

Hệ điều hành mạng

Nội dung: cài đặt, vận hành, bảo trì hệ thống Linux

Tư cách: 3+ (bài tập thực hành, trắc nghiệm)

Kết thúc học phần: trắc nghiệm (được sử dụng tài liệu - viết tay)

Câu hỏi?

Phải cài đặt hệ thống Linux

- + Cài đặt trên máy ảo: VMware, Vbox, Hyper-V, ...
- + Cài trên máy thật.

Hệ thống Linux là hệ thống sử dụng nhân Linux (Linux kernel).

Bản phân phối (Distribution): sự kết hợp của nhân Linux và các công cụ, chương trình đi kèm để hình thành hệ điều hành. Có một số bản phân phối phổ biến:

- + Ubuntu, Debian
- + Fedora, CentOS, RedHat Enterprise Linux (RHEL)
- + OpenSUSE, SUSE Enterprise Linux Server (SLES)
- + Slackware

Một số quy ước khi minh họa chế độ dòng lệnh.

Với dấu nhắc lệnh bắt đầu bằng:

- > Ở chế độ người dùng thông thường
- # Ở chế độ siêu người dùng (root)

Cửa sổ dòng lệnh thường có tên gọi: Console, Konsole, Terminal

Chương trình sử dụng dòng lệnh: CLI (Command Line Interface)

Yêu cầu chuẩn bị (về nhà):

- + Cài đặt một bản phân phối/hệ điều hành Linux

Một số giao diện đồ họa phổ biến

- + GNOME
- + KDE
- + Xfce: rất nhẹ

Với chế độ dòng lệnh:

- + RAM: 512MB+

Với chế độ đồ họa:

- + RAM: 1GB+

Tổng quan về hệ thống Linux

+ Hệ thống tệp tin (filesystems): là cách dữ liệu được tổ chức trong hệ điều hành. Nó giúp người dùng có thể tương tác (truy cập, thay đổi, tác động) với dữ liệu.

+ Trong Linux thì có một số định dạng filesystems được hỗ trợ tối ưu cho hoạt động của nó:

- ext3: Hệ thống tệp tin được sử dụng rất phổ biến.
- ext4: Là phiên bản kế tiếp của ext3 với các cải thiện về hiệu năng.
- btrfs: Đây là một định dạng tương đối mới dành cho Linux. Đặc trưng hệ thống tệp tin này là nó cho phép snapshot (ghi lại lịch sử thay đổi của dữ liệu) trên phân vùng được định dạng btrfs dựa vào cơ chế copy-on-write.
- zfs: là định dạng vốn phát triển hệ thống Unix, nhưng nó tương thích tốt với hệ thống Linux.
- xfs: là định dạng hiệu năng cao dành cho Linux, đặc biệt là các tệp tin có dung lượng lớn (VD: cơ sở dữ liệu).

+ Ngoài ra Linux vẫn hỗ trợ các định dạng vốn phát triển cho MS-DOS, Windows (chỉ để chứa dữ liệu chứ không cài đặt hệ thống Linux vào được):

- FAT: là định dạng phát triển cho MS-DOS
- FAT32: là định dạng mở rộng FAT cho phép hoạt động các hệ thống sử dụng cluster size 32bit. Hỗ trợ dung lượng tệp tin tối đa là 4GB.
- exFAT: là định dạng mở rộng tiếp theo của FAT thường dùng trên các thiết bị lưu trữ ngoài (VD: thẻ nhớ). exFAT cho phép lưu trữ tệp tin lớn hơn 4GB.
- NTFS: đây là định dạng phổ biến được sử dụng trên Window hiện nay.

+ Lưu ý về hệ thống tệp tin trong Linux:

- Trong Linux thì tên tệp tin/đường dẫn phân biệt chữ hoa và chữ thường.
- Đa phần các hệ thống tệp tin trong Linux đa phần có độ dài giới hạn ở mức tối thiểu là 255 bytes.
- Tuy nhiên chiều dài đường dẫn về lý thuyết thường không giới hạn.
- Trong Linux các ký tự thể hiện văn bản (các ngôn ngữ khác nhau) đều có thể dùng để đặt tên trừ ký tự /

+ Cấu trúc hệ thống tệp tin trong Linux:

- Hệ thống tệp tin được tổ chức trong cây thư mục. Các nút gọi là inode.
- Có một thư mục gọi là thư mục gốc/root directory (chính là gốc của cây thư mục). Và được ký hiệu là /
- Các phân vùng ổ đĩa sẽ được ánh xạ (mount) vào cây thư mục.
- Một số thư mục và mục đích của nó:
 - + /boot - là thư mục chứa các thành phần để khởi động hệ thống (bootloader, kernel)

