

## 实验一：常用命令和 vi 编辑器

### *Hollow Man*

## 实验目的

1. 熟练使用 Linux 字符界面的常用命令。
2. 熟练运用 Linux 的在线求助系统

## 实验时间

3 学时

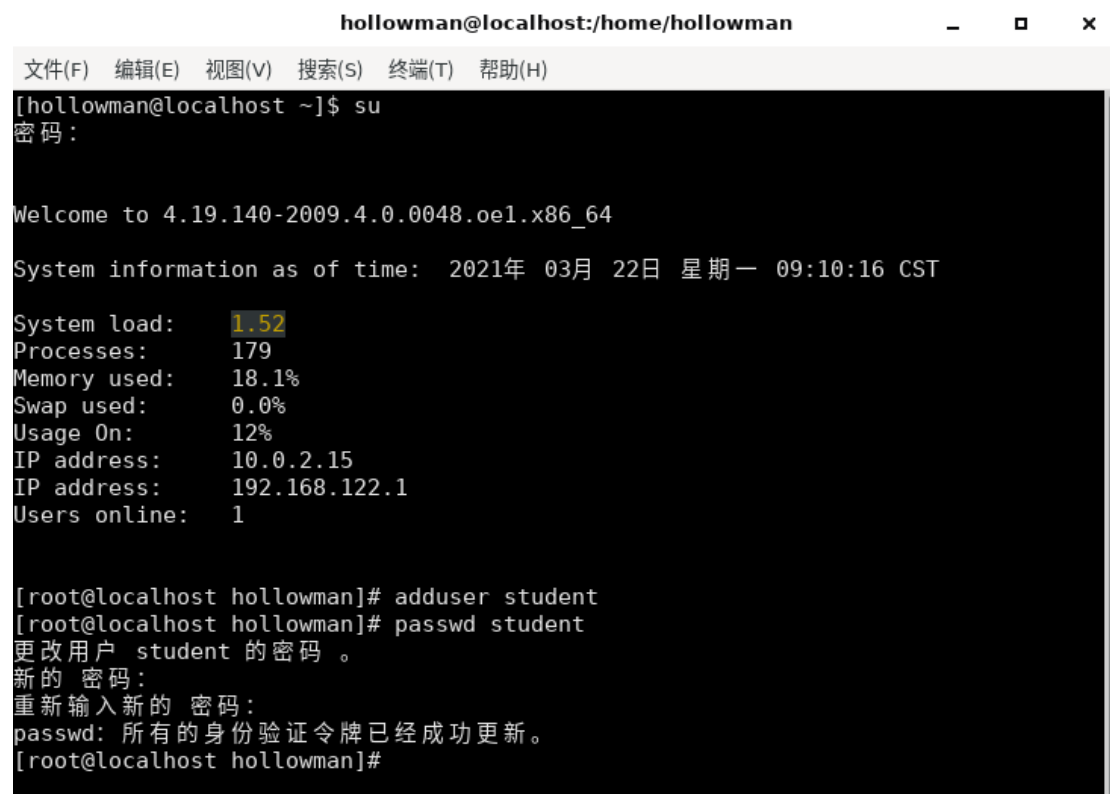
## 实验内容

(一) Linux 操作基础

1. 新建一个用户,用此用户名和口令登陆系统.注意屏幕上列出的提示信息

[解答] 1)adduser student

2)passwd student



```
hollowman@localhost:/home/hollowman
文件(F) 编辑(E) 视图(V) 搜索(S) 终端(T) 帮助(H)
[hollowman@localhost ~]$ su
密码:

Welcome to 4.19.140-2009.4.0.0048.oe1.x86_64

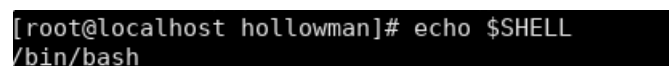
System information as of time: 2021年 03月 22日 星期一 09:10:16 CST

System load:      1.52
Processes:        179
Memory used:      18.1%
Swap used:        0.0%
Usage On:         12%
IP address:       10.0.2.15
IP address:       192.168.122.1
Users online:     1

[root@localhost hollowman]# adduser student
[root@localhost hollowman]# passwd student
更改用户 student 的密码。
新的 密码:
重新输入新的 密码:
passwd: 所有的身份验证令牌已经成功更新。
[root@localhost hollowman]#
```

