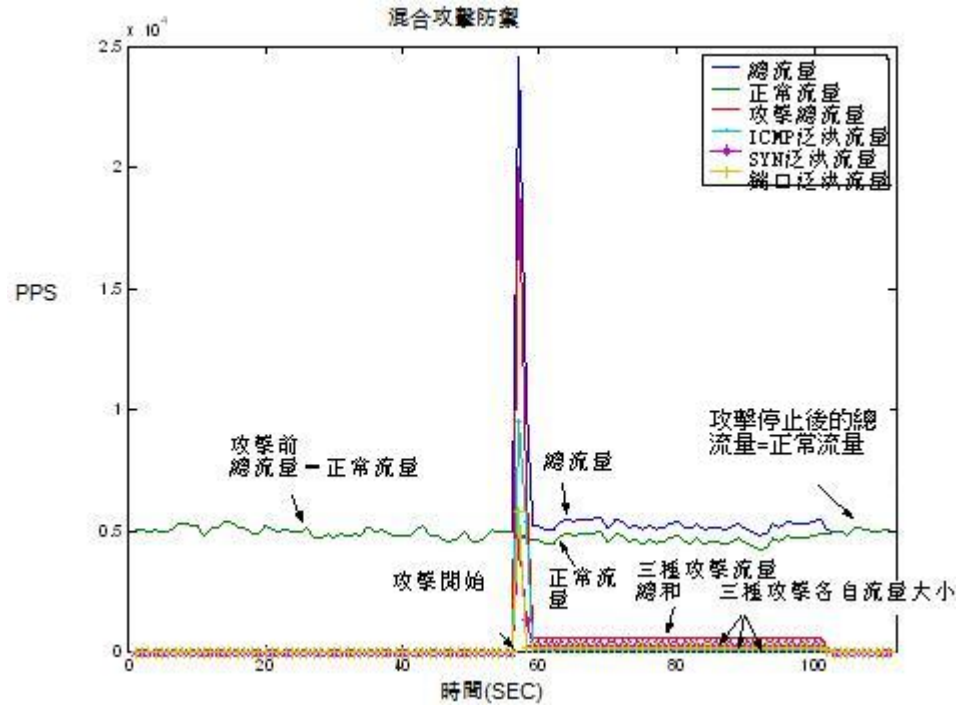

圖 4-19 混合攻擊防禦示意圖（輸出埠）

雖然本系統已經能比較有效地對不同類型及強度的攻擊流進行控制，但存在的不足是，在攻擊發生的初期仍然存在著瞬間的高流量。導致這種情況出現的原因是由於本系統的檢測統計週期間隔為 1 秒。即使攻擊在某一統計週期內啟動，系統也需要等到該統計週期結束後才能檢測到攻擊的發生，同時流量限制措施也要在這個統計週期結束時才能更新。若要改善這種情況就需要縮小統計間隔，然而較短的統計週期也可能會使某些小流量位元組的統計準確率下降。而在實際應

用中，數秒內的高流量所造成的影響還是比較有限的，故此兩相平衡之下，本設計仍然以一秒作為檢測的統計週期。

表 9 SYN 攻擊期間系統檢測結果及控制需完成的數量

攻擊警告訓示器	[異常位元組，異常指數] 集	位元組流量控制需完成的數量
1	[77 (ICMP) ，1256 (>>3)] [92 (port 25) ，1061 (>>3)] [108 (SYN) ，973 (>>3)]	77 (ICMP)--102 pps 92 (port 25)--179 pps 108 (SYN)-- 310 pps

第五章　結論及未來發展方向

5.1 結論

現代網際網路安全仍然面臨眾多威脅。DDoS 攻擊所具有的破壞力強，易於隱藏，特徵較難提取以及難以預先防範等特點，使其越來越成為對網際網路的嚴峻威脅之一。而已有的 DDoS 攻擊檢測或控制系統均仍然存在各種不足，難以非常有效地抗禦 DDoS 攻擊。

在充分研究正常流量以及 DDoS 攻擊流特性的基礎上，本文實現了基於網路處理器 IXP2400 的多位元元組檢測控制系統模組，採用多位元組異常檢測及監控方法監測網路流量，對異常網路流量尤其是 DDoS 攻擊流進行識別並利用權杖桶技術對其作出合適的流量控制。一系列的系統性能評估實驗證明，本系統在令正常流量基本不受影響地通過的同時，能夠非常敏感地對 DDoS 攻擊進行識別並對攻擊流量作出非常有效的控制。通過良好的防禦性能，系統順利的保護系統出口埠網路的網路安全。

5.2 未來發展方向

為了使本系統功能更加有效及完善，準備在下一步工作中作出以下改善：

根據當前網路狀況及流量控制回饋效果來自適應調整各級限速佇列的權杖產生速率。對程式代碼流程結構等進行更好的優化

同時，充分利用 IXP2400 編程的靈活性及易於進行功能擴展的特點，將計畫向系統增加以下功能：

增加人機互動圖形介面模組，以便管理人員可以通過此介面瞭解系統運行狀態並對系統進行線上設定。

未來可以再以網路不同使用時段(尖峰與離峰)為檢測的另一依據，可讓檢測的模型更精準，也可再以不用的統計檢測方式如：格拉布斯準則(Grubbs criterion)、狄克遜準則(Dixon criterion)，改良的演算法來做為多重的檢測，以達更精確的成果。

參考文獻

[1] Garber， L.;“Denial-of-service attacks rip the internet”，Computer，Volume 33， Issue 4， PP.12 – 17， April 2000 .

