

当你进入了 Linux 的世界，在[下载](#)、[安装](#)了某个 [Linux 发行版](#)，体验了 Linux 桌面并安装了一些你喜爱和需要的[软件](#)之后，应该去了解 Linux 真正的魅力所在：**命令行**。每一个 Linux 命令其实就是一个程序，借助这些命令，我们可以办到非常多的事情。下面将会为大家介绍一下几个常用的命令。

如何寻求帮助？

在 Linux 下遇到问题，**最重要的是要自己寻求帮助**，下面是三种寻求帮助的方法。

man

man 是 Linux 的帮助手册，即 manual。因为大多数程序都会自带手册，所以可以通过 man 命令获取帮助。执行以后，在 **man page** 页面中按 q 退出。

获取 ls 的帮助

```
1. $ man ls
```

查看有多少（针对不同方面的）同名的手册

```
1. $ man -f ls
2. ls (1) - list directory contents
3. ls (lp) - list directory contents
```

查看特定的手册

```
1. $ man lp ls
```

info

与 man 不同的是，可以像浏览网页一样在各个节点中跳转。

从文档首页开始浏览

```
1. $ info
```

获取特定程序的帮助

```
1. $ info program
```

help

除了上面的两种方法外，还有一种简单使用的方法，那就是 `--help` 参数，一般程序都会有这个参数，会输出最简单有用的介绍。

1. `$ man --help ###` 获取 `man` 的帮助
2. `$ info --help ###` 获取 `info` 的帮助
3. `$ ls --help ###` 获取 `ls` 的帮助

如何简单操作?

在 Terminal (终端) 中, 有许多操作技巧, 这里就介绍几个简单的。

光标

- `up`(方向键上) 可以调出输入历史执行记录, 快速执行命令
- `down`(方向键下) 配合 `up` 选择历史执行记录
- `Home` 移动光标到本行开头
- `End` 移动光标到本行结尾
- `PgUp` 向上翻页
- `PaDN` 向下翻页
- `ctrl + c` 终止当前程序

Tab 补全

Tab 补全是非常有用的一个功能, 可以用来自动补全命令或文件名, 省时准确。

- 未输入状态下连接**两次** Tab 列出所有可用命令
- 已输入部分命令名或文件名, 按 Tab 进行自动补全, 多用你就肯定会喜欢的了。

常用命令

以下命令按照通常的使用频度排列。

cd

`cd` 是打开某个路径的命令, 也就是打开某个文件夹, 并跳转到该处。

1. `$ cd path ###` `path` 为你要打开的路径。

其中 `path` 有绝对路径和相对路径之分, 绝对路径强调从 `/` 起, 一直到所在路径。相对路径则相对于当前路径来说, 假设当前家目录有 `etc` 文件夹 (绝对路径应为 `/home/username/etc`), 如果直接 `cd etc` 则进入此文件夹, 但若是 `cd /etc/` 则是进入系统 `etc`, 多琢磨一下就可以理解了。另外在 Linux 中, `.` 代表当前目录, `..` 代表上级目录, 因此返回上级目录可以 `cd ..`。

ls

`ls` 即 **list**, 列出文件。

1. `$ ls ###` 仅列出当前目录可见文件

2. \$ ls -l #### 列出当前目录可见文件详细信息
3. \$ ls -hl #### 列出详细信息并以可读大小显示文件大小
4. \$ ls -al #### 列出所有文件（包括隐藏）的详细信息

注意： Linux 中 以 . 开头的文件或文件夹均为隐藏文件或隐藏文件夹。

pwd

pwd 用于返回当前工作目录的名字，为绝对路径名。

1. \$ pwd
2. /home

mkdir

mkdir 用于新建文件夹。

1. \$ mkdir folder
2. \$ mkdir -p folder/subfolder #### -p 参数为当父目录存在时忽略，若不存在则建立，用此参数可建立多级文件夹

rm

rm 即 **remove**，删除文件。

1. \$ rm filename #### 删除 filename
2. \$ rm -i filename #### 删除 filename 前提示，若多个文件则每次提示
3. \$ rm -rf folder/subfolder/ #### 递归删除 subfolder 下所有文件及文件夹，包括 subfolder 自身
4. \$ rm -d folder #### 删除空文件夹

cp

cp 即 **copy**，复制文件。

1. \$ cp source dest #### 将 source 复制到 dest
2. \$ cp folder/* dest #### 将 folder 下所有文件(不含子文件夹中的文件)复制到 dest
3. \$ cp -r folder dest #### 将 folder 下所有文件（包含子文件夹中的所有文件）复制到 dest

mv

mv 即 **move**，移动文件。

1. \$ mv source folder ### 将 source 移动到 folder 下，完成后则为 folder/source
2. \$ mv -i source folder ### 在移动时，若文件已存在则提示 **是否覆盖**
3. \$ mv source dest ### 在 dest 不为目录的前提下，重命名 source 为 dest

cat

cat 用于输出文件内容到 Terminal 。

1. \$ cat /etc/locale.gen ### 输出 locale.gen 的内容
2. \$ cat -n /etc/locale.gen ### 输出 locale.gen 的内容并显示行号

more

more 与 cat 相似，都可以查看文件内容，所不同的是，当一个文档太长时，cat 只能展示最后布满屏幕的内容，前面的内容是不可见的。这时候可用 more 逐行显示内容。

