# 目录

[Linux常用命令 1](#_Toc457209054)

[shell快捷键 1](#_Toc457209055)

[/etc/hosts 文件 1](#_Toc457209056)

[查看系统多少位 2](#_Toc457209057)

[查看系统有没有安装jdk 2](#_Toc457209058)

[安装jdk1.6 2](#_Toc457209059)

[设置环境变量 4](#_Toc457209060)

[启动tomcat 内存溢出解决 5](#_Toc457209061)

[命令使用 6](#_Toc457209062)

[卸载安装包 6](#_Toc457209063)

[查看安装包 6](#_Toc457209064)

[卸载安装包 6](#_Toc457209065)

[安装文件 8](#_Toc457209066)

[bin文件安装： 8](#_Toc457209067)

[rpm包安装方式步骤： 8](#_Toc457209068)

[卸载 8](#_Toc457209069)

[查找文件 9](#_Toc457209070)

[查看一个程序是否运行 9](#_Toc457209071)

[终止进程 9](#_Toc457209072)

[查看文件，包含隐藏文件 9](#_Toc457209073)

[当前工作目录 10](#_Toc457209074)

[创建目录 10](#_Toc457209075)

[复制文件 10](#_Toc457209076)

[移动文件 11](#_Toc457209077)

[删除文件 11](#_Toc457209078)

[统计文本文件行数、字数、字符数 11](#_Toc457209079)

[重命名 12](#_Toc457209080)

[切换用户 12](#_Toc457209081)

[压缩文件 12](#_Toc457209082)

[列出压缩文件列表 12](#_Toc457209083)

[解压文件 12](#_Toc457209084)

[解压到指定目录 13](#_Toc457209085)

[查看文件内容 13](#_Toc457209086)

[查看文件头10行 14](#_Toc457209087)

[查看文件尾10行 14](#_Toc457209088)

[查看日志文件类型文件 14](#_Toc457209089)

[向下翻页 14](#_Toc457209090)

[使用超级管理员身份执行命令 14](#_Toc457209091)

[查看端口属于哪个程序 14](#_Toc457209092)

[free -m 查看内存使用和交换区使用量 15](#_Toc457209093)

[df -h查看各个分区使用情况 15](#_Toc457209094)

[du -sh 查看指定目录的大小 15](#_Toc457209095)

[/etc/sysconfig修改计算机名字 15](#_Toc457209096)

[重新配置IP 15](#_Toc457209097)

[设置系统时间 15](#_Toc457209098)

[文件权限管理 15](#_Toc457209099)

[取消权限 16](#_Toc457209100)

[添加权限 16](#_Toc457209101)

[关机/重启机器 17](#_Toc457209102)

[Word快键键操作 17](#_Toc457209103)

[快捷一 17](#_Toc457209104)

[快捷二 18](#_Toc457209105)

[快捷三 18](#_Toc457209106)

[快键四 19](#_Toc457209107)

[快捷五 20](#_Toc457209108)

[快捷六 20](#_Toc457209109)

# Linux常用命令

# Linux文件系统

当windows启动的时候，它会检测电脑已连接的所有硬盘上它能够识别的分区（NTFS、FAT32等），并自动为其分配盘符。  
这个“分配盘符”的过程，就是挂载（mount）过程（请一定记住这个mount）

“文件系统”，跟一般意义上的NTFS、EXT3文件系统不是一码事，别拿来对比哈！  
和windows的C盘D盘E盘的概念一样，这个“文件系统”或者叫“组织体系”是软件级别的，是linux系统“内在”的体系，就像你脑子里的一个“想法”，并无需依赖于任何的实际一样。

 linux的系统组织方式是——整个系统从根开始，按树形目录依次向下逐渐扩大，分类存放不同用途的文件，/读作“斜线”，英文slash；当其写作一个路径时，第一个/表示根，即root，其他的/表示路径分割符，但都读作“斜线”或“slash”。

用一个庸俗的比喻来说，“/”相当于c:\，而“/etc、/bin、/sbin、/lib”这些目录大概相当于c:\windows和c:\program files，“/home”相当于c:\Documents and Settings，而当你把第二分区挂载到“/mnt/partition2”的时候，这个目录就相当于d:\了，你地明白？

# 查询用户列表

# cat /etc/passwd

在一个命令行中可以执行多个命令，用分号把各个命令隔开即可，例如：   
＃last －x；halt

# Vi编辑命令

命令模式下：

u 回退上一步；

:q                      退出

:q!                     强制退出

:wq                   保存并退出

:x 保存并退出，假如文件没有修改，则文件时间不会变动

:set number     显示行号

:set nonumber  隐藏行号

/apache            在文档中查找apache 按n跳到下一个，shift+n上一个

yyp                   复制光标所在行，并粘贴

h(左移一个字符←)、j(下一行↓)、k(上一行↑)、l(右移一个字符→)

# shell快捷键

Ctrl + a 切换到命令行开始   
Ctrl + e 切换到命令行末尾   
Ctrl + l 清除屏幕内容   
Ctrl + u 清除剪切光标之前的内容   
Ctrl + k 剪切清除光标之后的内容   
Ctrl + y 复制剪切板里的内容   
Ctrl + h 类似于退格键   
Ctrl + r 在历史命令中查找 （这个非常好用，输入关键字就调出以前的命令了）   
Ctrl + c 终止命令   
Ctrl + d 退出shell   
Ctrl + z 转入后台运行..

