

** section 1 **

Ctrl + Alt + F1 ~ F6 文字介面

Ctrl + Alt + F7 圖形介面

root 管理者提示符號 #

一般使用者提示符號 >

#yast <文字介面 控制中心>

#yast2 <圖形介面 控制中心>

#yast -l <列出 yast 的模組>

Filesystem Hierarchy Standard

<http://www.pathname.com/fhs/>

路徑描述

- 絕對路徑：從 根目錄 / 開始描述
- 相對路徑：從 現在目錄 開始描述
 - . 目前的目錄
 - .. 上一層目錄

** section 2 **

Human --(shell 使用者介面)--> Kernel ----> Hardware

	收發e-mail	登入系統	FTP
/bin/bash	V	V	V
/bin/false	V	X	利用PAM控管

Login shell 需要登入才能取得的shell

- /etc/profile
- /etc/bash.bashrc

Non-Login shell 需要登入才能取得的shell

- /etc/bash.bashrc

指令語法

指令 [-] [--] [TARGET]
 short option long option 對象
 參數 敘述

可以使用 **TAB** 按鍵 或是 **Ctrl + i** 來使用檔案名稱/命令補齊

變數名稱=值 (定義變數, **current shell**有效)

'' 一律視為字串

" 允許命令代換

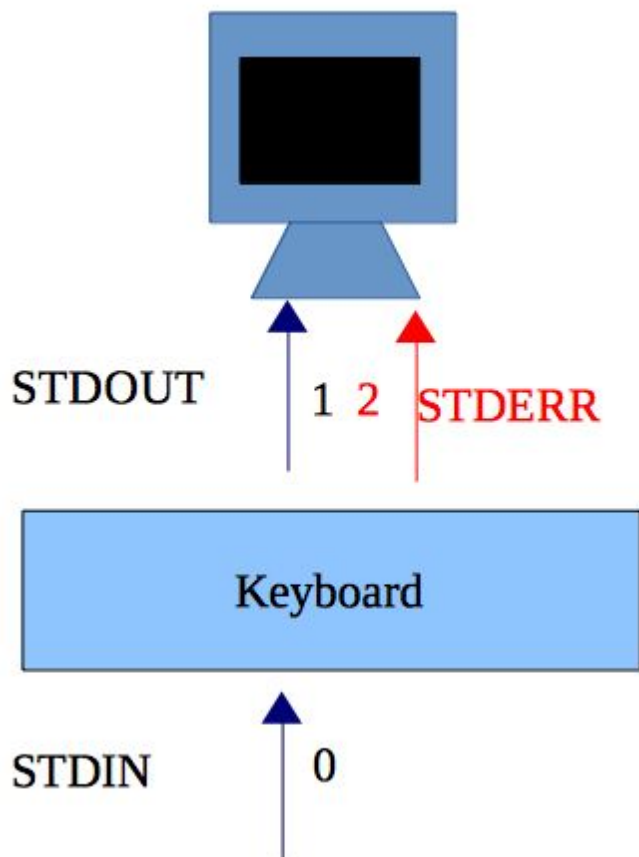
`` 先命令代換再輸出 也等同於**\$()**

history 列出指令歷程

- 預設 HISTSIZE=1000
- 存放在 ~/.bash_history
 - 以秒來進行紀錄並註解 date + %s
- 可以用 Ctrl + R 來搜尋

su 切換使用者 shell

- -l login shell



> 輸出導向(新的會覆蓋舊的)
>> 以附加方式輸出導向(新的會附加舊的後面)
< 輸入導向
\$? Return value (回覆值)

&& --> AND(兩者為一才為一), 當指令1成功時, 才執行指令2
|| --> OR (兩者有一就為一), 當指令1失敗時, 才執行指令2

指令1 && 指令2 當指令1執行成功的時候才去執行指令2

指令1 || 指令2 當指令1執行失敗的時候才去執行指令2

cd 切換工作目錄

- cd - 回到上次工作目錄
- cd ~ 切換到家目錄
- cd ~max 切換到使用者 max 的家目錄

pwd 顯示目前的工作目錄

在shell 內可以使用

- ctrl + c 中斷
- ctrl + l 清除螢幕

ls 列出目錄下的物件

touch 不存在的檔案名稱 <建立檔案>
touch 已存在的檔案名稱 <更新時間>

時間的部分, 有 mtime / atime / ctime 使用 stat 觀察

cat 列出檔案的內容
tac 以反序的方式列出檔案內容

more 以page by page 的方式列出檔案內容
less 以page by page 的方式列出檔案內容

df 列出系統的可用空間(以partition為基準)
du 系出磁碟使用量(以Directory 為基準)

