

Windows Internals

Part 2 **Sixth Edition**

深入解析 Windows操作系统 卷2 (英文版·第6版)

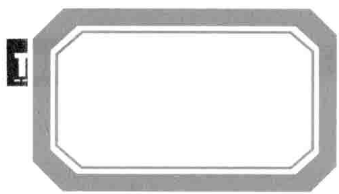
[美] Mark Russinovich
David Solomon 著
[加] Alex Ionescu



- 微软官方权威著作最新版
- 深入剖析Windows技术内幕
- 大幅更新，涵盖Windows内核最新特性



人民邮电出版社
POSTS & TELECOM PRESS



计丛书

Windows Internals

Part 2 **Sixth Edition**

深入解析 Windows操作系统 卷2 (英文版·第6版)

[美] Mark Russinovich
David Solomon 著
[加] Alex Ionescu

人民邮电出版社
北 京

图书在版编目 (C I P) 数据

深入解析Windows操作系统：第6版. 卷2 = Windows Internals:Sixth Edition.Part 2：英文 / (美) 拉希诺维奇 (Russinovich, M.) , (美) 所罗门 (Solomon, D.) , (加) 艾欧内斯库 (Ionescu, A.) 著. — 北京：人民邮电出版社, 2013. 2
(图灵程序设计丛书)
ISBN 978-7-115-30450-6

I. ①深… II. ①拉… ②所… ③艾… III. ①Windows操作系统—英文 IV. ①TP316.7

中国版本图书馆CIP数据核字(2012)第308318号

内 容 提 要

本书是操作系统内核专家 Russinovich 等人的 Windows 操作系统原理的最新版著作, 针对 Windows 7 和 Windows Server 2008 R2 进行了全面的更新, 主要讲述 Windows 的底层关键机制、Windows 的核心组件(包括进程/线程/作业, 安全性, I/O 系统, 存储管理、内存管理、缓存管理、文件系统和网络), 并分析了启动进程、关机进程以及缓存存储。书中提供了许多实例, 读者可以借此更好地理解 Windows 的内部行为。

本书内容丰富, 信息全面, 适合众多 Windows 平台开发人员、系统管理员阅读。

图灵程序设计丛书

深入解析Windows操作系统, 卷2 (英文版·第6版)

◆ 著 [美] Mark Russinovich [美] David Solomon
[加] Alex Ionescu

责任编辑 朱 巍

◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街14号
邮编 100061 电子邮件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>

北京艺辉印刷有限公司印刷

◆ 开本: 800×1000 1/16
印张: 41.25
字数: 748千字 2013年2月第1版
印数: 1-3 000册 2013年2月北京第1次印刷

著作权合同登记号 图字: 01-2012-8544号

ISBN 978-7-115-30450-6

定价: 99.00元

读者服务热线: (010)51095186转604 印装质量热线: (010)67129223

反盗版热线: (010)67171154

Introduction

Windows *Internals, Sixth Edition* is intended for advanced computer professionals (both developers and system administrators) who want to understand how the core components of the Microsoft Windows 7 and Windows Server 2008 R2 operating systems work internally. With this knowledge, developers can better comprehend the rationale behind design choices when building applications specific to the Windows platform. Such knowledge can also help developers debug complex problems. System administrators can benefit from this information as well, because understanding how the operating system works “under the covers” facilitates understanding the performance behavior of the system and makes troubleshooting system problems much easier when things go wrong. After reading this book, you should have a better understanding of how Windows works and why it behaves as it does.

Structure of the Book

For the first time, the book has been divided in two parts. This was done to get the information out more quickly since it takes considerable time to update the book for each release of Windows.

Part 1 begins with two chapters that define key concepts, introduce the tools used in the book, and describe the overall system architecture and components. The next two chapters present key underlying system and management mechanisms. Part 1 wraps up by covering three core components of the operating system: processes, threads, and jobs; security; and networking.

Part 2 covers the remaining core subsystems: I/O, storage, memory management, the cache manager, and file systems. Part 2 concludes with a description of the startup and shutdown processes and a description of crash-dump analysis.

