

第6章 简单文件系统设计

《计算机操作系统实验指导》

王红玲 褚晓敏

内容

- Linux文件目录简介
- 文件操作函数介绍
- 实验6.1 文件备份实验
- 实验6.2 简单文件系统的模拟

Linux文件目录简介

Linux文件系统类型

文件系统	描 述
ext	Linux扩展文件系统，最早的Linux文件系统
ext2	第二扩展文件系统，在ext的基础上提供了更多的功能
ext3	第三扩展文件系统，支持日志功能
ext4	第四扩展文件系统，支持高级日志功能
hpfs	OS/2高性能文件系统
jfs	IBM日志文件系统
iso9660	ISO 9660文件系统（CD-ROM）
minix	MINIX文件系统
msdos	微软的FAT16
nep	Netware文件系统
nfs	网络文件系统
ntfs	支持Microsoft NT文件系统
proc	访问系统信息
ReiserFS	高级Linux文件系统，能提供更好的性能和硬盘恢复功能
smb	支持网络访问的Samba SMB文件系统
sysv	较早期的Unix文件系统
ufs	BSD文件系统
umsdos	建立在msdos上的类Unix文件系统
vfat	Windows 95文件系统（FAT32）
XFS	高性能64位日志文件系统

Linux 文件目录简介

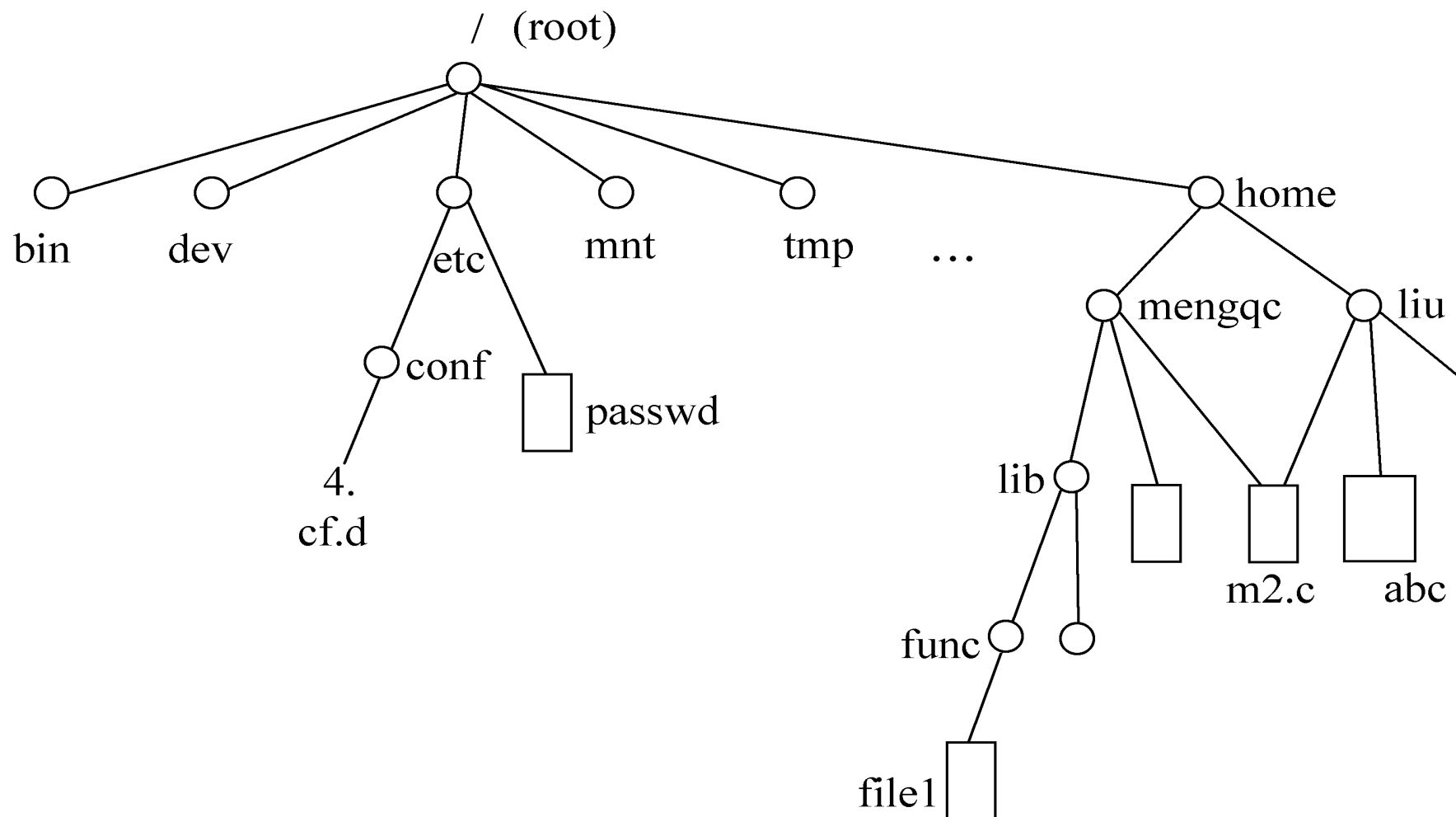


表3-3 常见Linux目录名称

目 录	用 途
/	虚拟目录的根目录。通常不会在这里存储文件
/bin	二进制目录，存放许多用户级的GNU工具
/boot	启动目录，存放启动文件
/dev	设备目录，Linux在这里创建设备节点
/etc	系统配置文件目录
/home	主目录，Linux在这里创建用户目录
/lib	库目录，存放系统和应用程序的库文件
/media	媒体目录，可移动媒体设备的常用挂载点
/mnt	挂载目录，另一个可移动媒体设备的常用挂载点
/opt	可选目录，常用于存放第三方软件包和数据文件
/proc	进程目录，存放现有硬件及当前进程的相关信息
/root	root用户的主目录
/sbin	系统二进制目录，存放许多GNU管理员级工具
/run	运行目录，存放系统运作时的运行时数据
/srv	服务目录，存放本地服务的相关文件
/sys	系统目录，存放系统硬件信息的相关文件
/tmp	临时目录，可以在该目录中创建和删除临时工作文件
/usr	用户二进制目录，大量用户级的GNU工具和数据文件都存储在这里
/var	可变目录，用以存放经常变化的文件，比如日志文件

文件操作函数简介

常用文件操作函数介绍

- Linux系统调用
 - open() :打开文件
 - close() :关闭文件
 - read():从文件读数据
 - write():向文件写数据
 - lseek():定位文件指针
- C语言库函数
 - Fopen()、 fclose()、 fread()、 fwrite()、 fseek()等

实验6.1 文件备份实验

