Win32调试API第二部分 PDF转换可能丢失图片或格式，建议阅读原文
https://www.100test.com/kao_ti2020/259/2021_2022_Win32_E8_B0_83_E8_AF_c98_259395.htm 理论:在前面一章中，我们学会了如何装载被调试的进程以及如何处理进程中发生的事件。为了有实际用途，我们的程序应具有修改被调试程序的能力。有好几个API函数用于这一目的。ReadProcessMemory该函数允许你去读指定的进程的内存。函数原型如下:
ReadProcessMemory proto hProcess:DWORD, lpBaseAddress:DWORD, lpBuffer:DWORD, nSize:DWORD, lpNumberOfBytesRead:DWORDhProcess 待读进程的句柄.lpBaseAddress 目标进程中待读内存起始地址。例如，如果你想要读目标 进程中从地址401000h开始的4个字节，该参数值应置为401000h。lpBuffer 接收缓冲区地址nSize 想要读的字节数。lpNumberOfBytesRead 记录实际读取的字节数的变量地址。如果对这个值 不关心，填入NULL即可
。WriteProcessMemory 是对应于ReadProcessMemory的函数，通过它 可以写目标进程的内存。其参数和ReadProcessMemory相同。 理解接下去的两个函数需要一些进程上下文的有关背景知识。在象Windows这样的 多任务操作系统中，同一时间里可能运行着几个程序。Windows分配给每个线程一个 时间片，当时间片结束后，Windows将冻结当前线程并切换到下一具有最高优先级的 线程。在切换之前，Windows将保存当前进程的寄存器的 内容，这样当在该线程再 次恢复运行时，Windows可以恢复最近一次线程运行的*环境*。保存的寄存器内容总 称为进程上下文。现在回到我们的主题。当一个调

试事件发生时，Windows暂停被调试进程，并保存其 进程上下文。由于进程被暂停运行，我们可以确信其进程上下文内容将保持不变。 可以用GetThreadContext来获取进程上下文内容，并且也可以用GetThreadContext 来修改进程上下文内容。这两个函数威力非凡。有了他们，对被调试进程你就具有象VxD的能力: 如改变其寄 存器内容，而在被调试程序恢复运行前，这些值将会写回寄存器中。在进程上下文中 所做的任何改动，将都会反映到被调试程序中。想象一下: 甚至可以改变eip寄存器 的内容，这样你可以让程序运行到你想要的任何地方! 在正常情况下是不可能做到这 一点的
。GetThreadContext proto hThread:DWORD, lpContext:DWORD hThread 你想要获得上下文的线程句柄lpContext 函数成功返回时用来保存上下文内容的结构指针
。SetThreadContext 参数相同。让我们来看看上下文的结构:
CONTEXT STRUCT ContextFlags dd ?
.----------.当ContextFlags包含CONTEXT_DEBUG_REGISTERS，返回本部分
.----------iDr0 dd ? iDr1 dd ? iDr2 dd ? iDr3 dd ? iDr6 dd ? iDr7 dd ?
.----------.当ContextFlags包含CONTEXT_FLOATING_POINT，返回本部分
.----------FloatSave

FLOATING_SAVE_AREA
.----------
----------.当ContextFlags包
含CONTEXT_SEGMENTS，返回本部分
.----------
---------- regGs dd ? regFs dd ? regEs dd ?
regDs dd ?
.----------
----------.当ContextFlags包
含CONTEXT_INTEGER，返回本部分
.----------
---------- regEdi dd ? regEsi dd ? regEbx
dd ? regEdx dd ? regEcx dd ? regEax dd ?
.----------
----------.当ContextFlags包
含CONTEXT_CONTROL，返回本部分
.----------
---------- regEbp dd ? regEip dd ? regCs
dd ? regFlag dd ? regEsp dd ? regSs dd ?
.----------
----------.当ContextFlags包
含CONTEXT_EXTENDED_REGISTERS，返回本部分
.----------
---------- ExtendedRegisters db
MAXIMUM_SUPPORTED_EXTENSION dup(?) CONTEXT
ENDS 可以看出，该结构中的成员是对实际处理器的寄存器

的模仿。在使用该结构之前 要在ContextFlags 中指定哪些寄存器组用来读写。如要访问所有的寄存器， 你可以置ContextFlags 为CONTEXT_FULL 。或者只访问regEbp, regEip, regCs, regFlag, regEsp 或 regSs, 应置ContextFlags 为CONTEXT_CONTROL 。 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