实用信息安全技术

SHIYONG XINXI ANQUAN JISHU

雷敏 王剑锋 李凯佳 杨义先 编著

国防工业出版社
National Defense Industry Press

实用信息安全技术

雷　敏　王剑锋　李凯佳　杨义先　编著

国防工业出版社

·北京·

内 容 简 介

信息安全是目前社会关注的热点。本书主要内容包括实用信息安全技术和常见信息安全工具的使用方法。重点介绍现代密码学、网络安全、信息系统安全、数字内容安全和Web安全等方面常见的攻击技术、工具以及防范措施。

本书可作为高等院校信息安全和计算机等相关专业学生的课程实验、课程设计、项目实践、项目实训教材，同时可作为各培训机构实践教材。

图书在版编目(CIP)数据

实用信息安全技术／雷敏，王剑锋，李凯佳编著.
—北京：国防工业出版社，2014.1
ISBN 978-7-118-09287-5

Ⅰ.①实… Ⅱ.①雷…②王…③李… Ⅲ.①信息安全-安全技术 Ⅳ.①TP309

中国版本图书馆CIP数据核字(2014)第015555号

※

国防工业出版社出版发行
(北京市海淀区紫竹院南路23号 邮政编码100048)
北京奥鑫印刷厂印刷
新华书店经售

*

开本 787×1092 1/16 **印张** 13 **字数** 295千字
2014年1月第1版第1次印刷 **印数** 1—3000册 **定价** 28.00元

(本书如有印装错误，我社负责调换)

国防书店：(010)88540777 发行邮购：(010)88540776
发行传真：(010)88540755 发行业务：(010)88540717

哈尔滨理工大学 藏書 图書馆

前　言

信息安全的重要性不言而喻。随着电子政务、电子商务的进一步普及，政府、企业以及各种社会机构对信息安全专业人才的需求逐渐增加，信息安全人才的总量和结构远远不能满足需求，如何培养复合性和专业性人才以满足日益增长的人才需求，并适应日益复杂的网络应用环境是当前的重要任务之一。

信息安全专业人才的培养绝非易事，原因在于信息安全涉及到密码学、数学、计算机、操作系统、通信工程、信息工程、数据库等多门学科的交叉，一方面知识体系庞杂、难于掌握；另一方面实践性很强，技术发展更新非常快。高校在长期的教学实践过程中，形成了一种共识：构建合理、完善的实验课程体系是帮助学生掌握理论体系、培养动手实践能力的有效途径。诸多高校在网络信息安全专业建设过程中纷纷加大了对实验室建设的投入，加强了实验课程体系的建设，在学生培养过程中为学生提供动手实践环境。与此同时，作为高校学历教育的补充，社会培训机构也有各种信息安全技能培训，为行业内从业人员或高校毕业生提供技能培训，以适应信息安全专业快速发展的需要。

本书是作者多年信息安全实验教学的积累，并根据社会需求和学生特点的变化不断调整实验内容和实验教学方法。本书重点突出实用性，让学生快速掌握网络信息安全基本技能，并掌握工具的使用方法。基于上述考虑，本书重新梳理了信息安全实验教学内容，结合典型教学案例和教学工具的使用方法，目的是引导学生快速掌握使用工具。由于篇幅的限制，不可能对所有的技术、理论、算法和工具均进行深入的介绍和剖析，只能选择一些典型的、通用性的工具来介绍，关键是让学生掌握基本方法。这本质上是一种以点带面的方法：学生只要深入理解本书中所讲授的方法和工具，同样能够举一反三，自己去研究、理解和解决其他可能遇到的问题。

本书在编写过程中引用了来自互联网的一些素材和工具，目的是服务教学，为学生提供更优秀、更便于理解的教学素材和工具。

本书受到中国科学技术协会的资助，并得到中国密码学会的大力支持，在此表示感谢。同时，由于作者水平有限，书中肯定存在各种问题，欢迎读者在使用过程中予以批评指正。

作者

2013 年 10 月

目　录

此为试读，需要完整PDF请访问：www.ertongbook.com

第 1 章　密码学及应用

密码学是信息安全核心技术之一。人类的文明史、战争史无不伴随着密码领域两股力量的生死博弈：一股力量是密码编码学，关心的是怎样使密码变成牢不可破的坚盾，而另一股力量是密码破译学，则处心积虑地打造无坚不摧之利矛来摧毁任何坚固的密码。正是由于前者的智慧，古今中外、形形色色的密码算法浩如烟海，而后者的思想不仅留下了玛丽女王的叹息，也使得第一次世界大战、第二次世界大战的战局更加扑朔迷离。这场“道高一尺，魔高一丈”的对弈虽历经千年也未分出胜败，今后的斗争恐将更加激烈。从事信息安全技术的工作人员作为未来博弈的两方之一，只有熟悉双方的发展历史和趋势，才能在斗争中占据更加主动的位置。

