

## 第四章 首次登入与在线求助

- 1.若我们使用 X Window 的窗口管理员环境，那么要切换 X Window 与命令行模式，可以使用 Ctrl+Alt+F1（F1~F6）方式。Linux 默认会提供 6 个 Terminal 让用户登录。
- 2.Linux 默认提供了 7 个 Run level，最常用的是 run level 3 和 run level 5。
- 3.在终端登录 Linux 界面如下：

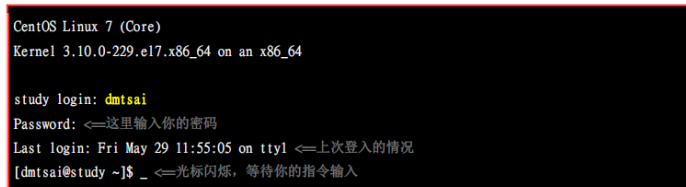


图 4.1 Linux 登录界面

- 4.开始执行命令  
格式如下：

```
[dmtsai@study ~]$ command [-options] parameter1 parameter2 ...
```

指令          选项          参数(1)      参数(2)

解释如下：

- (1) 一行指令中第一个输入的部分绝对是『命令』或『可执行文件（例如脚本,script）』。其中，command 为指令的名称。
- (2) 中括号[]并不存在于实际的指令中，而加入选项设定时，通常选项前会带 - 号。
- (3) parameter1, parameter2...为依附在选项后面的参数，或者是命令的参数。注意，**参数之间的空格无论几个都被视为一个！**
- (4) 指令太长的时候，可以使用反斜杠 (\) 来[Enter]符号，使指令连续到下一行。
- (5) 在 Linux 系统中，英文大小写字母是不一样的。

- 5.几个简单的常用命令：

命令	结果或说明
locale	LANG=zh_TW.utf8（剩余此处省略）
date	鑒? 5??29 14:24:36 CST 2015（纯文本界面下，无法显示中文字，所以前面是乱码）
date +%Y/%m/%d	2018/11/25
date +%H:%M	22:39
cal [month] [year]	（显示日历）
bc	（进入简单的计算器，接下来开始四则运算操作）
scale=number	（计算器中设置小数点位数，否则是整数计算）
LANG=en_US.utf8 export LC_ALL=en_US.utf8	（LANG 只与输出信息有关，若需要更改其他不同的信息，要同步更新 LC_ALL 才行）

- 6.常用快捷键

- (1) Tab

[Tab] 接在一串指令的第一个字的后面，则为『命令补全』；

[Tab] 接在一串指令的第二个字以后时，则为『文件补齐』！

- (2) Ctrl+C：停止系统正在运行的程序。

- (3) Ctrl+D：键盘输入结束（End Of File, EOF）。另外，他也可以用来取代 exit 的输入。

(4) [shift]+{[PageUP][Page Down]}：使用[Shift]+[Page Up]来往前翻页，也能够使用[Shift]+[Page Down]来往后翻页。

7.指令的 help 帮助，格式如下，以 cal 为例：

cal -help (有的是--help，例如 date 就是)

8.使用 man page：输入“man date”即可，在这里我以 date 命令来做的演示。输出结果如下：

```
DATE(1)                                User Commands                                DATE(1)
//后续内容全部省略
```

其中，date 后面括号中的数字 1 代表的是“一般用户可使用的指令”。具体不同数字的含义如下：

代码	代表内容
<b>1</b>	<b>用户在 shell 环境中可以操作的指令或可执行文件</b>
2	系统核心可调用的函数与工具等
3	一些常用的函数(function)与函数库(library)，大部分为 C 的函数库(libc)
4	设备文档的说明，通常在/dev 下的文件
<b>5</b>	<b>配置文件或者是某些文件的格式</b>
6	游戏(games)
7	惯例与协议等，例如 Linux 文件系统、网络协议、ASCII code 等等的说明
<b>8</b>	<b>系统管理员可用的管理指令</b>
9	跟 kernel 有关的文件

man page 的内容说明：

代码	内容说明
NAME	简短指令、数据名称的说明
SYNOPSIS	简短的命令执行语法(syntax)简介
DESCRIPTION	较为完整的说明
OPTIONS	针对 SYNOPSIS 部分中，有列举的所有可用的选项说明
COMMANDS	当这个程序(软件)在执行的时候，可以在此程序(软件)中下达的指令
FILES	这个程序或数据所使用或参考或连接到的某些文档
SEE ALSO	可以参考的，跟这个指令或数据有相关的其他说明！
EXAMPLE	一些可以参考的范例

