Malicious Cryptography

Exposing Cryptovirology

Adam Young

Moti Yung

WILEY

Wiley Publishing, Inc.

Malicious Cryptography

Malicious Cryptography

Exposing Cryptovirology

Adam Young

Moti Yung

WILEY

Wiley Publishing, Inc.

Executive Publisher: Robert Ipsen
Executive Editor: Carol A. Long
Developmental Editor: Eileen Bien Calabro
Editorial Manager: Kathryn A. Malm
Production Manager: Fred Bernardi

This book is printed on acid-free paper.

Published by Wiley Publishing, Inc., Indianapolis, Indiana
Published simultaneously in Canada

ISBN: 0-7645-4975-8

Printed in the United States of America

10 9 8 7 6 5 4 3 2 1

Dedicated to Elisa (A. Y.)
and to Maya (M. Y.)

Contents

Foreword **xiii**

Acknowledgments **xix**

Introduction **xxi**

1 Through Hacker's Eyes **1**

2 Cryptovirology **33**

3 Tools for Security and Insecurity **51**
3.1 Sources of Entropy 53
3.2 Entropy Extraction via Hashing 54
3.3 Unbiasing a Biased Coin 57
3.3.1 Von Neumann's Coin Flipping Algorithm 57
3.3.2 Iterating Neumann's Algorithm 59
3.3.3 Heuristic Bias Matching 60
3.4 Combining Weak Sources of Entropy 62
3.5 Pseudorandom Number Generators 66
3.5.1 Heuristic Pseudorandom Number Generation 66
3.5.2 PRNGs Based on Reduction Arguments 67
3.6 Uniform Sampling 68
3.7 Random Permutation Generation 71
3.7.1 Shuffling Cards by Repeated Sampling 71
3.7.2 Shuffling Cards Using Trotter-Johnson 73
3.8 Sound Approach to Random Number Generation and Use 76
3.9 RNGs Are the Beating Heart of System Security 77
3.10 Cryptovirology Benefits from General Advances 78
3.10.1 Strong Crypto Yields Strong Cryptoviruses 78
3.10.2 Mix Networks and Cryptovirus Extortion 80

3.11 Anonymizing Program Propagation 85

4 The Two Faces of Anonymity 89
4.1 Anonymity in a Digital Age 89
4.1.1 From Free Elections to the Unabomber 90
4.1.2 Electronic Money and Anonymous Payments 90
4.1.3 Anonymous Assassination Lotteries 92
4.1.4 Kidnapping and Perfect Crimes 93
4.1.5 Conducting Criminal Operations with Mixes 94
4.2 Deniable Password Snatching 97
4.2.1 Password Snatching and Security by Obscurity . . . 97
4.2.2 Solving the Problem Using Cryptovirology 98
4.2.3 Zero-Knowledge Proofs to the Rescue 100
4.2.4 Improving the Attack Using ElGamal 101

5 Cryptocounters 103
5.1 Overview of Cryptocounters 104
5.2 Implementing Cryptocounters 105
5.2.1 A Simple Counter Based on ElGamal 105
5.2.2 Drawback to the ElGamal Solution 106
5.2.3 Cryptocounter Based on Squaring 107
5.2.4 The Paillier Encryption Algorithm 108
5.2.5 A Simple Counter Based on Paillier 111
5.3 Other Approaches to Cryptocounters 111

6 Computationally Secure Information Stealing 113
6.1 Using Viruses to Steal Information 114
6.2 Private Information Retrieval 115
6.2.1 PIR Based on the Phi-Hiding Problem 117
6.2.2 Security of the Phi-Hiding PIR 120
6.2.3 Application of the Phi-Hiding Technique 122
6.3 A Variant of the Phi-Hiding Scheme 122
6.4 Tagged Private Information Retrieval 126
6.5 Secure Information Stealing Malware 131
6.6 Deniable Password Snatching Based on Phi-Hiding 132
6.6.1 Improved Password-Snatching Algorithm 133
6.6.2 Questionable Encryptions 134
6.6.3 Deniable Encryptions 139
6.7 Malware Loaders 140
6.8 Cryptographic Computing 141

7 Non-Zero Sum Games and Survivable Malware 147
7.1 Survivable Malware . . . 148
7.2 Elements of Game Theory . . . 150
7.3 Attacking a Brokerage Firm . . . 151
7.3.1 Assumptions for the Attack . . . 152
7.3.2 The Distributed Cryptoviral Attack . . . 153
7.3.3 Security of the Attack . . . 158
7.3.4 Utility of the Attack . . . 159
7.4 Other Two-Player Game Attacks . . . 161
7.4.1 Key Search via Facehuggers . . . 161
7.4.2 Catalyzing Conflict Among Hosts . . . 167
7.5 Future Possibilities . . . 167

