

# Linux系统编程-目录

2024年5月18日 11:16

# 基本概念

2024年5月18日 11:17

## 一、基本概念

- 1.1 操作系统
- 1.2 库函数和系统调用
- 1.3 内核
- 1.4 程序和进程

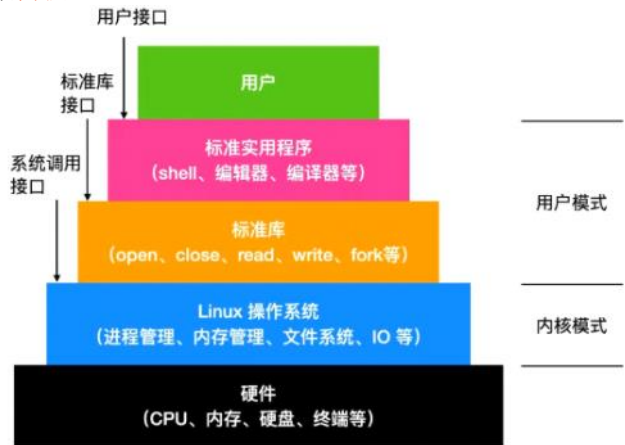
## 一、 基本概念

### 1.1 操作系统

操作系统（operating system, OS）：是管理硬件资源和软件资源的计算机程序。

操作系统通俗解释：计算机是由各种硬件组成，如果没有操作系统，开发人员就需要熟悉对所有计算机硬件的使用，掌握所有硬件的细节，这样程序员就不用写代码了。所以就给计算机安装了一个软件，称为操作系统，能为用户程序提供一个更简单，方便的计算机模型。本质上操作系统也是一个程序，最大的作用就是管理硬件资源。

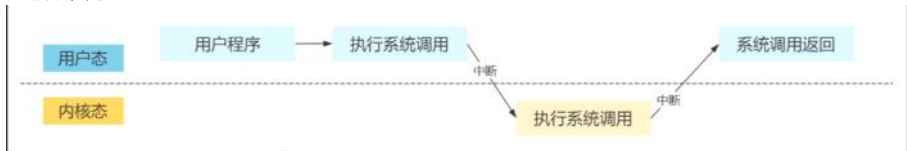
其中管理硬件资源的程序叫做**内核**。



因此，操作系统一般分为两种模式：**用户模式**（用户态）、**内核模式**（内核态）  
一般情况下，内核态可以访问硬件资源，用户态不能访问硬件资源。

例如：printf() 函数在用户态下是无法打印输出的，在函数内部进行了系统调用write() 函数，进入了内核态

可以通过**系统调用**进行内核态。



流程：用户利用程序（开发工具）调用库函数（printf），库函数内部进行系统调用，然后进入内核态，内核态下执行系统调用，返回系统调用，回到用户态。

### 1.2 库函数和系统调用

本质上都是函数

**库函数**：是把函数放到库里，方便其他人使用。目的就是把常用的一些函数放到一个文件了，方便重复使用。一般放在lib文件中。

**系统调用**：是内核的入口，是为了能够进入内核的一个函数。如果需要进入内核空间，就需要通过系统调用，当程序进行系统调用时，会产生一个中断，CPU会中断当前执行的应用程序，跳转到中断处理程序，也就是开始执行内

核程序，内核处理完成后，主动触发中断，把CPU执行权交回应用程序。

使用man手册查看系统调用

```
1  可执行程序或 shell 命令
2  系统调用(内核提供的函数)
3  库调用(程序库中的函数)
4  特殊文件(通常位于 /dev)
5  文件格式和规范, 如 /etc/passwd
6  游戏
7  杂项 (包括宏包和规范), 如 man(7), groff(7), man-pages(7)
8  系统管理命令(通常只针对 root 用户)
Manual page man(1) line 1 (press h for help or q to quit)
```

补全man手册

```
sudo apt-get install manpages-dev glibc-doc manpages-posixmanpages-posix-dev
```

### 1.3 内核

内核是操作系统的核心，是应用程序与硬件直接的桥梁，应用程序只关心与内核的交互，不用关系硬件的细节

内核的能力：

- (1) 进程调度：管理进程、线程，决定哪个进程、线程使用CPU
- (2) 内存管理：决定内存的分配和回收
- (3) 提供文件系统：提供文件系统，允许对文件进行创建、获取、更新以及删除等操作。
- (4) 管理硬件设备：为进程与硬件设备直接提供通信能力
- (5) 提供系统调用接口：进程可以通过内核入口（也就是系统调用），请求去内核执行各种任务。

内核具有很高的权限，可以控制CUP、内存、硬盘等硬件。而应用程序权限很小，因此大多数操作系统一般将内存分为两个区域：

内核空间：只有内核程序才能访问

用户空间：专门给应用程序使用

因此CUP可以在两种状态下运行：用户态、内核态，  
用户态下的CPU，只能访问用户空间的内存  
内核态下，内核空间内存与用户空间内存都能访问

### 1.4 程序和进程

**程序**：所有的程序都必须能运行。使用的软件就是程序，或者是自己写的代码经过编译和链接处理，得到计算机能够理解和执行的指令（可执行文件）。

**进程**：进程是正在执行程序实例。是操作系统进行资源分配和调度的基本单位。在内核看来，进程是一个个实体，内核需要在他们之间共享各种计算机资源。例如内存资源：内核会在程序开始时为进程分配一定的内存空间，并统筹该进程和整个系统对内存的需求。程序终止时，内核会是否该进程的所有资源，以供其他进程使用。

程序运行起来之后，就叫做进程（运行起来的程序）。