2. 查看系统的命令提示符,确定系统使用的 SHELL 程序类型

[解答] 1) echo \$SHELL



```
[root@localhost hollowman]# echo $SHELL
/bin/bash
```

3.用 who 命令来查看当前登陆在系统中的用户列表、用户总数和系统启动时间

[解答] 1) who

2) who -q

3) who -b

```
hollowman@localhost:/home/hollowman
文件(F) 编辑(E) 视图(V) 搜索(S) 终端(T) 帮助(H)
[root@localhost hollowman]# who
hollowman tty1          2021-03-22 08:56 (:0)
[root@localhost hollowman]# who -q
hollowman
# 用户数=1
[root@localhost hollowman]# who -b
      系统引导 2021-03-22 08:54
[root@localhost hollowman]#
```

5.系统中的帮助系统工具：help,info,man (man 是手册 manual 的缩写)

[解答] 1) info echo

2) help echo

3) man echo

```
hollowman@localhost:/home/hollowman
文件(F) 编辑(E) 视图(V) 搜索(S) 终端(T) 帮助(H)
[root@localhost hollowman]# info echo
info: No menu item 'echo' in node '(dir)Top'
[root@localhost hollowman]# help echo
echo: echo [-neE] [参数 ...]
      Write arguments to the standard output.

      Display the ARGs, separated by a single space character and followed by a
      newline, on the standard output.

      Options:
      -n      do not append a newline
      -e      enable interpretation of the following backslash escapes
      -E      explicitly suppress interpretation of backslash escapes

      `echo' interprets the following backslash-escaped characters:
      \a      alert (bell)
      \b      backspace
      \c      suppress further output
      \e      escape character
      \E      escape character
      \f      form feed
      \n      new line
      \r      carriage return
      \t      horizontal tab
      \v      vertical tab
      \\      backslash
      \0nnn   the character whose ASCII code is NNN (octal).  NNN can be
              0 to 3 octal digits
      \xHH    the eight-bit character whose value is HH (hexadecimal).  HH
              can be one or two hex digits
      \uHHHH  the Unicode character whose value is the hexadecimal value HHHH.
              HHHH can be one to four hex digits.
      \UHHHHHHHH the Unicode character whose value is the hexadecimal value
              HHHHHHHH.  HHHHHHHH can be one to eight hex digits.
```

```
Exit Status:
Returns success unless a write error occurs.
[root@localhost hollowman]# man echo
没有 echo 的手册页条目
```

6.用 date 命令查看当前的日期和时间

[解答]date

```
[root@localhost hollowman]# date
2021年 03月 22日 星期一 09:16:21 CST
```

7.用 cal 命令查看 2003 年 1 月 1 日是星期几

[解答]cal -m 1 2003

```
hollowman@localhost:/home/hollowman
文件(F) 编辑(E) 视图(V) 搜索(S) 终端(T) 帮助(H)
[root@localhost hollowman]# cal -m 1 2003
一月 2003
一 二 三 四 五 六 日
      1  2  3  4  5
 6  7  8  9 10 11 12
13 14 15 16 17 18 19
20 21 22 23 24 25 26
27 28 29 30 31
[root@localhost hollowman]#
```

