

1 Linux常用命令

在命令行中要输入的内容，或者是终端上显示的内容，会另起一行来说明

1.1 环境设置

1.1.1 调整时间

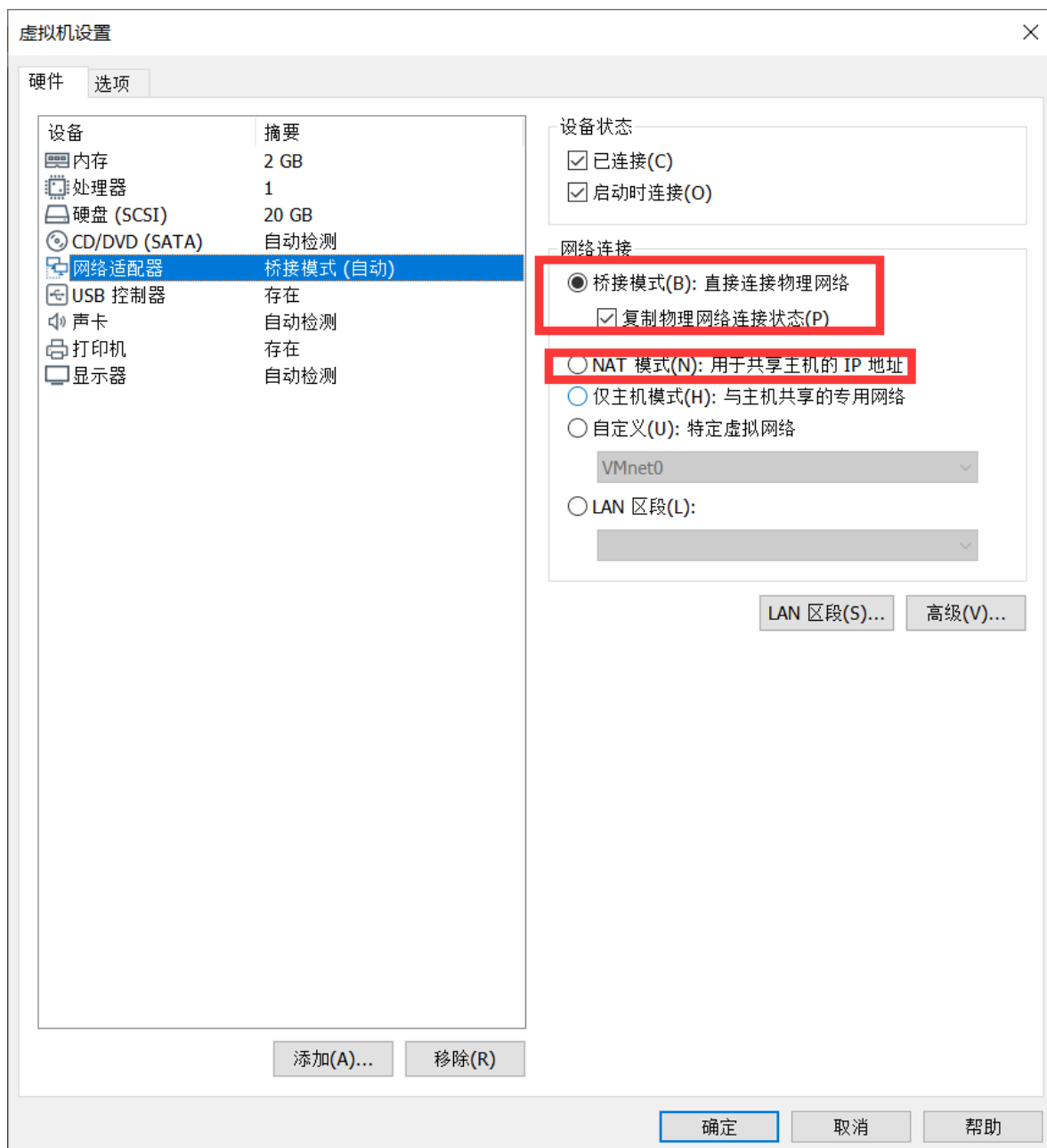
选项 --> 详细 --> 时间和日期，将时区调整到上海即可

1.1.2 使用xshell

xshell是一款终端模拟软件，它可以使用ssh协议连接到远程服务器上。在本课程，我们使用xshell连接到我们的虚拟机。在将来的工作中，服务器通常放置在机房当中，程序员们通常是使用xshell、putty等ssh软件来连接到服务器，从而能够在服务器上面编程

使用xshell之前，需要保证虚拟机和主机能连接上网络

根据自己的网络情况，选择NAT模式或者是桥接模式。



在shell里面使用ifconfig 查看自己的IP，如果提示没有此软件，则键入apt install命令来安装，输入命令之后要键入自己用户的密码：

```
$sudo apt install net-tools
```

#使用sudo命令以后需要输入用户口令（注意输入内容的时候，屏幕是没有变化的）

然后再使用ifconfig命令查看本机的网络信息：

```
$ifconfig
```

如果安装失败，通常的原因是本机的软件列表没有更新，使用下面的命令可以更新软件列表，然后再来重新执行安装的指令：

```
$sudo apt update
```

