用VC获取其它程序的命令行参数 PDF转换可能丢失图片或格式，建议阅读原文 https://www.100test.com/kao_ti2020/273/2021_2022__E7_94_A8VC_E8_8E_B7_E5_8F_c97_273527.htm 我们都知道，在程序里获取命令行参数很简单，WinMain函数会以参数的形式传递给我们，或者可以调用API GetCommandLine 获取。但是GetCommandLine函数不接受参数，获取的只是自己程序的命令行参数。那么如果我们想获取别的应用程序的命令行参数应该怎么办呢？ 有的同学说，既然GetCommandLine只能获取本程序的命令行参数，我们可以在其它进程里插入一个Dll，在那个进程的地址空间调用GetCommandLine函数，然后传回来就可以了。这样好像有点儿不太友好。让我们想想还有没有别的办法。 我们想，自己的命令行参数既然随时都可以获取到，那么在该进程里一定有一个地方存放它。那么在哪儿呢？看一下GetCommandLine函数的反汇编代码，我们发现，原来世界是如此的美好！ 以下是WinXP系统的GetCommandLine函数反汇编代码：.text:7C812C8D GetCommandLineA proc near.text:7C812C8D mov eax, dword_7C8835F4 //dword_7C8835F4 就是命令行参数字符串的地址 //该指令机器码为 A1 F4 35 88 7C，从第2个字节开始的4个字节就是我们要的地址.text:7C812C92 retn.text:7C812C92 GetCommandLineA endp 既然知道了放在哪儿了，我们自己去拿就可以了。因为GetCommandLine函数的地址在各个进程内都是一样的，所以可以直接用我们进程里的地址。 win2000/xp系统很简单，98下稍微麻烦一点儿，需要进行一些简单的计算。 以下是GetCommandLine函数在win98下的汇编

代码： .text:BFF8C907 GetCommandLineA proc near .text:BFF8C907 mov eax, dword_BFFCADE4 .text:BFF8C90C mov ecx, [eax] .text:BFF8C90E mov eax, [ecx 0C0h] .text:BFF8C914 test eax, eax.text:BFF8C916 jnz short locret_BFF8C91E .text:BFF8C918 mov eax, [ecx 40h] .text:BFF8C91B mov eax, [eax 8] //算到这儿，才是我们想要的地址.text:BFF8C91E .text:BFF8C91E locret_BFF8C91E: . CODE XREF: GetCommandLineA F. .text:BFF8C91E retn 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