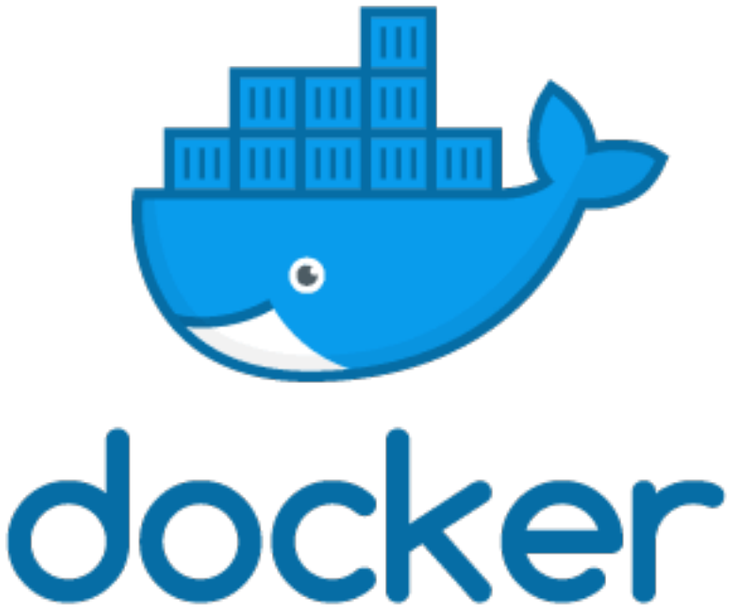


輕量化虛擬技術的探討與應用

2017/12/06

洪稟凱
Jed Hung

“從入門到放棄”

–蛤？ 調查時間～

“隨時打斷我，隨時提問”

–對 就是這樣 XD 來個互動

“關於我”

-目前我有兩個身份

Jed Hung

台灣科技大學資工所

Sentra Smart Technology Inc.

學生 & 顧問

都在做什麼呢～

GitHub

syneart.com

“虛擬機器 (Virtual Machines；VMs)”

–是什麼？

在電腦科學中的體系結構裏，是指一種特殊的軟體，可以在電腦平台和終端用戶之間建立一種環境，而終端用戶則是基於這個軟體所建立的環境來操作軟體 **[wikipedia]**

“雲端運算平台”

– cloud computing 是什麼？

是一種基於網際網路的運算方式，通過這種方式，共享的軟硬體資源和資訊可以按需求提供給電腦各種終端和其他裝置。[wikipedia]

“雲端運算平台架構”

通常分為三種

雲端運算平台 主要分為三種架構

- **SaaS** 軟體即服務
  (Software as a Service)

- **PaaS** 平台即服務
  (Platform as a Service)

- **IaaS** 基礎設施即服務
  (Infrastructure as a Service)

SaaS 軟體即服務 (Software as a Service)

–虛擬桌面, 遊戲等 (Web Application)

PaaS 平台即服務 (Platform as a Service)

–運行時環境, 資料庫, Web伺服器, 開發工具等

EX: Google App Engine (GAE), App Inventor

IaaS 基礎設施即服務 (Infrastructure as a Service)

– 虛擬機, 儲存, 負載平衡, 網路等

EX: Amazon AWS, Google GCP

“虛擬機器歷史”

–為什麼需要虛擬機器？跟 雲端運算 有什麼關係？

虛擬機器 主要分為三種型態

- 全虛擬化
  (Full virtualization)

- 半虛擬化
  (Para virtualization)

- 作業系統層虛擬化
  (Operating system–level virtualization)

全虚擬化

(Full virtualization)

FV

“Hypervisor”

虛擬機器監視器

（virtual machine monitor；VMM）

用來建立與執行虛擬機器的部份電腦軟體、韌體或硬體

“VMware”

–主要透過 Binary translation 的方式
去解決虛擬和真實的硬體彼此溝通的橋樑

半虚擬化

(Para virtualization)

PV

“Xen”

是一個開放原始碼虛擬機監視器，由XenProject開發

“KVM”

is a full virtualization solution for Linux on x86 hardware containing virtualization extensions (Intel VT or AMD-V)

修改核心但還是 **FV**

作業系統層虛擬化 (Operating system–level virtualization)

