

1. 文件权限

- Linux系统对文件的权限有着严格的控制，如果要对某个文件执行某种操作（读、写、执行），必须具有对应的权限。
- 修改文件权限 - chmod命令
- 修改文件所属用户 - chown命令

2. 修改文件权限 - chmod命令

• 2.1 格式

- `chmod [-cfvR] [--help] [--version] mode file...`

• 2.2 常用参数说明

- -R：以递归的方式对当前目录下的所有档案与子目录进行相同的权限变更
- mode：`[ugoa...][[+|=][rwxX]...][, ...]`

1. [ugoa...]

u：表示该文件的拥有者

g：表示与该文件的拥有者属于同一个组(group)的用户

o：other，表示其他以外的人（不属于该用户和用户组）

a：表示所有（包含上面三者）。

2. [+|=]

＋：表示增加权限

-：表示取消权限

=：表示唯一设定权限

3. [rwxX]

r：表示可以读

w：表示可以写入

x：表示可以执行

x：只有当文件为目录，或者其他类型的用户有可执行权限时，才将文件权限设置可执行

4. [file...]

文件列表（单个或者多个文件、目录）

• 2.3 八进制语法

- chmod命令可以使用八进制数来指定权限。
- 文件或目录的权限位是由9个权限位来控制，每三位为一组，它们分别是文件所属用户的读、写、执行，用户组的读、写、执行其它用户的读、写、执行。

| 数字 | 权限 | rwX | 二进制 |
|----|------------|-----|-----|
| 7 | 读 + 写 + 执行 | rwX | 111 |
| 6 | 读 + 写 | rw- | 110 |
| 5 | 读 + 执行 | r-X | 101 |
| 4 | 只读 | r-- | 100 |
| 3 | 写 + 执行 | -wX | 011 |
| 2 | 只写 | -w- | 010 |
| 1 | 只执行 | --X | 001 |
| 0 | 无 | --- | 000 |

• 2.4 举例

1. 设置a.sh文件的所属用户对该文件有执行权限

```
chmod u+x a.sh
```

2. 增加其它用户对a.sh文件的写入和执行权限

```
chmod o+w,o+x a.sh
```

3. 设置所有用户对a.sh文件拥有执行权限

```
chmod ugo+x a.sh
```

```
chmod a+x a.sh
```

4. 使用数字方式

```
chmod 766 a.sh
```

3. 更改文件拥有者 - chown命令

Linux是多用户系统，每个的文件都有拥有者（所有者），如果我们想变更文件的拥有者（利用 chown 将文件拥有者加以改变），一般只有系统管理员(root)拥有此操作权限；

• 3.1 格式

- `chown [-cfhvR] [--help] [--version] user[:group] file...`

• 3.2 常用参数说明

- -R: 处理指定目录以及其子目录下的所有文件
- user: 指定要修改的所属用户

- group : 指定要修改的所属用户组
- file... : 文件或目录

• 3.3 举例

1. 将文件a.sh所属用户和用户组修改为etoak

```
chown etoak:etoak a.sh
```

2. 递归修改a目录所有文件或子目录的所属用户和组修改etoak

```
chown -R etoak:etoak a
```