# Memory Dump Analysis Anthology

## Volumes 1 – 17

### Tables of Contents and Indexes

Dmitry Vostokov
Software Diagnostics Institute

Facebook LinkedIn Twitter

## About the Author

Dmitry Vostokov is an internationally recognized expert, speaker, educator, scientist, inventor, and author. He founded the pattern-oriented software diagnostics, forensics, and prognostics discipline (Systematic Software Diagnostics) and Software Diagnostics and Observability Institute (DA+TA: DumpAnalysis.org + TraceAnalysis.org). Vostokov has also authored over 50 books on software diagnostics, anomaly detection and analysis, software and memory forensics, root cause analysis and problem solving, memory dump analysis, debugging, software trace and log analysis, reverse engineering, and malware analysis. He has over 30 years of experience in software architecture, design, development, and maintenance in various industries, including leadership, technical, and people management roles. Dmitry founded OpenTask Iterative and Incremental Publishing (OpenTask.com) and Software Diagnostics Technology and Services (former Memory Dump Analysis Services) PatternDiagnostics.com. In his spare time, he explores Software Narratology, Software Pathology, Software Morphology, Quantum Software Diagnostics, and Pattern-Oriented AI. His interest areas are theoretical software diagnostics and its mathematical and computer science foundations, application of formal logic, semiotics, artificial intelligence, machine learning, and data mining to diagnostics and anomaly detection, software diagnostics engineering and diagnostics-driven development, diagnostics workflow and interaction. Recent interest areas also include functional programming, cloud native computing, monitoring, observability, visualization, security, automation, applications of category theory to software diagnostics, development and big data, and diagnostics of artificial intelligence.

# Memory Dump Analysis Anthology Volume 1

## Revised Edition

**Dmitry Vostokov**
**Software Diagnostics Institute**

OpenTask

Published by OpenTask, Republic of Ireland

Copyright © 2020 by Dmitry Vostokov

Copyright © 2020 by Software Diagnostics Institute

All rights reserved. No part of this book may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, without the prior written permission of the publisher.

Product and company names mentioned in this book may be trademarks of their owners.

OpenTask books are available through booksellers and distributors worldwide. For further information or comments, send requests to press@opentask.com.

A CIP catalog record for this book is available from the British Library.

ISBN-13: 978-1912636211 (Paperback)

Revision 4.01 (April 2020)

# Contents

Preface ........ 19

Acknowledgments ........ 21

About the Author ........ 23

PART 1: Crash Dumps for Beginners ........ 25

Crash Dumps Depicted ........ 25

Right Crash Dumps ........ 26

Crashes Explained ........ 28

Hangs Explained ........ 31

Symbol Files Explained ........ 34

Crashes and Hangs Differentiated ........ 36

Proactive Crash Dumps ........ 39

PART 2: Professional Crash Dump Analysis ........ 43

Minidump Analysis ........ 43

Scripts and WinDbg Commands ........ 43

Component Identification ........ 46

Raw Stack Data Analysis ........ 53

Symbols and Images ........ 63

Interrupts and Exceptions Explained ........ 68

Exceptions Ab Initio ........ 68

X86 Interrupts ........ 69

X64 Interrupts ........ 76

Interrupt Frames and Stack Reconstruction ........ 83

Trap Command on x86 ........ 92

Trap Command on x64 ........ 100

Exceptions in User Mode ........ 104

How to Distinguish Between 1st and 2nd Chances ........ 109

Who Calls the Postmortem Debugger? ........ 113

Inside Vista Error Reporting ........ 117

Another Look at Page Faults ........ 132

Bugchecks Depicted ........ 135

NMI_HARDWARE_FAILURE ........ 135

IRQL_NOT_LESS_OR_EQUAL ........ 136

KERNEL_MODE_EXCEPTION_NOT_HANDLED ........ 141

KMODE_EXCEPTION_NOT_HANDLED ........ 143

SYSTEM_THREAD_EXCEPTION_NOT_HANDLED ........ 144

CAFF ........ 150

CF ........ 152

Manual Stack Trace Reconstruction ........ 157

WinDbg Tips and Tricks ........ 167

Looking for Strings in a Dump ........ 167

Tracing Win32 API While Debugging a Process ........ 168

Exported NTDLL and Kernel Structures ........ 170

Easy List Traversing ........ 178

Suspending Threads ........ 181

Heap Stack Traces ........ 182

Hypertext Commands ........ 183

Analyzing Hangs Faster ........ 187

Triple Dereference ........ 188

Finding a Needle in a Hay ........ 191

Guessing Stack Trace ........ 193

Coping with Missing Symbolic Information ........ 199

Resolving Symbol Messages ........ 204

The Search for Tags ........ 206

Old Dumps, New Extensions ........ 212

Object Names and Waiting Threads ........ 214

Memory Dumps from Virtual Images ........ 219

Filtering Processes ........ 220

WinDbg Scripts ........ 221

First Encounters ........ 221

Yet another WinDbg Script ........ 222

Deadlocks and Critical Sections ........ 223

Security Problem ........ 224

Hundreds of Crash Dumps ........ 227

Parameterized Scripts ........ 229

Security Issues and Scripts ........ 230

Raw Stack Dump of All Threads (Process Dump) ........ 231

Raw Stack Dump of All Threads (Complete Dump) ........ 236

Case Study ... 241

Detecting Loops in Code ... 244

Crash Dump Analysis Checklist ... 251

Crash Dump Analysis Poster (HTML version) ... 254

PART 3: Crash Dump Analysis Patterns ... 255

Multiple Exceptions ... 255

Dynamic Memory Corruption ... 257

False Positive Dump ... 259

Lateral Damage ... 264

Optimized Code ... 265

Invalid Pointer ... 267

Inconsistent Dump ... 269

Hidden Exception (User Space) ... 271

Deadlock (Critical Sections) ... 276

Changed Environment ... 283

Incorrect Stack Trace ... 288

OMAP Code Optimization ... 294

No Component Symbols ... 298

Insufficient Memory (Committed Memory) ... 302

Spiking Thread ... 305

Module Variety ... 310

Stack Overflow (Kernel Mode) ... 314

Deadlock (Executive Resources) ... 323

Insufficient Memory (Handle Leak) ..... 327

Managed Code Exception ..... 331

Truncated Dump ..... 340

Waiting Thread Time (Kernel Dumps) ..... 343

Deadlock (Mixed Objects, User Space) ..... 348

Memory Leak (Process Heap) ..... 356

Missing Thread ..... 362

Unknown Component ..... 367

Memory Leak (.NET Heap) ..... 371

Double Free (Process Heap) ..... 378

Double Free (Kernel Pool) ..... 387

Coincidental Symbolic Information ..... 390

Stack Trace ..... 395

Virtualized Process (WOW64) ..... 400

Stack Trace Collection (Unmanaged Space) ..... 409

Coupled Processes (Strong) ..... 419

High Contention (Executive Resources) ..... 421

Accidental Lock ..... 423

Passive Thread (User Space) ..... 430

Main Thread ..... 437

Insufficient Memory (Kernel Pool) ..... 441

Busy System ..... 449

Historical Information ..... 458

Object Distribution Anomaly (IRP) .......... 459

Local Buffer Overflow .......... 461

Passive System Thread (Kernel Space) .......... 462

Early Crash Dump .......... 466

Hooked Functions (User Space) .......... 469

Custom Exception Handler (User Space) .......... 471

Deadlock (LPC) .......... 474

Special Stack Trace .......... 479

Manual Dump (Kernel) .......... 480

Wait Chain (General) .......... 482

Manual Dump (Process) .......... 487

Wait Chain (Critical Sections) .......... 490

PART 4: Crash Dump Analysis AntiPatterns .......... 493

Alien Component .......... 493

Zippocricy .......... 494

Word of Mouth .......... 495

Wrong Dump .......... 496

Fooled by Description .......... 497

Need the Crash Dump .......... 498

Be Language .......... 499

Fooled by Abbreviation .......... 500

PART 5: A Bit of Science .......... 501

Memory Dump - A Mathematical Definition .......... 501

Threads as Braided Strings in Abstract Space ........ 503
What is Memory Dump Analysis? ........ 506
Memorillion and Quadrimemorillion ........ 507
Four Causes of Crash Dumps ........ 508
Complexity and Memory Dumps ........ 510
What is a Software Defect? ........ 511
PART 6: Fun with Crash Dumps ........ 513
Dump Analysis and Voice Recognition ........ 513
Sending SMS Messages via Dumps ........ 514
WinDbg as a Big Calculator ........ 515
Dumps, Debuggers, and Virtualization ........ 516
Musical Dumps ........ 518
Debugging the Debugger ........ 519
Musical Dumps: Dump2Wave ........ 521
Dump Tomography ........ 522
The Smallest Program ........ 523
Voices from Process Space ........ 526
Crash Dump Analysis Card ........ 528
Listening to Computer Memory ........ 529
Visualizing Memory Dumps ........ 532
Visualizing Memory Leaks ........ 544
Picturing Computer Memory ........ 556
Unicode Illuminated ........ 559

Teaching Binary to Decimal Conversion ........ 560

Crash Dumps and Global Conspiracy ........ 561

PART 7: WinDbg For GDB Users and Vice Versa ........ 563

AT&T and Intel Syntax ........ 563

Installation ........ 565

Disassembler ........ 568

Stack Trace (Backtrace) ........ 573

Local Variables ........ 581

PART 8: Software Troubleshooting ........ 589

Four Pillars ........ 589

Five Golden Rules ........ 590

Critical Thinking ........ 591

Troubleshooting as Debugging ........ 592

PART 9: Reversing and Reconstruction ........ 593

Pooltags ........ 593

The List of Services ........ 594

Reverse Engineering Component Dependencies ........ 596

PART 10: Security ........ 599

Memory Visualization ........ 599

WinDbg is Privacy-Aware ........ 600

Crash Dumps and Security ........ 604

PART 11: The Origin of Crash Dumps ........ 605

JIT Service Debugging ........ 605

Local Crash Dumps in Vista ... 606

COM+ Crash Dumps ... 607

Correcting Microsoft Article about Userdump.exe ... 612

Where did the Crash Dump Come from? ... 616

Custom Postmortem Debuggers in Vista ... 618

Resurrecting Dr. Watson in Vista ... 621

Process Crash - Getting the Dump Manually ... 624

Upgrading Dr. Watson ... 627

Savedump.exe and Pagefile ... 628

Dumping Vista ... 629

Dumping Processes without Breaking Them ... 631

Userdump.exe on x64 ... 632

NTSD on x64 Windows ... 633

Need a Dump? Common Use Cases ... 634

PART 12: Tools ... 635

Memory Dump Analysis Using Excel ... 635

TestDefaultDebugger.NET ... 636

Cons of Symbol Server ... 637

StressPrinters: Stressing Printer Autocreation ... 638

InstantDump (JIT Process Dumper) ... 639

TestDefaultDebugger ... 641

DumpAlerts ... 643

DumpDepends ... 644

Dump Monitor Suite ........ 645

SystemDump ........ 646

PART 13: Miscellaneous ........ 649

What is KiFastSystemCallRet? ........ 649

Understanding I/O Completion Ports ........ 653

Symbol File Warnings ........ 656

Windows Service Crash Dumps in Vista ........ 658

The Road to Kernel Space ........ 664

Memory Dump Analysis Interview Questions ........ 665

Music for Debugging ........ 666

PDBFinder ........ 667

When a Process Dies Silently ........ 668

ASLR: Address Space Layout Randomization ........ 673

Process and Thread Startup in Vista ........ 678

Race Conditions on a Uniprocessor Machine ........ 680

Yet Another Look at Zw* and Nt* Functions ........ 683

Programmer Universalis ........ 686

Dr. Watson Logs Analysis ........ 687

Post-Debugging Complications ........ 690

The Elements of Crash Dump Analysis Style ........ 691

Crash Dump Analysis in Visual Studio ........ 692

32-bit Stack from 64-bit Dump ........ 694

Asmpedia ........ 695

How WINE Can Help in Crash Dump Analysis ........................................................ 696

Horrors of Debugging Legacy Code ........................................................ 697

UML and Device Drivers ........................................................ 699

Statistics: 100% CPU Spread over all Processes ........................................................ 702

Appendix ........................................................ 703

Crash Dump Analysis Portal ........................................................ 703

Reference Stack Traces ........................................................ 706

Index of WinDbg Commands ........................................................ 707

Cover Images ........................................................ 711

# Index of WinDbg Commands

!

!alpc, 252
!analyze -hang, 187, 251, 252
!analyze -show, 300
!analyze -v, 43, 44, 46, 49, 53, 73, 81, 94, 144, 187, 224, 225, 226, 227, 251, 253, 255, 271, 302, 318, 331, 339, 387, 395, 401, 405, 423, 490, 496, 513, 519, 625, 656
!bugdump, 253
!chkimg, 252, 469
!CLRStack, 253, 335, 336
!cs, 251, 252
!dh, 208, 209, 252, 298, 369
!dlls, 251
!dml_proc, 184, 185
!dpcs, 187, 252
!dumpheap, 372, 374, 375
!DumpHeap, 253
!DumpRuntimeTypes, 253
!dumpstack, 331, 333
!DumpStackObjects, 253
!eeheap, 253, 372, 374, 375
!EEHeap, 376
!EEStack, 333, 337, 338
!envvar, 251, 252
!exchain, 252, 471, 472
!exqueue, 224, 225, 226, 252, 463
!filecache, 252
!fileobj, 252
!FinalizeQueue, 253
!findstack, 409, 410
!for_each_process, 222, 223, 226
!for_each_thread, 236
!GCHandleLeaks, 253
!GCHandles, 253
!gflag, 251, 358, 384, 386
!handle, 328, 349
!heap, 182, 252, 254, 356, 357, 358, 359, 360, 361, 373, 674, 675, 677
!help, 337, 338
!htrace, 458
!idt, 75
!irp, 252, 458, 485
!irpfind, 224, 225, 226, 252, 458, 459, 635
!list, 178, 189, 190
!locks, 187, 224, 225, 226, 227, 251, 252, 269, 276, 278, 323, 324, 348, 349, 421, 423, 424, 448, 476, 491
!logc, 168
!logexts, 168, 169
!logo, 168
!lpc, 252, 458, 475, 477
!ndiskd.miniports, 252
!ndiskd.pktpools, 252
!object, 215, 216, 217, 218
!pcr, 72, 76, 77, 314, 317
!pe, 253
!peb, 252, 616, 675, 677
!pool, 132, 253, 387
!poolused, 206, 208, 224, 225, 226, 252, 328, 441, 444, 445, 447, 593
!PrintException, 335
!process, 84, 105, 214, 216, 220, 222, 225, 226, 236, 252, 270, 328, 409, 414, 415, 422, 462, 594, 702

!pte, 132, 133, 154, 155, 156, 319
!qlocks, 252
!ready, 252, 449, 452, 453
!runaway, 251, 305, 307, 471
!running, 252, 309, 449, 452, 453
!scsikd.classext, 252
!session, 252
!smt, 654
!sprocess, 252
!stacks, 187, 212, 224, 225, 226, 252, 309, 347, 409, 411, 462, 635
!sym, 204, 205
!syncblk, 253
!sysinfo, 43, 44, 45, 219, 251, 253
!teb, 109, 119, 165, 167, 179, 199, 232, 233, 236, 271, 289, 671, 676
!thread, 236, 307, 330, 344, 421, 427, 428, 429, 449, 450, 453, 454, 455, 456, 457, 475, 476, 477, 702
!Threads, 253
!uniqstack, 191, 251, 409, 430
!VerifyHeap, 253
!vm, 206, 219, 224, 225, 226, 252, 253, 303, 327, 340, 441, 442, 443, 444, 445, 447
!whattime, 347, 439
!xpool, 446

$

$$><, 233
$$>a<, 229, 530, 557

.

.asm, 88, 107, 127, 129, 130, 207, 246, 316, 450, 569, 678
.catch, 189, 221
.chain, 253, 332
.cxr, 145, 215, 273, 275, 319, 321, 398, 614, 625, 626, 652
.echo, 44
.effmach, 404
.else, 529, 556
.elsif, 529
.enumtag, 253
.exr, 144, 321, 331, 401, 405, 489
.for, 78, 79, 188, 189, 221, 222, 223, 225, 226
.foreach, 350
.formats, 133, 207, 211, 515, 560
.frame, 584, 585, 586, 587
.if, 221, 236, 529, 556
.kframes, 251
.load, 339, 404
.loadby, 332, 338, 339, 372, 376
.logclose, 44, 225, 226, 227, 235
.logopen, 44, 225, 226, 227, 233, 359
.printf, 78, 79, 188, 189, 221, 350, 635
.process, 84, 154, 155, 184, 214, 221, 222, 223, 225, 226, 328, 414, 415, 476
.reload, 35, 65, 67, 84, 200, 204, 214, 222, 223, 225, 328, 409, 410, 414, 496
.shell, 529, 530, 531, 556, 557
.symfix, 35, 49, 251, 496
.sympath, 35, 64, 200
.thread, 86, 106, 214, 215, 236, 319, 321, 398, 424, 652
.trap, 92, 93, 94, 97, 98, 100, 103, 132, 154, 302, 303, 314, 398, 652
.tss, 314, 316, 318
.writemem, 529, 556

?

?, 50, 103, 294, 344, 515
??, 344, 345

~

~, 231, 253, 364
~*, 118, 119, 126, 128, 191, 192, 197, 233, 251, 277, 279, 305, 307, 363, 364, 402, 407, 410, 450, 488, 579, 609, 625, 626, 663, 678
~*e, 233, 253
~[], 185
~~, 490, 492
~e, 233
~n, 181

B

bp, 115, 123, 130, 168, 574, 580, 585, 587

D

da, 128, 259, 282, 337
db, 164, 367, 370, 559
dd, 132, 165, 168, 178, 188, 203, 215, 319, 341, 344, 363, 364, 610, 617
dds, 74, 86, 90, 95, 120, 145, 147, 159, 160, 161, 165, 189, 209, 210, 272, 290, 320, 341, 350, 364, 417, 525, 601, 603, 679
ddu, 130
dpa, 43, 44, 167, 235
dpp, 167, 188, 189
dps, 43, 44, 53, 82, 188, 232, 233, 235, 236, 299, 650
dpu, 43, 44, 114, 167, 235, 657
dqs, 102, 105, 107, 111
dS, 251, 252, 253, 616
dt, 73, 76, 77, 78, 93, 94, 97, 100, 116, 170, 177, 178, 179, 180, 199, 200, 270, 315, 317, 323, 343, 402, 616, 617, 626, 654
du, 115, 128, 282, 391, 394, 420, 610, 611, 617
dv, 185, 584, 585, 586, 587
dx, 185

G

g, 115, 123, 130, 168, 365, 585, 669, 671, 672

K

k, 102, 103, 113, 118, 130, 140, 144, 193, 246, 247, 288, 295, 303, 318, 388, 398, 402, 407, 410, 571, 577, 624, 626, 659, 661, 663, 669, 671, 674, 678
k L=, 89, 91, 146, 148, 163, 292
kb, 279
kL, 63, 65, 67, 98, 99, 104, 106, 125, 151, 256, 273, 363, 364, 366, 380, 384, 386, 389, 439, 440, 461, 472, 473, 487, 488, 519, 520, 524, 525, 577, 579, 609, 649, 652, 701
kM, 183, 186
kn, 613, 614
knf, 319, 321
knL, 578, 584, 585
kPL, 578
kv, 43, 44, 103, 159, 164, 187, 215, 227, 251, 271, 277, 288, 305, 314, 340, 409, 580, 587, 614, 625, 651, 657
kvL, 490, 492, 577, 600, 602

L

lm, 153, 183, 199, 200, 283, 391, 392, 510, 557, 673, 676
lmD, 183, 186
lmt, 219, 310, 510

lmv, 43, 44, 47, 51, 52, 60, 61, 64, 224, 225, 226, 227, 251, 253, 298, 310, 312, 321, 341, 367, 368, 389, 399, 448, 656

ln, 73, 78, 79, 253, 300, 315, 316, 688

P

p, 123

poi, 189, 221, 222, 223, 225

R

r, 43, 63, 65, 78, 79, 81, 95, 102, 104, 124, 188, 221, 222, 225, 246, 340, 380, 385, 461, 659, 670, 672

r?, 78, 79, 221, 222, 223, 225, 233, 236

S

s, 109, 110, 210, 251, 349

T

t, 124

U

u, 44, 78, 88, 89, 107, 140, 164, 169, 210, 259, 294, 295, 296, 297, 470, 523, 525, 570, 650, 651, 683, 684, 685, 689

ub, 43, 44, 66, 67, 88, 89, 161, 162, 211, 248, 296, 297, 388, 392, 393, 571, 572, 580, 587, 624, 649, 650, 651, 657

uf, 43, 44, 66, 92, 100, 114, 127, 130, 148, 163, 207, 241, 246, 248, 265, 266, 296, 316, 354, 450, 569, 570, 572, 575, 580, 587, 650, 669, 670, 678

V

vertarget, 44, 471

X

x, 298

# Memory Dump Analysis Anthology Volume 2

## Revised Edition

**Dmitry Vostokov**
**Software Diagnostics Institute**

OpenTask

Published by OpenTask, Republic of Ireland

Copyright © 2020 by Dmitry Vostokov

Copyright © 2020 by Software Diagnostics Institute

All rights reserved. No part of this book may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, without the prior written permission of the publisher.

Product and company names mentioned in this book may be trademarks of their owners.

OpenTask books are available through booksellers and distributors worldwide. For further information or comments, send requests to press@opentask.com.

A CIP catalog record for this book is available from the British Library.

ISBN-13: 978-1912636228 (Paperback)

Revision 4.00 (May 2020)

# Contents

Preface ... 15

Acknowledgments ... 17

About the Author ... 18

PART 1: Crash Dumps for Beginners ... 19

The Time of the Crash ... 19

Stack Trace ... 20

EasyDbg ... 22

Citrix Symbol Server ... 27

PART 2: Professional Crash Dump Analysis ... 29

WinDbg Scripts ... 29

Introduction for C/C++ Users ... 29

Generating File Name for .dump Command ... 37

All at Once: Postmortem Logs and Dump Files ... 38

Common Mistakes ... 39

Not Looking at Full Stack Traces ... 39

Not Seeing Semantic and Pragmatic Inconsistencies ... 41

Pattern Interaction ... 43

Heuristic Stack Trace ... 43

Multiple Patterns ... 50

Exception and Deadlock ... 55

Heap and Spike ... 59

Hooksware ........ 63

Heap and Early Crash Dump ........ 65

WinDbg Shortcuts ........ 67

WinDbg as a Binary Editor ........ 67

Command Autocompletion ........ 70

!envvar ........ 71

.quit_lock ........ 72

.dumpcab ........ 73

.f+, .f- ........ 74

.exptr ........ 75

WinDbg as a Simple PE Viewer ........ 76

.sound_notify ........ 79

Signaled Objects ........ 80

Memory Search Revisited ........ 87

WDF and PNP BSOD: Case Study ........ 95

Exploring NDIS Extension ........ 105

The Hunt for the Debugger ........ 109

Complete Dump: User Space Critical Sections ........ 115

Microsoft DLL Help Database ........ 116

What Does This Function Do? ........ 118

What Was This Process Doing? ........ 119

STL and WinDbg ........ 122

WinDbg Cheat Sheet ........ 125

How Old Is Your Application or System? ... 126

Demystifying First-chance Exceptions ... 129

.NET Managed Code Analysis in Complete Memory Dumps ... 131

Who Opened That File? ... 134

In Search of Lost CID ... 136

Large Heap Allocations ... 137

First-order and Second-order Memory Leaks ... 140

Hooked Modules ... 145

PART 3: Crash Dump Analysis Patterns ... 147

Wait Chain (Executive Resources) ... 147

Corrupt Dump ... 151

Dispatch Level Spin ... 154

No Process Dumps ... 157

No System Dumps ... 158

Insufficient Memory (PTE) ... 159

Suspended Thread ... 161

Special Process ... 164

Frame Pointer Omission ... 169

False Function Parameters ... 173

Message Box ... 177

Self-Dump ... 181

Blocked Thread (Software) ... 184

Zombie Processes ... 196

Wild Pointer ........ 202

Dynamic Memory Corruption (Kernel Pool) ........ 204

Insufficient Memory (Module Fragmentation) ........ 210

Wild Code ........ 219

Hardware Error ........ 221

Handle Limit (GDI, Kernel Space) ........ 226

Missing Component (General) ........ 233

NULL Pointer (Code) ........ 237

Execution Residue (Unmanaged Space) ........ 239

Optimized VM Layout ........ 267

Invalid Handle (General) ........ 269

Overaged System ........ 273

Thread Starvation (Realtime Priority) ........ 274

Stack Overflow (User Mode) ........ 279

Missing Component (Static Linkage, User Mode) ........ 283

Duplicated Module ........ 294

Not My Version (Software) ........ 299

Data Contents Locality ........ 300

Nested Exceptions (Unmanaged Code) ........ 305

Nested Exceptions (Managed Code) ........ 310

Affine Thread ........ 314

Self-Diagnosis (User Mode) ........ 318

Waiting Thread Time (User Dumps) ........ 319

Inline Function Optimization (Unmanaged Code) .......... 322
Critical Section Corruption .......... 324
Lost Opportunity .......... 332
Young System .......... 335
Last Error Collection .......... 337
Hidden Module .......... 339
High Contention (Critical Sections) .......... 341
PART 4: Crash Dump Analysis AntiPatterns .......... 343
Debugging Architects .......... 343
Symbolless Analysis .......... 344
Myopic Troubleshooting and Debugging .......... 345
PART 5: A Bit of Science .......... 347
Memoretics .......... 347
Memory Analysis .......... 348
Memoidealism .......... 349
Memiotics .......... 350
PART 6: Fun with Crash Dumps .......... 351
Music for Debugging .......... 351
The Glory of Debugging .......... 351
Memory Analysis Album .......... 352
Biography of a Bug .......... 354
Visual Computer Memories .......... 355
The First Defect .......... 356

The Songs for Remote Debugging ........ 357

Thinking Out of the Box ........ 358

Crash Dumps and Science Fiction ........ 359

Colorimetric Computer Memory Dating ........ 360

On CSI Abbreviation ........ 362

The First Memory Dump Book ........ 363

On SOS Abbreviation ........ 365

Software Exceptions: a Paranormal View ........ 366

Bug Entanglement (Bugtanglement) ........ 367

The Standard Model of Debugging ........ 368

Physics of Debugging ........ 369

Can Computers Debug? ........ 371

PART 7: Data Recovery ........ 375

With the Help of Memory Dump Analysis ........ 375

PART 8: Software Troubleshooting ........ 377

Troubleshooter’s Block ........ 377

Causal Models ........ 378

Object-Oriented Debugging and Troubleshooting ........ 379

Component-Based Debugging and Troubleshooting ........ 380

Domain-Driven Debugging and Troubleshooting ........ 381

Myths and Facts about Software Support ........ 382

Ceteris Paribus in Comparative Troubleshooting ........ 383

Dancing in Software Support Environment ........ 384

PARTS: Problem Solving Power of Thought ........ 385

The Hidden Tomb in Pyramid of Software Change ........ 386

Tracing ........ 387

CDF Traces: Analyzing Process Launch Sequence ........ 387

ETW Tracing Tools ........ 389

Lean Tracing ........ 390

DebugWare Patterns ........ 391

API Query ........ 391

Tool Façade ........ 392

Configuration Wrapper ........ 393

Dual Interface ........ 394

Tool Chain ........ 395

Tool Box ........ 396

PART 9: Security ........ 397

Data Hiding in Crash Dumps ........ 397

Hardening Dump Security: Beware of PEB Data ........ 400

PART 10: The Origin of Crash Dumps ........ 401

Memory Dumps from Xen-virtualized Windows ........ 401

Bugchecks: SYSTEM_SERVICE_EXCEPTION ........ 402

Bugcheck Callbacks ........ 406

Application Verifier on x64 Platforms ........ 413

Who Saved the Dump File? ........ 414

ADPlus in 21 Seconds and 13 Steps ........ 416

PART 11: Miscellaneous ........ 425
Three Main Ideas of Debugging ........ 425
Pseudo-corrupt Memory Dumps ........ 426
Win32 Exception Frequencies ........ 427
Bugcheck Frequencies ........ 429
Time Travel Debugging ........ 440
I/O and Memory Priority in Vista ........ 441
Appendix A ........ 443
Crash Dump File Examples ........ 443
Appendix B ........ 445
WinDbg.Org: WinDbg Quick Links ........ 445
Appendix C ........ 447
Dump2Wave Source Code ........ 447
Appendix D ........ 451
Dump2Picture Source Code ........ 451
Appendix E ........ 455
Crash Dump Analysis Checklist ........ 455
CMDTREE.TXT ........ 458
Appendix F ........ 459
Index of WinDbg Commands ........ 460
Cover Images ........ 463

# Index of WinDbg Commands

!

