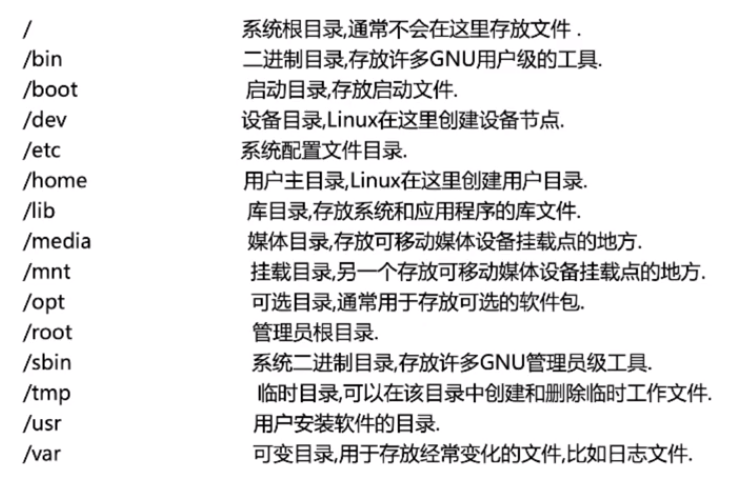
1.linux基础

常见目录



磁盘管理

pwd 查看当前位置路径

ls 查看目录文件 (-a 查看所有文件，含隐藏) (-ltr 详细信息)

cd 切换目录

mkdir 创建目录 (-p 递归创建目录)

rmdir 删除目录

文件管理

cp 拷贝文件 目录

mv 移动文件 文件改名

rm 删除普通文件 (-rf 删除文件及文件夹)

cat 查看文件内容

head/tail -n num xxx 查看xxx文件的开头/结尾num行

diff A B 对比A和B的异同

文件压缩

tar (-zcvf 先打包后压缩) (-zxvf 解压 -C 解压到指定目录)

zip压缩 unzip解压

zip 选项[-r] [压缩后文件名称] [文件或目录] (-r 压缩目录)

tar打包和zip压缩

tar使用在unix系统下，zip用于windows系统

linux主要有三种压缩方式：  
1.gzip：是公认的压缩这速度最快，压缩大文件的时候与其他的压缩方式相比更加明显，历史最久，应用最广泛的压缩方式。gzip只能压缩文件 , 不能压缩目录 ,后缀名为.gz , 而且不保留原文件。解压使用gzip –d或者 gunzip  
2.bzip：压缩形成的文件小，但是可用性不如gzip  
3.xz：是最新的压缩方式，可以自动提供最佳的压缩率

帮助命令 man 查看命令或文件的详细信息

用户管理

useradd 创建用户

userdel 删除用户

passwd 修改密码

权限管理

-rwxrwxrwx- 777 从前到后分别为归属人、归属组、其他人的权限

读4、写2、执行1

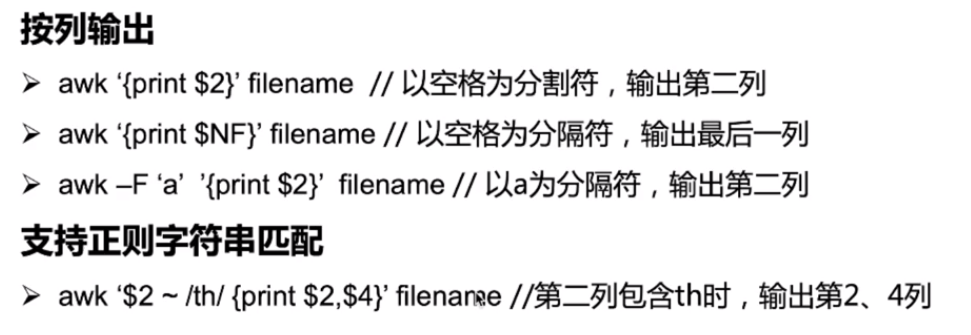
修改权限 chmod 750 filename (-R 递归修改目录下所有文件)

修改归属 chmod user:group filename (-R 递归修改目录下所有文件)

文件修改 sed -i ‘s/aaa/bbb/g’ filename 将所有的aaa替换为bbb

删除文件1-3行内容 sed -i 1,3d filename

文本输出命令awk



文本编辑vim

命令模式





插入模式



可视模式

通过(ctrl+v, v)进入

搜索内容、文件命令

grep -n ‘t[ae]st’ filename

grep abc -rl dirname 查找包含abc内容的文件

find / -name filename 从根目录查找名为filename的文件

find / -name filename -exec ls -l {} \; 并查看文件属性

软件包管理

Centos：yum

Ubuntu：apt

进程管理

ps / top 查看进程

kill <pid> 杀死pid编号的进程 (-9 强制杀死)

磁盘管理

df 显示磁盘分区上可以使用的磁盘空间 (-h 带有单位)

du 显示每个文件和目录的磁盘使用空间

2. git基础

SVN 集中式版本控制系统 有一个单一集中管理的服务器

Git 分布式版本控制系统 每个人的电脑上都是一个完整的版本库

git init 创建本地版本库

git add 添加到缓存区

git commit 提交到本地仓库

git stash 放入暂存区，暂时不想提交的修改

git checkout -f 撤销丢弃本地修改

git rm –cached 撤销添加到缓存区的修改

git reset bb0b350xxx 回滚到hashId的状态

--soft HEAD 回到上一次文件改变之后，未commit的状态

--hard HEAD 回到上一次文件没改之前的状态

git tag v1 给本次提交贴标签v1

git status 查看当前状态

git log 查看提交记录

git分支只是一个包含所指对象校验和的文件，创建分支本质上是向一个文件写入41个字节。

git branch test 创建test分支

git checkout test 切换到test分支

git merge test 合并test分支

git branch -d test 删除test分支

git branch -d -r <branchname> //删除远程分支，删除后还需推送到服务器

git push origin:<branchname> //删除后推送至服务器

git clone 克隆仓库到本地文件夹

git remote -v 获取远程仓库信息

git remote update origin 更新远程仓库变更 拉取所有分支变更

git fetch <远程主机名><远程分支名>:<本地分支名>

git pull = git fetch + git merge origin/develop

git push <远程主机名><本地分支名>:<远程分支名>

git add – git commit – git pull – 解决冲突 – git push