

Linux常用命令

- Linux 刚出世时没有什么图形界面，所有的操作全靠命令完成,就如同电视里的黑客那样，充满了神秘与晦涩。近几年来，尽管 Linux 发展得非常迅速，图形界面越来越友好，但是在真正的开发过程中，Linux 命令行的应用还是占有非常重要的席位，而且许多Linux功能在命令行界面要比图形化界面下运行的快，可以说不会命令,就不算会Linux。
- 在职场中，大量的**服务器维护工作**都是在远程通过**SSH 客户端**来完成的，并没有图形界面，所有的维护工作都需要通过命令来完成。如**磁盘操作、文件存取、目录操作、进程管理、文件权限设定**等

1. 目录类命令

命令	含义
/	代表根目录，/home代表根目录下的home目录
.	相对路径，代表当前目录
..	相对路径，代表上一级目录
cd	切换目录，如： <code>cd /</code> 进入根目录， <code>cd /home</code> 进入根目录下的home目录， <code>cd ..</code> 进入上一级目录
ls	查看当前目录下所有的文件
ll	查看当前目录下所有文件的详细信息
pwd	显示当前目录的完整路径

2 文件操作类

命令	含义
cp	格式: cp 源文件路径 目标路径 拷贝文件, <code>cp /home/a.txt ./b.txt</code> , 将/home目录下的a.txt复制到当前目录并命名为b.txt 拷贝文件夹, <code>cp -r /home/test1 /root/test2</code> , 将home目录下的test1文件夹, 复制到root目录下, 并命名为test2
mv	格式: mv 源文件路径 目标路径 移动文件/文件夹, <code>mv ./a.txt /home/b.txt</code> 将当前目录下的a.txt移动到home目录下, 并命名为b.txt
rm	格式: rm [选项] 文件路径 删除文件, <code>rm -rf ./a.txt</code> 删除当前目录下的a.txt
touch	格式: touch 文件路径 创建文件, <code>touch a.txt</code> 在当前目录下创建一个文件a.txt
mkdir	格式: mkdir [选项] 文件夹路径 创建文件夹, <code>mkdir test</code> 在当前目录下创建一个test文件夹 <code>mkdir -p /home/servers/test</code> , 在/home/servers目录下创建一个test文件夹, 如果父目录不存在, 则一并创建
./xxx.sh	格式: ./脚本名称 执行脚本文件, <code>./xxx.sh</code> 执行当前目录下的xxx.sh文件

3 文件编辑

在Linux系统中, 如果想对文本文件进行编辑/查看, 可以使用vi命令, vi就是Linux下的“记事本”

格式: `vi a.txt`

进入到vi界面后, 默认是非编辑模式, 不能进行编辑操作。

在非编辑模式下:

按`ctrl+f`: 向前翻一页
按`ctrl+b`: 向后翻一页
按`ctrl+d`: 向前翻半页
按`ctrl+u`: 向后翻半页
按`G`: 移动光标到文件最后
按`:0`: 移动光标到文件最开始
按`$`: 移动光标到行尾
按`^`: 移动光标到行首
`yy`: 复制光标当前行, `5yy` 复制包含当前行的5行数据
`p`: 粘贴复制的内容
`dd`: 删除光标当前行, `5dd`删除包含当前行的5行数据
输入`/abc`回车 搜索文件中包含`abc`的内容
输入`:wq` 保存文件并退出

在非编辑模式下, 按 `i` 键进入编辑模式, 移动光标, 可以随意输入/删除字符。按 `Esc` 退出编辑模式。

vi命令特点: 一次性加载整个文件, 如果文件比较大, vi加载的速度就比较慢。

4 仅查看文件

命令	含义
cat	<code>cat ./a.txt</code> 一次性读取并打印a.txt文件里的所有信息
more	<code>more ./a.txt</code> 读取a.txt文件，每次读取一屏，按空格键翻页，按 b 向后翻页
less	<code>less a.txt</code> 读取a.txt文件，每次读取一屏，按空格键下一页，b上一页
tail	<code>tail -200 a.txt</code> 读取a.txt文件的后两百行； <code>tail -f a.txt</code> 实时读取a.txt文件
head	<code>head -10 a.txt</code> 读取a.txt文件的前10行

