Linux基础实验

目 录

[1. 环境搭建 3](#_Toc396403203)

[a. VMWare WorkStation的安装 3](#_Toc396403204)

[b. Linux虚拟机的安装 3](#_Toc396403205)

[2. Linux命令 3](#_Toc396403206)

[2.1 man命令 4](#_Toc396403207)

[2.2 ls命令 4](#_Toc396403208)

[2.3 pwd命令 5](#_Toc396403209)

[2.4 cd命令 6](#_Toc396403210)

[2.5 echo命令 7](#_Toc396403211)

[2.6 more命令 7](#_Toc396403212)

[2.7 mv命令 8](#_Toc396403213)

[2.8 cp命令 9](#_Toc396403214)

[2.9 chmod命令 10](#_Toc396403215)

[2.10 grep命令 12](#_Toc396403216)

[2.11 mkdir/rmdir命令 13](#_Toc396403217)

[2.12 rm命令 14](#_Toc396403218)

[2.13 find命令 15](#_Toc396403219)

[2.14 df命令 16](#_Toc396403220)

[2.15 gzip/gunzip命令 16](#_Toc396403221)

[2.16 ps/kill命令 17](#_Toc396403222)

[2.17 Linux命令综合训练 18](#_Toc396403223)

[3. Linux系统的工具 18](#_Toc396403224)

[3.1 vi编辑器 19](#_Toc396403225)

[3.2 数据流定向 20](#_Toc396403226)

[3.3 管道 21](#_Toc396403227)

[3.4 Linux系统工具综合训练 21](#_Toc396403228)

[4. 课后作业 23](#_Toc396403229)

Linux基础的实验主要是在Linux环境下进行熟悉命令以及相关工具的学习和使用，并能够编写简单的shell脚本，所以该实验的主要内容包括VMware的安装、Linux的安装和打开、Linux的基本命令学习和操作、Linux虚拟机与宿主机之间的文件传输、Linux系统中vi工具的应用、Linux系统下数据流定向和管道的应用、Linux系统下简单说shell脚本的编写以及Linux系统下yum相关文件的查阅等内容。

# 环境搭建（演练）

Linux基础的实验所以依赖的环境是Linux虚拟机，而Linux虚拟机则是运行在VMware WorkStation中的虚拟操作系统，所以Linux基础的环境搭建主要包括VMWare WorkStation的安装以及Linux虚拟机环境的搭建。

## VMWare WorkStation的安装

VMWare公司是全球桌面到数据中心虚拟化解决方案的领导厂商，它提供了一系列的VMWare软件来为全球不同规模的用户提供相应的服务来降低他们的生产成本，确保他们的业务连续性和生产安全性。

其中，VMWare WorkStation是该公司的商业软件产品之一。该工作站软件包含一个用于英特尔x86相容电脑的虚拟机套装，其允许用户同时创建和运行多个x86虚拟机。每个虚拟机实例可以运行其自己的客户机操作系统，如（但不限于）Windows、Linux、BSD变生版本。用简单术语来描述就是，VMware工作站允许一台真实的电脑在一个操作系统中同时开启并运行数个操作系统。其它VMware产品帮助在多个宿主电脑之间管理或移植VMware虚拟机。

将工作站和服务器转移到虚拟机环境，可使系统管理简单化、缩减实际的底板面积、并减少对硬件的需求。

VMWare Workstation现有不同的版本，针对当前的Windows XP系统、Win7系统和Win8系统都有相应版本的支持。

所以在安装Linux虚拟机之前，需要安装适合当前系统的VMWare Workstation，安装过程主要参考以下步骤：

<http://jingyan.baidu.com/article/86f4a73e3ae31337d65269ee.html>

## Linux虚拟机的安装

由于教材提供的并非是Linux系统的iso的安装介质，而是已经安装好Linux虚拟机，它是以文件夹的形式存储的，可以通过VMware直接打开。Linux虚拟机的打开过程参考以下步骤：

1. 双击桌面上的WMware WorkStation图标
2. WMware Workstation启动后，进入主页界面
3. 点击“打开虚拟机”
4. 找到虚拟机的解压路径
5. 选择其中的.vmx文件，即可完成虚拟机导入
6. 点击启动客户机，即可启动虚拟机

# Linux命令

Linux系统提供了命令行的人机交互接口，所以学习和熟练操作Linux命令是学习和操作Linux系统的第一步。

注意：加黑的带有“#”的为命令输入。例如“**# man man”**

## man命令

【实验目的】熟练掌握Linux系统中的man命令的用法

【实验结果】显示查询命令的用法信息

man命令是获取关于命令的帮助信息，会显示先关命令的用法。

命令格式为：

* **man + command**

man输出的command使用信息会默认一次显示一页，可以按SPACE(空格键)进行向下翻页，按b键向上翻页，按q键退出。

1）查看man命令的用法

**# man man**

输出的结果是man命令的先关用法，仔细阅读该命令的用法信息，直至当前显示结尾。

2）按下空格键，进行翻页显示

显示的man命令用法的信息向后翻页，仔细阅读当前显示的信息

3）按下b键

显示的man命令用法的信息向前翻页

4）继续按空格键，阅读完man命令用法信息，知道出现“(END)”字符

5）按下q键，结束man命令用法信息显示

6）查看ls命令的用法

**# man ls**

输出的结果是ls命令的先关用法，仔细阅读该命令的用法信息，直至当前显示结尾。

7）按下空格键，进行翻页显示

显示的ls命令用法的信息向后翻页，仔细阅读当前显示的信息

8）按下b键

显示的ls命令用法的信息向前翻页

9）按下q键，结束ls命令用法信息显示

## ls命令

【实验目的】熟练掌握Linux系统中的ls命令的用法

【实验结果】显示相应目录的信息

ls命令是获取目标参数目录的信息的命令，添加不同的选项可以按照不同的格式进行显示。命令格式为：

* **ls + [选项] + command**
* **不带任何选项，是对目录内容的普通显示**
* **选项-a：列出目录内的所有文件（包含隐含文件）**
* **选项–l：列出文件的详细信息（包括文件属性，所属用户，文件大小，文件创建时间，文件名等信息）**

