

Linux Systems and Open Source Software

Linux File and Filesystem Command Line Tools

Chia-Heng Tu
Dept. of Computer Science and Information Engineering
National Cheng Kung University
Fall 2022



Outline

- Linux Directory Tree
  - Filesystem Hierarchy Standard (FHS)
  - Absolute and relative paths
- Commands for Browsing File System
  - File and directory management
  - File content inspection
  - File searching
- User and Group in Linux
  - File and operation permissions



Filesystem Hierarchy Standard (FHS)

Absolute and relative paths

LINUX DIRECTORY TREE



Filesystem Hierarchy Standard (FHS)

- Filesystem Hierarchy Standard (FHS) defines the hierarchy of directory trees in Linux
  - including the directory structure and directory contents
  - The latest version is 3.0, released on 3 June 2015
- In FHS, all files and directories appear under **the root directory /**
- Most of these directories exist in all Unix-like operating systems and are generally used in much the same way

An Example of Filesystem Hierarchy of Ubuntu16.04

```
shaohua@NCKU-AV-IPC:/$ ls -l
total 65896
drwxr-xr-x   2 root root     4096 二   6 06:10 bin
drwxr-xr-x   3 root root     4096 二  20 06:50 boot
drwxrwxr-x   2 root root     4096 一   7  2019 cdrom
-rw-------   1 root root 77357056 十  31 18:20 core
drwxr-xr-x  20 root root     4080 十二 12 23:09 dev
drwxr-xr-x 150 root root    12288 二  12 06:47 etc
drwxr-xr-x   5 root root     4096 一   7  2019 home
lrwxrwxrwx   1 root root       33 二  19 06:31 initrd.img -> boot/initrd.img-4.15.0-88-generic
lrwxrwxrwx   1 root root       33 二  19 06:31 initrd.img.old -> boot/initrd.img-4.15.0-76-generic
drwxr-xr-x  24 root root     4096 九   2 20:53 lib
drwxr-xr-x   2 root root     4096 三   2  2019 lib32
drwxr-xr-x   2 root root     4096 三   2  2019 lib64
drwxr-xr-x   2 root root    12288 三   2  2019 libx32
drwx------   2 root root    16384 一   7  2019 lost+found
drwxr-xr-x   4 root root     4096 十   2 17:26 media
drwxr-xr-x   2 root root     4096 七  31  2018 mnt
drwxr-xr-x   5 root root     4096 九   3 09:45 opt
dr-xr-xr-x 287 root root        0 十  31 18:21 proc
drwx------   5 root root     4096 二   3 10:08 root
drwxr-xr-x  30 root root     1220 二  20 09:15 run
drwxr-xr-x   2 root root    12288 二   6 06:10 sbin
drwxr-xr-x   2 root root     4096 一   7  2019 snap
drwxr-xr-x   2 root root     4096 七  31  2018 srv
dr-xr-xr-x  13 root root        0 十  31 18:21 sys
drwxrwxrwt 131 root root    20480 二  20 09:17 tmp
drwxr-xr-x  16 root root     4096 十   2 17:26 usr
drwxr-xr-x  15 root root     4096 十   2 17:26 var
lrwxrwxrwx   1 root root       30 二  19 06:31 vmlinuz -> boot/vmlinuz-4.15.0-88-generic
lrwxrwxrwx   1 root root       30 二  19 06:31 vmlinuz.old -> boot/vmlinuz-4.15.0-76-generic
```

Courtesy of https://en.wikipedia.org/wiki/Filesystem_Hierarchy_Standard



Filesystem Hierarchy Standard (FHS)

/boot	Boot loader files, *e.g.*, kernels, initrd.
/dev	Device files, *e.g.*, /dev/null, /dev/disk0, /dev/sda1, /dev/tty, /dev/random.
/etc	Host-specific system-wide configuration files
/home	Users' home directories, containing saved files, personal settings, etc.
/lib	Libraries essential for the binaries in /bin and /sbin.
/media	Mount points for removable media such as CD-ROMs (appeared in FHS-2.3 in 2004).
/mnt	Temporarily mounted filesystems.
/opt	Optional application software packages.
/proc	Virtual filesystem providing process and kernel information as files. In Linux, corresponds to a procfs mount. Generally automatically generated and populated by the system, on the fly.
/root	Home directory for the root user.
/run	Run-time variable data: Information about the running system since last boot, *e.g.*, currently logged-in users and running daemons. Files under this directory must be either removed or truncated at the beginning of the boot process
/sbin	Essential system binaries, *e.g.*, fsck, init, route.
/srv	Site-specific data served by this system, such as data and scripts for web servers, data offered by FTP servers, and repositories for version control systems (appeared in FHS-2.3 in 2004).
/sys	Contains information about devices, drivers, and some kernel features.
/tmp	Temporary files (see also /var/tmp). Often not preserved between system reboots, and may be severely size restricted.
/usr	*Secondary hierarchy* for read-only user data; contains the majority of (multi-)user utilities and applications.
/var	Variable files—files whose content is expected to continually change during normal operation of the system—such as logs, spool files, and temporary e-mail files.
/	*Primary hierarchy* root and root directory of the entire file system hierarchy.
/bin	Essential command binaries that need to be available in single user mode; for all users, *e.g.*, cat, ls, cp.