8.查看 2003 年的日历表

[解答]cal -y 2003

```
hollowman@localhost:/home/hollowman
文件(F) 编辑(E) 视图(V) 搜索(S) 终端(T) 帮助(H)
[root@localhost hollowman]# cal -y 2003
2003

    一月                二月                三月
一 二 三 四 五 六 日  一 二 三 四 五 六 日  一 二 三 四 五 六 日
    1  2  3  4  5          1  2          1  2
  6  7  8  9 10 11 12    3  4  5  6  7  8  9    3  4  5  6  7  8  9
13 14 15 16 17 18 19    10 11 12 13 14 15 16   10 11 12 13 14 15 16
20 21 22 23 24 25 26    17 18 19 20 21 22 23   17 18 19 20 21 22 23
27 28 29 30 31         24 25 26 27 28         24 25 26 27 28 29 30
                                     31

    四月                五月                六月
一 二 三 四 五 六 日  一 二 三 四 五 六 日  一 二 三 四 五 六 日
    1  2  3  4  5  6          1  2  3  4          1
  7  8  9 10 11 12 13    5  6  7  8  9 10 11    2  3  4  5  6  7  8
14 15 16 17 18 19 20    12 13 14 15 16 17 18    9 10 11 12 13 14 15
21 22 23 24 25 26 27    19 20 21 22 23 24 25   16 17 18 19 20 21 22
28 29 30                26 27 28 29 30 31        23 24 25 26 27 28 29
                                     30

    七月                八月                九月
一 二 三 四 五 六 日  一 二 三 四 五 六 日  一 二 三 四 五 六 日
    1  2  3  4  5  6          1  2  3          1  2  3  4  5  6  7
  7  8  9 10 11 12 13    4  5  6  7  8  9 10    8  9 10 11 12 13 14
14 15 16 17 18 19 20    11 12 13 14 15 16 17   15 16 17 18 19 20 21
21 22 23 24 25 26 27    18 19 20 21 22 23 24   22 23 24 25 26 27 28
28 29 30 31            25 26 27 28 29 30 31    29 30

    十月                十一月                十二月
一 二 三 四 五 六 日  一 二 三 四 五 六 日  一 二 三 四 五 六 日
    1  2  3  4  5          1  2          1  2  3  4  5  6  7
  6  7  8  9 10 11 12    3  4  5  6  7  8  9    8  9 10 11 12 13 14
13 14 15 16 17 18 19    10 11 12 13 14 15 16   15 16 17 18 19 20 21
20 21 22 23 24 25 26    17 18 19 20 21 22 23   22 23 24 25 26 27 28
27 28 29 30 31         24 25 26 27 28 29 30   29 30 31

[root@localhost hollowman]#
```

## 9.切换用户

[解答]su student

```
[root@localhost hollowman]# su student

Welcome to 4.19.140-2009.4.0.0048.oel.x86_64

System information as of time:  2021年 03月 22日 星期一 09:21:25 CST

System load:      1.54
Processes:        180
Memory used:      18.3%
Swap used:        0.0%
Usage On:         12%
IP address:       10.0.2.15
IP address:       192.168.122.1
Users online:     1

[student@localhost hollowman]$
```

## (二) Linux 文件系统 (一)

### 1.显示当前目录

- [解答] 1) pwd  
2) echo \$PWD

```
student@localhost:/home/hollowman
文件(F) 编辑(E) 视图(V) 搜索(S) 终端(T) 帮助(H)
[student@localhost hollowman]$ pwd
/home/hollowman
[student@localhost hollowman]$ echo $PWD
/home/hollowman
```

2.改变到用户主目录

- [解答] 1) cd  
2) cd ~

```
[student@localhost hollowman]$ cd
[student@localhost ~]$ cd ~
[student@localhost ~]$
```

3.确认用户主目录

[解答]echo \$HOME

```
[student@localhost ~]$ echo $HOME
/home/student
```

4.当前目录下,创建名为 ABC 的新目录

[解答]mkdir ABC

```
[student@localhost ~]$ mkdir ABC
```

5.在 ABC 目录创建名为 abc 的新文件

[解答]touch abc

```
[student@localhost ~]$ cd ABC
[student@localhost ABC]$ touch abc
```

6.显示当前目录内容

- 1)按照字母逆序 ls -r  
2)按照长格式 ls -l  
3)按照水平格式 ls --format=horizontal  
4)列出所有文件, 包括不可见的文件 ls -a

```
[student@localhost ABC]$ ls -r
abc
[student@localhost ABC]$ ls -l
总用量 0
-rw-rw-r--. 1 student student 0 3月 22 09:24 abc
[student@localhost ABC]$ ls --format=horizontal
abc
[student@localhost ABC]$ ls -a
. .. abc
```

7.在屏幕上显示文件 abc 的内容

[解答]cat abc

```
[student@localhost ABC]$ cat abc  
[student@localhost ABC]$
```

8.删除当前目录下的 abc 文件.删除 abc 文件前，要求确认

[解答]rm -i ./abc

```
[student@localhost ABC]$ rm -i ./abc  
rm: 是否删除普通空文件 './abc'? y  
[student@localhost ABC]$
```