1）查看ls命令的用法

**# man ls**

输出的结果是ls命令的先关用法，仔细阅读该命令的用法信息，直至当前显示结尾。

2）按下空格键，进行翻页显示，直至“[END]”字符

显示的ls命令用法的信息向后翻页，仔细阅读当前显示的信息

3）按下q键，结束ls命令用法信息显示

4）查看当前目录的可见文件

**# cd /root**

**# ls**

**# ls .**

输出的结果是当前目录的具体信息，输出结果是以普通的形式显示当前目录下可见文件

5）查看当前目录下所有文件

**# ls –a**

输出的结果是当前目录的具体信息，输出结果是以普通的形式显示当前目录下所有文件

6）查看当前目录下可见文件的具体信息

**# ls –l**

输出的结果是当前目录的具体信息，输出结果是显示当前目录下可见文件的详细信息，包括文件属性，所属用户，文件大小，文件创建时间，文件名称等主要内容。

7）查看目标目录的可见文件

**# ls /etc/rc.d**

输出的结果是/etc/rc.d目录的具体信息，输出结果是以普通的形式显示当前目录下可见文件

**# ls ..**

输出的结果是..目录(当前目录的上一级目录)的具体信息，输出结果是以普通的形式显示当前目录下可见文件

8）查看目标目录下所有文件

**# ls –a /etc/rc.d**

输出的结果是/etc/rc.d目录的具体信息，输出结果是以普通的形式显示当前目录下所有文件

**# ls –a ..**

输出的结果是..目录(当前目录的上一级目录)的具体信息，输出结果是以普通的形式显示当前目录下所有文件

**# ls –a ../ect (此命令在root目录下运行)**

输出的结果是/etc目录的具体信息，输出结果是以普通的形式显示当前目录下所有文件

9）查看当前目录下可见文件的具体信息

**# ls –l /etc**

输出的结果是/etc目录下的具体信息，输出结果是显示当前目录下可见文件的详细信息，包括文件属性，所属用户，文件大小，文件创建时间，文件名称等主要内容。

**# ls –l ..**

输出的结果是..目录(当前目录的上一级目录)的具体信息，输出结果是显示当前目录下可见文件的详细信息，包括文件属性，所属用户，文件大小，文件创建时间，文件名称等主要内容。

**# ls –l ../ect (此命令在root目录下运行)**

输出的结果是/etc目录下的具体信息，输出结果是显示当前目录下可见文件的详细信息，包括文件属性，所属用户，文件大小，文件创建时间，文件名称等主要内容。

## pwd命令

【实验目的】熟练掌握Linux系统中的pwd命令的用法

【实验结果】查看当前所在目录

pwd命令是查看用户当前所在的目录信息

* **pwd + [选项]**

1）查看pwd命令的用法

**# man pwd**

输出的结果是pwd命令的先关用法，仔细阅读该命令的用法信息，直至当前显示结尾。

2）按下空格键，进行翻页显示，直至“[END]”字符

显示的pwd命令用法的信息向后翻页，仔细阅读当前显示的信息

3）按下q键，结束ls命令用法信息显示

4）查看当前所在的目录

**# pwd**

输出的结果是用户当前所在的工作目录，输出结果是以绝对路径的形式显示出来。

## cd命令

【实验目的】熟练掌握Linux系统中的cd命令的用法

【实验结果】切换操作目录

cd命令是用来切换操作目录

* **cd + [目标目录]**

1）查看cd命令的用法

**# man cd**

输出的结果是cd命令的先关用法，仔细阅读该命令的用法信息，直至当前显示结尾。

2）按下空格键，进行翻页显示，直至“[END]”字符

显示的cd命令用法的信息向后翻页，仔细阅读当前显示的信息

3）按下q键，结束cd命令用法信息显示

4）使用cd进行目录切换

**# cd /etc/rc.d**

**# pwd**

输出结果是/etc/rc.d，说明已经成功切换到该目录

**# cd**

**# pwd**

输出结果应该是当前用户的登陆目录，如果是root用户，则是/root，如果是其他用户，如用户user1，则结果为/home/user1

**# cd /etc/rc.d**

**# cd ..**

**# pwd**

输出结果是/etc，说明..目录对于/etc/rc.d来说是上一级目录

**# cd .**

**# pwd**

输出结果是/etc，说明.目录对于/etc来说是当前目录

**# cd ./X11**

**# pwd**

输出结果是/etc/X11，说明./X11目录对于/etc来说是当前目录中的子目录，.目录则是相对于/etc来说的当前目录

**# cd ../rc.d**

**# pwd**

输出结果是/etc/rc.d，说明..是对于X11来说的上一级目录，rc.d为上一级目录中的子目录，../rc.d相对于X11来说，是上一级目录的子目录

## echo命令

【实验目的】熟练掌握Linux系统中的echo命令的用法

【实验结果】输出相应的字符串或变量值

echo命令是用来显示字符或变量值

* **echo + [目标目录]**
* **不带任何选项时默认换行输出；当需要输出变量时，可以在前面添加“$”符号**
* **选项-n：输出的字符之后不换行**

