LINUX终端命令

1. 常用linux命令的基本使用

|  |  |  |
| --- | --- | --- |
| 命令 | 对应英文 | 作用 |
| ls | list | 查看当前文件夹下的内容 |
| pwd | Print work directory | 打印当前所在文件夹 |
| cd [目录名] | Change directory | 切换文件夹 |
| touch [文件名] | Touch | 如果文件不存在，新建文件 |
| mkdir[目录名] | Make directory | 创建目录 |
| rm[文件名] | Remove | 删除指定的文件名 |
| clear | clear | 清屏 |
|  |  |  |
|  |  |  |

放大终端窗口的字体：ctrl+shift+=

缩小终端字体：ctrl+-

2、终端命令格式

command [-options] [parameter]

说明 ：

Command:命令名，对应功能的英文单词或者单词的缩写

[-options]：选项，可用来对命令进行控制，也可以省略

[parameter] 传给命令的参数，可以是零个、一个或者多个、

[] 代表可选

3、查阅命令帮助信息

只需要先知道以下两种方式可以查询命令的帮助信息

1. --help

Eg: command --help

显示command命令的帮助信息

1. man

eg: man command

man 是manual的缩写,是linux提供的一个手册，包含绝大部分的命令、函数的详细使用说明

使用man时的操作键：

|  |  |
| --- | --- |
| 操作键 | 功能 |
| 空格键 | 显示手册页的下一屏 |
| Enter键 | 一次滚动手册页的一行 |
| b | 回滚一屏 |
| f | 前滚一屏 |
| /word | 搜索word字符串 |
| q | 退出 |

4、终端实用技巧

1）自动补全

在敲出 文件/目录/命令 的前几个字母后，按下tab键

如果输入的没有歧义，系统会自动补全；

如果还存在其他 文件/目录/命令 再按下 tab键会系统提示可能存在的命令。

2）曾经使用过的命令

按 上/下 光标键可以在曾经使用过的命令之间来回切换；

如果想退出选择，并且不想执行当前选中的命令。可以按ctrl+c.

5、ls命令说明

ls功能为列出目录的内容。

Linux下文件和目录的特点:

1. Linux文件或目录名称最长有256个字符;
2. 以 . 开头的文件为隐藏文件，需要用-a参数才能显示；
3. . 代表当前目录
4. .. 代表上一级目录

使用cd .. 返回上一级目录。如果创建的文件想要隐藏，则需要让文件名以. 开头。

ls常用选项

|  |  |
| --- | --- |
| 参数 | 含义 |
| -a | 显示指定目录下所有子目录与文件，包括隐藏文件 |
| -l | 以列表方式显示文件的详细信息 |
| -h | 配合-l以人性化的方式显示文件大小 |

6、通配符的使用

|  |  |
| --- | --- |
| 通配符 | 含义 |
| \* | 代表任意个数个字符 |
| ? | 代表任意一个字符，至少一个 |
| [] | 表示匹配字符组中的任意一个 |
| [abc] | 匹配a、b、c中的任意一个 |
| [a-f] | 匹配从a到f范围中的任意一个 |

7、cd 的功能为更改当前的工作目录

\*\*\*linux所有的目录和文件名都是大小写敏感的

中间要有空格

|  |  |
| --- | --- |
| 命令 | 含义 |
| cd | 切换到当前用户的主目录(/home/用户目录) |
| cd ~ | 切换到当前用户的主目录(/home/用户目录) |
| cd . | 保持在当前目录不变 |
| cd .. | 切换到上级目录 |
| cd - | 可以在最近两次工作目录之间来回切换 |

8、 相对路径和绝对路径

相对路径：在输入路径时，最前面不是/或者~，表示相对当前目录 所在的目录位置。

绝对路径： 在输入路径时，最前面是/或者~，表示从根目录/家目录开始的具体目录位置。

9、touch 创建文件或者修改文件时间

如果文件不存在，可以创建一个空白文件；

如果文件已经存在，可以修改文件的末次修改时间  
10、mkdir 创建一个新的目录

|  |  |
| --- | --- |
| 选项 | 含义 |
| -p | 可以递归创建目录 |

新建目录的名称 不能与当前目录中已有的目录或文件重名

11、rm

删除文件或目录

文件删除后不能恢复

|  |  |
| --- | --- |
| 参数 | 含义 |
| -f | 强制删除，忽略不存在的文件，无需提示 |
| -r | 递归地删除目录下的内容，删除文件夹时必须加此参数 |

12、拷贝和移动文件

|  |  |  |
| --- | --- | --- |
| 命令 | 对应英文 | 作用 |
| tree [目录名] | tree | 以树状图列出文件目录结构 |
| cp 源文件 目标文件 | copy | 复制文件或目录 |
| mv源文件 目标文件 | move | 移动文件或者目录/文件或者目录重命名 |

13、tree

|  |  |
| --- | --- |
| 选项 | 含义 |
| -d | 只显示目录 |

14、cp

Cp命令的功能是将给出的文件或目录复制到另外一个文件或目录中

|  |  |
| --- | --- |
| 选项 | 含义 |
| -f | 已经存在的目标文件直接覆盖，不会提示 |
| -i | 覆盖文件前提示 |
| -r | 若给出的源文件为目录文件，则cp.将递归复制该目录下的所有子目录和文件，目录文件必须是一个目录名 |

15、mv 可以用来移动文件或目录，也可以给文件或目录重命名

|  |  |
| --- | --- |
| 选项 | 含义 |
| -i | 覆盖文件前提示 |

Cp 和mv最好加-i

16、查看文件内容

|  |  |  |
| --- | --- | --- |
| 命令 | 对应英文 | 作用 |
| cat文件名 | concatenate | 查看文件内容、创建文件、文件合并、追加文件内容等功能 |
| more文件名 | more | 分屏显示文件内容 |
| grep搜索文本 文件名 | grep | 搜索文本文件内容 |

