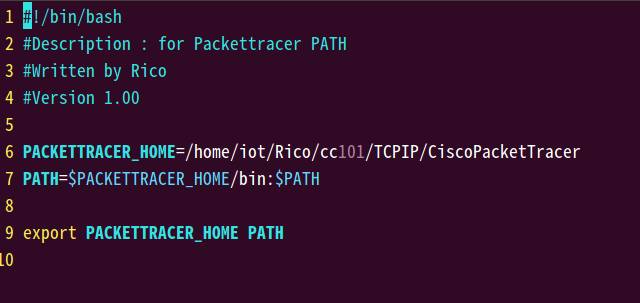
1-1-4 關於type command 用何種方式判別為內外部指令

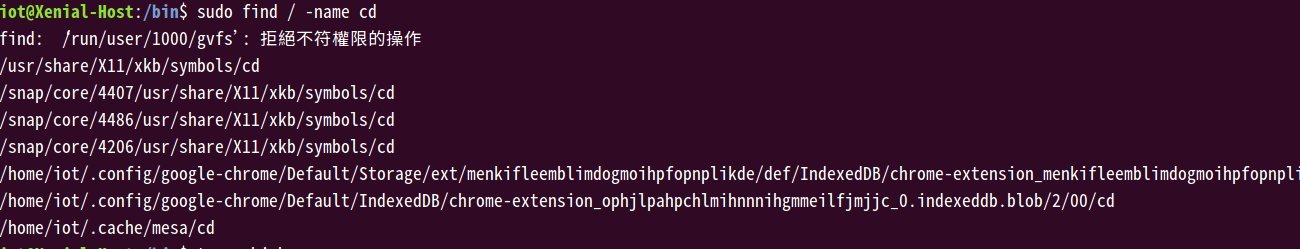
type這個指令本質上為尋找執行檔在哪裡的指令並且再加以分析類型。

比較特別的是keyword的time與為內部指令的kill，他們本身在bash的執行檔裡面外還是有另外獨自的執行檔。而shell keyword不在目前的討論範圍內，故不多闡述。

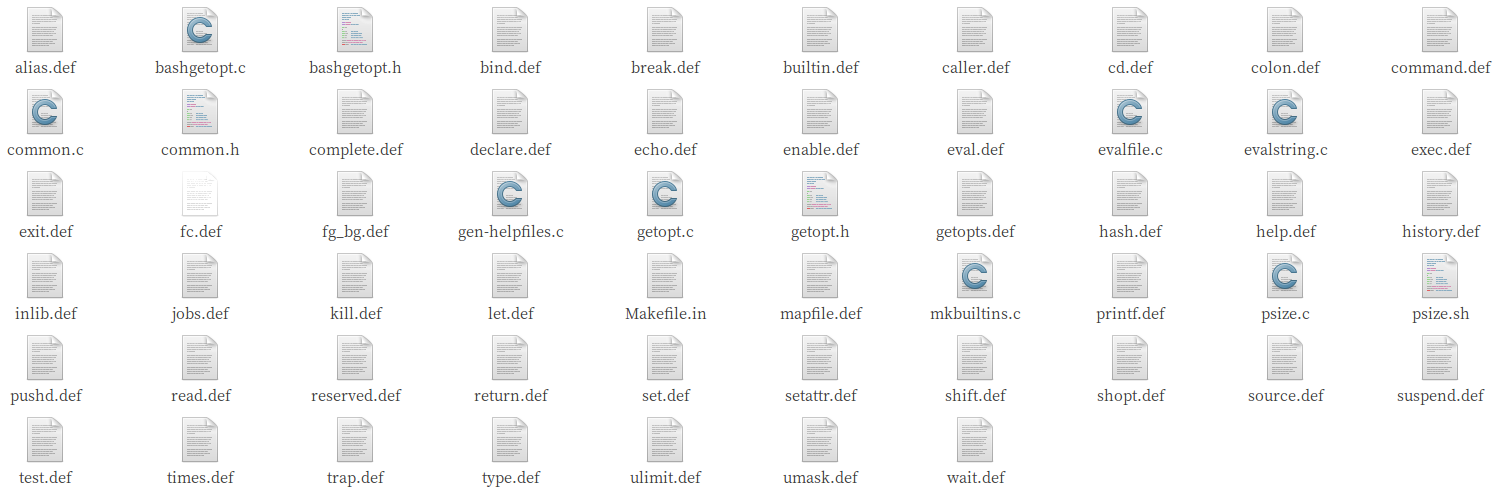
而cat為外部的指令，而type究竟是如何去做尋找，為了驗證我做了個小實驗，從網路上下載PacketTracer7並且把路徑加入環境變數中，自己寫了個shell script為/etc/profile.d/PacketTracer7.sh 讓$PATH有對應的路徑中做驗證。由此可以得知type的本質為何。



