Module 8： 防火牆

學習目的

1. 防火牆是一種用來控制網路存取的設備，並阻絕所有不允許放行的流量，防火牆通常會提供許多精細的組態設定等級。可以設定成依據服務、來源或目標的IP位址、要求服務的使用者識別碼等，做為流量過濾的基礎。
2. 防火牆基本上可以分為『應用層防火牆（application layer）』和『封包過濾型防火牆（packet filtering）』兩種。這兩種防火牆各自提供不同的功能，如果能妥善設定相關組態設定，都可以達到阻斷不符合安全需求的不正常流量。

3. 本課程模組將引領學生了解各類防火牆的運作原理，組態設定及防火牆規則設計，使其發揮安全防護的功能，讓防火牆的警示系統及時地提供防火牆重要的事件訊息給相關的系統安全人員。
4. 本模組共有三個小節包括(1)防火牆簡介(2) iptables(3) FireStarter(4)專案實作，共須三個鐘點。

Module 8：防火牆

- Module 8-1：防火牆簡介(*)
- Module 8-2：iptables(*)
- Module 8-3：FireStarter(**)
- Module 8-4：專案實作(*)

* 初級(basic):基礎性教材內容
**中級(moderate):教師依據學生的吸收情況，選擇性介紹本節的內容
***高級(advanced):適用於深入研究的內容

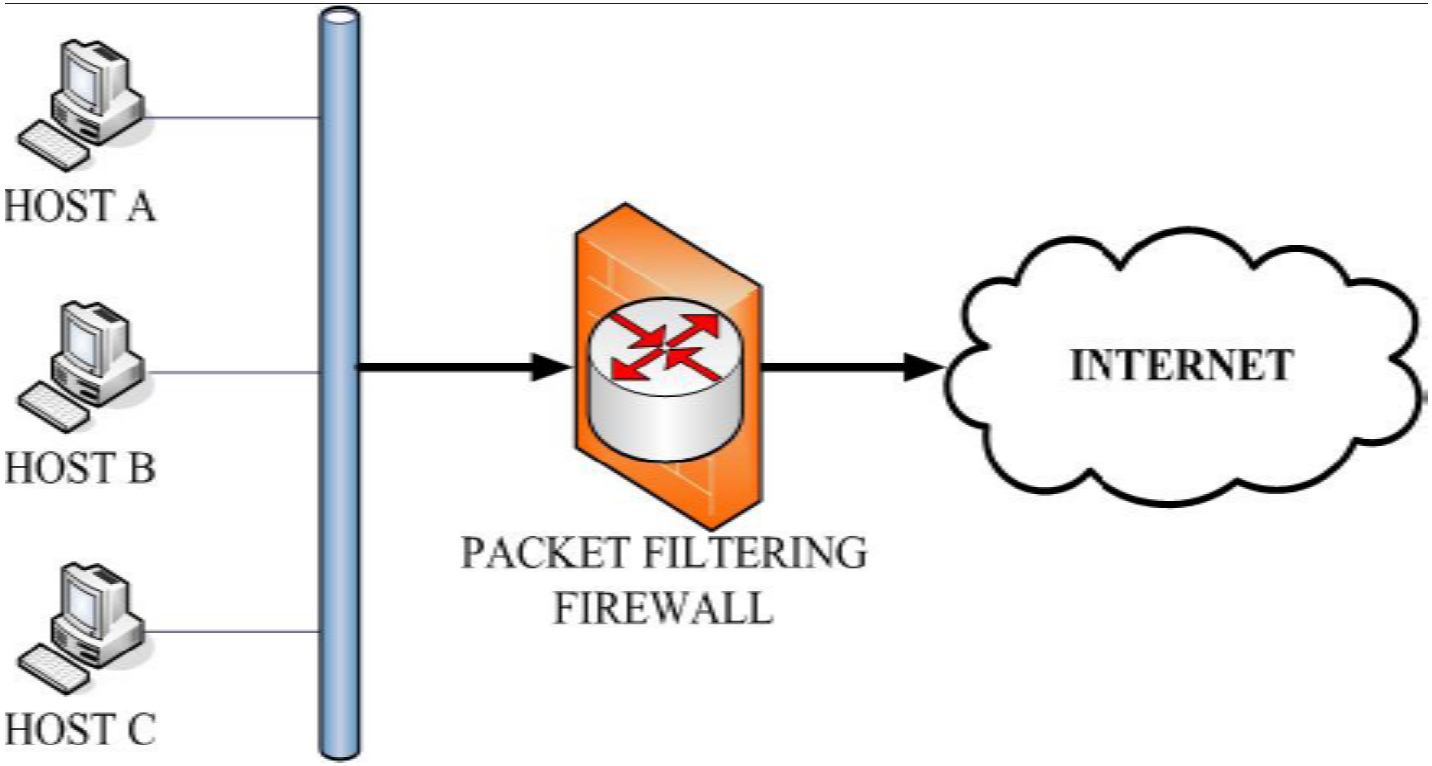
Module 8-1：防火牆簡介(*)

Module 8-1 ：防火牆簡介

- 防火牆通常是由一組軟硬體所組成，基本上是由一台主機，包含作業系統及安裝防火牆應用軟體而構成，通常建置於網際網路與內部網路之間，作為內部與外部溝通與管制的橋樑。
- 防火牆的原理，就是利用預先設定的規則，對遠端連接的封包進行檢查，符合規則的就允許通過，否則便進行阻止。

防火牆的作用

- 過濾、過濾再過濾



防火牆過濾、控制哪些東西

***Service control* –** determine types of Internet services, in/out bound.

–Used in most FW

- ***Direction control* –**determine the direction a service can allow to flow thru
- ***User control* –**which user is allowed to access
- ***Behavior control* –**control how a service is used

防火牆做不到的事

- 無法防止防火牆自己內部的不法行為
- 無法管理不經過防火牆的連線
- 無法防範全新的威脅
- 無法防範病毒

防火牆的組成架構分類

- **硬體防火牆**
  量身定制的硬體(ASIC)
  量身設計的作業系統
  EX:Netscreen(screen os)
- **軟體防火牆**
  通用架構的PC硬體
  Unix或是windows系列的通用作業系統
  EX:Checkpoint(unix,windows)
- **運作平台**
  提供最佳化過的硬體平台（模組化設計）
  運作專屬的作業系統
  EX:Crossbeam(X-Series Operating System (XOS)

軟硬體防火牆的差異

- Hardware firewall

  強調高效能，實用性，處理速度。

- Software firewall

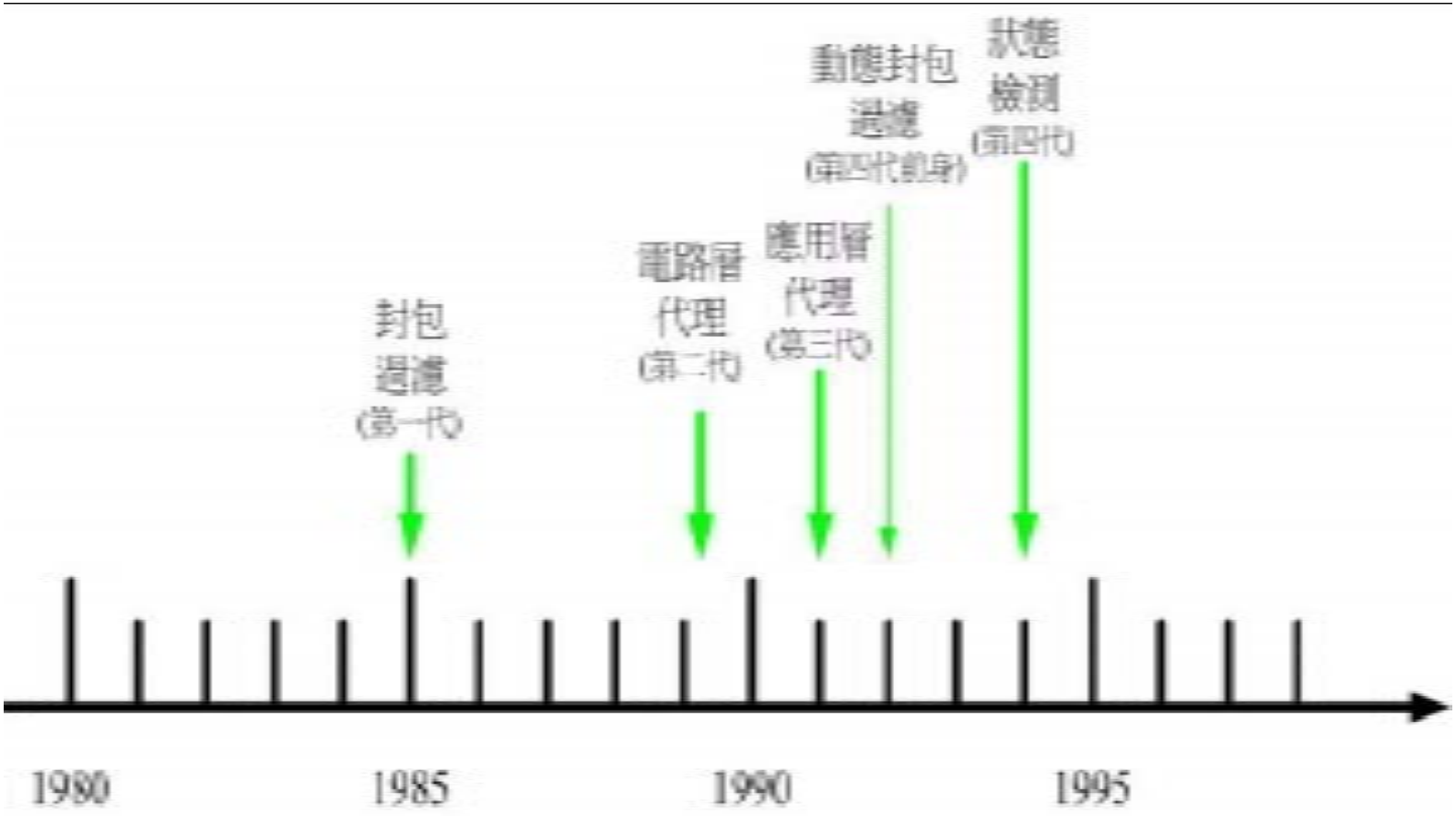
  設定較為彈性，可自行微調，近代作業系統多有內建各自的軟體防火牆。

硬體防火牆的內部實際上也是靠軟體在運作。

防火牆的歷史(技術)發展

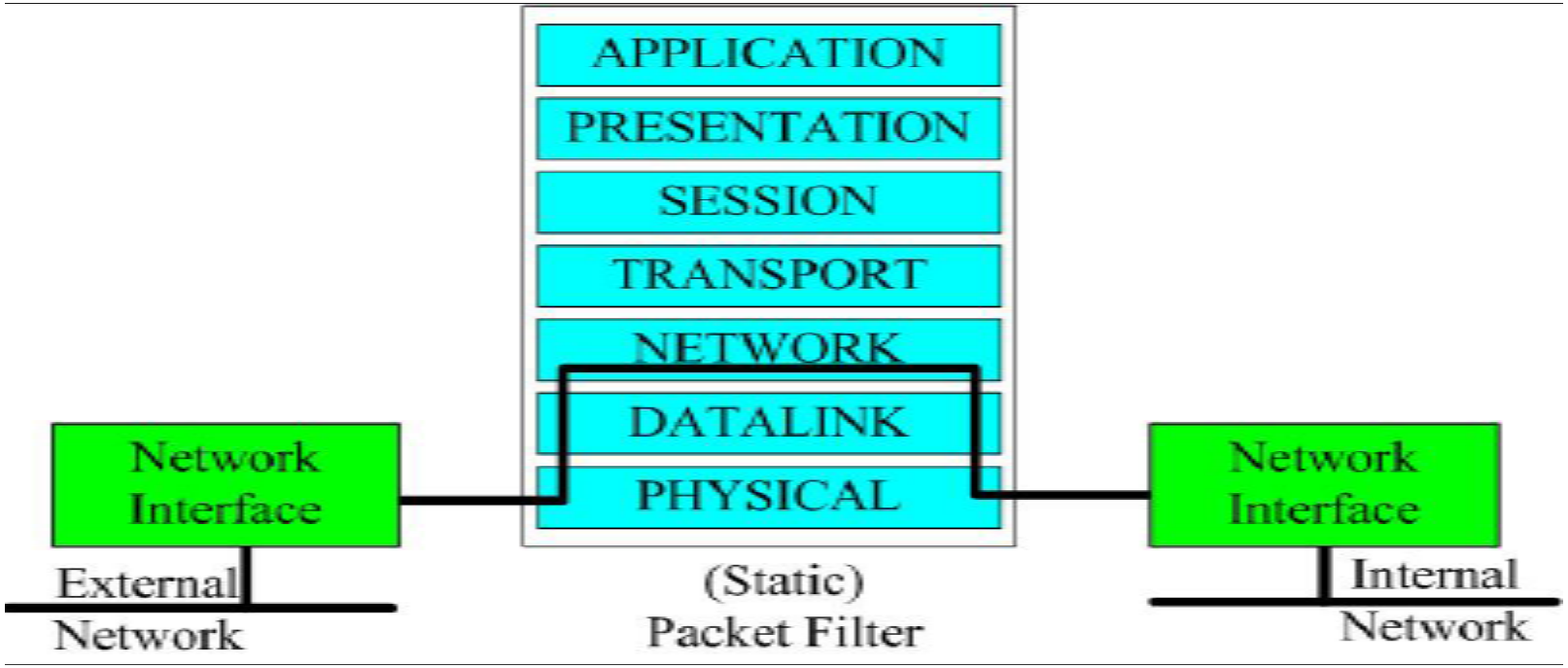
- 依類型分類：
  - 封包過濾防火牆packet filter
  - 代理防火牆proxy(gateway)
- 依發展時間分類：
  - 封包過濾packet filter
  - 電路層代理circuit-level proxy
  - 應用層代理application-layer proxy
  - 動態封包過濾dynamic(stateful) packet filter
  - 狀態檢測statefulinspection

