

Preface

GENTLE READER. Your interest in this book is understandable. Computer security has become one of the most important areas in the entire discipline of computing. Computers today are used not only in the home and office, but in a multitude of crucial and sensitive applications. Computers control long distance telephone conversations, the flow of information on the Internet, the distribution of electrical power to cities, and they monitor the operations of nuclear power plants and the performance of space satellites, to name just a few important applications.

We have become used to these small, quiet machines that permeate our lives and we take them for granted, but from time to time, when they don't perform their tasks, we immediately become aware that something has gone terribly wrong. Considering the complexity of today's computers and their functions, and considering especially the physical hazards that abound in the world, it is a wonder that our computers function at all, yet we expect them to be reliable and we entrust them with more and more delicate, sensitive, and complex assignments.

It is easy to disrupt a computer. Just brush your elbow accidentally against your desk and you may spill your cup of coffee on your computer. A power loss lasting a fraction of a second may lead to a head crash of the hard disk, resulting in a complete loss of the disk and all its data. Carelessness on the part of operators or administrators in a large computations center can cause a costly loss of data or even physical damage to expensive equipment. Yet all these dangers (and there are many more like them) pale in comparison with the many types of intentional criminal damage that we have come to expect and that we collectively associate with the field of computer security.

A term closely related to computer security is computer crime. A computer crime is an incident of computer security in which a law is broken. Traditionally, computer crime has had a low profile. After all, in a computer crime there are no smoking guns, no blood-stained victims, and no getaway cars. Often, such a crime is solved just by sheer accident. In contrast, computer security is a high-visibility discipline because it involves most of us.

Experience has shown that the more sophisticated a civilization is, the more vulnerable it is to natural or man-made disruptions. A tree that fell on power lines in

viii Preface

Ohio in August 2004 plunged 50 million people from Detroit to New York into darkness. A computer glitch at an airport on 26 December 2004 (the day this paragraph was written) caused the cancellation of 1100 flights of Comair, a subsidiary of Delta Air Lines, and similar examples abound. Our civilization depends more and more on computers, which is why any disruption of our computers is at least inconvenient and at worst catastrophic.

In the past, computer security violations, such as viruses and DoS (denial of service, Section 7.5) attacks were caused by hackers, most of whom were believed to be young adults who did this for fun or enjoyed the feeling of power and notoriety. However, it seems that this situation is rapidly changing. Security experts are warning that future attacks on computers may be planned and funded by terrorists (better called cyberterrorists) and may be devastating. A powerful hurricane, a huge earthquake, or a tsunami may kill many and wreak untold havoc, but a large-scale, concerted attack on key computers may bring the economy of an entire country to its knees, even though no one may actually get killed.

The reason for such dire predictions is our experience with computer security in the last two decades. We know that a single computer virus, perhaps written and released by a teenager living in a remote town in a distant country, can propagate quickly, infect a vast number of computers within hours, and cause economic damage in the billions (of Dollars, Euros, or whatever currency is affected).

Today, computers are responsible for the distribution of electrical power and for routing telephone conversations. They store information on passenger and cargo flights, on large cash transfers between banks, and on military plans, to name just a few crucial applications. It is generally agreed that a well-organized attack that takes over several important, sensitive computers may cause at least a temporary collapse of an entire country.

What makes this kind of attack attractive to organized terrorists is that it can be carried out from the comfort of their homes. There is no need to actually go anywhere, to obtain and use dangerous nuclear or chemical materials, or to smuggle anything across international borders. The fact that we depend so much on computers may be crucial to our future survival, and the least that we can do now is to learn as much as possible about potential threats to computers and how to defend against them.

Virus writing is a crazy activity. People who write viruses just don't consider the consequences of their actions. At the same time, I believe in the American constitution, and the first amendment, which gives people freedom to write and to talk, so I don't have a problem in the larger sense of people discussing or studying viruses.

—Peter Tippett (Symantec) in [Virus bulletin 05] May 1994 issue.