Courtesy of https://en.wikipedia.org/wiki/Filesystem_Hierarchy_Standard



Filesystem Hierarchy Standard (FHS)

- Sub-directory of **/etc**

/etc	Host-specific system-wide configuration files There has been controversy over the meaning of the name itself. In early versions of the UNIX Implementation Document from Bell labs, /etc is referred to as the *etcetera directory*,[3] as this directory historically held everything that did not belong elsewhere (however, the FHS restricts /etc to static configuration files and may not contain binaries).[4] Since the publication of early documentation, the directory name has been re-explained in various ways. Recent interpretations include backronyms such as "Editable Text Configuration" or "Extended Tool Chest".[5]
/etc/opt	Configuration files for add-on packages that are stored in /opt.
/etc/sgml	Configuration files, such as catalogs, for software that processes SGML.
/etc/X11	Configuration files for the X Window System, version 11.
/etc/xml	Configuration files, such as catalogs, for software that processes XML.

Courtesy of https://en.wikipedia.org/wiki/Filesystem_Hierarchy_Standard



Filesystem Hierarchy Standard (FHS)

- Sub-directory of **/usr**

/usr	*Secondary hierarchy* for read-only user data; contains the majority of (multi-)user utilities and applications.[8]
/usr/bin	Non-essential command binaries (not needed in single user mode); for all users.
/usr/include	Standard include files.
/usr/lib	Libraries for the binaries in /usr/bin and /usr/sbin.
/usr/lib<qual>	Alternative format libraries, *e.g.* /usr/lib32 for 32-bit libraries on a 64-bit machine (optional).
/usr/local	*Tertiary hierarchy* for local data, specific to this host. Typically has further subdirectories, *e.g.*, bin, lib, share.[9]
/usr/sbin	Non-essential system binaries, *e.g.*, daemons for various network-services.
/usr/share	Architecture-independent (shared) data.
/usr/src	Source code, *e.g.*, the kernel source code with its header files.
/usr/X11R6	X Window System, Version 11, Release 6 (up to FHS-2.3, optional).

Courtesy of https://en.wikipedia.org/wiki/Filesystem_Hierarchy_Standard



Filesystem Hierarchy Standard (FHS)

- Sub-directory of **/var**

/var	Variable files—files whose content is expected to continually change during normal operation of the system—such as logs, spool files, and temporary e-mail files.
/var/cache	Application cache data. Such data are locally generated as a result of time-consuming I/O or calculation. The application must be able to regenerate or restore the data. The cached files can be deleted without loss of data.
/var/lib	State information. Persistent data modified by programs as they run, *e.g.*, databases, packaging system metadata, etc.
/var/lock	Lock files. Files keeping track of resources currently in use.
/var/log	Log files. Various logs.
/var/mail	Mailbox files. In some distributions, these files may be located in the deprecated /var/spool/mail.
/var/opt	Variable data from add-on packages that are stored in /opt.
/var/run	Run-time variable data. This directory contains system information data describing the system since it was booted.[10] In FHS 3.0, /var/run is replaced by /run; a system should either continue to provide a /var/run directory, or provide a symbolic link from /var/run to /run, for backwards compatibility.[11]
/var/spool	Spool for tasks waiting to be processed, *e.g.*, print queues and outgoing mail queue.
/var/spool/mail	Deprecated location for users' mailboxes.[12]
/var/tmp	Temporary files to be preserved between reboots.

Courtesy of https://en.wikipedia.org/wiki/Filesystem_Hierarchy_Standard



Filesystem Hierarchy Standard (FHS)

Absolute and relative path

LINUX DIRECTORY TREE



Absolute and Relative Paths

- An absolute path
  - describes the location of a file or folder, regardless of the current working directory
  - In fact, an absolute path is relative to **the root directory /**
- A relative path
  - describes the location of a file or folder in relative to **the current working directory**
- **Example**
  When you are in **/usr** and want to change to **/home**, you have two choices:
  1. Using absolute path: **$/> cd /home**
  2. Using relative path: **$/> cd ../home**
- All directories contain two entries, "**.**" and "**..**" ,
  - which stand for the directory itself and its parent, respectively
  - E.g., the command **$/> ./run.sh** means to execute the **run.sh** file from the current directory

Courtesy of http://www.differencebetween.net/technology/difference-between-absolute-and-relative-path/



File and directory management

File content inspection

File searching

COMMANDS FOR BROWSING FILE SYSTEM



File and Directory Management (1/5)

