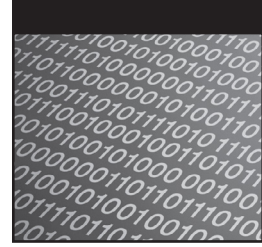


CRYPTOGRAPHY ENGINEERING

Design
Principles
and Practical
Applications

Niels Ferguson
Bruce Schneier
Tadayoshi Kohno

Cryptography Engineering



Cryptography Engineering

Design Principles and Practical Applications

Niels Ferguson
Bruce Schneier
Tadayoshi Kohno



WILEY

Wiley Publishing, Inc.

Cryptography Engineering: Design Principles and Practical Applications

Published by
Wiley Publishing, Inc.
10475 Crosspoint Boulevard
Indianapolis, IN 46256
www.wiley.com

Copyright © 2010 by Niels Ferguson, Bruce Schneier, and Tadayoshi Kohno

Published by Wiley Publishing, Inc., Indianapolis, Indiana

Published simultaneously in Canada

ISBN: 978-0-470-47424-2

Manufactured in the United States of America

10 9 8 7 6 5 4 3 2 1

No part of this publication may be reproduced, stored in a retrieval system or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, scanning or otherwise, except as permitted under Sections 107 or 108 of the 1976 United States Copyright Act, without either the prior written permission of the Publisher, or authorization through payment of the appropriate per-copy fee to the Copyright Clearance Center, 222 Rosewood Drive, Danvers, MA 01923, (978) 750-8400, fax (978) 646-8600. Requests to the Publisher for permission should be addressed to the Permissions Department, John Wiley & Sons, Inc., 111 River Street, Hoboken, NJ 07030, (201) 748-6011, fax (201) 748-6008, or online at <http://www.wiley.com/go/permissions>.

Limit of Liability/Disclaimer of Warranty: The publisher and the author make no representations or warranties with respect to the accuracy or completeness of the contents of this work and specifically disclaim all warranties, including without limitation warranties of fitness for a particular purpose. No warranty may be created or extended by sales or promotional materials. The advice and strategies contained herein may not be suitable for every situation. This work is sold with the understanding that the publisher is not engaged in rendering legal, accounting, or other professional services. If professional assistance is required, the services of a competent professional person should be sought. Neither the publisher nor the author shall be liable for damages arising herefrom. The fact that an organization or Web site is referred to in this work as a citation and/or a potential source of further information does not mean that the author or the publisher endorses the information the organization or Web site may provide or recommendations it may make. Further, readers should be aware that Internet Web sites listed in this work may have changed or disappeared between when this work was written and when it is read.

For general information on our other products and services please contact our Customer Care Department within the United States at (877) 762-2974, outside the United States at (317) 572-3993 or fax (317) 572-4002.

Wiley also publishes its books in a variety of electronic formats. Some content that appears in print may not be available in electronic books.

Library of Congress Control Number: 2010920648

Trademarks: Wiley and the Wiley logo are trademarks or registered trademarks of John Wiley & Sons, Inc. and/or its affiliates, in the United States and other countries, and may not be used without written permission. All other trademarks are the property of their respective owners. Wiley Publishing, Inc. is not associated with any product or vendor mentioned in this book.

To Denise, who has made me truly happy.

—Niels Ferguson

To Karen; still, after all these years.

—Bruce Schneier

To Taryn, for making everything possible.

—Tadayoshi Kohno



Credits

Executive Editor

Carol Long

Project Editor

Tom Dinse

Production Editor

Daniel Scribner

Editorial Director

Robyn B. Siesky

Editorial Manager

Mary Beth Wakefield

Production Manager

Tim Tate

Vice President and Executive

Group Publisher

Richard Swadley

Vice President and Executive

Publisher

Barry Pruett

Associate Publisher

Jim Minatel

Project Coordinator, Cover

Lynsey Stanford

Proofreader

Publication Services, Inc.