head 列出檔案的前幾行

tail 列出檔案的後幾行

指令1 | 指令2

| pipe 會把指令1處理完的結果丟給指令2繼續處理

cp 複製 檔案/資料夾

mv 舊檔案名稱 新檔案名稱 <重新命名>

mv 檔案名稱 資料夾(目的地) <移動>

	消耗inode	原檔案刪除是否有效	是否可以跨Partition
Hard link	NO	有效	NO
Symbolic link	YES	無效	YES

tar – The GNU version of the tar archiving utility

- -c 建立
- -v 顯示資訊
- -f 檔案名稱
- -x 解開
- -z *.gz 格式
- -j *.bz2格式
- -C 指定目的地資料夾

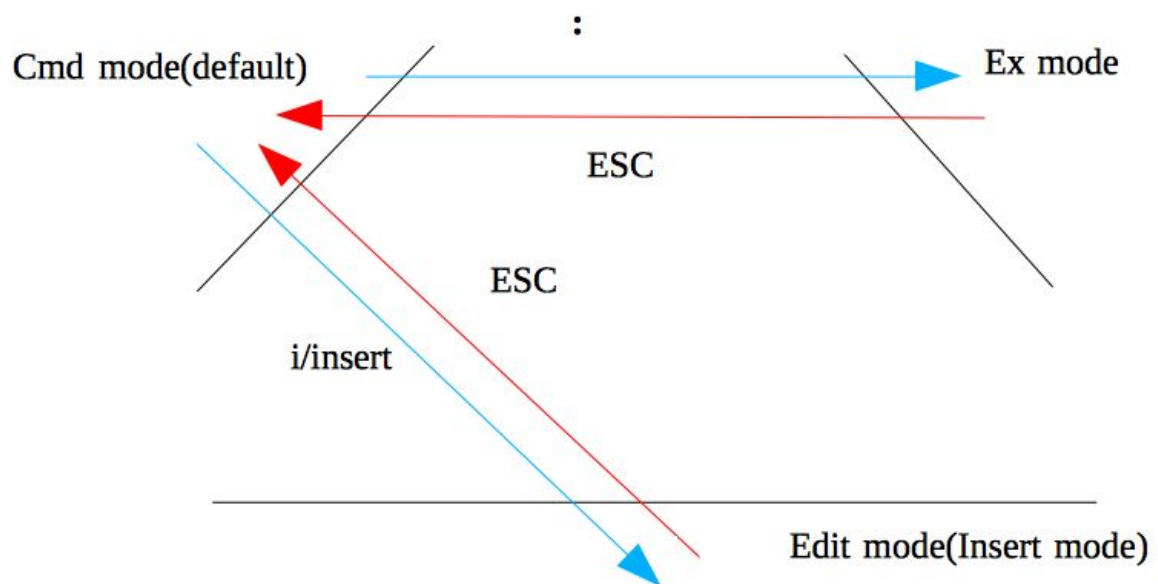
grep 過濾檔案的內容

- -i 忽略大小寫
- -w 符合整個字串
- -v 符合就不要

MAN section

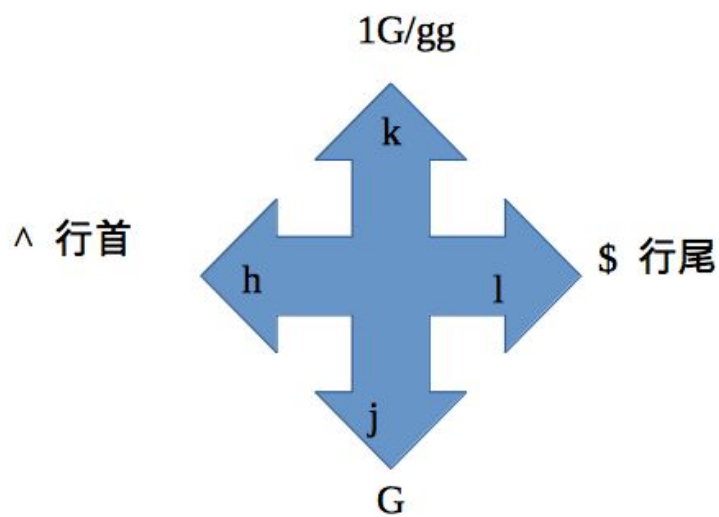
- 1 User commands 使用者指令
- 5 Configuration file and formats 設定檔及格式
- 8 System administrator commands 管理者指令

Vi mode



|

在 vi 內游標的移動



練習取代

`:%s/httpd.conf/httpd.bak/gc`

% --> global全部(搜尋範圍)

s --> 取代

httpd.conf --> 原字串

httpd.bak --> 新字串

g --> global全部(如果一行內有兩個以上符合,全部取代)
c --> 確認

ssh client 設定檔 /etc/ssh/ssh_config
sshd 設定檔 /etc/ssh/sshd_config

ssh 用戶端與主機的驗證
client 端存放於 ~/.ssh/known_hosts
與server端 /etc/ssh/ssh_host_rsa_key.pub

使用金鑰登入伺服器
*將用戶端的公鑰 複製到 伺服器上使用者的 ~/.ssh/authorized_keys
有關於第二把公鑰以後的附加

可以使用 ssh-copy-id 的指令來輕鬆完成
ssh-copy-id -i /home/user4/.ssh/id_dsa.pub(公鑰) 使用者@主機的IP

