**5.1 通信进程观察实验**

**1、实验目的与内容**

在Linux下，用ipcs()命令观察进程通信情况，了解Linux基本通信机制。

**2、实验原理**

Linux IPC继承了Unix System V及DSD等，共有6种机制： 信号(signal)、管道(pipe和命名管道(named piped)、消息队列（message queues）、共享内存（shared memory segments）、信号量（semaphore）、套接字（socket）。

本实验中用到的几种进程间通信方式：

（1）共享内存段（shared memory segments）方式

– 将２个进程的虚拟地址映射到同一内存物理地址，实现内存共享

– 对共享内存的访问同步需由用户进程自身或其它IPC机制实现（如信号量）

– 用户空间内实现，访问速度最快。

– Linux利用shmid\_ds结构描述所有的共享内存对象。

（2）信号量（semaphore）方式

– 实现进程间的同步与互斥

– P/V操作， Signal/wait操作

– Linux利用semid\_ds结构表示IPC信号量

（3）消息队列（message queues）方式

– 消息组成的链表，进程可从中读写消息。

– Linux维护消息队列向量表msgque，向量表中的每个元素都有一个指向msqid\_ds结构的指针，每个msqid\_ds结构完整描述一个消息队列

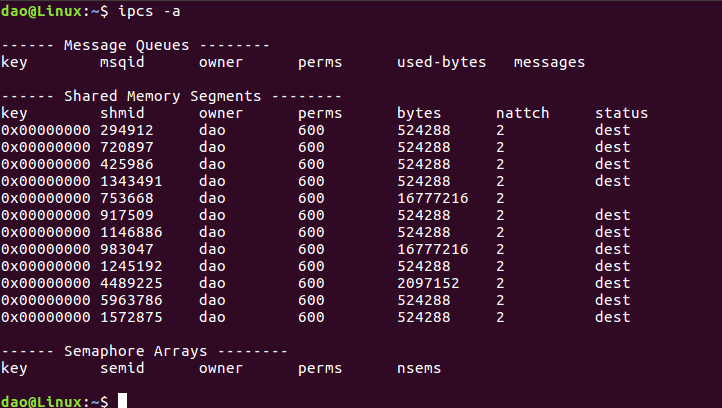
LINUX系统提供的IPC函数有：

* msgget(关键字，方式）：创建或打开一个消息队列
* msgsnd(消息队列标志符，消息体指针，消息体大小，消息类型）：　向队列传递消息
* msgrcv(消息队列标志符，消息体指针，消息体大小，消息类型）：　从队列中取消息
* msgctl(消息队列标志符，获取／设置／删除，maqid\_ds缓冲区指针）：　获取或设置某个队列信息，或删除某消息队列

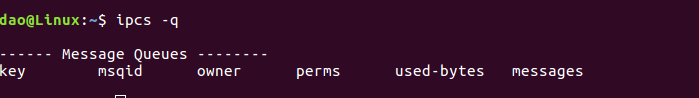
Linux系统中，内核，I/O任务，服务器进程和用户进程之间采用消息队列方式，许多微内核OS中，内核和各组件间的基本通信也采用消息队列方式.

**3、实验步骤**

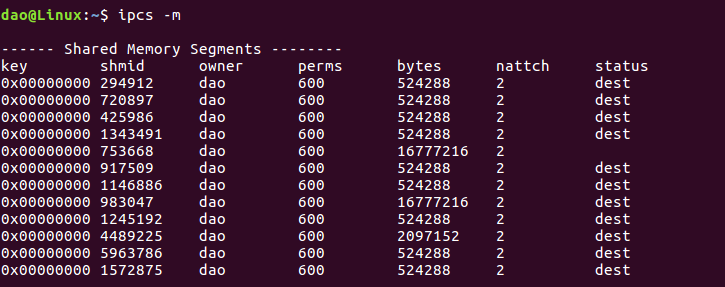
在命令行中使用ipcs -a就可以看到所有进程的情况。



使用-q打印出使用消息队列进行进程间通信的信息



使用-m参数打印出使用共享内存进行进程间通信的信息



使用-s打印出使用信号进行进程间通信的信息

