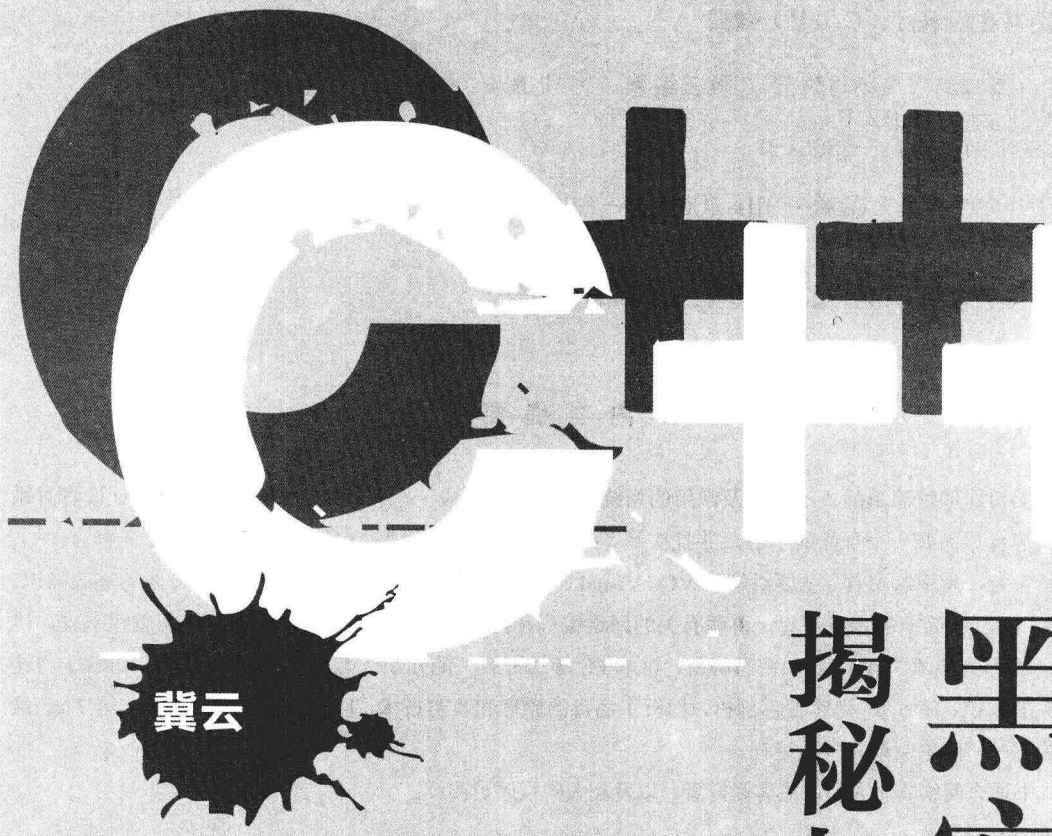
# C++黑客编程

冀云 编著

讲解了Windows安全和网络编程知识，如注册表、系统服务、文件读写、进程和线程

介绍了Windows下PE格式、调试技术、挂钩技术等各种软件安全知识

人民邮电出版社
POSTS & TELECOM PRESS




# C++黑客编程揭秘与防范

冀云

人民邮电出版社
北京

**图书在版编目（CIP）数据**

C++黑客编程揭秘与防范 / 冀云编著. -- 北京 : 人民邮电出版社, 2012.6
ISBN 978-7-115-28064-0

Ⅰ. ①C… Ⅱ. ①冀… Ⅲ. ①C语言－程序设计 Ⅳ. ①TP312

中国版本图书馆CIP数据核字(2012)第081089号

## 内 容 提 要

本书旨在通过简单的语法知识及常用的系统函数编程，完成一些有特定功能的安全工具，让读者对系统知识等各方面有一个全面的了解，并且在笔者的带领下一步步完成书中的实例。本书主要内容为：

第 1 章了解黑客编程，主要讲解了 VC（Visual C++的缩写）和 Windwos 下安全编程方面的基础知识。第 2 章 从剖析简单的木马说起，讲解有关的网络编程和协议知识。第 3 章 Windows 应用编程基础，讲解 API 编程的技术。第 4 章 加密与解密，讲解 PE 等加密有关的知识。第 5 章 HOOK 编程，讲解了与钩子有关的知识。第 6 章 黑客编程剖析，剖析了病毒的原理和攻防技术，以及安全工具的开发。第 7 章 最后的旅程——简单驱动开发及逆向。