在 man page 页面能进行的操作及说明：

按键	进行工作
空格键	向下翻一页

[PageDown]	向下翻一页
[PageUp]	向上翻一页
[Home]	去到第一页
[End]	去到最后一页
/string	向『下』搜寻 string 这个字符串，如果要搜寻 vbird 的话，就输入/vbird
?string	向『上』搜寻 string 这个字符串
n,N	利用/或?来搜寻字符串时，可以用 n 来继续下一个搜寻(不论是/或?)，可以利用 N 来进行『反向』搜寻。举例来说，我以/vbird 搜寻 vbird 字符串，那么可以 n 继续往下查询，用 N 往上查询。若以?vbird 向上查询 vbird 字符串，那我可以用 n 继续『向上』查询，用 N 反向查询。
q	结束这次的 man page

## 9. 搜寻特定指令/文件的 man page 说明文件

例题：

你可否查出来，系统中还有哪些跟『man』这个指令有关的说明文件呢？

答：

你可以使用底下的指令来查询一下：

```
[dmtsai@study ~]$ man -f man
```

如果需要查找关键字，可以使用下面这个命令：

`man -k man`

10.info page: info page 是将文件数据拆成一个一个的段落，每个段落用自己的页面来撰写，并且在各个页面还有类似网页的“超链接”来跳到各个不同的页面，每个独立的页面也被称为一个节点（Node）。

假如输入 info wget，得到下面的页面：

```
Next: Overview, Prev: (dir), Up: (dir)

Wget 1.19.4
*****

This file documents the GNU Wget utility for downloading network data.

Copyright © 1996-2011, 2015, 2018 Free Software Foundation, Inc.

Permission is granted to copy, distribute and/or modify this document
under the terms of the GNU Free Documentation License, Version 1.3 or
any later version published by the Free Software Foundation; with no
Invariant Sections, with no Front-Cover Texts, and with no Back-Cover
Texts. A copy of the license is included in the section entitled "GNU
Free Documentation License".

* Menu:

* Overview::          Features of Wget.
* Invoking::          Wget command-line arguments.
* Recursive_Download:: Downloading interlinked pages.
* Following_Links::   The available methods of chasing links.
* Time-Stamping::     Mirroring according to time-stamps.
* Startup_File::      Wget's initialization file.
* Examples::          Examples of usage.
* Various::           The stuff that doesn't fit anywhere else.

-----Info: (wget)Top, 30 lines  --Top-----
Welcome to Info version 6.5. Type H for help, h for tutorial.
```

图 4.2 Info wget 页面

Next: 下一个节点的名称为 Overview，也可以按 N 到下一个节点。

Up: 回到上一层的节点总揽画面，也可以按 U 回到上一层。

Prev: 前一个节点。按 P 到前一个节点。

按键	进行工作
空格键	向下翻一页
[Page Down]	向下翻一页
[Page Up]	向上翻一页
[TAB]	在 node 之间移动, 有 node 的地方, 通常会以 * 显示。
[ENTER]	当光标在 node 上面时, 按下 Enter 可以进入该 node 。
B	移动光标到该 info 画面当中的第一个 node 处
E	移动光标到该 info 画面当中的最后一个 node 处
N	前往下一个 node 处
P	前往上一个 node 处
U	向上移动一层
S(/)	在 info page 当中进行搜寻
H, ?	显示求助选单
Q	结束这次的 info page

总结上面的三个命令(man,info,/usr/share/doc/):

(1) 在终端机模式中, 如果你知道某个指令, 但却忘记了相关选项与参数, 请先善用 --help 的功能来查询相关信息;

(2) 当有任何你不知道的指令或文件格式这种玩意儿, 但是你想要了解他, 请赶快使用 man 或者是 info 来查询!

(3) 而如果你想要架设一些其他的服, 或想要利用一整组软件来达成某项功能时, 请赶快到/usr/share/doc 底下查一查有没有该服务的说明档喔!