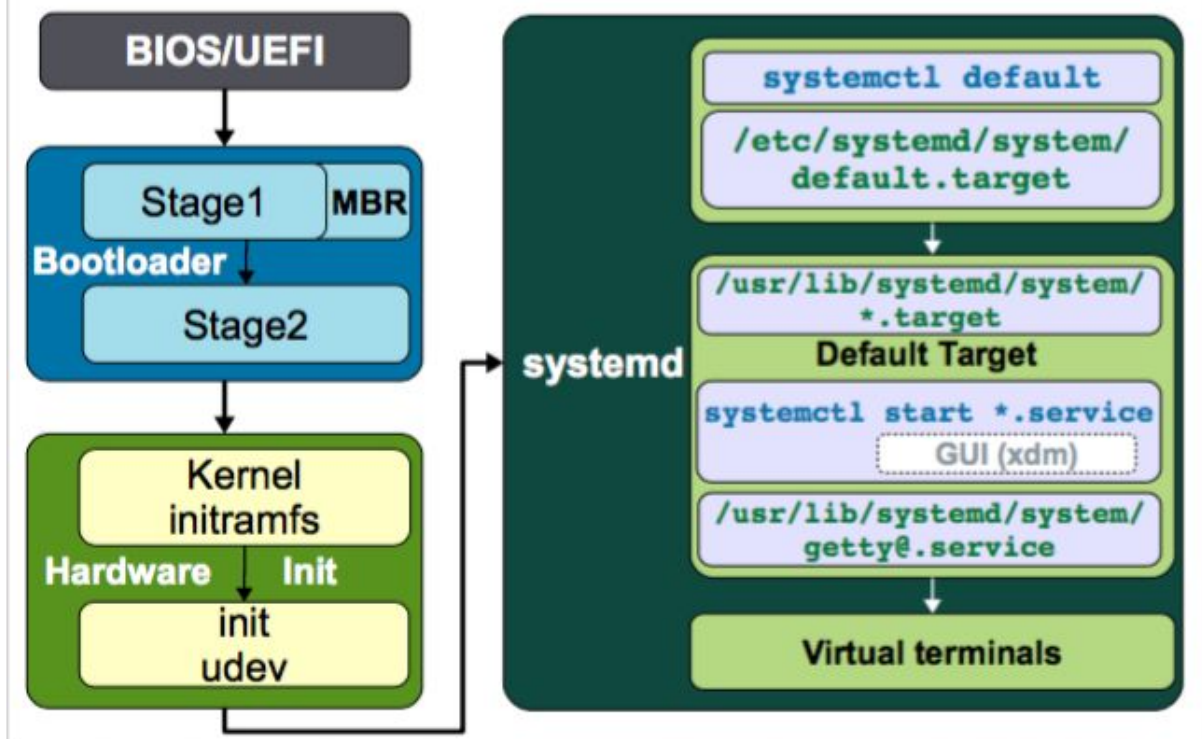
可以修改/etc/ssh/sshd_config 設定來指定
Protocol 2 <只使用 SSH2>
PermitRootlogin no <不允許root直接登入ssh>

用戶端可以藉由
#ssh -X 遠端IP <使用X forwarding>
#ssh -Y 遠端IP <使用X forwarding>

啟用遠端管理 可以使用
#yast2 remote
啟用 VNC, 連線方式
#vncviewer 主機IP:display
#vncviewer 主機IP:port
或是使用有 java 的瀏覽器 http://主機IP:5901

** section 3 **

Linux Boot Process



Initrd

- # ls /lib/modules/
- # lsinitrd /boot/initrd-3.12.49-11-default

UEFI

- <https://zh.wikipedia.org/wiki/%E7%B5%B1%E4%B8%80%E5%8F%AF%E5%BB%B6%E4%BC%B8%E9%9F%8C%E9%AB%94%E4%BB%8B%E9%9D%A2>
- <http://www.techbang.com/posts/4359>

GRUB2

- /boot/grub2 - 主要目錄
- /boot/grub2/grub.cfg - 主要設定檔, 不建議直接修改
 - http://linux.vbird.org/linux_basic/0510osloader.php#grub_menulist
- 主要修改 # /etc/default/grub 內常用設定
- 相關設定 /etc/grub.d/
 - 按順序執行

systemd

- 相關目錄
 - /etc/systemd/system
 - /usr/lib/systemd/system

■ 相關服務目錄

- /etc/systemd/system/default.target
 - 預設設定, 可以用 `ls -l /etc/systemd/system/default.target` 觀察
 - `lrwxrwxrwx 1 root root 40 11月 28 15:06 /etc/systemd/system/default.target -> /usr/lib/systemd/system/graphical.target`
- 觀察
 - `# systemctl list-dependencies graphical.target`
 - `# systemctl list-dependencies multi-user.target`
- 可參考
 - http://linux.vbird.org/linux_basic/0510osloader.php#startup_init

與原本 systemV runlevel 對照

`# ls -l /usr/lib/systemd/system/runlevel*.target`

```
lrwxrwxrwx 1 root root 15 11月 28 15:01 /usr/lib/systemd/system/runlevel0.target -> poweroff.target
lrwxrwxrwx 1 root root 13 11月 28 15:01 /usr/lib/systemd/system/runlevel1.target -> rescue.target
lrwxrwxrwx 1 root root 17 11月 28 15:01 /usr/lib/systemd/system/runlevel2.target -> multi-user.target
lrwxrwxrwx 1 root root 17 11月 28 15:01 /usr/lib/systemd/system/runlevel3.target -> multi-user.target
lrwxrwxrwx 1 root root 17 11月 28 15:01 /usr/lib/systemd/system/runlevel4.target -> multi-user.target
lrwxrwxrwx 1 root root 16 11月 28 15:01 /usr/lib/systemd/system/runlevel5.target -> graphical.target
lrwxrwxrwx 1 root root 13 11月 28 15:01 /usr/lib/systemd/system/runlevel6.target -> reboot.target
```

