

# 計算機概論A班 實習課

W2

# 課程內容

## → Linux 背景知識

- ◆ 什麼是 Linux
- ◆ 發展及升級
- ◆ 使用者與群組
- ◆ 檔案系統目錄結構

## → 基本指令實作

- ◆ 絕對/相對路徑的移動
- ◆ 更改使用者權限
- ◆ 編輯文檔相關

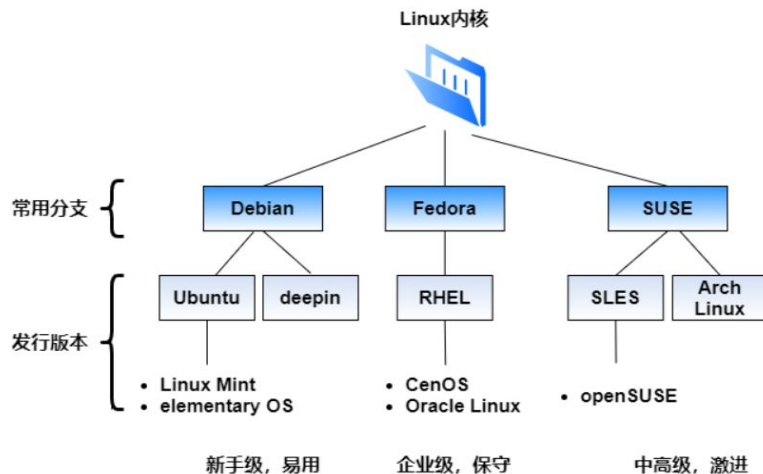
## → 問答時間

## → HW

- ◆ 查詢TCP/IP背景知識?
- ◆ Port是什麼?
- ◆ TCP/IP是什麼?

# Linux 背景知識 - 什麼是 Linux ?

- Linux 是一種自由開放原始碼的類 Unix 作業系統
- 廣泛運用於伺服器 and 嵌入式系統中
- 具有可移植性的 Open Source 的作業系統，它的程式碼可以被修改適合在各種機器上運行
- 目前主流的 Linux 發佈版本包括: Debian、Fedora、CentOS、Ubuntu 等



# Linux 背景知識 - 發展與線上升級

Linux 的內核版本編號:2.6.32-358.el6.x86\_64 (通過指令 `uname -mrs` 可以看到)  
2→主版本 6→次版本 32→釋出版本 358.el6.x86\_64→修改版本

版本會分為:1.開發中版本範例(次版本為奇數):2.5.x  
2.穩定版本範例(次版本為偶數):2.6.x

| OS / Command                     | Debian | Ubuntu | Fedora | CentOS | FreeBSD |
|----------------------------------|--------|--------|--------|--------|---------|
| <code>uname -mrs</code>          | v      | v      | v      | v      | v       |
| <code>cat /os-release</code>     | v      | v      | v      | v      |         |
| <code>hostnamectl</code>         | v      | v      | v      | v      |         |
| <code>lsb_release -a</code>      | v      | v      |        |        |         |
| <code>freebsd-version -ku</code> | v      |        |        |        | v       |

# Linux 背景知識 - 發展與線上升級

Linux 開發商有提供『線上升級』機制，透過此機制，原版光碟就只有第一次安裝時需要用到，其他時候只要有網路，就能夠取得原開發商所提供的任何軟體。

在 `dpkg` 管理機制上就有開發出 `APT` 的線上升級機制，`RPM` 則依開發商的不同，有不同的下載軟體指令模式。

| distribution 代表 | 軟體管理機制      | 使用指令                       | 線上升級機制(指令)                 |
|-----------------|-------------|----------------------------|----------------------------|
| Red Hat/Fedora  | <b>RPM</b>  | <code>rpm, rpmbuild</code> | <code>YUM (yum)</code>     |
| Debian/Ubuntu   | <b>DPKG</b> | <code>dpkg</code>          | <code>APT (apt-get)</code> |

