

## Linux权限理解

- 掌握Linux中的两种角色，先能够进行两种角色的自由切换
- 掌握Linux中常见的文件权限概念，使用相关命令进行Linux进行权限操作
- 掌握Linux中目录权限的相关概念
- 了解一般用户如何进行sudo配置

## Linux用户的概念

Linux下有两种用户：超级用户（root）、普通用户。

- 超级用户：可以再linux系统下做任何事情，不受限制
- 普通用户：在linux下做有限的事情。
- 超级用户的命令提示符是“#”，普通用户的命令提示符是“\$”。

### 角色切换

普通用户切换到超级用户

```
[whb@VM_0_12_centos ~]$ whoami
whb
[whb@VM_0_12_centos ~]$ sudo -s
[sudo] password for whb: //输入自己的密码
[root@VM_0_12_centos whb]# whoami
root
```

超级用户切换到普通用户

```
[root@VM_0_12_centos whb]# whoami
root
[root@VM_0_12_centos whb]# exit //这里只要退出自己曾经的切换，就可以回到当前账户
exit
[whb@VM_0_12_centos ~]$ whoami
whb
```

另外，用户角色切换还有其他方式，还有很多细节，但是我们是刚刚接触，现在已尽快使用起来为唯一目标。

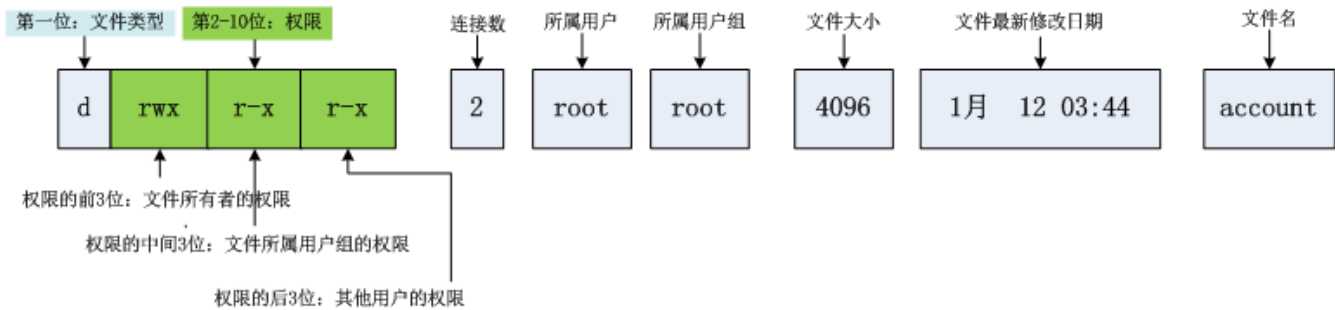
还有，Linux是多用户的，所以可以同时登陆多人，甚至你可以切成别人的身份，但是目前我们的Linux机器，大都是自己使用，所以先不考虑这些。

## Linux权限管理

文件访问者的分类（人）

- 文件和文件目录的所有者：u---User（中国平民 法律问题）
- 文件和文件目录的所有者所在的组的用户：g---Group（不多说）
- 其它用户：o---Others（外国人）

文件类型和访问权限（事物属性）



## 文件类型

- d: 文件夹
- : 普通文件
- l: 软链接（类似windows的快捷方式）
- b: 块设备文件（例如硬盘、光驱等）
- p: 管道文件
- c: 字符设备文件（例如屏幕等串口设备）
- s: 套接口文件

### file指令:

功能说明: 辨识文件类型。

语法: file [选项] 文件或目录...

```
[whb@VM_0_12_centos http]$ file wwwroot/
wwwroot/: directory
[whb@VM_0_12_centos http]$ file testCgi
testCgi: ELF 64-bit LSB executable, x86-64, version 1 (SYSV), dynamically linked (uses
shared libs), for GNU/Linux 2.6.32, BuildID[sha1]=64d3a89bc5fea57f0e6ce18d4d9168fbc235b012,
not stripped
[whb@VM_0_12_centos http]$ file Log.hpp
Log.hpp: C source, UTF-8 Unicode text
```

### 基本权限

- 读 (r/4) : Read对文件而言, 具有读取文件内容的权限; 对目录来说, 具有浏览该目录信息的权限
- 写 (w/2) : Write对文件而言, 具有修改文件内容的权限; 对目录来说具有删除移动目录内文件的权限
- 执行 (x/1) : execute对文件而言, 具有执行文件的权限; 对目录来说, 具有进入目录的权限
- “-”表示不具有该项权限

### 文件权限值的表示方法

字符表示方法

Linux表示	说明	Linux表示	说明
r--	只读	-w-	仅可写
--x	仅可执行	rw-	可读可写
-wx	可写和可执行	r-x	可读可执行
rwX	可读可写可执行	---	无权限

#### 8进制数值表示方法

权限符号 (读写执行)	八进制	二进制
r	4	100
w	2	010
x	1	001
rw	6	110
rx	5	101
wx	3	011
rwX	7	111
---	0	000

