How to setup a linux firewall

Jing, mqjing@gmail.com

Note

- 請不要在遠端主機上進行防火牆的練習，因為你很有可能一不小心將自己關在家門外！

Quick

Not allow to respond any request.

```
範例：將本機的 INPUT 設定為 DROP，其他設定為 ACCEPT
[root@www ~]# iptables -P INPUT   DROP
[root@www ~]# iptables -P OUTPUT  ACCEPT
[root@www ~]# iptables -P FORWARD ACCEPT
[root@www ~]# iptables-save
```

Lots Firewall Example

Document

Why

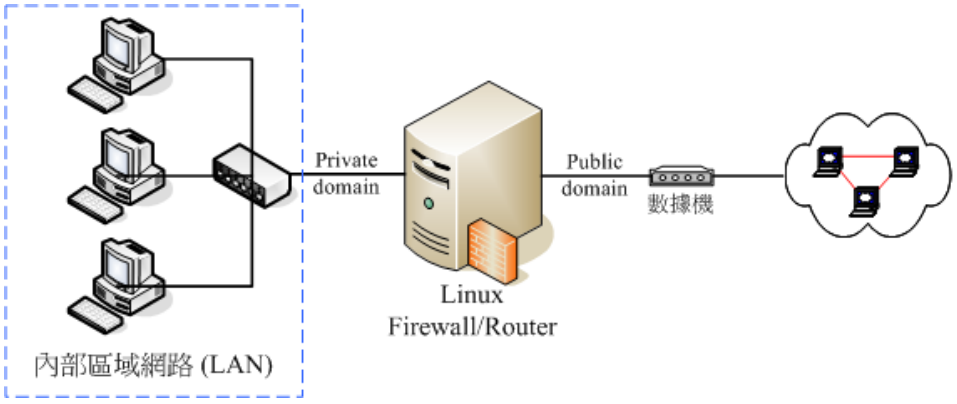
- 畢竟網路世界是很複雜的，而 Linux 主機也不是一個簡單的東西，說不定哪一天你在進行某個軟體的測試時，主機突然間就啟動了一個網路服務，如果你沒有管制該服務的使用範圍，那麼該服務就等於對所有 Internet 開放，那就麻煩了！
    - 可以限制檔案傳輸服務 (FTP) 只在子網域內的主機才能夠使用，而不對整個 Internet 開放
    - 限制整部 Linux 主機僅可以接受客戶端的 WWW 要求，其他的服務都關閉
    - 限制整部主機僅能主動對外連線。反過來說，若有用戶端對我們主機發送主動連線的封包狀態 (TCP 封包的 SYN flag) 就予以抵擋

Linux 系統上防火牆的主要類別

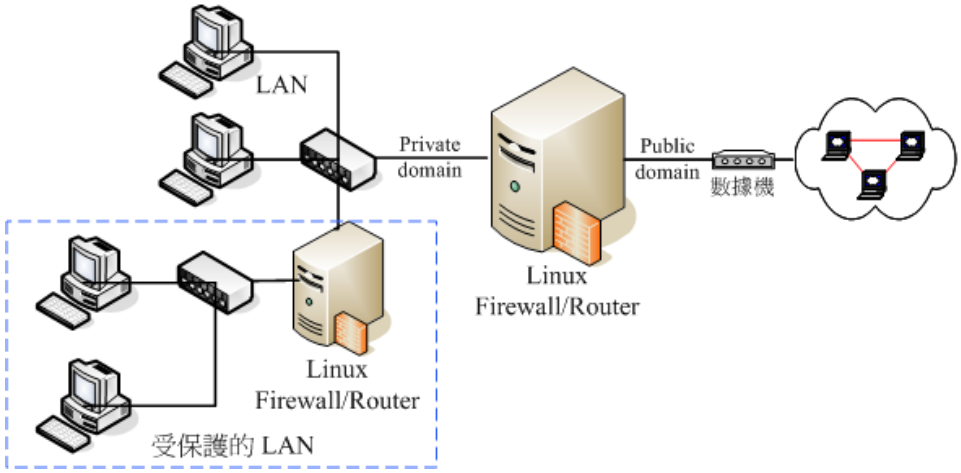
- Netfilter (封包過濾機制) -- iptables: 即是分析進入主機的網路封包，將封包的表頭資料捉出來進行分析，以決定該連線為放行或抵擋的機制
    - Netfilter 提供了 iptables 這個軟體來作為防火牆封包過濾的指令. 利用一些封包過濾的規則設定，來定義出什麼資料可以接收，什麼資料需要剔除
- TCP Wrappers (程式控管): 分析該伺服器程式誰能夠連線、誰不能連線.
    - Example: 那麼你只要知道 FTP 的軟體名稱 (vsftpd)，然後對他作限制，則不管 FTP 啟動在哪個埠口，都會被該規則管理的.
- Proxy (代理伺服器)