# Linux 背景知識 - 使用者與群組

→ 關於身份：

◆ 身份分類：

- user (owner)
- group
- others

◆ 群組 (group) 的主要用意：

- 將帳號歸類，有助於專案開發
- 使用者可加入多個群組

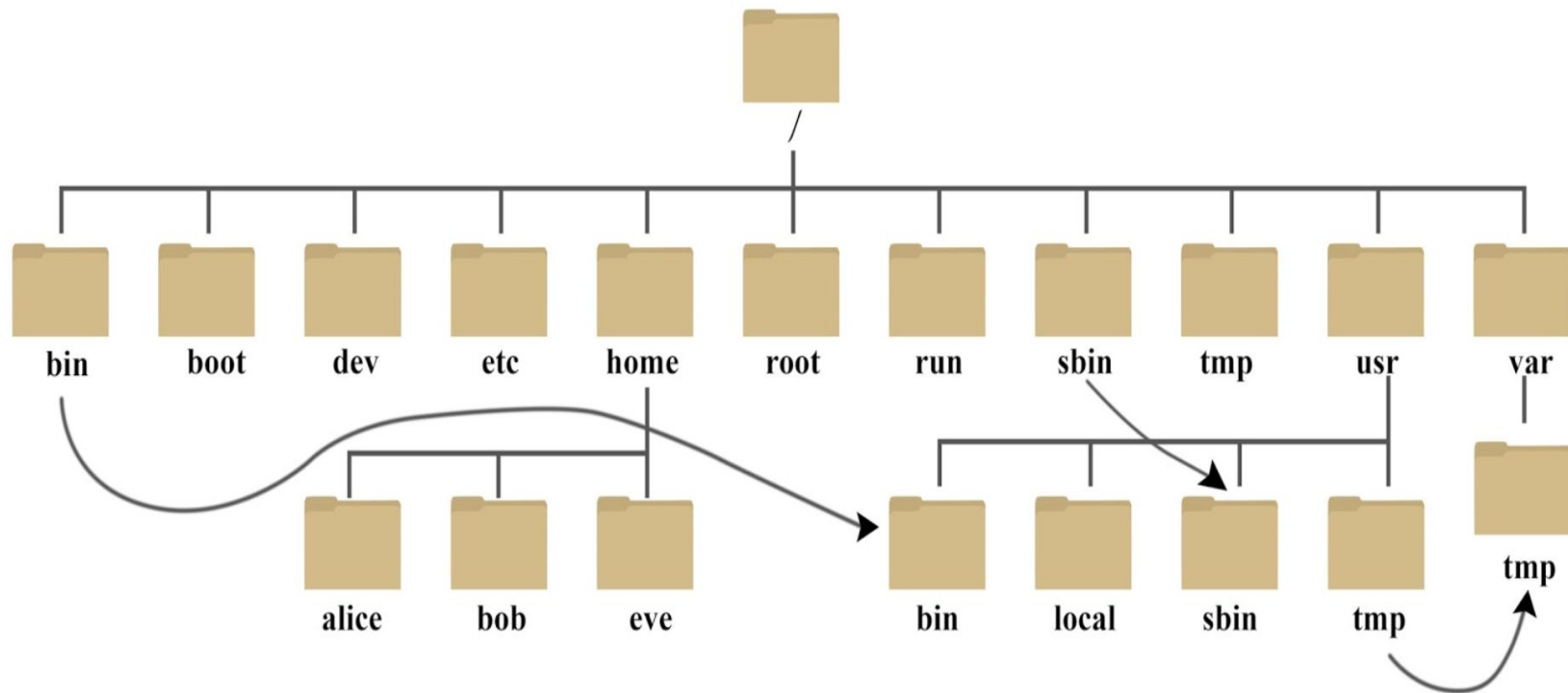
→ 關於身份類別：

◆ 系統管理員：root

◆ 一般帳號：

- 系統帳號（用於操作系統工作的帳號 e.g. 關閉電腦 etc.）
- 一般使用者帳號（用於一般使用者登入用）

# Linux 背景知識 - 檔案系統目錄結構



# Linux 各個目錄存放的內容

- **/**: 根目錄, 所有檔案皆從根目錄開始, 只有root 使用者具該目錄的許可權
- **/root**: 該目錄為系統管理員, 也稱作超級許可權者的使用者主目錄
- **/bin**: 存放著指令的資料夾
- **/boot**: 系統啟動時必須讀取的檔案, 包括系統核心
- **/home**: 存放普通使用者的家目錄, 每個使用者都有自己的家目錄
- **/etc**: 放置與系統設定、管理相關的檔案
- **/usr**: 這是一個非常重要的目錄, 使用者的應用程式和檔案都放在這個目錄下, 類似與 windows 下的 program files 目錄
- **/media**: 可用來做為光碟、軟碟片、隨身碟與其他分割區的自動掛載點
- **/tmp**: 供全部使用者暫時放置檔案的目錄
- **/var**: 系統執行時, 內容經常變動的資料或暫存檔(log file), 都會放置在這個目錄裡