8 Coping with Malicious Software 171
8.1 Undecidability of Virus Detection . . . 171
8.2 Virus Identification and Obfuscation . . . 172
8.2.1 Virus String Matching . . . 173
8.2.2 Polymorphic Viruses . . . 176
8.3 Heuristic Virus Detection . . . 182
8.3.1 Detecting Code Abnormalities . . . 182
8.3.2 Detecting Abnormal Program Behavior . . . 183
8.3.3 Detecting Cryptographic Code . . . 191
8.4 Change Detection . . . 197
8.4.1 Integrity Self-Checks . . . 197
8.4.2 Program Inoculation . . . 198
8.4.3 Kernel Based Signature Verification . . . 199

9 The Nature of Trojan Horses 201
9.1 Text Editor Trojan Horse . . . 202
9.2 Salami Slicing Attacks . . . 202
9.3 Thompson's Password Snatcher . . . 203
9.4 The Subtle Nature of Trojan Horses . . . 206
9.4.1 Bugs May In Fact Be Trojans . . . 208
9.4.2 RNG Biasing Trojan Horse . . . 208

10 Subliminal Channels 211
10.1 Brief History of Subliminal Channels . . . 212
10.2 The Difference Between a Subliminal and a Covert Channel 214
10.3 The Prisoner's Problem of Gustavus Simmons . . . 215
10.4 Subliminal Channels New and Old . . . 216

10.4.1 The Legendre Channel of Gus Simmons 217
10.4.2 The Oracle Channel 220
10.4.3 Subliminal Card Marking 222
10.4.4 The Newton Channel 223
10.4.5 Subliminal Channel in Composites 224
10.5 The Impact of Subliminal Channels on Key Escrow 226

11 SETUP Attack on Factoring Based Key Generation 229
11.1 Honest Composite Key Generation 231
11.2 Weak Backdoor Attacks on Composite Key Generation . . 232
11.2.1 Using a Fixed Prime 233
11.2.2 Using a Pseudorandom Function 234
11.2.3 Using a Pseudorandom Generator 236
11.3 Probabilistic Bias Removal Method 239
11.4 Secretly Embedded Trapdoors 241
11.5 Key Generation SETUP Attack 244
11.6 Security of the SETUP Attack 249
11.6.1 Indistinguishability of Outputs 249
11.6.2 Confidentiality of Outputs 252
11.7 Detecting the Attack in Code Reviews 256
11.8 Countering the SETUP Attack 259
11.9 Thinking Outside the Box 261
11.10 The Isaac Newton Institute Lecture 262

12 SETUP Attacks on Discrete-Log Cryptosystems 265
12.1 The Discrete-Log SETUP Primitive 266
12.2 Diffie-Hellman SETUP Attack 268
12.3 Security of the Diffie-Hellman SETUP Attack 270
12.3.1 Indistinguishability of Outputs 270
12.3.2 Confidentiality of Outputs 271
12.4 Intuition Behind the Attack 275
12.5 Kleptogram Attack Methodology 276
12.6 PKCS SETUP Attacks 277
12.6.1 ElGamal PKCS SETUP Attack 277
12.6.2 Cramer-Shoup PKCS SETUP Attack 279
12.7 SETUP Attacks on Digital Signature Algorithms 280
12.7.1 SETUP in the ElGamal Signature Algorithm 281
12.7.2 SETUP in the Pointcheval-Stern Algorithm 282
12.7.3 SETUP in DSA 283

12.7.4 SETUP in the Schnorr Signature Algorithm 284
12.8 Rogue Use of DSA for Encryption 285
12.9 Other Work in Kleptography 286
12.10 Should You Trust Your Smart Card? 288

Appendix A: Computer Virus Basics 295
A.1 Origins of Malicious Software 295
A.2 Trojans, Viruses, and Worms: What Is the Difference? . . 297
A.3 A Simple DOS COM Infector 299
A.4 Viruses Don't Have to Gain Control Before the Host . . . 303

Appendix B: Notation and Other Background Information 307
B.1 Notation Used Throughout the Book 307
B.2 Basic Facts from Number Theory and Algorithmics 309
B.3 Intractability: Malware's Biggest Ally 312
B.3.1 The Factoring Problem 313
B.3.2 The e^{th} Roots Problem 314
B.3.3 The Composite Residuosity Problem 314
B.3.4 The Decision Composite Residuosity Problem . . . 315
B.3.5 The Quadratic Residuosity Problem 315
B.3.6 The Phi-Hiding Problem 315
B.3.7 The Phi-Sampling Problem 317
B.3.8 The Discrete Logarithm Problem 318
B.3.9 The Computational Diffie-Hellman Problem 318
B.3.10 The Decision Diffie-Hellman Problem 318
B.4 Random Oracles and Functions 319

