操作系统课程设计

1552729 李依璇

1652690 苏昭帆

**同济大学**

目录

**[一、 团队介绍](#_Toc15429_WPSOffice_Level1)** **[3](#_Toc15429_WPSOffice_Level1)**

**[二、 开发环境及安装说明](#_Toc17058_WPSOffice_Level1)** **[3](#_Toc17058_WPSOffice_Level1)**

[1. 开发环境](#_Toc17058_WPSOffice_Level2) [3](#_Toc17058_WPSOffice_Level2)

[2. 开发语言](#_Toc9044_WPSOffice_Level2) [3](#_Toc9044_WPSOffice_Level2)

[3. 安装说明](#_Toc7889_WPSOffice_Level2) [3](#_Toc7889_WPSOffice_Level2)

**[三、 操作系统功能设计](#_Toc9044_WPSOffice_Level1)** **[3](#_Toc9044_WPSOffice_Level1)**

[1. 总体功能](#_Toc20492_WPSOffice_Level2) [3](#_Toc20492_WPSOffice_Level2)

[2. 操作系统组成](#_Toc7156_WPSOffice_Level2) [3](#_Toc7156_WPSOffice_Level2)

[3. 具体功能介绍](#_Toc31076_WPSOffice_Level2) [3](#_Toc31076_WPSOffice_Level2)

**[四、 源码中遇到的BUG](#_Toc7889_WPSOffice_Level1)** **[5](#_Toc7889_WPSOffice_Level1)**

[a. 光盘源码 make 报错](#_Toc15814_WPSOffice_Level2) [5](#_Toc15814_WPSOffice_Level2)

[b. Alt+Fn无法切换TTY](#_Toc14846_WPSOffice_Level2) [5](#_Toc14846_WPSOffice_Level2)

[c. GCC编译不通过](#_Toc2799_WPSOffice_Level2) [5](#_Toc2799_WPSOffice_Level2)

[d. 解压cmd.tar时在pwd程序使用write()写进硬盘时卡住](#_Toc8766_WPSOffice_Level2) [5](#_Toc8766_WPSOffice_Level2)

**[五、 修改的源码](#_Toc20492_WPSOffice_Level1)** **[5](#_Toc20492_WPSOffice_Level1)**

**[六、 操作系统使用说明](#_Toc7156_WPSOffice_Level1)** **[6](#_Toc7156_WPSOffice_Level1)**

1. **团队介绍**

**1552729 李依璇**：应用制作、系统时间、TTY操作、开机动画、文档、PPT 50%

**1652690 苏昭帆**：多级文件系统及持久化制作、底层代码修改、部分开机动画、文档、PPT 50%

1. **开发环境及安装说明**
2. **开发环境**

·ubuntukylin-16.04-32bit

·bochs-2.6.9

1. **开发语言**

C语言、汇编语言

1. **安装说明**

在终端进入项目文件之后，依次输入以下命令：

$make image

$cd ./command

$make install

$cd ..

$bochs

1. **操作系统功能设计**
2. **总体功能**

实现了微内核进程间通信，多级文件系统，内存管理并最终在当前终端上显示一个shell的操作系统。

1. **操作系统组成**

boot（引导）、command（应用集）

fs（文件系统）、include（头文件集）

kernel（内核）、lib（可用代码库）

mm（内存调度系统）

1. **具体功能介绍**
2. **多级文件系统**

对Orange fs文件夹下的源码进行修改，使得能够直接向源码中文件操作接口传入多级目录后就能对指定的正确的文件进行操作，而不需要在上层使用一个文件存储目录、对目录进行解析。也就是说，在shell中向open()函数直接传入用户请求的多级目录，就能够返回正确的文件描述符fd。

本项目多级文件系统的底层实现中，文件与原来相同，文件夹结构与原来扁平系统中的根目录相同、大小与普通文件大小相同。

为实现该多级文件系统，在代码中进行了下述修改：

修改fs/link.c：增加了deletedirectory()函数，以实现对文件夹的删除；增加do\_unlink\_my()函数，使得能够根据文件类型（type=0，为文件）进行文件链接或者文件夹链接的删除；增加了recursivedelete()函数，进行递归删除，当索引结点是文件时，直接删除文件，当索引结点是文件夹时则递归删除文件夹。