# /etc/hosts 文件

Linux系统在向DNS[服务器](http://server.chinabyte.com/)发出域名解析请求之前会查询/etc/hosts文件，如果里面有相应的记录，就会使用hosts里面的记录。/etc/hosts文件通常里面包含这[一条](http://www.chinabyte.com/keyword/%E4%B8%80%E6%9D%A1/)记录

hosts文件格式是一行一条记录，分别是IP地址 hostname aliases，三者用空白字符分隔，aliases可选。

hosts文件的作用相当如DNS，提供IP地址到hostname的对应

/etc/hosts于设置hostname是没直接关系的，仅仅当你要在本机上用新的hostname来访问自己的时候才会用到/etc/hosts文件。两者没有必然的联系。

# 查看系统多少位

getconf LONG\_BIT

# 查看系统有没有安装jdk

**[root@HH005 usr]# rpm -qa |grep jdk**

jdk-1.6.0\_45-fcs.x86\_64

[root@HH005 usr]#

# 安装jdk1.6

**[root@HH005 java]# ls**

jdk-6u45-linux-x64-rpm.bin

**[root@HH005 java]# ls –l //查看权限**

total 67268

-rw-r--r--. 1 root root 68881069 Jun 2 00:56 jdk-6u45-linux-x64-rpm.bin

**[root@HH005 java]# chmod 777 jdk-6u45-linux-x64-rpm.bin //修改权限**

[root@HH005 java]# ls -l

total 67268

-rwxrwxrwx. 1 root root 68881069 Jun 2 00:56 jdk-6u45-linux-x64-rpm.bin

[root@HH005 java]# ./ jdk-6u45-linux-x64-rpm.bin

-bash: ./: is a directory

**[root@HH005 java]# ./jdk-6u45-linux-x64-rpm.bin //执行安装**

Unpacking...

Checksumming...

Extracting...

UnZipSFX 5.50 of 17 February 2002, by Info-ZIP (Zip-Bugs@lists.wku.edu).

inflating: jdk-6u45-linux-amd64.rpm

inflating: sun-javadb-common-10.6.2-1.1.i386.rpm

inflating: sun-javadb-core-10.6.2-1.1.i386.rpm

inflating: sun-javadb-client-10.6.2-1.1.i386.rpm

inflating: sun-javadb-demo-10.6.2-1.1.i386.rpm

inflating: sun-javadb-docs-10.6.2-1.1.i386.rpm

inflating: sun-javadb-javadoc-10.6.2-1.1.i386.rpm

Preparing... ########################################### [100%]

1:jdk ########################################### [100%]

Unpacking JAR files...

rt.jar...

jsse.jar...

charsets.jar...

tools.jar...

localedata.jar...

plugin.jar...

javaws.jar...

deploy.jar...

Installing JavaDB

Preparing... ########################################### [100%]

1:sun-javadb-common ########################################### [ 17%]

2:sun-javadb-core ########################################### [ 33%]

3:sun-javadb-client ########################################### [ 50%]

4:sun-javadb-demo ########################################### [ 67%]

5:sun-javadb-docs ########################################### [ 83%]

6:sun-javadb-javadoc ########################################### [100%]

Done.

[root@HH005 java]#

# 设置环境变量

设置环境变量

**[root@HH005 tomcat-7.0.29]# vi /etc/profile**

#set jdk environment

JAVA\_HOME=/usr/java/jdk1.6.0\_45

JRE\_HOME=/usr/java/jdk1.6.0\_45/jre

//$PATH:$JAVA\_HOME/bin:$JRE\_HOME/bin意思是追加在PATH后

//$JAVA\_HOME/bin:$JRE\_HOME/bin:$PATH意思是追加在PATH前

PATH=$PATH :$JAVA\_HOME/bin:$JRE\_HOME/bin

CLASSPATH=.:$JAVA\_HOME/lib/dt.jar:$JAVA\_HOME/lib/tools.jar:$JRE\_HOME/lib:

export JAVA\_HOME JRE\_HOME PATH CLASSPATHecho

**[root@HH005 tomcat-7.0.29]#resource /etc/profile //执行生效**

命令查看单个环境变量

**[root@HH005 etc]# echo $PATH**

/usr/lib64/qt-3.3/bin:/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin:/root/bin:/usr/lib/jvm/java-1.6.0-openjdk-1.6.0.0.x86\_64/bin

**[root@HH005 etc]# echo $CLASSPATH**

/usr/lib/jvm/java-1.6.0-openjdk-1.6.0.0.x86\_64/lib:/usr/lib/jvm/java-1.6.0-openjdk-1.6.0.0.x86\_64/jre/lib

[root@HH005 etc]#

删除环境变量

[root@HH005 etc]# unset JRE\_HOME JAVA\_HOME

启动tomcat

# sh startup.sh //启动

# sh shutdown.sh //关闭

# 启动tomcat 内存溢出解决

修改catalina.sh 文件

堆内存分配:

**-Xms　　 JVM**初始分配的内存由**-Xms**指定，**默认是物理内存的1/64**；

**-Xmx　 JVM**最大分配的内存由**-Xmx**指定，**默认是物理内存的1/4;**

默认空余堆内存小于**40%**时，**JVM**就会增大堆直到**-Xmx**的最大限制；空余堆内存大于**70%**时，**JVM**会减少堆直到**-Xms**的最小限制。

　　因此服务器一般设置**-Xms、-Xmx**相等以避免在每次**GC** 后调整堆的大小。

　　非堆内存分配

**-XX:PermSize　　　 JVM**使用**-XX:PermSize**设置非堆内存初始值，默认是物理内存的**1/64**；

　　-**XX:MaxPermSize**由**XX:MaxPermSize**设置最大非堆内存的大小，默认是物理内存的**1/4**。

**[root@HH005 ~]# locate catalina.sh** //查找定位catalina.sh 这个文件

/root/tomcat/tomcat-7.0.29/bin/catalina.sh

**[root@HH005 ~]# vi /root/tomcat/tomcat-7.0.29/bin/catalina.sh //编辑**

**查找并在**cygwin=false之前**添加红色的内容 vi 命令查找 /**cygwin

# OS specific support. $var \_must\_ be set to either true or false.

**#设置时用单个空格隔开**

JAVA\_OPTS="-server -Xms1024m -Xmx1024m -XX:PermSize=256m -XX:MaxPermSize=512m -XX:MaxNewSize=512m"

cygwin=false

darwin=false

os400=false

case "`uname`" in

CYGWIN\*) cygwin=true;;