Container

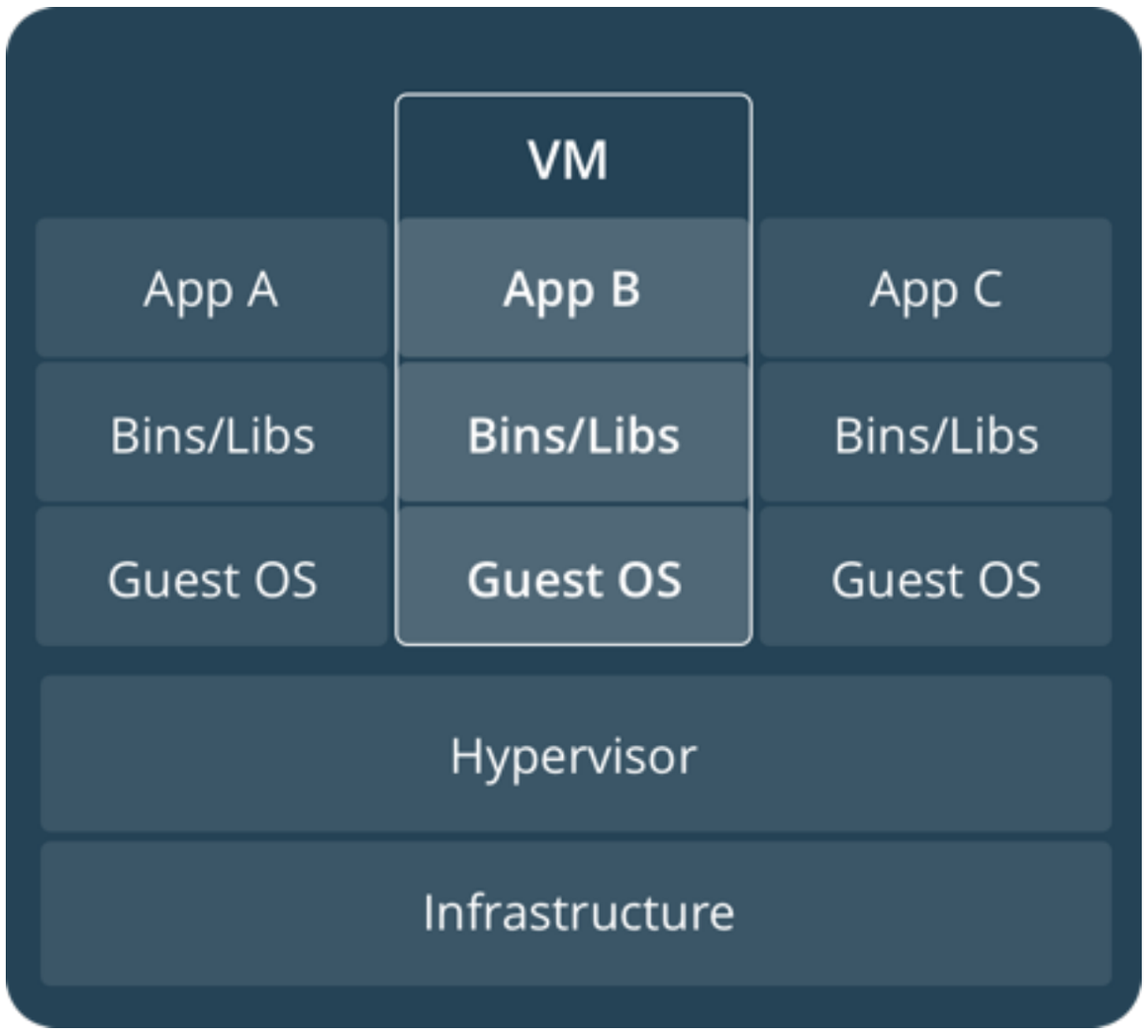
LXC
(Linux Containers)