1. \$ more /etc/locale.gen
2. \$ more +100 /etc/locale.gen ### 从 100 行开始显示

less

less 与 more 相似，不过 less 支持上下滚动查看内容，而 more 只支持逐行显示。

1. \$ less /etc/locale.gen
2. \$ less +100 /etc/locale.gen

nano

nano 是一个简单实用的文本编辑器，使用简单。

1. \$ nano filename ### 编辑 filename 文件，若文件不存在，则新打开一个文件，若退出时保存，则创建该文件

编辑完后，ctrl + X 提示是否保存，按 y 确定保存即可。

注意：在使用过程中可用 ctrl + G 获取帮助。

reboot

reboot 为重启命令。

1. # reboot ### '\$' 和 '#' 的区别在于 '\$' 普通用户即可执行 ### 而 '#' 为 root 用户才可执行，或普通用户使用 'sudo'

poweroff

poweroff 为关机命令。

1. # poweroff ### 马上关机

ping

ping 主要用于测试网络连通，通过对目标机器发送数据包来测试两台主机是否连通，及延时情况。

1. \$ ping locez.com ### 通过域名 ping，若 DNS 未设置好，可能无法 ping 通
2. \$ ping linux.cn
3. PING linux.cn (211.157.2.94) 56(84) bytes of data.
4. 64 bytes from 211.157.2.94.static.in-addr.arpa (211.157.2.94):
icmp_seq=1 ttl=53 time=41.5 ms
5. 64 bytes from 211.157.2.94.static.in-addr.arpa (211.157.2.94):
icmp_seq=2 ttl=53 time=40.4 ms
6. 64 bytes from 211.157.2.94.static.in-addr.arpa (211.157.2.94):
icmp_seq=3 ttl=53 time=41.9 ms
7. ^C
8. --- linux.cn ping statistics ---
9. 3 packets transmitted, 3 received, 0% packet loss, time 2002ms
10. rtt min/avg/max/mdev = 40.406/41.287/41.931/0.644 ms
- 11.
12. \$ ping 211.157.2.94 ### 通过 IP 地址 ping，若无法 ping 通可能是网络连接出现问题

grep

grep 主要用于返回匹配的项目，支持正则表达式。

1. \$ grep PATTERN filename ### 返回所有含有 PATTERN 的行
2. \$ grep zh_CN /etc/locale.gen ### 返回所有含 zh_CN 的行

mount

mount 用于挂载一个文件系统，需要 root 用户执行。一个磁盘可分为若干个分区，在分区上面可以创建文件系统，而挂载点则是提供一个访问的入口，将一个分区的文件系统挂载到某个目录中，称这个目录为挂载点，并且可以通过这个挂载点访问该文件系统中的内容。

例如一块硬盘在 Linux 中表示为 /dev/sda 那么它上面的分区应该表示为 /dev/sda1、/dev/sda2。

1. # mount ### 输出系统目前的挂载信息

2. # mount /dev/sda1 /mnt ### 将 sda1 挂载到 /mnt 中
3. # cd /mnt ### 直接通过 /mnt 访问内容
4. # mount -o remount,rw /mnt ### 重新挂载 sda1 到 /mnt 并设置为可读写
5. # mount -a ### 挂载 fstab 文件配置好的文件系统

umount

umount 与 mount 相反，是卸载一个挂载点，即取消该入口。

1. # umount /mnt ### 卸载 /mnt 这个挂载点的文件系统
2. # umount -a ### 卸载所有已挂载的文件系统

tar

tar 主要用于创建归档文件，和解压归档文件，其本身是没有压缩功能的，但可以调用 gzip、bzip2 进行压缩处理。

参数解释：

- -c 创建归档
- -x 解压归档
- -v 显示处理过程
- -f 目标文件，其后必须紧跟 目标文件
- -j 调用 bzip2 进行解压缩
- -z 调用 gzip 进行解压缩
- -t 列出归档中的文件

