

Linux mkdir、tar 和 kill 命令的 4 个有用小技巧

linux.cn

我们一直以常规的方式完成一个任务，直到我们知道有更好的处理方法。作为 Linux 技巧和绝招系列的后续，我会在这里介绍能在各个方面给你帮助的 4 个小技巧。开始吧！



4 个有用的 Linux 小技巧

1. 假设你要创建一个类似于下面很长的/复杂的目录树。实现这最有效的方法是什么呢？

类似下面要实现的目录树结构。

```
$ cd /home/$USER/Desktop
$ mkdir tecmint
$ mkdir tecmint/etc
$ mkdir tecmint/lib
$ mkdir tecmint/usr
$ mkdir tecmint/bin
$ mkdir tecmint/tmp
$ mkdir tecmint/opt
$ mkdir tecmint/var
$ mkdir tecmint/etc/x1
$ mkdir tecmint/usr/x2
```

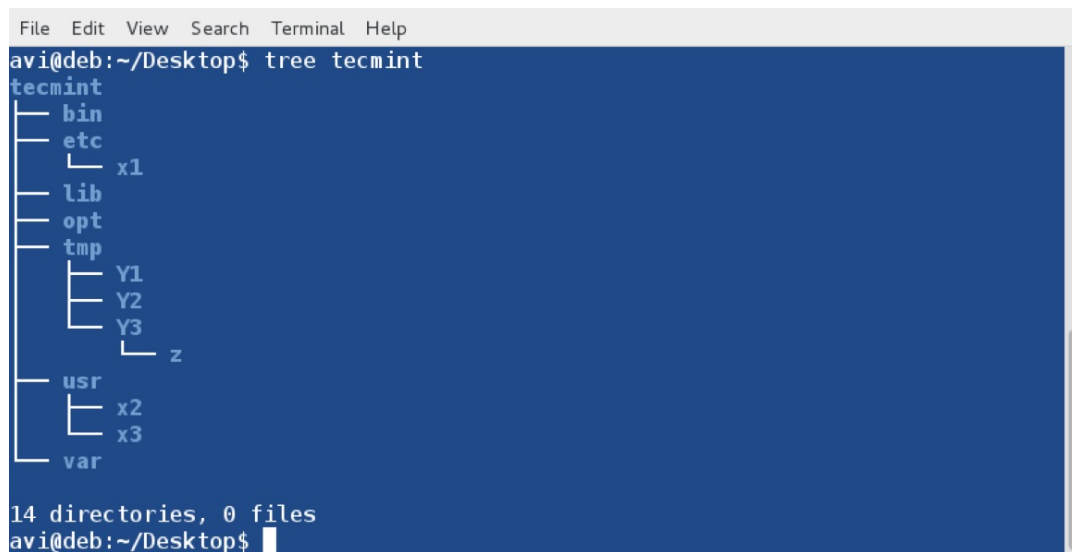
```
$ mkdir tecmint/usr/x3
$ mkdir tecmint/tmp/Y1
$ mkdir tecmint/tmp/Y2
$ mkdir tecmint/tmp/Y3
$ mkdir tecmint/tmp/Y3/z
```

上述情况可以简单地通过运行下面一行命令来实现。

```
$ mkdir -p
/home/$USER/Desktop/tecmint/{etc/x1,lib,usr/{x2,x3},bin,tmp/{Y1,Y2,Y3/z},opt,var}
```

你可以用 `tree` 命令验证。如果没有安装你可以使用 `apt` 或 `yum` 安装 ‘tree’ 软件包。

```
$ tree tecmint
```



```
File Edit View Search Terminal Help
avi@deb:~/Desktop$ tree tecmint
tecmint
├── bin
├── etc
│   └── x1
├── lib
├── opt
├── tmp
│   ├── Y1
│   ├── Y2
│   └── Y3
│       └── z
├── usr
│   ├── x2
│   └── x3
└── var

14 directories, 0 files
avi@deb:~/Desktop$
```

检查目录结构

我们可以用上面的方式创建任意复杂的目录树结构。注意这仅仅是一个普通的命令，但是用 ‘{}’ 来创建层级目录。需要的时候如果在 shell 脚本中使用是非常有用的。