- + /bin - chứa các chương trình cần thực thi.
- + /lib - chứa các thư viện.
- + /usr - chứa các thành phần liên quan đến người dùng.
- + /var - chứa các dữ liệu thường xuyên thay đổi.
- + /etc - chứa các tệp tin cấu hình cho hệ thống và phần mềm.
- + /home - chứa dữ liệu của các người dùng.
- + /proc - chứa các inode để tương tác với các tiến trình (process).
- + /dev - chứa các inode để tương tác với thiết bị phần cứng (chủ yếu là các thiết bị ngoại vi).
- + Đường dẫn tới tệp tin/thư mục:
 - + Đường dẫn tuyệt đối: là đường dẫn bắt đầu từ thư mục gốc (bắt đầu bằng /)
Ví dụ: /home/userA/thong-tin.txt
 - + Đường dẫn tương đối: là đường dẫn bắt đầu từ thư mục hiện tại (thư mục đang làm việc - working directory).
Ví dụ: ../thong-tin.txt
 thong-tin.txt
- Trong đường dẫn tương đối có một số thư mục đặc biệt mang tính đại diện:
 - . (dấu chấm) đại diện cho thư mục hiện tại.
 - .. (hai dấu chấm viết liền nhau) đại diện cho thư mục cha.
- Ví dụ:
../../thong-tin.txt
- + Đối với các đường dẫn/tên tệp tin có ký tự đặc biệt thì cần escape
 - Cách 1: Đối với ký tự khoảng trắng/space, hoặc các ký tự khác thì dùng \ ở phía trước nó.
 - Cách 2: Có thể cho đường dẫn vào trong "path"
- + Ví dụ:
"./thông tin.txt"
./thông\ tin.txt
- + Chú ý:
Các tệp tin hoặc thư mục có tên bắt đầu bằng dấu chấm '.' được coi là tệp tin/thư mục ẩn.

Các lệnh thao tác với tệp tin và thư mục

Xem thư mục hiện thành (thư mục đang làm việc - working directory)

- + Cú pháp: pwd

Thay đổi thư mục làm việc:

- + Cú pháp: cd đường_dẫn

- + Ví dụ:

```
cd bai-tap
cd /home/userA
cd ..
cd ../../
```

Xem danh sách các tệp tin và thư mục bên trong đường dẫn

- + Cú pháp: ls [tùy_chọn] [đường_dẫn]

- + Một số tùy chọn thường dùng:

- l hiện thông tin chi tiết (có thông tin phân quyền, người dùng, dung lượng tệp tin, thời gian)
- a hiện thị cả tệp tin ẩn (tệp tin bắt đầu bằng .)
- A hiện thị tệp tin ẩn (nhưng không hiện . và ..)
- h hiện thị dung lượng ở dạng 1M, 4K, ...
- R hiện thị một cách đệ quy (hiển thị danh sách tệp tin và thư mục của các thư mục con).
- Z hiện thị thông tin về SELinux.
- i hiện thị inode number.

- + Nếu không có "đường_dẫn" lệnh ls sẽ thực hiện với thư mục hiện tại/thư mục làm việc.

- + Ví dụ:

```
ls
ls -l bai-tap
ls -la bai-tap
ls -l -a bai-tap
ls -lh
```

Tạo thư mục:

- + Cú pháp: mkdir [tùy_chọn] đường_dẫn

- + Tùy chọn hay dùng

- p tạo cả cây thư mục (tạo thư mục đệ quy)

- + Ví dụ:

```
mkdir tuan-3
mkdir tuan-3/bai-tap
mkdir -p danh-sach/he-dieu-hanh-mang
```

Di chuyển/đổi tên tệp tin hoặc thư mục:

- + Cú pháp: mv đường_dẫn_nguồn đường_dẫn_đích

- + Ví dụ:

```
mv tuan-3 week-3
mv danh-sach week-3/
```

Xóa tệp tin/thư mục:

- + Cú pháp: `rm [tùy_chọn] đường_dẫn`
- + Một số tùy chọn thường dùng
 - r thực hiện thao tác xóa một cách đệ quy (xóa cả thư mục)
 - f thực hiện thao tác xóa một cách bắt buộc (force)
- + Ví dụ:
 - `rm tep-tin.txt`
 - `rm -r week-3`
 - `rm -rf week-3`
 - `rmdir week-3` (lệnh này chỉ xóa thư mục week-3 nếu nó rỗng)

Thao tác với tệp tin:

- + Tạo tệp tin không có nội dung
 - `touch tên_tệp_tin`
- + Chỉnh sửa/tạo tệp tin với nội dung sử dụng các công cụ soạn thảo như: nano, vi
 - Ví dụ: `nano tên_tệp_tin`
 - `Ctrl + X` : thoát khỏi nano
 - `Ctrl + O` : lưu tệp tin trong nano
- + Lưu nội dung từ một lệnh vào tệp tin
 - `lệnh > tên_tệp_tin` : lưu nội dung được in ra stdout bởi "lệnh" vào trong "tên_tệp_tin" (đồng thời xóa nội dung cũ của tệp tin trên).
 - `lệnh >> tên_tệp_tin` : thêm phần nội dung được in ra stdout bởi "lệnh" vào trong "tên_tệp_tin".

