

# Linux常用命令

2021年9月6日 15:11

## ##Linux常用命令

1.ls	list	查看当前文件夹下的内容
2.pwd	print wrok directory	查看当前所在文件夹
3.cd[目录名]	change directory	切换文件夹
4.touch[文件名]	touch	如果文件不存在，新建文件
5.mkdir[目录名]	mkdir	创建目录
6.rm[文件名]	remove	删除指定的文件名
7.clear	clear	清屏

## ##Linux终端命令格式

- 1.终端命令格式
  - command [-options] [parameter]
    - command: 命令名，相应功能的英文单词或单词的缩写
    - [-options]: 选项，可以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0个，1个或者多个
    - []代表可选
- 2.查询命令帮助信息
  - command --help: 显示command命令的帮助信息
  - man command: 查询command命令的使用手册
- 3.隐藏文件
  - 建立隐藏文件，在文件前加. 例如: touch .123.txt
  - 查看隐藏文件, ls -a (ls -l可查看详细文件列表, ls -l -h可以人性化方式查看文件列表)
- 4.返回命令
  - cd ..: 返回上一层
- 5.ls命令
  - ls的通配符
    - \*: 代表任意个数的字符
    - ?: 代表任意一个字符，至少1个
    - []: 表示可以匹配字符组的任意一个
    - [abc]: 匹配a、b、c中的任意一个
    - [a-f]: 匹配a到f范围内的任意一个字符
- 6.cd命令
  - cd ~或者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 -: 在最近两个目录之间切换
- 7.删除和创建命令

- 1.touch: 创建文件或修改文件时间。如果文件不存在, 则创建空白文件; 如果存在, 则修改文件的末次修改日期
- 2.mkdir:
  - mkdir -p [目录]: 可以创建递归目录
- 3.rm: 删除文件, **文件删除后不能回恢复**
  - -r: 删除文件夹必须使用此命令
  - -f: 强制删除, 没有提示信息
- 8.拷贝和移动文件
  - tree: 可以以树状图的方式列出文件
    - -d: 只显示目录
  - cp: copy, 复制文件或者目录
    - 使用方法: cp 源文件目录 目标文件目录
    - -i: 覆盖文件前提示
    - -r: 若给出的源文件是目录文件, 则cp将地柜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
  - mv: move, 可以用来移动文件或目录, 也可以给文件或目录重命名
    - 使用方法: mv 源文件 目标文件
    - -i: 覆盖文件前提示
- 9.查看文件内容
  - cat 文件名: concatenate (连接), 查看文件内容、创建文件、文件合并、追加文件内容等功能
    - -b: 对非空行增加行号
    - -n: 对所有行增加行号
  - more 文件名: 分屏显示文件内容
  - grep 搜索文本文件名: 搜索文本文件内容
    - 例子: grep -v as 123.txt
    - -n: 显示匹配行及行号
    - -v: 显示不包含匹配文本的所有行
    - -i: 忽略大小写
    - 常用的两种模式查找:
      - ^a: 行首, 搜寻以a开头的行
      - ke\$: 行尾, 搜寻以ke结束的行
- 10.echo文字内容
  - 例子: echo Hello
- 11.重定向>和>>
  - >表示输出, 会覆盖文件原有内容
  - >>表示追加, 将内容追加到文本中
- 13.管道 |
  - Linux允许将一个命令的输出 可以通过管道作为另一个命令的输入
  - 例子: ls -lha ~ | more