Appendix C: Public Key Cryptography in a Nutshell 321
C.1 Overview of Cryptography 321
C.1.1 Classical Cryptography 322
C.1.2 The Diffie-Hellman Key Exchange 324
C.1.3 Public Key Cryptography 325
C.1.4 Attacks on Cryptosystems 326
C.1.5 The Rabin Encryption Algorithm 330
C.1.6 The Rabin Signature Algorithm 331
C.1.7 The RSA Encryption Algorithm 332
C.1.8 The RSA Signature Algorithm 334
C.1.9 The Goldwasser-Micali Algorithm 335
C.1.10 Public Key Infrastructures 336
C.2 Discrete-Log Based Cryptosystems 337

C.2.1 The ElGamal Encryption Algorithm 338
C.2.2 Security of ElGamal 338
C.2.3 The Cramer-Shoup Encryption Algorithm 340
C.2.4 The ElGamal Signature Algorithm 342
C.2.5 The Pointcheval-Stern Signature Algorithm 343
C.2.6 The Schnorr Signature Algorithm 344
C.2.7 The Digital Signature Algorithm (DSA) 345

Glossary **347**

References **357**

Index **387**

Foreword

Terms such as cryptovirology, malware, kleptogram, or kleptography may be unfamiliar to the reader, but the basic concepts associated with them certainly are familiar. Everyone knows—often from sad experience—about viruses, Trojan horses, and worms and many have had a password "harvested" by a piece of software planted surreptitiously on their computer while browsing the Net. The realization that a public key could be placed in a virus so that part of its payload would be to perform a one-way operation on the host computer that could only be undone using the private key held by the virus' author was the discovery from which *Malicious Cryptography* sprang. Rather than describe these notions here, intriguing as they are, I'll only try to set the stage for the authors' lucid description of these and other related notions.

Superficially, information security, or information integrity, doesn't appear to be much different from other functions concerned with preserving the quality of information while in storage or during transmission. Error detecting and correcting codes, for example, are intended to ensure that the information that a receiver receives is the same as that sent by the transmitter. Authentication codes, or authentication in general, are also intended to ensure that information can neither be modified nor substituted without detection, thus allowing a receiver to be confident that what he receives is what was sent and that it came from the purported transmitter. These sound remarkably alike in function, but they are fundamentally different in ways that are at the heart of *Malicious Cryptography*. The greatest service this Foreword can render is to give the reader a crisp, clear understanding of the nature of this difference in order to set the stage for the book that follows.

Most system functions can be quantitatively specified and tested to verify that the specifications are met. If a piece of electronic equipment is supposed to operate within a specified range of a parameter (such as voltage, acceleration, temperature, shock, vibration, and so forth), then

it is a straightforward matter to devise tests to verify that it does. Closer in spirit to information security and integrity than physical environmental specifications would be a specification of a communication system's immunity to noise or bit errors. One might specify the minimum data bandwidth for a given signal to noise (SN) ratio or the allowable bit error rate. Again it is a straightforward matter to devise tests that verify the data bandwidth or the bit error rate for a signal possessing the specified signal to noise ratio. Error detecting and correcting codes may be tailored to the expected statistical nature of the noise, Fire codes for burst errors or Grey codes for an angular position reading device, etc. But the verification that the system is meeting specifications remains straightforward and quantitative.

Security is fundamentally different from any other system parameter, however. One of the largest alarm and vault manufacturers in the U.S. discovered this in a costly example a few years ago. Vaults and safes are routinely certified for the time documents will survive undamaged in a fire—itself specified by temperature and type (oil, structural, electrical, etc.). They had developed a new composite material that was very resistant to cutting, drilling, burning, etc. Extensive tests had been conducted with cutting tools of all sorts including oxyhydrogen burning bars, drilling with mechanical drills and hypervelocity air-abrasive drills, etc. Based on these results, they guaranteed their safes and vaults made of the new material would provide a specified minimum time for penetration. What they had overlooked was that linear cutting charges (shaped charges) that were widely used in the oil industry for cutting oil well casings and in the demolition business for slicing building supports to bring down buildings could be used to cut out a panel from the side of a safe or vault in milliseconds instead of requiring hours. This long aside is very germane to this Foreword. The safe and vault company had measured the resistance of their product to the attacks they anticipated would be used against them. The robbers used an entirely unexpected means to open the vault—and the company paid dearly for their oversight. *Malicious Cryptography* is almost entirely about doing things in completely unexpected ways in information integrity protocols.

Going back to the example with which we started, the fundamental difference between error detecting and correcting codes and authentication, both of which function to ensure the integrity of information, is that the first is pitted against nature and the other against a human adversary. Nature may be hostile, the signal to noise ratio may be large, the signal

may drop out for extended periods of time, other signals may randomly mask the desired signal, but nature is neither intelligent nor adaptive. A human opponent is both. He may also be interactive, probing to gain information to allow him to refine and adapt his attacks. As those of us in the information security business like to say, there is no standard attacker and no standard attack. This is in contrast with all other specifications where standard environments, no matter how hostile or unpredictable, are the norm.