#### 文件访问权限的相关设置方法

a)chmod 功能：设置文件的访问权限

格式：chmod [参数] 权限 文件名

常用选项：

- R -> 递归修改目录文件的权限
- 说明：只有文件的拥有者和root才可以改变文件的权限

chmod命令权限值的格式

##### ① 用户表示符+/-=权限字符

- +:向权限范围增加权限代号所表示的权限
- -:向权限范围取消权限代号所表示的权限
- =:向权限范围赋予权限代号所表示的权限
- 用户符号：
  - u: 拥有者
  - g: 拥有者同组用
  - o: 其它用户
  - a: 所有用户

实例：

```
# chmod u+w /home/abc.txt
# chmod o-x /home/abc.txt
# chmod a=x /home/abc.txt•
```

## ②三位8进制数字

实例：

```
# chmod 664 /home/abc.txt
# chmod 640 /home/abc.txt
```

### b)chown

功能：修改文件的拥有者

格式：chown [参数] 用户名 文件名

实例：

```
# chown user1 f1
# chown -R user1 filegroup1
```

### c)chgrp

功能：修改文件或目录的所属组

格式：chgrp [参数] 用户组名 文件名

常用选项：-R 递归修改文件或目录的所属组

实例：

```
# chgrp users /abc/f2
```

### 使用 sudo分配权限

(1) 修改/etc/sudoers 文件分配文件 - 可以不讲，但是同学这块可能会有问题，所以可以提一下

```
# chmod 740 /etc/sudoers
# vi /etc/sudoer
```

(2) 使用 sudo 调用授权的命令

- \$ sudo -u 用户名 命令

实例：

```
[whb@VM_0_12_centos http]$ whoami
whb
[whb@VM_0_12_centos ~]$ sudo -u root whoami
[sudo] password for whb:
root
```

or

```
[whb@VM_0_12_centos http]$ whoami
whb
[whb@VM_0_12_centos http]$ sudo whoami
[sudo] password for whb:
root
```

## 目录的权限

- 可执行权限: 如果目录没有可执行权限, 则无法cd到目录中.
- 可读权限: 如果目录没有可读权限, 则无法用ls等命令查看目录中的文件内容.
- 可写权限: 如果目录没有可写权限, 则无法在目录中创建文件, 也无法在目录中删除文件.

于是, 问题来了~~

换句话说来讲, 就是只要用户具有目录的写权限, 用户就可以删除目录中的文件, 而不论这个用户是否有这个文件的写权限.

这好像不太科学啊, 我张三创建的一个文件, 凭什么被你李四可以删掉? 我们用下面的过程印证一下.

```
[whb@VM_0_12_centos ~]$ mkdir fortest
[whb@VM_0_12_centos ~]$ sudo chown root:root fortest
[whb@VM_0_12_centos ~]$ ll
drwxrwxrwt  2 root root 4096 Sep  9 18:18 fortest
[whb@VM_0_12_centos ~]$ chmod 0777 fortest
[whb@VM_0_12_centos ~]$ ls -ld fortest/
drwxrwxrwt 2 root root 4096 Sep  9 18:18 fortest/
[whb@VM_0_12_centos ~]$ cd fortest/
[whb@VM_0_12_centos fortest]$ touch test.c
[whb@VM_0_12_centos fortest]$ ll
total 0
-rw-rw-r-- 1 whb whb 0 Sep  9 18:10 test.c
[whb@VM_0_12_centos fortest]$ sudo touch test_root.c
[whb@VM_0_12_centos fortest]$ ll
total 0
-rw-rw-r-- 1 whb whb 0 Sep  9 18:10 test.c          //普通用户的
-rw-r--r-- 1 root root 0 Sep  9 18:10 test_root.c  //超级用户的
[whb@VM_0_12_centos fortest]$ whoami
whb
[whb@VM_0_12_centos fortest]$ rm test_root.c          //普通用户删除root的文件
rm: remove write-protected regular empty file 'test_root.c'? y
[whb@VM_0_12_centos fortest]$ ls                      //删除成功
test.c
```

为了解决这个不科学的问题, Linux引入了粘滞位的概念.

但是, 高潮来了, 我们不讲.

这个问题对同学们来讲, 目前理解过难了, 我们暂时不讲, 但是这个问题大家得知道.

后续的正式课程, 这块是专门讲解的, 有兴趣的可以先了解一下, 或者私信我哦~

## 关于目录权限的总结

- 目录的可执行权限是表示你可否在目录下执行命令.

- 如果目录没有-x权限，则无法对目录执行任何命令，甚至无法cd 进入目, 即使目录仍然有-r 读权限（这个地方很容易犯错，认为有读权限就可以进入目录读取目录下的文件）
- 而如果目录具有-x权限，但没有-r权限，则用户可以执行命令，可以cd进入目录。但由于没有目录的读权限
- 所以在目录下，即使可以执行ls命令，但仍然没有权限读出目录下的文档。

比特就业课