Indexer

Robert Swanson

Cover Image

© DSGpro/istockphoto

Cover Designer

Michael E. Trent



About the Authors

Niels Ferguson has spent his entire career working as a cryptographic engineer. After studying mathematics in Eindhoven, he worked for DigiCash analyzing, designing, and implementing advanced electronic payment systems that protect the privacy of the user. Later he worked as a cryptographic consultant for Counterpane and MacFergus, analyzing hundreds of systems and designing dozens. He was part of the team that designed the Twofish block cipher, performed some of the best initial analysis of AES, and co-designed the encryption system currently used by WiFi. Since 2004 he works at Microsoft where he helped design and implement the BitLocker disk encryption system. He currently works in the Windows cryptography team that is responsible for the cryptographic implementations in Windows and other Microsoft products.

Bruce Schneier is an internationally renowned security technologist, referred to by *The Economist* as a “security guru.” He is the author of eight books—including the best sellers *Beyond Fear: Thinking Sensibly about Security in an Uncertain World*, *Secrets and Lies*, and *Applied Cryptography*—as well as hundreds of articles and essays in national and international publications, and many more academic papers. His influential newsletter *Crypto-Gram*, and his blog *Schneier on Security*, are read by over 250,000 people. He is a frequent guest on television and radio, and is regularly quoted in the press on issues surrounding security and privacy. He has testified before Congress on multiple occasions, and has served on several government technical committees. Schneier is the Chief Security Technology Officer of BT.

Tadayoshi Kohno (Yoshi) is an assistant professor of computer science and engineering at the University of Washington. His research focuses on improving the security and privacy properties of current and future technologies. He conducted the initial security analysis of the Diebold AccuVote-TS electronic voting machine source code in 2003, and has since turned his attention to securing emerging technologies ranging from wireless implantable pacemakers and defibrillators to cloud computing. He is the recipient of a National Science Foundation CAREER Award and an Alfred P. Sloan Research Fellowship. In 2007 he was awarded the MIT Technology Review TR-35 Award for his work in applied cryptography, recognizing him as one of the world's top innovators under the age of 35. He received his PhD in computer science from the University of California at San Diego.

Niels, Bruce, and Yoshi are part of the team that designed the Skein hash function, one of the competitors in NIST's SHA-3 competition.



Acknowledgments for *Cryptography Engineering*

We are deeply indebted to the cryptography and security community at large. This book would not have been possible without all of their efforts in advancing the field. This book also reflects our knowledge and experience as cryptographers, and we are deeply grateful to our peers and mentors for helping shape our understanding of cryptography.

We thank Jon Callas, Ben Greenstein, Gordon Goetz, Alex Halderman, John Kelsey, Karl Koscher, Jack Lloyd, Gabriel Maganis, Theresa Portzer, Jesse Walker, Doug Whiting, Zooko Wilcox-O'Hearn, and Hussein Yaptit for providing invaluable feedback on earlier versions of this book.

Part of this book was developed and refined in an undergraduate computer security course at the University of Washington. We thank all those students, teaching assistants, and student mentors for the course. We especially thank Joshua Barr, Jonathan Beall, Iva Dermendjieva, Lisa Glendenning, Steven Myhre, Erik Turnquist, and Heather Underwood for providing specific comments and suggestions on the text.

We thank Melody Kadenko and Julie Svendsen for all their administrative support throughout this process. We are indebted to Beth Friedman for all her work copyediting this manuscript. Finally, we thank Carol Long, Tom Dinse, and the entire Wiley team for encouraging us to prepare this book and helping us all along the way.

We are also indebted to all the other wonderful people in our lives who worked silently behind the scenes to make this book possible.

Acknowledgments for *Practical Cryptography* (the 1st Edition)

This book is based on our collective experience over the many years we have worked in cryptography. We are heavily indebted to all the people we worked with. They made our work fun and helped us reach the insights that fill this book. We would also like to thank our customers, both for providing the funding that enabled us to continue our cryptography research and for providing the real-world experiences necessary to write this book.

Certain individuals deserve special mention. Beth Friedman conducted an invaluable copyediting job, and Denise Dick greatly improved our manuscript by proofreading it. John Kelsey provided valuable feedback on the cryptographic contents. And the Internet made our collaboration possible. We would also like to thank Carol Long and the rest of the team at Wiley for bringing our ideas to reality.