2. 在桌面 (/home/\$USER/Desktop) 创建一个文件（例如 test）并填入

以下内容。

```
ABC
DEF
GHI
JKL
MNO
PQR
STU
VWX
Y
Z
```

这种情况一个普通用户会怎么做呢？

a. 他首先会创建文件，最好使用 `touch` 命令，例如：

```
$ touch /home/$USER/Desktop/test
```

b. 他会用一个文本编辑器打开文件，这可能是 `nano`、`vim` 或其它编辑器。

```
$ nano /home/$USER/Desktop/test
```

c. 然后他会将上面的内容输入到文件中，保存并退出。

忽略他/她使用的时间，他至少需要 3 步来执行上面的情况。

一个经验丰富的 Linux 用户会怎么做呢？他会在终端中输入下面的文本然后完成所有任务。他不需要单独执行每一步。

```
cat << EOF > /home/$USER/Desktop/test
ABC
DEF
GHI
JKL
MNO
```

```
PQR
STU
VWX
Y
Z
EOF
```

你可以用 ‘cat’ 命令检查是否成功创建了文件和内容。

```
$ cat /home/avi/Desktop/test
```

3. 我们经常在 Linux 中处理归档文件（尤其是 TAR 包）。很多情况下我们会在某些位置，而不是在 Downloads 目录中使用 TAR 包。这种情况下我们怎么做呢？

在这种情况下我们通常会做两件事。

a. 复制/移动 tar 包到目标位置并解压，例如：

```
$ cp firefox-37.0.2.tar.bz2 /opt/
或
$ mv firefox-37.0.2.tar.bz2 /opt/
```

b. cd 到 /opt/ 目录。

```
$ cd /opt/
```

c. 解压 tar 包。

```
# tar -jxvf firefox-37.0.2.tar.bz2
```

我们也可以采用另外一种方式。

我们也可以在 Tar 包所在位置解压并复制/移动解压后的文件到所需的目标位置，例如：

```
$ tar -jxvf firefox-37.0.2.tar.bz2
$ cp -R firefox/ /opt/
或
$ mv firefox/ /opt/
```

不管哪种方式都需要两步才能完成任务。专业的人可以只用一步就完成这个任务：

```
$ tar -jxvf firefox-37.0.2.tar.bz2 -C /opt/
```

-C 选项提取文件到指定目录（这里是 /opt/）。

这并不是关于选项（-C）的问题，而是习惯的问题。养成使用带 -C 选项 tar 命令的习惯。这会使你的工作更加轻松。从现在开始不要再移动归档文件或复制/移动解压后的文件了，在 Downloads 文件夹保存 tar 包并解压到你想要的任何地方吧。

4. 常规方式我们怎样杀掉一个进程？

最普遍的方法，我们首先用 ps -A 命令列出所有进程，然后通过管道输入到 grep 来查找进程/服务（假如 apache2），如下：

```
$ ps -A | grep -i apache2
```

输出样例

```
1006 ?      00:00:00 apache2
2702 ?      00:00:00 apache2
2703 ?      00:00:00 apache2
2704 ?      00:00:00 apache2
2705 ?      00:00:00 apache2
2706 ?      00:00:00 apache2
2707 ?      00:00:00 apache2
```

上面的输出显示了所有正在运行 apache2 的进程以及它们的 PID，然后你可以使用这些 PID 在下面命令的帮助下杀掉 apache2。

```
# kill 1006 2702 2703 2704 2705 2706 2707
```

然后交叉检查是否还有名称中包含 ‘apache2’ 的进程/服务在运行，如下：

```
$ ps -A | grep -i apache2
```

实际上我们可以使用类似 pgrep 和 pkill 的工具以一种更容易理解的方式实现。你可以使用 pgrep 找到和一个进程相关的信息。假如你要找和 apache2 相关的进程信息，你只需要运行：

```
$ pgrep apache2
```

输出样例

```
15396
15400
15401
15402
15403
15404
15405
```

你也可以通过运行下面命令列出进程名称以及 pid。

```
$ pgrep -l apache2
```

输出样例

```
15396 apache2
15400 apache2
15401 apache2
```

```
15402 apache2
15403 apache2
15404 apache2
15405 apache2
```

用 pkill 杀掉进程非常简单。你只需要输入想要杀死的资源名称。我写了一篇关于 pkill 的博文，你可以参考：<http://www.tecmint.com/how-to-kill-a-process-in-linux/>。

用 pkill 杀死一个进程（例如 apache2），你只需要输入以下命令：

```
# pkill apache2
```

你可以通过运行下面命令验证是否杀掉了 apache2。

```
$ pgrep -l apache2
```

它没有输出任何东西并返回到窗口意味着没有名称中包含 apache2 的进程在运行。

这就是我要说的所有东西。上面讨论的点肯定远远不够，但也肯定对你有所帮助。我们不仅仅是介绍教程使你学到一些新的东西，更重要的是想告诉你‘在同样的情况下如何变得更有效率’。在下面的评论框中告诉我们你的反馈吧。保持联系，继续评论。