

Linux 常用命令

编辑人：董炳江

以下只说明各指令的基本用法，若需详细说明，请用 `man` 去读详细的 `manual`。

1. 关于文件/目录处理的指令：

1.1 `ls`

这是最基本的文件指令。`ls` 的意义为 “list”，也就是将某一个目录下的内容显示出来。如果你在 `ls` 指令后没有跟任何文件，它将会显示出目前目录中所有文件。也可以在 `ls` 后面加上所要查看的目录名称或文件的名称，如

```
$ls/home/
```

```
$lsfoo
```

`ls` 有一些参数，可以给予使用者更多有关的资讯，如下：

-a: 在 `linux` 中若一个目录或文件名字的第一个字元为 “.”，则使用 `ls` 将不会显示出这个文件的名字，我们称此类文件为隐藏文件。如 `tcsh` 的初设档。`tcshrc`；如果我们要察看这类文件，则必须加上参数 “-a”。

-l: 这个参数代表使用 `ls` 的长格式，可以显示更多的资讯，如文件存取权，文件拥有者 (owner)，文件大小，文件最后更新日期，甚而 `symboliclink` 的文件是 `link` 哪一个文件等等。如下：

```
$ls-l
```

```
drwx--x--x2jtseng512Aug805: 0818
```

```
drwx--x--x2jtseng512Aug822: 0019
```

```
-rw-----1jtseng566Aug805: 28makefile
```

1.2 `cp`

`cp` 这个指令的意义是复制 (“COPY”)，也就是将一个或多个文件复制成另一个文件或将其复制到另一个目录去。

`cp` 的用法如下：

`cpf1f2`: 将文件名为 `f1` 的文件复制一份为文件名为 `f2` 的文件。

`cpf1f2f3...dir`: 将文件 `f1f2f3...` 都以相同的文件名复制一份放到目录 `dir` 里面。

`cp-rdir1dir2`: 将 `dir1` 的全部内容全部复制到 `dir2` 里面。

`cp` 也有一些参数，如下：

-i: 此参数是当已有文件名为 `f2` 的文件时，若迳自使用 `cp` 将会将原来 `f2` 的内容掩盖过去，因此在要盖过之前必须先询问使用者一下。如使用者的回答是 `y(yes)`才执行复制的动作。

-r: 可将一整颗子树都复制到另一个目录中。

1.3 `mv`

`mv` 的意义为 `move`，主要是将一文件改名或换至另一个目录。如同 `cp`，它也有三种格式：

`mvf1f2`: 将文件名为 `f1` 的文件变更成文件名为 `f2` 的文件。

`mvdir1dir2`: 将文件名为 `dir1` 的目录变更成文件名为 `dir2` 的目录。

`mvf1f2f3...dir`: 将文件 `f1f2f3...` 都移至目录 `dir` 里面。

`mv` 的参数有两个, `-f` 和 `-i`, 其中 `-i` 的意义与 `cp` 中的相同, 均是 `interactive` 询问之意。而 `-f` 为强迫(force), 就是不管有没有同名的文件, 都进行操作, 所有其他的参数遇到 `-f` 均会失效。

1.4 rm

`rm` 的意义是 `remove`, 也就是用来删除一个文件的指令。在 `Linux` 中一个被删除的文件除非是系统恰好有做备份, 否则是无法像 `DOS` 里面一样还能够救回来的。所以在做 `rm` 动作的时候使用者应该要特别小心。

`rm` 的格式如下:

```
rm f1 f2 f3 ...
```

而 `rm` 的参数比较常用的有几个: `-f`, `-i`, 与 `-r`

`-f`: 将会使得系统在删除时, 不提出任何警告讯息。

`-i`: 在除去文件之前均会询问是否真要除去。

`-r`: 递归式的删除。

小心不要随便使用 `rm -r f`, 否则有一天你会“欲哭无泪”⊗

1.5 mkdir

`mkdir` 是一个让使用者建立一个目录的指令。你可以在一个目录底下使用 `mkdir` 建立一个子目录, 使用的方法如下:

```
mkdir dirname1[dirname2...]
```

如此你就可以建立一个或多个目录。

1.6 chdir(cd)