9. 返回上一级目录

[解答]cd ..

```
[student@localhost ABC]$ cd ..  
[student@localhost ~]$
```

10.删除当前目录中的 ABC 目录

[解答]rmdir ABC

```
[student@localhost ~]$ rmdir ABC  
[student@localhost ~]$
```

### (三) Linux 文件系统(二)

1. 在用户主目录下创建名为 memos 的目录

[解答] 1) cd //返回用户主目录

2) mkdir memos //建立 memos 目录

```
[student@localhost ~]$ cd  
[student@localhost ~]$ mkdir memos  
[student@localhost ~]$
```

2. 用 vi 编辑器, 在用户主目录中创建名为 myfile 的文件, 输入几行文字, 保存退出

[解答] 1) vi myfile //建立 myfile 文件

2) i //进入编辑模式, 输入文字

3) ESC --> :wq //先按 Esc 再输入:wq, 存盘退出

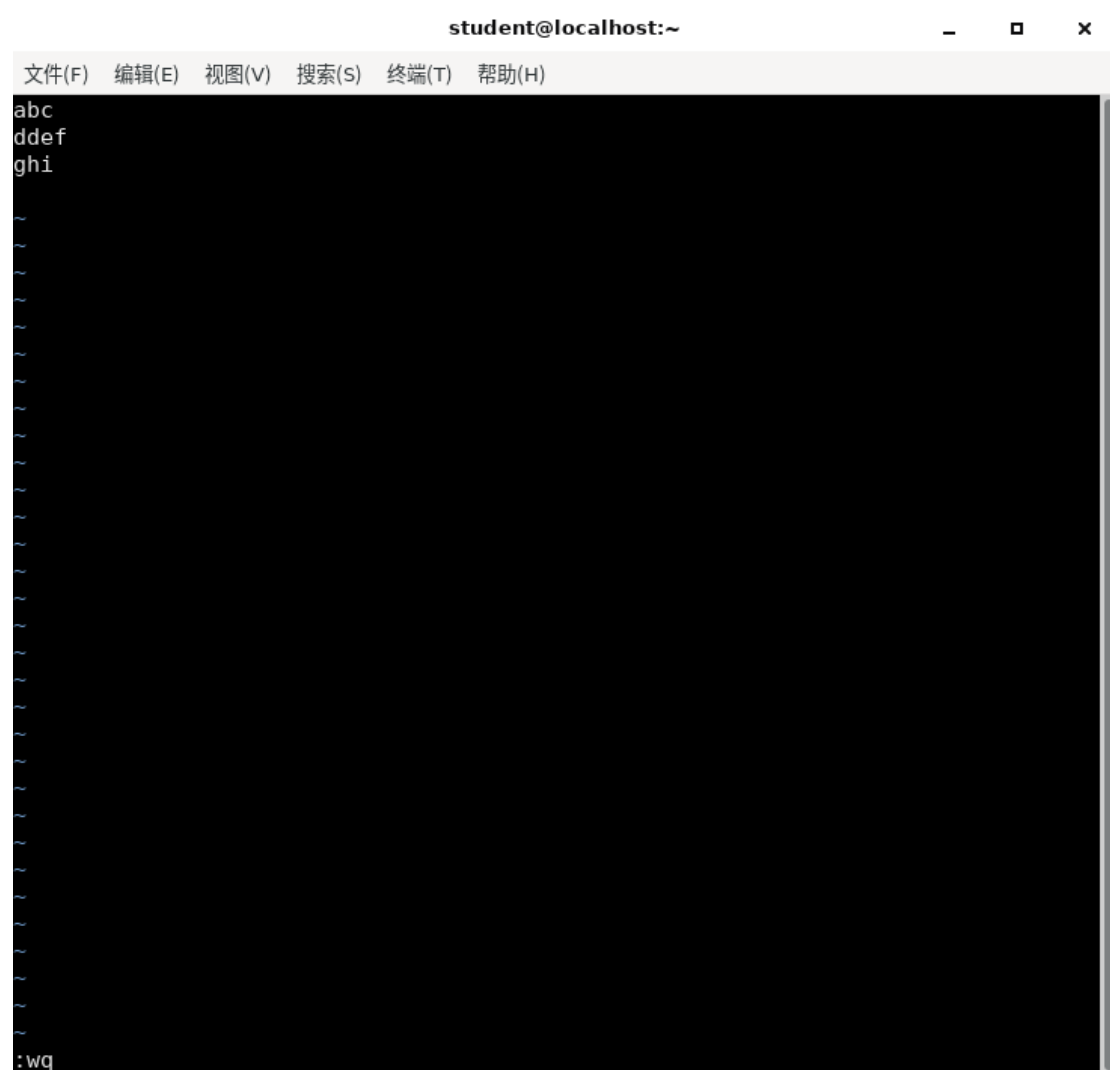
```
[student@localhost ~]$ vi myfile
```