Darwin\*) darwin=true;;

OS400\*) os400=true;;

# 命令使用

## 卸载安装包

### 查看安装包

# rpm -qa|grep java //查看jdk信息

java-1.6.0-openjdk-1.6.0.0-11.1.13.4.el6.x86\_64

tzdata-java-2014g-1.el6.noarch

java-1.7.0-openjdk-1.7.0.65-2.5.1.2.el6\_5.x86\_64

### 卸载安装包

[root@123 /]# yum -y remove java-1.7.0-openjdk-1.7.0.65-2.5.1.2.el6\_5.x86\_64

Loaded plugins: product-id, refresh-packagekit, security, subscription-manager

This system is not registered to Red Hat Subscription Management. You can use subscription-manager to register.

Setting up Remove Process

Resolving Dependencies

--> Running transaction check

---> Package java-1.7.0-openjdk.x86\_64 1:1.7.0.65-2.5.1.2.el6\_5 will be erased

--> Finished Dependency Resolution

Dependencies Resolved

==============================================================================================================================================================================================================================================================================

Package Arch Version Repository Size

==============================================================================================================================================================================================================================================================================

Removing:

java-1.7.0-openjdk x86\_64 1:1.7.0.65-2.5.1.2.el6\_5 @anaconda-RedHatEnterpriseLinux-201409260744.x86\_64/6.6 91 M

Transaction Summary

==============================================================================================================================================================================================================================================================================

Remove 1 Package(s)

Installed size: 91 M

Downloading Packages:

Running rpm\_check\_debug

Running Transaction Test

Transaction Test Succeeded

Running Transaction

Erasing : 1:java-1.7.0-openjdk-1.7.0.65-2.5.1.2.el6\_5.x86\_64 1/1

Verifying : 1:java-1.7.0-openjdk-1.7.0.65-2.5.1.2.el6\_5.x86\_64 1/1

Removed:

java-1.7.0-openjdk.x86\_64 1:1.7.0.65-2.5.1.2.el6\_5

Complete!

[root@123 /]# java -version

java version "1.6.0\_32"

OpenJDK Runtime Environment (IcedTea6 1.13.4) (rhel-11.1.13.4.el6-x86\_64)

OpenJDK 64-Bit Server VM (build 23.25-b01, mixed mode)

[root@123 /]#

## 安装文件

### bin文件安装：

如果你下载到的软件名是soft.bin，一般情况下是个可执行文件，安装方法如下：

引用:  
1、打开一个终端，su -成root用户；  
2、chmod +x soft.bin  
3、./soft.bin //运行这个命令就可以安装软件了

### rpm包安装方式步骤：

[root@localhost beinan]# rpm –initdb //**初始化rpm 数据库；**  
[root@localhost beinan]# rpm --rebuilddb 注：这个要花好长时间；

-i, --install                     install package(s)  
-v, --verbose                     provide more detailed output  
-h, --hash    print hash marks as package installs (good with v)  
-e, --erase                       erase (uninstall) package #卸载  
-U, --upgrade=<packagefile>+      upgrade package(s)  
－-replacepkge                    无论软件包是否已被安装，都强行安装软件包  
--test                            安装测试，并不实际安装  
--nodeps                          忽略软件包的依赖关系强行安装  
--force                           忽略软件包及文件的冲突

1、找到相应的软件包，比如soft.version.rpm，下载到本机某个目录；  
2、打开一个终端，su -成root用户；  
3、cd soft.version.rpm所在的目录；  
4、输入rpm -ivh soft.version.rpm 这是安装命令

### 卸载

**[root@hh05 jdk]# rpm -qa | grep jdk**

jdk-1.6.0\_45-fcs.x86\_64

jdk1.8.0\_91-1.8.0\_91-fcs.x86\_64

[root@hh05 jdk]# rpm -e jdk-8u91-linux-x64.rpm

error: package jdk-8u91-linux-x64.rpm is not installed

**[root@hh05 jdk]# rpm -e jdk1.8.0\_91-1.8.0\_91-fcs.x86\_64 //卸载 rpm –e 包名**

[root@hh05 jdk]# rpm -qa | grep jdk

jdk-1.6.0\_45-fcs.x86\_64

[root@hh05 jdk]#

## 查找文件

Find / -name filename.txt 根据文件名称查找/目录下的 filename.txt文件

locate filename

查找内容

grep 关键字 file

## 查看一个程序是否运行

ps –ef|grep tomcat 查看所有有关tomcat 的进程

## 终止进程

Kill -9 19979 终止线程号为19979的进程 // -9 代表终止

Pkill -9 java 批量终止java 进程

## 查看文件，包含隐藏文件

ls –all

## 当前工作目录

pwd

## 创建目录

-v 显示创建过程信息

# mkdir -v ubuntu redhat slackware //同时创建多个文件夹

递归创建子目录 –p参数递归创建

**[root@HH005 tomcat]# mkdir -vp java jdk/java jdk/lib**

mkdir: created directory `jdk'

mkdir: created directory `jdk/java'

mkdir: created directory `jdk/lib'

**[root@HH005 tomcat]# ls**

java jdk

**[root@HH005 tomcat]# ls jdk**

java lib

## 复制文件

cp source dest 复制文件

cp –r sourceFolder targetFolder 递归复制整个文件夹

scp sourceFile remoteUserName@remoteIp:remoteAddres 远程拷贝

例如:

[root@cdh02 local]# scp apache-tomcat-7.0.29.tar.gz root@10.18.13.152:/root

root@10.18.13.152's password:

apache-tomcat-7.0.29.tar.gz 100% 7499KB 7.3MB/s 00:00

[root@cdh02 local]# ls

## 移动文件

mv / temp/movefile /targetFolder

## 删除文件

直接rm就可以了，不过要加两个参数-rf 即：rm -rf 目录名字  
-r 就是向下递归，不管有多少级目录，一并删除  
-f 就是直接强行删除，不作任何提示的意思

rm –rf tomcat-7.0.59/

清空 **jdk**目录文件

**[root@HH005 tomcat]# rm -vrf jdk/\***

removed directory: `jdk/java'

removed directory: `jdk/lib'

**[root@HH005 tomcat]# ls**

java jdk

**[root@HH005 tomcat]# ls jdk //该目录下已经没有文件**

[root@HH005 tomcat]#

## 批量删除文件夹

[root@hh05 ftpReport]# ls

cron.log dataSource.properties day-201601 day-201602 day-201603 day-201604 day-201605 day-201606 dhjhour.sh log4j.properties report.jar test

[root@hh05 ftpReport]# rm -irfv day-20160[1-6]

removed directory: `day-201601'

removed directory: `day-201602'

removed directory: `day-201603'

removed directory: `day-201604'

removed directory: `day-201605'

removed directory: `day-201606'

## 统计文本文件行数、字数、字符数

**[root@hh05 tomcat6.0]# wc a.txt**

1 3 12 a.txt

**[root@hh05 tomcat6.0]# cat a.txt**

how are you

**[root@hh05 tomcat6.0]#**

## 重命名

mv olderNameFile newNameFile

## 切换用户

su -userName

## 压缩文件

tar –czf jpg.tar.gz \*.jpg   //将目录里所有jpg文件打包成jpg.tar后，并且将其用gzip压缩，生成一  个gzip压缩过的包，命名为jpg.tar.gz

tar –cjf jpg.tar.bz2 \*.jpg //将目录里所有jpg文件打包成jpg.tar后，并且将其用bzip2压缩，生成一个bzip2压缩过的包，命名为jpg.tar.bz2

tar -czf test.tar.gz /test1 /test2

zip -r test.zip /test

## 列出压缩文件列表

tar -tzf test.tar.gz

## 解压文件

-c: 建立压缩档案

-x：解压  
-t：查看内容  
-r：向压缩归档文件末尾追加文件  
-u：更新原压缩包中的文件

-z：有gzip属性的  
-j：有bz2属性的  
-Z：有compress属性的  
-v：显示所有过程  
-O：将文件解开到标准输出

-f: 使用档案名字，切记，这个参数是最后一个参数，后面只能接档案名

tar - xvzf test.tar.gz

upzip test.zip

tar –xvf file.tar //解压 tar包

tar -xzvf file.tar.gz //解压tar.gz

tar -xjvf file.tar.bz2 //解压 tar.bz2

tar –xZvf file.tar.Z //解压tar.Z

unrar e file.rar //解压rar

unzip file.zip //解压zip

unzip mydata.zip -d mydatabak // –d 参数解压到指定目录里

查看把/home目录下面的wwwroot.zip里面的内容  
unzip -v wwwroot.zip //加-v参数查看

查看文件是否完整

# unzip -t compressFile.zip //加 –t参数

压缩a.txt ,b.txt文件 -r参数

# zip -r ab.zip a.txt b.txt

### 解压到指定目录

[root@hh05 jdk]# ls

jdk jdk1.7.0\_07 jdk-6u45-linux-x64-rpm.bin jdk-7u7-linux-i586.tar.gz jdk-7u80-linux-x64.rpm jdk-8u91-linux-x64.rpm

**[root@hh05 jdk]# tar -zxvf jdk-7u7-linux-i586.tar.gz -C jdk/**

**[root@hh05 jdk]# ls jdk**

jdk1.7.0\_07

## 查看文件内容

less(选项)(参数)  
  
-e：文件内容显示完毕后，自动退出；

-f：强制显示文件；

-g：不加亮显示搜索到的所有关键词，仅显示当前显示的关键字，以提高显示速度；

-l：搜索时忽略大小写的差异；

-N：每一行行首显示行号；

-s：将连续多个空行压缩成一行显示；

-S：在单行显示较长的内容，而不换行显示；

-x<数字>：将TAB字符显示为指定个数的空格字符。

### 查看文件头10行

head -n 10 example.txt

### 查看文件尾10行

tail –n 10 example.txt

## 查看日志文件类型文件

tail -f example.log //这个命令会自动显示新增内容，屏幕只显示10行内容的（可设置）

## 向下翻页

Ctrl f 向下滚动一屏；

空格键 向下滚动一屏；

Ctrl b 返回上一屏

## 使用超级管理员身份执行命令

sudo rm a.txt 使用管理员身份删除文件

## 查看端口属于哪个程序

Isof -i :8080

## free -m 查看内存使用和交换区使用量

#free -m

## df -h查看各个分区使用情况

#df -h

## du -sh 查看指定目录的大小

du -sh /mnt/online/

## /etc/sysconfig修改计算机名字

如果要永久修改RedHat的hostname，就修改/etc/sysconfig/network文件，将里面的HOSTNAME这一行修改成HOSTNAME=NEWNAME，其中NEWNAME就是你要设置的hostname。