Ví dụ: `ls >> tep-tin.txt`
`ls > tep-tin.txt`

Lưu ý: Nếu muốn ghi cả stderr vào trong "tên_tệp_tin" thêm `2>&1` vào cuối lệnh ở trên.
Ví dụ: `rmdir thu >> tep-tin.txt 2>&1`

- + Xem nội dung tệp tin
 - Cú pháp: `cat tên_tệp_tin`
 - Ví dụ: `cat tep-tin.txt`
- + Xem phần cuối của tệp tin
 - Cú pháp: `tail [tùy_chọn] tên_tệp_tin`
 - Tùy chọn:
 - n num hiển thị num dòng cuối cùng của tệp tin (num mặc định là 10)
 - f cập nhật khi tệp tin thay đổi (thường dùng khi theo dõi lịch sử - logs)

Ví dụ:
`tail tep-tin.txt`
`tail -n15 tep-tin.txt`
`tail -n15 -f tep-tin.txt`
`tail -fn15 tep-tin.txt`

Lưu ý: Khi sử dụng -f dùng tổ hợp phím `Ctrl + C` để dừng việc theo dõi sự thay đổi tệp tin lại (dừng lệnh đang chạy).

- + Xem phần đầu của tệp tin
 - Cú pháp: `head [tùy_chọn] tên_tệp_tin`
 - Tùy chọn:
 - n num hiển thị num dòng đầu tiên của tệp tin (num mặc định là 10)

Ví dụ:
`head -n3 tep-tin.txt`
`head -n 25 tep-tin.txt`

Sao chép tệp tin/thư mục:

- Cú pháp: `cp [tùy_chọn] nguồn_đích`
- Một số tùy chọn:
 - r, -R thực hiện thao tác sao chép đệ quy (sao chép cả thư mục)

Ví dụ:
`cp tep-tin.txt tep-tin.bak`
`cp -r bai-tap bai-tap.bak`

Liên kết (link):

- + Phân loại:
 - Liên kết cứng (hard link): liên kết đến một inode cụ thể (chỉ áp dụng được đối với tệp tin). Một tệp tin có thể có 1 hoặc nhiều liên kết cứng. Nếu tất cả các liên kết cứng đến tệp tin bị xóa thì tệp tin mới thực sự bị xóa.
 - Liên kết tượng trưng (symbolic link): là dạng chỉ dẫn đến địa chỉ/đường dẫn thật sự của tệp tin hoặc thư mục. Loại liên kết này có thể chỉ dẫn đến đường dẫn không tồn tại hoặc đã bị xóa.

+ Các tạo liên kết:

Cú pháp: `ln [tùy_chọn] <nguồn> <đích>`

Một số tùy chọn:

-s tạo liên kết tượng trưng (symbolic link)

Nếu không có tham số -s thì liên kết cứng sẽ được tạo ra (hard link)

Ví dụ:

`ln -s today hom-nay`

`ln today now`

`ln -s data du-lieu`

+ Về cơ bản các thao tác đối với liên kết tương tự như đối với đường dẫn bình thường.

Tìm kiếm:

Cú pháp: `find [đường_dẫn] [tùy_chọn] [biểu_thức]`

Các tùy chọn thường dùng:

-name kiểm tra tên tệp tin/thư mục (filesystem) thỏa mãn biểu_thức.

-iname tương tự như -name nhưng không phân biệt chữ hoa và chữ thường

-type f/d/c/l/b loại tệp tin/thư mục/character device/link (liên kết)/block device

-regex pattern sử dụng biểu thức regular expression

// Các tham số kiểm tra thời gian truy cập

-atime [+/-]n

-amin [+/-]n

-anewer filename

// Các tham số kiểm tra thời gian chỉnh sửa (nội dung)

-mtime

-mmin

-mnewer filename

// Các tham số kiểm tra thay đổi về các thuộc tính tệp tin.

-ctime

-cmin

-cnewer filename

Đối với -atime, -ctime, -mtime thì thời gian tính là n*24 giờ (n ngày).

Đối với -amin, -cmin, -mmin thì thời gian tính là n phút.

-size [+/-]n[kMG] tìm kiếm theo kích thước tệp tin/thư mục

-inum inode_number tìm kiếm tệp tin có giá trị inode_number tương ứng.

-samefile filename tìm kiếm tệp tin trỏ đến cùng inode với "filename"

-maxdepth levels giới hạn độ sâu tìm kiếm tối đa

-mindepth levels độ sâu tìm kiếm tối thiểu

Ví dụ:

`find -name "*.txt"`

`find -iname "*o*"`

`find -name "???"`

`find -type f -iname "*o*"`

`find -ctime +5`

`find ISOs -size +1G`

`find /etc -iname "os-*"`

`find -samefile today`

`find -maxdepth 1 -mtime +5`

`find -mindepth 2 -mtime +5`

Lọc dữ liệu

Cú pháp: `<lệnh> | grep [tùy_chọn] <biểu_thức_lọc>`

Tùy chọn thường dùng:

-i không phân biệt chữ hoa và chữ thường

-E hỗ trợ regular expression (tương đương dùng egrep)

Ví dụ:

`ls -la | grep today`

`ls -la | egrep 't(o|a)day'`

`ls -la | grep -E 'tuan-[:digit:]'`

Xem đường dẫn tới một lệnh:

Cú pháp:

`whereis <tên_lệnh>`

which <tên_lệnh>

Ví dụ:

```
whereis ls
which ls
```

Một số lệnh thao tác chung:

Xóa màn hình

Cú pháp: clear

Phân trang (hiển thị kết quả của những lệnh có nhiều thông tin)

Cú pháp: lệnh_trước | more

Ví dụ: ls -la | more

```
cat tep-tin.txt | more
```

```
ls -la /etc | more
```

Dùng các ký tự điều hướng/enter để hiển thị tiếp các dòng ở phía sau.

