

BÀI TẬP PHẦN HỆ THỐNG TẬP TIN

Thực hành nhanh các bài sau:

B1. Giả sử đang ở gốc. Tạo một thư mục mới bin trong /practice (chưa có practice thì tạo) .

Trong /practice/bin tạo một thư mục tên foo và một file tên newfile

mkdir -p /practice/bin/foo

touch /practice/bin/newfile

Chuyển đến thư mục bin. Xem nội dung của newfile từ vị trí này.

cd /practice/bin

cat ../newfile

Tương đối

Chuyển đến thư mục foo. Xem nội dung của newfile từ vị trí này.

cd foo

Câu lệnh ngắn nhất nào giúp bạn quay trở về practice?

cd /practice

Tuyệt đối

Câu lệnh ngắn nhất nào giúp bạn chuyển đến thư mục home của bạn?

cd \$HOME

Tuyệt đối

Biến PWD là cục bộ hay toàn cục?

Khi bạn thay đổi thư mục (cd), biến PWD được cập nhật để hiển thị thư mục hiện tại.

Trong số các đường dẫn dùng trên, đâu là đường dẫn tương đối, tuyệt đối

B2. Chuyển vào thư mục /etc/init.d, so sánh và giải thích kết quả của lệnh pwd và pwd -P

pwd: Là lệnh để hiển thị đường dẫn của thư mục hiện tại. Nó sẽ in ra đường dẫn tương đối (nếu có) từ thư mục home đến thư mục hiện tại.

pwd -P: Là lệnh để hiển thị đường dẫn hiện tại dưới dạng tuyệt đối (absolute path), không sử dụng đường dẫn tương đối.

B3. Chuyển vào thư mục /etc

So sánh và giải thích kết quả của các lệnh **ls**, **ls -i**, **ls -l**, **ls -a**, **ls -ila**

Lệnh ls là lệnh liệt kê đơn giản nhất. Nó sẽ liệt kê các tập tin và thư mục trong một thư mục, theo thứ tự bảng chữ cái.

Lệnh ls -i sẽ liệt kê các tập tin và thư mục, kèm theo số inode của chúng. Số inode là một số duy nhất được hệ thống tệp Linux sử dụng để xác định một tập tin hoặc thư mục.

Lệnh ls -l sẽ liệt kê các tập tin và thư mục, kèm theo các thông tin chi tiết hơn, bao gồm: Kiểu tập tin, Quyền truy cập, Số lượng liên kết, Kích thước, Ngày tạo, Ngày sửa đổi

Lệnh ls -a sẽ liệt kê tất cả các tập tin và thư mục trong một thư mục, bao gồm cả các tập tin và thư mục ẩn. Tập tin ẩn là các tập tin hoặc thư mục có tên bắt đầu bằng dấu chấm (.)

Lệnh ls -ila là sự kết hợp của lệnh ls -l và ls -a. Nó sẽ liệt kê tất cả các tập tin và thư mục trong một thư mục, kèm theo các thông tin chi tiết hơn, bao gồm cả các tập tin và thư mục ẩn.

Thực hiện hiển thị kết xuất từng lệnh trên theo từng trang màn hình

The image displays four terminal windows from an Ubuntu VirtualBox environment, showing the output of the 'ls' command in the /etc directory. The files are color-coded: white for regular files, blue for directories, green for executables, and cyan for symbolic links.

Terminal 1 (Top): Shows the output of 'ls -la' in /etc. The first line is 'total 1124'. Subsequent lines show file details including permissions, size, date, owner, group, and filename. Filenames are color-coded: directories (blue), regular files (white), executables (green), and symbolic links (cyan).

Terminal 2 (Bottom Left): Shows the output of 'ls -l' in /etc. The first line is 'total 1108'. The output is similar to the first terminal but with a different total size and slightly different file details.

Terminal 3 (Bottom Middle): Shows the output of 'ls -l' in /etc. The first line is 'total 1108'. The output is similar to the previous terminals.

Terminal 4 (Bottom Right): Shows the output of 'ls -la' in /etc. The first line is 'total 1124'. The output is similar to the first terminal.

Các tập tin có màu white, blue, green, cyan, orange có kiểu gì?

White (trắng): Thường không có ý nghĩa đặc biệt, màu mặc định cho các tập tin và thư mục.

Blue (xanh dương): Thường được sử dụng để đặc biệt hoá các thư mục trong lệnh ls. Các tập tin có màu khác.