17、cat

Cat 命令可以用来 查看文件内容、创建文件、文件合并、追加文件内容等功能。

Cat 会一次显示所有的内容，适合查看内容较少的文本文件

|  |  |
| --- | --- |
| 选项 | 含义 |
| -b | 对非空输出行编号 |
| -n | 对输出的所有行编号 |

Linux中还有一个nl的命令和cat –b的效果等价

18、more

More命令可以用于分屏显示文件内容，每次只显示一页内容

适合于查看文件内容较多的文本文件

使用more的操作键

|  |  |
| --- | --- |
| 操作键 | 功能 |
| 空格键 | 显示手册页的下一屏 |
| Enter键 | 一次滚动手册页的一行 |
| b | 回滚一屏 |
| f | 前滚一屏 |
| /word | 搜索word字符串 |
| q | 退出 |

19、grep

Grep命令是一个强大的文本搜索工具。

Grep允许对文本文件进行模式查找，所谓模式查找，也被称为正则表达式。

|  |  |
| --- | --- |
| 选项 | 含义 |
| -n | 显示匹配行及行号 |
| -v | 显示不包含匹配文本的所有行（相当于求反） |
| -i | 忽略大小写 |

常用的两种模式查找

|  |  |
| --- | --- |
| 参数 | 含义 |
| ^a | 行首，搜索以a开头的行 |
| Ke$ | 行尾,搜索以ke结束的行 |

20、其他

1)echo文字内容

Eg: echo hello

echo会在终端中显示参数指定的文字，通常会和重定向联合使用

1. 重定向> 和>>

Linux允许将命令执行结果重定向到一个文件；

将本应显示在终端上的内容 输出/追加 到指定文件中

>表示输出，会覆盖文件原有的内容

>>表示追加，会将内容追加到已有文件的末尾。

Eg: echo hello python > a.txt

1. 管道 |

Linux允许将一个命令的输出通过管道，作为另一个命令的输入。

这里的 | 的左右分为两端，左端塞东西（写），右端取东西(读).

常用的管道命令有：

More:分屏显示内容；

Grep:在命令执行结果的基础上查询指定的文本。

Eg : ls –lha ~|grep Do

21、远程管理常用命令

1）关机/重启

|  |  |  |
| --- | --- | --- |
| 命令 | 对应英文 | 作用 |
| shutdown 选项 时间 | shutdown | 关机/重新启动 |

Shutdown可以安全关闭或这重新启动系统

|  |  |
| --- | --- |
| 选项 | 含义 |
| -r | 重新启动 |

注意点：

1. 不指定选项和参数时，默认表示一分钟之后关闭电脑
2. 远程维护服务器时，最好不要关闭系统，而应该重新启动系统

常用命令示例：

# 重新启动操作系统，其中 now表示现在

$ shutdown –r now

# 立刻关机，其中 now 表示现在

$ shutdown now

# 系统会在今天的20：25关机

$ shutdown 20:25

#系统再过十分钟后自动关机

$ shutdown +10

#取消之前指定的关机计划

$ shutdown -c

2）查看或配置网卡信息

|  |  |  |
| --- | --- | --- |
| 命令 | 对应英文 | 作用 |
| ifconfig | Configure a network interface | 查看/配置计算机当前的网卡配置信息 |
| ping ip 地址 | ping | 检测到目标ip地址的连接是否正常 |

网卡是一个专门负责网络通讯的硬件设备，IP地址是设置在网卡上的地址信息。

通过命令 ip addr也可以查看ip地址

<1>查看网卡配置信息

$ ip addr

<2>查看网卡对应的ip地址

$ ip addr|grep inet 通过管道筛选

Ping

# 检测到目标主机是否连接正常

$ ping IP地址

# 检测本地网卡工作正常

$ ping 127.0.0.1

Ping一般用于检测当前计算机到目标计算机之间的网络是否通畅，数值越大，速度越慢。

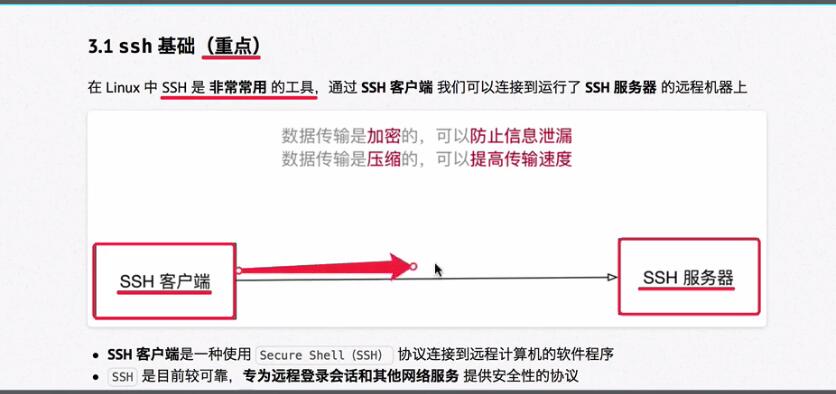
Ping的工作原理类似于潜水艇的声纳，网络管理员之间也常将ping用作动词，ping一下计算机X,看他是否开着。

原理：网络上的机器都有唯一确定的ip地址，我们给目标ip地址发送一个数据包，对方就要返回一个数据包，根据返回的数据包以及时间，我们可以确定目标主机的存在。

在linux中，想要终止一个终端程序的执行，绝大多数都是使用ctrl +c.