There is an ongoing debate about whether newly-discovered security holes and vulnerabilities in operating systems and communications software should be made public. Publicizing a security weakness allows users to avoid it until a patch is issued or a solution is found. On the other hand, it gives the bad guys ideas. So far, advocates of public exposure have had the upper hand, with the result that any item of news about a new computer security problem ignites a race between attackers and defenders. The following is a list of some of those races:

- **SNMP flaw.** A flaw in the Simple Network Management Protocol (SNMP) leaves open many network devices to attack. The flaw has not been widely exploited.
- **Microsoft SQL vulnerability.** A hole in a common component of Microsoft's SQL database software leaves PCs open to remote attack. Six months after it was found, the vulnerability was exploited by the slammer worm (see year 2003 in Appendix B).
- **Microsoft RPC flaw.** In July 2003, Microsoft published details of a flaw in the remote procedure call (RPC) functions of Windows. About three weeks later, the MSBlast worm arrived and exploited this flaw to infect as many as 10 million computers.
- **Microsoft LSASS flaw.** A hole in Local Security Authority Subsystem Service (LSASS) exposed personal computers running the Windows operating system. A month after it was revealed, the sasser worm hit the Internet and spread among computers that still had this hole (see year 2004 in Appendix B).
- **iFrame flaw.** In late October 2004, a security researcher discovered the existence of a flaw in Internet Explorer, a popular Web browser (page 61). Hackers with nothing better to do immediately exploited the vulnerability to compromise personal computers running this software.

Three types of persons are involved in computer security: experts who study this field and recommend preventive measures and solutions, the general public, which suffers from the breakdown of computer security, and the (mostly anonymous) perpetrators of the various misdeeds and attacks. Most of these perpetrators are known as *hackers*, which is why this important, popular term is discussed here.

From the dictionary

Expert: someone widely recognized as a reliable source of knowledge or skill whose judgement is accorded authority and status by the public or their peers.

The Hacker

Madame Curie once said “En science, nous devons nous intéresser aux choses, non aux personnes [In science, we should be interested in things, not in people].” Things, however, have since changed, and today we have to be interested not just in the facts of computer security and crime, but in the people who perpetrate these acts. Hence this discussion of hackers.

Over the centuries, the term “hacker” has referred to various activities. We are familiar with usages such as “a carpenter hacking wood with an ax” and “a butcher hacking meat with a cleaver,” but it seems that the modern, computer-related form of this term originated in the many pranks and practical jokes perpetrated by students at MIT in the 1960s. As an example of the many meanings assigned to this term, see [Schneier 04] which, among much other information, explains why Galileo was a hacker but Aristotle wasn't.

A hack is a person lacking talent or ability, as in a “hack writer.” Hack as a verb is used in contexts such as “hack the media,” “hack your brain,” and “hack your reputation.” Recently, it has also come to mean either a kludge, or the opposite of a

x Preface

kludge, as in a clever or elegant solution to a difficult problem. A hack also means a simple but often inelegant solution or technique. The following tentative definitions are quoted from the jargon file ([jargon 04], edited by Eric S. Raymond):

1. A person who enjoys exploring the details of programmable systems and how to stretch their capabilities, as opposed to most users, who prefer to learn only the minimum necessary.
2. One who programs enthusiastically (even obsessively) or who enjoys programming rather than just theorizing about programming.
3. A person capable of appreciating hack value.
4. A person who is good at programming quickly.
5. An expert at a particular program, or one who frequently does work using it or on it; as in “a Unix hacker.” (Definitions 1 through 5 are correlated, and people who fit them congregate.)
6. An expert or enthusiast of any kind. One might be an astronomy hacker, for example.
7. One who enjoys the intellectual challenge of creatively overcoming or circumventing limitations.
8. [deprecated] A malicious meddler who tries to discover sensitive information by poking around. Hence “password hacker” and “network hacker.” The correct term for this sense is cracker (which stands for criminal hacker).

Today’s computer hacker is often an expert in a computer-related field who finds a way to exploit a weakness or a vulnerability in a certain component of that field. This component may be a piece of hardware, part of the operating system, or a software application. Not all hackers are experts and not all are malicious. A notable example is Linus Torvalds, the creator of the well-known, free Linux operating system. Many Linux users will agree that this activity of Torvalds is a hack, but everyone (except commercial competitors) agrees that it is useful.

I think any time you expose vulnerabilities it’s a good thing.

