網路的攻擊與防護機制

張智晴　林盈達

投稿領域：網路安全

國立交通大學資訊科學研究所

新竹市大學路 1001 號

TEL：(03) 5712121　EXT. 56667

FAX：(03) 5712121　EXT. 59263

E-mail：{ace, ydlin}@cis.nctu.edu.tw

主要聯絡人：張智晴

摘要

有鑑於網際網路的快速發展和網路安全的日益重要，我們將焦點放在企業內網路如何在網際網路這個混亂的大環境中確保自己的安全。這篇報告將會介紹現有的各種攻擊方法和防守方法，並分類爲各種典型的攻擊模式，包括監聽、密碼破解、漏洞、掃描、惡意程式碼、阻斷服務和 Social Engineering，以及各種典型的防守模式，包括資料加密、身份認證、存取控制、稽核、監控和掃描，接著歸納出有那些問題目前仍然無法解決，包括未知漏洞、阻斷服務和 Social Engineering，最後做出結論。

關鍵字：網際網路、網路安全、企業內網路、攻擊、防守、監聽、密碼破解、漏洞、掃描、惡意程式碼、病毒、後門程式、阻斷服務、Social Engineering、資料加密、身份認證、數位簽章、存取控制、防火牆、稽核、監控。

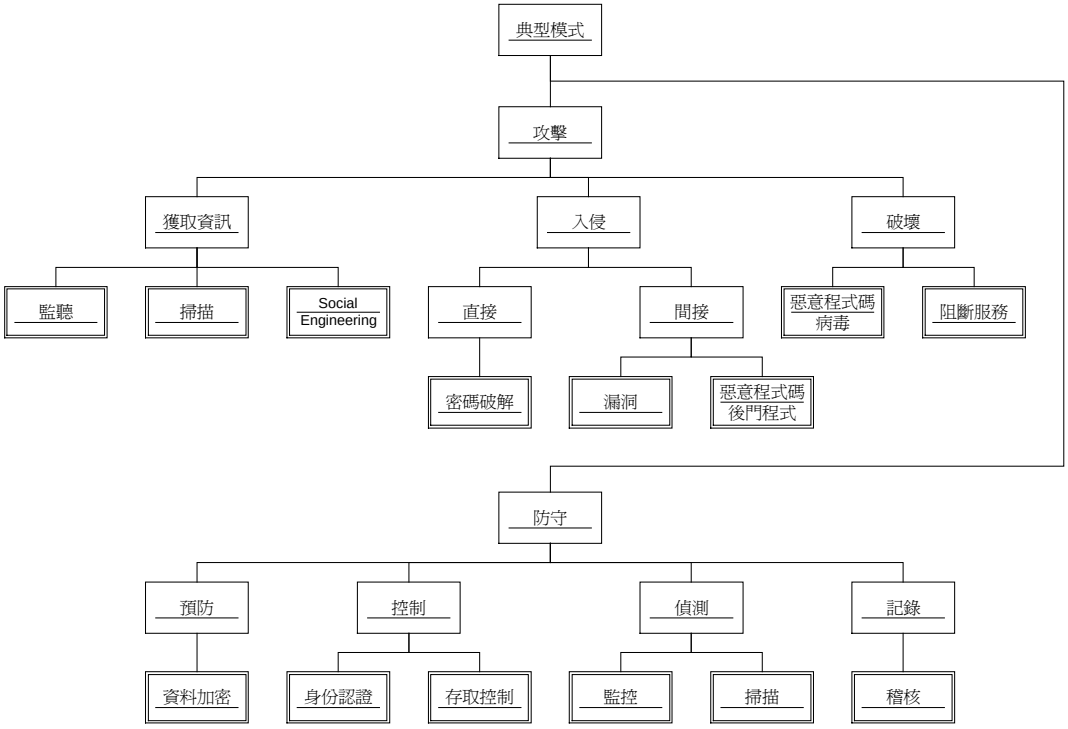
1. 簡介

網路在剛開始的主要應用在研究人員互相交換研究結果或公司內員工資源分享，這時候網路安全

並未受到重視。時至今日，由於資訊科技(IT, Information Technology)和網際網路(Internet)[1]的快速發展，參與網際網路的人數日益增加，網際網路上的組成份子愈來愈複雜，電腦主機上和在網際網路間傳送的資料愈來愈重要，網際網路上的網站提供的服務也愈來愈關鍵(critical)，而網路安全的發展速度卻沒有跟上，常常因爲效率或方便性的考量而被忽略掉。

網路安全可以概分爲三個部分：保護資訊(information)、保護資源(resource)和保護隱密性(privacy)。保護資訊是確保資訊不被未獲授權的使用者取得或竄改。保護資源是確保資源不被未獲授權的使用者使用，這裡的資源可以是網際網路上網站提供的服務或是網路的頻寬。保護隱密性是確保個人在網路上的私密資料和在網路上的行爲資訊不被其他人取得，例如網路上的消費行爲或是瀏覽全球資訊網網站的軌跡。

有鑑於網際網路的快速發展和網路安全的日益重要，我們將焦點放在企業內網路(Intranet)如何在網際網路這個混亂的大環境中確保自己的安全。這篇報告將會介紹現有的各種攻擊方法和防守方法，並分類[1][2]爲各種典型的攻擊模式、典型的防守模式，如圖一中的雙框部分，接著歸納出有那些問題目前仍然無法解決，最後做出結論。



圖一、各種攻擊和防守典型模式的分類樹狀圖

[1]能利用 TCP/IP 在全球各地彼此之間進行通訊的主機的集合。

2. 典型的攻擊模式

爲了確保企業內網路在網際網路的安全，我們首先來了解現今在網際網路中的攻擊方法，以在遭受到攻擊時可以做出適當的反應，甚至在遭受攻擊前就事先預防。在這一節中，我們將介紹現今企業內網路在網際網路中可能遭受到的攻擊，並對這些攻擊分類，歸納爲幾類典型的攻擊模式。我們將攻擊方法分類爲下列七類典型模式：監聽(Monitoring)、密碼破解(Password Cracking)、漏洞(Exploits)、掃描(Scanning)、惡意程式碼(Malicious Code)、阻斷服務(Denial of Service) [3]和 Social Engineering。表一是目前常用攻擊程式和軟體。

攻擊類型	程式或軟體名稱	相關網頁
監聽(網路封包監聽)	Sniffit	http://reptile.rug.ac.be/~coder/sniffit/sniffit.html
	NetXRay	http://www.netxray.co.uk
密碼破解	Crack5.0a	http://www.ja.net/CERT/Software/crack/Crack5.0a.tar.gz
	L0phtCrack	http://www.l0pht.com/l0phtcrack
掃描(遠端掃描)	SATAN	http://www.fish.com/~zen/satan/satan.html
	SAINT	http://www.wwdsi.com/saint
	Nessus	http://www.nessus.org
	NMAP	http://www.insecure.org/nmap
	Internet Scanner	http://www.iss.net
掃描(本機掃描)	Tiger	ftp://net.tamu.edu/pub/security/TAMU
	check.pl	http://opop.nols.com/proggie.html
惡意程式碼(後門程式)	BO2K	http://sourceforge.net/projects/bo2k
	NetBus	http://www.netbus.org
阻斷服務	Smurf	http://www.rootshell.com/archive-j457nxiqi3gq59dv/19971 0/smurf.c.html
	Trinoo	http://staff.washington.edu/dittrich/misc/trinoo.analysis
	TFN	http://staff.washington.edu/dittrich/misc/tfn.analysis
	Stacheldraht	http://staff.washington.edu/dittrich/misc/stacheldraht.anal ysis.txt
	Mstream	http://staff.washington.edu/dittrich/misc/mstream.analysis .txt