防火牆的歷史(技術)發展


封包過濾(第一代)
電路層代理(第二代)
應用層代理(第三代)
動態封包過濾(第四代前身)
狀態檢測(第四代)
1980
1985
1990
1995

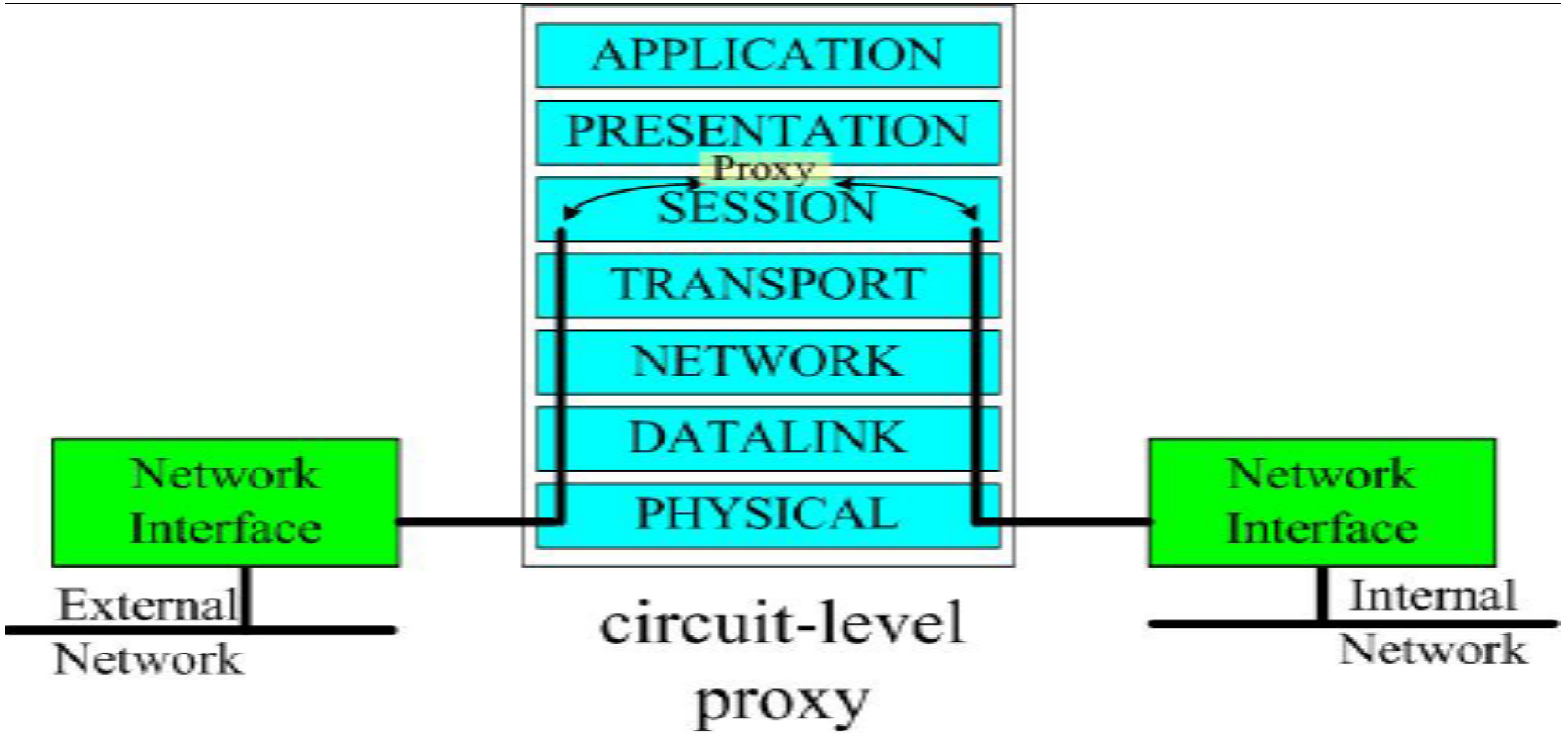
封包過濾式(第一代)packet filter

- 針對IP封包的表頭欄位與管理者制定的規則進行過濾



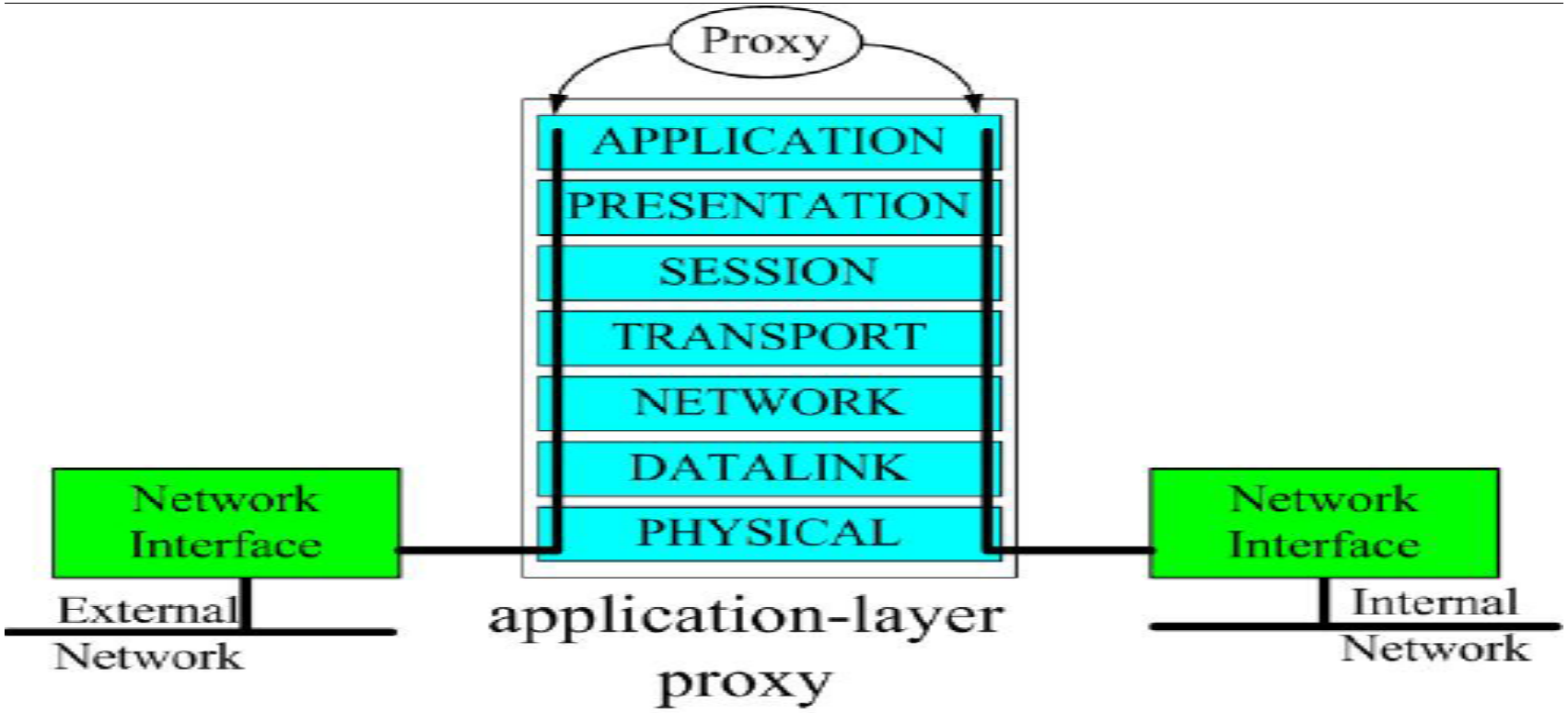
電路層代理(第二代)circuit-level proxy

- 在Session層上建立兩邊的TCP連線ex:socks



應用層代理(第三代) application-layer proxy

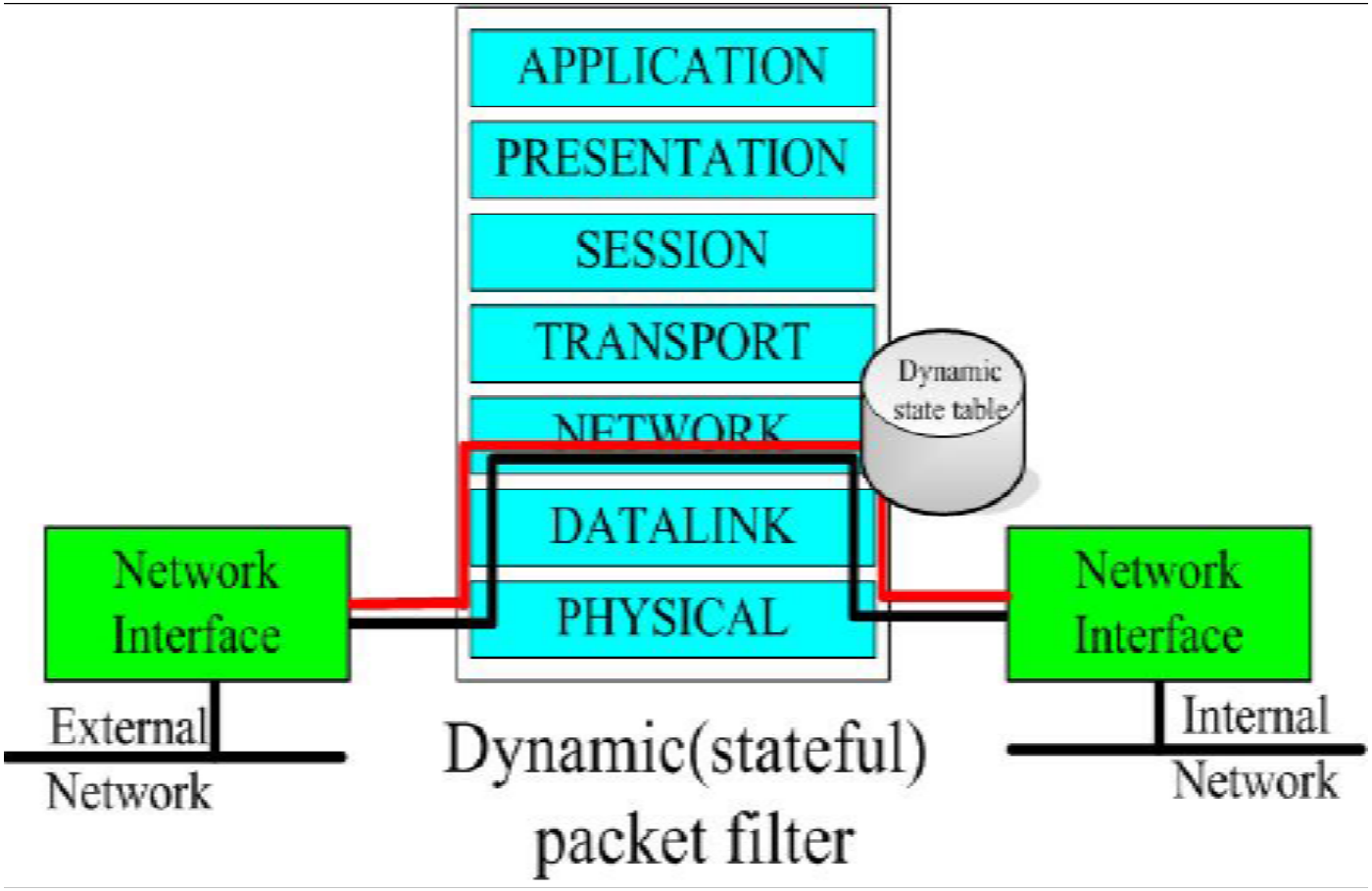
- 在應用層上建立兩邊的TCP連線ex:squid



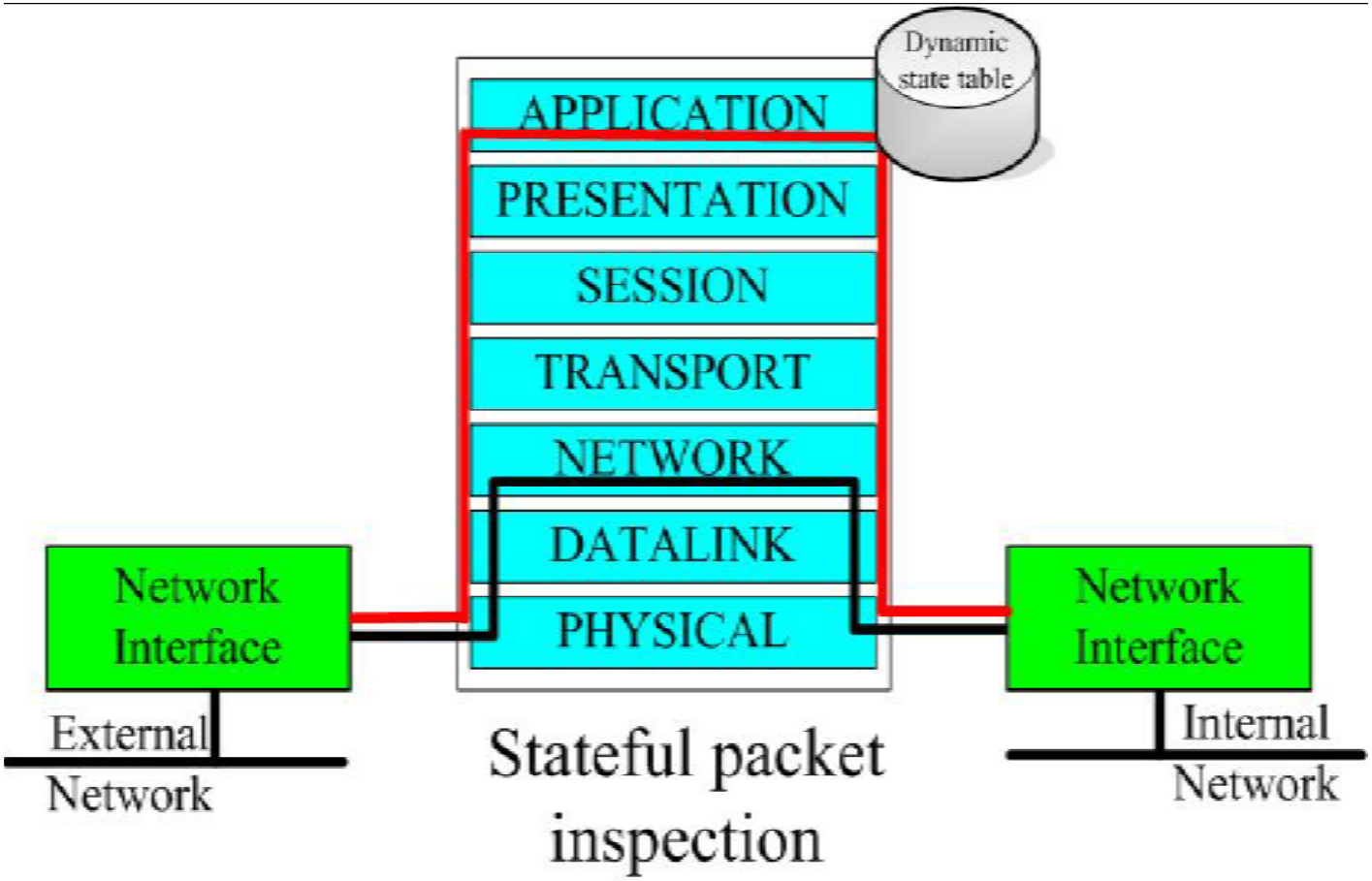
動態封包過濾(第四代前身)
Dynamic (stateful) packet filter

- 具keep state能力的防火牆，能只讓屬於正常且已記錄的連線封包進入受保護的網路。
- WINXP ICF
- LINUX iptables
- BSD ipfilter

動態封包過濾(第四代前身)
dynamic (stateful) packet filter



狀態檢測(第四代) statefulinspection

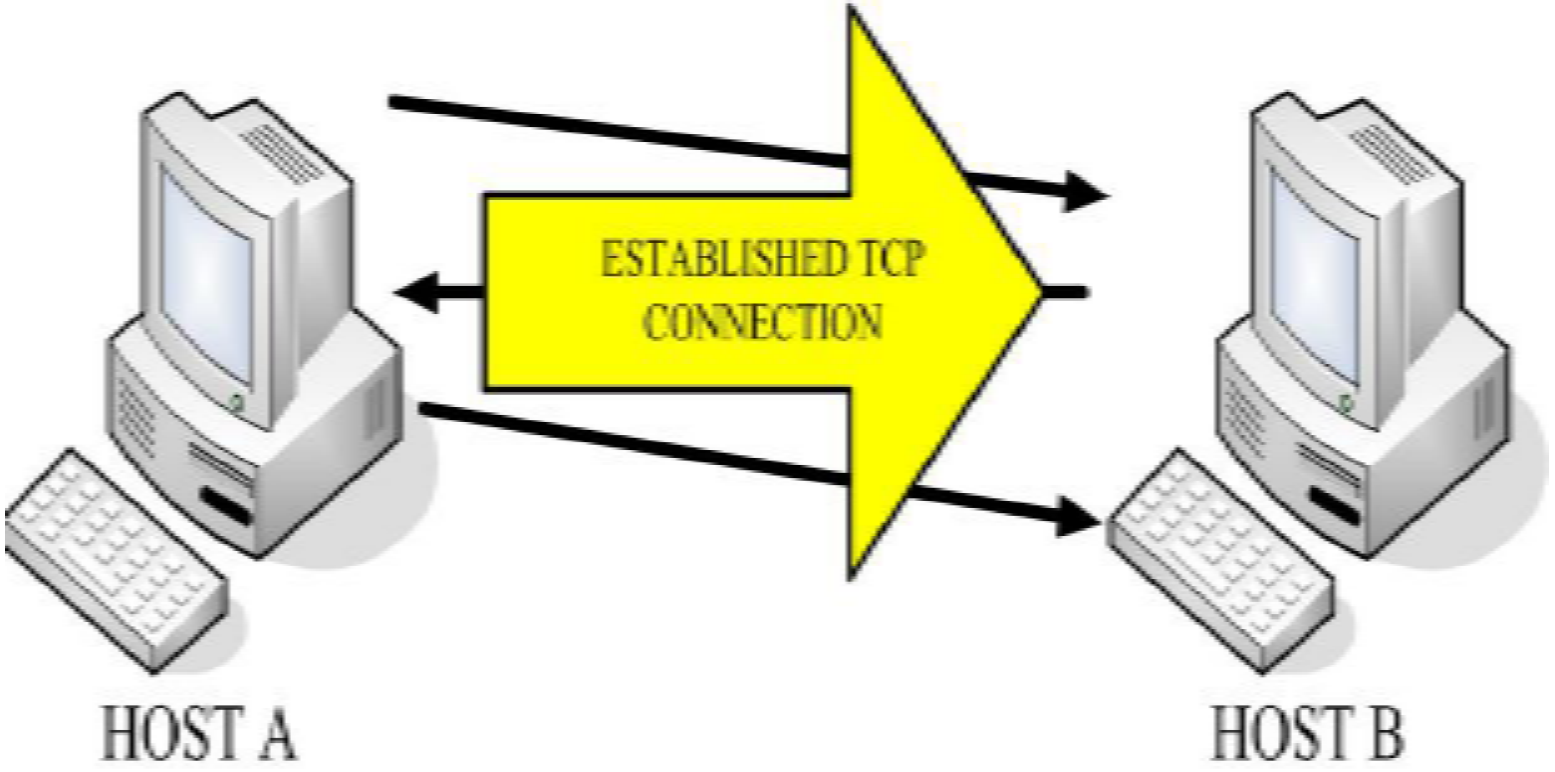


狀態檢測(第四代) statefulinspection

- Checkpoint FireWall-1(invented and patented)
- Cisco pix
- Netscreen
- 目前一些防火牆大廠宣稱有此功能

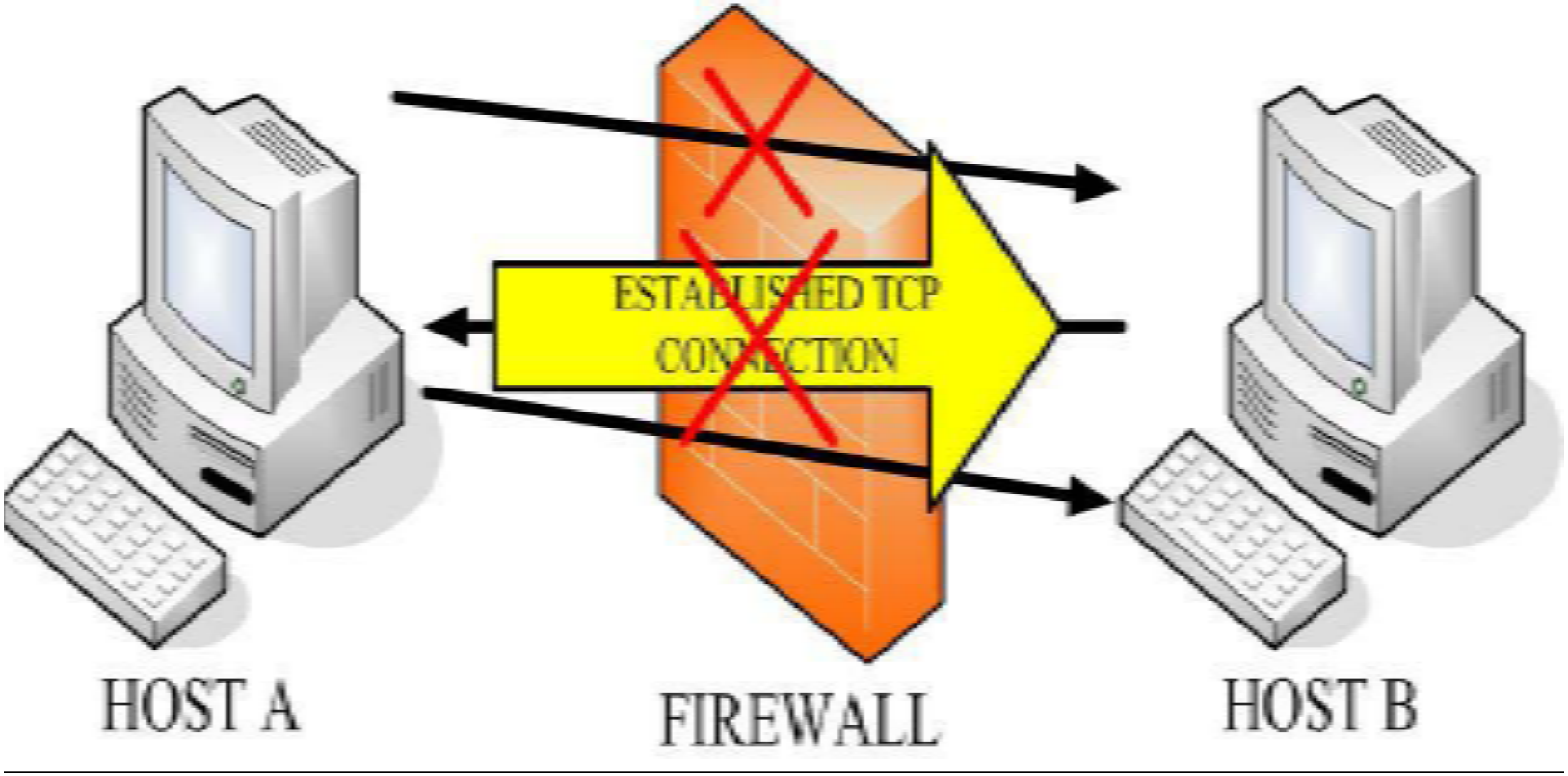
一般TCP 正常連線

- TCP threeway handshake



封包過濾阻止TCP連線

- 過濾SYN封包以阻止SYN連線



進階封包過濾

- 能夠重組封包，對內容進行辨識與處理
- 能夠進行所謂的content security
- 能夠分辨不同protocol的內容，如www,email
- 可以配合網路防毒軟體的運作
- 可以針對網頁內容進行過濾
- 缺點：需強大的硬體效能及昂貴的軟體
- EX:Checkpoint Firewll配合CVP server

防火牆的運作模式

- ROUTING MODE
- NAT MODE
- BRIDGE
- 個人單機防火牆

ROUTEING MODE

- Router
  - 防火牆扮演Router的角色，或者是Router設定封包過濾規則
  - 設定和實用上比較複雜，初期網路建構好之後較不易更動
  - 此為最早最古典的防火牆運作模式
  - 實例：Cisco Router的ACL

BRIDGING MODE

- Bridge
- 防火牆本身以Bridge的方式運作，本身可以不需要有IP，也可設定IP方便管理，但是會增加風險。
- 對原有的網路架構幾乎不影響。
- 又稱為Transparence Firewall
- 實例：Netscreen
- 目前支援此運作模式的防火牆較少

NAT

- 防火牆兼具NAT Server的功能，內部網路使用Private IP，經轉址後向外連線
- 外部網路要存取內部主機需設定對映或是轉址規則
- 為目前最常見的防火牆建構模式
- 通常有stateful的能力
- 應付目前外部IP不夠用的情況
- 大部份市售的防火牆皆有此功能

個人單機防火牆

- 個人使用的作業系統上安裝的防火牆
- 以防禦主機本身為主
- 以限制IP、PORT的存取限制為主或結合小型的入侵偵測系統運作

OS內建的封包過濾(Linux)

- 2.2 kernel以前ipchain
- 2.4 kernel以後iptable(netfilter)
  - 核心支援封包過濾能力，可作為單機、網路防火牆或是NAT主機。
  - 已有方便管理的圖形化管理工具。
  - 市面上有些防火牆、IP分享器其運作的軟體即以Linux為核心運作。

OS內建的封包過濾(BSD)

- 主要為ipfilter(Freebsd預設為ipfw)
- 核心支援封包過濾，可以router,bridge,nat等防火牆模式運作。
- 功能相當完整，運作順暢，有許多商用防火牆也是以BSD核心修改運作使用。
- 亦有管理工具可協助使用，但通常是直接修改文字規則(ipf.rules)。

Module 8-2 ： iptables簡介(*)

Module 8-2 ： iptables簡介

- 不同種類的封包處理，有個別的規則表
- 規則表依功能劃分，由不同的模組來實行
- 三個模組分別為：
  - filter
  - nat
  - mangle

  以上三個模組稱為iptables的**tables**

Filter 過濾封包

- 關於權限存取的設定都在這模組中。此模組中有三個鏈結（chain），分別為：
  - INPUT
  - OUTPUT
  - FORWARD
- 若封包的目的地是本機，則會依序通過INPUT與OUTPUT的檢驗；若封包只是經過本機則會通過FORWARD的檢驗。

NAT（轉譯ip位址）

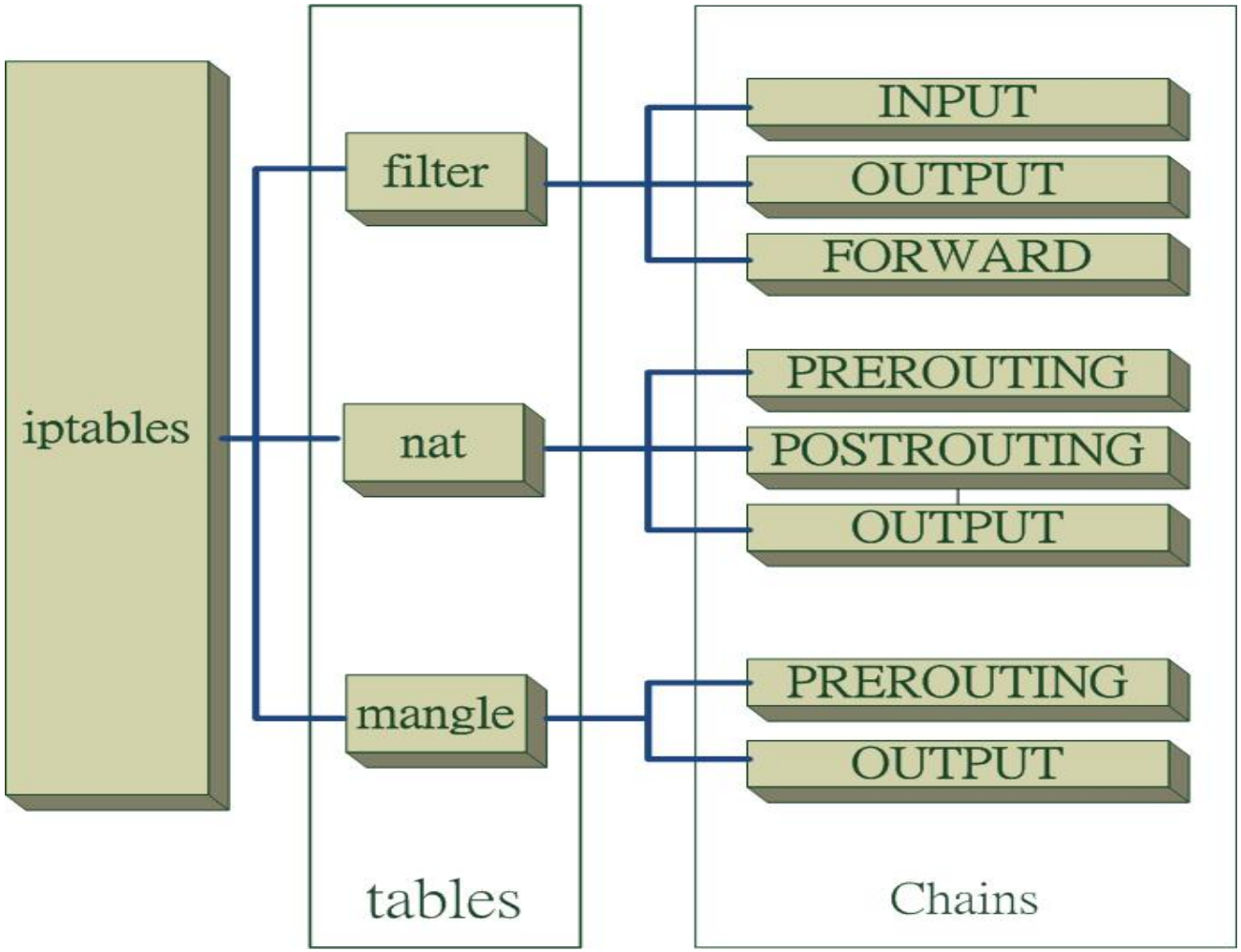
- 這個模組的作用是將封包轉送到其他位址，因此所有跟轉址有關的設定都在這個模組中定義，包括NAT服務轉向等等。