History of the Book

This is the sixth edition of a book that was originally called *Inside Windows NT* (Microsoft Press, 1992), written by Helen Custer (prior to the initial release of Microsoft Windows NT 3.1). *Inside Windows NT* was the first book ever published about Windows NT and provided key insights into the architecture and design of the system. *Inside Windows NT, Second Edition* (Microsoft Press, 1998) was written by David Solomon. It updated the original book to cover Windows NT 4.0 and had a greatly increased level of technical depth.

Inside Windows 2000, Third Edition (Microsoft Press, 2000) was authored by David Solomon and Mark Russinovich. It added many new topics, such as startup and shutdown, service internals, registry internals, file-system drivers, and networking. It also covered kernel changes in Windows 2000, such as the Windows Driver Model (WDM), Plug and Play, power management, Windows Management Instrumentation (WMI), encryption, the job object, and Terminal Services. *Windows Internals, Fourth Edition* was the Windows XP and Windows Server 2003 update and added more content focused on helping IT professionals make use of their knowledge of Windows internals, such as using key tools from Windows Sysinternals (www.microsoft.com/technet/sysinternals) and analyzing crash dumps. *Windows Internals, Fifth Edition* was the update for Windows Vista and Windows Server 2008. New content included the image loader, user-mode debugging facility, and Hyper-V.

Sixth Edition Changes

This latest edition has been updated to cover the kernel changes made in Windows 7 and Windows Server 2008 R2. Hands-on experiments have been updated to reflect changes in tools.

Hands-on Experiments

Even without access to the Windows source code, you can glean much about Windows internals from tools such as the kernel debugger and tools from Sysinternals and Winsider Seminars & Solutions. When a tool can be used to expose or demonstrate some aspect of the internal behavior of Windows, the steps for trying the tool yourself are listed in “EXPERIMENT” boxes. These appear throughout the book, and we encourage you to try these as you’re reading—seeing visible proof of how Windows works internally will make much more of an impression on you than just reading about it will.

Topics Not Covered

Windows is a large and complex operating system. This book doesn't cover everything relevant to Windows internals but instead focuses on the base system components. For example, this book doesn't describe COM+, the Windows distributed object-oriented programming infrastructure, or the Microsoft .NET Framework, the foundation of managed code applications.

Because this is an internals book and not a user, programming, or system administration book, it doesn't describe how to use, program, or configure Windows.

A Warning and a Caveat

Because this book describes undocumented behavior of the internal architecture and the operation of the Windows operating system (such as internal kernel structures and functions), this content is subject to change between releases. (External interfaces, such as the Windows API, are not subject to incompatible changes.)

By "subject to change," we don't necessarily mean that details described in this book will change between releases, but you can't count on them not changing. Any software that uses these undocumented interfaces might not work on future releases of Windows. Even worse, software that runs in kernel mode (such as device drivers) and uses these undocumented interfaces might experience a system crash when running on a newer release of Windows.

Acknowledgments

First, thanks to Jamie Hanrahan and Brian Catlin of Azius, LLC for joining us on this project—the book would not have been finished without their help. They did the bulk of the updates on the "Security" and "Networking" chapters and contributed to the update of the "Management Mechanisms" and "Processes and Threads" chapters. Azius provides Windows-internals and device-driver training. See www.azius.com for more information.

We want to recognize Alex Ionescu, who for this edition is a full coauthor. This is a reflection of Alex's extensive work on the fifth edition, as well as his continuing work on this edition.

Also thanks to Daniel Pearson, who updated the "Crash Dump Analysis" chapter. His many years of dump analysis experience helped to make the information more practical.

Thanks to Eric Traut and Jon DeVaan for continuing to allow David Solomon access to the Windows source code for his work on this book as well as continued development of his Windows Internals courses.

Three key reviewers were not acknowledged for their review and contributions to the fifth edition: Arun Kishan, Landy Wang, and Aaron Margosis—thanks again to them! And thanks again to Arun and Landy for their detailed review and helpful input for this edition.