- Use **cd** to **c**hange **d**irectory
  - **.** stands for current directory
  - **..** stands for parent directory
  - **-** stands for previous directory
  - **~** stands for user's home directory
  - **~account** stands for ***account***'s home directory (***account*** is a user)
- Use **pwd** to **p**rint **w**orking **d**irectory

```
[root@study mail]# pwd
/var/spool/mail
```

Courtesy of http://linux.vbird.org/linux_basic/0220filemanager.php#dir_opdir



File and Directory Management (2/5)

- Use **mkdir** to **make** **dir**ectory

```
[root@study ~]# mkdir [options] 目錄名稱
選項與參數：
-m ：設定檔案的權限
-p ：遞迴建立所需要的目錄

範例：請到/tmp底下嘗試建立數個新目錄看看：
[root@study ~]# cd /tmp
[root@study tmp]# mkdir test <==建立 test 目錄
[root@study tmp]# mkdir -p test1/test2/test3/test4 <==遞迴建立新目錄
```

- Use **rmdir** to **rem**ove **dir**ectory

```
[root@study ~]# rmdir [options] 目錄名稱
選項與參數：
-p ：連同『上層』『空的』目錄也一起刪除

範例：將於mkdir範例中建立的目錄(/tmp底下)刪除掉
[root@study tmp]# rmdir test <==可直接刪除
[root@study tmp]# rmdir test1 <==因為含有其他內容，所以無法刪除！
rmdir: failed to remove 'test1': Directory not empty
[root@study tmp]# rmdir -p test1/test2/test3/test4 <==加入 -p， 可順利刪除
```

Courtesy of http://linux.vbird.org/linux_basic/0220filemanager.php#dir_opdir



File and Directory Management (3/5)

- Use **ls** to **ls**t directory contents

```
[root@study ~]# ls [options] 檔名或目錄名稱
選項與參數(僅列出部分參數，其餘請使用 man ls 指令查詢)
-a ：全部的檔案，連同隱藏檔( 開頭為 . 的檔案) 一起列出來
-d ：僅列出目錄本身，而不是列出目錄內的檔案資料
-h ：將檔案容量以人類較易讀的方式(例如 GB, KB 等等)列出來
-l ：長資料串列出，包含檔案的屬性與權限等等資料

範例：將家目錄下的所有檔案列出來(含屬性與隱藏檔)
[root@study ~]# ls -al ~
total 56
dr-xr-x---.  5 root root 4096 Jun  4 19:49 .
dr-xr-xr-x. 17 root root 4096 May  4 17:56 ..
-rw-------.  1 root root 1816 May  4 17:57 anaconda-ks.cfg
-rw-------.  1 root root 6798 Jun  4 19:53 .bash_history
-rw-r--r--.  1 root root   18 Dec 29  2013 .bash_logout
-rw-r--r--.  1 root root  176 Dec 29  2013 .bash_profile
-rw-rw-rw-.  1 root root  176 Dec 29  2013 .bashrc
-rw-r--r--.  1 root root  176 Jun  3 00:04 .bashrc_test
drwx------.  4 root root   29 May  6 00:14 .cache
drwxr-xr-x.  3 root root   17 May  6 00:14 .config
# 以 . 為開頭的檔案為隱藏檔
```

Courtesy of http://linux.vbird.org/linux_basic/0220filemanager.php#dir_opdir



File and Directory Management (4/5)

- Use **cp** to **co**py files and directories

```
[root@study ~]# cp [options] 來源檔(source) 目標檔(destination)
[root@study ~]# cp [options] source1 source2 source3 .... Directory
選項與參數：(僅列出部分參數，其餘請使用 man cp 指令查詢)
-a ：相當於 -dr --preserve=all 的意思， 至於 dr 請參考下列說明
-i ：若目標檔(destination)已經存在時， 在覆蓋時會先詢問動作的進行
-r ：遞迴持續複製， 用於目錄的複製行為

範例一：將家目錄下的 .bashrc 複製到 /tmp， 並更名為 bashrc
[root@study ~]# cp ~/.bashrc /tmp/bashrc

範例二：將家目錄下資料夾 test1 test2 test3 複製到 /tmp
[root@study ~]# cp -r ~/test1 ~/test2 ~/test3 /tmp
```

- Use **rm** to **rem**ove files or directories

```
[root@study ~]# rm [options] 檔案或目錄
選項與參數：(僅列出部分參數，其餘請使用 man rm 指令查詢)
-f ：就是 force 的意思， 忽略不存在的檔案， 不會出現警告訊息
-i ：互動模式， 在刪除前會詢問使用者是否動作
-r ：遞迴刪除， 常用在刪除目錄

範例：將 /tmp 中資料夾 test1 test2 test3 刪除掉
[root@study ~]# cd /tmp
[root@study tmp]# rm -r test1 test2 test3
```