systemctl 指令 對象

觀察狀態

`#systemctl status 對象`

啟用服務

`#systemctl start 對象`

停用服務

`#systemctl stop 對象`

服務啟用並不代表開機會啟動

設定開機啟動服務

`#systemctl enable 對象`

設定開機停用服務

`#systemctl disable 對象`

檢查服務是否開機啟動

`#systemctl is-enabled 對象`

YaST 模組

`#yast2 services-manager`

切換 target

systemctl isolate multi-user

<切換到文字模式>

- #init 3

systemctl isolate graphical

<切換到圖形模式>

- #init 5

更改預設 target

systemctl set-default multi-user

systemctl set-default graphical

- 可觀察 # ls -l /etc/systemd/system/default.target

** section 4 **

Program: An executable file.

Process: A running program.

User Process: 由使用者啟動的 process.

Daemon Process: 系統預設啟動的 process.

在Linux系統可以利用

#ps 來觀察目前的process 的狀態

每一個process 都有自己的PID(Process ID)

#pstree 來觀察process 的關係

程式執行的優先性

給定的範圍為 +19 到 -20 數字越**小**越**優先**

只有root 可以給負的NICE值

nice 針對還沒有執行的指令指定優先性

renice 針對已經執行的指令指定優先性

工作控制

指令 **&** 直接在背景執行

jobs 觀察背景執行或是暫停的工作

bg 把暫停的工作丟到背景執行

fg 把暫停的工作丟到前景執行

Ctrl + Z 暫停前景正在執行的工作

screen 補充

- Ctrl + a --> d 卸載 screen
- -ls 列出目前 screen
- -r 連接 screen
- <https://blog.gtwang.org/linux/screen-command-examples-to-manage-linux-terminals/>

cron 定時排程

cron 符合條件就執行

crontab -e 語法

分 時 日 月 星期幾 指令

* * * * *

<代表**每分鐘**執行>

0 * * * *

<**每小時**執行, 分針為0時>

*/5 * * * *

<**每五分鐘**執行一次>

/var/spool/cron/tabs/

存放cron 的排程工作 (個人)

/etc/crontab

系統的排程工作

- 利用 /usr/lib/cron/run-crons

檢查相關工作是否被執行

- /etc/cron.hourly

每**小時**要做的排程工作

- /etc/cron.daily/

每**日**要做的排程工作

- /etc/cron.weekly

每**周**要做的排程工作

- /etc/cron.monthly

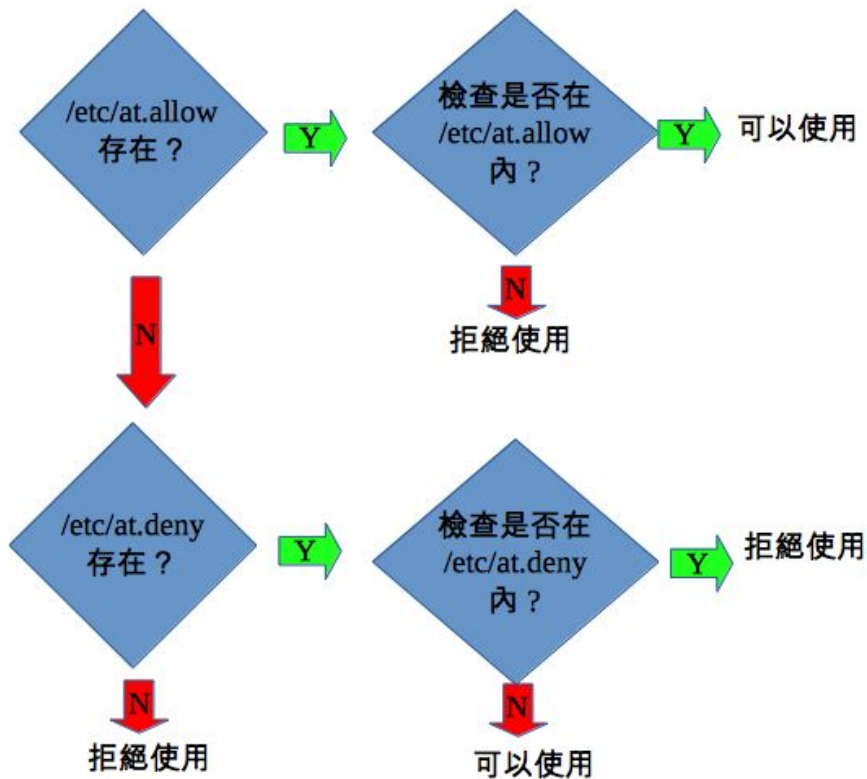
每**月**要做的排程工作

at 在指定的時間執行**一次**

針對突發的工作, 來進行工作的安排

/var/spool/atjobs

存放 at 工作



** section 5 **

	UID	
root	0	
系統帳號	100-499	(參考/etc/login.defs)
一般使用者	>= 1000	(以redHat系列來說 >= 500)

使用 `useradd` 新增使用者

- `-m` 建立家目錄
- `/etc/default/useradd` 決定相關設定(HOME, SHELL ..)
- `/etc/login.defs` 決定相關設定(UID,GID ..)
- 於 `/etc/passwd` (帳號資料庫)新增一筆資料
- 於 `/etc/shadow` (密碼資料庫)新增一筆資料
- 如果有 `-m` 選項, 建立家目錄並且把`/etc/skel`下的檔案複製到使用者家目錄
- 根據`/etc/default/useradd` 加入指定群組於 `/etc/group`, 設定主要群組於 `/etc/passwd`

