**Linux进程的管理 <一>获取进程信息函数**

[上一篇](http://www.lupaworld.com/batch.common.php?action=viewspace&op=up&itemid=43156&uid=146267) / [下一篇](http://www.lupaworld.com/batch.common.php?action=viewspace&op=next&itemid=43156&uid=146267)  2008-03-10 23:28:49 / 个人分类：[linux应用](http://www.lupaworld.com/146267/spacelist-blog-itemtypeid-1722.html)

[查看( 525 )](http://www.lupaworld.com/146267/viewspace_43156.html#xspace-tracks) / [评论( 6 )](http://www.lupaworld.com/146267/viewspace_43156.html#xspace-itemreply) / [评分( 15 / 11 )](http://www.lupaworld.com/146267/viewspace_43156.html#xspace-itemform)

  进程又称任务，是一个动态的使用系统资源、处于活动状态的应用程序。  
进程的管理由进程控制块PCB、进程调度、中断管理、任务队列等组成，它是linux文件系统、存储管理、设备管理和驱动程序的基础。  
进程控制块PCB中包含了进程的所有信息，主要包括进程PID、进程所占有的内存区域、文件描述符和进程环境等信息。  
他用task\_struct的数据结构表示,存在于include/linux/sch.h

进程状态及转换  
#define TASK\_RUNNING 0 //运行状态  
#define TASK\_INTERRUPTIBLE 1 //等待状态（可被中断）  
#define TASK\_UNINTERRUPTIBLE 2  //等待状态（不可被中断）  
#define TASK\_STOPPED 4  //停止状态  
#define TASK\_ZOMBIE 8  //睡眠状态  
#define TASK\_DEAD 16  //僵死状态

进程的基本操作，六大类：  
1.获取进程信息函数：主要通过读取进程控制块PCB中的信息。  
(1)getpid()  
功能：用来获取目前进程的进程标识。  
定义函数：pid\_t getpid(void)  
返回值：返回当前进程的进程识别号。  
头文件：#include <unistd.h>

(2)getppid()  
功能：用来获取目前进程的父进程标识。  
定义函数：pid\_t getppid(void)  
返回值：返回当前进程的父进程识别号。  
头文件：#include <unistd.h>

(3)getpgid()  
功能：用来获得参数pid指令进程所属于的组识别号，若参数为0，则返回当前进程的组识别码。  
定义函数：pid\_t getpgid(pid\_t pid)  
返回值：执行成功则返回正确的组识别码，若有错则返-1，错误原因存在于errno中。  
头文件：#include <unistd.h>

(4)getpgrp()  
功能：用来获得目前进程所属于的组识别号，等价于getpgid(0)。  
定义函数：pid\_t getpgrp(void)  
返回值：执行成功则返回正确的组识别码。  
头文件：#include <unistd.h>

(5)getpriotity(void)  
功能：用来获得进程，进程组和用户的进程执行优先权。  
定义函数：int getpriority(int which,int who)  
参数含义：  
which:  
PRIO\_PROCESS   who为进程的识别码  
PRIO\_PGRP     who为进程的组识别码  
PRIO\_USER     who为用户识别码  
返回值：执行成功则返回当前进程的优先级(-20--20),值越小优先级越高。若出错则返-1，原因在errno中。  
头文件：#include <sys/time.h>