



链滴

linux 实际工作中用的命令总结

作者: [zhangzeshan](#)

原文链接: <https://ld246.com/article/1609589523534>

来源网站: 链滴

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)



基本命令：

1.ls 路径：

a.ls+路径 列出路径下的文件夹和文件的名字
如果没有路径 则列出当前下的文件夹和文件的名字

b.ll:以列表的形式列出当前下的文件夹和文件的名字 显示文件信息

c.ll -l :以列表的形式列出当前下的文件夹和文件的名字 显示文件信息

d.ll -al;以列表的形式列出当前下的文件夹和文件的名字 显示文件信息 包含隐藏文件

2.clear

清空当前屏幕的所有内容 clear

3.su

切换用户

su 用户名 su admin

4.logout

退出当前用户

5.shutdown 关机命令

shutdown -h 分钟数 表示多少分钟后关机

shutdown -r 分钟数 多少分钟后重启

shutdown -k 分钟数 多少分钟后关机 只发出警告 不执行关机

shutdown -c 取消关机

halt 立即关机

reboot 立即重启

6.cat指令

1.直接打开一个文件的功能

如: cat 文件的路径

2.对文件进行合并

如:cat 文件1 文件2 文件3。。。。

也可以 cat 文件1 文件2 文件3>>新的文件

这样会把三个文件的内容合并到新的文件中

7.目录切换命令

1.cd 需要切换的路径 (路径可以是相对路径 也可以是绝对路径)

切换目录 cd ../

例: cd /home

2.pwd

打印当前的路径

例: pwd

8.输出当前目录下各个子目录所使用的空间

du -h --max-depth=1

文件操作命令

1.文件创建: touch 文件名 文件名也可以是路径加文件名

例: touch test.php

2.文件赋值: cp 文件名 存放的新路径

例: `cp a.php ../home`

3.文件移动: `mv` 文件名 移动到的位置

例: `mv a.php /home`

4.文件删除: `rm` 文件名

例: `rm a.php` 确认删除 输入y 或者yes

强制删除: `rm -f a.php`

5.文件重命名: `mv` 文件名 新名字

例: `mv a.php b.php`

文件夹操作命令

1.文件夹创建: `mkdir` 文件夹名字 文件夹名字也可以是路径加文件夹名字

例: `mkdir zzs`

2.文件夹复制: `cp -r` 文件夹名字 位置

例: `cp -r clp ../`

3.文件夹移动: `mv` 文件夹名字 位置

例: `mv clp ../`

4.文件夹删除: `rm -f` 文件夹名字

例: `rm -f clp clp`

5.文件夹重命名: `mv` 文件夹名字 新名字

例: `mv zzs clp`

其他命令

1.`ctrl+c` 停止正在运行的命令

2.`ctrl+a` 光标快速移动到命令的第一位

3.`ctrl+e` 光标快速移动到命令的最后一位

4.向上键和向下键 切换历史命令

5.`tab` 补全文件名

修改服务器ssh默认端口

```
vim /etc/ssh/sshd_config  
#Port22 去掉默认22 另起一行改为Port 5157  
/bin/systemctl restart sshd.service
```

vim使用：（vi 没有vim强大）

1.打开需要编辑的文件

```
vim 文件名  
进入之后 想退出 按：  
wq（保存并退出） wq！（强制保存并退出）  
q！（强制退出）  
进入文件 显示行号 输入 :set nu
```

2.打开需要编辑的文件 并将光标定位到该文件的第N行

```
vim + n 文件名  
例:vim +5 a.php 打开a.php文件 并将光标定位到第五行
```

3.打开需要编辑的文件 显示所要查询的字符串 加上高亮效果

```
vim +/字符串 文件名  
例: vim +a/ index.php  
如果要在高亮的结果中进行快速跳转 可以按 n（向下切换） N（向上切换）  
想去除高亮显示 输入:nohl 按下回车
```

