

<<软件加密技术内幕>>

图书基本信息

书名：<<软件加密技术内幕>>

13位ISBN编号：9787121000980

10位ISBN编号：7121000989

出版时间：2004-8-1

出版时间：电子工业出版社

作者：看雪学院

页数：404

字数：448000

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

<<软件加密技术内幕>>

内容概要

本书结合实例，重点讲述了软件加密技术及其实施方案，以帮助程序员更好地保护自己的软件。书中介绍了相关系统底层知识，例如，PE格式深入分析、调试API应用、未公开技术SEH的深入研究等，从而使读者在了解这些底层知识后，可以应用到自己的软件保护方案如各种反跟踪技术的实现中。

本书还首度公开了如何编写加壳软件，以及如何将壳与程序融合在一起等一些热门技术。

本书由密界一流高手共同打造，读者将从本书中获得许多极具商业价值的内幕技术，是专业开发人员不可多得的一本好书。

<<软件加密技术内幕>>

作者简介

段钢，网名看雪，1994年毕业于上海同济大学，看雪学院（<http://www.pediy.com>）站长，致力于软件加密与解密研究。

2001年9月组织推出国内第一本全面介绍Windows平台下软件的加密与解密技术的书籍《加密与解密——软件保护技术及完全解决方案》。

2003年6月出版《加密与解密

<<软件加密技术内幕>>

书籍目录

第1章 PE文件格式的深入分析 1.1 PE文件格式纵览 1.1.1 区块 1.1.2 相对虚拟地址 1.1.3 数据目录表 1.1.4 输入函数 1.2 PE文件结构 1.2.1 MS-DOS头部 1.2.2 IMAGE_NT_HEADERS头部 1.2.3 区块表 1.2.4 各种区块的描述 1.2.5 输出表 1.2.6 输出转向 1.2.7 输入表 1.2.8 绑定输入 1.2.9 延迟装入数据 1.2.10 资源 1.2.11 基址重定位 1.2.12 调试目录 1.2.13 .NET头部 1.2.14 TLS初始化 1.2.15 程序异常数据第2章 PE分析工具编写 2.1 文件格式检查 2.2 FileHeader和OptionalHeader内容的读取 2.3 得到数据目录表信息 2.4 得到区块表信息 2.5 得到输出表信息 2.6 得到输入表信息第3章 Win32 调试API 3.1 Win32调试API原理 3.1.1 调试相关函数简要说明 3.1.2 调试事件 3.1.3 如何在调试时创建并跟踪一个进程 3.1.4 调试循环体 3.1.5 如何处理调试事件 3.1.6 线程环境详解 3.1.7 如何在另一个进程中注入代码 3.2 利用调试API编写脱壳机 3.2.1 tElock 0.98脱壳简介 3.2.2 脱壳机的编写 3.3 利用调试API制作内存补丁 3.3.1 跨进程内存存取机制 3.3.2 Debug API机制 第4章 Windows下的异常处理 4.1 基本概念 4.1.1 Windows下的软件异常 4.1.2 异常处理的基本过程 4.1.3 SEH的分类 4.1.4 未公开的可靠吗 4.2 SEH相关数据结构 4.2.1 TIB结构 4.2.2 EXCEPTION_REGISTRATION结构 4.2.3 EXCEPTION_POINTERS, EXCEPTION_RECORD, CONTEXT结构 4.3 异常处理程序原理及设计 4.3.1 相关API 4.3.2 顶层异常处理 4.3.3 线程异常处理 4.3.4 异常处理的堆栈展开 4.3.5 异常处理程序设计中的注意事项 4.4 SEH的简单应用 4.4.1 Windows 9x下利用SEH进入Ring0 4.4.2 利用SEH实现对自身的单步自跟踪 4.4.3 其他应用 4.5 系统背后的秘密 4.6 VC如何封装系统提供的SEH机制 4.6.1 扩展的EXCEPTION_REGISTRATION级相关结构 4.6.2 数据结构组织 4.7 Windows XP下的向量化异常处理第5章 反跟踪技术 5.1 反调试技术 5.1.1 句柄检测 5.1.2 SoftICE后门指令 5.1.3 int68子类型 