表一、目前常用攻擊程式和軟體

一般來說，攻擊依類型可以分爲三類，獲取資訊、入侵和破壞，如圖一所示。獲取資訊是指

取得重要或私密資訊，包括監聽、掃描和 Social Engineering。入侵是指取得系統管理者權限，可以分爲直接和間接兩種方式，直接入侵是指取得系統管理者密碼後直接進入系統，例如密碼破解；間接入侵是指利用其它方式取得系統管理者權限，例如漏洞和惡意程式碼中的後門程式(Backdoor)。破壞是指造成資料的毀損或是讓一個網站的網路服務中斷，如惡意程式碼中的病毒和阻斷服務。

另外，入侵通常可以分成三個步驟，收集資訊、入侵和入侵後的處理。收集資訊是指盡量收集有關目標主機的一切資訊，例如主機網址、主機開放的服務、主機內的使用者帳號(User ID)、甚至是使用者密碼(User Password)或系統管理者密碼。接著是入侵動作，可以是直接以前一步驟取得的系統管理者密碼進入主機或間接利用漏洞進入。成功入侵主機後，接著是最後的處理動作，包括清除系統中有關入侵行爲可能造成的記錄以避免留下犯罪證據和執行後門程式以方便下次的進入。

2.1 監聽

這類型的攻擊是指監聽電腦系統或網路封包以獲取資訊。監聽實質上並沒有進行真正的破壞性攻擊或入侵，但卻通常是攻擊前的準備動作，攻擊者利用監聽來獲取他想攻擊對象的資訊，像是網址、使用者帳號，甚至直接拿到使用者密碼。這個類型的攻擊可以分成兩個子類別：網路封包監聽(Sniffing)和電腦系統監聽(Snooping)。

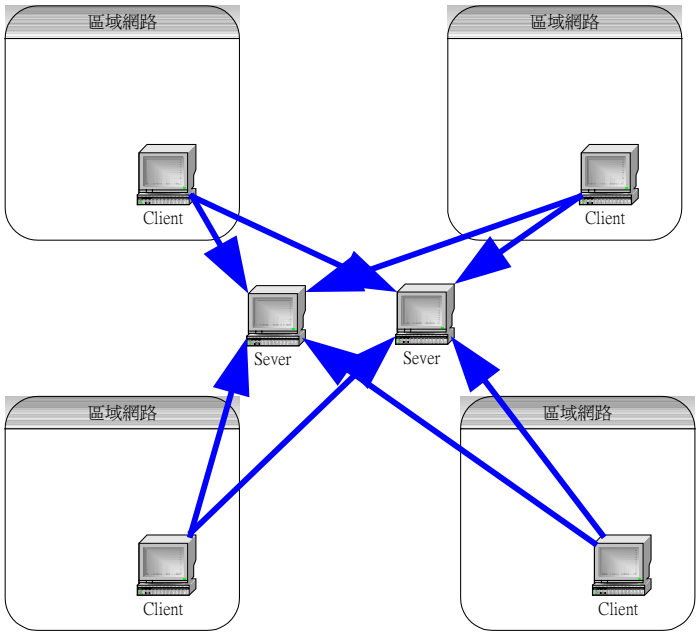
網路封包監聽

網路封包監聽指的是攔截流經你所在主機的封包，取得封包裡的資訊。正常來說，網路主機只會接收目的位址是它的封包，但是可以經由改變網路卡的操作模式來讓它接受所有流經它的封包，例如乙太網路(Ethernet)網路卡的”promiscuous mode”。

這一類型的攻擊程式稱作 Sniffer。Sniffit [4]就是這一類型的程式，它是一個在 Unix 系統下執行的程式，目前最新版本是 0.3.7.beta。它可以針對不同的來源位址、來源埠(port)、目的位址和目的埠監聽封包，並可以選擇將結果記錄下來或是導向其它的終端機(terminal)。

另外，最近 CERT [5]收到一種有關網路封包監聽的新攻擊程式的回報，名稱是 Distributed

Network Sniffer [6]。Distributed Network Sniffer 包括 server 和 client 兩個部分，攻擊者事先入侵網路上的主機並在主機上安裝 client 程式，然後利用 client 程式監聽流經它的封包，分析出使用者帳號跟使用者密碼，最後把這些使用者帳號與使用者密碼傳送給 server 程式。圖二可以說明這個攻擊方法。目前發現被裝置 client 程式的主機皆爲 Linux 作業系統，而 client 是經由 port 21845/udp 傳送使用者帳號跟使用者密碼給 server。這種新的攻擊方式相當有威脅性，只要區域網路中有一部主機被入侵並安裝 client 程式，那整個區域網路的主機都可能被入侵。



圖二、Distributed Network Sniffer

電腦系統監聽

電腦系統監聽是指監聽電腦系統的記憶體、磁碟或其它儲存資料的媒體，以獲取它所儲存的資料。例如監聽電腦系統的記憶體，觀察或記錄使用者按了那些按鍵，以獲得密碼。攻擊者可能利用電腦系統監聽來獲取使用者和其它主機溝通的行爲或資料，以用來入侵其它主機。

這一類型的攻擊程式稱作 Snooper。通常是後門程式的一部分，後門程式我們在接下來惡意程式碼這一類型攻擊中會介紹，並會一併介紹它的電腦系統監聽功能。

2.2 密碼破解

這類型的攻擊是指使用程式或其它方法來破解密碼。破解密碼主要有兩個方式，猜出密碼或是使用暴力法(brute force)一個一個嘗試所有可能試出密碼，如果想用猜的方式來破解密碼，可能還需要一個字典(dictionary)檔。這個密碼可能是 Unix 主機的使用者密碼、用來解開加密資料的密碼或其它用來解密的密碼。這類型的攻擊主要是針對像 Unix 這類使用密碼來確認使用者身份的系統，如果 root[2]的密碼被破解，攻擊者將可以完全接管這台主機，又 Unix 系統通常提供遠端存取的功能，因此攻擊者將可從任何地方控制這台主機。

這類型的攻擊程式相當多，如果是要破解系統使用者密碼的程式，通常需要一個儲存著使用者帳號和加密過的使用者密碼的系統檔案，例如 Unix 系統的 passwd 和 Windows NT 系統的 SAM，破解程式就利用這個系統檔案來猜或試密碼。如果不利用這個檔案而直接連上那個主機去試密碼的話，可能會被那台主機記錄下你的位置，而且通常系統只允許一定次數的嘗試錯誤。另外，破解出密碼的時間由執行破解程式的電腦速度和密碼複雜度決定，電腦速度愈快和密碼複雜度愈低，破解所需的時間就愈短。

L0phtCrack [7]就是這類型的程式，它可以破解 Windows NT 系統的密碼，是一個在 Windows 系統下執行的程式，目前最新版本是 2.5.2。它除了可以使用 SAM 系統檔案取得加密過的使用者密碼來破解 Windows NT 系統的密碼外，還可以使用另外兩種方式取得加密過的使用者密碼，從系統登錄(registry)和攔截網路上的 SAM 封包。系統登錄裡儲存著加密過的使用者密碼，L0phtCrack 可以從系統登錄取得加密過的使用者密碼。另外，如果使用者不是從 PDC[3]登錄 NT 網域(domain)，則會送出 SAM 封包給 PDC 來驗證身份，而 L0phtCrack 可以攔截所有經過執行 L0phtCrack 主機的 SAM 封包，從 SAM 封包取出加密過的使用者密碼。

2.3 漏洞

漏洞是指程式或軟體在設計、實作或操作上的錯誤，而被攻擊者用來獲得資訊、取得使用者權限、取得系統管理者權限或破壞系統。因爲世界上有太多種類和數量的程式或軟體，而每個程式或軟體在設計或實作上都可能發生錯誤，就算程式或軟體在設計或實作上沒有錯誤，使用者也