在虚拟机上ping百度，确定虚拟机能连接上网络：

```
Sping www.baidu.com
#使用ctrl+c可以终止
```

然后需要在虚拟机上安装ssh协议的客户端：

```
$sudo apt install ssh
```

使用ps命令，可以确认ssh客户端有没有运行：

```
$ps -elf|grep sshd
```

使用apt install命令来安装vim编辑器：

```
$ sudo apt install vim
```

键入vim，再键入tab键（**注意不要使用回车键!**），如果会联想vim相关命令，就说明安装成功

静态ip配置之桥接模式（适用于使用路由器连接网络的同学）

1. 选择一个合适的静态ip，不要和其他Windows、Linux或者手机等设备的IP冲突
2. 如何判断IP冲突没有？在shell或者是Windows下的命令行里面使用ping命令测试你想要的IP地址，如果能ping通，那么就不要再使用这个IP地址了
3. 打开虚拟机的网络设定，进入IPV4分页，选择manual。修改IP，网络掩码（255.255.255.0）和网关（这个保持和自动情况下一致即可）
4. 修改DNS让它和Windows里面的DNS一致（例如：202.103.44.150）
5. 关闭网络，然后重新打开，再使用ifconfig来检查是否设置成功

桥接模式定位网络问题

1. ping本机IP，如果ping不通，则将IP修改为DHCP后，再重新查看IP
2. 检查虚拟机是否使用桥接模式
3. 修改虚拟网络编辑器，然后还原默认设置，然后选择一个正确的网络适配器

静态ip配置之NAT模式（适用于所有同学）

1. 在虚拟机中选择NAT模式，并且将Ubuntu的网络设置设为自动
2. 检查当前能否ping同百度，若不能则尝试重启网络
3. 记录当前的ip地址、网关和DNS
4. 将Ubuntu的网络设置为静态的，其中所有的信息和第3步记录的内容一致
5. 关闭网络，然后重新打开，再使用ifconfig来检查是否设置成功

使用xshell

1. 先在命令行窗口ping通虚拟机
2. 打开xshell，点击“新建”，将名称修改为自己想要的名字，将主机名修改为虚拟机的IP地址
3. 进入“用户身份验证”界面，输入账户和密码，然后点击“连接”就可以连上虚拟机了

1.2 shell命令

1.2.1 用户配置

用户是Linux的使用者

- Linux是多用户的系统，一个用户可以多次（同时）登录一个Linux系统，多个用户也可以（同时）登录一个Linux系统。（与Windows桌面版本不一样）
- 使用Linux系统就像是入住一间豪宅。豪宅有它的主人，家人和客人；类似地，Linux系统的用户也分为普通用户和特权用户
- 普通用户有很多个，它们只能使用系统的部分功能，而超级用户只有一个，它的名字是root，它可以使用系统的所有功能--包括删除另外一个用户
- 对于部分普通用户，它们可以使用sudo命令来临时提高自己的权限，从而可以执行一些（而不是所有）特权命令，比如想要使用apt install命令在系统上安装软件，使用shutdown命令关机等等
- 普通用户的shell命令提示符是\$，root的shell命令提示符是#
- 每个用户都在一个组当中。组被用于将若干用户集合到项目或部门当中去。组机制允许同组的各个成员可以共享资源
- 为了节约空间，操作在保存用户和组信息的时候，只保存两个整型数字，称为用户ID和组ID。用户ID和组ID和用户名和组名是一一对应的

配置口令

Ubuntu安装的时候，root用户是没有口令的。所以需要首次进入的账户里面键入如下命令：

```
$sudo passwd root
```

先输入本账户的密码，以获取sudo权限（注意输入密码和Windows不一样，它不会显示键入了多少字符）然后输入root用户的密码即可

切换用户

切换到根用户，执行su命令

```
$su
```

- 切换到其他用户，在su命令后面添加用户名即可。从root用户进入其他用户的时候不需要密码

```
$su 用户名
```

显示所有用户

所有用户的信息是存放在文件/etc/passwd里面，所以使用查看文件内容的命令cat就能显示所有用户的信息

```
$cat /etc/passwd
```

使用之后会发现打印了大量信息，可以直接看最后面，因为最后面几行才是自定义的用户。打印的信息中每一行代表一个用户，一行里面依次显示的信息是：登录名、口令、用户ID、组ID、注释字段、家目录 and shell程序的位置（口令实际上存储在/etc/passwd当中）

可以看出，root用户的家目录是/root，而普通用户的家目录是/home/用户名

退出当前用户

使用exit指令可以退出当前用户。

```
$exit
```

用户使用su指令会进入其他用户，使用exit指令会回到当前的用户

当嵌套使用su指令来依次进入多个用户时候，多个用户是使用 **栈结构** 来管理的。执行su指令相当于将新用户压入栈顶，执行exit指令相当于弹出栈顶

当用户在上述的 **栈结构** 中存在的时候，那个用户是不能够被删除的

添加用户

使用useradd命令来添加用户

```
$useradd 用户名
```

只有root用户或者拥有sudo权限的用户使用sudo命令后才能添加用户

- 什么是拥有sudo权限的用户？

如果把root看成是宫中的皇帝，root作为皇帝自然是拥有至高无上的权力，但是这种权力往往也会带来大量的风险（比如下了一个错误的指令，例如删除所有的内容），另一方面，root也不希望事必躬亲。这个情况下，root可以将部分权力委托给一些普通用户（类似宫中的太监），这些普通用户只要拿到“圣旨”以后，就能以root的身份来处理事情。我们会在将来说明查看一个用户是否拥有sudo权限和设置sudo权限

在Linux系统中，没有消息就是最好的消息。通常命令执行成功时是不会有提示的，只有失败的时候才会在shell里面打印错误信息

添加用户并指定家目录

给useradd命令添加参数，在用户名之前使用-m，可以为用户添加默认家目录（如果不添加家目录，这个用户将无法创建文件）。使用-m参数的同时还需要使用-s参数来指定shell的位置（如果不添加shell的位置，用户的默认shell使用的是sh，它的功能比较弱）

```
$ useradd -m 用户名 -s /bin/bash
```