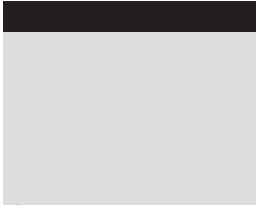
And finally, we would like to thank all of the programmers in the world who continue to write cryptographic code and make it available, free of charge, to the world.



Contents at a Glance

Preface to <i>Cryptography Engineering</i>	xxiii
Preface to <i>Practical Cryptography</i> (the 1st Edition)	xxvii
Part I Introduction	1
Chapter 1 The Context of Cryptography	3
Chapter 2 Introduction to Cryptography	23
Part II Message Security	41
Chapter 3 Block Ciphers	43
Chapter 4 Block Cipher Modes	63
Chapter 5 Hash Functions	77
Chapter 6 Message Authentication Codes	89
Chapter 7 The Secure Channel	99
Chapter 8 Implementation Issues (I)	115
Part III Key Negotiation	135
Chapter 9 Generating Randomness	137
Chapter 10 Primes	163
Chapter 11 Diffie-Hellman	181
Chapter 12 RSA	195

Chapter 13	Introduction to Cryptographic Protocols	213
Chapter 14	Key Negotiation	227
Chapter 15	Implementation Issues (II)	243
Part IV	Key Management	257
Chapter 16	The Clock	259
Chapter 17	Key Servers	269
Chapter 18	The Dream of PKI	275
Chapter 19	PKI Reality	281
Chapter 20	PKI Practicalities	295
Chapter 21	Storing Secrets	301
Part V	Miscellaneous	315
Chapter 22	Standards and Patents	317
Chapter 23	Involving Experts	323
	Bibliography	327
	Index	339



Contents

Preface to <i>Cryptography Engineering</i>	xxiii
History	xxiv
Example Syllabi	xxiv
Additional Information	xxvi
Preface to <i>Practical Cryptography</i> (the 1st Edition)	xxvii
How to Read this Book	xxix
Part I Introduction	1
Chapter 1 The Context of Cryptography	3
1.1 The Role of Cryptography	4
1.2 The Weakest Link Property	5
1.3 The Adversarial Setting	7
1.4 Professional Paranoia	8
1.4.1 Broader Benefits	9
1.4.2 Discussing Attacks	9
1.5 Threat Model	10
1.6 Cryptography Is Not the Solution	12
1.7 Cryptography Is Very Difficult	13
1.8 Cryptography Is the Easy Part	13
1.9 Generic Attacks	14
1.10 Security and Other Design Criteria	14
1.10.1 Security Versus Performance	14
1.10.2 Security Versus Features	17
1.10.3 Security Versus Evolving Systems	17

1.11	Further Reading	18
1.12	Exercises for Professional Paranoia	18
1.12.1	Current Event Exercises	19
1.12.2	Security Review Exercises	20
1.13	General Exercises	21
Chapter 2	Introduction to Cryptography	23
2.1	Encryption	23
2.1.1	Kerckhoffs' Principle	24
2.2	Authentication	25
2.3	Public-Key Encryption	27
2.4	Digital Signatures	29
2.5	PKI	29
2.6	Attacks	31
2.6.1	The Ciphertext-Only Model	31
2.6.2	The Known-Plaintext Model	31
2.6.3	The Chosen-Plaintext Model	32
2.6.4	The Chosen-Ciphertext Model	32
2.6.5	The Distinguishing Attack Goal	32
2.6.6	Other Types of Attack	33
2.7	Under the Hood	33
2.7.1	Birthday Attacks	33
2.7.2	Meet-in-the-Middle Attacks	34
2.8	Security Level	36
2.9	Performance	37
2.10	Complexity	37
2.11	Exercises	38
Part II	Message Security	41
Chapter 3	Block Ciphers	43
3.1	What Is a Block Cipher?	43
3.2	Types of Attack	44
3.3	The Ideal Block Cipher	46
3.4	Definition of Block Cipher Security	46
3.4.1	Parity of a Permutation	49
3.5	Real Block Ciphers	50
3.5.1	DES	51
3.5.2	AES	54
3.5.3	Serpent	56