—Janet Reno

Some security experts claim that today’s computer hackers should be termed crackers or intruders, but the general public and the media seem to love the term hacker. The word “cracker” is used to designate someone who breaks the security code of software, so that it can be used without pay. The term “intruder” is commonly used to indicate a person who breaks into a remote computer.

The following classification of the various hacker categories is informal and is by no means universally accepted.

- The highest category of hacker may be a brilliant programmer (although such a hacker may prefer the title of guru, cracksman, or wizard). Someone who is intimately familiar with a certain communications program, protocol, operating system, or encryption algorithm. Such a person can identify weaknesses or vulnerabilities and then come up with a clever, original way of penetrating a computer and inflicting damage. Alternatively, such an expert may develop ways and means to plug up security holes in software, or even completely rewrite a weak routine or procedure to make it invulnerable.

- The next category is that of the good programmer. Such a person hears of a new security threat, for example, a new type of virus, and may decide to “improve” it. A good programmer can disassemble the code of a virus, read and understand it, and come up with more “efficient” ways of employing the basic principle of the virus. Such a person may also be a good guy (a white-hat hacker) and work as a security expert. Disassembling and reading the code of a virus uncovers the vulnerabilities the virus exploits and leads directly to eliminating them.
- A script kid is a hacker with little or no programming skills who simply follows directions created by a higher-rank hacker or who uses a cookbook approach without fully understanding the principles and details of what he is constructing.
- A hacktivist is an activist who employs hacking to promote a cause. In 1995, a virus attached a political message “Stop all French nuclear testing in the Pacific” to the footer of letters printed from Microsoft Word, so users who trusted the computer and didn’t check their printouts became unwilling supporters of a cause.
- A sneaker or a gray-hat is a hacker who breaks security for altruistic motives or other non-malicious reasons. The darker the hat, the more the ethics of the activity should be considered dubious.
- The least harmful hacker is the white-hat type. This term is often used to describe self-appointed security gurus who attempt to break into computers or networks in order to find security flaws and inform the owners/administrators of the problem.

The following is a list of “tools of the trade,” methods, approaches, and special software used by hackers to gain unauthorized access to data, to computers, and to entire computer installations:

- Rogue software. These are computer programs especially designed to propagate among computers and either inflict damage or collect data and send it back to the hacker. They are also known as malware. The chief types of rogue software are viruses, worms, Trojan horses, and the various kinds of spyware. Each is described in one paragraph below.

Virus (Chapter 2, a term borrowed from biology). A program that invades a computer and embeds itself inside a host program, where it replicates and propagates from computer to computer, infecting each in turn. A virus spreads by infected removable disks, or over a network.

Worm. A program that exploits weaknesses in an operating system or in communications software in order to replicate itself on other computers on a network. A worm does not reside in a host program. Worms are discussed in Chapter 3.

Trojan horse. A program that seems useful, but has a backdoor, installed by its creator and employed later to gather information or to damage software. Examples are programs that mimic login sequences or that fool a user into downloading and executing them by claiming to be useful applications. This type of rogue software is described in Chapter 4.

Spyware is the general name assigned to a whole range of nasty software that runs on a computer, monitors its users’ activities, collects information such as keystrokes,

xii Preface

screen dumps, and file directories, and either saves this information or sends it to a remote location without the knowledge or consent of the computer owner. Spyware is described in Chapter 9.

- **Scanning.** This term refers to software and equipment that methodically probes computers on the Internet for vulnerabilities. Two of the main tools used for this purpose are a vulnerability scanner and a sniffer. They are described here.

Vulnerability scanner. A program designed to quickly check computers on a network for known weaknesses. A port scanner (Section 7.2) is a special case. It is a program that attempts to find open ports on a target computer or ports that are available to access the computer. A firewall is a piece of hardware or software that defends computers from intruders by closing off all unused ports.

Sniffer. A program that captures passwords and other data while the data is in transit either within the computer or between computers or routers on a network.

- **Exploit.** A ready-to-run program that takes advantage of a known weakness. These can often be found in hackers' newsgroups.