防火牆的一般網路佈線

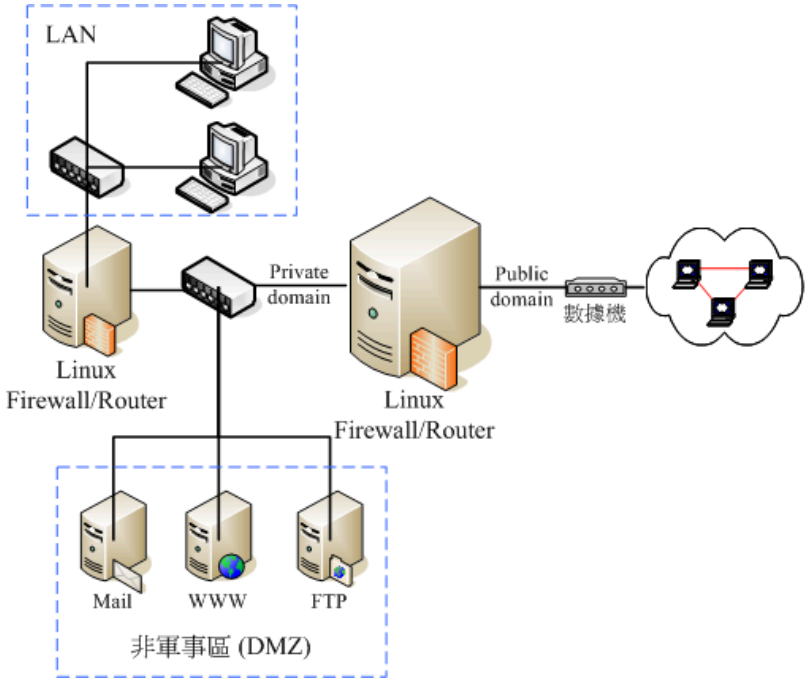
- 單一網域，僅有一個路由器: 架設在路由器上面以管控整個區域網路的封包進出。



- 內部網路包含安全性更高的子網路



- 在防火牆的後面架設網路伺服器主機: 將提供網路服務的伺服器放在防火牆後面
    - 如 Web, Mail 與 FTP 都是透過防火牆連到 Internet 上面去，所以在 Public 的 IP 都是一樣的！ 只是透過防火牆的封包分析後，將 WWW 的要求封包轉送到 Web 主機，將 Mail 送給 Mail Server 去處理而已(透過 port 的不同來轉遞)