1. 远程登陆和复制文件

在linux中ssh是非常常用的工具，通过ssh客户端，我们可以连接到运行了ssh服务器的远程机器上。



ssh客户端是一种使用secure shell(ssh)协议连接到远程计算机的软件程序

ssh是目前较可靠，专为远程登陆会话和其他网络服务提供安全性的协议

<1>域名

由一串 用点分隔 的名字组成 例如: [www.itcast.cn](http://www.itcast.cn)

是 ip地址的别名，方便用户记忆。

<2>端口号

IP地址：通过ip地址找到网络上的计算机；

端口号：通过端口号找到计算机上运行的应用程序。

常用服务端口号列表：

|  |  |
| --- | --- |
| 服务 | 端口号 |
| Ssh服务器 | 22 |
| web服务器 | 80 |
| Https | 443 |
| FTP服务器 | 21 |

Ssh客户端的简单操作

ssh [-p port] user@remote

user是在远程机器上的用户名，如果不指定的话默认是当前用户；

remote为远程机器的地址，可以是ip/域名，或者是别名

port 是ssh server监听的端口，如果不指定的话，默认为22.

使用exit退出当前用户的登录

Ssh这个终端命令只能在linux或者unix系统上使用;

如果在windows系统中，可以安装putty或者xshell客户端软件即可。

1. scp(掌握)

scp就是secure copy,是一个在linux下用来进行远程拷贝文件的命令。

它的地址格式与ssh基本相同，需要注意的是，在指定端口时用的是大写的-P而不是小写的



|  |  |
| --- | --- |
| 选项 | 含义 |
| -r | 若给出的源文件为目录文件，则scp将递归复制该目录下的所有子目录和文件，目标文件必须为一个目录名 |
| -P | 若远程ssh服务器的端口不是22,需要使用大写字母-P选项指定端口 |

scp这个终端命令只能在linux或者unix系统下使用

如果在windows系统中，可以安装PuTTY，使用pscp命令行工具或者安装FileZilla使用FTP进行文件传输。FileZilla在传输文件时，使用的是FTP服务而不是SSH服务,因此端口号应该设置为21.

1. SSH高级(知道)

免密码登陆

配置别名

22、用户和权限的基本概念

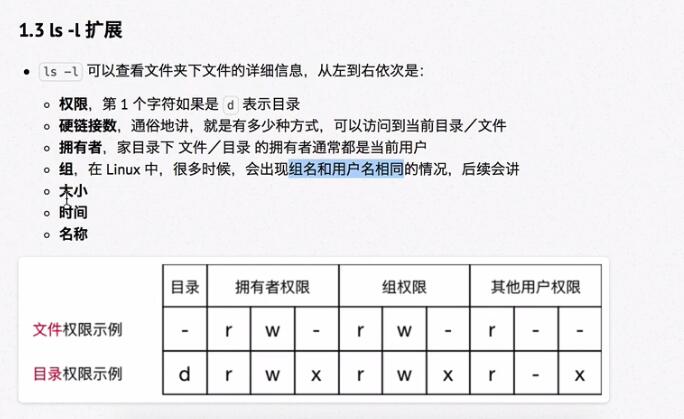
在linux系统中，不论是由主机还是远程登陆系统，每个系统都必须拥有一个账号，并且对于不同的系统资源拥有不同的使用权限

在linux中，可以指定每一个用户针对不同的文件或者目录的不同权限，

对文件/目录的权限包括：

|  |  |  |  |
| --- | --- | --- | --- |
| 权限 | 英文 | 缩写 | 数字代码 |
| 读 | read | r | 4 |
| 写 | write | w | 2 |
| 执行 | excute | x | 1 |

在实际应用中，可以预先针对**组**设置好权限，然后将不同的用户添加到对应的组中，从而不用一次为每个用户设置权限。



23、**chmod简单使用(重要)**

Chmod可以修改 用户/组 对文件/目录的权限。

命令格式如下：

chmod +/-rwx 文件名|目录名

+：增加

24、超级用户

Linux系统中的**root 账**号通常用于系统的维护和管理，对操作系统的**所有资源具有所有访问权限。**

在大多数版本的linux中，不推荐直接使用root账号登陆系统。

在linux安装过程中，系统会自动创建一个用户账号，而这个默认的用户就被称为“标准用户”。

sudo

su是substitute user的缩写，表示 使用另一个用户的身份；

sudo命令用来以其他身份来执行命令，预设的身份为root

用户使用sudo时，必须先输入密码，之后有5分钟的有效期限，超过期限则必须重新输入密码。

若其未经授权的用户企图使用sudo，则会发出警告邮件给管理员。

25、组管理 终端命令

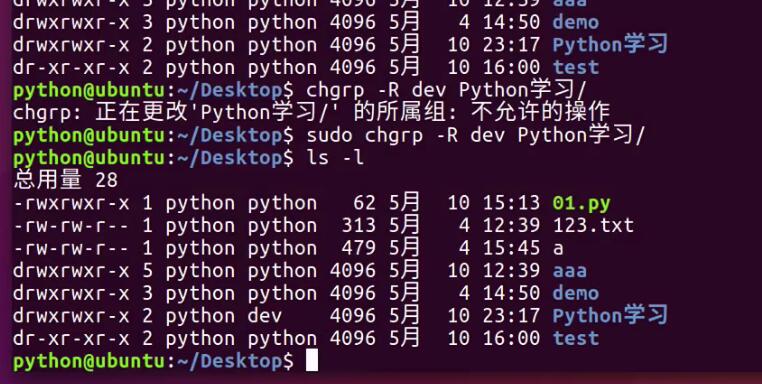
提示：创建组/删除组 的终端命令都需要通过 sudo执行

|  |  |
| --- | --- |
| 命令 | 作用 |
| groupadd 组名 | 添加组 |
| groupdel 组名 | 删除组 |
| cat /etc/group | 确定组信息（cat 后边有空格） |
| chgrp **-R**组名 文件/目录名 | 修改文件/目录的所属组 |