执行完命令，先使用pwd命令获取当前工作目录，使用cd命令进入/home目录，再使用ls命令显示当前目录下的所有文件。就会发现home的下面新建了一个新的目录，目录的名字就是用户名，这里就是新用户的家目录

```
$ pwd
$ cd /home
$ ls
```

- ~是家目录的简写，它和/home/用户名/是等价的

但是，目前的新用户是没有配置密码的，这也意味我们暂时不能够通过用户+密码的方式来登录用户。使用passwd命令可以给新用户配置密码，配置完成以后就能够正常登录了。

```
passwd 用户名
```

删除用户

- 使用userdel命令来删除用户

```
$userdel 用户名
```

- 如果用户正在使用中，那么这个用户就不能被删除
- 如果用户在su命令和exit命令构成的用户 **栈结构** 当中的时候，那么这个用户也不能被删除
- 在userdel后面添加-r选项，可以删除用户家目录下的文件

```
$userdel -r 用户名
```

安全设置

无法直接使用root的用户名来登录root用户，因为root用户的用户名是固定的，它被穷举法找出密码的概率更大

- 服务器可以设置安全措施，例如对连续输入3次错误密码的IP进行封禁处理
- 我们所使用的xshell就使用了ssh协议。ssh就是secure shell的简称，相较于其他的不安全的协议（这些协议传递口令都是明文传递的），比如telnet，tftp之类的，它提供了一种安全的远程登录协议

在日常工作当中，一般是不会使用root用户进行操作的，root用户拥有非常高的权限，它可以在任意位置去添加和删除文件，一方面不方便工作内容管理，另外很容易误操作带来严重的后果

1.2.2 文件相关命令

文件是存放磁盘上的数据

- Linux的文件系统是一套**单根**目录结构。
- 目录的根基是被称为 **根root** 的目录，它写作"/"
- 文件拥有不同的类型，比如目录文件、普通文件等等（目录也是一种文件）
- 由文件名组成的字符串，以"/"分隔称为**路径**。根目录开始的路径称为绝对路径，当前目录下的路径称为相对路径。最后一个"/"之前的部分称为**目录部分**，后面的称为**文件部分**

Linux系统自带目录的含义

目录名	含义
/bin	存放二进制文件，通常是一些可执行程序
/sys	系统文件
/dev	设备文件
/etc	配置文件和启动脚本
/home	普通用户家目录的父目录
/lib	系统库文件
/proc	特殊的目录，里面的内容是进程内容对文件系统的映射
/usr	普通用户可以访问的文件，包括库文件、二进制文件和代码文件等等
/var	存储变化很快的文件，比如日志

当前工作目录

在shell的命令输入栏左边，会提示当前目录的路径。其中，如果路径的最左边是~，那么路径的根是用户的家目录（即/home/用户名）；如果路径的最左边是/，那么路径的根是root目录。当前目录是相对路径的起始位置，也是进程的“当前位置”。使用pwd命令可以显示当前工作目录：

```
$pwd
```

查看当前工作目录下的所有文件

使用ls命令可以显示当前工作目录下的所有文件的文件名

```
$ls
```

无论是在哪个目录下面使用ls，都会发现显示结果必有二个目录文件，依次是.和..

- .代表的是当前目录，..代表的是上级目录
- 在ls后面可以添加路径名，这样可以打印对应路径下的所有文件的文件名

```
$ls /
```

与Windows不一样，Linux没有盘和盘符的概念，如果在物理上确实添加了一块磁盘（磁盘上也有文件系统），为了访问磁盘内容，那么Linux会将磁盘的文件系统的根目录“添加”（实际上是将磁盘内容映射到Linux文件系统里面）到Linux根目录下的某个目录上，这个过程称为挂载。使用mount命令可以查看当前系统的挂载信息。

```
$mount
```

- **熟练使用tab键的自动联想功能。**当你忘记想要输入的路径的时候，使用tab键会给你新的灵感。在某些情况下，如果你输入了路径的前缀，使用tab键会自动补全路径

在ls命令后面添加-a参数，可以显示隐藏文件（文件名以.开头的文件）

```
$ls -a
```

在ls命令后面添加-l参数，可以显示文件的详细信息

```
$ls -l
```

我们使用两行信息作为例子，来展示详细信息里面的具体内容

```
drwxr-xr-x  2 liao liao 4096 Nov  5 10:01 Public
-rw-r--r--  1 liao liao   0 Nov  5 10:02 .sudo_as_admin_successful
```

- 第一个字母用来说明文件的类型，这种方法是和Windows使用后缀来代表文件类型不同的。如果第一字母是d，那么说明这是一个目录文件，如果是-，那么说明是普通文件（包括文本文件和二进制文件）。文件类型还有很多种，例如：块特殊文件、字符特殊文件、命名管道（FIFO）、套接字（socket）和符号链接等
- 后续的九字母代表的是权限。依次是代表拥有者的读，写，执行、拥有组的读，写，执行和其他用户的读，写，执行权限。如果文件可读则显示r，可写则显示w，可执行则显示x，没有相关权限则显示-
- 后续的数字代表的是指向这个文件的硬链接数
- 后续的两个字符串依次代表文件拥有者和拥有组
- 随后的数字代表了文件的所占用的空间，这个属性只对普通文件，目录文件和符号链接有意义。普通文件的大小可以是0。目录的大小是固定的，在本机中是4096，它和目录下面的文件大小无关，目录文件主要是存放目录下的文件的文件名。符号链接的大小是它指向的文件的路径名长度