5. 文件压缩/解压

命令	含义
zip	功能：压缩为zip文件 格式：zip [选项] 目标压缩包路径 源文件路径 zip -r test.zip test，将当前目test文件夹压缩为test.zip
unzip	功能：解压缩zip文件 格式：unzip 压缩包路径 示例： <code>unzip test.zip</code> 将test.zip解压到当前目录
tar cvf	功能：压缩为tar.gz文件 格式：tar cvf 目标压缩包路径 源文件路径 示例： <code>tar cvf test.tar.gz test</code> 将test文件夹压缩为test.tar.gz
tar xvf	功能：压缩为tar.gz文件 格式：tar xvf 压缩包路径 示例： <code>tar xvf test.tar.gz</code> 将test.tar.gz包解压到当前目录

6. 文件搜索

如果不知道某文件在什么路径下，可以使用find命令。

find命令功能非常强大，通常用来在特定的目录下搜索符合条件的文件

常用用法：

命令	含义
<code>find / -name test.sh</code>	查找根目录下所有名为test.sh的文件
<code>find . -name '*.txt'</code>	查找当前目录下所有后缀为.txt的文件

7. 文件内容过滤

如果想在文本文件中搜索指定的内容，可以使用grep命令。

grep命令是一种强大的文本搜索工具，grep允许对文本文件进行模式查找。如果找到匹配模式，grep打印包含模式的所有行。

grep的一般格式为：

```
grep [选项] '搜索内容串' 文件名
```

在grep命令中输入字符串参数时，最好引号或双引号括起来。

例如：

```
grep 'a' 1.txt
```

选项	含义
-v	显示不包含匹配文本的所有行（相当于求反）
-n	显示匹配行及行号
-w	精准匹配，完全匹配整个单词
-A	显示后n行， <code>grep -A 10 'error' test.log</code> 搜索test.log中包含error的行及其后10行
-B	显示前n行， <code>grep -B 10 'error' test.log</code> 搜索test.log中包含error的行及其前10行
-C	显示前n行+后n行

8. 文件权限控制

在执行ls -l或者ll命令时，可以看到文件的详细信息，如：

```
[root@VM-0-12-centos classes]# ll
total 32
-rw-rw-r-- 1 ptest ptest 1278 Dec 14 2019 application.properties
-rw-rw-r-- 1 ptest ptest 169 Dec 14 2019 banner.txt
drwxrwxr-x 3 ptest ptest 4096 Mar 1 2020 cn
-rw-rw-r-- 1 ptest ptest 1055 Dec 14 2019 log4j.properties
drwxrwxr-x 2 ptest ptest 4096 Mar 1 2020 mapper
drwxrwxr-x 4 ptest ptest 4096 Mar 1 2020 static
drwxrwxr-x 2 ptest ptest 4096 Mar 1 2020 templates
-rw-rw-r-- 1 ptest ptest 10 Dec 14 2019 test.properties
```

8.1 文件属性介绍

在上图红框处，显示的各文件的权限信息，如drwxrwxr-x，由四部分组成，各自含义如下：

第1个字符：表示该文件的类型，可以是d（目录）、-（普通文件）。

第2~4个字符：表示该文件的所属用户（user）对该文件的访问权限。

第5~7个字符：表示该文件的属组内各成员用户对该文件的访问权限。

第8~10个字符：表示其他任何用户（Other）对该文件的访问权限。

在每一部分中，“r、w、x、-”分别代表以下含义

字符	含义
r	查看文件内容
w	写入文件内容
x	执行文件
-	无权限

8.2 文件权限修改

命令：chmod 权限 文件路径

权限部分，可以采用数字来表示，其中

r=4

w=2

x=1

如，`chmod 777 a.txt`，给a.txt文件添加777权限，相当于rwxrwxrwx，这是最高的权限。

同理，`chmod 755 a.txt` 相当于rwxr-xr-x。

`chmod -R 777 mydir`，给mydir文件夹以及其子文件都添加777权限。

9. CentOS安装包管理工具yum

在CentOS中安装某些程序，可以使用yum来安装，yum能够从指定的服务器自动下载安装包并且安装，自动安装其依赖包。yum需要联网使用。

常用命令如下：

```
yum list: 列出所有的包（已安装+可安装）
yum search <包名>: 查找指定包
yum install -y <包名>: 安装指定的包（-y: 遇到询问直接确认yes）
yum remove <包名>: 删除指定包
yum update <包名>: 更新指定包
```

