



Linux 的檔案權限與目錄配置

Linux 最優秀的地方之一，就在於它的多人多工環境。而為了讓各個使用者具有較保密的檔案資料，因此檔案的權限管理就變的很重要了。Linux 一般將檔案可存取的身份分為三個類別，分別是 owner/group/others，且三種身份各有 read/write/execute 等權限。若管理不當，你的 Linux 主機將會變的很『不蘇服！@_@』。另外，你如果首次接觸 Linux 的話，那麼，在 Linux 底下這麼多的目錄/檔案，到底每個目錄/檔案代表什麼意義呢？底下我們就來一一介紹呢！

6.1 使用者與群組

經過第五章的洗禮之後，你應該可以在 Linux 的指令列模式底下輸入指令了吧？接下來，當然是要讓你好好的瀏覽一下 Linux 系統裡面有哪些重要的檔案囉。不過，每個檔案都有相當多的屬性與權限，其中最重要的可能就是檔案的擁有者的概念了。所以，在開始檔案相關資訊的介紹前，鳥哥就先就簡單的(1)使用者及(2)群組與(3)非本群組外的其他人等概念作個說明吧～好讓你快點進入狀況的哩！^_^

1. 檔案擁有者

初次接觸 Linux 的朋友大概會覺得很怪異，怎麼『Linux 有這麼多使用者，還分什麼群組，有什麼用？』。這個『使用者與群組』的功能可是相當健全而好用的一個安全防護呢！怎麼說呢？由於 Linux 是個多人多工的系統，因此可能常常會有多人同時使用這部主機來進行工作的情況發生，為了考慮每個人的隱私權以及每個人喜好的工作環境，因此，這個『檔案擁有者』的角色就顯的相當的重要了！

例如當你將你的 e-mail 情書轉存成檔案之後，放在你自己的家目錄，你總不希望被其他人看見自己的情書吧？這個時候，你就把該檔案設定成『只有檔案擁有者，就是我，才能看與修改這個檔案的內容』，那麼即使其他人知道你有這個相當『有趣』的檔案，不過由於你有設定適當的權限，所以其他人自然也就無法知道該檔案的內容囉！

2. 群組概念

那麼群組呢？為何要設定檔案還有所屬的群組？其實，**群組最有用的功能之一，就是當你在團隊開發資源的時候啦！**舉例來說，假設有兩組專題生在我的主機裡面，第一個專題組別為 projecta，裡面的成員有 class1, class2, class3 三個；第二個專題組別為 projectb，裡面的成員有 class4, class5, class6。這兩個專題之間是有競爭性質的，但卻要繳交同一份報告。每組的組員之間必須要能夠互相修改對方的資料，但是其他組的組員則不能看到本組自己的檔案內容，此時該如何是好？

在 Linux 底下這樣的限制是很簡單啦！我可以經由簡易的檔案權限設定，就能限制非自己團隊(亦即是群組囉)的其他人不能夠閱覽內容囉！而且亦可以讓自己的團隊成員可以修改我所建立的檔案！同時，如果我自己還有私人隱密的文件，仍然可以設定成讓自己的團隊成員也看不到我的檔案資料。很方便吧！

另外，如果 teacher 這個帳號是 projecta 與 projectb 這兩個專題的老師，他想要同時觀察兩者的進度，因此需要能夠進入這兩個群組的權限時，你可以設定 teacher 這個帳號，『同時支援 projecta 與 projectb 這兩個群組！』，也就是說：**每個帳號都可以有多個群組的支援呢！**

這樣說或許你還不容易理解這個使用者與群組的關係吧？沒關係，我們可以使用目前『家庭』的觀念來進行解說喔！假設有一家人，家裡只有三兄弟，分別是王大毛、王二毛與王三毛三個人，而這個家庭是登記在王大毛的名下的！所以，『王大毛家有三個人，分別是王大毛、王二毛與王三毛』，而且這三個人都有自己的房間，並且共同擁有一個客廳喔！

- **使用者的意義**：由於王家三人各自擁有自己的房間，所以，王二毛雖然可以進入王三毛的房間，但是二毛不能翻三毛的抽屜喔！那樣會被三毛 K 的！因為抽屜裡面可能有三毛自己私人的東西，例如情書啦，日記啦等等的，這是『私人的空間』，所以當然不能讓二毛拿囉！
- **群組的概念**：由於共同擁有客廳，所以王家三兄弟可以在客廳打開電視機啦、翻閱報紙啦、坐在沙發上面發呆啦等等的！反正，只要是在客廳的玩意兒，三兄弟都可以使用喔！因為大家都是一家人嘛！

這樣說來應該有點曉得了喔！那個『王大毛家』就是所謂的『群組』囉，至於三兄弟就是分別為三個『使用者』，而這三個使用者是在同一個群組裡面的喔！而三個使用者雖然在同一群組內，但是我們可以設定『權限』，好讓某些使用者個人的資訊不被群組的擁有者查詢，以保有個人『私人的空間』啦！而設定群組共享，則可讓大家共同分享喔！