1）查看echo命令的用法

**# man echo**

输出的结果是echo命令的先关用法，仔细阅读该命令的用法信息，直至当前显示结尾。

2）按下空格键，进行翻页显示，直至“[END]”字符

显示的echo命令用法的信息向后翻页，仔细阅读当前显示的信息

3）按下q键，结束echo命令用法信息显示

4）使用echo进行字符串显示

**# echo “Hello world”**

Hello world

**# echo ‘Hello world’**

Hello world

**# echo hello world**

Hello world

**# echo -n hello world**

Hello world **# …**

5）使用echo进行变量值显示

**# str1=“Hello world”**

**# str2=‘Hello world’**

**# str3=Hello world**

**# echo $str1**

Hello world

**# echo $str2**

Hello world

**# echo -n $str3**

Hello world**# …**

## more命令

【实验目的】熟练掌握Linux系统中的more命令的用法

【实验结果】显示文件内容

more命令可以用来显示文件内容

* **more + [选项] + 文件名**
* **more命令是将文件内容逐页在屏幕上显示，通过按SPACE（空格键）往下翻页，按b键往上翻页，按q键结束显示**
* **不带参数是对文件的正常显示**
* **选项+n：从第n行开始显示文件**
* **选项 +/pattern：从文件中搜索符合pattern的字符串，并从该行的前两行开始显示**

1）查看more命令的用法

**# man more**

输出的结果是more命令的先关用法，仔细阅读该命令的用法信息，直至当前显示结尾。

2）按下空格键，进行翻页显示，直至“[END]”字符

显示的more命令用法的信息向后翻页，仔细阅读当前显示的信息

3）按下q键，结束more命令用法信息显示

4）通过数据流定向创建一个含有较多内容的文件

**# cd root** （切换到/root目录下）

**# man more > more\_help.file**（通过数据流定向创建more\_help.file）

**# more more\_help.file**

按下空格键向下翻页，查看相应内容；按下b键向上翻页；按q键结束显示

**# more +14 more\_help.file**

分析与**more more\_help.file**命令有什么不同，该命令有什么好处

**# more +/verison more\_help.file**

分析与**more more\_help.file**命令有什么不同，该命令有什么好处

## mv命令

【实验目的】熟练掌握Linux系统中的mv命令的用法

【实验结果】进行文件、目录移动或重命名

mv命令可以用来进行文件、目录移动或者重命名

* **mv + [选项]+ 源文件或者目录 + 目标文件或者目录**
* **如果目标参数不存在，那么就是对源参数进行重命名**
* **如果目标参数是已经的文件或者目录，那么就是将源文件或者目录进行覆盖或者移动，此时源文件消失**

1）查看mv命令的用法

**# man mv**

输出的结果是mv命令的相关用法，仔细阅读该命令的用法信息，直至当前显示结尾。

2）按下空格键，进行翻页显示，直至“[END]”字符

显示的mv命令用法的信息向后翻页，仔细阅读当前显示的信息

3）按下q键，结束mv命令用法信息显示

4）使用mv对文件或文件夹进行重命名，此时目标参数是不存在的

**# cd root** （切换到/root目录下）

**# man mv > more\_help.file**（通过数据流定向，将mv的帮助用法重新写入到more\_help.file）

**# ls**

more\_help.file

**# mv more\_help.file mv\_help.file**

**mv\_help.file**

**# mkdir folder1** （在当前目录下创建folder1文件夹）

**# ls**

folder1 mv\_help.file

**# mv folder1 folder2**

**# ls**

folder2 mv\_help.file

4）使用mv对文件或文件夹进行文件覆盖或移动，此时目标参数是存在的

**# pwd**

确保处在/root目录下，如果不是在/root目下，通过cd命令切换

**# man mv > mv\_help.file2**（通过数据流定向，将mv的帮助用法写入到more\_help.file2）

**# mkdir folder1**

**# ls**

查看当前目录下有以下内容（mv\_help.file mv\_help.file2 foler1 folder2，其中前两个文件为mv命令的帮助用法文件，如果缺失，可通过man命令和数据流定向生成该文件，后两个为文件夹，如果缺失，可通过mkdir命令创建）

**# mv mv\_help.file mv\_help.file2**

mv:是否覆盖“mv\_help.file2”?**y** （此处y为用户输入，此命令为覆盖命令，mv将mv\_help.file的内容覆盖到mv\_help.file2文件中，mv\_help.file文件消失）

**# ls**

folder1 folder2 mv\_help.file2

**# mv folder1 folder2**（此命令为文件夹移动命令，是将folder1移动到folder2中）

**# ls**

folder2 mv\_help.file2

**# mv mv\_help.file2 folder2** （此命令为文件移动命令，是将mv\_help.file2移动到folder2中）

**# ls**

folder2

## cp命令

【实验目的】熟练掌握Linux系统中的cp命令的用法

【实验结果】进行文件、目录拷贝

cp命令可以用来进行文件、目录的拷贝

* **cp + [选项]+ 源文件或者目录 +目标文件或者目录**
* **源参数是文件**
* **如果目标参数不存在或存在且为文件夹，就是对源参数进行拷贝**
* **如果目标参数存在且为文件，则是对目标参数的覆盖，此时源文件不丢失**
* **源参数是文件夹**
* **对将该文件夹复制到目标参数中。（目标参数为文件时则不能复制）**
* **选项-R：对源参数为带有文件或者多级文件夹时，需要添加该参数进行拷贝**

