

用纯JAVA语言编程读取MAC地址的实现 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/252/2021_2022__E7_94_A8_E7_BA_AFJAVA_c104_252501.htm

1 引言 用Java编写的程序，可以很方便地运行在各种平台的环境。但在实际的开发过程中，有时不得不涉及一些底层的编程。比如为了防止软件盗用，我们希望软件只能在指定计算机上运行，所以需要程序读取该机区别于其它计算机的硬件特征，如MAC地址等。作为一种跨平台语言，给Java语言提出了挑战。本文正是针对该问题，提出一种直接用纯Java语言，读取MAC地址的编程方法。我们知道，在每一个Java应用程序中都存在着一个与其运行环境相联系的Runtime对象。该对象可执行外部命令、查可用内存等。而多数操作系统都提供有查询该机MAC地址的命令。如在Microsoft的操作系统中，命令IPCONFIG等。本文的思路是在程序中运行一个外部命令，将该命令的运行结果作为一个流(Stream)，读取并分析之，进而实现获取MAC地址的目的。

2 Runtime类 在每一个Java应用程序里面，都有唯一的一个Runtime对象。通过这个Runtime对象，应用程序可以与其运行环境发生相互作用。一般不实例化一个Runtime对象。但是可以通过调用静态方法Runtime.getRuntime()而获得对当前Runtime对象的引用。Runtime类的大多数方法是实例方法。Runtime对象的作用主要有：执行外部命令.返回空闲内存.运行垃圾回收器.加载动态库等。Applets和其他不可信赖的程序由于没有引起一个安全异常(SecurityException)而不能调用任何的Runtime方法。下面的例子演示了怎样使用Runtime对象运行一个外部命令。以下是引用片段：

`Process process = Runtime.getRuntime().exec("cmd.exe /c dir").`
`process.waitFor();` : `Runtime.getRuntime()`返回当前应用程序的Runtime对象，该对象的`exec()`方法指示Java虚拟机创建一个子进程执行指定的可执行程序，并返回与该子进程对应的Process对象实例。通过Process可以控制该子进程的执行或获取该子进程的信息。第二条语句的目的是等待子进程完成后再往下执行。上面的程序在运行时执行dir命令。如果在Windows95/98下，命令格式可以写成"command.exe /c dir"。开关/C指明后面跟随的字符串是命令，并在执行命令后关闭DOS窗口。方法exec还可以打开一个不可执行的程序，但该文​​件存在关联的应用程序。以打开一个word文档Mydoc.doc文件为例，Java中可以有以下两种写法：以下是引用片段：
`exec("""cmd /E:ON /c start MyDoc.doc""").` `exec(" c:Program FilesMicrosoft Officeofficewinword.exe .mydoc.doc").` 在第一种方式中，被执行的命令是start Mydoc.doc，开关E:ON指定DOS命令处理器允许命令扩展，而start命令会开启一个单独的窗口执行所提供的命令。100Test 下载频道开通，各类考试题目直接下载。详细请访问 www.100test.com