# Linux 背景知識 - 命名規則

- 區分英文字元的大小寫, 以小寫為主
- 長度至多 256 個字元, 可包括字母、數字、"."(點)、"\_"(下劃線) 和 "-"(連字元)
- 不可使用的字元: "?"(問號), "\*" (星號), " "(空格), "\$"(貨幣符), "&", "("括號, "/" (斜線), ".."
- 前面有"."(點), 為隱藏檔 e.g. .filename

# Linux 指令

| 指令    | 功能, 用途       |
|-------|--------------|
| cd    | 變換目錄         |
| pwd   | 顯示目前的目錄的所在位置 |
| ls    | 顯示目錄下的所有檔案   |
| touch | 建立檔案         |
| cp    | 複製           |
| cat   | 查看檔案內容       |
| vi    | 編輯檔案         |

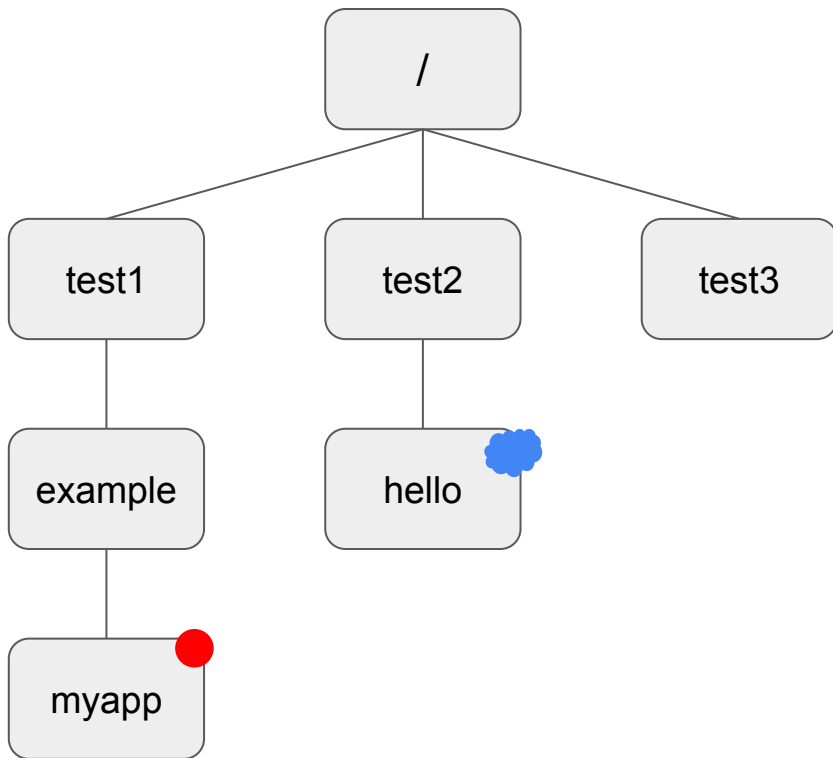
# Linux 指令

| 指令       | 功能, 用途       |
|----------|--------------|
| mkdir    | 建立一個新目錄      |
| mkdir -p | 同時建立多個目錄     |
| rmdir    | 刪除一個裡面是空的空目錄 |
| rm       | 刪除檔案         |
| rm -r    | 強制刪除         |
| mv       | 移動, 重新命名     |

# Linux 指令 - 絕對/相對路徑的移動

- 絕對路徑:路徑的寫法『一定由根目錄 / 寫起』,  
例如: `/usr/share/doc` 這個目錄。
- 相對路徑:路徑的寫法『不是由 / 寫起』,  
例如:由 `/usr/share/doc` 要到 `/usr/share/man` 底下時, 寫成:『`cd ../man`』, 相對路徑意指『相對於目前工作目錄的路徑!』
- 比較特殊的目錄
  - `.` :代表此層目錄
  - `..` :代表上一層目錄
  - `-` :代表前一個工作目錄
  - `~` :代表『目前使用者身分』的家目錄