Courtesy of http://linux.vbird.org/linux_basic/0220filemanager.php#dir_opdir



File and Directory Management (5/5)

- Use **mv** to **mo**v**e** (rename) files

```
[root@study ~]# mv [options] source destination
[root@study ~]# mv [options] source1 source2 source3 .... Directory
選項與參數：(僅列出部分參數，其餘請使用 man mv 指令查詢)
-f ：force 強制的意思，如果目標檔案已經存在，不會詢問而直接覆蓋
-i ：若目標檔案 (destination) 已經存在時，就會詢問是否覆蓋
-u ：若目標檔案已經存在，且 source 比較新，才會更新 (update)

範例一：複製一檔案，建立一目錄，將檔案移動到目錄中
[root@study ~]# cd /tmp
[root@study tmp]# cp ~/.bashrc bashrc
[root@study tmp]# mkdir mvtest
[root@study tmp]# mv bashrc mvtest

範例二：將剛剛的目錄名稱更名為 mvtest2
[root@study tmp]# mv mvtest mvtest2 <== 完成更名
# 其實在 Linux 底下有名為 rename 的指令，但該指令專職進行多個檔名的
# 同時更名，並非針對單一檔名變更，與 mv 不同。請 man rename。
```



File and directory management

File content inspection

File searching

COMMAND FOR BROWSING FILE SYSTEM



File Content Inspection (1/4)

- Use **cat** to con**cat**enate files and print on the standard output

```
[root@study ~]# cat [option] file
選項與參數：(僅列出部分參數，其餘請使用 man cat 指令查詢)
-b ：列出行號，僅針對非空白行做行號顯示，空白行不標行號
-n ：列印出行號，連同空白行也會有行號，與 -b 的選項不同；
-v ：列出特殊字符

範例一：檢閱 /etc/issue 這個檔案的內容
[root@study ~]# cat /etc/issue
Ubuntu 16.04.6 LTS \n \l
```

- Use **nl** to **n**umber **l**ines of files

```
[root@study ~]# nl [option] file
選項與參數：(僅列出部分參數，其餘請使用 man nl 指令查詢)
-b ：指定行號指定的方式，主要有兩種：
   -b a ：表示不論是否為空行，也同樣列出行號(類似 cat -n)；
   -b t ：如果有空行，空的那一行不要列出行號(預設值)；
-w ：指定行號欄位的可佔用的字元數。

範例一：用 nl 列出 /etc/issue 的內容
[root@study ~]# nl /etc/issue
1 Ubuntu 16.04.6 LTS \n \l
```

Courtesy of http://linux.vbird.org/linux_basic/0220filemanager.php#file_content



File Content Inspection (2/4)

- **more** is file perusal filter for CRT
  Commands for its browsing mode
  - **space**: next page
  - **enter**: next line
  - **q**: exit more
  - **b**: front page
  - **/string**: search string

```
[root@study ~]# more /etc/manpath.config
# manpath.config
#
# This file is used by the man-db package to configure the man and cat paths.
# It is also used to provide a manpath for those without one by examining
# their PATH environment variable. For details see the manpath(5) man page.
# .....(中間省略).....

--More--(28%) <== 顯示當前瀏覽進度
```

- **less** is more flexible than **more**
  - **space**: next page
  - **[pagedown]**: next page
  - **[pageup]**: front page
  - **q**: exit less
  - **/string**: search string below current line
  - **?string**: search string above the current line

```
[root@study ~]# less /etc/manpath.config
# manpath.config
#
# This file is used by the man-db package to configure the man and cat paths.
# It is also used to provide a manpath for those without one by examining
# their PATH environment variable. For details see the manpath(5) man page.
# .....(中間省略).....
```

Courtesy of http://linux.vbird.org/linux_basic/0220filemanager.php#file_content



File Content Inspection (3/4)

- **head**
  - Dump partial file contents from the beginning of the file

```
[root@study ~]# head [-n number] file
選項與參數：
-n ：後面接數字，代表顯示幾行的意思
[root@study ~]# head /etc/man_db.conf
 # 預設將顯示十行

範例一：列印 20 行
[root@study ~]# head -n 20 /etc/man_db.conf

範例二：除了後面100行，其餘皆列印
[root@study ~]# head -n -100 /etc/man_db.conf
```

- **tail**
  - Dump partial file contents nearby the end of the file

```
[root@study ~]# tail [-n number] file
選項與參數：
-n ：後面接數字，代表顯示幾行的意思
-f ：表示持續偵測後面所接的檔名，要等到按下[ctrl]-c才會結束tail的偵測
[root@study ~]# tail /etc/man_db.conf
# 預設將顯示最後十行

範例一：列印最後 20 行
[root@study ~]# tail -n 20 /etc/man_db.conf

範例二：列出100行以後的資料
[root@study ~]# tail -n +100 /etc/man_db.conf
```