组信息保存在 /etc/group 文件中

/etc 目录是专门用来保存 **系统配置信息** 的目录。

在实际应用中，可以预先针对 组 设置好权限 ，然后将不同的用户添加到对应的组中，从而不用依次为每个用户设置权限。



26、用户管理终端命令

提示：创建用户/删除用户/修改其他用户密码的终端命令都需要通过sudo执行

|  |  |  |
| --- | --- | --- |
| 命令 | 作用 | 说明 |
| useradd –m –g 组 新建用户名 | 添加新用户 | -m 自动建立用户家目录  -g 指定用户所在的组，否则会创建一个和同名的组 |
| **passwd** 用户名 | 设置用户密码 | 如果是普通用户,直接用password,可以修改自己的账号密码 |
| userdel **–r** 用户名 | 删除用户 | -r 选项会自动删除用户家目录 |
| cat /etc/passwd|grep 用户名 | 确定用户信息 | 新建用户后吗，用户信息会保存在 /etc/passwd文件中 |

Eg:

sudo useradd –m bc

sudo passwd bc

sudo usermod –s /bin/bash bc（指定shell，否则会非常不方便终端操作）

提示：

创建用户时，如果忘记添加-m选项指定新用户的家目录 –最简单的方法是删除用户，重新创建

创建用户时，默认会创建一个和用户名同名的组名

用户信息保存在 /etc/passwd 文件中

27、查看用户信息

|  |  |
| --- | --- |
| 命令 | 作用 |
| id [用户名] | 查看用户UID和GID信息（u为用户，g为组） |
| who | 查看当前所有登录的用户列表 |
| whoami | 查看当前登录用户的账号名 |

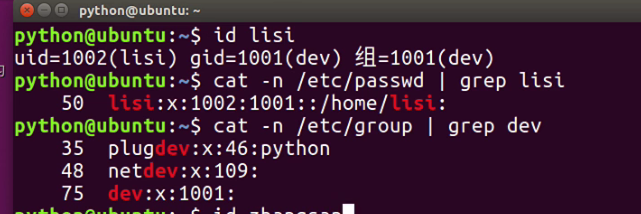
whoami：巧记：我是谁

Passwd文件:

/etc/passwd 文件存放的是用户信息。由6个分号组成的7个信息，分别是

1. 用户名
2. 密码（x 表示加密的密码）
3. Uid(用户标识)
4. GID（组标识）
5. 用户全名或本地账号
6. 家目录
7. 登录使用的shell，就是登录之后，使用的终端命令，**ubuntu默认的是dash**

id查询用户信息



28、usermod

Usermod 可以用来设置 用户的主组 / 附加组 和登录 shell,命令格式如下：

主组：通常在新建用户时指定，在etc/passwd的第四列GID对应的组；

附加组：在etc/group中最后一列表示该组的用户列表，用来指定用户的附加权限。

注意**：设置了用户的附加组之后，需要重新登录才能生效**。

# 修改用户的主组（passwd 中的GID） **小写g**

usermod –g 组 用户名

# 修改用户的附加组 **大写G**

usermod –G 组 用户名

# 修改用户登录 shell

**usermod –s /bin/bash**

**注意：默认使用useradd 添加的用户是没有权限使用sudo以root身份执行命令的，可以使用以下命令，将用户添加到sudo附加组中**

**usermod – G sudo 用户名**

29、which(重要)

提示：

/etc/passwd 是用来保存用户信息的文件（不能执行）

/usr/bin/passwd 是用来修改用户密码的程序（可以执行）。

Which 命令可以查看执行命令所在位置。比如 ：

which ls

# 输出

#/bin/ls

which useradd

# 输出

# /usr/sbin/useradd

**bin 和sbin**

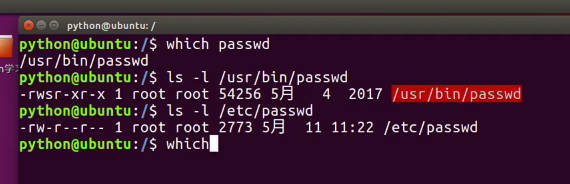
在linux中，绝大多数可执行文件都是保存在 /bin、/sbin、/usr/bin、/usr/sbin。

/bin(binary)是二进制执行文件目录，主要用于具体应用；

/**s**bin(system binary)是系统管理员专用的二进制代码存放目录，主要用于**系统**管理；

/usr/bin (user commands for applications)后期安装的一些软件；

/usr/sbin (super user commands for applications)超级用户的一些管理程序。



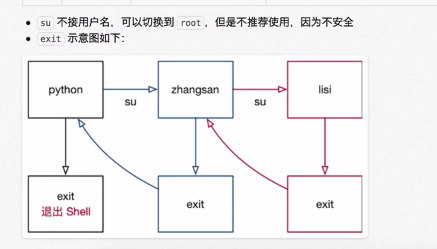
30、切换用户

|  |  |  |
| --- | --- | --- |
| 命令 | 作用 | 说明 |
| su –用户名 | 切换用户，并且切换目录 | -可以切换到用户家目录。否则保持位置不变 |
| exit | 退出当前登录账户 |  |

**su 不接用户名，可以切换到root，但是不推荐使用，因为不安全；**

exit示意图如下

exit是返回上一级目录。



31、修改文件权限

|  |  |
| --- | --- |
| 命令 | 作用 |
| chown | 修改拥有人 |
| chgrp | 修改组 |
| chmod | 修改权限 |