內部指令基本上都已經包含在/bin/bash這個執行檔裡面，於是做一個簡單的小驗證，用指令find / -name cd來找cd這個內部指令看看有什麼結果

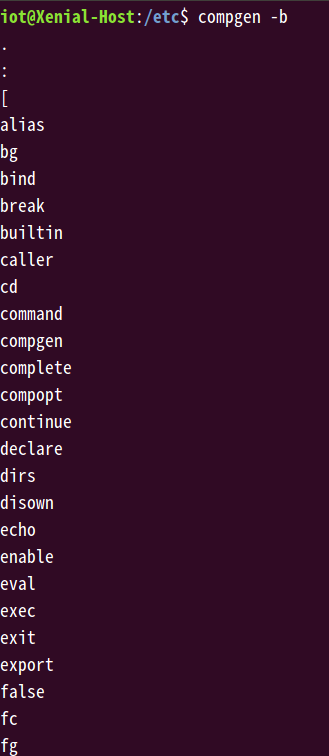
顯示的結果那寫cd檔案都不是執行檔，且也不在$PATH的路徑下

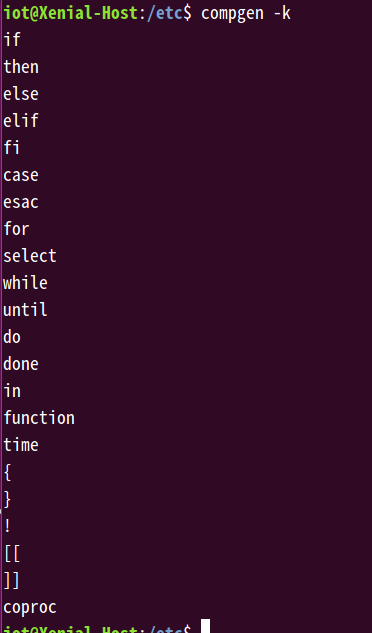
而為了驗證bash這程式裡面究竟有沒有我們常用的內部指令，在網路上找了bash的原始碼(Version 4.4)在builtins的資料夾裡有一系列的指令def檔



補充：

其實用compgen這個自動補齊的程式也可以達到同樣的效果 -b for 內部指令 -k for keyword





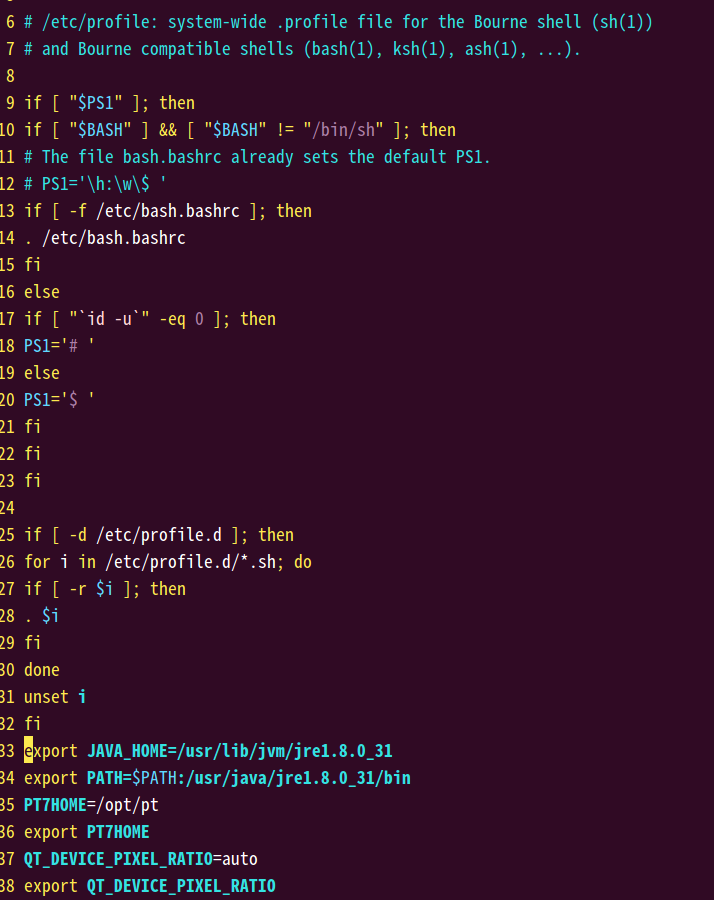
1-1-5 請解釋Login shell其中一個組態內容

一開始會先去讀/etc/profile這個檔案的內容，由此可發現他會先判斷使用者是用Login shell還是non-Login shell的方式登入，如果是non-Login shell則會被指引到/etc/bash.bashrc的檔案去