!address, 57, 67, 137, 139, 141, 211, 213, 324, 325, 329, 330, 342
!alpc, 456
!analyze, 70
!analyze -hang, 455, 456
!analyze -show, 204, 205
!analyze -v, 38, 46, 51, 152, 169, 203, 205, 284, 310, 455, 457
!analyzeuexception, 70
!bugdump, 406, 457
!chkimg, 145, 302, 303, 455
!CLRStack, 457
!cs, 455, 456
!dc, 89, 93
!devobj, 102
!devstack, 102
!dh, 76, 216, 340, 455
!dlls, 290, 296, 298, 455
!dpcs, 456
!DumpHeap, 457
!dumpil, 133
!dumpmt, 133
!DumpRuntimeTypes, 457
!DumpStackObjects, 457
!eeheap, 457
!EEHeap, 132
!envvar, 71, 455, 456
!error, 151, 283
!exchain, 403, 456
!exqueue, 456
!filecache, 456
!fileobj, 456
!FinalizeQueue, 457
!for_each_module, 145
!for_each_process, 115, 134
!for_each_thread, 337, 338
!GCHandleLeaks, 457
!GCHandles, 457
!gflag, 455
!gle, 65, 286, 337, 338
!handle, 121, 134
!heap, 60, 137, 139, 143, 144, 342, 456
!irp, 82, 101, 180, 456
!irpfind, 456
!kdexts.handle, 83, 200
!locks, 50, 51, 55, 61, 147, 325, 326, 327, 330, 341, 371, 455, 456
!lpc, 456
!ndiskd.miniports, 456
!ndiskd.pktpools, 456
!ndiskd.protocols, 105, 106, 108
!object, 82, 187, 188
!pe, 457
!peb, 138, 289, 297, 397, 400, 455
!pool, 90, 95, 96, 100, 208, 300, 301, 457
!poolused, 226, 456
!process, 40, 53, 80, 83, 113, 131, 132, 162, 164, 165, 166, 167, 177, 185, 186, 199, 200, 336, 397, 398, 456
!pte, 96, 107
!qlocks, 456
!ready, 186, 456
!runaway, 59, 319, 321, 455
!running, 184, 274, 316, 456
!scsikd.classext, 456
!search, 88, 89, 90, 93, 94
!session, 335, 456
!sprocess, 456
!stacks, 114, 456
!stl, 122
!syncblk, 457
!sysinfo, 401, 455, 457
!teb, 44, 110, 170, 234, 241, 280, 286, 307, 333, 337, 338

!thread, 52, 80, 81, 86, 100, 150, 154, 178, 185, 187, 188, 189, 275, 276, 277, 278, 301, 314, 315, 316, 441
!threads, 132
!Threads, 457
!uniqstack, 455
!vad, 399
!VerifyHeap, 457
!vm, 52, 53, 159, 196, 335, 456, 457

## $

$$><, 31
$$>a<, 32, 33

## .

.asm, 209
.block, 31, 32, 33, 35, 36
.chain, 457
.cxr, 45, 56, 57, 190, 222, 226, 308, 309, 402
.dump, 37, 38, 73, 297, 319, 398, 399, 400
.dumpcab, 73
.else, 35
.enumtag, 406, 410, 457
.exptr, 75
.exr, 43, 222, 233, 270, 271, 279, 284, 288
.f-, 74
.f+, 74
.for, 36
.foreach, 375
.formats, 88, 202
.frame, 74, 174
.if, 33, 35, 337, 338
.ignore_missing_pages, 93
.imgscan, 339
.kframes, 38, 40, 455
.lastevent, 286
.loadby, 132
.logclose, 38
.logopen, 38
.printf, 31, 32, 33, 35, 36, 37
.process, 115, 132, 165, 167, 178, 190, 397
.quit_lock, 72
.readmem, 68
.sound_notify, 79
.symfix, 455
.thread, 52, 57, 185, 189, 190, 274, 276, 337, 338
.trap, 95, 155, 190, 300
.while, 36
.writemem, 69, 375

## ~

~, 156, 241, 315, 320, 332, 457
~*, 29, 38, 39, 59, 62, 65, 240, 332, 455
~*e, 457
~~, 341

## A

as, 37

## D

da, 318, 372
dA, 179
db, 215, 372
dc, 67, 68, 103, 340
dd, 99, 120, 123, 136, 219, 301, 397, 398
dds, 45, 64, 110, 171, 234, 241, 251, 281, 286, 333, 415
dl, 138
dpa, 124, 235
dpp, 136
dps, 64, 208, 271, 406
dpu, 235
dq, 138, 220
dqs, 308, 309
dS, 455, 456, 457
dt, 85, 86, 91, 97, 123, 124, 138, 161, 162, 301, 315, 317, 324, 325, 329, 330, 372, 400, 404, 405
du, 289, 330, 415
dv, 122
dyd, 156

E

ea, 68

F

f, 397

G

g, 79, 269

K

k, 156

k L=, 172, 183, 282

kbL, 175

kbnL, 173

kL, 43, 45, 57, 66, 107, 109, 119, 181, 190, 233, 270, 279, 280, 284, 295, 297, 299, 315, 322, 403

kn, 74

kv, 60, 75, 120, 136, 169, 237, 302, 318, 321, 372, 455

L

lm, 67, 76, 210, 212, 267, 294, 296, 298, 304, 339, 373, 375, 401

lmM, 37

lmv, 38, 103, 214, 296, 400, 455, 457

ln, 457

R

r, 32, 52, 66, 170, 185, 189, 237, 282, 295, 302, 304, 316, 322

S

s, 93, 455

T

t, 220

U

u, 42, 109, 190, 203, 209, 219, 302, 303, 304

ub, 170, 171, 173, 189, 209, 237, 271, 308

uf, 97, 155, 172, 174, 175, 404

V

vertarget, 19, 160, 320

X

x, 323

# Memory Dump Analysis Anthology
# Volume 3

## Revised Edition

**Dmitry Vostokov**
**Software Diagnostics Institute**

OpenTask

Published by OpenTask, Republic of Ireland

Copyright © 2020 by Dmitry Vostokov

Copyright © 2020 by Software Diagnostics Institute

All rights reserved. No part of this book may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, without the prior written permission of the publisher.

Product and company names mentioned in this book may be trademarks of their owners.

OpenTask books are available through booksellers and distributors worldwide. For further information or comments, send requests to press@opentask.com.

A CIP catalog record for this book is available from the British Library.

ISBN-13: 978-1912636235 (Paperback)

Revision 3.01 (July 2020)

# Contents

Preface ........ 17

Acknowledgments ........ 19

About the Author ........ 20

PART 1: Professional Crash Dump Analysis ........ 21

Sparse Complete x64 Memory Dumps ........ 21

Common Mistakes ........ 24

Not Looking at All Stack Traces ........ 24

Dump Analysis on Windows 7 ........ 28

32-bit Stack Traces from x64 Complete Memory Dumps ........ 43

Debugger Log Reading Technique ........ 48

Variable Kernel Stack in Vista and W2K8 ........ 49

Advanced Local Procedure Call WinDbg Extension ........ 52

!cs vs. !ntsdexts.locks ........ 54

Copyright as Timestamp ........ 55

NULL Data Pointer Pattern: Case Study ........ 56

Looking for Abnormal: Case Study ........ 60

Raw Stack Dump of All Threads ........ 62

Comparative Memory Dump Analysis: CPU Spikes ........ 63

Graphical Notation for Memory Dumps ........ 68

Exception Addresses from Event Logs ........ 71

The Importance of Symbols ........ 72

Platformorphism ........ 75

PART 2: Crash Dump Analysis Patterns ........ 77

Data Alignment (Page Boundary) ........ 77

Multiple Exceptions (Kernel Mode) ........ 78

C++ Exception ........ 84

Deadlock (Mixed Objects, Kernel Space) ........ 85

Wait Chain (Thread Objects) ........ 92

Divide by Zero (User Mode) ........ 96

Wait Chain (LPC/ALPC) ........ 97

Insufficient Memory (Physical Memory) ........ 104

Swarm of Shared Locks ........ 107

Process Factory ........ 112

Paged Out Data ........ 118

Semantic Split ........ 120

Pass-Through Function ........ 129

NULL Pointer (Data) ........ 131

JIT Code (.NET) ........ 132

PART 3: Crash Dump Analysis AntiPatterns ........ 137

No Question ........ 137

Missing Space ........ 138

PART 4: Pattern Interaction ........ 141

Early Crash Dump, Blocked Thread, Not My Version, and Lost Opportunity ........ 141

Lateral Damage, Stack Overflow, and Execution Residue ........ 144

Truncated Dump, Spiking Thread, Not My Version, and Hooked Functions ........ 149

Stack Trace Collection, Hidden Exception, and NULL Code Pointer ........ 155

WOW64, Blocked Threads, and Coupled Processes ........ 160

Invalid Handle, Stack Trace Collection, Multiple Exceptions, Invalid Pointer, Data Alignment on Page Boundary, Dynamic Memory Corruption, and Not My Version 163

Wait Chain and Spiking Thread ........ 167

Blocked GUI Thread, Wait Chain, and Virtualized Process ........ 170

Insufficient Memory, Handle Leak, Wait Chain, Deadlock, Inconsistent Dump, and Overaged System ........ 175

Memory Leak, Spiking Threads, Wait Chain, High Critical Section Contention, and Module Variety ........ 181

NULL Code Pointer, Changed Environment, Hooked Functions, and Execution Residue ........ 196

Swarm of Shared Locks, Blocked Threads, and Waiting Time ........ 201

Stack Trace Collection, Blocked Thread, and Coupled Processes ........ 205

Insufficient Memory, Handle Leak, Process Factory, High Contention, and Busy System ........ 209

Busy System, Blocked Threads, Wait Chains, and Deadlock .................................... 215

Manual Dump, Dynamic Memory Corruption, Blocked Threads, Stack Trace Collection, Multiple Exceptions, Wait Chains and Deadlock..................................................... 224

Coupled Processes, Wait chains, Message Box, Waiting Thread Time, Paged Out Data, Incorrect Stack Trace, Hidden Exception, Unknown Component, and Execution Residue ........................................................................................................... 228

Manual Dump, Wait Chain, Blocked Thread, Dynamic Memory Corruption, and Historical Information ............................................................................... 236

Blocked Threads, Message Box, and Self-Diagnosis............................................... 240

Manual and Early Crash Dump, Stack Trace Collection, Main Thread, Blocked Threads, and Pass-Through Functions ........................................................................... 241

Blocked Thread, Historical Information, Execution Residue, Hidden Exception, Dynamic Memory Corruption, Incorrect Stack Trace, and Not My Version ........................... 245

Null Data Pointer, Incorrect Stack Trace, Changed Environment, Hooked Functions, and Coincidental Symbolic Information....................................................................... 248

Heap Corruption, Module Variety, Execution Residue, Coincidental Symbolic Information, and Critical Section Corruption .......................................................... 255

Stack Trace Collection, Blocked Threads, Pass-Through Functions, and Main Thread ........................................................................................................... 262

Stack Trace, Invalid Code Pointer, and Hooked Functions...................................... 264

Manual Dump, Virtualized Process, Stack Trace Collection, Multiple Exceptions, Optimized Code, Wild Code Pointer, Incorrect Stack Trace, and Hidden Exception 268

Main Blocked Thread, Missing Component, Execution Residue, and Data Contents Locality ............................................................................................................ 275

Inconsistent Dump, Blocked Threads, Wait Chains, Incorrect Stack Trace, and Process Factory ............................................................................................................ 279

Invalid Pointer, Incorrect Stack Trace, Multiple Exceptions, Insufficient Memory, and Memory Leak ........ 288

PART 5: A Bit of Science and Philosophy ........ 295

Universal Memory Dump: A Definition ........ 295

The Source of Intuition about Infinite ........ 296

Geometrical Debugging ........ 297

Riemann Programming Language ........ 299

Is Memory Dump Analysis a Science? ........ 300

My Dangerous Idea: Parameterized Science ........ 301

Unique Events and Historical Narratives ........ 302

Notes on Memoidealism ........ 303

A Copernican Revolution in Debugging ........ 305

On Subjectivity of Software Defects ........ 306

Memory Field Theories of Memuonics ........ 307

Software Trace: A Mathematical Definition ........ 308

Quantum Memory Dumps ........ 309

Chemistry of Virtual Memory ........ 310

PART 6: Fun with Crash Dumps ........ 313

Music for Debugging ........ 313

Bugs Never Disappear ........ 313

Horrors of Computation ........ 314

Passion, Intellect, and Expression ... 315

Headphones for Debugging ... 316

In the Memory Dump File ... 317

Bugteriology ... 318

Implausible Debugging Book Titles ... 319

Build Date Astrology ... 320

Breaking Technical Barrier ... 321

Occult Debugging ... 322

The Year of Dump Analysis! ... 323

Stack Traces and Poetry ... 324

Debugging Slang ... 326

Memory Dump Analysis Walks ... 327

E-Acheri ... 329

The Meaning of DATA ... 330

Irish Government on Dumps ... 331

Memory Dumps as Relics ... 332

The Ghost of Adelphi Training Center ... 333

PART 7: Software Troubleshooting ... 335

I’m RARE ... 335

To Bugcheck or Not To Bugcheck ... 336

T&D Labyrinth ............................................................................................... 337

Efficient vs. Effective: DATA View ................................................................ 339

PART 8: Software Trace Analysis........................................................................ 341

Tracing Best Practices ................................................................................... 341

Software Narratology: A Definition................................................................. 342

PART 9: Software Trace Analysis Patterns .......................................................... 343

Introduction ................................................................................................... 343

Periodic Error ................................................................................................. 344

Basic Facts ...................................................................................................... 345

Circular Trace ................................................................................................. 346

Intra-Correlation ............................................................................................ 347

PART 10: The Origin of Crash Dumps.................................................................. 351

Hide, Seek, and Dump .................................................................................... 351

OSMOSIS Memory Dumps .............................................................................. 353

Tools................................................................................................................ 356

Crash2Hang ................................................................................................. 356

MTCrash ...................................................................................................... 358

Where did the Crash Dump Come from?.......................................................... 363

FinalExceptionHandler ................................................................................... 364

PART 11: Memory Visualization.......................................................................... 367

The Art of Memory Corruption ........ 367

Visualizing Secondary Storage ........ 368

Pictures from Memory Space ........ 369

PART 12: Miscellaneous ........ 375

Hexadecimal/Decimal Chaos ........ 375

The Measure of Debugging and Memory Dump Analysis Complexity ........ 376

How To Simulate a Process Hang? ........ 377

A Windows Case for Delta Debugging ........ 378

Sentinel Pointers ........ 380

Collapsed Stack Trace ........ 381

Appendix A ........ 383

Crash Dump File Examples ........ 383

Appendix B ........ 385

Crash Dump Analysis Checklist ........ 385

Appendix C ........ 389

Memory Dump Analysis Pattern: A Definition ........ 389

Wait Chain Patterns ........ 389

DLL Link Patterns ........ 389

Insufficient Memory Patterns ........ 390

Dynamic Memory Corruption Patterns ........ 390

Deadlock Patterns ........................................................................................ 390

Index of WinDbg Commands ........................................................................... 391

Cover Images.................................................................................................. 394

# Index of WinDbg Commands

!

!address, 22, 23, 166
!alpc, 52, 386
!analyze -hang, 385, 386
!analyze -v, 56, 72, 78, 85, 144, 149, 163, 224, 227, 236, 241, 269, 270, 275, 288, 376, 385, 387
!bugdump, 387
!chkimg, 153, 197, 248, 385
!CLRStack, 387
**!cs**, 54, 171, 172, 233, 237, 240, 260, 385, 386
!dh, 38, 234, 253, 385
!dlls, 276, 385
!dpcs, 386
!DumpHeap, 387
!DumpRuntimeTypes, 387
!DumpStackObjects, 387
!eeheap, 387
!envvar, 385, 386
!error, 143
!exchain, 386
!exqueue, 386
!filecache, 386
!fileobj, 243, 386
!FinalizeQueue, 387
!GCHandleLeaks, 387
!GCHandles, 387
!gflag, 166, 385
!handle, 94, 162
!heap, 181, 386
!irp, 57, 59, 243, 386
!irpfind, 386
!locks, 22, 24, 54, 85, 86, 105, 107, 120, 141, 167, 189, 212, 217, 226, 262, 313, 385, 386
!lpc, 52, 98, 101, 118, 171, 173, 177, 178, 179, 229, 282, 386
!ndiskd.miniports, 386
!ndiskd.pktpools, 386
!pcr, 145
!pe, 387
!peb, 23, 385
!pool, 387
!poolused, 175, 293, 386
!process, 28, 43, 44, 48, 52, 61, 97, 106, 113, 116, 120, 146, 176, 211, 214, 230, 242, 262, 279, 285, 386
!pte, 22, 23, 77, 80, 289, 290
!qlocks, 386
!ready, 111, 213, 386
!runaway, 63, 64, 185, 385
!running, 79, 111, 138, 213, 215, 262, 386
!scsikd.classext, 386
!session, 107, 216, 386
!sprocess, 210, 292, 386
!stacks, 24, 244, 262, 386
!syncblk, 387
!sysinfo, 385, 387
!teb, 47, 62, 73, 143, 157, 198, 231, 246, 250, 258, 276, 364
!thread, 79, 88, 89, 92, 93, 99, 100, 101, 102, 104, 109, 110, 111, 118, 119, 122, 123, 124, 125, 126, 127, 138, 139, 150, 169, 171, 172, 173, 174, 176, 177, 178, 179, 180, 203, 204, 215, 216, 221, 222, 228, 229, 231, 244, 282, 283, 284
!Threads, 387
!uniqstack, 385
!VerifyHeap, 387
!vm, 21, 60, 105, 112, 152, 175, 209, 286, 291, 386, 387

.

.asm, 57, 133, 264
.bugcheck, 104
.chain, 387
.cxr, 76, 158, 165, 233, 238, 246, 247, 272, 361
.dump, 195, 378
.effmach, 160, 269
.enumtag, 387
.exptr, 133, 165, 272
.kframes, 224, 385
.load, 45, 160, 269
.process, 23, 28, 44, 115, 152, 172, 214, 230, 262, 287
.reload, 46
.symfix, 385
.thread, 43, 45, 79, 117, 158, 165, 214, 231, 233, 238, 247, 262
.trap, 56, 77, 78, 80, 144, 288, 290

~

~, 64, 65, 66, 79, 145, 151, 188, 205, 207, 272, 290, 324, 387
~*, 155, 163, 225, 245, 270, 359, 361, 377, 385
~*e, 62, 387
~~, 142, 189

D

da, 214, 231, 260
dc, 191, 278
dd, 93, 161, 162, 166, 208
ddp, 208
dds, 39, 47, 73, 80, 143, 147, 199, 232, 246, 250, 258, 264, 265, 364
dpa, 277
dps, 267
dpu, 235, 277
dq, 22, 159
dqs, 157
dS, 385, 386, 387
dt, 58, 59, 87, 94, 146
du, 22, 117, 240, 278

K

k, 196, 205
k L=, 148
kb, 225
kc, 270, 381
kL, 49, 74, 75, 132, 141, 142, 155, 158, 160, 163, 238, 245, 247, 255, 262, 264, 273, 324, 359, 361
kv, 46, 50, 57, 64, 65, 66, 117, 133, 142, 145, 160, 161, 165, 188, 189, 207, 214, 233, 248, 272, 290, 380, 385
kv L=, 235

L

lm, 191, 245, 253
lmt, 142, 238, 247, 256
lmv, 37, 55, 110, 150, 162, 163, 190, 198, 233, 234, 274, 287, 320, 385, 387
ln, 71, 231, 387

N

n, 377

R

r, 65, 66, 72, 75, 80, 131, 196, 248, 380

S

s, 79, 385

## U

u, 57, 67, 133, 134, 153, 154, 197, 198, 199, 249, 251, 253, 259, 266, 267, 289, 380
ub, 50, 67, 131, 134, 135, 158, 199, 251, 252, 259, 265, 289

## V

version, 180, 241, 275
vertarget, 28

# Memory Dump Analysis Anthology
# Volume 4

## Revised Edition

**Dmitry Vostokov**
**Software Diagnostics Institute**

OpenTask

Published by OpenTask, Republic of Ireland

Copyright © 2021 by Dmitry Vostokov

Copyright © 2021 by Software Diagnostics Institute

All rights reserved. No part of this book may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, without the prior written permission of the publisher.

Product and company names mentioned in this book may be trademarks of their owners.

OpenTask books are available through booksellers and distributors worldwide. For further information or comments, send requests to press@opentask.com.

A CIP catalog record for this book is available from the British Library.

