

Linux的共享内存技术 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/253/2021_2022_Linux_E7_9A_84_E5_85_c103_253021.htm 第一个是写数据： /*****

```
testwrite.c *****/ #include #include #include #include typedef struct{ char name[4]. int age. } people. main(int argc, char** argv) { int shm_id,i. key_t key. char temp. people *p_map. char* name = "/dev/shm/myshm2". key = ftok(name,0). if(key== -1) perror("ftok error"). /*先用ftok创建一个key,再调用shmget，创建一块共享内存区域*/ shm_id=shmget(key,4096,IPC_CREAT). if(shm_id== -1) { perror("shmget error"). return. } /*将这块共享内存区附加到自己的内存段*/ p_map=(people*)shmat(shm_id,NULL,0). temp=a. for(i = 0;i { temp =1. memcpy((*p_map i).name,&temp,1). (*p_map i).age=20 i. } /*写数据完毕，将其从自己的内存段中“删除”出去*/ if(shmdt(p_map)== -1) perror("detach error"). } /*读进程首先要得到那块内存块，这个时候，ftok根据文件名和ID创建的key就很用了，步骤同创建是一样的*/ /***** testread.c *****/ #include #include #include #include typedef struct{ char name[4]. int age. } people. main(int argc, char** argv) { int shm_id,i. key_t key. people *p_map. char* name = "/dev/shm/myshm2". key = ftok(name,0). if(key == -1) perror("ftok error"). shm_id = shmget(key,4096,IPC_CREAT). if(shm_id == -1) { perror("shmget error"). return. } p_map = (people*)shmat(shm_id,NULL,0). for(i = 0;i { printf("name:%s\n",(*p_map i).name ). printf( "age %d\n",(*p_map
```

```
i)).age ). } if(shmctl(p_map) == -1) perror(" detach error "). }
```

注：事实上，ftok函数仅仅是根据文件名和ID生成一个起标识作用的key，根据《GNU Linux编程指南》说，Linux下实现这一函数还有点问题，不推荐用。于是我一般来说都是自己定义一个唯一的key，而不使用ftok函数。

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