3.5.4	Twofish	57
3.5.5	Other AES Finalists	58
3.5.6	Which Block Cipher Should I Choose?	59
3.5.7	What Key Size Should I Use?	60
3.6	Exercises	61
Chapter 4	Block Cipher Modes	63
4.1	Padding	64
4.2	ECB	65
4.3	CBC	65
4.3.1	Fixed IV	66
4.3.2	Counter IV	66
4.3.3	Random IV	66
4.3.4	Nonce-Generated IV	67
4.4	OFB	68
4.5	CTR	70
4.6	Combined Encryption and Authentication	71
4.7	Which Mode Should I Use?	71
4.8	Information Leakage	72
4.8.1	Chances of a Collision	73
4.8.2	How to Deal With Leakage	74
4.8.3	About Our Math	75
4.9	Exercises	75
Chapter 5	Hash Functions	77
5.1	Security of Hash Functions	78
5.2	Real Hash Functions	79
5.2.1	A Simple But Insecure Hash Function	80
5.2.2	MD5	81
5.2.3	SHA-1	82
5.2.4	SHA-224, SHA-256, SHA-384, and SHA-512	82
5.3	Weaknesses of Hash Functions	83
5.3.1	Length Extensions	83
5.3.2	Partial-Message Collision	84
5.4	Fixing the Weaknesses	84
5.4.1	Toward a Short-term Fix	85
5.4.2	A More Efficient Short-term Fix	85
5.4.3	Another Fix	87
5.5	Which Hash Function Should I Choose?	87
5.6	Exercises	87

Chapter 6	Message Authentication Codes	89
6.1	What a MAC Does	89
6.2	The Ideal MAC and MAC Security	90
6.3	CBC-MAC and CMAC	91
6.4	HMAC	93
6.5	GMAC	94
6.6	Which MAC to Choose?	95
6.7	Using a MAC	95
6.8	Exercises	97
Chapter 7	The Secure Channel	99
7.1	Properties of a Secure Channel	99
7.1.1	Roles	99
7.1.2	Key	100
7.1.3	Messages or Stream	100
7.1.4	Security Properties	101
7.2	Order of Authentication and Encryption	102
7.3	Designing a Secure Channel: Overview	104
7.3.1	Message Numbers	105
7.3.2	Authentication	106
7.3.3	Encryption	106
7.3.4	Frame Format	107
7.4	Design Details	107
7.4.1	Initialization	107
7.4.2	Sending a Message	108
7.4.3	Receiving a Message	109
7.4.4	Message Order	111
7.5	Alternatives	112
7.6	Exercises	113
Chapter 8	Implementation Issues (I)	115
8.1	Creating Correct Programs	116
8.1.1	Specifications	117
8.1.2	Test and Fix	118
8.1.3	Lax Attitude	119
8.1.4	So How Do We Proceed?	119
8.2	Creating Secure Software	120
8.3	Keeping Secrets	120
8.3.1	Wiping State	121
8.3.2	Swap File	122

	8.3.3	Caches	124
	8.3.4	Data Retention by Memory	125
	8.3.5	Access by Others	127
	8.3.6	Data Integrity	127
	8.3.7	What to Do	128
8.4		Quality of Code	128
	8.4.1	Simplicity	129
	8.4.2	Modularization	129
	8.4.3	Assertions	130
	8.4.4	Buffer Overflows	131
	8.4.5	Testing	131
8.5		Side-Channel Attacks	132
8.6		Beyond this Chapter	133
8.7		Exercises	133
Part III		Key Negotiation	135
Chapter 9		Generating Randomness	137
	9.1	Real Random	138
	9.1.1	Problems With Using Real Random Data	139
	9.1.2	Pseudorandom Data	140
	9.1.3	Real Random Data and PRNGs	140
	9.2	Attack Models for a PRNG	141
	9.3	Fortuna	142
	9.4	The Generator	143
	9.4.1	Initialization	145
	9.4.2	Reseed	145
	9.4.3	Generate Blocks	146
	9.4.4	Generate Random Data	146
	9.4.5	Generator Speed	147
	9.5	Accumulator	147
	9.5.1	Entropy Sources	147
	9.5.2	Pools	148
	9.5.3	Implementation Considerations	150
		9.5.3.1 Distribution of Events Over Pools	150
		9.5.3.2 Running Time of Event Passing	151
	9.5.4	Initialization	152
	9.5.5	Getting Random Data	153
	9.5.6	Add an Event	154
	9.6	Seed File Management	155
	9.6.1	Write Seed File	156

