Cryptography in C and C++

MICHAEL WELSCHENBACH
Translated by DAVID KRAMER

Apress™

Cryptography in C and C++

Copyright ©2005 by Michael Welschenbach

Translator and Compositor: David Kramer

Editorial Board: Steve Anglin, Dan Appleman, Ewan Buckingham, Gary Cornell, Tony Davis, Jason Gilmore, Jonathan Hassell, Chris Mills, Dominic Shakeshaft, Jim Sumser

Assistant Publisher: Grace Wong

Copy Manager: Nicole LeClerc

Production Manager: Kari Brooks-Copony

Proofreader: Anne Friedman

TEX Support: Fred Bartlett and Arthur Ogawa

Manufacturing Manager: Tom Debolski

Cover Designer: Kurt Krames

Library of Congress Cataloging-in-Publication Data
Welschenbach, Michael.

[Kryptographie in C und C++. English]
Cryptography in C and C++ / Michael Welschenbach ; translated by David Kramer.–
2nd American ed., rev. and enl.
p. cm.
The first American edition is a translation of the second German edition, which has
been revised and expanded from the first German edition.
Includes bibliographical references and index.
ISBN 1-59059-502-5
1. Computer security. 2. Cryptography. 3. C (Computer program
language) 4. C++ (Computer program language) I. Title.

QA76.9.A25W4313 2005
005.8–dc22 2005002553

All rights reserved. No part of this work may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or by any information storage or retrieval system, without the prior written permission of the copyright owner and the publisher.

Printed and bound in the United States of America 9 8 7 6 5 4 3 2 1

Trademarked names may appear in this book. Rather than use a trademark symbol with every occurrence of a trademarked name, we use the names only in an editorial fashion and to the benefit of the trademark owner, with no intention of infringement of the trademark.

Distributed to the book trade in the United States by Springer-Verlag New York, Inc., 233 Spring Street, 6th Floor, New York, NY 10013, and outside the United States by Springer-Verlag GmbH & Co. KG, Tiergartenstr. 17, 69112 Heidelberg, Germany.

In the United States: phone 1-800-SPRINGER, fax 201-348-4505, e-mail orders@springer-ny.com, or visit http://www.springer-ny.com. Outside the United States: fax +49 6221 345229, e-mail orders@springer.de, or visit http://www.springer.de. For information on translations, please contact Apress directly at 2560 Ninth Street, Suite 219, Berkeley, CA 94710. Phone 510-549-5930, fax 510-549-5939, e-mail info@apress.com, or visit http://www.apress.com.

The information in this book is distributed on an “as is” basis, without warranty. Although every precaution has been taken in the preparation of this work, neither the author(s) nor Apress shall have any liability to any person or entity with respect to any loss or damage caused or alleged to be caused directly or indirectly by the information contained in this work.

The source code for this book is available to readers at http://www.apress.com in the Downloads section. You will need to answer questions pertaining to this book in order to successfully download the code.

Contents

Foreword xiii

About the Author xv

About the Translator xvi

Preface to the Second American Edition xvii

Preface to the First American Edition xix

Preface to the First German Edition xxiii

I Arithmetic and Number Theory in C 1

1 Introduction 3

2 Number Formats: The Representation of Large Numbers in C 13

3 Interface Semantics 19

4 The Fundamental Operations 23

4.1 Addition and Subtraction . . . 24

4.2 Multiplication . . . 33

4.2.1 The Grade School Method . . . 34

4.2.2 Squaring Is Faster . . . 40

4.2.3 Do Things Go Better with Karatsuba? . . . 45

4.3 Division with Remainder . . . 50

5 Modular Arithmetic: Calculating with Residue Classes 67

6 Where All Roads Meet: Modular Exponentiation 81

6.1 First Approaches . . . 81

6.2 M-ary Exponentiation . . . 86

6.3 Addition Chains and Windows . . . 101

6.4 Montgomery Reduction and Exponentiation . . . 106

6.5 Cryptographic Application of Exponentiation . . . 118

7 Bitwise and Logical Functions 125

7.1 Shift Operations 125
7.2 All or Nothing: Bitwise Relations 131
7.3 Direct Access to Individual Binary Digits 137
7.4 Comparison Operators 140

