Red Hat

RH124 红帽系统管理 I

RH124-06-控制文件访问

Red Hat

一、Linux文件系统权限

Linux文件系统权限

文件只具有三个应用权限的用户类别

- 文件归用户所有，通常是创建文件的用户。
- 文件归单个组所有，通常是创建该文件的主要用户组所有，但是可以进行更改。
- 可以为所属用户、所属组和系统上的非用户和非所属组成员的其他用户设置不同权限。

只有三种权限可应用

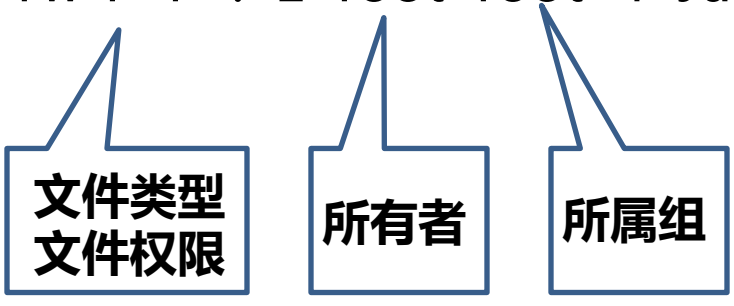
- 读(**r**):用户是否有权限读文件的内容
- 写(**w**):用户是否有权限改变文件内容
- 执行(**x**):用户是否有权限执行文件

文件系统权限

- 文件的一般权限：无、可读、读写（有执行权限的文件一般为第三方应用软件、脚本等）
- 目录的一般权限：无、可读可执行、可读可写可执行。读权限代表可以使用ls命令列出目录内容，执行权限代表可以使用cd命令进入目录
- 写权限赋予文件代表可以更改文件的内容
- 写权限给目录代表可以更改目录中的内容（touch、mkdir、cp、mv、rm）

系统中查看文件/目录的权限和归属

```
[root@xmws ~]# ls -l 11.txt
-rw-r--r--. 1 root root 4 Jul 17 14:35 11.txt
```



权限项	读	写	执行	读	写	执行	读	写	执行
字符表示	r	w	x	r	w	x	r	w	X
数字表示	4	2	1	4	2	1	4	2	1
权限分配	文件所有者(u)			文件所属组(g)			其他用户(o)		

r	w	-	r	-	-	r	-	-
4	2	0	4	0	0	4	0	0
6			4			4		
root			root			其他用户		

Red Hat

二、查看文件/目录权限和所有权

ls -l

查看文件的归属、类型和权限：

- ls -l file：将展开文件列表，以包括文件的权限和所有权
- ls –l directoryname

  显示驻留在该目录中的所有文件的扩展列表
- ls –ld directoryname

  显示目录本身的扩展列表

Red Hat

三、从命令行管理文件系统权限

符号法更改文件/目录权限

命令：chmod

符号法关键字：

chmod whowhatwhich file|directory

- who：u、g、o、a
- what：+、-、=
- which：r、w、x

可以使用-R选项递归修改目录下的所有子项

数值法更改文件/目录权限

数值法：

chmod ### file | directory

- 3个#分别代表访问级别：用户、组、其他
- #是r=4、w=2、x=1的和

更改文件/目录用户或组所有权

- 新建的文件由创建该文件的用户所有。默认情况下，新文件的组所有权为创建该文件的主要用户组
- **只有root用户**可以更改文件的**所有权**
- **root和文件的所有者**可以设置**组所有权**
- root用户可将所有权授予给任何组，而非root用户仅可将所有权授予给他们所属的组

实例：

- chown student foofile　　#将文件foofile所有权授予student
- chown –R student fooflie　　#递归更改整个目录树的所有权
- chown ：admins foodir　　#将foodir组更改为admins
- chown visitor：guests foodir　　#所有权改为visitor，组改为guests

实例

- 用户和组关系

lucy	lucy,ricardo
ricky	ricky,ricardo
ethe1	ethe1,mertz
fred	fred,mertz

- 文件权限和用户/组关系

drwxrwxr-x	ricky	ricardo	dir
-rw-rw-r--	lucy	lucy	lfile1
-rw-r--rw-	lucy	ricardo	lfile2
-rw-rw-r--	ricky	ricardo	rfile1
-rw-r-----	ricky	ricardo	rfile2

- lucy是可以更改lfile1内容的唯一用户
- ricky可以查看lfile2内容，但不能修改
- ethe1可以更改lfile2的内容
- lucy可以更改rfile1内容
- ricky可以查看和修改rfile2内容
- lucy可以查看rfile2内容，但不能修改
- ethe1和fred不能访问rfile2
- ricky可以删除lfile1和lfile2

Red Hat

四、管理默认权限和文件访问

特殊权限

set位权限类型：

- **SUID**：表示对可执行文件setuid权限表示将以文件的用户（或组）身份运行命令，而不是以运行命令的用户身份。

  实例：ls –l /usr/bin/passwd

  chmod u+s 文件

- **SGID**：对某目录setgid表示在该目录中创建的文件将继承目录的组所属关系，而不是继承自创建用户。

  实例：创建目录及目录下的子文件

  chmod g+s 目录

- 粘滞位设置：

  STICKY：目录的粘滞位可以为文件删除设置特殊限制，仅文件所有者和root用户可以删除目录中的文件。

  实例：ls –ld /tmp

  语法：chmod o+t 目录

默认文件权限

- 文件的默认权限由创建它们的进程设置。这些权限通常不是在新的文件和目录创建时予以设置。是因为其中一些权限被shell进程的umask（权限掩码）清除。
- umask用于清除由该进程创建的新文件和目录的权限
- 系统默认umask在**/etc/profile**和**/etc/bashrc**中定义。用户可以在他们的.bash_profile和.bashrc文件中覆盖系统默认值
- **系统用户**的默认umask值为0022。
- 系统默认的目录权限为0755
- 系统默认的文件权限为0644

umask实例

1）创建新文件newfile1和目录newdir1，并观察权限

2）设置umask为007，屏蔽“其他”权限，并观察效果

随堂练习

1、在home家目录下创建一个目录，名为/home/user-text

2、修改user-text的目录的所属组为ussr

3、确保user-text目录的所属组能够在该目录中创建和删除文件

4、禁止其他用户访问user-text目录并切换到其他用户验证

5、修改user-text目录权限为777

6、切换到普通用户并在user-text目录创建1.txt文件

7、切换到root用户并给user-text添加SGID

8、再次切换到普通用用并在user-text目录创建2.txt文件

比较1.txt和2.txt文件权限的区别。


WISDOM NET