- 如某些使用者不良操作導致中毒啊、被社交工程攻陷導致內部主機被綁架啊等等的，是不會影響到網路伺服器的正常運作的

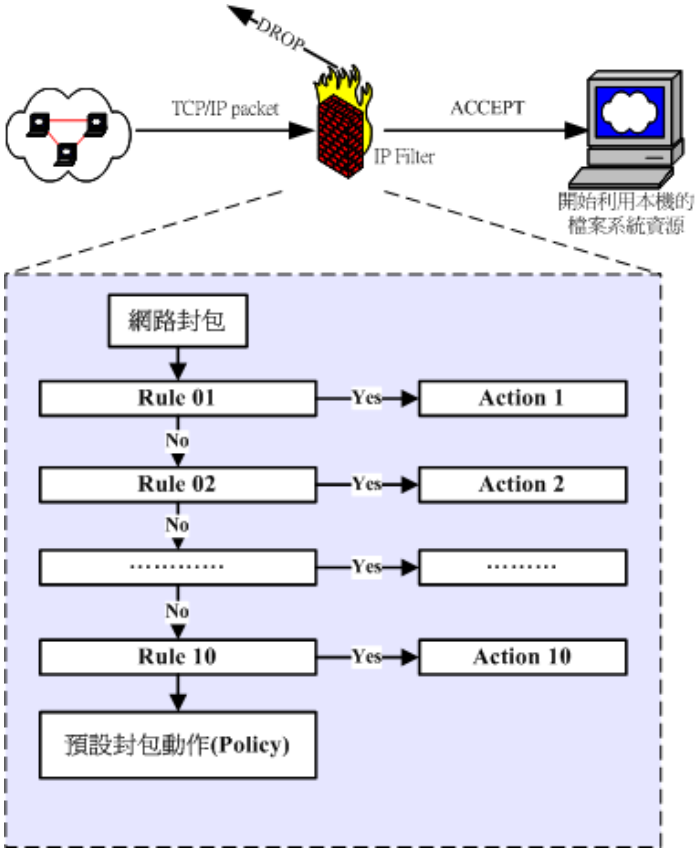
防火牆的使用

Netfilter 機制

- 拒絕讓 Internet 的封包進入主機的某些埠口. ex: close port 21
- 拒絕讓某些來源 IP 的封包進入. block source IP
- 拒絕讓帶有某些特殊旗標 (flag) 的封包進入
- 分析硬體位址 (MAC) 來決定連線與否
-

封包進入流程：規則順序的重要性

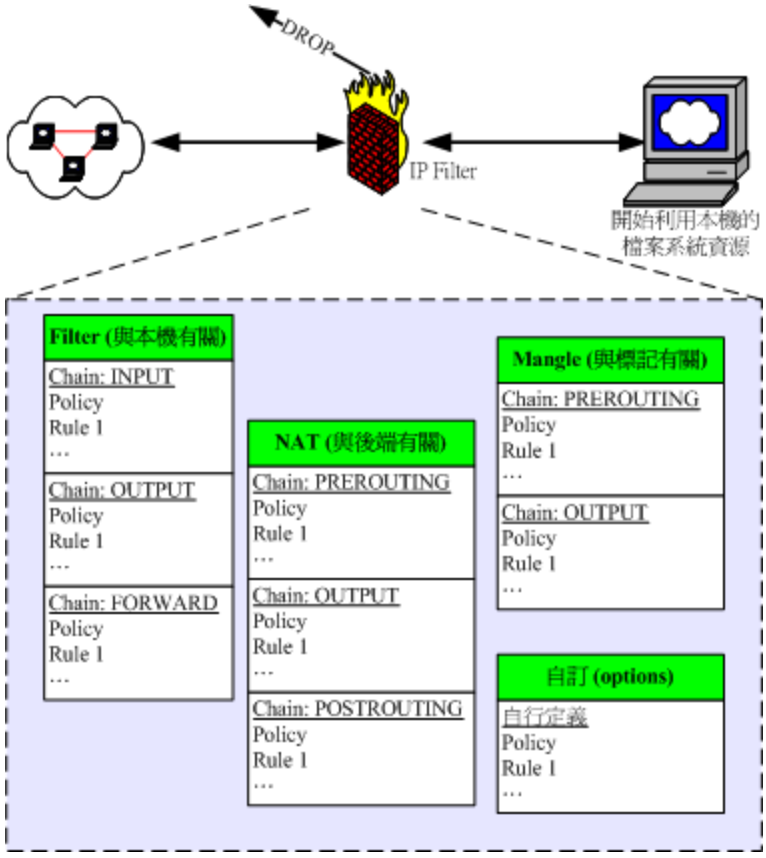
- 檢查通過則接受 (ACCEPT) 進入本機取得資源，如果檢查不通過，則可能予以丟棄 (DROP)



Example Rule Setup
Rule 1 先讓要求 WWW 服務的封包通過;
Rule 2 再抵擋 192.168.100.100 ;
Rule 3 將所有的封包丟棄。

iptables 的表格 (table) 與鏈 (chain)