## 重新配置IP

#dhclient

## 设置系统时间

**[root@HH005 tomcat6.0]# date -s "2016-6-6 9:10"**

Mon Jun 6 09:10:00 CST 2016

## 文件权限管理

三种基本权限

R           读         数值表示为4

W          写         数值表示为2

X           可执行  数值表示为1

文件属主 u

文件属组 g

其他人 o

+ 添加某个权限。  
  
- 取消某个权限。

# chown yanghuazhen workFile/ 改变文件夹权限所有者

### 取消权限

**[root@hh05 jdk]# ls –l //查看权限**

total 156412

-rwxr-xr-x. 1 root root 160162581 Jun 6 15:47 jdk-8u91-linux-x64.rpm

**[root@hh05 jdk]# chmod u-x,g-x,o-x jdk-8u91-linux-x64.rpm //修改权限**

**[root@hh05 jdk]# ls –l //查看**

total 156412

-rw-r--r--. 1 root root 160162581 Jun 6 15:47 jdk-8u91-linux-x64.rpm

[root@hh05 jdk]#

### 添加权限

**[root@hh05 jdk]# chmod u+x,g+x,o+x jdk-8u91-linux-x64.rpm**

[root@hh05 jdk]# ls -l

total 156412

-rwxr-xr-x. 1 root root 160162581 Jun 6 15:47 jdk-8u91-linux-x64.rpm

**[root@hh05 jdk]#**

## 关机/重启机器

shutdown

     -r             关机重启

     -h             关机不重启

     now          立刻关机

halt               关机

reboot          重启

# ftp服务器配置：

# vi /etc/vsftpd/vsftpd.conf

anonymous\_enable=NO

local\_umask=022

anon\_upload\_enable=YES

dirmessage\_enable=YES

connect\_from\_port\_20=YES

idle\_session\_timeout=600

ascii\_upload\_enable=YES

ascii\_download\_enable=YES

userlist\_deny=NO

userlist\_file=/etc/vsftpd/user\_list

/etc/vsftpd/vsftpd.conf   //主配置文件

/etc/vsftpd/ftpusers      //被禁止登录FTP的用户文件

/etc/vsftpd/user\_list     //允许登录FTP的用户文件

创建用户目录文件夹

#useradd –d /home/test yanghuazhen

#passwd yanghuazhen //添加密码

#echo yanghuazhen>>/etc/vsftpd/user\_list //添加用户权限

## 远程登录Ftp:

#sftp username@ip

SFTP是Secure File Transfer Protocol的缩写，安全文件传送协议。

要使用JSch，需要下载它的jar包，请从官网下载它：<http://www.jcraft.com/jsch/>

# Linux添加用户

#useradd yhzh

#passwd yhzh //给用户yhzh添加密码

限制IP 访问FTP:

#vi /etc/hosts.allow

vsftpd:192.168.5.128:DENY 设置该IP地址不可以访问ftp服务

FTP 访问时间限制：

#cp /usr/share/doc/vsftpd-1.1.3/vsftpd.xinetd /etc/xinetd.d/vsftpd

#vi /etc/xinetd.d/vsftpd/

修改 disable = no

access\_time = hour:min-hour:min (添加配置访问的时间限制(注：与vsftpd.conf中listen=NO相对应)

例: access\_time = 8:30-11:30 17:30-21:30 表示只有这两个时间段可以访问ftp

# 修改防火墙

# vi /etc/sysconfig/iptables

//添加可访问端口

-A INPUT -m state --state NEW -m tcp -p tcp --dport 22 -j ACCEPT

-A INPUT -m state --state NEW -m tcp -p tcp --dport 21 -j ACCEPT

-A （–append): 该命令会把一条规则附件到chain的末尾。

-D（–delete)用来删除某个规则。

-F （–flush) 如果指定了chain, 删除该chain中的所有规则，如果未指定chain, 则删除所有chain中的所有规则。

target: 是由规则指定的操作。 包括下面几种：

ACCEPT: 接收信息包(允许它前往目的地），并且将停止遍历chain.

DROP：  拒绝，

此外还有REJECT, RETURN, LOG, REDIRECT, MARK, MIRROR, MAQUERADE等。

查看防火墙状态：

#service iptables status

# 设置开机启动某些服务程序

如果需要自启动某些服务，只需使用chkconfig 服务名 on即可，若想关闭，将on改为off

# *查看当前目录占用空间大小*

# du –sh /

# 查看目录下所有各个文件大小

# du -kh /FtpReport/day/day-201703/\*

# 查看磁盘空间使用情况

#df –kh

# 查看目录挂载点

#df 目录名/ -kh -----df . –kh

# Linux错误处理：

[java项目打包成jar文件，在linux上执行出现 Invalid or corrupt jarfile XXXX.jar 错误](http://blog.csdn.net/camel113/article/details/7403496)

 确保文件传输到[**Linux**](http://lib.csdn.net/base/linux)上的时候未损坏，在ftp上传时选在 Binary 模式上传

# 任务调度

## cron表达式

Seconds:可出现", - \* /"四个字符，有效范围为0-59的整数   
Minutes:可出现", - \* /"四个字符，有效范围为0-59的整数   
Hours:可出现", - \* /"四个字符，有效范围为0-23的整数   
DayofMonth:可出现", - \* / ? L W C"八个字符，有效范围为0-31的整数   
Month:可出现", - \* /"四个字符，有效范围为1-12的整数或JAN-DEc   
DayofWeek:可出现", - \* / ? L C #"四个字符，有效范围为1-7的整数或SUN-SAT两个范围。1表示星期天，2表示星期一， 依次类推 {周一：2，周二：3，周三：4，周四：5，周五：6，周六：7，周天：1}  
Year:可出现", - \* /"四个字符，有效范围为1970-2099年