**—      □      ×**

[illegible]





3. 用 cat 命令, 把 myfile 附加到新创建的名为 large 的文件中.

[解答] cat myfile>>large

```
[student@localhost ~]$ cat myfile>>large
```

4. 用 cp 命令把当前目录下的所有文件复制到目录 memos 下

[解答]cp \* ./memos

```
[student@localhost ~]$ cp * ./memos
cp: 未指定 -r; 略过目录 'memos'
```

5. 用 `ln` 命令创建 `large` 的一个别名

[解答]ln large large. old

```
[student@localhost ~]$ ln large large.old
```

6. 用 mv 命令, 把 large 更名为 large.new

[解答]mv large large.new

```
[student@localhost ~]$ mv large large.new
```

7. 用 mv 命令, 把 large.new 移到 memos 下, 改名 large

[解答]mv large.new memos/large

```
[student@localhost ~]$ mv large.new memos/large
```

8. 显示一个文件的后五行

[解答] tail -n 5 large.old

```
[student@localhost ~]$ tail -n 5 large.old
abc
ddef
ghi

[student@localhost ~]$
```

9. 显示一个文件的前五行

[解答] head -n 5 large.old

```
[student@localhost ~]$ head -n 5 large.old
abc
ddef
ghi
```

10. 把一个文件的最后 10 个字符存到另一个文件中

[解答] tail -c 10 large.old >>newfile

```
[student@localhost ~]$ tail -c 10 large.old >>newfile
```

11. 用 more, less 命令读大文件

[解答] 1) more newfile

2) less newfile

```
[student@localhost ~]$ more newfile
ddef
ghi

[student@localhost ~]$ less newfile
```

```
ddef
ghi
newfile (END)
```

12. 统计指定文本文件的行数

[解答] wc -l newfile

```
student@localhost:~
文件(F) 编辑(E) 视图(V) 搜索(S) 终端(T) 帮助(H)
[student@localhost ~]$ wc -l newfile
3 newfile
```

13. 保存用户主目录中所有 7 天前创建的文件列表

[解答] 1) cd ~

2) find ./ -ctime 7 -name "\*" > 7days

```
[student@localhost ~]$ find ./ -ctime 7 -name "*" > 7days
[student@localhost ~]$
```

14. 在当前目录下查找名为 passwd 的文件

[解答] find ./ -name "passwd"

```
[student@localhost ~]$ find ./ -name "passwd"
[student@localhost ~]$
```

15. 从根目录开始, 查找所有 7 天前创建的文件

[解答] find ./ -atime +7

```
[student@localhost ~]$ find ./ -atime +7
[student@localhost ~]$
```

16. 用 paste 命令, 将两个文件连在一起

[解答] paste large.old myfile>myfile2

```
[student@localhost ~]$ paste large.old myfile>myfile2
[student@localhost ~]$
```

17. 在指定文本文件中, 查找指定字符串

[解答] grep abc myfile

```
[student@localhost ~]$ grep abc myfile
abc
```

#### (4) vi 编辑器的使用

详见文档 vi.doc。要求熟悉:

##### 1. 三种工作模式及其转化

Vi 有三种基本的工作模式: 指令行模式、文本输入模式、末行模式。

指令模式(Command Mode) 下输入 a、i、o 进入文本输入模式(Input Mode)