9.6.2	Update Seed File	156
9.6.3	When to Read and Write the Seed File	157
9.6.4	Backups and Virtual Machines	157
9.6.5	Atomicity of File System Updates	158
9.6.6	First Boot	158
9.7	Choosing Random Elements	159
9.8	Exercises	161
Chapter 10	Primes	163
10.1	Divisibility and Primes	163
10.2	Generating Small Primes	166
10.3	Computations Modulo a Prime	167
10.3.1	Addition and Subtraction	168
10.3.2	Multiplication	169
10.3.3	Groups and Finite Fields	169
10.3.4	The GCD Algorithm	170
10.3.5	The Extended Euclidean Algorithm	171
10.3.6	Working Modulo 2	172
10.4	Large Primes	173
10.4.1	Primality Testing	176
10.4.2	Evaluating Powers	178
10.5	Exercises	179
Chapter 11	Diffie-Hellman	181
11.1	Groups	182
11.2	Basic DH	183
11.3	Man in the Middle	184
11.4	Pitfalls	185
11.5	Safe Primes	186
11.6	Using a Smaller Subgroup	187
11.7	The Size of p	188
11.8	Practical Rules	190
11.9	What Can Go Wrong?	191
11.10	Exercises	193
Chapter 12	RSA	195
12.1	Introduction	195
12.2	The Chinese Remainder Theorem	196
12.2.1	Garner's Formula	196
12.2.2	Generalizations	197
12.2.3	Uses	198
12.2.4	Conclusion	199
12.3	Multiplication Modulo n	199

12.4	RSA Defined	200
12.4.1	Digital Signatures with RSA	200
12.4.2	Public Exponents	201
12.4.3	The Private Key	202
12.4.4	The Size of n	203
12.4.5	Generating RSA Keys	203
12.5	Pitfalls Using RSA	205
12.6	Encryption	206
12.7	Signatures	209
12.8	Exercises	211
Chapter 13	Introduction to Cryptographic Protocols	213
13.1	Roles	213
13.2	Trust	214
13.2.1	Risk	215
13.3	Incentive	215
13.4	Trust in Cryptographic Protocols	217
13.5	Messages and Steps	218
13.5.1	The Transport Layer	219
13.5.2	Protocol and Message Identity	219
13.5.3	Message Encoding and Parsing	220
13.5.4	Protocol Execution States	221
13.5.5	Errors	221
13.5.6	Replay and Retries	223
13.6	Exercises	225
Chapter 14	Key Negotiation	227
14.1	The Setting	227
14.2	A First Try	228
14.3	Protocols Live Forever	229
14.4	An Authentication Convention	230
14.5	A Second Attempt	231
14.6	A Third Attempt	232
14.7	The Final Protocol	233
14.8	Different Views of the Protocol	235
14.8.1	Alice's View	235
14.8.2	Bob's View	236
14.8.3	Attacker's View	236
14.8.4	Key Compromise	238
14.9	Computational Complexity of the Protocol	238
14.9.1	Optimization Tricks	239
14.10	Protocol Complexity	240

14.11	A Gentle Warning	241
14.12	Key Negotiation from a Password	241
14.13	Exercises	241
Chapter 15	Implementation Issues (II)	243
15.1	Large Integer Arithmetic	243
15.1.1	Wooping	245
15.1.2	Checking DH Computations	248
15.1.3	Checking RSA Encryption	248
15.1.4	Checking RSA Signatures	249
15.1.5	Conclusion	249
15.2	Faster Multiplication	249
15.3	Side-Channel Attacks	250
15.3.1	Countermeasures	251
15.4	Protocols	252
15.4.1	Protocols Over a Secure Channel	253
15.4.2	Receiving a Message	253
15.4.3	Timeouts	255
15.5	Exercises	255
Part IV	Key Management	257
Chapter 16	The Clock	259
16.1	Uses for a Clock	259
16.1.1	Expiration	259
16.1.2	Unique Value	260
16.1.3	Monotonicity	260
16.1.4	Real-Time Transactions	260
16.2	Using the Real-Time Clock Chip	261
16.3	Security Dangers	262
16.3.1	Setting the Clock Back	262
16.3.2	Stopping the Clock	262
16.3.3	Setting the Clock Forward	263
16.4	Creating a Reliable Clock	264
16.5	The Same-State Problem	265
16.6	Time	266
16.7	Closing Recommendations	267
16.8	Exercises	267
Chapter 17	Key Servers	269
17.1	Basics	270
17.2	Kerberos	270

