800页 我真的想去si

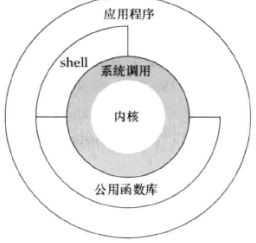
不多说了，我去看书了==。==

UNIX环境高级编程

第一章：UNIX基础知识

引言：所有操作系统都为它们所运行的程序提供服务，典型的包括执行新程序，获得时间，分配存储，打开文件等

UNIX体系结构：操作系统定义为一种控制计算机硬件资源，提供程序运行环境的软件，通常将这种软件称为内核。内核的接口被称作系统调用，公用函数库构建在系统调用接口上，应用程序即可使用公用函数库，也可使用系统调用，shell是一个特殊的应用程序，为运行其他应用程序提供了一个接口。广义上的操作系统包括内核和一些使得计算机能够发挥作用，并使计算机具有自己的特性，其他软件包括系统实用程序、应用程序、shell、公共函数库等。



登录：登录名，shell（有许多不同的种类），所有的linux系统都使用GNU shell

文件系统：为层次结构，起点为根（root “/”）,

文件名：字母，数字，句点，短横线，下划线，创建新目录时会自动创建两个新文件名. ..指向当前目录和父目录

路径名：斜线分隔的一个或多个文件名组成的序列构成路径，以斜线开头的路径名称为绝对路径名，否则为相对路径名

工作目录：每个进程都有工作目录

起始目录：登录时，工作目录设为起始目录

文件描述符：通常是一个小的非负整数，内核用来标识一个特定进程正在访问的文件，当内核打开一个现有文件或创建一个新文件，都返回这个，读写文件可以使用这个

标准输入、输出、错误：如果不做特殊处理，这三个描述符都链接向终端，大多数shell都提供一种办法让描述符重新定位到文件，如ls > file.list

不带缓冲的I.O：open、read、write、lseek以及close、

标准IO：

程序：存储在磁盘上某个目录中的可执行文件。内核使用exec函数将程序入读内存执行程序

进程：程序的执行实例

进程ID：一个非负整数，进程的唯一标识符

进程控制：fork exec waitpid

线程：通常一个进程只有一个控制线程，某一时刻执行的一组机器指令。一个进程内的所有线程共享同一地址空间、文件描述符、栈以及进程相关的属性。因为它们能访问同一存储区，所以要各线程访问共享数据时需要采取同步措施避免不一致性

线程ID：只在它所属的进程内起作用。

用户ID：一个数值向系统标识各个不同的用户。系统管理员在确定一个用户的登录名的同时，确定其用户ID，用户不能更改用户ID

组ID：口令文件登录项包括用户的组ID，它是一个数值。

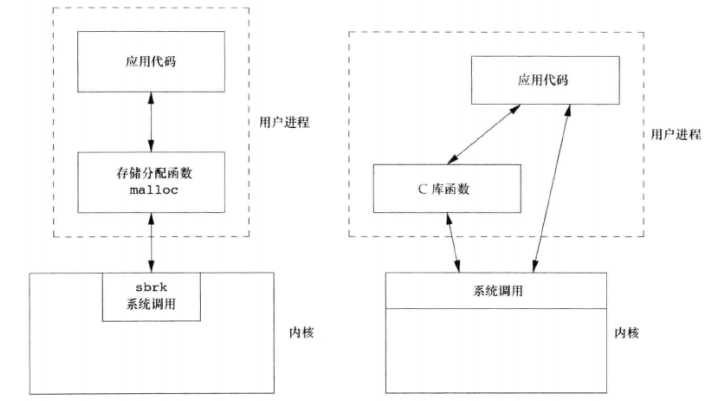
附属组ID：除了在口令文件中对一个登录名指定一个组ID外，大多数UNIX系统版本还允许一个用户属于另外一些组

信号：信号通知进程发生了某种情况。例如若某一进程执行除法操作，除数为0，则将名为SIGFPE的信号发送给该进程。进程有一下三种处理信号的方式1、忽略哦2、按照默认方式处理3、提供一个函数信号发生时调用这个函数。

UNIX使用过两种不同的时间值：1、日历时间2、进程时间。 度量进程执行时间有三个，时钟时间，用户CPU时间，系统CPU时间

系统调用：指良好定义，数量有限，直接进入内核的入口点

库函数：UNIX使用的技术是为每个系统调用在标准C库中设置一个具有同样名字的函数。



系统调用通常提供一种最小接口，而库函数通常提供比较复杂的功能。

第二章：UNIX标准及实现