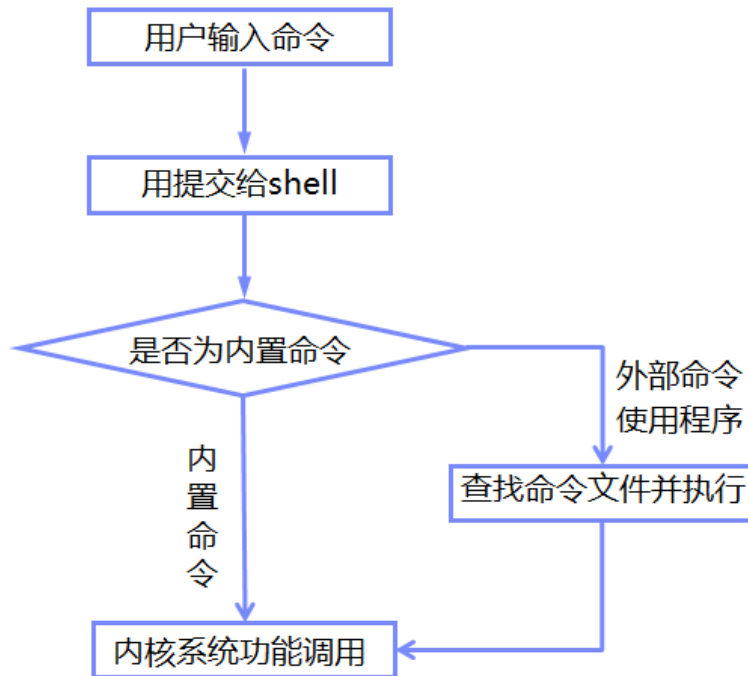


Linux操作系统

- 为什么需要学习Linux操作系统
 - 服务器后端一般都是类unix操作系统
 - 常见的Linux命令
 - 查看日志
 - 面试官：如何查看日志文件？
 - 搭建测试环境
 - 高级：系统资源监控与优化
- Linux操作系统的版本
 - Red Hat Linux、unbutu linux、centos linux...
- 安装Linux
 - VM ware虚拟机
 - 虚拟化技术
 - 虚拟化的技术有两种：vm和docker
 - 在VM中安装Linux操作系统
- 客户端连接工具
 - putty
 - 连接成功后的字符串
 - [root@localhost ~]# root：表示当前登录的帐户名 localhost：表示连接的服务器 ~：表示默认的路径位置，当前帐户的home目录
- Linux文件系统结构
 - bin：系统自带的应用程序
 - boot：主要存放操作系统启动时的相关引导文件
 - etc：存放相关配置文件
 - root：root用户的工作目录
 - home：除root之外的其它用户的工作目录
 - lib：操作系统使用到的链接库文件
 - logs：存放日志文件的地方
 - mnt：挂载点
 - sbin：操作系统启动时的应用程序
 - var：存放变量的
- Linux命令工作原理
 - 命令格式：命令 -选项 参数 命令：表示一个应用程序的名称

- 工作原理

-



- 文件管理命令

- ls全称list 显示当前目录下文件的列表信息 或一个文件详细信息

- ls

- ls -l或ll

- drwxr-xr-x 2 root root 4096 11月 16 2021 abc drwxr-xr-x: 第一位表示当前的文件类型, 第2-4位表示该文件所属用户的权限, 第5-7位该文件所属组的权限, 第8-10位表示其它用户的权限 第一个root, 表示当前文件所属的用户 第二个root, 表示当前文件所属的组 4096表示文件大