Dùng phím 'q' để dừng việc xem nội dung lại.

Lưu ý: có thể thay "more" thành "less"

Xem hướng dẫn sử dụng một lệnh

Cú pháp: <tên_lệnh> -h

<tên_lệnh> --help

man <tên_lệnh>

Ví dụ:

```
ls --help
```

```
man ls
```

Người dùng và các vấn đề liên quan đến phân quyền:

+ Hệ thống Linux là hệ thống đa nhiệm và đa người dùng.

+ Một người dùng thuộc một nhóm chính và có thể thuộc nhiều nhóm mở rộng khác (auxiliary group).

+ Trong hệ thống Linux có người dùng đặc biệt gọi là người dùng "root" có id = 0. Đây là người dùng đặc biệt có nhiều đặc quyền để tác động tới hệ thống cũng như thao tác với người dùng, nhóm người dùng, phân quyền.

+ Người dùng thông thường sẽ có id từ 1000 trở đi.

+ Các quyền đối với tệp tin/thư mục:

+ Quyền đọc ("r")

+ Quyền ghi ("w")

+ Quyền thực thi ("x")

+ Đối với tệp tin thông thường thì quyền thực thi là cho phép chương trình có thể chạy (thực thi).

+ Đối với thư mục thì quyền thực thi cho phép thực hiện thao tác xem danh sách thư mục/tệp tin bên trong (thực hiện được lệnh "ls").

+ Đối với một tệp tin/thư mục các quyền được chia thành 3 phần:

+ Phần đầu tiên để quy định các quyền đối với chủ sở hữu.

+ Phần tiếp theo để quy định các quyền đối với nhóm người dùng của tệp tin/thư mục.

+ Phần còn lại quy định các quyền đối với những người dùng khác (không phải là chủ sở hữu và cũng không thuộc nhóm người dùng).

+ Lưu ý: một tệp tin/thư mục chỉ có đúng một chủ sở hữu và thuộc về đúng một nhóm người dùng.

+ Cách ký hiệu về phân quyền người dùng dùng:

Loại quyền	Dạng chữ	Dạng số
Đọc	r	4
Ghi	w	2
Thực thi	x	1

Ví dụ:

```
rw-r--r--
```

Chủ sở hữu có quyền:

Đọc: r

Ghi: w

Người dùng thuộc nhóm người dùng và các người dùng khác có quyền:

Đọc: r

Biểu diễn dưới dạng số:

```
644
```

Quyền biểu diễn dưới dạng số:

```
755
```

Tương ứng với dạng chữ:

```
rw-r--r--
```

Chủ sở hữu có quyền:

Đọc: r

Ghi: w

Thực thi: x

Người dùng không phải chủ sở hữu có quyền

Đọc: r

Thực thi: x

Các lệnh thao tác liên quan đến việc phân quyền.

+ Thay đổi quyền đối với tệp tin/thư mục:

Cú pháp: `chmod [tùy_chọn] <quyền> <đường_dẫn>`

Tùy chọn hay dùng:

-R thực hiện với cả các tệp tin và thư mục bên trong (thực hiện đệ quy).

Cách viết <quyền> có thể dạng số hoặc dạng chữ hoặc mô tả sự thay đổi về quyền.

Xác định nhóm quyền cần thay đổi.

u : chủ sở hữu

g : nhóm người dùng

o : những người dùng còn lại

a : tương đương với cả u, g và o

Về thay đổi quyền:

r : đọc

w : ghi

x : thực thi

X : quyền truy cập thư mục (thực thi đối với thư mục)

Thay đổi:

+ : bổ sung quyền

- : xóa bỏ quyền

= : thiết lập quyền

Ví dụ:

`chmod 644 note2`

`chmod -R 755 data`

`chmod ug=rwx,o-rwx week`

`chmod -x,+X week`

Lưu ý:

Chỉ có chủ sở hữu hoặc tài khoản "root" có thể thay đổi được quyền đối với tệp tin/thư mục.

Có thể thiết lập sticky bit (s) sẽ chỉ cho phép chủ sở hữu có thể xóa được tệp tin trong khi những người dùng khác vẫn có thể sửa/đọc tệp tin nếu được cấp quyền ghi và đọc.

Thay đổi về chủ sở hữu và nhóm người dùng

Cú pháp: `chown [tùy_chọn] user[:group] <đường_dẫn>`

Tùy chọn hay dùng:

-R thực hiện cho cả các thư mục con và tệp tin ở trong (thực hiện đệ quy)

Ví dụ:

`chown -R root:www-data data`

`chown root data`

`chown userA:groupA data`

Thay đổi nhóm người dùng

Cú pháp: `chgrp [tùy_chọn] group <đường_dẫn>`

Tùy chọn hay dùng:

-R thực hiện cho cả các thư mục con và tệp tin ở trong (thực hiện đệ quy)

Ví dụ:

`chgrp root data`

`chgrp -R www-data data`

Lưu ý: việc thay đổi nhóm người dùng có thể thực hiện bởi chủ sở hữu. Tuy nhiên, nhóm mới phải là nhóm của chủ sở hữu (chủ sở hữu cũng nằm trong nhóm mới).