- 14.远程管理常用命令
  - shutdown 选项 时间：关机或者重新启动
    - 例子：shutdown -r now：表示现在立即重启
  - 查看或配置网卡信息：
    - ifconfig：可以查看/配置计算机当前的网卡配置信息
    - ping ip地址：检测到目标IP地址的连接是否正常
- 15.远程登录和复制文件
  - 1.ssh基础
    - 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
  - 2.SSH客户端的简单实用
    - ssh [-p port] user@remote
    - user是远程及其上的用户名，如果不指定的话默认为当前用户
    - remote是远程机器的地址，可以使ip/域名，或者是别名
    - port是SSH Server监听的窗口，如果不指定，就为默认值22
    - 提示：使用exit退出当前用户登录
    - 注意：
      - ssh这个终端命令只能在Linux或Unix系统下实用
      - 如果在Windows系统中，可以安装PuTTY或者XShell客户端软件即可
- 16.scp:secure copy,在Linux下用来进行远程拷贝文件的命令
- 例子：
  - scp -P port 01.py user@remote:Desktop/01.py
  - 把本地当前目录下的01.py文件复制到远程家目录下的Desktop/01.py
  - scp -P port user@remote:Desktop/01.py 01.py
  - 把远程家目录下的01.py复制到本地当前目录下的01.py
    - 注意：P要大写
  - -r：复制文件夹时需要加-r
- 17.ssh免密码登陆
  - 1.配置公钥：
    - 执行ssh-keygen即可生成SSH钥匙，一路回车即可
  - 2.上传公钥到服务器
    - 执行ssh-copy-id -p port user@remote，可以让远程服务器记住我们的公钥
- 18.chmod使用（重要）
  - chmod可以修改用户/组对文件/目录的权限
  - 命令格式：chmod +/-rwx 文件名|目录名
  - 提示：r为read可读，w为write可写，x为可执行
- 19.组管理终端命令（需要超级用户权限）
  - 添加组：groupadd 组名
  - 删除组：groupdel 组名
  - 确认组信息：cat/etc/group

- 修改文件/目录的所属组: `chgrp -R 组名 文件/目录名`
- 20.用户管理 终端命令
  - 1.添加新用户: `useradd -m -g 组 新建用户名`
    - 说明: `-m` 自动建立用户家目录; `-g`, 指定用户所在组, 否则会建立一个和用户同名的组
  - 2.设置用户密码: `passwd 用户名`
    - 说明: 如果是普通用户可以用`passwd`直接修改密码
  - 3.删除用户: `userdel -r 用户名`
    - 说明: `-r`会自动删除用户家目录
  - 4.确认用户信息: `cat/etc/passwd | grep 用户名`
    - 说明: 新建用户后, 用户信息会保存在`/etc/passwd`文件中
  - 5.查看用户信息
    - 1.查看用户UID和GID信息: `id [用户名]`
    - 2.查看当前所有登录的用户列表: `who`
    - 3.查看当前登录用户的账户名: `whoami`
- 21.usermode
  - `usermod`可以用来设置用户的主组/附加组和登录shell
    - #修改用户的主组 (`passwd`中的GID)
    - `usermod -g 组 用户名`
    - #修改用户的附加组
    - `usermod -G 组 用户名`
    - #修改用户登录 Shell
    - `usermod -s /bin/bash`
    - 注意: 默认使用`useradd`添加的用户没有root权限, 即无法使用`sudo`命令需要用以下命令将用户添加到`sudo`附加组中: `usermod -G sudo 用户名`
- 22.which (重要)
  - `which`命令可以查看执行命令所在位置, 例如:
    - `which ls`
      - #输出
      - `#!/bin/ls`
    - `which useradd`
      - #输出
      - `#!/usr/sbin/useradd`
- 23.切换用户
  - 1.切换用户, 并且切换目录: `su - 用户名`
  - 2.退出当前登录账户: `exit`
- 24.修改文件权限
  - 1.修改拥有者: `chown`
    - #修改文件|目录拥有者
    - `chown 用户名 文件名|目录名`

- 2.修改组：chgrp
  - 递归修改文件|目录的组
  - chgrp -R 组名 文件名|目录名
- 3.修改权限：chmod
  - 递归修改文件权限
  - chmod -R 755 文件名|目录名

拥	有	者	组			其	他	
r	w	x	r	w	x	r	w	x
4	2	1	4	2	1	4	2	1

- 注意：755代表rwx r-x r-x
- 25.系统信息相关命令
  - 1.时间和日期
    - 1.date：查看系统时间
    - 2.cal：calendar查看日历，-y选项可以查看一年日历
  - 2.磁盘信息
    - 1.df -h：disk free，显示磁盘剩余空间
    - 2.du -h [目录名]：disk usage，显示目录下的文件大小
    - 选项说明：-h，以人性化的方式显示
  - 3.进程信息相关命令：
    - 1.ps aux：process status查看进程的详细状况
      - ps默认只会显示当前用户通过终端启动的应用程序
      - 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
    - 2.top：动态显示运行中的进程并且排序
      - 要退出top可以直接输入q
    - 3.kill [-9] 进程代号：终止指定代号的进程，-9表示强行终止
- 26.其他命令
  - 1.查找文件
    - find [路径] -name "\*.py"：查找指定路径下拓展名是.py的文件，包括子目录
      - 注意：如果省略路径，表示在当前文件夹下的查找；可使用通配符
  - 2.软链接：
    - ln -s 被链接的源文件 链接文件：建立文件的软连接，类似于Windows的快捷方式
    - 注意：
      - 1.没有-s选项建立的是一个硬链接文件，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