3. 其他人的概念

好了，那麼今天又有個人，叫做張小豬，他是張小豬家的人，與王家沒有關係啦！這個時候，除非王家認識張小豬，然後開門讓張小豬進來王家，否則張小豬永遠沒有辦法進入王家，更不要說進到王三毛的房間啦！不過，如果張小豬透過關係認識了三毛，並且跟王三毛成為好朋友，那麼張小豬就可以透過三毛進入王家啦！呵呵！沒錯！那個張小豬就是所謂的『其他人，Others』囉！

因此，我們就可以知道啦，在 Linux 裡面，任何一個檔案都具有『User, Group 及 Others』三種身份的個別權限，我們可以將上面的說明以底下的圖示來解釋：

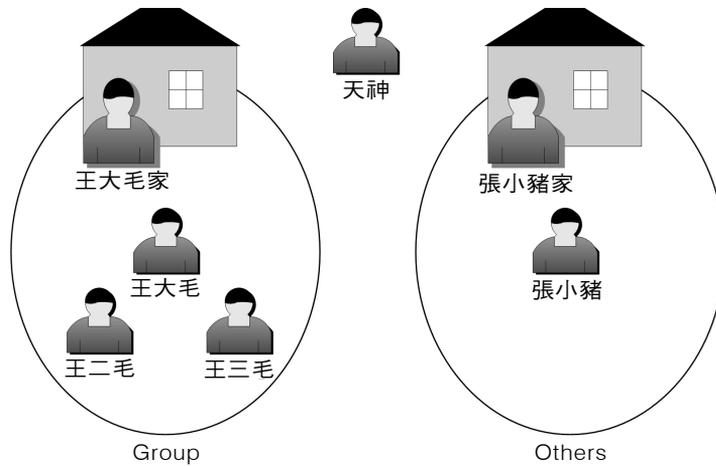


圖 6.1.1 每個檔案的擁有者、群組與其他人的示意圖

我們以王三毛為例，王三毛這個『檔案』的擁有者為王三毛，他屬於王大毛這個群組，而張小豬相對於王三毛，則只是一個『其他人(others)』而已。

不過，這裡有個特殊的人物要來介紹的，那就是『萬能的天神』！這個天神具有無限的神力，所以他可以到達任何他想要去的地方，呵呵！那個人在 Linux 系統中的身份代號是『root』啦！所以要小心喔！那個 root 可是『萬能的天神』喔！

無論如何，『使用者身份』與該使用者所支援的『群組』概念，在 Linux 的世界裡面是相當的重要的，它可以幫助你讓你的多工 Linux 環境變的更容易管理！更詳細的『身份與群組』設定，我們將在第十四章帳號管理再進行解說。底下我們將針對檔案系統與檔案權限來進行說明。

◆ Linux 使用者身份與群組記錄的檔案

在我們 Linux 系統當中，預設的情況下，所有的系統上的帳號與一般身份使用者，還有那個 root 的相關資訊，都是記錄在/etc/passwd 這個檔案內的。至於個人的密碼則是記錄在/etc/shadow 這個檔案下。此外，Linux 所有的群組名稱都記錄在/etc/group 內！這三個檔案可以說是 Linux 系統裡面帳號、密碼、群組資訊的集中地囉！不要隨便刪除這三個檔案啊！^_^

至於更多的與帳號群組有關的設定，還有這三個檔案的格式，不要急，我們在第十四章的帳號管理時，會再跟大家詳細的介紹的！這裡先有概念即可。

6.2 Linux 檔案權限概念

大致瞭解了 Linux 的使用者與群組之後，接著下來，我們要來談一談，這個檔案的權限要如何針對這些所謂的『使用者』與『群組』來設定呢？這個部分是相當重要的，尤其對於初學者來說，因為檔案的權限與屬性是學習 Linux 的一個相當重要的關卡，如果沒有這部份的概念，那麼你將老是聽不懂別人在講什麼呢！尤其是當你在你的螢幕前面出現了『Permission deny』的時候，不要擔心，『肯定是權限設定錯誤』啦！呵呵！好了，閒話不多聊，趕快來瞧一瞧先。