本章由古典密码入手，通过使用替换加密方法，消除初学者对密码的神秘感，进而引导学生采用现代技术构建更复杂的密码体制，本章内容覆盖了古典密码学实验、对称密码实验和非对称密码实验等内容。

实验 1.1　古典密码之凯撒密码

1.1.1　实验原理

凯撒密码作为一种最古老的对称加密体制，在古罗马就已经很流行，它是一种替代密码，通过将字母按顺序推后 3 位起到加密作用，如将字母 A 换作字母 D，将字母 B 换作字母 E。由此可见，移位位数就是凯撒密码加密和解密的密钥。这种密码较易破解，只需简单地统计字频就可以破译。

凯撒密码的基本思想是通过把字母移动一定的位数来实现加密和解密。实验原理是将明文中的每个字母用字母表中后面第 k 个字母替代。它的加密过程可以表示为以下函数：$E(m)=(m+k)\ \mathrm{mod}n$。其中，$m$ 为明文字母在字母表中的位置数；n 为字母表中的字母个数；k 为密钥；$E(m)$ 为密文字母在字母表中对应的位置数。同理可得解密过程。

1.1.2　实验目的

通过对凯撒密码的 C ++ 源程序进行修改，了解和掌握对称密码体制的运行原理和编程思想。

1.1.3　实验环境

安装 Windows XP 操作系统的计算机，且其上安装 C ++ 语言编译环境，如 VC6.0 或 VC2005。

1.1.4 实验内容

本实验根据凯撒密码算法原理，编写循环移位密码算法。代码演示了在选择不同密钥时，加密和解密的结果。

请读者分析代码，了解加密和解密实现的具体过程。

请根据实验原理，创建明文信息，并选择一个密钥，编写循环移位密码算法程序，实现加密和解密操作。

1. 创建明文

过程可描述为：

```
unsigned char *str = (unsigned char *)malloc(sizeof(char)*1024)
for(i=0;i<1025;i++)
    scanf("%c",str+i);
    if(str[i] == '\n')
        str[i]='\0';
        break;
```

2. 选择密钥

过程可描述为：

```
scanf("%c",code);
    code[1]='\0';
```

3. 加密过程

```
int Encrypt(unsigned char* str,unsigned char* code)
for(i=0;i<str_length;i++)
    str[i]+=key;
    if(str[i]>122)
    str[i]-=26;
return CRYPT_OK;
```

4. 解密过程

过程可描述为：

```
int key = code[0]-97;
for(i=0;i<str_length;i++)
    str[i]-=key;
    if(str[i]<97)
        str[i]+=26;
return CRYPT_OK;
```

5. 算法编辑过程

1）打开文件

打开 VC ++6.0 编辑界面，在“资源管理器”或“我的电脑”中找到已存在的循环移位算法。双击此文件名，自动进入 VC 集成环境，并打开该文件，程序显示在如图 1.1.1 所示的编辑窗口中，也可选择“文件”菜单下的“打开”命令、或快捷键 Ctrl + O 从中选择所需