1）查看cp命令的用法

**# man cp**

输出的结果是cp命令的先关用法，仔细阅读该命令的用法信息，直至当前显示结尾。

2）按下空格键，进行翻页显示，直至“[END]”字符

显示的cp命令用法的信息向后翻页，仔细阅读当前显示的信息

3）按下q键，结束cp命令用法信息显示

4）当cp的源参数为文件时的拷贝

**# cd root** （切换到/root目录下）

**# man cp > cp\_help.file**（通过数据流定向，将mv的帮助用法重新写入到cp\_help.file）

**# man cp > cp\_help.file1**（通过数据流定向，将mv的帮助用法重新写入到cp\_help.file1）

**# mkdir folder1**

**# ls**

查看当前目录下有以下内容（cp\_help.file cp\_help.file1 foler1 folder2，其中前两个文件为cp

命令的帮助用法文件，如果缺失，可通过man命令和数据流定向生成该文件，后两个为文件夹，如果缺失，可通过mkdir命令创建）

**# cp cp\_help.file cp\_help.file2**

**# ls**

cp\_help.file cp\_help.file1 cp\_help.file2 folder1 folder2

**# cp cp\_help.file1 folder1**

**# ls**

cp\_help.file cp\_help.file1 cp\_help.file2 folder1 folder2

**# ls folder1**

… cp\_help.file1 …

5）当cp的源参数为文件时的覆盖

**# ls**

cp\_help.file cp\_help.file1 cp\_help.file2 folder1 folder2

**# cp cp\_help.file1 cp\_help.file2**

cp：是否覆盖”cp\_help.file2”？**y**

**# ls**

cp\_help.file cp\_help.file1 cp\_help.file2 folder1 folder2

6）当cp的源参数为文件夹时的拷贝

**# ls**

cp\_help.file cp\_help.file1 cp\_help.file2 folder1 folder2

**# cp –R folder2 folder1**

**# ls**

cp\_help.file cp\_help.file1 cp\_help.file2 folder1 folder2

**# ls folder1**

… cp\_help.file1 folder2…

## chmod命令

【实验目的】熟练掌握Linux系统中的chmod命令的用法

【实验结果】修改文件或者目录权限

chmod命令可以用来修改文件或者目录权限

* **chmod + [-R] [Who] [opt] [mod] + 文件或者目录**
* **选项-R：对文件夹进行递归更改，即连同子目录下的所有文件、目录进行更新成当前Who选项代表的用户**
* **选项opt：表示对文件的权限进行更改，“+”就是添加某个权限，“-” 表示取消某个权限，“=”表示取消原先所有权限，赋予当前权限**
* **选项mod：表示文件的权限，可为“r”、“w”、“x”.**

1）查看chmod命令的用法

**# man chmod**

输出的结果是chmod命令的先关用法，仔细阅读该命令的用法信息，直至当前显示结尾。

2）按下空格键，进行翻页显示，直至“[END]”字符

显示的chmod命令用法的信息向后翻页，仔细阅读当前显示的信息

3）按下q键，结束chmod命令用法信息显示

4）使用chmod添加权限

**# ls**

cp\_help.file cp\_help.file1 cp\_help.file2 folder1 folder2

**# rm –rf \*** （强制删除当前文件夹下的所有内容）

**# man chmod > chmod\_help.file**

**# ls –l**

-rw-r--r-- 1 root root 5804 7月28 16:28 chmod\_help.file

**# chmod +x chmod\_help.file**

**# ls –l**

-rwxr-xr-x 1 root root 5804 7月28 16:28 chmod\_help.file

5）使用chmod删除权限

**# ls –l**

-rwxr-xr-x 1 root root 5804 7月28 16:28 chmod\_help.file

**# chmod -x chmod\_help.file**

**# ls –l**

-rw-r--r-- 1 root root 5804 7月28 16:28 chmod\_help.file

**# chmod -r chmod\_help.file**

**# ls –l**

--w------- 1 root root 5804 7月28 16:28 chmod\_help.file

6）使用chmod修改权限

**# ls –l**

--w------- 1 root root 5804 7月28 16:28 chmod\_help.file

**# chmod =x chmod\_help.file**

**# ls –l**

---x--x--x 1 root root 5804 7月28 16:28 chmod\_help.file

**# chmod =r chmod\_help.file**

**# ls –l**

-r--r--r-- 1 root root 5804 7月28 16:28 chmod\_help.file

7）使用chmod的二进制方法修改权限

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 文件属性 | 文件类型 | 读 | 写 | 执行 | 读 | 写 | 执行 | 读 | 写 | 执行 |
| 表示字符 | -/d | r | w | x | r | w | x | r | w | x |
| 权限所有 | 文件/目录 | 文件所有者的权限 | | | 文件所有者同组权限 | | | 其他用户组的权限 | | |

1. 文件属性到二进制的映射:

- rwx rw- r-- : 111 110 100

- rw- r-x --x : 110 101 001

1. 不同进制向十进制的转换:

十进制转换成十进制：123 = 1\*10^2 + 2\*10^1 + 3\*10^0

二进制转换成十进制：0b111= 1\*2^2+1\*2^1+1\*2^0

1. 文件属性向十进制转换

- rwx rw- r-- : 111 110 100 = 1\*2^2+1\*2^1+1\*2^0 1\*2^2+1\*2^1+1\*2^0 1\*2^2+1\*2^1+1\*2^0 = 7 6 4

- rw- r-x --x : 110 101 001= 1\*2^2+1\*2^1+0\*2^0 1\*2^2+0\*2^1+1\*2^0 0\*2^2+0\*2^1+1\*2^0 = 6 5 1