\* \* \* \* \* /command path

0 2 \* \* \* /u01/test.sh >/dev/null 2>&1 &

这句话的意思就是在后台执行这条命令，并将错误输出2重定向到标准输出1，然后将标准输出1全部放到/dev/null 文件，也就是清空。

0表示键盘输入

1表示标准输出

2表示错误输出

2>&1 是将错误输出重定向到标准输出。 然后将标准输入重定向到文件out.file。

&1 表示的是文件描述1，表示标准输出，如果这里少了&就成了数字1，就表示重定向到文件1。

& ：后台执行

前5个字段分别表示：

分钟：0-59

小时：1-23

日期：1-31

月份：1-12

星期：0-6（0表示周日）

### ", - \* /? L W C #"字符解释

,:表示列出枚举值值。例如：在Minutes域使用5,20，则意味着在5和20分每分钟触发一次。

-:表示范围，例如在Minutes域使用5-20，表示从5分到20分钟每分钟触发一次

\*：表示匹配该域的任意值

/：表示起始时间开始触发，然后每隔固定时间触发一次，例如在Minutes域使用5/20,则意味着5分钟触发一次，而25，45等分别触发一次

?:只能用在DayofMonth和DayofWeek两个域。它也匹配域的任意值，但实际不会。

L:表示最后，只能出现在DayofWeek和DayofMonth域，如果在DayofWeek域使用5L,意味着在最后的一个星期四触发。

W:表示有效工作日(周一到周五),只能出现在DayofMonth域，另外一点，W的最近寻找不会跨过月份

LW:这两个字符可以连用，表示在某个月最后一个工作日，即最后一个星期五。

#:用于确定每个月第几个星期几，只能出现在DayofMonth域。例如在4#2，表示某月的第二个星期三。

### 语法：

字符串以5或6个空格隔开，分为6或7个域，每一个域代表一个含义

Seconds Minutes Hours DayofMonth Month DayofWeek Year或Seconds Minutes Hours DayofMonth Month DayofWeek

秒 分 时 天 月 星期几 年 （7个域）

秒 分 时 天 月 星期几 （6个域）

秒（0~59）   
分钟（0~59）   
小时（0~23）   
天（月）（0~31，但是你需要考虑你月的天数）   
月（0~11）   
天（星期）（1~7 1=SUN 或 SUN，MON，TUE，WED，THU，FRI，SAT）   
年份（1970－2099）

**注：由于"月份中的日期"和"星期中的日期"这两个元素互斥的,必须要对其中一个设置” ?”**

**“？”字符仅被用于天（月）和天（星期）两个子表达式，表示不指定值**

### 例子：

"0 15 10 \* \* ?" 每天上午10:15触发   
"0 15 10 \* \* ? \*" 每天上午10:15触发

"0 0/5 14 \* \* ?" 在每天下午2点到下午2:55期间的每5分钟触发   
"0 0/5 14,18 \* \* ?" 在每天下午2点到2:55期间和下午6点到6:55期间的每5分钟触发

"0 0-5 14 \* \* ?" 在每天下午2点到下午2:05期间的每1分钟触发

"0 15 10 ? \* MON-FRI" 周一至周五的上午10:15触发   
"0 15 10 15 \* ?" 每月15日上午10:15触发

"0 15 10 L \* ?" 每月最后一日的上午10:15触发

"0 15 10 ? \* 6L" 每月的最后一个星期五上午10:15触发

"0 15 10 ? \* 6#3" 每月的第三个星期五上午10:15触发

"0 10,44 14 ? 3 WED" 每年三月的星期三的下午2:10和2:44触发

## 公司实例

31,32,33 \* \* \* \* /ftpReport/dhjhour.sh > /dev/null 2>&1 每个小时的 31,32,33,分触发

\*/1 \* \* \* \* /ftpReport/dhjhour.sh > /dev/null 2>&1 每间隔1分钟触发一次

## Cron的启动与关闭

由于Cron是Linux的内置服务，可以用以下的方法启动.关闭这个服务:

/sbin/service crond start           //启动服务  
/sbin/service crond stop            //关闭服务  
/sbin/service crond restart        //重启服务  
/sbin/service crond reload         //重新载入配置

## 查看任务

#crontab –l

查看cron 执行日志

# tail -n 200 /var/log/cron

## 编辑任务

#crontab –e

|  |
| --- |
| #dhj  00 09 \* \* \* /usr/sanss/report/dhjday.sh > /dev/null 2>&1  15 09 \* \* \* /usr/sanss/report/dhj24h.sh > /dev/null 2>&1  00 04 1 \* \* /usr/sanss/report/dhjmonth.sh > /dev/null 2>&1  05 \* \* \* \* /usr/sanss/report/dhjhour.sh > /dev/null 2>&1  20 00 \* \* 5 /usr/sanss/report/dhjweek.sh > /dev/null 2>&1  #dhj |

## 写 .sh文件

#vi dhjday.sh

|  |
| --- |
| #!/bin/bash  cd /usr/sanss/report //切换到文件所在目录  java -jar report.jar dhjday |

## 修改文件权限：添加执行权限

#chmod u+x,g+x,o+x dhjhour.sh

# 远程文件下载上传

1，你用winSCP软件连接到远程服务器操作文件上传上传

2.跳板机连接远程服务器，查看数据库

# 安装版本库svn

## 安装svn #yum -y install subversion

## 建立版本库根目录

[root@hh05 home]# mkdir /home/svn

## 建立版本库 创建一个项目

[root@hh05 home]# svnadmin create /home/svn/repos //会生成以下红色文件

[root@hh05 home]# cd svn/repos/

[root@hh05 repos]# ls

conf db format hooks locks README.txt

## 配置运行用户访问

# cd /home/svn/repos/conf/

# ls

authz passwd svnserve.conf

#vi svnserve.conf //配置

anon-access = none

auth-access = write

password-db = passwd

authz-db = authz

# vi passwd

[users]

# harry = harryssecret

# sally = sallyssecret

yhzh = 123456

[root@hh05 conf]# vi authz //修改权限

[groups]