Courtesy of http://linux.vbird.org/linux_basic/0220filemanager.php#file_content



File Content Inspection (4/4)

- Use **od** to dump files in octal and other formats

```
[root@study ~]# od [-t TYPE] file
選項或參數：
 -t ：接各『類型 (TYPE)』的輸出，例如：
        a ：利用預設的字元來輸出
        c ：使用 ASCII 字元來輸出
        o[size] ：利用八進位(octal)來輸出資料，每個整數佔用 size bytes ；

範例一：請將/usr/bin/passwd的內容使用ASCII方式來展現
[root@study ~]# od -t c /usr/bin/passwd
0000000 177   E   L   F 002 001 001  \0  \0  \0  \0  \0  \0  \0  \0  \0
0000020 003  \0   >  \0 001  \0  \0  \0   0   >  \0  \0  \0  \0  \0  \0
0000040   @  \0  \0  \0  \0  \0  \0  \0 360 314  \0  \0  \0  \0  \0  \0
.....(後面省略)....
# 最左邊第一欄是以 8 進位來表示bytes數。以上面範例來說，
# 第二欄 0000020 代表開頭是第 16 個 byes (2x8) 的內容之意。

範例二：請將/etc/issue這個檔案的內容以8進位列出儲存值與ASCII的對照表
[root@study ~]# od -t oCc /etc/issue
0000000 125 142 165 156 164 165 040 061 066 056 060 064 056 066 040 114
      U   b   u   n   t   u       1   6   .   0   4   .   6       L
0000020 124 123 040 134 156 040 134 154 012 012
      T   S       \   n       \   l  \n  \n
0000032
# 如上所示，可以發現每個字元可以對應到的數值為何！要注意的是，該數值是 8 進位喔！
# 例如 S 對應的記錄數值為 123，轉成十進位：1x8^2+2x8+3=83。
```



File and directory management

File content inspection

File searching

COMMANDS FOR BROWSING FILE SYSTEM



File Searching

- **which**
  - to shows the full path of the specified shell command by searching it in the **$PATH**

```
[root@study ~]# which command

範例一：搜尋 ifconfig 這個指令的完整檔名
[root@study ~]# which ifconfig
/sbin/ifconfig

範例三：請找出 history 這個指令的完整檔名
[root@study ~]# which history
/usr/bin/which: no history in (/usr/local/sbin:/usr/local/bin:/sbin:/bin:
/usr/sbin:/usr/bin:/root/bin)
# history is a built-in BASH command (and is not within folders in $PATH)
```

- **whereis**
  - locate the binary, source, and manual page files for a given command
  - from a limited number of folders
    - e.g., /bin /sbin, /usr/share/man
  - It is faster than **find**

```
[root@study ~]# whereis [-bmsu] 檔案或目錄名
選項與參數：
-l :列出 whereis 查詢的主要目錄
-b :只找 binary 格式的檔案
-m :只找在說明檔 manual 路徑下的檔案
-s :只找 source 來源檔案
-u :搜尋不在上述三個項目當中的其他特殊檔案

範例一：請找出 ifconfig 這個檔名
[root@study ~]# whereis ifconfig
ifconfig: /sbin/ifconfig /usr/share/man/man8/ifconfig.8.gz

範例二：只找出跟 passwd 有關的『說明文件』檔名(man page)
[root@study ~]# whereis -m passwd # 只有在 man 裡面的檔名才抓出來！
passwd: /usr/share/man/man1/passwd.1.gz /usr/share/man/man5/passwd.5.gz
```



File Searching

- **locate**
  - find files by name
- **find**
  - search for files in a given directory hierarchy
- Difference between them
  - The **locate** uses **a pre-built database**
  - If database is not updated, then **locate** cannot find the files added recently
  - To sync the database, one must execute **updatedb** command
  - The **find** command is more flexible than **locate**
  - There are many ways to reduce the depth and breadth of your search and make the search itself faster

```
[root@study ~]# locate [-ir] keyword
選項與參數：
-l ：僅輸出幾行
-S ：輸出所使用的資料庫檔案的相關資訊

範例一：找出系統中所有與 passwd 相關的檔名，且只列出 3 個
[root@study ~]# locate -l 3 passwd
/etc/passwd
/etc/passwd-
/etc/pam.d/passwd

範例二：列出 locate 查詢所使用的資料庫檔案之檔名與各資料數量
[root@study ~]# locate –S
Database /var/lib/mlocate/mlocate.db:
        8,086 directories # 總紀錄目錄數
        109,605 files # 總紀錄檔案數
        5,190,295 bytes in file names
        2,349,150 bytes used to store database
```

```
[root@study ~]# find [PATH] [option] [action]
選項與參數請自行參考 man find

範例五：找出檔名為 passwd 這個檔案
[root@study ~]# find / -name passwd
/etc/passwd
...省略...

範例五-1：找出檔名包含了 passwd 這個關鍵字的檔案
[root@study ~]# find / -name *passwd*
# 利用這個 -name 可以搜尋檔名啊！預設是完整檔名，
# 如果想要找關鍵字，可以使用類似 * 的任意字元來處理
```