6.2.1 Linux 檔案屬性

嗯！既然要讓你瞭解 Linux 的檔案屬性，那麼有個重要的也是常用的指令就必須要先跟你說囉！哪一個？就是『ls』這一個察看檔案的指令囉！在你以 root 的身份登入 Linux 之後，下達『ls -al』看看，會看到底下的幾個咚咚：

```
[root@www ~]# ls -al
total 156
drwxr-x---  4   root   root     4096   Sep  8 14:06  .
drwxr-xr-x 23   root   root     4096   Sep  8 14:21  ..
-rw-----  1   root   root     1474   Sep  4 18:27  anaconda-ks.cfg
-rw-----  1   root   root      199   Sep  8 17:14  .bash_history
-rw-r--r--  1   root   root      24    Jan  6 2007  .bash_logout
-rw-r--r--  1   root   root     191   Jan  6 2007  .bash_profile
-rw-r--r--  1   root   root     176   Jan  6 2007  .bashrc
-rw-r--r--  1   root   root     100   Jan  6 2007  .cshrc
drwx-----  3   root   root     4096   Sep  5 10:37  .gconf          <=範例說明處
drwx-----  2   root   root     4096   Sep  5 14:09  .gconfd
-rw-r--r--  1   root   root    42304   Sep  4 18:26  install.log    <=範例說明處
-rw-r--r--  1   root   root     5661   Sep  4 18:25  install.log.syslog
[  1  ][ 2  ][ 3  ][ 4  ][ 5  ][ 6  ][ 7  ]
[ 權限 ][連結][擁有者][群組][檔案容量][ 修改日期 ][ 檔名 ]
```



由於本章後續的 `chgrp`, `chown` 等指令可能都需要使用 `root` 的身份才能夠處理，所以這裡建議你以 `root` 的身份登入 Linux 來學習本章。

`ls` 是『list』的意思，重點在顯示檔案的檔名與相關屬性。而選項『-al』則表示列出所有的檔案詳細的權限與屬性(包含隱藏檔，就是檔名第一個字元為『.』的檔案)。如上所示，在

你第一次以 root 身份登入 Linux 時，如果你輸入上述指令後，應該有上列的幾個東西，先解釋一下上面七個欄位個別的意思：



圖 6.2.1.1 檔案屬性的示意圖

◆ 第一欄代表這個檔案的類型與權限(permission)：

這個地方最需要注意了！仔細看的話，你應該可以發現這一欄其實共有十個字元：(圖 6.2.1.1 及圖 6.2.1.2 內的權限並無關係)

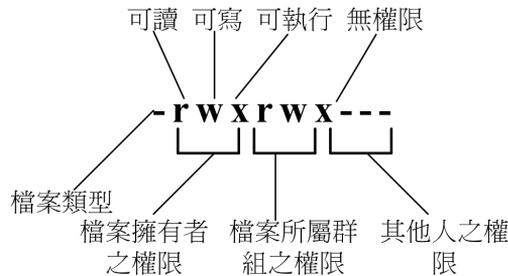


圖 6.2.1.2 檔案的類型與權限之內容

- 第一個字元代表這個檔案是『目錄、檔案或連結檔等等』：
 - 當為[d]則是目錄，例如上表檔名為『.gconf』的那一行；
 - 當為[-]則是檔案，例如上表檔名為『install.log』那一行；
 - 若是[l]則表示為連結檔(link file)；
 - 若是[b]則表示為裝置檔裡面的可供儲存的周邊設備(可隨機存取裝置)；
 - 若是[c]則表示為裝置檔裡面的序列埠設備，例如鍵盤、滑鼠(一次性讀取裝置)。
- 接下來的字元中，以三個為一組，且均為『rwx』的三個參數的組合。其中，[r]代表可讀(read)、[w]代表可寫(write)、[x]代表可執行(execute)。要注意的是，這三個權限的位置不會改變，如果沒有權限，就會出現減號[-]而已。

- 第一組為『檔案擁有者的權限』，以『install.log』那個檔案為例，該檔案的擁有者可以讀寫，但不可執行；
- 第二組為『同群組的權限』；
- 第三組為『其他非本群組的權限』。

例題

若有一個檔案的類型與權限資料為『-rwxr-xr--』，請說明其意義為何？

答：先將整個類型與權限資料分開查閱，並將十個字元整理成為如下所示：

```
[-][rwx][r-x][r--]
 1 234 567 890
```

1 為：代表這個檔名為目錄或檔案，本例中為檔案(-)；

234 為：擁有者的權限，本例中為可讀、可寫、可執行(rwx)；

567 為：同群組使用者權限，本例中為可讀可執行(rx)；

890 為：其他使用者權限，本例中為可讀(r)

同時注意到，rwx 所在的位置是不會改變的，有該權限就會顯示字元，沒有該權限就變成減號(-)就是了。

另外，目錄與檔案的權限意義並不相同，這是因為目錄與檔案所記錄的資料內容不相同所致。由於目錄與檔案的權限意義非常的重要，所以鳥哥將它獨立到 6.2.3 小節的目錄與檔案之權限意義中再來談。

◆ 第二欄表示有多少檔名連結到此節點(i-node)：

每個檔案都會將它的權限與屬性記錄到檔案系統的 i-node 中，不過，我們使用的目錄樹卻是使用檔名來記錄，因此每個檔名就會連結到一個 i-node 囉！這個屬性記錄的，就是有多少不同的檔名連結到相同的一個 i-node 號碼去就是了。關於 i-node 的相關資料我們會在第八章談到檔案系統時再加強介紹的。