**# ls –l**

-r--r--r-- 1 root root 5804 7月28 16:28 chmod\_help.file

- r-- r-- r-- : 100 100 100= 1\*2^2+0\*2^1+0\*2^0 1\*2^2+0\*2^1+0\*2^0 1\*2^2+0\*2^1+0\*2^0 =4 4 4

欲改变成的权限=> -rwx r-- r--

-rwx r-- r--:111 100 100= 1\*2^2+1\*2^1+1\*2^0 1\*2^2+0\*2^1+0\*2^0 1\*2^2+0\*2^1+0\*2^0 =7 4 4

**# chmod 744 chmod\_help.file**

**# ls –l**

查看文件权限是否修改成功

欲改变成的权限=> -rwx rwx r-x

-rwx rwx r-x:111 111 101= 1\*2^2+1\*2^1+1\*2^0 1\*2^2+1\*2^1+1\*2^0 1\*2^2+0\*2^1+1\*2^0 =7 7 5

**# chmod 775 chmod\_help.file**

**# ls –l**

查看文件权限是否修改成功

## grep命令

【实验目的】熟练掌握Linux系统中的grep命令的用法

【实验结果】根据既定条件搜索相应内容

grep命令可以在既定的条件下搜索相应内容

* **grep + [选项] + 搜索条件 + 目标文件**
* **不带任何选项是对搜索结果所在行整行输出**
* **选项-c：计算出符合搜索条件的次数**
* **选项-n：同时添加上符合搜索添加行的行号**

1）查看grep命令的用法

**# man grep**

输出的结果是grep命令的先关用法，仔细阅读该命令的用法信息，直至当前显示结尾。

2）按下空格键，进行翻页显示，直至“[END]”字符

显示的grep命令用法的信息向后翻页，仔细阅读当前显示的信息

3）按下q键，结束grep命令用法信息显示

4）使用grep命令进行搜索

**# man grep > grep\_help.file**

**# grep print grep\_help.file**

**# grep –c print grep\_help.file**

查看输出结果，分析与上一条命令的不同

**# grep –n print grep\_help.file**

查看输出结果，分析与上一条命令的不同

## mkdir/rmdir命令

【实验目的】熟练掌握Linux系统中的mkdir/rmdir命令的用法

【实验结果】创建目录/删除目录

mkdir命令可以创建目录

rmdir命令可以删除目录

* **mkdir + [选项] + 目录**
* **不带任何选项是对单个目录进行操作**
* **选项-p：可以创建多层目录**
* **rmdir + [选项] + 目录**
* **不带任何选项是对单个目录进行操作**
* **选项-p：可以删除多层目录，如果在操作后上层目录为空，那么将一起删除上层目录**

1）查看mkdir命令的用法

**# man mkdir**

输出的结果是mkdir命令的先关用法，仔细阅读该命令的用法信息，直至当前显示结尾。

2）按下空格键，进行翻页显示，直至“[END]”字符

显示的mkdir命令用法的信息向后翻页，仔细阅读当前显示的信息

3）按下q键，结束mkdir命令用法信息显示

4）查看rmdir命令的用法

**# man rmdir**

输出的结果是rmdir命令的相关用法，仔细阅读该命令的用法信息，直至当前显示结尾。

5）按下空格键，进行翻页显示，直至“[END]”字符

显示的rmdir命令用法的信息向后翻页，仔细阅读当前显示的信息

6）按下q键，结束rmdir命令用法信息显示

**7）使用mkdir创建目录**

**# cd /root**

**# rm –rf \***

**# man mkdir > file1**

**#mkdir folder1**

**# ls**

file1 folder1

**# mkdir folder2**

**# ls**

file1 folder1 folder2

**# mkdir –p foler3/subfolder1/sub2folder1** （当目录不存在时，创建递归目录需要加-p选项）

**# ls**

file1 folder1 folder2 folder3

**# ls folder3**

subfolder1

**# mkdir folder3/subfolder2** （当目录存在时，创建子目录，可直接创建）

**# ls folder3**

subfolder1 subfolder2

**8）使用rmdir创建目录**

**# rmdir folder2**

**# ls**

file1 folder1 folder3

**# rmdir folder3**

rmdir: ‘folder3’ :目录非空

**# rmdir –p foler3/subfolder1/sub2folder1**

rmdir: ‘folder3’ :目录非空

**# ls folder3**

subfolder2

**# rmdir –p folder3/subfolder2**

**# ls folder3**

file1 folder1

## rm命令

【实验目的】熟练掌握Linux系统中的rm命令的用法

【实验结果】删除目录或文件

rm命令可以用来删除目录或者文件

* **rm + [选项] + 文件或者目录**
* **选项-f：强制删除，不给出任何提示**
* **选项-i：删除文件时提供交互信息**
* **选项-r：对列出的目录进行递归删除**

1）查看rm命令的用法

**# man rm**

输出的结果是rm命令的先关用法，仔细阅读该命令的用法信息，直至当前显示结尾。

2）按下空格键，进行翻页显示，直至“[END]”字符

显示的rm命令用法的信息向后翻页，仔细阅读当前显示的信息

3）按下q键，结束rm命令用法信息显示

4）使用rm删除文件

**# cd /root**

**# rm –rf \***

**# man rm > file1**

**# man rm > file2**

**# mkdir folder1**

**# ls**

查看当前目录下的内容信息

**# rm file2**

rm: 是否删除一般文件‘file2’？**y**

**# ls**

file1 folder1

**# rm folder2**

rm: 是否删除目录‘folder2’？**y**

rm: 无法删除目录‘folder2’：是一个目录

运行以下命令尝试删除文件或目录

**# rm –i file1**

**# rm –f file1**

**# rm –r foler2**

**# rm –rf file1**

**# rm –rf folder2**

**注意：由于rm命令的-rf选项是强制删除文件或目录，没有确认模式，所以用户尤其是root用户要谨慎使用rm的-rf选项，切勿删除根目录下的文件目录。**

## find命令

【实验目的】熟练掌握Linux系统中的find命令的用法

【实验结果】搜索文件并执行相应的操作

find命令可以用来查询文件并可以在查询之后执行相应操作。

* **find + 查询目录 + 选项1 + 查询参数 + [选项2] + [执行操作]**
* **选项1 –name: 查询文件名**
* **选项2 –print: 将匹配结果输出到标准输出设备**
* **选项2 –exec: 搜索到文件后执行后续命令，常见格式为：**