17.3	Simpler Solutions	271
17.3.1	Secure Connection	272
17.3.2	Setting Up a Key	272
17.3.3	Rekeying	272
17.3.4	Other Properties	273
17.4	What to Choose	273
17.5	Exercises	274
Chapter 18	The Dream of PKI	275
18.1	A Very Short PKI Overview	275
18.2	PKI Examples	276
18.2.1	The Universal PKI	276
18.2.2	VPN Access	276
18.2.3	Electronic Banking	276
18.2.4	Refinery Sensors	277
18.2.5	Credit Card Organization	277
18.3	Additional Details	277
18.3.1	Multilevel Certificates	277
18.3.2	Expiration	278
18.3.3	Separate Registration Authority	279
18.4	Summary	280
18.5	Exercises	280
Chapter 19	PKI Reality	281
19.1	Names	281
19.2	Authority	283
19.3	Trust	284
19.4	Indirect Authorization	285
19.5	Direct Authorization	286
19.6	Credential Systems	286
19.7	The Modified Dream	288
19.8	Revocation	289
19.8.1	Revocation List	289
19.8.2	Fast Expiration	290
19.8.3	Online Certificate Verification	291
19.8.4	Revocation Is Required	291
19.9	So What Is a PKI Good For?	292
19.10	What to Choose	293
19.11	Exercises	294

Chapter 20	PKI Practicalities	295
20.1	Certificate Format	295
20.1.1	Permission Language	295
20.1.2	The Root Key	296
20.2	The Life of a Key	297
20.3	Why Keys Wear Out	298
20.4	Going Further	300
20.5	Exercises	300
Chapter 21	Storing Secrets	301
21.1	Disk	301
21.2	Human Memory	302
21.2.1	Salting and Stretching	304
21.3	Portable Storage	306
21.4	Secure Token	306
21.5	Secure UI	307
21.6	Biometrics	308
21.7	Single Sign-On	309
21.8	Risk of Loss	310
21.9	Secret Sharing	310
21.10	Wiping Secrets	311
21.10.1	Paper	311
21.10.2	Magnetic Storage	312
21.10.3	Solid-State Storage	313
21.11	Exercises	313
Part V	Miscellaneous	315
Chapter 22	Standards and Patents	317
22.1	Standards	317
22.1.1	The Standards Process	317
22.1.1.1	The Standard	319
22.1.1.2	Functionality	319
22.1.1.3	Security	320
22.1.2	SSL	320
22.1.3	AES: Standardization by Competition	321
22.2	Patents	322
Chapter 23	Involving Experts	323
Bibliography		327
Index		339



Preface to *Cryptography Engineering*

Most books cover what cryptography is—what current cryptographic designs are and how existing cryptographic protocols, like SSL/TLS, work. Bruce Schneier’s earlier book, *Applied Cryptography*, is like this. Such books serve as invaluable references for anyone working with cryptography. But such books are also one step removed from the needs of cryptography and security engineers in practice. Cryptography and security engineers need to know more than how current cryptographic protocols work; they need to know how to use cryptography.

To know how to use cryptography, one must learn to think like a cryptographer. This book is designed to help you achieve that goal. We do this through immersion. Rather than broadly discuss all the protocols one might encounter in cryptography, we dive deeply into the design and analysis of specific, concrete protocols. We walk you—hand-in-hand—through how we go about designing cryptographic protocols. We share with you the reasons we make certain design decisions over others, and point out potential pitfalls along the way.

By learning how to think like a cryptographer, you will also learn how to be a more intelligent user of cryptography. You will be able to look at existing cryptography toolkits, understand their core functionality, and know how to use them. You will also better understand the challenges involved with cryptography, and how to think about and overcome those challenges.