修改fs/misc.c：不再使用原来的strip\_path()函数，将其改为strip\_path\_dir()函数，能够通过全路径进行文件查找，是分离多级目录结构的核心。

修改fs/open.c：增加函数new\_inode\_dir()、create\_directory()，实现增加新目录、设置目录结点为索引结点的功能，并在创建文件夹之后，能在磁盘上为该文件夹分配位置；增加tempsearch()函数，能够根据路径查找相应的文件的名称、父文件以及根；增加printallchildfiles()，来判断是否打印某一文件夹下的全部文件；更改原来的do\_open()函数，使得当文件不存在于磁盘上时，创建一个新的文件或者文件夹。

修改lib/open.c：增加一个参数，通过消息传递机制将是文件还是文件夹的信息传递到fs。

其他小的修改：比如消息中需要多添加一项FILETYPE（文件or文件夹）等。

最后，本项目参照orange第11章的内容进行了持久化实现，在fs/main.c中的init\_fs()函数初始化文件系统时，每次先读取超级块，如果发现了magic number，则认为分区已经装上了文件系统，否则调用mkfs(),制作一个文件系统。同时在kernel/main.c中的untar()函数中增加一个记号，当看到该记号后，操作系统不再对cmd.tar进行解包。如果解包时文件已存在于磁盘上，则需将原来的文件内容删除，写入新内容，故引入O\_TRUNC，加入到open()的参数中，同时需要修改fs中的do\_open()函数。

1. **多功能控制台**

本项目中的控制台可以完成文件操作，使用常规应用和游戏；进行该控制台的实现需要对kernel/main.c中的shabby\_shell进行修改。控制台中输入的指令分为两类，一类去读取已经安装在硬盘上并读入文件系统的应用（即游戏），如果读取成功则执行该游戏，如果读取失败，即不存在这样的游戏，则会比对系统内部指令，比对成功则执行相应的系统内部指令（例如ls，cd等）。

1. **游戏及应用**

**·五子棋控制台游戏**

使用评估函数算法实现人机对弈，1为用户，2为机器，对战棋盘为8\*8布局，可以随时退出游戏。

**·扫雷控制台游戏**

游戏布局为9\*9的棋盘，棋盘上共有10个雷，用户只有在点开除了雷之外的所有位置才能获得胜利，可以随时退出游戏。

**·计算机控制台应用**

可以进行包括正数和负数在内的二元整数的四则运算，可以随时退出应用。

1. **读取系统时间**

利用消息传递机制实现一个系统调用来获取时间，用户接口是kernel/main.c中的get\_time函数，消息类型为orange自带的GET\_RTC\_TIME。处理函数在kernel/systask.c中的get\_rtc\_time函数中，用以读取操作系统的时间。

1. **进程管理**

能够展示当前正在运行的进程的具体信息，分别展示系统任务和用户进程。

1. **多TTY操作**

通过使用Ctrl+Fn组合键来解决在bochs中无法使用Alt+Fn组合键进行控制台切换的问题，从而实现了TTY的切换。

1. **源码中遇到的BUG**
2. **光盘源码 make 报错**

**原因**： ubuntu 18.04 LTS 64位，书中默认是在32位环境下开发的

**解决**： ubuntu 18.04 不支持32位，固改用ubutun 16.04 LTS 32位进行源码编译

1. **Alt+Fn无法切换TTY**

**原因：**bochs下使用Alt+Fn组合键只会响应Ubuntu的全局快捷键，虚拟机中的操作无法响应。

**解决：**将tty.c文件中的Alt+Fn组合键改为Ctrl+Fn组合键

1. **GCC编译不通过**

**原因**：编译器版本问题

**解决**：在MakeFile中的CFLAGS加上“’-fno-stack-protector’”

1. **解压cmd.tar时在pwd程序使用write()写进硬盘时卡住**

**原因：**并行代码共享数据不同步

**解决：**在kernel/proc.c的

第468行添加disable\_in(p\_who\_wanna\_recv->p\_flage)，

第480行添加enable\_in(p\_who\_wanna\_recv->p\_flags)

1. **修改的源码**
2. 将fs/main.c中的strip\_path(char\* filename, const char\* pathname, struct inode\*\* ppinode)函数更新为strip\_path\_dir(char\* filename, const char\* pathname, struct inode\*\* ppinode)。