11.正确的关机方法:

命令	功能
who	查看有谁在线
netstat -a	查看网络的联机状态
ps -aux	查看后台执行的程序
sync	将数据同步写入到硬盘中
shutdown	管用的关机命令
reboot, halt, poweroff	重新启动, 关机

shutdown 的作用:

(1) 可以自由选择关机模式: 是要关机或重新启动均可;

(2) 可以设定关机时间:可以设定成现在立刻关机,也可以设定某一个特定的时间才关机。

(3) 可以自定义关机讯息: 在关机之前, 可以将自己设定的讯息传送给在线 user。

(4) 可以仅发出警告讯息: 有时有可能你要进行一些测试, 而不想让其他的使用者干扰, 或者是明白的告诉使用者某段时间要注意一下! 这个时候可以使用 shutdown 来吓一吓

使用者，但却不是真的要关机啦！

使用方式举例：

选项与参数：

```
-k      : 不要真的关机，只是发送警告讯息出去！
-r      : 在将系统的服务停掉之后就重新启动(常用)
-h      : 将系统的服务停掉后，立即关机。(常用)
-c      : 取消已经在进行的 shutdown 指令内容。
```

## 12. systemctl

[root@study ~]# **systemctl** [指令]

指令项目包括如下：

```
halt      进入系统停止的模式，屏幕可能会保留一些讯息，这与你的电源管理模式有关
poweroff  进入系统关机模式，直接关机没有提供电力喔！
reboot    直接重新启动
suspend   进入休眠模式
```

[root@study ~]# **systemctl reboot** # 系统重新启动

[root@study ~]# **systemctl poweroff** # 系统关机

## 第五章 Linux 的文件权限与目录配置

1. 默认的情况下，所有的系统上的账号与一般身份使用者，还有那个 root 的相关信息，都是记录在/etc/passwd 这个文件内的。至于个人的密码则是记录在/etc/shadow 这个文件下。此外，Linux 所有的组名都纪录在/etc/group 内！

2. 以 su 切换为 root 身份后，输入 ls -al 后，得到下面的结果：

```
root@xiaoao-virtual-machine:/home/xiaoao# ls -al
总用量 4188
drwxr-xr-x 20 xiaoao xiaoao 4096 11月 26 19:31 .
drwxr-xr-x  3 root   root   4096 9月   5 22:07 ..
-rw-----  1 xiaoao xiaoao 13420 11月 26 16:06 .bash_history
-rw-r--r--  1 xiaoao xiaoao  220 9月   5 22:07 .bash_logout
-rw-r--r--  1 xiaoao xiaoao 3771 9月   5 22:07 .bashrc
drwx----- 19 xiaoao xiaoao 4096 11月 19 15:35 .cache
drwx----- 20 xiaoao xiaoao 4096 11月 20 10:39 .config
drwx-----  3 xiaoao xiaoao 4096 9月   6 11:05 .dbus
-rw-r--r--  1 xiaoao xiaoao 8980 9月   5 22:07 examples.desktop
drwxr-xr-x 18 xiaoao xiaoao 4096 9月   8 18:42 fftw-3.3.8
-rw-rw-r--  1 xiaoao xiaoao 4110137 9月   7 17:29 fftw-3.3.8.tar.gz
```

图 5.1 ls 结果

(1) **第一列：权限类型**。共有 10 个字符，分别为以下内容：

①第一个字符：**文件类型**（目录/文件/链接文件等）。

若为 **d** 则是目录，若为 **-** 则是文件，若为 **l** 则是连接文件，若为 **b** 则是设备文件里面的可供存储的接口设备，若为 **c** 则表示设备文件里面的串行端口设备。

②第二至四个字符：**文件所有者的权限**，为 **rw** 三个参数的组合，**r** 为可读，**w** 为可写，

x 为可执行，若为-则表示无权限。

③第五至七个字符：同用户组的用户的权限。同上。

④第八至十个字符：其它非本用户组的用户的权限。同上。

## (2) 第二列：多少文件名连接到此节点 (i-node)

每个文档都会将他的权限与属性记录到文件系统的 i-node 中，不过，我们使用的目录树却是使用文件名来记录，因此每个档名就会连结到一个 i-node。这个属性记录的，就是有多少不同的档名连结到相同的一个 i-node 号码去就是了。