◆ 第三欄表示這個檔案(或目錄)的『擁有者帳號』

◆ 第四欄表示這個檔案的所屬群組

在 Linux 系統下，你的帳號會附屬於一個或多個的群組中。舉剛剛我們提到的例子，class1, class2, class3 均屬於 projecta 這個群組，假設某個檔案所屬的群組為 projecta，且該檔案的權限如圖 6.2.1.2 所示(-rwxrwx---)，則 class1, class2, class3 三人對於該檔案都具有可讀、可寫、可執行的權限(看群組權限)。但如果是不屬於 projecta 的其他帳號，對於此檔案就不具有任何權限了。

- ◆ 第五欄為這個檔案的容量大小，預設單位為 bytes ；
- ◆ 第六欄為這個檔案的建檔日期或者是最近的修改日期 ；

這一欄的內容分別為日期(月/日)及時間。如果這個檔案被修改的時間距離現在太久了，那麼時間部分會僅顯示年份而已。如下所示：

```
[root@www ~]# ls -l /etc/termcap /root/install.log
-rw-r--r-- 1 root root 807103 Jan 7 2007 /etc/termcap
-rw-r--r-- 1 root root 42304 Sep 4 18:26 /root/install.log
# 如上所示，/etc/termcap 為 2007 年所修改過的檔案，離現在太遠之故；
# 至於 install.log 是今年（2009）所建立的，所以就顯示完整的時間了。
```

如果想要顯示完整的時間格式，可以利用 ls 的選項，亦即：『ls -l --full-time』就能夠顯示出完整的時間格式了！包括年、月、日、時間喔。另外，如果你當初是以繁體中文安裝你的 Linux 系統，那麼日期欄位將會以中文來顯示。可惜的是，中文並沒有辦法在純文字的終端機模式中正確的顯示，所以此欄會變成亂碼。那麼你就得使用『LANG=en_US』來修改語系囉！

如果想要讓系統預設的語系變成英文的話，那麼你可以修改系統設定檔『/etc/sysconfig/i18n』，利用第 21 頁談到的 nano 來修改該檔案的內容，使 LANG 這個變數成為上述的內容即可。

- ◆ 第七欄為這個檔案的檔名

這個欄位就是檔名了。比較特殊的是：如果檔名之前多一個『.』，則代表這個檔案為『隱藏檔』，例如上表中的.gconf那一行，該檔案就是隱藏檔。你可以使用『ls』及『ls -a』這兩個指令去感受一下什麼是隱藏檔囉！



對於更詳細的 ls 用法，還記得怎麼查詢嗎？對啦！使用 man ls 或 info ls 去看看它的基礎用法去！自我進修是很重要的，因為『師傅帶進門，修行看個人！』，自古只有天才學生，沒有天才老師嘞！加油吧！^_^

這七個欄位的意義是很重要的！務必清楚的知道各個欄位代表的意義！尤其是第一個欄位的各個權限，那是整個 Linux 檔案權限的重點之一。接下來我們來做幾個簡單的練習，你就會比較清楚囉！

例題

假設 test1, test2, test3 同屬於 testgroup 這個群組，如果有下面的兩個檔案，請說明兩個檔案的擁有者與其相關的權限為何？

```
-rw-r--r--  1 root      root          238 Jun 18 17:22 test.txt
-rwxr-xr--  1 test1    testgroup   5238 Jun 19 10:25 ping_tsai
```

- 答：◆ 檔案 test.txt 的擁有者為 root，所屬群組為 root。至於權限方面則只有 root 這個帳號可以存取此檔案，其他人則僅能讀此檔案；
- ◆ 另一個檔案 ping_tsai 的擁有者為 test1，而所屬群組為 testgroup。其中：
 - test1 可以針對此檔案具有可讀可寫可執行的權力；
 - 而同群組的 test2, test3 兩個人與 test1 同樣是 testgroup 的群組帳號，則僅可讀可執行但不能寫(亦即不能修改)；
 - 至於非 testgroup 這一個群組的人則僅可以讀，不能寫也不能執行！

例題

如果我的目錄為底下的樣式，請問 testgroup 這個群組的成員與其他人(others)是否可以進入本目錄？

```
drwxr-xr--  1 test1    testgroup   5238 Jun 19 10:25 groups/
```

- 答：◆ 檔案擁有者 test1[rwx]可以在本目錄中進行任何工作；
- ◆ 而 testgroup 這個群組[r-x]的帳號，例如 test2, test3 亦可以進入本目錄進行工作，但是不能在本目錄下進行寫入的動作；
 - ◆ 至於 other 的權限中[r-]雖然有 r，但是由於沒有 x 的權限，因此 others 的使用者，並不能進入此目錄！

◆ **Linux 檔案權限的重要性：**

與 Windows 系統不一樣的是，在 Linux 系統當中，每一個檔案都多加了很多的屬性進來，尤其是群組的概念，這樣有什麼用途呢？其實，最大的用途是在『資料安全性』上面的。