`/etc/passwd` 帳號資料庫

- john:x:1001:100::/home/john:/bin/bash
- 帳號:使用密碼:UID:GID:註解:家目錄:Shell

`/etc/shadow` 密碼資料庫

- max:\$2a\$05\$1X/4up2NsF30bP9Th2yRVOL.5AIWLbD/uV744bLbsxmdPxfizc6HO:14188:0:99999:7:::
- 帳號名稱:加密過後的密碼::

可以使用`passwd` 變更自己或是別人的密碼(只有root 可以變更別人的密碼)

- root 變更密碼沒有限制
- 一般使用者變更密碼有限制

可以使用 `userdel` 來刪除使用者, `-r` 連同家目錄及信箱都要刪除

群組資料庫 `/etc/group`

- **主要群組**: 紀錄於 `/etc/passwd` 的第四個欄位, 用於決定檔案的群組歸屬
- 一個使用者可以隸屬一到多個群組
- 可以使用 `groupadd` 來新增群組
- 使用 `gpasswd` 來編輯群組 `-a` 加入到群組, `-d` 從群組移除
- 可以使用 `groupdel` 來刪除群組

YaST 模組

`#yast2 users`

權限 Permission

使用 `ls -l` 來觀察權限

`-rwxrw-r-- 1 root users 0 2008-06-21 11:27 test`

第一個位元 file type

`-` --> 檔案

`d` --> 資料夾

`l` --> 連結 link

後面有九個bits 每三個位元為一組, 分別是 **user(owner)**, **group**, **other**的權限

1 link count (連結的數量)

root 擁有者(owner)

users 群組(group)

0 檔案大小

2008-06-21 11:27 m-time

test 檔案名稱

文字權限	數字表示
------	------

r	read	4
---	------	---

w	write	2
---	-------	---

x	execute	1
---	---------	---

Lab: 轉換練習(文字權限表示轉數字表示)

`rw-r--wx` --> **643**

652	-->	rw-r-x-w-
rwX rw r-x	-->	775
755	-->	rwXr-xr-x
rwX --- wX	-->	703

誰可以改變權限？

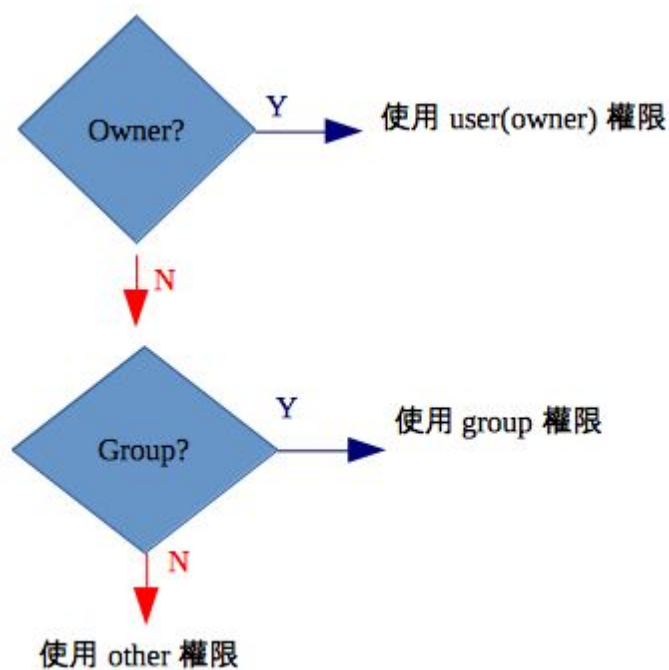
- root
- 檔案擁有人

使用 **chmod** 改變權限

- 數字方式
 - **chmod xxx 檔案名稱**
- 文字方式
 - **chmod 對象 方式 權限 檔案名稱**
 - u / g / o / a
 - + - =
 - r w x

權限判斷流程

-rwXrw-r-- 1 root mis 0 2008-06-21 11:27 test



建立檔案/資料夾的預設權限計算

建立檔案預設權限 666
建立資料夾預設權限 777
umask 022 (定義於/etc/login.defs)

umask 權限遮罩

預設權限計算：預設權限 使用 umask 遮罩(移除)住特定權限

舉例來說, 如果在系統內建立一個檔案

預設權限

r	w	-	r	w	-	r	w	-
---	---	---	---	---	---	---	---	---

umask

-	-	-	-	w	-	-	w	-
---	---	---	---	---	---	---	---	---

r	w	-	r	-	-	r	-	-
	6			4			4	

舉例來說, 如果在系統內建立一個資料夾

預設權限

r	w	x	r	w	x	r	w	x
---	---	---	---	---	---	---	---	---

umask

-	-	-	-	w	-	-	w	-
---	---	---	---	---	---	---	---	---

r	w	x	r	-	x	r	-	x
	7			5			5	

特殊權限

SUID	-->	user的特殊權限,	會繼承 使用者ID
SGID	-->	group的特殊權限,	會繼承群組
Sticky bit	-->	other的特殊權限,	非擁有者本人不能刪除檔案