此为试读，需要完整PDF请访问：www.ertongbook.com

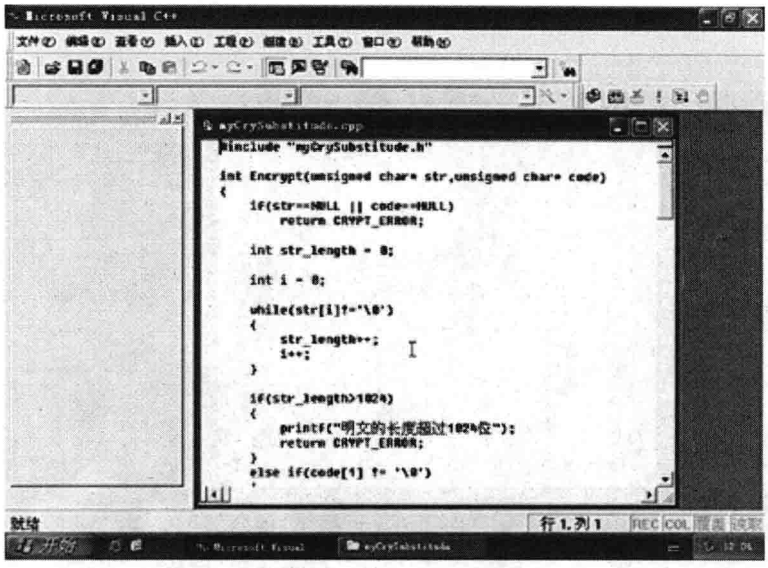

图 1.1.1　打开程序

文件。

2）运行文件

单机工具栏上的 Build 按钮，或快捷键 F7，即可通过编译、链接生产目标后缀为 obj 的目标文件，如图 1.1.2 所示。若链接成功，则可生成一个后缀为 exe 的可执行文件。

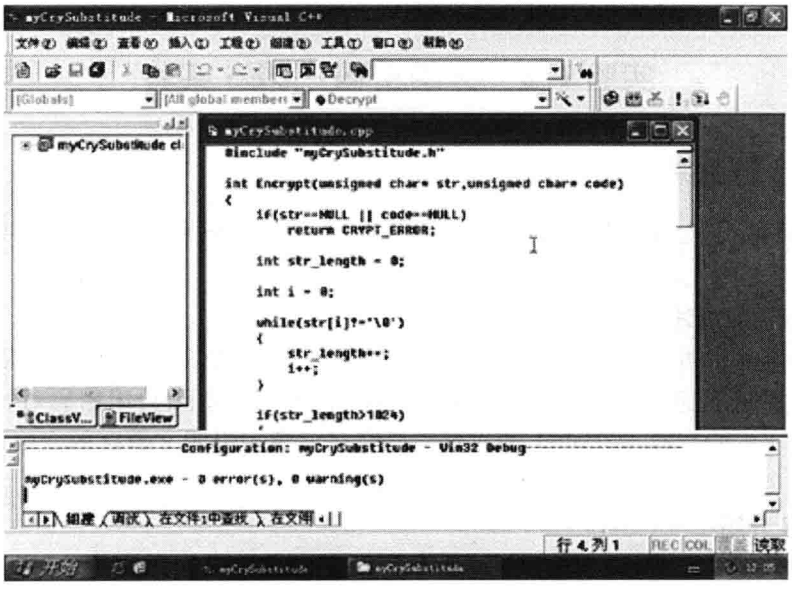

图 1.1.2　调试程序

3）查看运行结果

单击工具栏上的 Execute 按钮或快捷键 F5，将显示程序运行结果如图 1.1.3 所示。

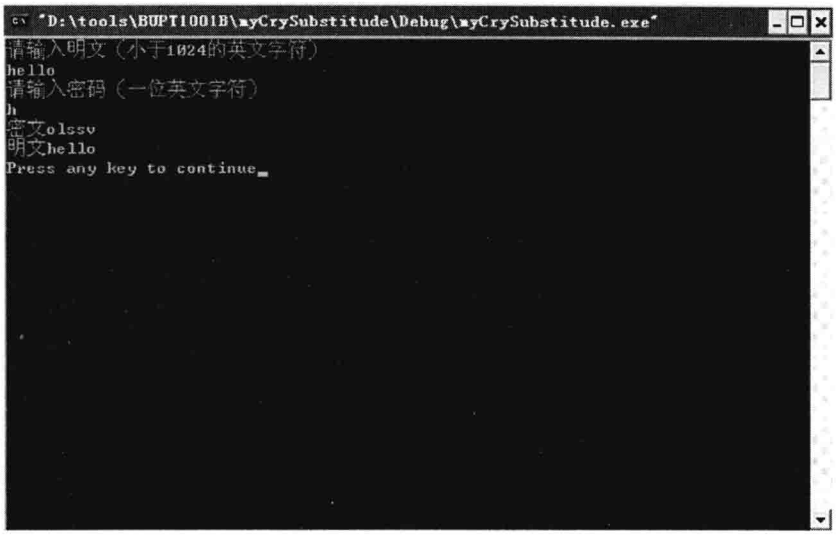

图 1.1.3　程序运行结果

1.1.5　思考题

思考如何在凯撒密码算法的基础上，将 3 位循环移位的密码算法修改为 4 位循环移位的密码算法，并实现加密和解密的过程？

实验 1.2　分组密码 - DES 实验

1.2.1　实验原理

分组密码易于构造拟随机数生成器、流密码、消息认证码和杂凑函数等，还可以成为消息认证技术的核心组成部分。

1. 分组密码概述

分组密码是将明文消息编码表示后的数字序列，划分为长度为 m 的分组，各组分别在密钥 k 的控制下变换成等长的输出数字序列，在相同密钥下，分组密码对长为 m 的输入明文组所实施的变换是等同的，所以只需研究对任意一组明文数字的变换规则。这种密码实质上是字长为 m 的数字序列的代换密码。

