

# Hướng dẫn Chi tiết về Đường dẫn trong Linux

## 1. Đường dẫn tuyệt đối (Absolute Path)

### 1.1 Định nghĩa và Đặc điểm

- Bắt đầu bằng dấu gạch chéo (/)
- Chỉ định đường dẫn đầy đủ từ thư mục gốc
- Độc lập với vị trí hiện tại của người dùng
- Luôn trỏ đến cùng một vị trí

### 1.2 Cú pháp và Ví dụ

```
# Cú pháp cơ bản
/thư_mục_1/thư_mục_2/tập_tin

# Ví dụ thực tế
/home/test           # Thư mục home của user test
/etc/passwd          # File cấu hình người dùng
/var/log/syslog       # File log hệ thống
```

### 1.3 Ưu điểm và Nhược điểm

#### Ưu điểm:

- Chính xác và rõ ràng
- Không phụ thuộc vào vị trí hiện tại
- An toàn khi sử dụng trong scripts

#### Nhược điểm:

- Dài và khó nhớ
- Phải gõ nhiều ký tự hơn
- Cần biết cấu trúc thư mục đầy đủ

## 2. Đường dẫn tương đối (Relative Path)

### 2.1 Định nghĩa và Đặc điểm

- Không bắt đầu bằng dấu /
- Phụ thuộc vào thư mục hiện tại
- Sử dụng các ký hiệu đặc biệt (., ..)
- Ngắn gọn và tiện lợi

### 2.2 Cú pháp và Ví dụ

```
# Trong thư mục hiện tại
touch filename          # Tạo file trong thư mục hiện tại

# Di chuyển lên thư mục cha
cd ..                   # Di chuyển lên một cấp

# Di chuyển vào thư mục con
cd folder_name          # Di chuyển vào thư mục con
```

## 2.3 Ký hiệu đặc biệt

- `.` : Thư mục hiện tại
- `..` : Thư mục cha
- `~` : Thư mục home của user

## 3. Command Path (Đường dẫn Lệnh)

### 3.1 Khái niệm PATH

- Biến môi trường chứa danh sách các thư mục
- Hệ thống tìm kiếm lệnh trong các thư mục này
- Định nghĩa trong `$PATH`

### 3.2 Lệnh which

```
# Xem đường dẫn của lệnh
which cd          # /usr/bin/cd
which cat         # /usr/bin/cat
which touch       # /usr/bin/touch
```

### 3.3 Giải thích chi tiết

#### 1. Tại sao cần which:

- Xác định vị trí thực của lệnh
- Kiểm tra xem lệnh có tồn tại
- Phân biệt giữa lệnh built-in và external

#### 2. Thư mục `/bin`:

- Chứa các lệnh cơ bản
- Cần thiết cho khởi động hệ thống
- Accessible cho mọi user

## 4. Best Practices và Use Cases

### 4.1 Khi nào dùng Absolute Path

1. Trong shell scripts
2. Cấu hình hệ thống
3. Cron jobs
4. Các tác vụ tự động

## 4.2 Khi nào dùng Relative Path

1. Làm việc trên command line
2. Di chuyển giữa các thư mục
3. Thao tác với files gần đó
4. Phát triển và testing

## 4.3 Làm việc với Command Path

```
# Xem PATH hiện tại
echo $PATH

# Thêm thư mục vào PATH
export PATH=$PATH:/new/path

# Kiểm tra vị trí lệnh
type command_name
```

# 5. Ví dụ Thực tế

## 5.1 Quản lý Project

```
# Sử dụng absolute path
/home/user/projects/website/index.html

# Sử dụng relative path
cd ~/projects
cd website
touch index.html
```

## 5.2 System Administration

```
# Kiểm tra binary location
which nginx
which python
which java

# Xem nội dung /bin
cd /bin
ls -l
```

## 6. Troubleshooting

### 6.1 Common Issues

#### 1. Command not found

```
# Kiểm tra PATH
echo $PATH

# Kiểm tra tồn tại
which command_name
```

#### 2. Permission denied

```
# Kiểm tra quyền
ls -l /path/to/command

# Sửa quyền nếu cần
chmod +x /path/to/command
```

### 6.2 Validation và Testing

```
# Kiểm tra path tồn tại
test -e /path/to/check && echo "Exists" || echo "Not found"

# Kiểm tra loại path
test -d /path && echo "Directory" || echo "Not directory"
test -f /path && echo "File" || echo "Not file"
```

## 7. Security Considerations

### 7.1 Path Security

1. Sử dụng absolute path trong scripts
2. Kiểm tra quyền truy cập
3. Validate input paths
4. Tránh symbolic link attacks

### 7.2 Best Practices

1. Sử dụng quotes cho paths
2. Kiểm tra tồn tại trước khi sử dụng
3. Handle spaces và special characters
4. Regular audit của PATH