This book also serves as a gateway to learning about computer security. Computer security is, in many ways, a superset of cryptography. Both computer security and cryptography are about designing and evaluating objects (systems or algorithms) intended to behave in certain ways even in the presence

of an adversary. In this book, you will learn how to think about the adversary in the context of cryptography. Once you know how to think like adversaries, you can apply that mindset to the security of computer systems in general.

History

This book began with *Practical Cryptography* by Niels Ferguson and Bruce Schneier, and evolved with the addition of Tadayoshi Kohno—Yoshi—as an author. Yoshi is a professor of computer science and engineering at the University of Washington, and also a past colleague of Niels and Bruce. Yoshi took *Practical Cryptography* and revised it to be suitable for classroom use and self-study, while staying true to the goals and themes of Niels’s and Bruce’s original book.

Example Syllabi

There are numerous ways to read this book. You can use it as a self-study guide for applied cryptographic engineering, or you can use it in a course. A quarter- or semester-long course on computer security might use this book as the foundation for a 6-week intensive unit on cryptography. This book could also serve as the foundation for a full quarter- or semester-long course on cryptography, augmented with additional advanced material if time allows. To facilitate classroom use, we present several possible syllabi below.

The following syllabus is appropriate for a 6-week intensive unit on cryptography. For this 6-week unit, we assume that the contents of Chapter 1 are discussed separately, in the broader context of computer security in general.

- **Week 1:** Chapters 2, 3, and 4;
- **Week 2:** Chapters 5, 6, and 7;
- **Week 3:** Chapters 8, 9, and 10;
- **Week 4:** Chapters 11, 12, and 13;
- **Week 5:** Chapters 14, 15, 16, and 17;
- **Week 6:** Chapters 18, 19, 20, and 21.

The following syllabus is for a 10-week quarter on cryptography engineering.

- **Week 1:** Chapters 1 and 2;
- **Week 2:** Chapters 3 and 4;

- **Week 3:** Chapters 5 and 6;
- **Week 4:** Chapters 7 and 8;
- **Week 5:** Chapters 9 and 10;
- **Week 6:** Chapters 11 and 12;
- **Week 7:** Chapters 13 and 14;
- **Week 8:** Chapters 15, 16, and 17;
- **Week 9:** Chapters 18, 19, 20;
- **Week 10:** Chapter 21.

The following syllabus is appropriate for schools with 12-week semesters. It can also be augmented with advanced materials in cryptography or computer security for longer semesters.

- **Week 1:** Chapters 1 and 2;
- **Week 2:** Chapters 3 and 4;
- **Week 3:** Chapters 5 and 6;
- **Week 4:** Chapter 7;
- **Week 5:** Chapters 8 and 9;
- **Week 6:** Chapters 9 (continued) and 10;
- **Week 7:** Chapters 11 and 12;
- **Week 8:** Chapters 13 and 14;
- **Week 9:** Chapters 15 and 16;
- **Week 10:** Chapters 17 and 18;
- **Week 11:** Chapters 19 and 20;
- **Week 12:** Chapter 21.

This book has several types of exercises, and we encourage readers to complete as many of these exercises as possible. There are traditional exercises designed to test your understanding of the technical properties of cryptography. However, since our goal is to help you learn how to think about cryptography in real systems, we have also introduced a set of non-traditional exercises (see Section 1.12). Cryptography doesn't exist in isolation; rather, cryptography is only part of a larger ecosystem consisting of other hardware

and software systems, people, economics, ethics, cultural differences, politics, law, and so on. Our non-traditional exercises are explicitly designed to force you to think about cryptography in the context of real systems and the surrounding ecosystem. These exercises will provide you with an opportunity to directly apply the contents of this book as thought exercises to real systems. Moreover, by weaving these exercises together throughout this book, you will be able to see your knowledge grow as you progress from chapter to chapter.

Additional Information

While we strove to make this book as error-free as possible, errors have undoubtedly crept in. We maintain an online errata list for this book. The procedure for using this errata list is below.

- Before reading this book, go to <http://www.schneier.com/ce.html> and download the current list of corrections.
- If you find an error in the book, please check to see if it is already on the list.
- If it is not on the list, please alert us at cryptographyengineering@schneier.com. We will add the error to the list.

