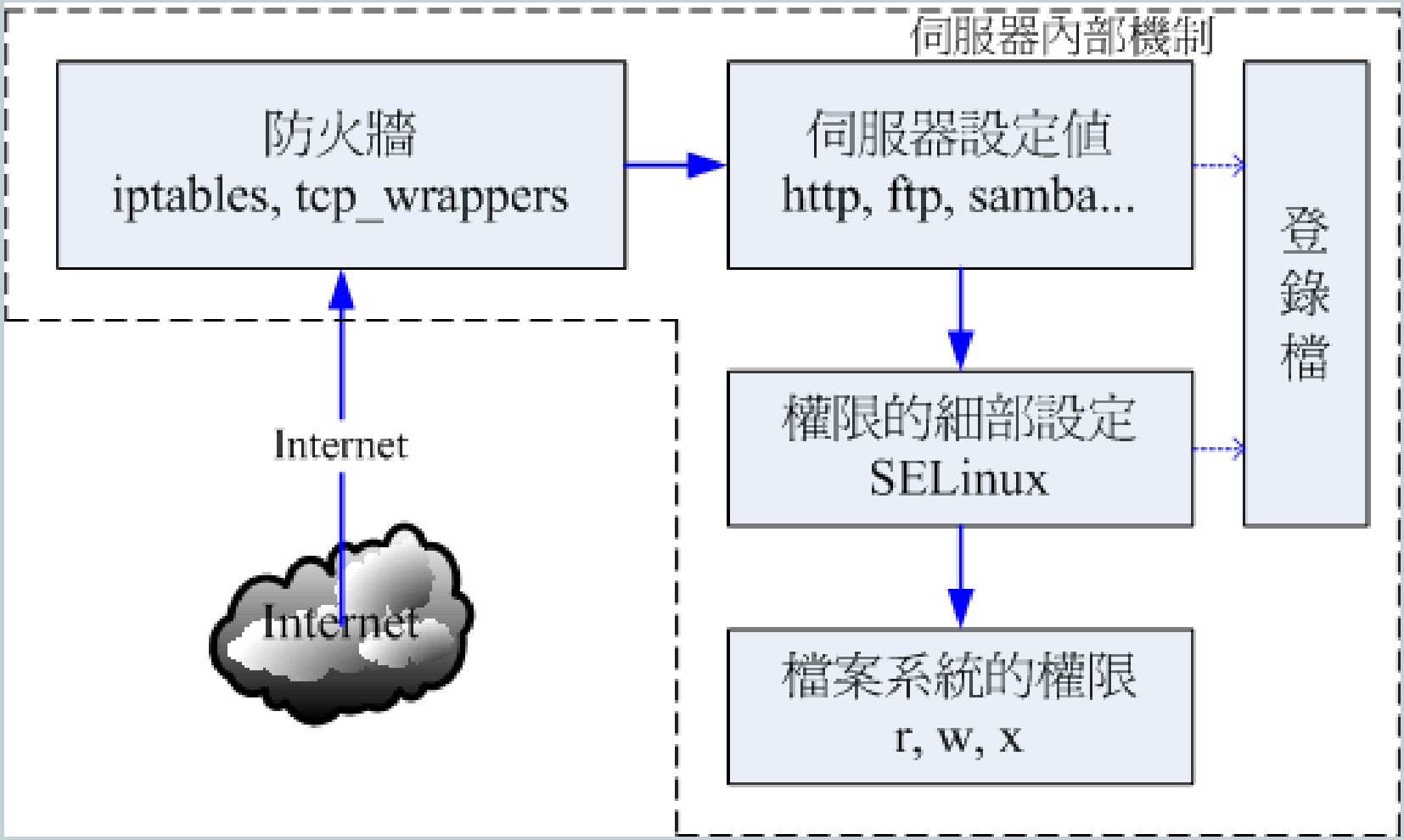
Linux 網路與服務管理

鳥哥

2015/11/17

回顧：如何取得Server的資料？



回顧：如何取得Server的資料？

1. 要先有正確的網路才能連線到Server
2. 要通過Server的防火牆規則
3. 要Server有提供網路服務才行
4. Server要有開放SELinux的相關規定：
   a. 有啟動SELinux與否(SELinux模式)
   b. 需放行SELinux的規則(rule)
   c. 正確的SELinux安全文本(Type)
5. 正確的目錄/檔案權限