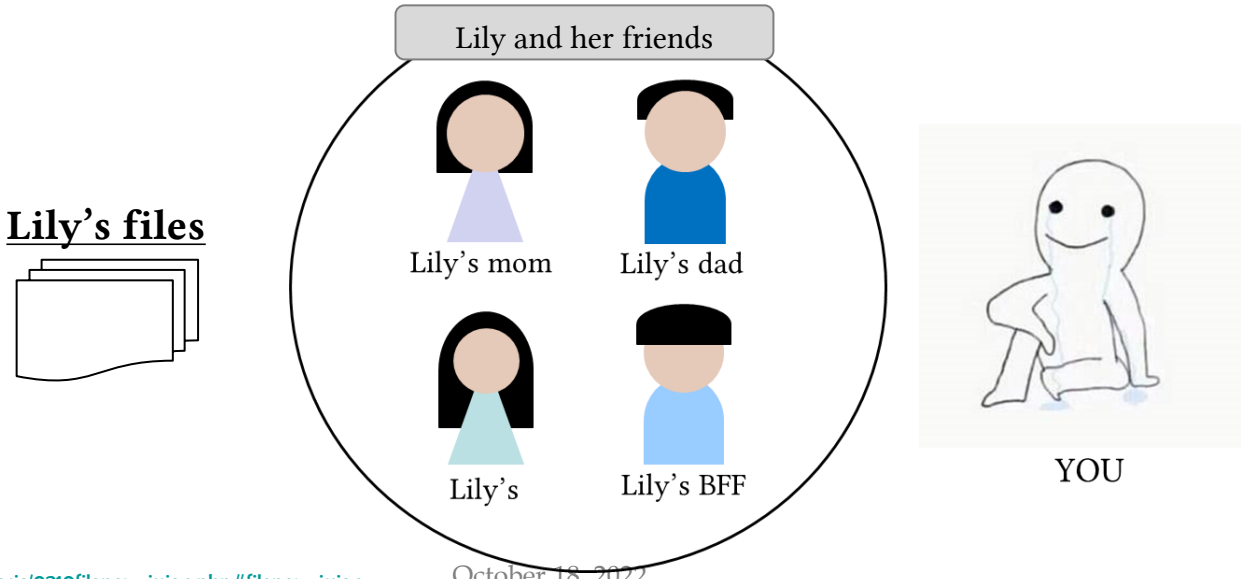
File permissions

USER AND GROUP IN LINUX



Users and Groups in Linux

- In Linux, any file may have different permissions by its "**owner**, **group,** and **others**"
- Example
  - Your friend, ***Lily***, is the ***owner*** of her files
  - Lily's family is a ***group*** that shares the same access permissions
  - And you, a stranger to Lily's world, belongs to ***others***