以上三种打开文件的方式 有一个特点 当打开的文件不存在的时候 会以一个新的文件进行打开

4.替换内容

s/需要替换的字符串/替换成的字符串 表示替换

s/需要替换的字符串/替换成的字符串/g 表示替换光标所在行全部符合条件的地方

%s/需要替换的字符串/替换成的字符串 表示替换当前文件中的每一行的第一个符合条件的地方

%s/需要替换的字符串/替换成的字符串/g 表示替换整个文档中全部符合条件的地方

5.查找

/查找的字符串

6.光标移动

在非输入操作的情况下

G: 大写G 光标直接跳到最后一行

在非输入操作的情况下
第n行+G：光标直接跳到第N行 如：5G 跳到第五行
在非输入操作的情况下
gg:两个小写的g 光标直接跳到第一行

7.删除文本内容

在非输入操作的情况下
dd：删除光标所在一行 之后 下一行往上移动

D：删除当前行 下一行不会往上移动

数字+dd:删除多行 以当前光标为起始点 向下算起删掉n行（起始点为1）

8.复制文本内容

yy:复制光标所在的这一行
数字+yy:复制多行 以当前光标为起始点 向下算起复制n行（起始点为1）
复制完成后按 P 进行粘贴

用户和用户组

用户

1.添加用户

useradd 用户名
例：useradd zzs

2.设置密码

passwd 需要设置密码的用户名
例：passwd zzs

3.编辑用户

usermod -l 用户的新名字 用户的旧名字

例:usermod -l clp zzs

usermod -g 新的用户组id 用户名

例：usermod -g 500 clp

4.删除用户

userdel 用户名
例：userdel clp

5.切换用户

su 用户名

用户组

1.添加用户组

groupadd 用户组名字

例: groupadd china

2.用户组编辑

groupmod 用户组名

例: groupmod -n 新的用户组名 旧的用户组名

3.用户组删除

groupdel 用户组名

例: groupdel

权限设置

第一个字符 表示文档的类型 d表示的是目录 -表示文件 ugo

第2-4个字符 表示文件的所有者的权限情况 u表示 user

第5-7个字符 表示所有者同组的组内成员的权限情况 g 表示 group

第8-10个字符 表示除了上述两类人员的其他用户的权限情况 o 表示other

1.通过字符形式设置权限

读4 写2 执行1

755, 三个数字, 第一个, 是所有者的权限, 第二个是自己用户组权限, 第三个是其他用户权限

命令: chmod

第一种情况 针对单个对象进行拆分 u所有者 g同组用户 o其他用户

chmod 权限组成信息 需要操作的对象 (文件夹或者文件)

如果对象是文件夹 则需要加上-r参数 表示递归赋予权限

给某个组成部分添加读权限

chmod u+r,g+r,o+r 需要操作的对象 (文件夹或者文件)

给某个组成部分删减读权限

chmod u-r,g-r,o-r 需要操作的对象 (文件夹或者文件)

给某个组成部分设置一个目标值
比如权限的全部组成部分都设为rwx
则 `chmod u=rwx,g=rwx,o=rwx` 需要操作的对象（文件夹或者文件）

第二种情况 设置全部的组成部分 `a=ugo`
`chmod a+r` 文件 给全部的组成部分添加读权限
`chmod a-r` 文件 给全部的组成部分删减读权限
`chmod a=rwx` 文件 给全部的组成部分赋予可读可写可执行权限

2.通过数字形式设置权限

777表示全部的权限 4读 2写 1执行
比如：给一个文件设置 所有者的拥有全部权限 同组用户拥有读写 其他用户拥有读
则所有者 7 同组用户4+2 其他用户4 所以是764
`chmod 764` 文件名

查找命令

`find` 查找范围（路径） `-name` 文件名

linux下的>和>>

保存操作命令的结果到文件 可以使用> 或>>
>表示覆盖写 每次执行 就覆盖之前写入的内容
`echo 22 > start.sh`

>>表示追加写 每次执行 就追加到之前写入的内容后面
`echo 111 >> start.sh`

服务的开/关

服务：
`service` 服务名称 `start/stop/restart`

开机设置：
`systemctl start/stop/restart` 服务名称

rpm软件管理

1.查询安装

`rpm -qa` 【|grep】 需要查询的关键词
`-q` 表示query 查询 `-a`表示all 全部 |在linux中称为管道 `grep`表示从结果中进行过滤