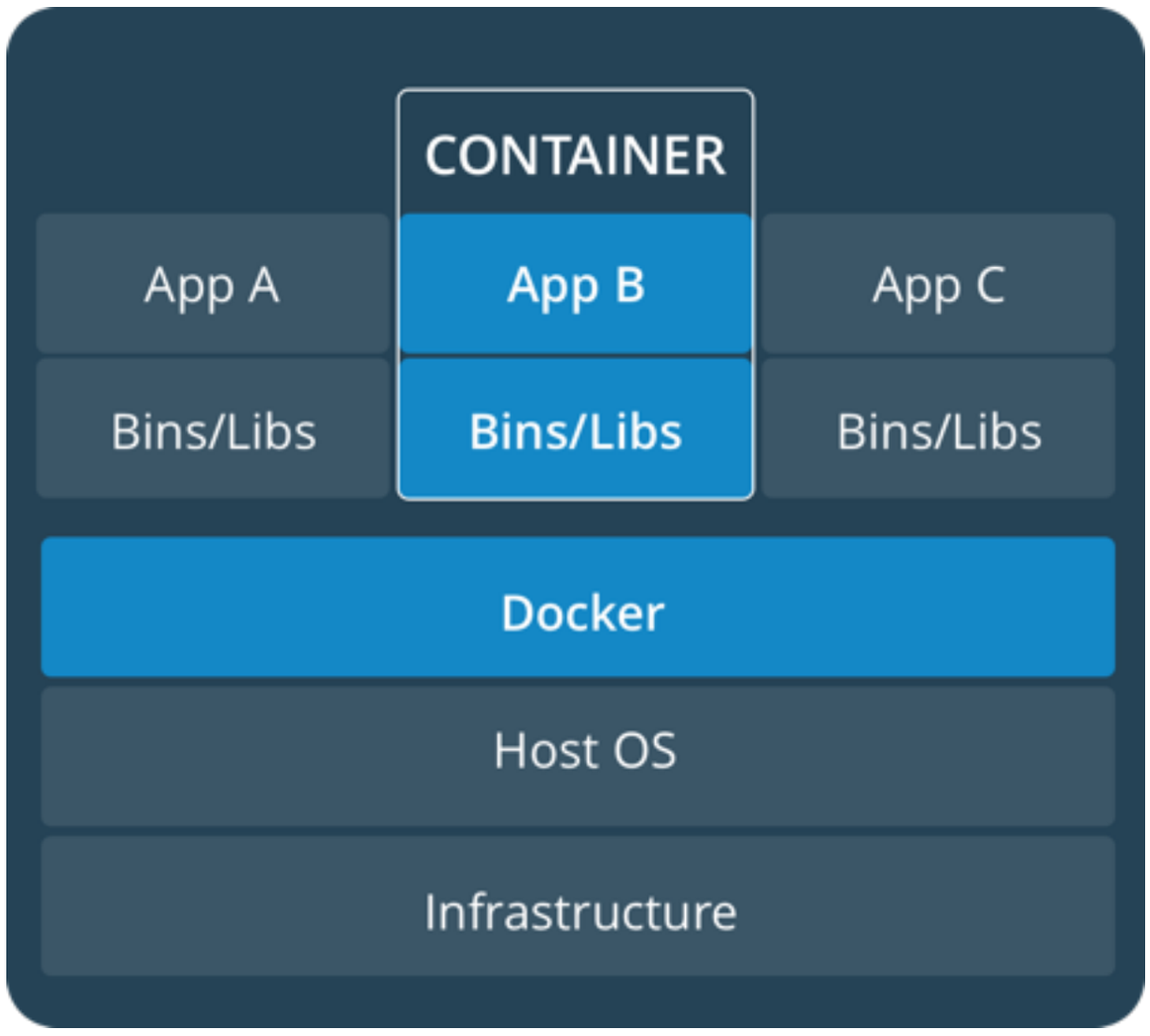
作業系統層虛擬化

（Operating system–level virtualization）

Docker

Container (容器) 速度好快又輕量？





Virtual Machines and Containers

source: https://www.docker.com/what-container

“DevOps”

DevOps（Development和Operations的組合詞）是一種重視「軟件開發人員（Dev）」和「IT運維技術人員（Ops）」之間溝通合作的文化、運動或慣例。[wikipedia]

“Linux”

是一種自由和開放原始碼的類UNIX作業系統
[wikipedia]

- 檔案屬性：
  Linux 的檔案有很多的屬性，包括是否可讀、可寫、可執行，
  以及檔案所屬人、所屬群組，
  每個檔案所建立、修改過的時間等等，都算是檔案的屬性。