ISBN-13: 978-1912636242 (Paperback)

Revision 3.01 (March 2021)

# Contents

Preface ........................................................................................................ 17

Acknowledgments ........................................................................................ 19

About the Author .......................................................................................... 20

PART 1: Professional Crash Dump Analysis and Debugging ............................... 21

Common Mistakes ........................................................................................ 21

Not Using Checklists ...................................................................................... 21

Not Paying Attention to All Aspects of Default Analysis ...................................... 23

Not Paying Attention to Context ....................................................................... 26

Raw Stack Dump of WOW64 Process ................................................................ 31

On Space and Mode ...................................................................................... 35

Registry Corruption: A Case Study ................................................................... 36

Wild Code and Partial Stack Reconstruction ....................................................... 39

Manual Parameter Reconstruction on x64 Windows Systems ................................. 42

Counterfactual Debugging ............................................................................... 46

Dereference Fixpoints ..................................................................................... 46

Data Ordering ................................................................................................ 48

Clean Raw Stack Execution Residue ................................................................. 64

Essential and Derived Properties ...................................................................... 71

Software Defect Researcher: A New Profession ........ 74

WinDbg Shortcuts ........ 75

lmu and lmk ........ 75

.opendump ........ 80

Live Kernel Debugging of System Freeze ........ 82

Mode-Independent WinDbg Scripts ........ 91

PART 2: Crash Dump Analysis Patterns ........ 93

Succession of Patterns ........ 93

Ubiquitous Component (User Space) ........ 94

Nested Offender ........ 120

Hunting for a Driver ........ 124

Virtualized System ........ 131

Effect Component ........ 137

Well-Tested Function ........ 144

Mixed Exception ........ 145

Random Object ........ 150

Not My Version (Hardware) ........ 153

Missing Process ........ 154

Platform-Specific Debugger ........ 156

Value Deviation (Stack Trace) ........ 159

CLR Thread ........ 163

Insufficient Memory (Control Blocks) ........ 166

PART 3: Crash Dump Analysis AntiPatterns ........ 167

Habitual Reply ........ 167

PART 4: Pattern Interaction ........ 169

Null Data Pointer, Pass-Through Functions, and Platformorphic Fault ........ 169

Stack Trace Collection, Message Box, Hidden Exception, Nested Offender, Insufficient Memory, C++ Exception, Heap Leak, and Ubiquitous Component ........ 172

Blocked LPC Thread, Coupled Processes, Stack Trace Collection, and Blocked GUI Thread ........ 181

Virtualized Process, Incorrect Stack Trace, Stack Trace Collection, Multiple Exceptions, Optimized Code, and C++ Exception ........ 182

NULL Data Pointer, Stack Trace, Inline Function Optimization, and Platformorphic Fault ........ 189

Stack Trace Collection, Suspended Threads, Not My Version, Special Process, Main Thread, and Blocked LPC Chain Threads ........ 192

Truncated Dump, Stack Trace Collection, Waiting Thread Time, and Wait Chains .. 200

ALPC Wait Chain, Missing Threads, Message Box, Zombie, and Special Processes .. 202

Critical Section High Contention and Wait Chains, Blocked Threads and Periodic Error: Memory Dump and Trace Analysis Pattern Cooperation ........ 208

WOW64 Process, NULL Data Pointer, Stack Overflow, Main Thread, Incorrect Stack Trace, Nested Exceptions, Hidden Exception, Manual Dump, Multiple Exceptions, and Virtualized System ........ 211

Statement Current, Coupled Processes, Wait Chain, Spiking Thread, Hidden Exception, Message Box, and Not My Version ........ 223

Stack Trace Collection, Missing Threads, Waiting Thread Time, Critical Section, and LPC Wait Chains ........ 226

Wait Chain, Blocked Thread, Waiting Thread Time, IRP Distribution Anomaly, and Stack Trace Collection........ 231

PART 5: A Bit of Science and Philosophy........ 235

Memory Exponentiation (PowerSet) ........ 235

Memory Dump View of Artificial Intelligence ........ 236

Memoidealism as Monistic Aspect Pluralism........ 237

Memory Dumps as Posets........ 239

Metaphorical Bijectionism: A Method of Inquiry........ 241

Notes on Memoidealism ........ 246

Panmemorism ........ 247

Qubic Memory Representation........ 248

Manifold Memory Space........ 250

Ars Recordatio........ 252

Categories for the Working Software Defect Researcher ........ 253

MemD Category ........ 253

Operating Closure of Memory ........ 256

Memoidealism Defined........ 258

Memuon: A Definition ........ 259

PART 6: Fun with Crash Dumps........ 261

Music for Debugging ........ 261

THE ALL MIGHTY DEBUGGER ........ 261

Memory Space Music ........ 262

The Duet of Threads ........ 263

The Memory Dump of the Dead ........ 264

Ancient Computations and a Vision of the New Dump ........ 265

The Meaning of DUMP ........ 266

Memory Analysis Ritual ........ 267

The Intelligent Memory Movement ........ 268

Moving towards the Psi Point ........ 269

Experiments on Poor Bugs ........ 270

Exception Processing Of Crash Hypothesis (EPOCH) ........ 271

Debugging Slang ........ 272

SAD Events ........ 272

BoBo Address ........ 273

Mad Day ........ 274

Bug-sistential and Bug-sistentialism ........ 275

Debugging Spy Network ........ 276

Games for Debugging: Go ........ 277

The Tsar of Memory Dump Analysis ........ 278

DNA and RNA of Ruptured Computation ........ 279

BAD0B0B0 Address: Childhood Memories ........ 280

Bugs in Passing ........ 281

Named Process: Vostokov.exe ........ 283

Memory Analysts and Debuggers Day ........ 286

After Volume 3 ........ 287

Crash, Core, and Memory Dumps in Science Fiction and Fantasy ........ 288

Reasoning with a Bug ........ 301

PART 7: Software Troubleshooting ........ 303

RADII and SDSD ........ 303

Epistemic Troubleshooting and Debugging ........ 304

RADII Process Illustrated ........ 305

Debugware Patterns ........ 307

Trace Expert ........ 307

Troubleshooting Unit of Work ........ 308

Checklist ........ 309

Supporting Module ........ 310

Span Differentiator ........ 311

Self-Extractor ........ 312

A Case Study ........ 314

Can Software Tweet? ........................................................................................ 319

The Law of Simple Tools.................................................................................... 320

Workaround Patterns ........................................................................................ 321

Hidden Output ............................................................................................... 321

Frozen Process ............................................................................................... 324

Axed Code ...................................................................................................... 325

PART 8: Software Trace Analysis........................................................................ 327

The Tool for Analysis of ETW Traces ................................................................ 327

There ought to be a Planet at that Location!..................................................... 328

Software Trace: Bird’s Eye View........................................................................ 329

Extending Multithreading to Multibraiding (Adjoint Threading) ............................. 330

PART 9: Software Trace Analysis Patterns ......................................................... 335

Statement Density and Current ........................................................................ 335

Exception Stack Trace ...................................................................................... 337

Thread of Activity............................................................................................. 339

Discontinuity .................................................................................................... 341

Missing Component ......................................................................................... 342

Bifurcation Point .............................................................................................. 343

Characteristic Message Block............................................................................. 345

Activity Region ................................................................................................. 348

Vocabulary Index ........ 349

Inter-Correlation ........ 350

PART 10: The Origin of Crash Dumps ........ 353

Full Page Heap Settings on x64 Windows ........ 353

Memory Dumps from Hyper-Virtualized Windows ........ 354

Fiber Bundle of Memory Space ........ 357

On Self Dumps of Secure String API ........ 358

PART 11: Memory Visualization ........ 361

Pictures from Memory Space ........ 361

Large-scale Structure of Memory Space ........ 363

Advanced Memory Visualization ........ 365

3D Memory Visualization ........ 376

Memory Map Visualization Tools ........ 389

PART 12: Art ........ 391

Opcodism: The Art of Opcodes ........ 391

Memory Dump and Minidumps ........ 394

Hot Issues from Physicalist Artist Perspective ........ 395

Memory Dumps from Physicalist Artist Perspective ........ 396

Memory Hot Spot and the Illusion of Fix ........ 397

Shared Section ........ 398

Memory Space Road to the Ultimate Fix ........ 399

Structure and Noise ........ 400

PART 13: Miscellaneous ........ 401

Assembling Code in WinDbg ........ 401

Free Stack Traces ........ 403

Stack Space and Program Database Types ........ 405

The Longest Stack Trace ........ 409

Software Victimology ........ 414

Debugger as a Shut up Application ........ 415

Two Great Windows Software Engineering Magazines ........ 416

Appendix ........ 417

Crash Dump Analysis Checklist ........ 417

Index of WinDbg Commands ........ 421

Cover Images ........ 423

# Index of WinDbg Commands

!

!alpc, 203, 204, 418
!analyze -hang, 417, 418
!analyze -v, 23, 26, 27, 30, 36, 84, 125, 131, 145, 156, 157, 169, 172, 182, 183, 189, 219, 267, 417, 419
!bugdump, 419
!chkimg, 23, 417
!CLRStack, 419
!cs, 208, 229, 417, 418
!devobj, 126, 128
!devstack, 130
!dh, 417
!dlls, 417
!dpcs, 418
!DumpHeap, 419
!dumpobj, 157
!DumpRuntimeTypes, 419
!dumpstack, 27, 30, 156, 157
!DumpStackObjects, 419
!eeheap, 419
!envvar, 417, 418
!error, 86, 223, 358
!exchain, 123, 148, 418
!exqueue, 418
!filecache, 166, 418
!fileobj, 71, 130, 418
!FinalizeQueue, 419
!for_each_thread, 91
!GCHandleLeaks, 419
!GCHandles, 419
!gflag, 417
!handle, 71
!heap, 173, 178, 179, 418
!irp, 126, 128, 418
!irpfind, 232, 418
!locks, 21, 231, 417, 418
!lpc, 196, 197, 230, 418
!ndiskd.miniports, 418
!ndiskd.pktpools, 418
!pe, 145, 157, 419
!peb, 417
!pool, 151, 419
!poolused, 418
!process, 75, 154, 193, 194, 195, 196, 200, 202, 226, 228, 230, 283, 418
!pte, 39
!qlocks, 418
!ready, 418
!runaway, 64, 160, 224, 239, 409, 417
!running, 133, 418
!scsikd.classext, 418
!session, 418
!sprocess, 418
!stacks, 150, 200, 226, 233, 418
!syncblk, 419
!sysinfo, 153, 356, 417, 419
!teb, 31, 32, 48, 121, 146, 161, 212, 224
!thread, 37, 39, 91, 133, 137, 150, 151, 204, 205, 221, 229, 230, 231, 234
!Threads, 419
!uniqstack, 417
!VerifyHeap, 419
!vm, 154, 200, 202, 206, 220, 283, 418, 419
!whattime, 233
!wow64exts.info, 33, 65, 67, 186, 216

.

.asm, 44, 121, 126, 132, 148, 187
.chain, 419
.cxr, 39, 44, 86, 87, 122, 125, 147, 186, 187, 193, 208, 209, 219

.effmach, 32, 34, 182, 217
.enumtag, 419
.exptr, 185, 358
.exr, 84, 122, 125, 145, 156, 182, 186, 187, 193, 194, 219, 222
.kframes, 91, 417
.load, 27, 182
.opendump, 80
.process, 192, 208, 228, 284
.reboot, 87
.symfix, 417
.thread, 39, 44, 87, 122, 147, 187, 193, 208, 209, 229
.trap, 25, 39, 189, 190, 193, 194

|

||, 80

~

~, 419
~*, 417
~*e, 419
~*k, 184
~*kc, 94, 213
~~, 22

B

bc, 62, 326
bm, 325
bp, 42, 53, 56, 70, 326

D

dds, 34, 40, 66, 67, 69, 70, 121, 133, 161, 216, 224
dp, 32, 34, 53, 56, 57, 60, 62, 128, 132, 151
dps, 33, 34, 37, 137, 147, 186, 188
dpu, 38
dqs, 34, 218, 222
dS, 417, 418, 419
dt, 72, 128, 129, 151, 152, 209

K

kc, 22, 169, 170, 211, 213, 220
kcf, 212
KERNEL_MODE_EXCEPTION_NOT_HANDLED, 23
kL, 27, 65, 68, 81, 120, 122, 146, 147, 156, 159, 163, 167, 189, 190, 208, 225, 325, 360, 391, 393, 409
kv, 36, 38, 43, 44, 87, 91, 132, 185, 187, 193, 209, 358, 404, 417

L

lm, 222, 354
lmk, 75, 76
lmu, 75, 76
lmv, 75, 194, 225, 284, 417, 419
ln, 419

S

s, 417

U

ub, 45, 121, 122, 126, 160, 162, 187, 215, 359, 392, 393
uf, 42, 49, 53, 127, 132, 148, 359, 401, 405, 406, 407, 408

# Memory Dump Analysis Anthology
# Volume 5

## Revised Edition

**Dmitry Vostokov**
**Software Diagnostics Institute**

OpenTask

Published by OpenTask, Republic of Ireland

Copyright © 2021 by Dmitry Vostokov

Copyright © 2021 by Software Diagnostics Institute

All rights reserved. No part of this book may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, without the prior written permission of the publisher.

Product and company names mentioned in this book may be trademarks of their owners.

OpenTask books are available through booksellers and distributors worldwide. For further information or comments, send requests to press@opentask.com.

A CIP catalog record for this book is available from the British Library.

ISBN-13: 978-1912636259 (Paperback)

Revision 3.00 (September 2021)

# Contents

Preface ........ 17

Acknowledgements ........ 19

About the Author ........ 20

PART 1: Professional Crash Dump Analysis and Debugging ........ 21

Common Mistakes ........ 21

Not Double-Checking Symbolic Output ........ 21

Not Looking Past the First Found Evidence ........ 24

Not Recognizing Data as UNICODE or ASCII Fragments ........ 26

Common Questions ........ 28

What Service is This? ........ 28

Complete Stack Traces from x64 System ........ 30

Software Behavior Patterns ........ 32

Crash and Hang Analysis Audit Service ........ 33

Case Study: Extremely Inconsistent Dump and CPU Spike ........ 34

Raw Stack Dump of All Thread Stacks ........ 39

Architecture of CARE ........ 41

PART 2: Crash Dump Analysis Patterns ........ 43

Succession of Patterns ........ 43

Wait Chain (Process Objects) ........ 49

Coincidental Frames ........ 55

Fault Context ........ 59

Coupled Processes (Weak) ........ 60

Hooked Functions (Kernel Space) ........ 63

Hardware Activity ........ 66

Incorrect Symbolic Information ........ 71

Message Hooks ........ 76

Blocked Thread (Hardware) ........ 79

Coupled Machines ........ 81

High Contention (Processors) ........ 82

Thread Starvation (Normal Priority) ........ 85

Coupled Processes (Semantics) ........ 87

Abridged Dump ........ 88

Exception Stack Trace ........ 93

Wait Chain (RPC) ........ 95

Distributed Spike ........ 99

Instrumentation Information ........ 108

Template Module ........ 112

Invalid Exception Information ........ 116

Shared Buffer Overwrite ........................................................................................ 120

Pervasive System........................................................................................................ 125

Problem Exception Handler ........................................................................................ 126

Deadlock (Self) ............................................................................................................ 127

Same Vendor................................................................................................................ 128

PART 3: Crash Dump Analysis AntiPatterns ................................................................ 129

Wild Explanations........................................................................................................ 129

PART 4: Pattern Interaction ......................................................................................... 133

Inconsistent Dump, Stack Trace Collection, LPC, Thread, Process, Executive Resource Wait Chains, Missing Threads, and Waiting Thread Time....................................... 133

Fault Context, Wild Code, and Hardware Error ....................................................... 137

Main Thread, Critical Section Wait Chains, Critical Section Deadlock, Stack Trace Collection, Execution Residue, Data Contents Locality, Self-Diagnosis, and Not My Version ......................................................................................................................... 145

Strong Process Coupling, Stack Trace Collection, Critical Section Corruption, Wait Chains, Message Box, Self-Diagnosis, Hidden Exception, and Dynamic Memory Corruption ..................................................................................................................... 158

Object Distribution Anomaly, Inconsistent Dump, Execution Residue, Hardware Activity, Coincidental Symbolic Information, Not My Version, and Virtualized System ....................................................................................................................................... 169

Spiking Thread, Main Thread, Message Hooks, Hooked Functions, Semantic Split, Coincidental Symbolic Information, and Not My Version....................................... 180

Stack Trace Collection, Special Process, LPC and Critical Section Wait Chains, Blocked Thread, Coupled Machines, Thread Waiting Time, and Object Distribution Anomaly ....................................................................................................................................... 188

ALPC Wait Chains, Missing Threads, Waiting Thread Time and Semantic Process Coupling ........................................................................................................ 200

Insufficient Kernel Pool Memory, Spiking Thread, and Data Contents Locality........ 201

Incorrect Stack Trace, Stack Overflow, Early Crash Dump, Nested Exception, Problem Exception Handler, and Same Vendor ..................................................................... 206

PART 5: A Bit of Science and Philosophy........................................................................ 213

Memory Systems Language ......................................................................................... 213

Categories for the Working Software Defect Researcher ........................................ 214

Collective Pointer ......................................................................................................... 214

Notes on Memoidealism ............................................................................................... 217

Archaeological Foundations for Memory Analysis.................................................... 218

On God and Miracles..................................................................................................... 220

Psychoanalysis of Software Troubleshooting and Debugging ................................. 221

Ontological and Epistemological Memoidealism ...................................................... 222

On Unconscious ............................................................................................................ 223

Ruminations on Automated Debugging...................................................................... 224

General Memory Analysis ............................................................................................ 225

Notation for Memory and Trace Analysis ................................................................. 226

Category Theory and Troubleshooting ..................................................................... 227

Software Chorography and Chorology: A Definition................................................. 229

PART 6: Fun with Crash Dumps..................................................................................... 231

Music for Debugging ........................................ 231

Retry, Abort, Escape ........................................ 231

Debugging Slang ........................................ 232

STUPID ........................................ 232

On the Same Page ........................................ 233

.SYS ........................................ 234

PLOT ........................................ 235

Freedom ........................................ 236

Free Verse ........................................ 237

BCE, BC, and CE ........................................ 238

HCI ........................................ 239

Blog ........................................ 240

Inherit a Fortune ........................................ 241

Dr. Watson’s Observational Patterns ........................................ 242

Memory Dumps in Myths ........................................ 245

Bus Debugging ........................................ 246

Debugging the Debugger (16-bit) ........................................ 247

Dr. DebugLove and Nature ........................................ 249

Sailing Memory Spaces under an RGB Flag ........................................ 253

Don’t Name Your Driver a “Missile” ........................................ 254

Notepad Debugging ... 255

!analyze -vostokov ... 263

Contemplating Crash Dumps in Unicode ... 264

Memory Dump Analysis Services Cap and T-Shirt ... 266

Troubleshooting Poem in Six Stanzas ... 267

On the Interpretation of M-Theory ... 268

Check the Name of Your Driver in Reverse ... 269

PART 7: Software Trace Analysis ... 271

Pattern Interaction ... 271

Adjoint Threads, Discontinuity, and Time Delta ... 271

Basic Software PLOTs ... 272

Two Readings of a Software Trace ... 274

CDFMarker Tool ... 276

The Extended Software Trace ... 277

Presenting a Software Story ... 278

Adjoint Threading in Process Monitor ... 279

PART 8: Software Trace Analysis Patterns ... 281

Significant Event ... 281

Time Delta ... 282

Adjoint Thread of Activity ... 283

Trace Acceleration ........ 284

Incomplete History ........ 286

Background and Foreground Components ........ 287

Defamiliarizing Effect ........ 290

Anchor Messages ........ 293

No Trace Metafile ........ 296

No Activity ........ 297

Trace Partition ........ 299

Truncated Trace ........ 301

Diegetic Messages ........ 302

False Positive Error ........ 303

Guest Component ........ 304

Message Change ........ 305

Layered Periodization ........ 306

PART 9: Models of Software Behaviour ........ 311

Multiple Exceptions Pattern ........ 311

Memory Leak (Process Heap) Pattern ........ 315

Message Hooks Pattern ........ 326

Modeling C++ Object Corruption ........ 330

PART 10: The Origin of Crash Dumps ........ 335

More on Demystifying First-chance Exceptions ..................... 335

PART 11: Structural Memory Patterns ..................... 343

Memory Snapshot ..................... 343

Aggregate Snapshot ..................... 345

Snapshot Collection ..................... 346

Memory Region ..................... 347

Region Boundary ..................... 348

Memory Hierarchy ..................... 350

Anchor Region ..................... 351

PART 12: Memory Visualization ..................... 353

Memory Map Visualization Tools (Revised) ..................... 353

Decomposing Memory Dumps via DumpFilter ..................... 355

Can a Memory Dump be Blue? ..................... 359

Virtual to Physical Memory Mapping ..................... 360

The Memory Visualization Question ..................... 363

PART 13: Art ..................... 375

Sweet Oil of Memory ..................... 375

Night Sky ..................... 376

Component Trace ..................... 377

Ana-Trace-Log-Lyzer and Closed Session ..................... 378

Computer Memory Gardens ........ 380

Debugging Venue ........ 381

Inside a Memory File ........ 382

Fabric of Memory Dumps ........ 383

Race Condition in a Kernel Pool ........ 394

Memory Interfaces ........ 395

Bleeding Memory ........ 396

Picture Frame for Memory Dumps ........ 398

Front Cover Glitch ........ 399

Chance Exceptions in a Turing Machine ........ 400

PART 14: Security and Malware Analysis ........ 401

Crash Dumps and Password Exposure ........ 401

Crash Dump Analysis of Defective Malware ........ 406

PART 15: Miscellaneous ........ 411

Native Script Debugging ........ 411

Component Heap ........ 414

Attached Processes ........ 416

User/Kernel Diagramming Styles ........ 419

Appendix ........ 423

Contention Patterns ........ 423

Raw Stack Analysis Scripts .......... 424

Crash Dump Analysis Checklist .......... 425

Index of WinDbg Commands .......... 429

Cover Images .......... 431

# Index of WinDbg Commands

!

!address, 317, 363, 372
!alpc, 50, 52
!analyze, 11, 21, 59, 93, 116, 120, 122, 137, 138, 139, 140, 141, 142, 145, 158, 206, 210, 263, 312
!avrf, 109
!chkimg, 63, 184, 185, 186
!cmkd, 348
!cs, 157, 159, 192, 193, 298
!devobj, 177
!devstack, 177
!dh, 254, 408
!dpcs, 176
!exchain, 126
!fileobj, 29, 198
!for_each_thread, 30
!gflag, 109, 110
!heap, 316, 321, 332
!irp, 25, 29, 177, 198
!irpfind, 169, 198
!lmi, 115
!locks, 43, 48, 135
!lpc, 133, 189, 191
!pool, 121, 123, 204
!poolused, 202
!process, 28, 34, 53, 71, 73, 188, 197
!pte, 348
!ready, 35, 82, 85
!runaway, 88, 99, 104, 105, 180
!running, 34, 35, 47, 82, 85, 170, 202
!stacks, 48, 169
!sysinfo, 179
!teb, 39, 76, 152, 154, 165, 181, 207, 327, 338
!thread, 30, 36, 48, 56, 66, 80, 82, 83, 85, 88, 135, 169, 170, 190, 192, 197, 202, 203, 348, 349, 416, 417
!verifier, 108
!vm, 201
!wow64exts, 39, 40

$

$$, 365, 372, 412

.

.asm, 103, 328, 331
.cxr, 116, 119, 120, 122, 167, 210, 330, 331
.ecxr, 118, 331, 338
.effmach, 30, 40
.exptr, 94, 150, 314
.exr, 58, 116, 120, 122, 150, 158, 206, 312
.formats, 75, 123
.frame, 332, 411, 412
.imgscan, 408, 409
.load, 30, 39, 40
.opendump, 208, 210
.process, 30, 31, 71, 73, 192
.reload, 30, 72, 326
.symfix, 326
.thread, 30, 84, 119, 193, 194, 195, 196, 203, 330, 331

~

~*e, 40
~*kn, 411
~~, 155, 298
~0s, 327

D

da, 165, 264, 405

db, 121, 123, 264, 408
dc, 154, 155, 369
dd, 40, 91, 118, 332
dds, 26, 40, 57, 64, 66, 153, 154, 166
dp, 347, 407
dps, 76, 89, 170, 181, 204, 207, 318, 324, 327, 333, 405
dpu, 412
dqs, 322, 338
dt, 84, 332, 333
du, 22, 27, 81, 196, 264, 405
dv, 332

G

g, 28, 193, 209, 210, 248, 335, 405

K

k, 91, 152, 159, 316, 321, 326, 327, 330, 331, 408
kb, 30, 411, 412
kL, 55, 74, 78, 79, 110, 111, 145, 156, 164, 167, 180, 206, 209, 211, 262, 297, 313, 318, 401, 402, 406
kv, 81, 84, 93, 118, 119, 126, 149, 157, 165, 195, 196, 203, 210, 314

L

lm, 22, 27, 205, 363
lmt, 73, 113, 168
lmu, 72
lmv, 113, 114, 125, 128, 157, 183, 254

R

r, 30, 31, 40, 73, 79, 155, 192, 203, 207

U

u, 22, 23, 27, 58, 65, 103, 138, 141, 143, 178, 179, 183, 185, 186, 187, 247, 332, 335, 407
ub, 27, 55, 56, 69, 70, 77, 80, 104, 121, 139, 143, 176, 178, 182, 183, 184, 187, 324, 329, 331, 332, 404, 407
uf, 138, 139, 143

V

version, 80

# Memory Dump Analysis Anthology

## Volume 6

**Dmitry Vostokov**
**Software Diagnostics Institute**

OpenTask

Published by OpenTask, Republic of Ireland

Copyright © 2013 by Dmitry Vostokov

Copyright © 2015 by Software Diagnostics Institute

All rights reserved. No part of this book may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, without the prior written permission of the publisher.

You must not circulate this book in any other binding or cover, and you must impose the same condition on any acquirer.

OpenTask books are available through booksellers and distributors worldwide. For further information or comments send requests to press@opentask.com.

Product and company names mentioned in this book may be trademarks of their owners.

A CIP catalog record for this book is available from the British Library.

