

进程通信之消息队列

北京迅为电子有限公司

—Linux系统编程视频教程





消息队列

- 消息队列就是一个消息的链表。可以把消息看作一个记录，具有特定的格式以及特定的优先级
- 对消息队列有写权限的进程可以向其中按照一定的规则添加新消息；对消息队列有读权限的进程则可以从消息队列中读走消息。
- `man 2 msgrcv`
- 函数 `ssize_t msgrcv(int msqid, void *msgp, size_t msgsz, long msgtyp, int msgflg)`
 - 参数 `msqid`：消息队列的标识码
 - 参数 `*msgp`：指向消息缓冲区的指针
 - 参数 `msgsz`：消息的长短
 - 参数 `msgflg`：标志位
 - 返回值：成功返回数据长度，错误返回-1



消息队列

- 函数 `int msgsnd(int msqid, const void *msgp, size_t msgsz, int msgflg)`
 - 参数 `msqid` : 消息队列的标识码
 - 参数 `*msgp` : 指向消息缓冲区的指针, 此位置用来暂时存储发送和接收的消息, 是一个用户可定义的通用结构
 - 参数 `msgsz` : 消息的长短
 - 参数 `msgflg` : 标志位
 - 返回值 : 成功返回0, 错误返回-1



消息队列

- 结构体msgp，是一个标准的通用结构
 - struct msgstru{
long mtype; //大于0
char mtext[nbyte];}
- 函数int msgget(key_t key, int msgflg)
 - 参数“key”：消息队列关联的标识符
 - 参数“msgflg”：消息队列的建立标志和存取权限。IPC_CREAT 如果内核中没有此队列则创建它；IPC_EXCL 当和IPC_CREAT 一起使用时，如果队列已经存在，则失败
 - 返回值：执行成功则返回消息队列的标识符，否则返回-1
- 编写编译运行测试
 - 代码有点多，切分为多个视频



补充

- 函数 `ssize_t msgrcv(int msqid, void *msgp, size_t msgsz, long msgtyp, int msgflg)`
 - 参数 `msgtyp`
 - `msgtyp` 等于 0 , 则返回队列的最早的一个消息
 - `msgtyp` 大于 0 , 则返回其类型为 `mtype` 的第一个消息
 - `msgtyp` 小于 0 , 则返回其类型小于或等于 `mtype` 参数的绝对值的最小的一个消息
 - 参数 `msgflg` : 标志位为 0 , 则表示忽略



谢谢！