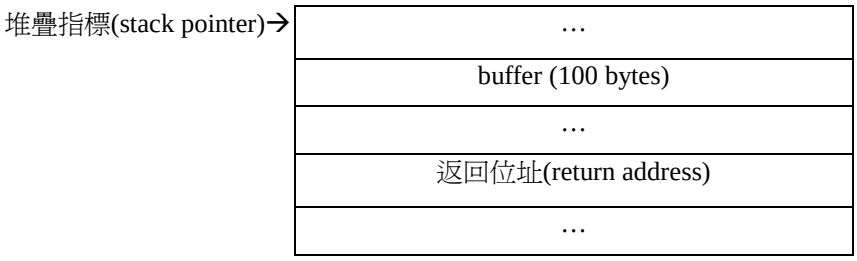
[2] Unix 系統的系統管理者。
[3] Primary Domain Controller，一個 NT 網域中只能有一台 PDC。

可能操作錯誤，所以這個類型的數量相當龐大。

緩衝區溢位(buffer overflow)是程式或軟體在實做上最常發生的錯誤，也是最多漏洞產生的原因。緩衝區溢位的發生原因是放超過緩衝區大小的資料到緩衝區，例如放 101 個位元組(byte)的資料到宣告大小爲 100 個位元組的陣列(array)上，造成多出來的資料會覆蓋到其它變數上，絕大多數的狀況是程式發生錯誤而結束。但是如果適當的放入資料，就可以利用緩衝區溢位來執行自己的程式。底下是一個例子，一個函數(function)如下：

```
void fuction() {
    …
    char buffer[100];
    …
}
```

當這函數被呼叫時，作業系統會爲這個函數建立一個堆疊(stack)，如下：



在這個例子中，我們只要放入適當大小的資料，資料中包含我們要執行的程式碼，資料的大小要剛好把返回位址覆蓋成指向我們所放入想執行程式的位址。當這個程式執行完這個函數時，正常狀況下會返回呼叫它的函數，可是因爲緩衝區溢位和放入適當資料的關係，而會執行我們放入的程式。

這個類型的程式有兩個主要的子類別：遠端漏洞(Remote Exploits)和本機漏洞(Local Exploits)。

遠端漏洞

這個子類別的攻擊是指攻擊者利用漏洞來入侵遠端電腦主機取得未授權給予之資訊、使用者

權限，或是系統管理者權限，而攻擊者本身沒有遠端電腦主機的使用者帳號。因爲要攻擊的對象是遠端電腦主機，所以這個子類別的漏洞通常是發生在提供網路服務的程式或軟體上。例如 sendmail [8]，它是 Unix 系統上最常被使用的電子郵件伺服器(mail server)，也是最著名的遠端漏洞程式，sendmail 目前最新的版本是 8.11.0，到現在更新了相當多次，之前的各個版本有著各種漏洞，大部分是我們介紹過的緩衝區溢位，導致攻擊者可以以系統管理者的權限執行他的程式。

至於最近的遠端漏洞，有 Redhat 6.2 Linux 作業系統操作錯誤[9]造成的漏洞和 wu-ftpd 的類似緩衝區溢位漏洞[10]。Redhat 6.2 中有一個套件名稱爲 Piranha，它的功能是全球資訊網網站叢集(web clustering)，它另外包含了網頁界面(web-based GUI)軟體來管理全球資訊網網站叢集，在安裝這套軟體後，這個軟體會產生一個名爲 piranha 的預設使用者帳號和預設的密碼 q，如果系統管理者安裝了這套軟體卻沒有更改 piranha 這個使用者帳號的預設密碼 q，攻擊者就可以利這個使用者帳號來執行任何程式，而目前大部分安裝 Redhat 的人都是選擇完整安裝(full install)，卻不知道安裝了這個軟體而沒有更改預設密碼，造成這個因操作錯誤產生的漏洞；Wu-ftpd 是 Unix 系統上最常被使用的檔案傳輸協定伺服器(FTP Server, File Transfer Protocol Server)，最近被發現一個類似緩衝區溢位的漏洞，它是發生在實作 site exec 這是指令中的函式*printf()，攻擊者可以利用 format string 來覆蓋返回位址，做到類似緩衝區溢位的效果。表二是可取得系管理者權限的遠端漏洞(資料來源：SecurityFocus [11])。

漏洞名稱	**軟體或程式名稱**	**版本**	**漏洞產生原因**
phf Remote Command Execution Vulnerability	Apache Group Apache	1.0.3	Input Validation Error
Multiple Vendor BIND (NXT Oveflow) Vulnerabilities	ISC BIND	8.2.1	Buffer Overflow
MS IIS FrontPage 98 Extensions Buffer Overflow Vulnerability	Microsoft IIS	4.0	Buffer Overflow
Univ. Of Washington imapd Buffer Overflow Vulnerability	University of Washington imapd	12.264	Buffer Overflow
ProFTPD Remote Buffer Overflow	Professional FTP proftpd	1.2pre5	Buffer Overflow
Berkeley Sendmail Daemon Mode Vulnerability	Eric Allman Sendmail	8.8.2	Input Validation Error
RedHat Piranha Virtual Server	RedHat Linux	6.2	Configuration Error

Package Default Account and Password Vulnerability			
Wu-Ftpd Remote Format String Stack Overwrite Vulnerability	Washington University wu-ftpd	2.6	Input Validation Error

表二、可取得系管理者權限的遠端漏洞

另外，在這個子類別中又有一個主要的子類別：The protocol-based attack。網際網路所使用的通訊協定是 TCP/IP，所以網際網路上的主機都必須實作 TCP/IP 這個通訊協定，才可以和網際網路上的其它主機溝通。而這個子類別就是利用各種作業系統實作 TCP/IP 的錯誤或 TCP/IP 本身的設計不良或定義不清楚而產生的漏洞，來攻擊遠端主機，例如 IP 偽造(IP spoofing)，可以用來攻擊 Address-based authentication[4]的系統，攻擊者偽造 IP 來源位址成系統接受的來源位址，以進入系統。這個子類別大部分是破壞性的攻擊，所以我們會在接下來的阻斷服務這一類型中介紹。

本機漏洞

這個子類別的攻擊是指，攻擊者利用漏洞取得未授權給予之資訊或取得更高的權限，例如系統管理者權限，而攻擊者原先在這台主機上已有使用者帳號。這個子類別的漏洞通常是發生在特權程式(privileged program)[5]的設計或實作上的錯誤。

Xterm 是 X Window 系統[6]下的一個終端機模擬程式，在早期的版本曾被發現存在本機漏洞[12]，是之前提過的緩衝區溢位，如果系統把 Xterm 安裝成 SUID (Set User ID)[7] root 檔案，攻擊者就可以利用這個漏洞取得系統管理者權限。

2.4 掃描

這個類型的攻擊是指掃描電腦系統以獲取資訊。掃描和監聽一樣，實質上並沒有進行真正的破壞性攻擊或入侵，但卻通常是攻擊前的準備動作，攻擊者利用掃描來獲取他想攻擊對象的資訊，像是開放那些服務、提供服務的程式，甚至利用已發現的漏洞樣本(pattern)做比對直接找出

[4] 以 IP 來源位址來驗證身份的系統，在接下來的防守方法中會介紹。
[5] 程式執行時用以系統管理者的權限執行。
[6] Unix 系統下的圖型使用者界面。
[7] 執行時以檔案擁有者的權限執行。

漏洞。這個類型的攻擊可以分成兩個子類別：遠端掃描(Remote Scanning)和本機掃描(Local Scanning)。

遠端掃描

這個子類別的攻擊是指攻擊者掃描遠端主機或網路以獲取資訊。這些資訊包括主機名稱(host name)、主機開放的服務、提供服務的程式和可能的遠端漏洞。這個類型的代表性攻擊程式是 Security Administrator's Tool for Analyzing Networks (SATAN) [13]，它是在 Unix 系統下執行的程式，最新版本是 1995 年的 1.1.1，後來就沒有新版本出現。比較新的遠端掃描程式有 Security Administrator's Integrated Network Tool (SAINT) [14]，它可以說是 SATAN 更新及加強版本，目前最新版本爲 2.1.2，這是一個在 Unix 系統下執行的遠端掃描程式，使用主從(client/server)架構，client 端是以 WWW 爲界面。