## (3) 第三列：这个文件(或目录)的『拥有者账号』

## (4) 第四列：表示这个文件的所属群组

## (5) 第五列：文件的容量大小，默认单位为 bytes；

## (6) 第六列：文件的建档日期或者是最近的修改日期：

## (7) 第七列：文件名。注意：如果文件名之前多一个『.』，则代表这个文件为『隐藏文件』。

## 2.Linux 文件权限的重要性

(1) 系统的保护功能；

(2) 团队开发软件或数据共享的功能；

(3) 未将权限设置妥当的危害：如果你的目录权限没有作好的话，可能造成其他人都可以在你的系统上面乱搞。

**在你修改你的 linux 文件与目录的属性之前，一定要先搞清楚，什么数据是可变的，什么是不可变的。**

## 3.改变文件属性和权限

(1) chgrp：改变文件所属群组（群组名必须在/etc/group 中有记录）

格式如下：

```
[root@study ~]# chgrp [-R] dirname/filename ...
```

选项与参数：

-R：进行递归(recursive)的持续变更，亦即连同次目录下的所有文件、目录都更新成为这个群组之意。常常用在变更某一目录内所有的文件之情况。

(2) chown：改变文件拥有者（用户名必须在/etc/passwd 中有记录）

```
[root@study ~]# chown [-R] 账号名称 文件或目录
```

```
[root@study ~]# chown [-R] 账号名称:组名 文件或目录
```

选项与参数：

-R：进行递归(recursive)的持续变更，亦即连同子目录下的所有文件都变更

(3) chmod：改变文件的权限，SUID、SGID、SBIT 等等的特性。

权限设定的方法有两种，第一种是使用数字（自己在以前管理服务器的时候常用），第二种是采用符号来变更。

①采用数字来变更。读/写/执行采用的是三位二进制码（xxx）依序来表示，但是一般直接表示为一位十进制数。例如-rwxrwxrwx 表示为 777，这也是在服务器设置很多时候用到的数字。格式如下：

```
[root@study ~]# chmod [-R] xyz 文件或目录
```

选项与参数：

xyz：就是刚刚提到的数字类型的权限属性，为 rwx 属性数值的相加。

-R：进行递归(recursive)的持续变更，亦即连同次目录下的所有文件都会变更



②采用符号改变文件权限。格式如下：

chmod	u g o a	+(加入) -(去除) =(设定)	r w x	文件或目录
-------	------------------	-------------------------	-------------	-------

这里面，u 表示 user，g 表示 group，o 表示 others，a 表示 all。

使用举例：

```
[root@study ~]# chmod u=rwx,go=rx .bashrc
[root@study ~]# chmod a+w .bashrc
[root@study ~]# chmod a-x .bashrc
# 注意！那个 u=rwx,go=rx 是连在一起的，中间并没有任何空格符！
```

### 3. 目录与文件的权限意义

#### (1) 权限对于文件的重要性

- ①r (Read): 可读取此文件的实际内容，如读取文本文件的文字内容等；
- ②w (Write): 可以编辑、新增或者是修改该文件的内容(但不含删除该文件)；
- ③x (eXecute): 该文件具有可以被系统执行的权限。

注意：在 Windows 下，一个文件是否能够执行是由扩展名决定，而 **Linux 下文件是否能被执行是由是否具有「x」这个权限来决定的**，跟文件名是没有绝对的关系的。

当你对于一个文件具有 w 权限时，你可以具有写入/编辑/新增/修改文件的内容的权限，**但并不具备有删除该文件本身的权限。**

#### (2) 权限对于目录的重要性

**目录主要的内容在记录文件名列表，文件名与目录有强烈的关联。**

①r (read contents in directory): 表示具有读取目录结构列表的权限，所以当你具有读取 (r) 一个目录的权限时，表示你可以查询该目录下的文件名数据。所以你就可以利用 ls 这个指令将该目录的内容列表显示出来！

②w (modify contents of directory): 具有更改该目录结构列表的权限，也就是底下这些权限：

建立新的文件与目录；

删除已经存在的文件与目录（不论该文件的权限为何！）

将已存在的文件或目录进行更名；

移动该目录内的文件、目录位置。

③x (access directory): 目录的 x 代表的是用户能否进入该目录成为工作目录的用途。

总的来说，如下表所示：

类型	内容	迭代类型	r	w	x
文件	详细资料	文件（夹）	读到内容	修改内容	执行文件内容
目录	文件名	可分类抽屉	读到档名	修改档名	进入该目录的权限

### 4. Linux 文件种类与扩展名

#### (1) 文件种类（regular file）

①普通文件：就是一般我们在进行存取类型的文件，在由 ls -al 所显示出来的属性方面，第一个字符为[-]，例如[-rwxrwxrwx]。另外，依照文件的内容，又大略可以分为：