案例一：CentOS 7的網路管理

- 傳統網路管理的方式：
  - 傳統上使用的是編輯 /etc/sysconfig/network-scripts/ 檔案
  - 再透過 service network restart 來處理
  - 最好要關閉 NetworkManager 這個自動網路服務

- CentOS 7 預設與建議的方式：啟用 NetworkManager
  - 啟用 NetworkManager 服務
  - 使用 nmcli 這個指令(已提供 [tab] 參數補齊功能) 設定/觀察

案例一：以 nmcli 觀察網路連線

- 觀察的方式：
  - nmcli connection show
  - nmcli connection show etho
- 刪除連線的方式：
  - nmcli connection delete xxx
- 增加連線的方式：
  - nmcli connection add \
    - con-name etho \
    - if-name etho \
    - type ethernet...

案例一：以 nmcli 設定與啟動網路連線

- 修改連線的方式：
  - nmcli connection modify eth0 \
    - ipv4.method [auto|manual] \
    - ipv4.address “IP/netmask, IP2/netmask...” \
    - ipv4.gateway “GatewayIP” \
    - ipv4.dns “DNSIP,DNSIP2,...” \
    - connection.autoconnection [yes|no] ....
- 啟動這個修改過的連線
  - nmcli connection up eth0
  - nmcli connection show eth0
    - 注意最後幾頁大寫的項目，看對否！

案例一：判斷網路連線正確否？

- 底下的動作來判斷看看網路正常否？
  - ip addr show etho　　看看網路設定是否正確
  - route -n　　看看 gateway 設定是否正確
  - ping –c 3 GWIP　　看看我們與 gateway 連線是否正確
  - dig www.google.com　　看看 DNS 是否有動作
  - ping 168.95.1.1　　額外的作法，連到中華電信(不一定成功)

動手實做案例一：實際使用 nmcli

- 由於我們系統的網路是虛擬來的，因此會有使用上的問題，請將該虛擬界面刪除，再重建一個新的網路界面，同時網路參數改為：
  - IP/Netmask == 172.20.101.*/16 (*為你的號碼)
  - Gateway == 172.20.0.254
  - DNS == 168.95.1.1, 172.16.200.254

案例二：主機名稱的設定

- 網路伺服器應該都有一個主機名稱
- 觀察的方式：
  - hostname
  - hostnamectl
- 設定的方式：
  - hostnamectl set-hostname YOURNAME
  - /etc/hostname
  - 設定即生效，但此次的 bash shell 可能需要登出再登入

案例二：主機名稱要有 IP 嗎？

- 一般來說，伺服器主機名稱要有 IP 對應才行！
  - Q：為何會有 localhost 這個主機名稱？
  - ex> ping -c 3 localhost
- 主機名稱沒有 IP 會怎樣？
  - 最大的問題：某些服務會 IP 反查，導致連線等待時間過長
  - Ex> ssh, ftp 等等
  - 連線等待時間長，但是連線成功後，速度就又正常了。
  - 最常發生在內部 private IP 的問題上

案例二：主機名稱的 IP 對應方法

- 常見的主機名稱← →IP對應方法
  - DNS 設定
    - 針對 Internet 的伺服器來設定
    - 優點是伺服器設定好，全世界都生效
    - 缺點是，額外要架設 DNS server 與設定
  - /etc/hosts 直接寫入
    - 針對內部不公開連網的 private IP 設定
    - 優點是設定簡單
    - 缺點是，需要每部 private IP 都設定

動作實做案例二：主機名稱的設定

- 我們的主機名稱與 IP 對應這樣做：
  - 主機名稱設定為： station*.ncku
  - 主機名稱對應：
    - station1.ncku → 172.20.101.1
    - station2.ncku → 172.20.101.2
    - …
    - station100.ncku → 172.20.101.100
      - (使用 shell script 來處理較快速)