ISBN-13: 978-1-908043-19-1 (Paperback)

ISBN-13: 978-1-908043-20-7 (Hardback)

First printing, 2013

Revision 2 (July 2015)

# Contents

Preface ........................................................................................................ 15

Acknowledgements........................................................................................ 17

PART 1: Professional Crash Dump Analysis and Debugging ......................................... 19

Memory Dump Analysis Best Practices ...................................................................... 19

Windows Debugging Expert System WinDbg Extension ........................................... 20

Common Mistakes .................................................................................................. 21

Not Comparing to Reference Debugger Output ....................................................... 21

From Bugchecks to Patterns .................................................................................... 23

Raw Stack from Laterally Damaged Memory Dumps ............................................... 24

WinDbg Tips and Tricks: Getting the Bottom of a Stack Trace ................................. 26

PART 2: Crash Dump Analysis Patterns ..................................................................... 31

Divide by Zero (Kernel Mode) ................................................................................... 31

Fat Process Dump .................................................................................................... 33

Blocked Queue ......................................................................................................... 34

Crash Signature ........................................................................................................ 37

Invalid Parameter (Process Heap) ............................................................................ 40

Hooking Level ........................................................................................................... 43

Embedded Comments ............................................................................................... 47

Well-Tested Module .................................................................................................. 48

String Parameter ........ 49

Environment Hint ........ 51

Dual Stack Trace ........ 52

Blocking Module ........ 54

Wait Chain (Window Messaging) ........ 55

Wait Chain (Named Pipes) ........ 60

Top Module ........ 62

Dialog Box ........ 63

Technology-Specific Subtrace (COM Interface Invocation) ........ 67

Livelock ........ 70

Semantic Structure (PID.TID) ........ 73

Instrumentation Side Effect ........ 77

Directing Module ........ 80

Stack Overflow (Software Implementation) ........ 82

Data Correlation ........ 84

Truncated Stack Trace ........ 86

Least Common Frame ........ 87

Self-Diagnosis (Kernel Mode) ........ 89

Technology-Specific Subtrace (Dynamic Memory) ........ 90

Module Hint ........ 92

Custom Exception Handler (Kernel Space) ........ 94

No Data Types ........ 96

Cloud Environment ........ 97

Version-Specific Extension ........ 99

Multiple Exceptions (Managed Space) ........ 104

Blocking File ........ 105

Quiet Dump ........ 109

Pleiades ........ 110

Thread Age ........ 111

Unsynchronized Dumps ........ 113

Coupled Modules ........ 114

Managed Stack Trace ........ 115

Problem Vocabulary ........ 116

Activation Context ........ 117

Stack Trace Set ........ 120

Special Thread (.NET CLR) ........ 123

Dynamic Memory Corruption (Managed Heap) ........ 124

Stack Trace Collection (Managed Space) ........ 127

Duplicate Extension ........ 131

Deadlock (Managed Space) ........ 135

Caller-n-Callee ... 138

Handled Exception (User Space) ... 141

Handled Exception (.NET CLR) ... 144

Execution Residue (Managed Space) ... 149

Annotated Disassembly (JIT .NET code) ... 151

Wait Chain (Mutex Objects) ... 153

Inline Function Optimization (Managed Code) ... 155

Technology-Specific Subtrace (JIT .NET Code) ... 157

Double IRP Completion ... 160

PART 3: Pattern Interaction ... 163

Main Thread, Self-Diagnosis, Window Message Chain, Blocking Module, Ubiquitous Component, Dual Stack Trace, Pipe Wait Chain and Coupled Machines ... 163

Abridged Dump, Embedded Comment, Spiking Thread, Incorrect Stack Trace and Top Module ... 166

Stack Trace Collection, Message Box, Self-Diagnosis, Version-Specific Extension, Managed Stack Trace and Managed Code Exception ... 168

PART 4: Unified and Generative Debugging ... 171

A Periodic Table of Software Defects ... 171

Analysis, Architectural, Design, Implementation and Usage Debugging Patterns ... 172

Generative Debugging ... 173

Metadefect Template Library ... 174

PART 5: A Bit of Science and Philosophy ... 175

On Memory Perspectives ........ 175

Orbifold Memory Space ........ 176

Notes on Memoidealism ........ 177

M->analysis ........ 178

Memiosphere ........ 179

On Memory-Time vs. Space-Time ........ 180

The Will to Be Memorized ........ 181

The Trinity of Memory Worldview ........ 182

Uses of Memoretics ........ 183

Crossdisciplinary Memoretics as Interdisciplinary Science ........ 184

Private Property on Memory Spaces ........ 185

Coarse vs. Fine Grained DNA of Software Behavior ........ 187

PART 6: Fun with Crash Dumps ........ 189

Music for Debugging ........ 189

555 Binary Threads ........ 189

Out of Memory and Losing My Data (Comment Impact) ........ 190

Navigating the Long List ........ 191

Debugging Joke ........ 192

Memory Dump Barcodes ........ 193

MessageBox at Dublin Zoo ........ 194

CDB for Kids ..... 195

Snow Spike Residue ..... 196

Second Snowfall Spike in Dublin ..... 197

MMXI ..... 198

Happy New Year and Decade of Debugging 0×7DB - 0×7E4! ..... 199

Do Security Professionals Dream? ..... 204

Debugging Slang ..... 205

- Golden Bug ..... 205
- Beer Time ..... 206
- Finger Exercise ..... 207
- Resolution Rush ..... 208
- The Window of Opportunity ..... 209
- Dump ..... 210
- Pre-analysis ..... 211
- Tapping ..... 212
- Having Fun ..... 213
- Adult Debugging ..... 214
- Second Eye ..... 215
- Abscess ..... 216
- Finction ..... 217

Mad OS and other Publishing Blunders ........ 218
The Ultimate Debugger’s Desk ........ 221
Memceptions: Flags and Handles are Everywhere! ........ 222
Computer Memory Monsters ........ 223
On President’s Daily Briefs (PDBs) ........ 226
The First Evidence for Process Resurrection ........ 227
Vacuum Pages ........ 228
WinDbg Command on Certificate ........ 230
Pleasing WinDbg SOS Extension ........ 231
Airport Terminal Services Incident ........ 232
Philosophical Self-Interview ........ 233
PART 7: A Bit of Religion ........ 235
Memory Creates God ........ 235
Morality and Memorianity ........ 236
On Natural Theology ........ 237
PART 8: Software Trace Analysis ........ 239
Pattern Interaction ........ 239
Basic Facts, Periodic Error, and Defamiliarizing Effect ........ 239
Close and Deconstructive Readings of a Software Trace ........ 240
Software Tracing Best Practices ........ 241

No Longer Seeing Nothing: The Advantage of Patterns ........................................ 242

PART 9: Software Trace Analysis Patterns ........................................ 243

Focus of Tracing ........................................ 243

Event Sequence Order ........................................ 244

Implementation Discourse ........................................ 245

News Value ........................................ 246

Master Trace ........................................ 247

Gossip ........................................ 248

Impossible Trace ........................................ 249

Glued Activity ........................................ 250

Message Invariant ........................................ 251

UI Message ........................................ 252

Original Message ........................................ 253

PART 10: Software Troubleshooting and Debugging ........................................ 255

Debugware Patterns ........................................ 255

System Description Snapshot ........................................ 255

Debugging in 2021: Trends for the Next Decade ........................................ 256

The Way of Philip Marlowe: Abductive Reasoning for Troubleshooting and Debugging ........................................ 257

Workaround Patterns ........................................ 258

Fake API ........................................ 258

User Interface Problem Analysis Patterns ... 259

Message Box ... 259

PART 11: Software Victimology ... 263

Function Activity Theory ... 263

PART 12: Art ... 265

No E-numbers Software Product Sticker ... 265

Paleo-debugging: Excavated Minidump ... 266

Stack Trace Art ... 267

Debugger’s Dream ... 268

Defect in Defect ... 269

Memorianity Cross ... 270

Memioart: The New Art Form ... 271

Clouded ... 272

Cloud Traces ... 273

What Is To Be Done? ... 274

PART 13: Miscellaneous ... 277

GI Index of Memory Dump Analysis ... 277

The New School of Debugging ... 279

TestWER Tool to Test Windows Error Reporting ... 280

Moving to ARM ... 283

The New School of Debugging: What's New ........ 284

A.C.P. Root Cause Analysis Methodology ........ 285

TestWAER Tool to Test Windows Azure Error Reporting ........ 286

PART 14: Intelligence Analysis ........ 289

Intelligence Analysis Patterns ........ 289

The Birth of Memory Intelligence Agency ........ 290

Appendix ........ 291

Memory Analysis as a Service ........ 291

Stack Overflow Patterns ........ 292

.NET / CLR / Managed Space Patterns ........ 293

Stack Trace Patterns ........ 294

Symbol Patterns ........ 295

Analysis Compass ........ 296

Software Trace Analysis Checklist ........ 297

Crash Dump Analysis Checklist ........ 298

Index of WinDbg Commands ........ 301

About the Author ........ 304

Cover Images ........ 305

# Index of WinDbg Commands

!address, 38
!alpc, 35, 75, 303
!analyze, 303
!analyze -v, 31, 32, 37, 41, 82, 94, 99, 118, 133, 161, 282, 301
!analyze -v -hang, 301
!apc, 303
!bugdump, 303
!chkimg, 43, 44, 301
!CLRStack, 101, 102, 116, 128, 133, 134, 135, 152, 159, 305
!cs, 301, 303
!dh, 301
!dlk, 136, 138
!dlls, 301
!dpcs, 303
!DumpHeap, 127, 305
!DumpObj, 127, 151
!DumpStack, 116, 139, 140, 141, 149, 150, 170
!DumpStackObjects, 150, 305
!envvar, 301
!error, 109
!exchain, 95, 301
!exqueue, 303
!filecache, 303
!fileobj, 61, 107, 303
!gflag, 40, 77, 301
!handle, 108
!help, 20
!IP2MD, 153, 157, 158, 159
!irp, 60, 83, 107, 162, 303
!irpfind, 303
!locks, 70, 212, 301, 303
!lpc, 303
!ntsdexts.locks, 303
!pe, 102, 105, 116, 170, 305
!peb, 51, 301
!pload, 20
!pool, 91, 93, 303
!poolused, 115, 303
!PrintException, 102, 104, 105
!process, 58, 299, 303
!qlocks, 303
!ready, 303

!runaway, 78, 112, 301
!running, 70, 303
!search, 200
!session, 303
!sprocess, 303
!stacks, 303
!syncblk, 136, 305
!sysinfo, 301, 303
!teb, 24, 142
!thread, 52, 73
!Threads, 105, 124, 305
!U, 152, 157
!uniqstack, 121, 122, 301
!VerifyHeap, 126, 127, 305
!vm, 229, 303
.asm, 152, 157
.chain, 100, 101, 132, 133, 134, 305
.cordll, 99, 101, 102
.cxr, 32, 38, 84, 88, 108, 118
.dump, 232
.enumtag, 303
.formats, 85
.kframes, 28, 301
.load, 20, 101, 132, 232
.process, 108
.reload, 108
.symfix, 301
.thread, 84, 88, 108
.time, 113, 114
.ttime, 112
.unload, 100, 135
?, 84
~, 167
~*e, 105, 128, 305
~*kbL, 55, 56
da, 49, 50
dc, 63, 64, 65, 93
dd, 120, 152
dp, 137, 138
dpp, 74
dps, 25, 140, 143
dpu, 49, 50
dS, 303
dt, 96, 119, 120
du, 49, 169, 262

g, 283
-hang, 303
k, 282
k L=, 29
kc, 26, 27, 77, 112, 121
kL, 24, 37, 40, 41, 54, 62, 68, 82, 84, 94, 125, 156, 158, 164, 167, 169, 262
kv, 31, 37, 42, 49, 56, 63, 64, 84, 108, 121, 301
lm, 97, 111, 224
lmv, 62, 97, 98, 100, 132, 301, 305
ln, 303
r, 87, 157
s-d, 200
u, 44, 46
ub, 42, 62, 83, 137, 138, 140, 143, 262
version, 228

# Memory Dump Analysis Anthology

## Volume 7

**Dmitry Vostokov**
**Software Diagnostics Institute**

OpenTask

Published by OpenTask, Republic of Ireland

Copyright © 2014 by Dmitry Vostokov

Copyright © 2014 by Software Diagnostics Institute

All rights reserved. No part of this book may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, without the prior written permission of the publisher.

You must not circulate this book in any other binding or cover, and you must impose the same condition on any acquirer.

OpenTask books are available through booksellers and distributors worldwide. For further information or comments send requests to press@opentask.com.

Product and company names mentioned in this book may be trademarks of their owners.

A CIP catalog record for this book is available from the British Library.

ISBN-13: 978-1-908043-51-1 (Paperback)

ISBN-13: 978-1-908043-52-8 (Hardback)

First printing, 2014

Revision 2 (July 2015)

## Contents

Preface ........ 23

Acknowledgements ........ 25

PART 1: Professional Crash Dump Analysis and Debugging ........ 27

WinDbg Shortcuts ........ 27

.ecxr ........ 27

!heap -x -v ........ 29

!sw and !k ........ 31

Two WinDbg Scripts That Changed the World ........ 32

Raw Stack Dump of All Threads (Kernel Space) ........ 37

The Design of Memory Dump Analysis: 7 Steps of Highly Successful Analysts ........ 38

Postmortem Effects of -g ........ 39

Event Owners ........ 42

Improbable Occurrence ........ 48

Pattern Cooperation ........ 49

Page Heap Implementation ........ 54

More Common Mistakes in Memory Analysis ........ 60

Memory Dump Analysis Best Practices ........ 63

PART 2: Crash Dump Analysis Patterns ........ 65

FPU Exception ........ 65

Hidden Parameter ........ 67

Memory Leak (Page Tables) ........ 69

Unrecognizable Symbolic Information ........ 76

Network Packet Buildup ........ 82

Disconnected Network Adapter ........ 83

Problem Module ........ 85

Empty Stack Trace ........ 86

Debugger Bug ........ 90

Value References ........ 92

Self-Diagnosis (Registry) ........ 93

System Object ........ 95

Module Variable ........ 98

Stack Trace Collection (Predicate) ........ 100

Stack Trace Collection (I/O Requests) ........ 101

Regular Data ........ 106

Translated Exception ........ 107

Blocked DPC ........ 108

Late Crash Dump ........ 109

Blocked Thread (Timeout) ........ 110

Punctuated Memory Leak ........ 111

Insufficient Memory (Reserved Virtual Memory) .................................................. 114

Coincidental Error Code .................................................. 117

Stored Exception .................................................. 119

Activity Resonance .................................................. 120

Value Adding Process .................................................. 122

Memory Leak (I/O Completion Packets) .................................................. 123

No Current Thread .................................................. 124

Unloaded Module .................................................. 126

Stack Trace Change .................................................. 131

Spike Interval .................................................. 132

Deviant Module .................................................. 133

Hidden Exception (Kernel Space) .................................................. 140

Handled Exception (Kernel Space) .................................................. 141

High Contention (.NET CLR Monitors) .................................................. 142

Frozen Process .................................................. 145

Incomplete Session .................................................. 150

Error Reporting Fault .................................................. 152

First Fault Stack Trace .................................................. 155

Hidden Process .................................................. 156

Disk Packet Buildup .................................................. 158

Deviant Token ........ 161

Module Collection ........ 162

Handle Leak ........ 164

Critical Stack Trace ........ 165

Debugger Omission ........ 166

Broken Link ........ 168

Wait Chain (Pushlocks) ........ 170

Insufficient Memory (Session Pool) ........ 172

Step Dumps ........ 173

Reduced Symbolic Information ........ 174

Injected Symbols ........ 175

Glued Stack Trace ........ 178

Distributed Wait Chain ........ 182

Ubiquitous Component (Kernel Space) ........ 184

One-Thread Process ........ 187

Module Product Process ........ 189

Crash Signature Invariant ........ 190

Small Values ........ 191

Shared Structure ........ 193

Wait Chain (CLR Monitors) ........ 194

Thread Cluster ........ 195

Module Collection (Predicate) ........ 196

False Effective Address ........ 197

Screwbolt Wait Chain ........ 198

PART 3: Core Dump Analysis Patterns (Mac OS X) ........ 201

GDB for WinDbg Users ........ 201

Stack Trace ........ 203

GDB Annoyances: Incomplete Stack Trace ........ 205

NULL Pointer (Data) ........ 206

Shared Buffer Overwrite ........ 207

Multiple Exceptions ........ 211

Double Free (Process Heap) ........ 213

Dynamic Memory Corruption (Process Heap) ........ 214

Spiking Thread ........ 216

NULL Pointer (Code) ........ 218

Execution Residue ........ 220

Coincidental Symbolic Information ........ 223

Paratext ........ 225

Truncated Dump ........ 227

C++ Exception ........ 228

Local Buffer Overflow ........ 229

Divide by Zero (User Mode) ........ 231

Stack Overflow (User Mode) ........ 232

Active Thread ........ 236

PART 4: Malware Analysis Patterns ........ 239

Malware: A Definition ........ 239

Fake Module ........ 240

RIP Stack Trace ........ 244

Driver Device Collection ........ 246

Pre-Obfuscation Residue ........ 247

Packed Code ........ 248

Raw Pointer ........ 251

Out-of-Module Pointer ........ 252

Patched Code ........ 253

String Hint ........ 254

Namespace ........ 257

PART 5: A Bit of Science and Philosophy ........ 259

On Matter ........ 259

Commodities as Memories ........ 260

Software as Means of Production ........ 261

Notes on Memoidealism ........ 262

The Confluence of Computers, Philosophy, and Religion ........ 264

Analytic Memory Dump - A Mathematical Definition ........ 265

Sorting and Early Greek Philosophers ........ 266

General Abnormal Patterns of Structure and Behavior ........ 267

On Matter and Substances ........ 268

M-Memory ........ 269

Ontology of Memoidealism ........ 270

Philosophies of Persistence ........ 273

Information as Arrow ........ 275

Dialectical Triad in Memoidealism ........ 276

PART 6: Software Trace Analysis Patterns ........ 279

Software Trace Diagrams (STDiagrams) ........ 279

Macrofunction ........ 283

Linked Messages ........ 284

Marked Message ........ 285

Trace Frames ........ 286

Counter Value ........ 288

Message Context ........ 289

Error Distribution ........ 290

Break-in Activity ........ 291

Resume Activity ........ 292

Fiber Bundle ........ 294

Data Flow ........ 296

Empty Trace ........ 298

Error Message ........ 299

Periodic Message Block ........ 300

Visibility Limit ........ 301

Relative Density ........ 302

Sparse Trace ........ 303

Opposition Messages ........ 304

Split Trace ........ 305

Message Interleave ........ 306

Sheaf of Activities ........ 307

Indexical Trace ........ 310

Abnormal Value ........ 311

Dominant Event Sequence ........ 313

Pivot Message ........ 314

Traces of Individuality ........ 318

Indirect Facts ........ 319

Hidden Error ........ 320

Last Activity ........ 322

State and Event ........ 324

Dialogue ........ 326

Motif ........ 329

Exception Stack Trace (Java) ........ 330

Correlated Discontinuity ........ 332

Piecewise Activity ........ 333

Density Distribution ........ 335

Factor Group ........ 336

Silent Messages ........ 339

Shared Point ........ 341

Meta Trace ........ 343

Data Association ........ 344

State Dump ........ 346

Message Cover ........ 347

Message Set ........ 349

Error Thread ........ 351

Activity Divergence ........ 352

PART 7: Fun with Crash Dumps ........ 355

Debugging Slang ........ 355

LoL ........ 355

Watching a Movie ........ 356

PonOS ........ 357

Typology, Typological ........ 358

Memorandum ........ 359

HELL ........ 360

FBI ........ 361

poo ........ 362

STaMPs ........ 363

A NoSQL Problem ........ 364

Matrix ........ 365

Fool ........ 366

B2B, B2C, H2H ........ 367

New Year Eve Debugging ........ 368

Happy New Spiking Year of Software Trace Analysis ........ 369

Happy New Year (from Windows 8) ........ 370

Music for Debugging ........ 372

Going Romantic ........ 372

Make It through This Trace ........ 373

Fiction for Debugging ............................................................................................ 374

The Problem and The Solution ............................................................................ 374

Pilgrimage to Harvard University .......................................................................... 375

Welcome to Ki* and Ke* ........................................................................................ 376

I Memory Dump ..................................................................................................... 377

A Blue Screen Watch .............................................................................................. 379

Poetry ...................................................................................................................... 380

Surfaces in Nature ................................................................................................... 381

PART 8: Software Narratology ................................................................................ 383

Software Anti-Narrative ........................................................................................... 383

Software Narratology Helps Fiction Writers ........................................................... 384

Narremes in Software Narratology ......................................................................... 386

Narralog - A Software Trace Modeling Language .................................................... 387

What is a Software Narrative? ................................................................................. 388

Software Narrative Planes ....................................................................................... 389

Software Narratology Square ................................................................................... 391

Writing and Validation of Historical Narratives ...................................................... 392

Software Trace Analysis Patterns Domain Hierarchy ............................................... 393

Process Monitor as Modeling Tool ........................................................................... 394

Generalized Software Narrative and Trace .............................................................. 395

Unified Computer Diagnostics: Incorporating Hardware Narratology .................... 396

Introducing Software Narratology of Things (Software NT) .................................... 397

PART 9: Software Diagnostics, Troubleshooting, and Debugging ................................ 399

Unified and Generative Debugging ........................................................................ 399

Analysis, Architectural, Design, Implementation and Usage Debugging Patterns 399

Software Problem Description Language ................................................................ 401

What are Software Trace and Memory Dump Analysis? A One Sentence Definition402

Software Problem Solving Tools as a Service ......................................................... 403

Software Problem Description Patterns ................................................................. 404

Software Behavior Pattern Prediction .................................................................... 405

Patterns of Software Diagnostics ........................................................................... 406

First Fault ................................................................................................... 406

Highly Effective Diagnostics ................................................................................. 407

Network Trace Analysis Patterns ........................................................................... 408

Software Diagnostics Services ............................................................................... 411

Architecture of Process Memory Dump Capture Done Right .................................. 412

An Introduction to General Systems Thinking (Book Review) ................................ 413

Software Diagnostics Institute Logo ....................................................................... 414

User Interface Problem Analysis Patterns ............................................................... 415

Unresponsive Window ............................................................................... 415

Pattern-Based Software Diagnostics ........ 418

Software Diagnostics Discipline ........ 419

Architecture of memCPU ........ 420

Phenomenology of Software Diagnostics: A First Sketch ........ 421

Software Diagnostics Report Schemes ........ 422

Missing Cause Trace ........ 422

Software Diagnostics Training: Two Approaches ........ 423

Software Disruption Patterns ........ 425

Space Precondition ........ 425

Static Code Analysis Patterns ........ 426

Loop Construct ........ 426

The Structure of Software Problem Solving Organization ........ 427

Bridging the Great Divide ........ 428

Elementary Software Diagnostics Patterns ........ 429

Zero Fault Software Diagnostics ........ 430

Agile Software Diagnostics ........ 432

ADDR Pattern Catalogue ........ 433

Thinking-Based Software Diagnostics ........ 434

Memory Acquisition Pattern Catalog ........ 436

Trace Acquisition Pattern Catalog ........ 437

Patterns of Software Diagnostics Architecture ..... 438

Detecting and Predicting the Unknown ..... 440

Software Diagnostics Metaphors ..... 442

Software Diagnostics as Psychology ..... 442

Software Diagnostics as Literary Criticism ..... 443

Rapid Software Diagnostics Process (RSDP) ..... 444

Right First Time Software Diagnosis ..... 445

Software Diagnosis Codes ..... 446

Vulnerability Analysis Patterns (VAP) ..... 447

Versioned Namespace ..... 449

PART 10: Art and Visualization ..... 451

2012 (Pessimistic) ..... 451

2012 (Optimistic) ..... 452

A Bug in a Bag (Collections, Ex-hi-bit 1) ..... 453

A Bug Meets a Bug (The Clash of Civilizations) ..... 454

A Bug Catcher ..... 455

The Second Generation of CARE System (Trademark) ..... 456

RawStackGram ..... 457

A Memory Window ..... 458

Liquid Memory ..... 459

Computer Brain ........ 460

Computer Evolution ........ 461

M Spaces ........ 462

Happy Hellowin! ........ 463

Pointers in Nature ........ 464

Drink Sensibly Before The End Of The World! ........ 465

MM=DD=YY ........ 466

Process Monitor Log Visualized ........ 468

Holes Infinity (HI OS) ........ 472

Cyber Vostok Missions ........ 473

A Dump Machine ........ 474

The Power of Simplicity ........ 475

Happy St. Patrick’s Screen ........ 476

Happy New Year 2014! ........ 477

I Love Software Diagnostics ........ 478

Puree Windows Cooking ........ 479

- Salad Winterminal ........ 479

- Kernel Soup ........ 481

- Neolithic Soup ........ 482

- Food Subsystems ........ 483

An Accident of Creation ........ 484

So Chi Salad, 2014 ........ 485

Self-Organized Window-ed soup ........ 486

Political Computicarts ........ 487

Needs Non-Invasive Debugging! ........ 487

Russian Spaces ........ 488

The Day I Quit ........ 489

Hero of Dump Analysis, a Medal for Labor Day ........ 490

Diagnosed by Vostokov®™ ........ 491

Stack Trace Shapes ........ 492

The Art of Internals ........ 494

Threadinking ........ 495

PART 11: Miscellaneous ........ 497

C and C++ Programming Books That Made a Great Impression on the Author........ 497

Outside ........ 499

After Debugging ........ 500

Crash Dumps, Acquisitions, and Layoffs ........ 501

Cadaver Worm: An Exercise in Malware Fiction ........ 502

WinDbg as UNICODE to ASCII Converter ........ 504

Appendix ........ 505

Falsity and Coincidence Patterns ........ 505

Process Patterns ........ 506

Thread Patterns ........ 507

Optimization Patterns ........ 508

Exception Patterns ........ 509

Module Patterns ........ 510

RPC, LPC and ALPC Patterns and Case Studies ........ 511

ERESOURCE Patterns and Case Studies ........ 513

Meta-Memory Dump Patterns ........ 515

Crash Dump Analysis Checklist ........ 516

Index of WinDbg Commands ........ 519

About the Author ........ 521

Notes ........ 522

Cover Images ........ 523

# Index of WinDbg Commands

!address, 54, 111, 112, 113, 115, 117, 118, 125, 133, 137, 202, 377, 378
!alpc, 51, 511, 517
!analyze, 27, 38, 167, 516, 517
!bugdump, 518
!chkimg, 240, 253, 516
!CLRStack, 518
!cs, 52, 183, 516, 517
!dd, 69, 70
!dh, 133, 137, 138, 166, 243, 248, 516
!dlls, 516
!dpcs, 108, 517
!DumpHeap, 518
!DumpRuntimeTypes, 518
!DumpStackObjects, 518
!eeheap, 518
!envvar, 516
!error, 66
!exchain, 516
!exqueue, 517
!filecache, 517
!fileobj, 517
!FinalizeQueue, 518
!for_each_module, 92, 240, 242
!for_each_process, 85
!for_each_thread, 32, 34, 37, 163, 168
!GCHandleLeaks, 518
!GCHandles, 518
!gflag, 54, 516
!handle, 191, 192
!heap, 29, 115, 117, 516
!irp, 44, 45, 517
!irpfind, 45, 101, 517
!k, 31
!lmi, 77, 78, 117, 118, 189, 243
!locks, 38, 513, 516, 517
!lpc, 511, 517
!ndiskd.miniport, 83
!ndiskd.miniports, 83, 517
!ndiskd.pktpools, 82, 517
!object, 95, 96, 246
!pe, 518
!peb, 242, 516
!pool, 517
!poolfind, 123, 156, 168
!poolused, 123, 164, 172, 517
!process, 38, 50, 69, 77, 85, 97, 123, 146, 152, 153, 154, 156, 164, 168, 169, 187, 356, 517
!pte, 74, 75
!ptov, 70, 71, 72
!qlocks, 517
!ready, 517
!reg, 93
!runaway, 32, 38, 132, 142, 237, 416, 516
!running, 38, 120, 517
!scsikd.classext, 158, 159, 517
!session, 49, 150, 172, 517
!sprocess, 49, 122, 145, 150, 151, 517
!stacks, 184, 517
!sw, 31
!syncblk, 144, 518
!sysinfo, 516, 518
!teb, 86, 107
!thread, 32, 34, 37, 51, 108, 120, 121, 140
!Threads, 518
!token, 161
!uniqstack, 516
!VerifyHeap, 518
!vm, 85, 90, 96, 98, 99, 168, 172, 517
.chain, 518
.cxr, 27, 52, 53, 65, 67, 79, 80, 81, 128, 129, 140, 153, 197, 240
.echo, 32
.ecxr, 27, 28, 119
.effmach, 31, 153, 163
.enumtag, 518
.exr, 66, 119, 125
.imgscan, 133, 166
.kframes, 516

.load, 31, 153, 158, 162, 163
.process, 52, 74, 76, 85, 153, 182
.reload, 80, 114, 162, 163, 176, 202
.symfix, 114, 516
.sympath+, 80, 176
.thread, 27, 37, 52, 53, 67, 79, 80, 81, 129, 153, 162, 163, 183, 240
~, 28, 38, 60, 109, 124, 126, 142, 191, 202, 204, 516, 518
~*k, 28, 109, 126, 142, 202
~*kv, 38, 516
Checklist, 516
d*, 38
dc, 56, 168, 174
dd, 69, 117, 129
dp, 29, 98, 118, 129, 191, 377
dps, 37, 65, 86, 107, 138, 140, 144
dpu, 68, 417
dS, 99, 516, 517, 518
dt, 43, 60, 61, 62, 93, 95, 145, 146, 158, 174, 177
du, 504
eb, 504
g, 39, 40, 41, 115, 183, 202, 213, 385
k, 52, 53, 59, 89, 115, 124, 144, 153, 180, 202, 240, 244, 415, 416
kc, 190
kL, 27, 39, 40, 67, 79, 80, 81, 86
kv, 60, 128, 129, 174, 176, 191, 202
kvL, 110
lm, 85, 175, 176, 252
lmp, 240
lmu, 85, 162
lmv, 38, 77, 78, 79, 117, 118, 127, 133, 139, 162, 189, 242, 417, 516, 517, 518
ln, 117, 118, 130, 517
rMF, 66
u, 48, 98, 117, 143, 154, 202, 253
ub, 67, 129, 141, 193, 201, 202

# Memory Dump Analysis Anthology

## Volume 8a

**Dmitry Vostokov**
**Software Diagnostics Institute**

OpenTask

Published by OpenTask, Republic of Ireland

Copyright © 2014 by Dmitry Vostokov

Copyright © 2014 by Software Diagnostics Institute

All rights reserved. No part of this book may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, without the prior written permission of the publisher.