This book wouldn't contain the depth of technical detail or the level of accuracy it has without the review, input, and support of key members of the Microsoft Windows development team. Therefore, we want to thank the following people, who provided technical review and input to the book:

- Greg Cottingham
- Joe Hamburg
- Jeff Lambert
- Pavel Lebedinsky
- Joseph East
- Adi Oltean
- Alexey Pakhunov
- Valerie See
- Brad Waters
- Bruce Worthington
- Robin Alexander
- Bernard Ourghanlian

Also thanks to Scott Lee, Tim Shoultz, and Eric Kratzer for their assistance with the "Crash Dump Analysis" chapter.

For the "Networking" chapter, a special thanks to Gianluigi Nуска and Tom Jolly, who really went beyond the call of duty: Gianluigi for his extraordinary help with the BranchCache material and the amount of suggestions (and many paragraphs of

material he wrote), and Tom Jolly not only for his own review and suggestions (which were excellent), but for getting many other developers to assist with the review. Here are all those who reviewed and contributed to the “Networking” chapter:

- Roopesh Battepati
- Molly Brown
- Greg Cottingham
- Dotan Elharrar
- Eric Hanson
- Tom Jolly
- Manoj Kadam
- Greg Kramer
- David Kruse
- Jeff Lambert
- Darene Lewis
- Dan Lovinger
- Gianluigi Nusca
- Amos Ortal
- Ivan Pashov
- Ganesh Prasad
- Paul Swan
- Shiva Kumar Thangapandi

Amos Ortal and Dotan Elharrar were extremely helpful on NAP, and Shiva Kumar Thangapandi helped extensively with EAP.

Thanks to Gerard Murphy for reviewing the shutdown mechanisms in Windows 7 and clarifying Group Policy behaviors.

Thanks to Tristan Brown from the Power Management team at Microsoft for spending a few late hours at the office with Alex going over core parking’s algorithms and behaviors, as well as for the invaluable diagram he provided.

Thanks to Apurva Doshi for sending Alex a detailed document of cache manager changes in Windows 7, which was used to capture some of the new behaviors and changes described in the book.

Thanks to Matthieu Suiche for his kernel symbol file database, which allowed Alex to discover new and removed fields from core kernel data structures and led to the investigations to discover the underlying functionality changes.

Thanks to Cenk Ergun, Michel Fortin, and Mehmet Iyigun for their review and input on the Superfetch details.

The detailed checking Christophe Nasarre, overall technical reviewer, performed contributed greatly to the technical accuracy and consistency in the book.

We would like to again thank Ilfak Guilfanov of Hex-Rays (www.hex-rays.com) for the IDA Pro Advanced and Hex-Rays licenses they granted to Alex so that he could speed up his reverse engineering of the Windows kernel.

Finally, the authors would like to thank the great staff at Microsoft Press behind turning this book into a reality. Devon Musgrave served double duty as acquisitions editor and developmental editor, while Carol Dillingham oversaw the title as its project editor. Editorial and production manager Curtis Philips, copy editor John Pierce, proof-reader Andrea Fox, and indexer Jan Wright also contributed to the quality of this book.

Last but not least, thanks to Ben Ryan, publisher of Microsoft Press, who continues to believe in the importance of continuing to provide this level of detail about Windows to their readers!

Errata & Book Support

We've made every effort to ensure the accuracy of this book and its companion content. Any errors that have been reported since this book was published are listed on our Microsoft Press site at oreilly.com:

<http://go.microsoft.com/fwlink/?Linkid=258649>

If you find an error that is not already listed, you can report it to us through the same page.

If you need additional support, email Microsoft Press Book Support at mspinput@microsoft.com.

Please note that product support for Microsoft software is not offered through the addresses above.

We Want to Hear from You

At Microsoft Press, your satisfaction is our top priority, and your feedback our most valuable asset. Please tell us what you think of this book at:

<http://www.microsoft.com/learning/booksurvey>

The survey is short, and we read every one of your comments and ideas. Thanks in advance for your input!