- ls -a

- ls -l test.txt

- pwd全称print work directory

- 显示当前的工作目录

- cd全称change directory 更改工作目录

- cd 目录

- cd 绝对路径 如cd /home

- cd 相对路径 如cd abc

- cd .. 返回到上一级目录

- cd ../software/

- cd 切换到当前用户的home目录

- cd ~

- cd ~username 表示切换到username用户所在的home目录

- touch 创建一个空文件
 - touch filename
- cp 全称copy
 - cp 选项 源文件 目标目录
 - cp -r 源目录 目标目录 拷贝文件夹，需要使用-r选项，表示递归拷贝
 - cp -p 源文件 目标目录 表示拷贝文件时保存原文件的属性
- mv 全称move
 - mv 源文件 目标目录
 - mv test.txt ./1.txt
 - mv 1.txt test.txt 重命名文件
- mkdir 创建一个目录
 - mkdir 选项 目录名
 - mkdir abc
 - mkdir -p 1/2/3/4
- rmdir 删除一个目录
 - rmdir 选项 目录名
 - rmdir 2 只能删除空目录
 - rmdir -p 1/2/3/4 依次删除空目录
- rm 全称remove，删除文件或目录
 - rm 选项 文件或目录
 - rm 1.txt
 - rm -f 1.txt
 - rm -rf 1 递归+强制删除
 - rm -rf 目录/* 删除当前目录下所有的文件，但该目录还是保留
- cat 查看文件的内容
 - cat filename
 - cat filename|grep '条件'
 - 查日志 cat access_log |grep '192.168.1.113'|grep '13/Apr/2022'|awk '{print \$2}'|uniq -c
 - 重定向输入 cat < grep.txt
 - 重定向输出 cat grep.txt > 1.txt 将1.txt文件中原来的内容覆盖
 - 追加重定向输出 cat grep.txt >> 1.txt 将新添加的内容写到1.txt文件的尾部
 - 合并多个文件 cat filename1 filename2 > filename
- more 分页显示
 - more filename
- less 分页显示

- less filename
- head 显示文件的前多少行
 - head -n 10 显示文件的前10行
- tail 显示文件的后多少行
 - tail -n 10 显示文件的后10行
 - 动态查看日志文件 tail -f filename
- chmod 修改文件
 - 符号模式
 - chmod who operation permission filename
 - who: u、g、o、a
 - operation: +、-、=
 - permission: w、r、x
 - chmod ug-w 1.sh chmod u-x,g+w 1.sh
 - 绝对模式
 - chmod mode filename
 - 权限是一个三位数字: xxx 第一位表示用户的权限 第二位表示组的权限 第三位表示其它用户的权限 权限使用一个八进制的数表示 r 4 w 2 x 1
 - chmod 755 1.sh
- chown 修改拥有者
 - chown user filename
 - chown test grep.txt
- chgrp 修改所属的组
 - chgrp group filename
 - chgrp test grep.txt
- ln 全称link, 创建一个链接文件, 相当于windows下的快捷方式
 - link -s src-file file_link_name
 - ln -s test.sh test.sh.ln
- which
 - which 文件或程序
- whereis
 - whereis 文件或程序
- 用户管理
 - adduser 添加一个帐户
 - adduser username
 - passwd 修改密码

- passwd username
- userdel 删除用户
 - userdel -r username
- 用户组管理
 - groupadd 添加用户组
 - groupadd groupname
 - gpasswd 修改组的密码
 - gpasswd groupname
 - 将用户添加到某个组
 - gpasswd -a username groupname
 - 将组中某个用户删除
 - gpasswd -d username groupname
- 系统管理
 - ps 查看进程信息
 - ps -aux ps -ef 查看所有进程
 - 查看某个应用程序是否运行
 - ps aux|grep 'mysql'|grep -v 'grep'
 - kill 结束进程
 - kill 选项 PID -2 中断进程，相当于ctrl+c -15 正常结束进程 -9 强制结束进程
 - top 查看系统资源
 - top 选项 参数 top -d 5 //每5秒刷新一次系统资源的信息
 - 其它查看系统资源的命令 free、iostat、vmstat、sar
- 磁盘管理
 - df 查看文件挂载的相关信息
 - df -h
 - du 查文件或目录的相关信息
 - du
- 网络管理
 - ifconfig 查网络设备的信息
 - netstat 查看端口
 - netstat -anp 查看所有商品的信息
 - 查看某端口是否被占用
 - netstat -anp |grep ':80'
 - 查看某个应用程序所使用的端口号
 - netstat -anp |grep 'httpd'

- 压缩与解压命令

- gzip

- gzip 选项 文件或目录 -1: 快速压缩 -9: 最佳方式压缩 -r: 递归压缩
 - gzip -9r md

- gunzip

- gunzip 选项 压缩包
 - gunzip -9r md

- zip

- zip 选项 压缩后的压缩包名 文件或目录
 - zip md.zip md 把目录看成一个文件进行压缩，里面的文件不会依次进行压缩
 - zip -r md.zip md 依次对目录下每个文件进行压缩