这是让使用者用来转移工作目录用的。

`chdir` 的用法如下:

```
chdir dirname
```

如此你就可以将目前的目录转移到 `dirname` 这一个目录去。

或使用“`chdir ..`”来转移到上一层目录。

1.7 rmdir

相对于 `mkdir`, `rmdir` 是用来将一个“空的”目录杀掉的。如果一个目录下面没有任何文件, 你就可以用 `rmdir` 指令将其除去。`rmdir` 的使用法如下:

```
rmdir dirname1[dirname2...]
```

如果一个目录底下有其他的文件, `rmdir` 将无法将这个目录杀掉, 除非使用 `rm` 指令的 `-r` 选项。

1.8 pwd

`pwd` 会将目前目录的路径(path)显示出来, 例如:

1.9 cat/more/less

以上三个指令均为察看文件内容的指令。`cat` 的意义是猫…? 不不不, 是 `concatenate`, 在字典上的意思是“连结, 将…串成锁状”(语出资工电子词典 `cdict`), 其实就是把文件的内容显示出来的意思。`cat` 有许多奇怪的参数, 较常为人所使用的是 `-n` 参数, 也就是把显示出来的内容加上行号。`cat` 的用法如下:

`cat [-n]`: 自标准输入读进内容, 你可以用 `pipe` 将别的程式的输出转向给 `cat`。

`cat [-n]filename`: 将 `filename` 的内容读进来, 显示在标准输出上。

问题在于 `cat` 它是不会停下来的, 因此并不好用(试想如果一个萤幕二十四行, 而一个文件四百行, `cat` 一出来将会劈哩啪啦不断的卷上去, 使用者很难据此得到他们所需的资讯。)

所以才有人又写了 `more` 出来。

`more`，跟据蔡文能老师的说法是“再多一点就好”，`more` 可以将所观察的文件跟据终端机的形态一页页的显示出来，再根据使用者的要求换页或卷行。如果使用者要在某一个文件中搜寻一个特定的字符串，则按/然后跟着打所要搜寻的单字即可进行搜寻。`more` 也可以找得到。`more` 的使用法如下：

```
more filename
```

如果你在使用中觉得已经看到了所要看的部份，可以按'q'离开 `more` 的使用。在使用中按'v'亦可以使用编辑器来编辑所观看的文件。

`less` 的用法与 `more` 极类似，原先它就是为了弥补 `more` 只能往前方卷页的缺点而设计。`less` 的用法如下：

```
less filename
```

其与 `more` 不同的是它可以按 y 来往上卷一行，并且可以用“?”来来回搜寻你所要找的单字。

1.10 chmod

`chmod` 为变更文件模式用(change mode)。这个指令是用来更改文件的存取模式(accessmode)。在 linux 一个文件上有可读(r)可写(w)可执行(x)三种模式，分别针对该文件的拥有者(owner)、同群者(groupmember)(你可以 `ls -lg` 来观看某一文件的所属的 group)，以及其他其他(other)。一个文件如果改成可执行模式则系统就将其视为一个可执行档，而一个目录的可执行模式代表使用者有进入该目录之权利。`chmod` 就是用来变更一些文件的模式，其使用方式如下：

```
chmod [-fR] modefilename...
```

其参数的意义如下：

-f Force。`chmod` 不会理会失败的动作。

-R Recurive。会将所有子树下的所有子目录及文件改为你所要改成的模式。

mode 可以为一个三位或四位的八进位数字，来表示对某些对象的存取权。详情可参阅 `chmod(1)` 的 manual page 中有关 Absolute Modes 的说明。或是用一个字符串来表示，请参考 `chmod(1)` 的说明。

2 关于进程处理的指令：

2.1 ps

`ps` 是用来显示目前你的进程或系统进程的状况。以下列出比较常用的参数，其选项说明如下：

-a 列出包括其他 users 的进程状况。

-u 显示 user-oriented 的进程状况。

-x 显示包括没有 terminal 控制的进程状况。

-w 使用较宽的显示模式来显示进程状况。

我们可以经由 `ps` 取得目前进程的状况，如 pid, running state 等。

2.2 kill

`kill` 指令的用途是送一个 signal 给某一个进程。因为大部份送的都是用来杀掉进程的 SIGKILL 或 SIGHUP，因此称为 kill。`kill` 的用法为：

```
kill [-SIGNAL] pid...
```

```
kill -l
```

SIGNAL 为一个 singal 的数字，从 0 到 31，其中 9 是 SIGKILL，也就是一般用来杀掉

一些无法正常 terminate 的讯号。其余讯号的用途可参考 sigvec(2)中对 signal 的说明。你也可以用 kill -l 来察看可代替 signal 号码的数目字。kill 的详细情形请参阅 mankill。

