

重定向符:

>: 把屏幕输出的内容重定向覆盖到一个文件中, 文件原有的内容会丢失

>>: 把屏幕输出的内容重定向追加到一个文件中, 会在文件末尾添加进去

Echo hello >a.txt(hello)    Echo world >>a.txt(hello world)

Vi 编辑器:

类似 windows 中的记事本程序, 它能编辑普通的字符文件

B. txt ,在里面写入 5 行内容 并且进行保存。

Vi 文件名, 打开一个文件进行编辑 vi 刚打开文件的时候, 进入的是命令模式, 输入 `!a\o\s` 中的其中一个, 可以进行编辑模式 -- 输入完内容后 -- 按 `esc` 键切换到命令模式 -- `:wq` 保存并退出

vi: 编辑模式 、命令模式

`!a\o\s`: i 当前位置插入 a 在当前位置的下一个位置插入 o 换行插入 s 删除当前字符插入

`:q` 退出

`:wq` 保存退出

`:q!` 不保存退出

`:set nu` 查看行号 `:set nonu` 取消行号 `:n` 跳到第几行 `:$`跳到文件末尾行

/搜索的内容 往下搜索 n 表示下一个 N 表示上一个

`dd` 表示删除当前光标行

`rm -rf` 命令: -r 表示删除目录 -f 删除不提醒 sudo rm -rf / 尽量慎用

用户及权限命令:

假设: 分为两个用户组 one two

每个组里两个用户 one: xiaobai xiaohei two: dapang daliu

Groupadd 组名 添加用户组

```
[root@192 liusir]# groupadd one
```

useradd 用户名 -g 组名 创建用户并分配组 passwd 用户名 设置密码

```
[root@192 liusir]# useradd xiaobai -g one
[root@192 liusir]# passwd xiaobai
```

ll //ls -l 显示文件详情

```
[root@192 liusir]# ls -l b.txt
-rw-r--r--. 1 root root 46 Oct 27 19:17 b.txt
```

前 10 个字符:

第一个字符 表示文件类型 - 普通文件 d 表示文件夹 l 表示链式文件 (快捷方式)

第 2-9 字符: 表示文件权限 分为三组: rwx rwx rwx(读写执行) 如果是 - 表示没有该权限

第一组表示所有者的权限(u) 第二组表示所属组用户的权限(g) 第三组表示其它用户权限

(o)

Root root :第一个 root 代表用户名 及文件的所有者 ;第二个 root 代表用户组, 及文件的所属组

46 表示文件大小 , 46 个字节

Oct 27 : 文件的创建时间

B.txt 文件名

su - 用户名: 切换账号并同时切换用户的环境变量

Chmod == 修改文件权限的命令

两种写法 :

写法一: chmod u+x b.txt chmod (u/g/o)+x b.txt === +x +rwx -rwx

+增加权限 - 取消权限

```
[root@192 xiaobai]# chmod u+x b.txt
[root@192 xiaobai]# ls -l
total 4
-rwxr--r--. 1 xiaobai one 6 Oct 27 20:33 b.txt
```

```
[root@192 xiaobai]# chmod o+wx b.txt
[root@192 xiaobai]# ls -l
total 4
-rwxr--rwx. 1 xiaobai one 6 Oct 27 20:33 b.txt
```

写法二: 用数字描述

1 表示有权限 0 表示没权限

chmod 777 b.txt 给所有用户都有读写执行的权限

7 -- 111 -- rwx

634: u 6--110 -- rw- ; g 3 - 011 -- -wx ; o 4 -- 100 r--

chgrp 组名 文件名: 修改文件的所属组

chown 用户名[:组名] 文件名: 修改文件的所有者[组名]

```
[root@192 xiaobai]# chgrp two b.txt
[root@192 xiaobai]# ls -l
total 4
-rwxrwxrwx. 1 xiaobai two 6 Oct 27 20:33 b.txt
[root@192 xiaobai]#
[root@192 xiaobai]#
[root@192 xiaobai]# chown dapang b.txt
[root@192 xiaobai]# ls -l
total 4
-rwxrwxrwx. 1 dapang two 6 Oct 27 20:33 b.txt
[root@192 xiaobai]# chown xiaobai:one b.txt
[root@192 xiaobai]# ls -l
total 4
-rwxrwxrwx. 1 xiaobai one 6 Oct 27 20:33 b.txt
```

压缩及解压缩命令:

常用的压缩格式: .tar .tar.gz .zip .rar

tar 包压缩及解压:

tar -cf 压缩文件名.tar 要压缩的文件夹或文件 : 压缩 -c 压缩 -f 指定压缩文件名

`tar -xf` 压缩文件名 : 解压缩

gz 包的压缩及解压: 压缩的时候会删除源文件

gzip 要压缩的文件名 : 压缩

gzip -d 压缩文件名 : 解压 会把 gz 的压缩包删除

特殊写法:

`tar -czvf` 压缩的文件名.tar.gz 要压缩的文件夹或文件 : 压缩 -z 表示压缩成 gz 格式

`tar -xzf` 压缩的文件名.tar.gz : 解压缩 -v 显示压缩或解压过程

zip 包压缩及解压:

zip 压缩的文件名.zip 压缩文件 : 压缩

unzip 压缩的文件名.zip : 解压

```
[root@192 xiaobai]# zip a.zip a.txt
adding: a.txt (stored 0%)
```

安装 jdk 步骤: (绿色安装)

第一步: 上传 jdk 安装包到 linux 服务器的 /opt 目录

第二步: 使用 命令解压 jdk 安装包

```
[root@192 opt]# tar -xzf jdk-linux-x64.tar.gz
```

第三步: 配置环境变量

用户环境变量文件

```
[root@192 opt]# vi ~/.bash_profile
```

系统环境变量文件

```
[root@192 opt]# vi /etc/profile
```

上面两个环境变量配置其中一个即可,都是到文件末尾增加三行

```
export JAVA_HOME=/opt/jdk1.8.0_131
export CLASSPATH=$JAVA_HOME/lib/dt.jar:$JAVA_HOME/lib/tools.jar
export PATH=$PATH:$JAVA_HOME/bin
```

让上述的配置生效: `source /etc/profile`

检查 jdk 是否安装成功: `java -version javac -version`

tomcat 安装: == > web 应用服务器/web 容器/web 服务器 默认端口 8080

类似 tomcat 软件: apache jboss weblogic jetty 等

第一步: 上传 tomcat 的安装包到 opt 目录并解压, 解压后更名

```
[root@192 opt]# mv apache-tomcat-9.0.0.M26 tomcat9.0
```

第二步: 进入 tomcat / bin 目录 , 对 sh 文件添加执行权限

```
[root@192 opt]# cd tomcat9.0/bin/
[root@192 bin]# chmod +x *.sh
```

第三步：启动 tomcat `sh startup.sh` 或 `./startup.sh` 关闭 `./shutdown.sh`

第四步：关闭防火墙 `service firewalld stop`

第五步：在浏览器中使用 <http://linux> IP 地址:8080 进行测试是否安装成功

文件查看命令：

`cat` 文件名：查看小文件

`more` 文件名：查看大文件，按回车键翻页 按 `q` 退出

`Head` 文件名：默认显示文件的前 10 行 `-n` 指定行数

```
head -n 3 a.txt
```

`tail` 文件名：默认显示文件的最后 10 行 `-n` 指定行数 `-f` 刷新文件末尾

tomcat 目录结构：

```
[root@192 apache-tomcat-9.0.0.M26]# ls
bin    lib      logs     RELEASE-NOTES  temp      work
conf   LICENSE NOTICE  RUNNING.txt    webapps
```

Bin:存放 tomcat 的可执行程序 如 启动、停止等

Lib: 存放 tomcat 支持库 .jar 包文件

**Logs:** 存放 tomcat 运行日志文件

Conf: 存放 tomcat 配置文件的路径 1、配置端口 2、配置数据库连接池等

**Webapps:** 存放 tomcat 工程项目文件，即开发写的代码存放的路径

查看 tomcat 流水日志: `tail -f catalina.out`

Find 查找命令：

Find 查找起始路径 `-name` 文件名 `-type` 文件类型 (`d` 查找文件夹 `f` 查找文件)

```
[root@192 logs]# find / -name a.txt -type f
```

查杀进程命令：

Windows:

`tasklist | findstr QQ.exe` == 查看 QQ.exe 进程的信息

`taskkill /F /PID 进程编号` : 杀死进程 `/F` 强制杀死进程

| 特殊符号，管道命令 功能：把上一条命令的结果作为下一条命令的输入

linux 查杀进程：

`ps -ef | grep tomcat` : 查找 tomcat 的进程信息

`kill -9 进程 ID` : `-9` 强制杀死

netstat : 根据进程编号查看端口信息 46783 为应用程序 (tomcat) 进程 id 编号

```
netstat -anp |grep 46783
```

lsof : 根据端口 查看是被哪个应用程序占用

```
[root@192 logs]# lsof -i :8080
COMMAND  PID USER  FD   TYPE DEVICE SIZE/OFF NODE NAME
java    46783 root   48u  IPv6 235236      0t0  TCP *:webcache (LISTEN)
[root@192 logs]# ps -ef |grep 46783
root      46783      1  0 02:15 pts/6    00:00:02 /opt/jdk1.8.0_131/bin/java
            il.logging.config.file=/opt/apache-tomcat-9.0.0.M26/conf/logging.properties -
            l.logging.manager=org.apache.juli.ClassLoaderLogManager -Djdk.tls.ephemeralDH
            048 -Djava.protocol.handler.pkgs=org.apache.catalina.webresources -classpath
```