[2] Rocky K.C. Chang; “Defending against Flooding-based Distributed Denial-of-Service Attacks: A Tutorial”. Communications Magazine，IEEE Volume 40，Issue 10， PP.42 – 51，Oct. 2002 .

[3] Christos Douligeris， Aikaterini Mitrokotsa; “DDoS attacks and defense mechanisms: classification and state-of-the-art”， Computer Networks PP. 643–666， Oct. 2004.

[4] D. Moore， G. Voelker， and S. Savage; “Inferring Internet Denial-of-Service Activity”， Proceedings of the 2001 USENIX Security Symposium， Washington D.C.， pp.9-22，August 2001.

[5] J. Mirkovic， J. Martin， and P. Reiher， “A taxonomy of DDoS attacks and DDoS defense mechanisms”， Technical report 020018.Computer Science Dept.， University of California， Los Angeles.

[6] CERT® Coordination Center; “Trends in Denial of Service Attack Technology” http://www.cert.org/archive/pdf/DoS_trends.pdf， October 2001

[7] Advanced Networking Management Lab; “Distributed Denial of Service Attacks(DDoS) Resources”; http://www.anml.iu.edu/ddos/tools.html; 2001

[8] CERT® Coordination Center; “CERT® Incident Note IN-99-07”， http://www.cert.org/incident_notes/IN-99-07.html， 1999

[9] CERT® Coordination Center; “CERT® Advisory CA-1999-17 Denial-of-Service Tools”; http://www.cert.org/advisories/CA-1999-17.html， 1999

[10] David Dittrich; “The ‘stacheldraht’ distributed denial of service attack tool”，http://staff.washington.edu/dittrich/misc/ stacheldraht.analysis，1999

[11] Phreeon; “blitznet.tgz”; http://www.ussrback.com/distributed.htm

[12] D. Dittrich， G. Weaver， S. Dietrich， N. Long; “The_mstream_Distributed Denial of ervice attack tool”; http://staff.washington.edu/dittrich/misc.mstream.analysis.txt; 2000

[13] J. Jung， B. Krishnamurthy， M. Rabinovich. “Flash Crowds and Denial of Service Attacks: Characterization and Implications for CDNs and Web Sites”. The Eleventh International World Wide Web Conference，Honolulu， Hawaii， PP. 252—262， May 2002.，

[14] Shuyuan Jin; Yeung， D.S. “A Covariance Analysis Model for DDoS Attack Detection”; American Control Conference， 1997. Proceedings of the 1997 Volume 4， 4-6 PP:2313 – 2322，June 1997.

[15] Siaterlis， C.; Maglaris， V.; “Detecting incoming and outgoing DDoS attacks at the edge using a single set of network characteristics”. 2005. ISCC 2005. Proceedings. 10th IEEE Symposium on 27-30， PP:469-475，June 2005

[16] BoonPing Lim; Uddin， Md.S.; “Statistical-Based SYN-Flooding Detection Using Programmable Network Processor”，Decision and Control， 2005 and 2005 European Control Conference. CDC-ECC '05. 44th IEEE Conference on 12-15 PP.7347 – 7351，Dec. 2005 .

[17] “Intel® IXP2400 Network Processor: Flexible， High-Performance Solution for Access and Edge Applications”，http://www.intel.com/design/network/papers/ixp2400.pdf.

[18] “Intel® Internet Exchange Architecture Programmable Network Processors for Today’s Modular Networks”; http://www.intel.com/design/network/papers/intelixa.pdf，2003

[19] “Intel® Internet Exchange Architecture Software Building Blocks Developer’s Manual”. Intel Corporation， November 2003.

[20] “Intel® IXP2400/IXP2800 Network Processor Programmer’s Reference Manual”， Intel Corporation， 2003.

[21].李駿偉、田筱榮、黃世昆，入侵偵測分析方法評估與比較，Communications of the CCISA Vol. 8 No.2 March 2002

[22].Davis，Data Mining Methods for Network Intrusion Detection，STERRY BRUGGER University of California，June 9， 2004
[23]Lee， W.， S. J. Stolfo， and K. W. Mok，A data mining framework for building intrusion detection models，1999 IEEE
[24]Rao X， Dong CX， Yang SQ. An intrusion detection system based on support vector machine. Journal of Software， 2003，14(4): 798~803
[25].Mike Paquette，Intrusion Prevention Bolsters Network Security，Top Layer http://www.toplayer.com/pdf/faq.pdf
[26].朱瑞秋、賴冠州，從入侵偵測到入侵防禦，微電腦傳真雜誌，6 月 2004 年
[27].于振凡，數據的統計處理和解釋-(第二版)，中國標準出版社，中國，2006

[28] Gaeil Ahn; Kiyoung Kim; Jongsoo Jang; “MF (minority first) scheme for defeating distributed denial of service attacks”; Computers and Communication， 2003. (ISCC 2003). Proceedings. Eighth IEEE International Symposium on 2003 Page(s): 1233 - 1238