- **Social engineering.** A general term for methods that exploit human weaknesses. A hacker may discover someone's password by calling and pretending to be an official, by looking over someone's shoulder while a password is being typed, or by sending email that poses as an official notice asking for sensitive information. Bribing and blackmailing are also included in this class. Even though no special software may be needed and no software weakness is exploited, this is still a powerful tool used by many miscreants. Social engineering (page 204) is a wide class that includes, among others, the following methods:

Shoulder spying (or shoulder watching or surfing). A hacker enters a secure computer installation or a restricted computer lab (often disguised as a pizza delivery man) and looks behind users' shoulders for passwords typed by them or being taped to the sides of computer monitors.

Optical spying. The hacker watches from a nearby room or building, perhaps with a binocular, and tries to read keystrokes typed by legitimate users.

Scavenging (or dumpster diving). Hackers have been known to collect trash and examine it for passwords and credit card numbers (see also page 205).

- **Side-channel attacks.** A hacker can spy on a secure installation "from the side" by capturing and listening to information that is continuously and unintentionally leaked by electronic devices inside. The basis of this approach is the well-known fact that people are nosy and machines are noisy. Side-channel methods are discussed in Section 1.1, but the following are typical examples.

Eavesdropping. A hacker, often disguised as a telephone company repair man, enters a computer room and plants devices that later transmit to him useful data on the activities of users. Such devices may include radio transmitters, acoustic microphones (Section 1.1.1), and cameras.

Acoustic keyboard eavesdropping. This recent, sophisticated approach to spying employs the little-known fact that each key in a keyboard emits a slightly different sound when pressed. Recording the sounds of keys with a sensitive microphone may

enable a hacker to analyze them by computer and discover the actual keys pressed by a user. A similar approach is to use a high-gain antenna outside a building to receive the electromagnetic waves emitted by CRT monitors inside and analyze them to recreate the displays. These methods are discussed in Section 1.1.1.

Root kit. A program especially designed to hide the fact that a computer's security has been compromised. A root kit may replace an operating system program, thereby making it impossible for the user/owner to detect the presence of the intruder by looking at activity inside the computer.

Leet (l33t speak). Slang used by hackers to obfuscate discussions in newsgroups and other "gathering places" on the Internet. Examples of leet are "warez" (for pirated software), "pr0n" for pornography, and "sploit3" for exploits. See Appendix A.

A honeypot is the name of the opposite tool. A honeypot is a server that acts as a decoy, attracting hackers in order to study their methods and monitor their activities. Security workers use honeypots to collect valuable information about new methods and tricks employed by hackers to break into computers.

Hacker motivation and psychology. Why does someone become a hacker? In most cases, hacking involves much study (of programming, communications protocols, and the internal workings of operating systems), expense (the hacker must have a computer and normally also Internet connection), time, and effort.

We all hear about teenagers, high-school kids who spend days in front of a computer, trying to hack into another computer for the satisfying feeling of achievement, of (false) success. This type of hacker, who "works" for the challenge of penetrating a secure computer or a secret computer installation, for the sheer pleasure and the rush of adrenalin, may also be an adult. There are many known cases of disgruntled employees who plant a time bomb in sensitive software and schedule it to go off when they are terminated. Another category is a computer-savvy person who hears about successful hacking episodes and decides to try and make money this way. Spies are also potential hackers. A spy may acquire a great deal of useful information by hacking into a military computer and can do it "from the comfort of his home." A case in point is discussed by [Stoll 88, 90, 04]. Various kinds of terrorists, both home grown and foreigners, are also believed to be active in hacking, because this is one activity that causes much harm with relatively small risk for the hacker. Finally, there is organized crime, as the following quote (from [Brenner 02]) makes clear:

"The Internet is still in its infancy, but we have already seen large segments of human activity migrate wholly or partially into cyberspace, a trend that will only accelerate. Criminal activity has also moved into cyberspace, and this, too, is a trend that will only accelerate; lawbreakers will shift much of their activity into cyberspace because it will increasingly be the venue where illicit profits are to be made and because it offers operational advantages."