命令格式如下：

# 修改文件|目录的拥有人

chown 用户名 文件名| 目录名

# **递归修改**文件|目录 的组

chgrp **–R** 组名 文件名|目录名

# 递归修改文件权限

chmod **–R** 755 文件名目录名

chmod 在设置权限时，可以简单地使用三个数字分别对应**拥有者/组和其他** 用户的权限。

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 拥有者 | | | 组 | | | 其他 | | |
| r | w | x | r | w | x | r | w | x |
| 4 | 2 | 1 | 4 | 2 | 1 | 4 | 2 | 1 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 4 | 2 | 1 |  |  | 7 | rwx |
| 4 | 2 | 0 |  |  | 6 | rw- |
|  | … |  |  |  |  |  |
|  | … |  |  |  |  |  |
|  | … |  |  |  |  |  |
| 0 | 0 | 0 |  |  | 0 | --- |

常用数字组合有（u表示用户/g表示组/o 表示其他）

777=🡺u=rwx,g=rwx,o=rwx

755🡺u=rwx,g=rx,o=rx

644=>u=rw,g=r,o=r

采用上边的方式，7表示对拥有者操作rwx，5表示对组操作rx，5表示对其他操作rx。

# **直接修改文件|目录的 读|写|执行 权限 ，但是不能精确到拥有者|组|其他**

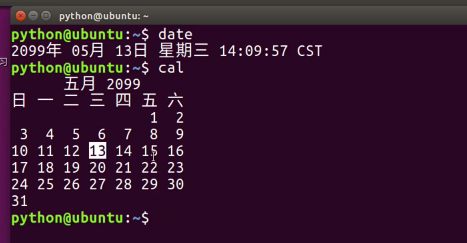
chmod +/- rwx 文件名|目录名

32、系统信息相关命令

方便通过远程终端维护服务器时，查看服务器上当前系统日期和时间/磁盘空间占用情况/程序执行情况

33、日期和时间

|  |  |
| --- | --- |
| 命令 | 作用 |
| date | 查看系统时间 |
| cal | Calendar查看日历，-y选项可以查看一年的日历 |

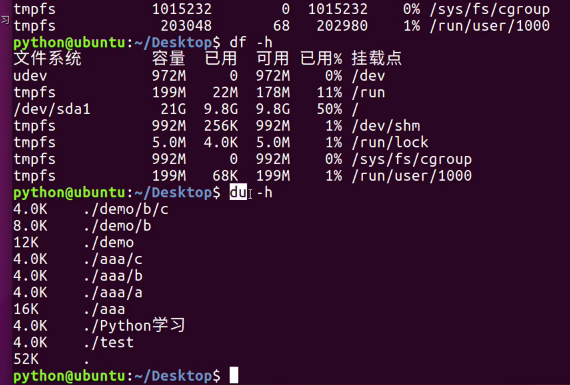


34、磁盘信息

|  |  |
| --- | --- |
| 命令 | 作用 |
| df -h | Disk free 显示磁盘剩余空间 |
| du –h[目录名] | Disk usage 显示目录下的文件大小 |

选项说明：

|  |  |
| --- | --- |
| 参数 | 含义 |
| -h | 以人性化的方式显示文件大小 |



35、进程信息

所谓**进程**，通俗讲就是当前正在执行的一个程序。

|  |  |
| --- | --- |
| 命令 | 作用 |
| ps aux | Process status 查看进程的详细状况 |
| top | 动态显示运行中的进程并且排序 |
| kill [-9] 进程代号 | 终止指定代号的进程，-9表示强行终止 |

**ps：默认只会显示当前用户通过终端启动的应用程序**

**ps的选项不用加-号**

ps选项说明功能：

|  |  |
| --- | --- |
| 选项 | 含义 |
| a | 显示终端上的所有进程，包括其他用户的进程 |
| u | 显示进程的详细状态 |
| x | 显示没有控制终端的进程 |

提示**：使用kill命令时，最好只终止由当前用户开启的进程，而不是终止以root身份开启的进程，否则可能导致系统崩溃**。

**要退出 top可以直接输入q**。

36、其他命令

查找命令 find

软链接 ln

打包和压缩 tar

软件安装 apt-get

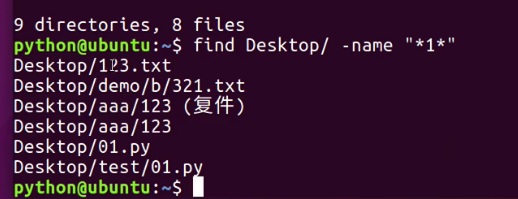
37、查找文件

find命令，通常用来在特定的目录下搜索符合条件的文件

|  |  |
| --- | --- |
| 命令 | 作用 |
| find [路径] –name “\*.py” | 查找指定路径下扩展名为.py的文件，包括子目录 |

如果省略路径，表示在当前文件夹下查找

之前学习的通配符，在使用find命令时同时可用



38、软链接

|  |  |
| --- | --- |
| 命令 | 作用 |
| ln –s 被链接的**源文件** 链接文件 | 建立文件的软链接，用通俗的方式讲，类似于windows下的快捷方式 |

注意：

1. 没有-s选项建立的是一个**硬链接文件**
2. 两个文件占用相同大小的硬盘空间，工作中几乎不会建立文件的硬链接
3. *源文件要使用绝对路径，*不要使用相对路径，这样方便移动链接文件后，任然能够正常使用。

