

Chương 5: QUẢN TRỊ HỆ THỐNG (tiếp theo)

5.3. Thông tin tiến trình, hệ thống tập tin /proc

- Nhân hệ điều hành Linux có hai nhiệm vụ chính:
 - Điều khiển truy nhập tới các thiết bị vật lý trên máy tính
 - Lập kế hoạch về thời điểm và cách thức các tiến trình tương tác với thiết bị này.
- Thư mục **/proc/** có chứa các tập tin đặc biệt đại diện cho trạng thái hiện hành của nhân hệ điều hành.

- Trong Linux, tất cả dữ liệu đều được lưu dưới dạng những tập tin. Ngoài các kiểu tập tin thông thường (text, nhị phân), thư mục **/proc/** có chứa một kiểu tập tin khác được gọi là tập tin giả. Do đó, thư mục **/proc/** thường được gọi là một hệ thống tập tin giả (**process information pseudo-file system**)
- Những tập tin giả này có đặc tính riêng biệt. Phần lớn các tập tin này có kích thước bằng 0, nhưng chúng có chứa một số lượng lớn thông tin.

- **Xem nội dung tập tin giả**

- Để xem thông tin về hệ thống, ta có thể sử dụng các lệnh **cat**, **more** hay **less** để xem các tập tin trong thư mục **/proc/**
- Một số tập tin thuộc về các tiện ích đặc biệt (như top, free...) ta không thể đọc được kết quả hiển thị ra màn hình do chúng được định dạng đặc biệt theo yêu cầu của ứng dụng.
- Phần lớn các tập tin trong thư mục **/proc/** chỉ cho phép đọc. Ngoại trừ một số tập tin trong **/proc/sys/** có thể được thay đổi để hiệu chỉnh các thiết lập trong nhân hệ điều hành.

- **Vai trò, ý nghĩa của một số thư mục tập tin trong /proc/**

- **/proc/cpuinfo**: Tập tin này cho biết kiểu của bộ xử lý trên hệ thống.
- **/proc/devices**: Hiển thị các character devices và các block devices được cấu hình cho nhân hệ điều hành sử dụng.
- **/proc/filesystems**: Hiển thị danh sách các kiểu file system được nhân hệ điều hành hỗ trợ

- **/proc/iomem**: ánh xạ hiện hành của bộ nhớ hệ thống cho mỗi thiết bị vật lý.
- **/proc/meminfo**: Hiển thị thông tin về tình trạng sử dụng bộ nhớ hiện hành trên hệ thống.
- **/proc/mounts**: Liệt kê danh sách tất cả các mountpoint đang có trên hệ thống.
- **/proc/process_dir**: Là các thư mục có tên là con số. Những thư mục này có tên là thư mục tiến trình, chúng được đặt tên theo mã số tiến trình và có chứa các thông tin xác định cho tiến trình đó. Khi tiến trình kết thúc, các thư mục tiến trình tương ứng sẽ bị xóa.

5.4. Quản lý đĩa và file system

- Linux tham chiếu đến các partition trên đĩa cứng sử dụng một sự kết hợp của các chữ cái và các con số
- Các partition được đặt tên linh hoạt và mang nhiều thông tin hơn so với phương pháp sử dụng bởi các hệ điều hành khác.
- Mọi partition trong Linux đều được tham chiếu qua các tập tin, tên các tập tin này được đặt như sau:

/dev/xxyN• **Trong đó**

- **/dev/**: Là tên thư mục chứa tất cả các tập tin thiết bị. Do các partition nằm trên đĩa cứng và đĩa cứng là một loại thiết bị nên tất cả các tập tin đại diện cho các partition sẽ nằm trong **/dev/**.
- **xx**: Hai chữ cái đầu tiên của tên partition, chỉ ra kiểu thiết bị của partition. Một số thiết bị thông thường bao gồm:

- **hd**: Ổ đĩa cứng IDE
- **sd**: Ổ đĩa cứng SCSI
- **sr**: Ổ đĩa CD-ROM SCSI
- **fd**: Ổ đĩa mềm
- **y**: Xác định thiết bị mà partition nằm trên đó:
 - **a**: là ổ đĩa cứng thứ nhất, master
 - **b**: là ổ đĩa cứng thứ nhất, slave
 - **c**: là ổ đĩa cứng thứ hai, master
 - **d**: là ổ đĩa cứng thứ hai, slave...
- **N**: Là một con số biểu diễn loại partition. Các số từ 1-4 xác định 4 partition đầu tiên (primary hay extended). Các partition logic được đánh số từ 5 trở lên. Ví dụ /dev/hda3, /dev/sdb6...

