NETWORK SECURITY PRIVATE Communication in a PUBLIC World Second Edition This page is intentionally left blank NETWORK SECURITY PRIVATE Communication in a PUBLIC World CHARLIE KAUFMAN • RADIA PERLMAN • MIKE SPECINER Second Edition Authorized adaptation from the United States edition, entitled Network Security: Private Communication in a Public World, Second Edition, ISBN 978-01-304-6019-6, by Kaufman, Charlie; Perlman, Radia; Speciner, Mike; published by Pearson Education, Inc, ISBN 978-93-325-7821-0 e-ISBN 978-93-325-8600-0 First Impression This edition is manufactured in India and is authorized for sale only in India, Bangladesh, Bhutan, Pakistan, Nepal, Sri Lanka and the Maldives. Circulation of this edition outside of these territories is UNAUTHORIZED. Published by Pearson India Education Services Pvt. Ltd, CIN: U72200TN2005PTC057128. Formerly known as TutorVista Global Pvt. Ltd, licensee of Pearson Education in South Asia. Head Office: A-8 (A), 7th Floor, Knowledge Boulevard, Sector 62, Noida 201 309, Uttar Pradesh, India. Registered Office: 4th Floor, Software Block, Elnet Software City, TS-140, Block 2 & 9, Rajiv Gandhi Salai, Taramani, Chennai 600 113, Tamil Nadu, India. Fax: 080-30461003, Phone: 080-30461060 www.pearson.co.in, Email id: companysecretary.india@pearson.com Printed in India Si spy net work, big fedjaw iog link kyxogy This page is intentionally left blank vii CONTENTS

Glossary Index Online Chapters CHAPTER 23 Number Theory 641 CHAPTER 24 Math with AES and Elliptic Curves 653 CHAPTER 25 SSL/TLS 669 CHAPTER 26 Web Issues 691 CONTENTS xxviiii PREFACE It was a dark and stormy night. Somewhere in the distance a dog howled. A shiny object caught Alice's eye. A diamond cufflink! Only one person in the household could afford diamond cufflinks! So it was the butler, after all! Alice had to warn Bob. But how could she get a message to him without alerting the butler? If she phoned Bob, the butler might listen on an extension. If she sent a carrier pigeon out the window with the message taped to its foot, how would Bob know it was Alice that was sending the message and not Trudy attempting to frame the butler because he spurned her advances? That's what this book is about. Not much character development for Alice and Bob, we're afraid; nor do we really get to know the butler. But we do discuss how to communicate securely over an insecure medium. What do we mean by communicating securely? Alice should be able to send a message to Bob that only Bob can understand, even though Alice can't avoid having others see what she sends. When Bob receives a message, he should be able to know for certain that it was Alice who sent the message, and that nobody tampered with the contents of the message in the time between when Alice launched the message and Bob received it. What do we mean by an insecure medium? Well, in some dictionary or another, under the definition of "insecure medium" should be a picture of the Internet. The world is evolving towards interconnecting every computer, and people talk about connecting household appliances as well, all into some wonderful global internetwork. How wonderful! You'd be able to send electronic mail to anyone in the world. You'd also be able to control your nuclear power plant with simple commands sent across the network while you were vacationing in Fiji. Or sunny Libya. Or historic Iraq. Inside the network the world is scary. There are links that eavesdroppers can listen in on. Information needs to be forwarded through packet switches, and these switches can be reprogrammed to listen to or modify data in transit. The situation might seem hopeless, but we may yet be saved by the magic of mathematics, and in particular cryptography, which can take a message and transform it into a bunch of numbers known as ciphertext. The ciphertext is unintelligible gibberish except to someone who knows the secret to reversing the transformation. Cryptography allows us to disguise our data so that eavesdroppers gain no information from listening to the information as transmitted. Cryptography also allows us to create an unforgeable message and detect if it has been modified in transit. One method xxviii PREFACE of accomplishing this is with a digital signature, a number associated with a message and its sender that can be verified as authentic by others, but can only be generated by the sender. This should seem astonishing. How can there be a number which you can verify but not generate? A person's handwritten signature can (more or less) only be generated by that person, though it can be verified by others. But it would seem as if a number shouldn't be hard to generate, especially if it can be verified. Theoretically, you could generate someone's signature by trying lots of numbers and testing each one until one passed the verification test. But with the size of the numbers used, it would take too much compute time (for instance, several universe lifetimes) to generate the signature that way. So a digital signature has the same property as a handwritten signature, in that it can only be generated by one person. But a digital signature does more than a handwritten signature. Since the digital signature depends on the contents of the message, if someone alters the message the signature will no longer be correct and the tampering will be detected. This will all become clear if you read Chapter 2 Introduction to Cryptography. Cryptography is a major theme in this book, not because cryptography is intrinsically interesting (which it is), but because many of the security features people want in a computer network can best be provided through cryptography. ROADMAP TO THE BOOK After this introductory chapter, there are five main sections in the book: • Part 1