What the authors of *Malicious Cryptography* have done very successfully is to capture the essence of how security can be subverted in this non-standard environment. On several occasions, they refer to game theory without actually invoking the formalism of game theory—emphasizing instead the game-like setting in which security is the value of the ongoing competition between a system designer and its attackers.

There have been many books on hacking, software subversion, network security, etc., which consist mainly of descriptions of successful attacks—some exceedingly clever and many very devious in their execution. These are similar in style and feeling to Modern Chess Openings (MCO) that every chess player knows, studies, and on which he depends. There are of course many possible lines of play in chess, but the several hundred openings that have stood the test of time and repeated tournament play make up the MCO. Roughly the first twenty moves or so of these openings, with promising variations, have been so thoroughly analyzed and understood that it is rare indeed for an opening not in the MCO to be successful in match play. A similar situation is true for the end game—not that the endings are so cataloged and restricted, but rather that the game has simplified to where almost a counting-like analysis reveals the outcome to a knowledgeable player. Masters will resign a game as lost at a point where a less experienced player may not even be able to see who has the advantage. As most books on hacking recount one clever attack after another, MCO recounts one opening after another with an ! or !! in the annotation to flag a particularly brilliant move. I almost expect to find an exclamation mark in the margin of most books on software subversion when the deception on which a particular protocol failure turns is revealed.

The middle game in chess, though, must be guided by general principles since the number of lines of play—the attack, counter attack—between two masters is virtually unlimited. So it is with information security protocols and cryptosystems. The possibilities are virtually unlimited so general principles must guide both the system designer and the counter

designer; the attacker seeking to exploit hidden weaknesses in the design; the designer seeking to prevent such attacks or failing that, to detect them when they occur. *Malicious Cryptography* pioneers in motivating and clearly enunciating some of these principles.

Cryptography, authentication, digital signatures, and indeed, virtually every digital information security function depend for their security on pieces of information known only to a select company of authorized insiders and unknown to outsiders. Following the usual convention in cryptography we will refer to this privileged information as the key although in many situations the only thing in common with the usual notion of a cryptographic key is that it is secret from all but a designated select few. It may well be that no individual knows the key but that a specified set of them have the joint capability to either recover it (shared secret schemes) or to jointly execute a function that in all probability no outsider or any proper subset of them can do (shared capability schemes). It is almost always the case that this secret piece of information is supposed to be chosen randomly—from a specified range of values and with a specified probability distribution, generally the uniform distribution. The assumption is that this insures that an unauthorized user will have no better chance of discerning the secret key than the probability the same key will be drawn in an independent drawing of a new value under the same conditions. It is also generally assumed that only the person choosing the random number knows it. In fact he may share it with someone else at the time it is drawn, or they may have chosen the number in advance of the supposed drawing. In the most extreme case it may be dictated by some other participant and not chosen by the person supposed to be choosing it all. Every one of these surreptitious variants has been the basis for serious subversions of information integrity and security protocols. One of the central themes in *Malicious Cryptography* is the mischief that is possible if these conditions are not met; in other words, if the "random" value is not random in the sense supposed.

Since security or integrity is directly measured by the probability the secret key can be discovered (computed) by unauthorized cabals of attackers, the information content of the key (roughly speaking, the size of the random number) must be great enough that it is computationally infeasible to simply try all possible values—known as a brute force key space search. But this means that it is then computationally infeasible for a monitor to tell whether the random values produced were actually randomly chosen as supposed or not. This is at the heart of subliminal

channels, for example. The subliminal transmitter and receiver share in secret information about the bias imposed on the selection of the session keys which enables them to communicate covertly in the overt communications while it remains computationally infeasible (impossible?) for a monitor to detect a bias in the session key selection process, and hence impossible for him to detect either the presence or use of a subliminal channel.

The dilemma is that if the key is large enough to be secure, it is also large enough to make it impossible to detect a bias in the selection process. It therefore becomes possible to hide information in the keys, to communicate other keys subliminally, to make it computationally feasible for designated receivers to perform a key space search while a full search remains computationally infeasible for outsiders to do, to subvert information integrity protocols from within, etc. The list of possible deceptions is virtually unlimited and the authors of *Malicious Cryptography* have exploited many of these in innovative ways.

In information integrity protocols nothing can be taken for granted, i.e., nothing can be assumed that cannot be enforced. If the protocol calls for a number to be chosen from a specified range using a particular probability distribution, then the assumption must be that it isn't unless the other parties to the protocol can force it to be in a secondary protocol. Otherwise you must assume it could be chosen from a restricted range or chosen using a different probability distribution, or that it was chosen earlier and shared with persons assumed not to know it, or that it isn't being selected at random at all by the person supposed to be choosing it, or that it is dictated to him by another party not even considered in the protocol. Several of the subversions described in *Malicious Cryptography* depend on this ability to undetectably hide information in keys. The point germane to this Foreword, though, is that it is the general principle that is vital for both the designer and the counter-designer to keep in mind. There are interactive protocols to insure that the objectives of randomness are met. Those protocols are not the subject of *Malicious Cryptography*, but made all the more important because of the weaknesses exposed in it.