Computer crime is perpetrated not just by hackers. Many honest people who have access to computers with important data are tempted to commit a crime in order to enrich themselves. Inevitably, some yield to the temptation. The following story from the 1960s (which may even be true) is just one of many examples. A low-level programmer in a bank had noticed that the quarterly interest payments on the many savings accounts held by the bank (there were tens of thousands of such accounts)

were computed to four decimal places, then rounded off. Thus, anything above \$0.0075 was rounded up to the next cent and any amount below that was truncated to the nearest cent. In other words, anything below three quarters of a cent earned in interest was going back to the bank. The programmer simply modified the source code of the program that did these computations, directing it to send all this extra money to his account. The story (there are many versions of it) goes on to say that the programmer was unmasked only because he bought an expensive car, too expensive for his salary, and parked it prominently in the bank's parking lot. This story may or may not be true, but in response to it many banks have instituted a policy that requires each programmer to take his annual vacation every year, at which time any software the programmer worked on is scrutinized by special auditors.

◇ **Exercise Pre.1:** Who audits the auditors?

(A joke. Today, after decades of inflation, it is even possible for a bank programmer to simply take a penny or two from each bank account without the account's owner noticing or caring about the loss, and channel this money to his private account. Before going on vacation, the programmer can clean his program for the benefit of the auditors. While on vacation, the programmer enjoys the extra money. Upon returning, the program can be doctored again. Naturally, this author does not condone such behavior, but it helps to improve the vacation patterns of low-paid bank programmers. On second thought, is this just a joke?)

Another, even more bizarre story is about a pair of programmers who started appearing to work in a matching pair of Rolls-Royces. The company's executives immediately became suspicious and started an investigation. When the pair heard of it, they promptly bolted. However, in spite of a long and careful investigation, nothing untoward was ever discovered. If the two programmers were guilty, they managed to completely cover their tracks, and got scared needlessly.

In the early days of hacking and breaking into computers, some security experts maintained that "hackers have done less damage to corporate computer systems than overflowing lavatories." Today, such a claim seems ludicrous. The damage done to computers, to networks, to individuals, and to the economy is getting worse and has become a global concern. Fighting it involves governments, law enforcement agencies, and security experts all over the world.

For more information, see *How to Become a Hacker* and *Brief History of Hackerdom* by Eric Raymond [Raymond 04].

Not all computer crime and attacks are perpetrated by hackers. Much harm is done by insiders, trusted employees who do it for a variety of reasons. This is the human side of computer security. The history of computer crime is riddled with stories about users who take their frustration out on the computer. They drop it on the floor, shoot it, pound it with a hammer, and even urinate on it, just to vent their feelings and frustration. Some employees strike at their machines as a way to get back at the boss, while others act out of political convictions and allow their fellow party members to sabotage equipment. However, the main reason for insider computer crime is money. An employee or a trusted consultant suddenly realize they have enough knowledge to

induce a computer into printing a check, transferring money to their account, or releasing information that can later be sold (such as a mailing list or credit card numbers) and this temptation may prove too much. Such a treacherous insider suddenly turns into a living Trojan horse, as dangerous as those discussed in Chapter 4. The best an employer can do to defend against such employees is to compartmentalize information, to make sure an employee knows only as much as he or she needs to know for their jobs. This policy is difficult to implement in practice, it adversely affects employees' morale and productivity, and it is not full proof.

We have all heard of bank robbers, but one of the most notorious bank robbers, one who kept the title “biggest computer fraud” in the Guinness Book of World Records [Guinness 04] from 1978 to 1999, was someone called Stanley Rifkin, a name most of us would have trouble recognizing. He is virtually forgotten today, perhaps because he didn't use a gun in his exploit and didn't even hack the bank's computer. He was a consultant to the now defunct Security Pacific National Bank in Los Angeles and in this capacity he learned some of the codes used by bank personnel to make large money transfers. He used this knowledge to call the employees in the wire transfer room, pretending to be Mike Hansen, a member of the bank's international department, and con them into transferring ten million dollars to a temporary account that he had previously opened. He later transferred the money to Switzerland and used it to buy diamonds that he then smuggled back to the United States. He was caught by the FBI very quickly, but only because he had bragged about his exploit to his lawyer, trusting the confidentiality of attorney-client relations. The lawyer notified the FBI and Rifkin was arrested. The final twist of this story is that the bank didn't even miss the money when notified by the FBI of the successful solution of this crime.