■ **系統保護的功能：**

舉個簡單的例子，在你的系統中，關於系統服務的檔案通常只有 root 才能讀寫或者是執行，例如/etc/shadow 這一個帳號管理的檔案，由於該檔案記錄了你系統中所有帳號的資料，因此是很重要的一個設定檔，當然不能讓任何人讀取(否則密碼會被竊取啊)，只有 root 才能夠來讀取囉！所以該檔案的權限就會成為[-rw-----]囉！

■ 團隊開發軟體或資料共用的功能：

此外，如果你有一個軟體開發團隊，在你的團隊中，你希望每個人都可以使用某一些目錄下的檔案，而非你的團隊的其他人則不予以開放呢？以上面的例子來說，testgroup 的團隊共有三個人，分別是 test1, test2, test3，那麼我就可以將團隊所需的檔案權限訂為[-rwxrwx---]來提供給 testgroup 的工作團隊使用囉！

■ 未將權限設定妥當的危害：

再舉個例子來說，如果你的目錄權限沒有作好的話，可能造成其他人都可以在你的系統上面亂搞囉！例如本來只有 root 才能做的開關機、ADSL 的撥接程式、新增或刪除使用者等等的指令，若被你改成任何人都可以執行的話，那麼如果使用者不小心給你重新開機啦！重新撥接啦！等等的！那麼你的系統不就會常常莫名其妙的掛掉囉！而且萬一你的使用者的密碼被其他不明人士取得的話，只要他登入你的系統就可以輕而易舉的執行一些 root 的工作！

可怕吧！因此，在**你修改你的 linux 檔案與目錄的屬性之前，一定要先搞清楚，什麼資料是可變的，什麼是不可變的！**千萬注意囉！接下來我們來處理一下檔案屬性與權限的變更吧！

6.2.2 如何改變檔案屬性與權限

我們現在知道檔案權限對於一個系統的安全重要性了，也知道檔案的權限對於使用者與群組的相關性，那麼如何修改一個檔案的屬性與權限呢？又！有多少檔案的權限我們可以修改呢？其實一個檔案的屬性與權限有很多！我們先介紹幾個常用於群組、擁有者、各種身份的權限之修改的指令，如下所示：

- chgrp：改變檔案所屬群組
- chown：改變檔案擁有者
- chmod：改變檔案的權限, SUID, SGID, SBIT 等等的特性

◆ 改變所屬群組, chgrp

改變一個檔案的群組真是很簡單的，直接以 chgrp 來改變即可，噢！這個指令就是 change group 的縮寫嘛！這樣就很好記了吧！^_^。不過，請記得，要被改變的群組名稱必須要在/etc/group 檔案內存在才行，否則就會顯示錯誤！

假設你是以 root 的身份登入 Linux 系統的，那麼在你的家目錄內有一個 install.log 的檔案，如何將該檔案的群組改變一下呢？假設你已經知道在/etc/group 裡面已經存在一個名為

users 的群組，但是 testing 這個群組名字就不存在/etc/group 當中了，此時改變群組成為 users 與 testing 分別會有什麼現象發生呢？

```
[root@www ~]# chgrp [-R] dirname/filename ...
選項與參數：
-R : 進行遞迴(recursive)的持續變更，亦即連同次目錄下的所有檔案、目錄
      都更新成爲這個群組之意。常常用在變更某一目錄內所有的檔案之情況。
範例：
[root@www ~]# chgrp users install.log
[root@www ~]# ls -l
-rw-r--r--  1 root users 68495 Jun 25 08:53 install.log
[root@www ~]# chgrp testing install.log
chgrp: invalid group name `testing' <== 發生錯誤訊息囉~找不到這個群組名~
```

發現了嗎？檔案的群組被改成 users 了，但是要改成 testing 的時候，就會發生錯誤～注意喔！發生錯誤訊息還是要努力的查一查錯誤訊息的內容才好！將它英文翻譯成爲中文，就知道問題出在哪裡了。

◆ 改變檔案擁有者, chown

如何改變一個檔案的擁有者呢？很簡單呀！既然改變群組是 change group，那麼改變擁有者就是 change owner 囉！BINGO！那就是 chown 這個指令的用途，要注意的是，使用者必須是已經存在系統中的帳號，也就是在/etc/passwd 這個檔案中有記錄的使用者名稱才能改變。

chown 的用途還滿多的，它還可以順便直接修改群組的名稱呢！此外，**如果要連目錄下的所有次目錄或檔案同時更改檔案擁有者的話，直接加上 -R 的選項即可！**我們來看看語法與範例：

```
[root@www ~]# chown [-R] 帳號名稱 檔案或目錄
[root@www ~]# chown [-R] 帳號名稱:群組名稱 檔案或目錄
選項與參數：
-R : 進行遞迴(recursive)的持續變更，亦即連同次目錄下的所有檔案都變更
範例：將 install.log 的擁有者改爲 bin 這個帳號：
[root@www ~]# chown bin install.log
[root@www ~]# ls -l
-rw-r--r--  1 bin users 68495 Jun 25 08:53 install.log
範例：將 install.log 的擁有者與群組改回爲 root：
[root@www ~]# chown root:root install.log
[root@www ~]# ls -l
-rw-r--r--  1 root root 68495 Jun 25 08:53 install.log
```