There are other examples, though, in which no means is known to enforce the desired outcome. Several protocols call for a public modulus to be the product of two secret primes chosen so as to make it computationally infeasible to factor the modulus—usually only a function of the size of the factors although in some protocols the factors must satisfy some number theoretic side condition such as belonging to a particular residue

class, etc. It is possible to work a variety of mischiefs if a modulus that is the product of more than two prime factors can be passed off as the product of only two. In particular, a subliminal channel becomes possible with the desirable feature that while the subliminal receiver can receive subliminal messages sent by the transmitter he cannot falsely attribute a forged message to the transmitter. It is only polynomially difficult to distinguish between primes and composite numbers. But so far as is known it is just as hard to tell if a composite number has three or more factors as it is to factor the number itself! In the absence of an interactive protocol to ensure that a modulus has two and only two prime factors, deceptions that depend on the existence of three or more factors remain a possibility. Deceptions of this sort do not appear in *Malicious Cryptography* and are mentioned here only to illustrate that not all general principles for deception have solutions available to the designer at the moment.

Malicious Cryptography is a remarkable book; remarkable for what it attempts and remarkable for what it achieves. The realization that cryptography can be exploited to achieve malicious ends as easily as it can to achieve beneficial ones is a novel and valuable insight—to both designers and counter-designers of information security and integrity protocols.

Gus Simmons

September, 2003

Acknowledgments

We have so many people to thank that it is difficult to figure out where to begin. It has been said that ideas cannot be created in a vacuum and in this we believe wholeheartedly. *Malicious Cryptography* is the product of interactions and collaborations that span over a decade. In truth we have family, friends, teachers, coworkers, researchers, students, anonymous referees, journalists,[1] science-fiction authors, movie writers, artists, and musicians[2] to acknowledge. Without such support, enthusiasm, artistic creativity, teachers, and listeners, this book would not have been possible.

First and foremost we thank Columbia University, our mutual alma mater. It was at Columbia that our research began, and it was at Columbia where we met a great number of brilliant people from whom we learned, and with whom we worked and shared ideas. We thank Zvi Galil, Dean of the School of Engineering and Applied Science, who served as faculty advisor to us both. We thank Jonathan Gross and Andrew Kosoresow, both of whom served on Adam's PhD committee. Andrew was a great and dedicated educator, and we mourn his untimely passing. We thank Matt Franklin and Stuart Haber, both of whom graduated from Columbia. Matt and Stuart have served as collaborators to us both as well as lecturers in graduate courses taken by Adam. On numerous occasions Adam flew into Matt Franklin's office, wide-eyed and somewhat insane looking, for the sole purpose of scrawling a brand new attack on his blackboard just to see how he would react. Adam also thanks Matt Blaze for teaching an inspiring course on computer security in 1995 and for fostering great interest in cryptography among his students. Moti extends his gratitude to

[1] John Markoff, Steven Levy, Katie Hafner, and Bruce Sterling among others.

[2] Adam thanks Nine-Inch-Nails, Sonic Mayhem, White Zombie, Looking Glass Studios (System Shock 2 Soundtrack), Devo, and Danzig for setting the mood for the beginning of the book.

all of his coauthors and everyone he has worked with over the years, since it is through scientific work and the exchange of ideas that one develops as a researcher.

We thank Markus Jakobsson from RSA Data Security. Moti mentored Markus throughout his dissertation defense preparation and Markus in turn served on Adam's PhD committee. Markus reviewed this text and has sponsored annual lectures on Cryptovirology at NYU. We thank Yiannis Tsiounis, another student that Moti assisted, for sharing ideas and for reviewing this book. We thank our colleague Yair Frankel for sponsoring an invited lecture on kleptography for the Information Surety Group at Sandia National Labs. We thank Michael Reiter for supporting Adam while at Lucent Technologies in the Secure Systems Research Division, and for hosting a lecture on subliminal channels and kleptography.

Adam thanks Matthew Hastings from Los Alamos National Laboratory. Over the course of four years at Yale, Matt and Adam jointly experimented with self-replicating code in a safe and controlled environment. Many of the discoveries and open problems that were found gave impetus to investigating advanced malicious software attacks. Adam also thanks Mark Reed from the Yale University Department of Electrical Engineering. Mark served as Adam's undergraduate faculty advisor and provided support for his career both inside and outside of the classroom.