- 這個模組中有三個鏈結（chain）：
  - PREROUTING
  - POSTROUTING
  - OUTPUT

Mangle（重組）

- 這個模組會修改封包內資訊，重新組合資訊。這個模組中包含五個鏈結（chain）：
  - PREROUTING
  - OUTPUT

iptables 架構圖



Netfilter tables and chains

Filter point	tables		
	filter	nat	mangle
INPUT	◎		
FORWARD	◎		
OUTPUT	◎	◎	◎
PREROUTING		◎	◎
POSTROUTING		◎	

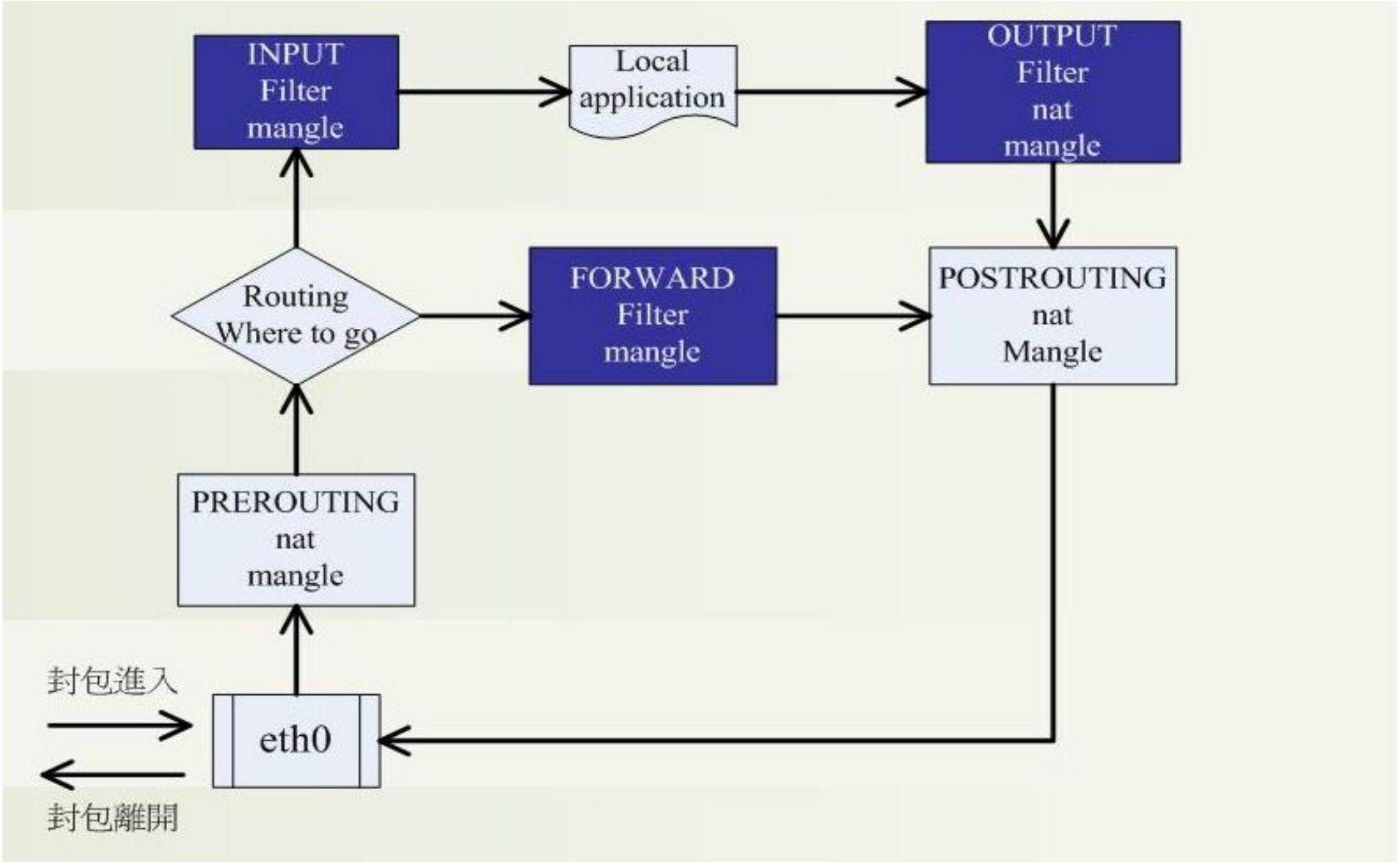
Match Arguments

parameter		specified
-p	!	• TCP • UDP • ICMP • A protocol name from /etc/protocols • All
-s		• network name • hostname • subnet (192.168.0.0/24 ; 192.168.0.0/255.255.255.0) • ip address
-d		
-i		• Interface name • Interface name ends in a “+” (eth+)
-o		
--sport		• Service name • Port number • Port range (1024:65535)
--dport		

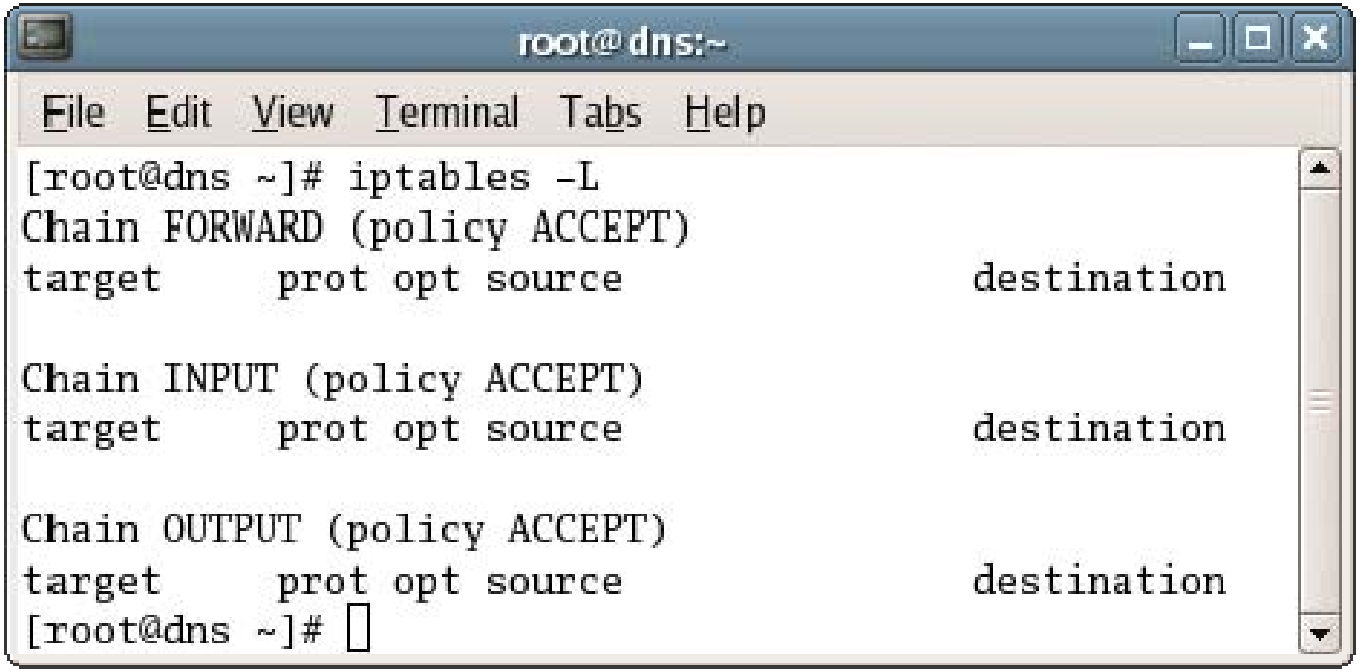
設定規則

	Tables	command	Chain	Parameter	target
iptables	-t filter	-A -D -I -R -L -F -Z -N -X -P	INPUT FORWARD OUTPUT PREROUTING POSTROUTING	-p -s -d -I -o --sport --dport	-j ACCEPT -j REJECT -j DROP

列出過濾表格的設定

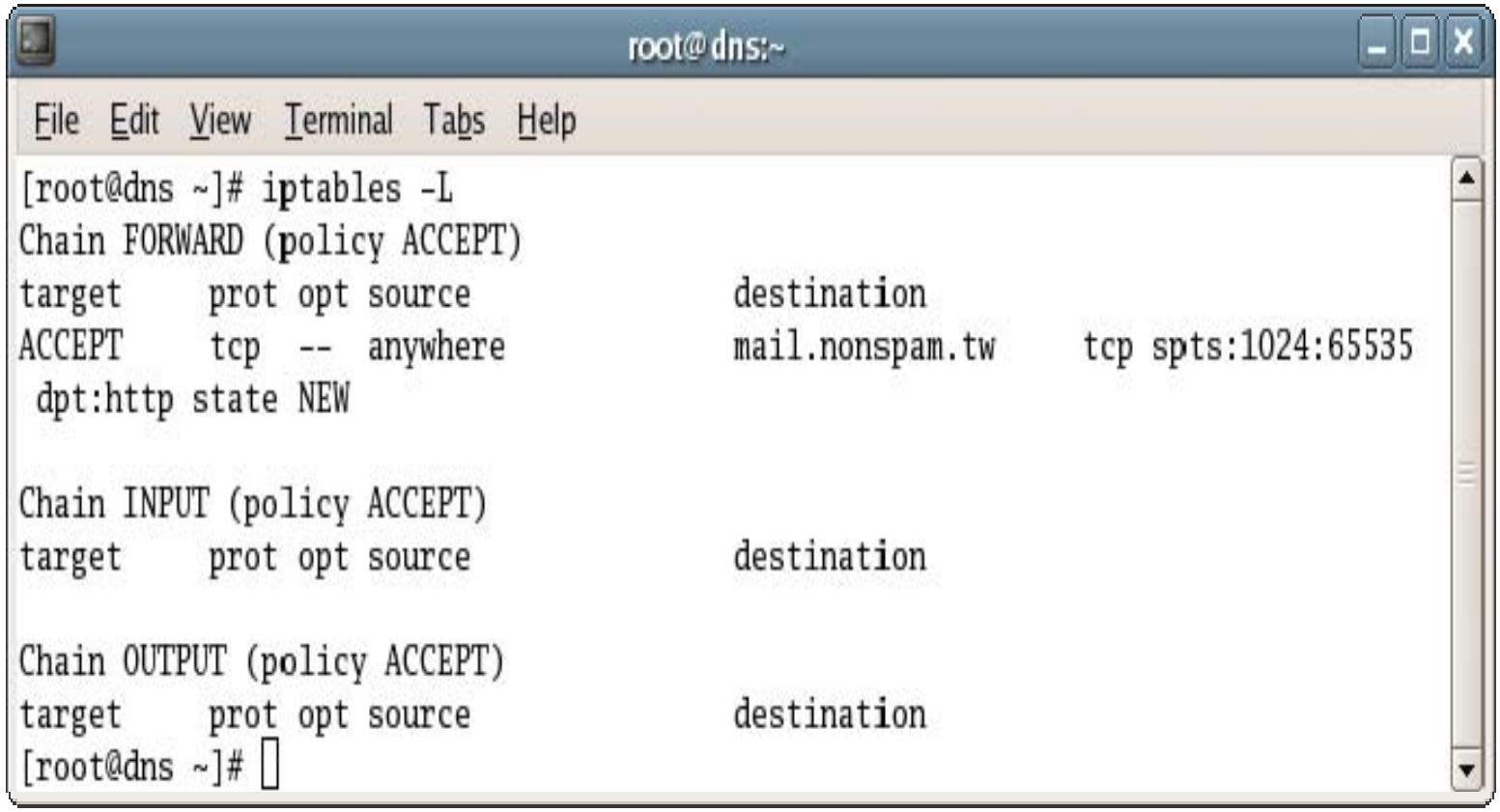

INPUT
Filter
mangle
Local
application
OUTPUT
Filter
nat
mangle
Routing
Where to go
FORWARD
Filter
mangle
POSTROUTING
nat
Mangle
PREROUTING
nat
mangle
封包進入
eth0
封包離開

列出過濾表格的設定



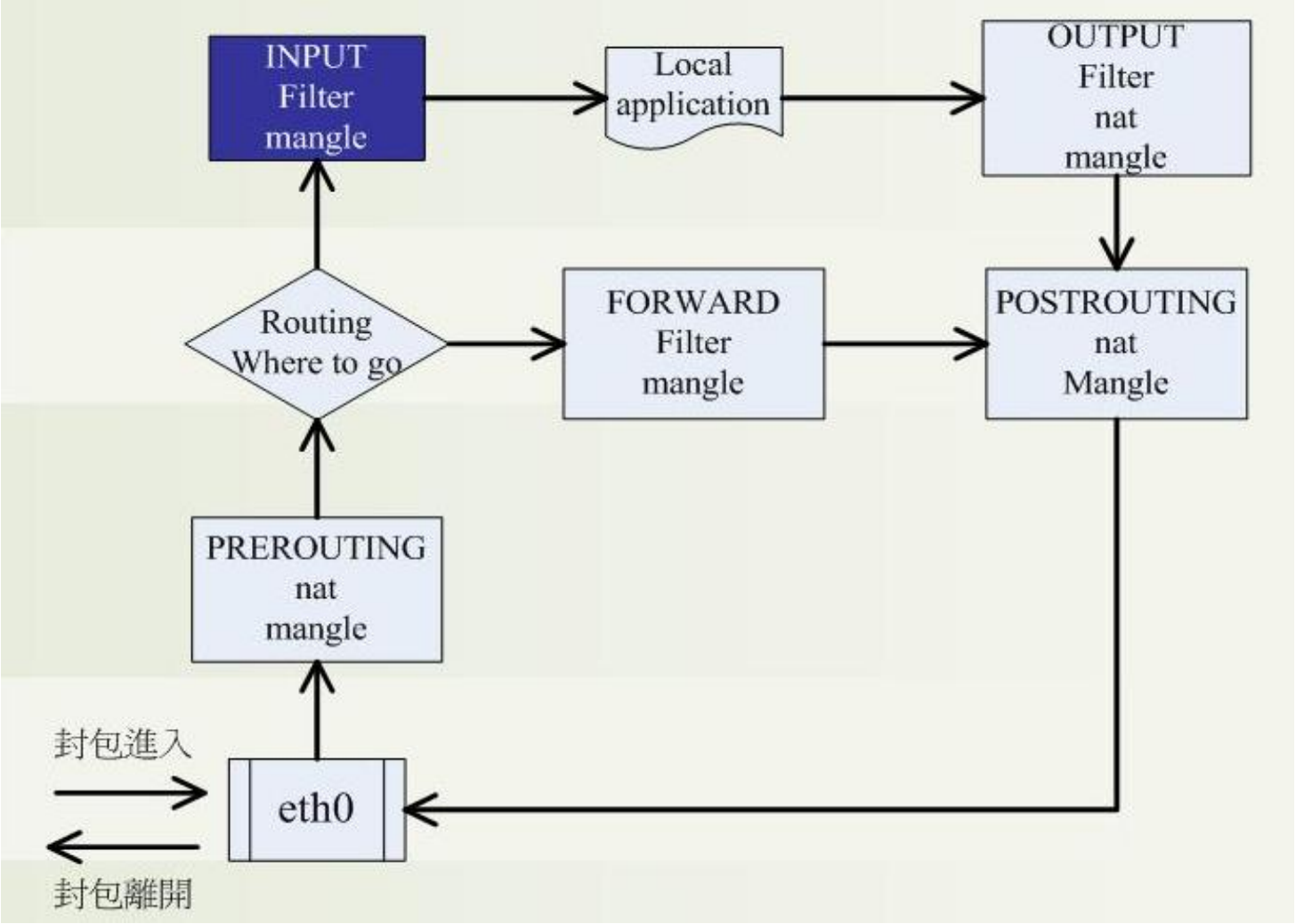
- 利用iptables –L查詢目前所定義的規則表，上圖為尚未定義任何規則的情況。

