

03 权限控制

by ZWN

用户和组的概念

Linux中每个用户都至少属于一个用户组，以保证对一个用户组中的所有用户进行集中管理

实际应用

文件权限对命令执行的限制

- 在移动文件时不需要被移动文件的权限，但需其所在目录具有写权限。
- 在目录下增删文件、子目录时要有目录的写权限（touch,rm,mkdir,rmdir）
- 用ls列目录要有目录的读权限
- 进入目录或将该目录作路径分量时(cd)要有目录的执行权限
- 要使用任一个文件，必须有该文件及找到该文件的路径上所有目录分量的执行权限
- 例题参见PPT

chmod

用来改变文件或目录的权限

使用方法

数字形式：chmod [-options] xyz [filelist]
xyz：权限的十进制表示

- 符号形式：

chmod	[-options]	u	+	r	[filelist]	+: 加入权限 -: 去掉权限 =: 设定权限
		g	-	w		
		o	=	x		
		a				

选项

- c：只有在文件权限确实改变时才进行详细说明
- R：递归改变目录及其内容的权限
- f：不打印“权限不能改变”之类的文件错误信息
- v：详细说明权限的变化

chown

用于改变文件的所有者。还可以用chown命令在改变文件所有者的同时也改变其所属用户组

该命令仅用于系统管理员对文件和管理

格式：chown [-options] user[: group] filelist
user表示将要改变的所有者用户名或用户id
group表示将要改变的所属用户组名或组id

选项

- R：递归改变目录及其内容的所属用户或用户组

用户类型及表示方法

用户类型

文件所有者：能够设定同组用户和其他用户对该文件的访问权限。一般来说是文件的创建者

同组用户：具有相同性质的所有用户,被系统管理员分在同一组。文件所有者或系统管理员可将文件的权限赋予组内的其它用户

其它用户：文件所有者或系统管理员还可以将文件的访问权赋予系统中所有其它的用户。这样,系统中每一位用户可能都能访问你的某一文件或目录

用字符表示用户类型

- u 文件的所有者
- g 同组用户
- o 其它用户
- a all=ugo,即所有用户

文件和目录的权限表示

符号化表示

共9位，每3位为一组，共3组
各组依次代表文件所有者、同组用户和其他用户对该文件的权限。

每组的3位依次代表读、写、执行权限。

用三个十进制数字（文件权限值）表示文件权限

文件权限值为该类型用户对文件的读、写、执行权限值的累加：

十进制表示

读(r): 4 写(w): 2 执行(x): 1 禁止(-): 0

rwx	7	4+2+1	读、写、执行
rw-	6	4+2+0	读、写
r-x	5	4+1	读、执行
r--	4	4+0+0	只读
-wx	3	2+1	写、执行
-w-	2	0+2+0	只写
---	0	0+0+0	禁止读、写、执行

--x 1 0+0+1 只执行