案例三：系統的網路服務有哪些？

- 網路服務？
  - 某個服務啟動後，會啟動一個(含)以上的埠口 (port)
  - 這些埠口如果是不正常的就會被稱作後門
  - 這些服務其實就是某些程式的執行
  - 所以如何關閉？
    - 就關掉這些程式/服務，即可關閉這些埠口

案例三：系統的網路服務有哪些？

- 觀察網路服務的簡單方式：
  - netstat -tlunp
    - 可以查詢 TCP UDP 的服務
    - 同時觀察啟動該埠口的指令名稱
- 列出目前系統管理的服務
  - systemctl list-units --type=service [--all]
  - systemctl list-unit-files [--all]

案例三：啟動/關閉系統管理的服務

- 啟動服務(底下均以 cups 服務為例)：
  - systemctl start cups
  - netstat -tlunp | grep cups
- 觀察服務狀態：
  - systemctl status cups
- 關閉服務：
  - systemctl stop cups
- 下次開機是否啟動這個服務？
  - systemctl [enable|disable] cups

實做案例三：關閉不要的系統網路服務

- 關閉不必要的網路服務，這樣對你的系統當然就有一定的保護功能！但是不能關掉有用的服務！預設來說，你的系統應該要有 port 22, 25，若有 dhcp 則會有 dhclient 這個軟體產生的埠口。除此之外：
  - 保留 port 22, 25
  - 關閉其他不要的埠口
  - 且這些被關閉的埠口，下次開機也不會被啟動！

案例四：系統的軟體來源

- 原版軟體在哪裡？
  - CentOS 除了官網之外，其軟體提供 mirror 服務
  - 除非你知道最近的 mirror 站，否則無須重新變更
- 但是我的伺服器有超過 10 台以上，如何節省頻寬？
  - 找到最近的那部 mirror 站
  - 更改設定檔，讓你的伺服器到那部 mirror 更新

案例四：如何在校內或區網找到mirror

- 雲端機器在崑大，那該如何搜尋？
  - 崑大的 FTP ： http://ftp.ksu.edu.tw
  - 找到 CentOS 7 的最新的那個連結
  - 找到有 repodata/ 那個目錄的 URL 就對了！

案例四：如何修改設定？

- 主要設定檔位置：
  - /etc/yum.repos.d/*.repo
  - /etc/yum.repos.d/CentOS-Base.repo
- 設定檔的基本內容：
  - [base]
  - name = xxx
  - baseurl = …
  - gpgcheck = 1
  - gpgkey = xxx

案例四：如何使用？

- 主要實做軟體：
  - yum update
  - yum update somepackage
  - yum search packagename
  - yum install somepackage
  - yum remove somepackage
- 若需要定期處理，可以使用 crontab 來處置！
  - vim /etc/crontab
    - [分] [時] [日] [月] [周] [身份] [指令]
    - 0 2 * * * root xxxx

案例四：軟體為何要升級？

- 系統的軟體升級是很重要的一件事喔：
  - 軟體有 bug / security 的問題時，官網會立刻更新
  - 但是你的伺服器軟體系統也需要同步更新才行！