You must not circulate this book in any other binding or cover, and you must impose the same condition on any acquirer.

OpenTask books are available through booksellers and distributors worldwide. For further information or comments send requests to press@opentask.com.

Product and company names mentioned in this book may be trademarks of their owners.

A CIP catalog record for this book is available from the British Library.

ISBN-13: 978-1-908043-53-5 (Paperback)

First printing, 2014

Revision 2.0 (July 2015)

# Table of Contents

**Preface ........................................................................................ 7**

**About the Author ........................................................................ 9**

**PART 1: Professional Crash Dump Analysis and Debugging ............................... 11**

Software Diagnostics Professional Certification ............................................. 11
Three Roads to Kernel Space ............................................................ 13

**PART 2: Crash Dump Analysis Patterns .................................................. 15**

Design Value ................................................................................ 15
Hidden IRP.................................................................................... 16
Tampered Dump ............................................................................ 17
Wait Chain (RTL_RESOURCE) ........................................................... 29
Memory Fluctuation (Process Heap) ................................................... 35
Last Object ................................................................................... 37
Rough Stack Trace.......................................................................... 39
Past Stack Trace ............................................................................ 43
Stack Trace (I/O Request) ................................................................ 46
Stack Trace (File System Filters)....................................................... 48
Stack Trace (Database).................................................................... 51
Wait Chain (Modules) ..................................................................... 56
Insufficient Memory (Stack Trace Database) ........................................ 57
Insufficient Memory (Region) ........................................................... 63
Memory Leak (Regions) .................................................................. 65
Invalid Handle (Managed Space) ....................................................... 69
Ghost Thread ................................................................................ 77
Dry Weight.................................................................................... 79
Exception Module.......................................................................... 80

**PART 3: Memory Forensics ............................................................... 83**

Memory Forensics Professional Certification ........................................ 83
Native Memory Forensics ................................................................ 84

**PART 4: A Bit of Science and Philosophy ........................................................ 85**

Memory Symmetry Breaking ........................................................ 85
Memoevolutionism ........................................................ 86
Entropy as Memory and Memory as Entropy ........................................................ 87
Notes on Memoidealism ........................................................ 88
Welcome to Memorianism ........................................................ 89
United Memory Lands, Memorianites, EthnOS ........................................................ 90
Quotes from Memoriarch ........................................................ 91
Pattern-Oriented Philosophy ........................................................ 92

**PART 5: Software Trace Analysis Patterns ........................................................ 93**

Hidden Facts ........................................................ 93
Back Trace ........................................................ 95
Blackout ........................................................ 97
Missing Message ........................................................ 99
Use Case Trail ........................................................ 101
Event Sequence Phase ........................................................ 103
Milestones ........................................................ 105
File Size ........................................................ 107
Singleton Event ........................................................ 108
Visitor Trace ........................................................ 110

**PART 6: Fun with Crash Dumps ........................................................ 111**

Debugging Slang and Proverbs ........................................................ 111

*PUS ........................................................ 111*
*Coollect ........................................................ 111*
*Dump-out ........................................................ 111*
*LOGIC ........................................................ 111*
*DiagNose ........................................................ 112*
*Consolidation ........................................................ 112*
*No Pass a Run! ........................................................ 112*
*ID IoT Zone ........................................................ 112*
*Putty in Someone's Hands ........................................................ 112*
*DisPatched vs. DESPatched ........................................................ 112*
*Programmatica Nervosa ........................................................ 113*

*GOTCHA* ........ *113*
*Pan-o-RAM-ic* ........ *113*
*VLSI* ........ *113*
*Debugging Proverb* ........ *113*

Space Opera ........ 114
If Programmers Were Writers ........ 115
My Computer Celebrates Halloween ........ 116
Look, there's a Bug! ........ 117
Diagnostics in Science Fiction ........ 118
Hard Copy Natives ........ 119

**PART 7: Software Narratology ........ 121**

Malnarratives ........ 121
Higher-Order Pattern Narratives (Analyzing Diagnostic Analysis) ........ 123

**PART 8: Software Diagnostics, Troubleshooting, and Debugging ........ 127**

A Pattern Language for Performance Analysis ........ 127
The Timeless Way of Diagnostics ........ 128
Pattern-Oriented Debugging Process ........ 130

**PART 9: Art and Visualization ........ 133**

Café WoW ........ 133
Bang Debugging ........ 134
Bug Hunter ........ 135
Glass of Water Dump ........ 136
Memory Dump Analysis ........ 137
Organic Incidents and Bad Stench ........ 138

**PART 10: Miscellaneous ........ 139**

Book Discovery ........ 139
Quotes ........ 140

**Appendix ........ 143**

Crash Dump Analysis Checklist ........................................................... 143

**Index of WinDbg Commands ........................................................... 147**

**Notes........................................................................................... 149**

# Index of WinDbg Commands

!address, 52, 58, 63, 64, 65, 66, 67, 68, 79
!alpc, 15, 144
!analyze, 143, 144, 145
!bugdump, 145
!chkimg, 144
!CLRStack, 73, 75, 76, 145
!cs, 78, 143, 144
!dh, 143
!dlls, 143
!do, 74, 76
!dpcs, 144
!dso, 73, 75
!DumpHeap, 145
!DumpRuntimeTypes, 145
!DumpStackObjects, 145
!eeheap, 145
!envvar, 143, 144
!exchain, 144
!exqueue, 144
!filecache, 144
!fileobj, 144
!FinalizeQueue, 145
!fltkd, 49
!GCHandleLeaks, 145
!GCHandles, 145
!gflag, 52, 74, 143
!handle, 33
!heap, 57, 59, 63, 64, 144
!irp, 16, 46, 49, 144
!irpfind, 16, 144
!locks, 143, 144
!lpc, 144
!ndiskd.miniports, 144
!ndiskd.pktpools, 144
!pe, 70, 75, 145
!peb, 143
!pool, 145
!poolused, 144
!process, 144
!qlocks, 144
!ready, 144
!runaway, 60, 143
!running, 144
!scsikd.classext, 144
!session, 144
!sprocess, 144
!stacks, 144
!syncblk, 145
!sysinfo, 143, 145
!teb, 39
!thread, 16, 33, 46, 78
!Threads, 145
!uniqstack, 143
!VerifyHeap, 145
!vm, 144, 145
!process, 144
.chain, 145
.cxr, 18, 21, 28, 31
.ecxr, 18
.enumtag, 145
.for, 13
.frame, 31, 32, 72
.kframes, 143
.load, 70
.process, 77
.symfix, 143
.thread, 18, 28, 30, 31, 78
~, 143, 145
~*kv, 143
dp, 32
dps, 21, 52, 61
dpS, 39
dS, 143, 144, 145
dt, 31
k, 17, 18, 21, 28, 43, 51, 71, 74
kc, 60, 81
kn, 31, 72
lmv, 70, 143, 145
ln, 145

s, 143, 144
ub, 51, 73
uf, 30

# Memory Dump Analysis Anthology

## Volume 8b

**Dmitry Vostokov**
**Software Diagnostics Institute**

OpenTask

Published by OpenTask, Republic of Ireland

Copyright © 2015 by Dmitry Vostokov

Copyright © 2015 by Software Diagnostics Institute

All rights reserved. No part of this book may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, without the prior written permission of the publisher.

You must not circulate this book in any other binding or cover, and you must impose the same condition on any acquirer.

OpenTask books are available through booksellers and distributors worldwide. For further information or comments send requests to press@opentask.com.

Product and company names mentioned in this book may be trademarks of their owners.

A CIP catalog record for this book is available from the British Library.

ISBN-13: 978-1-908043-54-2 (Paperback)

First printing, 2015

Revision 1.02

## Table of Contents

**Preface** ........................................ 7

**About the Author** ........................................ 9

**PART 1: Professional Crash Dump Analysis and Debugging** ........................................ 11

Win32 Start Address Fallacy ........................................ 11
Multidimensionality of Exceptions ........................................ 13

**PART 2: Crash Dump Analysis Patterns** ........................................ 15

Reference Leak ........................................ 15
Origin Module ........................................ 19
Hidden Call ........................................ 21
Corrupt Structure ........................................ 26
Software Exception ........................................ 29
Crashed Process ........................................ 30
Variable Subtrace ........................................ 31
User Space Evidence ........................................ 37
Technology-Specific Subtrace (COM Client Call) ........................................ 38
Internal Stack Trace ........................................ 39
Distributed Exception (Managed Code) ........................................ 41
Thread Poset ........................................ 43

**PART 3: Pattern Interaction** ........................................ 45

Virtualized Process, Stack Trace Collection, COM Interface Invocation Subtrace, Active Thread, Spiking Thread, Last Error Collection, RIP Stack Trace, Value References, Namespace, and Module Hint ........................................ 45

**PART 4: A Bit of Science and Philosophy** ........................................ 57

Cantor Operating System ........................................ 57
Metaphor of Memory as a Directed Container ........................................ 57
Praxiverse ........................................ 58
When Universe is Going to End? ........................................ 58

Notes on Memoidealism ........ 59

**PART 5: Software Trace Analysis Patterns ........ 61**

Timeout ........ 61
Activity Overlap ........ 65
Adjoint Space ........ 67
Indirect Message ........ 70
Watch Thread ........ 75
Punctuated Activity ........ 77
Trace Mask ........ 78
Trace Viewpoints ........ 81
Data Reversal ........ 83
Recovered Messages ........ 85
Palimpsest Messages ........ 87
Message Space ........ 90
Interspace ........ 92
Translated Message ........ 94
Activity Disruption ........ 96

**PART 6: Fun with Debugging, Crash Dumps, and Traces ........ 99**

The Dump from the Future ........ 99
Exchange Rate on 16.12.14 ........ 99
Check the Plug ........ 100
Debugging Slang ........ 101

*YAWE ........ 101*
*Embedded Software Engineer ........ 101*
*Minute-wise ........ 101*
*Developer ........ 101*
*Multidigitalist ........ 101*
*KgB ........ 102*
*CIQ (Crash IQ) ........ 102*
*Pat Ching ........ 102*
*Explosive Mixture ........ 102*
*POEM ........ 102*
*YearNormous Day ........ 103*
*eNormous ........ 103*

2015 - The Year of RAM .......... 104
Diagnostics and Debugging in Science Fiction .......... 105
Software and Hardware Exceptions.......... 108
Logging for Kids.......... 110
Find the Bug .......... 111
Music for Debugging .......... 112
Tracing and Counting Book .......... 113
The Last Error.......... 114
Patching the Hardware Defect.......... 115
Pattern Match .......... 116

**PART 7: Software Narratology .......... 117**

Coding and Articoding.......... 117

**PART 8: Software Diagnostics, Troubleshooting, and Debugging .......... 119**

Special and General Trace and Log Analysis .......... 119
Projective Debugging .......... 123
Pattern! What Pattern? .......... 132
I Didn’t See Anything .......... 135

**PART 9: Art and Photography .......... 137**

Diagnostics Designer Glasses .......... 137
Pattern Diagnostics Logo .......... 138
Happy Valentine’s Day .......... 139
50 Shades of Crash Dump .......... 140
Computer Universe .......... 141
Failed Surveillance .......... 142
Debugging Allegory on FEB 23 .......... 143
Object in Signaled State .......... 144
Kernel Space Starts with 8 .......... 145
The Day of ST. P. The Elimination of Snakes .......... 146
The Fifth Column.......... 147
Proportionate Disproportionate Proportion.......... 148
Autoportrait in 5 Objects .......... 149
Kernel Works.......... 150
Chip Forensics .......... 151

Industrial Windows ........................................................................ 152
The Meaning of Life ........................................................................ 153
Hidden Bug........................................................................................ 154

**PART 10: Memory Forensics ........................................................... 155**

Artifact-Malware and its Primary and Secondary Effects .............................. 155

**PART 11: Miscellaneous................................................................. 161**

Quotes................................................................................................ 161
Status Updates .................................................................................... 163
Execution Residue ............................................................................... 164

**Appendix ........................................................................................ 165**

Patterns are Weapons for Massive Debugging............................................. 165
Crash Dump Analysis Checklist ............................................................... 166

**Index of WinDbg Commands ........................................................... 169**

# Index of WinDbg Commands

!address, 50, 55
!alpc, 167
!analyze, 41, 166, 167, 168
!bugdump, 168
!chkimg, 166
!CLRStack, 23, 168
!cs, 166, 167
!dh, 166
!dlls, 166
!dpcs, 167
!DumpHeap, 168
!DumpObj, 41
!DumpRuntimeTypes, 168
!DumpStackObjects, 168
!eeheap, 168
!envvar, 166, 167
!error, 49
!exchain, 166
!exqueue, 167
!filecache, 167
!fileobj, 167
!FinalizeQueue, 168
!GCHandleLeaks, 168
!GCHandles, 168
!gflag, 166
!gle, 48
!heap, 167
!irp, 167
!irpfind, 167
!locks, 166, 167
!lpc, 167
!ndiskd.miniports, 167
!ndiskd.pktpools, 167
!object, 16
!obtrace, 18
!pe, 168
!peb, 166
!pool, 168
!poolused, 16, 17, 167
!process, 11, 15, 30, 92, 167
!qlocks, 167
!ready, 167
!runaway, 47, 166
!running, 167
!scsikd.classext, 167
!session, 99, 167
!sprocess, 11, 167
!stacks, 43, 167
!sw, 45
!syncblk, 168
!sysinfo, 166, 168
!teb, 48
!thread, 11
!Threads, 168
!uniqstack, 166
!VerifyHeap, 168
!vm, 167, 168
.asm, 50
.chain, 168
.enumtag, 168
.kframes, 166
.load, 45
.reload, 45, 157
.symfix, 45, 157, 166
~, 166, 168
~*kv, 166
dc, 50, 155
dps, 49, 55
dS, 166, 167, 168
dt, 26, 27, 28
kL, 29, 46, 158
lmv, 56, 155, 156, 158, 166, 168
ln, 168
s, 166, 167
s-d, 55
ub, 24, 25, 49

# Memory Dump Analysis Anthology

## Volume 9a

**Dmitry Vostokov**
**Software Diagnostics Institute**

OpenTask

Published by OpenTask, Republic of Ireland

Copyright © 2016 by Dmitry Vostokov

Copyright © 2016 by Software Diagnostics Institute

All rights reserved. No part of this book may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, without the prior written permission of the publisher.

You must not circulate this book in any other binding or cover, and you must impose the same condition on any acquirer.

OpenTask books are available through booksellers and distributors worldwide. For further information or comments send requests to press@opentask.com.

Product and company names mentioned in this book may be trademarks of their owners.

A CIP catalog record for this book is available from the British Library.

ISBN-13: 978-1-908043-35-1 (Paperback)

First printing, 2016

# Table of Contents

**Preface ........................................................................................ 7**

**About the Author ........................................................................... 9**

**PART 1: Professional Crash Dump Analysis and Debugging ............................. 11**

When realloc is not a realloc............................................................ 11
WinDbg Shortcut !ddstack ................................................................ 12

**PART 2: Crash Dump Analysis Patterns ............................................... 15**

Stack Trace Collection (CPUs) ........................................................... 15
Object Distribution Anomaly (.NET Heap) ............................................ 19
Stack Trace Surface ......................................................................... 22
Hidden Stack Trace ......................................................................... 24
Evental Dumps ................................................................................ 27
Active Thread (Windows)................................................................... 55
Clone Dump ................................................................................... 59
Parameter Flow................................................................................ 63
Diachronic Module............................................................................ 67

**PART 3: Pattern Interaction ............................................................... 69**

Spiking Thread, Top Module, Module Hint, and Memory Fluctuation ............. 69

**PART 4: A Bit of Science and Philosophy ............................................... 83**

Quotes from Memoriarch ................................................................... 83

**PART 5: Software Trace Analysis Patterns ............................................. 85**

Ruptured Trace ............................................................................... 85
Sequence Repeat Anomaly ................................................................ 88
Adjoint Message .............................................................................. 90
Coupled Activities ........................................................................... 92
Error Powerset ................................................................................. 94

Trace Dimension ........................................................................................ 96
Calibrating Trace ........................................................................................ 98
Data Interval ........................................................................................ 99
Identification Messages ........................................................................................ 101

**PART 6: Fun with Debugging, Crash Dumps, and Traces................................ 103**

Dangerous Words ........................................................................................ 103
Debugging Slang........................................................................................ 104

*MOAN ........................................................................................ 104*
*LOG ........................................................................................ 104*
*Diplodoc ........................................................................................ 104*
*pMud........................................................................................ 104*
*HLL ........................................................................................ 104*
*Success ........................................................................................ 105*
*F00D........................................................................................ 105*
*Tor-mented ........................................................................................ 105*
*Obsession ........................................................................................ 105*
*Literature ........................................................................................ 105*
*CLERK ........................................................................................ 105*
*Analysis Paralysis ........................................................................................ 106*
*3D Dump ........................................................................................ 106*
*Star Wars ........................................................................................ 106*
*Daily Standup........................................................................................ 106*

Debugging Curiosities ........................................................................................ 107

*Hung vs. Hanged........................................................................................ 107*
*Trace Messages ........................................................................................ 107*
*13 ........................................................................................ 107*
*Similar Cases ........................................................................................ 107*
*Error 1917 ........................................................................................ 108*

Dump2Wave Update ........................................................................................ 109
Diagnostics and Debugging in Science Fiction ................................................ 110
Suspicious Volume 9a ........................................................................................ 111
Music for Debugging........................................................................................ 112

*Shpongle: Nothing Lasts But Nothing Is Lost* ............ *112*

**PART 7: Linux Core Dump Analysis Patterns ............ 113**

NULL Pointer (Data) ............ 113
Stack Trace ............ 114
NULL Pointer (Code) ............ 115
Spiking Thread ............ 116
Dynamic Memory Corruption (process heap) ............ 118
Execution Residue ............ 119
Coincidental Symbolic Information ............ 121
Stack Overflow (user mode) ............ 122
Divide by Zero (user mode) ............ 124
Local Buffer Overflow ............ 125
C++ Exception ............ 126
Paratext ............ 127
Active Thread ............ 129
Lateral Damage ............ 130
Critical Region ............ 131

**PART 8: Software Diagnostics, Root Cause Analysis, Debugging ............ 135**

Workaround Patterns ............ 135

*Axed Data* ............ *135*

Diagnostics, Forensics, Prognostics: The Copernican Revolution ............ 137
Pattern Repertoire ............ 140
Pattern-Oriented Software Internals: Pattern Paradigms and Software Internals Pattern Stack ............ 142
Software Diagnostics Canvas ............ 147
Software Traces and Logs as Proteins ............ 149
Patterns-Based Root Cause Analysis Methodology ............ 152
Teaching Complex Diagnostic Scenarios with Artificial Debugger (ArtDbg) and Pseudo-Memory Dumps ............ 156
The Scope of Software Diagnostics ............ 159

**PART 9: Art and Photography ............ 163**

W - I'M DEBUGGIN' IT® .......... 163
Coincidental Symbolic Information Pattern .......... 164
Pisa Fault System Model .......... 165
System Playing Tetris .......... 166
A Pattern of Zeroes .......... 167
Abnormal Structure .......... 168
Control Your Software Emissions! .......... 169
Component-Based Bug Architecture .......... 170

**PART 10: Miscellaneous .......... 171**

Quotes .......... 171
World Software Diagnostics Day .......... 173
Train Journey .......... 174

**Appendix .......... 175**

Crash Dump Analysis Checklist .......... 175
Pattern Changes .......... 178

**Index of WinDbg Commands .......... 179**

# Index of WinDbg Commands

!address, 60, 73, 76, 77
!alpc, 176
!analyze, 15, 25, 175, 176, 177
!bugdump, 177
!chkimg, 176
!CLRStack, 177
!cs, 175, 176
!ddstack, 3, 12
!dh, 175
!dlls, 175
!dpcs, 176
!DumpHeap, 19, 177
!DumpRuntimeTypes, 177
!DumpStackObjects, 177
!eeheap, 177
!envvar, 175, 176
!exchain, 176
!exqueue, 176
!filecache, 176
!fileobj, 176
!FinalizeQueue, 177
!GCHandleLeaks, 177
!GCHandles, 177
!gflag, 175
!heap, 61, 74, 76, 77, 81, 176
!irp, 176
!irpfind, 176
!locks, 175, 176
!lpc, 176
!ndiskd.miniports, 176
!ndiskd.pktpools, 176
!pe, 177
!peb, 175
!pool, 177
!poolused, 176
!process, 176
!qlocks, 176
!ready, 176
!runaway, 57, 69, 78, 175
!running, 15, 176
!scsikd.classext, 176
!session, 176
!sprocess, 176
!stacks, 176
!syncblk, 177
!sysinfo, 175, 177
!teb, 12
!Threads, 177
!uniqstack, 175
!VerifyHeap, 177
!vm, 176, 177
.chain, 177
.cxr, 25, 63
.ecxr, 26
.enumtag, 177
.exr, 25
.kframes, 175
.symfix, 175
~, 175, 177
~*k, 24, 59
~*kv, 175
dc, 64, 77
dd, 25
dp, 64
dps, 12, 65
dpS, 12, 13, 62
dS, 175, 176, 177
dt, 62
du, 78
kL, 64
kv, 24, 66
kvL, 63, 65
lmn, 59
lmv, 73, 175, 177
ln, 177
poi, 64
s, 175, 176
ub, 57, 64

# Memory Dump Analysis Anthology

## Volume 9b

**Dmitry Vostokov**
**Software Diagnostics Institute**

OpenTask

Published by OpenTask, Republic of Ireland

Copyright © 2016 by Dmitry Vostokov

Copyright © 2016 by Software Diagnostics Institute

All rights reserved. No part of this book may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, without the prior written permission of the publisher.

You must not circulate this book in any other binding or cover, and you must impose the same condition on any acquirer.

OpenTask books are available through booksellers and distributors worldwide. For further information or comments send requests to press@opentask.com.

Product and company names mentioned in this book may be trademarks of their owners.

A CIP catalog record for this book is available from the British Library.

