

**** 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 >

**** Section 2 ****

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

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

**** section 3 ****

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

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>

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

? 單一字元
* 零或多個字元
[] 單一字元符合條件

find 語法
find 路徑 條件式 執行命令

locate 利用 updatedb 搜尋整個系統
whereis 尋找命令, 文件, source
which 尋找命令
type 判斷檔案的類型

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

**** section 4 ****

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

history 列出指令歷程

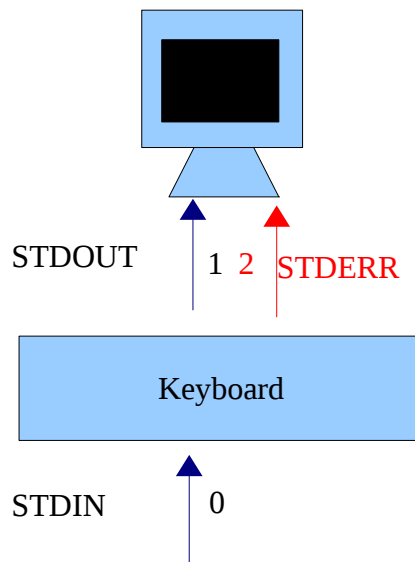
- 預設 HISTSIZE=1000
- 存放在 ~/.bash_history
- 可以用 Ctrl + R 來搜尋

su 切換使用者 shell

- -l login shell

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\$()



> 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

**** Section 5 ****

#yast 文字介面控制中心
#yast2 圖形介面控制中心

**** Section 6 ****

 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.5A1WLbD/uV744bLbsxmdPxfizc6H0:14188:0:99999:7:::
- **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 **groupmod** 來編輯群組 -A 加入到群組, -R 從群組移除
- 可以使用 **groupdel** 來刪除群組