- 2.源文件要使用绝对路径，不能使用相对路径，这样可以方便移动链接文件后，仍然能够正常使用
- 3.打包压缩
  - 1.打包/解包
    - 说明：tar是Linux中最常用的备份工具，此命令可以吧一系列文件打包到一个大文件中，也可以把一个打包的大文件恢复成一系列文件
    - #打包文件
      - ◆ tar -cvf 打包文件.tar 被打包文件/路径...
    - #解包文件
      - ◆ tar -xvf 打包文件.tar
    - tar选项说明
      - ◆ c：生产档案文件，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后面一定是.tar文件，所以必须放在选项最后
  - 2.压缩/解压缩
    - 1.gzip
      - ◆ tar与gzip命令结合可以使用实现文件的 打包和压缩
        - ◇ tar只负责打包文件，但不压缩
        - ◇ 用gzip压缩tar打包后的文件，其拓展名一般用xxx.tar.gz
      - ◆ 在tar命令中一个选项-z可以调用gzip，从而可以方便实现压缩和解压缩的功能
      - ◆ #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
      - ◆ #解压缩文件到指定路径
        - ◇ tar -zxvf 打包文件.tar.gz -C 目标路径
    - 2.bzip2 (two)
      - ◆ 用bzip压缩tar打包后的文件，其拓展名一般用xxx.tar.bz2
      - ◆ 在tar命令中有一个选项-j可以调用bzip2，从而方便的实现压缩和解压缩的功能
      - ◆ #压缩文件
        - ◇ tar -jcvf 打包文件.tar.bz2 被压缩的文件/路径
      - ◆ #解压缩文件
        - ◇ tar -jxvf 打包文件.tar.bz2
      - ◆ #解压缩文件到指定路径
        - ◇ tar -jxvf 打包文件.tar.bz2 -C 目标路径、
- 27.软件安装
  - 1.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
- #1.安装软件
  - \$ sudo apt install 软件包
- #2.卸载软件
  - \$ sudo apt remove 软件名
- #3.更新已安装的包
  - \$ sudo apt upgrade
- 2.配置软件源

## ##vi——终端中的编辑器

- 1.vi和vim
  - vi
    - vi是Visual interface的简称，是Linux中最经典的文本编辑器
    - vi的特点：
      - 没有图形界面 的 功能强大的编辑器
      - 只能编辑文本内容，不能对字体、段落进行排版
      - 不支持鼠标操作
      - 没有菜单
      - 只有命令
  - vim
    - vim = vi improved
- 2.打开和新建文件
  - 在终端中输入 vi 在后面跟上文件名即可
    - \$ vi 文件名
  - 如果文件已经存在，会直接打开该文件
  - 如果文件不存在，会新建一个文件
- 3.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VI的入口
  - 2.末行模式——执行保存、退出等操作
    - 命令模式下按：进入末行模式，按esc返回命令模式
    - 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
    - 末行模式命令：
      - w：保存
      - q：退出，如果没有保存，不允许退出
      - q!：强行退出，不保存退出
      - wq：保存并退出
      - x：保存并退出

