

Lab 1. Linux Bash shell

1. Mục đích

Thông qua bài thực hành này, sinh viên sẽ nắm vững các kiến thức sau:

- Các command cơ bản trong Linux
- Các tính năng cơ bản của bash shell
- Tạo và thực thi file script trên bash shell

2. Tóm tắt lý thuyết

2.1. Các command cơ bản:

Command về file/dir			
ls <paths>	Liệt kê thông tin các file/dir tại <paths>	stat <paths>	Hiển thị thông tin của các file/dir tại <paths>
ls -alF <paths>	Liệt kê chi tiết thông tin các file/dir tại <paths>	file <paths>	Xác định loại của các file tại <paths>
pwd	Hiển thị path của thư mục làm việc hiện tại	less ⁽¹⁾ <paths>	Xem nội dung các file text tại <paths>
cd <path>	Thay đổi thư mục làm việc tới <path>	touch <paths>	Tạo ra các file trống tại <paths>
mkdir <paths>	Tạo ra các dir tại <paths>	du -bs <paths>	Hiển thị kích thước theo byte của file/dir tại <paths>
chmod <mode> <path>	Thay đổi mode của file/dir tại <path> theo <mode> <mode> có định dạng [u,g,o,a] [+,-,=] [r,w,x], trong đó: u: owner, g: group, o: other, a: all +: set, -: unset, =: assign r: read, w: write, x: execute		
Command về program/system			
free -h	Kiểm tra dung lượng RAM	type <name>	Loại của <name>, với <name> là tên command
df -h	Kiểm tra dung lượng ổ cứng	whatis <name>	Mô tả ngắn gọn của <name>
uname -a	Hiển thị thông tin kernel, OS, platform	which <name>	Tìm đường dẫn của <name>, với <name> là tên program
ps	Hiển thị thông tin process trong phiên làm việc	man <name>	Hiển thị tài liệu trợ giúp của <name>

ps aux	Hiển thị thông tin tất cả process		
Các command khác			
wc ⁽¹⁾ -l <paths>	Đếm số line của các file tại <paths>	grep ⁽¹⁾ <name> <paths>	Hiển thị các dòng có chứa <name> trong các file tại <paths>
wc ⁽¹⁾ -w <paths>	Đếm số word của các file tại <paths>	head ⁽¹⁾ -n <N> <path>	Hiển thị <N> dòng đầu tiên của file text tại <path>
echo <args>	Hiển thị các tham số <args> nhận được	tail ⁽¹⁾ -n <N> <path>	Hiển thị <N> dòng cuối cùng của file text tại <path>
cat ⁽¹⁾ <paths>	Hiển thị nội dung các files tại <paths>	sort ⁽¹⁾ -nk <K> <paths>	Sắp xếp các file tại <paths> theo cột <K>, cột <K> phải có dạng số

⁽¹⁾ Nếu <path> không được chỉ định, command sẽ đọc dữ liệu từ stdin

2.2. Các tính năng cơ bản của bash:

Wildcard: Sử dụng một số kí tự đặc biệt để chọn một nhóm các file có path thỏa mãn “pattern matching”. Ví dụ:

Pattern	Matches
*	All files
g*	Any file beginning with g
b*.txt	Any file beginning with b followed by any characters and ending with .txt
Data???	Any file beginning with Data followed by exactly three characters
[abc]*	Any file beginning with either an a, a b, or a c
BACKUP.[0-9][0-9][0-9]	Any file beginning with BACKUP. followed by exactly three numerals
[:upper:]*	Any file beginning with an uppercase letter
[![:digit:]]*	Any file not beginning with a numeral
*[[:lower:]]123]	Any file ending with a lowercase letter or the numerals 1, 2, or 3

I/O redirection và pipeline:

I/O redirection và pipeline là tính năng của shell cho phép chuyển hướng stdin/stdout/stderr của một process tới file hoặc tới một process khác. Ví dụ:

ls /usr/bin/ > temp 2>&1	Liệt kê nội dung của dir /usr/bin và lưu kết quả và lỗi (nếu có) vào file temp
ls /home/fetel/ >> temp	Liệt kê nội dung của dir /home/fetel và thêm kết quả vào cuối file temp
ls /usr/bin less	Liệt kê nội dung của dir /usr/bin và đọc kết quả bằng command less

Command substitution:

Command substitution là tính năng cho phép sử dụng kết quả thực thi (từ stdout) của một command như tham số (argument) của một command khác. Command substitution có dạng `$(...)`. Ví dụ:

<code>echo "hello \$(echo world)"</code>	In ra "hello world"
<code>echo "THIS TEXT WILL BE ADDED \$(cat temp)" > temp</code>	Thêm dòng chữ "THIS TEXT WILL BE ADDED" vào đầu file temp

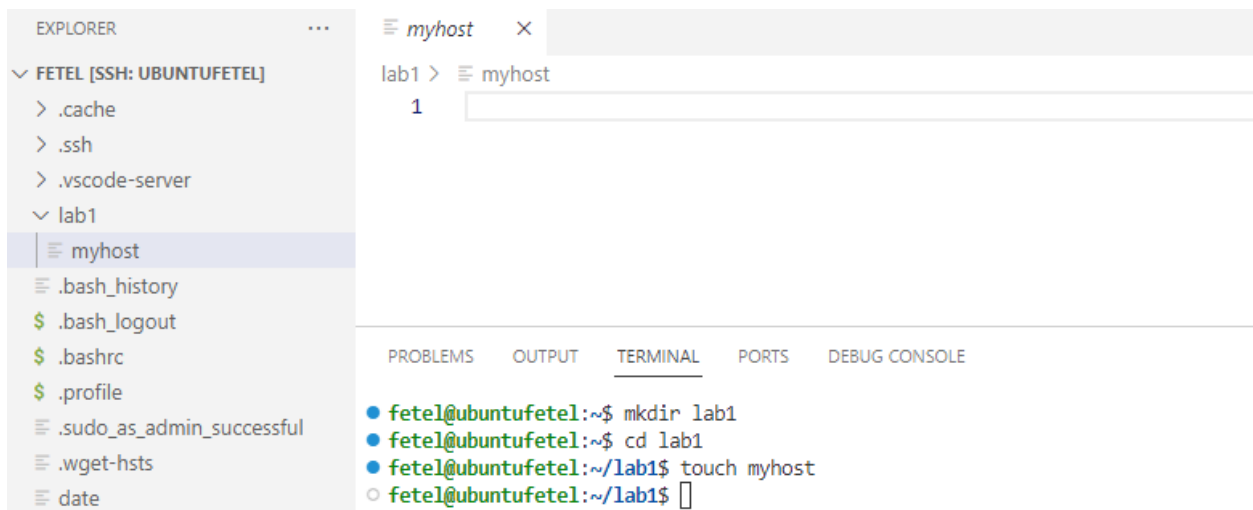
Scripting: Thay vì thực thi từng command trên Terminal, bash cho phép viết các command cần thực thi vào file văn bản (script) và thực thi file này. Bash còn cho phép kết hợp các command này với các cấu trúc lập trình như if/else, for/while, function, variable,... và tạo thành một ngôn ngữ lập trình hoàn chỉnh (shell scripting language). File script có thể nhận các option và argument từ người dùng như một command thông thường trên Terminal.

3. Thực hành

Tại thư mục home, gõ command `mkdir lab1` để tạo thư mục lab1 cho bài thực hành. Chuyển thư mục làm việc sang lab1 bằng command `cd lab1`.