故權限的表示方式為四碼
第一碼為特殊權限, 後三碼為一般權限

特殊權限的加總			
---------	--	--	--

SUID --> 4 SGID --> 2 Sticky bit --> 1	User 權限	Group 權限	Other 權限
--	---------	----------	----------

sudo

管理者可以利用sudo 授權使用者去執行特定的指令

sudo, sudoedit - execute a command as another user

visudo 會編輯 /etc/sudoers 檔案來授權使用者可以變更其他使用者執行指令
語法如下

使用者 電腦=(使用者) 指令

** section 6 **

RPM 套件的安裝

	如果有舊的軟體	沒有舊版軟體	適用於
-ivh	新舊並存	安裝	Kernel, Lab
-Uvh	升級	安裝	應用程式
-Fvh	升級	不安裝	保持一致性

- -i : install 安裝
- -v : verbose 秀出詳細資訊
- -h : hash 使用#代表安裝進度
- -U : 升級
- -F : 更新

RPM套件查詢/移除

- -q 查詢
- -a 全部
- -l 列出安裝的檔案
- -i 列出相關資訊
- -e 移除套件

YaST 相關模組

```
# yast2 repositories
# yast2 sw_single
```

**** section 7 ****

固定IP

- IP: 網際網路上的邏輯位址
- Subnet Mask: 用來決定/計算所屬網路
- Default Gateway: 預設閘道
- DNS Server: DNS 伺服器位址伺服器位址

可以使用

```
#ifconfig 觀察IP位址以及子網路遮罩
#cat /etc/resolv.conf 觀察DNS 伺服器設定
#route 觀察路由設定
```

網卡設定檔案存放於 [/etc/sysconfig/network/ifcfg-裝置代號](#)
名稱解析 /etc/hosts --> /etc/resolv.conf (根據 /etc/nsswitch.conf 決定順序)

Default Gateway 設定錯誤: 會導致對外連線皆產生錯誤
DNS Server 設定錯誤: 會導致對外名稱連線產生錯誤但是對外的IP連線成功

YaST 模組

```
#yast2 lan
```

iptables 補充

- http://linux.vbird.org/linux_server/0250simple_firewall.php#netfilter
- table
 - 檢查的機制 filter / nat / mangle
- chain
 - 檢查點 INPUT / OUTPUT / FORWARD / PREROUTING / POSTROUTING
- policy
 - 預設的規則
- action
 - 處理的方式 ACCEPT / REJECT / DROP

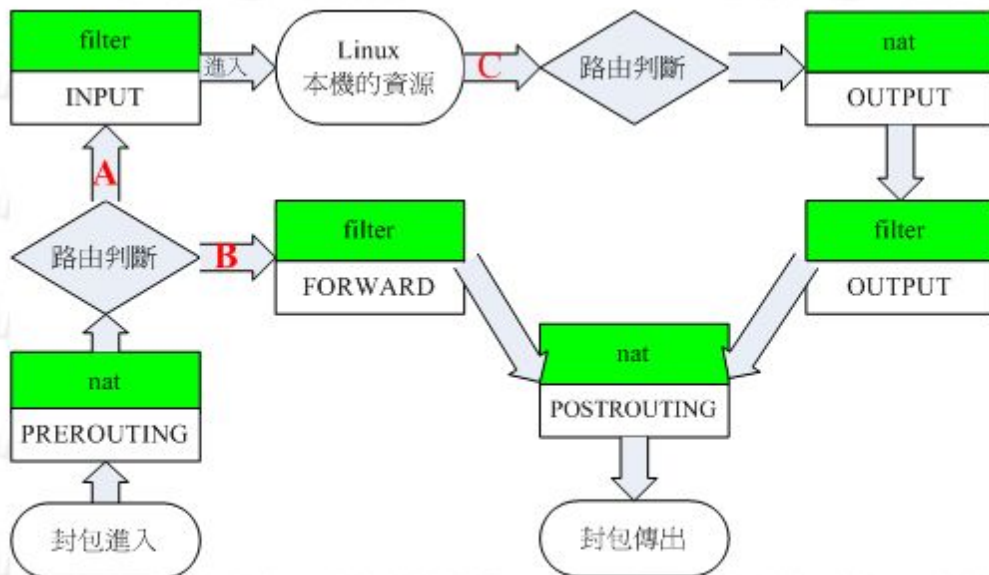


圖 9.3-4、iptables 內建各表格與鏈的相關性(簡圖)

SUSE 透過 SuSEfirewall2 來管理防火牆

- 設定檔 /etc/sysconfig/SuSEfirewall2
- 建議使用 #yast2 firewall 管理

** section 8 **

MBR vs GPT

- 補充
 - <https://read01.com/4NNB6z.html>
 - <https://zh.wikipedia.org/wiki/GUID%E7%A3%81%E7%A2%9F%E5%88%86%E5%89%B2%E8%A1%A8>
 - MBR
 - 有主要分割區 邏輯分割區, 最多到 4P 或是 3P + 1E
 - 單一分割區最大 2TB
 - GPT
 - 全部都是主要分割區, 可以到 128 分割區
 - 單一分割區可到 9.2 ZiB
 - 建議與 UEFI 搭配, 與BIOS會有相容問題
 - 與早期的 Windows 版本有相容性問題