5.5. Gắn kết hệ thống tập tin

5.5.1. Lệnh mount và umount

- Để truy xuất được tập tin/thư mục nằm ở partition hệ thống khác hay sử dụng hệ thống tập tin khác ta cần phải thực hiện “**gắn kết**”- **mount** các phần chia, các hệ thống tập tin khác này vào một thư mục trong hệ thống của Linux.
- Nội dung của thư mục đó sẽ đại diện cho đối tượng được **mount** nghĩa là truy xuất đến thư mục này là truy xuất đến thiết bị.

- Để thực hiện **mount** một thiết bị vào hệ thống tại thư mục **directory** (điểm treo-điểm gắn kết-mount point) ta sử dụng lệnh **mount** như sau:
- **Cú pháp**
mount devicefile directory
devicefile là đường dẫn đến file thiết bị (thông thường lưu trong thư mục **/dev**). Linux qui định ổ đĩa A là file thiết bị **/dev/fd0**, CD-ROM là **/dev/cdrom**, các partition là **/dev/hda1**, **/dev/hda2...**

- **Chú ý:**

- Chỉ có root mới thực hiện được lệnh mount.
- **Directory-mountpoint** phải được tạo ra trước khi mount và có thể được tạo ra ở bất kỳ vị trí nào trên hệ thống.

- **Ví dụ:**

mount /dev/sdb1 /diaUSB

mount /dev/cdrom /CDROM

- **/mnt/sub_dir** thường là **mountpoint** mặc định (qui định trong file **/etc/fstab**) của các thiết bị thông thường như CD-ROM(cdrom), USB(flash), floppyDisk(floppy)...

- **Ví dụ:**

mount /dev/cdrom

mount /dev/sdb1

mount /dev/floppy

- Muốn xóa gắn kết một thiết bị vào hệ thống ta dùng lệnh **umount** và khi đó ta phải đứng ngoài thư mục gắn kết
- **Cú pháp**
umount devicefile
umount mountpoint
umount -a
(xóa tất cả các gắn kết của các file system)

5.5.2. Tự động gắn kết-Tập tin **/etc/fstab**

- Để thực hiện tự động gắn kết một file system khi khởi động hệ thống Linux thì ta có thể khai báo các thông tin gắn kết vào tập tin cấu hình **/etc/fstab**.
- Tập tin này chứa các thông tin mô tả các file system khác nhau. Mỗi file system được mô tả trên một dòng và có 6 trường.
- Các trường trên mỗi dòng cách nhau bằng các khoảng trắng hay ký tự tab, theo định dạng sau:

Khoa Công nghệ Thông tin Trường Đại học Nông lâm TP. Hồ Chí Minh

```

fstab [----] 52 L: [ 1+ 8 9/ 9] *(646 / 656b)= . 9 0x09
LABEL=/ / ext3 defaults 1 1
LABEL=/boot /boot ext3 defaults 1 2
none /dev/pts devpts gid=5,mode=620 0 0
none /proc proc defaults 0 0
none /dev/shm tmpfs defaults 0 0
/dev/sda3 swap swap defaults 0 0
/dev/cdrom /mnt/cdrom udf,iso9660 noauto,owner,kudzu,r
/dev/fd0 /mnt/floppy auto noauto,owner,kudzu 0 0
/dev/sdb1 /mnt/usb auto_ user 0 0