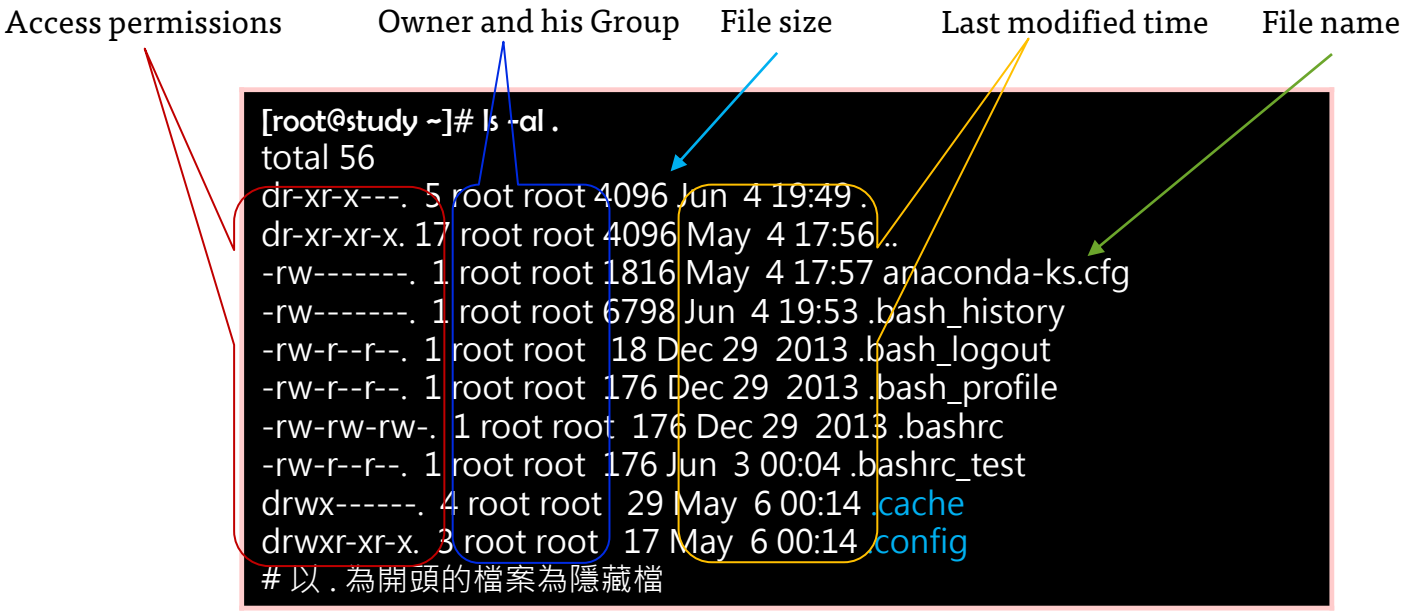


Courtesy of http://linux.vbird.org/linux_basic/0210filepermission.php#filepermission



Linux File Attribute

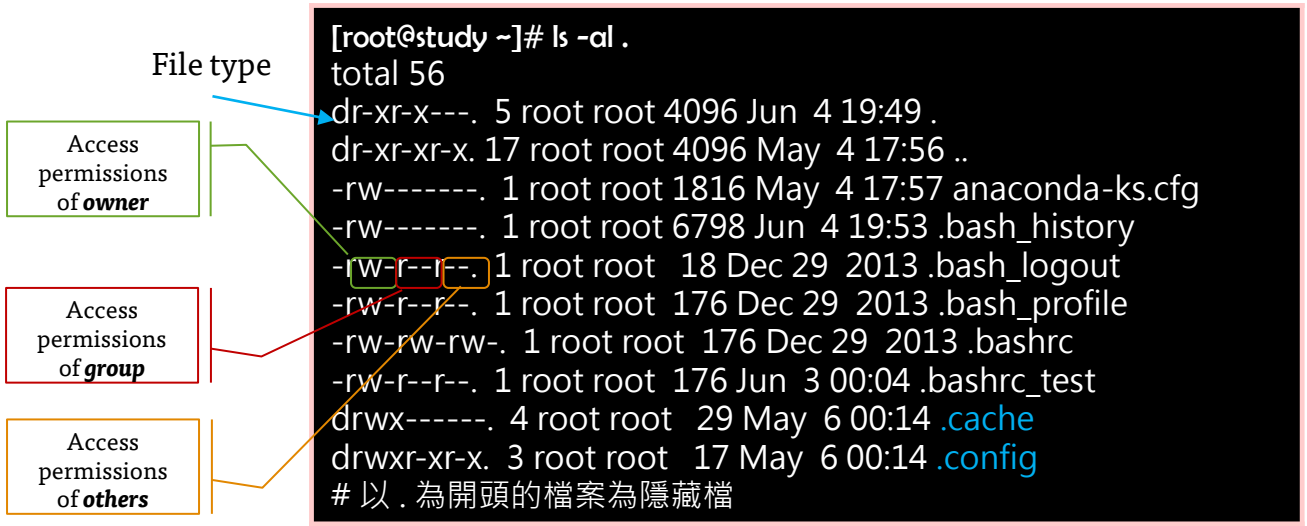
- View the details of the files with **$/> ls –al .**
  - file permissions, number of links, owner name, owner group,
  - file size, time of last modification, and file/directory name



Courtesy of http://linux.vbird.org/linux_basic/0210filepermission.php#filepermission



File Access Permission



- The fields of access permissions for a file are expressed via characters
- First character is '**-**' or '**l**' or '**d**'
  - '**d**' indicates a directory, '**-**' represents a file, '**l**' is a symlink (or soft link; special type of file)
- The following nine characters are divided into three sets, indicating the permissions for ***owner***, ***group*** and ***other***:
  - '**r**': readable, '**w**': writable, '**x**': executable, '**-**': no permission

Courtesy of http://linux.vbird.org/linux_basic/0210filepermission.php#filepermission



Special File Permissions (1/2)

- A Unix-like operating system, the ownership of files and directories is based on the default **uid** (user-id) and **gid** (group-id) of the user who created them

- setuid (Set UID; SUID; set user ID on execution), setgid (set GID; SGID)
  - Allow users to run an ***executable*** with the permissions of the executable's owner or group, respectively
    - **When an executable file's setuid permission is set, users may execute that program with a level of access that matches the user who owns the file**
  - Often used to allow users on a computer system to run programs with **temporarily elevated privileges** in order to perform a specific task
  - Example: An executable with **setuid permission set** is **passwd**, which can be use to change our login password
    -rwsr-xr-x. 1 root root 27768 Feb 11 2017 /bin/passwd
  - E.g., SUID and GUID permission looks like: **-rwsrwxr--** and **drwxrwsrwx**