- ◇ **Exercise Pre.2:** Imagine that you are an operator of a large computer. You've been with the company for years, and you have suddenly been switched to the night shift, forcing you to sleep during the day so you rarely get to see your family. You don't want to quit, because in just a few years you'd be eligible for retirement. What can you do to improve your lot?

FBI: Why do you rob banks?

Willie Sutton: Because that's where the money is.

<http://www.fbi.gov/libref/historic/famcases/sutton/sutton.htm>.

Computer security: an example

The following incident illustrates the serious nature of Internet security, hacking, and cyber vandalism. On 1 April 2001, a Chinese military jet collided with an American spy plane. The Chinese pilot was killed and the American plane was crippled and had to land in Chinese territory. The crew of 24 was held by China and released 11 days later.

The diplomatic row between the two countries was well publicized, short lived, and did not lead to any long-term animosity. In contrast, the cyber war between Chinese and American hackers was less known, was very intense, and has inflicted much damage to Web sites on both sides. American hackers started scanning Chinese Web sites,

looking for vulnerabilities that make it possible to deface or hijack a site. A typical attack ended up leaving offending messages on the target site.

In response, a Chinese hacking group calling itself the Honker (Chinese for “red user”) Union of China decided to retaliate. The Honker Web site [honker 04] prompted its members for action with the message “We are obligated to strike back with utmost force after such provocation by American hackers.” The group managed to disable many American Web sites and left pro-China messages in others. Among the victims were the Department of Labor, Department of Health and Human Services, and the Web site of the United States Surgeon General. The White House Historical Association Web site (<http://www.whitehousehistory.org/>) was also defaced, presumably because the Chinese assumed it to be a government site (it is a charitable nonprofit institution dedicated to the understanding, appreciation, and enjoyment of the White House).

To an outside observer, this and similar incidents serve as a useful lesson. They do not involve any physical casualties, while keeping Web site owners and administrators on their toes. To the victims, however, this affair seemed at best an annoyance.

About this book

This book is intended as a starting point for those familiar with basic concepts of computers and computations who would like to extend their knowledge into the realm of computer and network security. The book is primarily a textbook for undergraduate classes on computer security. It is mostly nonmathematical and makes no attempt to be complete. The only prerequisite for understanding the material presented here is familiarity with the basic concepts of computers and computations such as (1) the organization of data in bits and bytes, (2) data structures (arrays, trees, and graphs), and (3) network concepts such as IP numbers, input/output ports, and communications protocols.

Timing. The many phrases “at the time of this writing” found in the book refer to the period from October 2004 to mid 2005 during which this book was written.

Special features that enhance the textbook aspect of the book are the many exercises sprinkled throughout the text, the virus timeline (Appendix B), and the Glossary. Another attractive feature is the jokes (check the index). There are no riddles.

A note on references. The text refers to many resources using notation of the form [Thompson 84] where the 2-digit number is a year. All the references are listed in the Bibliography and many are Web sites. As we all know, Web sites tend to have a relatively short life, so by the time this book is in your hands, many of the references may be broken links. However, given the context of a reference, an Internet search engine may locate a cached copy of the original page or a similar page. Don’t give up easily.

An interesting (and, I believe, also original) feature of this book is its minimal use of the vague term “system.” This word is used only (1) in connection with well-defined or commonly-used terms such as “operating system,” “file system,” and “notational system,” (2) when it is part of names of organizations, or (3) when it is included in a quotation. Many texts use this vague term liberally, thereby confusing the reader. Sentences such as “In addition, the blah flood may exhaust system memory, resulting in a system crash. The net result is that the system is unavailable or nonfunctional,”

are confusing. Instead of “system” the author should specify what is being discussed, whether it is a computer, a piece of software, a router, or something else. Here is what William Strunk [Strunk 18] has to say about this term.

System. Frequently used without need.	
Dayton has adopted the commission system of government	Dayton has adopted government by commission
The dormitory system	Dormitories
—William Strunk Jr., <i>The Elements of Style</i> .	

While I was at it, I also avoided the use of the cliché “basically,” employing “essentially” or “fundamentally” instead.

On the other hand, the term “user” is a favorite in this book.