**函数作用：**

1. pathname为输入参数，filename以及ppinode均为输出参数。该函数输入一个路径之后，能够得到最终要操作的文件/文件夹名称，存储在filename中；ppinode则是指向filename上一级目录inode的指针。以输入pathname为/a/m/n为例，使用strip\_path\_dir函数之后，filename为m，ppinode为一个指向n的inode指针。
2. 源码中的strip\_path函数只能将ppinode置为root\_node,因为源码中的文件系统是一个扁平文件系统。
3. 在fs/open.c中新增printallchildfiles(struct inode\* pin)函数

**函数作用：**

输出pin对应的文件夹下所有的内容（文件及文件夹）

1. 对fs/open.c中的do\_open函数进行修改

**函数作用：**

1. 增加操作，使得根目录能够被ls。如此一来，如果在/a/下使用ls，则会转化为open路径为/a的目录文件。
2. open完成后再返回fd之前调用函数printallchildfiles(pin)函数并进行说明。pin为要打开的目录/文件的inode指针。
3. 添加对filetype的判断，如果为创建，则根据filetype的不同调用createfile或者createdirectory函数并进行说明。
4. 在fs/open.c中新增createdirectory函数

**函数作用：**

实现文件夹的创建，创建其他分配位都与createfile函数相同，创建inode不同，该函数中，I\_mode为I\_DIRECTORY。

1. 将fs/link.c中的do\_unlink()函数替换为do\_unlink\_my()函数

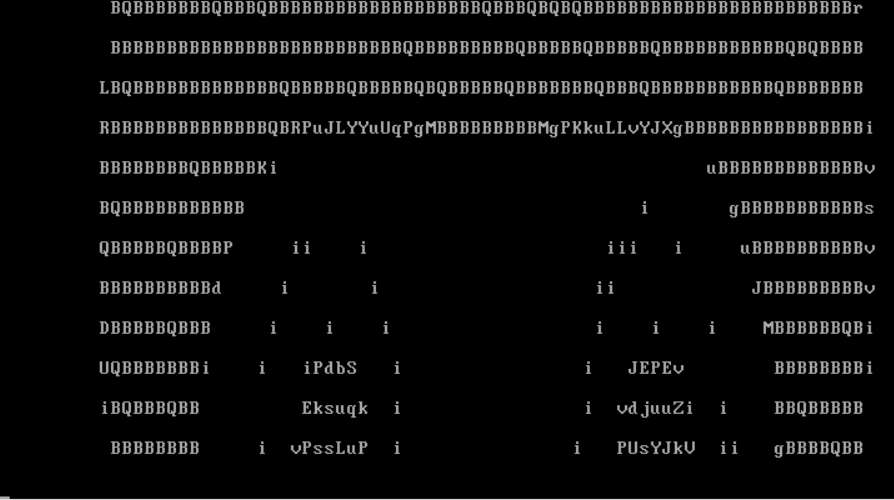
**函数作用：**

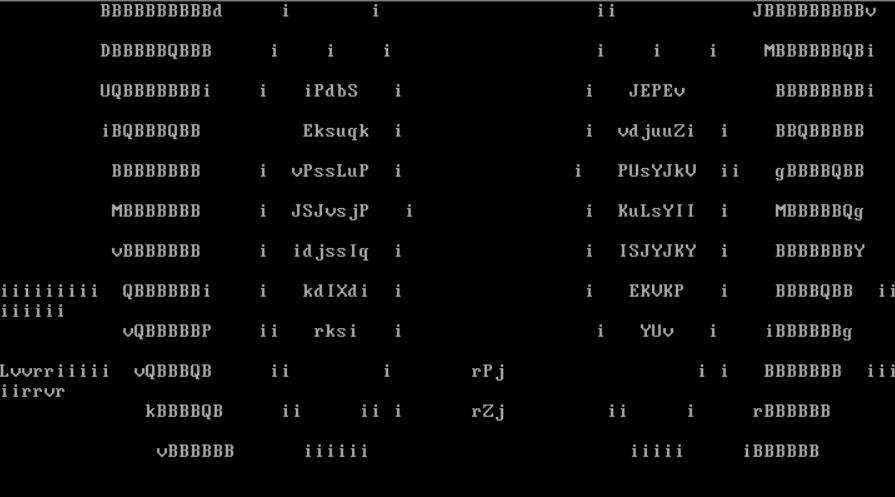
1. 实现删除文件以及递归删除文件夹。
2. 实现目录存在的情况下，I\_CNT记录打开文件数依旧有效。
3. **操作系统使用说明**