Green (xanh lá cây): Thường được sử dụng để đặc biệt hoá các tập tin có quyền thực thi (executable files).

Cyan (xanh dương nhạt): Một số hệ thống sử dụng màu cyan để đặc biệt hoá các liên kết (symbolic links) trong lệnh ls.

B4. A.Giải thích kết quả khi thực hiện các lệnh

mkdir a b c d e f

Tạo các thư mục a, b, c, d, e, và f trong thư mục hiện tại.

```
mkdir /a/b/c/d/e/f
```

Cố gắng tạo các thư mục theo đường dẫn tuyệt đối, nhưng sẽ gặp lỗi nếu có một trong các thư mục trước đó không tồn tại.

```
mkdir -p /a/b/c/d/e/f
```

Sử dụng tùy chọn *-p* để tạo các thư mục cấp cao nếu chúng không tồn tại. Điều này sẽ tạo ra cấu trúc thư mục */a/b/c/d/e/f* mà không yêu cầu các thư mục cha phải tồn tại trước đó.

```
mkdir /a /a/b /a/b/c
```

Tạo các thư mục */a*, */a/b*, và */a/b/c*.

B. Cách nào là nhanh nhất để tạo các thư mục theo thứ bậc: */dir1/dir2/dir3/dir4*?

```
mkdir -p /dir1/dir2/dir3/dir4
```

Tạo thêm trong *dir1* các file: *f1,f2*; trong *dir2* các file: *f3,f4*

```
touch /dir1/f1 /dir1/f2
```

```
touch /dir1/dir2/f3 /dir1/dir2/f4
```

Tạo một thư mục */save*, sau đó copy nhánh thư mục *dir1* sang */save*

```
mkdir /save
```

```
cp -r /dir1 /save
```

Xoá nhánh thư mục */save/dir1* bằng *rmdir*, xoá nhánh thư mục */dir1* với *rm*. Phân biệt sự khác nhau. Cách xoá nhanh nhất?

```
rmdir /save/dir1
```

```
rm -r /dir1
```

Nếu */save/dir1* trống rỗng, sử dụng *rmdir* để xoá nhanh chóng. Nếu có nội dung bên trong thư mục và muốn xoá toàn bộ cả cây thư mục, sử dụng *rm -r* mà không cần kiểm tra.

B5. So sánh và giải thích kết quả khi thực hiện:

```
echo "line1"
```

In chuỗi *line1* lên terminal

```
echo "line1">abc.txt ; echo "line2">abc.txt ; echo  
"line3">abc.txt
```

Lệnh này sẽ ghi đè nội dung của tệp tin *abc.txt* bằng chuỗi *"line1"*. Nếu tệp tin không tồn tại, nó sẽ được tạo mới.

```
echo "line1">abc.txt ; echo "line2">>abc.txt ; echo  
"line3">>abc.txt
```

Câu lệnh này sử dụng dấu chấm phẩy để thực hiện nhiều lệnh liên tiếp. Đoạn *echo "line1">abc.txt* sẽ ghi đè nội dung của tệp tin *abc.txt* bằng chuỗi *"line1"*. Đoạn *echo "line2">>abc.txt* sẽ thêm chuỗi *"line2"* vào cuối nội dung của tệp tin *abc.txt* mà không ghi

đề lên nó. Đoạn `echo "line3">>abc.txt` tương tự, thêm chuỗi "line3" vào cuối nội dung của tệp tin `abc.txt` mà không ghi đề lên nó.

B6. So sánh và giải thích kết quả khi thực hiện:

```
cat taptinkhongco > abc1.txt
```

Lệnh này sẽ đọc nội dung của tệp tin `taptinkhongco` và ghi đề nó vào tệp tin `abc1.txt`. Nếu tệp tin `taptinkhongco` không tồn tại, lệnh sẽ tạo một tệp tin mới có tên `abc1.txt`.

```
cat taptinkhongco 2> abc1.txt
```

Lệnh này sử dụng `2>` để chuyển hướng lỗi tiêu chuẩn (`stderr`) sang tệp tin `abc1.txt`. Điều này có nghĩa là nếu có bất kỳ lỗi nào xuất hiện trong quá trình thực hiện lệnh `cat taptinkhongco`, thì thông điệp lỗi đó sẽ được ghi vào tệp tin `abc1.txt`.