文本输入模式(Input Mode) 下按 ESC 进入指令模式(Command Mode)

指令模式(Command Mode) 下输入: 进入末行模式 (Last line Mode)

末行模式 (Last line Mode) 下指令错误则返回指令模式(Command Mode)

##### 2. 常用编辑命令

###### 1、指令模式(Command Mode)

指令模式主要使用方向键移动光标位置进行文字的编辑, 常用的操作命令及含义。

0 ————— 光标移动至行首

h ————— 光标左移一格

l ————— 光标右移一格

j ————— 光标下移一行

k ————— 光标上移一行

\$+A ————— 将光标移动到该行最后

PageDn ————— 向下移动一页

PageUp ————— 向上移动一页

d+方向键 ————— 删除文字

dd ————— 删除整行

pp ————— 整行复制

r ————— 修改光标所在的字符

S ————— 删除光标所在的列, 并进入输入模式

###### 2、文本输入模式 (Input Mode)

在指令模式下 (Command Mode) 按 a / A 键、i / I 键、o / O 键进入文本模式，文本输入模式的命令及其含义如下所示。

a ----- 在光标后开始插入  
A ----- 在行尾开始插入  
i ----- 从光标所在位置前面开始插入  
I ----- 从光标所在列的第一个非空白字元前面开始插入  
o ----- 在光标所在列下新增一行并进入输入模式  
O ----- 在光标所在列上方新增一行并进入输入模式  
ESC ----- 返回命令行模式

### 3、末行模式 (Last line Mode)

末行模式主要进行一些文字编辑辅助功能，比如字符串搜索、替代、保存文件等操作。主要命令如下

: q ----- 结束 Vi 程序，如果文件有过修改，先保存文件  
: q! ----- 强制退出 Vi 程序  
: wq ----- 保存修改并退出程序  
: set nu ----- 设置行号

### 3. 查找、替换

搜索及替换命令：

/pattern: 从光标开始处向文件尾搜索 pattern

?pattern: 从光标开始处向文件首搜索 pattern

n: 在同一方向重复上一次搜索命令

N: 在反方向上重复上一次搜索命令

: s/p1/p2/g: 将当前行中所有 p1 均用 p2 替代

: n1,n2s/p1/p2/g: 将第 n1 至 n2 行中所有 p1 均用 p2 替代

: g/p1/s//p2/g: 将文件中所有 p1 均用 p2 替换

## 思考题

### 1. 如果忘记某个命令的用法，怎么办？

1 使用 help+命令字

比如 help pwd，这种获得帮助的方法仅针对于内部命令

2 使用命令字+--help 查看

这种获得帮助的方法是针对于外部命令的

3 手册页即 man

无论是内部还是外部通吃，而且注重命令的使用方法

4 用的不多 info+命令

### 2. 如何使用：命令历史？

上下方向键控制

压缩、解压命令: `compress`, `zip`, `gzip`, `tar` 的用法。

#### Compress

执行命令后会替代目标文件

压缩            `compress [-rcv]` 文件名或目录名

解压缩        `uncompress` 文件名.z

执行命令不替代目标文件

压缩            `compress -c` 目标文件名 > 新文件名

#### zip

`zip -r myfile.zip ./` 将当前目录下的所有文件和文件夹全部压缩成 `myfile.zip` 文件, `-r` 表示递归压缩子目录下所有文件.

`zip -d myfile.zip smart.txt`

删除压缩文件中 `smart.txt` 文件

`zip -m myfile.zip ./rpm_info.txt`

向压缩文件中 `myfile.zip` 中添加 `rpm_info.txt` 文件

#### gzip

可以解开 `compress`, `zip` 和 `gzip` 压缩的文件, 压缩文件扩展名为 `gz`

`gzip [-cdtv#]` 文件名或目录名

执行命令不替代目标文件

`gzip -c` 目标文件名 > 新文件名

`gzip -d` 目标文件名 > 新文件名

#### tar

打包与压缩: `tar [-j|-z] [cv] [-f 文件名]`

查看文件:        `tar [-j|-z] [tv] [-f 文件名]`

解压缩:            `tar [-j|-z] [xv] [-f 文件名] [-c 目录]`