

Linux系统环境进程间通信：信号灯（5）PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/144/2021_2022_Linux_E7_B3_BB_E7_BB_c103_144773.htm 注：读者可以尝试一下注释掉初始化步骤，进程在运行时会出现何种情况（进程在申请资源时会睡眠），同时可以像程序结尾给出的注释那样，把该程序编译成两个不同版本。下面是本程序的运行结果（操作系统redhat8.0）：
owners uid is 0owners gid is 0creators uid is 0creators gid is 0the number of entries in semaphore map is 32000
max number of semaphore identifiers is 128 mas number of semaphores in system is 32000 the number of undo structures system wide is 32000 max number of semaphores per semid is 250 max number of ops per semop call is 32 max number of undo entries per process is 32 the sizeof of struct sem_undo is 20 the maximum semaphore value is 32767 now free the resourceremove sem

okSummary：信号灯与其它进程间通信方式有所不同，它主要用于进程间同步。通常所说的系统V信号灯实际上是一个信号灯的集合，可用于多种共享资源的进程间同步。每个信号灯都有一个值，可以用来表示当前该信号灯代表的共享资源可用（available）数量，如果一个进程要申请共享资源，那么就从信号灯值中减去要申请的数目，如果当前没有足够的可用资源，进程可以睡眠等待，也可以立即返回。当进程要申请多种共享资源时，linux可以保证操作的原子性，即要么申请到所有的共享资源，要么放弃所有资源，这样能够保证多个进程不会造成互锁。Linux对信号灯有各种各样的限制，程序中给出了输出结果。另外，如果读者想对信号灯作进一

步的理解，建议阅读sem.h源代码，该文件不长，但给出了信号灯相关的重要数据结构。附录1：struct sem_array如下：/* 系统中的每个信号灯集对应一个sem_array 结构 */struct sem_array {struct kern_ipc_perm sem_perm. /* permissions .. see ipc.h */time_t sem_otime. /* last semop time */time_t sem_ctime. /* last change time */struct sem *sem_base. /* ptr to first semaphore in array */struct sem_queue *sem_pending. /* pending operations to be processed */struct sem_queue **sem_pending_last. /* last pending operation */struct sem_undo *undo. /* undo requests on this array */unsigned long sem_nsems. /* no. of semaphores in array */}.其中，sem_queue结构如下：/* 系统中每个因为信号灯而睡眠的进程，都对应一个sem_queue结构*/struct sem_queue {struct sem_queue * next. /* next entry in the queue */struct sem_queue ** prev. /* previous entry in the queue, *(q->prev) == q */struct task_struct* sleeper. /* this process */struct sem_undo * undo. /* undo structure */int pid. /* process id of requesting process */int status. /* completion status of operation */struct sem_array * sma. /* semaphore array for operations */int id. /* internal sem id */struct sembuf * sops. /* array of pending operations */int nsops. /* number of operations */int alter. /* operation will alter semaphore */}.附录2：union semun是系统调用semctl中的重要参数：union semun {int val. /* value for SETVAL */struct semid_ds *buf. /* buffer for IPC_STAT amp. SETALL */struct seminfo * __buf. /* buffer for IPC_INFO */ //test!!void * __pad.}.struct seminfo {int semmap.int semmni.int semmns.int semmnu.int semmsl.int semopm.int semume.int semusz.int semvmx.int semaem.}.参考文

献： [1] UNIX网络编程第二卷：进程间通信，作者：W.Richard Stevens，译者：杨继张，清华大学出版社。对POSIX以及系统V信号灯都有阐述，对Linux环境下的程序开发有极大的启发意义。 [2] linux内核源代码情景分析（上），毛德操、胡希明著，浙江大学出版社，给出了系统V信号灯相关的源代码分析，尤其在阐述保证操作原子性方面，以及阐述undo标志位时，讨论的很深刻。 [3]GNU/Linux编程指南，第二版，Kurt Wall等著，张辉译 [4]semget、semop、semctl手册 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