

## 操作系统作业

毛浩宇 PB16061023

### 1. 什么是系统调用？我们为什么需要系统调用？

根据上一章的内容，为了保证系统的稳定与安全，需要操作系统提供特定的应用程序接口（API）来执行系统调用，以此限制用户的操作；一些 I/O 设备的控制十分复杂，需要操作系统提供接口，通过操作系统执行系统调用来简化程序员的工作。系统调用是操作系统与计算机硬件设施沟通的方式，应用程序通过 API 这一接口来进行系统调用。同一套 API 往往可以在同一操作系统的不同版本上工作，使得应用程序具备很好的移植性。使用了 API 之后，应用程序不需要管操作系统执行了什么，只需要调用 API 就行了，这对程序员来说十分友好。

### 2. 如何向操作系统传递参数？

应用程序要进行系统调用，就要向操作系统传递参数。一种最自然的方法就是把参数值放入寄存器中。但是参数的数量往往会比寄存器的个数多。这时可以把存放参数放在内存里，让操作系统通过获取地址值间接获取参数；一些操作系统会要求应用程序把参数存入块或堆栈中，操作系统从堆栈中获取参数。

### 3. 系统调用有哪些种类？

系统调用大致可分成五大类：进程控制、文件管理、设备管理、信息维护和通信。下面依次介绍这五大类系统调用。

#### 进程控制

一个应用程序在执行完毕之后需要结束，应用程序在运行过程中遇到错误时有时也需要结束，这时就需要一个控制进程结束，并返回程序当前状态值的操作。通过状态值可以得知程序结束时遇到的错误级别。

运行一个进程的时候可能需要载入或执行另一个程序。如果执行完另一个程序之后还要回到原有进程继续执行，就必须要把现有进程的状态信息完整地保存下来。如果要创建另一个进程，系统调用就应当可以控制它的执行。

进程在运行过程中往往要等待 I/O 设备的信号，或者等待其他进程的响应。这时需要操作系统提供等待功能，以使应用程序正确地运行。

#### 文件管理

用户通常要对存放在操作系统里的文件进行修改或整理。这时需要操作系统为文件管理提供相应的系统调用与接口。操作系统首先需要能创建和删除文件，并读取和修改其属性（比如文件名）。用户需要读写文件，因此操作系统需要提供打开关闭文件、读写文件以及重定位读写位置的操作。操作系统对目录也应该具备同样的操作。

#### 设备管理

程序在运行过程中需要资源，因此操作系统需要对计算机的资源进行合理的分配，以使每一个进程都有合适的资源分配，达到运行效果的最大化。

除了资源以外，操作系统还要对一些其他的物理设备或虚拟设备进行有效的管理。操作系统要对设备进行合理的分配，以保证某一设备在同一时刻只有一个用户在控制它；在用户使用完设备之后，操作系统应当断开用户与设备之间的关系。有时一些 I/O 设备的控制比较复杂，操作系统需要提供特定的接口，以简化程序员的操作。

#### 信息维护

操作系统应当知道计算机当前处于什么样的状态。操作系统应该保证其存储的当前计算机的状态（比如系统版本，内存大小，磁盘剩余空间等）是最新的，而且在用户需要这些信息时，操作系统应当能立即提供。此外，对于一些因为出现错误而异常结束的进程，操作系统应当保存错误信息，分析错误原因，以便开发人员予以修补。有的操作系统还提供了时间中断用于调试。

#### 通信

进程之间通过消息传递或共享内存来传递信息。当进程间通过消息传递来通信时，进程向操作系统提供目标进程的进程名与 PID，然后操作系统打开通道，通信开始。如果通信完毕，操作系统会把通信通道关闭。因为消息传递实际上是数据的复制与转移，需要占用一定的资源，因此这种方式一般只适用于小量数据。当进程之间通过共享内存来通信时，操作系统要避免两个进程同时访问用一块内存，以避免访问冲突。在这种通信方式中，数据实际上并没有被转移，因此这种通信方式适合较大量数据的传递。

### 4. 总结

以上就是系统调用的主要分类。系统调用的详细信息以及具体的使用方法将在之后的章节里逐一介绍。