

四川大学计算机学院、软件学院

实验报告

学号: 2022141460180 姓名: 封欢欢 专业: 计算机科学与技术 班级: 行政4班 第 6-7 周

课程名称	操作系统课程设计	实验课时	4 小时
实验项目	Linux 环境及 Shell 程序	实验时间	第 6 周到第 7 周
实验目的	1) 了解 Linux 编程环境和编程工具 2) 掌握基本的 linux 系统命令及执行过程 3) 了解 shell 的作用及主要分类 4) 掌握 shell 脚本程序运行原理及基础语法, 学会编写简单的 shell 脚本程序。		
实验环境	HUAWEI 电脑 WSL2 Ubuntu2204		
实验内容 (算法、程序、步骤和方法)	<p>实验一 文件与搜索:</p> <p>1. 以自己的姓名在/home 下新建目录, 通过 ls 命令验证目录创建成功</p> <pre> 密码: root@LAPTOP-GPNDGL26-wsl:/home# mkdir fenghuanhuan root@LAPTOP-GPNDGL26-wsl:/home# ls fenghuanhuan java root@LAPTOP-GPNDGL26-wsl:/home# ls -l 总用量 8 drwxr-xr-x 2 root root 4096 4月 11 18:11 fenghuanhuan drwxr-xr-x 9 java java 4096 4月 11 17:37 java root@LAPTOP-GPNDGL26-wsl:/home# </pre> <p>2. 切换到/fenghuanhuan 目录下, 新建文件 file。</p> <pre> root@LAPTOP-GPNDGL26-wsl:/home# cd ./fenghuanhuan root@LAPTOP-GPNDGL26-wsl:/home/fenghuanhuan# touch file root@LAPTOP-GPNDGL26-wsl:/home/fenghuanhuan# ls file root@LAPTOP-GPNDGL26-wsl:/home/fenghuanhuan# </pre> <p>3. 为 file 分别创建硬链接 fileHard 及符号链接 fileSoft。</p> <pre> root@LAPTOP-GPNDGL26-wsl:/home/fenghuanhuan# ln file filehard root@LAPTOP-GPNDGL26-wsl:/home/fenghuanhuan# ln -s file filessoft </pre> <p>4. 通过 ls -li 命令查看两者的区别, 可知硬连接文件 fileHard 是一般文件, 其节点与原文件 file 的 inode 节点相同, 而符号连接文件 fileSoft 是链接文档, 其节点等于 file 的节点加一。</p> <pre> root@LAPTOP-GPNDGL26-wsl:/home/fenghuanhuan# ls -li 总用量 0 1320 -rw-r--r-- 2 root root 0 4月 11 23:24 file 1320 -rw-r--r-- 2 root root 0 4月 11 23:24 filehard 1496 lrwxrwxrwx 1 root root 4 4月 11 23:25 filessoft -> file root@LAPTOP-GPNDGL26-wsl:/home/fenghuanhuan# </pre>		

5. 删除 file 文件, 硬连接 fileHard 不受影响, 但是符号连接 fileSoft 文件出错。

```
root@LAPTOP-GPNDGL26-wsl:/home/fenghuanhuan# rm file
root@LAPTOP-GPNDGL26-wsl:/home/fenghuanhuan# ls
filehard  filesoft
```

6. 查找/etc 目录中所有大于 100k 的文件清单。

```
root@LAPTOP-GPNDGL26-wsl:/home# find /etc -size +100k -type f
/etc/lvm/lvm.conf
/etc/ssl/certs/ca-certificates.crt
/etc/ssh/moduli
```

7. 将清单及行数都重定向输出到名为 out 的文件中, 通过 cat 命令查看 out 文件的内容。

```
root@LAPTOP-GPNDGL26-wsl:/home# find /etc -size +100k -type f >>/home/fenghuanhuan/out
root@LAPTOP-GPNDGL26-wsl:/home# find /etc -size +100k -type f | wc -l>>/home/fenghuanhuan/out
root@LAPTOP-GPNDGL26-wsl:/home# cat /home/fenghuanhuan/out
/etc/lvm/lvm.conf
/etc/ssl/certs/ca-certificates.crt
/etc/ssh/moduli
3
root@LAPTOP-GPNDGL26-wsl:/home#
```

实验 2. 用户与用户组

1. 建立两个用户组 grp1 和 grp2, 通过/etc/group 文件查看用户组信息。

```
root@LAPTOP-GPNDGL26-wsl:/home# groupadd grp1
root@LAPTOP-GPNDGL26-wsl:/home# groupadd grp2
root@LAPTOP-GPNDGL26-wsl:/home# tail -n 2 /etc/group
grp1:x:1001:
grp2:x:1002:
```

2. 每个组下分别有两个用户 u11,u12 以及 u21,u22, 通过/etc/passwd 文件查看用户信息。

```
root@LAPTOP-GPNDGL26-wsl:/home# useradd -g grp1 u11
root@LAPTOP-GPNDGL26-wsl:/home# useradd -g grp1 u12
root@LAPTOP-GPNDGL26-wsl:/home# useradd -g grp2 u21
root@LAPTOP-GPNDGL26-wsl:/home# useradd -g grp2 u22
root@LAPTOP-GPNDGL26-wsl:/home# tail -n 4 /etc/passwd
u11:x:1001:1001::/home/u11:/bin/sh
u12:x:1002:1001::/home/u12:/bin/sh
u21:x:1003:1002::/home/u21:/bin/sh
u22:x:1004:1002::/home/u22:/bin/sh
```