Adam thanks Cigital Labs and in particular Jeff Voas, Jeff Payne, Gary McGraw, and Matt Schmid for encouraging this work. We thank Christoph C. Michael, senior research scientist at Cigital Labs, for engaging conversations, contributing artwork, and for lending an ear to a never-ending stream of clandestine malware rhetoric. We also thank Alexander Antonov and Paul DesRivières from the Cigital Secure Software Group for reviewing the manuscript line by line and Mike Copenhafer, Bruce Potter, Mike Firetti, Viren Shah, Frank Hill, Coleman Baker, and Chris Ren from Cigital for helpful reviews and discussions.

From Wiley we thank Carol Long,[3] Eileen Calabro, Fred Bernardi, Robert Ipsen, and Kathryn Malm. Carol and her team produced this book in remarkably short order with the utmost degree of professionalism.

Special thanks goes to Dmitriy Pozdnyakov, Michael Makarius, Leo C. Petroski, and H. Robert Feinberg for helpful feedback and overall support of this work. Finally Adam would like to thank his wife, Elisa Young, for being. Without her this book would cease to have meaning.

[3]Or *cryptolady*, as she is known at Wiley.

Introduction

This book is a compendium of malicious software and hardware attacks geared towards subverting computer systems. The attacks are not of the sort that exploit software bugs, design flaws, and so forth. The business of bypassing security measures is outside the scope of this work. Rather, we present a series of cryptographic methods for defiling computer systems once internal access is acquired.

Some of the attacks are more technical than others, involving recent advances in the field of cryptology. As a result this book is likely to be received in a variety of different ways. To hackers it may serve as a vade mecum. To security professionals it may serve as a long overdue warning. To science fiction buffs it may serve as a good read, and to intelligence agencies it may serve as a challenge to our First Amendment rights.

Chapter 1 is a motivational chapter that portrays the world through the eyes of a hacker. It reveals the very fabric of a hacker's existence and due to its illicit nature we mention the standard disclaimer that reads, "do not try this at home." To perform any of the acts described therein is to risk violating the Computer Fraud and Abuse Act of 1986, among others. Hackers face *scientific* problems when trying to infiltrate computer systems. It was by experiencing these problems first hand that many of these attacks were discovered.

A great number of people share a close kinship with our digital brethren and to hackers it is no different. But whereas to writers it is through text, to artists it is through images, and to musicians it is through music, to hackers it is through the very language that computers speak when speaking with each other, the language of *binary*. To speak in binary and hear every word they say is to be one with the machine and that feeling can be hopelessly and utterly addictive.

To the uncorrupt of spirit the need to join with the machine can be controlled to a degree. This need is illustrated in Chapter 1 over the course of three short stories. They are written in second person singular

and as such force the reader to play the role of the subduer. It is the reader that steals passwords using a Trojan horse program. It is the reader that spends years developing an insidious computer virus, and it is the reader that takes over the local area network of a small company. Yet everywhere in the storyline the privacy and integrity of other people's data is respected. It portrays the pursuit of knowledge and the thrill of the hunt, not the kill.

As Lord Acton once said, "power corrupts; absolute power corrupts absolutely." This could not be truer with respect to hacking. For this reason we urge readers not to abuse the ideas presented in this book. If our efforts coax so much as a single hacker to embrace the greater mathematical challenges facing system security, then our writing will not have been for naught, for such a hacker is likely to seek recognition in the form of conference papers in lieu of news reports.

Given the clandestine nature of the algorithms and protocols that are presented, it is important to emphasize the nature of secure systems research. Cryptanalysis exists to help make cryptosystems more secure. *The goal of cryptanalysis is not to undo the honorable work of others, but to find vulnerabilities and fix them.* Many a cryptographer has suffered the disheartening realization that his or her cipher has been broken. Lucky are those who discover this themselves, but many are they who learn the hard way when another researcher publishes the discovery in an academic forum. Cryptanalysis is the mathematician's version of hacking: it is both devil's advocate and antithesis of cryptography. History has proven the need for cryptanalysis and hence the need to find weaknesses in cryptosystems and publish them. It may be reasoned that the need for cryptanalysis extends directly to the need to investigate attacks on modern computer systems. This, we argue, is the realm of cryptovirology and in this treatise we take a first step in this direction.

In the public eye, the word *cryptography* is virtually synonymous with *security*. It is a means to an end, a way to send e-mail privately and purchase items securely on-line. If nothing else this book will challenge that view. In the chapters that follow it is shown how modern cryptographic paradigms and tools including semantic security, reduction arguments, polynomial indistinguishability, random oracles, one-way functions, Feistel ciphers, entropy extractors, pseudorandom number generators, etc., can in fact be used to *degrade system security*.

It is shown how to devise a cryptovirus to usurp data from a host machine without revealing that which is sought, even if the virus is observed

at every turn. It is shown how to design a password-snatching cryptotrojan that makes it virtually impossible to identify the author when the encrypted passwords are retrieved. Furthermore, it is intractable to determine if the cryptotrojan is encrypting anything at all even when it is under constant surveillance.

