

认证其实Unix很简单[2] PDF转换可能丢失图片或格式，建议
阅读原文

https://www.100test.com/kao_ti2020/167/2021_2022__E8_AE_A4_

[E8_AF_81_E2_80_94_E2_c103_167280.htm](https://www.100test.com/kao_ti2020/167/2021_2022__E8_AE_A4_E8_AF_81_E2_80_94_E2_c103_167280.htm) 2) 我们在Unix下操作文件权限很简单，文件权限分三组（本人，同组，别人）

，每组都是可读，可写，可执行。两个简单的系统调用chmod/chown就搞定。到了Windows下，如果是NTFS，如果要以程序的方式设置文件权限，呵呵，你需要先了解什么是：SID，什么是DAACL，什么是SACL，什么是ACE，还有十几相关的系统API函数等着你。（参看我的《以程序操作NTFS文件权限》）

你也许会觉得这么复杂的安全策略是让系统更安全的基础，可以自从Windows出现的那一天以后，在安全方面的表现就没有胜过Unix。这无疑是一个笑话。（插一句：Unix下的用户切换是相当简易方便的。而Windows下的用户切换会导致你需要退出当前用户的前台程序。这导致了Windows下的用户几乎无一例外地都会选择在超级用户的权限下工作/上网，这完全是让自己的机器在裸奔，所以，Windows下的病毒一旦在系统中运行就为所欲为了。

而Unix下，很少用户会以root身份操作本机，因为切换用户非常方便。）3) 在Unix下，用户有ID，用户组有ID，进程/线程都有ID。ID很简单易懂，就像我们的身份证一样。到了Windows下，用户标识叫Token，进程标识叫Handle（其实也就是一个DWORD的类型），我看到网上很多问题都在问Windows下的Handle的概念问题。我一直在想，为什么微软不取一个简单易懂的术语？要取得那么抽象，那么让人很迷惑。虽然这让Windows看起很NB，但也会因此加大了学习

复杂度。（Windows的开发学习复杂度要比Unix复杂多了，而且有太多的看似很高深的术语让人一头雾水）4）让我们再来看一下用户管理和程序所有者方面的东西。在Unix下，需要你做的是配置NIS服务器和NFS服务器（以Autofs自动mount），简洁，清楚。到了Windows下，与其相似的是一个叫Domain的东西（主域控制器），首先，为了加入域，你需要重启电脑（Unix下只需要配置/etc/nsswitch.conf文件来告诉本机的用户登录源，无需重启），而对开发者来说，Unix的这个配置对程序是完全透明的。而Windows的域用户和本地用户需要一个域名来区分。在程序中切换用户时，Unix只需要setuid/seteuid就行了，Windows对此有三个比较复杂的API：CreateProcessAsUser, ImpersonateLoggedOnUser, LogonUser，其复杂度就不用比较了。另外，在Domain方式下，你的Document and Setting目录下的文件，会全部放到Domain服务器上去，你在别的机器上登录时，需要下载这些文件。最后，我倒不担心你和网友的MSN聊天记录会因为你的登录而到处都是，我担心的是，你在这么复杂的管理环境中写出来的代码是否能让别人放心？：-(5) 在Unix下，要把自己的程序加入系统的启动服务只需要在/etc/init.d中配置就行了。写下一个有启动停止功能脚本，以特殊命名的方式链接到不同启动模式目录下就行了。Windows下加一个启动服务，如果你不编写程序，估计比较困难。6) 在Unix下，如果要取得系统的信息。只需要到/proc目录下去cat那堆文件。所有进程/线程的状态、命令行、内存/交换区使用情况、打开的文件描述符，等等，系统的CPU，内存，交换区，内存文件IO，分区，信息，网络，系统运行状态，系统设备，等

等，要有多详细就有多详细，而且完全是纯文本的，直接就可以看了。到了Windows下，要穷举系统当前进程的信息，就不是一样很容易的事，更别说要取得别的信息了。我很佩服微软把操作系统搞得那么复杂，又是注册表，又是安全策略，又是OLE，又是COM……。每次打开regedit.exe时，我根本不敢碰HKEY_CLASSES_ROOT，因为我看着里面那些成千上万个CLSID，我有点晕菜。自1995年以来，10来年过去了。微软推出了多种各式各样的技术。我还记得Visual Studio 6.0中还有两个叫做FoxPro和J的东西，FoxPro来自Foxbase，还有个几年，J好像就没有几年。ActiveX Control完全是一个失败的技术，而那个叫做VB的编程语言，今天看来，它的确毁了很多很有潜力的程序员。在COM出现的时候，不知道今天还有多少人还记得一个叫MTS的玩意？今天，不知道还有多少人记得有一个叫ODBC的东西？在这种复杂混乱的Windows世界中，是否让你疲于追赶？今天的.NET不知道又有多少技术会随着时间所沉淀？在Windows上面，我们学习了许多的失败技术或是说是过渡技术。而我们的Unix自从上世纪70年代以来，就没有多大的变化，而因为Unix应运而生的C语言直到今天依然光彩夺目。我相信这个30多年来久经考验还那么简单的Unix。 Unix就是这么简单，各位想在Unix下学开发的朋友，Windows那么复杂的操作系统都过来了，你还会怕这么简单的Unix么。 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