- unzip

- unzip 压缩包名

- tar 调用gzip或bzip进行压缩和解压

- 对文件打包，但不压缩 tar cvf anaconda.log.tar anaconda.log
 - 查看压缩包的信息 tar -tvf anaconda.log.tar
 - 把一个打包好的文件进行解包 tar -xvf anaconda.log.tar
 - 指定解包的路径 tar -xvf anaconda.log.tar -C test
 - 使用gzip打包并压缩 tar czvf anaconda.log.tar.gz anaconda.log
 - 使用bzip2打包并压缩 tar cjvf anaconda.log.tar.bz2 anaconda.log
 - 使用gzip解压缩 tar xzvf anaconda.log.tar.gz
 - 使用bzip2解压缩 tar xjvf anaconda.log.tar.bz2

- rpm redhat package manager

- 查找某个应用程序是否已经安装 rpm -qa |grep 'mysql'
 - 安装应用程序 rpm -ivh XXX.rpm
 - 卸载应用程序 rpm -e 应用程序名

- 安装非rpm类的应用程序

- 1.使用tar对应用程序的安装包进行解压缩
 - 2.configure 配置安装过程中需要的参数 ./configure --prefix=/desired/path/of/apr
 - 3.make 编译安装程序
 - 4.make install 安装应用程序

- vi编辑器

- vi编辑器的三种模式：命令模式、未行模式、输入模式
 - 打开、新建、保存、退出
 - vi filename 打开一个文件

- vi filename1 filename2.... 打开多个文件
 - first 跳到第一个文件
 - next 跳到下一个文件
 - prev 跳到上一个文件
 - last 跳到最后一个文件
 - args 显示所有打开的文件
- :q 退出文件，但是一定该文件未修改
- :q! 不保存修改并退出
- :w 保存文件
- :w filename 另存为
- :wq 保存并退出
- :e filename 如果之前打开的文件未修改，那么关闭原来的文件，打开一个新的文件
- :e! filename 如果之前的文件修改过，那么不保存原来的文件，直接打开新的文件
- :r filename 将一个文件的内容添加到当前已经打开的文件内容中，添加的位置为光标所在位置的后边
- 进入输入模式
 - i 当前光标进入插入状态
 - a 光标后进入插入状态
 - A 行尾进入插入状态
 - o 当前光标所在行的下面插入一行新行
 - O 当前光标所在行的上面插入一行新行
 - cw 删除光标到词尾，并进入插入状态
 - cb 删除光标到词首，并进入插入状态
 - c\$ 删除光标到行尾，并进入插入状态
 - c^ 删除光标到行首，并进入插入状态
- 光标跳转
 - :set nu 显示行号
 - :set nonu 不显示行号
 - n G 表示跳到第n行
 - G 表示跳到最后一行
 - \$ 光标跳到行尾
 - ^ 表示跳到行首
 - w 表示跳到下一个词的词首
 - b 上一个词的词首
 - e 当前词的词尾或下一个词的词尾

- CTRL+f 显示下一屏的内容
- CTRL+b 显示上一屏的内容
- CTRL+u 显示下一个半屏的内容
- CTRL+d 显示上一个半屏的内容
- 删除操作
 - x 删除一个字符
 - dd 删除一行
 - d\$ 删除当标到行尾的内容
 - d^ 删除当标到行首的内容
 - db 删除光标到词首的内容
 - dw 删除光标到下一个词的词首内容
 - de 删除光标到当前词词尾的内容
 - J 删除回车符，即合并行的意思
- 复制与粘贴
 - yy 复制整个行的内容
 - yw 复制光标到下一个词的词首的内容
 - ye 复制光标到当前词词尾的内容
 - yb 复制光标到词首的内容
 - y\$ 复制光标到行尾的内容
 - y^ 复制光标到行首的内容
 - p 将复制的内容进行粘贴
- 查找
 - :/pattern 表示当标往下查找
 - n 查找上一个内容
 - N 查找一个内容
 - :?pattern 表示往上查找
- 字符串替换操作
 - :s/old/new 表示只替换当前行第一次出现的内容
 - :s/old/new/g 表示只替换当前行所有的内容
 - :#1,#2s/old/new 表示从#1行到#2行第一次出现的内容替换掉
 - :#1,#2s/old/new/g 表示从#1行到#2行所有出现的内容替换掉
 - :%s/old/new/g 表示所有行所有出现的内容替换掉
- grep 可以在文本文件中按要求提取需要的行信息
 - 语法格式
 - grep [选项] 正则表达式 [文件]