ISBN-13: 978-1-908043-36-8 (Paperback)

First printing, 2016

## Table of Contents

**Preface** ........ 7

**About the Author** ........ 9

**PART 1: Crash Dump Analysis Patterns** ........ 11

Constant Subtrace ........ 11
Wait Chain (Nonstandard Synchronization) ........ 13
Not My Thread ........ 16
Window Hint ........ 17
Place Trace ........ 20
Handle Limit (GDI, User Space) ........ 22
Multiple Exceptions (Stowed) ........ 28
Stack Trace Signature ........ 35
Relative Memory Leak ........ 37
JIT Code (Java) ........ 40
Wait Chain (C++11, Condition Variable) ........ 42

**PART 2: A Bit of Science, Philosophy, and Religion** ........ 45

Morality and Virtual Worlds ........ 45
Quotes from Memoriarch ........ 45
On Lives, Narratives, and Memory ........ 45
Notes on Memoidealism ........ 46
Worst Simulation World Hypothesis ........ 46
Memory Ablution ........ 46

**PART 3: Software Trace Analysis Patterns** ........ 47

Data Selector ........ 47
Declarative Trace ........ 49
Trace Extension ........ 50
Fourier Activity ........ 51
Fiber of Activity ........ 54
Missing Data ........ 56
Message Pattern ........ 57

Activity Theatre ... 58
Small DA+TA ... 59
Surveyor ... 61
Quotient Trace ... 62

**PART 4: Fun with Debugging, Crash Dumps, and Traces ... 63**

Debugging Slang ... 63

*Apoology ... 63*
*MedioCriticalSection ... 63*
*SPASM ... 63*
*NoOO ... 63*
*AI ... 63*
*To Come Out of the Shell ... 64*
*3D Weekend ... 64*
*To Crawl into (One's Shell) ... 64*
*Bad Feeling ... 64*
*The Valley of Crash Dumps ... 65*
*Early Debugging ... 65*
*CHARLATAN ... 65*

Diagnostics and Debugging in Science Fiction ... 66
James Bond's Bugcheck and Error ... 66
Two-field System Agriculture ... 66
Bugs and InfoSec ... 66
Program Evolution ... 67
Roman + Hex ... 67
Debugging Curiosities ... 67

*Trace Messages ... 67*

Moscow Scare ... 68
Vacuum Needs PDB ... 69
My surname decomposed ... 69
Slavery ... 69
PORCA ... 70
Double Fee Request ... 70
Word Symmetry and Soviet History ... 70

Everything You Need for Debugging ... 71
Bugs in the System ... 72
Direct HR Reporting ... 72
The Devil at My Heels ... 72

**PART 5: Software Narratology ... 73**

**PART 6: Software Diagnostics, Root Cause Analysis, Troubleshooting, and Debugging ... 75**

Diagnostics of Things (DoT) ... 75
Riemann Root Cause Analysis Language ... 76
Problem Solving as Code ... 80
Dia|gram Graphical Diagnostic Analysis Language ... 82
Iterative Pattern-Oriented Root Cause Analysis ... 84
Theoretical Software Diagnostics and Education ... 86

**PART 7: Art and Photography ... 89**

Heap Corruption Explained by Lego Bricks ... 89
Linked List Illustrated by Lego Bricks ... 90
The Stack of Words ... 91
Packed and Unpacked Structures Illustrated by Lego Bricks ... 92
What Color is Your Instruction? ... 93
Sluggish System under Observation ... 104
Happy Debugging Card (Halloween Style) ... 105

**PART 8: Structural Memory Patterns ... 107**

Region Strata ... 107

**PART 9: Miscellaneous ... 111**

Quotes ... 111
English for Software Engineers (with UML) ... 114
Visual Learning Guide to Stack Traces ... 115
Real Programmers - No Impossible Code ... 116
Debugger Log Analyzer: Inception ... 118
Technical Books as Software ... 120

Job Forensic Archaeology ........................................................................ 122
The Physical Spike ........................................................................ 123
Software Experience Reuse through Generations .......................................... 124
Fiber Bundle Reading ........................................................................ 125

**PART 10: Software Generalist ........................................................................ 127**

Sorting and Early Greek Philosophers ........................................................ 127
Software as Means of Production ........................................................ 127
MVC Worldview and the Origin of Economic Order ........................................ 128
Software Generalist View of Religion ........................................................ 129
Mod N Reading System ........................................................................ 130
Computational Collectives ........................................................................ 133
Software Generalist Worldview ........................................................ 134
Event Tracing for Windows in UML ........................................................ 135
Empires of the Code ........................................................................ 136
Standard Model and UML ........................................................................ 137
Software Accommodation ........................................................................ 138
Software Generalism ........................................................................ 138
Software Labour and Alienation ........................................................ 139
Finite Sets ........................................................................ 140
Computational Slotting Fees ........................................................ 141
On Facts about Software ........................................................................ 141
Optimal Discrete Reading Chunks ........................................................ 142
On Software and Ethics ........................................................................ 142
On Software Space-Time ........................................................................ 143
Worship of Memory ........................................................................ 143
Remembering d’Alembert ........................................................................ 144
Software and Philosophical Beliefs ........................................................ 144
On Good Software ........................................................................ 145
On Babbage-Chambers Paradox ........................................................ 145
On Abandonment ........................................................................ 146
Cooperative Multireading Revisited and Started ........................................... 147

**Index of WinDbg Commands ........................................................................ 149**

# Index of WinDbg Commands

!address, 17
!cs, 15
!error, 28, 30, 32, 33, 66
!handle, 42
!lmi, 69
.asm, 23
.cxr, 13, 14, 70
.exr, 28
.formats, 29
.lines, 30
.while, 25, 26
?, 26
~, 25
~*k, 16
dp, 23, 29
dps, 30, 32, 33, 113
dq, 23, 25
dt, 29
dw, 25
k, 40, 72
kc, 13, 14, 15, 22, 36, 68
kL, 42
kv, 42
lmp, 93, 98
r, 25, 26
s-a, 17
s-sa, 17
ub, 33, 34, 40, 41
uf, 15, 23

# Memory Dump Analysis Anthology

## Volume 10

**Dmitry Vostokov**
**Software Diagnostics Institute**

OpenTask

Published by OpenTask, Republic of Ireland

Copyright © 2017 by Dmitry Vostokov

Copyright © 2017 by Software Diagnostics Institute

All rights reserved. No part of this book may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, without the prior written permission of the publisher.

You must not circulate this book in any other binding or cover, and you must impose the same condition on any acquirer.

OpenTask books are available through booksellers and distributors worldwide. For further information or comments send requests to press@opentask.com.

Product and company names mentioned in this book may be trademarks of their owners.

A CIP catalog record for this book is available from the British Library.

ISBN-13: 978-1-908043-85-6 (Paperback)

First printing, 2017

Revision 1.03 (May 2017)

# Table of Contents

**Preface ........................................................................................... 7**

**About the Author ............................................................................ 9**

**PART 1: Crash Dump Analysis Patterns ................................................ 11**

Quotient Stack Trace ........................................................................ 11
Module Stack Trace .......................................................................... 12
Foreign Module Frame ....................................................................... 13
Unified Stack Trace .......................................................................... 16
Mirror Dump Set .............................................................................. 18
Memory Fibration ............................................................................. 20
Aggregated Frames ........................................................................... 21
Value Deviation (Structure Field) ...................................................... 22
Stack Trace (I/O Devices) ................................................................. 24
High Contention (.NET Heap) ............................................................ 26
Frame Regularity .............................................................................. 30
Deadlock (.NET Finalizer) ................................................................. 34
Invalid Parameter (Runtime Function) ................................................ 38
Wait Chain (SRW Lock) ..................................................................... 40
Stack Trace Motif .............................................................................. 42

**PART 2: Linux Core Dump Analysis Pattern ........................................... 43**

Module Stack Trace .......................................................................... 43

**PART 3: Software Trace Analysis Patterns ............................................ 45**

Corrupt Message .............................................................................. 45
Projective Space ............................................................................... 47
Ornament ......................................................................................... 50
Poincaré Trace ................................................................................. 52
De Broglie Trace Duality ................................................................... 55
Braid Group ...................................................................................... 57
Delay Dynamics ................................................................................ 59
Activity Quantum .............................................................................. 60

Trace Presheaf ........ 61
Message Directory ........ 63
Galois Trace ........ 66
Singleton Trace ........ 68
Braid of Activity ........ 69
Tensor Trace ........ 70
Unsynchronized Traces ........ 72
Intrinsic ID ........ 74
Combed Trace ........ 75
Activity Packet ........ 76

**PART 4: Software Diagnostics, Root Cause Analysis, Troubleshooting, and Debugging ........ 77**

Topological Software Trace and Log Analysis ........ 77
Is Your Security Healthy? ........ 79
Software Diagnostic Space as a General Graph of Software Narratives ........ 80
Software Diagnostics Metaphors ........ 85

*Software Diagnostics as Archaeology ........ 85*

Pattern-Oriented Diagnostic Analysis Process ........ 86
Principles of Pattern-Oriented Software Data Analysis ........ 88
Abstract Debugging Commands (ADC) Initiative ........ 91
Reducing Analysis Pattern Complexity via Elementary Analysis Patterns ........ 92
Categorical Foundations of Software Diagnostics ........ 96
Existential Prognostics: Periodic Table of Diagnostic Patterns ........ 98
Software Codiagnostics ........ 100
The Unity of Pattern-Oriented Software Diagnostics ........ 103

**PART 5: A Bit of Science, Philosophy, and Religion ........ 105**

Quotes from Memoriarch ........ 105
Notes on Memoidealism ........ 105

**PART 6: Fun with Debugging, Crash Dumps, and Traces ........ 107**

Debugging Slang ........ 107

*SCANDAL* ........ *107*
*WTF* ........ *107*
*UOP* ........ *107*
*HOT* ........ *107*
*HOME* ........ *107*
*anOS* ........ *108*
*theOS* ........ *108*
*TCH* ........ *108*
*Top NoTCH* ........ *108*

Problem Solving Techniques ........ 109

*Dissolution* ........ *109*

Software Temperature ........ 109
Watson ........ 109
Cosmic Rays in Memory ........ 110
Area 51 ........ 113
Measuring Software Diagnostics ........ 114
Cash and Crash ........ 114
Debugging Law ........ 114
Suggested Pool Tags ........ 114
TOR ........ 114
Attitude to Debugging ........ 115
British vs. American Spelling ........ 116
How I Became a Grandmaster ........ 117
Memory Dumps and VAT ........ 117
Updatician ........ 117
Traces of Cyrillic Alphabet ........ 117
Prolific Letter ........ 118
Code and Edoc ........ 118
The Three-Software Vendor Body Problem ........ 118
Modem Troubleshooting and Putin ........ 118
Bugs for Fireworks ........ 119
Critical SnowLOB ........ 120
Apotypomamnimisophobia ........ 121
Schadenfreude of AI ........ 121
Diagnostics and Debugging in Science Fiction ........ 121
The Mozart of Diagnostics ........ 122

Keyboard Problems ........................................ 124

PART 7: Art and Photography ........................................ 125

(t)ra(c)in(g) ........................................ 125
Development Muses ........................................ 126
Pattern Inside and Outside ........................................ 127
Between Diagnostic Activities ........................................ 128
Patched Bug Construction Kit ........................................ 129
Russian Binary ........................................ 130
Secret Russian Binary ........................................ 131
Postanalysm ........................................ 132

PART 8: Miscellaneous ........................................ 135

Quotes ........................................ 135
Is There Any Life Inside Windows? ........................................ 137
The Condition of My Productivity ........................................ 138
CyberSpace and the Solution to CyberProblems ........................................ 139
VAX/VMS Debugging Artefact ........................................ 141
Observing Patterns of Cloud Structure and Behavior ........................................ 143

Appendix ........................................ 145

Crash Dump Analysis Checklist ........................................ 145

Volume Index ........................................ 149

Memory Analysis Patterns ........................................ 149
Trace and Log Analysis Patterns ........................................ 161

Index of WinDbg Commands ........................................ 167

# Index of WinDbg Commands

!alpc, 146
!analyze, 38, 39, 145, 146, 147
!bugdump, 147
!chkimg, 146
!CLRStack, 34, 147
!cs, 30, 145, 146
!dc, 113
!devob, 25
!devobj, 24, 25
!devstack, 24
!dh, 145
!dlls, 145
!dpcs, 146
!DumpHeap, 147
!DumpRuntimeTypes, 147
!DumpStackObjects, 147
!eeheap, 147
!envvar, 145, 146
!exchain, 146
!exqueue, 146
!filecache, 146
!fileobj, 146
!FinalizeQueue, 147
!GCHandleLeaks, 147
!GCHandles, 147
!gflag, 145
!heap, 146
!irp, 24, 146
!irpfind, 146
!locks, 145, 146
!lpc, 146
!ndiskd.miniports, 146
!ndiskd.pktpools, 146
!pe, 147
!peb, 145
!pool, 147
!poolused, 146
!process, 23, 146
!qlocks, 146
!ready, 146
!runaway, 29, 145
!running, 146
!scsikd.classext, 146
!search, 113
!session, 146
!sprocess, 146
!stacks, 19, 146
!syncblk, 147
!sysinfo, 145, 147
!Threads, 147
!uniqstack, 145
!VerifyHeap, 147
!vm, 146, 147
.chain, 147
.enumtag, 147
.formats, 113
.kframes, 145
.symfix, 145
~, 37, 145, 147
~*k, 26, 40
~*kc, 16
~*kv, 145
dd, 37
dpp, 36
dS, 145, 146, 147
dt, 22
k, 13, 34, 35, 38
kL, 12
kvL, 36
lmv, 145, 147
ln, 147
s, 145, 146
ub, 33
version, 19

# Memory Dump Analysis Anthology

## Volume 11

**Dmitry Vostokov**
**Software Diagnostics Institute**

OpenTask

Published by OpenTask, Republic of Ireland

Copyright © 2018 by Dmitry Vostokov

Copyright © 2018 by Software Diagnostics Institute

All rights reserved. No part of this book may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, without the prior written permission of the publisher.

You must not circulate this book in any other binding or cover, and you must impose the same condition on any acquirer.

OpenTask books are available through booksellers and distributors worldwide. For further information or comments send requests to press@opentask.com.

Product and company names mentioned in this book may be trademarks of their owners.

A CIP catalog record for this book is available from the British Library.

ISBN-13: 978-1-912636-11-2 (Paperback)

First printing, 2018

Revision 1.0 (October 2018)

## Table of Contents

**Preface ..... 9**

**About the Author ..... 11**

**PART 1: Crash Dump Analysis Patterns ..... 13**

System Call ..... 13
Data Correlation (CPU Times) ..... 15
Stack Trace Race ..... 17
Hyperdump ..... 19
Disassembly Ambiguity ..... 24
Hidden Exception (Managed Space) ..... 25
Insufficient Memory (Stack) ..... 28
Exception Reporting Thread ..... 34

**PART 2: Software Trace Analysis Patterns ..... 35**

Ultrasimilar Messages ..... 35
Hedges ..... 37
Trace Field ..... 39
Script Messages ..... 41
Working Set ..... 42
Trace Homotopy ..... 44
Signal ..... 45
Renormalization ..... 47
Motivic Trace ..... 48
Significant Interval ..... 50
Random Data ..... 52
Truncated Data ..... 53
Time Scale ..... 55
Trace Sharding ..... 56
Phantom Activity ..... 57

**PART 3: Software Diagnostics, Root Cause Analysis, Troubleshooting, and Debugging ..... 59**

The Most Important Skill in Software Diagnostics. ........................................ 59

Pattern-Oriented Data Analysis Example ........................................ 61

Diagnostic Operads ........................................ 63

Mathematical Concepts in Software Diagnostics and Software Data Analysis.. 66

Software Diagnostics Engineering ........................................ 69

Narrachain ........................................ 71

Diagnostics-Driven Development ........................................ 74

Integral Diamathics – Tracing the Road to Root Cause ........................................ 75

Anolog.io ........................................ 77

Meso-problem Solving using Meso-patterns ........................................ 79

Lego Log Analysis ........................................ 82

Artificial Chemistry Approach to Software Trace and Log Analysis ........................................ 87

**PART 4: Fun with Debugging, Crash Dumps, and Traces ........................................ 93**

Debugging Slang ........................................ 93

*QUICK ........................................ 93*

*DREAM ........................................ 93*

*DOSE ........................................ 93*

*Fex ........................................ 93*

*DANCE ........................................ 94*

*CORPSE ........................................ 94*

*Hi ........................................ 94*

*SOS ........................................ 94*

*GUT ........................................ 94*

*Autopsy ........................................ 95*

*FILOsophy and FILOlogy ........................................ 95*

*Timesheets and Timeshits ........................................ 95*

*Software Logomancy ........................................ 95*

*RhaPSODy ........................................ 95*

*Developer ........................................ 96*

Diagnostics and Debugging in Science Fiction ........................................ 97

Russian-English Lexical Connection ........................................ 98

Defects in Logs ........................................ 98

A Space Makes a Difference ........................................ 98

Digital Transformation ........................................ 98

Pathology ........................................ 98

Freemallocers .................... 99
Double Layoff .................... 99
Dump in Morse Code .................... 99
Pushkin and Updates .................... 99
Law of Misprints .................... 99
SoftwareLog, MaintainLog, MountainLog .................... 100
Cloud Patches .................... 101
Puree Windows Cooking .................... 102

*An Edible CPU Chip .................... 102*

Trying on a Sherlock's Hat .................... 103
The Reality is a Matrix .................... 105
Problem Solving Exercises .................... 106
Counting to 10 .................... 107
Music for Debugging .................... 107
A Bug Climbs a Book .................... 108
A Computer Crash .................... 109
Traces on Roads .................... 110
Lego Dump Analysis .................... 112

**PART 5: Art and Photography .................... 113**

Cyberspace Diagnostics .................... 113
Chasing a Beautiful Bug .................... 114
Diagnostics and Poetry .................... 116
Diagnostics in a Wild .................... 117
Interview Preparation Deque .................... 118
Practicing Sorting Algorithms .................... 119
The Open/Closed Principle .................... 120
Microsoft Campus in Redmond .................... 121
Soviet Glasses and Apple Hardware .................... 123
Design/Testing Proportion .................... 124

**PART 6: Debugging Dictionary .................... 125**

7 .................... 125
8 .................... 128
Breakpoint .................... 131

Crash ........ 133
Hang ........ 134
Kernel Space ........ 135
Memory Dump ........ 137
Memory Space ........ 138
Physical Memory ........ 139
Thread ........ 140
User Space ........ 141
Virtual Memory ........ 143

**PART 7: Tools ........ 145**

Dump2Picture Version 2.0 ........ 145
Window2Dump ........ 149
WindowHistory ........ 150
WindowHistory Mobile ........ 155
MessageHistory ........ 159
ScreenHistory ........ 161
ProcessHistory ........ 164
Using SSSL Principle to Design Support Tools ........ 165
Repair Clipboard Chain ........ 166
The Inception of Debugging Studio ........ 168
Easter Egg ........ 169

**PART 8: Miscellaneous ........ 171**

Selected Crashes from My Computers ........ 171
WinDbg Notes ........ 211
Resume and CV as Memory Analysis Artifacts and General Traces ........ 219
Quotes ........ 220
My Road to Modern C++ ........ 222
Algorithms for Breakfast ........ 225
Applying API Wrapper Pattern ........ 229
Clipboard Issues Explained ........ 233
Inside Citrix - November 2006 ........ 238
Looking at Software Problems from a Different Angle ........ 243
Me and "Windows NT/2000 Native API Reference" ........ 244

**Appendix ........ 245**

Curriculum Vitae ........................................................................................ 245
Resume in WinDbg Style ........................................................................... 254
Resume in GDB Style ................................................................................. 256
Windows Internals Certificate ................................................................... 258
Dump2Picture 2.2.3 Source Code .............................................................. 259
Dump2Wave 1.3.3 Source Code ................................................................ 262
Window2Dump 1.0 Source Code ............................................................... 265

**Index of WinDbg Commands ........................................................................ 273**

# Index of WinDbg Commands

!address, 19, 172
!alpc, 215
!analyze -v, 17
!chkimg, 214
!CLRStack, 26
!dh, 20
!DumpStackObjects, 27
!error, 30, 204
!for_each_process, 214
!gle, 216
!heap, 184, 211
!runaway, 15, 177, 185, 206
!running, 218
!sprocess, 216
!stacks, 18
!teb, 26, 30, 33, 216
!thread, 215
!vm, 175
.chain, 212
.cxr, 191, 195, 196
.ecxr, 34, 209, 210, 215
.exr, 192, 215, 216
.formats, 191, 212
.frame, 33
.imgscan, 20
.lastevent, 28
.process, 214
.trap, 215, 218
~*k, 25, 208, 209
~<n>k, 29, 34, 172, 177, 199
~<n>s, 26, 30, 206, 215
a, 213
dps, 30
dpS, 31
g, 214
k, 17, 24, 176, 183, 190, 191, 195, 200, 201, 202, 210, 218
kc, 28, 32, 34, 171, 172, 173, 178, 180, 185, 187, 188, 192, 193, 196, 197, 203, 205, 206, 207, 213
lm, 126, 129
lmt, 174
lmu, 218
lmv, 217, 218
r, 30, 188
s-a, 213
s-su, 33
ub, 14, 24
x, 189, 214

# Memory Dump Analysis Anthology

## Volume 12

**Dmitry Vostokov**
**Software Diagnostics Institute**

OpenTask

Published by OpenTask, Republic of Ireland

Copyright © 2019 by Dmitry Vostokov

Copyright © 2019 by Software Diagnostics Institute

All rights reserved. No part of this book may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, without the prior written permission of the publisher.

You must not circulate this book in any other binding or cover, and you must impose the same condition on any acquirer.

OpenTask books are available through booksellers and distributors worldwide. For further information or comments, send requests to press@opentask.com.

Product and company names mentioned in this book may be trademarks of their owners.

A CIP catalog record for this book is available from the British Library.

ISBN-13: 978-1-912636-12-9 (Paperback)

First printing, 2019

Revision 1.01 (December 2019)

# Table of Contents

**Preface ........................................................................................... 7**

**About the Author ............................................................................. 9**

**PART 1: Crash Dump Analysis Patterns ........................................................ 11**

Active Space ........................................................................................ 11
Stack Overflow (Insufficient Memory) ............................................................ 13
Subsystem Modules ............................................................................... 17
Region Profile ...................................................................................... 18
Region Clusters .................................................................................... 19
Source Stack Trace ................................................................................ 23

**PART 2: Pattern Interaction ................................................................ 25**

Abridged Dump, C++ Exception, Incorrect Stack Trace, Stack Trace Collection, Exception Stack Trace and Not My Version ............................................................ 25
Python Crash Dump Analysis Case Study ........................................................ 29

**PART 3: Software Trace Analysis Patterns .................................................. 37**

Critical Point ........................................................................................ 37
Drone Message ..................................................................................... 39
Minimal Trace ...................................................................................... 41
Polytrace ............................................................................................ 43
Trace String ........................................................................................ 44
Equivalent Messages ............................................................................... 47
Cartesian Trace .................................................................................... 48
Message Annotations .............................................................................. 50
CoTrace (CoLog, CoData) ........................................................................ 51
Moduli Trace ....................................................................................... 54
Trace Similarity .................................................................................... 57
Explanation Trace .................................................................................. 59
Split Message ...................................................................................... 60
Phase Transition ................................................................................... 62
Message Flow ...................................................................................... 65

Trace Constants ........................................................ 66

**PART 4: Software Diagnostics, Root Cause Analysis, Troubleshooting, and Debugging ........................................................ 67**

Introducing Software Pathology ........................................................ 67
10 Years of Trace and Log Analysis Patterns ........................................................ 69
Log's Loxels and Trace Message's Mexels Graphical Representation of Software Traces and Logs ........................................................ 74
Analysis Pattern Duality ........................................................ 80
10 Years of Software Narratology ........................................................ 81
Application of Trace and Log Analysis Patterns to Image Analysis: Introducing Space-like Narratology ........................................................ 86
Machine Learning Square and Software Diagnostics Institute Roadmap ......... 88

**PART 5: Fun with Debugging, Crash Dumps, and Traces .................................. 91**

Debugging Slang ........................................................ 91

*SANTA ........................................................ 91*
*Call Center ........................................................ 91*
*Fu ........................................................ 91*
*ID ........................................................ 91*

Clear Message ........................................................ 92
10-Year Challenge ........................................................ 93
Space Matters ........................................................ 97
Loop with a Bug ........................................................ 98
Dalkey ........................................................ 98
Ominous PID ........................................................ 98
Blue Screen at 32K Feet ........................................................ 99
Music for Debugging ........................................................ 100
Machine Learning ........................................................ 101

**PART 6: Art and Photography ........................................................ 103**

Crash Dump Collection ........................................................ 103
Failure Code ........................................................ 104
Automated Debugging ........................................................ 105

Real and Artificial Bugs ........ 106
Two Windows ........ 107
Tracing Tools ........ 108
Log ........ 109
Threads ........ 110
Window ........ 111
Linux Trace ........ 112
Logs ........ 113
Stop Code ........ 114
Monolithic Architecture Leak ........ 115
Visual Studios ........ 116
Data Structures ........ 117
Early Data Visualization ........ 118
Early Data Science, Clustering, and Histogramming ........ 119
Doing Exercise 0 in Sports Club ........ 120
Burger Trace Frames and Adjont Space of Chips ........ 121
Patterns of Macro and Micro ........ 122
Convex Programming Layout ........ 123

**PART 7: Book Covers ........ 125**

Writing Bad Code: Software Defect Construction, Simulation and Modeling of Software Bugs ........ 125
Software Internals for Machine Learning ........ 126

**PART 8: Miscellaneous ........ 127**

Selected Crashes from My Computers ........ 127
WinDbg Notes ........ 129
Quotes ........ 135
In a Chemical Laboratory ........ 136
Historical Reminiscences ........ 137
Baseplate Representation of Chemical Structure ........ 138
What I'm passionate about? ........ 148
C++ as a Scripting Tool ........ 152
In Memory ........ 155
The Road to Linux Kernel Space ........ 156

**Appendix ........ 159**

Stack Trace Patterns ........................................................................................ 159

**Volume Index ....................................................................................... 161**

Memory Analysis Patterns ........................................................................... 161
Trace and Log Analysis Patterns ...................................................................... 172

**Index of WinDbg Commands ........................................................................ 179**

# Index of WinDbg Commands

!address, 14
!analyze -v, 26, 94, 133
!blackboxbsd, 134
!blackboxntfs, 134
!blackboxpnp, 134
!dh, 129
!error, 14, 15
!mrt100sos, 130
!runaway, 11, 127
!teb, 15, 135
!vm, 133
.cxr, 26, 27, 32
.ecxr, 32
.exr, 13
.reload, 25, 135
.symfix, 25
.sympath, 32, 93
~*, 131
~*kc, 14
~s, 15, 27
dt, 135
k, 23, 26, 27, 30
kc, 13, 127, 128, 130
kL, 11, 32
lmf, 17
lmt, 129
lmv, 28, 31, 129
r, 13, 127

# Memory Dump Analysis Anthology

## Volume 13

**Dmitry Vostokov**
**Software Diagnostics Institute**

OpenTask

Published by OpenTask, Republic of Ireland

Copyright © 2020 by Dmitry Vostokov

Copyright © 2020 by Software Diagnostics Institute

All rights reserved. No part of this book may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, without the prior written permission of the publisher.

Product and company names mentioned in this book may be trademarks of their owners.

OpenTask books are available through booksellers and distributors worldwide. For further information or comments, send requests to press@opentask.com.

A CIP catalog record for this book is available from the British Library.

