对象列表，以及每个对象上可执行操作的指示。这一栏叫做权能字列表（capability list或C-list），而且每个单独的项目叫做权能字（Dennis和Van Horn，1966；Fabry，1974）。一个3进程集和它们的权能字列表如图9-9所示。

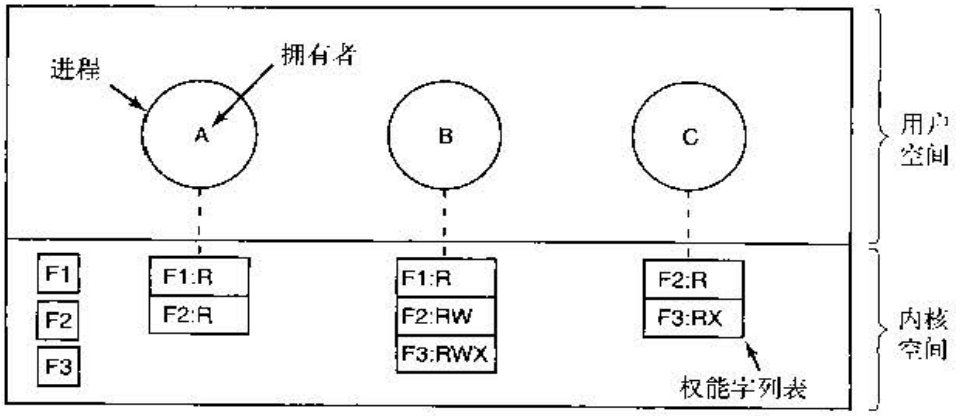

图9-9　在使用权能字时，每个进程都有一个权能字列表

每一个权能字赋予所有者针对特定对象的权限。如在图9-9中，用户A所拥有的进程可以读文件F1和F2。一个权能字通常包含了文件（或者更一般的情况下是一个对象）的标识符和用于不同权限的位图。在类似UNIX的系统中，文件标识符可能是i节点号。权能字列表本身也是对象，也可以从其他权能字列表处指定，这样就有助于共享子域。

很明显权能字列表必须防止用户进行篡改。已知的保护方法有三种。第一种方法需要建立带标记的体系结构（tagged architecture），在这种硬件设计中，每个内存字节必须拥有额外的位（或标记）来判断该字节是否包含了权限字。标记位不能被算术、比较或相似的指令使用，它仅可以被在核心态下运行的程序修改（如操作系统）。人们已经构造了带标记体系结构的计算机，并可以稳定地运行（Feustal，1972）。IBM AS/400就是一个公认的例子。

第二种方法是在操作系统里保存权能字列表。随后根据权能字在列表中的位置引用权能字。某个进程也许会说："从权能字2所指向的文件中读取1KB"。这种寻址方法有些类似UNIX里的文件描述符。Hydra（Wulf等人，1974）采用的就是这种方法。

第三种方法是把权能字列表放在用户空间里，并用加密方法进行管理，这样用户就不能篡改它们。这种方法特别适合分布式操作系统，并可按下述的方式工作。当客户进程发送消息到远程服务器（如一台文件服务器）时，请求为自己创建一个对象时，服务器会在创建对象的同时创建一条长随机码作为校验字段附在该对象上。文件服务器为对象预留了槽口，以便存放校验字段和磁盘扇区地址等。在UNIX术语中，校验字段被存放在服务器的i节点中。校验字段不会返回用户，也决不会被放在网络上。服务器会生成并回送给用户如图9-10格式的权能字。

服务器标识符	对象号	权限	f（对象，权限，校验）

图9-10　采用了密码保护的权能字

返回给用户的权能字包括服务器的标识符、对象号（服务器列表索引，主要是i-node码）以及以位图形式存放的权限。对一个新建的对象来说，所有的权限位都是处于打开状态的，这显然是因为该对象的拥有者有权限对该对象做任何事情。最后的字段包含了对象、权限以及校验字段。校验字段运行在通过密码体制保护的单向函数f上，我们已经讨论过这种函数。

当用户想访问对象时，首先要把权能字作为发送请求的一部分传送到服务器。然后服务器提取对象编号并通过服务器列表索引找到对象。再计算f（对象，权限，校验）。前两个参数来自于权能字本身，而第三个参数来自于服务器表。如果计算值符合权能字的第四个字段，请求就被接受，否则被拒绝。如果用户想要访问其他人的对象，他就不能伪造第四个域的值，因为他不知道校验字段，所以请求将被拒绝。

用户可以要求服务器建立一个较弱的权能字，如只读访问。服务器首先检查权能字的合法性，检查成功则计算f（对象，新的权限，校验）并产生新的权能字放入第四个字段中。请注意原来的校验值仍