2. DES 介绍

DES 是 Data Encryption Standard 的缩写。它是由 IBM 公司研制的一种加密算法，美国国家标准局于 1977 年公布把它作为非机要部门使用的数据加密标准。分组密码，其中的消息被分成定长的数据分组，每一分组称为 M 或 C 中的一个消息，分组长度为 16bit，因含有 8 个奇偶校验位，所以实际密钥长度为 56bit。在 DES 中，有 $M = C = \{0,1\}^{64}$，$K = \{0,1\}^{56}$ 也是 DES 加密和解密算法输入 64bit 明文或密文消息和 56bit 密钥，输出 64bit 密文或明文消息。

3. DES 运算步骤

(1) 对输入分组进行固定的初始置换,写为(L_0,R_0)←IP(Input Block),这里的L_0,R_0分别称为左半边分组和右半边分组,都是 32 比特的分组。

(2) 然后是具有相同功能的 16 轮变换,每轮中都有转换和代换的运算。第 16 轮变换的输出分为左右两半,共被交换次序。

(3) 最后经过一个逆初始置换,从而产生 64 比特的密文。

加密和解密都用这 3 个步骤。

1.2.2 实验目的

通过对 DES 算法的代码编写,了解分组密码算法的设计思想和分组密码算法的工作模式。

1.2.3 实验环境

安装 Windows XP 操作系统的计算机,其上安装 VC6.0 以上版本的编译器。

1.2.4 实验内容

1. 置换规则表

在 DES 算法中需要置换表,其功能是把输入的 64 位数据块按位重新组合,并把输出分为L_0、R_0两部分,每部分各长 32 位,其置换规则如下:

58,50,42,34,26,18,10,2,60,52,44,36,28,20,12,4,

62,54,46,38,30,22,14,6,64,56,48,40,32,24,16,8,

57,49,41,33,25,17,9,1,59,51,43,35,27,19,11,3,

61,53,45,37,29,21,13,5,63,55,47,39,31,23,15,7,

即将输入的第 58 位换到第一位,第 50 位换到第 2 位,…,依此类推,最后一位是原来的第 7 位。L_0、R_0则是换位输出后的两部分,L_0是输出的左 32 位,R_0是右 32 位,例:设置换前的输入值为$D_1D_2D_3\cdots D_{64}$,则经过初始置换后的结果为:$L_0 = D_{58}D_{50}\cdots D_8$;$R_0 = D_{57}D_{49}\cdots D_7$。

经过 16 次迭代运算后,得到L_{16}、R_{16},将此作为输入,进行逆置换,即得到密文输出。逆置换正好是初始置换的逆运算。例如,第 1 位经过初始置换后,处于第 40 位,而通过逆置换,又将第 40 位换至第 1 位,其逆置换规则如下:

40,8,48,16,56,24,64,32,39,7,47,15,55,23,63,31,

38,6,46,14,54,22,62,30,37,5,45,13,53,21,61,29,

36,4,44,12,52,20,60,28,35,3,43,11,51,19,59,27,

34,2,42,10,50,18,58 26,33,1,41,9,49,17,57,25,

2. 密钥生成算法

根据 DES 算法原理分析密钥的生成方法,分析代码中的密钥。

(1) 生成加密或者解密用的 16 轮子密钥,基本过程如下。

```
ulong32 i, j, l, m, n, kn[32];
    unsigned char pc1m[56], pcr[56];
```

```
    for (j =0; j < 56; j ++)
        l = (ulong32)pc1[j];
        m = l & 7;
pc1m[j] = (unsigned char)((key[l >> 3U] & bytebit[m]) ==
          bytebit[m] ?1 : 0);
    for (i =0; i < 16; i ++) {
        if (edf == DE1)
            m = (15 - i) << 1;
        else m = i << 1;
            n = m + 1;
            kn[m] = kn[n] = 0L;
```

(2) 子密钥的前半部分循环移位,基本过程如下。

```
for (j =0; j < 28; j ++)
    l = j + (ulong32)totrot[i];
    if (l < 28)
        pcr[j] = pc1m[l];
    else
        pcr[j] = pc1m[l - 28];
```