ISBN-13: 978-1-912636-13-6 (Paperback)

First printing, 2020

Revision 1.00 (August 2020)

# Table of Contents

**Preface** ........................................................................................................ 6

**About the Author** ........................................................................................... 7

**PART 1: Crash Dump Analysis Patterns** ............................................................ 9

Hidden Stack ........................................................................................................ 9
Interrupt Stack ..................................................................................................... 12
Invalid Pointer (Objects) ...................................................................................... 14
False Memory ....................................................................................................... 17
Frame Trace ......................................................................................................... 26
Pointer Cone ........................................................................................................ 35
Context Pointer .................................................................................................... 37
Pointer Class ........................................................................................................ 38
Lateral Damage (CPU Mode).................................................................................. 42

**PART 2: Software Trace Analysis Patterns** ....................................................... 47

Generative Trace................................................................................................... 47
Defect Group......................................................................................................... 49
Trace Flux ............................................................................................................. 50
Trace Path ............................................................................................................ 51
Trace Summary ..................................................................................................... 52
Causal History ...................................................................................................... 53
Causal Messages ................................................................................................... 55
Causal Chains ....................................................................................................... 56
Trace D'Enfant ...................................................................................................... 58
Sorted Trace.......................................................................................................... 60
Trace Shape........................................................................................................... 61
Trace Contour ....................................................................................................... 62
Trace Fabric........................................................................................................... 63
Semantic Field....................................................................................................... 65
Trace Foliation ...................................................................................................... 67

**PART 3: Software Diagnostics, Root Cause Analysis, Troubleshooting, and Debugging** ............................................................................................................. 69

Tool Objects ........................................ 69
Debugging and Category Theory ........................................ 71
Introducing Diags: Diagnostic Analysis Gestures and Logues ........................................ 83

**PART 4: Fun with Debugging, Crash Dumps, and Traces ........................................ 87**

Happy New Year 2020! ........................................ 87
Frightening Bug ........................................ 91
42 ........................................ 91
Secure Coding ........................................ 91
Music for Debugging ........................................ 92

*Virtuoso Piano Etudes ........................................ 92*
*A Trace and Log of Classical Music Events ........................................ 92*

Grammatical Deadlock ........................................ 93
Debugging Pain ........................................ 94
Rude Calculator ........................................ 95
Debugging Architecture Using Grammar Tools ........................................ 96
Incredible Observations ........................................ 97
Debugging by Action ........................................ 98
I have a Joke… ........................................ 99

**PART 5: Art and Photography ........................................ 101**

Client-Server Stack ........................................ 101
Post-modeling State ........................................ 102
Arrow ........................................ 103
Categorical Lunch ........................................ 104
Less Space, More Time (Reflections in Lockdown) ........................................ 105
Coffee Before and After Lockdown ........................................ 106
Ad Errors ........................................ 107
Worse than Bluescreen ........................................ 108

**PART 6: Miscellaneous ........................................ 109**

My Pathology Book Collection ........................................ 109
WinDbg Notes ........................................ 110
Quotes ........................................ 111

My Dream Modeling Book ... 112
Tracing My Data Science Path ... 113
The Memory of a Bat ... 117
My Favorite Category Theory Books ... 118
Two Security Book Collections ... 183
My First Crash Dump Analysis Presentation ... 186
Books as Categories ... 187
Personal Learning: A Historical Parallelism ... 188
My First Computer Program ... 189
Programming Paradigms and Software Execution Artifacts ... 190
CI/CD: Computational Intelligence/Correct Diagnosis ... 191
My Number Theory Book Collection (Beginning) ... 192

**Appendix ... 195**

Hidden Artifact Patterns ... 195
Pointer Patterns ... 196

**Index of WinDbg Commands ... 197**

# Index of WinDbg Commands

!address, 9, 12, 13, 14, 24, 41
!analyze -v, 23, 42, 88
!ca, 22
!for_each_frame, 27, 29
!for_each_register, 37
!process, 18, 23
!thread, 12, 43
!vad, 12, 19, 22
.cxr, 34
.ecxr, 14, 23, 24, 40, 88, 89
.effmach, 42, 110
.frame, 21, 29, 38, 41
.lines, 10
.segmentation, 44
.thread, 19, 21, 33, 40
?, 35, 110
~, 9, 24
da, 89
dc, 19, 22, 24
dp, 38, 43
dps, 35
dpS, 10, 11, 13, 44
dt, 41
du, 89
dv, 38, 41
dw, 89
dx, 24, 89
g, 21
k, 42, 43, 44
kc, 33, 34
kL, 9, 15, 19, 21, 24, 38, 40
kPL, 26
r, 35, 43
u, 15
ub, 13, 15

# Memory Dump Analysis Anthology

## Volume 14

**Dmitry Vostokov**
**Software Diagnostics Institute**

OpenTask

Published by OpenTask, Republic of Ireland

Copyright © 2021 by Dmitry Vostokov

Copyright © 2021 by Software Diagnostics Institute

All rights reserved. No part of this book may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, without the prior written permission of the publisher.

Product and company names mentioned in this book may be trademarks of their owners.

OpenTask books are available through booksellers and distributors worldwide. For further information or comments, send requests to press@opentask.com.

A CIP catalog record for this book is available from the British Library.

ISBN-13: 978-1-912636-14-3 (Paperback)

Revision 1.00 (August 2021)

## Table of Contents

**Preface** ........................................................................................ 6

**Acknowledgments** ........................................................................... 7

**About the Author** ............................................................................ 8

**PART 1: Crash Dump Analysis Patterns** ......................................... 9

False Frame ........................................................................................ 9
Procedure Call Chain ......................................................................... 11
C++ Object.......................................................................................... 12
COM Exception .................................................................................. 16
Structure Sheaf .................................................................................. 19
Saved Exception Context (.NET)......................................................... 20
Rough Stack Trace (Managed Space) .................................................. 25

**PART 2: Pattern Interaction** ............................................................ 27

Exception Reporting Thread, Stored Exception, Exception Stack Trace, Exception Module, Foreign Module Frame, and Stack Trace Motif ................................. 27

**PART 3: Software Trace Analysis Patterns** ...................................... 35

Flag.................................................................................................... 35
Strand of Activity ............................................................................... 36
Cord of Activity .................................................................................. 37
Text Trace .......................................................................................... 39
Weave of Activity ............................................................................... 41
Multidimensional Message ................................................................ 43
Message Metadata ............................................................................ 44
Trace Quilt.......................................................................................... 46
Feature of Activity.............................................................................. 48
Trace Schema ..................................................................................... 50
Definition Trace ................................................................................. 52
Serial Trace......................................................................................... 53
Container Trace.................................................................................. 55

Trace Join ... 56

**PART 4: Cloud Analysis Patterns ... 57**

Introducing Methodology and System of CAPS ... 57

**PART 5: Software Diagnostics, Root Cause Analysis, Troubleshooting, and Debugging ... 67**

The Fractal Nature of Software Traces and Logs ... 67
General Architecture of Analysis Pattern Networks ... 68

**PART 6: Software Narratology ... 73**

Exercises in Tracing Style ... 73

*Prologue ... 73*

Literary Theory Terms ... 75

*Ab ovo, in medias res, flashback, abridged edition ... 75*
*Abstract, accent, act, action, adaptation, address ... 76*

**PART 7: Fun with Debugging, Crash Dumps, and Traces ... 79**

Reading Capital ... 79
INSECuriTy ... 79
Desperate AI ... 79
Imperator ... 79
Buratino and Security ... 79
Two Types of Dump Analysis Jobs ... 79
Debugging Slang ... 80

*POET ... 80*
*ijit ... 80*
*DREAM ... 80*
*Logistics ... 80*
*Poorrific ... 80*

**PART 8: Art and Photography ... 81**

Chasing a Trace ... 81
Love of Logs ... 82
When Anomaly Detection is Your Bread ... 83
Observability of Traces ... 84
Array[T] ... 85
Old Card Bookmark ... 86
Cloud Tea Break ... 87
Fiber Bundle in the Wild ... 88

**PART 9: A Bit of Science, Philosophy, and Religion ... 89**

Plotinus on Overflow ... 89
Ereignis and Error ... 90

**PART 10: Miscellaneous ... 91**

WinDbg Notes ... 91
Quotes ... 95
My Favorite Category Theory Books (Continuation) ... 96
Data Pathology ... 101
FP Police ... 102
The Seed of Trace and Log Analysis Patterns ... 103
My Number Theory Book Collection (Continuation) ... 104
Artifacts for Lockdown ... 110
Books that Influenced Me ... 111
Streaming Architecture of Pattern-Oriented Software Diagnostics Publications ... 178

**Appendix ... 179**

My IT Reading List in 2006 ... 179

**Index of WinDbg Commands ... 189**

# Index of WinDbg Commands

!address, 9, 14
!analyze -v, 20, 92
!error, 18
!teb, 12, 25
.cxr, 16, 20, 23, 24, 91, 93
.ecxr, 20, 21, 29, 92
.exr, 16, 20, 21, 23, 29, 92
.for, 25
.reload, 10
.sympath, 9
~, 94
da, 13, 14
dp, 22
dpp, 13, 18
dps, 12, 14
dt, 13, 14, 18, 23
du, 18
k, 9
kc, 91, 93
kL, 10, 16, 21, 27, 29, 32
lm, 10
lmv, 31
u, 22
ub, 9, 31
x, 23

# Memory Dump Analysis Anthology

## Volume 15

**Dmitry Vostokov**
**Software Diagnostics Institute**

OpenTask

Published by OpenTask, Republic of Ireland

Copyright © 2023 by Dmitry Vostokov

Copyright © 2023 by Software Diagnostics Institute

All rights reserved. No part of this book may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, without the prior written permission of the publisher.

Product and company names mentioned in this book may be trademarks of their owners.

OpenTask books are available through booksellers and distributors worldwide. For further information or comments, send requests to press@opentask.com.

A CIP catalog record for this book is available from the British Library.

ISBN-13: 978-1-912636-15-0 (Paperback)

Revision 1.01 (April 2023)

# Table of Contents

**Preface** ........ 9

**Acknowledgments** ........ 10

**About the Author** ........ 11

**PART 1: Crash Dump Analysis Patterns** ........ 13

Shared Thread ........ 13
Execution Residue (Unmanaged Space, Kernel) ........ 14
Local Buffer Overflow (Kernel Space) ........ 27
Spiking Interrupts ........ 29
Structure Field Collection ........ 35
Stack Overflow (Managed Space) ........ 39
Black Box ........ 41
Rough Stack Trace Collection (Unmanaged Space) ........ 44
High Contention (Sockets) ........ 45
COM Object ........ 46
Shared Page ........ 48

**PART 2: Pattern Interaction** ........ 49

Exception Stack Trace, Stored Exception, Translated Exception, Execution Residue, Hidden Exception, NULL Pointer, Exception Module, Stack Trace Motif, No Component Symbols, and Coincidental Symbolic Information ........ 49

**PART 3: Software Trace Analysis Patterns** ........ 53

Message Essence ........ 53
Unsynchronized Messages ........ 54
Measurement ........ 56
Collapsed Message ........ 58
Trace Skeleton ........ 59
Null Reference ........ 61
Trace Braidoids ........ 63
Message Complex ........ 65

Traceme ........................................................................................................ 67
Trace Molecule ............................................................................................ 68
CoActivity ..................................................................................................... 70
Trace Window ............................................................................................... 72
Embedded Trace .......................................................................................... 73
Whisker Trace ............................................................................................... 74
Trace Nerve ................................................................................................... 75
Message Bond ............................................................................................... 76
Trace Retract ................................................................................................. 77
Denormalized Message ................................................................................ 78
Case Messages ............................................................................................... 79
Iconic Trace .................................................................................................... 80
Pattern Vector ................................................................................................ 82

**PART 4: Software Diagnostics, Root Cause Analysis, Troubleshooting, and Debugging ........................................................................................................ 83**

The Dream of Quantum Software Diagnostics ............................................. 83
Systematic Software Diagnostics .................................................................. 84
REPL Streaming (REPLS) .............................................................................. 87
Dia|gram Language and Memory Dump Analysis Patterns ......................... 89
Traces and Logs as 2-categories ................................................................... 95
Diagnostics of Artificial Intelligence ............................................................ 97
LaMDA ............................................................................................................ 98

**PART 5: Software Narratology ..................................................................... 99**

New Impetus .................................................................................................. 99

**PART 6: Fun with Debugging, Crash Dumps, and Traces ............................... 101**

Diagnostics and Debugging in Science Fiction ............................................ 101
Software Musical Scale ................................................................................ 102
The Spine Bug .............................................................................................. 102
Number Patterns .......................................................................................... 102
Prefixes ......................................................................................................... 102
Debugging Slang ........................................................................................... 103

*lav ................................................................................................................ 103*

*DRAMA* ........ *103*
*ESSE* ........ *103*
*Comical* ........ *103*
*VERSES from POEM* ........ *103*
*Communist* ........ *103*

Planck’s Numbers ........ 104
Pattern Recognition of Pattern Recognition ........ 105
Positions and Balances ........ 106
MS Word Blackout ........ 107
Diagnostics has an Error ........ 108
Variable Naming Convention ........ 108
Demon, Absolute Emperor, Wargamer. ........ 109
Beware of Mars Bugs Now ........ 110
Selling Dumps ........ 110
Solid WFH table for SOLID programming ........ 111
Metadata NULL Pointer ........ 111
Windows and Society ........ 112
Studying *bugs and Erro* ........ 113
Nature CSS Anomaly ........ 114
LP64 ........ 115
NaNonaire ........ 115
Bedtime ........ 116
Functional Programming ........ 116
Santa ........ 117
Basic Pointer Operations ........ 117
Surveillance ........ 118
Tips Programming Language ........ 118
Finest Data Structures ........ 119
Only Sundays ........ 120

**PART 7: Art and Photography ........ 121**

Defect Detect Logo ........ 121
Legacy Transport Containerized ........ 122
Clock Microservices ........ 122
Fuzzy Banknote ........ 123
Clustering ........ 123
Morning Traces ........ 124

Python is Buried Here ........ 124
Parallelism ........ 125
Three Generations ........ 125
Crios ........ 126
Memory Dumps ........ 127
The New Twitter ........ 128

**PART 8: Miscellaneous ........ 129**

WinDbg Notes ........ 129
Quotes ........ 130
My Favorite Category Theory Books (Continuation) ........ 132
Books That Influenced Me ........ 145
System Programmer and User Manual ........ 165
Running Linux in 2000 ........ 166
Data Science, AI, ML Books ........ 167
Logic Books ........ 172
Quantum Books ........ 175
Weekly IT (WIT) Books ........ 186
Other Books ........ 223
Algèbre d'Enfants (Children’s Algebra) ........ 266
X and Being vs. Being and X ........ 266
Color-enhanced Bookmarks ........ 267
Personalized Books with a Certificate ........ 267
Game Applets ........ 268
USSR Chemistry Olympiads ........ 269
My Longest Career ........ 278
Editing Memory Dump Analysis Anthology ........ 279
Window2Dump Use Case ........ 281
Massive Parallel Reading ........ 282
Lego Lambda Calculus ........ 282
GCD/LCM Software Engineering Principle ........ 283
Supervised ML (Music Listening) ........ 283
Proof of Concept Engineering Patterns ........ 284
Reflections on 2022 ........ 286
Training Categories and Functors ........ 288

**Appendix ........ 289**

Frame Patterns .......................................................................................... 289
CPU Consumption Patterns.................................................................................. 290

**Index of WinDbg Commands ..................................................................... 291**

# Index of WinDbg Commands

!analyze -v, 50, 129
!blackboxpnp, 41
!CLRStack, 39
!DumpIL, 40
!ext.help, 129
!findstack, 45
!for_each_thread, 35
!IP2MD, 39, 40
!locks, 129
!prcb, 29
!process, 13
!teb, 46, 50
!thread, 13, 14, 27
!whattime, 29
.cxr, 50, 51
.dumpdebug, 41
.exr, 49
.kframes, 39
.trap, 27
?, 20
dpp, 46
dps, 14, 20, 28, 46, 50
dt, 29
k, 51
k L=, 28
kc, 27, 49
kL, 39
uf, 52

# Memory Dump Analysis Anthology

# Volume 16

**Dmitry Vostokov**
**Software Diagnostics Institute**

OpenTask

Published by OpenTask, Republic of Ireland

Copyright © 2024 by Dmitry Vostokov

Copyright © 2024 by Software Diagnostics Institute

All rights reserved. No part of this book may be reproduced, stored in a retrieval system, or transmitted in any form or by any means without the publisher's prior written permission.

Product and company names mentioned in this book may be trademarks of their owners.

OpenTask books are available through booksellers and distributors worldwide. For further information or comments, send requests to press@opentask.com.

A CIP catalog record for this book is available from the British Library.

ISBN-13: 978-1-912636-16-7 (Paperback)

Revision 1.00 (April 2023)

# Table of Contents

**Preface** ........ 7

**About the Author** ........ 8

**PART 1: Crash Dump Analysis Patterns** ........ 9

Managed Code Exception (Scala) and Managed Stack Trace (Scala) ........ 9
Managed Code Exception (Python) and Managed Stack Trace (Python) ........ 10
Runtime Thread (Python, Linux) ........ 11
Exception Collection ........ 13
Dereference Nearpoint ........ 14
Address Representations ........ 15
Annotated Disassembly (Unmanaged Code) ........ 16
Near Exception ........ 17
Shadow Stack Trace ........ 20
Past Process ........ 22
Foreign Stack ........ 23
Annotated Stack Trace ........ 25
Disassembly Summary ........ 28
Region Summary ........ 29
Analysis Summary ........ 33
Region Spectrum ........ 38
Normalized Region ........ 39

**PART 2: Software Trace Analysis Patterns** ........ 41

Trace Tessellation ........ 41
Canonical Trace ........ 43
Trace Benchmark ........ 44
Trace Graph ........ 46
Trace Amplitude ........ 48
Trace Class ........ 49
Trace Volume ........ 51
Trace Temperature ........ 52
Trace Pressure ........ 53
Trace Lattice ........ 54

Trace Bias ........................................ 55
Trace Variance ........................................ 57
Message Kinetics........................................ 58

**PART 3: Software Diagnostics, Root Cause Analysis, Troubleshooting, and Debugging ........................................ 61**

Introducing Lov Language ........................................ 61
Carnot Cycle Metaphor for Trace and Log Analysis ........................................ 63
Defect Mechanism Patterns (DMP), Part 1 ........................................ 64
Attention ........................................ 65
ERROR ALERT! The World of Trace and Log Analysis ........................................ 66

**PART 4: Fun with Debugging, Crash Dumps, and Traces ........................................ 71**

Debugging Slang........................................ 71

*STEM ........................................ 71*
*WIT........................................ 71*
*LIT........................................ 71*

Software Development Dream ........................................ 72
Structures | Bugs | Effects | Debuggers ........................................ 73
Early Dump Analysis ........................................ 74

**PART 5: Art and Photography........................................ 75**

Classification anomaly ........................................ 75
Debuggers and Weather ........................................ 76
Cartoons from GPT-4 ........................................ 79
BSOD ........................................ 83
WinDbg Driver........................................ 84
Rust vs. C++........................................ 85
Moon........................................ 86

**PART 6: Miscellaneous ........................................ 87**

Quotes ........................................ 87
My Favorite Category Theory Books (Continuation) ........................................ 88

Books That Influenced Me ... 92
Data Science, AI, ML Books ... 107
Logic Books ... 115
Quantum Books ... 117
Weekly IT (WIT) Books ... 120
Other Books ... 130
Reflections on 2023 ... 146
Nonlinear Multiple Regression in 1994 ... 147
Complex Exploration ... 151
Discman ... 152
Voice Mouse Demo ... 153
Windows System Programming ... 154
Cats and Categories ... 155

*Localized Categories ... 155*
*Interest is Explainable ... 156*

Interviews as Discussions ... 157
Commutative Learning Diagram ... 158
Books as a Standing Desk ... 159
Software Factory in 1999 ... 160
At Home with Rational Unified Process ... 161
My Book on a Shelf ... 162
Irish Embassy, Moscow ... 163
Between Voltaire and Weber ... 165
Software Diagnostic Quackery ... 166
Intel Insides ... 167

**Appendix ... 169**

Region Patterns ... 169

**Index of WinDbg Commands ... 170**

# Index of WinDbg Commands

!analyze -v, 33
!ca, 22
!findthreads, 24
!heap -s -v, 13
!memusage, 22
!pe -nested, 13
!teb, 23
!Threads, 13
~*e, 13
~e, 23
dpp, 31
dps, 20, 23
dpS, 29, 31
dv, 15
k, 20
kL, 17, 18
r, 20
ub, 28
uf, 15, 16

# Memory Dump Analysis Anthology

## Volume 17

**Dmitry Vostokov**
**Software Diagnostics Institute**

OpenTask

Published by OpenTask, Republic of Ireland

Copyright © 2025 by Dmitry Vostokov

Copyright © 2025 by Software Diagnostics Institute

All rights reserved. No part of this book may be reproduced, stored in a retrieval system, or transmitted in any form or by any means without the publisher's prior written permission.

Product and company names mentioned in this book may be trademarks of their owners.

OpenTask books are available through booksellers and distributors worldwide. For further information or comments, send requests to press@opentask.com.

A CIP catalog record for this book is available from the British Library.

ISBN-13: 978-1-912636-17-4 (Paperback)

Revision 1.01 (November 2025)

# Table of Contents

**Preface ........ 9**

**About the Author ........ 10**

**PART 1: Crash Dump Analysis Patterns ........ 11**

Not My Version (Linux) ........ 11
Function Pointer (Linux) ........ 12
Missing Thread (Kernel Space) ........ 13
Interrupt Stack Collection ........ 15
DPC Stack Collection ........ 18
Dump Context ........ 26

**PART 2: Software Trace Analysis Patterns ........ 27**

Feynman Trace ........ 27
Trace Sketch ........ 29
Trace Model ........ 30
Message Type ........ 32
Implicit Message ........ 33
Trace Parameters and Hyperparameters ........ 34
Trace Precision and Recall ........ 36
Trace Polynomial ........ 38
Trace Embedding ........ 39
Trace Context ........ 40
Trace Plan ........ 41

**PART 3: Software Diagnostics, Root Cause Analysis, Troubleshooting, and Debugging ........ 43**

Declarative Memory ........ 43
Biographies of Software ........ 45
Situational Awareness in Pattern-Oriented Software Diagnostics, Forensics, Prognostics ........ 46
OS Internals and Category Theory ........ 49
From Software Diagnostics to Digital Pathology ........ 50

Logic of Pattern-Oriented Memory Dump Analysis ........ 51
Logic of Pattern-Oriented Trace and Log Analysis ........ 52
Pattern-Oriented System Logic ........ 53
Ontological Diagnostics ........ 54
Geometric Theory of Traces and Logs ........ 55
Traces and Logs as Pseudocode ........ 56
Reversing Traces and Logs and Generating Source Code ........ 57
Introducing Software Morphology ........ 61

**PART 4: Pattern-Oriented Observability ........ 65**

The Base Slab and Foundation of Observability Pillars ........ 65
Semiotics of Memory, Trace, and Log Analysis ........ 67
Observation Spaces ........ 69
Diagnostics – Observability Square (DOS) ........ 72
Books about Observability ........ 75

**PART 5: Pattern-Oriented AI ........ 77**

Definition ........ 77
Pattern-Oriented LLM Context Management ........ 78
Context Management Pattern Language ........ 81
Agentic Workflow Diagnostic Pattern Language ........ 85
Algebra of Prompting and Conversation with LLMs ........ 86
Software and Cloud Internals Analogies to LLMs and Agentic AI Systems ........ 87
Agentic Narratology ........ 88
Memoidealism as a Post-computational Philosophy of AI ........ 89
Introducing Pattern Metabolism ........ 90
The Stomach Metaphor for AI ........ 92
Pattern-Oriented Intelligence (POI) and WinDbg ........ 94
Pattern Category Theory ........ 96
Machine Learning as Memory Dump Analysis ........ 98

**PART 6: Fun with Debugging, Crash Dumps, and Traces ........ 101**

Diagnostics and Debugging in Science Fiction ........ 101
Debugging Slang ........ 102

*MAIN PAIN ........ 102*

*SF* ... *102*
*Peasant* ... *102*
*Orgasm* ... *102*
*LLM* ... *102*
*POETIC* ... *103*
*POETRY* ... *103*

Debugging Food ... 104
We are Bugs ... 104
DAMn IT ... 104
Can AI be Sick? ... 105
The Good Software Bugs ... 106
Three Categories of Time ... 107
Uncountable ... 108
Esenin and Debugging ... 109
It's Overloaded. I'm Overwhelmed ... 110
Less Exceptional C++ ... 111
Superstitions and Software Diagnostics ... 112
Questioning Error Message ... 113
Ireland's Most Prolific IT Book Author ... 113
Vostorg ... 114
ParadoxDbg ... 115
LLMous ... 116
Forensic Analysis ... 119
Gulliver’s Travels in Cyberspace and AI Land ... 120
AI Body Sickness ... 121
Halloween Influence ... 122
Pseudo-Scientific Abstract ... 123

**PART 7: Art and Photography ... 125**

Diagnostician’s Delights ... 125
Where Do You Want to Go Today? ... 126
Layers ... 127
COFFee ... 127
π ... 129
Picture of Myself ... 130
Work Environment ... 132
Impossible at Vikings’ Time ... 133

Russian History Underflow and Overflow ........ 134
Kazimir Malevich's Art ........ 136
Yet Another Mind Model ........ 137
From Bears to Sheep in 2001 ........ 138
It Got My Attention ........ 139

**PART 8: Miscellaneous........ 141**

Quotes........ 141
My Favorite Category Theory Books (Continuation) ........ 143
Books That Influenced Me ........ 162
Data Science, AI, ML Books ........ 174
Logic Books ........ 192
Quantum Books ........ 194
Weekly IT (WIT) Books ........ 196
Other Books ........ 207
When I was a Book Reviewer ........ 246
The Book That Inspired Me to Science ........ 248
Reflections on 2024 ........ 249
Linear Computers........ 250
My First Programming Language ........ 251
URSS Principle ........ 251
Treasure Box ........ 252
Irish Public Libraries ........ 253
InTouch ........ 254
First English Fiction Book ........ 255
My First Published Books........ 256
AI Big Bang ........ 257
Software Surgery ........ 258
Citrix, Karlsruhe, Germany ........ 259
One Hundred Years of Mausoleum........ 260
Cats and Rabbits ........ 261
A Software Engineer Reborn........ 262
My Old Reading Friend ........ 264
Book Collections as Categories: Another Reminder ........ 265
The First Book from Amazon........ 266
7 Books on a Train........ 267
The Selfish Genius........ 274
Debugged! MZ/PE Covers ........ 275

The Post-structure of Scientific Revolutions ... 285
Promotion Stack 2003 – 2008 ... 286
Propis' Manual ... 287
Handwritten Timestamp ... 290
Memory Visualization ... 291
Metabooks ... 295
Original Inception of MemOS ... 296
Memory Dump Analysis Services Anniversary ... 297
Debugging Press ... 298
Semantics and Vector Semantics ... 299
Attention to Books ... 300
Book Summaries ... 301
AI, 1993 ... 302
Speaking Mouse ... 303
Selected Conversations with ChatGPT ... 304

*Marr and the Universal Language of Embeddings ... 304*
*Fourier Analysis for Vibe Coding with Example Vibe Spectra ... 304*
*Classical Marxism and GenAI ... 304*
*Precision and Recall and Category Theory ... 304*
*A Lego Model as An Autoencoder ... 305*
*Lego Disassembly as Tokenization ... 305*
*Beware of Pseudo-LLMs ... 307*

Counterfactual CV ... 308
Philosophy Breakfasts ... 309

**PART 9: Autobiobibliography ... 311**

Terminology ... 311
2001 ... 311
2002 ... 312
2003 (January) ... 313
2003 (February) ... 314
2003 (March) ... 314
2003 (May) ... 315
2003 (June) ... 316