- 檔案內容：
  例如文字檔內的文字，
  資料庫檔案內的資料，這些就是所謂的檔案內容。

檔案結構 (1/2)

source: http://linux.vbird.org/linux_desktop/0110linuxbasic.php#linux

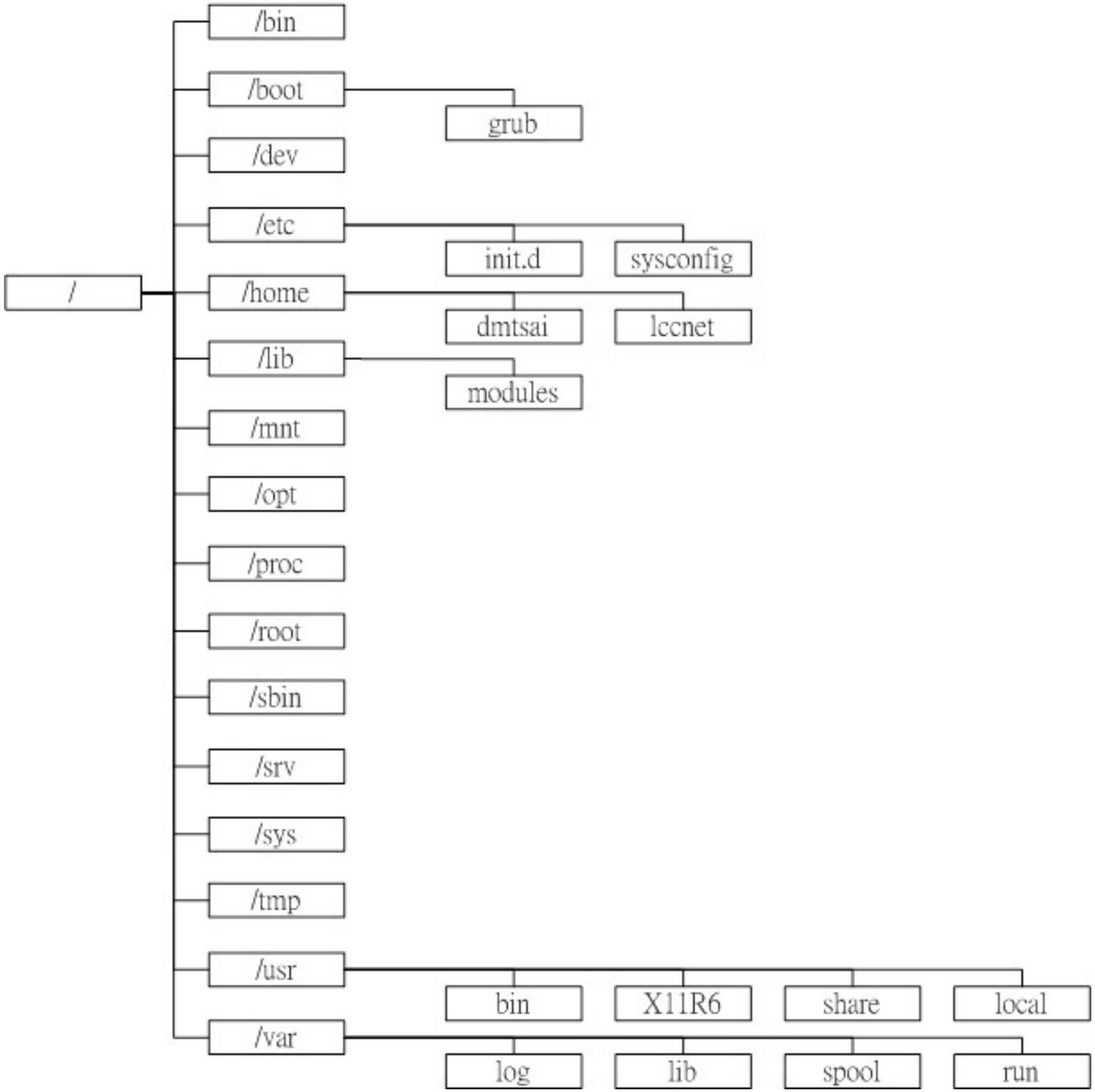
- inode：inode 的區塊主要在儲存檔案的屬性，每個 inode 有 128bytes 這麼大， 而且其中除了屬性之外，還會記錄該檔案的內容所在的 block 區塊位置。
  請注意，每個 inode 都是有號碼的
- block：就是檔案內容放置的地方，標準的 block 一個約 4Kbytes 這麼大，同樣的，每個 block 都有號碼喔！
  這個號碼可以被記錄到 inode 裡頭去， 這樣檔案才會知道他的內容被放置到那個 block 區塊