- 接着就是一个时间和日期，说明的是最后修改的时间
- 最后的字符串说明了文件名字，如果是符号链接，它还会显示指向文件的名字

在ls命令后面添加-lh参数，可以将文件详细信息以“人类可读”的方式来显示，也就是文件的占用空间会自动根据大小来按照K, M, G的单位显示。

```
$ls -lh
```

改变当前工作目录

使用cd命令可以改变工作目录

```
$cd /dev
```

- 在dev目录上面使用ls -l指令以后，会发现这个目录下有大量的块特殊文件和字符特殊文件。其中块特殊文件用b来表示，字符特殊文件用c来表示。特殊文件实际上是硬件设备在文件系统上的抽象，所以特殊文件也叫设备文件
- 块特殊文件的特点是每次读取都读取一“块”数据，也就是说会设置一个专门的缓冲区来存放数据，典型的块特殊文件有磁盘。字符特殊文件的特点是每次读取都读取一个字符，典型的有键盘、鼠标、打印机等等

在cd后面添加-，可以回到上次cd之前的目录（连续使用cd - 只能在两个目录之间切换，因为之前目录是存放在环境变量里面，使用env命令可以查看，每次使用cd命令都会修改环境变量）

```
$cd -
```

- 如何使用cd命令，使工作目录转移到父目录，根目录，家目录？

```
$cd ..  
$cd /  
$cd ~ 或者 cd
```

创建目录

使用mkdir可以创建一个目录

```
$mkdir 目录名
```

- 如果直接去/home下面使用mkdir，它会提示没有权限。能否在一个目录下去添加新的文件，依赖于用户对目录的写权限，因为在目录下添加文件，实际上是修改了目录文件里面的内容。
- 目录文件里面实际上是存储一个链表，其中每个链表结点代表一个文件的文件名和磁盘地址信息

删除目录

使用rmdir命令可以删除一个空目录

```
$ rmdir 目录名
```

- 注意：rmdir不能删除非空的目录

拷贝文件或者目录

使用cp命令可以拷贝文件和目录


```
$ cp [选项] 源文件 目标路径|目标文件
```

```
# 当目标是目录的时候，cp命令会直接在目标目录下拷贝一份副本，文件名和源文件一致
```

```
# 当目标是文件的时候，cp命令会拷贝源文件的内容，重命名为目标文件的名字
```

- Ubuntu当中，如果cp命令中目标路径已经存在同名文件，那么就会直接覆盖（其他发行版中在选项中添加-f(force 强制)也可以实现同样的效果)

如果希望在覆盖之间进行警告，则应该在选项中添加-i(interactive 交互式)。比如，如果dir1/text1已经存在，则键入下面的命令：

```
$ cp -i text1 dir1
```

```
# 那么就会出现提示：
```

```
cp: overwrite 'dir1/text1'?
```

```
# 如果输入n或者no就会不覆盖，输入其他则会覆盖
```

如果源文件是一个目录，那么在选项中必须要添加-r，这里r是recursive（循环递归）的意思，也就是说添加-r以后，cp命令会递归地将目录下的所有文件都进行拷贝

```
$ cp -r dir1 path
```

```
# 执行完毕后会出现在dir1的副本path/dir1
```

移动文件或者目录

使用mv命令可以移动文件或者目录

```
$ mv [选项] 源文件 目标路径|目标文件
```

- 执行mv命令的效果和Windows下的剪切-->粘贴的效果一样
- 和cp命令一样，mv命令会出现覆盖现存文件的情况，mv也可以使用-f或者-i选项来关闭或者开启提醒
- Linux里面没有专门的重命名命令，使用mv命令移动到原来的目录下面就可以实现重命名了

删除文件

使用rm命令可以删除文件

```
$ rm [选项] 文件|目录
```

- 使用rm，cp和mv导致文件被覆盖或者删除都是 **不可恢复** 的！
- **不要以root的身份或者是sudo权限来执行 rm -rf，这会造成不可恢复的后果**
- 和rm以及cp命令一样，rm也可以使用-f选项来实现强制删除，使用-i选项来实现删除前交互提示，使用-r选项来实现递归删除目录及目录下文件

树状目录结构显示

首次使用之前需要先安装tree

```
$ sudo apt install tree
```

使用tree命令就可以显示目录的树状结构

```
$ tree 路径名
```

- 如果对tree命令的详细命令不了解，可以在安装完tree以后再使用man命令查看命令帮助，下面是使用tree以后的输出结果

```
.
├── dir1
│   ├── text1
│   └── text3
├── dir2
│   ├── text1
│   └── text2
├── text1
└── text2
```

显然，tree命令可以将目录的树形结构用图形化的方式打印出来

在tree命令之后添加-h选项，可以增加文件大小输出

1.2.3 权限相关

修改权限

- 文件的权限有三种，依次是读、写和执行
- 从命令ls -l的打印结果中可以得知，这种三种权限还会根据用户的身份而有所不同。文件的拥有者、文件拥有者的同组其他成员以及其他用户对文件的权限会有所区别

1. 如何判断某个用户能否读取/修改/执行文件？

先判断当前用户和文件之间的身份关系，然后再检查文件的权限位

2. 如何判断某个用户能否删除文件？

先找到文件所在目录，判断当前用户是否拥有目录的写权限

使用chmod可以修改文件的权限，chmod命令比较灵活，它拥有比较多种使用模式，接下来会依次介绍：

1. 文字设定法

```
$ chmod [who][+|-|=][mode] 文件名
# who代表用户类型：u表示用户，g表示组成员，o表示其他用户，a或者不写代表所有用户（ugo三个字母可以组合）
# +|-|=代表操作符：+表示添加某个权限，-表示取消某个权限，=表示设置成某种权限
# mode代表模式，使用rwx三个字母的组合，r表示可读，w表示可写，x表示可执行
```