事實上，chown 也可以使用『chown user.group file』，亦即在擁有者與群組間加上小數點『.』也行！不過很多朋友設定帳號時，喜歡在帳號當中加入小數點(例如 vbird.tsai 這樣的帳號格式)，這就會造成系統的誤判了！所以我們比較建議使用冒號『:』來隔開擁有者與群組啦！此外，chown 也能單純的修改所屬群組呢！例如『chown .sshd install.log』就是修改群組～看到了嗎？就是那個小數點的用途！

知道如何改變檔案的群組與擁有者了，那什麼時候要使用 chown 或 chgrp 呢？或許你會覺得奇怪吧？是的，確實有時候需要變更檔案的擁有者的，最常見的例子就是在複製檔案給你之外的其他人時，我們使用最簡單的 cp 指令來說明好了：

```
[root@www ~]# cp 來源檔案 目的檔案
```

假設你今天要將 .bashrc 這個檔案拷貝成 .bashrc_test 檔名，且是要給 bin 這個人，你可以這樣做：

```
[root@www ~]# cp .bashrc .bashrc_test
[root@www ~]# ls -al .bashrc*
-rw-r--r--  1 root root 395 Jul  4 11:45 .bashrc
-rw-r--r--  1 root root 395 Jul 13 11:31 .bashrc_test <==新檔案的屬性沒變
```

由於複製行為(cp)會複製執行者的屬性與權限，所以！怎麼辦？.bashrc_test 還是屬於 root 所擁有，如此一來，即使你將檔案拿給 bin 這個使用者了，那他也無法修改的(看屬性/權限就知道了吧)，所以你就必須要把這個檔案的擁有者與群組修改一下囉！知道如何修改了吧？

◆ 改變權限, chmod

檔案權限的改變使用的是 chmod 這個指令，但是，權限的設定方法有兩種，分別可以使用數字或者是符號來進行權限的變更。我們就來談一談：

■ 數字類型改變檔案權限

Linux 檔案的基本權限就有 3 個，分別是 owner/group/others 三種身份各有自己的 read/write/execute 權限，先複習一下剛剛上面提到的資料：檔案的權限字元為：『-rwxrwxrwx』，這 3 個權限是三個三個一組的！其中，我們可以使用數字來代表各個權限，各權限的數字對照表如下：

```
r:4
w:2
x:1
```

每種身份(owner/group/others)各自的三個權限(r/w/x)分數是需要累加的，例如當權限為：[-rwxrwx---] 分數則是：

```
owner = rwx = 4+2+1 = 7
group = rwx = 4+2+1 = 7
others= --- = 0+0+0 = 0
```

所以等一下我們設定權限的變更時，該檔案的權限數字就是 770 啦！變更權限的指令 chmod 的語法是這樣的：

```
[root@www ~]# chmod [-R] xyz 檔案或目錄
```

選項與參數：

xyz：就是剛剛提到的數字類型的權限屬性，為 rwx 屬性數值的相加。

-R：進行遞迴(recursive)的持續變更，亦即連同次目錄下的所有檔案都會變更

舉例來說，如果要將.bashrc 這個檔案所有的權限都設定啟用，那麼就下達：

```
[root@www ~]# ls -al .bashrc
-rw-r--r-- 1 root root 395 Jul  4 11:45 .bashrc
[root@www ~]# chmod 777 .bashrc
[root@www ~]# ls -al .bashrc
-rwxrwxrwx 1 root root 395 Jul  4 11:45 .bashrc
```

那如果要將權限變成『-rwxr-xr--』呢？那麼權限的分數就成為 [4+2+1][4+0+1][4+0+0]=754 囉！所以你需要下達『chmod 754 filename』。另外，在實際的系統運作中最常發生的一個問題就是，常常我們以 vim 編輯一個 shell 的文字批次檔後，它的權限通常是 -rw-rw-r-- 也就是 644，如果要將該檔案變成可執行檔，並且不要讓其他人修改此一檔案的話，那麼就需要-rwxr-xr-x 這樣的權限，此時就得要下達：『chmod 755 test.sh』的指令囉！

另外，如果有些檔案你不希望被其他人看到，那麼應該將檔案的權限設定為例如：『-rwxr-----』，那就下達『chmod 740 filename』吧！

例題

將剛剛你的.bashrc 這個檔案的權限修改回-rw-r--r--的情況吧！

答：-rw-r--r--的分數是 644，所以指令為：

```
chmod 644 .bashrc
```

■ 符號類型改變檔案權限

還有一個改變權限的方法呦！從之前的介紹中我們可以發現，基本上就九個權限分別是(1)user (2)group (3)others 三種身份啦！那麼我們就可以藉由 **u, g, o** 來代表三種身份的權限！此外，**a** 則代表 all 亦即全部的身份！那麼讀寫的權限就可以寫成 **r, w, x** 囉！也就是可以使用底下的方式來看：