Stay in Touch

Let's keep the conversation going! We're on Twitter: <http://twitter.com/MicrosoftPress>.

Contents

Windows Internals, Sixth Edition, Part 1

(See appendix for Part 1's table of contents)

Chapter 8 I/O System	1
I/O System Components	1
The I/O Manager	3
Typical I/O Processing	4
Device Drivers	5
Types of Device Drivers	5
Structure of a Driver	12
Driver Objects and Device Objects	14
Opening Devices	19
I/O Processing	25
Types of I/O	25
I/O Request to a Single-Layered Driver	33
I/O Requests to Layered Drivers	40
I/O Cancellation	48
I/O Completion Ports	53
I/O Prioritization	58
Container Notifications	65
Driver Verifier	65
Kernel-Mode Driver Framework (KMDF)	68
Structure and Operation of a KMDF Driver	68
KMDF Data Model	70
KMDF I/O Model	74

User-Mode Driver Framework (UMDF).....	78
The Plug and Play (PnP) Manager	81
Level of Plug and Play Support	82
Driver Support for Plug and Play	82
Driver Loading, Initialization, and Installation	84
Driver Installation.....	94
The Power Manager.....	98
Power Manager Operation.....	100
Driver Power Operation	101
Driver and Application Control of Device Power.....	105
Power Availability Requests	105
Processor Power Management (PPM)	108
Conclusion	123
Chapter 9 Storage Management	125
Storage Terminology	125
Disk Devices.....	126
Rotating Magnetic Disks.....	126
Solid State Disks.....	128
Disk Drivers	131
Winload	132
Disk Class, Port, and Miniport Drivers	132
Disk Device Objects.....	136
Partition Manager	138
Volume Management.....	138
Basic Disks.....	139
Dynamic Disks.....	141
Multipartition Volume Management.....	147
The Volume Namespace.....	153
Volume I/O Operations.....	159
Virtual Disk Service	160
Virtual Hard Disk Support.....	162
Attaching VHDs	163
Nested File Systems	163
BitLocker Drive Encryption	163
Encryption Keys	165
Trusted Platform Module (TPM)	168
BitLocker Boot Process	170
BitLocker Key Recovery.....	172

Full-Volume Encryption Driver	173
BitLocker Management	174
BitLocker To Go	175
Volume Shadow Copy Service	177
Shadow Copies	177
VSS Architecture	177
VSS Operation	178
Uses in Windows	181
Conclusion	186

Chapter 10 Memory Management 187

Introduction to the Memory Manager	187
Memory Manager Components	188
Internal Synchronization	189
Examining Memory Usage	190
Services Provided by the Memory Manager	193
Large and Small Pages	193
Reserving and Committing Pages	195
Commit Limit	199
Locking Memory	199
Allocation Granularity	199
Shared Memory and Mapped Files	200
Protecting Memory	203
No Execute Page Protection	204
Copy-on-Write	209
Address Windowing Extensions	210
Kernel-Mode Heaps (System Memory Pools)	212
Pool Sizes	213
Monitoring Pool Usage	215
Look-Aside Lists	219
Heap Manager	220
Types of Heaps	221
Heap Manager Structure	222
Heap Synchronization	223
The Low Fragmentation Heap	223
Heap Security Features	224
Heap Debugging Features	225
Pageheap	226
Fault Tolerant Heap	227