- 軟體更新完畢後要不要重新開機？
  - 一般軟體：不用
  - 核心或重要函式庫軟體：要重新開機！

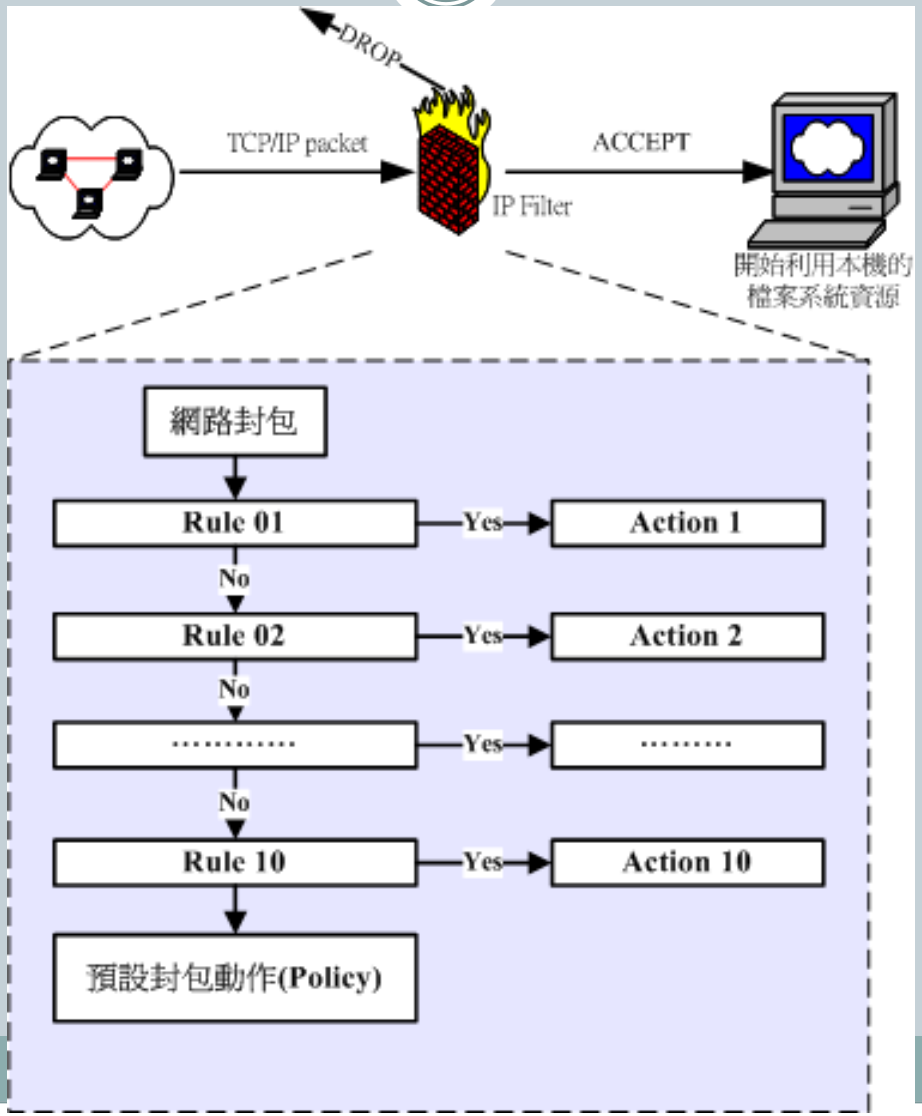
實做案例四：使用 yum 升級

- 透過剛剛談到的技巧，使用 vim 修改設定檔，同時透過 yum update 來升級全系統。另外，指定每天 3 點時，系統自動升級，且升級過程全部傳送到 /tmp/update.log

案例五：防火牆

- 什麼是防火牆？
  - 主要分為封包過濾與軟體功能來實施
  - 封包過濾：
    - 過濾網路封包，IP/TCP/UDP等表頭資料都會被解析
    - 通常不用來過濾封包內的資料內容，主要針對表頭資料
    - 主要針對 OSI layer2, 3, 4 層來管理
  - 軟體功能：
    - 例如 proxy 等，透過某些特殊軟體來進行控制
    - 因為是軟體控制，效能方面可能較不理想
    - 是否能夠處理某些任務，與軟體的設計有關
  - 整體來說→透過一條條規則比對來放行/抵擋網路連線