8 Input, Output, Assignment, Conversion 145

9 Dynamic Registers 157

10 Basic Number-Theoretic Functions 167

10.1 Greatest Common Divisor 168
10.2 Multiplicative Inverse in Residue Class Rings 175
10.3 Roots and Logarithms 183
10.4 Square Roots in Residue Class Rings 191
10.4.1 The Jacobi Symbol 192
10.4.2 Square Roots Modulo p^k 198
10.4.3 Square Roots Modulo n 203
10.4.4 Cryptography with Quadratic Residues 211
10.5 A Primality Test 214

11 *Rijndael*: A Successor to the Data Encryption Standard 237

11.1 Arithmetic with Polynomials 239
11.2 The Rijndael Algorithm 244
11.3 Calculating the Round Key 247
11.4 The S-Box 248
11.5 The `ShiftRows`Transformation 249
11.6 The `MixColumns`Transformation 250
11.7 The `AddRoundKey`Step 252
11.8 Encryption as a Complete Process 253
11.9 Decryption 256
11.10 Performance 259
11.11 Modes of Operation 260

12 Large Random Numbers 261

12.1 A Simple Random Number Generator 265
12.2 Cryptographic Random Number Generators 268
12.2.1 The Generation of Start Values 269
12.2.2 The BBS Random Number Generator 273
12.2.3 The AES Generator 279
12.2.4 The RMDSHA-1 Generator 283

12.3 Quality Testing 286
12.3.1 Chi-Squared Test 287
12.3.2 Monobit Test 289
12.3.3 Poker Test 289
12.3.4 Runs Test 289
12.3.5 Longruns Test 289
12.3.6 Autocorrelation Test 290
12.3.7 Quality of the FLINT/C Random Number Generators . . . 290
12.4 More Complex Functions 291

13 Strategies for Testing LINT 305
13.1 Static Analysis 307
13.2 Run-Time Tests 309

II Arithmetic in C++ with the Class `LINT` 317

14 Let C++ Simplify Your Life 319
14.1 Not a Public Affair: The Representation of Numbers in `LINT` . . . 324
14.2 Constructors 325
14.3 Overloaded Operators 329

15 The `LINT`Public Interface: Members and Friends 337
15.1 Arithmetic 337
15.2 Number Theory 347
15.3 Stream I/O of `LINT`Objects 352
15.3.1 Formatted Output of `LINT`Objects 353
15.3.2 Manipulators 360
15.3.3 File I/O for `LINT`Objects 362

16 Error Handling 367
16.1 (Don't) Panic 367
16.2 User-Defined Error Handling 369
16.3 `LINT`Exceptions 370

17 An Application Example: The RSA Cryptosystem 377
17.1 Asymmetric Cryptosystems 378
17.2 The RSA Algorithm 380
17.3 Digital RSA Signatures 395
17.4 RSA Classes in C++ 403

18 Do It Yourself: Test LINT 413

19 Approaches for Further Extensions 417

III Appendices 419

A Directory of C Functions 421
A.1 Input/Output, Assignment, Conversions, Comparisons 421
A.2 Basic Calculations 422
A.3 Modular Arithmetic 423
A.4 Bitwise Operations 425
A.5 Number-Theoretic Functions 426
A.6 Generation of Pseudorandom Numbers 427
A.7 Register Management 431

B Directory of C++ Functions 433
B.1 Input/Output, Conversion, Comparison: Member Functions . . . 433
B.2 Input/Output, Conversion, Comparison: Friend Functions 436
B.3 Basic Operations: Member Functions 438
B.4 Basic Operations: Friend Functions 439
B.5 Modular Arithmetic: Member Functions 440
B.6 Modular Arithmetic: Friend Functions 442
B.7 Bitwise Operations: Member Functions 443
B.8 Bitwise Operations: Friend Functions 444
B.9 Number-Theoretic Member Functions 445
B.10 Number-Theoretic Friend Functions 446
B.11 Generation of Pseudorandom Numbers 450
B.12 Miscellaneous Functions 450

C Macros 451
C.1 Error Codes and Status Values 451
C.2 Additional Constants 451
C.3 Macros with Parameters 453

D Calculation Times 459

E Notation 461

F Arithmetic and Number-Theoretic Packages 463

References 465

Index 473