# 一、Linux权限管理

## 1.权限概述

在Linux系统对于权限的设定非常的敏感，如果某个用户执行一个操作时，提示权限不足，那么根据Linux系统的权限设定的思想(没有权限绝对不会睁一只眼闭一只眼)，就能够判断出该用户不具备此文件的执行权限。

**在Linux系统中，有以下的权限表示。业内人士称之为：**

逻辑权限

物理权限

普通用户的root的权限。

## 2.逻辑权限(chmod)

在Linux系统中不管是文件还是目录。(在Linux系统中，将所有的东西都视为文件。)都有固定权限表示。

例：

drwxr-xr-x. 2 root root 4096 5月 13 15:27 home

-rw-r--r--. 1 root root 45537 5月 13 11:15 install.log

两个文件分别是：第一个是目录，第二个是普通文件，根据信息的第一个字母(文件类型)来查看，d表示该文件是一个目录文件，-表示该文件是一个普通文件。

后面每三个权限成为一组，每组中分别有三个权限：

|  |  |  |
| --- | --- | --- |
| 字符 | 权限 | 数字 |
| r | 读 | 4 |
| w | 写 | 2 |
| x | 执行 | 1 |

说明：除了第一个字母不参与权限的表示，其他的都为权限标识符。

每三个为一组，共有三组：

|  |  |  |
| --- | --- | --- |
| 第一组 | 用户 | user |
| 第二组 | 用户组 | group |
| 第三组 | 其他人 | other |

目录 用户权限 组权限 其他人权限

d rwx r-x r-x

以上面的意思为是一个文件夹目录，用户的权限为读写执行，组权限为读跟执行，其他人的权限为读跟执行。

说明：Linux系统中，不管是什么版本的系统，权限的标识符号的位置是不会发生任何的改变，也就是说：

第1个永远是文件类型，

第2-4(第一组)永远是读、写、执行，用户的权限，

第5-7(第二组)永远是读、写、执行，用户组的权限，

第8-10(第三组)永远是读、写、执行，其他人的权限。

如果某个文件权限标识为----------，那么则说明此文件不允许任何的读取、写入、执行。

### 2.1.修改文件/夹的权限(chmod)

chmod命令可以用来修改某个文件或文件夹的权限。

选项：

|  |  |
| --- | --- |
| -R | 递归处理 |

修改文件/夹的权限时，可以使用字符权限，也可以使用数字权限。

#### 2.1.2修改文件权限为任何人不允许读写指定

# bash

touch test\_1 # 当前文件的权限是-rw-r--r--

将此文件的权限修改为----------

# bash

chmod 000 test\_1

#### 2.1.3.为此文件，每组都增加一个读的权限

# bash

chmod 444 test\_1

or

chmod +r test\_1

#### 2.1.4.使用别名添加权限

为test\_1此文件的用户增加一个rw-，组增加一个r-x，其他人---，0表示没有权限。

# a=all,u=user,g=group,o=other

chmod u+rw,g+rx test\_1

or

chmod 650 test\_1

chmod在修改文件权限的时候，哪个便捷用哪个方法。

比如：

如果要是给三组增加执行权限的时候，+x就数字计算要快。字符权限就比数字要便捷(不需要计算)，如果是为每组增加不同权限的时候，用数字比较便捷(书写便捷)。

### 2.2.修改文件/文件夹的所有者（属主）:属组

修改所有者：

chown {用户名称} {文件/文件夹}

修改属组：

chown :{组名} {文件/文件夹}

修改所有者以及属组

chown {用户名}:{组名} {文件/文件夹}

## 3.物理权限(chattr)

修饰某个文件/夹不允许被修改。注意：不能给/、/tmp、/dev、/var这几个目录加保护。这些目录都是受系统所保护以及维护的。

添加了物理权限之后，即便是root权限也不一定所有的文件都可以删。

chattr [选项] file/dir

选项：

|  |  |
| --- | --- |
| i | 表示不能以任何方式进行文件/夹的修改，追加，删除 |
| a | 表示文件/夹只能追加，不能修改，删除 |
| + <属性> | 表示开启某文件/夹的权限 |
| * <属性> | 表示关闭某文件/夹的权限 |
| R | 表示递归处理。 |