本书适合网络安全人员、黑客爱好者，以及相关的程序员阅读。

**C++ 黑客编程揭秘与防范**

◆ 编　　著　冀　云
责任编辑　张　涛

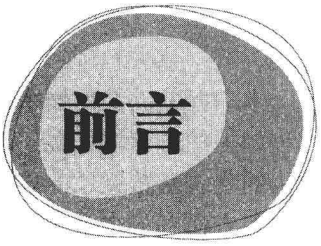
◆ 人民邮电出版社出版发行　　北京市东城区夕照寺街 14 号
邮编　100061　　电子邮件　315@ptpress.com.cn
网址　http://www.ptpress.com.cn
北京鑫正大印刷有限公司印刷

◆ 开本：787×1092　1/16
印张：17.25
字数：406 千字　　2012 年 6 月第 1 版
印数：1－3 500 册　　2012 年 6 月北京第 1 次印刷

ISBN 978-7-115-28064-0

定价：39.00 元

读者服务热线：(010)67132692　印装质量热线：(010)67129223
反盗版热线：(010)67171154




什么是黑客？摘自百度百科中的一句话，“黑客一词，原指热心于计算机技术，水平高超的电脑专家，尤其是程序设计人员”。其实，黑客并不利用自己已有的技术去对他人的系统进行渗透并破坏。黑客的为人处世也非常低调，不会整天拿着别人写好的工具去入侵网站或“抓肉鸡”，做这么没意义的事。如果是黑客天天做这些事，怎么可能有多余的时间真正地研究技术？

### 编程、破解、入侵

编程、破解、入侵是黑客所掌握的技能，但是后两者都是以前者的编程为基础的。破解别人的程序是站在写程序的角度去考虑的，而入侵依靠的是系统的漏洞，发掘漏洞同样是需要编程知识、系统底层知识和调试技术。也就是说，想做一名黑客，在自身的知识体系中编程知识是占据很大份额的。也就应了网上的一句话——“不会编程的黑客就不是黑客”。

### 黑客编程与普通编程的区别

黑客编程，其实也就是利用普通的编程技术编写一些黑客工具，或者是网络安全工具。该方面的知识是一把双刃剑，无论是编写黑客工具，还是编写安全工具，都离不开这些知识。本书的重点是通过简单的编程知识配合良性的实例让大家了解黑客编程，并对漏洞进行防范，希望大家正确对待技术的合理应用。

### 本书的前置知识

阅读本书需要有C、C++语言的基础知识，本书并不是一门编程语言关于语法知识的教科书。如果读者希望能够顺利阅读此书，至少要有阅读C、C++语言编程的能力。如果没有C、C++语言的基础，而有其他语言的基础，那么也是没有问题的。在掌握了编程思想，或者会使用API函数后，用自己熟悉的语言进行相应的开发也是可以的。但是，为了将来能更好、更深入地学习系统的底层知识，建议学习C、C++和汇编语言。

### 本书适合的读者

长期研究黑客工具的程序员，且有打算自己开发网络安全工具的人员。

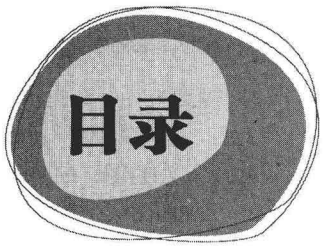
掌握编程基本语法想要进行相关编程实践的读者。

**本书的定位**

本书并非高深的技术性书籍，市场上较深入的研究系统底层、加解密相关的、网络安全编程相关的书籍非常多。但是，很多并非入门类的书籍。本书旨在，通过简单的语法知识及常用的系统函数完成一些有特定功能的安全工具。在读者有基础的前提下，让读者对系统知识等各方面有一个全面的了解，并且在笔者的带领下一步步地完成书中的实例，也可以增强读者的动手能力，源程序下载地址 www.ptpress.com.cn。