本機掃描

這個子類別的攻擊是指攻擊者掃描本地電腦主機以獲取資訊。這些資訊包括權限有問題的重要系統檔案、有問題的特權程式和可能的本機漏洞。這個類型的代表性攻擊程式是 COPS [15]，它是在 Unix 系統下執行的程式，不過它一直沒有新版本出現。TIGER [16]是另一個本機掃描程式，也是在 Unix 系統下執行，最新版本是 2.2.4p1，它還有在維護。

2.5 惡意程式碼

這個類型的攻擊是指攻擊者透過外部裝置(external device)和網路把惡意程式碼安裝到系統內，外部裝置可能是軟碟機、光碟機，抽取式硬碟或其它可攜式媒體的裝置。這個類型的攻擊通常是攻擊者成功入侵後做的後續動作，可以分成兩個子類別：病毒(Virus)和後門程式(Backdoor)。

病毒

病毒的有兩特性，自我複製性(self-replicating)和破壞性(destruct)[8]，這個子類別的攻擊就是把病毒安裝到系統內，利用病毒的特性破壞系統和感染其它系統。

最有名的病毒就是世界上第一位網際網路駭客 Rober T. Morris Jr.所寫的網際網路害蟲(Internet Worm) [17]，網際網路害蟲的攻擊行爲其實很簡單，就是複製，以複製同時做到感染和破壞的目的。它是從一台網路主機開始，第一步先尋找這台主機是否有對外的網路連結，如果有的話，就複製自己然後送到對外連結的網路主機，第二步則是在所在主機上複製自己，增加自己的數量，就這樣一直重覆這兩個步驟。當網際網路害蟲的數量多到系統無法處理時，系統就會無法運作而當機。

最近比較熱門的病毒是情書害蟲(Love Letter Worm) [18]，這是一個新型態的病毒，它是用 VBScript 寫成的，主要是利用電子郵件來感染被害主機，它並不會破壞被害主機，但由於大量的主機被感染，造成網路頻寬的浪費。它的主要的攻擊行爲是利用 Microsoft outlook 中通訊錄(address book)的資料來傳送自己，以感染其它主機。

後門程式

這個子類別的攻擊通常是攻擊者在入侵成功後，爲了方便下次入侵而安裝的程式。早期的後門程式是爲了讓攻擊者方便下次入侵，通常是安裝在 Unix 作業系統，最近的後門程式則出現在 Windows 作業系統，而且具有完全控制系統的能力。例如 Back Orifice 2000 (BO2K) [19]，這是一個 Windows 環境下的後門程式，它可以透過網路完全控制被安裝 BO2K 的主機，連線方式可以透過 TCP 或 UDP，並提供密碼認證。它提供了檔案傳輸的功能及之前提過的電腦系統監聽功能，可以記錄下主機上使用者按了那些按鍵。另外，它還可以加入附加程式(plug-in)來增加他的功能，例如被害主機連上網路後就送一封電子郵件給你或某個信箱。

2.6 阻斷服務

這個類型攻擊的目的並不是要入侵系統或是取得資訊，而是阻斷被害主機的某種服務，使得

[8] 較嚴謹的定義是將病毒定義成片段程式碼，Worm 則是指完整程式。而在這篇報告中的病毒指的是一個攻擊的分類。

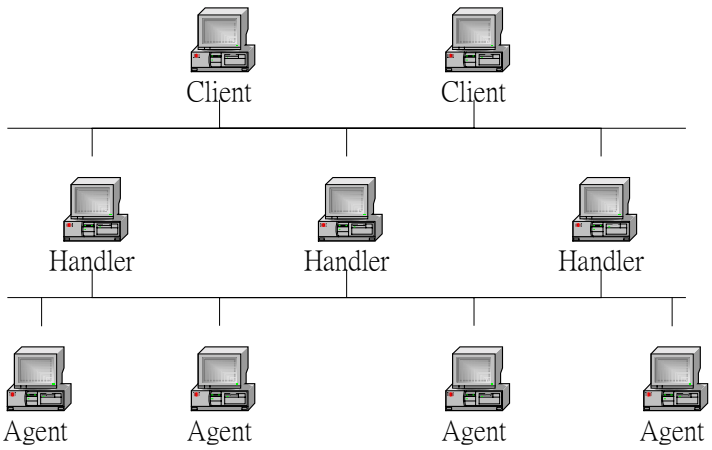
正常使用者無法接受網路主機所提供的服務。這個類型的攻擊有很大部分是從系統漏洞這個攻擊類型中獨立出來的，尤其是遠端漏洞和它的子類別 The protocol-based attack。

這一類型的攻擊主要是把稀少的資源用盡，讓服務無法繼續。例如 TCP 同步訊號洪水型攻擊(TCP SYN flood attack) [20]是把被害主機的等待佇列填滿，而 ICMP 洪水型攻擊(ICMP echo reply flood attack)是把被害主機的網路頻寬用盡。

TCP 建立連線需要三個步驟(three-way handshake)，攻擊者對被害者發出連續的連線要求(SYN 封包)，並將這些 SYN 封包填入不存在或不正確的來源位址，被害者接著會送出 SYN ACK 封包並等待 ACK 封包，但是由於來源是不存在或不正確，所以被害者不可能收到要求端的 ACK 封包，造成被害者的等候佇列被填滿而無法再接受建立連線的要求。

ICMP 洪水型攻擊指攻擊者產生大量的 ICMP echo request 封包(可能偽造來源位址以避免被追蹤)給被害者，被害者會回應等量的 ICMP echo reply 封包，使被害者的網路擁塞甚至中斷。

最近出現一種有關阻斷服務攻擊的新攻擊模式，分散式阻斷服務攻擊(DDoS, Distributed DoS)。分散式阻斷服務攻擊可以用圖三來解釋，攻擊者從 client 端控制 handler，每個 handler 控制許多 agent，因此攻擊者可以同時命令多個 agent 來對被害者做大量的攻擊。而且 client 與 handler 間的溝通是經過加密的。



圖三、DDoS 多層架構

Trinoo [21]是一個主從式架構的分散式阻斷服務攻擊程式，攻擊方式是 UDP 洪水型攻擊(UDP flood attacks)，攻擊者送出大量 UDP 封包(可能偽造來源位址以避免被追蹤)給被害者，使被害者的網路擁塞甚至中斷。一個 Trinoo 攻擊網路包含數個主控端(masters)與數量更多的守護神常駐程

式(daemons)。攻擊者首先連接上主控端，下達攻擊命令(如攻擊目標的 IP 位址，何時發動攻擊和其它參數)。主控端獲得攻擊命令後，會連接上所有守護神常駐程式，由守護神常駐程式來做真正攻擊。

以下是攻擊的步驟：

1.攻擊者連接主控端：利用連接埠 27665/TCP。

2.主控端連接守護神常駐程式：利用連接埠 27444/UDP。

3.守護神常駐程式回應主控端：利用連接埠 31335/UDP。

4.守護神常駐程式開始攻擊被害者：開始 UDP 洪水型攻擊。

其它的分散式阻斷服務攻擊程式還有 TFN [22]和 TFN2K。這些攻擊程式的架構都很類似，主要差別在提供的攻擊種類多寡。

2.7 Social Engineering

這個類型的是指不透過電腦或網路的攻擊行爲。一個 Social Engineering 的例子是攻擊者自稱是系統管理者，發電子郵件或打電話給使用者，要求使用者提供密碼，以便測試程式或其它理由。其它像是躲在使用者背後偷看他的密碼也屬於 Social Engineering。