### 3.1.不允许修改追加删除文件

新建bigdata\_1空文件，使用i选项添加/移除物理权限。

# bash

touch bigdata\_1

chattr +i bigdata\_1 # 表示该文件不允许修改，删除，增加。

chattr -i bigdata\_1 # 表示关闭改文件的物理权限

### 3.2.只允许追加文件内容

新建bigdata\_2空文件，使用a选项添加/移除物理权限

touch big bigdata\_2

chattr +a bigdata\_2 # 表示该文件只允许追加内容，不允许删除和修改。

chattr -a bigdata\_2 # 表示关闭改文件的物理权限

说明：a、i选项的使用场景比较特殊，在通常情况下，log文件用a的属性。如果是cfg（配置文件）文件用i的属性。

### 3.2.lsattr 查看文件的物理权限以及属性(lsattr)

lsattr [选项] 文件/夹

选项：

|  |  |
| --- | --- |
| R | 表示递归处理 |
| a | 表示查看所有文件的属性，包括隐藏 |
| d | 显示目录的属性，而不是目录下的文件的属性 |

## 4.普通超级权限(sudo)

普通用户的超级权限：了解

sudo(SuperUser Do)，它可以让普通用户执行root的权限。sudo可以限制用户执行部分root的权限。sudo会记录用户执行过的每一条命令，便于查阅服务起出事之前的状态。