- 預設的情況下，Linux 的 iptables 至少就有三個表格
  - filter: 管理本機進出
    - INPUT: 想要進入我們 Linux 本機的封包有關
    - OUTPUT: 主要與我們 Linux 本機所要送出的封包有關
  - nat: 管理後端主機 (防火牆內部的其他電腦)
    - 主要在進行來源與目的之 IP 或 port 的轉換
    - PREROUTING:在進行路由判斷之前所要進行的規則(DNAT/REDIRECT)
    - POSTROUTING:在進行路由判斷之後所要進行的規則 (SNAT/MASQUERADE)
    - OUTPUT:與發送出去的封包有關
  - mangle: 管理特殊旗標使用
    - 主要是與特殊的封包的路由旗標有關

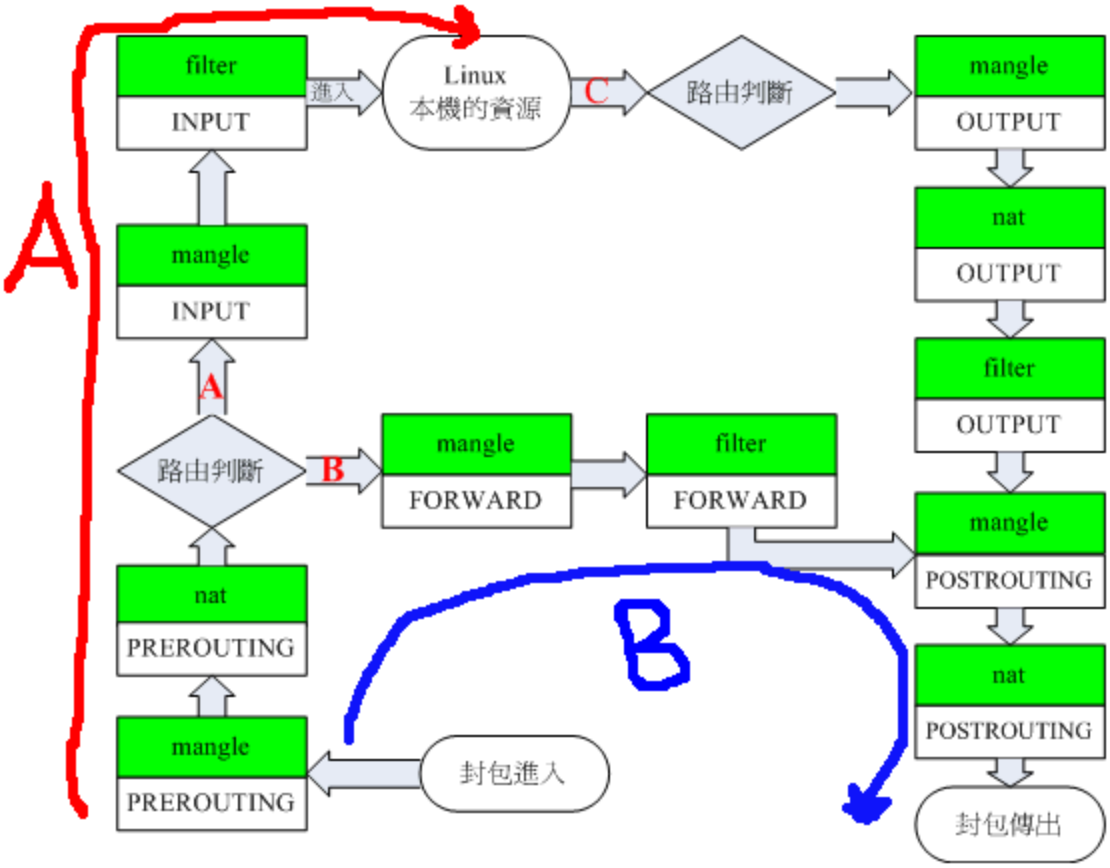


- Rule Application
    - 如果你的 Linux 是作為 www 服務, for www 要求有回應, 那麼你就得要處理 filter 的 INPUT 鏈
    - 如果你的 Linux 是作為區域網路的路由器, 那麼就得要分析 nat 的各個鏈以及 filter 的 FORWARD 鏈才行