Courtesy of http://linux.vbird.org/linux_basic/0210filepermission.php#filepermission
https://www.computerhope.com/jargon/s/setuid.htm
https://en.wikipedia.org/wiki/Setuid
https://linuxconfig.org/how-to-use-special-permissions-the-setuid-setgid-and-sticky-bits



Special File Permissions (2/2)

- Sticky Bit (SBIT)
  - While it has no effect on files, when used on a ***directory***, all the files in said directory will be **modifiable only by their owners**
  - Typically, this is set on the **/tmp** directory to prevent ordinary users from deleting or moving other users' files
    $/> ls -ld /tmp
    drwxrwxrwt. 14 root root 300 Nov 1 16:48 /tmp
  - The sticky bit is identifiable by a **t** which is reported where normally the executable **x** bit is shown, in the "***other***" section
  - A lowercase **t** implies that the executable bit is also present, otherwise you would see a capital **T**

Courtesy of http://linux.vbird.org/linux_basic/0210filepermission.php#filepermission
https://www.computerhope.com/jargon/s/setuid.htm
https://en.wikipedia.org/wiki/Setuid
https://linuxconfig.org/how-to-use-special-permissions-the-setuid-setgid-and-sticky-bits



User Commands

- **whoami**
  - displays the username of the current user when this command is invoked

```
[NCKU@study ~] $> whoami
NCKU
# The name of currently logged-in user
```

- **id**
  - A command-line utility that prints the real and effective user and group IDs

```
[one@study ~] $> id
uid=1000(linuxize) gid=1000(linuxize)
groups=1000(linuxize),4(adm),27(sudo),998(docker)
# The information of currently logged-in user is displayed
```

- **su**
  - Short for substitute or switch user
  - Allow you to run commands with the privileges of another user, by default the ***root*** user
  - Check **sudo**

```
[one@study ~] $> su              <==輸入指令
Password:          <==在這裡輸入 root 的密碼
[root@tsai vibrd]#       <==看，使用者名稱與提示符號#變囉！
```

Courtesy of https://linuxize.com/post/id-command-in-linux/
https://www.geeksforgeeks.org/whoami-command-linux-example/
https://linuxize.com/post/whoami-command-in-linux/
https://linuxize.com/post/su-command-in-linux/



Commands for File Permissions (1/2)

- chgrp
  - change ***group*** ownership
- chown
  - change ***file owner and group***

```
[root@study ~]# chgrp [-R] group 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、目錄都更新成為這個群組之意。常常用在變更某一目錄內所有的檔案之情況。

範例：
[root@study ~]# chgrp users initial-setup-ks.cfg
[root@study ~]# ls –l
-rw-r--r--. 1 root users 1864 May 4 18:01 initial-setup-ks.cfg
[root@study ~]# chgrp testing initial-setup-ks.cfg
chgrp: invalid group: `testing' <== 因無此群組，所以發生錯誤
```

```
[root@study ~]# chown [-R] 帳號名稱 檔案或目錄
[root@study ~]# 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 initial-setup-ks.cfg 的擁有者改為bin這個帳號：
[root@study ~]# chown bin initial-setup-ks.cfg
[root@study ~]# ls –l
-rw-r--r--. 1 bin users 1864 May 4 18:01 initial-setup-ks.cfg

範例：將 initial-setup-ks.cfg 的擁有者與群組改回為root：
[root@study ~]# chown root:root initial-setup-ks.cfg
[root@study ~]# ls -l -rw-r--r--. 1 root root 1864 May 4 18:01 initial-setup-ks.cfg
```



Commands for File Permissions (2/2)

- chmod
  - change file access permission ***bits***
  - Each bits stands for a number, **r**:4, **w**:2, **x**:1
  - Special permissions also have their number, **SUID**:4, **GUID**:2 and **SBIT**:1
  - E.g., permission **-rwxrwx---** is **770**
    - ***owner*** = rwx = 4+2+1 = 7
      group = rwx = 4+2+1 = 7
      others= --- = 0+0+0 = 0
  - E.g., permission **-rwsrwx---** is **4770** (4+770)
  - Alternatively, you can set the permissions in different ways

```
[root@study ~]# chmod [-R] xyz 檔案或目錄
選項與參數：
xyz : 權限， 為 rwx 屬性數值的相加。
-R : 遞迴(recursive)變更

[root@study ~]# cd /tmp
[root@study tmp]# touch test <==建立一個測試用空檔
[root@study tmp]# chmod 4755 test; ls -l test <==加入具有 SUID 的權限
-rwsr-xr-x 1 root root 0 Jun 16 02:53 test
```

Courtesy of https://www.thegeekdiary.com/linux-interview-questions-special-permissions-suid-sgid-and-sticky-bit/



THANK YOU!