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

來實作一下吧！假如我們要『設定』一個檔案的權限成為『-rwxr-xr-x』時，基本上就是：

- user (u)：具有可讀、可寫、可執行的權限；
- group 與 others (g/o)：具有可讀與執行的權限。

所以就是：

```
[root@www ~]# chmod u=rwx,go=rx .bashrc
# 注意喔！那個 u=rwx,go=rx 是連在一起的，中間並沒有任何空白字元！
[root@www ~]# ls -al .bashrc
-rwxr-xr-x 1 root root 395 Jul  4 11:45 .bashrc
```

那麼假如是『-rwxr-xr--』這樣的權限呢？可以使用『**chmod u=rwx,g=rx,o=r filename**』來設定。此外，如果我不知道原先的檔案屬性，而我只想要增加.bashrc 這個檔案的每個人均可寫入的權限，那麼我就可以使用：

```
[root@www ~]# ls -al .bashrc
-rwxr-xr-x 1 root root 395 Jul  4 11:45 .bashrc
[root@www ~]# chmod a+w .bashrc
[root@www ~]# ls -al .bashrc
-rwxrwxrwx 1 root root 395 Jul  4 11:45 .bashrc
```

而如果是要將權限去掉而不更動其他已存在的權限呢？例如要拿掉全部人的可執行權限，則：

```
[root@www ~]# chmod a-x .bashrc
[root@www ~]# ls -al .bashrc
-rw-rw-rw- 1 root root 395 Jul  4 11:45 .bashrc
```

知道 +, -, = 的不同點了嗎？對啦！+ 與 - 的狀態下，只要是沒有指定到的項目，則該權限『不會被變動』，例如上面的例子中，由於僅以 - 拿掉 x 則其他兩個保持當時的值不變！多多實作一下，你就會知道如何改變權限囉！這在某些情況底下很好用的～舉例來說，你想要教一個朋友如何讓一個程式可以擁有執行的權限，但你又不知道該檔案原本的權限為何，此時，利用『`chmod a+x filename`』，就可以讓該程式擁有執行的權限了。是否很方便？

6.2.3 目錄與檔案之權限意義

現在我們知道了 Linux 系統內檔案的三種身份(擁有者、群組與其他人)，知道每種身份都有三種權限(rwx)，已知道能夠使用 `chown`, `chgrp`, `chmod` 去修改這些權限與屬性，當然，利用 `ls -l` 去觀察檔案也沒問題。前兩小節也談到了這些檔案權限對於資料安全的重要性。那麼，這些檔案權限對於一般檔案與目錄檔案有何不同呢？有太大的不同啊！底下就讓烏哥來說清楚，講明白！

◆ 權限對檔案的重要性

檔案是實際含有資料的地方，包括一般文字檔、資料庫內容檔、二進位可執行檔(binary program)等等。因此，權限對於檔案來說，它的意義是這樣的：

- **r (read)**：可讀取此一檔案的實際內容，如讀取文字檔的文字內容等；
- **w (write)**：可以編輯、新增或者是修改該檔案的內容(但不含刪除該檔案)；
- **x (eXecute)**：該檔案具有可以被系統執行的權限。

那個可讀(r)代表讀取檔案內容是還好瞭解，那麼可執行(x)呢？這裡你就必須要小心啦！因為在 Windows 底下一個檔案是否具有執行的能力是藉由『**副檔名**』來判斷的，例如：`.exe`, `.bat`, `.com` 等等，但是在 Linux 底下，**我們的檔案是否能被執行，則是藉由是否具有『x』這個權限來決定的！跟檔名是沒有絕對的關係的！**

至於最後一個 w 這個權限呢？當你對一個檔案具有 w 權限時，你可以具有寫入/編輯/新增/修改檔案的內容的權限，**但並不具備有刪除該檔案本身的權限！**對於檔案的 rwx 來說，主要都是針對『檔案的內容』而言，與檔案檔名的存在與否沒有關係喔！因為檔案記錄的是實際的資料嘛！

◆ 權限對目錄的重要性

檔案是存放實際資料的所在，那麼目錄主要是儲存啥玩意啊？**目錄主要的內容在記錄檔名清單，檔名與目錄有強烈的關連啦！**所以如果是針對目錄時，那個 r, w, x 對目錄是什麼意義呢？

- **r (read contents in directory) :**

表示具有讀取目錄結構清單的權限，所以當你具有讀取(r)一個目錄的權限時，表示你可以查詢該目錄下的檔名資料。所以你就可以利用 `ls` 這個指令將該目錄的內容列表顯示出來！