還有命令提示字元 PS1的設定

如果為Login shell的方式，他會引庫其他地方的shell script，來做更完整環境配置，分拆功能上的組態檔的確真的有助於troubleshooting。

而最後為建置PATH的內容，為了讓執行檔方便執行而設。



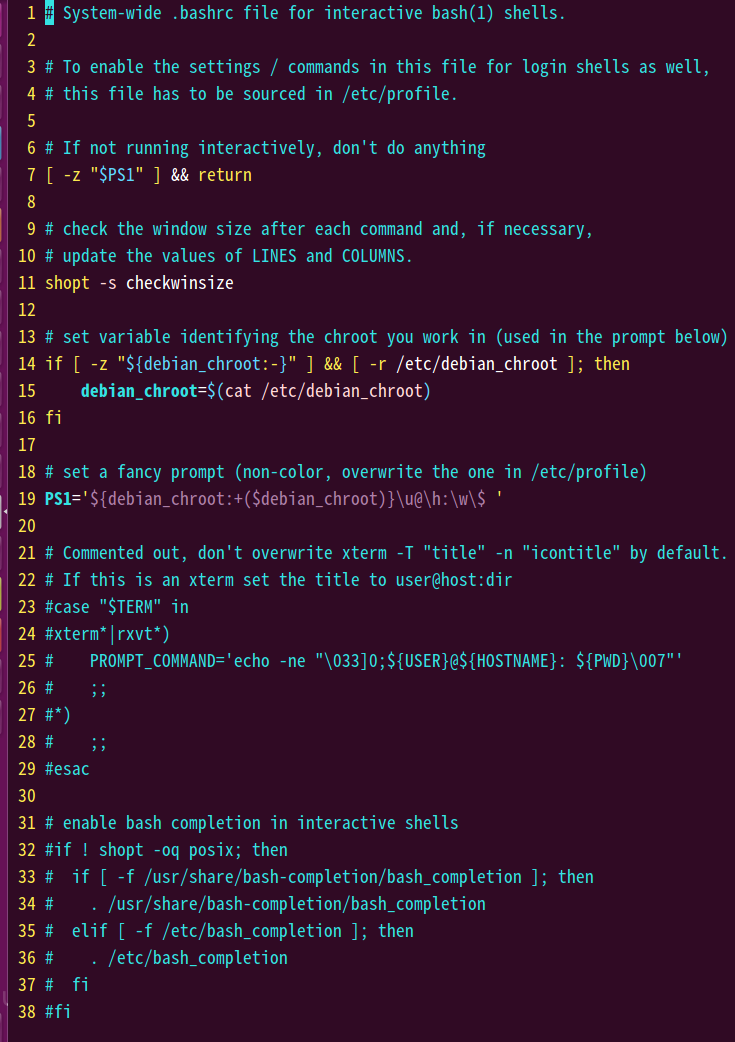
1-1-6 請解釋non-Login shell其中一個組態檔內容

此為/etc/bash.bashrc的檔案，一開始如果使用者登入為非互動式的話會做出跳出的動作。

會注意使用者終端機的行數與字數一次顯示多少的控制

如果登入的身份是否為root，判定方式為登入身份對於/etc/debian\_chroot的可讀權限

給予提示字元



關於使用sudo的驗證方式，會先看家目錄裡是否有無sudo相關的憑證後在看是否為admin或sudo的群組，如果有就啟動sudo這個程式

關於指令找不到的處理方式，會先執行command-not-found這個程式，這個程式會去找command-not-found這個程式本身的函式，如果都沒有就直接顯示command-not-found這個字串 。



參考資料

鳥哥 BASH學習

<http://linux.vbird.org/linux_basic/0320bash.php>

stackexchange question about type

<https://unix.stackexchange.com/questions/101599/bash-what-is-the-use-of-type-bash-builtins?utm_medium=organic&utm_source=google_rich_qa&utm_campaign=google_rich_qa>

牛牛網站 GUN.org Bash reference manual

[http://www.gnu.org/software/bash/manual/bashref.html#Comments](http://www.gnu.org/software/bash/manual/bashref.html" \l "Comments)

牛牛網站 Bash open source

<http://ftp.gnu.org/gnu/bash/>