3. 典型的防守模式

在了解現有的攻擊方法後，接著我們會介紹現有的防守方法，以決定要使用那些防守方法來增加系統安全。在這一節中，我們將介紹現今企業內網路在網際網路中可以使用的防守方法，並對這些防守方法分類，歸納爲幾類典型的防守模式。我們將防守方法分類爲下列六類典型模式：資料加密(Data Encryption)、身份認證(Authentication)、存取控制(Access Control)、稽核(Auditing)、監控(Monitoring)和掃描(Scanning)。

這六類典型防守模式依防守特性可以分爲四個類別，預防、控制、偵測和記錄，如圖一所示。預防是指可以預防攻擊者攻擊，如資料加密。控制包括身份認證和存取控制，可以控制未授權的使用者無法取得未授權的權限。偵測是指可以偵制到攻擊，如監控和掃描。記錄是指可以記錄下

受到攻擊後留下的資訊，可以用來追蹤攻擊者，如稽核。表三是目前常用防守程式和軟體。

防守類型	程式或軟體名稱	相關網頁
資料加密	PGP	http://web.mit.edu/network/pgp.html
	SSH	http://www.ssh.org
存取控制	Firewall-1	http://www.checkpoint.com
	ipchains	http://netfilter.filewatcher.org/ipchains
	TCP Wrappers	ftp://ftp.porcupine.org/pub/security/index.html
	portmap	ftp://ftp.porcupine.org/pub/security/index.html
	xinetd	http://synack.net/xinetd
監控	Tripwire	http://www.tripwiresecurity.com
	RealSecure	http://www.iss.net
掃描	Pc-cillin	http://www.trend.com.tw

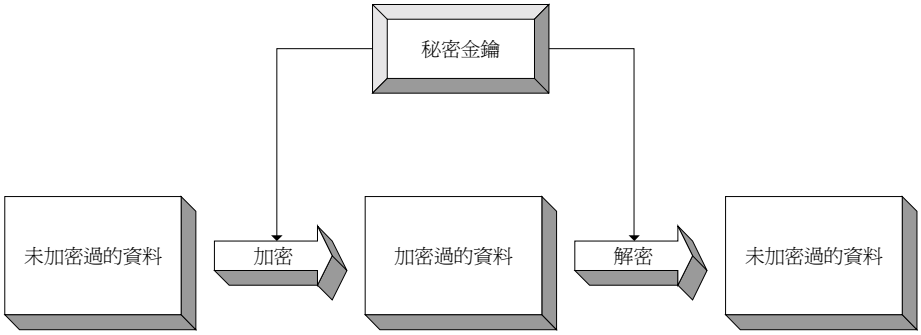
表三、目前常用防守程式和軟體。

3.1 資料加密

資料加密是將資料加密，使得攻擊者即使取得加密過的資料，也無法獲取正確的資料內容，所以資料加密可以保護資料，防止監聽攻擊。這個防守類型可以分爲兩個子類別：對稱式加密(Symmetric Encryption)和非對稱式加密(Asymmetric Encryption)。

3.1.1 對稱式加密

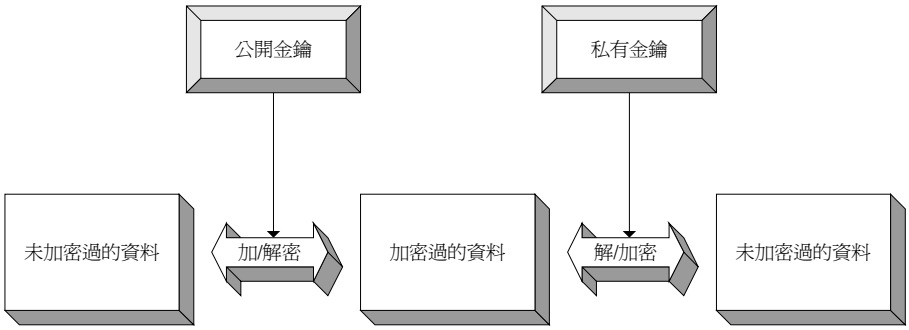
對稱式加密使用同一隻秘密金鑰(secret key)來加密和解密。圖四說明了對稱式加密的運作方式，傳送者(sender)用秘密金鑰將未加密過的資料(plaintext)加密爲加密過的資料(ciphertext)，然後傳送給接收者(receiver)，接收者收到加密過的資料後，用同一隻秘密金鑰將加密過的資料解密爲未加密過的資料，所以使用對稱式加密傳送者和接收者必須要事先知道同一隻秘密金鑰。DES (Data Encryption Standard) [23]和 IDEA (International Data Encryption Algorithm) [24]就是這個種類的加密演算法。



圖四、對稱式加密

3.1.1 非對稱式加密

非對稱式加密分別使用不同金鑰來加密和解密，每個使用者有一對金鑰，公開金鑰(public key)和私有金鑰(private key)，如同字面的意思，公開金鑰可以公開給每個使用者知道。圖五說明了非對稱式加密的運作方式[9]，傳送者用接收者的公開金鑰將未加密過的資料加密爲加密過的資料，然後傳送給接收者，接收者收到加密過的資料後，用自己的私有金鑰將加密過的資料解密爲未加密過的資料，所以使用非對稱式加密傳送者和接收者不須知道對方的私有金鑰。RSA (Rivest, Shamir and Adleman) [25]和 Diffie-Hellman [26]就是這個種類的加密演算法。



圖五、非對稱式加密

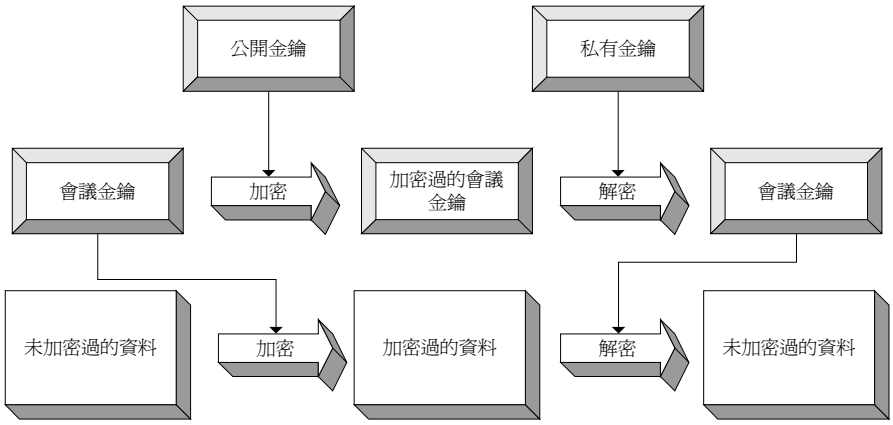
非對稱式加密改善了對稱式加密必須事先知道同一隻秘密金鑰的缺點，可是由於非對稱式加密在加密和解密過程需要位元數很大的數值運算，造成它的加密和解密速度比對稱式加密慢。表四是對稱式加密和非對稱式加密的比較。

	使用同一隻金鑰	**加/解密效率**
對稱式加密	是	較好
非對稱式加密	否	較差

[9] 也可以使用私有金鑰加密，而用公開金鑰解密，但這樣的應用較少見。

表四、對稱式加密和非對稱式加密的比較

目前比較多的應用是結合對稱式加密和非對稱式加密，圖六說明這個方式，傳送者先隨機產生一隻秘密金鑰稱爲會議金鑰(session key)，然後用會議金鑰將未加密過的資料加密爲加密過的資料，另外用接收者的公開金鑰將會議金鑰加密，接著把加密過的會議金鑰和加密過的資料一起傳送給接收者，接收者收到後，先用自己的私有金鑰將加密過的會議金鑰解密，再用會議金鑰將加密過的資料解密爲未加密過的資料。這個方式可以結合對稱式加密和非對稱式加密的優點，傳送者和接收者不須事先知道同一隻秘密金鑰，且因爲用非對稱式加密來加密會議金鑰和用對稱式加密來加密資料，比用非對稱式加密來加密資料來得有效率。