- 纯文本文档（ASCII）
- 二进制文件（binary）

- 数据格式文件 (data)

(2) 目录 (directory): 第一个属性为 d 的。

(3) 连接文件 (link): 类似于 windows 下面的快捷方式, 第一个属性为 l。

(4) 设备与设备文件 (device): 与系统周边及储存等相关的一些文件, 通常都集中在 /dev 这个目录之下! 通常又分为两种:

- 块设备文件: 就是一些存储数据, 以提供系统随机访问的借口设备, 例如硬盘灯。第一个属性为 b。

- 字符设备文件: 也即是一些串行端口的接口设备, 如键盘、鼠标。第一个属性为 c。

(5) 套接字 (sockets): 通常被用在网络上的数据连接。我们可以启动一个程序来监听客户端的请求, 而客户端就可以通过这个 socket 来进行数据的通信, 其第一个属性为 s, 通常在 /var/run 目录中可以看到这种文件类型。

(6) 管道 (FIFO, pipe): 一种特殊的文件类型, 主要目的是为了解决多个程序同时存取一个文件所造成的错误问题, 第一个属性为 p。

## (2) Linux 文件扩展名

一个 Linux 文件能不能被执行, 与他的第一栏的十个属性有关, 与文件名根本一点关系也没有。

不过, 可以被执行与可以执行成功是不一样的概念。我们仍然希望可以由扩展名来了解该文件是什么东西, 所以通常我们还是会以适当的扩展名来表示该文件是什么种类的。例如: \*.sh: 脚本或批处理文件(scripts), 因为批处理文件为使用 shell 写成的, 所以扩展名就编成.sh。

## (3) Linux 文件长度限制

单一文件或目录的最大容许文件名为 **255bytes**, 以一个 ASCII 英文占用一个 bytes 来说, 则大约可达 **255 个字符**长度。若是以每个中文字占用 2bytes 来说, 最大文件名就是大约在 **128 个中文字**。

## (3) Linux 文件名的限制

应避开 \* ? > < ; & ! [ ] \ ' " ` ( ) { } , 同时, 小数点 { . } 表示隐藏文件。另外, 因为常常用到 -option 之类的选项, 应避免文件名第一个字符为 - 或 +。

## 5. Linux 目录配置

### (1) FHS (Filesystem Hierarchy Standard) 标准

	可分享的(shareable)	不可分享的(unshareable)
不变的(static)	/usr (软件放置处)	/etc (配置文件)
	/opt (第三方软件)	/boot (开机与内核文件)
可变动的(variable)	/var/mail (用户邮件信箱)	/var/run (程序相关)
	/var/spool/news (新闻组)	/var/lock (程序相关)

类型说明:

①可分享的: 可以分享给其他系统挂载使用的目录, 所以包括执行文件与用户的邮件等数据, 是能够分享给网络上其他主机挂载用的目录;

②不可分享的: 自己机器上面运行的设备文件或者是与程序有关的 socket 文件等, 由于仅与自身机器有关, 所以当然就不适合分享给其他主机了。

③不变的: 有些数据是不会经常变动的, 跟随着 distribution 而不变动。例如函数库、文件的说明文件、系统管理员所管理的主机服务配置文件等等;

④可变动的: 经常改变的数据, 例如登录文件、一般用户可自行收受的新闻组等。

FHS 针对目录树架构仅定义出三层目录底下应该放置什么数据而已, 分别是底下这三



个目录的定义：

**a) /(root, 根目录)：与开机系统有关；**

根目录是整个系统最重要的一个目录，因为不但所有的目录都是由根目录衍生出来的，同时**根目录也与开机/还原/系统修复等动作有关**。FHS 标准建议：根目录(/)所在分割槽应该越小越好，且应用程序所安装的软件最好不要与根目录放在同一个分割槽内，保持根目录越小越好。如此不但效能较佳，根目录所在的文件系统也较不容易发生问题。