CRYPTOGRAPHY Chapter 2 Introduction to Cryptography is the only part of the cryptography section of the book essential for understanding the rest of the book, since it explains the generic properties of secret key, message digest, and public key algorithms, and how each is used. We've tried our best to make the descriptions of the actual cryptographic algorithms nonthreatening yet thorough, and to give intuition into why they work. It's intended to be readable by anyone, not just graduate students in mathematics. Never once do we use the term lemma. We do hope you read Chapter 3 Secret Key Cryptography, Chapter 4 Modes of Operation, Chapter 5 Hashes and Message Digests, and Chapter 6 Public Key Algorithms which give the details of the popular standards, but it's also OK to skip them and save them for later, or just for reference. • Part 2 AUTHENTICATION Chapter 7 Overview of Authentication Systems introduces the general issues involved in proving your identity across a network. Chapter 8 Authentication of People deals with the special circumstances when the device proving its identity is a PREFACE xxixix human being. Chapter 9 Security Handshake Pitfalls deals with the details of authentication handshakes. There are many security flaws that keep getting designed into protocols. This chapter attempts to describe variations of authentication handshakes and their relative security and performance strengths. We end the chapter with a checklist of security attacks so that someone designing a protocol can specifically check their protocol for these flaws. • Part 3 STANDARDS This portion of the book describes the standards: Kerberos versions 4 and 5, certificate and PKI standards, IPsec, and SSL. We hope that our descriptions will be much more readable than the standards themselves. And aside from just describing the standards, we give intuition behind the various choices, and criticisms where they are overly complex or have flaws. We hope that our commentary will make the descriptions more interesting and provide a deeper understanding of the design decisions. Our descriptions are not meant to, and cannot, replace reading the standards themselves, since the standards are subject to change. But we hope that after reading our description, it will be much easier to understand the standards. • Part 4 ELECTRONIC MAIL Chapter 17 Electronic Mail Security describes the various types of security features one might want, and how they might be provided. Chapter 18 PEM & S/MIME and Chapter 19 PGP (Pretty Good Privacy) describe the specifics of PEM, S/MIME, and PGP. • Part 5 LEFTOVERS Chapter 20 Firewalls talks about what firewalls are, what problems they solve, and what problems they do not solve. Chapter 21 More Security Systems, describes a variety of security systems, including Novell NetWare (Versions 3 and 4), Lotus Notes, DCE, KryptoKnight/NetSP, Clipper, SNMP, DASS/SPX, Microsoft (LAN Manager and Windows NT), and sabotage-proof routing protocols. We close with Chapter 22 Folklore, which describes the reasoning behind some of the advice you will hear from cryptographers. WHAT TYPE OF BOOK IS THIS? We believe the reason most computer science is hard to understand is because of jargon and irrelevant details. When people work with something long enough they invent their own language, come up with some meta-architectural framework or other, and forget that the rest of the world doesn't talk or think that way. We intend this book to be reader-friendly. We try to extract the concepts and xxx PREFACE ignore the meta-architectural framework, since whatever a meta-architectural framework is, it's irrelevant to what something does and how it works. We believe someone who is a relative novice to the field ought to be able to read this book. But readability doesn't mean lack of technical depth. We try to go beyond the information one might find in specifications. The goal is not just to describe exactly how the various standards and de facto standards work, but to explain why they are the way they are, why some protocols designed for similar purposes are different, and the implications of the design decisions. Sometimes engineering tradeoffs were made. Sometimes the designers could have made better choices (they are human after all), in which case we explain how the protocol could have been better. This analysis should make it easier to understand the current protocols, and aid in design of future protocols. The primary audience for this book is engineers, especially those who might need to evaluate the security of, or add security features to, a distributed system; but the book is also intended to be usable as a textbook, either on the advanced undergraduate or graduate level. Most of the chapters have homework problems at the end. Not all the chapters will be of interest to all readers. In some cases we describe and critique a standard in great detail. These chapters might not be of interest to students or people trying to get a conceptual understanding of the field. But in many cases the standards are written fairly unintelligibly. People who need to understand the standard, perhaps to implement it, or maybe even to use it, need to have a place where it is described in a readable way (and we strive for readability), but also a place in which mistakes in the standard are pointed out as such. It's very difficult to understand why, for instance, two fields are included which both give the same information. Sometimes it is because the designers of the protocol made a mistake. Once something like that is pointed out as a simple mistake, it's much easier to understand the specification. We hope that reading the descriptions in the book will make the specifications more intelligible. TERMINOLOGY Computer science is filled with ill-defined terminology used by different authors in conflicting ways, often by the same author