Virtual Address Space Layouts	228
x86 Address Space Layouts	229
x86 System Address Space Layout	232
x86 Session Space	233
System Page Table Entries	235
64-Bit Address Space Layouts	237
x64 Virtual Addressing Limitations	240
Dynamic System Virtual Address Space Management	242
System Virtual Address Space Quotas	245
User Address Space Layout	246
Address Translation	251
x86 Virtual Address Translation	252
Translation Look-Aside Buffer	259
Physical Address Extension (PAE)	260
x64 Virtual Address Translation	265
IA64 Virtual Address Translation	266
Page Fault Handling	267
Invalid PTEs	268
Prototype PTEs	269
In-Paging I/O	271
Collided Page Faults	272
Clustered Page Faults	272
Page Files	273
Commit Charge and the System Commit Limit	275
Commit Charge and Page File Size	278
Stacks	279
User Stacks	280
Kernel Stacks	281
DPC Stack	282
Virtual Address Descriptors	282
Process VADs	283
Rotate VADs	284
NUMA	285
Section Objects	286
Driver Verifier	292
Page Frame Number Database	297
Page List Dynamics	300
Page Priority	310
Modified Page Writer	314

PFN Data Structures	315
Physical Memory Limits	320
Windows Client Memory Limits	321
Working Sets	324
Demand Paging	324
Logical Prefetcher	324
Placement Policy	328
Working Set Management	329
Balance Set Manager and Swapper	333
System Working Sets	334
Memory Notification Events	335
Proactive Memory Management (Superfetch)	338
Components	338
Tracing and Logging	341
Scenarios	342
Page Priority and Rebalancing	342
Robust Performance	344
ReadyBoost	346
ReadyDrive	348
Unified Caching	348
Process Reflection	351
Conclusion	354
Chapter 11 Cache Manager	355
Key Features of the Cache Manager	355
Single, Centralized System Cache	356
The Memory Manager	356
Cache Coherency	356
Virtual Block Caching	358
Stream-Based Caching	358
Recoverable File System Support	359
Cache Virtual Memory Management	360
Cache Size	361
Cache Virtual Size	361
Cache Working Set Size	361
Cache Physical Size	363
Cache Data Structures	364
Systemwide Cache Data Structures	365
Per-File Cache Data Structures	368

File System Interfaces	373
Copying to and from the Cache	374
Caching with the Mapping and Pinning Interfaces	374
Caching with the Direct Memory Access Interfaces	375
Fast I/O	375
Read-Ahead and Write-Behind	377
Intelligent Read-Ahead	378
Write-Back Caching and Lazy Writing	379
Write Throttling	388
System Threads	390
Conclusion	390

Chapter 12 File Systems 391

Windows File System Formats	392
CDFS	392
UDF	393
FAT12, FAT16, and FAT32	393
exFAT	396
NTFS	397
File System Driver Architecture	398
Local FSDs	398
Remote FSDs	400
File System Operation	407
File System Filter Drivers	413
Troubleshooting File System Problems	415
Process Monitor Basic vs. Advanced Modes	415
Process Monitor Troubleshooting Techniques	416
Common Log File System	416
NTFS Design Goals and Features	424
High-End File System Requirements	424
Advanced Features of NTFS	426
NTFS File System Driver	439
NTFS On-Disk Structure	442
Volumes	442
Clusters	442
Master File Table	443
File Record Numbers	447
File Records	447
File Names	449

Resident and Nonresident Attributes	453
Data Compression and Sparse Files	456
The Change Journal File	461
Indexing	464
Object IDs	466
Quota Tracking	466
Consolidated Security	467
Repase Points	469
Transaction Support	469
NTFS Recovery Support	477
Design	478
Metadata Logging	479
Recovery	483
NTFS Bad-Cluster Recovery	487
Self-Healing	490
Encrypting File System Security	491
Encrypting a File for the First Time	494
The Decryption Process	496
Backing Up Encrypted Files	497
Copying Encrypted Files	497
Conclusion	498

Chapter 13 Startup and Shutdown 499

Boot Process	499
BIOS Preboot	499
The BIOS Boot Sector and Bootmgr	502
The UEFI Boot Process	512
Booting from iSCSI	514
Initializing the Kernel and Executive Subsystems	514
Smss, Csrss, and Wininit	522
ReadyBoot	527
Images That Start Automatically	528
Troubleshooting Boot and Startup Problems	529
Last Known Good	530
Safe Mode	530
Windows Recovery Environment (WinRE)	534
Solving Common Boot Problems	537
Shutdown	542
Conclusion	545