**-exec *command* {} \;**

1）查看find命令的用法

**# man find**

输出的结果是find命令的先关用法，仔细阅读该命令的用法信息，直至当前显示结尾。

2）按下空格键，进行翻页显示，直至“[END]”字符

显示的find命令用法的信息向后翻页，仔细阅读当前显示的信息

3）按下q键，结束find命令用法信息显示

4）使用find命令查找文件

**# find /etc –name K92iptables**

查看输出结果

**# find /etc –name \*tables\***

查看输出结果

**# find /etc –name \*tables\* -print**

查看输出结果

**# find /etc –name \*tables\* -exec echo find it {} \;**

查看输出结果，与上面几条命令结果有何不同

## df命令

【实验目的】熟练掌握Linux系统中的df命令的用法

【实验结果】查看磁盘信息

df命令可以用来查询当前系统的磁盘信息。

* **df + [选项]**
* **不带选项默认以k字节为单位显示磁盘信息**
* **选项-h：以G字节为单位显示磁盘信息**

1）查看df命令的用法

**# man df**

输出的结果是df命令的先关用法，仔细阅读该命令的用法信息，直至当前显示结尾。

2）按下空格键，进行翻页显示，直至“[END]”字符

显示的df命令用法的信息向后翻页，仔细阅读当前显示的信息

3）按下q键，结束df命令用法信息显示

4）使用df命令查看磁盘信息

**# df**

**# df –k**

**# df –h**

**查看上述三条命令结果，比较有何不同**

## gzip/gunzip命令

【实验目的】熟练掌握Linux系统中的gzip/gunzip命令的用法

【实验结果】将目标文件或目录压缩成gz类型文件；将gz类型文件进行解压

gzip/gunzip命令可以用来解压缩gz类型的文件

* **gzip + [选项] + 操作对象**
* **gunzip + [选项] + 操作对象**

1）查看gzip命令的用法

**# man gzip**

输出的结果是gzip命令的先关用法，仔细阅读该命令的用法信息，直至当前显示结尾。

2）按下空格键，进行翻页显示，直至“[END]”字符

显示的gzip命令用法的信息向后翻页，仔细阅读当前显示的信息

3）按下q键，结束gzip命令用法信息显示

4）查看gunzip命令的用法

**# man gunzip**

输出的结果是gunzip命令的先关用法，仔细阅读该命令的用法信息，直至当前显示结尾。

5）按下空格键，进行翻页显示，直至“[END]”字符

显示的gunzip命令用法的信息向后翻页，仔细阅读当前显示的信息

6）按下q键，结束gunzip命令用法信息显示

7）使用gzip和gunzip进行解压缩

**# cd /root**

**# rm –rf \***

**# man gzip > gzip\_help.file**

**# man gunzip > gunzip\_help.file**

**# ls**

**gunzip\_help.file gzip\_help.file**

**# gzip \***

**# ls**

**查看当前目录内容信息**

**# gunzip \***

**# ls**

**查看当前面目录内容信息，比较gzip和gunzip操作前后的区别**

## ps/kill命令

【实验目的】熟练掌握Linux系统中的ps/kill命令的用法

【实验结果】查询系统进程运行状态；关闭系统运行的进程

ps命令可以用来查看系统进程状态

* **ps + [选项]**
* **选项-a：显示包括其他用户的所有进程**
* **选项-e：显示当前所有进程**
* **选项-f：用树形格式显示进程**
* **选项-u：按照用户名或者启动时间顺序来显示进程**
* **选项-x：显示无控制终端的进程**
* **使用ps命令查看成寻运行状态常用选项组合有-aux或者-ef**