# Linux 指令 - 絕對/相對路徑的移動



目前位置在 `myapp` 要移動到 `hello` 如何表示？

1. `cd /test2/hello`
2. `cd ../../../test2/hello`

# Linux 指令 - 查看檔案目錄資訊

- `ls -l`:顯示目錄下的所有檔案詳細資訊
- `ls -al`:顯示目錄下的所有檔案(包含隱藏檔)詳細資訊

```
[ec2-user@ip-172-31-50-248 ~]$ ls -al
total 12
drwx----- 3 ec2-user ec2-user 74 Sep 14 16:50 .
drwxr-xr-x 3 root      root      22 Sep 14 16:50 ..
-rw-r--r-- 1 ec2-user ec2-user 18 Jul 15 2020 .bash_logout
-rw-r--r-- 1 ec2-user ec2-user 193 Jul 15 2020 .bash_profile
-rw-r--r-- 1 ec2-user ec2-user 231 Jul 15 2020 .bashrc
drwx----- 2 ec2-user ec2-user 29 Sep 14 16:50 .ssh
```

# Linux 指令 - 更改檔案或目錄權限

- chown 可修改檔案或目錄的擁有者及群組

```
-rw-rw-r-- 1 ec2-user ec2-user  9 Sep 14 17:03 test.txt
[ec2-user@ip-172-31-50-248 ~]$ sudo chown root test.txt
[ec2-user@ip-172-31-50-248 ~]$ ls -al
total 20
drwx----- 3 ec2-user ec2-user 102 Sep 14 17:04 .
drwxr-xr-x 3 root      root      22 Sep 14 16:50 ..
-rw-r--r-- 1 ec2-user ec2-user  18 Jul 15  2020 .bash_logout
-rw-r--r-- 1 ec2-user ec2-user 193 Jul 15  2020 .bash_profile
-rw-r--r-- 1 ec2-user ec2-user 231 Jul 15  2020 .bashrc
drwx----- 2 ec2-user ec2-user  29 Sep 14 16:50 .ssh
-rw-rw-r-- 1 ec2-user ec2-user  9 Sep 14 17:04 test
-rw-rw-r-- 1 root      ec2-user  9 Sep 14 17:03 test.txt
```

# Linux 指令 - 更改使用者權限

```
[ec2-user@ip-172-31-50-248 ~]$ ls -al
total 12
drwx----- 3 ec2-user ec2-user 74 Sep 14 16:50 .
drwxr-xr-x 3 root      root      22 Sep 14 16:50 ..
-rw-r--r-- 1 ec2-user ec2-user 18 Jul 15 2020 .bash_logout
-rw-r--r-- 1 ec2-user ec2-user 193 Jul 15 2020 .bash_profile
-rw-r--r-- 1 ec2-user ec2-user 231 Jul 15 2020 .bashrc
drwx----- 2 ec2-user ec2-user 29 Sep 14 16:50 .ssh
```

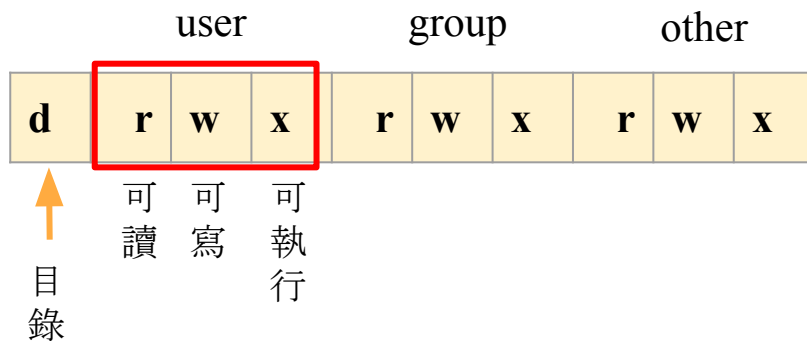
權限設定

擁有者

群組



# Linux 指令 - 目錄權限



r:可讀取  
w:可寫入  
x:可執行

|   |   |   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|---|---|
| d | r | w | x | r | w | - | r | - | - |
|---|---|---|---|---|---|---|---|---|---|

# Linux 指令 - 更改使用者權限-用法一

效果:更改 9 碼權限

用法一:`chmod [擁有者權限][群組權限][其他人權限] filename`

|      |
|------|
| r:4  |
| w:2  |
| x:1  |
| (加總) |

說明:會有三個數字, 分別代表「擁有者」「群組」「其他人」的權限, 數字代表的意義如右上。

範例:

```
chmod 755 testfile
```

// 755 就是 [4+2+1][4+1][4+1]

# Linux 指令 - 更改使用者權限-用法二

效果:更改 9 碼權限

用法二:

|       |   |                         |   |       |
|-------|---|-------------------------|---|-------|
| chmod | u | +(加入)<br>-(除去)<br>=(設定) | r | 檔案或目錄 |
|       | g |                         | w |       |
|       | o |                         | x |       |
|       | a |                         |   |       |

