



# Linux 开发环境及应用实验报告

## 实验四：Shell 管道和重定向功能的实现

付容天

学号 2020211616

班级 2020211310

计算机学院（国家示范性软件学院）

2023 年 6 月 2 日

# 1 实验目的与实验要求

本实验目的在于使学生掌握 Shell 管道功能与重定向功能的内容及其实现原理，并熟悉 Linux 系统中的各类命令，以及掌握 Linux 系统所提供的系统调用方法，能够通过 C 语言编程使用系统调用实现这一功能。

本实验要求使用 `fork()`、`exec()`、`dup2()`、`pipe()`、`open()`、`wait()` 等系统调用编写 C 语言程序完成与下列 shell 命令等价的功能：

```
grep -v usr < /etc/passwd | wc -l > r.txt; cat r.txt
```

## 2 任务分析与代码编写

上述命令实现的功能为：

- (1) `grep -v`：该命令结合输入重定向 `< /etc/passwd` 文件作为输入，通过 `-v` 选项筛选出不包含 `usr` 的行，并将它们输出到管道；
- (2) `wc -l`：将前一命令的输出结果作为输入进行行计数，并使用输出重定向 `> r.txt` 将结果输出到 `r.txt` 文件中，并以分号 “;” 结束该命令；
- (3) `cat`：将文件 `r.txt` 的内容打印到标准输出设备上。

为了在 C 程序中实现这一功能，我使用了下面的系统调用函数：

- (1) `fork()`：用于创建一个新进程，注意对管道机制的模拟；
- (2) `execlp()`：可以在 C 语言中模拟 linux 命令的执行；
- (3) `dup2()`：复制文件描述符；
- (4) `pipe()`：用于创建管道，以实现进程之间的通信；
- (5) `open()`：用来打开或创建一个文件；
- (6) `wait()`：阻塞并等待子进程退出，回收子进程残留资源；

我编写的代码如下所示：

```
#include <stdio.h>
#include <stdlib.h>
#include <fcntl.h>
#include <unistd.h>
#include <sys/wait.h>

int main() {
    int fd[2];
    pipe(fd);
    if(fork() == 0) {
```

```

        close(0);
        close(1);
        close(fd[0]);
        int passwd_fd = open("/etc/passwd", O_RDONLY);
        dup2(passwd_fd, 0);
        dup2(fd[1], 1);
        execlp("grep", "grep", "-v", "usr", NULL);
    }
    if(fork() == 0) {
        close(0);
        close(1);
        close(fd[1]);
        int r_fd = open("r.txt", O_WRONLY | O_CREAT, 0777);
        dup2(r_fd, 1);
        dup2(fd[0], 0);
        execlp("wc", "wc", "-l", NULL);
    }
    close(fd[0]);
    close(fd[1]);
    wait(NULL);
    wait(NULL);
    execlp("cat", "cat", "r.txt", NULL);
}

```

上述代码中，我使用 `pipe()` 对管道进行初始化，并在 `fork()` 创建的子进程中用 `dup2` 替换掉子进程的输入/输出，最后分别调用 `execlp()` 执行相应的命令，即可完成对要求的命令语句的模拟。

最后编译运行，并进行检测发现运行结果正确，如下图所示：

```

c1616@Ubuntu-bupt:~$ ./lab4
1178
c1616@Ubuntu-bupt:~$ grep -v usr < /etc/passwd | wc -l > r.txt; cat r.txt
1178
c1616@Ubuntu-bupt:~$

```

### 3 实验总结

在本次实验中，我通过分析和编写 `lab4.cpp` 代码文件，实现了对指定命令语句的模拟，学习了 `fork()` 机制、管道机制，还练习了 `dup2()` 和 `execlp()` 等函数的使用，并对系统调用有了一些认识，本次实验我收获满满。