Still other cryptotrojans are described that attack industry-standard cryptosystems. By design, these Trojans give the attacker covert access to the private keys of users and are extremely robust against reverse engineering. When implemented in tamper-resistant devices these transgressions cannot be detected by anyone save the attacker. Such Trojans are ideal for governments that wish to obtain covert access to the encrypted communications of their citizens. These Trojans show how to apply cryptography within cryptography itself to undermine the very trust that cryptosystems were designed to provide. In so doing we will expose the dark side of cryptography and thereby reveal its true dual-edged nature.

Several of the attacks have known countermeasures, some of which are ideal and others that are merely heuristic in nature. These defenses are described in detail to give the book a more balanced presentation to the community at large. It is our belief that these malicious software attacks should be exposed so that security analysts will recognize them in the event that they appear in fielded computer systems. Doing so has the potential of minimizing the malicious software learning curve that practitioners might otherwise face.

In all likelihood the attacks that are described in this book constitute the tip of the iceberg in terms of what is possible. Offensive information warfare is an area of research that is scarcely funded by the U.S. government, for obvious reasons. However, the notion of malicious software as well as cryptography is by no means new to the federal government, and so one would expect that there has been more classified research in this area than unclassified research. This book is our earnest attempt to expose the open research in this area, since corporations, governments, and individuals have a right to know about that which threatens the integrity of their computing machinery.

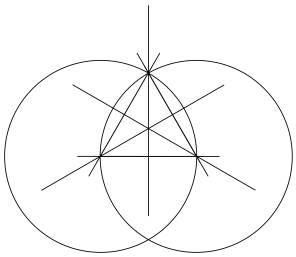
Some readers will inevitably object to the nature of this book. To this end we remark that these attacks exist, they are real, and that it is perilous to sweep them under the rug. We believe that they will surface sooner or later. It is our hope that this book will encourage the study of cryptography as a whole and at the same time reveal some of the more

serious threats that computer systems face, both *from within* and from without.

A. Y.

M. Y.

October, 2003



Chapter 1

Through Hacker's Eyes

There is no way to describe the feeling of approaching a computer system to download the data that your Trojan horse has been collecting for days. Your heart begins to race. You look over your shoulder out of instinct and start to have major second thoughts about proceeding. The computer terminal is unoccupied and sits directly in front of you.

Questions plague your thoughts: How many people are capable of finding the cleverly hidden Trojan? More importantly, does anyone *in this room* know it is there? You ease yourself down into the chair. Glancing to your right you see a student stare at his calculator with a perplexed look on his face. To your left a girl is laughing on her cell phone. If you could shrink yourself into nothing and crawl through the cracks in the machine you would gladly do so. But you are physical and there is nothing you can do about that now. The coast is clear. You reach for your floppy and insert it into the drive. Sheens of sweat glaze over your palms. Why? Because after all, you are returning to the scene of a crime.

Your crime.

Deep down, you rationalize your actions. There is no blood involved, no money is being stolen, and in the end no real harm is being done... or is there? The floppy drive begins to spin. In moments it will be over. In moments all of the login/password pairs will be on the disk and you will be hightailing it to your next class. Perspiration breaks out on your forehead but is easily dismissed with a waft of your hand. You navigate to the floppy drive and double-click on the game of Tetris. There is time for one quick game. The first block is 1 by 5, your favorite. If only they'd come down like that one after the other you'd have the game in the bag by laying them out horizontally. But it never works out that way. The law of probabilities won't allow it. A book hits the ground and you jump.

A lanky-looking freshman picks it up. The title—*Differential Equations: Theory and Applications*. Smart guy. Most students are only studying multivariable calculus in their first year. Words begin to echo in the back of your mind: *there has to be a better way, there has to be a better way....* An odd, misshapen block comes into view. You hate those. They make you lose Tetris every time. A whirring noise emanates from the drive and this time you know it is writing to the floppy. One more minute and your doctored up version of Tetris will have downloaded all of the passwords to the disk. Who'd ever guess this version of Tetris packed such a punch?

A four-sided cube comes down and you ease it over to the left-hand side of the screen. You love those shapes too. On the surface you are just playing a game. Your mouse button clicks and space bar presses are as innocuous as they come. But the real game you are playing is not so easy to see, and at times it feels like Russian roulette. Your thoughts wander to your password-snatching Trojan. The possibility that it was found and that silent sysadmin alarms are sounding *in a nearby room* is very, very real.

Something's wrong.

Something's not right; you can feel it in the air. The drive should have stopped spinning by now. Your heart goes still. Looking up, you catch a glimpse of a man you didn't notice before. He makes eye contact with you. Fighting the urge to flee, you quickly look back at your screen. You missed placing two blocks. You will not make high score. Your mind begins conjuring swear words without biblical precedent...it has *never* taken this long before.