进入操作系统之后首先展示欢迎界面，屏幕上出现一个彩色的HI！！！。



之后，会逐行打印章鱼猫Octocat图案。





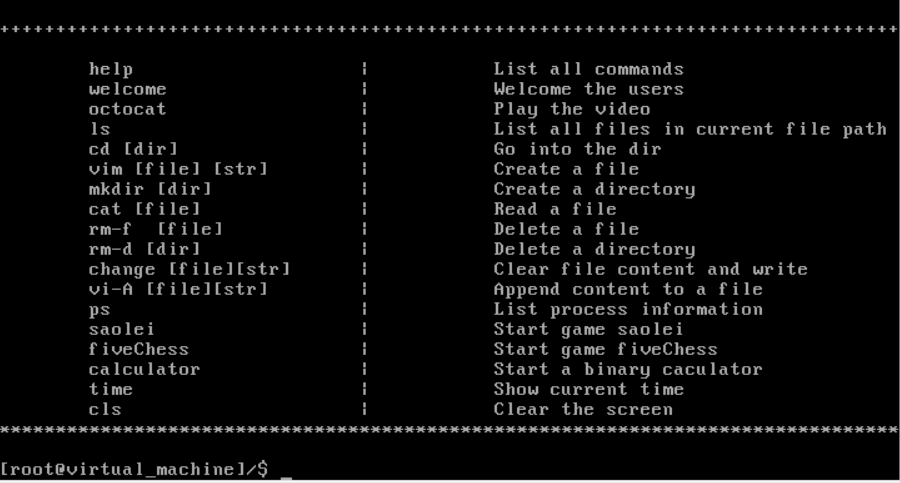
章鱼猫图案消失之后，进入操作系统的shell欢迎界面。该欢迎界面显示当前时间，同时用户按下任意按键可以开始使用操作系统。



以下进行各个指令的展示：

**指令：**help

**功能：**用户输入help之后，会显示所有的命令清单，主要包含：展示指令（展示HI欢迎界面、animation章鱼猫动画展示以及时间）、与文件操作相关的指令以及应用名称指令这三大部分。



**指令：**welcome

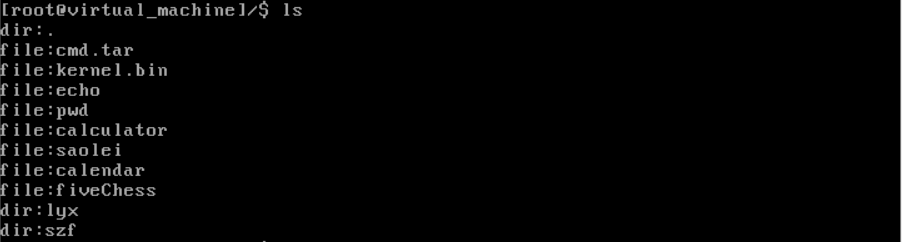
**功能：**展示刚进入操作系统的欢迎界面，显示彩色的“HI！！！”

**指令：**octocat

**功能：**展示章鱼猫动画

**指令：**ls

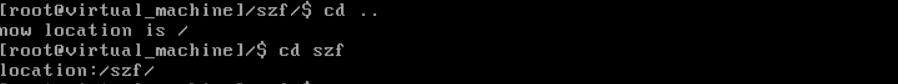
**功能：**显示当前目录下文件和目录



**指令：**cd [dir]

**功能：**进入当前目录下名为dir的文件夹。cd ..指令为返回上一级目录

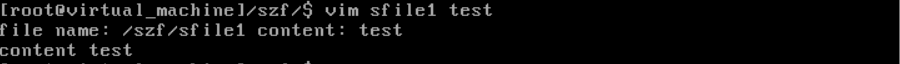
指令cd ..返回上一级目录以及指令cd szf 进入目录szf的运行结果如下：