列出過濾表格的設定



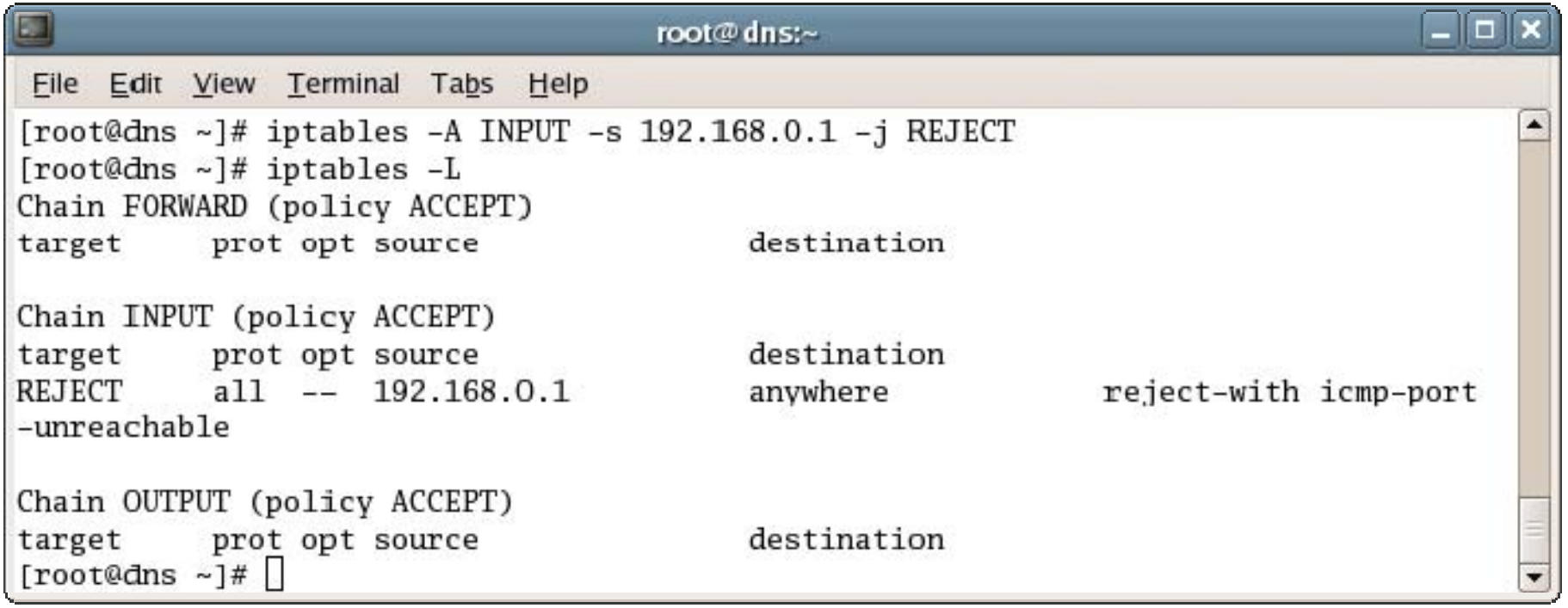
- 規則表中有定義規則的情形。

拒絕特定來源位址的封包

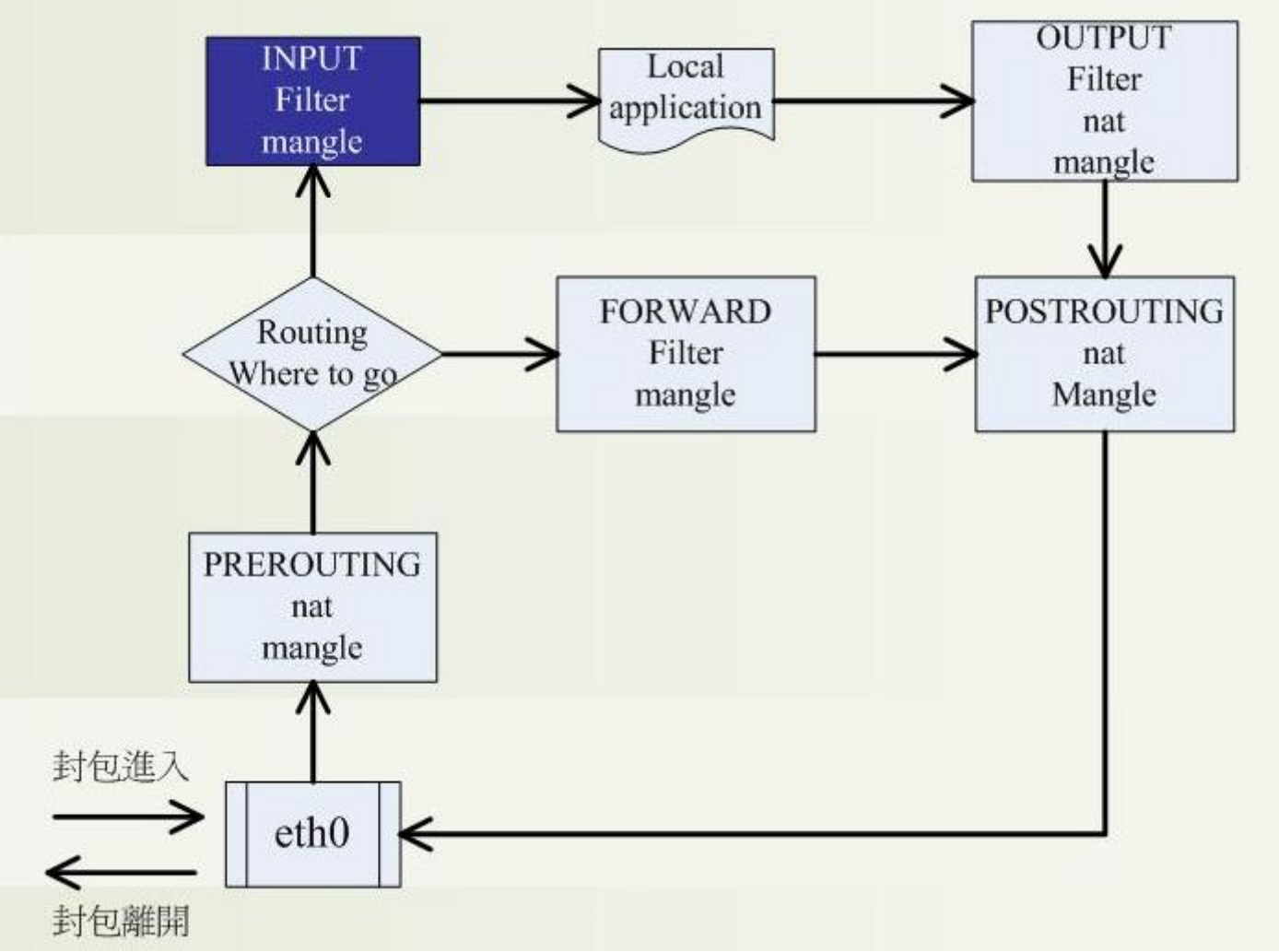


拒絕特定來源位址的封包

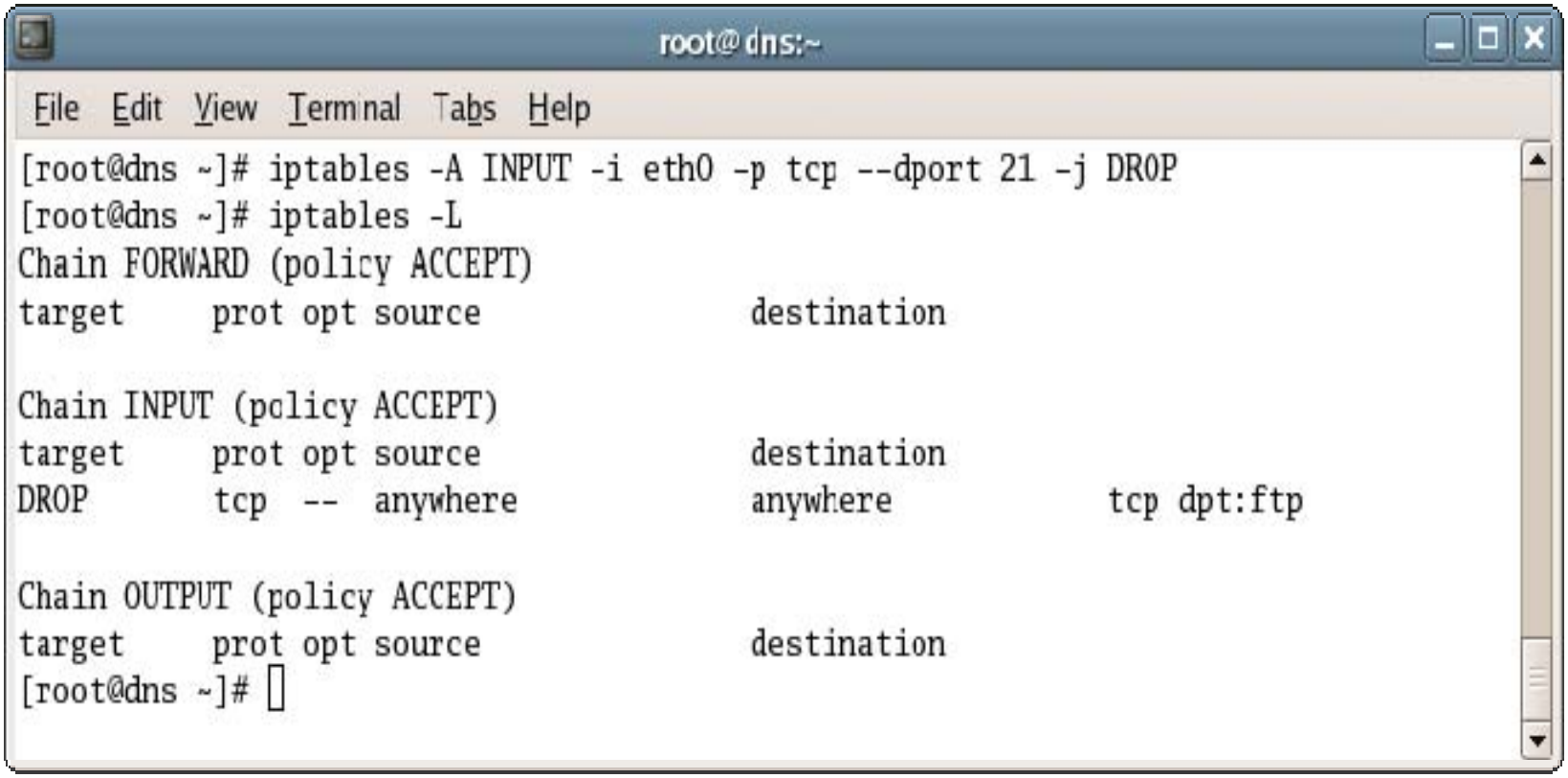
- 直接到達本機的封包，我們可以透過INPUT這個鏈結去規範。例如，拒絕本機連線到192.168.0.1這個位址：



丟棄來自本機特定埠號的封包

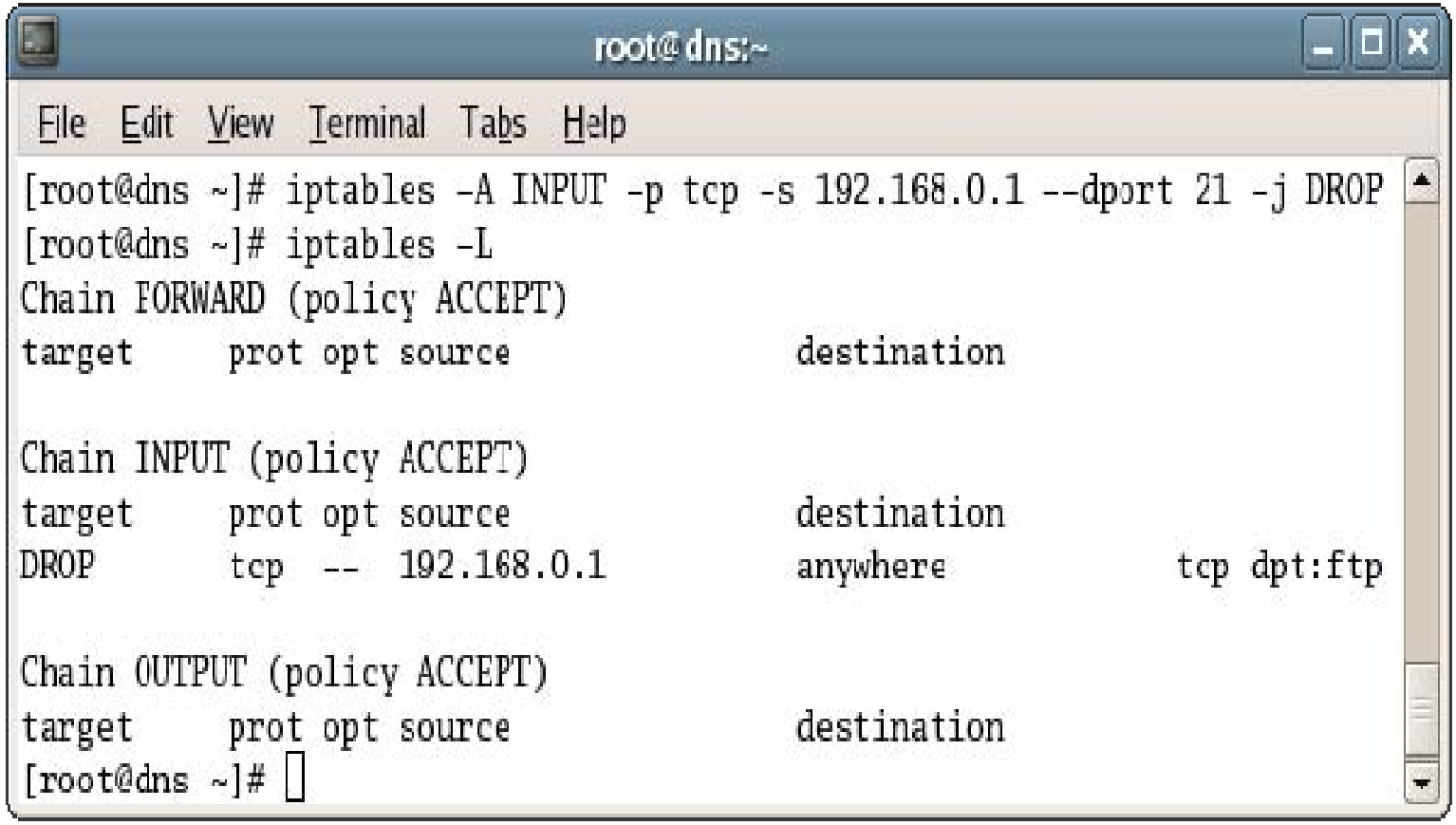


丟棄特定埠號的封包



- 拒絕所有用戶端連接到本機的21埠（FTP）

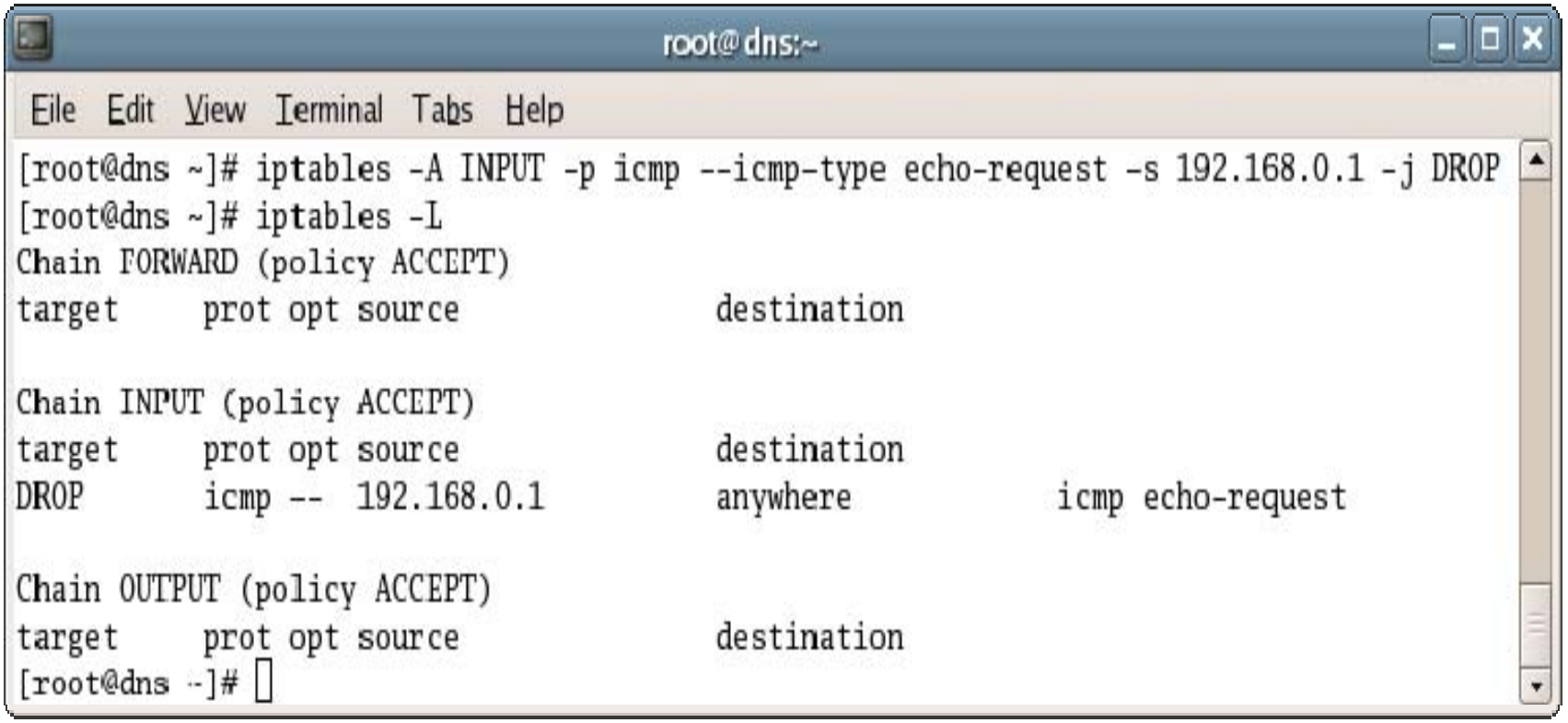
丟棄特定位址中特定埠號的封包



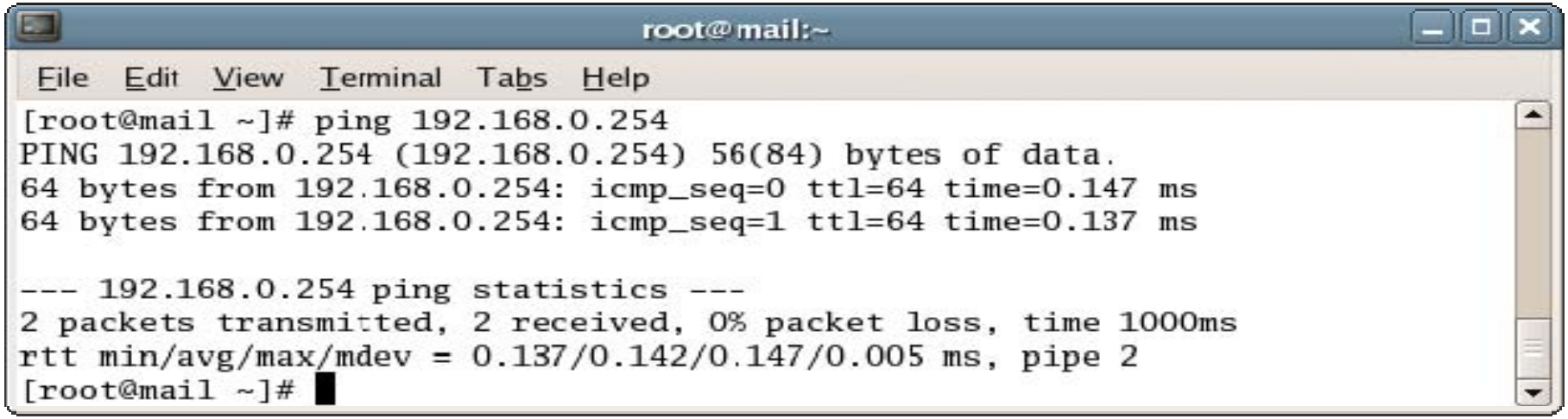
- 拒絕用戶端192.168.0.1連接到本機的21埠

對ICMP封包的處理

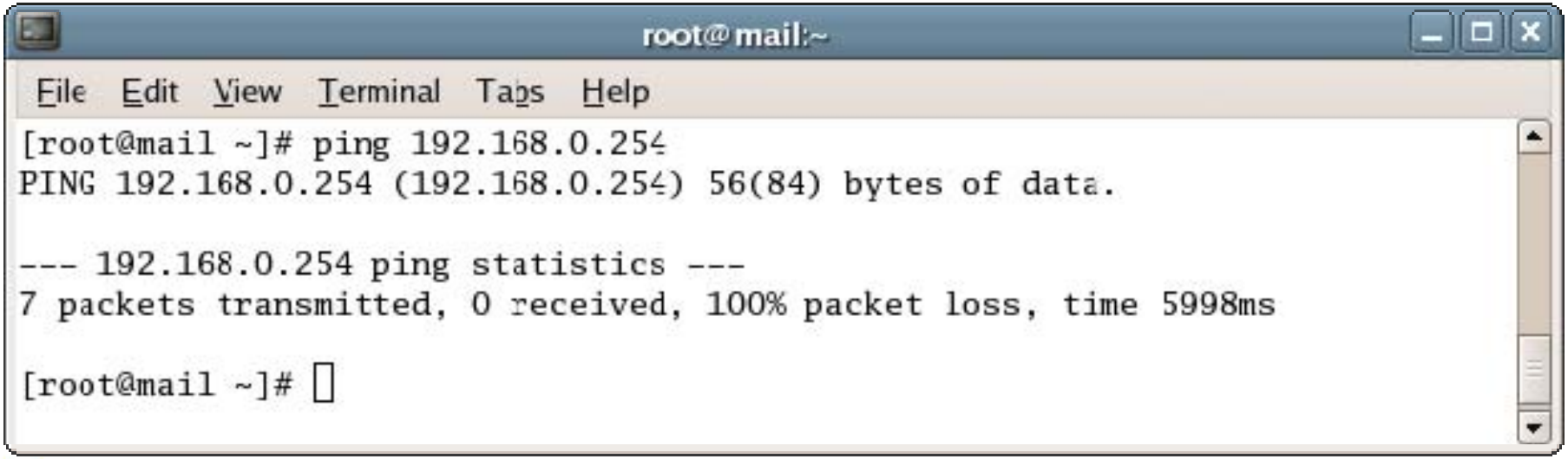
- 我們可以透過-p的參數指定icmp的封包，若我們不想回應192.168.0.1這位址對本機執行ping的動作，可執行以下指令：



對ICMP封包的處理

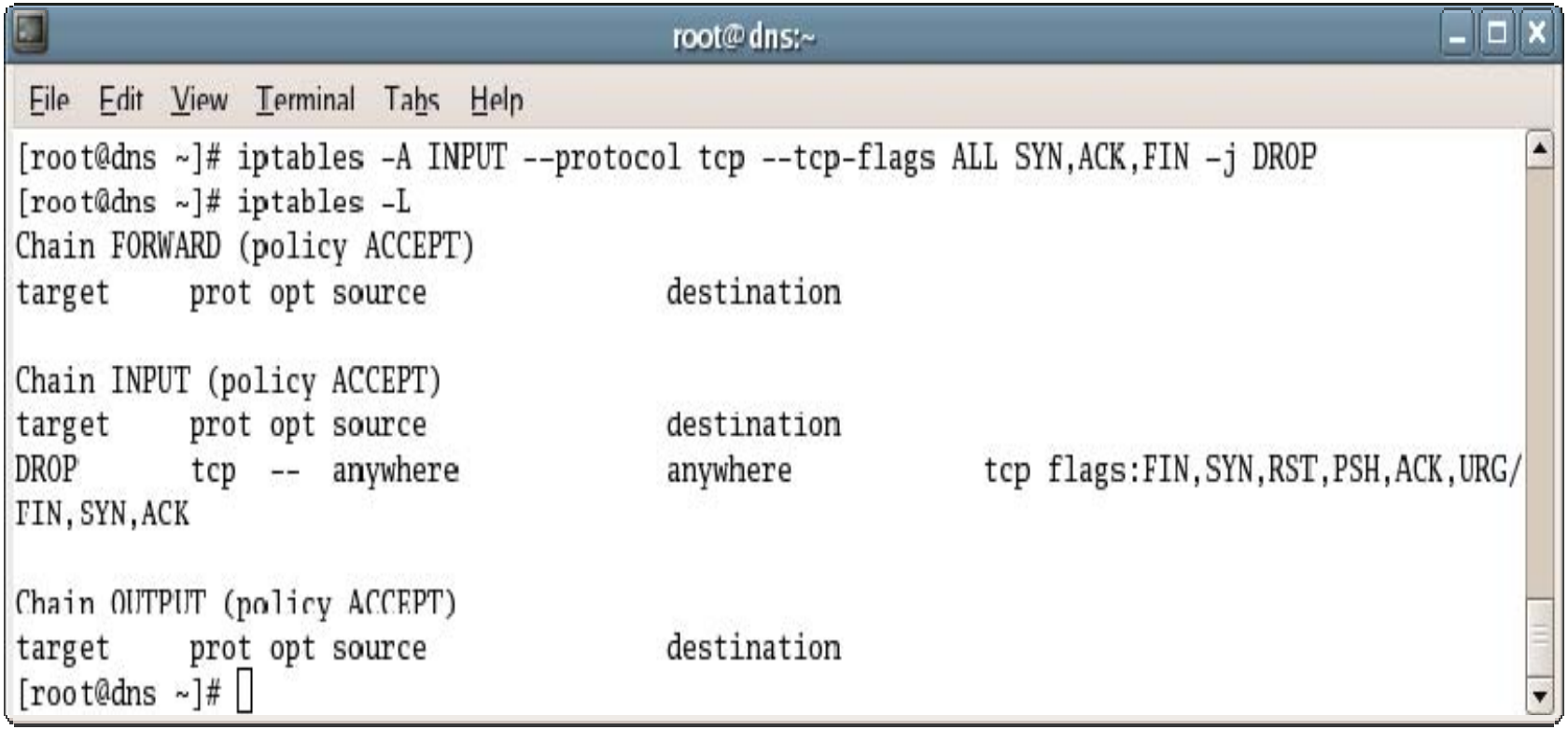


上圖為未阻擋icmp封包　　下圖為設定阻檔icmp封包

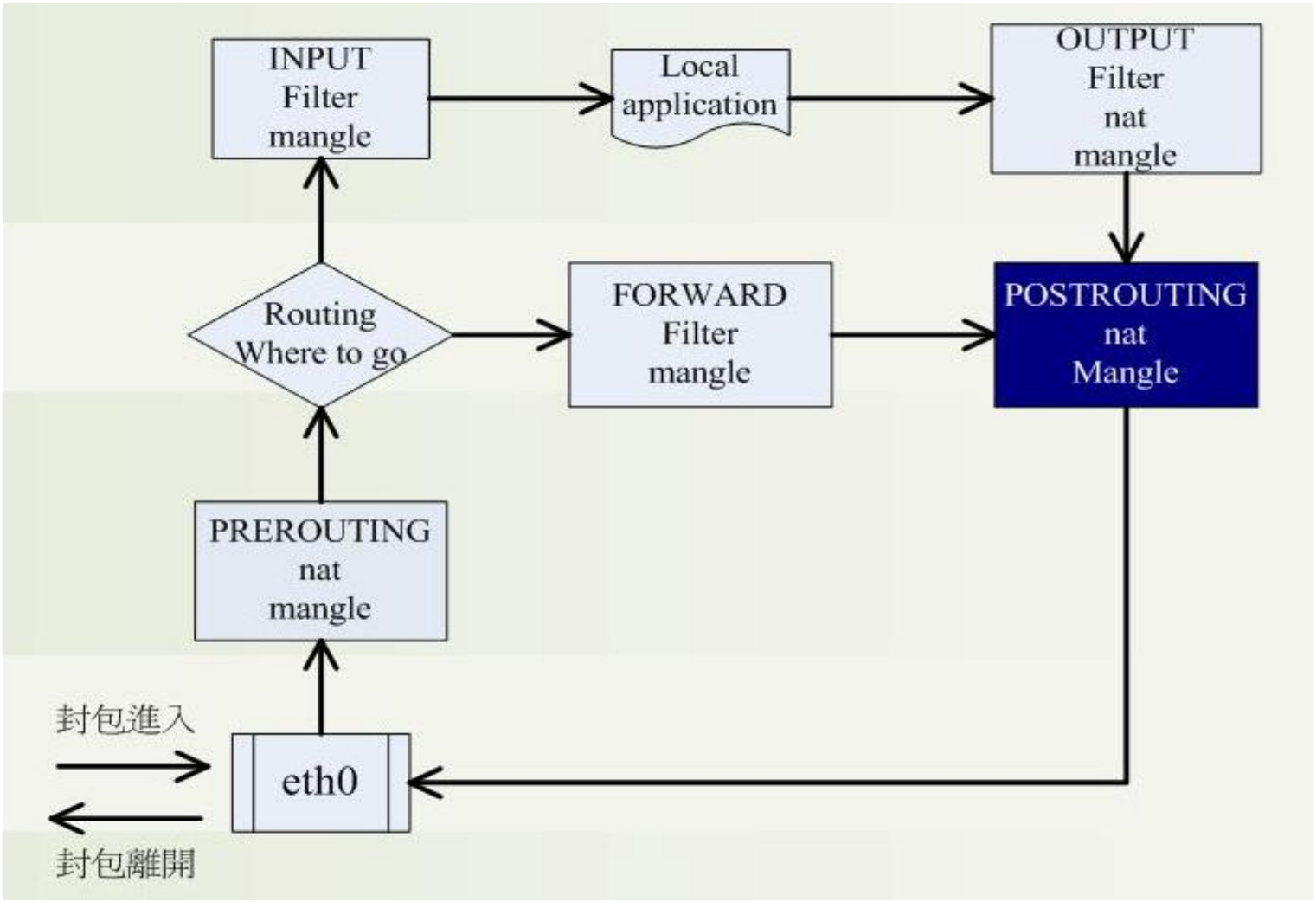


Denial of Service

- 設定對於特定標頭的過濾，能避免許多DoS的發生。我們可將不正常的標頭丟棄，如下所示：



設定NAT轉址功能


INPUT
Filter
mangle
Local
application
OUTPUT
Filter
nat
mangle
Routing
Where to go
FORWARD
Filter
mangle
POSTROUTING
nat
Mangle
PREROUTING
nat
mangle
封包進入
eth0
封包離開