5.1.4 ICECream子类型 5.1.5 判断NTICE服务是否运行 5.1.6 INT 1 检测 5.1.7 利用UnhandledExceptionFilter检测 5.1.8 INT 41子类型 5.2 断点检测技术 5.2.1 检测函数首地址 5.2.2 利用SEH防范BPX断点 5.2.3 利用SEH防范BPM断点 5.3 反加载技术 (Anti-Loader) 5.3.1 利用TEB检测 5.3.2 利用IsDebuggerPresent函数检测 5.3.3 检查父进程 5.4 反监视技术 (Anti-Monitor) 5.4.1 窗口方法检测 5.4.2 句柄检测 5.5 反静态分析技术 5.5.1 扰乱汇编代码 5.5.2 花指令 5.5.3 SMC技术实现 5.5.4 信息隐藏 5.6 反DUMP技术(Anti-Dump) 5.7 文件完整性检验 5.7.1 磁盘文件校验实现 5.7.2 校验和 5.7.3 内存映像校验 5.8 代码与数据结合技术 5.8.1 准备工作 5.8.2 加密算法选用 5.8.3 手动加密代码 5.8.4 使.text区块可写 5.8.5 重定位 5.9 软件保护的若干忠告第6章 加壳软件编写 6.1 外壳编写基础 6.1.1 判断文件是否是PE-EXE文件 6.1.2 文件基本数据的读入 6.1.3 额外数据保留 6.1.4 重定位数据的去除 6.1.5 文件的压缩 6.1.6 资源区块的处理 6.1.7 区块的融合 6.1.8 输入表的处理 6.1.9 外壳部分的编写 6.1.10 将外壳部分添加至原始程序 6.1.11 小结 6.2 加壳程序综合运用的实例 6.2.1 程序简介 6.2.2 加壳子程序 (WJQ_ShellBegin()) 6.2.3 PE外壳程序 6.2.4 加进Anti技术 6.2.5 通过外壳修改被加壳PE 6.2.6 VC++调用汇编子程序 第7章 如何让壳与程序融为一体 7.1 欺骗查壳工具 7.1.1 FileInfo是如何查壳的 7.1.2 欺骗FileInfo 7.2 判断自己是否被加壳 7.2.1 判断文件尺寸 7.2.2 使用同步对象检查标记 7.2.3 使用原子 (Atom) 检查标记 7.2.4 使用存储映像文件检查标记 7.2.5 使用线程优先级检查标记 7.2.6 使用外部文件检查标记 7.2.7 使用注册表检查标记 7.2.8 注入一个定时器 7.2.9 外部检测 (使用DLL) 7.2.10 Hook 相关的API (防止Loader和调试API) 7.3 使用SDK把程序和壳融为一体 7.3.1 SDK加密的标记 7.3.2 壳程序检测加密标志 7.3.3 开始加密相关的数据 7.3.4 输出函数的声明 7.3.5 输出函数的执行代码定位 7.3.6 为输出函数得到壳中加密函数做准备 7.3.7 程序中使用加密和解密函数 7.3.8 构造壳中的加密和解密函数 7.3.9 壳寻找程序的输出函数位置 7.3.10 “毁尸灭迹”, 擦除输出函数 7.3.11 壳中分配临时的内存存放加密和解密函数 7.3.12 壳中执行程序输出函数传递参数 第8章 Visual Basic 6 逆向工程 8.1 P-code传奇 8.2 VB编译奥秘 8.3 VB与COM 8.4 VB可执行程序结构研究 8.5 VB程序事件解读 8.6 VB程序图形界面解读 8.7 VB执行代码研究 8.7.1 VB函数的解读 8.7.2 VB函数调用约定 8.7.3 执行代码中对控件属性的操作 8.8 P-code代码 8.8.1 理解P-code代码指令 8.8.2 P-code程序调用约定 8.8.3 调试时中断P-code程序的几种方法 8.8.4 WKT VB Debugger实现原理 8.8.5 VB6 P-code Crackme分析实例 8.9 VB程序保护篇 8.9.1 Anti-Loader技术 8.9.2 VB “自

锁”功能实现 8.10 相关工具点评附录A在Visual C++中使用内联汇编附录B 在Visual Basic中使用汇编

<<软件加密技术内幕>>

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>