Siêu người dùng (root) không bị các ràng buộc trên.

Xem thông tin về tệp tin/thư mục

Cú pháp: `stat <đường_dẫn>`

Ví dụ: `stat note2`

Một số lệnh thao tác với người dùng

Thay đổi mật khẩu:

Cú pháp: `passwd [user_name]`

Ví dụ:

`passwd`

passwd centos

Thêm người dùng mới:

Cú pháp: useradd [tùy_chọn] <user_name>

Một số tùy chọn hay dùng:

- d, --home thư mục HOME
- e, --expiredate ngày hết hạn
- g, --gid group
- G, --groups group1[,group2, ...]

Ví dụ:

useradd -G users userA

Tạo nhóm mới

Cú pháp: groupadd [tùy_chọn] group_name

Một số tùy chọn:

- g, -gid GID thiết lập ID cho nhóm

Ví dụ:

groupadd hdhm

Một số lệnh thao tác với người dùng và nhóm người dùng khác

usermod - chỉnh sửa thông tin người dùng

userdel - xóa người dùng

groupmod - chỉnh sửa thông tin nhóm

groupdel - xóa nhóm người dùng

Xem các phiên làm việc hiện tại của hệ thống

Cú pháp:

who

w

Truy cập hệ thống từ xa thông qua giao thức SSH

Cú pháp: ssh [tùy_chọn] user@hostname

Một số tùy chọn

- i path_to_key đường dẫn tới khóa dùng cho kết nối
- p PORT_NUMBER thay đổi số hiệu cổng (mặc định 22)

Ví dụ:

ssh centos@192.168.11.23

ssh -p 3000 ec2-user@amazon.com

ssh -i ~/.ssh/id_vmu_rsa ubuntu@hocstructuyen.vimaru.edu.vn

Sao chép tệp tin thông qua giao thức SSH

Cú pháp: scp [tùy_chọn] src dst

Các tùy chọn tương tự như lệnh ssh

- i path_to_key đường dẫn tới khóa dùng cho kết nối
- p PORT_NUMBER thay đổi số hiệu cổng (mặc định 22)

Với đường dẫn src, dst có dạng

[user@hostname:]path/to/file

Ví dụ:

scp data.txt centos@database:~/data/

Đồng bộ dữ liệu giữa hai thư mục (có thể đồng bộ qua mạng)

Cú pháp: rsync [tùy_chọn] src dst

Các tùy chọn:

- a đồng bộ thư mục
- v hiện thị các thao tác thực hiện
- r thực hiện cả thư mục con và tệp tin (đệ quy)
- z nén dữ liệu trước khi truyền

Đường dẫn src, dst có dạng tương tự như lệnh scp

Ví dụ:

rsync -avr data week

rsync -avrz data centos@database:~/

Xem dung lượng đang sử dụng

Cú pháp: du [tùy_chọn] [đường_dẫn]

Một số tùy chọn hay dùng

- h hiện thị dạng 1M, 2G, ...
- s, --summarize hiện tổng dung lượng của các mục
- max-depth n giới hạn độ sâu
- c hiện tổng dung lượng

Ví dụ:

```
du -sch
du -h --max-depth=1 /usr/
```

Xem dung lượng còn trống của các ổ đĩa.

Cú pháp: `df [tùy_chọn]`

Tùy chọn hay dùng

`-h` hiện thị dạng 1M, 2G

Ví dụ:

`df -h`

Xem danh mục các ổ đĩa (block storage/block device)

Cú pháp: `lsblk`

Xem các module đã được nạp vào trong hệ thống (kernel)

Cú pháp: `lsmod`

Xem các thiết bị kết nối vào giao tiếp USB

Cú pháp: `lsusb`

Xem các tệp tin đang được mở (đang sử dụng)

Cú pháp: `lsof`

Xem các thiết bị kết nối qua pci

Cú pháp: `lspci`

Xem thời gian hoạt động của hệ thống

Cú pháp: `uptime`

Truy cập hệ thống Linux thông qua giao thức SSH từ máy tính sử dụng hệ điều hành Windows:

+ Truy cập qua giao thức SSH (Terminal): `putty`

+ Trao đổi dữ liệu: `winscp`

Để tải tệp tin thông qua giao thức http

Cách 1:

Cú pháp: `wget url_tệp_tin`

Ví dụ: `wget http://vimaru.edu.vn/tai-lieu/he-dieu-hanh-mang.pdf`

Cách 2:

Cú pháp: `curl [tùy_chọn] tệp_tin`

Tùy chọn hay dùng:

`-o filename` lưu dữ liệu tải về vào tệp tin.

`-O` lưu dữ liệu tải về vào tệp tin có tên trùng với tên trong URL.

