

爱米家

艾米和小米的七七八八^_^

每个Mac OS X 命令行用户应当知道的八个终端工具

原文链接: [Eight Terminal Utilities Every OS X Command Line User Should Know](#)

OS X 的Terminal 终端开辟了强大的UNIX实用工具和脚本的世界。如果你是从Linux转移过来的用户，你会在OS X 终端上发现很多熟悉的命令。但是即使是高级用户也经常会忽略掉其实OS X终端也会有一系列在其他的操作系统中没有的终端工具。当你学习了以下这些只适用于Mac的课程会让你更高效的使用Mac命令行同时消除UNIX系统和你的Mac系统之间的差别。

1.open

open:打开文件，目录和程序。这是令人兴奋的，对吗？其实它真正的作用是用于命令行模拟鼠标双击操作。假如当你输入：

```
$ open /Applications/Safari.app/
```

该命令时会像你在Finder中双击Safari图标一样启动Safari浏览器。

如果你用open命令指向一个文件，它会启动相关的GUI应用程序加载该文件。open screenshot.png命令会在预览中打开该图片。当然你可以在命令中加入 -a 标识来选择你想要用来加载该文件的应用程序，或者使用 -e 标识直接在文本编辑器中打开编辑。

对于一个目录运行open命令会直接在Finder窗口中打开该目录。该方法特别适用于在命令行中运行open . 来打开当前目录。

请记住Finder和Terminal终端之间的集成是相互的，当你把Finder中的一个文件拖入到Terminal终端窗口时，它的绝对路径就会被粘贴在命令行中。

2.pbcopy 和 pbpaste

这两个命令可以让你复制和粘贴命令行中的文本内容。当然你可以直接使用鼠标完成上述操作。但是pbcopy和pbpaste真正的魔力是它们是UNIX命令，这使它们可以受益于管道（piping），重定向并且在脚

本中可以与其他命令一起使用。例如当输入：

```
$ ls ~ | pbcopy
```

该命令时会将你的home目录中的文件列表拷贝到OS X 系统的剪切板中。你可以通过如下命令轻易的获取文档的内容：

```
$ pbcopy < blogpost.txt
```

或者过一些更疯狂的事。下面这条脚本会抓取最新的Google doodle 的链接并且将它拷贝到你的剪切板中。

```
$ curl http://www.google.com/doodles#oodles/archive | grep -A5 'latest-doodle on' | grep 'img src' | sed s/.*'> tasklist.txt
```

3.mdfind

许多重度Linux用户会在Mac上使用locate命令搜索文件，但是他们很快发现这个方法不可行。在UNIX中有一个古老的find命令，但是在OS X中它有它自己的杀手级搜索工具：Spotlight。所以我们为啥不在命令行中借用它的力量呢？

这就是实际当中mdfind所做的事。任何Spotlight能搜索到的内容，mdfind同样也能搜索到。这当然也包括搜索文件内部及元数据。

mdfind因为它的一些便捷性使它在它的这个大哥中脱颖而出。例如，-onlyin 标识可以将搜索限制在一个指定的目录中：

```
$ mdfind -onlyin ~/Documents essay
```

mdfind数据库需要在后台经常更新数据，但你可以使用mdutil命令来排除故障（同样适用于Spotlight）。如果Spotlight没有正确的工作，使用mdutil -E将会清楚数据库中的索引然后通过抓取重建索引。你同样也可以通过运行mdutil -i off 来完全完毕索引。

4.screencapture

screencapture 可以让你拍摄多种屏幕快照。它类似于抓图程序或者cmd + shift +3 和 cmd + shift + 4 的截屏快捷键，但是它更加灵活。以下是几种你可以使用screencapture的方式：

抓取屏幕中的所有内容，包括光标，并且将该截图（以'image.png'命名）附再一封新的电子邮件中：

```
$ screencapture -C -M image.png
```

通过你的鼠标选择一个窗口，然后抓取该窗口中的内容（不包括该窗口的阴影效果）将该截图复制到剪贴板中：

```
$ screencapture -c -W
```

延时10秒截屏并且在预览中打开该截图：

```
$ screencapture -T 10 -P image.png
```

通过鼠标选定一个区域截图，同时将该内容保存为pdf文件：

```
$ screencapture -s -t pdf image.pdf
```

你可以通过输入 screencapture —help 来查看更多选项。

5.launchctl

launchctl 可以让你与OS X 的初始化脚本系统launchd进行交互。通过启动守护进程与启动代理，你可以在启动你的电脑时控制你的启动服务项。你甚至可以通过编写脚本定期或再指定的时间间隔内执行操作，类似于Linux中的cron工具。

例如，当你想要在你启动你的Mac时自动启动Apache服务器时，你可以输入：

```
$ sudo launchctl load -w /System/Library/LaunchDaemons/org.apache.httpd.plist
```