可以在一行语句中对某个文件调整多次权限，使用逗号来分隔用户操作类型即可，例如：

```
$ chmod g+r,o+r example
```

执行完成以后，增加了组成员和其他用户的读权限

另外一个例子，删除example文件所有用户的执行权限（三种写法等价）

```
$ chmod a-x example
$ chmod -x example
$ chmod ugo-x example
```

2. 数字设定法

- 通常文件的权限是采用类似于位图的形式来进行存储，3位的2进制数字可以用来表示文件的读写执行权限，9位的2进制数字就可以表示文件的所有用户权限。
- 在3位2进制数中，通常使用最高位表示读权限，随后是写权限，最低位是执行权限。然后再将拥有用户，拥有组和其他用户组合起来，就能得到一个9位2进制数字，最后按照8进制和2进制的转换方法，文件权限就能很轻松地用一个3位8进制数字来表示
- 例如 **0644** 就是 **拥有者可读可写不可执行，组用户和其他用户可读不可写不可执行**

```
$ chmod 644 example
```

执行完成以后就会将权限设置成拥有者可读可写不可执行，组用户和其他用户可读不可写不可执行

```
$ chmod 775 example
```

执行完成以后就会将权限设置成拥有者和组用户可读可写可执行，其他用户可读不可写可执行（工作中经常使用）

掩码

umask指文件（默认是0666）或目录（默认0777）创建时在全部权限中要去掉的一些权限，普通用户缺省时umask的值为002，root用户为022

- 提问：请根据实际创建的文件的权限回答，掩码是怎么运算的？

通过umask查看默认的缺省的掩码值，通过umask 001来修改掩码值

```
$ umask
$ umask 001
```

- 这种操作是临时的，如果要永久修改，需要修改配置文件（例如~/.bashrc）

1.2.4 文件查找

索引

- 在Windows的旧版本当中，想要搜索一个文件是非常缓慢的。为了实现快速查找，在Windows 10和Linux系统当中，系统会专门维护一个索引文件
- 在Linux当中，索引文件是存放在具体的磁盘文件前面的，索引文件里面存放了各个文件的位置信息和其他信息。有关使用索引来实现快速查找的详细算法将会在数据库阶段进行讲解

使用find命令可以根据查找条件来在起始目录之下来查找文件，然后再执行指定的操作

```
$ find 起始目录 查找条件 操作
```

使用find命令可以用来查找某个头文件的位置

```
$ sudo apt install gcc
#先安装gcc编译器
$ find /usr/include -name stdio.h
# 在/usr/include目录下查找文件stdio.h
```

常见的查找条件

根据名称和文件属性查找

- -name '字串' 查找文件名匹配所给字串的所有文件，字串内可用通配符*、?、[]
- -gid n 查找属于ID号为n的用户组的所有文件
- -uid n 查找属于ID号为n的用户的所有文件

```
$ find . -uid 1000
```

- -group '字串' 查找属于用户组名为所给字串的所有的文件
- -user '字串' 查找属于用户名为所给字串的所有的文件

```
$ find . -user liao
# 这两个查找是等价的
```

- -empty 查找大小为0的目录或文件。
- -perm 权限 查找具有指定权限的文件和目录，权限的表示可以如711, 644
- -size n[bckw] 查找指定文件大小的文件，n后面的字符表示单位，缺省为b，代表512字节的块
- -type x 查找类型为x的文件

根据时间查找

- amin n 查找n分钟以前被访问过的所有文件（+表示n分钟之前，-表示n分钟之内，+号和-号都不能省略）
- cmin n 查找n分钟以前文件状态被修改过的所有文件
- mmin n 查找n分钟以前文件内容被修改过的所有文件
- atime n 查找n天以前被访问过的所有文件
- ctime n 查找n天以前文件状态被修改过的所有文件
- mtime n 查找n天以前文件内容被修改过的所有文件
- 其他的查找条件可以在man帮助里面找到

应用通配符

通配符是字符串里面的特殊符号。如果将字符串里面的通配符按照一定的规则用字符来替换以后，得到的字符串和目标字符串一致，那么目标字符串就能匹配

例子，当前目录里面有文件file, file1, file12, file2, file3, file4

```
$ find . -name "file*"
# *用来匹配0至多个任意字符
```

结果是

```
./file1
./file
./file12
./file2
./file3
./file4
```

```
$ find . -name "file?"  
# ?用来匹配1个任意字符
```

结果是

```
./file1  
./file2  
./file3  
./file4
```

```
$ find . -name "file[0-9]"  
# [范围] 用来匹配1个范围内的字符
```

结果是

```
./file1  
./file2  
./file3  
./file4
```

查找条件的运算

查找条件可以进行逻辑运算，比如逻辑与，逻辑或和逻辑非

1. 在命令中用-a表示逻辑与，表示只有当所给的条件都满足时，查找条件才满足。例如在/home/user目录下查找名为0718 类型是一个目录的文件（使用频率最高）：

```
$ find /home/user -name 0718 -a -type d
```

2. 在命令中用-o表示逻辑或，表示只要所给的条件有一个满足，查找条件就满足。例如在/home/user目录下查找名字为main.cc 或名字为 main.c的文件：

```
$ find /home/user -name main.cc -o -name main.c
```

