

(即/etc/rc)的进程,这个进程可以进行文件系统一致性检测、挂载附加文件系统、开启守护进程等。然后这个进程从/etc/ttys中读取数据,其中/etc/ttys列出了所有的终端和它们的属性。对于每一个启用的终端,这个进程调用fork函数创建一个自身的副本,进行内部处理并运行一个名为getty的程序。

getty程序设置行速率以及其他的行属性(比如,有一些可能是调制解调器),然后在终端的屏幕上输出:

login:

等待用户从键盘键入用户名。当有人坐在终端前,提供了一个用户名后,getty程序就结束了,登录程序/bin/login开始运行。login程序要求输入密码,给密码加密,并与保存在密码文件/etc/passwd中的加密密码进行对比。如果是正确的,login程序以用户shell程序替换自身,等待第一个命令。如果是不正确的,login程序要求输入另一个用户名。这种机制如图10-11所示,该系统具有三个终端。

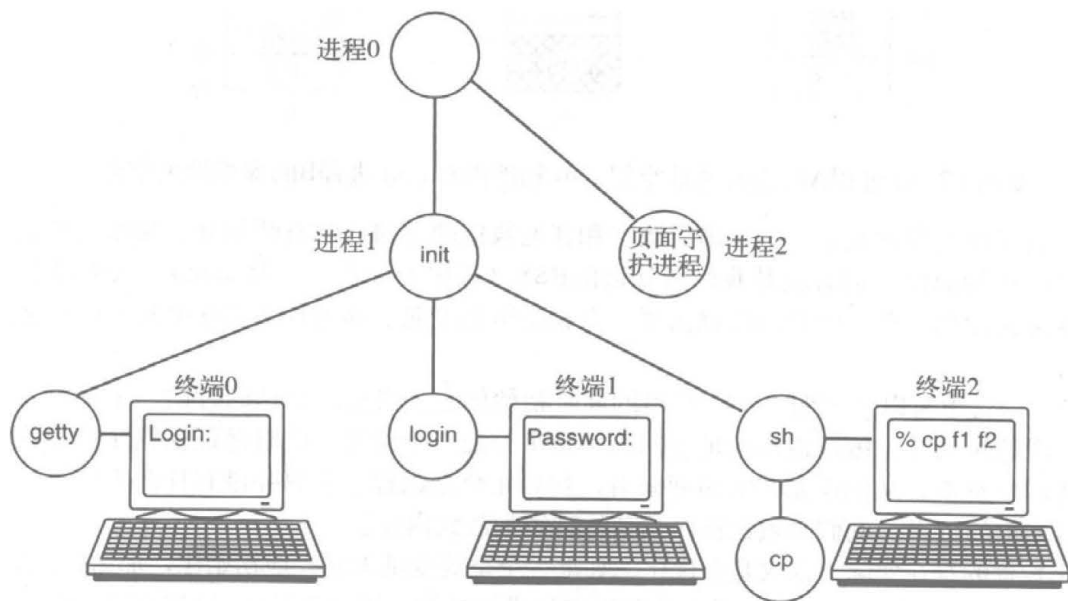


图10-11 用于启动一些Linux系统的进程顺序

在图中,0号终端上运行的getty程序仍然在等待用户输入。1号终端上,用户已经键入了登录名,所以getty程序已经用login程序替换掉自身,目前正在等待用户输入密码。2号终端上,用户已经成功登录,shell程序显示提示符(%).然后用户输入

cp f1 f2

shell程序将调用fork函数创建一个子进程,并使这个子进程运行cp程序。然后shell程序被阻塞,等待子进程结束,子进程结束之后,shell程序会显示新的提示符并且读取键盘输入。如果2号终端的用户不是键入了cp命令而是cc命令,C语言编译器的主程序就会被启动,这将生成更多的子进程来运行不同的编译过程。

## 10.4 Linux中的内存管理

Linux的内存模型简单明了,这样使得程序可移植并且能够在内存管理单元大不相同的机器上实现Linux,比如:从没有内存管理单元的机器(如,原始的IBM PC)到有复杂分页硬件支持的机器。这一块设计领域在过去数十年几乎没有发生改变。下面要介绍该模型以及它是如何实现的。

### 10.4.1 基本概念

每个Linux进程都有一个地址空间,逻辑上有三段组成:代码、数据和堆栈段。图10-12a中的进程A就给出了一个进程空间的例子。代码段包含了形成程序可执行代码的机器指令。它是由编译器和汇编器把C、C++或者其他程序源码转换成机器代码而产生的。通常,代码段是只读的。由于难以理解和调试,自修改程序早在大约1950年就不再时兴了。因此,代码段既不增长也不减少,总之不会发生改变。