

# CONTENTS IN DETAIL

## FOREWORD by John Baldwin

xiii

## INTRODUCTION

xv

What Is a Rootkit? .....	xvi
Why FreeBSD? .....	xvi
The Goals of This Book .....	xvi
Who Should Read This Book? .....	xvi
Contents Overview .....	xvi
Conventions Used in This Book .....	xvii
Concluding Remarks .....	xvii

## 1

### LOADABLE KERNEL MODULES

1

1.1 Module Event Handler .....	2
1.2 The DECLARE_MODULE Macro .....	3
1.3 "Hello, world!" .....	4
1.4 System Call Modules .....	6
1.4.1 The System Call Function .....	6
1.4.2 The sysent Structure .....	7
1.4.3 The Offset Value .....	8
1.4.4 The SYSCALL_MODULE Macro .....	8
1.4.5 Example .....	9
1.4.6 The modfind Function .....	10
1.4.7 The modstat Function .....	10
1.4.8 The syscall Function .....	11
1.4.9 Executing the System Call .....	11
1.4.10 Executing the System Call Without C Code .....	12
1.5 Kernel/User Space Transitions .....	12
1.5.1 The copyin and copyinstr Functions .....	13
1.5.2 The copyout Function .....	13
1.5.3 The copystr Function .....	13
1.6 Character Device Modules .....	14
1.6.1 The cdevsw Structure .....	14
1.6.2 Character Device Functions .....	15
1.6.3 The Device Registration Routine .....	16
1.6.4 Example .....	17
1.6.5 Testing the Character Device .....	19
1.7 Linker Files and Modules .....	21
1.8 Concluding Remarks .....	22

<b>2</b>		
<b>HOOKING</b>		<b>23</b>
2.1	Hooking a System Call .....	24
2.2	Keystroke Logging .....	26
2.3	Kernel Process Tracing .....	28
2.4	Common System Call Hooks .....	29
2.5	Communication Protocols .....	30
	2.5.1 The protosw Structure .....	30
	2.5.2 The inetsw[] Switch Table .....	31
	2.5.3 The mbuf Structure .....	32
2.6	Hooking a Communication Protocol .....	32
2.7	Concluding Remarks .....	35
<b>3</b>		
<b>DIRECT KERNEL OBJECT MANIPULATION</b>		<b>37</b>
3.1	Kernel Queue Data Structures .....	37
	3.1.1 The LIST_HEAD Macro .....	38
	3.1.2 The LIST_HEAD_INITIALIZER Macro .....	38
	3.1.3 The LIST_ENTRY Macro .....	38
	3.1.4 The LIST_FOREACH Macro .....	39
	3.1.5 The LIST_REMOVE Macro .....	39
3.2	Synchronization Issues .....	39
	3.2.1 The mtx_lock Function .....	40
	3.2.2 The mtx_unlock Function .....	40
	3.2.3 The sx_slock and sx_xlock Functions .....	40
	3.2.4 The sx_sunlock and sx_xunlock Functions .....	41
3.3	Hiding a Running Process .....	41
	3.3.1 The proc Structure .....	41
	3.3.2 The allproc List .....	42
	3.3.3 Example .....	43
3.4	Hiding a Running Process Redux .....	46
	3.4.1 The hashinit Function .....	47
	3.4.2 pidhashtbl .....	47
	3.4.3 The pfind Function .....	48
	3.4.4 Example .....	48
3.5	Hiding with DKOM .....	51
3.6	Hiding an Open TCP-based Port .....	52
	3.6.1 The inpcb Structure .....	52
	3.6.2 The tcbinfo.listhead List .....	53
	3.6.3 Example .....	54
3.7	Corrupting Kernel Data .....	56
3.8	Concluding Remarks .....	57
<b>4</b>		
<b>KERNEL OBJECT HOOKING</b>		<b>59</b>
4.1	Hooking a Character Device .....	59
	4.1.1 The cdevp_list and cdev_priv Structures .....	60
	4.1.2 The devmtx Mutex .....	60
	4.1.3 Example .....	60
4.2	Concluding Remarks .....	62

<b>5</b>	<b>RUN-TIME KERNEL MEMORY PATCHING</b>	<b>63</b>
5.1	Kernel Data Access Library .....	63
5.1.1	The kvm_openfiles Function .....	64
5.1.2	The kvm_nlist Function .....	64
5.1.3	The kvm_geterr Function .....	65
5.1.4	The kvm_read Function .....	65
5.1.5	The kvm_write Function .....	65
5.1.6	The kvm_close Function .....	66
5.2	Patching Code Bytes .....	66
5.3	Understanding x86 Call Statements .....	70
5.3.1	Patching Call Statements .....	70
5.4	Allocating Kernel Memory .....	73
5.4.1	The malloc Function .....	73
5.4.2	The MALLOC Macro .....	74
5.4.3	The free Function .....	74
5.4.4	The FREE Macro .....	74
5.4.5	Example .....	75
5.5	Allocating Kernel Memory from User Space .....	77
5.5.1	Example .....	77
5.6	Inline Function Hooking .....	81
5.6.1	Example .....	82
5.6.2	Gotchas .....	88
5.7	Cloaking System Call Hooks .....	88
5.8	Concluding Remarks .....	90
<b>6</b>	<b>PUTTING IT ALL TOGETHER</b>	<b>91</b>
6.1	What HIDses Do .....	91
6.2	Bypassing HIDses .....	92
6.3	Execution Redirection .....	92
6.4	File Hiding .....	96
6.5	Hiding a KLD .....	101
6.5.1	The linker_files List .....	102
6.5.2	The linker_file Structure .....	102
6.5.3	The modules List .....	103
6.5.4	The module Structure .....	103
6.5.5	Example .....	104
6.6	Preventing Access, Modification, and Change Time Updates .....	107
6.6.1	Change Time .....	108
6.6.2	Example .....	112
6.7	Proof of Concept: Faking Out Tripwire .....	114
6.8	Concluding Remarks .....	117
<b>7</b>	<b>DETECTION</b>	<b>119</b>
7.1	Detecting Call Hooks .....	120
7.1.1	Finding System Call Hooks .....	120

7.2	Detecting DKOM .....	123
7.2.1	Finding Hidden Processes .....	123
7.2.2	Finding Hidden Ports .....	125
7.3	Detecting Run-Time Kernel Memory Patching .....	125
7.3.1	Finding Inline Function Hooks .....	125
7.3.2	Finding Code Byte Patches .....	125
7.4	Concluding Remarks .....	126

**CLOSING WORDS** **127**

**BIBLIOGRAPHY** **129**

**INDEX** **131**