3 关于字串处理的指令：

3.1 echo

echo 是用来显示一字串在终端机上。echo -n 则是当显示完之后不会有跳行的动作。

3.2 grep/fgrep

grep 为一过滤器，它可自一个或多个文件中过滤出具有某个字串的行，或是自标准输入过滤出具有某个字串的行。

fgrep 可将欲过滤的一群字串放在某一个文件中，然后使用 fgrep 将包含有属于这一群字串的行过滤出来。

grep 与 fgrep 的用法如下：

grep [-nv] match_pattern file1 file2...。

fgrep [-nv] -f pattern_file file1 file2...。

-n 把所找到的行在行前加上行号列出

-v 把不包含 match_pattern 的行列出

match_pattern 所要搜寻的字串

-f 以 pattern_file 存放所要搜寻的字串

4 网路上查询状况的指令：

4.1 man

man 是手册(manual)的意思。Linux 提供线上辅助(on-line help)的功能，man 就是用来让使用者在使用时查询指令、系统呼叫、标准程式库函式、各种表格等的使用所用的。man 的用法如下：

man [-m path] [[section]title]...

man [-m path] -k keyword...

-m path man 所需要的 manual database 的路径。我们也可以用设定环境变数 MANPATH 的方式来取代-M 选项。

title 这是所要查询的目的物。

section 为一个数字表示 manual 的分类，通常 1 代表可执行指令，2 代表系统呼叫(systemcall)，3 代表标准函数，等等。

像下面 man 查询的片段：

SEE ALSO

apropos(1), cat(1V), col(1V), eqn(1), lpr(1), more(1),

nroff(1), refer(1), tbl(1), troff(1), vgrind(1), vtroff(1),

whatis(1), eqnchar(7), man(7), catman(8)

我们如要参考 eqnchar(7)的资料，则我们就输入 man7 eqnchar，便能取得我们所要的辅助讯息。

-k keyword 用来将含有这项 keyword 的 title 列出来。

man 在 Linux 上是一项非常重要的指令，我们在本讲义中所述之用法均仅只是一个大家比较常用的用法以及简单的说明，真正详细的用法与说明还是要请你使用 man 来得到。

4.2 who

who 指令是用来查询目前有那些人在线上。

4.3 w

w 指令是用来查询目前有那些人在线上，同时显示出那些人目前的工作。

4.4 ku

ku 可以用来搜寻整个网路上的 user，不像 w 跟 who 只是针对 localhost 的查询。而且 ku 提供让使用者建立搜寻特定使用者名单的功能。你可以建立一个档案 information-file 以条列的方式存放你的朋友资料，再建立一个文件 hosts-file 来指定搜寻的机器名称。ku 的指令格式可由 ku -h 得到。

5 网路指令：

Linux 提供网路的连接，使得你可以在各个不同的机器上做一些特殊的事情，如你可以在系上的 iris 图形工作站上做图形的处理，在系上的 Sun 上读 News，甚至到学校的计中去找别系的同学 talk。这些工作可以利用 Linux 的网路指令，在你的位子上连到各个不同的机器上工作。如此一来，即使你在寝室，也能轻易的连至系上或计中来工作，不用像以前的人必须泡在冷冰冰的机房面。

这些网路的指令如下所述：

5.1 rlogin 与 rsh

rlogin 的意义是 remotelogin，也就是经由网路到另外一部机器 login。

rlogin 的格式是：

rlogin host[-l username]

选项-l username 是当你在远方的机器上的 username 和 localhost 不同的时候，必须输入的选项，否则 rlogin 将会假设你在那边的 username 与 localhost 相同，然后在第一次 login 时必然会发生错误。

rsh 是在远方的机器上执行某些指令，而把结果传回 localhost。rsh 的格式如下：

rsh host [-l username][command]

如同 rlogin 的参数-l username，rsh 的-l username 也是指定 remote host 的 username。而 command 则是要在 remotehost 上执行的指令。如果没有指定 command，则 rsh 会去执行 rlogin，如同直接执行 rlogin。