(3) 子密钥的后半部分循环移位,基本过程如下。

```
for (/*j = 28 */; j < 56; j ++) {
        l = j + (ulong32)totrot[i];
    if (l < 56) {
        pcr[j] = pc1m[l];
    } else {
        pcr[j] = pc1m[l - 28];
```

(4) 对48bit密钥进行置换,基本过程如下。

```
for (j =0; j < 24; j ++)  {
    if ((int)pcr[(int)pc2[j]] ! = 0) {
        kn[m] |= bigbyte[j];
    }
    if ((int)pcr[(int)pc2[j +24]] ! = 0) {
            kn[n] |= bigbyte[j];
```

3. 程序实现过程

(1) 打开 VC ++6.0 编辑界面,建立一个工程,单击"文件"→"新建",如图 1.2.1 所示。选择 Win32 Console Application,命名工程名称,选择保存位置,如图 1.2.2 所示。点击"确定"按钮,进入下一步,看到如图 1.2.3 的提示界面。建立一个空工程,点击"完成"按钮,显示所创建工程的信息。

(2) 打开工作区,选择 FileView 选项卡,如图 1.2.4 所示。右键点击工程文件名称,选择"添加文件到工程"。可到相关路径中找到相应文件(G_des. c,test. cpp,des. h),如图

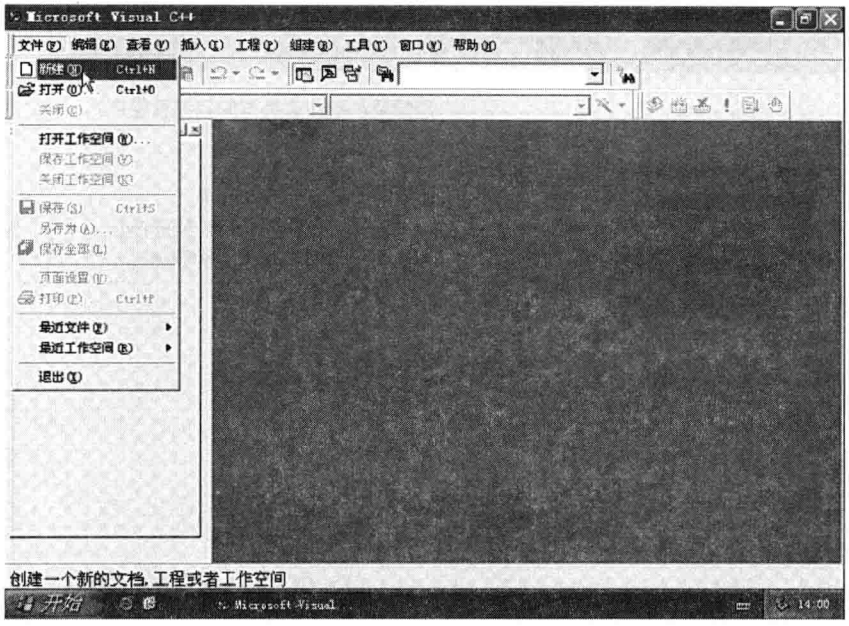

图 1.2.1　VC 运行界面

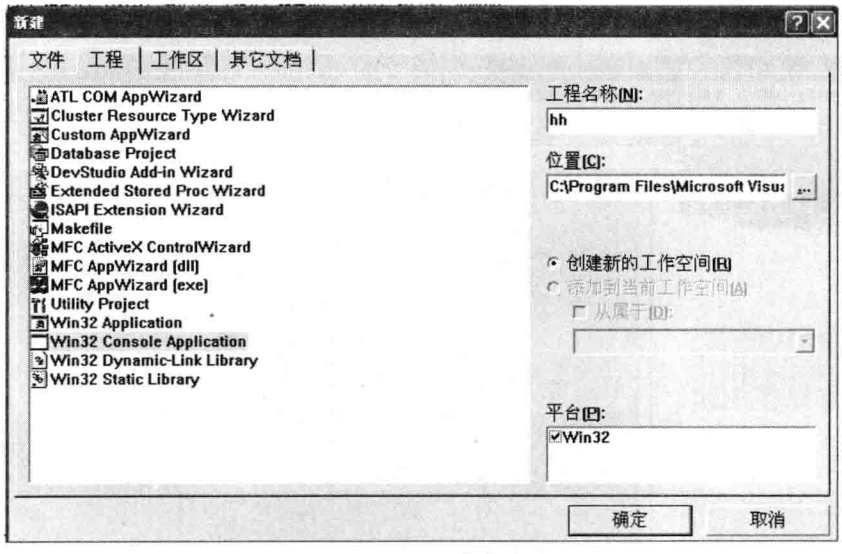

图 1.2.2　新建工程

1.2.5 所示。其中文件扩展名为 h，代表头文件，一般书写函数说明。文件扩展名为 cpp，表示 C ++ 中的源文件，每次建立一个工程都要有至少一个源文件，该文件中包含核心代码。