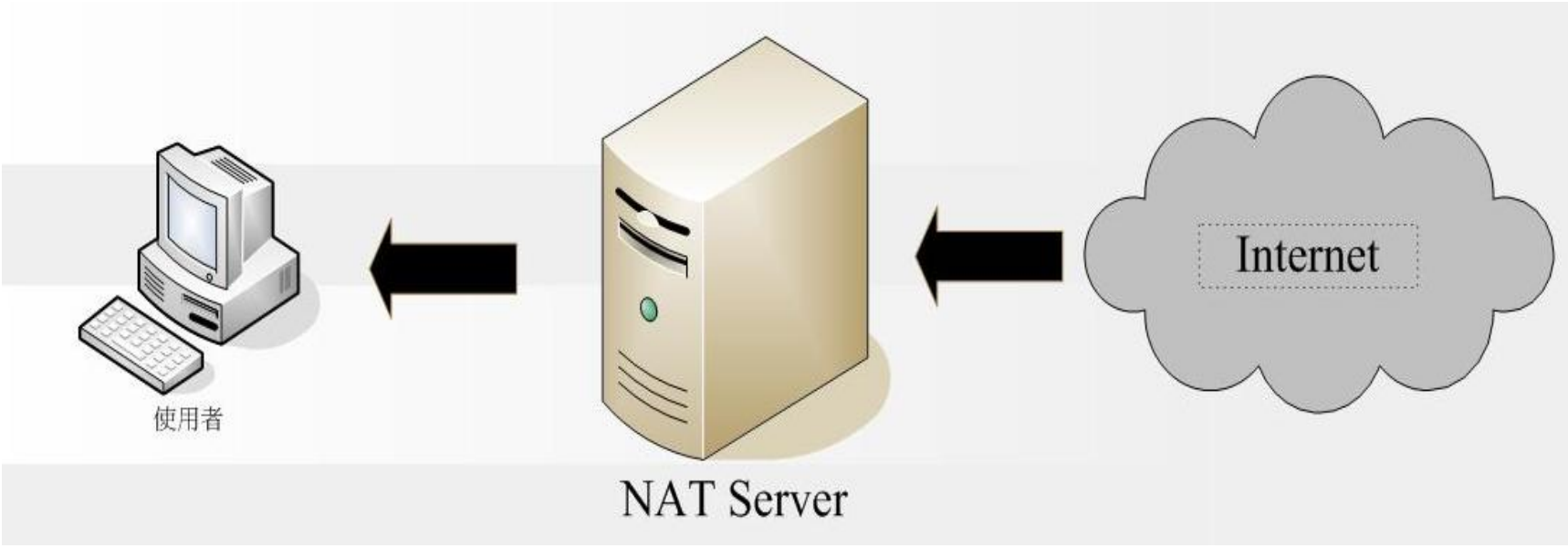
iptables nat

- iptables所支援的NAT
  - SNAT
  - DNAT

- 修改
  - PREROUTING
  - OUTPUT
  - POSTROUTING

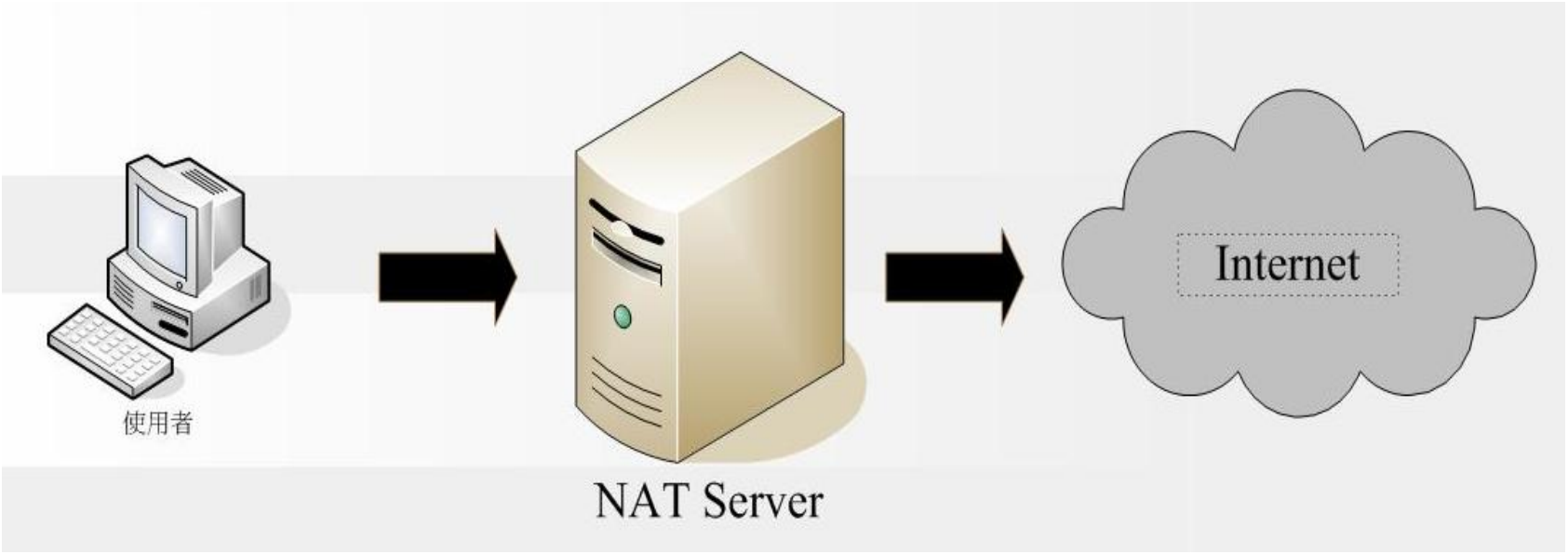
Destination NAT

- Destination NAT會改變封包的目的地位置，一般應用在負載平衡，可將封包指向不同的目的地，分散網路的流量。

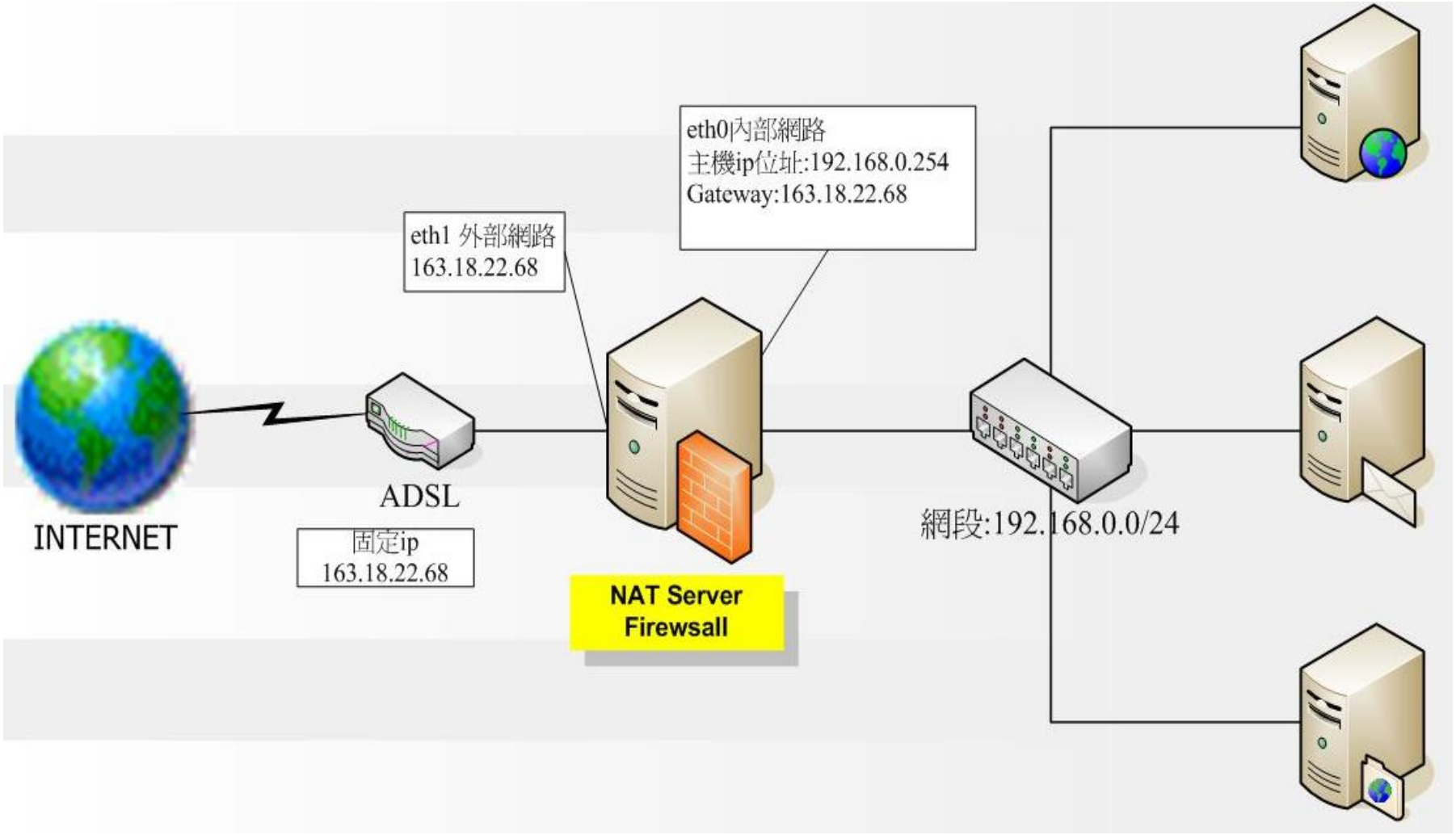


Source NAT

- Source NAT會改變IP封包的來源位置，這樣的機制常見於IP偽裝。

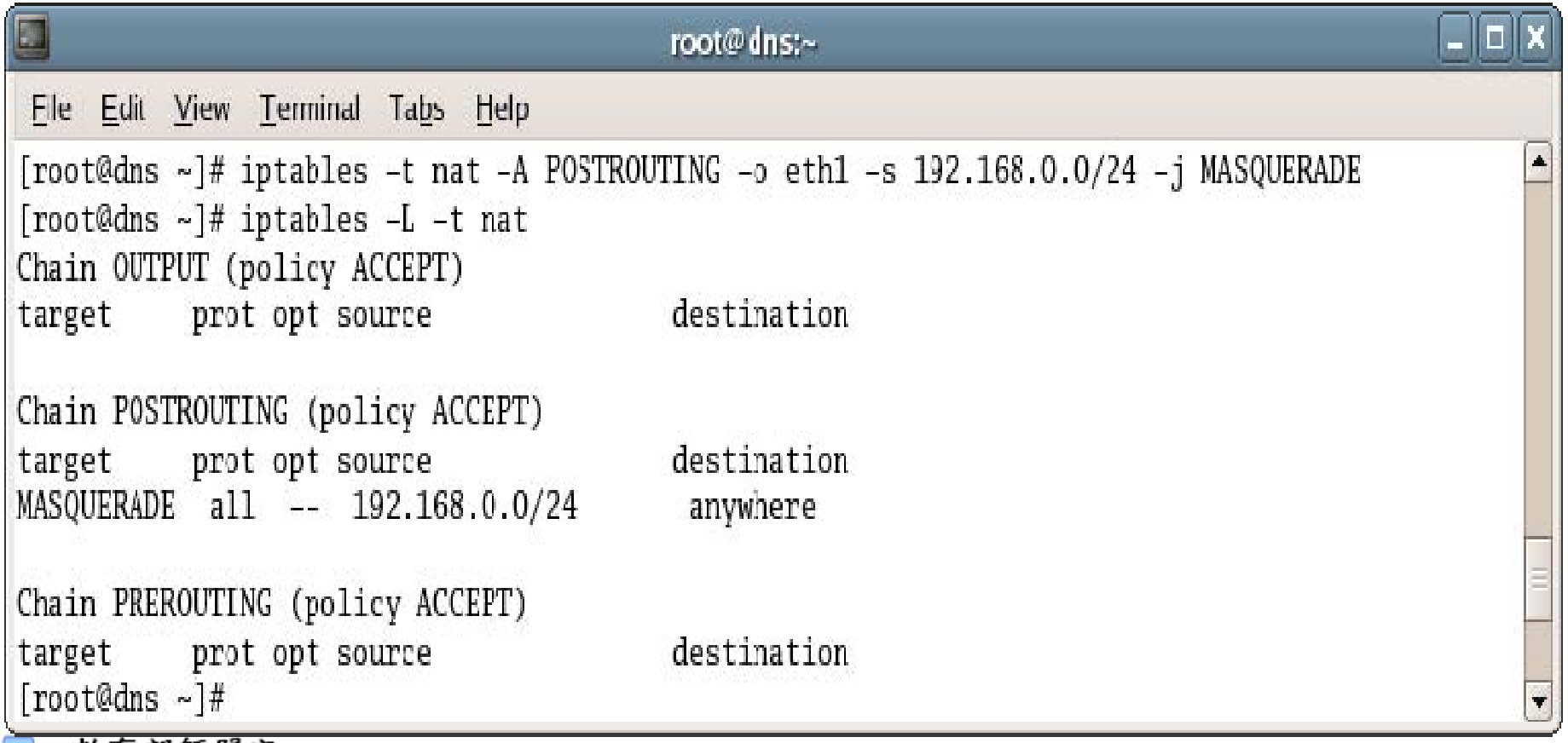


環境架構圖

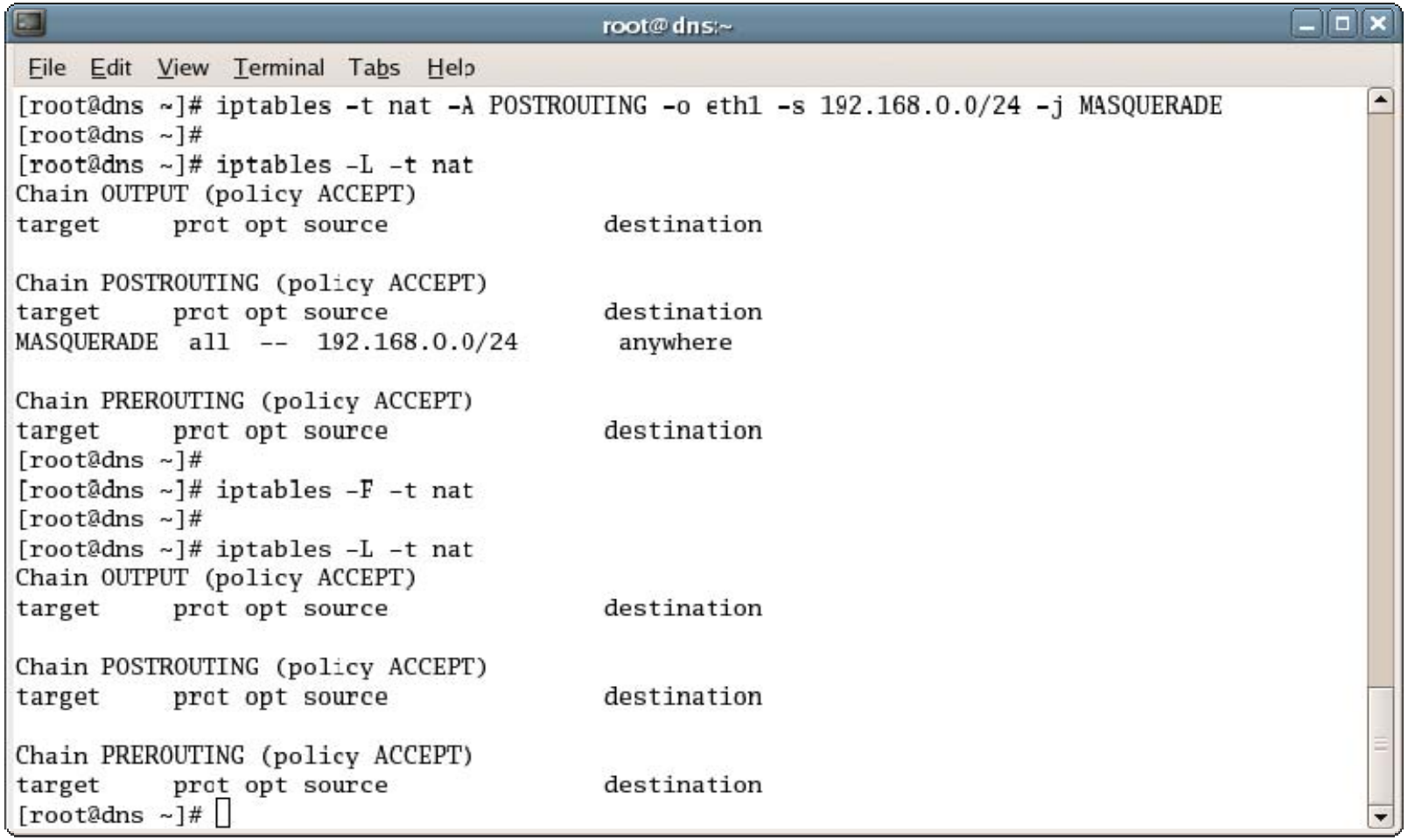

eth0內部網路
主機ip位址:192.168.0.254
Gateway:163.18.22.68
eth1 外部網路
163.18.22.68
INTERNET
ADSL
固定ip
163.18.22.68
NAT Server
Firewsall
網段:192.168.0.0/24