示例：yum install -y lrzsz，安装lrzsz工具包，实现文件上传和下载

`sz a.txt` 下载a.txt，在弹出窗口中选择保存路径

`rz` 上传文件，在弹出窗口中选择需要上传的文件

10. 管道符操作"|"

管道：一个命令的输出可以通过管道做为另一个命令的输入。

管道我们可以理解现实生活中的管子，管子的一头塞东西进去，另一头取出来，这里“|”的左右分为两端，左端塞东西(写)，右端取东西(读)。

```
ls / | more
```

11. 系统命令

查看系统ip

```
ip addr
```

```
[root@iZo8eovxsx3ru2Z ~]# ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default qlen 1000
    link/ether 00:16:3e:08:65:c2 brd ff:ff:ff:ff:ff:ff
    inet 172.31.140.175/20 brd 172.31.143.255 scope global dynamic eth0
        valid_lft 283134048sec preferred_lft 283134048sec
```

防火墙操作

```
关闭防火墙: systemctl stop firewalld
启动防火墙: systemctl start firewalld
重启防火墙: systemctl restart firewalld
```

网卡操作

```
关闭网卡: systemctl stop network
启动网卡: systemctl start network
重启网卡: systemctl restart network
```

进程操作

```
ps -ef | grep 进程名称
```

```
[root@iZo8eovxsx3ru2Z ~]# ps -ef | grep mysql
root      4419      1 0 Mar10 ?        00:00:00 /bin/sh /opt/zbox/run/mysql/mysqld_safe --defaults-file=/opt
nobody    4652    4419 0 Mar10 ?        00:00:20 /opt/zbox/run/mysql/mysqld --defaults-file=/opt/zbox/etc/my
box/run/mysql -datadir=/opt/zbox/data/mysql --plugin-dir=/opt/zbox/run/lib/mysql/plugin --user=nobody --lo
l_error.log --pid-file=/opt/zbox/tmp/mysql/mysqld.pid --socket=/opt/zbox/tmp/mysql/mysql.sock --port=3307
mysql     5273      1 0 Feb01 ?        00:23:49 /usr/sbin/mysqld --daemonize --pid-file=/var/run/mysqld/mys
root     31460  31126  0 17:39 pts/0    00:00:00 grep --color=auto mysql
```

上图中红框内的数字就是进程号。

如果想停止某个进程，使用kill命令：

```
kill -9 进程号
```

查看网络连接

查询3306端口建立的链接

```
netstat -anp | grep 3306
```

```
[root@iZo8eovxsx3ru2Z ~]# netstat -anp | grep 3306
tcp        0      0 172.31.140.175:58980 82.156.77.202:3306 ESTABLISHED 8429/java
tcp        0      0 172.31.140.175:45260 82.156.77.202:3306 ESTABLISHED 8425/java
tcp        0      1 172.31.140.175:46130 172.21.0.16:3306  SYN_SENT    8430/java
tcp        0      0 172.31.140.175:55924 82.156.77.202:3306 ESTABLISHED 8429/java
tcp        0      0 172.31.140.175:60240 82.156.77.202:3306 ESTABLISHED 8426/java
tcp        0      0 127.0.0.1:39200      127.0.0.1:3306  ESTABLISHED 2841/node
tcp        0      0 172.31.140.175:54260 82.156.77.202:3306 ESTABLISHED 8426/java
tcp        0      0 172.31.140.175:55950 82.156.77.202:3306 ESTABLISHED 8429/java
tcp        0      0 172.31.140.175:55920 82.156.77.202:3306 ESTABLISHED 8429/java
tcp        0      0 172.31.140.175:60242 82.156.77.202:3306 ESTABLISHED 8426/java
tcp        0      0 172.31.140.175:44446 82.156.77.202:3306 ESTABLISHED 8429/java
tcp        0      0 172.31.140.175:55922 82.156.77.202:3306 ESTABLISHED 8429/java
tcp        0      0 172.31.140.175:54262 82.156.77.202:3306 ESTABLISHED 8426/java
tcp        0      0 172.31.140.175:44598 82.156.77.202:3306 ESTABLISHED 8425/java
tcp        0      0 172.31.140.175:55934 82.156.77.202:3306 ESTABLISHED 8429/java
tcp        0      0 172.31.140.175:40664 82.156.77.202:3306 ESTABLISHED 8426/java
tcp        0      0 172.31.140.175:34192 82.156.77.202:3306 ESTABLISHED 8426/java
tcp        0      0 172.31.140.175:46744 82.156.77.202:3306 ESTABLISHED 8426/java
tcp        0      0 172.31.140.175:55918 82.156.77.202:3306 ESTABLISHED 8429/java
tcp        0      0 172.31.140.175:37186 82.156.77.202:3306 ESTABLISHED 8426/java
tcp        0      0 172.31.140.175:34190 82.156.77.202:3306 ESTABLISHED 5273/mysql
tcp        0      0 172.31.140.175:45892 82.156.77.202:3306 ESTABLISHED 8426/java
tcp6       0      0 :::3306             :::*             LISTEN      5273/mysql
tcp6       0      0 172.31.140.175:3306 172.31.140.174:61515 ESTABLISHED 5273/mysql
tcp6       0      0 127.0.0.1:3306      127.0.0.1:39200  ESTABLISHED 5273/mysql
tcp6       0      0 172.31.140.175:3306 172.31.140.174:57446 ESTABLISHED 5273/mysql
```