3. 在命令中用! 表示逻辑非，表示查找不满足所给条件的文件。例如在/home/user下查找名字不是main.c的文件：

```
$ find /home/user ! -name main.c
```

查找的权限

必须要拥有某个目录的读权限，才能够进入该目录进行查找

组合操作

在find命令后面添加-exec 新命令 \;就可以对每个查找出来的文件执行新命令

```
$ find /usr/include -name stdio.h -exec ls -l {} \;
```

工作中一般使用xargs，通常可以使用|将find命令的执行结果输出到一个缓冲区（而不是显示屏）里面，这个缓冲区称为管道。然后xargs可以从管道中按行读取内容，并且循环执行命令

```
$ find /usr/include -name stdio.h main.c|xargs ls -l
```

如果对于管道里面的内容，不需要按行循环执行命令，那么就不需要使用xargs，直接使用管道即可

```
$ ls|wc -l  
# 列出所有文件然后统计行数
```

1.2.5 磁盘相关的命令

文件系统的整体磁盘空间使用情况

使用df命令可以列出文件系统的整体磁盘空间使用情况

```
$ df [选项] [文件名]  
$ df -h  
# 显示整个磁盘使用情况
```

显示每个文件和目录的磁盘使用空间

使用du命令可以显示每个文件和目录的磁盘使用空间
显示目录的每一级

```
$ du [选项] [文件名]
```

只显示当前目录

```
$ du -h --max-depth=1 /home/liao
```

1.2.6 文件查看和处理

查看文件内容

使用cat命令可以查看文件内容

```
$cat [选项] [文件]
```

一些选项参数

- -b 对非空输出行编号
- -E 在每行结束处显示\$
- -n 对输出的所有行编号
- -s 不输出多行空行
查看密码文件

```
$cat etc/shadow  
# 如果出现拒绝访问，是什么原因？
```

shadow文件里面存放了盐值以及通过sha512算法得到的密文

文件类型

使用file命令可以查看文件的类型

```
$file a.out
$file test.c
```

标准的输入输出与重定向

文件描述符是一个整数，它代表一个打开的文件

标准的三个描述符号：

- 1. 标准输入：一般指键盘，描述符为：0
 - 2. 标准输出：一般指屏幕输出，描述符为：1
 - 3. 错误输出：也是屏幕，描述符为：2
- 重定向符号包括重定向输入<，重定向输出>，添加输出>>，错误重定向2>，错误和信息重定向&>

- 可以尝试一下同时将信息打印到stdout和stderr，然后使用重定向符号进行调试
- 注意程序的信息要么只能重定向到stdout，要么只能重定向到stderr，不能分开重定向

cat常常与重定向一起使用。其中>表示创建，>>表示追加，<<表示以什么结束，如果cat的命令中没有参数，它就会从标准输入中读取数据，并将其送到标准输出

创建文件

linux中创建空文件的四种方式：

```
# 方式1, 使用echo:
$ echo > a.txt (会有一个字节)
$ echo -n > a.txt
# 方式2, 使用touch:
$ touch b.txt
# 方式3, 使用cat:
$ cat > c.txt
$ 按ctrl+c或ctrl+d退出
# 方式4: vi d.txt
$ 进入之后按: wq退出
```

部分显示

使用head命令可以显示文件的前面几行

```
$ head [-n 行数值] [文件名]
```

使用tail命令可以显示文件的后面几行

```
$ tail [-n 行数值] [文件名]
```

提问：怎么样将历史记录的后10行存入文件latestHistory.txt里面？

使用C语言怎么做？打开~/bash_history文件，然后记录最后10行，然后再输出

使用shell怎么做？history|tail -n 10> latestHistory.txt

由此可见，shell程序的编写效率要远远高于C语言，这也是shell出现的目的

使用more或者less命令来单页浏览文件

```
$ more 文件名
# 进入以后使用f和b向前向后翻页，使用q退出浏览
```

其他操作

使用sort命令可以对文件内容进行排序

```
$ sort 文件名
```

- 提问：怎么将结果进行保存？

使用file命令查看文件内容类型

```
$ file 文件名  
# 根据文件内容，判别文件类型：比如可执行文件、文本等等
```

使用uniq命令报告或删除文件中重复的行（只能去除相邻相同的）

```
$ uniq 文件名
```

- -c 在输出行前面加上每行在输入文件中出现的次数。
- -d 仅显示重复行。
- -u 仅显示不重复的行。

使用wc命令统计指定文件中的字节数、字数、行数

```
$ wc 文件名
```

- -c 统计字节数
- -l 统计行数
- 提问：统计文件数

```
# 统计当前目录下的文件  
$ ls|wc -l  
# 统计当前目录以及所有的子目录下的文件  
$ find . -name *|wc -l  
# -m 统计字符数。这个标志不能与 -c 标志一起使用。  
# -w 统计字数。一个字被定义为由空白、跳格或换行字符分隔的字符串
```

使用iconv命令汉字编码转换

```
$ iconv [-f 原编码方式] [-t 新的编码方式] 文件命令
```

Windows默认是采用GBK (gb2312) 编码，在Windows里面保存一个内容为中文“烫”的文件（它的二进制形式是0xcccc），再将其放入Linux当中，如果直接使用cat打印是不会显示内容的

使用下列的iconv命令转换语句以后，将结果重定向到新文件里面，再使用cat打印新文件就能显示正确的文本信息

```
$ iconv -f gb2312 -t utf-8 hanzi.txt>hanzi1  
# 不能重定向到本文件
```

utf-8编码使用最多3个字节来表示符号，它可以表示世界上所有语言的所有符号，而GB2312是中国的标准，它只能使用最多2个字节来表示汉字
在vim编辑器里面，输入:%!xxd可以令文本按16进制显示

