

图书基本信息

书名：<<深入解析Windows操作系统（卷2）>>

13位ISBN编号：9787115304506

10位ISBN编号：7115304505

出版时间：2013-2

出版时间：人民邮电出版社

作者：[美] Mark Russinovich,[加] Alex Ionescu,[美] David Solomon

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>

作者简介

作者：（美国）拉希诺维奇（Mark Russinovich）（美国）所罗门（David Solomon）（加拿大）艾欧内斯库（Alex Ionescu）

书籍目录

Windows Internals, Sixth Edition, Part 1 (See appendix for Part 1's table of contents) Chapter 8 I/O System I/O System Components The I/O Manager Typical I/O Processing Device Drivers Types of Device Drivers Structure of a Driver Driver Objects and Device Objects Opening Devices I/O Processing Types of I/O I/O Request to a Single-Layered Driver I/O Requests to Layered Drivers I/O Cancellation I/O Completion Ports I/O Prioritization Container Notifications Driver Verifier Kernel-Mode Driver Framework (KMDF) Structure and Operation of a KMDF Driver KMDF Data Model KMDF I/O Model User-Mode Driver Framework (UMDF) The Plug and Play (PnP) Manager Level of Plug and Play Support Driver Support for Plug and Play Driver Loading, Initialization, and Installation Driver Installation The Power Manager Power Manager Operation Driver Power Operation Driver and Application Control of Device Power Power Availability Requests Processor Power Management (PPM) Conclusion Chapter 9 Storage Management Storage Terminology Disk Devices Rotating Magnetic Disks Solid State Disks Disk Drivers Winload Disk Class, Port, and Miniport Drivers Disk Device Objects Partition Manager Volume Management Basic Disks Dynamic Disks Multipartition Volume Management The Volume Namespace Volume I/O Operations Virtual Disk Service Virtual Hard Disk Support Attaching VHDs Nested File Systems BitLocker Drive Encryption Encryption Keys Trusted Platform Module (TPM) BitLocker Boot Process BitLocker Key Recovery Full-Volume Encryption Driver BitLocker Management BitLocker To Go Volume Shadow Copy Service Shadow Copies VSS Architecture VSS Operation Uses in Windows Conclusion Chapter 10 Memory Management Introduction to the Memory Manager Memory Manager Components Internal Synchronization Examining Memory Usage Services Provided by the Memory Manager Large and Small Pages Reserving and Committing Pages Commit Limit Locking Memory Allocation Granularity Shared Memory and Mapped Files Protecting Memory No Execute Page Protection Copy-on-Write Address Windowing Extensions Kernel-Mode Heaps (System Memory Pools) Pool Sizes Monitoring Pool Usage Look-Aside Lists Heap Manager Types of Heaps Heap Manager Structure Heap Synchronization The Low Fragmentation Heap Heap Security Features Heap Debugging Features Pageheap Fault Tolerant Heap Virtual Address Space Layouts x86 Address Space Layouts x86 System Address Space Layout x86 Session Space System Page Table Entries 64-Bit Address Space Layouts x64 Virtual Addressing Limitations Dynamic System Virtual Address Space Management System Virtual Address Space Quotas User Address Space Layout Address Translation x86 Virtual Address Translation Translation Look-Aside Buffer Physical Address Extension (PAE) x64 Virtual Address Translation IA64 Virtual Address Translation Page Fault Handling Invalid PTEs Prototype PTEs In-Paging I/O Collided Page Faults Clustered Page Faults Page Files Commit Charge and the System Commit Limit Commit Charge and Page File Size Stacks User Stacks Kernel Stacks DPC Stack Virtual Address Descriptors Process VADs Rotate VADs NUMA Section Objects Driver Verifier Page Frame Number Database Page List Dynamics Page Priority Modified Page Writer PFN Data Structures Physical Memory Limits Windows Client Memory Limits Working Sets Demand Paging Logical Prefetcher Placement Policy Working Set Management Balance Set Manager and Swapper System Working Sets Memory Notification Events Proactive Memory Management (Superfetch) Components Tracing and Logging Scenarios Page Priority and Rebalancing Robust Performance ReadyBoost ReadyDrive Unified Caching Process Reflection Conclusion Chapter 11 Cache Manager Chapter 12 File Systems Chapter 13 Startup and Shutdown Chapter 14 Crash Dump Analysis Appendix: Contents of Windows Internals, Sixth Edition, Part 1 Index