不过 rsh 在执行的时候并不会像一般的 login 程序一样还会问你 password，而是如果你没有设定 trusttable，则 remotehost 将不会接受你的 request。

rsh 须要在每个可能会做为 remotehost 的机器上设定一个文件，称为 rhosts。这个文件每一行分为两个部份，第一个是允许 login 的 hostname，第二个部份则是允许 login 的 username。例如，在 ccsun7.csie.nctu.edu.tw 上头你的 username 为 ysjuang，而你的 home 下面的 rhost 有以下的一行：

ccsun6.cc.nctu.edu.twu8217529

则在 ccsun6.cc.nctu.edu.tw 机器上的 useru8217529 就可以用以下的方法来执行 rsh 程式：

\$rsh ccsun7.csie.nctu.edu.tw -l ysjuang cat mbox

将 ysjuang 在 ccsun7.csie.nctu.edu.tw 上的 mbox 文件内容显示在 localhost ccsun6.cc.nctu.edu.tw 上。

而如果 rhost 有这样的一行，则 ccsun6.cc.nctu.edu.tw 上的 useru8217529 将可以不用输入 password 而直接经由 rsh 或 rloginlogin 到 ccsun7.csie.nctu.edu.tw 来。

注意：

.rhost 是一个设定可以信任的人 login 的表格，因此如果设定不当将会让不法之徒有可以乘机侵入系统的机会。如果你阅读 man5rhosts，将会发现你可以在第一栏用+来取代任何 hostname，第二栏用+来取代任何 username。

如一般 user 喜欢偷懒利用 “+username” 来代替列一长串 hostname，但是这样将会使得即使有一台 PC 上跑 UNIX 的 user 有与你相同的 username，也可以得到你的 trust 而侵入你的系统。这样容易造成系统安全上的危险。因此本系禁止使用这样子的方式写你的 .rhost 档，如果发现将予以停机直到你找中心的工作人员将其改正为止。同理，如果你的第二个栏位为+，如 “hostname+”，则你是允许在某一部机器上的“所有” user 可以不用经由输入 password 来进入你的帐号，是壹种更危险的行为。所以请自行小心。

5.2 telnet

telnet 是一个提供 user 经由网路连到 remotehost。

telnet 的格式如下：

telnet [hostname| ip -address][port]

hostname 为一个像 ccsun1 或是 ccsun1.cc.nctu.edu.tw 的 name address，ip-address 则为一个由四个小於 255 的数字组成的 ip address，如 ccsun1 的 ip-address 为 140.113.17.173，ccsun1.cc.nctu.edu.tw 的 ip-address 为 140.113.4.11。你可以利用 telnet ccsun1 或 telnet 140.113.17.173 来连到 ccsun1。

port 为一些特殊的程式所提供给外界的沟通点，如资工系的 MUD 其 server 便提供一些 port 让 user 由这些 port 进入 MUD 程式。详情请参阅 telnet(1)的说明。

5.3 ftp

ftp 的意义是 File Transfer Program，是一个很常应用在网路文件传输的程式。ftp 的格式如下：

ftp [hostname| ip-address]

其中 hostname |ip-address 的意义跟 telnet 中的相同。

在进入 ftp 之後，如果与 remotehost 连接上了，它将会询问你 username 与密码，如果输入对了就可以开始进行文件传输。

在 ftp 中有许多的命令，详细的使用方式请参考 ftp(1)，这里仅列出较常用的 cd，lcd，mkdir，put，mput，get，mget，binary，ascii，prompt，help 与 quit 的使用方式。

ascii 将传输模式设为 ascii 模式。通常用於传送文字档。

binary 将传输模式设为 binary 模式，通常用於传送执行档，压缩档与影像档等。

cd remote-directory 将 remotehost 上的工作目录改变。

lcd [directory]更改 localhost 的工作目录。

ls [remote-directory][local-file]列出 remotehost 上的文件。

get remote-file[local-file]取得远方的文件。

mget remote-files 可使用通用字元一次取得多个文件。

put local-file[remote-file]将 localhost 的文件送到 remotehost。

mput local-files 可使用通用字元一次将多个文件放到 remotehost 上。

help[command]线上辅助指令。

mkdir directory-name 在 remotehost 造一个目录。

prompt 更改交谈模式，若为 on 则在 mput 与 mget 时每作一个文件之传输时均会询问。

quit/bye 离开 ftp。

利用 ftp，我们便可以在不同的机器上将所需要的资料做转移，某些特别的机器更存放大量的资料以供各地的使用者抓取，本校较著名的 ftpserver 有 NCTUCCCA 与系上的

ftp.csie.nctu.edu.tw。这些 ftpserver 均有提供一个 user 称为 anonymous，一般的“外来客”可以利用这个 username 取得该 server 的公共资料。不过 anonymous 在询问 password 时是要求使用 anonymous 的使用者输入其 emailaddress，以往有许多台湾的使用者在使用国外的 ftpserver 时并没有按照人家的要求输入其 emailaddress，而仅是随便打一些字串，引起许多 internetuser 和管理者的不满，对台湾的使用者的风评变得很差，因此遵循各 ftpserver 的使用规则也是一件相当重要的事。