通过运行launchctl list会列出所有当前状态下加载了的启动脚本。sudo launchctl unload [path/to/script] 命令会停止并且卸载运行中的脚本，该命令增加 -w 标识时会将该脚本永久的从你的启动队列中清除。我喜欢用这个命令来终止运行所有类似于Adobe程序及微软Office的自动更新“助手”。

启动脚本储存在以下几个位置中：

~/Library/LaunchAgents

/Library/LaunchAgents

```
/Library/LaunchDaemons
```

```
/System/Library/LaunchAgents
```

```
/System/Library/LaunchDaemons
```

对于想要了解启动进程与启动代理的详细机制，推荐你去浏览[Paul Annesley](#) 的博客，该博客将从文件格式带你了解该内容。同时如果你想要学习如何编写你自己的启动脚本，Apple在他们[开发者网站](#)上提供了许多很有帮助的文档供参考。如果你希望完全避免命令行的操作，同样有一个非常棒的叫[Lingon](#)的程序可以帮助你。

6.say

这是一个很有趣的命令：`say` 将文本转化为语音，它使用了OS X中[VoiceOver](#)使用的文字语音转换系统。无需任何选项，`say`命令会将你输入的任何文本内容转化为语音输出：

```
$ say "Never trust a computer you can't lift."
```

你可以使用带-f 标识的`say`命令来朗读一个文本文档中的内容，同时使用-o 标识来保存输出的音频内容：

```
$ say -f mynovel.txt -o myaudiobook.aiff
```

`say` 命令可以用于脚本的控制台日志和报警声音。例如，你可以设置批处理或者[Hazel](#)脚本处理文件，在处理完成后使用`say`命令发出提醒。

但是对于`say`命令最好玩的用法是比较阴险的：如果对于你的朋友或者同事的Mac你有ssh登陆权限，你可以通过命令行悄无声息的登陆他们的电脑。然后给他们一个Siri-ous惊喜。

你可以在系统偏好设置中的听写与语音面板中设置`say`命令的语音（和语言）。

7.diskutil

`diskutil` 是OS X中磁盘管理工具的命令行界面。它可以完成任何它的图形界面兄弟能完成的任务，同时它也包含一些额外的能力—例如在一个磁盘中写满零或者随机数据。简单的输入`diskutil list`会列出所有磁盘的路径名和链接到你电脑上的可移除的媒体介质，然后再指向你想要操作的卷的命令。请注意：如果不正确的使用`diskutil`命令会永久的清楚磁盘上的数据。

8.brew

好吧，这个严格上说不是一个本地命令。但是没有一个OS X高级用户会离得开[Homebrew](#)。它的官网将它定义为“OSX 不可或缺的套件管理器”，它说的很正确。如果你在Linux系统上使用过`apt-get`命令，你会在

Homebrew中有找到组织的感觉。

brew 让你轻松的从开源社区获取成千上万的开放的实用工具和软件库。例如，`brew install imagemagick` 命令会为你安装ImageMagick工具，它是一个功能强大的工具包，它可以从事任何从GIF动画到几十种不同类型的图片转换任务。`brew install node` 命令会为你安装NodeJS，它是一个最新的热门的在服务器端开发和运行JavaScript程序的工具。

你同样也可以用Homebrew来获得乐趣：`brew install archey` 命令会为你安装 **ArcheY**，它是一个很酷脚脚本，用来在一个多彩的Apple Logo显示你的Mac的配置。Homebrew的可选内容很广泛，同时它很容易创建新规则，一直都会有新的包被添加进来。

ARCHEY —我的命令行工具中把所有的配置都显示出来。

但是Homebrew最大的优势是什么？它将它所有的文件都保存在一个独立的目录中：`/usr/local/`。这意味着你可以无需干涉系统内建的软件而安装系统软件的最新版本，比如python和mysql。而且当你卸载Homebrew 安装包时，它很方便就可以被移除。

5.15修改：有人对我建议不能盲目的删除/usr/local/目录中的内容。这是一个非常重要的预防措施。为了安全的移除Homebrew，请使用[卸载脚本](#)。

关于终端程序的更多操作，这里有一个从A-Z排序的[在OS X 10.9 Mavericks中所有可用的控制台命令行](#)。

分享到：

本条目发布于2014/05/16 [<http://aimijia.net/2014/05/each-mac-os-x-command-line-users-should-be-aware-of-the-eight-terminal-tool/>]。属于小米的七七分类，被贴了 MAC OS X、翻译 标签。作者是micky。

《每个Mac OS X 命令行用户应当知道的八个终端工具》上有4条评论

Pingback引用通告：[每个Mac OS X 命令行用户应当知道的八个终端工具 | 小虎的开发笔记](#)



mojojo

2014/05/2409:07

brew是“OSX 不可或缺的套件管理器”



micky 文章作者

2014/05/25 19:15

已更新，多谢^_^

Pingback引用通告： [每个Mac OS X 命令行用户应当知道的终端工具 | 网站采集](#)