```

1 Help 2 Save 3 Mark 4 Replace 5 Copy 6 Move 7 Search 8 Delete 9 PullDn 10 Quit

Bài giảng môn học Nhập Môn Hệ Điều Hành 15

Khoa Công nghệ Thông tin Trường Đại học Nông lâm TP. Hồ Chí Minh

device mountpoint fstype option dump fsck

- **device**: tên file system cần gắn kết
- **mountpoint**: Điểm mà file system gắn kết vào
- **fstype**: Mô tả kiểu file system
 - **ext2**: file system Linux
 - **ext3**: file system Linux
 - **swap**: partition swap của Linux
 - **iso9660**: file system của CD-ROM
 - **vfat**: file system của window 32 bit
 - **msdos**: file system msdos

Bài giảng môn học Nhập Môn Hệ Điều Hành 16

- **options**: Mô tả các lựa chọn của lệnh mount đối với file system. Các lựa chọn cách nhau dấu phẩy. Trường này có ít nhất một lựa chọn. Một số lựa chọn hay dùng bao gồm
 - **auto**: Tự động gắn kết khi khởi động hệ thống
 - **noauto**: Không tự động gắn kết khi khởi động hệ thống.
 - **ro**: Gắn kết và cấp quyền chỉ đọc
 - **rw**: Gắn kết và cấp quyền đọc ghi
 - **user**: Cho phép người dùng thường gắn kết file system
 - **default**: Sử dụng các giá trị mặc định: **rw**, **auto**, **nouser**

- Hai trường sau cùng có chứa một giá trị nguyên.
 - Giá trị đầu được sử dụng bởi lệnh **dump** để xác định xem hệ thống tập tin có chứa các tập tin thực và có được sao lưu hay không. Trường này không thường được sử dụng.
 - Giá trị sau cùng xác định thứ tự mà lệnh **fsck** sẽ thực hiện trên các partition của hệ thống khi máy tính khởi động. Partition / cần được kiểm tra đầu tiên với giá trị **1**, và tất cả các partition khác cần được kiểm tra ngay sau đó. Bằng cách gán trị **2**, ta có thể đảm bảo rằng các partition trên các đĩa cứng khác có thể được kiểm tra song song. Giá trị **0** thông báo cho **fsck** không cần kiểm tra hệ thống tập tin.

5.6. Kiểm tra và khôi phục file system Linux

- Các file system có cấu trúc phức tạp và vì vậy chúng hay xảy ra lỗi.
- Các lỗi này thường xảy ra khi hệ điều hành bị ngưng hoạt động trước khi **umount file system**.
- Để sửa lỗi cho file system ta dùng lệnh **fsck**. Lệnh **fsck** sẽ quét file system, tìm và sửa các lỗi (nếu có) sau đó thông báo kết quả thực hiện.
- **Cú pháp**
fsck [option] fileSystem

Trong đó **fileSystem** có thể là tên của một partion . Ví dụ:
/dev/hda1...

- **Option**
 - a: Tự động khôi phục lại file system mà không cần xác nhận từ người dùng.
 - r: Yêu cầu xác nhận từ người dùng trước khi thực hiện mỗi thao tác khôi phục file system.
 - t **type**: Chỉ ra kiểu của file system được kiểm tra.
- **Chú ý**: Không thi hành lệnh **fsck** trên một file system đã được mount vào hệ thống, ngoại trừ trường hợp root ở trạng thái chỉ đọc trong quá trình khởi động.

5.7. Tạo và định dạng file system

5.7.1. Tạo partition

- Để có thể thay đổi các partition trên ổ đĩa hiện hành hay sử dụng một ổ đĩa cứng mới, ta phải tạo các partition bằng lệnh **fdisk** và sau đó phải định dạng partition mới tạo bằng lệnh **mkfs**.

- **Cú pháp**

fdisk -l [device]... (1)

fdisk [device] (2)

- (1): Liệt kê các partition của các thiết bị device chỉ ra **device**. Nếu không chỉ ra **device**, sẽ hiển thị thông tin của tất cả các partition (có trong tập tin **/proc/partitions**)
- (2): Dùng để tạo một partition mới. Trong đó **device** là ổ đĩa muốn tạo partition mới hay partition muốn tạo lại. Lệnh này còn cho phép tạo mới, xóa, thay đổi mã nhận diện (ID) của các partition trên ổ đĩa device, tương tự như thao tác thực hiện tạo các partition trong quá trình cài đặt.