圖六、結合對稱式加密和非對稱式加密

這一類型的程式相當多。例如相當常用的信件加密軟體 PGP (Pretty Good Privacy) [27]和 SSH(Secure Shell) [28]。PGP 同時使用了對稱式加密的 IDEA 和非對稱式加密的 RSA 或 Diffie-Hellman，可以確保電子郵件的真實內容不被攻擊者獲取；SSH 使用 client/server 架構和利用非對稱式加密的 RSA 來做認證，可以對 TCP 連線加密，防止監聽攻擊。

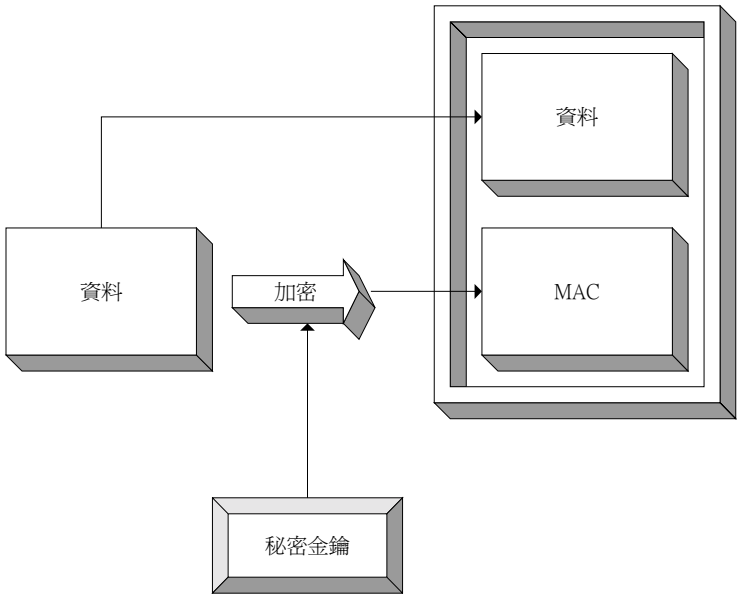
3.2 身份認證

身份認證是用來判斷某個身份的確實性，例如使用者、網路主機、檔案或資料，確認身份後，系統才可以依不同的身份給予不同的權限。確認身份的方法很多，分別可以做到不同程度的身份確認。身份認證可以根據三個種類的方法來達成：秘密資訊(secret information)、擁有物件(possession of object)跟特徵(characteristic)。

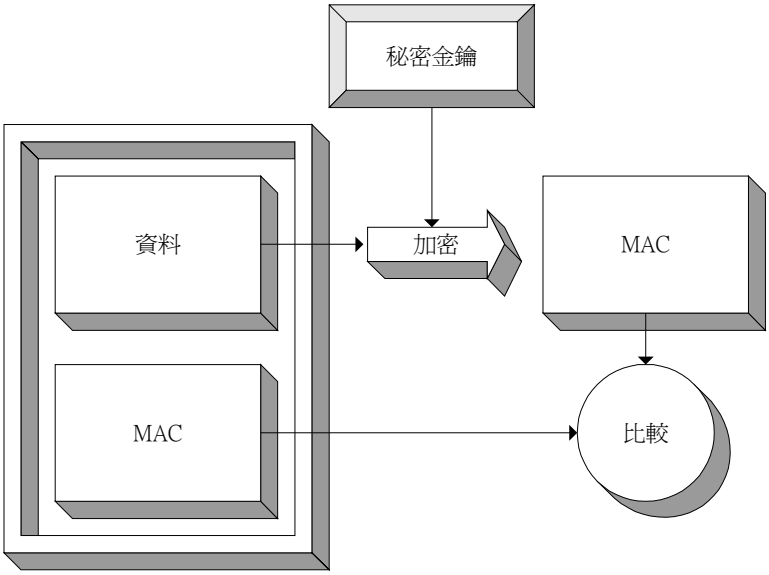
根據秘密資訊來做身份認證的方法有密碼和加密。密碼是最常被用來確認使用者身份的方

法，但它可能會遭受到攻擊者使用破解密碼的攻擊方式或監聽攻擊取得密碼，所以密碼做到的身份確認程度並不是相當高。加密主要是用來確認資料擁有者的身份，如同前一節如介紹的，可以分成兩種方式，對稱式加密和非對稱式加密。

使用對稱式加密來做身份確認的雙方都必須知道同一隻秘密金鑰，圖七和圖八說明了以對稱式加密來做身份認證的運作方式，圖七是傳送者傳送資料前的加密部分，圖八是接收者收到資料後的驗證部分。傳送者用秘密金鑰將資料加密產生 MAC (Message Authentication Code)，接著將資料和 MAC 一起傳送給接收者，接收者收到後，用相同的秘密金鑰將收到的資料加密產生 MAC，然後把產生的 MAC 跟收到的 MAC 比較，如果兩個 MAC 相同，就可以確認這個資料的擁有者身份。