1.2.7 文件内容查找

搜索文件内容

使用grep命令可以查找文件内容

```
$ grep [选项][查找模式][文件名1, 文件名2, ...]
```

- grep过滤器查找指定字符模式的文件，并显示含有此模式的所有行，符合规则的所有模式会显示为红色。被寻找的模式称为正则表达式
- 选项中使用-E，可以使用拓展正则表达式规则，否则默认采用基本规则
- 其他参数：

```
-F 每个模式作为固定的字符串对待  
-c 只显示匹配行的数量  
-i 比较式不区分大小写  
-n 在输出前加上匹配串所在的行号
```

正则表达式

和find命令使用通配符有所区别，grep使用正则表达式的方式来进行模式匹配。正则表达式是利用通配符（特殊字符）来定义了规则，从而和文本进行匹配。grep命令的目标就是找到符合规则的字符串集合

正则表达式语法：

1. 单个字符是匹配的基本单位，比如a就能够匹配含有字母的字符串集合。几乎所有的字母和数字都能够匹配它们本身
2. 反斜杠+某些字符构成转义字符，它们作为整体可以看成是一个特殊的单个字符
3. 大部分的通配符规则：

```
a|b      匹配 a 或 b 使用-E  
gr(a|e)y 匹配 gray 或 grey  
.  
[abc]    匹配任一字符:a或b或c  
[^abc]   匹配任一字符，但不包括a,b,c  
[a-z]    匹配从 a 到 z 之间的任一字符  
[a-zA-Z] 匹配从a到z，及从A到Z之间的任一字符  
^表达式 匹配字符串的头部 使用-E  
表达式$ 匹配字符串的尾部 使用-E  
( )      匹配标记的子表达式 使用-E  
\b       匹配字词边界  
\n       匹配第n个标记的子表达式，n代表1到9  
*        匹配前一项内容0或多次  
?        匹配前一项内容0或1次 使用-E  
+        匹配前一项内容1或多次 使用-E  
{x}     匹配前一项内容x次  
{x,}   匹配前一项内容最少x次  
{,x}   匹配前一项内容最多x次  
{x,y}  匹配前一项内容次数介于x和y之间 使用-E  
\       转义字符
```

grep应用

grep的运行原理是打开文件，然后按行来读取文件内容，如果本行里面拥有匹配模板的字符串，那么就染色这个字符串，并打印输出

- 在C程序文件里面找到int开始的语句

```
$ grep -n ^int main.c
```

- 在C程序文件里面找到;结尾的语句

```
$ grep -n \;$ main.c
$ grep -n ";$" main.c
```

- 提问：为什么不能使用grep -n ;\$ main.c
- 提问：怎么样匹配任意多个字符？
- 怎么样查找特殊字符？

```
$ grep -nF ^ main.c
```

在C程序文件里面找到main函数

```
$ find . -name "*.c"|xargs grep -n "main"
```

1.2.8 其他命令相关

查看命令的别名

使用alias命令可以打印当前所使用的别名情况。一旦给命令起了别名，那么输入命令的别名就可以起到和原来命令一样的效果。

```
$ alias
```

下面的例子是使用alias命令以后，终端打印的部分信息：

```
alias l='ls -CF'
alias la='ls -A'
alias ll='ls -aLF'
alias ls='ls --color=auto'
```

- 根据别名信息可以得知，当我们键入l的时候，等价于输入ls -CF（目录名后增加/符号显示）；当我们键入ll，等价于键入ls -aLF

输入命令的历史记录

使用history命令可以获取输入命令的历史记录，在history命令后面再添加> 文本文件，就可以将历史记录保存到文本文件（本例子中是myHistory.txt）里面

```
$ history > myHistory.txt
```

查看命令的帮助

Linux系统中会内置帮助手册，帮助手册有8个部分，包括用户手册、程序员手册和系统管理员手册等等。通常通过关键字+数字的形式来说明是第几个手册的帮助信息。例如，ls(1)就说明是第一部分（用户手册，即shell命令手册）的ls命令的信息

使用man命令可以查看帮助信息

```
$ man [手册编号] 命令名字
```

- 进入帮助界面以后，使用q可以退出界面，使用方向键可以向前向后查看，使用b和f可以前后翻页，使用/+字符串可以进行查找

一次输入以后执行多条命令

输入完命令以后不键入回车键，而是键入分号，然后再输入下一条命令，就可以实现按顺序依次执行多条命令的效果。例如先创建目录，再展示所有文件

```
$ mkdir dir1;ls -l  
# 第二条的ls命令的结果当中就会出现dir1的信息
```

管道和命令替换

管道是重定向的一种，就像一个导管一样，将一个程序或命令的输出作为另一个程序或命令的输入

```
$ ls -l /etc|wc -w
```

- 提问：这个命令的结果和ls /etc|wc -w的结果有什么区别？

命令替换和重定向有点相似，但区别在于命令替换是将一个命令的输出作为另一个命令的参数

常用的格式为

```
$ command1 `command2`  
# command1 是任意命令  
# 这里不是单引号，是键盘1左边的按键  
$ command1 $(command2)
```

首先列出当前的所有信息，并重定向到aa文件中：

```
$ ls|cat>aa  
#或  
$ ls>aa
```

然后，通过命令替换，列出aa文件中所有的文件信息

```
$ ls -l `cat aa`  
# 或者用  
$ ls -l $(cat aa)
```

打包和压缩

Linux当中是可以把一大堆的文件和目录全部打包成一个文件，这对于备份文件或将几个文件组合成为一个文件以便于网络传输是非常有用的。使用tar命令大小可以打包，可以将文件加入到某个包文件或者是将文件移除出某个包文件

```
$ tar [主选项+辅选项] 目标文档 源文件或目录
```