39、打包压缩

不同操作系统中，常用的打包压缩方式是不同的

1. windows 常用rar
2. mac常用zip
3. linux常用tar.gz

**tar是linux中最常用的备份工具**，此工具可以**把一系列文件**打包到**一个大文件**中，也可以将一个打包的大文件恢复到一系列文件。

# 打包文件

tar **–cvf** 打包文件**.tar** 被打包的文件/路径…

# 解包文件

tar –xvf 打包文件.tar

tar选项说明

|  |  |
| --- | --- |
| 选项 | 含义 |
| c | 生成档案文件，创建打包文件 |
| x | 解开档案文件 |
| v | 列出归档解档的详细过程，显示进度 |
| f | 指定档案文件名称，f后边一定是.tar文件，所以必须放选项最后 |

注意：f选项必须放在最后，其他选项顺序可以随意。

压缩/解压缩

1. gzip

tar与gzip命令结合可以使用实现文件 打包和压缩

tar负责打包，但不包括压缩

用gzip压缩tar打包后的文件，扩展名一般用 xxx.tar.gz

在linux中，最常用的压缩文件格式是：xxx.tar.gz

在tar命令中有一个选项 –z可以调用gzip，从而可以方便的实现压缩和解压缩的功能。

命令格式如下：

# 压缩文件

tar **–zcvf** 打包文件**.tar.gz** 被压缩的文件/路径…

#解压缩文件

tar **–zxvf** 打包文件**.tar.gz**

#解压缩到指定路径

tar **–zxvf** 打包文件.**tar.gz**  **-C** 目标路径

|  |  |
| --- | --- |
| 选项 | 含义 |
| -C | 解压缩到指定目录。**注意：要解压缩的目录必须存在。** |

1. bzip2(two)

tar与bzip2命令结合可以实现文件 打包和压缩（用法和gzip一样）

tar只负责打包文件，但不压缩

用bzip2压缩tar打包后的文件，扩展名一般用xxx.tar.bz2

在tar命令中有一个选项-j可以调用bzip2,从而可以方便的实现压缩和解压缩的功能。

命令如下：

# 压缩文件

tar **–jcvf** 打包文件**.tars.bz2** 被压缩的文件/路径

# 解压缩文件

tar **–jxvf** 打包文件**.tar.bz2**

40、通过**apt** 安装/卸载 软件

apt 是Advanced packaging tool,是linux下的一款安装包管理工具；

可以在终端中方便的安装/卸载/更新软件包

# 安装软件

$ sudo apt install 软件包

# 卸载软件

$ sudo apt remove 软件包

# 更新已安装的包

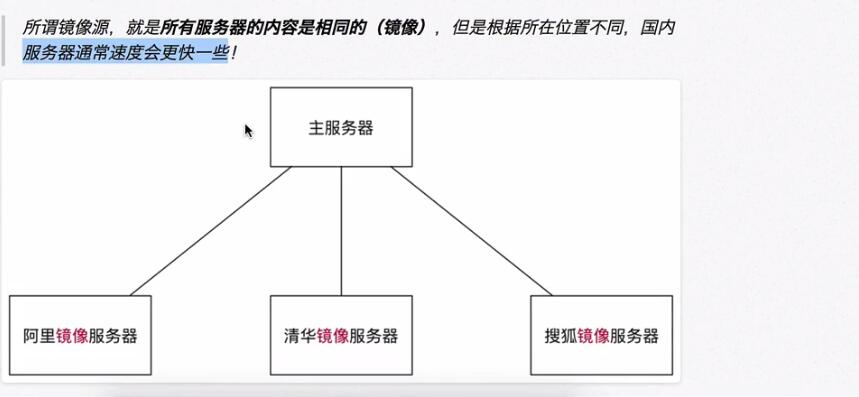
$ sudo apt upgrade

41、配置软件源

如果希望在ubuntu中安装软件，更加快速，可以通过设置镜像源，选择一个访问网速更快的服务器来提供软件下载/安装服务

提示：更换服务器之后，需要一个相对比较长时间的更新过程，需要耐心等待。更新完成后，再安装软件都会从新设置的服务器下载软件了。

所谓镜像源，就是所有服务器的内容是相同的（镜像），但是根据所在位置的不同，国内服务器通常速度会更快一些。



42、vi – 终端中的编辑器

要对服务器上的文件进行简单的修改，可以使用ssh远程登录到服务器，并且使用vi 进行快速的编辑即可。

常见需要修改的文件包括：

源程序；

配置文件，例如 ssh的配置文件 ~/.ssh/config

在没有图形界面的环境下，要编辑文件，vi是最佳选择。

43、vi

Vi的核心设计思想-让程序员的手指始终保持在键盘的核心区域，就能完成所有的编辑操作。

Vi的特点：

1. **没有图形界面**的**功能强大的编辑器**；
2. **只能编辑文本内容**，不能对文本、段落进行排版；
3. **不支持鼠标操作**；
4. **没有菜单**；
5. **只有命令**；

Vi编辑器在系统管理、服务器管理 编辑文件时，其功能永远不是图形界面的编辑器能比拟的。

44、**vim是首选，在很多linux发行版中，直接将vi做成vim的软链接。Vim被称为编辑器之神。**

45、打开和新建文件

在终端中输入vi在后边跟上文件名即可

**$ vi 文件名**

如果文件已经存在，则会直接打开该文件；

如果文件不存在，则会新建一个文件。

46、打开文件并且定位行

在日常工作中，有可能会遇到打开一个文件，并定位到指定行的情况。

例如：在开发时，知道某一行代码有错误，可以快速定位到出错代码的位置。

这个时候使用如下命令打开文件：  
**$vi 文件名 + 行数**

提示：**如果只带上+而不指定行号，会直接定位到文件末尾。**

47、异常处理

如果vi异常退出，在磁盘上可能会保存有**交换文件。**

下次再使用vi编辑该文件时，会看到以下屏幕信息，**按下字母 d 可以 删除交换文件** 即可。

提示：按下键盘时，注意关闭输入法。



48、vi 三种基本工作模式

1）命令模式：

**打开文件首先进入命令模式**，是使用vi的入口；

通过命令对文件进行常规的编辑操作，比如：**定位、翻页、复制、粘贴、删除**…;