目录	应放置文件内容
第一部份：FHS 要求必须要存在的目录	
/bin	系统有很多放置执行文件的目录，但/bin 比较特殊。因为/bin 放置的是在单人维护模式下还能够被操作的指令。在/bin 底下的指令可以被 root 与一般账号所使用，主要有：cat, chmod, chown, date, mv, mkdir, cp, bash 等等常用的指令。
/boot	这个目录主要在放置开机会使用到的文件，包括 Linux 核心文件以及开机选单与开机所需配置文件等等。Linux kernel 常用的档名为：vmlinuz，如果使用的是 grub2 这个开机管理程序，则还会存在/boot/grub2/这个目录喔！
/dev	在 Linux 系统上，任何装置与接口设备都是以文件的型态存在于这个目录当中的。你只要透过存取这个目录底下的某个文件，就等于存取某个装置啰～ 比要重要的文件有/dev/null, /dev/zero, /dev/tty, /dev/loop*, /dev/sd* 等等
/etc	<p>系统主要的配置文件几乎都放置在这个目录内，例如人员的账号密码文件、各种服务的启始档等等。一般来说，这个目录下的各文件属性是可以让一般使用者查阅的，但是只有 root 有权力修改。FHS 建议不要放置可执行文件(binary)在这个目录中喔。比较重要的文件有：/etc/modprobe.d/, /etc/passwd, /etc/fstab, /etc/issue 等等。另外 FHS 还规范几个重要的目录最好要存在 /etc/ 目录下喔：</p> <ul style="list-style-type: none"><li>• /etc/opt(必要)：这个目录在放置第三方协力软件 /opt 的相关配置文件</li><li>• /etc/X11/(建议)：与 X Window 有关的各种配置文件都在这里，尤其是 xorg.conf 这个 X Server 的配置文件。</li><li>• /etc/sgml/(建议)：与 SGML 格式有关的各项配置文件</li><li>• /etc/xml/(建议)：与 XML 格式有关的各项配置文件</li></ul>
/lib	系统的函式库非常的多，而/lib 放置的则是在开机时会用到的函式库，以及在/bin 或/sbin 底下的指令会呼叫的函式库而已。什么是函式库呢？妳可以将他想成是『外挂』，某些指令必须要有这些『外挂』才能够顺利完成程序的执行之意。另外 FHS 还要求底下的目录必须要存在：

	<ul style="list-style-type: none"> <li>• <b>/lib/modules/</b>: 这个目录主要放置可抽换式的核心相关模块(驱动程序)喔!</li> </ul>
<b>/media</b>	media 是『媒体』的英文, 顾名思义, 这个 <b>/media</b> 底下放置的就是可移除的装置啦! 包括软盘、光盘、DVD 等等装置都暂时挂载于此。常见的档名有: <b>/media/floppy</b> , <b>/media/cdrom</b> 等等。
<b>/mnt</b>	如果你想要暂时挂载某些额外的装置, 一般建议你可以放置到这个目录中。 在古早时候, 这个目录的用途与 <b>/media</b> 相同啦! 只是有了 <b>/media</b> 之后, 这个目录就用来暂时挂载用了。
<b>/opt</b>	这个是给 <b>第三方协力软件放置的目录</b> 。什么是第三方协力软件啊? 举例来说, KDE 这个桌面管理系统是一个独立的计划, 不过他可以安装到 Linux 系统中, 因此 KDE 的软件就建议放置到此目录下了。 另外, 如果你想要自行安装额外的软件(非原本的 <b>distribution</b> 提供的), 那么也能够将你的软件安装到这里来。 不过, 以前的 Linux 系统中, 我们还是习惯放置在 <b>/usr/local</b> 目录下呢!
<b>/run</b>	早期的 FHS 规定系统开机后所产生的各项信息应该要放置到 <b>/var/run</b> 目录下, 新版的 FHS 则规范到 <b>/run</b> 底下。 由于 <b>/run</b> 可以使用内存来仿真, 因此效能上会好很多!
<b>/sbin</b>	Linux 有非常多指令是用来设定系统环境的, 这些指令只有 <b>root</b> 才能够利用来『设定』系统, 其他用户最多只能用来『查询』而已。 放在 <b>/sbin</b> 底下的为开机过程中所需要的, 里面包括了开机、修复、还原系统所需要的指令。 至于某些服务器软件程序, 一般则放置到 <b>/usr/sbin</b> /当中。至于本机自行安装的软件所产生的系统执行文件(system binary), 则放置到 <b>/usr/local/sbin</b> /当中了。常见的指令包括: <b>fdisk</b> , <b>fsck</b> , <b>ifconfig</b> , <b>mkfs</b> 等等。
<b>/srv</b>	<b>srv</b> 可视为『service』的缩写, 是一些网络服务启动之后, 这些服务所需要取用的数据目录。 常见的服务例如 <b>WWW</b> , <b>FTP</b> 等等。举例来说, <b>WWW</b> 服务器需要的网页资料就可以放置在 <b>/srv/www</b> /里面。 不过, 系统的服务数据如果尚未要提供给因特网任何人浏览的话, 预设还是建议放置到 <b>/var/lib</b> 底下即可。
<b>/tmp</b>	这是让一般用户或者是正在执行的程序暂时放置文件的地方。 这个目录是任何人都能够存取的, 所以你需要定期的清理一下。当然, 重要数据不可放置在此目录啊! 因为 FHS 甚至建议在开机时, 应该要将 <b>/tmp</b> 下的数据都删除唷!
<b>/usr</b>	第二层 FHS 设定, 后续介绍
<b>/var</b>	第二层 FHS 设定, 主要为放置变动性的数据, 后续介绍
第二部份: FHS 建议可以存在的目录	