Ví dụ:

`curl -o he-dieu-hanh-mang.pdf http://vimaru.edu.vn/tai-lieu/he-dieu-hanh-mang.pdf`

`curl -O http://vimaru.edu.vn/tai-lieu/he-dieu-hanh-mang.pdf`

Một số lệnh liên quan đến mạng

+ Xem cấu hình giao diện mạng (cổng mạng):

`ifconfig [tên_cổng]` (áp dụng với hệ thống dùng `iproute`)

`ip address` (áp dụng với các hệ thống dùng `iproute2`)

+ Xem bảng định tuyến

`ip route`

+ Xem địa chỉ lân cận

`ip neigh`

Bộ nhớ lưu trữ (bộ nhớ ngoài)

+ RAID: là một kỹ thuật (giải pháp) để ghép các ổ đĩa vật lý thành một ổ đĩa logic cho phép cải thiện một số đặc điểm: dung lượng lưu trữ, tốc độ, độ tin cậy (mất dữ liệu khi ổ đĩa bị lỗi/hỏng).

+ Các loại/chế độ RAID hiện nay thường được dùng:

+ RAID 0: dữ liệu sẽ được chia ra và ghi lên các ổ đĩa trong nhóm (array)

+ RAID 1: dữ liệu được ghi trên nhiều ổ đĩa đồng thời. Mỗi ổ đĩa đều chứa một bản đầy đủ dữ liệu nên chỉ cần 1 ổ đĩa còn tốt thì dữ liệu sẽ không bị mất mát.

+ RAID 5: dữ liệu được chia ra ghi trên các ổ đĩa và dữ liệu kiểm tra (dữ liệu dư thừa để khôi phục) được ghi trên một ổ đĩa nữa. Điều này cho phép 1 ổ đĩa có thể bị hỏng mà vẫn khôi phục được dữ liệu ban đầu.

+ RAID 6: tương tự RAID 5 nhưng nó cho phép 2 ổ đĩa có thể bị hỏng mà vẫn không bị mất dữ liệu.

+ RAID 10 = RAID 1 + 0

+ RAID 50 = RAID 5 + 0

+ RAID 60 = RAID 6 + 0

+ Có hai phương pháp triển khai RAID:

+ Sử dụng phần mềm hay còn gọi là soft-RAID: cách này thì RAID được triển khai trên phần mềm, các

tính toán xử lý được thực hiện thông qua bộ xử lý trung tâm (CPU). Thông thường việc triển khai RAID mềm sẽ phụ thuộc vào hệ điều hành.

+ Sử dụng các RAID (phần cứng chuyên dụng) để kết nối các ổ đĩa và thực hiện các tính toán xử lý cho các ổ đĩa vật lý.

+ Việc sử dụng phần cứng sẽ đảm bảo không tốn tài nguyên CPU cho việc xử lý.

+ Nó cũng không phụ thuộc vào hệ điều hành (toàn bộ xử lý cũng như cấu hình cho RAID đều thực hiện và được lưu trữ trên các RAID).

+ Thông thường phần cứng có tốc độ xử lý cao hơn sử dụng RAID mềm.

+ Các RAID thường trang bị nguồn năng lượng dự phòng (thường là pin có thể sạc lại được) để duy trì dữ liệu trên bộ nhớ đệm đảm bảo quá trình ghi không bị lỗi nếu xảy ra mất điện đột ngột.

+ Các RAID thường sử dụng giao tiếp SAS/SATA để kết nối tới ổ cứng (HDD, SSD).

+ Đối với các cấu hình RAID có tính dự phòng:

+ Khi có một số ổ đĩa bị hỏng mà hệ thống vẫn chưa bị mất dữ liệu (lỗi) thì nó hoạt động ở trạng thái gọi là "degrade".

+ Các ổ đĩa bị hỏng có thể được thay thế. Sau khi thay thế xong cần thực hiện quá trình "rebuild" để tính toán và ghi lại dữ liệu cho ổ đĩa mới được thay vào.

+ Đối với RAID cứng (các RAID):

+ Hầu hết đều cho phép tháo cắm nóng ổ cứng (hot-swap) - thay thế hoặc bổ sung ổ cứng mới khi hệ thống đang hoạt động.

+ Đa phần các hệ thống cũng hỗ trợ tự động "rebuild" khi thay thế ổ cứng hỏng mà không cần khởi động lại.

+ Quá trình "rebuild" thường sẽ làm giảm đáng kể hiệu năng của hệ thống do cần thực hiện tính toán lại dữ liệu trên ổ cứng mới thay vào.

+ Dung lượng có thể sử dụng để lưu trữ dữ liệu với các thiết lập RAID:

Ký hiệu (quy ước):

RS : dung lượng để lưu trữ sau khi RAID

DS : dung lượng của ổ đĩa dùng cho RAID (dung lượng ổ đĩa nhỏ nhất nếu các ổ đĩa không cùng dung lượng).

n : số lượng ổ đĩa (trong array).

RAID 0:

$RS = n \times DS$

RAID 1:

$RS = DS$

RAID 5:

$RS = (n - 1) \times DS$

RAID 6:

$RS = (n - 2) \times DS$

Ví dụ:

Sử dụng 2 ổ đĩa có dung lượng 500GB cấu hình ở chế độ RAID 1 có thể sử dụng lưu được:
500GB
dữ liệu.