查看CPU使用率

```
top
```

(下图中红框的值是空闲率，用100-空闲率就是CPU使用率，单位为%)

```
top - 16:58:26 up 24 days, 5:30, 1 user, load average: 0.00, 0.00, 0.00
Tasks: 317 total, 1 running, 316 sleeping, 0 stopped, 0 zombie
Cpu(s): 4.7%us, 1.0%sy, 0.0%ni, 93.8%id, 0.0%wa, 0.0%hi, 0.4%si, 0.0%st
Mem: 32879608k total, 11094612k used, 21784996k free, 175632k buffers
Swap: 4194300k total, 0k used, 4194300k free, 4296872k cached
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
4195	root	20	0	12.9g	2.3g	19m	S	100.7	7.3	111:47.38	java
5456	root	20	0	15164	1400	924	R	0.3	0.0	0:01.06	top
19765	root	20	0	12.7g	2.2g	18m	S	0.3	7.1	92:33.30	java
1	root	20	0	19356	1528	1224	S	0.0	0.0	0:00.75	init
2	root	20	0	0	0	0	S	0.0	0.0	0:00.00	kthreadd
3	root	RT	0	0	0	0	S	0.0	0.0	0:00.50	migration/0
4	root	20	0	0	0	0	S	0.0	0.0	0:00.91	ksoftirqd/0
5	root	RT	0	0	0	0	S	0.0	0.0	0:00.00	stopper/0
6	root	RT	0	0	0	0	S	0.0	0.0	0:02.09	watchdog/0
7	root	RT	0	0	0	0	S	0.0	0.0	0:00.59	migration/1
8	root	RT	0	0	0	0	S	0.0	0.0	0:00.00	stopper/1
9	root	20	0	0	0	0	S	0.0	0.0	0:01.70	ksoftirqd/1
10	root	RT	0	0	0	0	S	0.0	0.0	0:03.51	watchdog/1

查看内存使用

```
free -m
```

以mb 为单位显示系统的内存使用情况，total为总内存，available为可用内存

```
[root@VM-0-16-centos ~]# free -m
Mem:      total      used      free      shared  buff/cache   available
Swap:      0         0         0
```

查看磁盘使用

```
df -h
```

查看磁盘空间的使用情况，一般关注Mounted On为根目录/的那行数据即可，Size为总容量，Use%为空间使用率

```
[root@VM-0-16-centos ~]# df -h
Filesystem      Size  Used Avail Use% Mounted on
devtmpfs        3.9G   0    3.9G   0% /dev
tmpfs           3.9G  8.1M   3.9G   1% /dev/shm
tmpfs           3.9G  688K   3.9G   1% /run
tmpfs           3.9G   0    3.9G   0% /sys/fs/cgroup
/dev/vda1       50G   16G   32G   33% /
tmpfs           783M   0    783M   0% /run/user/0
```