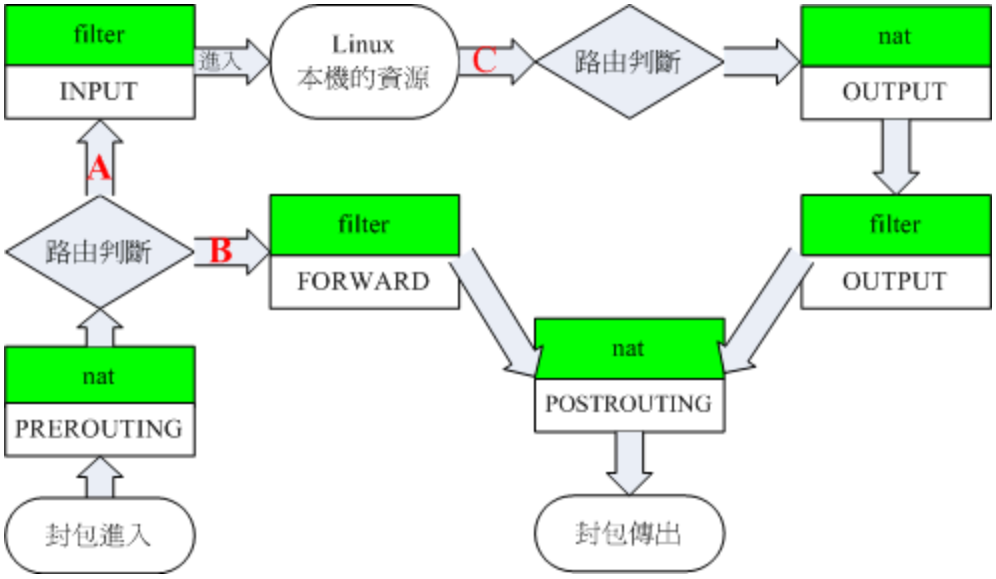
iptables 內建各表格與鏈的相關性

兩個『路由判斷』

- 1 => determine the package target is the host or just forward
- 2 => determine the output routing. ex: eth0 -> gateway1, eth1 -> gateway2



- (路徑 **A**) 封包進入 Linux 主機使用資源：
  - 在路由判斷後確定是向 Linux 主機要求資料的封包，主要就會透過 filter 的 INPUT 鏈來進行控管
- (路徑 **B**) 封包經由 Linux 主機的轉遞，沒有使用主機資源，而是向後端主機流動
  - 在路由判斷之前進行封包表頭的修訂作業後，發現到封包主要是要透過防火牆而去後端，此時封包就會透過路徑 B 來跑動。也就是說，該封包的目標並非我們的 Linux 本機。(filter 的 FORWARD 以及 nat 的 POSTROUTING, PREROUTING)
- (路徑 **C**) 封包由 Linux 本機發送出去
  - 例如回應用戶端的要求
    - 先是透過路由判斷，決定了輸出的路徑後，再透過 filter 的 OUTPUT 鏈來傳送的！當然，最終還是會經過 nat 的 POSTROUTING 鏈



若針對單機來說，INPUT 與 FORWARD 算是比較重要的管制防火牆鏈

列出完整的防火牆規則

- iptables-save

```
[root@www ~]# iptables-save [-t table]
選項與參數：
-t ：可以僅針對某些表格來輸出，例如僅針對 nat 或 filter 等等

[root@www ~]# iptables-save
# Generated by iptables-save v1.4.7 on Fri Jul 22 15:51:52 2011
*filter                         <==星號開頭的指的是表格，這裡為 filter
:INPUT ACCEPT [0:0]             <==冒號開頭的指的是鏈，三條內建的鏈
:FORWARD ACCEPT [0:0]           <==三條內建鏈的政策都是 ACCEPT 囉！
:OUTPUT ACCEPT [680:100461]
-A INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT <==針對 INPUT 的規則
-A INPUT -p icmp -j ACCEPT
-A INPUT -i lo -j ACCEPT   <==這條很重要！針對本機內部介面開放！
-A INPUT -p tcp -m state --state NEW -m tcp --dport 22 -j ACCEPT
-A INPUT -j REJECT --reject-with icmp-host-prohibited
-A FORWARD -j REJECT --reject-with icmp-host-prohibited <==針對 FORWARD 的規則
COMMIT
# Completed on Fri Jul 22 15:51:52 2011
```

清除規則

- 一般來說，我們在重新定義防火牆的時候，都會先將規則給他清除掉

選項與參數：
-F ：清除所有的已訂定的規則；(?)