**Appendix ... 317**

Collection Patterns ... 317
New Social Media Groups ... 318

**Index of WinDbg Commands ... 320**

# Index of WinDbg Commands

!dpcs, 18
!idt, 15, 16
!ms_process, 13
!pcr, 15
!prcb, 18, 22
!thread, 13
~s, 15
dd, 18, 43
dps, 16, 17
dpS, 14, 17, 18, 22
dq, 43
dt, 16, 18, 22, 43
poi, 94

## Memory Analysis Patterns

| | |
|---|---|
| Abridged Dump | 5 |
| Accidental Lock | 1 |
| Activation Context | 6 |
| Active Space | 12 |
| Active Thread (Linux) | 9 |
| Active Thread (macOS) | 7 |
| Active Thread (Windows) | 9 |
| Activity Resonance | 7 |
| Address Representations | 16 |
| Affine Thread | 2 |
| Aggregated Frames | 10 |
| Analysis Summary | 16 |
| Annotated Disassembly (JIT .NET code) | 6 |
| Annotated Disassembly (Unmanaged Code) | 16 |
| Annotated Stack Trace | 16 |
| Black Box | 15 |
| Blocked DPC | 7 |
| Blocked Queue (LPC/ALPC) | 6 |
| Blocked Thread (hardware) | 5 |
| Blocked Thread (software) | 2 |
| Blocked Thread (timeout) | 7 |
| Blocking File | 6 |
| Blocking Module | 6 |
| Broken Link | 7 |
| Busy System | 1 |
| C++ Exception | 3 |
| C++ Exception (Linux) | 9 |
| C++ Exception (macOS) | 7 |
| C++ Object | 14 |
| COM Object | 15 |
| Caller-n-Callee | 6 |
| Changed Environment | 1 |
| Clone Dump | 9 |
| Cloud Environment | 6 |
| CLR Thread | 4 |
| Coincidental Error Code | 7 |

| | |
|---|---|
| Coincidental Frames | 5 |
| Coincidental Symbolic Information | 1 |
| Coincidental Symbolic Information (Linux) | 9 |
| Coincidental Symbolic Information (Mac OS  X) | 7 |
| COM Exception | 14 |
| Constant Subtrace | 9 |
| Context Pointer | 13 |
| Corrupt Dump | 2 |
| Corrupt Structure | 8 |
| Coupled Machines | 5 |
| Coupled Modules | 6 |
| Coupled Processes (semantics) | 5 |
| Coupled Processes (strong) | 1 |
| Coupled Processes (weak) | 5 |
| Crash Signature | 6 |
| Crash Signature Invariant | 7 |
| Crashed Process | 8 |
| Critical Region (Linux) | 9 |
| Critical Section Corruption | 2 |
| Critical Stack Trace | 7 |
| Custom Exception Handler (kernel space) | 6 |
| Custom Exception Handler (user space) | 1 |
| Data Alignment (page boundary) | 3 |
| Data Contents Locality | 2 |
| Data Correlation (CPU Times) | 11 |
| Data Correlation (function parameters) | 6 |
| Deadlock (critical sections) | 1 |
| Deadlock (executive resources) | 1 |
| Deadlock (.NET Finalizer) | `10 |
| Deadlock (LPC) | 1 |
| Deadlock (managed space) | 6 |
| Deadlock (mixed objects, kernel space) | 3 |
| Deadlock (mixed objects, user space) | 1 |
| Deadlock (self) | 5 |
| Debugger Bug | 7 |
| Debugger Omission | 7 |
| Dereference Nearpoint | 16 |
| Design Value | 8 |
| Deviant Module | 7 |

| | |
|---|---|
| Deviant Token | 7 |
| Diachronic Module | 9 |
| Dialog Box | 6 |
| Directing Module | 6 |
| Disassembly Ambiguity | 11 |
| Disassembly Summary | 16 |
| Disconnected Network Adapter | 7 |
| Disk Packet Buildup | 7 |
| Dispatch Level Spin | 2 |
| Distributed Exception (Managed Code) | 8 |
| Distributed Spike | 5 |
| Distributed Wait Chain | 7 |
| Divide by Zero (kernel mode) | 6 |
| Divide by Zero (user mode) | 3 |
| Divide by Zero (User Mode. Linux) | 9 |
| Divide by Zero (User Mode. macOS) | 7 |
| Double Free (kernel pool) | 1 |
| Double Free (process heap) | 1 |
| Double Free (Process Heap, macOS) | 7 |
| Double IRP Completion | 6 |
| DPC Stack Collection | 17 |
| Driver Device Collection (Malware) | 7 |
| Dry Weight | 8 |
| Dual Stack Trace | 6 |
| Dump Context | 17 |
| Duplicate Extension | 6 |
| Duplicated Module | 2 |
| Dynamic Memory Corruption (kernel pool) | 2 |
| Dynamic Memory Corruption (managed heap) | 6 |
| Dynamic Memory Corruption (process heap) | 1 |
| Dynamic Memory Corruption (Process Heap, Linux) | 9 |
| Dynamic Memory Corruption (Process Heap, macOS) | 7 |
| Early Crash Dump | 1 |
| Effect Component | 4 |
| Embedded Comments | 6 |
| Empty Stack Trace | 7 |
| Environment Hint | 6 |
| Error Reporting Fault | 7 |
| Evental Dumps | 9 |

| | |
|---|---|
| Exception Collection | 16 |
| Exception Module | 8 |
| Exception Stack Trace | 5 |
| Exception Reporting Thread | 11 |
| Execution Residue (Linux) | 9 |
| Execution Residue (macOS) | 7 |
| Execution Residue (managed space) | 6 |
| Execution Residue (Unmanaged Space, User) | 2 |
| Execution Residue (Unmanaged Space, Kernel) | 15 |
| Fake Module (Malware) | 7 |
| False Effective Address | 7 |
| False Frame | 14 |
| False Function Parameters | 2 |
| False Memory | 13 |
| False Positive Dump | 1 |
| Fat Process Dump | 6 |
| Fault Context | 5 |
| First Fault Stack Trace | 7 |
| Foreign Module Frame | 10 |
| Foreign Stack | 16 |
| FPU Exception | 7 |
| Frame Pointer Omission | 2 |
| Frame Regularity | 10 |
| Frame Trace | 13 |
| Frozen Process | 7 |
| Function Pointer (Linux) | 17 |
| Ghost Thread | 8 |
| Glued Stack Trace | 7 |
| Handle Leak | 7 |
| Handle Limit (GDI, Kernel Space) | 2 |
| Handle Limit (GDI, User Space) | 9 |
| Handled Exception (.NET CLR) | 6 |
| Handled Exception (kernel space) | 7 |
| Handled Exception (user space) | 6 |
| Hardware Activity | 5 |
| Hardware Error | 2 |
| Hidden Call | 8 |
| Hidden Exception (kernel space) | 7 |
| Hidden Exception (managed space) | 11 |

| | |
|---|---|
| Hidden Exception (user space) | 1 |
| Hidden IRP | 8 |
| Hidden Module | 2 |
| Hidden Parameter | 7 |
| Hidden Process | 7 |
| Hidden Stack | 13 |
| Hidden Stack Trace | 9 |
| High Contention (.NET Heap) | 10 |
| High Contention (.NET CLR monitors) | 7 |
| High Contention (critical sections) | 2 |
| High Contention (executive resources) | 1 |
| High Contention (processors) | 5 |
| High Contention (Sockets) | 15 |
| Historical Information | 1 |
| Hooked Functions (kernel space) | 5 |
| Hooked Functions (user space) | 1 |
| Hooked Modules | 2 |
| Hooking Level | 6 |
| Hyperdump | 11 |
| Incomplete  Stack Trace (macOS) | 7 |
| Incomplete Session | 7 |
| Inconsistent Dump | 1 |
| Incorrect Stack Trace | 1 |
| Incorrect Symbolic Information | 5 |
| Injected Symbols | 7 |
| Inline Function Optimization (managed code) | 6 |
| Inline Function Optimization (unmanaged code) | 2 |
| Instrumentation Information | 5 |
| Instrumentation Side Effect | 6 |
| Insufficient Memory (committed memory) | 1 |
| Insufficient Memory (control blocks) | 4 |
| Insufficient Memory (handle leak) | 1 |
| Insufficient Memory (kernel pool) | 1 |
| Insufficient Memory (module fragmentation) | 2 |
| Insufficient Memory (physical memory) | 3 |
| Insufficient Memory (PTE) | 2 |
| Insufficient Memory (region) | 8 |
| Insufficient Memory (reserved virtual memory) | 7 |
| Insufficient Memory (session pool) | 7 |

| | |
|---|---|
| Insufficient Memory (stack) | 11 |
| Insufficient Memory (stack trace database) | 8 |
| Internal Stack Trace | 8 |
| Interrupt Stack | 13 |
| Interrupt Stack Collection | 17 |
| Invalid Exception Information | 5 |
| Invalid Handle (general) | 2 |
| Invalid Handle (managed space) | 8 |
| Invalid Parameter (process heap) | 6 |
| Invalid Parameter (Runtime Function) | 10 |
| Invalid Pointer (general) | 1 |
| Invalid Pointer (Objects) | 13 |
| JIT Code (.NET) | 3 |
| JIT Code (Java) | 9 |
| Last Error Collection | 2 |
| Last Object | 8 |
| Late Crash Dump | 7 |
| Lateral Damage | 1 |
| Lateral Damage (CPU Mode) | 13 |
| Lateral Damage (Linux) | 9 |
| Least Common Frame | 6 |
| Livelock | 6 |
| Local Buffer Overflow | 1 |
| Local Buffer Overflow (Linux) | 9 |
| Local Buffer Overflow (macOS) | 7 |
| Local Buffer Overflow (Kernel Space) | 15 |
| Lost Opportunity | 2 |
| Main Thread | 1 |
| Managed Code Exception | 1 |
| Managed Stack Trace | 6 |
| Managed Stack Trace (Python) | 16 |
| Managed Stack Trace (Scala) | 16 |
| Managed Code Exception (Python) | 16 |
| Managed Code Exception (Scala) | 16 |
| Manual Dump (kernel) | 1 |
| Manual Dump (process) | 1 |
| Memory Fibration | 10 |
| Memory Fluctuation (process heap) | 8 |
| Memory Leak (.NET heap) | 1 |

| | |
|---|---|
| Memory Leak (I/O completion packets) | 7 |
| Memory Leak (page tables) | 7 |
| Memory Leak (process heap) - Modeling Example | 1 |
| Memory Leak (regions) | 8 |
| Message Box | 2 |
| Message Hooks - Modeling Example | 5 |
| Mirror Dump Set | 10 |
| Missing Component (general) | 2 |
| Missing Component (static linking, user mode) | 2 |
| Missing Process | 4 |
| Missing Thread (User Space) | 1 |
| Missing Thread (Kernel Space) | 17 |
| Mixed Exception | 4 |
| Module Collection | 7 |
| Module Collection (predicate) | 7 |
| Module Hint | 6 |
| Module Product Process | 7 |
| Module Stack Trace (Linux) | 10 |
| Module Stack Trace (Windows) | 10 |
| Module Variable | 7 |
| Module Variety | 1 |
| Multiple Exceptions (kernel mode) | 3 |
| Multiple Exceptions (macOS) | 7 |
| Multiple Exceptions (managed space) | 6 |
| Multiple Exceptions (Stowed) | 9 |
| Multiple Exceptions (user mode) - Modeling Example | 1 |
| Namespace (Malware) | 7 |
| Near Exception | 16 |
| Nested Exceptions (managed code) | 2 |
| Nested Exceptions (unmanaged code) | 2 |
| Nested Offender | 4 |
| Network Packet Buildup | 7 |
| No Component Symbols | 1 |
| No Current Thread | 7 |
| No Data Types | 6 |
| No Process Dumps | 2 |
| No System Dumps | 2 |
| Normalized Region | 16 |
| Not My Thread | 9 |

| | |
|---|---|
| Not My Version (hardware) | 4 |
| Not My Version (Linux) | 17 |
| Not My Version (software) | 2 |
| NULL Pointer (code) | 2 |
| NULL Pointer (Code, Linux) | 9 |
| NULL Pointer (Code, macOS) | 7 |
| NULL Pointer (data) | 3 |
| NULL Pointer (Data, Linux) | 9 |
| NULL Pointer (Data, macOS) | 7 |
| Object Distribution Anomaly (.NET Heap) | 9 |
| Object Distribution Anomaly (IRP) | 1 |
| OMAP Code Optimization | 1 |
| One-Thread Process | 7 |
| Optimized Code | 1 |
| Optimized VM Layout | 2 |
| Origin Module | 8 |
| Out-of-Module Pointer (Malware) | 7 |
| Overaged System | 2 |
| Packed Code (Malware) | 7 |
| Paged Out Data | 3 |
| Parameter Flow | 9 |
| Paratext | 7 |
| Paratext (Linux) | 9 |
| Pass Through Function | 3 |
| Passive System Thread (kernel space) | 1 |
| Passive Thread (user space) | 1 |
| Past Process | 16 |
| Past Stack Trace | 8 |
| Patched Code (Malware) | 7 |
| Pervasive System | 5 |
| Place Trace | 9 |
| Platform-Specific Debugger | 4 |
| Pleiades | 6 |
| Pointer Cone | 13 |
| Pointer Class | 13 |
| Pre-Obfuscation Residue (Malware) | 7 |
| Problem Exception Handler | 5 |
| Problem Module | 7 |
| Problem Vocabulary | 6 |

| | |
|---|---|
| Procedure Call Chain | 14 |
| Process Factory | 3 |
| Punctuated Memory Leak | 7 |
| Quotient Stack Trace | 10 |
| Quiet Dump | 6 |
| Random Object | 4 |
| Raw Pointer (Malware) | 7 |
| Reduced Symbolic Information | 7 |
| Reference Leak | 8 |
| Region Clusters | 12 |
| Region Profile | 12 |
| Region Spectrum | 16 |
| Region Summary | 16 |
| Regular Data | 7 |
| Relative Memory Leak | 9 |
| RIP Stack Trace (Malware) | 7 |
| Rough Stack Trace | 8 |
| Rough Stack Trace Collection (Unmanaged Space) | 15 |
| Rough Stack Trace (Managed Space) | 14 |
| Runtime Thread (Python, Linux) | 16 |
| Same Vendor | 5 |
| Saved Exception Context (.NET) | 14 |
| Screwbolt Wait Chain | 7 |
| Self-Diagnosis (kernel mode) | 6 |
| Self-Diagnosis (registry) | 7 |
| Self-Diagnosis (user mode) | 2 |
| Self-Dump | 2 |
| Semantic Split | 3 |
| Semantic Structure (PID.TID) | 6 |
| Shadow Stack Trace | 16 |
| Shared Buffer Overwrite | 5 |
| Shared Buffer Overwrite (macOS) | 7 |
| Shared Page | 15 |
| Shared Structure | 7 |
| Shared Thread | 15 |
| Small Value | 7 |
| Software Exception | 8 |
| Source Stack Trace | 12 |
| Special Process | 2 |

| | |
|---|---|
| Special Stack Trace | 1 |
| Special Thread (.NET CLR) | 6 |
| Spike Interval | 7 |
| Spiking Interrupts | 15 |
| Spiking Thread | 1 |
| Spiking Thread (Linux) | 9 |
| Spiking Thread (macOS) | 7 |
| Stack Overflow (insufficient memory) | 12 |
| Stack Overflow (kernel mode) | 1 |
| Stack Overflow (Managed Space) | 15 |
| Stack Overflow (software implementation) | 6 |
| Stack Overflow (user mode) | 2 |
| Stack Overflow (User Mode, Linux) | 9 |
| Stack Overflow (User Mode, macOS) | 7 |
| Stack Trace | 1 |
| Stack Trace (database) | 8 |
| Stack Trace (file system filters) | 8 |
| Stack Trace (I/O Devices) | 10 |
| Stack Trace (I/O request) | 8 |
| Stack Trace (Linux) | 9 |
| Stack Trace (macOS) | 7 |
| Stack Trace Change | 7 |
| Stack Trace Collection (CPUs) | 9 |
| Stack Trace Collection (I/O requests) | 7 |
| Stack Trace Collection (managed space) | 6 |
| Stack Trace Collection (predicate) | 7 |
| Stack Trace Collection (unmanaged space) | 1 |
| Stack Trace Motif | 10 |
| Stack Trace Race | 11 |
| Stack Trace Set | 6 |
| Stack Trace Signature | 9 |
| Stack Trace Surface | 9 |
| Step Dumps | 7 |
| Stored Exception | 7 |
| String Hint (Malware) | 7 |
| String Parameter | 6 |
| Structure Field Collection | 15 |
| Structure Sheaf | 14 |
| Subsystem Modules | 12 |

| | |
|---|---|
| Suspended Thread | 2 |
| Swarm of Shared Locks | 3 |
| System Call | 11 |
| System Object | 7 |
| Tampered Dump | 8 |
| Technology-Specific Subtrace (COM Client Call) | 8 |
| Technology-Specific Subtrace (COM interface invocation) | 6 |
| Technology-Specific Subtrace (dynamic memory) | 6 |
| Technology-Specific Subtrace (JIT .NET code) | 6 |
| Template Module | 5 |
| Thread Age | 6 |
| Thread Cluster | 7 |
| Thread Poset | 8 |
| Thread Starvation (normal priority) | 5 |
| Thread Starvation (realtime priority) | 2 |
| Top Module | 6 |
| Translated Exception | 7 |
| Truncated Dump | 1 |
| Truncated Dump (macOS) | 7 |
| Truncated Stack Trace | 6 |
| Ubiquitous Component (kernel space) | 7 |
| Ubiquitous Component (user space) | 4 |
| Unified Stack Trace | 10 |
| Unknown Component | 1 |
| Unloaded Module | 7 |
| Unrecognizable Symbolic Information | 7 |
| Unsynchronized Dumps | 6 |
| User Space Evidence | 8 |
| Value Adding Process | 7 |
| Value Deviation (stack trace) | 4 |
| Value Deviation (Structure Field) | 10 |
| Value References | 7 |
| Variable Subtrace | 8 |
| Version-Specific Extension | 6 |
| Virtualized Process (WOW64) | 1 |
| Virtualized System | 4 |
| Wait Chain (C++11, Condition Variable) | 9 |
| Wait Chain (CLR monitors) | 7 |
| Wait Chain (critical sections) | 1 |

| | |
|---|---|
| Wait Chain (executive resources) | 2 |
| Wait Chain (general) | 1 |
| Wait Chain (LPC/ALPC) | 3 |
| Wait Chain (modules) | 8 |
| Wait Chain (mutex objects) | 6 |
| Wait Chain (named pipes) | 6 |
| Wait Chain (Nonstandard Synchronization) | 9 |
| Wait Chain (process objects) | 5 |
| Wait Chain (pushlocks) | 7 |
| Wait Chain (RPC) | 5 |
| Wait Chain (RTL_RESOURCE) | 8 |
| Wait Chain (SRW Lock) | 10 |
| Wait Chain (thread objects) | 3 |
| Wait Chain (window messaging) | 6 |
| Waiting Thread Time (kernel dumps) | 1 |
| Waiting Thread Time (user dumps) | 2 |
| Well-Tested Function | 4 |
| Well-Tested Module | 6 |
| Wild Code | 2 |
| Wild Pointer | 2 |
| Window Hint | 9 |
| Young System | 2 |
| Zombie Processes | 2 |

## Trace and Log Analysis Patterns

| | |
|---|---|
| Abnormal Value | 7 |
| Activity Disruption | 8 |
| Activity Divergence | 7 |
| Activity Overlap | 8 |
| Activity Packet | 10 |
| Activity Quantum | 10 |
| Activity Region | 4 |
| Activity Theatre | 9 |
| Adjoint Message | 9 |
| Adjoint Space | 8 |
| Adjoint Thread of Activity | 5 |
| Anchor Messages | 5 |
| Back Trace | 8 |
| Background and Foreground Components | 5 |
| Basic Facts | 3 |
| Bifurcation Point | 4 |
| Blackout | 8 |
| Braid Group | 10 |
| Braid of Activity | 10 |
| Break-in Activity | 7 |
| Calibrating Trace | 9 |
| Canonical Trace | 16 |
| Cartesian Trace | 12 |
| Case Messages | 15 |
| Causal Chains | 13 |
| Causal History | 13 |
| Causal Messages | 13 |
| Characteristic Message Block | 4 |
| Circular Trace | 3 |
| CoActivity | 15 |
| Collapsed Message | 15 |
| Combed Trace | 10 |
| Container Trace | 14 |
| Cord of Activity | 14 |
| Correlated Discontinuity | 7 |
| Corrupt Message | 10 |

CoTrace (CoLog, CoData) 12
Counter Value 7
Coupled Activities 9
Critical Point 12
Data Association 7
Data Flow 7
Data Interval 9
Data Reversal 8
Data Selector 9
De Broglie Trace Duality 10
Declarative Trace 9
Defamiliarizing Effect 5
Defect Group 13
Definition Trace 14
Delay Dynamics 10
Denormalized Message 15
Density Distribution 7
Dialogue 7
Diegetic Messages 5
Discontinuity 4
Dominant Event Sequence 7
Drone Message 12
Embedded Trace 15
Empty Trace 7
Equivalent Messages 12
Error Distribution 7
Error Message 7
Error Powerset 9
Error Thread 7
Event Sequence Order 6
Event Sequence Phase 8
Exception Stack Trace 4
Explanation Trace 12
Factor Group 7
False Positive Error 5
Feature of Activity 14
Feynman Trace 17
Fiber Bundle 7
Fiber of Activity 9

| | |
|---|---|
| File Size | 8 |
| Flag | 14 |
| Focus of Tracing | 6 |
| Fourier Activity | 9 |
| Galois Trace | 10 |
| Generative Trace | 13 |
| Glued Activity (ATID reuse) | 6 |
| Gossip | 6 |
| Guest Component | 5 |
| Hedges | 11 |
| Hidden Error | 7 |
| Hidden Facts | 8 |
| Iconic Trace | 15 |
| Identification Messages | 9 |
| Implementation Discourse | 6 |
| Implicit Message | 17 |
| Impossible Trace | 6 |
| Incomplete History | 5 |
| Indexical Trace | 7 |
| Indirect Facts | 7 |
| Indirect Message | 8 |
| Inter-Correlation | 4 |
| Intra-Correlation | 3 |
| Intrinsic ID | 10 |
| Interspace | 8 |
| Last Activity | 7 |
| Layered Periodization | 5 |
| Linked Messages | 7 |
| Macrofunction | 7 |
| Marked Message | 7 |
| Master Trace | 6 |
| Measurement | 15 |
| Message Annotations | 12 |
| Message Bond | 15 |
| Message Change | 5 |
| Message Complex | 15 |
| Message Context | 7 |
| Message Cover | 7 |
| Message Directory | 10 |

| | |
|---|---|
| Message Essence | 15 |
| Message Flow | 12 |
| Message Interleave | 7 |
| Message Invariant | 6 |
| Message Kinetics | 16 |
| Message Metadata | 14 |
| Message Pattern | 9 |
| Message Set | 7 |
| Message Space | 8 |
| Message Type | 17 |
| Meta Trace | 7 |
| Milestones | 8 |
| Minimal Trace | 12 |
| Missing Component | 4 |
| Missing Data | 9 |
| Missing Message | 8 |
| Moduli Trace | 12 |
| Motif | 7 |
| Motivic Trace | 11 |
| Multidimensional Message | 14 |
| News Value | 6 |
| No Activity | 5 |
| No Trace Metafile | 5 |
| Null Reference | 15 |
| Opposition Messages | 7 |
| Original Message | 6 |
| Ornament | 10 |
| Palimpsest Messages | 8 |
| Pattern Vector | 15 |
| Periodic Error | 3 |
| Periodic Message Block | 7 |
| Phantom Activity | 11 |
| Phase Transition | 12 |
| Piecewise Activity | 7 |
| Pivot Message | 7 |
| Poincaré Trace | 10 |
| Polytrace | 12 |
| Projective Space | 10 |
| Punctuated Activity | 8 |

Quotient Trace 9
Random Data 11
Recovered Messages 8
Relative Density 7
Renormalization 11
Resume Activity 7
Ruptured Trace 9
Script Messages 11
Semantic Field 13
Sequence Repeat Anomaly 9
Serial Trace 14
Shared Point 7
Sheaf of Activities 7
Signal 11
Significant Event 5
Significant Interval 11
Silent Messages 7
Singleton Event 8
Singleton Trace 10
Small DA+TA 9
Sorted Trace 13
Sparse Trace 7
Split Message 12
Split Trace 7
State and Event 7
State Dump 7
Statement Density and Current 4
Strand of Activity 14
Surveyor 9
Tensor Trace 10
Text Trace 14
Thread of Activity 4
Time Delta 5
Time Scale 11
Timeout 8
Trace Acceleration 5
Trace Amplitude 16
Trace Benchmark 16
Trace Bias 16

Trace Braidoids 15
Trace Class 16
Trace Constants 12
Trace Context 17
Trace Contour 13
Trace D’Enfant 13
Trace Dimension 9
Trace Embedding 17
Trace Extension 9
Trace Fabric 13
Trace Field 11
Trace Flux 13
Trace Foliation 13
Trace Frames 7
Trace Graph 16
Trace Homotopy 11
Trace Join 14
Trace Lattice 16
Trace Mask 8
Trace Model 17
Trace Molecule 15
Trace Nerve 15
Trace Parameters and Hyperparameters 17
Trace Partition 5
Trace Path 13
Trace Plan 17
Trace Polynomial 17
Trace Precision and Recall 17
Trace Presheaf 10
Trace Pressure 16
Trace Quilt 14
Trace Retract 15
Trace Schema 14
Trace Shape 13
Trace Sharding 11
Trace Similarity 12
Trace Skeleton 15
Trace Sketch 17
Trace String 12

Trace Summary 13
Trace Temperature 16
Trace Tessellation 16
Trace Variance 16
Trace Viewpoints 8
Trace Volume 16
Trace Window 15
Traceme 15
Traces of Individuality 7
Translated Message 8
Truncated Data 11
Truncated Trace 5
Unsynchronized Messages 15
Unsynchronized Traces 10
UI Message 6
Ultrasimilar Messages 11
Use Case Trail 8
Visibility Limit 7
Visitor Trace 8
Vocabulary Index 4
Watch Thread 8
Weave of Activity 14
Whisker Trace 15
Working Set 11