案例五：防火牆

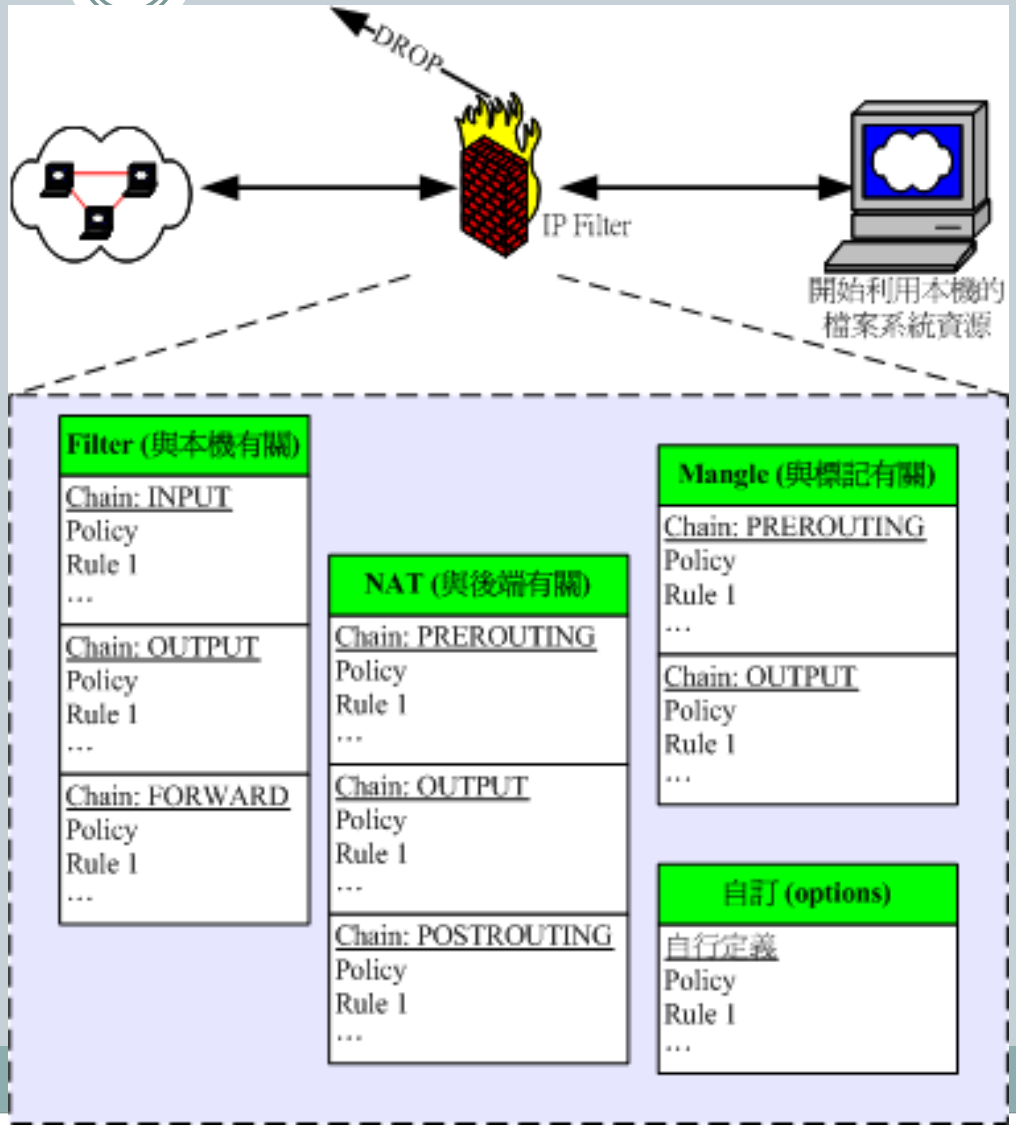

DROP
TCP/IP packet
IP Filter
ACCEPT
開始利用本機的
檔案系統資源
網路封包
Rule 01
Yes
Action 1
No
Rule 02
Yes
Action 2
No
.....
Yes
.....
No
Rule 10
Yes
Action 10
預設封包動作(Policy)

案例五：認識Linux防火牆

- Linux防火牆：
  - 嵌入在核心中，效能強，封包過濾形式
  - 名稱為： netfilter
  - 分為
    - Table
      - Chain
        - Rules
  - 可以類比：
    - Excel 檔案
      - 檔案內的工作表
        - 每個欄位寫的規則資料 (check list)