- 选项

- -c 只输出匹配行的计数
- -i 不区分大小写
- -h 查询多个文件时不显示文件名
- -l 如果在多文件中进行查询，只现示匹配行的文件名
- -n 显示匹配行及行号
- -v 显示不包含匹配行的所有内容

- 正则表达式

- [] 范围匹配 中括号只占一位，这一位的内容可能是中括号中任意的一个字符 例：grep '199[a-zA-Z0-9]' grep.txt
- \$: 表示行尾 ^: 表示行首 用于匹配以什么开头，以什么结尾情况 grep '^48' grep.txt grep '48\$'grep.txt grep '^\$' grep.txt
- 匹配大小写 grep 'US[Pp]' grep.txt
- 精确匹配 grep '48\>' grep.txt
- .: 表示任意一个字符 grep '5A....6' grep.txt
- 匹配所有内容 grep '*' grep.txt
- 匹配特殊字符需要使用转义符\ grep '\.\\.' grep.txt
- p\{m,n\}: 表示p字符出现[m,n]次 p\{m,\}: 表示p字符出现大于等于m次 p\{m\}: 表示字符出现等于m次 例：grep '[1-9]\{1,3\}\.[0-9]\{1,3\}\.' grep.txt
- 匹配或的情况 grep -E '48|216' grep.txt
- 类名
 - [[:digit:]] 相当[0-9]
 - [[:upper:]] 相当[A-Z]
 - [[:lower:]] 相当[a-z]
 - [[:alnum:]] 相当[a-zA-Z0-9]
 - [[:space:]] 空格或跳格TAB键
 - [[:alpha:]] 相当[a-zA-Z]

- awk 如果需要提取一个数据包或数据域

- 语法格式

- awk -F feild-sperarator command inputfile
- -F 'feild-sperarator' 设置列与列之间的分隔符，默认为空格
- 域 \$#: #是数字，表示第几列 1: 表示第1列 2: 表示第2列 \$0: 表示所有列，所有域
- 打印报告开头部分
 - BEGIN {print "..."} awk 'BEGIN {print "price\tfilght"} {print 3}' grep.txt
- 打印报告结尾部分

- END {print "..."} awk 'BEGIN {print "price\tflight"} {print 3} END {print "end-of-report"}' grep.txt price flight
- 条件运行符
 - 正则表达式匹配
 - {if(0) awk '{if(1 /48/)print0}' grep.txt0} awk '0}' grep.txt
- 操作符
 - 输入域到变量名 awk '{price=0}' grep.txt
 - 域值进行比较，域值通过BEGIN来传递 awk 'BEGIN {PRICE=400} {if(0)} grep.txt
 - 修改域的值 awk '{if(1 /48/)7 = 7 + 10; print0}' grep.txt
 - 只有显示修改过的域值 awk '{if(7 = 7 + 10; print0}}' grep.txt
- find 查找文件，查找之后可以打印、对文件进行操作
 - 语法格式
 - find pathname option [print exec ok]
 - name 按文件名来查找文件 find . -name 'test?.txt' -print
 - perm 按权限查找 find . -perm 644 -print
 - user 按文件的用户属主来查找文件 find . -user root -print nouser 查找没有有效用户属主的文件 find . nouser -print
 - group 按文件属组来查找文件 find . -group root -print nogroup 查找没有有效用户属组的文件 find . -nogroup -print
 - type 按文件类型来查找 b - 块设备文件。d - 目录。c - 字符设备文件。p - 管道文件。l - 符号链接文件。f - 普通文件。 find . -type d -print find . -type f -print
 - mtime 按文件的创建时间来查找文件 查找1天内的文件 find . -mtime -1 -print 查找1天前的文件 find . -mtime +1 -print
 - size 按文件大小来查找文件 文件大小表示的方式为两种：一是字节数；二是块大小，一块等于512个字节 文件大小比较的方式有三种：大于、等于、小于 find . -size +100c -print 查找文件大于100个字节 find . -size -100c -print 查找文件小于100个字节 find . -size 100c -print 查找文件等于100个字节 find . -size +100 -print 查找文件大于100个块的文件
 - 对查找到的文件进行操作 exec或ok exec command {} \; ok command {} \; find . -name 'test1.txt' -ok rm -rf {} \;