

Linux 命令行工具使用小技巧（一）

Linux 命令行工具使用的一些小技巧

我们假设你已经掌握了一些 Linux 命令行的基础知识，比如什么是 root 账号及 home 目录，什么是环境变量，如何查看目录内容等等。说明这些小技巧的同时也会介绍涉及到的概念，如果有的话。

轻松切换目录 —— 快捷方式

假设你正在命令行下做一些操作，并且你需要经常在两个目录间来回切换。而且这两个目录在完全不同的两个路径下，比如说，分别在 /home/ 和 /usr/ 下。你会怎么做呢？

其中，最简单直接的方式就是输入这些目录的全路径。虽然这种方式本身没什么问题，但是却很浪费时间。另外一种方式就是打开两个终端窗口分别进行操作。但是这两种方式使用起来既不方便，也显得没啥技术含量。

你应该感到庆幸的是，还有另外一种更为简捷的方法来解决这个问题。你需要做的就是先手动切换到这两个目录（通过 cd 命令分别加上各自的路径），之后你就可以使用 cd - 命令在两个目录之间来回快速切换了。

例如：

我现在在下面的目录：

1. \$ pwd
 2. /home/himanshu/Downloads
- 然后，我切换到 /usr/ 路径下的其它目录：

1. cd /usr/lib/
现在，我可以很方便的使用下面的命令来向前、向后快速地切换到两个目录：

1. cd -
下面是 cd - 命令的操作截图：

The Linux cd command

有一点我得跟大家强调下，如果你在操作的过程中使用 cd 加路径的方式切换到第三个目录下，那么 cd - 命令将应用于当前目录及第三个目录之间进行切换。

轻松切换目录 —— 相关细节

对于那些有强烈好奇心的用户，他们想搞懂 cd - 的工作原理，解释如下：如大家所知道的那样，cd 命令需要加上一个路径作为它的参数。现在，当 - 符号作为参数传输给 cd 命令时，它将被 OLDPWD 环境变量的值所替代。

The cd command explained

现在应该明白了吧，`OLDPWD` 环境变量存储的是前一个操作目录的路径。这个解释在 `cd` 命令的 `man` 帮助文档中有说明，但是，很遗憾的是你的系统中可能没有预先安装 `man` 命令帮助工具（至少在 `Ubuntu` 系统下没有安装）。

但是，安装这个 `man` 帮助工具也很简单，你只需要执行下的安装命令即可：

1. `sudo apt-get install manpages-posix`

然后做如下操作：

1. `man cd`

打开 `man` 帮助文档主页后，你将会看到下面很明确的解释：

1. —— 当 `-` 符号被用作 `cd` 命令的参数值时，将等同于下面的操作命令：
- 2.
3. `cd "$OLDPWD" && pwd`

毫无疑问，`cd` 命令设置了 `OLDPWD` 环境变量值。因此每一次你切换操作目录时，上一个目录的路径就会被保存到这个变量里。这还让我们看到很重要的一点就是：任何时候启动一个新的 `shell` 实例（包括手动执行或是使用 `shell` 脚本），都不存在‘上一个工作目录’。

Hyphen and the cd command

这也很符合逻辑，因为 `cd` 命令设置了 `OLDPWD` 环境变量值。因此，除非你至少执行了一次 `cd` 命令，否则 `OLDPWD` 环境变量不会包含任何值。

继续，尽管这有些难以理解，`cd -` 和 `cd $OLDPWD` 命令的执行结果并非在所有环境下都相同。比如说，你重新打开一个新的 `shell` 窗口时。

cd command example

从上面的截图可以清楚的看出，当执行 `cd -` 命令提示未设置 `OLDPWD` 值时，`cd $OLDPWD` 命令没有报任何错；实际上，它把当前的工作目录改变到用户的 `home` 目录里。那是因为 `OLDPWD` 变量目前还没有被设置，`$OLDPWD` 仅仅是一个空字符串。因此，`cd $OLDPWD` 命令跟 `cd` 命令的执行结果是一致的，默认情况下，会把用户当前的工作目录切换到用户的 `home` 目录里。

最后，我还遇到过这样的要求，需要让 `cd -` 命令执行的结果不显示出来。我的意思是，有这样的情况（比如说，在写 `shell` 脚本的时候），你想让 `cd -` 命令的执行结果不要把目录信息显示出来。那种情况下，你就可以使用下面的命令方式了：

1. `cd - &>/dev/null`

上面的命令把文件描述符 `2`（标准错误）和 `1`（标准输出）的结果重定向到 `/dev/null` 目录。这意味着，这个命令产生的所有的错误不会显示出来。但是，你也可以使用通用的 `$? 方式`来检查这个命令的执行是否异常。如果这个命令执行报错，`echo $?` 将会返回 `1`，否则返回 `0`。

或者说，如果你觉得 `cd -` 命令出错时输出信息没有关系，你也可以使用下面的命令来代替：

1. `cd - > /dev/null`

这个命令仅用于将文件描述符 `1`（标准输出）重定向到 `/dev/null`。

总结

遗憾的是，这篇文章仅包含了一个跟命令行相关的小技巧，但是，我们已经地对 `cd` - 命令的使用进行了深入地探讨。建议你在自己的 **Linux** 系统的命令行终端中测试本文中的实例。此外，也强烈建议你查看 `man` 帮助文档，然后对 `cd` 命令进行全面测试。

如果你对这篇文章有什么疑问，请在下面的评论区跟大家交流。同时，敬请关注下一篇文章，我们将以同样的方式探讨更多有用的命令行使用技巧。

via: <https://www.howtoforge.com/tutorial/linux-command-line-navigation-tips-and-tricks-part-1/>

作者: [Ansh](#) 译者: [rusking](#) 校对: [jasminepeng](#)

本文由 [LCTT](#) 原创编译, [Linux 中国](#) 荣誉推出