案例五：認識Linux防火牆

- Filter
  - INPUT
  - OUTPUT
  - FORWARD
- 一般設定
  - INPUT管理
  - OUTPUT 放行
  - FORWARD暫不理



案例五：Linux防火牆機制

- CentOS 7 的防火牆機制：
  - firewalld
    - 新型態的防火牆機制，分成多個 domain 來處理
    - 適合彈性規劃不同的 domain 的用途
    - 為目前預設的防火牆機制
  - iptables.service
    - 傳統的機制
    - 分類簡單，適合學習與了解
    - 與 firewalld 相牴觸，故須關閉 firewalld 才可以啟動 iptables

實做案例五：啟動 iptables 防火牆機制

- 為了方便學習，我們現在需要關閉 firewalld 並且啟動 iptables 的服務，該如何處理？
  - 關閉 firewalld
  - 安裝 iptables-services
  - 啟動 iptables.service
  - 未來都會啟動 iptables.service

案例六：變更防火牆規則

- 觀察防火牆規則的方式： iptables-save
  - 每條鏈的觀察重點：
    - 三條鏈的政策
  - OUTPUT 鏈的規則重點：
    - 除非必要，否則 OUTPUT 應該是完全放行的
  - INPUT 鏈的規則重點：
    - 內部 lo 一定要放行
    - 自我封包 (ESTABLISHED,RELATED) 一定要放行
    - 針對區網、信任服務等相對放行
    - 最終最好再全部拒絕一次！

案例六：變更防火牆規則

- 基礎防火牆規則：
  - **iptables [-A 鏈名] [-i 網路介面] **
    - **[-s 來源IP/網域] [-d 目標IP/網域] **
    - **-j [ACCEPT|DROP|REJECT|LOG]**
  - **iptables [-A 鏈] [-i 網路介面] [-p tcp,udp] **
    - **[-s 來源IP/網域] [--sport 埠口範圍] **
    - **[-d 目標IP/網域] [--dport 埠口範圍] **
    - **-j [ACCEPT|DROP|REJECT]**

案例六：變更防火牆規則

- 訂定防火牆規則的流程：
  - 訂定預設政策：
    - INPUT 為 DROP 其他 ACCEPT
  - 針對 INPUT 的規則流程：
    - 放行基礎防火牆 (lo, ESTABLISHED, icmp)
    - 放行信任用戶 (例如區網/自訂的後門來源)
    - 放行信任服務
    - 其他考量
    - 全部拒絕
  - 最終一定要儲存規則才行喔！

實做案例六：變更防火牆規則

- 以既有的防火牆規則為基礎，透過建立 /root/firewall.sh 腳本，來建置屬於你自己的簡單防火牆規則！
  - 記得要放行 port 21, 80, 443,
  - port 22 不要隨便放行，請針對來源放行即可