（3）根据原理编写程序，并编译运行。在图 1.2.6 中根据提示输入需要加密的明文，加密密钥，按回车键，就可看到密文输入，如图 1.2.6 所示。

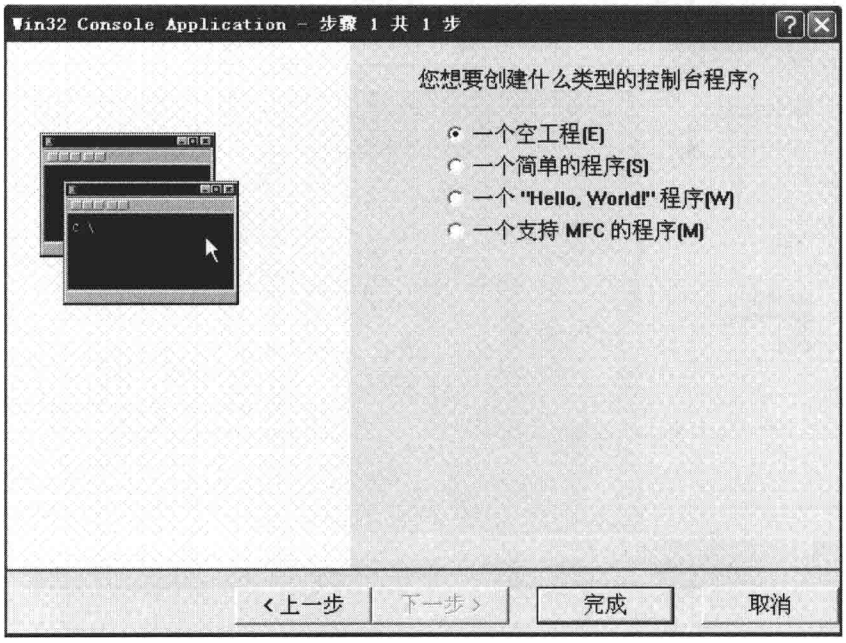

图 1.2.3　选择程序类型

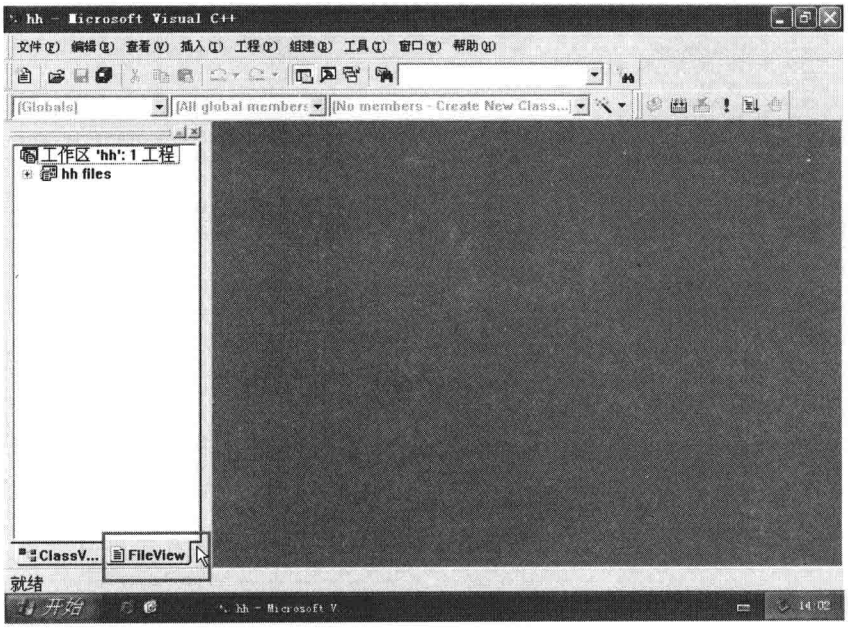

图 1.2.4　编辑界面

此为试读,需要完整PDF请访问: www.ertongbook.com