kill命令可以用来删除系统中正在运行的进程

* **kill + [进程PID]**

1）查看ps命令的用法

**# man ps**

输出的结果是ps命令的先关用法，仔细阅读该命令的用法信息，直至当前显示结尾。

2）按下空格键，进行翻页显示，直至“[END]”字符

显示的ps命令用法的信息向后翻页，仔细阅读当前显示的信息

3）按下q键，结束ps命令用法信息显示

4）查看kill命令的用法

**# man kill**

输出的结果是kill命令的先关用法，仔细阅读该命令的用法信息，直至当前显示结尾。

5）按下空格键，进行翻页显示，直至“[END]”字符

显示的kill命令用法的信息向后翻页，仔细阅读当前显示的信息

6）按下q键，结束kill命令用法信息显示

7）使用ps和kill命令来关闭某个进程

**# ps -aux**

**USER PID %CPU %MEM VSZ RSS TTY STAT START TIME COMMAND**

**…**

**root 1932 … … command**

**…**

**# kill 1932**

**# ps -ef**

**UID PID PPID C STIME TTY TIME CMD**

**…**

**root 1932 … … command**

**…**

**# kill 1932**

**注意：此处可以通过管道和grep设定过滤器，来显示满足条件的进程，方便查询。**

## Linux命令综合训练

1. 查看vi命令的使用用法。
2. 将vi命令的使用方法导入文件中，并将其传送至宿主机（本机）上。
3. 将vi命令的使用用法压缩成gz类型的文件，并将其传送至宿主机上。
4. 从宿主机上的文件上传到虚拟机中，然后将其压缩成gz文件，传回宿主机。
5. 查找/var/log目录下的带有maillog字样的文件，将其复制到/root目录下，并创建新的文件夹，将所有文件移动到新建文件夹内，先用tar命令对文件夹进行打包，然后使用gzip命令将其压缩成gz文件，上传到宿主机。
6. 将vi命令的用法导入到文件1中，将该文件的面向所有用户的所有权限取消，使其变成一个不可读不可写不可执行的文件；将在vi命令的使用方法导入文件2中，将该文件的面向所有用户的所有权限都添加上；将vi命令的用法导入到文件3中，只保留该文件的面向所有用户的可读权限；将vi命令导入到文件4中，将该文件的面向所属用户的权限全部保留，面向该用户组和其他用户的权限全部取消。

# Linux系统的工具

Linux系统内部提供了很多工具供Linux用户使用，其中vi编辑器、数据流定向、管道等。

## vi编辑器

vi编辑器是Linux系统提供的文本编辑器，用户可以使用vi编辑器进行文件编辑，可以完成普通文档、编程文件的创建和修改。

1. 利用vi编辑器编写一个简单的文档

**跳转到/root目录下，创建vi\_test文件夹，跳转到vi\_test目录下。**

**# vi hello.file**

**进入vi编辑器模式之后，即进入了一般模式；点击i键，即可切换到编辑模式。在编辑模式下，输入一下内容。**

**############################**

**##This is a test file for vi editor##**

**############################**

**Hello, Linux!**

**在编辑模式下，可以输入任何字符串，也可以使用光标箭头控制输入光标的位置，在完成编辑之后，可以按[ESC]键，输入“：wq”即可强制保存。**

**然后通过more命令查看此文件内容。**

**# more hello.file**

1. 利用vi编辑器编写一个C语言文件

**跳转到/root目录下，创建vi\_test文件夹，跳转到vi\_test目录下。**

**# vi hello.c**

**进入vi编辑器模式之后，即进入了一般模式；点击i键，即可切换到编辑模式。在编辑模式下，输入一下内容。**

**//This is C resource file for vi editor##**

**#include<stdio.h>**

**void main(void)**

**{**

**printf(“This is a Linux C resource file!\n”);**

**}**

**在编辑模式下，可以输入任何字符串，也可以使用光标箭头控制输入光标的位置，在完成编辑之后，可以按[ESC]键，输入“：wq”即可强制保存。**

**然后通过more命令查看此文件内容。**

**# more hello.c**

1. 利用vi编辑器编写一个简单的shell文件

**跳转到/root目录下，创建vi\_test文件夹，跳转到vi\_test目录下。**

**# vi hello.sh**

**进入vi编辑器模式之后，即进入了一般模式；点击i键，即可切换到编辑模式。在编辑模式下，输入一下内容。**

**##This is shell resource file for vi editor##**

**echo hello Linux!**

**在编辑模式下，可以输入任何字符串，也可以使用光标箭头控制输入光标的位置，在完成编辑之后，可以按[ESC]键，输入“：wq”即可强制保存。**

**然后通过more命令查看此文件内容。**

**# more hello.sh**

1. 利用vi编辑器对已有的文档进行修改

**跳转到/root目录下，创建vi\_test文件夹，跳转到vi\_test目录下。**

**# vi hello.sh**

**进入vi编辑器模式之后，即进入了一般模式；点击i键，即可切换到编辑模式。在编辑模式下，移动光标至第一行尾，点击回车键，输入一下内容。**

**##Add some content for shell resource file with vi editor##**

**ls**

**可以按[ESC]键，输入“：wq”即可强制保存。**

**然后通过more命令查看此文件内容。**

**# more hello.sh**

显示内容应为以下内容：

**##This is shell resource file for vi editor##**

**##Add some content for shell resource file with vi editor##**

**ls**

**echo hello Linux!**

## 数据流定向

数据流定向是将某个命令执行后的在屏幕上的输出信息传送到其他地方，比如文件、设备等。

* **数据流定向**
* **标准输入(stdin)：代码为0**
* **标准输出(stdout)：代码为1**
* **标准错误输出(stderr)：代码为2**
* **数据流定向符**
* **针对输入设备的定向，使用 < 或者 <<**
* **针对输出设备的定向，使用 > 或者 >>**

一个箭头的意思是覆盖重写，两个箭头的意思是追加。

1）使用数据流定向将屏幕的正确信息传送至文件

**# ls /**

该命令是显示/目录（根目录）下的内容信息

**# ls / > list.file**

**# more list.file**

查看list.file的内容

**# ls /etc > list.file**

**# more list.file**

查看list.file有什么变化

**# ls / >> list.file**

**# more list.file**

查看list.file有什么变化

2）使用数据流定向将屏幕的错误信息传送至文件

**# cd /root**

**# ls**

该命令是显示/root目录下的内容信息，确保没有no\_exist\_folder no\_exist\_another目录存在，如果存在，请使用rmdir或rm命令删除。