**需要声明的是：**

本书的内容是帮助读者提升防范黑客攻击的能力和技术，普及网络安全知识，绝不是为那些怀有不良动机的人提供支持，也不承担因为技术被滥用所产生的连带责任，请读者自觉遵守国家相关法律。



# 第1章 黑客编程入门

你是否曾经在用别人开发的工具尝试“入侵”，你是否希望开发出自己的黑器……相信很多人有着这种近似相同的经历。本章将简单介绍黑客编程及工具开发。如果你是初学编程，如果你从来没有接触过黑客软件的开发，如果你急于想了解黑客编程方面的知识……那么就请继续往下阅读。

## 1.1 编程语言和开发环境的选择

初学者刚开始学习编程语言最头疼的问题就是如何选择编程语言及合适的开发环境，下面就来具体介绍一下。

有人认为学编程就是学编程语言，而 VC、VB 这样的开发环境只是工具，不需要学。这个想法是错误的，因为开发环境提供了很多开发工具，如 VC 这个集成开发环境就提供了与之对应的 PSDK、MFC 等。除了语言以外，要开发特定的软件是需要开发包和开发工具支持的。况且，编程语言也是一种工具，用于和计算机进行交流的工具。所以我们既要学习编程语言，也要学习开发工具。

对于选择哪种编程语言或者开发环境其实也没有特定的标准。有这样一句话，“真正的程序员用 VC，聪明的程序员用 Delphi，用 VB 的不是程序员”。笔者却并不这么认为，因为在很多编程的书籍上常常这样提醒并告诫学习者，编程的精髓是“算法”，而语言是用来描述算法的。因此，大家也不必因为无法选择而无从下手。

黑客一般都掌握多种编程语言，他们不但掌握着与底层相关的如汇编、C 之类的编程语言，而且还掌握很多脚本语言，如 Python、Perl、Ruby……很多黑客在发现 0Day 以后用 Perl 或者 Python 来写 POC；MSF 使用的是 Ruby 来进行开发 Exploit；有的黑客在反病毒时竟然写个批处理就搞定了……对于黑客来说，一切语言都是服务于自己的思想的，只要能快速实现自己的想法，能完成自己所要完成的功能就行，从不拘泥于任何语言和工具。在网上有很多学习不同编程语言的人们之间经常互相攻击，这其实是一种极端的行为，大家还是理性地对待这些问题比较好。

前面说了这么多，仿佛是在绕圈子，一直没有介绍到底应该选择什么编程语言和开发环境。我们这里选择使用 C 语言作为黑客编程的学习语言，选择 VC6（Visual C++ 6.0 的缩写）来作为我们的开发环境。VS 6 相对于 Visual Studio 2005、Visual Studio 2008 和 Visual Studio

2010 之类的开发环境来说要小巧很多，当前是可以满足我们的开发需求的。选择 C 语言的原因是由于 Windows 的 API 都是用 C 语言定义的，相对于使用其他编程语言会方便很多。笔者认为在 VB 下使用 API 就非常不方便，尤其是涉及指针这个概念的时候。除了 VC6 以外，还需要下载新版的 PSDK，因为 VC6 中包含的 PSDK 过于旧，有些新的 API 我们无法通过包含头文件而直接使用，因此这个也是必须的。

### 1.1.1 何为 SDK、API 和 MFC

既然选择 VC 作为开发环境，那么先来了解一下 VC 开发环境中今后会用到的一些工具的概念，这些概念都相对比较简单，常见的概念有 SDK、API 和 MFC。

SDK 是 Software Develop Kits 的缩写，即软件开发工具包。SDK 是一个泛指，比如对视频采集卡进行二次开发，那么视频采集卡会提供 SDK；再比如对动态令牌进行二次开发，那么动态令牌也会提供 SDK。操作系统为了程序员在其平台下开发应用程序也会提供 SDK，我们对系统提供的开发包称之为 PSDK。PSDK 是 Platform SDK 的意思，也就是平台 SDK。对于我们来说，平台就是 Windows 操作系统。Windows 下的 PSDK 包含了进行 Windows 软件开发的文档和 API 函数的输入库、头文件等一些与 Windows 开发相关的工具。PSDK 是一个单独的开发包，不过每个不同版本的 VC 和其他一些开发环境中也已经包含了它。