实验6.1 实验目的和实验内容

- 实验目的

- (1) 熟悉Linux 文件系统的文件和目录结构。
- (2) 掌握文件系统的基本特征。
- (3) 掌握常用的文件操作函数。

- 实验内容

编写C程序，模拟实现Linux文件的简单I/O流操作：备份文件，将源文件source.dat备份为target.dat文件。要求：

- (1) 使用C库函数实现文件备份
- (2) 使用系统调用函数实现文件备份

实验6.1 实验指导（1）

对于实验要求（1），涉及的C库函数有fopen()、fclose()、fread()和fwrite()，并需要经过以下步骤：

- （1）使用fopen()函数以只读方式打开想要备份的源文件source和以只写方式打开想要写入内容的目标文件target。
- （2）使用fread()循环读取源文件一个缓冲区大小的内容，使用fwrite()将内容写入目标文件。
- （3）读取与写入完毕，使用fclose()关闭读写文件流。

实验6.1 实验指导（2）

对于实验要求（2），涉及的Linux相关系统调用有有open()、close()、read()和write()，并需要经过以下步骤：

- （1）使用open()系统调用函数以只读方式打开想要备份的源文件source和以只写方式打开想要写入内容的目标文件target。
- （2）使用read()循环读取源文件一个缓冲区大小的内容，使用write()将内容写入目标文件。
- （3）读取与写入完毕，使用close()关闭读写文件流。

实验6.1 实验结果

```
This program backup file based on Linux system calls.  
Success in reading source file.  
Success in close source file.  
Success in close target file.  
[hlwang@localhost program]$ ls -l *.dat  
-rwxrwxr-x. 1 hlwang hlwang 9444 Sep  4 09:25 source.dat  
-rwxrwxr-x. 1 hlwang hlwang 9444 Sep  5 03:17 target.dat
```

```
This program backup file based on C Library.  
Success in reading source file.  
Success in close source file.  
Success in close target file.  
[hlwang@localhost program]$ ls -l *.dat  
-rwxrwxr-x. 1 hlwang hlwang 9444 Sep  4 09:25 source.dat  
-rwxrwxr-x. 1 hlwang hlwang 9444 Sep  5 03:21 target.dat
```