案例七：WWW簡易管理

- 服務的口訣：
  - 安裝
  - 啟動
  - 開機啟動
  - 防火牆
  - 測試
  - 上傳資料

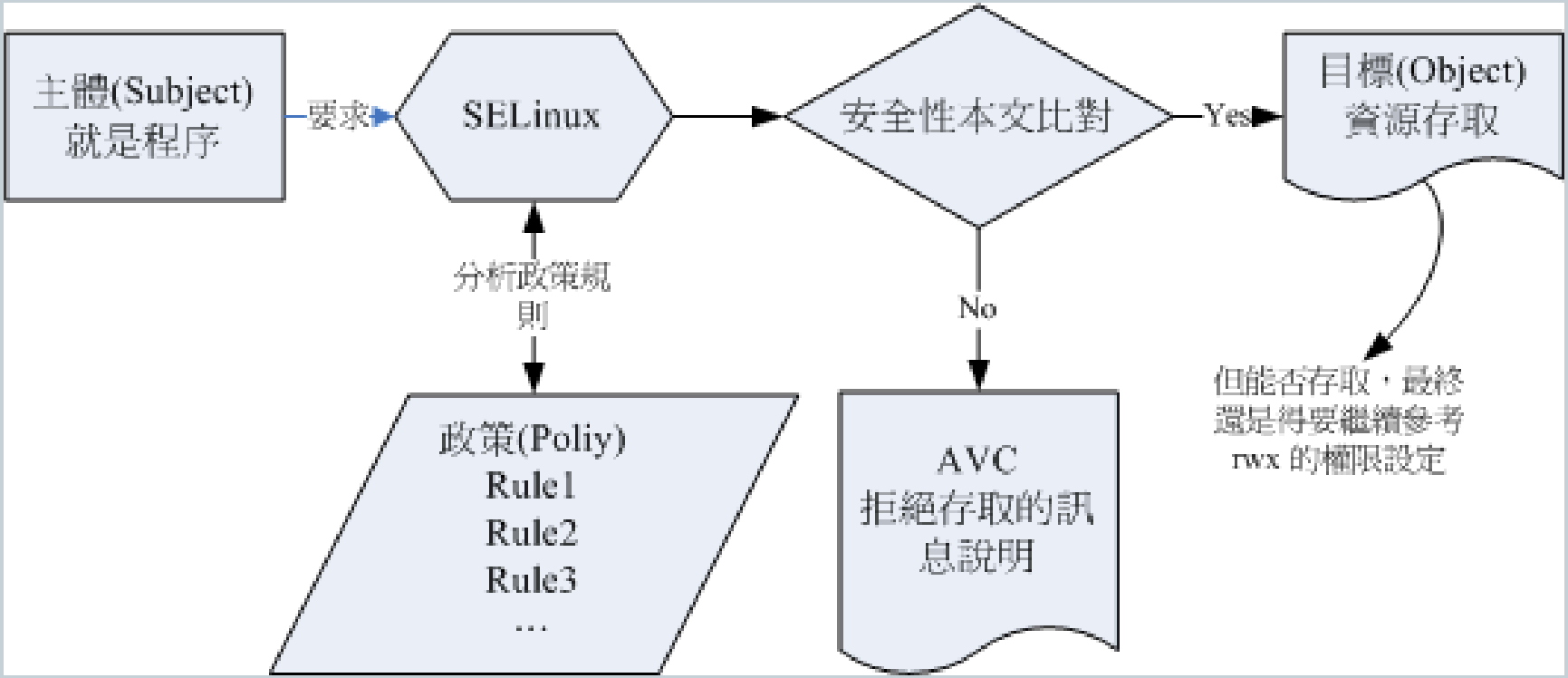
實做案例七：建立WWW服務

- 在Linux上面的WWW服務名稱為httpd，該如何啟動並且測試呢？

案例八：SELinux管理

- WWW服務的延續：
  - 在 /root 底下建立 index.html，內容請填寫一些基本資料
  - 將 /root/index.html 『移動』到 /var/www/html 去
  - 使用 http://127.0.0.1/index.html 查閱出了什麼事？
  - 設定 setenforce 0
  - 再次重複 http://127.0.0.1/index.html 是否能看到資料？
  - 為什麼會這樣？

案例八：SELinux 的運作流程



案例八：SELinux的模式

- SELinux 的模式：
  - Enforcing
  - Permissive
  - Disabled
- 模式的觀察與設定：
  - getenforce
  - setenforce [0|1]
  - /etc/selinux/config

案例八：SELinux 的規則

- SELinux 的規則
  - 觀察　　getsebool -a
  - 設定　　setsebool -P rulename [0|1]

案例八：SELinux 的安全本文

- 安全本文的觀察
  - ls -Z [filename]
- 安全本文的修改
  - chcon -t type_t filename
  - restorecon [-Rv] filename

案例八：SELinux出錯的處理方式

- 若 setenforce 0 再次查閱網路服務，若服務就正確了，那表示一定是由 SELinux 引起的亂子
  - 查閱 /var/log/messages
  - 根據該檔案的內容來處理相關的動作即可
  - 最終請務必 setenforce 1 調整回 Enforcing 狀態！

實做案例八：克服服務錯誤

- 我們知道 index.html 確實是 SELinux 引起的錯誤了，該如何處理這項錯誤呢？敬請解決他吧！