in conflicting ways. We apologize in advance for probably being guilty sometimes ourselves. Some people take terminology very seriously, and once they start to use a certain word in a certain way, are extremely offended if the rest of the world does not follow. When I use a word, it means just what I choose it to mean—neither more nor less. —Humpty Dumpty (in Through the Looking Glass) PREFACE xxxi Some terminology we feel fairly strongly about. We do not use the term hacker to describe the vandals that break into computer systems. These criminals call themselves hackers, and that is how they got the name. But they do not deserve the name. True hackers are master programmers, incorruptibly honest, unmotivated by money, and careful not to harm anyone. The criminals termed "hackers" are not brilliant and accomplished. It is really too bad that they not only steal money, people's time, and worse, but they've also stolen a beautiful word that had been used to describe some remarkable and wonderful people. We instead use words like intruder, bad guy, and impostor. When we need a name for a bad guy, we usually choose Trudy (since it sounds like intruder). We grappled with the terms secret key and public key cryptography. Often in the security literature the terms symmetric and asymmetric are used instead of secret and public. We found the terms symmetric and asymmetric intimidating and sometimes confusing, so opted instead for secret key and public key. We occasionally regretted our decision to avoid the words symmetric and asymmetric when we found ourselves writing things like secret key based interchange keys rather than symmetric interchange keys. We use the term privacy when referring to the desire to keep communication from being seen by anyone other than the intended recipients. Some people in the security community avoid the term privacy because they feel its meaning has been corrupted to mean the right to know, because in some countries there are laws known as privacy laws which state that citizens have the right to see records kept about themselves. Privacy also tends to be used when referring to keeping personal information about people from being collected and misused. The security community also avoids the use of the word secrecy, because secret has special meaning within the military context, and they feel it would be confusing to talk about the secrecy of a message that was not actually labeled top secret or secret. The term most commonly used in the security community for keeping communication from being seen is confidentiality. We find that strange because confidential, like secret, is a security label, and the security community should have scorned use of confidential, too. In the first edition, we chose not to use confidentiality because we felt it had too many syllables, and saw no reason not to use privacy. For the second edition we reconsidered this decision, and were about to change all use of privacy to confidentiality until one of us pointed out we'd have to change the title of the book to something like Network Security: Confidential Communication in a Non-Confidential World, at which point we decided to stick with privacy. Speaker: Isn't it terrifying that on the Internet we have no privacy? Heckler1: You mean confidentiality. Get your terms straight. Heckler2: Why do security types insist on inventing their own language? Heckler3: It's a denial-of-service attack. —Overheard at recent gathering of security types We often refer to things involved in a conversation by name, for instance, Alice and Bob, whether the things are people or computers. This is a convenient way of making things unambigu- xxxiiii PREFACE ous with relatively few words, since the pronoun she can be used for Alice and he can be used for Bob. It also avoids lengthy inter- (and even intra-) author arguments about whether to use the politically incorrect he, a confusing she, an awkward he/she or (s)he, an ungrammatical they, an impersonal it, or an incredibly awkward rewriting to avoid the problem. We remain slightly worried that people will assume when we've named things with human names that we are referring to people. Assume Alice, Bob, and the rest of the gang may be computers unless we specifically say something like the user Alice, in which case we're talking about a human. With a name like yours, you might be any shape, almost. —Humpty Dumpty to Alice (in Through the Looking Glass) Occasionally, one of the three of us authors will want to make a personal comment. In that case we use I or me with a subscript. When it's a comment that we all agree with, or that we managed to slip past me3 (the rest of us are wimpier), we use the term we. NOTATION We use the symbol ⊕ (pronounced ex-or) for the bitwise-exclusive-or operation. We use the symbol | for concatenation. We denote secret key encryption with curly brackets preceded by the key with which something was encrypted, as in K{message}, which means message is secret key encrypted with K. Public key encryption we denote with curly braces, and the name of the owner of the public key subscripting the close brace, as in {message}Bob. Signing (which means using the private key), we denote with square brackets, with the name of the owner of the key subscripting the close bracket, as in [message]Bob. Table of Notation ⊕ bitwise exclusive or (pronounced ex-or) | concatenation (pronounced concatenated with) K{message} message encrypted with secret key K {message}Bob message encrypted with Bob's public key [message]Bob message signed with Bob's private key xxxiii ACKNOWLEDGMENTS Despite the controversies that crop up around security issues, it has been our experience that people in the security community are generally generous with their wisdom and time. It's always a little scary thanking specific people, for