

# 《深入解析Windows操作系统》

## 图书基本信息

书名：《深入解析Windows操作系统》

13位ISBN编号：9787115290908

10位ISBN编号：7115290903

出版时间：2012-9

出版社：人民邮电出版社

页数：726

版权说明：本站所提供下载的PDF图书仅提供预览和简介以及在线试读，请支持正版图书。

更多资源请访问：[www.tushu000.com](http://www.tushu000.com)

# 《深入解析Windows操作系统》

## 内容概要

# 《深入解析Windows操作系统》

## 作者简介

作者:(美)Mark Russinovich , (美)David Solomon , (加)Alex Ionescu

## 书籍目录

Chapter 1 Concepts and Tools	1
Windows Operating System Versions	1
Foundation Concepts and Terms	2
Windows API	2
Services, Functions, and Routines	4
Processes, Threads, and Jobs	5
Virtual Memory	15
Kernel Mode vs. User Mode	17
Terminal Services and Multiple Sessions	20
Objects and Handles	21
Security	22
Registry	23
Unicode	24
Digging into Windows Internals	24
Performance Monitor	25
Kernel Debugging	26
Windows Software Development Kit	31
Windows Driver Kit	31
Sysinternals Tools	32
Conclusion	32
Chapter 2 System Architecture	33
Requirements and Design Goals	33
Operating System Model	34
Architecture Overview	35
Portability	37
Symmetric Multiprocessing	38
Scalability	40
Differences Between Client and Server Versions	41
Checked Build	45
Key System Components	46
Environment Subsystems and Subsystem DLLs	48
Ntdll.dll	53
Executive	54
Kernel	57
Hardware Abstraction Layer	60
Device Drivers	63
System Processes	68
Conclusion	78
Chapter 3 System Mechanisms	79
Trap Dispatching	79
Interrupt Dispatching	81
Timer Processing	112
Exception Dispatching	123
System Service Dispatching	132
Object Manager	140
Executive Objects	143
Object Structure	145

- Synchronization176
  - High-IRQL Synchronization178
  - Low-IRQL Synchronization183
- System Worker Threads205
- Windows Global Flags207
- Advanced Local Procedure Call209
- Connection Model210
- Message Model211
- Asynchronous Operation213
- Views, Regions, and Sections214
- Attributes215
- Blobs, Handles, and Resources215
- Security216
- Performance217
- Debugging and Tracing218
- Kernel Event Tracing220
- Wow64224
  - Wow64 Process Address Space Layout224
- System Calls225
  - Exception Dispatching225
  - User APC Dispatching225
  - Console Support225
  - User Callbacks226
  - File System Redirection226
  - Registry Redirection227
  - I/O Control Requests227
  - 16-Bit Installer Applications228
- Printing228
- Restrictions228
- User-Mode Debugging229
- Kernel Support229
- Native Support230
- Windows Subsystem Support232
- Image Loader232
- Early Process Initialization234
- DLL Name Resolution and Redirection235
- Loaded Module Database238
- Import Parsing242
- Post-Import Process Initialization243
- SwitchBack244
- API Sets245
- Hypervisor (Hyper-V)248
- Partitions249
  - Parent Partition249
  - Child Partitions251
- Hardware Emulation and Support254
- Kernel Transaction Manager268
- Hotpatch Support270
- Kernel Patch Protection272

Code Integrity	274
Conclusion	276
Chapter 4 Management Mechanisms	277
The Registry	277
Viewing and Changing the Registry	277
Registry Usage	278
Registry Data Types	279
Registry Logical Structure	280
Transactional Registry (TxR)	287
Monitoring Registry Activity	289
Process Monitor Internals	289
Registry Internals	293
Services	305
Service Applications	305
The Service Control Manager	321
Service Startup	323
Startup Errors	327
Accepting the Boot and Last Known Good	328
Service Failures	330
Service Shutdown	331
Shared Service Processes	332
Service Tags	335
Unified Background Process Manager	336
Initialization	337
UBPM API	338
Provider Registration	338
Consumer Registration	339
Task Host	341
Service Control Programs	341
Windows Management Instrumentation	342
Providers	344
The Common Information Model and the Managed Object Format Language	345
Class Association	349
WMI Implementation	351
WMI Security	353
Windows Diagnostic Infrastructure	354
WDI Instrumentation	354
Diagnostic Policy Service	354
Diagnostic Functionality	356
Conclusion	357
Chapter 5 Processes, Threads, and Jobs	359
Process Internals	359
Data Structures	359
Protected Processes	368
Flow of CreateProcess	369
Stage 1: Converting and Validating Parameters and Flags	371
Stage 2: Opening the Image to Be Executed	373
Stage 3: Creating the Windows Executive Process Object (PspAllocateProcess)	376
Stage 4: Creating the Initial Thread and Its Stack and Context	381