API 是 Application Programming Interface 的缩写，即应用程序接口。不同的 SDK 提供不同的 API。PSDK 提供的 API 就是操作系统提供的开发应用程序的接口，比如 Windows 系统下删除文件的 API 函数是 DeleteFile()；再比如 Windows 系统下移动文件的 API 函数是 MoveFile()，而其他一些供二次开发的 SDK 中附带的 API，也是为了进行开发应用程序而提供的接口。

MFC 是 Microsoft Foundation Class 的缩写，即微软基础类库。它是微软为了简化程序员的开发工作量所提供的基于 C++类的一套面向对象的库，它封装了常见的 API 函数，使得开发较为方便。

我们在书中会用到 API 进行开发，也会使用 MFC 进行开发。不过对于 MFC 的使用，基本上用在与界面相关的部分，一般是简单地带过，不会进行过多的讨论。我们的重点是放在 API 函数的使用上。关于 MFC 的相关内容，还请大家自行参考学习。

### 1.1.2 VC6 和 SDK 的配置

新版的 PSDK（Windows Server 2003 SP1 Platform SDK）的下载地址为 http://www.microsoft.com/downloads/en/details.aspx?FamilyID=eba0128f-a770-45f1-86f3-7ab010b398a3。如果此地址过期的话，请大家在网上自行搜索并下载。

SDK 和 VC6 互相是独立的，不需要安装在同一个目录下，根据自己的实际情况安装就可以了。在安装好 VC6 和新版的 SDK 后，需要在 VC6 中进行相应的设置才能使用新版的 SDK，否则 VC6 仍然使用其自带的旧的 SDK。SDK 和 VC6 的安装步骤这里就不介绍了（提示：请把 VC6 安装完整，VC6 会提供一些代码，对我们的学习是非常有帮助的），下面介绍新版的 SDK 如何配置才能在 VC6 中使用。

启动 VC6，单击菜单“Tools”->“Options”命令，打开“Options”对话框，如图 1-1

所示。

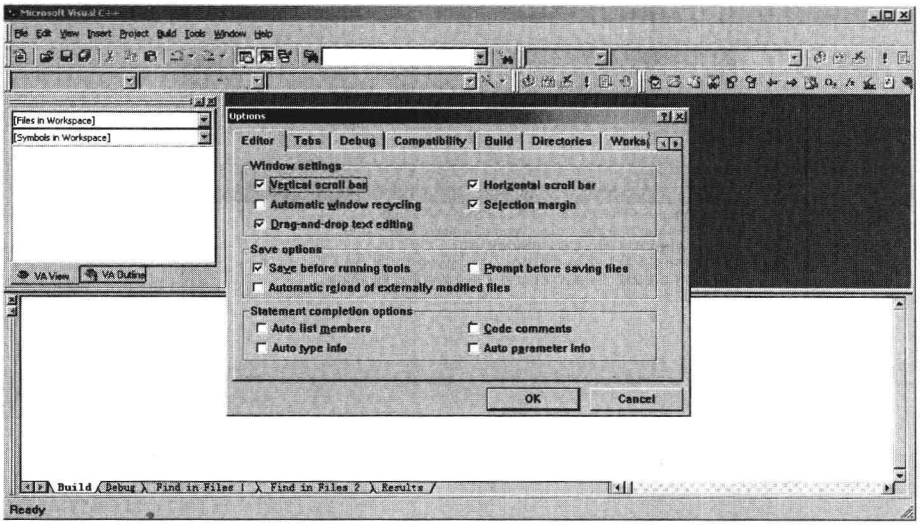



图 1-1 “Options”对话框

选择“Directories”选项卡，在“Show directories for”下拉列表中选择“Include files”，选项并在“Directories”列表框中添加新的 PSDK 头文件的目录，放在列表的最上面，如图 1-2 所示。