The floppy drive finally stops whirring. You quit out of Tetris, eject the floppy, and reboot the machine. You leave the computer cluster and enter the hallway half expecting to be halted by university officials. But none are there. You think yourself silly. You think that there was no way it could have been found. But the reality is that you know all too well how to write a background process capable of catching you in the act and that is what makes you scared. Stepping outside the building, you breathe a sigh of relief in the midday sun. You made it this time, but maybe you were just lucky. Maybe it wasn't in the cards just yet. Like a junkie to drugs, you are drawn to these machines. They speak to you the way they speak to no one else. You put in your time. You paid your dues, and yet for some reason your vision is still shrouded in darkness. There is something they are not telling you. Perhaps it is something they don't even know. It is a question that nags at you like no other, and you sense

that the answer lies hidden somewhere within the deepest recesses of your soul, somewhere out of sight and just beyond your grasp. *There has to be a better way.*

Shortly before sundown that same day...

The dull roar of thunder reverberates somewhere far off in the distance as menacing storm clouds roil in from the west. They exhibit all the signs of a true nor'easter and threaten to engulf the entire city of New Haven. You swear you just felt a drop of rain hit your left shoulder. Reaching down, you feel for the disk at your side. The floppy is still there, its presence reassured at the touch of a thumb. The data it contains is dear to you, and you'll be damned if you're gonna let a little H_2O seep through your denim pocket and claim your catch of the day. So you decide to pick up the pace a bit.

The path you follow winds in and around, gently sloping downward as you go, eventually leading to a clearing that overlooks a stand of maples. The trees are enormous and have stood here for ages. At their center lies a lone apple tree. It is dwarfed by the older trees and is helplessly sheltered under a canopy of leaves. Having sensed your unexpected approach, a nearby squirrel dashes for the safety of a nearby tree. Before reaching the trunk, it fumbles over an apple and sends it rolling along the ground. The fruits around you give off a racy odor, a telltale reminder of the approaching change of season.

Had it not been for the disk, you would chance a brief pause underneath the eaves to contemplate greater things. Physics lectures always left you spellbound regarding the mysteries of the world. It was the dream of being struck in the head by a falling apple that guided you to this school in the first place, a dream that you summarily dismissed upon meeting your brilliant roommate. He is a National Merit scholar and received 1580 out of a possible 1600 on his SATs. The deduction was in the verbal section, and you always attributed it to his difficulty in comprehending the human condition. On many levels he is more machine than man, yet his inference engine is second to none. Physics is his second language and he speaks it fluently. You abandoned the idea of majoring in physics since the thought of taking the same classes as he was too much to bear, and since he had an uncanny ability to make you feel stupid without even trying. Answers to scientific problems just came *naturally* to him. Your hacking obsession combined with a thoroughly tenderized ego would do little to help you finish school.

A gust of wind billows through the trees. The limbs creak and sway in response, causing rain droplets to roll off their leaves. The water splashes onto your face and exposed arms, causing you to start. You realize that you had zoned out completely and had lost all track of time. Your eyes had stared off into space, fixated on some solitary trees, and subconsciously absorbed the surrounding scenery. You shrug in spite of yourself. No use in crying over spilled milk. Your true path has yet to be determined and there is no reason to worry about it now.

You shift the weight of your backpack to your other shoulder and leave the small wooded area behind. As always the students took Prospect Street back to Old Campus while you ventured along an overgrown yet shorter route, preferring to take the road less traveled. *Hypotenuse* action your roommate called it. Over time you discerned the shortest route between the Sloane Physics Lab and your dorm and it took you through more than one private yard, not to mention a vast cemetery. It saved you an innumerable number of backaches to be sure. Take aside any science student and you will hear the same tale of woe. The cumbersome textbooks are murderous to haul and the university couldn't place the science buildings at a more remote location if it tried.

The Payne Whitney Gymnasium looms ahead, shadowed by the black storm cover above. Were it not for the parked cars and street signs, the darkness could easily lead one to mistake it for a castle. Gulls from the nearby seashore circle above the parapets that line the rooftop. Some dive and soar, some pick up speed, and still others hover in place in blatant defiance of the wind. Nightfall descended prematurely on the city, and what had been just a few droplets of rain minutes before has turned into a veritable deluge. A small pack of students run through the stone archway at the base of the gym with newspapers outstretched overhead. The brunt of the storm is upon you and rainwater quickly seeps into every quarter. You break into a sprint down Tower Parkway in a last-ditch effort to keep your data dry.

The torrential rain pummels your body in sheets as you approach the backdoor of Morse College. You pass quietly into the building under cover of dusk and enter the underground labyrinth of steam tunnels and storage rooms. The humdrum of washers and dryers from a nearby laundry room fills your ears. You take a brief moment to wring what water you can from your clothing. After regaining your composure, you head down the narrow hallway and pass alongside the laundry room. It is empty and devoid of movement, save for a loose ball of lint circling beneath a ventilation shaft.