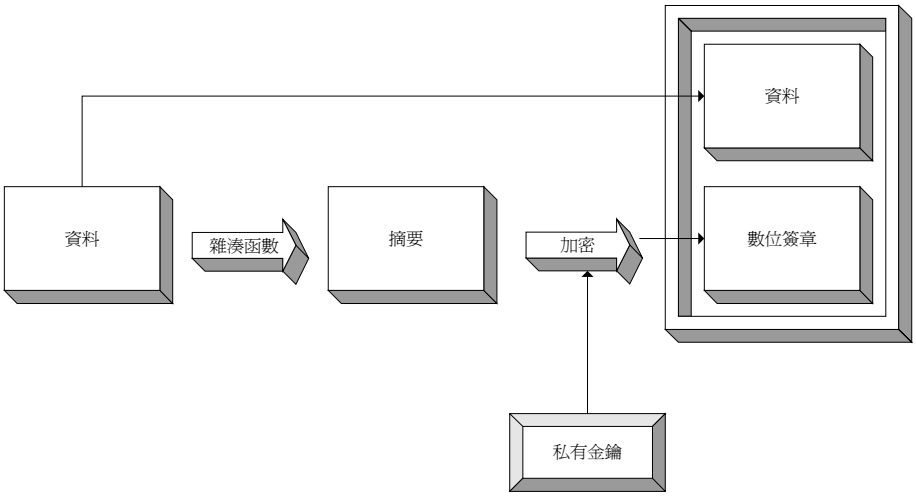
圖七、以對稱式加密來確認身份加密部分



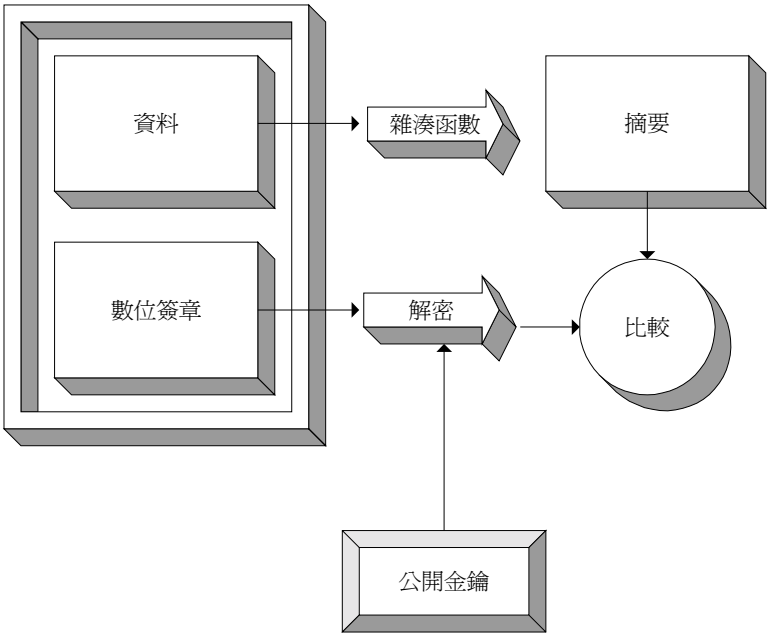
圖八、以對稱式加密來確認身份驗證部分

以非對稱式加密來做身份認證就是愈來愈常用的數位簽章(Digital Signature) [29]，圖九和圖十說明了以數位簽章來做身份確認的運作方式，圖九是傳送者傳送資料前的加簽部分，圖十是接收者收到資料後的驗證部分。傳送者首先將資料經過雜湊函數[10]運算產生摘要(digest)，接著再以自己的私有金鑰將摘要加密產生數位簽章，然後將資料和數位簽章一起送給接收者，接收者收到後，分別把收到的資料經過雜湊函數運算產生摘要，和用傳送者的公開金鑰將收到的數位簽章解密爲摘要，然後這兩個摘要比較，如果兩個摘要相同，就可以確認這個資料的擁有者身份。數位簽章除了可以確認資料的擁有者身份外，還可以確認資料的完整性(integrity)和傳送者的不可否認性(non-repudiation)。因爲資料經過雜湊函數產生摘要，如果在傳送過程中資料有被改變，接收者將產生不相同的摘要，因此可以確認資料的完整性。另外，由於傳送者是用自己的私有金鑰加密摘要產生數位簽章，別人沒辦法產生相同的數位簽章，所以他無法否認這個數位簽章不是他產生的。

[10]資料經過雜湊函數運算後會產生唯一的摘要，MD5 [30]和 SHA (Secure Hash Algorithm) [31]就是這個種類的演算法。



圖九、以數位簽章來確認身份加簽部分



圖十、以數位簽章來確認身份驗證部分

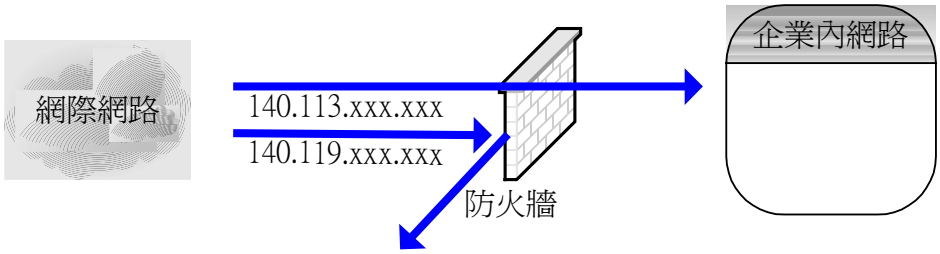
智慧卡(Smart Card)是根據擁有物件來做身份認證的方法。根據特徵來做身份確認的方法有聲音、指紋等，另外，之前提到的 Address-based authentication 系統就是屬於這個種類，它是靠 IP 來源位址來確認身份，但是目前 IP 來源位址很容易被偽造[11]，因此這個方式做到的身份確認程度並不是相當高

3.3 存取控制

[11] 這個缺點將在新版本的 IP (IPV6 [32])中改善。

存取控制是利用前面提到的確認身份方法獲知使用者的確實性後，根據不同身份給予不同的存取控制，控制使用者存取系統資源或資料。Unix 系統和 Windows NT 系統都有類似的機制，它們使用密碼來確認使用者身份，然後給予不同的存取控制權限。

防火牆(firewall) [33]的過濾封包(packet filtering)功能也是屬於這個類型的防守方法。圖十一可以說明防火牆如何做到存取控制，防火牆可以根據不同來源位址、來源埠、目的位址和目的埠來決定是否要讓這個封包通過，如圖十一所示，它讓來源位址開頭爲 114.113 的封包通過，而不讓來源位址開頭爲 114.119 的封包通過。防火牆通常被放置於網際網路和企業內網路間，這樣才可以過濾所有網際網路和企業內網路間的通訊。



圖十一、防火牆的過濾封包功能

3.4 稽核

這一類型的防守是指把系統中和安全有關的事件(security-related event)記錄下來。例如使用者登入來源與登入系統失敗次數及各種重要的網路活動。當遭受到攻擊者攻擊時，這個資料可以用來幫助調查攻擊者的來源，也就是一種證據收集，以試著追蹤攻擊者；或是用來幫助分析攻擊者攻擊的方法，以思考可能的方式來預防下次的攻擊。

現在的作業系統都有提供稽核的功能。例如 Unix 系統的 wtmp 系統檔案，這個檔案會記錄使用者登入與登出的歷史記錄；Windows NT 系統則是可以使用 Event Viewer 來檢視系統中各個重要記錄。

3.5 監控

這一類型的防守是指監控系統或網路是否有異常的活動，例如某個使用者持續的登入失敗，以偵測出攻擊者的攻擊。當偵測到攻擊者的攻擊時，系統可以做出以下反應：1.呼叫系統管理者，

可能是送出電子郵件、呼叫傳呼機或是發出警告聲音。2.停止系統或系統的某項服務，讓損害降到最低。3.試著追蹤攻擊者，系統可以利用比較已知的攻擊特徵(attack signature)，來得知攻擊者攻擊的方法，以試圖追蹤攻擊者。

這個類型的程式可以分成兩個子類別：網路監控程式(Network-based monitor)和主機監控程式(Host-based monitor)。網路監控程式可以監控網路中主機是否有異常的網路活動，它藉著開啓網路卡的 promiscuous mode 來攔截網路中的封包，分析這些封包及流量對網路內主機是否有不正常影響，然後做出適常的反應。網路監控程式可以偵測部分阻斷服務攻擊，例如阻斷服務攻擊中的 TCP 同步訊號洪水型攻擊，網路監控程式可以監控網路中的 SYN 封包，如果發現 SYN 封包的來源地位是不合法的，它就會藉著送一個 RST 封包[12]給網路中被攻擊的主機，讓被害主機放棄等待不可能的回應。

主機監控程式可以監控主機對外的網路活動和內部的異常行爲，例如外部主機對內的連線要求、使用者登入狀況、系統管理員的活動和檔案系統等，如果偵測到不正常的活動，主機監控程式就會做出適當的反應。

Intrusion Detecting System (IDS) [34]和 Tripwire [35]就是屬於這個類型的防守程式。Tripwire 會把系統中重要檔案經過雜湊函數處理，並將結果存在資料庫，然後 Tripwire 會定期檢查這些重要檔案，把這些重要檔案經過雜湊函數處理的結果和資料庫中的結果比較，如果這些重要檔案有被修改，比較結果將會不同，因此 Tripwire 可以用來監控系統中重要檔案。

3.6 掃描

這裡的掃描和攻擊模式中的掃描是不一樣的，這裡的掃描指的是使用已知的樣本，來掃描系統內是否有惡意程式碼，也就是病毒或後門程式。一般所謂的防毒軟體就是屬於這個種類。由於掃描程式是靠已知樣本來找到惡意程式碼，所以使用者必須時常更新已知樣本，也就是防毒軟體中的病毒碼，掃描程式才能發現較新的惡意程式碼。

[12] 要求接收端重設(reset)TCP 連線。

4. 無法解決的問題

表五是典型的攻擊(列)對典型的防守(欄)表。其中，資料加密可以防止監聽。身份認證可以防止遠端漏洞中的利用僞造來源地址的攻擊。存取控制可以防止部分利用漏洞的攻擊和防止攻擊者的掃描，另外它也可以減少部分阻斷服務攻擊。稽核可以對漏洞攻擊、攻擊者的掃描、惡意程式碼的破壞行爲和阻斷服務攻擊做下記錄。監控可以偵測出漏洞攻擊、攻擊者的掃描、惡意程式碼的破壞行爲和阻斷服務攻擊。掃描則是專門用來偵測系統中是否有惡意程式碼。由這個表我們可以知道，目前無法解決的問題有三個：未知漏洞、阻斷服務和 Social Engineering。

