

## Bài 4: Quản lý người dùng

Nguyễn Minh Hải

Ngày 3 tháng 10 năm 2017

# Nội dung

## 1 Người dùng và nhóm người dùng

- Giới thiệu

## 2 Quyền của tệp tin

# Nội dung

## 1 Người dùng và nhóm người dùng

- Giới thiệu

## 2 Quyền của tệp tin

# Nội dung

## 1 Người dùng và nhóm người dùng

- Giới thiệu

## 2 Quyền của tệp tin

# Người dùng và nhóm người dùng

- Là cách tổ chức truy cập và phân quyền sử dụng tài nguyên của hệ điều hành Linux khi có nhiều người dùng
- Mục đích chính của tổ chức người dùng là bảo mật thông qua việc đặt giới hạn cho quyền truy cập các tài nguyên của hệ thống
- Mỗi người dùng có một tên định danh phân biệt và một số id phân biệt và có thể kèm theo mật khẩu để bảo mật
- Người dùng cao nhất là *root* người dùng này có toàn bộ quyền đối với mọi tài nguyên của hệ điều hành
- Những người dùng được phân ra thành các nhóm; mỗi nhóm có các quyền mà tất cả người dùng trong nhóm đó đều được thừa hưởng

## Các lệnh in thông tin người dùng

- Xem thông tin người dùng hiện tại dùng **whoami**
- In thông tin của một người dùng và nhóm người dùng của một người dùng nào đó

```
$id -u username
```

```
$id -g username
```

- Thông tin của tất cả các người dùng và nhóm người dùng được lưu trong thư mục */etc/passwd*

```
$cut -d: -f1 /etc/passwd
```

- Mật khẩu của người dùng thường được mã hóa và lưu vào tệp tin */etc/shadow*
- Ví dụ về thông tin của người dùng trong tệp */etc/passwd*  
*robert:x:502:502::/home/robert:/bin/bash*

Định danh: *robert*; Mật khẩu: *x* mã hóa lưu trong tệp *shadow*, UID: 502, GID giống UID: 502; Thư mục người dùng: */home/robert*; Shell mặc định khi login: */bin/bash*

# Các lệnh quản lý người dùng

- Thêm người dùng, sử dụng lệnh **useradd**

```
$useradd [-m] [-G groupname] [-s /dir/of/shell] username
```

Để thêm mật khẩu dùng luật lựa chọn *-p* hoặc dùng lệnh **passwd** (lệnh này cũng dùng để đổi mật khẩu)

```
$passwd username
```

- Thay đổi thông tin người dùng

- Đổi tên

```
$usermod -l newname oldname
```

- Đổi thư mục của người dùng

```
$usermod -d /new/home -m username
```

- Thêm người dùng vào nhóm người dùng khác

```
$usermod -aG additional-groups
```

# Đăng nhập người dùng

- Khi đăng nhập vào hệ thống Linux sẽ phải khai báo người dùng và mật khẩu
- Lệnh **su** chuyển sang người dùng khác, sau đó nhập mật khẩu `$su username`
- Nếu muốn chuyển người dùng *root* thì bỏ *username* hoặc thay bằng - `$su`



# Nội dung

## 1 Người dùng và nhóm người dùng

- Giới thiệu

## 2 Quyền của tệp tin

## Quyền cấp của tệp tin

- Mỗi tệp tin có quy định về 3 nhóm quyền là: chủ sở hữu *owner*; nhóm *group* và người dùng khác *others*
- Mỗi nhóm quyền quy định về 3 quyền: đọc *read*, ghi *write* và thực thi *execute*
- Ví dụ tệp tin có quyền *rwrx\_x\_\_x*
- Biểu diễn nhị phân của quyền trên là *111.101.001* và biểu diễn trong hệ cơ số 10 là *751*
- Cấp quyền cho tệp tin bằng lệnh **chmod**  
chmod 754 filename  
chmod u=rwx,g-rx,o+r filename  
chmod g+x,o-r filename  
chmod +rwx filename

## các quyền khác

- **setuid** cho phép một tệp khi chạy (bởi một người khác) có quyền của người sở hữu tệp đó; sử dụng kí hiệu, số **s**, **4**  
`chmod u+s filename`  
`chmod 4750 filename`
- **setgid** cho phép một tệp khi chạy (bởi một người khác) có quyền của nhóm người dùng sở hữu tệp đó; sử dụng kí hiệu, số **s**, **2**  
`chmod g+x filename`
- **sticky** khi cấp quyền này cho một thư mục, người dùng khác không được phép xóa nội dung của thư mục đó ngoại trừ chủ sở hữu nội dung đó và root; sử dụng kí hiệu, số **t**, **1**  
`chmod o+t /opt/dump`

# umask

- Thông thường khi các tệp mới được tạo thành sẽ được cấp quyền mặc định
- Sử dụng mặt nạ (*mask*) để giới hạn các quyền này
- Xem quyền mặc định hiện tại dùng lệnh **umask**
- Chạy một số ví dụ