实验6.2 简单文件系统的模拟

实验6.2 实验目的

- ① 理解文件存储空间的管理、文件的物理结构和目录结构以及文件操作的实现。
- ② 加深对文件系统内部功能和实现过程的理解。

实验6.2 实验内容

- 模拟实现一个简单的二级文件管理系统，要求做到以下几点：

(1) 可以实现常用文件目录和文件操作，如：

login 用户登录

dir 列文件目录

create 创建文件

delete 删除文件

open 打开文件

close 关闭文件

read 读文件

write 写文件

(2) 列目录时要列出文件名、物理地址、保护码和文件长度

(3) 源文件可以进行读写保护

实验6.2 设计思路

- 采用两级目录
 - 第一级对应于用户账号
 - 第二级对应于用户账号下的文件
 - 不考虑文件共享、文件系统安全及特殊文件等内容
- 设计时，确定文件系统的数据结构
 - 主目录：链表的形式存放
 - 子目录：链表的形式存放
 - 活动文件：以编号存储于磁盘上，并在目录中进行登记

实验6.2 主要数据结构 (1)

- 磁盘块结构

```
typedef struct distTable
```

```
{
```

```
    int maxlength;                //容量
```

```
    int start;                    //起始地址
```

```
    int useFlag;                  //是否被使用
```

```
    struct distTable *next;       //指向下一块的指针
```

```
} diskNode;
```

实验6.2 主要数据结构 (2)

- 文件块结构（即FCB）

```
typedef struct fileTable
```

```
{
```

```
    char fileName[10]; //文件名
```

```
    int strat;         //文件在磁盘存储空间的起始地址
```

```
    int length;        //文件内容长度
```

```
    int maxlength;     //文件的最大长度
```

```
    char fileKind[3];  //文件的属性——读写方式
```

```
    struct tm *timeinfo; //文件相关的时间信息
```

```
    bool openFlag;     //判断是否有进程打开了该文件
```

```
}fileTableN;
```

实验6.2 主要数据结构 (3)

- 用户文件目录UFD

```
typedef struct user_file_directory
{
    struct fileTable *file;      //文件
    struct user_file_directory *next; //下一个用户文件目录
}UFD;
```

- 主文件目录MFD

```
typedef struct master_file_directory
{
    char userName[10]; //用户账号
    char password[10]; //用户密码
    UFD *user;        //用户文件目录
}MFD;
```

实验6.2 主要数据结构 (4)

- 磁盘块结构

```
#define MaxDisk 512*1024
```

```
char disk[MaxDisk]; //模拟512K的磁盘存储空间
```

```
typedef struct distTable //磁盘块结构体
```

```
{
```

```
    int maxlength;
```

```
    int start;
```

```
    int useFlag;
```

```
    struct distTable *next;
```

```
} diskNode;
```

```
diskNode *diskHead;
```

实验6.2 实验结果

```
*****
1、Creat user
2、login
*****
Please chooce the function key:>1
请输入用户名：user1
请输入密码：123
***
创建用户成功
*****
1、Creat user
2、login
*****
Please chooce the function key:>2
```

```
*****
```

Please chooce the function key:>2

请输入用户名:user1

user1

请输入密码:123

用户登录成功

create-创建 格式：create a1 1000 rw,将创建名为a1,长度为1000字节可读可写的文件

rm-删除 格式：rm a1,将删除名为a1的文件

cat-查看文件内容 格式：cat a1,显示a1的内容

write-写入 格式：write a1

fine-查询 格式：fine a1 ,将显示文件 a1的属性

chmod-修改 格式：chmod a1 r,将文件a1的权限改为只读方式

ren-重命名 格式：ren a1 b1 ,将a1改名为b1

dir-显示文件 格式：dir aaa,将显示aaa用户的所有文件

df-显示磁盘空间使用情况 格式：df

close-关闭文件 格式：close a1,将关闭文件a1

return-退出用户，返回登录界面

exit-退出程序

please input your command:>|