	資料加密	**身份認證**	**存取控制**	**稽核**	**監控**	**掃描**
監聽	避免					
密碼破解[13]						
漏洞		避免(僞造)	減低	記錄	偵測	
掃描			避免	記錄	偵測	
惡意程式碼				記錄	偵測	偵測
阻斷服務			減低	記錄	偵測	
Social Engineering						

表五、典型的攻擊(列)對典型的防守(欄)表

未知漏洞是指未被公開的漏洞，也就不會有修補(patch)程式或方法出現，所以當然沒辦法防止攻擊者利用未知漏洞來攻擊。而且目前有太多種類和數量的軟體和程式，因此未知漏洞的數量是相當多的。

現有的防守方法最多只能靠防火牆來降低阻斷服務攻擊的程度、或是用稽核程式記錄下攻擊和用監控程式偵測到攻擊，但是攻擊者的來源位址通常是僞造的，所以記錄下來的資料也沒任何意義。拿一個全球資訊網網站來說，它必須接受全世界的連線提供服務，跟本沒辦法分辨攻擊者和正常使用者，所以阻斷服務也是現今仍無法解決的問題。

最後就是 Social Engineering，這類型的攻擊通常是利用使用者對安全的不重視和不了解，而每個人對安全的認知和重視程度通常也不盡相同，所以這類型的攻擊也是無法解決的。

[13] 密碼破解是一種嘗試錯誤的過程，一定無法防止，所以我們在此不予以討論。

5. 結論

網路安全是一個必須全面考量的問題，只要任何一個環節出問題，攻擊者就可以利用這個弱點加以攻擊。這篇報告討論的是企業內網路如何在網際網路這個混亂的大環境中確保自己的安全，所以我們就分別就企業內網路的使用者和系統管理者兩個方面來探討需要注意的事項。

就使用者來說：由於使用者大多是使用 Windows 系統，所以必須安裝掃描程式和隨時取得最新的已知惡意程式碼樣本來確保自己的主機不被病毒感染和被安裝後門程式。另外，要注意自己的密碼不被攻擊者使用 Social Engineering 或其它方法獲取，不要認爲自己的主機上並沒有存放重要資料就認爲無所謂，卻不知本身雖無重要資訊外流的危險或損失，但是卻可能成爲攻擊者的攻擊進入點或是分散式阻斷服務攻擊的守護神常駐程式，造成企業內網路主機被攻擊。

就系統管理者來說：系統管理者必須經常注意系統漏洞及修補漏洞，架設防火牆來過濾外來的封包，經常性使用監控程式來偵測系統有無異常現象，使用稽核程式來記錄系統的各種有關安全的活動，使用複雜度較高的密碼，儘可能不讓系統被入侵。因爲一但系統被入侵，則系統內的所有使用者的密碼和資料將一併被取得，同時也可以成爲攻擊者的一個打手，用來攻擊或入侵其他主機。

電腦安全有句至理明言—**沒有絕對的安全。**

參考資料

[1] Dorothy E. Denning, Peter J. Denning, "Internet Besieged", Addison Wesley, Oct 1997

[2] A. Boulanger, "Catapults and grappling hooks: The tools and techniques of information warfare", Sep 30 1997, http://www.research.ibm.com/journal/sj/371/boulanger.html

[3] CERT Coordination Center, "Denial of Service Attacks", Feb 12 1999, http://www.cert.org/tech_tips/denial_of_service.html

[4] Sniffit, "Sniffit Page", http://reptile.rug.ac.be/~coder/sniffit/sniffit.html

[5] CERT Coordination Center, "CERT Coordination Center", http://www.cert.org

[6] CERT Incident Note IN-99-06, "Distributed Network Sniffer", Oct 25 1999, http://www.cert.org/incident_notes/IN-99-06.html

[7] L0phtCrack, "L0phtCrack", http://www.l0pht.com/l0phtcrack

[8] Sendmail, "Sendmail", http://www.sendmail.org

[9] redhat.com, "Red Hat Linux 6.2 Release Security Advisory", Apr 26 2000, http://www.redhat.com/support/errata/RHSA-2000014-16.html

[10] CERT Advisory CA-2000-13, "Two Input Validation Problems In FTPD", Jul 18 2000, http://www.cert.org/advisories/CA-2000-13.html

[11] SecurityFocus, "SecurityFocus.com", http://www.securityfocus.com

[12] CERT Vulnerability Note VN-98.01, "Vulnerabilities in the XFree86 Distribution of the xterm Program and Xaw Library", Mar 3 1998, http://www.cert.org/vul_notes/VN-98.01.XFree86.html

[13] SATAN, "Security Administrator's Tool for Analyzing Networks", http://www.fish.com/~zen/satan/satan.html

[14] SAINT, "The Security Administrator's Integrated Network Tool", http://www.wwdsi.com/saint

[15] D. Farmer and E. Spafford, "The COPS Security Checker System," Proceedings of Summer USENIX Conference, Anaheim, CA (June 1990), pp. 165-170.

[16] TIGER, "TIGER", ftp://net.tamu.edu/pub/security/TAMU

[17] Charles Schmidt, Tom Darby, "The What, Why, and How of the 1988 Internet Worm", http://www.software.com.pl/newarchive/misc/Worm/darbyt/pages/worm.html

[18] CERT Advisory CA-2000-04, "Love Letter Worm", May 4 2000, http://www.cert.org/advisories/CA-2000-04.html

[19] Back Orifice 2000, "Back Orifice 2000", http://sourceforge.net/projects/bo2k

[20] CERT Coordination Center, "TCP SYN Flooding and IP Spoofing Attacks", Aug 24 1998, http://www.cert.org/advisories/CA-96.21.tcp_syn_flooding.html

[21] David Dittrich, "The DoS Project's "trinoo" distributed denial of service attack tool", Oct 21 1999, http://staff.washington.edu/dittrich/misc/trinoo.analysis

[22] David Dittrich, "The "Tribe Flood Network" distributed denial of service attack tool", Oct 21 1999, http://staff.washington.edu/dittrich/misc/tfn.analysis

[23] Cryptographic Algorithms, "DES", http://www.ssh.fi/tech/crypto/algorithms.html#DES

[24] Cryptographic Algorithms, "IDEA", http://www.ssh.fi/tech/crypto/algorithms.html#IDEA

[25] Cryptographic Algorithms, "RSA", http://www.ssh.fi/tech/crypto/algorithms.html#RSA

[26] Cryptographic Algorithms, "Diffie-Hellman", http://www.ssh.fi/tech/crypto/algorithms.html#Diffie-Hellman

[27] MIT distribution site for PGP, "Welcome to the MIT Distribution Center for PGP (Pretty Good Privacy)", http://web.mit.edu/network/pgp.html

[28] The Secure Shell Community Site, "The Secure Shell Community Site", http://www.ssh.org

[29] ABA Section of Science and Technology Electronic Commerce Division Information Security Committee, "Digital Signature Guidelines", http://www.abanet.org/scitech/ec/isc/dsgfree.html

[30] R. Rivest, "The MD5 Message-Digest Algorithm", Apr 1992, http://sunsite.auc.dk/RFC/rfc/rfc1321.html

[31] SECURE HASH STANDARD, "SECURE HASH STANDARD (SHS)", http://www.fastekk.com/news/dig_sig.html

[32] S. Deering, R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", Dec 1995, http://sunsite.auc.dk/RFC/rfc/rfc1883.html

[33] David Rudder, "Firewalling and Proxy Server HOWTO", Jul 17 1995, http://www.cc.ncu.edu.tw/~center5/linux/Redhat/ldp/Firewall-HOWTO.html

[34] David "Del" Elson, "Intrusion Detection, Theory and Practice", Mar 27 2000, http://www.securityfocus.com/focus/ids/articles/davidelson.html

[35] Tripwire, Inc., "Data Security Software", http://www.tripwiresecurity.com