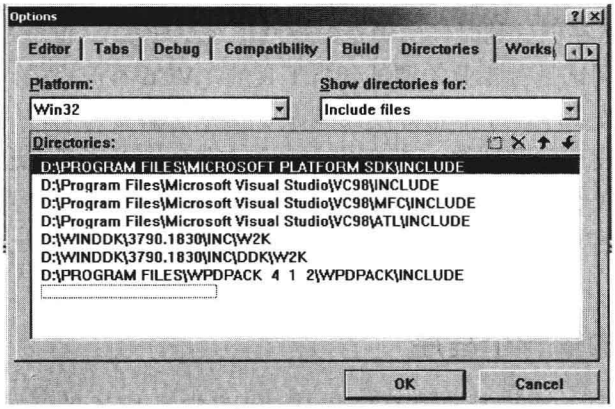



图 1-2 头文件的路径

在“Show directories for”下拉列表中选择“Library files”选项，并在“Directories”列表框中添加新的 PSDK 库文件的目录，放在列表的最上面，如图 1-3 所示。

切记要把所添加的目录放到列表的最上边，因为在 VC 编译代码的时候会搜索这些目录里的文件，如果随便放，编译器会因找不到相关 API 函数定义而报函数未定义的错误。

另外，还必须下载一个 MSDN。MSDN 即 Microsoft Developer Network，它是微软开发的联机帮助文档，可以帮助我们在使用 API 的时候进行快速的查阅，以方便我们对 API 的使用和理解。但是 MSDN 里的内容全部都是英文的，如果你英文不太好可以借助搜索引擎来学

习 API 的使用。本书只对所提到的 API 函数常用的参数进行介绍，其他参数需要大家自行进行学习。

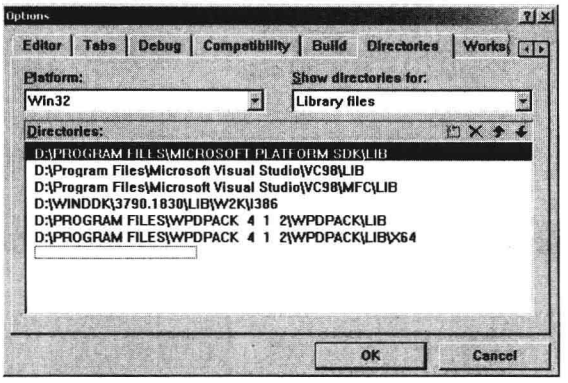



图 1-3　库文件的路径

# 1.2　应用程序的调试

在开发程序的过程中，除了编码以外还需要对程序进行调试，当编写的程序出现问题后，就要对程序进行调试。调试不是仅使用一个 printf()或 MessageBox()进行简单的输出来观察某个函数的返回值（虽然在调试的时候的确是对返回值观察较多），也不是对某个变量、某一时间的具体值的输出。调试是有专业的调试分析工具的，VC6 不但提供代码编辑、代码编译、编译连接等功能，还提供了一个非常好用的调试工具。在编写完代码后，如果程序输出的结果是未知的，或者是没有预测到的，都可以通过调试来对代码的逻辑进行分析，以找到问题的所在。掌握调试的技能，对软件的开发有非常大的帮助。掌握好的调试工具，对于调试者来说，也同样会起到事半功倍的作用。下面通过一个简单的例子了解一下 VC6 提供的调试功能吧。

## 1.2.1　编写我们的第一个程序

下面介绍用 VC6 写一个控制台版的 HelloWorld 来学习 VC6 的开发。也许大家认为这个程序很简单，但是请记住，我们的重点是要介绍 VC6 这个集成开发环境中提供的调试功能。

启动 VC6，单击菜单“File”->“New”命令，在弹出的对话框中选择“Projects”选项卡，然后在左边的列表框中选择“Win32 Console Application”选项，在“Project Name:”文本框中填写“HelloWorld”，如图 1-4 所示。

单击“OK”按钮，出现如图 1-5 所示窗口。

选择“An empty project”单选项，单击“Finish”按钮，然后在弹出的对话框中单击“OK”按钮。

单击菜单“File”->“New”命令，选择“Files”选项卡，在左边的列表中选择“C++ Source File”选项，在右边的“File”文本框中填写“HelloWorld”，如图 1-6 所示。

