

Linux 命令行工具使用小技巧（三）

环境变量 CDPATH

即使你的命令行所有操作都在特定的目录下 - 例如你的主目录，然而在你切换目录时也不得不提供绝对路径。比如，考虑我现在的情况，就是在 `/home/himanshu/Downloads` 目录下：

1. `$ pwd`
2. `/home/himanshu/Downloads`

现在要求切换至 `/home/himanshu/Desktop` 目录，我一般会这样做：

1. `cd /home/himanshu/Desktop/`

或者

1. `cd ~/Desktop/`

或者

1. `cd ../Desktop/`

能不能只是运行以下命令就能简单地实现呢：

1. `cd Desktop`

是的，这完全有可能。这就是环境变量 `CDPATH` 出现的时候了。你可使用这个变量来为 `cd` 命令定义基础目录。

如果你尝试打印它的值，你会看见这个环境变量默认是空值的：

1. `$ echo $CDPATH`

现在，考虑到上面提到的场景，我们使用这个环境变量，将 `/home/himanshu` 作为 `cd` 命令的基础目录来使用。

最简单的做法这样：

1. `export CDPATH=/home/himanshu`

现在，我能做到之前所不能做到的事了 - 当前工作目录在 `/home/himanshu/Downloads` 目录里时，成功地运行了 `cd Desktop` 命令。

1. `$ pwd`

2. `/home/himanshu/Downloads`
3. `$ cd Desktop/`
4. `/home/himanshu/Desktop`
5. `$`

这表明了我可以使用 `cd` 命令来到达 `/home/himanshu` 下的任意一个目录，而不需要在 `cd` 命令中显式地指定 `/home/himanshu` 或者 `~`，又或者是 `../` (或者多个 `../`)。

要点

现在你应该知道了怎样利用环境变量 `CDPATH` 在 `/home/himanshu/Downloads` 和 `/home/himanshu/Desktop` 之间轻松切换。现在，考虑以下这种情况，在 `/home/himanshu/Desktop` 目录里包含一个名字叫做 `Downloads` 的子目录，这是将要切换到的目录。

但突然你会意识到 `cd Downloads` 会切换到 `/home/himanshu/Downloads`。所以，为了确保这不会发生，你可以这样做：

1. `cd ./Downloads`

虽然上述命令本身没有问题，但你还是需要耗费点额外的精力（虽然很小），尤其是每次这种情况发生时你都不得不这样做。所以，有一个更加优雅的解决方案来处理，就是以如下方式来设定 `CDPATH` 环境变量。

1. `export CDPATH=".: /home/himanshu"`

它的意思是告诉 `cd` 命令先在当前的工作目录查找该目录，然后再尝试搜寻 `/home/himanshu` 目录。当然，`cd` 命令是否以这样的方式运行，完全取决于你的偏好和要求 - 讨论这一点的目的是为了让你知道这种情况可能会发生。

就如你现在所知道的，一旦环境变量 `CDPATH` 被设置，它的值 - 或者它所包含的路径集合 - 就是系统中 `cd` 命令搜索目录的地方（当然除了使用绝对路径的场景）。所以，完全取决于你来确保该命令行为的一致性。

继续说，如果一个 `bash` 脚本以相对路径使用 `cd` 命令，最好还是先清除或者重置环境变量 `CDPATH`，除非你觉得遇上不可预测的麻烦也无所谓。还有一个可选的方法，比起在终端使用 `export` 命令来设置 `CDPATH`，你可以在测试完当前的 `shell` 是交互式还是非交互式之后，再在你的 `.bashrc` 文件里设置环境变量，这样可以确保你对环境变量的改动只对交互式 `shell` 生效。

环境变量中，路径出现的顺序同样也是很重要。举个例子，如果当前目录是在 `/home/himanshu` 目录之前列出来，`cd` 命令就会先搜索当前的工作目录然后才会搜索 `/home/himanshu` 目录。然而，如果该值为 `/home/himanshu:.`，搜索就首先从 `/home/himanshu` 开始，然后到当前目录。不用说，这会影响 `cd` 命令的行为，并且不注意路径的顺序可能会导致一些麻烦。

要牢记在心的是，环境变量 CDPATH，就像其名字表达的，只对 `cd` 命令有作用。意味着在 `/home/himanshu/Downloads` 目录里面时，你能运行 `cd Desktop` 命令来切换到 `/home/himanshu/Desktop` 目录，但你不能使用 `ls`。以下是一个例子：

```
1. $ pwd
2. /home/himanshu/Downloads
3. $ ls Desktop
4. ls: cannot access Desktop: No such file or directory
5. $
```

然而，这还是有简单的变通处理的。例如，我们可以用以下不怎么费力的方式来达到目的：

```
1. $ cd Desktop/;ls
2. /home/himanshu/Desktop
3. backup backup~ Downloads gdb.html outline~ outline.txt
   outline.txt~
```

不过，不是每种情况就能变通处理的。

另一个重点是：就像你可能已经观察到的，每次你使用 CDPATH 环境变量集来运行 `cd` 命令时，该命令都会在输出里显示你切换到的目录的完整路径。不用说，不是所有人都想在每次运行 `cd` 命令时看到这些信息。

为了确保该输出被制止，你可以使用以下命令：

```
1. alias cd='>/dev/null cd'
```

如果 `cd` 命令运行成功，上述命令不会输出任何东西，如果失败，则允许产生错误信息。

最后，假如你遇到设置 CDPATH 环境变量后，不能使用 `shell` 的 `tab` 自动补全功能的问题，可以尝试安装并启用 `bash` 自动补全（`bash-completion`）。更多请参考 [这里](#)。

总结

CDPATH 环境变量是一把双刃剑，如果没有掌握完善的知识和随意使用，可能会令你陷入困境，并花费你大量宝贵时间去解决问题。当然，这不代表你不应该去试一下；只需要了解一下所有的可用选项，如果你得出结论，使用 CDPATH 会带来很大的帮助，就继续使用它吧。

你已经能够熟练地使用 CDPATH 了吗？你有更多的贴士要分享？请在评论里发表一下你的想法吧。

（高清图题图：<http://anguerde.com/TTF-222621-archlinux.html>）

via: <https://www.howtoforge.com/tutorial/linux-command-line-tips-tricks-part-3-cdpath/>

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