設定匝道伺服器

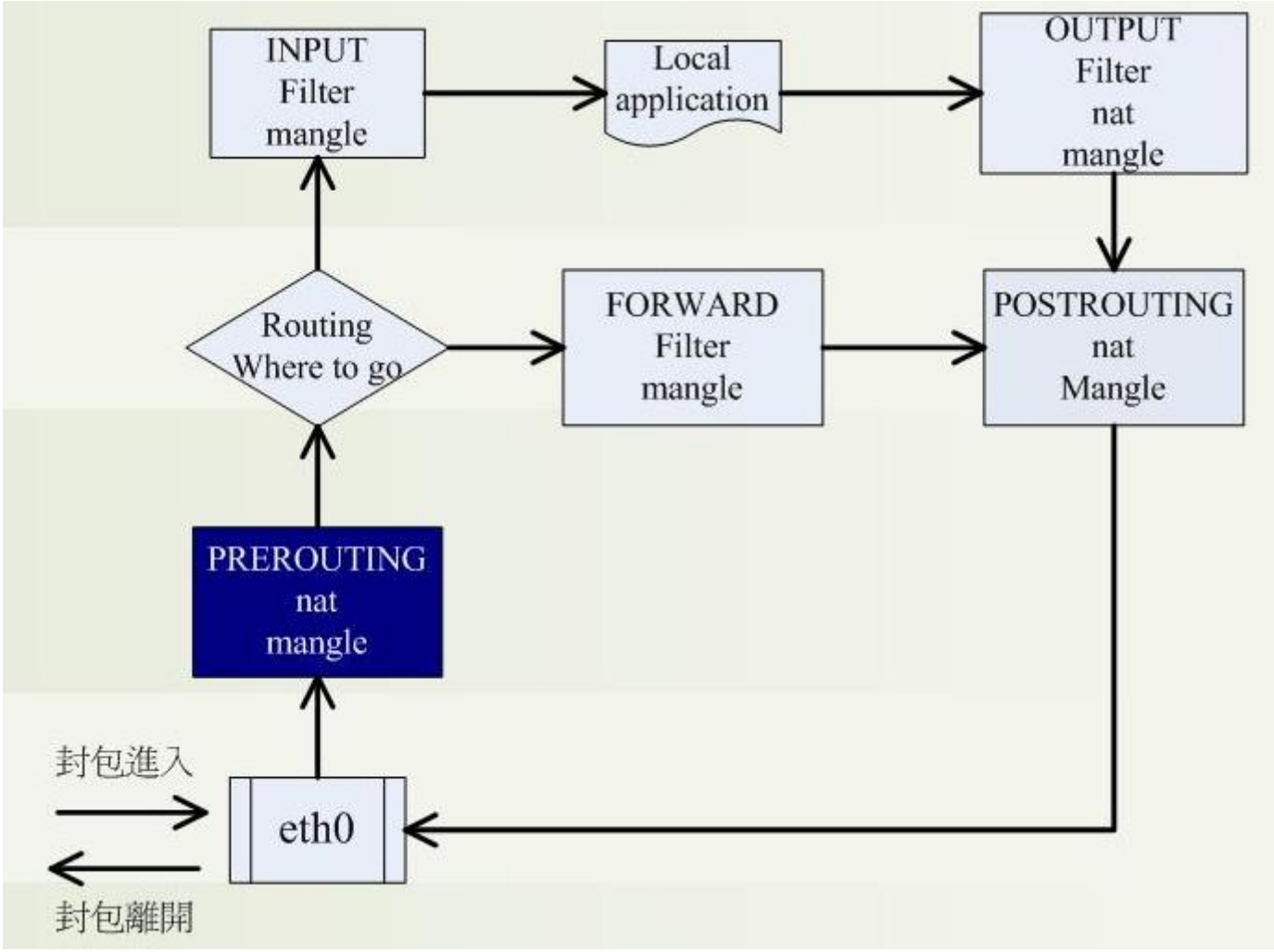
- 設定讓192.168.0.0/24這個網段的電腦可以透過163.18.22.68這個匝道伺服器連線到外部。
- -o參數後面所接的裝置是對外的網路卡。



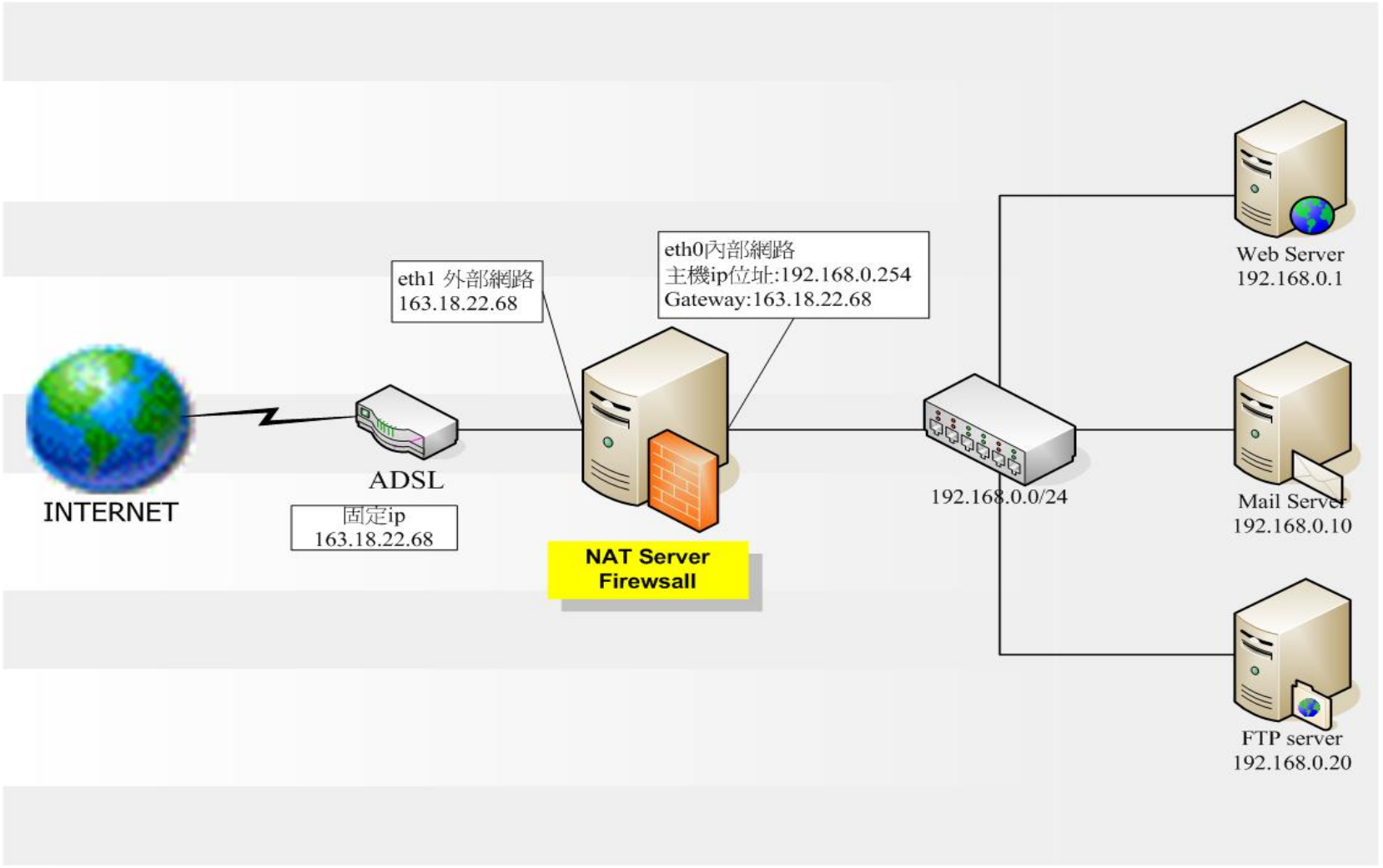
清除轉址表格的設定


root@dns:~
File Edit View Terminal Tabs Help
[root@dns ~]# iptables -t nat -A POSTROUTING -o eth1 -s 192.168.0.0/24 -j MASQUERADE
[root@dns ~]#
[root@dns ~]# iptables -L -t nat
Chain OUTPUT (policy ACCEPT)
target prot opt source destination
Chain POSTROUTING (policy ACCEPT)
target prot opt source destination
MASQUERADE all -- 192.168.0.0/24 anywhere
Chain PREROUTING (policy ACCEPT)
target prot opt source destination
[root@dns ~]#
[root@dns ~]# iptables -F -t nat
[root@dns ~]#
[root@dns ~]# iptables -L -t nat
Chain OUTPUT (policy ACCEPT)
target prot opt source destination
Chain POSTROUTING (policy ACCEPT)
target prot opt source destination
Chain PREROUTING (policy ACCEPT)
target prot opt source destination
[root@dns ~]#

設定NAT上的服務轉到內部的主機

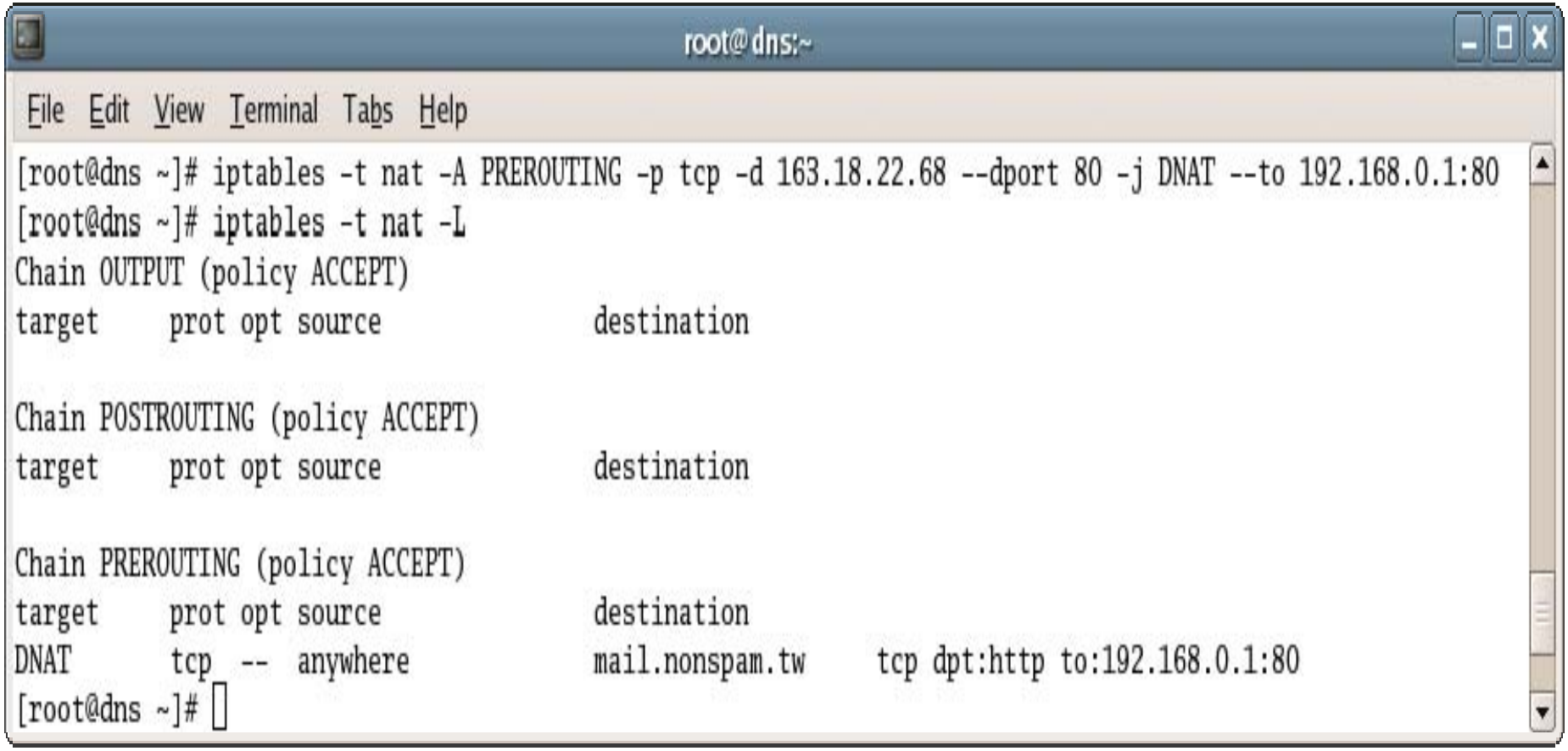


設定NAT上的服務轉到內部的主機



設定NAT上的服務轉到內部的主機

- 設定由外部連往本機163.18.22.1
  埠號80的服務轉向內部主機192.168.0.1



Standard Source NAT 通用語法

- 通用語法

 # iptables -t nat -A POSTROUTING -o 外送介面 -j SNAT -to--source 來源位址

- 說明
  - 來源位址: 單一位址、一段位址、連接port範圍(TCP/UDP)

- 範例
  - 將封包來源位址變更為1.2.3.4

 # iptables -t nat -A POSTROUTING -o eth0 -j SNAT -to--source 1.2.3.4

Standard Destination NAT 通用語法

- 通用語法

iptables -t nat -A PREROUTING -i 內送介面 -j DNAT -to--destination 目的地位址

iptables -t nat -A OUTPUT -o 外送介面 -j DNAT -to--destination 目的地位址

- 範例
  - 將封包目的地位址變更為5.6.7.8

iptables -t nat -A PREROUTING -i eth1 -j DNAT -to--destination 5.6.7.8

  - 將原本要送到5.6.7.8的封包，改送到127.0.0.1

iptables -t nat -A OUTPUT -d 5.6.7.8 -j DNAT -to--destination 127.0.0.1

iptables的儲存與還原方法

- 利用iptables指令所設的規則，在主機重新開機後，所有規則就會消失，為了避免重新輸入，可使用*iptables-save*指令，將所設定的規則儲存於檔案中。

iptables的儲存與還原方法

iptables-save > /etc/iptables0925

- 將設定的規則儲存至/etc的下的iptables0925檔案

iptables-restore < /etc/iptables0925

- 將儲存的規則iptables0925檔案還原至系統

手動啟動與開機自動啟動

- 開機時自動啟動：

chkconfig iptables on

- 手動啟動：

/etc/init.d/iptables start

- 手動關閉：

/etc/init.d/iptables stop

- 重新啟動：

/etc/init.d/iptables restart

封包的處理

項目	說明
IP位址	針對網路封包的來源ip位址、目的地ip位址或MAC位址進行比對。
通訊port	針對網路封包的通訊port資訊比對。
通訊協定	針對所指定的通訊協定比對。
網路介面	針對網路介面所傳送的封包比對。
切割	將資訊進行切割，以符合網路介面傳送上的要求。
數量	針對網路封包的數量進行比對。

常用的設定參數

- 設定檢察目標的語法格式

參數	功能
-t	指定表格名稱
-I	在鏈的最前面插入規則
-A	在鏈的最後面附加規則
-D	在鏈中刪除某個規則
-N	自訂新鏈

封包比對的規則

參數	功能
-I	指定封包進入的網路介面
-o	指定封包出去的網路介面
-p	指定封包的類型（TCP、UDP、ICMP)
-s	指定封包的來源ip位址
-d	指定封包的目的ip位址
--sport	指定封包來源端的通訊port
--dport	指定封包目的端的通訊port

處理封包的動作

參數	功能
-j 處理方式	設定規則的處理方式
-p處理方式	設定鏈的預設處理原則

設定規則的處理動作

參數	功能
DROP	丟棄該封包，且不回應任何錯誤訊息
REJECT	丟棄該封包，回應錯誤訊息
ACCEPT	允許封包通過

其他常用參數

參數	功能
-L	列出中所有規則
-n	不要將ip反查為網域名稱
-F	清除規則
-X	移除自訂的鏈

Module 8-3： FireStarter(**)

Module 8-3： FireStarter

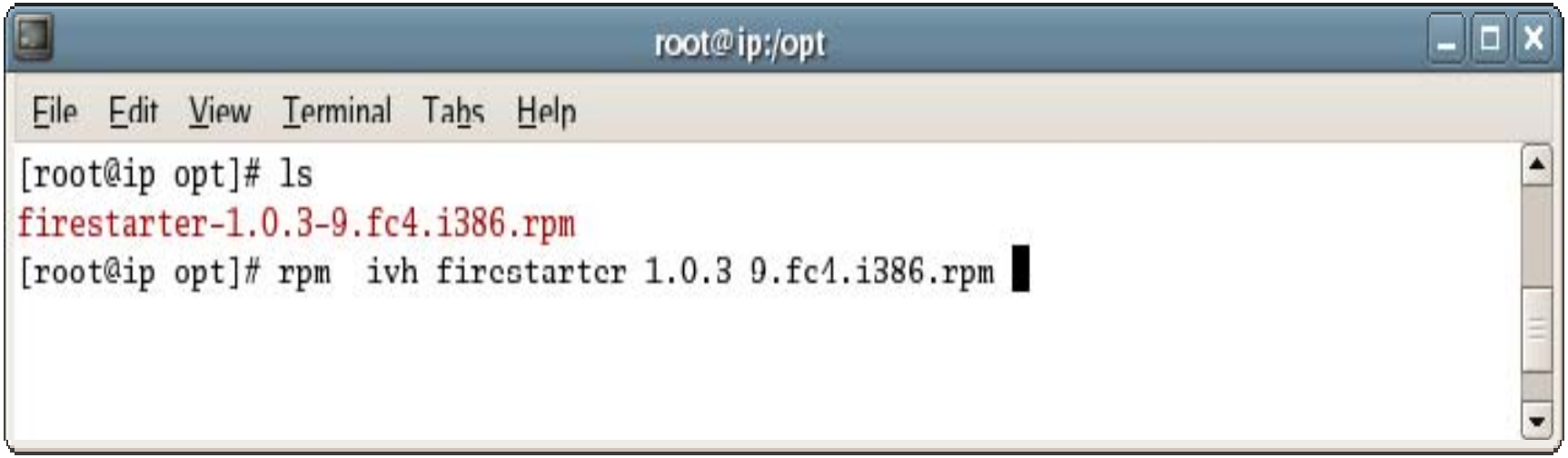
- FireStarter是一款open source的視覺化防火牆配置工具，以GTK+為基礎，可以直接在X Window圖形介面中管理防火牆與制訂規則，目前大多數流行的Linux Distribution皆能執行。

FireStarter官方網站

- http://www.fs-security.com/
- 提供多種Linux Distribution安裝版本，包括：
  - Fedora
  - Red Hat Enterprise
  - SuSE
  - Ubuntu
  - Debian
  - Mandrake

利用RPM套件安裝FireStarter

- 以Fedora core 4為例，下載符合的RPM檔
- http://fedoraproject.org/extras/4/i386/firestarter-1.0.3-9.fc4.i386.rpm
- 安裝rpm檔案如下圖

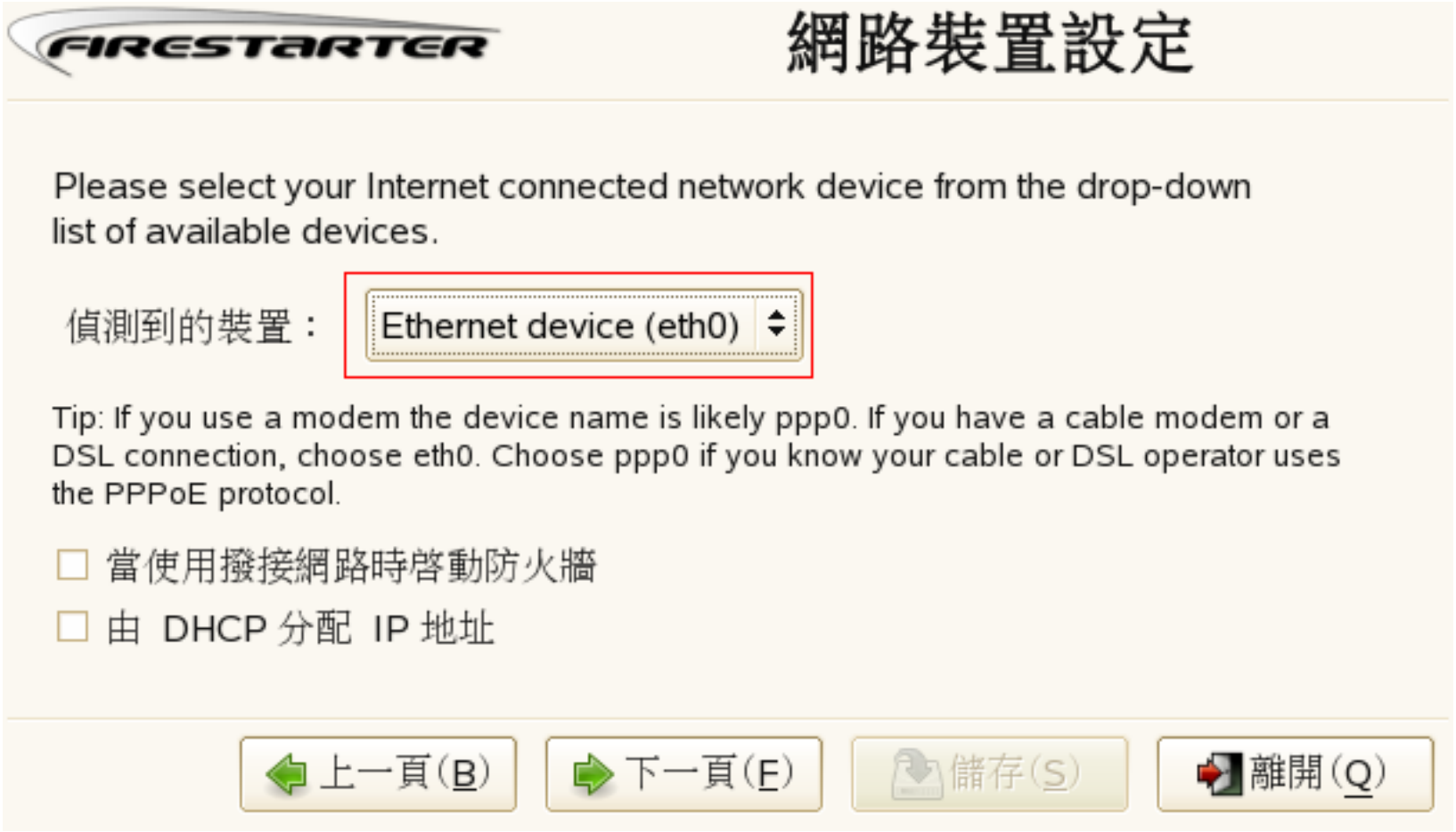


利用Source Tarball安裝

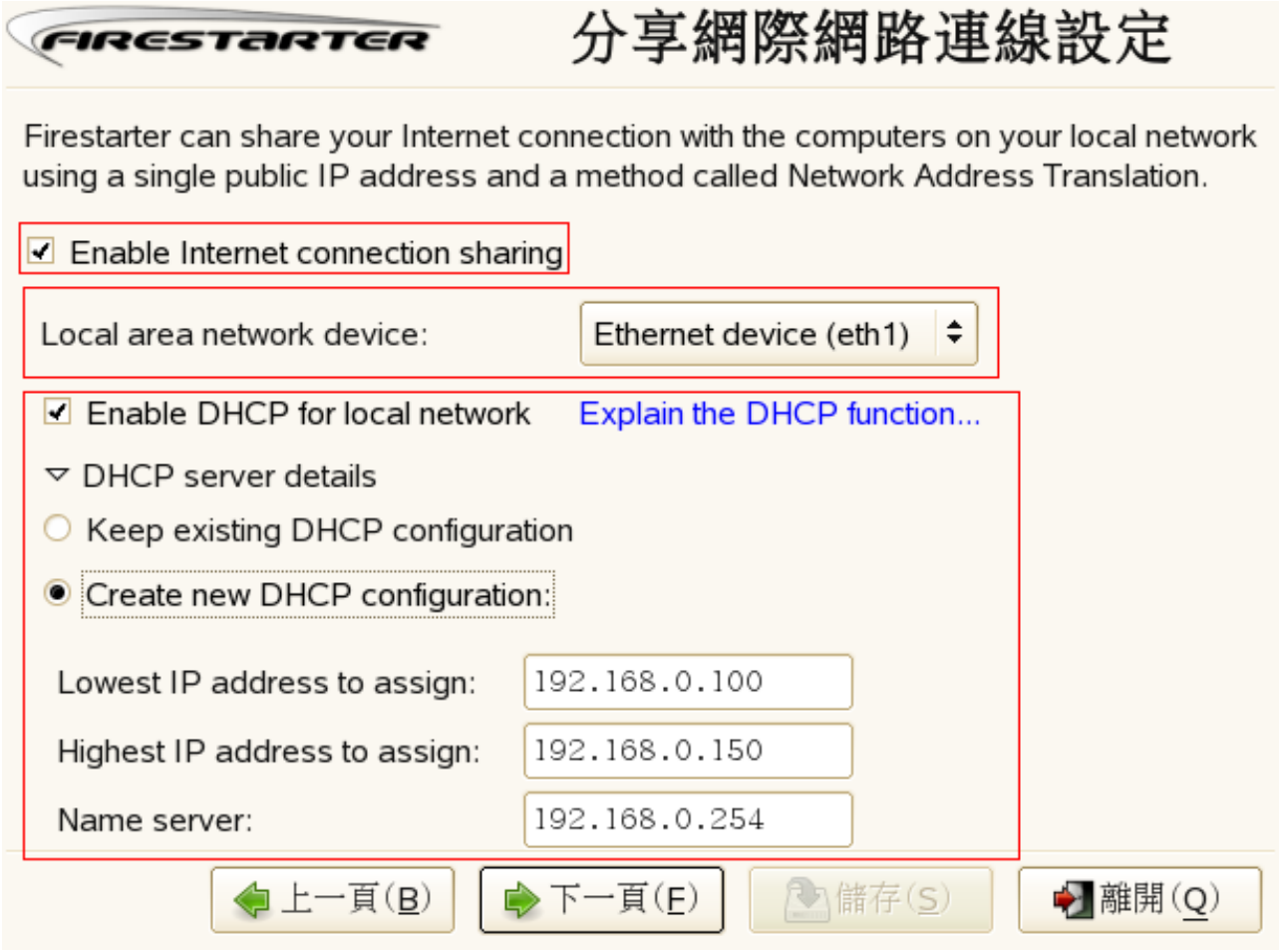
- 於官網下載Source Tarball檔案自行編譯安裝
- http://www.fs-security.com/download.php

```
# tar zxvf firestarter-1.0.3.tar.gz
# cd firestarter-1.0.3
# ./configure –sysconfdir=/etc
# make
# make install
```