+ Lưu ý:

RAID 0, RAID 1 cần tối thiểu 2 ổ đĩa trong nhóm (array).

RAID 5 cần tối thiểu 3 ổ đĩa trong nhóm.

RAID 6 cần tối thiểu 4 ổ đĩa trong nhóm.

+ Ngoài ra có thể có các ổ đĩa dự phòng ("spare disk") đây là các ổ đĩa được cắm (kết nối) vào hệ thống nhưng chưa được sử dụng. Nó sẽ tự động được lựa chọn bởi các RAID để thay thế cho các ổ bị hỏng.

+ LVM (Logical Volume Management): là một giải pháp sử dụng phần mềm trên Linux cho phép xây dựng các ổ đĩa logic dựa trên các ổ đĩa vật lý. Nó có một số đặc điểm sau:

+ Cho phép trừu tượng hóa (tạo ra ổ đĩa logic) dựa trên ổ đĩa thật. Nên có thể thay đổi dung lượng, thay thế các ổ đĩa vật lý mà không ảnh hưởng tới hệ thống.

+ Có thể triển khai các tính năng tương tự như RAID trên các nhóm ổ đĩa.

+ JBOD (Just a Bunch of Disk) là việc ghép các ổ đĩa lại với nhau một cách đơn giản để tăng dung lượng (không thực hiện việc phân phối quá trình ghi dữ liệu như RAID-0).

+ Một số lệnh liên quan thiết lập LVM:

pvcreate: thêm ổ đĩa vật lý vào PV (Physical Volume)

vgcreate: tạo một VG (Volume Group)

vgextend: thêm PV vào VG

vgdisplay: hiển thị danh sách của VG

lvcreate: tạo LV (Logical Volume)

lvresize: thay đổi kích thước LV

lvextend: mở rộng kích thước LV

lvreduce: giảm kích thước LV

lvconvert: chuyển đổi LV từ chế độ "linear" sang "mirror" hoặc "snapshot".

+ Mount ổ đĩa: là quá trình ánh xạ (gắn) một ổ đĩa (block storage) vào một điểm trong cây thư mục

(mountpoint) để có thể sử dụng (đọc/ghi/sửa/xóa).

Cú pháp: `mount [-t type] device dir`

Trong đó:

-t type là định dạng ổ đĩa (VD: `zfs`, `xfs`, `ext3`, `ext4`)

device là ổ đĩa/phân vùng cần mount

dir là đường dẫn tới nơi cần gán (mountpoint) vào cây thư mục. "dir" phải là đường dẫn tới một thư mục rỗng.

Ví dụ:

`mount /dev/sdb1 /mnt/data`

--> mount ổ đĩa `/dev/sdb1` vào đường dẫn `/mnt/data`

+ Xem danh sách các ổ đĩa mount:

`mount -l`

+ Để thực hiện quá trình dừng mount (unmount) sử dụng lệnh:

`umount mountpoint`

Ví dụ:

`umount /mnt/data`

+ Để quá trình mount được thực hiện tự động khi khởi động hệ điều hành thì các thiết lập mount lưu trong tệp tin:

`/etc/fstab`

Một số lệnh nén và giải nén thư mục (dùng cho sao lưu và phục hồi)

+ Lệnh tar:

Cú pháp: `tar [tùy_chọn] dir/file`

-c tạo bản sao lưu - tạo ra một tệp tin (.tar) từ một thư mục.

-v hiển thị các thao tác được thực thi.

-f file chỉ định dữ liệu ghi vào hoặc đọc ra từ một tệp tin.

-x khôi phục lại dữ liệu (ngược lại với -a)

-z thực hiện nén dữ liệu (mặc định thuật toán gzip sẽ được sử dụng)

Ví dụ:

Tạo file backup.tar từ thư mục vidu

`tar -cvf backup.tar vidu`

Tạo file nén backup.tar.gz từ thư mục vidu

`tar -cvzf backup.tar.gz vidu`

Giải nén tệp tin backup.tar.gz

`tar -xvzf backup.tar.gz`

`tar xvzf backup.tar.gz`

+ Một số lệnh để nén và giải nén

`gzip/gunzip`

`xz`

`bzip2`

Một số lệnh để thao tác với hệ thống:

+ Tắt máy:

`shutdown [tùy_chọn] [thời_gian]`

Một số tùy chọn:

-h, -P, --poweroff đưa về trạng thái poweroff (tắt máy)

-H, --halt đưa về trạng thái halt

-r, --reboot để khởi động lại máy

-k chỉ hiện thông báo cần tắt máy mà không thực hiện thao tác tắt máy

-c hủy thao tác tắt ở trên

Thời gian có thể có dạng:

now - thực hiện ngay

+x - thực hiện sau x phút

hh:mm tắt máy vào thời điểm hh:mm

+ Một số lệnh tắt máy/khởi động lại máy khác:

`reboot`

`poweroff`

`systemctl poweroff`

`systemctl reboot`

+ Thực hiện một lệnh dưới quyền siêu người dùng (root)