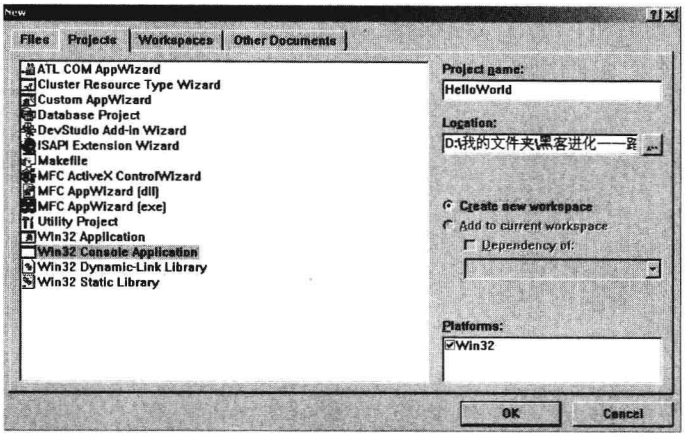



图 1-4 “Projects”选项卡

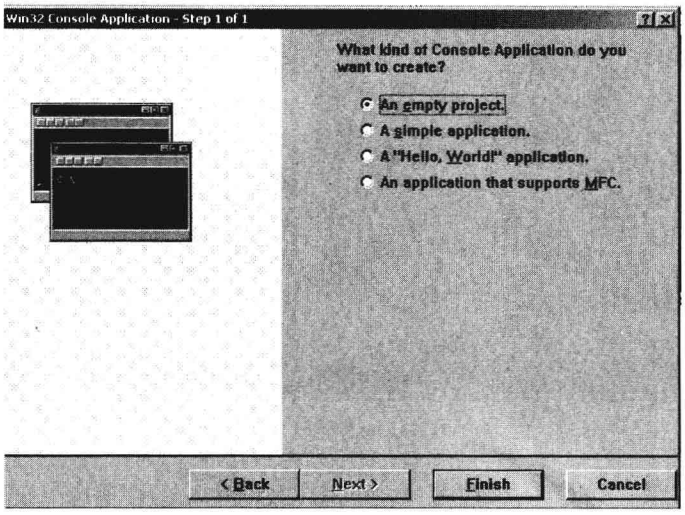



图 1-5 “Win32 Console Application”项目向导

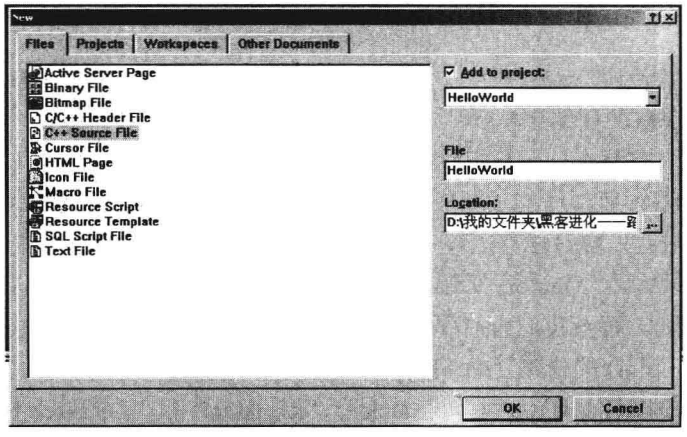



图 1-6 “Files”选项卡

单击“OK”按钮就可以进行代码编辑了。

在代码编辑处录入如下代码：

```
#include <stdio.h>

int main()
{
    printf("Hello World ! \r\n");

    return 0;
}
```

按 F7 键进行编译连接（按 Ctrl + F7 组合键是只编译不进行连接），按 Ctrl + F5 组合键进行运行，如图 1-7 所示。


图 1-7 “Hello World”运行界面

这就是我们值得纪念的第一个程序。这个程序很简单，有 C 语言基础的读者应该都能看懂，这里就不进行介绍了。如果看不懂，请先找本关于 C 语言入门的书学习一下。

## 1.2.2 用 VC6 调试第一个程序

现在来学习如何使用 VC6 对第一个程序进行调试。在代码编辑状态下，按下键盘上的 F10 键，进入调试状态，如图 1-8 所示。

常用的调试窗口有两个，一个是“Watch”窗口（标注“1”的那个窗口），一个是“Memory”窗口（标注“2”的那个窗口）。打开“Watch”窗口的方法是单击“View”->“Debug Windows”->“Watch”命令（或按 Alt + 3 组合键）打开。打开“Memory”窗口的方法是单击“View”->“Debug Windows”->“Memory”命令（或按 Alt + 6 组合键）打开。“Watch”窗口用来监视我们感兴趣的变量，而当我们有时无法通过变量的值进行判断时，就需要借助“Memory”窗口中的值，比如，指针的值来进行判断。

