vi笔记

# 同步两台服务器下指定目录文件

rsync -avzu --progress /cloud/ftp/ dell@<10.155.137.131>:/data/ftp/

# 查端口号,过滤加grep

netstat -tunlp

# 文件统计

|  |  |
| --- | --- |
| 统计：文件行数 | wc -l test.txt |
| 统计：字节数 | wc -c test.txt |
| 统计：单词数 | wc -w test.txt |
| 统计：行数 单词数 字节数 文件名 | wc test.txt |
| **统计当前目录下的文件数** | ls -l | wc -l |

# 切换用户

sudo su root 输docker密码，而不是root密码

# top free df

|  |  |
| --- | --- |
| top -M | 内存以M为单位显示 |
| free –m | 内存以M为单位显示 |
| du -sh ~/\* | 统计指定目录的子目录大小 |

# VI

不在冒号下输入：

1）把光标移动到要复制的行上，按yy （复制当前行）

2）把光标移动到要复制的位置 ，按p （粘贴到指定行）

3）把光标移动到要复制的位置 ，按dd（删除当前行）

VIM

显示行号：命令模式下set nu

定位到指定行：

命令模式下，:n 比如想到第2行，:2

编辑模式下，ngg 比如想到第5行 5gg(或者5G)

打开文件定位到指定行 vim +n test.txt

查看当前行信息 ctrl+g (显示类似信息："start\_computer.sh" 13 行 --30%-- )

# 磁盘操作

建立软连接（快捷方式）：ln –sf 目标目录 快捷方式目录

ls –lh 文件大小以K或M来显示

# 日志查找

过滤要查看的行：grep 关键字 文件名 | grep 关键字2

grep -n -B10 –A40 "关键字" file #匹配关键字的前一行与后一行.

grep [参数] 关键字 文件名

|  |  |
| --- | --- |
| 参数 |  |
| -n | 显示行号 |
| -B10 | 匹配关键字的前10行 |
| -A40 | 匹配关键字的后40行 |
| -C | 统计匹配的行数 |

显示文件x行到y行的内容：sed -n 'x,yp' filename

# tar

打包 tar -zcvf /tmp/etc.tar.gz /etc

解包 tar -zxvf fcitx-3.2-050827.tar.gz

# 修改环境变量

编辑 /etc/profile

让环境变量生效：source /etc/profile

# 查看系统配置

|  |
| --- |
| 查CPU  cat /proc/cpuinfo |
| 查内存  free |
| 查硬盘 |
| 查个目录大小  df |

# 修改系统时间

sudo date -s 10:20:35

# 搜索

## 1. find 指定目录查找文件 慢

find是最常见和最强大的查找命令，你可以用它找到任何你想找的文件。

find的使用格式如下：

　　$ find <指定目录> <指定条件> <指定动作>

　　- <指定目录>： 所要搜索的目录及其所有子目录。默认为当前目录。

　　- <指定条件>： 所要搜索的文件的特征。

　　- <指定动作>： 对搜索结果进行特定的处理。

如果什么参数也不加，find默认搜索当前目录及其子目录，并且不过滤任何结果（也就是返回所有文件），将它们全都显示在屏幕上。

find的使用实例：

　　$ find . -name 'my\*'

搜索当前目录（含子目录，以下同）中，所有文件名以my开头的文件。

　　$ find . -name 'my\*' -ls

搜索当前目录中，所有文件名以my开头的文件，并显示它们的详细信息。

　　$ find . -type f -mmin -10

搜索当前目录中，所有过去10分钟中更新过的普通文件。如果不加-type f参数，则搜索普通文件+特殊文件+目录。

**实例：查找并删除.svn**

|  |
| --- |
| find ./ -name ".svn" -print -exec rm -fr {} \; |
| find ./ -name "\*.DS\_Store" -print -exec rm -fr {} \; |
| find ./ -name ".gitignore" -print -exec rm -fr {} \; |

