

HỆ ĐIỀU HÀNH LINUX (Quyền truy xuất)

Phạm Nguyên Khang, Đỗ Thanh Nghị
Email: pnkhang,dtngghi@cit.ctu.edu.vn

Quyền truy xuất

2

Tất cả các phần tử của cây thư mục đều được gán 3 quyền truy xuất:

Đọc (read)	r
Ghi (write)	w
Thực thi (execute)	x
cho 3 nhóm người dùng	
Chủ sở hữu (user)	u
Nhóm sở hữu (group)	g
Người khác (others)	o
Tất cả (all)	a

Ví dụ, một file được gán quyền: `rwX r-x r--`

Chủ sở hữu được quyền đọc, ghi, thực thi

Nhóm sở hữu được quyền đọc và thực thi

Những người khác chỉ được quyền đọc

Quyền truy xuất

3

Thay đổi chủ sở hữu

Thay đổi chủ sở hữu

`chown <user> <file>`

Ví dụ: **chown pnkhang data**

`chown <user>:<group> <file>` (có thể sử dụng dấu `.` thay cho `:`)

Ví dụ: **chown pnkhang:develop data**

Thay đổi nhóm sở hữu

`chgrp group file`

Ví dụ: **chgrp admin data**

Ghi chú:

Chỉ có root mới có quyền thay đổi chủ sở hữu

Chủ sở hữu có quyền thay đổi nhóm sở hữu nếu anh ta thuộc về nhóm sở hữu mới

Quyền truy xuất

4

Thay đổi quyền truy cập

Sử dụng lệnh:

`chmod <u|g|o|a><+|-|=><r|w|x> <tên file>`

Ví dụ:

`chmod u+x data` (hay **`chmod u=x data`**)

`chmod o-r data`

`chmod g=u data`

Sử dụng hệ bát phân

Quyền = read + write + execute

Với read = 4, write = 2, execute = 1

`chmod 644 data` **→ `rw-r-- r--`**

`chmod 755 data` **→ `rw-x r-x r-x`**

Ghi chú: chỉ có chủ sở hữu hoặc root mới có quyền thay đổi quyền truy cập

Quyền truy xuất

5

Quyền truy cập mặc định (khi tạo file hoặc thư mục)

Phụ thuộc vào giá trị của **umask** (có thể xem giá trị này bằng lệnh **umask**)

Quyền truy cập file = $666 \text{ AND } (\text{NOT } \mathbf{umask})$

Quyền truy cập thư mục = $777 \text{ AND } (\text{NOT } \mathbf{umask})$

Ví dụ: nếu $\mathbf{umask} = 022$, quyền truy cập mặc định khi tạo file sẽ là:

$666 \text{ AND } (\text{NOT } 022) = 666 \text{ AND } 755 = 644$

và quyền truy cập mặc định khi tạo thư mục sẽ là:

$777 \text{ AND } (\text{NOT } 022) = 777 \text{ AND } 755 = 755$

Ghi chú: các phép toán AND, NOT được thực hiện trên các số nhị phân, ví dụ:

$\text{NOT } (022) = \text{NOT } (000 \ 010 \ 010) = 111 \ 101 \ 101 = 755$

$666 \text{ AND } 755 = (110 \ 110 \ 110) \text{ AND } (111 \ 101 \ 101) = 110 \ 100 \ 100 = 644$

Quyền truy xuất

6

Trường hợp đặc biệt:

Tất cả các thao tác trên file được điều khiển bằng 3 quyền cơ bản: r, w, x

Tạo một file = thêm một mục (inode, tên file) vào một thư mục → Do đó, để tạo được file mới ta phải có **quyền ghi (w) trên thư mục chứa file này**

Tương tự cho việc **đổi tên** (rename) và **xóa file** (delete)

Như vậy, muốn xóa một file ta chỉ cần có quyền ghi (w) trên **thư mục chứa file cần xóa** chứ không cần quyền ghi trên file cần xóa

Chú ý:

Đối với một thư mục, quyền thực thi (x) sẽ cho phép đi vào bên trong thư mục này (lệnh cd chẳng hạn)

Đối với tập tin bình thường, quyền x không có ý nghĩa gì cả

Quyền truy xuất



Đối với các **file thực thi**, ngoài 3 quyền cho 3 nhóm người dùng (u, g, o), còn có thêm 3 quyền khác nữa:

Quyền SUID (SUID bit): u+s hay 4

Quyền SGID (SGID bit): g+s hay 2

Cờ giới hạn xóa hay Sticky bit: t hay 1

Chú ý:

Có thể kết hợp SUID, GUID và Sticky thành một con số bát phân, ví dụ đặt SUID (4), và sticky bit (1) $\rightarrow 4 + 1 = 5$

Không thể xóa các bit này dùng các số bát phân mà chỉ có thể xóa bằng cách sử dụng các ký tự, ví dụ: **chmod u-s data**

Nếu bit SUID, hoặc SGID được đặt, khi xem quyền truy cập của một file hay thư mục (lệnh `ls -l`) ta sẽ thấy **x** chuyển thành **s**

Quyền truy xuất

8

Quyền SUID (SUID bit):

File: cho phép người dùng khác thực thi file này với tất cả các quyền của chủ sở hữu

Thư mục: tự động đổi chủ sở hữu của file được tạo ra hoặc được di chuyển đến thư mục này thành chủ sở của thư mục

Quyền SGID (SGID bit):

Tương tự như SUID nhưng áp dụng với quyền của nhóm sở hữu

Ví dụ:

```
chmod g+s /bin/wc  
chmod 4755 /bin/cat
```

Sticky bit:

File: lần thực thi sau sẽ nạp chương trình nhanh hơn

Thư mục:

Nếu thêm quyền t cho thư mục, thì chỉ có chủ sở hữu mới có thể xóa/đổi tên bỏ file hoặc thư mục bên trong thư mục này

Ví dụ:

```
$ mkdir public  
$ chmod 777 public  
$ chmod +t public
```