**# cd no\_exist\_folder**

bash: cd : no\_exist\_folder: 没有那个文件或目录

**# cd no\_exist\_folder 2 > error.file**

注意查看此时屏幕上是否有信息输出

**# more error.file**

查看error.file文件内容

**# cd no\_exist\_another 2 > error.file**

**# more error.file**

查看error.file有什么变化

**# cd no\_exist\_folder 2 >> error.file**

注意查看此时屏幕上是否有信息输出

**# more error.file**

查看error.file文件内容变化

## 管道

管道就是将一个命令的标准输出信息，通过“|”传递给另外一个命令进行处理。

1）使用管道进行命令拼接

**# cd /root**

**# ls /etc**

在输出结果中人工查找ip开头的文件或目录，在找到之后，运行以下命令

**# ls /etc | grep ip**

**# ps -ef**

在输出结果中人工查找bash的进程，在找到之后，运行以下命令

**# ps –ef | grep bash**

## Linux系统工具综合训练

1. **使用Linux命令和工具完成如下综合训练**
2. 新建一个文件夹，跳转到该文件夹下，然后进行以下操作：（如果在上一个综合训练中完成下一步操作，可直接使用，不必再重复第b步骤）
3. 将vi命令的用法导入到文件1中，将该文件的面向所有用户的所有权限取消，使其变成一个不可读不可写不可执行的文件；将在vi命令的使用方法导入文件2中，将该文件的面向所有用户的所有权限都添加上；将vi命令的用法导入到文件3中，只保留该文件的面向所有可读权限；将vi命令导入到文件4中，将该文件的面向所属用户的权限全部保留，面向其他用户的权限全部取消。
4. 查看该文件夹下的所有文件属性，然后将屏幕上显示的文件属性的信息传输到file\_properties文件中。使用vi命令打开该文件，在文件末尾添加以下内容并保存：

### This is added via vi editor###

Hello, this sentence is added through vi editor

1. 使用数据流定向，将echo命令输出的字符串追加到file\_properties末尾，输入的字符串内容如下：

### This is added via Data flow oriented###

Hello, this sentence is added through Data flow oriented

1. 将该文件发送至宿主机，在宿主机中通过文本编辑器打开该文件，添加如下内容并保存。

### This is added via text editor in host machine###

Hello, this sentence is added through text editor in host machine

1. 利用管道和数据流重定向，分别将上述三句话传送到vi\_editor.file, data\_flow.file, host\_machine.file
2. 使用more命令查看三个文件的内容
3. **使用Linux命令和工具完成如下综合训练**
4. 使用vi编辑器创建一个shell文件，文件名为test.sh，在编辑模式输入以下内容并保存。

##This is a shell file

#!/bin/sh

ls

echo hello Linux!

cd no\_exist\_folder

1. 运行以下命令，运行shell。

# sh test.sh

此时，屏幕上会输出ls命令、echo命令、cd命令的运行结果。由于ls和echo命令都能正确运行，所以屏幕上显示输出的是正确信息，由于no\_exist\_folder目录不存在，所以cd命令将会出错，屏幕上显示的错误信息。

1. 使用chmod命令将test.sh的权限全都添加上。
2. 运行命令

# ./test.sh

1. 通过管道，根据上一步运行后屏幕上输出的错误信息提示，将shell文件运行结果的错误信息输出在屏幕上；通过管道，根据上一步运行后屏幕上输出的正确信息，分别将ls命令的输出结果和echo命令的输出结果输出在屏幕上
2. 通过管道和数据流定向，先将上述运行结果中的ls命令的结果输送至right.file文件中，然后将echo命令的输出结果追加到right.file文件中；通过管道和数据流定向，先将上述运行结果中的cd命令的输出结果传输到error.file文件中。
3. 通过数据流定向的代码号，分别将正确输出和错误输出的结果重写到right.file和error.file中。

# 课后作业

注意：以下作业要求在虚拟机上的所有命令的标准输出都要有log文件进行记录。

1. 在/root目录下创建man\_test目录，并在该目录下，将man命令的使用方法的帮助信息写入man.file中，并将其压缩成gz格式文件，传送至宿主机，在宿主机上将其重命名为man.gz，然后重新导入回虚拟机中，在/root目录下将其解压，并查看文件内容。

**要求：**所有在Linux虚拟中的命令操作的屏幕上显示正确结果全部保存在man.log文件中。

1. 创建一个文档，不输入任何字符进行保存。然后再次编辑此文档，输入任意一串字符串。然后通过echo命令输出以下内容，并将其重写入新建的文档中。

###This file shows the root directory content###

然后将根目录下的所有文档信息（包含文件属性信息、大小、创建日期）等追加到此文档中。然后将该文件权限更名为list.file，修改其权限，对所有用户只有可读权限。

然后使用vi对其进行编辑（如果使用wq保存不成功，为其添加相应权限），在文件末尾添加如下内容并保存。

###This is the end of the list.file file###

**要求：**所有在Linux虚拟中的命令操作的屏幕上显示正确结果全部保存在list.log文件中。

1. 在命令行输入export，将该命令的输出重定向到export.file，同时将标准输出的设备重定向到export.file1，通过一条命令完成。完成之后，将该条命令的运行结果通过数据流定向的方式送到grep.log。

然后通过管道命令，显示export.file文件中的带有SSH字样的行，将结果显示在屏幕上。

然后将带有SSH输出到export.file1，并将其内部的内容覆盖。查看export.file1内容。使用cp命令，利用export.file文件内容覆盖export.file1文件内容，再次查看export.file1内容。对比cp前后的不同，通过echo命令将二者时间的不同追加到grep.log。

1. 在/root目录下创建一个log目录，将man.log，list.log，grep.log移动到该目录下，将该目录使用tar压缩，然后再次压缩生成gz类型文件，提交给老师。