## 2. whereis 只能用于程序名的搜索

whereis命令只能用于程序名的搜索，而且只搜索二进制文件（参数-b）、man说明文件（参数-m）和源代码文件（参数-s）。如果省略参数，则返回所有信息。

whereis命令的使用实例：

　　$ whereis grep

## 3. which 搜索命令

which命令的作用是，在PATH变量指定的路径中，搜索某个系统命令的位置，并且返回第一个搜索结果。也就是说，使用which命令，就可以看到某个系统命令是否存在，以及执行的到底是哪一个位置的命令。

which命令的使用实例：

　　$ which grep

# 压缩、解压命令

## .tar

解包：tar xvf FileName.tar

打包：tar cvf FileName.tar DirName

（注：tar是打包，不是压缩！）

———————————————

## .gz

解压1：gunzip FileName.gz

解压2：gzip -d FileName.gz

压缩：gzip FileName

## .tar.gz 和 .tgz

解压：tar zxvf FileName.tar.gz

压缩：tar zcvf FileName.tar.gz DirName

———————————————

## .bz2

解压1：bzip2 -d FileName.bz2

解压2：bunzip2 FileName.bz2

压缩： bzip2 -z FileName

## .tar.bz2

解压：tar jxvf FileName.tar.bz2

压缩：tar jcvf FileName.tar.bz2 DirName

———————————————

## .bz

解压1：bzip2 -d FileName.bz

解压2：bunzip2 FileName.bz

压缩：未知

## .tar.bz

解压：tar jxvf FileName.tar.bz

压缩：未知

———————————————

## .Z

解压：uncompress FileName.Z

压缩：compress FileName

## .tar.Z

解压：tar Zxvf FileName.tar.Z

压缩：tar Zcvf FileName.tar.Z DirName

———————————————

## .zip

解压：unzip FileName.zip

压缩：zip FileName.zip DirName

———————————————

## .rar

解压：rar x FileName.rar

压缩：rar a FileName.rar DirName

———————————————

## .lha

解压：lha -e FileName.lha

压缩：lha -a FileName.lha FileName

———————————————

## .rpm

解包：rpm2cpio FileName.rpm | cpio -div

———————————————

## .deb

解包：ar p FileName.deb data.tar.gz | tar zxf -

———————————————

.tar .tgz .tar.gz .tar.Z .tar.bz .tar.bz2 .zip .cpio .rpm .deb .slp .arj .rar .ace .lha .lzh .lzx .lzs .arc .sda .sfx .lnx .zoo .cab .kar .cpt .pit .sit .sea

解压：sEx x FileName.\*

压缩：sEx a FileName.\* FileName

注：sEx只是调用相关程序，本身并无压缩、解压功能，请注意！

## gzip 命令

减少文件大小有两个明显的好处，一是可以减少存储空间，二是通过网络传输文件时，可以减少传输的时间。gzip 是在 Linux 系统中经常使用的一个对文件进行压缩和解压缩的命令，既方便又好用。

语法：gzip [选项] 压缩（解压缩）的文件名该命令的各选项含义如下：

-c 将输出写到标准输出上，并保留原有文件。-d 将压缩文件解压。-l 对每个压缩文件，显示下列字段： 压缩文件的大小；未压缩文件的大小；压缩比；未压缩文件的名字-r 递归式地查找指定目录并压缩其中的所有文件或者是解压缩。-t 测试，检查压缩文件是否完整。-v 对每一个压缩和解压的文件，显示文件名和压缩比。-num 用指定的数字 num 调整压缩的速度，-1 或 --fast 表示最快压缩方法（低压缩比），-9 或--best表示最慢压缩方法（高压缩比）。系统缺省值为 6。指令实例：

gzip \*% 把当前目录下的每个文件压缩成 .gz 文件。gzip -dv \*% 把当前目录下每个压缩的文件解压，并列出详细的信息。gzip -l \*% 详细显示例1中每个压缩的文件的信息，并不解压。gzip usr.tar% 压缩 tar 备份文件 usr.tar，此时压缩文件的扩展名为.tar.gz。

# date

|  |  |
| --- | --- |
| date +%Y%m%d%H%M | 当前时间 |
| date -d "-1 hour" +%Y%m%d%H | 1小时前 |
| date -d "yesterday" +%Y%m%d | 1天前 |