可以让普通用户，使用自己配置好的用户环境，不需要知道root密码，保证root的密码安全，可以限制用户执行有限的root权限

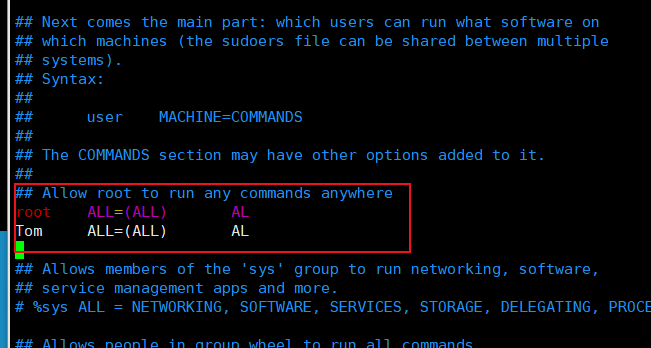
### 4.1.将普通用户配置为超级权限

#配置sudoers核心文件

vim /etc/sudoers

#配置成功之后Tom提升为超级用户，只需要在指令前添加sudo即可执行管理员命令

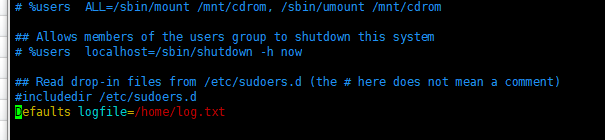
sudo service network restart



### 4.2.配置超级用户的操作日志记录

sudo执行的每条命令都会被记录，便于日后的日志审计，例如用户执行过高危操作命令。想要记录执行过哪些命令需要修改配置文件 /etc/sudoers

添加 Defaults logfile=/home/log.txt





# 二、练习

## 1.练习1

需求：现在大数据部门有一个加密狗，大数据部门的人都可以去连接它给大家上课，但是别的部门不行

1，在/home目录下touch 加密狗文件。

2，创建大数据组

3，修改加密狗文件的所有者为root和大数据组

4，修改加密狗文件的权限为：root用户可读可写可执行，大数据组的成员可读可写可执行，其他人没有权限

4，编辑加密狗文件实现其功能（写一句话就行）

5，创建大数据组的成员，并将其加入到大数据组

通过不同权限的人，来观察加密狗的使用情况

### 1.2.练习1答案

touch UsbKey #创建加密狗文件

groupadd bigdata #创建组

chown :bigdata UsbKey #修改文件属组

chmod 770 UsbKey #修改文件权限

useradd -g bigdata caolaoshi #添加caolaoshi到bigdata组

useradd -g bigdata piaolaoshi #添加piaolaoshi到bigdata组

## 2.练习2

需求：现在2002班有两个组，一个是A组，另外一个是B组。

其中A组成员有zhangsan，lisi。B组成员有wangwu，zhaoliu。

A组成员的家目录在/big2002/A/目录下，且只有A组的人可以进。

B组成员的家目录在/big2002/B/目录下，且只有B组的人可以进。

现在要实现，组内之间资料共享，其他人无权查看。

资料如下：A组有两本书：

book1：所有者和属组为root用户和A组。权限为root用户可读可写可执行，A组的人可读可写可执行，其他人没有权限。

book2：所有者和属组为root用户和A组，权限为root用户可读可写可执行，A组的可读可执行不可写，其他人没有权限。

### 2.1.练习2答案

#创建用户家目录

mkdir -p /big2002/A

mkdir /big2002/B

#创建用户并指定家目录

useradd -d /big2002/A/zhangsan zhangsan

useradd -d /big2002/A/lisi lisi

useradd -d /big2002/B/wangwu wangwu

useradd -d /big2002/B/zhaoliu zhaoliu

#创建用户组：

groupadd -g 1000 A

groupadd -g 1001 B

#将用户分别加入对应的组内

usermod -g A zhangsan

usermod -g A lisi

usermod -g B wangwu

usermod -g B zhaoliu

既然是组内的成员进行共享，那么就应该可以使用权限的方式来实现，先修改组的目录权限为其他人无权限。

#配置A、B两个目录为，用户为读写执行，组为读写执行，其他人没有权限

cd /big2002

chmod 750 A

chmod 750 B

通过ll命令我们能看到A组和B组所有者为root用户和root组。

所以修改A目录的属组为A，B组的属组为B

chown root:A A

chown root:B B

在A组目录中touch book1，并修改权限为770，所有者为root:A，同时创建book2，修改权限为750，所有者为root:A

# 三、磁盘、文件系统

计算机生成了可选文字:
磁 盘 盘 片 
读 写 磁 头 
主 轴 
转 动 釉 
矩 弹 簧 

磁盘，它是由一个个盘片组成的，从盘片的结构上来看 图中的一圈圈灰色同心圆为一条条磁道，从圆心向外画直线，可以将磁道划分为若干个弧段，每个磁道上一个弧段被称为一个扇区（图中绿色部分）。扇区是磁盘的最小组成单 元，常是512字节。

计算机生成了可选文字:
扇 仅 
道 

## 1.磁盘分区：

指定分割区域起始与结束磁柱，磁盘存取的区域，例如A磁柱到B磁柱之间的区块，磁盘在此分割区域内操作系统能够知道它可以在指定区块进行文件读，写，查询等操作，但是需要注意使用硬盘之前需要格式化！，因为每种操作系统所设定的文件属性/权限，以及存放数据的格式 有所不同 。

## 2.文件格式

Ext\*、NTFS和FAT32这三个都是文件系统格式。

## 3.Ext2/Ext3/Ext4区别

Linux kernel自2.6.28开始正式支持新的文件系统Ext4。

Ext4是Ext3的改进版，修改了Ext3中部分重要的数据结构。

Ext3对Ext2，只是增加了一个日志功能。

Ext4可以提供更佳的性能和可靠性，还有更为丰富的功能，更大的文件系统和更大的文件。 较之Ext3所支持的最大16TB文件系统和最大2TB文件，Ext4分别支持1EB（1,048,576TB，1EB=1024PB，1PB=1024TB）的文件系统，以及16TB的文件。

### 3.1.无限数量的子目录

Ext3只支持32,000个子目录，而Ext4支持理论值的无限数量的子目录。

### 3.2.延迟分配

Ext3的数据块分配策略是尽快分配，而Ext4是尽可能地延迟分配，直 到文件在cache中写完才开始分配数据块并写入磁盘。 如此能优化整个文件的数据块分配，显著提升性能。

### 3.3.快速fsck（文件系统检查）

老的fsck会很慢，因为它要检查所有的索引节点(inode)。

Ext4给每个组的索引节点表中添加了一份未使用inode的列表，执行fsck就可以跳过它们而只去检查那些在用的索引。

### 3.3.持久预分配（Persistentpreallocation）

常常会预先创建 一个与所下载文件大小相同的空文件，以免未来的数小时或数天 之内磁盘空间不足导致下载失败。Ext4在文件系统层面实现了持久预分配并提供相应的API，比应用软件自己实现更有效率。

