域	对象：文件1	文件2	文件3	文件4	文件5	文件6	打印机1	绘仪图2
1	Read	Read Write						
2			Read	Read Write Execute	Read Write		Write	
3						Read Write Execute	Write	Write

图9-5 保护矩阵

域	对象：文件1	文件2	文件3	文件4	文件5	文件6	打印机1	绘仪图2	域1	域2	域3
1	Read	Read Write								Enter	
2			Read	Read Write Execute	Read Write		Write				
3						Read Write Execute	Write	Write			

图9-6 将域作为对象的保护矩阵

9.3.2 访问控制列表

在实际应用中，很少会存储如图9-6的矩阵，因为矩阵太大、太稀疏了。大多数的域都不能访问大多数的对象，所以存储一个非常大的、几乎是空的矩阵浪费空间。但是也有两种方法是可行的。一种是按行或按列存放，而仅仅存放非空的元素。这两种方法有着很大的不同。这一节将介绍按列存放的方法，下一章节再介绍按行存放。

第一种方法包括一个关联于每个对象的（有序）列表里，列表里包含了所有可访问对象的域以及这些域如何访问这些对象的方法。这一列表叫做访问控制表（Access Control List，ACL），如图9-7所示。这里我们看到了三个进程，每一个都属于不同的域。A、B和C以及三个文件F1、F2和F3。为了简便，我们假设每个域相当于某一个用户，即用户A、B和C。若用通常的安全性语言表达，用户被叫做主体（subjects或principals），以便与它们所拥有的对象（如文件）区分开来。

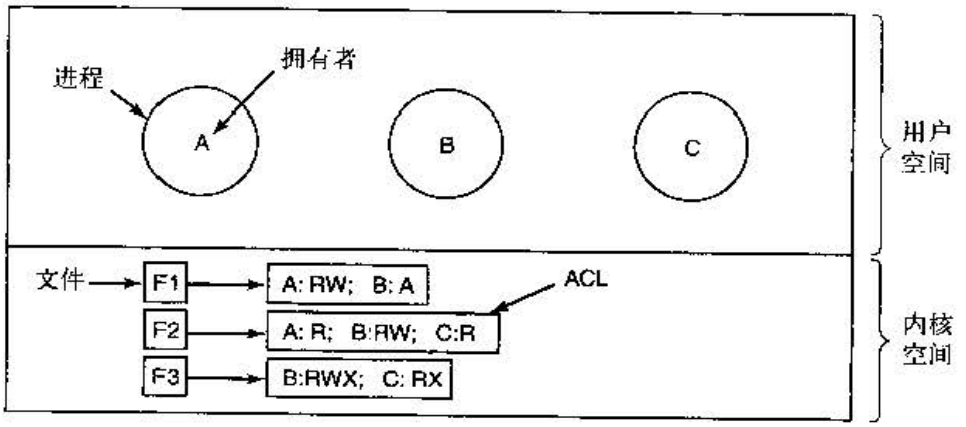

图9-7 用访问控制表管理文件的访问

每个文件都有一个相关联的ACL。文件F1在ACL中有两个登录项（用逗号区分）。第一个登录项表示任何用户A拥有的进程都可以读写文件。第二个登录项表示任何用户B拥有的进程都可以读文件。所有这些用户的其他访问和其他用户的任何访问都被禁止。请注意这里的权限是用户赋予的，而不是进程。只要系统运行了保护机制，用户A拥有的任何进程都能够读写文件F1。系统并不在乎是否有1个还是100个进程。所关心的是所有者而不是进程ID。