We wish you a wonderful journey through cryptography engineering. Cryptography is a wonderful and fascinating topic. We hope you learn a great deal from this book, and come to enjoy cryptography engineering as much as we do.

October 2009

Niels Ferguson
Redmond, Washington
USA
niels@ferguson.net

Bruce Schneier
Minneapolis, Minnesota
USA
schneier@schneier.com

Tadayoshi Kohno
Seattle, Washington
USA
yoshi@cs.washington.edu



Preface to *Practical Cryptography* (the 1st Edition)

In the past decade, cryptography has done more to damage the security of digital systems than it has to enhance it. Cryptography burst onto the world stage in the early 1990s as the securer of the Internet. Some saw cryptography as a great technological equalizer, a mathematical tool that would put the lowliest privacy-seeking individual on the same footing as the greatest national intelligence agencies. Some saw it as the weapon that would bring about the downfall of nations when governments lost the ability to police people in cyberspace. Others saw it as the perfect and terrifying tool of drug dealers, terrorists, and child pornographers, who would be able to communicate in perfect secrecy. Even those with more realistic attitudes imagined cryptography as a technology that would enable global commerce in this new online world.

Ten years later, none of this has come to pass. Despite the prevalence of cryptography, the Internet's national borders are more apparent than ever. The ability to detect and eavesdrop on criminal communications has more to do with politics and human resources than mathematics. Individuals still don't stand a chance against powerful and well-funded government agencies. And the rise of global commerce had nothing to do with the prevalence of cryptography.

For the most part, cryptography has done little more than give Internet users a false sense of security by promising security but not delivering it. And that's not good for anyone except the attackers.

The reasons for this have less to do with cryptography as a mathematical science, and much more to do with cryptography as an engineering discipline. We have developed, implemented, and fielded cryptographic systems over the

past decade. What we've been less effective at is converting the mathematical promise of cryptographic security into a reality of security. As it turns out, this is the hard part.

Too many engineers consider cryptography to be a sort of magic security dust that they can sprinkle over their hardware or software, and which will imbue those products with the mythical property of "security." Too many consumers read product claims like "encrypted" and believe in that same magic security dust. Reviewers are no better, comparing things like key lengths and on that basis, pronouncing one product to be more secure than another.

Security is only as strong as the weakest link, and the mathematics of cryptography is almost never the weakest link. The fundamentals of cryptography are important, but far more important is how those fundamentals are implemented and used. Arguing about whether a key should be 112 bits or 128 bits long is rather like pounding a huge stake into the ground and hoping the attacker runs right into it. You can argue whether the stake should be a mile or a mile-and-a-half high, but the attacker is simply going to walk around the stake. Security is a broad stockade: it's the things around the cryptography that make the cryptography effective.

The cryptographic books of the last decade have contributed to that aura of magic. Book after book extolled the virtues of, say, 112-bit triple-DES without saying much about how its keys should be generated or used. Book after book presented complicated protocols for this or that without any mention of the business and social constraints within which those protocols would have to work. Book after book explained cryptography as a pure mathematical ideal, unsullied by real-world constraints and realities. But it's exactly those real-world constraints and realities that mean the difference between the promise of cryptographic magic and the reality of digital security.

Practical Cryptography is also a book about cryptography, but it's a book about sullied cryptography. Our goal is to explicitly describe the real-world constraints and realities of cryptography, and to talk about how to engineer secure cryptographic systems. In some ways, this book is a sequel to Bruce Schneier's first book, *Applied Cryptography*, which was first published ten years ago. But while *Applied Cryptography* gives a broad overview of cryptography and the myriad possibilities cryptography can offer, this book is narrow and focused. We don't give you dozens of choices; we give you one option and tell you how to implement it correctly. *Applied Cryptography* displays the wondrous possibilities of cryptography as a mathematical science—what is possible and what is attainable; *Practical Cryptography* gives concrete advice to people who design and implement cryptographic systems.

Practical Cryptography is our attempt to bridge the gap between the promise of cryptography and the reality of cryptography. It's our attempt to teach engineers how to use cryptography to increase security.