

LinuxKernel2.6进程调度的分析(2)Linux认证考试 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/645/2021_2022_LinuxKerne_c103_645048.htm

三、调度策略 1. 进程优先级 (1) 优先级的计算 前面已经说过，优先级由两部分构成，一是静态优先级static_prio，一是动态优先级prio。静态优先级在进程创建的时候就被赋值，并且不变（除非用系统调用改变进程的nice值）；而进程的动态优先级则是跟static_prio和sleep_avg有关。对于实时进程的优先级在创建的时候就确定了，而且一旦确定以后就不再改变，所以下面部分仅对于非实时进程而言。具体的计算由函数effective_prio()（kernel/sched.c）完成。函数将进程的sleep_avg映射成范围是-MAX_BONUS/2 ~ MAX_BONUS/2的变量bonus，而MAX_BONUS是等于10，可见sleep_avg仅能影响的优先级范围在-5 ~ 5之间。具体的映射是由以下规则完成的：那么进程的动态优先级就等于：（当然必须在MAX_RT_PRIO和MAX_PRIO-1之间）。可见

，sleep_avg和bonus是一个线性关系。进程的sleep_avg越大，bonus越大，从而进程的动态优先级也就越高。(2) 何时计算优先级 计算进程的动态优先级一般调用两个函数，一个是effective_prio()，一个是recalc_task_prio()。函数recalc_task_prio()先要根据进程被唤醒前的状态（即actived）、interactive_credit等来计算进程的sleep_avg（详见"平均等待时间 sleep_avg"一节），在最后调用effective_prio()来计算函数的动态优先级。总的来说，有以下几种情况需要计算进程的优先级： a. 创建新进程，使用函数effective_prio()（因为此时进程尚未进行调度，没有sleep_avg和interactive_credit可言）；

b. 唤醒等待进程时，使用函数`recalc_task_prio()`来计算进程动态优先级。 c. 进程用完时间片以后，被重新插入到`active array`或者`expired array`的时候需要重新计算动态优先级，以便将进程插入到队列的相应位置。此时，使用函数`effective_prio()`； d. 其他情况，如IDLE进程初始化等时候。

2. 进程时间片 (1)

时间片的计算 进程的时间片`time_slice`是基于进程静态优先级的，静态优先级越高（值越小），时间片就越大。计算时间片是同过函数`task_timeslice()`（`kernel/sched.c`）来完成的。该函数也是使用线性映射的方法，将进程优先级`[MAX_RT_PRIO, MAX_PRIO-1]`映射到时间片`[MIN_TIMESLICE, MAX_TIMESLICE]`范围内。通过优先级来计算时间片的等式为：
$$\text{timeslice} = \text{MIN_TIMESLICE} + ((\text{MAX_TIMESLICE} - \text{MIN_TIMESLICE}) * (\text{MAX_PRIO} - 1 - (p) - \text{static_prio}) / (\text{MAX_USER_PRIO} - 1))$$

(2) 何时计算时间片

当就绪进程的所有进程的时间片都是0的时候，许多操作系统（包括旧版本的Linux）是使用下面的循环来给进程队列计算时间片的：

```
for (each task on the system) {
    recalculate priority.
    recalculate timeslice }
```

这样的循环计算会导致以下问题：循环可能会花很长时间，而且算法的复杂度 $O(n)$ ；计算过程中必须给进程队列和`task_struct`上锁，这样可能导致大量的竞争；因为计算时间不可预计，所以可能给实时进程带来问题；在Kernel 2.6中时间片的计算是分散的，具体的计算既可以用`task_timeslice()`，也可以用其他方法。

a. 进程创建时，将父进程的时间片分一半给子进程，同时父进程的时间片减半。（详见“`sched_fork`”一节）； b. 进程用完时间片以后，需要重新计算时间片，并将进程插入到相应的运行队列。（详

见"scheduler_tick"一节)；c. 进程退出时，根据first_timeslice的值来决定是否将子进程的时间片返还给父进程。（详见"退出调度"一节）。可见Kernel2.6通过分散计算时间片的办法很好解决了上面循环计算所带来的几个问题。100Test 下载频道开通，各类考试题目直接下载。详细请访问 www.100test.com