# harry\_and\_sally = harry,sally

# harry\_sally\_and\_joe = harry,sally,&joe

admin = yhzh

# [/foo/bar]

# harry = rw

# &joe = r

# \* =

# [repository:/baz/fuz]

# @harry\_and\_sally = rw

# \* = r

[repos:/]

@admin = rw

## 运行svn服务器

#svnserve -d -r /home/svn/

# ps -ef |grep svn

root 10147 1 0 01:24 ? 00:00:00 svnserve -d -r /home/svn/

root 10487 3181 0 02:09 pts/2 00:00:00 grep svn

[root@hh05 conf]#

## 修改防火墙

允许对 3690端口的访问

#vi /etc/sysconfig/iptables

# Firewall configuration written by system-config-firewall

# Manual customization of this file is not recommended.

\*filter

:INPUT ACCEPT [0:0]

:FORWARD ACCEPT [0:0]

:OUTPUT ACCEPT [0:0]

-A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT

-A INPUT -p icmp -j ACCEPT

-A INPUT -i lo -j ACCEPT

-A INPUT -m state --state NEW -m tcp -p tcp --dport 22 -j ACCEPT

-A INPUT -m state --state NEW -m tcp -p tcp --dport 3690 -j ACCEPT

-A INPUT -j REJECT --reject-with icmp-host-prohibited

-A FORWARD -j REJECT --reject-with icmp-host-prohibited

COMMIT

## 搭建http svn服务器

1. 在 /etc/httpd/modules下面 有两个 mod 关于 svn的文件

LoadModule dav\_module modules/mod\_dav.so

LoadModule dav\_svn\_module modules/mod\_dav\_svn.so

### **一、关闭SELINUX**

vi /etc/selinux/config

#SELINUX=enforcing #注释掉

#SELINUXTYPE=targeted #注释掉

SELINUX=disabled #增加

[root@hh05 /]#setenforce 0 #使配置立即生效

### **二、开启防火墙端口**

系统运维  www.osyunwei.com  温馨提醒：系统运维原创内容©版权所有,转载请注明出处及原文链接

基于Apache的http模式，默认端口为80

基于svnserve的独立服务器模式，默认端口为3690

vi /etc/sysconfig/iptables #编辑防火墙配置文件

开放 80端口

-A INPUT -m state --state NEW -m tcp -p tcp --dport 80 -j ACCEPT

### **三、安装Apache**

[root@hh05 /]# yum install httpd apr apr-util httpd-devel

[root@hh05 /]# yum install mod\_dav\_svn #安装基于Apache的http模式访问的支持模块

[root@hh05 /]# httpd –version  #查看Apache版本信息

chkconfig httpd on #设置开机启动

service httpd start #启动Apache

### 安装svn

yum install subversion #使用yum命令在线安装

svnserve --version #查看svn版本信息

创建svn版本库

mkdir -p /home/svn #创建svn版本库存放目录

cd /home/svn #进入目录

svnadmin create /home/svn/project1 #创建svn版本库project1

svnadmin create /home/svn/project2 #创建svn版本库project2

2、设置配置文件

mkdir -p /home/svn/conf #创建配置文件目录

#cd /home/svn/conf

#cp /home/svn/project1/conf/passwd ./ #拷贝账号密码配置文件模板

#cp /home/svn/project1/conf/authz ./ #拷贝目录权限配置文件模板

#cp /home/svn/project1/conf/ svnserve.conf  ./#拷贝全局配置文件模板

#vi /home/svn/conf/passwd #编辑，添加以下代码

[users]

# harry = harryssecret

# sally = sallyssecret

osyunwei=123456

osyunwei1=123456

vi /home/svn/conf/authz #编辑，添加以下代码

[groups]

admin = osyunwei

project1 = osyunwei1

project2 = osyunwei2

project3 = osyunwei3

[/]

@admin = rw

\* =

[project1:/]

@admin = rw

@project1 = rw

\* =

[project2:/]

@admin = rw

@project2 = rw

\* =

#vi /home/svn/conf/svnserve.conf #配置全局文件，在最后添加以下代码

[general]

anon-access=none #禁止匿名访问，设置为none。默认为read，参数：read,write,none

auth-access=write #授权用户写权限

password-db=/home/svn/conf/passwd #用户账号密码文件路径，可以写绝对路径

authz-db=/home/svn/conf/authz #访问控制权限文件路径，可以写绝对路径

realm=svn #每个SVN项目的认证命，会在认证提示里显示，建议写项目名称。

:wq! #保存退出

3、启动SVN

svnserve -d -r /home/svn --config-file /home/svn/conf/svnserve.conf --listen-port 3690

#--config-file后面跟全局配置参数文件

### 1.修改文件，配置

#vi /etc/init.d/svn #编辑 添加以下代码

/etc/init.d下面放的是系统启动时要启动的服务

|  |
| --- |
| #!/bin/sh  # chkconfig: 2345 85 85  # processname: svn  svn\_bin=/usr/local/svn/bin  svn\_port=3690  svn\_home=/home/svn  svn\_config=/home/svn/conf/svnserve.conf  if [ ! -f "$svn\_bin/svnserve" ]  then  echo "svnserver startup: cannot start"  exit  fi  case "$1" in  start)  echo "Starting svnserve..."  $svn\_bin/svnserve -d -r $svn\_home --config-file $svn\_config --listen-port $svn\_port  echo "Successfully!"  ;;  stop)  echo "Stoping svnserve..."  killall svnserve  echo "Successfully!"  ;;  restart)  $0 stop  $0 start  ;;  \*)  echo "Usage: svn { start | stop | restart } "  exit 1  esac |

chmod +x /etc/init.d/svn #添加执行权限

chkconfig svn on #开机自启动

[root@hh05 conf]# svnserve -d -r /home/svn --config-file /home/svn/conf/svnserve.conf --listen-port 3690 #启动