FireStarter的初始化設定



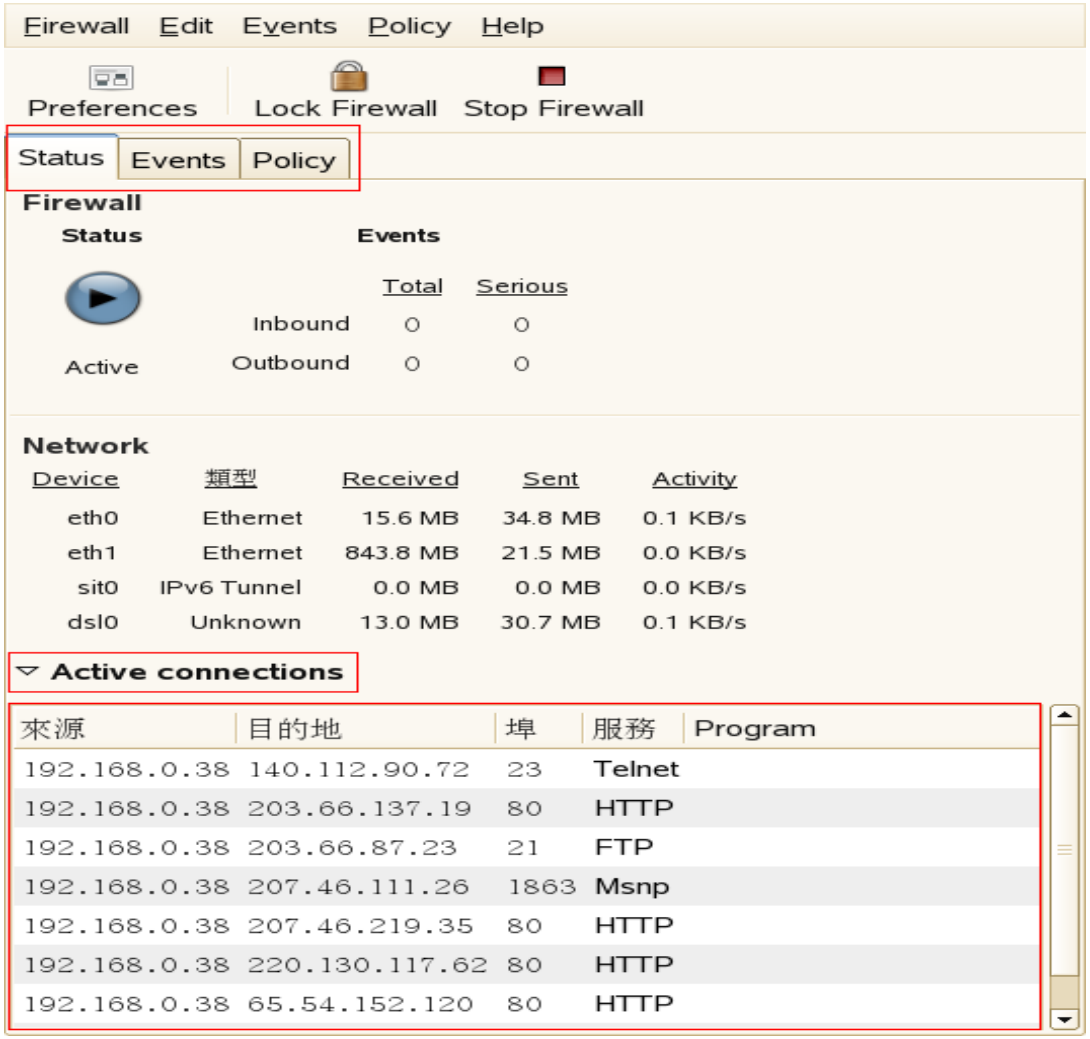
設定網路共享服務


FIRESTARTER
分享網際網路連線設定
Firestarter can share your Internet connection with the computers on your local network using a single public IP address and a method called Network Address Translation.
Enable Internet connection sharing
Local area network device:
Ethernet device (eth1)
Enable DHCP for local network
Explain the DHCP function...
DHCP server details
Keep existing DHCP configuration
Create new DHCP configuration:
Lowest IP address to assign:
192.168.0.100
Highest IP address to assign:
192.168.0.150
Name server:
192.168.0.254
上一頁(B)
下一頁(F)
儲存(S)
離開(Q)

啟動防火牆


FIRESTARTER
準備啓動防火牆
The wizard is now ready to start your firewall.
Press the save button to continue, or the back button to review your choices.
Start firewall now
Tip: If you are connecting to the firewall host remotely you might want to defer starting the firewall until you have created additional policy.
上一頁(B)
下一頁(F)
儲存(S)
離開(Q)

FireStarter 執行畫面



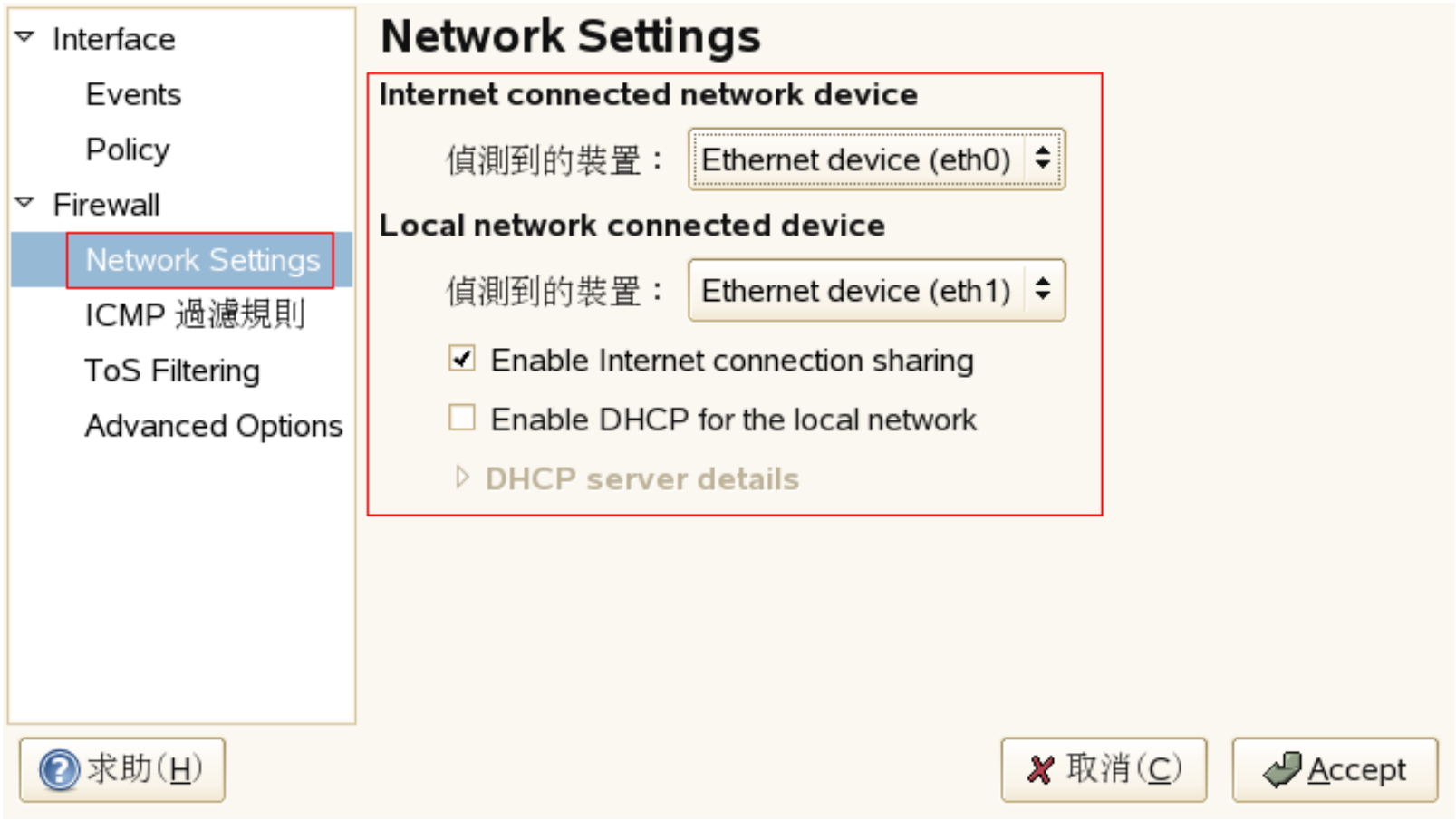
記錄事件


Interface
Events
Policy
Firewall
Network Settings
ICMP 過濾規則
ToS Filtering
Advanced Options
Events List
Blocked connections
Skip redundant entries
Skip entries where the destination is not the firewall
Do not log events for the following
Hosts
Ports
移除(R)
新增(A)
移除(R)
新增(A)
求助(H)
取消(C)
Accept

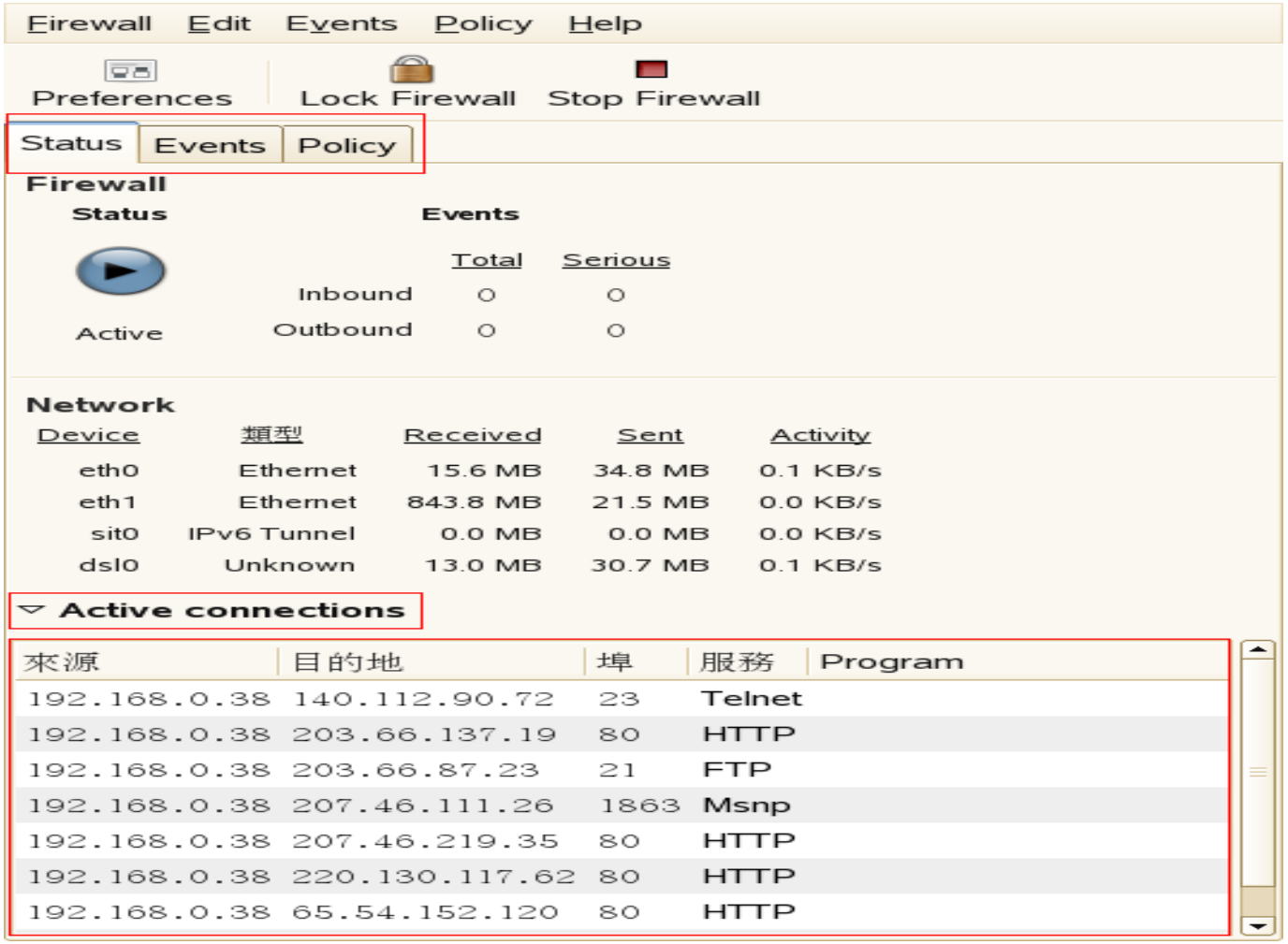
Policy


Interface
Events
Policy
Firewall
Network Settings
ICMP 過濾規則
ToS Filtering
Advanced Options
Policy Editor
Rule editing
Apply policy changes immediately
求助(H)
取消(C)
Accept

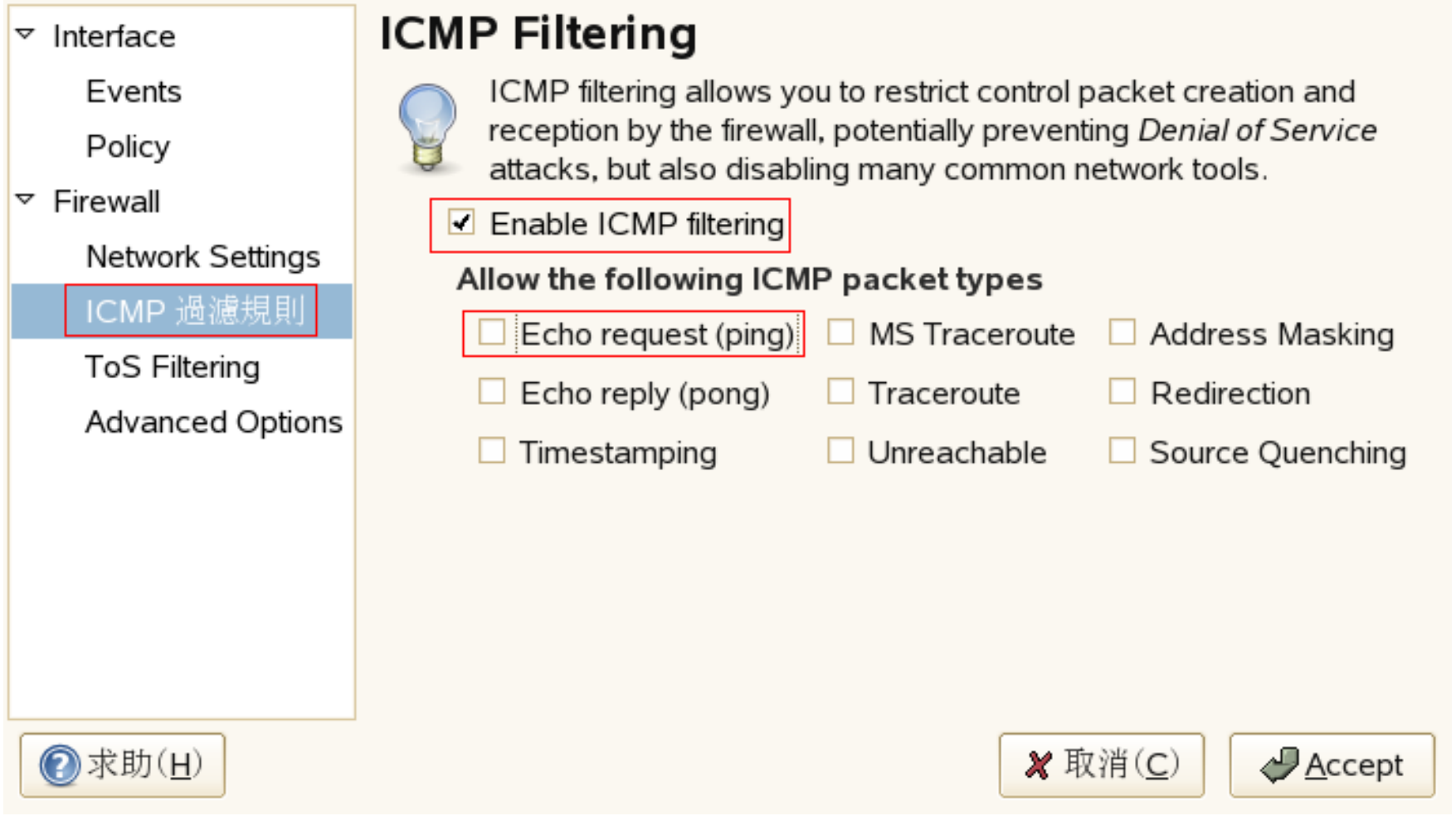
設定NAT服務


Interface
Events
Policy
Firewall
Network Settings
ICMP 過濾規則
ToS Filtering
Advanced Options
Network Settings
Internet connected network device
偵測到的裝置：
Ethernet device (eth0)
Local network connected device
偵測到的裝置：
Ethernet device (eth1)
Enable Internet connection sharing
Enable DHCP for the local network
DHCP server details
求助(H)
取消(C)
Accept