/home	这是系统默认的用户家目录(home directory)。在你新增一个一般使用者账号时，默认的用户家目录都会规范到这里来。比较重要的是，家目录有两种代号喔： <ul style="list-style-type: none"> <li>• ~: 代表目前这个用户的家目录</li> <li>• ~dmtsai: 则代表 dmtsai 的家目录！</li> </ul>
/lib<qual>	用来存放与 /lib 不同的格式的二进制函式库，例如支持 64 位的 /lib64 函式库等
/root	系统管理员(root)的家目录。之所以放在这里，是因为如果进入单人维护模式而仅挂载根目录时，该目录就能够拥有 root 的家目录，所以我们会希望 root 的家目录与根目录放置在同一个分割槽中。

FHS 针对根目录的定义就只有以上。不过 Linux 下面还有很多目录很重要，其中几个如下所示：

目录	应放置档案内容
/lost+found	这个目录是使用标准的 ext2/ext3/ext4 文件系统格式才会产生的一个目录，目的在于当文件系统发生错误时，将一些遗失的片段放置到这个目录下。不过如果使用的是 xfs 文件系统的话，就不会存在这个目录了！
/proc	这个目录本身是一个『虚拟文件系统(virtual filesystem)』喔！他放置的数据都是在内存当中，例如系统核心、行程信息(process)、周边装置的状态及网络状态等等。因为这个目录下的数据都是在内存当中，所以本身不占任何硬盘空间啊！比较重要的档案例如：/proc/cpuinfo, /proc/dma, /proc/interrupts, /proc/ioports, /proc/net/*等等。
/sys	这个目录其实跟/proc 非常类似，也是一个虚拟的文件系统，主要也是记录核心与系统硬件信息较相关的信息。包括目前已加载的核心模块与核心侦测到的硬件装置信息等等。这个目录同样不占硬盘容量喔！

b) /usr (unix software resource): 与软件安装/执行有关；

/usr 里面放置的数据属于**可分享的与不可变动的**。在安装时会占用较大硬盘容量的目录。

c) /var (variable): 与系统运作过程有关。在系统运行后才会占用硬盘容量的目录。这里面主要针对常态性变动文件，包括缓存(cache)、登录文件(log file)以及某些软件运行所产生的文件，包括程序文件，或者如 MySQL 数据库的文件等。

## (2) 目录树 (directory tree)

在 Linux 底下，所有的文件与目录都是由根目录开始的，然后再一个一个的分支下来，我们称这种目录配置方式为**目录树 (directory tree)**。它有以下特性：

- ①目录树的起始点为根目录 (/ , root)；
- ②每一个目录不只能使用本地端的文件系统，也可以使用网络上的文件系统；
- ③每一个文件在此目录树中的文件名（包括完整路径）都是独一无二的。

## (3) 绝对路径与相对路径

绝对路径：由根目录(/)开始写起的文件名或目录名称，例如 /home/dmtsai/.bashrc；

相对路径：相对于目前路径的文件名写法。例如./home/dmtsai 或../..home/dmtsai/等等。

反正开头不是[/]就属于相对路径的写法。

要注意这样两点：

- ①[.]: 代表当前的目录，也可以使用[./]来表示。
- ②[..]: 代表上一层目录，也可以用[..]来表示。