****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-**

rwxrwxr-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

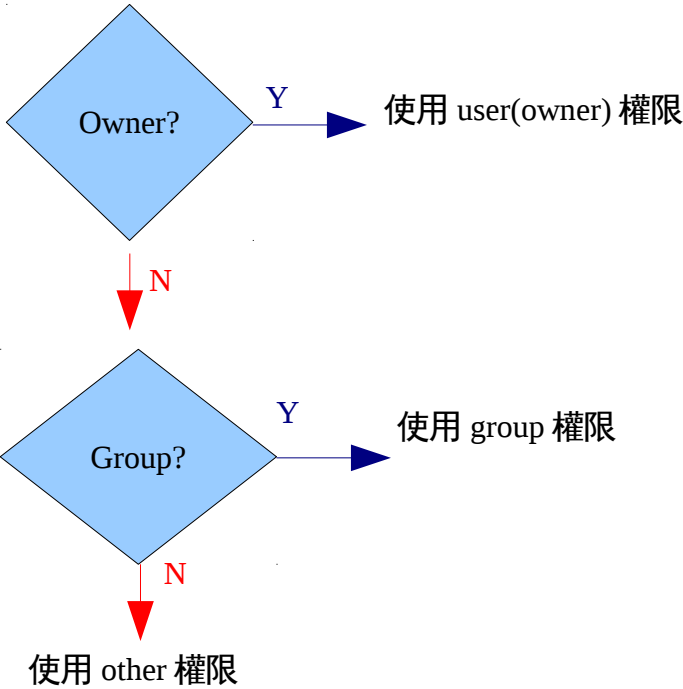
- 數字方式
`chmod xxx 檔案名稱`

- 文字方式

chmod	u	+	r	檔案名稱
	g	-	w	
	o	=	x	
	a	-		

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/profile)
```

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 -

$$r \quad w \quad - \quad r \quad - \quad - \quad r \quad - \quad -$$

$$6 \qquad \qquad \qquad 4 \qquad \qquad \qquad 4$$

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

Q:差異為?

s:原來有 x

S:原來無 x

t:原來有 x

T:原來無_x

Lab: 權限轉換練習

rwSr--wT	-->	5742
4752	-->	rwsr-x-w-
rwSrwsr-t	-->	7675
6744	-->	rwsr-Sr--

ACL

傳統的檔案權限針對的對象

- 擁有者(owner)
- 群組(group)
- 其他(other)

倘若檔案或是資料夾的權限, 指定的方式超過三個以上

此時可以利用 ACL 指定特定使用者 / 群組 所可以擁有的權限, 來彌補傳統權限的不足

ACL(Access Control List)

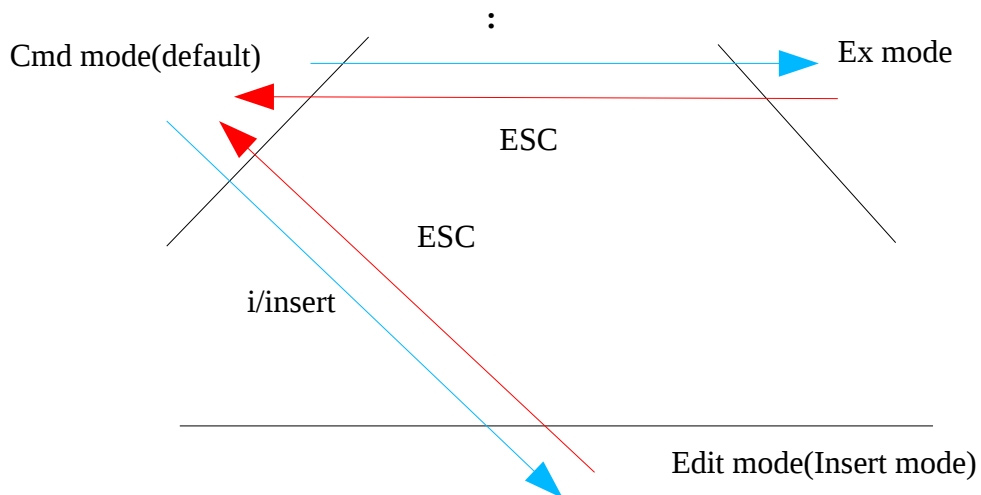
- 在 file system 上面必須設定 **acl** 的選項
- 可以使用 **setfacl** 設定 ACL
 - **-m** 編輯 ACL 設定
 - **-x** 移除單項 ACL 設定
 - **-b** 移除所有 ACL 設定
 - **-d** 設定 Default ACL
 - **-M** 還原 ACL 設定
- 可以使用 **getfacl** 觀察 ACL 設定

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

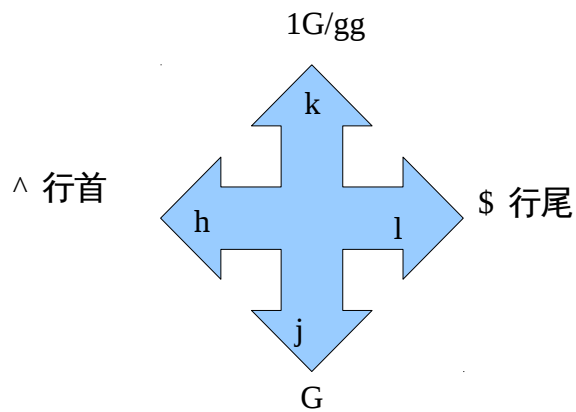
當設定 ACL, 中間的 4-6 的 bit 就非原本的群組權限, 而是 **acl mask**

- **acl mask** --> 特定使用者或是特定群組可以使用的最大權限
- **default acl** --> 針對資料夾設定 default acl 其子資料夾及檔案可以繼承設定的 acl

Section 7 Use Linux Text Editors



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

**Section 8 **

套件的安裝

Source code *.tar.gz <必須解決相依性問題>
RPM 套件 *.rpm <必須解決相依性問題>

現行狀況 <利用線上的 Sever 解決相依性的問題>

APT(Debian)

YAST(SuSE)

YUM(RedHat)

zypper 文字介面的軟體安裝指令

語法

zypper [全域選項] 指令 [指令選項] [參數] ...

常用的 zypper 指令

```
zypper                   # 列出可用的全域選項與指令
zypper help search      # 列出有關 search 指令的說明
zypper lp               # 列出所需的修補程式
zypper patch            # 安裝所需的修補程式
zypper se sqlite        # 搜尋 sqlite 套件
zypper rm sqlite2       # 移除 sqlite2 套件
zypper in sqlite3       # 安裝 sqlite3 套件
zypper in yast*          # 安裝所有符合 'yast*' 名稱的套件
zypper repos            # 列出安裝的套件庫
zypper up               # 使用較新版本更新已安裝的套件
```

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

<http://rpmfind.net> 搜尋 RPM 網站