1 文件系统的特点

像windows一样，linux也需要在一个分区上存放系统启动所需的文件，如内核镜像文件、内核启动后运行的第一个程序、给用户提供操作界面的shell程序、应用程序依赖的库等。这些必需的、基本的文件合称为跟文件系统。

2根文件系统的目录结构

3 linux文件属性

linux系统有以下几种文件类型

普通文件

目录文件

字符设备文件

块设备文件

FIFO 用于进程间通讯

套接口 用于进程间的网络通讯

连接文件 指向另一个文件，类似快捷方式

使用ls-lih可以查看文件的具体属性

busybox

内核启动的第一个进程是init，init根据配置文件决定启动哪些程序，如执行哪些脚本、启动shell或运行用户程序

init进程执行的通常是/sbin/init

内核如何启动init进程

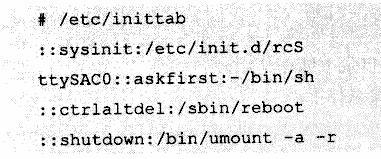
内核启动最后一步是启动init进程，

构建文件系统

1 构建etc目录

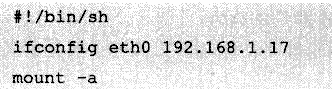
init进程根据/etc/inittab文来创建其他子进程，如调用脚本文件配置IP地址、挂载其他文件系统、最后启动shell。etc目录下的内容取决于要运行的程序，我们只需要etc/inittab、etc/init.d/rcS、etc/fstab

* 1. 创建etc/inittab文件



* 1. 创建etc/init.d/rcS

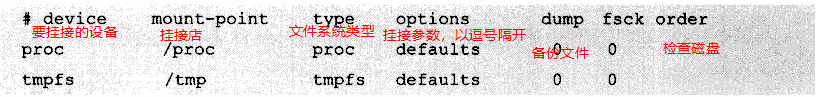
rcS是一个脚本文件，里面的指令是自动执行的指令





* 1. 创建etc/fstab

fstab里的内容表示执行“mount -a”命令后将挂接proc、tmpfs文件系统



busybox->init\_main()

parse\_inittab

file = fopen(INITTAB, "r");//打开配置文件

new\_init\_action() //1 new一个init\_action结构，并把参数赋值于它

//2 把结构体放入链表

run\_actions(SYSINIT);

waitfor(a, 0); //执行应用程序，等待其执行完毕

delete\_init\_action(a); //在init\_action\_list链表里删除

run\_actions(WAIT);

waitfor(a, 0); //执行应用程序，等待其执行完毕

delete\_init\_action(a); //在init\_action\_list链表里删除

run\_actions(ONCE);

run(a) //创建进程但不等待其运行结束;

delete\_init\_action(a); //链表里删除节点

while(1){

run\_actions(RESPAWN);

if (a->pid == 0) {

a->pid = run(a);

}

run\_actions(ASKFIRST);

wpid = wait(NULL) ; //等待子进程

while (wpid > 0) {

a->pid = 0; //子进程完成后pid清0

}

}

配置文件：指定程序 何时执行

inittab格式

id =>/dev/id，用作终端：stdin,stdout,stderr:printf，scanf,err

runlevels:忽略

action：执行时机

process:应用程序或脚本

static void new\_init\_action(int action, const char \*command, const char \*cons)的作用

1 new一个init\_action结构，并把参数赋值于它

1. 把结构体放入链表

最小根文件系统

/dev/console

/dev/null

init -> busybox

/etc/inittab

配置文件中指定的应用程序

C库