6 关于通讯用的指令：

6.1 write

这个指令是提供使用者传送讯息给另一个使用者，使用方式：

```
write username[tty]
```

6.2 talk/ytalk/cytalk/ctalk

Linux 专用的交谈程式。会将萤幕分隔开为你的区域和交谈对象的区域，同时也可和不同机器的使用者交谈。使用方式：

```
talk username[@host][tty]
```

6.3 mesg

选择是否接受他人的 messege，若为 messegeno 则他人的 messege 将无法传送给你，同时他也无法干扰你的工作。使用方法：

```
mesg [-n|-y]
```

6.4 mail/elm

在网路上的 email 程式，可经由此程式将信件 mail 给他人。使用方式：

```
mail [username]
```

```
mail -fmailboxfile
```

如有信件，则直接键入 mail 可以读取你的 mail。

elm 提供较 mail 更为方便的介面，而且可做线上的 alias。你可以进入 elm 使用上下左右键来选读取的信件，并可按 h 取得线上的 help 文件。

使用方式：

```
elm[username]
```

```
elm-fmailboxfile
```

7 编译器(Compiler)：

Compiler 的用处在于将你所撰写的程式翻译成一个可执行文件。在资工系常用的程式语言是 C, pascal, FORTRAN 等。你可以先写好一个 C 或 Pascal 或 FORTRAN 的原始程式档，再用这些 compiler 将其翻成可执行档。你可以用这个方法来制造你自己的特殊指令。

7.1 cc/gcc(CCompiler)

```
/usr/bin/cc
```

```
/usr/local/bin/gcc
```

语法：cc [-oexecfile]source

```
gcc [-oexecfile]source
```

execfile 是你所希望的执行档的名称，如果没有加上 -o 选项编译出来的可执行档会以 a.out 做为档名。source 为一个以 .c 做为结尾的 C 程式档。请参阅 cc(1) 的说明。

7.2 pc(PascalCompiler)

/usr/local/bin/pc

语法: pc[-oexecfile]source

execfile 是你所希望的执行档的名称, 如果没有加上-o 选项编译出来的可执行档会以 a.out 做为档名。source 为一个以 .p 做为结尾的 Pascal 程式档。请参阅/net/home5/lang/man 中 pc(1)的说明。

7.3 f77(FortranCompiler)

/net/home5/lang/f77

语法: f77[-oexecfile]source

execfile 是你所希望的执行档的名称, 如果没有加上-o 选项编译出来的可执行档会以 a.out 做为档名。source 为一个以 .p 做为结尾的 FORTRAN 程式档。

8 有关列印的指令:

以下为印表所会用到的指令, 在本系的印表机有 lp1, lp2(点矩阵印表机), lw, sp, ps, compaq(雷射印表机), 供使用者使用。

8.1 lpr

lpr 为用来将一个文件印至列表机的指令。

用法:

lpr -P[printer]file1file2file3...

或

lpr-P[printer]<file1

例子:

lpr-Plp1hello。chello。lsthello。map

lpr-Plp1<hello。c

前者以参数输入所要印出的文件内容, 後者列印标准输入文件(standardinput)的内容, 因已将 hello。c 转向到标准输入, 故会印出 hello。c 的文件内容。

8.2 lpq

lpq 是用来观察 printerqueue 上的 Jobs。

用法:

lpq-P[printer]

8.3 lprm

lprm 是用来取消列印要求的指令。通常我们有时会印错, 或是误送非文字档资料至 printer, 此时就必须利用 lprm 取消列印 request, 以免造成资源的浪费。

用法:

lprm -P[printer][Jobsid|username]

lprm 用来清除 printerqueue 中的 Jobs, 如果你使用 JobId 作为参数, 则它将此 Job 自 printerqueue 清除, 如果你用 username 作为参数, 则它将此 queue 中所有 Owner 为此 username 的 Jobs 清除。