YaST 模組

yast2 disk

有關於裝置的名稱

可以使用 df 來觀察

/dev/hda 第一個IDE控制器的Master (大概是2.6.20以前核心大部分的使用方式)

hda --> IDE裝置 第一個IDE控制器的Master
sda --> SCSI, SATA裝置
約莫2.6.20以後核心, 不管IDE, SCSI, SATA都使用 sda

fdisk vs parted

- fdisk
 - 傳統分割工具, 不支援 GPT
 - 以互動的方式進行, 不適合 batch 方式進行
- parted
 - 支援 GPT 以及超過 2TB分割區
 - 語法 parted [選項] [裝置] [指令] [參數]
 - 如果只下 parted 或是 parted 裝置就是以互動的方式進行
 - 如果下 parted [裝置] [指令] 以非互動的方式進行
 - 選項
 - -l 列出所有裝置的資訊
 - 指令
 - print 列出相關資訊
 - mkpart 建立分割區
 - mkpart <part-type> [<fs-type>] <start> <end>
 - <part-type>
 - primary, logical, extended
 - rm 刪除分割區
 - rm <part-number>
 - 補充
 - http://linux.vbird.org/linux_basic/0230filesystem.php#parted

常見的partition system id

- 82 swap
- 83 linux
- 8e linux LVM
- fd linxx RAID

可以利用 #dumpe2fs 來觀察 ext file system的 super block

使用 #mke2fs 裝置代號 來建立 EXT2 檔案系統

使用 #mke2fs -j 裝置代號 來建立 EXT3 檔案系統

使用 #mkfs -t 系統類型 來建立 指定 檔案系統

使用 #tune2fs -j 裝置代號 來將EXT2 轉成 EXT3 檔案系統

可以使用 df 來列出 已經掛載 file system

mount 語法

mount 裝置/資源 本地目錄

mount device/source mount point

可以使用 **umount** **裝置/資料夾** 來卸載裝置或是資料夾
可以使用 **fuser -v** **資料夾** 來觀察 該資料夾使用情形

blkid 顯示 block device UUID

傳統方式mount的注意事項

- 掛載的目標為**裝置的絕對位置**,如果絕對位置改變,會找不到分割區

利用 **e2label** 來顯示/更改 ext2/ext3 檔案系統的 Label name

可以使用 **dumpe2fs 裝置代號** | grep vol 來觀察File system volume name

開機時系統會根據 **/etc/fstab** 來決定

開機要掛載那些**裝置/資源**

/etc/fstab 內有6個欄位 分別是

裝置/資源 **本機目錄** **FS_type** **Mount_Options** **dump_fre** **fsck_order**

Btrfs

- 內建 Copy On Write
- 具有快照功能
- 可以使用 subvolume 進行進階管理(quota / isolate)
- 支援較大的磁碟容量, 動態 inode, 壓縮功能, SSD 最佳化
- 可以透過 btrfs-convert 將 ext2 / 3 / 4 轉成 btrfs
- 使用空間的觀察
 - #btrfs filesystem df 路徑**
 - #btrfs filesystem usage 路徑**

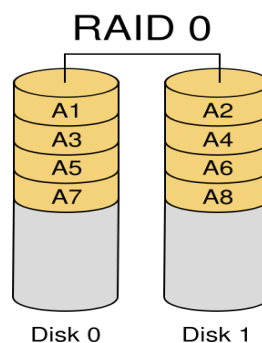
RAID 磁碟陣列

RAID 0

可用容量: 4G

優點: **讀寫速度快**

缺點: **沒有容錯能力**

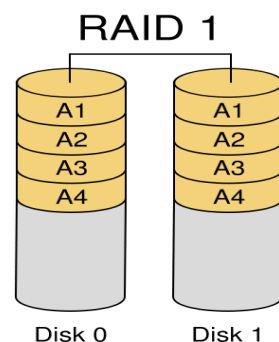


RAID 1

可用容量: 2G

優點: **有容錯能力**

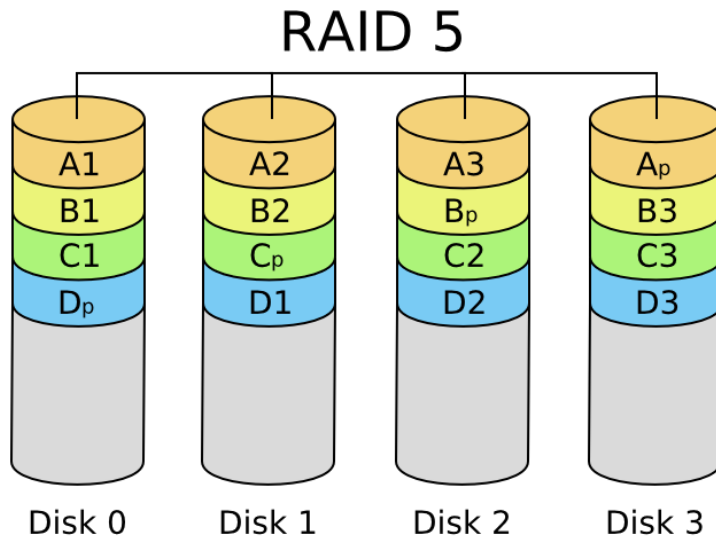
缺點: **寫速度慢**



RAID 5 (至少三顆HD)