在其他图形编辑器下，**通过快捷键或者鼠标实现的操作，都在命令模式下实现**。

2）末行模式：

执行保存、退出等操作

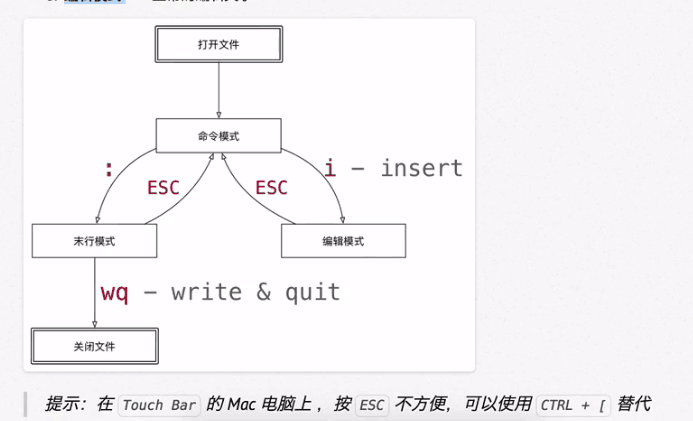
要退出vi，返回到控制台，需要在末行模式下输入命令；

末行模式是vi的出口

1. 编辑模式---正常的编辑文字

w: 保存

q:退出



|  |  |  |
| --- | --- | --- |
| 命令 | 英文 | 功能 |
| w | write | 保存 |
| q | quit | 退出，如果没有保存，不允许退出 |
| q! | quit | 强制退出，不保存退出 |
| wq | Write&quit | 保存并退出 |
| x |  | 保存并退出 |

49、常用命令

命令线路图

1. 重复次数

在命令模式下，先输入一个数字，再跟上一个命令，可以让该命令重复执行指定次数；

1. 移动和选择（多练）

Vi之所以快，关键在于能够**快速定位到要编辑的行；**

**移动命令**能够和**编辑操作命令**组合使用。

3．编辑操作

删除、复制、粘贴、替换、缩排

4．撤销和重复

5.查找替换

6编辑

Vi的命令较多，个别命令忘记了，只是会影响编辑速度而已。**在使用vi命令时，注意关闭中文输入法**。

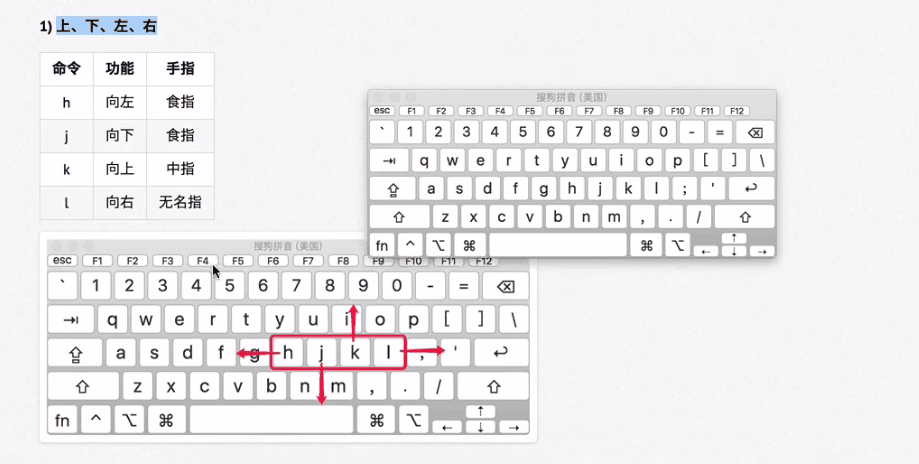
50、移动（基本）

要熟练使用vi，首先应该学会怎么在命令模式下快速移动光标；

编辑操作命令，能够和移动命令结合在一起使用。

1. 上下左右

|  |  |  |
| --- | --- | --- |
| 命令 | 功能 | 手指 |
| h | 向左 | 食指 |
| j | 向下 | 食指 |
| k | 向上 | 中指 |
| l | 向右 | 无名指 |



1. 行内移动

|  |  |  |
| --- | --- | --- |
| 命令 | 英文 | 功能 |
| w | word | 向后移动一个单词 |
| b | back | 向前移动一个单词 |
| 0 |  | 数字零，行首 |
| ^ |  | 行首，第一个不是空白字符的位置 |
| $ |  | 行尾 |

1. 行数移动

|  |  |  |
| --- | --- | --- |
| 命令 | 英文 | 功能 |
| gg | go | 文件顶部 |
| G | go | 文件末尾 |
| 数字G | go | 移动到数字对应行数 |
| 数字gg | go | 移动到数字对应行数 |
| :数字 |  | 移动到数字对应行数 |

1. 屏幕移动

|  |  |  |
| --- | --- | --- |
| 命令 | 英文 | 功能 |
| Ctrl+b | back | 向上翻页 |
| Ctrl+f | forward | 向下翻页 |
| H | head | 屏幕顶部 |
| M | Middle | 屏幕中间 |
| L | Low | 屏幕底部 |