3.1. Script myhost

Tại thư mục lab1, gõ command `touch myhost` để tạo một file trống tên là myhost. Mở file này ra bằng VSCode như hình bên dưới



Nhập vào file nội dung sau và save lại

```
#!/usr/bin/bash

echo
echo "username: $(whoami)"
echo "working dir: $(pwd)"
echo "hostname: $(uname -n)"
echo "kernel name: $(uname -s)"
echo "kernel release: $(uname -r)"
```

```
echo "kernel version: $(uname -v)"
echo "machine name: $(uname -m)"
echo "processor type: $(uname -p)"
echo "hardware platform: $(uname -i)"
echo "operating system: $(uname -o)"
echo "$(lsb_release -a)"

echo
echo "RAM info:"
free -h

echo
echo "Disk info:"
df -h /

echo
echo "Process info:"
echo "Total process running: $(ps aux | wc -l)" # should be 1 less than
```

Gõ command `chmod u+x myhost` để thiết lập quyền thực thi cho file. Để thực thi file, gõ đường dẫn của file vào Terminal và nhấn Enter:

- Sử dụng đường dẫn tuyệt đối: `/home/fetel/lab1/myhost`
- Sử dụng đường dẫn tương đối: `./myhost`

Sau khi thực thi xong, myhost in ra Terminal thông tin cơ bản của máy như phiên bản kernel của hệ điều hành, bộ nhớ RAM, ổ cứng, ...

3.2. Script mydir

Tại thư mục lab1, tạo file script mydir có nội dung như sau

```
#!/usr/bin/bash
echo "Number of file/dir in $1: $(ls $1 | wc -w)"
echo "Total size (byte) of $1: $(du -bs $1)"
echo "Largest file/dir (byte) in $1: $(du -bs $1/* | sort -nk1 | tail -n 1)"
echo "Smallest file/dir (byte) in $1: $(du -bs $1/* | sort -nk1 | head -n 1)"
```

Gõ command `chmod u+x mydir` để thiết lập quyền thực thi cho file. Thực thi file bằng command `./mydir <dir path>` (ví dụ: `./mydir /usr/bin`). Sau khi thực thi, mydir in ra Terminal thông tin thống kê của thư mục như tổng dung lượng ổ cứng sử dụng, tổng số file/dir, file/dir có kích thước lớn nhất, nhỏ nhất trong thư mục đó

BÁO CÁO THỰC HÀNH

Sinh viên:

MSSV: Nhóm:

Bài 1: Thay vì in thông tin ra Terminal, lưu kết quả của myhost và mydir vào cùng một file info.txt và đọc file này bằng command cat và less theo 2 cách:

- a) Sử dụng path của file làm argument của command
- b) Sử dụng IO redirection để đọc file từ stdin của process

Bài 2: Tìm các file trong thư mục /usr/bin có chứa từ khóa “zip” và in ra Terminal theo 2 cách:

- a) Chỉ sử dụng command echo
- b) Chỉ sử dụng command ls và grep

Bài 3: Viết các file script cho phép nhập và tra cứu dữ liệu của sinh viên:

mylistset <MSSV> <họ tên> <ngày sinh>: Thêm thông tin sinh viên

mylistget <MSSV>: Tra cứu thông tin sinh viên theo MSSV. Không in ra nếu không tìm thấy MSSV

mylistls: In ra tất cả danh sách sinh viên đã thêm. Không in ra nếu danh sách trống

Tất cả dữ liệu nhập vào được đảm bảo đúng định dạng như ví dụ bên dưới

Ví dụ:

```
$ ./mylistls
$ ./mylistset 1920000 NguyenVanA 01/01/2000
$ ./mylistset 1920001 NguyenVanB 02/02/2000
$ ./mylistget 1920002

$ ./mylistget 1920000
1920000 NguyenVanA 01/01/2000
$ ./mylistls
1920000 NguyenVanA 01/01/2000
1920001 NguyenVanB 02/02/2000
```

Gợi ý: Lưu trữ dữ liệu của sinh viên trong file list.txt, các file script sẽ đọc/ghi file list.txt này

Bài 4*: Viết 1 file script duy nhất mylist có chức năng giống như bài tập 3 nhưng có thêm tính năng: chỉnh sửa, xóa, sắp xếp thông tin sinh viên