2.软件卸载

`rpm -e` 需要卸载的软件名称（软件的完整名称 也就是通过上个命令查询到的名称） 【`--nodeps`】

-e表示卸载 --nodeps 表示忽略依赖关系

3.软件安装

rpm -ivh 需要安装的完整软件名称

-i表示安装 install -v表示显示进度条 -h表示进度条以#形式

进阶命令

1.df 查看磁盘空间

例: df -h

2.free 查看内存使用情况

例: free -m 表示以mb为单位查看

3.head 输出一个文件的前n行 如果不指定 则默认显示前10

例 head -n 文件路径

head -n 10 a.php

4.tail 输出文件的末尾的n行 不指定则默认末尾10

例 tail -n 文件路径

5.less 查看文件，以较少的内容进行输出

例: less 文件路径

6.wc 统计文件内容信息（行数 单词数 字节数）

例: wc -lwc 文件路径

l行数 w单词数 c字节数

7.date 操作时间日期（读取和设置）

例: date 输出 年月日 星期几 时分秒

date +%F 输出 年月日

date +"%F %T" 输出年月日 时分秒

8.cal 用来操作日历

例: cal查看日历 直接输出当前月份的日历

cal -3 输出上一个月 本月 下一个月的日历

cal -y 年 输出某一年的日历 cal -y 2008

9.hostname 读取服务器的主机名

10.id 查看用户的基本信息 不指定则默认当前的用户

11.whoami 显示当前用户名

12.ps 查看服务器的进程信息

-e 列出全部的进程 -f显示全部的列

13.top 查看服务器的进程占的资源

14.杀死端口号对应的进程

kill -9 `lsof -ti:端口号`
如: kill -9 `lsof -ti:9902`

15.查看某个文件的第几行内容

sed -n 5p 文件名 输出第5行

16.查看系统版本

cat /etc/redhat-release

17.启动服务和开机自启和查看服务状态

systemctl start 服务名.service
systemctl enable 服务名.service
systemctl status 服务名.service

18.创建软连接 为环境中的apache和mysql创建

ln -s 源文件路径 目标文件路径

例如: ln -s /usr/local/mysql/bin/mysql /bin/mysql

例如: ln -s /usr/local/apache/bin/apachectl /bin/apache

19.查看端口号状态

```
netstat -anp | grep 9501
```

20.查看进程加上端口

```
ps -def |grep 5200
```

21.查看已经开放的端口

```
firewall-cmd --list-ports
```

22.防火墙

增加开放的端口（添加之后记得重启防火墙）

```
firewall-cmd --zone=public --add-port=5200/tcp --permanent
```

查看开放的端口

```
firewall-cmd --zone=public --list-ports
```

linux中常见的压缩包有两种格式：gz bz2

解压gz格式的文件：tar -zxvf 需要解压的gz文件

解压bz2格式的文件：tar -jxvf 需要解压的bz2文件

yum的使用

yum list 列出当前已经装的和可以装的全部软件

yum search 名字 列出搜索的的包

yum -y install 包名 安装指定的包 -y表示允许不再确认

yum -y update 包名 更新指定的包 不指定包名则更新全部

yum -y remove 包名 卸载指定的包

日志切割

1./usr/local/nginx/logs/下创建文件log_clean.sh,编写内容:

```
#!/bin/bash
```

```
#LOGS_PATH为日志存放路径
```

```
LOGS_PATH=/usr/local/nginx/logs
```

```
YESTERDAY=$(date -d "yesterday" +%Y-%m-%d)
```

```
#切分日志文件
```

```
mv ${LOGS_PATH}/access.log ${LOGS_PATH}/access_${YESTERDAY}.log
```

```
mv ${LOGS_PATH}/error.log ${LOGS_PATH}/error_${YESTERDAY}.log
```

```
# 清理1天前log文件
```

```
rm -f ${LOGS_PATH}/access_$(date -d "2 days ago" +"%Y-%m-%d").log
rm -f ${LOGS_PATH}/error_$(date -d "2 days ago" +"%Y-%m-%d").log
kill -USR1 $(cat ${LOGS_PATH}/nginx.pid)
```

```
echo 0
```

2.修改权限

```
chmod a+x log_clean.sh
```

3.添加定时任务

```
# 定时任务
crontab -e
```

```
# 添加定时任务
0 0 * * * /usr/local/nginx/logs/log_clean.sh
```

4.重启和重载

```
# 重启crontab
/bin/systemctl restart crond.service
# 重载crontab配置
/bin/systemctl reload crond.service
```