Why is it drug addicts and computer aficionados are both called users?
—Clifford Stoll.

Following is a short description of the chapters and appendixes of the book.

- Chapter 1 is a collection of topics that have to do with the physical security of computer hardware, computer networks, and digital data. The topics discussed cover a variety of issues ranging from computer theft and static electricity on carpets to laptop security.
- Chapter 2 is the first of the chapters on rogue software (the term *malware* is often also used). The chapter is devoted to computer viruses, and it covers all the important aspects of this unusual type of software. The various types of viruses, the way viruses propagate, the damage they may inflict (their payload), and the people who write them, are among the topics covered in this chapter.
- Another type of rogue software, namely worms, is the topic of Chapter 3. Techniques for worm propagation are discussed and the historically important Internet worm is described.
- Trojan horses are the topic of Chapter 4. The discussion concentrates on the types of damage done by this type of malware and on how Trojan horses are installed on a computer. Of special interest is Section 4.3 that describes an interesting technique for bugging or rigging a compiler. A Trojan horse can be embedded inside a compiler in such a way that certain programs compiled by it will be infected with the horse, yet nothing suspicious remains in the source code of the compiler itself and even a recompilation of the compiler does not get rid of the malicious software secretly embedded in it.
- Chapter 5 is full of examples of malware. About a dozen examples of viruses, worms, and Trojans are discussed and described in detail. Many (shorter) descriptions can be found in Appendix B.
- The important topics of preventing malware and defending against it make up Chapter 6. Among the methods discussed in this chapter are backing up files, anti-virus software and its applications, activity monitors, vaccines, and file permissions. The interesting topic of hoaxes is also included in this chapter.

xviii Preface

- Network security is the topic of Chapters 7 through 10. Chapter 7 starts this important subject with a detailed discussion of important threats that relate to networks. Topics such as port scanning, spoofing, password cracking, firewalls, and denial of service (DoS) are described and analyzed.
- Chapter 8 concentrates on authentication. Both local and remote methods for authentication are included. Of special interest are the biometric authentication techniques of Section 8.2.
- Spyware, the topic of Chapter 9, is a relatively new threat and is already serious enough to merit its own discussion and methods of defense. Material on spyware and terrorism and on remote reporting is also included, as are several varieties of spyware such as adware and researchware.
- Chapter 10 tries to familiarize the reader with the growing crime of identity theft. The topic of phishing is also covered in detail, including examples.
- Privacy and trust in the online world are the topics of Chapter 11. General privacy concerns as well as children's privacy and safety are discussed, together with how to generate trust in visitors to Web sites (and how to keep it). Notice that privacy issues are also discussed in Section 1.5.
- Chapter 12 is an introduction to cryptography and how it works. The chapter starts with the concepts of cipher and code and follows this by examples of old monoalphabetic and polyalphabetic ciphers. The important method of the one-time pad and the problem of key distribution are discussed next. The chapter continues with the principles of public-key cryptography, RSA encryption, and the all-important secure socket layer (SSL) protocol.
- Appendix A introduces “l33t Speak” (pronounced “leet”), a language or a notational system widely used by hackers.
- Appendix B is a detailed virus timeline. The history of viruses and other types of rogue software is traced from its infancy in the late 1940s to the present day (early 2005), stressing “firsts” such as the first stealth virus and the first boot sector infector.

The book's Web site, with an errata list and Bib_TE_X information, is part of the author's Web site, located at <http://www.ecs.csun.edu/~dsalomon/>. Domain name `www.DavidSalomon.name` has been registered and is used as a mirror. The author's email address is `dsalomon@csun.edu`, but `<anyname>@DavidSalomon.name` is an alternative address.

Disclaimer. This is not a fact-free book. A book like this could not have been written without the help of many people, but this book was! As a result, the author is the only one responsible for both the correct and useful material in the book and for the many errors that may or may not be discovered in the future.

Lakeside, California

David Salomon

I offer this advice without fee; it is included in the price of this book.

—Muriel Spark, *A Far Cry From Kensington* (1988).



<http://www.springer.com/978-1-84628-193-8>

Foundations of Computer Security

Salomon, D.

2006, XXI, 369 p., Hardcover

ISBN: 978-1-84628-193-8