

“操作系统原理与实践”实验报告

系统调用

系统调用

实验目的

- 建立对系统调用接口的深入认识
- 掌握系统调用的基本过程
- 能完成系统调用的全面控制
- 为后续实验做准备

实验内容

在linux0.11 添加两个系统调用，并编写简单应用程序测试它们

int iam (const char *name) ; 功能是将字符串name的内容保存下来。要求name的长度不能超过23个字节。返回值是拷贝的字符数。如果name的字符个数超过23，则返回-1，并置errno为EINVAL。

int whoami (char *name,unsigned int size) ;它将内核中由iam()保存的名字名字拷贝到name指向的用户空间，同时保证不对name的越界访存。返回值是拷贝的字符数。如果size小于需要的空间，则返回-1，并置errno为EINVAL。

实验步骤

1.创建kernel/who.c文件，添加系统函数

```
/*
 * file   name:   linux/kernel/who.c
 * author :   justin
 */
#define    __LIBRARY__
#include    <unistd.h>

#include    <errno.h>

#include    <linux/kernel.h>
#include    <asm/segment.h>

//_syscall1(int,          iam,  const   char,   name)

//_syscall2(int,          whoami,char,name,unsigned      int,size)
static    char    namebuf[ 16 ];
static    int namesize;

int  sys_iam  const   char
{
    int  retval    =- 1, cnt= 0,i= 0;
    printk( "sys_iam  is  called\n");

    while ( get_fs_byte(name+cnt)    !=  '\0'    && cnt  < 16 )
        cnt++;
    if (cnt  <= 16){
        for (i= 0; i  < cnt;    i++)
            namebuf[i]=get_fs_byte(name+i);
        retval    = 0;
        namesize=cnt;
    }else {
        errno    = - EINVAL;
        retval=  - 1;
        namesize    = 0;
    }

    return    retval;
}

int  sys_whoami  char          unsigned   int
{
    int  retval    =- 1,i;
    printk( "sys_whoami  is  called\n");

    if (namesize    > 0 && namesize    <= size){
        for (i= 0; i<namesize;    i++)
            put_fs_byte(namebuf[i],    name+i);
        retval    = 0;
    }else {
        retval    = - 1;
        errno    = - EINVAL;
    }

    return    retval;
}
```

2.在include/linux/sys.h将系统函数添加到sys_call_table系统函数调用表

3.在include/unistd.h添加系统调用号。注意系统函数在sys_call_table表的位置与系统调用号是一一对应的。

添加系统函数api接口声明

4.kernel/sys_calls中修改系统调用接口总数nr_sys_calls

5.kernel/makefile增加who.c的编译 （1）objs增加who.o

(2)who.o的依赖

到这里可以到linux0.11目录下执行make编译内核。

编写测试程序 （1）iam.c

(2)whoami.c

7.将iam.c和 whoami.c放到linux0.11的文件系统里 （1）oslab目录下执行sudo ./mount-hdc 将hdc0.11.imag挂载到hdc目录 （2）将iam.c 和whoami.c拷贝到hdc目录下， （3）将include/unistd.h include/linux/sys.h分别拷贝到usr/include,usr/linux目录 （4）sudo umount hdc

8.启动linux0.11并进入文件根目录编译并执行iam和whoami

实验扩展

从linux 0.11现在的机制看，他的系统调用最多能传几个参数？如何扩大这个限制
用文件简要描述想linux 0.11添加系统调用foo () 的步骤。

FAQ