NAT服務運作情形

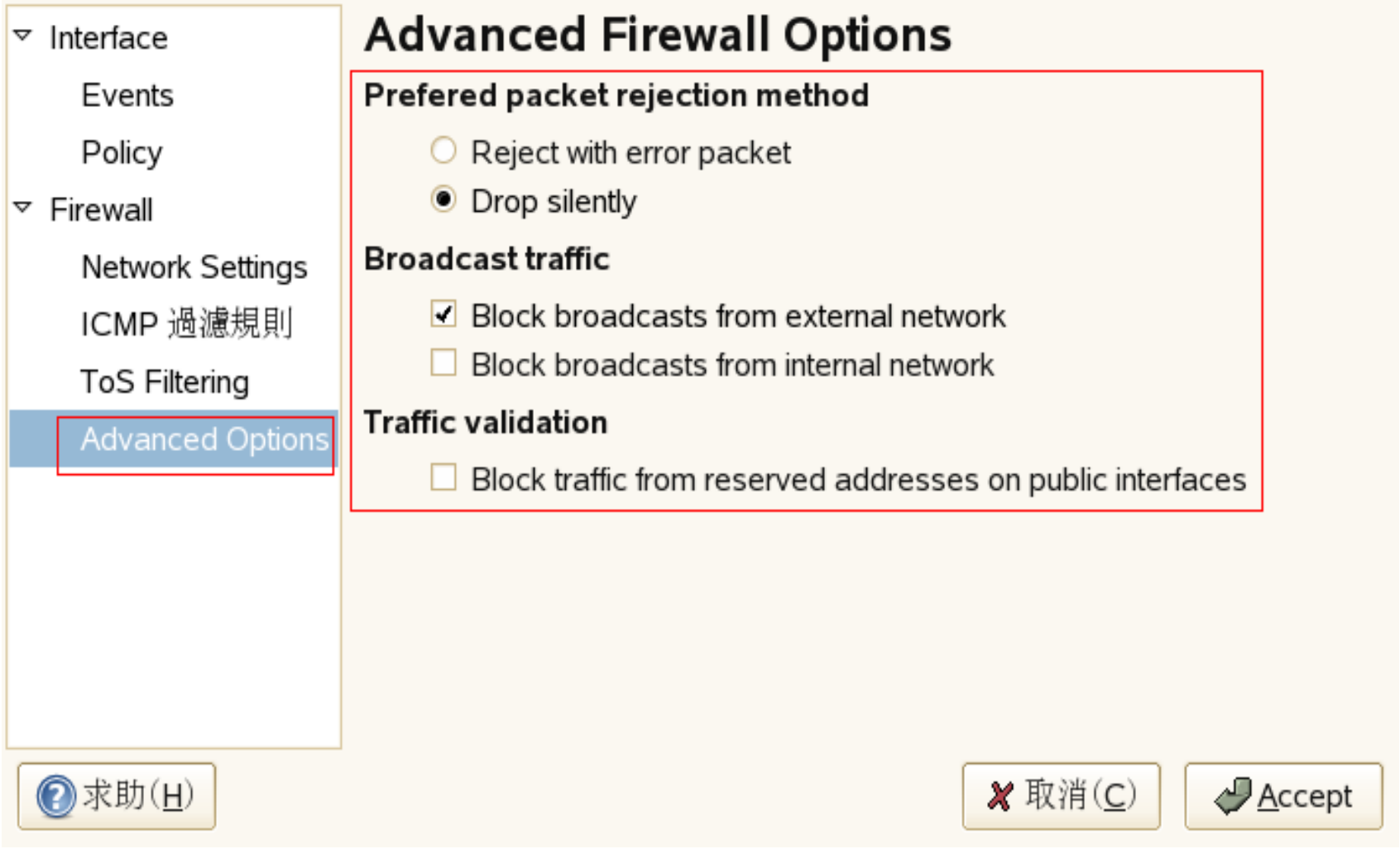

Firewall Edit Events Policy Help
Preferences
Lock Firewall
Stop Firewall
Status
Events
Policy
Firewall
Status
Events
Total
Serious
Inbound 0 0
Outbound 0 0
Active
Network
Device 類型 Received Sent Activity
eth0 Ethernet 15.6 MB 34.8 MB 0.1 KB/s
eth1 Ethernet 843.8 MB 21.5 MB 0.0 KB/s
sit0 IPv6 Tunnel 0.0 MB 0.0 MB 0.0 KB/s
dsl0 Unknown 13.0 MB 30.7 MB 0.1 KB/s
Active connections
來源 目的地 埠 服務 Program
192.168.0.38 140.112.90.72 23 Telnet
192.168.0.38 203.66.137.19 80 HTTP
192.168.0.38 203.66.87.23 21 FTP
192.168.0.38 207.46.111.26 1863 Msnp
192.168.0.38 207.46.219.35 80 HTTP
192.168.0.38 220.130.117.62 80 HTTP
192.168.0.38 65.54.152.120 80 HTTP

ICMP過濾規則


Interface
Events
Policy
Firewall
Network Settings
ICMP 過濾規則
ToS Filtering
Advanced Options
ICMP Filtering
ICMP filtering allows you to restrict control packet creation and reception by the firewall, potentially preventing Denial of Service attacks, but also disabling many common network tools.
Enable ICMP filtering
Allow the following ICMP packet types
Echo request (ping)
MS Traceroute
Address Masking
Echo reply (pong)
Traceroute
Redirection
Timestamping
Unreachable
Source Quenching
求助(H)
取消(C)
Accept

Type of Service

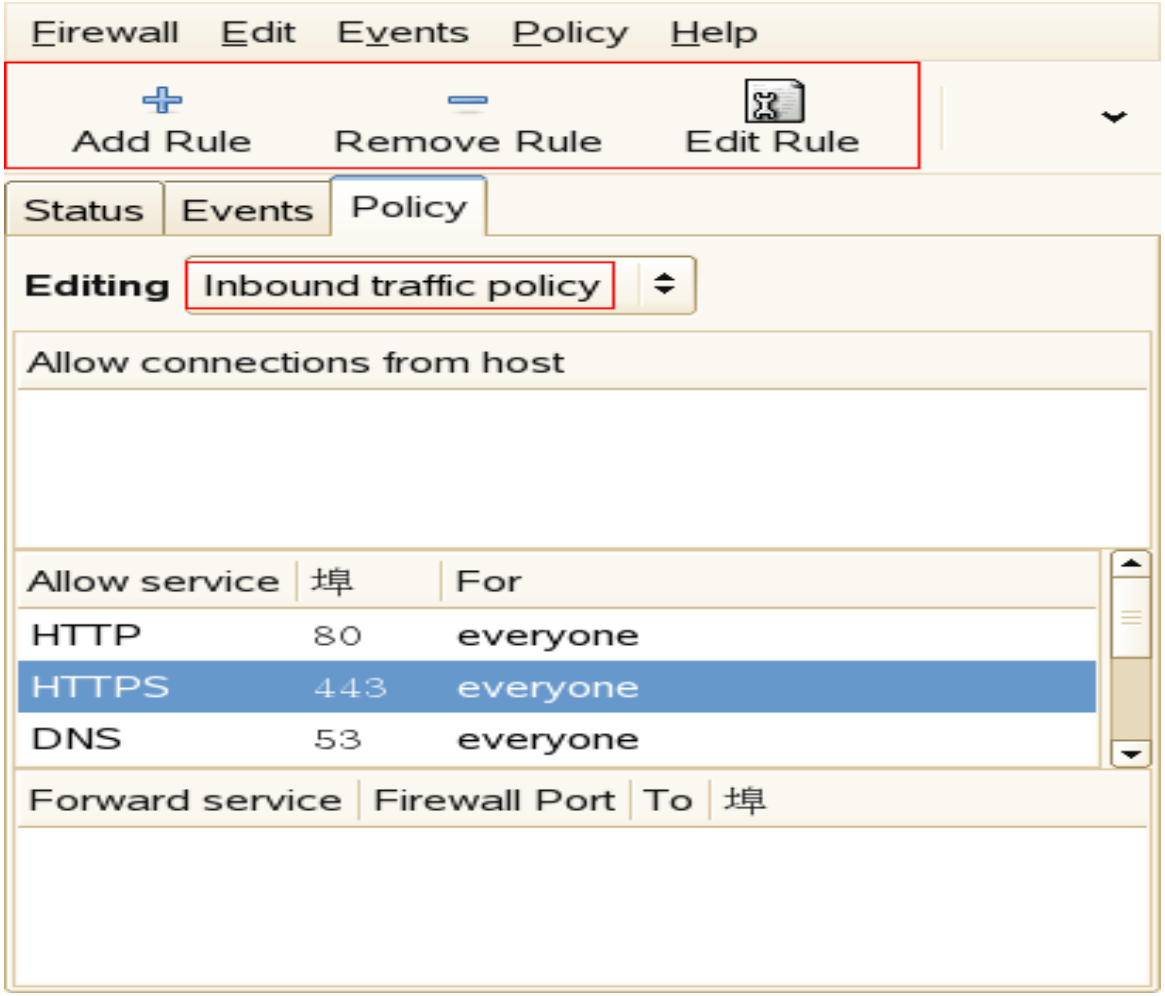
FireStarter進階設定



Policy的制定

- Inbound
  - 來自Internet到Local網段或防火牆主機
- Outbound
  - 由防火牆主機的Local網段連線到Internet

Inbound traffic policy設定

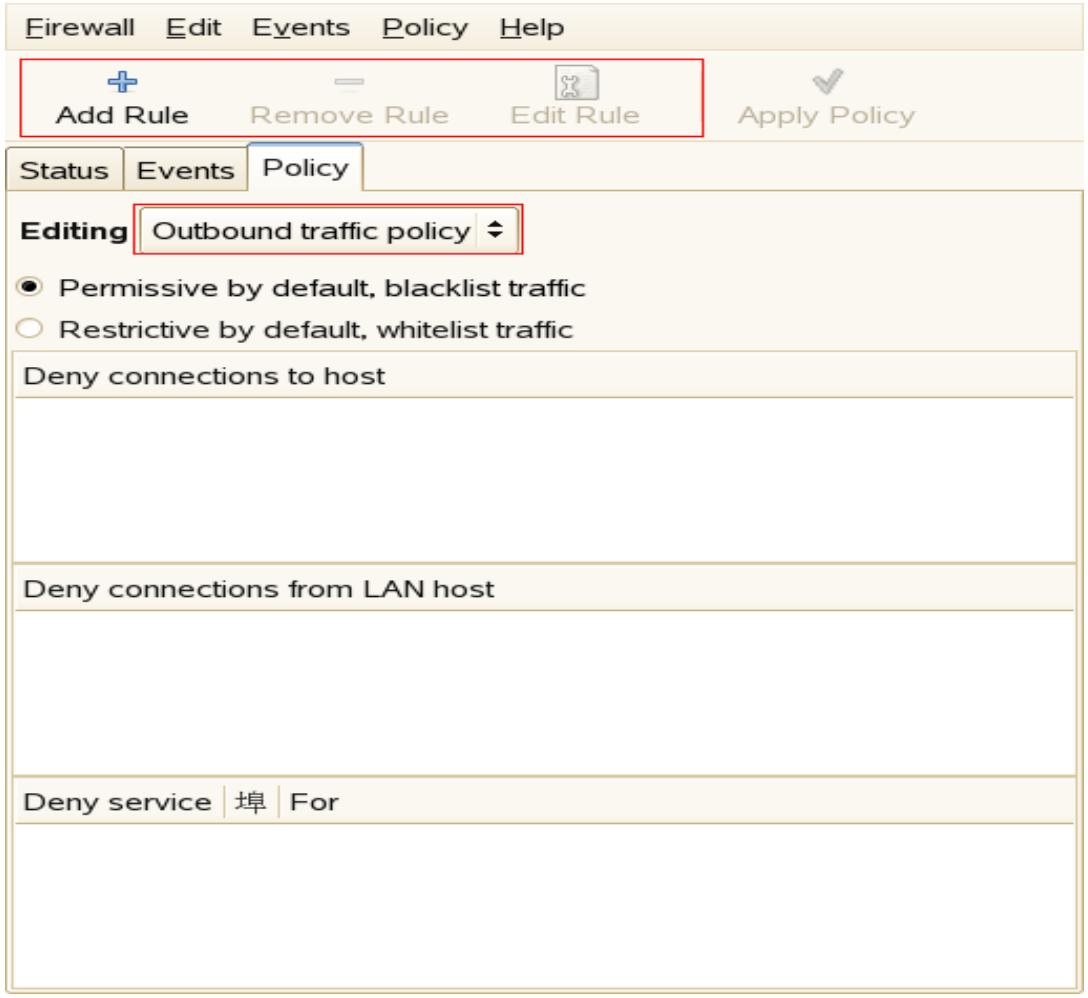


Add Inbound policy Rule

轉送網路服務


Forward service from firewall
Name
FTP
埠
20–21
To internal host
IP or host
192.168.0.100
埠
20–21
Comment
取消(C)
新增(A)

Outbound traffic policy設定



Add outbound policy Rule

結論

- 『防外不防內』是一般防火牆設定的通則。即限制外部的連線，而不阻擋由內部向外部的連線請求。
- 這樣的原則是為了確保本身的安全性，當要設定網路規則前,必須先確認被限制或允許的目標，如此才能夠制定出符合需求的規則。

習 題

習題一

- 說明netfilter比對rule的原則為何？

習題二

- 假設你發現目前有一個ip位置為192.168.0.100的使用者對你的主機嘗試使用ssh連線，你該如何禁止該ip位置的連線？

習題三

- 檢查你本機目前所設定的iptables規則，並將所有規則清空。

習題四

- 利用iptables-save，將你目前所設定的規則另存成一份檔案，並利用日期為該檔案命名。

習題五

- 先清空目前系統中的iptables的所有規則，再利用iptables-restore，將上題所存規則重新載入。

習題六

- 設定每次開機時，都自動建立iptables規則。

Module 8-4:專案實作(**)

專案目的

- 學習建置Firewall應用環境。
- 利用實際操作的方式讓同學了解iptables運作原理。
- 切割信任網段與不信任網段，例如子網路與外部網路。
- 劃分出可提供internet的服務與必須受保護的服務。
- 撰寫shell script檔案，編輯預設防火牆規則，隨各種不同環境重新編輯運用。

專案實作（一）

專案描述

參考環境配置圖，架設一台NAT Server，並完成下列要求：

① 啟用IP FORWARD功能。

② 架設nat server，使子網路內的所有電腦能透過這台nat server連線到Internet 。

③ 將NAT Server設定為DROP所有icmp封包。

④ 於子網路中架設一台ftp server，讓外部網路可以透過nat server存取資料。

專案描述

⑤ 於子網路中架設一台web server，讓外部網路可以透過nat server瀏覽。

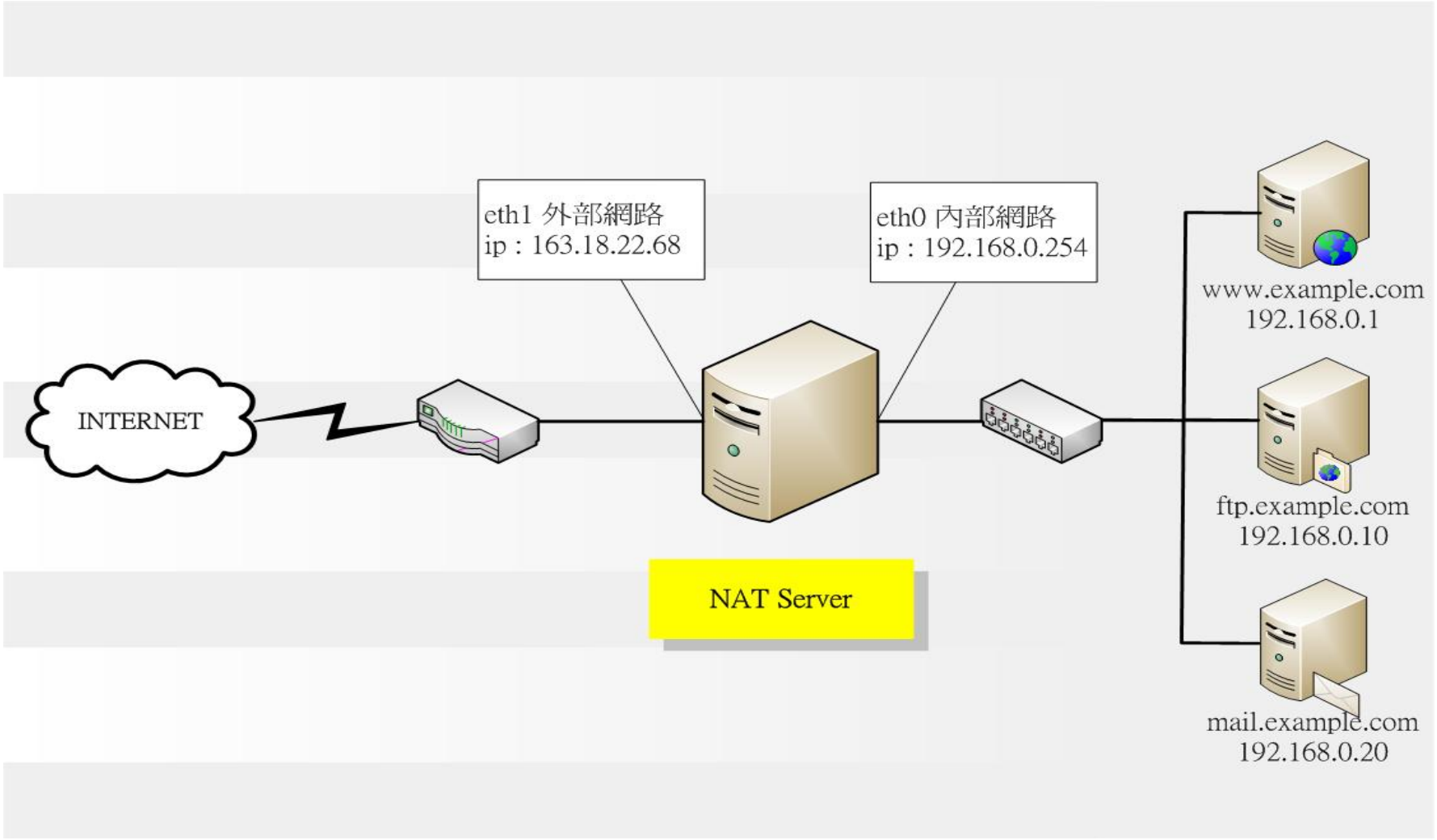
⑥ 於子網路中架設一台mail server，讓外部網路可以透過nat server收發mail。

⑦ 只允許子網路192.168.0.0/24存取ftp服務，禁止外部網路存取ftp server。

⑧ 允許所有網路位置存取web server，只禁止來自192.168.0.100這個ip位置的主機存取web server。

⑨ 完成上列要求後，撰寫一shell script檔案，設定為當主機重新啟動時可自動讀取該檔案，並同時完成以上8項題目要求。

環境配置圖


eth1 外部網路
ip：163.18.22.68
eth0 內部網路
ip：192.168.0.254
INTERNET
NAT Server
www.example.com
192.168.0.1
ftp.example.com
192.168.0.10
mail.example.com
192.168.0.20

專案實作（二）

專案說明

- 專案實作（一）中，利用iptables實作了NAT 的服務，雖然做到了頻寬共享的功能，但是並沒有使用到iptables防火牆的功能。

- 在專案實作（二）中，利用FireStarter套件，並整合MRTG套件與SNMP 簡易網路管理協定及DHCP服務，實作一個功能完整的IP分享器。

- 專案實作（二）將以共享一個ADSL連線做為範例。

專案描述

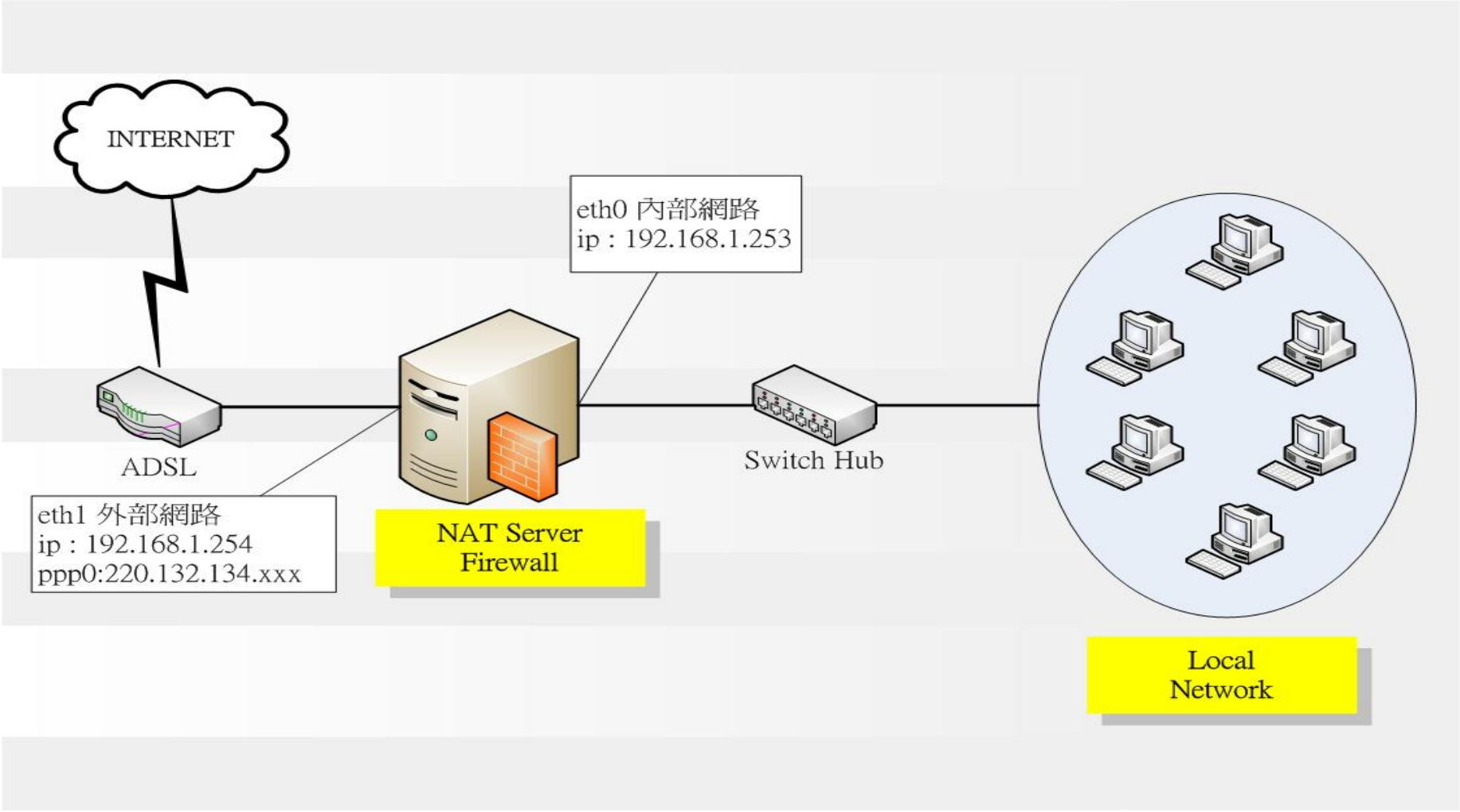
① 設定ADSL連線。

② 安裝FireStarter 套件，並設定子網路可由DHCP自動取得IP 位址。

③ 設定防火牆規則，阻擋子網路內的電腦使用p2p軟體。

④ 利用MRTG與SNMP套件，設定管理者可由web瀏覽器直接觀察網路使用情形。

環境配置圖


INTERNET
eth0 內部網路
ip：192.168.1.253
ADSL
Switch Hub
eth1 外部網路
ip：192.168.1.254
ppp0:220.132.134.xxx
NAT Server
Firewall
Local
Network

參考文獻

1. 施威銘研究室，Linux iptables技術實務 防火牆、頻寬管理、連線管制，旗標出版股份有限公司
2. 李蔚澤，iptables與Linux安全防護，碁峯資訊股份有限公司
3. 酆士昌，Linux架站與管理 應用範例大全集， 博碩文化股份有限公司
4. 鳥哥的Linux伺服器架設篇，上奇科技出版事業處
5. 蔡一郎、邱敏乘，Linux網管技術
6. Linuxpilot 國際中文版 51期
7. 網管人 2006 第三期
8. http://www.netfilter.org/
9. http://www.fs-security.com/