– Khi sử dụng lệnh `fdisk`, Linux xuất hiện lời nhắc:

Command (m for help):

Có thể nhập các lệnh cho **fdisk** thực hiện như sau:

- **m**: danh sách các lệnh của **fdisk**
- **p**: Hiển thị thông tin về các partition hiện hành
- **q**: Thoát lệnh `fdisk` mà không ghi lại các thay đổi

- **w**: Ghi lại các thay đổi và thoát
- **v**: Kiểm tra bảng partition
- **d**: Xóa một partition
- **n**: Tạo partition mới
- **t**: Chọn kiểu file system cho một partition
- **a**: Thiết lập/ loại bỏ cờ khởi động boot cho một partition.
- **l**: Liệt kê danh sách kiểu partition và mã số của chúng.

5.7.2. Định dạng partition

- Khi tạo xong một partition ta cần phải định dạng (format) lại chúng bằng lệnh **mkfs**
- **Cú pháp**
mkfs [option] fileSystem
 Trong đó **fileSystem** là tên partition cần định dạng
- **Option**
 - t **type**: Chỉ ra kiểu file system cần định dạng. Trường hợp không xác định kiểu file system muốn định dạng, **mkfs** sử dụng giá trị mặc định là ext2.

- c: Kiểm tra những block hỏng của partition trước khi định dạng.
- **Chú ý**: Lệnh **mkfs** chỉ được sử dụng để định dạng các partition chưa được gắn kết.

5.8. Thông tin về tình trạng sử dụng đĩa

5.8.1. Lệnh df

- Để tìm xem không gian còn trống trên một file system ta sử dụng lệnh **df**.

- **Cú pháp**

df [-m] [fileSystem]

Trong đó **fileSystem** là partition muốn kiểm tra dung lượng đĩa còn trống

-m: hiển thị kích thước theo Megabyte

Trường hợp không chỉ ra **fileSystem** thì **df** sẽ cho biết tình trạng sử dụng đĩa của tất cả các partition của hệ thống.

5.8.2. Lệnh du

- Thực hiện tính toán mức chiếm dụng đĩa của các tập tin.

- **Cú pháp**

du [option] [directory]

Trong đó **directory** thư mục cần xem. Nếu không có tham số này lệnh **du** sẽ tính toán mức chiếm dụng đĩa của thư mục hiện hành.

- **Option**

- a: Thực hiện tính mức chiếm dụng đĩa của từng tập tin có trong thư mục được yêu cầu.
- m: Hiển thị kích thước theo Megabyte
- b: Hiển thị kích thước theo byte
- s: Không thực hiện tính toán kích thước của các thư mục con
- c: Hiển thị thông tin mức chiếm dụng đĩa tổng cộng.

5.9. Tạo đĩa mềm khởi động (Đọc thêm)

- Đĩa mềm khởi động hệ điều hành Linux có thể
 - khởi động hệ thống trong trường hợp có hư hỏng
 - Kiểm tra một nhân hệ điều hành mới và có thể giúp khởi động hệ điều hành Linux trong trường hợp máy tính sử dụng nhiều hệ điều hành.
- Đĩa mềm khởi động hệ thống có thể tạo khi cài đặt Linux và cũng có thể được tạo bằng lệnh **mkbootdisk** như sau

- **Cú pháp**

mkbootdisk [--device devicefile] kernel

Trong đó

--device devicefile: Chỉ ra thiết bị sẽ tạo ảnh khởi động trên đó. Nếu **--device** không được chỉ ra, mặc định **/dev/fd0** sẽ được sử dụng.

kernel: Xác định phiên bản nhân hệ điều hành muốn tạo đĩa khởi động

(**chú ý**: nhân hệ điều hành được xác định bằng lệnh **uname -r**)

Ví dụ:

mkbootdisk --device /dev/fd0 2.4.2-2

5.10. Hiện thị/ thay đổi ngày giờ hệ thống

- Để hiện thị hay thay đổi ngày giờ hệ thống ta dùng lệnh **date**.

- **Cú pháp**

date [MMDDhhmmYY]