### 2、设置Apache配置文件

vi /etc/httpd/conf.d/subversion.conf #编辑，在最后添加以下代码

|  |
| --- |
| <Location /svn>  DAV svn  #SVNPath /home/svn  SVNParentPath /home/svn  # # Limit write permission to list of valid users.  # <LimitExcept GET PROPFIND OPTIONS REPORT>  # # Require SSL connection for password protection.  # # SSLRequireSSL  #  AuthType Basic  AuthName "Authorization SVN"  AuthzSVNAccessFile /home/svn/conf/authz  AuthUserFile /home/svn/conf/http\_passwd  Require valid-user  # </LimitExcept>  </Location>  :wq! #保存退出 |

### 配置svn 支持http访问

#### 1、创建账号密码认证文件

#htpasswd -cm /home/svn/conf/http\_passwd yhzh

#htpasswd -m /home/svn/conf/http\_passwd yhzh2

**注意：**

/home/svn/conf/目录下面passwd文件是svnserve独立服务器使用的认证文件，密码没有加密，明文显示。

/home/svn/conf/目录下面http\_passwd文件是Apache的http模式使用的认证文件，密码使用MD5加密。

passwd和http\_passwd文件中，账号密码必须设置相同。

1、Apache htpasswd命令选项参数说明

-c 创建一个加密文件

-n 不更新加密文件，只将apache htpasswd命令加密后的用户名密码显示在屏幕上

-m 默认apache htpassswd命令采用MD5算法对密码进行加密

-d apache htpassswd命令采用CRYPT算法对密码进行加密

-p apache htpassswd命令不对密码进行进行加密，即明文密码

-s apache htpassswd命令采用SHA算法对密码进行加密

-b 在apache htpassswd命令行中一并输入用户名和密码而不是根据提示输入密码

-D 删除指定的用户

2、SVNPath 与 SVNParentPath区别:

SVNParentPath：支持多个相同父目录的SVN版本库。

SVNPath：只支持一个主目录的SVN版本库，如果在主目录下面建新项目,则提示无权访问。

#### 2、设置Apache配置文件

修改目录权限

chown apache:apache /home/svn -R #设置svn目录所有者为Apache服务运行账号apache

### 关闭svn

[root@hh05 svn]# killall svnserve

## 查看端口

[root@hh05 svn]# netstat -ln|grep 3690

# Word快键键操作

## 快捷一

**ctrl+shift+N 降级为正文**

**ctrl+alt+(1-6) 升级标题**

**Alt+shift+down 在大纲内下移项目**

**Alt+shift+up 在大纲内上移项目**

ctrl+a 全选

**ctrl+b 粗体**

ctrl+c 拷贝

ctrl+d 字体格式

ctrl+e 居中对齐

**ctrl+f 查找**

**ctrl+g 定位**

**ctrl+h 替换**

ctrl+i 斜体

ctrl+j 两端对齐

ctrl+k 超级链接

## 快捷二

**ctrl+l 左对齐**

ctrl+m 左缩进

ctrl+n 新建

ctrl+o 打开

ctrl+p 打印

**ctrl+r 右对齐**

ctrl+s 保存

ctrl+t 首行缩进

ctrl+u 下划线

ctrl+v 粘贴

ctrl+w 写盘关闭

ctrl+x 剪切

ctrl+y 重复

ctrl+z 撤消

## 快捷三

ctrl+0 段前6P切换

**ctrl+1 单倍行距**

ctrl+2 双倍行距

ctrl+3 锁定

ctrl+5 1.5倍行距

ctrl+= 下标和正常切换

**ctrl+shift+a 大写**

ctrl+shift+b 粗体

ctrl+shift+c 格式拷贝

ctrl+shift+d 分散对齐

ctrl+shift+e 修订

**ctrl+shift+f 定义字体**

ctrl+shift+H 应用隐藏格式 ctrl+shift+I 斜体

ctrl+shift+K 小型大写字母 ctrl+shift+L

 列表样式 ctrl+shift+M 减少缩进

## 快键四

ctrl+shift+P 定义字符大小

ctrl+shift+Q symbol字体

**ctrl+shift+S 定义样式**

ctrl+shift+T 减小首行缩进

**ctrl+shift+U 下划线**

ctrl+shift+V 格式粘贴

ctrl+shift+W 只给词加下划线

ctrl+shift+Z 默认字体样式

ctrl+shift+= 上标与正常切换

**Alt+shift+a 显示所有标题**

Alt+shift+C 关闭预览 Alt+shift+D 插入日期

## 快捷五

Alt+shift+E 编辑邮件合并数据

Alt+shift+F 插入合并域

Alt+shift+K 预览邮件合并

Alt+shift+L 显示

Alt+shift+M 打印已合并文档

Alt+shift+N 合并文档

Alt+shift+O 标记目录项

**Alt+shift+P 插入页码**

Alt+shift+R 页眉页脚链接

Alt+shift+T 时间 Alt+shift+U 更新域

## 快捷六

Alt+shift+X 标记index entry

Alt+shift+1 显示“标题1”

Alt+shift+2 显示“标题2”

Alt+shift+3 显示“标题3”

 Alt+shift+5 应用“标题3”

**ctrl+alt+c 版权copyright符号**

ctrl+alt+E 插入尾注

 ctrl+alt+F 插入脚注

ctrl+alt+I 预览

ctrl+alt+K 自动套用格式

ctrl+alt+L listnum域

ctrl+alt+M 批注 动词margin

ctrl+alt+N 普通视图

**ctrl+alt+O 大纲视图**

 ctrl+alt+P 页面视图

ctrl+alt+R 注册商标register

ctrl+alt+S 拆分窗口

ctrl+alt+T 商标trademarkrm

ctrl+alt+U 更新表格格式

ctrl+alt+V 插入自动图文集