- 参数说明

```
# c: 创建新的包
$ tar cfv packet.tar file*
```

包文件通常是比较庞大的，具体原因是包文件创建默认分配了较大的磁盘空间，所以需要压缩来解决磁盘空间

```
# r: 要把存档的文件追加到包文件的末尾。
$ tar rfv packet.tar test
# x: 从包文件中释放文件
$ tar xfv packet.tar
# f: 生成使用包文件或设备。
# v: 在归档过程中显示处理的文件。
# z: 用gzip来压缩/解压缩文件，后缀名为.gz，加上该选项后可以将档案文件进行压缩。
$ tar czvf packet.tar.gz file* *.txt
```

- 这时候检查压缩文件的大小，它的大小实际上比原文件还要小

解压释放文件的命令

```
$ tar xzvf packet.tar.gz
```

- 一种典型的压缩方法是哈夫曼编码树的方法

使用gzip/bzip2命令来文件压缩与解压，gzip用来将文件压缩成后缀为.gz的压缩文件，或者将后缀为.gz的文件进行解压。Bzip2用来将文件压缩成后缀名为.bz2的压缩文件，或者将后缀为.bz2的压缩文件解压

```
$ gzip/bzip2 [选项] [压缩或解压缩的文件名]
```

常用参数:

- -d: 将压缩文件进行解压。
- -v: 在压缩或解压过程中显示解压或压缩的文件。
例如，将main.c 进行压缩，则使用

```
$ gzip/bzip2 -v main.c。
```

- 会将main.c 压缩成 main.c.gz 或者 main.c.bz2
如果需要将刚才的压缩文件解压，则使用

```
$ gzip -dv main.c.gz,或者 bzip2 -dv main.c.bz2
```

远程拷贝

远程拷贝之前，需要先ping通目标服务器

```
$ ping 112.124.31.45
```

使用scp命令从本机拷贝到远程服务器

```
$ scp /home/liao/pre/file1 liaozs@112.124.31.45:~/
```

如果是首次连接，还需要使用一个授权（输入yes即可）。然后输入密码

使用scp命令从远程服务器拷贝到本机

```
$ scp liaozs@112.124.31.45:~/file1 ~
```

使用脚本自动将文件在本地和服务器之间交换

上传文件:

```
scp -r $1 [用户名]@[目标IP]:[路径] `date -d "0 week" +%Y%m%d`/code
```

\$1是命令行的参数，类似C语言的argv[1]

下载文件:

```
scp -r [用户名]@[目标IP]:[路径] `date -d "0 week" +%Y%m%d`/code .
```

使用的时候将上述命令保存到文件，添加执行权限，使用sudo放在/bin下面

无密钥登录可以实现不使用密码登录

每个用户的密钥存放在~/.ssh/里面，其中id_rsa是私钥，id_rsa.pub是公钥

使用ssh-keygen来生成钥匙（一直回车即可），让后会生成一对钥匙（这里是采用了非对称加密的算法，有兴趣的同学可以课后自行查阅相关知识），然后将公钥的内容拷贝到目标机器的.ssh/authorized_keys文件里面（如果没有必需要自己创建）

1. 使用scp命令拷贝目标主机
 2. 登陆到目标主机（ssh 用户名@ip地址）
 3. 使用cat打开公钥，然后使用>>追加到授权文件里面
- 注意使用无密钥登录是单向的，并且本机的私钥不能移动位置或者修改

使用ssh命令可以远程执行命令

```
$ ssh 用户名@ip地址 "命令"
```

1.3 杂项

WinScp软件的使用

工作当中经常需要在服务器和本地之间交换文件，使用WinScp软件可以高效地在不同的设备当中交换文件

打开WinScp软件，然后登录服务器。然后在本地和服务器之间的文件系统当中拖动文件就可以便捷地实现文件上传和下载

Mac系统转移文件

1. 在Mac里面开启sshd服务

```
$ sudo launchctl load -w /System/Library/LauchDaemons/ssh.plist
```

2. 在Ubuntu里面使用scp命令复制文件

```
$ scp 本地文件路径 [Mac用户名]@[mac的ip例如192.168.6.58]:~
```

xshell软件设置调整

修改xshell设置实现选中复制，右击粘贴

- 进入工具-->选项-->键盘和鼠标，勾选将选中的文本自动复制到剪切板，将鼠标中键的功能修改为打开弹出式菜单，将右键的功能设置为粘贴剪切板内容
修改xshell设置，实现退格键正常退格
- 进入文件-->打开，右键点击当前会话，进入属性-->终端-->键盘，将delete和backspace都修改为ASCII，然后重启会话。

命令输入的常用快捷键

- **tab** 输入预测，根据前缀自动补全路径名
- **ctrl+a** 光标回到命令输入区开始位置
- **ctrl+e** 光标回到命令输入区结束位置
- **ctrl+r** 搜索之前出现过的命令
- **方向键上** 键入上一条命令

安装帮助手册

Ubuntu系统中的帮助手册默认只有1部分（命令帮助），其他部分需要自行安装
安装POSIX帮助

```
$ sudo apt install manpages-posix-dev
```

安装完成以后，使用man命令就可以查看POSIX接口和ISO C接口的帮助了

网络相关的命令

当网络不通时，通过执行route命令查看路由，查看网关配置是否正常

```
$ route
```