**移动（程序）**

1. 段落移动

Vi中使用空行来区分段落

在程序开发时，通常一段功能相关的代码会写在一起。--之间没有空行

|  |  |
| --- | --- |
| 命令 | 功能 |
| { | 上一段 |
| } | 下一段 |

1. 括号切换

在程序世界中，()、[]、{}使用频率很高，而且都是成对出现的

|  |  |
| --- | --- |
| 命令 | 功能 |
| % | 括号匹配及切换 |

利用%在()、[]、{}间实现快速的切换。

1. 标记

在开发时，某一块代码可能需要稍后处理，比如：编辑、查看；

此时先使用m添加一个标记，这样可以在需要时快速地跳转回来或者执行其他编辑操作；

标记名称 可以是a-z或者A-Z之间的任意一个字母；

添加了标记的行，如果被删除，标记同时被删除；

如果 在其他行添加了相同名称的标记，之前添加的标记也会被替换掉。

|  |  |  |
| --- | --- | --- |
| 命令 | 英文 | 功能 |
| mx | mark | 添加标记x,x是a-z或者A-Z之间的任意一个字母 |
| ‘x | 单引号加标记的名称 | 直接定位到标记x所在位置 |

选中文本（可视模式）

学习**复制**命令前，首先应该学会**怎么样选中要复制的代码**，、

在vi中要选择文本，首先先使用visual命令切换到视图模式；

Vi中提供了三种可视模式。可以方便程序员选择选中文本的方式；

**按esc可以放弃选中，返回到命令模式**。

|  |  |  |
| --- | --- | --- |
| 命令 | 模式 | 功能 |
| v | 可视模式 | 从光标位置开始按照正常模式选择文本 |
| V | 可视行模式 | 选中光标经过的完整行 |
| CTRL+V | 可视块模式 | 垂直方向选中文本 |

可视模式下，**可以和移动命令连用**，例如：ggVG能够选中所有内容

51、撤销和恢复命令

在学习编辑命令之前，先要知道**怎样撤销之前一次错误的编辑操作**。

|  |  |  |
| --- | --- | --- |
| 命令 | 英文 | 功能 |
| u | undo | 撤销上次命令 |
| Ctrl+r | redo | 恢复撤销的命令 |

52、删除文本

|  |  |  |
| --- | --- | --- |
| 命令 | 英文 | 功能 |
| x | cut | 删除光标所在字符，或者选中文字 |
| d(移动命令) | Delete | 删除移动命令对应的内容 |
| dd | delete | 删除光标所在行，可以ndd复制多行 |
| D | delete | 删除至行尾 |

提示:

如果使用可视模式已经选中了一段文本，那么无论使用d还是x，都可以删除选中文本。

**删除命令可以和移动命令连用**，以下是常用的组合命令：

dw # 从光标位置删除到单词末尾

d0 # 从光标位置删除到一行的起始位置

d} # 从光标位置删除到段落结尾

ndd #从光标位置向下连续删除n行

d代码行G # 从光标所在行 删除到 指定代码行 之间的所有代码

d’a # 从光标所在行 删除到 标记a 之间的所有代码。

53、复制、粘贴

Vi中提供有一个被复制文本的缓冲区

复制 命令会将选中的文字保存在缓冲区；

删除 命令删除的文字会被保存在缓冲区；

在需要的位置，使用粘贴命令可以将缓冲区的文字插入到光标所在位置

|  |  |  |
| --- | --- | --- |
| 命令 | 英文 | 功能 |
| y(移动命令) | copy | 复制 |
| yy | copy | 复制一行，可以nyy复制多行 |
| p | paste | 粘贴 |

提示：

命令 d、x类似于图形界面的剪切操作—ctrl+x;

命令y类似于图形界面的复制操作—ctrl+copy；

命令p类似于图形界面的粘贴操作—ctrl+v

**Vi中的文本缓冲区同样只有一个，如果后续做过复制、剪切操作，之前缓冲区中的内容会被替换**。

注意：

Vi中的**文本缓冲区**和系统的**剪切板**不是同一个；

所以在其他软件中使用ctrl+c复制的内容，不能在vi中通过p命令粘贴；

可以在编辑模式下，**使用鼠标右键粘贴**。

54、替换

|  |  |  |  |
| --- | --- | --- | --- |
| 命令 | 英文 | 功能 | 工作模式 |
| r | replace | 替换当前字符 | 命令模式 |
| R | replace | 替换当前行光标后的字符 | 替换模式 |

R命令可以直接进入替换模式，替换完成后，按下esc可以回到命令模式。

替换命令的作用就是不用进入编辑模式，对文件进行轻量级的修改。

55、缩排和重复执行

|  |  |
| --- | --- |
| 命令 | 功能 |
| >> | 向右增加缩进 |
| << | 向左减少缩进 |
| . | 重复上次命令 |

**缩排命令** 在开发程序时，**统一增加代码的缩进** 比较有用！

一次性 在选中代码前**增加四个空格**，就**叫做增加缩进**；

一次性 再选中代码前**删除4个空格**，就**叫做减少缩进。**

在**可视模式**下，缩排命令只需要使用**一个>或者<**

在**程序**中，**缩进通**常用来表示**代码的归属关系**：

前面的空格越少，代码的级别越高；

前面的空格越少，代码的级别越低。

56、查找

常规查找 **任意**

|  |  |
| --- | --- |
| 命令 | 功能 |
| /str | 查找str |
|  |  |

查找到指定内容之后，使用next查找下一个出现的位置：

n:查找下一个

N:查找上一个

如果不想看到高亮显示，可以随便查找一个文件中不存在的内容即可。

单词快速匹配

|  |  |
| --- | --- |
| 命令 | 功能 |
| \* | 向后查找**当前光标所在**单词 |
| # | 向前查找**当前光标所在**单词 |