# 网络配置

cd /etc/sysconfig/network-scripts/ #进入网络配置文件目录

vi ifcfg-eno16777736 #编辑配置文件，添加修改以下内容

|  |
| --- |
| HWADDR=00:0C:29:8D:24:73  TYPE=Ethernet  BOOTPROTO=static  #启用静态IP地址  DEFROUTE=yes  PEERDNS=yes  PEERROUTES=yes  IPV4\_FAILURE\_FATAL=no  IPV6INIT=yes  IPV6\_AUTOCONF=yes  IPV6\_DEFROUTE=yes  IPV6\_PEERDNS=yes  IPV6\_PEERROUTES=yes  IPV6\_FAILURE\_FATAL=no  NAME=eno16777736  UUID=ae0965e7-22b9-45aa-8ec9-3f0a20a85d11  ONBOOT=yes  #开启自动启用网络连接  IPADDR0=192.168.21.128  #设置IP地址  PREFIXO0=24  #设置子网掩码  GATEWAY0=192.168.21.2  #设置网关  DNS1=8.8.8.8  #设置主DNS  DNS2=8.8.4.4  #设置备DNS |
| 静态IP  TYPE=Ethernet  BOOTPROTO=none  DEFROUTE=yes  IPV4\_FAILURE\_FATAL=no  IPV6INIT=no  IPV6\_AUTOCONF=yes  IPV6\_DEFROUTE=yes  IPV6\_FAILURE\_FATAL=no  NAME=eth0  UUID=e8541cad-8639-4d8e-95af-afd3982ff0a4  ONBOOT=yes  IPADDR0=10.0.88.30  #GATEWAY0=10.0.0.1  #DNS1=114.114.114.114  PREFIX0=8  IPV6\_PEERDNS=yes  IPV6\_PEERROUTES=yes  HWADDR=40:16:7E:36:48:99 |

:wq!  #保存退出

service network restart   #重启网络

ping www.baidu.com  #测试网络是否正常

ip addr  #查看IP地址

# 设置主机名

|  |
| --- |
| vi /etc/hostname #编辑配置文件  www   #修改localhost.localdomain为www  :wq!  #保存退出 |

# 防火墙

## firewall 主要是centos上

sudo systemctl stop firewalld.service #停止firewall  
systemctl disable firewalld.service #禁止firewall开机启动

systemctl status firewalld.service #查看防火墙状态

sudo systemctl start firewalld.service #启动防火墙

## iptables 使用最广泛

sudo systemctl start iptables

/etc/sysconfig/iptables

# 其它系统防火墙

1） 临时生效，重启后复原

开启： service iptables start

关闭： service iptables stop

2） 永久性生效，重启后不会复原

开启： chkconfig iptables on

关闭： chkconfig iptables off

# 查看端口占用进程

netstat –nplt | grep 3306

或

netstat -tunlp|grep 3306

# centos修改主机名

hostnamectl set-hostname node1

需要记住的linux命令

# 自启动

centos加入自启动：启动命令添加到/etc/rc.d/rc.local

echo /usr/local/mysql/bin/mysqld\_safe --user=mysql& >> /etc/rc.d/rc.local

echo /xjgzbj/apache-tomcat-7.0.61/bin/startup.sh >> /etc/rc.d/rc.local

echo /usr/local/nginx/sbin/nginx >> /etc/rc.d/rc.local

# 添加环境变量：

方法一：vi ~/.bash\_profile

MYSQL\_HOME=/xjgzbj/mysql

export PATH=$PATH:$MYSQL\_HOME/bin

方法二：

export PATH=/usr/soft/jdk1.5.0\_06/bin:$PATH

export JAVA\_HOME=/usr/soft/jdk1.5.0\_06

# 系统信息

uptime 查询最后启动时间

关机命令：halt

----------------------------

# 常用脚本

## 定时执行脚本

|  |
| --- |
| 添加定时  vi /etc/crontab  　　添加一行  01 5 \* \* \* root /root/mysqlautobak    　　保存退出.  　　重新启动你的crond服务进程  # /etc/rc.d/init.d/crond restart  或者  sudo service crond restart |
| 定时删临时目录  #!/bin/bash  filename=`date -d "-1 hour" +%Y%m%d%H`  rm /tmp/mapreduce\_shell\_${filename}\* |
| 备份数据库脚本  #!/bin/bash  filename=`**date** +%Y-%m-%d\_%H\_%M`  mysqldump -uroot -paskserver zmhzx > /root/db\_backup/zmhzx\_${filename}.sql |
| crontab -e |

## killall 杀掉所有tomcat

|  |
| --- |
| #!/bin/bash  ID=`ps -ef|grep 'tomcat' |awk '{print $2}'`  echo ${ID}  for id in ${ID}  do  kill -9 ${id}  echo "killed ${id}"  done |