範例:

```
chmod u=rwx,g=rx,o=rx testfile
```

```
chmod o+w testfile
```

```
chmod a-x testfile
```

u:user 擁有者  
g:group 群組  
o:other 其他人  
a:all 全部人

**注意！**

Windows 的檔案系統不同於 Linux，所以 git bash 雖然可以打 chmod 指令，但是不會實際影響檔案的權限。

# 檔案、目錄 執行權限

**r**：可以讀取

**w**：可以寫入(包含建立、刪除、修改、移動)

**x**：可以執行？

對於檔案來說，可執行是指 **bash** 指令內容

對於目錄來說，可執行是指有沒有權限進入資料夾

# Linux 指令 - 編輯文檔

→ nano

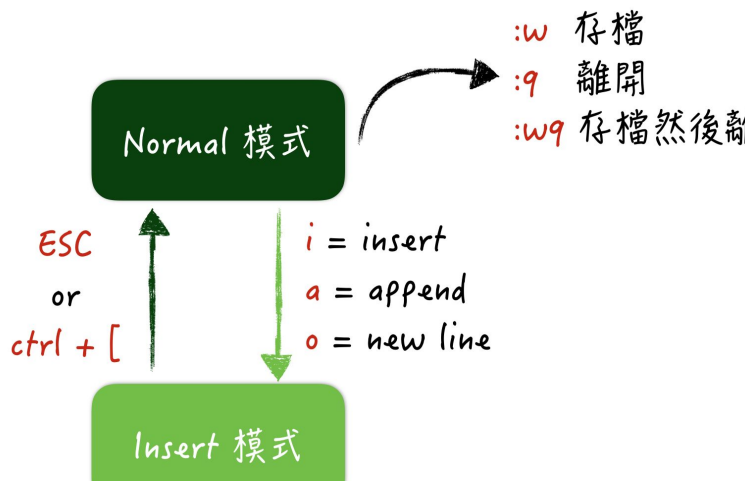
- ◆ Ctrl + C: 顯示游標所在
- ◆ Ctrl + W: 查詢命令, 按下後會跳轉到末行的反白位置, 輸入要查詢的內容在按 enter 及可。

```
[ Welcome to nano. For basic help, type Ctrl+G. ]
^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute    ^C Location   M-U Undo
^X Exit      ^R Read File  ^\ Replace    ^U Paste       ^J Justify    ^/ Go To Line M-E Redo
```

# Linux 指令 - 編輯文檔

→ vi/vim (vim相當於vi的升級版)

- ◆ 一般模式:可使用上下左右進行游標宜動
- ◆ 編輯模式:編輯文字、刪除字元及複製貼上檔案資料



# Linux 指令 - 編輯文檔

→ vi/vim(vim相當於vi的升級版)

## ◆ 命令列模式

※！為『強制』的意思

|     |                      |                       |                                |
|-----|----------------------|-----------------------|--------------------------------|
| :w  | 將編輯的資料寫入硬碟檔案中        | :wq                   | 儲存後離開(若為 :wq! 則為強制儲存後離開)       |
| :w! | 若檔案屬性為『只讀』時, 強制寫入該檔案 | :w[filename]          | 將編輯的資料儲存成另一個檔案(類似另存新檔)         |
| :q  | 離開                   | :r[filename]          | 在編輯的資料中, 讀入另一個檔案的資料            |
| :q! | 強制離開不儲存檔案            | :n1,n2<br>w[filename] | 將 n1 到 n2 的內容儲存成 filename 這個檔案 |

# Linux 指令 - 複製檔案或目錄

- `cp`: 複製檔案
  - 指令: `cp -參數 來源檔案 目標檔案`
  - 參數:
    - p: 將檔案本身屬性(權限、所有者、時間)同時複製過去(一般用於備份居多)
    - r: 針對目錄下檔案做遞歸複製(整個目錄下每一個檔案複製到你想要的位置)
- e.g. `cp -r dir1 dir2`



# 討論時間

允許擁有者讀取和寫入, 並允許其他人讀取？

`chmod a+rwx filename` 是什麼意思？

# 作業一

## 查閱 TCP/IP 背景知識！

1. TCP/IP 是甚麼？
2. Port 是甚麼？

- 各組作業請組長上傳至tronclass，實習課 HW1
- 檔名：e.g. linux-hw1-group1 (繳交 word 檔即可)
- 期限：10/17 23:59前