-X：殺掉所有使用者 "自訂" 的 chain (應該說的是 tables)囉；
-Z：將所有的 chain 的計數與流量統計都歸零

範例：清除本機防火牆 (filter) 的所有規則
[root@www ~]# iptables -F ;(?)
[root@www ~]# iptables -X
[root@www ~]# iptables -Z

定義預設政策 **(policy)**

- 當你的封包不在你設定的規則之內時，則該封包的通過與否，是以 Policy 的設定為準

Example

No package respond any request.

範例：將本機的 INPUT 設定為 DROP，其他設定為 ACCEPT
[root@www ~]# iptables -P INPUT DROP
[root@www ~]# iptables -P OUTPUT ACCEPT
[root@www ~]# iptables -P FORWARD ACCEPT
[root@www ~]# iptables-save

不論封包來自何處或去到哪裡，只要是來自 lo 這個介面，就予以接受

```
iptables -A INPUT -i lo -j ACCEPT
```

假如你的主機有兩張乙太網路卡，其中一張是對內部的網域，假設該網卡的代號為 eth1 好了，如果內部網域是可信任的，那麼該網卡的進出封包就通通會被接受

```
iptables -A INPUT -i eth1 -j ACCEPT
```

只要是來自內網的 (192.168.100.0/24) 的封包通通接受

```
iptables -A INPUT -i eth1 -s 192.168.100.0/24 -j ACCEPT
```

只要是來自 192.168.100.10 就接受，但 192.168.100.230 這個惡意來源就丟棄

```
[root@www ~]# iptables -A INPUT -i eth1 -s 192.168.100.10 -j ACCEPT
[root@www ~]# iptables -A INPUT -i eth1 -s 192.168.100.230 -j DROP
```

想要連線進入本機 port 21 的封包都抵擋掉

```
[root@www ~]# iptables -A INPUT -i eth0 -p tcp --dport 21 -j DROP
```

想連到我這部主機的網芳 (upd port 137,138 tcp port 139,445) 就放行

```
[root@www ~]# iptables -A INPUT -i eth0 -p udp --dport 137:138 -j ACCEPT
[root@www ~]# iptables -A INPUT -i eth0 -p tcp --dport 139 -j ACCEPT
[root@www ~]# iptables -A INPUT -i eth0 -p tcp --dport 445 -j ACCEPT
```

只要來自 192.168.1.0/24 的 1024:65535 埠口的封包，且想要連線到本機的 ssh port 就予以抵擋

```
[root@www ~]# iptables -A INPUT -i eth0 -p tcp -s 192.168.1.0/24 \
> --sport 1024:65534 --dport ssh -j DROP
```

不需要針對回應的封包來撰寫個別的防火牆規則

- 這個想要進入的封包是否為剛剛我發出去的回應？

只要已建立或相關封包就予以通過，只要是不合法封包就丟棄

```
[root@www ~]# iptables -A INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT
[root@www ~]# iptables -A INPUT -m state --state INVALID -j DROP
```

ICMP 封包規則的比對：針對是否回應 **ping** 來設計

不會接受 ping 的回應

```
// iptables -A INPUT [-p icmp] [--icmp-type 類型] -j ACCEPT
iptables -A INPUT -p icmp --icmp-type 8 -j DROP
```

Reference

- 防火牆與 NAT 伺服器, http://linux.vbird.org/linux_server/0250simple_firewall.php

Further Reading

- How the iptables Firewall work, https://www.digitalocean.com/community/tutorials/how-the-iptables-firewall-works