可用容量: 4G

優點: 有容錯能力, 寫入速度OK



可以觀察 `/proc/mdstat`
來觀察RAID的情形

補充相關指令 `mdadm`

模擬 硬碟 fail 請同時觀察 `/proc/mdstat`

```
# mdadm --manage /dev/md0 -f /dev/sda8
```

```
mdadm: set /dev/sda8 faulty in /dev/md0
```

```
# mdadm --manage /dev/md0 -r /dev/sda8
```

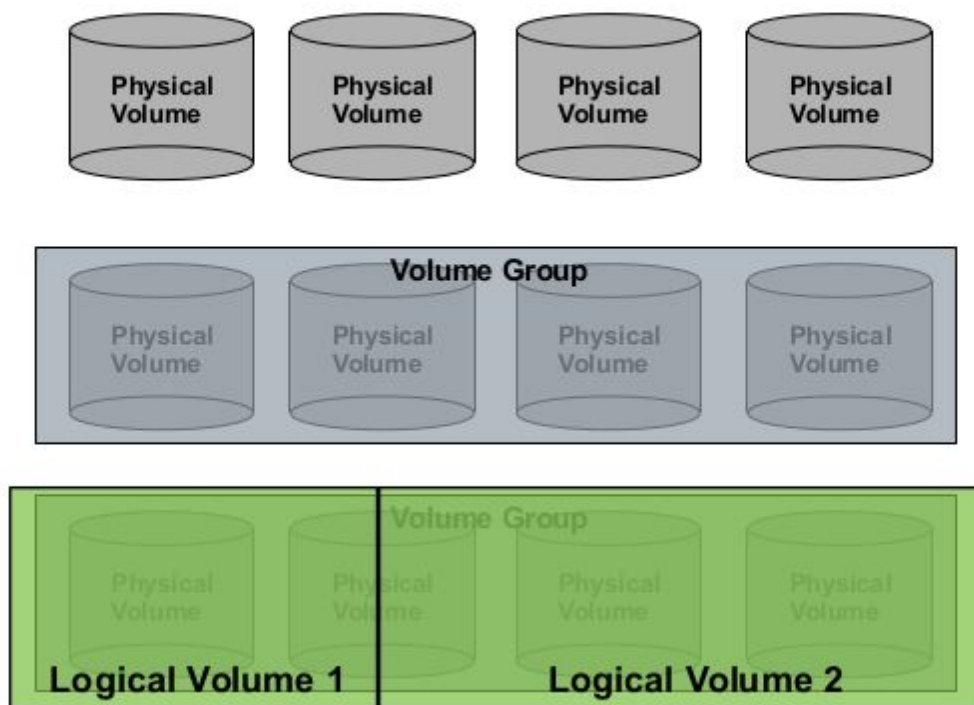
```
mdadm: hot removed /dev/sda8
```

```
# mdadm --manage /dev/md0 -a /dev/sda8
```

```
mdadm: re-added /dev/sda8
```

LVM

LVM Components



手動建立方式

Step1:

- - 建立分割區, system id: 8e
 - vgscan
 - pvcreate

Step2

- vgcreate
- lvcreate
- mke2fs
- 建立 mount point
- mount
- 修改/etc/fstab

建議使用 YaST 來管理

YaST 模組

yast2 disk

補充:

- 在 SLES 10 使用 ext2online 來針對 LVM2 來讓變更生效, 使用 e2fsadmin 來針對 LVM1 來讓變更生效.
- 在 SLES 11 / 12 ext2online 指令被包括在 resize2fs 指令內, 所以使用 resize2fs 來讓變更生效.
- 在 SLES 12 XFS 使用 xfs_growfs 路徑 來讓變更生效.
- <http://tldp.org/HOWTO/LVM-HOWTO/extendlv.html>

** section 9 **

網路對時

ntpdate time.stdtime.gov.tw

#dmesg | grep 關鍵字 觀察相關訊息
系統內大部分的log皆存放於 [/var/log](#) 目錄

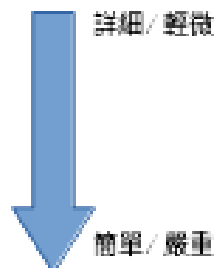
rsyslog

- 設定檔 /etc/rsyslog.conf
- 補充
 - http://linux.vbird.org/linux_basic/0570syslog.php#syslogd

Facilities 從哪一個服務或是設備傳來
Priorities log level, 優先性

Priorities

- debug
- info
- notice
- warning
- err
- crit
- alert
- emerg/panic



YaST 模組

#yast2 view_anymsg

```
# journalctl – the old “cat /var/log/messages”
# journalctl -f – the old “tail -f /var/log/messages”
```

logrotate

/etc/logrotate.conf log封存的設定檔(Global)

- **weekly** 每週封存
- **rotate 4** 保留4份
- **create** 封存完建立空檔
- **dateext** 使用日期為檔名一部份
- **include /etc/logrotate.d/** 相關要封存的log 設定存放目錄

#supportconfig - SLES 狀態檢查

**** section 10 ****

Linux 的 modules 存放於 /lib/modules/核心版本 目錄下
可以使用下列指令管理 modules

lsmod	列出 modules
modprobe	載入 modules
rmmod	移除 modules
modinfo	顯示 modules 資訊

系統 使用 udev 來建立 / 維護硬體名稱
可以觀察 /etc/udev/rules.d 下面的設定來取得