1. \$ tar -cvf filename.tar . ### 将当前目录所有文件归档，但不压缩，注意后面有个 ' . '，不可省略，代表当前目录的意思
2. \$ tar -xvf filename.tar ### 解压 filename.tar 到当前文件夹
3. \$ tar -cvjf filename.tar.bz2 . ### 使用 bzip2 压缩
4. \$ tar -xvjf filename.tar.bz2 ### 解压 filename.tar.bz2 到当前文件夹
5. \$ tar -cvzf filename.tar.gz ### 使用 gzip 压缩
6. \$ tar -xvzf filename.tar.gz ### 解压 filename.tar.gz 到当前文件夹
7. \$ tar -tf filename ### 只查看 filename 归档中的文件，不解压

ln

ln 主要用于在两个文件中创建链接，链接又分为 Hard Links (硬链接)和 Symbolic Links (符号链接或软链接)，其中默认为创建**硬链接**，使用 -s 参数指定创建软链接。

- 硬链接主要是增加一个文件的链接数，只要该文件的链接数不为 0，该文件就不会被物理删除，所以删除一个具有多个硬链接数的文件，必须删除所有它的硬链接才可删除。

- 软链接简单来说是为文件创建了一个类似快捷方式的东西，通过该链接可以访问文件，修改文件，但不会增加该文件的链接数，删除一个软链接并不会删除源文件，即使源文件被删除，软链接也存在，当重新创建一个同名的源文件，该软链接则指向新创建的文件。
- 硬链接只可链接两个文件，不可链接目录，而软链接可链接目录，所以软链接是非常灵活的。

1. `$ ln source dest ###` 为 source 创建一个名为 dest 的硬链接
2. `$ ln -s source dest ###` 为 source 创建一个名为 dest 的软链接

chown

chown 用于改变一个文件的所有者及所在的组。

1. `# chown user filename ###` 改变 filename 的所有者为 user
2. `# chown user:group filename ###` 改变 filename 的所有者为 user，组为 group
3. `# chown -R root folder ###` 改变 folder 文件夹及其子文件的所有者为 root

chmod

chmod 永远更改一个文件的权限，主要有 **读取**、**写入**、**执行**，三种权限，其中 **所有者**、**用户组**、**其他** 各占三个，因此 `ls -l` 可以看到如下的信息

1. `-rwxr--r-- 1 locez users 154 Aug 30 18:09 filename`

其中 r=read，w=write，x=execute

1. `# chmod +x filename ###` 为 user，group，others 添加执行权限
2. `# chmod -x filename ###` 取消 user，group，others 的执行权限
3. `# chmod +w filename ###` 为 user 添加写入权限
4. `# chmod ugo=rwx filename ###` 设置 user，group，others 具有读取、写入、执行权限
5. `# chmod ug=rw filename ###` 设置 user，group 添加读取、写入权限
6. `# chmod ugo=--- filename ###` 取消所有权限

useradd

useradd 用于添加一个普通用户。

1. `# useradd -m -g users -G audio -s /usr/bin/bash newuser`
2. `### -m` 创建 home 目录，`-g` 所属的主组，`-G` 指定该用户在哪些附加组，`-s` 设定默认的 shell，newuser 为新的用户名

passwd

passwd 用于改变用户登录密码。

1. \$ passwd ### 不带参数更改当前用户密码
2. # passwd newuser ### 更改上述新建的 newuser 的用户密码

whereis

whereis 用于查找文件、手册等。

1. \$ whereis bash
2. bash: /usr/bin/bash /etc/bash.bashrc /etc/bash.bash_logout
/usr/share/man/man1/bash.1.gz /usr/share/info/bash.info.gz
3. \$ whereis -b bash ### 仅查找 binary
4. bash: /usr/bin/bash /etc/bash.bashrc /etc/bash.bash_logout
5. \$ whereis -m bash ### 仅查找 manual
6. bash: /usr/share/man/man1/bash.1.gz
/usr/share/info/bash.info.gz

find

find 也用于查找文件，但更为强大，支持正则，并且可将查找结果传递到其他命令。

1. \$ find . -name PATTERN ### 从当前目录查找符合 PATTERN 的文件
2. \$ find /home -name PATTERN -exec ls -l {} \; # 从 /home 文件查找所有符合 PATTERN 的文件，并交由 ls 输出详细信息

wget

wget 是一个下载工具，简单强大。

1. \$ wget -O newname.md
https://github.com/LCTT/TranslateProject/blob/master/README.md
下载 README 文件并重命名为 newname.md
2. \$ wget -c url ### 下载 url 并开启断点续传

恭喜你，你已经学习了完了 26 个基础的 Linux 命令。虽然这里只是一些最基础的命令，但是熟练使用这些命令就踏出了你从一位 Linux 新手成为 Linux 玩家的第一步！