**指令：**vim [file] [str]

**功能：**在当前目录下创建一个名为file的文件，并写入内容str

指令vim sfile1 test，创建一个名为sfile1的文件，文件内容为test。成功运行结果如下，显示文件内容为test：



**指令：**mkdir [dir]

**功能：**创建一个名为dir的目录（文件夹）

指令mkdir szf，创建一个名为szf的目录。运行结果如下：

mkdir-创建文件夹

**指令：**cat [file]

**功能：**在当前目录下查看名为file的文件内容

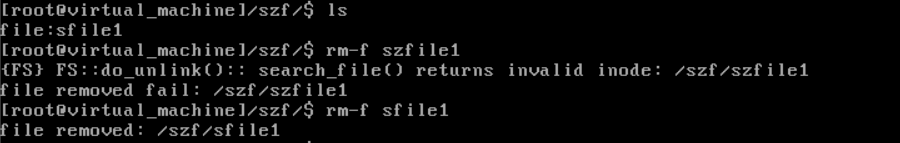
指令cat sfile1查看szf目录下名为sfile1的文件的内容。成功运行结果如下，显示文件中的内容test：

cat[filename]-查看文件内容

**指令：**rm-f [file]

**功能：**在当前目录下删除名为file的文件。若file不存在，则显示没有这个文件，若file存在，则删除成功之后显示相应的提示信息。

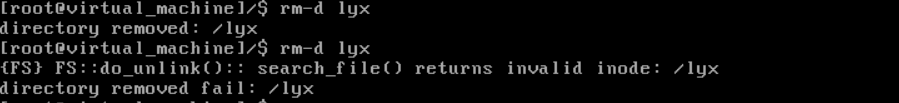
指令rm-f szfile1删除szf目录下，名为szfile1的文件以及指令rm-f sfile1删除szf目录下，名为sfile1的文件。运行结果如下，显示/szf/sfile1已经被删除：



**指令：**rm-d [dir]

**功能：**删除名为dir的目录（文件夹）。若dir不存在，则显示没有这个目录，若dir存在，则删除成功之后显示相应的提示信息。

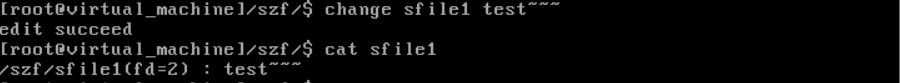
指令rm-d lyx删除名为lyx的目录以及再次执行该指令的运行结果如下：



**指令：**change [file][str]

**功能：**在当前目录下将名为file的文件中的内容全部替换为str中的内容。

指令change sfile1 test~~~将szf目录下名为sfile1的文件中的原内容test替换为test~~~；再执行cat sfile1指令，读取sfile1文件中现在的内容。运行结果如下：

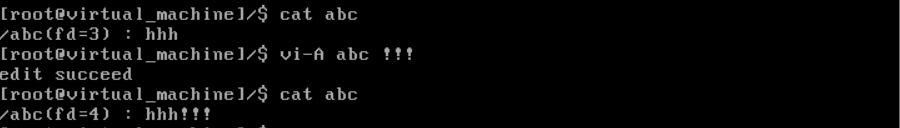


**指令：**vi-A [file][str]

**功能：**在当前目录下以追加模式在名为file的文件中写入str中的内容

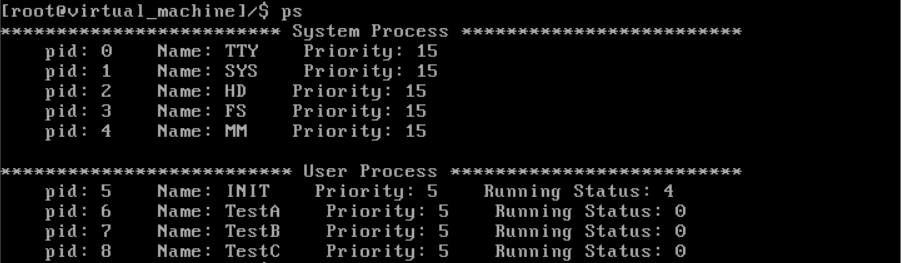
先执行指令cat abc 读取文件abc中的内容，为hhh；再执行指令vi-A abc