章节摘录

版权页：插图： Early in the boot process, the memory manager reads the Driver Verifier registry values to determine which drivers to verify and which Driver Verifier options you enabled. (Note that if you boot in safe mode, any Driver Verifier settings are ignored.) Subsequently, if you've selected at least one driver for verification, the kernel checks the name of every device driver it loads into memory against the list of drivers you've selected for verification. For every device driver that appears in both places, the kernel invokes the VfLoadDriver function, which calls other internal Vf* functions to replace the driver's references to a number of kernel functions with references to Driver Verifier-equivalent versions of those functions. For example, ExAllocatePool is replaced with a call to VerifierAllocatePool. The windowing system driver (Win32k.sys) also makes similar changes to use Driver Verifier-equivalent functions. Now that we've reviewed how Driver Verifier is set up, we'll examine the six memory-related verification options that can be applied to device drivers: Special Pool, Pool Tracking, Force IRQL Checking, Low Resources Simulation, Miscellaneous Checks, and Automatic Checks Special Pool The Special Pool option causes the pool allocation routines to bracket pool allocations with an invalid page so that references before or after the allocation will result in a kernel-mode access violation, thus crashing the system with the finger pointed at the buggy driver. Special pool also causes some additional validation checks to be performed when a driver allocates or frees memory. When special pool is enabled, the pool allocation routines allocate a region of kernel memory for Driver Verifier to use. Driver Verifier redirects memory allocation requests that drivers under verification make to the special pool area rather than to the standard kernel-mode memory pools. When a device driver allocates memory from special pool, Driver Verifier rounds up the allocation to an evenpage boundary. Because Driver Verifier brackets the allocated page with invalid pages, if a device driver attempts to read or write past the end of the buffer, the driver will access an invalid page, and the memory manager will raise a kernel-mode access violation. Figure 10-36 shows an example of the special pool buffer that Driver Verifier allocates to a device driver when Driver Verifier checks for overrun errors.

媒体关注与评论

"在微软，我们一直用本书培训新员工.....本书是深入理解Windows的绝佳入门书。

" --Windows之父Jim Allchin"每一位操作系统开发人员都应该拥有本书。

" --微软技术院士、Windows NT首席设计师David Cutler"我想不出还有哪一本书比本书更具权威性。

" --微软公司副总裁Ben Fathi

编辑推荐

微软官方Windows权威著作最新版，深入剖析Windows技术内幕，大幅更新，涵盖Windows内核最新特性，带你身入Windows技术核心，了解Windows内部是如何工作的。

全书内容丰富、信息全面，主要包括的Windows操作系统深度知识有：理解Windows的关键机制，包括系统服务分发和调度机制、启动和停机，以及注册表；挖掘Windows的安全模型，包括访问控制、特权和审计；利用内核调试器和其他的工具来检查内部系统结构；检查与进程、线程和作业相关的数据结构和算法；观察Windows如何管理虚拟内存和物理内存；理解NTFS的操作和格式，诊断文件系统访问问题；从上往下查看Windows的网络栈，包括映射、API、名称解析和协议驱动程序；诊断引导问题，执行崩溃分析。

名人推荐

“在微软，我们一直用本书培训新员工……本书是深入理解Windows的绝佳入门书”——Windows之父jim Allchin “每一位操作系统开发人员都应该拥有本书”——微软技术院士、Windows NT首席设计师David Cutler “我想不还有哪一本书比本书更具权威性。”——微软公司副总裁Ben Fathi

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>