计算机生成了可选文字:
[ eclipse-jee-oxygen-3a-win32-x86 64.zip.xltd 
[ 〕 eclipse-Jee-oxygen-3a-win32-x86_64.zip.xltd.cfg 
2018 / 5 / 16 1 ： 25 
2018 / 5 / 16 1 ： 25 
迅 雷 临 时 数 据 文 
CFG 文 件 
341 ， 8g6 KB 
4 KB 

## 4.SWAP（交换分区）概述

使用磁盘来存储内存不够而“溢出来”的内容(拿硬盘空间来存储内存“溢出”的数据）。 当系统的物理内存不够用的时候，就需要将磁盘没存中的一部分空间释放出来，以供当前运行的程序使用。

最容易成为被释放的对象：一些很长时间没有什么操作的程序。被保存到Swap空间中。等到那些被换出的程序要继续运行时，再从Swap中恢复保存的数据到内存中。

一般来说可以按照如下规则设置swap大小：

•8G以内的物理内存，SWAP 设置为内存的2倍。

•8G-16G以内的的物理内存，SWAP 等于内存大小或者设置为8G。

•16G-256G 的物理内存，SWAP 设置为实际内存的1/2即可。

### 4.1.系统什么时候会使用swap？

实际上，并不是等所有的物理内存都消耗完毕之后，才去使用 swap的空间，什么时候使用是由swappiness 参数值控制。

[root@localhost ~]# cat /proc/sys/vm/swappiness

60

[root@localhost ~]#

默认值是60。swappiness=0的时候表示最大限度使用物理内存 ，然后才是 swap空间，swappiness＝100的时候表示积极的使 用swap分区，并且把内存上的数据及时的搬运到swap空间里面。

### 4.2.如何修改swap参数（sysctl）

临时性修改：

[root@localhost ~]# sysctl vm.swappiness=10

[root@localhost ~]# cat /proc/sys/vm/swappiness

10

这里我们的修改已经生效，但是如果我们重启了系统，又会变成60.

永久修改：

在/etc/sysctl.conf 文件里添加如下参数： vm.swappiness=10

# 四、挂载(mount)

为什么要挂载，因为文件系统并不能够直接使用。Windows的文件系统需要盘符来表示

Linux的文件系统需要目录作为入口。

分区的格式就是文件系统。

挂载语法：

mount 文件系统 目录(挂载点)

## 1.挂载光盘镜像文件

# bash

mkdir /home/cdrom # 此处创建目录cdrom并不是非要这个名称，是因为想做到见名知意。

mount /dev/cdrom /home/cdrom

## 2.挂载U盘

需要注意：U盘的格式如果为NTFS，那么需要安装一个插件之后才能够进行挂载，否则无法识别。

yum install ntfs-3g

如果是fat32的，那么可以直接进行挂载。

# bash

mkdir /home/udisk

# 此处创建目录udisk并不是非要这个名称，是因为想做到见名知意。

lsblk # 用来查看文件系统

mount -o iocharset=utf8 /dev/sdb1 /home/udisk

上面的命令，其中 "-o iocharset=utf8" 是用于解决U盘挂载之后的字符乱码问题。

## 3.取消挂载

umount 挂载点，可以通过df -h的命令查看当前文件系统的状态。如下图：

计算机生成了可选文字:
[root尽locathost一］#df一h
FilesystemSizeUsedAvailUse%Mountedon
/dev/sdaZ18G3.SG14G21%/
tmpfs491M049lM0%/dev/shm
/dev/sdal194M30M155M16%/boot
/dev/5ro4.ZG4.ZG0100%/home/cdrom
/dev/sdb4400M299M102M75%/home/udisk

umount /home/udisk

无法取消挂载：如图

计算机生成了可选文字:
root尽localhostl鱼左可＃umount/home/cdrom/
umount:/home,cdrom：匹巫三工巫互】
(In
the
SOmeCaseS
deVICe15
USefUIinfoab0Ut
foundbylsof(8)
processesthatuse
0rfuser(l))

图中出现的情况是由于当前root账户处于cdrom目录中，所以导致无法取消挂载。还有其他可能是由于别的用户或软件仍在使用该目录中的文件所导致。