！！！，在文件abc后面增加“！！！”内容；最后执行指令cat abc，再次读取文件abc中的内容。运行结果如下：



**指令：**ps

**功能：**展示当前包括用户进程以及系统进程在内的所有进程



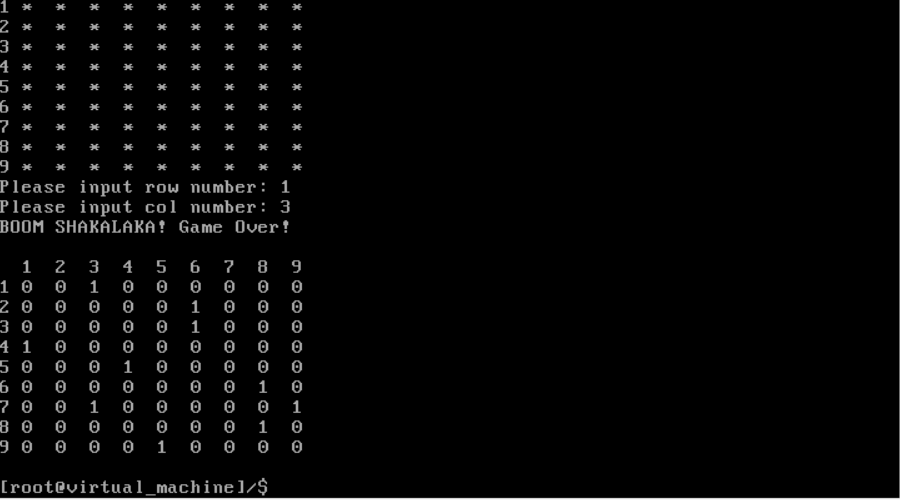
指令：saolei

功能：开始扫雷游戏，输入行与列进行排雷，当成功选择除地雷之外的所有位置之后，游戏成功。用户也可随时退出游戏。

游戏开始界面：



游戏结束界面：



**指令：**fivechess

**功能：**开始五子棋游戏，输入行列坐标进行下棋，为人机对弈模式，任意一方棋子5个相连时，该方获胜。

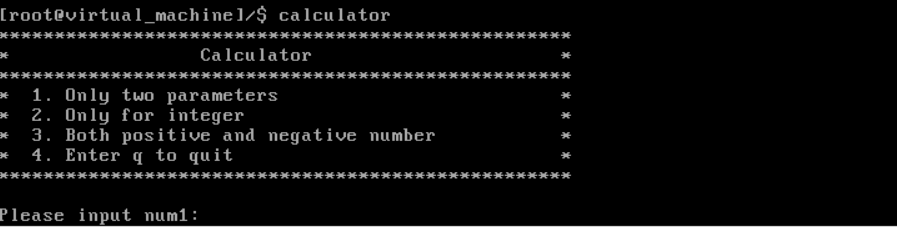




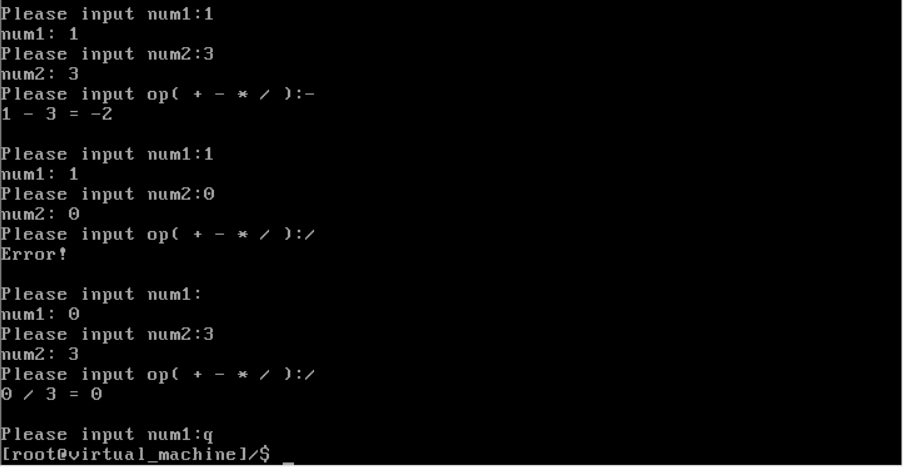
**指令：**calculator

**功能：**开启计算器，可以计算包括正数与负数在内的二元整数的四则运算

计算器开启界面：



计算器使用界面：



**指令：**time

**功能：**获取当前系统时间

time-获取当前系统时间

**指令：**cls

**功能：**清屏



最后展示修复的源码中的一个BUG，使得该操作系统可以支持多TTY