檔案結構 (2/2)

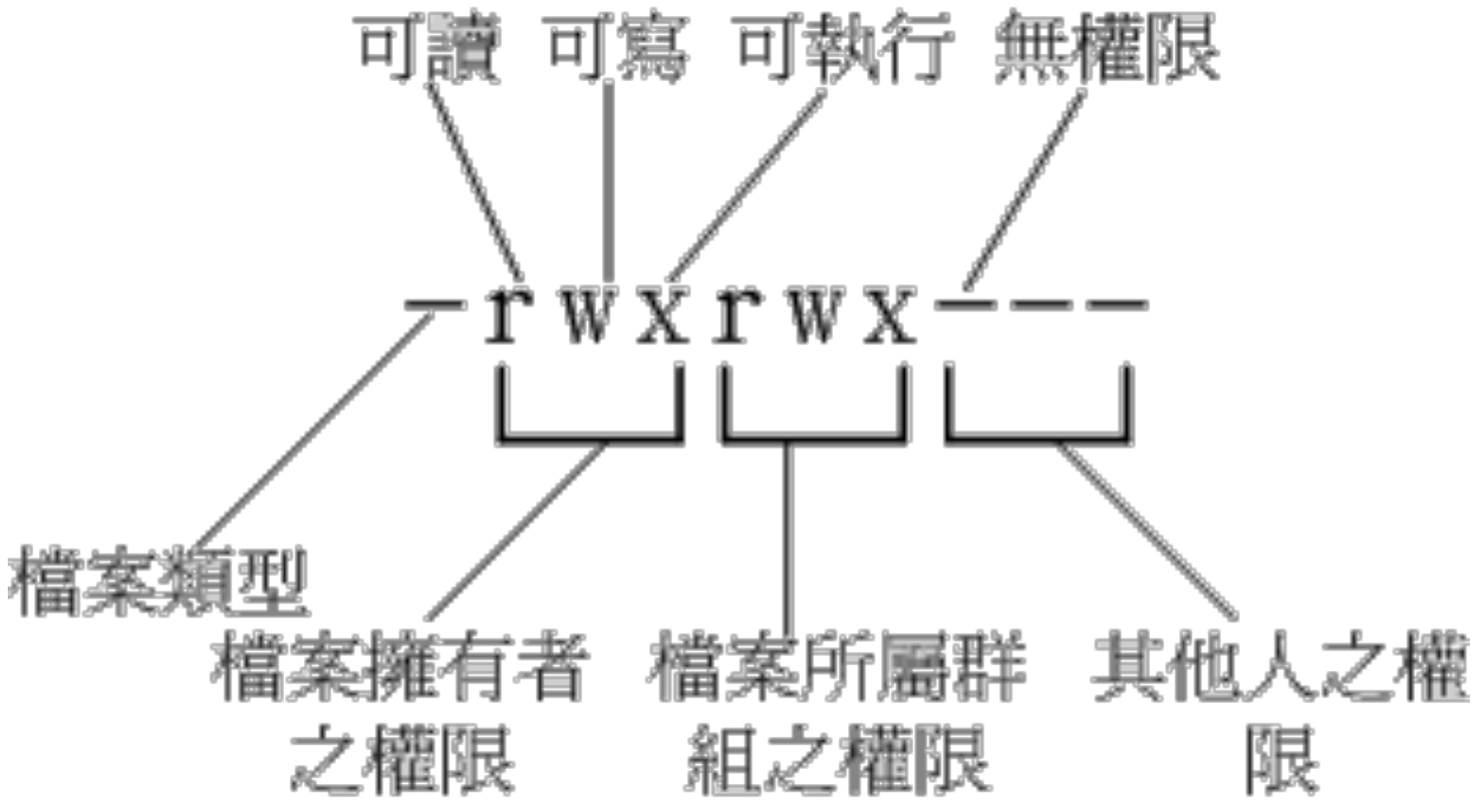
source: http://linux.vbird.org/linux_desktop/0110linuxbasic.php#linux

“Linux 目錄樹資料架構”

沒有所謂的裝置名稱 (C:, D: ..)



source: http://linux.vbird.org/linux_desktop/0110linuxbasic.php#linux



檔案權限

source: http://linux.vbird.org/linux_desktop/0110linuxbasic.php#linux

“Linux Kernel”