`sudo <lệnh>`

Ví dụ:

`sudo rm -rf /var/www`

`sudo vi /etc/fstab`

`sudo su`

- + Chuyển đổi tài khoản
 - Cú pháp: su user_name
 - user_name mặc định là "root"

 - Vidu:
 - su user1
 - su
- + Thoát khỏi phiên đăng nhập hiện tại
 - exit
- + Xem thời gian hoạt động của hệ thống
 - uptime
- + Xem hoặc cài đặt lại ngày giờ hiện tại
 - date
- + Đo thời gian thực thi chương trình/lệnh
 - time lệnh/chương_trình
- + Xem các tiến trình đang hoạt động trong hệ thống
 - ps

 - Ví dụ:
 - ps -aux
- + Xem mức độ sử dụng bộ nhớ
 - vmstat
 - free
- + Xem các tiến trình sử dụng nhiều tài nguyên
 - top
- + Tắt một tiến trình (gửi tín hiệu tắt)
 - kill SIGNAL PID

 - SIGNAL - là loại tín hiệu tắt (TERM, SIGKILL, ...)
 - PID - số hiệu của tiến trình

 - Ví dụ:
 - kill -TERM 1234
- + Một số phím/tổ hợp phím dùng trong chế độ dòng lệnh
 - Ctrl + C gửi tín hiệu SIGINT đến lệnh/chương trình đang chạy.
 - Ctrl + Z chuyển chương trình/lệnh đang chạy sang chế độ chạy nền (gõ lệnh khác trong khi lệnh cũ tiếp tục thực hiện)
 - Sử dụng lệnh "fg" để chuyển các lệnh/chương trình đang chạy nền về chế độ bình thường.
 - TAB - hoàn thành các lệnh đang gõ dở (TAB TAB - đưa danh sách gợi ý để hoàn thành)
- + Xóa màn hình:
 - clear
- + Xem thông tin về bộ xử lý
 - lscpu
- + Xem thông tin hệ thống (kernel)
 - Cú pháp: uname [tùy_chọn]
 - Một số tùy chọn:
 - v, --kernel-version phiên bản (version) của kernel
 - r, --kernel-release nhân hệ điều hành
 - p, --processor kiến trúc bộ xử lý
 - n, --nodename hiện thông tin hostname
 - m, --machine in tên phần cứng
 - i, --hardware-platform
 - o, --operating-system
 - s, --kernel-name tên của kernel
 - a, --all in tất cả các thông tin
- + Một số lệnh về nạp, xem và xóa module (trong kernel space)
 - lsmod - xem danh sách các module

rmmod - xóa một module
modprobe - cài đặt một module

+ Xem phiên đăng nhập trong quá khứ
last [-x]

+ Lập lịch (cronjob)

+ Mục đích: thực hiện một chương trình (thường là các scripts) một cách định kỳ theo lịch đặt trước.

+ Các lệnh liên quan:

crontab -e|-l|-r|-i -u user_name

-e chỉnh sửa
-l hiện danh sách
-r xóa
-i hỏi khi xóa
-u tên người dùng

+ Cấu trúc tệp tin cấu hình cronjob

# minute	hour	day_of_month	month	day_of_week	command
0-59	0-23	1-31	1-12	0-7	program

* day_of_week dạng số 0-7 (0, 7 chỉ ngày Chủ Nhật), viết tắt dạng MON, TUE, WED, ...

* Các trường có thể sử dụng các ký tự:

'-' để biểu diễn khoảng thời gian liên tục

',' để biểu diễn danh sách

'*' biểu diễn cho tất cả

* Sử dụng '#' để thể hiện phía sau nó là chú thích

* Một số thời điểm định nghĩa sẵn:

@reboot - thực hiện mỗi khi khởi động máy

@yearly - thực hiện mỗi năm một lần

@annually - thực hiện một lần mỗi năm

@monthly - thực hiện một tháng một lần

@weekly - thực hiện một tuần một lần

@daily - thực hiện hàng ngày

@hourly - thực hiện mỗi giờ một lần

Ví dụ:

25 10-12 * * WED,THU /reboot.sh

Script /reboot.sh sẽ được thực hiện vào 10:25, 11:25, 12:25 mỗi thứ 4 và thứ 5 hàng tuần.

@monthly /reboot.sh

+ Một số kỹ thuật dùng trong shell

& thêm vào cuối lệnh để thực hiện lệnh ở chế độ chạy nền (background)

&& để nối 2 lệnh với nhau và lệnh phía sau (bên phải) chỉ thực hiện khi lệnh phía trước hoàn thành mà không lỗi

|| để nối 2 lệnh với nhau và lệnh phía sau chỉ thực hiện khi lệnh phía trước trả về lỗi.

| pipe - chuyển dữ liệu stdout của lệnh phía trước vào stdin của lệnh phía sau.

;& để nối 2 lệnh trên cùng một dòng và thực hiện theo thứ tự

+ Một số lưu ý về đường dẫn

'~' --> \$HOME (thư mục home của người dùng)

'.' --> thư mục hiện tại

'..' --> thư mục cha

Ví dụ:

~/vidu

/home/mushu/vidu

./vidu

vidu