- **w (modify contents of directory) :**

這個可寫入的權限對目錄來說，是很了不起的！**因為它表示你具有異動該目錄結構清單的權限**，也就是底下這些權限：

- **建立新的檔案與目錄；**
- **刪除已經存在的檔案與目錄(不論該檔案的權限為何！)**
- **將已存在的檔案或目錄進行更名；**
- **搬移該目錄內的檔案、目錄位置。**

總之，目錄的 `w` 權限就與該目錄底下的檔名異動有關就對了啦！

- **x (access directory) :**

咦！目錄的執行權限有啥用途啊？目錄只是記錄檔名而已，總不能拿來執行吧？沒錯！目錄不可以被執行，**目錄的 `x` 代表的是使用者能否進入該目錄成為工作目錄的用途**！所謂的工作目錄(work directory)就是你目前所在的目錄啦！舉例來說，當你登入 Linux 時，你所在的家目錄就是你當下的工作目錄。而變換目錄的指令是『`cd`』(change directory)囉！

大致的目錄權限概念是這樣，底下我們來看幾個範例，讓你瞭解一下啥是目錄的權限囉！

例題

有個目錄的權限如下所示：

```
drwxr--r-- 3 root root 4096 Jun 25 08:35 .ssh
```

系統有個帳號名稱為 `vbird`，這個帳號並沒有支援 `root` 群組，請問 `vbird` 對這個目錄有何權限？是否可切換到此目錄中？

答：`vbird` 對此目錄僅具有 `r` 的權限，因此 `vbird` 可以查詢此目錄下的檔名列表。因為 `vbird` 不具有 `x` 的權限，因此 `vbird` 並不能切換到此目錄內！(相當重要的概念！)

上面這個例題中因為 `vbird` 具有 `r` 的權限，因為是 `r` 乍看之下好像就具有可以進入此目錄的權限，其實那是錯的。能不能進入某一個目錄，只與該目錄的 `x` 權限有關啦！此外，工作目錄對於指令的執行是非常重要的，**如果你在某目錄下不具有 `x` 的權限，那麼你就無法切換到該目錄下，也就無法執行該目錄下的任何指令，即使你具有該目錄的 `r` 的權限。**

很多朋友在架設網站的時候都會卡在一些權限的設定上，他們開放目錄資料給網際網路的任何人來瀏覽，卻只開放 r 的權限，如上面的範例所示那樣，那樣的結果就是導致網站伺服器軟體無法到該目錄下讀取檔案(最多只能看到檔名)，最終用戶總是無法正確的查閱到檔案的內容(顯示權限不足啊!)。要注意：**要開放目錄給任何人瀏覽時，應該至少也要給予 r 及 x 的權限，但 w 權限不可隨便給！**為什麼 w 不能隨便給，我們來看下一個例子：

例題

假設有個帳號名為 dmtsai，他的家目錄在/home/dmtsai/，dmtsai 對此目錄具有[rwx]的權限。若在此目錄下有個名為 the_root.data 的檔案，該檔案的權限如下：

```
-rwx----- 1 root root 4365 Sep 19 23:20 the_root.data
```

請問 dmtsai 對此檔案的權限為何？可否刪除此檔案？

答：如上所示，由於 dmtsai 對此檔案來說是『others』的身份，因此這個檔案他無法讀、無法編輯也無法執行，也就是說，他無法變動這個檔案的內容就是了。

但是由於這個檔案在他的家目錄下，他在此目錄下具有 rwx 的完整權限，因此對於 the_root.data 這個『檔名』來說，他是能夠『刪除』的！結論就是，dmtsai 這個用戶能夠刪除 the_root.data 這個檔案！

還是看不太懂？有聽沒有懂喔！沒關係～我們底下就來設計一個練習，讓你實際玩玩看，應該就能夠比較近入狀況啦！不過，由於很多指令我們還沒有教，所以底下的指令有的先瞭解即可，詳細的指令用法我們會在後面繼續介紹的。

◆ 先用 root 的身份建立所需要的檔案與目錄環境

我們用 root 的身份在所有人都可以工作的/tmp 目錄中建立一個名為 testing 的目錄，該目錄的權限為 744 且目錄擁有者為 root。另外，在 testing 目錄下在建立一個空的檔案，檔名亦為 testing。建立目錄可用 mkdir(make directory)，建立空檔案可用 touch(下一章會說明)來處理。所以過程如下所示：

```
[root@www ~]# cd /tmp <==切換工作目錄到/tmp
[root@www tmp]# mkdir testing <==建立新目錄
[root@www tmp]# chmod 744 testing <==變更權限
[root@www tmp]# touch testing/testing <==建立空的檔案
[root@www tmp]# chmod 600 testing/testing <==變更權限
[root@www tmp]# ls -ald testing testing/testing
drwxr--r-- 2 root root 4096 Sep 19 16:01 testing
-rw----- 1 root root 0 Sep 19 16:01 testing/testing
# 仔細看一下，目錄的權限是 744，且所屬群組與使用者均是 root 喔！
# 那麼在這樣的情況底下，一般身份使用者對這個目錄/檔案的權限為何？
```