3. 删除用户 u21,u22, 删除用户组 grp2

```
root@LAPTOP-GPNDGL26-wsl:/home# userdel u21
root@LAPTOP-GPNDGL26-wsl:/home# userdel u22
root@LAPTOP-GPNDGL26-wsl:/home# groupdel grp2
```

验证删除成功

```
root@LAPTOP-GPNDGL26-wsl:/home# tail -n 2 /etc/passwd
u11:x:1001:1001::/home/u11:/bin/sh
u12:x:1002:1001::/home/u12:/bin/sh
root@LAPTOP-GPNDGL26-wsl:/home# tail -n 1 /etc/group
grp1:x:1001:
```

4. 新建文件夹 file.txt, 并把 file.txt 的所有者改为 u11, 拥有组改为 grp1. 使同组用户可读写和执行, 其他无任何权限。

```
root@LAPTOP-GPNDGL26-wsl:/home# touch file.txt
root@LAPTOP-GPNDGL26-wsl:/home# chown u11:grp1 file.txt
root@LAPTOP-GPNDGL26-wsl:/home# chmod 770 file.txt
```

5. 使用 `ls -l` 命令验证权限修改成功。

```
root@LAPTOP-GPNDGL26-wsl:/home# ls -l
总用量 8
drwxr-xr-x 2 root root 4096 4月 11 23:32 fenghuanhuan
-rwxrwx--- 1 u11 grp1 0 4月 12 00:11 file.txt
drwxr-x--- 9 java java 4096 4月 11 17:37 java
root@LAPTOP-GPNDGL26-wsl:/home#
```

实验 3. Shell 编程

1.1 编写 shell 脚本、添加执行权限、运行脚本，由运行结果可知程序正确。

```
root@LAPTOP-GPNDGL26-wsl:/home/fenghuanhuan# vi sum.sh
root@LAPTOP-GPNDGL26-wsl:/home/fenghuanhuan# chmod u+x sum.sh
root@LAPTOP-GPNDGL26-wsl:/home/fenghuanhuan# ./sum.sh
1+2+3+...+100=5050
```

1.2 该 shell 脚本内容如下。

```
#!/bin/bash
#1+2+3+...+100
sum=0
for((i=0;i<=100;i++))
do
    ((sum=sum+i))
done
echo "1+2+3+...+100=$sum"
```

2.1 切换到根目录，在根目录下创建 4 个文件

```
root@LAPTOP-GPNDGL26-wsl:/home/fenghuanhuan# cd /
root@LAPTOP-GPNDGL26-wsl:/# touch m1.txt m2.txt m3.txt m4.txt
root@LAPTOP-GPNDGL26-wsl:/# ls
bin  dev  home  lib  lib64  lost+found  m2.txt  m4.txt  mnt  proc  run  snap  sys  usr
boot  etc  init  lib32  libx32  m1.txt  m3.txt  media  opt  root  sbin  srv  tmp  var
root@LAPTOP-GPNDGL26-wsl:/#
```

2.2 编写 shell 脚本、添加执行权限、运行脚本。

```
root@LAPTOP-GPNDGL26-wsl:/# vi move.sh
root@LAPTOP-GPNDGL26-wsl:/# chmod u+x move.sh
root@LAPTOP-GPNDGL26-wsl:/# ./move.sh
root@LAPTOP-GPNDGL26-wsl:/# vi move.sh
```

2.3 该 shell 脚本内容如下，使用循环语句依次将文件移到相应目录下。

```
#!/bin/bash
#move m1.txt m2.txt m3.txt m4.txt to m1 m2 m3 m4
for((i=1;i<=4;i++))
do
    mkdir m${i}
    mv m${i}.txt m${i}
done
```

2.4 验证根目录下存在 m1、m2、m3、m4 目录，m1、m2、m3、m4 目

录下分别存在 m1.txt、m2.txt、m3.txt、m4.txt 文件

```
root@LAPTOP-GPNDGL26-wsl:/# ls
bin  dev  home  lib  lib64  lost+found  m2  m4  mnt  opt  root  sbin  srv  tmp  var
boot  etc  init  lib32  libx32  m1  m3  media  move.sh  proc  run  snap  sys  usr
root@LAPTOP-GPNDGL26-wsl:/# ls m1 m2 m3 m4
m1:
m1.txt
m2:
m2.txt
m3:
m3.txt
m4:
m4.txt
root@LAPTOP-GPNDGL26-wsl:/#
```

3.1 编写 shell 脚本、添加执行权限、运行脚本。

```
root@LAPTOP-GPNDGL26-wsl:/# vi mkdiruser.sh
root@LAPTOP-GPNDGL26-wsl:/# chmod u+x mkdiruser.sh
root@LAPTOP-GPNDGL26-wsl:/# ./mkdiruser.sh
```