除了这两个窗口以外，还有“Call Stack”、“Register”和“Disassembly”这 3 个窗口，分别如图 1-9、图 1-10 和图 1-11 所示。

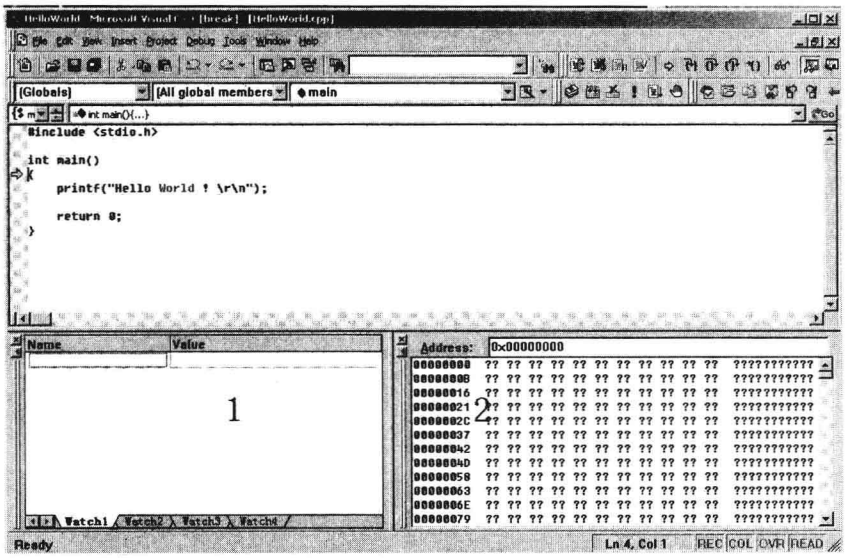



图 1-8 VC6 处于调试状态

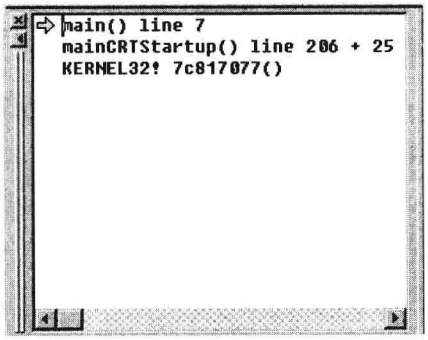



图 1-9 “Call Stack”窗口

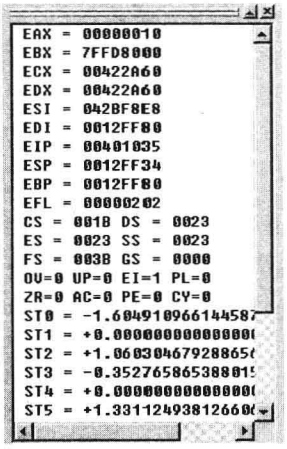



图 1-10 “Register”窗口

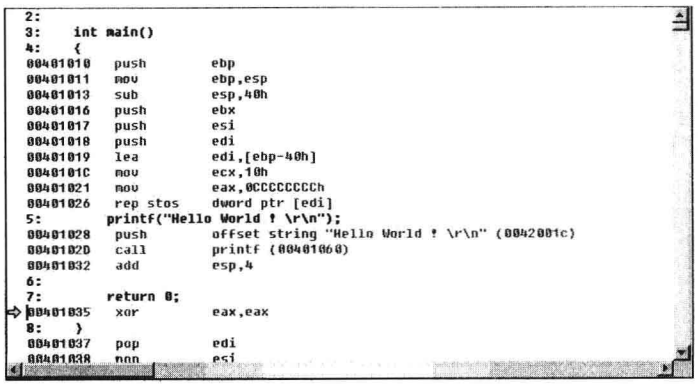



图 1-11 “Disassembly”窗口