- 3.编辑模式——正常的编辑模式
  - 命令模式下按 i 进入编辑模式，按esc返回命令模式
- 4.常用命令
  - 1.移动 (基本)
    - 1) 上、下、左、右
      - ◆ h: 向左——食指
      - ◆ j: 向下——食指
      - ◆ k: 向上——中指
      - ◆ l: 向右——无名指
    - 2) 行内移动
      - ◆ w: word 向后移动一个单词
      - ◆ b: back 向前移动一个单词
      - ◆ 0: 行首
      - ◆ ^: 行首，第一个不是空白字符的位置
      - ◆ \$: 行尾
    - 3) 行数移动
      - ◆ gg: go——文件顶部
      - ◆ G: go——文件末尾
      - ◆ 数字gg: go——移动到 数字 对应行数
      - ◆ 数字G: go——移动到 数字 对应行数
      - ◆ :数字: 移动到 数字 对应行数
    - 4) 屏幕移动
      - ◆ 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
  - 2.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卸载一起——之间没有空行
      - ◆ { : 上一段
      - ◆ } : 下一段
    - 2) 括号切换
      - ◆ %: 括号匹配及切换
    - 3) 标记
      - ◆ 使用m增加一个标记
      - ◆ 标记名称 可以使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：
  - ◇ mx: 即mark——添加标记x, x为a-z或者A-Z
  - ◇ 'x: 直接定位到标记x所在位置
- 4) 选中文本 (可视模式)
  - ◆ v: 可视模式, 从光标处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
- 5) 撤销和恢复撤销
  - ◆ u: 即undo, 撤销上次命令
  - ◆ CTRL+r: 即redo, 恢复撤销的命令
- 6) 删除文本
  - ◆ x: 即cut, 删除光标所在字符, 或者选中文字
  - ◆ d (移动命令): 即delete, 删除移动命令对应的内容
  - ◆ dd: 删除光标所在行, 可以n dd复制多行
  - ◆ D: 删除至行尾
  - ◆ #删除命令可以和移动命令连用
- 7) 复制、粘贴
  - ◆ y (移动命令): 复制
  - ◆ yy: 复制一行, 可以n yy复制多行
  - ◆ p: 粘贴
  - ◆ 注意: vi中的文本缓冲区只有一个, 如果后续做过复制、剪切操作, 之前缓冲区的内容会被替换
- 8) 替换
  - ◆ r: 即replace, 替换当前字符, 命令模式下使用
  - ◆ R: 替换当前行光标后的字符, 替换模式下使用
  - ◆ R命令可以进入 替换模式, 替换完成后, 按下ESC可以回到 命令模式
- 9) 缩排和重复执行
  - ◆ >>: 向右增加缩进
  - ◆ <<: 向左减少缩进
  - ◆ .: 重复上次命令
- 10) 查找
  - ◆ /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
  - ◆ 单词快速匹配
    - ◇ \*: 向后查找当前光标所在单词
    - ◇ #: 向前查找当前光标所在单词
- 11) 查找并替换
  - ◆ 在 vi 中查找和替换命令需要在末行模式下执行

- ◆ 1) 全局替换
  - ◇ : %s/旧文本/新文本/g
- ◆ 2) 可视区域替换
  - ◇ 先选中要替换文字的范围
  - ◇ : s/旧文本/新文本/g
- ◆ 3) 确认替换
  - ◇ : %s/旧文本/新文本/gc
  - ◇ 命令确认说明:
    - ▶ y——yes 替换
    - ▶ n——no 不替换
    - ▶ a——quit 替换所有
    - ▶ q——last 退出替换
    - ▶ CTRL+E——向下滚屏
    - ▶ CTRL+Y——向上滚屏
- 12) 插入命令
  - ◆ i: 在当前字符前插入文本
  - ◆ I: 在行首插入文本
  - ◆ a: 在当前字符后添加文本
  - ◆ A: 在行末添加文本
  - ◆ o: 在当前行后面插入一空行
  - ◆ O: 在当前行前面插入一空行
- 5.分屏命令
  - 1.末行命令拓展
    - :e . : 即edit, 会打开内置的文件浏览器, 浏览当前目录下的文件
    - :n [文件名] : 即new, 新建文件
    - :w [文件名] : 即write, 另存为, 但是仍然编辑当前文件, 并不会切换文件
  - 2.分屏命令:
    - :sp [文件名]: 横向增加分屏
    - :vsp [文件名]: 纵向增加分屏
    - 1) 切换分屏窗口
      - ◆ 分屏窗口都是基于CTRL+W这个快捷键, 即都要先按下ctrl+w
      - ◆ w: 即window, 切换到下一个窗口
      - ◆ r: 即reverse, 互换窗口
      - ◆ c: 即close, 关闭当前窗口, 但是不能关闭最后一个窗口
      - ◆ q: 即quit, 退出当前当前窗口, 如果是最后一个窗口, 则关闭vi
      - ◆ o: 即other, 关闭其他窗口