fear we'll leave someone out, but leaving everyone out seems wrong. It's not even the fair thing to do, since some people would be more egregiously wronged by being left out than others. Eric Rescorla and Hilarie Orman have been particularly helpful with answering questions and reviewing chapters for this edition. Other reviewers, and people who have been helpful answering questions, include Tom Wu, Kevin Fu, Marshall Rose, Joe Tardo, Joe Pato, Seth Proctor, Timothy Spiller, Tom Rice, Kristen McIntyre, Gary Winiger, Dan Harkins, Peter Memishian, Jeff Schiller, Burt Kaliski, Tony Lauck, Phil Karn, Ron Rivest, Steve Crocker, Steve Kent, John Linn, Steve Hanna, Jim Bidzos, Dave Jablon, Ted Ts'o, Matthew Barnes, Keith McCloughrie, Jeffrey Case, Kathrin Winkler, Philippe Auphelle, Sig Handelman, Phillip Hallam-Baker, Uri Blumenthal, Serge Vaudenay, and Boyd Roberts. We could not have done Chapter 21 More Security Systems without help from the various companies involved, since for the most part the security systems were previously undocumented. We'd like to thank Al Eldridge from Iris (Lotus Notes), Amir Herzberg and Mark Davis from IBM (KryptoKnight), Walt Tuvell from OSF, and Cliff Van Dyke from Microsoft (LAN Manager and Windows NT security) for explaining their systems to us, doing timely reviews of what we wrote, and being enthusiastic and supportive of the project. Although nearly 67% of us work for compa-nies that have products in this area, the opinions we

offer are ours alone, and not those of our com-panies. Mary Franz, our editor at Prentice Hall, has been enthusiastic and optimistic and patient with us throughout. She's shown good judgment about when to be helpful, when to keep out of the way, when to nag, and when to just look soulful so we feel guilty enough to meet a deadline. Despite the fact that this book has kept both of his parents busy for a significant part of his life, Ray Perlner has kept us inspired with his wholehearted and unselfish enthusiasm for the project. He's shown genuine interest in the subject matter, offered useful advice during interauthor arguments, helped search for quotes, reviewed part of the book, and particularly liked the subscripted pronouns. If we overdo those, it's just because it's fun to see him giggle. Dawn Perlner has also been a great supporter of the project, and manages to convince a surprising number of her friends, as well as complete strangers, to buy the book. xxxivxiv ACKNOWLEDGEMENTS And of course we thank you, our reader. We welcome your comments and suggestions. Compliments are always welcome. We hope to update the book periodically, so if there are topics you wish we'd covered or errors you'd like us to correct, let us know. Errata can be found at http://www.phptr.com/networksecurity. Our current email addresses are ckaufman@us.ibm.com, radia@alum.mit.edu, and ms@alum.mit.edu. We wish to thank the following for their permission to use their quotes in this book: • Quote on page 10 from The Hollywood Book of Quotes, Omnibus Press. • Quotes on page 17 and page 25 Copyright © 1994 Newsweek, Inc. All rights reserved. Reprinted by permission. • Quote on page 43 reprinted by permission of Singer Media Corporation. • Quote on page 19 reprinted by permission of Turner Entertainment. • Quote on page 117 courtesy of Donald Knuth. • Quote on page 253 reprinted by permission of The Wall Street Journal, Copyright © 1992 Dow Jones & Company, Inc. 1 1.1 PRIMER ON NETWORKING You have to know something about computer networks to understand computer network security, so we're including this primer. For a more detailed understanding, we recommend PERL99, TANE96, COME00, STEV94, KURO00. Networks today need to be very easy to use and configure. Networks are no longer an expensive educational toy for researchers, but instead are being used by real people. Most sites with networks will not be able to hire a full-time person with networking expertise to start and keep the network running. 1.1.1 OSI Reference Model Somehow, a book about computer networks would seem incomplete without a picture of the OSI (Open Systems Interconnection) Reference Model, so here it is. The OSI Reference Model is useful because it gives some commonly used terminology, though it might mislead you into thinking that there is only one way to construct a network. The reference model was designed by an organization known as the International Standards Organization (ISO). The ISO decided it would be a good idea to standardize computer networking. Since that was too big a task for a single committee, they decided to subdivide the problem among several committees. They somewhat arbitrarily chose seven, each responsible for one layer. The basic idea is that each layer uses the services of the layer below, adds functionality, and provides a service to the layer above. When you start looking at real networks, they seldom neatly fit into the seven-layer model, but for basic understanding of networking, the OSI Reference Model is a good place to start. 1. physical layer. This layer delivers an unstructured stream of bits across a link of some sort. 2. data link layer. This layer delivers a piece of information across a single link. It organizes the physical layer's bits into packets and controls who on a shared link gets each packet. application layer presentation layer session layer transport layer network layer data link layer physical layer Figure 1-1. OSI Reference Model INTRODUCTION 2 INTRODUCTION 1.1.2 3. network layer. This layer computes paths across an interconnected mesh of links and packet switches, and forwards packets over multiple links from source to destination. 4. transport layer. This layer establishes a reliable communication stream between a pair of systems across a network by putting sequence numbers in packets,