

# 系统调用和程序接口

翻译 cany1000 2016-06-09 09:29:52 4457 已收藏 11

分类专栏: 操作系统



操作系统 专栏收录该内容

5 订阅

57 篇文章

订阅专栏

## 1、系统调用

系统调用是操作系统内核和用户态运行程序之间的接口，它把用户程序的请求传送至内核，调用相应的内核函数完成所需的处理，将处理结果返回给用户程序。

系统中各种共享资源都由操作系统统一管理，因此在操作系统的外层软件或用户程序中，凡是与资源有关的操作（如存储分配、I/O等）都必须通过系统调用的方式向操作系统提出服务请求，并由操作系统代为完成，所以系统调用是用户程序获得操作系统服务的唯一途径。

---- 操作系统提供的系统调用很多，按功能分为：

进程和作业管理、文件操作、设备管理、内存管理、信息维护和进程通信系统调用六大类。

---- 操作系统实现系统调用功能的机制称为陷阱或异常处理器机制。

由于系统调用而引起处理器中断的机器指令称为访管指令、自陷指令或中断指令。

其中，访管指令为非特权指令，在用户态下执行时会将CPU转换为内核态。

----- 系统调用的实现要点：

---- 1) 编写系统调用处理内核函数；

---- 2) 设计一张系统调用入口地址表，每个入口地址都指向一个系统调用的处理内核函数，有的系统还包含系统调用自带参数的个数；

---- 3) 陷阱处理器机制需开辟现场保护区，以保存发生系统调用时的处理器现场。

## 2、程序接口

尽管应用程序可以直接使用系统调用完成相关的功能，但系统调用的参数复杂，使用困难，所以操作系统通常提供了相应的程序接口，如UNIX/Linux在标准C函数库中为每个系统调用构造一个同名的封装函数，由它来屏蔽下层的复杂性，负责把操作系统的服务接口即系统调用封装成应用程序能够直接使用的API。API（Application Programming Interface）应用程序接口：是一些预先定义的函数，目的是提供应用程序与开发人员基于某软件或硬件得以访问一组例程的能力，而又无需访问源码，或理解内部工作机制的细节。

---- 应用程序采用标准C调用序列来调用封装函数，封装函数又按系统所要求的形式和方法传递参数，执行访管指令，调用内核相应的内核函数。

---- 一个库函数（封装函数）就是一种API，它介于应用程序和操作系统之间。API是一个函数定义，说明如何获得一个给定的服务。一个API的实现可能会用到一个或多个系统调用，此时库函数调用系统调用完成其部分工作(如fprintf())库函数也可能与系统调用无关(如strcpy())。

---- 库函数与系统调用的联系和差别如下：

- 1) 从应用程序使用角度来看，库函数与系统调用没有差别。从实现的角度来看，两者有重大区别。
- 2) 使用库函数的好处是隐藏访管指令细节，使系统调用更像函数调用。对用户隐蔽系统调用，使得在改动内核时，不影响应用程序的正确性。
- 3) 库函数属于应用程序，在用户态运行，系统调用属于系统程序，在内核态运行。
- 4) 如果需要的话，用户可以替换库函数，但通常不能替换系统调用。



cany1000

关注

5

1