Stage 5: Performing Windows Subsystem – Specific Post-Initialization	383
Stage 6: Starting Execution of the Initial Thread	385
Stage 7: Performing Process Initialization in the Context of the New Process	386
Thread Internals	391
Data Structures	391
Birth of a Thread	398
Examining Thread Activity	398
Limitations on Protected Process Threads	401
Worker Factories (Thread Pools)	403
Thread Scheduling	408
Overview of Windows Scheduling	408
Priority Levels	410
Thread States	416
Dispatcher Database	421
Quantum	422
Priority Boosts	430
Context Switching	448
Scheduling Scenarios	449
Idle Threads	453
Thread Selection	456
Multiprocessor Systems	458
Thread Selection on Multiprocessor Systems	467
Processor Selection	468
Processor Share-Based Scheduling	470
Distributed Fair Share Scheduling	471
CPU Rate Limits	478
Dynamic Processor Addition and Replacement	479
Job Objects	480
Job Limits	481
Job Sets	482
Conclusion	485
Chapter 6 Security	487
Security Ratings	487
Trusted Computer System Evaluation Criteria	487
The Common Criteria	489
Security System Components	490
Protecting Objects	494
Access Checks	495
Security Identifiers	497
Virtual Service Accounts	518
Security Descriptors and Access Control	522
The AuthZ API	536
Account Rights and Privileges	538
Account Rights	540
Privileges	540
Super Privileges	546
Access Tokens of Processes and Threads	547
Security Auditing	548
Object Access Auditing	549

Global Audit Policy	552
Advanced Audit Policy Settings	554
Logon	555
Winlogon Initialization	556
User Logon Steps	558
Assured Authentication	562
Biometric Framework for User Authentication	563
User Account Control and Virtualization	566
File System and Registry Virtualization	566
Elevation	573
Application Identification (AppID)	581
AppLocker	583
Software Restriction Policies	589
Conclusion	590
Chapter 7 Networking	591
Windows Networking Architecture	591
The OSI Reference Model	592
Windows Networking Components	594
Networking APIs	597
Windows Sockets	597
Winsock Kernel	603
Remote Procedure Call	605
Web Access APIs	610
Named Pipes and Mailslots	612
NetBIOS	618
Other Networking APIs	620
Multiple Redirector Support	627
Multiple Provider Router	627
Multiple UNC Provider	630
Surrogate Providers	632
Redirector	633
Mini-Redirectors	634
Server Message Block and Sub-Redirectors	635
Distributed File System Namespace	637
Distributed File System Replication	638
Offline Files	639
Caching Modes	641
Ghosts	643
Data Security	643
Cache Structure	643
BranchCache	645
Caching Modes	647
BranchCache Optimized Application Retrieval:SMB Sequence	651
BranchCache Optimized Application Retrieval:HTTP Sequence	653
Name Resolution	655
Domain Name System	655
Peer Name Resolution Protocol	656
Location and Topology	658
Network Location Awareness	658



Network Connectivity Status Indicator659  
Link-Layer Topology Discovery662  
Protocol Drivers663  
Windows Filtering Platform666  
NDIS Drivers672  
Variations on the NDIS Miniport677  
Connection-Oriented NDIS677  
Remote NDIS680  
QoS682  
Binding684  
Layered Network Services685  
Remote Access685  
Active Directory686  
Network Load Balancing688  
Network Access Protection689  
Direct Access695  
Conclusion696  
Index697

## 章节摘录

版权页：插图： This logical behavior (which helps ensure that administrators will always have full control of the running code on the system) clashes with the system behavior for digital rights management requirements imposed by the media industry on computer operating systems that need to support playback of advanced, high-quality digital content such as Blu-ray and HD-DVD media. To support reliable and protected playback of such content, Windows uses protected processes. These processes exist along-side normal Windows processes, but they add significant constraints to the access rights that other processes on the system (even when running with administrative privileges) can request. Protected processes can be created by any application; however, the operating system will allow a process to be protected only if the image file has been digitally signed with a special Windows Media Certificate. The Protected Media Path (PMP) in Windows makes use of protected processes to provide protection for high-value media, and developers of applications such as DVD players can make use of protected processes by using the Media Foundation API. The Audio Device Graph process (Audiodg.exe) is a protected process because protected music content can be decoded through it. Similarly, the Windows Error Reporting (or WER, discussed in Chapter 3) client process (Werfault.exe) can also run protected because it needs to have access to protected processes in case one of them crashes. Finally, the System process itself is protected because some of the decryption information is generated by the Ksecdd.sys driver and stored in its user-mode memory. The System process is also protected to protect the integrity of all kernel handles (because the System process' handle table contains all the kernel handles on the system).



# 《深入解析Windows操作系统》

## 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 “我想不出还有哪一本书比本书更具权威性。”——微软公司副总裁 Ben Fathi



## 章节试读

### 1、《深入解析Windows操作系统》的笔记-第45页

Checked Build , DBG宏打开编译版 , 主要用于辅助设备驱动开发

### 2、《深入解析Windows操作系统》的笔记-第13页

SwitchToFiber , Fiber由用户线程自己调度 ( 但Thread本身由内核调度 ) , 64位下的UMS线程。

# 《深入解析Windows操作系统》

## 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:[www.tushu000.com](http://www.tushu000.com)