最早是由芬蘭駭客林納斯·托瓦茲為嘗試在自己的英特爾x86架構電腦上提供自由免費的類Unix系統而開發的，現代許多開放原始碼的作業系統都是採用Linux核心進行實作。 [wikipedia]

“Linux Distribution”

EX: RedHat, CentOS, Fedora, Debian, Ubuntu
(Kernel + 自己的 Tools)

Docker 怎麼辦到的

核心原理

“作業系統核心技術”

–為容器（Container）提供資源隔離與安全保障

“容器需要什麼？”

資源隔離, 資源限制

“namespace”

資源隔離 (網路, 程序, 使用者 ...)

“namespace 提供 6 種隔離”

namespace 提供六種隔離

- **UTS** - 主機名稱與域名
- **IPC** - 訊息佇列和用共記憶體
- **PID** - 處理程序的編號
- **Network** - 網路裝置, 網路堆疊, **Port**
- **Mount** - 檔案系統掛載
- **User** - 使用者及使用者群組

“namespace API”

clone(), setns(), unshare()

namespace API (1/3)

- 透過 **clone()** 在建立新處理程序的同時建立 **namespace**
- 可以檢視 **/pro/[pid]/ns**

namespace API (2/3)

- 透過 **setns()**

  加入一個已存在的 **namespace**

- **Docker exec**

namespace API (3/3)

- 透過 **unshare()**
  在原先處理程序上進行
  namespace 隔離

- **Docker** 沒用到 **XD**

“cgroup”

資源限制

(記憶體使用上限, 優先權分配, 資源統計, 工作控制)

“cgroup 提供 4 種作用”

cgroup 提供 4 種作用 (1/2)

- 資源限制：

  例如，限制執行時期記憶體使用上限

  (所以如果 **OOM** …)

- 優先順序分配：

  例如，分配 **CPU** 時間切片及磁碟 **IO**

  (工作執行優先順序)

cgroup 提供 4 種作用 (2/2)

- 資源統計：
  例如, **CPU** 時長, 記憶體用量
  (用於費率)

- 工作控制：
  工作執行暫停, 恢復等

“容器能力 (Capability)”

(Linux Kernel 2.2)

user namespace

Docker 儲存驅動

aufs, btrfs, device mapper, vfs, overlay

“AUFS”

(Another Union File System)

–Copy on Write (寫入時複製)

“Device mapper”

–利用 index

“Docker 映像檔 (Image)”

“Docker client”

接收指令

“Docker daemon”

處理指令

“Volume”

外部掛載

“Docker 網路管理”

分為 4 種 (Bridge, Host, Container, None)

“Docker daemon 通訊方式”

pipeline （network namespace）

“Docker 安全”

穿透

SELinux

(Security Enhanced Linux Basics)

在進行程序、檔案等細部權限設定依據的
一個核心模組

SELinux 三種模式

- **enforcing**：強制模式，代表 **SELinux** 運作中
- **permissive**：寬容模式：代表 **SELinux** 運作中，不過僅會有警告訊息並不會實際限制存取
- **disabled**：關閉，**SELinux** 並沒有實際運作

source: http://linux.vbird.org/linux_basic/0440processcontrol.php#selinux

“Docker Registry”

& 私有庫

使用類似 **git** 指令的方式

“Dockerfile”

https://philipzheng.gitbooks.io/docker_practice/content/dockerfile/basic_structure.html

“Running GUI apps with Docker”

http://fabiorehm.com/blog/2014/09/11/running-gui-apps-with-docker/

“Container with VNC”

https://github.com/fcwu/docker-ubuntu-vnc-desktop

“其他實際應用”

“Docker 入門與實踐”

https://philipzheng.gitbooks.io/docker_practice/content/introduction/what.html

“Docker 安装”

Windows (Boot2docker), MacOS (Boot2docker) , Linux

“Docker 安裝”

不是只能在 Linux 上用嗎？

Boot2docker

是一個專門在Mac及Windows下使用Docker的套件，包括了:

- 一個VirtualBox程式
- VirtualBox格式的極小Linux VM
- 位於該VM中的Docker程式
- Boot2Docker管理工具

source: http://linux.vbird.org/linux_desktop/0110linuxbasic.php#linux

“Q&A”

歡迎提問