

linux同步与互斥

shui

2014年11月13日

<https://drock.sinaapp.com>

简单说明内核中和用户态中进程和线程同步与互斥的问题。

在用户空间应用程序和内核自身中，如果多个进程共享一个资源，进程彼此在访问共享资源时，会发生竞争的现象，也就会干扰彼此的正常运行。

进程采用"同步"和"互斥"两种方式来解决这个问题。对于"同步"来说，就是对于共享资源，进程必须严格按照一定的先后顺序来运行，而先后顺序依赖于要完成的的任务。（对于读写锁，读锁是可以同时访问的，写锁当前只能有一个进程执行）。对于"互斥"来说，对于共享资源，当前当且仅当只能有一个进程在执行，其他的进程会互相排斥，不能执行，必须等待一个进程执行完毕。对于"同步"和"互斥"，"同步"是一种更复杂的"互斥"，"互斥"是一个特殊的"同步"。

当然我们不得不提到"临界区"这个概念，就是进程的执行在不应该的地方被中断，从而导致进程工作不正确。对于同步和互斥都是围绕这临界区资源来说（临界区特指程序的数据段，而不是代码段）。

我们采用了锁的原理来保证同步和互斥这两种方法。

原子操作是最简单的锁的操作。所谓的原子操作就是不可中断的一个或一系列的操作。