3.2 该 shell 脚本内容如下,先创建/userdata 目录,切换到该目录,在该目录下建立 50 个目录 user1~user50,设置权限。

```
root@LAPTOP-GPNDGL26-ws  ×  +  v
#!/bin/bash
#mkdir user1-user50
mkdir /userdata
cd /userdata
for((i=1;i<=50;i++))
do
    mkdir user${i}
    chmod 754 user${i}
done
```

3.3 验证根目录下存在/uerdata 目录,/uerdata 目录存在 50 个目录 user1~user50,且权限符合要求。

	<pre>root@LAPTOP-GPNDGL26-wsl:/# ls -l /userdata 总用量 200 drwxr-xr-- 2 root root 4096 4月 12 12:17 user1 drwxr-xr-- 2 root root 4096 4月 12 12:17 user10 drwxr-xr-- 2 root root 4096 4月 12 12:17 user11 drwxr-xr-- 2 root root 4096 4月 12 12:17 user12 drwxr-xr-- 2 root root 4096 4月 12 12:17 user13 drwxr-xr-- 2 root root 4096 4月 12 12:17 user14 drwxr-xr-- 2 root root 4096 4月 12 12:17 user15 drwxr-xr-- 2 root root 4096 4月 12 12:17 user16 drwxr-xr-- 2 root root 4096 4月 12 12:17 user17 drwxr-xr-- 2 root root 4096 4月 12 12:17 user18 drwxr-xr-- 2 root root 4096 4月 12 12:17 user19 drwxr-xr-- 2 root root 4096 4月 12 12:17 user2 drwxr-xr-- 2 root root 4096 4月 12 12:17 user20 drwxr-xr-- 2 root root 4096 4月 12 12:17 user21 drwxr-xr-- 2 root root 4096 4月 12 12:17 user22 drwxr-xr-- 2 root root 4096 4月 12 12:17 user23 drwxr-xr-- 2 root root 4096 4月 12 12:17 user24 drwxr-xr-- 2 root root 4096 4月 12 12:17 user25 drwxr-xr-- 2 root root 4096 4月 12 12:17 user26 drwxr-xr-- 2 root root 4096 4月 12 12:17 user27 drwxr-xr-- 2 root root 4096 4月 12 12:17 user28 drwxr-xr-- 2 root root 4096 4月 12 12:17 user29 drwxr-xr-- 2 root root 4096 4月 12 12:17 user3 drwxr-xr-- 2 root root 4096 4月 12 12:17 user30 drwxr-xr-- 2 root root 4096 4月 12 12:17 user31 drwxr-xr-- 2 root root 4096 4月 12 12:17 user32 drwxr-xr-- 2 root root 4096 4月 12 12:17 user33</pre>
<p>(接上)</p> <p>实验内容（算法、程序、步骤和方法）</p>	
<p>数据记录 和计算</p>	<p>无数据记录</p>

<p>结 论 (结 果)</p>	<p>实验 1 实现了文件的创建、硬链接和符号链接，通过 <code>ls-li</code> 命令和删除源文件了解到两者的区别。硬链接使得多个文件名指向同一个文件，它们是“平等”的，删除其中任何一个都不会影响其他访问；符号链接文件是链接文件，指向源文件，属于“主从”关系，如果源文件被删除，符号链接文件仍然存在，但指向的是一个无效的链接。其次实现了用 <code>find</code> 命令进行文件搜索，用 <code>wc-l</code> 命令统计行数，用重定向输出命令将目标内容保存到文件中，通过管道操作将前一条命令的执行结果作为输入数据传送给后一命令，以实现复杂的操作。</p> <p>实验 2 实现了创建和删除用户组、用户，在创建用户时指定其所属用户组，并且通过 <code>/etc/group</code> <code>/etc/passwd</code> 两个文件查看用户组和用户信息，以及为使用 <code>chown</code> 命令改变文件所有者和拥有组，使用 <code>chmod</code> 命令改变属主、属组、其他用户为文件的读写执行权限。</p> <p>实验 3 shell 编程通过使用变量和循环语句等，实现了数据计算和批量处理文件与目录，数据计算 shell 编程与其他编程语言差不多，只是语法上有一些差别，但文件与目录管理 shell 编程更加容易实现，简化了文件与目录的重复操作。</p>
<p>小 结</p>	<p>通过此次实验，我掌握了基本 linux 系统命令的使用以及它们的作用，如 <code>mkdir</code>、<code>touch</code>、<code>groupadd</code>、<code>useradd</code>、<code>ls</code>、<code>cat</code>、<code>chown</code>、<code>chmod</code> 命令等，了解到硬链接及符号链接的区别。学会了 shell 编程的操作流程：编写 shell 脚本、添加执行权限、运行脚本，掌握了 shell 脚本的基本语法，能编写简单的 shell 脚本程序。在学习 shell 编程过程中发现运算操作、循环语言的使用语法多样，还需要不断学习总结，掌握更多语法，以便能实现复杂的 shell 编程。</p>
<p>指导老师评议</p>	<p>成绩评定：_____ 指导教师签名：_____</p>